FI Scientific Library

Sietske Tacoma

Automated intelligent feedback in university statistics education

Faculteit Bètawetenschappen **FI**

Automated intelligent feedback in university statistics education

ico
This research was carried out in the context of the Dutch Interuniversity Centre for Educational Research

Review Commission
Prof. dr. H. J. A. Hoijtink
Prof. dr. W. R. van Joolingen
Prof. dr. L. Kester
Prof. dr. U. Kortenkamp
Dr. A. J. Cabo

Sietske Tacoma

Automated intelligent feedback in university statistics education / Sietske Tacoma – Utrecht: Freudenthal Institute, Faculty of Science, Utrecht University / FI Scientific Library (formerly published as CD-β Scientific Library), no.105, 2020.

Dissertation Utrecht University. With references. Met een samenvatting in het Nederlands.
ISBN: 978-90-70786-45-8
Keywords: Domain reasoner; Feedback; Hypothesis testing; Intelligent tutoring systems; Statistics education; Open student models

Cover design: Vormgeving Faculteit Bètawetenschappen
Printed by: Xerox, Utrecht

Automated intelligent feedback in university statistics education

Automatische intelligente feedback in universitair statistiekonderwijs

(met een samenvatting in het Nederlands)

Proefschrift
ter verkrijging van de graad van doctor aan de Universiteit Utrecht
op gezag van de rector magnificus, prof.dr. H.R.B.M. Kummeling,
ingevolge het besluit van het college voor promoties
in het openbaar te verdedigen
op woensdag 25 november 2020 des middags te 2.30 uur

door

Sietske Gerkje Tacoma

geboren op 18 juni 1987
te Borne

Promotoren:

Prof. dr. P. H. M. Drijvers
Prof. dr. J. T. Jeuring

Contents

CHAPTER 1 General introduction

United Nations' Fundamental Principles of Official Statistics, Principle 1:

Official statistics provide an indispensable element in the information systems of a democratic society, serving the Government, the economy and the public with data about the economic, demographic, social and environmental situation. To this end, official statistics that meet the test of practical utility are to be compiled and made available on an impartial basis by official statistical agencies to honour citizens' entitlement to public information (United Nations Statistics Division, 2014).

1.1 Introduction

The quotation above, Principle 1 from the United Nations' Fundamental Principles of Official Statistics, illustrates that statistics has become indispensable in today's society. Due to the emergence of powerful digital tools to collect, store, analyze, and represent big datasets, statistical analysis has become tremendously important for governments and companies to inform decisions. Consequently, people nowadays are confronted more and more with statistical information in the media. Moreover, statistical methods are essential for conducting research in almost all scientific disciplines. Because of this ubiquity of data and statistics, education needs to prepare students for conducting and interpreting statistical analyses. Introductory statistics courses are, therefore, an essential element in many university study programs (Castro Sotos, Vanhoof, Van den Noortgate, & Onghena, 2007).

With the advancement of technology and the resulting changes in statistical practice, the goals and nature of statistics education have been changing as well (Chance, Ben-Zvi, Garfield, & Medina, 2007). Only a few decades ago, being able to use formulas to calculate statistics (e.g., means, standard deviations, or *t*-values) was a valuable skill for a statistician. Nowadays, however, statisticians usually outsource such calculations to calculators and computers. Meanwhile, the multitude of statistical techniques currently available requires knowledge and skills to choose the appropriate techniques, given the context and the questions at stake. Developing such knowledge and skills requires learning some

statistical techniques, but it is more important to understand the statistical concepts and principles underlying these techniques (Carver et al., 2016). Rather than knowing exactly how to manipulate statistical formulas, students, as well as professionals and citizens, need to know why data and statistical formulas are needed, how these can inform decisions and how variability in data can influence the results that statistical formulas – or software using these formulas – produce. We refer to this combination of knowing, using, and reasoning with statistical concepts as *statistical proficiency*. This includes, but is not limited to, statistical literacy, which can be described as knowing basic statistical terms, understanding simple statistical symbols, and being able to interpret different representations of data (Garfield et al., 2008).

Developing statistical proficiency is not easy. Success rates for introductory statistics courses are regularly low, meaning that for many students these courses are obstacles in obtaining their bachelor's degree (Murtonen & Lehtinen, 2003; Tishkovskaya & Lancaster, 2012). Students struggle to understand the large number of abstract concepts, such as probability distributions, sampling variability and confidence intervals (Castro Sotos et al., 2007). Even more problematic is the ability to integrate such abstract concepts into complex chains of reasoning involving uncertainty (Falk & Greenbaum, 1995). As an example, consider the method of null hypothesis significance testing, which is widely applied in scientific research. In addition to knowledge of, among other things, sampling variability, significance level, and p-values, applying this method requires the ability to reason using conditional statements (e.g., "under the assumption that the null hypothesis is true, this outcome, or a more extreme one, is very unlikely"). A final issue that may hinder students in appropriately applying statistical techniques to reason about real-world problems is that formal definitions of statistical concepts, such as variability, often conflict with students' prior, informal knowledge and their view of the real world (Garfield & Ahlgren, 1988).

It is because of these issues that many students still perceive statistics as a disconnected collection of methods and techniques, rather than as a problem-solving and decision-making process that uses these methods and techniques (Carver et al., 2016). In higher education, matters

are even more complicated, because of the typically large student group sizes in introductory statistics courses. This makes it unachievable for teachers to provide individual guidance and feedback, which could support students in developing statistical proficiency. Apart from teachers, there is, however, another agent that could provide sophisticated individual guidance and feedback: the computer.

1.2 Feedback in computer-based learning environments

Over the past decades, many computer-based learning environments have been developed to facilitate learning of many topics at all educational levels. One of these environments' largest promises for enhancing learning is the provision of individualized and timely feedback on student work (Pardo, 2018; VanLehn, 2011). Fulfilling this promise is not straightforward, though, because there are many design choices to make when implementing feedback, regarding specificity, timing, type and complexity of information provided, and visual presentation (Shute, 2008). These design choices have been found to influence feedback effects: while feedback from computer-based learning environments mostly influences student learning in a positive way, implementations with negative effects have been reported as well (Van der Kleij, Feskens, & Eggen, 2015). In this thesis, therefore, we explore whether and how feedback by computer-based learning environments can support students in developing statistical proficiency.

Before turning to the specific domain of statistics, we start by outlining how theory postulates that feedback may contribute to student learning in general. To this end, we consider the following feedback definition by Pardo (2018):

> A process to positively influence how students engage with their work in a learning experience so that they can improve its overall quality with respect to an appropriate reference and increase their self-evaluative capacity. (Pardo, 2018, p. 433)

An important aspect of this definition is that feedback is considered a process. More specifically, it involves phases of evidence collection, information delivery and feedback assimilation. Information delivery may

be the element of the feedback process that comes to mind first when thinking of feedback: an agent (which may be a teacher, but in our case is a computer-based learning environment) provides information to a student concerning the student's learning process. Before information can be delivered, though, evidence about the student's learning process needs to be collected, to allow for tailoring the feedback information to the student's individual needs (Gikandi, Morrow, & Davis, 2011). After information delivery, the feedback process enters a new phase: the student needs to assimilate the information and decide which, if any, subsequent actions to carry out (Timmers, Braber-van den Broek, & Van den Berg, 2013). This assimilation phase may result in changes in the student's knowledge, skills, beliefs, attitudes, goals, strategies, and tactics (Pardo, 2018), which can be seen as the ultimate goal of providing feedback.

Let us now consider how this feedback process can be shaped within computer-based learning environments. As well as inducing changes in statistical practice, as discussed in section 1.1, the advancement of technology has also incited the development of artificial intelligence techniques to provide sophisticated intelligent feedback. Computer-based learning environments that employ artificial intelligence techniques to generate feedback are called Intelligent Tutoring Systems (ITSs). In ITSs, two general feedback types can be distinguished: *inner loop feedback* on steps within tasks, and *outer loop feedback* over complete tasks or multiple tasks at once (Santos & Jorge, 2013; VanLehn, 2006). Inner loop feedback typically provides information about the correctness of a (partial) solution, combined with guidance on how to resolve mistakes and how to proceed in solving the current task. Outer loop feedback concerns the student's current knowledge state regarding the domain and, possibly, the selection or suggestion of appropriate subsequent tasks or study activities. For both types, positive effects on student learning have been reported (see, for example, VanLehn (2011) for inner loop feedback and Bull & Kay (2016) for outer loop feedback). It is, therefore, not surprising that both feedback types have been implemented in computer-based learning environments that are used in educational practice today.

1.3 Aims and research question

Given the promising general affordances of automated feedback in computer-based learning environments, the main aim of this research project was to investigate whether and how these techniques could also be employed to foster the development of students' statistical proficiency. To this end, we implemented automated inner and outer loop feedback in two university statistics courses for first-year social sciences students. The guiding research question for this investigation was:

> *How can automated intelligent feedback support first-year university students in developing statistical proficiency?*

To answer this research question, we focused on three aspects of the implementation process: feedback design, students' use of the feedback, and the effects of feedback use on the students' statistical proficiency. We outline the goals related to these three aspects below.

Concerning the first aspect, feedback design, the goal was to investigate how artificial intelligence techniques – such as model-tracing, constraint-based modeling and user modeling – could be employed to generate feedback that addressed statistical proficiency. This raised questions related to the first and second phase of the feedback process described in section 1.2: which evidence about student learning can be collected and how can this evidence be automatically analyzed to generate useful feedback information? Addressing statistical proficiency was regarded as a challenge, since ITSs have a reputation of promoting procedural skills rather than conceptual understanding (Salden, Aleven, Renkl, & Schwonke, 2009). Therefore, recommendations from statistics education literature, such as the use of real contexts and datasets (Ben-Zvi, 2000; Carver et al., 2016), were deemed important to enable a focus on statistical proficiency. Incorporating these recommendations was expected to result in instructional content that contained many clusters of closely related tasks referring to the same context.

The instructional design structure that ITSs typically use is quite different: a collection of mutually independent, interchangeable items. The question was whether in these two different instructional design structures,

students' knowledge could be inferred from students' answers in the same way. In other words, a goal concerning feedback design was to investigate whether applying artificial intelligence techniques on an instructional design of clusters of tasks would yield valid inferences about student knowledge. Another point for consideration, in the context of university statistics education, was that teachers in higher education are usually responsible for designing their own courses. To do so, these teachers need to be able to adopt the designed system and to adjust course contents to address the specific needs and interests of their students. Hence, the feasibility of adopting the designed feedback implementations for university teachers was an important consideration during the design.

The second aspect of feedback implementation concerned evaluating the students' use of the feedback. As illustrated by Pardo's feedback definition in section 1.2, the feedback process does not stop once feedback information has been generated and delivered to the student. To benefit from the available feedback, students need to actively notice its availability, assimilate the information and use it to decide what to do next (Timmers et al., 2013). Various factors, such as motivation and accessibility of feedback information, may influence whether and how students engage in such behavior. At this point, the goal was to investigate whether and how the students used the available inner and outer loop feedback during their engagement with the computer-based learning environment in their statistics course. This entailed the quantity as well as the quality of their feedback use. Quantity of feedback use is straightforward to define and observe, in terms of the frequency and duration of interactions with the feedback. Quality of feedback use is somewhat more implicit: it can be inferred from the students' actions in the computer-based learning environment that occur immediately after interacting with the feedback. These actions could reflect changes in student knowledge evoked by the feedback, for example when a student corrects a mistake after receiving inner loop feedback. They could also reflect feedback effects on students' strategies, for example when a student starts to work on a new task concerning a specific topic immediately after receiving outer loop feedback. In this sense, these subsequent actions are considered indications of how feedback use may influence both students' knowledge and their learning behavior.

Regarding the third aspect of feedback implementation, changes in students' learning behavior were expected to eventually result in changes in students' statistical proficiency as well, which was the ultimate goal of implementing automated intelligent feedback in this research project. In the case described above, by evoking a decision to start working on a certain task, the feedback may encourage the student to practice more, which, in turn, may lead to more opportunities for learning. To assess whether the feedback did indeed induce such changes, we wanted to consider the effects of the implemented feedback on students' statistical proficiency. The goal here was to evaluate whether students receiving the designed automated intelligent feedback indeed developed better statistical proficiency than students who did not receive such feedback. The two types of implemented feedback, inner and outer loop feedback, were evaluated separately as well as in combination. This allowed for identifying the effects of both types, but also for evaluating whether the two interacted and whether students benefited from the combination of both types.

1.4 Methods and educational setting

The goals of designing, providing and evaluating automated intelligent feedback to address statistical proficiency in higher education align well with characteristics of design-based research. In this research paradigm, the development of theories about domain-specific learning and the design of means to support that learning go hand-in-hand (Bakker & Van Eerde, 2015). Design-based research is a cyclic process of repeated design, implementation and evaluation. In this process, theoretical ideas about student learning inform the design and are subsequently adapted, informed by the implementation and evaluation.

In our research project, both inner and outer loop feedback were designed and implemented in cycles: one cycle for inner loop feedback only, two cycles for outer loop feedback only, and one final cycle for the two feedback types combined. The two cycles involving only outer loop feedback were explorative in nature, to investigate the feasibility of the selected design approaches and to identify the various ways in which students used the outer loop feedback. The most important data source for these explorations were the logs of the students' interactions with the

computer-based learning environment. These data were supplemented by students' answers to a questionnaire and their exam results. Data analysis in these cycles focused on identifying patterns in these data through, among other methods, learning curve analysis (Martin, Mitrovic, Koedinger, & Mathan, 2011) and categorization of students according to the quantity and quality of their feedback use. The other two cycles, that is, the cycle involving only inner loop feedback and the final cycle for the two feedback types combined, had an evaluative nature. To evaluate the effects of the designed feedback on student learning, these cycles were set up as randomized controlled experiments. Like in the exploratory cycles, logs of student work in the computer-based learning environment were an important data source. Based on these logs, student-specific measures such as time-on-task and number of solved tasks were calculated. Additionally, exam results were used as the final measure of the students' statistical proficiency. Multiple linear regression models were used to assess the effects of feedback types, student characteristics and their interactions on the students' learning processes and their statistical proficiency.

The educational setting for this research project was formed by two first-year statistics courses for students enrolled in social sciences bachelor programs at Utrecht University: Methods and Statistics I and Methods and Statistics II. Design and implementation cycles within these courses took place in 2016, 2017 and 2018. In the first inner loop and first outer loop cycles, participants in this study were subgroups of the students enrolled in the courses, resulting in groups of 160 to 300 students. In the second outer loop cycle and in the final cycle for both feedback types all students enrolled in the courses were asked to participate in the research project. This resulted in groups of between 500 and 600 participating students. In all three years, students received weekly online homework sets about statistical topics. These homework sets were offered in the Digital Mathematics Environment (DME), a computer-based learning environment developed by the Freudenthal Institute (Drijvers, Boon, Doorman, Bokhove, & Tacoma, 2013). Tasks in the homework sets addressed, for example, selecting appropriate measures of center and spread for given variables and testing hypotheses for given situations and samples. The tasks were designed by the teachers of the course and used a variety of interaction types, such as number input, multiple choice tasks and drag-and-drop

tasks. Students received immediate verification feedback in all tasks, informing them on whether their answer was correct, but not about what the correct answer was. The courses were concluded with a final exam consisting of multiple choice items.

The two feedback types designed in the context of this research project were added to these homework sets. Inner loop feedback was designed in the form of a *domain reasoner* for hypothesis testing (Goguadze, 2011). The topic of hypothesis testing is central in many introductory statistics courses, but especially understanding the logic of the stepwise hypothesis-testing procedure and the role of the abstract statistical concepts involved is challenging for students (Falk & Greenbaum, 1995). The aim of implementing the domain reasoner was, therefore, to especially address this logical reasoning within the hypothesis-testing procedure. The domain reasoner feedback was used in nine tasks on hypothesis testing in the homework sets. Originally, these tasks provided pre-structured hypothesis tests to students, in which students were asked to complete all pre-defined steps. For this research project, these tasks were replaced by open-ended versions, in which students were challenged to construct the hypothesis tests step-by-step. Inner loop feedback provided information about the correctness of each step and hints on how to proceed in adding a next step.

Outer loop feedback was designed in the form of *inspectable student models* (Bull & Kay, 2016). Informed by the students' correct and incorrect attempts on all tasks in the homework sets, these student models provided the students with an overview of their current estimated knowledge level concerning important statistical topics. The student models were not automatically shown to students, but students always had access to their student models while working in the DME. Furthermore, on the final page of each homework set, students were encouraged to view their student models and to use them to decide on subsequent study steps.

1.5 Thesis outline

The four design research cycles outlined in section 1.4 are discussed in separate chapters of this thesis. We now outline how these four cycles

align with the goals concerning the three implementation aspects we discussed in section 1.3: feedback design, students' use of the feedback and feedback effects on students' statistical proficiency. For a schematic overview, see Figure 1.1.

In **Chapter 2** we address the design, the use by students and the direct effects of inner loop feedback: the domain reasoner for hypothesis testing. This chapter concerns a randomized controlled trial with 314 first-year psychology students, 163 receiving domain reasoner feedback and 151 receiving stepwise verification feedback only. It addresses the following research question:

2.1 Does automated intelligent feedback about the logic of hypothesis testing contribute to student proficiency in carrying out hypothesis tests?

Although all three implementation aspects were addressed in this cycle, no exam results were used yet. In this cycle, feedback effects only concerned direct effects on the students' work within the DME. More specifically, we compared the number of hypothesis-testing tasks students solved and the number of errors students made in these tasks between students who did and did not receive domain reasoner feedback. Longer-term feedback effects were assessed in the final cycle, in combination with outer loop feedback effects, and are discussed in Chapter 5. Before moving to this final cycle, we first address the design and implementation of outer loop feedback.

The design of outer loop feedback, in the form of inspectable student models, is discussed in **Chapter 3**. As outlined in section 1.3, the homework sets in our study contained many sets of tasks that were clustered around the same real datasets and contexts, while many ITSs rely on sets of mutually independent tasks. In Chapter 3, we investigate the feasibility and validity of implementing inspectable student models in this different instructional design. In this exploratory study, DME log files and questionnaire results from 160 first-year students in educational studies were used to address the following research questions:

3.1 Are inspectable student models suitable for implementation in didactically grounded, sequential statistics modules consisting of closely related tasks?

3.2 How can didactical analysis inform design of inspectable student models and, vice versa, how can student model evaluation methods inform didactical design?

The findings here informed a new design research cycle for the inspectable student models. In this cycle, our research focused on the students' use of the inspectable student models. **Chapter 4** discusses feedback use of 599 first-year social sciences students and is guided by three research questions:

4.1 How do first-year university students in social science seek feedback from inspectable student models in an introductory statistics course?

4.2 How does feedback from inspectable student models inform these students' decisions about subsequent actions?

4.3 How does these students' feedback-seeking and decision-making behavior relate to performance on a statistics exam?

After having discussed feedback design and students' use of the designed feedback for both inner and outer loop feedback, in **Chapter 5** we turn to an evaluation of feedback effects on students' statistical proficiency. In a randomized controlled trial with 521 participants (first-year social sciences students) and a factorial 2x2 design (inner loop feedback vs. no inner loop feedback and outer loop feedback vs. no outer loop feedback), the effects of both feedback types and their interaction on the students' learning processes and course performance were evaluated. The research question for this evaluation is:

5.1 What effects does providing both inner and outer loop feedback on online homework have on students' learning process and course performance in a university statistics course?

As this research question indicates, in Chapter 5 we do not only focus on feedback effects on students' statistical proficiency, but also on effects that offering both inner and outer loop feedback have on the students' learning processes. This allows us to verify and corroborate findings from the earlier cycles. The main findings of all four cycles combined are summarized and interpreted in **Chapter 6**. This final chapter also discusses the study's contributions, limitations, implications, and directions for future research.

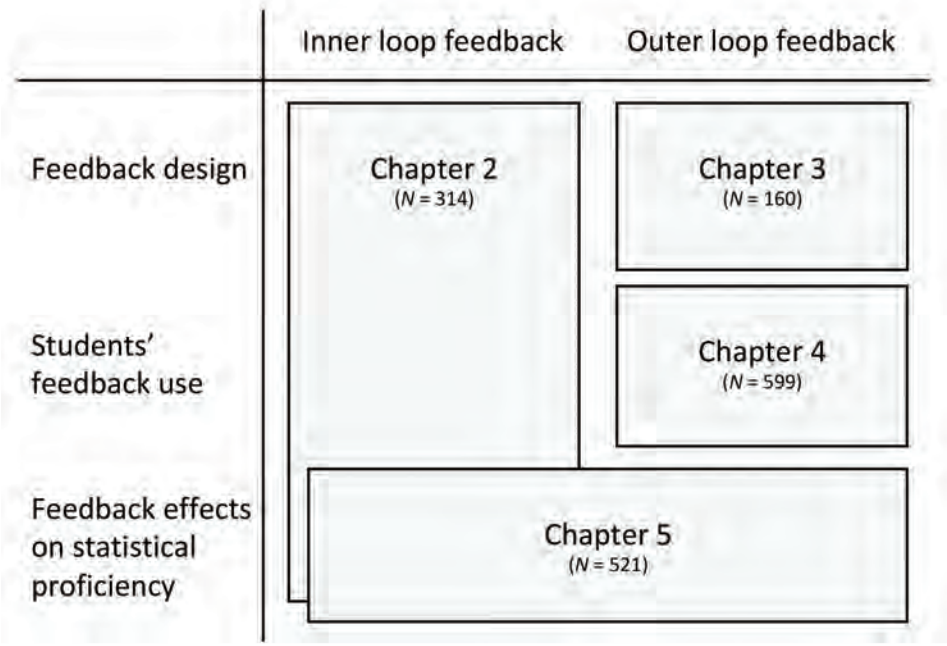

Figure 1.1 Alignment of chapters with feedback implementation aspects (design, use and effects) and feedback type (inner and outer loop)

CHAPTER 2 Intelligent feedback on hypothesis testing

Tacoma, S. G., Heeren, B. J., Jeuring, J. T., & Drijvers, P. H. M. (2020). Intelligent feedback on hypothesis testing. *International Journal of Artificial Intelligence in Education*. doi: 10.1007/s40593-020-00218-y

Earlier parts of this chapter are published as:

- Tacoma, S. G., Heeren, B. J., Jeuring, J. T., & Drijvers, P. H. M. (2019). Automated feedback on the structure of hypothesis tests. In S. Isotani, E. Millán, A. Ogan, P. Hastings, B. McLaren, & R. Luckin (Eds.), *Artificial Intelligence in Education*. AIED 2019. LNCS 11626. Cham, Switzerland: Springer.
- Tacoma, S. G., Heeren, B. J., Jeuring, J. T., & Drijvers, P. H. M. (2019). Automated feedback on the structure of hypothesis tests. In U. T. Jankvist, M. van den Heuvel-Panhuizen, & M. Veldhuis (Eds.), *Proceedings of the Eleventh Congress of the European Society for Research in Mathematics Education* (pp. 2969–2976). Utrecht, the Netherlands: Freudenthal Group & Freudenthal Institute, Utrecht University and ERME.

Author contributions: Sietske Tacoma: Conceptualization, formal analysis, software, writing - original draft. Bastiaan Heeren: Software, writing - review & editing. Johan Jeuring: Writing - review & editing, supervision. Paul Drijvers: Writing - review & editing, supervision.

Abstract Hypothesis testing involves a complex stepwise procedure dealing with statistical concepts and uncertainty and is, therefore, challenging for many students in introductory university statistics courses. In this paper we assess whether and how feedback from an Intelligent Tutoring System addressing the logic of this procedure can contribute to first-year social sciences students' proficiency in carrying out hypothesis tests. Students in an experimental group (N = 163) received intelligent feedback addressing the logic of the hypothesis-testing procedure, while students in a control group (N = 151) only received stepwise verification feedback. Immediate feedback effects were measured by comparing time on task and numbers of attempted tasks, complete solutions, and errors between the groups. Transfer of feedback effects was measured by student performance on follow-up tasks. Results showed that students receiving intelligent feedback spent more time on the tasks, solved more tasks and made fewer errors than students receiving only verification feedback. These positive results did not transfer to follow-up tasks, which might be a consequence of the isolated nature of these tasks. We conclude that intelligent feedback may stimulate students to devote more effort to hypothesis-testing tasks and may support them in learning to solve such tasks independently.

Keywords Domain reasoner ◆ Hypothesis testing ◆ Intelligent tutoring systems ◆ Statistics education

2.1 Introduction

Hypothesis testing is widely used in scientific research, and is therefore covered in most introductory statistics courses in higher education (Carver et al., 2016). This topic is challenging for many students, because it requires the ability to follow a complex line of reasoning involving uncertainty (Falk & Greenbaum, 1995; Garfield et al., 2008). Additionally, this line of reasoning involves several complex concepts, such as significance level, test value and *p*-value (Castro Sotos, Vanhoof, Van den Noortgate, & Onghena, 2007). Students struggle to understand the role and interdependence of these concepts in the hypothesis-testing procedure, or, in other words, the logic of hypothesis testing (Vallecillos, 1999). Appropriate feedback could support students in comprehending this logic, by focusing the student's attention to currently relevant aspects and thus reducing cognitive load (Shute, 2008). To address the logic of hypothesis testing, feedback should address all aspects of a solution: not only the content of a current step, but also its relations to earlier steps.

Since groups in introductory statistics courses are often large, it is difficult for teachers to provide such sophisticated feedback to individual students. Intelligent Tutoring Systems (ITSs) could offer a solution: like human tutors they can provide feedback on the level of steps, as well as detailed diagnostics of student errors (Nwana, 1990). Some ITSs have been found to be as effective as human tutors and, generally, ITSs that provide feedback on the level of steps have been found to be more effective than ITSs that provide feedback on the level of complete solutions (VanLehn, 2011). However, ITSs are highly domain dependent and while ITSs have been designed for the domain of hypothesis testing (Kodaganallur, Weitz, & Rosenthal, 2005), to our knowledge no critical evaluations of their effectiveness for learning have been reported up to date.

The contribution of this paper is a thorough evaluation of the impact of ITS feedback, which especially addresses the logic of hypothesis testing, on students' ability to carry out hypothesis tests. This evaluation is guided by the question: Does automated intelligent feedback about the logic of hypothesis testing contribute to student proficiency in carrying out hypothesis tests?

2.2 Stepwise feedback in Intelligent Tutoring Systems

Although ITSs vary considerably in design, they generally contain the following four components: an expert knowledge module, a student model module, a tutoring module, and a user interface module (Nwana, 1990). Of these four, the expert knowledge module is mainly responsible for diagnosing errors in student solutions and is, hence, highly domain dependent. It contains information about domain knowledge required to solve tasks in the domain (Heeren & Jeuring, 2014), and is therefore also referred to as domain reasoner (Goguadze, 2011). Two important paradigms for constructing domain reasoners are model-tracing (Anderson, Corbett, Koedinger, & Pelletier, 1995) and constraint-based modeling (Mitrovic, Martin, & Suraweera, 2007).

In the model-tracing approach, the ITS checks whether a student follows the rules of a model solution (Anderson et al., 1995). The domain reasoner contains a set of expert rules, which an expert would apply to solve tasks in the domain. It may also contain buggy rules: incorrect rules reflecting incorrect domain knowledge. Finally, the domain reasoner contains a model tracer that can identify which expert and buggy rules a student has applied to arrive at a (partial) solution. A student's step is marked as an error if it either does not match any expert rule, or matches a buggy rule (Mitrovic, Koedinger, & Martin, 2003). Furthermore, model-tracing domain reasoners can provide hints for appropriate next steps.

Constraint-based modeling concentrates on partial solutions, rather than on the solution process. The underlying idea is that incorrect knowledge emerges as inconsistencies in students' partial solutions (Mitrovic et al., 2007). Domain knowledge is represented as a set of constraints, consisting of a relevance condition and a satisfaction condition. Errors in student solutions emerge as violated constraints, that is, constraints for which the relevance condition is satisfied, but the satisfaction condition is not. If a student's partial solution does not violate any constraints, it is diagnosed as correct.

ITSs that support hypothesis testing have been designed based on either of these approaches (Kodaganallur et al., 2005). We do not believe that one or the other is a superior paradigm, but rather concur with Mitrovic

and colleagues (2003) that both have their strengths and weaknesses. We therefore combined the two paradigms within an ITS supporting hypothesis testing, similarly to what Goguadze and Melis (2009) did for arithmetic. To illustrate the merits of both paradigms for hypothesis testing, in the next section we discuss how the paradigms separately would diagnose typical student errors in carrying out hypothesis tests. This pedagogical discussion is followed by the description and evaluation of our ITS combining both paradigms.

2.3 Stepwise feedback on hypothesis testing

Feedback typically signals a gap between a student's current performance and desired performance, the feedback-standard gap (Kluger & DeNisi, 1996). In the case of hypothesis testing, a feedback-standard gap can manifest itself in several ways:

- an error within a single step, such as an erroneous value of the test statistic;
- missing information, such as a solution that contains a value for the test statistic, but no hypotheses to test;
- inconsistent information, such as a right-sided rejection region for a left-sided test.

The latter two are especially related to the logic of the hypothesis-testing procedure, since they concern the order of steps and the relations between steps. Model-tracing and constraint-based modeling typically approach these gaps in different ways, which we illustrate with two examples.

The first example concerns a student who starts the solution process with calculating a value of the test statistic, without stating hypotheses. Although technically possible, from a pedagogical perspective this step is not desirable, because the meaning and interpretation of a value of the test statistic depend on the hypotheses that are tested. A constraint-based tutor, on the one hand, typically contains constraints that check for necessary elements in the solution (Mitrovic et al., 2003). For hypothesis testing, such a constraint could have relevance condition "the solution contains a value of the test statistic" and satisfaction condition "the solution contains hypotheses". In this example, this constraint would be violated and a feedback message could encourage the student to first formulate

hypotheses before proceeding with carrying out the test. A model-tracing tutor, on the other hand, would contain a rule for adding hypotheses as well as a rule for calculating the value of the test statistic. In this example, adding hypotheses would be an expected step, whereas calculating the value of the test statistic would not. Depending on the implementation, the student's step of calculating the test statistic could be recognized as a detour from the expert strategy and this could be given as feedback to the student. However, why it is a detour (in this case, because hypotheses are missing) would be much more difficult for a model-tracing tutor to diagnose. Hence, providing explicit feedback about missing elements of a (partial) solution is generally more straightforward in the constraint-based paradigm.

The second example concerns inconsistent information in a solution. Suppose a student has almost finished the task: the hypotheses, critical value, rejection region and value of the test statistic comprise a logical line of reasoning. In the final step, however, the student draws an incorrect conclusion about the hypotheses. If the correct answer would be to reject the null hypothesis, then two conceptually different incorrect conclusions are possible: "Do not reject the null hypothesis" and "Accept the alternative hypothesis". The first reflects an inconsistency between the previous steps and the final conclusion, while the second concerns a misunderstanding of the convention in hypothesis testing to draw conclusions about the null hypothesis and not about the alternative hypothesis. In a constraint-based tutor, these two pieces of domain knowledge could be captured in two constraints. The first would have relevance condition "the test statistic lies inside the rejection region and a conclusion is drawn" and as satisfaction condition "the conclusion is to reject the null hypothesis." This constraint is violated by both errors described above. The second constraint, addressing the convention, would have as relevance condition "a conclusion is drawn" and as satisfaction condition "the conclusion concerns the null hypothesis" and is only violated by the second incorrect answer. Here, the prioritization of constraints is important to distinguish between such errors. The model-tracing approach for this situation is more straightforward: a model-tracing tutor can contain buggy rules for each of the two error types and provide appropriate feedback for each one of them (Mitrovic et al., 2003).

To summarize, both the constraint-based and the model-tracing paradigm have their merits for addressing the logic of hypothesis testing. A final typical feature of model-tracing tutors that is much less straightforward to achieve in constraint-based tutors is the provision of hints on next steps (Goguadze & Melis, 2009). More specifically, hints by model-tracing tutors can be expressed in terms of what a student needs for a logical next step in the current line of reasoning, while advice from constraint-based tutors typically focuses more on desired features of the solution (Mitrovic et al., 2003). Together, these two aspects can help students gain understanding of the steps that are essential for hypothesis testing and the order in which they are typically carried out. From a pedagogical perspective, therefore, combining both paradigms into a single ITS for hypothesis testing seems promising. In the following sections we turn to a design study evaluating this combination in practice.

2.4 Methods

2.4.1 Design of the domain reasoner

The technical design of the domain reasoner evaluated in this study is based on the Ideas framework (Heeren & Jeuring, 2014), which uses a model-tracing approach to calculate feedback and hints. For this study, this framework was expanded to also support constraints. The final domain reasoner contains 36 expert rules, 16 buggy rules, and 49 constraints.

Each time a student adds a step to a hypothesis-testing procedure, such as defining an alternative hypothesis or calculating the value of a test statistic, the domain reasoner checks the student's solution so far. Figure 2.1 illustrates the domain reasoner's checking procedure, which results in a diagnosis about the current partial solution. First, all constraints are checked. The constraints are assumed to be complete, which means that together they separate correct from incorrect (partial) solutions: a partial solution is correct if and only if it does not violate any constraint. If a solution violates one or more constraints, the domain reasoner determines whether a buggy rule was applied. If so, a feedback message specific to this buggy rule is displayed to the student, and otherwise a general message for the violated constraint is reported. For example, a partial solution that contains a rejection region but no alternative hypothesis

violates the constraint with relevance condition "the solution contains a rejection region" and satisfaction condition "the solution contains an alternative hypothesis." The corresponding feedback message addresses the role of the hypotheses: "To which hypotheses does this rejection region correspond? First state hypotheses."

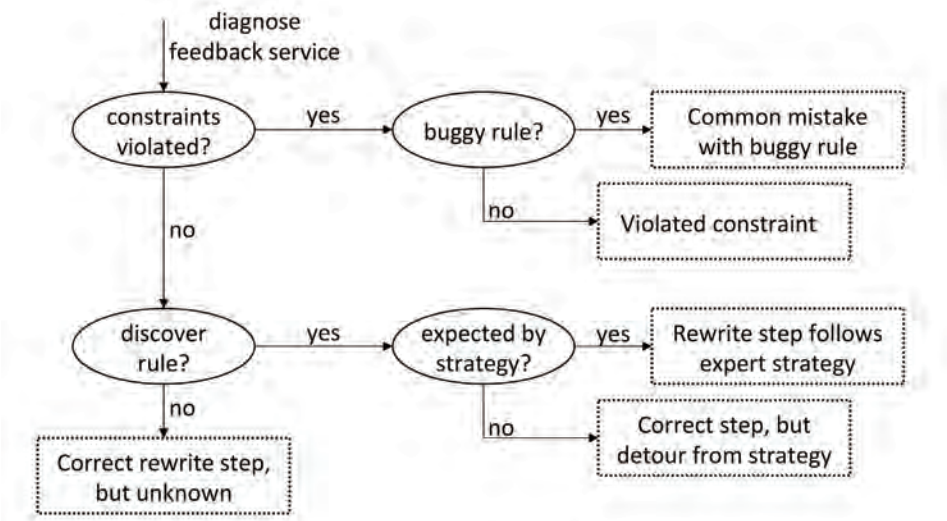

Figure 2.1 Domain reasoner's diagnose feedback service

If no constraints are violated, there is no need to check the buggy rules, because of the completeness of the constraints: if a buggy rule was applied, then at least one constraint would have been violated as well. Therefore, the domain reasoner only needs to attempt to discover which rule the student has applied to arrive at the current partial solution. If no rule is identified, the student's partial solution is marked as a correct, but unknown, step. This is an advantage of the constraints structure: students can add multiple steps at once and, as long as no constraints are violated, this is regarded correct. In a tutor based solely on model-tracing, to allow adding multiple steps at once all possible combinations of steps should be checked. If a rule is identified, the domain reasoner checks whether this is an expected rule in the expert strategy, so that detours from this strategy can be signaled. In the implementation in this study, though, no distinction was made between rules following the strategy and not following the strategy. In both cases, a feedback message for the identified rule is displayed, for example: "Your rejection region is correct". Besides

checking partial solutions, the domain reasoner can also provide hints on next steps to take, by identifying a rule that would be appropriate to apply for the current partial solution. This feature could also be used to generate a worked-out solution, which is a strength of the model-tracing approach. In this study, though, the possibility of worked-out solutions was not exploited.

The design of expert rules, buggy rules and constraints was informed by discussions with four teachers of introductory university statistics courses about the logic of hypothesis testing and common errors by students. Furthermore, textbooks were consulted. Based on this input, we decided to support two methods for logical reasoning in carrying out a hypothesis test: the conclusion about the hypotheses can be drawn based on comparison of the test statistic with a critical value, or based on comparison of a *p*-value with a significance level. In each method, a complete solution should include four essential steps: (1) state hypotheses, (2) calculate a test statistic, (3) either find a critical value or find a p-value, and (4) draw a conclusion about the hypotheses. Although crucial for the logic of hypothesis testing, stating a significance level and selecting an appropriate statistical test were not regarded as essential steps, because they were specified in all task descriptions. Besides these essential steps, students could include several other steps, such as a summary of sample statistics and a specification of whether the test was left-sided, right-sided or two-sided.

To identify and resolve technical flaws and unclarities in the design, a first version of the domain reasoner was piloted with five students. After the pilot, several improvements were made to feedback formulation and prioritization of rules and constraints.

2.4.2 Study design

The study consisted of a randomized controlled experiment that was embedded in a compulsory course on Methods and Statistics for first-year psychology students at a Dutch research university. In five weeks of this ten-week course students received online homework sets containing 7 to 13 tasks, which were designed in the Digital Mathematics Environment (DME, see Drijvers, Boon, Doorman, Bokhove, & Tacoma, 2013). The DME

supports various interaction types, such as formula input and multiple choice items, and was connected to the domain reasoner to enable intelligent feedback on hypothesis-testing tasks.

The third, fourth and fifth homework set concerned hypothesis testing. Each of these homework sets contained two tasks specifically aimed at developing the students' proficiency in carrying out hypothesis tests, by asking the students to select steps from a drop-down menu and to complete these steps. An example is shown in Figure 2.2: after selecting a step from the drop-down menu called "Action", it appears as next step in the step construction area. Next, the student can complete the step by filling in the answer boxes and use the check button to check the procedure so far. After finishing the hypothesis-testing procedure, the student should state the overall conclusion in the final conclusion area below the drop-down menu with steps.

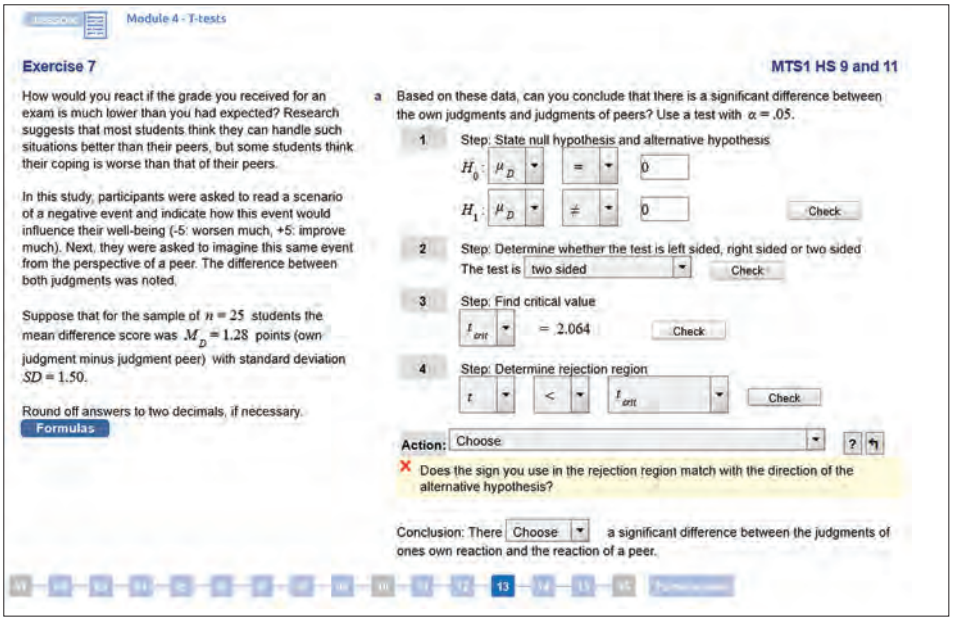

Figure 2.2 Hypothesis-testing task in the DME (translated)

Two versions of the homework sets were designed: an experimental version in which intelligent feedback on the steps in the hypothesis-testing procedure was provided by the domain reasoner, and a control version that only provided verification feedback on the individual answer boxes in the

steps. Hence, in the experimental condition students received elaborate feedback on fallacies in the logic of their hypothesis tests, while in the control condition students only received feedback on the correctness of their current step, irrespective of previous steps. Figure 2.3 shows an enlarged version of the feedback in the experimental condition that is shown in Figure 2.2 and Figure 2.4 shows the feedback for the same partial solution in the control condition. This example illustrates how the domain reasoner feedback addresses the student's error in relation with the statistical concepts involved, while in the control condition the error is only flagged, without further elaboration.

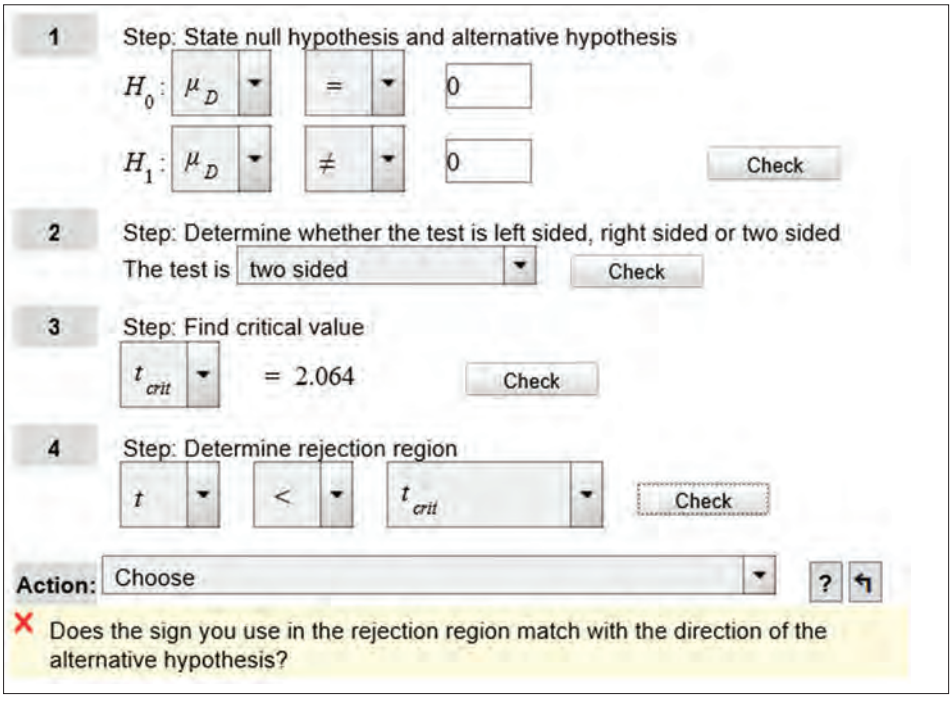

Figure 2.3 Example of feedback in the experimental condition

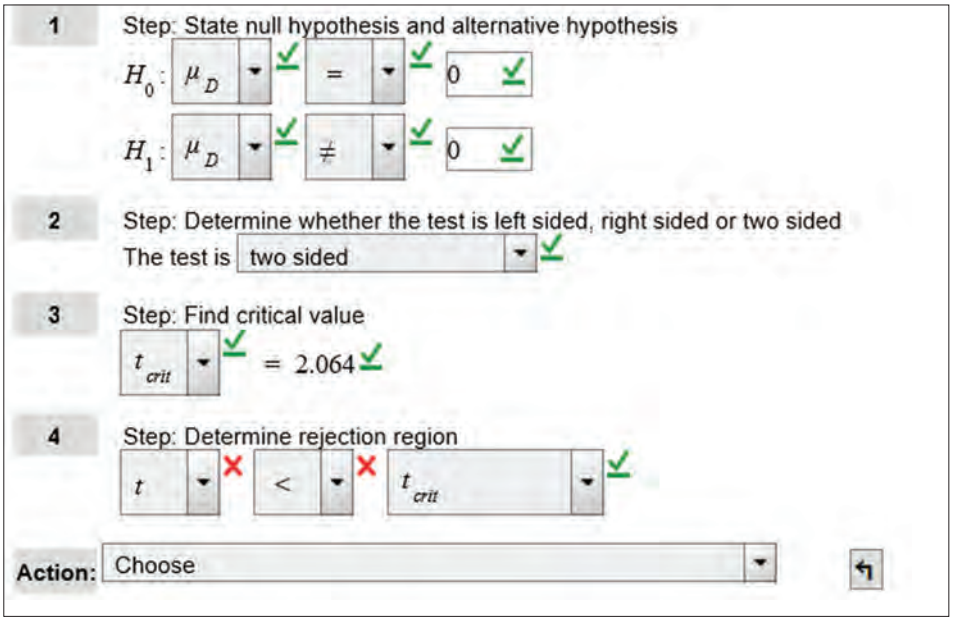

Figure 2.4 Example of feedback in the control condition, for the same partial solution as in Figure 2.3

Because of the differences in feedback students received, assessment criteria for correct solutions differed between conditions: in the experimental version, correct solutions needed to include all four essential steps, since otherwise one or more constraints would be violated. Since in the control condition the relations between steps were not checked, students only needed to include a correct conclusion about the null hypothesis for a solution to be correct. A final difference between the two versions was the presence of a hint button in the experimental version, which students could use to request a hint on which next step to take. All other tasks were equal in both versions.

2.4.3 Participants

Participants in this study, the first-year psychology students enrolled in the Methods and Statistics course, were divided randomly into an experimental and a control group. From the 310 students in the experimental group 226 students worked on the hypothesis-testing tasks, of which 163 gave consent for the use of their work in this study. From the 309 students in the control group 216 students worked on the tasks, of which 151 gave consent. The participants were between 17 and 31 years old (M = 19.3,

SD = 1.7) and 77% were female. To reduce research participation effects, i.e. students possibly behaving differently because they were part of an experiment (McCambridge, Witton, & Elbourne, 2014), the students, both in the experimental and the control group, were not given all information: they were told that they were part of an experiment and asked for their consent, but they were not told about the different conditions and which condition they were assigned to.

2.4.4 Data collection and analysis

Data for this study consisted of logs of the students' actions on the online homework sets. These logs included all attempts students made to construct correct answers to the tasks, and all feedback and hint requests. After exporting the logs from the DME, logs from students who did not give consent were deleted and all other logs were anonymized.

Data analysis focused on three aspects of the students' work:

A1. The amount of work students in the ITS feedback condition and the control condition did and the amount of feedback they received on the six hypothesis-testing tasks;
A2. Performance on the six hypothesis-testing tasks, as measured by (1) number of tasks attempted, (2) number of tasks solved and (3) number of errors concerning the logic of hypothesis testing;
A3. Performance on follow-up tasks about hypothesis testing without intelligent feedback.

The first aspect, A1, was deemed relevant, because students can only learn from feedback if they indeed receive it. And to receive feedback, students need to work on the tasks. The time students worked on the tasks and mean number of steps students selected were compared between groups. Since samples were large (more than 100 students in each group), independent samples *t*-tests were used for all comparisons between groups (Field, 2009). Welch two sample *t*-tests were used when variances were not equal in both groups, as tested by Levene's test. Furthermore, for students in the experimental group the number of feedback messages received and hints requested were calculated per task.

Regarding A2, three measures were used to assess student performance on the six tasks: (1) number of tasks in which students attempted to construct steps, (2) number of tasks that students solved completely, and (3) number of errors students made concerning the logic of hypothesis testing. The first measure (A2, measure 1) was regarded as indicator of feedback effectiveness, since the domain reasoner feedback was designed to support students in the step construction process. Students who did not attempt to construct steps in later tasks apparently did not perceive the feedback on steps in earlier tasks as helpful (Narciss et al., 2014). While the more elaborate feedback by the domain reasoner was expected to encourage students to attempt constructing steps, at the same time it required students to include steps in a correct order, which could lead to frustration and giving up on tasks.

Since the feedback was intended to contribute to the students' ability to solve the tasks, the number of solved tasks (A2, measure 2) is also an indicator of feedback effectiveness (Narciss et al., 2014). Students' solutions in the control group were assessed twice: according to their own group's criterion of stating a correct conclusion about the null hypothesis and according to the experimental group's criterion of including all four essential steps. Due to the intelligent feedback, students in the experimental group were expected to solve more tasks than students in the control group. Due to the difference in assessment criteria, however, students in the control group could be expected to solve more tasks under their own assessment criteria than students in the experimental group. The comparison between groups with a *t*-test was complemented with a logistic multilevel regression model (Hox, Moerbeek, & Van der Schoot, 2018) to assess the progression of the difference between groups over time. The regression model was built in the software program HLM using full maximum likelihood estimation, as described in Hox et al. (ibid.).

The final measure of student performance on the six tasks was the number of errors that students made in the logical reasoning of their hypothesis tests (A2, measure 3). The domain reasoner was especially designed to provide students with feedback about the logic of hypothesis testing, that is, the order of and relations between steps. The number of errors concerning this logic was expected to decrease over time in both

groups, but more strongly in the experimental than in the control group. To assess the evolution of the difference between groups over time, we employed a *t*-test and a multilevel regression model (Hox et al., 2018).

Concerning A3, we notice that promising effects of feedback on student performance on the tasks for which feedback is provided do not automatically guarantee transfer to new tasks (Shute, 2008). We therefore also assessed student performance on follow-up tasks about hypothesis testing, in which no intelligent feedback was provided. From the online homework sets 31 follow-up sub-tasks on hypothesis testing were selected. For all students who received feedback on constructed steps at least once the ratio between the number of these 31 sub-tasks that they answered correctly on their first attempt and the number of sub-tasks they attempted was calculated and these ratios were compared between groups.

2.5 Results

2.5.1 Results on A1: summary of steps done and feedback received

Table 2.1 summarizes the average number of steps that students in both groups made and the number of feedback messages and hints students in the experimental group received. Students in the experimental group made slightly but significantly more steps (M = 8.0, SD = 5.4) than students in the control group (M = 6.7, SD = 3.9, $t(293.7) = 2.41$, $p = .016$, Cohen's $d = 0.27$). This is also reflected in the total time students worked on the six hypothesis-testing tasks: in the experimental group, this was 41 minutes (SD = 27 minutes) and in the control group, it was 32 minutes (SD = 19 minutes), a significant difference ($t(291.8) = 3.41$, $p < .001$, Cohen's $d = 0.38$). In both groups, the number of steps decreased over tasks. It should be noted that in the final two tasks the test statistic was given, so fewer steps were needed for a complete solution than in earlier tasks. Finally, the number of feedback messages per student in the experimental group is quite high, especially in the first two tasks, implying that students received feedback on a regular basis. Students also regularly made use of the hints, with an average of two hint requests per student per task.

Table 2.1 Steps in both groups and feedback messages and hints in experimental group

	Experimental group				Control group	
Task	*N*	Steps per student (*SD*)	Feedback messages per student (*SD*)	Hints per student (*SD*)	*N*	Steps per student (*SD*)
3.4	154	14.0 (10.3)	23.2 (21.2)	3.7 (7.5)	143	11.3 (7.9)
3.6	111	11.2 (6.7)	22.3 (23.3)	2.4 (4.6)	105	9.1 (6.9)
4.7	134	6.8 (6.7)	11.4 (13.4)	1.3 (4.2)	130	6.5 (5.8)
4.8	118	7.1 (6.2)	16.2 (23.1)	2.5 (5.0)	115	6.1 (5.5)
5.3	134	4.9 (5.6)	7.7 (14.2)	1.5 (4.0)	127	4.1 (5.0)
5.6	127	3.9 (4.8)	5.6 (7.7)	1.4 (3.7)	123	3.3 (4.0)
All	163	8.0 (5.4)	14.1 (12.7)	2.0 (3.5)	151	6.7 (3.9)

2.5.2 Results on A2: performance on six hypothesis-testing tasks

The average number of tasks students worked on, i.e., tasks in which they filled in the final answer box, and the average number of tasks in which students tried to construct steps (A2, measure 1) are summarized in Table 2.2. In both groups, students attempted to construct steps using the drop-down menu for almost 80% of the tasks they worked on. For the other 20% of the tasks, students may have used other means than the stepwise construction area to solve the task or may have collaborated with a peer. The numbers of tasks students worked on and attempted step construction for did not differ significantly between groups.

Table 2.2 Student results on the six hypothesis-testing tasks

	Experimental group (N = 163)	Control group (N = 151)	t (df = 312)	p
Tasks worked on	4.8 (1.5)	4.9 (1.5)	0.86	.391
Tasks tried constructing steps	3.8 (1.7)	3.9 (1.6)	0.62	.537
Tasks with complete solution	1.7 (1.8)	2.0 (1.7)	1.33	.184
Tasks with correct essential steps	1.7 (1.8)	1.4 (1.6)	-1.59	.113

In Table 2.2, the third and fourth line summarize the average number of tasks that students solved completely (A2, measure 2). Students succeeded in solving the task in approximately half of the cases in which they attempted to construct steps. Over all six tasks, students in the control group solved slightly more tasks than students in the experimental group. This could be a consequence of the stricter assessment criterion for complete solutions

in the experimental group, that required students to include all essential steps in their solution. When assessed following this stricter criterion, the number of complete solutions in the control group dropped to an average of 1.4 per student. Over all six tasks together, these differences between groups were not significant, as the results in Table 2.2 show. Given that students started off with the same prior knowledge, however, differences between groups were expected to emerge over time. A logistic regression model was created to take this effect of time into account. The model is summarized in Table 2.3.

Table 2.3 Logistic multilevel regression model predicting the probability of solving a task from task number, domain reasoner availability and their interaction

	M1: Baseline	M2: + condition	M3: + interaction condition/task
Predictor coefficients			
Intercept	0.23	0.46*	0.79***
Task number	-0.24***	-0.24***	-0.41***
Domain reasoner		-0.43	-1.07***
Domain reasoner × Task number			0.32***
Model fit			
Deviance	3762.80	3759.10	3745.18
Estimated parameters	3	4	5
Deviance change		3.70	13.92***
Explanatory power			
Proportion solved tasks predicted correctly	.51	.60	.60
φ correlation coefficient	.16	.17	.17

*p < .05; **p < .01; ***p < .001.

The baseline model in Table 2.3 only included task number as predictor for solving the task. It reveals that the probability of solving a task decreased with task number, meaning that, generally, for higher task numbers the proportion of students who solved the task decreased. Including domain reasoner availability (M2) did not significantly improve the model: the deviance change was 3.70, which, with one degree of freedom for one extra estimated parameter, results in a p-value of .054. This aligns with our

previous finding that over all tasks together domain reasoner availability did not make a difference for the number of tasks students solved. The explanatory power of M2 was slightly higher than that of M1, though. Especially, while M1 only predicted 51% of the solved tasks correctly, M2 predicted 60% correctly. The addition of an interaction effect between task number and domain reasoner availability (M3) improved the model further: the deviance change was 13.92, which, with one degree of freedom for one extra estimated parameter, results in $p < .001$, hence a significant improvement to the model. The regression equation for this final model is:

$$\text{logit}(p_{ij}) = 0.79 - 0.41 \cdot (i - 1) - 1.07 \cdot \textit{domain reasoner}_j + 0.32 \cdot (i - 1) \cdot \textit{domain reasoner}_j + u_{0j},$$

with:

p_{ij} the estimated probability that student j solved task i correctly

$\textit{domain reasoner}_j$ equal to 0 (control group) or 1 (experimental group),

i representing the task number (between 1 and 6), and

u_{0j} a residual variance term for student j.

As in the baseline model M1, the negative regression coefficient for task number in the final model indicates that the probability of solving tasks decreased for later tasks. Filling in $i = 1$ and taking the inverse logit shows that the estimated probability of solving the first task was on average $\text{logit}^{-1}(0.79) = 0.69$ in the control group and $\text{logit}^{-1}(0.79 - 1.07) = 0.43$ in the experimental group, showing that initially students in the experimental group had more difficulty solving the tasks than students in the control group. This could be a consequence of the stricter assessment criteria in the experimental group, which students needed to get used to. Finally, the coefficient for the interaction term between domain reasoner availability and task number is positive. Hence, while for students in the control group the logit decreased by 0.41 per task, for students in the experimental group it only decreased by $0.41 - 0.32 = 0.09$ per task. This suggests that the domain reasoner feedback more effectively supported students in persevering to solve tasks than the control feedback, even though the assessment criteria for their solutions were stricter.

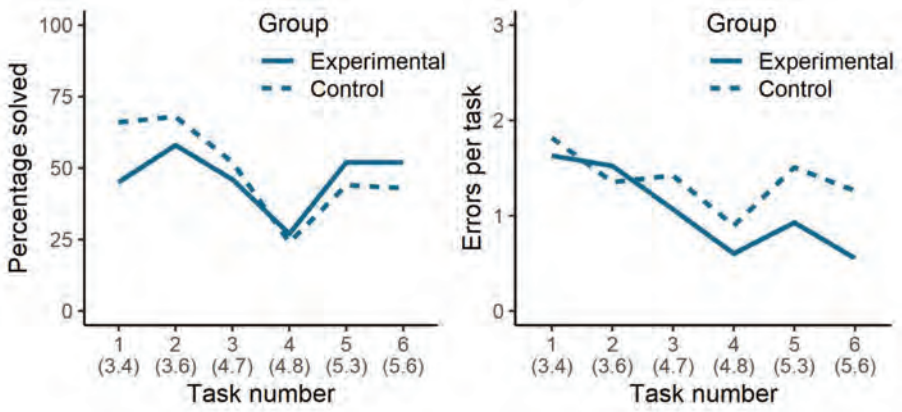

Figure 2.5 Percentage of students who correctly solved tasks according to group's assessment criteria (left) and mean number of errors concerning the logic of hypothesis testing (right)

This is also reflected in Figure 2.5 (left), which displays the percentage of students who found complete solutions to each task, as percentage of students who attempted constructing steps for each task. For the first three tasks the percentage was smaller for students in the experimental group than for students in the control group, but for the latter three tasks this was reversed. Hence, over time, students in the experimental group seemed to become relatively more proficient in solving hypothesis-testing tasks than students in the control group.

The final measure of student performance on the six tasks was the number of errors that students made in the logical reasoning of their hypothesis tests (A2, measure 3). The domain reasoner could diagnose 15 different errors concerning hypothesis-testing logic, such as a missing alternative hypothesis. On average, students in the experimental group made 1.12 (SD = 0.79) different errors per solution, while students in the control group made 1.42 (SD = 0.86) different errors, which was significantly more ($t(312) = 3.22$, $p = .001$, Cohen's $d = 0.36$). The graph in Figure 2.5 (right) displays the mean number of errors by students in both groups for each task. It shows that in both groups the number of errors decreased over tasks, but this trend was stronger in the experimental group. Fitting a multilevel regression model confirmed this impression. The resulting model is summarized in Table 2.4.

Table 2.4 Multilevel regression model predicting number of errors concerning hypothesis-testing logic from task number and task number squared, domain reasoner availability and interaction between task number and domain reasoner availability

	M1: Baseline	M2: + condition	M3: + interaction condition/task	M4: - condition
Fixed part				
Intercept	2.18***	2.33***	2.12***	2.17***
Task	-0.49***	-0.49***	-0.42***	-0.43***
Task quadratic	0.04***	0.05***	0.05***	0.05***
Domain reasoner		-0.29***	-0.10	
Domain reasoner × task			-0.13***	-0.11***
Random part				
σ_e^2	1.208	1.207	1.189	1.190
σ_{u0}^2	0.238	0.219	0.227	0.227
Model fit				
Deviance	3827.82	3816.12	3804.75	3805.22
Estimated parameters	4	5	6	5
Deviance change		11.70***	11.37***	-0.47

*p < .05; **p < .01; ***p < .001.

The baseline model (M1) included a linear and quadratic term for task number as predictors and showed that, generally, the number of errors decreased over time. The significance of the quadratic term suggests that the number of errors decreased quickly for the first tasks and more slowly for later tasks. In M2, domain reasoner availability was added to the baseline model, which resulted in a significantly better model fit (p < .001). The coefficient for domain reasoner availability was negative and significantly different from 0, confirming that the number of errors concerning hypothesis-testing logic was lower in the experimental group than in the control group. The variance at the student level decreased by 0.019, or 8.0% of the initial variance of 0.238. Hence, experimental condition explained 8% of the variance in number of errors per student. Adding the interaction effect between task number and domain reasoner availability (M3) again yielded a significantly better model fit (p < .001). In this model, the effect of domain reasoner availability itself became non-

significant. This implies that for the first task, domain reasoner availability did not have a significant effect on the number of errors students made. Meanwhile, the significant interaction effect between domain reasoner availability and time implies that, over time, students in the experimental group made significantly fewer errors concerning the logic of their hypothesis tests than students in the control group. Removing the non-significant predictor domain reasoner availability (M4) yielded an equally good model – the deviance change is very small and not significant (p = .493) – with fewer estimated parameters. Comparing this model to the baseline model shows that the interaction between domain reasoner availability and task number explained 1.5% of the variance at task level and 4.6% of the variance at student level. In other words, the domain reasoner feedback resulted in a slightly stronger decrease in number of errors for students in the experimental group than for students in the control group.

2.5.3 Results on A3: transfer of feedback effects to follow-up tasks

Students in the experimental group (N = 158) and the control group (N = 147) performed similarly on the selection of follow-up hypothesis-testing tasks: the mean ratio of correct answers was 0.72 (SD = 0.07) in the experimental group and 0.71 (SD = 0.08) in the control group. The time students worked on these tasks was also very similar in the experimental and control group: 49 minutes (SD = 17 minutes) for both groups. Hence, the effects of the domain reasoner feedback did not transfer to the follow-up tasks that students were offered in the course. For comparison, though, we note that the mean ratio of immediately correct answers over all tasks that were identical in both groups (i.e., all tasks except for the six tasks concerning stepwise hypothesis testing) was 0.67 (SD = 0.05). Hence, compared to other tasks the students in both groups performed relatively well on the follow-up tasks on hypothesis testing.

2.6 Conclusion and discussion

In this paper we have evaluated the influence of ITS feedback addressing the logic of hypothesis testing, guided by the research question: Does automated intelligent feedback about the logic of hypothesis testing contribute to student proficiency in carrying out hypothesis tests? Students in an experimental and a control group worked on six hypothesis-testing

tasks, in which they received a substantial amount of feedback and hints. While the ITS feedback did not seem to influence the number of tasks students attempted to construct steps in, it did affect their success in solving tasks. The first three tasks were solved by relatively fewer students in the experimental than in the control group, while for the later three tasks students in the experimental group persevered and succeeded more in solving the tasks. This suggests that after a period of familiarization with the ITS feedback students started to benefit from it. Furthermore, the number of errors students made in the logical reasoning of the hypothesis-testing procedure decreased significantly stronger over time for students receiving ITS feedback than for students receiving verification feedback only. Hence, the ITS feedback seemed to effectively support students in resolving their misunderstandings and, in this way, to contribute to student proficiency in carrying out hypothesis tests. Despite these promising results, no differences between groups were found in performance on follow-up tasks, which implies that there was no automatic transfer from the positive ITS feedback effects.

Although such a lack of transfer is often found (Shute, 2008), in the case of this study it could be due to the design of the follow-up tasks. This was a limitation of the study: contrary to the six hypothesis-testing tasks, none of the follow-up tasks specifically addressed the logical reasoning in the hypothesis-testing procedure. Instead, the steps of the hypothesis-testing procedure were already given and students were only asked to fill in contents of individual steps. From a research perspective, availability of tasks addressing the logical reasoning could have provided more insight into transfer of the positive ITS feedback effects to other tasks. From an educational perspective, availability of such tasks would have been valuable as well, to avoid that students rely too much on the ITS feedback (Shute, 2008).

A second limitation of this study was that, despite serious testing and pilots, in this first large-scale implementation of the domain reasoner inevitably some unclarities and technical flaws became apparent. A small number of feedback messages provided incorrect or unsuitable information about current errors, and hints could only suggest a next step to take, regardless of whether the student's current partial solution was correct.

For incorrect solutions, a hint containing guidance on how to resolve the current error would have been more appropriate. Nonetheless, the large collection of student data did provide a strong basis to inform improvements to the domain reasoner, and especially for designing a hint structure that suits the combination of the model-tracing and constraint-based modeling approach. Furthermore, even though sometimes encountering confusing feedback messages and hints, students in general kept attempting to construct steps and, as the results above show, did still benefit from the feedback.

Despite these limitations, combining the model-tracing and constraint-based paradigm seems to have resulted in a useful ITS for hypothesis testing. The constraint-based characteristics of the ITS enabled identifying missing elements and inconsistencies in students' solutions, and thus addressing fallacies in the logic of the students' hypothesis tests. Simultaneously, model-tracing elements allowed to address common errors, for example related to the convention to draw conclusions about the null hypothesis, and to provide hints. Combined, these characteristics have not only supported students in solving more of the later tasks and making fewer errors in these tasks, but also to work significantly longer on the tasks and make significantly more steps. As Narciss and colleagues (2014) argue, doing more work may result in more opportunities to practice, meaning that the ITS feedback may stimulate students to engage more deeply with the concepts and logical reasoning involved in hypothesis testing. Finally, the finding that students in the experimental group made fewer errors in later tasks than students in the control group indicates that students became less and less dependent on the feedback for solving the tasks. This effect is in line with earlier findings for ITS feedback effectiveness (Steenbergen-Hu & Cooper, 2014; Van der Kleij, Feskens, & Eggen, 2015) and the effect size found in this study, Cohen's $d = 0.36$, is similar to those reported in Steenbergen-Hu and Cooper's review on the effectiveness of ITS feedback in higher education (Steenbergen-Hu & Cooper, 2014).

Overall, this study suggests that combining the model-tracing and constraint-based modeling paradigms is not only promising in theory, but also in educational practice. An additional aspect of this approach that is worth mentioning is that, albeit after a considerable initial design effort,

it allows for easy adjustment of tasks to create new tasks. Once a start situation for a task is given, the model-tracing components of the domain reasoner can generate the solution and all steps towards the solution. This means that, contrary to the design in the control condition, the designer does not need to provide answers for all intermediate steps. Hence, even if the results do not transfer to follow-up tasks, with the ITS feedback available less design effort is needed for similar learning results. This invites the design of more tasks, offering students who need it more practice. In future designs, the domain reasoner's potential for generating worked-out solutions, as well as the possibility to distinguish between expected steps and steps that deviate from the expected strategy, could be exploited further. Finally, a challenging aspect of hypothesis testing that is not yet addressed by the ITS feedback in this study is the role of uncertainty in the interpretation of the results from hypothesis tests (Falk & Greenbaum, 1995). Future research could focus on broadening the scope of the domain reasoner for hypothesis testing to include this reasoning with uncertainty.

Acknowledgements

We thank teachers Jeltje Wassenberg-Severijnen and Corine Geurts for their collaboration in designing teaching tasks and delivering the course. Furthermore, we thank Noeri Huisman, Martijn Fleuren, Peter Boon and Wim van Velthoven who assisted in developing the domain reasoner.

CHAPTER 3 The interplay between inspectable student models and didactics of statistics

Tacoma, S. G., Sosnovsky, S. A., Boon, P. B. J., Jeuring, J. T., & Drijvers, P. H. M. (2018). The interplay between inspectable student models and didactics of statistics. *Digital Experiences in Mathematics Education, 4*(2-3), 139–162. doi: 10.1007/s40751-018-0040-9.

Author contributions: Sietske Tacoma: Conceptualization, Visualization, Formal analysis, Software, Writing - original draft. Sergey Sosnovsky: Methodology, Writing - review & editing. Peter Boon: Conceptualization, Software, Writing - review & editing. Johan Jeuring: Writing - review & editing, Supervision. Paul Drijvers: Conceptualization, Writing - review & editing, Supervision.

Abstract Statistics is a challenging subject for many university students. In addition to dedicated methods of didactics of statistics, adaptive educational technologies can also offer a promising approach to target this challenge. Inspectable student models provide students with information about their mastery of the domain, thus triggering reflection and supporting the planning of subsequent study steps. In this article, we investigate the question of whether insights from didactics of statistics can be combined with inspectable student models and examine if the two can reinforce each other. Five inspectable student models were implemented within five didactically grounded online statistics modules, which were offered to 160 Social Sciences students as a part of their first-year university statistics course. The student models were evaluated using several methods. Learning curve analysis and predictive validity analysis examined the quality of the student models from the technical point of view, while a questionnaire and a task analysis provided a didactical perspective. The results suggest that students appreciated the overall design, but the learning curve analysis revealed several weaknesses in the implemented domain structure. The task analysis revealed four underlying problems that help to explain these weaknesses. Addressing these problems improved both the predictive validity of the adjusted student models and the quality of the instructional modules themselves. These results provide insight into how inspectable student models and didactics of statistics can augment each other in the design of rich instructional modules for statistics.

Keywords Inspectable student model ◆ Open student model ◆ Statistics education ◆ Higher education ◆ Learning curve analysis

3.1 Introduction

Statistical methods are highly relevant for conducting research in many fields of science. Therefore, many university programs include introductory statistics courses (Castro Sotos, Vanhoof, Van den Noortgate, & Onghena, 2007), which are often challenging for students (Murtonen & Lehtinen, 2003; Tishkovskaya & Lancaster, 2012). This is partly due to the complexity of the domain itself (Castro Sotos et al., 2007), and partly to the large size of the groups of students to whom these courses are taught, which greatly reduces teachers' ability to provide individual guidance to students.

From the field of statistics education research, various suggestions for enhancing statistics education have emerged in the past decades. A major change that has taken place concerns the main goals of statistics courses. Whereas traditionally the primary focus was on deriving statistical formulas and carrying out calculations, nowadays much more attention is paid to the interpretation of data and the ability to reason statistically about real-world problems, also referred to as "statistical literacy" (Lovett & Greenhouse, 2000). This shift in goals is partly evoked by the large-scale availability of statistical software that can take care of calculations. Accomplishing this shift asks for specific didactical considerations in instructional design, such as using real contexts and data for promoting meaningful statistical reasoning (Ben-Zvi, 2000).

Another possible enhancement of statistics education, which is especially relevant when individual guidance by teachers is difficult to achieve, comes from a different area: adaptive educational technologies (Herder, Sosnovsky, & Dimitrova, 2017). These technologies help convert results of automated assessment into detailed information for students and teachers, including diagnostic feedback (Stacey & Wiliam, 2013). In the case of statistics education, with its challenging number of concepts to master, it seems particularly promising to provide students with information on their mastery of these individual concepts. One popular adaptive educational technology for providing such information is the inspectable student model (Bull & Kay, 2007).

A student model is a structured collection of information about the individual student's characteristics, such as knowledge, difficulties and

misconceptions, in the domain of study. Adaptive educational systems elicit this information based on students' interaction with learning content: solving tasks, taking tests, studying examples, etc. Presenting this information to students as feedback and allowing them to inspect it freely is known to promote reflection, increase motivation and provide metacognitive support for self-regulated learning (Bull & Kay, 2007). In other words, an inspectable student model can support a student in forming an opinion about his or her current progress and making a well-considered decision about the next learning step (which concepts to focus on, which task to attempt, etc.).

However, the effectiveness of such an enhancement of the learning process in many respects depends on whether inspectable student models can be combined with the employed didactical approach. In the context of this article, the question is: how can the fields of didactics of statistics and inspectable student models be integrated? And can they strengthen each other?

To address these questions, inspectable student models were implemented in five modules containing practice exercises on introductory statistics. These modules were embedded in an online educational system and were offered to 160 students in the Social Sciences as a part of their first-year, introductory statistics course. The inspectable student models were evaluated from two standpoints: the perception of the students who worked with them and their internal quality. Students' perceptions were collected through a questionnaire and served to evaluate whether combining the fields of didactics of statistics and inspectable student models was appreciated by students. For the quality analysis, evaluation methods from both fields were used. This quality analysis served two goals: to evaluate whether the implemented student models were successful and to explore how this implementation could be improved. Four main problems in the implementation were identified, for which solutions were sought both in the student model design and in the instructional design of the statistics modules.

3.2 Theoretical background

Before attempting to combine the two fields of didactics of statistics and inspectable student models, we would like to explore both fields separately. In this exploration, we explicate the difficulties that students experience in statistics education and examine how these difficulties are addressed both by didactical methods (i.e. methods informed by domain-specific pedagogical considerations) and by the information provided to students through inspectable student models. Moreover, we look for differences between the two fields that might lead to challenges in integrating them.

3.2.1 Didactics of statistics

Research in statistics education has identified several causes for the challenging character of statistics. First of all, the field of statistics involves a large number of abstract concepts, such as probability distributions, sampling variability and confidence intervals. Second, constructing sound statistical conclusions requires the ability to integrate such abstract concepts both into calculations and into complex chains of reasoning (Castro Sotos et al., 2007). For example, understanding the method of hypothesis testing requires knowledge of probability distributions, sampling variability and significance level, as well as the ability to reason using conditional statements (e.g., "under the assumption that the null hypothesis is true, this outcome, or a more extreme one, is very unlikely"). Finally, abstract definitions of statistical concepts such as variability often conflict with students' prior, informal knowledge and their view of the real world (Garfield & Ahlgren, 1988).

To support students in overcoming these challenges – that is, in gaining understanding of these abstract concepts, calculations and chains of reasoning – various strategies are prevalent in statistics education. Recommendations by Ben-Zvi (2000) and the GAISE college report (Garfield et al., 2005) include the use of real data sets and a focus on conceptual understanding and statistical reasoning, rather than mere knowledge of procedures. Real data sets can engage students in thinking about the data and relevant statistical concepts. The recommendation to focus on conceptual understanding and statistical reasoning rather than on procedures is based on the assumption that students with a good conceptual foundation will easily grasp new procedures and techniques,

whereas procedural knowledge without conceptual understanding tends to be too superficial and not well integrated (Garfield et al., 2005).

These insights may guide instructional design. Taking real data sets from real contexts as a starting point for instructional design results in clusters of tasks that are related to each other through these contexts. A single context may, for example, be used for comparing different representations of the data, calculating and interpreting confidence intervals and carrying out hypothesis tests. The sequencing of such closely related tasks is crucial (Drijvers, Boon, Doorman, Bokhove, & Tacoma, 2013). Deliberate task sequencing can serve to introduce concepts gradually, first informally and only later in a more formal way (e.g. Aberson, Berger, Healy, & Romero, 2003), or to evoke crises to promote deeper reflection (Bokhove & Drijvers, 2012). When exploring a context, earlier tasks are typically aimed at becoming acquainted with the context and data, whereas in later tasks the by now familiar context can serve as a concrete example, and hence support understanding of abstract concepts and their interrelationships. In other words, well-considered clustering and sequencing of tasks in the instructional design is essential both for engaging with real contexts and addressing conceptual understanding and statistical reasoning.

3.2.2 Inspectable student models

Student models are the core components of adaptive intelligent educational systems. They infer, store and update a system's estimations of the current knowledge state of each individual student, thus providing a basis for adaptively optimized support that the system can offer. A frequently used student model organization is an overlay model, which computes individual student mastery scores for a set of *knowledge components*: important concepts, methods, or other coherent pieces of domain semantics (Carr & Goldstein, 1977). In combination, these knowledge components constitute a model of the domain under study. A (partial) example of such a domain model is shown in the left-hand column of the inspectable student model displayed in Figure 3.1. In this example, the knowledge components are grouped into five categories and for two of these categories individual knowledge components are shown.

Category	Score
⊕ Types of random variables	66%
⊕ Visual data representations	74%
⊖ Cumulative frequencies and percentiles	60%
Cumulative frequency	54%
Percentile rank	100%
Percentile	33%
⊕ Measures of central tendency	75%
⊖ Measures of statistical dispersion	90%
Measuring variability	100%
Standard deviation sample	69%
Standard deviation population	100%
Range	100%

Figure 3.1 An inspectable student model on descriptive statistics

The knowledge components of different domain models can differ in several aspects, thus allowing for tailoring the domain model design to specific characteristics of the domain and the educational setting at hand. First of all, knowledge components can represent elements of procedural knowledge ("how" knowledge) that define procedures or skills in the domain or they can represent declarative knowledge ("what" knowledge) that define important concepts and facts (Brusilovsky & Millán, 2007). For statistics education, in which a focus on conceptual understanding is advocated, the latter type seems more appropriate. A second layer of diversity comes from the degree of granularity. A designer of a model might decide to break the knowledge in the domain into as small elements as possible, thus improving the potential precision of the model. She might also decide to define knowledge components at the level of larger categories and topics, thus facilitating easier content modeling – connecting learning tasks to knowledge components.

The student's mastery of the knowledge components (KCs) in the domain model is represented in an overlay: a set of scores that is usually based on the student's performance on learning tasks associated with corresponding KCs. The connection between tasks and KCs can be represented by a so-called Q-matrix (Barnes, 2005; Tatsuoka, 1983), with

a row for each KC and a column for each task. The entry (*i*, *j*) is equal to 1 if the *j*th task is connected to the *i*th KC, i.e. if the *i*th KC is relevant for solving the *j*th task, and 0 otherwise. The scores in the overlay may be either qualitative (poor, medium, good), simple numeric (a percentage, for example) or uncertainty-based (Brusilovsky & Millán, 2007). An example of an overlay is displayed in the right-hand column of the student model in Figure 3.1.

The main purpose of student models in adaptive educational systems is usually to provide a basis for adaptation. However, the information that the student model contains can also be used as valuable feedback for the student: if shown to the student, a student model can promote reflection and support planning and navigation (Bull & Kay, 2007). Reflection may for instance be promoted by a low score on a concept that a student thought she already mastered, and as such the open student model may reveal gaps in the student's knowledge of the domain. For these purposes, a fairly simple student model design may suffice. Although sophisticated methods exist for enabling students to edit or negotiate with their student model (e.g. Dimitrova, Self, & Brna, 2001; Zapata-Rivera & Greer, 2002), for the purpose of reflection, planning and navigation, promising results have been obtained with much simpler inspectable student models (Arroyo et al., 2007; Long & Aleven, 2011; Mitrovic & Martin, 2002) that do not allow a student to adjust the contents of the model (Bull, 2004). Moreover, Bull and Kay (2007) argue that in student models with the purpose of promoting reflection the scores can be presented in a simple way, without mentioning the uncertainties surrounding them. Reflection is most likely evoked by differences between the model and the student's own view, which are presumably larger if uncertainty is omitted.

A final remark on student models concerns their relation to the instructional design. Student models are often used to inform adaptation. In such cases, the instructional design includes variation of the order of tasks, depending on student achievement so far. To this end, the tasks need to stand alone rather than to be organized in a pre-structured sequence. Even in many cases where student models are made inspectable, they have initially been designed to inform adaptation, and are therefore connected

to a set of independent tasks, rather than to a sequence of closely related tasks that share contexts or build on one another.

To summarize, the main difficulties for students in statistics education are the large number of abstract concepts involved and the ability needed to integrate these concepts into calculations and chains of reasoning. Methods from the field of didactics of statistics to address these difficulties are the use of real data sets and contexts and a focus on conceptual understanding. Inspectable student models provide an additional method by supporting students in gaining insight into the structure of the domain of statistics and revealing knowledge gaps. An important difference between the methods from the two fields lies in the instructional design: the didactical methods result in sequences of closely related tasks that share contexts and build on one another, whereas inspectable student models traditionally are connected to sets of rather independent tasks. Therefore, the first question from the introduction, whether the fields of didactics of statistics and inspectable student models can be combined, can be explicated as follows: (RQ1) Are inspectable student models suitable for implementation in didactically grounded, sequential statistics modules consisting of closely related tasks? The second question, whether the two fields can strengthen each other, focuses on the evaluation methods available in both fields: (RQ2) How can didactical analysis inform design of inspectable student models, and, vice versa, how can student model evaluation methods inform didactical design?

3.3 Methods

To address these research questions, inspectable student models were designed and implemented in five didactically grounded modules which were used in an introductory statistics course at Utrecht University. In the following sections, we first describe the educational setting for this study, including a description of the online educational system that was used. Next, we discuss the didactical design of the modules and student model design. Lastly, we outline data collection and describe the methods we have used for analyzing the quality of the different components of the student models.

3.3.1 Educational setting and the Digital Mathematics Environment

The participants in this study were 160 first-year students in the Social Sciences at Utrecht University. In the fall of 2016, these students took part in a mandatory statistics course as one of the first courses in their bachelor's degree program. This course lasted ten weeks, and covered the following five topics:

1. Descriptive statistics
2. *z*-values and sampling distributions
3. Hypothesis testing: *z*-tests
4. Hypothesis testing: *t*-tests for one sample and dependent samples
5. Hypothesis testing: *t*-tests for independent samples

Each topic consisted of a lecture followed by practice in a digital statistics module. Students were allowed to work on the modules individually or in groups and could choose to work at home or in supervised lab sessions.

The five digital modules were offered in the Freudenthal Institute's Digital Mathematics Environment (DME, see Drijvers et al., 2013). The DME offers support for a variety of interactions, such as number and formula input, multiple choice tasks, drag-and-drop tasks, and interactive animations. Immediate verification feedback is provided on students' answers, telling students whether their answer is correct, but not what the correct answer is. Moreover, for most task types, elaboration feedback is available to explain errors that have been made. Students are allowed to attempt tasks multiple times, and usually continue trying to solve each task until they succeed.

A typical DME page is shown in Figure 3.2. The circles in the bottom bar of the page indicate the student's progress in the module. These indicators turn green once the student has solved all tasks on the page correctly while they remain red as long as this is not the case. As suggested in literature (Brusilovsky, Sosnovsky, & Yudelson, 2009), such coloring of progress indicators has a strikingly motivational effect: in order to obtain green progress indicators students keep attempting tasks until finding the correct answer. In Figure 3.2, the indicators reveal that this student has completed pages 2, 3 and 4 correctly, and still has to work

on pages 5 to 11. Since page 1 only contains an introductory text and no tasks, its indicator is grey.

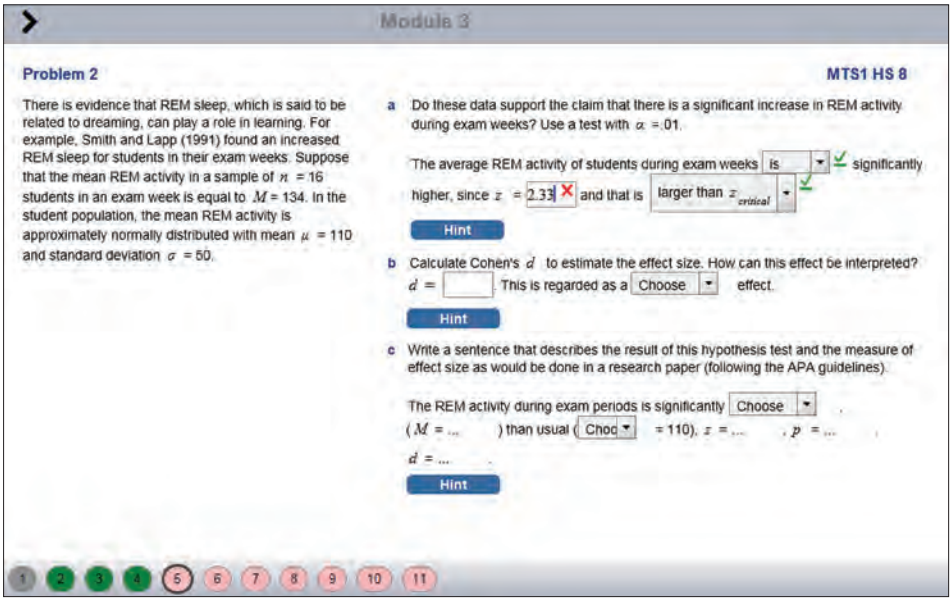

Figure 3.2 Translated DME page from the third module, on hypothesis testing

3.3.2 Didactical design

The modules used in this study were designed by the teachers of the statistics course, supported by DME experts. Each module consisted of a series of pages containing sets of closely related tasks. The number of pages varied between 12 and 22 and the number of tasks in the modules varied between 98 and 232. The page shown in Figure 3.2 is a translated version from the fifth page of the third module. It contains a context description on the left-hand side of the page and three sets of tasks on the right-hand side. Each individual interaction component is regarded as a task.

DME pages have a very flexible layout, which allows for different numbers of tasks on each page. Moreover, the DME facilitates initially hiding information that might not be needed by all students. The teachers made extensive use of this option to include hints and extra tasks serving as intermediate steps. On the page shown in Figure 3.2, hidden information is available through the hint buttons. The information that is revealed upon

clicking the topmost hint button is shown in Figure 3.3. Whereas students were obliged to complete all tasks on the main pages, the use of these hints and intermediate tasks was optional. Moreover, use of the hidden information did not affect the page indicators, so these would turn green once all tasks on the main page were completed correctly.

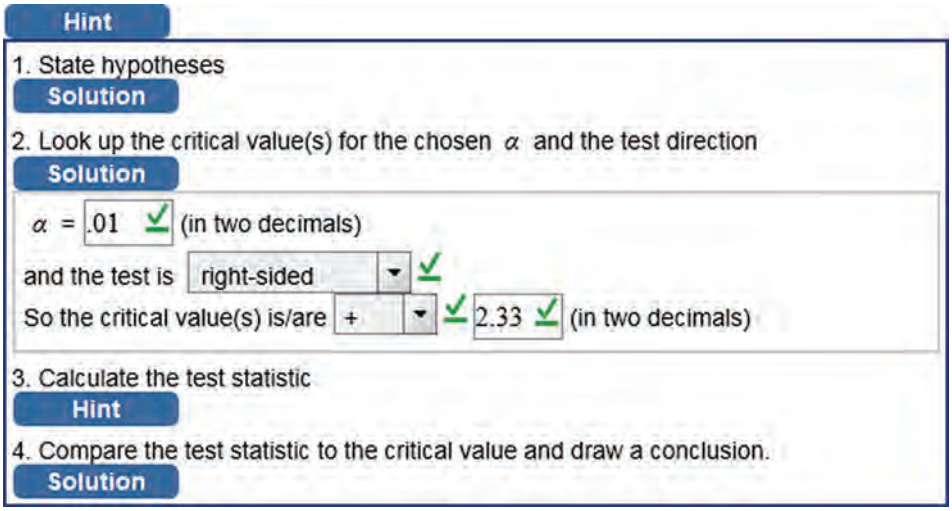

Figure 3.3 Initially hidden intermediate steps for the DME page shown in Figure 3.2

As recommended in literature, the modules made extensive use of real data sets and contained many tasks that focused on conceptual understanding and statistical reasoning. Most tasks in the modules were connected to a context and all contexts were based on real research projects and contained real data. In the modules, students were invited to engage deeply with these contexts. Contexts were used to address multiple concepts and to highlight aspects of the relations between concepts. On the example page from Figure 3.2, students are asked to carry out a hypothesis test, determine the effect size and report the results as would be done in a research article. Furthermore, contexts were deliberately varied to confront students with various applications and appearances of the different concepts: testing left-sided, right-sided or two-sided, positive and negative values of the test statistic, known and unknown population variances, significant and non-significant results and so on. Conceptual understanding was for example addressed by tasks asking students to interpret the rejection of a null hypothesis in the given context, or to describe the influence of sample

size on concepts like effect size or power. The number of procedural tasks was kept low by using SPSS-output regularly, instead of asking students to calculate the test statistic themselves in all tasks on hypothesis testing.

The use of real data sets and a focus on conceptual understanding and statistical reasoning had consequences for the ordering of tasks on each page and for the ordering of the pages themselves. In Figure 3.2, the ordering of the three sets of tasks is determined by their content: the hypothesis test needs to be carried out before calculating effect size or reporting the results. In the ordering of pages, difficulty level was taken into account, for example by introducing the more complicated *t*-test for independent groups after sufficient exposure to the easier *z*-test and *t*-tests for one group and for dependent groups. Finally, whereas on earlier pages concepts were typically addressed in isolation, later tasks required more and more understanding of combinations of and relations between concepts. For example, early pages contained separate sets of tasks for stating hypotheses, finding a critical value and calculating a test statistic, whereas later pages contained only one set of tasks asking the student to carry out the complete hypothesis test. After finishing the module, students were presented with their student model, which they could revisit any time after that.

3.3.3 Design of inspectable student models

Student models were implemented in all five modules. The student models were devised by the first author, in collaboration with two experts: the main teacher of the course and the fifth author. Three separate components were designed: domain models, Q-matrices with connections between KCs in the domain models and tasks in the modules, and a calculation method for calculating overlay scores.

The first step in domain model design was formulating KCs based on the tasks. Taking the tasks as the starting point may seem a reversed approach, since tasks are designed to cover a certain domain rather than vice versa. However, it is also an approach that teachers or designers could easily pursue. Because of the large responsibility that university teachers have for designing their own instructional material, design feasibility was deemed important in the context of this study. Since the purpose of the

student model was to promote conceptual understanding, KCs were mainly designed to represent declarative rather than procedural knowledge. To ensure that the domain model completely covered the domain in the end, the second step consisted of adding and adjusting KCs based on a consultation of already available domain models (ALEKS[1]) and other instructional material on the same topic (SURF[2]). In the third and final step of domain model design, the two experts were consulted and KC definitions were fine-tuned based on their comments.

A rather coarse-grained approach was adopted to design KCs, which means that KCs were relatively broad in scope. For example, instead of defining KCs for calculating different test statistics (*z*-test, *t*-test for one sample, and so on), a single KC was defined for calculating the test statistic. Although finer-grained domain models generally allow for more sophisticated diagnoses (Sosnovsky & Brusilovsky, 2015), we had two decisive reasons to opt for a coarse-grained approach. The first reason was student model comprehensibility, since the main purpose of the models was to offer students insight into their understanding of the domain. The second reason was, again, design feasibility; in this approach a quick analysis of tasks suffices to determine the KC(s) involved.

For each module a separate domain model was designed. However, since modules 3, 4 and 5 all covered hypothesis testing, their domain models overlapped to a large extent. The final domain models contained between 8 and 19 KCs. To improve comprehensibility of the student models, the KCs in each domain model were grouped into two to five categories.

Design of the Q-matrices was straightforward. Tasks were connected to all KCs that were related to the task. For example, tasks that involved finding a critical value were connected to both the KC on the critical value and the KC on the significance level, since the significance level is needed to find the critical value. The majority of tasks was connected to only one KC, but for some up to six KCs were judged relevant. To improve Q-matrix consistency, the two experts were invited to connect a subset of

1 Course materials downloaded from *www.aleks.com/about_aleks/course_products*, on October 16, 2015.
2 http://bit.ly/surfstat

the tasks to the domain models. Most expert connections were equal to the researcher's, and differences were discussed until consensus was reached.

The final component in student model design was the calculation method for overlay scores. This method was based on the number of attempts students needed to finish the tasks connected to each KC. A straightforward, simple numeric, implementation was chosen, in which each task connected to a KC contributed equally to its score. The formula used for calculation of the overlay scores is: formula used for calculation of the overlay scores is:

$$score_{KC,student} = \frac{\sum_{i=1}^{n} \frac{\sum_{j=1}^{m} a_{t_i,j}}{m}}{n},$$

with:

t_i: the ith task connected to this KC (i in {1, ..., n})
$a_{t_i,j}$: the jth attempt score by this student for task t_i (j in {1, ..., m}).

This formula can be explained as follows: for each task a task score was calculated as the mean attempt score over all attempts by this student on this task. The attempt score was 0 for incorrect attempts, 0.5 for half correct attempts (for example, if the answer still needed to be rounded off) and 1 for correct attempts. For instance, the task score for a student who first gave two incorrect answers before answering correctly was 0.33. The student's score for a KC was then calculated by averaging the task scores for all tasks connected to the KC.

Giving all tasks equal weight in the calculation of overlay scores may seem unfair, since students are likely to learn and hence perform better on later tasks than on earlier tasks. However, tasks also tended to become more complicated throughout the modules, requiring students to combine several concepts rather than using them in isolation. Since students were invited to study their student model only at the end of the modules, a final difficult task could easily result in an underestimation of the student's knowledge, if more recent tasks were assigned larger weight.

A translated example of a student model for the first module is presented in Figure 3.1. The domain model for this module contained five categories. In the inspectable student model, students could unfold each category (by clicking the + button) to view the individual KCs and their overlay scores. Category scores were calculated as the weighted mean of the KC scores in the category, weighted by the number of tasks to which they were connected.

3.3.4 Data collection

After the student models were implemented in the instructional modules, they were offered to the Social Sciences first-year students. Data collection focused on student perception (RQ1) and student model quality (RQ1 and RQ2). To investigate student perceptions about the student models, a short questionnaire was added at the end of each module, on the page in which students could inspect their student model. In this questionnaire, students were asked to respond to three statements, concerning the match between the tasks and the KCs, the clarity of the KC descriptions, and the scores in the overlay. Students could indicate their agreement with each statement on a five-point Likert scale.

The log files with student work on the five modules were the most important data source for evaluating the quality of the student model. Each week the student work for that week's module was exported from the DME. The first module contained a page with information on this study and asked students for their consent. Work from students who did not give consent was deleted (N = 26 out of 186 students) and all other log files were anonymized before further analysis. For each module, all students who attempted at least one task were included in the analysis. Table 3.1 summarizes properties of the students' work and provides the number of tasks in each module and the number of KCs in each student model. As can be seen in the table, student numbers slowly decreased from 160 students in the first module to 117 in the fifth. This can be attributed to students quitting the course or choosing other means for studying the course material.

Table 3.1 Data collection in the five modules

Module	Tasks	KCs	Students	Attempts per student (*SD*)	% attempted tasks (*SD*)
1	98	19	160	109 (36)	57 (14)
2	107	8	141	113 (43)	63 (17)
3	110	14	129	89 (40)	45 (17)
4	232	14	127	190 (75)	52 (18)
5	132	16	117	137 (66)	58 (18)

3.3.5 Data analysis

Student perceptions

The questionnaires were used to assess the suitability of the student models in the statistics modules (RQ1) from the students' perspective. Each of the five modules contained a questionnaire on the last page, and each questionnaire contained three statements, to which students could respond on a five-point Likert scale. For each of the fifteen statements, a mean score over all students was calculated as a measure of agreement of the students to the statement.

Student model quality

For the evaluation of student model quality, methods from both the didactics of statistics field and the student modeling field were combined. First, a learning curve analysis (Martin, Mitrovic, Koedinger, & Mathan, 2011) was carried out to assess domain model quality (RQ1) and to identify weaknesses in their design and implementation (RQ2). Next, these weaknesses were further investigated through didactical task analysis, which led to possible improvements of both the student models and the instructional modules (RQ2). Finally, predictive validity analyses (Sosnovsky & Brusilovsky, 2015) were carried out to assess both the quality of the overlays in the original design (RQ1) and in the design after implementing the improvements identified in the learning curve analysis and didactical task analysis (RQ2). In the following, the three methods and our implementation are described in more detail.

Learning curve analysis

Learning curve analysis is specifically aimed at evaluating the domain model. The assumption behind learning curve analysis is that learning

generally follows a power law. When first encountering a concept, students' incomplete understanding results in errors on tasks related to that concept. After more and more encounters with the concept, the students' understanding becomes more complete, resulting in a decrease in the number of errors related to the concept (Martin et al., 2011). In other words, for each KC in the domain model the error rate is expected to decrease, and if this is indeed the case, the KC is regarded as a valid unit of knowledge.

To generate learning curves, first for each student and each KC, the student's attempts on tasks connected to the KC were sorted in chronological order. This resulted in lists of attempts, in which, for example, the sixth attempt could be the first attempt by a student on the sixth task connected to the KC, the sixth attempt by this student on the first task, or anything in between. The length of the lists varied over students and KCs, since students needed different numbers of attempts to finish the tasks on the different KCs.

After ordering attempts, the correctness of each attempt was indicated. Because we were interested in the number of errors, we marked errors as 1 and other attempts as 0. Next, error rates for individual KCs were calculated for each attempt number, by dividing the number of students who made an error related to the current attempt number for a given KC by the total number of students who made an attempt for that attempt number for that KC:

$$\textit{Error rate nth attempt} = \frac{\textit{number of incorrect nth attempts on KC}}{\textit{total number of nth attempts on KC}},$$

These error rates were plotted against the attempt numbers and a power law was fitted, using the formula

$$\textit{Error Rate} = B \cdot \textit{AttemptNo}^{-\alpha}$$

with the decay factor α and starting value B as parameters. Moreover, R^2 was calculated as measure of goodness of fit.

Since students could attempt tasks multiple times and not all tasks were obligatory, the number of attempts for the different KCs varied over students. Consequently, the number of students decreased as the attempt

number increased. That is, for higher attempt numbers, the error rates were based on attempts by fewer students.

To ensure reliable error rates, Martin et al. (2011) recommend cutting off the learning curve after a certain attempt number. They propose two methods for defining the cut-off point: either by selecting an acceptable reduction in the number of students or by making a judgement call on where to cut off after examining where the learning curve seems to be deteriorating. Martin and his colleagues (ibid.) used a one-half cut-off, meaning that they cut off once only half of the students remained. In examining the learning curves for our domain models, we noticed that many learning curves deteriorated already after losing one third of the students who made a first attempt. Therefore, we decided to use a two-thirds cut-off. This higher cut-off level can be explained by the large number of non-obligatory intermediate steps in the modules: as can be seen in Table 3.1, the average number of attempted tasks was considerably lower than the total number of tasks in each module, caused by students skipping the non-obligatory intermediate steps. Therefore, for high attempt numbers, error rates were predominantly based on students who made use of the intermediate steps. This was a smaller, and probably weaker, group of students than the complete student population and hence the error rate was likely to increase with this decrease in student numbers.

Additional didactical task analysis

After assessing the quality of all individual KCs, some were found to have increasing rather than decreasing error rates. To explain these increasing error rates a didactical inspection of the instructional modules was carried out. To this end, repeatedly single tasks and sets of similar tasks were disconnected from the KC and new learning curves were generated. Once a decreasing learning curve was found, the set of tasks that was currently disconnected was designated as a possible cause for the originally increasing learning curve. Next, a didactical analysis of these tasks was performed to find a sound didactical explanation for the increasing learning curve. In cases where it did not prove possible to designate just one task or set of similar tasks as a possible cause, all tasks connected to the KC were analyzed from a didactical perspective, and especially the concepts judged to be addressed in the tasks were reconsidered. Through this interplay

between learning curve analysis and didactical task analysis, we attempted to improve the quality of both the student models and the instructional models themselves.

Predictive validity analysis

Both the learning curve analysis and the subsequent didactical task analysis specifically targeted the domain model and did not address the overlays. Therefore, overlay quality was assessed through a third method: predictive validity analysis.

In a predictive validity analysis, student performance is predicted based on the student's previous attempts. The correlation between these predictions and the actual student performances is used as a quality measure for the overlays. To find this correlation, the following two values were calculated for each KC involved in each attempt by each student:

- The student's prior knowledge level for the current KC up to the current attempt;
- The student's posterior actual performance for the current KC after the current attempt.

The prior knowledge levels were calculated based on the tasks that a student had already attempted, using our calculation method for overlay scores. Posterior student performance was based on attempts following the current attempt. Sosnovsky and Brusilovsky (2015) argue that correlating single step performances with knowledge predictions is problematic and propose using simple moving averages over five attempts. We followed this suggestion by selecting the first five attempts after the current attempt that also involved the current KC. For these five attempts, the average attempt score was calculated (again, correct = 1, half = 0.5 and incorrect = 0) as a measure of actual performance.

3.4 Results

We first present the results of the questionnaires about the students' perceptions of the student models. Next, we present the main quality assessment of the implemented domain models: learning curve analysis. The results of this learning curve analysis form a starting point for

didactical task analysis. This leads to the identification of four problems in implementing inspectable student models in rich instructional modules for statistics, and to possible improvements of the domain models, Q-matrices and the instructional modules to resolve these four problems. Finally, the results of the predictive validity analysis reveal the quality of the overlays, as well as the value of the improvements from the didactical analysis.

3.4.1 Student perceptions of the student models

Table 3.2 summarizes the results of the questionnaires at the end of each module. A score of 1 corresponded with Totally disagree and a score of 5 corresponded with Totally agree. From the table, it can be seen that students agreed to a large extent with statements 1 and 2, and to a moderate extent with statement 3. The strong agreement with statements 1 and 2 suggests that students perceived the tasks in the modules and the KCs in the domain models to match well and that the descriptions of the KCs were clear. The moderate agreement with statement 3 implies that students thought the scores from the overlays represented their current knowledge of the concepts quite well. All in all, students seemed to perceive the student models as comprehensible and plausible.

Table 3.2 Questionnaire results

	Module 1	Module 2	Module 3	Module 4	Module 5
Statement 1: The tasks in the DME match well with the topics in the student model					
Mean	4.30	4.42	4.26	4.44	4.43
SD	0.67	0.69	0.68	0.58	0.59
N	125	89	54	27	23
Statement 2: The descriptions of the topics are clear					
Mean	4.23	4.21	4.17	4.15	4.22
SD	0.73	0.82	0.91	0.82	0.80
N	125	89	54	27	23
Statement 3: I think the scores on the topics are a good representation of my knowledge					
Mean	3.85	4.08	4.10	4.00	3.90
SD	0.73	0.70	0.77	0.89	0.89
N	120	88	52	26	21

3.4.2 Domain model quality according to learning curve analysis

To assess the quality of the domain models, learning curves were generated for 56 out of 71 KCs in the five domain models. For the remaining 15 KCs, not enough data were available to obtain a learning curve. For each of these 56 KCs the error rates were computed and a power law curve was fitted. This resulted in decreasing learning curves for 34 KCs and increasing learning curves for 22 KCs.

For the 34 KCs with decreasing learning curves the mean goodness of fit (R^2) was 0.49 (SD = 0.29). For the increasing learning curves the fits were generally weaker, with mean goodness of fit 0.35 (SD = 0.33).

As mentioned before, KCs with learning curves that decrease as a power function represent cognitively valid units of knowledge. Therefore, in the initial design 34 out of 71 KCs were well-defined. It may seem disappointing that only half of the KCs were well-defined, and this indeed suggests that just implementing student models in didactically grounded instructional modules does not automatically result in high-quality feedback to students. But the presence of increasing learning curves also provides a good starting point for further analysis: didactical inspection of connected tasks may shed light on prerequisites, opportunities and limitations in implementing student models in didactically grounded statistics modules.

3.4.3 Underlying problems based on didactical analysis

For the 22 KCs with increasing learning curves connected tasks were analyzed from a didactical perspective to find underlying problems that caused the learning curves to increase. Four different problems were identified; some increasing learning curves were completely explained by one of these problems, whereas for other KCs two problems applied. The first problem relates to single tasks distorting the learning curve. The second concerns groups of tasks that address concepts from different perspectives and the third problem concerns tasks that involve multiple concepts. The fourth and final problem concerns a lack of opportunities for in-depth thinking about the KC in the learning module. This final problem also applies to the 15 KCs for which not enough data were available to obtain a learning curve. The four problems are elaborated below.

Problem 1: Tasks with specific purposes

For 12 out of the 22 KCs, the increasing learning curve could be attributed to one or two single tasks; disconnecting these tasks from their KC yielded a decreasing learning curve. Didactical analysis of the disconnected tasks, compared with the tasks that remained connected, revealed that these tasks often had a specific purpose in the module.

In six of these cases, the tasks that were disconnected were the first tasks in which the students encountered the KC. An example is a KC on the significance level, for which the learning curve is shown in the left-hand graph in Figure 3.4. The error rate for the first attempt is remarkably lower than for the subsequent attempts. Disconnecting the first task from the KC yielded the right-hand learning curve in Figure 3.4. Without the first task, the learning curve became decreasing with a very high goodness of fit (R^2 = .98), indicating that the remaining tasks constituted a valid KC.

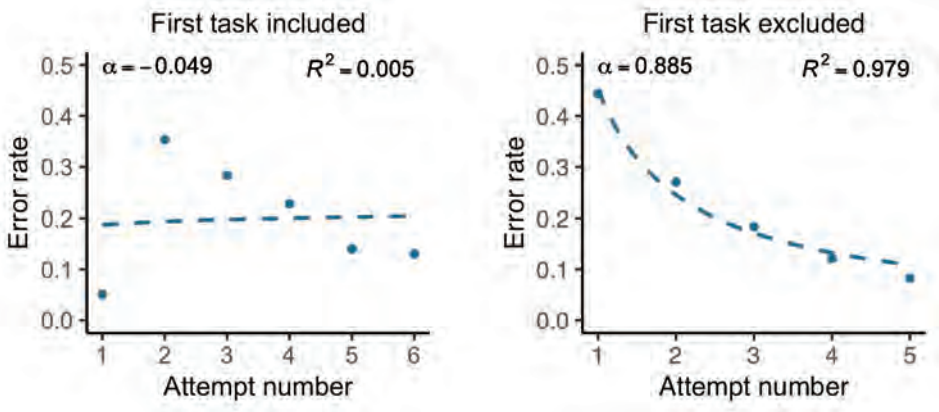

Figure 3.4 Error rates for KC Significance level

A didactical inspection of the disconnected task and tasks that remained connected revealed that the first task was easier than the other tasks. The first task only asked students to reproduce a value for the significance level from the problem description. Later tasks required students to use the significance level for defining the rejection region in a hypothesis test. Such easy first tasks on a concept occurred more often in the modules. Apparently, for the designers of these modules it was natural to introduce concepts in a quite gentle manner. The purpose of these easy first tasks is to enhance students' self-confidence, rather than to give students the

opportunity to practice. This resulted in very low initial error rates and increasing error rates once tasks became more demanding.

Another specific purpose that tasks could have was to emphasize a specific aspect or detail of a KC. This was the case for three KCs. At first sight, the tasks causing the increasing learning curve were very similar to the tasks that remained connected. A closer didactical inspection revealed that the disconnected tasks had a slightly different emphasis concerning the KC. This was, for example, the case for the KC on calculating Cohen's *d* in the third module. The learning curve for this KC is shown in the left-hand graph of Figure 3.5.

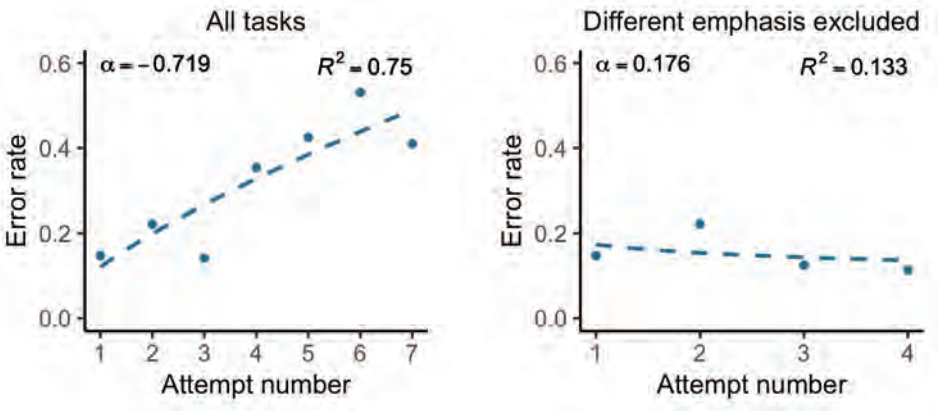

Figure 3.5 Learning curves for KC Cohen's *d*

The six tasks connected to this KC required students to calculate or interpret a value of Cohen's *d*, a measure of effect size. Cohen's *d* is calculated as $d=\frac{|M-\mu|}{\sigma}$, with M the sample mean, μ the population mean and σ the standard deviation in the population. In four out of the six tasks, M was larger than μ and hence explicitly taking the absolute value was not necessary. In the two tasks causing the increasing learning curve, however, M was smaller than μ. Many students forgot to take the absolute value and erroneously gave a negative effect size as answer. In other words, these two tasks emphasized the fact that Cohen's *d* is always positive, whereas the other tasks only concerned using the correct values in the calculation. Since these two tasks were the fourth and fifth task connected to this KC, the students' errors on these tasks caused relatively high error rates for attempt numbers four and higher. Disconnecting these tasks with a slightly

different emphasis resulted in the decreasing learning curve displayed in the right-hand graph of Figure 3.5.

Most increasing learning curves that could be attributed to one or two single tasks could be explained by these specific purposes for tasks. However, for five KCs the didactical analysis, and especially an inspection of the errors students made, identified flaws in task design. Although the modules were thoroughly tested by colleagues of the designers, this was the first time that students worked with them. Flaws in task design resulted in confusion among students, and consequently in high error rates.

Problem 2: Concepts addressed from multiple perspectives

For six KCs with increasing learning curves, we have been able to partition the connected tasks into groups that each had a decreasing learning curve. These groups were identified by setting up a detailed description of the concepts addressed and actions required in all connected tasks.

An example is the KC on the mean. Its learning curve is shown in the top-left graph in Figure 3.6. By analyzing the connected tasks, three conceptually different task types were distinguished. Of the fifteen connected tasks, eight concerned calculating or estimating the value of a mean, based on given data. Four others concerned the appropriateness of using the mean for different types of variables. The remaining three tasks concerned the calculation of a standard deviation, for which calculating the mean is an intermediate step.

The learning curves for each of the three groups of tasks are shown in the top-right, bottom-left and bottom-right graph in Figure 3.6. While the learning curve for the complete KC increases, the learning curves for each of these groups of tasks decrease. This implies that for students finding the mean, judging the appropriateness of the mean for different types of variables, and finding the standard deviation involved different procedures and types of reasoning.

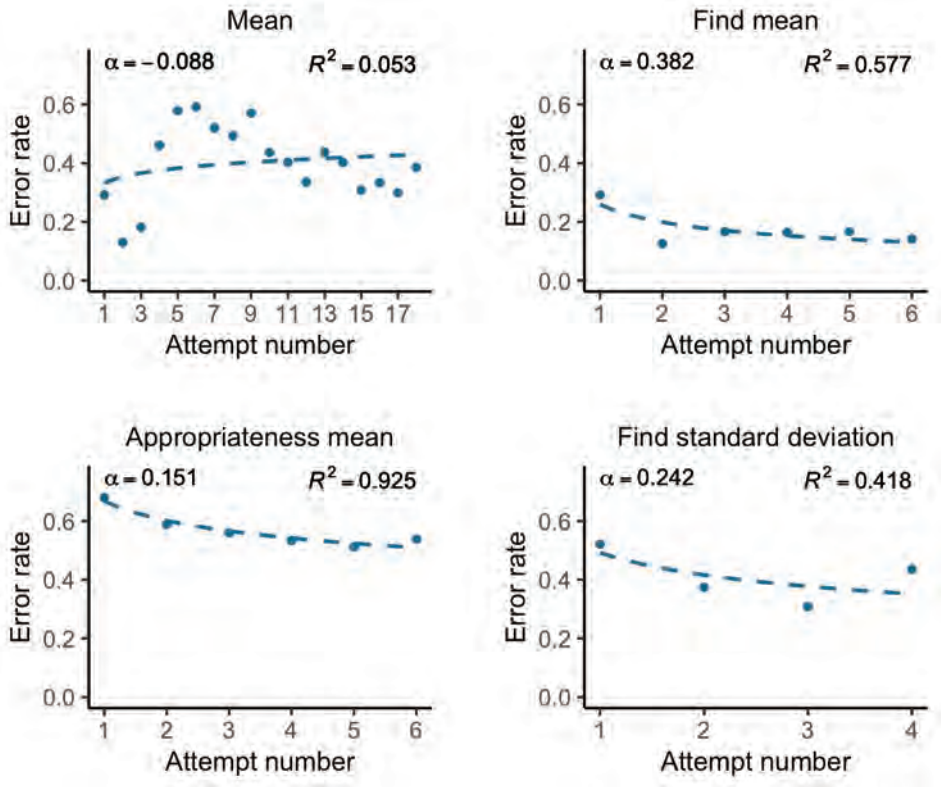

Figure 3.6 Learning curves for the mean and three subgroups

Problem 3: Tasks involving multiple concepts

In the example above, finding the mean and judging the appropriateness of the mean are both only related to the concept of mean, but finding the standard deviation additionally relates to the concept of standard deviation. This occurs frequently: in many tasks multiple concepts are involved. As mentioned before, in designing the Q-matrices tasks were connected to all KCs that were judged to be involved in the task. For four KCs, including the KC on the mean, this turned out to be problematic. Although these KCs were involved in all tasks connected to them, not all errors that students made could be attributed to these KCs. For some of the connected tasks, different KCs turned out to be the *bottleneck* KC, that is, the KC that mainly caused errors on the task. The example with different subgroups of "mean" above can serve to illustrate this idea of bottleneck KCs: errors that students made are more likely due to a lack of understanding of the

standard deviation itself, than to a lack of understanding of the mean. Therefore, errors that students made on these tasks caused an unfairly high error rate for the KC on the mean. Since the tasks on standard deviations appeared later in the module than other tasks involving the mean, this may have contributed to the increasing learning curve for the mean.

Problem 4: Lack of opportunities for in-depth thinking about concepts

The final problem that was identified for increasing learning curves concerns KCs with an overall low error rate. For four KCs with increasing learning curves the error rates never rose above 0.3. For such small error rates, slight fluctuations that are likely due to chance may have caused the learning curve to increase rather than decrease. Although a low overall error rate indicates that the KC is easy for students, most of these five KCs were not judged to be easy by the designers of the course. Rather, they concerned interpreting the meaning of concepts and understanding the relation between concepts, which are generally considered as difficult aspects of the statistics domain. In other words, although the designers included tasks addressing these difficult KCs, they did not succeed in addressing the actual difficulties that students have regarding these KCs. This discrepancy can be attributed to task design; apparently the tasks connected to the KC were easy for students and did not engage them in thinking about the statistical concepts in depth. Indeed, tasks connected to these KCs were often multiple choice, with only two options to choose from. With such little variation in possible answers, any misconceptions that students may have had were likely to stay unnoticed; students did not have the opportunity to make many errors and learn from these errors.

Another case in which students did not have enough opportunity to make errors and reflect on them was formed by the 15 KCs for which no learning curve could be generated. These KCs were all connected to at most two tasks, which were often multiple choice tasks with at most four options to choose from. For these KCs, students just did not make enough attempts to obtain a learning curve. This suggests that they probably also did not make enough attempts to gain deeper understanding of the KCs.

Improving modules and student models

The four problems together provide a basis for improvement of both the student models and the instructional modules. To obtain a first impression of the value of these improvements, we carried out a second learning curve analysis with a revised version of the domain models and Q-matrices. This evaluation was performed with the same student data as the original analysis, which means that no adjustments could be made to the tasks. Therefore, connections to tasks that needed redesign were just removed from the Q-matrix. Moreover, KCs for which task redesign was needed to create more opportunities for errors were removed from the domain model to enable this analysis.

The improvements that we could make, adjustments to the domain models and Q-matrices, were mostly easy and straightforward. Tasks with specific purposes that distorted the learning curves were easily recognized and disconnected from their KCs. For concepts that were addressed from multiple perspectives, a more thorough analysis was needed to identify the different perspectives to split the KC into, but subsequently reconnecting tasks was again straightforward. Similarly, identifying bottlenecks for tasks needed some analysis, but next disconnecting tasks from non-bottleneck KCs was easy.

For the new domain models, error rates were again calculated for all individual KCs and learning curves were fitted. Table 3.3 summarizes the results of the learning curve analysis for both the original and the new domain models.

Table 3.3 Comparing individual KCs in the original and new domain models

Module	Original			New		
	Increasing	Decreasing	Too little data	Increasing	Decreasing	Too little data
1	6	8	5	0	14	0
2	2	4	2	1	7	0
3	7	4	3	3	8	0
4	3	9	2	0	14	0
5	4	9	3	1	11	0
Total	22	34	15	5	54	0

The number of increasing learning curves decreased drastically from original to new domain models. Moreover, for all remaining KCs in the new models enough data were available to generate a learning curve, and hence, for each KC enough tasks were available to provide students with ample practice opportunities. The five KCs for which the learning curves were still increasing all had overall low error rates and were regarded as easy KCs. All in all, the combination of learning curve analysis and didactical task analysis has led to a marked improvement of the domain models.

3.4.4 Overlay quality according to predictive validity analysis

Our final analysis is aimed at evaluating the quality of the final part of the student models, the overlays. To evaluate overlay quality, prior and posterior student performances were calculated for each attempt by each student on each KC. The prior student performance was the score the student model would attribute to that KC for that student, up to that attempt. The posterior student performance was calculated based on the five next attempts the student made on that KC. In total, the list of prior and posterior student performances contained 116729 prior-posterior pairs. Pearson's correlation coefficient for this list was $r = .315$. Although this value indicates a positive correlation between the students' understanding as predicted by the model and the students' actual performance, the correlation is regarded as weak (Evans, 1996). One possible explanation can be found in the formula used, which is, as we discussed earlier, a fairly naïve implementation. But since prior and posterior performances were calculated for each individual KC, the quality of the KCs themselves is also likely to influence the quality of the overlays. Therefore, after improving the domain models and Q-matrices based on the learning curve analysis and didactical task analysis, we reassessed the overlay scores with a second predictive validity analysis. For the overlays resulting from the new domain models, we found a Pearson's correlation coefficient of $r = .423$. This is a moderate positive correlation (Evans, 1996) that is markedly better than the one for the original domain models. This implies that the improvements in the domain model indeed contributed to more sound student models for didactically grounded sequential modules.

3.5 Conclusion

The two research questions addressed in this study have been:

RQ1 *Are inspectable student models suitable for implementation in didactically grounded, sequential statistics modules consisting of closely related tasks?*

RQ2 *How can didactical analysis inform design of inspectable student models, and, vice versa, how can student model evaluation methods inform didactical design?*

The suitability of inspectable student models (RQ1) was evaluated at two levels: a questionnaire asked students about their perception of the student models, while learning curve analysis and predictive validity analysis were used to assess the internal quality of the student models.

Results from the questionnaire showed that students valued the student models for their clarity and close connections to the tasks in the modules. These results are in line with findings by Bull (2004) that inspectable student models are useful to students, and suggest that these findings can be extended to sequential instructional modules. However, the results from the learning curve analysis and the predictive validity analysis were less positive. The learning curve analysis revealed that in the initial domain models, only half of the KCs were immediately well-defined. Furthermore, in the predictive validity analysis we only found a weak positive correlation of $r = .315$ between the predicted and the actual student performance. These results provided us with a starting point for improving our design and addressing the second research question of the article.

Learning curve analysis combined with didactical task analysis indeed proved to be an insightful approach for identifying weaknesses of the student models and instructional modules. We identified four specific problems: tasks with specific purposes in the instructional modules, concepts addressed from multiple perspectives, tasks involving multiple concepts and lack of opportunities for in-depth thinking about statistical concepts.

The first of these problems is a product of the didactical design of sequential modules: easy tasks deliberately crafted to introduce a concept gently or to emphasize a particular aspect of a concept. Although such tasks are useful in the module, they are not suitable for informing student models, because their error rates are very different from error rates of other tasks involving the same KC. Rather than discarding tasks with specific purposes, which would be the approach for databases of independent tasks (Pavlik, Cen, & Koedinger, 2009), the most sensible approach for sequential modules is to exclude connections between such tasks and the related KCs from the Q-matrix. As a consequence, instructional modules can contain tasks that are didactically meaningful for the module, but do not inform the student model.

The second problem, concepts addressed from multiple perspectives, results from our choice of coarse-grained domain models. Since coarse-grained KCs accumulate evidence from many underlying atomic KCs, the models they produce are often messy (Sosnovsky & Brusilovsky, 2015). Yet, in spite of this low modeling quality, coarse-grained KCs can still provide good navigational anchors, since they are easy to understand and interpret for students and easy to design for teachers (Sosnovsky & Brusilovsky, 2015). Furthermore, learning curve analysis combined with a didactical inspection of connected tasks has long been recognized as a useful tool for identifying and splitting KCs with too broad definitions (Corbett & Anderson, 1995).

The third problem also results from a choice made during the design of student models, namely connecting tasks to all related KCs. For correctly answered tasks, this approach works well: a correct answer is a proof that a student understands all related KCs. However, an incorrect answer can have as many causes as there are KCs connected to a task (and their combinations). Didactical task analysis may reveal which KC is the most likely cause for errors on a task, that is, which KC is the bottleneck for that task. Since errors on the task may cause unfairly high error rates (and thus unfairly low overlay scores) for the other connected KCs, it may be advisable to remove connections between tasks and non-bottleneck KCs.

Finally, the fourth problem (lack of opportunities for in-depth thinking about statistical concepts) can manifest itself in two ways: an overall low error rate, or a lack of sufficient information from which to generate a learning curve. In both cases, the combined learning curve analysis and didactical task analysis may reveal weaknesses in the design of the instructional module itself which would have otherwise stayed unnoticed. Redesign of tasks should focus on creating more opportunities for students to make errors that reflect their misconceptions and to learn from these errors.

We used the findings from the combined learning curve analysis and didactical task analysis to redesign the instructional modules. The resulting inspectable student models performed markedly better than the original ones. In the original models, only 34 out of 71 KCs were characterized by learning curves that decreased according to a power law. In the new models, the number of such learning curves is 54 out of 59. Moreover, the combined predictive validity of the new student models improved considerably compared to the original models: $r = .423$ vs. the original $r = .315$. This shows that didactical analysis can indeed provide valuable information for designing student models. Moreover, learning curve analysis did not only provide a basis for improving student models, but also yielded leads for improving the design of the instructional modules themselves. In this way, the fields of didactics of statistics and inspectable student models can strengthen each other in the design of interactive and engaging instructional material.

3.6 Discussion

The four identified problems together comprised explanations for all increasing learning curves we found and provided a basis for improving both the student models and the instructional statistics modules. It can be noted that whereas the first problem is specific for sequential instructional modules, the other problems could also apply to sets of independent tasks. In fact, as mentioned above, Corbett and Anderson (1995) already used capricious learning curves as motivation for adjusting their domain model by splitting KCs. Nevertheless, all four problems illustrate how didactical task analysis can inform explanations for increasing learning curves, and

vice versa, how increasing learning curves can identify tasks that need didactical reconsideration.

Although combining the fields turned out to be fruitful in this study, some remarks are in order. First of all, the setting was a university statistics course. Since university teachers often have a large responsibility for designing and arranging their own teaching, we pursued a design approach that seemed feasible for them. To this end, we designed modest sets of independent KCs, and connected tasks to all related KCs. This resulted in several increasing learning curves, which we resolved by removing connections between KCs and tasks for which that KC did not prove a bottleneck. Drawbacks of removing this connection are that correct answers can no longer be used to increase the score for such a non-bottleneck KC, and that, in fact, different KCs may prove the bottleneck for different students. A more robust solution would therefore be to implement relations between KCs (Brusilovsky & Millán, 2007). Further research is needed to establish the feasibility of this approach for university teachers.

Another drawback of our student model was the rather low predictive validity, which was probably caused by our choice of a simple numeric overlay model. An uncertainty-based overlay model seems promising for improving predictive validity (Sosnovsky & Brusilovsky, 2015). A second advantage of implementing an uncertainty-based approach may be that uncertainty can be made visible to the students, which might offer them useful information for their planning and navigation (Bull & Kay, 2007).

Finally, in our evaluation of possible improvements to the student models and instructional modules, no tasks were redesigned and no new data were collected, so further research is needed to fully establish the value of these improvements. One aspect to specifically consider is whether identified weaknesses in the instructional modules do indeed concern the modules themselves, or rather the suitability of the modules for the implementation of a student model. In other words, otherwise appropriate learning modules might need adjustment (and addition of tasks in particular) to also collect enough information for every KC in a student model.

Acknowledgements

We would like to thank teachers Jeltje Wassenberg-Severijnen and Corine Geurts for their constructive collaboration in designing and delivering the instructional modules and student models.

CHAPTER 4 Enhancing learning with inspectable student models: worth the effort?

Tacoma S. G., Geurts, C., Slof, B., Jeuring, J. T., & Drijvers, P. H. M. (2020). Enhancing learning with inspectable student models. *Computers in Human Behavior*, *107*, 106276. doi: 10.1016/j.chb.2020.106276

Author contributions: Sietske Tacoma: Conceptualization, Visualization, Formal analysis, Software, Writing - original draft. Corine Geurts: Conceptualization, Methodology, Writing - original draft. Bert Slof: Conceptualization, Methodology, Writing - review & editing. Johan Jeuring: Writing - review & editing, Supervision. Paul Drijvers: Writing - review & editing, Supervision.

Abstract In electronic learning environments, information about a student's performance can be provided to the student in the form of an inspectable student model. While relatively easy to implement, little is known about whether students use the feedback provided by such models and whether they benefit from it. In this study, the use of inspectable student models in an introductory university statistics course by 599 first-year social science students was monitored. Research questions focused on whether students sought feedback from the student models, which decisions for subsequent study steps they made, and how this feedback seeking and decision making related to results on their statistics exams. Results showed a large variety among students in feedback-seeking and decision-making behavior. Lower student model scores seemed to encourage students to practice more on the same topic and higher scores seemed to evoke the decision to move to a different topic. Viewing frequency and amount of variety in decision making were positively related to exam results, even when controlling for total time students worked. These findings imply that inspectable student models can be a valuable addition to electronic learning environments and suggest that more intensive use of inspectable student models may contribute to learning.

Keywords Feedback-seeking behavior ◆ Higher education ◆ Inspectable student model ◆ Log file analysis ◆ Statistics education

4.1 Introduction

University education puts high demands on students in taking responsibility for their learning (Krause & Coates, 2008; Torenbeek, Jansen, & Suhre, 2013). A potentially effective way to support them in doing so is to offer formative assessment opportunities: assessment of their performance aimed at improving the learning process prior to grading (Birenbaum et al., 2015; Timmers, Braber-van den Broek, & Van den Berg, 2013). Whereas many educators and researchers advocate the potential of formative assessment for learning, sound empirical evidence for this is lacking (Hendriks, 2014). Scarce (Robinson, Myran, Strauss, & Reed, 2014) and ineffective (Bennett, 2011; Heitink, Van der Kleij, Veldkamp, Schildkamp, & Kippers, 2016) implementations of formative assessment in educational settings are regularly voiced explanations for this lack. To reach its full potential, formative assessment should be a cyclical process (Gikandi, Morrow, & Davis, 2011). Besides gathering information about student performance, two other elements are part of such formative assessment cycles, namely providing tailored feedback on performance and deciding on actions to enhance learning based on the provided feedback (Antoniou & James, 2014; Black & Wiliam, 2012). Whereas educational practitioners gather a lot of assessment data (e.g., Tempelaar, Rienties, & Giesbers, 2015), they often experience difficulties in proving tailored feedback and determining how their students make use of it. To address this, more insight into the interplay between the provided feedback and students' feedback-seeking and decision-making behavior is needed. The current study addresses this by implementing and examining formative assessment cycles – by means of inspectable student models – in an electronic learning environment in the context of a university statistics course.

For statistics education, the use of formative assessment – e.g., self-tests – has been recommended by several authors (Carver et al., 2016; Tishkovskaya & Lancaster, 2012). The low-stake assessment setting might support students in reducing statistics anxiety (Chew & Dillon, 2014) and procrastination (Onwuegbuzie, 2004), two factors that often result in lower grades for statistics (Paechter, Macher, Martskvishvili, Wimmer, & Papousek, 2017). By conducting self-tests, students have the opportunity to gain insight into their current mastery of the study domain (Dirkx, Kester,

& Kirschner, 2014). For the case of statistics, this study domain involves a large number of abstract concepts (Castro Sotos, Vanhoof, Van den Noortgate, & Onghena, 2007). Hence, the tailored feedback element of the formative assessment cycle should support students in gaining insight into their mastery of these various concepts. A promising operationalization of feedback in this respect is the inspectable student model, that can be offered to students within electronic learning environments. In this study, we examine students' use of inspectable student models, i.e., whether and how students consult inspectable student models and make decisions on actions after consultation, and its effect on students' performance on a statistics exam.

4.2 Inspectable student models in electronic learning environments

Electronic learning environments are gaining in popularity for realizing formative assessment in education (Van der Kleij, Timmers, & Eggen, 2011). Due to technological advancements (e.g., open source, interactive visualizations, learning analytics) implementing such tools in educational practices nowadays requires less money and effort than in the past, and these advancements also provide more opportunities for integrating the complete formative assessment cycle in the educational design. Electronic learning environments have the advantage that information about student performance is usually automatically captured and stored (e.g., log files) by means of a student model: a representation of a student's current mastery of important topics in the study domain (Herder, Sosnovsky, & Dimitrova, 2017). A visualization (e.g., figure, table) of the student model that students can consult – an *inspectable student model* – can serve as the tailored feedback element in the formative assessment cycle. Enriching electronic learning environments with inspectable student models has the potential to foster student learning in two ways. First, inspectable student models provide an overview of the important topics in the domain, which can support students in understanding the domain structure (Mitrovic & Martin, 2007). Second, inspectable student models provide an estimate of the student's current mastery of the topics included in the model. Low estimates for topics might stimulate students to exert more effort and practice on these topics. Furthermore, when estimates conflict with a

student's own perception of his or her mastery level, the student is more likely to consider further practice (Bull & Kay, 2007; Long & Aleven, 2011). Hence, enriching electronic learning environments with inspectable student models is an added service, which could support students in deciding on appropriate subsequent actions (e.g., selecting additional practice tasks).

Earlier studies revealed that students value the presence of inspectable student models in weekly homework sets (Mitrovic & Martin, 2007; Tacoma, Sosnovsky, Boon, Jeuring, & Drijvers, 2018 (see Chapter 3 of this thesis)). To our knowledge, previous research focused on the effects of inspectable student models combined with either (1) task selection adapted by the electronic learning environment based on the content of the student model (e.g., Brusilovsky, Sosnovsky, & Yudelson, 2009), or (2) monitoring of and feedback on task selection by the student (Mitrovic & Martin, 2007). Hence, the potential of integrating inspectable student models as an added service, to strengthen the formative assessment cycle through tailored feedback while leaving control over task selection fully with students, has not been studied extensively. Thus, it remains unclear how this added service affects student learning, a knowledge gap that this paper aims to fill.

For feedback to affect student learning, students need to actively seek for it, process it and decide which, if any, subsequent actions to carry out (Timmers et al., 2013). Various factors, such as motivation and accessibility of feedback information, may influence whether and how students engage in such behavior. To better understand these factors and, more specifically, how providing inspectable student models might foster student learning, more insight into students' feedback-seeking and decision-making behavior is required.

4.3 Feedback-seeking and decision-making behavior

Feedback-seeking behavior has been defined as the proactive search for feedback information in one's environment (Ashford & Cummings, 1983). Although inspectable student models are intended to foster student learning, there is no guarantee that students will engage in a proactive search for the feedback the student models provide, especially when this

is not a mandatory learning activity. For a student to exhibit feedback-seeking behavior, the assumed values should outweigh the assumed costs (Anseel, Beatty, Shen, Lievens, & Sackett, 2015; Ashford & Cummings, 1983). In the context of the present study, students should see the value of inspecting the student model as well as undertaking subsequent actions based on the provided feedback. According to Anseel, Lievens and Levy (2007) students might value feedback for different motives, namely: self-assessment (i.e., knowing how well one is doing), self-improvement (i.e., acquiring a higher mastery level), self-enhancement (i.e., coping with stress and anxiety), and self-verification (i.e., maintaining consistency between self-conceptions and new self-relevant information).

Especially students with strong self-improvement motives are more inclined to exhibit feedback-seeking behavior when they value the tool's potential for their learning process. The self-improvement value is particularly relevant when a student considers appropriate subsequent study steps, for example immediately after completing an initial set of practice tasks (Gikandi et al., 2011). Whether and how the provided feedback affects a student's decision making at such moments depends on several factors, such as perceived usefulness of the feedback (Harks, Rakoczy, Hattie, Besser, & Klieme, 2014) and the student's desire and intention to respond to the feedback (Kinicki, Prussia, Wu, & McKee-Ryan, 2004). In the context of this study, feedback indicating that current mastery is below the expected standards could lead to more practice and more feedback-seeking behavior (Hattie & Yates, 2014; VandeWalle & Cummings, 1997). If, however, the perceived costs of exposing one's uncertainty and need for help outweigh the student's value of self-improvement, such feedback might also lead to less feedback-seeking behavior, to avoid loss of face and ego costs of repeated negative feedback (Abraham, Burnett, & Morrison, 2006; Timmers et al., 2013). Yet, for inspectable student models these costs are relatively low compared to seeking feedback from a tutor or peer (Timmers et al., 2013). Receiving feedback indicating that the current mastery level is above the expected standard can also have diverse effects on both practice and subsequent feedback-seeking behavior. Students will only be inclined to practice more and exhibit more feedback-seeking behavior when they expect that the additional time investment will result in a gain in mastery level.

Previous studies on feedback-seeking behavior revealed no strong relationship between feedback-seeking behavior and performance (Anseel et al., 2015). When one attaches a high value to the feedback, one is inclined to proactively seek for (additional) feedback (Morrison & Cummings, 1992; Tuckey, Brewer, & Williamson, 2002). However, more feedback-seeking behavior does not automatically result in better performance such as a higher mastery level (Ang, Cummings, Straub, & Earley, 1993; Ashford & Black, 1996). Similarly, a review by Crommelinck and Anseel (2013) questioned the implicit assumption that feedback seeking is positively associated with performance, since most of the studies pay little empirical attention to the question whether and how feedback-seeking behavior affects performance. Consequently, a more in-depth understanding of the factors that explain whether and how feedback seeking leads to better performance is needed.

To this end, the current study examines the interplay between students' use of inspectable student models, i.e., their feedback-seeking and decision-making behavior, and their exam grades for the case of a university statistics course. The study is guided by three research questions:

RQ1 *How do first-year university students in social science seek feedback from inspectable student models in an introductory statistics course?*

RQ2 *How does feedback from inspectable student models inform these students' decisions about subsequent actions?*

RQ3 *How does these students' feedback-seeking and decision-making behavior relate to performance on a statistics exam?*

4.4 Materials and methods

4.4.1 Participants

Participants were 599 first-year university students who were enrolled in an introductory Methods and Statistics course at a Dutch research university. To be eligible for enrollment at this university, students needed to have followed a pre-university track in secondary education or at a university of applied sciences, which means that these students belonged to the top 20% of students their age. The course was mandatory for all bachelor's

degree programs in the social sciences. The students were informed about this study and were asked for their consent. Of the 1025 students who were enrolled in the course, 599 made use of the electronic learning environment and gave consent for the use of their work and exam results for this study. Of the 599 students, 77% was female and 23% was male. Their ages varied between 17 and 43 years (M = 19.5, SD = 2.2).

4.4.2 Description of the course and the electronic learning environment

The Methods and Statistics course was an eight-week course in which new methods and statistical concepts were introduced in week 1, 2, 4, 5 and 7. Intermediate exams were administered in week 3 and in week 6, and the final exam was administered in week 8. Learning objectives of the course were outlined in a course manual. In the weeks in which new concepts were introduced, a lecture on these concepts was given and students were offered online homework sets on the statistical topics. Students could choose to work on these homework sets at home or in lab sessions supervised by teachers. Tasks from the homework sets and their relations with the learning objectives were discussed in weekly discussion sessions.

The electronic learning environment in which the homework sets were made available was the Digital Mathematics Environment (DME, see Drijvers, Boon, Doorman, Bokhove, & Tacoma, 2013). Tasks in the homework sets addressed, for example, selecting appropriate measures of center and spread for given variables, or carrying out hypothesis tests for given situations and samples. Students received immediate feedback on the correctness of their answers, but the correct answer itself was not provided to students. Students could attempt answering tasks until they found the correct answer. A typical task from the first homework set is displayed in Figure 4.1. The tasks were designed by a team of teachers in the university's Methods and Statistics department.

In the weeks prior to the intermediate and final exams, extra practice sets were provided in the DME, allowing students to prepare for the exams. The extra practice sets contained between six and eleven new practice tasks on all topics covered so far. All homework and extra practice sets remained available for the students until the end of the course period. All interactions of the students in the DME were logged.

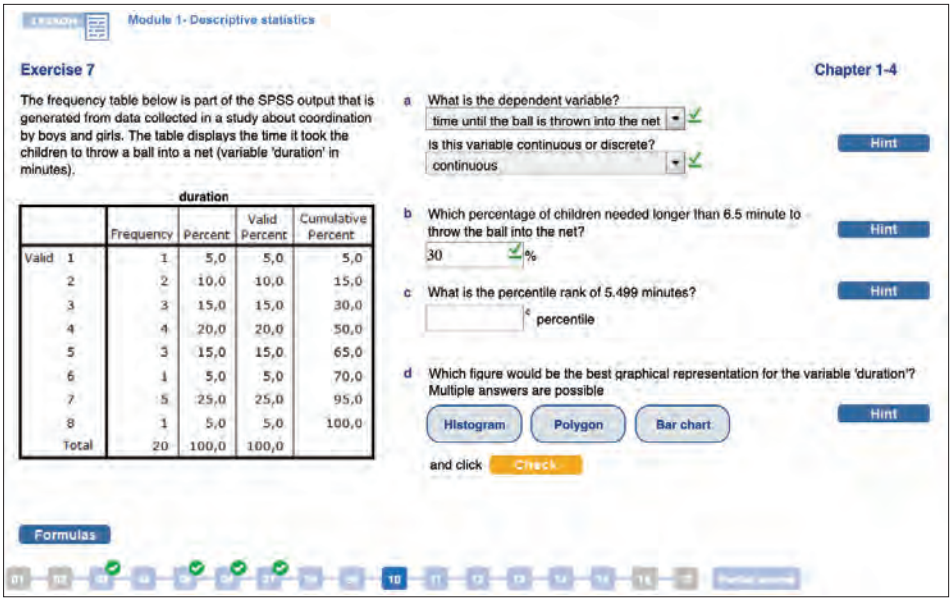

Figure 4.1 Example of a practice task in the first homework set in the DME

4.4.3 Design and implementation of the inspectable student models

The DME was enriched with an inspectable student model for each homework set. Figure 4.2 shows two examples of an inspectable student model for the first homework set. Each student model contained a list of important topics in the homework set, grouped into two or three categories. The number of topics per category varied between two and seven. Most tasks in the homework sets were connected to the topic(s) they were related to. Lists of topics, connections between tasks and topics and the tasks themselves were optimized informed by findings by Tacoma et al. (2018; see Chapter 3 of this thesis), based on the same course in the previous academic year. In particular, this previous study showed that some tasks served a useful function in the homework set (such as introducing a new topic), but were not appropriate for informing student models, and hence should not be connected to any topic. Furthermore, new tasks were added to address topics that had been underrepresented in the previous year, and a number of multiple choice tasks that had been found to offer too few opportunities to learn from (i.e., asking students to select one out of only two options) were redesigned.

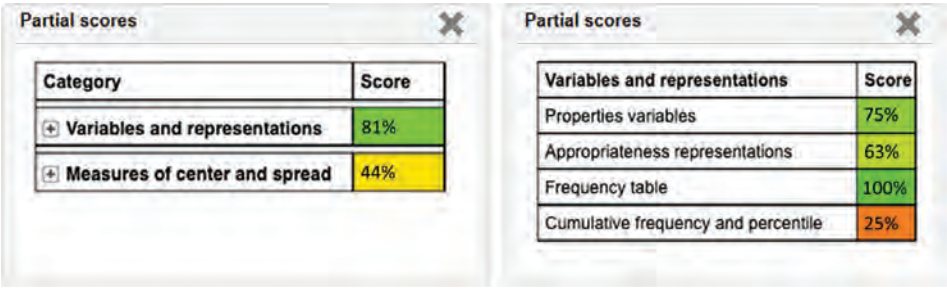

Figure 4.2 Student models for the first homework set when a student has worked on several categories (left) or on one category only (right)

Scores in the student models were calculated based on the student's correct and incorrect attempts on the tasks in the homework set: for each task a task score was calculated as the number of correct attempts on that task (usually 1) divided by the total number of attempts the student made on the task. Topic scores were calculated as the mean task score of all tasks that were connected to the topic and for which the student had made at least one attempt. Category scores were a weighted average of topic scores, weighted by the number of tasks connected to the topic.

Students could access the inspectable student model for a homework set by clicking on the button "Partial scores" (bottom right corner in Figure 4.1). On the final page of each homework set this service was explicitly mentioned to students, with the suggestion to use the student model to select topics for further practice. When students opened the student model, only the categories and category scores were shown (Figure 4.2, left). Students could use the plus-buttons to view the topics in each category and their scores on these topics. Only categories that the student had worked on were shown and if a student had only worked on one category yet, this category was shown folded out immediately (Figure 4.2, right).

On the first page of the extra practice sets, students received instruction that they could either choose to work on all extra practice tasks, or to make a selection based on their inspectable student models. Links to the homework sets were included, so that students could easily access the student models for the different homework sets. In each extra practice set, the first page also contained an overview indicating which extra practice

tasks addressed which topics. This enabled students to select extra practice tasks for topics that needed their attention.

4.4.4 Data collection

Data for this study consisted of log files of the students' work on homework and extra practice sets in the DME, including logs of student model views. Additionally, students' grades for the final exam were collected. The possibility to log student's actions in electronic learning environments provides an opportunity to monitor meticulously what students do with inspectable student models that are provided to them. For each student model view, the DME logged the time of opening and closing the student model, the corresponding homework set, current student model scores for all topics and categories in the student model, and categories that the student opened (if any). After the end of the course period, log files were exported from the DME. Logs from students who did not give consent were deleted and all other logs were rendered anonymous. Exam results were rendered anonymous as well, using the same key to enable connecting them to the students' use of the DME. The final exam lasted two hours and consisted of 30 4-option multiple choice items: 14 about methods and 16 about statistics. Only the students' results on the statistics items were included in this study. For these 16 items, Cronbach's α was .60, which seems an appropriate value for an exam consisting of relatively few items that assess a wide range of topics (e.g., normal distribution, confidence intervals, hypothesis testing) within the domain of statistics (Taber, 2018). An example question is:

> It was investigated whether in the 2010 elections politicians who were active on Twitter received more preference votes than their colleagues who were not active on Twitter. The report mentioned both a *p*-value (.001) as well as the effect size (d = .01). What is the correct conclusion when testing with α = 1%?

Multiple choice options for this item were (a) The result is not significant and the effect is small; (b) The result is significant, but the effect is small; (c) The result is significant and the effect is large; (d) The result is not significant, but the effect is large.

4.4.5 Data analysis

To answer RQ1 on feedback-seeking behavior, the logged information was used to describe how often, how long and in how much detail students inspected their student models. Student model views that lasted shorter than two seconds were omitted from analysis: these views were considered too short for students to be able to interpret the contents of the student model[3]. This concerned 173 student model views, out of a total of 2710. Regarding the detailedness of student model inspection, a Chi-Square proportion test served to examine whether students tended to select the categories with the lowest scores for further inspection, if they opened any categories at all. For all statistical tests, a significance level of .05 was used. To enable an interpretation of the frequency of student model views, working sessions were defined. Following Chen, Breslow and DeBoer (2018) working sessions were defined as series of student actions in the DME in which the time period between two actions was never longer than one hour. Working sessions were mapped over time to determine in what proportion of working sessions students viewed the inspectable student models and to investigate whether students kept inspecting the student models during the course period. To enable further analysis at the level of individual students rather than at the level of student model views, students were assigned to groups based on their feedback-seeking behavior, as will be explained in the results section.

To answer RQ2 on how consulting a student model affects students' decision making on subsequent actions, only student model views after which the student continued working in the DME were included. Three general decisions were possible for students who continued working after viewing a student model, namely work on (1) the homework set for which the student model was viewed ("Homework"), (2) extra practice related to the homework set for which the student model was viewed ("Practice"), or (3) a homework set or extra practice on a different topic than addressed in the student model just viewed ("Other topic"). Students were grouped based on which of the three decisions they made at least once. This resulted in seven groups, namely:

3 Assuming a reading speed of approximately 250 msec per word (H. van Oostendorp, personal communication, April 9, 2019), reading the concepts listed in figure 4.2 (left) would take two seconds, which makes two seconds a reasonable lower bound.

- HPO: Homework-Practice-Other topic, students who made all three decisions at least once;
- HP: Homework-Practice, students who made the decisions Homework and Practice at least once and never made the decision Other topic;
- HO: Homework-Other, students who made the decisions Homework and Other topic at least once and never made the decision Practice;
- PO: Practice-Other topic, students who made the decisions Practice and Other topic at least once and never made the decision Homework;
- H: Homework, students who always continued to work on the homework set after viewing the student model;
- P: Practice, students who always worked on an extra practice set after viewing the student model;
- O: Other, students who always worked on another topic after viewing the student model.

To compare student model scores between different decisions within each group, for each student model view a mean student model score was calculated: the mean of all topic scores currently in the student model. Next, for each decision within each group, the median of the mean student model scores preceding that decision was calculated. Medians and non-parametric tests were used, since the distribution of mean student model scores was negatively skewed. For group HPO, a Friedman's ANOVA and follow-up pairwise Wilcoxon signed rank tests were used to compare median scores for the three decisions. A Bonferroni correction was used to control for the inflated chance of a type I error in multiple comparisons (Shaffer, 1995). For groups HP, HO and PO, the median scores were compared using Wilcoxon signed rank tests.

To examine the relations between feedback-seeking behavior, decision-making behavior and exam results (RQ3), a Chi-square test was used to assess whether feedback-seeking and decision-making behavior were independent. The seven groups for seven possible combinations of decisions were supplemented with an eighth group, "Nothing", for students who never viewed a student model or who never continued working in the DME after viewing a student model. A possible confounding variable

in relations between feedback-seeking behavior, decision-making behavior and exam results was the students' activity in the learning environment. More active students may be more likely to view and use the student models and may also be more likely to perform well on the exam. To assess the influence of this confounding variable, the total time students worked on the tasks in the DME was calculated (including breaks of up to five minutes). Two one-way ANOVAs were carried out to examine the relations between feedback-seeking and decision-making behavior on the one hand, and time on task on the other hand. When the ANOVAs yielded significant results, they were followed up with pairwise comparisons with Bonferroni correction. Finally, a hierarchical multiple linear regression model was set up to assess the relations between students' exam grade as outcome variable and feedback-seeking behavior, decision-making behavior and time on task as predictor variables. Because of possible interaction effects between time on task and feedback-seeking and decision-making behavior, a regression model was deemed more suitable than an ANCOVA, in which interaction effects between grouping variables and covariates are not included.

4.5 Results

4.5.1 Students' feedback-seeking behavior

To gain insight into students' feedback-seeking behavior (RQ1), we summarized all students' views of the inspectable student models. Furthermore, the distribution of student model views over the course period was examined and students were grouped according to their feedback-seeking behavior.

Table 4.1 gives an overview of the students' working sessions in the DME, the number of sessions in which they viewed a student model and the number of times they inspected student models more closely by opening one or more categories. The table reveals that students viewed a student model in 25% of all working sessions in the DME (1874 out of 7410 sessions). There were more student model views (2522) than sessions in which a student model was viewed (1874), which implies that students viewed the same student model more than once or viewed the student models for more than one homework set, in some sessions.

In most sessions, however, students consulted only the student model concerning the homework set they were working on and consulted it just once. Students inspected the student model more closely in 40% of all student model views (997 out of 2522 views). Closer inspection in most cases entailed opening all categories, namely in 713 (72%) of the 997 views. When students did select categories, they tended to select the one or two categories with the lowest score(s): 246 (87%) out of 284 views, $\chi^2(1, N = 284) = 150.88, p < .001$.

Table 4.1 Summary of working sessions and student model views

	Total number	Number of unique students	Mean[1] per unique student (SD)	Median[1]
Working sessions in DME	7410	599	12.4 (6.2)	12
Working sessions with student model view	1874	531	3.5 (2.3)	3
Student model views	2522	531	4.7 (4.1)	4
Views with inspection of categories	997	337	3.0 (2.5)	2

[1] Means and medians were calculated over the students involved.

Regarding duration of student model views, the logs revealed that most student model views were rather short; the median viewing duration was six seconds. Longer views also occurred: 155 views lasted longer than 30 seconds. Logs of student work in the DME also revealed that students mostly viewed student models after they had finished all tasks in the corresponding homework set: this was the case in 2030 out of 2522 views (80%).

The upper part of Figure 4.3 displays the distribution of the students' working sessions over time. Each bar represents the number of working sessions for one day in the course period and the dashed lines indicate the dates of the intermediate and final exams. The figure shows that students kept inspecting the student models throughout the course and that on days before exams both the number of working sessions and the number of student model views increased rapidly. The lower part of Figure 4.3 shows the percentage of sessions in which students inspected a student model, as percentage of the total number of sessions that day, together with a fitted linear regression line. It reveals that the percentage of sessions in which students inspected a student model decreased slightly but significantly

over the course period. Taking the values from the regression line, the percentage dropped by 0.26 percentage points per day, from 34% to 21%.

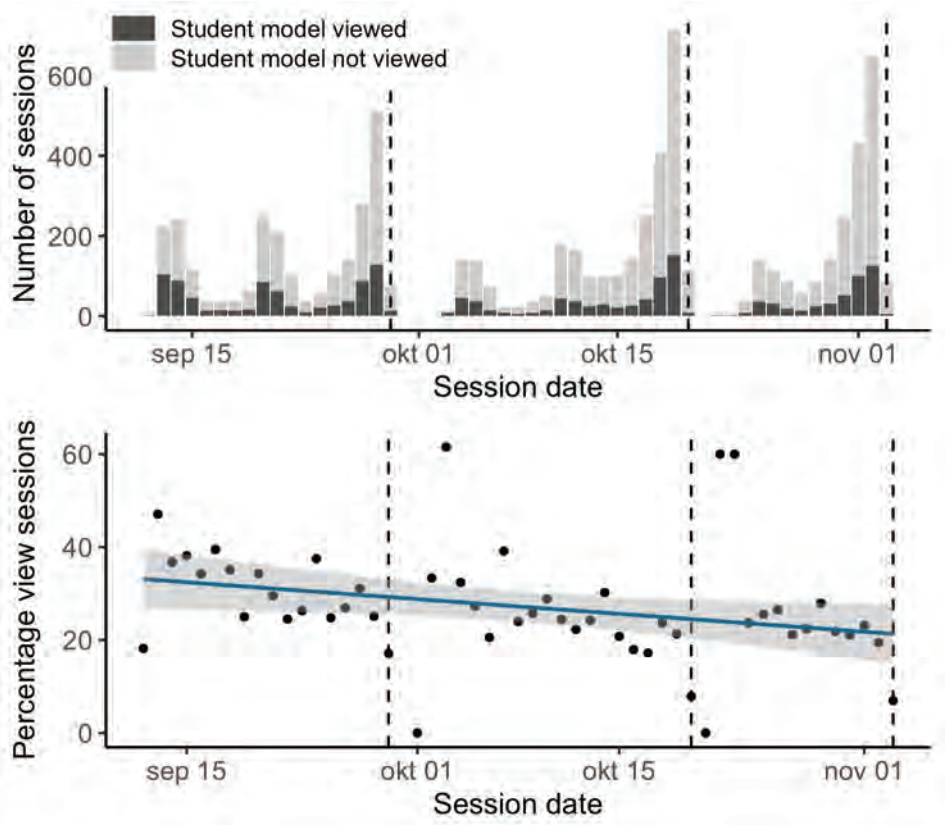

Figure 4.3 Number (top) and percentage (bottom) of working sessions in which students viewed a student model per day over the course period

On average, the 531 students who viewed a student model at least once viewed student models for 2.9 out of five homework sets. Student models for all five homework sets were viewed by 103 students.

Based on the number of homework sets for which students viewed their student model over the course period and the number of times they opened categories for closer inspection, students were assigned to one of three groups: limited, moderate and extensive feedback seekers. The following definitions led to approximately equal group sizes:

- Limited: student viewed student model of at most one homework set;
- Moderate: student viewed student models of at most four homework sets. If four student models were viewed, the student never inspected categories further;
- Extensive: student viewed student models for four or five homework sets. If four student models were viewed, the student inspected categories at least once.

This resulted in a group of 190 limited feedback seekers, a group of 222 moderate feedback seekers and a group of 187 extensive feedback seekers.

4.5.2 Students' decision-making behavior

With respect to the students' decision-making behavior (RQ2), the following results were found. From the 1244 student model views after which the student continued to work in the DME (49% of all 2522 student model views), 587 (47%) were followed by the decision to work on the homework set for which the student model was viewed (Homework). For 281 views (23%), the student's decision was to work on extra practice tasks related to the just viewed student model (Practice), and for the remaining 376 views (30%), the student decided to work on a different topic (Other). Table 4.2 summarizes the allocation of students to the seven decision-combination groups, as well as the medians of the mean student model scores for each of the decisions in each group at the moment of student model consultation.

Table 4.2 Students' decisions after viewing student model and median student model scores preceding the different decisions

Group	*N*	Median score preceding decision Homework	Median score preceding decision Practice	Median score preceding decision Other topic	Value test statistic		*p*
HPO†	62	79.1	83.0	84.8	χ^2 =	17.61	<.001
HP	48	80.8	83.6	-	*T* =	399	.082
HO	62	78.2	-	82.0	*T* =	305	<.001
PO	36	-	79.9	81.0	*T* =	216.5	.108
H	98	78.8	-	-	-		
P	35	-	81.5	-	-		
O	76	-	-	79.5	-		

†H = Homework, P = Practice, O = Other topic

From the table it can be inferred that median scores when students continued with Homework were generally lower than those for the decisions Practice and Other topic. Furthermore, median scores for Practice and Other topic seemed fairly similar. These impressions were confirmed by the tests comparing the median scores in the four groups in which students had made multiple decisions. For students who made all three decisions at least once, group HPO, the Friedman's ANOVA yielded that median scores differed significantly between possible decisions, $\chi^2(2) = 17.61$, $p < .001$. Follow-up pairwise Wilcoxon signed rank tests yielded that scores when students decided to work on Homework were significantly lower than scores when students (1) decided to work on Practice ($T = 385$, $p < .001$, $r = -.35$) and (2) decided to work on Other topic ($T = 402$, $p < .001$, $r = -.30$). The scores did not differ significantly between the decisions Practice and Other topic ($T = 873.5$, $p = .608$, $r = -.05$). For group HO, the Wilcoxon signed rank test yielded a significant difference ($p < .001$, $r = .37$), indicating that students in this group chose to work on the homework set for lower scores and chose to work on another topic for higher scores. For groups HP and PO, scores did not differ significantly for the two decisions. Altogether, students tended to continue to work on the homework set for lower student model scores and started working on extra practice or another topic for higher student model scores.

4.5.3 Relation between feedback seeking, decision making and exam results

Before looking at exam results (RQ3), we first examined the relations between feedback-seeking behavior, decision-making behavior and time on task. Table 4.3 characterizes students by their feedback-seeking and by their decision-making behavior. A Chi-square test yielded that the characterizations were strongly related: $\chi^2(14, N = 599) = 323.86$, $p < .001$, Cramer's $V = .52$. Table 4.3 reveals that most limited feedback seekers, if they viewed a student model at all, indeed made just one single decision after viewing, while many extensive feedback seekers made different decisions on different occasions of viewing the student model. Hence, both characterizations seem to describe how intensively students used the student models. Although a strong relationship was found, Table 4.3 also reveals that students' decision-making behavior varied considerably among students exhibiting similar feedback-seeking behavior,

especially for moderate feedback seekers. The final row and column of Table 4.3 summarize student activity, as measured by time (in hours) that students worked on the tasks in the DME. Time on task was found to be significantly different for students with different feedback-seeking behaviors ($F(2, 596) = 74.88$, $p < .001$, $\eta^2 = .20$). Post-hoc comparisons revealed that all differences between the three groups were significant (all p-values smaller than .001). As expected, extensive feedback seekers spent most time on the tasks and limited feedback seekers the least. Time on task also differed significantly between groups of decision-making combinations, ($F(7, 591) = 16.26$, $p < .001$, $\eta^2 = .16$). Post-hoc comparisons revealed that students in the Nothing group worked significantly shorter than students in all other groups, that students in group O worked significantly shorter than students in groups HP and HPO, and that students in groups H and HO worked significantly shorter than students in group HPO.

Table 4.3 Student characterization by feedback-seeking and decision-making behavior and time on task (in hours) for all groups

Decision making / Feedback seeking	HPO	HP	HO	PO	H	P	O	Nothing	Total	Time on task (SD)
Limited	0	0	3	0	23	12	22	130	190	5.3 (3.9)
Moderate	9	19	26	18	44	14	45	47	222	8.0 (3.4)
Extensive	53	29	33	18	31	9	9	5	187	9.6 (3.1)
Total	62	48	62	36	98	35	76	182	599	7.6 (3.9)
Time on task (SD)	10.7 (3.8)	9.1 (3.3)	8.2 (3.2)	8.9 (3.2)	7.9 (3.4)	8.3 (3.6)	6.7 (3.5)	5.8 (3.9)	7.6 (3.9)	

Table 4.4 summarizes the parameter estimates and model fits of the hierarchical regression model predicting exam grade from time on task, feedback-seeking behavior, decision-making behavior and interactions. The base model shows that time on task was a significant predictor and explained 13% of the variability in exam grade. Adding feedback-seeking behavior resulted in a significantly better model ($F(2, 561) = 17.82$, $p < .001$), explaining an additional 5% of the variance. Time on task was a confounding variable in this relationship between feedback-seeking

behavior and exam grade, as indicated by the relations found between time on task and both feedback-seeking behavior and exam grades.

Table 4.4 Parameter estimates and model fits for the linear regression model predicting exam grade from time on task (in hours), feedback-seeking behavior, decision-making behavior, and the interaction between time on task and feedback-seeking behavior

	Base model with time on task	+ Feedback seeking	+ Decision making	+ Time on task × Feedback seeking
Intercept	8.90***	8.65***	8.77***	8.22***
Time on task	0.25***	0.17***	0.18***	0.28***
Feedback seeking: Extensive		1.65***	2.23***	4.00***
Feedback seeking: Moderate		0.79**	1.14***	1.90***
Decision making: HPO			-1.14*	-1.00*
Decision making: HP			-0.60	-0.58
Decision making: HO			-0.66	-0.70
Decision making: PO			-1.33**	-1.32**
Decision making: H			-0.21	-0.29
Decision making: P			0.20	0.20
Decision making: O			-0.96**	-0.95**
Time on task × Extensive				-0.23**
Time on task × Moderate				-0.13
Adjusted R^2	0.129	.176	.192	.204
Adjusted R^2 change		.047	.015	.012
F change		17.82***	2.56*	5.17**

*p < .05; **p < .01; ***p < .001.

Still, the positive model parameters for moderate and, especially, extensive feedback seekers (compared to the reference group limited feedback seekers) suggest that, regardless of time spent on tasks, more extensive feedback seeking resulted in higher exam grades. Adding decision-making behavior to the model added another 1.5% to the amount of explained variance and significantly improved the model ($F(7, 554) = 2.56$, $p = .013$), but to a lesser extent than adding feedback-seeking behavior did. The parameter values for the different decision-making groups (compared to the reference group Nothing) are difficult to interpret, given the interplay between feedback-seeking behavior and decision-making behavior that is illustrated by Table 4.3. Finally, interactions between the predictor variables were added to the model. Only the interaction between feedback-seeking behavior and time on task significantly improved the model ($F(2, 552)$ =

5.17, p = .006) and explained an additional 1.2% of the variance in exam grade. Figure 4.4 illustrates this interaction effect. It reveals that for moderate and limited feedback seekers the time worked in the DME was strongly related with exam grade. For extensive feedback seekers, however, there seemed to be no relation between time on task and exam grade.

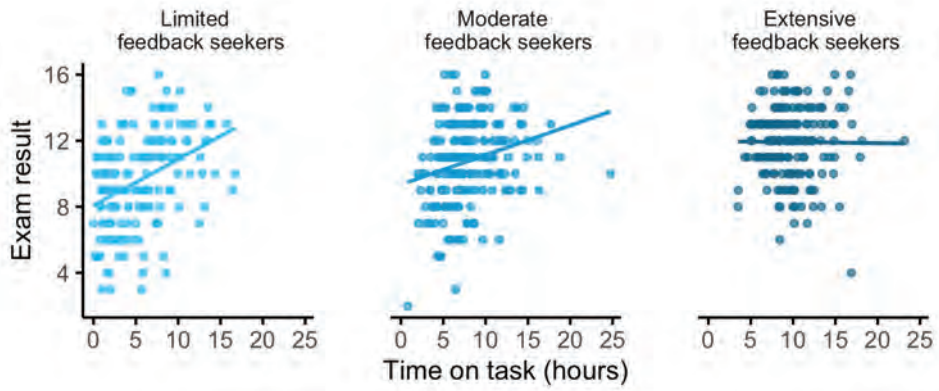

Figure 4.4 Relation between exam result and time on task for limited, moderate and extensive feedback seekers

4.6 Discussion

In this study, we investigated whether and how first-year university students used inspectable student models in a statistics course, and whether students seemed to benefit from these student models. We examined the students' feedback-seeking behavior (RQ1), decision-making behavior (RQ2), and the interplay between student behavior and exam grades (RQ3).

Concerning RQ1, a wide variety was found in students' feedback-seeking behavior, or, more specifically, in frequency, timing, duration and amount of detail of student model views. This diversity seems to reflect a variety in self-motives underlying feedback-seeking behavior (Anseel et al., 2007), both among students as well as within students over time. For example, a student model view of a few seconds that takes place before the student has fully completed a homework set may be driven by a self-verification motive (i.e., quickly verify what one's weaker and stronger topics currently are). A long view that takes place after completing the homework set is more likely to be driven by a self-improvement motive

(i.e., consider what to do next to improve one's mastery). Regardless of their exact motives, for most students the perceived values of the inspectable student models outweigh the costs, and hence, in line with earlier research, students seem to appreciate the availability of inspectable student models (Bull, 2004; Mitrovic & Martin, 2007).

Concerning RQ2, student appreciation is an important factor for inspectable student models to affect learning, but appreciation alone is not enough. Students also need to actively process the provided feedback and use it to decide on subsequent study steps (Timmers et al., 2013). Students in this study made a wide variety of decisions, which is in line with earlier findings (Bull et al., 2008). It suggests that inspectable student models fit into many different learning paths and, thus, allow students to take responsibility for their own learning. Across this variety of learning paths, students seemed inclined to improve their homework sets when student model scores were low, and to work on extra practice tasks or other topics when student model scores were higher. Hence, lower scores may encourage students to devote more effort to the homework sets than they would do without student models available, which is a valuable – hidden – effect of feedback (Hattie & Yates, 2014). This effect implies that inspectable student models may indeed support students in reducing the academic procrastination that is common in many introductory statistics courses (Onwuegbuzie, 2004).

The aim of RQ3 was to evaluate how students' use of the inspectable student models, i.e., their feedback-seeking and decision-making behavior, related to performance on the final exam, as an indication of how this operationalization of formative assessment can contribute to student performance. While student activity, as measured by time on task, was found to be an important predictor of exam result, the frequency of student model viewing explained significant additional variance in students' exam grades. So did, to a lesser extent, the amount of variety in decisions students made after viewing the student model. These findings suggest that frequently inspecting student models and using them to inform subsequent study steps seems a fruitful learning strategy. Furthermore, these findings support the assumption that feedback-seeking behavior and decision-making behavior are influenced by students' individual self-

motives (Anseel, Lievens, & Levy, 2007), and not only by the amount of time they spend in the learning environment. Especially in the group of frequent feedback-seekers, no relation was found between time on task and exam grade, suggesting that other factors than activity determined how efficiently and effectively these students could use inspectable student models in their learning strategies.

While these findings imply that inspectable student models can be a valuable enrichment for electronic learning environments, especially in university statistics courses, this study has some limitations. First, due to the explorative nature of this study no causal inferences could be made about the influence of students' feedback-seeking and decision-making behavior on exam grades. It is, for example, likely that students differ in self-regulated learning abilities and that stronger self-regulated learners have strong self-improvement motives, which results in a high frequency of viewing student models. At the same time, these stronger self-regulated learners are also likely to perform well on an exam. Future research, with a randomized control design, is needed to establish whether there is a causal relation between availability of inspectable student models and exam results.

A second limitation relates to the main source of data for this study: logs from student work. While they provide valuable information and have the large advantage that collecting them is minimally invasive for students, for this study they also have a drawback: it is difficult, if not impossible, to infer students' intentions or self-motives from log files. We do not know whether long student model views indicate intensive engagement with a student model, or off-task behavior. Likewise, we assumed, but cannot prove, that students' decisions were influenced by the contents of the student models. Meanwhile, the ways in which students could benefit from inspectable student models might vary along with their varying self-motives. For example, students with weaker self-improvement motives might be expected to benefit relatively much from inspectable student models, because of the low costs of seeking feedback from them (Timmers et al., 2013) and the support they can give for selecting appropriate subsequent tasks (Corbalan, Kester, & Van Merriënboer, 2006). Future research that more directly addresses the students' self-motives and self-regulated

learning capabilities, for example through questionnaires or interviews with focus groups, could provide more insight into how feedback from inspectable student models can best be tailored to the students' individual needs and preferences.

A final limitation is that the studied decision-making behavior concerned quite general decisions: continue working on the same homework set, go to extra practice or move to another topic. Inspectable student models have the potential to inform more specific decisions about topics for which students need to exert their effort and thinking (Hattie & Yates, 2014). In this study, however, due to the design of homework and extra practice sets, only a few such topic-specific decisions could be identified. In the homework sets, connections between topics and tasks were not made explicit for students and in the extra practice sets, topic descriptions for the tasks did not align completely with the terminology used in the inspectable student models. This may have hindered students in responding to the feedback according to their intentions (Kinicki et al., 2004). Consistent and explicit connections between tasks and topics could better support students in making deliberate decisions on topics to work on (Brusilovsky et al., 2009; Kicken, Brand-Gruwel, & van Merriënboer, 2008). This should receive careful attention in both further research and implementation of inspectable student models in practice, to realize their full potential for supporting more efficient and effective learning processes.

As concluding remarks, we note that the current research has revealed that students exhibit a wide variety in feedback-seeking and decision-making behavior when inspectable student models are available. Hence, this operationalization of the tailored feedback element that is essential for a cyclical formative assessment approach (Gikandi et al., 2011) seems to fit well within many learning paths. This allows students to take the responsibility for their learning that is required in university education (Krause & Coates, 2008). Furthermore, students' decision making appeared to be, at least partly, informed by the provided feedback, suggesting that inspectable student models also facilitate a second essential element of the formative assessment cycle: deciding on subsequent actions to enhance learning. Regarding performance, this study supports the claim that feedback-seeking behavior positively relates

to performance, as well as the hypothesis that performance is enhanced by a high variety in decisions based on inspectable student models. For practice, this suggests that inspectable student models can indeed be a valuable enrichment of electronic learning environments, even in cases where student models do not inform task selection directly. While our implementation required students to actively seek for feedback by clicking a button, many other systems automatically show students their student models, which reduces the cost of seeking feedback. Whether this would result in more students engaging with the feedback and using it to decide on subsequent study steps than in our implementation is uncertain, though, because the students' self-motives also play a crucial role in engaging with feedback (Anseel, Lievens, & Levy, 2007). This could be an interesting venue for further research.

Finally, to answer the title question, a note on implementation effort is in place. Once the infrastructure within the electronic learning environment is set up, a simple inspectable student model implementation – like the one used in this study – only requires a list of concepts in the domain and connections between the tasks and these topics. Given that students value the inspectable student models, that students seem to practice more when such models are available, and that performance seems positively related to both feedback-seeking behavior and variety of decisions, our answer to the title question would be positive: implementing inspectable student models does seem to be worth the relatively small effort, and, while this study was conducted in the frame of a statistics course, we expect this to be worthwhile for other domains as well.

Acknowledgements

Design and implementation of the educational materials used in this study was supported by Utrecht University's Education Incentive Fund. We are grateful to Jeltje Wassenberg-Severijnen, teacher of the course, and Peter Boon, developer of the Digital Mathematics Environment, for the fruitful collaboration.

CHAPTER 5 Combined inner and outer loop feedback in an intelligent tutoring system for statistics in higher education

Tacoma, S. G., Drijvers, P. H. M., & Jeuring, J. T. (2020). Combined inner and outer loop feedback in an intelligent tutoring system for statistics in higher education. *Journal of Computer Assisted Learning*, 1–14. doi: /10.1111/jcal.12491

Author contributions: Sietske Tacoma: Conceptualization, Formal analysis, Writing - original draft. Paul Drijvers: Writing - review & editing, Supervision. Johan Jeuring: Writing - review & editing, Supervision.

Abstract Intelligent Tutoring Systems (ITSs) can provide inner loop feedback about steps within tasks, and outer loop feedback about performance on multiple tasks. While research typically addresses these feedback types separately, many ITSs offer them simultaneously. This study evaluates the effects of providing combined inner and outer loop feedback on social sciences students' learning process and performance in a first-year university statistics course. In a 2x2 factorial design (inner loop vs. no inner loop and outer loop vs. no outer loop feedback) with 521 participants, the effects of both feedback types and their combination were assessed through multiple linear regression models. Results showed mixed effects, depending on students' prior knowledge and experience, and no overall effects on course performance. Students tended to use outer loop feedback less when also receiving inner loop feedback. We therefore recommend introducing feedback types one by one and offering them for substantial periods of time.

Keywords Domain reasoner ◆ Feedback ◆ Intelligent Tutoring Systems ◆ Inspectable student models ◆ Statistics education

5.1 Introduction

Over the past decades, a huge number of computer-based learning environments have been developed that facilitate learning of many topics at all educational levels. One of their largest promises for enhancing learning is the provision of individualized and timely feedback on student work (Pardo, 2018; VanLehn, 2011). Fulfilling this promise is not straightforward, though, because there are many design choices to make when implementing feedback, regarding specificity, timing, type and complexity of information provided, and visual presentation (Shute, 2008). To better understand the consequences of such design choices, many theories have been developed about whether and how feedback contributes to student learning and motivation, both in general (Hattie & Timperley, 2007; Kluger & DeNisi, 1996; Shute, 2008) as well as specifically for computer-based settings (Pardo, 2018; Van der Kleij, Feskens, & Eggen, 2015). Attempts have been made to capture feedback in all its appearances in one model, such as Kluger and DeNisi's (1996) feedback intervention theory and Pardo's (2018) model for data-supported feedback. These models have in common that there is a large variety in feedback effects – positive and negative – on student learning and motivation across feedback operationalizations. Factors that influence feedback effects include not only feedback design, but also the instructional context and the learners involved (Narciss & Huth, 2004). The overall tendency in the research literature is that feedback, both in general and in computer-based learning environments, may contribute to learning (Van der Kleij et al., 2015). It is, therefore, not surprising that feedback provided by computer-based learning environments has become widespread in education (Gikandi, Morrow, & Davis, 2011).

Computer-based learning environments that provide sophisticated individualized feedback are called Intelligent Tutoring Systems (ITSs). In ITSs, two general feedback types can be distinguished: inner loop feedback on steps within tasks, and outer loop feedback over complete tasks or multiple tasks at once (Santos & Jorge, 2013; VanLehn, 2006). Inner loop feedback typically provides information about the correctness of a (partial) solution, combined with guidance on how to resolve mistakes and how to proceed in solving the current task. According to VanLehn (2006), availability of inner loop feedback classifies a computer-based learning

environment as an ITS. Outer loop feedback concerns the student's current knowledge state regarding the domain and the selection of appropriate subsequent tasks or study activities. For both types, positive effects on student learning have been reported (see, for example, VanLehn (2011) for inner loop feedback and Bull & Kay (2016) for outer loop feedback). As a consequence, both types of feedback have been implemented in computer-based learning environments that are used in educational practice today.

Implementing research findings in educational practice, however, is not straightforward (Vanderlinde & van Braak, 2010). To enable drawing causal inferences, the research community puts great emphasis on randomized experiments and controlling context variables (Farley-Ripple, May, Karpyn, Tilley, & McDonough, 2018). Consequently, many studies focus on only one of the feedback types (inner loop or outer loop feedback) and only on specific aspects (Narciss et al., 2014). In contrast, teachers and educational designers are inclined to use multiple promising approaches for delivering rich, inspiring education. As a consequence, many ITSs provide inner and outer loop feedback simultaneously, thus offering guidance both at the level of steps within tasks as well as at the level of task selection. Ideally, this results in optimal guidance to students during their engagement with the ITS; it might, however, also lead to an overwhelming amount of feedback information for students. To our knowledge, this question of whether combining inner and outer loop feedback influences their effects has not been studied yet.

The aim of this study is, therefore, to assess the effects of offering inner and outer loop feedback concurrently. To this end, an ITS providing both inner and outer loop feedback was offered to students in social sciences bachelor programs, in a large enrolment first-year statistics course. The topic of statistics was deemed suitable for providing ITS feedback, because statistics courses are challenging for many students (Tishkovskaya & Lancaster, 2012). Students struggle to understand the large number of complex concepts involved, such as sampling variability, probability distributions and *p*-values (Castro Sotos, Vanhoof, Van den Noortgate, & Onghena, 2007). A particularly important and challenging topic is the method of null hypothesis significance testing (further referred to as "hypothesis testing"), which does not only require understanding

of these complex concepts, but also an ability to follow a complex line of reasoning involving uncertainty (Falk & Greenbaum, 1995; Garfield et al., 2008). Besides this challenging character of the topic, a second reason for implementing and evaluating ITS feedback in such a course was the large group size, which makes providing individual guidance and feedback difficult for the teachers involved. The guiding research question for this evaluation was: What effects does providing both inner and outer loop feedback on online homework have on students' learning process and course performance in a university statistics course?

5.2 What is feedback?

Let us first take a closer look at how existing feedback theories postulate the effects of feedback. Pardo (2018), after reviewing feedback literature, defined feedback as follows:

> A process to positively influence how students engage with their work in a learning experience so that they can improve its overall quality with respect to an appropriate reference and increase their self-evaluative capacity. (Pardo, 2018, p. 433)

Four elements of this definition are worth highlighting when considering feedback effects. First, the phrases "positively influence" and "improve its overall quality" emphasize the general aim of enhancing the learning process. Second, and more important, the word "process" signifies that feedback entails much more than instantaneous information delivery to a student. This also becomes clear from Pardo's model for data-supported feedback, depicted in Figure 5.1: Information & Delivery comprises only one of ten components in the feedback process. It is preceded by a phase of collecting evidence about the student's learning process (nodes 5 and 7), which is influenced by factors including the student's knowledge, goals and strategies (nodes 2, 3, 4 and 6). This evidence allows for tailoring the feedback information to the student's individual needs (Gikandi et al., 2011). The collected evidence is then analyzed (nodes 8 and 9), before the feedback information is delivered to the student (node 10). Next, another important phase of the feedback process takes place: the student assimilates the information, which may result in changes in the student's

knowledge, beliefs, attitudes, goals and/or strategies and tactics (Pardo, 2018; Timmers, Braber-van den Broek, & Van den Berg, 2013).

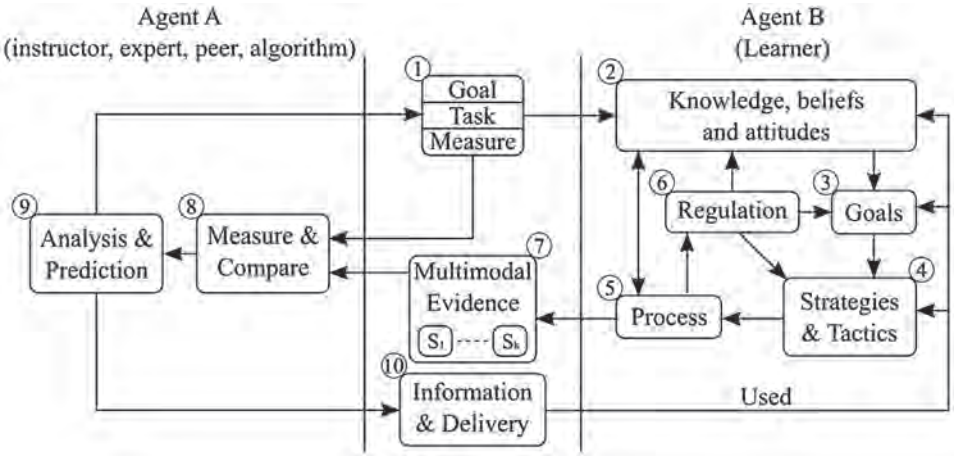

Figure 5.1 Pardo's (2018) model for data-supported feedback. Reprinted with permission of the publisher (Taylor & Francis Ltd*, http://www.tandfonline.com)*

The third element of Pardo's feedback definition that is worth highlighting is the phrase "with respect to an appropriate reference". Feedback information can only impact learning if the student knows which goals or standards to strive for. This is referred to as the feedback-standard gap (Kluger & DeNisi, 1996): feedback signals whether there is a gap between the student's current state and a desired state. If the student's current state is not up to the level of the desired state, the feedback should evoke a desire to close or reduce this gap. Hence, effective feedback should help students clarify what good performance is, in terms of desired goals or standards, and provide opportunities to close the gap between current and desired performance (Nicol & Macfarlane-Dick, 2006). Closing the gap should, however, remain the responsibility of the student and not of the feedback: feedback should not be so prescriptive that it diminishes the need for students to think for themselves (Evans, 2013).

The fourth and final phrase from Pardo's feedback definition that we highlight is "increase their self-evaluative capacity." A characteristic of higher education, where our study is situated, is that students are expected to work independently on learning activities, especially in large enrolment courses such as first-year statistics courses. Because of this characteristic

and of the goal of higher education to prepare students for professional careers, feedback in higher education should facilitate the development of reflection on learning (Nicol & Macfarlane-Dick, 2006).

To summarize, from Pardo's feedback model and related literature we take that (1) feedback is a process, including phases of evidence collection and analysis, information delivery, and students' use of the feedback; (2) feedback should address the gap between current and desired performance and offer opportunities to close this feedback-standard gap; and (3) feedback should facilitate students' reflection on learning. In the following two sections, we outline how these feedback principles informed the two feedback implementations in this study: a domain reasoner for hypothesis testing as inner loop feedback and inspectable student models as outer loop feedback. We explicitly address these principles in the design description, which makes this study not only an evaluation of offering the combination of the two feedback types, but also of the usefulness of these guiding theory-based principles for designing feedback.

5.3 Inner loop feedback: domain reasoners

Inner loop feedback is feedback concerning intermediate steps in the solution to a task (VanLehn, 2006). It provides information about the correctness of the student's solution to the task so far. The evidence needed to generate this information consists of the student's steps in solving tasks. To analyze this evidence, the ITS needs to have domain knowledge: knowledge of the rules required to solve tasks in the domain at stake. In this study, the domain at stake is the topic of hypothesis testing. The component of the ITS that deals with this domain knowledge is referred to as *domain reasoner* (Goguadze, 2011). Two prevailing paradigms for the design of domain reasoners are model-tracing, in which the domain reasoner checks whether the student's solution so far follows the rules of a model solution (Anderson, Corbett, Koedinger, & Pelletier, 1995), and constraint-based modeling, in which the domain reasoner evaluates whether the student's solution violates one or more predefined constraints (Mitrovic, Martin, & Suraweera, 2007). For the topic of hypothesis testing, domain reasoners have been designed following each of the paradigms (Kodaganallur, Weitz,

& Rosenthal, 2005) and even combining both (Tacoma, Heeren, Jeuring, & Drijvers, 2019; see Chapter 2 of this thesis).

Information delivery typically happens after each intermediate step a student takes in solving a task. If the student's solution path is correct but not complete yet, the feedback generally acknowledges this and encourages the student to continue solving the task. This encouragement implicitly signifies the incompleteness of the solution, which is a gap between current and desired performance. For incorrect solution paths, the feedback usually addresses this gap more directly, by flagging the error and optionally providing information about the error and ways to repair it (Zakharov, Mitrovic, & Ohlsson, 2005). Van der Kleij and colleagues (2015) found that providing elaborate information is generally more effective for learning than just providing information about the correctness of the response, but noted that the type of elaborate information provided in different studies varied widely. According to Evans (2013) and Zakharov, Mitrovic and Ohlsson (2005), such information should enable students to gain insights about underlying concepts, without explicitly dictating what the next step or these insights should be. Ideally, students would use such information to reflect on their current understanding of the concepts involved and to improve their solution to the task (VanLehn, 2011). In his review, VanLehn found that ITSs providing feedback at the level of intermediate steps are as effective as human tutors, with a mean effect size of $d = 0.76$ compared to no tutoring. Other research findings were less optimistic: for example, Narciss and colleagues (2014) found that students more often gave up on tasks when feedback elaborately addressed key concepts. In the domain reasoner feedback designed for the current study, therefore, we strived for providing enough, but not too much information, by briefly mentioning key concepts in hypothesis testing and relevant relations between them.

5.4 Outer loop feedback: inspectable student models

According to VanLehn (2006), the main concern of the outer feedback loop of an ITS is the sequencing of tasks. The most rigid way of sequencing tasks is offering them in a predefined, fixed order. In this case, no actual feedback is involved. Alternatively, the ITS could analyze evidence from

prior work to estimate the student's current knowledge state and use this information to select appropriate tasks. This estimation of student knowledge is called a *student model* (Brusilovsky & Millán, 2007). Such informed task selection could be interpreted as feedback, albeit rather implicitly: after analyzing evidence, the ITS provides the information: "I think that this problem is appropriate for you right now." Thus, it provides opportunities to close the feedback-standard gap, but without explicitly indicating what this gap is.

Category	Score
⊖ **Two-way analysis of variance**	58%
Recognizing main effects and interaction effects	45%
Finding degrees of freedom effects	73%
Finding degrees of freedom error	57%
Read and fill table analysis of variance	69%
Effect size	39%
⊕ **General procedure**	84%
State hypotheses	73%
Find rejection region	100%
Calculate value test statistic	78%
Use critical value and p-value	88%
Reject null hypothesis or not	81%

Figure 5.2 Inspectable student model evaluated in this study

A third alternative for task sequencing is to offer a set of tasks and let the student decide. In this case, the student model that informed task selection in the previous alternative can be used as explicit feedback to aid students in the task selection process. When the student model is visualized and delivered to the student, it is called an *inspectable student model* (Bull & Kay, 2016). In its most popular form, the student model represents the student's knowledge state as a subset of expert-level knowledge of the domain (Brusilovsky & Millán, 2007). It allows the student to quickly identify

a feedback-standard gap as those *knowledge components* – elements of knowledge in the domain under study – for which the current knowledge state is below the expert-level. An example of an inspectable student model that was used in this study is shown in Figure 5.2. Especially the low percentages for the elements "Recognizing main effects and interaction effects" and "Effect size" indicate a feedback-standard gap. Based on this information, a student could decide to work on more tasks addressing these topics. Besides this function of informing task selection, the information can also influence the student's knowledge, beliefs and attitudes – for example, when the student believes to understand an element well, but finds a low percentage in the student model. In this way, inspectable student models may promote reflection (Bull & Kay, 2016; Long & Aleven, 2011). Finally, inspectable student models could provide opportunities to close the feedback-standard gap, for example by suggesting appropriate follow-up tasks (e.g., Sosnovsky & Brusilovsky, 2015).

Although the ideas behind inspectable student models have become common (Bull & Kay, 2016), questions such as how to collect evidence and which information to provide to which students still receive considerable research attention. Sosnovsky and Brusilovsky (2015) found that using broader topics for collecting evidence, rather than very detailed ones, results in less accurate models, but still provides a basis for successful personalization. For educational practice this is a promising result, given time constraints for teachers to develop detailed student models. Regarding information to provide, Al-Shanfari, Epp and Baber (2017) found that students who were presented with more details and information about uncertainty in their student models, viewed their student models more often and worked on more tasks. Finally, several studies suggest that low-achieving students benefit more from availability of student models than high-achieving students, especially when student models include a social component (Brusilovsky, Somyürek, Guerra, Hosseini, & Zadorozhny, 2015) or when they are combined with support for task selection (Mitrovic & Martin, 2007). In the current study, we focus on the effects of offering student models without such additional support, this being the most common implementation in educational practice.

5.5 Materials and methods

5.5.1 Participants

The study was carried out within a first-year statistics course at a Dutch research university. This course was mandatory for all students who enrolled in any social science bachelor program. Participants in this study were 521 out of the cohort of 1294 students who met all inclusion criteria as described in section 5.5.4. The majority of students, 82%, were female, and their ages varied between 17 and 28 years (M = 19 years and 11 months, SD = 1 year and 5 months).

Students took the eight-week Methods and Statistics course after an Introductory Methods and Statistics course. The course was offered in three variants, for three different groups of students:

- Educational sciences (EDU, 94 students included out of the 186 enrolled);
- General social sciences, Cultural anthropology and Sociology (GCS, 150 students included out of the 390 enrolled);
- Psychology (PSY, 277 students included out of the 718 enrolled).

5.5.2 Educational setting

The three course variants covered the same content. In five weeks of the course, students received online homework sets on statistical topics: correlation, regression, one-way ANOVA, two-way ANOVA, and Chi-square tests. These homework sets contained five to ten tasks, each with one to eight sub-tasks. The homework sets were delivered through the Digital Mathematics Environment (DME), a computer-based learning environment developed by the Freudenthal Institute (Drijvers, Boon, Doorman, Bokhove, & Tacoma, 2013). The DME is a widespread learning environment for mathematics education in the Netherlands, used by approximately 80 secondary schools and higher education institutions. Participants already had experience with the DME from their introductory Methods and Statistics course. The five homework sets had been used in the previous academic year and were only slightly adjusted. One of the DME's standard features, common in computer-based learning environments, is immediate verification feedback. For all tasks, students received feedback on the correctness of their solution immediately after answering. For incorrect

solutions, no information was given about the correct solution, but students could attempt answering tasks until their solution was correct.

The course concluded with an exam that lasted for two hours and consisted of 30 four-option multiple choice items: 15 on methods and 15 on statistics. In this study, only the students' results on the statistics items were used. The exams for the GCS and PSY groups were identical. The EDU exam was offered later in the academic year and hence was different. Typical exam items, for example, provided partial SPSS-output of a statistical test for a given context, and students had to choose the correct null hypothesis, or the correct estimation of the *p*-value, given the test value and degrees of freedom.

5.5.3 Feedback design

In this study, two additional feedback types were implemented in the DME and evaluated: inner loop feedback on steps within hypothesis-testing tasks by means of a domain reasoner and outer loop feedback in the form of inspectable student models. By adding these two feedback types, the DME became an ITS as defined by VanLehn (2006). To facilitate evaluating the effects of both feedback types separately and in combination, four versions of the homework sets were designed: providing both domain reasoner feedback and student model feedback, domain reasoner feedback only, student model feedback only and none of the feedback types, respectively. As mentioned in section 5.5.2, all four versions provided immediate verification feedback on the correctness of responses to tasks.

Design of inner loop feedback

Because hypothesis testing plays an important role in statistics, the homework sets contained many tasks that involved hypothesis testing. Most of these tasks were quite structured, paving the way for students to smoothly solve them and become familiar with the many abstract concepts that play a role in hypothesis testing. As argued before, however, offering only such highly structured tasks may reduce the need for students to think for themselves (Evans, 2013). Therefore, in three of the five homework sets one highly structured hypothesis-testing task was replaced by an open-ended version, in which students could stepwise set up a hypothesis test for a given research context. To do so, they could select

general steps ("State hypotheses", "Calculate the test statistic", etc.) from a drop-down menu. After selecting a step, the student could complete it with specific details for the current task. In the versions of the homework sets that provided domain reasoner feedback, this feedback was provided on each step in these hypothesis-testing tasks. The other versions of the homework sets only provided verification feedback on each step, without further elaboration.

The evidence that the domain reasoner collected consisted of all steps the student had taken in the hypothesis-testing procedure so far. In the analysis phase, the domain reasoner checked whether the student was on a correct solution path and, if not, diagnosed which parts of the solution were inconsistent, incomplete or incorrect. For full details of the domain reasoner design, we refer to Tacoma et al. (2019; see Chapter 2 of this thesis). Figure 5.3 shows an example of information that was provided by the domain reasoner. For comparison, Figure 5.4 shows the feedback that was given for the same partial solution in the versions of the homework sets that did not provide domain reasoner feedback. The domain reasoner feedback was intended to facilitate improvement of the current solution, hence providing opportunities to reduce the feedback-standard gap. Also, we tried to provide just enough information, allowing the students to think of some of the required steps by themselves and hence to critically reflect on and expand their current knowledge of hypothesis testing. In other words, the feedback pinpointed inconsistencies in student solutions and mentioned key concepts related to these inconsistencies, but did not explicitly describe how these inconsistencies could be resolved. Besides checking the correctness of partial solutions, the domain reasoner could also provide hints for appropriate next steps. Students could ask for a hint at any time while solving the tasks. Hints were formulated in general terms ("State hypotheses" or "Calculate the value of the test statistic"), so that the students still needed to fill in the details for the current situation.

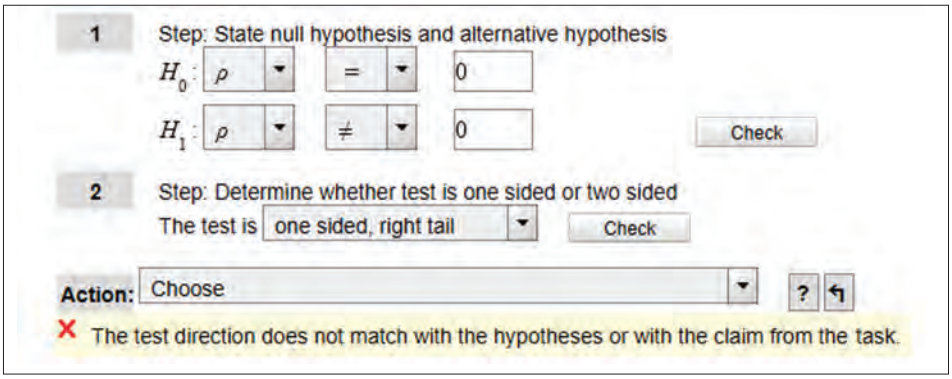

Figure 5.3 Domain reasoner feedback for an incorrect test direction

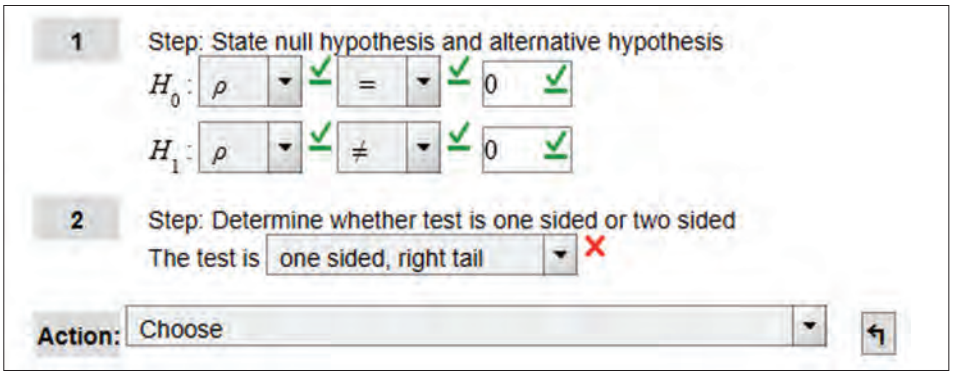

Figure 5.4 Verification feedback for an incorrect test direction

Before the study, the domain reasoner had been implemented and evaluated in a course for psychology students (Tacoma et al., 2019; see Chapter 2 of this thesis). Consequently, students in the PSY course variant already had experience with the domain reasoner feedback, while students in the EDU and GCS variants did not. For this study, improvements were made based on the previous evaluation and the domain reasoner software was extended to support tests for correlation, ANOVA and Chi-Square.

Design of outer loop feedback

Each homework set had a student model, containing nine to fourteen knowledge components divided into two or three categories. In the student model conditions, students could access the models through a button at all times, and as many times they liked. When they opened the student model, all their attempts at answering tasks in the homework set so far

were used to generate the student model. To encourage students to use the student models, the final page of the homework sets explicitly mentioned the student models and suggested that students could use them to decide which topics they still needed to work on.

Figure 5.2 shows an example of a student model that was used. The coloring served to help students identify where the feedback-standard gap was the largest. No explicit standards were communicated to the students (e.g., "keep practicing until you reached 80% for all knowledge components"). For the sake of clarity, the descriptions in the student model were very concise. To promote reflection, the final page of each homework set contained a more detailed description of the knowledge components in the model.

After viewing the student model, students were free to choose whether and how to proceed their practice session. The tasks in the homework sets gradually increased in difficulty level, but students could choose to deviate from this predefined order of tasks. They were also allowed to resubmit solutions to tasks as many times they liked. Allowing resubmission offers students important opportunities to use the feedback from the student models in their learning process (Nicol & Macfarlane-Dick, 2006).

The student model design was evaluated in two earlier courses (Tacoma, Sosnovsky, Boon, Jeuring, & Drijvers, 2018; see Chapter 3 of this thesis). Participants in the current study had been enrolled in one of these earlier courses and, consequently, were already familiar with inspectable student models.

5.5.4 Study design

Students were randomly assigned to one of the four conditions (domain reasoner and student models available, only domain reasoner available, only student models available, and none of the feedback types available). Randomization took place within the three course variants, ensuring approximately equal group sizes within all three variants. Students were only included as participants if they met all of the following criteria:

- They gave active consent for use of their DME work and exam results for this study (618 students excluded);
- They worked on the homework sets for at least one hour, including breaks of up to five minutes (55 students excluded);
- They worked on at least one task with stepwise construction of hypothesis tests (3 students excluded);
- Their exam results for both the current course and the previous course were available (94 students excluded);
- They worked in only one condition (3 students excluded).

These inclusion criteria resulted in 521 students being included. Table 5.1 summarizes the number of students in each course variant and each experimental condition.

Table 5.1 Numbers of students in the three course variants and four experimental conditions

	SM†			No SM			All		
Course variant	DR	No DR	All	DR	No DR	All	DR	No DR	All
EDU	25	30	55	20	19	39	45	49	94
GCS	41	34	75	38	37	75	79	71	150
PSY	80	81	161	55	61	116	135	142	277
Total	146	145	291	113	117	230	259	262	521

†SM = student model feedback available, DR = domain reasoner feedback available

Data consisted of the students' work in the DME, exam results in the course and exam results in the previous course. DME work included all attempts at all tasks, as well as information about student model viewing. After anonymization, these DME log data were used to calculate the students' time-on-task in the DME, the number of open-ended hypothesis-testing tasks students correctly solved and the number of student model views, for those students who had student models available. From both exams, only the results on the statistics items were used. For the exam of the current course, values of Cronbach's α for the 15 statistics items were .61 for the GCS and PSY variant and .56 for the EDU variant. For the regular exam of the previous course, Cronbach's α for the 16 statistics items was similar, namely .59. Although these values are not high, they seem reasonable for exams that assess a wide variety of topics within the domain

of statistics (e.g., choice of statistical test, stating hypotheses, interpreting test output) with relatively few items (Taber, 2018). The exam results of the previous course were used as a measure of prior performance. The average of this prior performance score over all students was 11.3 points (SD = 2.5) and prior performance did not differ significantly between the four experimental conditions ($F(3, 517) = 0.28$, $p = .839$).

5.5.5 Data analysis

Four outcome variables were used to describe the students' learning processes and performance: time-on-task, student model views, hypothesis-testing score and exam result. Descriptions of these outcome variables are given in Table 5.2. For each of these four outcome variables, a multiple linear regression model was created, using the experimental conditions and several other variables as predictors.

Table 5.2 Outcome variables and potential predictor variables for the four multiple regression models

Model no.	Outcome variable	Description outcome variable	Extra predictor variables†
1	Time-on-task	Total time (in hours) students worked in the DME, including breaks of up to five minutes	Prior performance
2	Student model views	Number of times students viewed their student models	Prior performance
			Time-on-task‡
3	Hypothesis-testing score	Number of hypothesis-testing tasks (out of 3) in the DME for which students gave a complete correct solution	Course variant
			Prior performance
4	Exam result	Score on statistics items in the exam (out of 15 items)	Course variant
			Prior performance
			Time-on-task‡

†Variables domain reasoner and student model and their interaction were always used as predictor variables, except for model 2. In model 2, only students in the student model conditions were taken into account and the variable domain reasoner was included.

‡Time-on-task was only included as predictor variable if no strong relationship between time-on-task and condition variables would be found in model 1.

The outcome variable in model 1, time-on-task, was measured as the number of hours that students worked in the DME and was expected to possibly be influenced by student model availability: students with student models available were expected to reflect more on their learning and hence, possibly, to choose to work on more tasks. At the same time, however, time-on-task was also expected to be largely determined by student characteristics, such as diligence and motivation, that were not directly measured in this experiment. To enable taking these student characteristics into account in later models, it was, therefore, desirable to include time-on-task as independent variable in these models. Evidently, this could only be done if no strong relationship between experimental conditions and time-on-task would be found in model 1, because otherwise time-on-task would not be an independent variable. Hence, model 1 served to shed light on the relation between time-on-task and the experimental conditions, with the extra goal to assess whether time-on-task could be used as independent variable in later models.

Model 2, concerning the number of student model views, only included the students who had student models available. It allowed us to explore whether students with and without domain reasoner feedback used the student models differently. Model 3 concerned hypothesis-testing score: the number of hypothesis-testing tasks in the DME that students solved completely, out of the three stepwise hypothesis-testing tasks. This score was expected to be mainly influenced by domain reasoner availability. Finally, the outcome variable in model 4, exam result, concerned the students' score on the 15 statistics items in the exam and was expected to be influenced by both feedback conditions. Besides the two experimental conditions, three other variables were deemed important: prior course performance (score between 0 and 16), course variant (EDU, GCS, PSY) and time-on-task (if model 1 would not yield strong dependence of time-on-task on experimental conditions). The most widespread method to take such covariates into account when evaluating experimental conditions is an ANCOVA, but this method does not account for potential effects of interactions between experimental conditions and covariates on the outcome variables. Since we were especially interested in such potential interaction effects – for example, to investigate whether the effect of student model availability on exam results was different for

students with different prior performance scores – we opted for the more general approach of multiple linear regression.

For each of the four outcome variables, we started with a model including all predictor variables and interactions that were judged to be relevant. Table 5.2 summarizes the variables used at the start of creating each model. As argued above, time-on-task would only be included as predictor variable in models 2 and 4 if no strong relationship between time-on-task and the experimental conditions would be found in model 1. The predictor variables prior performance and time-on-task were always added centered to their mean. Next, step-by-step, non-significant interactions and predictors were removed from the model. In this phase, the experimental conditions and their interaction were retained in the models, regardless of their significance. Once a model was obtained in which all predictors apart from the experimental conditions were significant, this model was regarded as the complete model for that outcome variable. Because of the large influence that outliers can have on model parameters, outliers were removed and the model was fitted again, until no more outliers were found. The normality assumption of residual distribution and the assumption of homoscedasticity were checked as well. Next, to assess the influence of the experimental conditions and their interactions with predictor variables and with each other, these were removed from the complete model one by one. After each removal, the value of for the new model was calculated as measure of effect size, as well as an *F*-ratio for the comparison of the models before and after the removal.

5.6 Results

Table 5.3 summarizes the means and standard deviations of the outcome variables for all experimental conditions. In addition to the information in the table, it is worth mentioning that the students attempted on average 31 out of 34 tasks (*SD* = 5) and that the students with student models available viewed their student models for on average 48 seconds (*SD* = 63 seconds). The results of the regression analyses are presented in the following four sections.

Table 5.3 Overview of outcome variables for students in the four experimental conditions

		SM†			No SM			All		
Outcome variable		DR	No DR	All	DR	No DR	All	DR	No DR	All
	N	146	145	291	113	117	230	259	262	521
Time-on-task	Mean	6.0	6.1	6.1	5.6	5.7	5.6	5.8	5.9	5.9
	SD	2.5	2.5	2.5	2.3	2.4	2.4	2.5	2.4	2.4
Student model views	Mean	5.4	6.3	5.9	-	-	-	-	-	-
	SD	4.4	5.3	4.9	-	-	-	-	-	-
Hypothesis-testing score	Mean	1.1	1.1	1.1	12.2	1.0	1.1	1.2	1.1	1.1
	SD	1.1	1.0	1.1	1.0	1.0	1.0	1.0	1.0	1.0
Exam result	Mean	11.0	11.2	11.1	11.3	11.1	11.2	11.1	11.1	11.1
	SD	2.5	2.3	2.4	2.5	2.4	2.4	2.5	2.3	2.4

†SM = student model feedback available, DR = domain reasoner feedback available

5.6.1 Time-on-task

The parameter estimates of the regression model predicting time-on-task are summarized in Table 5.4. Four outliers were removed: students who worked more than 13.2 hours in the DME during the course period. The complete model showed no significant effects of either of the experimental conditions, nor of their interaction, on time-on-task in the DME. Prior performance significantly influenced time-on-task, with students who performed better in the previous course also working more in the DME. Furthermore, after removing the interaction between domain reasoner and student model availability, the availability of student models had a significant positive effect on students' time-on-task. This effect is also reflected in the average time-on-task reported in Table 5.3, which was 6.1 hours for students with student models and 5.6 hours for students without. The model fit for the complete model, as well as the model without the interaction, was, however, poor: R^2 = .023. This means that prior performance, student model availability and domain reasoner availability together only explained 2.3% of the variance in time students worked in the DME. Removing student model availability from the model showed that it explained 0.9% of the variability in time-on-task, a small but significant contribution ($F(1, 513) = 4.26$, p = .040). Since this relationship was only weak, it was deemed justifiable to include time-on-task as predictor variable in later regression models: these results imply that time-on-task seemed to be mainly determined by other factors than student model or domain reasoner availability.

Table 5.4 Parameter estimates and model fits for model 1: time-on-task predicted by experimental conditions, their interaction, and prior performance

	Complete model	Interaction removed	Student model removed	Domain reasoner removed
Intercept	5.53***	5.59***	5.82***	5.82***
Prior performance	0.11**	0.11**	0.11**	0.11**
Domain reasoner available	0.09	-0.03	-0.03	
Student model available	0.53	0.42*		
Domain reasoner × Student model	-0.21			
R^2	.023	.023	.014	.014
R^2 change		.000	.009	.000
F change		0.27	4.26*	0.02

*p < .05; **p < .01; ***p < .001.

5.6.2 Student model views

The model predicting the number of student model views included the predictor time-on-task and its interaction with domain reasoner availability. Prior performance was found not to be a significant predictor. The parameter estimates and model fits for the complete model are given in Table 5.5. One outlier was removed and the model showed some heteroscedasticity: the number of student model views varied widely for high values of time-on-task and far less for low values of time-on-task.

The complete model explained 25% of the variance in number of student model views. It should be noted that this might be a slight overestimation, because of the above-mentioned heteroscedasticity. The model shows, not surprisingly, that students who worked more in the DME also tended to view their student models more often. Removing the interaction between domain reasoner and time-on-task from the model showed that this interaction term accounted for 3.4% of the variance and contributed significantly to the model ($F(1, 287) = 10.63$, $p = .001$). This means that while there was no main effect of domain reasoner availability, there was an interaction effect, which is illustrated in Figure 5.5. Of the students who used the DME intensively, those with domain reasoner feedback available viewed their student models less often than their peers who did not receive domain reasoner feedback.

Table 5.5 Parameter estimates and model fits for model 2: number of student model views predicted by domain reasoner availability, time-on-task and their interaction

	Complete model	Interaction removed	Domain reasoner removed
Intercept	5.84***	5.94***	5.58***
Time-on-task	1.22***	0.89***	0.89***
Domain reasoner available	-0.56	-0.73	
Domain reasoner × Time-on-task	-0.65**		
R^2	.246	.212	.212
R^2 change		.034	.000
F change		10.63**	2.17

*p < .05; **p < .01; ***p < .001.

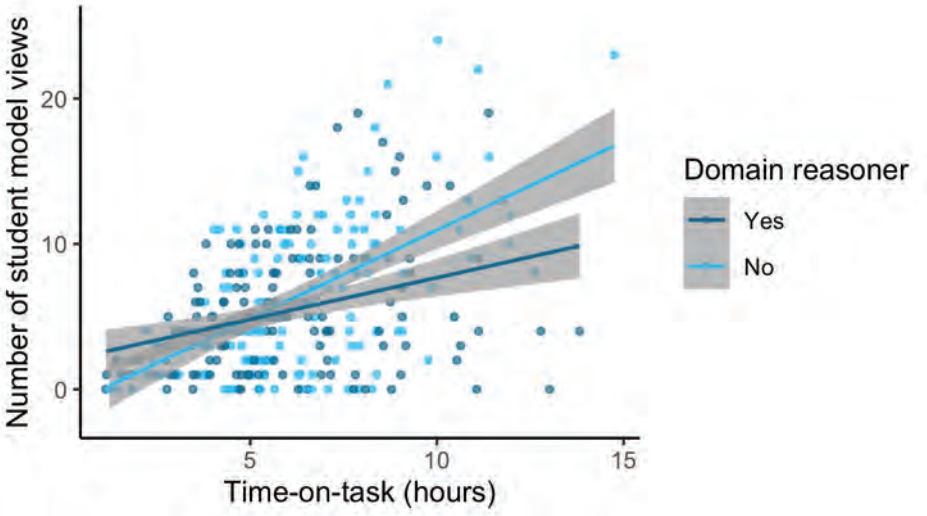

Figure 5.5 Effect of the interaction between time-on-task and domain reasoner availability on number of student model views

5.6.3 Hypothesis-testing score

Since students in the PSY variant and domain reasoner conditions already had previous experience with the domain reasoner, whereas students in the other two course variants had not, an interaction effect between the domain reasoner condition and course variant was included in the regression model predicting students' hypothesis-testing score. The model's parameter estimates are summarized in Table 5.6. No outliers were detected, but the normality assumption of residual distribution was

violated. This was not regarded a problem, because of the large sample size of 521 students.

Table 5.6 Parameter estimates and model fits for model 3: hypothesis-testing score predicted by experimental conditions, course variant, their interactions, and prior performance

	Complete model	Interaction DR† course variants removed	Interaction SM DR removed	DR removed	SM removed
Intercept	1.03***	0.74***	0.79***	0.84***	0.83***
Prior performance	0.04*	0.04*	0.04*	0.04*	0.04*
Variant EDU	0.11	0.42**	0.42**	0.42**	0.42**
Variant PSY	-0.03	0.42***	0.42***	0.42***	0.42***
Domain reasoner available	-0.36	0.18	0.08		
Student model available	0.09	0.06	-0.03	-0.02	
Domain reasoner × EDU	0.59*				
Domain reasoner × PSY	0.88***				
Domain reasoner × Student model	-0.23	-0.18			
R^2	.078	.045	.043	.042	.041
R^2 change		.033	.002	.001	.001
F change		9.31***	0.99	0.88	0.07

†DR = domain reasoner feedback available, SM = student model feedback available
*p < .05; **p < .01; ***p < .001.

The complete model explained 8% of the variance in hypothesis-testing score. Consecutively removing interactions and the experimental conditions revealed that the interaction between domain reasoner availability and course variant explained 3.3% of the variance, which was a significant contribution to the model ($F(2, 514) = 9.31$, $p < .001$). Domain reasoner availability itself, student model availability and their interaction did not significantly contribute to the model. The significant effect of the interaction between domain reasoner availability and course variant reveals that the domain reasoner's effectiveness was different for the different groups of students. To further investigate this interaction, Table 5.7 summarizes hypothesis-testing scores for students with and without domain reasoner feedback within the three course variants. The t-test results show that the

students in the GCS variant performed better without than with domain reasoner feedback, while students in the PSY course variant performed significantly better when domain reasoner feedback was available to them. No significant effect was found for the EDU group.

Table 5.7 Hypothesis-testing score by course variant

Course variant	DR†	No DR	*t*	df	*p*	Cohen's *d*
EDU (*N* = 94)	1.3 (0.9)	1.2 (1.1)	0.42	92	.678	-
GCS (*N* = 150)	0.6 (0.8)	1.1 (1.0)	-3.26	148	.001	0.53
PSY (*N* = 277)	1.4 (1.1)	1.1 (1.0)	2.95	275	.003	0.35

†DR = domain reasoner feedback available

5.6.4 Exam result

The parameter estimates of the model predicting exam result from the experimental conditions, prior performance and time-on-task are given in Table 5.8. Course variant was no significant predictor of exam result and one outlier was removed.

Table 5.8 Parameter estimates and model fits for model 4: exam result predicted by experimental conditions, prior performance, time-on-task and interactions

	Complete model	Interaction SM prior performance removed	Interaction SM DR removed	SM removed	DR removed
Intercept	11.17***	11.16***	11.31***	11.17***	11.13***
Prior performance	0.47***	0.40***	0.39***	0.39***	0.39***
Time-on-task	0.19***	0.19***	0.19***	0.19***	0.19***
Domain reasoner available	0.22	0.22	-0.09	-0.09	
Student model available	0.01	0.01	-0.26		
Student model × Prior performance	-0.16*				
Domain reasoner × Student model	-0.54	-0.54			
R^2	.244	.237	.234	.231	.231
R^2 change		.007	.003	.003	.000
F change		4.63*	2.18	1.94	0.24

*p < .05; **p < .01; ***p < .001.

The complete model explained 24% of the variance and showed that prior performance and time-on-task significantly affected exam result. Furthermore, it revealed an interaction effect between student model availability and prior performance. This interaction effect explained 0.7% of the variance in exam result and contributed significantly to the model ($F(1, 514) = 4.63$, $p = .032$). Student model availability, domain reasoner availability and their interaction did not contribute significantly to the model. Figure 5.6 illustrates the interaction effect between student model availability and prior performance. For low prior performance scores, the regression line for students with student models is higher than the one for students without student models, meaning that students with low prior performance benefited from the student models. The opposite holds for students with high prior performance: for these students, availability of student models had a negative effect on course performance.

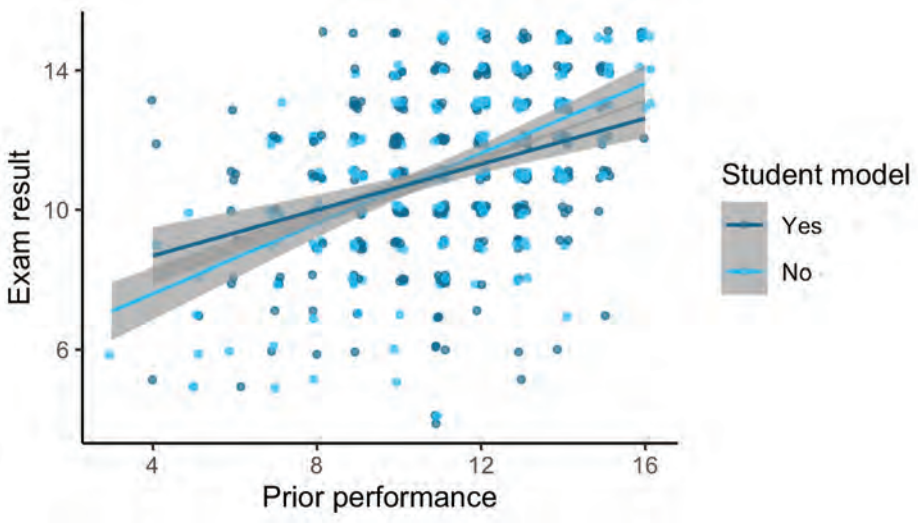

Figure 5.6 Effect of the interaction between student model availability and prior performance on exam result. For visualization purposes, some random noise was added to prior performance and exam result

5.7 Discussion

While many research studies address only inner loop or only outer loop feedback, many ITSs used in educational practice offer these two feedback types simultaneously. Therefore, the aim of this study was to

assess the effects of combined inner and outer loop feedback. The guiding research question was: what effects does providing both inner and outer loop feedback on online homework have on students' learning process and course performance in a university statistics course? To answer this question, in the following section we first discuss the effects we found of both feedback types separately and next reflect on their combination.

5.7.1 Effects of the feedback types and their combination

Inner loop feedback was provided in the form of a domain reasoner for tasks in which students set up hypothesis tests. Students with prior experience with the domain reasoner benefited from its feedback for solving hypothesis-testing tasks, but students without prior experience did not. This corroborates earlier findings that students may need some time to familiarize themselves with inner loop feedback (Tacoma et al., 2019; see Chapter 2 of this thesis). The effect size for students already familiar with the feedback was $d = 0.35$, which is slightly smaller than the average effect size of 0.50 that Van der Kleij and colleagues found for elaborate feedback (Van der Kleij et al., 2015). Furthermore, no direct effects were found of domain reasoner availability on time-on-task and exam result. Given the mixed effects of domain reasoner availability on students' hypothesis-testing score, this lack of a positive effect on exam result is not surprising (Shute, 2008).

Outer loop feedback was implemented in the form of inspectable student models. Student model availability slightly influenced the time students worked in the ITS: students with student models tended to work slightly longer than students without. This effect was small: student model availability only explained 0.9% of the variance in time-on-task. This is similar to results reported by Sosnovsky and Brusilovsky (2015), who found a correlation coefficient of $r = .13$, which, squared, yields 1.7% explained variance. Student model availability did not affect hypothesis-testing scores, but did seem to affect exam results. The exam results revealed that students with low prior performance slightly benefited from the student models, while students with high prior performance were slightly hindered by them. This finding is similar to findings from studies in which extra components – a social component or support for appropriate

task selection – were added to the student models (Brusilovsky et al., 2015; Mitrovic & Martin, 2007).

Having established the effects of both feedback types separately, we now turn to their combination to answer our research question completely. Our findings showed no interaction effects between the feedback types concerning performance outcome variables. Hence, in this study the two feedback types did not amplify or attenuate each other's effects on students' course performance. Regarding the students' learning process, however, domain reasoner availability did influence the students' use of the student models: students who used the ITS intensively tended to view the student models less often if they also received domain reasoner feedback. This means that for the students' learning process, domain reasoner availability did attenuate students' student model use, but this did not affect the students' course performance.

5.7.2 Revisiting feedback principles

How can these findings be interpreted in the light of the feedback principles we identified in section 5.2? The first principle states that feedback is a process, including phases of evidence collection and analysis, information delivery and students' use of the feedback (Pardo, 2018; Timmers et al., 2013). Our results illustrate that these phases may influence each other in different ways for high-achieving and low-achieving students. The finding that high-achieving students did not benefit from the student models could be a consequence of design choices in the evidence analysis phase: our procedure may have resulted in too optimistic estimations of high-achieving students' domain knowledge. This, in turn, may have given these students the impression that they were well prepared for the exam and did not need further practice, resulting in suboptimal use of the feedback. Meanwhile, for lower-achieving students the calibration of the estimations seems to have been more appropriate, given that student model availability had a positive, though small, effect on these students' exam results. Other subtle design choices may have influenced the feedback process as well, such as a page notifying students of the student models, the setting that students could attempt tasks as often as they liked and the exact wording of the feedback messages that the domain reasoner provided. Even when adhering to general guidelines for effective feedback, such design details

can considerably influence its actual effectiveness (Kluger & DeNisi, 1996; Zakharov et al., 2005).

The second feedback principle states that feedback should address the feedback-standard gap and offer opportunities to close it. While domain reasoner feedback quite explicitly provided guidance in closing this gap by mentioning key concepts, our implementation of student models did not provide explicit suggestions about how to reduce the gap, in terms of appropriate tasks or reading material. This could explain why students with both feedback types available tended to use the student models less: the more explicit suggestions by the domain reasoner feedback may have been easier to follow than the implicit messages the student models gave them. Earlier research on student models has shown that explicit suggestions can contribute to keeping students engaged with learning material (Arroyo et al., 2007) and to help students to allocate their attention to appropriate tasks given their current knowledge (Sosnovsky & Brusilovsky, 2015). It should be noted, however, that implementing such explicit suggestions puts higher demands on course design than the approach opted for in this study.

The third feedback principle states that feedback should facilitate students' reflection on learning. Inspectable student models are valued for the opportunities for planning and reflection that they offer (Bull & Kay, 2016), but in this study only the weaker students benefited to some extent from these opportunities. Furthermore, the finding that outer loop feedback was used less by students who also received inner loop feedback suggests that the students' need or capacity for reflection is limited. Students may not have had the cognitive capacity to process both feedback types together optimally. In other words, our results may indicate a feedback ceiling effect: a maximum amount of feedback that students can process at once.

5.7.3 Limitations

While revisiting the feedback principles in section 5.7.2, we discussed two aspects of this study that could be regarded as limitations: the exact calibration of estimations in the student models and the absence of explicit suggestions in the student models to close the feedback-standard gap. A

third limitation of this study is that both feedback interventions were rather small. On average, students worked almost six hours on homework tasks in the ITS, but typically spent less than one minute viewing their student models. Likewise, only three out of the 34 tasks in the homework sets were stepwise hypothesis-testing tasks in which students could receive feedback from the domain reasoner. Hence, opportunities to learn from the student models and domain reasoner feedback were rather sparse. For the domain reasoner feedback this was especially true for students who had no prior experience with it and hence, presumably, needed time to familiarize themselves with its feedback (Tacoma et al., 2019; see Chapter 2 of this thesis). Arguably, providing the evaluated feedback types for longer time periods (one academic year, or throughout a complete study program) could lead to larger learning effects (Evans, 2013) and would be a promising direction for further research.

5.7.4 Implications and recommendations

This study has a number of implications for theory and practice. First, the interaction effects found between feedback conditions, prior performance and prior experience illustrate that the same feedback may have different effects for different learners. This finding supports theory reflected in Pardo's model for data-supported feedback (Pardo, 2018): the learner's knowledge, beliefs and attitudes influence how feedback changes a student's strategies and learning process. For educational practice, this implies that providing multiple types of feedback, such as both inner loop and outer loop feedback, may result in more students receiving feedback that is helpful for them.

In this study, we also found indications, though, that introducing multiple feedback types at once may result in suboptimal use of the feedback, a feedback ceiling effect. Introducing feedback types one by one would therefore be recommendable. Taking this a step further, based on our results we suggest that lower-achieving students could first be provided with inspectable student models, since they seem to benefit from this outer loop feedback. Higher-achieving students could be expected to familiarize themselves more quickly with a new feedback type and, hence, could benefit more quickly from detailed inner loop feedback. We encourage further research investigating this hypothesis.

Finally, regardless of the exact implementation, our findings imply that students need time to get used to new feedback and to know how the feedback can help them learn. From a theoretical perspective, this suggests that the three feedback principles could be supplemented with a fourth regarding the amount of feedback: students should be given enough time and opportunities to familiarize themselves with and to learn from feedback. For educational practice, this implies that new feedback implementations should be offered for substantial periods of time (i.e., preferably longer than one semester) and students should be offered sufficient guidance in interpreting feedback information.

Acknowledgements

Utrecht University's Education Incentive Fund supported the design and implementation of the educational materials used in this study. The authors wish to thank Jeltje Wassenberg-Severijnen, teacher of the course, for the valuable collaboration and Wouter van Joolingen and Nathalie Kuijpers for their comments on earlier versions of this paper.

CHAPTER 6 General discussion

6.1 Introduction

Due to technological advancements in the past decades, in particular the emergence of powerful digital tools to collect, store, analyze and represent big datasets, the field of statistics is changing rapidly. Because most calculations can now be outsourced to calculators and computers, introductory university statistics courses are changing as well: the focus is shifting from manipulating statistical formulas and carrying out statistical techniques to knowing and being able to reason with the underlying concepts and principles (Carver et al., 2016). Students need to develop statistical proficiency: knowledge of why data and statistical formulas are needed, how these can inform decisions, and how variability in data can affect the results of statistical techniques. Developing statistical proficiency is challenging for students, though. Students struggle to build logical chains of reasoning involving many abstract statistical concepts, such as probability distributions and sampling variability (Castro Sotos, Vanhoof, Van den Noortgate, & Onghena, 2007; Falk & Greenbaum, 1995). Individual guidance and feedback might be a means to support students in facing these challenges, but in the context of higher education – with typically large student group sizes – this is difficult for teachers to achieve. In this research project, a solution to this issue was sought in the provision of automated intelligent feedback. The aim of this research project, therefore, was to design and evaluate automated intelligent feedback in a computer-based learning environment that addressed the difficulties that social sciences students experience in developing statistical proficiency. The guiding research question was:

> *How can automated intelligent feedback support first-year university students in developing statistical proficiency?*

Two types of automated intelligent feedback were designed and implemented in online homework sets within two introductory statistics courses for students enrolled in social sciences bachelor programs at Utrecht University. Inner loop feedback addressed the students' steps in

carrying out hypothesis tests, while outer loop feedback provided students with overviews of their current understanding of important statistical concepts. For addressing our research question, three aspects of feedback implementation were deemed important: feedback design, students' use of the feedback, and effects of the feedback on the students' statistical proficiency. Four design research cycles were carried out to address these three aspects for both inner and outer loop feedback. These four cycles have been addressed in separate chapters of this thesis.

6.2 Research overview and main findings

In **Chapter 2** we discussed the first cycle concerning inner loop feedback, in the form of a domain reasoner for hypothesis testing. In this cycle, the design, use by students, and direct effects of the inner loop feedback were addressed. This chapter was guided by the research question:

> *2.1 Does automated intelligent feedback about the logic of hypothesis testing contribute to student proficiency in carrying out hypothesis tests?*

The design of the domain reasoner combined characteristics of two prevailing paradigms in research on Intelligent Tutoring Systems (ITSs): constraint-based modeling (Mitrovic, Martin, & Suraweera, 2007) and model-tracing (Anderson, Corbett, Koedinger, & Pelletier, 1995). The constraint-based characteristics allowed the system to identify missing elements and inconsistencies in students' solutions, while the model-tracing elements enabled addressing common errors and providing hints for subsequent steps. Concerning the research question, we concluded that, after a familiarization phase, the intelligent feedback effectively supported students in solving tasks on hypothesis testing. Additionally, the number of errors that students made in their logical reasoning decreased more strongly over tasks for students who received feedback from the domain reasoner than for students who received verification feedback only. This suggests that the intelligent feedback fostered student proficiency in independently carrying out hypothesis tests. These positive effects did, however, not transfer to follow-up tasks about hypothesis testing. In section 6.4 we discuss possible causes of this lack of transfer.

Whereas the implementation of inner loop feedback concerned only a limited number of tasks, the outer loop feedback, in the form of inspectable student models, made use of evidence from almost all tasks in the online homework sets. Informed by didactical recommendations from statistics education research, these tasks were clustered around real datasets and contexts and were consciously sequenced. Designing inspectable student models based on such clusters of tasks, as opposed to mutually independent tasks, was regarded as an important design challenge. This design challenge was discussed in **Chapter 3**, guided by the following research questions:

3.1 Are inspectable student models suitable for implementation in didactically grounded, sequential statistics modules consisting of closely related tasks?

3.2 How can didactical analysis inform design of inspectable student models and, vice versa, how can student model evaluation methods inform didactical design?

From a student perspective, the answer to research question 3.1 was positive. The students valued the student models for their clarity and close connections to the tasks. The learning curve analysis that we used to assess the internal validity of the student models revealed a different picture, however: for almost half of the knowledge components in the student models the number of errors students made was found to increase, rather than decrease, over time. Hence, additional effort was needed to design well-defined, valid student models for these didactically grounded, sequential homework sets on statistics. This additional effort consisted of complementing the learning curve analysis with didactical task analysis and, informed by these analyses, redesigning the student models and tasks. Concerning research question 3.2, both analysis techniques proved valuable in this process. For the knowledge components with increasing error rates over time, the learning curves were scrutinized and the tasks connected to the knowledge components were analyzed to find possible causes of the increasing error rates. Four problems causing increasing error rates were identified and are briefly discussed here.

The first two problems that caused increasing error rates concerned the design of the student models. The first one was that definitions of some knowledge components were too broad, meaning that in fact they constituted more than a single statistical concept. In the redesign, such knowledge components were split into two or more knowledge components. The second problem was that mistakes students made in a task affected the student model scores for all knowledge components connected to the task. In many cases, only one of these knowledge components was problematic for solving the task and, hence, student model scores for the other connected knowledge components became unreasonably low. To resolve these issues, in the redesign some connections between tasks and non-problematic knowledge components were removed. Remarkably, the third and fourth problems causing increasing error rates did not concern the design of student models, but the design of the tasks in the homework sets. The third problem was that some tasks were found to be didactically meaningful for the homework set, but not suitable for informing student models. This concerned, for example, easy introductory tasks, or tasks to illustrate a specific characteristic of a concept. In the redesign, connections between these tasks and knowledge components were removed, since these tasks did not provide useful information for the student models. The fourth and final problem was the most striking example of how didactical analysis and student model evaluation methods can strengthen each other: together they revealed that some tasks related to difficult concepts addressed these concepts too superficially. Error rates were very low in these cases, meaning that students barely made errors regarding these difficult concepts. We concluded that the designed tasks in these cases did not provide students with enough opportunities to make mistakes and to learn from these mistakes. In this way, learning curve analysis did not only disclose weaknesses in student model design, but also in the design of the tasks in the homework sets themselves. For these tasks, redesign focused on creating more opportunities to reason with the difficult concepts and to make mistakes in this reasoning.

Based on the findings presented in Chapter 3, in the next cycle the homework sets and inspectable student models were revised, taking the various roles and goals of the tasks in the homework sets into account. **Chapter 4** focused on the students' use of the student models and on the

extent to which these informed the students' choices for subsequent study steps. The research questions for Chapter 4 were:

4.1 How do first-year university students in social science seek feedback from inspectable student models in an introductory statistics course?

4.2 How does feedback from inspectable student models inform these students' decisions about subsequent actions?

4.3 How does these students' feedback-seeking and decision-making behavior relate to performance on a statistics exam?

Concerning question 4.1, students were found to keep consulting their student models throughout the course period, albeit slightly less frequently towards the end of the course. A wide variety in timing, duration and amount of detail of student model views was found, both between students as well as between different student model views by the same student. This variety could be explained from differences in underlying student self-motives (Anseel, Lievens, & Levy, 2007). For example, for quickly verifying one's weaker and stronger topics, a shorter and less detailed student model view suffices than would do for consciously planning subsequent study steps. A similar variety was found regarding research question 4.2 concerning student decisions about subsequent actions, based on the feedback from their inspectable student models. Across this diversity, though, students seemed inclined to improve their work on the homework sets when student model scores were low, and to work on extra practice tasks or new topics when student model scores were high. This suggests that the inspectable student models may have encouraged students to devote more effort to their homework sets than they would have done without having the student models available. Concerning research question 4.3, we conclude that both the frequency of student model viewing and the amount of variety in decisions made after viewing the student model are positively related to exam results. Not surprisingly, student activity, as measured by time on task, plays a role in these relationships: students who spend much time in the learning environment tend to view their student models often and also tend to score high on the final exam. Yet,

our findings suggest that other factors, such as the students' self-motives mentioned above, seem to be important as well. Although the absence of a control group in this study prevented us from drawing causal inferences, these findings suggest that frequently inspecting student models and using them to inform subsequent study steps can be a fruitful learning strategy.

Teachers and educational designers are inclined to combine different promising approaches for delivering rich, inspiring courses. Therefore, many ITSs offer both inner loop feedback, to support students while working on specific tasks, and outer loop feedback, to guide students in their learning process. **Chapter 5** provided a thorough evaluation of the effects of offering both feedback types designed in this research project together, guided by the research question:

> 5.1 *What effects does providing both inner and outer loop feedback on online homework have on students' learning process and course performance in a university statistics course?*

The effects found for inner loop feedback were similar to those discussed in Chapter 2: students receiving inner loop feedback performed better on hypothesis-testing tasks, but only if they had prior experience with the domain reasoner. No effects of domain reasoner availability on course performance were found. In section 6.4, we explore possible causes for this lack of effects. We did find small effects of student model availability on course performance, though. As in Chapter 4, these effects varied widely between students: students with low prior performance seemed to benefit from the student models, while students with high prior performance seemed to be hindered by them. We also found a small effect of student model availability on the time students worked in the ITS, with students receiving outer loop feedback working slightly longer. Finally, inner and outer loop feedback did not influence each other's effects on course performance, but did compete for attention in the students' learning processes: students who used the ITS intensively tended to use the outer loop feedback less if they also received inner loop feedback.

All in all, our research project has shed light on the design and implementation of automated intelligent feedback to support university

students in developing statistical proficiency. The field of artificial intelligence in education provided excellent starting points for designing both inner and outer loop feedback. In designing inner loop feedback, combining the prevailing paradigms model-tracing and constraint-based modeling resulted in a domain reasoner that could effectively detect and address inconsistencies in students' hypothesis-testing procedures. For the outer loop feedback, theory on generating and visualizing student models informed our design of inspectable student models that offered students insight into their current understanding of the domain of statistics. Both feedback types affected the students' learning processes. The domain reasoner supported students in solving more hypothesis-testing tasks and in learning from their mistakes. Hence, after working with the domain reasoner for a while, students needed its feedback less and less for solving hypothesis-testing tasks correctly. The inspectable student models seemed to stimulate students to devote more effort to their homework tasks, which, for lower-achieving students, ultimately led to better exam results. The research project also revealed, however, that the interaction between the two feedback types did not seem to have positive effects on the students' statistical proficiency and that the effects of both feedback types varied widely among students.

6.3 Contributions

The main contribution of this research project is the evidence found that feedback tools developed within the field of artificial intelligence in education, in particular domain reasoners and inspectable student models, can be valuable in university statistics education. Many students can benefit from these feedback types, especially when they are given enough time to get used to them. The domain reasoner enables checking students' solutions to hypothesis-testing tasks on a conceptual and detailed level that would otherwise be unachievable for such large groups of students. Furthermore, the domain reasoner allows teachers and designers to quickly create new tasks that provide automated intelligent feedback. The domain reasoner is independent of the datasets and contexts used in the tasks and hence supports a wide variety of contexts from both within and outside the social sciences. Students value the insights into their current knowledge that the inspectable student models provide. Especially in a domain with

so many abstract concepts and complex relations between these concepts (Castro Sotos et al., 2007), such insights are valuable means to inform further study steps. Finally, the results of this research project show that, with caution, it is quite feasible to integrate these feedback types into didactically grounded online homework sets that use real datasets, rich interaction types, and conscious task sequencing to address statistical proficiency.

A second contribution concerns the combination of inner loop and outer loop feedback. Although these feedback types are often combined in educational practice, the question of whether combining them influences their effects had not been studied yet. Our results show that combining the two feedback types within one learning environment is possible: our combination of inner and outer loop feedback did not reduce the effects of both feedback types on student performance. The two feedback types may, however, slightly compete for the students' attention in the learning process: students who have access to both inner and outer loop feedback may use outer loop feedback less than students with only outer loop feedback available. Given the variety of feedback effects for different students, we still regard implementing both feedback types as an added value: it increases the number of students who receive feedback that they perceive as helpful.

A contribution we hoped for, but did not find, was clear positive feedback effects on students' statistical proficiency in the longer term. Although inner loop feedback did support students in solving more of the hypothesis-testing tasks that provided inner loop feedback, these positive effects did not transfer to follow-up tasks on hypothesis testing. Furthermore, no effects of inner loop feedback and mixed effects of outer loop feedback on students' course performance were found. In the next section, we discuss potential causes for the absence of these positive longer-term effects and possible avenues of further research into whether such effects could be achieved.

6.4 Limitations

The main research question of this research project concerned statistical proficiency. One might wonder whether the data we have used for assessing students' statistical proficiency – student performance on follow-up hypothesis-testing tasks and on final exams – do adequately reflect statistical proficiency. The follow-up tasks and exam items were designed by the teachers of the course and were not specifically designed as research instruments to measure statistical proficiency. Many of these tasks and items addressed very specific elements of the hypothesis-testing procedure, rather than the hypothesis-testing procedure as a whole. Students may, therefore, have developed statistical proficiency regarding the logic of the hypothesis-testing procedure that was not assessed in these tasks and items. Designing tasks to not only address, but also assess statistical proficiency is a challenge for future research.

Other methods to gain insight into the development of statistical proficiency might be focus groups discussions and interviews. Compared to the methods used in this research project, however, these methods have the disadvantage of being labor-intensive and time-consuming for both researchers and participants. Furthermore, although statistical proficiency was not measured directly, is was also not neglected: from the beginning, it played a crucial role in the design of the tasks and of both feedback types. The wish to provide students with insight into their understanding of statistical concepts was, for example, an important argument to decide to implement inspectable student models in the first place. Regarding the domain reasoner design, attention was paid to formulating feedback messages in such a way that they addressed the meaning of steps and conceptual relations between steps, rather than only the sequence of the steps.

A second limitation of this research project is that embedding the two feedback types within the homework sets resulted in relatively small adjustments to the total course: the domain reasoner provided feedback on only three to six out of over thirty tasks and the student models were only an addition to the homework tasks. Not all connections between tasks and concepts in the student models were made explicit and those that were, may not always have been clear to students. It may, therefore, have

been difficult for students to select suitable tasks after viewing the student models. Hence, while both feedback types may have been effective in helping students troubleshoot their performance, students could have been given more opportunities to self-correct, or, in other words, to close the gap between current and desired performance (Nicol & Macfarlane-Dick, 2006). It should be noted, though, that implementing such opportunities evidently requires additional design effort and time and, hence, decreases the feasibility of such an approach from the perspective of educational practice.

A final limitation concerns our choices in the implementation of the student models. Despite findings that were useful to improve the student models (Chapter 3), the predictive validity of the student models remained relatively low. This could be a consequence of our choice to implement coarse-grained domain models, similar to work by Sosnovsky and Brusilovsky (2015), to facilitate development of inspectable student models by teachers. Literature does offer suggestions, though, to improve student model validity even under such conditions. Most promising seems the use of Conjunctive Bayesian Knowledge Tracing (CBKT), an alternative method for calculating student model scores (Koedinger, Pavlik, Stamper, Nixon, & Ritter, 2011). This method does not require extra design effort from teachers, but does have better mechanisms for coping with students' mistakes on tasks that involve multiple statistical concepts. Instead of lowering the student model scores for all related concepts, as was the case in our implementation, CBKT determines for which of these concepts the student's score is already lowest. A lack of understanding of this concept is taken as the most likely cause of the student's mistake and, therefore, the score for this concept is lowered most. This calculation method may also facilitate better student model calibration, which may prevent the possibly too optimistic student models for high-achieving students that we conjectured in Chapter 5. For these reasons, if we were to start implementing inspectable student models today, we would opt for this CBKT approach.

6.5 Implications for future research and educational design

Despite the limitations discussed in section 6.4, this research project has a number of implications for future research and educational design. In this section we outline a long-range vision for the use of inner and outer loop feedback in university statistics education and identify directions for further research to move towards that vision. Short-term recommendations based on this research project are discussed in section 6.6.

6.5.1 Assessment through homework tasks

By integrating inner and outer loop feedback into homework tasks, these homework sets obtain characteristics of formative assessment: the students' performance is continuously assessed with the aim of improving the learning process (Birenbaum et al., 2015). Providing many low-stakes assessment tasks is presumed to enhance students' motivation and self-esteem (Nicol & Macfarlane-Dick, 2006). More importantly, a main goal of higher education is to support students in becoming independent learners, who are capable of monitoring, evaluating and regulating their learning (Evans, 2013). Low-stakes formative assessment can play a valuable role in this process, by generating accessible information regarding students' knowledge and understanding that students can attempt to interpret and use to enhance their learning process. Such information should help students clarify what good performance is, in terms of desired goals or standards (Nicol & Macfarlane-Dick, 2006). In a carefully designed curriculum, such assessment information does not have to be provided solely by the computer-based learning environment that offers the homework sets, but could also be addressed in other curriculum elements such as a course manual, lectures and discussion sessions guided by teachers. As such, online homework sets with inner and outer loop feedback should be designed as an integrated part of the curriculum, keeping their role and relations with other curriculum elements in mind (Ritter, Anderson, Koedinger, & Corbett, 2007).

Taking the idea of assessment through homework tasks a step further, VanLehn (2008) argued that homework in a computer-based learning environment can be used as summative assessment as well. The

students' interactions with homework tasks, and especially their attempts at constructing multistep solutions to complex problems such as hypothesis-testing tasks, contain valuable information about their understanding of the concepts involved. Using this evidence for summative assessment may have several advantages, such as freeing up assessment time, assessing "usual performance" rather than "assessment performance" and tracking students' evolving knowledge continuously (VanLehn, 2008). In this sense, implementing inner and outer loop feedback into online homework sets could be a step towards eliminating the distinction between practice and assessment in education, a long-range vision for the use of technology and artificial intelligence in education (Gobert, Sao Pedro, Raziuddin, & Baker, 2013; Quellmalz & Pellegrino, 2009).

For this long-range vision to become reality, future research should address a number of questions. In the following, we focus on three such questions: the validity of inferences about student knowledge from practicing tasks, the role of the teacher when implementing substantial computer-based elements in a (statistics) curriculum, and ongoing developments in the field of statistics, which affect statistics education as well.

6.5.2 Inferring knowledge from practicing tasks

The first question relates to the nature of tasks and feedback in homework sets that have the aim to provide practice opportunities. To facilitate practice, homework tasks typically provide immediate feedback and the option to try until correct. Because of these characteristics, students' answers to homework tasks do not reflect student knowledge in the same way as answers to assessment items that can be attempted only once. In such assessment items, a correct answer can be assumed to be the result of correct reasoning. In practicing tasks, meanwhile, a correct answer could also be the result of correct interpretation of feedback along the way. In a practice setting, both these ways of arriving at a correct answer are acceptable, but this makes inferring what a student knows challenging. Likewise, incorrect answers may be the result of lacking understanding, but, in our experiments, students also indicated to sometimes fill in or select incorrect responses out of curiosity, to see what immediate feedback that would evoke (Tacoma, Drijvers, & Boon, 2017). In other words, students get used to the way a learning environment responds to

their answers and adjust their behavior accordingly. Further research is needed to provide more insight into how specific system characteristics affect student behavior when they engage in practicing tasks that provide immediate feedback and, consequently, into what this means for validly inferring student knowledge from their responses. Such research could also include the question of how inner loop feedback and outer loop feedback can inform each other. For example, when a domain reasoner identifies multiple problems in a student's solutions state, the current student model could be used to prioritize them. The amount of inner loop feedback could also be gradually faded for students whose student model scores are high, while students with lower student model scores are offered elaborate inner loop feedback as long as they need it.

6.5.3 The role of the teacher

A second direction for further research concerns the role of the teacher in a curriculum involving substantial computer-based elements. Computer-based learning environments are typically designed to carry out duties that, traditionally, a teacher would take on (VanLehn, 2011). Providing detailed feedback on students' answers, as in our research project, is one example: other duties may be to explain theory in videos or animations and to discuss content or negotiate with students about their knowledge (e.g., Bull & Kay, 2016). This does not imply, however, that computer-based learning environments could or should replace human teachers. First, in higher education computer-based learning environments can often *supplement* human teachers, rather than *replace* them. For a teacher, providing individualized feedback to large groups of students generally requires too much time to be feasible. Employing computer-based learning environments for this purpose can, therefore, be an enrichment of the education that the teacher provides. Second, a concern regarding the use of computer-based learning environments may be their reputation of promoting learning of procedural skills, rather than of conceptual understanding (Salden, Aleven, Renkl, & Schwonke, 2009). In this research project, we have taken this concern into account in the feedback design process and have attempted to explicitly address conceptual understanding. Here, the teacher does also play a valuable role. For example, like VanLehn (2008), we observed that students still ask help from instructors when working in a computer-based learning environment, but that conversations tend to focus more on

conceptual understanding. When the computer-based learning environment contains inspectable student models, these can be valuable starting points for conversations as well: discussing differences between the student's own knowledge estimations and the scores in the student models may be a fruitful teaching strategy to further evoke student reflection (Tacoma et al., 2017). Third, computer-based learning environments may benefit many, but not all students. Especially when machine-learning techniques are employed, decisions are informed by data and the majority tends to rule, which may result in a group of students who have difficulties in engaging with the online activities (Treviranus, 2018). In the results of this research project, this is reflected in the wide variety of feedback effects found. Learning analytics that the computer-based learning environment provides may help in identifying the students at risk (Tempelaar, Rienties, Mittelmeier, & Nguyen, 2018). The teacher may decide to especially provide guidance to these students while partly outsourcing teaching the majority of students to the computer-based learning environment. Future research is needed to further investigate the consequences of such a division of labor between teachers and computer-based learning environments for higher education.

In higher education, teachers are often not only responsible for teaching, but also for designing their courses. Due to time constraints and limited resources, feasibility is a major concern when teachers consider implementing computer-based learning activities. This concern is legitimate, given that the authoring time for one hour of instruction in adaptive educational systems has been estimated to range from 200 to 300 hours (Aleven, McLaren, Sewall, & Koedinger, 2006). In this research project, important decisions were informed by this feasibility concern, such as the decision to implement coarse-grained rather than fine-grained inspectable student models. Despite these choices, a large initial design effort was needed to implement the feedback. Further research and development should continue to strive for accessible authoring tools that can facilitate this design process for teachers and designers. After this initial effort, though, extending the design with new tasks was rather straightforward and, for the case of inner loop feedback, even easier than without the domain reasoner. Furthermore, after the initial design hurdle has been overcome and students have engaged with the online learning

material, analysis of student data can provide valuable information for improving the educational design. Reports that many computer-based learning environments generate can be helpful in checking item reliability and validity (Gilbert, Whitelock, & Gale, 2011). Based on our findings, we encourage the development of more advanced analytical tools that include results of learning curve analyses, in order to provide sophisticated insight into the quality of both tasks as well as (inspectable) student models in the system. Teachers should use such analysis tools together with their pedagogical content knowledge to improve the quality of the educational material.

6.5.4 New directions in statistics and statistics education

The third and final direction for further research concerns the ongoing development of statistics and, consequently, statistics education. In 2015, the editors of Basic and Applied Social Psychology announced that they were banning the null hypothesis significance testing procedure from their journal, because they regarded the underlying logic of this procedure as flawed (Trafimow & Marks, 2015). Although most journals do not go this far, hypothesis testing and *p*-values have been subjected to ongoing scientific debate and critique in the past decades (Wasserstein & Lazar, 2016). Concerns include that (1) important scientific conclusions and policy decisions are too often based on whether or not a *p*-value passes a specific threshold (typically 0.05); (2) *p*-values are often erroneously interpreted as the probability that the null hypothesis is true; and (3) statistical significance alone is often erroneously interpreted as being equivalent to the importance or relevance of an effect (Falk & Greenbaum, 1995; Wasserstein & Lazar, 2016). This discussion, together with the advancement of technology, provokes the invention and adoption of new statistical methods, which will have consequences for statistics education as well. After all, education largely influences what practitioners do, and what practitioners do should influence education.

Methods that have gained in popularity in recent years include Bayesian inference, which may offer an appropriate alternative to null hypothesis significance testing for assessing whether a hypothesis is supported by available data (Kaplan, 2015). Resampling methods, such as bootstrapping, have become more feasible and have, therefore, been

advocated to be addressed in statistics education as well (Cobb, 2007; Hesterberg, 2015). A development that even more strongly reflects the advancement of technology has been the emergence of the field of data science, and, consequently, data science courses and degree programs (Donoho, 2017). Typically, such data science programs combine elements of statistics programs and computer science programs and, for example, discuss machine-learning techniques for identifying patterns and making predictions based on data.

Designing appropriate automated intelligent feedback addressing these concerns regarding *p*-values and supporting these newer statistical methods is an important direction for further research. Here, we reflect on how the approaches designed in this research project could be adjusted to accommodate these developments. For inner loop feedback, supporting new methods is not easy. To provide feedback on the logical reasoning within a statistical procedure, inner loop feedback relies heavily on the assumptions underlying the specific procedure. Since different statistical methods involve different assumptions, extending our existing domain reasoner to support other statistical methods as well would require considerable extra design effort. Nonetheless, smaller adjustments to our domain reasoner can be envisioned to address the concerns regarding *p*-values outlined above and to implement recommendations in statistics education literature, such as the American Statistical Association's GAISE college report (Carver et al., 2016). In the design of the domain reasoner, some of the concerns were already addressed: its feedback specifically addressed common misinterpretations of the procedure of null hypothesis significance testing, such as making statements about the truth of an alternative hypothesis. However, the domain reasoner's task was finished once a conclusion about the null hypothesis was drawn.

In light of the concerns regarding *p*-values outlined above, a first possible improvement to the domain reasoner would be to extend its scope beyond this conclusion, by requiring an interpretation of the results in the context of the original research question and by including measures of effect size as well. Regarding effect size, caution should be taken not to limit this approach to calculating effect sizes and comparing them with traditional benchmark values, but to interpret these values within the

research context too (Bakker et al., 2019). Furthermore, in the current implementation students could freely choose how to find critical values and *p*-values: for example, by using online statistical tools or by looking them up in a table. Although this latter method is still common in education, it has become old-fashioned, does not reflect statistical practice anymore and should therefore be discouraged (Carver et al., 2016). Tools to be used in education for this part of the hypothesis-testing procedure ideally do not only provide values of test statistics and *p*-values, but also show visualizations to illustrate the relations between the *p*-value, critical value, significance level and distribution under the null hypothesis in the current problem situation. A final possible improvement of the domain reasoner relates to the tasks in which it can be used. All too often, tasks in introductory statistics courses focus on simple questions about how two groups differ or how two variables are correlated (Carver et al., 2016). Because of the domain reasoner's ability to carry out the hypothesis-testing procedure given an appropriate starting situation, it should be possible to extend its range of supported tasks to much more open tasks, for example within students' research projects. In this scenario, students would be able to investigate their own research questions, state their own hypotheses and collect their own data, after which the domain reasoner could support them in carrying out appropriate statistical analyses given these hypotheses and data.

Our implementation of outer loop feedback, in the form of inspectable student models, could more easily be adapted to accommodate new statistical methods. The concepts included in the student models may change, but student models remain valuable for supporting students in gaining insight into the structure of the domain and into their current knowledge of important concepts. A student model could also include multiple techniques at once, thus providing a basis for offering students practice with selecting appropriate techniques to address specific research questions, as the GAISE college report recommends (Carver et al., 2016). A design aspect for further consideration is that the conceptual knowledge structure in a domain may be obvious to experts, but is usually not evident for novices (Lovett, Meyer, & Thille, 2008). Furthermore, novices are likely to have informal knowledge about concepts such as chance, probability and hypothesis, which may contain misconceptions (Lovett & Greenhouse,

2000). New knowledge is likely to be linked to this prior knowledge, rather than to replace it. To address these considerations, inspectable student models could explicate relations between concepts and address possible informal notions of concepts, for example by including visualizations. To summarize, future research and educational design should strive to accommodate new statistical methods in automated intelligent feedback, while maintaining a focus on statistical proficiency and, especially, on the concepts rather than on the procedures and calculations involved.

6.6 Recommendations for educational practice

In the previous section, we pictured a long-range vision for the use of technology and artificial intelligence in education and we outlined directions for further research to develop towards this vision. On the basis of our research project, we can also formulate recommendations for the shorter term, about implementing automated intelligent feedback in education, in statistics and other domains.

First, when implementing automated feedback, we recommend paying attention to making the feedback findable and usable. In the implementation process, think of when and how students can find feedback. The lower the cost of seeking feedback in the learning environment – in terms of time, number of clicks and the need to ask help from peers or an instructor about where to find feedback – the more likely it becomes that the assumed value of the feedback outweighs the assumed cost. In our project, we believe that pages in the learning environment encouraging students to use the inspectable student models and clear instructions in the hypothesis-testing tasks have contributed to the findability of the implemented feedback.

Usability of the feedback is another important aspect to consider in feedback design: what can students do with the feedback after receiving it, to improve their work? This aspect is often forgotten in formative assessment (Nicol & Macfarlane-Dick, 2006). Inspectable student models can assist students in identifying topics to practice with, but to be able to practice, students need tasks or activities on these topics as well. Clear annotation of tasks with the topics involved may therefore support

students in using inspectable student models to their full potential. Inner loop feedback, as provided by our domain reasoner for hypothesis testing, can often be used more directly to improve the current step in a solution process. Still, the availability of enough tasks to practice, possibly with gradual fading of domain reasoner feedback, is necessary to provide students enough opportunities to make errors, learn from these errors through feedback and demonstrate this gained understanding by not making the same errors again.

Our second recommendation is to give students time to familiarize themselves with the computer-based learning environment and the feedback it provides. Initial confusion about how students are supposed to interact with the learning environment may result in erroneous answers that do not result from a lack of conceptual knowledge, but from a lack of knowledge of the system. Using these answers to inform a student model may result in unfairly low scores. Especially when students are given freedom to construct multi-step solutions, they may need time to understand what reasoning and syntax the system expects. Specifically, we advise teachers to use a system over the course of at least one semester, but preferably more, and to reserve time for familiarization with the affordances and constraints of a system throughout the course.

Our third and final recommendation for educational practice relates to time as well: allow enough time for implementation and improvement of automated intelligent feedback – or other technological enhancements – in education. In our educational innovation projects, we have experienced that initial design effort can be considerable. Content development and software development influence each other and need to be adjusted to one another, which takes time. In this phase, content quality can already benefit from availability of student models: scrutinizing connections between tasks and topics can serve to assess whether all topics are sufficiently addressed. Furthermore, pilots can be time-consuming, but are indispensable for finding out how students engage differently than anticipated with the software. After this initial effort, eventually the time will come that a large group of students works with the technology. And then, despite all the careful designing and piloting, unexpected things will happen. So, after the first implementation, time should be reserved for additional design effort.

Luckily, incredibly valuable information is usually available to inform this additional design: data about students' engagement with the technology. In this thesis we have described ways to use these data, combined with pedagogical content knowledge, to improve the quality of automated intelligent feedback and learning activities themselves.

To summarize, let us combine these recommendations for educational practice with the contributions of this research project (section 6.3) and the long-range vision for the use of technology and artificial intelligence in education outlined in section 6.5. From this, we obtain the following characteristics that, in our view, an implementation of automated intelligent feedback to support university students in developing statistical proficiency ideally should have:

- Tasks are didactically rich, meaning that they involve real or realistic contexts and data and provide plenty of opportunities for alternative answers;
- Both inner and outer loop feedback are offered, possibly in a variety of implementations, to cater to the needs of many different learners;
- Students are given time and guidance to learn how the feedback types can support them in their learning process;
- The material offers students opportunities to use the knowledge they have gained from the feedback. Navigation to useful tasks given the current state of the student model is facilitated, and tasks are available in which students can demonstrate their statistical proficiency;
- The homework tasks are aligned well with other curriculum elements and students receive clear information about assessment objectives;
- The teacher uses information obtained from the learning environment to support students who need extra guidance and to improve the learning material and the course.

Thus, we encourage further efforts to enrich (higher) education, both in statistics and in other domains, with intelligent feedback, while striving for didactically rich tasks and interactions as well. This is not always easy and requires a considerable amount of work. Yet, as we hope to have

demonstrated with this thesis, we strongly believe that this approach has the potential to provide many students with useful feedback on their learning process, and, hence, is worth the effort.

References

Aberson, C. L., Berger, D. E., Healy, M. R., & Romero, V. L. (2003). Evaluation of an interactive tutorial for teaching hypothesis testing concepts. *Teaching of Psychology, 30*(1), 75–78. doi:10.1207/S15328023TOP3001_12

Abraham, J. D., Burnett, D. D., & Morrison, J. D. J. (2006). Feedback seeking among developmental assessment center participants. *Journal of Business and Psychology, 20*(3), 383–394. doi:10.1007/s10869-005-9008-z

Aleven, V., McLaren, B., Sewall, J., & Koedinger, K. (2006). The cognitive tutor authoring tools (CTAT): Preliminary evaluation of efficiency gains. In M. Ikeda, K. Ashley & TW. Chan (Eds.), *Intelligent tutoring systems, LNCS 4053* (pp. 61–70). Berlin, Heidelberg, Germany: Springer. doi:10.1007/11774303_7

Al-Shanfari, L., Epp, C. D., & Baber, C. (2017). Evaluating the effect of uncertainty visualisation in open learner models on students' metacognitive skills. In E. André, R. Baker, X. Hu, M. Rodrigo, & B. du Boulay (Eds.), *Artificial Intelligence in Education, AIED 2017, LNCS 10331* (pp. 15–27). Cham, Switzerland: Springer. doi:10.1007/978-3-319-61425-0_2

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. *The Journal of the Learning Sciences, 4*(2), 167–207. doi:10.1207/s15327809jls0402_2

Ang, S., Cummings, L. L., Straub, D. W., & Earley, P. C. (1993). The effects of information technology and the perceived mood of the feedback giver on feedback seeking. *Information Systems Research, 4*(3), 240–261. doi:10.1287/isre.4.3.240

Anseel, F., Beatty, A. S., Shen, W., Lievens, F., & Sackett, P. R. (2015). How are we doing after 30 years? A meta-analytic review of the antecedents and outcomes of feedback-seeking behavior. *Journal of Management, 41*(1), 318–348. doi:10.1177/0149206313484521

Anseel, F., Lievens, F., & Levy, P. E. (2007). A self-motives perspective on feedback-seeking behavior: Linking organizational behavior and social psychology research. *International Journal of Management Reviews, 9*(3), 211–236. doi:10.1111/j.1468-2370.2007.00210.x

Antoniou, P., & James, M. (2014). Exploring formative assessment in primary school classrooms: Developing a framework of actions and

strategies. *Educational Assessment, Evaluation and Accountability, 26*(2), 153–176. doi:10.1007/s11092-013-9188-4

Arroyo, I., Ferguson, K., Johns, J., Dragon, T., Meheranian, H., Fisher, D., . . . Woolf, B. P. (2007). Repairing disengagement with non-invasive interventions. In R. Luckin, K. Koedinger, & J. Greer (Eds.), *Proceedings of the 2007 conference of artificial intelligence in education: Building technology-rich learning contexts that work* (pp. 195–202). Amsterdam, the Netherlands: ISO Press.

Ashford, S. J., & Black, J. S. (1996). Proactivity during organizational entry: The role of desire for control. *Journal of Applied Psychology, 81*(2), 199–214. doi:10.1037/0021-9010.81.2.199

Ashford, S. J., & Cummings, L. L. (1983). Feedback as an individual resource: Personal strategies of creating information. *Organizational Behavior and Human Performance, 32*(3), 370–398. doi:10.1016/0030-5073(83)90156-3

Bakker, A., Cai, J., English, L., Kaiser, G., Mesa, V., & Van Dooren, W. (2019). Beyond small, medium, or large: Points of consideration when interpreting effect sizes. *Educational Studies in Mathematics, 102*(1), 1–8. doi:10.1007/s10649-019-09908-4

Bakker, A., & Van Eerde, D. (2015). An introduction to design-based research with an example from statistics education. In A. Bikner-Ahsbahs, C. Knipping & N. C. Presmeg (Eds.), *doing qualitative research: Methodology and methods in mathematics education* (pp. 429–466). Dordrecht, the Netherlands: Springer. doi:10.1007/978-94-017-9181-6_16

Barnes, T. (2005). The q-matrix method: Mining student response data for knowledge. In *American Association for Artificial Intelligence 2005 Educational Data Mining Workshop,* 1–8.

Bennett, R. E. (2011). Formative assessment: A critical review. *Assessment in Education: Principles, Policy & Practice, 18*(1), 5–25. doi:10.1080/0969594X.2010.513678

Ben-Zvi, D. (2000). Toward understanding the role of technological tools in statistical learning. Mathematical Thinking and Learning, 2(1–2), 127–155. doi:10.1207/S15327833MTL0202_6

Birenbaum, M., DeLuca, C., Earl, L., Heritage, M., Klenowski, V., Looney, A., . . . Wyatt-Smith, C. (2015). International trends in the implementation of assessment for learning: Implications for policy and practice. *Policy Futures in Education*, *13*(1), 117–140. doi:10.1177/1478210314566733

Black, P., & Wiliam, D. (2012). Developing a theory of formative assessment. In J. Gardner (Ed.), *Assessment and learning* (pp. 206–229). London, UK: SAGE.

Bokhove, C., & Drijvers, P. H. M. (2012). Effects of feedback in an online algebra intervention. *Technology, Knowledge and Learning*, *17*(1–2), 43–59. doi: 10.1007/s10758-012-9191-8

Brusilovsky, P., & Millán, E. (2007). User models for adaptive hypermedia and adaptive educational systems. In P. Brusilovsky, A. Kobsa, & W. Nejdl (Eds.), *The adaptive web, LNCS 4321* (pp. 3–53). Berlin Heidelberg, Germany: Springer-Verlag. doi:10.1007/978-3-540-72079-9_1

Brusilovsky, P., Somyürek, S., Guerra, J., Hosseini, R., & Zadorozhny, V. (2015). The value of social: Comparing open student modeling and open social student modeling. In F. Ricci, K. Bontcheva, O. Conlan, & S. Lawless (Eds.), *User modeling, adaptation and personalization* (pp. 44–55). Cham, Switzerland: Springer International Publishing. doi:10.1007/978-3-319-20267-9_4

Brusilovsky, P., Sosnovsky, S., & Yudelson, M. (2009). Addictive links: The motivational value of adaptive link annotation. *New Review of Hypermedia and Multimedia, 15*(1), 97–118. doi: 10.1080/13614560902803570

Bull, S. (2004). Supporting learning with open learner models. In *Proceedings of 4th Hellenic Conference in Information and Communication Technologies in Education* (pp. 47–61). Athens, Greece.

Bull, S., & Kay, J. (2007). Student models that invite the learner in: The SMILI:() Open learner modelling framework. International Journal of Artificial Intelligence in Education, 17(2), 89–120.

Bull, S., & Kay, J. (2016). SMILI☺: A framework for interfaces to learning data in open learner models, learning analytics and related fields. *International Journal of Artificial Intelligence in Education*, *26*(1), 293–331. doi:10.1007/s40593-015-0090-8

Bull, S., Mabbott, A., Gardner, P., Jackson, T., Lancaster, M., Quigley, S., & Childs, P. (2008). *Supporting interaction preferences and recognition of misconceptions with independent open learner models. Adaptive hypermedia and adaptive web-based systems* (pp. 62–72). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-540-70987-9_9

Carr, B., & Goldstein, I. P. (1977). *Overlays: A theory of modelling for computer aided instruction.* Cambridge, MA: Massachusetts Institute of Technology, Artificial Intelligence Lab.

Carver, R., Everson, M., Gabrosek, J., Horton, N., Lock, R., Mocko, M., . . . Wood, B. (2016). Guidelines for assessment and instruction in statistics education (GAISE) college report 2016. American Statistical Association. *http://www.amstat.org/education/gaise*

Castro Sotos, A. E., Vanhoof, S., Van den Noortgate, W., & Onghena, P. (2007). Students' misconceptions of statistical inference: A review of the empirical evidence from research on statistics education. *Educational Research Review*, *2*(2), 98–113. doi:10.1016/j.edurev.2007.04.001

Chance, B., Ben-Zvi, D., Garfield, J., & Medina, E. (2007). The role of technology in improving student learning of statistics. *Technology Innovations in Statistics Education, 1*(1). Retrieved from *https://escholarship.org/uc/item/8sd2t4rr*

Chen, X., Breslow, L., & DeBoer, J. (2018). Analyzing productive learning behaviors for students using immediate corrective feedback in a blended learning environment. *Computers & Education, 117*, 59–74. doi:10.1016/j.compedu.2017.09.013

Chew, P. K. H., & Dillon, D. B. (2014). Statistics anxiety update. *Perspectives on Psychological Science, 9*(2), 196–208. doi:10.1177/1745691613518077

Cobb, G. W. (2007). The introductory statistics course: A ptolemaic curriculum? *Technology Innovations in Statistics Education, 1*(1). Retrieved from *http://escholarship.org/uc/item/6hb3k0nz*

Corbalan, G., Kester, L., & Van Merriënboer, J. J. (2006). Towards a personalized task selection model with shared instructional control. *Instructional Science, 34*(5), 399–422. doi:10.1007/s11251-005-5774-2

Corbett, A. T., & Anderson, J. R. (1995). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction, 4*(4), 253–278. doi:10.1007/BF01099821

Crommelinck, M., & Anseel, F. (2013). Understanding and encouraging feedback-seeking behavior: A literature review. *Medical Education, 47*(3), 232–241. doi:10.1111/medu.12075

Dimitrova, V., Self, J., & Brna, P. (2001). Applying interactive open learner models to learning technical terminology. In M. Bauer, P.J. Gmytrasiewicz, J. Vassileva (Eds.), *User Modeling 2001. UM 2001.*

LNCS 2109 (pp. 148–157). Berlin, Heidelberg, Germany: Springer. doi:10.1007/3-540-44566-8_15

Dirkx, K. J., Kester, L., & Kirschner, P. A. (2014). The testing effect for learning principles and procedures from texts. *The Journal of Educational Research, 107*(5), 357–364. doi:10.1080/00220671.2013.823370

Donoho, D. (2017). 50 years of data science. *Journal of Computational and Graphical Statistics, 26*(4), 745–766. doi:10.1080/10618600.2017.1384734

Drijvers, P. H. M., Boon, P. B. J., Doorman, L.M., Bokhove, C., & Tacoma, S. G. (2013). Digital design: RME principles for designing online tasks. In C. Margolinas (Ed.), *Proceedings of ICMI study 22 task Design in Mathematics Education* (pp. 55–62). Clermont-Ferrand, France: ICMI.

Evans, C. (2013). Making sense of assessment feedback in higher education. *Review of Educational Research, 83*(1), 70–120. doi:10.3102/0034654312474350

Evans, J. D. (1996). *Straightforward statistics for the behavioral sciences*. Pacific Grove, CA: Brooks/Cole Publishing.

Falk, R., & Greenbaum, C. W. (1995). Significance tests die hard: The amazing persistence of a probabilistic misconception. *Theory & Psychology, 5*(1), 75–98. doi:10.1177/0959354395051004

Farley-Ripple, E., May, H., Karpyn, A., Tilley, K., & McDonough, K. (2018). Rethinking connections between research and practice in education: A conceptual framework. *Educational Researcher, 47*(4), 235–245. doi: 10.3102/0013189X18761042

Field, A. (2009). *Discovering statistics using SPSS* (3rd ed.). London: Sage publications.

Garfield, J., & Ahlgren, A. (1988). Difficulties in learning basic concepts in probability and statistics: Implications for research. *Journal for Research in Mathematics Education, 19*(1), 44–63. doi:10.2307/749110

Garfield, J., Aliaga, M., Cobb, G., Cuff, C., Gould, R., Lock, R., . . . Utts, J. (2005). *Guidelines for assessment and instruction in statistics education (GAISE) college report 2005*. American Statistical Association. Retrieved from *http://www.amstat.org/education/gaise/GaiseCollege_Full.pdf*

Garfield, J., Ben-Zvi, D., Chance, B., Medina, E., Roseth, C., & Zieffler, A. (2008). Learning to reason about statistical inference. In J. Garfield & D. Ben-Zvi (Eds.), *Developing students' statistical reasoning* (pp. 261–288). Dordrecht, the Netherlands: Springer.

Gikandi, J. W., Morrow, D., & Davis, N. E. (2011). Online formative assessment in higher education: A review of the literature. *Computers & Education, 57*(4), 2333–2351. doi:10.1016/j.compedu.2011.06.004

Gilbert, L., Whitelock, D., & Gale, V. (2011). *Synthesis report on assessment and feedback with technology enhancement*. Southampton, UK: Electronics and Computer Science EPrints.

Gobert, J. D., Sao Pedro, M., Raziuddin, J., & Baker, R. S. (2013). From log files to assessment metrics: Measuring students' science inquiry skills using educational data mining. *Journal of the Learning Sciences, 22*(4), 521–563. doi:10.1080/10508406.2013.837391

Goguadze, G. (2011). *ActiveMath - generation and reuse of interactive exercises using domain reasoners and automated tutorial strategies* (Doctoral dissertation, Saarland University, Saarbrücken, Germany). Retrieved from *https://publikationen.sulb.uni-saarland.de/bitstream/20.500.11880/26153/1/goguadzeDiss2011.pdf*

Goguadze, G., & Melis, E. (2009). Combining evaluative and generative diagnosis in ACTIVEMATH. In V. Dimitrova, R. Mizoguchi, B. du Boulay & A. Graesser (Eds.), *Artificial Intelligence in Education* (pp. 668–670). doi:10.3233/978-1-60750-028-5-91

Harks, B., Rakoczy, K., Hattie, J., Besser, M., & Klieme, E. (2014). The effects of feedback on achievement, interest and self-evaluation: The role of feedback's perceived usefulness. *Educational Psychology, 34*(3), 269–290. doi:10.1080/01443410.2013.785384

Hattie, J., & Timperley, H. (2007). The power of feedback. *Review of Educational Research, 77*(1), 81–112. doi:10.3102/003465430298487

Hattie, J., & Yates, G. (2014). Using feedback to promote learning. In V. A. Benassi, C. E. Overson & C. M. Hakala (Eds.), *Applying science of learning in education: Infusing psychological science into the curriculum* (5th ed., pp. 45–58). Washington, DC: Society for the Teaching of Psychology.

Heeren, B. J., & Jeuring, J. T. (2014). Feedback services for stepwise exercises. *Science of Computer Programming*, 88, 110–129. doi:10.1016/j.scico.2014.02.021

Heitink, M. C., Van der Kleij, F., Veldkamp, B. P., Schildkamp, K., & Kippers, W. B. (2016). A systematic review of prerequisites for implementing assessment for learning in classroom practice. *Educational Research Review, 17*, 50–62. doi:10.1016/j.edurev.2015.12.002

Hendriks, M. A. (2014). *The influence on school size, leadership, evaluation, and time on student outcomes: Four reviews and meta-analyses.* (Doctoral dissertation, University of Twente, Enschede, the Netherlands). doi:10.3990/1.9789036538008

Herder, E., Sosnovsky, S., & Dimitrova, V. (2017). Adaptive intelligent learning environments. In E. Duval, M. Sharples & R. Sutherland (Eds.), *Technology enhanced learning* (pp. 109–114). Cham, Switzerland: Springer. doi:10.1007/978-3-319-02600-8_10

Hesterberg, T. C. (2015). What teachers should know about the bootstrap: Resampling in the undergraduate statistics curriculum. *The American Statistician, 69*(4), 371–386. doi:10.1080/00031305.2015.1089789

Hox, J. J., Moerbeek, M., & Van der Schoot, R. (2018). *Multilevel analysis: Techniques and applications* (3rd ed.). New York, NY: Routledge.

Kaplan, D. (2015). The future of quantitative inquiry in education: Challenges and opportunities. In M. Feuer, A. Berman & R. Atkinson (Eds.), *Past as prologue: The national academy of education at 50. Members reflect* (pp. 109–115). Washington, DC: National Academy of Education.

Kicken, W., Brand-Gruwel, S., & van Merriënboer, J. J. (2008). Scaffolding advice on task selection: A safe path toward self-directed learning in on-demand education. *Journal of Vocational Education and Training, 60*(3), 223–239. doi:10.1080/13636820802305561

Kinicki, A. J., Prussia, G. E., Wu, B. J., & McKee-Ryan, F. M. (2004). A covariance structure analysis of employees' response to performance feedback. *Journal of Applied Psychology, 89*(6), 1057–1069. doi:10.1037/0021-9010.89.6.1057

Kluger, A. N., & DeNisi, A. (1996). The effects of feedback interventions on performance. *Psychological Bulletin, 119*(2), 254–284. doi:10.1037/0033-2909.119.2.254

Kodaganallur, V., Weitz, R. R., & Rosenthal, D. (2005). A comparison of model-tracing and constraint-based intelligent tutoring paradigms. *International Journal of Artificial Intelligence in Education, 15*(2), 117–144.

Koedinger, K. R., Pavlik, P. I., Stamper, J., Nixon, T., & Ritter, S. (2011). Avoiding problem selection thrashing with conjunctive knowledge tracing. In M. Pechenizkiy, T. Calders, C. Conati, S. Ventura, C. Romero & J. Stamper (Eds.), *Proceedings of the 4th International Conference on Educational Data Mining* (pp. 91–100). Eindhoven, the Netherlands.

Krause, K., & Coates, H. (2008). Students' engagement in first-year university. *Assessment & Evaluation in Higher Education, 33*(5), 493–505. doi:10.1080/02602930701698892

Long, Y., & Aleven, V. (2011). Students' understanding of their student model. In G. Biswas, S. Bull, J. Kay, & A. Mitrovic (Eds.), *Artificial Intelligence in education, 15th international conference* (pp. 179–186). Berlin, Heidelberg: Springer-Verlag. doi:10.1007/978-3-642-21869-9_25

Lovett, M., & Greenhouse, J. (2000). Applying cognitive theory to statistics instruction. *The American Statistician, 54*(3), 196–206. doi: 10.1080/00031305.2000.10474545

Lovett, M., Meyer, O., & Thille, C. (2008). The open learning initiative: Measuring the effectiveness of the OLI statistics course in accelerating student learning. *Journal of Interactive Media in Education, 2008*(1), 13. doi:10.5334/2008-14

Martin, B., Mitrovic, A., Koedinger, K., & Mathan, S. (2011). Evaluating and improving adaptive educational systems with learning curves. *User Modeling and User-Adapted Interaction, 21*(3), 249–283. doi:10.1007/s11257-010-9084-2

McCambridge, J., Witton, J., & Elbourne, D. R. (2014). Systematic review of the hawthorne effect: New concepts are needed to study research participation effects. *Journal of Clinical Epidemiology, 67*(3), 267–277. doi:10.1016/j.jclinepi.2013.08.015

Mitrovic, A., Koedinger, K. R., & Martin, B. (2003). A comparative analysis of cognitive tutoring and constraint-based modeling. In P. Brusilovsky, A. Corbett & F. de Rosis (Eds.), *User Modeling 2003* (pp. 313–322). Berlin, Heidelberg, Germany: Springer.

Mitrovic, A., & Martin, B. (2002). Evaluating the effects of open student models on learning. In P. De Bra, P. Brusilovsky, & R. Conejo (Eds.), *Adaptive Hypermedia and Adaptive Web-Based Systems. AH 2002. LNCS 2347* (pp.296–305). Berlin, Heidelberg, Germany: Springer. doi:10.1007/3-540-47952-X_31

Mitrovic, A., & Martin, B. (2007). Evaluating the effect of open student models on self-assessment. *International Journal of Artificial Intelligence in Education, 17*, 121–144.

Mitrovic, A., Martin, B., & Suraweera, P. (2007). Intelligent tutors for all: The constraint-based approach. *IEEE Intelligent Systems, 4*, 38–45. doi:10.1109/MIS.2007.74

Morrison, E. W., & Cummings, L. L. (1992). The impact of feedback diagnosticity and performance expectations on feedback seeking behavior. *Human Performance, 5*(4), 251–264. doi:10.1207/s15327043hup0504_1

Murtonen, M., & Lehtinen, E. (2003). Difficulties experienced by education and sociology students in quantitative methods courses. *Studies in Higher Education, 28*(2), 171–185. doi:10.1080/0307507032000058064

Narciss, S., & Huth, K. (2004). How to design informative tutoring feedback for multimedia learning. In H. M. Niegemann, D. Leutner, & R. Brünken (Eds.), *Instructional Design for Multimedia Learning* (pp. 181–195). Münster, Germany: Waxmann.

Narciss, S., Sosnovsky, S., Schnaubert, L., Andrès, E., Eichelmann, A., Goguadze, G., & Melis, E. (2014). Exploring feedback and student characteristics relevant for personalizing feedback strategies. *Computers & Education, 71*, 56–76. doi:10.1016/j.compedu.2013.09.011

Nicol, D. J., & Macfarlane-Dick, D. (2006). Formative assessment and self-regulated learning: A model and seven principles of good feedback practice. *Studies in Higher Education, 31*(2), 199–218. doi:10.1080/03075070600572090

Nwana, H. S. (1990). Intelligent tutoring systems: An overview. *Artificial Intelligence Review, 4*(4), 251–277. doi:10.1007/BF00168958

Onwuegbuzie, A. J. (2004). Academic procrastination and statistics anxiety. *Assessment & Evaluation in Higher Education, 29*(1), 3–19. doi:10.1080/0260293042000160384

Paechter, M., Macher, D., Martskvishvili, K., Wimmer, S., & Papousek, I. (2017). Mathematics anxiety and statistics anxiety. Shared but also unshared components and antagonistic contributions to performance in statistics. *Frontiers in Psychology, 8*, 1196. doi:10.3389/fpsyg.2017.01196

Pardo, A. (2018). A feedback model for data-rich learning experiences. *Assessment & Evaluation in Higher Education, 43*(3), 428–438. doi:10.1080/02602938.2017.1356905

Pavlik, P. I., Cen, H., & Koedinger, K. R. (2009). Performance factors analysis--A new alternative to knowledge tracing. In V. Dimitrova, R. Mizoguchi, B. du Boulay & A. Graesser (Eds.) *Artificial Intelligence in Education. AIED 2009 Frontiers in Artificial Intelligence and Applications 200* (pp. 531–538). Brighton, UK: IOS Press.

Quellmalz, E. S., & Pellegrino, J. W. (2009). Technology and testing. *Science, 323*(5910), 75–79. doi:10.1126/science.1168046

Ritter, S., Anderson, J. R., Koedinger, K. R., & Corbett, A. (2007). Cognitive tutor: Applied research in mathematics education. *Psychonomic Bulletin & Review, 14*(2), 249–255. doi:10.3758/BF03194060

Robinson, J., Myran, S., Strauss, R., & Reed, W. (2014). The impact of an alternative professional development model on teacher practices in formative assessment and student learning. *Teacher Development, 18*(2), 141–162. doi:10.1080/13664530.2014.900516

Salden, R., Aleven, V., Renkl, A., & Schwonke, R. (2009). Worked examples and tutored problem solving: Redundant or synergistic forms of support? *Topics in Cognitive Science, 1*(1), 203–213. doi:10.1111/j.1756-8765.2008.01011.x

Santos, G. S., & Jorge, J. (2013). Interoperable intelligent tutoring systems as open educational resources. *IEEE Transactions on Learning Technologies, 6*(3), 271–282. doi:10.1109/TLT.2013.17

Shaffer, J. P. (1995). Multiple hypothesis testing. *Annual Review of Psychology, 46*(1), 561–584. doi:10.1146/annurev.ps.46.020195.003021

Shute, V. J. (2008). Focus on formative feedback. *Review of Educational Research, 78*(1), 153–189. doi:10.3102/0034654307313795

Sosnovsky, S. A., & Brusilovsky, P. (2015). Evaluation of topic-based adaptation and student modeling in QuizGuide. *User Modeling and User-Adapted Interaction, 25*(4), 371–424. doi:10.1007/s11257-015-9164-4

Stacey, K., & Wiliam, D. (2013). Technology and assessment in mathematics. In M. A. Clements, A. J. Bishop, C. Keitel, J. Kilpatrick & F. K. S. Leung (Eds.), *Third International Handbook of Mathematics Education* (pp. 721–751). New York: Springer. doi:10.1007/978-1-4614-4684-2_23

Steenbergen-Hu, S., & Cooper, H. (2014). A meta-analysis of the effectiveness of intelligent tutoring systems on college students' academic learning. *Journal of Educational Psychology, 106*(2), 331–347. doi:10.1037/a0034752

Taber, K. S. (2018). The use of Cronbach's alpha when developing and reporting research instruments in science education. *Research in Science Education, 48*(6), 1273–1296. doi:10.1007/s11165-016-9602-2

Tacoma, S. G., Drijvers, P. H. M., & Boon, P. B. J. (2017). Using student models to generate feedback in a university course on statistical sampling. In T. Dooley & G. Gueudet (Eds.), *Proceedings of the Tenth Congress of the European Society for Research in Mathematics Education* (pp. 844–851). Dublin, Ireland: DCU Institute of Education and ERME.

Tacoma, S. G., Heeren, B. J., Jeuring, J. T., & Drijvers, P. H. M. (2019). Automated feedback on the structure of hypothesis tests. . In U. T. Jankvist, M. van den Heuvel-Panhuizen, & M. Veldhuis (Eds.), *Proceedings of the Eleventh Congress of the European Society for Research in Mathematics Education* (pp. 2969–2976). Utrecht, the Netherlands: Freudenthal Group & Freudenthal Institute, Utrecht University and ERME.

Tacoma, S. G., Sosnovsky, S. A., Boon, P. B. J., Jeuring, J. T., & Drijvers, P. H. M. (2018). The interplay between inspectable student models and didactics of statistics. *Digital Experiences in Mathematics Education, 4*(2–3), 139–162. doi:10.1007/s40751-018-0040-9

Tatsuoka, K. K. (1983). Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of Educational Measurement, 20*(4), 345–354. doi:10.1111/j.1745-3984.1983.tb00212.x

Tempelaar, D. T., Rienties, B., & Giesbers, B. (2015). In search for the most informative data for feedback generation: Learning analytics in a data-rich context. *Computers in Human Behavior, 47*, 157–167. doi:10.1016/j.chb.2014.05.038

Tempelaar, D., Rienties, B., Mittelmeier, J., & Nguyen, Q. (2018). Student profiling in a dispositional learning analytics application using formative assessment. *Computers in Human Behavior, 78*, 408–420. doi:10.1016/j.chb.2017.08.010

Timmers, C. F., Braber-van den Broek, J., & Van den Berg, S. (2013). Motivational beliefs, student effort, and feedback behaviour in

computer-based formative assessment. *Computers & Education, 60*(1), 25–31. doi:10.1016/j.compedu.2012.07.007

Tishkovskaya, S., & Lancaster, G. A. (2012). Statistical education in the 21st century: A review of challenges, teaching innovations and strategies for reform. *Journal of Statistics Education, 20*(2), 1–55. doi:10.1080/10691898.2012.11889641

Torenbeek, M., Jansen, E., & Suhre, C. (2013). Predicting undergraduates' academic achievement: The role of the curriculum, time investment and self-regulated learning. *Studies in Higher Education, 38*(9), 1393–1406. doi:10.1080/03075079.2011.640996

Trafimow, D., & Marks, M. (2015). Editorial. *Basic and Applied Social Psychology, 37*(1), 1–2. doi:10.1080/01973533.2015.1012991

Treviranus, J. (2018). Learning differences and digital equity in the classroom. In J. Voogt, G. Knezek, R. Christensen & KW. Lai (Eds.), *Second handbook of information technology in primary and secondary education* (pp. 1025–1046). Cham, Switzerland: Springer. doi:10.1007/978-3-319-71054-9_74

Tuckey, M., Brewer, N., & Williamson, P. (2002). The influence of motives and goal orientation on feedback seeking. *Journal of Occupational and Organizational Psychology, 75*(2), 195–216. doi:10.1348/09631790260098677

United Nations Statistics Division (2014, January 29). *Fundamental Principles of Official Statistics.* Retrieved from *https://unstats.un.org/unsd/dnss/hb/E-fundamental%20principles_A4-WEB.pdf*

Vallecillos, A. (1999). Some empirical evidences on learning difficulties about testing hypotheses. *Bulletin of the International Statistical Institute: Proceedings of the Fifty-Second Session of the International Statistical Institute, 58*, 201–204.

Van der Kleij, F., Feskens, R., & Eggen, T. (2015). Effects of feedback in a computer-based learning environment on students' learning outcomes. A meta-analysis. *Review of Educational Research, 85*(4), 475–511. doi:10.3102/0034654314564881

Van der Kleij, F., Timmers, C. F., & Eggen, T. (2011). The effectiveness of methods for providing written feedback through a computer-based assessment for learning: A systematic review. *Cadmo, 1*, 21–38. doi:10.3280/CAD2011-001004

VandeWalle, D., & Cummings, L. L. (1997). A test of the influence of goal orientation on the feedback-seeking process. *Journal of Applied Psychology, 82*(3), 390–400. doi:10.1037/0021-9010.82.3.390

Vanderlinde, R., & van Braak, J. (2010). The gap between educational research and practice: Views of teachers, school leaders, intermediaries and researchers. *British Educational Research Journal, 36*(2), 299–316. doi:10.1080/01411920902919257

VanLehn, K. (2006). The behavior of tutoring systems. *International Journal of Artificial Intelligence in Education, 16*(3), 227–265.

VanLehn, K. (2008). Intelligent tutoring systems for continuous, embedded assessment. In C. A. Dwyer (Ed.), *The future of assessment: Shaping teaching and learning* (pp. 113–138). New York, NY: Erlbaum.

VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist, 46*(4), 197–221. doi:10.1080/00461520.2011.611369

Wasserstein, R. L., & Lazar, N. A. (2016). The ASA's statement on p-values: Context, process, and purpose. *The American Statistician, 70*(2), 129–133. doi:10.1080/00031305.2016.1154108

Zakharov, K., Mitrovic, A., & Ohlsson, S. (2005). Feedback micro-engineering in EER-tutor. In C.-K. Looi, G. McCalla, B. Bredeweg, & J. Breuker (Eds.), *Proceedings of the 12th International Conference on Artificial Intelligence in Education: Supporting Learning through Intelligent and Socially Informed Technology* (pp. 718–725). Amsterdam, the Netherlands: IOS Press.

Zapata-Rivera, D., & Greer, J. E. (2002). Exploring various guidance mechanisms to support interaction with inspectable learner models. In S. Cerri, G. Gouardères, & F. Paraguaçu (Eds.), *Intelligent Tutoring Systems. ITS 2002. LNCS 2363* (pp. 442–452). Berlin Heidelberg, Germany: Springer. doi: 10.1007/3-540-47987-2_47

Summary

Nederlandse samenvatting (summary in Dutch)

Dankwoord (acknowledgements in Dutch)

Curriculum Vitae

Publications related to this thesis

Presentations related to this thesis

Summary

Due to technological advancements in the past decades, in particular the emergence of powerful digital tools to collect, store, analyze and represent big datasets, the field of statistics is changing rapidly. Because most calculations can now be outsourced to calculators and computers, introductory university statistics courses are changing as well: the focus is shifting from manipulating statistical formulas and carrying out statistical techniques to knowing and being able to reason with the underlying concepts and principles. Students need to develop statistical proficiency: knowledge of why data and statistical formulas are needed, how these can inform decisions, and how variability in data can affect the results of statistical techniques.

Developing statistical proficiency is challenging for students, though. Students struggle to build logical chains of reasoning involving many abstract statistical concepts, such as probability distributions and sampling variability. Individual guidance and feedback might be a means to support students in facing these challenges, but in the context of higher education – with typically large student group sizes – this is difficult for teachers to achieve. In this thesis, a solution to this issue is sought in the provision of automated intelligent feedback. The aim of this research project, therefore, was to design and evaluate automated intelligent feedback in a computer-based learning environment that addresses the difficulties that social sciences students experience in developing statistical proficiency. The guiding research question was:

> *How can automated intelligent feedback support first-year university students in developing statistical proficiency?*

Two types of automated intelligent feedback were designed and implemented in online homework sets within two introductory statistics courses for students enrolled in social sciences bachelor programs at Utrecht University. The online homework sets were offered in the Digital Mathematics Environment (DME). Tasks in the homework sets addressed, for example, selecting appropriate measures of center and spread for given variables and testing hypotheses for given situations and samples. Students received immediate verification feedback in all tasks,

informing them whether their answer was correct, but not of what the correct answer was. Additionally, the two types of automated intelligent feedback were offered. Inner loop feedback addressed the students' steps in carrying out hypothesis tests, one of the most important techniques in many introductory statistics courses in higher education. Outer loop feedback provided students with overviews of their current understanding of important statistical concepts. For addressing our research question, three aspects of feedback implementation were deemed important: feedback design, students' use of the feedback, and effects of the feedback on the students' statistical proficiency. These aspects align well with characteristics of design-based research, in which theoretical ideas about student learning inform the design. Subsequently, the theoretical ideas are adapted, informed by the implementation and evaluation of the design. In this research project, four design research cycles were carried out to address the three implementation aspects for both inner and outer loop feedback. In the following, we summarize the findings from these four cycles, which are discussed in detail in Chapters 2 to 5 of this thesis.

In Chapter 2 we discuss the first cycle concerning inner loop feedback, in the form of a domain reasoner for hypothesis testing. In this cycle, the design, use by students, and direct effects of the inner loop feedback were addressed. This chapter concerns a randomized controlled trial with 314 first-year psychology students. In six of the homework tasks, 163 of these students received domain reasoner feedback: every time they added a step to their hypothesis-testing procedure, the domain reasoner checked the new partial solution and provided feedback. This feedback did not only address the correctness of the current partial solution, but also explained why it was correct or incorrect. The other 151 students only received feedback on the correctness of the steps they added to their solution in these six tasks, without further elaboration. As such, Chapter 2 addresses the following research question:

2.1 Does automated intelligent feedback about the logic of hypothesis testing contribute to student proficiency in carrying out hypothesis tests?

The design of the domain reasoner combined characteristics of two prevailing paradigms in research on Intelligent Tutoring Systems (ITSs): constraint-based modeling and model-tracing. The designed domain reasoner contains a collection of constraints that a (partial) solution should satisfy, as well as a collection of expert rules and buggy rules that determine how one (partial) solution can be obtained from another. The constraints allow the system to identify missing elements and inconsistencies in students' solutions, while the rules enable addressing common errors and providing hints for subsequent steps. To address the research question, *t*-tests and multilevel regression models served to compare the numbers of attempted and solved hypothesis-testing tasks and the numbers of errors between the two groups of students. We conclude that, after a familiarization phase, the intelligent feedback effectively supported students in solving tasks on hypothesis testing. Additionally, the number of errors that students made in their logical reasoning decreased more strongly over tasks for students who received feedback from the domain reasoner than for students who received verification feedback only. This suggests that the intelligent feedback fostered student proficiency in independently carrying out hypothesis tests. These positive effects did, however, not transfer to follow-up tasks about hypothesis testing.

Whereas the implementation of inner loop feedback concerned only a limited number of tasks, the outer loop feedback, in the form of inspectable student models, made use of evidence from almost all tasks in the online homework sets. Informed by didactical recommendations from statistics education research, these tasks were clustered around real datasets and contexts and were consciously sequenced. Designing inspectable student models based on such clusters of tasks, as opposed to mutually independent tasks, was regarded as an important design challenge. In the exploratory study described in Chapter 3, we investigated the feasibility and validity of implementing inspectable student models in this different instructional design. DME log files and questionnaire results from 160 first-year students in educational studies were used to address the following research questions:

3.1 Are inspectable student models suitable for implementation in didactically grounded, sequential statistics modules consisting of closely related tasks?

3.2 How can didactical analysis inform design of inspectable student models and, vice versa, how can student model evaluation methods inform didactical design?

From a student perspective, the answer to research question 3.1 was positive. The students valued the student models for their clarity and close connections to the tasks. The learning curve analysis that we used to assess the internal validity of the student models revealed a different picture, however: for almost half of the knowledge components in the student models the number of errors students made was found to increase, rather than decrease, over time. Hence, additional effort was needed to design well-defined, valid student models for these didactically grounded, sequential homework sets on statistics. This additional effort consisted of complementing the learning curve analysis with didactical task analysis and, informed by these analyses, redesigning the student models and tasks. Concerning research question 3.2, both analysis techniques proved valuable in this process. For knowledge components with increasing error rates over time, the learning curves were scrutinized and the tasks connected to the knowledge components were analyzed to find possible causes of the increasing error rates. Four problems causing increasing error rates were identified and are briefly discussed here.

The first two problems that caused increasing error rates concerned the design of the student models. The first one was that definitions of some knowledge components were too broad, meaning that in fact they constituted more than a single statistical concept. In the redesign, such knowledge components were split into two or more knowledge components. The second problem was that mistakes students made in a task affected the student model scores for all knowledge components connected to the task. In many cases, only one of these knowledge components was problematic for solving the task and, hence, student model scores for the other connected knowledge components became unreasonably low. To resolve these issues, in the redesign some connections between tasks and

non-problematic knowledge components were removed. Remarkably, the third and fourth problems causing increasing error rates did not concern the design of student models, but the design of the tasks in the homework sets. The third problem was that some tasks were found to be didactically meaningful for the homework set, but not suitable for informing student models. This concerned, for example, easy introductory tasks, or tasks to illustrate a specific characteristic of a concept. In the redesign, connections between these tasks and knowledge components were removed, since these tasks did not provide useful information for the student models. The fourth and final problem was the most striking example of how didactical analysis and student model evaluation methods can strengthen each other: together they revealed that some tasks related to difficult concepts addressed these concepts too superficially. Error rates were very low in these cases, meaning that students barely made errors regarding these difficult concepts. We concluded that the designed tasks in these cases did not provide students with enough opportunities to make mistakes and to learn from these mistakes. In this way, learning curve analysis did not only disclose weaknesses in student model design, but also in the design of the tasks in the homework sets themselves. For these tasks, redesign focused on creating more opportunities to reason with the difficult concepts and to make mistakes in this reasoning.

Based on the findings presented in Chapter 3, in the next cycle the homework sets and inspectable student models were revised, taking the various roles and goals of the tasks in the homework sets into account. Chapter 4 focuses on the students' use of the student models (i.e., their feedback-seeking behavior) and on the extent to which these student models informed the students' choices of subsequent study steps (i.e., their decision-making behavior). This was done in an exploratory study with 599 participants, guided by the following research questions:

4.1 *How do first-year university students in social science seek feedback from inspectable student models in an introductory statistics course?*

4.2 *How does feedback from inspectable student models inform these students' decisions about subsequent actions?*

4.3 How does these students' feedback-seeking and decision-making behavior relate to performance on a statistics exam?

Concerning question 4.1, students were found to keep consulting their student models throughout the course period, albeit slightly less frequent towards the end of the course. A wide variety in timing, duration and amount of detail of student model views was found, both between students as well as between different student model views by the same student. This variety could be explained from differences in underlying student self-motives. For example, for quickly verifying one's weaker and stronger topics, a shorter and less detailed student model view suffices than would do for consciously planning subsequent study steps. A similar variety was found regarding research question 4.2 concerning the student decisions about subsequent actions, based on the feedback from their inspectable student models. Across this diversity, though, students seemed inclined to improve their work on the homework sets when student model scores were low, and to work on extra practice tasks or new topics when student model scores were high. This suggests that the inspectable student models may have encouraged students to devote more effort to their homework sets than they would have done without having the student models available. Concerning research question 4.3, based on a multiple linear regression model we conclude that both the frequency of student model viewing and the amount of variety in decisions made after viewing the student model were positively related to exam results. Not surprisingly, student activity, as measured by time on task, played a role in these relationships: students who spent much time in the learning environment tended to view their student models often and also tended to score high on the final exam. Yet, our findings suggest that other factors, such as the students' self-motives mentioned above, seem to be important as well. Although the absence of a control group in this study prevented us from drawing causal inferences, these findings suggest that frequently inspecting student models and using them to inform subsequent study steps can be a fruitful learning strategy.

Teachers and educational designers are inclined to combine different promising approaches for delivering rich, inspiring courses. Therefore, many ITSs offer both inner loop feedback, to support students while working on

specific tasks, and outer loop feedback, to guide students in their learning process. After having discussed feedback design and students' use of the designed feedback for both inner and outer loop feedback separately, in Chapter 5 we turn to the effects of their combination on students' statistical proficiency. In a randomized controlled trial with 521 participants (first-year social sciences students) and a factorial 2x2 design (inner loop feedback vs. no inner loop feedback and outer loop feedback vs. no outer loop feedback), the effects of both feedback types and their interaction on the students' learning processes and course performance were evaluated. The research question for this evaluation was:

> 5.1 *What effects does providing both inner and outer loop feedback on online homework have on students' learning process and course performance in a university statistics course?*

As this research question indicates, in Chapter 5 we do not only focus on feedback effects on students' statistical proficiency, but also on effects that offering both inner and outer loop feedback have on the students' learning processes. This choice was motivated by the view that feedback is a process, in which students should actively engage in order to benefit from it. It also allowed us to verify and corroborate findings from earlier cycles. Like in earlier cycles, multiple linear regression models were used to assess the influence of both feedback types, as well as student-specific characteristics, on the students' learning process and course performance. The effects found for inner loop feedback were indeed similar to those discussed in Chapter 2: students receiving inner loop feedback performed better on hypothesis-testing tasks, but only if they had prior experience with the domain reasoner. No effects of domain reasoner availability on course performance were found, but we did find small effects of student model availability on course performance. As in Chapter 4, these effects varied widely between students: students with low prior performance seemed to benefit from the student models, while students with high prior performance seemed to be hindered by them. We also found a small effect of student model availability on the time students worked in the ITS, with students receiving outer loop feedback working slightly longer. Finally, inner and outer loop feedback did not influence each other's effects

on course performance, but did compete for attention in the students' learning processes: students who used the ITS intensively tended to use the outer loop feedback less if they also received inner loop feedback.

All in all, this thesis sheds light on the design and implementation of automated intelligent feedback to support university students in developing statistical proficiency. The field of artificial intelligence in education provided excellent starting points for designing both inner and outer loop feedback. We have explored ways to use artificial intelligence techniques in combination with didactically rich tasks that cluster around real or realistic contexts and data. Furthermore, we have demonstrated how information obtained from the learning environment can be used to improve both the feedback as well as the tasks in the learning materials. We have found that many students can benefit from the designed feedback, especially when they are given enough time to get used to them. When both inner and outer loop feedback are offered within one learning environment, they do not seem to influence each other's effects on students' performance, but may slightly compete for the students' attention in the learning process. Given the variety of feedback effects for different students, we still regard implementing both feedback types as an added value: it increases the number of students who receive feedback that they perceive as helpful.

Further research and educational design are needed to develop the feedback to its full potential. We especially encourage further research addressing three questions: the validity of inferences about student knowledge from practicing tasks that provide immediate feedback, the role of the teacher when implementing substantial computer-based elements in a (statistics) curriculum, and ongoing developments in the field of statistics, which affect statistics education as well. Developing this vision is a long-term goal. In addition, we provide some short-term recommendations. First, we recommend making sure students can find and use the feedback. Our second recommendation is to give students time to familiarize themselves with the feedback. The third and final recommendation is to also provide teachers and educational designers with enough time, both for implementing the feedback as well as for redesigning the feedback, informed by data about the students' engagement with the feedback. We acknowledge that implementing intelligent feedback, especially in

combination with open-ended and didactically rich tasks, requires a considerable amount of work. Yet, we strongly believe that this approach has the potential to provide many students with useful feedback on their learning process, and, hence, we conclude that it is worth the effort.

Nederlandse samenvatting (summary in Dutch)

Door technologische ontwikkelingen is het in de afgelopen decennia mogelijk geworden om grote dataverzamelingen aan te leggen, op te slaan, te analyseren en te visualiseren. Hierdoor verandert de manier waarop we statistiek gebruiken in hoog tempo. Omdat we berekeningen en procedures steeds meer kunnen overlaten aan rekenmachines en computers, verandert ook het statistiekonderwijs op de universiteit: waar de focus vroeger vaak lag op het afleiden en manipuleren van statistische formules en technieken, ligt deze tegenwoordig steeds meer op kennis van en redeneren met de onderliggende statistische begrippen en principes. Studenten moeten statistische bekwaamheid ontwikkelen: weten waarom we data en statistische formules gebruiken, hoe deze kunnen helpen bij het nemen van beslissingen en hoe variantie in data de uitkomsten van statistische procedures kan beïnvloeden. Het ontwikkelen van deze statistische bekwaamheid is niet eenvoudig. Studenten vinden het moeilijk om logisch te redeneren met abstracte statistische begrippen zoals kansverdelingen en steekproefvariantie. Individuele begeleiding en feedback zou hen hierbij kunnen helpen, maar het bieden van deze begeleiding is voor docenten in het hoger onderwijs, waarin studentencohorten vaak groot zijn, meestal niet haalbaar. In dit proefschrift wordt een oplossing gezocht in het aanbieden van automatische intelligente feedback. Het doel van dit onderzoek is het ontwikkelen en evalueren van automatische intelligente feedback in een digitale leeromgeving, gericht op de problemen die studenten sociale wetenschappen ervaren bij het ontwikkelen van statistische bekwaamheid. De onderzoeksvraag voor dit project luidt:

> *Hoe kan automatische intelligente feedback eerstejaarsstudenten aan de universiteit helpen bij het ontwikkelen van statistische bekwaamheid?*

In dit project hebben we twee typen automatische intelligente feedback ontworpen en geïmplementeerd in twee introductiecursussen statistiek voor eerstejaarsstudenten sociale wetenschappen aan de Universiteit Utrecht. In deze cursussen werkten de studenten elke week aan huiswerkopgaven, die werden aangeboden in de Digitale Wiskunde Omgeving (DWO). In deze opgaven moesten studenten bijvoorbeeld geschikte centrum- en

spreidingsmaten selecteren voor gegeven variabelen en voerden ze hypothesetoetsen uit bij gegeven contexten en steekproeven. In alle opgaven ontvingen de studenten directe verificatiefeedback, die aangaf of hun antwoord goed of fout was. Daarbij werd niet aangegeven wat het juiste antwoord was. Naast deze verificatiefeedback ontvingen ze ook de twee typen automatische intelligente feedback. De lokale feedback ging over de stappen die studenten maakten bij het toetsen van hypothesen, één van de belangrijkste technieken in veel introductiecursussen statistiek. De globale feedback bestond uit overzichten waarin werd ingeschat in welke mate de student de belangrijke statistische concepten begreep. Om de onderzoeksvraag te kunnen beantwoorden hebben we ons gericht op drie aspecten van het implementeren van feedback: feedbackontwerp, het gebruik van de feedback door studenten en de effecten van de feedback op hun statistische bekwaamheid. Deze aspecten passen goed bij de kenmerken van ontwerpgericht onderwijsonderzoek, waarin theorieën over het leren van studenten een basis vormen voor het te ontwerpen onderwijs. Vervolgens worden deze theorieën verder aangescherpt op basis van de implementatie en evaluatie van het onderwijsontwerp. In dit onderzoeksproject hebben we vier cycli van ontwerp en implementatie uitgevoerd. In deze cycli kwamen de drie bovengenoemde implementatieaspecten aan bod voor zowel lokale als globale feedback. Hieronder vatten we de bevindingen uit de vier cycli, die uitgebreid worden beschreven in de hoofdstukken 2 tot en met 5 van dit proefschrift, kort samen.

In hoofdstuk 2 behandelen we het ontwerp, gebruik en de directe effecten van de lokale feedback: een *domain reasoner* voor hypothesetoetsen. In dit hoofdstuk bespreken we een gerandomiseerd experiment waaraan 314 eerstejaars psychologiestudenten deelnamen. Van hen ontvingen 163 studenten in zes van de huiswerkopgaven stapsgewijze feedback van de domain reasoner: telkens als zij een nieuwe stap aan hun hypothesetoets toevoegden, werd deze nieuwe deeloplossing gecontroleerd en van feedback voorzien. Daarin werd niet alleen de juistheid van de huidige deeloplossing aangegeven, maar ook uitgelegd waarom deze deeloplossing goed of fout was. De andere 151 studenten kregen in deze zes opgaven alleen maar verificatiefeedback op de juistheid van de stappen die ze maakten, zonder verdere uitleg. Hiermee konden we in hoofdstuk 2 de volgende onderzoeksvraag centraal stellen:

2.1 Draagt automatische intelligente feedback over de logica van hypothesetoetsen bij aan de bekwaamheid van studenten in het uitvoeren van hypothesetoetsen?

In het ontwerp van de domain reasoner hebben we twee gangbare modelleermethoden uit onderzoek naar Intelligente Tutorsystemen (ITS'en) gecombineerd: *constraint-based modeling*, oftewel, een model gebaseerd op voorwaarden, en *model-tracing*, een model gebaseerd op regels. De ontworpen domain reasoner bevat daarom zowel een verzameling voorwaarden waaraan een correcte deeloplossing moet voldoen, als een verzameling regels (zowel juiste als foutieve) die aangeven hoe een deeloplossing uit een vorige deeloplossing verkregen kan worden. Met behulp van de voorwaarden kan het systeem bepalen of een deeloplossing onderdelen mist of tegenstrijdigheden bevat, terwijl de regels het mogelijk maken om veelgemaakte fouten op te sporen en hints te geven voor vervolgstappen. Voor het beantwoorden van de onderzoeksvraag hebben we het aantal geprobeerde en opgeloste opgaven en het aantal fouten dat studenten in de twee groepen maakten geanalyseerd met behulp van *t*-toetsen en multilevel regressiemodellen. Op basis hiervan hebben we geconcludeerd dat studenten even tijd nodig hadden om te wennen aan de feedback, maar dat ze er daarna bij het oplossen van opgaven over hypothesetoetsen profijt van hadden. Beide groepen studenten maakten in latere opgaven minder fouten in de logische structuur van hun hypothesetoetsen, maar dit aantal daalde sneller bij de studenten die feedback ontvingen van de domain reasoner dan bij de studenten die alleen feedback over de juistheid van hun stappen kregen. Het lijkt er dus op dat de intelligente feedback bijdroeg aan de bekwaamheid van de studenten in het uitvoeren van hypothesetoetsen. Deze positieve effecten zagen we echter niet terug in latere opgaven over hypothesetoetsen, waarin studenten geen feedback van de domain reasoner meer ontvingen.

De implementatie van lokale feedback betrof slechts een beperkt aantal opgaven. Voor de implementatie van globale feedback, in de vorm van *inspecteerbare studentmodellen*, werd daarentegen informatie gebruikt uit bijna alle online huiswerkopgaven. Op basis van didactische aanbevelingen uit onderzoek naar statistiekonderwijs waren deze huiswerkopgaven gegroepeerd rondom echte dataverzamelingen en contexten en waren

ze zorgvuldig geordend. Omdat in veel ITS'en de gebruikte opgaven juist onderling onafhankelijk zijn, beschouwden we het ontwerpen van inspecteerbare studentmodellen voor zulke gegroepeerde opgaven als een belangrijke uitdaging in het ontwerp. In hoofdstuk 3 onderzoeken we hoe haalbaar en valide het is om inspecteerbare studentmodellen te implementeren bij zo'n andere structuur van de opgaven. In dit exploratieve deelonderzoek, waaraan 160 eerstejaarsstudenten pedagogiek en onderwijskunde deelnamen, hebben we logbestanden uit de DWO en resultaten van een vragenlijst gebruikt om de volgende onderzoeksvragen te beantwoorden:

3.1 Kunnen inspecteerbare studentmodellen op zinvolle wijze geïmplementeerd worden in didactisch gefundeerde statistiekmodules die bestaan uit groepen nauw verwante opgaven?

3.2 Hoe kan didactische analyse het ontwerp van inspecteerbare studentmodellen informeren en, omgekeerd, hoe kunnen evaluatiemethoden voor studentmodellen het didactische ontwerp informeren?

Vanuit studentperspectief was het antwoord op onderzoeksvraag 3.1 positief. De studenten vonden de studentmodellen duidelijk en zagen goed het verband tussen de opgaven en de studentmodellen. Voor het toetsen van de interne validiteit van de studentmodellen hebben we leercurves geanalyseerd (*learning curve analysis*) en dat leverde een ander beeld op: voor bijna de helft van de kenniscomponenten in de studentmodellen maakten de studenten na verloop van tijd niet minder, maar meer fouten. Er was dus nog een stap nodig om goed gedefinieerde, valide studentmodellen te ontwerpen voor deze didactisch gefundeerde huiswerkopgaven over statistiek. In deze extra stap hebben we de analyse van leercurves aangevuld met een didactische analyse van de opgaven, om op basis daarvan de studentmodellen en huiswerkopgaven te herontwerpen. Met betrekking tot onderzoeksvraag 3.2 concluderen we dat beide technieken waardevol waren in dit proces. Om mogelijke oorzaken te vinden voor het stijgende aantal fouten bij veel kenniscomponenten, hebben we zowel de leercurves als de aan de kenniscomponenten verbonden opgaven

nauwgezet bestudeerd. Dit leverde vier mogelijke oorzaken op, die we hieronder kort behandelen.

De eerste twee oorzaken hadden betrekking op het ontwerp van de studentmodellen. De eerste was dat de definities van sommige kenniscomponenten eigenlijk meerdere statistische begrippen tegelijk betroffen. In het herontwerp hebben we zulke kenniscomponenten gesplitst in twee of meer kenniscomponenten. De tweede oorzaak was dat fouten die studenten maakten in een opgave werden toegeschreven aan alle kenniscomponenten die aan die opgave verbonden waren. In veel gevallen leek echter slechts één van de verbonden kenniscomponenten daadwerkelijk een probleem te vormen voor het correct oplossen van de opgave. De studentmodelscores van de andere verbonden kenniscomponenten werden dus ten onrechte verlaagd. Om dit probleem op te lossen is in het herontwerp een aantal verbindingen tussen opgaven en niet-problematische kenniscomponenten verwijderd. Opmerkelijk genoeg lagen de derde en vierde oorzaak voor stijgende aantallen fouten niet in het ontwerp van de studentmodellen, maar in het ontwerp van de huiswerkopgaven. De derde oorzaak was dat sommige opgaven een didactische functie hadden binnen de gehele module, maar niet geschikt waren om een studentmodel op te baseren. Dit betrof bijvoorbeeld eenvoudige introductieopgaven, of opgaven waarin studenten een heel specifieke eigenschap van een statistisch begrip moesten gebruiken. In het herontwerp werden de verbindingen tussen deze opgaven en kenniscomponenten verwijderd, omdat deze opgaven geen zinvolle informatie leverden voor de studentmodellen. De vierde en laatste oorzaak laat het best zien hoe didactische analyse en evaluatiemethoden voor studentmodellen elkaar kunnen versterken: door deze combinatie ontdekten we dat studenten moeilijke begrippen in sommige opgaven slechts heel oppervlakkig hoefden te gebruiken. In de leercurves zagen we dat studenten heel weinig fouten maakten met deze begrippen, die we wel als moeilijk beschouwden. Hieruit concludeerden we dat de ontworpen opgaven de studenten niet genoeg mogelijkheden boden om fouten te maken en te leren van deze fouten. Op deze manier heeft de analyse van leercurves niet alleen zwakke punten in het ontwerp van de studentmodellen blootgelegd, maar ook in het ontwerp van de huiswerkopgaven zelf. Het herontwerp van deze opgaven richtte zich op

het creëren van meer mogelijkheden om met deze moeilijke begrippen te redeneren en daar fouten in te maken.

Op basis van de resultaten uit hoofdstuk 3 zijn in de volgende cyclus de huiswerkopgaven en inspecteerbare studentmodellen herzien, rekening houdend met de verschillende rollen en doelen van de huiswerkopgaven. Hoofdstuk 4 richt zich op hoe de studenten de studentmodellen gebruikten (zoekgedrag naar feedback) en de mate waarin de studentmodellen richting gaven aan de verdere studieactiviteiten van de studenten (besluitgedrag op basis van feedback). Dit gebeurde in een exploratief deelonderzoek met 599 deelnemers, aan de hand van de volgende onderzoeksvragen:

4.1 Hoe zoeken eerstejaarsstudenten sociale wetenschappen aan de universiteit feedback die wordt gegeven door inspecteerbare studentmodellen in een introductiecursus statistiek?

4.2 Hoe geeft feedback van inspecteerbare studentmodellen richting aan de besluiten van studenten over verdere studieactiviteiten?

4.3 Hoe hangen het zoekgedrag naar feedback en het besluitgedrag op basis van feedback van deze studenten samen met hun prestaties op een statistiektentamen?

Met betrekking tot onderzoeksvraag 4.1 zagen we dat studenten hun studentmodellen door de cursus heen regelmatig bleven bekijken, al deden ze dat tegen het einde van de cursusperiode iets minder vaak dan aan het begin. De momenten waarop en de mate van detail waarin studenten hun studentmodellen bekeken varieerden sterk, net als de tijd die ze eraan besteedden. Deze verscheidenheid zagen we niet alleen tussen verschillende studenten, maar ook tussen de verschillende keren dat dezelfde student zijn of haar studentmodellen bekeek. Een mogelijke verklaring hiervoor ligt in de onderliggende motieven van studenten. Om snel te controleren wat op dit moment sterke en zwakke punten zijn, volstaat het bijvoorbeeld om veel korter en minder gedetailleerd naar een studentmodel te kijken dan om bewust volgende studiestappen te plannen. Ook de besluiten die studenten op basis van hun studentmodellen namen over verdere studieactiviteiten

(onderzoeksvraag 4.2) waren zeer divers. Wel zagen we een algemene trend: bij lage scores in het studentmodel waren studenten vaak geneigd hun werk op de huiswerkopgaven te verbeteren, terwijl ze bij hoge scores vaak besloten aan extra oefenopgaven of aan opgaven over een ander onderwerp te gaan werken. Het lijkt er dus op dat de studentmodellen studenten hebben aangemoedigd om meer tijd aan hun huiswerkopgaven te besteden dan ze anders zouden hebben gedaan. Over onderzoeksvraag 4.3 concluderen we op basis van een meervoudig lineair regressiemodel dat zowel de frequentie waarmee studenten hun studentmodellen bekeken als de verscheidenheid in besluiten die ze op basis van hun studentmodellen namen samenhangen met tentamenresultaten. Zoals verwacht speelde studentactiviteit, gemeten als de tijd die studenten doorbrachten in de DWO, een rol in deze samenhang: studenten die veel tijd doorbrachten in de leeromgeving waren geneigd vaak naar hun studentmodellen te kijken en haalden ook vaak een hoge score op het tentamen. Toch laten onze resultaten zien dat ook andere factoren, zoals de zelf-motieven die we hierboven noemden, belangrijk lijken te zijn. Omdat er in deze deelstudie geen controlegroep was, kunnen we geen uitspraken doen over oorzaak en gevolg. Toch lijken de resultaten erop te duiden dat het voor studenten zinvol is om regelmatig hun studentmodellen te bekijken en op basis hiervan op verschillende momenten voor verschillende vervolgactiviteiten te kiezen.

Docenten en onderwijsontwerpers combineren graag verschillende veelbelovende aanpakken om inspirerend onderwijs te geven. Daarom bieden veel ITS'en zowel lokale feedback, om studenten te ondersteunen tijdens het werken aan specifieke opgaven, als globale feedback, om studenten te begeleiden in het gehele leerproces. Waar we tot nu toe het ontwerp en gebruik van lokale en globale feedback afzonderlijk hebben besproken, richten we ons in hoofdstuk 5 op de effecten van de combinatie van beide feedbacktypen op de statistische bekwaamheid van studenten. In een gerandomiseerd experiment met 521 studenten (eerstejaarsstudenten sociale wetenschappen) en een 2x2 factorieel ontwerp (wel/geen lokale feedback en wel/geen globale feedback) werden de effecten van beide feedbacktypen en hun interactie op het leerproces en de prestaties van de studenten geëvalueerd. De onderzoeksvraag voor deze evaluatie was:

5.1 Welke effecten heeft het aanbieden van zowel lokale als globale feedback op online huiswerk in een universitaire statistiekcursus op het leerproces en de leerresultaten van studenten?

Zoals aan deze onderzoeksvraag te zien is, richten we ons in hoofdstuk 5 niet alleen op de effecten van de feedback op de statistische bekwaamheid van de studenten, maar ook op de effecten van beide feedbacktypen op de leerprocessen van de studenten. Deze keuze is ingegeven door de theorie dat feedback een proces is waaraan studenten actief moeten deelnemen om er profijt van te kunnen hebben. Ook geeft deze keuze ons de gelegenheid resultaten uit eerdere cycli te verifiëren. Ook in deze cyclus werd gebruik gemaakt van meervoudige lineaire regressiemodellen om zowel de invloed van beide feedbacktypen als van student-specifieke eigenschappen op het leerproces en de leerresultaten vast te kunnen stellen. De effecten die we vonden voor lokale feedback leken inderdaad sterk op die uit hoofdstuk 2: studenten die lokale feedback ontvingen maakten de opgaven over hypothesetoetsen beter, maar alleen als ze al ervaring hadden met de feedback van de domain reasoner. De domain reasoner had geen effect op de uiteindelijke leerresultaten van de studenten, terwijl de studentmodellen wel een klein effect hadden. Net als in hoofdstuk 4 varieerden deze effecten sterk tussen studenten. In het bijzonder leken zwakkere studenten profijt te hebben van de studentmodellen, terwijl sterkere studenten met studentmodellen juist minder goed presteerden dan zonder. Studenten die studentmodellen tot hun beschikking hadden werkten in totaal ook iets langer in de DWO dan studenten die dat niet hadden. Tot slot vonden we geen aanwijzingen dat de twee feedbacktypen elkaars effecten op de leerresultaten beïnvloedden. Wel zagen we dat studenten die beide feedbacktypen tot hun beschikking hadden wat minder gebruik maakten van de globale feedback dan studenten die alleen globale feedback ontvingen.

Al met al geeft dit proefzicht inzicht in het ontwerp en de implementatie van automatische intelligente feedback om eerstejaarsstudenten te helpen bij het ontwikkelen van statistische bekwaamheid. Onderzoek naar kunstmatige intelligentie in onderwijs bood een goed gefundeerde basis voor het ontwerp van zowel lokale als globale feedback. We hebben

mogelijkheden verkend om kunstmatige intelligentietechnieken in te zetten in combinatie met didactisch onderbouwde opgaven die zijn gegroepeerd rondom echte contexten en data. Ook hebben we laten zien hoe informatie die de leeromgeving verzamelt, kan worden gebruikt om niet alleen de ontwikkelde feedback, maar ook de opgaven zelf te verbeteren. We hebben gezien dat veel studenten profijt kunnen hebben van de ontwikkelde feedback, zeker wanneer ze genoeg tijd krijgen om er vertrouwd mee te raken. Het aanbieden van zowel lokale als globale feedback lijkt de effecten van beide niet te verminderen, maar ze lijken elkaar wel enigszins te beconcurreren om de aandacht van de student. Omdat de effecten van beide feedbacktypen sterk verschillen per student, beschouwen we het aanbieden van beide feedbacktypen toch als heel waardevol: hierdoor ontvangen meer studenten feedback die ze nuttig vinden.

Er is meer onderzoek en onderwijsontwerp nodig om deze feedbacktypen verder te ontwikkelen. In het bijzonder moedigen we onderzoek in de volgende drie richtingen aan: de validiteit van het inschatten van kennis op basis van pogingen op oefenopgaven, waarin studenten ook direct feedback krijgen; de rol van de docent in een (statistiek)cursus die aanzienlijke digitale elementen bevat; en de continue ontwikkelingen op het gebied van statistiek, die ook gevolgen hebben voor statistiekonderwijs. Naast deze suggesties voor een langetermijnvisie voor het gebruik van intelligente feedback, doen we ook aanbevelingen voor de korte termijn. Ten eerste bevelen we aan om te zorgen dat studenten feedback kunnen vinden en gebruiken. Onze tweede aanbeveling is om studenten tijd te geven om vertrouwd te raken met de feedback. De derde en laatste aanbeveling is om ook docenten en onderwijsontwikkelaars voldoende tijd te geven, voor zowel het implementeren van feedback als voor het herontwerpen van feedback op basis van gegevens uit de leeromgeving over de interactie van studenten met de feedback. We erkennen dat het implementeren van intelligente feedback, zeker in combinatie met open en didactisch gefundeerde opgaven, een flinke tijdsinvestering vraagt. Toch zijn we ervan overtuigd dat deze aanpak de potentie heeft om veel studenten bruikbare feedback op hun leerproces te geven en daarom concluderen we dat het deze grote investering waard is.

Dankwoord (acknowledgements in Dutch)

In de eerste jaren dat ik bij het Freudenthal Instituut werkte, had ik soms mijn twijfels over promoveren. Leerzaam was het vast, maar zou het ook leuk zijn? Inmiddels, vijf jaar later, heb ik gewerkt aan een project waarin ik mijn eigen keuzes kon maken, met veel verschillende mensen kon samenwerken aan het ontwikkelen van onderwijs, analyseren van data en schrijven van artikelen, en ontzettend veel geleerd heb over statistiek, automatische feedback en onderwijskundig onderzoek doen. En nu kan ik volmondig zeggen: ja, een promotietraject kan ook heel leuk zijn! Daarom wil ik op deze plek graag een aantal mensen bedanken voor deze leerzame en leuke tijd.

Allereerst, Paul. Tussen al mijn twijfels over promoveren stond één ding al snel vast: als ik ging promoveren, dan graag bij jou. Dat jij hoogleraar werd, was dus ook goed nieuws voor mij. Vanaf het brainstormen over het promotieplan tot het afronden van het proefschrift kon ik altijd bij je terecht met vragen, twijfels en nieuwe ideeën. Jouw deskundige en pragmatische advies hielp me altijd weer op weg. Voor onze wekelijkse besprekingen heb ik regelmatig gedacht dat we niet veel te bespreken hadden, om vervolgens ruim een halfuur later vol nieuwe ideeën jouw kamer weer uit te lopen. Veel dank ook voor de ruimte die je me gaf om mijn eigen richting te kiezen in dit onderzoek. Johan, als tweede promotor bekeek je mijn onderzoek van iets verder weg. Toch was ook jij altijd bereikbaar voor vragen en voor het bekijken van de stukken die ik schreef. Dank voor al je scherpe commentaar en deskundige advies over intelligente tutorsystemen.

Ook de docenten uit het USO-project, Adriaan, Corine, Jeltje, Rutger en Yolanda, ben ik heel dankbaar. Ik kijk met plezier terug op de werkochtenden en -middagen waarin we statistiekopgaven ontwikkelden in de DWO. Bedankt voor de tijd die jullie namen om me het statistiekonderwijs in jullie opleidingen te laten zien en voor de moeite die jullie deden om te zorgen dat ik pilots en experimenten uit kon voeren in jullie vakken. In het bijzonder wil ik Jeltje bedanken voor alle ruimte die je me hebt gegeven om, vaak last-minute, aanpassingen te maken aan jouw cursusmateriaal en al je hulp bij het indelen van de studenten en verzamelen van

gegevens. En natuurlijk wil ik alle studenten bedanken die toestemming hebben gegeven voor het gebruik van hun werk en tentamenresultaten in mijn onderzoek. Ook Henk en Susanne wil ik bedanken voor de fijne samenwerking bij het ontwikkelen van het statistiekmateriaal in de DWO. Susanne, in het bijzonder bedankt voor de vele discussies over statistiek en de bijbehorende statistiekgrappen.

Tijdens mijn onderzoek heb ik, naast mijn promotoren, met een aantal andere onderzoekers samen kunnen werken. Wouter en Arthur, jullie hebben gedurende mijn promotietraject regelmatig meegedacht over plannen voor experimenten en analyses. Dank voor al jullie advies over implementatie van feedback, gebruik van terminologie en mogelijke analysemethoden. Nathalie, dank voor al je advies over het Engels in mijn artikelen en de opmaak van mijn proefschrift. Corine, jij was niet alleen docent in mijn onderzoeksproject, maar werd ook onderzoeker. Ik heb veel geleerd van de manier waarop jij naar onze data keek en vond het traject waarin jouw masterthesis en ons gezamenlijke artikel tot stand zijn gekomen niet alleen leerzaam, maar ook heel gezellig. Bastiaan, Bert and Sergey, you have all been co-authors for one of my articles. I would like to thank you for all your advice about literature and theory, and the time you took to help me in drafting and structuring the articles. Bastiaan, jou wil ik daarnaast ook bedanken voor je onvermoeibare werk aan de domain reasoner. We hebben vele uren samen achter de computer gezeten en het is eigenlijk jammer dat dat nu niet meer nodig is.

Peter, ook jij was coauteur van één van mijn artikelen, maar bovenal ben je natuurlijk de drijvende kracht achter de DWO (nu Numworx). Bedankt voor al het vertrouwen en de vrijheid die jij me de afgelopen negen jaar hebt gegeven in het ontwikkelen van nieuw materiaal en nieuwe componenten. De manier waarop jij altijd nieuwe mogelijkheden ziet voor het gebruik van ICT in didactisch rijk wiskundeonderwijs is heel inspirerend voor me geweest. De rest van het DWO/Numworx-team, Gert, Huub, Mieke, Rudi, Sylvia en Wim, wil ik graag bedanken voor jullie inzet om de DWO geschikt te maken voor het verzamelen van data voor mijn onderzoek en voor jullie interesse in en het meedenken over mijn onderzoek.

Naast collega's die min of meer direct bij mijn onderzoek betrokken waren, wil ik ook graag mijn andere collega's bij het Freudenthal Instituut bedanken. Fellow PhFIs, I really enjoyed being part of this group of PhD candidates at the Freudenthal Institute. The PhD meetings, workshops and weekends we had were useful, but most of all a lot of fun. Suzanne en Winnifred, jullie twee wil ik in het bijzonder bedanken voor onze talloze gesprekken over onderzoek en over de rest van het leven. Ik ben heel blij dat jullie mijn paranimfen willen zijn. Gjalt, ik ben blij dat we alle interne verhuizingen hebben doorstaan en ik ga de gezelligheid van kamer 3.64 missen. Michiel en Rogier, bij jullie kon ik altijd terecht voor een praatje en om vorderingen te delen, bedankt voor jullie enthousiasme. Alle collega's die me voor 2015 aangemoedigd hebben om aan een promotietraject te beginnen, onder anderen Christine, Joke, Peter, Toine en Wouter: dank voor deze aansporing. Alle collega's die kwamen lunchen als ik mijn rondje langs de kamers maakte (gelukkig te veel om hier op te noemen): bedankt voor de leuke lunchgesprekken. Carolien en Mara, ik zie jullie ook als collega's: het was fijn om de laatste anderhalf jaar een beetje samen op te trekken in het afronden van onze promotietrajecten. En alle collega's die bij de onderzoeksbesprekingen, in de gangen, bij de koffieautomaat, tijdens de Nationale Wiskunde Dagen of waar dan ook interesse toonden in mijn onderzoek: bedankt voor de fijne tijd.

Ook de wereld buiten het Freudenthal Instituut heeft geweten dat ik met een promotieonderzoek bezig was. Dit geldt zeker voor mijn medeorganisatoren van de European Girls' Mathematical Olympiad, Birgit, Ellen, Jetze en Quintijn: bedankt voor jullie steun en het opvangen van mijn taken toen dat nodig was, maar bovenal voor het enthousiasme en de gezelligheid bij het organiseren van zo'n groot evenement. Ook de mensen waarmee ik in de Euclidesredactie zat, in het bijzonder Gert, wil ik graag bedanken voor hun interesse in mijn onderzoek door de jaren heen.

Een promotietraject is extra leuk als er ook buiten werktijden genoeg leuke dingen gebeuren. Lieve badmintonners van Hercules, ik ben ontzettend blij om wekelijks met jullie op en naast de baan te staan. Lieve vrienden, bedankt voor alle etentjes, spelletjes-, puzzel- en filmavonden, koffieochtenden en boswandelingen de afgelopen jaren, ik hoop dat er nog vele volgen. Lieve Meindert, Gini, Lotte, Jasper, Noor, Loura, Jorrit, Siemen

en Sarina: wat fijn om een schoonfamilie te hebben die zo goed is in het vieren van de leuke dingen in het leven. Lieve papa, mama, Marten, Sigrid, Janneke en Harm: bedankt dat jullie me altijd, in alles, aanmoedigen en steunen. En lieve opa, jouw aansporing dat toch iemand van jouw F2 wel moest kunnen promoveren heeft me regelmatig geholpen om door te zetten. Dank voor je dappere pogingen om mijn onderzoek te begrijpen.

Tot slot, mijn eigen Vlammen. Lieve Leonie en Mette, toen ik begon met mijn promotieonderzoek waren jullie er nog niet en nu zijn jullie niet meer weg te denken, met al jullie verhalen, grapjes en knuffels. Lieve Mart, jij had natuurlijk al een fantastisch voorbeeld voor me neergezet met jouw promotieonderzoek. Met jouw buitenstaandersblik en interesse heb je me vaak geholpen om op een andere manier naar mijn data en onderzoek te kijken. Maar bovenal is het heel fijn om samen met jou plannen te maken en uit te voeren, te kijken naar twee kleine meisjes, en gewoon samen te zijn. Bedankt dat je er altijd voor me bent. Mijn conclusie is dus: ja, een promotietraject kan heel leuk zijn. Maar het haalt het niet bij thuiskomen bij jullie drie.

Curriculum Vitae

Sietske Tacoma was born on June 18, 1987 in Borne, the Netherlands. After completing her secondary education at the Gymnasium Apeldoorn in 2005, she enrolled in the bachelor programs Mathematics and Physics at Utrecht University. She obtained her bachelor's degree in Mathematics, with a minor in Physics, cum laude in 2008 and continued with the master program Science Education and Communication. For her master's thesis, which was supervised by Peter Boon and Paul Drijvers from the Freudenthal Institute, Utrecht University, she designed and evaluated algebra tasks for first-year chemistry students in the Freudenthal Institute's Digital Mathematics Environment. After completing this thesis, Sietske obtained her master's degree cum laude in 2010.

In 2011, Sietske started as a researcher and educational designer at the Freudenthal Institute. She taught several courses on mathematics and mathematics education and obtained her basic qualification education (Dutch: basiskwalificatie onderwijs) in 2015. Over the years, she learned to write code in Java and became a software developer of the Digital Mathematics Environment as well.

In 2015, Sietske started her PhD research on automated feedback in higher statistics education. This research project was connected to the project Innovative remedial digital learning module for statistics, which was supported by Utrecht University's Education Incentive Fund. This project was led by Johan Jeuring and involved teachers and students from three faculties within Utrecht University. In addition to conducting her research, Sietske was also responsible for designing educational material and supporting teachers in using the Digital Mathematics Environment.

Besides her work at the Freudenthal Institute, Sietske has been involved in the organization of the Dutch Mathematics Olympiad. She taught and supervised Dutch Olympiad students from 2007 until 2013

and has been one of the organizers of the European Girls' Mathematical Olympiad 2020 in the Netherlands.

After finishing her PhD in 2020, Sietske will start as a data science and statistics teacher at the Utrecht University of Applied Sciences. Sietske lives together with her husband Mart and two daughters Leonie and Mette.

Publications related to this thesis

Tacoma, S. G., Drijvers, P. H. M., & Jeuring, J. T. (2020). Combined inner and outer loop feedback in an intelligent tutoring system for statistics in higher education. *Journal of Computer Assisted Learning*, 1–14. doi: 10.1111/jcal.12491

Tacoma S. G., Geurts, C., Slof, B., Jeuring, J. T., & Drijvers, P. H. M. (2020). Enhancing learning with inspectable student models. *Computers in Human Behavior, 107,* 106276. doi: 10.1016/j.chb.2020.106276

Tacoma, S. G., Heeren, B. J., Jeuring, J. T., & Drijvers, P. H. M. (2020). Intelligent feedback on hypothesis testing. *International Journal of Artificial Intelligence in Education*. doi: 10.1007/s40593-020-00218-y

Tacoma, S.G., Heeren, B. J., Jeuring, J. T., & Drijvers, P. H. M. (2019). Automated feedback on the structure of hypothesis tests. In S. Isotani, E. Millán, A. Ogan, P. Hastings, B. McLaren, & R. Luckin (Eds.), Artificial Intelligence in Education. AIED 2019. LNCS 11626. Cham, Switzerland: Springer.

Tacoma, S. G., Heeren, B. J., Jeuring, J. T., & Drijvers, P. H. M. (2019). Automated feedback on the structure of hypothesis tests. In U. Jankvist, M. van den Heuvel-Panhuizen, & M. Veldhuis (Eds.), *Proceedings of the Eleventh Congress of the European Society for Research in Mathematics Education* (pp. 2969–2976). Utrecht, the Netherlands: Freudenthal Group & Freudenthal Institute, Utrecht University and ERME.

Tacoma, S. G., Sosnovsky, S. A., Boon, P. B. J., Jeuring, J. T., & Drijvers, P. H. M. (2018). The interplay between inspectable student models and didactics of statistics. *Digital Experiences in Mathematics Education, 4*(2-3), 139–162. doi: 10.1007/s40751-018-0040-9

Tacoma, S. G. (2017). Het Fizier gericht op studentmodellen voor gepersonaliseerde feedback. *Euclides*, *93*(2), 42–43.

Tacoma, S. G., Drijvers, P. H. M., & Boon, P. B. J. (2017). Using student models to generate feedback in a university course on statistical sampling. In T. Dooley & G. Gueudet (Eds.), *Proceedings of the Tenth Congress of the European Society for Research in Mathematics Education* (pp. 844–851). Dublin, Ireland: DCU Institute of Education and ERME.

Presentations related to this thesis

Tacoma, S. G., Heeren, B. J., Jeuring, J. T., & Drijvers, P. H. M. (2019). *Test yourself: is your hypothesis test correct?* Interactive event presented at the 20th International Conference on Artificial Intelligence in Education, Chicago, United States of America, June 25-29.

Tacoma, S. G., Heeren, B. J., Jeuring, J. T., & Drijvers, P. H. M. (2019). *Automated feedback on the structure of hypothesis tests.* Poster presented at the 20th International Conference on Artificial Intelligence in Education, Chicago, United States of America, June 25-29.

Drijvers, P. H. M. & Tacoma, S. G. (2019). *Trends and topics in next decade's mathematics education.* Presentation at the 4TU.AMI conference Mathematics Education: the next decade and beyond, Utrecht, the Netherlands, April 11-12.

Tacoma, S. G., Heeren, B. J., Jeuring, J. T., & Drijvers, P. H. M. (2019). *Automated feedback on the structure of hypothesis tests.* Paper presented at the Eleventh Congress of the European Society for Research in Mathematics Education, Utrecht, the Netherlands, February 6-10.

Tacoma, S. G. (2019). *Automatische feedback in universitair statistiekonderwijs.* Presentation at the Nationale Wiskunde Dagen, Veldhoven, the Netherlands, February 1-2.

Tacoma, S. G. (2018). *Blended and personalized statistics education.* Presentation at the Teaching and Learning Lab Herfstfestival (Autumn festival), Utrecht, the Netherlands, November 30.

Tacoma, S. G., Jeuring, J. T., Heeren, B. J., & Drijvers, P. H. M. (2018) *Automated feedback on the structure of hypothesis tests.* Poster presented at Onderwijs meets Onderzoek, Utrecht, the Netherlands, October 11.

Tacoma, S. G. (2017). *Feedback op hypothesetoetsen in de Digitale Wiskunde Omgeving.* Presentation at Noordhoff Uitgevers Wiskundecongres (Noordhoff Publishers Mathematics Congress), Nieuwegein, the Netherlands, November 23.

Tacoma, S. G., Boon, P. B. J., Drijvers, P. H. M. (2017). *Using student models to generate feedback in a university course on statistical sampling.* Paper

presented at the Tenth Congress of the European Society for Research in Mathematics Education, Dublin, Ireland, February 1-5.

Tacoma, S. G., Drijvers, P. H. M., Jeuring, J. T. (2016). *Automated feedback in higher statistics education.* Poster presented at the International Congress on Mathematical Education, Hamburg, Germany, July 24-31.

Tacoma, S. G., Drijvers, P. H. M., Jeuring, J. T. (2016). *Automated feedback in higher statistics education.* Poster presented at Onderwijs meets Onderzoek, Utrecht, the Netherlands, June 20.

FI Scientific Library

(formerly published as CD-b Scientific Library)

104 Zanten, M. van (2020). *Opportunities to learn offered by primary school mathematics textbooks in the Netherlands*

103. Walma, L. (2020). *Between Morpheus and Mary: The Public Debate on Morphine in Dutch Newspapers, 1880-1939*

102. Van der Gronde, A.G.M.P. (2019). *Systematic Review Methodology in Biomedical Evidence Generation.*

101. Klein, W. (2018). *New Drugs for the Dutch Republic. The Commodification of Fever Remedies in the Netherlands (c. 1650-1800).*

100. Flis, I. (2018). *Discipline Through Method - Recent history and philosophy of scientific psychology (1950-2018).*

99. Hoeneveld, F. (2018). *Een vinger in de Amerikaanse pap. Fundamenteel fysisch en defensie onderzoek in Nederland tijdens de vroege Koude Oorlog.*

98. Stubbé-Albers, H. (2018). *Designing learning opportunities for the hardest to reach: Game-based mathematics learning for out-of-school children in Sudan.*

97. Dijk, G. van (2018). *Het opleiden van taalbewuste docenten natuurkunde, scheikunde en techniek: Een ontwerpgericht onderzoek*.

96. Zhao, Xiaoyan (2018). *Classroom assessment in Chinese primary school mathematics education.*

95. Laan, S. van der (2017). *Een varken voor iedereen. De modernisering van de Nederlandse varkensfokkerij in de twintigste eeuw.*

94. Vis, C. (2017). *Strengthening local curricular capacity in international development cooperation.*

93. Benedictus, F. (2017). *Reichenbach: Probability & the A Priori. Has the Baby Been Thrown Out with the Bathwater?*

92. Ruiter, Peter de (2016). *Het Mijnwezen in Nederlands-Oost-Indië 1850-1950.*

91. Roersch van der Hoogte, Arjo (2015). *Colonial Agro-Industrialism. Science, industry and the state in the Dutch Golden Alkaloid Age, 1850-1950.*

90. Veldhuis, M. (2015). *Improving classroom assessment in primary mathematics education.*

89. Jupri, Al (2015). *The use of applets to improve Indonesian student performance in algebra.*

88. Wijaya, A. (2015). *Context-based mathematics tasks in Indonesia: Toward better practice and achievement.*

87. Klerk, S. (2015). *Galen reconsidered. Studying drug properties and the foundations of medicine in the Dutch Republic ca. 1550-1700.*
86. Krüger, J. (2014). *Actoren en factoren achter het wiskundecurriculum sinds 1600.*
85. Lijnse, P.L. (2014). *Omzien in verwondering. Een persoonlijke terugblik op 40 jaar werken in de natuurkundedidactiek.*
84. Weelie, D. van (2014). *Recontextualiseren van het concept biodiversiteit*.
83. Bakker, M. (2014). *Using mini-games for learning multiplication and division: a longitudinal effect study.*
82. Ngô Vũ Thu Hăng (2014). *Design of a social constructivism-based curriculum for primary science education in Confucian heritage culture*.
81. Sun, L. (2014). *From rhetoric to practice: enhancing environmental literacy of pupils in China*.
80. Mazereeuw, M. (2013). *The functionality of biological knowledge in the workplace. Integrating school and workplace learning about reproduction*.
79. Dierdorp, A. (2013). *Learning correlation and regression within authentic contexts.*
78. Dolfing, R. (2013). *Teachers' Professional Development in Context-based Chemistry Education. Strategies to Support Teachers in Developing Domain-specific Expertise*.
77. Mil, M.H.W. van (2013). *Learning and teaching the molecular basis of life*.
76. Antwi, V. (2013). *Interactive teaching of mechanics in a Ghanaian university context*.
75. Smit, J. (2013). *Scaffolding language in multilingual mathematics classrooms*.
74. Stolk, M. J. (2013). *Empowering chemistry teachers for context-based education. Towards a framework for design and evaluation of a teacher professional development programme in curriculum innovations*.
73. Agung, S. (2013). *Facilitating professional development of Madrasah chemistry teachers. Analysis of its establishment in the decentralized educational system of Indonesia*.
72. Wierdsma, M. (2012). *Recontextualising cellular respiration*.
71. Peltenburg, M. (2012). *Mathematical potential of special education students.*
70. Moolenbroek, A. van (2012). *Be aware of behaviour. Learning and teaching behavioural biology in secondary education.*
69. Prins, G. T., Vos, M. A. J., & Pilot, A. (2011). *Leerlingpercepties van onderzoek & ontwerpen in het technasium.*
68. Bokhove, Chr. (2011). *Use of ICT for acquiring, practicing and assessing algebraic expertise.*

67. Boerwinkel, D. J. & Waarlo, A. J. (2011). *Genomics education for decision-making. Proceedings of the second invitational workshop on genomics education, 2-3 December 2010.*
66. Kolovou, A. (2011). *Mathematical problem solving in primary school.*
65. Meijer, M. R. (2011). *Macro-meso-micro thinking with structure-property relations for chemistry. An explorative design-based study*.
64. Kortland, J., & Klaassen, C. J. W. M. (2010). *Designing theory-based teaching-learning sequences for science. Proceedings of the symposium in honour of Piet Lijnse at the time of his retirement as professor of Physics Didactics at Utrecht University*.
63. Prins, G. T. (2010). *Teaching and learning of modelling in chemistry education. Authentic practices as contexts for learning*.
62. Boerwinkel, D. J., & Waarlo, A. J. (2010). *Rethinking science curricula in the genomics era. Proceedings of an invitational workshop*.
61. Ormel, B. J. B. (2010). *Het natuurwetenschappelijk modelleren van dynamische systemen. Naar een didactiek voor het voortgezet onderwijs*.
60. Hammann, M., Waarlo, A. J., & Boersma, K. Th. (Eds.) (2010). *The nature of research in biological education: Old and new perspectives on theoretical and methodological issues – A selection of papers presented at the VIIth Conference of European Researchers in Didactics of Biology*.
59. Van Nes, F. (2009). Young children's spatial structuring ability and emerging number sense.
58. Engelbarts, M. (2009). *Op weg naar een didactiek voor natuurkunde-experimenten op afstand. Ontwerp en evaluatie van een via internet uitvoerbaar experiment voor leerlingen uit het voortgezet onderwijs*.
57. Buijs, K. (2008). *Leren vermenigvuldigen met meercijferige getallen*.
56. Westra, R. H. V. (2008). *Learning and teaching ecosystem behaviour in secondary education: Systems thinking and modelling in authentic practices.*
55. Hovinga, D. (2007). *Ont-dekken en toe-dekken: Leren over de veelvormige relatie van mensen met natuur in NME-leertrajecten duurzame ontwikkeling*.
54. *Westra, A. S. (2006). A new approach to teaching and learning mechanics.*
53. Van Berkel, B. (2005). *The structure of school chemistry: A quest for conditions for escape.*
52. Westbroek, H. B. (2005). *Characteristics of meaningful chemistry education: The case of water quality*.
51. Doorman, L. M. (2005). *Modelling motion: from trace graphs to instantaneous change*.
50. Bakker, A. (2004). *Design research in statistics education: on symbolizing and computer tools*.

49. Verhoeff, R. P. (2003). *Towards systems thinking in cell biology education*.
48. Drijvers, P. (2003). *Learning algebra in a computer algebra environment. Design research on the understanding of the concept of parameter*.
47. Van den Boer, C. (2003). *Een zoektocht naar verklaringen voor achterblijvende prestaties van allochtone leerlingen in het wiskundeonderwijs.*
46. Boerwinkel, D. J. (2003). *Het vormfunctieperspectief als leerdoel van natuuronderwijs. Leren kijken door de ontwerpersbril*.
45. Keijzer, R. (2003). *Teaching formal mathematics in primary education. Fraction learning as mathematising process*.
44. Smits, Th. J. M. (2003). *Werken aan kwaliteitsverbetering van leerlingonderzoek: Een studie naar de ontwikkeling en het resultaat van een scholing voor docenten.*
43. Knippels, M. C. P. J. (2002). *Coping with the abstract and complex nature of genetics in biology education – The yo-yo learning and teaching strategy.*
42. Dressler, M. (2002). *Education in Israel on collaborative management of shared water resources.*
41. Van Amerom, B.A. (2002). *Reinvention of early algebra: Developmental research on the transition from arithmetic to algebra*.
40. Van Groenestijn, M. (2002). *A gateway to numeracy. A study of numeracy in adult basic education.*
39. Menne, J. J. M. (2001). *Met sprongen vooruit: een productief oefenprogramma voor zwakke rekenaars in het getallengebied tot 100 – een onderwijsexperiment.*
38. De Jong, O., Savelsbergh, E.R., & Alblas, A. (2001). *Teaching for scientific literacy: context, competency, and curriculum.*
37. Kortland, J. (2001). *A problem-posing approach to teaching decision making about the waste issue.*
36. Lijmbach, S., Broens, M., & Hovinga, D. (2000). *Duurzaamheid als leergebied; conceptuele analyse en educatieve uitwerking.*
35. Margadant-van Arcken, M., & Van den Berg, C. (2000). *Natuur in pluralistisch perspectief – Theoretisch kader en voorbeeldlesmateriaal voor het omgaan met een veelheid aan natuurbeelden.*
34. Janssen, F. J. J. M. (1999). *Ontwerpend leren in het biologieonderwijs. Uitgewerkt en beproefd voor immunologie in het voortgezet onderwijs*.
33. De Moor, E. W. A. (1999). *Van vormleer naar realistische meetkunde – Een historisch-didactisch onderzoek van het meetkundeonderwijs aan kinderen van vier tot veertien jaar in Nederland gedurende de negentiende en twintigste eeuw.*

32. Van den Heuvel-Panhuizen, M., & Vermeer, H. J. (1999). *Verschillen tussen meisjes en jongens bij het vak rekenen-wiskunde op de basisschool – Eindrapport MOOJ-onderzoek.*
31. Beeftink, C. (2000). *Met het oog op integratie – Een studie over integratie van leerstof uit de natuurwetenschappelijke vakken in de tweede fase van het voortgezet onderwijs.*
30. Vollebregt, M. J. (1998). *A problem posing approach to teaching an initial particle model.*
29. Klein, A. S. (1998). *Flexibilization of mental arithmeticsstrategies on a different knowledge base – The empty number line in a realistic versus gradual program design*.
28. Genseberger, R. (1997). *Interessegeoriënteerd natuur- en scheikundeonderwijs – Een studie naar onderwijsontwikkeling op de Open Schoolgemeenschap Bijlmer*.
27. Kaper, W. H. (1997). *Thermodynamica leren onderwijzen*.
26. Gravemeijer, K. (1997). *The role of context and models in the development of mathematical strategies and procedures*.
25. Acampo, J. J. C. (1997). *Teaching electrochemical cells – A study on teachers' conceptions and teaching problems in secondary education*.
24. Reygel, P. C. F. (1997). *Het thema 'reproductie' in het schoolvak biologie*.
23. Roebertsen, H. (1996). *Integratie en toepassing van biologische kennis – Ontwikkeling en onderzoek van een curriculum rond het thema 'Lichaamsprocessen en Vergift'*.
22. Lijnse, P. L., & Wubbels, T. (1996). *Over natuurkundedidactiek, curriculumontwikkeling en lerarenopleiding*.
21. Buddingh', J. (1997). *Regulatie en homeostase als onderwijsthema: een biologie-didactisch onderzoek*.
20. Van Hoeve-Brouwer G. M. (1996). *Teaching structures in chemistry – An educational structure for chemical bonding*.
19. Van den Heuvel-Panhuizen, M. (1996). *Assessment and realistic mathematics education.*
18. Klaassen, C. W. J. M. (1995). *A problem-posing approach to teaching the topic of radioactivity.*
17. De Jong, O., Van Roon, P. H., & De Vos, W. (1995). *Perspectives on research in chemical education.*
16. Van Keulen, H. (1995). *Making sense – Simulation-of-research in organic chemistry education.*
15. Doorman, L. M., Drijvers, P. & Kindt, M. (1994). *De grafische rekenmachine in het wiskundeonderwijs.*
14. Gravemeijer, K. (1994). *Realistic mathematics education*.
13. Lijnse, P. L. (Ed.) (1993). *European research in science education*.

12. Zuidema, J., & Van der Gaag, L. (1993). *De volgende opgave van de computer.*
11. Gravemeijer, K., Van den Heuvel-Panhuizen, M., Van Donselaar, G., Ruesink, N., Streefland, L., Vermeulen, W., Te Woerd, E., & Van der Ploeg, D. (1993). *Methoden in het reken-wiskundeonderwijs, een rijke context voor vergelijkend onderzoek*.
10. Van der Valk, A. E. (1992). *Ontwikkeling in Energieonderwijs*.
9. Streefland, L. (Ed.) (1991). *Realistic mathematics education in primary schools.*
8. Van Galen, F., Dolk, M., Feijs, E., & Jonker, V. (1991). *Interactieve video in de nascholing reken-wiskunde.*
7. Elzenga, H. E. (1991). *Kwaliteit van kwantiteit*.
6. Lijnse, P. L., Licht, P., De Vos, W., & Waarlo, A. J. (Eds.) (1990). *Relating macroscopic phenomena to microscopic particles: a central problem in secondary science education.*
5. Van Driel, J. H. (1990). *Betrokken bij evenwicht*.
4. Vogelezang, M. J. (1990). *Een onverdeelbare eenheid*.
3. Wierstra, R. F. A. (1990). *Natuurkunde-onderwijs tussen leefwereld en vakstructuur.*
2. Eijkelhof, H. M. C. (1990). *Radiation and risk in physics education*.
1. Lijnse, P. L., & De Vos, W. (Eds.) (1990). *Didactiek in perspectief*.

ICO Publication List

388. Day, I.N.Z. (28-06-2018), *Intermediate assessment in higher education* (Leiden: Leiden University)

389. Huisman, B.A. (12-09-2018) *Peer feedback on academic writing.* Leiden: Leiden University.

390. Van Berg, M. (17-09-2018) *Classroom Formative Assessment. A quest for a practice that enhances students' mathematics performance*. Groningen: University of Groningen.

391. Tran, T.T.Q. (19-09-2018) *Cultural differences in Vietnam : differences in work-related values between Western and Vietnamese culture and cultural awareness at higher education*. Leiden: Leiden University

392. Boelens, R. (27-09-2018) *Studying blended learning designs for hands-on adult learners*. Ghent: Ghent University.

393. Van Laer, S. (4-10-2018) *Supporting learners in control: investigating self-regulated learning in blended learning environments*. Leuven: KU Leuven.

394. Van der Wilt, F.M. (08-10-18) *Being rejected*. Amsterdam: Vrije Universiteit Amsterdam.

395. Van Riesen, S.A.N. (26-10-2018) *Inquiring the effect of the experiment design tool: whose boat does it float?* Enschede: University of Twente.

396. Walhout, J.H. (26-10-2018) *Learning to organize digital information* Heerlen: Open University of the Netherlands.

397. Gresnigt, R. (08-11-2018) *Integrated curricula: An approach to strengthen Science & Technology in primary education*. Eindhoven: Eindhoven University of Technology.

398. De Vetten, A.J. (21-11-2018) *From sample to population*. Amsterdam: Vrije Universiteit Amsterdam.

399. Nederhand M.L. (22-11-2018) *Improving Calibration Accuracy Through Performance Feedback*. Rotterdam: Erasmus University Rotterdam.

400. Kippers, W.B. (28-11-2018) *Formative data use in schools. Unraveling the process*. Enschede: University of Twente.

401. Fix, G.M. (20-12-2018) *The football stadium as classroom. Exploring a program for at-risk students in secondary vocational education*. Enschede: University of Twente.

402. Gast, I. (13-12-2018) *Team-Based Professional Development – Possibilities and challenges of collaborative curriculum design in higher education*. Enschede: University of Twente.

403. Wijnen, M. (01-02-2019) *Introduction of problem-based learning at the Erasmus School of Law: Influences on study processes and outcomes*. Rotterdam: Erasmus University Rotterdam

404. Dobbelaer, M.J. (22-02-2019) *The quality and qualities of classroom observation systems*. Enschede: University of Twente

405. Van der Meulen, A.N. (28-02-2019) *Social cognition of children and young adults in context*. Amsterdam: Vrije Universiteit Amsterdam

406. Schep, M. (06-03-2019) *Guidance for guiding. Professionalization of guides in museums of art and history*. Amsterdam: University of Amsterdam

407. Jonker, H.M. (09-04-2019) *Teachers' perceptions of the collaborative design and implementation of flexibility in a blended curriculum*. Amsterdam: University of Amsterdam

408. Wanders, F. H. K. (03-05-2019). *The contribution of schools to societal participation of young adults: The role of teachers, parents, and friends in stimulating societal interest and societal involvement during adolescence*. Amsterdam: University of Amsterdam

409. Schrijvers, M.S.T. (03-05-2019) *The story, the self, the other. Developing insight into human nature in the literature classroom*. Amsterdam: University of Amsterdam

410. Degrande, T. (08-05-2019) *To add or to multiply? An investigation of children's preference for additive or multiplicative relations*. Leuven: KU Leuven.

411. Filius, R.M. (23-05-2019) *Peer feedback to promote deep learning in online education. Unravelling the process*. Utrecht: Utrecht University

412. Woldman, N. (24-05-2019) *Competence development of temporary agency workers*. Wageningen: Wageningen University

413. Donszelman, S. (06-06-2019) *Doeltaal-leertaal didactiek, professionalisering en leereffecten. Amsterdam*: Vrije Universiteit Amsterdam

414. Van Oeveren, C.D.P. (12-06-2019) *ITHAKA gaf je de reis*. Amsterdam: Vrije Universiteit Amsterdam

415. Agricola, B.T. (21-06-2019) *Who's in control? Finding balance in student-teacher interactions*. Utrecht: Utrecht University

416. Cuyvers, K. (28-08-2019), *Unravelling medical specialists self -regulated learning in the clinical environment*. Antwerp: University of Antwerp

417. Vossen, T.E. (04-09-2019) *Research and design in STEM education*. Leiden: Leiden University

418. Van Kampen, E. (05-09-2019) *What's CLIL about bilingual education?* Leiden: Leiden University

419. Henderikx, M.A. (06-09-2019) *Mind the Gap: Unravelling learner success and behaviour in Massive Open Online Courses*. Heerlen: Open University of the Netherlands

420. Liu, M. (13-09-2019) *Exploring culture-related values in Chinese student teachers' professional self-understanding and teaching experiences*. Utrecht: Utrecht University

421. Sun, X. (13-09-2019) *Teacher-student interpersonal relationships in Chinese secondary education classrooms*. Utrecht: Utrecht University

422. Wu, Q. (02-10-2019) *Making Construct-Irrelevant Variance Relevant: Modelling item position effects and response behaviors on multiple-choice tests*. Leuven: KU Leuven

423. Jansen, R.S. (11-10-2019) *Dealing with autonomy: Self-regulated learning in open online education*. Utrecht: Utrecht University

424. Van Ginkel, S.O. (23-10-2019) *Fostering oral presentation competence in higher education*. Wageningen: Wageningen University

425. Van der Zanden, P. (05-11-2019) *First-year student success at university: Domains, predictors, and preperation in secondary education*. Nijmegen: Radboud University Nijmegen

426. De Bruijn, A.G.M. (14-11-2019) *The brain in motion: Effects of different types of physical activity on primary school children's academic achievement and brain activation*. Groningen: University of Groningen

427. Hopster-Den Otter, D. (28-11-2019) *Formative assessment design: A balancing act*. Enschede: University of Twente

428. Harmsen, R. (10-12-2019) *Let's talk about stress. Beginning secondary school teachers' stress in the context of induction programmes*. Groningen: University of Groningen

429. Post, T. (11-12-2019) *Fostering inquiry-based pedagogy in primary school: a longitudinal study into the effects of a two-year school improvement project*. Enschede: University of Twente

430. Ackermans, K. (20-12-2019) *Designing Video-Enhanced Rubrics to Master Complex Skills*. Heerlen: Open University of the Netherlands

Universiteit Utrecht

Introductory statistics courses are both essential and challenging for many university students. Students struggle to understand the abstract concepts involved, such as significance level and *p*-value, and the role of uncertainty in statistical procedures. Appropriate feedback could support students in gaining understanding, but is difficult to provide for teachers, since the number of students enrolled in such courses is often large. In this thesis, a solution is sought in automated feedback in an Intelligent Tutoring System, guided by the question: How can automated feedback support students in higher education in gaining understanding of statistics? In two first-year introductory statistics courses for social-sciences students, two feedback types were implemented: inner loop feedback on steps in hypothesis-testing tasks by a *domain reasoner* and outer loop feedback over series of tasks in the form of *inspectable student models*.

Separate studies focused on the design, implementation, and students' use of the two feedback types. Design was based on promising paradigms, such as model-tracing and constraint-based modeling for the domain reasoner. Students' use of the feedback was evaluated by investigating their feedback-seeking and decision-making behavior. Finally, the influence of both feedback types on students' course performance was assessed. Lower-achieving students were found to benefit from student models, and students who had had enough time to familiarize themselves with the feedback were found to benefit from the domain reasoner. Hence, the combination of feedback types has the potential to provide many students with useful guidance in the process of learning statistics.