Data Mining Assignment -1

Submitted by: Sugandha
NetID: sugandh2

Answer 1:

Ans1.1 BIG DATA refers to data in very large volumes. Such data can originate from business applications, scientific experiments, websites, routine retail transactions etc. Big data is characterized by its bulk and a high frequency of creation. Such data cannot be processed in a timely manner using traditional softwares and create a need for a specialized data management system. Advanced data analytics and mining on data warehouses are used for big data processing.

Analysis of Big Data can give a lot of insights into the processes that have created them, and can be used to find useful patterns and make corrections/ adjustments to processes. It can also be used to make predictions about future data by extrapolating current trends.

Ans 1.2. Big Data analysis is important in the near future for the following reasons:

a. For businesses and retail sector, it can provide a competitive edge by basing the business decisions on all the available data rather than a small subset. It can help in pattern discovery, fraud detection and demand forecasting.

b. For scientific and research applications, the ability to process data rapidly will lead to advancement in the knowledge of fields where data analysis was previously a bottleneck. This includes areas like astronomy, bioinformatics, statistical surveys.

c. Increased capability to analyze large data volumes will provide a new perspective to solve problems in areas like Artificial Intelligence, Machine Learning and Information Retrieval that have traditionally relied on mathematical modeling of processing.

Ans 1.3. Two such prominent applications in today's time are big data in education and healthcare sectors:

EDUCATION:

Big data analysis can provide a better understanding of the learning process of students. Softwares to analyse the learning process and where majority of students are typically making similar mistakes can help the instructors to improve their teaching methods. Statistical study of learning patterns will be of greater help than classroom teaching methods, since it is not possible to analyse such trends in a classroom environment.

For an exam administered to thousands of students at the same time, a number of parameters such as the time taken to answer a particular question, performance in each subject area, and general blind spots in understanding can be easily recorded and analysed. It can help the instructors to gauge what teaching methods work better than the others. Moreover, this also opens up the option for customizations of study softwares based on the classes of students depending on their recorded performance in online tests. Data driven techniques can also be used to provide immediate performance feedbacks when the questions are still fresh in the student's mind.

HEALTHCARE:

Use of data mining in healthcare sector is gaining importance. It benefits all parties involved. Doctors can use it to study past cases that were similar to current and the treatment given, for fast processing of X-Ray scans etc. Patients will receive more accurate and affordable treatment.
A good example of this is Google flu trends that can predict flu many days before official reports. Such mining can help hospitals and pharmacies be prepared in advanced to handle the increased inflow of such patients.
There is a huge volume of readily available health care records data. Once a large database of such patterns is collated, standardized treatment methodologies can be developed for diseases.
Apart from these obvious applications, big data analysis in healthcare can also help hospitals in CRM and insurance companies in detecting fraud. There can be other research based applications like data mining for study of DNA genome samples for discovering new cures.

Answer 2:

Ans 2.1. a. Equal width partitioning

Width of each bin = (90.7 - 6)/5 = 16.94

Bin 1 (6-22.94(inclusive)) [23 values]: 6, 6.7, 7.5, 7.6, 8.4, 9.3, 9.4, 9.6, 10.6, 11.2, 11.3, 11.5, 11.5, 11.9, 13, 13.8, 14.1, 14.9, 15.2, 15.8, 16.5, 17.7, 21.4
Bin 2 (22.94-39.88(inclusive)) [0 values]:
Bin 3 (39.88-56.82(inclusive)) [1 value]: 55.5
Bin 4 (56.82-73.76(inclusive)) [8 values]: 60, 61.4, 62.1, 64.9, 64.9, 69, 70.1, 73.4
Bin 5 (73.76-90.7(inclusive)) [8 values]: 74.9, 76.8, 77, 77.9, 80.1, 81.2, 83.8, 90.7

b. equal depth partitioning
Depth of each bin = 40/5 = 8

Bin 1: 6, 6.7, 7.5, 7.6, 8.4, 9.3, 9.4, 9.6
Bin 2: 10.6, 11.2, 11.3, 11.5, 11.5, 11.9, 13, 13.8
Bin 3: 14.1, 14.9, 15.2, 15.8, 16.5, 17.7, 21.4, 55.5
Bin 4: 60, 61.4, 62.1, 64.9, 64.9, 69, 70.1, 73.4
Bin 5: 74.9, 76.8, 77, 77.9, 80.1, 81.2, 83.8, 90.7

Ans2.2. Sorting and partitioning at 21.4 (because the gap in T-virus value is the highest at this point), we get 2 partitions with 23 and 17 values each:
Partition 1: 6, 6.7, 7.5, 7.6, 8.4, 9.3, 9.4, 9.6, 10.6, 11.2, 11.3, 11.5, 11.5, 11.9, 13, 13.8, 14.1, 14.9, 15.2, 15.8, 16.5, 17.7, 21.4
Partition 2: 55.5, 60, 61.4, 62.1, 64.9, 64.9, 69, 70.1, 73.4, 74.9, 76.8, 77, 77.9, 80.1, 81.2, 83.8, 90.7

a. *Group 1*:

- Mean = sum of values/ 23 = 274.9/23 = 11.952
- Q1 = 25th percentile = 6th value (since 23/4~6) = 9.3
- Median = middle value = 12th value = 11.5

- Q3 = 75th percentile = 14.9
- standard deviation

$$s = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \overline{x})^2}$$

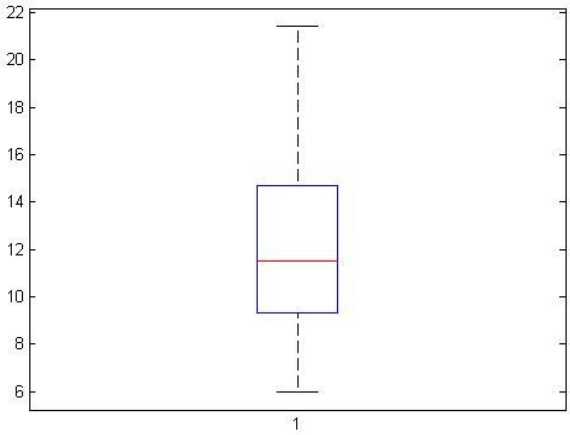
= √14.67534 = 3.83

Group 2:

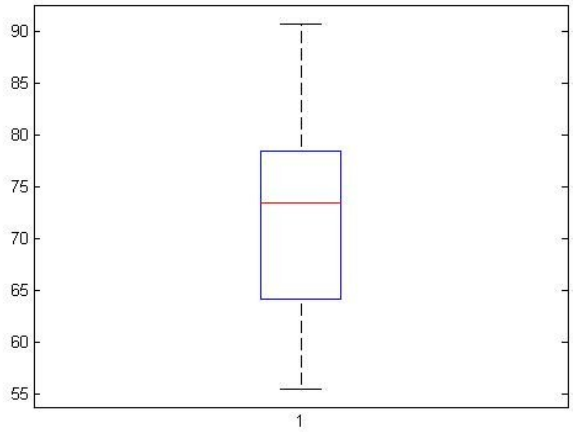
- Mean = 71.982
- Q1= 63.5
- Median = 73.4
- Q3= 79
- standard deviation = √92.02779 = 9.593

b. Box plots for T-virus measures for each group:

Group 1:



Group 2:



c. Normalized values of Anti-A and Anti-B for each group are calculated as:
Normalized value = new_min + (value – min/ max - min)

Assuming new_min=0 and new_max=1, the corresponding values in the two groups become:

Group 1:

Anti-A (ml)	Anti-B (ml)	T-Virus (%)	Anti A normalized	Anti B normalized
2.77	98.1	6	0.2661	1.0399
9.79	16.8	6.7	0.9846	0.1638
9.29	13.6	7.5	0.9335	0.1293
3.41	1.6	7.6	0.3316	0.0000
9.83	55.3	8.4	0.9887	0.5787
5.76	94.4	9.3	0.5722	1.0000
2.73	67.2	9.4	0.2620	0.7069
0.17	41.9	9.6	0.0000	0.4343
7.5	22.7	10.6	0.7503	0.2274
0.21	39.6	11.2	0.0041	0.4095
0.66	61.6	11.3	0.0502	0.6466
7.9	15.5	11.5	0.7912	0.1498
2.84	78.9	11.5	0.2733	0.8330
2	93.3	11.9	0.1873	0.9881
5.07	87.9	13	0.5015	0.9300
2.14	85	13.8	0.2016	0.8987
9.94	16.4	14.1	1.0000	0.1595
7.21	20.6	14.9	0.7206	0.2047
4.41	4.8	15.2	0.4340	0.0345
8.7	48.3	15.8	0.8731	0.5032
3.05	86.3	16.5	0.2948	0.9127

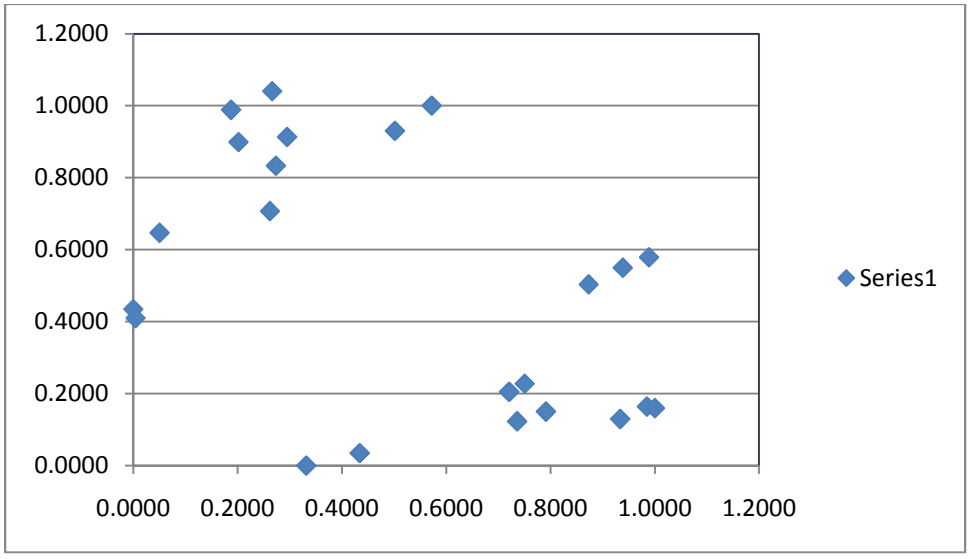
9.34	52.6	17.7	0.9386	0.5496
7.36	13	21.4	0.7359	0.1228

Group 2:

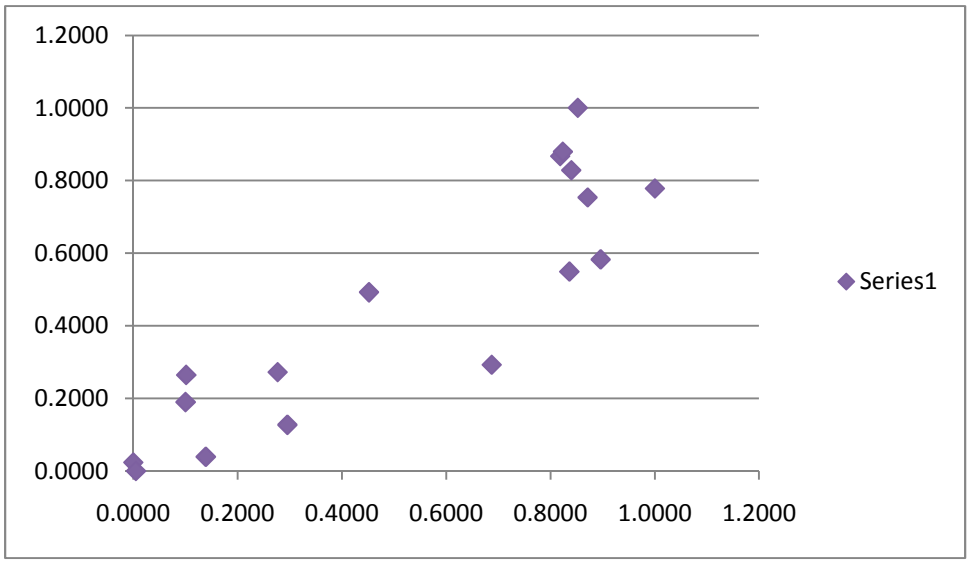
7.79	61.4	55.5	0.8961	0.5826
4.24	53.3	60	0.4518	0.4922
0.63	11.3	61.4	0.0000	0.0234
1.43	26.2	62.1	0.1001	0.1897
1.44	32.9	64.9	0.1014	0.2645
7.34	83.4	64.9	0.8398	0.8281
1.74	12.7	69	0.1389	0.0391
6.12	35.4	70.1	0.6871	0.2924
7.59	76.7	73.4	0.8711	0.7533
7.31	58.4	74.9	0.8360	0.5491
7.17	86.9	76.8	0.8185	0.8672
2.99	20.6	77	0.2954	0.1272
2.84	33.6	77.9	0.2766	0.2723
0.67	9.2	80.1	0.0050	0.0000
8.62	78.9	81.2	1.0000	0.7779
7.44	98.8	83.8	0.8523	1.0000
7.21	88	90.7	0.8235	0.8795

d. Scatter plots for Anti-A vs. Anti-B

Group 1:



Group 2:



e. Pearson's Coefficient:

$$r_{A,B} = \frac{\sum_{i=1}^{n}(a_i - \bar{A})(b_i - \bar{B})}{n\sigma_A\sigma_B} = \frac{\sum_{i=1}^{n}(a_i b_i) - n\bar{A}\bar{B}}{n\sigma_A\sigma_B},$$

where n is the number of tuples, a_i and b_i are the respective values of A and B in tuple i, $\overline{A}$ and $\overline{B}$ are the respective mean values of A and B, σ_A and σ_B are the respective standard deviations of A and B and $\sum a_i b_i$ is the sum of the AB cross-product (i.e., for each tuple, the value for A is multiplied by the value for B in that tuple).

From the normalized tables in part (c),

Group 1:

$\sum_{i=1}^{n} a_i b_i$ = 4.908412

$\overline{A}$ =0.5259, $\overline{B}$ = 0.4860

σ_A = 0.3443 , σ_B = 0.3399

n = 23 (size of first partition)

Therefore, $r_{A,B}$ = (4.720214 – 23 * 0.5259 *0.4860) / 23 * 0.3443 * 0.3399

= -1.158 / 2.6916= - 0.4302 (negatively correlated)

Group 2:

$\sum_{i=1}^{n} a_i b_i$ = 5.984573

$\overline{A}$ = 0.5290, $\overline{B}$ = 0.4670

σ_A = 0.3676 , σ_B = 0.3395

n = 17 (size of first partition)

Therefore, $r_{A,B}$ = (5.9845 – 17 * 0.5290 * 0.4670) / 17 * 0.3676 * 0.3395

= 1.7847 / 2.121 = 0.8414 (positively correlated)

f. Based on the analysis (a) – (e), we observe the following:
   - Mean and median values for T-virus are close to each other in both groups, which means that the data is evenly distributed. Standard deviation is much lower in first group, which means that values are closer to mean value.
   - The scatter plots and Pearson's coefficient show that Anti-A and Anti-B are negatively correlated for first group, and positively correlated for the second group.

     Based on these observations, it can be concluded that the average T-virus value is much lesser when the Anti-A and Anti-B quantities used are negatively correlated with each other, i.e*. the dose of both medicines should not be increased simultaneously*.

Answer 3.

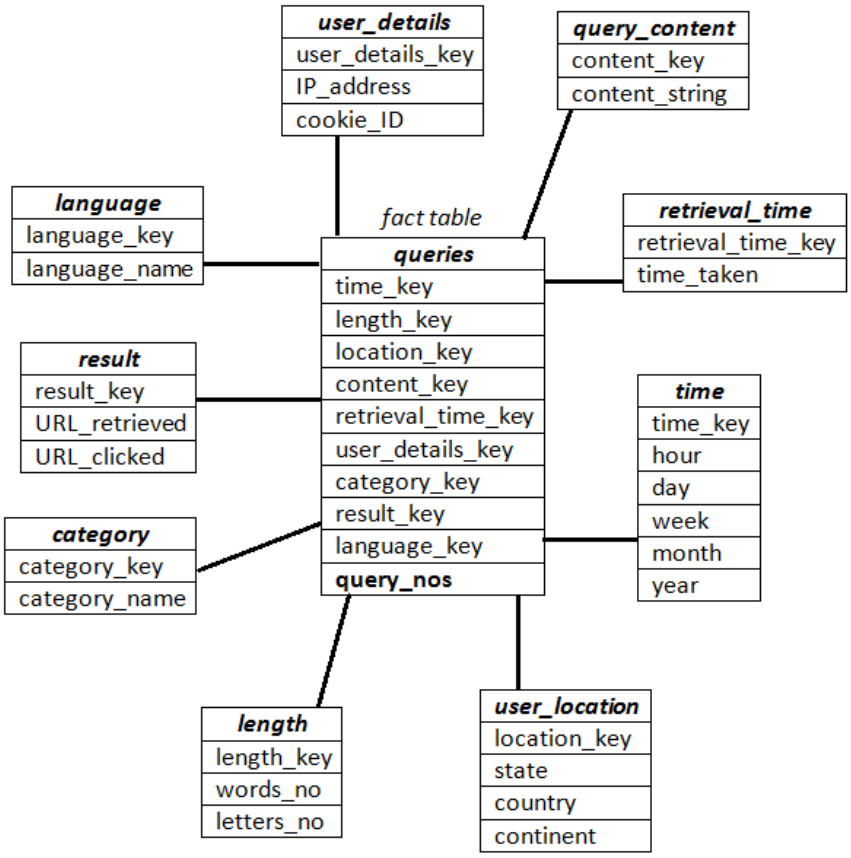
Ans 3.1. Description of the schema:

The fact table in our star schema measures the number of user queries.

Following are the dimension tables present:

1. user_location: contains the location hierarcy – continent to state
2. time: contains time of query – with query hour being the lowest granularity
3. category: query category name
4. language: query language
5. user_details: stores user's IP address and cookie ID for individual identification
6. length: query length in words and letters
7. query_content: stores the exact query for later performace analysis
8. retrieval_time: time taken for results to be displayed
9. result: URLs of link retrieved in response to the query, and user log entry of the results click (for relevance and precision studies later)

Note: I am assuming that an individual user is tracked by IP_address (query numbers cannot be taken equal to number of users, as one user might make multiple queries)



Ans. 3.2

a. Top query category in the United States during 6pm – 10pm can be found using the following OLAP operations:
   i) Drill down in query_category table to 'category_name' , user_location table to 'country' and time table to 'hours'
   ii) Dice along dimensions time and user_location selecting hours= 6 to 10 (range query) and user_location='USA'
   iii) Aggregate query numbers along time dimension using *sum*() operator
   iv) Find the cell with *max*() query number and retrieve the corresponding query category

b. Following sequence of OLAP operations can be used to find the average number of users who use the search engine for news with Spanish language
   i) Drill down to years in time table, IP_address in user_details table and category_name in query_category table.
   ii) Slice along dimensions category (performing a selection on news) and language (Spanish)
   iii) Retrieve *distinct*() IP_address from user_details table
   iv) Take *average*() over years (or other time measure, as desired)

c. Find countries where half of the users use the search engine for image

i) Drill down along dimension user_location to 'country' and along user_deails to 'IP_address'

ii) Save the country-wise *count*(IP_address) in a temporary table.

iii) Next, drill down to query_category level and slice along query_category='image' and retrieve country values corresponding to cells where *count*(IP_address) has reduced to half of step (ii)

Ans.3.3. Standard deviation can be calculated by drilling down the user_location and time dimensions to country / query hour and using sum() and count() functions to calculate mean values.
Standard deviation is an algebraic measure and its component calculations can be done using algebraic measures.
Standard deviation is given as:

$$s = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \overline{x})^2}$$

Sum() and count() are algebraic measures. Mean can be calculated efficiently by using sum() / count().
Being incremental in nature, subsequent values can be calculated efficiently without repeating the count() and sum() measures for all existing values.
Instead, we simply need to calculate the squared sum of the new numbers, add that to the existing squared sum, and plug in the updated count into the calculations to obtain the new standard deviation.
This can be done without looking at the whole data set and is thus easy to compute

Answer 4.

The data warehouse can help the linear regression to predict search engine market shares because it stores data along various dimensions in a way that OLAP operations can be used conveniently. For example, a category wise study of existing query numbes can be used to predict future query numbers in the same category, and a comparison of these with competitor's share can be used to predict the future market share. This calculation can be done for any dimension, such as language, user location, time of query etc., and OLAP operations such as min(), max() , average() can be used after dicing the relevant dimensions.
A linear regression equation is of the form:
y=a + bx
where x and y are the two dimensions.

Ans4.1
Parameters a and b can be calculated as:

$$a = \frac{(\sum y)(\sum x^2) - (\sum x)(\sum xy)}{n(\sum x^2) - (\sum x)^2}$$

$$b = \frac{n(\sum xy) - (\sum x)(\sum y)}{n(\sum x^2) - (\sum x)^2}$$

where n = number of data points (sample size)
To calculate these parameters in our data warehouse, we need values for x2, y2 and xy at various data points. Taking time as the independent variable (x) and market share along y-axis, we need to consider

history of n data points. The market share value here can be refined by choosing the dimensions to be considered (e.g. market share for image search in USA).

Ans.4.2 Count n of the number of samples can be carried out by using algebraic count() operator.
Other operations like products, division and squares can be reduced to sum() operations, which is again algebraic and can be computed efficiently.

Therefore, effectively, we only need to perform algebraic operations sum and count for the calculation of parameters. Since algebraic operations can be calculated efficiently in databases by using data cube technology, the data warehouse can be used for the linear regression analysis.

Further, since it will be an incremental operation with increasing data volumes (stream data), techniques exist for increasing the efficiency of calculations by using an exception based computation method using tilt time frame in a two layered (m and o) framework [Han et al., 2002].