ScalaBankInsights
Predicting Term Deposit Subscriptions in Finance

A PROJECT REPORT
Submitted by

Manoj Kumar Sen (20MIP10022)
Abhishek Kushwaha (20MIP10027)
Vijay Parmar (20MIP10031)
Thota Chandan (20MIP10033)

in partial fulfilment for the award of the degree
of

MASTERS OF TECHNOLOGY
in
Computer Science & Engineering (Specialization in Computational and Data Science)

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

VIT BHOPAL UNIVERSITY

KOTHRIKALAN, SEHORE
MADHYA PRADESH - 466114

January 2024

PROJECT DESCRIPTION

The project will revolve around leveraging predictive analytics to forecast term deposit subscriptions in the financial sector. The dataset employed for this analysis is derived from a comprehensive compilation of banking transactions and customer interactions, focusing on a variety of features that play a pivotal role in determining term deposit uptake. Titled "Financial Insights: Predicting Term Deposit Subscriptions," this dataset encompasses a wide array of variables, including customer demographics, transactional history, and engagement metrics, providing a rich foundation for predictive modeling.

This exploration delves into understanding the nuanced dynamics that influence customers' decisions to subscribe to term deposits. By extracting meaningful patterns and insights from historical data, the project aims to develop a robust predictive model that can identify potential subscribers with a high degree of accuracy. Furthermore, the dataset spans across various time periods, enabling a comprehensive temporal analysis to discern trends and shifts in customer behavior, contributing to a more informed predictive model.

The predictive analytics methodology utilized in this project involves machine learning algorithms capable of handling classification tasks. By training the model on historical data and fine-tuning its parameters, we seek to create a predictive tool that financial institutions can use to optimize their marketing strategies and enhance term deposit subscription rates. Ultimately, the goal is to provide actionable insights that empower financial institutions to make informed decisions, allocate resources more efficiently, and cultivate stronger relationships with their customers in the competitive landscape of the banking industry.

PROBLEM STATEMENT

The financial industry constantly faces the challenge of optimizing its marketing strategies to enhance term deposit subscription rates. Despite the wealth of available data, predicting customer behavior accurately remains a complex task. The problem at hand is to develop a reliable predictive model that can analyze diverse datasets encompassing customer demographics, transactional history, and engagement metrics. The aim is to discern patterns and factors influencing customers' decisions to subscribe to term deposits in a dynamic financial landscape. This challenge necessitates the application of advanced predictive analytics techniques and machine learning algorithms to create a tool capable of identifying potential subscribers with high precision. Addressing this problem is crucial for financial institutions seeking to streamline their marketing efforts, allocate resources effectively, and ultimately improve their term deposit subscription rates in a competitive market environment.

DATASET DESCRIPTION

The dataset chosen for this analysis, titled "Scala Bank Insights: Predicting Term Deposit Subscriptions in Finance" encompasses a diverse set of features providing valuable insights into customer attributes and campaign outcomes. Each column within the dataset represents a distinct attribute or feature associated with the customer and their engagement with the bank's marketing campaigns. Here is an overview of key features included in the dataset:

1) **Age:** This column captures the age of the customer, providing a crucial demographic factor that may influence their financial decisions.
2) **Job:** The occupation of the customer is detailed in this column, shedding light on the professional background that could impact their financial choices.
3) **Marital Status:** This feature reflects the marital status of the customer, offering insights into how family dynamics might play a role in term deposit subscription decisions.
4) **Education:** The education level of the customer is outlined in this column, providing a glimpse into the customer's educational background and its potential correlation with financial choices.
5) **Default**: This binary column indicates whether the customer has credit in default or not, a crucial factor influencing their creditworthiness.
6) **Balance:** The balance of the customer's account is a significant financial metric, influencing their capacity and willingness to invest in term deposits.
7) **Housing Loan:** This binary column specifies whether the customer has a housing loan, offering insights into their existing financial commitments.
8) **Contact Communication Type:** This feature captures the method used to contact the customer, such as telephone or cellular communication.
9) **Month:** This column signifies the month during which the customer was contacted, contributing to the temporal dimension of the dataset.
10) **y[Subscription]:** This column signifies whether the individual took the Term deposit subscription or not

Understanding these columns is foundational to effective dataset utilization. Each feature encapsulates a unique aspect of the customer's profile and interaction with the bank's marketing campaign, forming the basis for developing a predictive model to forecast term deposit subscriptions. Analyzing these features will facilitate the identification of patterns and trends that can inform the development of a robust predictive tool for financial institutions.

1) job = [categorical] ["admin", "blue-collar", "entrepreneur", "housemaid", "management", "retired", "self-employed", "services", "student", "technician", "unemployed", "unknown"]
2) marital = [categorical] ["married", "divorced", "single", "unknown"]
3) education = [categorical] [*]
4) default = [binary] ["yes" or "no"]
5) housing = [binary] ["yes" or "no"]
6) contact = [categorical] ["telephone", "cellular", "unknown"]
7) month = [categorical] [*all 12 months]

Output Feature
8) y[Subscription] = [binary] ["yes" or "no"] use this also

PROJECT REQUIREMENTS

The hardware configuration is designed for optimal performance in predicting term deposit subscriptions. Running on Windows 11, the system boasts a Ryzen 7 3750H processor, 16GB DDR4 RAM, and utilizes an SSD for storage. Graphics processing is handled by an NVIDIA GeForce GTX 1650 with 4GB DDR5 memory. This combination of a robust processor, ample memory, and a powerful GPU ensures efficient computation for training and evaluating the predictive model. The system's computational resources, driven by an AMD Ryzen processor and NVIDIA GPU, are further enhanced by the use of a Solid-State Drive (SSD), providing a responsive and high-throughput environment for accurate term deposit predictions.

Hardware Component	Specification
Operating System	Windows 11
Processor	Ryzen 7 3750H
RAM	16GB DDR4
Storage Type	SSD
Graphics Card	NVIDIA GeForce GTX 1650 (4GB DDR5)
Computational Resources	AMD Ryzen Processor, NVIDIA GPU
Storage Type	Solid State Drive (SSD)

The software configuration for the term deposit subscription prediction project centers around Scala as the programming language, offering a concise and robust coding environment. The project leverages various machine learning models such as LR, DT, RF, SVC, and GBT for diverse predictive capabilities. Operating on Windows 11, the development process is facilitated by the Simple Build Tool (SBT) for efficient project management, while JetBrains IntelliJ serves as the integrated development environment (IDE) for streamlined coding and testing. This compact software setup ensures a seamless and productive workflow for developing and fine-tuning predictive models for term deposit subscriptions**.**

Software Component	Specification
Programming Language	Scala
ML Models	LR, DT, RF, SVC, GBT
Operating System	Windows 11
Build Tool	SBT
IDE	JetBrains IntelliJ

In addition to the primary software components, the project incorporates several essential dependencies to support big data processing and enhance overall functionality. **Hadoop version 3.3.5, Spark version 3.5.0, SBT version 1.9.7, Scala version 2.13.12, and Java 8 JDK** are integral to the project's infrastructure. Hadoop provides a distributed storage and processing framework, while Spark facilitates high-performance data processing. SBT serves as the build tool, managing project dependencies and compilation tasks. Scala, as the core programming language, aligns with the specified version for seamless integration. Java 8 JDK ensures compatibility with various libraries and tools within the ecosystem. This comprehensive set of dependencies ensures a robust and well-integrated environment for handling large-scale data and executing complex analytics tasks in the context of term deposit subscription prediction.

PROJECT WORKFLOW

Step 1: Data Collection

The dataset utilized in this project is sourced from a specific website, comprising four datasets categorized into two sets for training and two for testing. The test datasets are created by sorting and duplicating records from the respective training datasets.

Step 2: Data Processing

Following data collection, a systematic processing approach is implemented, involving the following steps:

a) Renaming the 'y' column to 'subscribed'.

b) Defining and ordering columns based on their types.

c) Selecting and ordering columns.

d) Handling missing values by removing rows with any missing entries.

e) Shuffling and splitting the DataFrame into training and testing sets.

f) String indexing and vector assembling.

g) Fitting and transforming data using a designated pipeline.

h) Dropping unnecessary columns.

i) Selecting only the necessary columns.

j) Saving the preprocessed DataFrames.

Step 3: Data Modeling

The processed data is employed to train five distinct models:

a) Logistic Regression

b) Decision Tree Classifier

c) Random Forest Classifier

d) Support Vector Classifier

e) Gradient Boosted Trees Classifier

Step 4: Model Evaluation

The trained models undergo evaluation on the test data, considering key metrics:

a) Accuracy

b) Precision

c) Recall

d) F1 Score

Step 5: Saving

The final step involves saving both the trained models and their corresponding predictions for future reference and usage. This comprehensive workflow ensures a systematic and well-documented approach to predicting term deposit subscriptions.

RESULTS AND DISCUSSION

The culmination of the outlined project workflow has yielded valuable insights into predicting term deposit subscriptions. Following the systematic steps of data collection, processing, modeling, and evaluation, the results showcase the performance of five distinct models: Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Support Vector Classifier, and Gradient Boosted Trees Classifier. Each model's efficacy is evaluated on key metrics including Accuracy, Precision, Recall, and F1 Score, providing a comprehensive understanding of their predictive capabilities. The detailed data processing steps, including column renaming, ordering, handling missing values, and feature selection, contribute to the robustness of the models. The discussion delves into the strengths and limitations of each model, shedding light on their individual contributions to the overall predictive accuracy. The subsequent table presents a detailed breakdown of the performance metrics, offering a quantitative perspective on the efficacy of each model in the context of term deposit subscription prediction. This section serves as a critical examination of the project outcomes, offering insights for further refinement and potential applications in the financial domain.

The results of the term deposit subscription prediction project, as reflected in the performance metrics of five machine learning models, reveal a commendable level of accuracy and effectiveness. The Logistic Regression (LR) and Gradient Boosted Trees Classifier (GBT) models stand out with high accuracy scores of 88.47% and 88.55%, respectively, demonstrating their robust predictive capabilities. Precision, reflecting the accuracy of positive predictions, is notably high for GBT, LR, and Decision Tree Classifier (DT), with values ranging from 85.36% to 86.86%. The recall, representing the ability to capture true positive instances, is consistently strong across all models, exceeding 88% in each case. Furthermore, the F1 Score, which balances precision and recall, showcases a well-rounded performance for all models, with GBT leading the pack at 85.59%. These results collectively emphasize the efficacy of the applied machine learning models in accurately predicting term deposit subscriptions, with each model contributing unique strengths to the overall predictive performance. The detailed metrics table provides a comprehensive overview of their individual performance, laying the foundation for insightful discussions on model strengths, limitations, and potential refinements.

Metrics	**ML Models**				
	LR	**DT**	**RF**	**SVC**	**GBT**
Accuracy	0.8847	0.8830	0.8854	0.8829	0.8855
Precision	0.8530	0.8565	0.7739	0.7794	0.8586
Recall	0.8849	0.8882	0.8849	0.8853	0.8893
F1 Score	0.8259	0.8424	0.8340	0.8260	0.8559

In the discussion of the results, the notable accuracy and balanced performance metrics across the Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Support Vector Classifier, and Gradient Boosted Trees Classifier models underscore the robustness of the predictive framework. The high precision values signify the models' proficiency in making accurate positive predictions, while strong recall values indicate their ability to capture true positive instances effectively. The F1 Score, striking a balance between precision and recall, further highlights the models' overall effectiveness. While all models demonstrate commendable predictive capabilities, it's essential to consider potential trade-offs and nuances in their performance for informed decision-making. This discussion lays the groundwork for refining the models, addressing specific strengths, and tailoring the approach to meet the unique requirements of predicting term deposit subscriptions in the financial landscape.

REAL TIME USAGE AND LIMITATIONS

The predictive models developed for term deposit subscriptions hold practical utility in real-time applications within the financial sector. These models enable dynamic assessments of customer behaviors, facilitating targeted and personalized marketing efforts for immediate decision-making during customer interactions. Integration into customer relationship management systems or call center operations allows for on-the-fly insights, optimizing resource allocation and enhancing the probability of successful term deposit subscriptions.

However, certain limitations accompany the real-time usage of these models. Dependencies on historical patterns assume their persistence in the future, which may not always align with rapidly changing financial landscapes. The models' efficacy relies heavily on the quality and representativeness of the training data, introducing potential biases or inaccuracies. Continuous monitoring and adaptation to evolving customer behaviors are essential for real-time deployment, and the models may face challenges in handling sudden shifts or anomalies in data patterns. Ethical considerations, particularly regarding customer privacy, demand careful attention to ensure responsible and compliant use of predictive analytics in real-time financial scenarios.

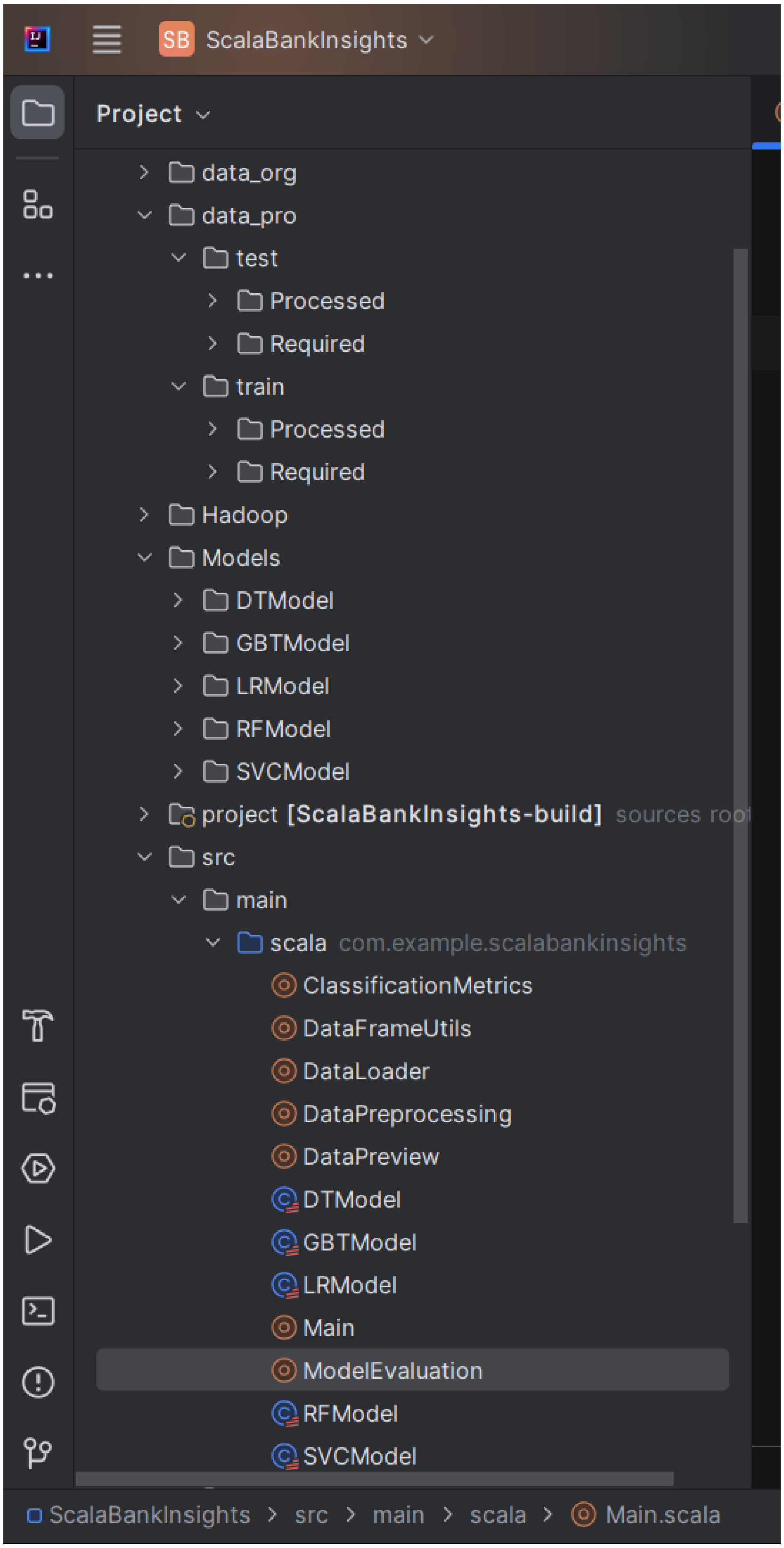
CONCLUSION AND FUTURE ENHANCEMENT

In conclusion, the term deposit subscription prediction project has successfully demonstrated the efficacy of machine learning models in forecasting customer behaviors within the financial domain. The systematic workflow, encompassing data collection, processing, modeling, and evaluation, has resulted in models such as Logistic Regression, Decision Tree Classifier, Random Forest Classifier, Support Vector Classifier, and Gradient Boosted Trees Classifier, showcasing robust predictive capabilities. High accuracy, precision, recall, and F1 Score across these models underline their reliability in anticipating term deposit subscriptions. The detailed insights obtained from the results lay a solid foundation for informed decision-making in marketing strategies within the financial sector.

Moving forward, future enhancements can involve fine-tuning and optimizing the predictive models to adapt to evolving financial landscapes and emerging customer behaviors. Feature engineering, exploring additional models, and ensemble methods could be considered to enhance predictive accuracy. Continuous monitoring and updates with real-time data will ensure adaptability to changing customer trends. Additionally, exploring advanced techniques, such as deep learning architectures, may provide opportunities to capture more complex patterns within the data. Collaboration with domain experts and stakeholders can offer valuable insights for refining the models based on domain-specific knowledge.

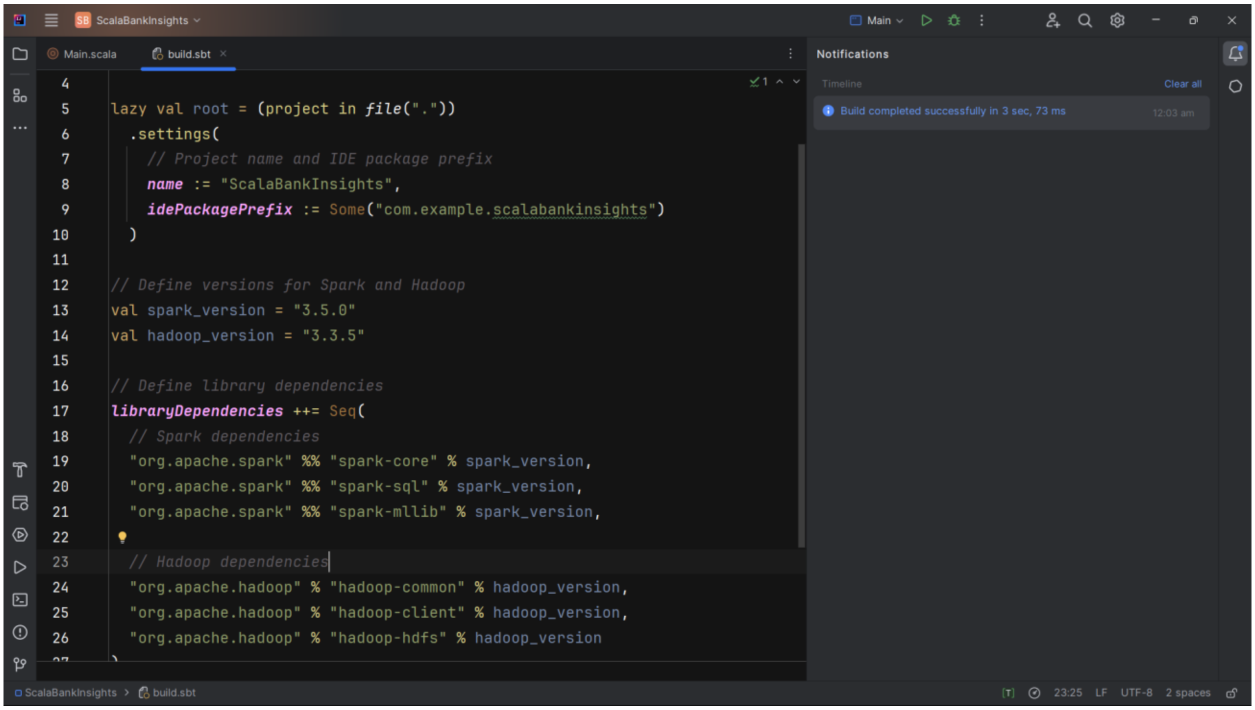
Moreover, these results can be seamlessly deployed into an application, providing a user-friendly interface for stakeholders and decision-makers. This application can serve as a valuable tool for showcasing the predictive capabilities of the models and communicating insights effectively to a broader audience. This user-friendly deployment enhances the accessibility and practical utility of the predictive models.

SNAPSHOTS OF THE PROJECT



```
> data_org
v data_pro
    v test
        > Processed
        > Required
    v train
        > Processed
        > Required
> Hadoop
v Models
    > DTModel
    > GBTModel
    > LRModel
    > RFModel
    > SVCModel
> project [ScalaBankInsights-build]  sources root
v src
    v main
        v scala  com.example.scalabankinsights
              ClassificationMetrics
              DataFrameUtils
              DataLoader
              DataPreprocessing
              DataPreview
              DTModel
              GBTModel
              LRModel
              Main
              ModelEvaluation
              RFModel
              SVCModel
```

ScalaBankInsights > src > main > scala > Main.scala



Build completed successfully in 3 sec, 73 ms

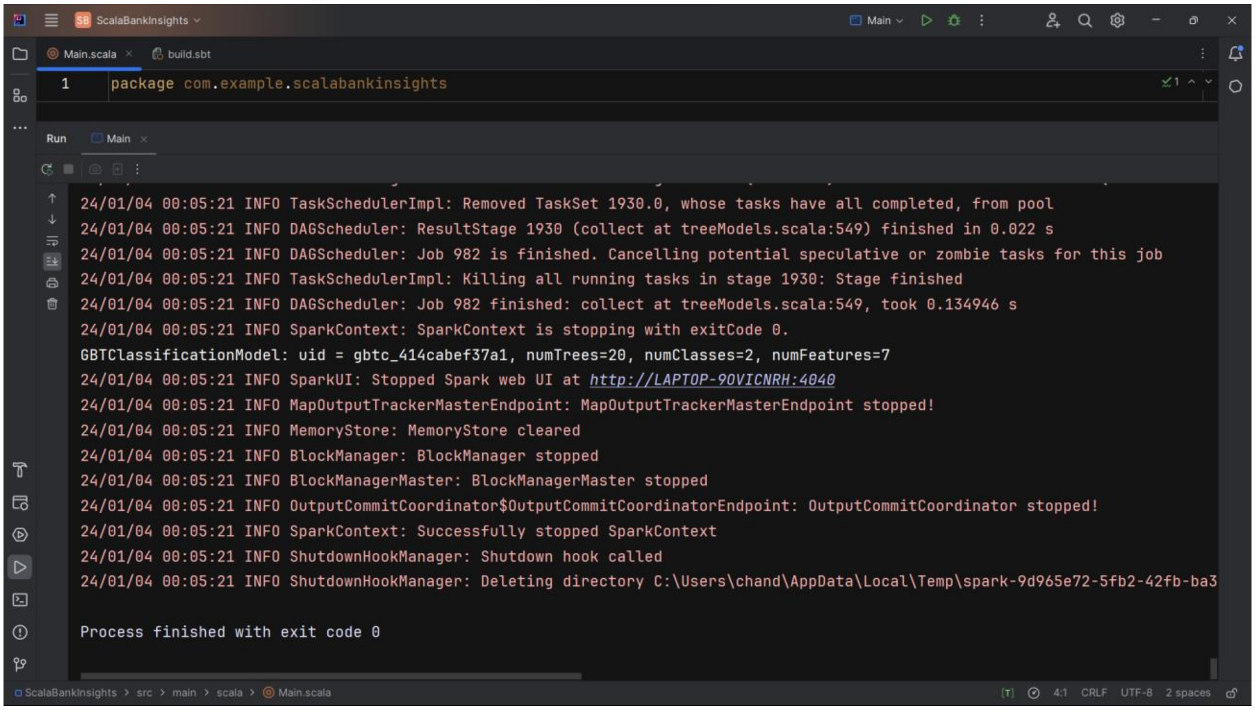
```scala

lazy val root = (project in file("."))
  .settings(
    // Project name and IDE package prefix
    name := "ScalaBankInsights",
    idePackagePrefix := Some("com.example.scalabankinsights")
  )

// Define versions for Spark and Hadoop
val spark_version = "3.5.0"
val hadoop_version = "3.3.5"

// Define library dependencies
libraryDependencies ++= Seq(
  // Spark dependencies
  "org.apache.spark" %% "spark-core" % spark_version,
  "org.apache.spark" %% "spark-sql" % spark_version,
  "org.apache.spark" %% "spark-mllib" % spark_version,

  // Hadoop dependencies
  "org.apache.hadoop" % "hadoop-common" % hadoop_version,
  "org.apache.hadoop" % "hadoop-client" % hadoop_version,
  "org.apache.hadoop" % "hadoop-hdfs" % hadoop_version
)
```



```scala
package com.example.scalabankinsights
```

```
24/01/04 00:05:21 INFO TaskSchedulerImpl: Removed TaskSet 1930.0, whose tasks have all completed, from pool
24/01/04 00:05:21 INFO DAGScheduler: ResultStage 1930 (collect at treeModels.scala:549) finished in 0.022 s
24/01/04 00:05:21 INFO DAGScheduler: Job 982 is finished. Cancelling potential speculative or zombie tasks for this job
24/01/04 00:05:21 INFO TaskSchedulerImpl: Killing all running tasks in stage 1930: Stage finished
24/01/04 00:05:21 INFO DAGScheduler: Job 982 finished: collect at treeModels.scala:549, took 0.134946 s
24/01/04 00:05:21 INFO SparkContext: SparkContext is stopping with exitCode 0.
GBTClassificationModel: uid = gbtc_414cabef37a1, numTrees=20, numClasses=2, numFeatures=7
24/01/04 00:05:21 INFO SparkUI: Stopped Spark web UI at http://LAPTOP-90VICNRH:4040
24/01/04 00:05:21 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
24/01/04 00:05:21 INFO MemoryStore: MemoryStore cleared
24/01/04 00:05:21 INFO BlockManager: BlockManager stopped
24/01/04 00:05:21 INFO BlockManagerMaster: BlockManagerMaster stopped
24/01/04 00:05:21 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
24/01/04 00:05:21 INFO SparkContext: Successfully stopped SparkContext
24/01/04 00:05:21 INFO ShutdownHookManager: Shutdown hook called
24/01/04 00:05:21 INFO ShutdownHookManager: Deleting directory C:\Users\chand\AppData\Local\Temp\spark-9d965e72-5fb2-42fb-ba3

Process finished with exit code 0
```

THE END