2023 Annual Conference & Exposition
Baltimore Convention Center, MD | June 25 - 28, 2023
The Harbor of Engineering Education for 130 Years
ASEE
Paper ID #39177

A Tool for the Discovery of Academic Misconduct in Online Assessments Using Student Activity Logs

Dr. Paul David Gordon, University of Cincinnati

Dr. Gordon is an Assistant Professor Educator at the University of Cincinnati, where he serves as the co-director of the Medical Device Innovation & Entrepreneurship Program in the Department of Biomedical Engineering. His research interests include evidence-based pedagogy, development of entrepreneurial teaching methods, diagnostic technologies, and global healthcare technology innovation.

Mary Preston McDougall

A Tool for the Discovery of Academic Misconduct in Online Assessments Using Student Activity Logs

Paul Gordon[1], Mary P. McDougall[2]

1) University of Cincinnati; Department of Biomedical Engineering
2) Texas A&M University; Department of Biomedical Engineering

Abstract
As the landscape of higher education evolves in a post-pandemic era, the use of hybrid and online forms of instruction and assessment continues to proliferate. In the midst of this ever-changing landscape, educators are challenged to maintain the integrity of educational systems and assessments. This work presents a novel tool for the discovery of prohibited collaboration during online examinations by comparing activity logs for student work during examination periods. The tool was developed during the COVID-19 shutdowns of spring semester 2020 and was successful in detecting significant incidents of academic misconduct in an undergraduate biomedical engineering class. The background, context, use, mechanisms, functionality, and potential impact of the tool are explored with the hope that the approach may be expanded and applied to enhance the integrity of various areas of higher education in the future.

Keywords
Misconduct, Assessment, Online, Remote, Examination, Tool, Integrity, Collaboration, Cheating, Screening

Introduction / Background
Whether it is due to the COVID-19 pandemic, or the looming prospect of artificial-intelligence augmented instructional techniques, the current dynamics of higher education make it critical to assess and respond to patterns of student behavior using electronic and remote media so that best practices can be established for future use. This is especially important for issues of academic misconduct so that the integrity of courses can be preserved during a time of rapid adjustment to remote learning. The use of electronic remote examinations was thrust to the forefront of higher education during the spring of 2020 when the COVID-19 pandemic prompted rapid lockdowns around the world and caused millions of students to transition to online-only education. The landscape has, for the time, stabilized in the subsequent years since lockdowns were first enforced, but electronic course administration is here to stay, with many challenges and opportunities yet to be solved [1]. While the shift in pedagogical modality was fraught with myriad challenges, the proper administration of assessments was central among them. Various studies have reported that incidence of academic misconduct during online examinations, specifically, has likely increased significantly since the pandemic began on a global scale in 2020, but the exact proportion of the increase varies significantly across contexts and may not extend to assignments or other classroom

activities [2, 3, 4]. It is not clear to what degree these trends have reversed since returning to higher rates of in-person learning with increased COVID-19 endemicity and vaccine availability.

While there have been significant efforts to create online examination proctoring tools on a global scale [5,6], the prospect of meaningfully proctoring remote exams using any sort of camera-based feed has been recently called into question in the United States in the aftermath of the Ogletree v. Cleveland State University ruling that declared that the use of a camera to ensure that a student's test area is free of prohibited personnel or material constitutes an illegal invasion of privacy [7]. The difficulty in directly monitoring student participation in remote exams places a higher impetus on the detection of academic misconduct during grading or in retrospect. Numerous approaches have been proposed, including the use of machine learning algorithms to detect anomalous performance [8], review of internet traffic data and integrated authentication methods [9], and multifaceted retrospective analyses of examination results [10-12]. An excellent review of the scope of online cheating research can be found in the 2022 publication by Noorbehbahani, et al. [5].

The purpose of this work is to describe an academic integrity tool developed to screen for potential cheating on a remote, take-home exam during the COVID-19 pandemic. The tool can be categorized as an after-exam, log-of-time analysis method. The work is a retrospective research assessment of an integrity preservation technique. It is a far more extensive exploration of the utility of time-log tracking than previous works that have simply considered overall test time in analysis to discover cheaters [9]. To understand the discovery and merits of the technique, a case study is presented in which the student population engaged in widespread academic misconduct during a high stress setting. The data are sourced from a required junior-level biomedical engineering course administered at Texas A&M University – College Station during the Spring 2020 semester. During the Spring 2020 semester, undergraduate students at Texas A&M University completed the first eight weeks of classes in-person before the advent of the COVID-19 pandemic required their departure from campus and completion of the term in a remote format. A required junior-level engineering course was taken by 152 students, which were split between two identically taught sections. After course transition to an online format, all students were combined into one virtual classroom to facilitate live instruction. During the term-end examination, which was delivered as an online "take-home" assessment, numerous accounts of student collaboration were discovered, which was prohibited by the examination rules that allowed use any resource, provided that the exam was completed solo. It is believed that the results of this work can inform best practices for remote assessment administration. Additionally, the computational methodology used to identify students engaging in academic misconduct may be a valuable resource for other instructors during future teaching and may be useful for integration in future iterations of electronic learning management platforms.

Materials & Methods

The online assessment was hosted through the Blackboard/eCampus online platform during a 36-hour window beginning at noon on Day 1 and ending at midnight at the end of Day 2, during which students were able to begin, stop, resume, and submit the exam at any time. The exam was composed of 59 total questions (16 matching, 7 multiple-choice, 29 fill-in-the-blank, 1 shortanswer, and 6 calculation), and questions were organized so that students could view all questions simultaneously in non-randomized order. Cheating on the exam was first suspected when several patterns of matching unique incorrect answers were noticed while grading question-by-question. Course instructors decided to pursue a more systematic way to determine whether cheating likely occurred due to the large class size and a desire to be judicious in administering punishments. To create a systematic approach to check for whether cheating likely occurred for these similar errors, instructors used the primary additional dataset provided for the assessment - the access logs provided by eCampus. The chosen approach was to explore whether students had worked on the same questions simultaneously. As students take an exam online through eCampus, their actions (clicks) are logged by the software. This record can be used to create a timeline of which questions were being worked on for every second of the examination window for each student. The access logs contain timestamp, question number, and action type data for each student interaction event with the online software.

During initial investigation of the access logs, several strong correlations between sets of student access logs were noticed. However, it was difficult to quantify the degree of similarity between access logs manually due to the large size of the dataset. To solve this problem, a MATLAB script was written to find and quantify similarities between all access logs for the entire class roster. The analysis code assigned a unique “correlation score” to each unique student pair in the class, allowing the results to be used as a screening tool to identify students with high likelihood of having cheated on the exam. Essentially, the tool works by running a modified autocorrelation algorithm over the aggregate exam completion timelines of all students in class. This tool is meant to be used primarily as a screening tool, not as definitive proof in its own right.

Table 1: Example of Student Activity Log Data (simulated data)

Timestamp	**Action**	**Time After Exam Start**	**Action Time**
4/21/20 1:47:07 PM	Saved question 16 multiple times over a period of: 00:01:09	04:11:46	00:19:59
4/21/20 1:53:17 PM	Saved question 17	04:17:56	00:06:10
4/21/20 1:57:15 PM	Saved question 21 multiple times over a period of: 00:00:57	04:21:54	00:03:57
4/21/20 2:02:22 PM	Saved question 18	04:27:01	00:05:06
4/21/20 2:02:46 PM	Saved question 3	04:27:25	00:00:24
4/21/20 2:02:46 PM	Saved question 4	04:27:25	00:00:00

The MATLAB script initializes by importing each individual student dataset as text strings, then parsing the date, time, question number, and duration values from each individual action entry. The script fundamentally operates by using the parsed access logs to create a unique action timeline for each student second-by-second for the entire 36-hour exam window. Initial timelines were created by directly mapping all active entries into a 36-hour timeline made up of 129,600 time-steps, one for each second of exam time. Each second of the action timelines contains a descriptor of the current student state at that moment, including whether the student was active or inactive, and, if active, which specific question they were actively working on, as determined by their last known click within the exam software. The raw dataset was modified to remove times in which students were suspected to be inactive during the exam by creating a maximum question-length threshold above which it is assumed that students were away-from-keyboard (AFK) and thus considered inactive. After the AFK times were removed, each timeline was cross correlated to all others in the class to all specific seconds during which two students were actively working on the same question at the same time. The sum-total of all seconds of shared activity between every student pair was assigned as that pair's "correlation score". With 152 students in the class, the size of the initial autocorrelation matrix was stored in a 152-by-152 array describing the total course correlation score dataset. Within this matrix, correlation scores that were significantly above the noise floor were flagged for further analysis, thus creating a screening tool with which to identify cheaters in a novel way. Following the end of course administration, IRB approval was approved to further analyze the dataset alongside additional course parameters. The MATLAB script was then modified to analyze the expanded dataset containing de-identified student information and overall course records.

Results

After using the access log analysis tool to screen for possible collaborators on the exam and confirming each suspected case of cheating by manually comparing the exam responses for flagged student pairs, 42 out of 152 (28%) students were initially identified with convincing evidence that they engaged in prohibited collaboration on the exam. When given the option to use a poll to self-admit and receive a lessened penalty, a total of 87 students self-admitted to collaborating on the exam, 39 denied collaborating, and 26 chose not to respond. Notably, eight of those who denied collaborating were among those with convincing evidence against them. These results are summarized in Figure 1 and indicate that at least 62% of students in the course likely engaged in prohibited collaboration of some type, and that the screening script was successful in identifying approximately 44% of all students who cheated on the examination.

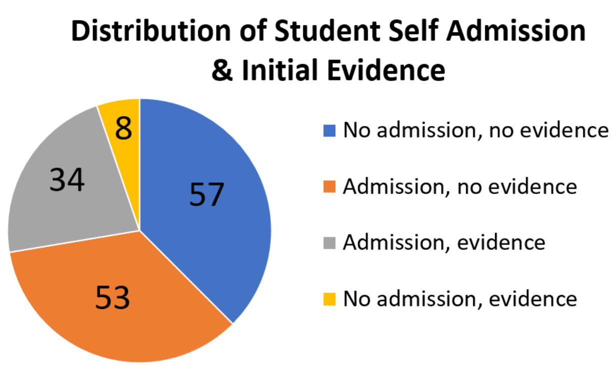

Figure 1: The number in each grouping of self-admission/evidence

From the results of the 152 x 152 cross-correlation matrix, 23,104, or 152^2 correlation scores were generated, as can be seen in Figure 2. Of these, 152 were autocorrelations generated when a student's activity timeline was compared to itself, which were retroactively set to null correlation score. The remaining 22,952 scores contain duplicates of each pairing, resulting in a total of 11,476 individual correlation scores from a class of 152 students. Naturally, the number of unique correlation scores for a class of any size is simply given by

$$n_{correlation\ scores} = \frac{(class\ size)^2 - (class\ size)}{2}. \qquad \textit{Equation 1}$$

As mentioned previously, two sets of correlation scores were created, one with original timeline data, and one with suspected AFK times removed. Details of and comparisons between the two sets of correlation scores can be observed in Table 1 and Figure 2. The units of all correlation scores is *(seconds)*. It was found that use of the AFK-removed dataset was preferable for use in screening for potential cheaters, as it removed many of the false-positive high correlation scores resulting from when two students arbitrarily happened to stop work on their exams on the same question number for an extended period of time. Therefore, the AFK-removed dataset is the primary that is analyzed in this work, unless specifically noted otherwise. To attempt to detect students who may have been working together but completed questions subsequently as opposed to concurrently, the tool was modified to detect students who worked on questions within 5 minutes of one another. No significant increase in prevalence of cheating occurrence was observed. All data presented here are for second-level concurrent problem work only, with no buffer time allowed during analysis.

Table 2: Statistics of Correlation Score Datasets

Dataset	**Mean Correlation Score (*seconds*)**	**Standard Deviation**	**Median**	**Mode**	**Min**	**Max**
Original Timeline	2,900	8,478	683	25	1	84,093
Timeline w/ AFK removed	304	723	73	26	1	16,002

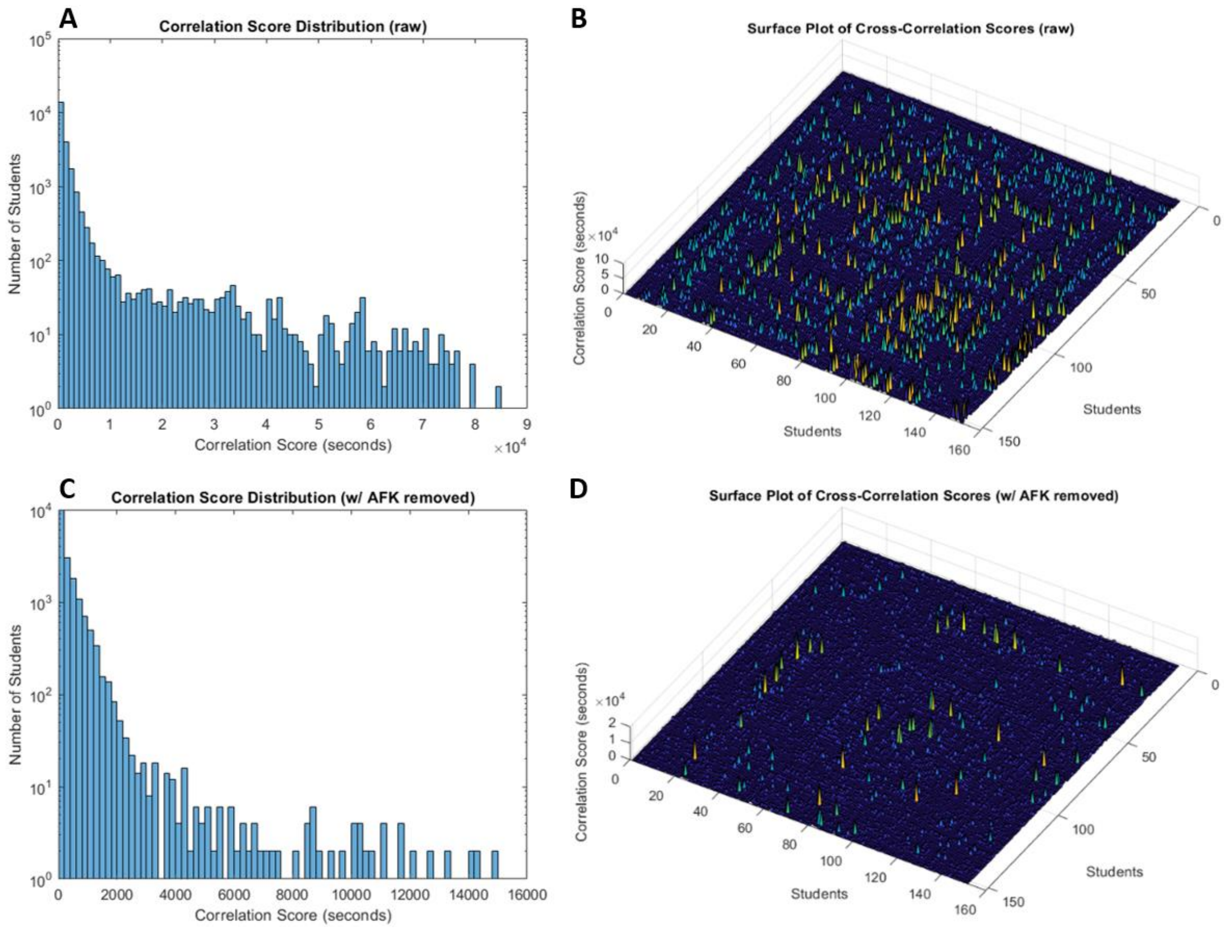

Figure 2: Distribution of correlation scores for the **A)** initial timeline dataset. **B)** 152 x 152 matrix of correlation scores displayed as a surface map for the initial timeline dataset. **C)** Distribution of correlation scores for the secondary dataset, after suspected away-from-keyboard (AFK) times were removed. **D)** Surface matrix of correlation scores for the secondary, AFK-removed dataset. Peaks in the surface represent likely collaborators.

Discussion

The use of the electronic access log data to screen for students who collaborated on remote examinations found a surprisingly high percentage of students who cheated, certainly more than would have been found using conventional grading methods. In a setting where students were not aware that they could be held accountable for collaborating on a remote examination, use of the tool provided a way to directly detect collaboration by 28% of the class, with another 35% of the class persuaded to self-admit to collaborating when faced with the possibility of punishment for their actions. This behavior indicates that, when presented with what they believed to be an opportunity to cheat without repercussions, at least 63% of all students in this course engaged in some form of academic misconduct. The course instructors believe that this high percentage of students engaging in academic misconduct was likely inflated due to the novelty of the stressful circumstances placed upon the students due the necessity to unexpectedly move courses to an all-online format mid-semester, in addition to the global stresses of living in a world currently in a pandemic. It was anecdotally observed that students felt especially burdened by the amount of additional work required of them as a result of the sudden conversion to remote learning, which could increase the likelihood of them to engage in academic misconduct as a shortcut to achieving desired course grades.

In its current form, the screening tool for collaboration may be most useful as a deterrent for academic misconduct. The twofold increase in the number of students who voluntarily self-reported collaboration after there was apparent evidence of cheating indicates that students believe that they can be caught. This conclusion, then, raises an important further question: "Why would students cheat if they believed they could be caught?" The authors hypothesize that, in short, they believed that they wouldn't be caught, despite the possibility of detection. In the pragmatic, day-to-day life of engineering study and instruction, the detection and prosecution of academic misconduct takes significant effort on the part of faculty, and students understand that oftentimes they will not be caught in cheating because the threshold of effort required to catch them will not be surmounted. For this reason, automated detection systems that can lower the administrative/sleuthing burden of detecting misconduct may lead to students being held accountable more frequently and overall support a culture of intolerance for such behavior. If students were made aware that "smart" tools were used to detect cheating, it may theoretically discourage the behavior in the first place. Indeed, if students are brought up throughout their educational careers in an ecosystem with mature, reliable tools in place to discover academic misconduct, then a culture that such behavior is impermissible may root out all but the most malicious incidences.

The time-log analysis tool does have potential drawbacks as well. Notably, its effectiveness in a variety of settings and compatibility with various classes and question types has not been studied. A larger-scale, diverse rollout of the method with relevant control data would be necessary to understand how its utility varies between application settings. Longitudinal monitoring of detection rates also may be merited, as students may adjust their behavior over time to simply avoid the appearance of cheating to this algorithm. For this reason, it may be detrimental to over-inform students of the way their collaboration can be detected. If students understand how their collaboration index scores are calculated, they can easily beat the current system by actively avoiding synchrony in their completion efforts. Perhaps the most difficult aspect of measuring long-term effectiveness would be separating potential decrease in the tool's effectiveness due to behavioral changes from the desired improvement in reduced cheating incidence.

It is also important to note that this paper focusses primarily on the technology and method itself and has no way of measuring the possible impact on students' subjective experience of a classroom environment, their overall learning performance, or disproportionate effects on diverse student population subgroups. The authors hypothesize that improved automatic detection tools would have greater impacts on the equity of a classroom than on its inclusivity due to their nature as administrative, behind-the-scenes actors that rarely affect the day-to-day operations of a course environment. Although the overall implementation of systematic, automated tools for detecting academic misconduct may eliminate subjective bias from faculty during grading, it is also possible that the screening method proposed here may detect misconduct from students of certain populations more than others based on a variety of factors. The potential for bias must be understood in a broader context of how it compares to possible bias that exists currently, with less automated detection methods. Further evaluation is merited, especially if it is to be combined with machine-learning based detection tools that are known to be able to impart bias based on training dataset quality.

The tool is easily generalizable to any educational program, major, and level – it is completely agnostic to the type of work being done and could as easily be applied to grading work done in a primary school English course as it is to a graduate level engineering examination. It is envisioned that this tool would be most useful if directly integrated into a suite of academic integrity preservation tools within an electronic learning management platform such as eCampus or Canvas. Currently, the most time-consuming parts of the process are the manual data copying from eCampus before running the script and the manual side-by-side comparisons necessary to confirm academic misconduct after students have been flagged. The most difficult part of the comparison code is to parse the data accurately prior to comparison. If automated data export/extraction from eCampus could be used via built-in-features or use of a web-crawler, the process would be greatly expedited. Additionally, it would simplify the process of comparing between student similarities in responses, as any problems that were coincidentally completed by two students could be automatically displayed side-by-side to allow for simple verification by grading personnel. It is not difficult to imagine that this approach could be integrated in existing systems and eventually used as input into artificial intelligence-based grading and integrity management tools that are likely being developed even now.

Conclusion

This work presents a case study in which students engaged in academic misconduct during a high stress setting due to the COVID-19 pandemic. The purpose of the work is to describe an academic integrity tool developed to screen for potential cheating on a remote, take-home exam during the pandemic. The tool is based on a MATLAB script that analyzes the exam completion timelines of all students in a class to find and quantify similarities between their access logs. The tool identified 42 out of 152 (28%) students with convincing evidence of collaboration during the online exam, and an additional 53 students later admitted to also collaborating on the exam. The results of the tool usage presented here may inform best practices for remote assessment administration, and the computational methodology used to identify students engaging in academic misconduct may be a valuable resource for other instructors during future teaching. The next steps to continue developing and leveraging the approach include the further development of the code to better recognize student access types and away-from-keyboard times and the integration into electronic learning management platforms.

Disclaimers & Ethical Statements

Funding: Not applicable

Conflicts of interest/Competing interests: The authors of this study also taught the course under study.

Ethics approval: Research conducted retrospectively under IRB approval through Texas A&M University.
Consent to participate: Not applicable (exempted through IRB approval)
Consent for publication: Publication was approved by IRB board.
Availability of data and material: All student record data were de-identified and approved for FERPA compliance by Texas A&M University's Office of the Registrar.

References

[1] M. Itani, S. Kaddoura, & F. al Husseiny. “The impact of the Covid-19 pandemic on on-line examination: challenges and opportunities,” Global Journal of Engineering Education, 24(2), 105–120, 2022.

[2] B. Ives, & A.-M. Cazan. “Did the COVID-19 pandemic lead to an increase in academic misconduct in higher education?” Higher Education, 2023. [Online]. Available: https://doi.org/10.1007/s10734-023-00996-z

[3] B. Jeffries, T. Baldwin, & M. Zalk. “Online examinations in a large Australian CS1 course,” ACM International Conference Proceeding Series, 20–26, 2022. [Online]. Available: https://doi.org/10.1145/3511861.3511864

[4] L. Elsalem, N. Al-Azzam, A. A. Jum’ah, & N. Obeidat. “Remote E-exams during Covid-19 pandemic: A cross-sectional study of students’ preferences and academic dishonesty in faculties of medical sciences,” Annals of Medicine and Surgery, 62, 326–333, 2021. [Online]. Available: https://doi.org/10.1016/j.amsu.2021.01.054

[5] F. Noorbehbahani, Aj. Mohammadi, & M. Aminazadeh. “A systematic review of research on cheating in online exams from 2010 to 2021,” Education and Information Technologies, 27(6), 8413-8460, 2022.

[6] M. B. Hidayanto, S. Prayogo, & M. Lubis. “Designing Web Based Proctoring System for Online Examination (SPIRIT 1.0) in Telkom University,” ACM International Conference Proceeding Series, 41–48, 2022. [Online]. Available: https://doi.org/10.1145/3571662.3571669

[7] J. Calabrese. “Aaron M. Ogletree v. Cleveland State University,” United States District Court Northern District of Ohio Eastern Division 547, 2022.

[8] F. Kamalov, H. Sulieman, & D. S. Calonge. “Machine learning based approach to exam cheating detection,” PLoS ONE, 16, 2021. [Online]. Available: https://doi.org/10.1371/journal.pone.0254340

[9] J. Moten Jr, A. Fitterer, E. Brazier, J. Leonard, & A. Brown. “Examining online college cyber cheating methods and prevention measures,” Electronic Journal of E-learning, 11(2), pp139-146, 2013.

[10] R. Bawarith, A. Basuhail, A. Fattouh, & S. Gamalel-Din. “E-exam cheating detection system,” International Journal of Advanced Computer Science and Applications, 8(4), 2017.

[11] J. H. Mott. “The detection and minimization of cheating during concurrent online assessments using statistical methods,” The Collegiate Aviation Review International, 28(2), 2010.

[12] C. Ng. “Evaluation of academic integrity of online open book assessments implemented in an undergraduate medical radiation science course during COVID-19 pandemic,” Journal of Medical Imaging and Radiation Sciences, 51(4), 610–616, 2020. [Online]. Available: https://doi.org/10.1016/j.jmir.2020.09.009

Supplemental Information

Access Log Analysis Tool Use Procedure:

Step 1:

Copy access log data from eCampus to Excel by navigating to the exam attempt of the first student on the roster. After selecting all data in the access log for that student, the data are copied into Excel as text, and the student's student ID (UIN, name, etc.) are logged above the first column of data. This process is repeated as necessary, leaving one column of space between each student's data. The final dataset comprises all raw data from eCampus in a single spreadsheet and is saved.

Step 2:

Import the dataset into MATLAB Using the "Import Data" tool. The Excel file where data is stored is selected, and data are imported into MATLAB as cell arrays.

Step 3:

The MATLAB code is run. The code displays an initial prompt for user input at the beginning of use to define class parameters, and it again asks for user input at the end of completion to define a threshold of collaboration index cutoff. The code takes several minutes to execute, depending on the computer processing speed and the size of the dataset.

Step 4:

Review code output. The code displays four figures by default. In the Output Figure 1, a histogram plot is shown of the amount of time students spent on the exam. In Output Figure 2, a surface plot of cross-comparison scores for all members in class is displayed. In Output Figure 3, a histogram of cross-comparison scores can be seen and used to assign a "cut-off" value to enter into the later user prompt.

Step 5:

An Excel sheet will be automatically saved into the MATLAB file directory containing the thresholded student pair data. The pair data show the identifiers of all students whose collaboration index was above the defined cutoff. The list is organized in descending order of correlation index magnitude.