**Store-Sales Time Series Forecasting**

**Using Machine Learning for predicting grocery sales**

# **Introduction**

## Dataset Description

In this dataset, we predict sales for the thousands of product families sold at Favorita stores located in Ecuador. The Information included in the dataset is as follows

The training data includes dates, store and product information.

It also holds whether that item was being promoted,

It also holds the sales numbers.

Additional files include supplementary information that may be useful in building your models.

## File Descriptions and Data Field Information

### **train.csv**

1. The training data, comprising time series of features **store\_nbr**, **family**, and **onpromotion** as well as the target **sales**.
2. **store\_nbr** identifies the store at which the products are sold.
3. **family** identifies the type of product sold.
4. **sales** gives the total sales for a product family at a particular store at a given date. Fractional values are possible since products can be sold in fractional units (1.5 kg of cheese, for instance, as opposed to 1 bag of chips).
5. **onpromotion** gives the total number of items in a product family that were being promoted at a store at a given date.

### **test.csv**

1. The test data, having the same features as the training data. You will predict the target **sales** for the dates in this file.
2. The dates in the test data are for the 15 days after the last date in the training data.

### **sample\_submission.csv**

1. A sample submission file in the correct format.

### **stores.csv**

* Store metadata, including **city**, **state**, **type**, and **cluster**.
* **cluster** is a grouping of similar stores.

### **oil.csv**

1. Daily oil price. Includes values during both the train and test data timeframes. (Ecuador is an oil-dependent country and it's economical health is highly vulnerable to shocks in oil prices.)

### **holidays\_events.csv**

1. Holidays and Events, with metadata
2. NOTE: Pay special attention to the **transferred** column. A holiday that is transferred officially falls on that calendar day, but was moved to another date by the government. A transferred day is more like a normal day than a holiday. To find the day that it was actually celebrated, look for the corresponding row where type is Transfer. For example, the holiday Independencia de Guayaquil was transferred from 2012-10-09 to 2012-10-12, which means it was celebrated on 2012-10-12. Days that are type Bridge are extra days that are added to a holiday (e.g., to extend the break across a long weekend). These are frequently made up by the type Work Day which is a day not normally scheduled for work (e.g., Saturday) that is meant to payback the Bridge.
3. Additional holidays are days added a regular calendar holiday, for example, as typically happens around Christmas (making Christmas Eve a holiday).

## **Additional Notes**

* Wages in the public sector are paid every two weeks on the 15 th and on the last day of the month. Supermarket sales could be affected by this.
* A magnitude 7.8 earthquake struck Ecuador on April 16, 2016. People rallied in relief efforts donating water and other first need products which greatly affected supermarket sales for several weeks after the earthquake.

# **Introduction**

The goal of the project was to predict the survival of passengers based on a set of data.

We used Kaggle DataSet

"Store Sales Time-Series Fore Casting Data Set"

**( https://www.kaggle.com/competitions/store-sales-time-series-forecasting/data)**

to retrieve necessary data and evaluate accuracy of our predictions.

The historical data has been split into five groups,

1. training set
2. test set
3. oil set
4. holiday set
5. store set

score after the training dataset was the percentage of correctly sales predictions.

**Goals Achieved during this project**

* Programming language Python and its libraries NumPy (to perform matrix operations) and SciKit-Learn (to apply machine learning algorithms)
* Machine learning algorithm (Linear Regression)
* Feature Engineering techniques **Tools used**
* Google Colab
* Google Drive for DataSet
* *Python* with the libraries *numpy*, *sklearn*, and *matplotlib*

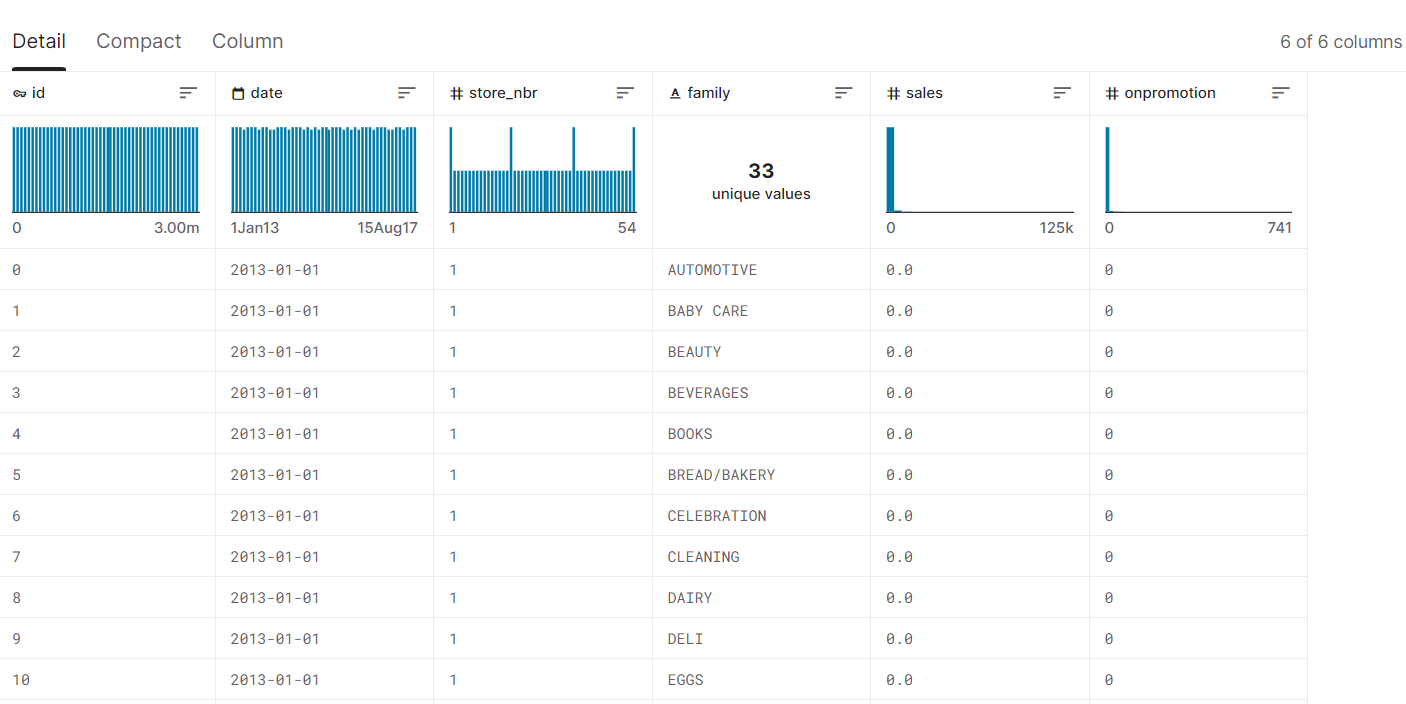
# **Dataset Details**

**Training and Test data come in CSV file and contain the following features:**

* ID
* Date
* Store\_Nbr
* family
* sales
* onpromotion

**The Dataset is a labelled dataset so we use supervised learning techniques..**

**Sample Data in csv Format**



# Preprocessing (Feature Engineering) Details

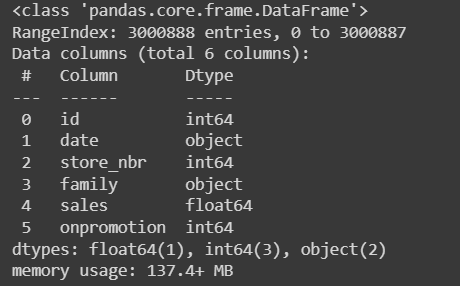
**Data Reterived using**

**Train.head()**

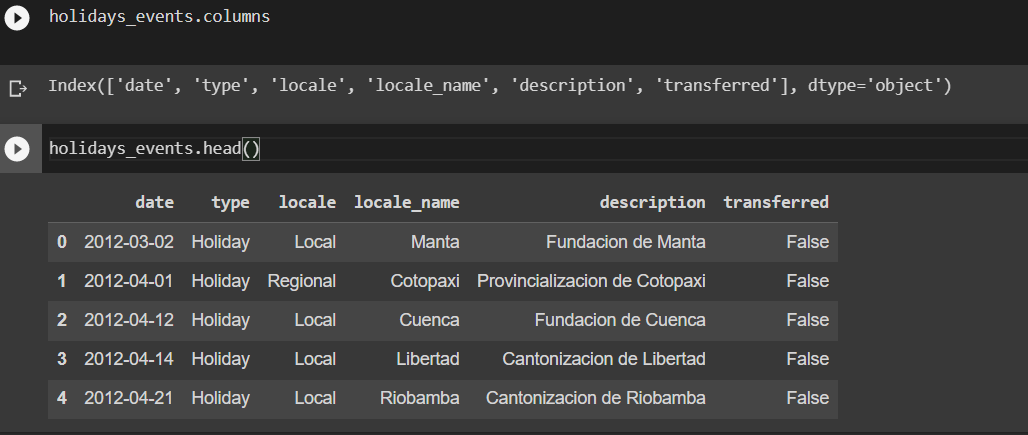


Features of DataSet

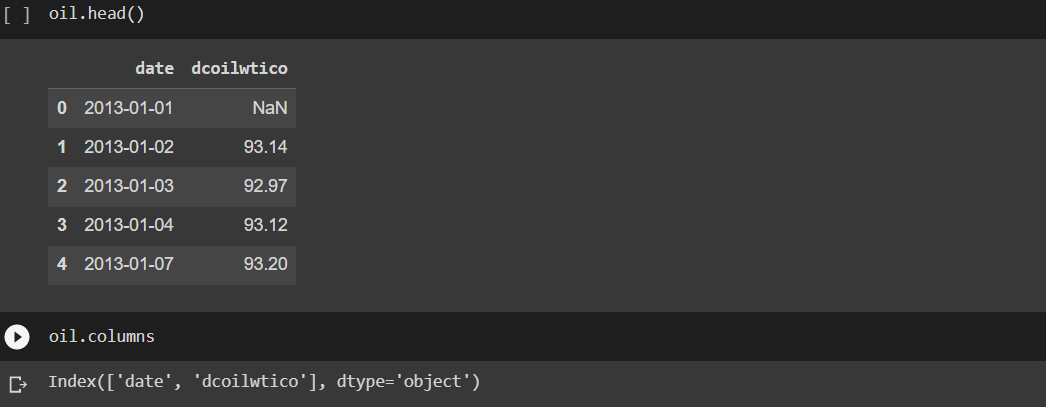
Train.info()



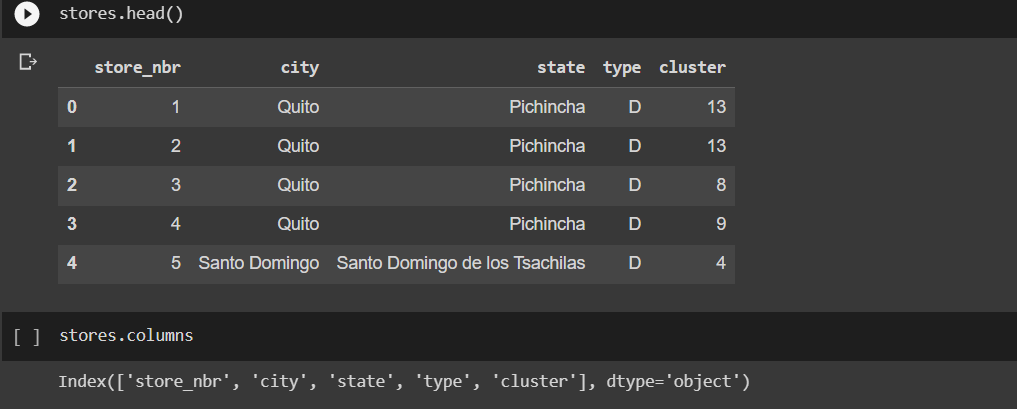
**Details of Holiday Dataset**



**Details of Oil Data**



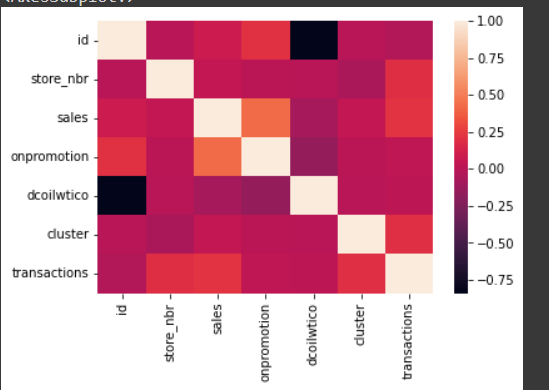
**Details of store data**



Details of Transaction data

# 

Heatmap for Train Dataset (Correlation)



# Linear Regression Model

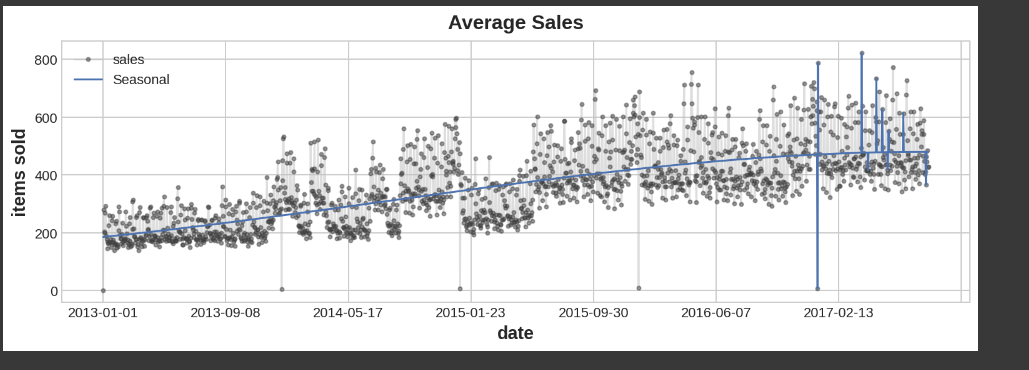
# Model Generation on Average Sales Amount V/S Date

# 

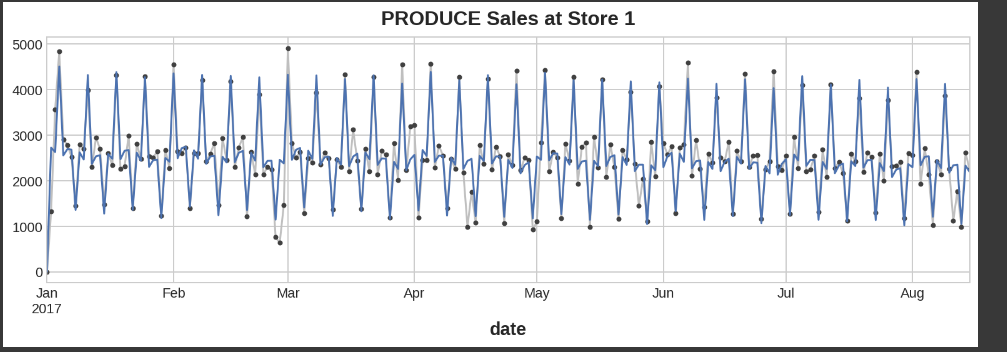
# Model Generation on Average Sales Amount V/S Date using Trend and Forecast

# 

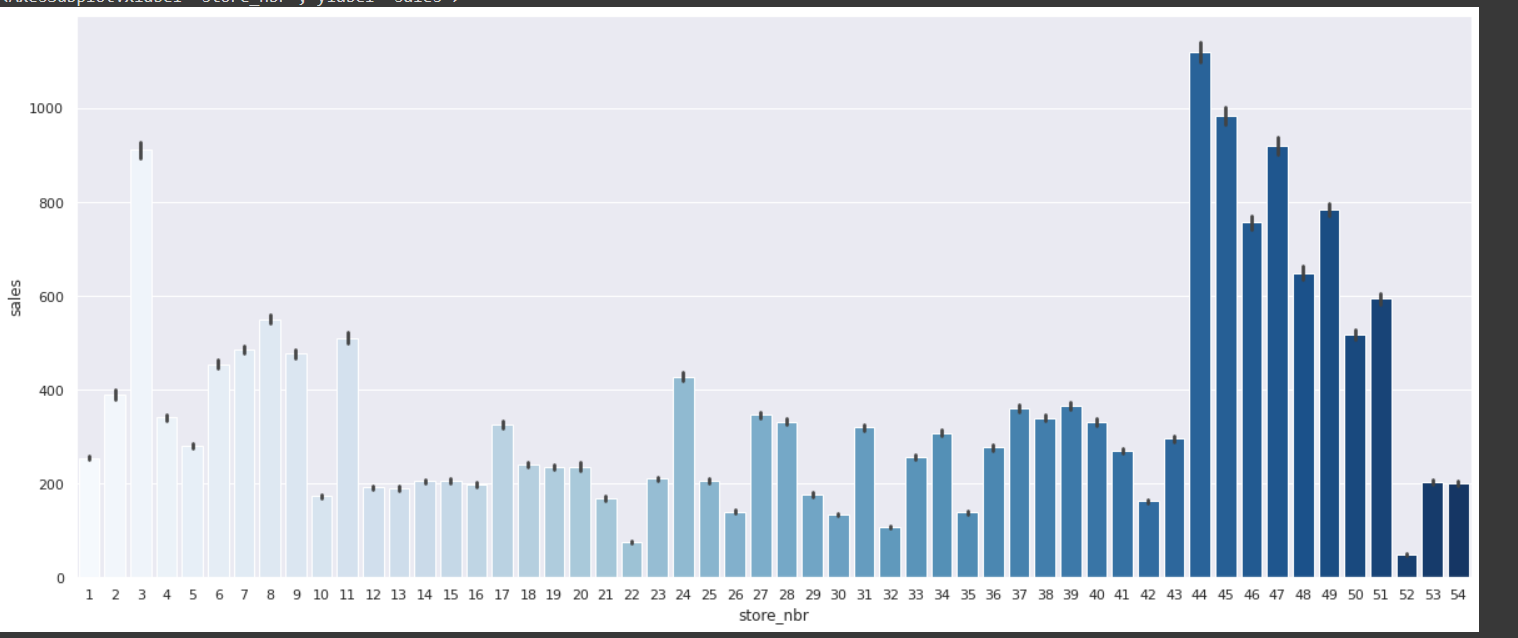
Model Generation on Average Sales Amount V/S Holiday dataset with sales and Seasonal



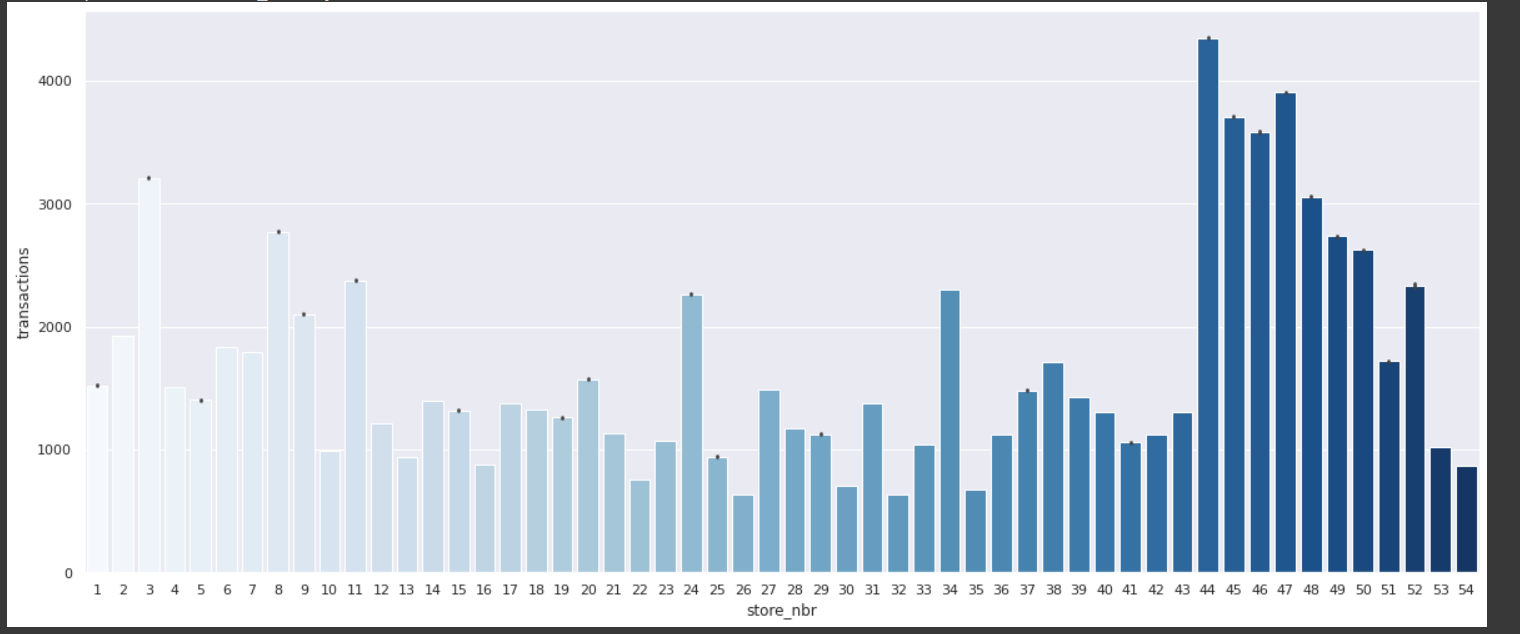
Sales Details at a particular Store with stores.csv



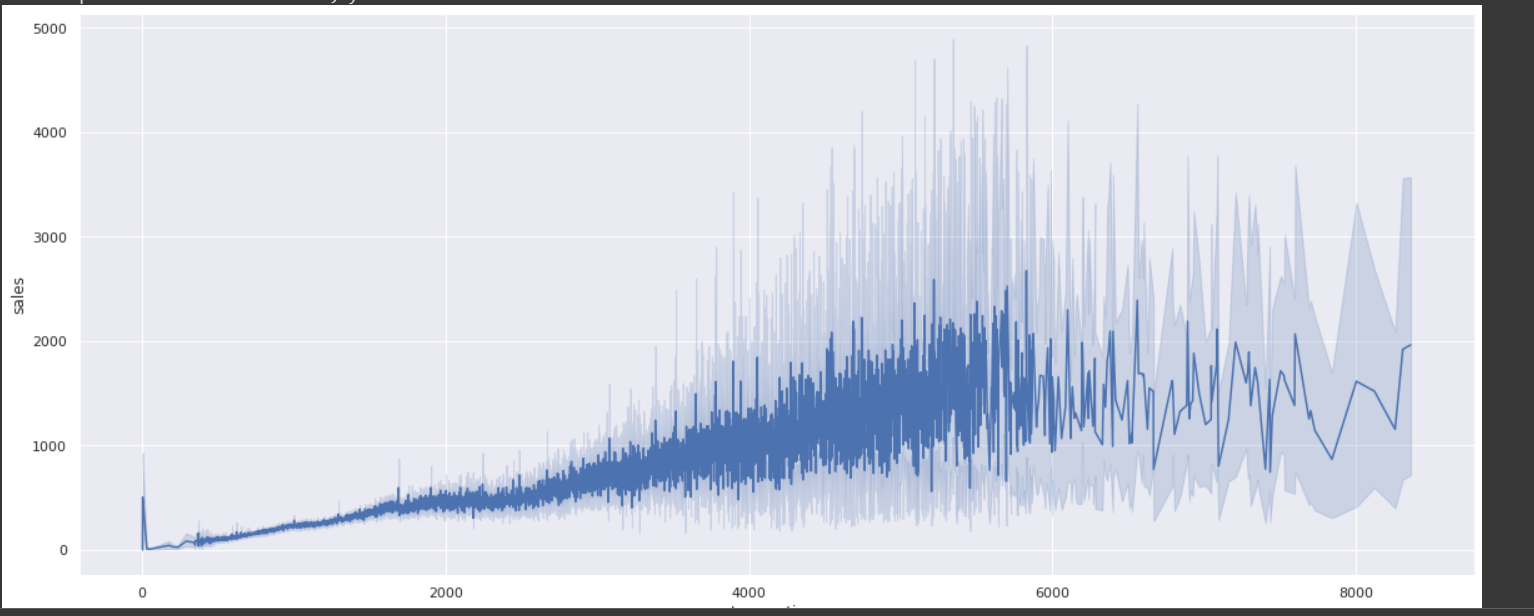
Store wise sales Chart



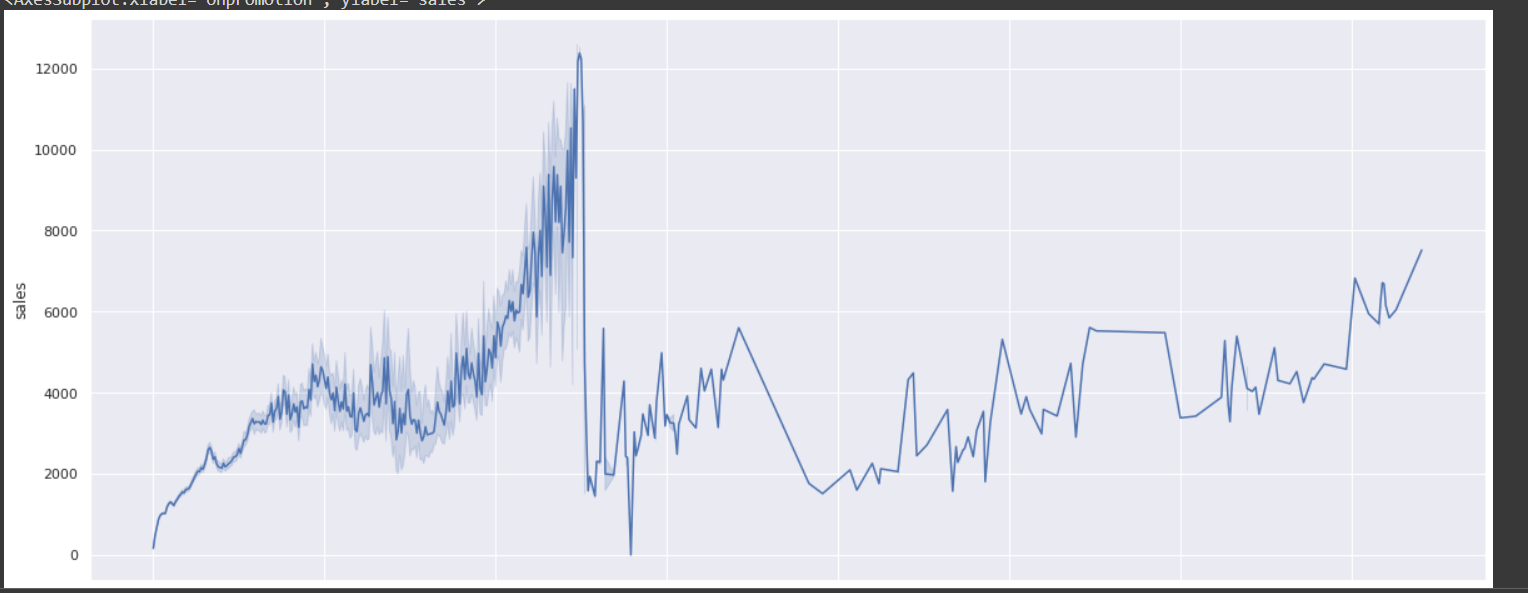
Store wise Transaction Chart



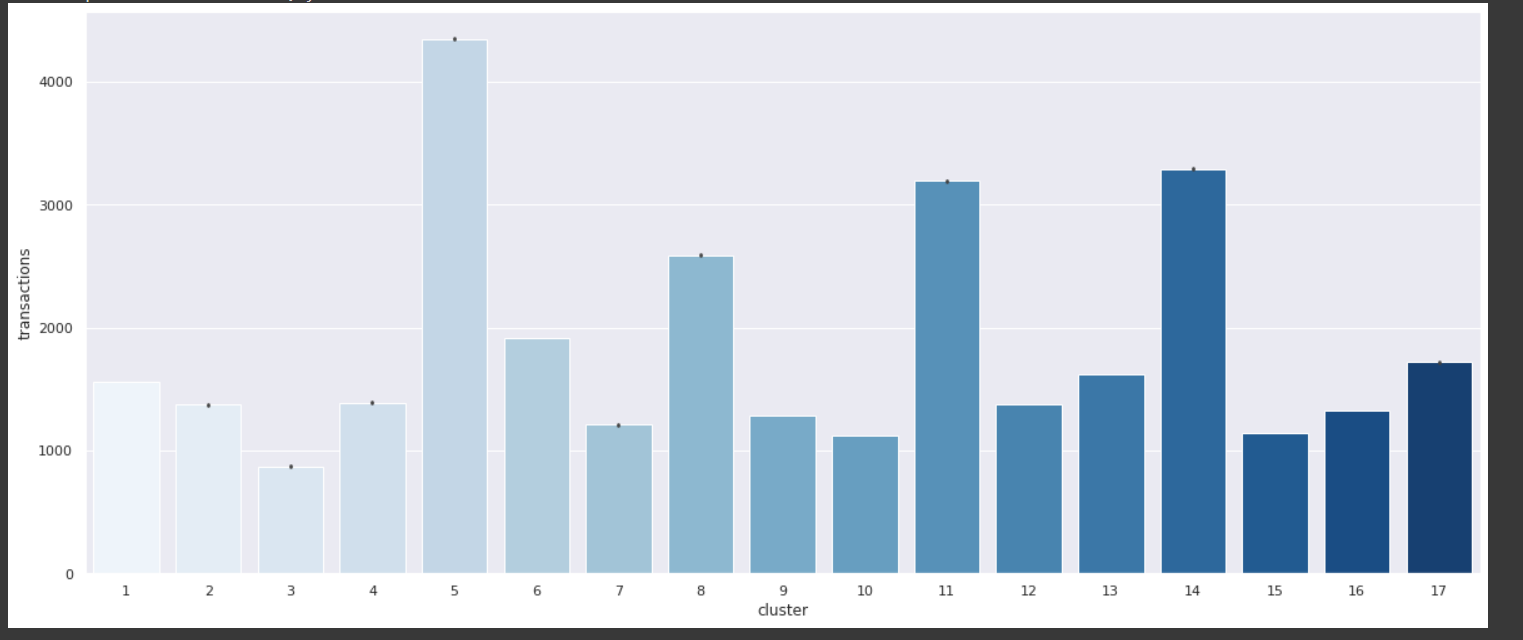
Transaction wise Sales Chart



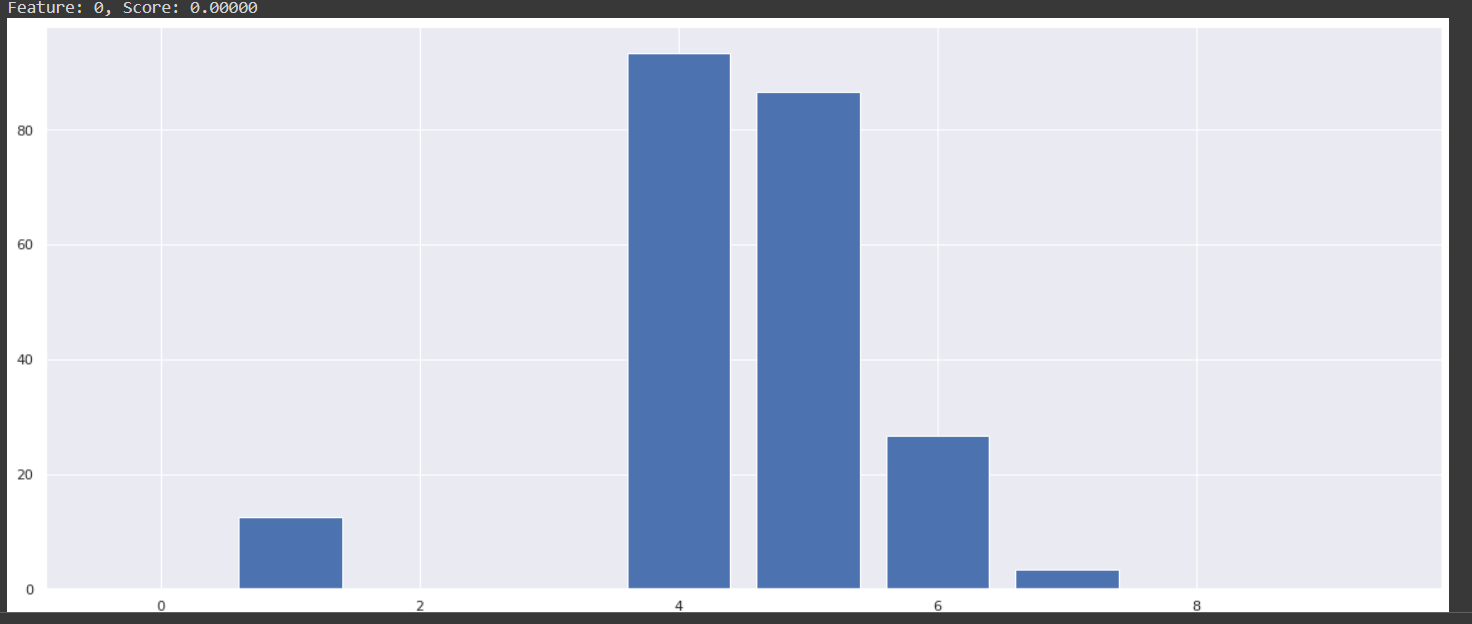
On Promotion Sales Analysis

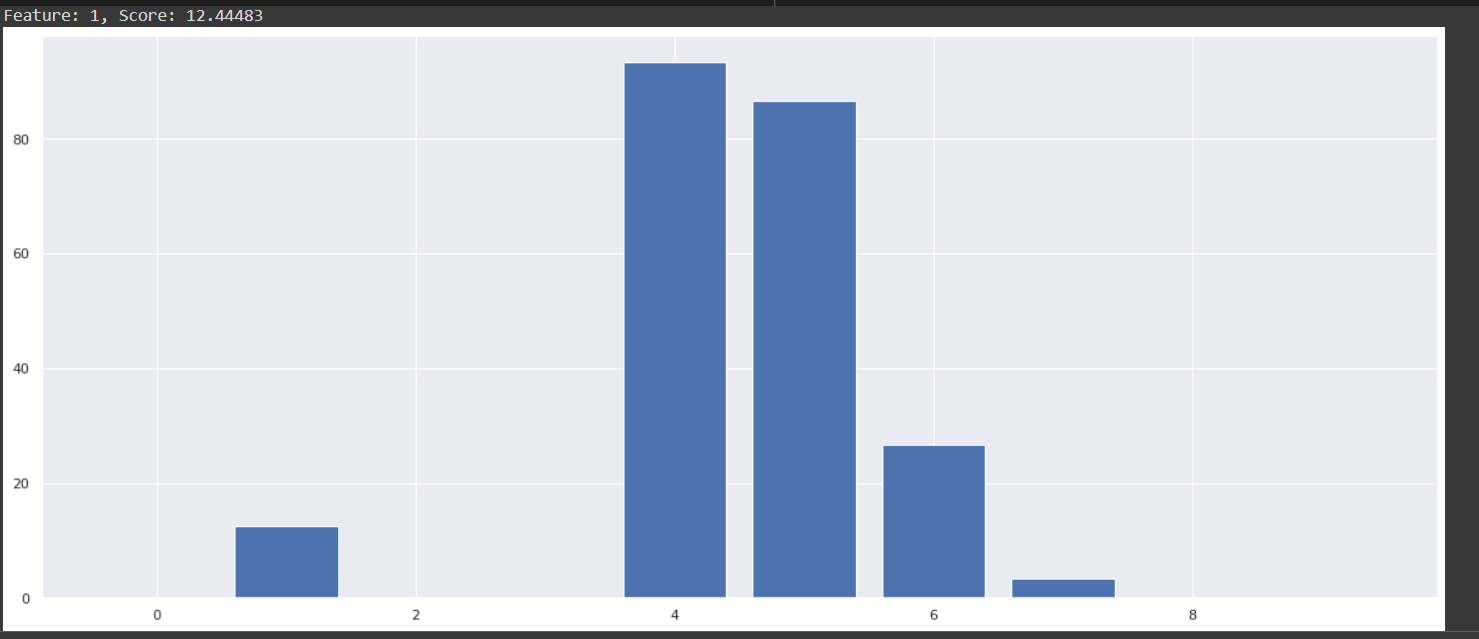


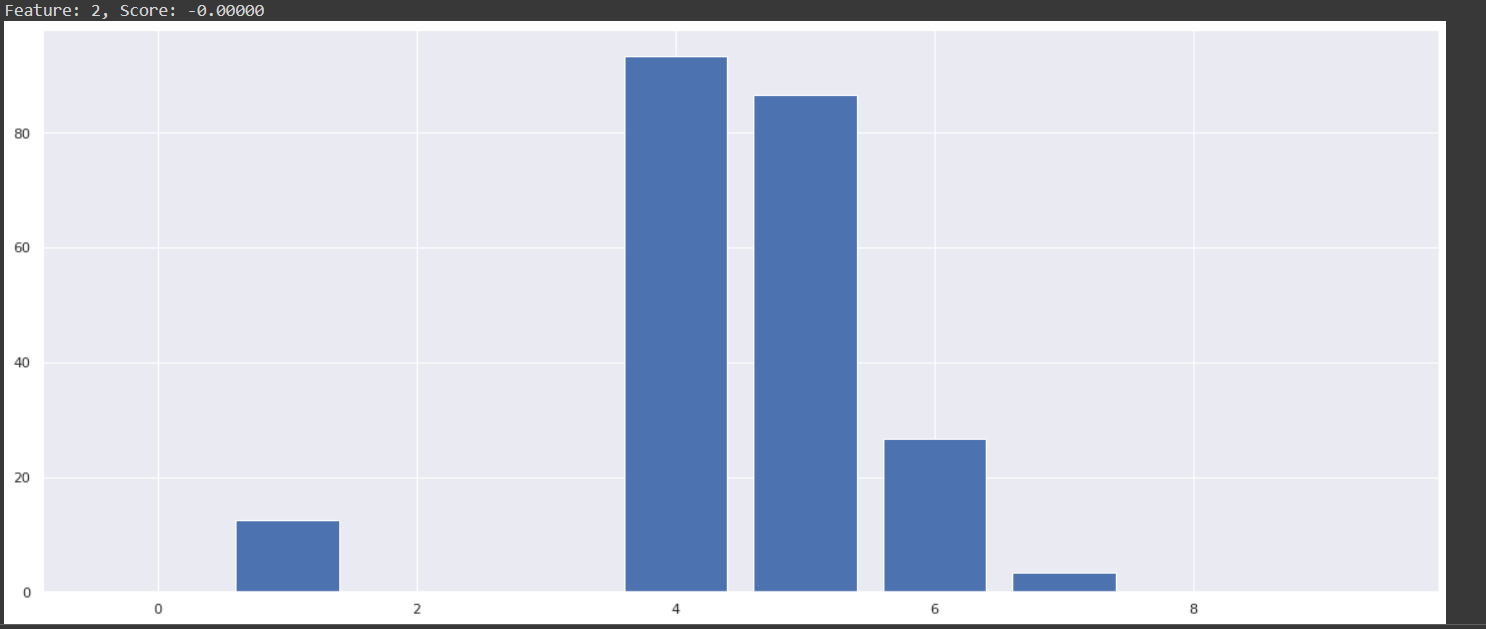
Cluster Wise Sales Transaction

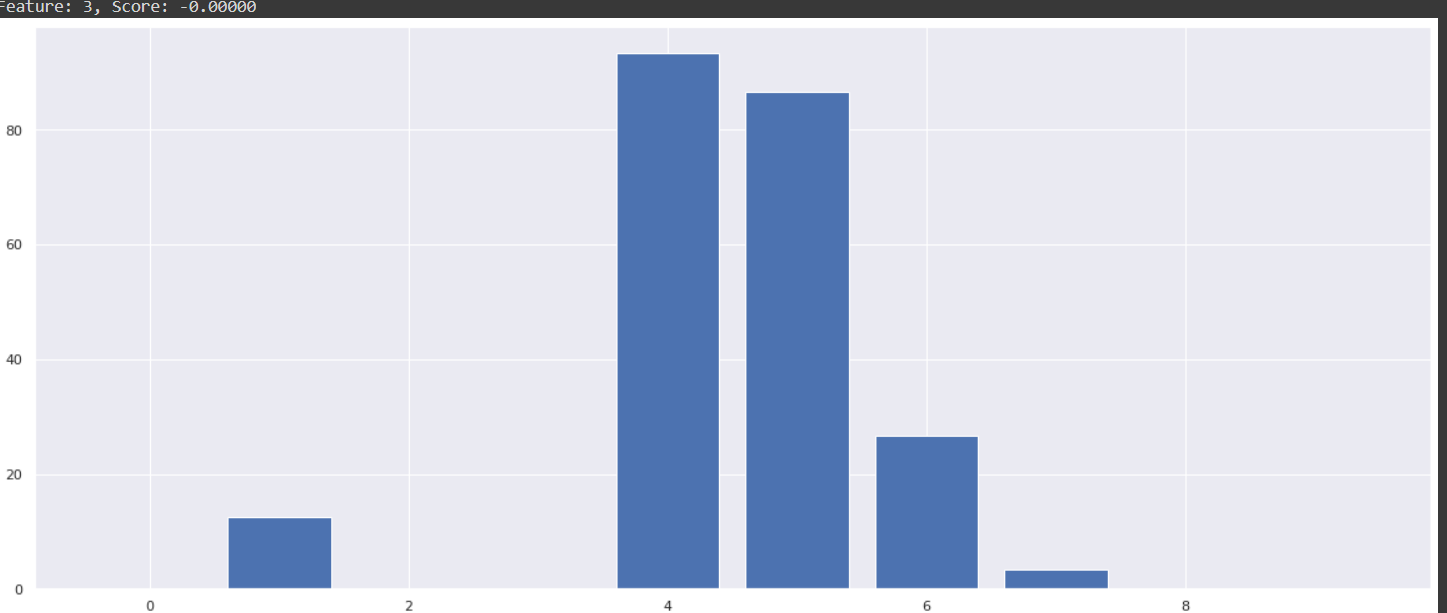


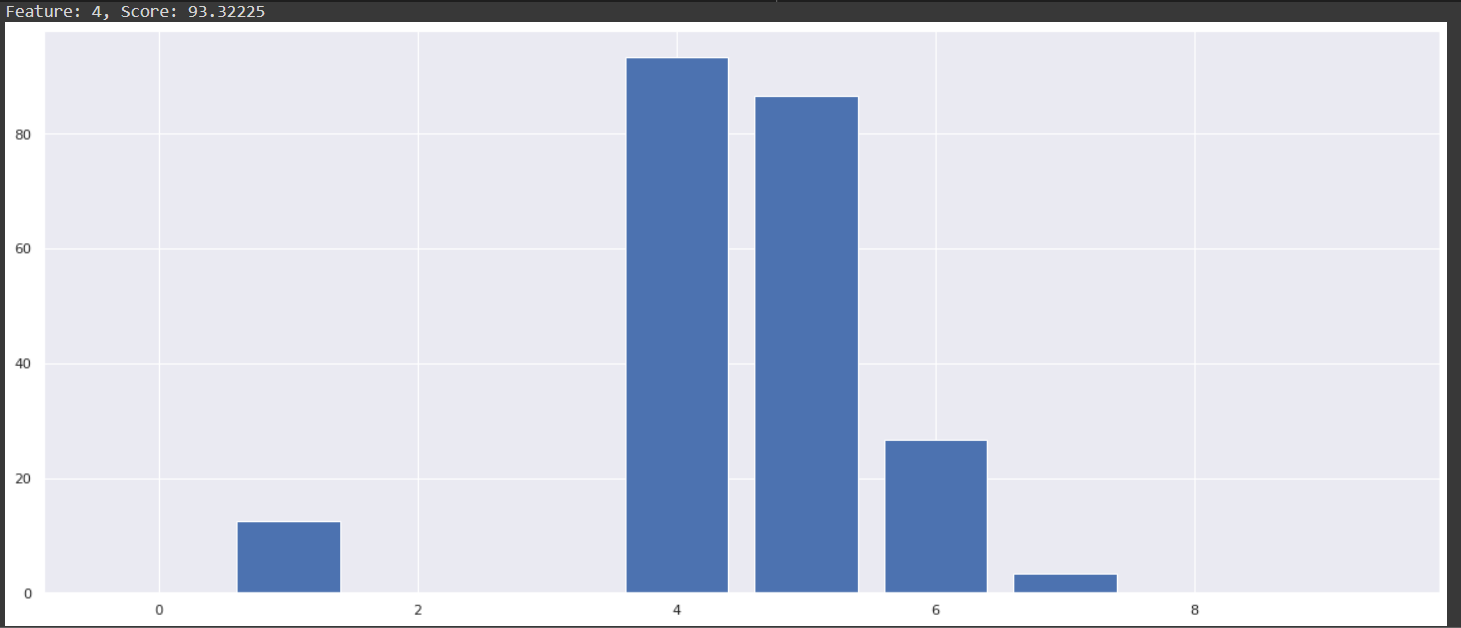
Linear Regression Feature Wise Score

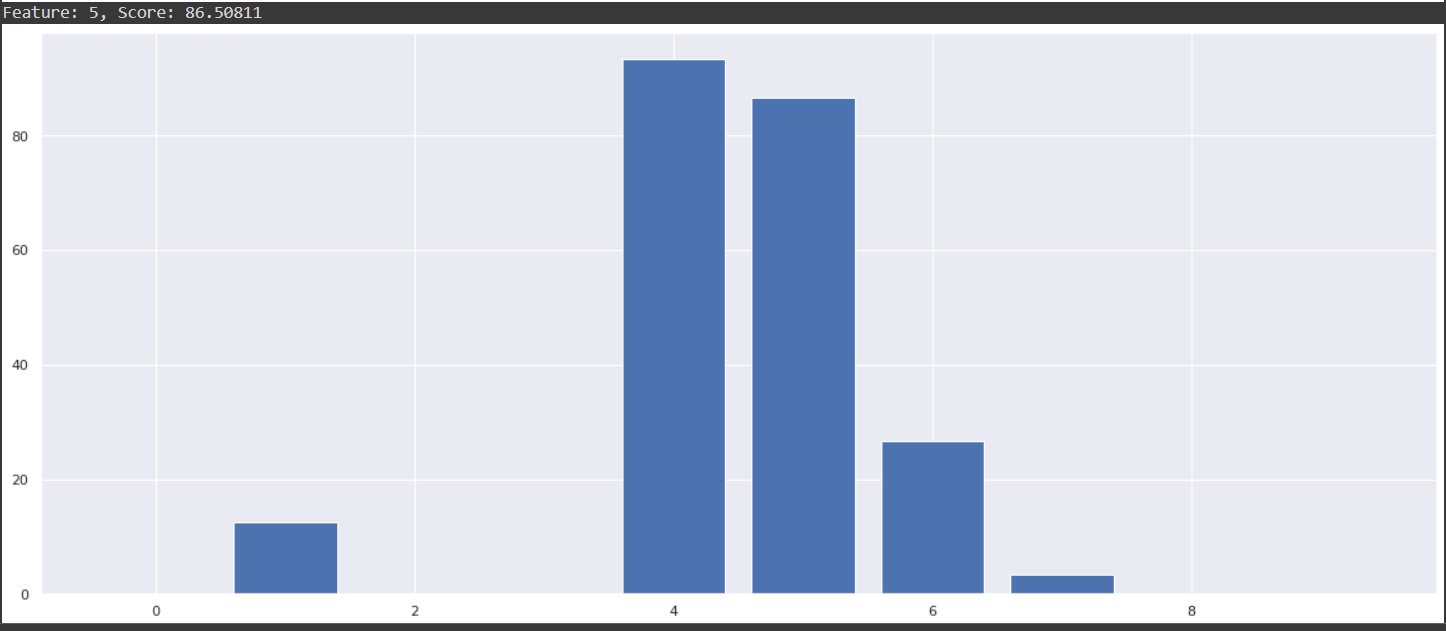










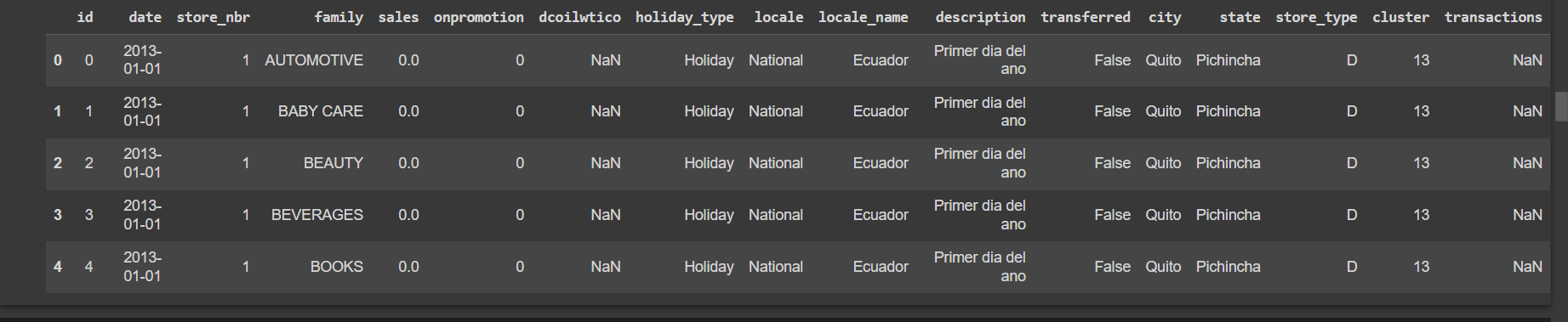


**Transformation into a categorical column.**

We have already noticed from the table, there is a column that contain string-type values: The “Holiday Description”

Let’s convert that into integer type values, and transform it into a categorical column: using OneHotEncoding

**Merging data using joins from different files**

****

**Basic Exploratory on Dataset**

**Data preprocessing for model**

making our data, model-ready. The objectives we have to fulfill are listed below:

1. Drop the null values from the Embarked column
2. Include only relevant data
3. Categorically transform all of the data, using something called a transformer.
4. Impute data with the central tendencies for age and fare.
5. Normalize the *fare*column to have a more normal distribution.
6. using standard scaler scale data 0-1

**Model Description**

## **What is Regression Analysis?**

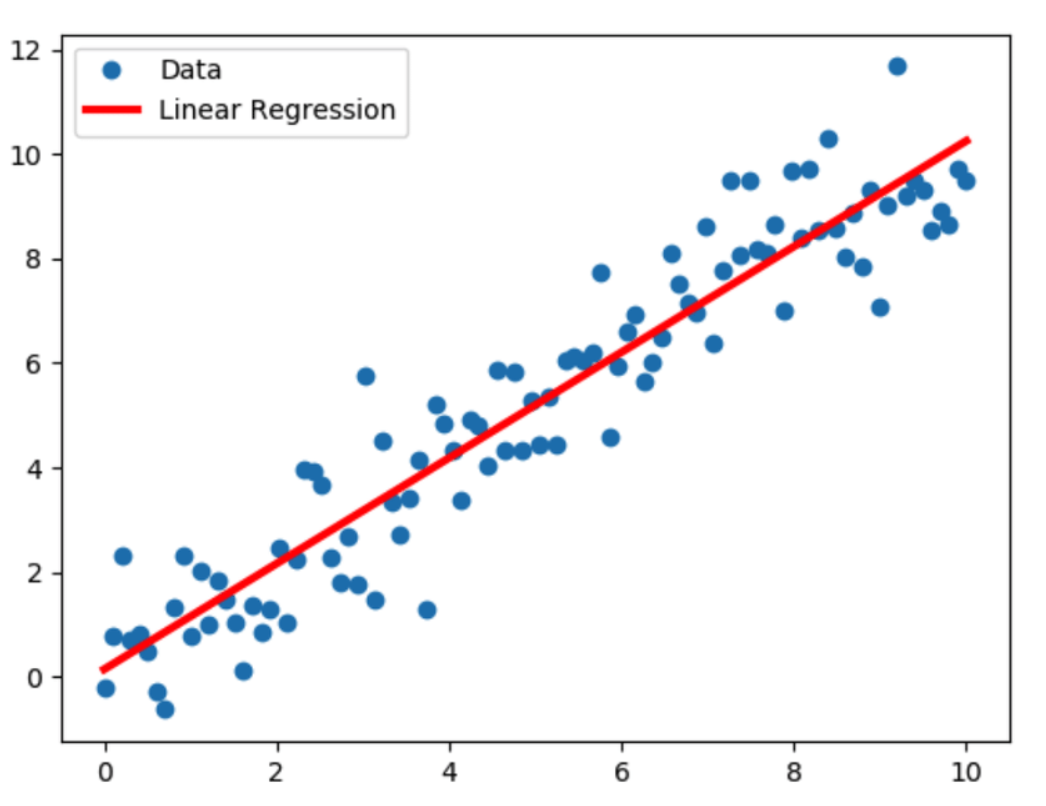
Predictive modelling techniques such as regression analysis may be used to determine the relationship between a dataset’s dependent (goal) and independent variables. It is widely used when the dependent and independent variables are linked in a linear or non-linear fashion, and the target variable has a set of continuous values. Thus, regression analysis approaches help establish causal relationships between variables, modelling time series, and forecasting. Regression analysis, for example, is the best way to examine the relationship between sales and advertising expenditures for a corporation.

Types of Regression Models Analysis / Different Regression Models  
1. Linear Regression  
2. Logistic Regression  
3. Polynomial Regression  
4. Ridge Regression  
5. Lasso Regression  
6. Quantile Regression

## **Linear Regression**

The most extensively used modelling technique is linear regression, which assumes a linear connection between a dependent variable (Y) and an independent variable (X). It employs a regression line, also known as a best-fit line. The linear connection is defined as Y = c+m\*X + e, where ‘c’ denotes the intercept, ‘m’ denotes the slope of the line, and ‘e’ is the error term.

The linear regression model can be simple (with only one dependent and one independent variable) or complex (with numerous dependent and independent variables) (with one dependent variable and more than one independent variable).



Splitting Dataset for Training and Testing

## **Training Data**

The observations in the training set form the experience that the algorithm uses to learn. In supervised learning problems, each observation consists of an observed output variable and one or more observed input variables.

## **Test Data**

The test set is a set of observations used to evaluate the performance of the model using some performance metric. It is important that no observations from the training set are included in the test set. If the test set does contain examples from the training set, it will be difficult to assess whether the algorithm has learned to generalize from the training set or has simply memorized it.

## **Underfitting and Overfitting**

Splitting a dataset might also be important for detecting if your model suffers from one of two very common problems, called [underfitting and overfitting](https://en.wikipedia.org/wiki/Overfitting):

1. **Underfitting** is usually the consequence of a model being unable to encapsulate the relations among data. For example, this can happen when trying to represent nonlinear relations with a linear model. Underfitted models will likely have poor performance with both training and test sets.
2. **Overfitting** usually takes place when a model has an excessively complex structure and learns both the existing relations among data and noise. Such models often have bad generalization capabilities. Although they work well with training data, they usually yield poor performance with unseen (test) data

## **Model Evaluation**

**Linear Regression Results**

After Regression Sales details score for data

