Ecommerce

Sally

7/19/2017

Online Retail

The "Online Retail" data set is about a gift set ecommerce website based in United Kingdom that has sold their merchandise all around the world via Internet. From the data set, each purchase of an item is recorded as one entry starting 12/1/2010 until 12/9/2011.

Problem

As its business growing, the company would like to find out which country is worth them to build a distribution center. Also, they would like to catergorize their top selling product for better inventory planning.

Download Dataset

Download dataset from the UCI Machine Learning Repository called **Online Retail**. Every entry variables include attribute as following: `Invoice number`, `Stock Code`, `Item Description`,`Quantity purchase`, `Invoice date`, `Unit Price`, `Customer ID` and `Country` which it sold to. The original data set has 541,909 rows, to use less data entry but still run a meaningful analysis; I have compared the each country's purchases units between all customer and only valid CustomerID purchased.

Load Package

Load the package that need for following analysis:

```
library(dplyr)
library(tidyr)
library(readxl)
library(httr)
library(ggplot2)
library(scales)
library(tidytext)
library(magrittr)
options(useFancyQuotes = FALSE)
options("scipen"=100, "digits"=4)
```

Load dataset

```
url <-"https://archive.ics.uci.edu/ml/machine-learning-databases/00352/Online%20Retail.xlsx"
GET(url, write_disk("Online%20Retail.xlsx", overwrite=TRUE))
```

```
## Response [https://archive.ics.uci.edu/ml/machine-learning-databases/00352/Online%20Retail.xlsx]
##   Date: 2017-07-20 04:17
##   Status: 200
##   Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet
```

```
##   Size: 23.7 MB
## <ON DISK>  Online%20Retail.xlsx
```

```
test <- read_excel("Online%20Retail.xlsx")
head(test, 3) #take a look at first 3 row
```

```
## # A tibble: 3 × 8
##   InvoiceNo StockCode                          Description Quantity
##       <chr>     <chr>                              <chr>    <dbl>
## 1    536365    85123A WHITE HANGING HEART T-LIGHT HOLDER        6
## 2    536365     71053                WHITE METAL LANTERN        6
## 3    536365    84406B     CREAM CUPID HEARTS COAT HANGER        8
## # ... with 4 more variables: InvoiceDate <dttm>, UnitPrice <dbl>,
## #   CustomerID <dbl>, Country <chr>
```

Clean dataset

First Step we have to check transform each variable's class

```
indx <- sapply(test, is.factor)
test[indx] <- lapply(test[indx], function(x) as.character(x))
```

if not numric, transform

```
numcolumn<-c("Quantity","UnitPrice","StockCode","CustomerID")
test[numcolumn] <- lapply(test[numcolumn], function(x) as.numeric(x))
```

```
## Warning in FUN(X[[i]], ...): NAs introduced by coercion
```

Take out Rows with NA but leave No Customer ID item by assigning *0* ID to it

```
test$CustomerID[is.na(test$CustomerID)]<-0
test<-subset(test,complete.cases(test))
```

Factor level assign to `Country` column

```
test$Country<-as.factor(test$Country)
```

Add column for total sales

```
test<-test%>%filter(UnitPrice>0 & Quantity>0)%>%
  mutate(Sales=Quantity*UnitPrice)
```

Assign Month

```
tmp<-as.Date(test$InvoiceDate,'%Y-%m-%d %H:%M:%S')
```

```
## Warning in as.POSIXlt.POSIXct(x, tz = tz): unknown timezone '%Y-%m-%d %H:
## %M:%S'
```

```
test$month<-format(tmp,'%m')
```

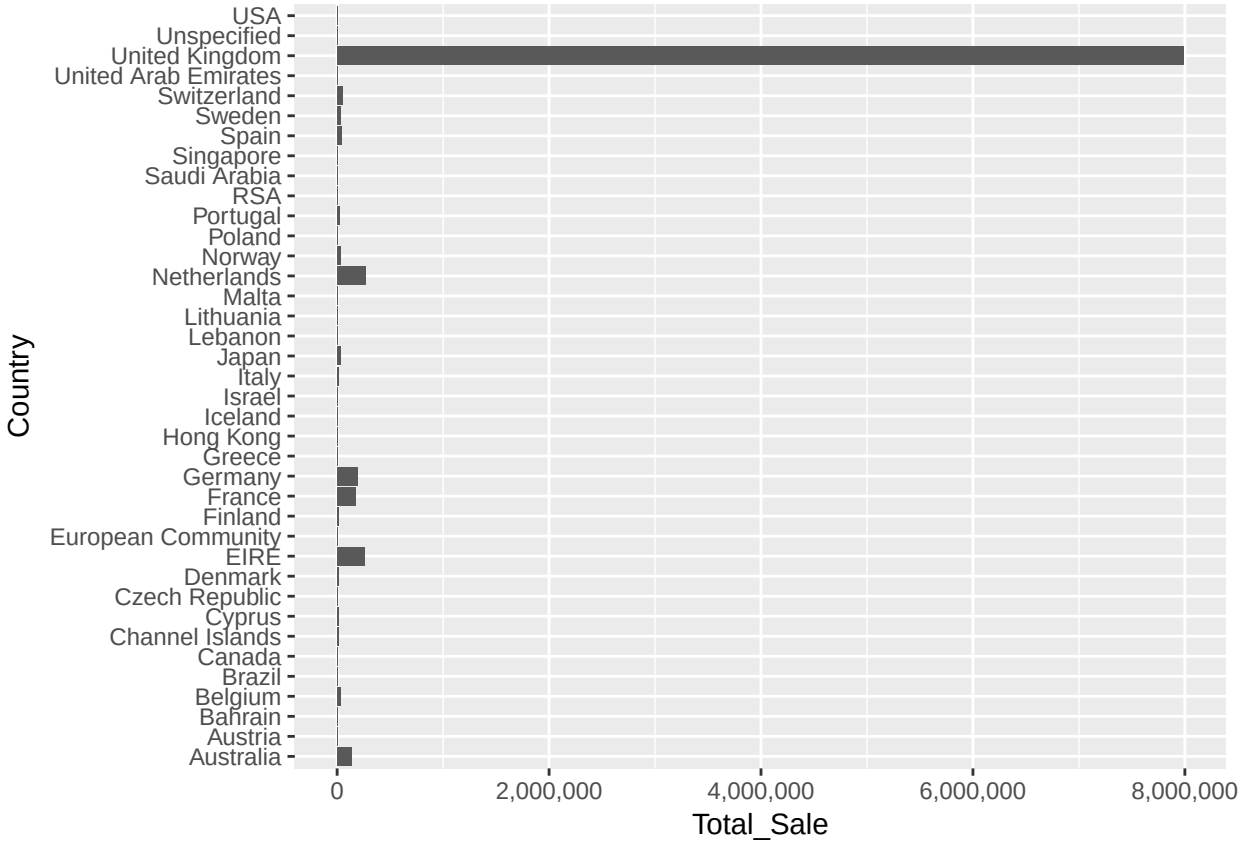
Analysis Data

Find best Selling Country

```
test%>%
group_by(Country)%>%
```

```
summarize(Total_Sale=sum(Sales))%>%
  ungroup()%>%
  arrange(desc(Total_Sale))
```

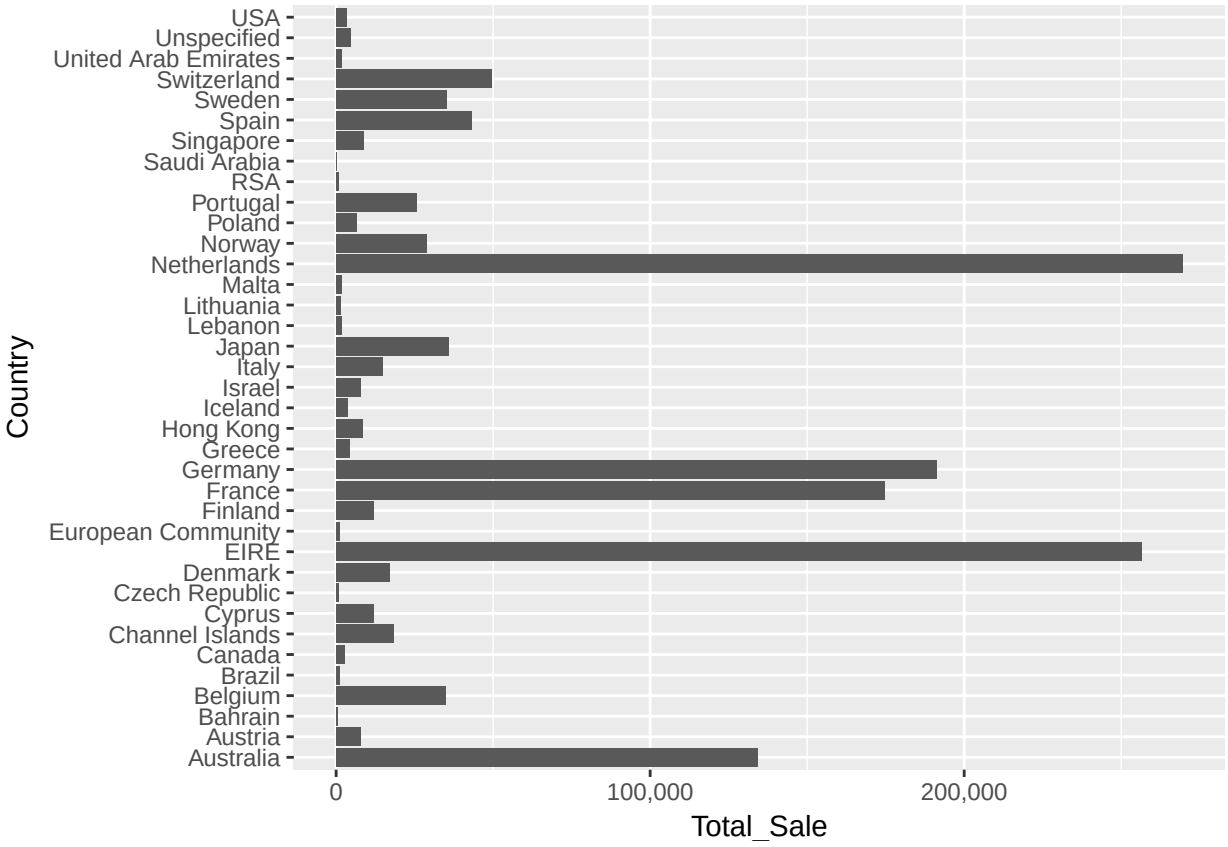
```
## # A tibble: 38 × 2
##           Country Total_Sale
##            <fctr>      <dbl>
## 1  United Kingdom    7989341
## 2     Netherlands     269617
## 3            EIRE     256569
## 4         Germany     191254
## 5          France     174731
## 6       Australia     134081
## 7     Switzerland      49519
## 8           Spain      43122
## 9           Japan      35975
## 10         Sweden      35128
## # ... with 28 more rows
```



Take out `United Kingdom` from the selling data to zoom in the sales in other countries

```
test%>%
group_by(Country)%>%
  filter(Country!="United Kingdom")%>%
  summarize(Total_Sale=sum(Sales))%>%
  ungroup()%>%
  arrange(desc(Total_Sale))
```

```
## # A tibble: 37 × 2
##        Country Total_Sale
##         <fctr>      <dbl>
## 1  Netherlands     269617
## 2         EIRE     256569
## 3      Germany     191254
## 4       France     174731
## 5    Australia     134081
## 6  Switzerland      49519
## 7        Spain      43122
## 8        Japan      35975
## 9       Sweden      35128
## 10     Belgium      34903
## # ... with 27 more rows
```



In order to zoom into the countries that has higher impact with our business, we take a look at all selling dollar and selling with customer ID without United Kingdom

```
#All
Sales_all<-test%>%
  group_by(Country)%>%
  filter(Country!="United Kingdom")%>%
  summarize(Total_Sale=sum(Sales))%>%
  ungroup()%>%
  arrange(desc(Total_Sale))

#Customer ID Only
Sales_member<-test%>%
```

```
  group_by(Country)%>%
  filter(Country!="United Kingdom")%>%
  filter(CustomerID>0)%>%
  summarize(Total_Sale=sum(Sales))%>%
  ungroup()%>%
  arrange(desc(Total_Sale))

Sales_member$Country[1:10] %in% Sales_all$Country[1:10]
```

```
##  [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

```
#if top10 ranking matched

Quant_all<-test%>%
  group_by(Country)%>%
  filter(Country!="United Kingdom")%>%
  summarize(Total_Quantity=sum(Quantity))%>%
  ungroup()%>%
  arrange(desc(Total_Quantity))

Quant_member<-test%>%
  group_by(Country)%>%
  filter(Country!="United Kingdom")%>%
  filter(CustomerID>0)%>%
  summarize(Total_Quantity=sum(Quantity))%>%
  ungroup()%>%
  arrange(desc(Total_Quantity))

Quant_member$Country[1:10] %in% Quant_all$Country[1:10]
```

```
##  [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

```
#if top10 Quant and Sales ranking matched
Quant_member$Country[1:10] %in% Sales_member$Country[1:10]
```

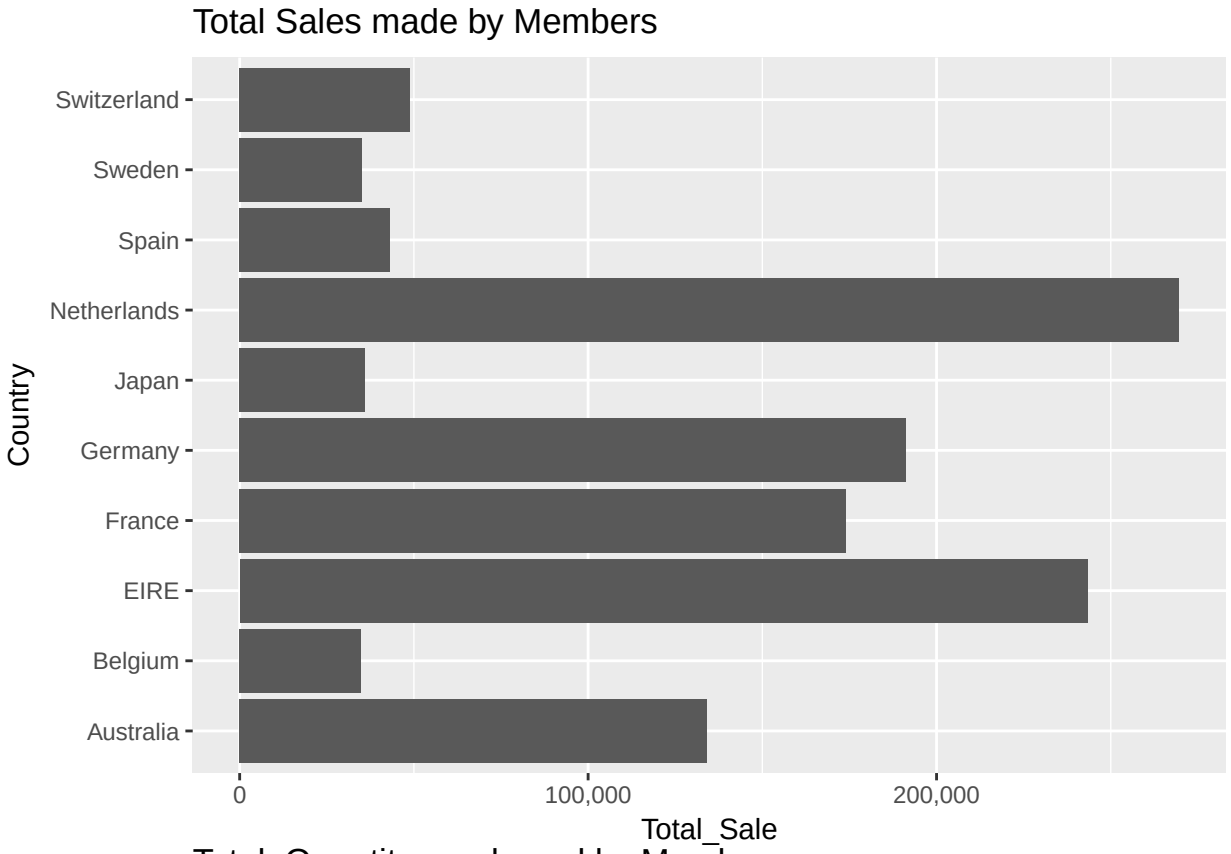
```
##  [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

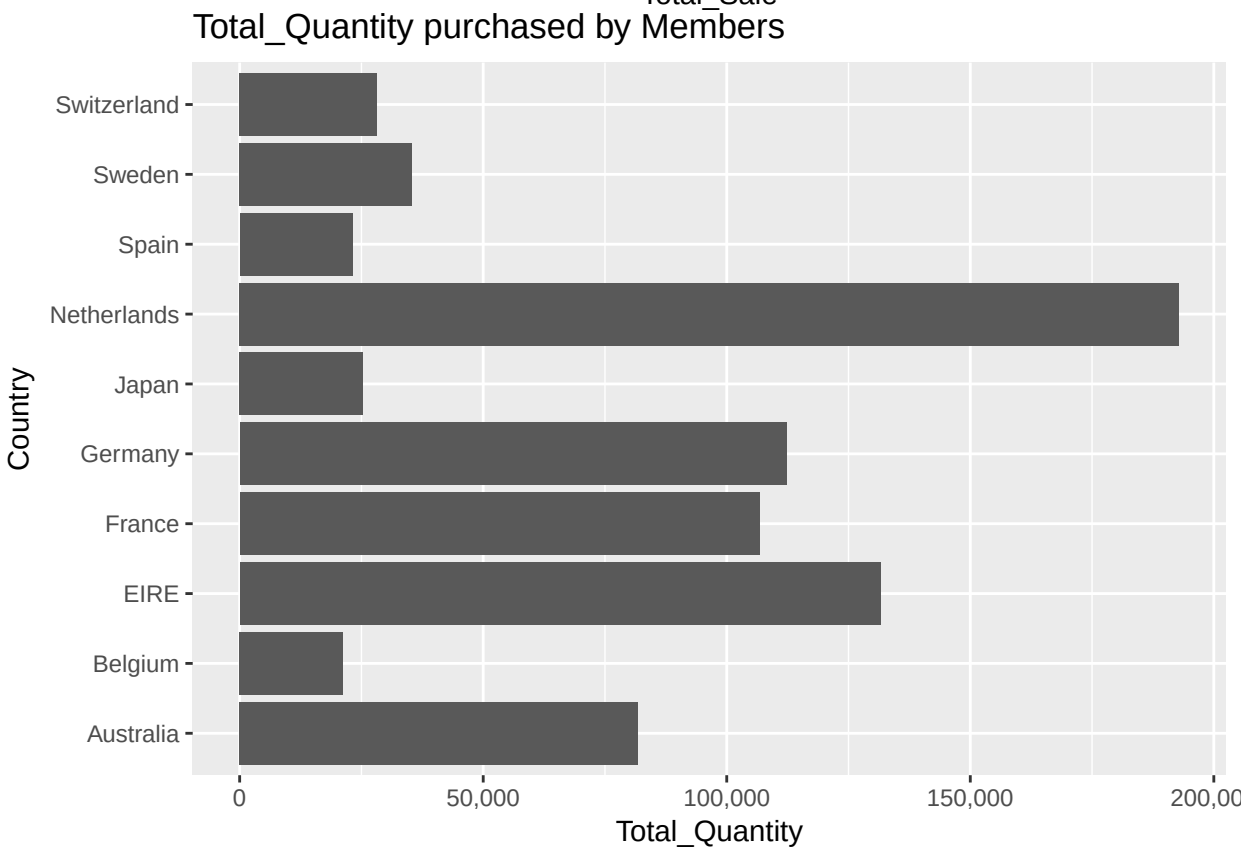
Therefore, we can built the distribution center at below places:

```
Top10<-Quant_all$Country[1:10]
Top10<-as.character(Top10)
Top10
```

```
##  [1] "Netherlands" "EIRE"        "Germany"     "France"      "Australia"
##  [6] "Sweden"      "Switzerland" "Japan"       "Spain"       "Belgium"
```

Also,we can see the top selling dollar and top selling quantity countrise are very identical, so we can zoom in into selling with only customer ID entry for further inventory analysis.





Pull the dataset as Customer ID only

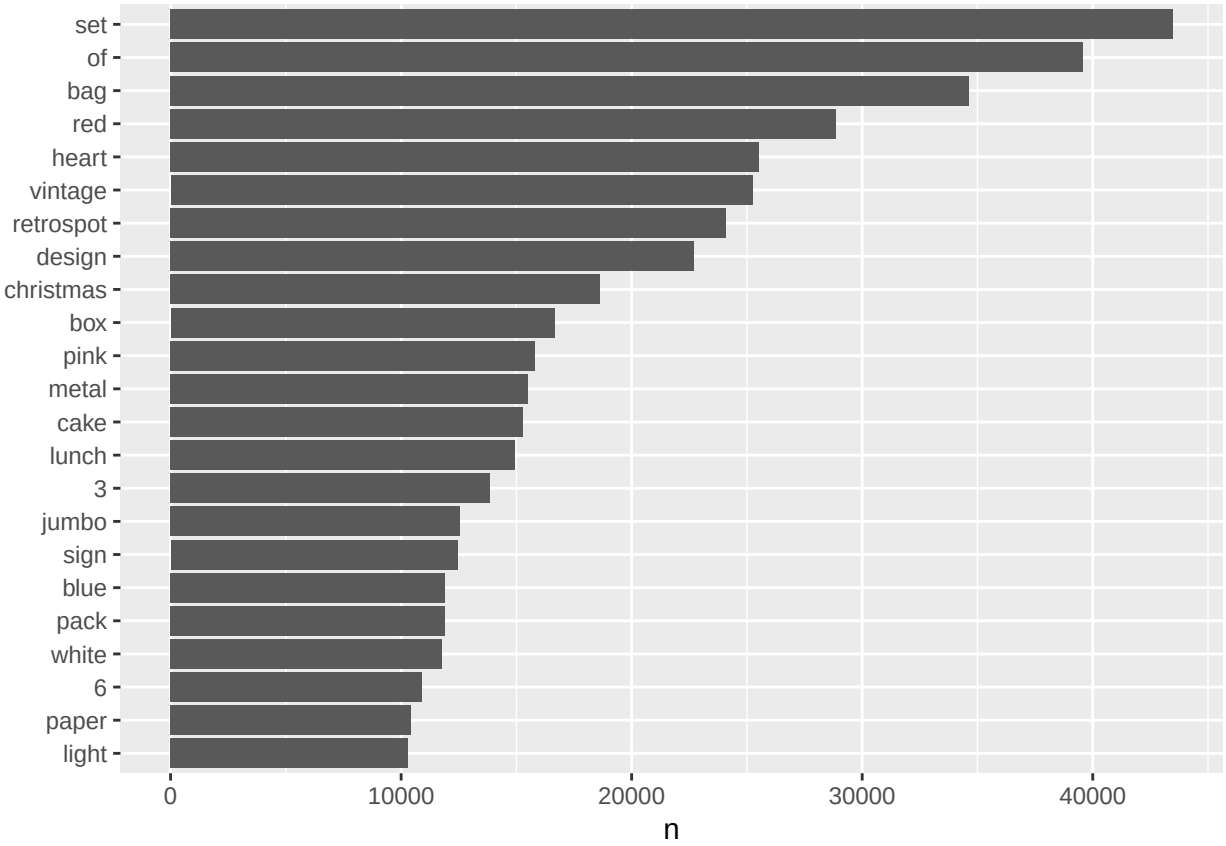
```
test<-filter(test,CustomerID>0)
```

Save back csv file

```
setwd("~/Desktop")
write.csv(test, file = "MyData.csv")
```

After found the top 10 countries that we can build distribution center, let's use the text mining to find inventory planning for Top 10 selling category items.

```
Des_df<-data_frame("Des"=test$Description,"Quant"=test$Quantity)
Des_tidy<-Des_df %>% unnest_tokens(word, Des)

Des_tidy%>%
  count(word, sort = TRUE)%>%filter(n > 10000) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(word, n))+
  geom_col()+xlab(NULL)+coord_flip()
```



From the bar chart, we can see set, bag and box are category name for top 10 repeated words in sales last year.

But, the most repeated word doesn't mean the best selling and profitable items. Therefore, we set up two dataframe, one is top selling quantity word and the other is top reapted words.

```
Des_count<-Des_tidy%>%filter(!grepl('the|of',word))%>%
  count(word, sort = TRUE)%>%
  mutate(word = reorder(word, n))%>%
  filter(n > 10000)
```

```
Des_count$word<-as.character(Des_count$word)
Des_Quant<-Des_tidy%>%group_by(word)%>%filter(!grepl('the|of',word))%>%
  summarize(Total_Quantity=sum(Quant))%>%filter(Total_Quantity>200000)
colnames(Des_count)[2] <- "repeat_times"
Des_comb<-merge(Des_count,Des_Quant,by="word",all=TRUE)
arrange(Des_comb,desc(Total_Quantity,repeat_times))
```

```
##         word repeat_times Total_Quantity
## 1        set        43482         472800
## 2        bag        34648         458403
## 3        red        28878         340762
## 4    vintage        25239         319689
## 5      heart        25531         309984
## 6  retrospot        24082         281567
## 7  christmas        18627         276250
## 8     design        22719         266098
## 9       pack        11904         263966
## 10      cake        15297         252998
## 11     paper        10441         229559
## 12     light        10308         224182
## 13     cases           NA         210550
## 14      pink        15817         208285
## 15         3        13861             NA
## 16         6        10907             NA
## 17      blue        11908             NA
## 18       box        16656             NA
## 19     jumbo        12541             NA
## 20     lunch        14940             NA
## 21     metal        15516             NA
## 22      sign        12444             NA
## 23     white        11757             NA
```

Now, we add a new column `Cat` and assigning category for populor items:

```
Top_item<-test
Top_item<-filter(Top_item,Top_item$Quantity>0)
Top_item$Cat<-ifelse(grepl("BAG",Top_item$Description),"BAG",
                     ifelse(grepl("BOX",Top_item$Description),"BOX",
                     ifelse(grepl("CHRISTMAS",Top_item$Description),"CHRISTMAS",
                     ifelse(grepl("CASES",Top_item$Description),"CASES",
                     ifelse(grepl("SET",Top_item$Description),"SET",
                     ifelse(grepl("BOTTLE",Top_item$Description),"BOTTLE","OTHER"))))))
```

However, we have to consider the set and box are perhaps a kind of items sold at bundle. In order to quicly take a look into our data, we can write a function called `finditem` for looking for best selling keyword item.

```
finditem<-function(x,y){
  x%>%filter(grepl(y,Description,ignore.case = TRUE))%>%group_by(Description,Cat)%>%
    summarize(Total_Quantity=sum(Quantity))%>%
    arrange(desc(Total_Quantity))
}
```

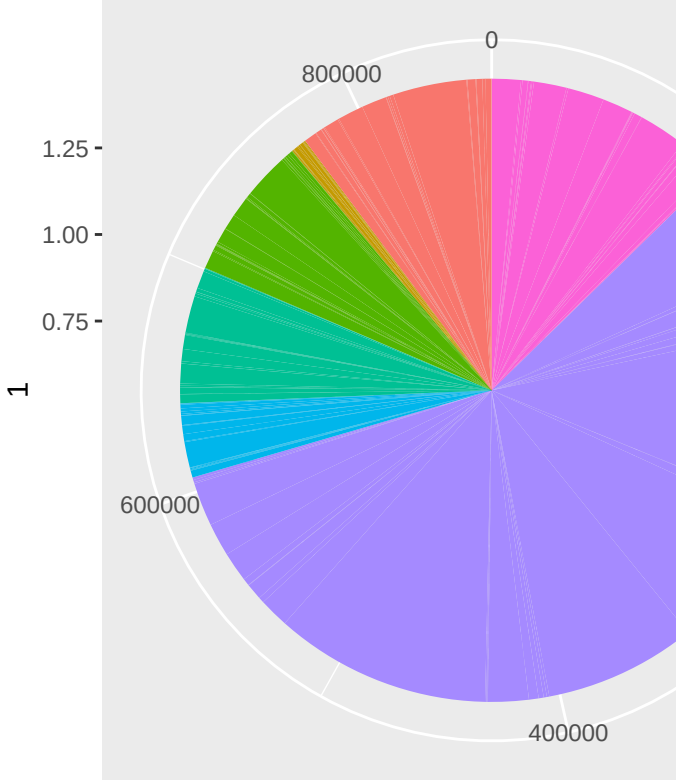
For example:

```
finditem(Top_item,"set")
```

```
## Source: local data frame [280 x 3]
```

```
## Groups: Description [280]
##
##                            Description       Cat Total_Quantity
##                                  <chr>     <chr>          <dbl>
## 1               MINI PAINT SET VINTAGE       SET          26076
## 2               JAM MAKING SET PRINTED       SET          15055
## 3            PLACE SETTING WHITE HEART       SET          14877
## 4   SET/20 RED RETROSPOT PAPER NAPKINS       SET          12313
## 5         SET OF 4 PANTRY JELLY MOULDS       SET          11792
## 6   SET OF 60 PANTRY DESIGN CAKE CASES     CASES          11675
## 7    SET OF 12 FAIRY CAKE BAKING CASES     CASES           8243
## 8             JAM MAKING SET WITH JARS       SET           8151
## 9  SET OF 20 VINTAGE CHRISTMAS NAPKINS CHRISTMAS           8122
## 10  ROUND SNACK BOXES SET OF4 WOODLAND       BOX           8100
## # ... with 270 more rows
```

So we can see there are sets of Jars, Napkins, Paints and Cases are sold in set are polpular.



Also, The pie chart below shows we have not assign even half of items

Take a look at items in *Other* category to find some other category that we might able to catch.

```
Top_item%>%
  group_by(Description)%>%
  filter(Cat=="OTHER")%>%
  summarize(Total_Quantity=sum(Quantity))%>%arrange(desc(Total_Quantity))
```

```
## # A tibble: 2,311 × 2
##                           Description Total_Quantity
##                                 <chr>          <dbl>
## 1        PAPER CRAFT , LITTLE BIRDIE          80995
```

```
## 2     MEDIUM CERAMIC TOP STORAGE JAR          77916
## 3  WORLD WAR 2 GLIDERS ASSTD DESIGNS          54415
## 4      ASSORTED COLOUR BIRD ORNAMENT          35362
## 5                    POPCORN HOLDER          30931
## 6                 RABBIT NIGHT LIGHT          27202
## 7          PACK OF 12 LONDON TISSUES          25345
## 8                BROCADE RING PURSE          22963
## 9    VICTORIAN GLASS HANGING T-LIGHT          22433
## 10         ASSORTED COLOURS SILK FAN          21876
## # ... with 2,301 more rows
```

```
finditem(Top_item,"tissues")
```

```
## Source: local data frame [18 x 3]
## Groups: Description [18]
##
##                          Description       Cat Total_Quantity
##                                <chr>     <chr>          <dbl>
## 1          PACK OF 12 LONDON TISSUES     OTHER          25345
## 2   PACK OF 12 HEARTS DESIGN TISSUES     OTHER           8569
## 3            PACK OF 12 SUKI TISSUES     OTHER           7346
## 4   PACK OF 12 RED RETROSPOT TISSUES     OTHER           6523
## 5        PACK OF 12 WOODLAND TISSUES     OTHER           5295
## 6           PACK OF 12 SKULL TISSUES     OTHER           5166
## 7   PACK OF 12 PINK POLKADOT TISSUES     OTHER           3985
## 8  PACK OF 12 50'S CHRISTMAS TISSUES CHRISTMAS           3419
## 9    PACK OF 12 PINK PAISLEY TISSUES     OTHER           3309
## 10       PACK OF 12 SPACEBOY TISSUES     OTHER           2957
## 11   PACK OF 12 BLUE PAISLEY TISSUES     OTHER           2223
## 12  PACK OF 12 VINTAGE DOILY TISSUES     OTHER           2087
## 13  PACK OF 12 CIRCUS PARADE TISSUES     OTHER           2024
## 14   PACK OF 12 PAISLEY PARK TISSUES     OTHER           1617
## 15   PACK OF 12 VINTAGE LEAF TISSUES     OTHER           1487
## 16     PACK OF 12 DOLLY GIRL TISSUES     OTHER           1203
## 17      PACK OF 12 RED APPLE TISSUES     OTHER           1175
## 18         PACK OF 12 DOILEY TISSUES     OTHER             72
```

```
finditem(Top_item,"light")
```

```
## Source: local data frame [123 x 3]
## Groups: Description [123]
##
##                             Description   Cat Total_Quantity
##                                   <chr> <chr>          <dbl>
## 1                   RABBIT NIGHT LIGHT OTHER          27202
## 2      VICTORIAN GLASS HANGING T-LIGHT OTHER          22433
## 3  COLOUR GLASS T-LIGHT HOLDER HANGING OTHER          15611
## 4         ANTIQUE SILVER T-LIGHT GLASS OTHER          12973
## 5        RED TOADSTOOL LED NIGHT LIGHT OTHER          12871
## 6     HANGING HEART JAR T-LIGHT HOLDER OTHER          10980
## 7       HANGING JAM JAR T-LIGHT HOLDER OTHER          10528
## 8   MULTI COLOUR SILVER T-LIGHT HOLDER OTHER          10279
## 9     AGED GLASS SILVER T-LIGHT HOLDER OTHER          10088
## 10                      CHILLI LIGHTS OTHER           9650
## # ... with 113 more rows
```

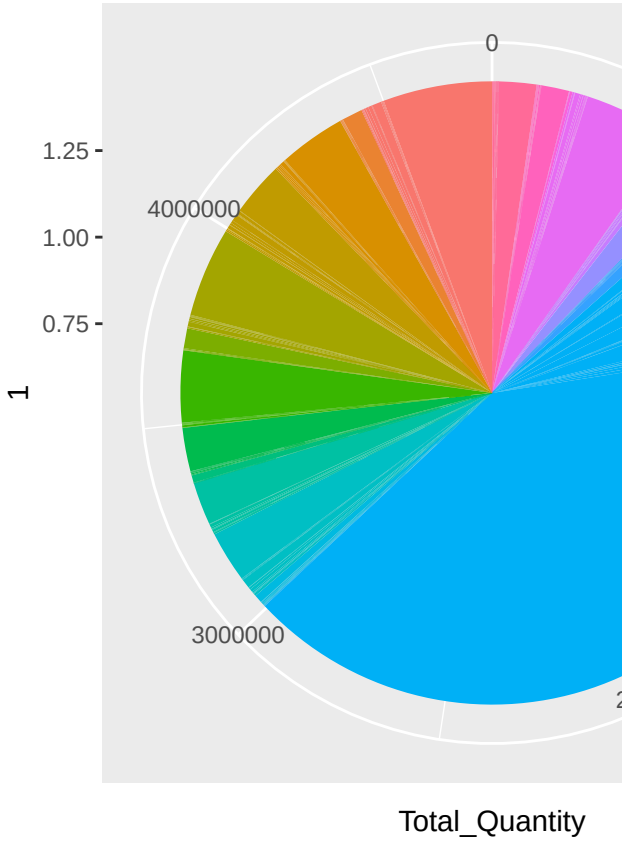
So, obviosly, “tissues” and “light” is another hit item that we can add into our category selection.

In order to add more category efficiently, I write a fuction **assignitem** to look into description of each item and assign the category name in the **Cat** column:

```
assignitem= function(x, patterns, replacements = patterns, fill = NA, ...)
{
  stopifnot(length(patterns) == length(replacements))

  ans = rep_len(as.character(fill), length(x))
  empty = seq_along(x)

  for(i in seq_along(patterns)) {
    greps = grepl(patterns[[i]], x[empty], ...)
    ans[empty[greps]] = replacements[[i]]
    empty = empty[!greps]
  }

  return(ans)
}
```

Now we get to see the new method of assigning method

```
Top_item$Cat<-assignitem(x = Top_item$Description, patterns = c("LUNCH","BAG","CHRISTMAS","PAINT","CASES
```



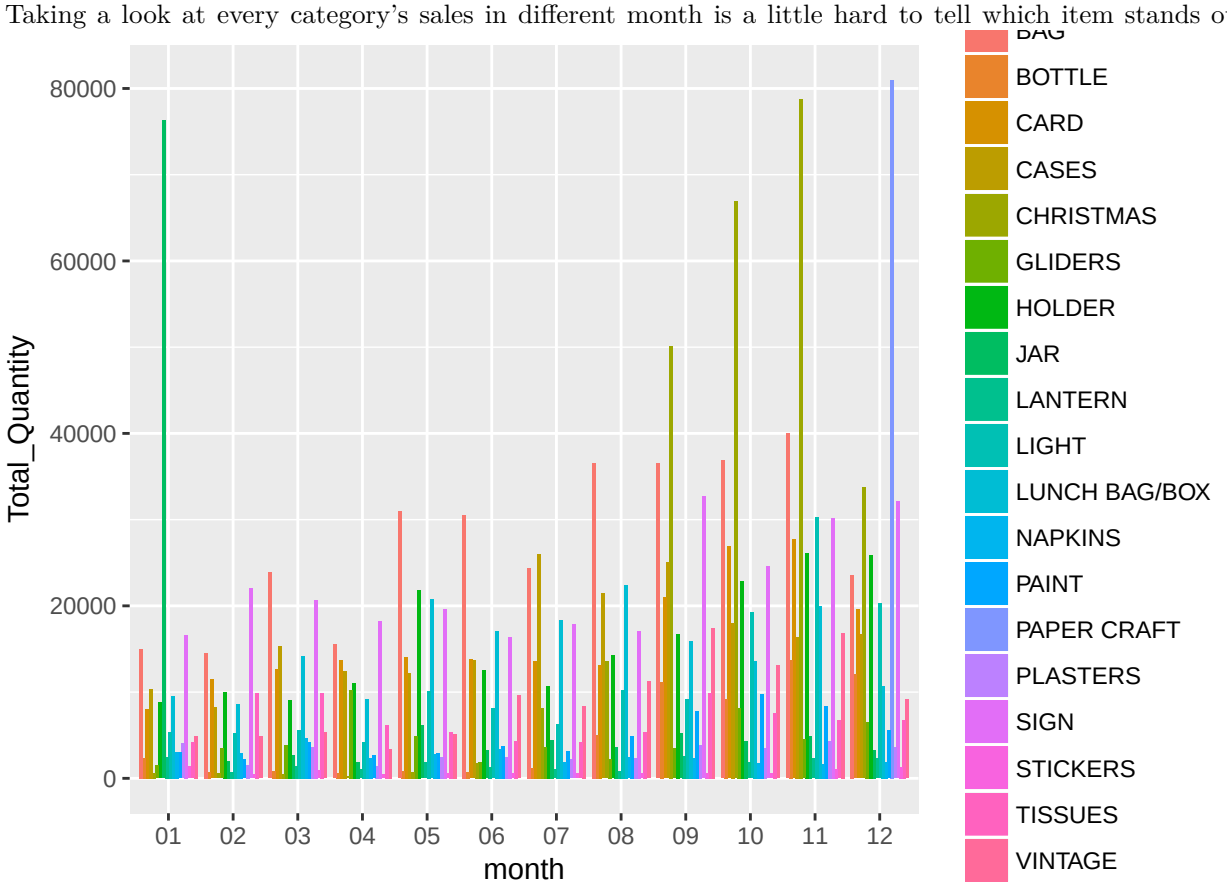
The pie chart below shows we have assign category to at least half of items

Finaly, filter out **Other** category for further analysis.

```
Top_item<-filter(Top_item,Cat!="OTHER")
```

Budget Allocation and Inventory Planning

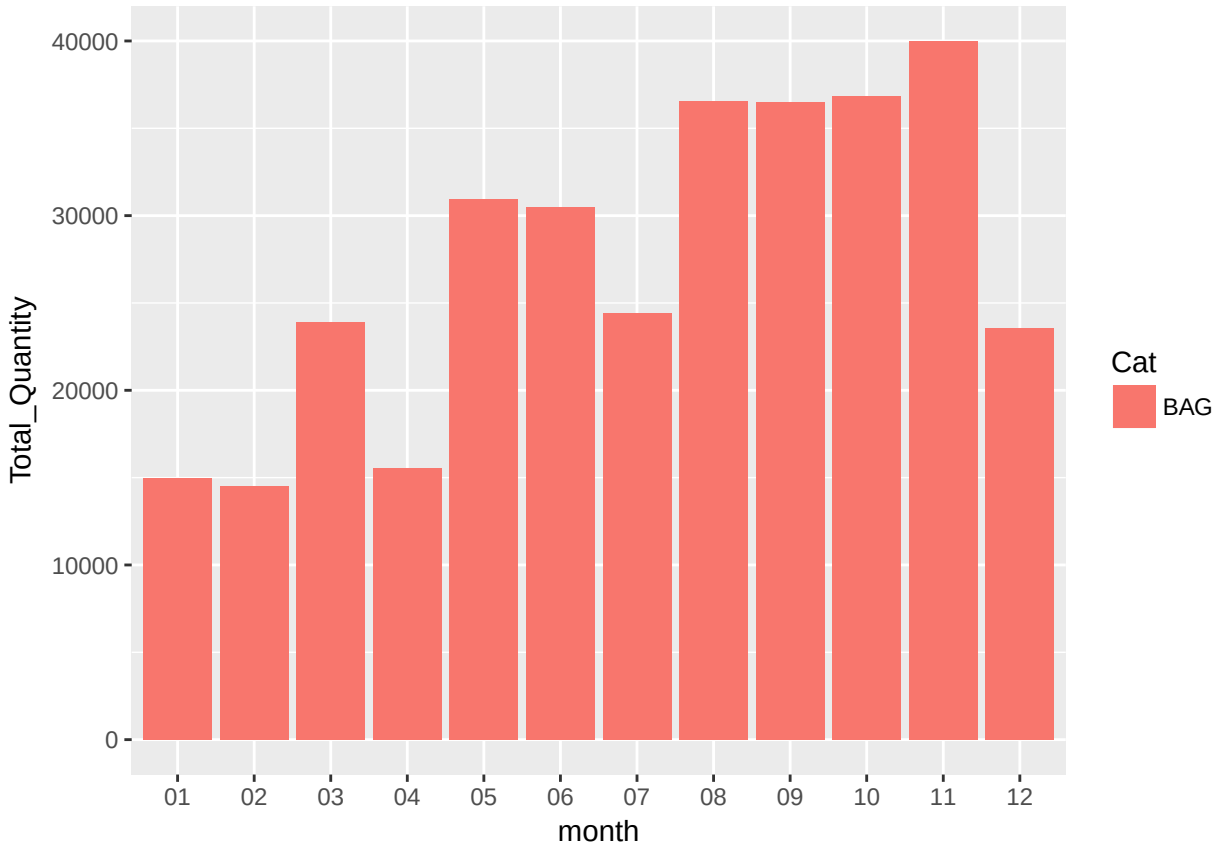
Taking a look at every category's sales in different month is a little hard to tell which item stands out.



We can see 'bag' has rather stable sales but `Christmas` item start pick up in September and reach the huge peak in November.

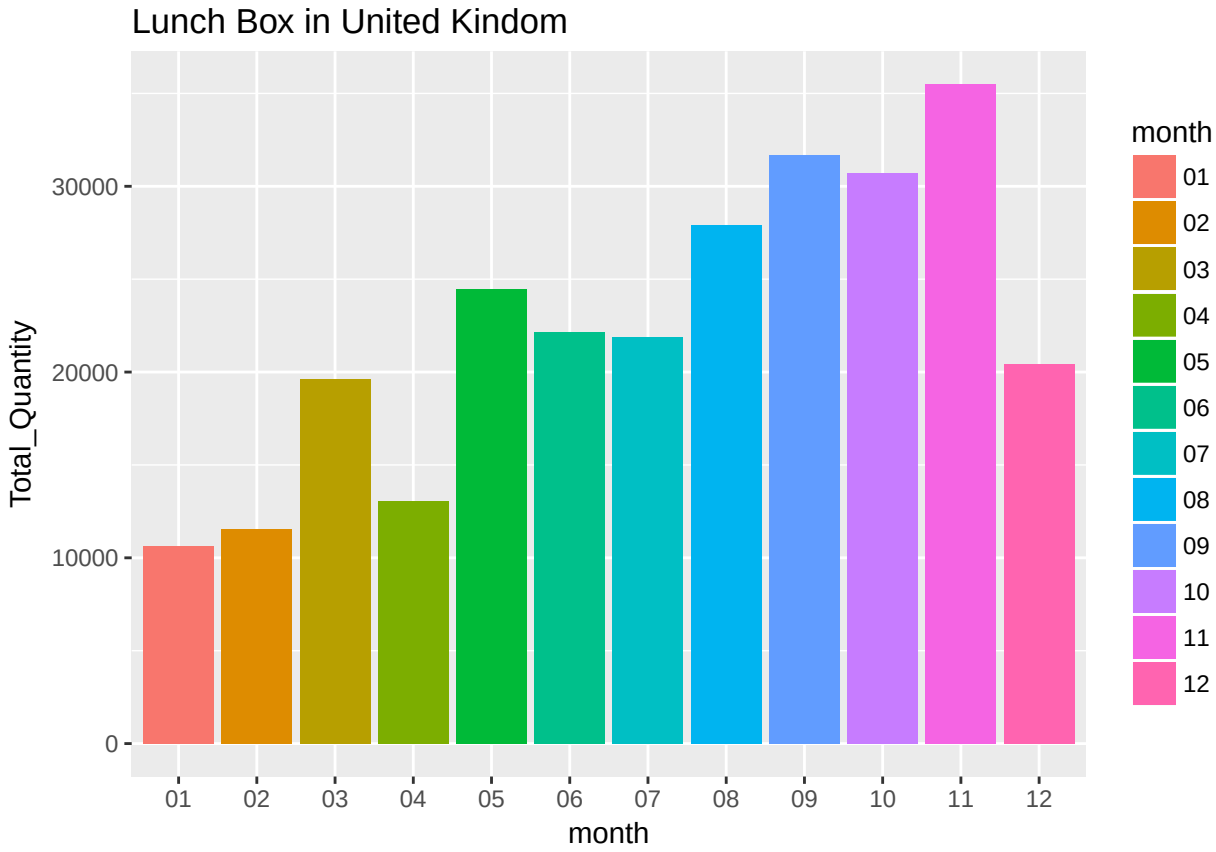
Also zoom in to `bag` in different months

```
Top_item%>%
  group_by(Cat,month)%>%
  filter(Cat=="BAG")%>%
  summarize(Total_Quantity=sum(Quantity))%>%
  ggplot(aes(x=month,y=Total_Quantity,group=Cat,fill=Cat))+geom_bar(stat = "identity",position = "dodge
```

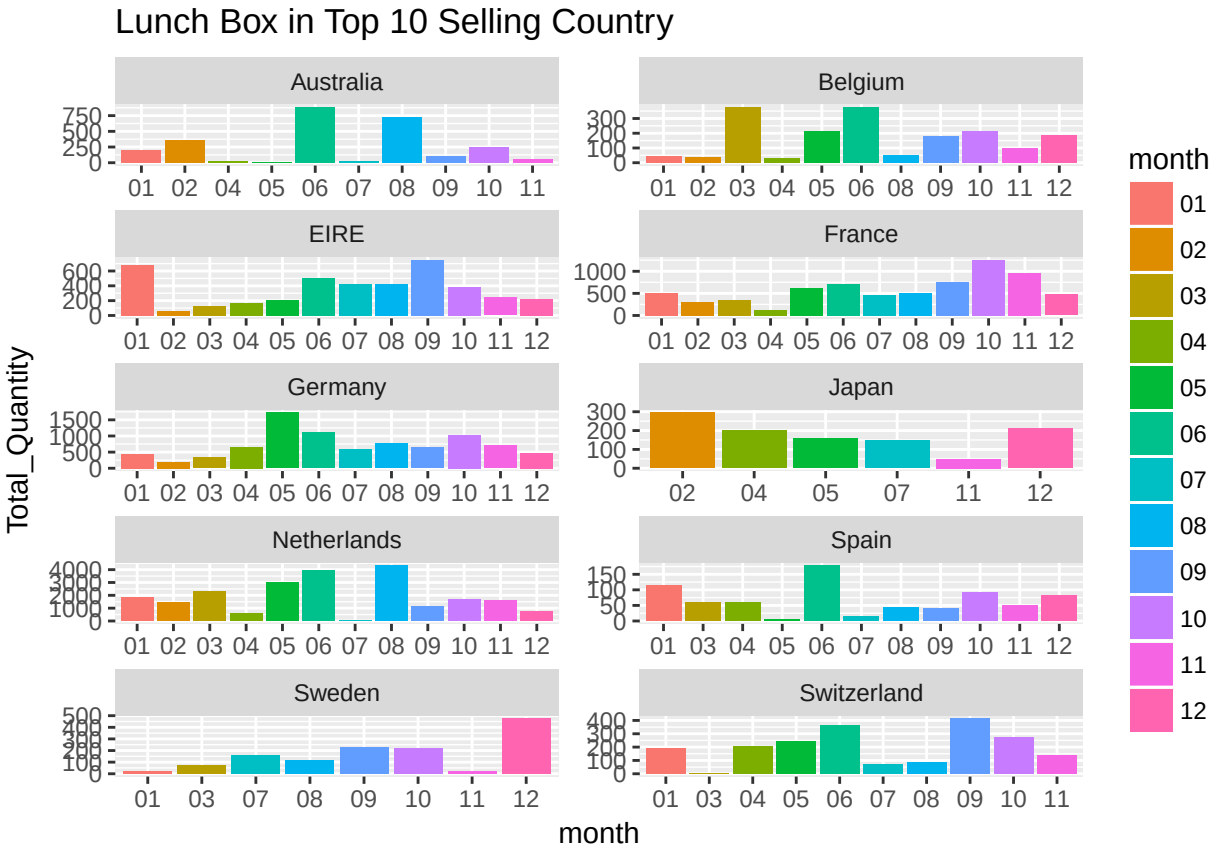


Take a look into lunch box category in United Kingdom

```
#United Kingdom
Top_item%>%
  group_by(month)%>%
  filter(Country=="United Kingdom")%>%
  filter(Cat=="BAG")%>%
  summarize(Total_Quantity=sum(Quantity))%>%
  ggplot(aes(x=month,y=Total_Quantity,fill=month))+geom_bar(stat = "identity",position = "dodge")+ggtitl
```




```
#Other Countries we are looking to build distribution center
  Top_item%>%
  group_by(Country,month)%>%
  filter(Country%in%Top10)%>%
  filter(Cat=="BAG")%>%
  summarize(Total_Quantity=sum(Quantity))%>%
  ggplot(aes(x=month,y=Total_Quantity,fill=month))+geom_bar(stat = "identity",position = "dodge")+
  facet_wrap(~Country,ncol=2,scales="free")+ggtitle("Lunch Box in Top 10 Selling Country")
```



From the above analysis, it gives a discriptive data plot for us to see the business in different country and help us understand how can we manage it, such as building distribution center, inventory planning and budget allocation. The data analytic and studies provide business insight that help the company shape their future.