Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Information Technology,

Pune-48 (An Autonomous Institute affiliated to Savitribai Phule Pune University)

Department of Computer Science and Engineering (Artificial Intelligence)

LAB SUBMISSION

Data Science and Machine Learning

CAUA22201

Submitted by:
Sanket Bhausaheb Devmunde,
PRN: 22210347,
Roll Number: 281016

Second Year
Semester II Academic Year 2023-24

Assignment: 1

Aim: To perform the following operations using R/Python on suitable data sets:

a) read data from different formats (like .csv, .xls)
b) indexing and selecting data, sort data,
c) describe attributes of data, checking data types of each column,
d) counting unique values of data, format of each column, converting variable data type (e.g. from long to short, vice versa),
e) identifying missing values and fill in the missing values

Theory:

1. Reading Data from Different Formats:

   Both R and Python offer libraries/packages to read data from various formats such as CSV, Excel, JSON, SQL databases, etc. In R, we can use functions like `read.csv()`, `read.table()`, or `readxl::read_excel()` for Excel files. In Python, we can use libraries like Pandas (`pd.read_csv()`, `pd.read_excel()`), NumPy (`np.loadtxt()`), or openpyxl (`openpyxl.load_workbook()`) for Excel files.

2. Indexing and Selecting Data, Sorting Data:

   Indexing and selecting specific data from datasets allow you to focus on relevant information for analysis. In R, we can use indexing `[rows, columns]` or functions like `subset()` to filter data. Sorting can be done using `order()` or `dplyr` functions. In Python, Pandas offer powerful indexing and selection capabilities using `.loc[]`, `.iloc[]`, and sorting using `.sort_values()`.

3. Describing Attributes of Data, Checking Data Types:

   Understanding the structure of data is crucial. Descriptive statistics summarize the main characteristics of a dataset. R provides functions like `summary()`, `str()`, `class()`, and `sapply()` to describe data attributes and check data types. In Python, Pandas offer methods like `describe()`, `info()`, `dtypes`, and `apply()` to achieve similar tasks.

4. Counting Unique Values, Converting Variable Data Types:

   Identifying unique values helps in understanding categorical variables, while converting data types is useful for compatibility and analysis. In R, you can use `table()`, `unique()`, `as.factor()`, `as.numeric()`, etc., to count unique values and convert data types. In Python Pandas, you can use `value_counts()`, `unique()`, and `astype()` methods to achieve the same.

5. Identifying and Filling Missing Values:

   Missing values can affect analysis results, so it is crucial to identify and handle them appropriately. Both R and Python provide functions to detect missing values (`is.na()` in R, `.isnull()` in Python). To fill missing values, R uses functions like `na.omit()`,

`complete.cases()`, or `zoo::na.locf()`. In Python Pandas, you can use `.fillna()`, `.dropna()`, or `.interpolate()`.

These are foundational concepts for data manipulation and preprocessing in both R and Python. Understanding these operations is essential for effective data analysis and modelling. Practice and familiarity with relevant libraries/packages in each language enhances proficiency in data handling tasks.

Conclusion:

In this assignment, we were able to perform various operations on data like reading files, sorting data, finding null values and missing values, removing null values, etc.

Assignment: 2

Aim: To perform the following operations using R/Python on the data sets:

a) Compute and display summary statistics for each feature available in the dataset. (e.g. minimum value, maximum value, mean, range, standard deviation, variance and percentiles)

b) Data Visualization-Create a histogram for each feature in the dataset to illustrate the feature distributions.

c) Data cleaning, Data integration, Data transformation, Data model building (e.g. Classification)

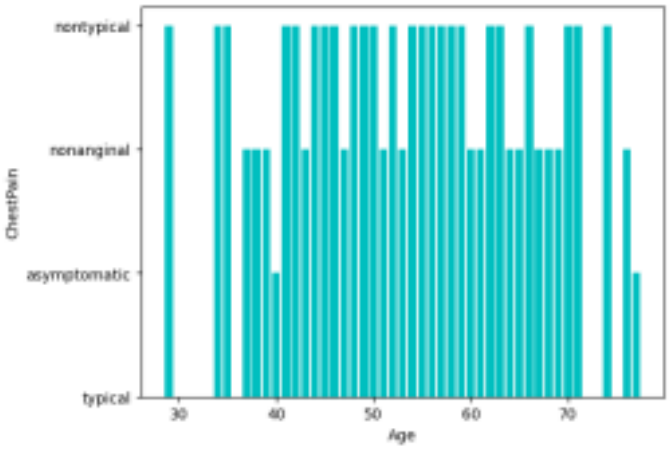
Theory:

Summary statistics provide a concise overview of the key characteristics of the dataset. These statistics include measures such as minimum and maximum values, mean, range, standard deviation, variance, and percentiles. The minimum and maximum values represent the smallest and largest values in the dataset, respectively. The mean is the average value of the dataset, providing a measure of central tendency. The range is the difference between the maximum and minimum values, indicating the spread of the data. Standard deviation and variance quantify the dispersion of data points around the mean, with higher values indicating greater variability. Percentiles divide the dataset into 100 equal parts, allowing us to understand the distribution of data across different percentiles.

Data visualization through histograms offers insights into the distribution of individual features within the dataset. Histograms represent the frequency distribution of numerical data by dividing the data into bins or intervals and plotting the frequency of observations within each bin. By visualizing the distribution of features, we can identify patterns, skewness, and outliers within the data. Histograms provide a visual representation of the data's shape, central tendency, and spread, allowing for a better understanding of its underlying characteristics.

Data cleaning, integration, transformation, and model building are essential steps in the data analysis process. Data cleaning involves identifying and handling missing values, outliers, and inconsistencies to ensure data quality and integrity. Data integration combines multiple datasets into a single, unified dataset, facilitating comprehensive analysis. Data transformation includes preprocessing steps such as feature scaling, encoding categorical variables, and creating new features to prepare the data for modelling. Finally, data model building involves selecting an appropriate machine learning algorithm, splitting the data into training and testing sets, training the model on the training data, and evaluating its performance on the testing data. These steps collectively enable the development of robust and accurate predictive models for classification tasks, aiding in decision-making and problem-solving in various domains.

Results:



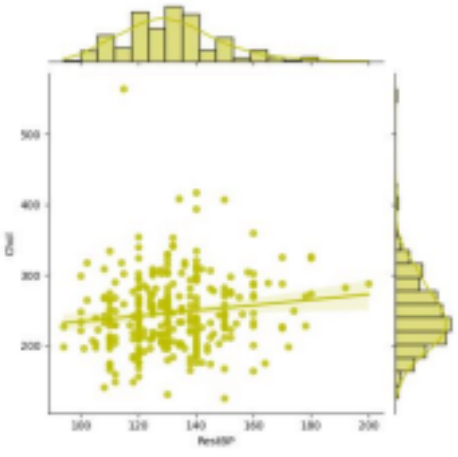
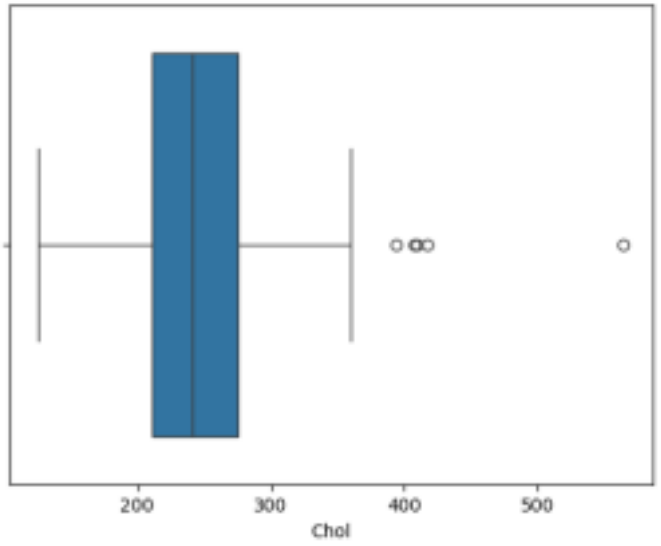


Fig. Bar Chart Fig. Jointplot



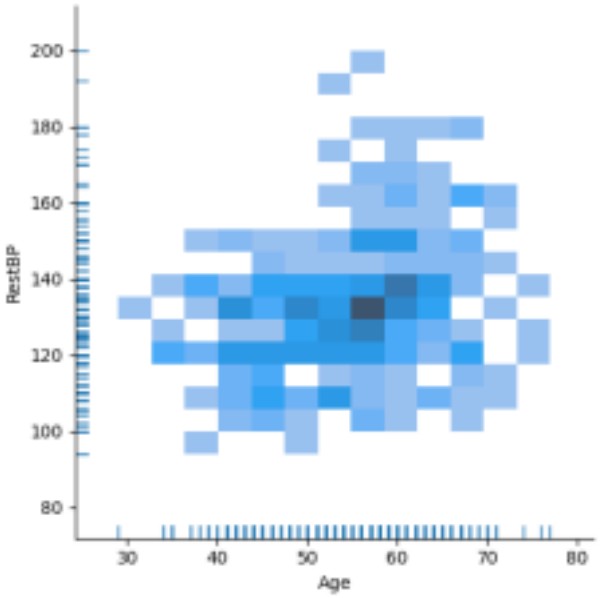


Fig. Boxplot Fig. Displot

	Unnamed: 0	Age	Sex	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpeak	Slope	Ca
count	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	299.000000
mean	152.000000	54.438944	0.679868	131.689769	246.693069	0.148515	0.990099	149.607261	0.326733	1.039604	1.600660	0.672241
std	87.612784	9.038662	0.467299	17.599748	51.776918	0.356198	0.994971	22.875003	0.469794	1.161075	0.616226	0.937438
min	1.000000	29.000000	0.000000	94.000000	126.000000	0.000000	0.000000	71.000000	0.000000	0.000000	1.000000	0.000000
25%	76.500000	48.000000	0.000000	120.000000	211.000000	0.000000	0.000000	133.500000	0.000000	0.000000	1.000000	0.000000
50%	152.000000	56.000000	1.000000	130.000000	241.000000	0.000000	1.000000	153.000000	0.000000	0.800000	2.000000	0.000000
75%	227.500000	61.000000	1.000000	140.000000	275.000000	0.000000	2.000000	166.000000	1.000000	1.600000	2.000000	1.000000
max	303.000000	77.000000	1.000000	200.000000	564.000000	1.000000	2.000000	202.000000	1.000000	6.200000	3.000000	3.000000

Fig. Brief Description about Dataset

Conclusion:

In this assignment, we were able to analyse the dataset and visualize it by using various charts and plots.

Assignment: 3

Aim: To apply appropriate ML algorithm on a dataset collected in a cosmetics shop showing details of customers to predict customer response for special offers.

Theory:

To predict customer responses for special offers in a cosmetics shop using machine learning (ML), several steps need to be followed.

1. Data collection from the cosmetics shop is crucial, encompassing various customer details and their responses to special offers. We have taken the dataset from Kaggle. Here is the dataset: "https://www.kaggle.com/datasets/kingabzpro/cosmetics-datasets". This dataset is diverse and includes 1472 rows and 11 columns.

2. After data collection, preprocessing is essential to clean and handle missing values, encode categorical variables, and scale numerical features as necessary. Feature engineering comes next, where new features are created or existing ones transformed to enhance predictive power. For instance, deriving features such as 'Price' or 'Rank' can provide valuable insights.

3. Subsequently, the dataset is split into training and testing sets for model evaluation. Then, an appropriate ML algorithm, such as logistic regression, decision trees, random forests, or gradient boosting, is chosen based on the nature of the problem (classification) and dataset characteristics. The selected algorithm is trained on the training data and evaluated using the testing data. Evaluation metrics like accuracy, precision, recall, F1-score, and ROC-AUC are commonly used to assess model performance. Once satisfied with the model's performance, it can be deployed in a production environment to predict customer responses in real time. Continuous monitoring and updating of the model are crucial to ensure its effectiveness in predicting customer responses accurately over time.
Here, I have used Random Forest Classifier as algorithm to build model.

Random Forest Classifier:

Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

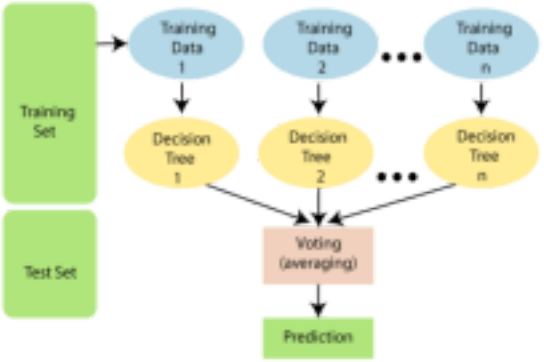


Fig. Random Forest Classifier

Advantages:

1. Random Forest is capable of performing both Classification and Regression tasks.
2. It is capable of handling large datasets with high dimensionality.
3. It enhances the accuracy of the model and prevents the overfitting issue.

Limitations:

Although random forest can be used for both classification and regression tasks, it is not more suitable for Regression tasks.

Applications:

There are mainly four sectors where Random Forest Algorithm is mostly used:

1. **Banking:** Banking sector mostly uses this algorithm for the identification of loan risk. 2. **Medicine:** With the help of this algorithm, disease trends and risks of the disease can be identified. 3. **Land Use:** We can identify the areas of similar land use by this algorithm.
4. **Marketing:** Marketing trends can be identified using this algorithm.

Results:

These are the results that are obtained:

```
Accuracy:  35.59%
Confusion Matrix:
[[32  1  7  8  3  4]
 [ 3 12  1  9  6  8]
 [16  8 20  6 11  9]
 [12  3  6 25  2  6]
 [10  1  3 10  6  4]
 [ 7  5  6 13  2 10]]
Classification Report:
              precision    recall  f1-score   support

    Cleanser       0.40      0.58      0.47        55
   Eye cream       0.40      0.31      0.35        39
   Face Mask       0.47      0.29      0.35        70
 Moisturizer       0.35      0.46      0.40        54
 Sun protect       0.20      0.18      0.19        34
   Treatment       0.24      0.23      0.24        43

    accuracy                           0.36       295
   macro avg       0.34      0.34      0.33       295
weighted avg       0.36      0.36      0.35       295
```

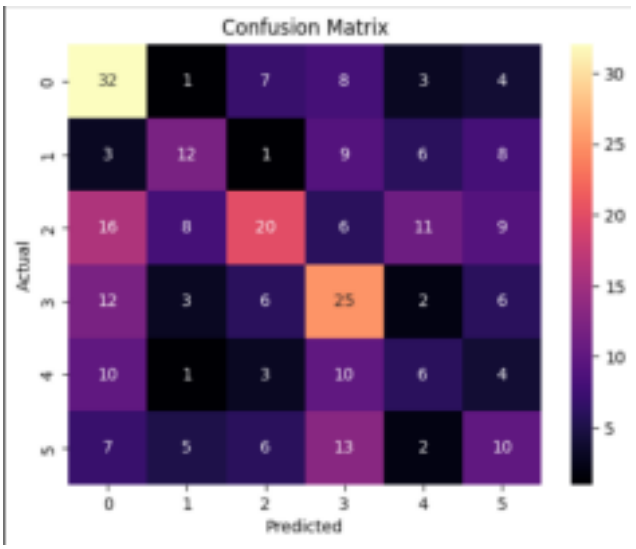


Fig. Classification Report Fig. Confusion Matrix

Conclusion:

In this assignment, we were able to implement Random Forest Classifier algorithm on a dataset collected in a cosmetics shop showing details of customers to predict customer response for special offers.

Assignment: 4

Aim: To write a program to do following:

We have given a collection of 8 points. P1=[0.1,0.6] P2=[0.15,0.71] P3=[0.08,0.9] P4=[0.16, 0.85] P5=[0.2,0.3] P6=[0.25,0.5] P7=[0.24,0.1] P8=[0.3,0.2]. Perform the k-mean clustering with initial centroids as m1=P1=Cluster#1=C1 and m2=P8=cluster#2=C2.

Theory:

K-means clustering:

K-means clustering is a popular unsupervised machine learning algorithm used for partitioning data into distinct groups or clusters based on similarities among data points. In the context of the given collection of points, K-means clustering aims to group the points into two clusters, each represented by a centroid. The algorithm starts by randomly initializing the centroids, which serve as the starting points for the clustering process. In this case, the initial centroids are chosen as P1=[0.1,0.6] for Cluster 1 (C1) and P8=[0.3,0.2] for Cluster 2 (C2). Here is the dataset that we have used:
"https://www.kaggle.com/datasets/vjchoudhary7/customer-segmentation-tutorial-in-python"

Next, the algorithm iteratively assigns each data point to the nearest centroid and then updates the centroids based on the mean of the data points assigned to each cluster. This process continues until convergence, where the centroids no longer change significantly or a specified number of iterations is reached.

During each iteration, the distance between each data point and the centroids is calculated using a distance metric, commonly the Euclidean distance. Data points are then assigned to the cluster whose centroid is closest to them. After all data points are assigned, the centroids are recalculated as the mean of all data points assigned to each cluster.

In summary, the K-means clustering algorithm partitions the data into clusters by minimizing the within cluster sum of squares, where each data point belongs to the cluster with the nearest centroid. This process repeats until convergence, resulting in well-defined clusters that can help in understanding the underlying structure of the data and making predictions or classifications based on those clusters.

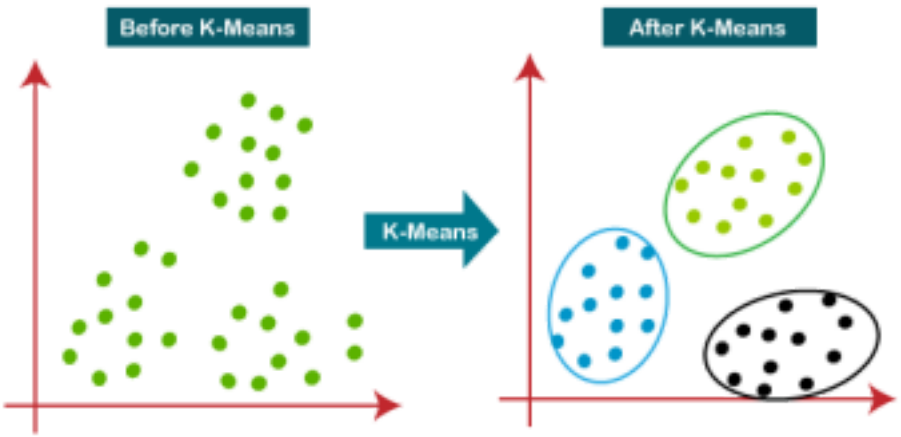


Fig. K Means Clustering

Advantages:

1. It is very easy to understand and implement.
2. If we have large number of variables then, K-means would be faster than Hierarchical clustering.
3. On re-computation of centroids, an instance can change the cluster.
4. Tighter clusters are formed with K-means as compared to Hierarchical clustering.

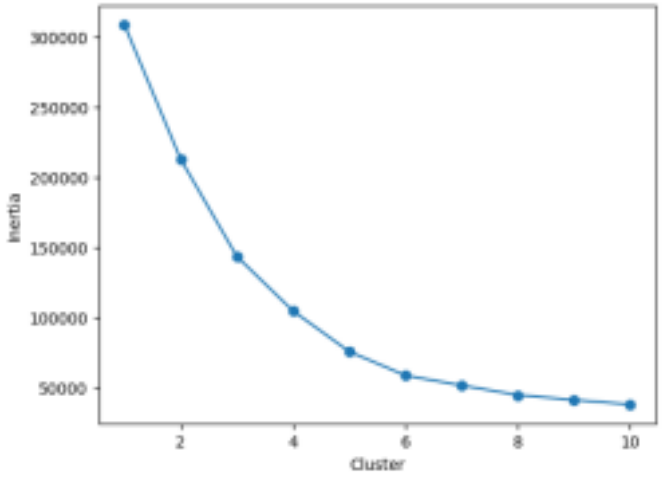
Limitations:

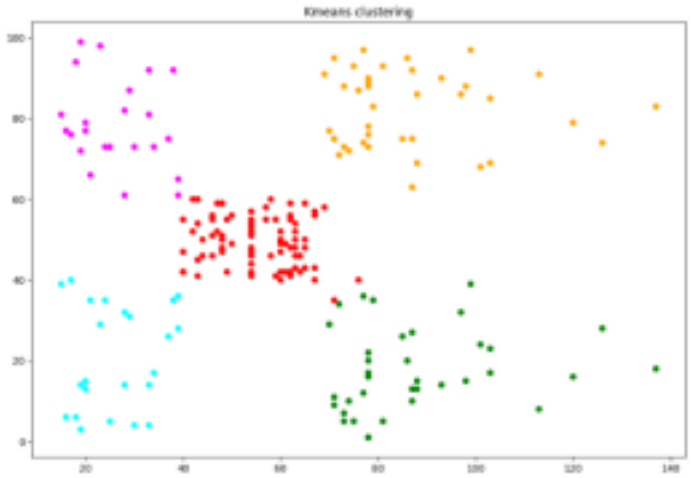
1. It is a bit difficult to predict the number of clusters i.e. the value of k.
2. Output is strongly impacted by initial inputs like number of clusters (value of k).
3. Order of data will have strong impact on the final output.
4. It is very sensitive to rescaling. If we will rescale our data by means of normalization or standardization, then the output will completely change the final output.
5. It is not good in doing clustering job if the clusters have a complicated geometric shape.

Applications:

1. Market segmentation
2. Document Clustering
3. Image segmentation
4. Image compression

Results:





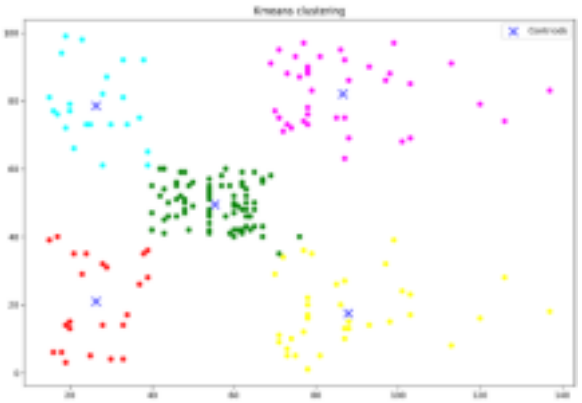


Fig. Elbow Curve Fig. Clustering

Fig. Clustering with Centroid

Conclusion:

In this assignment, we were able to implement K Means Clustering algorithm to classify the data into clusters.

Assignment: 5

Aim: To visualize the data using R/Python by plotting the graphs for assignment no. 1 and 2. Consider a suitable data set.

a) Use Scatter plot, bar plot, Box plot and Histogram

OR

b) Perform the data visualization operations using Tableau for the given dataset.

Theory:

To visualize the car crashes dataset using R or Python, we first need to load the dataset into our environment. Once loaded, we can use various plotting libraries such as matplotlib or seaborn in Python to create graphical representations of the data.

In Python, we can use matplotlib and seaborn libraries to create similar plots. With matplotlib, we can create scatter plots, histograms, and other types of plots to visualize the data. Seaborn provides high-level functions to create more sophisticated plots with less code, such as pair plots to visualize relationships between multiple variables simultaneously.

Overall, visualizing the car crashes dataset allows us to gain insights into patterns, trends, and relationships within the data, which can help in understanding the factors contributing to crashes and informing decision making processes aimed at improving road safety. Additionally, visualizations provide a more intuitive way to communicate findings and results to stakeholders and decision-makers.

Here is the dataset that we have used:
"https://drive.google.com/file/d/10z6R86pi-GGU4CWzQNJGqOAoDsxOW8db/view

" Matplotlib.pyplot:

Matplotlib.pyplot is a state-based interface to matplotlib. It provides an implicit, MATLAB-like, way of plotting. It also opens figures on your screen, and acts as the figure GUI manager. Pyplot is mainly intended for interactive plots and simple cases of programmatic plot generation. It is imported as:

For Python Environment: pip install matplotlib

For Anaconda Environment: conda install matplotlib

For Google Colab: import matplotlib.pyplot as plt

Seaborn:

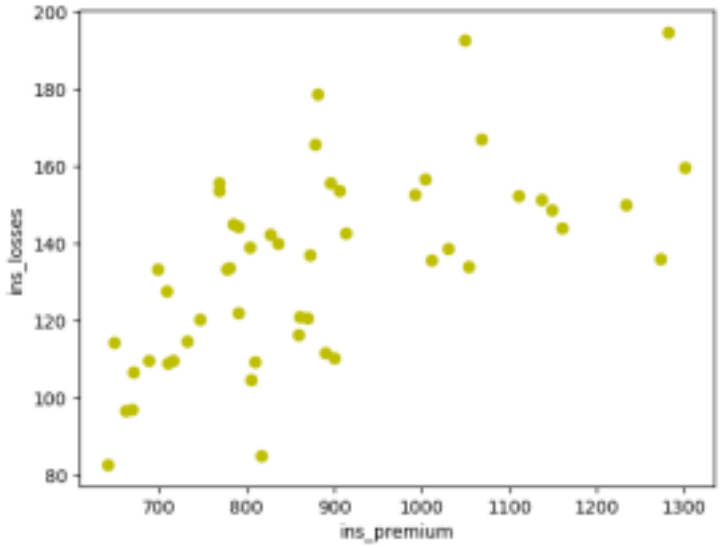
Seaborn is an amazing visualization library for statistical graphics plotting in Python. It provides beautiful default styles and colour palettes to make statistical plots more attractive. It is built on top matplotlib library and is also closely integrated with the data structures from pandas. Seaborn aims to make visualization the central part of exploring and understanding data. It provides dataset-oriented APIs so that we can switch between different visual representations for the same variables for a better understanding of the dataset.

For Python Environment: pip install seaborn

For Anaconda Environment: conda install seaborn

For Google Colab: import seaborn as sns

Results:



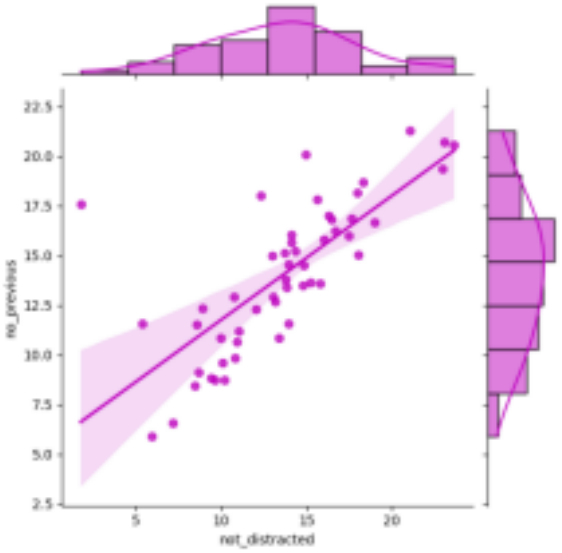
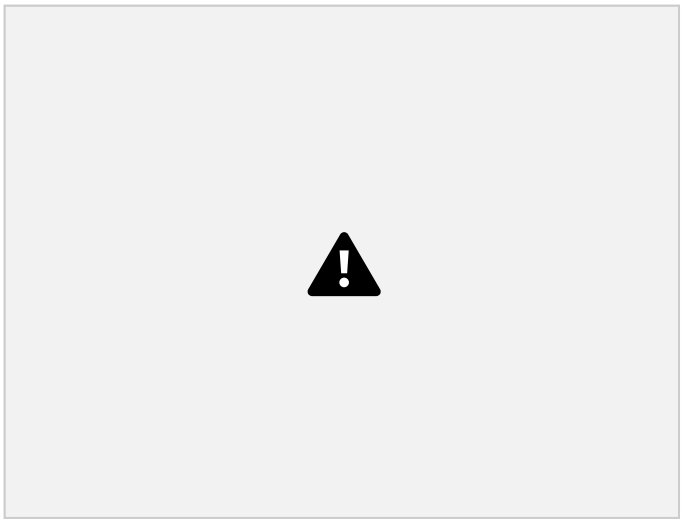


Fig. Scatter Plot Fig. Jointplot



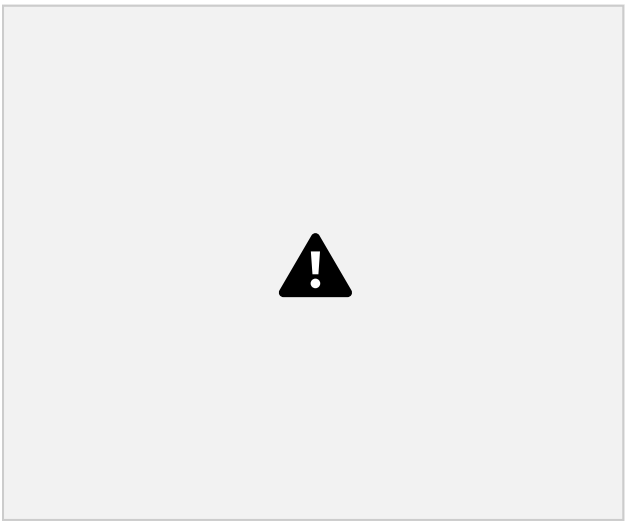


Fig. Bar Chart Fig. Boxplot

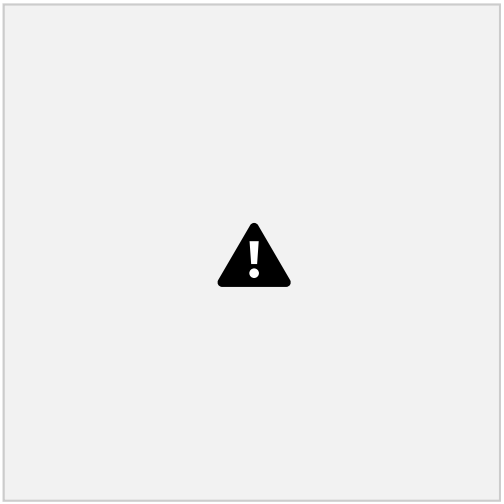


Fig. Displot

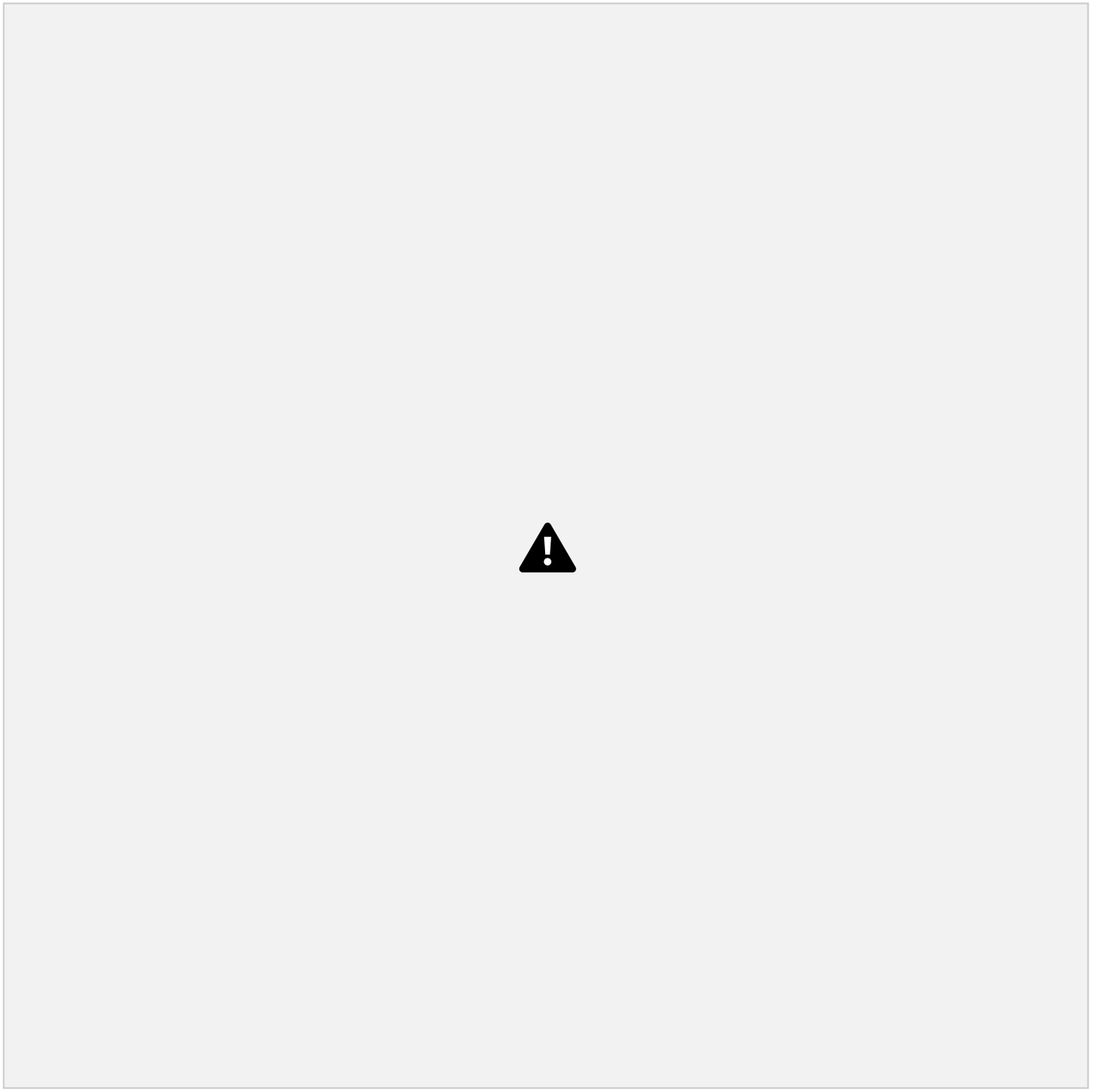


Fig. Pairplot

Conclusion:
In this assignment, we were able to visualise various charts and plots using most common libraries in Python- Matplotlib.pyplot and Seaborn.

Assignment: 6

Aim: To perform Regression technique on a dataset and

a) Apply Linear Regression using a suitable library function and predict the Month-wise temperature.

b) Assess the performance of regression models using MSE, MAE and R-Square metrics

c) Visualize a simple regression model.

Theory:

Linear Regression: Linear regression is a fundamental technique used in predictive modelling to establish a relationship between a dependent variable (target) and one or more independent variables (features). In the context of temperature prediction from the provided dataset, linear regression can be applied to understand how changes in independent variables (such as month) affect the dependent variable (temperature).

To apply linear regression, a suitable library function can be used, such as scikit-learn in Python. The dataset can be pre-processed to extract the necessary features (e.g., month) and target variable (temperature). Then, the linear regression model can be trained on the training data using the fit() function. Once trained, the model can be used to predict the temperature for each month.

The performance of the regression model can be assessed using several metrics, including Mean Squared Error (MSE), Mean Absolute Error (MAE), and R-Squared (R^2). MSE measures the average squared difference between the predicted and actual values, while MAE measures the average absolute difference. R^2 indicates the proportion of the variance in the dependent variable that is predictable from the independent variables.

Visualizing a simple regression model involves plotting the actual values against the predicted values. This can be done using scatter plots, where each point represents a data point, and the X-axis represents the actual values while the Y-axis represents the predicted values. A perfect regression model would result in all points lying on a diagonal line with a slope of 1. Visual inspection of the plot can provide insights into how well the model fits the data and whether there are any systematic errors or patterns in the predictions.

In summary, linear regression is a powerful technique for predicting temperature based on historical data. By assessing the performance of the regression model using appropriate metrics and visualizing the results, we can gain valuable insights into the accuracy and reliability of the predictions.

Here is the dataset that we have used:

“https://www.kaggle.com/datasets/venky73/temperatures-of-india”

Advantages:

1. Linear regression is a relatively simple algorithm, making it easy to understand and implement. The coefficients of the linear regression model can be interpreted as the change in the dependent variable for a one-unit change in the independent variable, providing insights into the relationships between variables.

2. Linear regression is computationally efficient and can handle large datasets effectively. It can be trained quickly on large datasets, making it suitable for real-time applications.
3. Linear regression is relatively robust to outliers compared to other machine learning algorithms. Outliers may have a smaller impact on the overall model performance.
4. Linear regression often serves as a good baseline model for comparison with more complex machine learning algorithms.
5. Linear regression is a well-established algorithm with a rich history and is widely available in various machine learning libraries and software packages.

Limitations:

1. Linear regression assumes a linear relationship between the dependent and independent variables. If the relationship is not linear, the model may not perform well.
2. Linear regression is sensitive to multicollinearity, which occurs when there is a high correlation between independent variables. Multicollinearity can inflate the variance of the coefficients and lead to unstable model predictions.
3. Linear regression assumes that the features are already in a suitable form for the model. Feature engineering may be required to transform features into a format that can be effectively used by the model.
4. Linear regression is susceptible to both overfitting and underfitting. Overfitting occurs when the model learns the training data too well and fails to generalize to unseen data. Underfitting occurs when the model is too simple to capture the underlying relationships in the data.
5. Linear regression provides limited explanatory power for complex relationships between variables. More advanced machine learning techniques may be necessary for deeper insights.

Applications:

Linear regression is used in many different fields, including finance, economics, and psychology, to understand and predict the behaviour of a particular variable. For example, in finance, linear regression might be used to understand the relationship between a company’s stock price and its earnings or to predict the future value of a currency based on its past performance.

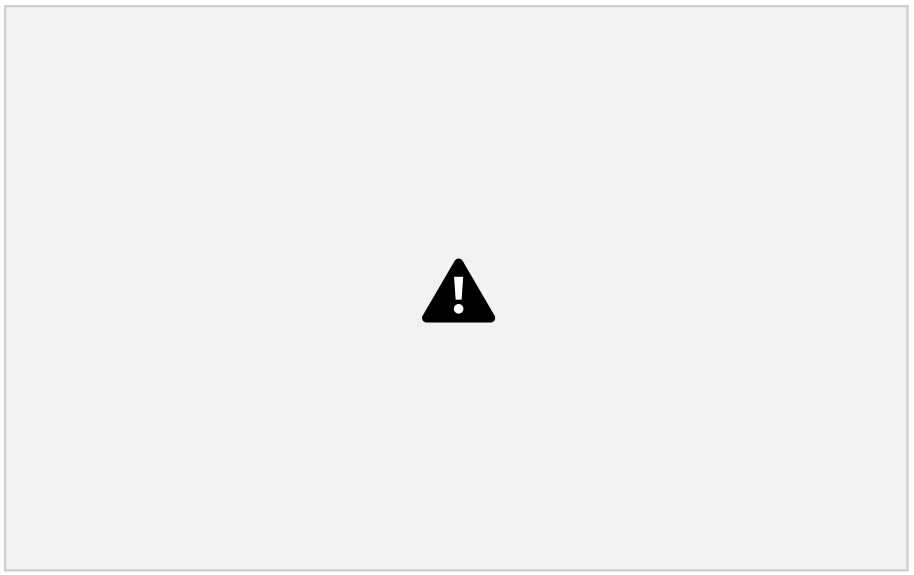


Fig. Linear Regression

Results:

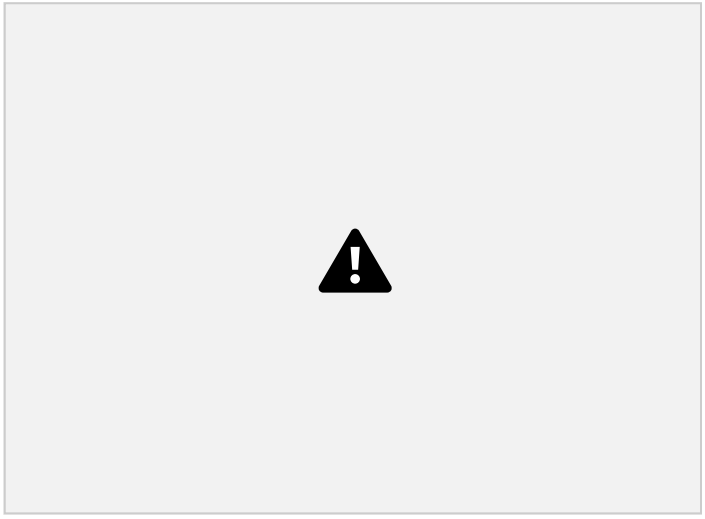


Fig. Linear Regression Fig. MAE, MSE & R-Square

Conclusion:

In this assignment, we were able to learn about Linear Regression and various errors like Mean Squared Error (MSE) and Mean Absolute Error (MAE) and R-Squared.

Assignment: 7

Aim: To implement Classification techniques for following scenario:

Every year many students give the GRE exam to get admission in foreign Universities. The data set contains GRE Scores (out of 340), TOEFL Scores (out of 120), University Rating (out of 5), Statement of Purpose strength (out of 5), Letter of Recommendation strength (out of 5), Undergraduate GPA (out of 10), Research Experience (0=no, 1=yes), Admitted (0=no, 1=yes). Admitted is the target variable.

The counselor of the firm is supposed to check whether the student will get an admission or not based on his/her GRE score and Academic Score. So to help the counselor to take appropriate decisions, build a machine learning model classifier using a Decision tree to predict whether a student will get admission or not.

a) Apply Data pre-processing (Label Encoding, Data Transformation....) techniques if necessary.

b) Perform data-preparation (Train-Test Split)

c) Apply Machine Learning Algorithm

d) Evaluate Model.

Theory:

A decision tree is one of the most powerful tools of supervised learning algorithms used for both classification and regression tasks. It builds a flowchart-like tree structure where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label. It is constructed by recursively splitting the training data into subsets based on the values of the attributes until a stopping criterion is met, such as the maximum depth of the tree or the minimum number of samples required to split a node.

During training, the Decision Tree algorithm selects the best attribute to split the data based on a metric such as entropy or Gini impurity, which measures the level of impurity or randomness in the subsets. The goal is to find the attribute that maximizes the information gain or the reduction in impurity after the split.

The decision tree operates by analysing the data set to predict its classification. It starts from the tree's root node, where the algorithm views the value of the root attribute compared to the attribute of the record in the actual data set. Based on the comparison, it proceeds to follow the branch and move to the next node. There are two popular techniques for Attribute Selection Measure (ASM), which are: Information Gain and Gini Index.

The algorithm repeats this action for every subsequent node by comparing its attribute values with those of the sub-nodes and continuing the process further. It repeats until it reaches the leaf node of the tree. The complete mechanism can be better explained through the algorithm given below.

Step-1: Begin the tree with the root node, says S, which contains the complete dataset.
Step-2: Find the best attribute in the dataset using Attribute Selection Measure (ASM).
Step-3: Divide the S into subsets that contains possible values for the best attributes.
Step-4: Generate the decision tree node, which contains the best attribute.

Step-5: Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node Classification and Regression Tree algorithm.
Here is the dataset that we have used:
"https://www.kaggle.com/mohansacharya/graduate-admissions"

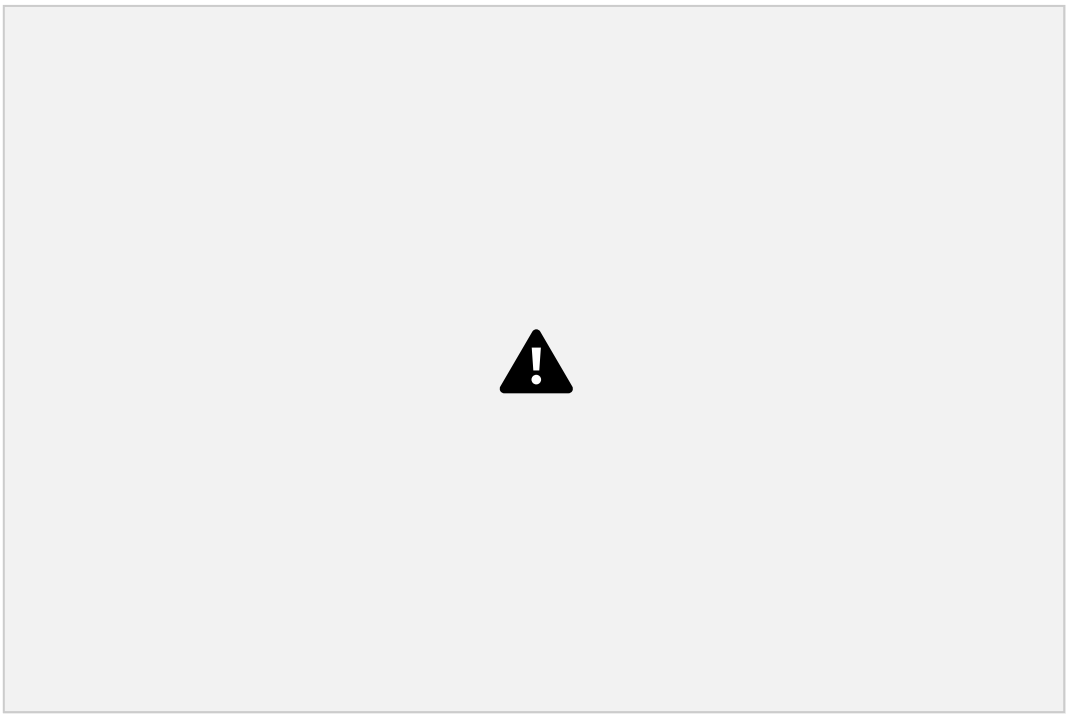


Fig. Decision Tree

Advantages:

1. It is simple to understand as it follows the same process which a human follow while making any decision in real-life.
2. It can be very useful for solving decision-related problems.
3. It helps to think about all the possible outcomes for a problem.
4. There is less requirement of data cleaning compared to other algorithms.

Limitations:

1. The decision tree contains lots of layers, which makes it complex.
2. It may have an overfitting issue, which can be resolved using the Random Forest algorithm. 3. For more class labels, the computational complexity of the decision tree may increase.

Results:

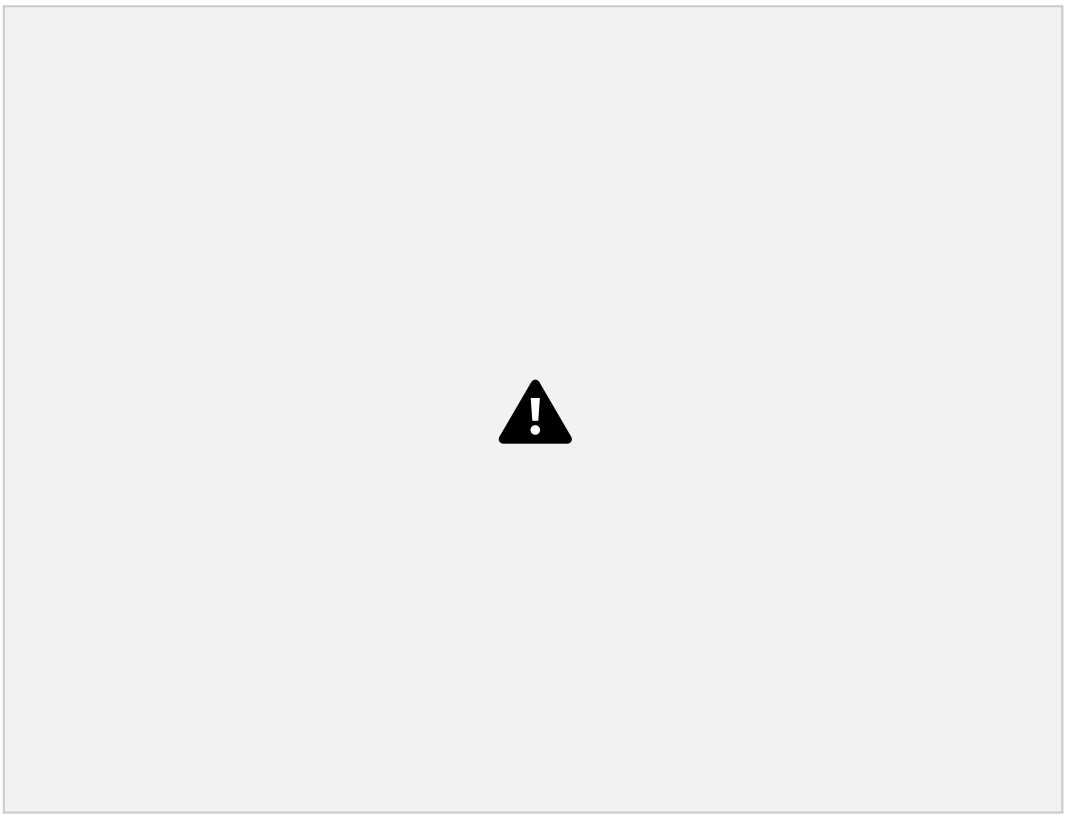


Fig. Confusion Matrix

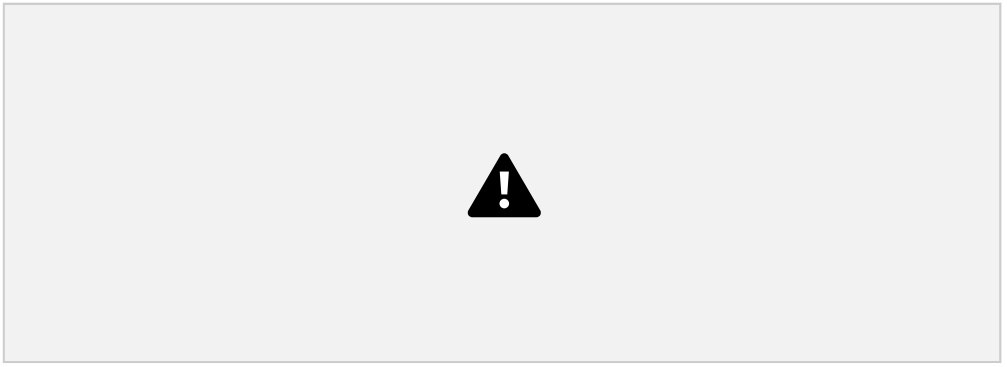


Fig. Classification Report

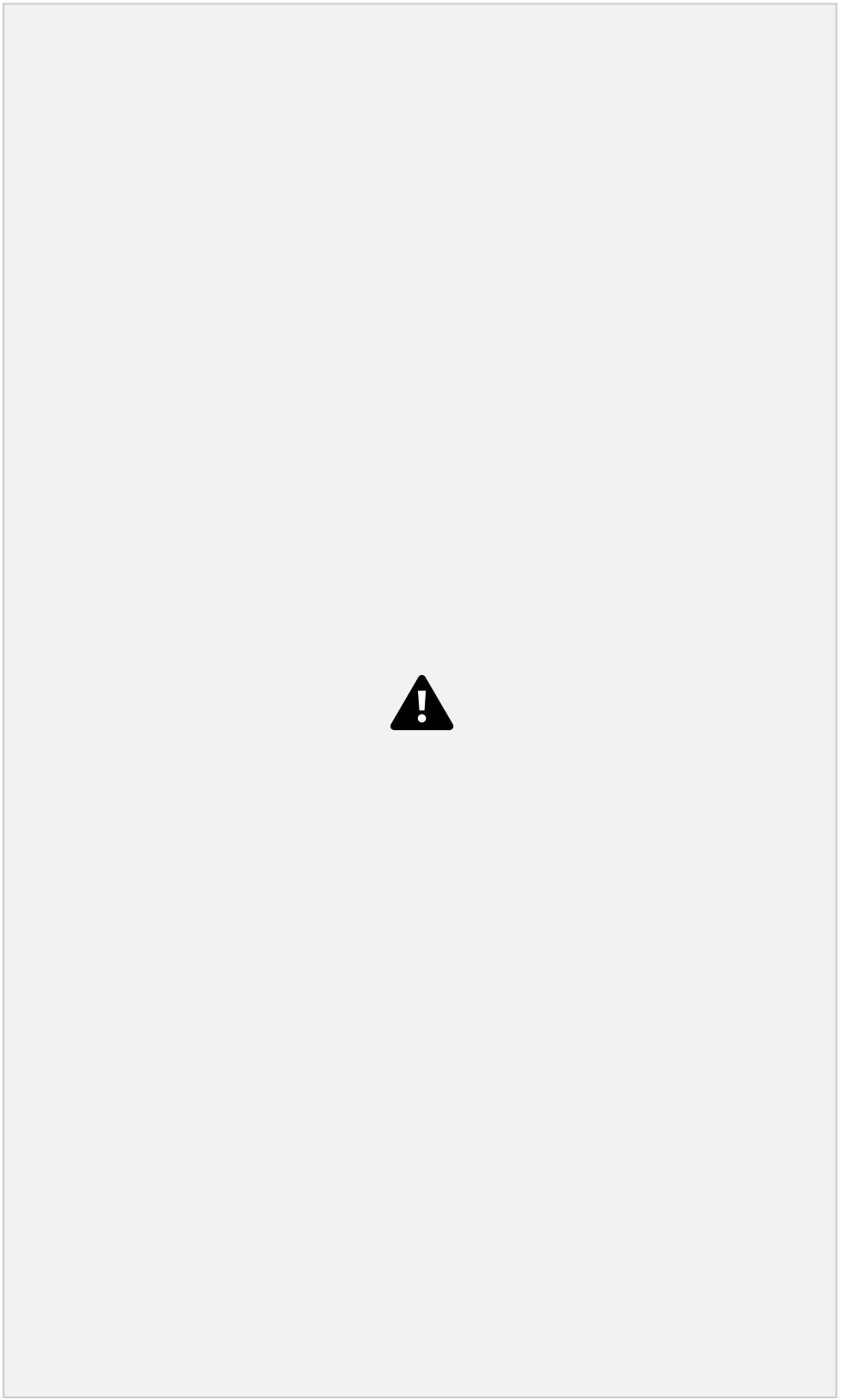


Fig. Decision Tree Model

Conclusion:

In this assignment, we were able to learn about Decision Tree Classifier and various terminologies related to it like Entropy, Information Gain and Gini Index.