Pro

Arka Ghosh

arkaghosh.nb@gmail.com

Abstract

This document outlines the steps undertaken to analyze data and build various ML models for predicting the sales for the problem statement defined in MachineHack 2021 competition.

Analytics olympiad ‘21

Machine Learning Solution Overview

Contents

[**Import Python Libraries** 2](#_Toc87700429)

[**Instantiate H2O server** 2](#_Toc87700430)

[**Attempts to start and/or connect to and H2O instance.** 2](#_Toc87700431)

[**Read data from Train and Test datasets** 2](#_Toc87700432)

[**Convert H2O frame to Pandas dataframe** 3](#_Toc87700433)

[**Exploratory Data Analysis** 3](#_Toc87700434)

[**Profile Report** 3](#_Toc87700435)

[**Correlation Matrix** 4](#_Toc87700436)

[**Check Missing values** 4](#_Toc87700437)

[**View Interactions** 5](#_Toc87700438)

[**Frequency Plot(Histograms)** 5](#_Toc87700439)

[Sales 5](#_Toc87700440)

[Item Weight 6](#_Toc87700441)

[Item MRP 7](#_Toc87700442)

[**Box n Whiskers Plot** 8](#_Toc87700443)

[**Times Series Analysis** 9](#_Toc87700444)

[Combined Sales 10](#_Toc87700445)

[Sales by Outlet 10](#_Toc87700446)

[Sales by Outlet Size 11](#_Toc87700447)

[Sales by Outlet Location Type 11](#_Toc87700448)

[Sales by Item Type 12](#_Toc87700449)

[**Time Series Forecast – Prophet** 12](#_Toc87700450)

[5-year predictions 12](#_Toc87700451)

[Upcoming Trend 13](#_Toc87700452)

[**Statistical Analysis** 14](#_Toc87700453)

[**Normality Distribution Tests** 14](#_Toc87700454)

[**Quantile-Quantile Plot** 14](#_Toc87700455)

[**Shapiro-Wilk Test** 15](#_Toc87700456)

[**Anderson-Darling Test** 15](#_Toc87700457)

[**Spearman Rank Correlation** 16](#_Toc87700458)

# **Import Python Libraries**

Firstly, we will need to install many libraries in Python that we are going to use in our analysis. Apart from a few common ones like Pandas, NumPy, ScipySklearn, we also need to import modules for particular algorithms like Linear Regression, XGBoost, H20 etc. Each of these libraries need to be first installed on our computer by doing ‘pip install <python-module>. I have done this analysis on my Anaconda – Jupyter Notebook. So, the installation can be done by launching a command prompt in the same tool.

# **Instantiate H2O server**

H2O is an open source, in-memory, distributed, fast, and scalable machine learning and predictive analytics platform that allows you to build machine learning models on big data and provides easy productionization of those models in an enterprise environment. H2O’s AutoML can be used for automating the machine learning workflow, which includes automatic training and tuning of many models within a user-specified time-limit.

## **Attempts to start and/or connect to and H2O instance.**

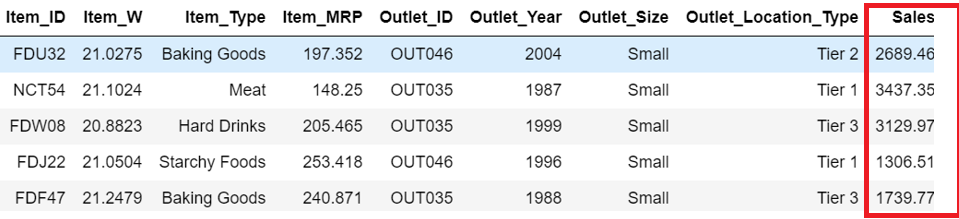
**# max\_mem\_size** - A character string specifying the maximum size, in bytes, of the memory allocation pool to H2O. This value must a multiple of 1024 greater than 2MB.

# Append the letter m or M to indicate megabytes, or g or G to indicate gigabytes.

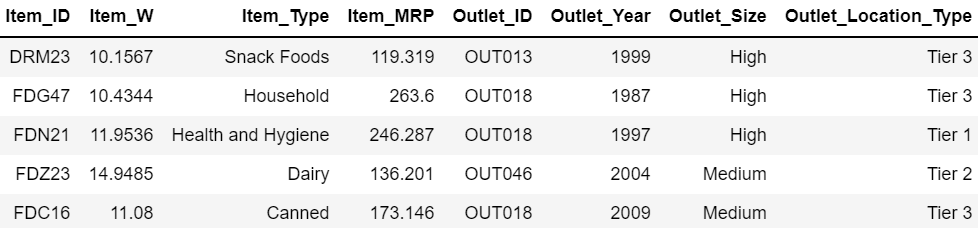
**# nthreads -** Number of threads in the thread pool. This relates very closely to the number of CPUs used. -1 means use all CPUs on the host (Default). A positive integer specifies the number of CPUs directly.

# **Read data from Train and Test datasets**

The import function is a parallelized reader and pulls information from the server from a location specified by the client. The path is a server-side path. This is a fast, scalable, highly optimized way to read data. H2O pulls the data from a data store and initiates the data transfer as a read operation.

**train**:

**test**:



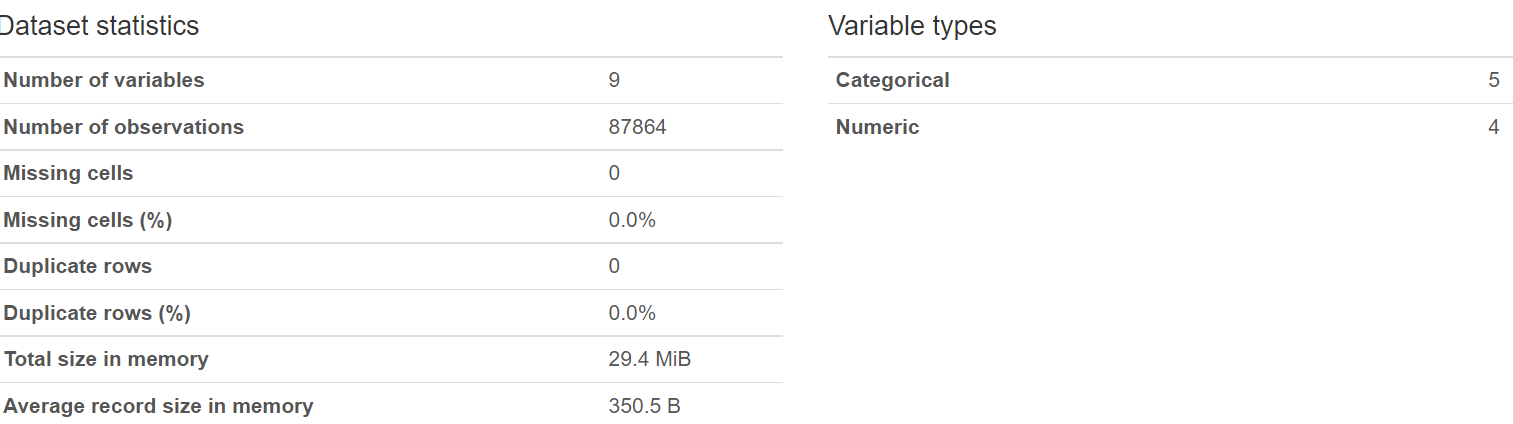
# **Convert H2O frame to Pandas dataframe**

To do data processing and feature engineering, we would like to work on a normal Pandas dataframe. We would also like to apply some base algorithms on this data. So we will convert H2O dataframe to a Pandas dataframe. In the final stage, when we will do a stacked-ensemble modelling, we will convert it back to a H2O dataframe.

# **Exploratory Data Analysis**

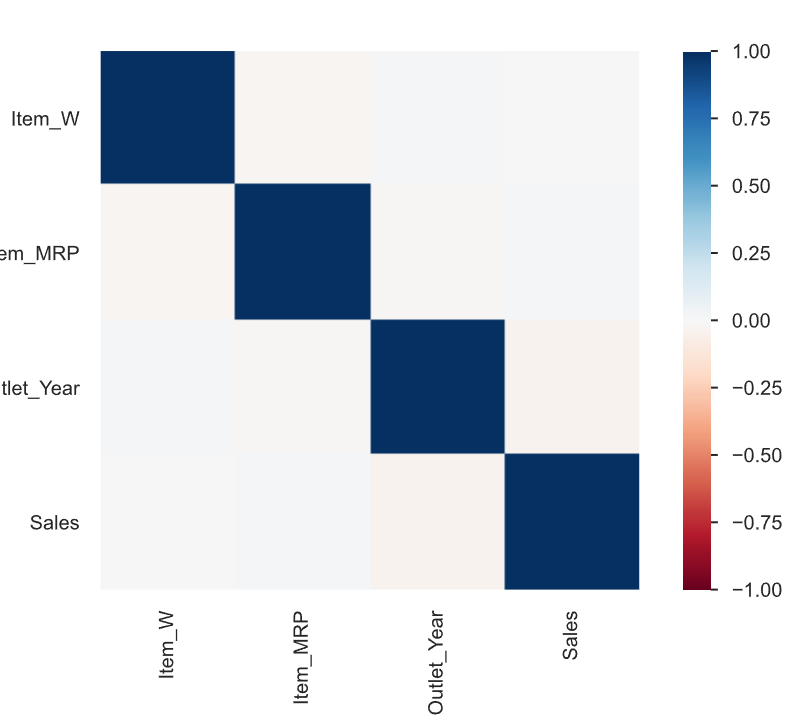
## **Profile Report**

The pandas df.describe() function is great but a little basic for serious exploratory data analysis. pandas\_profiling extends the pandas DataFrame with df.profile\_report() for quick data analysis.



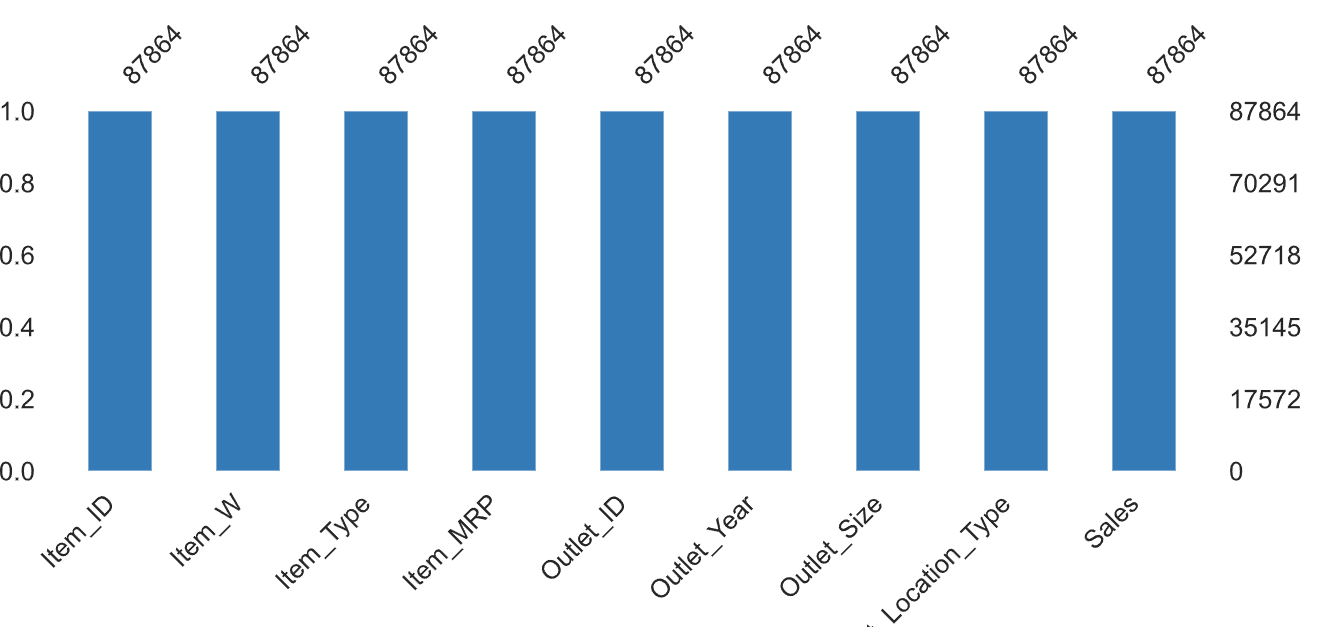
## **Correlation Matrix**

For the numeric features, we will build a correlation matrix, just to verify if there is any particular features that has a high degree of association(positive or negative) with Sales.

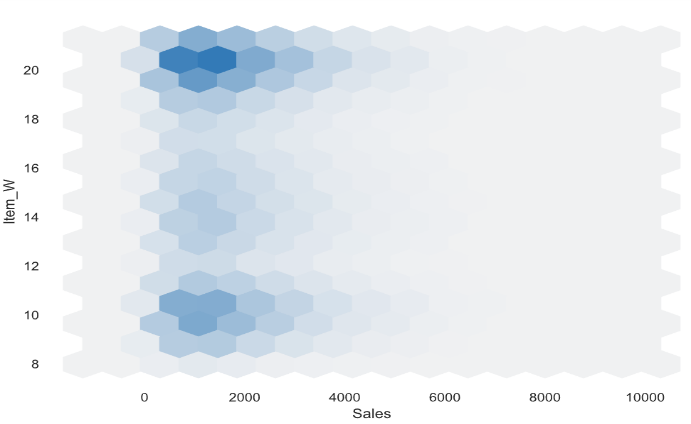
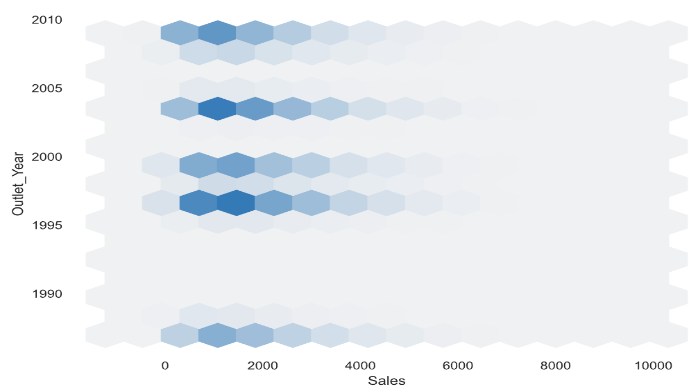
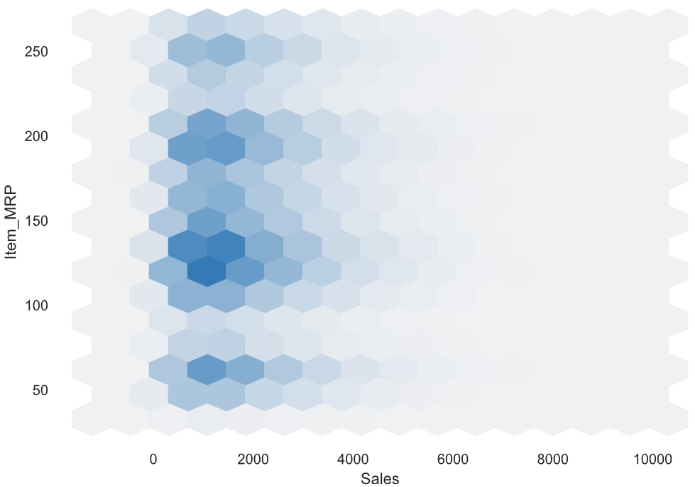
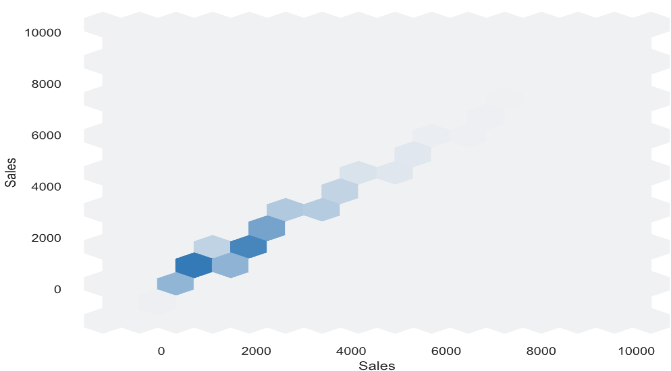


## **Check Missing values**

Sometimes, datasets have few columns with high percentage of missing values or NaN values. These values need to be treated before applying any ML algorithm. We can do a mean-imputation if the count of such missing values is small. But if a column has too many NA values, its better to drop it from the dataframe. Here, we don’t see any missing values , since all columns have exactly 87864 records.



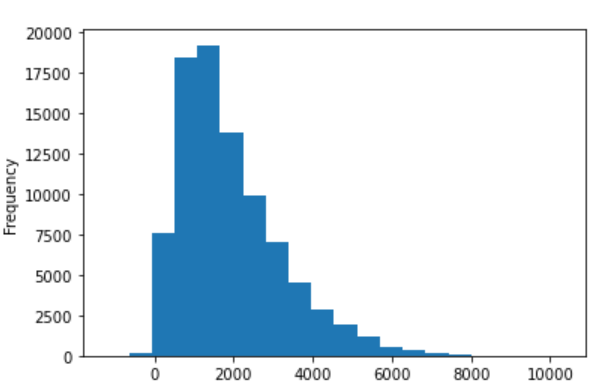
## **View Interactions**

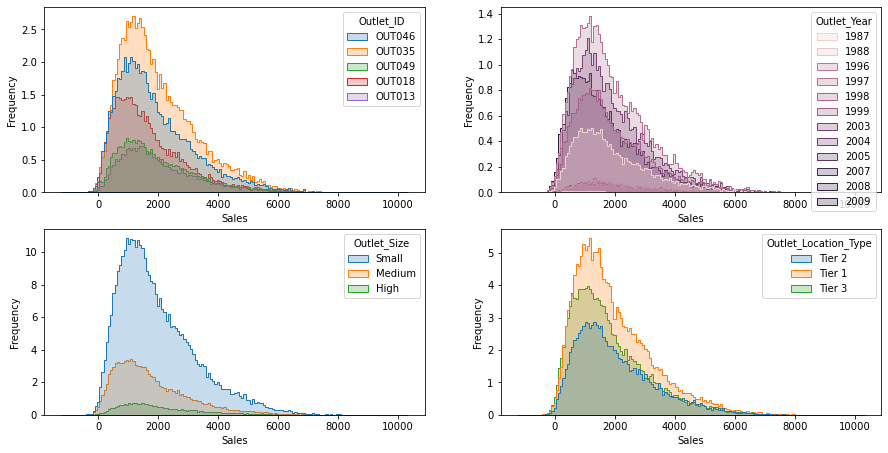
Plot any 2 columns to see how the data is distributed on x-axis and y-axis. If we want to see Sales plotted against other numeric features, we can do it.  

## **Frequency Plot(Histograms)**

### Sales

A histogram is a bar graph-like representation of data that buckets a range of outcomes into columns along the x-axis. The y-axis represents the number count or percentage of occurrences in the data for each column and can be used to visualize data distributions.

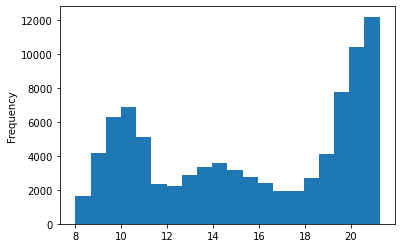
We have created a histogram with 20 bins for Sales. Most frequent occurrence is the range between Rs.1000 - 2000. This means that maximum products that are being sold fall in this range. Beyond this , the frequency of sales is on a downward slope. Next, we plot the frequency charts of Sales as per Outlet\_ID, Outlet\_Year, Outlet\_Size and Outlet\_Location\_Type.

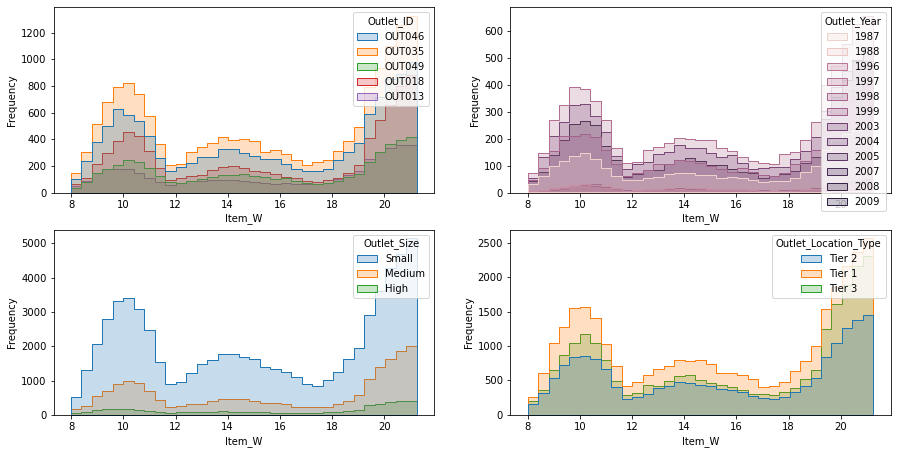


### Item Weight

We have created a histogram with 20 bins for Item Weight. Most frequent occurrence is when the items weighs 10, 14 and 22. This means that maximum item-weights of products that are being sold fall in this range. Beyond this , the frequency of Item Weight can vary.

Next, we plot the frequency charts of Item Weight as per Outlet\_ID, Outlet\_Year, Outlet\_Size and Outlet\_Location\_Type.

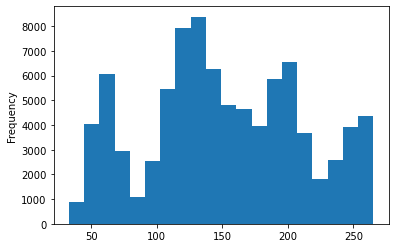


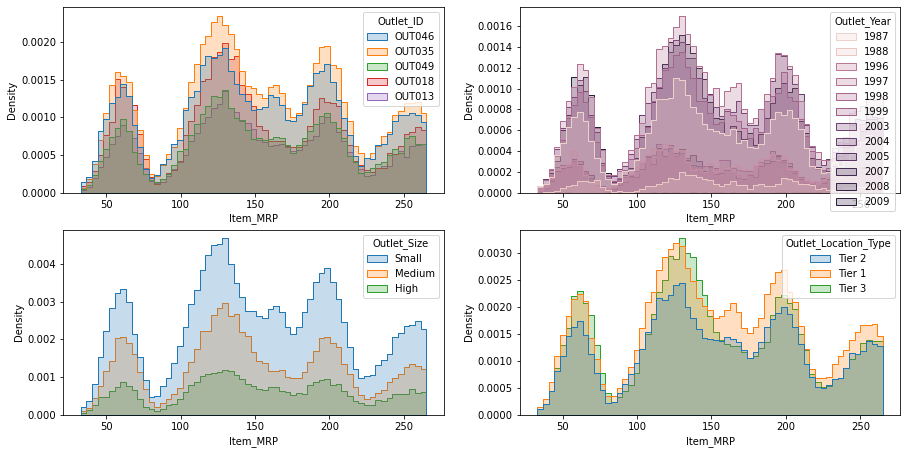


### Item MRP

We have created a histogram with 20 bins for Item MRP. Most frequent occurrence is when the items MRPS is around 60, 125 and 200. This means that maximum item-MRP of products that are being sold fall in this range. Beyond these values, the MRP can vary.

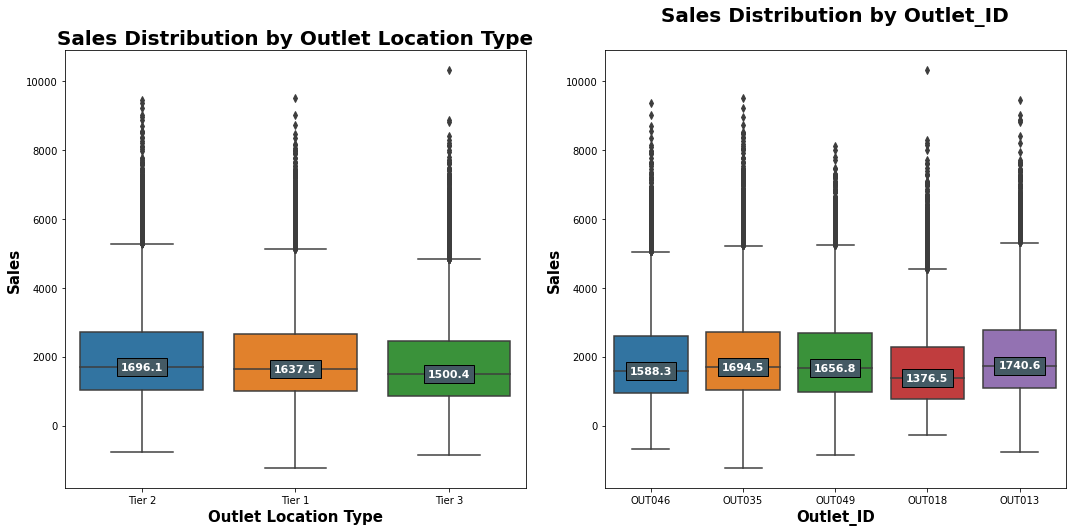
Next, we plot the frequency charts of Item MRP as per Outlet\_ID, Outlet\_Year, Outlet\_Size and Outlet\_Location\_Type.

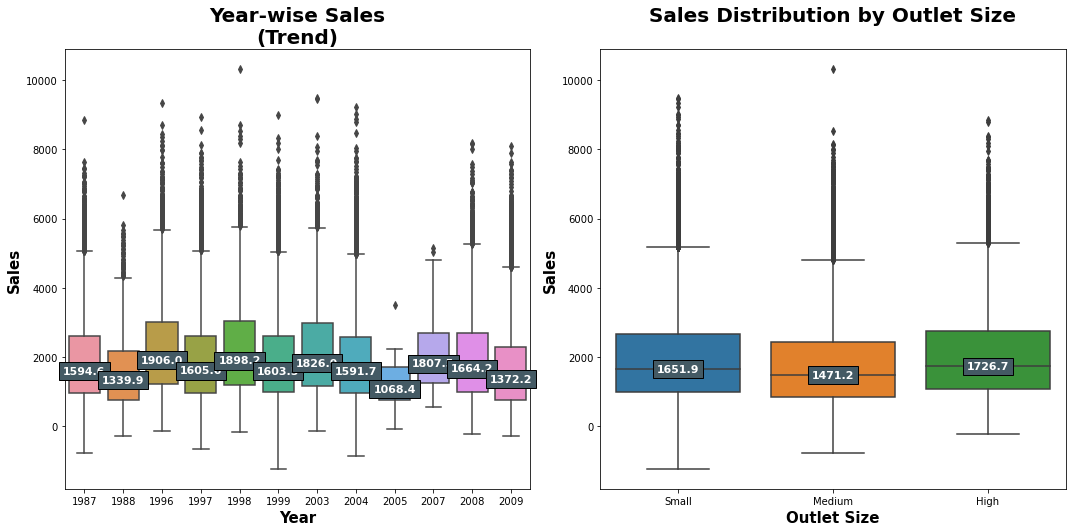




## **Box n Whiskers Plot**

We plot Sales numbers to check s KPI : Year-wise Sales(Trend) and Sales Distribution by Outlet Size**.** The numbers here indicate the median values of Sales every year and by Outlet size. Median sales was highest in the year 1996 and the lowest in 2005. Big outlets have maximum median sales and medium sized outlets have minimum median sales. Tier 2 cities have highest median sales and OUT013 had the highest median sales.

****

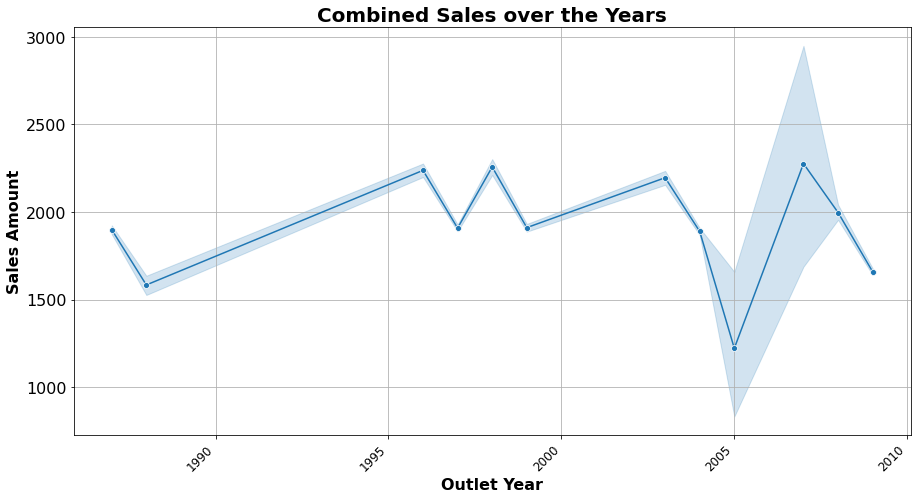


## **Times Series Analysis**

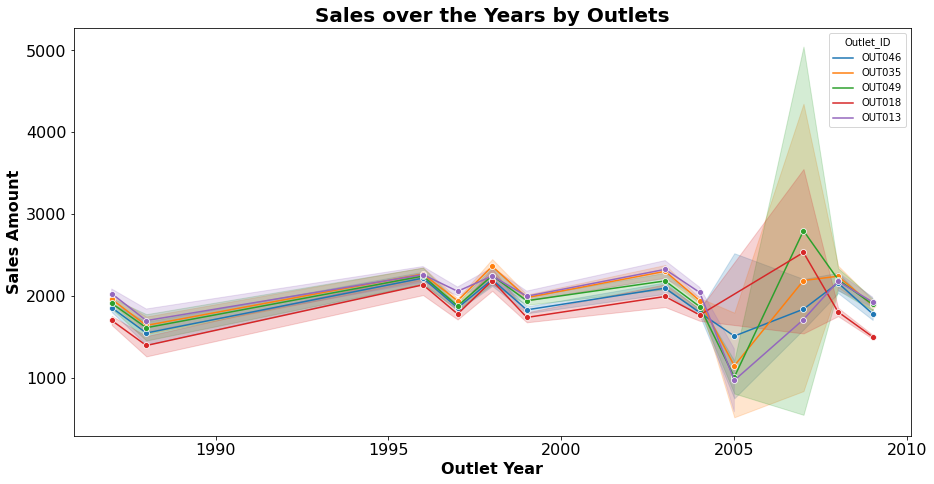
Time series analysis is a specific way of analyzing a sequence of data points collected over an interval of time. In time series analysis, analysts record data points at consistent intervals over a set period of time rather than just recording the data points intermittently or randomly.

Here, we see that sales amount varied around the Rs. 2000 mark all through the years, although in 2005 there was a sharp drop followed by a steep increase in 2006-2007 period.

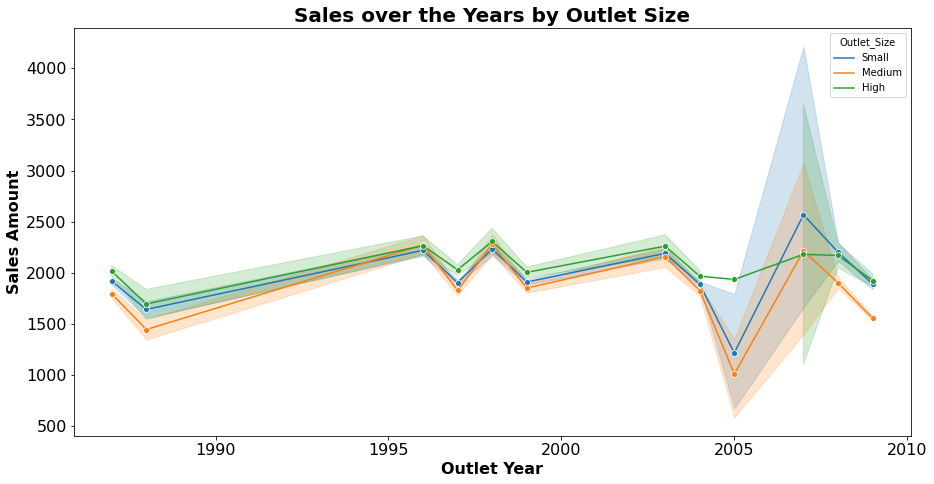
### Combined Sales



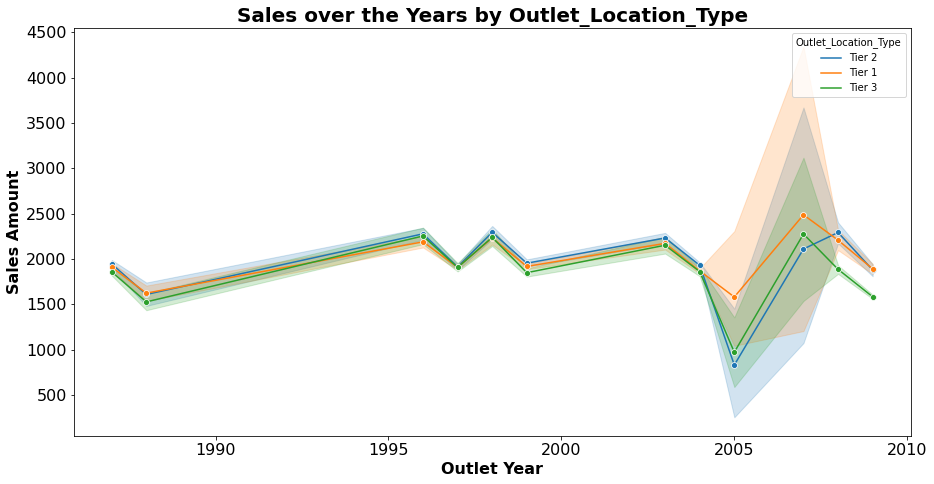
### Sales by Outlet



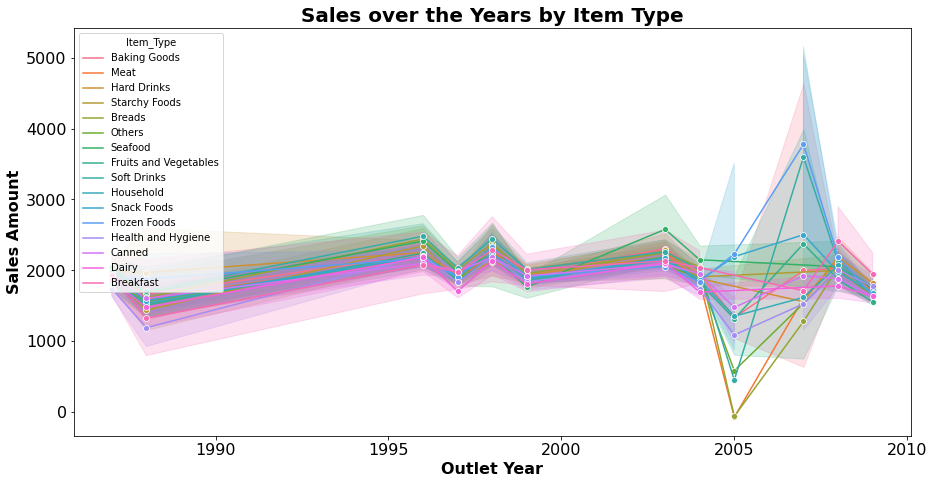
### Sales by Outlet Size



### Sales by Outlet Location Type



### Sales by Item Type



## **Time Series Forecast – Prophet**

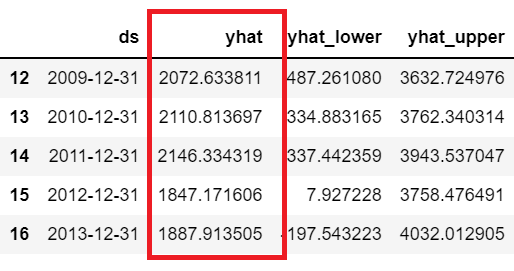
We will try and forecast the combined sales value for the next 5 years based on Facebook’s Prophet algorithm. Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well.

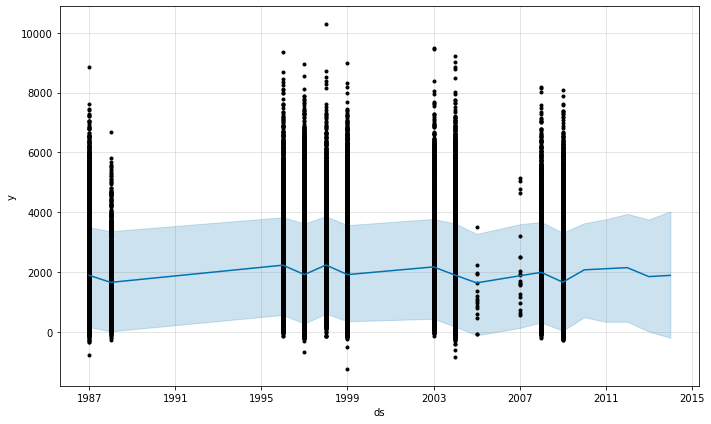
When creating the prophet models, I set the changepoint prior to 0.15, up from the default value of 0.05. This hyperparameter is used to control how sensitive the trend is to changes, with a higher value being more sensitive and a lower value less sensitive. This value is used to combat one of the most fundamental trade-offs in machine learning: bias vs. variance.

Here, under the predictions table, we are only concerned with ds, yhat\_lower, yhat\_upper, and yhat because these are the variables that will give us the predicted results with respect to the date specified.

### 5-year predictions

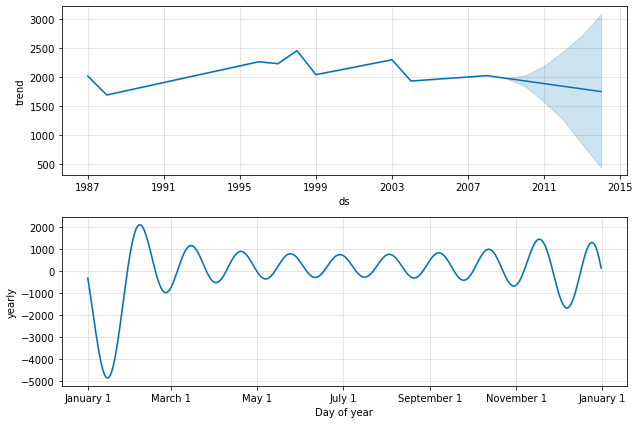
**yhat means the predicted output** based on the input fed to the model, yhat\_lower, and upper means the upper and lower value that can go based on the predicted output that is, the fluctuations that can happen.





### Upcoming Trend

Here, we can see the trends with respect to year and cyclicity in a year. The first graph represents a slightly decreasing trend as we progress through the years and the latter shows a fluctuating trend in the monthly sales. For most months it is steady but towards the end of the year from December to January there is some fluctuation. The fluctuation gains momemtum between January and February.

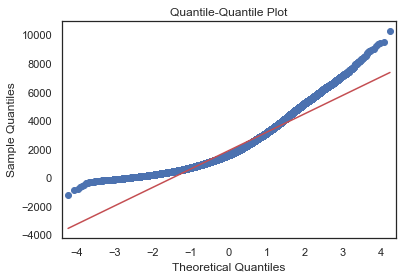


# **Statistical Analysis**

## **Normality Distribution Tests**

The given statistical tests prove that Sales data is not normally distributed. We can try some type of transformations like log, exponential, inversions etc. But those conversions are not helping improve the accuracy in this case.

## **Quantile-Quantile Plot**

The Q-Q plot, or quantile-quantile plot, is a graphical tool to help us assess if a set of data possibly came from some theoretical distribution such as a Normal or exponential. Q-Q plots are used to find the type of distribution for a random variable whether it be a Gaussian Distribution, Uniform Distribution, Exponential Distribution or even Pareto Distribution, etc. We can tell the type of distribution using the power of the Q-Q plot just by looking at the plot.

## **Shapiro-Wilk Test**

The Shapiro-Wilk test evaluates a data sample and quantifies how likely it is that the data was drawn from a Gaussian distribution, named for Samuel Shapiro and Martin Wilk.

**Results:**

H0 : Sample was drawn from a Gaussian distribution , Ha : Sample was not drawn from a Gaussian distribution

Statistics=0.9218, p-value=0.0000

Conclusion:

Sample does not look Gaussian (reject Null Hypothesis H0)

## **Anderson-Darling Test**

Anderson-Darling Test is a statistical test that can be used to evaluate whether a data sample comes from one of among many known data samples, named for Theodore Anderson and Donald Darling.

**Results:**

H0 : Sample was drawn from a Gaussian distribution , Ha : Sample was not drawn from a Gaussian distribution

Statistic: 1819.1218

Significance Level 15.0000: Critical Value 0.5760, Data does not look normal (reject Null Hypothesis H0)

Significance Level 10.0000: Critical Value 0.6560, Data does not look normal (reject Null Hypothesis H0)

Significance Level 5.0000: Critical Value 0.7870, Data does not look normal (reject Null Hypothesis H0)

Significance Level 2.5000: Critical Value 0.9180, Data does not look normal (reject Null Hypothesis H0)

Significance Level 1.0000: Critical Value 1.0920, Data does not look normal (reject Null Hypothesis H0)

## **Spearman Rank Correlation**

Spearman rank correlation coefficient measures the monotonic relation between two variables. Its values range from -1 to +1 and can be interpreted as:

+1: Perfectly monotonically increasing relationship

+0.8: Strong monotonically increasing relationship

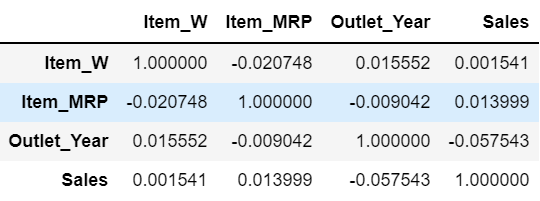
+0.2: Weak monotonically increasing relationship

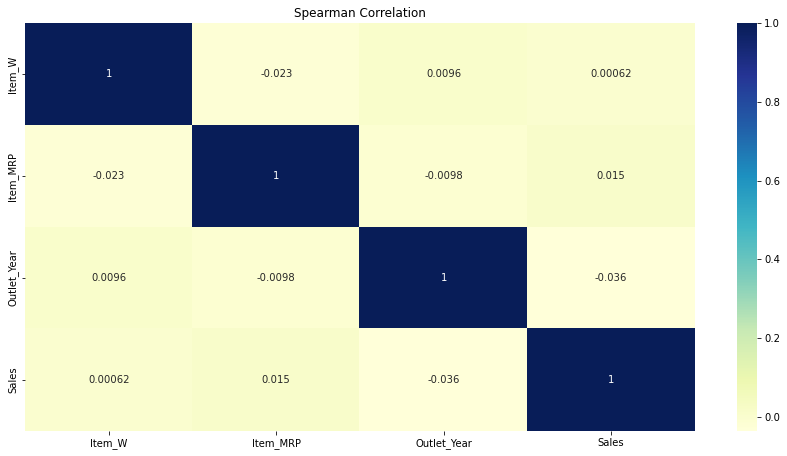
0: Non-monotonic relation

-0.2: Weak monotonically decreasing relationship

-0.8: Strong monotonically decreasing relationship

-1: Perfectly monotonically decreasing relationship





## **Kruskal-Wallis H Test**

The Kruskal-Wallis test is a nonparametric version of the one-way analysis of variance test or ANOVA for short. A Kruskal-Wallis test is used to determine whether or not there is a statistically significant difference between the medians of three or more independent groups. It is considered to be the non-parametric equivalent of the One-Way ANOVA.