**Data mining – CSC240**

**Student name: Hizkia Febianto**

2.3. *Compute an approximate median value for the data*

| ***age*** | ***frequency*** |
| --- | --- |
| 1–5 | 200 |
| 6–15 | 450 |
| 16–20 | 300 |
| 21–50 | 1500 |
| 51–80 | 700 |
| 81–110 | 44 |

Since we have N = 3,194 observations, the median value of these data must be around the (3194/2) = 1,597th observation. The 1597th observation fall within the range of 21 – 50 years old. The approximate median value is calculated as follows.

200 + 450 + 300 = 950

2.6. *Given two objects represented by the tuples (22, 1, 42, 10) and (20, 0, 36, 8):*

a. Euclidean distance:

b. Manhattan distance:

c. Minkowski distance:

d. Supremum distance: Since the third attribute gives us the greatest distance, the supremum distance in this case is (42-36) = 6.

2.7*. The median is one of the most important holistic measures in data analysis. Propose several methods for median approximation. Analyze their respective complexity under different parameter settings and decide to what extent the real value can be approximated. Moreover, suggest a heuristic strategy to balance between accuracy and complexity and then apply it to all methods you have given.*

The first method that can be used to approximate median value is called successive binning algorithm.[[1]](#footnote-1) This algorithm is based on a lemma that a median of a data set will always fall within one standard deviation from the mean value. In his paper, Tibshirani suggested two different approaches, namely *binmedian* and *binapprox.* Binmedian algorithm competes directly with *quickselect* algorithm, the fastest algorithm to calculate median. In a condition where the exact value of the median is not really important, binapprox algorithm can be used.

Binmedian algorithm is implemented as follows1:

1. Compute the mean µ and standard deviation σ.
2. Form B number of bins across [] and map each data point to these bins
3. Find the bin *b* that contains the median
4. Recurse on the set of points mapped to *b*

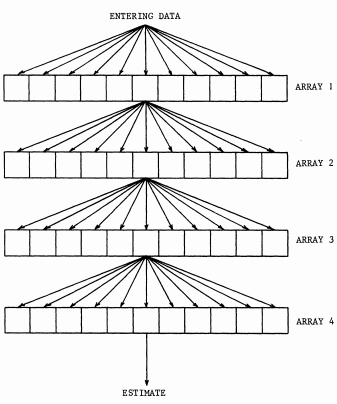
Tibshirani explained that the runtime of binmedian algorithm directly depended on the data distribution. If there are a lot of data points around the median value, this algorithm will need some time to generate its output. However, even in the worst case scenario, binmedian algorithm still has the same complexity as quickselect algorithm, which is O(n).

Binapprox algorithm is implemented as follows1:

1. Compute the mean µ and standard deviation σ.
2. Form B number of bins across [] and map each data point to these bins
3. Find the bin *b* that contains the median
4. Return the midpoint of bin *b.*

Since binapprox algorithm does not perform the recursive step (step #4), this algorithm does not depend on the distribution of the data. Even though this algorithm has O(n) complexity in the worst case scenario, it runs faster than binmedian and quickselect in practice. The weakness of this algorithm is observable when the standard deviation of the data is large. In this case, the approximation could be very different than the actual median value.

The second method that can be used to estimate median is the Remedian algorithm that was proposed by Peter J. Rouseeuw and Gilbert W. Basset, Jr.[[2]](#footnote-2) This algorithm solves the problem of high storage space that is required by conventional algorithms when dealing with a very big data set. Since this algorithm only needs *k* arrays of size *b*, total space required by this algorithm is only *bk*. Here we assume that the total number of data objects that we have is n = bk. Generalization when the number of data objects does not follow this rule is also available.

The illustration of this algorithm is shown in the figure below. Say we use 4 arrays of size 11. The data will first enter the topmost array (array 1). The first 11 observations fill array 1 and then the median of these observations is stored in array 2. After that, the next group of 11 observations is entered to array 1 and the median of these observation is stored in array 2. Similar to array 2, array 3 will be filled by median values from array 2. When 114 data values have been entered, array 4 is full and the median becomes the final estimate. This method uses only 44 storage positions and its speed is of the same order of magnitude as that of the ordinary average.

In his paper, Rouseeuw and Basset estimates that the computation time of their algorithm is O(n) but it will be faster if a large base is used.

2.8. *Suppose we have the following 2-D data set:*

|  | ***A*1** | ***A*2** |
| --- | --- | --- |
| ***x*1** | 1.5 | 1.7 |
| ***x*2** | 2 | 1.9 |
| ***x*3** | 1.6 | 1.8 |
| ***x*4** | 1.2 | 1.5 |
| ***x*5** | 1.5 | 1.0 |

1. *Given a new data point, x = (1.4, 1.6) as a query, rank the database points based on similarity with the query using Euclidean distance, Manhattan distance, supremum distance, and cosine similarity.*

The list of Euclidean distance between the query data and each data point is as follow.

Hence, the order of the database points based on similarity with the query using Euclidean distance from the most similar to the least similar is .

The list of Manhattan distance between the query and each data point is as follow.

Hence, the order of the database points based on similarity with the query using Manhattan distance from the most similar to the least similar is .

The list of supremum distance between the query and each data point is as follow.

Hence, the order of the database points based on similarity with the query using Manhattan distance from the most similar to the least similar is . If we use supremum distance to measure similarity between the query data and each data point, there are two pairs of data point that have identical distance to the query. Data point is the most similar data with the query data since the distance between these two data are the shortest. Data points and are in the second rank together since both of them have the same distance to the query data. Data point and are in the third rank.

Now, cosine similarity will be used to determine the order of data points based on their similarity with the query data.

Hence, the order of the data points from the most similar to the least similar is ,,,,.

1. *Normalize the data set to make the norm of each data point equal to 1. Use Euclidean distance on the transformed data to rank the data points.*

Normalized data points are calculated using the following formula , where is a 2-dimensional vector built from each data point and is a unit vector of . The following table summarizes the normalized data points.

|  | ***A*1** | ***A*2** |  |
| --- | --- | --- | --- |
| ***x*1** | 0.662 | 0.75 | 2.267 |
| ***x*2** | 0.72 | 0.69 | 2.759 |
| ***x*3** | 0.664 | 0.748 | 2.408 |
| ***x*4** | 0.62 | 0.781 | 1.921 |
| ***x*5** | 0.83 | 0.55 | 1.803 |

After normalization, the query data becomes

Euclidean distances between the query data and each data points are listed below.

Hence, the order from the most similar to the least similar is x1, x3, x4, x2, x5.

3.1. *Data quality can be assessed in terms of several issues, including accuracy, completeness, and consistency. For each of the above three issues, discuss how data quality assessment can depend on the intended use of the data, giving examples. Propose two other dimensions of data quality.*

Answer:

Accuracy of the data relates to the numbers of errors in the data. The importance of data accuracy varies based on the use of the data. For sensitive applications, such as fraud detection or money laundering detection, accuracy of the data is extremely crucial. For other applications, such as email spam detection or market basket analysis, some errors in the data would not really affect the performance a prediction model. In these applications, an accuracy of 70-80% may be enough to justify an action but in some more sensitive applications, a user may want to have an accuracy of 95% or more.

An example of data inconsistency is different date formats in the data. This may happen, for example, in a global firm that operates in two countries that use different date formats. When a subsidiary of this global company wants to analyze the data about their regional sales, the date format is consistent. However, if its global headquarter wants to analyze global sales using data from different countries, there might be an inconsistency in the data format.

Incompleteness in a data is usually due to missing values or aggregated values. A fashion retail company may store its sales volume per city even though it may have many stores in a particular city. If the data is used to analyze patterns in a nationwide data, the fact that sales volume is summed up per city is not a problem. But suppose that all stores in a particular city experience a sales decrease and the company wants to find a solution based on data. In this example, sales volume per store is missing and hence the data is incomplete.

Other dimensions of data quality are believability and interpretability. Believability reflects how much data are trusted by users and interpretability reflects how easy are the data understood. If a data set are outdated or contains too many errors, users may not trust the data anymore and the analysis based on the data will also be doubted. This is related to the believability dimension. Data containing many code that are difficult to be interpreted by users would create a problem in the analysis. This is about data interpretability.

3.3. *Exercise 2.2 gave the following data (in increasing order) for the attribute age: 13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70.*

a. *Use smoothing by bins means using a bin depth of 3.*

|  |  |  |
| --- | --- | --- |
| Bin # | Original | After smoothing |
| 1 | 13,15,16 | 15,15,15 |
| 2 | 16,19,20 | 18,18,18 |
| 3 | 20,21,22 | 21,21,21 |
| 4 | 22,25,25 | 24,24,24 |
| 5 | 25,25,30 | 27,27,27 |
| 6 | 33,33,35 | 34,34,34 |
| 7 | 35,35,35 | 35,35,35 |
| 8 | 36,40,45 | 40,40,40 |
| 9 | 46,52,70 | 56,56,56 |

I assume that we need to keep the data in an integer format so some mean values are rounded to the nearest integer.

b. *How might you determine outliers in the data?*

A simple method to detect outliers is by calculating mean and standard deviation of the data and find values that fall outside a range of two standard deviations from the mean. The mean of the data above is 29.96 and their standard deviation is 12.94. Hence, the range of two standard deviations from the mean is (4.08,55.85). From this range, we can conclude that 70 is probably an outlier.

c. *What other methods are there for data smoothing?*

Other method for smoothing data are exponential smoothing, moving average smoothing, and Holt-Winters Smoothing.

3.5. *What are the value ranges of the following normalization methods?*

min-max normalization: [new\_minA, new\_maxA]

z-score normalization: ()

z-score normalization using the mean absolute deviation instead of standard deviation: ()

normalization by decimal scaling: [-1, 1]

3.7. *Using the data for age given in Exercise 3.3, answer the following:*

*Use min-max normalization to transform the value 35 for age onto the range [0.0, 1.0]:*

Max\_value = 70; min\_value = 13; v = 35; new­\_max = 1.0; new\_min = 0.0

*Use z-score normalization to transform the value 35 for age, where the standard deviation of age is 12.94 years:*

Mean = 29.96; standard deviation = 12.94; v = 35

*Use normalization by decimal scaling to transform the value 35 for age:*

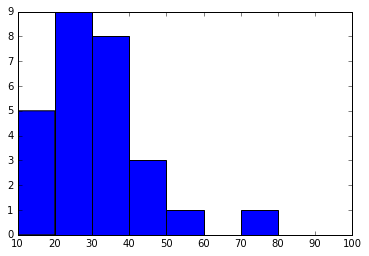
v = 35; j = 2

*Comment on which method you would prefer to use for the given data, giving reasons as to why.*

I would prefer to use normalization by decimal scaling for the given data. It is because decimal scaling keeps the distribution of the data. Besides, this method is also relatively simple.

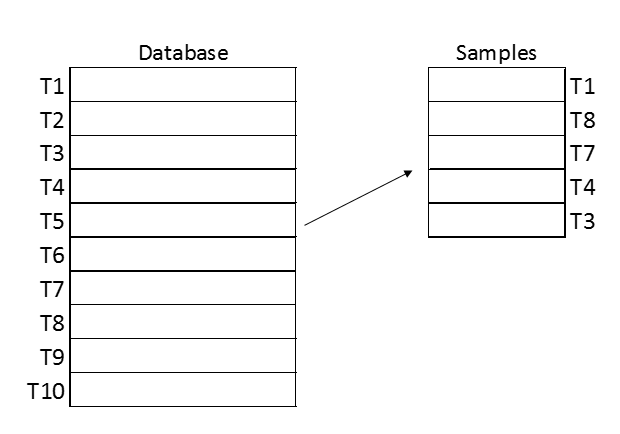
3.11. *Using the data for age given in Exercise 3.3,*

*a. Plot an equal-width histogram of width 10.*

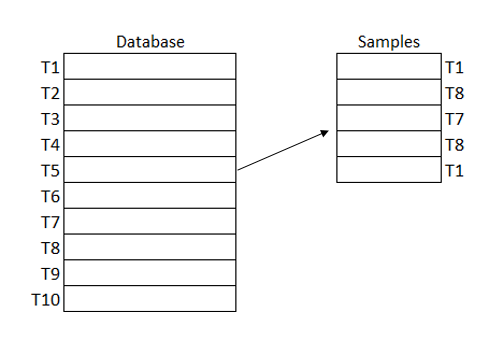


b. *Sketch examples of each of the following sampling techniques: SRSWOR, SRSWR, cluster sampling, and stratified sampling. Use samples of size 5 and the strata "youth," "middle-aged," and "senior."*

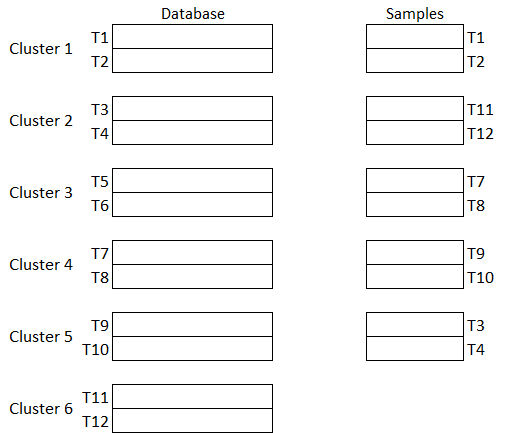
Simple random sample without replacement (SRSWOR):



Simple random sample with replacement (SRSWR):



Cluster sampling:



Stratified sampling:



3.13. *Propose an algorithm, in pseudocode or in your favorite programming language, for the following:*

*a. The automatic generation of a concept hierarchy for nominal data based on the number of distinct values of attributes in the given schema.*

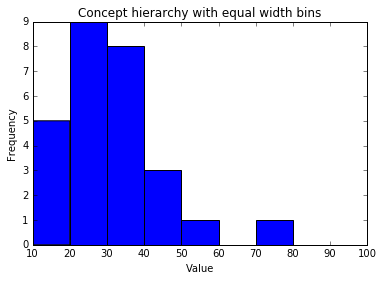
The code implements the following algorithm:

1. Load data set
2. Enumerate nominal attributes and count the number of distinct values for each attributes.
3. Sort the number of distinct values in ascending order.
4. Hierarchy is formed from attributes with lower number of distinct values to attributes with higher number of distinct values.

From the code, the following hierarchy is formed:

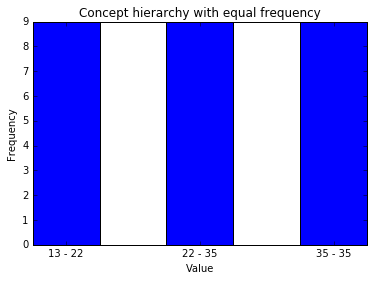
*b. The automatic generation of a concept hierarchy for numeric data based on the equal-width partitioning rule.*

Concept hierarchy for numeric data based on the equal-width partitioning rule is similar to a histogram with equal bin width. The following array is used as an example array = 13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70.



*c. The automatic generation of a concept hierarchy for numeric data based on the equal-frequency partitioning rule.*

The code is in a separate file. Please refer to the code.



1. Tibshirani, R. J. (2008), *‘Fast Computation of the Median by Succesive Binning’,* Retrieved October 1, 2016 from Carnergie Mellon University: <https://www.stat.cmu.edu/~ryantibs/papers/median.pdf> [↑](#footnote-ref-1)
2. Rouseeuw, P. J. & Gilbert W. B (1990), *‘The Remedian: A Robust Averaging Method for Large Data Sets’*, Journal of the American Statistical Association, vol. 85, pp. 97-104. [↑](#footnote-ref-2)