**BigMart Sales Prediction**

**Content:**

**Topics Page No.**

**Problem Statement 2**

**Variable Description 4**

**Univariate Analysis 4**

**Analysis and assumptions 5**

**Data Cleaning 6**

**Visualization of Data 6**

**Model Building 11**

**Model Evaluation 12**

**Forecasting 13**

**Full Source Code 14**

**Problem Statement:**

The data scientists at BigMart have collected 2013 sales data for 1559 products across 10 stores in different cities. Also, certain attributes of each product and store have been defined. The aim is to build a predictive model and find out the sales of each product at a particular store.

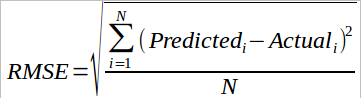
Using this model, BigMart will try to understand the properties of products and stores which play a key role in increasing sales.

Please note that the data may have missing values as some stores might not report all the data due to technical glitches. Hence, it will be required to treat them accordingly.

**Evaluation Metric:**

Your model performance will be evaluated on the basis of your prediction of the sales for the test data (test.csv), which contains similar data-points as train except for the sales to be predicted.

We at our end, have the actual sales for the test dataset, against which your predictions will be evaluated. We will use the Root Mean Square Error value to judge your response.



Where,

N: total number of observations

Predicted: the response entered by user

Actual: actual values of sales

Also, note that the test data is further divided into Public (25%) and Private (75%) data. Your initial responses will be checked and scored on the Public data. But, the final rankings will be based on score on Private data set. Since this is a practice problem, we will keep declare winners after specific time intervals and refresh the competition.

**Data Set**

We have train (8523) and test (5681) data set, train data set has both input and output variable(s). You need to predict the sales for test data set.

**Variable Description**

**Item\_Identifier** Unique product ID

**Item\_Weight** Weight of product

**Item\_Fat\_Content** Whether the product is low fat or not

**Item\_Visibility** The % of total display area of all products in a store allocated to the particular product

**Item\_Type** The category to which the product belongs

**Item\_MRP** Maximum Retail Price (list price) of the product

**Outlet\_Identifier** Unique store ID

**Outlet\_Establishment\_Year** The year in which store was established

**Outlet\_Size** The size of the store in terms of ground area covered

**Outlet\_Location\_Type** The type of city in which the store is located

**Outlet\_Type** Whether the outlet is just a grocery store or some sort of supermarket

**Item\_Outlet\_Sales** Sales of the product in the particular store. This is the outcome variable to be predicted.

*Importing the Data set*

*Codes*

*Train\_data<-read.csv("Train\_UWu5bXk.csv", header = T, na.strings = "")*

*View(Train\_data)*

**Univariate Analysis of the Data**

*Codes*

*library(psych)*

*describe(Train\_data)*

vars n mean sd median trimmed mad min max

Item\_Identifier\* 1 8523 780.71 449.22 784.00 781.25 572.28 1.00 1559.00

Item\_Weight 2 7060 12.86 4.64 12.60 12.80 6.08 4.55 21.35

Item\_Fat\_Content\* 3 8523 3.60 1.08 3.00 3.61 0.00 1.00 5.00

Item\_Visibility 4 8523 0.07 0.05 0.05 0.06 0.05 0.00 0.33

Item\_Type\* 5 8523 8.23 4.21 7.00 8.27 4.45 1.00 16.00

Item\_MRP 6 8523 140.99 62.28 143.01 139.70 68.26 31.29 266.89

Outlet\_Identifier\* 7 8523 5.72 2.84 6.00 5.73 4.45 1.00 10.00

Outlet\_Establishment\_Year 8 8523 1997.83 8.37 1999.00 1998.04 7.41 1985.00 2009.00

Outlet\_Size\* 9 6113 2.24 0.70 2.00 2.30 1.48 1.00 3.00

Outlet\_Location\_Type\* 10 8523 2.11 0.81 2.00 2.14 1.48 1.00 3.00

Outlet\_Type\* 11 8523 2.20 0.80 2.00 2.13 0.00 1.00 4.00

Item\_Outlet\_Sales 12 8523 2181.29 1706.50 1794.33 1971.33 1604.06 33.29 13086.96

range skew kurtosis se

Item\_Identifier\* 1558.00 -0.01 -1.20 4.87

Item\_Weight 16.80 0.08 -1.23 0.06

Item\_Fat\_Content\* 4.00 0.06 -0.68 0.01

Item\_Visibility 0.33 1.17 1.68 0.00

Item\_Type\* 15.00 0.10 -0.97 0.05

Item\_MRP 235.60 0.13 -0.89 0.67

Outlet\_Identifier\* 9.00 -0.06 -1.26 0.03

Outlet\_Establishment\_Year 24.00 -0.40 -1.21 0.09

Outlet\_Size\* 2.00 -0.36 -0.92 0.01

Outlet\_Location\_Type\* 2.00 -0.21 -1.46 0.01

Outlet\_Type\* 3.00 0.93 0.62 0.01

Item\_Outlet\_Sales 13053.67 1.18 1.61 18.48

**Analysis and assumptions**

Here we can see that there are 7 categorical variables and 5 continuous variables.

Categorical variables like Item\_Identifier, Outlet\_Identifier, Item\_Type have lots of categories and are unnecessary to built the model. So while building the model, these variables will not be considered.

So we can see that the variables Item\_Weight and Outlet\_Size has missing values.

Percentage of values missing in the variables Item\_Weight is 17.16% and Outlet\_Size is 28.27%

We would replace the missing values by mean and mode of the observations for numeric and categorical variables respectively.

*Finding the no. of missing values in each variable*

*Codes*

*colSums(is.na(Train\_data))*

Output

Item\_Identifier Item\_Weight Item\_Fat\_Content

0 1463 0

Item\_Visibility Item\_Type Item\_MRP

0 0 0

Outlet\_Identifier Outlet\_Establishment\_Year Outlet\_Size

0 0 2410

Outlet\_Location\_Type Outlet\_Type Item\_Outlet\_Sales

0 0 0

From the output we can see that the variables Item\_Weight and Outlet\_Size has missing values. We will treat the missing values of Item\_Weight by its mean ,since it is a continuous variable and we will treat the missing values of Outlet\_Size by mode, since it is categorical.

**DATA Cleaning**

*Replacing the missing values by mean and mode for Item\_Weight and Outlet\_Size respect*

*Codes*

*Train\_data$Item\_Weight[which(is.na(Train\_data$Item\_Weight))]<- mean(Train\_data$Item\_Weight,na.rm = T)*

*Train\_data$Outlet\_Size[is.na(Train\_data$Outlet\_Size)] <- "Medium"*

*summary(Train\_data)*

*Correction of the variable Item\_Fat\_Content:* In this variable the observations LF, low fat, Low Fat has same meaning and the obs reg and regular has same meaning. Replacing all the vobs types with only 'Low Fat' and 'Regular'

*Codes*

*library(plyr)*

*Train\_data$Item\_Fat\_Content <- revalue(Train\_data$Item\_Fat\_Content, c("LF" = "Low Fat"))*

*Train\_data$Item\_Fat\_Content <- revalue(Train\_data$Item\_Fat\_Content, c("low fat" = "Low Fat"))*

*Train\_data$Item\_Fat\_Content <- revalue(Train\_data$Item\_Fat\_Content, c("reg" = "Regular"))*

*summary(Train\_data)*

**Visualization of the data**

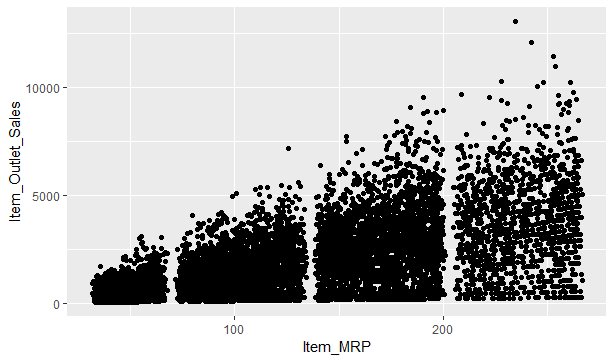
Here we are showing the scatter plots of Item\_MRP, Item\_Visibility, Item weight vs Item\_Outlet\_Sales individually

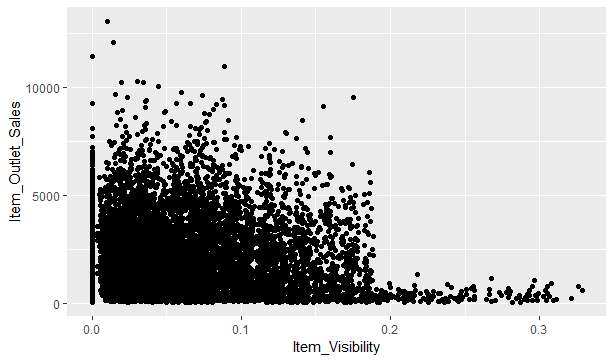
*Codes*

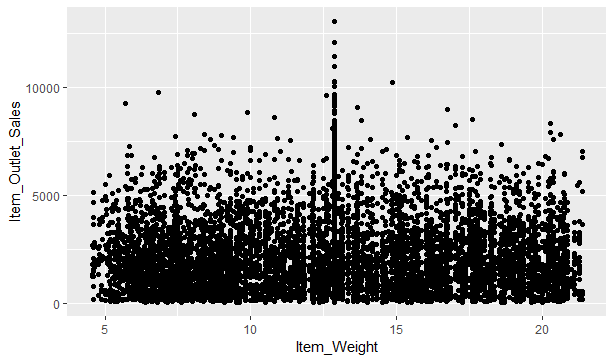
*ggplot(data = Train\_data,aes(x = Item\_MRP,y = Item\_Outlet\_Sales )) + geom\_point()*

*ggplot(data = Train\_data,aes(x = Item\_Visibility,y = Item\_Outlet\_Sales )) + geom\_point()*

*ggplot(data = Train\_data,aes(x = Item\_Weight,y = Item\_Outlet\_Sales )) + geom\_point()*







**Inference**

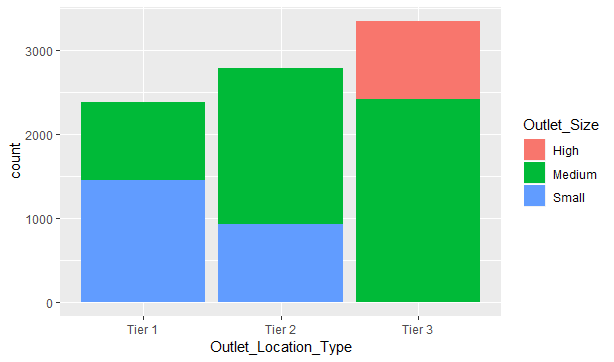
From the scatter plots we can see that there is no linear relationship between the given variables. So the use of these variables might not give a satisfactory linear model if not transformed.

**Barplots**

Here first we compare Outlet\_Location\_Type with Outlet\_Size to see how the location influence the size of the store

*Code*

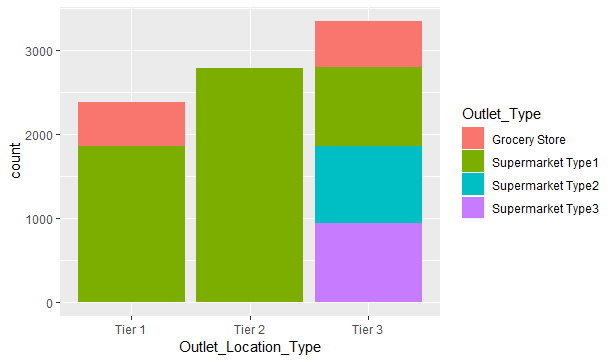
*ggplot(Train\_data,aes(x= Outlet\_Location\_Type, fill = Outlet\_Size)) + geom\_bar()*

**

**Inference** : Here we can see that Tier 1 has max no. of stores of small size Tier 2 has max no. of medium size stores and only Tier 3 has High sized stores. Thus location definitely has an influence on the size of the store.

Now we compare Outlet\_Location\_Type with Outlet\_Type to see how the location influence the type of the store

*Code: ggplot(Train\_data,aes(x= Outlet\_Location\_Type, fill = Outlet\_Type)) + geom\_bar()*

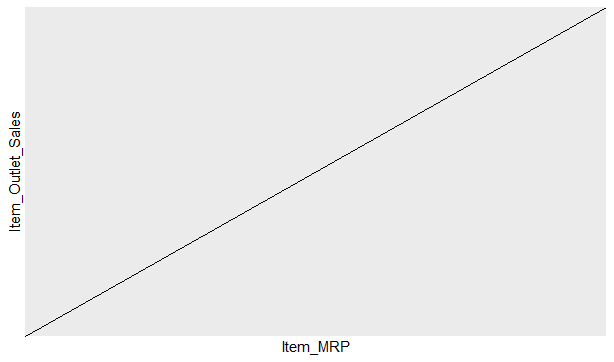
**

**Inference:** Tier 1 has grocery stores and supermarket type 1, Tier 2 has only supermarket type while Tier 3 has all the types of stores.

Now we would like to see how the MRP effects the sales of the item. For this we plot a simple line graph.

*Code*

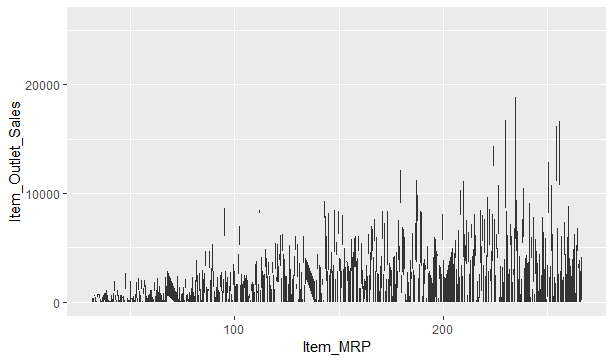
*ggplot(Train\_data,aes(x= Item\_MRP, y = Item\_Outlet\_Sales)) + geom\_abline()*

**

**Inference:** We can see that item of higher price has higher rate of sale.

To demonstrate it better and for the better understanding of the Sales and MRP relationship we here using a area graph.

*ggplot(Train\_data,aes(x= Item\_MRP, y = Item\_Outlet\_Sales)) + geom\_area()*



Thus we can understand that it has an increasing trend.

**Model Building**

After cleaning the data by the different procedures we fit the linear model on the train set of the data.

Here while building the model we take the following independent variables-

1. Item\_MRP
2. Outlet\_Location\_Type

We selected only this variables because, by testing other variables in the model we found them irrelevant.

*Codes*

*lm4=lm(log(Item\_Outlet\_Sales)~Item\_MRP+Outlet\_Type, data=new\_train )*

Output

Call:

*lm(formula = log(Item\_Outlet\_Sales) ~ Item\_MRP + Outlet\_Type,*

*data = new\_train)*

Residuals:

Min 1Q Median 3Q Max

-2.33248 -0.29288 0.07126 0.38095 1.36979

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 4.381e+00 2.099e-02 208.71 <2e-16 \*\*\*

Item\_MRP 8.301e-03 9.369e-05 88.59 <2e-16 \*\*\*

Outlet\_TypeSupermarket Type1 1.955e+00 1.789e-02 109.31 <2e-16 \*\*\*

Outlet\_TypeSupermarket Type2 1.774e+00 2.409e-02 73.62 <2e-16 \*\*\*

Outlet\_TypeSupermarket Type3 2.484e+00 2.405e-02 103.30 <2e-16 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.5386 on 8518 degrees of freedom

Multiple R-squared: 0.7197, Adjusted R-squared: 0.7196

F-statistic: 5468 on 4 and 8518 DF, p-value: < 2.2e-16

**Interpretation**

* From the above table we can see that the value of **Multiple R-squared = 0.72**

R-squared value measures the proportion of the variation in your dependent variable explained by all of your independent variables in the model.

Thus getting a high R-squared values signifies that the model is good.

* From the table we can see that the value of **Adjusted R-squared = 0.7197**

Adjusted R-squared value measures the proportion of variation explained by only those independent variables that really affect the dependent variable. It penalizes you for adding independent variable that do not affect the dependent variable.

Thus again higher the value of Adjusted R-squared better is the model

* Here **p-value: < 2.2e-16 ,** thus the null hypothesis that the dependent and independent variables are not linearly related is rejected.
* **Residual standard error = 0.5385**, it explains how close the actual data points are to the model’s predicted values. It measures standard deviation of the residuals.

**Model Evaluation**

*The VIF of the above independent variables is*

*Code*

*library(car)*

*vif(lm3)*

Output

**GVIF Df GVIF^(1/(2\*Df))**

**Item\_MRP**  1.000078 1 1.000039

**Outlet\_Type**  1.000078 3 1.000013

Here by look at the vif we can see that all the VIF values are less that 2. So we can say that there is no multicollinearity in between the variables.

*RMSE*

*Code*

*library(Metrics)*

*rmse(new\_train$Item\_Outlet\_Sales, exp(lm4$fitted.values))*

*Output:*

1142.023

Here we can see that the RSME value is also quite low

*Durbin Watson Statistic*

*Code:*

*durbinWatsonTest(lm4)*

*Output:*

**lag Autocorrelation D-W Statistic p-value**

1 -0.003549967 2.007044 0.71

Alternative hypothesis: rho != 0

Durbin Watson test(dwt ) -is done to check whether the errors are correlated or not.

Here the

Null hypothesis is-The errors are uncorrelated and

Alternate hypothesis is- The errors are correlated.

From the results that we get here we can see that p value is0.71which is higher than 0.05(Level of significance).

Thus we accept the null hypothesis here.

We can also see that the autocorrelation is also close to zero.

**Forecasting**

*Codes:*

*library(forecast)*

*prediction=predict.lm(lm4, newdata=new\_test)*

*a1=as.data.frame(prediction)*

*View(a1)*

*newdata=cbind(Test\_data,a1)*

*View(newdata)*

*predict\_final= subset(newdata, select = -c(Item\_Outlet\_Sales))*

*View(predict\_final)*

**Full Source Code**

*#Importing the data*

*Train\_data<-read.csv("Train\_UWu5bXk.csv", header = T, na.strings = "")*

*View(Train\_data)*

*Test\_data<-read.csv("Test\_u94Q5KV.csv", header = T, na.strings = "")*

*summary(Test\_data)*

*#Univariate analysis*

*library(psych)*

*describe(Train\_data)*

*summary(Train\_data)*

*#Finding the no. of missing values in each variable*

*colSums(is.na(Train\_data))*

*colSums(is.na(Test\_data))*

*#combining test and train*

*Test\_data$Item\_Outlet\_Sales <- 1*

*combi <- rbind(Train\_data, Test\_data)*

*#Replacing the missing values by mean for Item\_Weight and by "Others" Outlet\_Size respectively*

*combi$Item\_Weight[which(is.na(combi$Item\_Weight))]<- mean(combi$Item\_Weight,na.rm = T)*

*combi$Outlet\_Size[is.na(combi$Outlet\_Size)] <- "Medium"*

*summary(combi)*

*#renaming levels of Item\_Fat\_Content*

*library(plyr)*

*combi$Item\_Fat\_Content <- revalue(combi$Item\_Fat\_Content, c("LF" = "Low Fat"))*

*combi$Item\_Fat\_Content <- revalue(combi$Item\_Fat\_Content, c("low fat" = "Low Fat"))*

*combi$Item\_Fat\_Content <- revalue(combi$Item\_Fat\_Content, c("reg" = "Regular"))*

*summary(combi)*

*#Visualization of the data.*

*library(ggplot2)*

*#line plots*

*ggplot(data = Train\_data,aes(x = Item\_MRP,y = Item\_Outlet\_Sales )) + geom\_point()*

*ggplot(data = Train\_data,aes(x = Item\_Visibility,y = Item\_Outlet\_Sales )) + geom\_point()*

*ggplot(data = Train\_data,aes(x = Item\_Weight,y = Item\_Outlet\_Sales )) + geom\_point()*

*#barplots*

*ggplot(Train\_data,aes(x= Outlet\_Location\_Type, fill = Outlet\_Size)) + geom\_bar()*

*ggplot(Train\_data,aes(x= Outlet\_Location\_Type, fill = Outlet\_Type)) + geom\_bar()*

*ggplot(Train\_data,aes(x= Item\_MRP, y = Item\_Outlet\_Sales)) + geom\_abline()*

*ggplot(Train\_data,aes(x= Item\_MRP, y = Item\_Outlet\_Sales)) + geom\_area()*

*#removing the variables which is unnecessary*

*combi = subset(combi,select=-c(Item\_Identifier,Item\_Type,Outlet\_Identifier))*

*#sampling*

*new\_train <- combi[1:nrow(Train\_data),]*

*new\_test <- combi[-(1:nrow(Train\_data)),]*

*#Fitting model excluding Item\_Weight and Item\_Visibility*

*lm2=lm(Item\_Outlet\_Sales~Item\_Fat\_Content+Item\_MRP+Outlet\_Establishment\_Year+Outlet\_Size+Outlet\_Location\_Type+Outlet\_Type, data=new\_train )*

*summary(lm2)*

*#Fitting model taking log transformation of Item\_Outlet\_Sales*

*lm4=lm(log(Item\_Outlet\_Sales)~Item\_MRP+Outlet\_Type, data=new\_train )*

*summary(lm4)*

*#Finding VIF*

*library(car)*

*vif(lm4)*

*# get Durbin Watson statistic*

*durbinWatsonTest(lm4)*

*#calculate RMSE*

*library(Metrics)*

*rmse(new\_train$Item\_Outlet\_Sales, exp(lm4$fitted.values))*

*#Forecasting*

*library(forecast)*

*prediction=predict.lm(lm4, newdata=new\_test)*

*a1=as.data.frame(prediction)*

*View(a1)*

*newdata=cbind(Test\_data,a1)*

*View(newdata)*

*predict\_final= subset(newdata, select = -c(Item\_Outlet\_Sales))*

*View(predict\_final)*