Wells\_Fargo\_Challenge

Siddarth

06 October 2017

Table of Contents

[Introduction 2](#_Toc495399804)

[Read Data 2](#_Toc495399805)

[Installing Packages 3](#_Toc495399806)

[Data Exploration 3](#_Toc495399807)

[Analyzing Web Traffic 5](#_Toc495399808)

[Frequency of Visits 5](#_Toc495399809)

[Age Groups 6](#_Toc495399810)

[Frequency of pages visited by age group "Old" 8](#_Toc495399811)

[Frequency of pages visited by age group "Middle age" 8](#_Toc495399812)

[Frequency of pages visited by age group "Young" 9](#_Toc495399813)

[Conclusions & Recommendations 10](#_Toc495399814)

[Notes 10](#_Toc495399815)

[Analyzing Credit Card transactions 11](#_Toc495399816)

[Splitting given Descriptions 11](#_Toc495399817)

[Frequency of Transactions 12](#_Toc495399818)

[Amount spent per transaction 13](#_Toc495399819)

[Conclusions and Recommendations 14](#_Toc495399820)

[Analyzing Daily interactions 14](#_Toc495399821)

[Stop Word Removal 16](#_Toc495399822)

[Tokenization & Stemming 16](#_Toc495399823)

[Word Cloud 17](#_Toc495399824)

# 

# Introduction

The following code is written for submission to Wells Fargo analytics challenge. The data given belongs to 4 major categories.  
1. Month end balances of customers  
2. Daily uses of WF credit card  
3. Daily WF web traffic  
4. Daily Interactions with WF

The objective is to analyze the data and come up with insights that might increase profitability, reduce expenses, improve customer services or help in target marketing of customers. The graphs are included within the report itself.

## Read Data

The first step is to set the working directory and load the data.The data given can be saved as RData and loaded from the working directory to reduce the execution time.The commented code below shows how to convert the data into RData.  
 Please uncomment and execute the code to load as RData after downloading the data set from [<https://s3.amazonaws.com/mindsumo/public/Fake+Data+and+Metadata+-+Final+no+pass.xlsx>] and changing the name of the file to "Dataset". After executing the commented code, please comment it again to avoid any errors.

rm(list=ls(all=T))  
  
#Set your working directory  
setwd("C:/Users/sid/Desktop/Wells\_fargo")  
  
#install.packages("xlsx")  
#library(xlsx)  
#credit\_card=read.xlsx(file = "Dataset.xlsx",sheetName = "credit\_card")  
#web\_traffic=read.xlsx(file = "Dataset.xlsx",sheetName = "web\_traffic")  
#month\_end\_balances=read.xlsx(file ="Dataset.xlsx",sheetName = "sheet1" )   
#daily\_interactions=read.xlsx(file ="Dataset.xlsx",sheetName = "interactions" )  
# save(credit\_card,file = "credit\_card.RData")  
# save(web\_traffic,file = "web\_data.RData")  
# save(month\_end\_balances,file = "balances.RData")  
# save(daily\_interactions,file = "interactions.RData")  
  
load("balances.RData")  
load("interactions.RData")  
load("credit\_card.RData")  
load("web\_data.RData")

## Installing Packages

The install.packages command is commented in the code.Whenever required please uncomment the command and execute the line to install the libraries necessary.

## Data Exploration

We start with exploring the data given to us. We check for information such as the number of unique customers whose data is given and if any of the data given to us has blank values.We then check and understand the attributes given to us based on the metadata and also the structure of the data. We convert the attributes to appropriate data types such as numeric,character or factor.The code for pre-processing and converting to appropriate data types is given below:

length(unique(month\_end\_balances$masked\_id))

## [1] 50

length(unique(daily\_interactions$masked\_id))

## [1] 50

length(unique(credit\_card$masked\_id))

## [1] 26

length(unique(web\_traffic$masked\_id))

## [1] 39

str(month\_end\_balances)

## 'data.frame': 300 obs. of 25 variables:  
## $ masked\_id : num 12 12 12 12 12 12 33 33 33 33 ...  
## $ asof\_yyyymm : num 201612 201611 201610 201609 201608 ...  
## $ age : num 90 90 90 89 89 89 54 54 54 54 ...  
## $ tenure\_altered : num 43 42.9 42.8 42.7 42.7 ...  
## $ checking\_acct\_ct : num 2 2 2 2 2 2 4 4 4 4 ...  
## $ savings\_acct\_ct : num 2 2 2 2 2 2 2 2 2 2 ...  
## $ mortgage\_flag : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ heloc\_flag : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ personal\_loan\_flag : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ cc\_flag : num 1 1 1 1 1 1 0 0 0 0 ...  
## $ prot\_acct\_flag : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ check\_bal\_altered : num 18472 10294 9065 13502 7271 ...  
## $ sav\_bal\_altered : num 104300 107873 101799 98369 360590 ...  
## $ mortgage\_bal\_altered : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ heloc\_bal\_altered : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ personal\_loan\_bal\_altered: num 0 0 0 0 0 0 0 0 0 0 ...  
## $ atm\_withdrawls\_cnt : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ atm\_deposits\_cnt : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ branch\_visit\_cnt : num 13 11 13 19 12 14 7 12 14 11 ...  
## $ phone\_banker\_cnt : num 0 0 0 0 0 0 0 4 2 0 ...  
## $ mobile\_bank\_cnt : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ online\_bank\_cnt : num 0 1 0 1 0 0 8 11 12 10 ...  
## $ direct\_mail\_cnt : num 0 0 0 0 0 0 0 0 1 1 ...  
## $ direct\_email\_cnt : num 0 1 0 1 0 0 10 28 15 3 ...  
## $ direct\_phone\_cnt : num 0 3 3 5 2 0 0 2 3 1 ...

month\_end\_balances[month\_end\_balances==" "]=NA  
daily\_interactions[daily\_interactions==" "]=NA  
credit\_card[credit\_card==" "]=NA  
web\_traffic[web\_traffic==" "]=NA  
  
sum(is.na(month\_end\_balances))

## [1] 0

sum(is.na(daily\_interactions))

## [1] 0

sum(is.na(credit\_card))

## [1] 0

sum(is.na(web\_traffic))

## [1] 0

num\_atr1=c("age","tenure\_altered","checking\_acct\_ct","savings\_acct\_ct",  
"check\_bal\_altered","sav\_bal\_altered","mortgage\_bal\_altered",  
"heloc\_bal\_altered","personal\_loan\_bal\_altered",  
"atm\_withdrawls\_cnt","atm\_deposits\_cnt","branch\_visit\_cnt",  
"phone\_banker\_cnt","mobile\_bank\_cnt",  
"online\_bank\_cnt","direct\_mail\_cnt","direct\_email\_cnt","direct\_phone\_cnt","masked\_id")  
  
  
categ\_atr1=c("mortgage\_flag","heloc\_flag","personal\_loan\_flag","cc\_flag","prot\_acct\_flag")  
  
month\_end\_balances[num\_atr1]=data.frame(sapply(month\_end\_balances  
 [num\_atr1], as.numeric))  
  
month\_end\_balances[categ\_atr1]=data.frame(sapply(month\_end\_balances  
 [categ\_atr1], as.factor))  
  
web\_traffic$wf\_page=as.character(web\_traffic$wf\_page)

# Analyzing Web Traffic

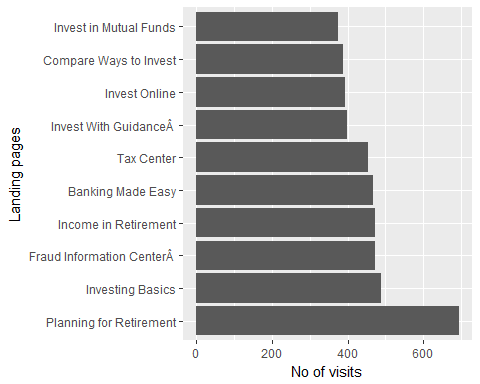
After understanding the structure of the data given to us, we first start with web traffic data. We have the list of web pages each customer visited.With this information we can figure out the purpose of the customer visit to the website.We assume that the final page the customer landed is the information the customer is looking for. The intution for this approach is that if we are able to find the most frequent pages a particular customer visited and and match it to the customer information given such as age, it might reveal some insights.

#install.packages(c("dplyr","tibble","tidyr","ggplot2"))  
library(dplyr)  
library(tibble)  
library(tidyr)  
library(ggplot2)  
  
temp\_data=web\_traffic%>%as\_data\_frame()%>%  
 separate(wf\_page,into=c("page\_1","page\_2","page\_3"),sep="/")  
  
temp\_data[temp\_data==" " | temp\_data==""]=NA

## Frequency of Visits

**Fig 1** shows the most frequent pages i.e., top 10 visited by all the customers.

temp\_data%>%mutate(landing\_page=(ifelse(is.na(page\_3),page\_2,page\_3)))%>%  
 select(masked\_id,landing\_page)%>%group\_by(landing\_page)%>%summarise(ct=n())%>%  
 top\_n(10,wt=ct)%>%ggplot(aes(reorder(landing\_page,-ct,FUN=median),ct))+  
 geom\_col()+labs(y="No of visits",x="Landing pages")+coord\_flip()



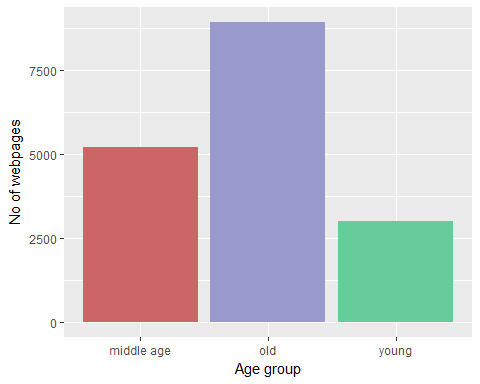
**Fig 1:** Frequency of visits Vs Landing Pages

## Age Groups

We select age groups as one of the factor to get further insights into customer behavior. For this purpose, we try to categorize the customers into different age groups and find out the most frequent pages visited by the age group. We can also use other factors to bin the customer based on the domain knowledge.  
 For this purpose we combine customer data and web traffic data based on customer id assuming the age of the customer as the maximum of the age attribute for that customer. We then bin the customers into age groups as following:  
 1. **"Old" if age>50**  
 2. **"Middle age" if 50>age>=30**  
 3. **"Young" if >30age>=14**  
 After binning we find out the information that each age group is looking for based on the most frequent web page visits.

**Fig 2** shows the number of times each group visited the website for information.

#Calculating age of each customer.Assuming max age of the given groups   
cust\_ages=data.frame(masked\_id=numeric(),age=numeric())  
  
for(i in unique(month\_end\_balances$masked\_id)){  
 x=subset(month\_end\_balances,masked\_id==i)  
 y=data.frame('masked\_id'=i,'age'=max(x$age))  
 cust\_ages=rbind(cust\_ages,y)  
}  
rm(x,y)  
  
cust\_ages=as\_tibble(cust\_ages)  
  
month\_end\_balances=as\_tibble(month\_end\_balances)  
  
#Binning.Ages 50 and above as old.Ages 30 -<50 middle age 14-<30 young  
  
x=temp\_data%>%mutate(landing\_page=(ifelse(is.na(page\_3),page\_2,page\_3)))%>%  
 select(masked\_id,landing\_page)%>%group\_by(masked\_id,landing\_page)%>%  
 summarise(ct=n())%>%full\_join(cust\_ages)%>%  
 mutate(set=ifelse(age>50,"old",ifelse(age>30,"middle age","young")))%>%select(set,landing\_page,ct)%>%  
 group\_by(set,landing\_page)%>%summarise(ct=sum(ct))%>%top\_n(10,ct)  
  
temp\_data%>%mutate(landing\_page=(ifelse(is.na(page\_3),page\_2,page\_3)))%>%  
 full\_join(cust\_ages)%>%mutate(set=ifelse(age>50,"old",ifelse(age>30,"middle age","young")))%>%group\_by(set)%>%summarise(ct=n())%>%ggplot(aes(set,ct))+  
 geom\_col(fill=c("#CC6666", "#9999CC", "#66CC99"))+labs(x="Age group",y="No of webpages")

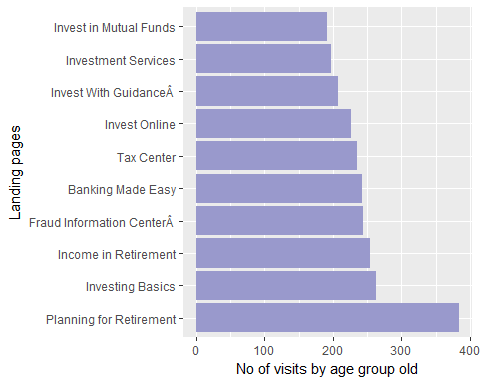


**Fig 2:** Age Groups vs Website visits

## Frequency of pages visited by age group "Old"

**Fig 3** shows the most frequent information the age group **Old** is looking for.

filter(x,set=="old")%>%ggplot(aes(reorder(landing\_page,-ct,FUN=median),ct))+  
 geom\_bar(stat="identity",fill="#9999CC")+  
 labs(y="No of visits by age group old",x="Landing pages")+coord\_flip()

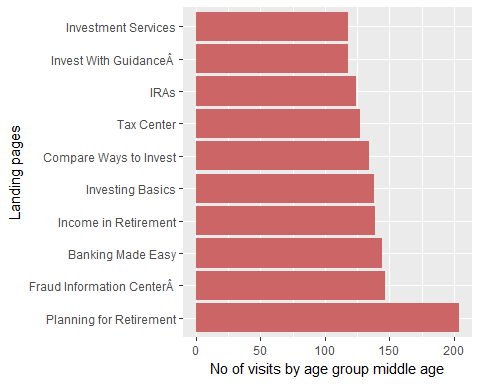


**Fig 3**: No of visits by group Old

## Frequency of pages visited by age group "Middle age"

**Fig 4** shows the most frequent information the age group **Middle age** is looking for.

filter(x,set=="middle age")%>%ggplot(aes(reorder(landing\_page,-ct,FUN=median),ct))+  
 geom\_bar(stat="identity",fill= "#CC6666")+  
 labs(y="No of visits by age group middle age",x="Landing pages")+coord\_flip()

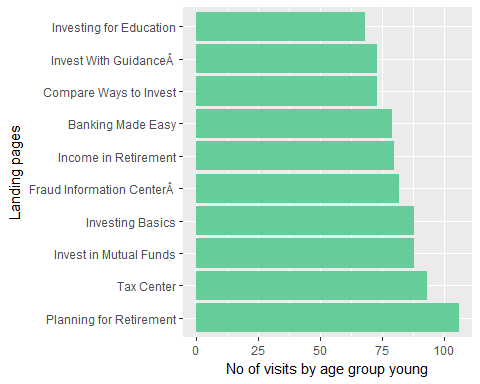


**Fig 4**: No of visits by group Middle age

## Frequency of pages visited by age group "Young"

**Fig 5** shows the most frequent information the age group **Young** is looking for.

filter(x,set=="young")%>%ggplot(aes(reorder(landing\_page,-ct,FUN=median),ct))+  
 geom\_bar(stat="identity",fill="#66CC99")+  
 labs(y="No of visits by age group young",x="Landing pages")+coord\_flip()



**Fig 5**: No of visits by group Young

## Conclusions & Recommendations

From Fig *3,4,5* we can conclude the following:

1. **Planning for Retirement** is the most common information customers are looking for.WF can plan more comprehensive product offering in the retirement space to attract more customers.
2. **Investment** is the other most prominent category among the customers. The bank should be proactive in offering the customers several investment options. The bank should try to promote its investment offerings.
3. **Fraud Information Centre** is also most frequently searched information for all the age groups.The bank should improve their security measures as more customers are looking for fraud information center.

## Notes

More amount of customer data is needed to further develop the model. As the data is limited to only 50 customers, this model does not give complete confidence for investments into advertising or product development. However, more data can further validate our insights. We can also group the data based on different factors.

# Analyzing Credit Card transactions

We have the credit card transaction information for 50 customers. The detailed description of each category and the amount spent in each transaction. We will now try to figure out insights such as the category which has the maximum number of transactions, category which has the highest amount spent per transaction.

For our purpose we split the credit card descriptions 2 and 3 to obtain the final descriptions. Further, as we see that Description 2 doesn't give the required number of transactions to mark it as prominent we use Description 3.

## Splitting given Descriptions

cc\_temp=credit\_card  
cc\_temp$Des1=as.character(cc\_temp$Des1)  
cc\_temp$Des2=as.character(cc\_temp$Des2)  
cc\_temp$Des3=as.character(cc\_temp$Des3)  
  
x=cc\_temp%>%as\_data\_frame()%>%separate(Des2,into=c("Des2\_1","Des2\_2"),sep="/")%>%  
 separate(Des3,into=c("Des3\_1","Des3\_2","Des3\_3"),sep=",")%>%  
 mutate(fin\_Des2=ifelse(is.na(Des2\_2),Des2\_1,Des2\_2))%>%  
 mutate(fin\_Des3=ifelse(is.na(Des3\_3),ifelse(is.na(Des3\_2),Des3\_1,Des3\_2),Des3\_3))  
  
x%>%group\_by(fin\_Des3)%>%summarise(count=n())%>%arrange(desc(count))%>%  
 top\_n(10,count)

## # A tibble: 13 x 2  
## fin\_Des3 count  
## <chr> <int>  
## 1 TAX PAYMENTS 51  
## 2 DELIVERY SERVICES (BMG INFERRED DEFINITION) 34  
## 3 POSTAL SERVICES - GOVERNMENT ONLY 27  
## 4 OR MISCELLANEOUS ADJUSTMENT (BMG INFERRED DEFINITION) 25  
## 5 FLORISTS 25  
## 6 AND BLUEPRINTING SERVICES 24  
## 7 BAKERY PRODUCTS 23  
## 8 GOVERNMENT SERVICES (NOT ELSEWHERE CLASSIFIED) 23  
## 9 MISCELLANEOUS STORES (BMG INFERRED DEFINITION) 23  
## 10 RENTALS 22  
## 11 U.S. GOVERNMENT U.S. FEDERAL GOVERNMENT AGENCIES OR DEPARTMENTS 22  
## 12 UK SUPERMARKETS HOT FILE 22  
## 13 WRECKING AND SALVAGE YARDS 22

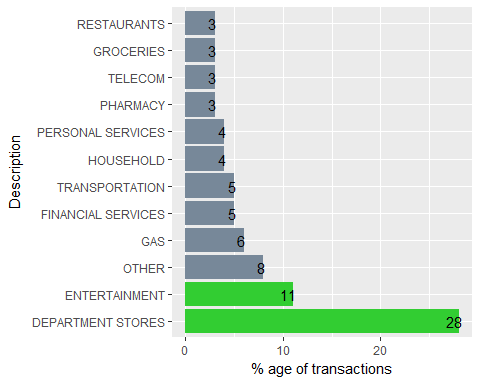
## 

## Frequency of Transactions

credit\_card%>%as\_data\_frame()%>%summarise(total\_no\_transactions=n())

## # A tibble: 1 x 1  
## total\_no\_transactions  
## <int>  
## 1 3468

x%>%group\_by(fin\_Des2)%>%summarise(count=n())%>%arrange(desc(count))%>%  
 mutate(percent=round(count\*100/3468))%>%top\_n(10,percent)%>%  
 ggplot(aes(reorder(fin\_Des2,-percent,FUN=min),percent))+  
 geom\_bar(stat="identity",fill=c(rep("limegreen",2),rep("lightslategray",10)))+  
 labs(x="Description",y="% age of transactions")+geom\_text(aes(label=percent,hjust=0.8)) +  
 coord\_flip()



**Fig 6**: % age of transactions Vs Category

**Fig 6** shows the category and the percentage of transactions in each category.We can observe that **Department stores** and **Entertainment** categories account for atleast **40%** of the transactions.

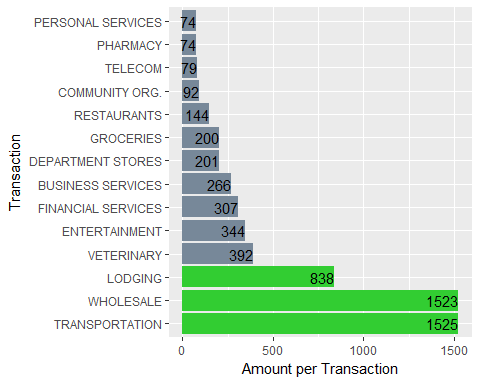
## Amount spent per transaction

We can also calculate the top 10 categories where the customers spent most amount per transaction. This gives us the information about the categories that WF can target and increase the discount offers to simulate furthur demand. The assumption here is that WF becomes more profitable with increase in transactions. Here,**Education** is removed as it has the highest amount per transaction of *$4908*.

cc\_amount=x%>%group\_by(fin\_Des2)%>%summarise(total\_amount=sum(Payment))  
  
cc\_numtransact=x%>%group\_by(fin\_Des2)%>%summarise(num\_transact=n())  
  
z=inner\_join(cc\_numtransact,cc\_amount,by="fin\_Des2")%>%  
 mutate(amtpertransact=round(total\_amount/num\_transact))%>%  
 arrange(desc(amtpertransact))  
z

## # A tibble: 21 x 4  
## fin\_Des2 num\_transact total\_amount amtpertransact  
## <chr> <int> <dbl> <dbl>  
## 1 EDUCATION 68 333764 4908  
## 2 TRANSPORTATION 190 289696 1525  
## 3 WHOLESALE 57 86804 1523  
## 4 LODGING 55 46109 838  
## 5 VETERINARY 41 16053 392  
## 6 ENTERTAINMENT 372 128014 344  
## 7 FINANCIAL SERVICES 185 56748 307  
## 8 BUSINESS SERVICES 74 19653 266  
## 9 DEPARTMENT STORES 981 197238 201  
## 10 GROCERIES 117 23381 200  
## # ... with 11 more rows

#Removing Education as it is outlier with $4908  
colnames(z)[1]="Transaction"  
  
z%>%slice(2:15)%>%ggplot(aes(reorder(Transaction,-amtpertransact,FUN=min),  
amtpertransact))+geom\_bar(stat="identity",fill=c(rep("limegreen",3),  
rep("lightslategray",11)))+labs(x="Transaction",y="Amount per Transaction")+coord\_flip()+geom\_text(aes(label=amtpertransact,hjust=1))+  
 theme(legend.position="none")



**Fig7:** Amount per transaction Vs Category

## Conclusions and Recommendations

From **Fig6** and **Fig7** we can conclude the following

1. **Department stores** and **Entertainment** categories account for atleast 40% of the transactions through WF Credit card.
2. **Transportation** and **Wholesale** have the highest amount spent per transactions.
3. WF can try to increase the number of transactions in the categories where the amount spent per transactions is higher. It can be achieved by increasing reward points or credit card offers in respective categories.

The assumption for recommendations is that the increase in number of transactions and the amount spent per transaction will lead to more profitablity for WF.

# Analyzing Daily interactions

Next we try to find insights from the daily interactions of customers with WF.The approach is to count the complaint categories and number of complaints in each category.We use description 3 to categorize the complaints.This model can be further granulated based on further discussions with the bank.We look for the most frequent descriptions in the customer complaints.

daily\_interactions$Des1=as.character(daily\_interactions$Des1)  
daily\_interactions$Des2=as.character(daily\_interactions$Des2)  
daily\_interactions$Des3=as.character(daily\_interactions$Des3)  
  
x=as\_data\_frame(daily\_interactions)  
glimpse(x)

## Observations: 6,669  
## Variables: 5  
## $ masked\_id <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...  
## $ Date <fctr> 07012016, 07022016, 07032016, 07032016, 07032016, 0...  
## $ Des1 <chr> "Add Banker Note", "Advance Reversal", "Consumer Rec...  
## $ Des2 <chr> "Customer Phone Change", "Customer to Customer Relat...  
## $ Des3 <chr> "POS Debit", "POS Preauthorization", "Settlements", ...

nrow(x)

## [1] 6669

x%>%group\_by(Des1)%>%summarise(count=n())%>%arrange(desc(count))

## # A tibble: 103 x 2  
## Des1 count  
## <chr> <int>  
## 1 Consumer Deposit Only flow selected 90  
## 2 CIVSALES\_CIP\_UPDATE 88  
## 3 BusinessÂ Â Deposit Only â Special Relationship flow 87  
## 4 Correction 85  
## 5 CIVSALES\_CIP\_VALIDATION 82  
## 6 Verify WFB Funds 82  
## 7 Balance Inquiry 81  
## 8 CIVSALES\_NEW\_PMA 81  
## 9 Authorized Debit 77  
## 10 Cash Check on Credit Card/LOC 77  
## # ... with 93 more rows

x%>%group\_by(Des3)%>%summarise(count=n())%>%arrange(desc(count))

## # A tibble: 103 x 2  
## Des3 count  
## <chr> <int>  
## 1 Service Open - Safe Deposit 90  
## 2 REPORT\_OUT\_OF\_WALLET\_DATA 88  
## 3 REPORT\_ACCOUNT\_CLOSE 87  
## 4 SOTA 85  
## 5 REPORT\_PASSWORD\_OPTIONS 82  
## 6 TRAVEL PLAN MAINTENANCE 82  
## 7 Referral to Business Insurance partner 81  
## 8 REPORT\_TELEPHONE\_TRANSFER 81  
## 9 Referral to Business Banking Group (BBG) partner 77  
## 10 REPORT\_AUTHENTICATION\_TRACKER\_DATA 77  
## # ... with 93 more rows

x%>%group\_by(Des2)%>%summarise(count=n())%>%arrange(desc(count))

## # A tibble: 103 x 2  
## Des2 count  
## <chr> <int>  
## 1 New Deposit Account Open funded via DDA to DDA Transfer 90  
## 2 Funds Transfer Detail Inquiry 88  
## 3 EFORM ACCESS 87  
## 4 NEW\_ACCT\_OWNERS 85  
## 5 Funds Transfer Maintained 82  
## 6 Withdrawal Reversal 82  
## 7 Deposit 81  
## 8 Interest Adjustment 81  
## 9 DEP\_PRODUCTS\_AND\_RATES 77  
## 10 Enroll Online Wires 77  
## # ... with 93 more rows

y=x%>%select(Des3)

## Stop Word Removal

The stop words in the interactions such as and,of,or,at needs to be removed inorder to analyze the interactions.We use the stop words present in **tidytext** library.

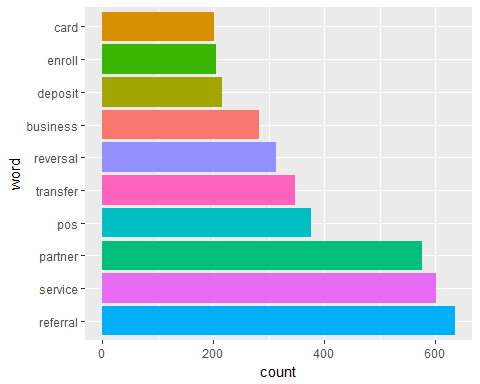
#install.packages(c("tidytext","stringr"))  
library(tidytext)  
data("stop\_words")  
my\_stopwords=data\_frame(word = c(as.character(1:100),"to","of","at","or"))

## Tokenization & Stemming

We also tokenize the sentence into single bag of words.We also stem the words inorder to reduce the computational complexity. More complicated tokenization can be performed with additional computational power.

library(stringr)  
  
#Tokenizing   
t1=y%>%unnest\_tokens(word,Des3)  
  
t1=t1%>%anti\_join(stop\_words,by="word")%>%anti\_join(my\_stopwords,by="word")%>% filter(str\_detect(word,"[a-z]"))

t1%>%group\_by(word)%>%summarise(ct=n())%>%top\_n(10)%>%  
 ggplot(aes(reorder(word,-ct,FUN=median),y=ct,fill=word))+  
 geom\_bar(stat="identity")+  
 labs(x="word",y="count")+coord\_flip()+theme(legend.position="none")

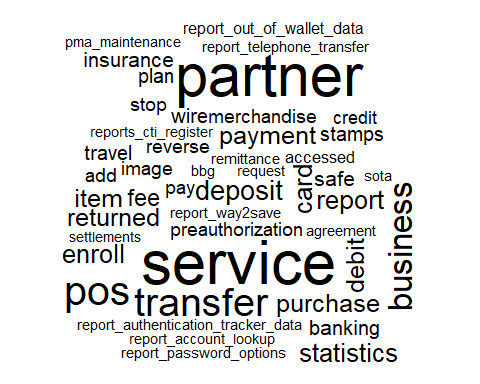


**Fig 8** : Most repeated words

## Word Cloud

**Fig 8** and **Fig9** shows that the most common interactions are for **referral**, **service** and **partner** categories. We should further analyze the categories to provide efficient customer service and to reduce cost by reducing the number of interactions.Depicting the same complaint freqency as word cloud as it offers better intution.With further domain knowledge the model can be considerably improved to observe frequent complaints and provide better service.

#install.packages("wordcloud")  
library(wordcloud)  
t1 %>% count(word) %>%with(wordcloud(word,n, max.words = 50))



**Fig 9:** Word Cloud of complaint categories

We only have the data of 50 customers. All the models built can be scaled and improved in terms of accuracy with more data and domain knowledge.