# Regression Method/CI Analysis

*Kayleigh Ryherd*

*10/26/2016*

## 15%/65% Confidence Intervals

Read in packages & data.

```
library(fBasics)
```

```
## Loading required package: timeDate

## Loading required package: timeSeries

##

## Rmetrics Package fBasics

## Analysing Markets and calculating Basic Statistics

## Copyright (C) 2005-2014 Rmetrics Association Zurich

## Educational Software for Financial Engineering and Computational Science

## Rmetrics is free software and comes with ABSOLUTELY NO WARRANTY.

## https://www.rmetrics.org --- Mail to: info@rmetrics.org
```

```
library(caret)
```

```
## Loading required package: lattice

## Loading required package: ggplot2
```

```
library(MASS)
library(ggplot2)
library(plyr)
library(reshape2)
setwd("~/Dropbox/Definition_of_PCs/Sept2016")
data <- read.csv("FullExport_NoOverlap_9-29-16.csv")
comp <- read.csv("decoding_composite.csv")
comp <- comp[,-1]
data <- merge(data, comp, by = "SubjectID")
```

Define some functions.

```r
# for plotting
gg_color_hue <- function(n) {
  hues = seq(15, 375, length=n+1)
  hcl(h=hues, l=65, c=100)[1:n]
}

# CIs to z-scores
ci2z <- function(ci)
{
  qnorm(ci + (1 - ci)/2)
}
```

Choose your CIs.

```r
ci.to.use <- c(.15, .65)
ci.title <- "CIs: 15%, 65%"
```

### Robust Model without Vocabulary

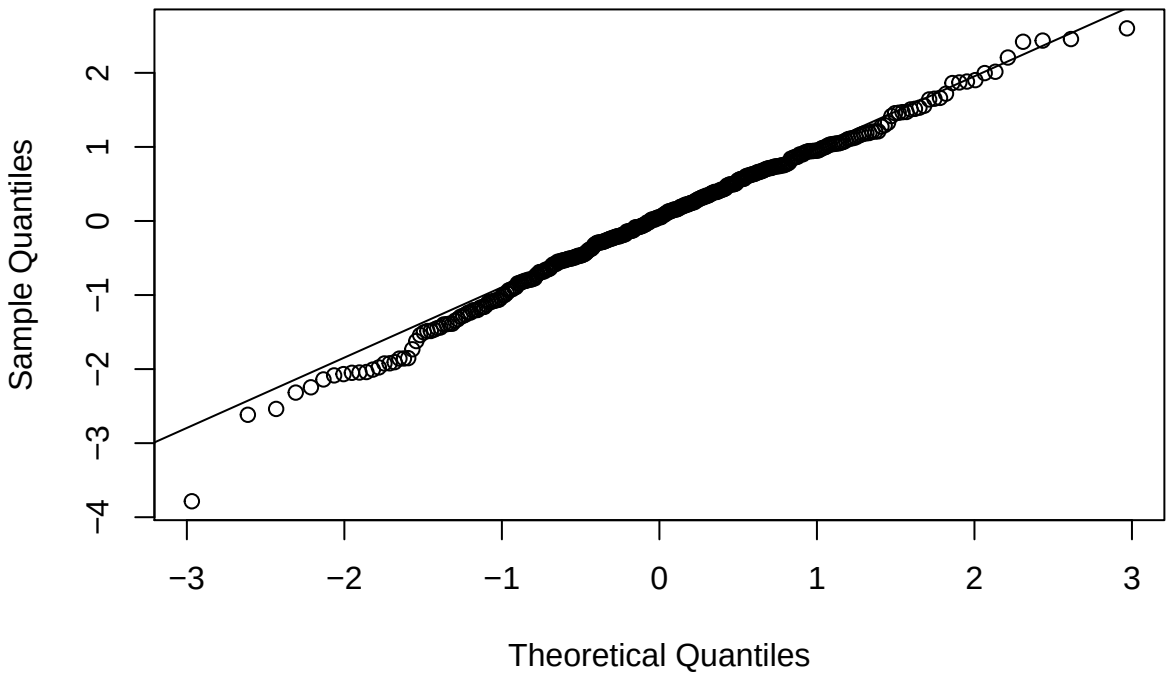
Run model, get groups.

```r
model.data.nv <- data[, c(1, 2, 7, 11, 18)]
nv.cc <- model.data.nv[complete.cases(model.data.nv),]
#just centering
nv.cc$age.c <- nv.cc$age - mean(nv.cc$age, na.rm = TRUE)
#centering and scaling
nv.cc$wj3.rcomp.cs <- scale(nv.cc$wj3.rcomp.raw)
nv.cc$wasi.matr.cs <- scale(nv.cc$wasi.matr.raw)

### Run regression
m1 <- rlm(wj3.rcomp.cs ~ age.c + composite1 + wasi.matr.cs, data = nv.cc, method = "MM")

#standardize predicted values
nv.cc$Pred <- predict(m1)
nv.cc$stdPred <- scale(nv.cc$Pred)

#obtained residuals
nv.cc$resid <- resid <- residuals(m1)
qqnorm(scale(resid)); qqline(scale(resid))
```




```
#create the range of SD (z value) bins
my.bin <- c(range(scale(resid))+.1, ci2z(ci.to.use), -ci2z(ci.to.use))
#we could also manually specify in terms of SDs
# my.bin <- c(-3, -1, -.25, .25, 1, 3)
my.bin <- sort(my.bin)
# names for each bin specified above
grouping <- c("UPC", "NSC", "EAC", "NSC", "UGC")

nv.cc$CI_group <- grouping[findInterval(scale(resid), vec=my.bin, all.inside=T)]
#those with Standardized Predicted Value below -1 should be "EPC"
nv.cc$CI_group[nv.cc$stdPred < -1] <- "EPC"

### there are five groups c("EAC", "NSC", "UGC", "UPC", "EPC")
### so the argument n = 5
my.col <- gg_color_hue(n=5)

## unscale the z value for each CI, so I can add that value to intercept
unscale.ci <- my.bin[-c(1,length(my.bin))]*sd(resid)+mean(resid)

## create plot
m2 <- lm(wj3.rcomp.cs ~ stdPred, data=nv.cc)

p1 <- ggplot(nv.cc, aes(x=stdPred, y=wj3.rcomp.cs)) + geom_point(aes(color=CI_group)) +
  geom_abline(intercept = coef(m2)[1], slope = coef(m2)[2]) +
  geom_abline(intercept = coef(m2)[1] + unscale.ci[2], slope = coef(m2)[2], color=my.col[1]) +
  geom_abline(intercept = coef(m2)[1] + unscale.ci[3], slope = coef(m2)[2], color=my.col[1]) +
  geom_abline(intercept = coef(m2)[1] + unscale.ci[1], slope = coef(m2)[2], color=my.col[4]) +
  geom_abline(intercept = coef(m2)[1] + unscale.ci[4], slope = coef(m2)[2], color=my.col[3]) +
  geom_vline(xintercept = -1, lty="dotted") +
```

```
  xlab("Standardized Predicted Value")+ylab("WJ3 Reading Comprehension") +
  ggtitle(ci.title)
```

Check for group differences.

```
# take away NSC group
nv.cc2 <- subset(nv.cc, CI_group != "NSC")
nv.cc2$CI_group <- as.factor(nv.cc2$CI_group)
# reading comp
a1 <- aov(wj3.rcomp.cs ~ CI_group, data = nv.cc2)
summary(a1)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## CI_group      3 171.37   57.12   145.6 <2e-16 ***
## Residuals   182  71.43    0.39
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
pairwise.t.test(nv.cc2$wj3.rcomp.cs, nv.cc2$CI_group, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  nv.cc2$wj3.rcomp.cs and nv.cc2$CI_group
##
##     EAC     EPC     UGC
## EPC < 2e-16 -       -
## UGC 3.9e-10 < 2e-16 -
## UPC < 2e-16 1       < 2e-16
##
## P value adjustment method: bonferroni
```

```
aggregate(nv.cc2$wj3.rcomp.cs, list(nv.cc2$CI_group), mean)
```

```
##   Group.1         V1
## 1     EAC  0.2862794
## 2     EPC -1.0076870
## 3     UGC  1.2302551
## 4     UPC -1.1031457
```

```
# age
a1 <- aov(age.c ~ CI_group, data = nv.cc2)
summary(a1)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## CI_group      3   74.3  24.759    3.16  0.026 *
## Residuals   182 1426.1   7.836
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
pairwise.t.test(nv.cc2$age.c, nv.cc2$CI_group, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  nv.cc2$age.c and nv.cc2$CI_group
##
##     EAC   EPC   UGC
## EPC 0.047 -     -
## UGC 1.000 0.125 -
## UPC 1.000 0.156 1.000
##
## P value adjustment method: bonferroni
```

```r
aggregate(nv.cc2$age.c, list(nv.cc2$CI_group), mean)
```

```
##   Group.1           x
## 1     EAC  0.30738238
## 2     EPC -1.24013085
## 3     UGC  0.09361161
## 4     UPC  0.01332463
```

```r
# matrices
a1 <- aov(wasi.matr.cs ~ CI_group, data = nv.cc2)
summary(a1)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## CI_group      3  104.4   34.80   50.73 <2e-16 ***
## Residuals   182  124.8    0.69
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
pairwise.t.test(nv.cc2$wasi.matr.cs, nv.cc2$CI_group, p.adjust.method = "bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  nv.cc2$wasi.matr.cs and nv.cc2$CI_group
##
##     EAC     EPC     UGC
## EPC < 2e-16 -       -
## UGC 1.000   < 2e-16 -
## UPC 0.063   2.6e-13 0.311
##
## P value adjustment method: bonferroni
```

```r
aggregate(nv.cc2$wasi.matr.cs, list(nv.cc2$CI_group), mean)
```

```
##   Group.1         V1
## 1     EAC  0.4067069
```

```
## 2     EPC -1.4020053
## 3     UGC  0.2940669
## 4     UPC -0.0480646
```

```
# decoding
a1 <- aov(composite1 ~ CI_group, data = nv.cc2)
summary(a1)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## CI_group      3  101.8   33.92   58.14 <2e-16 ***
## Residuals   182  106.2    0.58
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
pairwise.t.test(nv.cc2$composite1, nv.cc2$CI_group, p.adjust.method = "bonferroni")
```

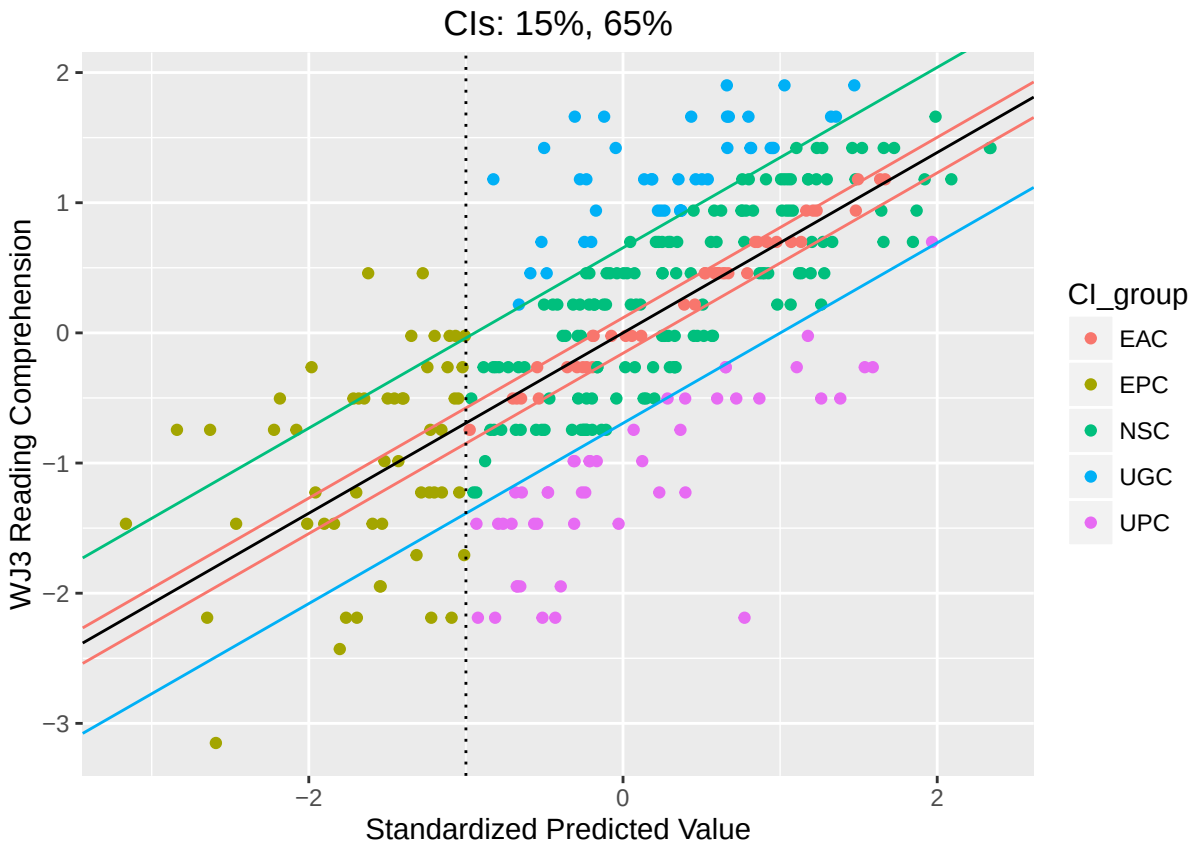
```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  nv.cc2$composite1 and nv.cc2$CI_group
##
##     EAC    EPC    UGC
## EPC <2e-16 -      -
## UGC 1      <2e-16 -
## UPC 1      <2e-16 1
##
## P value adjustment method: bonferroni
```

```
aggregate(nv.cc2$composite1, list(nv.cc2$CI_group), mean)
```

```
##   Group.1          x
## 1     EAC  0.3285975
## 2     EPC -1.3883516
## 3     UGC  0.2658353
## 4     UPC  0.1167800
```

Show group sizes, plot.

```
print(p1)
```




```
Robust.Without.Vocab <- data.frame(table(nv.cc$CI_group))
names(Robust.Without.Vocab) <- c("Group", "rlm.w/o.vocab")
Robust.Without.Vocab
```

```
##   Group rlm.w/o.vocab
## 1   EAC            42
## 2   EPC            54
## 3   NSC           148
## 4   UGC            43
## 5   UPC            47
```