**2018-2019 Annual Program Assessment Report Guide**

Please submit report to your department chair or program coordinator, the Associate Dean of your College, and to james.solomon@csun.edu, Director of the Office of Academic Assessment and Program Review, by **September 30, 2019**. You may, but are not required to, submit a separate report for each program, including graduate degree programs, which conducted assessment activities, or you may combine programs in a single report. **Please include this form with your report in the same file and identify your department/program in the file name.**

**College:** Science and Mathematics
**Department:** Mathematics
**Program:** B.A., B.S., and M.S.
**Assessment liaison:** Daniel Katz

1. **Please check off whichever is applicable:**

   A. ____**X**____ **Measured student work within program major/options.**

   B. ____**X**____ **Analyzed results of measurement within program major/options.**

   C. ____**X**____ **Applied results of analysis to program review/curriculum/review/revision major/options.**

   D. _____ **Focused exclusively on the direct assessment measurement of General Education Arts and Humanities student learning outcomes**

2. **Overview of Annual Assessment Project(s).** On a separate sheet, provide a brief overview of this year's assessment activities, including:

   - an explanation for why your department chose the assessment activities (measurement, analysis, application, or GE assessment) that it enacted
   - if your department implemented assessment **option A**, identify which program SLOs were assessed (please identify the SLOs in full), in which classes and/or contexts, what assessment instruments were used and the methodology employed, the resulting scores, and the relation between this year's measure of student work and that of past years: (include as an appendix any and all relevant materials that you wish to include)
   - if your department implemented assessment **option B**, identify what conclusions were drawn from the analysis of measured results, what changes to the program were planned in response, and the relation between this year's analyses and past and future assessment activities
   - if your department implemented **option C**, identify the program modifications that were adopted, and the relation between program modifications and past and future assessment activities
   - if your program implemented **option D**, exclusively or simultaneously with **options A, B, and/or C**, identify the basic skill(s) assessed and the precise learning outcomes assessed, the assessment instruments and methodology employed, and the resulting scores
   - in what way(s) your assessment activities may reflect the university's commitment to diversity in all its dimensions but especially with respect to underrepresented groups
   - any other assessment-related information you wish to include, including SLO revision (especially to ensure continuing alignment between program course offerings and both program and university student learning outcomes), and/or the creation and modification of new assessment instruments

3. **Preview of planned assessment activities for 2019-20.** Include a brief description as reflective of a continuous program of ongoing assessment.

# Overview of Assessment Projects
# 2018–2019 Academic Year
# Report to the Office of Academic Assessment

Prepared by the Assessment Committee
(Alberto Candel, Daniel Katz, and Jason Lo)
in Collaboration with the Graduate Committee
Department of Mathematics
California State University, Northridge

24 September 2019

# Contents

# Chapter 1

# Introduction

## 1.1 Synopsis of Activities Done for Academic Year 2018–2019

This report describes all the assessment activities performed by the Department of Mathematics at California State University, Northridge for the 2018–2019 academic year:

- We intervened in the Math 320 course (Foundations of Higher Mathematics), which is a required course for all bachelor's program options. We reminded the instructor of recommendations made the last time this course was assessed. When a measurement was made on the work of the students in this class (see the next point for this measurement), the issues that had been problematic in the previous measurement were barely evident in the student work, so this intervention can be judged a success.

- We assessed SLO #1 for the Mathematics Bachelor's Program (*Devise proofs of basic results concerning sets and number systems*) in the same Math 320 class. Our measurements indicate satisfactory achievement of the SLO, but with room for improvement in students' logical order and presentation.

- We then performed a second intervention in the same Math 320 course by relaying to the instructor our recommendations to emphasize logical order and presentation. When a measurement was made on the work of the students in this class (see the next point for this measurement), some improvement was evident.

- We then performed a second assessment of SLO #1 for the Mathematics Bachelor's Program in the same Math 320 course. Our measurements indicate satisfactory achievement of the SLO (at an even higher level than the previous measurement) and some improvement in logical order and presentation.

- We assessed SLO #5 for the Mathematics Bachelor's Program (*Write simple computer programs to perform computations arising in the mathematical sciences*) in Math 382/L (Introduction to Scientific Computing and Lab), which is a required course for all bachelor's program options. Our measurements indicate that satisfactory achievement of the SLO, but noted that, due to the nature of the interactive environment in which the students were programming, it is possible for students to get the right answer within a session but not save a program that reliably reproduces these results, and we suggest possible changes to the instruction or the curriculum, or even the SLO (since it does not address the issue of reliability).

- We assessed SLO #5 for the Master's Program Specialization I (Mathematics), which is at the same time SLO #5 for the Master's Program Specialization II (Applied Mathematics): *Solve mathematical problems at the graduate level.* We assessed this SLO by a measurement of the Math 540 class (Regression Analysis). Given the scores that the graduate students achieved in our measurement, it is our judgement that the graduate students are making satisfactory progress on this particular SLO.

## 1.2 Preview of Planned Assessment Activities for Academic Year 2019–2020

In the 2015–2016 academic year, the Department of Mathematics revised the undergraduate program SLOs and created graduate program SLOs. We have been assessing these new undergraduate and graduate SLOs to obtain a first measure of our programs so as to build a baseline for future assessment and a reference point to assist in curriculum development. So far we have assessed all five of the undergraduate program SLOs; SLOs #3, #4, and #5 for the M.S. Program Specialization I (Mathematics) out of five total; and SLOs #2, #4, and #5 for the M.S. Program Specialization II (Applied Mathematics) out of five total. For the undergraduate program, our focus will now be on applying our observations and recommendations and measuring the effects. For the graduate program, we plan to continue to assess

these new SLOs over the next few years until we have covered the full set of SLOs for each specialization.

In 2016–2017, we also assessed all four of the General Education Basic Skills SLOs in Mathematics, and we added an additional measurement to that assessment in 2017–2018. During our assessment of the General Education Basic Skills SLOs in 2016–2017, we mentioned to the Coordinator for Program Review and to the Director of the Office of Academic Assessment and Program Review that the General Education Basic Skills SLOs for Mathematics would benefit from revision to state clearer, more assessable goals. If the Office of Academic Assessment and Program Review is ready for us to begin this task, we may also be able to work on this revision in 2019–2020.

# Chapter 2

# Spring 2019: First Intervention in Math 320 Class

## 2.1 Relevant SLO and Background

The following SLO for the Mathematics Bachelor's Program was assessed in the Fall of 2015 in the Math 320 class:

**SLO #1: Devise proofs of basic results concerning sets and number systems.**

The signature assignment used in that assessment was a problem from the final exam in which students were asked to prove a statement using mathematical induction. Following that assessment measurement and its analysis, the following recommendations were made:

1. When teaching mathematical induction, emphasize the importance of establishing a base case.

2. Students try to use set-theoretic and logical notation to present their proofs, but often misuse it so that it does not improve the clarity of their presentation. Train the students in proper use of mathematical logic and set theory notation to improve clarity in communicating mathematical ideas.

## 2.2 Intervention

The above recommendations were relayed to the instructor of the Math 320 class in Spring 2019 before mathematical induction was taught. We then measured the

SLO using a signature assignment similar to the one used in Fall 2015 (although this one was a quiz during the term rather than a final exam). The details of that measurement and its analysis are given in full in Chapter 3, but in brief, the intervention appears to be a success because almost all students established the base case and the set-theoretic and logical notation used by the students, while not always perfect, did not often detract from the clarity of their presentation of the proof.

# Chapter 3

# Spring 2019: First Measurement and Analysis of Math 320 Class

## 3.1 SLO Assessed

The following SLO for the Mathematics Bachelor's Program was assessed:

**SLO #1: Devise proofs of basic results concerning sets and number systems.**

## 3.2 Role in the Program

Math 320 is a required course for all options of the bachelor's program in mathematics. Math 320 serves as a gateway into the upper-division math program at CSUN, and a large number of 300- and 400-level courses depend upon it as a prerequisite. Math 320 focuses on techniques of formal mathematical reasoning in context of number theory, combinatorics and analysis.

## 3.3 Section

One section of this course was offered in Spring 2019; it had 40 students. The textbook was *A Prelude to Advances Mathematics* by B. A. Sethuraman (a text by a CSUN faculty member).

## 3.4 Signature Assignment

The signature assignment was a quiz given during the term, which had a single problem which asked the students to prove an identity concerning natural numbers by means of induction.

## 3.5 Results, Analysis, and Recommendations

Of the 40 students enrolled in the class, 37 turned in the quiz. Two full time faculty of the Mathematics Department scored the quiz from 0 to 3 according to the rubric that is included in Appendix A.1. The mean score was 1.96 and the median score was 2. Figure 3.1 shows the distribution of scores.

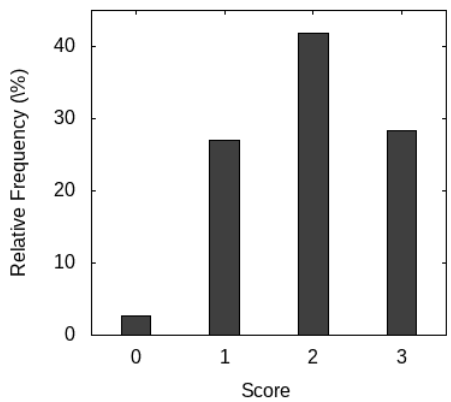


Figure 3.1: Score distribution for Math 320 quiz

The most critical obstacle to success in this problem was that some students did not make clear whether a statement was already known/proved, or whether it was a claim that they were about to prove. In some cases, students began by stating what they wanted to prove, then deduced from this a tautology, and then concluded from this that the desired statement must be true since it implies an obviously true statement. This is logically invalid, although sometimes a valid proof can be recovered from such work by proving that all the steps of the deduction are biconditional (that is, they remain valid if one reverses the direction of the implications). It is essential to make a sharp and clear distinction between what has been established from what one wishes to establish in a course emphasizing proofs.

The standard convention in mathematics is that statements that do not contain any specific description are things that are known or proved at the point when they are stated. (As such, they must be obvious statements, given assumptions, or things arrived at by valid deductions from previous statements on the page.) For motivation, it is sometimes useful to indicate what one wants to prove as a goal, but in this case it is necessary to make explicit that such a statement is what one wants to show, but has not yet shown. We therefore recommend to the current Math 320 instructor and to future instructors:

1. to emphasize the importance of clearly indicating when a statement is a desired goal rather than something already known or proved, and

2. to indicate that in proofs one must proceed from known and established statements to a desired conclusion, not begin with the conclusion and try to arrive at known facts.

Another problem which is not central to the SLO, but which nevertheless hindered many students, was that many students made errors in algebraic manipulation of polynomials. This difficulty was further compounded by the fact that many students did not use (or used incorrectly) factoring of polynomials, which could be used to lighten the work considerably.

Overall, we find the achievement of the SLO satisfactory. However, we feel that there is much room for improvement, especially with regards to the logical ordering and presentation of proofs.

# Chapter 4

# Spring 2019: Second Intervention in Math 320 Class

## 4.1 Relevant SLO and Background

The following SLO for the Mathematics Bachelor's Program was assessed earlier in the Spring of 2019 in the Math 320 class (see Chapter 3):

**SLO #1: Devise proofs of basic results concerning sets and number systems.** The signature assignment used in that assessment was a problem from a quiz in which students were asked to prove a statement using mathematical induction. Following that assessment measurement and its analysis, the following recommendations were made:

1. to emphasize the importance of clearly indicating when a statement is a desired goal rather than something already known or proved, and

2. to indicate that in proofs one must proceed from known and established statements to a desired conclusion, not begin with the conclusion and try to arrive at known facts.

## 4.2 Intervention

The above recommendations were relayed to the instructor of the Math 320 class in Spring 2019 after the quiz (first measurement: see Chapter 3) was scored for assessment and the results were analyzed. We then measured the SLO using a signature

assignment similar to the one used in the quiz, with the new signature assignment embedded in the final exam of the same class. This new signature assignment was scored according to the same rubric as was used for the quiz. The details of that measurement and its analysis are given in full in Chapter 5, but in brief, the intervention appears to have had some improvement in the logical ordering and presentation of proofs, and there is an increase in the mean score for the second measurement.

# Chapter 5

# Spring 2019: Second Measurement and Analysis of Math 320 Class

## 5.1 SLO Assessed

The following SLO for the Mathematics Bachelor's Program was assessed:

**SLO #1: Devise proofs of basic results concerning sets and number systems.**

## 5.2 Role in the Program

Same as in Section 3.2.

## 5.3 Section

Same as in Section 3.3.

## 5.4 Signature Assignment

The signature assignment was a portion of problem on the final exam, which asked the students to prove an identity concerning natural numbers by means of induction.

## 5.5 Results, Analysis, and Recommendations

Of the 40 students enrolled in the class, 39 turned in the final exam. Two full time faculty of the Mathematics Department scored the signature assignment from 0 to 3 according to the same rubric that was used for the first measurement in Chapter 3: see Appendix A.1 for text of the rubric. The mean score was 2.62 and the median score was 3. Figure 5.1 shows the distribution of scores; there are two scores per student, one for each scorer. All 37 students who took the quiz which was measured

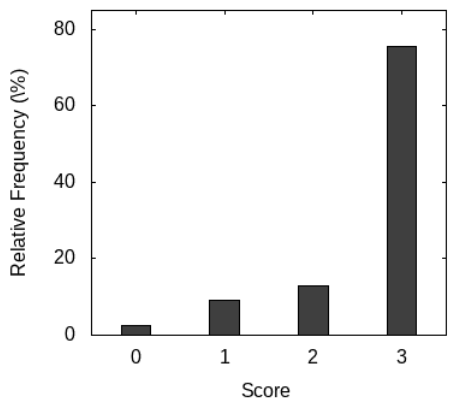


Figure 5.1: Score distribution for Math 320 final exam

earlier in the term (see Chapter 3) also took the final exam, and there were two students who took the final but not the quiz. For the 37 students who did both signature assignments, we look at the change in score (final exam score minus quiz score): the average was an improvement by 0.70 points, with a median improvement of 1 point. Figure 5.2 shows the distribution of changes (final exam score minus quiz score).

If each student is given a single score for the quiz by averaging the two scores of the two scorers, and similarly is given a single averaged score for the final exam, then the Pearson correlation between the students' quiz score and final exam score is 0.49, which indicates a low to moderate correlation.

As mentioned in the analysis of the previous measurement (see Chapter 3), the most critical obstacle to success in this problem was that some students did not make clear whether a statement was already known/proved, or whether it was a claim that they were about to prove. For example, students often begin by stating what they want to prove without indicating it out as such, and thus violating the

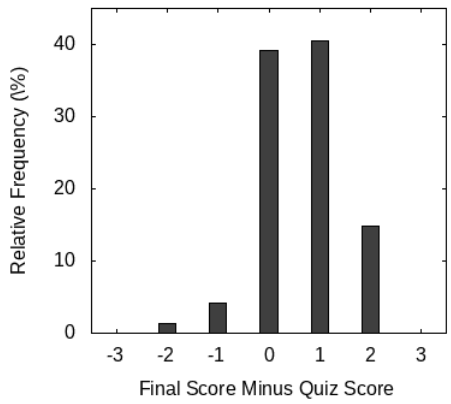


Figure 5.2: Distribution of Final Exam Score Minus Quiz Score for Math 320

convention that any unqualified statement is an assertion rather than an aspiration. Then students embark on the logically invalid method of trying to deduce from their goal a tautology so as to conclude from this that the desired statement must be true since it implies an obviously true statement. Almost half of the students had some problem of this sort on the quiz. After the intervention, about a third of the students still had problems of this sort on the final exam, although most of them were relatively mild. For example, it was common for students to use this backwards logic on the base case of the induction only, where the result is in any case so immediate that the backwards ordering of the logic is easily mentally corrected. Or students would write their desired statement only once without marking it as a goal, but then continue on correctly to not use this goal as a premise of what follows. Therefore, we can say that the intervention appears to have had some positive effect in diminishing logical errors of this nature. We should remark that this bad habit is likely the result of how mathematics is taught in secondary schools, where the emphasis is on the mechanics of getting from one expression to another via a series of manipulations rather than on the logical entailments of those manipulations. The consequences of this legacy of neglect of underlying logic are so widespread, that we recommend that the Mathematics Department discuss strategies for addressing it not only in Math 320, but throughout the undergraduate mathematics program. Since our intervention appears to have had some success, and since we hope for even more in this direction, we recommend that Math 320 instructors continue to work on the same recommendations that we made in the previous measurement (see Section 3.5):

1. to emphasize the importance of clearly indicating when a statement is a desired goal rather than something already known or proved, and

2. to indicate that in proofs one must proceed from known and established statements to a desired conclusion, not begin with the conclusion and try to arrive at known facts.

Overall we conclude that the achievement of SLO #1, already satisfactory in our previous measurement (see Chapter 3), has improved in this measurement, with some progress being made on the recommendations made in the intervention.

# Chapter 6

# Fall 2018: Measurement of Math 382/L Class

## 6.1 SLO Assessed

The following SLO for the Mathematics Bachelor's Program was assessed:

**SLO #5: Write simple computer programs to perform computations arising in the mathematical sciences.**

## 6.2 Role in the Program

Math 382/L is a required course for all options of the bachelor's program in mathematics. This course introduces students to computer programming aimed specifically at mathematical calculations. (Students fulfill another programming requirement by taking **COMP 110/L Introduction to Algorithms and Programming and Lab (3/1)**, which provides a more general introduction to computer programming, or else **COMP 106/L Computing in Engineering and Science and Lab (2/1)**, which no longer has an entry in the course catalog.)

## 6.3 Section

One section of this course was offered in Fall 2018; it had 40 students. The textbook was *Deep Learning* by Ian Goodfellow, Yoshua Bengio, and Aaron Courville (2016).

## 6.4 Signature Assignment

Portions of a three-part programming laboratory assignment were used as the signature assignment. They involve fitting a model to data using numerical linear algebra and displaying the results graphically.

## 6.5 Results, Analysis, and Recommendations

The scoring for formal assessment, based on the scoring rubric in Appendix A.2, was performed by two full-time faculty of the Department of Mathematics on the 27 assignments that were turned in (out of 30 enrolled students). The assignment was divided into three Sub-Problems (A, B, and C), each of which was scored from 0 to 2, thus giving the assignment a total score of 0 to 6. The two scorings were combined to give a list of 54 scores (two per exam). The additional appraisal of coding consistency was scored from 0 to 2 by one full-time faculty member of the Department of Mathematics on the same set of assignments using a rubric also included in Appendix A.2.

The mean and median scores for each of the three sub-problems and for the combined total are given in Table 6.1, and the distribution of scores is presented in Figures 6.1–6.4. Recall that each sub-problem is out of 2, and the total is out of 6 points possible.

Table 6.1: Mean and Median Scores for Math 382

| Problem | Mean Score | Median Score |
|---|---|---|
| Sub-Problem A | 1.46 | 2 |
| Sub-Problem B | 1.28 | 2 |
| Sub-Problem C | 1.37 | 2 |
| Total | 4.11 | 5 |

For the additional appraisal of coding consistency, the mean score was 0.93 and the median score was 1 (out of 2 points possible). The distribution of scores is presented in Figure 6.5.

The coding assignment was performed in an interactive environment (Python code in a Jupyter Notebook) in which students could attempt to run a program, possibly make errors, and then attempt to revise their code and try again. This is in many ways analogous to the use of hand calculator (but with more flexibility

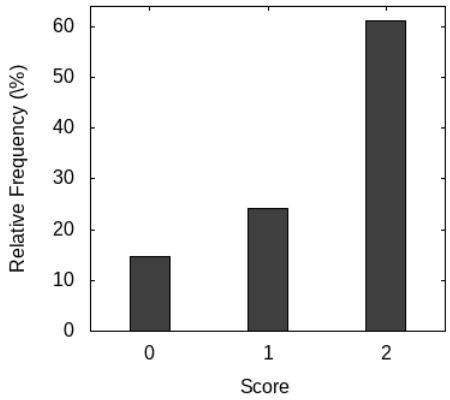


Figure 6.1: Score distribution for Math 382 Sub-Problem A

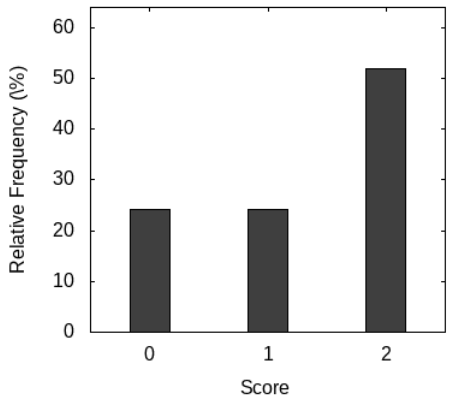


Figure 6.2: Score distribution for Math 382 Sub-Problem B

as to what can be programmed) to obtain a result. As intermediate values in the calculation are computed and assigned to named variables, such assignments of values to named variable will persist in the memory of the session even if the code that originally made the assignment is removed or modified (perhaps to assign a value to a different name) in a later revision. Therefore, although the students are asked to turn in their code and the output it produces, it is possible that what code remains after revision does not contain a full account of all the steps that were used to obtain the results. As such, when the revised code is reloaded into a fresh session, it may not produce the desired output: such was the case for many of the student

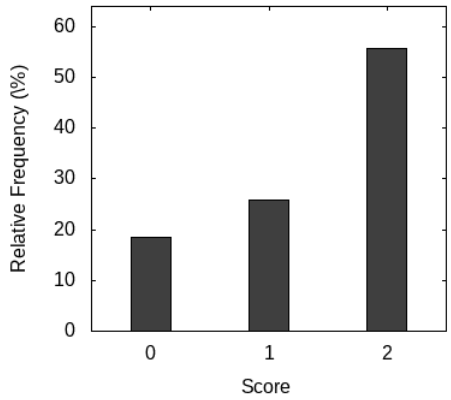


Figure 6.3: Score distribution for Math 382 Sub-Problem C

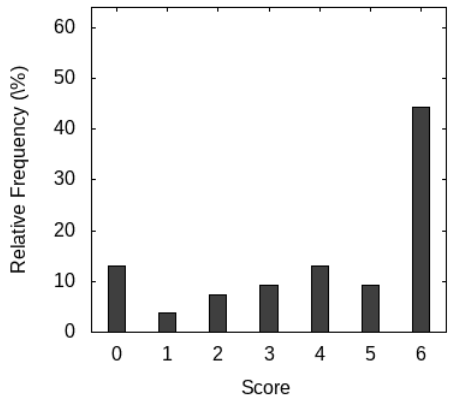


Figure 6.4: Score distribution for Math 382 total score

submissions. In fact, out of 2 points possible in the appraisal of consistency, the mean score was less than 1. This is highly problematic from the point of view of scoring, and we felt that it called for the additional appraisal of code consistency. We did not make this appraisal part of the assessment of the SLO because, as written, the SLO does not indicate whether the student programming should take place in a calculator-like interactive environment or whether the student should produce a code that can be run successfully without any human intervention during runtime. Although mathematics bachelor's students must also take a general programming class (COMP 110/L) that should make them aware of such issues, COMP 110/L is

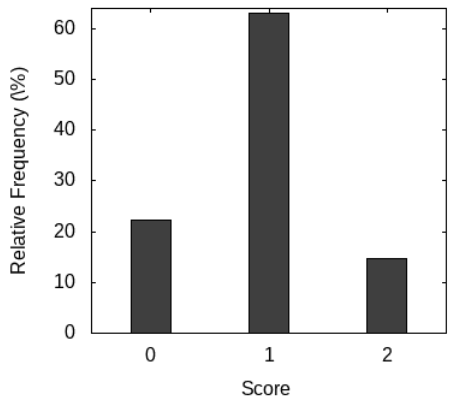


Figure 6.5: Score distribution for Math 382 coding consistency check

not listed as a prerequisite for Math 382/L. With this in mind, if we do not emphasize coding consistency issues, and focus solely on what is explicitly stated in the SLO (**Write simple computer programs to perform computations arising in the mathematical sciences**), then we see from the mean and median scores of 4.11 and 5, respectively, out of 6, that the students has satisfactory success at performing mathematical calculations using a programming language (Python) and so this is a satisfactory achievement of the SLO.

However, moving beyond the SLO, we feel that the lack of coding consistency is a real problem. Although at times one simply needs a quick answer to a question and will not want to re-use the code that was employed to obtain it, coding consistency is important in many academic and industrial jobs, for the following reasons:

- In mathematics, as in other branches of science or engineering, the value of a result depends largely on whether it can be checked or reproduced by other professionals. This is a hallmark of science and mathematics.
- Software is sold as commercial product, and as such, the reproducibility of codes is essential to the quality of software as a product.

This issue can be addressed at various levels:

1. Students should be warned of the non-reproducibility of code that arises from an interactive process, and they should be taught to test their codes by saving, terminating the current session, and then running the code in a fresh session. If they are working with Python in Jupyter Notebooks, they can achieve the

same effect by prefacing their code with the single command

```
%reset -f
```

which will clear all assigned variables each time the code is run afresh.

2. The Mathematics Department might consider making COMP 110/L a prerequisite for MATH 382/L. Experience in programming in compiled computer language (where the code itself is not modified by the programmer during runtime) teaches coding discipline that would serve students well in MATH 382/L.

3. The Mathematics Department might consider revising the SLO that we have measured here to indicate explicitly that the students are to produce reliable programs that perform computations arising in the mathematical sciences. The emphasis here is that the **programs**, once written properly, should perform the computations and obtain the results when run without any intelligent supervision. (As opposed to results obtained from an interactive programming activity that may not actually leave a workable program as its residue.)

# Chapter 7

# Spring 2019: Measurement of Math 540 Class and Analysis

## 7.1 SLO Assessed

We assessed SLO #5 for the Master's Program Specialization I (Mathematics), which is at the same time SLO #5 for the Master's Program Specialization II (Applied Mathematics):

**Solve mathematical problems at the graduate level.**

## 7.2 Role in the Program

For the Master's Program Specialization II (Applied Mathematics), Math 540 is on a short list of classes from which students must select two to fulfil 500-level requirements. It may also be taken as a 500-level elective by students from any specialization of the Master's Program. This course is one of the most popular statistics offerings, and is essential for any student who wants a complete background in statistics for work in industry or further academic studies.

## 7.3 Section

One section of this course was offered in Spring 2019; it had 7 students. The textbook was *Introduction to Linear Regression Analysis, 5th Edition* (2012) by Douglas C. Montgomery, Elizabeth A. Peck, G. Geoffrey Vining.

Table 7.1: Mean and Median Scores for Math 540 Assessment

| | Entire Class | Graduate Students Only |
|---|---|---|
| Problem 1, Mean | 0.83 | 1.33 |
| Problem 1, Median | 1 | 1.5 |
| Problem 2, Mean | 2.00 | 2.67 |
| Problem 2, Median | 2 | 3 |

## 7.4 Signature Assignment

The signature assignment consisted of two problems embedded in the final exam, which, for the purposes of this report, are labeled Problem 1 and Problem 2. Problem 1 is on distributional results about estimators in a simple linear regression model. Problem 2 is on the expected value of the mean square error of the estimator vector in a multiple regression model.

## 7.5 Results, Analysis, and Recommendations

Of the seven students enrolled in the class, six turned in the final exam. Two full time faculty of the Mathematics Department scored each of the assessment problems from 0 to 3 according to the rubric that is included in Appendix A.3. The mean and median scores for Problem 1 and 2 are given on Table 7.1. It should be noted that half the students who turned in the final exam were actually undergraduates, and their performance was considerably lower than that of the graduate students. To focus on the graduate program, we present statistics for the graduate students only in addition to statistics for the entire class. The distributions for the scores are given in Figures 7.1–7.4. Student performance on Problem #1 was generally rather poor. We conjecture that this may be due in large part to the fact that this material was covered near the beginning of the course. There are three parts to complete the answer to this problem. No student completed all the three parts. Some students did not attempt to answer one part of the question even though they had correct work on other parts. We are unable to tell whether they forgot those parts or whether they could not answer them. Some students solved problems that were not what was asked for. It seems the students were lost on what the problem is asking for, or they did not know how to answer it.

Student performance on Problem #2 was fairly good, much better than on the

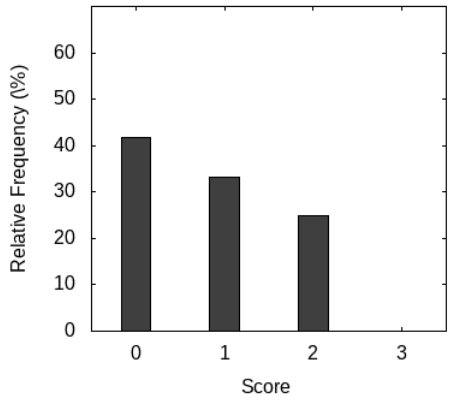


Figure 7.1: Score distribution for Math 540 Problem 1 for the entire class

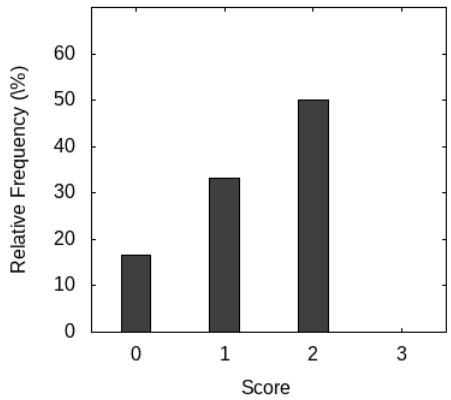


Figure 7.2: Score distribution for Math 540 Problem 1 for graduate students only

previous problem. Most points that students missed concern how to interpret the equation to obtain the requested results. The work of the graduate students in particular was perfect except that one part of the question was not answered by one student. The problem itself contains knowledge from linear algebra. Students showed a solid foundation on it.

Overall, graduate students did much better than undergraduates, as would be expected. Students demonstrated good capability with regard to the technical manipulations involved in problems at this level of mathematical statistics. Given that nine out of twelve of the aggregate ratings for the graduate students on the two prob-

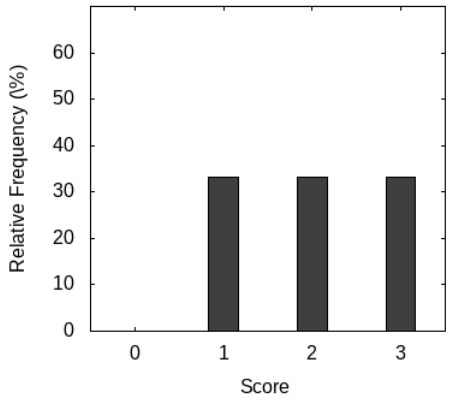


Figure 7.3: Score distribution for Math 540 Problem 2 for the entire class

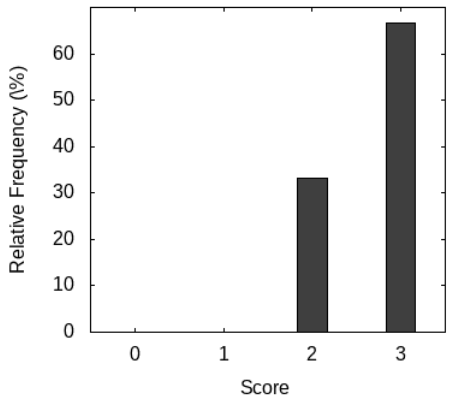


Figure 7.4: Score distribution for Math 540 Problem 2 for graduate students only

lems were either 2 (substantial correct ideas) or 3 (fully correct), it is our judgement that the graduate students are making satisfactory progress on this particular SLO. We recommend that additional emphasis be placed on getting students to focus on precisely what a question is asking (and what can be assumed) and in choosing the most effective method of solution. It is the nature of probability and statistics that some approaches to a problem may be legitimate but too inefficient to be feasible, where a simpler solution may instead be available. Students should be frequently reminded of this fact and see demonstrations through examples.

# Appendix A

# Rubrics

## A.1 Rubric for Math 320 (Spring 2019)

Students shall verify that an identity holds for all natural numbers using induction.

Students shall be given a score from 0 to 3 based on the following rubric:

| Score | Description |
|---|---|
| 0 | Student expresses no substantial idea about how to present the proof. |
| 1 | Student demonstrates a substantial idea, but falls far short of what is needed to present the complete argument. |
| 2 | Student makes substantial progress towards a proof, but with nontrivial incompleteness and/or flaws in logic and presentation.[1] |
| 3 | Student presents a valid, clear proof with no flaws or only trivial flaws |

[1]For example, the order of statements in the proof is such that the flow of the argument is partially obscured.

## A.2 Rubrics for Math 382/L (Fall 2018)

Students shall be scored based on three sub-problems, each of which is scored 0, 1, or 2. The total score is the sum of the scores on the three sub-problems, so the total score is from 0 to 6. The three sub-problems are:

- Sub-Problem A: Student shall use the Moore-Penrose pseudoinverse to calculate the slope and intercept of the line that is the least squares linear approximation of a given data set.
- Sub-Problem B: Student shall use an iterative gradient descent algorithm to approximate the slope and intercept of the line that is the least squares linear approximation of a given data set.
- Sub-Problem C: Student shall plot the data along with the least squares linear approximation.

The 0-1-2 scoring rubric for all of these parts is:

| Score | Description |
|---|---|
| 0 | Student's code shows no substantial idea or progress on the sub-problem |
| 1 | Student produced code that has at least one useful element relevant to the sub-problem, but did not obtain the correct solution (or only obtained the correct solution using a method other than that prescribed for the sub-problem) |
| 2 | Student was able to obtain the correct solution with a code that implements the correct mathematical ideas with no flaws or minor flaws |

Notes on implementation of this rubric:

1. Since the students did this assignment in a notebook-style interactive environment (Jupyter Notebook), it is possible for them to have done the correct mathematical programming and obtained the correct result during their session (and thus preserved the correct outputs), while at the same time not preserving a code that, if run from scratch in a fresh session, produces the same results. For the purposes of this assessment, we assume that the outputs displayed were validly obtained during the programming session. Therefore, we determine whether the student obtained the correct result from their own displayed outputs, and we determine whether their code implements the correct mathematical ideas by reading the code submitted. This is not an ideal situation, and we address the question of the reproducibility of coding outcomes in a separate appraisal (see below).

2. For Sub-Problems A and B, as the instructor specifically asks the students to print the slope and the intercept, students who compute these quantities within their programs but do not report them (by printing them so that the user can read them) cannot get full credit.

3. For Sub-Problem C, the student is asked to plot the data and the linear approximation in both Problems 3 and 4 of the assignment. For the purposes of this rubric, we report a score for whichever attempt was most successful. (For example, if a student does the plot correctly in Problem 3, but does not do so in Problem 4—perhaps because he or she could not make the gradient-descent algorithm work correctly and gave up—then that student would receive a score of 2 for Sub-Problem C.) If the student can correctly plot the data without the fitted line in Problem 2, then he or she can receive one of the two points for this Sub-Problem.

---

Rubric for Appraisal of Coding Practice: although this is not officially part of the SLO assessment, it is still important for students to produce solid code that gives reproducible outputs.

Using a 0-1-2 scoring rubric, we score the stability and usability of the students' code:

| Score | Description |
|---|---|
| 0 | When student's code is re-run, it does not produce the same output as shown in the student's submitted notebook, and minor modifications will not alter this situation |
| 1 | When student's code is re-run, it does not produce the same output as shown in the student's submitted notebook, but can be made to do so with minor modifications |
| 2 | Student's code produces the same output (allowing for usual platform-dependent implementation details) as shown in the student's submitted notebook |

Examples of what counts as a "minor modification":

1. Changing the path and name of the data file to point to where the instructor expected it to be.

2. Commenting out a small amount of code that is unnecessary for solution of the sub-problem but impedes the correct functioning of the code. For example, one

could comment out use a function from a library that is not needed and either has not been imported or cannot be relied upon to be present in all Python distributions.

3. Introducing a small amount of code to display things that have already been calculated but are not displayed when the code is run as written. For example, if the student has Python make a plot, but does not show it, one could invoke the "show()" command to make the code work.

4. One can obtain the correct behavior by running things a few times, so that dependencies of variables can be resolved by multiple passes of computation.

## A.3 Rubric for Math 540 (Spring 2019)

Problem #1: Students shall prove distributional results about estimators in a simple linear regression model.

Problem #2: Students shall determine the expected value of the mean square error of the estimator vector in a multiple regression model, and use the result to show what happens to it when predictor variables are multicollinear.

For each problem, students shall be given a score from 0 to 3 based on the following rubric:

| Score | Description |
|---|---|
| 0 | Demonstrates no relevant idea about how to solve any part of the problem. |
| 1 | Presents at least one relevant idea, but falls far short short of what is needed to solve all parts of the problem. |
| 2 | Presents a solution that contains substantial correct ideas, but not a fully correct solution. |
| 3 | Presents valid and clear solutions to all parts, with no flaws or very minor flaws. |