Linear Regression: Direct Marketing Analysis

Jonathan Velez

January 17, 2017

Introduction

The "DirectMarketing" dataset includes data from a direct marketer who sells his products only via direct mail. He sends catalogs with product characteristics to customers who then order directly from the catalogs. The marketer has developed customer records to learn what makes some customers spend more than others.

The data set includes n = 1000 customers and the following variables:

- Age: customer age – old, middle, or young
- Gender: male or female
- OwnHome: whether customer owns their home – own or rent
- Married: single or married
- Location: in terms of distance to nearest store selling similar products – far or close
- Salary: yearly salary of customer in dollars
- Children: number of children – 0-3
- History: history of previous purchase volume – low, medium, high, or NA; NA means customer has not yet completed a purchase
- Catalogs: number of catalogs sent – 6, 12, 18, or 24
- AmountSpent: the amount spent by the customer in dollars

The objective is to explain the amount spent by each customer in terms of the provided customer characteristics. Hence, for the resulting model, AmountSpent is the response variable, and Age, Gender, OwnHome, Married, Location, Salary, Children, History, and Catalogs are predictors.

Data Exploration

Preparation

```
# Load required packages
library("knitr")        ## summary table
library("ggplot2")      ## data visualization
library("e1071")        ## skewness
library("car")          ## scatter plot matrix
library("leaps")        ## regression subset selection
library("lars")         ## least absolute shrinkage and selection operator

# Load data
data.file = "http://www.yurulin.com/class/spring2017_datamining/data/DirectMarketing.csv"
df = read.csv(data.file, header = TRUE, sep = ',')

# Identify any missing values and handle missing data appropriately
summary(df)
```

```
##      Age         Gender    OwnHome       Married     Location
##  Middle:508   Female:506   Own :516   Married:502   Close:710
##  Old   :205   Male  :494   Rent:484   Single :498   Far  :290
##  Young :287
##
##
##
##      Salary          Children       History       Catalogs
##  Min.   : 10100   Min.   :0.000   High  :255   Min.   : 6.00
##  1st Qu.: 29975   1st Qu.:0.000   Low   :230   1st Qu.: 6.00
##  Median : 53700   Median :1.000   Medium:212   Median :12.00
##  Mean   : 56104   Mean   :0.934   NA's  :303   Mean   :14.68
##  3rd Qu.: 77025   3rd Qu.:2.000                3rd Qu.:18.00
##  Max.   :168800   Max.   :3.000                Max.   :24.00
##   AmountSpent
##  Min.   :  38.0
##  1st Qu.: 488.2
##  Median : 962.0
##  Mean   :1216.8
##  3rd Qu.:1688.5
##  Max.   :6217.0
```

```
## History contains missing values, but it is known that this means the customer
## has yet to make a purchase. Since these are not actually missing observations,
## add a new level named "NewCustomer" into History.
levels(df$History) = c(levels(df$History), "NewCustomer")
df$History[is.na(df$History)] = "NewCustomer"
summary(df$History)
```

```
##        High         Low      Medium NewCustomer
##         255         230         212         303
```

Quantitative Data

A table describing the central tendency and spread of each quantitative variable is included below.

The density distributions of features measured in monetary amounts are known to be sources of skewed distributions. Plotting the density of AmountSpent reveals a unimodal distribution with positive non-zero skewness. The skewness is greater than +1, indicating that the distribution is highly skewed in the positive direction. Plotting the density of Salary reveals a bimodal distribution with positive non-zero skewness. In this case, the skewness is between 0 and +0.5, indicating that the distribution is moderately skewed in the positive direction. The Shapiro-Wilk normality test confirms that these distributions are non-normal at a significance level of 0.01. The normal probability plot can additionally be used to explore the normality of these distributions, and these plots indicate the positive skewness for both features. It may be appropriate to apply a log transformation to these features if regression analysis results in a non-normal distribution of residuals.

Exploring the correlations (three bottom-left values of the correlation matrix) and scatter plots (three bottom-left figures of the scatter plot matrix) between numeric predictors and the response variable reveals some interesting trends. These results indicate that AmountSpent and Salary have a strong positive correlation, AmountSpent and Catalogs have a moderate positive correlation, and AmountSpent and Children have a slight negative correlation.

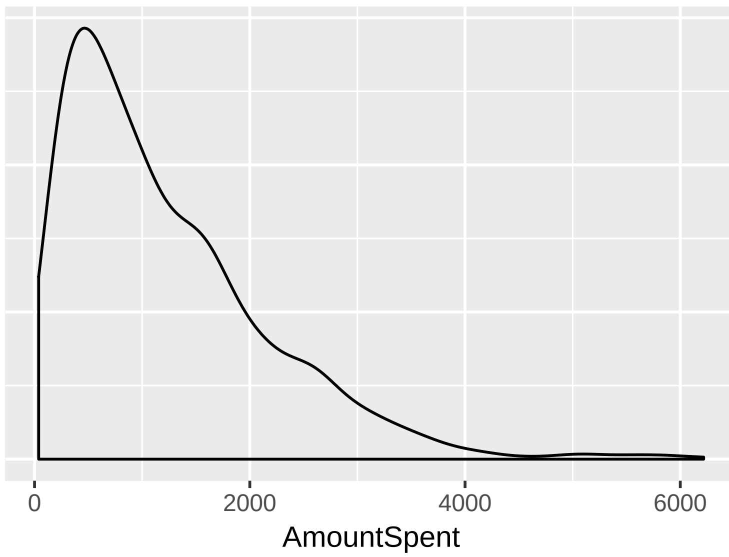
```r
# Generate a summary table for quantitative features
Salary = c(summary(df$Salary), sd(df$Salary))
Children = c(summary(df$Children), sd(df$Children))
Catalogs = c(summary(df$Catalogs), sd(df$Catalogs))
AmountSpent = c(summary(df$AmountSpent), sd(df$AmountSpent))
tbl = rbind(Salary, Children, Catalogs, AmountSpent)
tbl = as.data.frame(tbl)
colnames(tbl)[7] = c("sd")
kable(tbl, caption = "Table 1: Summary of attributes")
```

Table 1: Table 1: Summary of attributes

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	sd
Salary	10100	29980.0	53700	56100.000	77020	168800	30616.314826
Children	0	0.0	1	0.934	2	3	1.051070
Catalogs	6	6.0	12	14.680	18	24	6.622895
AmountSpent	38	488.2	962	1217.000	1688	6217	961.068612

```r
rm(list=c("Salary", "Children", "Catalogs", "AmountSpent", "tbl"))
```

```r
# Explore the density distribution of AmountSpent
no.y = theme(axis.title.y=element_blank(), ## remove clutter on y axis
             axis.text.y=element_blank(),
             axis.ticks.y=element_blank())
ggplot(df, aes(x=AmountSpent)) + geom_density() + no.y
```



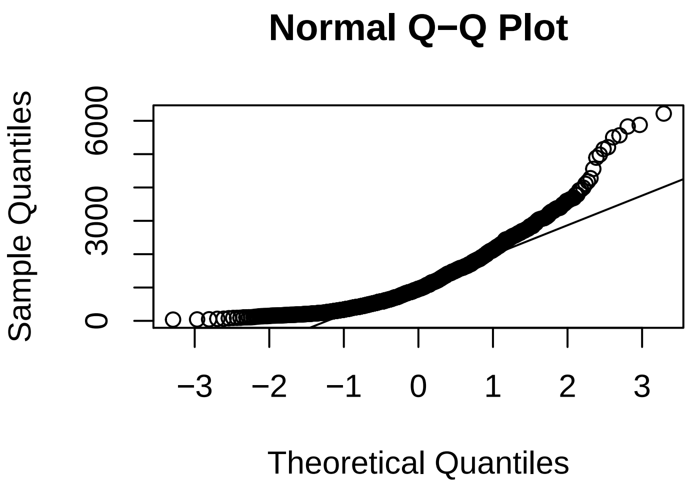

```r
skewness(df$AmountSpent)
```

```
## [1] 1.464872
```

```r
shapiro.test(df$AmountSpent)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$AmountSpent
## W = 0.8784, p-value < 2.2e-16
```

```r
qqnorm(df$AmountSpent)
qqline(df$AmountSpent)
```



Normal Q−Q Plot
Sample Quantiles
0
3000
6000
−3
−2
−1
0
1
2
3
Theoretical Quantiles

```r
# Explore the density distribution of Salary
ggplot(df, aes(x=Salary)) + geom_density() + no.y
```

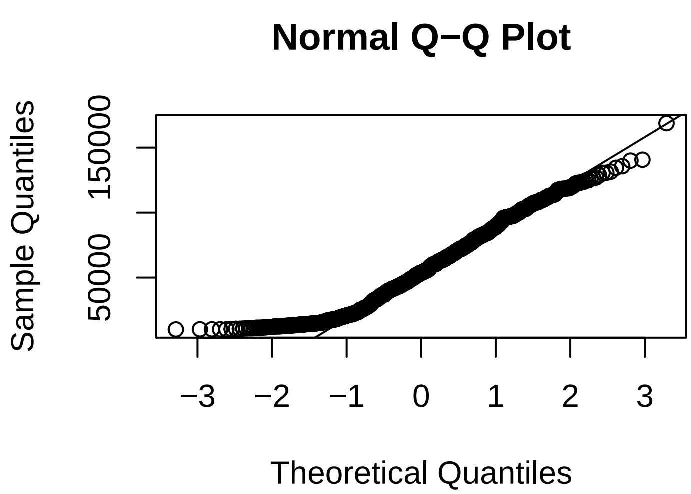

```r
skewness(df$Salary)
```

```
## [1] 0.4178385
```

```r
shapiro.test(df$Salary)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$Salary
## W = 0.96338, p-value = 3.763e-15
```

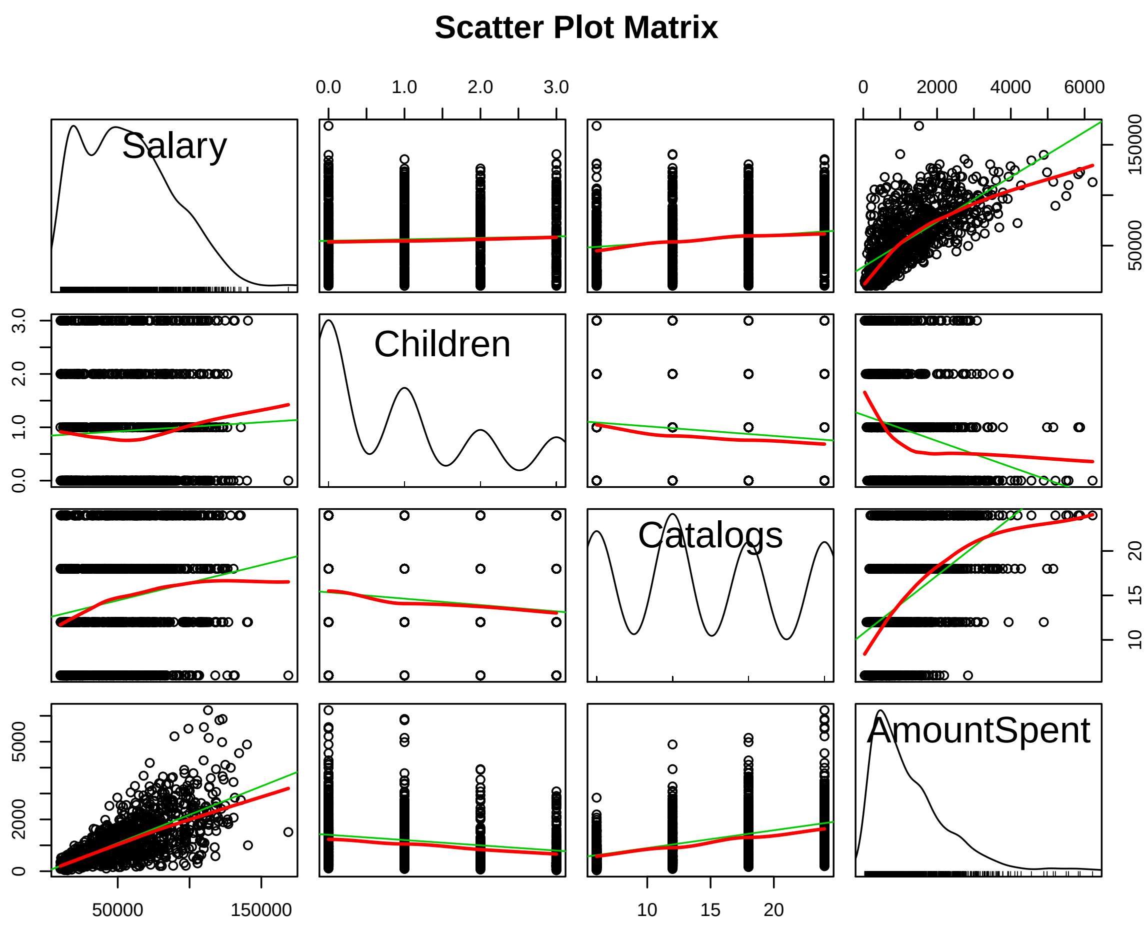
```r
qqnorm(df$Salary)
qqline(df$Salary)
```



```r
# Correlations
df.numeric = df[,sapply(df, is.numeric)]
cor(df.numeric)
```

```
##                 Salary    Children   Catalogs AmountSpent
## Salary      1.00000000  0.04966316  0.1835509   0.6995957
## Children    0.04966316  1.00000000 -0.1134554  -0.2223082
## Catalogs    0.18355086 -0.11345543  1.0000000   0.4726499
## AmountSpent 0.69959571 -0.22230817  0.4726499   1.0000000
```

```r
suppressWarnings(
  scatterplotMatrix(df.numeric, spread=F, lty.smooth=2, main="Scatter Plot Matrix")
)
```




```r
rm(df.numeric)
```

Qualitative Data

A conditional density plot of the response variable for each categorical predictor is generated. The mean and median of the response for each category of each predictor are additionally observed, and the various categories of each predictor are tested for significant differences in their means using ANOVA and pairwise t-testing.

The ANOVA table for AmountSpent by Age demonstrates an F-statistic of 116.7 with a p-value less than 2e-16, and clearly indicates a rejection of the null hypothesis of equal means for all three age groups. The pairwise t-test indicates significant differences in AmountSpent between Young and Middle groups, and Young and Old groups, but there are no significant differences between Middle and Old groups.

The ANOVA table for AmountSpent by Gender demonstrates an F-statistic of 42.32 with a p-value equal to 1.22e-10, and clearly indicates a rejection of the null hypothesis of equal means between Male and Female.

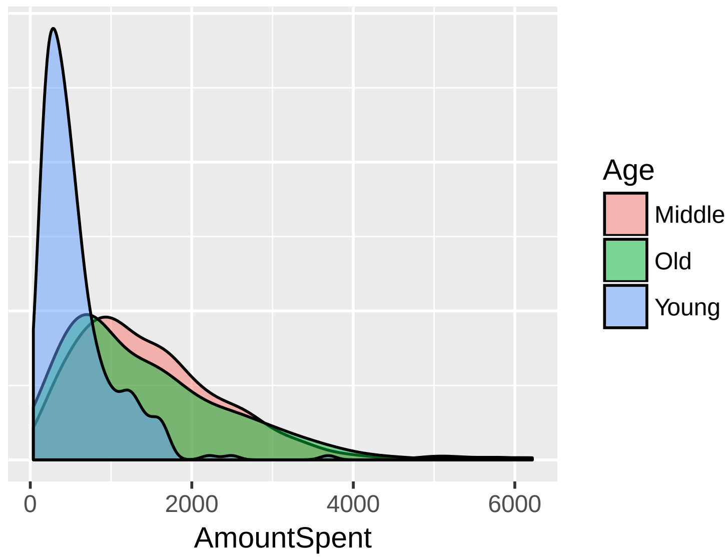
The ANOVA table for AmountSpent by OwnHome demonstrates an F-statistic of 140.1 with a p-value less than 2e-16, and clearly indicates a rejection of the null hypothesis of equal means between Own and Rent.

The ANOVA table for AmountSpent by Married demonstrates an F-statistic of 292.2 with a p-value less than 2e-16, and clearly indicates a rejection of the null hypothesis of equal means between Married and Single.

The ANOVA table for AmountSpent by Location demonstrates an F-statistic of 68.03 with a p-value equal to 5.05e-16, and clearly indicates a rejection of the null hypothesis of equal means between Close and Far.

The ANOVA table for AmountSpent by History demonstrates an F-statistic of 283.2 with a p-value less than 2e-16, and clearly indicates a rejection of the null hypothesis of equal for the four History groups. The pairwise t-test indicates that all mean comparisons are significantly different.

```r
# AmountSpent by Age
ggplot(df, aes(x=AmountSpent, fill=Age)) +
  geom_density(alpha=0.5) + no.y
```




```r
aggregate(AmountSpent~Age, data=df, mean)
```

```
##      Age AmountSpent
## 1 Middle   1501.6909
## 2    Old   1432.1268
## 3  Young    558.6237
```

```r
aggregate(AmountSpent~Age, data=df, median)
```

```
##      Age AmountSpent
## 1 Middle        1320
## 2    Old        1120
## 3  Young         422
```

```r
summary(aov(AmountSpent~Age, data=df))
```

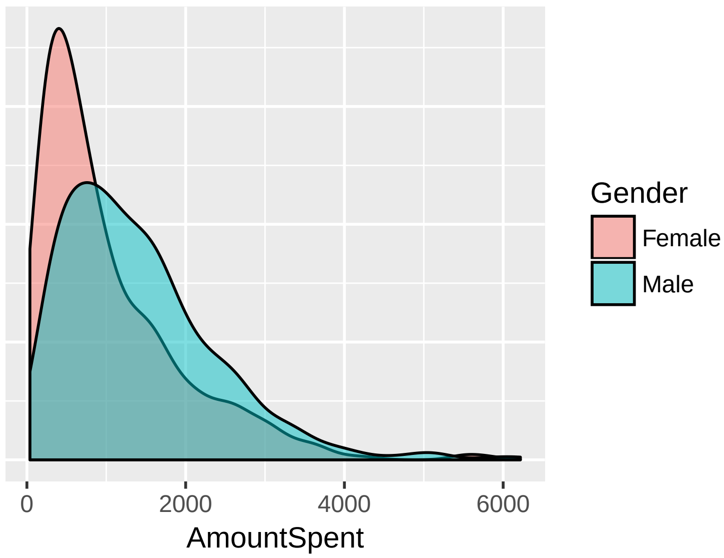
```
##              Df    Sum Sq  Mean Sq F value Pr(>F)
## Age           2 175062951 87531475   116.7 <2e-16 ***
## Residuals   997 747666275   749916
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
pairwise.t.test(df$AmountSpent, df$Age)
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  df$AmountSpent and df$Age
##
##       Middle Old
## Old   0.33   -
## Young <2e-16 <2e-16
##
```

```
## P value adjustment method: holm
```

```r
# AmountSpent by Gender
ggplot(df, aes(x=AmountSpent, fill=Gender)) +
  geom_density(alpha=0.5) + no.y
```




```r
aggregate(AmountSpent~Gender, data=df, mean)
```

```
##   Gender AmountSpent
## 1 Female     1025.34
## 2   Male     1412.85
```

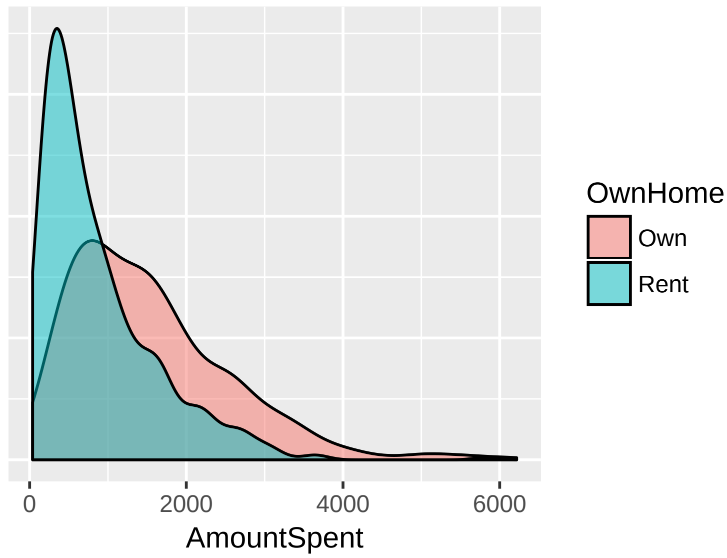
```r
aggregate(AmountSpent~Gender, data=df, median)
```

```
##   Gender AmountSpent
## 1 Female         706
## 2   Male        1216
```

```r
summary(aov(AmountSpent~Gender, data=df))
```

```
##              Df    Sum Sq  Mean Sq F value   Pr(>F)    
## Gender        1  37535649 37535649   42.32 1.22e-10 ***
## Residuals   998 885193576   886968                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#AmountSpent by OwnHome
ggplot(df, aes(x=AmountSpent, fill=OwnHome)) +
  geom_density(alpha=0.5) + no.y
```




```r
aggregate(AmountSpent~OwnHome, data=df, mean)
```

```
##   OwnHome AmountSpent
## 1     Own   1543.1357
## 2    Rent    868.8264
```

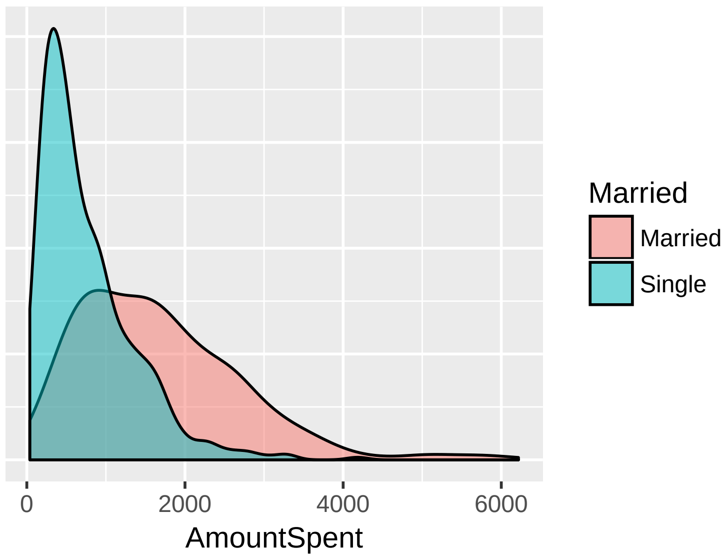
```r
aggregate(AmountSpent~OwnHome, data=df, median)
```

```
##   OwnHome AmountSpent
## 1     Own      1359.5
## 2    Rent       623.0
```

```r
summary(aov(AmountSpent~OwnHome, data=df))
```

```
##              Df    Sum Sq   Mean Sq F value Pr(>F)    
## OwnHome       1 113556827 113556827   140.1 <2e-16 ***
## Residuals   998 809172398    810794                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# AmountSpent by Married
ggplot(df, aes(x=AmountSpent, fill=Married)) +
  geom_density(alpha=0.5) + no.y
```




```r
aggregate(AmountSpent~Married, data=df, mean)
```

```
##   Married AmountSpent
## 1 Married   1672.0697
## 2  Single    757.8133
```

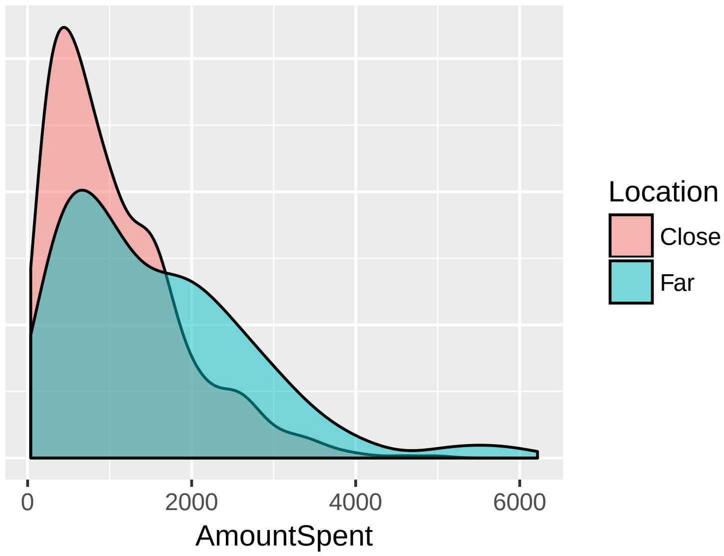
```r
aggregate(AmountSpent~Married, data=df, median)
```

```
##   Married AmountSpent
## 1 Married        1515
## 2  Single         576
```

```r
summary(aov(AmountSpent~Married, data=df))
```

```
##              Df    Sum Sq   Mean Sq F value Pr(>F)
## Married       1 208962879 208962879   292.2 <2e-16 ***
## Residuals   998 713766346    715197
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# AmountSpent by Location
ggplot(df, aes(x=AmountSpent, fill=Location)) +
  geom_density(alpha=0.5) + no.y
```




```r
aggregate(AmountSpent~Location, data=df, mean)
```

```
##   Location AmountSpent
## 1    Close    1061.686
## 2      Far    1596.459
```

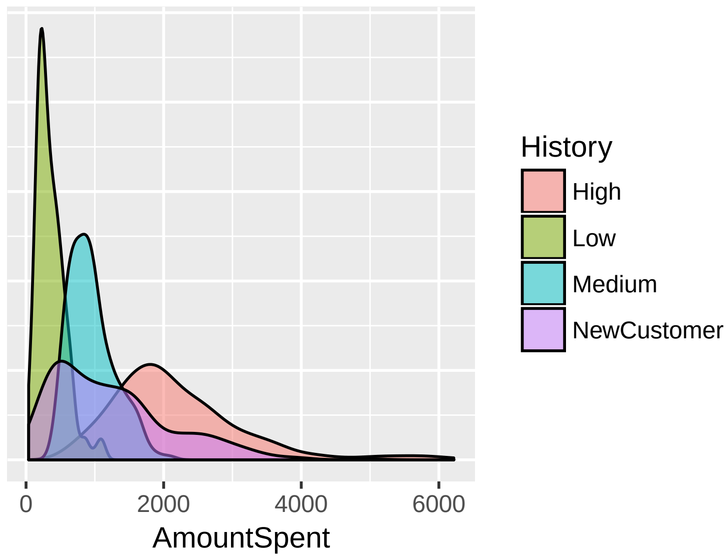
```r
aggregate(AmountSpent~Location, data=df, median)
```

```
##   Location AmountSpent
## 1    Close       858.5
## 2      Far      1317.0
```

```r
summary(aov(AmountSpent~Location, data=df))
```

```
##              Df    Sum Sq  Mean Sq F value   Pr(>F)    
## Location      1  58883662 58883662   68.03 5.05e-16 ***
## Residuals   998 863845563   865577                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# AmountSpent by History
ggplot(df, aes(x=AmountSpent, fill=History)) +
  geom_density(alpha=0.5) + no.y
```




```r
aggregate(AmountSpent~History, data=df, mean)
```

```
##       History AmountSpent
## 1        High   2186.1373
## 2         Low    357.0870
## 3      Medium    950.4009
## 4 NewCustomer   1239.9010
```

```r
aggregate(AmountSpent~History, data=df, median)
```

```
##       History AmountSpent
## 1        High      1974.0
## 2         Low       305.5
## 3      Medium       894.0
## 4 NewCustomer      1079.0
```

```r
summary(aov(AmountSpent~History, data=df))
```

```
##              Df    Sum Sq   Mean Sq F value Pr(>F)
## History       3 424803261 141601087   283.2 <2e-16 ***
## Residuals   996 497925964    499926
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
pairwise.t.test(df$AmountSpent, df$History)
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  df$AmountSpent and df$History
##
##             High    Low     Medium
## Low         < 2e-16 -       -
```

```
## Medium      < 2e-16 < 2e-16 -
## NewCustomer < 2e-16 < 2e-16 5.4e-06
##
## P value adjustment method: holm
```

Regression Analysis

Linear Regression

The linear regression model of AmountSpent against all predictors is statistically significant and accounts for 74.76% of the variance in AmountSpent. The introduction of a penalty for the number of estimated coefficients results in this model explaining 74.46% of the variance in AmountSpent. The leave-one-out cross-validation demonstrates a root mean square error of 489.30, and indicates prediction errors in the magnitude of hundreds of dollars. Coefficients are determined significantly different from zero at the $p < 0.001$ level. Hence, the coefficients for AgeOld, AgeYoung, GenderMale, OwnHomeRent, MarriedSingle, and HistoryNewCustomer are not significant. The coefficients for LocationFar, Salary, Children, HistoryLow, HistoryMedium, and Catalogs were found to be significant.

Evaluating linear regression models with various subsets of predictors reveals that while the linear model including all 9 predictors explains 74.76% of the variance in AmountSpent, models including only four or five predictors achieve comparable performance. A linear regression model of AmountSpent against Location, Salary, Children, and Catalogs results in an R-squared of 0.7148 and root mean square error of 516.25. A linear regression model additionally including History as a predictor results in an R-squared of 0.7462 and root mean square error of 488.34. The detailed results are included below.

```
# Function to compute RMSE via cross-validation (leave-one-out)
cross.val.rmse = function(data, response, formula) {
  n = length(data[,response])
  diff = NULL
  for(k in 1:n) {
    train = c(1:n)
    train = train[train != k]
    model = lm(formula, data=data[train,])
    predicted = predict(model, newdat=data[-train,])
    observed = data[-train, response]
    diff[k] = observed - predicted
  }
  return(sqrt(mean(diff^2))) ## return RMSE
}

# Linear regression model of AmountSpent against all predictors
f = AmountSpent~.
summary(lm(f, df))
```

```
##
## Call:
## lm(formula = f, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1711.44  -292.41   -17.56   237.87  2876.91
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -285.74892  116.39444  -2.455   0.0143 *
## AgeOld               63.36828   47.79586   1.326   0.1852
## AgeYoung              8.90120   49.70059   0.179   0.8579
## GenderMale          -46.99837   32.85192  -1.431   0.1529
## OwnHomeRent         -16.63382   36.64327  -0.454   0.6500
## MarriedSingle        32.74314   44.54067   0.735   0.4624
```

```
## LocationFar         436.50575   35.92138  12.152  < 2e-16 ***
## Salary                0.01920    0.00103  18.652  < 2e-16 ***
## Children           -162.73555   18.00348  -9.039  < 2e-16 ***
## HistoryLow         -352.89534   65.57529  -5.382 9.23e-08 ***
## HistoryMedium      -404.41014   52.94420  -7.638 5.19e-14 ***
## HistoryNewCustomer    6.99218   51.32915   0.136   0.8917
## Catalogs             41.86880    2.45796  17.034  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 485.7 on 987 degrees of freedom
## Multiple R-squared:  0.7476, Adjusted R-squared:  0.7446
## F-statistic: 243.7 on 12 and 987 DF,  p-value: < 2.2e-16
```

```r
cross.val.rmse(df, "AmountSpent", f)
```

```
## [1] 489.3011
```

```r
# Explore linear regression models including various subsets of predictors
models = summary(regsubsets(AmountSpent~., data=df, nbest=1, nvmax=ncol(df)-1))
tbl = cbind(models$which, models$rsq, models$adjr2)[,-1]
tbl = as.data.frame(tbl)
colnames(tbl)[13:14] = c("R2", "Adj.R2")
tbl
```

```
##   AgeOld AgeYoung GenderMale OwnHomeRent MarriedSingle LocationFar Salary
## 1      0        0          0           0             0           0      1
## 2      0        0          0           0             0           0      1
## 3      0        0          0           0             0           1      1
## 4      0        0          0           0             0           1      1
## 5      0        0          0           0             0           1      1
## 6      0        0          0           0             0           1      1
## 7      1        0          0           0             0           1      1
## 8      1        0          1           0             0           1      1
## 9      1        0          1           0             1           1      1
##   Children HistoryLow HistoryMedium HistoryNewCustomer Catalogs        R2
## 1        0          0             0                  0        0 0.4894342
## 2        0          0             0                  0        1 0.6120659
## 3        0          0             0                  0        1 0.6662321
## 4        1          0             0                  0        1 0.7148385
## 5        1          0             1                  0        1 0.7314816
## 6        1          1             1                  0        1 0.7461949
## 7        1          1             1                  0        1 0.7469350
## 8        1          1             1                  0        1 0.7474284
## 9        1          1             1                  0        1 0.7475789
##      Adj.R2
## 1 0.4889226
## 2 0.6112877
## 3 0.6652268
## 4 0.7136921
## 5 0.7301309
## 6 0.7446613
## 7 0.7451492
## 8 0.7453894
## 9 0.7452841
```

```r
rm(list=c("models", "tbl"))

# Linear regression model of AmountSpent against Location, Salary, Children, Catalogs
f = AmountSpent~Location+Salary+Children+Catalogs
summary(lm(f, df))
```

```
##
## Call:
## lm(formula = f, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1730.96  -329.39   -33.85   232.17  2864.85
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.398e+02  4.959e+01  -10.88   <2e-16 ***
## LocationFar  5.081e+02  3.622e+01   14.03   <2e-16 ***
## Salary       2.089e-02  5.431e-04   38.47   <2e-16 ***
## Children    -2.035e+02  1.562e+01  -13.02   <2e-16 ***
## Catalogs     4.272e+01  2.544e+00   16.79   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 514.2 on 995 degrees of freedom
## Multiple R-squared:  0.7148, Adjusted R-squared:  0.7137
## F-statistic: 623.6 on 4 and 995 DF,  p-value: < 2.2e-16
```

```r
cross.val.rmse(df, "AmountSpent", f)
```

```
## [1] 516.2468
```

```r
# Linear regression model of AmountSpent against Location, Salary, Children, Catalogs,
# and History
f = AmountSpent~Location+Salary+Children+Catalogs+History
summary(lm(f, df))
```

```
##
## Call:
## lm(formula = f, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1651.7  -287.9   -11.6   239.7  2913.2
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -2.446e+02  7.939e+01  -3.081  0.00212 **
## LocationFar         4.363e+02  3.589e+01  12.156  < 2e-16 ***
## Salary              1.871e-02  6.791e-04  27.551  < 2e-16 ***
## Children           -1.694e+02  1.665e+01 -10.179  < 2e-16 ***
## Catalogs            4.165e+01  2.453e+00  16.979  < 2e-16 ***
## HistoryLow         -3.509e+02  6.544e+01  -5.362 1.02e-07 ***
## HistoryMedium      -4.099e+02  5.241e+01  -7.821 1.34e-14 ***
## HistoryNewCustomer -1.875e+00  5.110e+01  -0.037  0.97073
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 485.9 on 992 degrees of freedom
## Multiple R-squared:  0.7462, Adjusted R-squared:  0.7444
## F-statistic: 416.6 on 7 and 992 DF,  p-value: < 2.2e-16
```

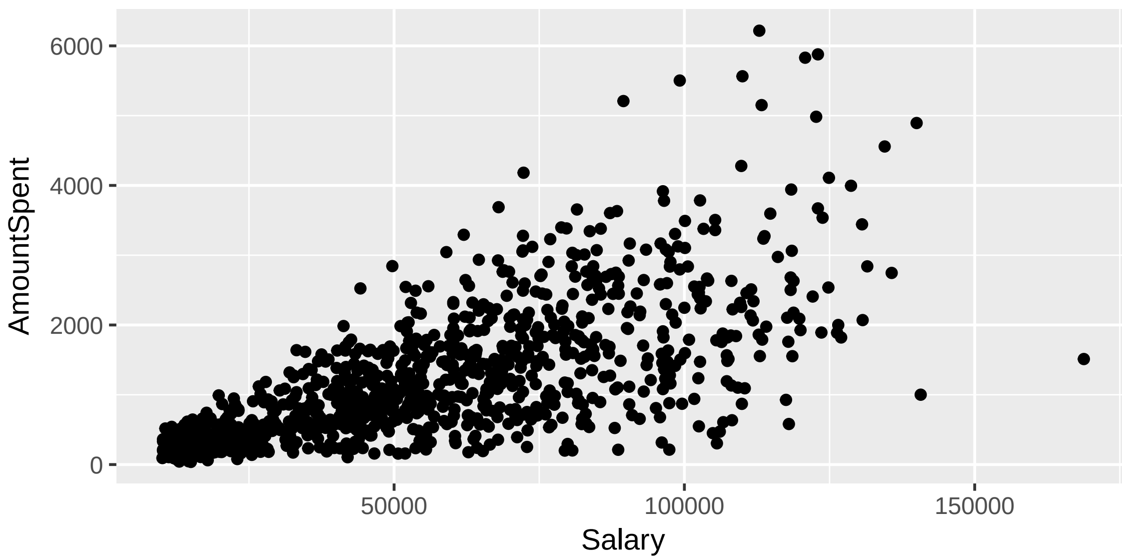
```
cross.val.rmse(df, "AmountSpent", f)
```

```
## [1] 488.3369
```

Polynomial Regression

The scatter plot of AmountSpent against Salary appears to somewhat follow a quadratic trend line that spreads as Salary increases. However, plotting the training sample prediction error and cross-validation prediction error of polynomial regression over various degrees suggests that polynomial regression would not offer any significant gains. An instance of polynomial regression that models Salary as a 2nd degree polynomial, Catalogs as a 3rd degree polynomial, and additionally includes Children, Location, and History as linear terms results in a model that accounts for 74.68% of the variance and produces a root mean square error of 490.40. This performance is comparable but not superior to the linear model. Furthermore, the summary of this polynomial regression model indicates that the coefficients of the quadratic and cubic terms are likely to not be significantly different from zero. These results indicate that the linear regression model is superior.

```
ggplot(df, aes(y=AmountSpent, x=Salary))+
    geom_point()
```



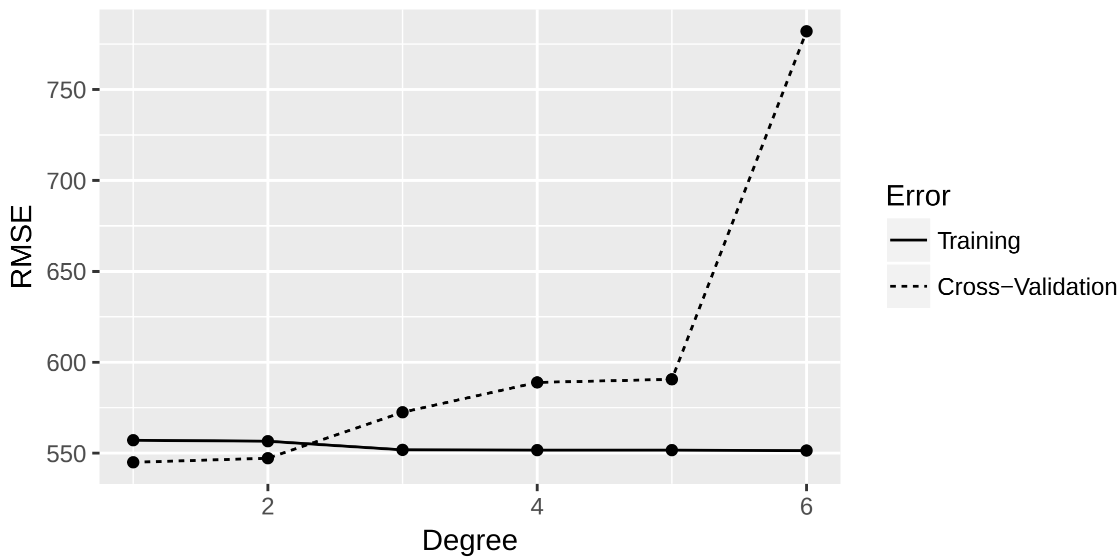

```
# Function for comparing in-sample and out-of-sample error of
# polynomial regression over various degrees
cross.val.poly.reg =
  function(data, response, poly.var, lin.var, deg=12, train.set=0.5) {
    ## measure performance in terms of RMSE
    rmse = function(y, p) { return(sqrt(mean((y - p)^2))) }
    performance = data.frame()
    ## split data into a training set and test set for cross-validation
    n = length(data[,response])
    train = sort(sample(1:n, round(train.set*n)))
    formula = as.formula(paste(response,"~poly(",poly.var,", degree=d)+",lin.var,sep=""))

    for (d in 1:deg) {
      poly.fit = lm(formula, data=data[train,])
      performance = rbind(performance,
                          data.frame(Degree=d, Error="Training",
                                     RMSE = rmse(data[train,response],
                                                 predict(poly.fit))
                                     )
                          )
```

```r
      performance = rbind(performance,
                          data.frame(Degree=d, Error="Cross-Validation",
                                     RMSE = rmse(data[-train,response],
                                                 predict(poly.fit, newdata=data[-train,]))
                                     )
                          )
      }

    ## Plot the performance of polynomial regression models for each degree
    require("ggplot2")
    require("scales")
    ggplot(performance , aes(x=Degree, y=RMSE, linetype=Error)) +
      geom_point() + geom_line() + scale_y_continuous(labels=comma)
}

set.seed(13)
cross.val.poly.reg(df, "AmountSpent", "Salary+Catalogs", "Children+Location+History", deg=6)
```




```r
# An instance of polynomial regression (summary and out-of-sample RMSE)
f = AmountSpent~poly(Salary, degree=2)+poly(Catalogs, degree=3)+Children+Location+History
summary(lm(f, df))
```

```
##
## Call:
## lm(formula = f, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1750.81  -286.98   -14.34   244.53  2893.39
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   1415.367     40.776  34.711  < 2e-16 ***
## poly(Salary, degree = 2)1    18123.655    665.798  27.221  < 2e-16 ***
## poly(Salary, degree = 2)2      238.242    520.290   0.458    0.647
```

```
## poly(Catalogs, degree = 3)1  8726.328    514.052  16.976  < 2e-16 ***
## poly(Catalogs, degree = 3)2   541.014    489.176   1.106    0.269
## poly(Catalogs, degree = 3)3  -511.943    489.407  -1.046    0.296
## Children                     -170.132     16.696 -10.190  < 2e-16 ***
## LocationFar                   434.965     35.991  12.086  < 2e-16 ***
## HistoryLow                   -354.471     67.168  -5.277 1.61e-07 ***
## HistoryMedium                -398.838     52.946  -7.533 1.12e-13 ***
## HistoryNewCustomer              0.822     51.514   0.016    0.987
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 486 on 989 degrees of freedom
## Multiple R-squared:  0.7468, Adjusted R-squared:  0.7443
## F-statistic: 291.7 on 10 and 989 DF,  p-value: < 2.2e-16
```

```r
cross.val.rmse(df, "AmountSpent", f)
```

```
## [1] 490.3985
```

LASSO

Although regularization is not necessary for modeling this data, it is interesting to examine the variable selection process and additionally confirm the previously selected predictors. The graph of the LASSO estimates as a function of the shrinkage illustrates the order in which variables enter the model as one relaxes the constraint on the L1 norm of their estimates. The first variable to enter is Salary, then Catalogs, followed by HistoryLow, Location, and Children, with the rest of the variables far off. Cross-validation (10-fold) indicates that the error is minimized at 0.8 of the final L1 norm. The results of LASSO confirm the selected set of predictors, but regularization would not offer any significant performance gains.

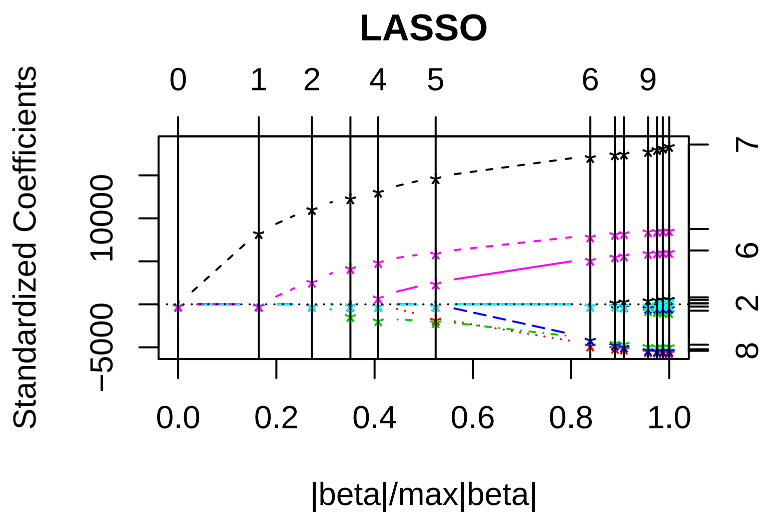
```
x = model.matrix(AmountSpent~., data=df)
x = x[,-1] ## remove the intercept
lasso = lars(x = x, y = df$AmountSpent, trace = TRUE)
```

```
## LASSO sequence
## Computing X'X .....
## LARS Step 1 :     Variable 7     added
## LARS Step 2 :     Variable 12    added
## LARS Step 3 :     Variable 9     added
## LARS Step 4 :     Variable 6     added
## LARS Step 5 :     Variable 8     added
## LARS Step 6 :     Variable 10    added
## LARS Step 7 :     Variable 1     added
## LARS Step 8 :     Variable 4     added
## LARS Step 9 :     Variable 3     added
## LARS Step 10 :    Variable 5     added
## LARS Step 11 :    Variable 11    added
## LARS Step 12 :    Variable 2     added
## Computing residuals, RSS etc .....
```

```
lasso
```

```
##
## Call:
## lars(x = x, y = df$AmountSpent, trace = TRUE)
## R-squared: 0.748
## Sequence of LASSO moves:
##      Salary Catalogs HistoryLow LocationFar Children HistoryMedium AgeOld
## Var       7       12          9           6        8            10      1
## Step      1        2          3           4        5             6      7
##      OwnHomeRent GenderMale MarriedSingle HistoryNewCustomer AgeYoung
## Var            4          3             5                 11        2
## Step           8          9            10                 11       12
```

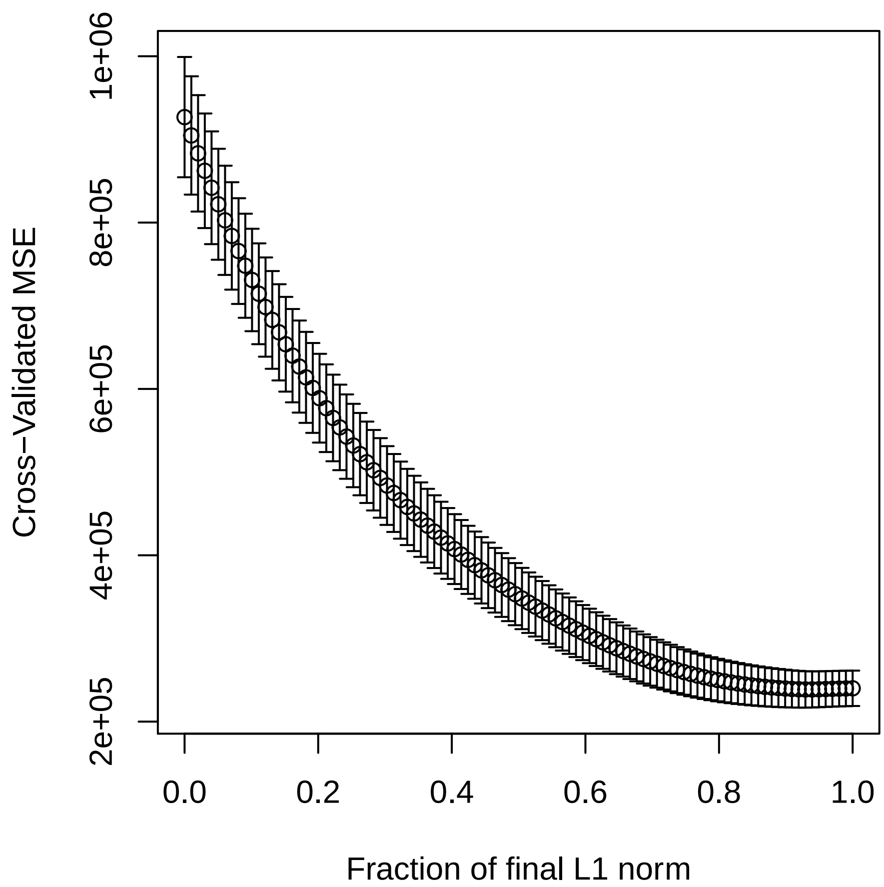
```r
plot(lasso)
```




```r
coef(lasso, s=c(.20, .40, .60, .80, 1.0), mode="fraction")
```

```
##        AgeOld AgeYoung GenderMale OwnHomeRent MarriedSingle LocationFar
## [1,]  0.00000 0.000000    0.00000     0.00000       0.00000     0.00000
## [2,]  0.00000 0.000000    0.00000     0.00000       0.00000    57.66378
## [3,]  0.00000 0.000000    0.00000     0.00000       0.00000   225.65223
## [4,]  0.00000 0.000000    0.00000     0.00000       0.00000   347.93455
## [5,] 63.36828 8.901204  -46.99837   -16.63382      32.74314   436.50575
##           Salary    Children HistoryLow HistoryMedium HistoryNewCustomer
## [1,] 0.009684991    0.00000     0.0000        0.0000            0.00000
## [2,] 0.013594695    0.00000  -122.5871        0.0000            0.00000
## [3,] 0.015937261  -70.50154  -181.0287      -73.4508            0.00000
## [4,] 0.017552213 -127.97841  -278.5213     -267.7573            0.00000
## [5,] 0.019203807 -162.73555  -352.8953     -404.4101            6.99218
##        Catalogs
## [1,]  4.435143
## [2,] 23.822171
## [3,] 31.245190
## [4,] 37.287113
## [5,] 41.868804
```

```r
cv.lars(x=x, y=df$AmountSpent, K=10)
```



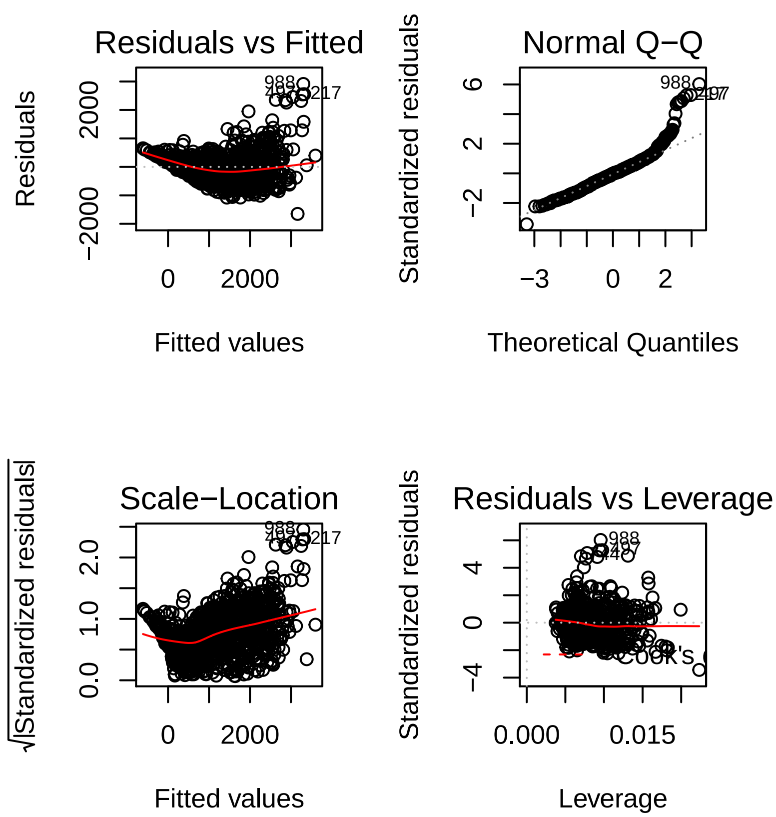

```r
rm(list=c("x", "lasso"))
```

Validation of Linear Regression

The normal probability plot of the standardized residuals indicates the non-normality of their distribution, and violates the assumption of normality. There is no a priori reason to believe that the amount spent by one customer is influenced by the amount spent by another customer, so the assumption of independence is met. The scatter plot of residuals against fitted values presents somewhat of a curved line that transitions into random noise, indicating that the model may not meet the assumption of linearity and a term may need to be added to the model. The scatter plot of scale against location presents a random band around a curved line, indicating the violation of the assumption of homoscedasticity.

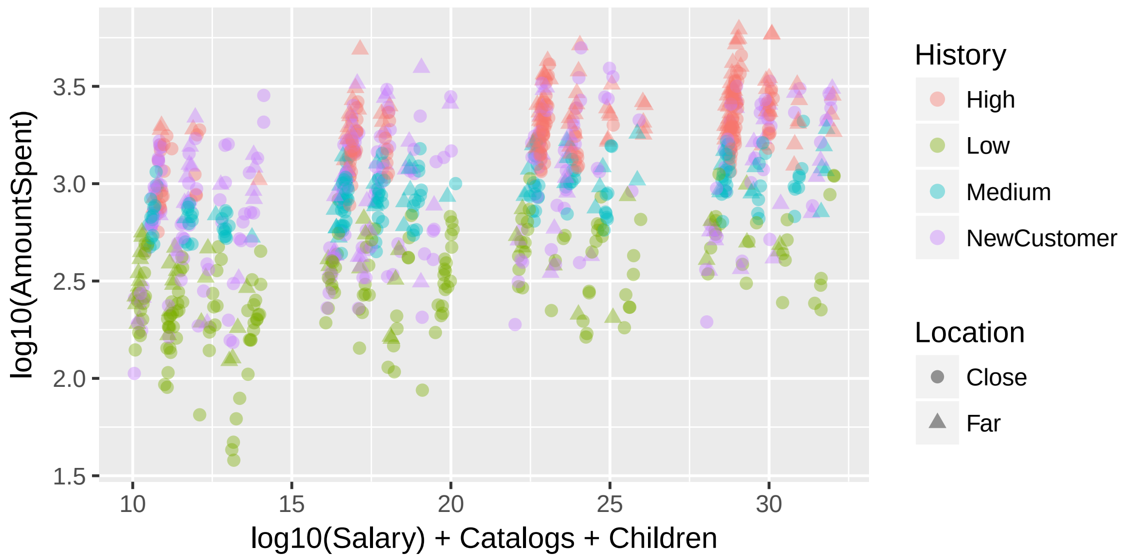
The exploration of the density distributions of quantitative features previously revealed that AmountSpent is highly skewed and Salary is moderately skewed (both in the positive direction). Applying a log transformation to these monetary features is justified given that the assumptions of linear regression do not hold. A log base-10 transformation of these features offers more organization for visual inspection of the graphed data. This transformation results in all assumptions of linear regression being satisfied. The linear regression model using transformed monetary data accounts for 87.33% of the variation, but the model estimates become more difficult to interpret.

```r
lin.model = lm(AmountSpent~Salary+Catalogs+Children+Location+History, df)
par(mfrow=c(2,2)) ## format plots in 2 by 2 figure
plot(lin.model)
```




```r
par(mfrow=c(1,1)) ## reset format
```

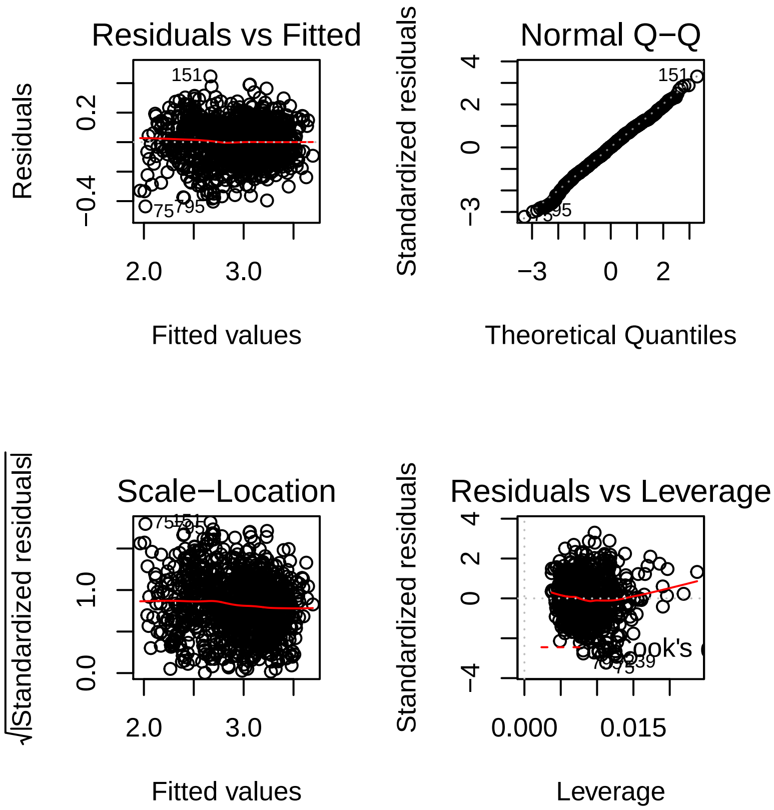
```r
ggplot(df, aes(x=log10(Salary)+Catalogs+Children, y=log10(AmountSpent),
               shape=Location, color=History)) +
  geom_point(alpha = 0.4, size=2) +
  scale_shape_manual(values=c(16,17))
```



```r
lin.model = lm(log10(AmountSpent)~log10(Salary)+Catalogs+Children+Location+History, df)
summary(lin.model)
```

```
##
## Call:
## lm(formula = log10(AmountSpent) ~ log10(Salary) + Catalogs +
##     Children + Location + History, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.43569 -0.08906  0.00326  0.09409  0.44572
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -1.192096   0.102997 -11.574  < 2e-16 ***
## log10(Salary)        0.848652   0.020757  40.886  < 2e-16 ***
## Catalogs             0.016565   0.000686  24.148  < 2e-16 ***
## Children            -0.082253   0.004622 -17.796  < 2e-16 ***
## LocationFar          0.152145   0.010035  15.162  < 2e-16 ***
## HistoryLow          -0.189112   0.018449 -10.251  < 2e-16 ***
## HistoryMedium       -0.066881   0.014055  -4.758 2.24e-06 ***
## HistoryNewCustomer  0.058407   0.014129   4.134 3.87e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1358 on 992 degrees of freedom
## Multiple R-squared:  0.8733, Adjusted R-squared:  0.8724
## F-statistic: 976.7 on 7 and 992 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2)) ## format plots in 2 by 2 figure
plot(lin.model)
```



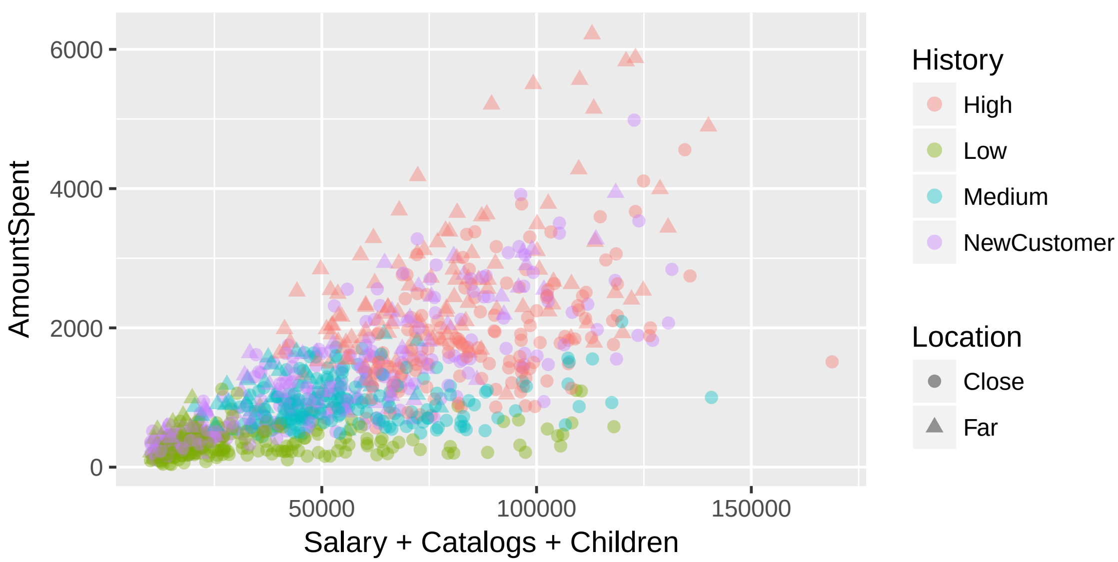

```r
par(mfrow=c(1,1)) ## reset format
```

Results

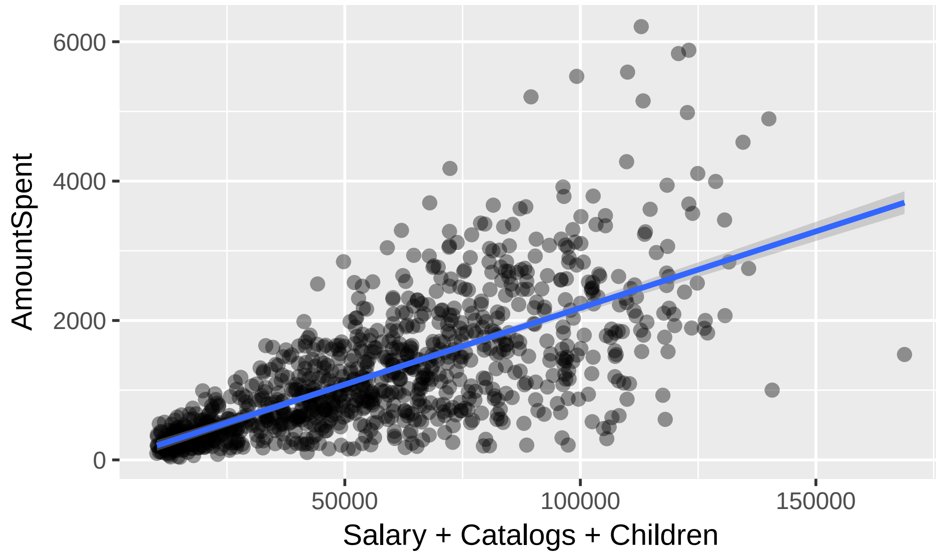
The regression analysis indicates that customer spending behavior can be predicted by their salary, the number of catalogs they have received, the number of children they have, whether they live close or far to the nearest competitor, and their history of previous purchase volume. Customers that live far from the closest competitor or have a history of high previous purchase volume tend to spend more. Salary is the strongest predictor of spending behavior, followed by the number of catalogs the customer has received and location. The strength of a predictor is determined with respect to the increase in error resulting from its exclusion in the model.

The linear regression model describing these findings accounts for 74.62% of the variation in customer spending behavior with a root mean square error of $488.34. A model that represents the amount spent and salary of each customer using base-10 logarithms has improved performance and accounts for 87.33% of the variation, but the model estimates become more difficult to interpret. These findings suggest that it may be beneficial to target customers that live close to competitors with more catalogs. Customers that live close to competitors with a low or medium previous purchase volume history tend to demonstrate low spending behavior regardless of salary. It may be advantageous to direct marketing efforts toward these customers in an attempt to boost sales.

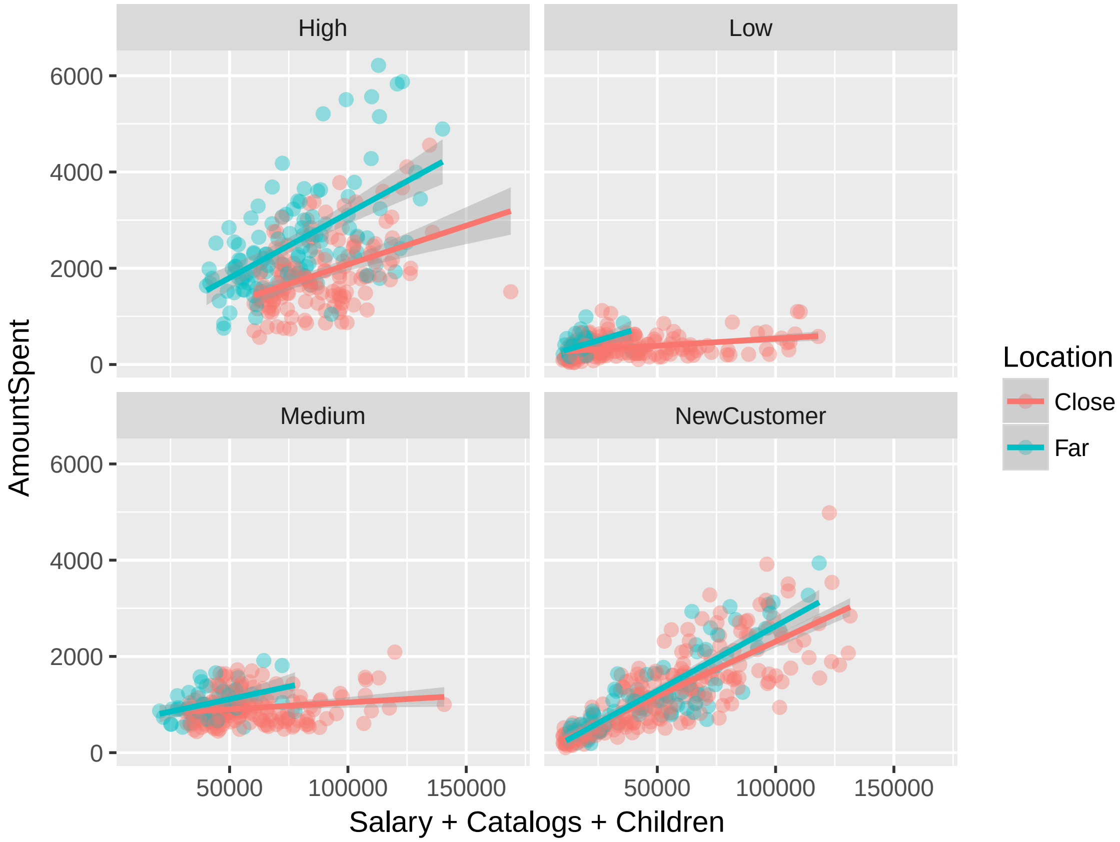
```r
# Graph of AmountSpent against Salary, Catalogs, Children, Location, and History
ggplot(df, aes(x=Salary+Catalogs+Children, y=AmountSpent,
               shape=Location, color=History)) +
  geom_point(alpha = 0.4, size=2) +
  scale_shape_manual(values=c(16,17))
```



```r
# Linear Regression Trend Line
ggplot(df, aes(x=Salary+Catalogs+Children, y=AmountSpent)) +
  geom_point(alpha = 0.4, size=2) +
  geom_smooth(method="lm")
```



```r
# Linear Regression Trend Lines by Location for AmountSpent against
# Salary, Catalogs, and Children (faceted on History)
ggplot(df, aes(x=Salary+Catalogs+Children, y=AmountSpent, color=Location)) +
  geom_point(alpha = 0.4, size=2) +
  geom_smooth(method="lm") +
  facet_wrap(~History)
```



```r
# Observation of the increase in RMSE as each variable is excluded
# from the model to determine the most important predictor

## baseline
cross.val.rmse(df, "AmountSpent", AmountSpent~Salary+Catalogs+Children+Location+History)
```

```
## [1] 488.3369
```

```r
## exclude Salary
cross.val.rmse(df, "AmountSpent", AmountSpent~Catalogs+Children+Location+History)
```

```
## [1] 647.2386
```

```r
## exclude Catalogs
cross.val.rmse(df, "AmountSpent", AmountSpent~Salary+Children+Location+History)
```

```
## [1] 554.0768
```

```r
## exclude Children
cross.val.rmse(df, "AmountSpent", AmountSpent~Salary+Catalogs+Location+History)
```

```
## [1] 512.6858
```

```r
## exclude Location
cross.val.rmse(df, "AmountSpent", AmountSpent~Salary+Catalogs+Children+History)
```

```
## [1] 522.6414
```

```r
## exclude History
cross.val.rmse(df, "AmountSpent", AmountSpent~Salary+Catalogs+Children+Location)
```

```
## [1] 516.2468
```