**Car Sales Prediction using Linear Regression**

November 28, 2023

**Abstract**

This project explores the application of linear regression in predicting car sales, utilizing cloud-based tools such as AWS Glue and AWS SageMaker. The dataset, sourced from Kaggle, encompasses comprehensive information on car specifications, pricing, and marketing details. This report outlines the methodology, data preprocessing steps, model development, and evaluation, shedding light on the effectiveness of linear regression in forecasting car sales.

The automotive industry is highly competitive, with manufacturers and dealers constantly seeking ways to optimize sales performance. Predicting car sales accurately is essential for strategic decision-making, allowing businesses to adjust pricing, marketing, and inventory strategies. This report outlines a solution pipeline for car sales prediction using linear regression, leveraging a dataset containing historical sales data and relevant features.

**Linear Regression Machine Learning:**

Linear regression is a fundamental machine learning approach that predicts a continuous result variable (also known as the dependent variable) using one or more predictor factors (independent variables). The variables' connection is considered to be linear, which means that changes in the predictor variables are connected with a linear change in the result variable.

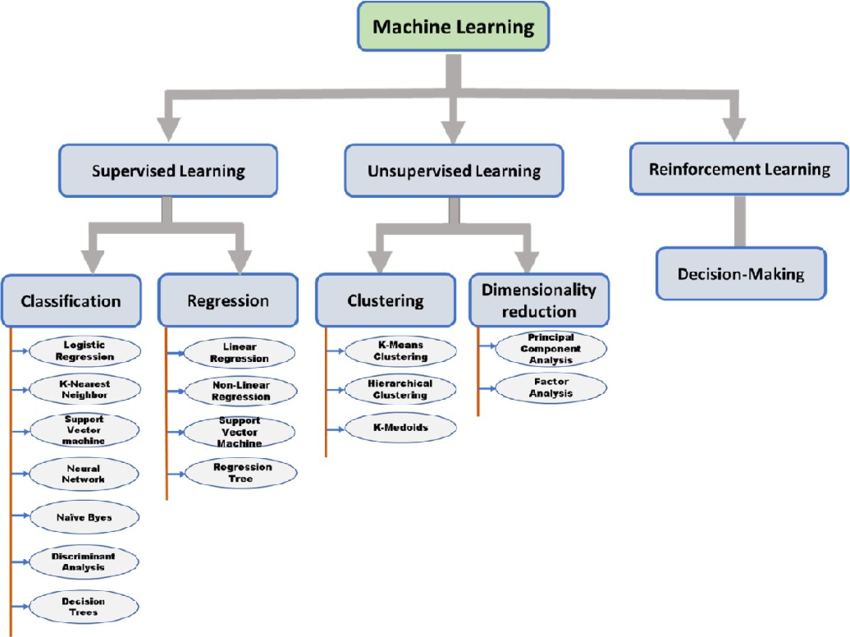


Fig 1: Machine Learning Types FlowChart

**Dataset Explanation:**

The dataset used for this project is sourced from Kaggle and is focused on car sales prediction. It contains a diverse set of information related to car specifications and pricing. The dataset includes the following columns:

* Year
* EngineV
* Model
* Registration
* Engine Type
* Mileage
* Body
* Price

**Target Label:**

* Car Sales: Car sale price with other variables, representing the outcome variable we aim to predict.

**Architecture:**

The AWS services and components used in this project is determined in the flow chart as follows:

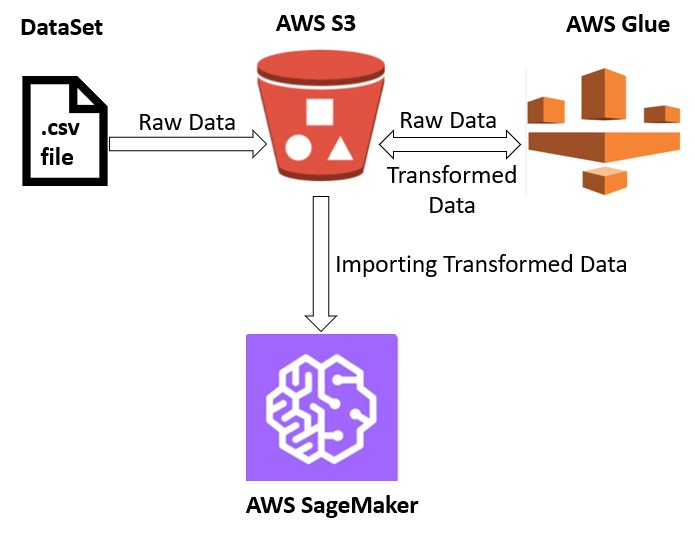


Fig 2: Machine Learning Model implementation Architecture using AWS services

Dataset has been taken from kaggle which is a .csv file provided as input to the AWS S3 bucket where the input datastore folder is created to store the input csv file i.e..,the dataset taken from kaggle and target datastore is also created to store the transformed file.

AWS Glue is used to perform ETL(Extract, Transform and Load) activity to transform the data according to the business requirements. ETL activity is performed using the database and crawlers, finally the data is again loaded to S3 target datastore.

The file that is transformed into the S3 bucket is taken as input data to sagemaker using python code and libraries to read the data. The SageMaker operations include implementing the code, training and testing the data using various parameters and determining the summary statistics which gives the accuracy of the model.

**Solution Pipeline:**

**1. Data Procurement:**

In this project, the dataset for car sales prediction was obtained from Kaggle. We chose a dataset that is well-maintained, with clear documentation on column meanings and potential issues. Consider the size of the dataset in relation to computational resources. Large datasets may require additional processing capabilities. Ensuring the dataset collection and usage align with ethical standards and legal regulations.

**2. Data Cleaning:**

Data cleaning is a crucial step in the solution pipeline that involves preparing the dataset for analysis by addressing issues such as missing values, duplicates, and outliers.

**3. Feature Selection:**

Relevant columns were selected based on their potential impact on car sales. This included car specifications, pricing information.

**4. Data Imputation:**

Missing values, if any, were imputed using appropriate methods such as mean or median imputation. This step helps maintain the integrity of the dataset.

**5. Splitting Data into Train/Test Sets:**

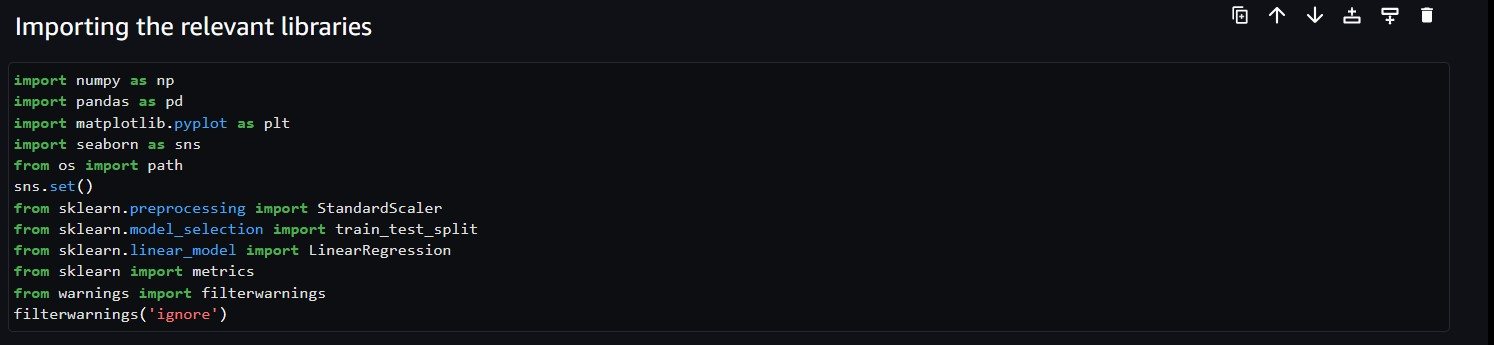
The dataset was divided into two categories:

* Training Data to train the model.
* Testing Data to test the model.

The common practice of a 70-30 split was applied, ensuring an adequate amount of data for both training and testing.

**Step 1: Importing the required libraries**

In this step, we are importing Numpy, Pandas, Seaborn, Sklearn, and Matplotlib libraries.



**Step 2: Importing the dataset.**

After downloading the dataset from Kaggle, ETL activities were performed in AWS Glue to transform the data according to the business requirements and load it to the destination datastore.



Using Pandas’ read\_csv() function, we obtain data from the dataset, which is stored in an AWS S3 bucket, and put it in the raw\_data variable.

**Step 3: Visualizing the Data**

Based on the dataset, the visualization of data is shown, and the data is scaled accordingly.

Some of the graphs are shown below. From Fig 2, the price is not normally distributed, so removing the outliers of data is needed.

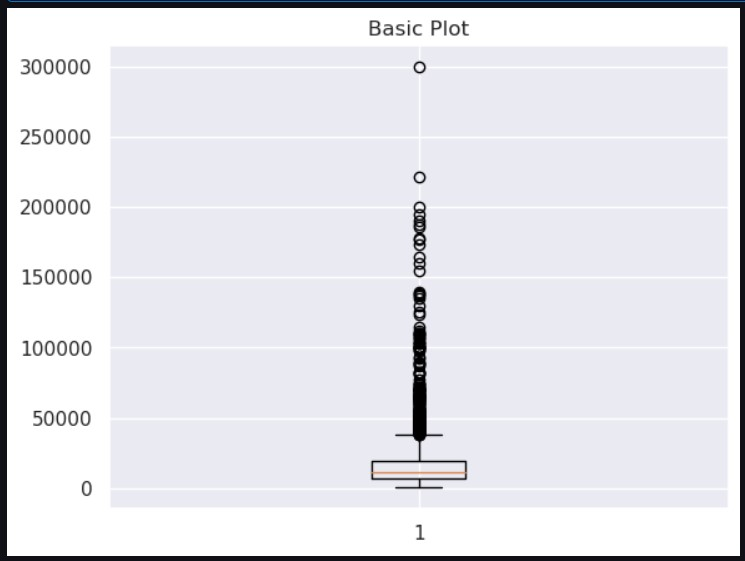
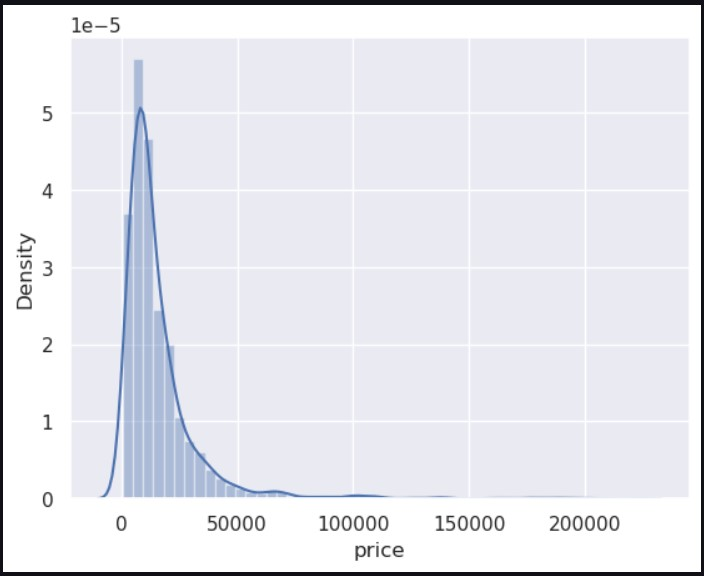
 

Fig 2: Box plot of price. Fig 3: Distplot of price.

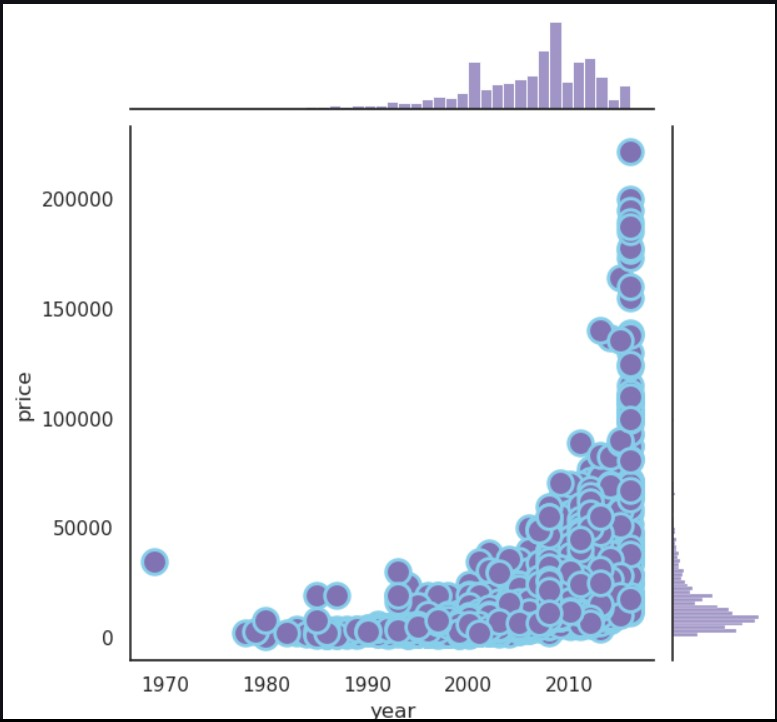
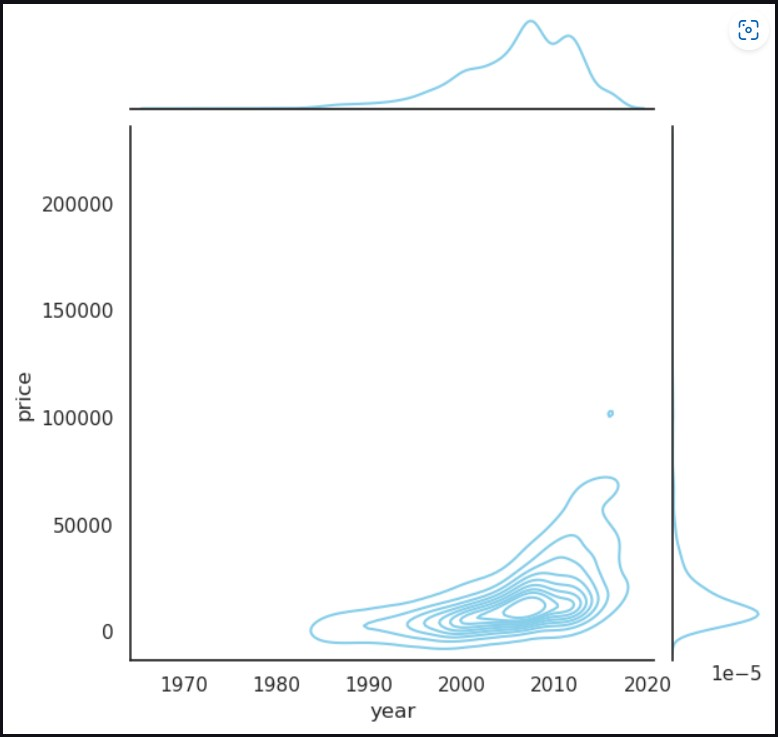
 

Fig 4: Jointplot of price versus year. Fig 5: Jointplot of price versus year.

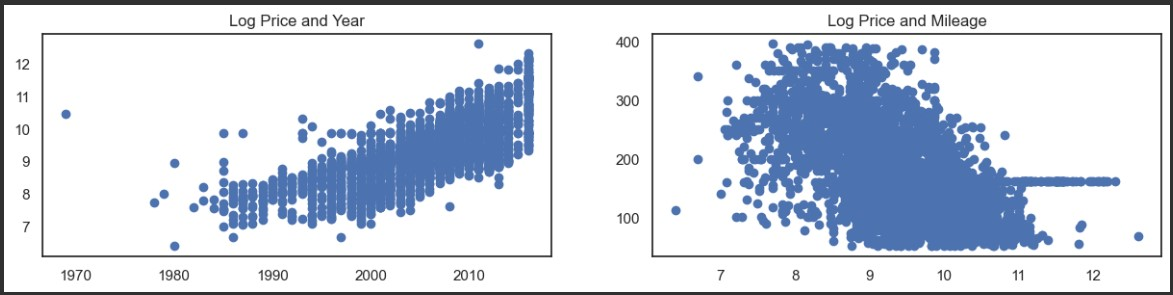
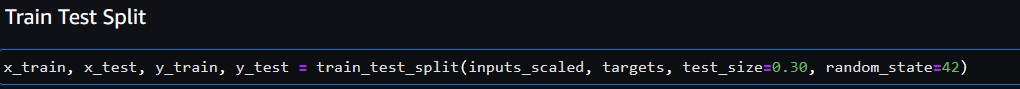


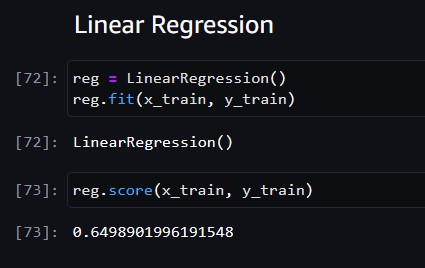
Fig 6: Scatter plot of Log price vs Year.

**Step 4: Training, Testing, and Splitting.**

At this point, the splitting of data is done where 70% of the data is being trained, and 30 % of the data is used for testing.



**Step 5: Linear Regression**



**Step 6: Testing the data**

Final Prediction:

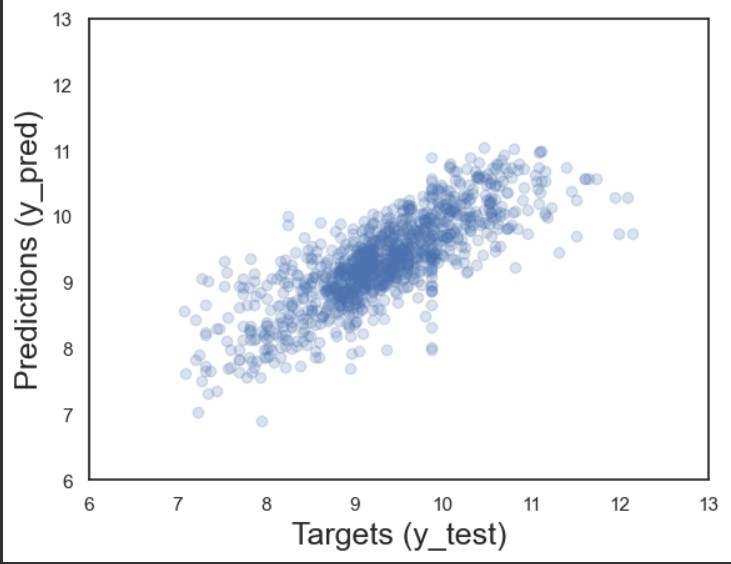
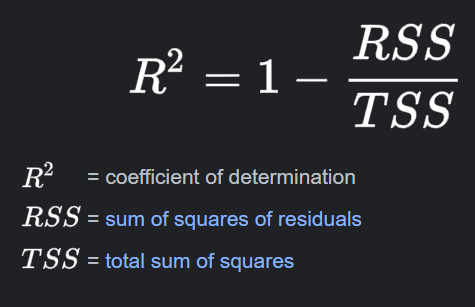


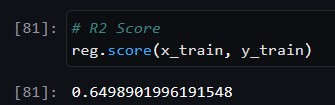
Fig 7: Scatter plot of Targets vs Predictions.

**R2 Score:**

The R-squared (R2) score is a statistical measure indicating the percentage of variation in the dependent variable of a regression model that can be accounted for by the independent variables. Essentially, it quantifies the degree of fit of the model. The R2 value is generally between 0 and 1, with 0 implying that the model describes no variation in the dependent variable and one indicating that the model explains all variability. A greater R2 value indicates that the model fits the data better.

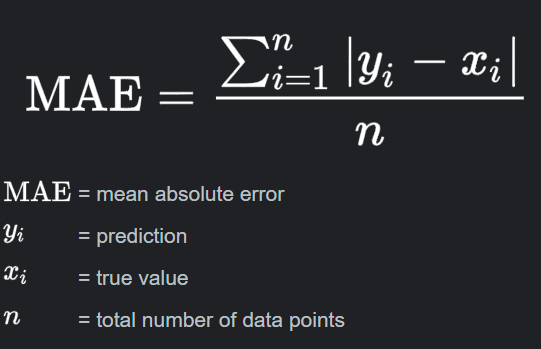


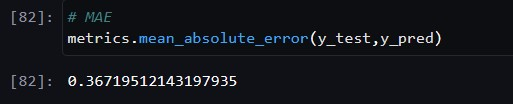
​



**MAE:**

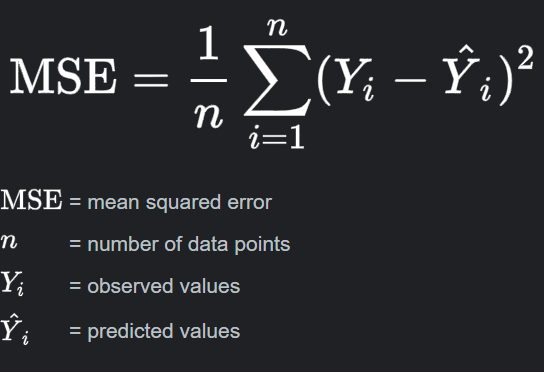
Another commonly employed statistic for evaluating the effectiveness of a regression model is MAE, which stands for Mean Absolute Error. It computes the average absolute difference between the actual and predicted values. Lower MAE values suggest greater accuracy of the model since they show that the model's estimation is closest to the actual values on average.

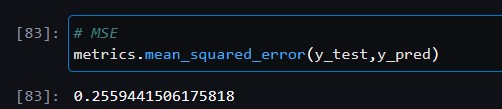




**MSE:**

Another frequently employed metric for assessing the effectiveness of a regression model is MSE, denoting Mean Squared Error. It computes the average of the squared differences between the actual and predicted values. Due to the squaring operation, MSE imposes a higher penalty on larger errors compared to smaller ones. Like MAE, MSE is expressed in the square of the data units, rendering it less intuitively interpretable than MAE. Nevertheless, MSE is commonly utilized in optimization problems due to its favorable mathematical properties.

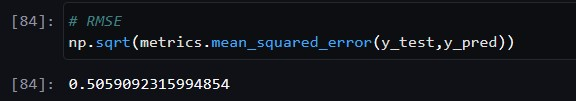




**RMSE:**

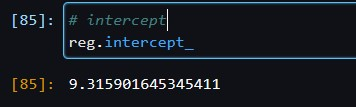
The Root Mean Squared Error, or RMSE, is a popular statistic for assessing the effectiveness of a regression model. It is a variant of Mean Squared Error (MSE), but it tackles the squared unit issue by calculating the square root of the MSE.RMSE is valuable as it offers a measure of the average size of prediction mistakes. It penalizes greater errors more harshly than smaller errors, similar to MSE, although the square root helps solve the problem of squared units.





**Intercept:**

In the context of regression analysis, the term "intercept" pertains to the y-intercept of the linear equation that characterizes the relationship between the independent variable(s) and the dependent variable. To ascertain the intercept mathematically, one must compute coefficients that minimize the sum of squared differences between the actual and predicted values. In the context of linear regression, this process is often executed through techniques like ordinary least squares (OLS).



**References:**

1. <https://www.researchgate.net/publication/366359344/figure/fig2/AS:11431281110663402@1672677311128/Flowchart-of-the-types-of-machine-learning-techniques.png>
2. “sklearn.metrics.r2\_score,”*Scikit-learn*.<https://scikit-learn.org/stable/modules/generated/sklearn.metrics.r2_score.html>
3. S. Manoj, J. P. Ananth, M. Rohini, B. Dhanka, N. Pooranam, and S. R. Arumugam, “FWS-DL: forecasting wind speed based on deep learning algorithms,” in *Elsevier eBooks*, 2022, pp. 353–374. doi: 10.1016/b978-0-323-90396-7.00007-9.
4. S. Online, “Mean squared error in machine learning - Shiksha Online,” *Shiksha.com*, Nov28,2022.<https://www.shiksha.com/online-courses/articles/mean-squared-error-in-machine-learning/#:~:text=Mean%20Squared%20Error%20(MSE)%20measures,as%20the%20model%20error%20increases>.
5. “SAPHelpPortal.”<https://help.sap.com/docs/SAP_PREDICTIVE_ANALYTICS/41d1a6d4e7574e32b815f1cc87c00f42/5e5198fd4afe4ae5b48fefe0d3161810.html>

**Ref 1) Figure 1 - Source URL:** https://www.researchgate.net/publication/366359344/figure/fig2/AS:114hart-of-the-tyes-of-machine-learning-techniques.png