PROJECT DOCUMENTATION AND REPORT

1.1 Problem Statement:

To find probability of new sign ups converting into active users

1.2 Data Description

Information was collected from 385 FieldSense Customers against 16 variables for predicting the probability of them being an Active user.

Data source:

1. Super Admin Panel of FS
2. Customer Care lead sheet
3. SLAM
4. Web sources - Indiamart, LinkedIn, Company website, Zauba Corp

1.3 Objective

- Design most appropriate machine learning model to predict probability of new sign ups converting into active users

- Identify the co-relation between the “**Active”** and “**In Active”** users of FieldSense.

Chapter 2: Data Preparation

2.1 Data Description

"fieldsense.csv" is a raw file in csv format, and contains 385 observations with 16 variables as below.

```
> summary(field)
  CustomerType Domain     Type                      Source
 Billing: 59   No : 70   New:292   Google           :132
 Free   :288   Yes:277   Old: 55   Referral         : 77
                                   ExistingCustomer: 55
                                   Website          : 43
                                   EmailCampaign    : 31
                                   Social           :  5
                                   (Other)          :  4
         Designation  Enquiry_freq      EmailType
 Director       :105  Repeat: 16   Corporate:282
 IT Head        : 81  Unique:331   General  : 63
 Manager        : 51               Personal :  2
 CEO_GM         : 19
 HeadMarketing  : 18
 HR             : 16
 (Other)        : 57
                 Industry          Size               Year
 Manufacturer        :110   Min.   :   10.0   Min.   :  1.00
 Technology          : 61   1st Qu.:   50.0   1st Qu.:  8.00
 WholesalerandTrader: 34   Median :  100.0   Median : 14.00
 ConsumerServices    : 31   Mean   :  494.3   Mean   : 19.45
 Healthcare          : 22   3rd Qu.:  200.0   3rd Qu.: 25.00
 RealEsate           : 16   Max.   :10000.0   Max.   :200.00
 (Other)             : 73
   UserStatus
 Min.   :0.000
 1st Qu.:0.000
 Median :1.000
 Mean   :0.732
 3rd Qu.:1.000
 Max.   :1.000
```

Figure: Summary of dataset

2.2 Predictor Variable

- **Domain**
- **Customer Type:** Billing or Free
- **Source:** Referral, Website, Google, Existing Customer, Email Campaign, Existing Customer etc
- **Designation** of the person who put the enquiry
- **Enquiry Frequency**
- **Email type:** Corporate or General
- **Industry of the organization**
- **Size of Company**
- **Year of Establishment**

2.3 Outcome Variable: User Status

Proportion of the outcome variable in our dataset.

```
> prop.table(table(field_1$UserStatus))

        0         1 
0.3402597 0.6597403 
```

Active customers in our Dataset: 66%

Inactive customers in our Dataset: 34 %

Chapter 3: Plan of Action

3.1 Load data into R and install required packages

Load the dataset into R environment and importing required packages/ libraries which will include Caret, data table, grid Extra, corrplot, GGally, ggplot2, e1071, dplyr, e1071 etc.

3.2 Data Cleaning

Dataset was checked for discrepancies. Data cleaning was done to make it ready for analysis which involved treating

- **Missing values** - Missing values will be replaced with mean values for numerical variables and with mode values for categorical variables.
- **Duplicate observations** - Duplicate observations will be dropped if any.
- **Outliers** - Boxplot will be used to check outliers if any in the numerical variables.

3.3 Making data models and validating data models

The dataset was split to training set and test set in 75:25 proportions.

3.4 Pre modelling Bi-variate Analysis

To find the relationship between two categorical variables we use **Chi-Square.** This test is used to derive the statistical significance of relationship between the variables. Also, it tests whether the evidence in the sample is strong enough to generalize that the relationship for a larger population as well. It returns probability for the computed chi-square distribution with the degree of freedom.

1. Corelation between **Customer Type and User Status**
We have a chi-squared value of 13.002 and p value = 0.0003112. Since we get a p-Value less than the significance level of 0.05, we reject the null hypothesis and conclude that the two variables are in fact dependent

```
            0   1
  Billing   8  51
  Free    123 203
> #for Categorical & Categorical:Pearson's Chi-squared test
> chisq.test(field$CustomerType, field$UserStatus, correct=FALSE)

        Pearson's Chi-squared test

data:  field$CustomerType and field$UserStatus
X-squared = 13.002, df = 1, p-value = 0.0003112
```

2. Corelation between **Domain and User Status**

We have a chi-squared value of 1.76 and p value = 0.18. Since we get a p-Value greater than the significance level of 0.05, we do not reject the null hypothesis and conclude that the two variables do not have high level of dependency.

```
> #DOMAIN
> table(field$Domain, field$UserStatus)

       0   1
  No   20  53
  Yes 111 201
> chisq.test(field$Domain, field$UserStatus, correct = FALSE)

        Pearson's Chi-squared test

data:  field$Domain and field$UserStatus
X-squared = 1.7632, df = 1, p-value = 0.1842
```

3. Corelation between **Type and User Status**

We have a chi-squared value of 1.2615 and p value = 0.26. Since we get a p-Value is greater than the significance level of 0.05, we do not reject the null hypothesis and conclude that the two variables do not have high level of dependency.

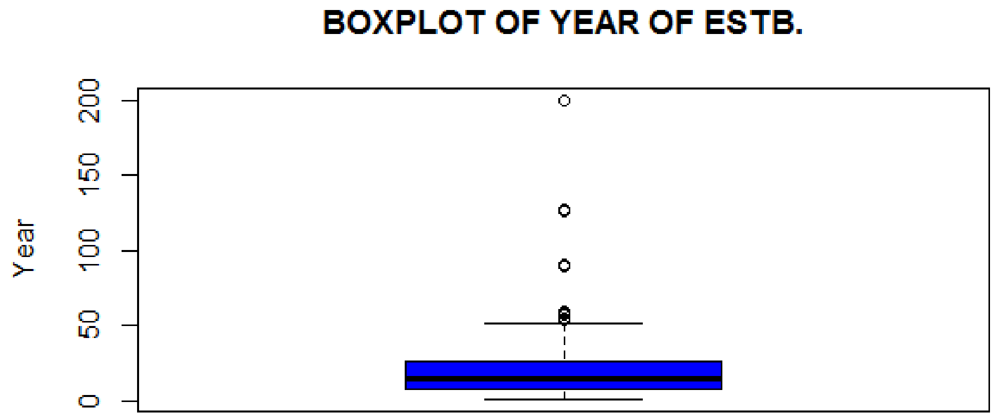
```
> #type
> table(field$Type, field$UserStatus)

       0   1
  New 115 212
  Old  16  42
> chisq.test(field$Type, field$UserStatus, correct = FALSE)

        Pearson's Chi-squared test

data:  field$Type and field$UserStatus
X-squared = 1.2615, df = 1, p-value = 0.2614
```

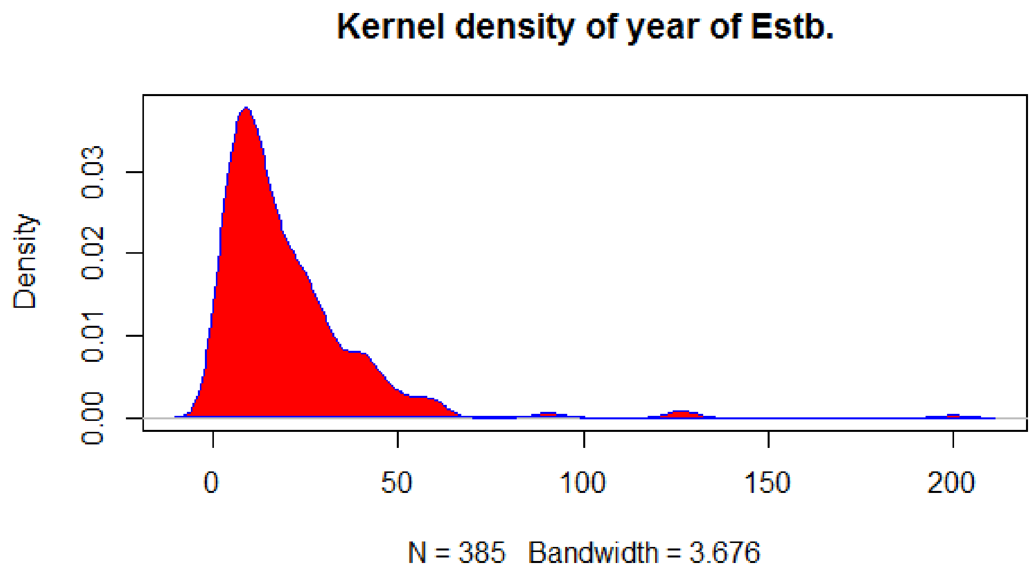
4. Corelation **between Enquiry frequency and User Status**

We have a chi-squared value of 3.5 and p value = 0.063. Since we get a p-Value less than the significance level of 0.05, we do not reject the null hypothesis and conclude that the two variables do not have high level of dependency.

```
> #Enq. freq
> table(field$Enquiry_freq, field$UserStatus)

          0   1
  Repeat    2  14
  Unique 129 240
> chisq.test(field$Enquiry_freq, field$UserStatus, correct = FALSE)

        Pearson's Chi-squared test

data:  field$Enquiry_freq and field$UserStatus
X-squared = 3.4459, df = 1, p-value = 0.06341
```

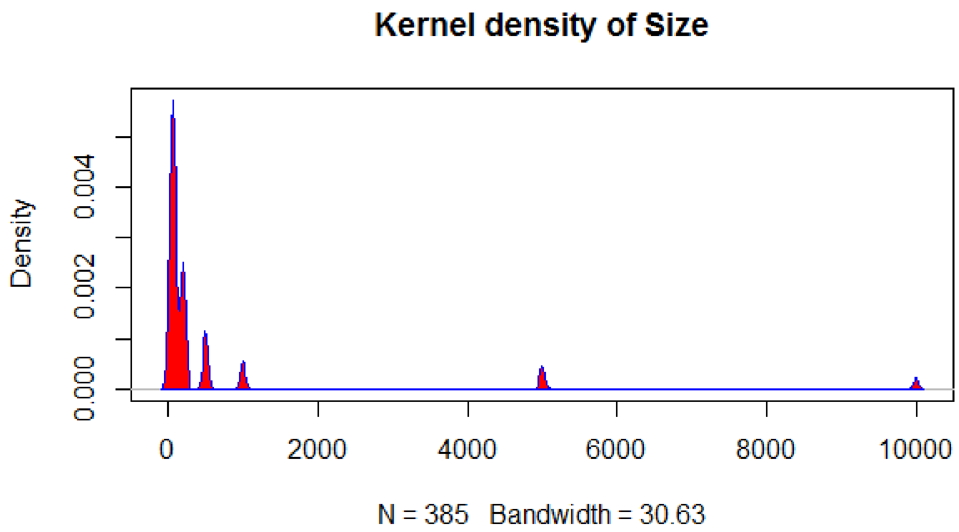
5. Box plot for **Year of Establishment**





Box plot and density map shows there is a need to normalise data for the variable "**Year of Establishment",** hence the data was normalized and added to the data list.

6. For Size



Density map shows there is a need to normalise data for the variable "**Size".**

Hence normalization was done and the data was clustered and grouped as follows.

```
#For year
field$Year_1 <- as.numeric(field$Year >= 41 & field$Year <= 1000)
field$Year_2 <- as.numeric(field$Year >= 20 & field$Year <= 40)
field$Year_3 <- as.numeric(field$Year >= 10 & field$Year <= 19)
field$Year_4 <- as.numeric(field$Year >= 6 & field$Year <= 9)
field$Year_5 <- as.numeric(field$Year >= 1 & field$Year <= 5)

#For Size
field$Size[field$Size>=500 & field$Size<10000]=10000
field$Size[field$Size>=100 & field$Size<200]=200
field$Size[field$Size>=10 & field$Size<75]=75
```

Feature Extraction

From **Chi-Square** test we found the correlation of the available variables in our data set.
So only variable with good significance level were considered and the rest of the variables were dropped.

The data was featured down to 9 independent variables and one outcome variable.

Chapter 4: Model Building

4.1 Using Gradient Boosting (GBM) model:

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function.

4.2 Confusion Matrix – GBM model

```
 Accuracy      Kappa
0.6736842 0.2403921
```

The classifier performed with an **accuracy** that is 24% (**kappa** of 67%) of 50%. is a statistic which measures inter-rater agreement for qualitative (categorical) items.

Y Pred. for n = 95		Y Actual		Model Accuracy
		Active	Inactive	67%
	Active	TP=51	FP= 12	
	Inactive	FN = 19	TN = 13	

- The model made a total of 95 predictions.
- Out of those 95 cases, the classifier predicted "Active" 63 times, and "Inactive" 32 times.
- In actual, 70 customers in the sample are active, and 25 are Inactive.

- **True positives (TP):** These are cases in which we predicted active and they are active in the real dataset also.
- **True negatives (TN):** We predicted Inactive, and they are inactive in the real dataset.
- **False positives (FP):** We predicted Active, but they are actually inactive. (Also known as a "Type I error.")
- **False negatives (FN):** We predicted Inactive, but they are actually active. (Also known as a "Type II error.")

Test Data set 63 ACTIVE, 32 INACTIVE

```
> table(testDF$UserStatus, prdval)
          prdval
           Active Inactive
  Active       51       12
  Inactive     19       13
```

4.3: Area under the curve?

AUC is an abbreviation for area under the **curve**. It is used to determine which of the used models predicts the classes best. It is a measure of how well a parameter can distinguish between two diagnostic groups.

```
Area under the curve: 0.684
```

4.4 Findings

4.4.1 Relative influence weightage of the variable in the dataset according to the model.

```
                                                          var   rel.inf
Size                                                     Size 9.6726441
Source.EmailCampaign                     Source.EmailCampaign 9.0770750
Industry.ConsumerServices           Industry.ConsumerServices 7.9216843
Designation.Director                     Designation.Director 6.5385559
Industry.Manufacturer                   Industry.Manufacturer 6.1854618
CustomerType.Free                           CustomerType.Free 5.1956123
Designation.IT Head                       Designation.IT Head 4.8121380
Source.Website                                 Source.Website 4.4696853
Source.Google                                   Source.Google 4.4424461
Year_3                                                 Year_3 3.8894445
EmailType.General                           EmailType.General 3.6061375
Source.Referral                               Source.Referral 3.4775868
Year_2                                                 Year_2 3.1186652
EmailType.Corporate                       EmailType.Corporate 3.0411695
Designation.Admin                           Designation.Admin 2.9270866
Year_1                                                 Year_1 2.8551651
Designation.Manager                       Designation.Manager 2.6118348
Year_4                                                 Year_4 2.6018079
Industry.Technology                       Industry.Technology 2.3443185
Type.Old                                             Type.Old 1.7001745
Industry.Healthcare                       Industry.Healthcare 1.6000418
Year_5                                                 Year_5 1.4289293
CustomerType.Billing                     CustomerType.Billing 1.2411637
Source.ExistingCustomer               Source.ExistingCustomer 1.1054389
Type.New                                             Type.New 1.0313787
Industry.WholesalerandTrader     Industry.WholesalerandTrader 0.8850329
Designation.SalesHead                   Designation.SalesHead 0.8386842
Designation.HeadMarketing           Designation.HeadMarketing  0.6261541
Designation.CEO_GM                         Designation.CEO_GM  0.3417870
Designation.ExecutiveAssistantAdmin Designation.ExecutiveAssistantAdmin 0.3057158
Industry.RealEsate                         Industry.RealEsate 0.1069798
Source.Acquist                                 Source.Acquist 0.0000000
Source.CustomerCare                       Source.CustomerCare 0.0000000
Source.Social                                   Source.Social 0.0000000
```

It was found by the model that the influence and weightage on being an active user was given to

1. Source: Email Campaign and Website.
2. Industry : Consumer Services and Manufacturer
3. Designation: Director and IT head.

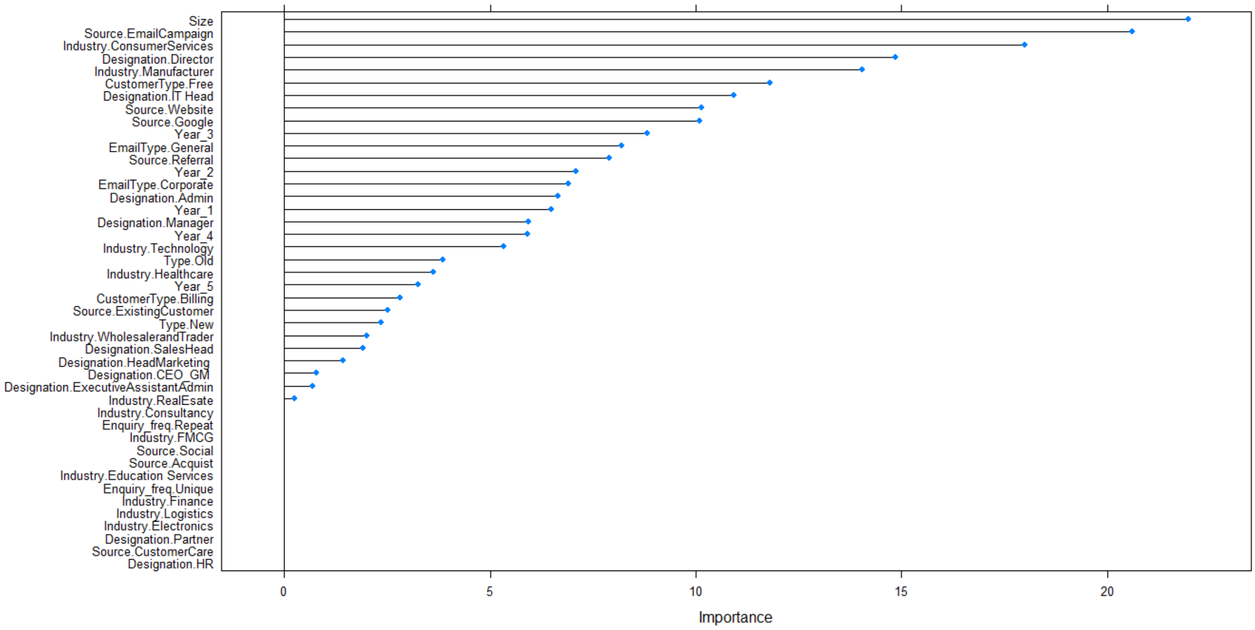


Figure: Relative importance of the variable

4.5 Output

For the test dataset of 95 variables, the model predicted the following probability of being an Active/Inactive user.

	Active	Inactive
1	0.752247	0.247753
2	0.979714	0.020286
3	0.988811	0.011189
4	0.952138	0.047862
5	0.984559	0.015441
6	0.765347	0.234653
7	0.959417	0.040583
8	0.93258	0.06742
9	0.90696	0.09304
10	0.891655	0.108345
11	0.905768	0.094232
12	0.82071	0.17929
13	0.234327	0.765673
14	0.101612	0.898388
15	0.924625	0.075375
16	0.459468	0.540532
17	0.665551	0.334449
18	0.573169	0.426831
19	0.559283	0.440717
20	0.38752	0.61248
21	0.63509	0.36491
22	0.894505	0.105495
23	0.953969	0.046031
24	0.915802	0.084198
25	0.901691	0.098309
26	0.937763	0.062237
27	0.592674	0.407326
28	0.368523	0.631477
29	0.916205	0.083795
30	0.761388	0.238612
31	0.932024	0.067976
32	0.858141	0.141859
33	0.502174	0.497826
34	0.702735	0.297265
35	0.921848	0.078152
36	0.923209	0.076791
37	0.340382	0.659618
38	0.591555	0.408445
39	0.465582	0.534418
40	0.069521	0.930479
41	0.275897	0.724103
42	0.295758	0.704242
43	0.708048	0.291952
44	0.806249	0.193751
45	0.830659	0.169341
46	0.959668	0.040332
47	0.710525	0.289475
48	0.80558	0.19442
49	0.849317	0.150683
50	0.891045	0.108955
51	0.908817	0.091183
52	0.945933	0.054067
53	0.86447	0.13553
54	0.946849	0.053151
55	0.805294	0.194706
56	0.817739	0.182261
57	0.70317	0.29683
58	0.424506	0.575494
59	0.79287	0.20713
60	0.234401	0.765599
61	0.725586	0.274414
62	0.612364	0.387636
63	0.967954	0.032046
64	0.705751	0.294249
65	0.485516	0.514484
66	0.766994	0.233006
67	0.940789	0.059211
68	0.32468	0.67532
69	0.668454	0.331546
70	0.704256	0.295744
71	0.974557	0.025443
72	0.828043	0.171957
73	0.898819	0.101181
74	0.200542	0.799458
75	0.933001	0.066999
76	0.427913	0.572087
77	0.83784	0.16216
78	0.676077	0.323923
79	0.109613	0.890387
80	0.62752	0.37248
81	0.863802	0.136198
82	0.914425	0.085575
83	0.602136	0.397864
84	0.313471	0.686529
85	0.138555	0.861445
86	0.238613	0.761387
87	0.602136	0.397864
88	0.677235	0.322765
89	0.138555	0.861445
90	0.187494	0.812506
91	0.0375	0.9625
92	0.569017	0.430983
93	0.676077	0.323923
94	0.471931	0.528069
95	0.138555	0.861445

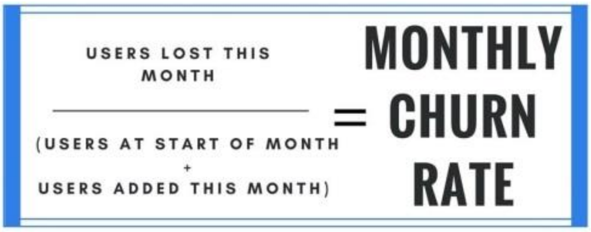
Chapter 5: Analysis and inferences

5.1 What caused the Churn in our place?

According to our interaction with the customers over the Phone and reading their previous activity in SLAM during the Data digging process, we found the following to be the problem of the churn.

- Poor on boarding experience
- Poor user interface or user experience – (Problems in calculating distance covered).
- Lack of features
- Competitor products

5.2 App activity using Clevertap dataset



December (1st to 31st)	
Users at the start of the month	313
New users added during that month	110
Users lost at the end of the month	62
Monthly churn rate	0.146572
Users at the end of the month	361

251 customers carried forward by end of the month

January (1st to 31st)	
Users at the start of the month	361
New users added during that month	84
Users lost at the end of the month	78
Monthly churn rate	0.175281
Users at the end of the month	367

283 customers carried forward by end of the month

February (1st to 28th)	
Users at the start of the month	367
New users added during that month	80
Users lost at the end of the month	65
Monthly churn rate	0.145414
Users at the end of the month	382

302 customers carried forward by end of the month

March (1st to 31st)	
Users at the start of the month	382
New users added during that month	102
Users lost at the end of the month	70
Monthly churn rate	0.144628
Users at the end of the month	414

312 customers carried forward by end of the month

Chapter 6: Recommendations

6.1 Few strategic to decrease App Churn

Optimize on boarding

If the on boarding process doesn’t immediately showcase your app’s core value, users will churn. Keep the on boarding process focused on benefits. Strip down complexity, limit the number of steps, and get users to experience your app’s aha moment as fast as possible. The recent DIY process should have optimised the process, but a proper watch has to be kept in step wise boarding experience of the new customers.

Leverage Push notifications

Send automated push notifications to a user’s home screen to encourage repeat visits, engagement. With a personalized approach, these notifications can reactivate users who are at risk of churning.

Consider deep linking

Mobile apps operate on URIs instead of URLs. That means deep links (or direct links) can take users right to a particular screen inside your app. These links can launch an app from exactly where a user left off, or take them to a specific product page. This can be a smart approach to reactivate monthly active users. Making sure the deep link provides an active productive call of action for the user.

Personalize

With personalized interactions and relevant messaging, users feel like you’re actually speaking to them. You can’t take one-size-fits all approach. Leverage user data like first name, behaviours, and preferences to customize interactions.

6.2 Further plan of action

We will be analyzing the Clevertap data on the basis of the User/Company’s daily activity. And check

- Are the customers doing more or less featured activity over time?
- Is the rate at which we are losing active customers getting better or worse?
- How and when are users returning back after their first App activity?
- Are the new customers sticking around for the second, third, fourth month?
- Also we will have to uncover the specific reasons users leave and which actions drive retention.

R - Code

```
field<- read.csv("Fieldsense_2.csv")
View(field)

colnames(field)
dim(field)
str(field)
summary(field)

#Finding missing values
table(field$Designation)
str(field)

#Check NAs and less than 0 values
sapply(field, function(x) sum(is.na(x)))

#Univariate analysis

#Chi q. test
table(field$CustomerType, field$UserStatus)

chisq.test(field$CustomerType, field$UserStatus, correct=FALSE)
#We have a chi-squared value of 13.002 and p value = 0.0003112 Since we get a p-Value less than the significance level of 0.05, we reject the null hypothesis and conclude that the two variables

#DOMAIN
table(field$Domain, field$UserStatus)
chisq.test(field$Domain, field$UserStatus, correct = FALSE)

#type
table(field$Type, field$UserStatus)
chisq.test(field$Type, field$UserStatus, correct = FALSE)

#Enq. freq
table(field$Enquiry_freq, field$UserStatus)
chisq.test(field$Enquiry_freq, field$UserStatus, correct = FALSE)

#ANOVA
field$UserStatus <- as.factor(field$UserStatus)
aov1 = aov(field$Industry ~ field$UserStatus)
summary(aov1)

stat.desc(field$Size)

boxplot(field$Year,
        main = toupper("Boxplot of Year of Estb."),
        ylab = "Year",
        col = "blue")

#Kernal desity plot
d <- density(field$Year)
plot(d, main = "Kernel density of year of Estb.")
polygon(d, col = "red", border = "blue")

#Kernal desity plot
d <- density(field$Size)
```

```
plot(d, main = "Kernel density of Size")
polygon(d, col = "red", border = "blue")

#for year
field$Year_1 <- as.numeric(field$Year >= 41 & field$Year <= 1000)
field$Year_2 <- as.numeric(field$Year >= 20 & field$Year <= 40)
field$Year_3 <- as.numeric(field$Year >= 10 & field$Year <= 19)
field$Year_4 <- as.numeric(field$Year >= 6 & field$Year <= 9)
field$Year_5 <- as.numeric(field$Year >= 1 & field$Year <= 5)

field$Year <- NULL

#for Size
field$Size[field$Size>=500 & field$Size<10000]=10000
field$Size[field$Size>=100 & field$Size<200]=200
field$Size[field$Size>=10 & field$Size<75]=75


#feature Extraction
field$Contacted <- NULL
field$Contact.Person <- NULL
field$Tel.No <- NULL
field$Email.ID <- NULL
field$Invoicee <- NULL
field$Domain <-NULL

View(field)

#field <- field[,c(11,1,2,3,5:11)] #Reorder variables to put target variable to the first place
#field$UserStatus.1 <- NULL

# dummy variables for factors/characters
field$CustomerType <- as.factor(field$CustomerType)
fielddummy <- dummyVars("~.",data=field, fullRank=F)
field_1 <- as.data.frame(predict(fielddummy,field))
print(names(field))


View(field_1)
str(field_1)
str(UserStatus)


#field_1$UserStatus.0 <- NULL
#field_1$UserStatus.1 <- NULL

#added usestatus column
field_1$UserStatus <- paste(field$UserStatus)


View(field_1)
str(field_1)

# Encoding the target feature as factor
field_1$UserStatus <- as.factor(field_1$UserStatus)

# what is the proportion of your outcome variable?
```

```
prop.table(table(field_1$UserStatus))

 # save the outcome for the glmnet model
tempOutcome <- field_1$UserStatus

# generalize outcome and predictor variables
outcomeName <- 'UserStatus'
predictorsNames <- names(field_1)[names(field_1) != outcomeName]

# model it
##################################################
# get names of all caret supported models
names(getModelInfo())

field_1$UserStatus <- ifelse(field_1$UserStatus==1,'Active','Inactive')

# pick model gbm and find out what type of model it is
getModelInfo()$gbm$type

# split data into training and testing chunks
set.seed(1234)

splitIndex <- createDataPartition(field_1[,"UserStatus"], p = .75, list = FALSE, times = 1)
trainDF <- field_1[ splitIndex,]
testDF  <- field_1[-splitIndex,]

View(testDF)
View(trainDF)
dim(trainDF)
dim(testDF)

splitIndex

# create caret trainControl object to control the number of cross-validations performed
objControl <- trainControl(method='cv', number=5, returnResamp='none',classProbs = TRUE)


# run model
objModel <- train(trainDF[,predictorsNames], as.factor(trainDF[,outcomeName]),
                  method='gbm',
                  trControl=objControl,
                  metric = "ROC",
                  preProc = c("center", "scale"))


# find out variable importance
summary(objModel)

# find out model details
objModel


# evalutate model
##################################################
# get predictions on your testing data

# class prediction
```

```
predictions <- predict(object=objModel, testDF[,predictorsNames], type='raw')
head(predictions)


library(klaR)
prdval <- predict(objModel, trainDF)
table(trainDF$UserStatus, prdval)
trainDF

# probabilities
predictions <- predict(object=objModel, testDF[,predictorsNames], type='prob')
View(predictions)

#PRINT roc
auc <- roc(ifelse(testDF[,outcomeName]=="Active",1,0), predictions[[2]])
print(auc$auc)

plot(varImp(objModel,scale=F))

#Exporting the Prob. on Test data set
write.csv(predictions, "Probabilty_test.csv")
write.csv(testDF, "Test_test.csv")

View(field)

library(ggplot2)
ggplot(field, aes(x=Size, y=Year)) + geom_point()
```