Machine Learning, Data analysis and visualization Task.

Part 1:

1. Identify 2 use case Datasets that solve real-world problems.

Below are the datasets that are interesting to me and the real-world problems they solve are described as the objective to be accomplished by the dataset.

1. Dataset A: Loan Prediction analysis.

Source Kaggle: <https://www.kaggle.com/code/harshavarshney/loan-prediction-analysis>

Field: Finance

Objective: Examine what are the Key Indicators in determining whether a person will likely have/ take a loan.

1. Dataset B: KMeans Clustering in Customer Segmentation

Source Kaggle: <https://www.kaggle.com/code/vjchoudhary7/kmeans-clustering-in-customer-segmentation/notebook>

Field: Business

Objective: Identify a Specific Population in a Range of Customers for Targeted Advertisements.

1. Dataset A – uses a supervised learning technique, Random Forest for predicting if a person will take a loan if they have some set of features that have been previously seen in the dataset. Random Forest uses a consensus-based way of making decisions by taking the majority results provided when several decision trees are trained on the same dataset. Therefore, RF is different from the decision trees since we use several decision trees and choose the results provided by the majority tress.

*The Fields in Dataset A are:*

Loan\_ID, Gender, Married, Dependents, Education, Self\_Employed, ApplicantIncome, CoapplicantIncome, LoanAmount, Loan\_Amount\_Term, Credit\_History, Property\_Area, Loan\_Status.

Dataset B – uses an unsupervised learning technique, KMeans Clustering to identify clustering in the data and interpret any useful information from the clusters or groups.

*The Fields in Dataset B are:*

CustomerID, Gender, Age, Annual Income (k$), Spending Score (1-100)

1. Preprocessing

Dataset A:

1. Check for Null Value–

fill in the missing values for numerical terms with the help of mean.

fill in the missing values for categorical terms with the help of mode

1. Drop unnecessary columns - ApplicantIncome, CoapplicantIncome, LoanAmount, Loan\_Amount\_Term, Total\_Income, Loan\_ID, CoapplicantIncomeLog

Dataset B:

1. Check for Null Value – There was none.
2. Feature Selection - Choose only 2 features from the dataset, Annual income, and Spending Score.
3. Data visualization, analysis, and results for Dataset A.
4. Seaborn Count plot – for plotting the categorical data. It shows the number of occurrences (counts), of an observation present in a categorical variable. Uses bar charts for the visual representation.

This is plotted for Gender, Married, Dependents, Education, Self\_Employed, Property\_Area, Loan\_Status.

1. Seaborn distplot - for plotting the numerical data. It shows a histogram with a distribution line on it. It plots a univariate distribution of observations.

This is plotted for ApplicantIncome, CoapplicantIncome, LoanAmount, Loan\_Amount\_Term, Credit\_History, LoanAmount (the log of LoanAmount),

Total\_Income (New Field created by summing ApplicantIncome, CoapplicantIncome).

1. Seaborn Heatmap – for showing the correlation in the data attributes(fields). A heatmap is a rectangular plot that represents data as a color-encoded matrix. It takes a 2D dataset and the data relationships are correlated into a ndarray.

A screenshot of a computer

Description automatically generated

In the scale, the color in 1.0 represents a higher influence between the coupled variables, and the color in 0.0 represents a small influence. Positive values show direct linear correlation, negative values show inverse correlation. Examination between the features and price:

1. Label encoding – for training the textual data, it is encoded and represented using 0 and 1. After label encoding, the data frame is now in numeric format and easily trained in the Random Forest model.
2. The data is then split into training and testing sets. Using the TRAIN-TEST SPLIT, the training set(X) is 0.75 and the test set(y) is 0.25.
3. A function, *classify(model,x,y)*: is used to train and validate the dataset using cross-validation.
4. Then, the model is created and passed to the function *classify*.

Different models are created, and the results of this function are compared:

For *LinearRegression Model - Accuracy is 77.27* and *Cross-validation is, 80.94.*

For Decision Tree Model - *Accuracy is 66.23* and *Cross-validation is, 71.66.*

For Random Forest Model - *Accuracy is 76.62* and *Cross-validation is, 79.32.*

For Extra Trees Model - *Accuracy is 73.38* and *Cross-validation is, 76.71.*

The optimal model should be Linear Regression and Random Forest.

1. Hyperparameter Tuning is done on the Random Forest Classifier to improve the accuracy. And new results are: *Accuracy is 75.97* and *Cross-validation is, 80.78.*

Actually, the results show the accuracy reduced slightly for training while the cross-validation increased slightly for testing.

1. A confusion matrix for the Random Forest classifier is then plotted the show the accuracy and cross-validation results.



The confusion matrix shows that:

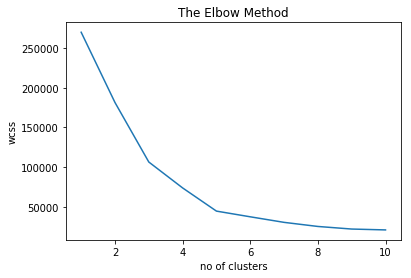
in 23 instances, the truth was 0 and the model predicted 0,

in 31 instances, the truth was 0 and the model predicted 1,

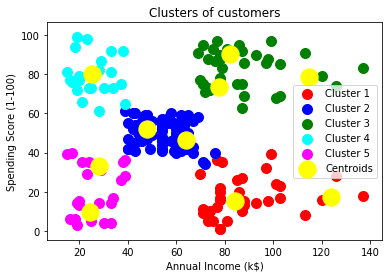
in 6 instances, the truth was 1 and the model predicted 0, and

in 94 instances, the truth was 1 and the model predicted 1.

1. Data visualization for Dataset B.
2. Using pyplot – for visualizing the ELBOW method to get the optimal value of K. The elbow points to 5, which means that k=5. This means there will be 5 centroids and 5 clusters in the dataset.



1. Visualize the clusters in the data



Model Interpretation

Cluster 1 (Red Color) -> earning high but spending less

Cluster 2 (Blue Colr) -> average in terms of earning and spending

Cluster 3 (Green Color) -> earning high and also spending high [TARGET SET]

Cluster 4 (cyan Color) -> earning less but spending more

Cluster 5 (magenta Color) -> Earning less, spending less

Cluster 3 can be put into some alerting system for targeted daily emails or notifications, whereas others we can be set like once in a week or once in a month.

Part 2

1. Data Preprocessing
2. Downloaded the dataset from <https://www.kaggle.com/andrewmvd/data-scientist-jobs>

and saved it to my working directory.

Then, import the necessary libraries for data cleaning, manipulation, and analysis.

Next is to import the dataset to the Jupyter notebook and store it in the data frame called *mydata*.

A screenshot of a computer

Description automatically generated

1. Data exploration is the next step.

mydata.head(5) – shows the first five entries in the dataframe.

mydata.shape – shows the number of rows and columns in the data frame. In this case, the data frame has 3909 rows and 17 columns(fields).

A screenshot of a computer

Description automatically generated

1. Identifying the unique values in the data is useful in determining which fields will not contribute meaningful insight in the patterns in the data. For example, fields *Unamed 0* and *index*, have unique entries in every row which means they will not add insight into patterns and relationships in the data.

By using visual examination and human judgement, we determine the fields which will be useful in the analysis. These useful fields are stored in a new data frame called *c\_data*.

A screenshot of a computer

Description automatically generated

1. Renaming the columns

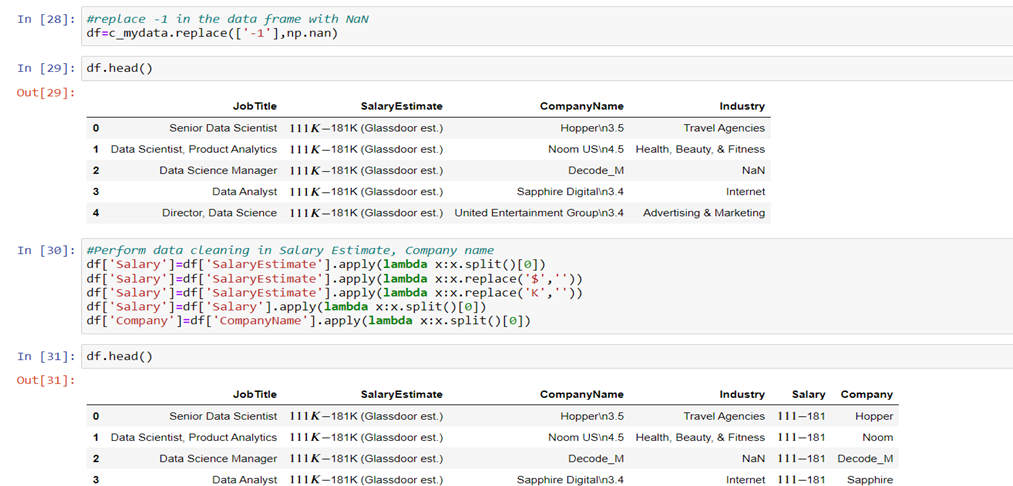
A screenshot of a computer

Description automatically generated

1. The data contains *-1* values, such as in industry field. Replace these with NaN values and handle them as NaN. Also, perform data cleaning *SalaryEstimate* and *CompanyName.*

A screenshot of a computer

Description automatically generated



The Cleaned data from *SalaryEstimate* and *CompanyName* is stored in the new *Salary* and *Company* field. Next is to drop the old columns *SalaryEstimate* and *CompanyName*.

A screenshot of a computer

Description automatically generated

1. Visualization techniques to aid in decision making.

Plot visualizations using pyplot bar plot.

1. Visualize top 20 companies with highest number of jobs in Data Science

A screenshot of a computer

Description automatically generated

Chart, histogram

Description automatically generated

1. Visualize top 20 Popular Jobs

Graphical user interface, text, application

Description automatically generated

Chart, histogram

Description automatically generated

1. Visualize the highest paying jobs according to salary

Graphical user interface, application

Description automatically generated

Graphical user interface, text

Description automatically generated

Chart, bar chart

Description automatically generated

1. Visualize the highest paying Companies

Chart, bar chart, histogram

Description automatically generated