Cal State Fullerton

Analyzing Supermarket Sales Trends: Leveraging AWS EMR, Spark, and QuickSight for Insightful Visualizations

CPSC - 531 ADVANCED DATABASE MANAGEMENT SYSTEMS
PROF: TSENG-CHING JAMES SHEN,PhD

Team members:
Ajaykumar Burigari
Dinesh Daki

CONTENTS

Project Statement:

This project aims to predict weekly sales for supermarkets while analyzing the factors influencing sales. This analysis will inform strategies for optimizing inventory management and resource allocation.

Dataset Overview:

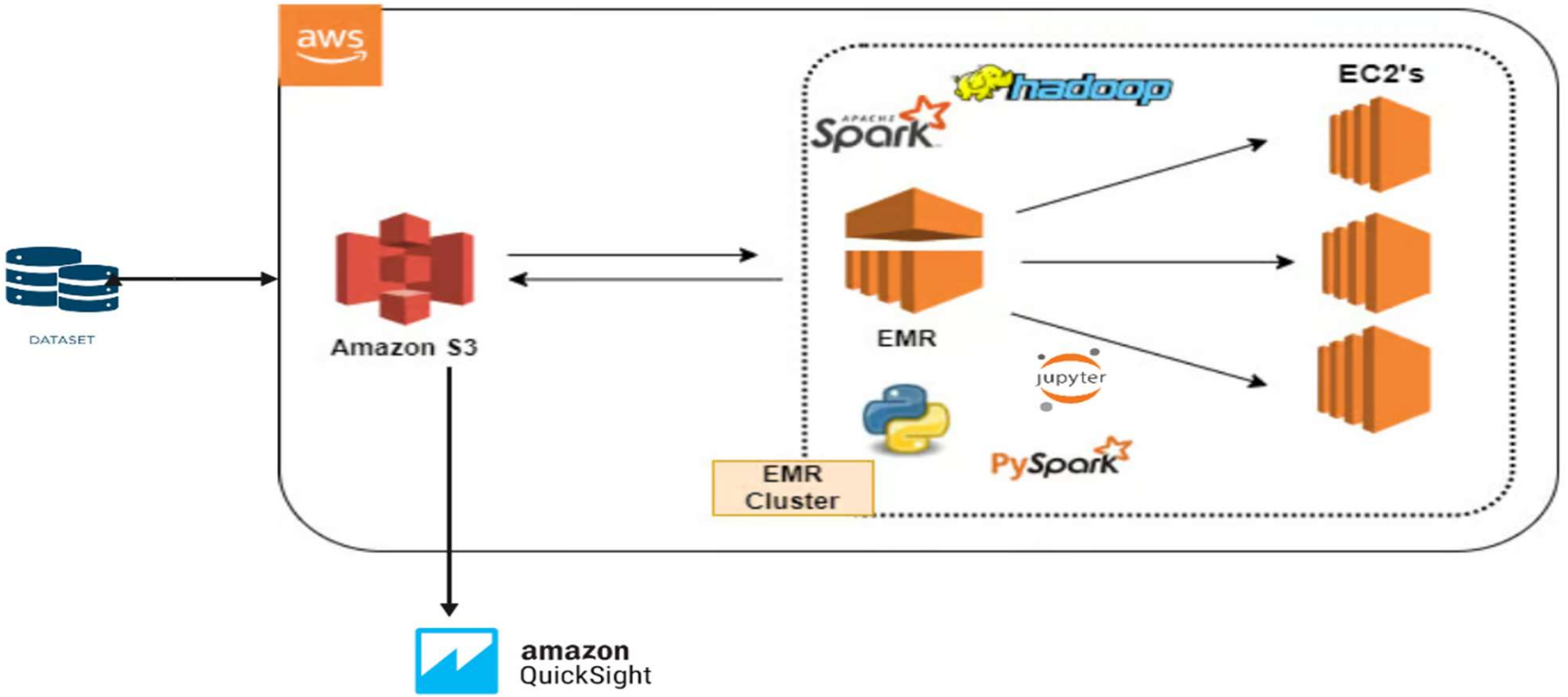
- The dataset provided comprises historical sales data for 45 Walmart stores situated across various regions.
- The dataset includes the following files:
- **stores.csv**: This file contains anonymized information about the 45 stores, specifying their type and size.
- **train.csv**: Historical training data covering the period from 2010-02-05 to 2012-11-01. Fields in this file include:
    - Store: Store number
    - Dept: Department number
    - Date: Week
    - Weekly_Sales: Sales for the given department in the specified store
    - IsHoliday: Indicates whether the week is a special holiday week

Dataset Overview:

- **test.csv**: Similar to train.csv, but with sales data withheld.
- **features.csv**: Contains additional data related to the store, department, and regional activity for the given dates.

  Fields include:
  - Store: Store number
  - Date: Week
  - Temperature: Average temperature in the region
  - Fuel_Price: Cost of fuel in the region
  - CPI: Consumer Price Index
  - Unemployment: Unemployment rate
  - IsHoliday: Indicates whether the week is a special holiday week

Architecture:

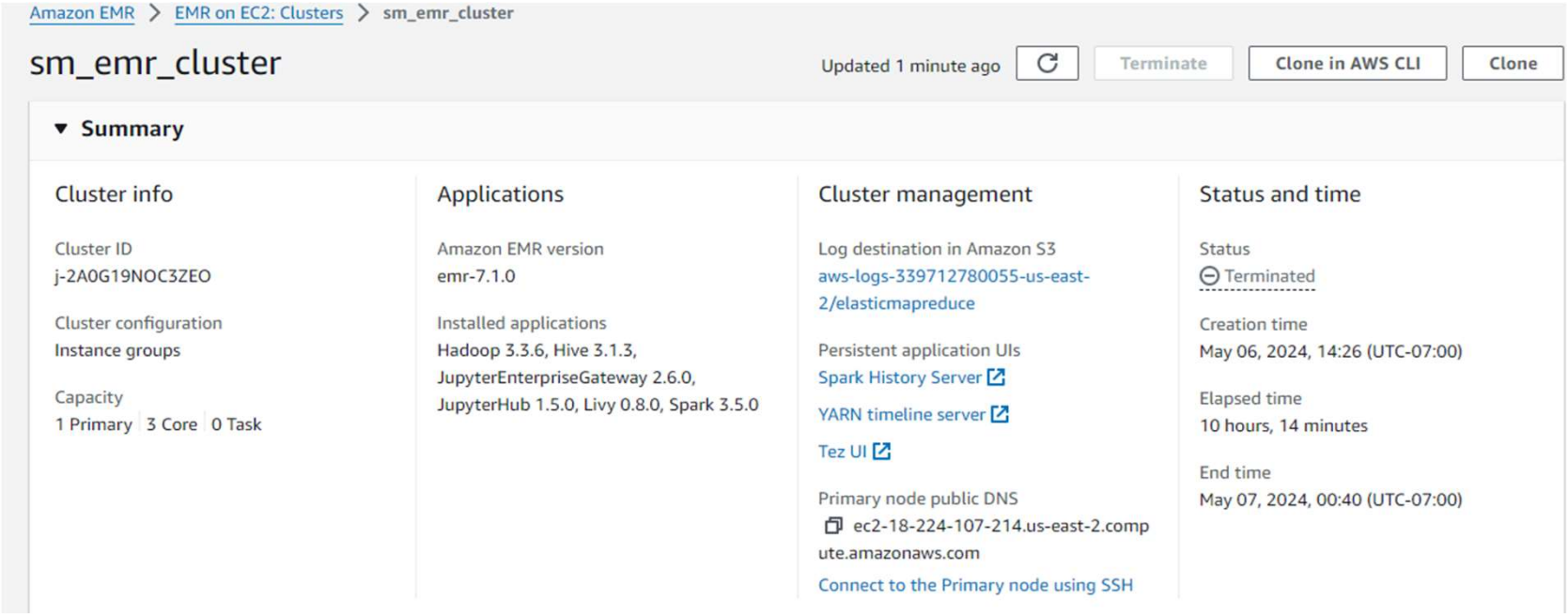


Implementation Approach:

- Creating an EMR cluster
- Creating S3 bucket
- Creating EMR studio within EMR cluster
- Fetching data from S3 Bucket
- Creating Quicksight and connecting it to S3 bucket.

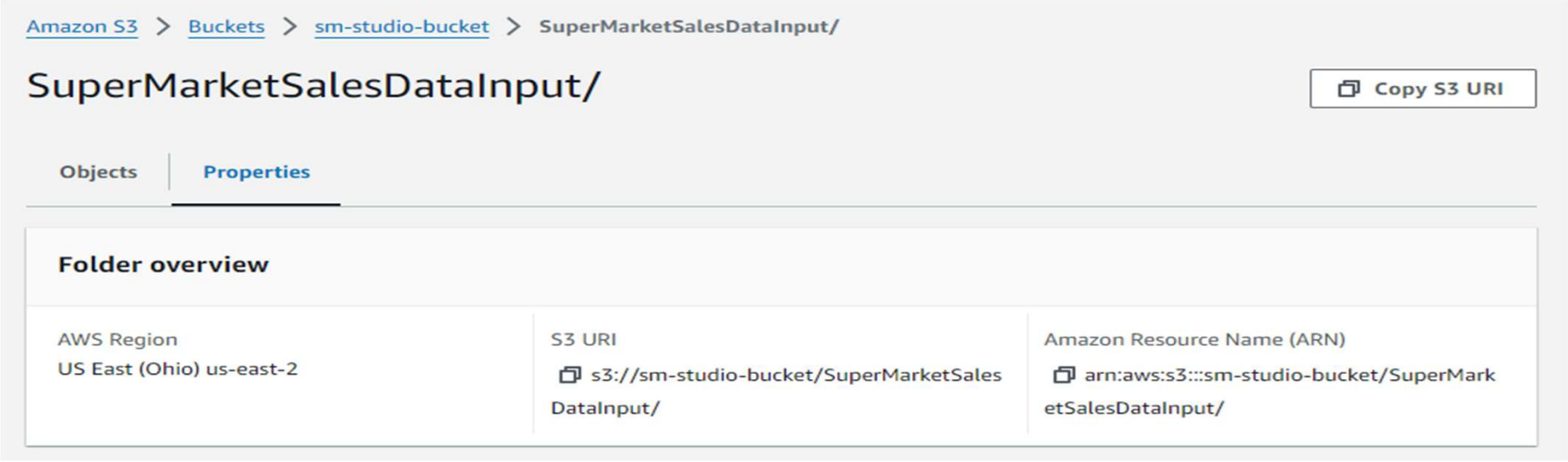
Creating EMR Cluster:

- created an EMR cluster with cluster size of min 3 instances and max 10 instances and 8 core nodes.
- Installed applications like Spark, Hadoop, JupyterEnterpriseGateway which helps in processing and analyzing large datasets.



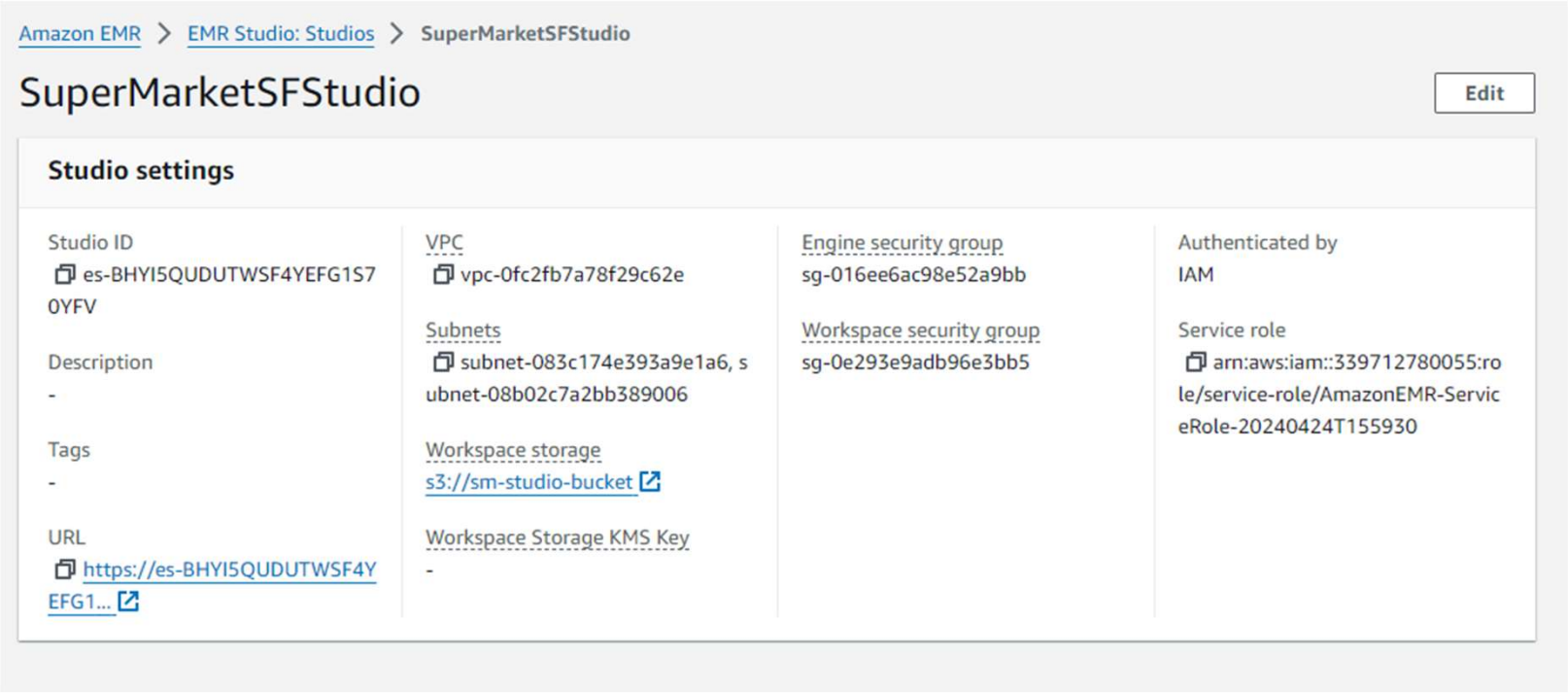
Creating S3 Bucket:

- Create an S3 bucket, This S3 bucket acts as a Storage location for EMR studio.
- Upload the Dataset into the S3 Bucket
- Use S3 URI to read dataset files in EMR Studio



Creating EMR Studio within EMR Cluster:

- Create EMR Studio
- Connect EMR Studio to S3 Bucket for dataset Storage access



- Create Spark Session with name SuperMarketSalesForecast
- Creating spark dataframes from csv files containing train, store, feature and test data

[57]:
```
import pyspark
from pyspark.sql import SparkSession

# Create a SparkSession
spark = SparkSession.builder.appName("SuperMarketSalesForecast").getOrCreate()

# Read the CSV files and create Spark DataFrames
train_df = spark.read.csv('s3://sm-studio-bucket/SuperMarketSalesDataInput/train.csv', header=True, inferSchema=True)
store_df = spark.read.csv('s3://sm-studio-bucket/SuperMarketSalesDataInput/stores.csv', header=True, inferSchema=True)
feature_df = spark.read.csv('s3://sm-studio-bucket/SuperMarketSalesDataInput/features.csv', header=True, inferSchema=True)
test_df = spark.read.csv('s3://sm-studio-bucket/SuperMarketSalesDataInput/test.csv', header=True, inferSchema=True)
```
```
Last executed at 2024-05-06 20:41:14 in 11.29s
```

- Joining train and test data dataframes with the store and feature dataframes based on columns ‘store’, ‘date’, ‘isHoliday’ to create train and test data.

```
#join the train and test data with the store and features data
train = train_df.join(store_df, on='Store', how='left').join(feature_df, on=['Store', 'Date', 'IsHoliday'], how='left')
test = test_df.join(store_df, on='Store', how='left').join(feature_df, on=['Store', 'Date', 'IsHoliday'], how='left')

# Show the first few rows of the merged DataFrames (optional)
train.show(5)
test.show(5)
```

```
+-----+----------+---------+----+------------+----+------+-----------+----------+----------+------------+
|Store|      Date|IsHoliday|Dept|Weekly_Sales|Type|  Size|Temperature|Fuel_Price|       CPI|Unemployment|
+-----+----------+---------+----+------------+----+------+-----------+----------+----------+------------+
|    1|2010-02-05|    false|   1|     24924.5|   A|151315|      42.31|     2.572|211.0963582|       8.106|
|    1|2010-02-12|     true|   1|    46039.49|   A|151315|      38.51|     2.548|211.2421698|       8.106|
|    1|2010-02-19|    false|   1|    41595.55|   A|151315|      39.93|     2.514|211.2891429|       8.106|
|    1|2010-02-26|    false|   1|    19403.54|   A|151315|      46.63|     2.561|211.3196429|       8.106|
|    1|2010-03-05|    false|   1|     21827.9|   A|151315|       46.5|     2.625|211.3501429|       8.106|
+-----+----------+---------+----+------------+----+------+-----------+----------+----------+------------+
only showing top 5 rows

+-----+----------+---------+----+----+------+-----------+----------+----------+------------+
|Store|      Date|IsHoliday|Dept|Type|  Size|Temperature|Fuel_Price|       CPI|Unemployment|
+-----+----------+---------+----+----+------+-----------+----------+----------+------------+
|    1|2012-11-02|    false|   1|   A|151315|      55.32|     3.386|223.4627793|       6.573|
|    1|2012-11-09|    false|   1|   A|151315|      61.24|     3.314|223.4813073|       6.573|
|    1|2012-11-16|    false|   1|   A|151315|      52.92|     3.252|223.5129105|       6.573|
|    1|2012-11-23|     true|   1|   A|151315|      56.23|     3.211|223.5619474|       6.573|
|    1|2012-11-30|    false|   1|   A|151315|      52.34|     3.207|223.6109842|       6.573|
+-----+----------+---------+----+----+------+-----------+----------+----------+------------+
```

- Created 3 regression models RandomForest, Gradient Booster Trees, Linear Regression.
- Predictions are made on the training data with each model and RMSE is computed.
- Each model is trained on the training data and predictions are made.
- Based on the RMSE values obtained, the code selects the model with lowest RMSE as the best model.

```
# Step: Model Evaluation
evaluator = RegressionEvaluator(labelCol="Weekly_Sales", predictionCol="prediction", metricName="rmse")

# Evaluate Random Forest
rf1_predictions = rf1_model.transform(train_data)
rf1_rmse = evaluator.evaluate(rf1_predictions)
print("Random Forest RMSE on training data:", rf1_rmse)

# Evaluate Gradient Boosted Trees (GBT)
gbt_predictions = gbt_model.transform(train_data)
gbt_rmse = evaluator.evaluate(gbt_predictions)
print("GBT RMSE on training data:", gbt_rmse)

# Evaluate Linear Regression
lr_predictions = lr_model.transform(train_data)
lr_rmse = evaluator.evaluate(lr_predictions)
print("Linear Regression RMSE on training data:", lr_rmse)
```

- Predictions are made on the test data using best model.

```
# Predict on test data using the best model
best_model_predictions = best_model.transform(test_data)
best_model_predictions.show(5)
```

Last executed at 2024-05-06 20:43:22 in 751ms

▸ Spark Job Progress

```
+-----+----------+---------+----+----+------+-----------+----------+-----------+------------+----+-----+----+------------------
--+------------------+
|Store|      Date|IsHoliday|Dept|Type|  Size|Temperature|Fuel_Price|        CPI|Unemployment|week|month|year|            featur
es|        prediction|
+-----+----------+---------+----+----+------+-----------+----------+-----------+------------+----+-----+----+------------------
--+------------------+
|    1|2012-11-02|    false|   1|   1|151315|      55.32|     3.386|223.4627793|       6.573|  44|   11|2012|[1.0,0.0,1.0,1.
0,...|33993.018829402834|
|    1|2012-11-09|    false|   1|   1|151315|      61.24|     3.314|223.4813073|       6.573|  45|   11|2012|[1.0,0.0,1.0,1.
0,...|33993.018829402834|
|    1|2012-11-16|    false|   1|   1|151315|      52.92|     3.252|223.5129105|       6.573|  46|   11|2012|[1.0,0.0,1.0,1.
0,...| 34453.29276501201|
|    1|2012-11-23|     true|   1|   1|151315|      56.23|     3.211|223.5619474|       6.573|  47|   11|2012|[1.0,1.0,1.0,1.
0,...| 34831.95169389938|
|    1|2012-11-30|    false|   1|   1|151315|      52.34|     3.207|223.6109842|       6.573|  48|   11|2012|[1.0,0.0,1.0,1.
0,...| 35501.71924763424|
+-----+----------+---------+----+----+------+-----------+----------+-----------+------------+----+-----+----+------------------
--+------------------+
only showing top 5 rows
```

- Saving the predictions made by the best regression model into a folder located at the specified amazon S3 bucket path.

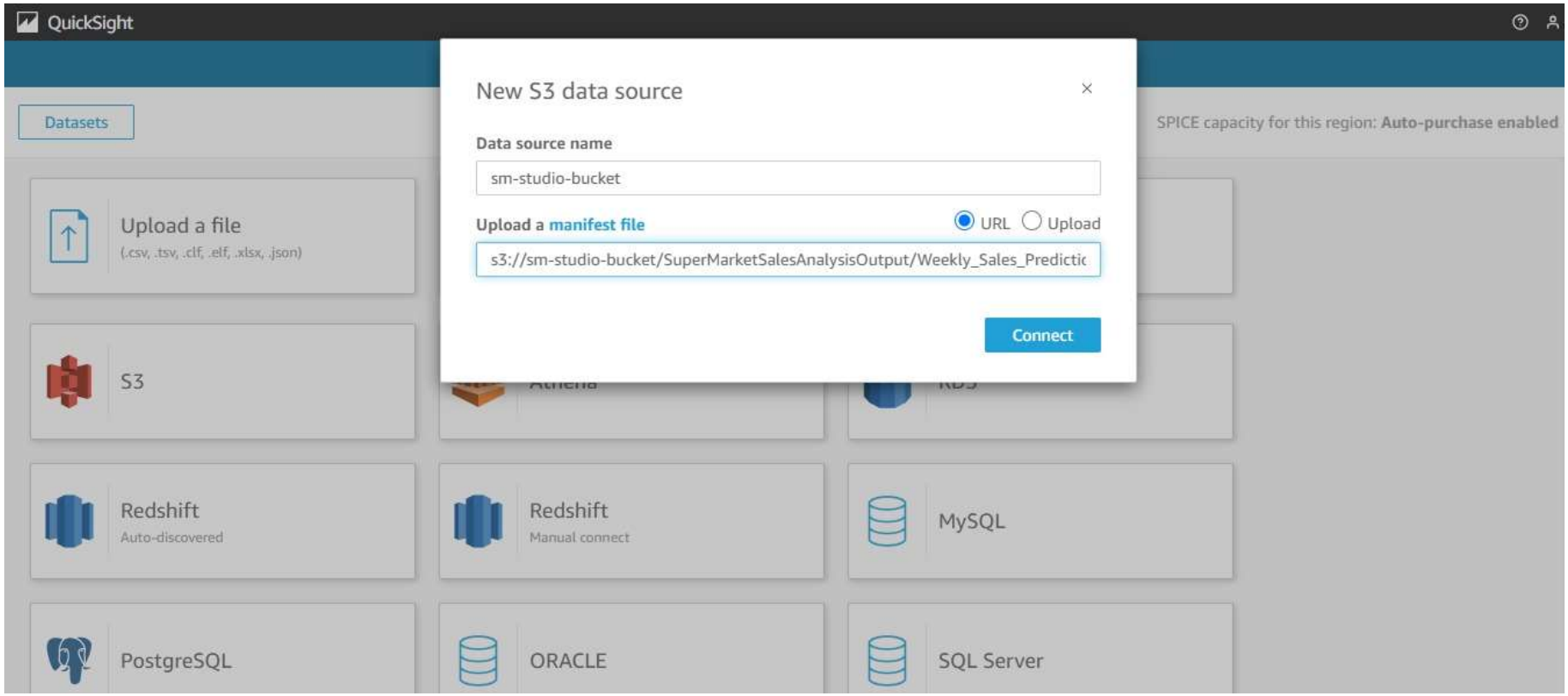
```
output_path = "s3://sm-studio-bucket/SuperMarketSalesAnalysisOutput/Weekly_Sales_Prediction"

# Write DataFrame to CSV format
best_model_predictions.write.csv(output_path, mode='overwrite', header=True)

# Print confirmation message
print("DataFrame saved as CSV to S3:", output_path)
```

Last executed at 2024-05-06 20:46:43 in 3.27s

▸ Spark Job Progress

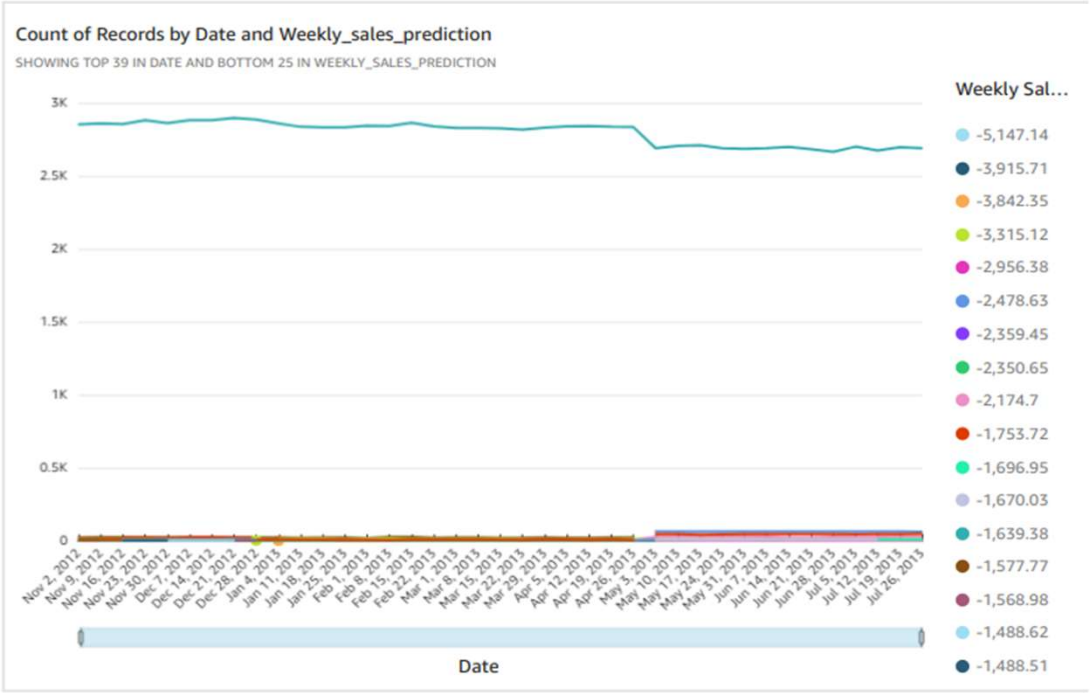
```
DataFrame saved as CSV to S3: s3://sm-studio-bucket/SuperMarketSalesAnalysisOutput/Weekly_Sales_Prediction
```

Connecting S3 Bucket to Quicksight:



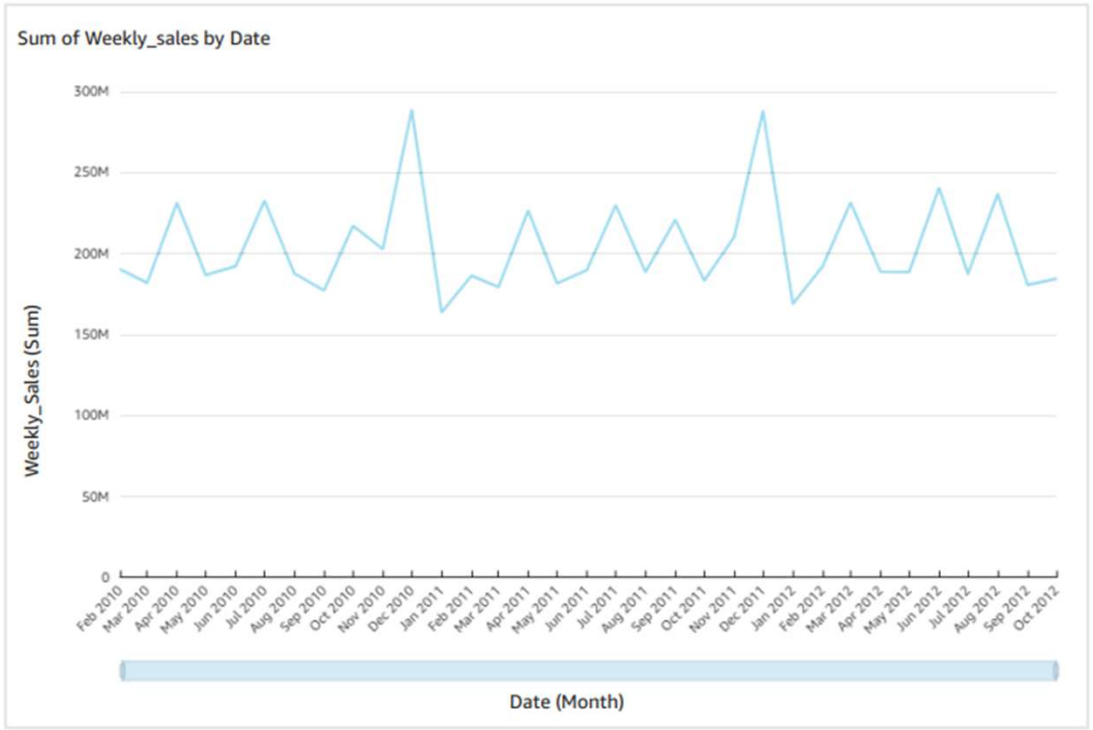
Results:

- Sales Predicted



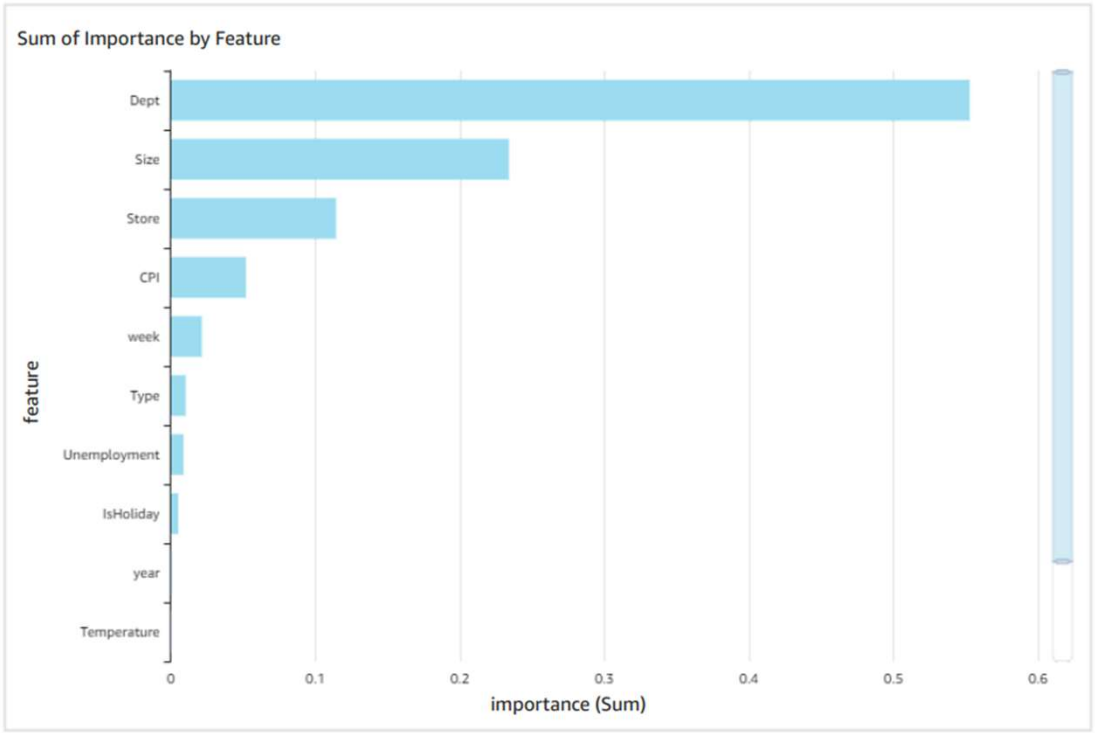
Results:

- Sales Obtained from the trained data



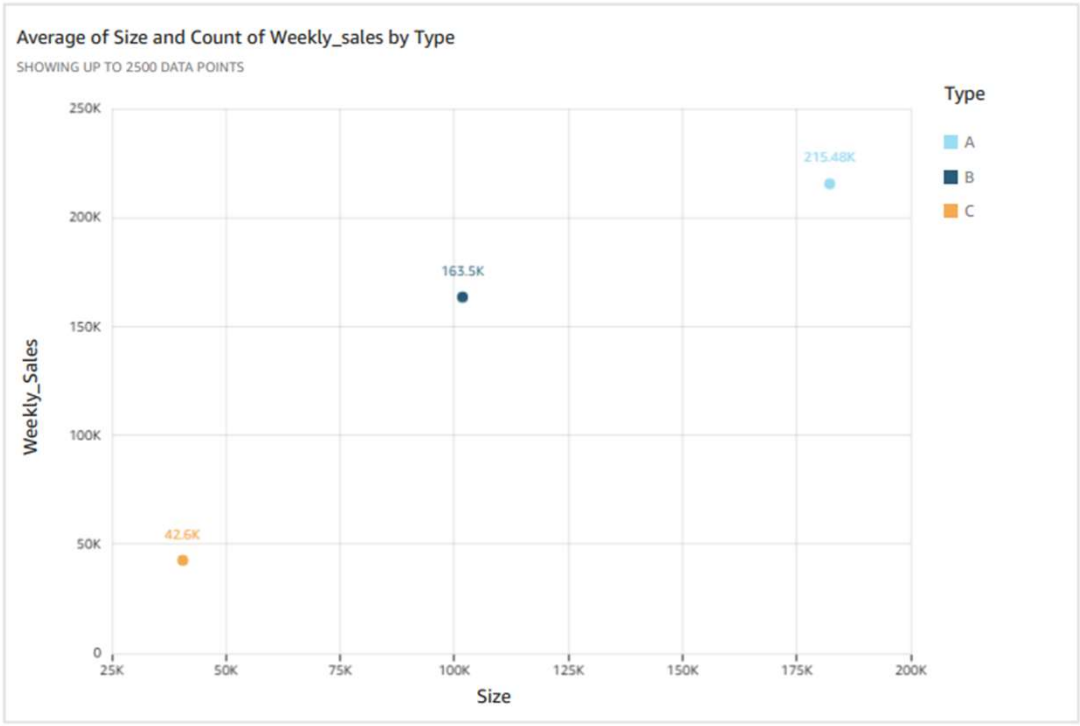
Results:

- Feature importance



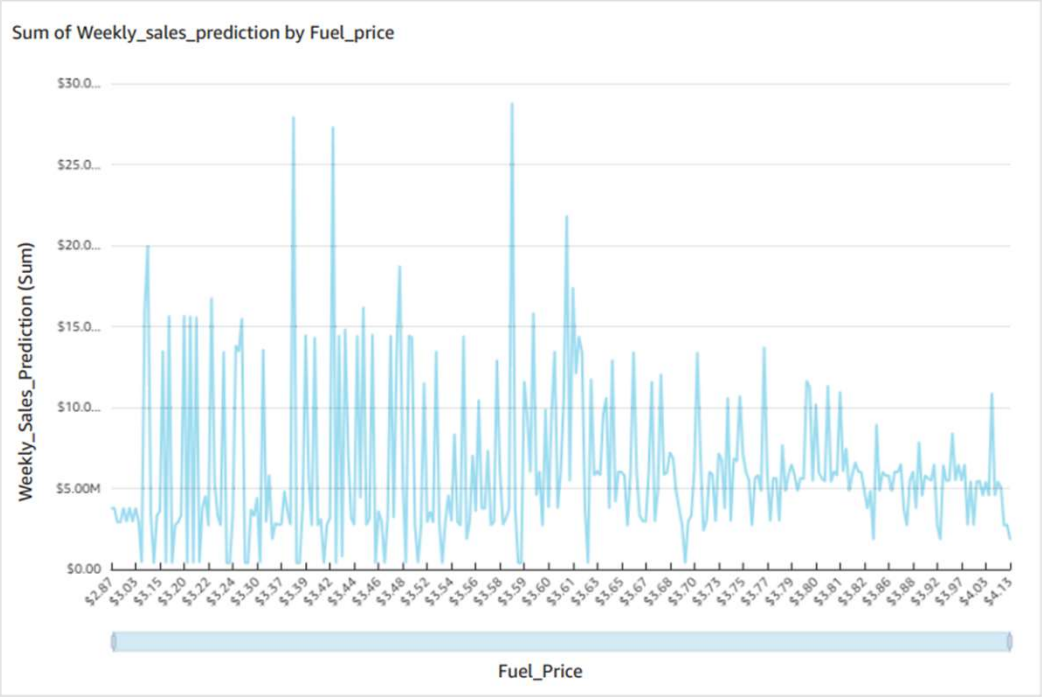
Analysis:

As the size of the store increases, so do its weekly sales.



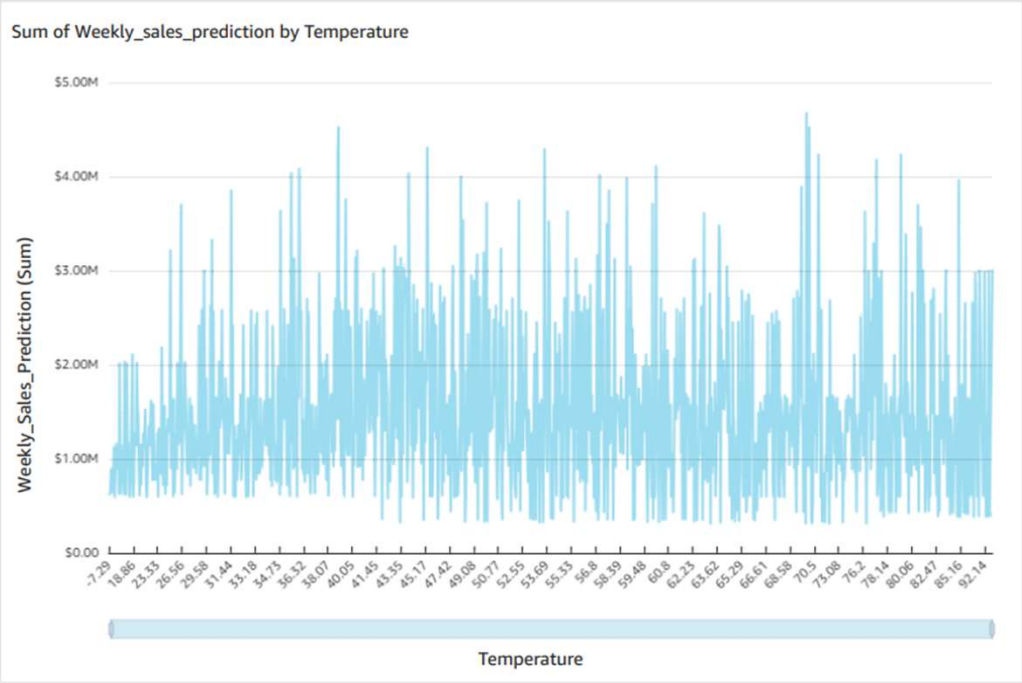
Analysis:

Fuel prices and weekly sales share a correlation, with sales demonstrating a decline during periods of higher fuel costs.



Analysis:

Temperature and weekly sales exhibit a correlation, with sales showing lower figures during both the highest and lowest temperatures, while demonstrating higher sales during moderate temperatures.



Tools and Technologies used:

- **AWS Ecosystem:**
  AWS EMR
  Amazon S3
  EMR Studio
  AWS QuickSight

- **Big Data Technologies:**
  Apache Spark and PySpark

- **Notebook:**
  Jupyter Notebook

THANK YOU