Capstone Project: Student Performance Matrix Predication (Regression Analysis)

Introduction

The Student Performance Dataset is a dataset designed to examine the factors influencing academic student performance. The dataset consists of 10,000 student records, with each record containing information about various predictors and a performance index.

Problem Statement:

The problem at hand revolves around understanding and analyzing the factors influencing academic student performance based on the Student Performance Dataset. With a dataset comprising 10,000 student records, the objective is to uncover the relationships between predictor variables and the performance index.

Key Objectives:

1. **Exploratory Analysis:** Conduct exploratory analysis to identify trends, patterns, and correlations within the dataset.
2. **Predictive Modeling:** Develop predictive models to determine the impact of various factors such as study hours, previous scores, extracurricular activities, sleep hours, and sample question papers on student performance.
3. **Insights Generation:** Generate actionable insights to aid educators, policymakers, and stakeholders in improving student performance and academic outcomes.

Potential Questions to Address:

1. What is the distribution of the performance index among students?
2. Are there any correlations between studying hours, previous scores, extracurricular activities, sleep hours, and performance index?
3. Which predictor variables have the most significant influence on student performance?
4. Can predictive models accurately predict student performance based on the provided predictor variables?
5. How can insights from the analysis be leveraged to enhance educational strategies and support student success?

ATTRIBUTE INFORMATION

Variables:

- **Hours Studied**: The total number of hours spent studying by each student.
- **Previous Scores**: The scores obtained by students in previous tests.
- **Extracurricular Activities**: Whether the student participates in extracurricular activities (Yes or No).
- **Sleep Hours**: The average number of hours of sleep the student had per day.
- **Sample Question Papers Practiced**: The number of sample question papers the student practiced.

Target Variable:

- **Performance Index**: A measure of the overall performance of each student. The performance index represents the student's academic performance and has been rounded to the nearest integer. The index ranges from 10 to 100, with higher values indicating better performance.

Steps involved.

1. Loading and discovering data

Now, we need to load our data from the external source, which in this case is uploaded to the drive. The data is in the format of the CSV (Comma Separated Values) file.

2. Data Cleaning

Data cleaning is an important step in the data analytics process in which you either remove or update information that is incomplete or improperly formatted.

3. Null values Treatment by different methods

We have our dataset in hand which is raw and unfiltered. As this step involves cleaning our data first by eliminating the columns which are not needed for our analysis. We have around

10000 rows × 6 columns in our dataset. Since there were no null values, we have left it untouched.

4. Exploratory Data Analysis

Exploratory Data Analysis is the approach of analyzing data, gathering and summarizing the important characteristics of the information, and using simple visualization that makes it easier to understand.

5. Importing necessary modules and libraries

We are importing the following libraries for their respective applications:

Pandas: Pandas is used to analyze data. It has functions for analyzing, cleaning, exploring, and manipulating data.

Matplotlib: Matplotlib is a graph plotting library in Python that serves as a visualization utility. Most of the Matplotlib utilities lie under the pyplot submodule.

Numpy: NumPy is a Python library used for working with arrays. It also has functions for working in the domain of linear algebra, Fourier transform, and matrices.

SciKit: Scikitlearn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools.

Plotly: The plotly is an interactive, opensource plotting library that supports over 40 unique chart types covering a wide range of statistical, financial, geographic, scientific, and 3dimensional usecases.

Seaborn: Seaborn is a library that uses Matplotlib underneath to plot graphs. It will be used to visualize random distributions.

Statsmodels: Statsmodels is a Python module that provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration.

for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

6. Plotting various graphs for different parameters.

7. Finding the key facts and relationships between various parameters.

8. Observations according to the outputs of the graph visualizations.

Multicollinearity:

Multicollinearity is the occurrence of high intercorrelations among two or more independent variables in a multiple regression model. Multicollinearity can lead to skewed or misleading results when a researcher or analyst attempts to determine how well each independent variable can be used most effectively to predict or understand the dependent variable in a statistical model.

VIF (Variable Inflation Factors):

VIF determines the strength of the correlation between the independent variables. It is predicted by taking a variable and regressing it against every other variable.

Outlier Detection:

An outlier detection technique (ODT) is used to detect anomalous observations/samples that do not fit the typical/normal statistical distribution of a dataset. Simple methods for outlier detection use statistical tools, such as boxplot and Zscore, on each individual feature of the dataset.

Optimization:

Function optimization is the problem of finding the set of inputs to a target objective function that result in the minimum or maximum of the function.

It can be a challenging problem as the function may have tens, hundreds, thousands, or even millions of inputs, and the structure of the function is unknown, and often nondifferentiable and noisy.

Regression:

Regression is a supervised learning technique which helps in finding the correlation between variables and enables us to predict the continuous output variable based on the one or more predictor variables.

Feature Engineering:

Feature engineering is a machine learning technique that leverages data to create new variables that aren't in the training set. It can produce new features for both supervised and unsupervised learning, with the goal of simplifying and speeding up data transformations while also enhancing model accuracy.

Train and Test Split:

The traintest split is used to estimate the performance of machine learning algorithms that are applicable for predictionbased Algorithms/Applications.

Linear Regression:

Linear Regression is a supervised machine learning algorithm where the predicted output is continuous and has a constant slope. It's used to predict values within a continuous range, rather than trying to classify them into categories. There are two main types:

Simple regression:

$y=mx+b$

Simple linear regression uses traditional slope-intercept form, where m and b are the variables our algorithm will try to "learn" to produce the most accurate predictions. X- represents our input data and y represents our prediction.

Multivariable regression:

$f(x,y,z)=w1x+w2y+w3z$

A more complex, multivariable linear equation might look like this, where w represents the coefficients or weights, our model will try to learn.

Ridge Regression:

In Ridge regression, we add a penalty term which is equal to the square of the coefficient. The L2 term is equal to the square of the magnitude of the coefficients. We also add a coefficient lambda to control that penalty term. In this case, if lambda is zero then the equation is the basic OLS else if lambda > 0 then it will add a constraint to the coefficient. As we increase the value of lambda this constraint causes the value of the coefficient to tend towards zero. This leads to a trade-off of higher bias (dependencies on certain coefficients tend to be 0 and on certain coefficients tend to be very large, making the model less flexible) for lower variance.

Lasso Regression:

Lasso regression stands for Least Absolute Shrinkage and Selection Operator. It adds a penalty term to the cost function. This term is the absolute sum of the coefficients. As the value of coefficients increases from 0 this term penalizes, and causes model, to decrease the value of coefficients in order to reduce loss. The difference between ridge and lasso regression is that it tends to make coefficients to absolute zero as compared to

Decision Tree Regressor:

Decision trees can be used for classification as well as regression problems. The name itself suggests that it uses a flowchart like a tree structure to show the predictions that result from a series of feature-based splits. It starts with a root node and ends with a decision made by leaves.

Decision trees are upside down which means the root is at the top and then this root is split into various several nodes. Decision trees are nothing but a bunch of if-else statements in layman's terms. It checks if the condition is true and if it is then it goes to the next node attached to that decision.

Random Forest Regressor:

Random Forest is a technique that uses ensemble learning, that combines many weak classifiers to provide solutions to complex problems.

As the name suggests random forest consists of many decision trees. Rather than depending on one tree it takes the prediction from each tree and based on the majority votes of predictions, predicts the final output.

Random forests use the bagging method. It creates a subset of the original dataset, and the final output is based on majority ranking and hence the problem of overfitting is taken care of.

XGBoost Regressor:

XGBoost is a powerful approach for building supervised regression models. The validity of this statement can be inferred by knowing about its (XGBoost) objective function and base learners. The objective function contains a loss function and a regularization term. It tells about the difference between actual values and

Data Visualization Analysis:

Data Visualization is the process of analyzing data in the form of graphs or maps, making it a lot easier to understand the trends or patterns in the data.

Correlation Heat map:

Analysis of the relation between the various columns of the cleaned data through the Correlation Heat map Matrix. A correlation heat map is a graphical representation of a correlation matrix representing the correlation between different variables,and a better one sums up to form final good predictions.

Box Plot:

A boxplot is a standardized way of displaying the distribution of data based on a five number summary (“minimum”, first quartile [Q1], median, third quartile [Q3] and “maximum”). It can tell you about your outliers and what their values are.

Bar Chart:

A bar chart is a chart that presents categorical data with rectangular bars with heights or lengths proportional to the values that they represent. A bar chart is sometimes called a column chart.

Horizontal Bar Chart:

A horizontal bar chart is a graph in the form of rectangular bars. The length of these bars is proportional to the values they represent.

Count Plot:

The countplot is used to represent the occurrence(counts) of the observation present in the categorical variable. It uses the concept of a bar chart for the visual depiction.

Distplot:

distplot() function is used to plot the distplot. The distplot represents the univariate distribution of data i.e. data distribution of a variable against the density distribution. The seaborn. distplot() function accepts the data variable as an argument and returns the plot with the density distribution.

Subplot:

A subplot is a narrative thread that is woven through a book to support the elements of the main plot.

Challenges:

•Large Dataset to handle.

•Needs to plot a lot of Graphs to analyse.

•Carefully handled the Feature selection part as it affects the R2 score.

•Carefully tuned Hyperparameters as it affects the R2 score.

•Handled the positive skewness of the target variable.

•Handled the high correlation between various features.

•Need to convert categorical features into numerical features using feature engineering.

Conclusion:

The Value of RMSE of the Test Data should as low as possible to perform the model better

1. Linear regression:- **455.711**

2. Ridge Regression:- **455.724**

3. Lasso Regression: - **455.73**

4. Decision Tree: - **357.498**

5. Random Forest Regression: - **277.647**

6. XG Boost Regressor: - **249.829**

7. **Random Forest Regression** and **XG Boost Regressor** gives the highest R2 Score which is 99% and 98% with the test data repectively and 98% with the train dataset which is very good and generalised model without getting any overfit.

8. The Most important feature of the dataset is the previous year score which is very important for the model and also followed by the study hours for a better prediction of the performance Index of the student.

9. Hence for the prediction of the performance Index of the Student **Random Forest Regression** and **XG Boost Regressor** can be used.

References

- Python Pandas Documentation:

https://pandas.pydata.org/pandas

docs/stable

- Python Numpy Documentation:

https://numpy.org/doc/

- Python MatPlotLib Documentation:

https://matplotlib.org/stable/index. html

- SciKit Documentation:

https://scikitlearn.org/stable/