CS 412: Homework #1

Due on Friday Sept. 30th

Kurt Rudolph
rudolph9

Problem 1

As a highly application-driven discipline, data mining has been widely applied in many areas. We briey presented two highly successful and popular application examples of data mining: business intelligence and Web search engines, in our textbooks. Do you think that data mining can also be applied to the following areas? If yes, please provide a brief yet concrete example, if not, please briefly state your reasons.

(1) Software Engineering

Solution

Software Engineering often requires months and even years of development time to develop a single application. Generally the projects involve a multitude of individuals and identifying where a the time of the development teams is being spent is an important when estimating completion dates and tracking completion of the various components. Data mining can be applied to the discipline of software engineering by tracking development trends. A real world example of this is the Atlassian Packages (Confluence being the wiki package currently utilized by the CS department here at UIUC).

(2) Transportation

Solution

The area of transportation can apply data mining to information collected on the various vehicles they use to provide service. Trends can be identified to busses being late or early, the times of the day in which a train arrives late. The information could be used to counteract the problems and provide better service to patrons.

(3) Sociology

Solution

Data mining may be applied to the area of sociology by identifying how persons interact with on another. A good example is Facebook, what people like, who people are friends with, when people post, etc. are all tracked. Methods of data mining can be applied to all information collected and identify various tends in human behavior.

Problem 2

Suppose a student collected the price and weight of 20 products in a shop with the following result

price	$11.78	$85.12	$10.47	$298.00	$38.45	$102.14	$123.62	$203.29	$65.00	$225.50
weight	3.2	3.4	4.5	35.4	9.1	5.7	1.5	23.8	8.6	42.3

price	$9.25	$164.32	$102.45	$120.45	$73.15	$625.00	$125.00	$242.64	$441.76	$325.45
weight	5.9	12.3	6.5	11.8	12.2	32.9	11.6	48.0	52.9	78.2

(1) Calculate the mean, Q1, median, Q3, and standard deviation of price and weight

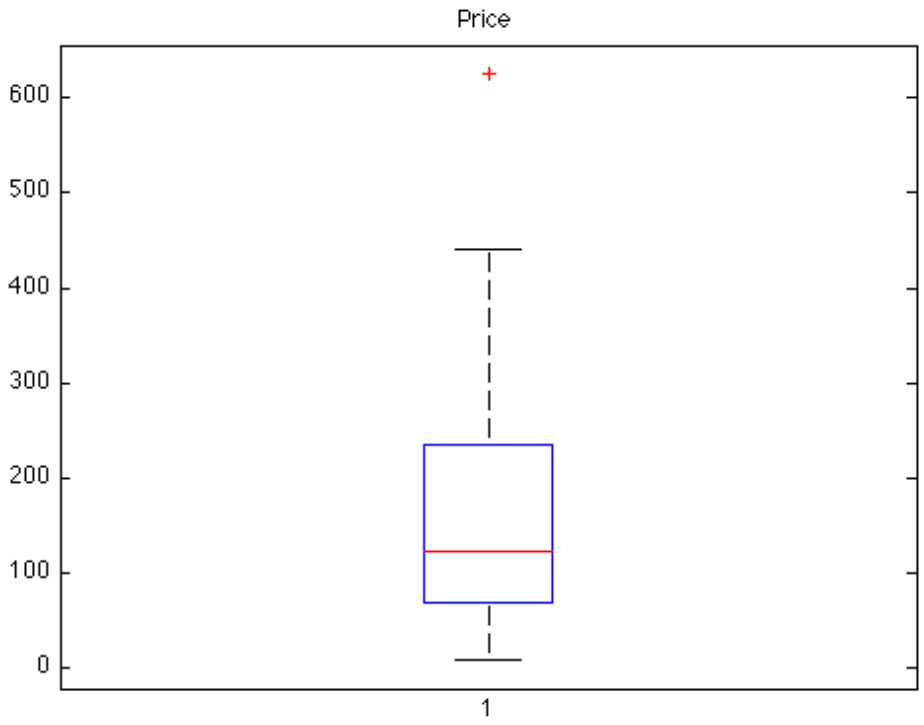
Problem 2 continued on next page...

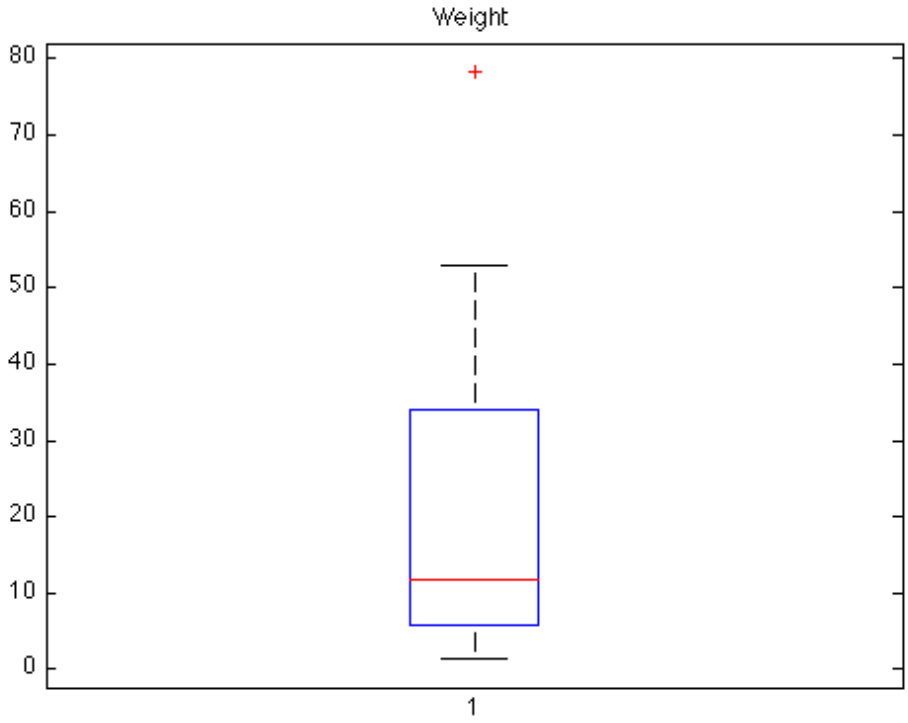
Solution

priceMean = 169.6420
priceMedian = 122.0350
priceStd = 156.9378
priceQ1 = 69.0750
priceQ2 = 234.0700
weightMean = 20.4900
weightMedian = 11.7000
weightStd = 21.0222
weightQ1 = 5.8000
weightQ2 = 34.1500

(2) Draw the boxplots for price and weight

Solution

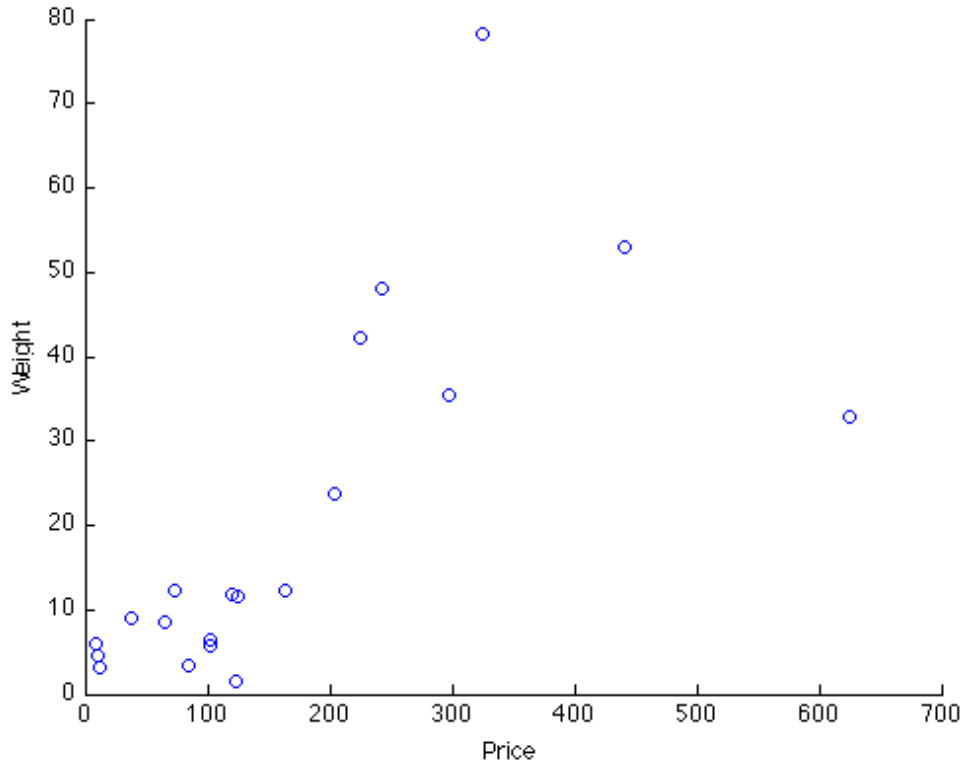


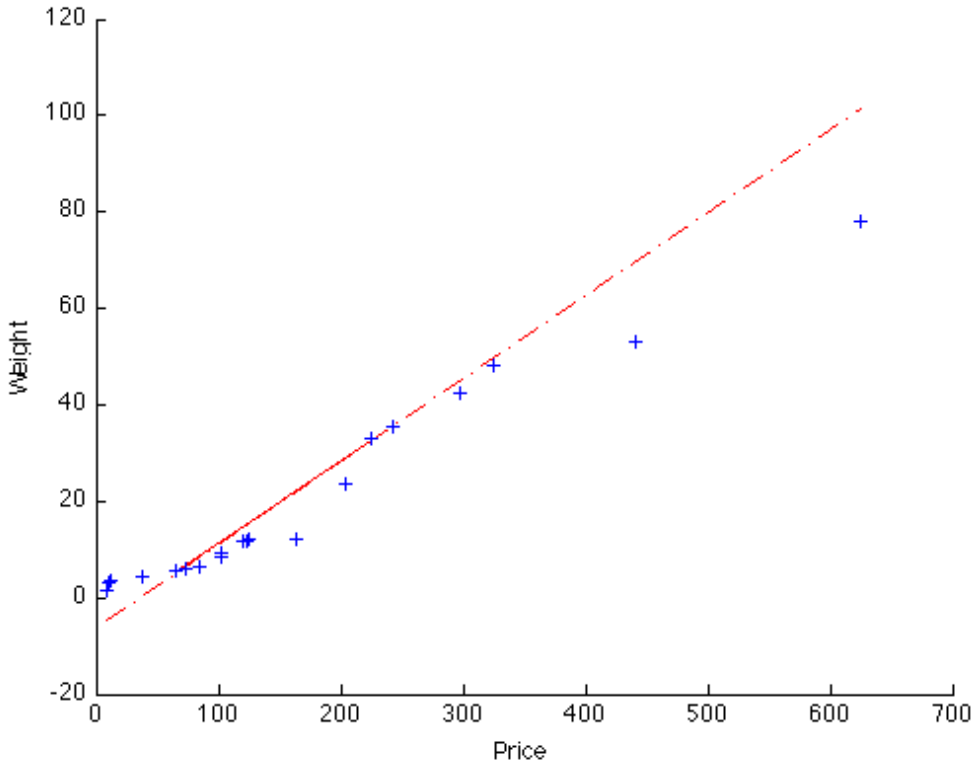


(3) Draw scatter plot and Q-Q plot based on these two variables

Problem 2 continued on next page...

Solution


Weight
Price
0
10
20
30
40
50
60
70
80
0
100
200
300
400
500
600
700


Weight
Price
-20
0
20
40
60
80
100
120
0
100
200
300
400
500
600
700

(4) Normalize the two variables based on the min-max normalization ($min = 1, max = 10$)

Solution

(5) Normalize the two variables based on the z-score normalization

Solution

(6) Calculate the Pearson correlation coefficient. Are these two variables positively or negatively correlated?

Solution

(7) Take the price of the above 20 products, partition them into four bins by each of the following methods

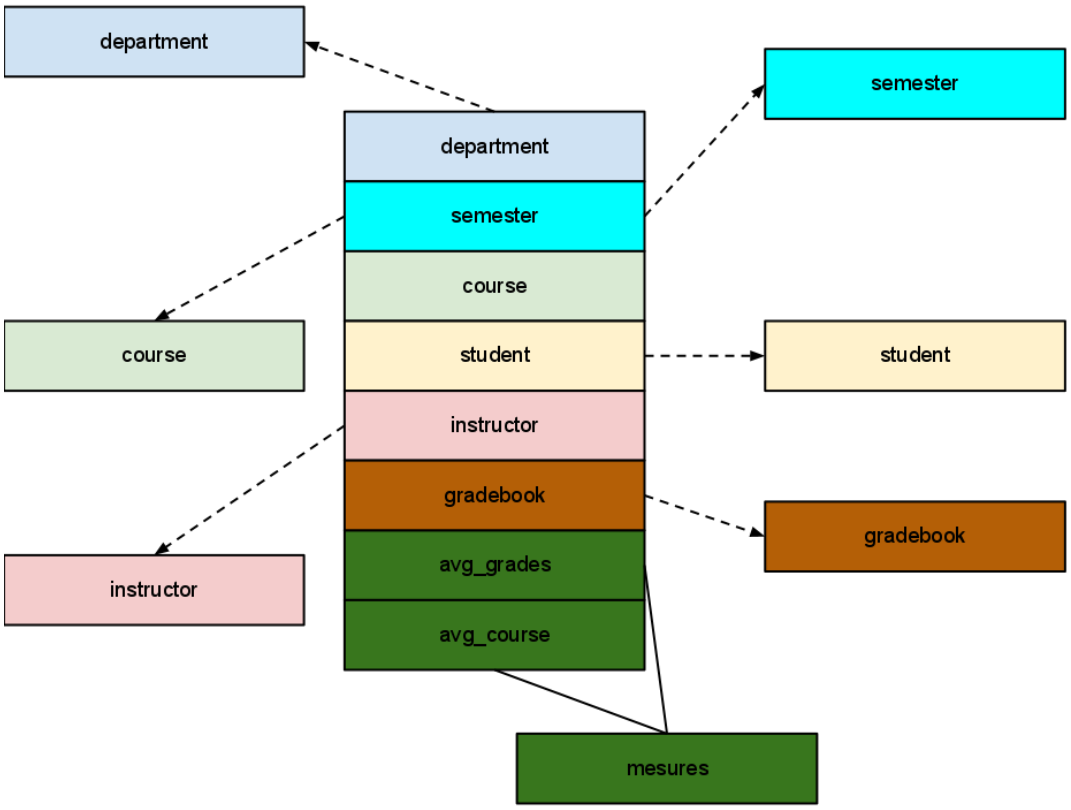
- equal-width partitioning
  Width = Max-Min/4 = 153.9375
  Bin1: 9.2500, 10.4700, 11.7800, 38.4500, 65.0000, 73.1500, 85.1200, 102.1400, 102.4500, 120.4500, 123.6200, 125.0000
  Bin2: 164.3200, 203.2900, 225.5000, 242.6400, 298.0000
  Bin3: 325.4500, 441.7600
  Bin4: 625.0000
- equal-depth (equal-frequency) partitioning Bin1: 9.2500, 10.4700, 11.7800, 38.4500, 65.0000
  Bin2: 73.1500, 85.1200, 102.1400, 102.4500, 120.4500
  Bin3: 123.6200, 125.0000, 164.3200, 203.2900, 225.5000
  Bin4: 242.6400, 298.0000, 325.4500, 441.7600, 625.0000

Problem 3

Design a data warehouse for a university's gradebook data to analyze the class performances. Suppose the data warehouse consisting of the following dimensions: department, semester, course, student, instructor, and gradebook; and a set of measures you would like to define.

(a) Draw a star-schema, based on your consideration of power and convenience of analysis of the warehouse

Solution



(b) Suppose we want to present the standard-deviation of final scores by course and year, and freely drilling up and down in multidimensional space, describe how this measure can be computed efficiently

Solution

We can construct our multidimensional as a function of semester, student, gradebook. Doing so we are able to feel drill up and down the space and gather the information year by year (each set of fall summer and spring semesters).

(c) Is top 10% in a class a holistic or algebraic measure? Discuss how to develop an efficient (maybe approximate) methods to compute a query like: find those Engineering students whose final score is within top 10% in class in at least 80% of the CS courses that he or she has taken?

Solution

This is a holistic measure. An efficient method to compute such a query is to conduct preprocessing to enable efficient execution of such query, measure such as equal depth binning enables a query such at top 10% to be 80% percent of the CS courses that this individual has taken to be excited with significantly quicker table lookups. The data is not longer sparse but ordered. The execution of the query may follow picking the top 80% of each students classes followed by picking the top 10% of those individuals, the buckets make the original execution of the query rather quick and while the ranking of the students for the top 80% of the classes differs from 100% of their classes, have the students sorted by buckets in this manor makes the 80% query is simi-sorted rank and may drastically reduce the running time.

(d) Is it a good idea to merge this data warehouse and the current university's gradebook database system together into one big data management/analysis system? Why?

Solution

Like with any decision their are pros and cons to one large data base. The current university grade book database system works! By merging all the data into one where house you run the risk of breaking a perfectly systems. While at the same having two locations where the data stored for current purposes and another location where the data is stored for analytical purposes, we run into synchronization issues. A single management system is the idea implementation for maximum efficiency, having the ability to perform analysis during non peak user hours makes best of the hardware and eliminates unnecessary redundant data but merging an existing working system into a new system always has it's down side.

Problem 4

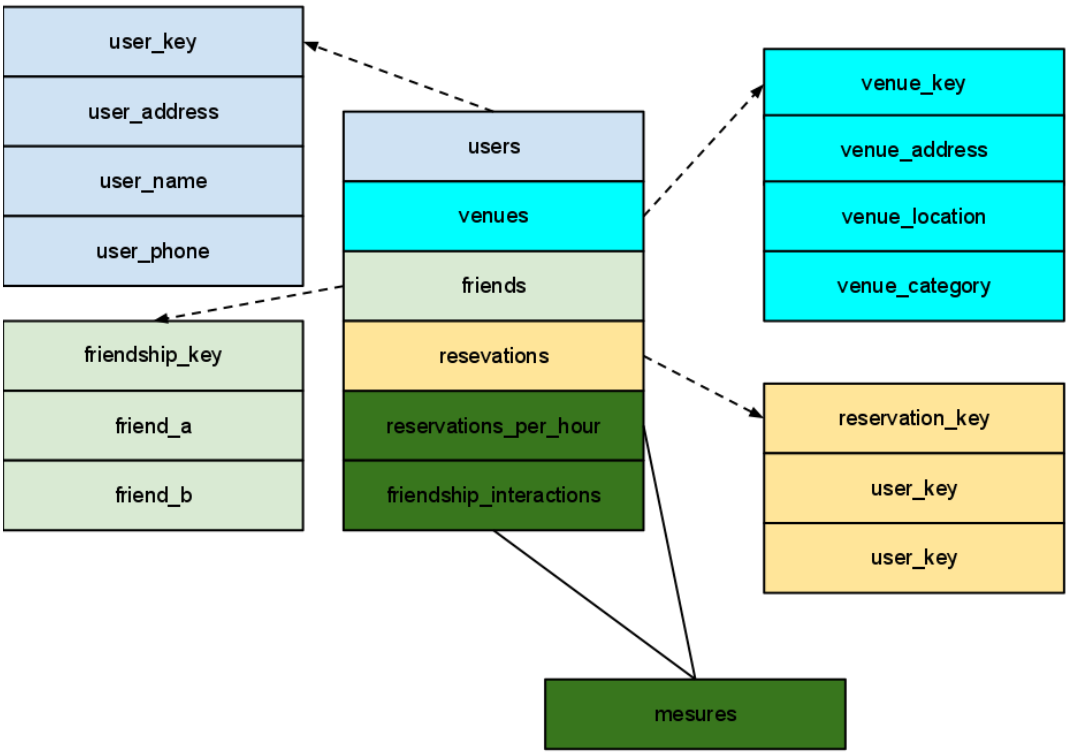
A location-based social networking website which provides check-in services hires you to help them build a data warehouse.

Users of this service can "check-in" at venues using mobile device applications by running the applications and selecting from a list of venues that the application locates nearby. Also, users can "add" each other as "friends". The website also has sufficient information about venues, including address, GPS location, and category of the venue (e.g., a Japanese restaurant), and users tend to provide their personal information to the website when they register.

(a) Design a data warehouse that may facilitate effective on-line analytical processing for this website (provide both schema and measures, also explain why).

Problem 4 continued on next page...

Solution



The schema chosen allows for easy analytical processing by itemizing the various components of the network. Storing friendship interaction in a separate table allows for optimized analytics to be performed on who is interacting with who, as well having reservation stored in an additional table enables cross comparison and can easily determine who's friends are in venues or even the same venue near by.

(b) Check-in data collected from the website and mobile applications are noisy. Besides network and device errors, are there any other reasons which might cause noises in this data set? For the reason you come up with, discuss a method that can clean-up check-in data effectively in the data warehouse.

Solution

Bots could be clogging the system to inflate the data collected about particular venues. This can be cleaned up by examining the frequency a user is checking into a venue and the locations they are checking into. If a user checks in at LA at 10:53 then Chicago at 10:54, this a strong indication that this is a bot clogging the system. Other factors may be checked as well such as the friendship interaction to determine if this is a bot whom is likely to have other friends whom are bots.

(c) One may like to performance on-line analytical processing to the checks-in data at different venues by month, by cities and by categories (Italian or Japanese, etc.). How can this be done efficiently in the data warehouse?

Problem 4 continued on next page...

Solution

This may be done efficiently by examining the data in a three demential cube and parsing it from top to bottom row by row.

(d) Hackers create fake profiles on this website. They are using bots to manipulate fake profiles, generate fake check-in data and try to add everyone as their friends (yes this is a common problem for many social network websites, and no, I am not telling you how to write bots). Although bots are trying to mimic real users, they still behave differently, e.g., they check-in at random places (Chicago this minute, Las Vegas next minute), they check-in way too often than real users, and their social network structures are usually very large but also very sparse (your friends on facebook tend to form communities but bots don't do that). Discuss possible solutions on how to identify fake profiles (bots) in your data warehouse.

Solution

A possible solution for finding bots is by ranking the frequency in which the check in, the high ones are likely candidates. Calculating the distance between the previous checkin and the current checkin comparing it to average and the standard deviation. As mentioned in the problem, check the friends of the bots whom are suspicious, likely the users who are bots have connected with networks of bot to mask the fact that they are not real user and likely the bots friends will have suspicious profiles as well.