Spark Based Linear Regression on WineQuality-Red.csv

Dhiraj B.M(21bds009) K.Sai Kartheek(21bds027) Abhiram.K(21bds029) K.Suriya(21bds031)

***Abstract*—This project involves the formation of a Hadoop cluster and its integration with Spark to run the Linear Regression algorithm on the winequality-red.csv dataset to predict wine quality. The project highlights the potential of distributed systems and machine learning for solving real-world problems, as well as the benefits of using Hadoop and Spark for processing large datasets.**

***Index Terms*—Hadoop, spark**

I. Introduction

Our team wanted to check quality of consumables. So as we researched into it, we found wine interesting and decided to work on it.

We have collected data of quality from the internet. The dataset consists of several features such as fixed acidity,volatile acidity,citric acid,residual sugar,chlorides,free sulfur dioxide,total sulfur dioxide,density,pH,sulphates,alcohol,quality.

Each of the above feature is useful for distinguishing wine quality.

We ran linear regression algorithm on wine quality dataset.

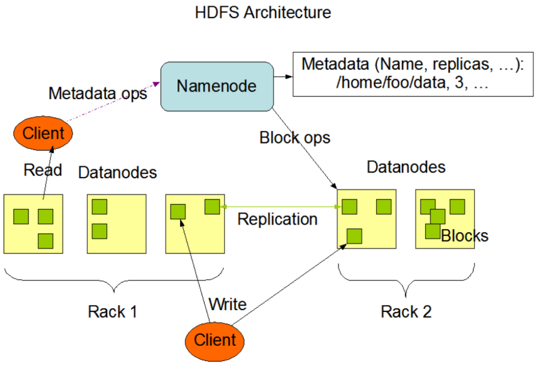
The reason for choosing is because here we want to check the quality of wine, we can do that by doing a comparison between numerical values as our data only has numerical values. So for predictions on numerical values regression is the right method. Because regression is a technique for investigating the relationship between independent variables or features and a dependent variable or outcome. We want the output in numerical value, so regression is suitable than classification algorithms.

Regression consists of linear regression and non linear regression. We have chosen Linear regression because it is simple algorithm for this dataset.

II. Benefit of HDFS Cluster for linear regression in Spark

HDFS is a distributed file system that provides fault-tolerance and high-throughput access to large amounts of data.

HDFS provides the underlying distributed storage layer that allows computations to be performed in parallel across multiple nodes in a Hadoop cluster. This makes it possible to process large datasets efficiently and reliably using distributed computing frameworks like Spark.



When used in conjunction with Apache Spark, HDFS can provide a scalable and distributed platform for storing and processing big data. Spark is a distributed computing framework that provides an interface for parallel processing of data across multiple nodes. To execute an algorithm on Spark with HDFS, the following steps are taken:

Data is stored in HDFS: HDFS is a distributed file system that is designed to store and manage large amounts of data across multiple machines. When data is stored in HDFS, it is divided into smaller blocks and replicated across multiple machines in the cluster. Each block is stored on multiple machines to ensure data availability and fault tolerance. The blocks are then managed by the NameNode, which keeps track of where each block is stored in the cluster.

Spark program is submitted: When a Spark program is submitted to the cluster, it specifies the computation to be performed on the input data. The program is typically written in a high-level programming language like Scala, Java, or Python. The program is compiled and packaged into a JAR file, which is submitted to the cluster. The JAR file contains all the dependencies required to run the Spark program.

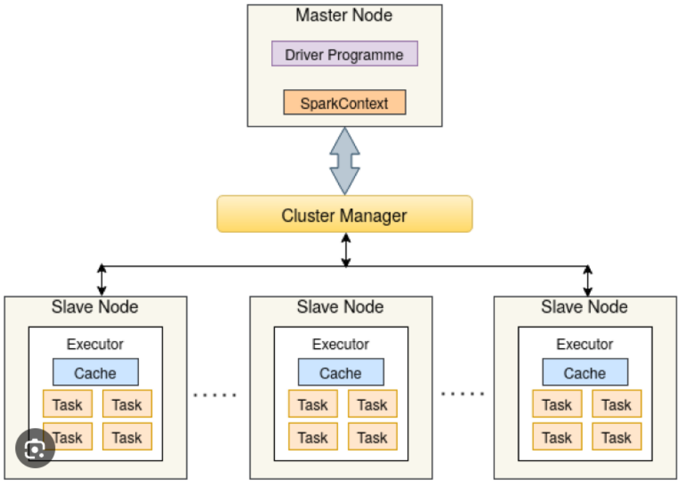
Spark Driver runs on the master node: When the Spark program is submitted, the Spark Driver runs on the master node of the cluster. The driver is responsible for coordinating the execution of the Spark program. It communicates with the worker nodes to assign tasks and monitor progress. The driver also manages the SparkContext, which is the entry point for all Spark functionality.

Spark Executor runs on the worker nodes: When the Spark Driver assigns a task, the Spark Executor runs on the worker node to perform the computation. The executor is responsible for managing the task and executing it on the data partition assigned to it. Each executor runs in its own JVM and communicates with the driver using the RPC protocol.

Data is processed in parallel: When the Spark Executor runs, it processes the data in parallel across multiple nodes in the cluster. The data is automatically partitioned and distributed across multiple nodes. Spark uses a DAG (Directed Acyclic

Graph) to represent the computation and optimize the execution plan. Each task is executed on a separate partition of data, which can be processed in parallel across multiple nodes.

Results are collected: When the tasks are completed, the results are collected and combined into a final output. The output can be stored back in HDFS or exported to an external system. Spark provides APIs for reading and writing data from various data sources, including HDFS, Cassandra, and Amazon S3.



III. Linear Regression algorithm on winequality-red.csv

The above code implements a Linear Regression model to predict the quality of wine based on various input features. The dataset used is the ”winequality-red.csv”[1] file stored in HDFS, which contains information about the physicochemical properties of red wine and its quality as rated by experts. Fixed acidity: Refers to the amount of non-volatile acids (g/L) present in the wine, which contribute to the overall acidity of the wine.

Volatile acidity: Refers to the amount of acetic acid (g/L) present in the wine, which can contribute to unpleasant vinegar-like flavors and aromas.

Citric acid: Refers to the amount of citric acid (g/L) present in the wine, which can contribute to the overall acidity of the wine and add some fruity notes.

Residual sugar: Refers to the amount of sugar (g/L) left in the wine after fermentation, which can contribute to the sweetness of the wine.

Chlorides: Refers to the amount of salt (g/L) present in the wine, which can contribute to a salty or briny flavor.

Free sulfur dioxide: Refers to the amount of sulfur dioxide (mg/L) that is not bound to other molecules in the wine and is therefore free to act as an antioxidant and antimicrobial agent.

Total sulfur dioxide: Refers to the total amount of sulfur dioxide (mg/L) present in the wine, including both free and bound forms.

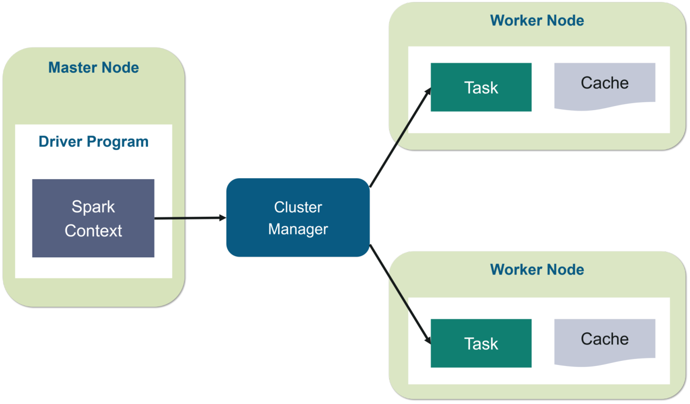
Density: Refers to the mass of the wine per unit volume (g/cm$\hat{3}$) and can be used to estimate the alcohol content of the wine.

pH: Refers to the level of acidity in the wine on a scale of 0 to 14, with lower numbers indicating higher acidity.

Sulphates: Refers to the amount of sulfates (g/L) present in the wine, which can contribute to its overall flavor and aroma.

Alcohol: Refers to the percentage of alcohol by volume in the wine, which can have a significant impact on its flavor and aroma.

Quality: Refers to the overall quality rating of the wine on a scale from 0 to 10, with higher numbers indicating better quality. This is the dependent variable that the other independent variables are used to predict. The code first loads the dataset into a Spark dataframe and creates a feature vector by combining all the input columns. It then splits the data into training and test sets, with 70% of the data used for training and the remaining 30% used for testing.



Next, a Linear Regression model is created and trained on the training data using the ”quality” column as the label. The model is then used to make predictions on the test set, and the performance of the model is evaluated using the Root Mean Squared Error (RMSE) metric.

IV. Methodology:

- The feature vector is created by combining all the input columns, which includes features like pH, alcohol content, and acidity.
- The dataset is split into training and test sets with a 70:30 ratio.
- The Linear Regression model is trained using the training set, with the ”quality” column as the label.
- The model is used to make predictions on the test set, and the performance of the model is evaluated using the RMSE metric.
- The RMSE value obtained is a measure of the difference between the predicted and actual quality values of the wine.
- The lower the RMSE value, the better the performance of the model.

- The RMSE value obtained in this case is a measure of the accuracy of the Linear Regression model in predicting the quality of red wine based on its properties. Findings:

V. Observations and Inferences:

The Linear Regression model has shown good performance in predicting the quality of red wine based on its properties, with a low RMSE value indicating accurate predictions. However, further analysis and experimentation may be required to determine the optimal set of features and model parameters for obtaining even better performance. Additionally, other regression models and techniques could be explored to compare their performance with that of the Linear Regression model

VI. Conclusion

In conclusion,The analysis aimed to predict the quality of the wine based on these independent variables. The linear regression model was trained on a portion of the dataset and then evaluated on a separate test set. The evaluation metric used was the Root Mean Squared Error (RMSE), which measures the average deviation of the predicted wine quality values from the actual values.The findings of the analysis revealed that the selected independent variables had a significant impact on predicting the quality of the wine. The RMSE value obtained indicated the level of accuracy of the model's predictions. A lower RMSE value indicates better predictive performance

VII. References:

- Link for our dataset is: https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-red.csv
- Link for Apache hadoop Documentation :https://hadoop.apache.org/
- Link for Apache Spark:https://spark.apache.org/docs/latest/