Big mart sales prediction based on linear regression and Lasso regression

Weilin Cheng

American Heritage School Plantation Campus

williamcheng200102@gmail.com

***Abstract* — The generalized linear regression is a flexible generalization of ordinary linear regression that allows for response variables that have error distribution models other than a normal distribution. And the unknown parameters are typically estimated with maximum likelihood. Lasso regression which is the generalized linear regression via penalized maximum likelihood can avoid the overfitting issue and help in selecting variables. Lasso regression uses shrinkage, where data values are shrunk towards a central point. This paper will demonstrate how generalized linear regression and generalized Lasso regression perform in big mart sales prediction.**

I Introduction

Generalized Linear regression also known as "GLM" that have error distribution models other than a normal distribution. The Generalized linear regression were invented by John Nelder and Rober Wedderburn, the purpose is to combine several statistical models together including linear regression, logistic regression and poisson regression. In this paper, the prediction model is built based on linear regression.

$$y_i = \beta_0 + \beta_1 x_{1i} + ... + \beta_p x_{pi} + \varepsilon_i$$

$$\min \sum_{i=1}^{n} (y_i - \sum_j x_{ij}\beta_j)^2$$

Lasso regression is a type of linear regression that uses shrinkage, which is a penalty item $\lambda \sum_{j+1}^{p} |\beta_j|$. Below model offers the option to specify a Lasso, ridge or elastic net penalty. When $\alpha = 1$, it's a Lasso regression. $\alpha = 0$, it's a Ridge regression. When $0 < \alpha < 1$, it's a ElasticNet regression.[2]

$$y_i = \beta_0 + \beta_1 x_{1i} + ... + \beta_p x_{pi} + \varepsilon_i$$

$$\min \sum_{i=1}^{n} (y_i - \sum_j x_{ij}\beta_j)^2 + \lambda((1-\alpha)\sum_{j=1}^{p} ||\beta_j||^2 + \alpha \sum_{j=1}^{p} |\beta_j|)$$

$$= RSS + \lambda((1-\alpha)\sum_{j=1}^{p} ||\beta_j||^2 + \alpha \sum_{j=1}^{p} |\beta_j|$$

A tuning parameter λ, which controls the strength of the L1 penalty. λ is basically the amount of shrinkage:

- When λ = 0, no parameters are eliminated. And it's the same as linear regression.
- As λ increases, more and more coefficients are set to zero and eliminated.
- As λ increases, bias increases.
- As λ decreases, variance increases.

People use Lasso regression model to predict data that will change in the future to make plans for the future. For many shops and supermarkets it is essential for them to see what can be the most factors that will influence the sales rates.

In machine learning, a set of data is given and a model is used to predict a result. This paper will focus on the process of training the big mart sales data. And predicting the sales on the testing set.

In data cleaning section, there are some missing data for Item weight and Item visibility in the mart. Missing data imputation methodology can be performed for those sets.

For outlier determination, we can use histogram and qq plot to check the distribution of the outlier and then use the corresponding methodology to detect outliers.

When selecting the variables, this paper will use "stepwise" variable selection method and pick up the important variables according to their significant.

In the modeling part, both generalized linear regression model and Lasso model will be built to predict the sales for different outlets and products. And then comparing the fitness of model in the testing data set.

II Materials and Methodology

For this task, R is the first-choice programming language. Its intuitive syntax and data analysis libraries makes coding more efficient. The following libraries are planned to be used.

- Outliers
- Stats
- Caret
- Extremevalues
- Corrplot
- MASS
- CaretEnsemble

A.Data Overview

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.[1] Below are the variables and its corresponding descriptions. We will use the variable name in the following paper.

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.

B. Data Cleaning

1. Missing Data Imputation

While checking the missing data, there are some missing entries in the Item Weight category. And there are two cases. First case is we can find the missing weight from another store for the specific product. The other case is there is only one record for some products, so we can use the average weight from the whole column to impute the missing values.

For the outlet size, according to the summary, we can find that there are missing values that need to be filled out. So I used mode which is "Medium" and replaced the empty value by mode.

When going over the Item visibility, few values that is 0, which is not possible, since the visibility cannot be 0. So I have to use the average item visibility from the same kind of product to replace the zero values.

2. Data correction

For the Item Fat Content we realized that there are typos in this section, where it marked "Low Fat" as "LF" and "low fat"; and marked "Regular" as "reg". In order to solve this problem we find out all the "LF/low fat"and "reg" and replace them with "Low Fat" and "Regular". And then I replace the item fact content for item type "Health and Hygiene,Household, and Others" with "NA" since they are not food and drink.

In order to avoid overfitting issue, we can combine the item type into "Food", "Drink", and "Non-Consumable" these three categories.

Results:

```
        Var1  Freq
1      Drink  1317
2       Food 10201
3 Non-Consumable  2686
```

3. Data Transformation

We can use the year operation instead of establishment year (2013 -Outlet_Establishment_Year) , which can be used as a numeric variable in the model.

C. Outlier

First, histogram and qq diagram can be used to check whether the Item outlet sales are normally distributed. Below is the histogram for Sales price. We can tell that sales data are obviously not normally distributed.

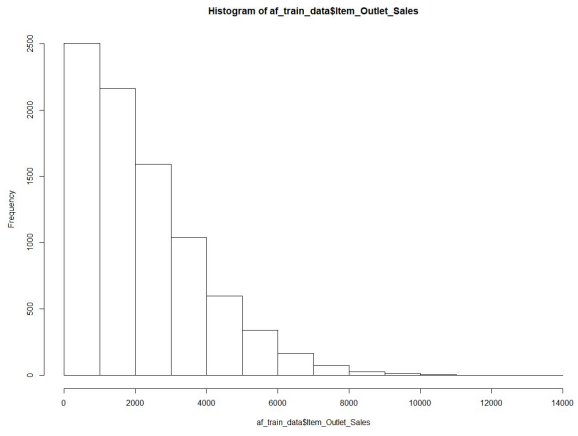

Figure 1: Histogram for sales data

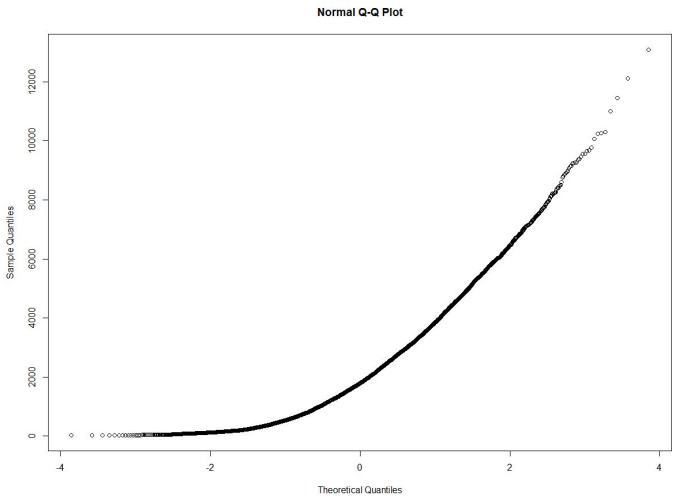

Figure 2: normal distribution qq plot

Then, boxplot is a good technique to detect the potential outliers which outside the 90% percentile of data. And I also used qq plot to test whether it is an exponential distribution as below. Below are the boxplot figure for sales data grouped by different outlets.

By looking at the box Figure 3, each outlet have few potential outliers. And OUT027 has much higher potential sales outlier data. Also OUT027 has higher average sales compared to other outlets. From Figure 4, we can see it's an approximate exponential distribution.

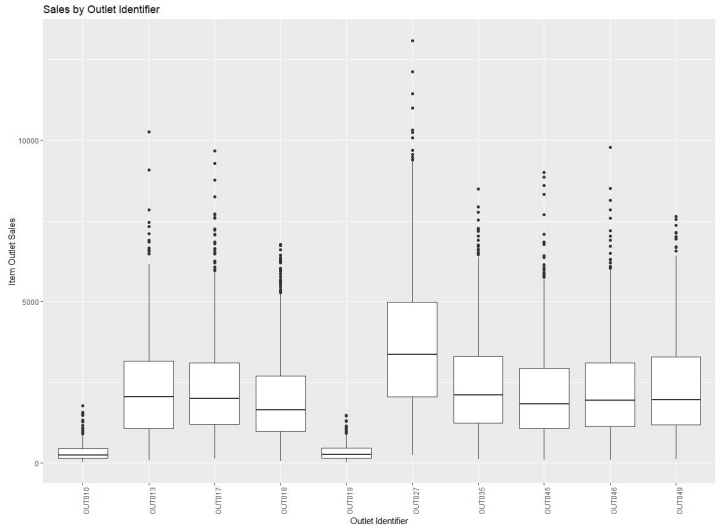

Figure 3: Box Whisker for outliers group by outlet

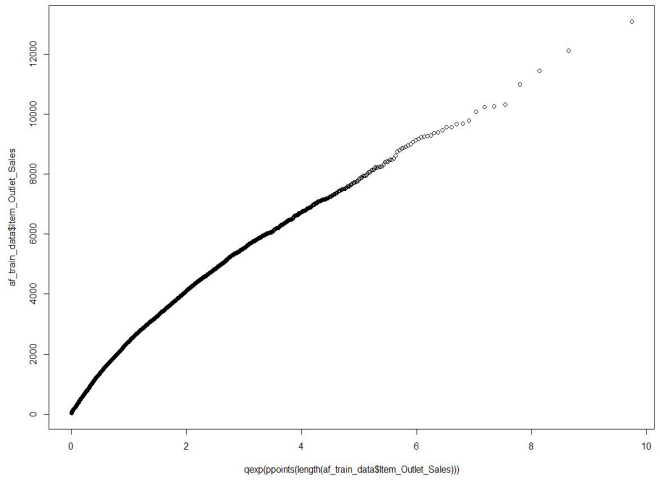

Figure 4: exponential distribution qq plot

And I used getOutlier in R to test the outliers which give use the results saying there is no outlier. So we don't need to remove any outliers here according to getOutliers tool.

```
getOutliers(af_train_data$Item_Outlet_Sales, distribution =
'exponential')
```

Below are the results from the getOutlier function performed above.

getOutliers:

Left	Right
0	0

D. Variable Selection

Before using the linear regression model, we need to check the correlation between variables. From the below correlation plot, we didn't find any positive/negative correlation between variables. But we can see there is a strong correlation between maximum retail price and sales.

```
corMatrix <-
cor(af_train_data[1:nrow(af_train_data),][sapply(af_train_data[1:n
row(af_train_data),], is.numeric)])
corMatrix
```

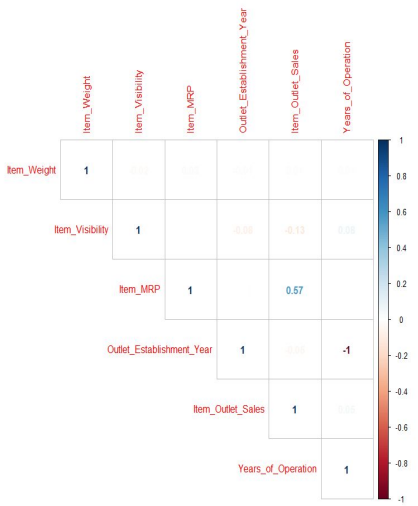

Figure 5: Correlation between different variables

After checking the correlation between variables, we can use "stepwise" variable selection for linear model.

```
model1 <- glm(Item_Outlet_Sales~
          Item_Weight
          +Item_Fat_Content
          +Item_Visibility
          +Combined_Item_Type
          +Item_MRP
          +Outlet_Location_Type
          +Outlet_Size
          +Outlet_Type
          +Years_of_Operation,
          data = af_train_data)

v_selection <- stepAIC(model1, direction = "both", trace =
FALSE)
```

From the below results, we can find out that variable - Item MRP, Location Type Tier, Outlet Size, Outlet Type and years of operation are significant. Here, I included these variables in the following modeling. Besides I add combined item type in the model to distinguish the product type.

```
Coefficients:
                                Estimate Std. Error t value Pr(>|t|)
(Intercept)                    116.0664   530.6665   0.219
0.826875
Item_Fat_ContentNone           -24.2064    33.4981  -0.723
0.469934
Item_Fat_ContentRegular         43.2439    27.3689   1.580
0.114135
Item_MRP                        15.5619     0.1964  79.250  < 2e-16
***
Outlet_Location_TypeTier 2    -228.8183    71.5159  -3.200
0.001382 **
Outlet_Location_TypeTier 3    -459.3302   143.4466  -3.202
0.001369 **
Outlet_SizeMedium             -919.3515   253.7750  -3.623
0.000293 ***
Outlet_SizeSmall              -843.1362   247.5967  -3.405
0.000664 ***
Outlet_TypeSupermarket Type1  1469.3914   139.7555  10.514  <
2e-16 ***
Outlet_TypeSupermarket Type2  1202.2189   127.7073   9.414  <
2e-16 ***
Outlet_TypeSupermarket Type3  3888.6350   145.9929  26.636
< 2e-16 ***
Years_of_Operation             -39.9401    10.2210  -3.908
9.39e-05 ***
```

E. Generalized Linear Regression Model vs LASSO model

With the selection of the variables with high significance, I built a generalized linear regression model[3] and Lasso regression in the training dataset with 3 fold cross validation.

```
control <- trainControl(method = "repeatedcv", number = 10,
repeats = 3, savePredictions = TRUE, classProbs = TRUE)
mList <- c('glm','glmnet')

fit_models <- caretList(Item_Outlet_Sales~
                  Combined_Item_Type
                  +Item_MRP
                  +Outlet_Identifier
                  +Outlet_Location_Type
                  +Outlet_Size
                  +Outlet_Type
                  +Years_of_Operation,
                  data = af_train_data,
                  trControl = control,
                  methodList = mList)
```

Here, the RMSE(Root Mean Square Error) for generalized linear regression model is 1128.372. And for the glmnet model we have the optimal value for the lowest RMSE is alpha = 1 and lambda = 1.937. So this is a LASSO regression model, and LASSO regression model also can help us select the variariable. With the optimal value for penalty item, I have the RMSE of model equal to 1128.335 which is slightly lower than generalized linear regression model.

```
$glm
Generalized Linear Model

  RMSE      Rsquared  MAE
  1128.372  0.563201  837.2549

$glmnet
  alpha  lambda     RMSE      Rsquared   MAE
  0.10    1.937018  1128.384  0.5631983  836.9739
  0.10   19.370175  1128.694  0.5630865  836.5560
  0.10  193.701751  1152.765  0.5537720  852.0912
  0.55    1.937018  1128.352  0.5632209  836.9141
  0.55   19.370175  1129.686  0.5626180  836.2742
  0.55  193.701751  1209.027  0.5178554  895.9940
  1.00    1.937018  1128.335  0.5632334  836.8594
  1.00   19.370175  1131.297  0.5617536  836.6575
  1.00  193.701751  1252.429  0.4958680  927.8833
```

```
RMSE was used to select the optimal model using the smallest value.
The final values used for the model were alpha = 1 and lambda = 1.937018.
```

III. Model Prediction Results

Based on the generalized linear regression and Lasso regression built above, I used prediction function to predict the sales on test data. And I have the RMSE for glm is 1202.0354, and for the glmnet model, I have RMSE equal to 1202.3587. The fitness of two models are very close. However, Lasso model helps to reduce the model complexity and minimize the error for the quantitative response variables. Also it avoids the overfitting issue. So I would still suggest Lasso as the better prediction model.

```
glmnet_model <- caretStack(fit_models, methodList = "glmnet", trControl = trainControl(method = "repeatedcv", number = 10, repeats = 3, savePredictions = TRUE))
glmnet_model

predict_on_test <- predict(glmnet_model, newdata = af_test_data )
Predict_on_test

glm_model <- caretStack(fit_models, method = "glm", trControl = trainControl(method = "repeatedcv", number = 10, repeats = 3, savePredictions = TRUE))
glm_model

predict_on_test2 <- predict(glm_model, newdata = af_test_data )
predict_on_test2
```

IV. Discussion & Future Work

In the outlier detection section, I used qq plot to distinguish the distribution of the sales data. It's close to exponential distribution. But it has some skewness and bias, which may influence the accuracy of getOutliers function. In this case, There are some improvements can be made to the outlier detection part.

Also, in the variable selection, we can figure out a way to combine the outlet identifier properly, which may help to reduce the model complexity and increase the model freedom degree.

V. I. Acknowledgement

I would like to thank Serena Zheng for insightful ideas and comments on this topic. I would also like to thank Xu Yuan who gave suggestions to this paper.

VI. Literature Cited

[1]https://datahack.analyticsvidhya.com/contest/practice-problem-big-mart-sales-iii/

[2]https://www.statisticshowto.datasciencecentral.com/lasso-regression/

[3]http://statmath.wu.ac.at/courses/heather_turner/glmCourse_001.pdf

[4]https://towardsdatascience.com/understanding-boxplots-5e2df7bcbd51