Logistic Regression II

Mike Strube

November 15, 2018

1 Preliminaries

In this section, the RStudio workspace and console panes are cleared of old output, variables, and other miscellaneous debris. Packages are loaded and any required data files are retrieved.

```
options(replace.assign = TRUE, width = 65, digits = 4, scipen = 4, fig.width = 4,
    fig.height = 4)
# Clear the workspace and console.
rm(list = ls(all = TRUE))
cat("\f")
```

```
# Turn off showing of significance asterisks.
options(show.signif.stars = F)
# Set the contrast option; important for ANOVAs.
options(contrasts = c("contr.sum", "contr.poly"))
how_long <- Sys.time()
set.seed(123)
library(knitr)
```

```
library(psych)

## Warning:  package 'psych' was built under R version 3.5.1

library(MASS)
library(sciplot)
library(plyr)
library(aod)
library(MVN)

## sROC 0.1-2 loaded

library(boot)

##
## Attaching package:  'boot'
## The following object is masked from 'package:psych':
##
##     logit

library(car)
```

```
## Warning: package 'car' was built under R version 3.5.1
## Loading required package: carData
##
## Attaching package: 'car'
## The following object is masked from 'package:boot':
##
##     logit
## The following object is masked from 'package:psych':
##
##     logit
```

```r
library(LogisticDx)
library(ROCR)
```

```
## Warning: package 'ROCR' was built under R version 3.5.1
## Loading required package: gplots
## Warning: package 'gplots' was built under R version 3.5.1
##
## Attaching package: 'gplots'
## The following object is masked from 'package:stats':
##
##     lowess
```

```r
library(nnet)
library(mnlogit)
```

```
## Warning: package 'mnlogit' was built under R version 3.5.1
## Package: mnlogit
## Version: 1.2.5
## Multinomial Logit Choice Models.
## Scientific Computing Group, Sentrana Inc.
```

```r
library(VGAM)
```

```
## Loading required package: stats4
## Loading required package: splines
##
## Attaching package: 'VGAM'
## The following object is masked from 'package:mnlogit':
##
##     lrtest
## The following object is masked from 'package:car':
##
##     logit
## The following objects are masked from 'package:boot':
##
##     logit, simplex
## The following objects are masked from 'package:psych':
##
##     fisherz, logistic, logit
```

```r
library(rms)
```

```
## Loading required package: Hmisc
## Loading required package: lattice
##
## Attaching package: 'lattice'
```

```
## The following object is masked from 'package:boot':
##
##     melanoma
## Loading required package:  survival
##
## Attaching package:  'survival'
## The following object is masked from 'package:boot':
##
##     aml
## The following object is masked from 'package:aod':
##
##     rats
## Loading required package:  Formula
## Loading required package:  ggplot2
## Warning:  package 'ggplot2' was built under R version 3.5.1
##
## Attaching package:  'ggplot2'
## The following objects are masked from 'package:psych':
##
##     %+%, alpha
##
## Attaching package:  'Hmisc'
## The following objects are masked from 'package:plyr':
##
##     is.discrete, summarize
## The following object is masked from 'package:psych':
##
##     describe
## The following objects are masked from 'package:base':
##
##     format.pval, units
## Loading required package:  SparseM
##
## Attaching package:  'SparseM'
## The following object is masked from 'package:base':
##
##     backsolve
##
## Attaching package:  'rms'
## The following objects are masked from 'package:VGAM':
##
##     calibrate, lrtest
## The following object is masked from 'package:mnlogit':
##
##     lrtest
## The following objects are masked from 'package:car':
##
##     Predict, vif
```

```
library(ordinal)
```

```
## Warning:  package 'ordinal' was built under R version 3.5.1
##
## Attaching package:  'ordinal'
```

```
## The following objects are masked from 'package:VGAM':
##
##     dgumbel, dlgamma, pgumbel, plgamma, qgumbel, rgumbel,
##     wine
## The following object is masked from 'package:psych':
##
##     income

library(qqplotr)

##
## Attaching package:  'qqplotr'
## The following objects are masked from 'package:ggplot2':
##
##     stat_qq_line, StatQqLine

library(gridExtra)
library(caret)

## Warning:  package 'caret' was built under R version 3.5.1
##
## Attaching package:  'caret'
## The following object is masked from 'package:survival':
##
##     cluster
## The following object is masked from 'package:VGAM':
##
##     predictors

library(GGally)
library(mlogit)

## Warning:  package 'mlogit' was built under R version 3.5.1
## Loading required package:  maxLik
## Loading required package:  miscTools
##
## Please cite the 'maxLik' package as:
## Henningsen, Arne and Toomet, Ott (2011).  maxLik:  A package for maximum likelihood estimation
in R. Computational Statistics 26(3), 443-458.  DOI 10.1007/s00180-010-0217-1.
##
## If you have questions, suggestions, or comments regarding the 'maxLik' package, please use
a forum or 'tracker' at maxLik's R-Forge site:
## https://r-forge.r-project.org/projects/maxlik/
##
## Attaching package:  'mlogit'
## The following object is masked from 'package:rms':
##
##     lrtest
## The following object is masked from 'package:VGAM':
##
##     lrtest
## The following objects are masked from 'package:mnlogit':
##
##     hmftest, index, scoretest

library(multcomp)
```

```
## Loading required package:  mvtnorm
## Loading required package:  TH.data
##
## Attaching package:  'TH.data'
## The following object is masked from 'package:MASS':
##
##    geyser
```

```
library(ggplot2)
```

1.1 Data

```
setwd("C:\\Courses\\Psychology 516\\PowerPoint\\2018")

Job <- read.table("jobs_example_for_ppt.csv", sep = ",", header = TRUE)
Job <- as.data.frame(Job)
Job <- Job[sample(1:nrow(Job)), ]
```

1.2 Data Modifications

Depending on the type of analysis, the outcome needs to be in a particular form. Residualized versions of continuous predictors are created so that preliminary analyses are not contaminated by outcome differences.

```
Job$job_result[Job$job == "0"] <- "No Job"
Job$job_result[Job$job == "1"] <- "Job"

Job$outcome_result[Job$outcome == 1] <- "No Interview"
Job$outcome_result[Job$outcome == 2] <- "Job"
Job$outcome_result[Job$outcome == 3] <- "Interview Only"

Job$ordered_result[Job$ordered == 1] <- "Not Interviewed"
Job$ordered_result[Job$ordered == 2] <- "Interviewed Only"
Job$ordered_result[Job$ordered == 3] <- "Hired"

# Dummy code for sex.
Job$sex_D <- ifelse(Job$sex == 2, 1, 0)

# Dummy codes for men and women
Job$M_D <- ifelse(Job$sex == 1, 1, 0)
Job$F_D <- ifelse(Job$sex == 2, 1, 0)

# Centered predictors.
Job$gre_c <- as.numeric(scale(Job$gre, scale = FALSE))
Job$pubs_c <- as.numeric(scale(Job$pubs, scale = FALSE))
Job$years_c <- as.numeric(scale(Job$years, scale = FALSE))

# Residuals, now based on the three-category outcome.
Job$gre_R <- lm(gre ~ as.factor(outcome), data = Job)$residuals
Job$pubs_R <- lm(pubs ~ as.factor(outcome), data = Job)$residuals
Job$years_R <- lm(years ~ as.factor(outcome), data = Job)$residuals
```

```r
# Some factor versions.
Job$sex_F <- factor(Job$sex, levels = c(1, 2), labels = c("Men", "Women"))
Job$ordered_F <- factor(Job$ordered, levels = c(1, 2, 3), labels = c("Not Interviewed",
    "Interviewed Only", "Hired"))
```

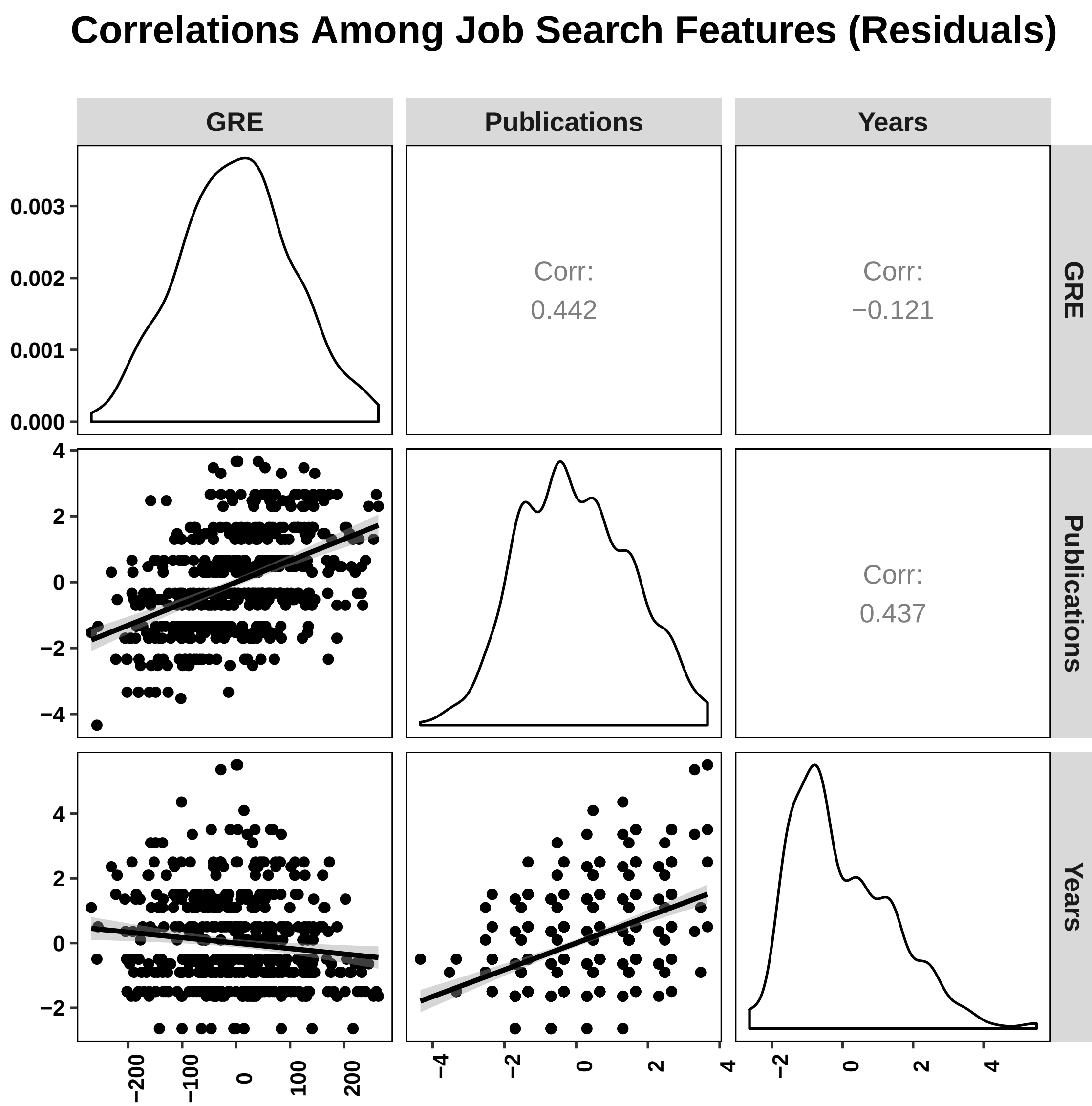
2 Job Search Data Characteristics

These hypothetical data simulate the factors that might contribute to successfully getting an academic job. The basic nature of these data is explored here.

3 Basic Visualization

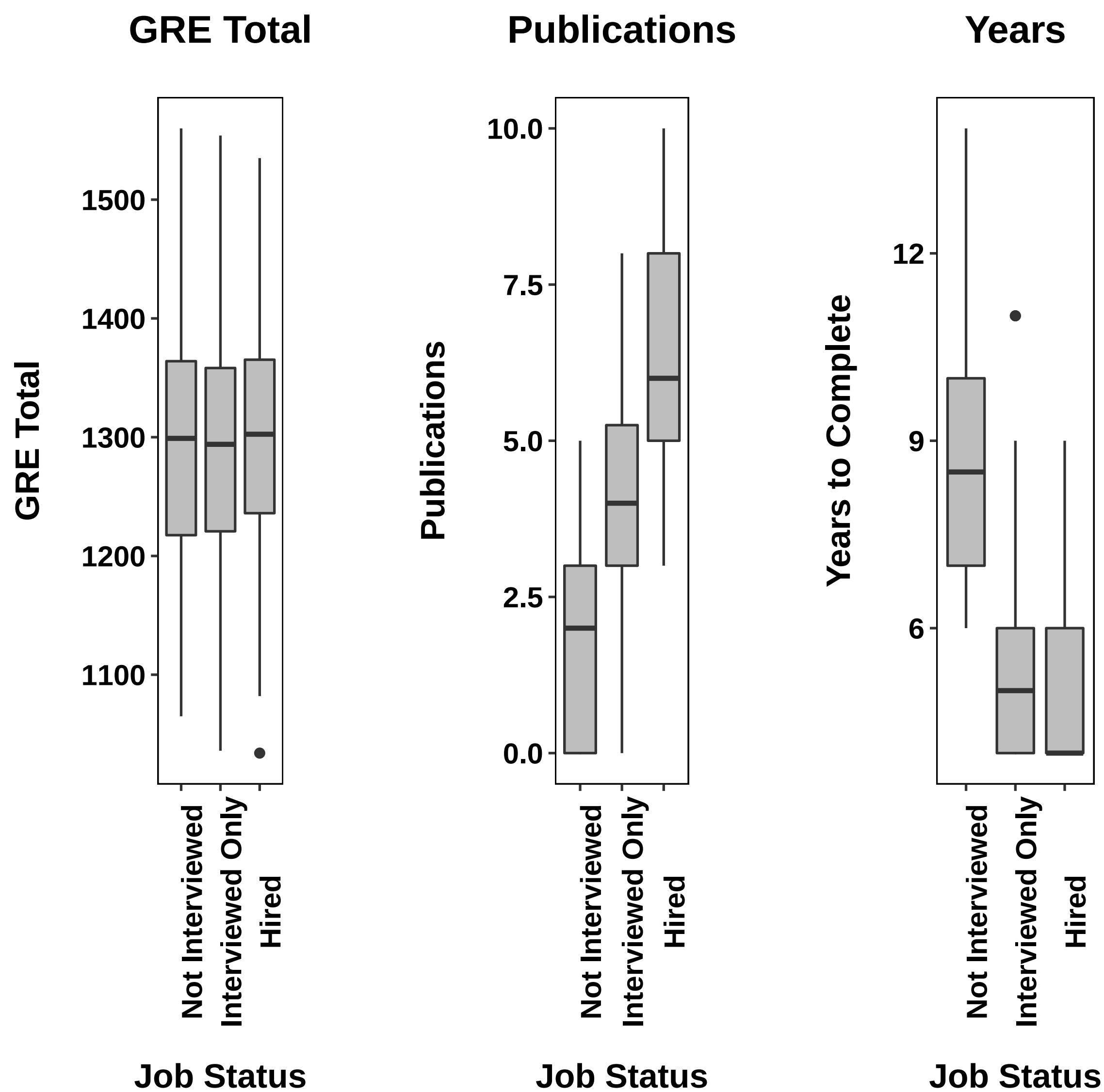
The basic nature of the data is easily viewed with some simple graphics.

```
ggpairs(Job[18:20], lower = list(continuous = "smooth"), upper = list(continuous = "cor"),
    columnLabels = c("GRE", "Publications", "Years")) + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 9, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 9, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 16), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 16), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Correlations Among Job Search Features (Residuals)")
```



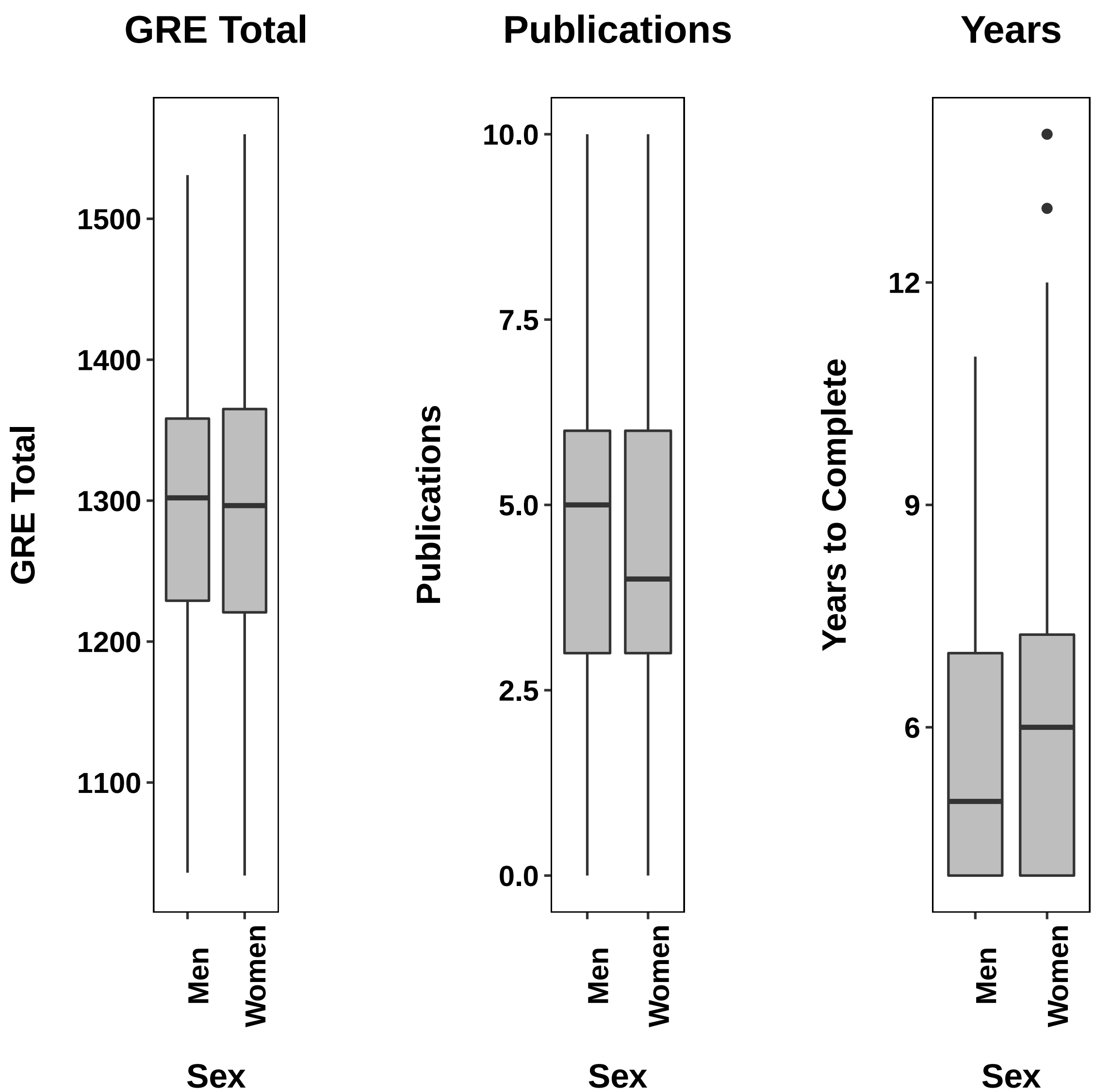

```
Job$ordered_result = factor(Job$ordered_result, levels(factor(Job$ordered_result))[c(3,
    2, 1)])

p1 <- ggplot(Job, aes(x = ordered_result, y = gre)) + geom_boxplot(fill = "gray") +
    ylab("GRE Total") + xlab("Job Status") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
```

```
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("GRE Total")

p2 <- ggplot(Job, aes(x = ordered_result, y = pubs)) + geom_boxplot(fill = "gray") +
    ylab("Publications") + xlab("Job Status") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Publications")

p3 <- ggplot(Job, aes(x = ordered_result, y = years)) + geom_boxplot(fill = "gray") +
    ylab("Years to Complete") + xlab("Job Status") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Years")
grid.arrange(p1, p2, p3, nrow = 1)
```




```
Job$sex_F <- factor(Job$sex, levels = c(1, 2), labels = c("Men", "Women"))
p1 <- ggplot(Job, aes(x = sex_F, y = gre)) + geom_boxplot(fill = "gray") +
    ylab("GRE Total") + xlab("Sex") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("GRE Total")
```

```r
p2 <- ggplot(Job, aes(x = sex_F, y = pubs)) + geom_boxplot(fill = "gray") +
    ylab("Publications") + xlab("Sex") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Publications")

p3 <- ggplot(Job, aes(x = sex_F, y = years)) + geom_boxplot(fill = "gray") +
    ylab("Years to Complete") + xlab("Sex") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Years")
grid.arrange(p1, p2, p3, nrow = 1)
```



3.1 Group Differences

A univariate look at the data will provide some clues about likely variables of influence in the logistic regression.

```
Job_MANOVA_1 <- manova(as.matrix(Job[, 3:5]) ~ Job$job)
summary(Job_MANOVA_1)

##              Df Pillai approx F num Df den Df Pr(>F)
## Job$job       1   0.41      115      3    496 <2e-16
## Residuals 498

summary.aov(Job_MANOVA_1)
```

```
##  Response gre :
##               Df  Sum Sq Mean Sq F value Pr(>F)
## Job$job       1    5693    5693    0.53   0.47
## Residuals   498 5362834   10769
##
##  Response pubs :
##               Df Sum Sq Mean Sq F value Pr(>F)
## Job$job       1    927     927     266 <2e-16
## Residuals   498   1733       3
##
##  Response years :
##               Df Sum Sq Mean Sq F value Pr(>F)
## Job$job       1    263   262.6    70.9  4e-16
## Residuals   498   1844     3.7
```

```
table_1 <- table(Job[c("ordered_F", "sex_F")])
table_1
```

```
##                     sex_F
## ordered_F            Men Women
##   Not Interviewed    34    86
##   Interviewed Only  92   152
##   Hired              62    74
```

```
p_table_1 <- prop.table(table(Job[c("ordered_F", "sex_F")]), 2)
p_table_1
```

```
##                     sex_F
## ordered_F               Men  Women
##   Not Interviewed   0.1809 0.2756
##   Interviewed Only 0.4894 0.4872
##   Hired             0.3298 0.2372
```

```
chisq.test(table_1)
```

```
##
##  Pearson's Chi-squared test
##
## data:  table_1
## X-squared = 8.1, df = 2, p-value = 0.02
```

```
Job_MANOVA_2 <- manova(as.matrix(Job[, 3:5]) ~ Job$sex_F)
summary(Job_MANOVA_2)
```

```
##             Df Pillai approx F num Df den Df Pr(>F)
## Job$sex_F    1 0.0159     2.67      3    496  0.047
## Residuals 498
```

```
summary.aov(Job_MANOVA_2)
```

```
##  Response gre :
##               Df  Sum Sq Mean Sq F value Pr(>F)
## Job$sex_F     1    1810    1810    0.17   0.68
## Residuals   498 5366717   10777
##
##  Response pubs :
```

```
##               Df Sum Sq Mean Sq F value Pr(>F)
## Job$sex_F      1     11   11.19     2.1   0.15
## Residuals    498   2648    5.32
##
##  Response years :
##               Df Sum Sq Mean Sq F value Pr(>F)
## Job$sex_F      1     28   27.62    6.61   0.01
## Residuals    498   2079    4.18
```

4 Basic Logistic Regression Model

We will examine basic diagnostics using the following binary logistic regression model.

```
Job_BLR_1 <- glm(Job$job ~ gre_c + pubs_c + years_c + sex_D, family = binomial("logit"),
    data = Job)
summary(Job_BLR_1)

##
## Call:
## glm(formula = Job$job ~ gre_c + pubs_c + years_c + sex_D, family = binomial("logit"),
##     data = Job)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5596  -0.3111  -0.0142   0.0755   2.7257
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.71205    0.46554   -7.97  1.5e-15
## gre_c       -0.01470    0.00231   -6.37  1.8e-10
## pubs_c       1.99614    0.22058    9.05  < 2e-16
## years_c     -1.43390    0.18667   -7.68  1.6e-14
## sex_D       -0.40619    0.35023   -1.16     0.25
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 585.24  on 499  degrees of freedom
## Residual deviance: 225.23  on 495  degrees of freedom
## AIC: 235.2
##
## Number of Fisher Scoring iterations: 8
```

4.1 Diagnostic Indices

Examination of diagnostic indices can help identify modeling problems. Generally we want to detect outliers and unusually influential cases, eliminate severe multicollinearity, and determine if the chosen distribution and link function are appropriate.

There are quite a number of ways to examine model problems, but we will focus on

the same basic diagnostic indices as used in OLS regression:

- *Residuals*
- *Leverages*
- *Cook's distances*
- *DFBETAs*
- *DFFITs*

4.1.1 Residuals

Residuals indicate lack of fit for each case and can be constructed in quite a few ways in logistic regression. One common form is the Pearson residual:

$$r_i = \frac{y_i - \hat{\pi}_i}{\sqrt{\hat{\pi}_i(1-\hat{\pi}_i)}}$$

in which y_i is the observed response (0 or 1) and $\hat{\pi}$ is the predicted probability of a response. Each squared residual is a χ^2 variable with df=1. The sum of the individual χ^2 is also a χ^2 variable:

$$\sum_{i=1}^{N} r_i^2 = \textit{Pearson } \chi^2_{N-p}$$

A second major residual is the deviance residual:

$$d_i = s_i\sqrt{-2[y_i ln\hat{\pi}_i + (1-y_i)ln(1-\hat{\pi}_i)]}$$

$$s_i = 1 \textit{ when } y_i = 1 \textit{ and } s_i = -1 \textit{ when } y_i = 0$$

Summing the squared deviances produces the deviance for the entire model. In other words, the deviance residual has the convenient interpretation that it represents each case's contribution to the model's badness of fit:

$$\textit{Deviance} = \sum_{i=1}^{N} d_i^2$$

A common transformation of these residuals is to standardize them by dividing by the leverage. The leverage is a measure of a case's potential influence on a regression model. It comes from the diagonal of the "hat" matrix and represents how different a case is from the remaining cases in the multivariate space of the predictors. It can range from 0 to 1.

$$r_{si} = \frac{r_i}{\sqrt{1-h_{ii}}}$$

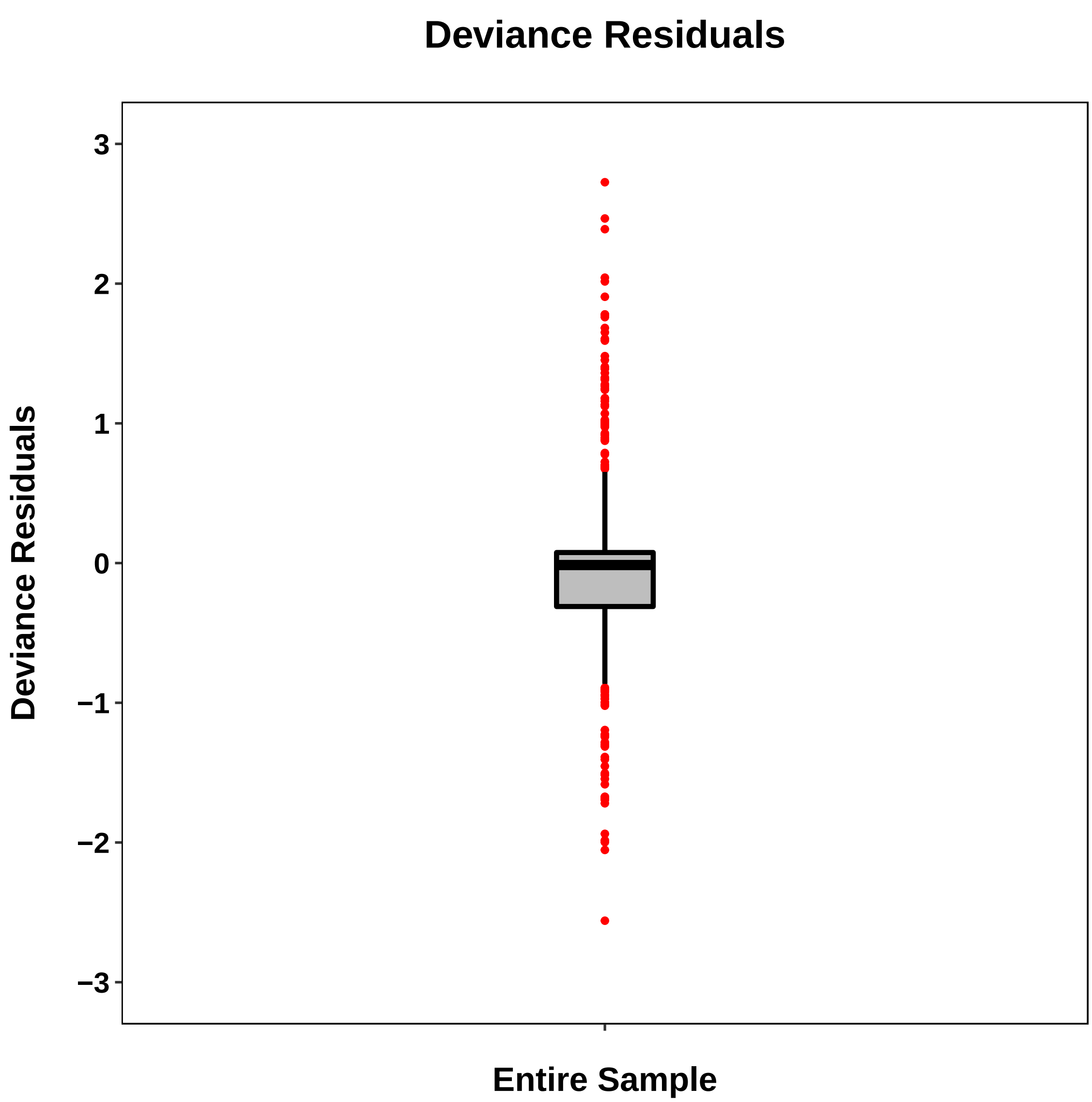
$$d_{si} = \frac{d_i}{\sqrt{1-h_{ii}}}$$

One other common adjustment is to calculate the residuals from a model that excludes the case so that it cannot influence the model on which its own residual is based. These are known as studentized residuals because their distribution follows the t distribution. All of these residuals tend to be very highly related. It is sometimes easier to identify unusual cases with standardized and studentized residuals.

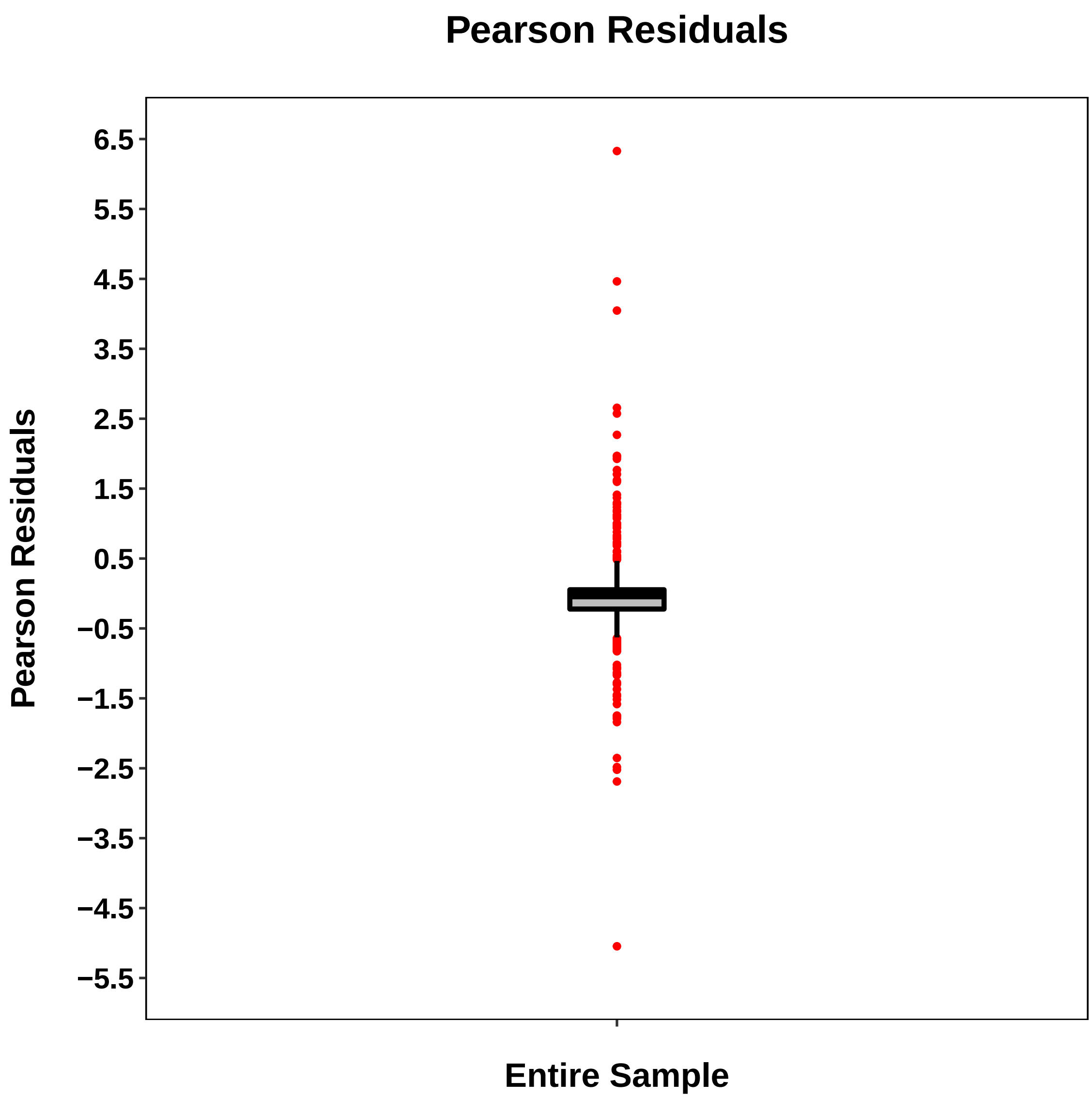
```
Job_BLR_1_R <- cbind(residuals(Job_BLR_1, type = "deviance"), residuals(Job_BLR_1,
    type = "pearson"), residuals(Job_BLR_1, type = "working"), residuals(Job_BLR_1,
    type = "response"), residuals(Job_BLR_1, type = "partial"), rstudent(Job_BLR_1,
    type = "deviance"), rstudent(Job_BLR_1, type = "pearson"), rstandard(Job_BLR_1,
    type = "deviance"), rstandard(Job_BLR_1, type = "pearson"))

Job_BLR_1_R <- as.data.frame(Job_BLR_1_R)

names(Job_BLR_1_R) <- c("deviance", "pearson", "working", "response",
    "partial_sex", "partial_gre", "partial_pubs", "partial_years",
    "stud_deviance", "stud_pearson", "stand_deviance", "stand_pearson")

psych::describe(Job_BLR_1_R)

##                vars   n  mean   sd median trimmed  mad    min
## deviance          1 500 -0.04 0.67  -0.01   -0.07 0.33  -2.56
## pearson           2 500 -0.01 0.75  -0.01   -0.05 0.23  -5.05
## working           3 500 -0.35 2.98  -1.00   -0.58 0.09 -26.46
## response          4 500  0.00 0.27   0.00   -0.01 0.04  -0.96
## partial_sex       5 500 -0.35 3.32  -0.53   -0.49 1.95 -27.03
## partial_gre       6 500 -0.35 5.86  -1.61   -0.58 5.91 -19.08
## partial_pubs      7 500 -0.35 4.39   0.27   -0.27 3.82 -26.34
## partial_years     8 500 -0.35 2.99  -1.15   -0.58 0.56 -26.21
## stud_deviance     9 500 -0.04 0.68  -0.01   -0.07 0.33  -2.61
## stud_pearson     10 500 -0.04 0.68  -0.01   -0.07 0.33  -2.61
## stand_deviance   11 500 -0.04 0.68  -0.01   -0.07 0.33  -2.57
## stand_pearson    12 500 -0.01 0.76  -0.01   -0.05 0.24  -5.07
##                  max range skew kurtosis   se
## deviance        2.73  5.29 0.44     2.47 0.03
## pearson         6.33 11.37 1.39    18.68 0.03
## working        41.05 67.52 5.23    91.47 0.13
## response        0.98  1.94 0.30     3.41 0.01
## partial_sex    40.52 67.54 3.51    56.32 0.15
## partial_gre    40.45 59.53 0.98     4.50 0.26
## partial_pubs   42.61 68.95 1.48    20.17 0.20
## partial_years  40.90 67.11 5.20    89.28 0.13
## stud_deviance   2.76  5.37 0.44     2.52 0.03
## stud_pearson    2.76  5.37 0.44     2.52 0.03
## stand_deviance  2.73  5.30 0.44     2.43 0.03
## stand_pearson   6.34 11.41 1.37    18.31 0.03
```

```
ggplot(Job_BLR_1_R, aes(x = 1, y = Job_BLR_1_R$deviance)) + geom_boxplot(fill = "grey",
    size = 1, color = "black", width = 0.01, outlier.colour = "red",
    outlier.shape = 19, outlier.size = 1) + scale_y_continuous(breaks = c(seq(-3,
    3, 1))) + scale_x_continuous(breaks = c(1)) + coord_cartesian(xlim = c(1,
    1), ylim = c(-3, 3)) + xlab("Entire Sample") + ylab("Deviance Residuals") +
    theme(text = element_text(size = 14, family = "sans", color = "black",
        face = "bold"), axis.text.y = element_text(colour = "black",
```

```
        size = 12, face = "bold"), axis.text.x = element_blank(),
        axis.title.x = element_text(margin = margin(15, 0, 0, 0),
            size = 14), axis.title.y = element_text(margin = margin(0,
            15, 0, 0), size = 14), axis.line.x = element_blank(),
        axis.line.y = element_blank(), plot.title = element_text(size = 16,
            face = "bold", margin = margin(0, 0, 20, 0), hjust = 0.5),
        panel.background = element_rect(fill = "white", linetype = 1,
            color = "black"), panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
        plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
        legend.title = element_blank()) + ggtitle("Deviance Residuals")
```

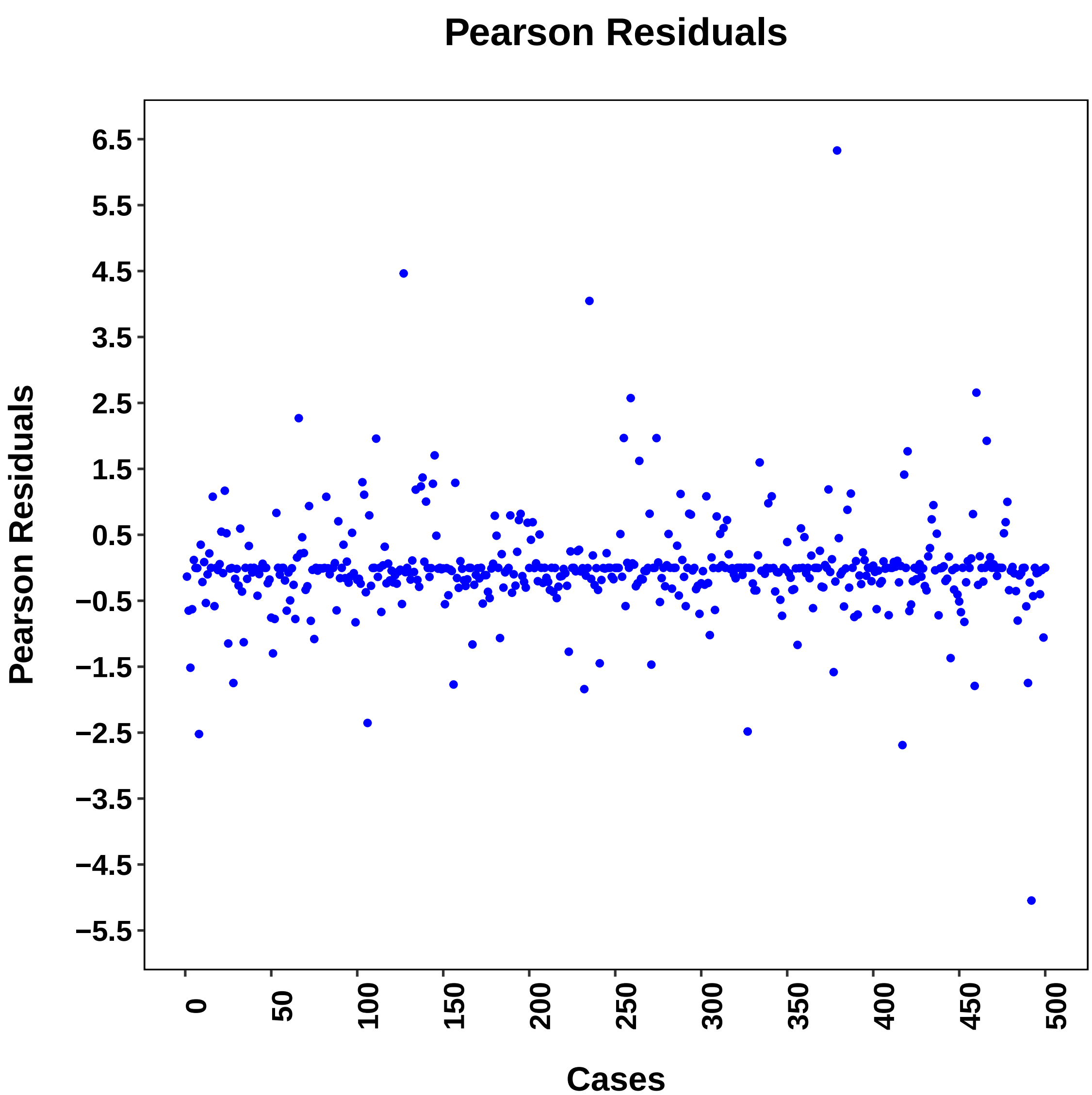


```r
ggplot(Job_BLR_1_R, aes(x = 1, y = Job_BLR_1_R$pearson)) + geom_boxplot(fill = "grey",
    size = 1, color = "black", width = 0.01, outlier.colour = "red",
    outlier.shape = 19, outlier.size = 1) + scale_y_continuous(breaks = c(seq(-5.5,
    6.5, 1))) + scale_x_continuous(breaks = c(1)) + coord_cartesian(xlim = c(1,
    1), ylim = c(-5.5, 6.5)) + xlab("Entire Sample") + ylab("Pearson Residuals") +
    theme(text = element_text(size = 14, family = "sans", color = "black",
        face = "bold"), axis.text.y = element_text(colour = "black",
        size = 12, face = "bold"), axis.text.x = element_blank(),
        axis.title.x = element_text(margin = margin(15, 0, 0, 0),
            size = 14), axis.title.y = element_text(margin = margin(0,
            15, 0, 0), size = 14), axis.line.x = element_blank(),
        axis.line.y = element_blank(), plot.title = element_text(size = 16,
            face = "bold", margin = margin(0, 0, 20, 0), hjust = 0.5),
        panel.background = element_rect(fill = "white", linetype = 1,
            color = "black"), panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
        plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
        legend.title = element_blank()) + ggtitle("Pearson Residuals")
```



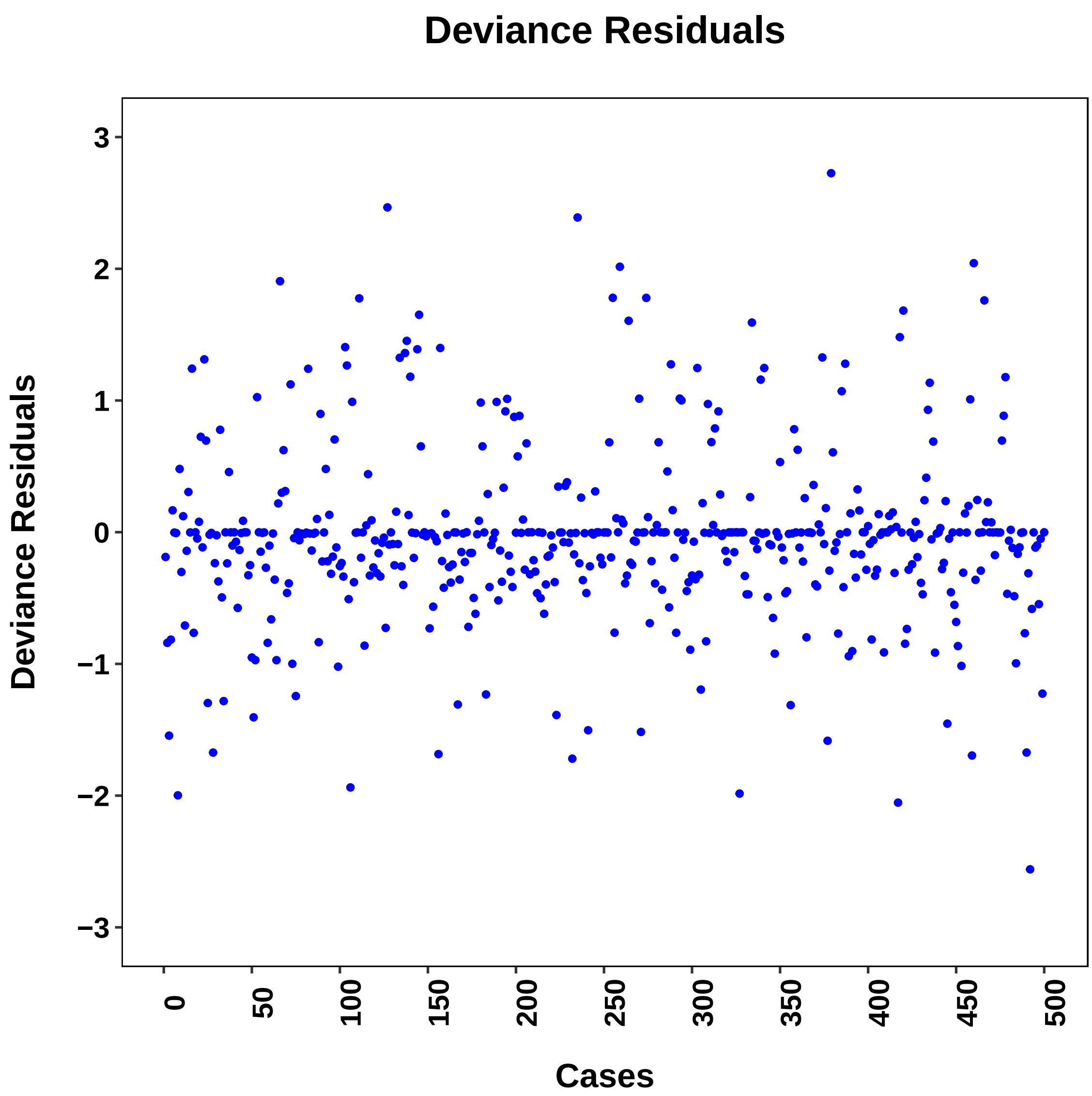

```
Job_BLR_1_R$Case_Num <- seq(1, length(Job_BLR_1_R[, 1]), 1)

ggplot(Job_BLR_1_R, aes(x = Case_Num, y = Job_BLR_1_R$pearson)) +
    geom_point(color = "blue", size = 1) + scale_y_continuous(breaks = c(seq(-5.5,
    6.5, 1))) + scale_x_continuous(breaks = seq(0, length(Job_BLR_1_R[,
    1]), 50)) + coord_cartesian(xlim = c(1, 500), ylim = c(-5.5, 6.5)) +
    xlab("Cases") + ylab("Pearson Residuals") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
```

```
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Pearson Residuals")
```



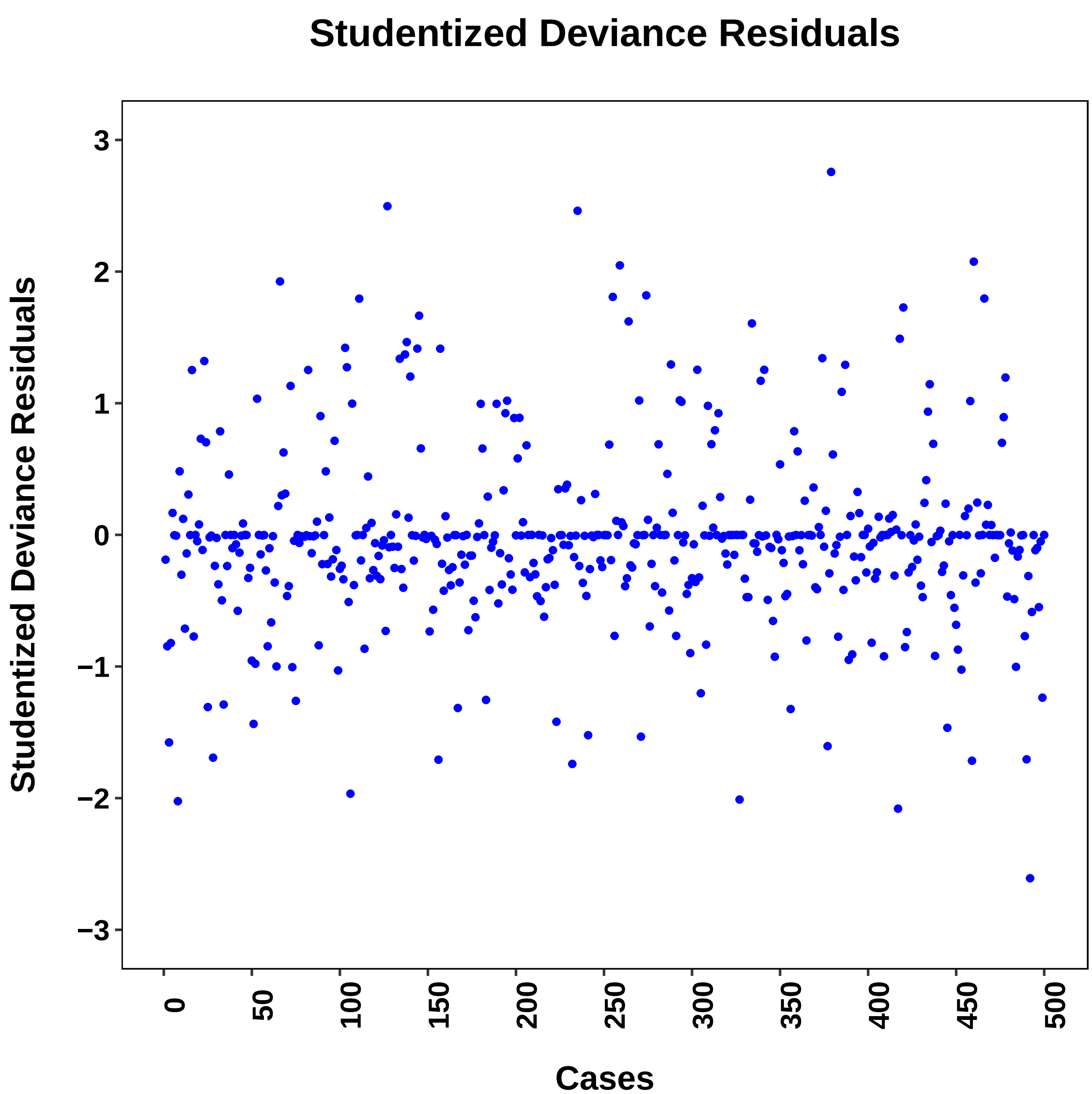

```
ggplot(Job_BLR_1_R, aes(x = Case_Num, y = Job_BLR_1_R$deviance)) +
    geom_point(color = "blue", size = 1) + scale_y_continuous(breaks = c(seq(-3,
    3, 1))) + scale_x_continuous(breaks = seq(0, length(Job_BLR_1_R[,
    1]), 50)) + coord_cartesian(xlim = c(1, 500), ylim = c(-3, 3)) +
    xlab("Cases") + ylab("Deviance Residuals") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
```

```
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Deviance Residuals")
```




```
ggplot(Job_BLR_1_R, aes(x = Case_Num, y = Job_BLR_1_R$stud_deviance)) +
    geom_point(color = "blue", size = 1) + scale_y_continuous(breaks = c(seq(-3,
    3, 1))) + scale_x_continuous(breaks = seq(0, length(Job_BLR_1_R[,
    1]), 50)) + coord_cartesian(xlim = c(1, 500), ylim = c(-3, 3)) +
```

```
xlab("Cases") + ylab("Studentized Deviance Residuals") + theme(text = element_text(size = 14,
family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
plot.title = element_text(size = 16, face = "bold", margin = margin(0,
    0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
    linetype = 1, color = "black"), panel.grid.major = element_blank(),
panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
legend.title = element_blank()) + ggtitle("Studentized Deviance Residuals")
```



```
Extreme_Cases_L <- function(resids, percentile) {
    r_2 <- resids[which(resids <= quantile(resids, probs = ((percentile/100))))]
    c_2 <- which(resids <= quantile(resids, probs = ((percentile/100))))
    results <- matrix(NA, nrow = length(r_1), ncol = 2)
    results[, 1] <- c_2
    results[, 2] <- r_2
    results <- results[order(results[, 2]), ]
    colnames(results) <- c("Case", "Value")
    return(results)
}
```

```
Extreme_Cases_U <- function(resids, percentile) {
    r_1 <- resids[which(resids >= quantile(resids, probs = 1 - ((percentile/100))))]
    c_1 <- which(resids >= quantile(resids, probs = 1 - ((percentile/100))))
    results <- matrix(NA, nrow = length(r_1), ncol = 2)
    results[, 1] <- c_1
    results[, 2] <- r_1
    results <- results[order(results[, 2]), ]
    colnames(results) <- c("Case", "Value")
    return(results)
}
```

```
Extreme_Cases_2 <- function(resids, percentile) {
    r_1 <- resids[which(resids >= quantile(resids, probs = 1 - ((percentile/100)/2)))]
    c_1 <- which(resids >= quantile(resids, probs = 1 - ((percentile/100)/2)))
    r_2 <- resids[which(resids <= quantile(resids, probs = ((percentile/100)/2)))]
    c_2 <- which(resids <= quantile(resids, probs = ((percentile/100)/2)))
    results <- matrix(NA, nrow = 2 * length(r_1), ncol = 2)
    results[, 1] <- c(c_1, c_2)
    results[, 2] <- c(r_1, r_2)
    results <- results[order(results[, 2]), ]
    colnames(results) <- c("Case", "Value")
    return(results)
}
```

```
Extreme_Cases_2(Job_BLR_1_R$deviance, 1)

##      Case  Value
## [1,]  492 -2.560
## [2,]  417 -2.053
## [3,]    8 -1.998
## [4,]  235  2.390
## [5,]  127  2.466
## [6,]  379  2.726

Extreme_Cases_2(Job_BLR_1_R$pearson, 1)

##      Case  Value
## [1,]  492 -5.046
## [2,]  417 -2.689
## [3,]    8 -2.521
## [4,]  235  4.047
```

```
## [5,]  127  4.464
## [6,]  379  6.329
```

```r
Extreme_Cases_2(Job_BLR_1_R$stud_deviance, 1)
```

```
##      Case  Value
## [1,]  492 -2.609
## [2,]  417 -2.080
## [3,]    8 -2.023
## [4,]  235  2.462
## [5,]  127  2.497
## [6,]  379  2.757
```

```r
Extreme_Cases_2(Job_BLR_1_R$stud_pearson, 1)
```

```
##      Case  Value
## [1,]  492 -2.609
## [2,]  417 -2.080
## [3,]    8 -2.023
## [4,]  235  2.462
## [5,]  127  2.497
## [6,]  379  2.757
```

```r
Extreme_Cases_2(Job_BLR_1_R$stand_deviance, 1)
```

```
##      Case  Value
## [1,]  492 -2.572
## [2,]  417 -2.069
## [3,]    8 -2.014
## [4,]  235  2.415
## [5,]  127  2.476
## [6,]  379  2.732
```

```r
Extreme_Cases_2(Job_BLR_1_R$stand_pearson, 1)
```

```
##      Case  Value
## [1,]  492 -5.071
## [2,]  417 -2.709
## [3,]    8 -2.541
## [4,]  235  4.090
## [5,]  127  4.481
## [6,]  379  6.342
```

4.1.2 Leverage

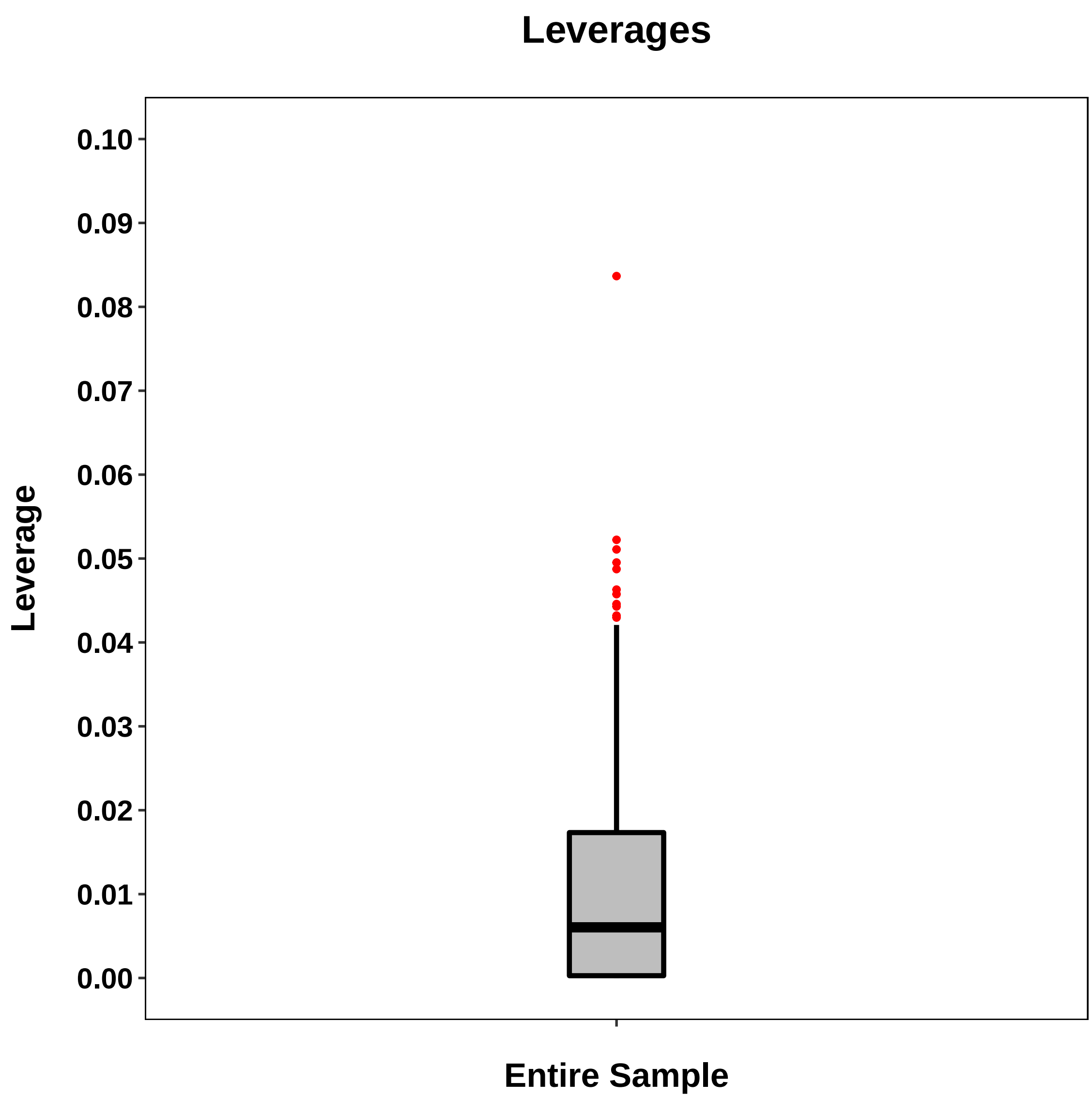
The leverage is a measure of a case's potential influence on a regression model. It comes from the diagonal of the "hat" matrix and represents how different a case is from the remaining cases in the multivariate space of the predictors. In the generalized linear model for binary data, it is defined as:

$$\boldsymbol{H = W^{.5}X(X^TWX)^{-1}X^TW^{.5}}$$

with weights solved iteratively as part of the solution.

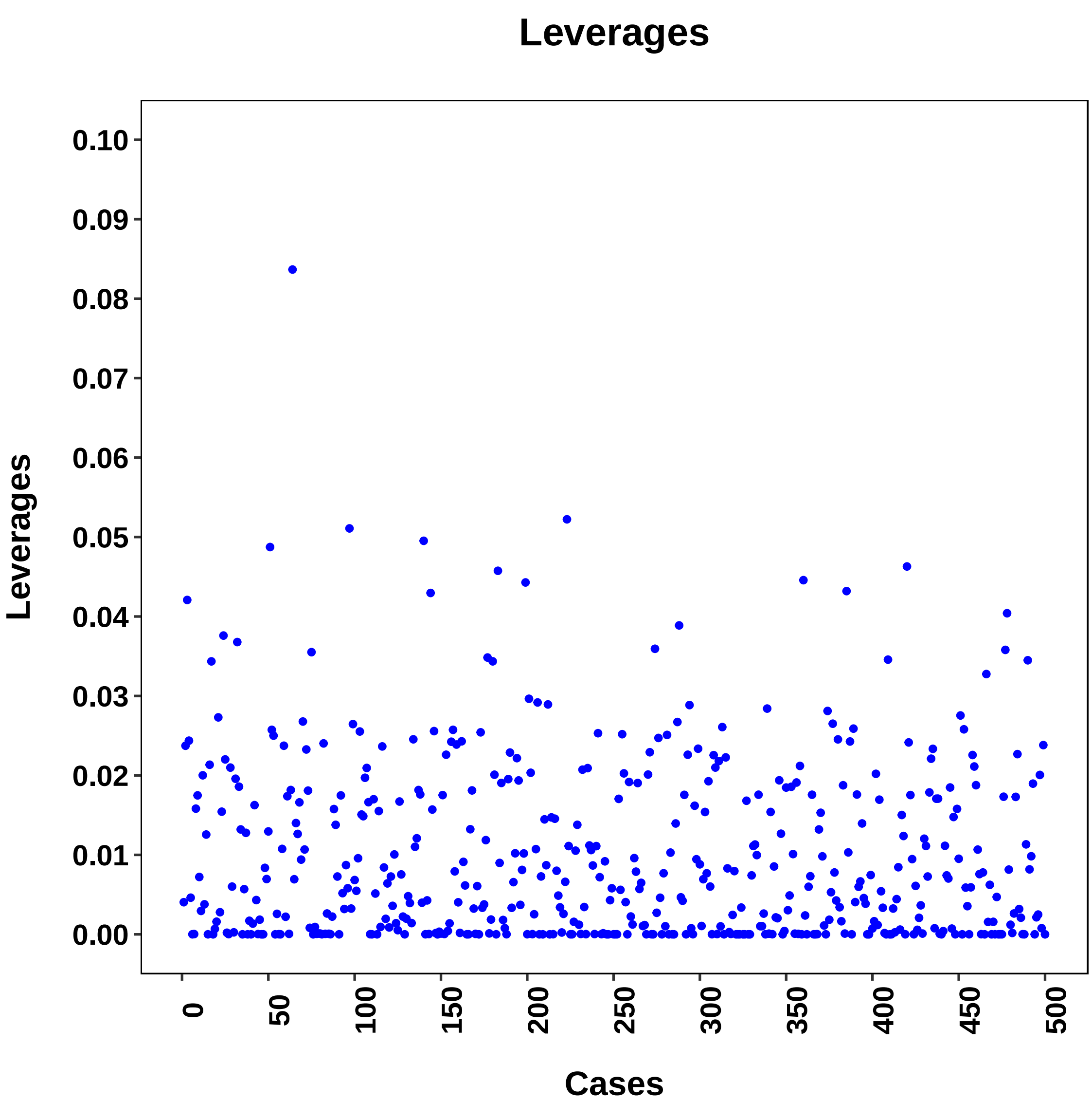
```r
Job_BLR_1_L <- hatvalues(Job_BLR_1)
Job_BLR_1_L <- as.data.frame(Job_BLR_1_L)
names(Job_BLR_1_L) <- c("leverage")
psych::describe(Job_BLR_1_L)

##    vars   n mean   sd median trimmed  mad min  max range skew
## X1    1 500 0.01 0.01   0.01    0.01 0.01   0 0.08  0.08 1.62
##    kurtosis se
## X1     4.04  0
```

```r
ggplot(Job_BLR_1_L, aes(x = 1, y = Job_BLR_1_L$leverage)) + geom_boxplot(fill = "grey",
    size = 1, color = "black", width = 0.01, outlier.colour = "red",
    outlier.shape = 19, outlier.size = 1) + scale_y_continuous(breaks = c(seq(0,
    0.1, 0.01))) + scale_x_continuous(breaks = c(1)) + coord_cartesian(xlim = c(1,
    1), ylim = c(0, 0.1)) + xlab("Entire Sample") + ylab("Leverage") +
    theme(text = element_text(size = 14, family = "sans", color = "black",
        face = "bold"), axis.text.y = element_text(colour = "black",
        size = 12, face = "bold"), axis.text.x = element_blank(),
        axis.title.x = element_text(margin = margin(15, 0, 0, 0),
            size = 14), axis.title.y = element_text(margin = margin(0,
            15, 0, 0), size = 14), axis.line.x = element_blank(),
        axis.line.y = element_blank(), plot.title = element_text(size = 16,
            face = "bold", margin = margin(0, 0, 20, 0), hjust = 0.5),
        panel.background = element_rect(fill = "white", linetype = 1,
            color = "black"), panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
        plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
        legend.title = element_blank()) + ggtitle("Leverages")
```




```
Job_BLR_1_L$Case_Num <- seq(1, length(Job_BLR_1_L[, 1]), 1)

ggplot(Job_BLR_1_L, aes(x = Case_Num, y = Job_BLR_1_L$leverage)) +
    geom_point(color = "blue", size = 1) + scale_y_continuous(breaks = c(seq(0,
    0.1, 0.01))) + scale_x_continuous(breaks = seq(0, 500, 50)) +
    coord_cartesian(xlim = c(1, 500), ylim = c(0, 0.1)) + xlab("Cases") +
    ylab("Leverages") + theme(text = element_text(size = 14, family = "sans",
    color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
```

```
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Leverages")
```




```
Extreme_Cases_U(Job_BLR_1_L$leverage, 1)

##       Case   Value
## [1,]    51 0.04874
## [2,]   140 0.04952
## [3,]    97 0.05109
## [4,]   223 0.05223
## [5,]    64 0.08367
```

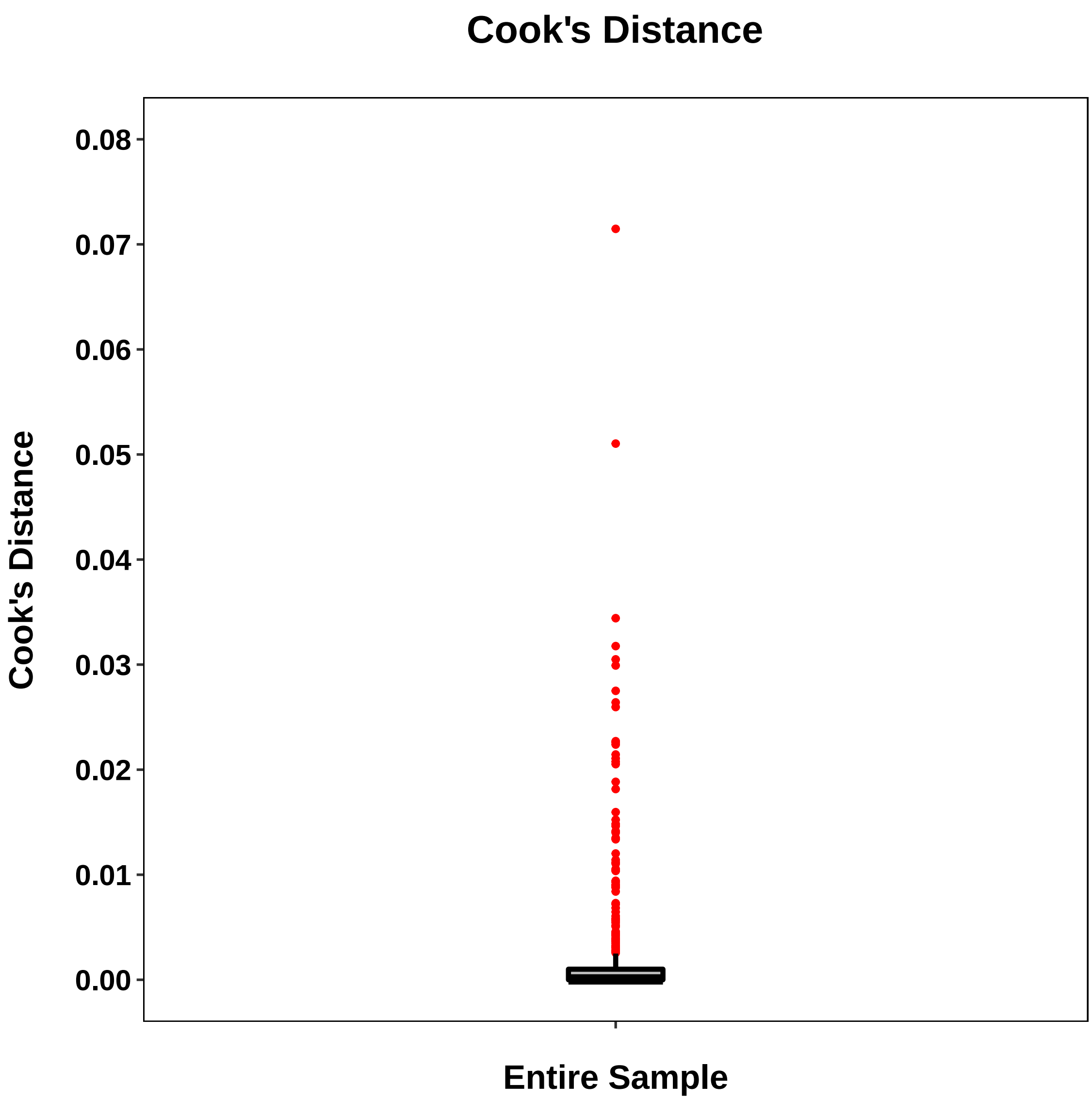
```r
Job[64, c(2:5, 9)]

##     sex  gre pubs years job_result
## 484   1 1297    8     9     No Job
```

4.1.3 Cook's Distance

Cook's distance is a general measure of influence. It does not have a convenient formula for logistic regression models, but is interpreted in the same way as in OLS regression. Cook's distance is the scaled change in fitted values. It represents the amount by which the fitted values change with the exclusion of a case from the model It can also be thought of as the normalized change in the vector of coefficients due to the deletion of an observation.

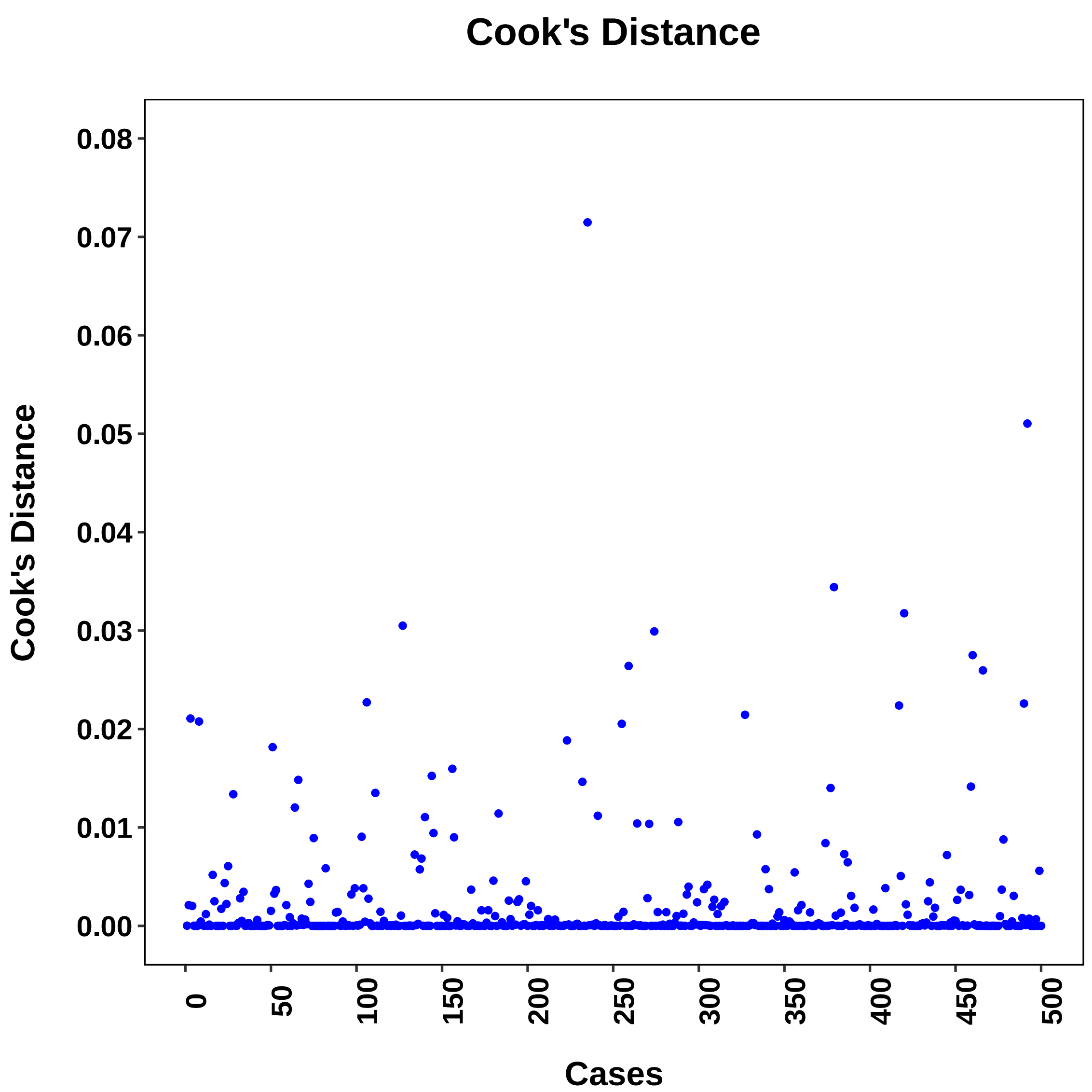
```r
Job_BLR_1_CD <- cooks.distance(Job_BLR_1)
Job_BLR_1_CD <- as.data.frame(Job_BLR_1_CD)
names(Job_BLR_1_CD) <- c("Cook")
psych::describe(Job_BLR_1_CD)

##    vars   n mean   sd median trimmed mad min  max range skew
## X1    1 500    0 0.01      0       0   0   0 0.07  0.07 5.49
##    kurtosis se
## X1    41.31  0
```

```r
ggplot(Job_BLR_1_CD, aes(x = 1, y = Job_BLR_1_CD$Cook)) + geom_boxplot(fill = "grey",
    size = 1, color = "black", width = 0.01, outlier.colour = "red",
    outlier.shape = 19, outlier.size = 1) + scale_y_continuous(breaks = c(seq(0,
    0.08, 0.01))) + scale_x_continuous(breaks = c(1)) + coord_cartesian(xlim = c(1,
    1), ylim = c(0, 0.08)) + xlab("Entire Sample") + ylab("Cook's Distance") +
    theme(text = element_text(size = 14, family = "sans", color = "black",
        face = "bold"), axis.text.y = element_text(colour = "black",
        size = 12, face = "bold"), axis.text.x = element_blank(),
        axis.title.x = element_text(margin = margin(15, 0, 0, 0),
            size = 14), axis.title.y = element_text(margin = margin(0,
            15, 0, 0), size = 14), axis.line.x = element_blank(),
        axis.line.y = element_blank(), plot.title = element_text(size = 16,
            face = "bold", margin = margin(0, 0, 20, 0), hjust = 0.5),
        panel.background = element_rect(fill = "white", linetype = 1,
            color = "black"), panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
        plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
        legend.title = element_blank()) + ggtitle("Cook's Distance")
```




```r
Job_BLR_1_CD$Case_Num <- seq(1, length(Job_BLR_1_CD[, 1]), 1)

ggplot(Job_BLR_1_CD, aes(x = Case_Num, y = Job_BLR_1_CD$Cook)) + geom_point(color = "blue",
    size = 1) + scale_y_continuous(breaks = c(seq(0, 0.08, 0.01))) +
    scale_x_continuous(breaks = seq(0, 500, 50)) + coord_cartesian(xlim = c(1,
    500), ylim = c(0, 0.08)) + xlab("Cases") + ylab("Cook's Distance") +
    theme(text = element_text(size = 14, family = "sans", color = "black",
        face = "bold"), axis.text.y = element_text(colour = "black",
        size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
        size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
        0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
        15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
        plot.title = element_text(size = 16, face = "bold", margin = margin(0,
            0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
```

```
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("Cook's Distance")
```




```
Extreme_Cases_U(Job_BLR_1_CD$Cook, 1)

##      Case   Value
## [1,]  127 0.03050
## [2,]  420 0.03176
## [3,]  379 0.03442
## [4,]  492 0.05104
## [5,]  235 0.07148
```

```r
Job[235, c(2:5, 9)]

##     sex  gre pubs years job_result
## 456   1 1317    7     9        Job
```

4.1.4 DFBETAS

Cook's distance is a general measure of influence. DFBETAS is coefficient-specific influence:

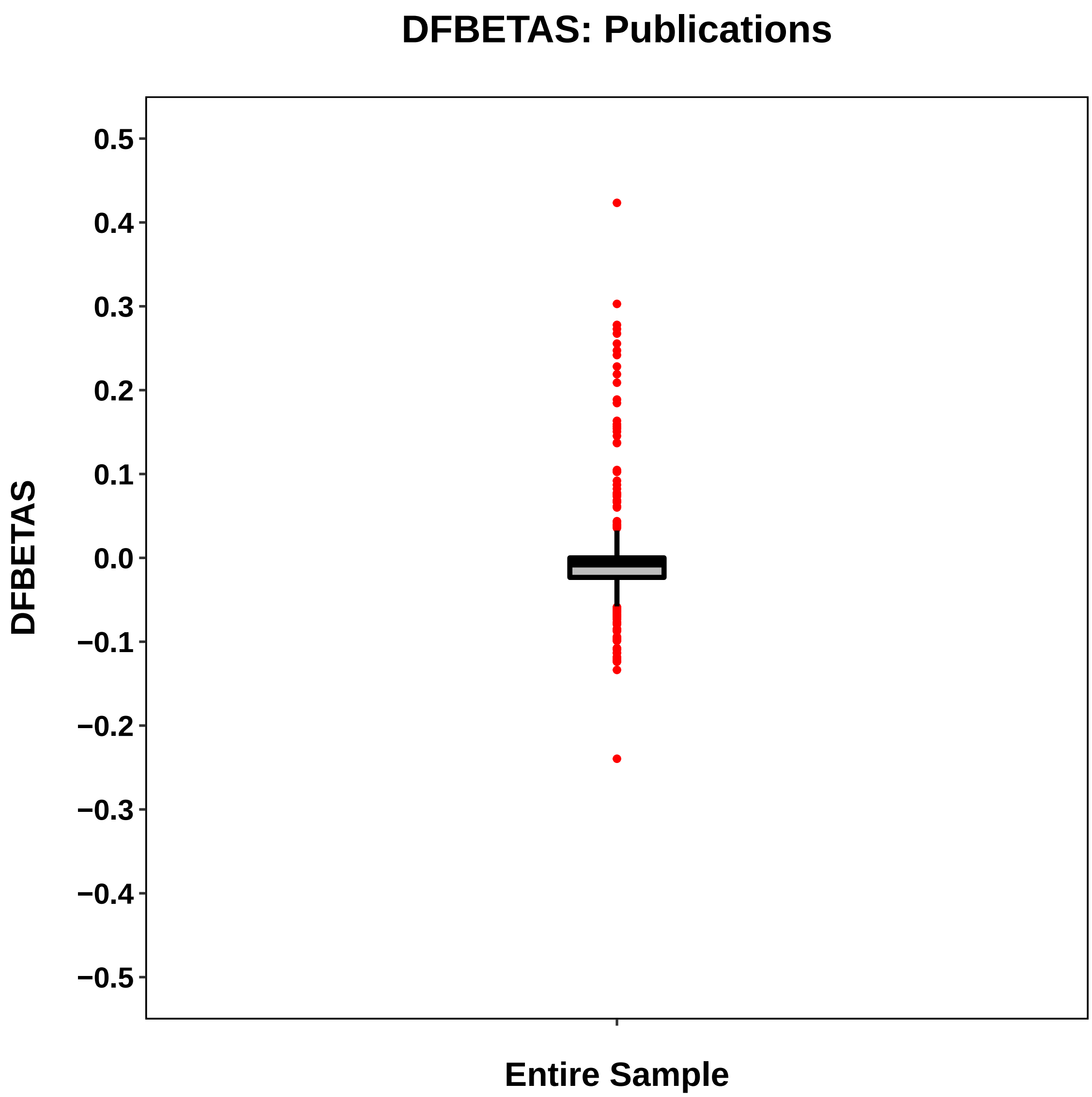
$$DFBETAS_{j(i)} = \frac{\beta_j - \beta_{j(i)}}{se(\beta_{j(i)})}$$

In other words, it is the standardized amount by which a coefficient changes when a case is excluded from the model.

```r
Job_BLR_1_DF <- cbind(dffits(Job_BLR_1), dfbetas(Job_BLR_1))
Job_BLR_1_DF <- as.data.frame(Job_BLR_1_DF)
names(Job_BLR_1_DF) <- c("dffits", "dfb_intercept", "dfb_sex", "dfb_gre",
    "dfb_pubs", "dfb_years")
psych::describe(Job_BLR_1_DF)

##               vars   n  mean   sd median trimmed  mad   min  max
## dffits           1 500  0.00 0.15   0.00   -0.01 0.04 -0.50 0.57
## dfb_intercept    2 500 -0.01 0.06  -0.01   -0.01 0.01 -0.20 0.38
## dfb_sex          3 500  0.00 0.06   0.00   -0.01 0.01 -0.23 0.35
## dfb_gre          4 500  0.01 0.05   0.01    0.01 0.01 -0.31 0.11
## dfb_pubs         5 500 -0.01 0.06  -0.01   -0.01 0.01 -0.24 0.42
## dfb_years        6 500  0.00 0.07   0.00    0.00 0.01 -0.27 0.28
##               range  skew kurtosis   se
## dffits         1.06  0.25     2.58 0.01
## dfb_intercept  0.58  2.79    13.98 0.00
## dfb_sex        0.58  1.45     7.47 0.00
## dfb_gre        0.42 -2.80    10.95 0.00
## dfb_pubs       0.66  2.46    11.99 0.00
## dfb_years      0.55  0.21     3.42 0.00
```

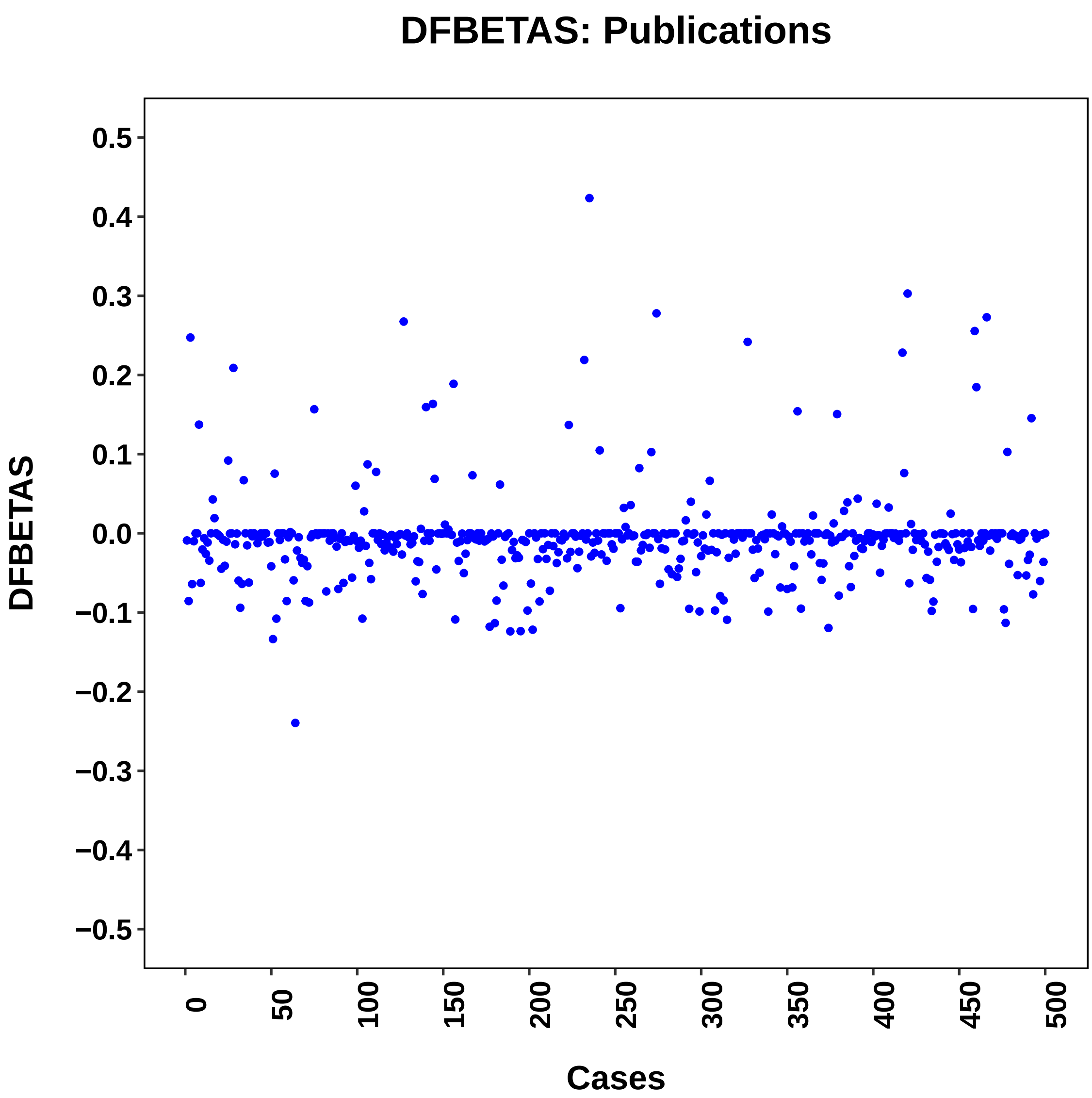
```r
ggplot(Job_BLR_1_DF, aes(x = 1, y = Job_BLR_1_DF$dfb_pubs)) + geom_boxplot(fill = "grey",
    size = 1, color = "black", width = 0.01, outlier.colour = "red",
    outlier.shape = 19, outlier.size = 1) + scale_y_continuous(breaks = round(c(seq(-0.5,
    0.5, 0.1)), 3)) + scale_x_continuous(breaks = c(1)) + coord_cartesian(xlim = c(1,
    1), ylim = c(-0.5, 0.5)) + xlab("Entire Sample") + ylab("DFBETAS") +
    theme(text = element_text(size = 14, family = "sans", color = "black",
        face = "bold"), axis.text.y = element_text(colour = "black",
        size = 12, face = "bold"), axis.text.x = element_blank(),
        axis.title.x = element_text(margin = margin(15, 0, 0, 0),
            size = 14), axis.title.y = element_text(margin = margin(0,
            15, 0, 0), size = 14), axis.line.x = element_blank(),
        axis.line.y = element_blank(), plot.title = element_text(size = 16,
            face = "bold", margin = margin(0, 0, 20, 0), hjust = 0.5),
        panel.background = element_rect(fill = "white", linetype = 1,
            color = "black"), panel.grid.major = element_blank(),
```

```
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("DFBETAS: Publications")
```




```
Job_BLR_1_DF$Case_Num <- seq(1, length(Job_BLR_1_DF[, 1]), 1)

ggplot(Job_BLR_1_DF, aes(x = Case_Num, y = Job_BLR_1_DF$dfb_pubs)) +
    geom_point(color = "blue", size = 1) + scale_y_continuous(breaks = round(c(seq(-0.5,
    0.5, 0.1)), 3)) + scale_x_continuous(breaks = seq(0, 500, 50)) +
    coord_cartesian(xlim = c(1, 500), ylim = c(-0.5, 0.5)) + xlab("Cases") +
    ylab("DFBETAS") + theme(text = element_text(size = 14, family = "sans",
    color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
```

```r
    size = 12, face = "bold", angle = 90), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + ggtitle("DFBETAS: Publications")
```




```r
Extreme_Cases_2(Job_BLR_1_DF$dfb_pubs, 1)

##       Case   Value
## [1,]    64 -0.2396
```

```
## [2,]   51 -0.1336
## [3,]  189 -0.1239
## [4,]  274  0.2778
## [5,]  420  0.3029
## [6,]  235  0.4234

Job[235, c(2:5, 9)]

##     sex  gre pubs years job_result
## 456   1 1317    7     9        Job
```

4.1.5 DFFITS

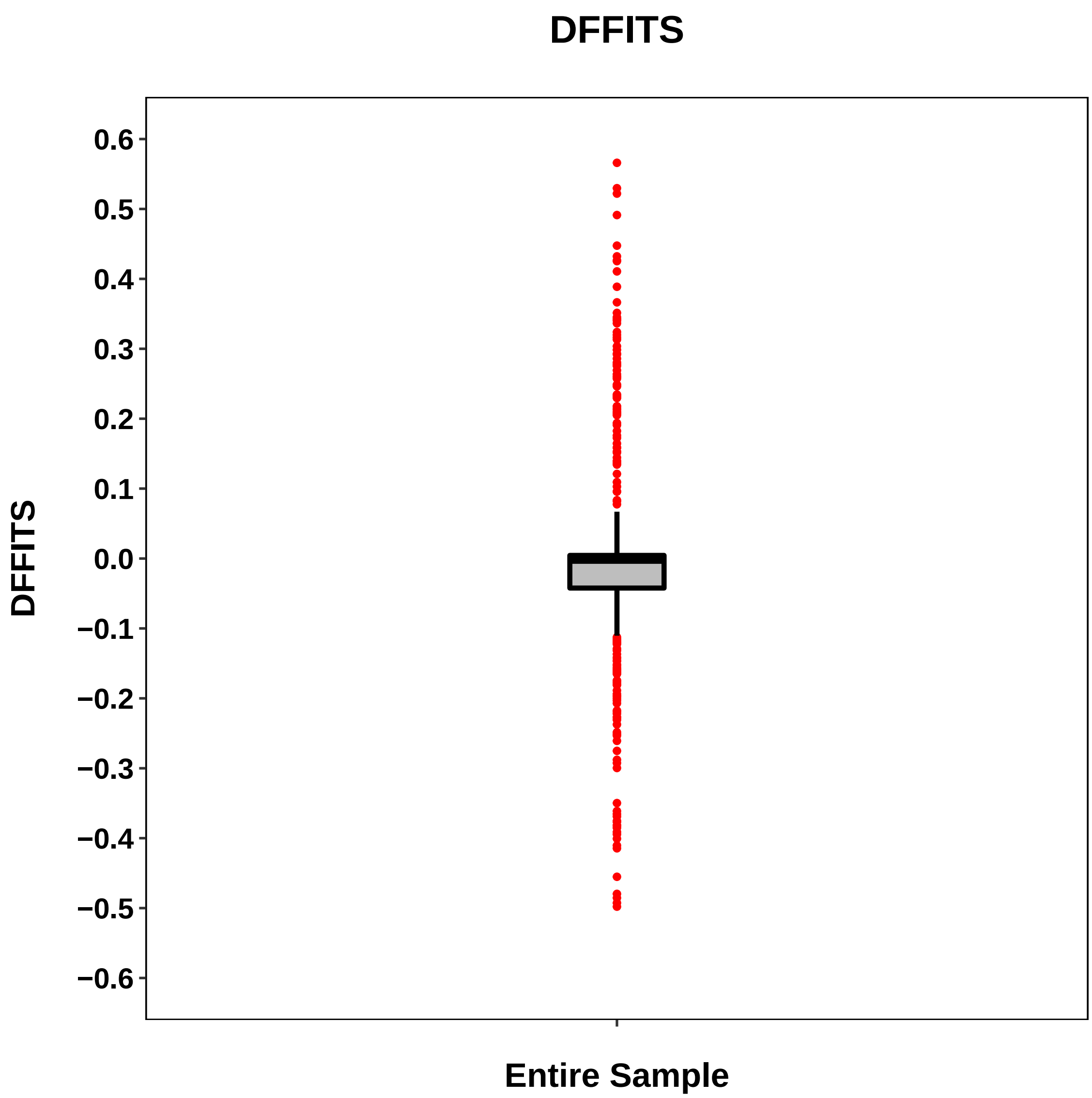
DFFITS is the extent to which the fitted values for a case change when the case is excluded from the model. The difference is studentized. In OLS regression, DFFITS is defined as:

$$DFFITS_{j(i)} = \frac{\hat{y}_j - \hat{y}_{j(i)}}{s_i\sqrt{h_{ii}}}$$

with s defined as the standard error when the case is excluded. The definition is more complicated in logistic regression, but the interpretation is similar. Cases with large DFFITS are overly influential in the model.

DFFITS provides the same kind of information as Cook's distance.

```
ggplot(Job_BLR_1_DF, aes(x = 1, y = Job_BLR_1_DF$dffits)) + geom_boxplot(fill = "grey",
    size = 1, color = "black", width = 0.01, outlier.colour = "red",
    outlier.shape = 19, outlier.size = 1) + scale_y_continuous(breaks = round(c(seq(-0.6,
    0.6, 0.1)), 3)) + scale_x_continuous(breaks = c(1)) + coord_cartesian(xlim = c(1,
    1), ylim = c(-0.6, 0.6)) + xlab("Entire Sample") + ylab("DFFITS") +
    theme(text = element_text(size = 14, family = "sans", color = "black",
        face = "bold"), axis.text.y = element_text(colour = "black",
        size = 12, face = "bold"), axis.text.x = element_blank(),
        axis.title.x = element_text(margin = margin(15, 0, 0, 0),
            size = 14), axis.title.y = element_text(margin = margin(0,
            15, 0, 0), size = 14), axis.line.x = element_blank(),
        axis.line.y = element_blank(), plot.title = element_text(size = 16,
            face = "bold", margin = margin(0, 0, 20, 0), hjust = 0.5),
        panel.background = element_rect(fill = "white", linetype = 1,
            color = "black"), panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
        plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
        legend.title = element_blank()) + ggtitle("DFFITS")
```

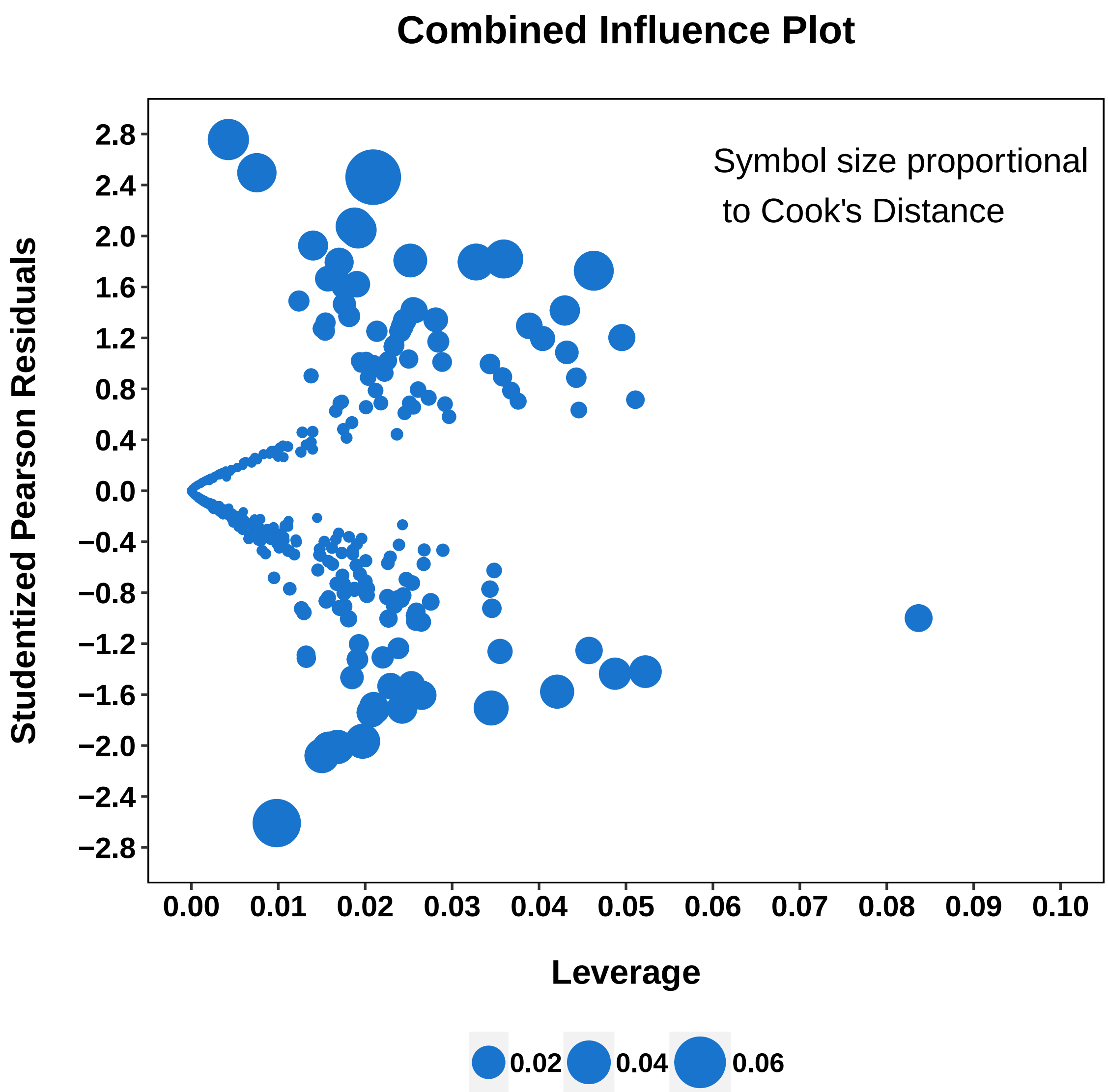



```
Extreme_Cases_2(Job_BLR_1_DF$dffits, 1)
```

```
##      Case   Value
## [1,]  223 -0.4979
## [2,]    3 -0.4926
## [3,]   51 -0.4854
## [4,]  274  0.5220
## [5,]  235  0.5297
## [6,]  420  0.5660
```

4.1.6 Combined Influence Plot

```
plot_data <- cbind(Job_BLR_1_CD, Job_BLR_1_L$leverage, Job_BLR_1_R$stud_pearson)
names(plot_data) <- c("Cook", "Case_Num", "Leverage", "Student_Pearson")

ggplot(plot_data, aes(x = Leverage, y = Student_Pearson, size = Cook)) +
    geom_point(color = "dodgerblue3") + scale_size(range = c(1, 10)) +
    scale_y_continuous(breaks = round(c(seq(-2.8, 2.8, 0.4)), 3)) +
    scale_x_continuous(breaks = seq(0, 0.1, 0.01)) + coord_cartesian(xlim = c(0,
    0.1), ylim = c(-2.8, 2.8)) + xlab("Leverage") + ylab("Studentized Pearson Residuals") +
    theme(text = element_text(size = 14, family = "sans", color = "black",
        face = "bold"), axis.text.y = element_text(colour = "black",
        size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
        size = 12, face = "bold", angle = 0), axis.title.x = element_text(margin = margin(15,
        0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
        15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
        plot.title = element_text(size = 16, face = "bold", margin = margin(0,
            0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
            linetype = 1, color = "black"), panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
        plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
        legend.title = element_blank()) + annotate("text", x = 0.06,
    y = 2.4, label = "Symbol size proportional\n to Cook's Distance",
    size = 5, hjust = 0) + ggtitle("Combined Influence Plot")
```



4.1.7 Basic Logistic Regression Model Case Omitted

Here we run the binary logistic regression model with the most unusual case removed.

```
Job_BLR_1 <- glm(Job$job ~ gre_c + pubs_c + years_c + sex_D, family = binomial("logit"),
    subset = c(-235), data = Job)
summary(Job_BLR_1)

##
## Call:
## glm(formula = Job$job ~ gre_c + pubs_c + years_c + sex_D, family = binomial("logit"),
##     data = Job, subset = c(-235))
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5314  -0.3016  -0.0124   0.0661   2.7451
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.92832    0.49389   -7.95  1.8e-15
## gre_c       -0.01518    0.00237   -6.42  1.4e-10
## pubs_c       2.03705    0.22675    8.98  < 2e-16
## years_c     -1.52950    0.19884   -7.69  1.4e-14
## sex_D       -0.31870    0.35600   -0.90     0.37
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 582.63  on 498  degrees of freedom
## Residual deviance: 219.15  on 494  degrees of freedom
## AIC: 229.2
##
## Number of Fisher Scoring iterations: 8
```

4.2 Multicollinearity

The variance inflation factor (VIF) is a common index for multicollinearity. It is the inverse of $1 - R^2$, with R being the multiple correlation relating a given predictor to the remaining predictors. This is also the reciprocal of the tolerance. It is called the variance inflation factor because it indicates the amount by which the variance for a coefficient is inflated because of dependence with other predictors. A VIF of 2.00 indicates that the variance (or the square of the standard error) of a particular coefficient is 2 times larger than it would be if that predictor was completely uncorrelated with all the other predictors. A VIF of 2.00 also means that a predictor shares 50% of its variance with other predictors. Some argue that a VIF of 4 or greater is a cause for concern, but some argue for even higher thresholds (5 to 10).

```
vif(Job_BLR_1)

##   gre_c  pubs_c years_c   sex_D
##   2.015   2.811   2.139   1.022

cor(model.matrix(Job_BLR_1)[, -1])

##            gre_c   pubs_c  years_c    sex_D
## gre_c    1.00000  0.30867 -0.09221  0.01890
## pubs_c   0.30867  1.00000 -0.29056 -0.06202
## years_c -0.09221 -0.29056  1.00000  0.11859
## sex_D    0.01890 -0.06202  0.11859  1.00000

1/vif(Job_BLR_1)

##   gre_c  pubs_c years_c   sex_D
##  0.4963  0.3558  0.4675  0.9787
```

4.3 Overdispersion

By selecting the binomial model, we assume that the dispersion is consistent with that model. In other words, in a binomial distribution we assume the mean and variance to be related (mean = p, variance = p[1-p]). Most commonly the variance will be larger than expected, known as overdispersion. The quasi-binomial family allows for the variance to be separately estimated, providing for overdispersed (and underdispersed) models.

```
Job_BLR_5 <- glm(Job$job ~ gre_c + pubs_c + years_c + sex_D, family = binomial("logit"),
    data = Job)
summary(Job_BLR_5)

##
## Call:
## glm(formula = Job$job ~ gre_c + pubs_c + years_c + sex_D, family = binomial("logit"),
##     data = Job)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5596  -0.3111  -0.0142   0.0755   2.7257
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.71205    0.46554   -7.97  1.5e-15
## gre_c       -0.01470    0.00231   -6.37  1.8e-10
## pubs_c       1.99614    0.22058    9.05  < 2e-16
## years_c     -1.43390    0.18667   -7.68  1.6e-14
## sex_D       -0.40619    0.35023   -1.16     0.25
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 585.24  on 499  degrees of freedom
## Residual deviance: 225.23  on 495  degrees of freedom
## AIC: 235.2
##
## Number of Fisher Scoring iterations: 8

Job_BLR_8 <- glm(Job$job ~ gre_c + pubs_c + years_c + sex_D, family = quasibinomial("logit"),
    data = Job)
summary(Job_BLR_8)

##
## Call:
## glm(formula = Job$job ~ gre_c + pubs_c + years_c + sex_D, family = quasibinomial("logit"),
##     data = Job)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5596  -0.3111  -0.0142   0.0755   2.7257
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.71205    0.35116  -10.57  < 2e-16
## gre_c       -0.01470    0.00174   -8.45  3.2e-16
## pubs_c       1.99614    0.16639   12.00  < 2e-16
```

```
## years_c     -1.43390    0.14081  -10.18  < 2e-16
## sex_D       -0.40619    0.26418   -1.54     0.12
##
## (Dispersion parameter for quasibinomial family taken to be 0.569)
##
##     Null deviance: 585.24  on 499  degrees of freedom
## Residual deviance: 225.23  on 495  degrees of freedom
## AIC: NA
##
## Number of Fisher Scoring iterations: 8
```

```
confint(Job_BLR_8)
```

```
## Waiting for profiling to be done...
```

```
##                2.5 %   97.5 %
## (Intercept) -4.44263 -3.06235
## gre_c       -0.01827 -0.01143
## pubs_c       1.69023  2.34398
## years_c     -1.72579 -1.17265
## sex_D       -0.92713  0.11102
```

```
confint.default(Job_BLR_8)
```

```
##                2.5 %   97.5 %
## (Intercept) -4.40031 -3.02379
## gre_c       -0.01811 -0.01129
## pubs_c       1.67003  2.32225
## years_c     -1.70987 -1.15792
## sex_D       -0.92396  0.11159
```

```
exp(cbind(OR = coef(Job_BLR_8), confint(Job_BLR_8)))
```

```
## Waiting for profiling to be done...
```

```
##                  OR   2.5 %   97.5 %
## (Intercept) 0.02443 0.01176  0.04678
## gre_c       0.98540 0.98190  0.98863
## pubs_c      7.36059 5.42075 10.42267
## years_c     0.23838 0.17803  0.30954
## sex_D       0.66619 0.39569  1.11742
```

```
with(Job_BLR_8, null.deviance - deviance)
```

```
## [1] 360
```

```
with(Job_BLR_8, df.null - df.residual)
```

```
## [1] 4
```

```
with(Job_BLR_8, pchisq(null.deviance - deviance, df.null - df.residual,
    lower.tail = FALSE))
```

```
## [1] 1.208e-76
```

```r
overdispersion <- function(model) {
    Overdispersion <- sum(resid(model, type = "pearson")^2)/df.residual(model)
    Overdispersion_chi_Square <- sum(resid(model, type = "pearson")^2)
    Overdispersion_p_value <- pchisq(Overdispersion_chi_Square, df = df.residual(model),
        lower.tail = FALSE)
    c(Overdispersion = Overdispersion, Chi_Sq = Overdispersion_chi_Square,
        df = df.residual(model), p = Overdispersion_p_value)
}

overdispersion(Job_BLR_5)

## Overdispersion         Chi_Sq             df              p
##          0.569        281.638        495.000          1.000
```

4.4 Adequacy of the Logistic Model

Is the logistic model adequate? To test this, we can save the fitted logistic values, square them, and then enter them as an additional predictor in the model. This tests if there is any relationship beyond a linear logistic in the data. The coefficient for this new predictor should be clearly nonsignificant.

```r
Job_BLR_5 <- glm(Job$job ~ gre_c + pubs_c + years_c + sex_D, family = binomial("logit"),
    data = Job)
summary(Job_BLR_5)

##
## Call:
## glm(formula = Job$job ~ gre_c + pubs_c + years_c + sex_D, family = binomial("logit"),
##     data = Job)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5596  -0.3111  -0.0142   0.0755   2.7257
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.71205    0.46554   -7.97  1.5e-15
## gre_c       -0.01470    0.00231   -6.37  1.8e-10
## pubs_c       1.99614    0.22058    9.05  < 2e-16
## years_c     -1.43390    0.18667   -7.68  1.6e-14
## sex_D       -0.40619    0.35023   -1.16     0.25
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 585.24  on 499  degrees of freedom
## Residual deviance: 225.23  on 495  degrees of freedom
## AIC: 235.2
##
## Number of Fisher Scoring iterations: 8

Job <- cbind(Job, predict(Job_BLR_5))

names(Job) <- c(names(Job[-length(Job[1, ])]), "p_logit")
```

```r
Job_BLR_5a <- glm(Job$job ~ gre_c + pubs_c + years_c + sex_D + I(p_logit^2),
    family = binomial("logit"), data = Job)

## Warning:  glm.fit:  fitted probabilities numerically 0 or 1 occurred

summary(Job_BLR_5a)

##
## Call:
## glm(formula = Job$job ~ gre_c + pubs_c + years_c + sex_D + I(p_logit^2),
##     family = binomial("logit"), data = Job)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4456  -0.2875  -0.0041   0.1314   2.8355
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.56595    0.48639   -7.33  2.3e-13
## gre_c        -0.01454    0.00228   -6.39  1.7e-10
## pubs_c        1.96572    0.21699    9.06  < 2e-16
## years_c      -1.41011    0.18550   -7.60  2.9e-14
## sex_D        -0.41671    0.35060   -1.19     0.23
## I(p_logit^2) -0.03206    0.03773   -0.85     0.40
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 585.24  on 499  degrees of freedom
## Residual deviance: 224.55  on 494  degrees of freedom
## AIC: 236.6
##
## Number of Fisher Scoring iterations: 10
```

5 Multinomial Logistic Regression

When there are more than two response categories, the multinomial model is used. In this model, one of the response categories becomes the reference and the analysis compares each of the other response categories to it. Multinomial regression is essential a collection of binary logistic regressions, each comparing a response category to the reference. In the following model, we will select the "interview only" group as the reference. This will allow us to determine what distinguishes those who do not get interviews from those who do get interview (but not a job offer), and, to determine what distinguishes those who get job offers from those who don't (but are interviewed).

5.1 Rearrange Data File for mlogit

The data need to be in a particular arrangement for the mlogit package.

```
jobs <- read.table("jobs.csv", sep = ",", header = TRUE)
jobs <- as.data.frame(jobs)
jobs$sex_D <- ifelse(jobs$sex == 2, 1, 0)
jobs$ordered <- factor(jobs$ordered, levels = c(1, 2, 3), labels = c("Not Interviewed",
    "Interviewed", "Hired"))
jobs$group_2 <- factor(jobs$group, levels = c(1, 2, 3), labels = c("Not Interviewed",
    "Hired", "Interviewed"))
with(jobs, table(group_2)/500)

## group_2
## Not Interviewed           Hired     Interviewed
##           0.240           0.272           0.488

with(jobs, table(group_2))

## group_2
## Not Interviewed           Hired     Interviewed
##             120             136             244

jobs_2 <- matrix(NA, nrow = 1500, ncol = 6)
colnames(jobs_2) <- c("case", "outcome", "sex", "gre", "years", "pubs")
jobs_2 <- as.data.frame(jobs_2)

counter <- 0
for (i in 1:500) {
    counter <- counter + 1
    jobs_2[counter, 1] <- i
    if (jobs[i, "group"] == 1) {
        jobs_2[counter, 2] <- 1
    } else {
        jobs_2[counter, 2] <- 0
    }
    jobs_2[counter, 3] <- jobs[i, "sex"]
    jobs_2[counter, 4] <- jobs[i, "gre"]
    jobs_2[counter, 5] <- jobs[i, "years"]
    jobs_2[counter, 6] <- jobs[i, "pubs"]

    counter <- counter + 1
    jobs_2[counter, 1] <- i
    if (jobs[i, "group"] == 2) {
        jobs_2[counter, 2] <- 1
    } else {
        jobs_2[counter, 2] <- 0
    }
    jobs_2[counter, 3] <- jobs[i, "sex"]
    jobs_2[counter, 4] <- jobs[i, "gre"]
    jobs_2[counter, 5] <- jobs[i, "years"]
    jobs_2[counter, 6] <- jobs[i, "pubs"]

    counter <- counter + 1
    jobs_2[counter, 1] <- i
```

```r
    if (jobs[i, "group"] == 3) {
        jobs_2[counter, 2] <- 1
    } else {
        jobs_2[counter, 2] <- 0
    }
    jobs_2[counter, 3] <- jobs[i, "sex"]
    jobs_2[counter, 4] <- jobs[i, "gre"]
    jobs_2[counter, 5] <- jobs[i, "years"]
    jobs_2[counter, 6] <- jobs[i, "pubs"]
}

jobs_2$outcome.ids <- factor(rep(1:3, 500), labels = c("Not Interviewed",
    "Hired", "Interviewed"))

# The mlogit( ) function requires the data to be reformatted to an
# mlogit object. The outcome variable is specified with the
# 'choice' option.
J <- mlogit.data(jobs_2, shape = "long", choice = "outcome", alt.var = "outcome.ids")
```

5.2 Model Fit

The mlogit( ) function from the mlogit package is used to fit a multinomial logistic regression model. The "not interviewed" group is used as the reference. That is changed later.

```r
Ref_Level <- "Not Interviewed"
Job_MLR_1 <- mlogit(outcome ~ 0 | 1 + sex + gre + years + pubs, data = J,
    reflevel = Ref_Level)
summary(Job_MLR_1)

##
## Call:
## mlogit(formula = outcome ~ 0 | 1 + sex + gre + years + pubs,
##     data = J, reflevel = Ref_Level, method = "nr", print.level = 0)
##
## Frequencies of alternatives:
## Not Interviewed           Hired     Interviewed
##           0.240           0.272           0.488
##
## nr method
## 11 iterations, 0h:0m:0s
## g'(-H)^-1g = 3.51E-08
## gradient close to zero
##
## Coefficients :
##                    Estimate Std. Error z-value    Pr(>|z|)
## 1                  -39.7187     9.7202   -4.09 0.000043847
## 2                   27.8101     5.2043    5.34 0.000000091
## Hired:sex            0.9829     1.5909    0.62     0.53671
## Interviewed:sex      1.3891     1.5520    0.90     0.37077
## Hired:gre           -0.0467     0.0109   -4.27 0.000019394
## Interviewed:gre     -0.0320     0.0107   -3.00     0.00274
## Hired:years         -5.0333     0.9902   -5.08 0.000000371
```

```
## Interviewed:years  -3.6000     0.9723   -3.70     0.00021
## Hired:pubs           6.4368     1.3141    4.90 0.000000968
## Interviewed:pubs     4.4414     1.2953    3.43     0.00061
##
## Log-Likelihood: -125
## McFadden R^2:  0.76
## Likelihood ratio test : chisq = 796 (p.value = <2e-16)
```

```
S_Job_MLR_1 <- summary(Job_MLR_1)
```

5.3 Odds Ratios and Confidence Intervals

```
OR <- exp(S_Job_MLR_1$CoefTable[, 1])
OR_LL <- exp(S_Job_MLR_1$CoefTable[, 1] - 1.96 * S_Job_MLR_1$CoefTable[,
    2])
OR_UL <- exp(S_Job_MLR_1$CoefTable[, 1] + 1.96 * S_Job_MLR_1$CoefTable[,
    2])
OR
```

```
##                 1                 2         Hired:sex
##         5.629e-18         1.196e+12         2.672e+00
##   Interviewed:sex         Hired:gre   Interviewed:gre
##         4.011e+00         9.544e-01         9.685e-01
##       Hired:years Interviewed:years        Hired:pubs
##         6.517e-03         2.732e-02         6.244e+02
##  Interviewed:pubs
##         8.489e+01
```

```
OR_LL
```

```
##                 1                 2         Hired:sex
##         2.995e-26         4.444e+07         1.182e-01
##   Interviewed:sex         Hired:gre   Interviewed:gre
##         1.915e-01         9.341e-01         9.484e-01
##       Hired:years Interviewed:years        Hired:pubs
##         9.358e-04         4.063e-03         4.752e+01
##  Interviewed:pubs
##         6.703e+00
```

```
OR_UL
```

```
##                 1                 2         Hired:sex
##         1.058e-09         3.219e+16         6.041e+01
##   Interviewed:sex         Hired:gre   Interviewed:gre
##         8.401e+01         9.750e-01         9.890e-01
##       Hired:years Interviewed:years        Hired:pubs
##         4.539e-02         1.837e-01         8.205e+03
##  Interviewed:pubs
##         1.075e+03
```

Odds Ratios			
Predictor	Odds Ratio	Lower 95%	Upper 95%
Interviewed: Sex	4.011	0.192	84.013
Interviewed: GRE	0.968	0.948	0.989
Interviewed: Years	0.027	0.004	0.184
Interviewed: Pubs	84.893	6.703	1075.197
Hired: Sex	2.672	0.118	60.409
Hired: GRE	0.954	0.934	0.975
Hired: Years	0.007	0.001	0.045
Hired: Pubs	624.389	47.515	8204.987

5.4 Classification

```
fit_prob <- as.data.frame(Job_MLR_1$probabilities)
for (i in 1:500) {
    if ((Job_MLR_1$probabilities[i, 1] > Job_MLR_1$probabilities[i,
        2]) & (Job_MLR_1$probabilities[i, 1] > Job_MLR_1$probabilities[i,
        3])) {
        jobs[i, "p_group"] <- names(fit_prob)[1]
    } else if ((Job_MLR_1$probabilities[i, 2] > Job_MLR_1$probabilities[i,
        1]) & (Job_MLR_1$probabilities[i, 2] > Job_MLR_1$probabilities[i,
        3])) {
        jobs[i, "p_group"] <- names(fit_prob)[2]
    } else if ((Job_MLR_1$probabilities[i, 3] > Job_MLR_1$probabilities[i,
        1]) & (Job_MLR_1$probabilities[i, 3] > Job_MLR_1$probabilities[i,
        2])) {
        jobs[i, "p_group"] <- names(fit_prob)[3]
    }
}

# cross_class <- with(jobs[order(jobs$group_2),],
# table(group_2,p_group))
cross_class <- with(jobs[order(jobs$p_group), ], table(group_2, p_group))
cross_class

##                   p_group
## group_2            Hired Interviewed Not Interviewed
##   Not Interviewed      0           3             117
##   Hired              105          31               0
##   Interviewed         25         216               3

cross_class/sum(cross_class)

##                   p_group
## group_2            Hired Interviewed Not Interviewed
##   Not Interviewed 0.000       0.006           0.234
##   Hired           0.210       0.062           0.000
##   Interviewed     0.050       0.432           0.006

correct_class <- 0
for (i in 1:3) {
```

```
    for (j in 1:3) {
        if (row.names(cross_class)[i] == colnames(cross_class)[j]) {
            correct_class <- correct_class + cross_class[i, j]
        }
    }
}
prop_class <- cross_class/sum(cross_class)
correct_class/sum(cross_class)

## [1] 0.876

summary(cross_class)

## Number of cases in table: 500
## Number of factors: 2
## Test for independence of all factors:
##  Chisq = 694, df = 4, p-value = 8e-149
```

	Classification		
	Not Interviewed	***Interviewed***	***Hired***
Predict: Not Interviewed	***117***	***3***	***0***
Predict: Interviewed	***3***	***216***	***31***
Predict: Hired	***0***	***25***	***105***

	Classification		
	Not Interviewed	***Interviewed***	***Hired***
Predict: Not Interviewed	***0.234***	***0.006***	***0***
Predict: Interviewed	***0.006***	***0.432***	***0.062***
Predict: Hired	***0***	***0.05***	***0.21***

5.5 Comparison of Coefficients

```
M1 = matrix(c(0, 0, 1, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, -1, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 1, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    1, -1), nrow = 4, ncol = 10, byrow = TRUE)
rownames(M1) <- c("Sex Difference", "GRE Difference", "Years Difference",
    "Pubs Difference")
M1

##                   [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
## Sex Difference       0    0    1   -1    0    0    0    0    0
## GRE Difference       0    0    0    0    1   -1    0    0    0
## Years Difference     0    0    0    0    0    0    1   -1    0
## Pubs Difference      0    0    0    0    0    0    0    0    1
##                   [,10]
## Sex Difference        0
## GRE Difference        0
## Years Difference      0
```

```
## Pubs Difference     -1

glht_M1 <- glht(Job_MLR_1, linfct = M1, alternative = "two.sided",
    rhs = 0)
summary(glht_M1)

##
##   Simultaneous Tests for General Linear Hypotheses
##
## Fit: mlogit(formula = outcome ~ 0 | 1 + sex + gre + years + pubs,
##     data = J, reflevel = Ref_Level, method = "nr", print.level = 0)
##
## Linear Hypotheses:
##                         Estimate Std. Error z value Pr(>|z|)
## Sex Difference == 0   -0.40619    0.35019   -1.16      0.6
## GRE Difference == 0   -0.01470    0.00231   -6.37  <0.0001
## Years Difference == 0 -1.43330    0.18684   -7.67  <0.0001
## Pubs Difference == 0   1.99538    0.22080    9.04  <0.0001
## (Adjusted p values reported -- single-step method)

confint(glht_M1, calpha = univariate_calpha())

##
##   Simultaneous Confidence Intervals
##
## Fit: mlogit(formula = outcome ~ 0 | 1 + sex + gre + years + pubs,
##     data = J, reflevel = Ref_Level, method = "nr", print.level = 0)
##
## Quantile = 1.96
## 95% confidence level
##
##
## Linear Hypotheses:
##                         Estimate lwr     upr
## Sex Difference == 0   -0.4062  -1.0926  0.2802
## GRE Difference == 0   -0.0147  -0.0192 -0.0102
## Years Difference == 0 -1.4333  -1.7995 -1.0671
## Pubs Difference == 0   1.9954   1.5626  2.4281
```

5.6 Predicted Outcomes

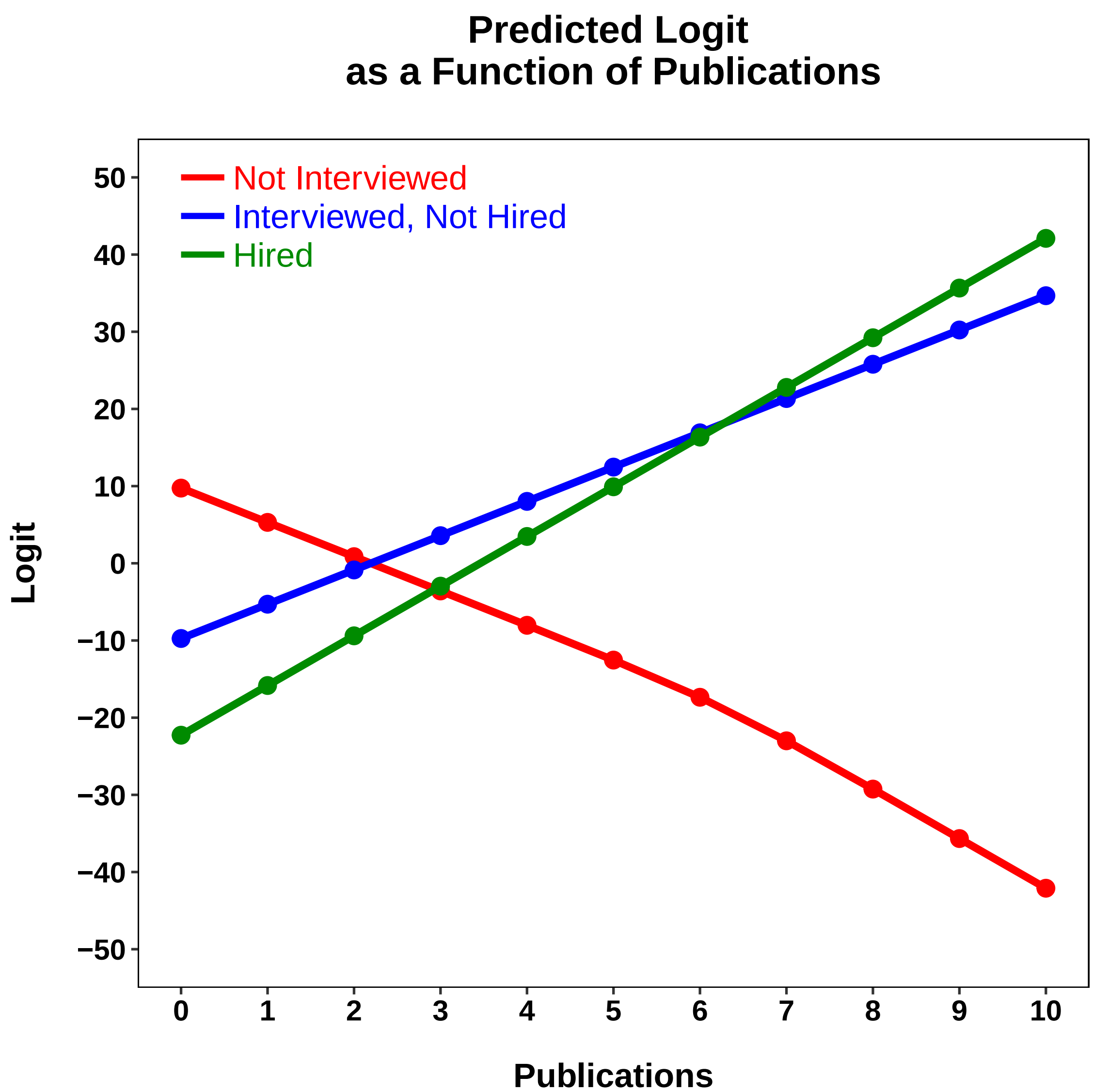
Getting predicted outcomes is complicated with the mlogit( ) function. The intercepts reported in the output do not correspond to those provided by other packages or other software. Using them does not produce correct predicted logits (or probabilities or odds). In the following, the multinom( ) function from the nnet package is used to get the correct intercepts. The multinom( ) has its own problems in that it does not produce the correct standard errors for model coefficients. Combining elements from both packages is currently the best that can be done.

```
pred_data <- matrix(NA, nrow = 11, ncol = 10)
colnames(pred_data) <- c("pubs", "p_not_I", "p_I", "p_H", "L_not_I",
    "L_I", "L_H", "O_not_I", "O_I", "O_H")

M_sex <- mean(jobs$sex)
```

```r
M_gre <- mean(jobs$gre)
M_years <- mean(jobs$years)

jobs$group_2 <- relevel(jobs$group_2, ref = Ref_Level)
MLR <- multinom(group_2 ~ sex + gre + years + pubs, data = jobs)

## # weights:  18 (10 variable)
## initial  value 549.306144
## iter  10 value 183.966154
## iter  20 value 129.042481
## iter  30 value 126.281154
## iter  40 value 125.503979
## final  value 125.492940
## converged

Job_MLR_1$coefficients[1] <- coefficients(MLR)[1]
Job_MLR_1$coefficients[2] <- coefficients(MLR)[2]

for (pubs in 0:10) {
    pred_data[pubs + 1, 1] <- pubs
    L_H <- Job_MLR_1$coefficients[1] + Job_MLR_1$coefficients[3] *
        M_sex + Job_MLR_1$coefficients[5] * M_gre + Job_MLR_1$coefficients[7] *
        M_years + Job_MLR_1$coefficients[9] * pubs
    L_I <- Job_MLR_1$coefficients[2] + Job_MLR_1$coefficients[4] *
        M_sex + Job_MLR_1$coefficients[6] * M_gre + Job_MLR_1$coefficients[8] *
        M_years + Job_MLR_1$coefficients[10] * pubs
    pred_data[pubs + 1, 2] <- 1/(1 + exp(L_H) + exp(L_I))
    pred_data[pubs + 1, 3] <- exp(L_I)/(1 + exp(L_H) + exp(L_I))
    pred_data[pubs + 1, 4] <- exp(L_H)/(1 + exp(L_H) + exp(L_I))
    pred_data[pubs + 1, 5] <- log(pred_data[pubs + 1, 2]/(1 - pred_data[pubs +
        1, 2]))
    pred_data[pubs + 1, 6] <- L_I
    pred_data[pubs + 1, 7] <- L_H
    pred_data[pubs + 1, 8] <- pred_data[pubs + 1, 2]/(1 - pred_data[pubs +
        1, 2])
    pred_data[pubs + 1, 9] <- pred_data[pubs + 1, 3]/(1 - pred_data[pubs +
        1, 3])
    pred_data[pubs + 1, 10] <- pred_data[pubs + 1, 4]/(1 - pred_data[pubs +
        1, 4])
}
pred_data <- as.data.frame(pred_data)
```

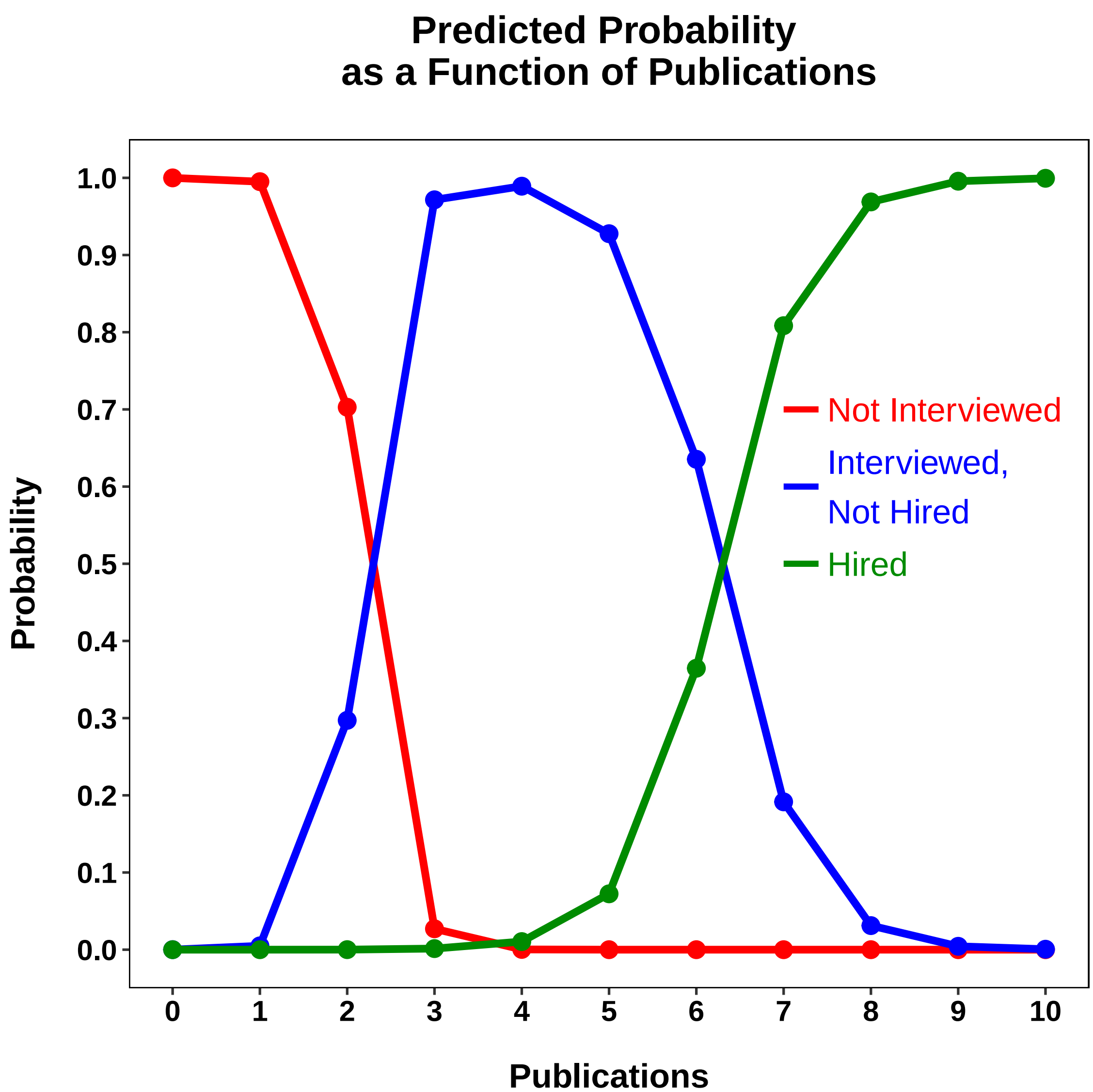
```r
ggplot(pred_data, aes(x = pubs, y = L_not_I)) + geom_line(size = 1.5,
    color = "red") + geom_point(size = 3, color = "red") + geom_line(aes(y = L_I),
    size = 1.5, color = "blue") + geom_point(aes(y = L_I), size = 3,
    color = "blue") + geom_line(aes(y = L_H), size = 1.5, color = "green4") +
    geom_point(aes(y = L_H), size = 3, color = "green4") + coord_cartesian(xlim = c(0,
    10), ylim = c(-50, 50)) + scale_x_continuous(breaks = c(seq(0,
    10, 1))) + scale_y_continuous(breaks = seq(-50, 50, 10)) + xlab("Publications") +
    ylab("Logit") + theme(text = element_text(size = 14, family = "sans",
    color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, angle = 0, face = "bold"), axis.title.x = element_text(margin = margin(15,
```

```
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + annotate("text", x = 0.6, y = 50,
    label = "Not Interviewed", color = "red", hjust = 0, size = 5) +
    annotate("text", x = 0.6, y = 45, label = "Interviewed, Not Hired",
        color = "blue", hjust = 0, size = 5) + annotate("text", x = 0.6,
    y = 40, label = "Hired", color = "green4", hjust = 0, size = 5) +
    annotate("segment", x = 0, xend = 0.5, y = 50, yend = 50, color = "red",
        size = 1.2, linetype = 1) + annotate("segment", x = 0, xend = 0.5,
    y = 45, yend = 45, color = "blue", size = 1.2, linetype = 1) +
    annotate("segment", x = 0, xend = 0.5, y = 40, yend = 40, color = "green4",
        size = 1.2, linetype = 1) + ggtitle("Predicted Logit \nas a Function of Publications")
```

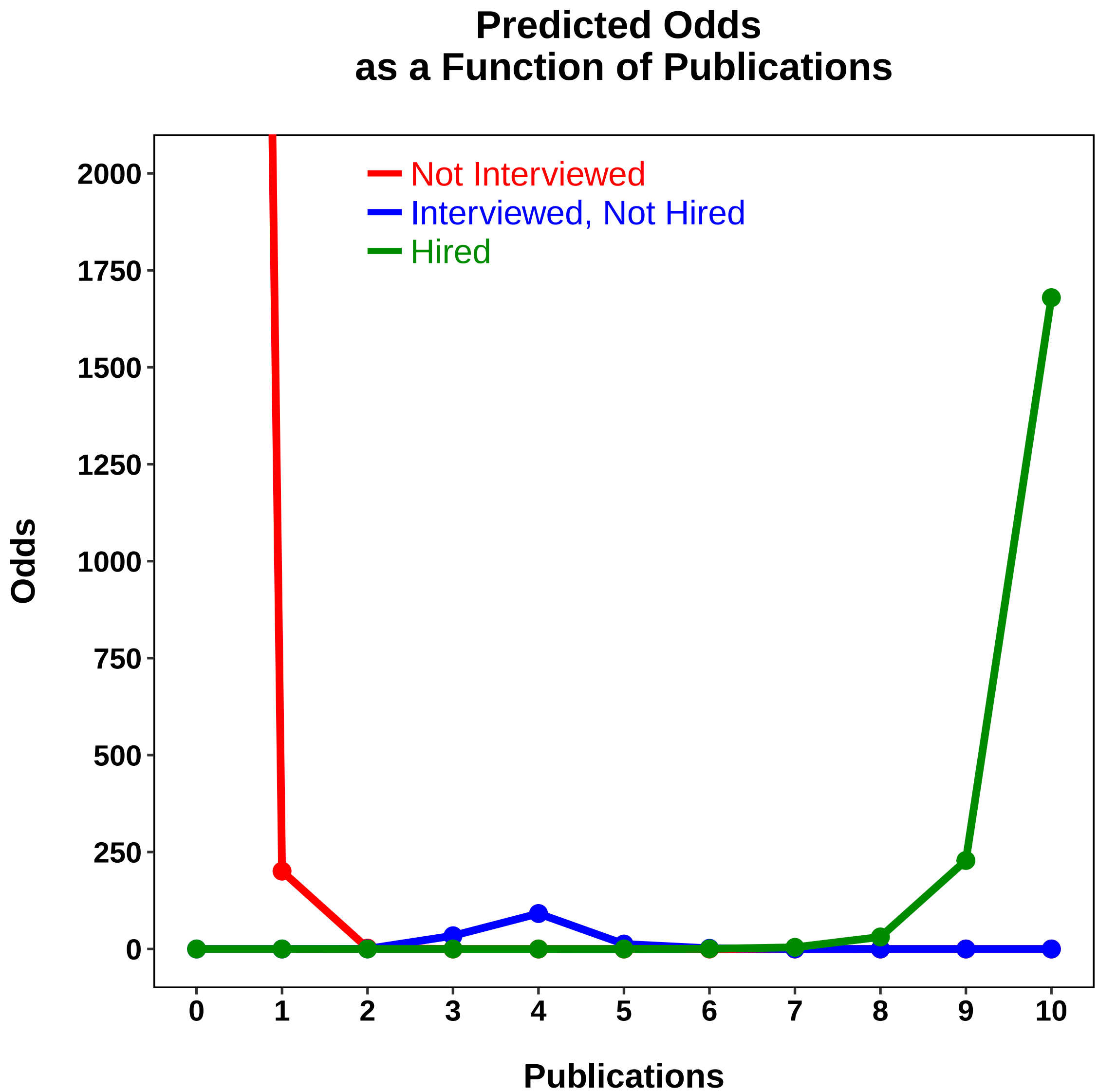



```
ggplot(pred_data, aes(x = pubs, y = p_not_I)) + geom_line(size = 1.5,
    color = "red") + geom_point(size = 3, color = "red") + geom_line(aes(y = p_I),
    size = 1.5, color = "blue") + geom_point(aes(y = p_I), size = 3,
    color = "blue") + geom_line(aes(y = p_H), size = 1.5, color = "green4") +
    geom_point(aes(y = p_H), size = 3, color = "green4") + coord_cartesian(xlim = c(0,
    10), ylim = c(0, 1)) + scale_x_continuous(breaks = c(seq(0, 10,
    1))) + scale_y_continuous(breaks = seq(0, 1, 0.1)) + xlab("Publications") +
    ylab("Probability") + theme(text = element_text(size = 14, family = "sans",
    color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, angle = 0, face = "bold"), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
```

```
    0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
    linetype = 1, color = "black"), panel.grid.major = element_blank(),
panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
legend.title = element_blank()) + annotate("text", x = 7.5, y = 0.7,
label = "Not Interviewed", color = "red", hjust = 0, size = 5) +
annotate("text", x = 7.5, y = 0.6, label = "Interviewed, \nNot Hired",
    color = "blue", hjust = 0, size = 5) + annotate("text", x = 7.5,
y = 0.5, label = "Hired", color = "green4", hjust = 0, size = 5) +
annotate("segment", x = 7, xend = 7.4, y = 0.7, yend = 0.7, color = "red",
    size = 1.2, linetype = 1) + annotate("segment", x = 7, xend = 7.4,
y = 0.6, yend = 0.6, color = "blue", size = 1.2, linetype = 1) +
annotate("segment", x = 7, xend = 7.4, y = 0.5, yend = 0.5, color = "green4",
    size = 1.2, linetype = 1) + ggtitle("Predicted Probability \nas a Function of Publications")
```

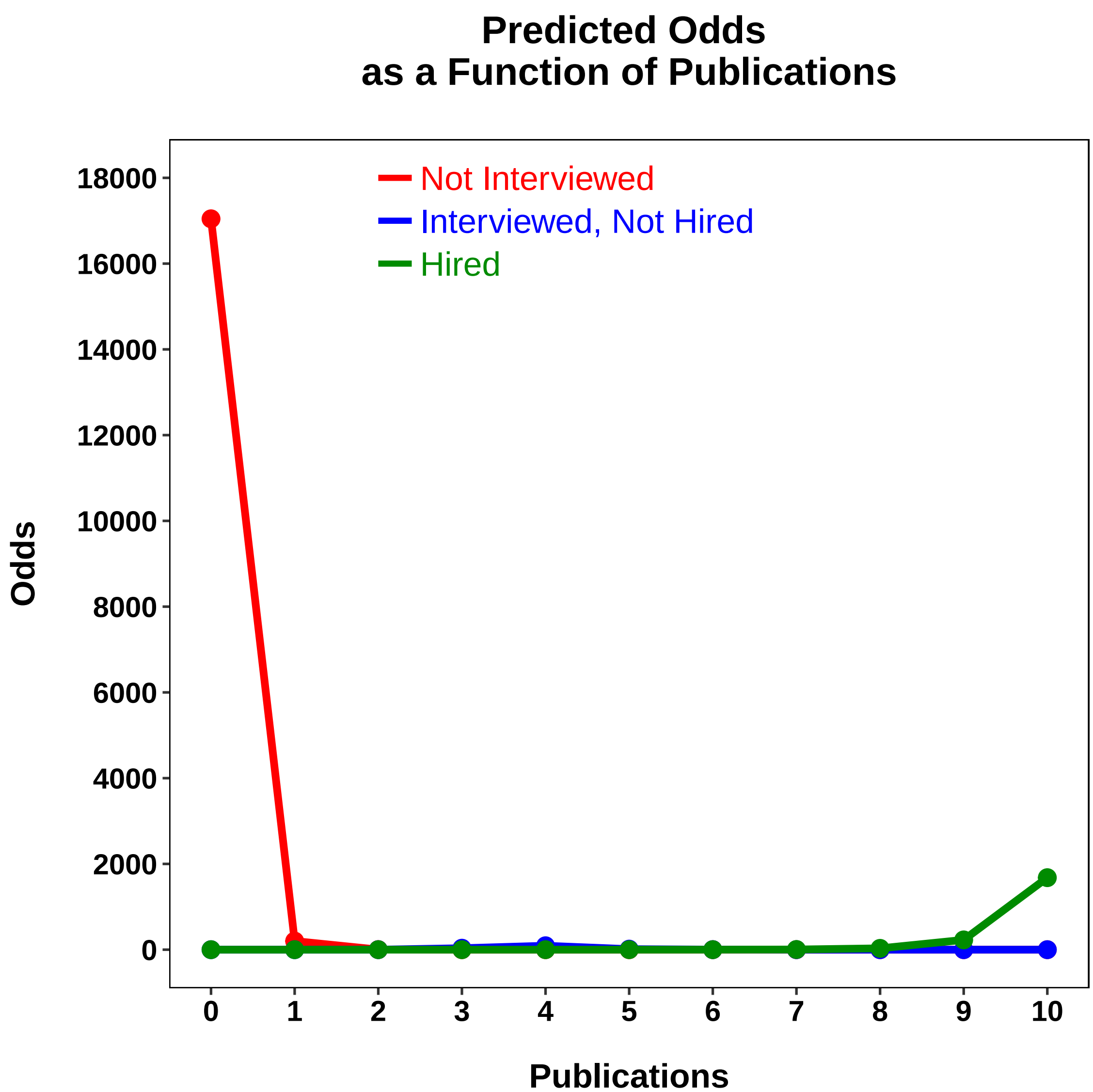


```
ggplot(pred_data, aes(x = pubs, y = O_not_I)) + geom_line(size = 1.5,
    color = "red") + geom_point(size = 3, color = "red") + geom_line(aes(y = O_I),
    size = 1.5, color = "blue") + geom_point(aes(y = O_I), size = 3,
    color = "blue") + geom_line(aes(y = O_H), size = 1.5, color = "green4") +
    geom_point(aes(y = O_H), size = 3, color = "green4") + coord_cartesian(xlim = c(0,
    10), ylim = c(0, 2000)) + scale_x_continuous(breaks = c(seq(0,
    10, 1))) + scale_y_continuous(breaks = seq(0, 2000, 250)) + xlab("Publications") +
    ylab("Odds") + theme(text = element_text(size = 14, family = "sans",
    color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, angle = 0, face = "bold"), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + annotate("text", x = 2.5, y = 2000,
    label = "Not Interviewed", color = "red", hjust = 0, size = 5) +
    annotate("text", x = 2.5, y = 1900, label = "Interviewed, Not Hired",
        color = "blue", hjust = 0, size = 5) + annotate("text", x = 2.5,
    y = 1800, label = "Hired", color = "green4", hjust = 0, size = 5) +
    annotate("segment", x = 2, xend = 2.4, y = 2000, yend = 2000,
        color = "red", size = 1.2, linetype = 1) + annotate("segment",
    x = 2, xend = 2.4, y = 1900, yend = 1900, color = "blue", size = 1.2,
    linetype = 1) + annotate("segment", x = 2, xend = 2.4, y = 1800,
    yend = 1800, color = "green4", size = 1.2, linetype = 1) + ggtitle("Predicted Odds \nas a Function o
```




```
ggplot(pred_data, aes(x = pubs, y = O_not_I)) + geom_line(size = 1.5,
    color = "red") + geom_point(size = 3, color = "red") + geom_line(aes(y = O_I),
    size = 1.5, color = "blue") + geom_point(aes(y = O_I), size = 3,
    color = "blue") + geom_line(aes(y = O_H), size = 1.5, color = "green4") +
    geom_point(aes(y = O_H), size = 3, color = "green4") + coord_cartesian(xlim = c(0,
    10), ylim = c(0, 18000)) + scale_x_continuous(breaks = c(seq(0,
    10, 1))) + scale_y_continuous(breaks = seq(0, 18000, 2000)) +
    xlab("Publications") + ylab("Odds") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, angle = 0, face = "bold"), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
```

```
    0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
    linetype = 1, color = "black"), panel.grid.major = element_blank(),
panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
legend.title = element_blank()) + annotate("text", x = 2.5, y = 18000,
label = "Not Interviewed", color = "red", hjust = 0, size = 5) +
annotate("text", x = 2.5, y = 17000, label = "Interviewed, Not Hired",
    color = "blue", hjust = 0, size = 5) + annotate("text", x = 2.5,
y = 16000, label = "Hired", color = "green4", hjust = 0, size = 5) +
annotate("segment", x = 2, xend = 2.4, y = 18000, yend = 18000,
    color = "red", size = 1.2, linetype = 1) + annotate("segment",
x = 2, xend = 2.4, y = 17000, yend = 17000, color = "blue", size = 1.2,
linetype = 1) + annotate("segment", x = 2, xend = 2.4, y = 16000,
yend = 16000, color = "green4", size = 1.2, linetype = 1) + ggtitle("Predicted Odds \nas a Function
```



6 Proportional Odds Logistic Regression

```
Job_POLR_1 <- polr(ordered ~ sex + gre + pubs + years, data = jobs,
    Hess = TRUE, method = "logistic")

## Warning:  glm.fit:  fitted probabilities numerically 0 or 1 occurred

summary(Job_POLR_1)

## Call:
## polr(formula = ordered ~ sex + gre + pubs + years, data = jobs,
##     Hess = TRUE, method = "logistic")
##
## Coefficients:
##          Value Std. Error t value
## sex   -0.3223   0.325079  -0.991
## gre   -0.0173   0.000684 -25.219
## pubs   2.2965   0.151110  15.198
## years -1.7506   0.126708 -13.816
##
## Intercepts:
##                              Value     Std. Error t value
## Not Interviewed|Interviewed   -28.760     0.012   -2356.515
## Interviewed|Hired             -19.089     0.632    -30.180
##
## Residual Deviance: 262.81
## AIC: 274.81

S_Job_POLR_1 <- summary(Job_POLR_1)
```

6.1 Classification

```
fit_prob <- as.data.frame(fitted(Job_POLR_1))
for (i in 1:500) {
    if ((fitted(Job_POLR_1)[i, 1] > fitted(Job_POLR_1)[i, 2]) & (fitted(Job_POLR_1)[i,
        1] > fitted(Job_POLR_1)[i, 3])) {
        jobs[i, "p_group"] <- names(fit_prob)[1]
    } else if ((fitted(Job_POLR_1)[i, 2] > fitted(Job_POLR_1)[i, 1]) &
        (fitted(Job_POLR_1)[i, 2] > fitted(Job_POLR_1)[i, 3])) {
        jobs[i, "p_group"] <- names(fit_prob)[2]
    } else if ((fitted(Job_POLR_1)[i, 3] > fitted(Job_POLR_1)[i, 1]) &
        (fitted(Job_POLR_1)[i, 3] > fitted(Job_POLR_1)[i, 2])) {
        jobs[i, "p_group"] <- names(fit_prob)[3]
    }
}

cross_class <- with(jobs, table(group_2, p_group))
cross_class

##                  p_group
## group_2           Hired Interviewed Not Interviewed
##   Not Interviewed     0           2             118
```

```
##   Hired               104          32               0
##   Interviewed          25         216               3
```

```r
cross_class/sum(cross_class)
```

```
##                   p_group
## group_2            Hired Interviewed Not Interviewed
##   Not Interviewed 0.000       0.004           0.236
##   Hired           0.208       0.064           0.000
##   Interviewed     0.050       0.432           0.006
```

```r
correct_class <- 0
for (i in 1:3) {
    for (j in 1:3) {
        if (row.names(cross_class)[i] == colnames(cross_class)[j]) {
            correct_class <- correct_class + cross_class[i, j]
        }
    }
}
correct_class/sum(cross_class)
```

```
## [1] 0.876
```

```r
summary(cross_class)
```

```
## Number of cases in table: 500
## Number of factors: 2
## Test for independence of all factors:
##  Chisq = 695, df = 4, p-value = 4e-149
```

	Classification		
	Not Interviewed	***Interviewed***	***Hired***
Predict: Not Interviewed	***118***	***3***	***0***
Predict: Interviewed	***2***	***216***	***32***
Predict: Hired	***0***	***25***	***104***

	Classification		
	Not Interviewed	***Interviewed***	***Hired***
Predict: Not Interviewed	***0.234***	***0.006***	***0***
Predict: Interviewed	***0.006***	***0.432***	***0.062***
Predict: Hired	***0***	***0.05***	***0.21***

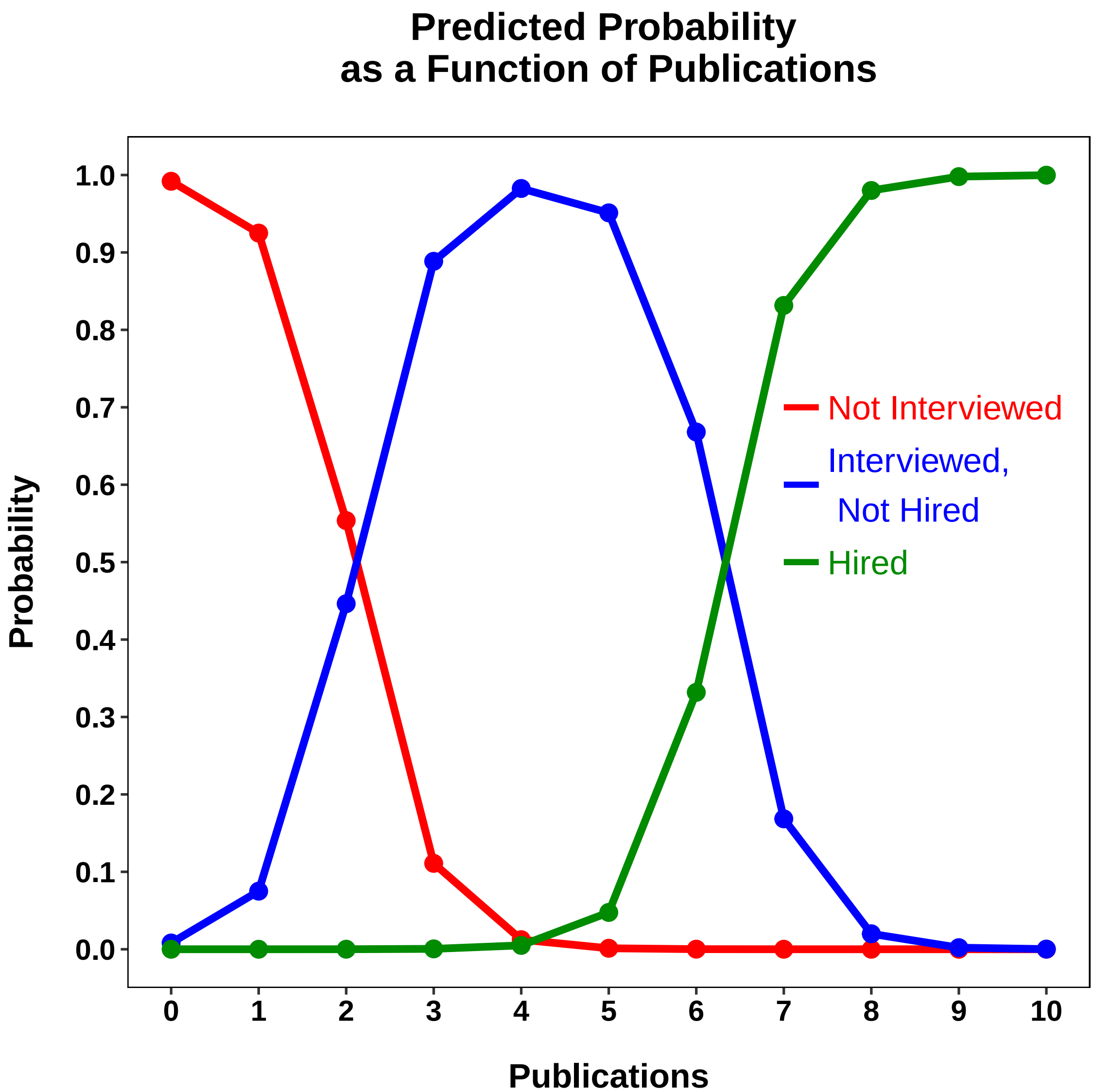
6.2 Predicted Outcomes

```r
predict_data <- with(jobs, data.frame(years = mean(jobs$years), gre = mean(jobs$gre),
    sex = mean(jobs$sex), pubs = seq(0, 10, 1)))
plot_data <- predict(Job_POLR_1, predict_data, type = "probs")
plot_data <- as.data.frame(plot_data)
```

```r
plot_data <- cbind(seq(0, 10, 1), plot_data)
names(plot_data) <- c("pubs", "p_not_I", "p_I", "p_H")
```

6.3 Graphs: Publications

```r
ggplot(plot_data, aes(x = pubs, y = p_not_I)) + geom_line(size = 1.5,
    color = "red") + geom_point(size = 3, color = "red") + geom_line(aes(y = p_I),
    size = 1.5, color = "blue") + geom_point(aes(y = p_I), size = 3,
    color = "blue") + geom_line(aes(y = p_H), size = 1.5, color = "green4") +
    geom_point(aes(y = p_H), size = 3, color = "green4") + coord_cartesian(xlim = c(0,
    10), ylim = c(0, 1)) + scale_x_continuous(breaks = c(seq(0, 10,
    1))) + scale_y_continuous(breaks = seq(0, 1, 0.1)) + xlab("Publications") +
    ylab("Probability") + theme(text = element_text(size = 14, family = "sans",
    color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, angle = 0, face = "bold"), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + annotate("text", x = 7.5, y = 0.7,
    label = "Not Interviewed", color = "red", hjust = 0, size = 5) +
    annotate("text", x = 7.5, y = 0.6, label = "Interviewed, \n Not Hired",
        color = "blue", hjust = 0, size = 5) + annotate("text", x = 7.5,
    y = 0.5, label = "Hired", color = "green4", hjust = 0, size = 5) +
    annotate("segment", x = 7, xend = 7.4, y = 0.7, yend = 0.7, color = "red",
        size = 1.2, linetype = 1) + annotate("segment", x = 7, xend = 7.4,
    y = 0.6, yend = 0.6, color = "blue", size = 1.2, linetype = 1) +
    annotate("segment", x = 7, xend = 7.4, y = 0.5, yend = 0.5, color = "green4",
        size = 1.2, linetype = 1) + ggtitle("Predicted Probability \nas a Function of Publications")
```



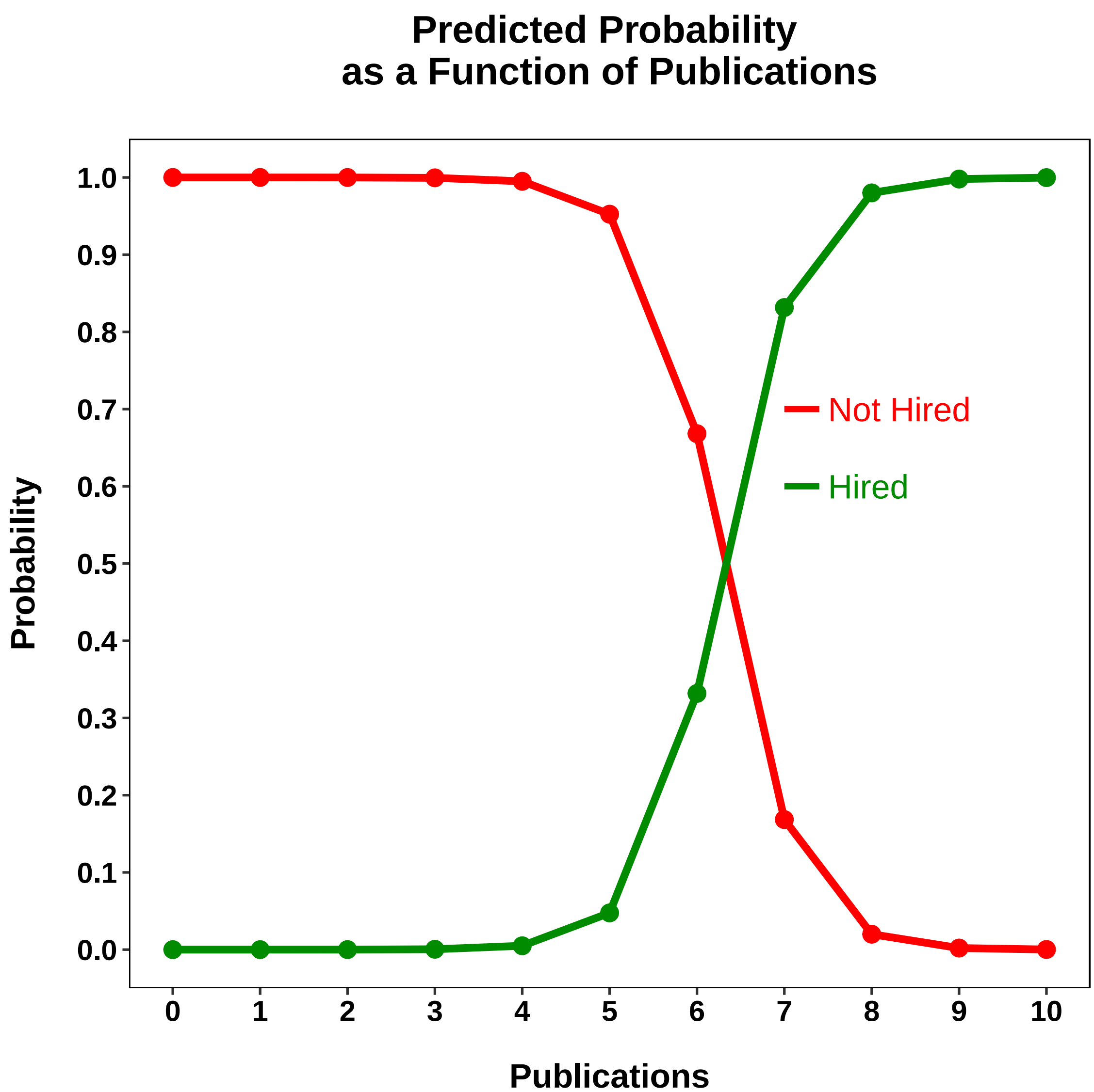
6.4 Predicted Outcomes

```
plot_data$p_not_H <- plot_data$p_not_I + plot_data$p_I
```

6.5 Graphs: Publications

```
ggplot(plot_data, aes(x = pubs, y = p_not_H)) + geom_line(size = 1.5,
    color = "red") + geom_point(size = 3, color = "red") + geom_line(aes(y = p_H),
    size = 1.5, color = "green4") + geom_point(aes(y = p_H), size = 3,
    color = "green4") + coord_cartesian(xlim = c(0, 10), ylim = c(0,
    1)) + scale_x_continuous(breaks = c(seq(0, 10, 1))) + scale_y_continuous(breaks = seq(0,
```

```r
    1, 0.1)) + xlab("Publications") + ylab("Probability") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, angle = 0, face = "bold"), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + annotate("text", x = 7.5, y = 0.7,
    label = "Not Hired", color = "red", hjust = 0, size = 5) + annotate("text",
    x = 7.5, y = 0.6, label = "Hired", color = "green4", hjust = 0,
    size = 5) + annotate("segment", x = 7, xend = 7.4, y = 0.7, yend = 0.7,
    color = "red", size = 1.2, linetype = 1) + annotate("segment",
    x = 7, xend = 7.4, y = 0.6, yend = 0.6, color = "green4", size = 1.2,
    linetype = 1) + ggtitle("Predicted Probability \nas a Function of Publications")
```



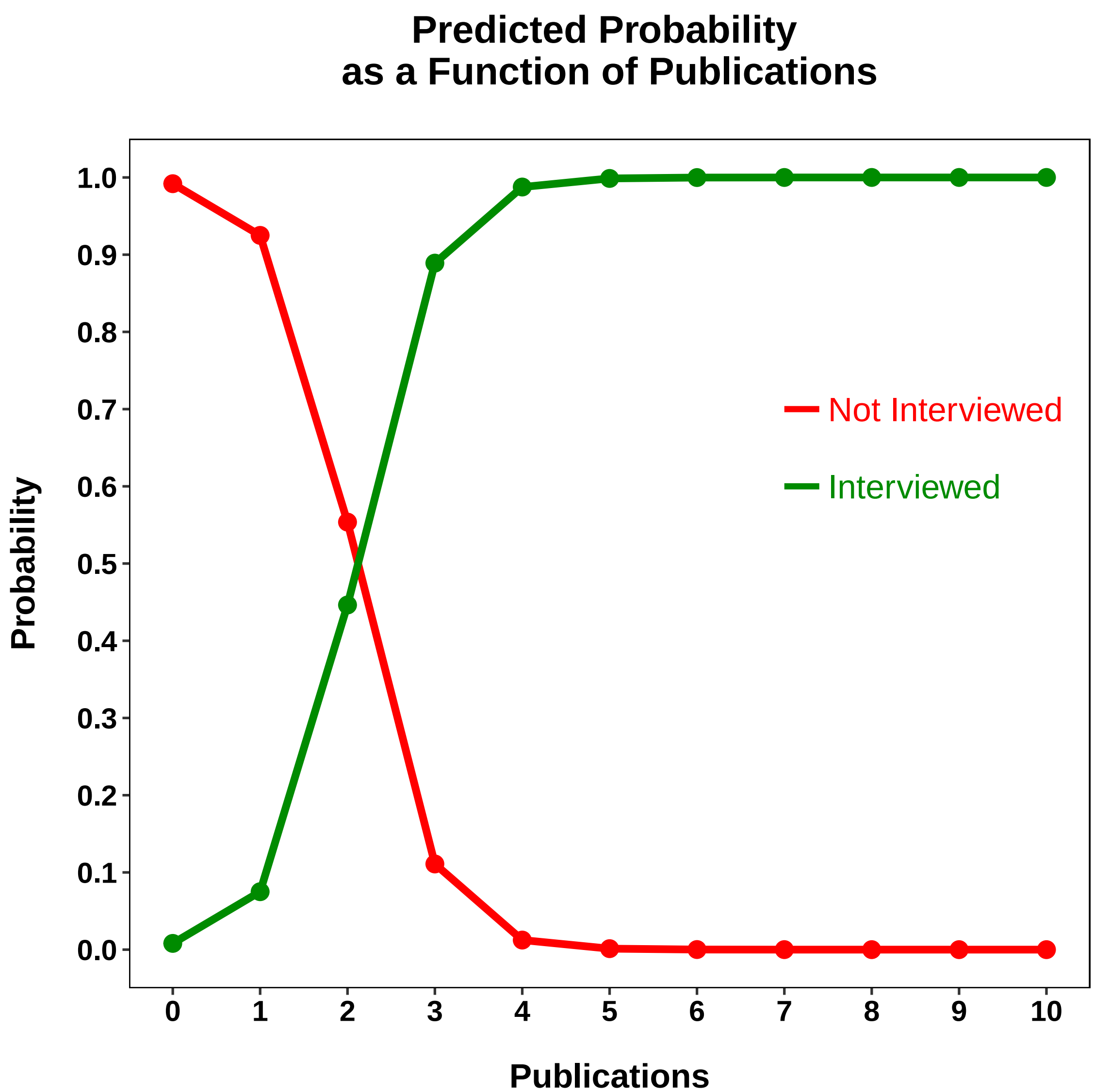
6.6 Predicted Outcomes

```
plot_data$p_all_I <- plot_data$p_I + plot_data$p_H
```

6.7 Graphs: Publications

```
ggplot(plot_data, aes(x = pubs, y = p_not_I)) + geom_line(size = 1.5,
    color = "red") + geom_point(size = 3, color = "red") + geom_line(aes(y = p_all_I),
    size = 1.5, color = "green4") + geom_point(aes(y = p_all_I), size = 3,
    color = "green4") + coord_cartesian(xlim = c(0, 10), ylim = c(0,
    1)) + scale_x_continuous(breaks = c(seq(0, 10, 1))) + scale_y_continuous(breaks = seq(0,
```

```
    1, 0.1)) + xlab("Publications") + ylab("Probability") + theme(text = element_text(size = 14,
    family = "sans", color = "black", face = "bold"), axis.text.y = element_text(colour = "black",
    size = 12, face = "bold"), axis.text.x = element_text(colour = "black",
    size = 12, angle = 0, face = "bold"), axis.title.x = element_text(margin = margin(15,
    0, 0, 0), size = 14), axis.title.y = element_text(margin = margin(0,
    15, 0, 0), size = 14), axis.line.x = element_blank(), axis.line.y = element_blank(),
    plot.title = element_text(size = 16, face = "bold", margin = margin(0,
        0, 20, 0), hjust = 0.5), panel.background = element_rect(fill = "white",
        linetype = 1, color = "black"), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), plot.background = element_rect(fill = "white"),
    plot.margin = unit(c(1, 1, 1, 1), "cm"), legend.position = "bottom",
    legend.title = element_blank()) + annotate("text", x = 7.5, y = 0.7,
    label = "Not Interviewed", color = "red", hjust = 0, size = 5) +
    annotate("text", x = 7.5, y = 0.6, label = "Interviewed", color = "green4",
        hjust = 0, size = 5) + annotate("segment", x = 7, xend = 7.4,
    y = 0.7, yend = 0.7, color = "red", size = 1.2, linetype = 1) +
    annotate("segment", x = 7, xend = 7.4, y = 0.6, yend = 0.6, color = "green4",
        size = 1.2, linetype = 1) + ggtitle("Predicted Probability \nas a Function of Publications")
```




```
Sys.time() - how_long

## Time difference of 22.2 secs
```