Analytical Data Mining (CIS 787)
Author: Advait Ramesh Iyer
NetID: aiyer01 SUID: 330623485

Homework Assignment 1

Problem 1: Chapter 2, Problem 2

Following are the appropriate classifications for attributes:
1) Time in terms of AM or PM: **Binary, qualitative, nominal**
2) Brightness as measured by a light meter: **Quantitative, continuous, ratio**
3) Brightness as measured by people's judgements: **Discrete, qualitative, ordinal**
4) Angles as measured in degrees between 0 and 360: **Quantitative, continuous, ratio**
5) Bronze, silver and gold medals as awarded at the Olympics: **Qualitative, ordinal, discrete**
6) Height above the sea level: **Quantitative, continuous, ratio**
7) Number of patients at a Hospital: **Quantitative, discrete, ratio**
8) ISBN numbers for books: **Qualitative, nominal, discrete**
9) Ability to pass light in terms of the following values: opaque, translucent, transparent: **Qualitative, ordinal, discrete**
10) Military rank: **Qualitative, ordinal, discrete**
11) Distance from the center of campus: **Quantitative, continuous, ratio**
12) Density of a substance in grams per cubic centimeter: **Quantitative, continuous, ratio**
13) Coat check number: **Qualitative, nominal, discrete**

Problem 1: Chapter 2, Problem 5

Identification numbers are a unique identifier to distinguish individual's data-points from others.

The best use for identification numbers is, if we do not know when the student enrolled at the University, we can generally conclude it by using this unique identifier. For example, a student who is a senior and arrived earlier will generally have an older ID than a new student. Due to the high correlation between the student ID and the date of enrollment, this is the most viable use-case of using Identification Numbers for prediction.

Problem 1: Chapter 2, Problem 13

a) In case of k-nearest neighbor algorithm with duplicates, number of duplicates can be denoted as:

$$Duplicates = K \times Number\,of\,Clusters \tag{1}$$

This leads to a data loss of factor $\frac{Duplicates}{Number\,of\,Clusters}$, which is substantial amount of information not being used in the model.

b) The only way to solve this problem is by removing duplicates, which would help us avoid the above-stated loss of information.

Problem 1: Chapter 2, Problem 15

a) When $n \times \frac{m_i}{m}$ elements are selected randomly from each group,

$$Total\,number\,of\,ways\,for\,selection\, = \prod_{i=1}^{k} m_i \times \frac{n}{m} \tag{2}$$

Now, this implies that, $Total\,number\,of\,ways\,for\,selection\, = (\frac{n}{m})^k \prod_{i=1}^{k} m_i$

b) When we randomly select n elements without taking into regard which group it belongs to,

$$Total number of ways of selection = \prod_{i=1}^{n} m_i = m^n \tag{3}$$

Now, we know that $m^n > (\frac{n}{m})^k \prod_{i=1}^{k} m_i$, which means that the possibility of more people of the same group getting selected is higher in the second case.

Therefore, if we are seeking to get a more uniform distribution, with equal number of people from separate groups, we will implement the first method. However, if we seek to randomly sample, we will use the second method.

Problem 1: Chapter 2, Problem 16

a) Case 1: When the term occurs only in one document:

$$tf_{i,j}^{'} = tf_{i,j} \times \log \frac{m}{df_i} = 0 + 0 + 0 + ... + (1) \times \log \frac{m}{1} = \log m \tag{4}$$

Case 2: When the term occurs in every document:

$$tf_{i,j}^{'} = tf_{i,j} \times \log \frac{m}{df_i} = (1) \times \log \frac{m}{m} + (1) \times \log \frac{m}{m} + ... + (1) \times \log \frac{m}{m} = 0 + 0 + 0.... + 0 = 0 \tag{5}$$

Overall, the effect of this transformation leads to increase in the preference of choosing less-frequent word instead of more-frequently occuring word.

b) The TF-IDF algorithm is used in Natural Language Processing (NLP), generally when sorting of documents according to the query passed by the user.

For example, if the query of the user contains "The", there may be documents online with "the the the the the" repeating endlessly. If the algorithm takes only frequency into consideration, it will output this irrelevant document instead of taking into consideration other parts of the document, just because "The" is the most common word. Therefore, TF-IDF helps in scaling down the effect of frequency.

Problem 2: Sampling: Question 1

When we sample with replacement, there are no dependencies between the two samples chosen. It works in the cases when we do not want variables to have any kind of covariance. Sampling with replacement helps with computations, since covariance is set to zero.
For example, if we repeatedly sample from a group of 20 students, allowing replacements, we can conduct infinite sampling without spending any resources in calculating the joint probabilities, since they are independent events.

On the other hand, in the case of sampling without replacement, the sampling already conducted effects the chances of other items getting selected (increases, since one item has left the population set). This means, there is covariance between each of the sample values. It is a little more computationally expensive than sampling with replacement.
For example, if we are sampling from a group of 20 students, without allowing replacements, it is important to calculate joint probabilities so that the rules of sampling are not broken. This causes increase in computational complexity.

Problem 2: Sampling: Question 2

The aim of Principle Components Analysis (PCA) is to identify best fit lines within the high-dimensional data, in a way that the axes are rotated to remove covariances of key variables with the noise. Once the covariances are removed, the variances within the diagonal of the matrix are the eigenvalues, which help in decorrelating the dataset to reduce the dimensionality of data.

It is important to note that the Principle Components pass through data-points with the highest variance, which captures the essence of more information than a random sampling algorithm, since it captures higher amount of variance without much data loss.

Problem 4: Weka

Step 1. Center data (having zero mean):
Command = weka.filters.unsupervised.attribute.Center
Parameter = Center

Step 2. Removing attribute 2 to 4:
Command: weka.filters.unsupervised.attribute.Remove -R 2-4
Parameter = -R 2-4

Step 3. Removing all attributes but the last:
Command: weka.filters.unsupervised.attribute.Remove -V -R last
Parameter = -V -R last

Step 4: Reordering attributes 1,2,3,4,5 as 5,4,1,2,3:
Command: weka.filters.unsupervised.attribute.Reorder -R 2-last,1
Parameter: -R 2-last,1

Step 5: Removing instances with missing values:
Command: weka.filters.unsupervised.instance.RemoveWithValues -S 0.0 -C first-last -L first-last -M
Parameter: -S 0.0 -C first-last -L first-last -M

Step 6: How is a missing value denoted in an ARFF file:
Command: (?) Single question mark

Step 7: What does visualize all do:
The command provides bird's eye view of the dataset through visualizing distribution of each attribute.

Step 8: Sampling 20% of your data:
Command: weka.filters.unsupervised.instance.Resample-S1-Z20.0
Parameter: -S1-Z20.0

Step 9: Removing all instances where the 3nd feature value is equal to 'x':
Command: weka.filters.unsupervised.instance.RemoveWithValues-S0.0-C3-Lfirst-last-M
Parameter: -S0.0-C3-Lfirst-last-M