Goal

Understanding why and when customers are most likely to contact for problems can lead to actions to improve customer satisfaction as well as planning resources in advance. contact.

```
library(data.table)
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------
---------- tidyverse 1.2.1 --
```

```
## v ggplot2 3.1.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.8
## v tidyr   0.8.2     v stringr 1.3.1
## v readr   1.3.1     v forcats 0.3.0
```

```
## -- Conflicts ------------------------------------------------------------
---- tidyverse_conflicts() --
## x dplyr::between()   masks data.table::between()
## x dplyr::filter()    masks stats::filter()
## x dplyr::first()     masks data.table::first()
## x dplyr::lag()       masks stats::lag()
## x dplyr::last()      masks data.table::last()
## x purrr::transpose() masks data.table::transpose()
```

```
library(tidyr)
library(dplyr)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(ggplot2)
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
library(xgboost)
```

```
##
## Attaching package: 'xgboost'
```

```
## The following object is masked from 'package:dplyr':
##
##     slice
```

```
library(Matrix)
```

```
##
## Attaching package: 'Matrix'

## The following object is masked from 'package:tidyr':
##
##     expand

library(Metrics)
library(mlr)

## Loading required package: ParamHelpers

data <- fread('dataset.csv')
```

We have 82442 unique asst_id and 100,000 rows with some duplicate entries in the data. Before diving into the data, let's clean up the data first.
1. Fill "" with NA & change character to lower case
2. Add few variables based on the duration of manufacture, contract, and their contact timing(proportion to the contract length)

```
data[data == ""] <- NA
data$contract_type <- tolower(data$contract_type)
length(unique(data$asst_id))

## [1] 82442

data           <-  data[!duplicated(data)] # remove redundant values
#glimpse(data)
```

Feature Engineering
Calculate warranty duration, contact for problems and other variables

```
data$mnfture.duration      <- data$contract_st-data$mnfture_wk   # how long i
t takes to manufacture
data$contract.duration     <- data$contract_end-data$contract_st # have warra
nty or not
data$have_contract         <- as.numeric(ifelse(data$contract.duration <= 0,
0,1))
data$contract.duration     <- ifelse(data$contract.duration <= 0, 1, data$con
tract.duration) # add one to prevent infinite values in contact.point
data$tot.duration          <- data$contract_end-data$mnfture_wk # total durat
ion(from manufacture to end of warranty)
data$interact.duration     <- data$contact_wk-data$mnfture_wk
data$contact.b4contract    <- as.numeric(ifelse((data$contact_wk-data$contrac
t_st)< 0, 1, 0)) # contact before or after warranty starts, 1 = before

# number of weeks to contact after warranty starts
data$contact.prob          <- data$contact_wk-data$contract_st
# proportion of #weeks to contact/warranty length
data$contact.point         <- data$contact.prob /data$contract.duration
data  <- data[, -c('contract_end','contract_st','contact_wk','mnfture_wk')]
```

Few entries have unusually long manufacture duration and were handled by an experienced agent.
Some of these unusual entries have contract and some don't have. They should be analyze seperately.

```
#  Flag and remove from further analysis
data$flag <- ifelse(data$mnfture.duration >250,1,0)
```

Data Cleaning

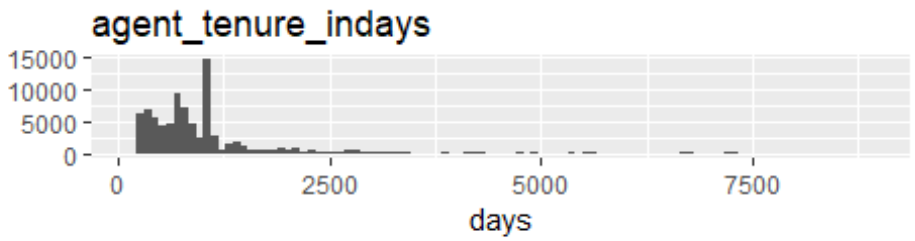
It does not make sense warranty becomes active before a product exists. Maybe, there are reasons for this activiy. Further investigaion will be need if I have more time. Remove from further analysis.

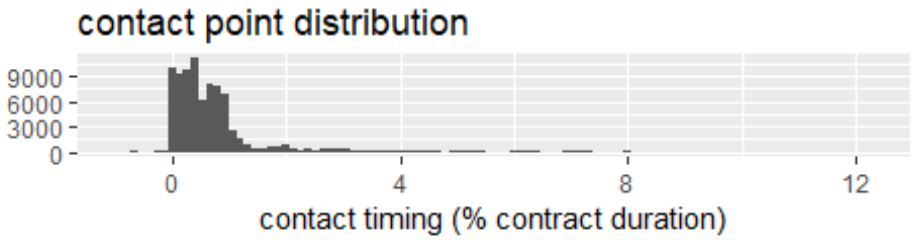
```
data  <- data[data[,mnfture.duration >= 0]]
```

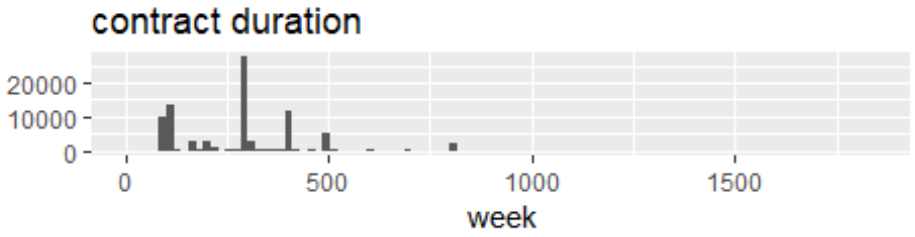
Distribution

Check the distribution of agent_tenure_indays, `contract.duration.` and contact.point.
All distributions are right skewed.

```
p1 <- qplot(data$agent_tenure_indays[!is.na(data$agent_tenure_indays)], bins
= 100, main = "agent_tenure_indays",xlab = 'days')
p2 <- qplot(data$contact.point[!is.na(data$contact.point) & data$flag ==0], b
ins = 100, main = "contact point distribution", xlab = 'contact timing (% con
tract duration)')
p3 <- qplot(data$contract.duration[data$flag ==0], bins = 100, main = "contra
ct duration", xlab = 'week')
grid.arrange(p1, p2,p3, nrow = 3)
```







Found duplicate asst_id

```
# t-test on agent_tenure_indays (single or duplicate asst_ids)
x               <-  data %>% group_by(asst_id) %>% mutate(dif_agent = ifelse(f
irst(agent_tenure_indays) == last(agent_tenure_indays),0,1))
data$dif_agent <-  x$dif_agent

sum(data$dif_agent,na.rm=T) # 682 entries

## [1] 682

mean(data$agent_tenure_indays,na.rm=T)

## [1] 986.008

var.test(data$agent_tenure_indays[data$dif_agent == 0], data$agent_tenure_ind
ays[data$dif_agent == 1] ) # unequal variance between groups

##
##  F test to compare two variances
##
## data:  data$agent_tenure_indays[data$dif_agent == 0] and data$agent_tenure
_indays[data$dif_agent == 1]
## F = 0.45678, num df = 81809, denom df = 681, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.4093731 0.5067887
## sample estimates:
```

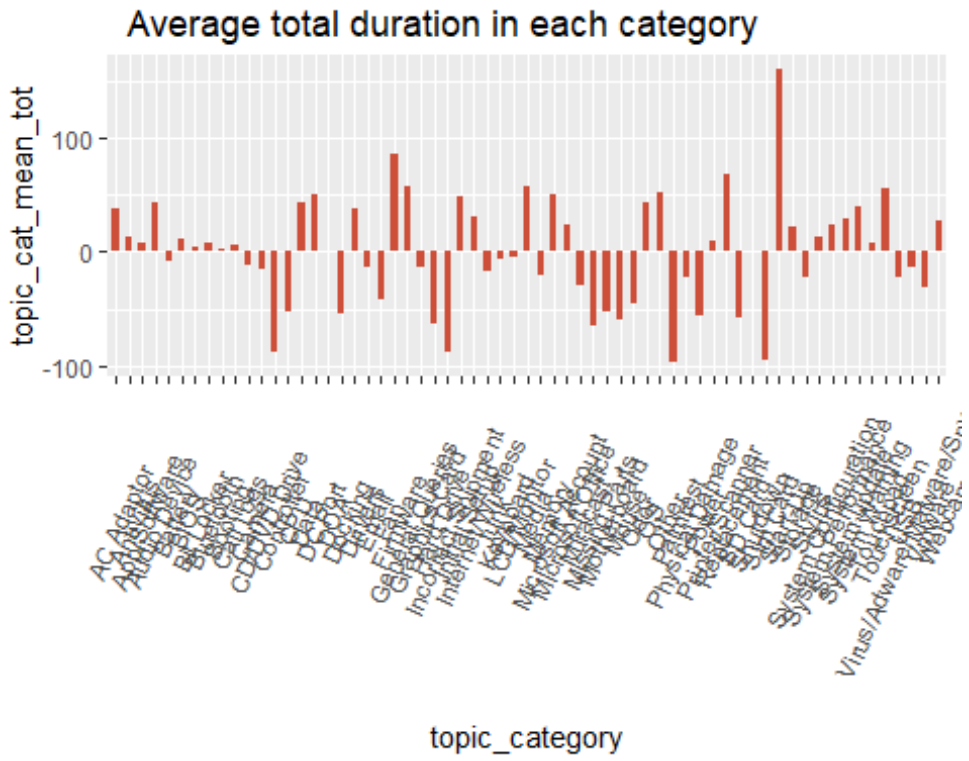
```
## ratio of variances
##          0.4567782

t.test(data$agent_tenure_indays[data$dif_agent == 0], data$agent_tenure_inday
s[data$dif_agent == 1], alternative = "two.sided", paired = FALSE, var.equal
= FALSE)

##
##  Welch Two Sample t-test
##
## data:  data$agent_tenure_indays[data$dif_agent == 0] and data$agent_tenure
_indays[data$dif_agent == 1]
## t = -3.9713, df = 686.2, p-value = 7.902e-05
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -287.40447  -97.23755
## sample estimates:
## mean of x mean of y
##   984.418  1176.739
```

Calculate average total duration in each group

Fill missing topic_category,product_type,and issue_type with average (manufacture + contractrct duration)

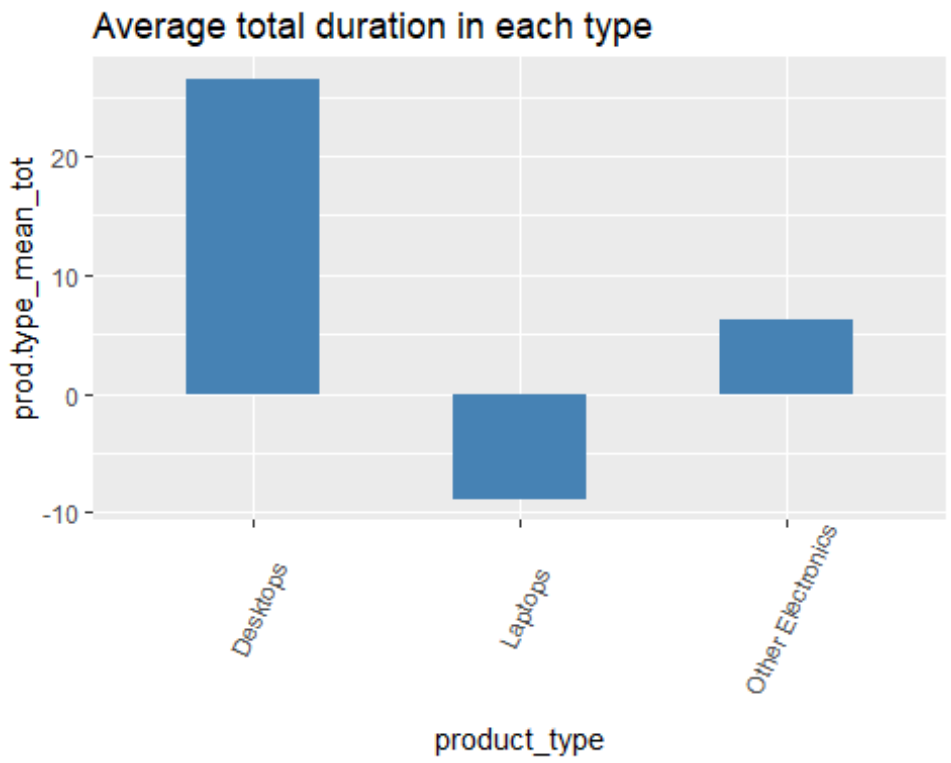
topic_category vs tot.duration

```
topic_category.new                          <- aggregate(x = data$tot.duratio
n, by= list(data$topic_category), FUN= mean,na.action = na.omit)
colnames(topic_category.new)                <- c("topic_category", "mean_tot"
)  # change column names
topic_category.new$topic_cat_mean_tot  <- mean(topic_category.new$mean_tot)-(
topic_category.new$mean_tot)

ggplot(topic_category.new, aes(x= topic_category, y = topic_cat_mean_tot )) +

   geom_bar(stat="identity", width=.5, fill="tomato3") +
   labs(title="  Average total duration in each category") +
   theme(axis.text.x = element_text(angle=65, vjust=0.6))
```



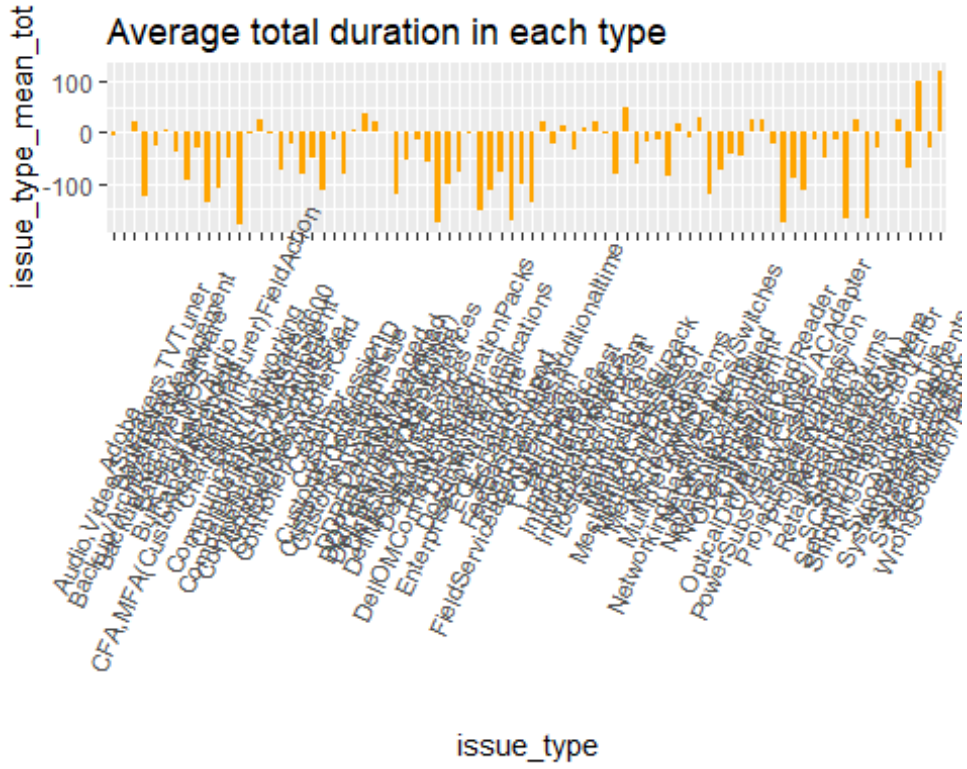
product_type vs tot.duration

```
product_type.new            <- aggregate(x = data$tot.duration, by= list(data
$product_type), FUN= mean,na.action = na.omit)
product_type.new$prod.type_mean_tot  <- product_type.new$x -mean(data$tot.dur
ation, na.rm = T)
colnames(product_type.new)  <- c("product_type", "mean_tot","prod.type_mean_t
ot")

ggplot(product_type.new, aes(x= product_type, y= prod.type_mean_tot)) +
   geom_bar(stat="identity", width=.5, fill="steelblue") +
   labs(title="Average total duration in each type")+
   theme(axis.text.x = element_text(angle=65, vjust=0.6))
```



issue_type vs tot.duration
Transform issue_type, trim white space, and map values

```
data$issue_type <- gsub(" ", "", data$issue_type, fixed = TRUE)
issue_type.new            <- aggregate(x = data$tot.duration, by= list(data$i
ssue_type), FUN= mean,na.action = na.omit)
issue_type.new$issue_type_mean_tot  <- issue_type.new$x -mean(data$tot.durati
on, na.rm = T)
colnames(issue_type.new)  <- c("issue_type", "mean_tot","issue_type_mean_tot"
)

ggplot(issue_type.new, aes(x= issue_type, y= issue_type_mean_tot)) +
   geom_bar(stat="identity", width=.5, fill="orange") +
   labs(title="Average total duration in each type") +
   theme(axis.text.x = element_text(angle=65, vjust=0.6))
```



Fill other missing values

```
data <- setDT(data)[, country:= country[!is.na(country)][1L] , by = asst_id]
#filling the missing values using same asst_id
(table(data$region))

##
##     Hogwarts Middle Earth    Milky Way
##        57344        20436         4718

data$region[is.na(data$region)]           <- 'Hogwarts' # fill missing values
 with the most common region
data$parts_sent                                          <- ifelse(data$parts_c
t==0, 0, data$parts_sent) # fill Na with zero, if no parts were sent
data$parts_sent                                          <- gsub("[[:digit:]\\(
)]",'',data$parts_sent)  # remove(n) #1741 unique
data$repeat_parts_sent                                   <- gsub("[[:digit:]\\(
)]",'',data$repeat_parts_sent)  # remove(n)
```

Merge Data

Replace categorical variables with new numeric varaible (topic_category,product_type,issue_type)

```
data <- merge(data, product_type.new[,c(1,3)], all.x = T, by = "product_type"
)
data <- merge(data, topic_category.new[,c(1,3)], all.x = T, by = "topic_categ
ory")
```

```
data <- merge(data, issue_type.new[,c(1,3)], all.x = T, by = "issue_type")

# Convert column type : character to factor
factor.names<-c('contact_type','repair_type','product_type','diagnostics', 'r
egion')
for (f in factor.names){if(class(data[[f]])=="character"){
   data[[f]]<-as.numeric(as.factor(data[[f]]))
}}

# Create a new set

df  <- data[!data$flag == 1] # ignore flag data for now
df  <- df[,-c("topic_category","product_type", "parts_sent","repeat_parts_sen
t","issue_type","contact.prob", "country","flag" )]

rm(product_type.new,topic_category.new,issue_type.new)#;gc()
colnames(df)[18:20] <- c('product_type','topic_category','issue_type')
```

Model Building

Approach :

Since there are still many missing values, and it will take a longer time to figure out all of them. I decide to use Xgboot which can handle missing values as well as generate important factors.

Goal: Predict contact.point

```
# create dense matrix
ff <-  ~ . - 1
df  <- df[!contact.b4contract ==1]  # only focus on customers who contact for
 a problem after having contract (include the contract expire ones)
mf  <- model.frame(formula = ff, data = df[,-c(1,4:8,13:17)], na.action = na.
pass)
mat <- model.matrix(object = ff, data = mf)
y <- as.matrix(sqrt(df$contact.point))

# create spare matrix, split train and test

set.seed(12243)
test_ind <- sample(0.2*nrow(df), replace = F)
train    <- Matrix::Matrix(mat[-test_ind,], sparse = T)
test     <- Matrix::Matrix(mat[test_ind,], sparse = T)
train_y  <- y[-test_ind]
test_y   <- y[test_ind]
rm(mat)

m1     <- xgboost(data = train, label = train_y,
                  max_depth = 9,
                  nround= 449,
                  lambda=0.535,
                  gamma=0.374,
```

```
                    alpha=0.576,
                    eta=0.11,
                    min_child_weight=3.9,
                    subsample = 0.805,
                    colsample_bytree = 0.69,
                    objective = "reg:linear",
                    eval_metric = "rmse",
                    seed = 123,
                    element = 10,
                    early_stopping_rounds = 20,
                    verbose = F, missing = NA)
y_pred <- predict(m1, newdata = test, missing = NA)
rmse(test_y,y_pred)

## [1] 0.2818246

data_frame(Actuals = test_y, Predictions = y_pred) %>%
   ggplot(aes(x = Actuals, y = Predictions)) +
   geom_point(color = palette()[3]) +
   geom_abline(intercept = 0, slope = 1, color = 'black', linetype = 'dashed'
) +
   labs(title = ' Contact Timing Predictions ')
```