Faculty of Science-Rabat

Machine Learning SparkMLlib

Module : *Big Data*

Thème

Road traffic analysis for accident prevention

Supervised

— Abderrahmane EZ-ZAHOUT

Directed by

— Brahim RABANY

— Yasser EL HARBILI

— Salah Eddine EL FARKH

— Ajfan AHMED

2023/2024

Abstract

This report demonstrates the synergy between machine learning, Big Data analytics, and Spark MLlib in addressing complex challenges related to road safety in Pakistan. By leveraging these advanced technologies, the analysis not only provides actionable insights but also incorporates algorithm performance metrics to assess the effectiveness of the predictive models.[6] The innovative approach involves real-time traffic analysis using predictive models that can detect abnormal patterns, contributing to an enhanced understanding of accident dynamics.[2]The performance of these predictive models is rigorously evaluated using algorithmic metrics such as mean squared error (MSE) and mean absolute error (MAE), ensuring their reliability and accuracy in forecasting outcomes.

Introduction

The rapid evolution of information technologies and the exponential growth of road traffic have underscored the importance of devising innovative methods to prevent road accidents.[2] In this context, this study explores the application of Spark MLlib, a distributed machine learning library, to analyze real-time road traffic and identify abnormal patterns conducive to accidents. Following a rigorous methodology, we elucidate how Spark MLlib can be seamlessly integrated into a comprehensive workflow, spanning from data preparation to result evaluation. [4]

The utilization of advanced methods such as cloud computing and data analytics in the realm of road safety showcases the potential of these approaches in enhancing accident prevention.[5] By harnessing Spark's MLlib algorithms, we not only detect abnormal traffic patterns but also assess the performance of predictive models using metrics like mean squared error (MSE) and mean absolute error (MAE).[3]This thorough evaluation ensures the reliability and accuracy of forecasts, thereby bolstering our capability to anticipate and forestall road accidents.

Chapter 1 : Literature Review

.

1.1 State of the Art in Road Safety

Road safety, also known as traffic safety, encompasses a set of measures designed to prevent fatal accidents or serious injuries to road users.[2] These measures aim to reduce the risks of accidents (risk prevention) and mitigate the consequences in the event of an accident (forecasting).[6]

1.2 Machine Learning Applications in Road Safety

Previous studies have emphasized the significance of integrating machine learning techniques in road safety efforts, showcasing their potential to enhance accident prevention strategies. The utilization of cloud computing infrastructure, coupled with advanced machine learning algorithms, presents a formidable toolset for analyzing vast amounts of road accident data. By leveraging these technologies, researchers can gain insights into the underlying patterns and contributing factors of accidents, enabling proactive measures to mitigate risks and improve road safety [3].

Furthermore, the comprehensive analysis of road accident data in the United States underscores the effectiveness of machine learning approaches in identifying and addressing various safety concerns. Through data-driven insights derived from machine learning models, stakeholders can develop targeted interventions and implement preventive measures tailored to specific risk factors observed in different geographic regions or demographic groups. This holistic approach, integrating cutting-edge technologies with robust data analytics, represents a proactive stance towards reducing road accidents and promoting safer transportation systems [2].

1.3 Overview of the Choice of Spark MLlib and Algorithms for Road Traffic Analysis

As part of our study on Road Traffic Analysis for Accident Prevention, the choice of Spark MLlib as the machine learning framework is motivated by its ability to process large amounts of data in a distributed manner. This provides an efficient solution for analyzing traffic patterns in real-time and predicting potential accidents [4]. By leveraging Spark MLlib's distributed processing capabilities, researchers can overcome the challenges associated with handling vast datasets, enabling the development of accurate and scalable predictive models for road safety applications. Moreover, Spark MLlib's integration with the broader Spark ecosystem enhances its versatility and usability, empowering researchers to explore innovative approaches to address complex road safety challenges [5].

1.4 Why Spark MLlib ?

The choice of Spark MLlib is driven by several compelling reasons, making it a preferred framework for our study on Road Traffic Analysis for Accident Prevention.

1.5 Machine Learning with MLlib

"SparkML and MLlib are core libraries of Spark that provide many useful utilities for machine learning tasks, including utilities suitable for :

- Classification
- Regression
- Clustering

Chapter 2 : Implementation

2.1 Data Collection

Data collection is the process of gathering and accumulating raw data from various sources. It involves systematically collecting information relevant to a particular research question, problem, or objective [2]. Data collection methods can vary depending on the nature of the data and the research context [4]. Common data collection techniques include surveys, interviews, observations, experiments, and document analysis [3]. The collected data can be quantitative (numerical) or qualitative (non-numerical), and it serves as the foundation for subsequent analysis and interpretation to derive meaningful insights and conclusions [1]. Effective data collection is essential for obtaining accurate and reliable information to address research objectives or make informed decisions [5].

2. Data Collection:

```
# Load the dataset into a Spark DataFrame
df = spark.read.csv("C:/Users/HP/Downloads/Traffic Accidents in Pakistan.csv",
                    header=True,
                    inferSchema=True)
df.show()
```

```
+-----------+------------------------+-------------+-----------------+-------------+-------------+--
---------+
|Month - Year|Total Number of Accident|Accident Fatal|Accident Non-Fatal|Person Killed|Person Injured|To
Involves|
+-----------+------------------------+-------------+-----------------+-------------+-------------+--
```

2.2 Data Cleaning and Preprocessing :

Data preprocessing is essential to ensure data quality before analysis.

Checking and handling missing data :

```
# Check for missing values in the entire DataFrame
#df_missing = df.select([col(c).isNull().alias(c) for c in df.columns]).show()

#df_missing
# # Check for missing values in a specific column
# df.select("your_column_name").isNull().sum().show()

# # Fill missing values with a specific value
# df = df.fillna({"your_column_name": "replacement_value"})

# Drop rows with missing values
df = df.dropna()
df.show()
```

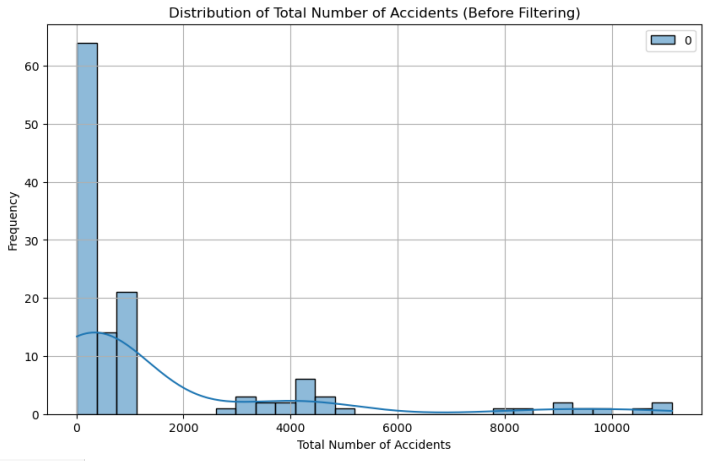
```
+-----------+------------------------+-------------+-----------------+-------------+-------------+--
---------+
|Month - Year|Total Number of Accident|Accident Fatal|Accident Non-Fatal|Person Killed|Person Injured|Total Number of vehicles
Involves|
+-----------+------------------------+-------------+-----------------+-------------+-------------+--
---------+
|  2012-2013|                    8988|         3884|             5104|         4719|         9710|
```

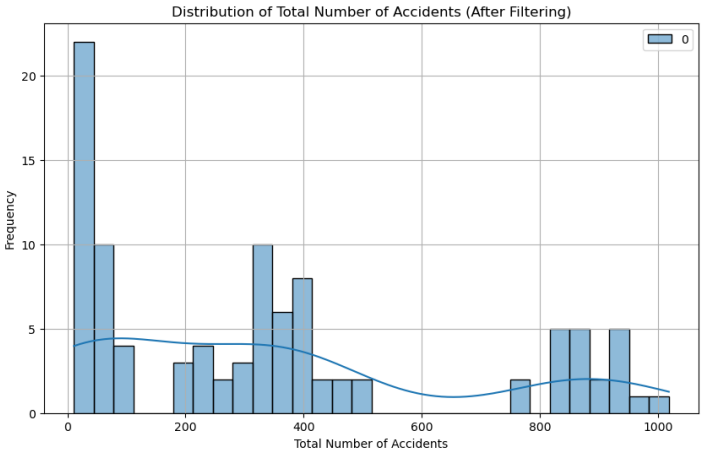
Checking data types and making necessary conversions :

```
# Assuming 'df' is your DataFrame
print(df.dtypes)
```

```
[('Month - Year', 'string'), ('Total Number of Accident', 'int'), ('Accident Fatal', 'int'), ('Accident Non-Fatal', 'int'), ('P
erson Killed', 'int'), ('Person Injured', 'int'), ('Total Number of vehicles Involves', 'int')]
```

Identifying and handling outliers :

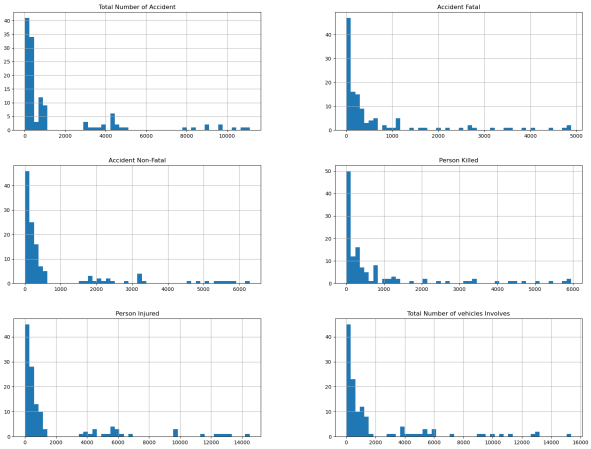




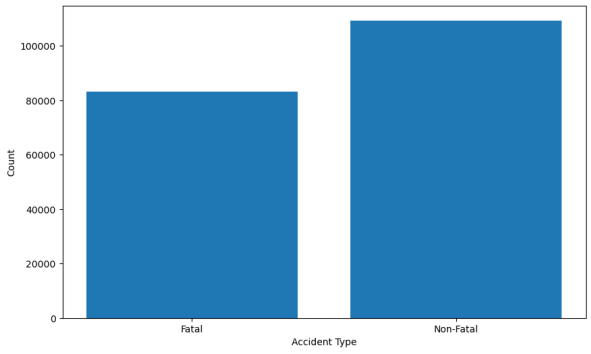
3. Exploratory Data Analysis (EDA)

The process of examining and understanding a dataset to uncover its underlying structure, patterns, and relationships. It involves summarizing the main characteristics of the data, visualizing its distributions and relationships, and identifying any anomalies or interesting features. EDA helps in formulating hypotheses, guiding feature engineering, and informing subsequent modeling decisions. It is an essential step in the data analysis workflow that provides valuable insights into the dataset and lays the groundwork for further analysis and modeling.

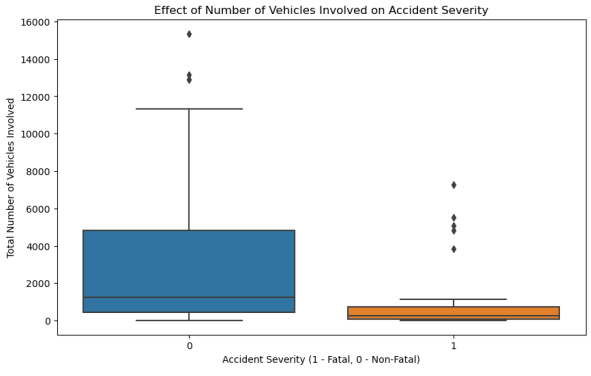
3.1 Analyzing the distribution of features in the dataset



3.2 Investigating the fatal and non-fatal nature of accidents



3.3 Investigating the effect of the number of vehicles involved in the severity of accidents



3.4 Combining features into a single feature vector column

Combining features into a single feature vector column is a crucial step in machine learning and data analysis. It simplifies data representation, making it more manageable and compatible with machine learning algorithms. This process also helps in reducing dimensionality, improving model performance, and facilitating feature engineering. Ultimately, it enhances the interpretability of the data and contributes to the effectiveness and efficiency of machine learning models [5].

3.5 Split the Data

Data splitting is a critical step in machine learning, dividing the dataset into separate subsets for training, validation, and testing. The training set is used to train the model, while the validation set aids in hyperparameter tuning and performance evaluation. Finally, the test set assesses the model's performance on unseen data, ensuring its generalization ability [4]. This process enhances the model's robustness and reliability by providing unbiased estimates of its effectiveness.

— Combining

```
assembler = VectorAssembler(inputCols=["Person_Killed", "Person_Injured"], outputCol="input_features")
df_assembled = assembler.transform(df)
```

— Split

```
# Split the data into training and testing sets
train_data, test_data = df_assembled.randomSplit([0.8, 0.2], seed=42)
```

4. Choose the Model :

Choosing the right model is crucial in machine learning, as it directly impacts the performance and interpretability of the system.[4] Several factors influence model selection, including the nature of the problem, dataset characteristics, and desired level of interpretability. For instance, linear regression may be suitable for continuous variable prediction tasks, while decision trees offer transparency in decision-making processes. The choice also depends on the dataset's complexity and size, with simpler models often preferred for smaller datasets to avoid overfitting. Moreover, interpretability plays a significant role, especially in fields like healthcare or finance, where comprehensible models are favored. Evaluating model performance using appropriate metrics and techniques like cross-validation is essential to ensure generalization to new data and detect issues like overfitting or underfitting. Ultimately, selecting the optimal model involves a trade-off between interpretability, computational complexity, and performance, requiring iterative experimentation and refinement.[5]

4.1 Linear regression

Linear regression is a fundamental statistical method used for modeling the relationship between a dependent variable and one or more independent variables. It assumes a linear relationship between the input variables and the target variable, making it suitable for tasks such as prediction and inference.

In linear regression, the relationship between the independent variables :

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \epsilon$$

où :

- y est la variable dépendante (la variable que nous voulons prédire).
- $x_1, x_2, \ldots, x_n$ sont les variables indépendantes (les caractéristiques).
- $\beta_0, \beta_1, \ldots, \beta_n$ sont les coefficients (paramètres) du modèle.
- ϵ représente le terme d'erreur, qui capture la différence entre les valeurs réelles et prédites.

The goal of linear regression is to estimate the coefficients $\beta_0, \beta_1, \ldots, \beta_n$ that minimize the difference between the observed and predicted values of the dependent variable. This is typically done using the method of least squares, which minimizes the sum of the squared differences between the observed and predicted values.

Linear regression is widely used in various fields, including economics, finance, biology, and social sciences, for tasks such as predicting stock prices, analyzing the impact of variables on outcomes, and modeling relationships between variables.

4.2 Define and Train the Model

```
# Creating Linear Regression model
lr = LinearRegression(labelCol='Person_Killed', featuresCol='input_features')

# Training the model
model = lr.fit(train_data)

# Making predictions on the test set
predictions = model.transform(test_data)

# Displaying the predictions
predictions.select("Person_Killed", "prediction", "input_features").show()

# Stop the Spark session
#spark.stop()

+-------------+------------------+--------------+
```

4.3 Result

Person_Killed	prediction	input_features
7	7.000000000000129	[7.0,5.0]
11	11.000000000000112	[11.0,4.0]
13	13.000000000000105	[13.0,4.0]
20	20.000000000000153	[20.0,56.0]
30	30.00000000000007	[30.0,25.0]
34	34.00000000000007	[34.0,36.0]
56	56.00000000000002	[56.0,52.0]
74	74.00000000000003	[74.0,105.0]
109	109.00000000000065	[109.0,621.0]
112	112.00000000000031	[112.0,399.0]
118	117.9999999999999	[118.0,121.0]
120	120.00000000000041	[120.0,486.0]
131	131.00000000000057	[131.0,617.0]
141	140.9999999999998	[141.0,123.0]
244	243.99999999999994	[244.0,471.0]
288	287.99999999999983	[288.0,498.0]
346	345.99999999999994	[346.0,678.0]
1059	1059.0000000000023	[1059.0,4016.0]
2692	2691.999999999997	[2692.0,4515.0]
3423	3422.999999999996	[3423.0,5916.0]

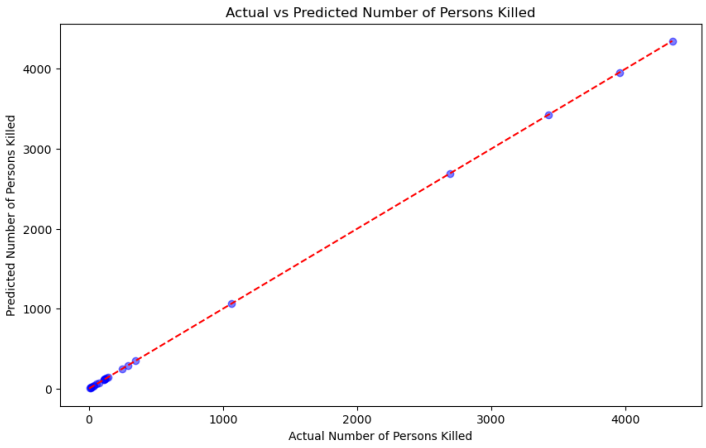
only showing top 20 rows

4.4 Evaluate the model performance

```
Root Mean Squared Error (RMSE) on test data = 1.2924e-12
Mean Absolute Error (MAE) on test data = 6.83776e-13
Mean Squared Error (MSE) on test data = 1.6703e-24
```

The choice to evaluate the model is cru-

cial as it provides insights into its performance and helps assess its effectiveness in real-world applications. By evaluating the model, we can determine its accuracy, reliability, and generalization ability on unseen data. This process allows us to identify potential issues such as overfitting or underfitting and fine-tune the model accordingly to improve its predictive capabilities.



4.5 Visualize the predictions and actual values

The scatter plot visually compares the actual and predicted number of persons killed in accidents. By observing the alignment of the points with the diagonal line, we can assess the accuracy of the regression model. If the points are closer to the diagonal line, it indicates that the model's predictions are closer to the actual values, suggesting a better model performance. Conversely, if the points deviate significantly from the diagonal line, it indicates potential inaccuracies in the predictions. Further analysis and refinement of the model may be required to improve its predictive accuracy.

5. Conclusion

In conclusion, this project demonstrates the value of the integrated approach between machine learning, Big Data analytics, and Spark MLlib in the field of road accident prevention. By leveraging these advanced technologies, we were able to not only provide actionable insights but also evaluate the performance of predictive models using algorithmic metrics such as mean squared error (MSE) and mean absolute error (MAE). The innovative approach of our project, focusing on real-time traffic analysis using predictive models, has enabled the detection of abnormal patterns, thereby contributing to a better understanding of accident dynamics. The results obtained confirm the reliability and effectiveness of our predictive models, paving the way for more targeted preventive interventions and a significant reduction in accident risks on the roads.

Future Work

For future work, it would be beneficial to further explore additional factors contributing to accidents, such as weather conditions, road types, and driver demographics. Additionally, applying more advanced modeling techniques and exploring other datasets related to road safety could provide new insights and improve prediction accuracy.

REFERENCES

[1] Mllib : Main guide - spark 3.3.1 documentation.

[2] Rapport de situation sur la sécurité routière dans le monde 2018. Wikipedia, 2018.

[3] J. Dai, Y. Wang, X. Qiu, D. Ding, Y. Zhang, and Y. Wang. Bigdl : A distributed deep learning framework for big data. *CoRR*, 2018.

[4] X. Meng, J.K. Bradley, B. Yavuz, E.R. Sparks, S. Venkataraman, and D. Liu. Mllib : Machine learning in apache spark. *JMLR*, 17 :34 :1–34 :7, 2016.

[5] E.P. Xing, Q. Ho, W. Dai, J.K. Kim, J. Wei, and S. Lee. Petuum : A new platform for distributed machine learning on big data. In *SIGKDD*, pages 1335–1344, 2015.

[6] C. Yang. Prédiction des accidents de voiture basée sur le cloud computing dans l'analyse des données. In *ISCTT 2022 ; 7e Conférence internationale sur les sciences de l'information, la technologie informatique et les transports*, Xishuangbanna, Chine, 2022.