Does Starting Deep Learning Homework Earlier Improve Grades?

Edward Raff[a,b,c] **and Cynthia Matuszek**[b]

[a]Booz Allen Hamilton
[b]University of Maryland, Baltimore County
[c]Syracuse University

Abstract. Intuitively, students who start a homework assignment earlier and spend more time on it should receive better grades on the assignment. However, existing literature on the impact of time spent on homework is not clear-cut and comes mostly from K-12 education. It is not clear that these prior studies can inform coursework in deep learning due to differences in demographics, as well as the computational time needed for assignments to be completed. We study this problem in a post-hoc study of three semesters of a deep learning course at the University of Maryland, Baltimore County (UMBC), and develop a hierarchical Bayesian model to help make principled conclusions about the impact on student success given an approximate measure of the total time spent on the homework, and how early they submitted the assignment. Our results show that both submitting early and spending more time positively relate with final grade. Surprisingly, the value of an additional day of work is apparently equal across students, even when some require less total time to complete an assignment.

1 Introduction

In developing a course on deep learning for the University of Maryland, Baltimore County (UMBC), we focused on practical coding experience and implementation of deep learning methods for the course content and evaluation. Compared to assignments in some machine learning classes, the course requirement to use a Graphics Processing Unit (GPU) led us to strongly emphasize throughout the semester that students should start their homework early, as they need to have sufficient time to run their code, iterate and try to fix bugs if errors occurred, and ask the instructor for assistance. As the course progressed and assignments were due, students would sometimes ask how early should they start an assignment, and we had no quantifiable justification for our answers. This paper remedies this issue, and studies the overall question: does starting and/or submitting an assignment earlier improve student's grades on that homework?

The literature studying the impact of time spent on homework, at large, is sparse. One set of work studies the impact of "procrastination," measured by comparing the time an assignment is due and the time the assignment was submitted [12, 11, 3]. This is the easiest form of data to study as it is readily available in modern electronic submission systems. The current studies regularly conclude that those who submit earlier obtain better grades. While we collect the same data, we do not study it directly as it is a proxy for time spent. That is to say, a student who does the assignment the day before and submits the day before has procrastinated, but would not show up in the data as a procrastinator. Similarly a student who starts weeks in advance, and submits at the literal "11'th hour" did not procrastinate, but would be marked a procrastinator when using only submission time. In our study we have the start time of an assignment, and we use that with the time submitted to compute a "total time" spent on the assignment. This is not a contiguous measure of time or effort, but we argue a likely better measure of the quantity we care about: total effort spent on an assignment.

Others have also attempted to look at the total time spent on homework and its relation to performance, and have regularly concluded that too much time spent on homework can result in *reduced* scholastic performance [7, 14, 5]. All of these works focus on students in high school or earlier, and are focused on overall scholastic outcomes rather than per-assignment results. Similarly, the data is the result of survey information, where our total time is determined via the edit history of the assignment. For this reason we believe our total time measure to be a more reliable, though still not perfect, measure of the goal. In a larger sense, there are numerous differences between our population and those studied before (we study graduate students vs. K-12, homework grade vs. overall performance, and coding and deep learning vs. general Science Technology Engineering and Math subjects). The consistency of the prior studies results' about negative returns for "over-studying" necessitate exploration of the question.

The rest of our paper is organized as follows. First we will give extensive background on our data, the course, and necessary background to interpret and understand the results in section 2. We have $N = 68$ total subjects over three semesters of the course. Next we detail the model we use for understanding the data in section 3, which uses a hierarchical Bayesian approach, as is generally encouraged in studies of this nature [6]. Our results will be presented in section 4, where we conclude that more time spent is better than less, and submitting earlier and spending more time have a statistically significant positive impact on a student's grades. Finally we will review other related works in section 5 and then conclude in section 6.

2 Data Collection & Background

To study the question, our data is collected from a course taught by the author(s) in the Spring 2020, Fall 2020, and Spring 2021 semesters at UMBC. The course content and questions were developed into a book Inside Deep Learning[1] [19]. We note that this im-

[1] Available at https://www.manning.com/books/inside-deep-learning

mediately introduces a set of biases into our results. Most notably, the semesters involved have occurred at the onset of and through the COVID-19 pandemic. This has introduced stresses on students and faculty that are beyond the scope of this study. In addition, one set of instructor(s) are involved, and so any instructor modulated response will not be observable.

As part of the course design, students were instructed to write, test, and submit all of their homework within a Google Colaboratory environment. This choice was originally made as a mechanism to satisfy the desiderata:

1. Free or cheap GPU availability to students
2. Avoiding versioning conflicts and software installation issues
3. Having a simple means of running student assignments (also avoiding student vs. teacher package mismatches)
4. Provides an easy way for students to get help / feedback on assignments

An unintended benefit of this course design choice was that Colab kept a *sparse* edit history of the assignments. Different from a normal version control system, Colab will take snapshots that can be differentiated against each other (or the current version) of a document at regular intervals, or as an explicit save request is called. The exact mechanics of this process are not documented, but there does appear to be an age-off process where some subset of snapshots are removed over time, eventually resulting in no edit history.

This edit history, collected close in time to course completion, was initially used as a means of assisting and helping students with feedback on how to perform better in the course. For example, we could see when a student started the homework assignment 40 minutes before the assignment was due, and thus, was unable to complete the assignment. In such cases the student was coached and advised on time management and the need to reach out to instructors early if something may prevent them from timely completion. Retroactively, this also became the driving force for this study: is there a significant difference in student's grades based on when they begin their assignment?

To answer the question, we went through every student's homework assignments version history, which is available when students submit an editable link to their Colab files (as required by the course submission). It was quickly determined that the first edit in the history was not a reliable method of determining the student's start date, as many students would create the homework file days or weeks before starting the assignment. Small edits to copy questions would also occur. Other faux starts included copying code from the course book that would be used by the homework solution (e.g., assignment says to modify the code), but had not yet started actual modification. For this reason, we subjectively reviewed all edit histories until the first edit that appears to show the student trying to make progress on the assignment, and recorded that as the start date of the assignment. The submission time on the assignment was obtained via email or Blackboard, depending on which method was used to receive homeworks during the given semester.

Combined, this gives us the time of submission and the time between start and submission. A limitation we make explicit is that we cannot reliably measure or quantify the true time spent working on the assignment, because the Colab history is coalesced and undocumented in its triggering frequency/characteristics. There may be instances where a student in our data "starts early," but does not work on the assignment for an extended period of time. Since we have no way to detect this, we leave the issue to future work.

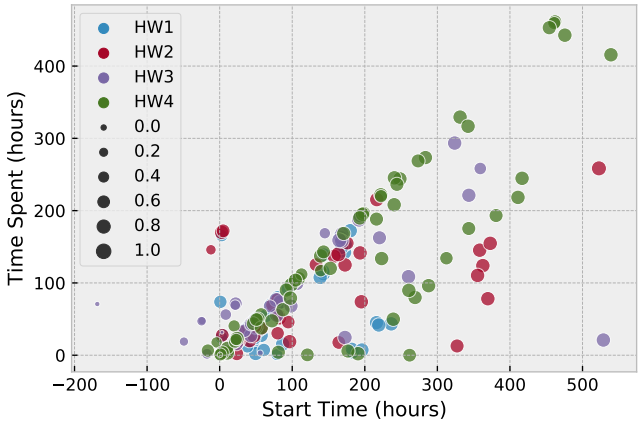

Figure 1. Visualization of the raw data used in this study. The number of hours in advance of the due date that a student started is on the x-axis, and the number of hours the student appears to have worked on the assignment is the y-axis. Color denotes which homework assignment, and size denotes the grade received on that assignment (1.0 = 100%, 0.0 = 0%). Note most students did well, so many large circles are present. Negative x-axis values occur when students start the assignment after it was due.

A visualization of the resulting data is given in fig. 1. Note that negative values on the x-axis indicate use of the late submission policy. Most students submit near the deadline, resulting in a strong linear trend. Some students realize they have completed the homework early, and fall to the bottom-right quadrant of the plot.

2.1 Homework Details

Each semester four homeworks were assigned, and their grades were the basis of this study. The design choices of these homeworks impact our modeling of the problem, and the reader's ability to subjectively interpret the applicability of the results to their own curricula and assignments. All four assignments were designed such that someone who knew how to perform all tasks could complete them within 40 minutes. This reflects the time of an expert practitioner/researcher who has previous done each assignment in the context of a job or research goal, and has over a decade of experience writing code and in machine learning. As such the 40 minutes is usually not reflective of how long a student will take to complete the assignment, but is done to serve as an upper-bound on the complexity of what is being asked of students to complete. This is not a theory oriented course, and is aimed at students looking to obtain practical knowledge and ultimately write code themselves in the future. The assignments assume that students do not have access to a GPU for days at a time, and so are designed that they can be completed within a day when done correctly. This constraint is born of insufficient funds to purchase GPUs for every student, while also not wanting to burden students with a large capital cost of a GPU when they do not yet know if they will enjoy deep learning.

Each homework assignment consisted of 4 or 5 coding questions with concise summaries in Table 1. Tasks included implementing feature processing, specific neural network architectures, and making specific modifications to an architecture in the book assigned, and comparing the impact of hyper-parameters on total run-time or accuracy of a model.

Table 1. Concise descriptions of the kinds of tasks each homework question(s) required of the students. Each is built from one or more problems from the book written for the course used in this study, Inside Deep Learning. The chapter and question of the full content under the problems column.

HW	Problems	Task
1	C2, Q2	Evaluate a model via AUC.
1	C2, Q3-4	Implement checkpointed training.
1	C2, Q5	Add more layers to a model
2	C3, Q2	Train a CNN on CIFAR10
2	C4, Q1-3	Train an RNN over text with a custom vocabulary.
3	C5, Q6	Perform hyperparameter tuning of a CNN.
3	C6, Q2	Train a deeper CNN with BatchNorm.
3	C6, Q6	Train an LSTM and compare to an RNN.
3	C7, Q1	Train and auto-encoder on MNIST without classes 5 & 9, then evaluate on the missing and included classes.
3	C7, Q8	Create an autoregressive loader aware of sentence boundaries.
4	C8, Q4	Replace pooling with strided convolutions in a U-Net
4	C9, Q1-2	Implement a convolutional GAN.
4	C10, Q1	Combine convolutions and attention for sequential image prediction.

The first and fourth (last) homework were designed to be easier to complete. The first to avoid overwhelming students at the onset, and the second to allow students more time to work on a semester-long final project. There were generally two weeks between each homework assignment and due date, with the next homework being assigned the day the previous was due. The third homework was intentionally designed to be harder, and the instructors suspected students would not give themselves sufficient effort to complete the assignment. For this reason, a 1-2 week extension was baked into the curricula and used every semester. For this reason, student start times can be significantly larger for the third homework.

The course policy included a "no questions asked" late grading policy, that allowed students to submit an assignment up to 72 hours late, for -10 points for each day the assignment was late. This meant a total late penalty of -30 hours was possible. This penalty was excluded from the data and calculations, as our goal is to make inferences about the value of additional time spent.

Because all courses are different, our results can not be used to infer that every deep learning assignment, class, or set of exercises will follow the results of this paper. Indeed, this will always be the case for any course taught, and no singular study can infer a recommendation appropriate to all universities and classes. Our hope is this study will be informative and help encourage others to study this aspect of education and determine if the results may be applicable and informative to their own instruction.

2.2 Removed Records

Not all student records were kept/used for this study. In total 7 student records (leaving 69 remaining) are excluded from our analysis due to the following:

- The student did not follow requirements on making the submission editable or starting the homework in Colab, meaning we did not have access to the needed information.
- The student cheated on the homework assignments, making them non-reflective of start time on student grades[2].
- Catastrophic life event such as death of an immediate family member or significant change in medical status.

The final project of the course is excluded from this study. Our experience was that students used multiple Colab instances in clever ways to make further progress on their final projects. This includes running multiple Colabs simultaneously to perform more experiments or using one to run experiments and a second to develop new code. Further, students were allowed to work with external companies/entities on their projects as a means of motivation and to provide more real-world experience (e.g., writing code in support of a favorite charity), and so was not always feasible to perform in a Colab environment. This made data collection on "start" times mostly meaningless, and further exacerbates the importance of true total time spent writing code over the gap between start and submission.

2.3 Institutional Review Board Approval

Our study considers human subjects (our students), and so was required to go through an Institutional Review Board (IRB) for approval. Our study was approved by the IRB based on two key factors: 1) Our study's design did not result in any change to student's grades, and was purely observational. 2) Our study did not infringe on any student's rights to privacy.

The later point is particularly important in consideration of limitations in the study's results. As we have stated and will emphasize again, our data does not reflect a granular measure of effort or time spent. We are thus unable to differentiate between a student starting early and spending only a few minutes a day, versus a second student who started later but spent the same amount of time in a single session, to complete the assignment. Getting information at a more granular level would not be passed by our IRB in discussion with them, and we will exemplify two common questions we have received that are not satisfiable by our IRB.

First, one may suggest that the students must engage in the use of some kind of version control system, with positive or negative incentives for frequent use of pushing code changes, such that the total time spent could be inferred from the edit history. This would be a noisy inference, but also would cause the study design to affect student grades or their perception of how they are graded. For this reason our IRB would not approve of a study with this kind of design.

Second, it has been suggested that students should be monitored continuously to track when and for exactly how long they are performing the assignments. Beyond the logistical difficulties of monitoring $\approx$25 students over several months for multiple semesters, this imposes a significant invasion of the student's privacy. Students have an expectation of privacy outside of the classroom, and the monitoring required not only violates that privacy, but is plausibly illegal. This avenue is thus also ill-advised.

For these reasons we find our approach satisficing in allowing us to study the question of interest, at a known and acknowledged level of imprecision. It is logistically feasible, does not alter student grades, and does not infringe on student's privacy rights.

3 Model

To study the impact of student start and "total" time spent on assignments toward the grade received, we will use a linear hierarchical Bayesian model. The overall plate diagram is given in fig. 2, and the generative story in algorithm 1. Capital Greek letters are used for hyper-priors and lower-case Greek letters for the priors. We use this

[2] Start time potentially impacted propensity to engage in academic misconduct, amongst other stressors with the pandemic. These considerations are critical but beyond our scope and data.

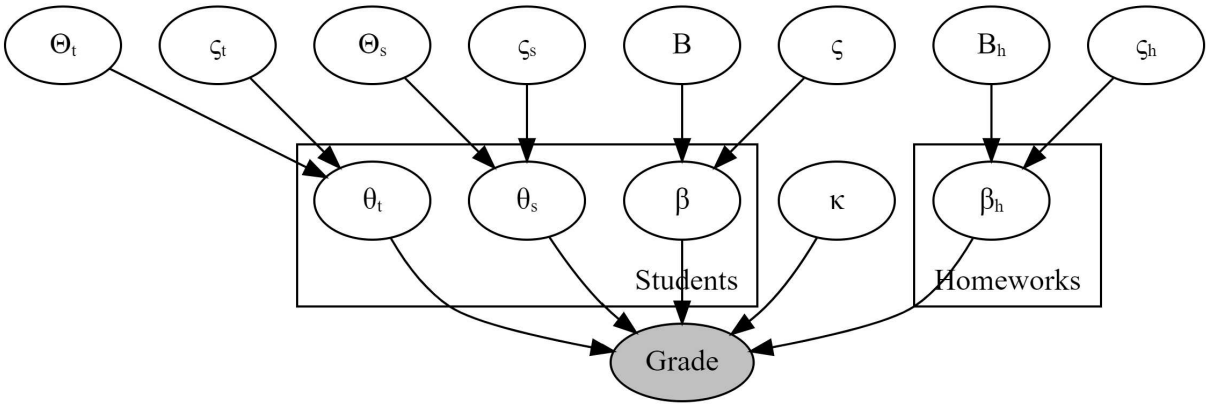

Figure 2. Plate diagram of the variables used in our model. The box in a plate diagram indicates a repeated measure, as each student receives their own individual bias β (how well does each individual student perform at baseline) and their own coefficients θ_t and θ_s describe their individual benefits from time spent on an assignment and submitting early. Similarly, each homework has a special bias term to quantify if certain homeworks required more time to complete than others. The top level hyper-priors describe the population level averages (Θ_t, Θ_s, B, B_h) and variances (ς_t, ς_s, ς, ς_h) for the variables they point to. The variance for the beta regression κ is inferred on its own as a property of the dataset as a whole.

hierarchical design because of the limited total amount of data, and our assumption is that there is shared information between students in behavior—but some are unique and should be modeled in such a way. Using a hierarchical model allows information sharing to occur across students and homeworks, while simultaneously allowing for variation between them [8]. At a high level, our model incorperates the following design factors with further explenation after.

1. A hierarchical linear model is used to follow best practices to incorporate information sharing (e.g., students working on the same assignment).
2. Each student has an independent bias term, which allows the model to account for intrinsic differences in capability to complete the assignment (regardless of the source of those differences, e.g., innate ability or prior exposure).
3. Using the population level hyper-prior to infer population level rates, and the per-student prior to allow for handling of per-student variance.

We will now detail the variables in our model and the logic behind their design. It also allows us to estimate credible intervals, that are a quantified estimate about the uncertainty of each hyper-prior (i.e., population level) and sample prior (i.e., student/homework level) to determine if there is a significant relationship, without being over-encumbered by multiple-test corrections increasing Type II errors in a more frequentist approach [9] . We will use the heavy tailed Cauchy distribution for all hyper-priors as it imposes minimal assumption of the population level values, and a Gaussian distribution for other priors as a reasonable default choice and we do not desire a heavy tail for the coefficients sampled from them.

First, we observe that some students often require less time than others to complete an assignment, and so we feel it would be inappropriate to use a single bias term. For this reason our model allows each of the j students to have their own bias term β^j, determined by its hyper-prior B. Similarly, the homeworks were designed with the intention that the first and last should be easier than the others. So we include additional homework-specific bias terms β_h^i for each of the i different homework assignments.

Using x_t^j and x_s^j to denote the jth's student total time and starting time respectively, we will have corresponding covariance θ_s^j and θ_t^j.

Algorithm 1 Generative Story

Input: Student start and total time spent on an assignment x_t and x_s for all students (index by j superscript).

$\Theta_t, \Theta_s, B, B_h \sim \text{Cauchy}(0, 1)$ # Location hyper-priors

$\varsigma_t, \varsigma_s, \varsigma, \varsigma_h, \kappa \sim \text{Cauchy}(0, 1)^+$ # Variance hyper-priors, truncated to non-negative values

for Each Homework i **do**

  $\beta_h^i \sim \mathcal{N}(B_h, \varsigma_h)$ # Each assignment gets a bias adjustment for difficulty

end for

for Each Student j **do**

  $\beta^j \sim \mathcal{N}(B, \varsigma)$

  $\theta_t^j \sim \mathcal{N}(\Theta_t, \varsigma_t)$

  $\theta_s^j \sim \mathcal{N}(\Theta_s, \varsigma_s)$

end for

$\hat{\mu}^{i,j} \leftarrow \sigma\left(\theta_t^j \cdot x_t^j + \theta_s^j \cdot x_s^j + \beta^j + \beta_h^i\right)$

measure likelihood against observed grades with eq. (1) using $\mu \leftarrow \hat{\mu}$ and concentration $\kappa \leftarrow \kappa$

Again these are student-specific, so that we may study if individual students benefit differently from having more time to work on an assignment. The means of the hyper-priors for these two covariates are then Θ_s and Θ_t, and we will look to the hyper-prior posterior after inference to answer the question: *do students at large benefit from more time spent on assignments*. Looking at the student specific θ_s^j and θ_t^j then tells us if students vary in their benefit of more time.

For all hyper-priors (upper-Greek) we sample the mean from a Cauchy distribution, and the variance parameter from the zero truncated Cauchy. This is done to impose little constraint on the location and variance of the hyper-prior. The prior variables (lower-Greek) are samples from Gaussian distributions $\mathcal{N}$(mean, variance) based on the hyper-priors.

The response variable of our model is treated as a Beta regression, and we use the proportional beta formulation as defined by eq. (1) that allows us to specify a mean μ and non-negative variance κ as it is easier to model[3].

[3] in this context B is the beta function, and is not used in this context anywhere else in the manuscript

$$\text{Beta\_Prop}(\theta|\mu,\kappa) = \frac{\theta^{\mu\kappa-1}\,(1-\theta)^{(1-\mu)\kappa-1}}{\text{B}(\mu\kappa,(1-\mu)\kappa)} \tag{1}$$

We use the Beta regression model as it is a popular means of regressing over $(0,1)$ constrained response variables and fits our grade distribution. We prefer to clip the maximum grade of 1.0 (i.e., 100%) to 99.9 and the minimum grade from 0.0 to 0.001, as adding a twice inflated Beta regression would result in a complex to specify model, and complicate analysis due to many 100% grades in our dataset. Functionally a 99% and 100% grade are equal demonstrations of content mastery, but a zero-one inflated model would treat these as meaningfully different events. To constrain our regression $\hat{\mu}$ of eq. (1) to the range $(0,1)$, we use the common sigmoid function as defined by $\sigma(x) = \frac{1}{1+\exp(-x)}$.

This fully specifies our model of student grades and the impact that time, measured by start time and total time spent, impacts students' grades. We use the Numpyro library [15] and the NUTS sampler [10] for our model's inference with 300 burn-in cycles and 600 samples after. This results in a $\hat{r} = 1.00$ for every parameter of the model, indicating full convergence.

4 Results

In this section we present our analysis of the results. We start with the posterior distribution of the hyper-priors shown with a 95% credible interval, which can be found in fig. 3. First are the global bias term B and the homework specific bias prior B_h. In each case the wide distribution indicates the variability in student behavior. More notably Θ_s and Θ_t show the impact of submitting early and total time spent on an assignment respectively. In both cases the impact is positive and significant as zero is outside the credible interval, with mean values of 0.78 per week early submission and 0.72 per week of additional total time. We remind the reader that this week early corresponds to *starting* the assignment a week early, and not a week of continuous effort.

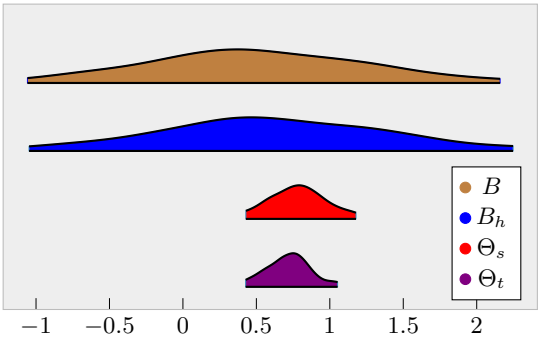

Figure 3. The 95% credible interval of the hyper-priors B, the global bias term, and B_h, the homework-specific bias prior. Θ_s and Θ_t show the impact of submitting early and total time spent on an assignment; the overall result is a statistically significant positive correlation between these factors and receiving a higher grade.

Crucially, this allows us to examine questions about what the average student can do to improve their grade. One way to look at this is the rate of growth for the function: $\sigma(0.52 + 0.72 \cdot x) - \sigma(0.52)$. This thus returns the impact of starting the assignment earlier, assuming the student will spend all available time to complete it (i.e, submits at the last minute the homework is due). In this form we can infer that starting $x = 2$ weeks early instead of $x = 1$ could yield a 10% improvement in average grade received. This will invariably be affected by individual student performance, and so we must also ask about the distribution of individual students.

Because Θ_s and Θ_t are hyper-priors over the student specific distributions of θ_s and θ_t, we can look at these later distributions to understand the variability of impact. First we consider the total time spent on an assignment θ_t in fig. 4.

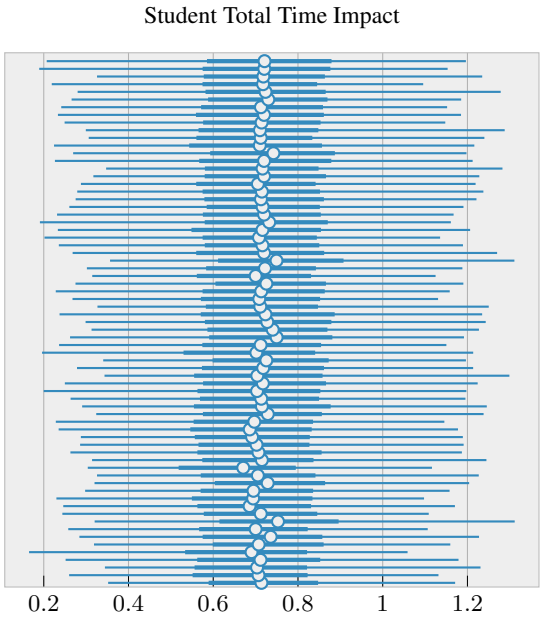

Figure 4. Forest plot of the 95% credible interval of the student specific distributions θ_t that measure the impact of spending more total time on their final grade. Each line represents a different specific student, the the circle showing the median, thick blue lines showing the middle 50% of the interval, and the thin blue lines showing the full 95% credible interval. Results suggest a very consistent cross-population benefit to spending more total time on assignments.

This plot shows the surprisingly consistent benefit that the student receives by spending a week's worth of time on each assignment. This would seem to imply that the benefits are stable and repeatable, and that given we model the problem with a sigmoid σ we suspect corresponds to an implied diminishing return on the benefits of spending more time studying. This would also correspond to the data as shown in fig. 1, where after starting 200 hours (1.2 weeks) before the deadline all students obtain $\geq 85\%$ on their assignments.

We note that the results in submission time are also highly consistent, as shown in fig. 5. We suspect this is an effect that the submit time x_s can be negative when students submitted late using the 72 hour late policy, as shown in fig. 1. Students *starting* late never received a score better than a 70% before the late penalty was applied. While not an original goal of our study, this does lead us to question the ultimate utility of the late submission policy. If removing the policy would encourage more students to start earlier, because the "backup" of using the late policy does not exist, we may obtain better total outcomes for all students. Simultaneously, our subjective feedback from student reviews and course evaluation is that the late policy is highly appreciated, and could lead to better performance via engagement. Answering this question is beyond the scope of this study, but an important point of future work identified by our data.

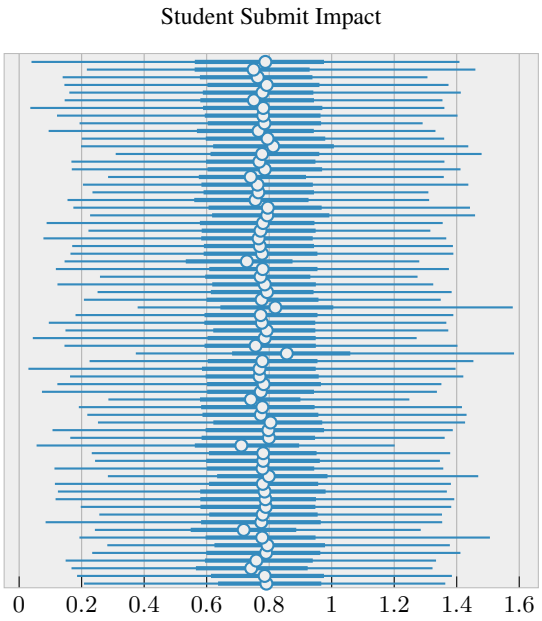

Figure 5. Forest plot of the 95% credible interval of the student specific distributions θ_s that measure the impact of submitting the homework earlier on individuals' grade. Each line represents a different specific student, with the circle showing the median, thick blue lines showing the middle 50% of the interval, and the thin blue lines showing the full 95% credible interval. Results suggest a consistent correlation between earlier submissions and improved outcomes.

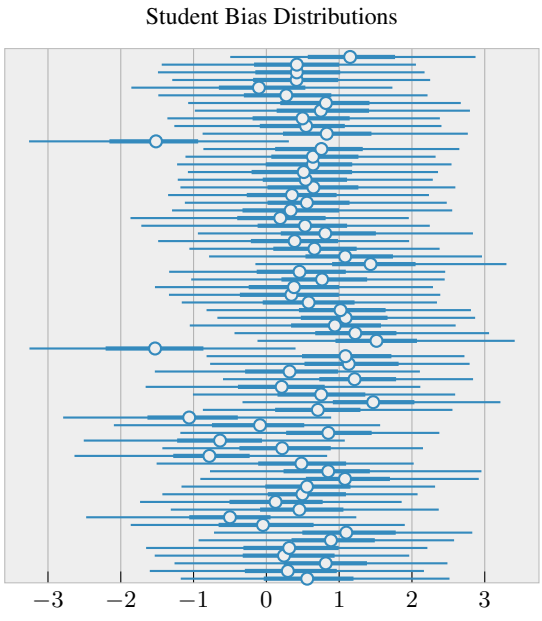

Figure 6. Forest plot of the 95% credible interval of the student specific distributions β that measure the student specific bias term on individual performance across all assignments. Each line represents a different specific student, the the circle showing the median, thick blue lines showing the middle 50% of the interval, and the thin blue lines showing the full 95% credible interval. Students' individual performance bias varies, suggesting that students vary significantly in effort required to complete assignments to a similar level.

While submission and total time have stable distributions across students, students' individual performance biases display more asperity as shown in fig. 6. In the more extreme cases two students had ≈ -2.5 bias terms, placing them at a deficit of 3 weeks time compared to the mean student. In such a case this would require the student to start each homework assignment immediately in order to obtain the same outcomes. While technically feasible for the course as administered, this requires no other heavy work loads from the student's other courses throughout the semester, and is not realistically possible on all assignments.

This result leads us to question what interventions may be possible to help such students. In our small study, these students are post-hoc identifiable by their performance on the first homework, where most students receive a perfect grade. We lack sufficient examples of such students to statistically confirm that this is reliably the case, but implies the possibility for early intervention may be possible.

5 Other Related Work

In our introduction we reviewed two areas of education research that inspire and informed our study. We note that there is little other work regarding the broader question of time to complete or implement deep learning. There has been limited work in studying the amount of time students spend on coding assignments, but most studies are with respect to introductory computer science courses [21]. Empirical reproducibility work has used survival analysis to study the time it takes to replicate machine (and deep) learning papers, finding that most can be done quickly, but a long and heavy tail exists in the effort required [18, 17, 20]. However, such work is focused on replicated academic peer-reviewed papers, rather than curated assignments in a course. That said, the gap between deep learning graduate course work and implementing academic papers is not too large, and while the annual ML Reproducibility Challenge [22] has demonstrated that students can succeed at such tasks, the additional information of time required has not been recorded. Adding such information to future years could prove valuable to reproducibility work and potentially inform the design of final course projects where attempting to replicate a paper is of a more appropriate scope.

Towards potential interventions, previous work has found that explicitly teaching students how to debug their own code lessens the time they spend on coding assignments [2]. It may be possible to develop similar interventions for deep learning, but most work in "debugging" deep learning is still oriented toward researchers[4], and it is not clear to us if the requisite stability in tools and techniques have been distilled to a level for students. We also note that while we are the first (to the best of our knowledge) to estimate time spent on an assignment from version history, we are not the first to use version history in relation to student assessment. In particular, [13] attempted to use version history to try and predict student outcomes on a given assignment, but found little predictive value. While such results could be improved today with better methods for predictive modeling, our more immediate concern would be on how to more precisely quantify actual time spent. Concerns about the use of Large Langage Models (LLMs) may also impact future results [1, 16, 4], which was not a popular tool at the time this study was done.

6 Conclusion

We have conducted the first study of the impact of start time (i.e., starting a homework early) and the total time spent on a deep learn-

[4] https://debug-ml-iclr2019.github.io/

ing assignment on the grade received on the assignment. We find that both factors are statistically significantly correlated with improved scores by a student, which—while intuitive—is not identical to results found in other populations using different methodologies. The current data suggests that students do vary significantly in the base amount of time needed to complete an assignment, but the unit benefit of improvement is remarkably stable across students. This suggests that improved identification and intervention for students who require more time to complete assignments is worth further investigation. Our results suggest the average student may be able to improve their grade by 10% by starting a week earlier than they would otherwise.

Notably, our study is limited to students at a single institution, during three semesters of the COVID-19 pandemic. As such, our results may have selection bias and additional pandemic-related confounding factors. Simultaneously, the current environment suggests that we may unfortunately be operating in a future in which COVID is endemic, and so some of these biases may be relevant to future education concerns. Comparing results across curricula for courses at different institutions, and developing more precise methods of quantifying the actual total time spent, rather than our proxy measure from edit history, are directions for future work to improve upon.

References

[1] Stella Biderman and Edward Raff, 'Fooling moss detection with pretrained language models', in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, CIKM '22, p. 2933–2943, New York, NY, USA, (2022). Association for Computing Machinery.

[2] Ryan Chmiel and Michael C Loui, 'Debugging: From Novice to Expert', in *Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '04, p. 17–21, New York, NY, USA, (2004). Association for Computing Machinery.

[3] Sophie H. Cormack, Laurence A. Eagle, and Mark S. Davies, 'A large-scale test of the relationship between procrastination and performance using learning analytics', *Assessment & Evaluation in Higher Education*, **45**(7), 1046–1059, (10 2020).

[4] Paul Denny, James Prather, Brett A Becker, James Finnie-Ansley, Arto Hellas, Juho Leinonen, Andrew Luxton-Reilly, Brent N Reeves, Eddie Antonio Santos, and Sami Sarsa, 'Computing education in the era of generative ai', *arXiv preprint arXiv:2306.02608*, (2023).

[5] Rubén Fernández-Alonso, Javier Suárez-Álvarez, and José Muñiz, 'Adolescents' homework performance in mathematics and science: Personal factors and teaching practices.', *Journal of Educational Psychology*, **107**(4), 1075–1085, (2015).

[6] Barbara Flunger, Ulrich Trautwein, Benjamin Nagengast, Oliver Lüdtke, Alois Niggli, and Inge Schnyder, 'Using Multilevel Mixture Models in Educational Research: An Illustration with Homework Research', *The Journal of Experimental Education*, **89**(1), 209–236, (1 2021).

[7] Mollie Galloway, Jerusha Conner, and Denise Pope, 'Nonacademic Effects of Homework in Privileged, High-Performing High Schools', *The Journal of Experimental Education*, **81**(4), 490–510, (10 2013).

[8] Andrew Gelman, John B. Carlin, Hal S. Stern, David B Dunson, Aki Vehtari, and Donald B Rubin, 'Bayesian Data Analysis Third edition (with errors fixed as of 13 February 2020)', (February), 677, (2013).

[9] Andrew Gelman, Jennifer Hill, and Masanao Yajima, 'Why We (Usually) Don't Have to Worry About Multiple Comparisons', *Journal of Research on Educational Effectiveness*, **5**(2), 189–211, (4 2012).

[10] Matthew D Hoffman and Andrew Gelman, 'The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo', *Journal of Machine Learning Research*, **15**(47), 1593–1623, (2014).

[11] Irma S Jones and Dianna Blankenship, 'Year two: Effect of procrastination on academic performance of undergraduate online students', *Research in Higher Education Journal*, **39**, (2020).

[12] Irma S. Jones and Dianna C. Blankenship, 'The Effect of Procrastination on Academic Performance of Online Students at a Hispanic Serving Institution', *Journal of Business Diversity*, **19**(2), (7 2019).

[13] Keir Mierle, Kevin Laven, Sam Roweis, and Greg Wilson, 'Mining Student CVS Repositories for Performance Indicators', *SIGSOFT Softw. Eng. Notes*, **30**(4), 1–5, (5 2005).

[14] Gulnar Ozyildirim, 'Time Spent on Homework and Academic Achievement: A Meta-analysis Study Related to Results of TIMSS', *Psicología Educativa*, **28**(1), 13–21, (11 2021).

[15] Du Phan, Neeraj Pradhan, and Martin Jankowiak, 'Composable Effects for Flexible and Accelerated Probabilistic Programming in NumPyro', *arXiv*, 1–10, (2019).

[16] James Prather, Brent N Reeves, Paul Denny, Brett A Becker, Juho Leinonen, Andrew Luxton-Reilly, Garrett Powell, James Finnie-Ansley, and Eddie Antonio Santos, '" it's weird that it knows what i want": Usability and interactions with copilot for novice programmers', *arXiv preprint arXiv:2304.02491*, (2023).

[17] Edward Raff, 'A Step Toward Quantifying Independently Reproducible Machine Learning Research', in *NeurIPS*, (2019).

[18] Edward Raff, 'Research Reproducibility as a Survival Analysis', in *The Thirty-Fifth AAAI Conference on Artificial Intelligence*, (2021).

[19] Edward Raff. Inside deep learning: Math, algorithms, models, Apr 2022.

[20] Edward Raff and Andrew L. Farris, 'A siren song of open source reproducibility, examples from machine learning', in *Proceedings of the 2023 ACM Conference on Reproducibility and Replicability*, ACM REP '23, p. 115–120, New York, NY, USA, (2023). Association for Computing Machinery.

[21] Mark Segall, 'How Much Time Do Students Spend On Programming Assignments? A Case for Self Reporting Completion Times', in *Proceedings of the EDSIG Conference ISSN*, volume 2473, p. 3857, (2016).

[22] Koustuv Sinha, Jesse Dodge, Sasha Luccioni, Jessica Zosa Forde, Sharath Chandra Raparthy, Joelle Pineau, and Robert Stojnic, 'ML Reproducibility Challenge 2021', *ReScience C*, **8**(2), (2022).