MTN TELECOM SUBSCRIBERS INSIGHTS

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.5
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

1. Defining the Question

1.1 Specifying the Data Analysis Question The management would like to get your assistance in understanding the subscribed customers. Your recommendations informed by your analysis will help them make decisions on effective customer retention programs

1.2 Defining the Metric for Success Understanding why customers leave for other operator

1.3 Understanding the context MTN Telecom offers mobile and internet services to its customers. These services include phone, multiple lines, internet, online security, online backup, device protection, tech support, and streaming TV and movies. You have been provided with the current customer data. Since you will be working towards a descriptive report than a predictive one, you decide to think critically of the kind of questions that would help you craft customer retention programs. You then later use the given data set to answer your questions but before you start, you reading, explore, clean and visualise your dataset.

1.4 Recording the Experimental Design The steps to be taken include: Load dataset and preview its summarized information to get a feel of what you will be working with. Carry out data cleaning. Carry out data analysis. Interpret results. Provide recommendations based on results of analysis. Challenge your solution.

1.5 Data Relevance For now, the data we have contains churn data which will be critical for our research specific analysis.

2 Data cleaning and preparation

```
mtn_customers_df <- read_csv('telecom_customer.csv')
```

2.1 load and preview the data

```
## Rows: 7050 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (17): customerID, GENDER, PARTNER, Dependents, PhoneService, MultipleLin...
## dbl  (4): SeniorCitizen, tenure, MonthlyCharges, TotalCharges
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(mtn_customers_df,5)
```

```
## # A tibble: 5 x 21
##   custom~1 GENDER Senio~2 PARTNER Depen~3 tenure Phone~4 Multi~5 Inter~6 Onlin~7
##   <chr>    <chr>    <dbl> <chr>   <chr>    <dbl> <chr>   <chr>   <chr>   <chr>
## 1 7590-VH~ Female       0 Yes     No           1 No      No pho~ DSL     No
## 2 5575-GN~ Male         0 No      No          34 Yes     No      DSL     Yes
## 3 3668-QP~ Male         0 No      No           2 Yes     No      DSL     Yes
## 4 7795-CF~ Male         0 No      No          45 No      No pho~ DSL     Yes
## 5 9237-HQ~ Female       0 No      No           2 Yes     No      Fiber ~ No
## # ... with 11 more variables: OnlineBackup <chr>, DeviceProtection <chr>,
## #   TECHSUPPORT <chr>, StreamingTV <chr>, StreamingMovies <chr>,
## #   Contract <chr>, PaperlessBilling <chr>, PaymentMethod <chr>,
## #   MonthlyCharges <dbl>, TotalCharges <dbl>, Churn <chr>, and abbreviated
## #   variable names 1: customerID, 2: SeniorCitizen, 3: Dependents,
## #   4: PhoneService, 5: MultipleLines, 6: InternetService, 7: OnlineSecurity
```

```
tail(mtn_customers_df,5)
```

```
## # A tibble: 5 x 21
##   custom~1 GENDER Senio~2 PARTNER Depen~3 tenure Phone~4 Multi~5 Inter~6 Onlin~7
##   <chr>    <chr>    <dbl> <chr>   <chr>    <dbl> <chr>   <chr>   <chr>   <chr>
## 1 6840-RE~ Male         0 Yes     Yes         24 Yes     Yes     DSL     Yes
## 2 2234-XA~ Female       0 Yes     Yes         72 Yes     Yes     Fiber ~ No
## 3 4801-JZ~ Female       0 Yes     Yes         11 No      No pho~ DSL     Yes
## 4 8361-LT~ Male         1 Yes     No           4 Yes     Yes     Fiber ~ No
## 5 3186-AJ~ Male         0 No      No          66 Yes     No      Fiber ~ Yes
## # ... with 11 more variables: OnlineBackup <chr>, DeviceProtection <chr>,
## #   TECHSUPPORT <chr>, StreamingTV <chr>, StreamingMovies <chr>,
## #   Contract <chr>, PaperlessBilling <chr>, PaymentMethod <chr>,
## #   MonthlyCharges <dbl>, TotalCharges <dbl>, Churn <chr>, and abbreviated
## #   variable names 1: customerID, 2: SeniorCitizen, 3: Dependents,
## #   4: PhoneService, 5: MultipleLines, 6: InternetService, 7: OnlineSecurity
```

```
glimpse(mtn_customers_df)
```

```
## Rows: 7,050
## Columns: 21
## $ customerID       <chr> "7590-VHVEG", "5575-GNVDE", "3668-QPYBK", "7795-CFOCW~
## $ GENDER           <chr> "Female", "Male", "Male", "Male", "Female", "Female",~
## $ SeniorCitizen    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ PARTNER          <chr> "Yes", "No", "No", "No", "No", "No", "No", "No", "Yes~
## $ Dependents       <chr> "No", "No", "No", "No", "No", "No", "Yes", "No", "No"~
## $ tenure           <dbl> 1, 34, 2, 45, 2, 8, 22, 10, 28, 62, 13, 16, 58, 49, 2~
## $ PhoneService     <chr> "No", "Yes", "Yes", "No", "Yes", "Yes", "Yes", "No", ~
## $ MultipleLines    <chr> "No phone service", "No", "No", "No phone service", "~
## $ InternetService  <chr> "DSL", "DSL", "DSL", "DSL", "Fiber optic", "Fiber opt~
## $ OnlineSecurity   <chr> "No", "Yes", "Yes", "Yes", "No", "No", "No", "Yes", "~
## $ OnlineBackup     <chr> "Yes", "No", "Yes", "No", "No", "No", "Yes", "No", "N~
## $ DeviceProtection <chr> "No", "Yes", "No", "Yes", "No", "Yes", "No", "No", "Y~
## $ TECHSUPPORT      <chr> "No", "No", "No", "Yes", "No", "No", "No", "No", "Yes~
## $ StreamingTV      <chr> "No", "No", "No", "No", "No", "Yes", "Yes", "No", "Ye~
## $ StreamingMovies  <chr> "No", "No", "No", "No", "No", "Yes", "No", "No", "Yes~
## $ Contract         <chr> "Month-to-month", "One year", "Month-to-month", "One ~
## $ PaperlessBilling <chr> "Yes", "No", "Yes", "No", "Yes", "Yes", "Yes", "No", ~
## $ PaymentMethod    <chr> "Electronic check", "Mailed check", "Mailed check", "~
## $ MonthlyCharges   <dbl> 29.85, 56.95, 53.85, 42.30, 70.70, 99.65, 89.10, 29.7~
## $ TotalCharges     <dbl> 29.85, 1889.50, 108.15, 1840.75, 151.65, 820.50, 1949~
## $ Churn            <chr> "No", "No", "Yes", "No", "Yes", "Yes", "No", "No", "Y~
```

```
str(mtn_customers_df)
```

```
## spec_tbl_df [7,050 x 21] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ customerID      : chr [1:7050] "7590-VHVEG" "5575-GNVDE" "3668-QPYBK" "7795-CFOCW" ...
##  $ GENDER          : chr [1:7050] "Female" "Male" "Male" "Male" ...
##  $ SeniorCitizen   : num [1:7050] 0 0 0 0 0 0 0 0 0 0 ...
##  $ PARTNER         : chr [1:7050] "Yes" "No" "No" "No" ...
##  $ Dependents      : chr [1:7050] "No" "No" "No" "No" ...
##  $ tenure          : num [1:7050] 1 34 2 45 2 8 22 10 28 62 ...
##  $ PhoneService    : chr [1:7050] "No" "Yes" "Yes" "No" ...
##  $ MultipleLines   : chr [1:7050] "No phone service" "No" "No" "No phone service" ...
##  $ InternetService : chr [1:7050] "DSL" "DSL" "DSL" "DSL" ...
##  $ OnlineSecurity  : chr [1:7050] "No" "Yes" "Yes" "Yes" ...
##  $ OnlineBackup    : chr [1:7050] "Yes" "No" "Yes" "No" ...
##  $ DeviceProtection: chr [1:7050] "No" "Yes" "No" "Yes" ...
##  $ TECHSUPPORT     : chr [1:7050] "No" "No" "No" "Yes" ...
##  $ StreamingTV     : chr [1:7050] "No" "No" "No" "No" ...
##  $ StreamingMovies : chr [1:7050] "No" "No" "No" "No" ...
##  $ Contract        : chr [1:7050] "Month-to-month" "One year" "Month-to-month" "One year" ...
##  $ PaperlessBilling: chr [1:7050] "Yes" "No" "Yes" "No" ...
##  $ PaymentMethod   : chr [1:7050] "Electronic check" "Mailed check" "Mailed check" "Bank transfer (a
##  $ MonthlyCharges  : num [1:7050] 29.9 57 53.9 42.3 70.7 ...
##  $ TotalCharges    : num [1:7050] 29.9 1889.5 108.2 1840.8 151.7 ...
##  $ Churn           : chr [1:7050] "No" "No" "Yes" "No" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   customerID = col_character(),
##   ..   GENDER = col_character(),
##   ..   SeniorCitizen = col_double(),
##   ..   PARTNER = col_character(),
##   ..   Dependents = col_character(),
##   ..   tenure = col_double(),
##   ..   PhoneService = col_character(),
##   ..   MultipleLines = col_character(),
##   ..   InternetService = col_character(),
##   ..   OnlineSecurity = col_character(),
##   ..   OnlineBackup = col_character(),
##   ..   DeviceProtection = col_character(),
##   ..   TECHSUPPORT = col_character(),
##   ..   StreamingTV = col_character(),
##   ..   StreamingMovies = col_character(),
##   ..   Contract = col_character(),
##   ..   PaperlessBilling = col_character(),
##   ..   PaymentMethod = col_character(),
##   ..   MonthlyCharges = col_double(),
##   ..   TotalCharges = col_double(),
##   ..   Churn = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
sample_n(mtn_customers_df, 10)
```

```
## # A tibble: 10 x 21
##    custo~1 GENDER Senio~2 PARTNER Depen~3 tenure Phone~4 Multi~5 Inter~6 Onlin~7
##    <chr>   <chr>    <dbl> <chr>   <chr>    <dbl> <chr>   <chr>   <chr>   <chr>
##  1 5118-M~ Female       0 Yes     No          48 Yes     Yes     Fiber ~ No
##  2 9000-P~ Female       1 Yes     No          60 No      No pho~ DSL     No
##  3 1125-S~ Female       1 No      No          49 Yes     No      DSL     No
##  4 7526-B~ Male         0 No      No          12 Yes     Yes     Fiber ~ No
##  5 5914-X~ Male         0 Yes     No          72 Yes     Yes     Fiber ~ Yes
##  6 7120-R~ Male         0 No      No           1 Yes     Yes     Fiber ~ No
##  7 5939-S~ Male         0 Yes     Yes         48 Yes     Yes     No      No int~
##  8 9530-E~ Male         0 No      No          11 Yes     Yes     DSL     No
##  9 6413-X~ Male         0 Yes     Yes         17 Yes     No      Fiber ~ Yes
## 10 7503-M~ Female       1 Yes     No          72 Yes     Yes     DSL     Yes
## # ... with 11 more variables: OnlineBackup <chr>, DeviceProtection <chr>,
## #   TECHSUPPORT <chr>, StreamingTV <chr>, StreamingMovies <chr>,
## #   Contract <chr>, PaperlessBilling <chr>, PaymentMethod <chr>,
## #   MonthlyCharges <dbl>, TotalCharges <dbl>, Churn <chr>, and abbreviated
## #   variable names 1: customerID, 2: SeniorCitizen, 3: Dependents,
## #   4: PhoneService, 5: MultipleLines, 6: InternetService, 7: OnlineSecurity
```

```
dim(mtn_customers_df)
```

```
## [1] 7050   21
```

2.2. standardise the data Convert columns names to lowercase and strip leading and ending spaces

```
names(mtn_customers_df) <- tolower(names(mtn_customers_df))
names(mtn_customers_df) <- trimws(names(mtn_customers_df), which="both")
head(mtn_customers_df)
```

```
## # A tibble: 6 x 21
##   custom~1 gender senio~2 partner depen~3 tenure phone~4 multi~5 inter~6 onlin~7
##   <chr>    <chr>    <dbl> <chr>   <chr>    <dbl> <chr>   <chr>   <chr>   <chr>
## 1 7590-VH~ Female       0 Yes     No           1 No      No pho~ DSL     No
## 2 5575-GN~ Male         0 No      No          34 Yes     No      DSL     Yes
## 3 3668-QP~ Male         0 No      No           2 Yes     No      DSL     Yes
## 4 7795-CF~ Male         0 No      No          45 No      No pho~ DSL     Yes
## 5 9237-HQ~ Female       0 No      No           2 Yes     No      Fiber ~ No
## 6 9305-CD~ Female       0 No      No           8 Yes     Yes     Fiber ~ No
## # ... with 11 more variables: onlinebackup <chr>, deviceprotection <chr>,
## #   techsupport <chr>, streamingtv <chr>, streamingmovies <chr>,
## #   contract <chr>, paperlessbilling <chr>, paymentmethod <chr>,
## #   monthlycharges <dbl>, totalcharges <dbl>, churn <chr>, and abbreviated
## #   variable names 1: customerid, 2: seniorcitizen, 3: dependents,
## #   4: phoneservice, 5: multiplelines, 6: internetservice, 7: onlinesecurity
```

2.3 Dealing with missing data Check for missing values in the data and remove or replace them i.e with mean of values

```
#remove the missing values since they are not many
dim(mtn_customers_df)
```

```
## [1] 7050   21
```

```
colSums(is.na(mtn_customers_df))
```

```
##       customerid           gender    seniorcitizen          partner
##                0                1                3               12
##       dependents           tenure     phoneservice    multiplelines
##               10               11               15               17
##  internetservice   onlinesecurity     onlinebackup deviceprotection
##               16               16               15               14
##      techsupport      streamingtv  streamingmovies         contract
##               13               13               12               12
## paperlessbilling    paymentmethod   monthlycharges     totalcharges
##               12               12               12               23
##            churn
##               12
```

```
mtn_customers_df <- na.omit(mtn_customers_df)
#check for missing data after removal of missingdata
dim(mtn_customers_df)
```

```
## [1] 7010   21
```

```
colSums(is.na(mtn_customers_df))
```

```
##       customerid           gender    seniorcitizen          partner
##                0                0                0                0
##       dependents           tenure     phoneservice    multiplelines
##                0                0                0                0
##  internetservice   onlinesecurity     onlinebackup deviceprotection
##                0                0                0                0
##      techsupport      streamingtv  streamingmovies         contract
##                0                0                0                0
## paperlessbilling    paymentmethod   monthlycharges     totalcharges
##                0                0                0                0
##            churn
##                0
```

```
#check for duplicates
dim(mtn_customers_df)
```

2.4 Dealing with duplicated entry

```
## [1] 7010   21
```

```
mtn_customers_df[duplicated(mtn_customers_df),]
```

```
## # A tibble: 7 x 21
##   custom~1 gender senio~2 partner depen~3 tenure phone~4 multi~5 inter~6 onlin~7
##   <chr>    <chr>    <dbl> <chr>   <chr>    <dbl> <chr>   <chr>   <chr>   <chr>
## 1 6876-AD~ Male         0 No      Yes          1 Yes     No      DSL     No
## 2 1427-VE~ Female       0 Yes     No          56 Yes     No      Fiber ~ No
## 3 3967-KX~ Male         0 Yes     No          72 Yes     Yes     DSL     Yes
## 4 3967-KX~ Male         0 Yes     No          72 Yes     Yes     DSL     Yes
## 5 2314-TN~ Female       0 Yes     Yes         72 No      No pho~ DSL     Yes
## 6 2314-TN~ Female       0 Yes     Yes         72 No      No pho~ DSL     Yes
## 7 4501-VC~ Male         0 No      No          26 No      No pho~ DSL     No
## # ... with 11 more variables: onlinebackup <chr>, deviceprotection <chr>,
## #   techsupport <chr>, streamingtv <chr>, streamingmovies <chr>,
## #   contract <chr>, paperlessbilling <chr>, paymentmethod <chr>,
## #   monthlycharges <dbl>, totalcharges <dbl>, churn <chr>, and abbreviated
## #   variable names 1: customerid, 2: seniorcitizen, 3: dependents,
## #   4: phoneservice, 5: multiplelines, 6: internetservice, 7: onlinesecurity
```

```
mtn_customers_df <- mtn_customers_df[!duplicated(mtn_customers_df),]
dim(mtn_customers_df)
```

```
## [1] 7003   21
```

```
#remove the customerid columns which is unique
unique_values_df <- mtn_customers_df
unique_values_df <- select(unique_values_df, -c("customerid", "tenure", "monthlycharges", "totalcharges
apply(unique_values_df, 2, table)
```

2.5 Checking for number of unique values in each column

```
## $gender
##
## Female   Male
##   3463   3540
##
## $seniorcitizen
##
##    0    1
## 5866 1137
##
## $partner
##
##   No  Yes
## 3624 3379
##
## $dependents
##
##   No  Yes
## 4911 2092
##
## $phoneservice
##
##   No  Yes
##  678 6325
##
## $multiplelines
##
##               No No phone service              Yes
##             3372              678             2953
##
## $internetservice
##
##          DSL Fiber optic          No
##         2407        3084        1512
##
## $onlinesecurity
##
##                   No No internet service                 Yes
##                 3485                1512                2006
##
```

```
## $onlinebackup
##
##                  No No internet service                 Yes
##                3071                1512                2420
##
## $deviceprotection
##
##                  No No internet service                 Yes
##                3080                1512                2411
##
## $techsupport
##
##                  No No internet service                 Yes
##                3459                1512                2032
##
## $streamingtv
##
##                  No No internet service                 Yes
##                2797                1512                2694
##
## $streamingmovies
##
##                  No No internet service                 Yes
##                2769                1512                2722
##
## $contract
##
## Month-to-month       One year       Two year
##           3858           1467           1678
##
## $paperlessbilling
##
##   No  Yes
## 2849 4154
##
## $paymentmethod
##
## Bank transfer (automatic)   Credit card (automatic)          Electronic check
##                      1536                      1516                      2354
##         Electronic checkk              Mailed check            Mailed checkkk
##                         1                      1594                         2
##
## $churn
##
##   No  Yes
## 5140 1863
```

```
dim(mtn_customers_df)
```

```
## [1] 7003   21
```

2.5.1 Resolving issues with unique values 'payment_method' has values with spelling errors such as "Mailed checkkk" and "Electronic chekk", which created duplicates

```
#resolve issue with payment method values by correcting "Mailed checkkk" to "Mailed check" and "Electro
mtn_customers_df$paymentmethod[mtn_customers_df$paymentmethod == "Mailed checkkk"] <- "Mailed check"
mtn_customers_df$paymentmethod[mtn_customers_df$paymentmethod == "Electronic checkk"] <- "Electronic ch
unique(mtn_customers_df$paymentmethod)
```

```
## [1] "Electronic check"          "Mailed check"
## [3] "Bank transfer (automatic)" "Credit card (automatic)"
```

2.6 Check for outlier for 'tenure', 'monthly__charges' and 'total__charges'

```
#using quantile function to find values below 2.5% and above 97.5%
tenure_lower_bound <- quantile(mtn_customers_df$tenure, 0.025)
tenure_upper_bound <- quantile(mtn_customers_df$tenure, 0.975)
tenure_lower_bound
```

Tenure

```
## 2.5%
##    1
```

```
tenure_upper_bound
```

```
## 97.5%
##    72
```

```
#use the which function to get index for the outliers
tenure_not_outliers <- which(mtn_customers_df$tenure >= tenure_lower_bound & mtn_customers_df$tenure <=
tenure_mtn_customers_df <- mtn_customers_df[tenure_not_outliers,]
```

```
dim(mtn_customers_df)
```

```
## [1] 7003   21
```

```
dim(tenure_mtn_customers_df)
```

```
## [1] 6996   21
```

```
#using quantile function to find values below 2.5% and above 97.5%
month_lower_bound <- quantile(tenure_mtn_customers_df$monthlycharges, 0.01)
month_upper_bound <- quantile(tenure_mtn_customers_df$monthlycharges, 0.99)
month_lower_bound
```

monthly__charges

```
##   1%
## 19.2
```

```
month_upper_bound
```

```
##   99%
## 114.9
```

```
#use the which function to get index for the outliers
month_not_outliers <- which(tenure_mtn_customers_df$monthlycharges >= month_lower_bound & tenure_mtn_cu
month_mtn_customers_df <- tenure_mtn_customers_df[month_not_outliers,]
```

```
dim(mtn_customers_df)
```

```
## [1] 7003   21
```

```
dim(tenure_mtn_customers_df)
```

```
## [1] 6996   21
```

```
dim(month_mtn_customers_df)
```

```
## [1] 6863   21
```

```

#using quantile function to find values below 2.5% and above 97.5%
totalcharge_lower_bound <- quantile(month_mtn_customers_df$totalcharges, 0.01)
totalcharge_upper_bound <- quantile(month_mtn_customers_df$totalcharges, 0.99)
totalcharge_lower_bound
```

monthly_charges

```
##    1%
## 19.95
```

```
totalcharge_upper_bound
```

```
##      99%
## 7855.318
```

```
#use the which function to get index for the outliers
totalcharge_not_outliers <- which(month_mtn_customers_df$totalcharges >= totalcharge_lower_bound & mont
cleaned_mtn_customers_df <- month_mtn_customers_df[totalcharge_not_outliers,]
head(cleaned_mtn_customers_df)
```

```
## # A tibble: 6 x 21
##   custom~1 gender senio~2 partner depen~3 tenure phone~4 multi~5 inter~6 onlin~7
##   <chr>    <chr>    <dbl> <chr>   <chr>    <dbl> <chr>   <chr>   <chr>   <chr>
## 1 7590-VH~ Female       0 Yes     No           1 No      No pho~ DSL     No
## 2 5575-GN~ Male         0 No      No          34 Yes     No      DSL     Yes
## 3 3668-QP~ Male         0 No      No           2 Yes     No      DSL     Yes
## 4 7795-CF~ Male         0 No      No          45 No      No pho~ DSL     Yes
## 5 9237-HQ~ Female       0 No      No           2 Yes     No      Fiber ~ No
## 6 9305-CD~ Female       0 No      No           8 Yes     Yes     Fiber ~ No
## # ... with 11 more variables: onlinebackup <chr>, deviceprotection <chr>,
## #   techsupport <chr>, streamingtv <chr>, streamingmovies <chr>,
## #   contract <chr>, paperlessbilling <chr>, paymentmethod <chr>,
## #   monthlycharges <dbl>, totalcharges <dbl>, churn <chr>, and abbreviated
## #   variable names 1: customerid, 2: seniorcitizen, 3: dependents,
## #   4: phoneservice, 5: multiplelines, 6: internetservice, 7: onlinesecurity
```

```
#check out the data
dim(mtn_customers_df)
```

```
## [1] 7003   21
```

```
dim(tenure_mtn_customers_df)
```

```
## [1] 6996   21
```

```
dim(month_mtn_customers_df)
```

```
## [1] 6863   21
```

```
dim(cleaned_mtn_customers_df)
```

```
## [1] 6727   21
```

```
head(cleaned_mtn_customers_df)
```

```
## # A tibble: 6 x 21
##   custom~1 gender senio~2 partner depen~3 tenure phone~4 multi~5 inter~6 onlin~7
##   <chr>    <chr>    <dbl> <chr>   <chr>    <dbl> <chr>   <chr>   <chr>   <chr>
## 1 7590-VH~ Female       0 Yes     No           1 No      No pho~ DSL     No
## 2 5575-GN~ Male         0 No      No          34 Yes     No      DSL     Yes
## 3 3668-QP~ Male         0 No      No           2 Yes     No      DSL     Yes
## 4 7795-CF~ Male         0 No      No          45 No      No pho~ DSL     Yes
## 5 9237-HQ~ Female       0 No      No           2 Yes     No      Fiber ~ No
## 6 9305-CD~ Female       0 No      No           8 Yes     Yes     Fiber ~ No
## # ... with 11 more variables: onlinebackup <chr>, deviceprotection <chr>,
## #   techsupport <chr>, streamingtv <chr>, streamingmovies <chr>,
## #   contract <chr>, paperlessbilling <chr>, paymentmethod <chr>,
## #   monthlycharges <dbl>, totalcharges <dbl>, churn <chr>, and abbreviated
## #   variable names 1: customerid, 2: seniorcitizen, 3: dependents,
## #   4: phoneservice, 5: multiplelines, 6: internetservice, 7: onlinesecurity
```

3 Research-specific Analysis

```
cleaned_mtn_customers_df %>%
  group_by(churn) %>%
  summarise(count_of_churned = length(churn) ) %>%
  mutate(percent_churned = 100* count_of_churned/sum(count_of_churned))
```

3.1 What percentage of customers from our dataset churned?

```
## # A tibble: 2 x 3
##   churn count_of_churned percent_churned
##   <chr>            <int>           <dbl>
## 1 No                4900            72.8
## 2 Yes               1827            27.2
```

We see that the majority of the customers in this dataset, 73% of the customers are still subscribed to MTN while 26.6% of the customers churned. #### 3.2 How many of each gender male and female churned? both male and female are churning in equal measure

```
cleaned_mtn_customers_df %>%
  group_by(gender, churn)%>%
  summarise(count_of_churn = length(churn))%>%
  mutate(percent_gender_churn = 100*(count_of_churn/sum(count_of_churn)) )
```

```
## `summarise()` has grouped output by 'gender'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 4 x 4
## # Groups:   gender [2]
##   gender churn count_of_churn percent_gender_churn
##   <chr>  <chr>          <int>                <dbl>
## 1 Female No              2405                 72.3
## 2 Female Yes              921                 27.7
## 3 Male   No               2495                      73.4
## 4 Male   Yes               906                      26.6
```

3.3 we investigate the distribution of churn by senior citizen and recording senior citizen leaving are higher rate than young people

```
cleaned_mtn_customers_df %>%
  group_by(seniorcitizen, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_senior_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'seniorcitizen'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 4 x 4
## # Groups:   seniorcitizen [2]
##   seniorcitizen churn count_of_churn percent_of_senior_churn
##           <dbl> <chr>          <int>                   <dbl>
## 1             0 No              4267                    75.9
## 2             0 Yes             1356                    24.1
## 3             1 No               633                    57.3
## 4             1 Yes              471                    42.7
```

3.4 distribution of churn by partner people with partner are less likely to churn compared to people with no partner

```
cleaned_mtn_customers_df %>%
  group_by(partner, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_partner_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'partner'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 4 x 4
## # Groups:   partner [2]
##   partner churn count_of_churn percent_of_partner_churn
##   <chr>   <chr>          <int>                    <dbl>
## 1 No      No              2322                     66.5
## 2 No      Yes             1169                     33.5
## 3 Yes     No              2578                     79.7
## 4 Yes     Yes              658                     20.3
```

3.5 distribution of churn by dependents people with dependents are less likely to churn compared to people with no partner

```
cleaned_mtn_customers_df %>%
  group_by(dependents, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_partner_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'dependents'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 4 x 4
## # Groups:   dependents [2]
##   dependents churn count_of_churn percent_of_partner_churn
##   <chr>      <chr>          <int>                      <dbl>
## 1 No         No              3213                       68.0
## 2 No         Yes             1509                       32.0
## 3 Yes        No              1687                       84.1
## 4 Yes        Yes              318                       15.9
```

3.6 distribution of churn by phone service people with phone service or no phone service have equal probability of churning

```
cleaned_mtn_customers_df %>%
  group_by(phoneservice, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_phoneservice_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'phoneservice'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 4 x 4
## # Groups:   phoneservice [2]
##   phoneservice churn count_of_churn percent_of_phoneservice_churn
##   <chr>        <chr>          <int>                         <dbl>
## 1 No           No               508                          74.9
## 2 No           Yes              170                          25.1
## 3 Yes          No              4392                          72.6
## 4 Yes          Yes             1657                          27.4
```

3.7 distribution of churn by multiple lines people with or without multiple have equal probability of leaving

```
cleaned_mtn_customers_df %>%
  group_by(multiplelines, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_multiple_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'multiplelines'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 6 x 4
## # Groups:   multiplelines [3]
##   multiplelines    churn count_of_churn percent_of_multiple_churn
##   <chr>            <chr>          <int>                     <dbl>
## 1 No               No              2413                      74.6
## 2 No               Yes              822                      25.4
## 3 No phone service No               508                      74.9
## 4 No phone service Yes              170                      25.1
## 5 Yes              No              1979                      70.3
## 6 Yes              Yes              835                      29.7
```

3.8 distribution of churn by internet service people with DSL or No internet service are less likely to leave, while people with fiber optic have equal probability of leaving

```
cleaned_mtn_customers_df %>%
  group_by(internetservice, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_internetservice_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'internetservice'. You can override using
## the `.groups` argument.
```

```
## # A tibble: 6 x 4
## # Groups:   internetservice [3]
##   internetservice churn count_of_churn percent_of_internetservice_churn
##   <chr>           <chr>          <int>                            <dbl>
## 1 DSL             No              1943                            80.9
## 2 DSL             Yes              458                            19.1
## 3 Fiber optic     No              1664                            56.5
## 4 Fiber optic     Yes             1282                            43.5
## 5 No              No              1293                            93.7
## 6 No              Yes               87                             6.30
```

3.9 distribution of churn by online security people with online security or no internet service are less likely to leave but people with no online security have equal probability of leaving

```
cleaned_mtn_customers_df %>%
  group_by(onlinesecurity, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_onlinesecurity_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'onlinesecurity'. You can override using
## the `.groups` argument.
```

```
## # A tibble: 6 x 4
## # Groups:   onlinesecurity [3]
##   onlinesecurity      churn count_of_churn percent_of_onlinesecurity_churn
##   <chr>               <chr>          <int>                           <dbl>
## 1 No                  No              2010                           58.0
## 2 No                  Yes             1455                           42.0
## 3 No internet service No              1293                           93.7
## 4 No internet service Yes               87                            6.30
## 5 Yes                 No              1597                           84.9
## 6 Yes                 Yes              285                           15.1
```

3.10 distribution of churn by online backup people with online backup or no internet service are less likely to leave but people with no online backup have equal probability of leaving

```
cleaned_mtn_customers_df %>%
  group_by(onlinebackup, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_onlinebackup_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'onlinebackup'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 6 x 4
## # Groups:   onlinebackup [3]
##   onlinebackup        churn count_of_churn percent_of_onlinebackup_churn
##   <chr>               <chr>          <int>                         <dbl>
## 1 No                  No              1836                         59.9
## 2 No                  Yes             1227                         40.1
## 3 No internet service No              1293                         93.7
## 4 No internet service Yes               87                          6.30
## 5 Yes                 No              1771                         77.5
## 6 Yes                 Yes              513                         22.5
```

3.11 distribution of churn by device protection people with no internet service are very less likely to leave and people WITH or NO device protection are also less likely to leave

```
cleaned_mtn_customers_df %>%
  group_by(deviceprotection, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_devprotection_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'deviceprotection'. You can override using
## the `.groups` argument.
```

```
## # A tibble: 6 x 4
## # Groups:   deviceprotection [3]
##   deviceprotection    churn count_of_churn percent_of_devprotection_churn
##   <chr>               <chr>          <int>                          <dbl>
## 1 No                  No              1867                          60.8
## 2 No                  Yes             1205                          39.2
## 3 No internet service No              1293                          93.7
## 4 No internet service Yes               87                           6.30
## 5 Yes                 No              1740                          76.5
## 6 Yes                 Yes              535                          23.5
```

3.12 distribution of churn by tech support people with tech sopport or no internet service are less likely to leave but people with no techsupport have equal probability of leaving

```
cleaned_mtn_customers_df %>%
  group_by(techsupport, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_techsupport_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'techsupport'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 6 x 4
## # Groups:   techsupport [3]
##   techsupport         churn count_of_churn percent_of_techsupport_churn
##   <chr>               <chr>          <int>                        <dbl>
## 1 No                  No              2003                        58.2
## 2 No                  Yes             1440                        41.8
## 3 No internet service No              1293                        93.7
## 4 No internet service Yes               87                         6.30
## 5 Yes                 No              1604                        84.2
## 6 Yes                 Yes              300                        15.8
```

3.13 distribution of churn by streaming tv people with no internet service are more likely to stay compare to people with or having no streaming tv service

```
cleaned_mtn_customers_df %>%
  group_by(streamingtv, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_streamigtv_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'streamingtv'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 6 x 4
## # Groups:   streamingtv [3]
##   streamingtv         churn count_of_churn percent_of_streamigtv_churn
##   <chr>               <chr>          <int>                       <dbl>
## 1 No                  No              1851                       66.4
## 2 No                  Yes              938                       33.6
## 3 No internet service No              1293                       93.7
## 4 No internet service Yes               87                        6.30
## 5 Yes                 No              1756                       68.6
## 6 Yes                 Yes              802                       31.4
```

3.14 distribution of churn by streaming movies people with no internet service are more likely to stay compare to people with or having no streaming movies service

```
cleaned_mtn_customers_df %>%
  group_by(streamingmovies, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_streamigmovies_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'streamingmovies'. You can override using
## the `.groups` argument.
```

```
## # A tibble: 6 x 4
## # Groups:   streamingmovies [3]
##   streamingmovies     churn count_of_churn percent_of_streamigmovies_churn
##   <chr>               <chr>          <int>                           <dbl>
## 1 No                  No              1828                           66.2
## 2 No                  Yes              934                           33.8
## 3 No internet service No              1293                           93.7
## 4 No internet service Yes               87                            6.30
## 5 Yes                 No              1779                           68.8
## 6 Yes                 Yes              806                           31.2
```

3.15 distribution of churn by contract people with long contract 1 or 2 year contract are more likely to stay compared to people with month to month contract

```
cleaned_mtn_customers_df %>%
  group_by(contract, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_contract_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'contract'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 6 x 4
## # Groups:   contract [3]
##   contract       churn count_of_churn percent_of_contract_churn
##   <chr>          <chr>          <int>                     <dbl>
## 1 Month-to-month No              2133                     56.8
## 2 Month-to-month Yes             1621                     43.2
## 3 One year       No              1264                     88.7
## 4 One year       Yes              161                     11.3
## 5 Two year       No              1503                     97.1
## 6 Two year       Yes               45                      2.91
```

3.16 distribution of churn by paperless billing people with no paperless billing are more likely to stay compared to people with paperless billing

```
cleaned_mtn_customers_df %>%
  group_by(paperlessbilling, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_paperless_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'paperlessbilling'. You can override using
## the `.groups` argument.
```

```
## # A tibble: 4 x 4
## # Groups:   paperlessbilling [2]
##   paperlessbilling churn count_of_churn percent_of_paperless_churn
##   <chr>            <chr>          <int>                      <dbl>
## 1 No               No              2269                       83.5
## 2 No               Yes              448                       16.5
## 3 Yes              No              2631                       65.6
## 4 Yes              Yes             1379                       34.4
```

3.17 distribution of churn by payment method people with bank transfer, credit card and mailed check are more likely to stay compared to people with electronic check

```
cleaned_mtn_customers_df %>%
  group_by(paymentmethod, churn)%>%
  summarise(count_of_churn=length(churn),)%>%
  mutate(percent_of_paymentmethod_churn = 100*( count_of_churn/sum(count_of_churn)))
```

```
## `summarise()` has grouped output by 'paymentmethod'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 8 x 4
## # Groups:   paymentmethod [4]
##   paymentmethod             churn count_of_churn percent_of_paymentmethod_churn
##   <chr>                     <chr>          <int>                          <dbl>
## 1 Bank transfer (automatic) No              1220                           82.8
## 2 Bank transfer (automatic) Yes              254                           17.2
## 3 Credit card (automatic)   No              1212                           84.2
## 4 Credit card (automatic)   Yes              228                           15.8
## 5 Electronic check          No              1256                           54.3
## 6 Electronic check          Yes             1059                           45.7
## 7 Mailed check              No              1212                           80.9
## 8 Mailed check              Yes              286                           19.1
```

4. General Analysis

Find the distribution for each aspect of the customer i.e what percentage of customer are male or female or have or dont have internet service

```
#cleaned_mtn_customers_df <- select(cleaned_mtn_customers_df, -c("customerid"))
columns = names(cleaned_mtn_customers_df)
i <- 0
for (column in colnames(cleaned_mtn_customers_df)){
  y <- cleaned_mtn_customers_df %>%
    group_by_at(column)%>%
    summarise(count=length(gender))%>%
    mutate(percent = 100* (count/sum(count) ))
  print(column)
  print(y)
}
```

```
## [1] "customerid"
## # A tibble: 6,727 x 3
##    customerid count percent
##    <chr>      <int>   <dbl>
##  1 0002-ORFBO     1  0.0149
##  2 0003-MKNFE     1  0.0149
##  3 0004-TLHLJ     1  0.0149
##  4 0011-IGKFF     1  0.0149
##  5 0013-EXCHZ     1  0.0149
##  6 0013-MHZWF     1  0.0149
##  7 0014-BMAQU     1  0.0149
##  8 0015-UOCOJ     1  0.0149
##  9 0016-QLJIS     1  0.0149
## 10 0017-DINOC     1  0.0149
## # ... with 6,717 more rows
## [1] "gender"
## # A tibble: 2 x 3
##   gender count percent
##   <chr>  <int>   <dbl>
## 1 Female  3326    49.4
## 2 Male    3401    50.6
## [1] "seniorcitizen"
## # A tibble: 2 x 3
##   seniorcitizen count percent
##           <dbl> <int>   <dbl>
## 1             0  5623    83.6
## 2             1  1104    16.4
## [1] "partner"
## # A tibble: 2 x 3
##   partner count percent
##   <chr>   <int>   <dbl>
## 1 No       3491    51.9
## 2 Yes      3236    48.1
## [1] "dependents"
## # A tibble: 2 x 3
##   dependents count percent
##   <chr>      <int>   <dbl>
## 1 No          4722    70.2
## 2 Yes         2005    29.8
## [1] "tenure"
## # A tibble: 72 x 3
##    tenure count percent
##     <dbl> <int>   <dbl>
##  1      1   535    7.95
##  2      2   237    3.52
##  3      3   196    2.91
##  4      4   173    2.57
##  5      5   133    1.98
##  6      6   105    1.56
##  7      7   128    1.90
##  8      8   122    1.81
##  9      9   117    1.74
## 10     10   115    1.71
## # ... with 62 more rows
## [1] "phoneservice"
## # A tibble: 2 x 3
##   phoneservice count percent
##   <chr>        <int>   <dbl>
## 1 No             678    10.1
## 2 Yes           6049    89.9
## [1] "multiplelines"
## # A tibble: 3 x 3
##   multiplelines    count percent
##   <chr>            <int>   <dbl>
## 1 No                3235    48.1
## 2 No phone service   678    10.1
## 3 Yes               2814    41.8
## [1] "internetservice"
## # A tibble: 3 x 3
##   internetservice count percent
##   <chr>           <int>   <dbl>
## 1 DSL              2401    35.7
## 2 Fiber optic      2946    43.8
## 3 No               1380    20.5
## [1] "onlinesecurity"
## # A tibble: 3 x 3
##   onlinesecurity      count percent
##   <chr>               <int>   <dbl>
## 1 No                   3465    51.5
## 2 No internet service  1380    20.5
## 3 Yes                  1882    28.0
## [1] "onlinebackup"
## # A tibble: 3 x 3
##   onlinebackup        count percent
##   <chr>               <int>   <dbl>
## 1 No                   3063    45.5
## 2 No internet service  1380    20.5
## 3 Yes                  2284    34.0
## [1] "deviceprotection"
## # A tibble: 3 x 3
##   deviceprotection    count percent
##   <chr>               <int>   <dbl>
## 1 No                   3072    45.7
## 2 No internet service  1380    20.5
## 3 Yes                  2275    33.8
## [1] "techsupport"
## # A tibble: 3 x 3
##   techsupport         count percent
##   <chr>               <int>   <dbl>
## 1 No                   3443    51.2
## 2 No internet service  1380    20.5
## 3 Yes                  1904    28.3
## [1] "streamingtv"
## # A tibble: 3 x 3
##   streamingtv         count percent
##   <chr>               <int>   <dbl>
## 1 No                   2789    41.5
## 2 No internet service  1380    20.5
## 3 Yes                  2558    38.0
## [1] "streamingmovies"
## # A tibble: 3 x 3
##   streamingmovies     count percent
##   <chr>               <int>   <dbl>
## 1 No                   2762    41.1
## 2 No internet service  1380    20.5
## 3 Yes                  2585    38.4
## [1] "contract"
## # A tibble: 3 x 3
##   contract       count percent
##   <chr>          <int>   <dbl>
## 1 Month-to-month  3754    55.8
## 2 One year        1425    21.2
## 3 Two year        1548    23.0
## [1] "paperlessbilling"
## # A tibble: 2 x 3
##   paperlessbilling count percent
##   <chr>            <int>   <dbl>
## 1 No                2717    40.4
## 2 Yes               4010    59.6
## [1] "paymentmethod"
## # A tibble: 4 x 3
##   paymentmethod             count percent
##   <chr>                     <int>   <dbl>
## 1 Bank transfer (automatic)  1474    21.9
## 2 Credit card (automatic)    1440    21.4
## 3 Electronic check           2315    34.4
## 4 Mailed check               1498    22.3
## [1] "monthlycharges"
## # A tibble: 1,508 x 3
##    monthlycharges count percent
##             <dbl> <int>   <dbl>
##  1           19.2    13   0.193
##  2           19.2    15   0.223
##  3           19.3    20   0.297
##  4           19.4    25   0.372
##  5           19.4    27   0.401
##  6           19.4    22   0.327
##  7           19.5    28   0.416
##  8           19.6    32   0.476
##  9           19.6    35   0.520
## 10           19.6    35   0.520
## # ... with 1,498 more rows
## [1] "totalcharges"
## # A tibble: 6,296 x 3
##    totalcharges count percent
##           <dbl> <int>   <dbl>
##  1         20.0     4  0.0595
##  2         20       3  0.0446
##  3         20.0     8  0.119
##  4         20.1      3  0.0446
##  5         20.2      6  0.0892
##  6         20.2     11  0.164
##  7         20.2      6  0.0892
##  8         20.3      5  0.0743
##  9         20.4      4  0.0595
## 10         20.4      4  0.0595
## # ... with 6,286 more rows
## [1] "churn"
## # A tibble: 2 x 3
##   churn count percent
##   <chr> <int>   <dbl>
## 1 No     4900    72.8
## 2 Yes    1827    27.2
```

5. Summary of Findings

Based on the results of the analysis, the following conclusions were arrived at:

1. There is no significant difference in churn rate between male and female subscribers. So this is not an area management needs to worry about.
2. Majority of the customers are not senior citizens so this makes this dataset biased and hard to identify whether being a senior citizen affects churn rate. 3.Not having a partner increases the likelihood of churning.
3. Not having dependents increases the likelihood of churning.
4. generally customers with No internet service are more likely to stay on the network followed by customer with the following service ("phoneservice", "internetservice", "onlinesecurity","onlinebackup","deviceprotection", "techsupport" "streamingtv", "streamingmovies"). customer with No those services are more likely to leave
5. customers with no paperless billing are more likely to stay compared to people with paperless billing
6. customers with bank transfer, credit card and mailed check are more likely to stay compared to people with electronic check
7. customers with long contract 1 or 2 year contract are more likely to stay compared to people with month to month contract
8. having multiple lines doesnt influence the customer staying or leaving
9. majority of customer are on short contract
10. customers with DSL internet service are the least

6.Recommendations

In order to create an effective customer retention program, management should take the following measures: 1. Focus more on meeting the needs of non-senior citizens. 2. Focus more on having customers that have partners and/or dependents since these people are less likely to churn. Alternatively, management can come up with services specifically designed for customers without parters and/or dependents. This would require additional research. 3. make initiative for people to have DSL internet service 4. make initiative to have customer subscribe to long term contracts 5. make initiative for customers to subscribes to different services like phoneservice", "internetservice", "onlinesecurity","onlinebackup","deviceprotection", "techsupport" "streamingtv", "streamingmovies")

7. Challenging your Solution

a). Did we have the right data? Do we need other data to answer our question?

As far as I can tell, we had the right data. However, more data is still needed, particularly those with more customers who churned so we can have a better understanding of why they might have churned.

8 Did we have the right question?

Yes, we did.