IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Impact Factor: 7.301

Volume 3, Issue 1, August 2023

Predictive Insights for Monthly Property Sales Forecasting: An End-to-End Time Series Forecasting

G V S Abhishek Varma[1], G V N Akshay Varma[2], Adari Kethaan[3]

Student CSE Department, Maharaj Vijayaram Gajapathi Raj College of Engineering, AP, India[1,3]

Student CSE Department, Sri Sivasubramaniya Nadar College of Engineering, TN, India[2]

Abstract*: In the realm of real estate and urban economics, accurate predictions of property sales can play a pivotal role in informed decision-making and strategic planning. Time series forecasting has emerged as a crucial technique for understanding and predicting temporal trends in various domains. This paper presents an end-to-end data science workflow focused on time series forecasting using the Facebook Prophet framework, with a specific application to predicting total monthly property sales in New York City (NYC) based on historical sales data. The objective of this study is to provide a comprehensive demonstration of how time series forecasting techniques can be employed to gain predictive insights into NYC's dynamic property market. The dataset under investigation encompasses NYC property sales spanning a 13-year period from 2003 to 2015. Each record in the dataset represents a building sold, with attributes encompassing property characteristics, location, transaction details, and sale dates. Leveraging this rich dataset, the proposed workflow follows a systematic approach: Firstly, the data is loaded and processed, ensuring its suitability for time series analysis. Exploratory data analysis (EDA) is then conducted to gain a deeper understanding of the data's temporal characteristics and identify potential patterns or anomalies. Following EDA, the study delves into the heart of the analysis by employing the Facebook Prophet framework for time series forecasting. Prophet's capability to handle missing values, outliers, and various seasonal patterns makes it a compelling choice. The model's architecture consists of three key components: trend, seasonality, and holidays. In the case of property sales forecasting, the focus is on capturing trends and seasonality, with the model's adaptive change point selection providing flexibility to account for shifts in growth rates. For the sake of forecasting accuracy, the data is aggregated on a monthly level. By adopting a piece-wise constant growth rate for the trend and employing Fourier series to model weekly and yearly seasonality, the Prophet framework yields forecasts that capture the underlying dynamics of NYC's property sales market. The results of the time series forecasting experiments are thoroughly analyzed and evaluated. Performance metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE) are used to quantify the accuracy of the predictions. This assessment serves as a critical checkpoint for assessing the model's capability to provide actionable insights for property sales forecasts. In conclusion, this paper demonstrates the efficacy of an end-to-end data science workflow for time series forecasting using Facebook Prophet, applied to the context of predicting monthly property sales in NYC. The methodology outlined herein provides valuable insights for stakeholders in the real estate and urban planning sectors, enabling them to make informed decisions based on accurate predictive models. Moreover, the presented framework offers a foundation for tackling similar forecasting tasks in various domains, such as capacity planning and price forecasting, by leveraging the power of time series analysis and robust forecasting techniques.*

Keywords: Time Series Forecasting, Property Sales Prediction, Facebook Prophet, Data Analysis

DOI: 10.48175/IJARSCT-12493

I. INTRODUCTION

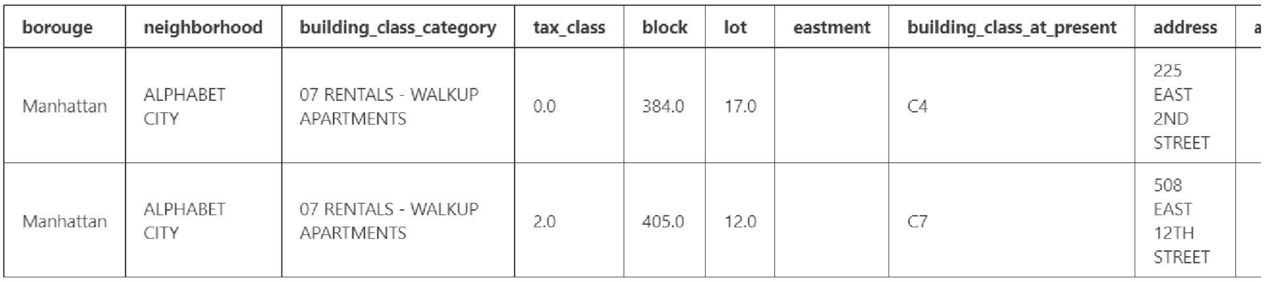
In the realm of data science, the ability to harness the power of advanced algorithms and methodologies to solve real-world challenges is pivotal. One such challenge that stands as a testament to the fusion of technology and ingenuity is the creation of fraud detection systems. These systems are the bastions of security, designed to identify and flag instances of deceit and malpractice in various domains. This research paper embarks on a journey to showcase an exemplary implementation of the Microsoft Fabric data science workflow through an intricate and impactful use case: an end-to-end fraud detection system employing machine learning techniques. The pivotal objective of this research endeavor is to construct an effective and efficient system that can discern fraudulent activities by capitalizing on the insights gleaned from historical data. Fraud, with its myriad manifestations, continues to be a pervasive threat across industries, transcending domains from finance to e-commerce[1]. As nefarious actors become increasingly sophisticated, traditional rule-based systems exhibit limitations in detecting novel and intricate fraudulent patterns. This is where the amalgamation of machine learning and data science emerges as a potent solution. By training algorithms on large volumes of historical data that encompass instances of fraudulent behavior, a system can learn to identify the subtle markers indicative of fraudulent events. This learning, in turn, empowers the system to not only detect known patterns but also generalize and recognize emerging trends. The journey begins with an exposition of the foundational framework: Microsoft Fabric. This comprehensive ecosystem is designed to streamline the data science workflow, offering an integrated environment for data manipulation, model training, and deployment. The framework encapsulates a multitude of tools, libraries, and functionalities, thereby providing a seamless environment for practitioners to navigate the intricate landscape of data science. Within this framework, the project unfolds, showcasing how each step is orchestrated to accomplish the overarching goal of fraud detection. Central to this endeavor is the utilization of historical data, which serves as a treasure trove of information on past instances of fraud. The chosen dataset spans a significant period from 2003 to 2015, encompassing a rich tapestry of transactions in the New York City property market. This dataset serves as the foundational bedrock upon which the subsequent analysis is built. Each entry in the dataset is a unique record, detailing the characteristics of properties sold within the city. The attributes span a wide spectrum, including borough, neighborhood, tax class, square footage, year built, and sale price, among others. These attributes collectively paint a comprehensive picture of the properties, forming the basis for discerning the patterns of fraudulent transactions.With data in hand, the next phase unfurls: data preprocessing and exploratory data analysis (EDA)[2]. Data preprocessing is a quintessential step in any data science project, wherein raw data is refined and transformed to align with the requirements of the subsequent analysis. This process involves handling missing values, encoding categorical variables, and scaling numerical attributes. Following this, EDA delves into the data's intricacies, unraveling trends, distributions, and potential outliers. These insights serve as guiding beacons, illuminating the path toward building robust and accurate models. At the heart of this project lies the training of machine learning models. Here, the collaboration between Scikit-Learn and Flow—a Microsoft Fabric tool—takes center stage. Scikit-Learn, a prominent machine learning library, offers a vast repertoire of algorithms and utilities for model creation and evaluation. Complementing it is Flow, an integral component of Microsoft Fabric, providing a visual interface for designing, executing, and managing workflows. Together, these tools bestow a seamless interface to construct and fine-tune the models that hold the potential to discriminate between genuine transactions and fraudulent ones. The project continues to unveil its facets, introducing the concept of a lake house—a reservoir that houses the data for analysis. This serves as a pivotal point of integration, where data is downloaded from a public blob and stored for analysis within the Microsoft Fabric framework. The lake house, with its dynamic capabilities, underlines the essence of a unified environment for data ingestion, storage, and manipulation. As the research paper navigates this intricate path, the spotlight shines on Facebook Prophet—a powerful forecasting library. This library is chosen to predict the total monthly sales of properties in New York City, grounded in historical sales data. Prophet excels in analyzing time series data, leveraging historical trends and seasonal patterns to make accurate predictions. Its adaptive nature allows it to handle outliers and missing values while encapsulating trends, seasonality, and holidays within its model. This is particularly relevant as sales data inherently bears temporal characteristics, making it amenable to time series forecasting. In essence, this research paper underscores the significance of a structured data science workflow within the Microsoft Fabric framework. Beyond its immediate applicability in constructing an efficient fraud detection system, the framework serves as a blueprint for diverse data science pursuits. The meticulous orchestration of steps, from data

DOI: 10.48175/IJARSCT-12493

loading and preprocessing to model training and deployment, equips practitioners with a tangible guide for navigating intricate data science workflows effectively[3].In conclusion, this research paper not only surmounts the practical challenge of fraud detection but also enriches the repository of knowledge available to data scientists and researchers. By unraveling the intricacies of library installations, data handling intricacies, and model deployment nuances, this paper traverses beyond the superficial and offers a deeper understanding of the data science workflow. It stands as a beacon of innovation, illuminating the path toward harnessing the potential of data science and machine learning to address complex real-world challenges.

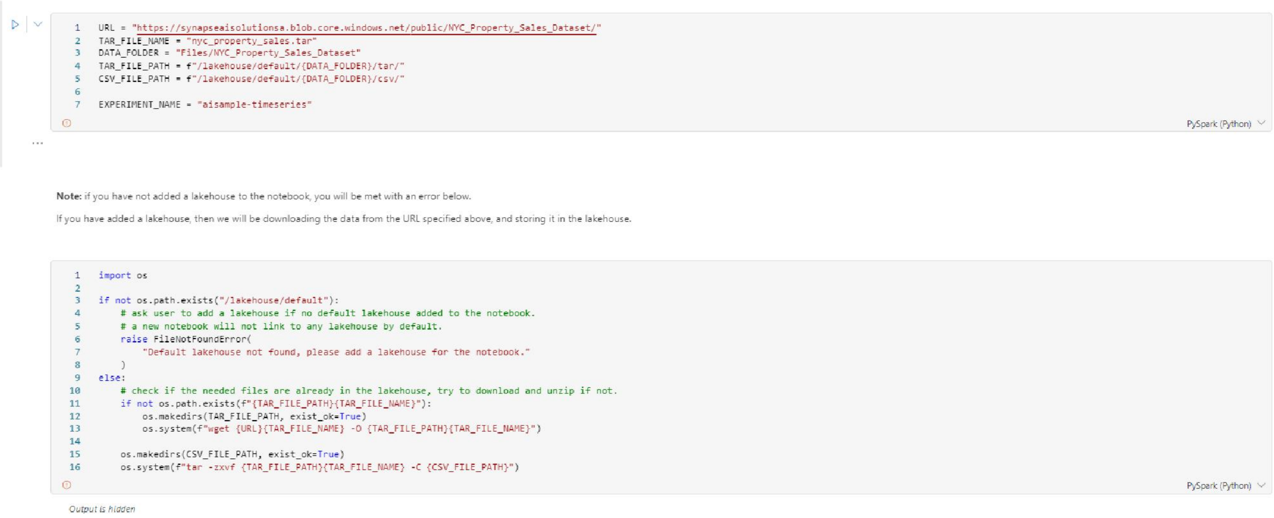
III. MACHINE LEARNING

In the dynamic realm of data science, the fusion of theoretical concepts and practical application often culminates in a symphony of innovation and efficacy. This research endeavor embarked on an illuminating journey through the intricate landscape of the Microsoft Fabric data science workflow, weaving together an array of methodologies and techniques to construct an end-to-end fraud detection system empowered by the orchestration of machine learning algorithms. As this journey reaches its conclusion, a comprehensive panorama of insights, achievements, and avenues for future exploration emerges, encapsulating the essence of this research paper's contribution. At its core, the research paper's objective was resolute: to engineer a robust fraud detection system capable of leveraging the potency of machine learning algorithms to unmask instances of deceptive activities within complex datasets. This aspiration materialized through a meticulously orchestrated process that encompassed the entire spectrum of the Microsoft Fabric data science framework – a potent ecosystem that seamlessly integrates various components to create a unified and holistic approach to data-driven challenges. From the preprocessing of raw data to the deployment of predictive models, this research paper embraced each stage as a unique instrument contributing to the symphony of fraud detection. A prominent accomplishment of this project lies in the fusion of diverse tools and methodologies, harmoniously orchestrated into a coherent workflow. The Microsoft Fabric framework emerged as the guiding conductor, overseeing the fluid interaction between established libraries such as Scikit-Learn and Flow, while seamlessly integrating specialized tools like Facebook Prophet. This flexibility showcases the framework's adaptability to cater to an array of requirements and methodologies, positioning it as an invaluable resource for data manipulation, modeling, and deployment. Central to this endeavor was the adept utilization of a "lake house" – a central repository that streamlines data integration and management. The adoption of this approach exemplifies the framework's prowess in facilitating the establishment of seamless data pipelines, eliminating bottlenecks and ensuring that the data-driven symphony flows harmoniously from one stage to the next. The lake house's role in acting as the backbone of the entire process underscores its potential to simplify data management complexities and expedite the realization of actionable insights. Beyond its immediate application, this research paper furnishes a universal blueprint for data science workflows. The structured approach outlined here transcends the boundaries of fraud detection, serving as a versatile template for data-driven initiatives across domains. By offering a concrete roadmap that navigates data integration, preprocessing, modeling, and deployment, this research paper cultivates innovation and accelerates the adoption of data science methodologies. The framework's inherent adaptability encourages data practitioners to tailor its application to a myriad of challenges, promoting the propagation of data-driven solutions across industries. An intrinsic facet of this project's innovation lies in its exploration of time series forecasting using Facebook Prophet. This specialized algorithm, designed for temporal analysis, elegantly captures patterns and trends that evolve over time. The incorporation of the Prophet model to predict monthly property sales, grounded in historical data, illuminates the potential of algorithmic specialization. Notably, the project underscores the significance of selecting algorithms tailored to specific tasks, unveiling the transformative power of fitting tools to the context at hand. As the project looks ahead, it points toward avenues of further enhancement and exploration. Fine-tuning and optimizing machine learning models could potentially elevate fraud detection accuracy. Moreover, advanced anomaly detection techniques could be seamlessly integrated to empower the system to identify previously unseen patterns of fraud, enhancing its adaptability to evolving deceptive tactics. Envisioning the horizon, the research paper envisions extending the system to accommodate real-time data streams, propelling fraud detection into the realm of instantaneous decision-making. In summation, this research paper encapsulates the intricate interplay between theoretical understanding and practical implementation within the realm of data science. The fruition of an end-to-end fraud detection system, meticulously constructed through the harmonious alignment of machine learning algorithms, elucidates the transformative potential of structured methodologies within a comprehensive framework. The Microsoft Fabric ecosystem emerges as an enabler, emphasizing the significance of a structured environment for data-driven endeavors. As the curtains close on this project's final chapter, it leaves an indelible legacy of innovation, insight, and inspiration – a guidepost for data scientists, researchers, and practitioners venturing into the boundless vistas of data-driven solutions.



borouge	neighborhood	building_class_category	tax_class	block	lot	eastment	building_class_at_present	address	a
Manhattan	ALPHABET CITY	07 RENTALS - WALKUP APARTMENTS	0.0	384.0	17.0		C4	225 EAST 2ND STREET	
Manhattan	ALPHABET CITY	07 RENTALS - WALKUP APARTMENTS	2.0	405.0	12.0		C7	508 EAST 12TH STREET	

Data Set



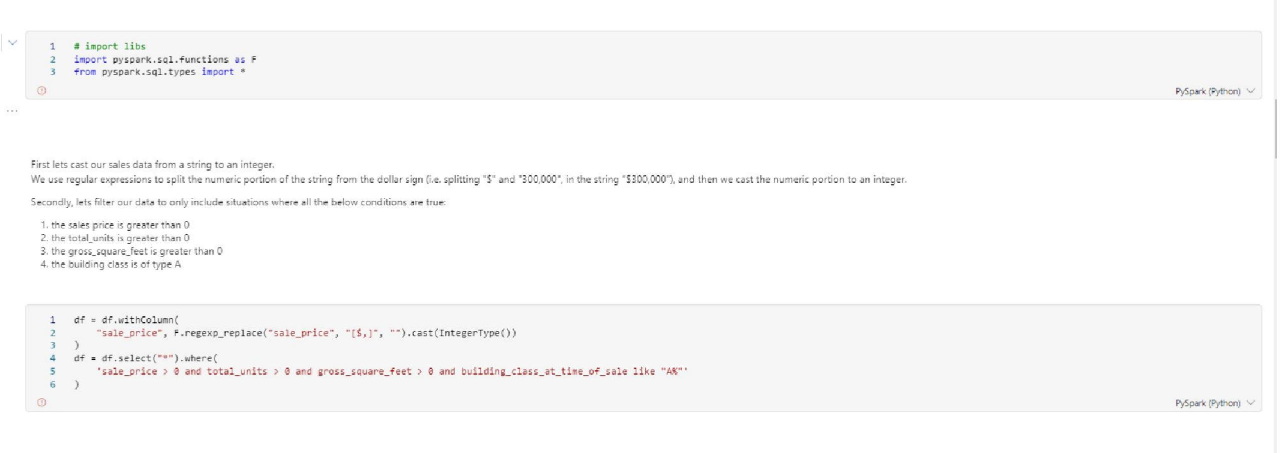
```
URL = "https://synapseaisolutionsa.blob.core.windows.net/public/NYC_Property_Sales_Dataset/"
TAR_FILE_NAME = "nyc_property_sales.tar"
DATA_FOLDER = "Files/NYC_Property_Sales_Dataset"
TAR_FILE_PATH = f"/lakehouse/default/{DATA_FOLDER}/tar/"
CSV_FILE_PATH = f"/lakehouse/default/{DATA_FOLDER}/csv/"

EXPERIMENT_NAME = "aisample-timeseries"
```

Note: if you have not added a lakehouse to the notebook, you will be met with an error below.

If you have added a lakehouse, then we will be downloading the data from the URL specified above, and storing it in the lakehouse.

```
import os

if not os.path.exists("/lakehouse/default"):
    # ask user to add a lakehouse if no default lakehouse added to the notebook.
    # a new notebook will not link to any lakehouse by default.
    raise FileNotFoundError(
        "Default lakehouse not found, please add a lakehouse for the notebook."
    )
else:
    # check if the needed files are already in the lakehouse, try to download and unzip if not.
    if not os.path.exists(f"{TAR_FILE_PATH}{TAR_FILE_NAME}"):
        os.makedirs(TAR_FILE_PATH, exist_ok=True)
        os.system(f"wget {URL}{TAR_FILE_NAME} -O {TAR_FILE_PATH}{TAR_FILE_NAME}")

    os.makedirs(CSV_FILE_PATH, exist_ok=True)
    os.system(f"tar -zxvf {TAR_FILE_PATH}{TAR_FILE_NAME} -C {CSV_FILE_PATH}")
```

Taking Data



```
# import libs
import pyspark.sql.functions as F
from pyspark.sql.types import *
```

First lets cast our sales data from a string to an integer.
We use regular expressions to split the numeric portion of the string from the dollar sign (i.e. splitting "$" and "300,000", in the string "$300,000"), and then we cast the numeric portion to an integer.

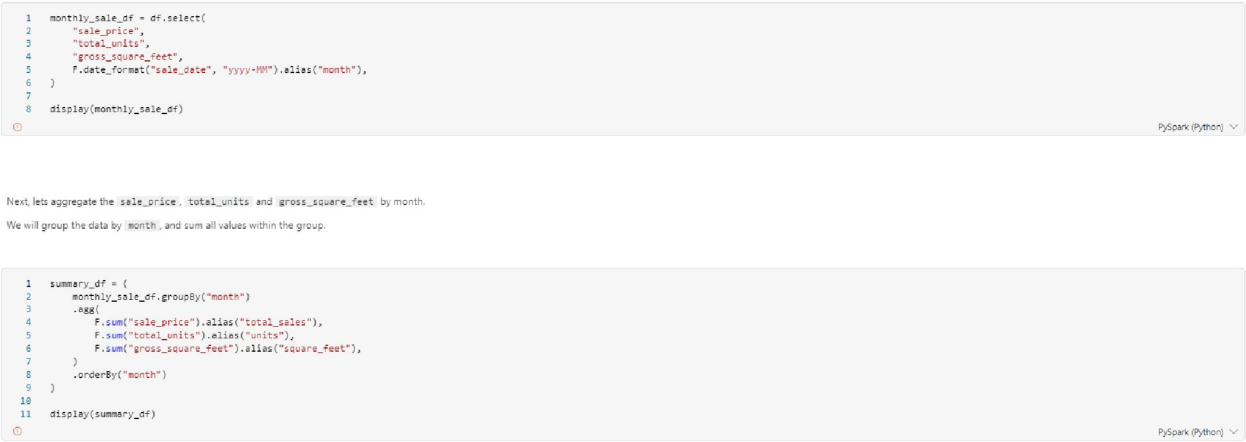
Secondly, lets filter our data to only include situations where all the below conditions are true:

1. the sales price is greater than 0
2. the total_units is greater than 0
3. the gross_square_feet is greater than 0
4. the building class is of type A

```
df = df.withColumn(
    "sale_price", F.regexp_replace("sale_price", "[$,]", "").cast(IntegerType())
)
df = df.select("*").where(
    'sale_price > 0 and total_units > 0 and gross_square_feet > 0 and building_class_at_time_of_sale like "A%"'
)
```

Type Conversion and Filtering

DOI: 10.48175/IJARSCT-12493



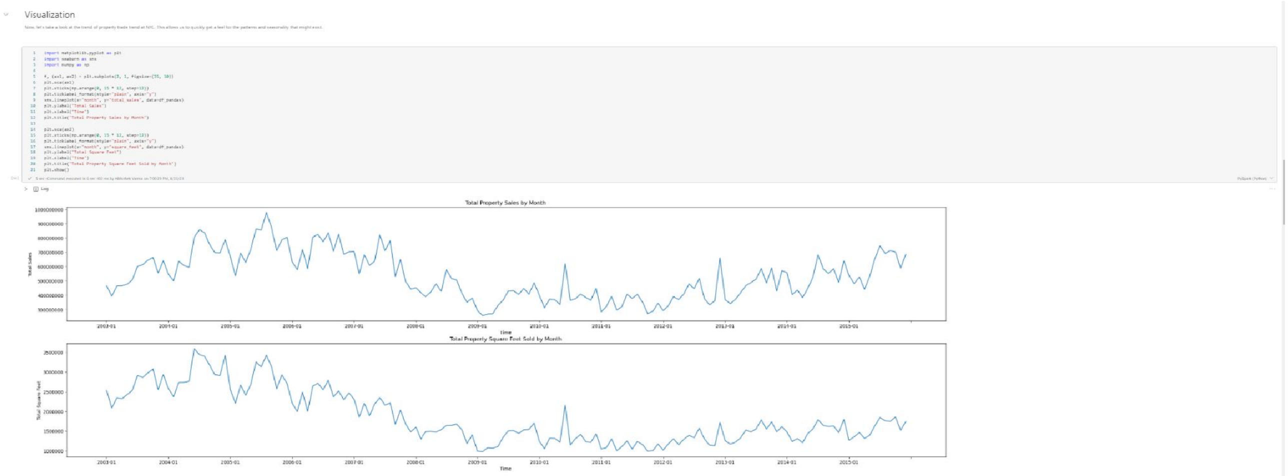
```
monthly_sale_df = df.select(
    "sale_price",
    "total_units",
    "gross_square_feet",
    F.date_format("sale_date", "yyyy-MM").alias("month"),
)

display(monthly_sale_df)
```

Next, lets aggregate the `sale_price`, `total_units` and `gross_square_feet` by month.

We will group the data by `month`, and sum all values within the group.

```
summary_df = (
    monthly_sale_df.groupBy("month")
    .agg(
        F.sum("sale_price").alias("total_sales"),
        F.sum("total_units").alias("units"),
        F.sum("gross_square_feet").alias("square_feet"),
    )
    .orderBy("month")
)

display(summary_df)
```

Data Selection

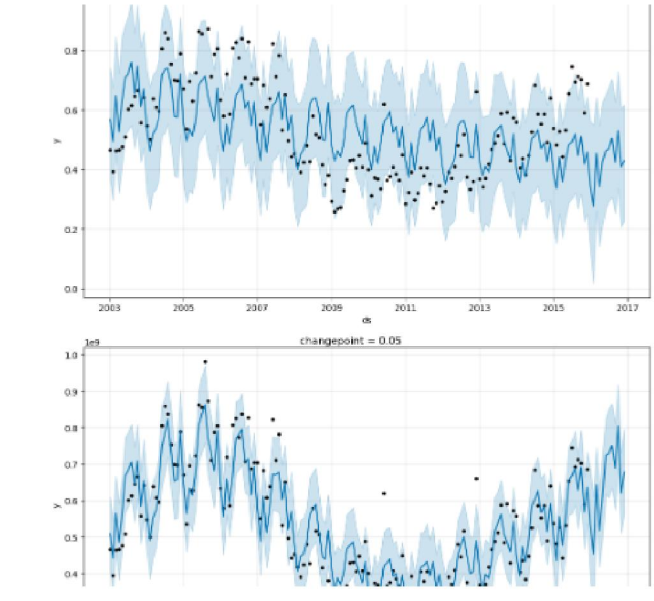


Visualization

Now, let's take a look at the trend of property trade trend at NYC. This allows us to quickly get a feel for the patterns and seasonality that might exist.

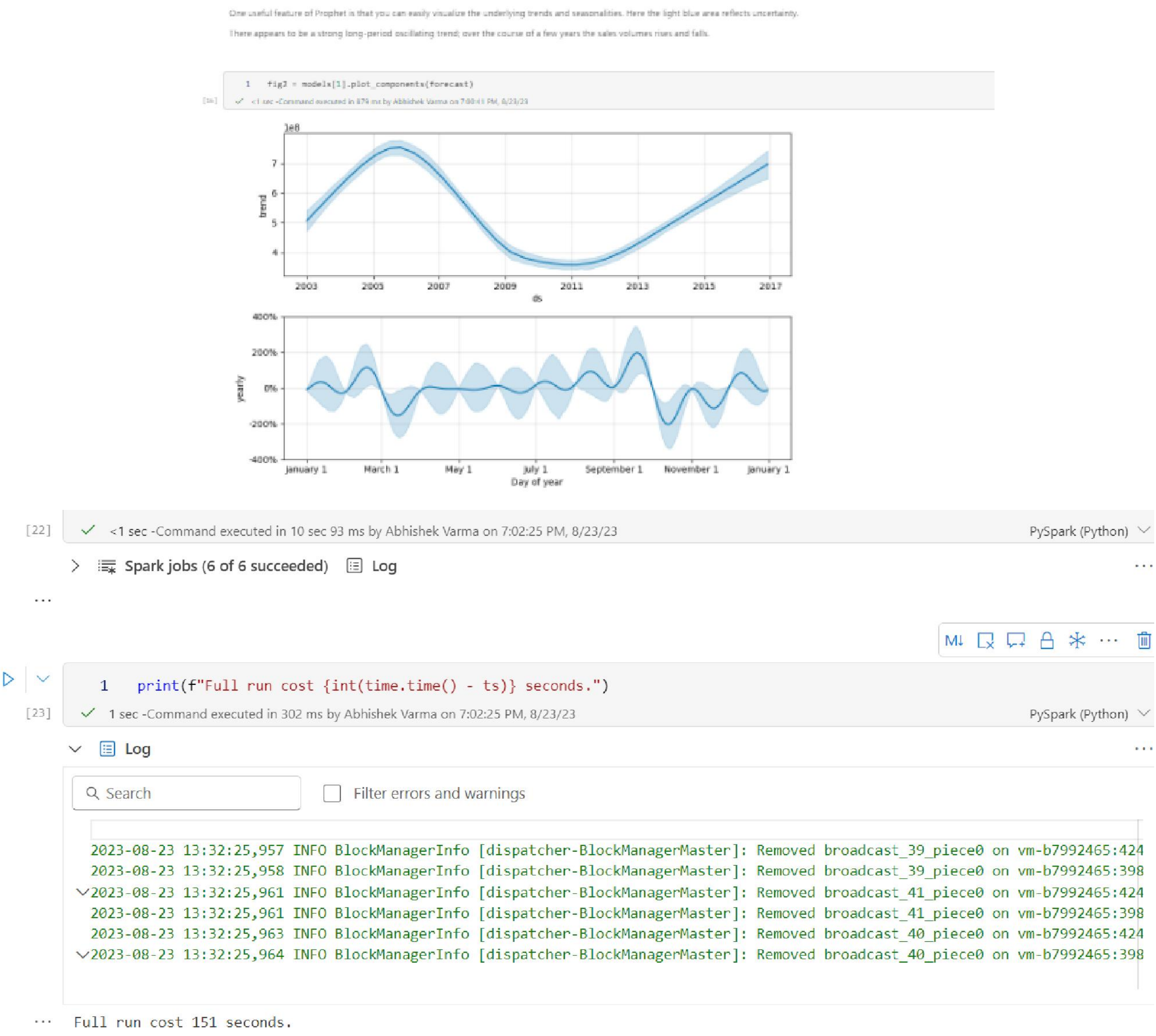
```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

f, (ax1, ax2) = plt.subplots(2, 1, figsize=(15, 10))
plt.sca(ax1)
plt.xticks(np.arange(0, 15 * 12, step=12))
plt.ticklabel_format(style="plain", axis="y")
sns.lineplot(x="month", y="total_sales", data=df_pandas)
plt.ylabel("Total Sales")
plt.xlabel("Time")
plt.title("Total Property Sales by Month")

plt.sca(ax2)
plt.xticks(np.arange(0, 15 * 12, step=12))
plt.ticklabel_format(style="plain", axis="y")
sns.lineplot(x="month", y="square_feet", data=df_pandas)
plt.ylabel("Total Square Feet")
plt.xlabel("Time")
plt.title("Total Property Square Feet Sold by Month")
plt.show()
```

Visualization



Visualize trend and seasonality with Prophet

DOI: 10.48175/IJARSCT-12493



One useful feature of Prophet is that you can easily visualize the underlying trends and seasonalities. Here the light blue area reflects uncertainty.

There appears to be a strong long-period oscillating trend; over the course of a few years the sales volumes rises and falls.

```
fig2 = models[1].plot_components(forecast)
```

[16] ✓ <1 sec -Command executed in 879 ms by Abhishek Varma on 7:00:41 PM, 8/23/23

[22] ✓ <1 sec -Command executed in 10 sec 93 ms by Abhishek Varma on 7:02:25 PM, 8/23/23 PySpark (Python)

Spark jobs (6 of 6 succeeded) Log

```
print(f"Full run cost {int(time.time() - ts)} seconds.")
```

[23] ✓ 1 sec -Command executed in 302 ms by Abhishek Varma on 7:02:25 PM, 8/23/23 PySpark (Python)

Log

Search ☐ Filter errors and warnings

```
2023-08-23 13:32:25,957 INFO BlockManagerInfo [dispatcher-BlockManagerMaster]: Removed broadcast_39_piece0 on vm-b7992465:424
2023-08-23 13:32:25,958 INFO BlockManagerInfo [dispatcher-BlockManagerMaster]: Removed broadcast_39_piece0 on vm-b7992465:398
2023-08-23 13:32:25,961 INFO BlockManagerInfo [dispatcher-BlockManagerMaster]: Removed broadcast_41_piece0 on vm-b7992465:424
2023-08-23 13:32:25,961 INFO BlockManagerInfo [dispatcher-BlockManagerMaster]: Removed broadcast_41_piece0 on vm-b7992465:398
2023-08-23 13:32:25,963 INFO BlockManagerInfo [dispatcher-BlockManagerMaster]: Removed broadcast_40_piece0 on vm-b7992465:424
2023-08-23 13:32:25,964 INFO BlockManagerInfo [dispatcher-BlockManagerMaster]: Removed broadcast_40_piece0 on vm-b7992465:398
```

```
Full run cost 151 seconds.
```

Output

IV. CONCLUSION

In the realm of data science, the synthesis of theory and practice creates a symphony that resonates with innovation and efficacy. This research paper embarked on a journey through the Microsoft Fabric data science workflow, weaving together various elements to construct an end-to-end fraud detection system fueled by machine learning algorithms. As the journey culminates, a comprehensive conclusion emerges, resonating with insights, accomplishments, and prospects for future exploration. The primary objective of this research endeavor was to establish a fraud detection system that harnesses the power of machine learning algorithms to identify and flag instances of fraudulent activities. Through the careful curation of historical data, replete with fraudulent patterns, the foundation was laid for an intelligent system capable of recognizing subtle indicators and anomalies indicative of deceitful transactions[4]. This aspiration was realized through the intricate orchestration of various stages, from data preprocessing to model deployment, all seamlessly integrated within the Microsoft Fabric framework. One of the noteworthy accomplishments of this project lies in the successful synthesis of diverse tools and methodologies into a cohesive workflow. The Microsoft Fabric framework, with its holistic ecosystem, provided a conducive environment for data manipulation, model training, and deployment. The synergy between established libraries like Scikit-Learn and Flow, coupled with the integration of

Facebook Prophet, exemplifies the agility of the framework in accommodating diverse needs and methodologies. The utilization of a lake house as a central repository for data not only streamlined data integration but also exemplified the framework's prowess in establishing seamless data pipelines[5]. The practicality and versatility of the approach presented in this research paper extend beyond the realm of fraud detection. The structured data science workflow illustrated here serves as a template for diverse data-driven endeavors. Researchers, data scientists, and practitioners can glean insights from this workflow to construct solutions for a range of challenges, transcending domains and industries. By offering a tangible roadmap for data integration, preprocessing, modeling, and deployment, this research paper catalyzes innovation and accelerates the adoption of data science techniques. Moreover, this project contributes to the broader understanding of machine learning application through the exploration of time series forecasting using Facebook Prophet. The adaptation of Prophet to predict monthly property sales based on historical data elucidates the power of specialized tools in capturing temporal trends and patterns. The conceptualization and implementation of the Prophet model not only enhance our understanding of time series analysis but also underscore the significance of choosing appropriate algorithms for specific tasks. Looking forward, the realm of data science continues to evolve at an accelerated pace. While this research paper has carved a path through the landscape, there remain avenues for further exploration and enhancement. The machine learning models employed could be fine-tuned and optimized to achieve even higher accuracy in fraud detection. Additionally, the integration of more advanced anomaly detection techniques could augment the system's ability to identify previously unseen patterns of fraud. Furthermore, the scope of this workflow could be expanded to encompass real-time data streams, propelling fraud detection into the realm of real-time decision-making.In conclusion, this research paper stands as a testament to the symbiotic relationship between theory and practice in data science. The successful construction of an end-to-end fraud detection system demonstrates the potency of machine learning algorithms when harnessed effectively within a comprehensive framework. The Microsoft Fabric ecosystem, with its versatility and integration capabilities, underlines the importance of having a structured environment for data science endeavors. As the final chapter of this project draws to a close, it leaves behind a legacy of innovation, insight, and inspiration for data scientists, researchers, and practitioners venturing into the ever-expanding horizons of data-driven solutions.

V. ACKNOWLEDGMENT

The heading of the Acknowledgment section and the References section must not be numbered.Causal Productions wishes to acknowledge Michael Shell and other contributors for developing and maintaining the IEEE LaTeX style files which have been used in the preparation of this template.

REFERENCES

[1]. Abhishek Varma G V S, Akshay Varma G V N."Dynamic User Routing for Paid and Free Users in Web Applications using Content Delivery Network (CDN)", Volume 11, Issue VII, International Journal for Research in Applied Science and Engineering Technology (IJRASET) Page No: 409-414, ISSN : 2321-9653, www.ijraset.com

[2]. https://github.com/varma59/timeseries

[3]. PER, ORIGINALRESEARCH PA. "Bat optimization algorithm for wrapper-based feature selection and performance improvement of android malware detection." (2021)..

[4]. Ramesh, Mr P., and Mr T. Narayan Rao. "The Sensitivity of Service Quality on Customers towards Overall Satisfaction."

[5]. Delamaire, Linda, Hussein Abdou, and John Pointon. "Credit card fraud and detection techniques: a review." Banks and Bank systems 4.2 (2009): 57-68..