# Customer Modeling and Prospecting

*Nathaniel Reed*

*October 21, 2015*

### Introduction

In this case study, we simulate customer accounts in Salesforce for a set of existing customers and build a linear regression model that could be used for prospecting.

Our fictitious company is a software company that sells a POS (point-of-sale) tax solution for the apparel industry. We're interested in understanding which of our existing customers are "best". Further, we would like to use the model of our best customers to predict sales to new customers.

We will use multi-variable linear regression to come up with this predictive model.

## Variables?

First, what constitutes a "good customer"? Although there are many ways to measure that, we are primarily concerned with Deal Size and Days to Close.

Our working hypothesis is that deal size and days to close are influenced strongly by the following independent variables:

- Sales Volume (revenue)
- Employee Count

## Obtain data

The data was obtained by quering a database of firmographics related data, which was then written to a CSV file (companies.csv):

```
require(RMySQL)
```

```
## Warning: package 'RMySQL' was built under R version 3.1.3
```

```
mydb = dbConnect(MySQL(), user='miheir', password='description', dbname='master', host='cluster1.cluste
rs = dbSendQuery(mydb, "SELECT `Duns Number` as duns_number,
                               `Business Name` as business_name,
                               `Physical City` as physical_city,
                               `Physical State Abbreviation` as physical_state_abbr,
                               `Emp Total` as emp_total,
                               `Sales Volume US` as sales_volume,
                               `Primary SIC 8 Digit` as primary_sic_8,
                               `Primary SIC 8 Digit Description` as primary_sic_desc,
                               `Viability Score` as viability_score,
                               `NAICS 6 Digit Description` as naics_6_desc,
                               `NAICS 6 Digit Code` as naics_6,
                               `Market Segmentation Cluster` as market_segmentation_cluster
```

```
FROM master.textgen_data
WHERE `Primary SIC 8 Digit` like '56%'
  AND `Sales Volume US` <= 500000000;");

data = fetch(rs, n=-1)
write.csv(data, file="/Users/reedn/DataChallenge/companies.csv")
```

Here we load the data, converting to the appropriate data types for analysis:

```
data = read.table(file="/Users/reedn/DataChallenge/companies.csv",
                  col.names = c("row_id", "duns", "business_name", "physical_city",
                                "physical_state_abbr", "emp_total", "sales_volume", "primary_sic",
                                "primary_sic_desc", "viability_score", "naics_desc", "naics",
                                "market_segmentation_cluster"),
                  colClasses=c("character",        # row id
                               "character",        # Duns
                               "character",        # Business Name
                               "character",        # Physical City
                               "factor",           # State
                               "numeric",          # Emp Total
                               "numeric",          # Sales Volume
                               "factor",           # Primary Sic
                               "character",        # Primary Sic Description
                               "factor",           # Viability Score
                               "character",        # NAICS DESC
                               "factor",           # NAICS
                               "factor"           # marketing segmentation cluster
                               ),
                  skip=2, quote="\"'", sep=",")
```

We will simulate accounts by generating Deal Size and Days to Close based on variables such as sales_volume:

```
library(dplyr)
n <- 1 : nrow(data);
mn_sales <- mean(data$sales_volume)
epsilon <- mn_sales / nrow(data) * 1000
accounts <- data %>%
mutate(expected_deal_size = sales_volume * .20,
        noise = runif(length(sales_volume),
                      -epsilon,
                      epsilon),
        deal_size = ifelse(expected_deal_size + noise < 0,
                           0,
                           expected_deal_size + noise))

# Days to close
accounts <- mutate(accounts, days_to_close = (sales_volume >= 18225000) * 90 + (sales_volume <= 1822500
```

For the analysis, we will select only the variables we're interested in:

```
accounts <- select(accounts, days_to_close, deal_size, emp_total, sales_volume, physical_state_abbr)
```

Here's what our data looks like:

```r
summary(accounts)
```

```
##  days_to_close       deal_size            emp_total      
##  Min.   :  5.008   Min.   :        0   Min.   :    0.0  
##  1st Qu.: 24.980   1st Qu.:        0   1st Qu.:    2.0  
##  Median : 42.641   Median :  2629699   Median :    6.0  
##  Mean   : 47.412   Mean   :  8836437   Mean   :  312.7  
##  3rd Qu.: 68.235   3rd Qu.: 12480382   3rd Qu.:   75.0  
##  Max.   :104.996   Max.   :113959376   Max.   :27220.0  
##                                                         
##   sales_volume       physical_state_abbr
##  Min.   :        0          :290        
##  1st Qu.:   100000   CA     :196        
##  Median :   500000   NY     :117        
##  Mean   : 27164007   FL     :110        
##  3rd Qu.:  7025000   NJ     : 69        
##  Max.   :497700000   PA     : 63        
##                      (Other):743
```

## Independent Variables

We're interested in understanding relationships among our dependent and independent variables. Scatterplot matrixes can help show these relationships.

Because there are over 100 states and provinces in this data set, it is helpful to create a smaller number of bins to visualize the correlations. Here we see the top 10 states / provinces in sales volume:

```r
by_state_abbr <- group_by(accounts, physical_state_abbr)
by_state_summary <- by_state_abbr %>% summarize(total_sales_volume = sum(sales_volume)) %>% arrange(des
by_state_summary[1:10,]$physical_state_abbr
```

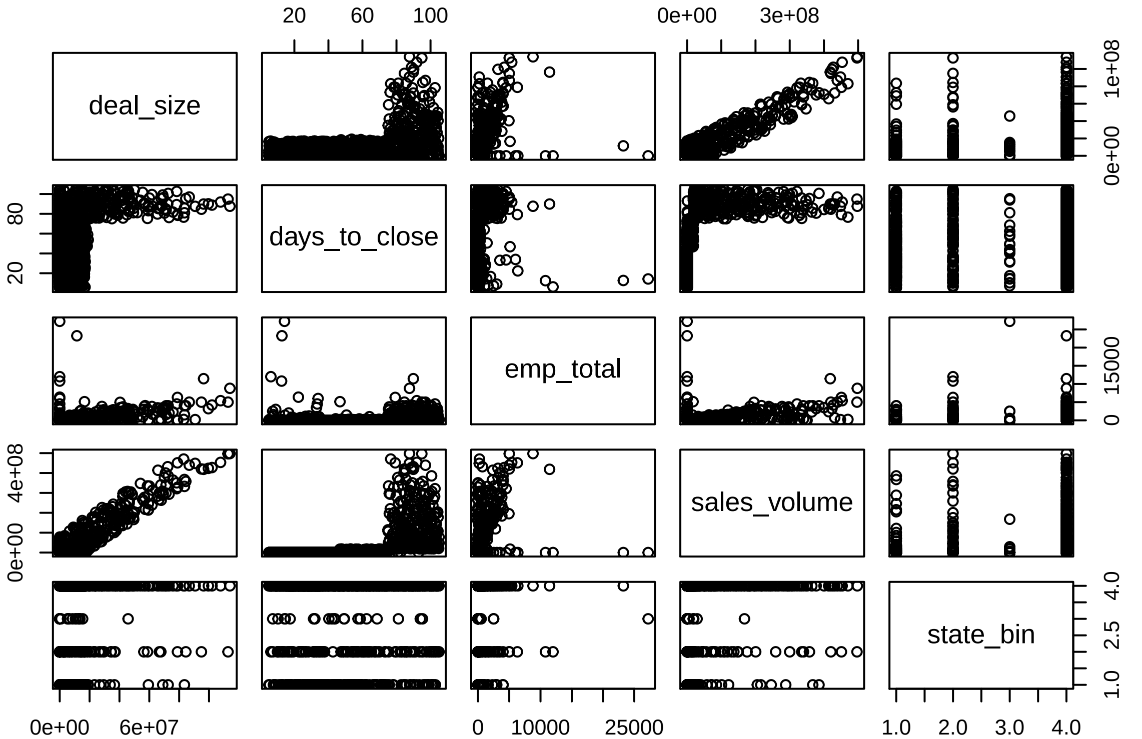
```
##  [1]     NY  CA  TKY KLN ON  NJ  FL  MA  QC 
## 127 Levels:  AB AG AIC AK AL AR AZ BA BC BE BEDS BL BS BUCKS CA CB ... ZH
```

Our top 3 states in terms of sales volume are California, New York and Ohio (no surprises, as those are very populous states).

```r
accounts <- accounts %>%
  mutate(state_bin = as.factor(ifelse(physical_state_abbr == 'CA', 'CA', ifelse(physical_state_abbr ==
```

We can look at the scatterplots of the various variables:

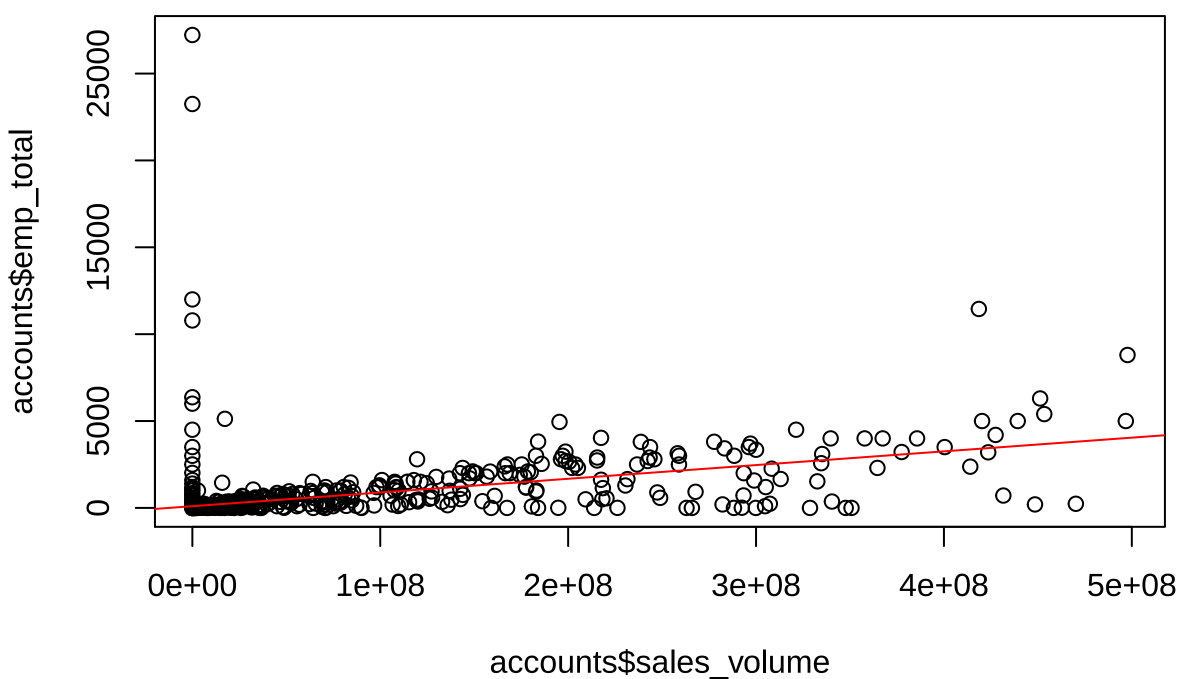
```r
pairs(~ deal_size + days_to_close + emp_total + sales_volume + state_bin, data=accounts)
```




## Collinearity

We see that there is a strong correlation between sales volume and employee total:

```r
plot(accounts$sales_volume, accounts$emp_total)
abline(lm(accounts$emp_total~accounts$sales_volume), col="red")
```



We can throw out one of these variables when we perform a regression analysis.

## Perform Linear Regression

Let's use sales_volume as a predictor of deal size:

```
fit <- lm(data=accounts, deal_size ~ sales_volume)
summary(fit)
```

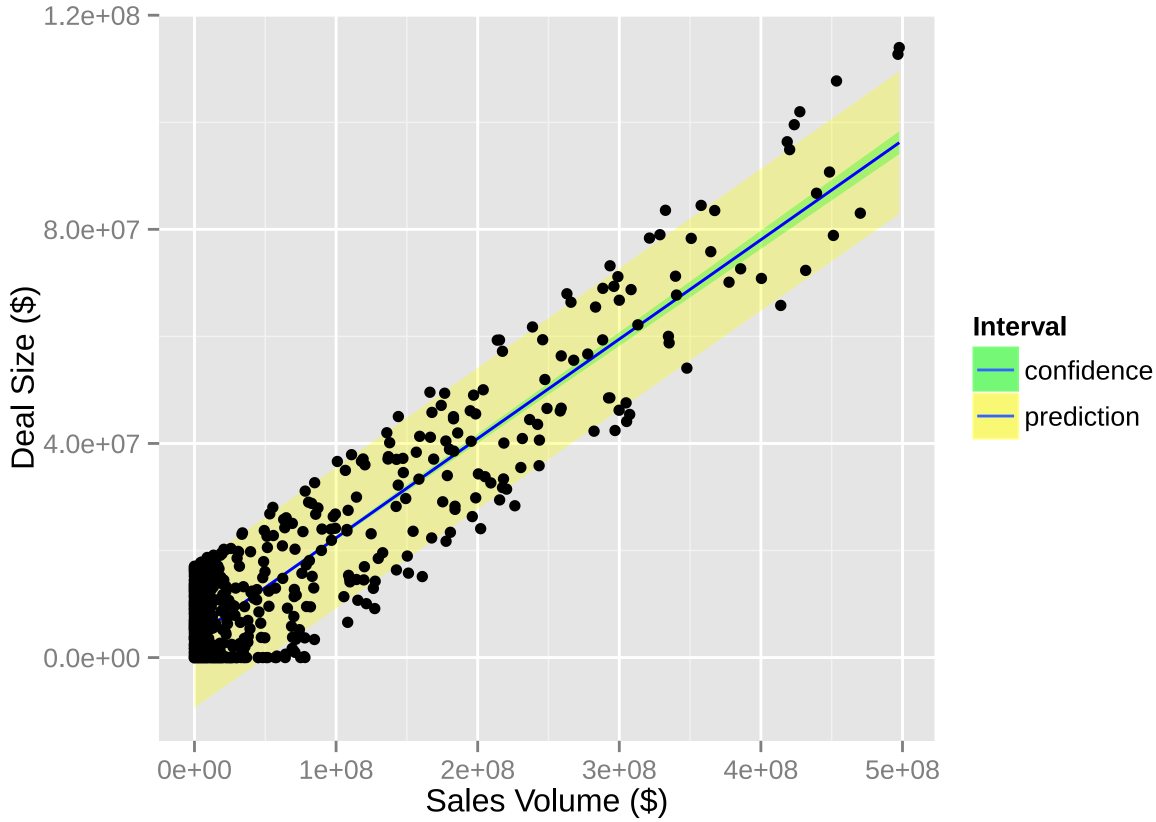
```
##
## Call:
## lm(formula = deal_size ~ sales_volume, data = accounts)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -18530726  -3872700  -3614290   5328624  19752922
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.793e+06  1.797e+05   21.11   <2e-16 ***
## sales_volume 1.857e-01  2.299e-03   80.76   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6713000 on 1586 degrees of freedom
## Multiple R-squared:  0.8044, Adjusted R-squared:  0.8043
## F-statistic:  6522 on 1 and 1586 DF,  p-value: < 2.2e-16
```

The model shows a slope of approximately 0.68 with a p-value $< 0.05$, suggesting a significant linear relationship between deal size and sales_volume.

## Plot prediction interval with linear regression line

```
## Warning: package 'ggplot2' was built under R version 3.1.3
```

```
## Warning in predict.lm(fit, interval = "prediction"): predictions on current data refer to _future_ r
```


1.2e+08
8.0e+07
4.0e+07
0.0e+00
Deal Size ($)
0e+00
1e+08
2e+08
3e+08
4e+08
5e+08
Sales Volume ($)
Interval
confidence
prediction

## Residuals and Model Diagnostics