Predicting Sales Win or Lose

Herman Toeante

February 12, 2019

How do you know which deals will close? You've worked your territory and leads. Demonstrated the value of your product or service to your customers' businesses and it should be a done deal. But is it? All too often, determining which customers will buy is a guessing game. You have pipeline reports, regional sales figures and wins and losses that you could analyze. Unlock the information in those sources and you'll unlock more revenue and more satisfied customers. Better understanding of sales pipeline can help any sales team organization can expect win or lose based on data. In this project, I am going to be the sales manager at an automotive supply company. Any B2B company like GE, AMAZON, GOOGLE, ADP, etc can use the same approach that I demonstrated on this project to their sales team. As a manager, I'm trying to assess a sales execution issue. We have not been able to convert enough opportunities lately. As a start, we start by loading the required packages and data for this project.

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 3.5.2
```

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.5.2
```

```
sales_win_loss <- read_csv("DATA/WA_Fn-UseC_-Sales-Win-Loss.csv")
```

Data for this project is publicly available from Wattson Analytic sample data. (https://community.watsonanalytics.com/wp-content/uploads/2015/04/WA_Fn-UseC_-Sales-Win-Loss.csv) The csv file contains 78K row of data and 19 columns. Each row is assigned a unique sales opportunity ID. The dependent variable is 'Opportunity Result' column with values of either 'Won' or 'Loss' There are several independent variables from the sample data that I can use such as: 'Supplies Group', 'Region', 'Route To Market', 'Elapsed Days In Sales Stage', 'Opportunity Result', 'Sales Stage Change Count', 'Total Days Identified Through Closing', 'Total Days Identified Through Qualified', 'Opportunity Amount USD', 'Client Size By Revenue', 'Client Size By Employee Count', 'Revenue From Client Past Two Years', and 'Competitor Type.' We use 'glimpse' and 'head' to understand the data structure of the data.

```
glimpse(sales_win_loss, give.attr = FALSE)
```

```
## Observations: 78,025
## Variables: 19
## $ `Opportunity Number`                    <int> 1641984, 1658010, 16...
## $ `Supplies Subgroup`                     <chr> "Exterior Accessorie...
## $ `Supplies Group`                        <chr> "Car Accessories", "...
## $ Region                                  <chr> "Northwest", "Pacifi...
## $ `Route To Market`                       <chr> "Fields Sales", "Res...
## $ `Elapsed Days In Sales Stage`           <int> 76, 63, 24, 16, 69, ...
## $ `Opportunity Result`                    <chr> "Won", "Loss", "Won"...
## $ `Sales Stage Change Count`              <int> 13, 2, 7, 5, 11, 3, ...
## $ `Total Days Identified Through Closing` <int> 104, 163, 82, 124, 9...
```

```
## $ `Total Days Identified Through Qualified` <int> 101, 163, 82, 124, 1...
## $ `Opportunity Amount USD`                  <int> 0, 0, 7750, 0, 69756...
## $ `Client Size By Revenue`                  <int> 5, 3, 1, 1, 1, 5, 4,...
## $ `Client Size By Employee Count`           <int> 5, 5, 1, 1, 1, 1, 5,...
## $ `Revenue From Client Past Two Years`      <int> 0, 0, 0, 0, 0, 0, 0,...
## $ `Competitor Type`                         <chr> "Unknown", "Unknown"...
## $ `Ratio Days Identified To Total Days`     <dbl> 0.696360, 0.000000, ...
## $ `Ratio Days Validated To Total Days`      <dbl> 0.113985, 1.000000, ...
## $ `Ratio Days Qualified To Total Days`      <dbl> 0.154215, 0.000000, ...
## $ `Deal Size Category`                      <int> 1, 1, 1, 1, 4, 5, 2,...
```

```
summary(sales_win_loss)
```

```
##  Opportunity Number Supplies Subgroup  Supplies Group
##  Min.   : 1641984   Length:78025       Length:78025
##  1st Qu.: 6900423   Class :character   Class :character
##  Median : 7545569   Mode  :character   Mode  :character
##  Mean   : 7653429
##  3rd Qu.: 8228329
##  Max.   :10094266
##     Region          Route To Market    Elapsed Days In Sales Stage
##  Length:78025       Length:78025       Min.   :  0.0
##  Class :character   Class :character   1st Qu.: 19.0
##  Mode  :character   Mode  :character   Median : 43.0
##                                        Mean   : 43.6
##                                        3rd Qu.: 65.0
##                                        Max.   :210.0
##  Opportunity Result Sales Stage Change Count
##  Length:78025       Min.   : 1.000
##  Class :character   1st Qu.: 2.000
##  Mode  :character   Median : 3.000
##                     Mean   : 2.956
##                     3rd Qu.: 3.000
##                     Max.   :23.000
##  Total Days Identified Through Closing
##  Min.   :  0.00
##  1st Qu.:  4.00
##  Median : 12.00
##  Mean   : 16.73
##  3rd Qu.: 24.00
##  Max.   :208.00
##  Total Days Identified Through Qualified Opportunity Amount USD
##  Min.   :  0.00                          Min.   :      0
##  1st Qu.:  4.00                          1st Qu.:  15000
##  Median : 12.00                          Median :  49000
##  Mean   : 16.31                          Mean   :  91637
##  3rd Qu.: 24.00                          3rd Qu.: 105099
##  Max.   :208.00                          Max.   :1000000
##  Client Size By Revenue Client Size By Employee Count
##  Min.   :1.00           Min.   :1.000
##  1st Qu.:1.00           1st Qu.:1.000
##  Median :1.00           Median :1.000
##  Mean   :1.62           Mean   :1.604
##  3rd Qu.:1.00           3rd Qu.:1.000
```

```
##  Max.   :5.00           Max.   :5.000
##  Revenue From Client Past Two Years Competitor Type
##  Min.   :0.0000                     Length:78025
##  1st Qu.:0.0000                     Class :character
##  Median :0.0000                     Mode  :character
##  Mean   :0.3033
##  3rd Qu.:0.0000
##  Max.   :4.0000
##  Ratio Days Identified To Total Days Ratio Days Validated To Total Days
##  Min.   :0.0000                      Min.   :0.0000
##  1st Qu.:0.0000                      1st Qu.:0.0000
##  Median :0.0000                      Median :0.4480
##  Mean   :0.2031                      Mean   :0.4883
##  3rd Qu.:0.1972                      3rd Qu.:1.0000
##  Max.   :1.0000                      Max.   :1.0000
##  Ratio Days Qualified To Total Days Deal Size Category
##  Min.   :0.0000                     Min.   :1.000
##  1st Qu.:0.0000                     1st Qu.:2.000
##  Median :0.0000                     Median :3.000
##  Mean   :0.1850                     Mean   :3.437
##  3rd Qu.:0.1886                     3rd Qu.:5.000
##  Max.   :1.0000                     Max.   :7.000
```

```
head(sales_win_loss[, 1:6])
```

```
## # A tibble: 6 x 6
##   `Opportunity Nu~ `Supplies Subgr~ `Supplies Group` Region
##              <int> <chr>            <chr>            <chr>
## 1          1641984 Exterior Access~ Car Accessories  North~
## 2          1658010 Exterior Access~ Car Accessories  Pacif~
## 3          1674737 Motorcycle Parts Performance & N~ Pacif~
## 4          1675224 Shelters & RV    Performance & N~ Midwe~
## 5          1689785 Exterior Access~ Car Accessories  Pacif~
## 6          1692390 Shelters & RV    Performance & N~ Pacif~
## # ... with 2 more variables: `Route To Market` <chr>, `Elapsed Days In
## #   Sales Stage` <int>
```

```
head(sales_win_loss[, 7:13])
```

```
## # A tibble: 6 x 7
##   `Opportunity Re~ `Sales Stage Ch~ `Total Days Ide~ `Total Days Ide~
##   <chr>                       <int>            <int>            <int>
## 1 Won                            13              104              101
## 2 Loss                            2              163              163
## 3 Won                             7               82               82
## 4 Loss                            5              124              124
## 5 Loss                           11               91               13
## 6 Loss                            3              114                0
## # ... with 3 more variables: `Opportunity Amount USD` <int>, `Client Size
## #   By Revenue` <int>, `Client Size By Employee Count` <int>
```

```
head(sales_win_loss[, 14:19])
```

```
## # A tibble: 6 x 6
##   `Revenue From C~ `Competitor Typ~ `Ratio Days Ide~ `Ratio Days Val~
##              <int> <chr>                       <dbl>            <dbl>
## 1                0 Unknown                     0.696         0.114
## 2                0 Unknown                     0             1
## 3                0 Unknown                     1             0
## 4                0 Known                       1             0
## 5                0 Unknown                     0             0.141
## 6                0 Unknown                     0             0.000877
## # ... with 2 more variables: `Ratio Days Qualified To Total Days` <dbl>,
## #   `Deal Size Category` <int>
```

We have to make sure there is no missing data by checking for missing values of the dataset

```
map_dbl(sales_win_loss, ~sum(is.na(.)))
```

```
##                      Opportunity Number
##                                       0
##                       Supplies Subgroup
##                                       0
##                          Supplies Group
##                                       0
##                                  Region
##                                       0
##                         Route To Market
##                                       0
##             Elapsed Days In Sales Stage
##                                       0
##                      Opportunity Result
##                                       0
##                Sales Stage Change Count
##                                       0
##   Total Days Identified Through Closing
##                                       0
## Total Days Identified Through Qualified
##                                       0
##                   Opportunity Amount USD
##                                       0
##                   Client Size By Revenue
##                                       0
##           Client Size By Employee Count
##                                       0
##      Revenue From Client Past Two Years
##                                       0
##                          Competitor Type
##                                       0
##     Ratio Days Identified To Total Days
##                                       0
##      Ratio Days Validated To Total Days
##                                       0
```

```
##      Ratio Days Qualified To Total Days
##                                       0
##                       Deal Size Category
##                                       0
```

The next step is setting the standard theme for the charts to theme_minimal with legend set at the bottom of the chart.

```
theme_set(theme_minimal() + theme(legend.position = "bottom"))
```

For better data visualization on the chart and easier to remember, we have to rename the columns with long name.

```
colnames(sales_win_loss) <- c("ID","SuppliesSubgroup","SuppliesGroup","Region", "Route",
                              "ElapsedDays", "Result","SalesStageCount",
                              "TotalDaysClosing","TotalDaysQualified",
                              "Opportunity","ClientSizeRev","ClientSizeCount",
                              "Revenue","Competitor","RDaysIdentified",
                              "RDaysValidated","RDaysQualified",
                              "DealSize")
```

Moreover, we have to make several assumptions to translate the categorical columns into meaningful information.

```
  sales_win_loss <- sales_win_loss %>%
    mutate(ClientSizeRev2 = case_when(
      ClientSizeRev == 1 ~ "ClientRev<$1M",
      ClientSizeRev == 2 ~ "$1M<=ClientRev<$10M",
      ClientSizeRev == 3 ~ "$10M<=ClientRev<$50M",
      ClientSizeRev == 4 ~ "$50M<=ClientRev<$100M",
      ClientSizeRev == 5 ~ "ClientRev>=$100M"))

  sales_win_loss <- sales_win_loss %>%
    mutate(ClientSizeCount2 = case_when(
      ClientSizeCount == 1 ~ "Count<1K",
      ClientSizeCount == 2 ~ "1K<=Count<5K",
      ClientSizeCount == 3 ~ "5K<=Count<10K",
      ClientSizeCount == 4 ~ "10K<=Count<30K",
      ClientSizeCount == 5 ~ "Count>=30K"))

  sales_win_loss <- sales_win_loss %>%
    mutate(Revenue2 = case_when(
      Revenue == 0 ~ "Rev=$0",
      Revenue == 1 ~ "$1<=Rev<$50K",
      Revenue == 2 ~ "$50K<=Rev<$400K",
      Revenue == 3 ~ "$400K<=Rev<$1.5M",
      Revenue == 4 ~ "Rev>=$1.5M"))
```

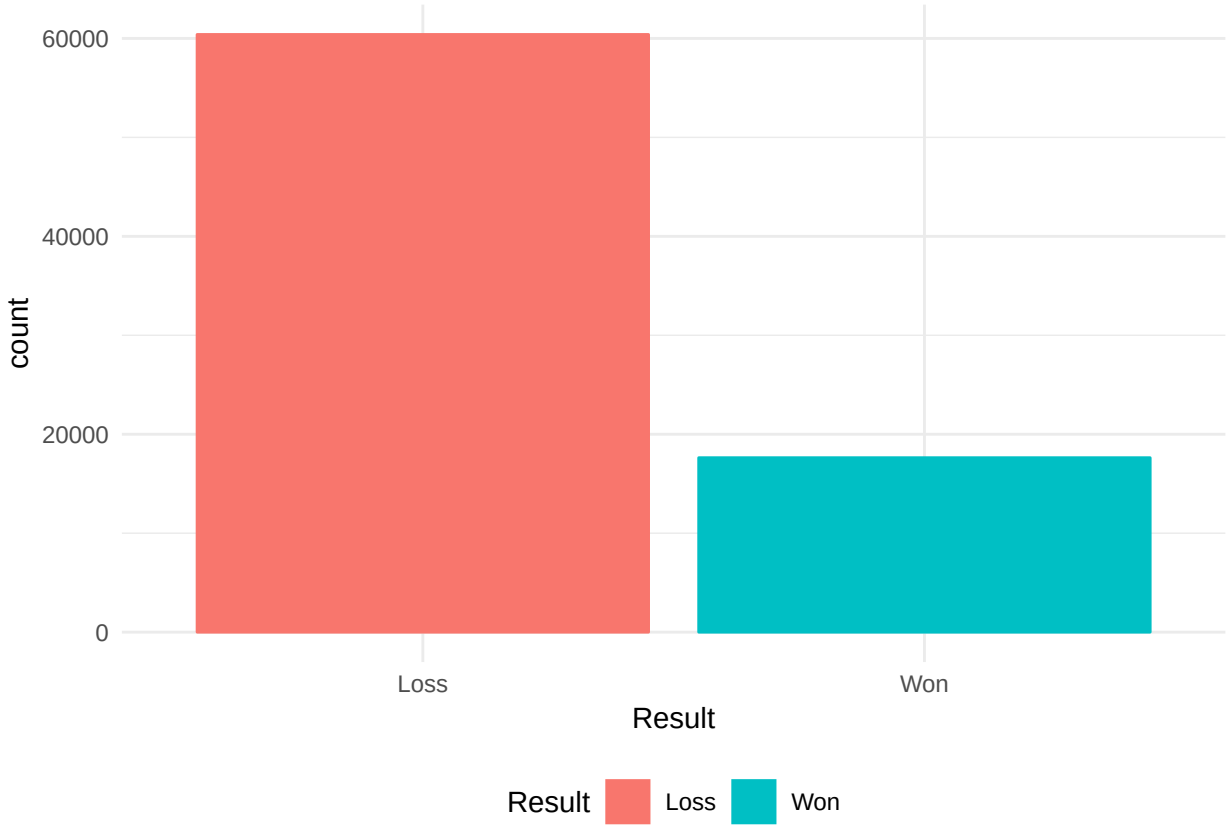
Data Dictionary

VariableID	VarType	VarDescription
ID	Integer	A random number assigned to the opportunity
Supplies Subgroup	Character	Supplies Subgroup
Supplies Group	Character	Supplies Group
Region	Character	Region
Route	Character	Route to market
ElapsedDays	Integer	The number of days between the change in sales stages
Result	Character	A closed opportunity. Values is either won or loss
SalesStageCount	Integer	A count of number of times an opportunity changes sales stages
TotalDayClosing	Integer	Total days from Identified to Gained Agreement/closing
TotalDayQualified	Integer	Total days from Identified to Qualified Agreement
Opportunity	Integer	Sum of line item revenue estimates
ClientSizeRev	Integer	Client size based on annual revenue
ClientSizeCount	Integer	Client size based on number of employees
Revenue	Integer	Revenue from client past two years assuming after the deal is closed
Competitor	Character	An indicator whether or not competitor has been identified
RDaysIdentified	Numeric	Ratio of Identified/Validating over total days
RDaysValidated	Numeric	Ratio of Qualified/Gaining Agreement over total days
RDaysQualified	Numeric	Ratio of Validated/Qualifying over total days
DealSize	Integer	Categorical grouping of the opportunity amount
ClientSizeRev2	Character	Similar to ClienSizeRev with additional revenue range info
ClientSizeCount2	Character	Similar to ClientSizeCount with additional employee size info
Revenue2	Character	Similar to Revenue - adding revenue range

Data Exploration

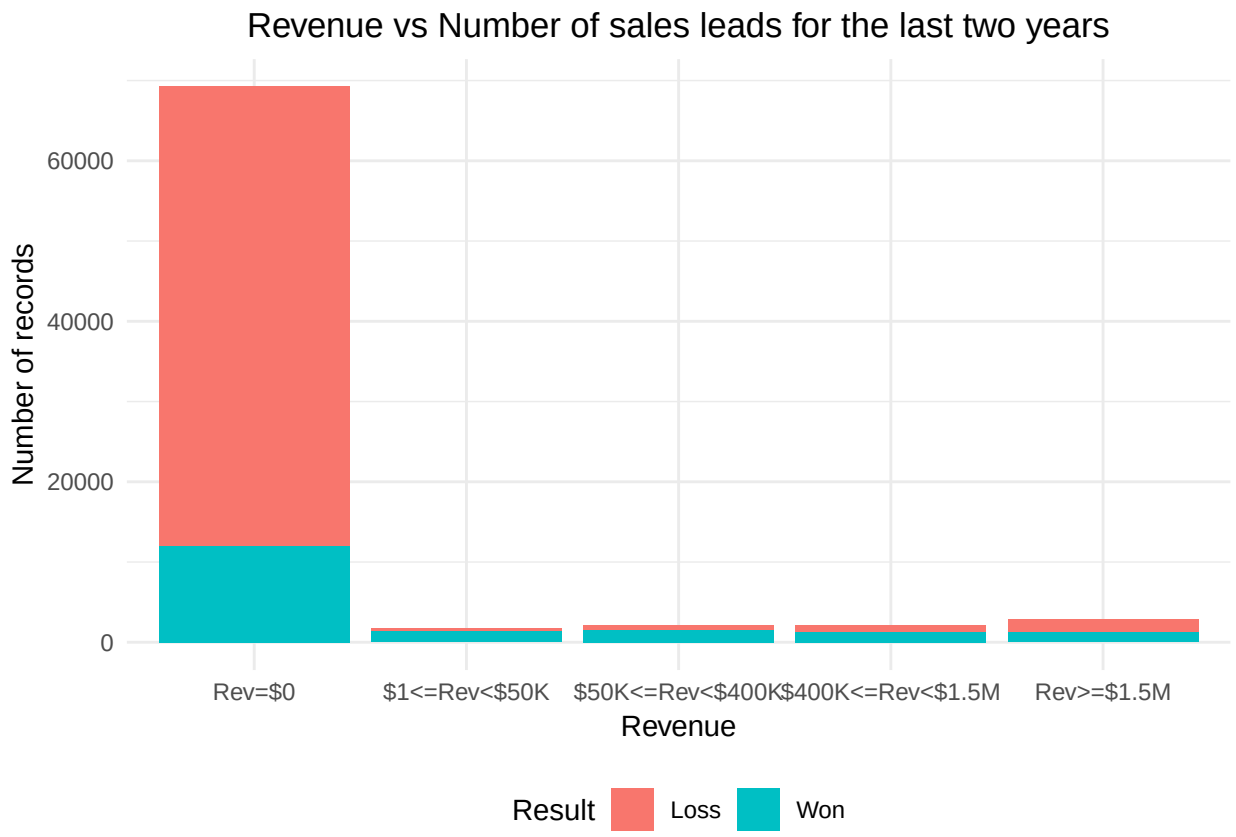
The first chart is to understand the number of sales leads that the company won versus loss in respect to revenue for the last two years.

```
ggplot(sales_win_loss, aes(x = Result, color = Result, fill = Result)) +
  geom_bar()
```



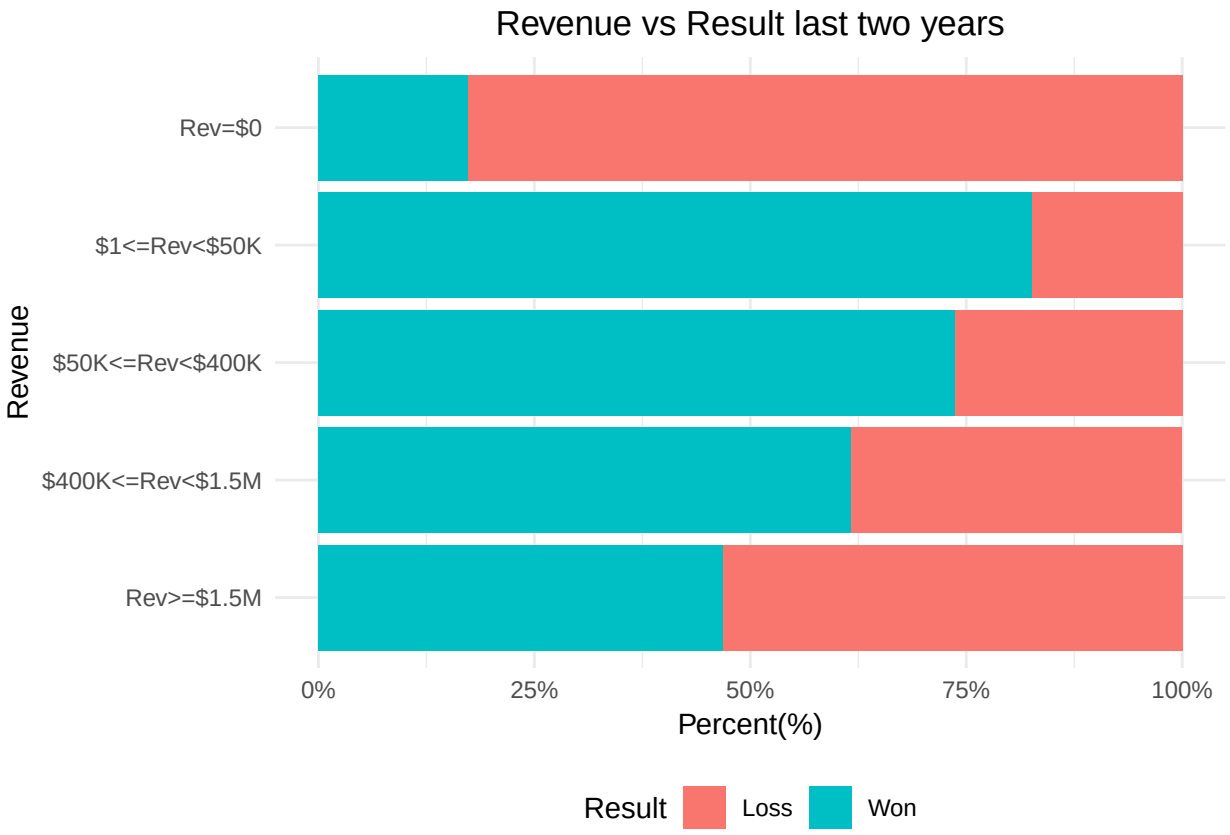
From the chart, we have ~18K records of 'Won' sales leads and ~60K 'Loss" sales leads for the last two years. Using this data, it looks the high number of loss opportunity could be attributed to business condition or we can explore deeper into the data set if there are variables that affecting the loss opportunity.

```
position <- c("Rev=$0", "$1<=Rev<$50K", "$50K<=Rev<$400K",
              "$400K<=Rev<$1.5M", "Rev>=$1.5M")
 ggplot(sales_win_loss, aes(x = Revenue2, fill = Result)) +
  geom_bar() +
  scale_x_discrete(limits = position) +
  xlab("Revenue") +
  ylab("Number of records") +
  ggtitle("Revenue vs Number of sales leads for the last two years") +
  theme(plot.title = element_text(hjust = 0.5))
```



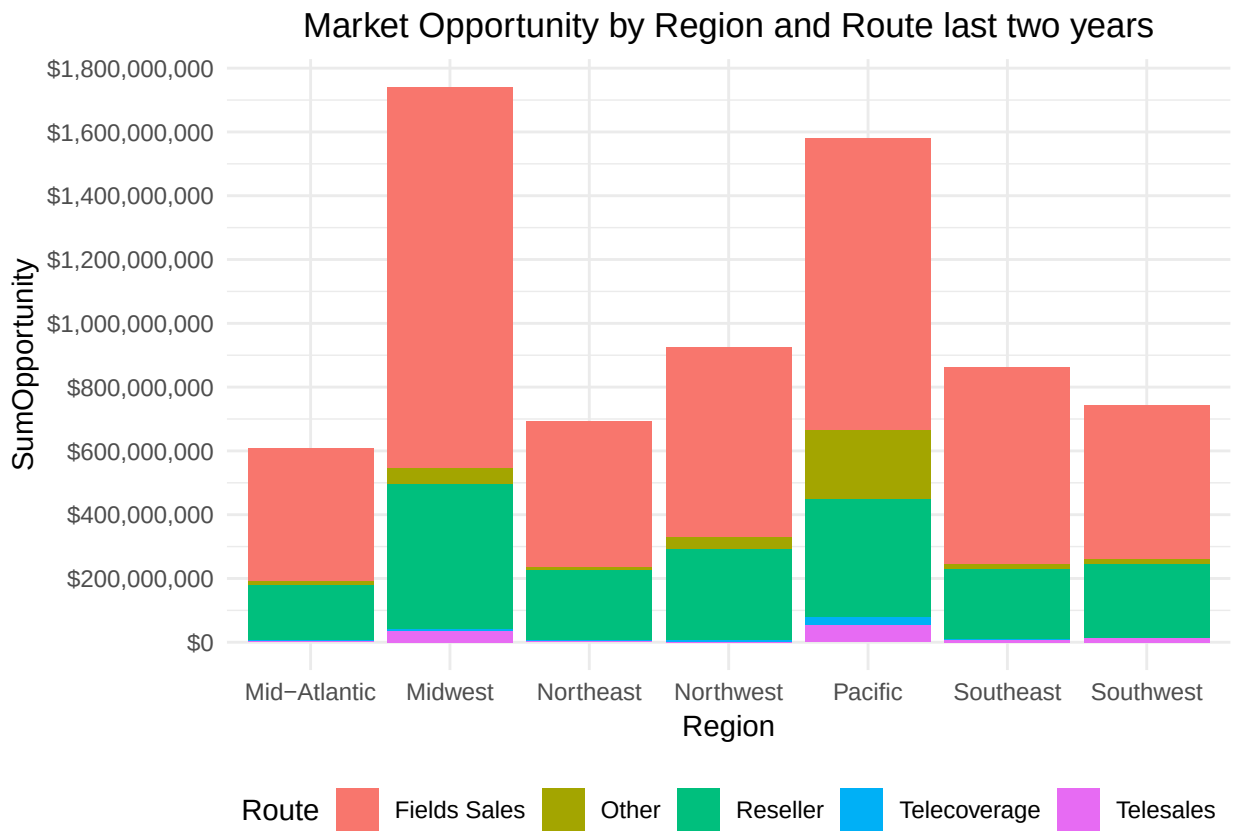
From the chart, we can see for the last two years, there are plenty of sales leads opportunity for the last two years. The one that resulting in $0 revenue is mainly the 'Loss' opportunity leads. The 'Won' opportunity that resulted in $0 revenue is possibly due to customer canceling the sales at the last minute.

```
position <- c("Rev>=$1.5M","$400K<=Rev<$1.5M","$50K<=Rev<$400K","$1<=Rev<$50K","Rev=$0")
ggplot(sales_win_loss) +
  geom_bar(aes(x = Revenue2, fill = Result), position = "fill") +
  scale_x_discrete(limits = position) +
  scale_y_continuous(labels = scales::percent_format()) +
  coord_flip() +
  ggtitle("Revenue vs Result last two years") +
  xlab("Revenue") +
  ylab("Percent(%)") +
  theme(plot.title = element_text(hjust = 0.5))
```



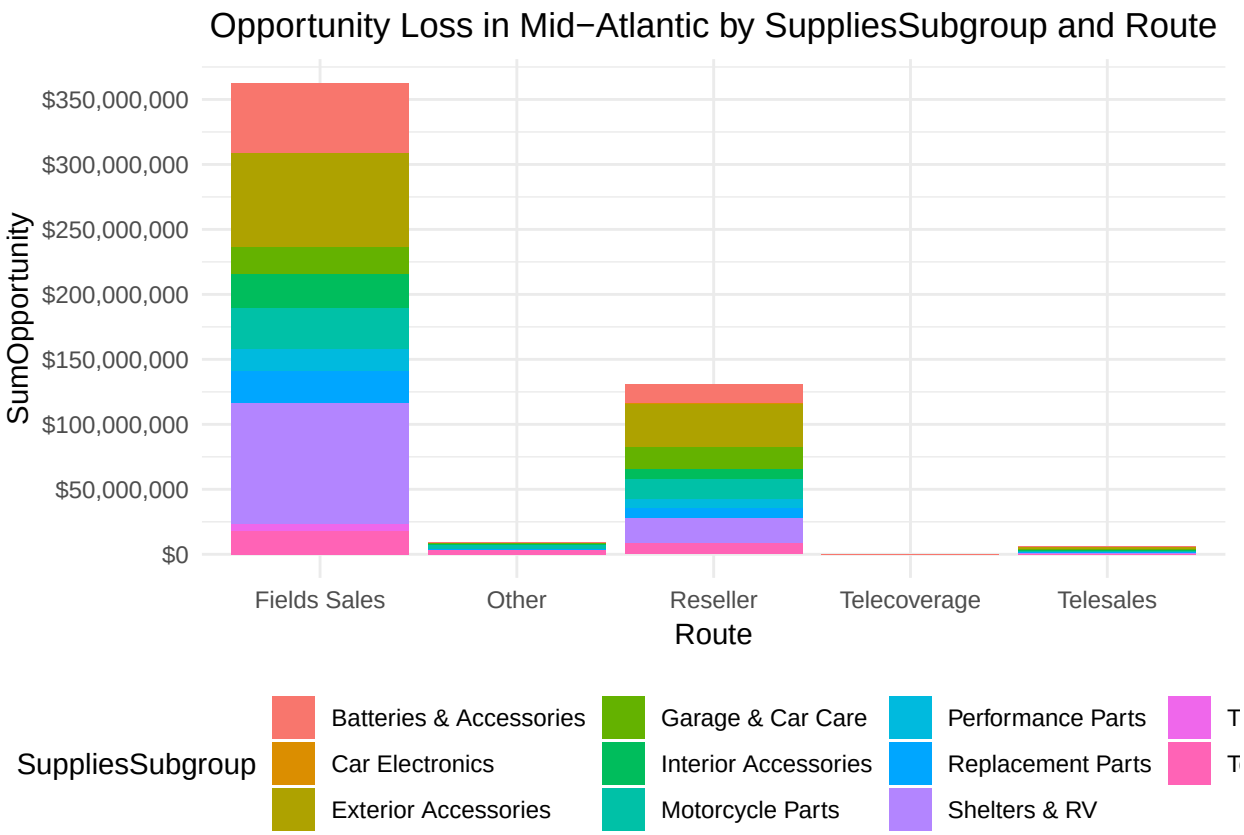
We can see that the probability of loss opportunity is higher if customer didn't buy anything in the last two years. If client purchase in the last two years, the chance of win decreases as sales deals rises

```
sales_win_loss %>%
  group_by(Region, Route) %>%
  summarise(SumOpportunity = sum(Opportunity)) %>%
  ggplot(aes(x = Region, y = SumOpportunity, fill = Route)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(breaks = seq(0,1e+11, 2e+08),
                     labels = scales::dollar_format(prefix = "$")) +
  ggtitle("Market Opportunity by Region and Route last two years") +
  theme(plot.title = element_text(hjust = 0.5))
```



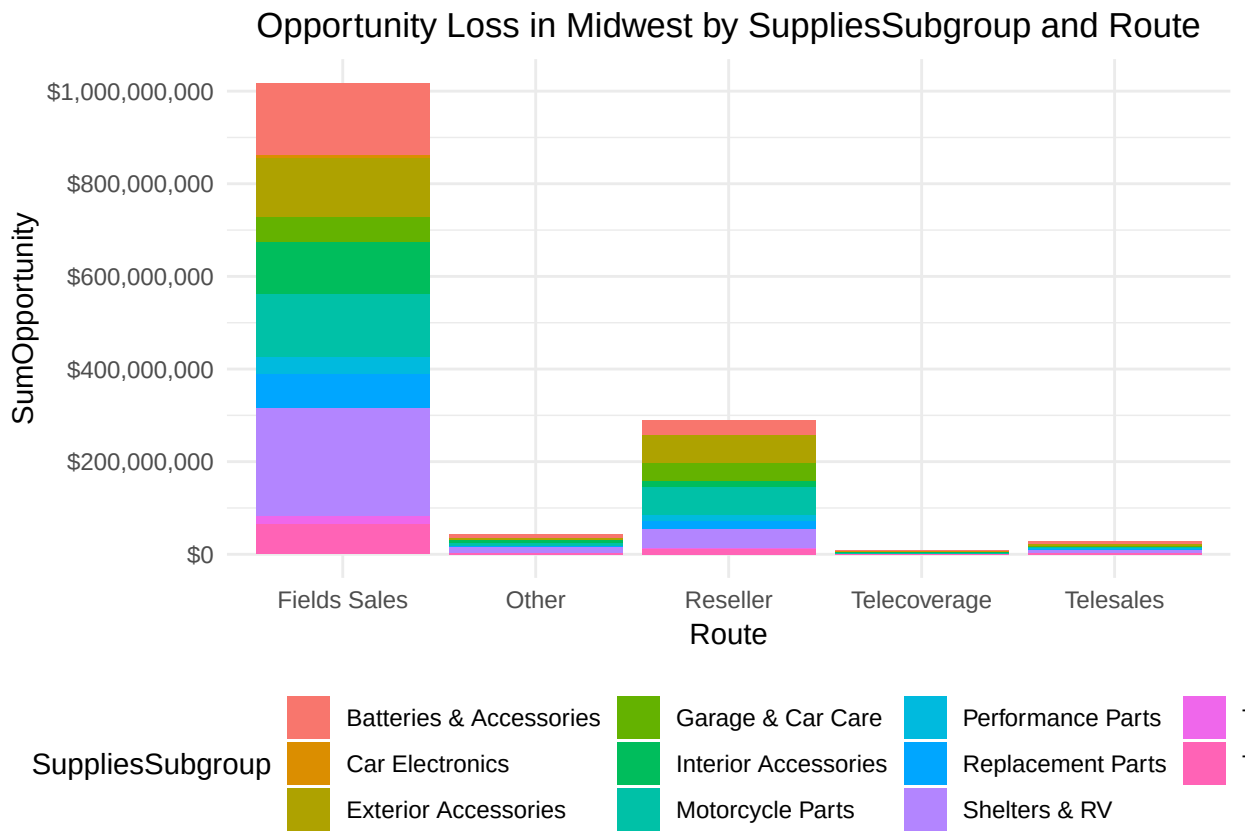
From the chart above, it's clear that the field sales and reseller are the most common channel of sales route to market across all regions. In the pacific region, other sales channel also play some role to bring in revenue in comparison to other regions. I want to see the breakdown of the 'Loss' opportunity by each region to dig deeper. The subsequent charts will show the break down of the supplies subgroup for each region where sales lead is 'Loss'

```
sales_win_loss %>% filter(Result == "Loss" & Region == "Mid-Atlantic") %>%
  group_by(Route, SuppliesSubgroup) %>%
  summarise(SumOpportunity = sum(Opportunity)) %>%
  ggplot(aes(x = Route, y = SumOpportunity, fill = SuppliesSubgroup)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(breaks = seq(0, 4e+08, 5e+07),
                     labels = scales::dollar_format(prefix = "$")) +
  ggtitle("Opportunity Loss in Mid-Atlantic by SuppliesSubgroup and Route") +
  theme(plot.title = element_text(hjust = 0.5))
```



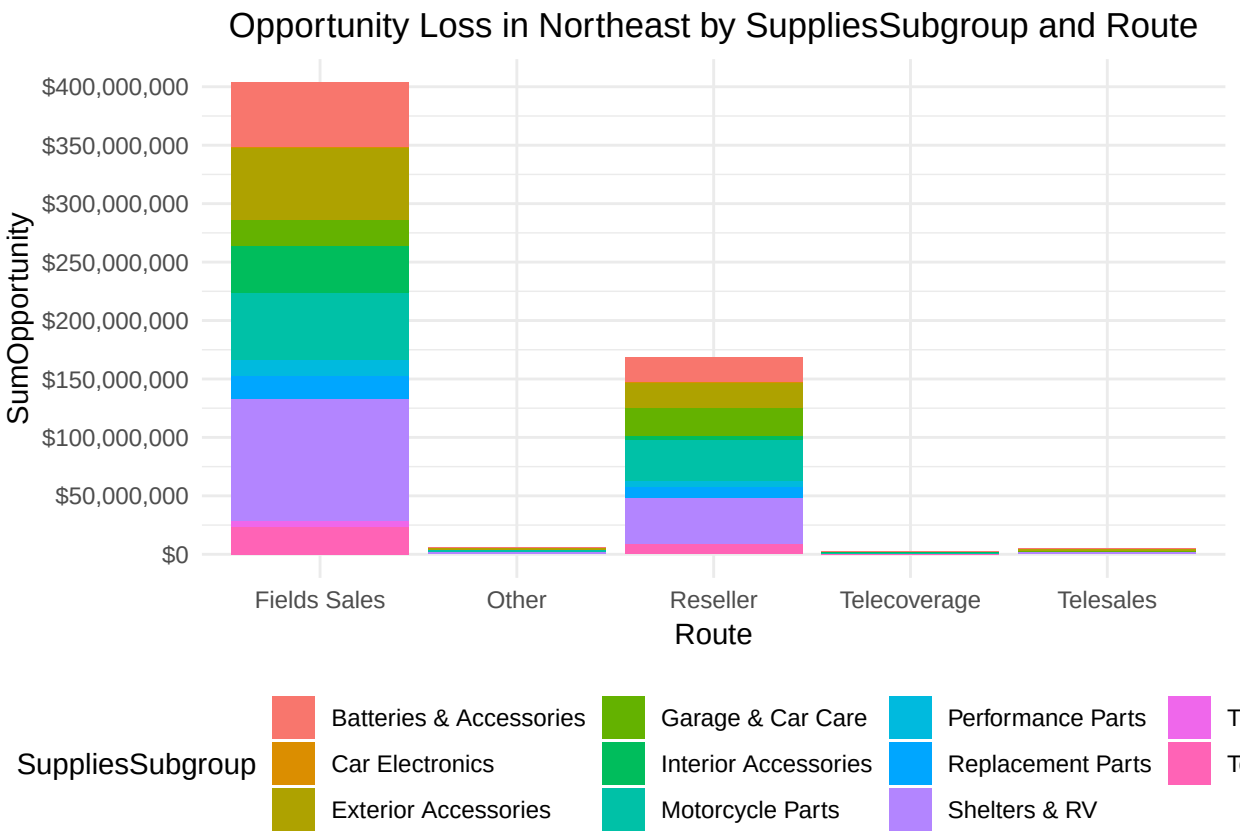
The chart above show the Mid-Atlantic region breakdown of loss opportunity by supplies subgroup. In this region, “Shelters & RV”, Batteries & Accessories“, and”Exterior Accessories" are the main supplies subgroup that contribute to the loss opportunity.

```
sales_win_loss %>% filter(Result == "Loss" & Region == "Midwest") %>%
  group_by(Route, SuppliesSubgroup) %>%
  summarise(SumOpportunity = sum(Opportunity)) %>%
  ggplot(aes(x = Route, y = SumOpportunity, fill = SuppliesSubgroup)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(breaks = seq(0, 1.5e+09, 2e+08),
                     labels = scales::dollar_format(prefix = "$")) +
  ggtitle("Opportunity Loss in Midwest by SuppliesSubgroup and Route") +
  theme(plot.title = element_text(hjust = 0.5))
```



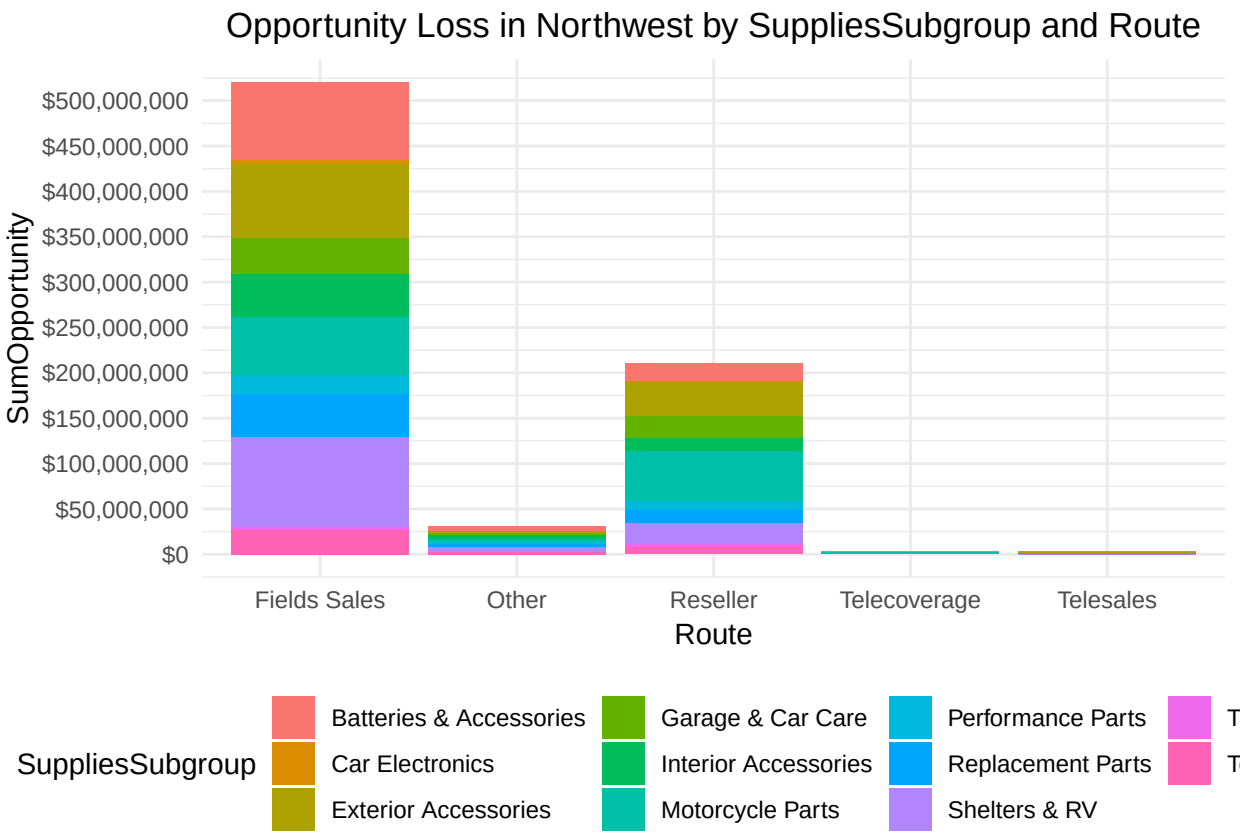
The chart above show the MidWest region breakdown of loss opportunity by supplies subgroup. In this region, “Shelters & RV”, Batteries & Accessories“, and”Exterior Accessories" are the main supplies subgroup that contribute to the loss opportunity.

```
sales_win_loss %>% filter(Result == "Loss" & Region == "Northeast") %>%
  group_by(Route, SuppliesSubgroup) %>%
  summarise(SumOpportunity = sum(Opportunity)) %>%
  ggplot(aes(x = Route, y = SumOpportunity, fill = SuppliesSubgroup)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(breaks = seq(0, 5e+08, 5e+07),
                     labels = scales::dollar_format(prefix = "$")) +
  ggtitle("Opportunity Loss in Northeast by SuppliesSubgroup and Route") +
  theme(plot.title = element_text(hjust = 0.5))
```



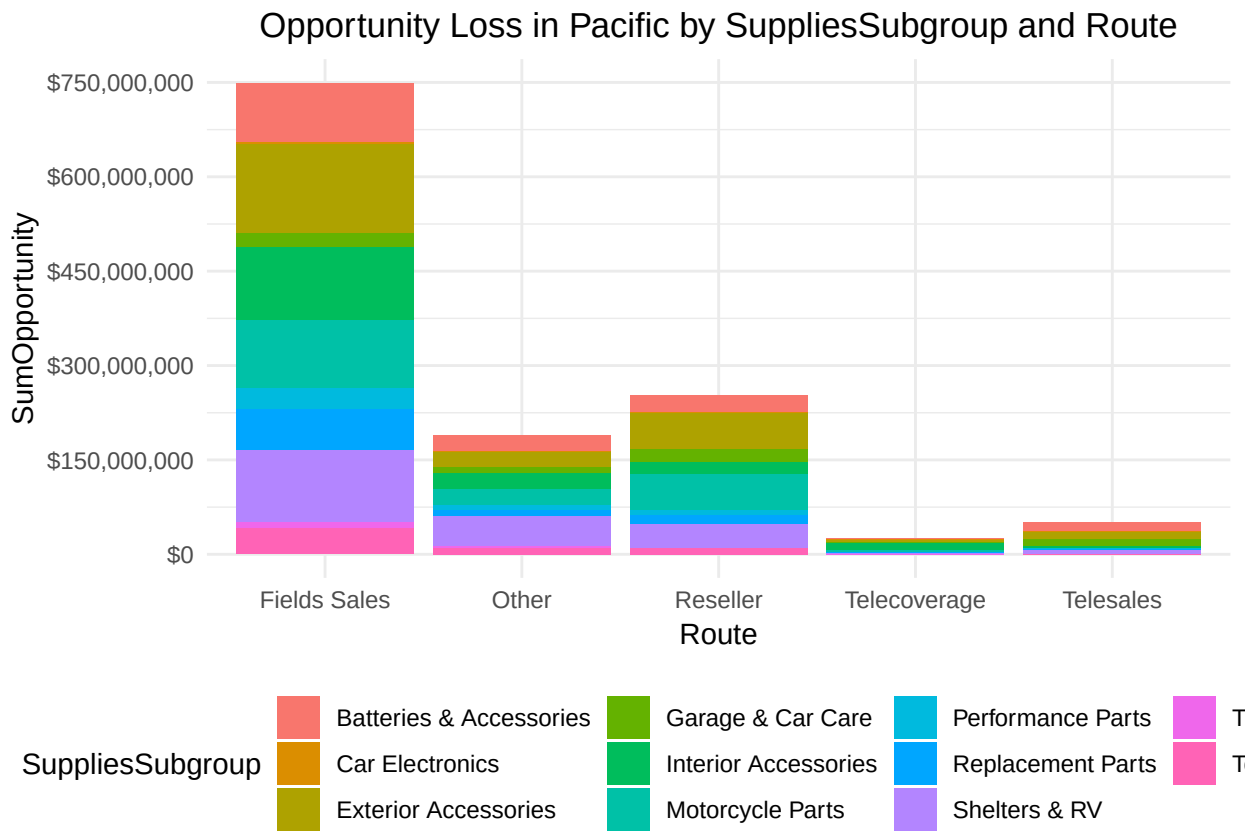
The chart above show the Northeast region breakdown of loss opportunity by supplies subgroup. In this region, "Shelters & RV" and "Motorcycle Parts"" are the main supplies subgroup that contribute to the loss opportunity.

```
sales_win_loss %>% filter(Result == "Loss" & Region == "Northwest") %>%
  group_by(Route, SuppliesSubgroup) %>%
  summarise(SumOpportunity = sum(Opportunity)) %>%
  ggplot(aes(x = Route, y = SumOpportunity, fill = SuppliesSubgroup)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(breaks = seq(0, 1e+11, 5e+07),
                     labels = scales::dollar_format(prefix = "$")) +
  ggtitle("Opportunity Loss in Northwest by SuppliesSubgroup and Route") +
  theme(plot.title = element_text(hjust = 0.5))
```



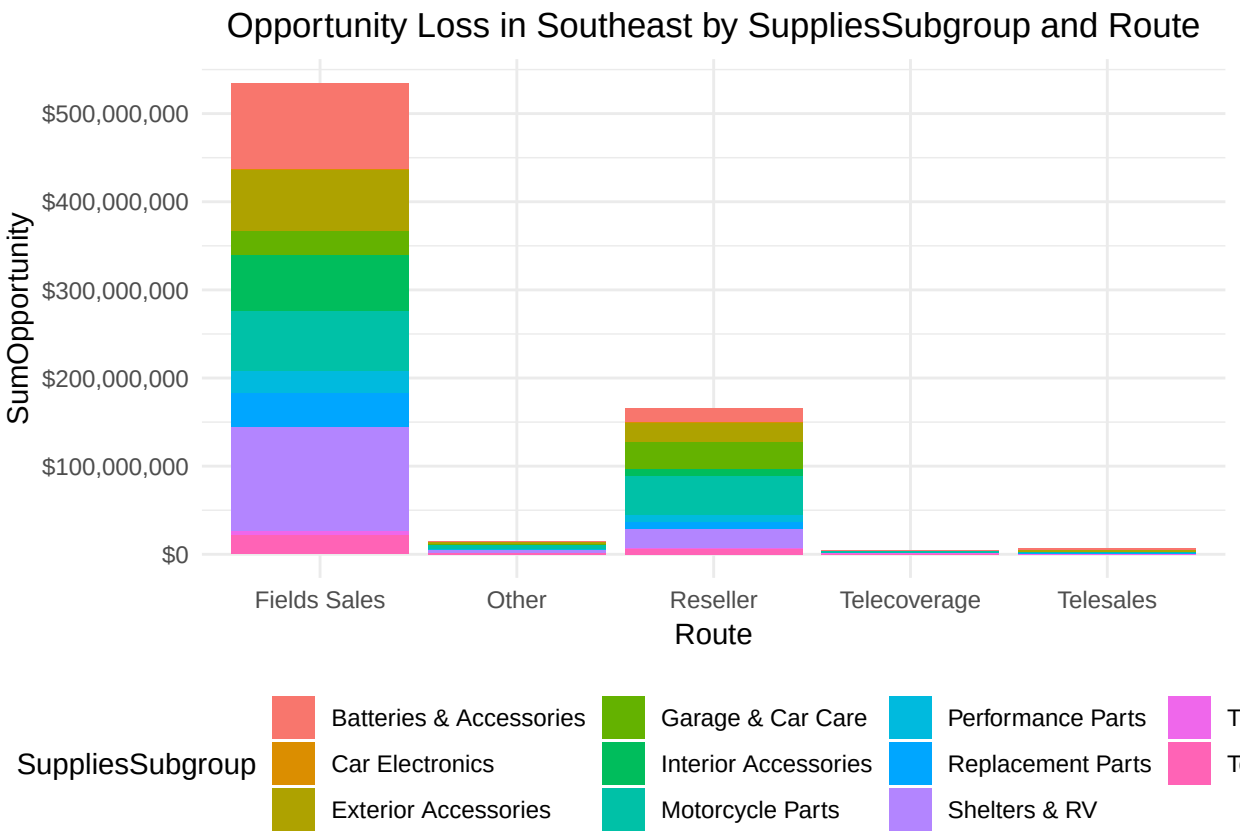
The chart above show the Northwest region breakdown of loss opportunity by supplies subgroup. In this region, “Shelters & RV”, Batteries & Accessories“, and”Exterior Accessories" are the main supplies subgroup that contribute to the loss opportunity.

```
sales_win_loss %>% filter(Result == "Loss" & Region == "Pacific") %>%
  group_by(Route, SuppliesSubgroup) %>%
  summarise(SumOpportunity = sum(Opportunity)) %>%
  ggplot(aes(x = Route, y = SumOpportunity, fill = SuppliesSubgroup)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(breaks = seq(0, 1e+11, 1.5e+08),
                     labels = scales::dollar_format(prefix = "$")) +
  ggtitle("Opportunity Loss in Pacific by SuppliesSubgroup and Route") +
  theme(plot.title = element_text(hjust = 0.5))
```



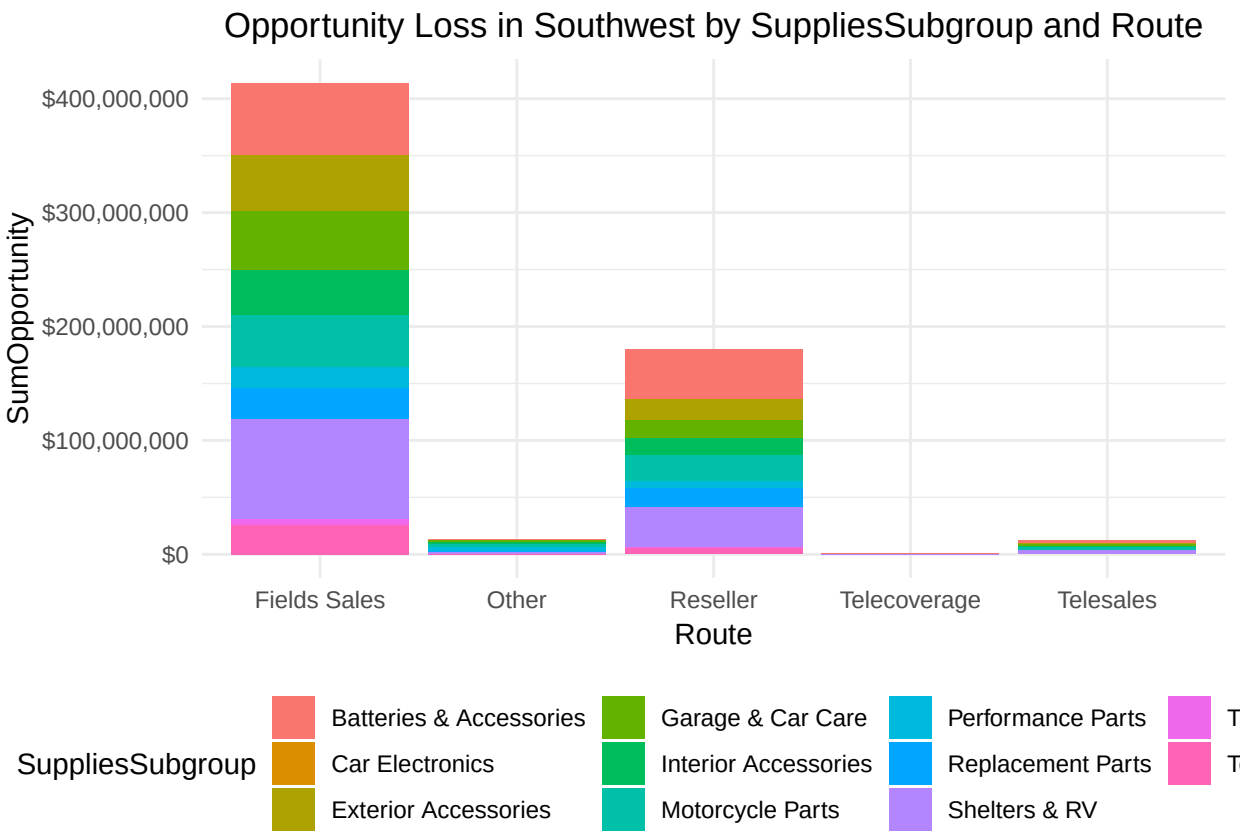
The chart above show the Pacific region breakdown of loss opportunity by supplies subgroup. In this region, “Shelters & RV”, Batteries & Accessories“, and”Exterior Accessories" are the main supplies subgroup that contribute to the loss opportunity.

```
sales_win_loss %>% filter(Result == "Loss" & Region == "Southeast") %>%
  group_by(Route, SuppliesSubgroup) %>%
  summarise(SumOpportunity = sum(Opportunity)) %>%
  ggplot(aes(x = Route, y = SumOpportunity, fill = SuppliesSubgroup)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(breaks = seq(0, 1e+11, 1e+08),
                     labels = scales::dollar_format(prefix = "$")) +
  ggtitle("Opportunity Loss in Southeast by SuppliesSubgroup and Route") +
  theme(plot.title = element_text(hjust = 0.5))
```



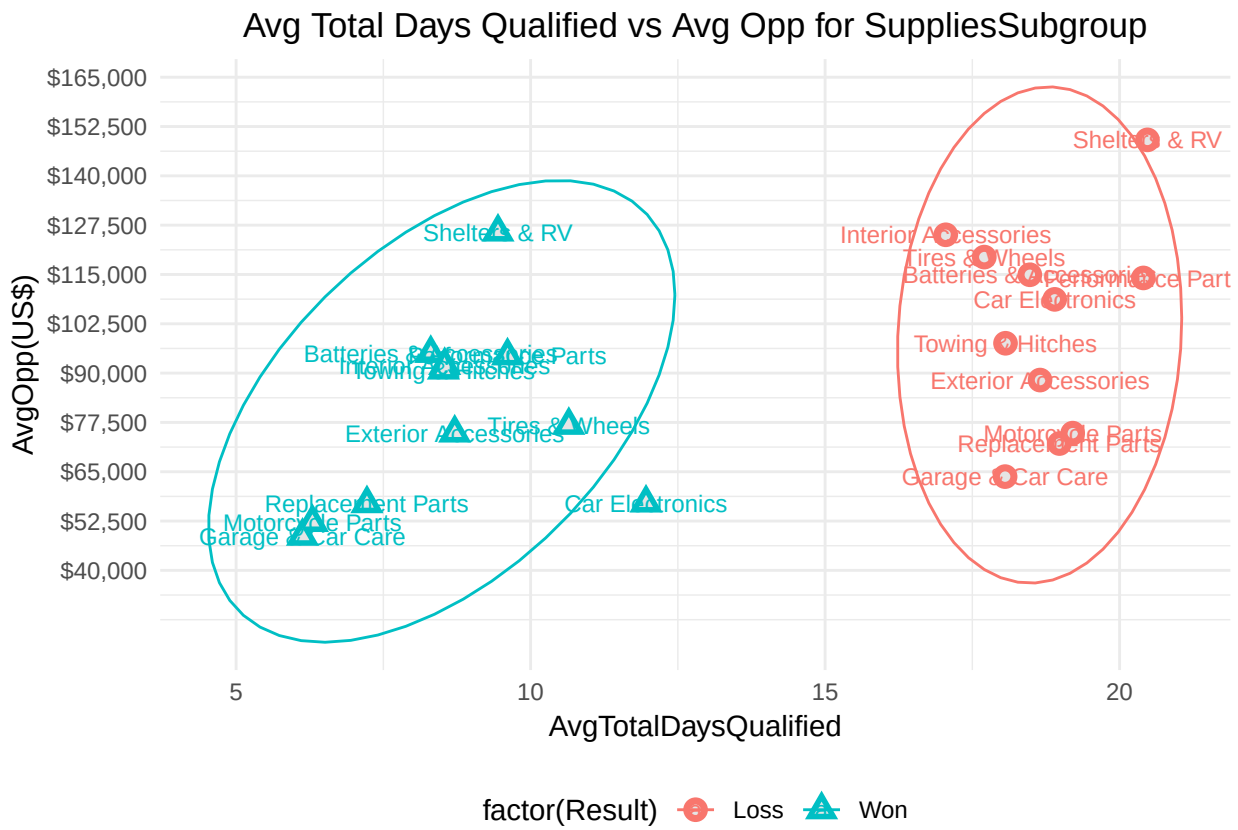
The chart above show the MidWest region breakdown of loss opportunity by supplies subgroup. In this region, “Shelters & RV”, Batteries & Accessories“, and”Exterior Accessories" are the main supplies subgroup that contribute to the loss opportunity.

```
sales_win_loss %>% filter(Result == "Loss" & Region == "Southwest") %>%
  group_by(Route, SuppliesSubgroup) %>%
  summarise(SumOpportunity = sum(Opportunity)) %>%
  ggplot(aes(x = Route, y = SumOpportunity, fill = SuppliesSubgroup)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(breaks = seq(0, 1e+11, 1e+08),
                     labels = scales::dollar_format(prefix = "$")) +
  ggtitle("Opportunity Loss in Southwest by SuppliesSubgroup and Route") +
  theme(plot.title = element_text(hjust = 0.5))
```



The chart above show the SouthWest region breakdown of loss opportunity by supplies subgroup. In this region, “Shelters & RV”, Batteries & Accessories“, and”Exterior Accessories" are the main supplies subgroup that contribute to the loss opportunity.

```
sales_win_loss %>%
  group_by(Result, SuppliesSubgroup) %>%
  summarise(AvgOpp = mean(Opportunity),
            AvgQual = mean(TotalDaysQualified )) %>%
  ggplot(aes(x = AvgQual, y = AvgOpp, shape = factor(Result), label = SuppliesSubgroup)) +
  geom_point(aes(colour = factor(Result)), size = 4) +
  geom_point(colour = "grey90", size = 1.5) +
  xlab("AvgTotalDaysQualified") + ylab("AvgOpp(US$)") +
  scale_x_continuous(breaks = seq(0, 30, 5)) +
  scale_y_continuous(breaks = seq(40000, 175000, 12500),
                     labels = scales::dollar_format(prefix = "$")) +
  geom_text(aes(label = SuppliesSubgroup,  color = Result), size = 3) +
  stat_ellipse(aes(color = Result), type = "t") +
  ggtitle("Avg Total Days Qualified vs Avg Opp for SuppliesSubgroup") +
  theme(plot.title = element_text(hjust = 0.5))
```



Looking from left to right, this Scatter chart shows that irrespective of opportunity amounts, we start losing deals as they stay longer in the pipeline. This could help formulate threshold levels for each supplier based on how many days a deal is in the pipeline and create alert mechanisms to expedite its progression.

Machine Learning Modeling

Create Modeling Data

The next step is partitioning the data set into training, validation, and testing dataset. We start with setting seed for reproducibility followed by using createDataPartition function from caret. For the 'ModelData', we have selected several columns from the original 19 available columns. The selection of which columns that got selected is based on the EDA phase where we've seen impact of these independent variables on the dependent variables.

```
ModelData <- sales_win_loss %>% select(SuppliesSubgroup, Region, Route, TotalDaysClosing,
                                       TotalDaysQualified, Opportunity, ClientSizeRev,
                                       ClientSizeCount, Competitor, DealSize, Result)

set.seed(3456)
TrainingValidationIndex <- caret::createDataPartition(ModelData$Result, p = .80,
                                                      list = FALSE,
                                                      times = 1)
str(TrainingValidationIndex)
```

```
##  int [1:62421, 1] 3 6 8 9 10 11 13 14 15 16 ...
```

```
##  - attr(*, "dimnames")=List of 2
##   ..$ : NULL
##   ..$ : chr "Resample1"
```

```
TrainingValidation <- ModelData[ TrainingValidationIndex,]
TrainingIndex <- caret::createDataPartition(TrainingValidation$Result, p = .75,
                                            list = FALSE,
                                            times = 1)

Training <- TrainingValidation[TrainingIndex,]
Validation <- TrainingValidation[-TrainingIndex,]
Testing  <- ModelData[-TrainingValidationIndex,]
```

Using glimpse function, we can see the selected variables for the model.

```
glimpse(Training)
```

```
## Observations: 46,817
## Variables: 11
## $ SuppliesSubgroup   <chr> "Shelters & RV", "Batteries & Accessories",...
## $ Region             <chr> "Pacific", "Northwest", "Pacific", "Northwe...
## $ Route              <chr> "Reseller", "Fields Sales", "Reseller", "Fi...
## $ TotalDaysClosing   <int> 114, 156, 50, 165, 31, 208, 138, 32, 130, 1...
## $ TotalDaysQualified <int> 0, 156, 50, 165, 31, 208, 138, 32, 130, 125...
## $ Opportunity        <int> 232522, 250000, 55003, 0, 10000, 232522, 20...
## $ ClientSizeRev      <int> 5, 1, 1, 1, 2, 1, 4, 5, 4, 1, 3, 1, 1, 5, 1...
## $ ClientSizeCount    <int> 1, 5, 1, 2, 1, 1, 5, 1, 3, 5, 5, 1, 4, 3, 5...
## $ Competitor         <chr> "Unknown", "None", "Unknown", "Unknown", "U...
## $ DealSize           <int> 5, 6, 4, 1, 2, 5, 5, 1, 4, 5, 4, 3, 4, 4, 7...
## $ Result             <chr> "Loss", "Loss", "Loss", "Loss", "Loss", "Lo...
```

Training the Logistic and Random Forest model using training dataset

Next, we setup the model parameter

```
control <- caret::trainControl(method = "cv", number = 2, classProbs = TRUE)
seed <- 7
metric <- "Accuracy"
set.seed(seed)
```

Training Logistic with training dataset

```
GLMModel <- caret::train(
  Result ~ SuppliesSubgroup + Region + Route + TotalDaysClosing + TotalDaysQualified +
    Opportunity + ClientSizeRev + ClientSizeCount + Competitor + DealSize,
  data = Training,
  method = "glm",
  trControl = control
)
```

Training Random Forest model with training dataset

```
RFModel <- caret::train(
  Result ~ SuppliesSubgroup + Region + Route + TotalDaysClosing + TotalDaysQualified +
    Opportunity + ClientSizeRev + ClientSizeCount + Competitor + DealSize,
  data = Training,
  method = "rf",
  trControl = control
)
```

Predicting the model using validation data set

Using predict function from caret package, we can use the result to find the best model. Caret package also have the confusion matrix function to calculate sensitivity, specificity, negative predictied value, positive predicted values, and F1 score. The F1 score is a harmonic average of precision and recall to select the best model. We use F1 score not accuracy because accuracy can be largely contributed by a large number of True Negatives which in most business circumstances, we do not focus on much whereas False Negative and False Positive usually has business costs (tangible & intangible) thus F1 Score might be a better measure to use if we need to seek a balance between Precision and Recall and there is an uneven class distribution (large number of Actual Negatives).

```
PredGLM <- predict(GLMModel, Validation)

ConfMatGLM <- caret::confusionMatrix(
  PredGLM, factor(Validation$Result), positive = "Won",
  mode = "everything")

ConfMatGLM
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  Loss   Won
##       Loss 11604  2703
##       Won    475   822
##
##                Accuracy : 0.7963
##                  95% CI : (0.7899, 0.8026)
##     No Information Rate : 0.7741
##     P-Value [Acc > NIR] : 9.572e-12
##
##                   Kappa : 0.2498
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.23319
##             Specificity : 0.96068
##          Pos Pred Value : 0.63377
##          Neg Pred Value : 0.81107
##               Precision : 0.63377
##                  Recall : 0.23319
##                      F1 : 0.34094
##              Prevalence : 0.22590
##          Detection Rate : 0.05268
##    Detection Prevalence : 0.08312
```

```
##       Balanced Accuracy : 0.59693
##
##        'Positive' Class : Won
##
```

The F1 score for the Logistic model is 0.3409. Next, we did the same thing using Random Forest model with validation data set

```
PredRF <- predict(RFModel, Validation)

ConfMatRF <- caret:: confusionMatrix(PredRF, factor(Validation$Result),
                                     positive = "Won",
                                     mode = "everything")

ConfMatRF
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  Loss   Won
##       Loss 11233  1893
##       Won    846  1632
##
##                Accuracy : 0.8245
##                  95% CI : (0.8184, 0.8304)
##     No Information Rate : 0.7741
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.4391
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.4630
##             Specificity : 0.9300
##          Pos Pred Value : 0.6586
##          Neg Pred Value : 0.8558
##               Precision : 0.6586
##                  Recall : 0.4630
##                      F1 : 0.5437
##              Prevalence : 0.2259
##          Detection Rate : 0.1046
##    Detection Prevalence : 0.1588
##       Balanced Accuracy : 0.6965
##
##        'Positive' Class : Won
##
```

Selecting the best model

The F1 score for the Random Forest model is 0.5437. Because the F1 score for Random Forest model is higher than the logistic model. The random forest model is selected as the better model. The next step is to evaluate the Random Forest model using testing dataset.

```
FinalPredRF <- predict(RFModel, Testing)

ConfMatFinalRF <- caret::confusionMatrix(FinalPredRF, factor(Testing$Result),
                                         positive = "Won", mode = "everything")

ConfMatFinalRF
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  Loss   Won
##       Loss 11256  1855
##       Won    823  1670
##
##                Accuracy : 0.8284
##                  95% CI : (0.8224, 0.8343)
##     No Information Rate : 0.7741
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.4525
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.4738
##             Specificity : 0.9319
##          Pos Pred Value : 0.6699
##          Neg Pred Value : 0.8585
##               Precision : 0.6699
##                  Recall : 0.4738
##                      F1 : 0.5550
##              Prevalence : 0.2259
##          Detection Rate : 0.1070
##    Detection Prevalence : 0.1598
##       Balanced Accuracy : 0.7028
##
##        'Positive' Class : Won
##
```

Testing the final model

The final prediction using testing data set(0.5550) appears to have about the same F1 score as using the validation data set(0.5437).

As someone who is going to need a new prediction, the variable such as TotalDaysClosing or TotalDaysQualified are moving variable as days have passed until the present time. In order to use the prediction model, based on the two years data, the user need to be aware of these type of variable. The data below provide some additional insight for these two variables.

```
sales_win_loss %>% filter(Opportunity > 0) %>%
  group_by(Result) %>%
  summarise(AvgTotalDaysClose = mean(TotalDaysClosing),
            AvgTotalDaysQualify = mean(TotalDaysQualified),
            MaxTotalDaysClose = max(TotalDaysClosing),
            MaxTotalDaysQualify = max(TotalDaysQualified),
```

```
            MedianDaysClose = median(TotalDaysClosing),
            MedianDaysQualify = median(TotalDaysQualified))
```

```
## # A tibble: 2 x 7
##   Result AvgTotalDaysClo~ AvgTotalDaysQua~ MaxTotalDaysClo~
##   <chr>             <dbl>            <dbl>            <dbl>
## 1 Loss              19.2             18.9               208
## 2 Won                8.53             7.63              192
## # ... with 3 more variables: MaxTotalDaysQualify <dbl>,
## #   MedianDaysClose <dbl>, MedianDaysQualify <dbl>
```

Recommendations

1. From the exploratory data analysis, it looks like the company start losing deals as the sales lead stays longer in the pipeline more than 12 days. Using this information, the sales team should be able to formulate threshold levels for each supplier based on how many days a deal is in the pipeline and create alert mechanisms to expedite its progression.

2. More depth machine learning model should also be done. Especially learning the interaction of other independent variables such as RDaysValidated and RDaysQualified.

3. The current Random Forest model F1 score is 0.5550. Other classification model such as clustering k-means or other classification model that can produce higher accuracy should be explored.

Acknowledgements

Many thanks to Mike Badescu, my mentor throughout my learning process of Introduction to Data Science class. Appreciate his unending patience, tips, and code snippets.