Name – Binay Borun Mann

College Name – Indian Institute of Information Technology, Kalyani

Company Name – Exposys Data Labs

**1.)Abstract**

In this report I have used various types of Regression models, and have their calculated their respective regression metrics to predict the profit margin of a company , given the amount of revenue spend on the R&D, Administration and Marketing of the company. The regression model uses the profit of the company as a target dependent variable which uses the R&D, Administration, Marketing as the dependent used to predict the profit.

**2.) Table of contents**

**3.) Introduction**

**Report on Various Regression Model for Predicting Startup Profit**

**Objective:** This report aims to evaluate the performance of a Regression model applied to predict the profit of startups based on features such as R&D spending, Administration spending, and Marketing spending.

**Dataset Overview:** The dataset used for this analysis is loaded from the file "50\_Startups.csv." It contains information on various startups, including features like R&D spending, Administration spending, Marketing spending, and Profit. The target variable for prediction is "Profit."

**Data Preprocessing:** The dataset is loaded into a Pandas DataFrame, and any missing values are dropped. The features (R&D spending, Administration spending, and Marketing spending) are separated into the variable 'x,' and the target variable (Profit) is stored in 'y.' The data is then split into training and testing sets using the train\_test\_split function from scikit-learn.

**Model Selection:** A Regression model is chosen for predicting the profit of startups. Linear

Here we have used six models like the linear regression , svm regressor model , knn regressor model , Neural Network Regressor model .

**Model Training and Evaluation:** The Regressor model is trained on the training data (x\_train, y\_train) and used to make predictions on the testing data (x\_test). Various regression metrics are employed to evaluate the model's performance:

1. **Mean Squared Error (MSE):**
   * The MSE measures the average squared difference between predicted and actual values.
   * A lower MSE indicates better model performance.
2. **Mean Absolute Error (MAE):**
   * The MAE calculates the average absolute difference between predicted and actual values.
   * It provides a more interpretable metric than MSE.
3. **R-squared (accuracy):**
   * R-squared quantifies the proportion of variance in the target variable explained by the model.
   * It ranges from 0 to 1, with higher values indicating a better fit.
4. **Explained Variance:**
   * Explained Variance measures the proportion of variance in the target variable that the model explains.
5. **Mean Squared Log Error (MSLE):**
   * MSLE is used for datasets with exponential growth, such as financial data.
   * It measures the average squared logarithmic difference between predicted and actual values.

**4.)** **Existing Method**

Traditionally, If we have the Revenue amount spend on the Research and Development , Administration work and Marketing for a given company , we had to actually calculate the profit margin for the company using the formula ,

**profit margin = (net profit/Revenue)\*100.**

Also we had to calculate the Gross profit, operating profit, net profit before tax and the net profit before calculating the profit margin for a company.

1. **Gross Profit:**
   * Calculate the gross profit by subtracting the Cost of Goods Sold (COGS) from revenue.
   * Gross Profit=Revenue−COGSGross Profit=Revenue−COGS
2. **Operating Profit:**
   * Calculate the operating profit by subtracting operating expenses from gross profit.
   * Operating Profit=Gross Profit−Operating ExpensesOperating Profit=Gross Profit−Operating Expenses
3. **Net Profit Before Tax:**
   * Deduct interest expenses and taxes from operating profit.
   * Net Profit Before Tax=Operating Profit−Interest−TaxesNet Profit Before Tax=Operating Profit−Interest−Taxes
4. **Net Profit:**
   * Subtract taxes from net profit before tax to obtain the net profit.
   * Net Profit=Net Profit Before Tax−TaxesNet Profit=Net Profit Before Tax−Taxes
5. **Profit Margin:**
   * Calculate the profit margin as a percentage of revenue.
   * Profit Margin (%)=(Net ProfitRevenue)×100Profit Margin (%)=(RevenueNet Profit​)×100

It's also important to note that these calculations will depend on having specific financial data, including revenue, costs, expenses, interest, and taxes. The accuracy of the estimates also relies on the completeness and accuracy of the financial information available.

**5.) PROPOSED METHOD WITH ARCHITECTURE**

1. Problem Statement and Objectives:

Clearly define the problem you are addressing and the goals you aim to achieve. For example:

Problem Statement: Predicting the profit of startups based on various features.

Objectives: Develop a predictive model to estimate startup profit using a dataset containing information on R&D spending, administration spending, and marketing spending.

2. Data Overview:

Provide a brief overview of the dataset you are working with, including its structure, features, and target variable. For example:

Dataset: "50\_Startups.csv"

Features: R&D spending, administration spending, marketing spending

Target Variable: Profit

3. Data Preprocessing:

Outline the steps you will take to clean and prepare the data for analysis. This may include handling missing values, encoding categorical variables, and splitting the data into training and testing sets. For example:

Data Cleaning: Drop missing values from the dataset.

Feature Selection: Consider only relevant features (e.g., R&D spending, administration spending, marketing spending).

Train-Test Split: Split the data into training and testing sets.

4. Model Selection:

Describe the machine learning or statistical models you plan to use. Explain the rationale behind choosing a specific model or set of models. For example:

Model: Linear Regression

Rationale: Linear Regression is a suitable starting point for predicting a continuous target variable and allows for interpretability.

5. Model Training:

Provide details on how you plan to train the selected model. Include any parameter tuning or hyperparameter optimization steps. For example:

Training Method: Fit the Linear Regression model on the training set.

Hyperparameter Tuning: Experiment with regularization terms if needed.

6. Evaluation Metrics:

Specify the metrics you will use to evaluate the model's performance. Discuss why these metrics are relevant to your problem. For example:

Metrics: Mean Squared Error, Mean Absolute Error, R-squared

Rationale: These metrics quantify the accuracy and explanatory power of the model.

7. Proposed Architecture:

If applicable, describe any specific architecture or framework you plan to use in implementing the model. This may include details about neural network architecture, if applicable.

8. Expected Outcomes:

Provide insights into what you expect to achieve with the proposed method. For example:

Expected Outcome: A well-performing model that accurately predicts startup profit based on the provided features.

**6.) Methodology**

In the Machine learning model for predicting startup profit, the primary methodology used is a supervised machine learning approach, specifically regression analysis. Here are the key methodologies and techniques applied in the code:

**1. Supervised Learning:**

Definition: Supervised learning is a type of machine learning where the algorithm is trained on a labeled dataset, which means the model learns from input-output pairs.

Application: The dataset is split into features (input) and the target variable (output), and the model is trained to predict the target variable (Profit) based on the features (R&D spending, administration spending, and marketing spending).

**2. Linear Regression:**

Definition: Linear Regression is a statistical method used for modeling the relationship between a dependent variable (target) and one or more independent variables (features) by fitting a linear equation to the observed data.

Application: The Linear Regression model is used to predict startup profit based on R&D spending, administration spending, and marketing spending.

**3. Train-Test Split:**

Definition: The dataset is split into training and testing sets. The model is trained on the training set and evaluated on the testing set to assess its generalization performance.

Application: The train\_test\_split function from scikit-learn is used to split the dataset into training and testing sets.

**4. Mean Squared Error (MSE):**

Definition: MSE is a regression metric that measures the average squared difference between predicted and actual values. It provides an indication of the model's accuracy.

Application: MSE is used to quantify the difference between predicted and actual profits.

**5. Mean Absolute Error (MAE):**

Definition: MAE is a regression metric that measures the average absolute difference between predicted and actual values. It provides a more interpretable measure of model performance.

Application: MAE is used to quantify the accuracy of predictions.

**6. R-squared (Coefficient of Determination):**

Definition: R-squared measures the proportion of variance in the dependent variable (Profit) that is explained by the independent variables (R&D spending, administration spending, and marketing spending).

Application: R-squared is used to evaluate the explanatory power of the Linear Regression model.

**7. Explained Variance Score:**

Definition: Explained Variance Score measures the proportion of variance in the target variable that the model explains.

Application: It provides additional insight into the performance of the Linear Regression model.

**8. Mean Squared Log Error (MSLE):**

Definition: MSLE is a metric used for datasets with exponential growth. It measures the average squared logarithmic difference between predicted and actual values.

Application: MSLE is used to assess the performance of the model on a logarithmic scale.

**9. Python Libraries:**

Usage: The code leverages popular Python libraries, including pandas for data manipulation, scikit-learn for machine learning functionalities, and other standard libraries.

**10. Data Preprocessing:**

Definition: Data preprocessing involves cleaning and transforming raw data into a format suitable for analysis.

Application: The code drops missing values, separates features and target variables, and performs other preprocessing steps.

**7.) Implementation**

1. Data Loading and Preprocessing:

Begin by loading the dataset into your chosen programming environment (e.g., Python with pandas).

Address any missing or irrelevant data by cleaning the dataset.

Handle categorical variables, if any, through encoding or other appropriate methods.

Provide code snippets and explanations for each step.

2. Data Exploration:

Conduct exploratory data analysis (EDA) to gain insights into the distribution and relationships within the dataset.

Visualize key features and their relationships with the target variable.

Include summary statistics, visualizations, or any relevant findings.

3. Train-Test Split:

Split the dataset into training and testing sets using the train\_test\_split function.

Specify the ratio of the split and the random seed for reproducibility.

Display the dimensions of the training and testing sets.

python

Copy code

# Example code for train-test split

from sklearn.model\_selection import train\_test\_split

# Split the data into features (x) and target variable (y)

x\_train, x\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size=0.2, random\_state=42)

4. Model Selection and Training:

Choose and instantiate the selected model (e.g., Linear Regression).

Fit the model to the training data.

Optionally, perform any necessary hyperparameter tuning.

python

Copy code

# Example code for model selection and training

from sklearn.linear\_model import LinearRegression

# Instantiate the Linear Regression model in the case of linear regression

model = LinearRegression()

# Fit the model to the training data

model.fit(x\_train, y\_train)

5. Model Evaluation:

Make predictions on the testing set using the trained model.

Evaluate the model's performance using relevant metrics (e.g., Mean Squared Error, R-squared).

Provide interpretations of the results.

python

Copy code

# Example code for model evaluation

from sklearn.metrics import mean\_squared\_error, r2\_score

# Make predictions on the testing set

y\_pred = model.predict(x\_test)

# Evaluate the model

mse = mean\_squared\_error(y\_test, y\_pred)

r2 = r2\_score(y\_test, y\_pred)

print(f'Mean Squared Error: {mse}')

print("R-squared (accuracy): ", r2)

6. Results and Discussion:

Summarize the key findings and results from the model evaluation.

Discuss the significance of the metrics and whether the model meets the predefined objectives.

Address any unexpected observations or challenges encountered during implementation.

7. Visualization (Optional):

Include visualizations to enhance the understanding of the model's performance.

This might include scatter plots, regression plots, or other relevant visualizations.

8. Future Steps:

Outline any potential improvements or additional steps to enhance the model.

Consider avenues for further analysis, such as feature engineering, parameter tuning, or exploring alternative models.

1. Linear Regression Model

The Linear Regression model is used in this context to establish a relationship between the features (R&D spending, Administration spending, and Marketing spending) and the target variable (Profit) in the dataset. Linear Regression assumes a linear relationship between the independent variables (features) and the dependent variable (target).

Here's how the Linear Regression model helps in this scenario:

Quantifying Relationships:

Linear Regression provides coefficients for each feature, indicating the strength and direction of their relationship with the target variable. For example, it helps understand how changes in R&D spending, Administration spending, and Marketing spending relate to changes in Profit.

Prediction:

Once trained on historical data, the Linear Regression model can make predictions on new, unseen data. In this case, it predicts the profit of startups based on the given features.

Interpretability:

Linear Regression is a highly interpretable model. The coefficients obtained from the model allow for easy interpretation of the impact of each feature on the target variable.

Evaluation Metrics:

The model performance is evaluated using metrics like Mean Squared Error (MSE), Mean Absolute Error (MAE), R-squared, Explained Variance, and Mean Squared Log Error. These metrics provide insights into how well the model generalizes to unseen data and how accurately it predicts the target variable.

Baseline Model:

Linear Regression serves as a baseline model, against which more complex models can be compared. If a relatively simple model like Linear Regression performs well, it suggests that the relationship between features and the target variable is somewhat linear.

Ease of Implementation:

Linear Regression is easy to implement and computationally efficient. It serves as a good starting point for regression problems, especially when the relationship between variables is expected to be approximately linear.

However, it's important to note that Linear Regression makes certain assumptions about the data, such as linearity, independence of errors, homoscedasticity, and normality of errors. If these assumptions are violated, other regression models or more advanced techniques might be considered. Additionally, exploring and understanding the data, as well as potentially engineering features, are crucial steps in building a reliable predictive model.

Output for the linear regression model:

Mean Squared Error: 80926321.22295158

Mean Absolute Error: 6979.152252370402

R-squared (accuracy): 0.9000653083037321

Explained Variance: 0.9035288469713902

R-squared percentage: 90.00653083037321

Mean Squared Error: 80926321.22295158

Mean Squared Log Error: 0.02572138240307621

1. Support Vector Regressor Model

Mean Squared Error: 103832131.51406816

Mean Absolute Error: 7702.6232454934825

R-squared (accuracy): 0.8717792691646302

Explained Variance: 0.9010723614424225

R-squared percentage: 87.17792691646302

Mean Squared Error: 103832131.51406816

Mean Squared Log Error: 0.029405587300442727

1. Lasso Regression Model

Mean Squared Error: 80926320.76116827

Mean Absolute Error: 6979.152235475119

R-squared (accuracy): 0.9000653088739813

Explained Variance: 0.9035288474534698

R-squared percentage: 90.00653088739813

Mean Squared Error: 80926320.76116827

Mean Squared Log Error: 0.025721382291390545

1. Neural network regressor model

Mean Squared Error: 768361286.0862532

Mean Absolute Error: 22538.07842052728

R-squared (accuracy): 0.051162253813159864

Explained Variance: 0.469447883150127

R-squared percentage: 5.116225381315987

Mean Squared Error: 768361286.0862532

Mean Squared Log Error: 0.16000283798834602

1. K Nearest Neighbours Regressor Model

Mean Squared Error: 570983489.7566228

Mean Absolute Error: 14869.748599999997

R-squared (accuracy): 0.2949011131337066

Explained Variance: 0.3877251915301302

R-squared percentage: 29.490111313370658

Mean Squared Error: 570983489.7566228

Mean Squared Log Error: 0.09416537009682295

1. Random Forest Regressor Model

Mean Squared Error: 72625008.62306513

Mean Absolute Error: 6437.497739999977

R-squared (accuracy): 0.9103164738430438

Explained Variance: 0.9121542131235096

R-squared percentage: 91.03164738430438

Mean Squared Error: 72625008.62306513

Mean Squared Log Error: 0.018918402267552027

**8.)Conclusion:**

The Linear Regression model appears to perform reasonably well in predicting the profit of startups based on R&D spending, Administration spending, and Marketing spending. However, further analysis, feature engineering, or trying different models could potentially enhance predictive accuracy. The Linear Regression model's performance is assessed using the above-mentioned metrics. The values of MSE, MAE, R-squared, Explained Variance, and MSLE provide insights into how well the model generalizes to unseen data.