Cleaning and Mining of Payroll Datasets

Anonymous

1. INTRODUCTION

The domain for the project is Employee Payroll Statistics. The reason behind it is that payroll data usually contains numeric attributes like salary, overtime, total earnings, etc, and textual attributes like job titles, department, names which generally have duplication, missing values and other noisy data which need to be cleansed properly and prepare them for useful statistical analysis and gain knowledge about trends in salaries, overtime payments and understand about salaries associated with different job titles.

The datasets[1][2] contain employee payroll details of every year starting from the year 2011 of the respective cities. Firstly, these datasets will be downloaded, then the two datasets will be merged. The relational schema will be developed and the dataset will be loaded to the relational database. The idea is to use PostgreSQL for storing the data. The next step will contain data cleaning techniques on the raw data and statistical analysis of the merged data. After the data is cleansed and prepared, a data mining technique will be used and in-depth discussions will be made on the results. Finally, data visualizations will be done on the dataset and insights will be drawn out.

In Section 2, the description of the datasets will be provided. In Section 3, the implementation phases and details will be provided for cleaning and preparing the data. In Section 4, we shall discuss the data mining technique that will be carried out. In Section 5, we shall discuss the results and visualizations that have been drawn after the analysis. In Section 6, the lessons learned while developing the project shall be discussed and in Section 7, the conclusion and future work of the project will be provided.

2. DATASET DESCRIPTION

The datasets will focus on payroll statistics of employees of two cities - Boston and New York City. The important attributes that will be focused on in the datasets are as follows:

- **ID** - The primary key of the datasets. Each ID will be unique.
  Data Type - Numeric

- **Department** - It is the name of the department that an individual works for.
  Data Type - Text

- **Job Title** - It is the job title of the employee.
  Data Type - Text

- **Base Salary** - It is the base annualized salary that is paid to the employee by the department.
  Data Type - Numeric

- **Overtime Paid** - Extra payments to employees for work done in excess hours. It also includes holiday pay.
  Data Type - Numeric

- **Other Compensations** - It includes allowances, reimbursements and other taxable incomes that are not included in the salaries.
  Data Type - Numeric

- **Total Earnings** - Gross earnings of an individual which includes all kinds of incomes of the individual of the whole year.
  Data Type - Numeric

3. DESIGN AND IMPLEMENTATION

The project will involve design of a database, cleaning and preparation of the database using different techniques and then a data mining task to retrieve information from the prepared data. The major technologies that are used for the project are:

- **Python** - a coding language to clean, prepare, store data and also carry out the required data mining task.

- **PostgreSQL** - a relational database management system that will store the data in a tabular form.

- **SQLAlchemy and Psycopg2 driver** - The data is cleaned and prepared thoroughly using Pandas. The SQLAlchemy driver is used to load the data to PostgreSQL and finally psycopg2 driver is used to add the primary key.

3.1 Data Cleaning and Preparation

The initial design involves understanding the data that has been selected and the important attributes that will help complete the data mining tasks. The next major step is to clean and then prepare the datasets and in order to do that, the following techniques have been implemented:

- The Boston Data[1] datasets are distributed over eight years. They have been merged together in order to generate one dataset that represents it completely.

- The NYC Data[2] dataset had two attributes, one which had instances of first names and other which had instances of last name. These attributes were combined to form the 'NAME' attribute. This helped to normalize the names as the Boston Data[1] has only one attribute for name and it also helped to recognize a person with only one attribute rather than two.

- Both datasets have different attribute headings with same meanings. In order to make them compatible and merge them, such attributes had to be renamed so that there are no duplicate attributes. The following example displays code that has been used in order to achieve it.

  ```
  dataset2012.rename(columns={'DETAIL':'DETAIL'},
  inplace=True)

  dataset2016.rename
  (columns={'DEPARTMENT_NAME':'DEPARTMENT'},
  inplace=True)
  ```

- The datasets contained '$' and ',' values in numeric instances which have been cleaned and replaced with empty string before loading them into the PostgreSQL database.

- Data types of three attributes - namely 'REGULAR', 'OVERTIME', 'TOTAL EARNINGS' have been transformed to float data type in order create clear and readable results while calculating central tendencies or else it would give results in exponential powers of 10.

- There are instances in 'REGULAR', 'OVERTIME', 'TOTAL EARNINGS' attributes with negative values which are inconsistent as salaries cannot be negative. In order to correct it, firstly they have been clipped to the value of 0. These instances still create problems in calculating central tendencies and will generate biased and inaccurate results. These instances are further converted to null values. When calculating central tendency measurements such null values will be ignored.

- The **pandas** library in *Python* has been used to create dataframes to clean and prepare the data and later, when the dataset is loaded into the database they are converted to CSV extension files.

- The target column of the prepared data set is a continuous variable called 'TOTAL EARNINGS'. It is difficult to predict a continuous variable because the range of these variables are very long. One way to handle a continuous variable is by quantizing the data into the bins and so, the target attribute is divided into four classes.

  - **Class 0**: Total earnings less than 35,000 dollars
  - **Class 1**: Total earnings between 35,000 and 70,000 dollars
  - **Class 2**: Total earnings between 70,000 and 105,000 dollars
  - **Class 0**: Total earnings greater than 105,000 dollars.

- The final data set is created through merging multiple data sets and each data set consist of different attributes with different units and ranges. In order to get better and efficient results from the data mining process, data was normalized through normalization techniques. It helped build association between the heterogeneous data. Two different normalization techniques have been used:

  - **Standard normalization**: In this, the range of the variable is updated by changing mean to 0 and standard deviation to 1.
  - **Sigmoidal normalization**: Firstly, the z-score is calculated and then we apply eq.(1)

    $$x' = \frac{1 - e^{-a}}{1 + e^{-a}}$$

    Here, 'a' is the value obtained from z-score equation and 'e' is Euler's number.

3.2 Data Mining

Further implementation will focus on a data mining task to gain information from the payroll dataset that was generated. The 'TOTAL EARNINGS' attribute is used as the target variable in different data mining techniques.

Now that the target variable has been created and the underlying groups are known, classification would be a good data mining technique to gain important information from the dataset. The important features in the dataset are understood and the following classification methods are implemented :

- **Zero Classifier** - To compare different classification techniques, a baseline model is created such that a basic expectation of effectiveness of the methods can be understood for the created data set. Zero classifier or the zero rule classifier is a classic example of a baseline model. For every test data instance, this classifier predicts and assigns the majority class randomly. This helps to set up a base for every other classification methods.

- **Random Forest** - This classifier fits a number of decision trees on various samples of a dataset. It uses the average to improve the predictive accuracy of the model and also controls over-fitting. For every decision tree created randomly generated training samples are provided and every tree has a different attribute to split on. The sample size always remains same but the samples are drawn with random replacement. Every tree votes for the the attributes and the best attribute is used by the main model.

A Confusion Matrix is generated in order to evaluate the results that are generated by the classifiers.

4. RESULTS

While preparing the data set and performing the experimentation on the clean data set, the clean and the raw data set were compared. This process helped to understand and visualise the importance of data cleaning in this project.

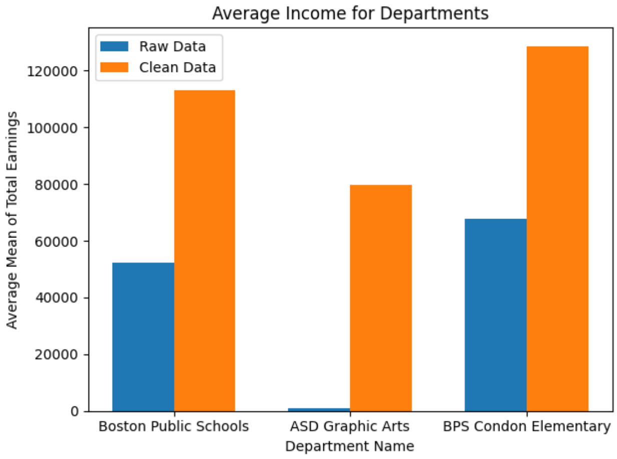

Figure 1: Comparison of average mean of raw data vs clean data

The raw data had many instances where TOTAL EARNINGS was either 0 or a negative number, which is not possible. Figure 1 clearly shows how these incorrect instances were giving a wrong average mean of Total Earnings attribute for the three departments. The ASD Graphic Arts department has an almost 0 average mean which can suggest that the employees in that department work for free.

The zero classifier and random forest classifier was used for three data sets. Firstly on un-normalized data, then on standard normalized data, and finally on sigmoidal standardized data. Table 1 shows the resulting accuracies for the described data set, when provided as input for data mining classification methods. 75% of the data from every data set is used to train the classification model and the remaining 25% is used as the test data.

Table 1: Comparing resulting accuracies of different classifiers

Data set	Zero classifier	Random Forest
Un-normalized	39.45	83.83
Standard normalized	39.45	83.83
Sigmoidal normalized	39.45	76.3

5. LESSONS LEARNED

Before starting any process on the data, it needs to be understood clearly. Metadata is important and domain knowledge about the datasets information is important. It also helps realize the need of domain experts. Once, data is understood, it is important to realize which data cleaning techniques would provide better results. If there is less data, then deleting missing value rows would only reduce data which would further be insufficient for data mining tasks. There should not be redundant data present in the dataset. It is important to select an appropriate database management system to store data easily and have an effective extraction of data, when required. If the aim for the kind of information gain is known, data can be prepared in a certain way without losing the important attributes.

Normalization is important because datasets generally have different sources and so the units and attributes can have different units. If normalization is not performed, then there are high chances of faulty results.

Data mining tasks should have a baseline model to compare results to and understand the performance and rank various data mining techniques accordingly.

6. CONCLUSION AND FUTURE WORK

Raw Data was understood and processed through different data cleaning methods. Data was loaded into a relational database (PostgreSQL) and prepared using different steps for future extraction during data mining tasks. Normalization and target class creation were carried out to facilitate the data mining process. Various comparisons were done in order to understand data, gain information and evaluate effectiveness of the applied data cleaning, preparation and mining techniques. It was also observed that the accuracy of the unnormalized data was almost as good as normalized data and even better with respect to sigmoidal normalization. This implies that the data set prepared was mostly homogeneous and the different units and attributes were optimally merged. As per the findings of the average mean of the income of three departments, the raw data consists of many non positive values that brought down the average mean of the departments and produced incorrect results. This implies that data preparation and cleaning helped regain the true average mean of the attributes. It increased the efficiency of the input to the classification method, and as can be seen clearly garbage input would result to a garbage output but proper input can result in a highly accurate output.

For most of the data mining projects, a structural process called CRISP-DM(Cross Industry standard process for data mining) is implemented. This process is used to predict and create the timeline for a project. CRISP-DM also helps to identify the importance of the project and check the efficiency. Lot of data scientist are unaware of this process, but it can help them to solve a lot of problem even before the problem occurs. This process can be implemented here to understand a different way to look at the data and then accordingly explore other data mining tasks. While merging the two datasets from different sources, only attributes that were common were chosen and the attributes that were different were dropped which resulted in data loss. Future work can also concentrate on using all the attributes that are in the datasets and then clean and prepare data, find out best features and then carry out appropriate data mining tasks.

7. REFERENCES

[1] Analyze Boston. Employee Earnings Report. `https://data.boston.gov/dataset/employee-earnings-report`, 2019. [Online; accessed 05-March-2019].

[2] NY Open Data. Salary Information for State Authorities. `https://data.ny.gov/Transparency/Salary-Information-for-State-Authorities/unag-2p27`, 2019. [Online; accessed 04-October-2019].