Data Analysis Project

Eriq Augustine
Matt Tognetti
Aldrin Montana
Chris Gilson

CSC_Student Dataset

This dataset features enrollment information for 314 computer science students for the Fall 2010 quarter. The students were asked to provide all CSC/CPE courses that they were currently enrolled in. In addition to enrollment information, the amount of time that each user spent on CSL (14-235) machines or CSC servers [vogon, falcon, multicore, hornet, unix1, unix2, unix3, and unix4] from October 14 to November 27. Time information was only seen for 179 of the 314 students.

Choice of servers

The machines/servers were chosen to capture students who were working outside of class. Because of that, closed labs (301 - 303, OS Lab, SE Lab, etc) were avoided. The large servers were also chosen to hopefully capture students that were working outside of the labs, but on CSC material.

Deficiencies in dataset

There are several deficiencies in the nature of this dataset that must be addressed:

1. A person logged onto a school server is not necessarily doing useful school work. There is not much that can be done about this. Even of we were to check what processes they were running, there is no guarantee that they are not away from the computer.
2. It is unknown what course that student is spending his/her time on.
    - All we can collect is the time that they are spending in total. Therefore a class that takes only a short amount of time outside of class can appear to be a more demanding course if it is taken at the same time as a demanding course. This generates two situations that must be explored:
    - A student is taking a demanding course, A, and another course, B, that is often taken at the same time as A. This situation arises very often because of the suggested courses given to all students (the flowchart). Because these courses are highly coupled, it is not incorrect to say that a student who is taking B will spend a lot of time working outside of class. Therefore, this situation is not hurtful.
    - A student is taking a demanding course, A, and another course, B, by pure coincidence and these courses are not often taken together. In this case A does artificiality inflate the effect of B. However, this situation can be avoided with a fair choice in support.

4. Many students prefer not to work in school machines. Because we do not have the ability/right to check every every machine that a student can be working on, this will just have to remain as an unresolved issue.

Questions

For this dataset, we wanted to answer three questions:

1. Given a class or set of classes, what is the average time that students taking these classes spend on work outside of class?
2. Given a class or set of classes, how much time will I have to spend outside of class?
3. Which classes take the most time outside of class?

Methods

The inspiration for the analytical methods used for this dataset is Apriori market basket analysis. However because of the nature of the data and the questions being asked, there are some drastic modifications that must be made.

- R = A rule. A rule is a collection of classes/courses.
- MinTime - The minimum time to consider.
- Time(t_i) - The time the student, t_i, spent on CSC machines.
- T = {t_1, t_2, …, t_n}. The collection of “market baskets” (sets of classes/courses that a student is concurrently enrolled in).
- Support(R) - “What percent of students also have this set of classes?”

  |{ti T | R ti| |n
- Confidence(R) - “Of the students that have this set of classes, what percent are spending at least the given threshold of time outside of class?”

  | {tj T | R tj & Time(tj) >= MinTime } || {t_i T | R ti } |
- Frequent Item Sets - A set of classes/courses meeting the minimum support.

Beside the stated modifications, the core algorithm for both apriori and rule generation remain the same.

Results

For both questions one and two, we will be using a course set that we know (from personal experience) are often taken together: CPE-308 and CPE-430.

Question 1

Students taking both CPE-308 and CPE-430 spend an average of 24.68 hours a week on work outside of class.

Question 2

A small, unofficial survey of CSC students revealed that 16 hours or more a week constitutes a “demanding” workload. Given both the assumed schedule (CPE-308 and CPE-430) and the minimum time of 16 hours a week we can predict with 3.4% support and 33% confidence that a student will spend at least 16 hours a week on work outside of class.

Question 3

For this question, we enforce a low support (1%) and not enforce a confidence to drag out any class (set) that takes a lot of time. Here are the top five schedules:

Rank	Courses	Average Time Per Week	Number of Students Seen	Support
1	CPE-471, CPE-308, CPE-305	109.851851851852	2	0.011173
2	CPE-300, CPE-349	108.702268518519	2	0.011173
3	CPE-464, CPE-308, CPE-305	108.554189814815	2	0.011173
4	CPE-464, CPE-308	108.554189814815	2	0.011173
5	CPE-464, CPE-305	108.554189814815	2	0.011173

Skylined:

Rank	Courses	Average Time Per Week	Number of Students Seen	Support
1	CPE-471, CPE-308, CPE-305	109.851851851852	2	0.011173
2	CPE-300, CPE-349	108.702268518519	2	0.011173
3	CPE-464, CPE-308, CPE-305	108.554189814815	2	0.011173
4	CPE-480, CPE-520	69.547337962963	2	0.011173

5	CPE-445, CPE-466	58.6690893518519	3	0.016760

Conclusions

Unsurprisingly, CPE-305 with Dr. Staley (the only 305 professor this quarter) appears to be the most time consuming taking both the first and third spot (in the skylined ranks). 308 also appears in the same sets as 305, however further analysis shows that the average time for 305 students is 48 hr/week while the average time for 308 students is only 31 hr/week.

CSC_Professor Dataset

This dataset consists of the personal preferences and basic biographical information relating to CSC topics of a few hundred CSC/CPE students. The students were asked to fill out a short survey asking for information about their major, time at cal poly, and experience and attitude towards programming. In addition, we asked the students to list their 3 favorite and 3 least favorite CSC/CPE courses and the professors that taught them.

Choice of Survey Questions

The questions were designed with two main objectives in mind. First, to gather basic information on the student, such as age, quarters at cal poly, and major. Second, to gain an understanding of the student's personality, learning, and programming style. This was accomplished by asking about preference in working with a group, programming as a hobby, and favorite programming language.

Deficiencies in Dataset

Impossibility of gaining complete understanding of a student and his/her preferences from a short survey. Not all courses/professors covered. Perhaps student's liked the professor but hated the material or vice versa

Questions

Given a student with certain preferences (based off the survey), which professor would he/she enjoy taking a course with the most?

1. Which professor is the most favored by students?
2. Which professor is the least favored?

Methods

Clustering was used to group the students by similarity based on their answer's to our biographical survey. Distance between points was computed using a special survey distance function that is handles non-numeric answers. For example, two students who both like linux will be 0 distance from each other for that question, if however the two students preferred different operating systems they would be 1 distance from each other, regardless of their choices. Once the students were clustered (using kmeans clustering) it became possible to introduce a new student, find the cluster he/she would belong too, and use their ratings of professors and courses to predict which professors the new student would like. I found, through trial and error, that 6 clusters struck just the right balance between cluster integrity and cluster size.

Results

Question 1

There is no one answer to this question, as it is entirely dependant upon the survey answers of the student and the desired course. On the success rate hovered around 60%, but I believe this has less to do with the methodology and more to do with the the dataset. In general, the answers were predictable based on general knowledge of student opinion, however, results did seem to suffer from the small size of the dataset, mostly because many courses received few or no ratings and because the survey questions may not have provided enough insight into the student's personalities.

Question 2

The rankings from most liked to least liked are:

Rank	Username	Relative Score	Rank	Username	Relative Score
1	zwood	50	22	jyoliver	3
2	cstaley	41	23	rsandige	3
3	jworkman	30	24	wpilking	3
4	mhaungs	24	25	mealy	2
5	pnico	22	26	kmammen	2
6	fkurfess	21	27	slivovsky	1
7	djanzen	18	28	cpokorny	1
8	cmclark	16	29	webb	1
9	dekhtyar	14	30	clements	1
10	csturner	14	31	tbell	-1

11	bucalew	10	32	ebuckalew	-1
12	akeen	9	33	lmyers	-1
13	gfisher	8	34	rgduncan	-1
14	hghariby	6	35	jharris	-1
15	jseng	6	36	lbrady	-1
16	clupo	6	37	jconnely	-3
17	husmith	6	38	phatalsk	-6
18	ivakalis	6	39	jgrimes	-9
19	jgerfen	5	40	nparham	-16
20	bellardo	4	41	jdalbey	-26
21	mliu	3	42	kogorman	-63

Conclusion

While it is possible to draw useful answers from this dataset, it would ultimately be more useful if we had more ratings and more factors to use in computing student similarity. It is however, very capable of answering questions relating to which professors are currently in favor, and which are not. Based off the results we've gathered from this relatively small dataset, I believe if the dataset we're to be expanded to include not only more students, but also survey questions that provide more insight into individual personalities, it would be capable of making very accurate recommendations.

HON Dataset:

This dataset features match statistics for the video game Heroes of Newerth. Heroes of Newerth is a video game consisting of two teams of 5 players that each control a 'hero' unit. There are AI-controlled units, automatically spawned at specific intervals, that are referred to as 'creep'. The specific statistics contained in this dataset are hero kills and deaths, creep kills and deaths, team hero kills and deaths, team creep kills and deaths, and gold and experience for each type of kill. There is also a statistic called 'bloodlust' which refers to the hero getting the first kill in the game.

Terms

- APM - Actions Per Minute
- Lane - Three main pathways that connect each team's base
- Creep - AI-controlled units spawned for each team that travel along a lane to attack the other team
- Deny - Killing an allied unit or structure to prevent the other team from getting the kill and subsequently, the experience and/or gold.
- Spamming - Doing a repeated or similar action in quick succession. The action performed is often useless. Ex: clicking the same spot many times when once would suffice.
- Lineup - A set of heroes on the same team.
- Escape Mechanisms - A way for a hero to quickly get away from an enemy. These include the ability of teleport or go invisible.
- Carry - A hero whose role is just to do damage. These heroes are often defenseless and weak early in the game, but grow very strong near the end.

Choice of Player and Team Statistics

The statistics that will be used for analysis will be creep kills and deaths, hero kills and deaths, and gold obtained.

Questions

Question 1

In a single match, does the overall APM for a team predict the winner of the match?

Justification

APM has been a debated thing in the HoN community (http://forums.heroesofnewerth.com/showthread.php?t=181307). Here is an overview of the arguments:

- ***APM is Important***: The more that you are doing, the more control that you have and the more precise you can be. There is always something that you can be doing instead of resting.
- ***APM is not important:*** You only need to make sure that you perform certain actions at certain times. There is a limit to how many useful activities you can do, anything else is just spamming.
- The idea behind this question is that although APM does not make a good player, good players tend to have a high APM. Using this idea, it would make sense to attempt to use APM as a predictor.
-
- **Methods:**

For this question, a simple query can provide the results. Just taking the number of teams that won and had a higher APM than their opponents over the total number of matches. The APM that will be used is the Team APM which is just the average of all of that team's players' APM.

Results

In 57847 matches from versions 1.0.18 and 1.0.19 (two very similar versions) 37010 matches ended with the victor having a higher APM than the loser. This means that about 64% of the time, the team with higher APM wins.

Conclusion

Although not decisive, teams that have a higher APM tend to win (64% of the time). Therefore, APM may be used as a predictor for the victor of a match.

Question 2

For any single hero, does the hero's average APM predict the winner of the match?

Justification

To extend our first question, we want to see if the APM of a single hero is also a predictor of match outcome. Some possibilities include:

1. The actions of a single hero can carry the team and compensate for other heroes that are being played poorly.
2. The actions of a single hero are not enough to compensate for other heroes that are being played poorly.

Methods

As for question 1, this question may be answered with a database query. We will take each match with a given hero (that is a 5 versus 5 match) and take the average APM for all matches won, and the average APM for all matches lost.

Result

Highest average winning APM is 125.70. Lowest average winning APM is 72.89. Average average winning APM is 101.93.

Highest average losing APM is 113.23. Lowest average winning APM is 69.56. Average average losing APM is 93.66.

Conclusion

The results for this analysis is consistent with question 1. If a hero's APM is higher, then it is expected that a team's APM will be higher. Because the average APM for a hero losing a match is not lower than the lowest average winning APM for a hero it seems that APM is not a consistent predictor or the sole predictor of what team will win the match. However, the average APM for a hero being on the winning team of a match is always higher than the average APM for the same hero being on the losing team of a match. This shows that the APM for a hero can be used as an indicator, though perhaps not a predictor.

Question 3

Given a particular hero, how much APM is best for the hero to be played to the best of its ability?

Justification

With the idea that some heroes require a higher level of control than other heroes, we want to know how much control is necessary to play a hero to the best of its ability. As mentioned before, more APM, arguably, means more control. We will use APM as an indicator of how much control is necessary to play a hero as best as possible.

Methods

For this question, all records for a given hero will be retrieved. These records will be clustered based on hero kills, hero deaths, creep kills, creep denies, experience gained, and gold gained. Then the average APM for the cluster with the best record will be considered the optimal APM for the hero. The best record is calculated by awarding a point to a record which has the highest of the following statistics:

- Hero kills - deaths
- Creep kills
- Denies
- Total experience gained
- Total gold gained (hero kill gold, creep kill gold, neutral creep kill gold)

Results

The average APM for every hero’s optimal APM is 100.79.

The highest APM of every hero’s optimal APM is 132.41. Heroes with this optimal APM are SoulReaper and Succubus.

The lowest APM of every hero’s optimal APM is 68.47. Heroes with this optimal APM are Ophelia and Pollywag Priest.

Conclusion

SoulReaper and Succubus seem to require the highest level of control. This is probably due to having a lot of targetable abilities and having to stay in lanes often. Pollywag Priest and Ophelia seem to require the lowest level of control, possibly because they have more passive abilities or tend to be more support and so are not the main fighters in a lineup.

Its important to note that since these APM values are calculated from clusters determined by several stats, these APMs aren’t optimal APM’s just for hero killing or gaining experience, but some balance between all of the stats used in the clustering process.

After seeing the wide range in optimal APM values, it seems that the required level of control for a hero can change a lot, meaning the heroes must be significantly different. This makes sense because there are many roles that heroes are expected to fill and some roles may require more control than others. Also, some heroes have many passive abilities while others do not, so this is also expected to have a large influence on optimal APM for a hero.

Question 4

For each even matchup (1v1, 2v2, … 5v5), what is the most successful lineup?

Justification

In HoN there are currently 72 differnet heroes. Every hero has differnet skills, abilities, and play styles. On thing the competitive player must do is find a team of heroes that work well *together*. It is usually more important for a team to do well than any single hero. Therefore, knowing the most successful lineups is something that most HoN players would like to know.

Methods

For this question pagerank is used. Five different graphs are made, one for each team size. Every seen lineup is an actor. Winning lineups get a directed edge from the losing lineup to the winning lineup. Then a standard pagerank algorithm is invoked.

For this question a subset of the database only containing matches from versions 1.0.17, 10.0.18 and 1.0.19 (all very similar versions) was used. In total, 504490 matches were examined.

Results

To conserve space, only the top five are included in this paper. The full results are published on the wiki.

1v1:

Place	Lineup	Pagerank
1	Scout	0.0619690902147692
2	Predator	0.0374704616688932
3	Zephyr	0.0361719310161104

4	Pestilence	0.0339647205059268
5	MageBane	0.0336453945511611

2v2:

Place	Lineup	Pagerank
1	Pestilence-Deadwood	0.00415485173753319
2	BloodHunter-Predator	0.00345204990284772
3	Pebbles-Deadwood	0.0031467743648104
4	Scout-Pestilence	0.00306461166197475
5	Predator-Pestilence	0.0030134809151569

3v3:

Place	Lineup	Pagerank
1	SwiftBlade-Pestilence-WretchedHag	0.000518236116406151
2	SwiftBlade-Valkerie-WretchedHag	0.000425022239166349
3	Electrician-Maliken-Bubbles	0.000364976490024145
4	Pebbles-Ophelia-WitchSlayer	0.000363832761469056
5	Legionnaire-Jeraziah-Andromeda	0.000312364976490024

4v4:

Place	Lineup	Pagerank

1	Scout-WretchedHag-SandWraith-FlintBeastwood	0.000179043743641913
2	Pebbles-Predator-WretchedHag-WitchSlayer	0.000142421159715158
3	Valkerie-Pandamonium-WitchSlayer-Bubbles	0.000142421159715158
4	SwiftBlade-Pandamonium-Gauntlet-FlintBeastwood	0.000142421159715158
5	Devourer-WitchSlayer-ForsakenArcher-Engineer	0.000142421159715158

5v5:

Place	Lineup	Pagerank
1	BlackSmith-NightHound-Accursed-PuppetMaster-Gauntlet	1.09565026843432e-05
2	Zephyr-Jeraziah-Valkerie-Vindicator-FlintBeastwood	1.09565026843432e-05
3	Armadon-Armadon-Armadon-Armadon-Armadon	7.99824695957051e-06
4	Thunderbringer-Thunderbringer-Thunderbringer-Thunderbringer-Thunderbringer	5.85351155911033e-06
5	Thunderbringer-Thunderbringer-Thunderbringer-Thunderbringer-Pyromancer	5.55768598663307e-06

Conclusion

For matches with 3 or less people per team, the heroes that did the best were heroes that could either heal themselves or heroes with built in escape mechanisms. This makes sense that heroes with the ability to survive without the need for support from allies do well in small games.

For matches with 4 people per team, the lineups that do well are lineups that have heroes that have complementing roles. There are know roles that players try to fill to best support their team. These include carry and support. The successful teams are teams that have heroes that are know to be specifically good in a specific role.

For matces with 5 people per team, we see the top two teams are teams that, like 4v4, has a balenced lineup with heroes that complement each other. However, for the 3rd, 4th, and 5th places we see teams that are mainly composed of a single hero. We see this because these are heroes that have abilities that do really well when done as many times as possible. When they have multiple players playing the same hero they are able to use thses abilities so often that the other team has a hard time winning.

Real Bakery Dataset

This dataset is an real-world version of Professor Dekhtyar's BAKERY dataset. The data was collected manually at SLO Donut Company, at all hours, over the course of approximately 2 weeks. Each customer's order is recorded, along with their gender and age, and how long they spent in the shop after making their purchase.

Deficiencies in Dataset

Because the data had to be gathered manually, the dataset's size is relatively small. Additionally, some of the customer data, including age, is subjective and may or may not be accurate.

Questions

For this dataset, we wanted to answer 2 questions:

1. What items are frequently bought together?
2. Does the number of each item purchased drastically effect what else a customer buys?

Methods

The analytical methods used is based on the Apriori algorithm of association rule mining. Rule Mining will be done using an unmodified Apriori Algorithm. Receipts will be mined twice, once accounting for the number of each different item purchased, one only considering which items were purchased.

Results

(Note, many different combinations of support and confidence were tried for both methods, only reporting the most helpful result sets in order to safe space)

Including Count
using support = .027, confidence = 0.6
{CF} --> {FC}
{CF} --> {GLAZED}
{SUGAR} --> {CCF}
{CF} --> {MB}
{PF} --> {MB}
{GLAZED, MB} --> {FC}
{GLAZED, CF} --> {FC}
{FC, MB} --> {CF}
{MB, CF} --> {GLAZED}
{CCSQ, CR} --> {AF}
{GLAZED, MB, CF} --> {FC}

Not Including Count
using support = .015, confidence = 0.08;
{MAPLE} --> {COFFEE}
{CRUMB} --> {COFFEE}
{CHOCO} --> {COFFEE}
{CB} --> {COFFEE}
{BACON} --> {COFFEE}
{BEAR} --> {COFFEE}
{CCSQ} --> {MILK}
{DH} --> {NCS}
{GLAZED} --> {MAPLE}
{CHOCO} --> {GLAZED}
{AF} --> {CCSQ}

Where:
CF = Chocolate Filled ;
GLAZED = Glazed;
MB = Maple Bar;
CCSQ = Chocolate Chip Square;
DH = Donut Holes;
MILK = milk;
BACON = Bacon Donut;
CRUMB = Crumb Donut;
CB = Chocolate Bar;

FC = French Chocolate;
CCF = Cream Cheese Filled;
PF = Powered and Filled;
AF = Apple Fritter;
NCS = Normal Cake w/ Sprinkles;
BEAR = Bear Claw;
MAPLE = Maple Donut;
CHOCO = Chocolate Donut;

Conclusions

Question 2

Immediately we notice that there is a significant difference in results between the counted and uncounted datasets. We see that the counted dataset produces rules with more items in them, possible because of the larger size of the market baskets. Additionally, we see that there are no beverages present in the counted association rules. This could possibly be because even if people buy multiple donuts, people are likely to only buy one beverage to go with those donuts. We also notice that there are many more filled donuts present in the counted section. A possible explanation for this is that someone buying many donuts, such as for an office party, is more likely to buy a filled donut because it is a more substantial food item than a regular donut.

Question 1

From the counted set, we really only glean one significant rule: One is that Glazed Donuts, Maple Bars, Chocolate Filled, and French Chocolate donuts tend to be bought together. It is possible this is because these are all larger donuts, and people are buying these donuts for an office or group setting.

From the uncounted set, we gather a number of significant rules that are related. We see that many donuts are bought alongside a beverage, mainly coffee. The reason for this obviously being that donuts tend to be a meal that people eat as a “light morning pickup”, and coffee is the beverage of choice to go along with those meals because of it’s warmth and caffeine content.

General

The unfortunate shortcoming of this dataset is it’s size. In total, the dataset is only 606 entries long, because they had to be gathered by hand. For a dataset of this nature (i.e. a dataset representing customers and the choices they make) with so many different choices (There are 86 different items available for purchase), it is difficult to glean any rules with high confidence. This is reflected in the minimum confidence and minimum support for both datasets. With a dataset of much larger size (10,000 or more entries), I believe this method of analysis would yield much more useful results.