Mining Course Groupings using Academic Performance

Daniel D. Leeds, Tianyi Zhang, Gary M. Weiss
Department of Computer and Information Science
Fordham University, New York, NY
{dleeds, tzhang130, gaweiss}@fordham.edu

ABSTRACT

This study computes the correlation of student grades between pairs of courses in a large university. Course network graphs are then generated, where courses are represented as nodes and courses are connected if they have a high degree of grade correlation. Graph mining and network analysis tools visualize the course networks, identify course clusters and course cliques, and compute informative network statistics. Results are analyzed for pairs of courses and courses grouped by academic department or program of study. Strong course similarity groupings are observed within scientific disciplines, between pre-health courses, and within subfields of computer science. No prior study using this notion of course similarity has been conducted.

Keywords

Educational, Clustering, Correlation, Network graphs

1. INTRODUCTION

This paper describes a method for grouping and analyzing courses based on similar student performance, where similarity is measured between pairs of courses using the Pearson correlation of the grades assigned to students who take both courses. A graph is then formed that represents courses as nodes, and has edges between course-pairs when the student grade correlation is above a specified threshold. The resulting graph is then analyzed using a variety of graph analysis techniques, to provide insights into the relationship between individual courses and course groupings. Data preprocessing steps are described to handle confounding factors, such as differing instructor grading schemes. The methodology is encapsulated in a software tool that was developed for this study and is publicly available [5]. This study utilizes eight years of undergraduate student course grade data from Fordham University. The results show that there are strong connections between pre-health courses and courses within subdisciplines of computer science, and that courses that teach specific skills are much more highly connected to other courses than introductory survey courses.

The knowledge gleaned from this research can be used to influence curriculum design and academic policies. For example, if a student performs poorly in the first course within a set of highly correlated courses, then they are likely to encounter future difficulty; therefore, they could be asked to repeat the course or be offered academic assistance. Results from this study have many possible applications, but as is the case with descriptive data mining tasks, it may take some time to discover some of them. However, we feel that the course correlation networks that we generate and the various metrics that we introduce are themselves key contributions, which will lead to further research in educational data mining. This study is unique in that no other analysis of university courses is based on a notion of similarity that relies exclusively on student performance. One study, which is superficially similar, measures course similarity based on student course co-enrollments [7]. That study, also conducted by our research group and based on the same data set, uses this much more traditional notion of similarity to perform similar analyses; namely course network graphs are generated and then analyzes using existing network analysis methods and metrics.

2. DATASET DESCRIPTION

Eight years of student-course records were obtained from three of Fordham university's undergraduate colleges, where each record describes the performance of a student in a course section. This study restricts the data and analysis to: pre-health courses required for medical school admission, popular university core curriculum courses, and Computer Science and Psychology courses (a detailed analysis would not be possible if courses from all 83 majors were included). Computer Science and Psychology courses were included due to our affiliations with those departments, while core curriculum courses were chosen because of their prominence in our university and their diversity (students complete more than twenty core courses covering philosophy, history, foreign languages, performing arts, mathematics, and science). Pre-health courses are included because they cover many key introductory STEM courses. This study will be expanded to other disciplines in the future.

Table 1 summarizes the data and its distribution across the course categories. The core courses contribute more than half of the total course sections and are largely responsible for the data covering $20,797$ students. Each record corresponds to one student in one course section and includes the following features: *student ID*, *final grade*, *department name*, *course number*, *course title*, *semester*, and *section number*.

Table 1: Distribution across Course Categories

Course Category	Records	Sections	Courses
Computer Science	14,137(13%)	705(15%)	53(39%)
Psychology	18,017(17%)	966(20%)	67(50%)
Core	62,005(58%)	2,706(56%)	8(6%)
Pre-Health	13,087(12%)	434(9%)	7(5%)
Total	107,246	4,811	135

The final grade uses a 4 point scale and most courses will have many sections. Student privacy concerns prohibit us from sharing the raw data, even though the student identifier values have been anonymized; however, the course correlation matrix central to our analysis is available [6].

3. DATA PROCESSING

An overview of the process for measuring similarity between courses is provided in Section 3.1, and the individual steps are described in successive subsections. The code that implements these steps is publicly available [5].

3.1 Overview

The initial data set contains records that describe the performance of each student in a each course section. A variety of preprocessing steps are executed, as summarized in Fig. 1. A course correlation matrix that measures the similarity of each pair of courses using the Pearson correlation of student grades is generated in Step 7. The course network graphs, modularity clusters, and course cliques are then generated from this correlation matrix, as described in Section 4.

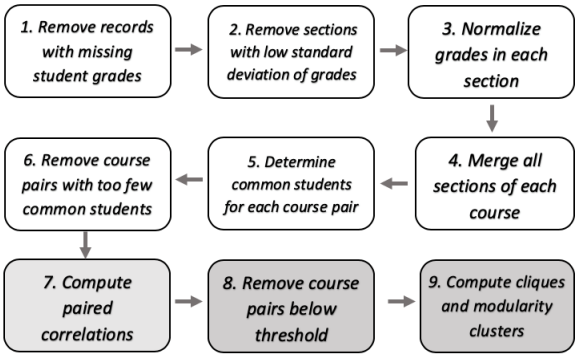

Figure 1: Overview of data processing steps

3.2 Initial Data Cleaning (Steps 1 and 2)

The first step removes records that do not have numerical grades, such as courses taken pass/fail. Some instructors sometimes assign students very similar grades, which makes it difficult to assess the similarity of courses based on grades. For this reason, Step 2 removes course sections where the standard deviation (σ) of student grades is below a specified threshold. This requires aggregation of the student course records to the section level, which yields $4,811$ sections. Fig. 2 provides the distribution of standard deviation values across these sections, and also provides a curve that shows the number of records and percentages of sections that are kept for each standard deviation threshold value (for each value we discard the sections with a lower threshold). Based on Fig. 2 we consider the values of 0.20, 0.30, and 0.40 to be reasonable candidates that maintain the majority of course sections. We ultimately selected a threshold of 0.30, which drops 6% of the sections and eliminates 6 courses (which are not left with any sections).

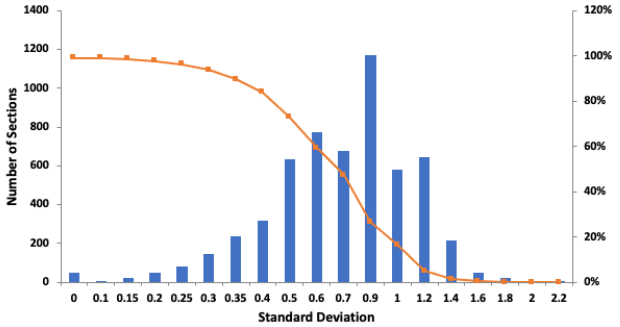

Figure 2: Distribution of grade standard deviation

3.3 Grade Normalization (Step 3)

Instructors may be easy or hard graders, and these differences will cause problems with grade correlation when a course is taught by multiple instructors. This issue is remedied by applying z-score normalization to the grades in each course section, which substracts the mean section grade from each grade and then divides it by the standard deviation of the section grades.

3.4 Generate Course-Pair Grades (Step 4-6)

Step 4 aggregates the data from the section level to the course level, which may combine dozens of course sections, spanning many years. Step 5 then forms pairs of courses, keeping on the grade data from students common to both courses. Course pairs are formed from every course that remains after application of the $\sigma = 0.3$ threshold in step 2. Step 6 then filters the course pairs that do not have at least 20 students in common, to ensure that the grade correlation is meaningful. This results in the removal of $4,585$ (25%) of the remaining course pairs.

3.5 Compute Paired Correlations (Step 7)

The final preprocessing step computes the Pearson correlation [2] between the remaining course pairs, which generates the correlation matrix that is central to our analysis. A small sample of the correlation matrix is provided in Table 2. The complete correlation matrix is publicly available [6]. Entries in the correlation matrix are not impacted by order, so values above the diagonal are omitted. Null values occur when a course pair does not have enough common students. In Table 2 we see that, as expected, there is a high correlation (0.94) between *Discrete Structures* and the associated lab. There is also a strong correlation (0.81) between *Computational Neuroscience* and *General Physics I*, which may be due to the heavy use of mathematical modeling of physical systems in both classes. *Bioinformatics* and *General Physics I* exhibit a low correlation (0.19), perhaps reflecting a heavier practical programming focus in the bioinformatics course. It is surprising that *Discrete Structures* and *Computer Algorithms* have a relatively low correlation (0.37), since they both require similar mathematical reasoning skills. This suggests that the *Discrete Structures* may not be preparing students sufficiently for future coursework.

Table 2: Representative Course-Pair Correlations

	Disc Struct	Disc Lab	Web Prog	Comp Neuro	Comp Alg	Bioinf	Gen Phys-I
Disc Struct	1						
Disc Lab	0.94	1					
Web Prog	–	–	1				
Comp Neuro	–	–	–	1			
Comp Algs	0.37	0.33	0.41	–	1		
Bioinfor	–	–	0.79	0.47	0.24	1	
Gen Phys I	–	–	–	0.81	–	0.19	1

4. RESULTS

This section describes the results derived from the course-pair correlation matrix. Section 4.1 covers the correlation results between individual course pairs, Section 4.2 covers the cliques within the course correlation graph, and Section 4.3 analyzes the course correlation network graphs.

4.1 Analysis of Course-Correlation Pairs

The distribution of Pearson course-pair correlations is displayed in Fig. 3. The leftmost bar is due to correlations between a course and itself. The top 25% of course-pair have a correlation greater than 0.5. The course network correlation graphs in Section 4.3 are generated using a threshold of 0.5.

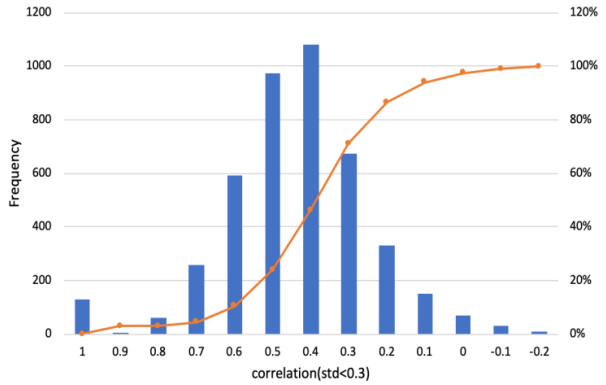

Figure 3: Distribution of course-pair correlations

Table 3 lists course pairs with correlations > 0.75. The top three entries cover matching lecture and lab courses, which is unsurprising since they cover complementary material. More than 80% of the entries are contained within an academic department, although there are interesting inter-departmental entries. The link between *General Physics I* and *Computational Neuroscience* was previously discussed and involves mathematical modeling. The link between *General Chemistry Lab II* and *Computer Algorithms* is not obvious, but both involve designing and applying a precise sequence of instructions. *Philosophy of Human Nature* shows an interesting connection with *Infant and Child Development*, potentially establishing a link between Philosophy and Psychology. The Philosophy class's link to *Scientific Computing* is more difficult to explain, although it may be related to the interdisciplinary nature of *Scientific Computing*.

4.2 Clique Results

A k-clique is a set of k nodes that are each directly connected to each other by an edge. Table 4 shows the number of cliques of each size in the course correlation network graph for correlation thresholds (ρ) of 0.55, 0.55 and 0.6. The table

Table 3: High Correlation (ρ) Course-Pairs

Course 1	Course 2	ρ
Discrete Struct II	Discrete Struct II Lab	0.96
Comp Sci II	Comp Sci II Lab	0.95
Comp Sci I	Comp Sci I Lab	0.93
Gen Phys I	Comp Neuro	0.81
Intro Bio I	Intro Bio Lab I	0.79
Web Program	Bioinformatics	0.79
Learning	Health Psychology	0.78
Perception Lab	Law and Psychology	0.78
Gen Chem Lab II	Comp Algorithms	0.78
Phil of Human Nature	Infant & Child Devel	0.78
Phil of Human Nature	Scientific Computing	0.77
Psych & Human Vals	Research Methds Lab	0.77
Law and Psych	Clinical Child Psych	0.77
Biopsych	Sens & Percep Lab	0.76
Intro Robotics	DataComm & Networks	0.76

shows that increasing the correlation threshold even slightly dramatically reduces the number of cliques, and hence we use 0.5 to retain a clear picture of course network structure. Each clique has many sub-cliques (e.g., each 7-clique has 7 6-cliques and 21 5-cliques), which we view as redundant, and hence the table excludes all sub-cliques. Cliques may span different course categories or fall entirely within one category. Table 5 shows how the cliques from Table 4 are distributed across the five course categories using $\rho = 0.5$. Cliques that do not fall within one category are included in the "Span" field.

Table 4: Number of Cliques as ρ Threshold Varies

Clique Size	$\rho \geq 0.5$	$\rho \geq 0.55$	$\rho \geq 0.6$
3-cliques	172	66	29
4-cliques	51	50	4
5-cliques	56	2	0
6-cliques	15	0	0
7-cliques	4	0	0
8-cliques	1	0	0

Table 5: Number of Cliques in Each Category

Clique Size	CS	Psych	Core	Pre-H	Span
3-cliques	46	9	0	0	117
4-cliques	11	32	0	0	8
5-cliques	14	39	0	0	3
6-cliques	0	15	0	0	0
7-cliques	0	3	0	1	0
8-cliques	0	1	0	0	0

Psychology courses form most of the large cliques with size 6 and greater. Psychology courses are more grouped together than Computer Science courses, which have many smaller-sized cliques. The 7 pre-health courses form a single clique, which suggests that performance in these courses is based on similar abilities or knowledge. Core courses lack even smaller 3 cliques. Despite their shared mission of core liberal arts training, it appears the differences in subject matter prevents similarity in course performance. No large cliques span course categories, but when $k = 3$, spanning cliques outnumber the other ones, which suggests that cliques only become meaningful at larger sizes. The largest cliques associated with the Computer Science, Psychology, and Pre-health courses are described in Table 6 of the appendix. Most of those cliques cover related courses (e.g., a 5-clique in Computer Science covers programming courses).

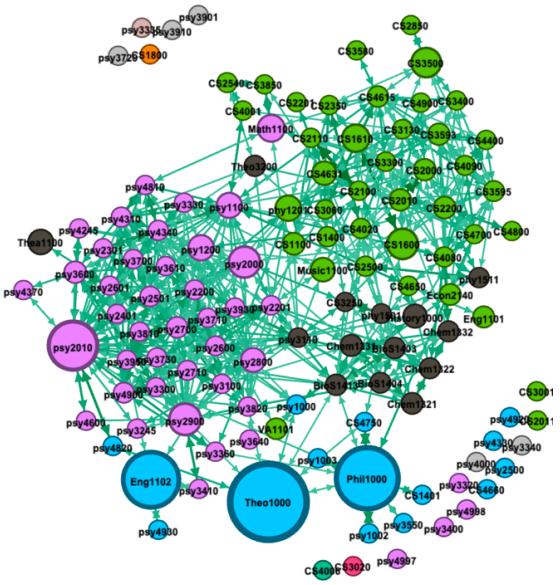

Figure 4: Network graph (all categories).

4.3 Course Correlation Network Graphs

The course correlation graphs generated with $\rho = 0.5$ were supplied to the Gephi social network analysis software [1]. Gephi partitions highly connnected nodes into modularity classes and assigns each a different color [3]. The size of each node is determined by ranking the node's "betweenness centrality," which is based on how often a node appears on shortest paths between all nodes in the network [4].

Fig. 4 shows the Gephi network that includes all courses. Nodes are labeled with a department abbreviation ("Eng" for English and "CS" for Computer Science), and 4-digit course number. Course numbers are not informative so our analysis refers to courses by title as needed. The figure shows a clear partitioning of courses between Computer Science (green, right) and Psychology (purple, left), with Pre-health courses (dark grey and below Computer Science) clustered together and forming a partial bridge between Computer Science and Psychology. While individual edges are difficult to distinguish, the figure shows that courses within a category are much better connected to each other than to courses in other categories. First-year core curriculum courses *English 1102*, *Theology 1000*, and *Philosophy 1000* are very large, indicating their large betweenness-centrality. These courses therefore often occur in the shortest paths between other courses and act as bridges between parts of the network. While these core courses do not have many connections, they connect to a diverse set of courses. *Philosophy 1000* is connected to well-connected courses from Economics, Psychology, and Computer Science, while. *Theology 1000* is connected to classes in Psychology, Pre-health (Biology), and *Philosophy 1000*. These core classes appear to be an indirect indicator of performance for classes across the university. Both classes introduce and carefully study selected core concepts in their respective fields.

Network graphs focusing on Computer Science courses and Psychology courses are provided in the appendix in Fig. 5 and Fig. 6, respectively. The modularity classes in Fig. 5 correspond to meaningful subdisciplines of Computer Science: the light-blue modularity class covers Information Science courses like Data Mining (4631); the magenta modularity class covers programming courses such as CS1 and Lab (1600, 1610), CS2 and Lab (2000, 2010), UNIX programming (3130), and Scientific Computing (4750); and the orange modularity class covers advanced courses like Algorithms (4080), Theory of Computation (4090), and Operating Systems (3595). The modularity class groupings differ from the cliques in Table 6 of the appendix, although both group the same programming courses together. Furthermore, the five largest nodes in the Fig. 5 based on betweeness centrality (*4631 Data Mining*, *4615 Data Communications*, *3593 Computer Organization*, *2200 Data Structures*, and *3300 Web Programming*) are well represented in the Computer Science cliques in Table 6. For Computer Science, high betweenness centrality reflects an abundance of both one-step and few-step connections to other courses. Within the department, it is known that a student with a poor grade in one of these classes will often struggle in the major. Most of these classes are designed to hone specialized skills within Computer Science. The key observation from the Gephi graph of Psychology courses in Fig. 6 is that the Research Methods Lab course is strongly connected with other psychology courses, while the introductory survey course is very poorly connected. This suggest that classes focused on specialized skills are more predictive of performance in advanced classes than a general survey class.

5. CONCLUSION

This descriptive data mining study defined an innovative notion of course similarity based on student performance, and then used this similarity metric to form course network graphs. These network graphs were then used to analyze the relationship between courses and course groupings. This methodology was applied to eight years of undergraduate student data at a large university.

The study established that there are many course pairs for which student performance is highly correlated. When requiring at least 20 common students, 25% of course pairs exceed the 0.5 correlation threshold used in this study, and 5% of pairs exceed 0.7 correlation. Courses with the highest correlations are often offered by the same department. In addition, multi-course clusters naturally occur, especially within subdisciplines of an academic department, such as the programming courses within Computer Science. Course clusters were identified as cliques and modularity classes within the course correlation networks. As an extreme example, all pre-health courses formed a single clique. A small number of courses with high betweeness centrality were shown to link a diverse set of topics—within one discipline or between disciplines, and those courses connecting discplines were much more likely to introduce specific skills than to provide a broad survey of an area.

This paper also introduced a methodology for generating a course grade correlation matrix from student data, and included several steps to address confounding factors such as differing instructor grading policies. This methodology is available to other education researchers through our software and associated documentation [5]. Our work presented a new way of looking at course relationships by a novel way of measuring similarity. We plan to continue to investigate this notion of course similarity and to apply it to a larger set of courses.

6. REFERENCES

[1] M. Bastian, S. Heymann, and M. Jacomy. Gephi: an open source software for exploring and manipulating networks. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 3, 2009.

[2] J. Benesty, J. Chen, Y. Huang, and I. Cohen. Pearson correlation coefficient. In *Noise reduction in speech processing*, pages 1–4. Springer, 2009.

[3] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[4] U. Brandes. A faster algorithm for betweenness centrality. *Journal of mathematical sociology*, 25(2):163–177, 2001.

[5] M. Riad-Zaky, G. M. Weiss, and D. D. Leeds. *Course Grade Analytics with Networks (CGAN) [computer software]*, April 2021. https://www.cis.fordham.edu/edmlab/software.

[6] G. M. Weiss and D. D. Leeds. *Fordham University Course Correlation Matrix [data file]*, January 2021. https://www.cis.fordham.edu/edmlab/datasets.

[7] G. M. Weiss, N. Nguyen, K. Dominguez, and D. D. Leeds. Identifying hubs in undergraduate course networks based on scaled co-enrollments. In *Proceedings of the 14th International Conference on Educational Data Mining (EDM 2021)*, 2021.

APPENDIX

Table 6 lists the large cliques associated with Computer Science, Psychology, and Pre-health courses. Many of the cliques have a common theme. Computer Science's second 5-clique includes three internet-focused courses: *Web Programming*, *Client Server Computing*, and *Data Communications*, while the third clique is dominated by programming courses (*Operating Systems* is an exception but includes programming projects). Psychology's 7-clique links classes covering complementary and overlapping elements of cognition; however, the 8-clique appears to span diverse topics. As mentioned earlier, the pre-health clique covers core science courses required by medical schools.

Table 6: Large Cliques in Different Categories

COMPUTER SCIENCE		
5-Clique	**5-Clique**	**5-Clique**
Data Mining	Data Mining	Comp Sci II
Web Programming	Web Programming	Comp Sci II Lab
Data Struct.	Data Comm.	Data Struct.
Client-server Comp	Client-server Comp	Operating Systems
Comp. Org.	Comp. Org.	Scientific Comput.
PSYCHOLOGY		
8-Clique		
Child Develop.	Biopsy.	Research Methods
Learning	Social Psych Lab	Human Sexuality
Aging and Society	Law and Psych	
7-Clique		
Child Develop.	Personality	Abnormal Psych
Intro Clin. Psych	Found. of Psych	Social Psych
Cognitive Psych		
PRE-HEALTH		
7-Clique		
Intro Bio I	Intro Bio II	Intro Bio Lab I
Gen Chem I	Gen Chem II	Gen Chem Lab I
Gen Chem Lab II		

The Gephi course correlation network graph for the Computer Science is displayed in Fig. 5. The contents of Fig. 5 were describe in detail in Section 4.3 and highlighted how the different modularity classes correspond to different subdisciplines within computer science. The Gephi course correlation network graph for Psychology, which was only briefly described in Section 4.3, is displayed in Fig. 6. Meaningful subcategories are much harder to identify, but it is notable that Research Methods Lab (2010) is most strongly connected with other psychology courses, indicating a valuable skill shared across the category. This contrasts with the required introductory survey class, Psychology 1200, which has a much lower betweenness centrality. This indicates that a class focused on specialized skills is more predictive of performance in more advanced classes than a general overview class. The psychology courses with largest betweenness centrality are all represented in the cliques in Table 6. The top four courses based on betweenness centrality are: *2010 Research Methods*, *2900 Abnormal Psychology*, *2800 Personality*, and *2700 Child Development* – all courses with specialized foci. As in Computer Science, high betweenness centrality in Psychology reflects an abundance of both one-step and few-step connections to other courses.

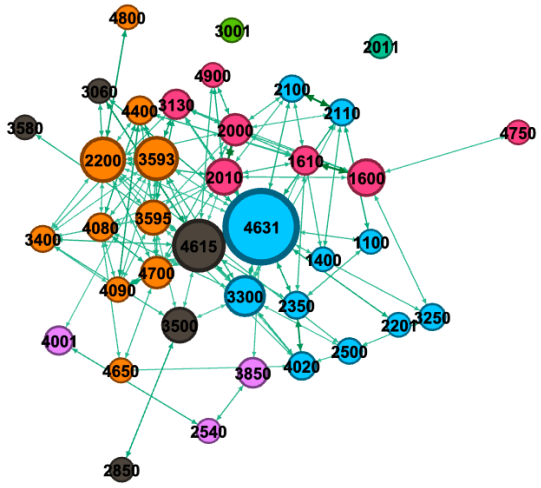

Figure 5: Computer Science network graph.

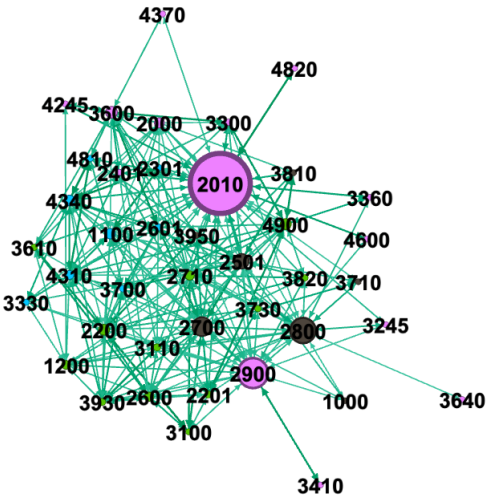

Figure 6: Psychology network graph.