ISSN 2277-2685
IJESR/April-June. 2025/ Vol-15/Issue-2s/1293-1300
M.Varshitha *et. al.*, / International Journal of Engineering & Science Research

Predicting Product Demand Using Batch-Processed Data In Azure Databricks Ml

Amtul Shanaz[1], M.Varshitha[2], CH.Venkata Vardhini[3]
[1] Assistant Professor, Department of CSE, Bhoj Reddy Engineering College for Women, India.
[2,3]B. Tech Students, Department of CSE, Bhoj Reddy Engineering College for Women, India.

ABSTRACT
Accurate demand forecasting is critical for inventory optimization, cost reduction, and customer satisfaction in data-driven enterprises. Traditional statistical approaches often fall short in managing high-volume data, handling complex seasonality, or incorporating external factors. This project addresses these limitations by building a scalable and automated demand forecasting system using Azure Databricks. Leveraging its Apache Spark-powered batch processing engine, Delta Lake for reliable data storage, and machine learning frameworks like MLflow, the system enables efficient data preprocessing, model training, and deployment. By analyzing historical sales data, the model predicts future product demand with improved accuracy and reduced manual effort, making it suitable for modern enterprise applications in retail and supply chain management.

1 INTRODUCTION

This project delves into the strategic use of batch-processed data within **Azure Databricks** to develop accurate **product demand forecasting models** for large-scale datasets. In today's data-driven world, where e-commerce platforms and retail giants handle **enormous volumes of transactional data**,there is an urgent need to transform raw information into actionable insights that can support timely business decisions.

This project addresses this need by harnessing **Azure Databricks' powerful integration of Apache Spark** with Microsoft's cloud ecosystem to create an end-to-end demand prediction pipeline.The core of this project revolves around the analysis of **historical sales data**, encompassing various attributes such as order frequency, product categories, customer purchase behavior, and seasonal trends.. With **in-memory computation, parallel processing, and autoscaling clusters**, Azure Databricks proves to be an ideal platform for building and executing machine learning models on large-scale datasets without performance bottlenecks.

By implementing **predictive algorithms** such as Random Forest, Decision Trees, and XGBoost, the project forecasts future product demand with high accuracy. These models learn from historical sales patterns to predict the quantity of each product likely to be demanded in upcoming cycles. Such insights are essential for **stock optimization, reducing overstock or understock risks, minimizing holding costs**, and improving overall customer satisfaction through better availability.

To enhance interpretability, the project incorporates **data visualization** tools and dashboards using platforms like **Power BI or Databricks notebooks**, enabling stakeholders to easily view demand forecasts, seasonal patterns, and category-level trends.

Overall, this project showcases a **robust, scalable, and cloud-native approach** to predictive analytics using **Azure Databricks ML**. It not only emphasizes the benefits of batch-processing architecture in big data environments but also presents a practical solution for **forecasting product demand**, leading to **smarter inventory decisions, optimized operations, and enhanced profitability** for modern businesses

2-LITERATURE SURVEY

Title: Designing Scalable Data Engineering Pipelines Using Azure and Databricks
Author: Santosh Kumar Singu
Year: 2021
Description: This paper explores the design and implementation of scalable data engineering pipelines using the robust capabilities of Azure and Apache Databricks. The author presents a systematic approach to tackling the challenges that arise while handling large-scale data workloads, particularly in industries like retail and e-commerce, where massive volumes of structured and semi-structured data need to be processed daily.The paper emphasizes the architecture of data pipelines that incorporate Azure Data Lake for cost-effective storage, Azure Data Factory for orchestration, and Databricks for processing and transformation. Spark jobs are optimized through parallel processing, schema evolution handling, and adaptive execution planning, which is especially useful for iterative machine learning workflows and batch data processing.An end-to-end pipeline is demonstrated that includes data ingestion, cleaning, transformation, and finally, data modeling for predictive analytics. This modular approach is particularly suitable for demand forecasting systems where periodic

ingestion of sales logs and customer activity logs is critical. The author showcases how Azure's autoscaling and Databricks' cluster configuration features can significantly reduce infrastructure overhead while maintaining high performance and fault tolerance.This paper contributes a valuable perspective on using cloud-native tools in modern data engineering workflows, particularly relevant to businesses aiming to predict product demand using batch-processed pipelines in Azure Databricks

Title: Predicting the Ratings of Amazon Products Using Big Data
Author: Jongwook Woo, Monika Mishra
Year: 2020
Description: This research explores how predictive modeling techniques can be employed on large-scale datasets to forecast product ratings on e-commerce platforms like Amazon. The authors use a real-world dataset of user reviews, product metadata, and star ratings, which are often used as indicators of product demand and consumer sentiment. The dataset is massive, consisting of millions of records, which necessitates the use of big data tools to handle storage, preprocessing, and model training.The study leverages Oracle Big Data and Azure Cloud services to preprocess and transform the data. It then implements machine learning models, including Random Forest, Gradient Boosting, and Decision Trees, to predict product ratings. The models are trained on various features like product category, user behavior, and review length, making them suitable for capturing nonlinear relationships between different factors affecting customer satisfaction and product demand.Azure's scalable infrastructure allows parallel training and testing of these models, reducing processing time and improving prediction accuracy. While the study primarily focuses on rating prediction, it has strong implications for product demand forecasting, as high ratings often correlate with increased demand.This paper is instrumental in showing how big data and machine learning, when paired with Azure's ecosystem, can be used to forecast product-related metrics, offering strong relevance to any demand prediction model designed using batch-processed data

Title: Demand Forecasting with Machine Learning
Author: K Chang
Year: 2024
Description: In this study, the focus is placed on implementing machine learning for demand forecasting in supply chain and inventory management. The research outlines a step-by-step methodology beginning with defining the business problem, understanding the characteristics of available data, and applying traditional forecasting techniques like ARIMA and exponential smoothing as a baseline.Once the baselines are established, the study transitions into machine learning territory, employing models such as XGBoost, LSTM (Long Short-Term Memory networks), and Random Forest to improve forecasting precision. The models are evaluated on their ability to handle temporal data, detect seasonality, and adapt to changing trends, which are essential components in accurately predicting product demand.A key contribution of this work is its emphasis on model evaluation and the importance of feature engineering. The study highlights how temporal features, promotional events, and even weather data can significantly enhance the forecasting capability. The implementation is performed in a cloud-based environment using services compatible with Azure Databricks, emphasizing scalability, automation, and integration with batch-processing pipelines.
This paper serves as a contemporary reference for using intelligent systems to forecast demand, aligning beautifully with the goals of projects focusing on predictive analytics in Azure Databricks ML. It emphasizes not just prediction accuracy but also the importance of efficient pipeline design and processing architecture

3-DESIGN AND DEVELOPMENT

Design and development cover a wide spectrum of activities, beginning with the initial concept and planning, followed by prototyping, testing, and final product delivery. This process includes problem identification, exploring potential solutions, and continuously refining through iterative design and development cycles to ensure the end product aligns with user requirements and business objectives

EXISTING SYSTEM

In the existing system, Azure Databricks is utilized for handling large workloads efficiently using Apache Spark. It supports batch and real-time data processing, provides shared workspaces for team collaboration, and integrates seamlessly with Azure services. Azure Databricks also supports popular programming languages such as Python and SQL for fast and efficient data processing.
The traditional approach to processing large-scale e-commerce data involves collecting data using ETL frameworks or custom scripts and storing it in RDBMS systems like MySQL or distributed storage like HDFS. Data cleaning and transformation are done using SQL or standalone tools, often requiring complex queries. Batch processing with tools like Apache Hive or Hadoop MapReduce handles analysis, but the reliance on disk operations makes it slow. Insights are derived from static datasets using BI tools, limiting real-

time analytics. Machine learning is typically applied in isolated environments on static data, restricting scalability and integration.
This approach faces challenges such as poor scalability, high latency, resource-intensive maintenance, and a lack of flexibility for real-time analytics. These limitations highlight the need for modern solutions like PySpark for efficient, scalable, and real-time data processing.

PROPOSED SYSTEM

The proposed system leverages PySpark, the Python API for Apache Spark, to optimize data processing pipelines for large-scale e-commerce datasets. Unlike traditional approaches, this system harnesses the power of in-memory distributed computing to process massive datasets with low latency and high efficiency. By implementing tailored algorithms, it ensures effective data transformation and preparation specific to e-commerce analytics. The system supports real-time data processing using Spark Streaming, enabling immediate insights into customer behavior, sales trends, and inventory needs.
PySpark enables the pipeline to handle vast datasets seamlessly by distributing the workload across multiple nodes, making it scalable and well-suited for growing e-commerce data demands. The system is designed to streamline data transformation and cleaning processes, using advanced algorithms that are optimized for e-commerce-specific needs, such as revenue analysis, customer behavior tracking, and product popularity assessments.
This open-source solution also reduces reliance on proprietary platforms, making it cost-effective and adaptable across different environments. Overall, the proposed system offers enhanced performance, scalability, and flexibility, addressing the shortcomings of traditional methods while enabling businesses to make data-driven decisions in real time.

4-DESIGN ENGINEERING

Design Engineering deals with the various UML [Unified Modelling language] diagrams for the implementation of project. Design is a meaningful engineering representation of a thing that is to be built. Software design is a process through which the requirements are translated into representation of the software. Design is the place where quality is rendered in software engineering. Design is the means to accurately translate customer requirements into finished products.

SYSTEM ARCHITECTURE

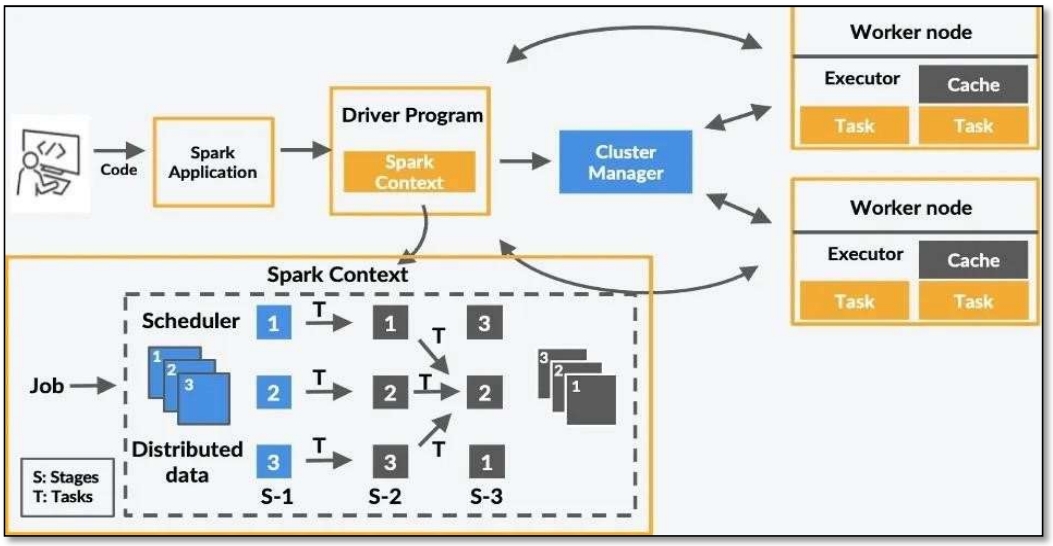


Figure - 4.2: SYSTEM ARCHITECTURE

Apache Spark is a distributed computing framework that processes large datasets efficiently across a cluster of machines. At the core of its architecture is the Driver Program, which acts as the brain of the Spark application. The driver breaks the big job (like analyzing data) into smaller tasks, manages them, and communicates with the Cluster Manager to request resources like CPU and memory.
The Cluster Manager coordinates with multiple machines, called Worker Nodes, where the actual work happens. Each worker node runs Executors, which execute tasks and temporarily store data. Spark divides the job into Stages, and these stages are further split into Tasks, which operate on smaller chunks of data called partitions. This division allows Spark to process data in parallel, making it highly scalable and fast.
Data and tasks flow between the Driver Program, Cluster Manager, and Worker Nodes, enabling

efficient execution. To optimize performance, Spark caches intermediate data in memory, avoiding repetitive computations. This combination of parallel processing and caching makes Spark powerful for large-scale data processing, such as filtering, aggregating, and analyzing datasets.

DATA FLOW DIAGRAM

Level 1

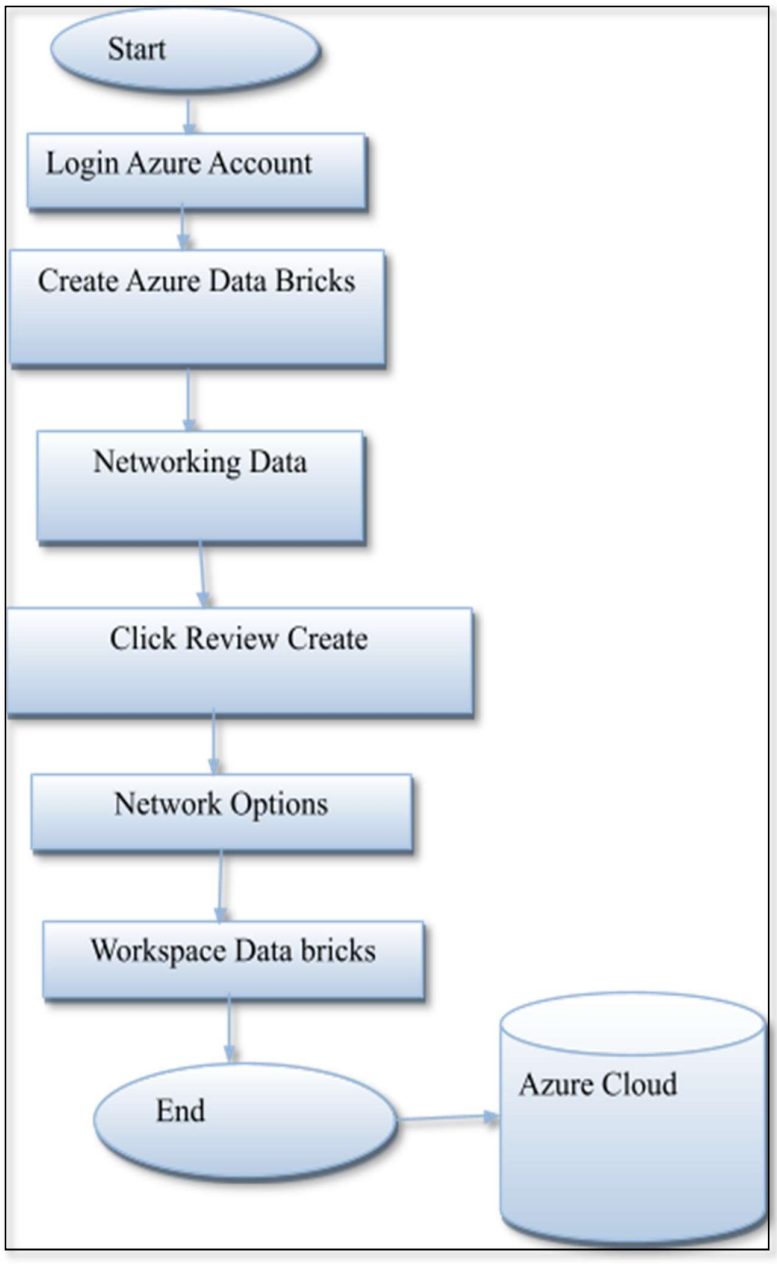


Figure - 4.4: DATA FLOW DIAGRAM

The refined representation of a process can be done in another data-flow diagram, which subdivides this process into sub-processes. The data-flow diagram is a tool that is part of structured analysis and data modeling. When using UML, the activity diagram typically takes over the role of the dataflow diagram.

5-SNAPSHOTS

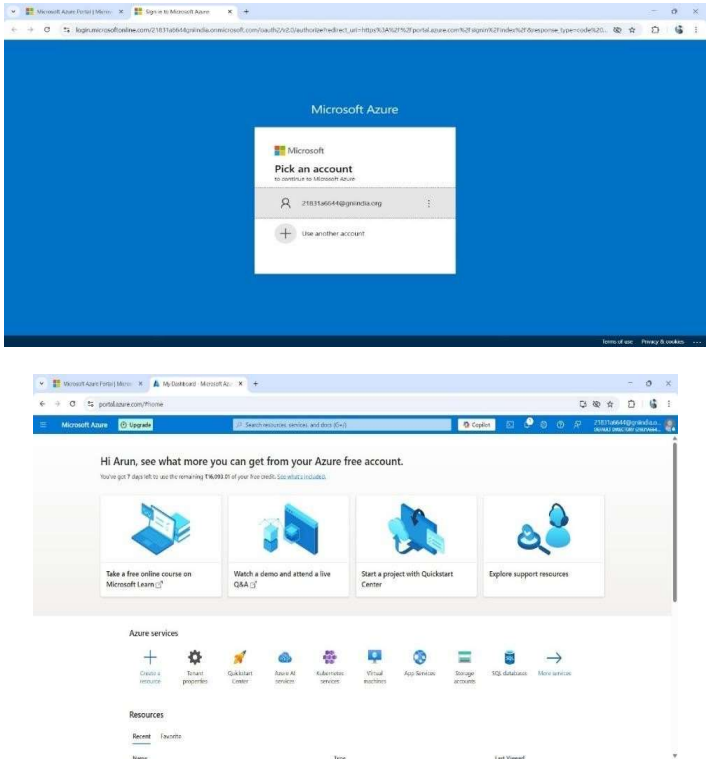
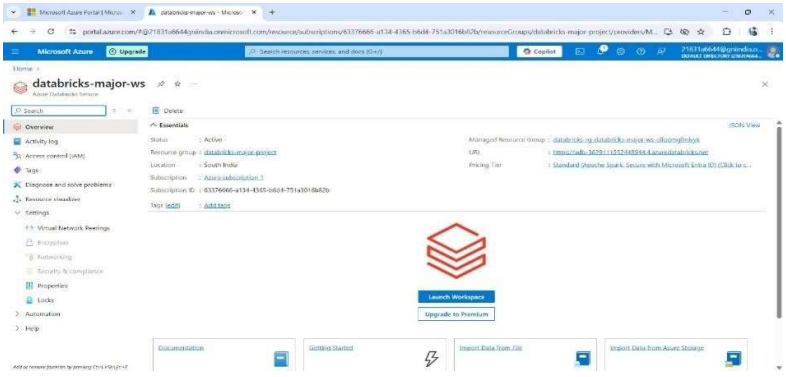


Fig 1 **:**Login to Azure and access Databricks service



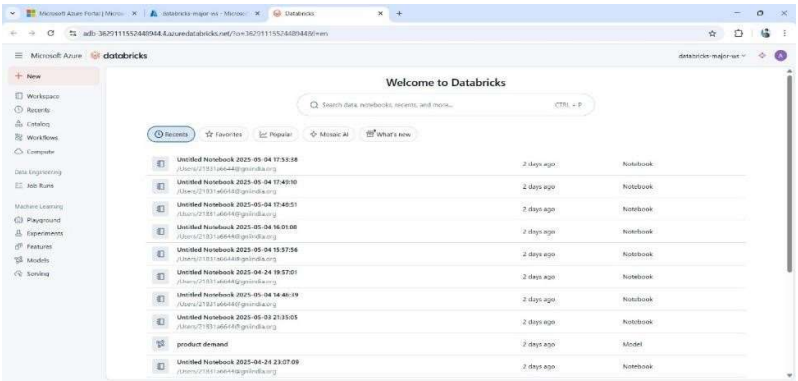
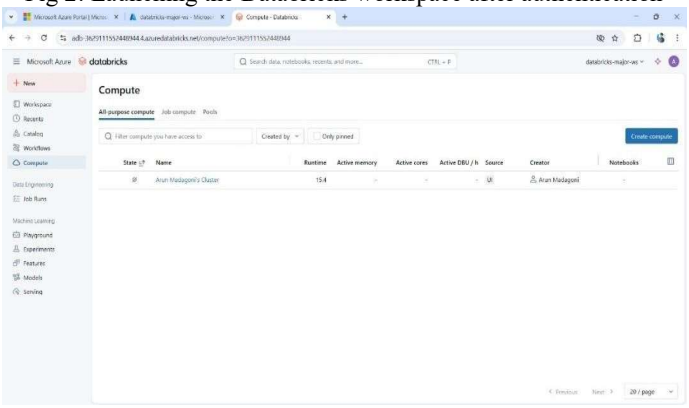


Fig 2: Launching the Databricks workspace after authentication



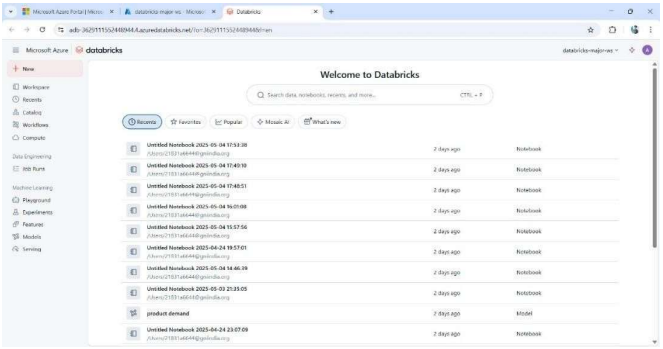
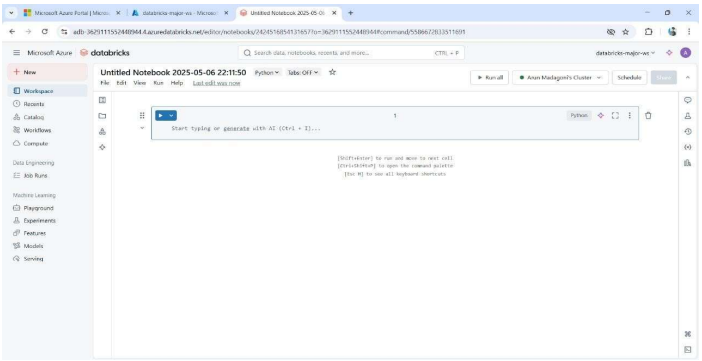


Fig 3: Creating and Running a Databricks Cluster



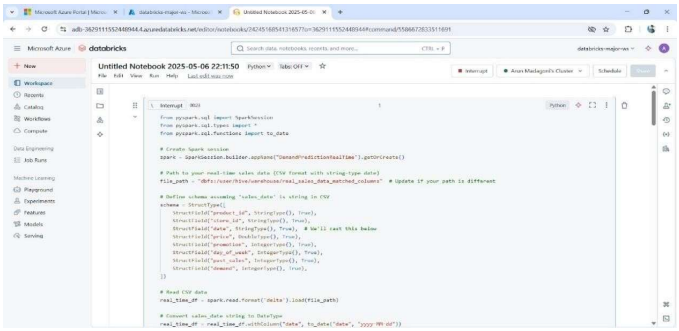
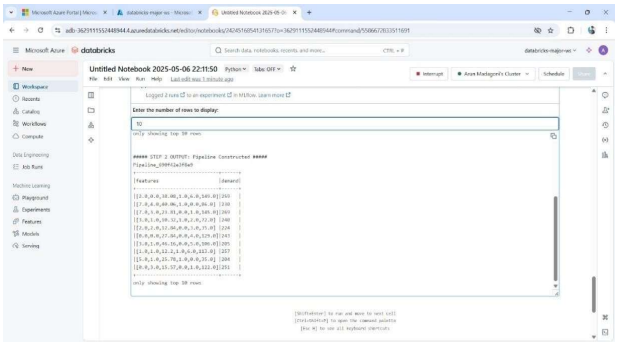


Fig 4 : Creating and opening a new notebook inside the workspace



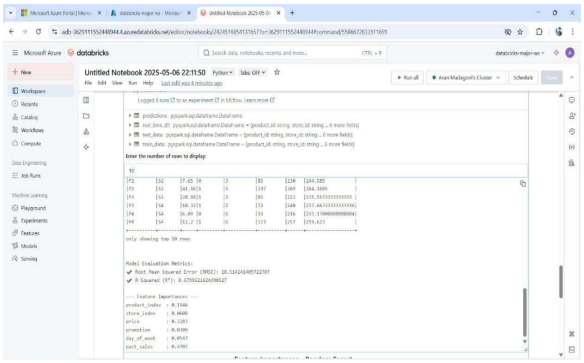


Fig 5 : Executing Code Cell-by-Cell

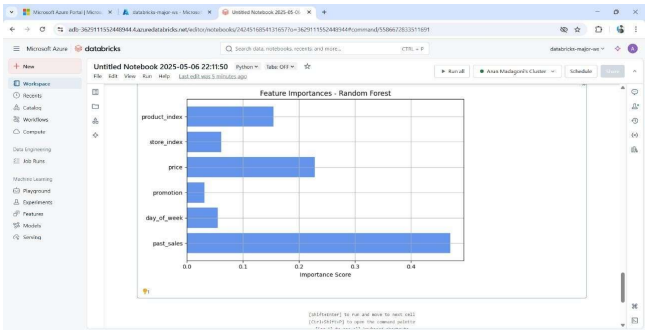


Fig 6 : Final visualization showcasing product popularity trends

6-CONCLUSION

The project successfully demonstrates how Azure Databricks, combined with modern machine learning techniques and batch-processing architectures, can significantly enhance the accuracy and scalability of product demand forecasting. By integrating tools like Apache Spark, Delta Lake, and MLflow, the system achieves efficient data handling, reliable model tracking, and reproducible results across massive datasets. Unlike traditional forecasting models that often fail under complexity or scale, this modern approach proves robust against data volume, missing values, and dynamic demand patterns. The predictive insights generated can empower businesses to optimize inventory, reduce waste, and meet customer expectations more effectively. This project paves the way for scalable and intelligent demand forecasting in datadriven industries such as retail, manufacturing, and logistics.

REFERENCES

1. Jagadish, H. V., Gehrke, J., Labrinidis, A., Papakonstantinou, Y., Patel, J. M., Ramakrishnan, R., & Shahabi, C., Big data and its technical challenges.
Communications of the ACM (2014).
2. Dean, J., & Ghemawat, S. , MapReduce: Simplified data processing on large clusters.
Communications of the ACM, (2008).
3. Breiman, L. ,Random forests. Machine Learning, (2001).
4. Kotsiantis, S. B., Zaharakis, I. D., & Pintelas, P. E. Supervised machine learning: A review of classification techniques. Emerging Artificial

Intelligence Applications in Computer Engineering (2007).
5. Han, J., Kamber, M., & Pei, J. , Data Mining: Concepts and Techniques (3rd ed.). Morgan Kaufmann (2011).
6. Provost, F., & Fawcett, T., Data Science for Business: What You Need to Know About Data Mining and Data-Analytic Thinking(2013).
7. Min, H. , Artificial intelligence in supply chain management: Theory and applications. International Journal of Logistics: Research and Applications (2010).
8. Manyika, J., Chui, M., Brown, B., et al. ,Big data: The next frontier for innovation, competition, and productivity. McKinsey Global Institute (2011).
9. Russom, P, Big Data Analytics. TDWI Best Practices Report. The Data Warehousing Institute (2011).
10. Chen, M., Mao, S., & Liu, Y. , Big data: A survey. Mobile Networks and Applications (2014).
11. Box, G.E.P., Jenkins, G.M., Reinsel, G.C., & Ljung, G.M. (2015). Time Series Analysis: Forecasting and Control, Wiley.
12. Taylor, S.J., & Letham, B. (2018). Forecasting at Scale, The American Statistician.
13. Zaharia, M., Chowdhury, M., Franklin, M.J., Shenker, S., & Stoica, I. (2016). Apache Spark: A Unified Engine for Big Data Processing, Communications of the ACM.
14. Armbrust, M., Das, T., Zhu, S., et al. (2020). Delta Lake: High-Performance ACID Table Storage Over Cloud Object Stores, VLDB Endowment.
15. Karau, H., & Warren, R. (2017). High Performance Spark: Best Practices for Scaling and Optimizing Apache Spark.
16. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system,

    Proceedings of the 22nd ACM SIGKDD Conference.

17. Guller, M. (2015). Big Data Analytics with Spark: A Practitioner's Guide to Using Spark for Large-Scale Data Analysis.
18. Hyndman, R.J., & Athanasopoulos, G. (2021). Forecasting: Principles and Practice.
19. Zhang, G., Eddy Patuwo, B., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting.
20. Mell, P., & Grance, T. (2011). The NIST definition of cloud computing. National Institute of Standards and Technology, U.S. Department of Commerce.
21. Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). Data Mining: Practical Machine Learning Tools and Techniques.
22. Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. International Journal of Information Management .
23. Delen, D., & Demirkan, H. (2013). Data, information and analytics as services.
    Decision Support Systems.