

SYR-MBA FIN 654 Financial Analytics Project 1

Khan and Mohamed

January 22, 2017

Problem

What is the influence of various terms and conditions on the lending rate of fully committed commercial loans with maturities greater than one year?

Data Preparation

Data is from St. Louis Federal Reserve Economic Data FRED.

Variable	Description	Units
rate	Weighted-Average Effective Loan Rate by Time that Pricing Terms Were Set and by Commitment, Before Survey Week, More than 365 Days, All Commercial Banks (EETP365PDNQ)	percent
prepay	Percent of Value of Loans Subject to Prepayment Penalty by Time that Pricing Terms Were Set and by Commitment, Before Survey Week, More than 365 Days, All Commercial Banks (EPTP365PDNQ)	percent
maturity	Weighted-Average Maturity/Repricing Interval in Days by Time that Pricing Terms Were Set and by Commitment, Before Survey Week, More than 365 Days, All Commercial Banks (EITP365PDNQ)	days
size	Average Loan Size by Time that Pricing Terms Were Set and by Commitment, Before Survey Week, More than 365 Days, All Commercial Banks (EATP365PDNQ)	thousandsUSD
volume	Total Value of Loans by Time that Pricing Terms Were Set and by Commitment, Before Survey Week, More than 365 Days, All Commercial Banks (EVTP365PDNQ)	millionsUSD

Data Analysis

Data Exploration

```
# Read data and Assign the data to a variable called x.data.
# Examine the first and last five entries.
x.data <- read.csv("data/commloans.csv")
x.data <- na.omit(x.data)
# Run a summary of the data set.
head(x.data, n=5)
```

```
##      date prepaypenalty maturity rate size volume
## 1  4/1/2003          16.5       124 3.77  449  11406
## 2  7/1/2003          18.1        70 3.09  356  14586
## 3 10/1/2003          44.9        48 2.83  532  21022
## 4  1/1/2004          30.4        87 3.06  602  21472
```

```
## 5 4/1/2004 23.5 68 2.97 600 22359
```

```
tail(x.data, n=5)
```

```
##      date prepaypenalty maturity rate size volume
## 50 7/1/2015      16.9      76 2.30 1405 30586
## 51 10/1/2015     11.7      77 2.31 1534 36840
## 52 1/1/2016      13.6      66 2.43 1317 36316
## 53 4/1/2016      20.6      93 2.63 1227 24803
## 54 7/1/2016      14.5      66 2.41 1460 40682
```

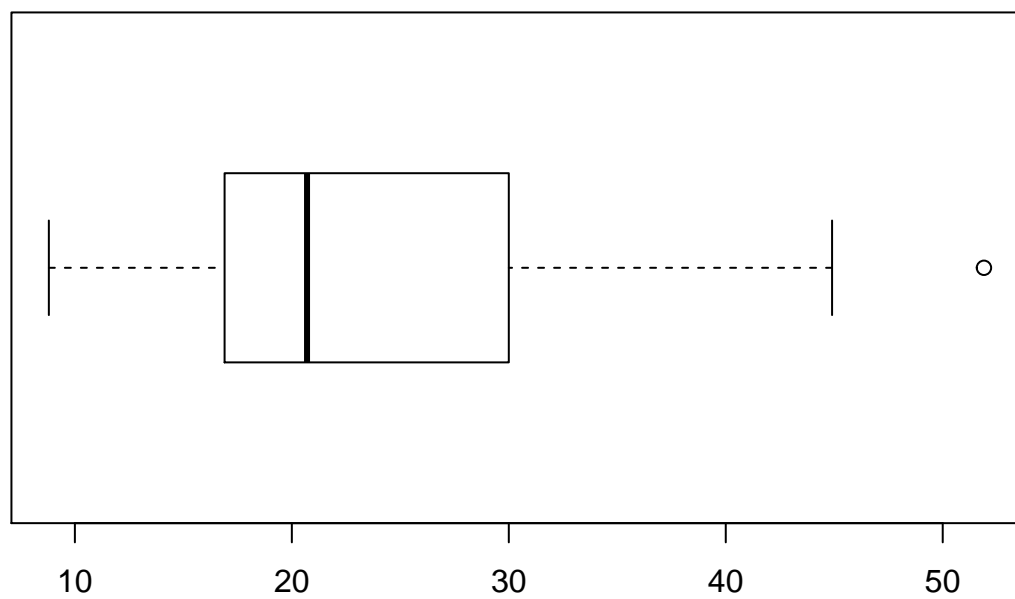
```
summary(x.data)
```

```
##      date      prepaypenalty      maturity      rate
## 1/1/2004: 1  Min.   : 8.80  Min.   : 40.00  Min.   :2.240
## 1/1/2005: 1  1st Qu.:16.95  1st Qu.: 68.25  1st Qu.:2.482
## 1/1/2006: 1  Median :20.70  Median : 89.00  Median :2.825
## 1/1/2007: 1  Mean    :23.06  Mean    : 95.28  Mean    :3.652
## 1/1/2008: 1  3rd Qu.:29.93  3rd Qu.:112.25  3rd Qu.:4.197
## 1/1/2009: 1  Max.    :51.90  Max.    :396.00  Max.    :7.410
## (Other) :48
##      size      volume
## Min.   : 356.0  Min.   :11406
## 1st Qu.: 639.5  1st Qu.:15451
## Median : 824.5  Median :18670
## Mean    : 881.7  Mean    :20824
## 3rd Qu.:1017.8  3rd Qu.:24258
## Max.    :1715.0  Max.    :40682
##
```

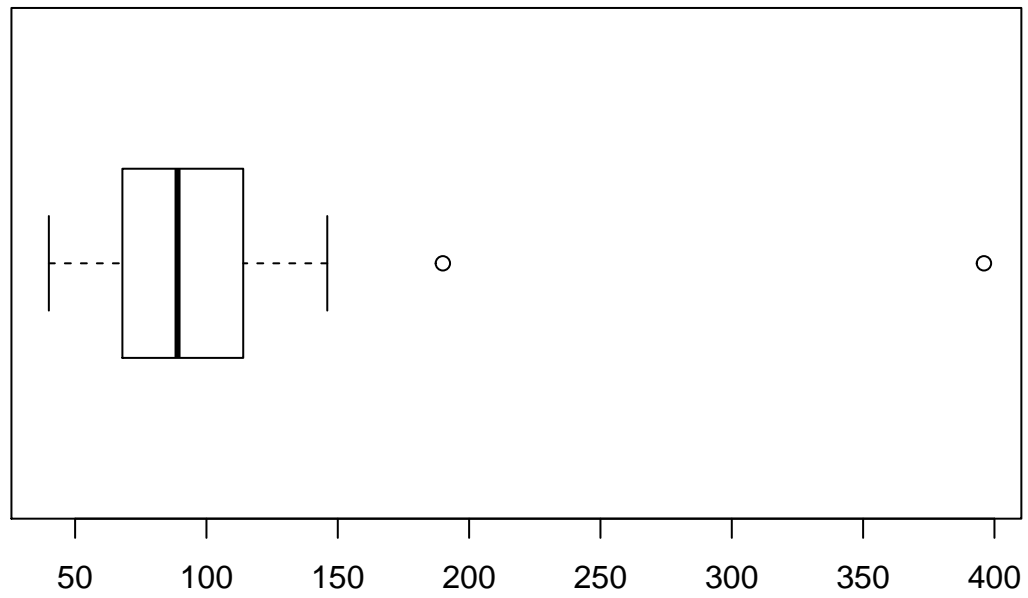
```
#
# What anomalies appear based on these procedures?
#
```

The min and max for the different attributes are quite large. The median is closer to first quartile Q1 and there are likely outliers, indicating that the data (for each variabe) is skewed to the right. This is borne out by the box-and-whisker charts.

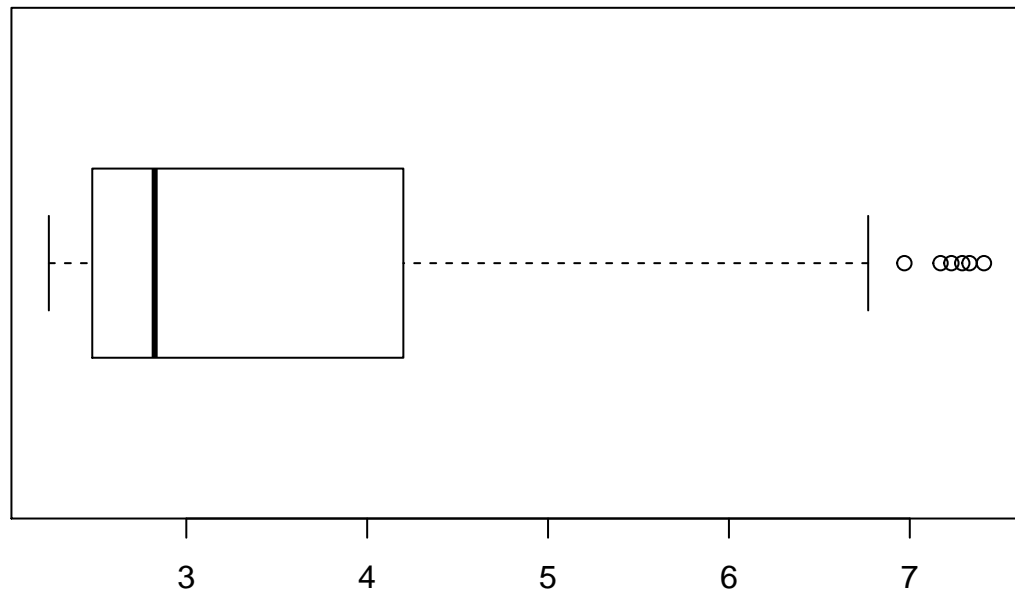
```
boxplot(x.data[2], names, horizontal=TRUE) # ppp = prepaypenalty
```



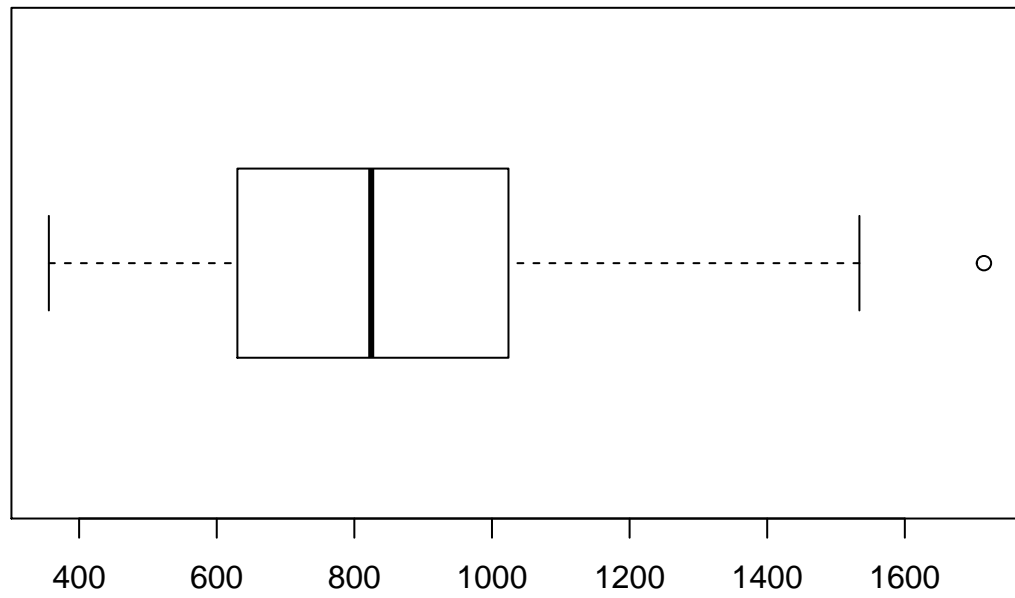
```
boxplot(x.data[3], names, horizontal=TRUE) # mty = maturity
```



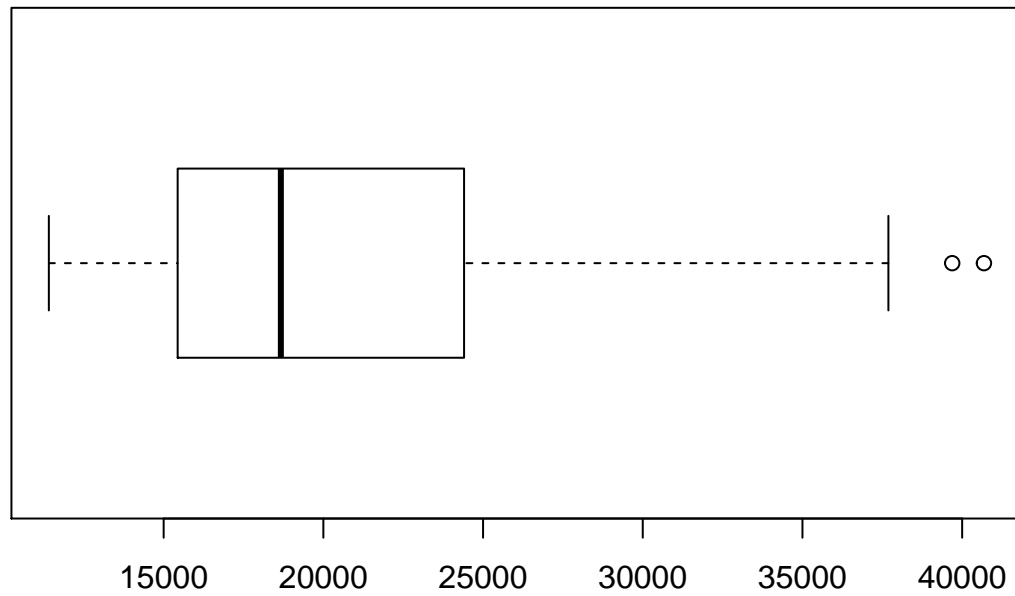
```
boxplot(x.data[4], names, horizontal=TRUE) # rte = rate
```



```
boxplot(x.data[5], names, horizontal=TRUE) # size = size
```



```
boxplot(x.data[6], names, horizontal=TRUE) # vol = volume
```



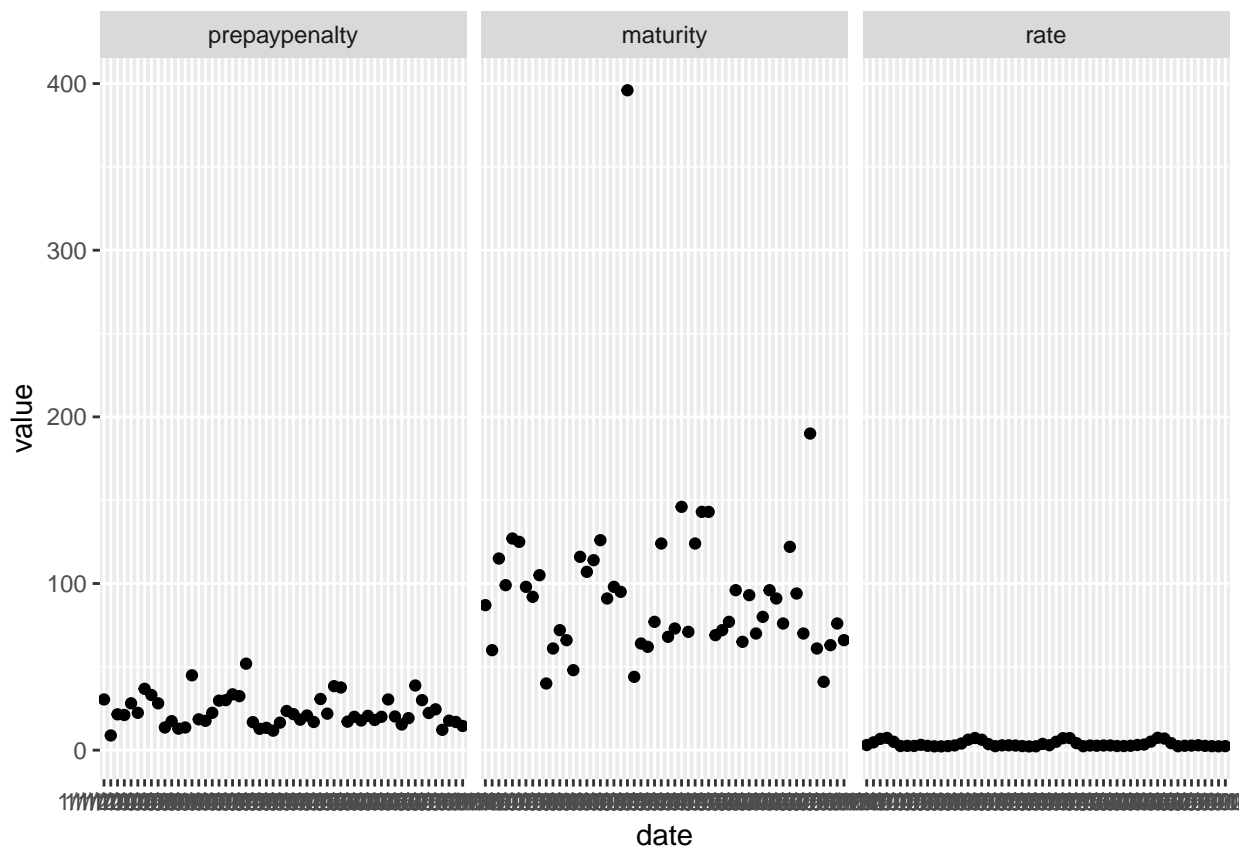
```

#
# Explore the data further.
# Plot the time series data using this code:
# Use melt() from reshape2 to build the data frame with data as id and values of variables
#
require(ggplot2)

## Loading required package: ggplot2
require(reshape2)

## Loading required package: reshape2
x.melted <- melt(x.data[, c(1:4)], id = "date")
ggplot(data = x.melted, aes(x = date, y = value)) + geom_point() + facet_wrap(~variable, scales = "free")

```



```

# Describe the data frame that melt() produces.

```

Data frame produced by `melt()` is from the first 4 columns inclusive of all rows but identifying the date column as the id. The data “melt” is such that each row is a unique id-variable combination, limited to the first 4 columns. The outliers for maturity variable are clearly visible here.

Relational Analysis

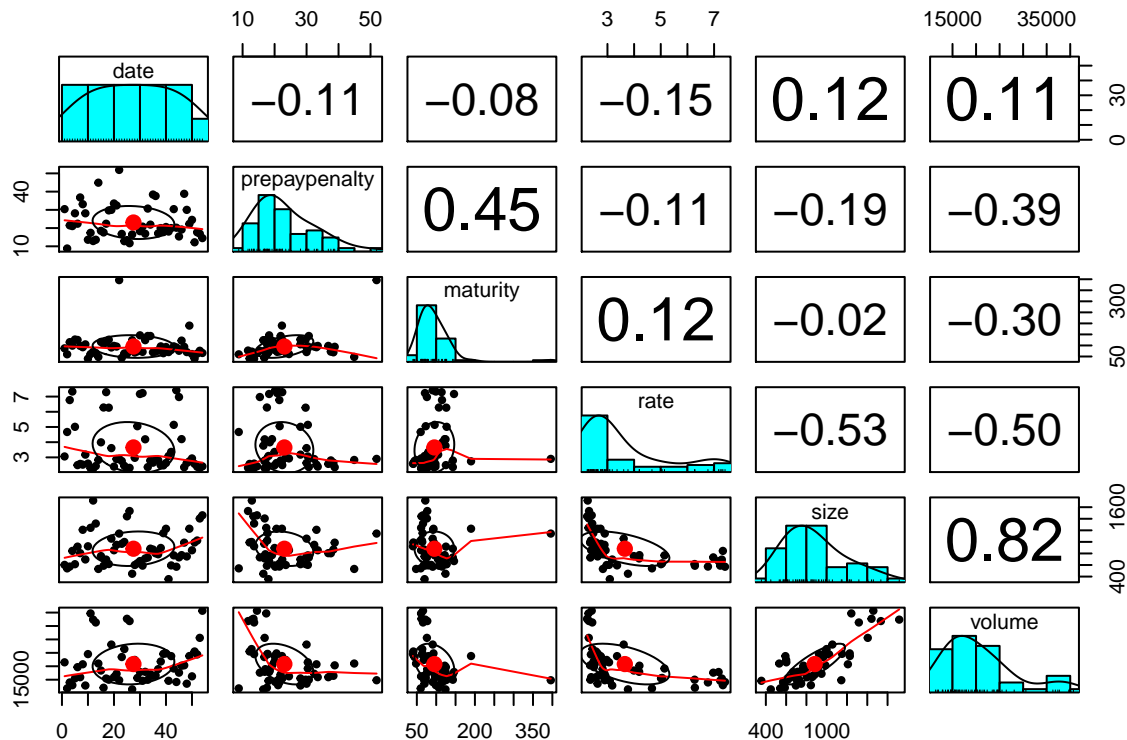
```

# Load the psych library and produce a scatterplot matrix. Interpret this exploration.
require(psych)

```



```
## Loading required package: psych
##
## Attaching package: 'psych'
## The following objects are masked from 'package:ggplot2':
##
##    %+%, alpha
pairs.panels(x.data)
```



Diagonals show histogram and trends for the variables: `ppp`, `mtv`, `rte`, `size` and `vol`. Upper right of diagonal show correlation numbers between pairs of the variables. Lower left of diagonal are scatter plots (partial correlations) between pairs of variables.

Regression Model

```
# Analyze the data.
# Regress rate on the rest of the variables in x.data.
# To do form a matrix of independent variables (predictor or explanatory variables)
# in the matrix X and a separate vector y for the dependent (response) variable rate.
# The 1 vector will produce a constant intercept in the regression model.
y <- as.vector(x.data[, "rate"])
X <- as.matrix(cbind(1, x.data[, c("prepaypenalty", "maturity", "size", "volume")]))
head(y)
```

```
## [1] 3.77 3.09 2.83 3.06 2.97 3.36
```

```
head(X)
```

```
##      1 prepaypenalty maturity size volume
## 1 1      16.5      124 449 11406
## 2 1      18.1      70 356 14586
## 3 1      44.9      48 532 21022
## 4 1      30.4      87 602 21472
## 5 1      23.5      68 600 22359
## 6 1      20.0      80 593 23780
```

```
#
# Explain the code used to form y and X.
#
```

y is a single vector transposed from the rate values in the rows. X is a matrix produced by stacking specific columns (ppp, mty, rte, sze and vol) from x.data side-by-side along with a first column filled with the value 1 in every row. This feeds the predictive model for lending rate, 1 being the multiplier for the intercept.

```
# Calculate the  $\hat{\beta}$  coefficients and interpret their meaning.
require(reshape2)
require(ggplot2)
XTX.inverse <- solve(t(X) %*% X)
(beta.hat <- XTX.inverse %*% t(X) %*% y)
```

```
##              [,1]
## 1              7.771438e+00
## prepaypenalty -6.968996e-02
## maturity      6.399952e-03
## size          -2.041351e-03
## volume        -6.347851e-05
```

The function solve calculates the inverse, giving the coefficients that may be used to weight the various terms and conditions in the regression model for the lending rate.

```
# Calculate actual and predicted rates and plot using this code.
actual <- y
predicted <- X %*% beta.hat
residual <- actual - predicted
results <- data.frame(actual = actual, predicted = predicted, residual = residual)
```

The vector y represents the actual lending rates based on the rows of values in the column rate. The predicted value is based on a best fitting regression line computed using the beta coefficients.

y-predicted = coeff1 (or, intercept) + coeff2 * ppp + coeff3 * mte + coeff4 * sze + coeff5 * vol

```
min_xy <- min(min(results$actual), min(results$predicted))
max_xy <- max(max(results$actual), max(results$predicted))
```

The above compares the min and max of the actual and the predicted values for each term.

```
plot.melt <- melt(results, id.vars = "predicted")
```

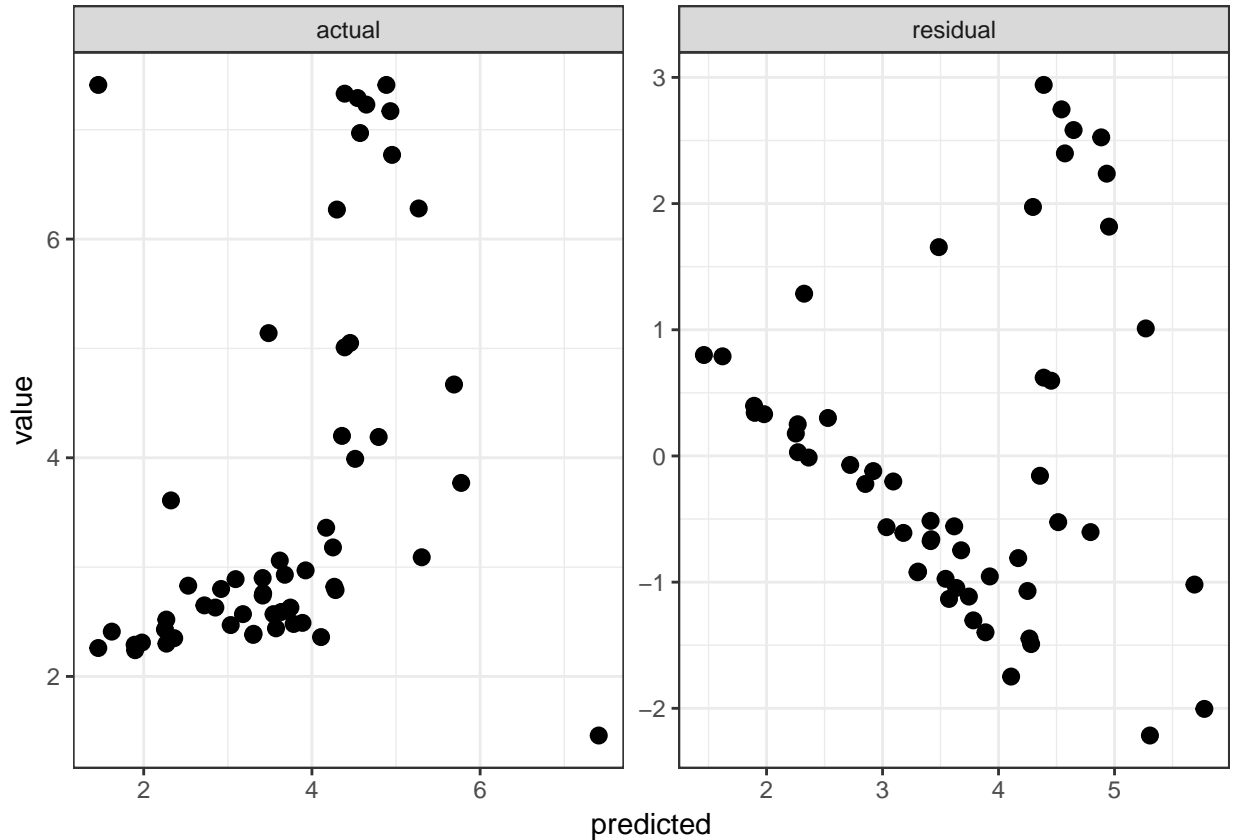
Here, melt is used to produce a data frame for the min and max, using predicted for id.

```
plot.data <- rbind(plot.melt, data.frame(predicted = c(min_xy, max_xy), variable = c("actual", "actual"))
```

The resulting data is stored in plot.data above.

```
p <- ggplot(plot.data, aes(x = predicted, y = value)) + geom_point(size = 2.5) + theme_bw()
p <- p + facet_wrap(~variable, scales = "free")
```

p



The above is the resulting plot for actual vs residual across predicted. The residuals are not completely random leading to a lack of confidence in a linear regression prediction model. The graph for actual against predicted value is clearly non-linear.

Calculate the standard error of the residuals.

```
e <- y - X %*% beta.hat
(e.sse <- t(e) %*% e)
```

```
##           [,1]
```

```
## [1,] 88.62341
```

```
(n <- dim(X) [1])
```

```
## [1] 54
```

```
(k <- nrow(beta.hat))
```

```
## [1] 5
```

```
(e.se <- (e.sse / (n - k))^0.5)
```

```
##           [,1]
```

```
## [1,] 1.344857
```

The above indicate the standard error (e.se) on the loan rates based on (n) observations and (k-1) degrees of freedom.

Observations and Recommendations

Insights

In the current model, the variables prepaypenalty, size and volume impact the lending rate negatively, and the variable maturity impacts the lending rate positively. The scatter plots for the data are skewed to the right. There is moderate positive correlation between prepaypenalty & maturity, and high correlation between size & volume. There is also moderate negative correlation between prepaypenalty & volume. There are outliers in the data. The prediction model cannot be relied upon.

Recommendations

Per the current model, loans with larger maturity are recommended for better lending rates. However, the F-statistic needs to be checked to have any confidence in the [linear] regression model. For a better prediction model, the p-values in the regression should be studied. For each of the measures, the p-value should be less than 0.05 (95% confidence). Also, the outliers should be examined and dropped for a more reliable analysis. Bonferroni outlier test may be applied to confirm this. Finally, to avoid multicollinearity, some of the variable should be dropped or combined. A revised prediction model is recommended.

Sources

Google, CRAN for ggplot2, reshape2, FRED, discussions with others, and varied sources.