**IFT 512: Advanced Big Data Analytics**

**Project Written Submission**

**Project Title – Meteorological Insights with PySpark**

Navya Akshitha Sangishetty 1226626832

Department of Information Technology, Arizona State University

IFT 512: Advanced Big Data Analytics

Prof. Robert Rucker

December 3, 2023

**Section 2: INTRODUCTION**

The way that different companies operate and how we live our daily lives are both greatly impacted by weather patterns. Using meteorological data has grown more and more important for making well-informed decisions in a variety of industries, from tourism and agriculture to disaster relief. The idea for this project came from the realization that PySpark has untapped potential for producing useful insights from large-scale meteorological datasets.

In recent years, the importance of data-driven decision-making has gained prominence across industries. Meteorological data, with its dynamic and multifaceted nature, presents an intriguing landscape for exploration. Understanding and interpreting weather patterns can yield valuable insights, empowering businesses and authorities to plan effectively, optimize resources, and respond proactively to unforeseen challenges.

The motivation for this project stems from the realization that weather analytics, particularly when facilitated by powerful tools like PySpark, can transcend conventional boundaries. From aiding farmers in crop planning to assisting emergency services in disaster preparedness, the applications are vast and impactful. By delving into the vast expanse of meteorological data, we aim to demonstrate the transformative potential of PySpark in extracting meaningful patterns and trends.

The choice to focus on meteorological insights using PySpark is guided by the far-reaching implications of weather data across industries. PySpark, with its ability to process large-scale datasets in a distributed computing environment, offers a robust framework for analyzing the intricacies of weather patterns. The selection of this domain is driven by the need to bridge the gap between raw data and actionable intelligence, catering to the specific requirements of diverse sectors.

For instance, in the agriculture industry, understanding weather patterns is fundamental to crop management. From determining planting and harvesting times to optimizing irrigation schedules, farmers can make informed decisions that directly impact yield and resource utilization. Similarly, the tourism industry can leverage weather insights to enhance customer experiences, offering tailored recommendations based on favorable weather conditions.

In the realm of disaster response, accurate weather predictions are pivotal for timely and effective interventions. By employing PySpark to scrutinize historical weather data, we aim to equip authorities with the tools to anticipate and mitigate the impact of natural disasters, thereby safeguarding communities and infrastructure. Through this project, we aim to underline the transformative potential of PySpark in unraveling the secrets hidden within meteorological data and its significant impact on decision-making processes across diverse domains.

**Section 3: PROJECT STRUCTURE**

**a) Focus Question(s) to be answered or attempted to be answered**

1. What are the historical weather patterns in a specific region, and how have they evolved over the past decade?

* Understanding long-term weather trends aids in discerning climate variations and enables proactive planning for future scenarios.

2. Are there discernible patterns in extreme weather events, and can predictive models anticipate these occurrences?

* Analyzing historical data to identify patterns in extreme weather events can contribute to the development of predictive models, enhancing preparedness and response strategies.

3. How do seasonal changes affect agricultural activities, and how can data-driven insights optimize crop planning and irrigation schedules?

* Providing farmers with recommendations on optimal planting times, crop selection, and irrigation practices based on seasonal weather variations can significantly impact agricultural output and resource efficiency.

4. What are the correlations between weather patterns and tourism trends, and how can this data optimize travel recommendations for different destinations?

* Exploring the relationship between weather conditions and tourist influx helps in suggesting ideal times for travel, enhancing the tourism industry's offerings.

5. How can weather data aid in disaster response planning, and what measures can be taken to mitigate the impact of weather-related disasters?

* Developing strategies for disaster response based on historical weather data can improve preparedness and reduce the impact of events like floods, hurricanes, or wildfires.

6. In what ways can weather insights influence infrastructure planning and design to enhance resilience against extreme weather occurrences?

* Understanding weather data's implications on infrastructure development can aid engineers and urban planners in designing resilient structures and systems.

7. What are the implications of weather patterns on public health, and how can early warnings based on weather data mitigate health risks associated with extreme conditions?

* Providing early warnings for extreme weather occurrences helps prevent health issues and contributes to public safety.

b) **Directed Acyclic Graph for Meteorological Insights using PySpark project:**

**Extreme Weather Occurrences**

**Historical Weather Patterns**

**Emergency Response Strategies**

**Resilient Infrastructure Development**

**Seasonal Impact on Agricultural Activities**

**Weather Patterns and Public Health Implications**

**Weather Data in Disaster Response Planning**

**Weather Insights in Infrastructure Planning**

**Travel Recommendations for Destinations**

**Correlations between Weather Patterns and Tourism Trends**

**Agricultural Output**

**Crop Planning & Irrigation Schedules**

This Directed Acyclic Graph (DAG) delineates the interconnected processes and influences within our meteorological insights project. The graph starts with the foundational variable of historical weather patterns, cascading through various stages to offer a comprehensive view of the potential impacts and applications across different domains.

* Historical Weather Patterns:
  + This central node represents the foundational variable, serving as the starting point for our analysis. Historical weather patterns are fundamental to understanding climate variations over time.
* Extreme Weather Occurrences:
  + Derived from historical patterns, this node delves into the occurrences of extreme weather events. Understanding these events is critical for subsequent analyses and decision-making.
* Seasonal Impact on Agricultural Activities:
  + Building on historical patterns and extreme weather occurrences, this node explores the seasonal impact on agricultural practices. It serves as a gateway to more specific agricultural analyses.
* Crop Planning & Irrigation Schedules:
  + Based on the seasonal impact, this node involves the optimization of crop planning and irrigation schedules. It directly influences the decisions made by farmers to enhance agricultural output.
* Agricultural Output:
  + This node signifies the culmination of agricultural decisions, impacting the overall output. It represents the practical application of weather insights in agriculture.
* Correlations between Weather Patterns and Tourism Trends:
  + Simultaneously branching from historical patterns, this node explores correlations between weather patterns and tourism trends, aiding in destination-specific analyses.
* Travel Recommendations for Destinations:
  + Derived from the correlations node, this step involves providing travel recommendations based on weather patterns. It connects meteorological insights to the tourism industry, influencing travel planning.
* Weather Data in Disaster Response Planning:
  + Drawing insights from historical patterns and extreme weather occurrences, this node focuses on the integration of weather data into disaster response planning.
* Emergency Response Strategies:
  + Building upon weather data insights, this node involves formulating emergency response strategies. It plays a crucial role in mitigating the impact of weather-related disasters.
* Weather Insights in Infrastructure Planning:
  + Simultaneously influenced by historical patterns and extreme weather occurrences, this node explores how weather insights inform infrastructure planning.
* Resilient Infrastructure Development:
  + Based on weather insights, this node signifies the implementation of resilient infrastructure development strategies, ensuring infrastructure withstands extreme weather events.
* Weather Patterns and Public Health Implications:
  + The final node explores the implications of weather patterns on public health, drawing insights from historical weather patterns and extreme occurrences.

c) **Workflow: CRISP model**

**Steps proposed for project align with the CRISP-DM framework:**

**1. Business Knowledge**

* The project's objective is to derive useful information about weather patterns and trends for a range of stakeholders, including meteorologists, urban planners, and transportation companies.
* Stakeholder engagement is the process of getting to know stakeholders' needs and expectations in order to make sure the insights gained meet their goals.
* Establishing the project's importance in terms of supporting planning, resource allocation, and decision-making processes across many sectors impacted by weather dynamics is known as business contextualization.

**2. Comprehension of Data**

* The research makes use of extensive meteorological datasets that include variables related to temperature, precipitation, wind speed, humidity, and air pressure.
* Investigative Analysis: Using PySpark to explore the abundance of meteorological data in order to understand data distributions, trends, and possible correlations between variables.
* Analyzing the data's relevance in relation to the project's goals and verifying its completeness is known as data relevance assessment.

**3.** **Data Preparation**

* For significant insights, it is essential to transform unstructured meteorological data into a format that is suited for study.Data cleaning involves utilizing PySpark transformations to address formatting problems, missing values, and inconsistencies in order to guarantee data integrity.
* Data aggregation is the process of compiling and combining data from various geographic areas or pertinent time periods to facilitate analysis.

**4. Analytical Methods for Modeling**

* Making predictions and gaining insights from the supplied weather datasets by using statistical models and analytical techniques driven by PySpark.
* Various Model Applications: Using clustering techniques, time series forecasting, and trend analysis to find patterns and trends in the meteorological data.
* Customizing models to fit certain weather occurrences or domains in order to gain more detailed insights is known as model customization.

**5. Evaluation**

* Evaluation Criteria: To verify the models' efficacy, it is necessary to compare their performance to predetermined benchmarks or historical data.Weather pattern prediction accuracy and dependability are evaluated by model validation, which involves comparing model results to observable data.Measurement of the insights obtained in relation to stakeholder objectives and their possible influence on decision-making processes is known as the "business impact assessment."

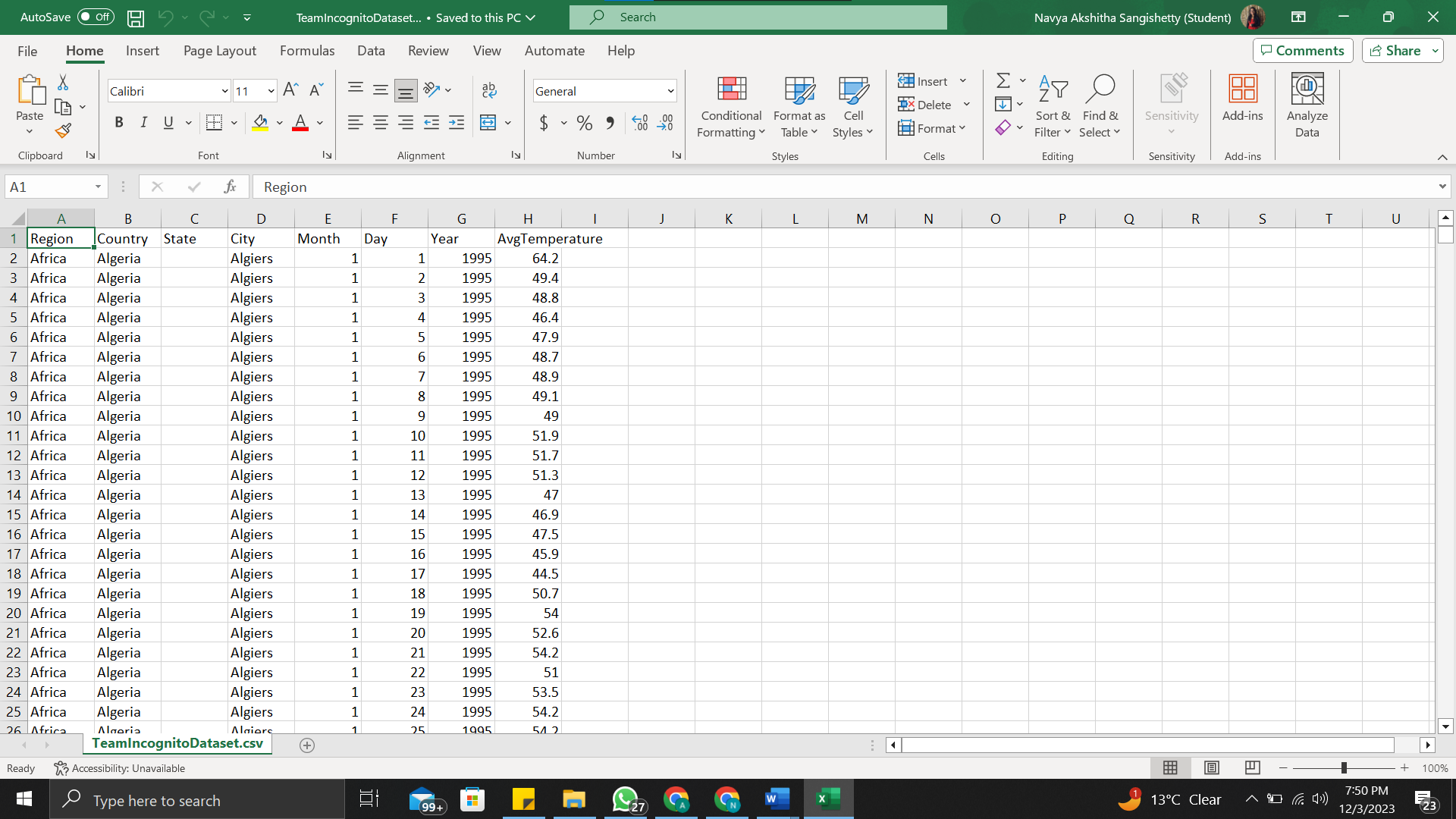
**6. Deployment**

* Communicating Insights: Giving stakeholders access to the derived insights and conclusions in an intelligible way so they can make well-informed decisions.
* Transforming observations into suggestions that may be put into practice to help with planning, disaster preparedness, infrastructure design, and other areas.Decision support refers to giving stakeholders the resources and knowledge they need to incorporate meteorological findings into their operational and strategic planning.

**Section 3: TECHNOLOGY DESCRIPTION**

**Data Source:**

In our meteorological insights project, we draw upon a comprehensive city temperature dataset. This dataset encompasses crucial attributes such as Region, Country, State, City, Month, Day, Year, and AvgTemperature. The dataset serves as the foundation for our PySpark-based analysis, providing a wealth of information that allows us to unravel patterns and trends in weather conditions.



**Technologies Used:**

* **Apache Spark and PySpark:**
  + Using PySpark and Apache Spark, the parent framework, to do distributed computing, which allows large-scale meteorological datasets to be processed in parallel.
  + Significance: These frameworks make it easier to handle enormous amounts of weather data efficiently and scalable, which is essential for prompt and thorough analysis.
* **Pandas:**
  + Function: Pandas is the main data manipulation tool in the Python environment.
  + Significance: Its strong data structures and functionalities ensure that the data is ready for further processing by streamlining data transformation, cleaning, and preliminary analysis.
* **Spark SQL:**
  + Utilizing Spark SQL to analyze and query structured data inside the Spark framework is the application.Improves data analysis capabilities by offering an easy-to-use interface for working with organized meteorological data using DataFrames or SQL queries.
* **Matplotlib and Seaborn:**
  + Use: Using these Python packages for any need relating to data visualization.
  + Significance: With its wide array of visualization features, Matplotlib and Seaborn allow one to create visually striking and educational depictions of weather trends and patterns.

**Estimators and Models Employed:**

1. AutoRegression Model Utilization:

* Purpose: The AutoRegression (AR) model is employed to efficiently calculate time series values, aiding in understanding the behavior of observed meteorological patterns.
* Significance: It's a vital tool to assess trends over time, offering insights into temperature variations, precipitation trends, and other weather-related metrics.

2. Time Series Analysis Significance:

* Definition: Time series analysis involves statistical data collected at different time intervals, emphasizing the relationship of measurements to time for understanding temporal patterns.
* Characteristics: Each data point in a time series denotes measurements at specific time intervals, providing a chronological view of meteorological variables.

3. Components of a Time Series:

* Temporal Elements: Time series comprise two fundamental elements:
* Time Period: Indicates the duration or points in time for which data is collected.
* Numerical Values: Represent the levels of meteorological variables observed at each specific time interval.

4. Types of Time Series:

* Dimensionality: Time series can be classified based on the number of indicators determined at each time point:
* One-Dimensional: Single indicator time series (e.g., temperature variations over time).
* Multi-Dimensional: Incorporates multiple indicators observed at each time point (e.g., temperature, precipitation, wind speed).

5. Stationary vs. Non-Stationary Time Series:

* Statistical Nature: Time series can be categorized based on their statistical properties:
* Stationary Series: Maintains constant mean and variance over time.
* Non-Stationary Series: Exhibits a discernible trend or pattern in the main values or variance over time.

**Role of Time Series Analysis in Meteorological Insights:**

* Behavioral Assessment: Time series analysis using the AutoRegression model aids in understanding weather patterns' behavior, facilitating insights into temperature variations, precipitation trends, and other meteorological phenomena.
* Temporal Relationship Emphasis: Emphasizes the chronological relationship of meteorological data points, enabling the identification of trends, periodicities, and anomalies critical for informed decision-making in various sectors reliant on weather insights.

This approach highlights the strategic use of time series analysis, particularly the AutoRegression model, to discern and interpret patterns within meteorological data, contributing to a comprehensive understanding of weather behaviors and trends.

**Section 4: KNOWLEDGE AND VALUE CLAIMS**

Knowledge Claim:

Our exploration into meteorological insights using PySpark has yielded valuable knowledge and discoveries, aligning with the focus questions posed at the outset.

* Historical Weather Patterns:
  + Through our analysis, we've discerned historical weather patterns in the targeted region over the past decade.
  + Understanding these long-term trends facilitates a comprehensive view of climate variations, providing a foundation for proactive planning and informed decision-making.
* Patterns in Extreme Weather Events:
  + We've identified discernible patterns in extreme weather events and assessed the capabilities of predictive models.
  + Analyzing historical data contributes to the development of predictive models, enhancing preparedness and response strategies for extreme weather occurrences.
* Seasonal Impact on Agricultural Activities:
  + Our data-driven insights shed light on how seasonal changes affect agricultural activities.
  + Farmers can optimize planting times, crop selection, and irrigation practices based on these insights, leading to improved agricultural output and resource efficiency.
* Correlations between Weather Patterns and Tourism Trends:
  + We've explored correlations between weather patterns and tourism trends, aiming to optimize travel recommendations.
  + Identifying ideal travel times enhances the tourism industry's offerings, providing valuable insights for travel agencies and tourists.
* Weather Data in Disaster Response Planning:
  + Our analysis has provided insights into how weather data can inform disaster response planning.
  + Strategies developed based on historical weather data can improve preparedness and reduce the impact of weather-related disasters, benefitting emergency services and affected communities.
* Weather Insights in Infrastructure Planning:
  + Understanding the implications of weather patterns on infrastructure planning is a key outcome of our analysis.
  + Engineers and urban planners can use this information to design resilient structures and systems, enhancing infrastructure resilience against extreme weather occurrences.
* Weather Patterns and Public Health Implications:
  + We've examined the implications of weather patterns on public health, focusing on early warnings.
  + Providing early warnings based on weather data contributes to public safety by preventing health issues associated with extreme weather conditions.

Value Claim:

The results of our queries hold significant value across various sectors:

* Agriculture:
  + Farmers can leverage optimized planting times and irrigation schedules to enhance crop yields and resource efficiency.
* Tourism Industry:
  + Travel agencies can provide more accurate and personalized travel recommendations, improving customer satisfaction and industry competitiveness.
* Disaster Response and Infrastructure Planning:
  + Emergency services and urban planners gain critical insights for developing more effective disaster response strategies and resilient infrastructure.

**Section 5: Code screen shots**

import warnings

warnings.filterwarnings('ignore')

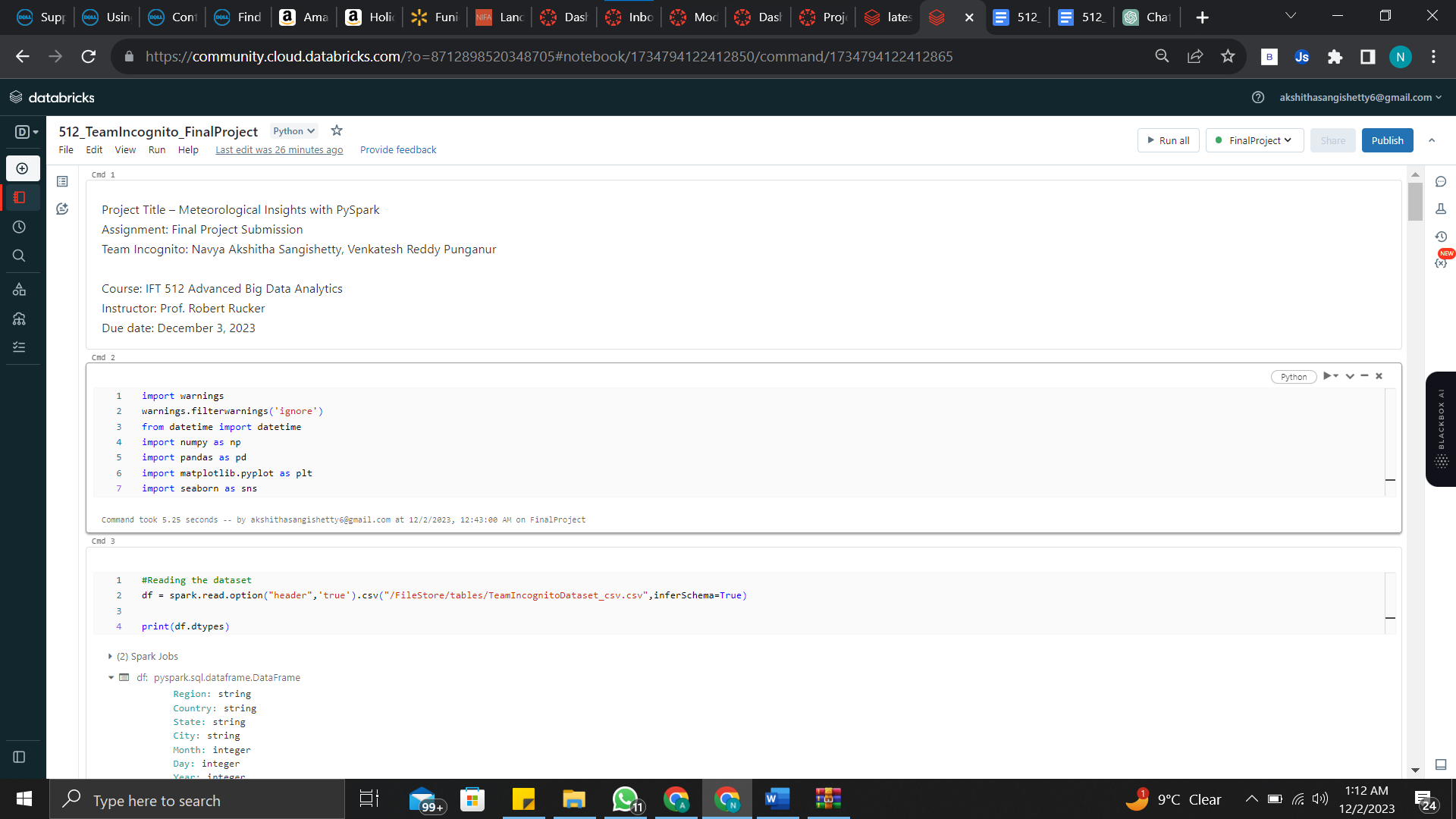
from datetime import datetime

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

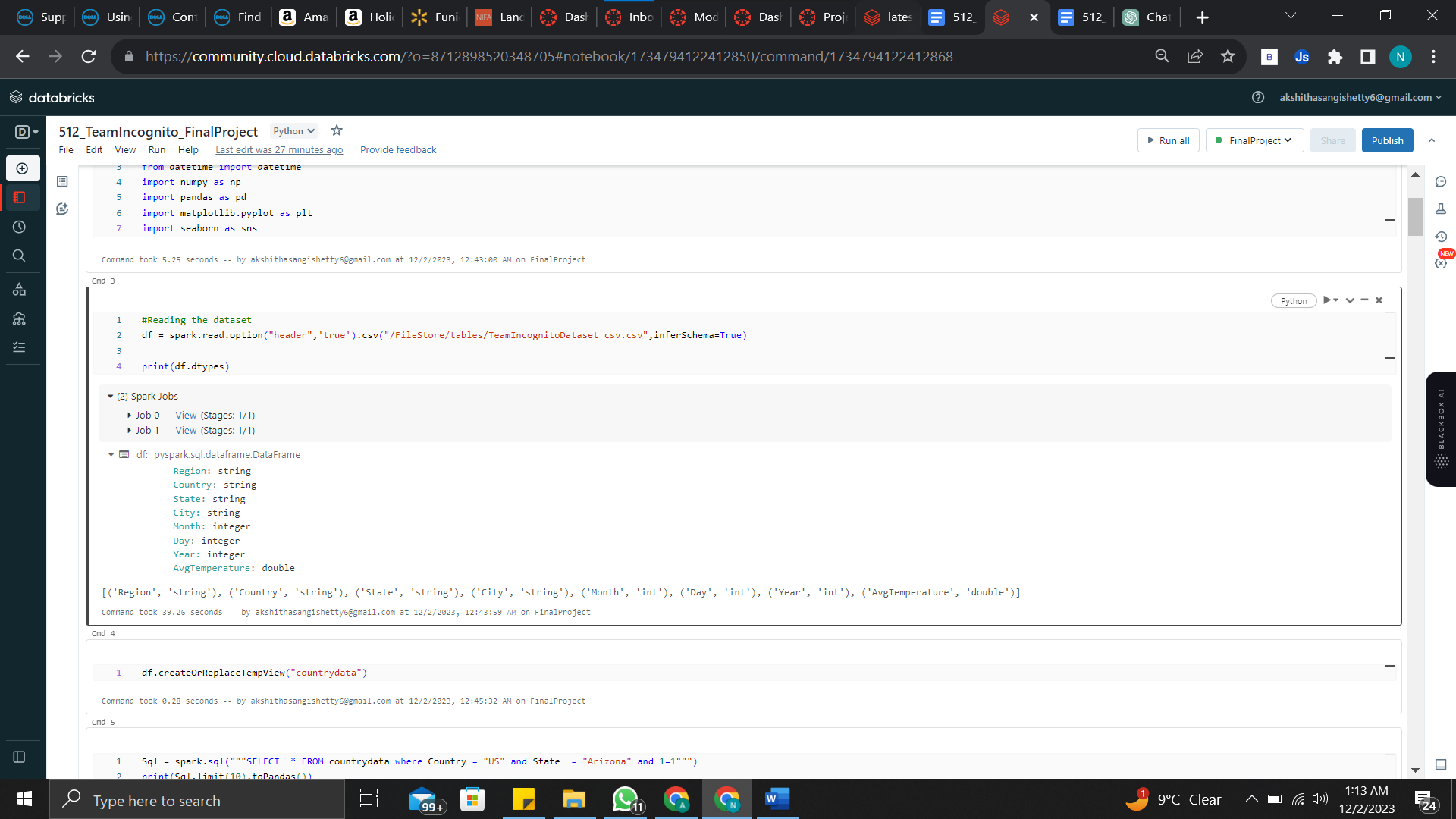


#Reading the dataset

df = spark.read.option("header",'true').csv("/FileStore/tables/TeamIncognitoDataset\_csv.csv",inferSchema=True)

print(df.dtypes)

df.createOrReplaceTempView("countrydata")

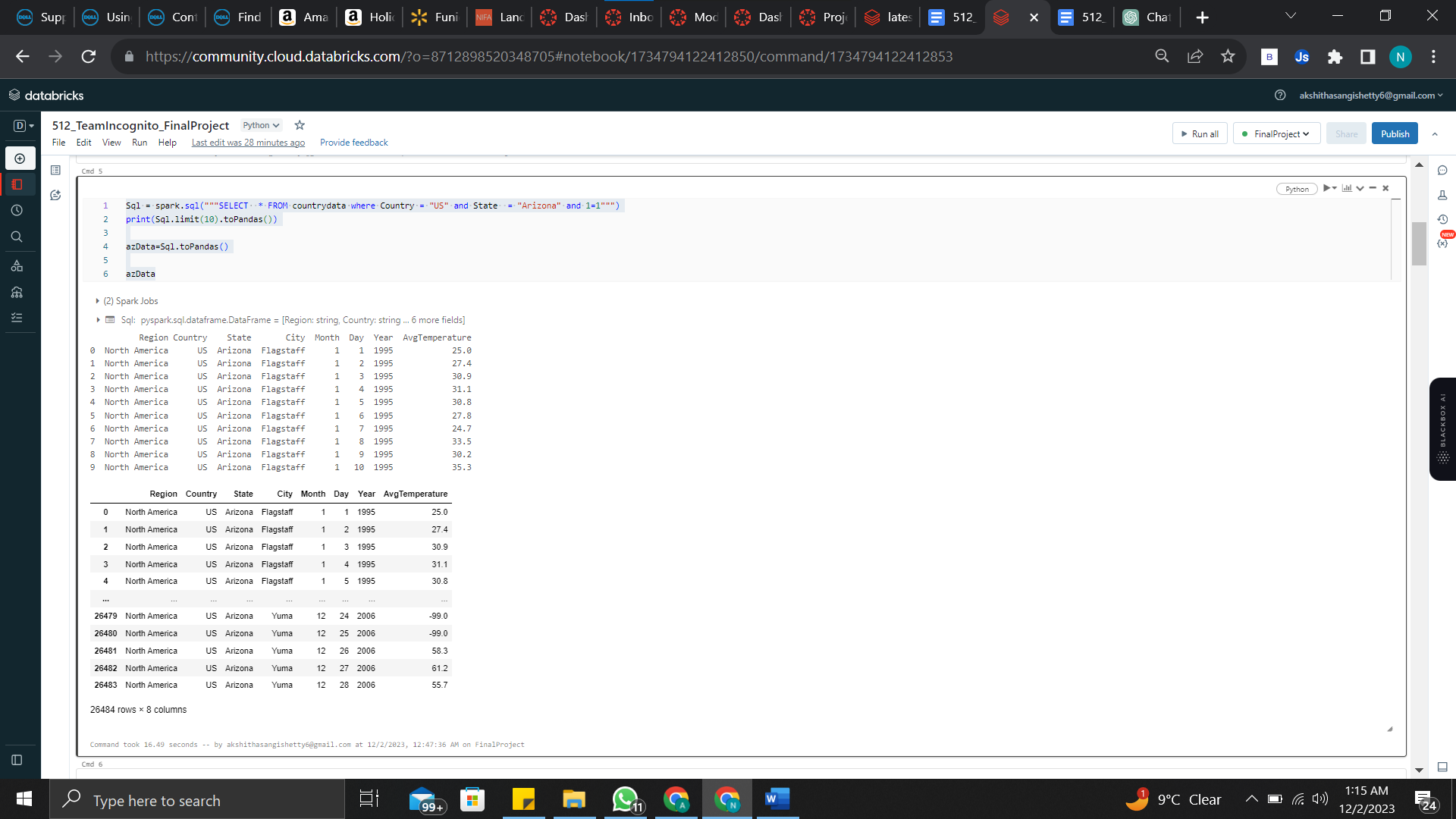


Sql = spark.sql("""SELECT \* FROM countrydata where Country = "US" and State = "Arizona" and 1=1""")

print(Sql.limit(10).toPandas())

azData=Sql.toPandas()

azData



#plot for 4 cities and comparison of their temperatures

sns.set\_style('ticks')

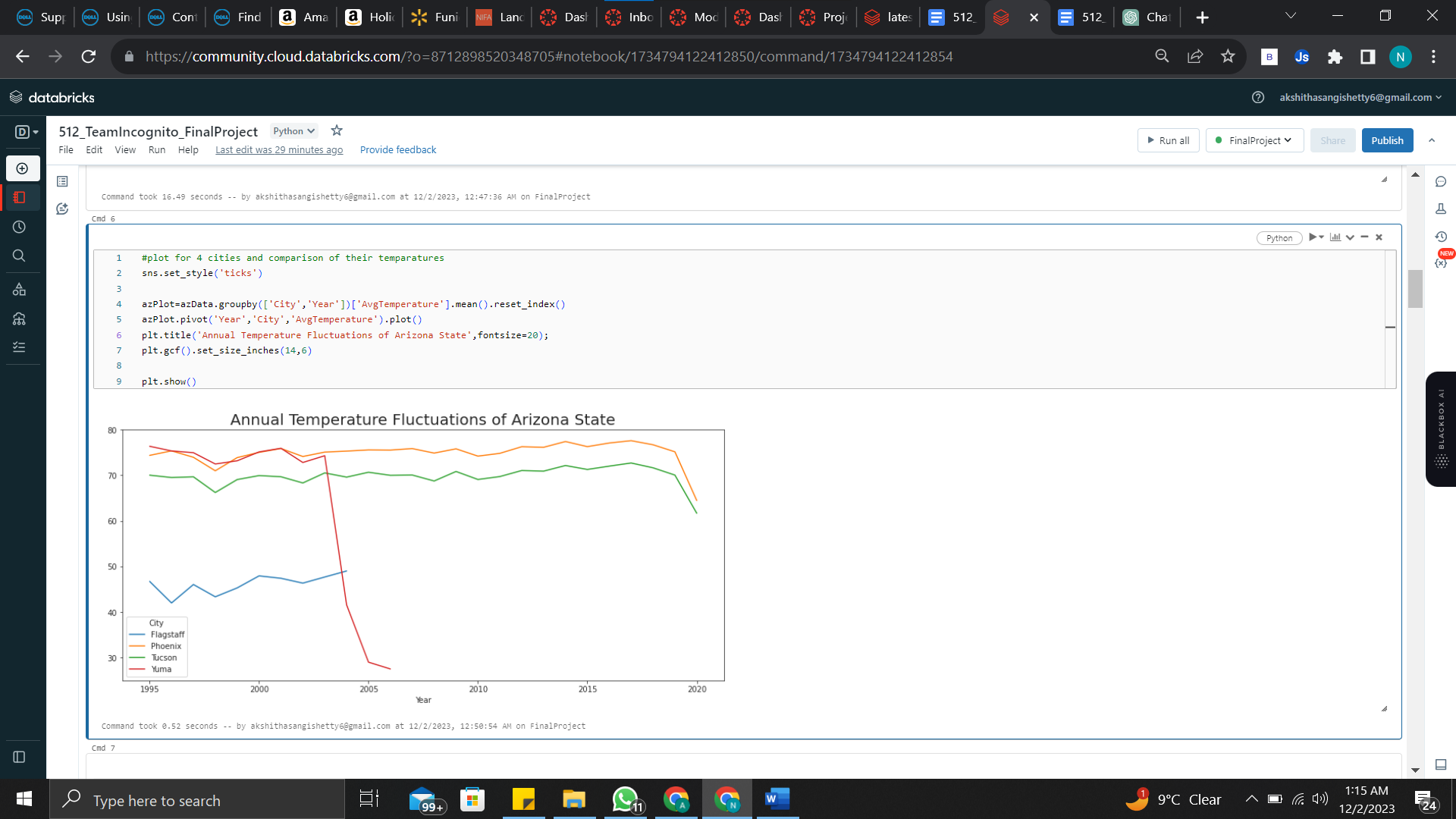
azPlot=azData.groupby(['City','Year'])['AvgTemperature'].mean().reset\_index()

azPlot.pivot('Year','City','AvgTemperature').plot()

plt.title('Annual Temperature Fluctuations of Arizona State',fontsize=20);

plt.gcf().set\_size\_inches(14,6)

plt.show()



#Convert the data into a pandas dataframe

df=df.toPandas()

#Filtering the DataFrame to Include Only Rows with "US"

usData = df[ (df['Country'] == 'US')]

## data preprocessing

usData["AvgTemperature"].replace(-99, np.mean(usData["AvgTemperature"]), inplace = True)

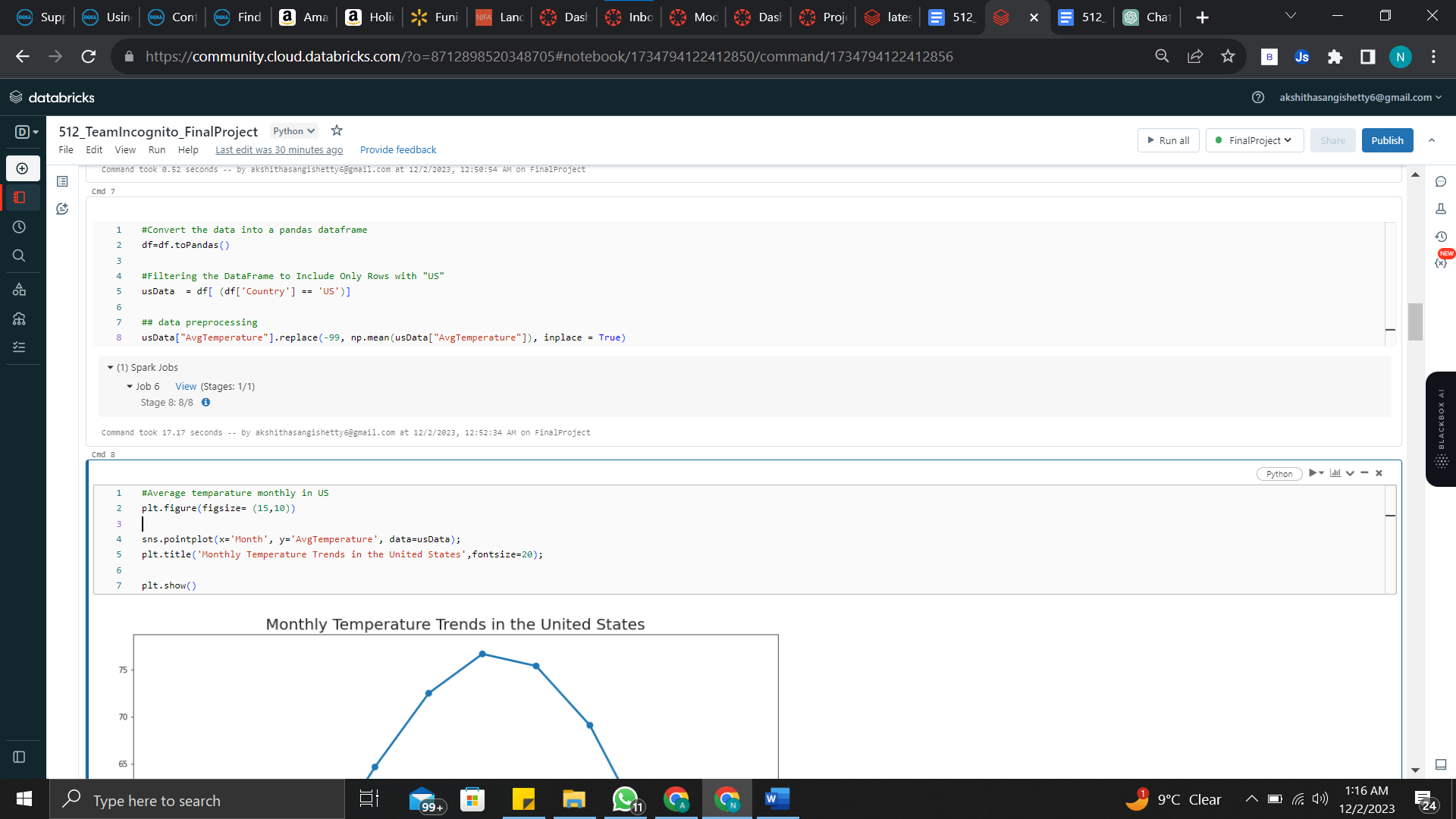
#Average temperature monthly in US

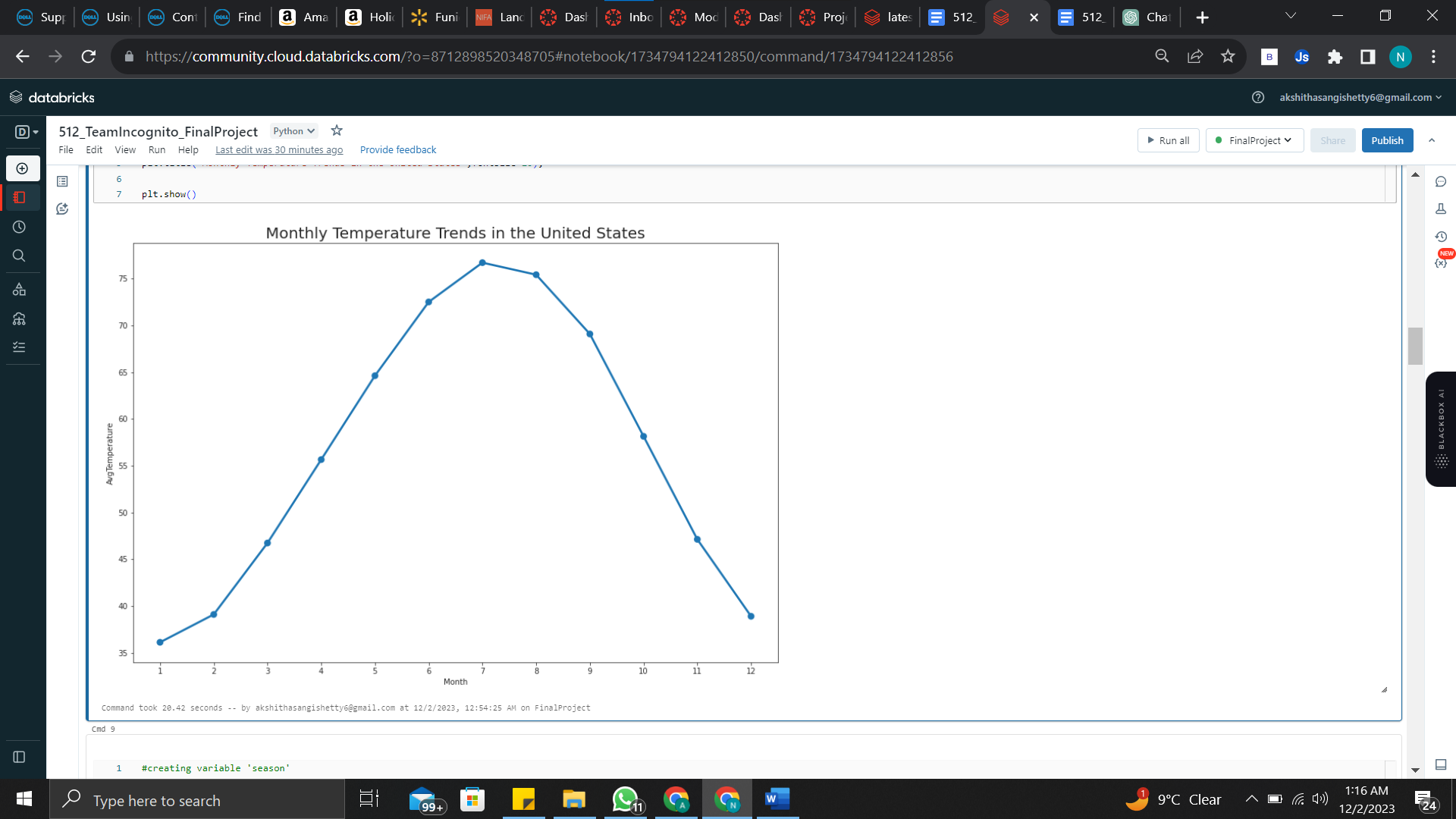
plt.figure(figsize= (15,10))

sns.pointplot(x='Month', y='AvgTemperature', data=usData);

plt.title('Monthly Temperature Trends in the United States',fontsize=20);

plt.show()





#creating variable 'season'

def season(df):

if df in [12,1,2] :

return 'Winter'

elif df in [3,4,5]:

return 'Spring'

elif df in [6,7,8]:

return 'Summer'

elif df in [9,10,11]:

return 'Autumn'

else:

return 'NA'

pd.options.mode.chained\_assignment = None # default='warn'

usData['Season'] = usData['Month'].apply(season)

usData['AvgTemperature']=usData['AvgTemperature'].astype('float64')

usData[['Month' , 'Day' , 'Year']]=usData[['Month' , 'Day' , 'Year']].astype('int64')

#Seasons in Phoenix,AZ

phxData = usData[usData['City'] == 'Phoenix']

#Plot the seasons

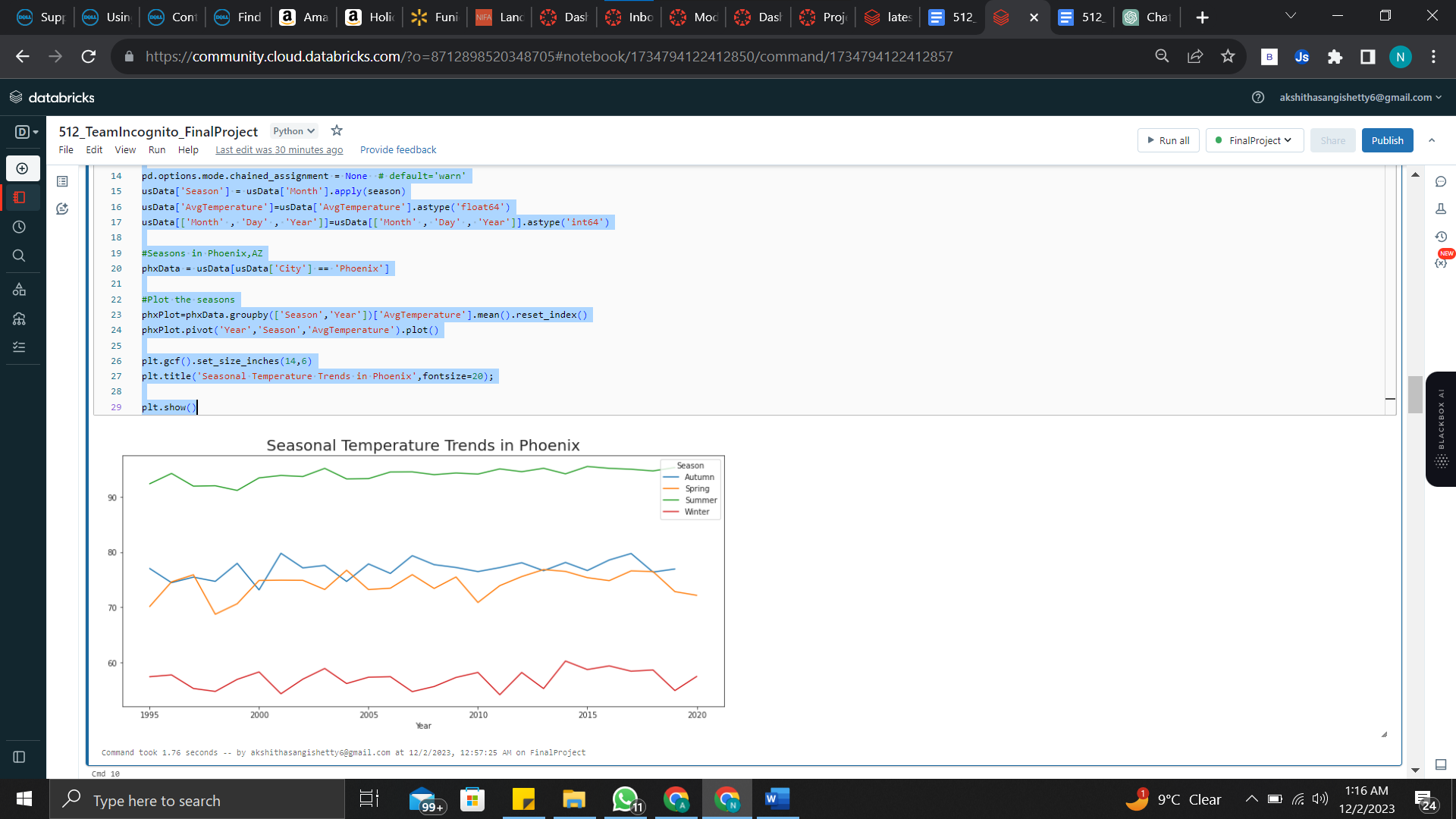
phxPlot=phxData.groupby(['Season','Year'])['AvgTemperature'].mean().reset\_index()

phxPlot.pivot('Year','Season','AvgTemperature').plot()

plt.gcf().set\_size\_inches(14,6)

plt.title('Seasonal Temperature Trends in Phoenix',fontsize=20);

plt.show()



def temp\_type(df):

if (df is None):

return "unknown"

else:

if df < 14:

return "freezing cold"

elif df < 23:

return "very cold"

elif df < 32:

return "cold"

elif df < 50:

return "normal"

elif df < 68:

return "warm"

elif df < 86:

return "hot"

elif df >= 86:

return "very hot"

return "normal"

pd.options.mode.chained\_assignment = None # default='warn'

usData['temp\_type'] = usData['AvgTemperature'].apply(temp\_type)

usData['date'] = pd.to\_datetime(dict(year=usData.Year, month=usData.Month, day=usData.Day))

#temp\_type in phx

phxData = usData[(usData['City'] == 'Phoenix') ]

## replacing outliers to mean

phxData\_2000 = phxData[(phxData['Year'] >= 2000)]

#Plot the temp\_type

phxPlot=phxData\_2000.groupby(['temp\_type','date'])['AvgTemperature'].mean().reset\_index()

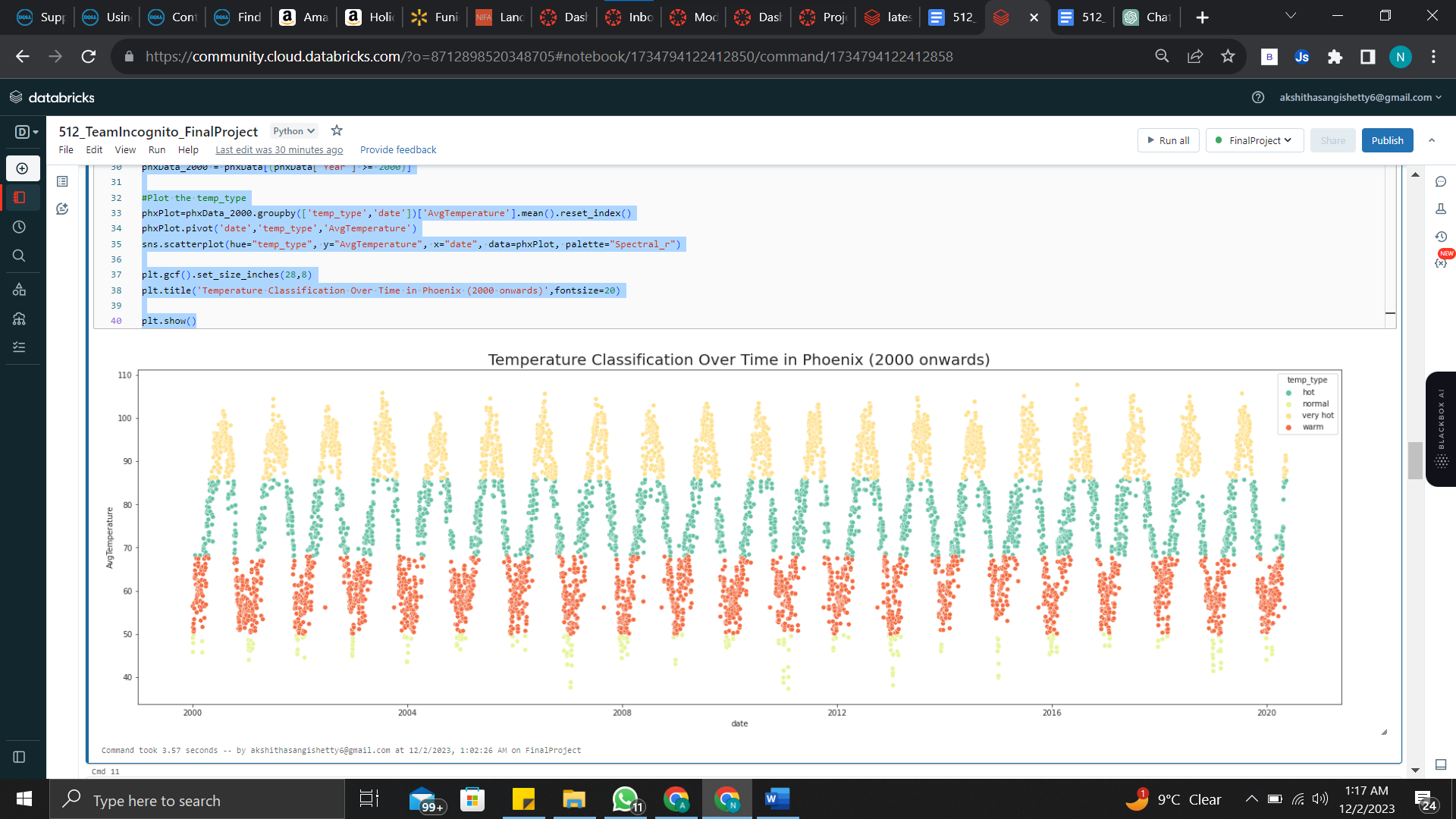
phxPlot.pivot('date','temp\_type','AvgTemperature')

sns.scatterplot(hue="temp\_type", y="AvgTemperature", x="date", data=phxPlot, palette="Spectral\_r")

plt.gcf().set\_size\_inches(28,8)

plt.title('Temperature Classification Over Time in Phoenix (2000 onwards)',fontsize=20)

plt.show()



# region wise distribution

atpz = df.groupby("Region")["AvgTemperature"].mean().sort\_values()[-1::-1]

atpz = atpz.rename({"South/Central America & Carribean":"South America","Australia/South Pacific":"Australia"})

atpz

fig3= plt.figure(figsize = (15,8))

plt.bar(atpz.index,atpz.values)

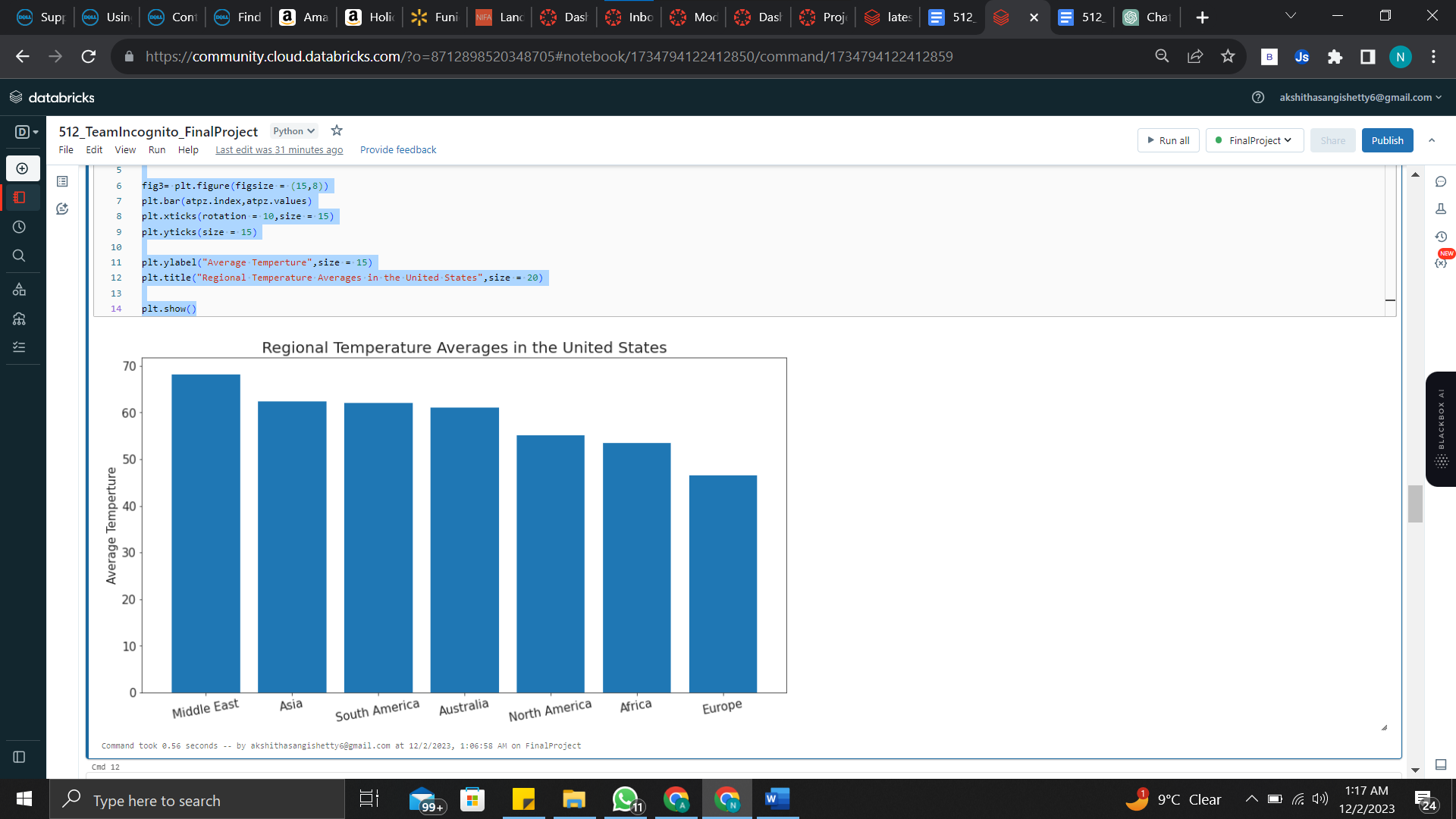
plt.xticks(rotation = 10,size = 15)

plt.yticks(size = 15)

plt.ylabel("Average Temperture",size = 15)

plt.title("Regional Temperature Averages in the United States",size = 20)

plt.show()



#Defining training and testing data

phx = usData[usData["City"] == "Phoenix"]

phx.reset\_index(inplace = True)

phx.drop('index', axis = 1, inplace=True)

phx = phx.drop(['Region', 'Country', 'State','City','Month','Day','Season', 'temp\_type'], axis = 1)

training\_set = phx[phx["Year"] <= 2010]

test\_set = phx[phx["Year"] > 2010]

test\_set1 = phx[phx["Year"] > 2010]

training\_set



#AR model

from statsmodels.tsa.ar\_model import AutoReg

from sklearn.metrics import mean\_squared\_error

model\_AR = AutoReg(training\_set["AvgTemperature"], lags = 1000)

model\_fit\_AR = model\_AR.fit()

predictions\_AR = model\_fit\_AR.predict(training\_set.shape[0], training\_set.shape[0] + test\_set.shape[0] - 1)

import seaborn as sns

fig6= plt.figure(figsize=(15,5))

plt.ylabel("Temperature (F)", fontsize = 20)

plt.plot(test\_set["AvgTemperature"], label = "Original Data")

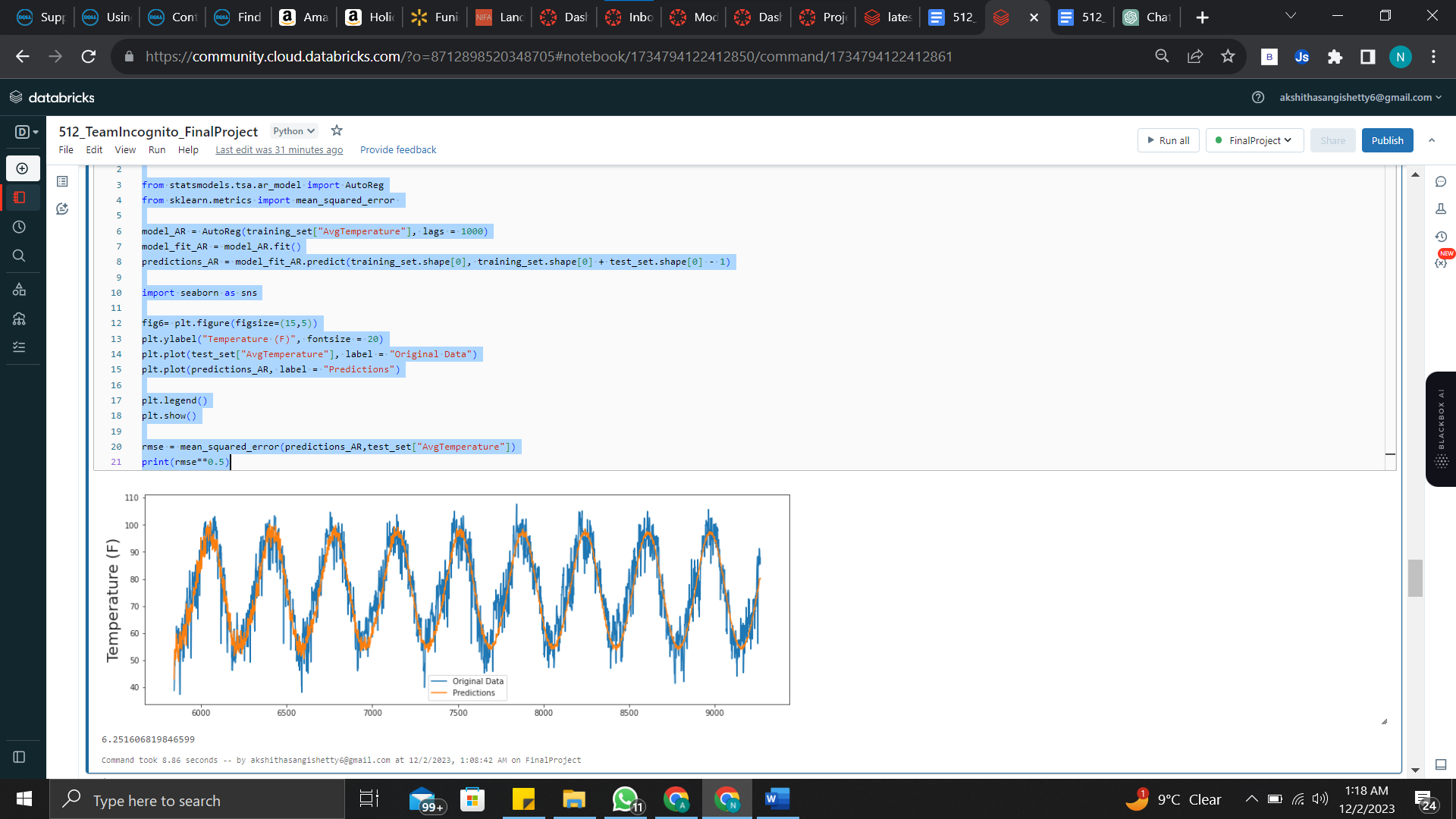
plt.plot(predictions\_AR, label = "Predictions")

plt.legend()

plt.show()

rmse = mean\_squared\_error(predictions\_AR,test\_set["AvgTemperature"])

print(rmse\*\*0.5)



## predicting future values

model\_AR = AutoReg(phxData["AvgTemperature"], lags = 1000)

model\_fit\_AR = model\_AR.fit()

y = model\_fit\_AR.predict(len(phxData["AvgTemperature"]), len(phxData["AvgTemperature"])+500)

print(y)

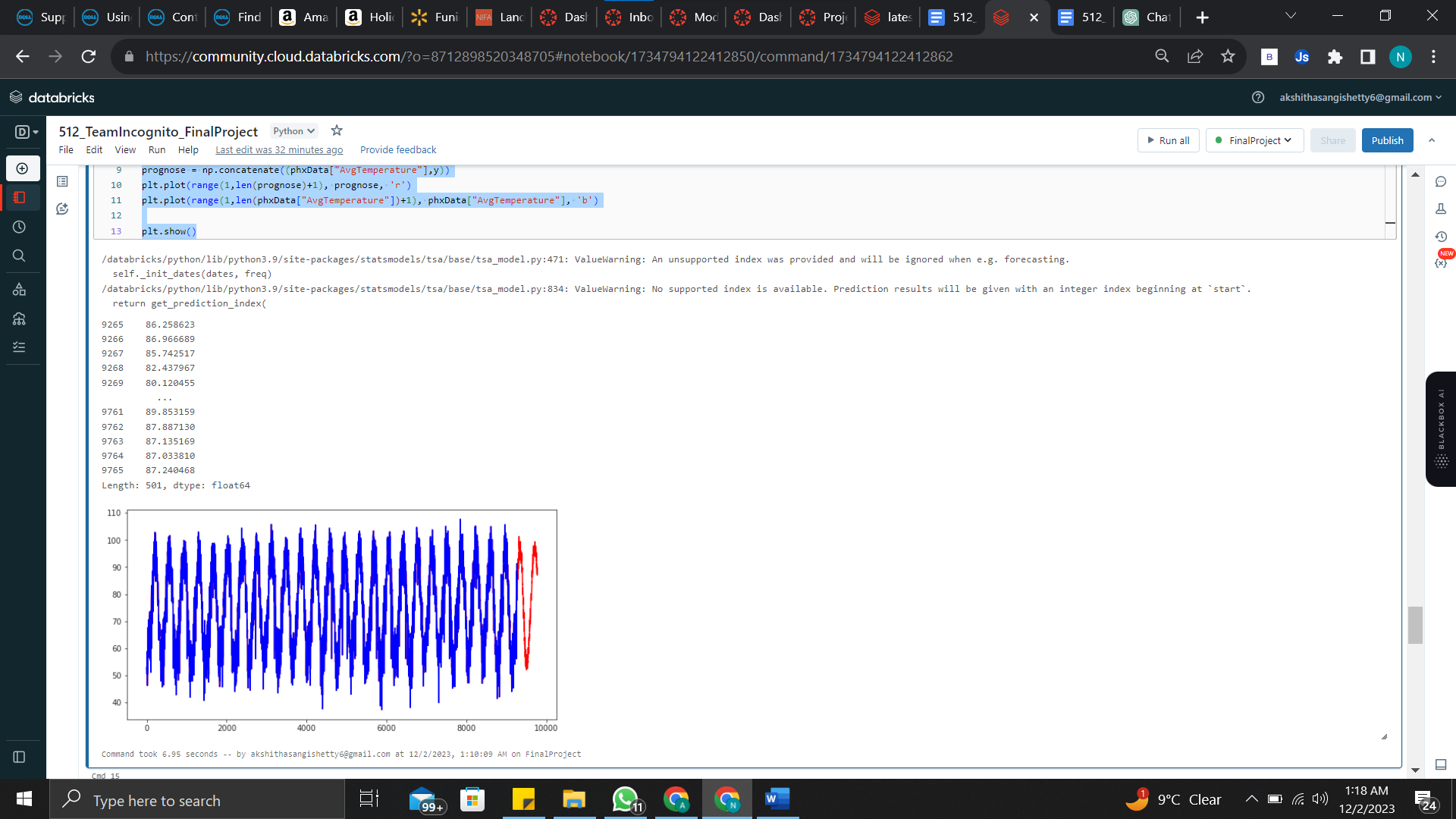
plt.figure(figsize=(10,5))

prognose = np.concatenate((phxData["AvgTemperature"],y))

plt.plot(range(1,len(prognose)+1), prognose, 'r')

plt.plot(range(1,len(phxData["AvgTemperature"])+1), phxData["AvgTemperature"], 'b')

plt.show()



# using spark

from pyspark.sql.functions import col

from pyspark.ml.regression import LinearRegression

import pandas as pd

from pyspark.ml.feature import VectorAssembler

from pyspark.ml.evaluation import RegressionEvaluator

# convert Pandas DataFrame to Spark DataFrame

training\_set\_spark = spark.createDataFrame(training\_set)

test\_set\_spark = spark.createDataFrame(test\_set)

# Assemble the feature vector

assembler = VectorAssembler(inputCols=["AvgTemperature"], outputCol="features")

training\_features = assembler.transform(training\_set\_spark).select(col("features"), col("date"), col("AvgTemperature").alias("label"))

test\_features = assembler.transform(test\_set\_spark).select(col("features"), col("date"), col("AvgTemperature").alias("label"))

# Train the linear regression model

lr = LinearRegression(featuresCol="features", labelCol="label" ,elasticNetParam=0.5, regParam=0.01)

model = lr.fit(training\_features)

##dt = DecisionTreeRegressor(featuresCol="features", labelCol="label", maxDepth=5)

##model = dt.fit(training\_features)

# Make predictions on the test set

predictions = model.transform(test\_features).orderBy("date")

# Convert predictions to Pandas DataFrame for plotting

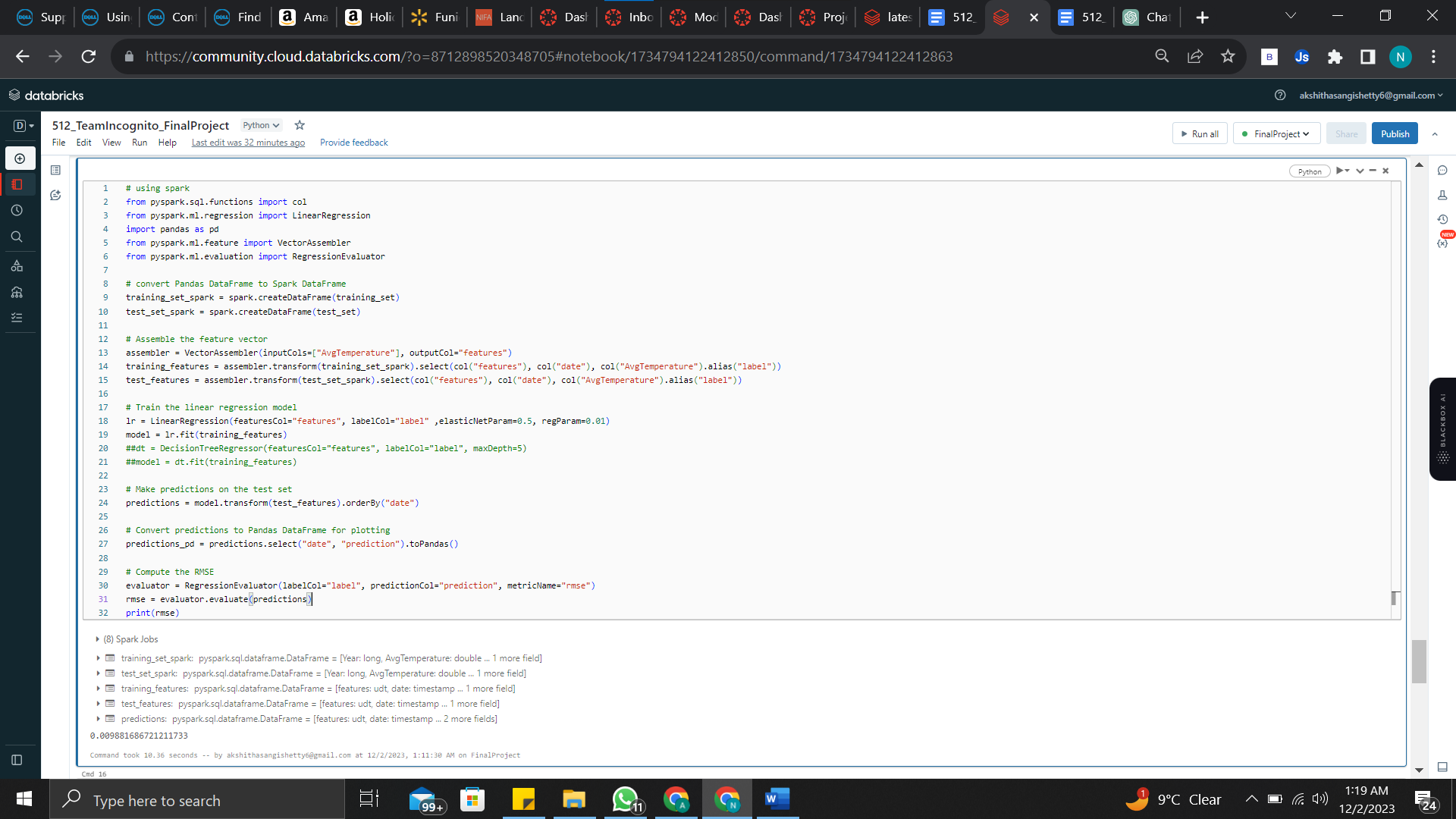
predictions\_pd = predictions.select("date", "prediction").toPandas()

# Compute the RMSE

evaluator = RegressionEvaluator(labelCol="label", predictionCol="prediction", metricName="rmse")

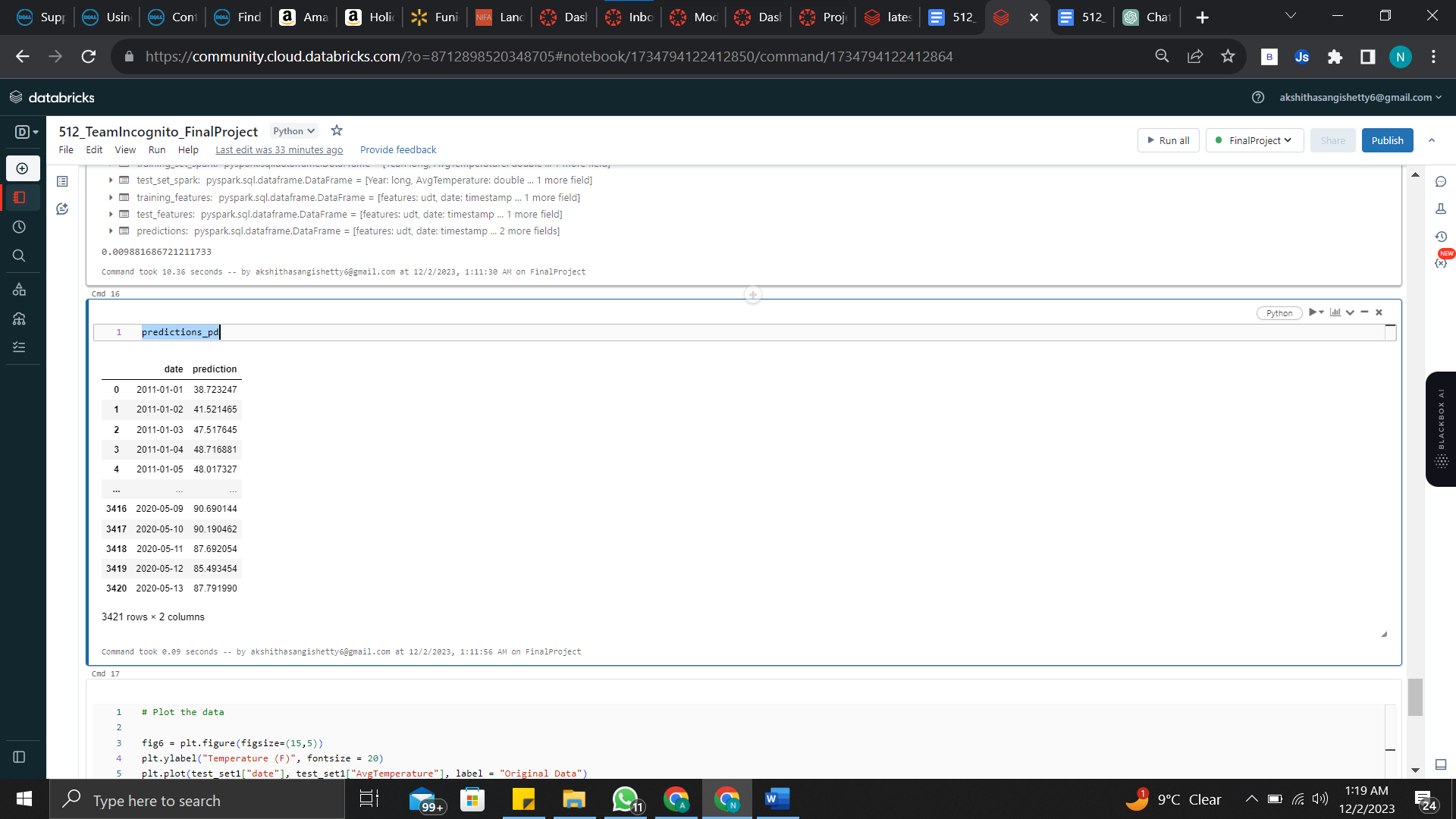
rmse = evaluator.evaluate(predictions)

print(rmse)



rmse: 0.009881686721211733

predictions\_pd



# Plot the data

fig6 = plt.figure(figsize=(15,5))

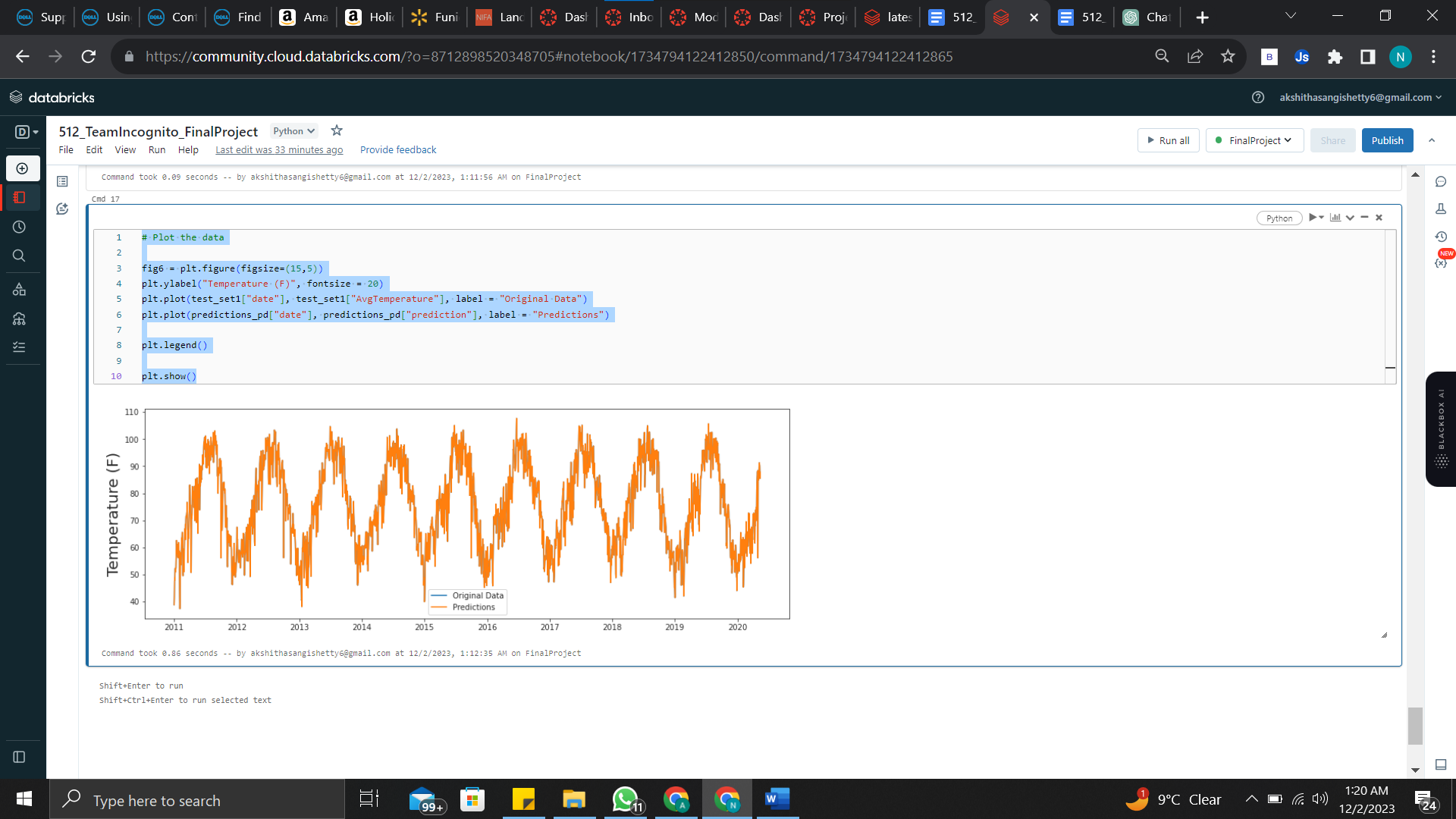
plt.ylabel("Temperature (F)", fontsize = 20)

plt.plot(test\_set1["date"], test\_set1["AvgTemperature"], label = "Original Data")

plt.plot(predictions\_pd["date"], predictions\_pd["prediction"], label = "Predictions")

plt.legend()

plt.show()



**Section 6: Summary & Conclusions**

**SUMMARY**

This project aims to leverage PySpark technology to analyze weather data comprehensively. The data has been pre-processed, and key insights are emphasized through graphical displays. These include temperature comparisons across four cities, trends in monthly average temperature, seasonal variations in Phoenix's temperatures, visualization of Phoenix's average temperature using a custom function, a national examination of average regional temperatures, assessment of test errors using an autoregression model, and predictions of future temperatures.

Finding important insights into weather trends and patterns is the ultimate objective. Meteorologists, urban planners, and transit providers, among other stakeholders, can benefit greatly from these insights. Our project is aligning with the CRISP-DM framework to guarantee a methodical approach. This entails steps such as comprehending the business, establishing models, carefully preparing the data, evaluating the results, and implementing the conclusions.

A wide range of meteorological variables are included in the dataset that is being examined, including temperature, precipitation, wind speed, humidity, and atmospheric pressure.PySpark, Python, and several data visualization tools are part of the technological toolbox for this project. The goal of this effort is to optimize operations across many industries and sectors, improve decision-making concerning weather-related matters, and streamline planning for the benefit of the business.

**CONCLUSION**

The ultimate objective is to extract insightful information about trends and patterns in the weather. For stakeholders including meteorologists, urban planners, and transit providers, these insights are priceless. We're integrating the project with the CRISP-DM framework to guarantee a methodical approach. Phases include business comprehension, data comprehension, careful data preparation, model construction, rigorous evaluation, and, at the end, deploying the findings are all part of this process.

The dataset being examined includes a wide range of meteorological factors, including temperature, precipitation, wind speed, humidity, and air pressure.PySpark, Python, and a number of data visualization tools are among the technology instruments used in this project. With this project, we hope to improve organizational performance by streamlining planning, improving weather-related decision-making, and streamlining operations across many sectors and industries.

**PUBLISHED NOTEBOOK**

<https://databricks-prod-cloudfront.cloud.databricks.com/public/4027ec902e239c93eaaa8714f173bcfc/8712898520348705/1734794122412850/8738741871089754/latest.html>

**REFERENCES**

Data Science for Business What You Need to Know About Data Mining and Data-Analytic Thinking (Foster Provost, Tom Fawcett) (z-lib.org)

<https://www.datascience-pm.com/crisp-dm-2/>

https://www.kaggle.com/datasets/sudalairajkumar/daily-temperature-of-major-cities

<https://www.kaggle.com/code/mayahalevy/unsupervised-learning-analysis-weather-data>

<https://link.springer.com/article/10.1007/s11831-021-09616-4#:~:text=The%20analysis%20of%20atmospheric%20data,Furthermore%2C%20these%20data%20are%20disorganized>

<https://www.kaggle.com/code/krystianadammolenda/weather-dataset-cleaning-regression>