Machine Learning - 1 Project Report

G. Neha (IMT2014018), Meghana Kotagiri (IMT2014034)
Vadari Dakshayani(IMT2014061)

1 Project Title

Our project is titled 'Store Sales Forecasting For Weather Sensitive Products'.

2 Problem Statement

To predict the sales of potentially weather-sensitive products (like umbrellas, bread and milk) around the time of major weather events (a weather event is any day in which more than an inch of rain or two inches of snow was observed) at dierent retail locations.

3 Problem Description

Supermarket stores like Walmart would like to know how much of each product is sold. It will help them to know how much of each product should be put in stock. This is especially true in times of extreme weather events. Intuitively, one can expect that the sales of umbrellas before a big thunderstorm would increase. But it's difficult for replenishment managers to correctly predict the level of inventory needed to avoid being out-of-stock or overstock during and after that storm. This can be true for any other product that is potentially weather sensitive. We are predicting the sales of products using Walmart data for different retail locations. This project is a problem posted on Kaggle for Walmart.

Link to the Kaggle Problem: Store Sales Forecasting For Weather Sensitive Products

4 Data Description

We have been given Walmart data across 45 stores, selling a total of 111 products. We have also been given sales data about how much of each product has been sold in a particular Walmart Store on a particular day, for three years. This dataset consists of around 6 lakh rows. In addition to the sales data, we have also been given weather information. The weather information gives us details about the weather station nearest to a given store. The weather information also has data for each day, that describes the amount of precipitation, rainfall etc.

To be more clear, the exact data description of all the attributes in our data is described as follows:

1. Sales Data

Attribute Name	Description
Store number	This is the store number for a particular Walmart Store.
Item number	This represents an item or a product.
Date	This represents the date on which a particular store sold an item.
Units	This is the sale of given item number in the given store number on the given date.

Altogether, each row of the sales data represents the item which was sold by a store on a particular day.

2. Weather Data: The main weather dataset consists of a lot of attributes, but the ones that we found relevant for our problem are shown below.

Attribute Name	Description
Date	The date when these weather parameters were recorded.
Station number	This is weather station number.
Total precipitation	This attribute represents the total amount of precipitation observed for a weather station.
Rainfall	This attribute represents the total amount of rainfall observed in a weather station.
Depart Flag	This says if there was a deviation from normal weather or not on that particular day.

3. Key Data: The relational mapping between stores and the weather stations that cover them.

Attribute Name	Description
Store number	This is the store number for a particular Walmart Store.
Station number	This is weather station number.

5 Data Preparation

- There were many store-item combinations in the training data for which units sold was zero for entire span of three years. We labelled them as Out of Sale items for all the stores in the training data. This narrowed down training space from 111*45 store-item combinations to just 255 of them, which had non zero sales data in the training set.
- Because of the evaluation criterion being RMSLE (root mean square logarithmic error), did log(1+x) transformation on the sales data.
- As the problem is related to sales, we thought it is intuitive for people to buy products on weekends/holidays. Therefore, we added new attributes like: isHoliday, isWeekend, and isWeekday.
- Data merging: We merged the sales data with the weather data based on station number.
- Discretization of weather attributes like precipitotal flag indicates that the precipitation in that area was over 0.8 etc.

6 Data Understanding

We visualised our data to explore patterns in the data. We plotted the amount of sales (i.e., the total number of products sold) of a given item in a given Walmart store. We observed two kinds of patterns, as shown in Figure 1 and Figure 2. The graphs also show the seasonal, trend and residual components, but we for our analysis, the first plot is necessary.

1. Stationary data: In figure 1, we can see that the mean of the data does not change over time. This kind of distributions, where statistical properties like mean or variance don't change over time are called **Stationary Distributions**.

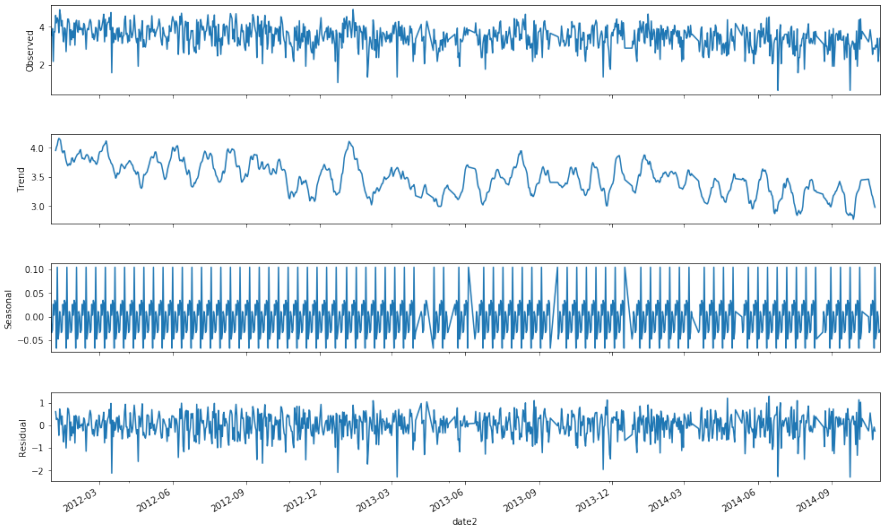

Figure 1: stationary time series: store 36, item 9

2. Non-Stationary data: In figure 2, we can see that the mean of the data changes over time. This kind of distributions, where statistical properties are different at different times are called **Non - Stationary Distributions**.

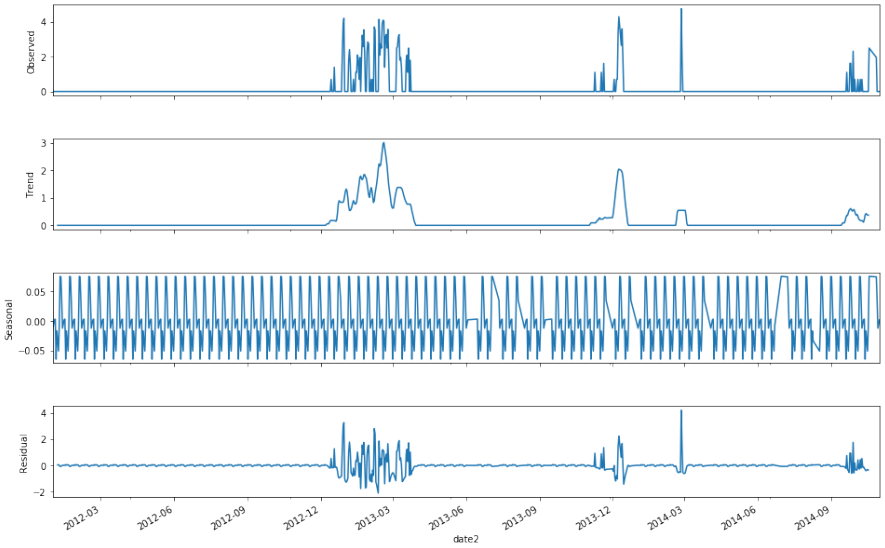

Figure 2: stationary time series: store 14, item 15

We divided our dataset into two datasets, one comprising of Stationary data and the other consisting of Non-Stationary data. We created models separately for both of them.

Dickey Fuller Test: The null hypothesis of the test is that the time series can be represented by a unit root, that it is not stationary (has some time-dependent structure). The alternate hypothesis (rejecting the null hypothesis) is that the time series is stationary.

If accepted, it suggests the time series has a unit root, meaning it is non-stationary. It has some time dependent structure.

If null hypothesis is rejected it suggests the time series does not have a unit root, meaning it is stationary. It does not have time-dependent structure.

For a given sales data for a particular item and store in the training data, we use this test to distinguish between Stationary and Non stationary data. Then we divide it into two different datasets.

7 Modeling

We divided valid store-item combinations in the training dataset to 2 datasets: stationary store-item combinations and non stationary store-item combinations as explained above.

- For stationary store-item combinations:

  Created ARMA model for each such combination (in total 156 ARMA models). In ARMA, AR is a model that uses the dependent relationship between an observation and some number of lagged observations and MA is a model that uses the dependency between an observation and a residual error from a moving average model applied to lagged observations. After fitting ARMA model, we plotted ACF (Auto Correlation Function) graph for the residues (predicted sales -actual sales) for one store-item combination as shown in figure 3 below:

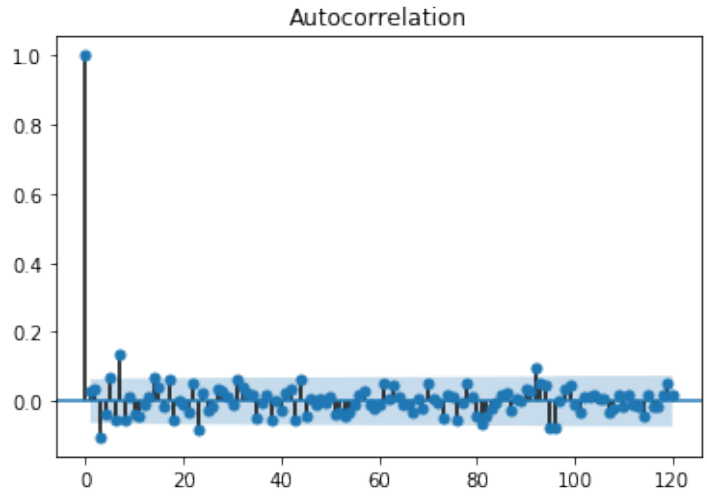

Figure 3: ACF plot of residues

The blue band in the image above is 95% confidence interval. Most of the points are within the blue band and the plot shows no correlation between points. This confirms that the residues are noise, which indicates that the model fit the data well.

- For non stationary store-item combinations: As the series was found to be non-stationary, we realized these combinations have trend component in them. We decided to use simple regressor model using the following features:

  store nbr, item nbr, station nbr, preciptotal flag, depart flag, weekday, is weekend, is holiday, day, month, year, rolling mean of sales (with window size 21 to capture local trend)

Comments on features:

- Though, the problem says the data is about weather sensitive products, weather features are not effective at all (correlation with sales is as low as 0.001)
- monthly periodicity is observed in some store-item pairs
- mostly sales of item look like iid

8 Evaluation Metric

Root Mean Squared Logarithmic Error (RMSLE), which is given as:

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}((log(p_i+1)-log(a_i+1))^2}$$

$$=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(log\frac{(p_i+1)}{(a_i+1)})^2}$$

Justification for the choice: We can see that RMSLE depends on percentage difference. This implies the following:

We will penalize significantly when the actual value is small and the difference between predicted and actual value is huge. Though, we don't want to penalize heavily when the difference between predicted and actual value is huge and actual value & predicted values are huge themselves.

This is in line with our intuition. When you have a small store, you have a few customers and each customer is important. So, it is important that each one is satisfied. This is in contrast to a big store, where missing few customers is tolerable.

9 Evaluation

On a side note, we used a few other techniques to solve this problem as listed below:

1. Random Forest Regressor using the same set of features as mentioned in the previous section on complete data.
2. KNN (with n=7 to capture weakly trends) model for every store
3. Clustering based approach: Instead of creating models for each store item combination, we clustered the data using above features and then fitted above model for each cluster.

Final RMSLE Scores:

Sno	Approach	Score
1	ARMA + REGRESSION (156+1 MODELS)	0.114
2	RANDOM FOREST (1 MODEL)	0.102
3	KNN (45 MODELS)	0.135
4	CLUSTERING BASED APPROACH	0.448

Comments: The one issue that we faced in approach (1) is that we had to create 156 ARMA models, and it was not feasible to select best order (parameter in ARMA) manually. Hence, we took the same order for all the ARMA models. This might be a place where we can improve accuracy by selecting optimal parameter for each model.

Also, the code folder we submitted along this report contains code for approach (1) and (2) only, as they have comparatively better score.