Final Group Project: Updated Checkpoint Submission

Team 5

2024-04-10

Introduction

Global Super Store is a data set which has around 50000 values. Its a customer centric data set , which has the data of all the orders that have been placed through different vendors and markets , starting from the year 2011 till 2014.

It provides data regarding profit gained over sale of various types of products, and can be used by a company planning to launch a certain type of product in a market.

Link to Project Github - https://github.com/Ayush-Srillex/DataScienceR

Data Columns are self-describing through their name, there data type are as follows:

Table 1: Column Type

Column	Type
Row ID	ID
Order ID	ID
Order Date	Date
Ship Date	Date
Ship Mode	Categorical
Customer ID	ID
Customer Name	Text
Segment	Categorical
City	Categorical
State	Categorical
Country	Categorical
Postal Code	Text
Market	Categorical
Region	Categorical
Product ID	ID
Category	Categorical
Sub-Category	Categorical
Product Name	Text
Sales	Numeric
Quantity	Numeric
Discount	Numeric
Profit	Numeric
Shipping Cost	Numeric
Order Priority	Categorical

Research question

If the company plans to release a new product in the market to maximize profits:

- Which category of product should it choose?
- Which region should it target the strongest?
- What kind of customer should it look towards?

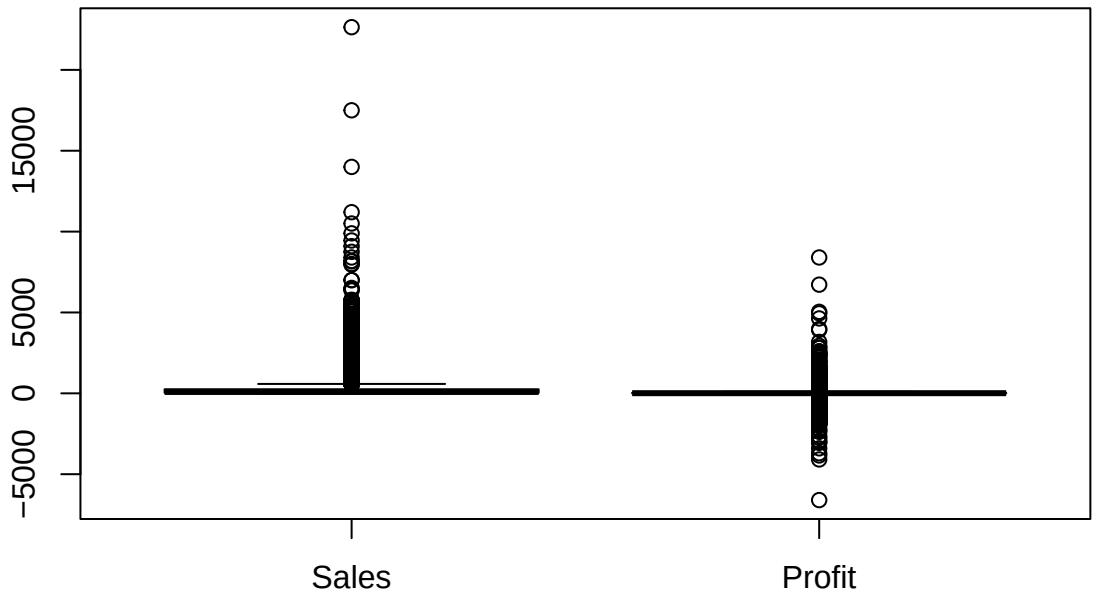
Once developed, company would also like to know:

- When would be the best timing to release products to maximize profits?
- What is the optimal quantity to sell in bulk to customers in terms of shipping costs/ profits?

#Data Preparation and Cleaning

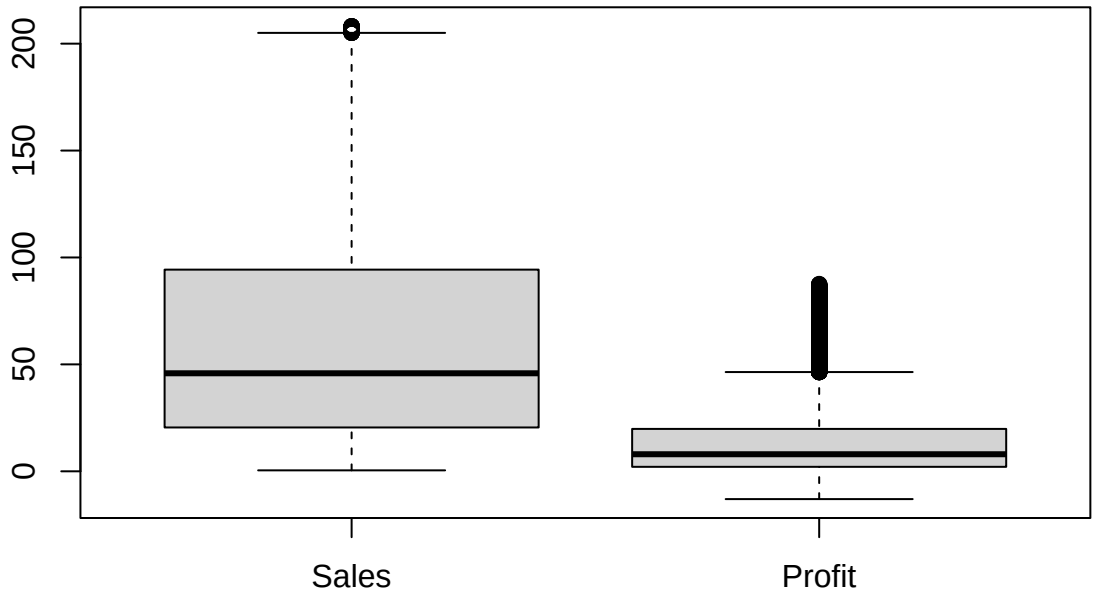
```
superstore <- read_excel("Global Data Superstore.xls",sheet = "Orders")

#Check Boxplots for Sales and Profit Variables:
boxplot(superstore[c(19,22)])
```



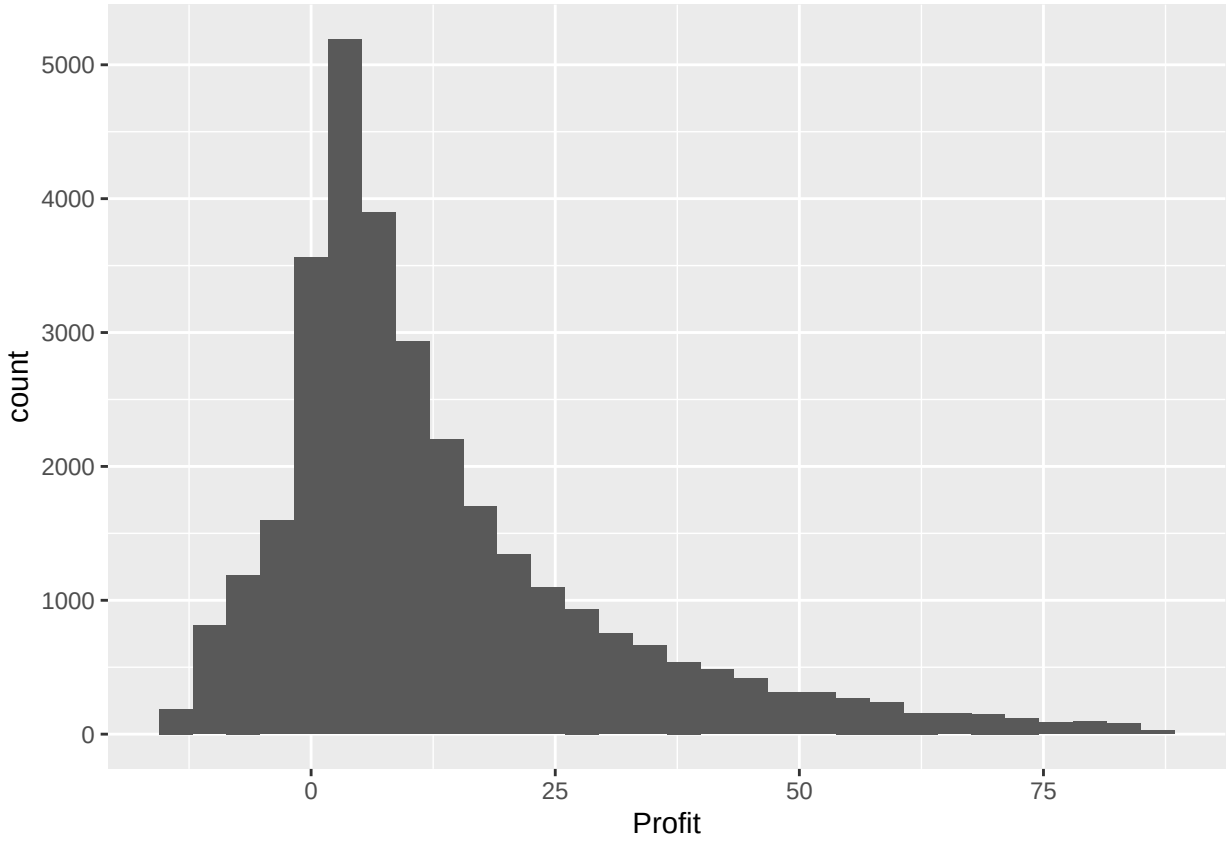

```
#Removing Outliers- 15% cutoff from both sides
df<-superstore[ superstore$Profit > quantile(superstore$Profit , 0.15 ) , ]
df<-df[ df$Profit < quantile(df$Profit , 0.85 ) , ]
```

```r
#Boxplot for Sales and profit after Outlier Removal
df<-df[df$Sales<quantile(df$Sales,0.85) ,]
boxplot(df[c(19,22)])
```



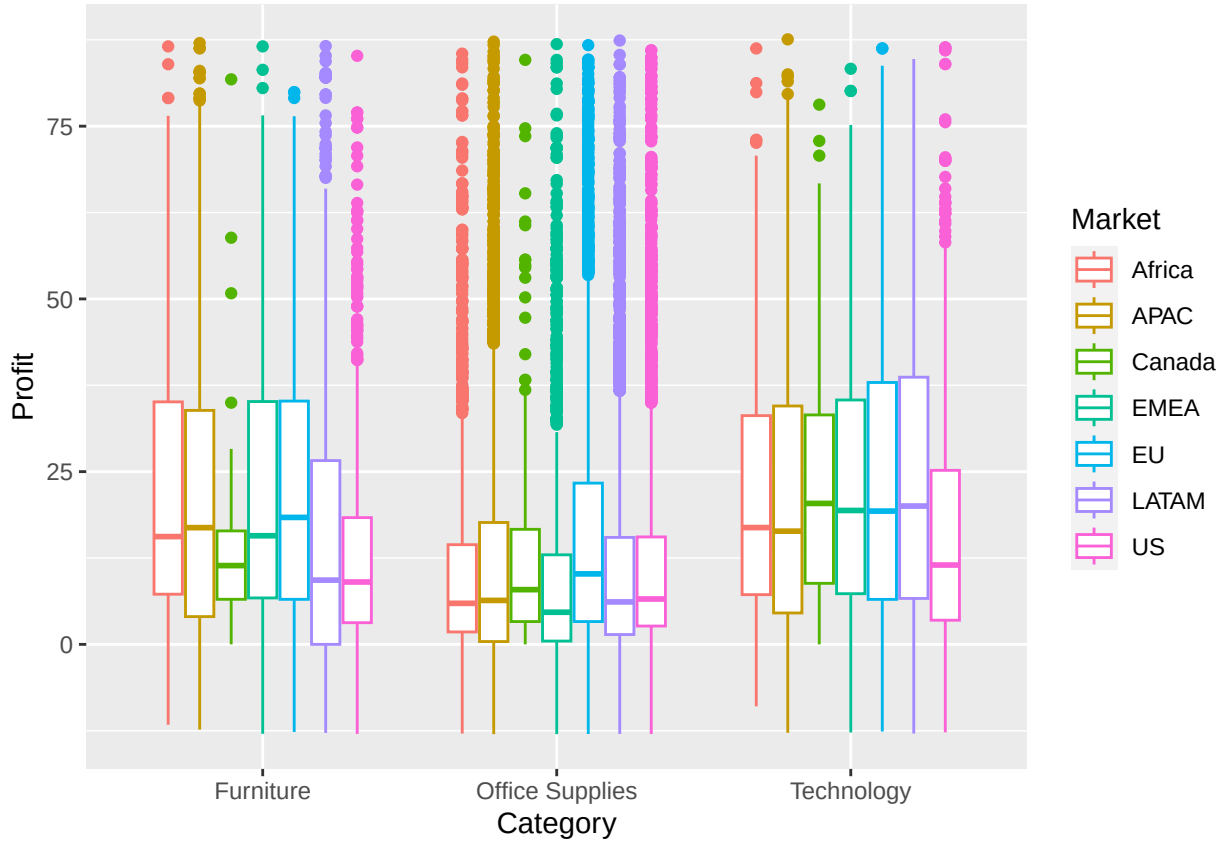
We can clearly see the distribution of dataset becoming more normal after outlier removal. We will not completely ignore these data points. Instead, we'll treat them separately # Exploratory Data Analysis (EDA)

```r
ggplot(data=df)+
  geom_histogram(mapping = aes(x= Profit))
```



Profit is more evenly distributed. We will not completely neglect the cutoff points, we will deal with separately as to why some products have high profits and high loss. Now we can analyse profit with each categorical variable clearly:

```
ggplot(data=df)+
  geom_boxplot(mapping = aes(x= Category,y = Profit,color = Market))
```



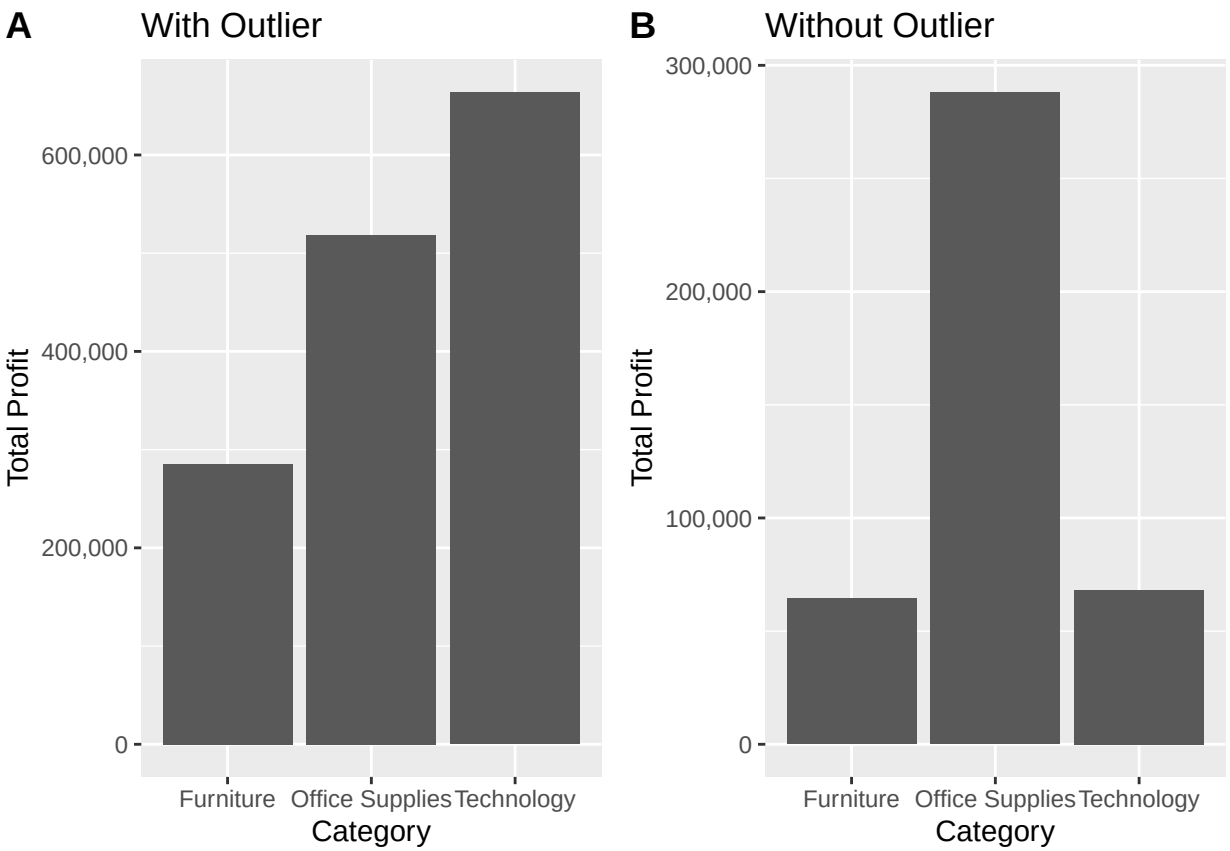
We can clearly see dependence of Profit on Category of Product. Further segregating each category into different markets where they were sold, we can see that Furniture has high dependency of profit over market than other category.

We will now plot total profit against various parameters. We will also look to analyse how removing the outliers affected our dataset:

```
#With Outliers:
p1<-ggplot(superstore, aes(x=Category, y=Profit)) +
  stat_summary(fun =sum, geom="bar") +
  labs(title="With Outlier", y="Total Profit", x="Category") +
  scale_y_continuous(labels = comma)

#Without Outliers
p2<-ggplot(df, aes(x=Category, y=Profit)) +
  stat_summary(fun =sum, geom="bar") +
  labs(title="Without Outlier", y="Total Profit", x="Category") +
  scale_y_continuous(labels = comma)

plot_grid(p1, p2, labels = "AUTO")
```

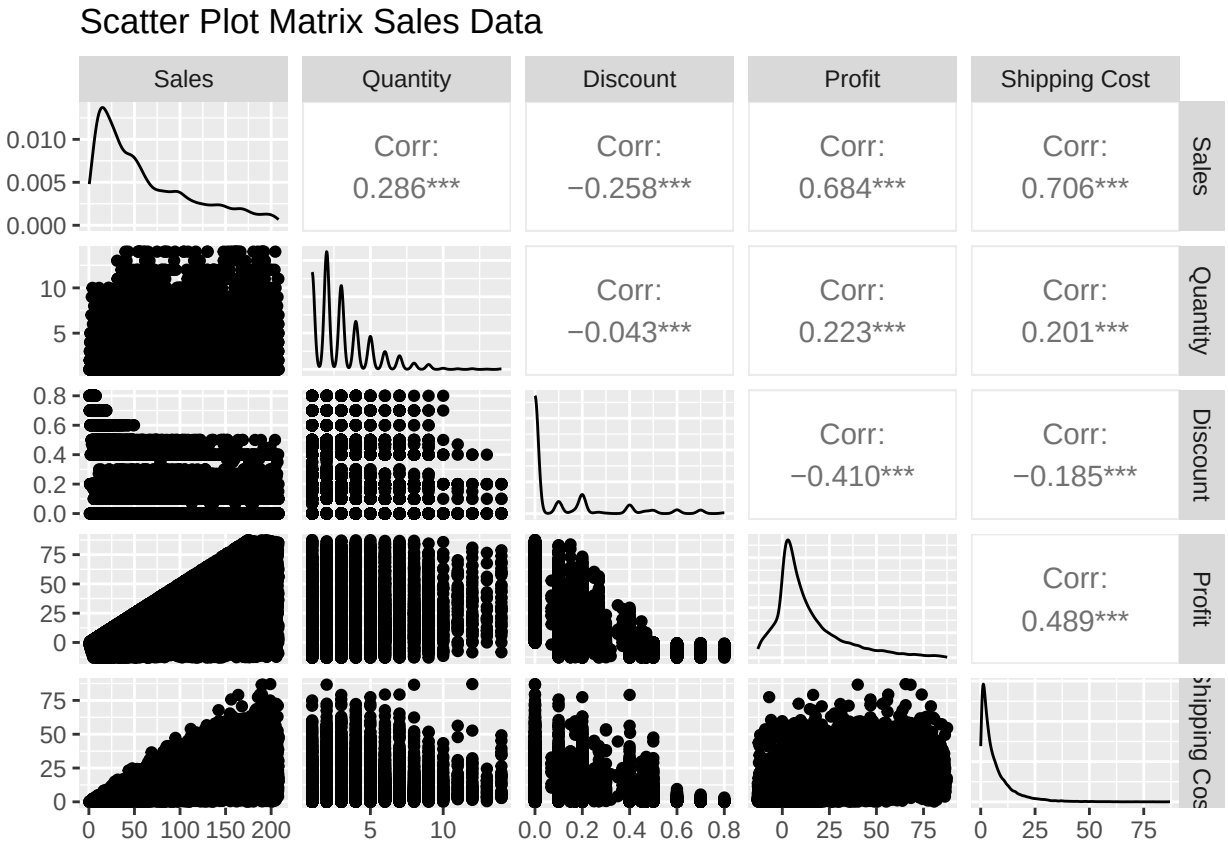


Removing outliers actually changes which category is the most profitable one in terms of pure sum of profit.

In the without outlier data, Office Supplies turns out to be the most profitable one, but one the overall dataset, Technology products are the most profitable one

We now move to visualise all numeric variables in our dataset: Sales, Quantity, Discount, Profit and Shipping Cost. We study the relationship between all the combinations of the variables as well.

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```



For our focus variable Profit, we see that it is positively correlated with Sales, Quantity and Shipping Cost, but negatively correlated with Discount. This makes sense with the common knowledge of commerce and finance.

We move to feature selection

Feature Selection

Before building models, we identify the most relevant features for predicting our target variable.

We start with identifying most important categorical variables. We carry out the similar analysis as before in EDA analysis but with some categorical variables and Profit as primary target variable. We will primarily use linear modeling to identify correlation with Profit. We will carry out test of Profit with Ship Mode, Segment, Market, Country, Sub-Category, Order Priority.

```
library(broom)
models <- list()
cat_variables=c("`Ship Mode`","Segment", "Country", "Market", "`Sub-Category`", "`Order Priority`")
table_list <- list()
i=1
for(c in cat_variables) {
  formula <- as.formula(paste("Profit", "~", paste(c,"-1")))
  reg<-lm(formula,data=df)

  coefs <- tidy(reg)
  a<-coefs[order(coefs$estimate, decreasing = TRUE),]
```

```
  table_list[[i]]<-knitr::kable(
    a,
    caption = c
  )
  i=i+1
}

for (i in seq_along(table_list)) {
  print(table_list[[i]])
}
```

```
##
##
## Table: ‘Ship Mode‘
##
## |term                       | estimate| std.error| statistic| p.value|
## |:--------------------------|--------:|---------:|---------:|-------:|
## |‘Ship Mode‘First Class     | 13.41009| 0.2594348|  51.68966|       0|
## |‘Ship Mode‘Standard Class  | 13.39333| 0.1283639| 104.33875|       0|
## |‘Ship Mode‘Second Class    | 13.26988| 0.2207273|  60.11890|       0|
## |‘Ship Mode‘Same Day        | 13.09881| 0.4296768|  30.48526|       0|
##
##
## Table: Segment
##
## |term                | estimate| std.error| statistic| p.value|
## |:-------------------|--------:|---------:|---------:|-------:|
## |SegmentHome Office  | 13.40822| 0.2323818|  57.69911|       0|
## |SegmentCorporate    | 13.35007| 0.1811724|  73.68712|       0|
## |SegmentConsumer     | 13.33928| 0.1380052|  96.65776|       0|
##
##
## Table: Country
##
## |term                                    |   estimate|  std.error|  statistic|   p.value|
## |:---------------------------------------|----------:|----------:|----------:|---------:|
## |CountrySwaziland                        | 43.3800000| 11.7116659|  3.7039991| 0.0002126|
## |CountryEritrea                          | 38.1000000| 11.7116659|  3.2531666| 0.0011425|
## |CountrySlovenia                         | 35.3700000| 11.7116659|  3.0200657| 0.0025292|
## |CountryLesotho                          | 32.0625000|  8.2813983|  3.8716288| 0.0001083|
## |CountryGuadeloupe                       | 29.0750000|  8.2813983|  3.5108805| 0.0004473|
## |CountryParaguay                         | 28.6177778|  5.5209322|  5.1835046| 0.0000002|
## |CountryLebanon                          | 26.5971429|  6.2601487|  4.2486439| 0.0000216|
## |CountryQatar                            | 25.4345455|  4.9938711|  5.0931522| 0.0000004|
## |CountryDjibouti                         | 24.5737500|  5.8558329|  4.1964568| 0.0000272|
## |CountryBangladesh                       | 24.5289474|  1.6993071| 14.4346761| 0.0000000|
## |CountryArmenia                          | 23.0300000|  9.5625351|  2.4083572| 0.0160302|
## |CountryBurundi                          | 22.8000000| 16.5627967|  1.3765791| 0.1686522|
## |CountryMalaysia                         | 22.6282105|  1.6993071| 13.3161397| 0.0000000|
## |CountrySingapore                        | 21.9857647|  1.7964875| 12.2381950| 0.0000000|
## |CountryEstonia                          | 21.8437500|  5.8558329|  3.7302550| 0.0001916|
## |CountryBelgium                          | 21.4943333|  1.7458721| 12.3115168| 0.0000000|
## |CountryTaiwan                           | 21.0600000|  6.2601487|  3.3641373| 0.0007688|
```

```
## |CountryZambia                          | 20.7570423|  1.9656423| 10.5599284| 0.0000000|
## |CountryJamaica                         | 20.4573913|  3.4535818|  5.9235287| 0.0000000|
## |CountryIndia                           | 20.3605537|  0.5457627| 37.3066050| 0.0000000|
## |CountryEthiopia                        | 20.3200000|  6.7617334|  3.0051466| 0.0026566|
## |CountryRepublic of the Congo           | 20.2650000| 11.7116659|  1.7303260| 0.0835819|
## |CountryChina                           | 20.1012166|  0.5028272| 39.9763870| 0.0000000|
## |CountryAfghanistan                     | 19.9363636|  2.8832128|  6.9146348| 0.0000000|
## |CountryTrinidad and Tobago             | 19.9094444|  3.9038886|  5.0999007| 0.0000003|
## |CountryJordan                          | 19.6738235|  2.8404962|  6.9261925| 0.0000000|
## |CountryMadagascar                      | 19.4936842|  2.6868404|  7.2552445| 0.0000000|
## |CountryNepal                           | 19.2257143|  4.4265936|  4.3432300| 0.0000141|
## |CountryFinland                         | 19.1288571|  2.7996236|  6.8326531| 0.0000000|
## |CountryChile                           | 19.0829474|  1.6993071| 11.2298404| 0.0000000|
## |CountryColombia                        | 19.0224829|  1.0475232| 18.1594853| 0.0000000|
## |CountryLibya                           | 18.8255172|  3.0756341|  6.1208572| 0.0000000|
## |CountryAustria                         | 18.8226761|  1.1348641| 16.5858410| 0.0000000|
## |CountrySpain                           | 18.6298686|  0.7228597| 25.7724539| 0.0000000|
## |CountryBarbados                        | 18.2152941|  4.0170682|  4.5344746| 0.0000058|
## |CountryNorway                          | 18.0474419|  2.5258028|  7.1452299| 0.0000000|
## |CountryGuatemala                       | 17.9862589|  0.8741510| 20.5756894| 0.0000000|
## |CountrySyria                           | 17.9535789|  3.7997662|  4.7249168| 0.0000023|
## |CountryItaly                           | 17.7877313|  0.6547020| 27.1692015| 0.0000000|
## |CountryJapan                           | 17.7508209|  1.4308077| 12.4061541| 0.0000000|
## |CountryHong Kong                       | 17.7320000|  4.2764957|  4.1463855| 0.0000339|
## |CountryEl Salvador                     | 17.6703616|  0.7355796| 24.0223641| 0.0000000|
## |CountryFrance                          | 17.6575443|  0.4054203| 43.5536747| 0.0000000|
## |CountryNicaragua                       | 17.4307153|  0.7959572| 21.8990606| 0.0000000|
## |CountryRomania                         | 17.4300000|  1.4755312| 11.8126949| 0.0000000|
## |CountryCentral African Republic        | 17.1480000|  7.4071079|  2.3150736| 0.0206153|
## |CountryGermany                         | 16.9011261|  0.4679055| 36.1208121| 0.0000000|
## |CountryAngola                          | 16.8745055|  1.7362529|  9.7189216| 0.0000000|
## |CountryBelarus                         | 16.7408824|  2.0085341|  8.3348757| 0.0000000|
## |CountryCuba                            | 16.6723667|  0.7377656| 22.5984603| 0.0000000|
## |CountryIraq                            | 16.6610265|  0.9530819| 17.4812126| 0.0000000|
## |CountryRussia                          | 16.5826071|  0.9898164| 16.7532148| 0.0000000|
## |CountryMoldova                         | 16.1477419|  2.9747661|  5.4282392| 0.0000001|
## |CountryUzbekistan                      | 16.1307692|  2.6521701|  6.0821022| 0.0000000|
## |CountryEgypt                           | 16.0963776|  0.8365476| 19.2414374| 0.0000000|
## |CountryUnited Kingdom                  | 15.9871348|  0.5353992| 29.8602153| 0.0000000|
## |CountrySlovakia                        | 15.9100000|  6.7617334|  2.3529469| 0.0186314|
## |CountryCameroon                        | 15.8972093|  1.7860123|  8.9009517| 0.0000000|
## |CountryUkraine                         | 15.7326545|  0.9987742| 15.7519630| 0.0000000|
## |CountrySouth Africa                    | 15.6145104|  0.9022330| 17.3065161| 0.0000000|
## |CountryMorocco                         | 15.4970571|  0.8853187| 17.5044950| 0.0000000|
## |CountryCambodia                        | 15.4755556|  3.1875117|  4.8550584| 0.0000012|
## |CountryMexico                          | 15.4452467|  0.3897398| 39.6296330| 0.0000000|
## |CountryMartinique                      | 15.1736842|  3.7997662|  3.9933205| 0.0000653|
## |CountryBosnia and Herzegovina          | 15.1361538|  4.5936933|  3.2949857| 0.0009854|
## |CountrySenegal                         | 15.0435000|  1.8517770|  8.1238185| 0.0000000|
## |CountrySwitzerland                     | 14.9760000|  2.4690360|  6.0655253| 0.0000000|
## |CountryPoland                          | 14.8276166|  1.1922162| 12.4370199| 0.0000000|
## |CountryEquatorial Guinea               | 14.8200000|  9.5625351|  1.5497982| 0.1212000|
## |CountryCroatia                         | 14.7807692|  3.2482317|  4.5504049| 0.0000054|
## |CountryBrazil                          | 14.7197531|  0.5536332| 26.5875532| 0.0000000|
```

```
## |CountryGhana                              | 14.7041667|  1.9519443|  7.5330872| 0.0000000|
## |CountryKyrgyzstan                         | 14.6962500|  2.9279165|  5.0193543| 0.0000005|
## |CountryIran                               | 14.6765762|  0.7567735| 19.3936181| 0.0000000|
## |CountryDemocratic Republic of the Congo   | 14.6568421|  0.9810954| 14.9392628| 0.0000000|
## |CountryCanada                             | 14.4669536|  0.9530819| 15.1791303| 0.0000000|
## |CountrySaudi Arabia                       | 14.4287547|  1.0174445| 14.1813677| 0.0000000|
## |CountryAlgeria                            | 14.2508000|  1.3523467| 10.5378304| 0.0000000|
## |CountryIsrael                             | 13.9604000|  1.9125070|  7.2995287| 0.0000000|
## |CountryKenya                              | 13.8962338|  1.8875059|  7.3622202| 0.0000000|
## |CountryTogo                               | 13.7378049|  2.5866743|  5.3109914| 0.0000001|
## |CountrySri Lanka                          | 13.6750000|  6.7617334|  2.0224104| 0.0431424|
## |CountryCzech Republic                     | 13.6067797|  2.1562925|  6.3102663| 0.0000000|
## |CountryCote d'Ivoire                      | 13.3777778|  1.8403107|  7.2693037| 0.0000000|
## |CountryGeorgia                            | 13.2288889|  3.1875117|  4.1502244| 0.0000333|
## |CountryHungary                            | 13.2005882|  2.3192554|  5.6917354| 0.0000000|
## |CountryAzerbaijan                         | 13.1400000|  3.3808667|  3.8865773| 0.0001019|
## |CountryBulgaria                           | 13.1117308|  2.2968466|  5.7085791| 0.0000000|
## |CountryNiger                              | 13.0627273|  2.8832128|  4.5306150| 0.0000059|
## |CountryMali                               | 12.9855882|  2.8404962|  4.5715915| 0.0000049|
## |CountryNew Zealand                        | 12.8422632|  0.8956135| 14.3390687| 0.0000000|
## |CountryEcuador                            | 12.8305263|  2.6868404|  4.7753213| 0.0000018|
## |CountryBolivia                            | 12.7458125|  2.9279165|  4.3532022| 0.0000135|
## |CountryUnited States                      | 12.4206887|  0.1996242| 62.2203479| 0.0000000|
## |CountryMozambique                         | 12.3958209|  2.0234677|  6.1260286| 0.0000000|
## |CountryLiberia                            | 12.2833333|  5.5209322|  2.2248658| 0.0260973|
## |CountryRwanda                             | 12.0631034|  3.0756341|  3.9221516| 0.0000879|
## |CountryAlbania                            | 11.8990909|  4.9938711|  2.3827389| 0.0171903|
## |CountryAustralia                          | 11.7542157|  0.4088645| 28.7484395| 0.0000000|
## |CountryTanzania                           | 11.6283025|  1.5183091|  7.6587189| 0.0000000|
## |CountryGabon                              | 11.2410000|  5.2376162|  2.1462054| 0.0318642|
## |CountrySudan                              | 11.0361290|  2.1034773|  5.2466119| 0.0000002|
## |CountryGuinea                             | 11.0275000|  4.7812676|  2.3063968| 0.0210950|
## |CountryMacedonia                          | 10.8450000|  8.2813983|  1.3095614| 0.1903538|
## |CountryTunisia                            | 10.7781818|  4.9938711|  2.1582819| 0.0309135|
## |CountrySomalia                            | 10.7080851|  2.4159322|  4.4322788| 0.0000094|
## |CountryNamibia                            |  9.9650000|  6.7617334|  1.4737345| 0.1405631|
## |CountryBenin                              |  9.3428571|  3.1300744|  2.9848675| 0.0028392|
## |CountryVietnam                            |  9.3275312|  1.3218551|  7.0563947| 0.0000000|
## |CountryUruguay                            |  9.2755556|  3.9038886|  2.3759785| 0.0175085|
## |CountryMongolia                           |  8.8692000|  3.3125593|  2.6774464| 0.0074224|
## |CountryGuinea-Bissau                      |  8.8600000|  6.7617334|  1.3103149| 0.1900989|
## |CountrySierra Leone                       |  7.3050000|  5.2376162|  1.3947185| 0.1631106|
## |CountryMyanmar (Burma)                    |  6.8767743|  1.9253860|  3.5716341| 0.0003553|
## |CountryBahrain                            |  6.6000000| 16.5627967|  0.3984834| 0.6902766|
## |CountryDominican Republic                 |  6.1448720|  0.7497626|  8.1957572| 0.0000000|
## |CountryMauritania                         |  5.5150000|  6.7617334|  0.8156193| 0.4147241|
## |CountryIndonesia                          |  3.5972181|  0.6547020|  5.4944356| 0.0000000|
## |CountryPhilippines                        |  2.9570435|  0.9969632|  2.9660508| 0.0030188|
## |CountryMontenegro                         |  2.4000000| 16.5627967|  0.1449031| 0.8847883|
## |CountryThailand                           |  1.6029575|  1.5119696|  1.0601784| 0.2890716|
## |CountryChad                               |  0.0000000| 16.5627967|  0.0000000| 1.0000000|
## |CountryPakistan                           | -0.3284831|  1.7556529| -0.1871003| 0.8515832|
## |CountrySouth Korea                        | -0.6714310|  2.1748018| -0.3087321| 0.7575274|
## |CountryPapua New Guinea                   | -0.7428000|  5.2376162| -0.1418202| 0.8872229|
```

```
## |CountryHaiti                          | -0.8764126|  2.2539111| -0.3888408| 0.6973966|
## |CountryVenezuela                      | -1.2764226|  1.7083220| -0.7471791| 0.4549611|
## |CountryPanama                         | -1.5733028|  1.1217741| -1.4025129| 0.1607720|
## |CountryPeru                           | -1.7818696|  1.7267909| -1.0318965| 0.3021286|
## |CountryArgentina                      | -2.3063947|  1.2079661| -1.9093207| 0.0562299|
## |CountryHonduras                       | -2.6591407|  0.8645710| -3.0756767| 0.0021021|
## |CountryNetherlands                    | -3.6300612|  1.3660764| -2.6572900| 0.0078812|
## |CountryIreland                        | -4.7304545|  2.8832128| -1.6406886| 0.1008721|
## |CountrySweden                         | -5.3484462|  2.0543621| -2.6034584| 0.0092332|
## |CountryTurkmenistan                   | -5.8080000|  8.2813983| -0.7013308| 0.4831018|
## |CountryTajikistan                     | -5.9190000| 11.7116659| -0.5053935| 0.6132860|
## |CountryLithuania                      | -5.9982857|  6.2601487| -0.9581698| 0.3379845|
## |CountryZimbabwe                       | -6.0838800|  3.3125593| -1.8366101| 0.0662769|
## |CountryDenmark                        | -6.2967391|  3.4535818| -1.8232488| 0.0682752|
## |CountryPortugal                       | -6.4413158|  3.7997662| -1.6951874| 0.0900498|
## |CountryTurkey                         | -6.4843065|  0.7436915| -8.7190807| 0.0000000|
## |CountryUganda                         | -6.6070000|  4.7812676| -1.3818511| 0.1670273|
## |CountryUnited Arab Emirates           | -6.7464000|  7.4071079| -0.9108008| 0.3624073|
## |CountryNigeria                        | -6.8671034|  1.0252114| -6.6982318| 0.0000000|
## |CountryKazakhstan                     | -7.1604545|  2.8832128| -2.4834984| 0.0130151|
## |CountryYemen                          | -9.3360000|  7.4071079| -1.2604110| 0.2075305|
##
##
## Table: Market
##
## |term         | estimate| std.error| statistic| p.value|
## |:------------|--------:|---------:|---------:|-------:|
## |MarketEU     | 16.30648| 0.2322729|  70.20400|       0|
## |MarketCanada | 14.46695| 1.0101575|  14.32148|       0|
## |MarketAPAC   | 13.50755| 0.2280789|  59.22314|       0|
## |MarketAfrica | 12.76615| 0.3204495|  39.83825|       0|
## |MarketLATAM  | 12.68094| 0.2172714|  58.36452|       0|
## |MarketUS     | 12.42069| 0.2115788|  58.70479|       0|
## |MarketEMEA   | 11.60800| 0.3129776|  37.08891|       0|
##
##
## Table: `Sub-Category`
##
## |term                       |  estimate| std.error| statistic|   p.value|
## |:--------------------------|---------:|---------:|---------:|---------:|
## |`Sub-Category`Copiers      | 33.422717| 1.1140837| 30.000185| 0.0000000|
## |`Sub-Category`Bookcases    | 28.513847| 0.9643008| 29.569452| 0.0000000|
## |`Sub-Category`Appliances   | 23.361150| 0.6747579| 34.621530| 0.0000000|
## |`Sub-Category`Machines     | 21.662717| 0.9217478| 23.501784| 0.0000000|
## |`Sub-Category`Phones       | 21.528478| 0.5369874| 40.091212| 0.0000000|
## |`Sub-Category`Chairs       | 21.498384| 0.5216677| 41.210875| 0.0000000|
## |`Sub-Category`Accessories  | 19.129132| 0.4141952| 46.183856| 0.0000000|
## |`Sub-Category`Paper        | 15.766378| 0.3054366| 51.619157| 0.0000000|
## |`Sub-Category`Furnishings  | 14.707674| 0.3567526| 41.226540| 0.0000000|
## |`Sub-Category`Supplies     | 14.492689| 0.3864006| 37.506902| 0.0000000|
## |`Sub-Category`Envelopes    | 13.336516| 0.3731683| 35.738604| 0.0000000|
## |`Sub-Category`Storage      | 13.183966| 0.3190181| 41.326706| 0.0000000|
## |`Sub-Category`Art          | 12.845158| 0.2639021| 48.673961| 0.0000000|
## |`Sub-Category`Binders      |  9.519355| 0.2317119| 41.082713| 0.0000000|
```

```
## |‘Sub-Category‘Fasteners   |  6.626665| 0.3550942| 18.661714| 0.0000000|
## |‘Sub-Category‘Tables      |  6.571408| 3.3135617|  1.983186| 0.0473554|
## |‘Sub-Category‘Labels      |  5.908213| 0.3367083| 17.546976| 0.0000000|
##
##
## Table: ‘Order Priority‘
##
## |term                     | estimate| std.error| statistic| p.value|
## |:------------------------|--------:|---------:|---------:|-------:|
## |‘Order Priority‘Critical | 13.63760| 0.3562714|  38.27869|       0|
## |‘Order Priority‘Medium   | 13.38136| 0.1310332| 102.12192|       0|
## |‘Order Priority‘High     | 13.28767| 0.1807418|  73.51740|       0|
## |‘Order Priority‘Low      | 12.99978| 0.4583847|  28.35998|       0|
```

We see that for categorical variables - Ship Mode, Segment, Order Priority have almost the same estimate for each of its categories. This means the Profit doesn't depend on these categories, each category would give the same Profit. We can run a t test to confirm the hypothesis that these estimates are not statistically significant.

For other categorical variables - Country, Market and Product Sub Category, there is difference in estimate for each category. The estimates are given in decreasing order to identify the categories with most positive impact on Profit. For example:

- In market variable- EU Market comes out to be most profitable one, and EMEA is the least profitable.
- In Country variable, we see that countries like Yemen and Kazakhstan as actually giving loss for the products sold there.

Other inferences can be drawn as per reader's wish.

Moving on to identifying most important numeric variables against Profit.

```
library(caret)
library(relaimpo)

cat_variables=c("Sales","Quantity","Discount","Shipping Cost","Profit")
df_train<-df[,cat_variables]
regressor <- lm(Profit ~ . , data = df_train)
relImportance <- calc.relimp(regressor, type = "lmg", rela = TRUE)
sort(relImportance$lmg, decreasing=TRUE)
```

```
##         Sales Shipping Cost      Discount      Quantity
##    0.57238081    0.19717086    0.19408118    0.03636715
```

We see that Sales is the most important feature for Profit.

With the above analysis, we identify the following most important features:

- Categorical Variables: Market, Product Sub-Category
- Numeric variables: Sales, Shipping Cost, Discount

Model Building

We divide the dataset in 70% train test split for model testing:

```
train_indices <- createDataPartition(df$Profit, p = 0.7, list = FALSE)
train_data <- df[train_indices, ]
test_data <- df[-train_indices, ]
```

Our preliminary model would be taking all the important variables: Model 0

```
reg0<-lm(Profit~Market+`Sub-Category`+Sales+`Shipping Cost`+Discount,data=train_data)
pred <- predict(reg0, newdata = test_data)

test0<-test_data
test0$predictions<-pred
```

Our next model would be taking the most important categorical and numeric variable. These are - Sub-Category and Sales

Model 1

```
reg1<-lm(Profit~`Sub-Category`+Sales,data=train_data)
summary(reg1)
```

```
##
## Call:
## lm(formula = Profit ~ `Sub-Category` + Sales, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -58.069  -6.093   0.723   5.933  52.409
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -3.507386   0.406351  -8.631  < 2e-16 ***
## `Sub-Category`Appliances    3.449831   0.706310   4.884 1.05e-06 ***
## `Sub-Category`Art           1.563560   0.442269   3.535 0.000408 ***
## `Sub-Category`Binders       3.007612   0.433834   6.933 4.24e-12 ***
## `Sub-Category`Bookcases    -4.102901   0.925933  -4.431 9.42e-06 ***
## `Sub-Category`Chairs       -4.947208   0.601996  -8.218  < 2e-16 ***
## `Sub-Category`Copiers       0.033718   1.078521   0.031 0.975060
## `Sub-Category`Envelopes     2.507919   0.500167   5.014 5.37e-07 ***
## `Sub-Category`Fasteners     1.810789   0.495810   3.652 0.000261 ***
## `Sub-Category`Furnishings   0.808935   0.489253   1.653 0.098262 .
## `Sub-Category`Labels        2.759145   0.489939   5.632 1.81e-08 ***
## `Sub-Category`Machines     -2.561398   0.899484  -2.848 0.004409 **
## `Sub-Category`Paper         5.643535   0.464225  12.157  < 2e-16 ***
## `Sub-Category`Phones       -2.147011   0.612706  -3.504 0.000459 ***
## `Sub-Category`Storage      -2.344240   0.467544  -5.014 5.37e-07 ***
## `Sub-Category`Supplies      0.826159   0.508546   1.625 0.104273
## `Sub-Category`Tables      -20.021943   3.017600  -6.635 3.32e-11 ***
## Sales                       0.247477   0.001895 130.612  < 2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.7 on 22032 degrees of freedom
## Multiple R-squared:  0.4831, Adjusted R-squared:  0.4827
## F-statistic:  1211 on 17 and 22032 DF,  p-value: < 2.2e-16
```

```
pred <- predict(reg1, newdata = test_data)

test1<-test_data
test1$predictions<-pred
```

Our next model takes Sales, Shipping Cost and Discount only, removing all categorical variables

Model 2

```
reg2<-lm(Profit~Sales+`Shipping Cost`+Discount,data=train_data)
summary(reg2)
```

```
##
## Call:
## lm(formula = Profit ~ Sales + `Shipping Cost` + Discount, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -53.969  -5.576  -0.139   5.347  48.008
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.791270   0.147480  18.926   <2e-16 ***
## Sales             0.205803   0.002243  91.752   <2e-16 ***
## `Shipping Cost`   0.027406   0.014269   1.921   0.0548 .
## Discount        -24.828487   0.477272 -52.022   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.19 on 22046 degrees of freedom
## Multiple R-squared:  0.5235, Adjusted R-squared:  0.5234
## F-statistic:  8072 on 3 and 22046 DF,  p-value: < 2.2e-16
```

```
pred <- predict(reg2, newdata = test_data)

test2<-test_data
test2$predictions<-pred
```

Our last model takes only Sales and Shipping Cost Variables

Model 3

```
reg3<-lm(Profit~Sales+`Shipping Cost`,data=train_data)
summary(reg3)
```

```
##
```

```
## Call:
## lm(formula = Profit ~ Sales + `Shipping Cost`, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -58.338  -6.261   0.880   5.884  48.048
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -1.027452   0.135534  -7.581 3.57e-14 ***
## Sales            0.227175   0.002336  97.231  < 2e-16 ***
## `Shipping Cost`  0.031290   0.015119   2.070   0.0385 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.92 on 22047 degrees of freedom
## Multiple R-squared:  0.465,  Adjusted R-squared:  0.4649
## F-statistic:  9580 on 2 and 22047 DF,  p-value: < 2.2e-16
```

```
pred <- predict(reg3, newdata = test_data)

test3<-test_data
test3$predictions<-pred
```

We now move to test our models for their performance

Model Selection

```
model_eval <- data.frame(
  model=character(),
  rms = numeric(),
  r_sq = numeric(),
  adj_r_sq = numeric()
)
#Model 0:
rmse <- sqrt(mean((test0$Profit - test0$predictions)^2))
r_squared <- summary(reg0)$r.squared
adjusted_r_squared <- summary(reg0)$adj.r.squared

model <- data.frame(
  model="Model 0",
  rms = rmse,
  r_sq = r_squared,
  adj_r_sq = adjusted_r_squared
)

model_eval<-rbind(model_eval,model)

#Model 1:

rmse <- sqrt(mean((test1$Profit - test1$predictions)^2))
```

```
r_squared <- summary(reg1)$r.squared
adjusted_r_squared <- summary(reg1)$adj.r.squared

model <- data.frame(
  model="Model 1",
  rms = rmse,
  r_sq = r_squared,
  adj_r_sq = adjusted_r_squared
)

model_eval<-rbind(model_eval,model)

#Model 2:

rmse <- sqrt(mean((test2$Profit - test2$predictions)^2))
r_squared <- summary(reg2)$r.squared
adjusted_r_squared <- summary(reg2)$adj.r.squared

model <- data.frame(
  model="Model 2",
  rms = rmse,
  r_sq = r_squared,
  adj_r_sq = adjusted_r_squared
)

model_eval<-rbind(model_eval,model)

#Model 3

rmse <- sqrt(mean((test3$Profit - test3$predictions)^2))
r_squared <- summary(reg3)$r.squared
adjusted_r_squared <- summary(reg3)$adj.r.squared

model <- data.frame(
  model="Model 3",
  rms = rmse,
  r_sq = r_squared,
  adj_r_sq = adjusted_r_squared
)

model_eval<-rbind(model_eval,model)

model_eval
```

```
##     model      rms      r_sq  adj_r_sq
## 1 Model 0 11.68632 0.5455971 0.5450813
## 2 Model 1 12.47960 0.4830909 0.4826920
## 3 Model 2 11.96603 0.5234589 0.5233941
## 4 Model 3 12.68925 0.4649612 0.4649127
```

We see that our preliminary Model works the best out of the 4 models we tested, mainly because of the R Squared Value for each model. Model 0 gives 55.01% R-Squared value.

Advanced Model Development

In real life, it is unlikely that a company would have sales and profit data of each market region before actually entering that market. In such cases, company usually have to build a model on regions it currently operates in, and then use that model to predict behaviour of other market. We would simulate one such scenario:

We assume that we only have sales data for EU and US market, and that we're trying to enter APAC market. We'll train our model using best features on EU and US market:

```
df_EUandUS<-subset(df, Market == "US" | Market == "EU")
df_APAC<-subset(df, Market == "APAC")

regNew<-lm(Profit~`Sub-Category`+Sales+`Shipping Cost`+Discount,data=df_EUandUS)
pred <- predict(regNew, newdata = df_APAC)

#test model accuracy
rmse <- sqrt(mean((df_APAC$Profit - pred)^2))
r_squared <- summary(regNew)$r.squared
print(paste("RMSE:", rmse))
```

```
## [1] "RMSE: 14.1962769437432"
```

```
print(paste("R Squared:", r_squared))
```

```
## [1] "R Squared: 0.564457366815323"
```

We see that RMSE for APAC data is 14.19. If we check the test data error when we initally built the model considering all the markets, it was 11.9. This shows that although the performance reduced when we tried to superimpose data from one market to other, it still performs reasonably with R Squared value - 56.44%

Results and Analysis

We have so far completed :

- EDA for our dataset
- Identified key features to model one variable - Profit
- Experimented with model trained on one category data and used it to predict data for other value of the category.

We identified the following most important features:

- Categorical Variables: Market, Product Sub-Category
- Numeric variables: Sales, Shipping Cost, Discount

We achieved best performance or R-Squared = 55.01% with Model 0.

Future Work

Our future work further develops on the model built till now. So far, we have considered different markets. We will be segmenting our analysis on Sub Category in next step. Analysing in detail effect of each sub category on profit and other numeric variables.

Also, taking into account feedback given to us by TA, the following steps would be carried out:

1. Model other numeric variables- Discount, Sales, Shipping Cost.
2. Treat Discount Variable as a categorical variable and carry out classification analysis
3. Treat other categorical variables as target and carry out classification modeling
4. Explore other datasets which provide other forms of cost - Revenue, Cost price, Selling Price etc so as to further refine our Profit Model
5. Treat removed outlier data separately