Stats-Thesis2.Rmd

Jackson Bandow

3/9/2021

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

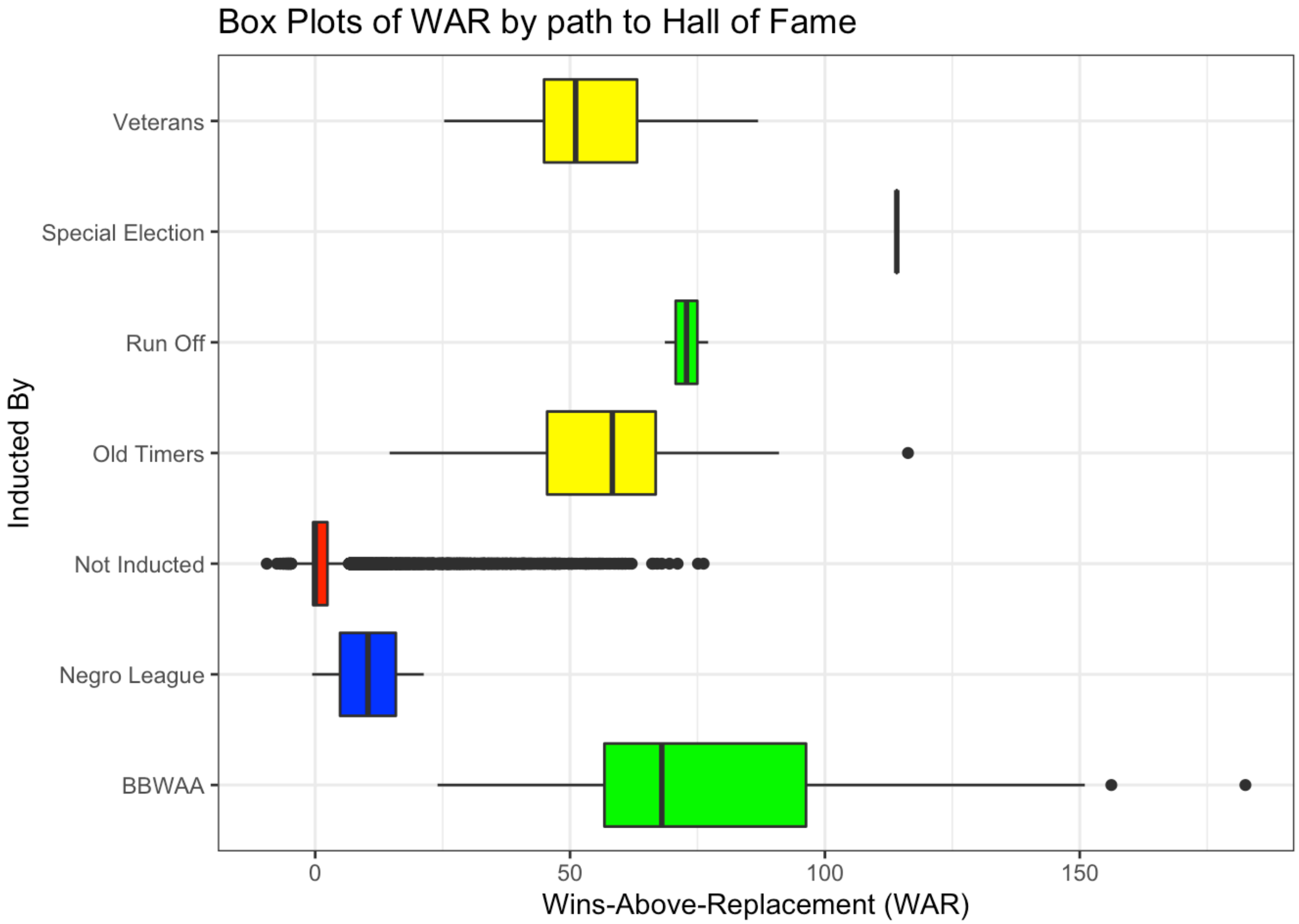
```
library(Lahman)
library(stringr)
```

```
baseball <- fread("baseball.csv")
```

```
baseball[baseball$votedBy == "", "votedBy"] <- "Not Inducted"
baseball$votedBy2 <- baseball$votedBy
baseball[baseball$votedBy2 == "Old Timers", "votedBy2"] <- "Veterans"
baseball[baseball$votedBy2 == "Run Off", "votedBy2"] <- "BBWAA"
baseball[baseball$votedBy2 == "Special Election", "votedBy2"] <- "BBWAA"
baseball$preHOFEra <- baseball$finalGame < as.Date("1922-01-01")
#Create a training and testing set
# By date
# bb_train <- baseball %>% filter(finalGame < as.Date("1991-01-01"))
# bb_test <- baseball %>% filter(finalGame > as.Date("1991-01-01"))

# Random
baseball_res <- baseball %>% filter(finalGame < as.Date("2004-01-01"))
indeces = sort(sample(nrow(baseball_res), nrow(baseball_res)*.7))
bb_train <- baseball_res[indeces,]
bb_test <- baseball_res[-indeces,]
```

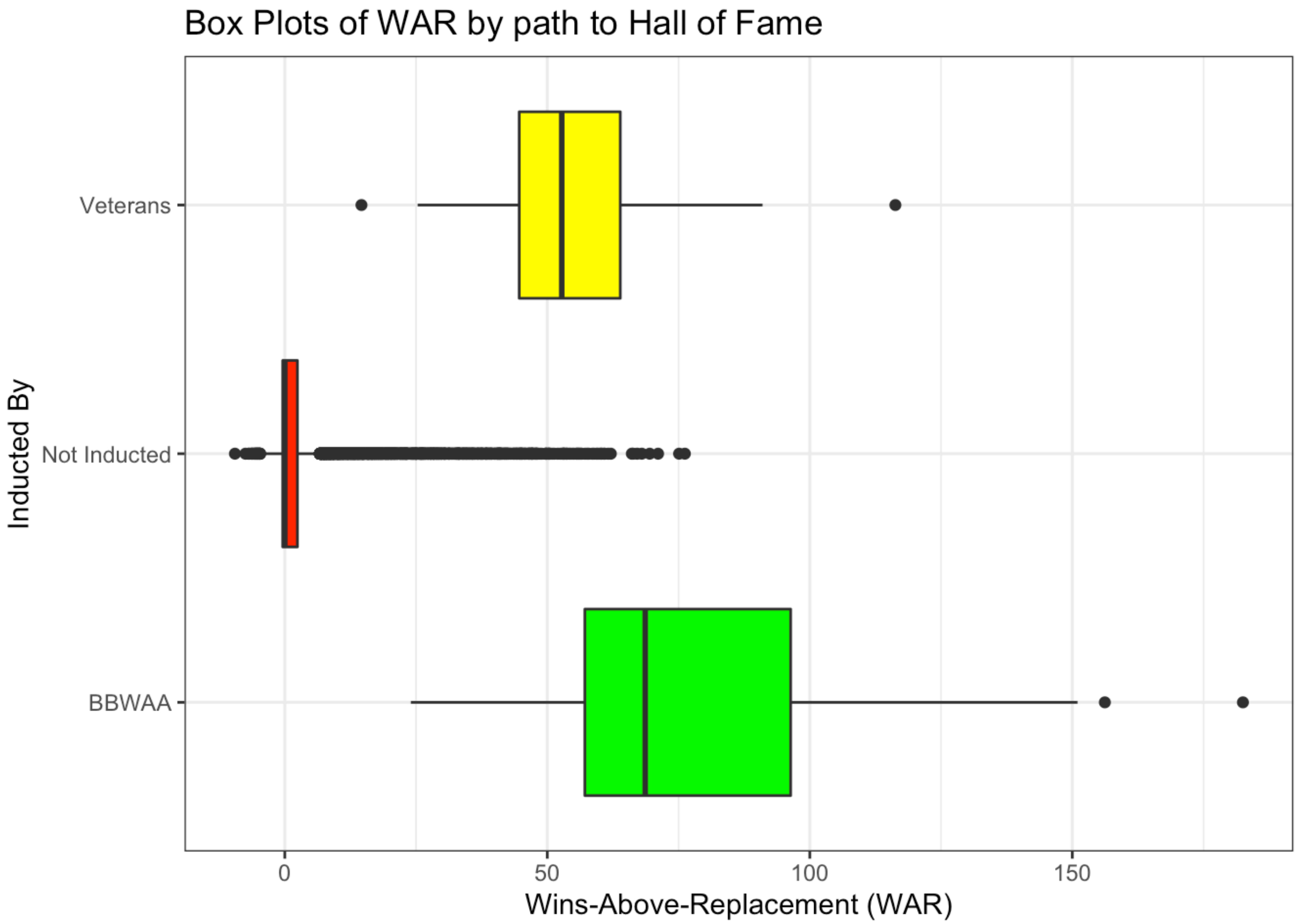
I've used the era from 1871 through 1990 as my training dataset because it is approximately 80% of the data available. Also, I avoid most players that are still in consideration for the Hall of Fame, so I am unlikely to falsely assume that they won't be inducted becuase they have not yet been.

```
ggplot(bb_train, aes(rWAR, votedBy)) + geom_boxplot(fill = c("green", "blue", "red",
"yellow","green","green","yellow")) + theme_bw() + labs(title = "Box Plots of WAR by
path to Hall of Fame", y = "Inducted By", x = "Wins-Above-Replacement (WAR)")
```



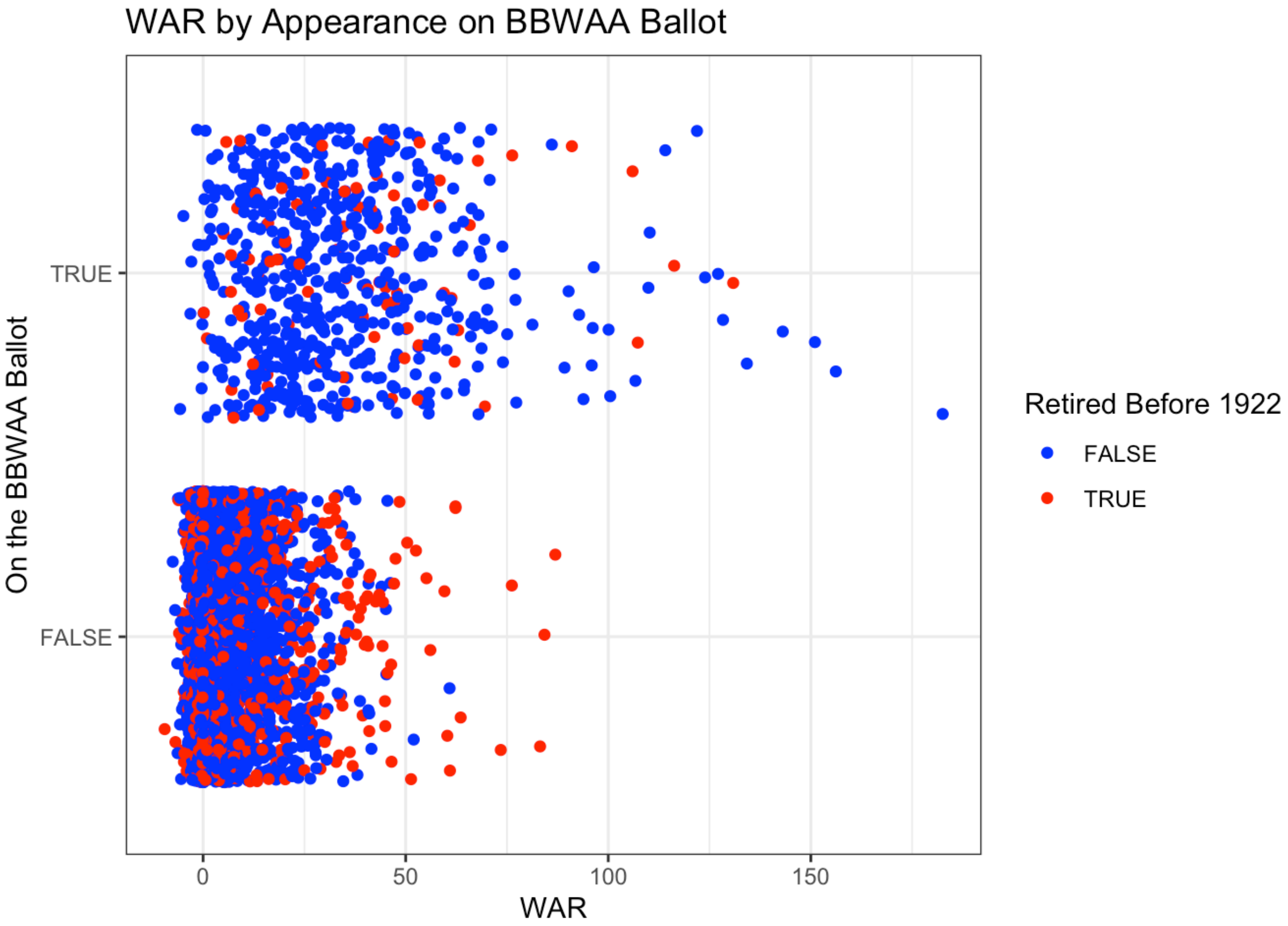
Although there are a lot of committees for induction to the Hall of Fame, they can be condensed. The BBWAA is the organization of sportswriters that vote in players that have retired 5-15 years prior. On two occasions (Lou Gehrig and Roberto Clemente), there have been a special election, which can be included under the BBWAA umbrella. Additionally, there is occasionally a run off where only the top voting result is inducted. This process can also be included under BBWAA. Generally after a player is passed on by the BBWAA (this rule has changed historically), they are still eligible for election under the Veterans committee. The first election for the Hall of Fame was in 1936, so a subcommittee of the Veterans committee, the Old Timers committee, was formed to fill in the gaps of older players. These categories can also be combined. Finally, although the Negro League is also a subcommittee of Veterans, I will consider them seperately. Because most of them did not play in the MLB, most of them do not have a WAR statistic. Those that joined the MLB during the integration era have a WAR, but joined the league part-way through their career, so their WAR only reflects a fraction of their career. For those reasons, I will remove the Negro League inductees for now.

```
# Excludes Negro League players
bb_train <- bb_train %>% filter(!is.na(rWAR) & votedBy != "Negro League")
bb_test <- bb_test %>% filter(!is.na(rWAR) & votedBy != "Negro League")
ggplot(bb_train, aes(rWAR, votedBy2)) + geom_boxplot(fill = c("green", "red", "yellow
")) + theme_bw() + labs(title = "Box Plots of WAR by path to Hall of Fame", y = "Indu
cted By", x = "Wins-Above-Replacement (WAR)")
```



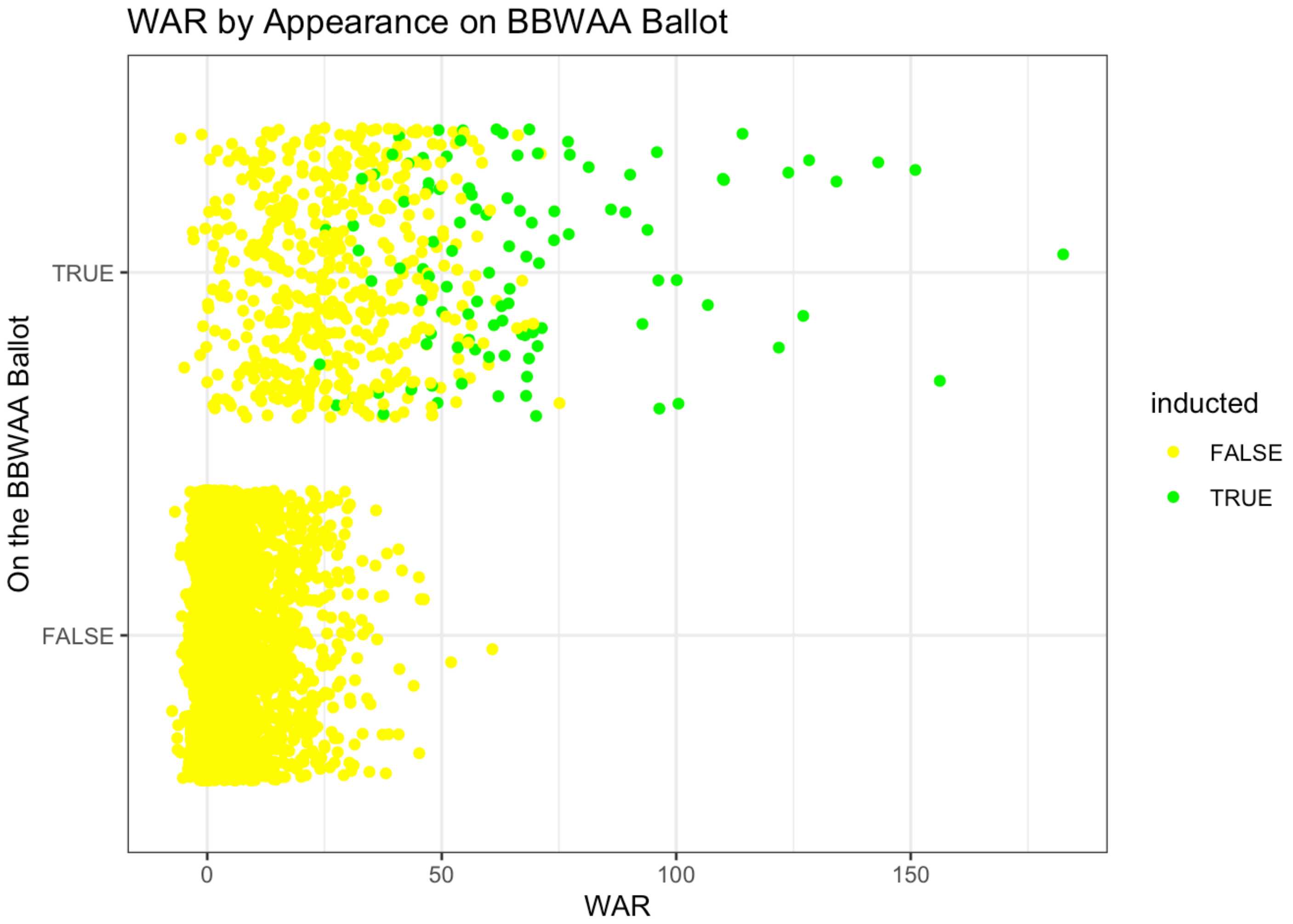
WAR seems to generally be a good indicator of Hall of Fame status, but there is clearly more to the story. There are definitly some outliers on either side that need some explaining. Perhaps they deserve to be in the Hall of Fame or perhaps there is more to a player's Hall of Fame candidacy than their value, described by WAR. The first step to induction is to be nominated for the BBWAA Hall of Fame ballot (this is determined by a subcomittee of the association nowadays).

```
ggplot(bb_train, aes(rWAR, onBallot, color = preHOFEra)) + geom_jitter() + theme_bw()
+ labs(title = "WAR by Appearance on BBWAA Ballot", x = "WAR", y = "On the BBWAA Ball
ot", color = "Retired Before 1922") + scale_color_manual(values = c("blue", "red"))
```



The blue points represent players that retired before the 1922 season. These players would be ineligible for the first HOF ballot in 1936 under today’s standards. Some players from this era were considered, as the rules were not solidifed until much later. However, the Old Timers committee was created to address the issue of underrepresentation amoung the older era, so that would explain why the majority of the higher outliers for WAR that were not included on any BBWAA ballot are from this older era.

```
bb_train.HOFEra <- bb_train %>% filter(!preHOFEra)
ggplot(bb_train.HOFEra, aes(rWAR, onBallot, color = inducted)) + geom_jitter() + them
e_bw() + labs(title = "WAR by Appearance on BBWAA Ballot", x = "WAR", y = "On the BBW
AA Ballot") + scale_color_manual(values = c("yellow", "green"))
```



With the pre Hall of Fame players eliminated, the plot is slighlty improved. There is a pretty clear difference between those eventually inducted in the Hall of Fame and those that were not included on the ballot. However, there are still a significant amount of mediocre to just bad players, considered by WAR that were included on the ballot. Why was Tommy Thevenow included with a career 5.7 WAR? Let’s see if a logistic regression can pick up on interesting trends.

```
mod1 <- glm(onBallot ~ rWAR, data = bb_train.HOFEra, family = "binomial")
summary(mod1)
```

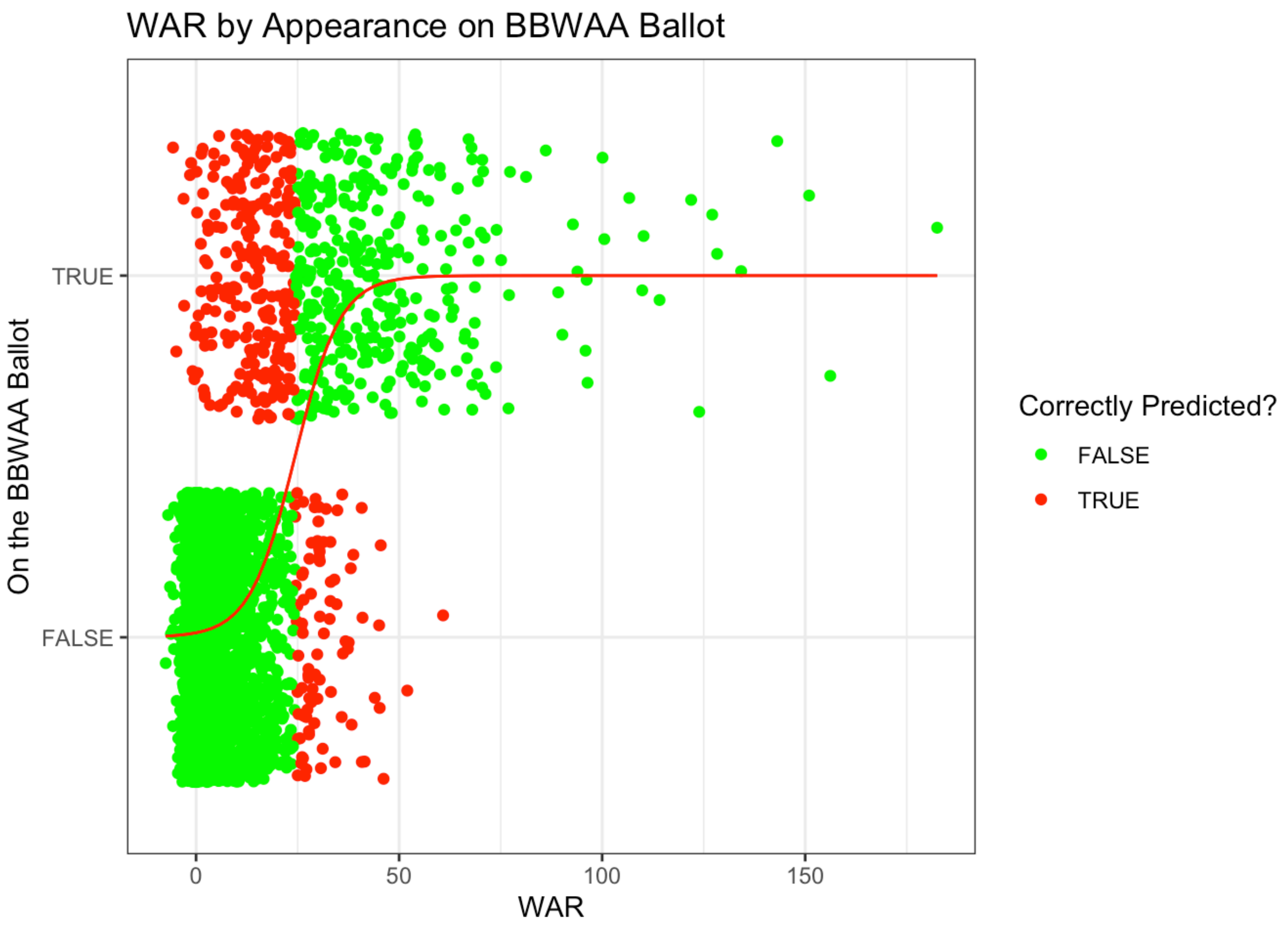
```
## 
## Call:
## glm(formula = onBallot ~ rWAR, family = "binomial", data = bb_train.HOFEra)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -3.6143  -0.1799  -0.1588  -0.1505   3.2843  
## 
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)    
## (Intercept) -4.367163   0.102536  -42.59   <2e-16 ***
## rWAR         0.179232   0.005687   31.52   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 4205.5  on 7166  degrees of freedom
## Residual deviance: 1845.7  on 7165  degrees of freedom
## AIC: 1849.7
## 
## Number of Fisher Scoring iterations: 7
```

```
b0 <- mod1$coefficients[1]
b1 <- mod1$coefficients[2]

get_prob_mod1 <- function(input){
  prob_new <- (exp(b0 + b1 * input)) / (1 + exp(b0 + b1 * input))
  names(prob_new) <- NULL
  return(prob_new)
}

plot_prob_mod1 <- function(input) {
  get_prob_mod1(input) + 1
}

(misclassification_rate <- sum(round(get_prob_mod1(bb_train.HOFEra$rWAR), 0) != bb_train.HOFEra$onBallot) / dim(bb_train.HOFEra)[1])
```

```
## [1] 0.04702107
```

```
bb_train.HOFEra$pred1_cor <- round(get_prob_mod1(bb_train.HOFEra$rWAR), 0) != bb_train.HOFEra$onBallot

ggplot(bb_train.HOFEra, aes(rWAR, onBallot, color = bb_train.HOFEra$pred1_cor)) + geom_jitter() + theme_bw() + labs(title = "WAR by Appearance on BBWAA Ballot", x = "WAR", y = "On the BBWAA Ballot", color = "Correctly Predicted?") + stat_function(fun = plot_prob_mod1, color = "red") + scale_color_manual(values = c("green", "red"))
```

```
## Warning: Multiple drawing groups in `geom_function()`. Did you use the correct
## `group`, `colour`, or `fill` aesthetics?
```



The logistic regression is a decent predictor of ballot inclusion, but it could be improved. Let’s add in features and observe changes in predictive ability.

```
# Need to figure out how to deal with missing data. All WAR data is present but need to find a way to fill in data for those that do not have any. Discuss more on Friday

# mod2 <- glm(onBallot ~ rWAR.B + AB + H + HR + BA + R + RBI + SB + OBP + SLG + OPS + rWAR.P + W + L + ERA + G + GS + SV + IP + SO + WHIP, data = bb_train.HOFEra, family = "binomial")
# summary(mod2)
# betas.2 <- mod2$coefficients
#
# get_prob_mod2 <- function(input){
#   prob_new <- (exp(betas.2[1] + betas.2[2:length(betas.2)] * input)) / (1 + exp(betas.2[1] + betas.2[2:length(betas.2)] * input))
#   names(prob_new) <- NULL
#   return(prob_new)
# }
#
# plot_prob_mod2 <- function(input) {
#   get_prob_mod1(input) + 1
# }
#
# (misclassification_rate <- sum(round(get_prob_mod2(bb_train.HOFEra[,c("rWAR.B", "AB", "H", "HR", "BA", "R", "RBI", "SB", "OBP", "SLG", "OPS", "rWAR.P", "W", "L", "ERA", "G", "GS", "SV", "IP", "SO", "WHIP")]), 0) != bb_train.HOFEra$onBallot) / dim(bb_train.HOFEra)[1])
#
# bb_train.HOFEra$pred1_cor <- round(get_prob_mod1(bb_train.HOFEra$rWAR), 0) != bb_train.HOFEra$onBallot
#
# ggplot(bb_train.HOFEra, aes(rWAR, onBallot, color = bb_train.HOFEra$pred1_cor)) + geom_jitter() + theme_bw() + labs(title = "WAR by Appearance on BBWAA Ballot", x = "WAR", y = "On the BBWAA Ballot", color = "Correctly Predicted?") + stat_function(fun = plot_prob_mod1, color = "red") + scale_color_manual(values = c("green", "red"))
```

March 19:

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(randomForest)
```

```
## randomForest 4.6-14
```

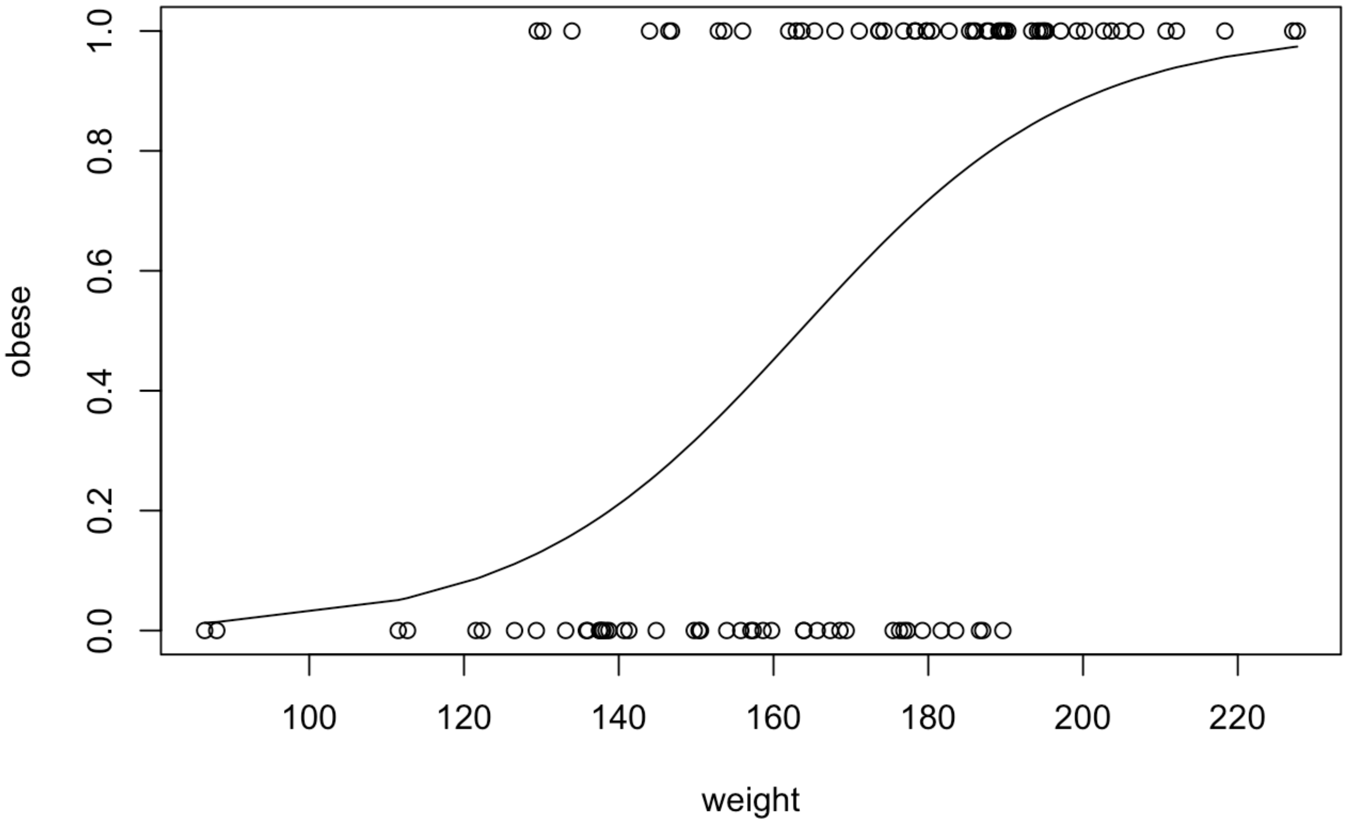
```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

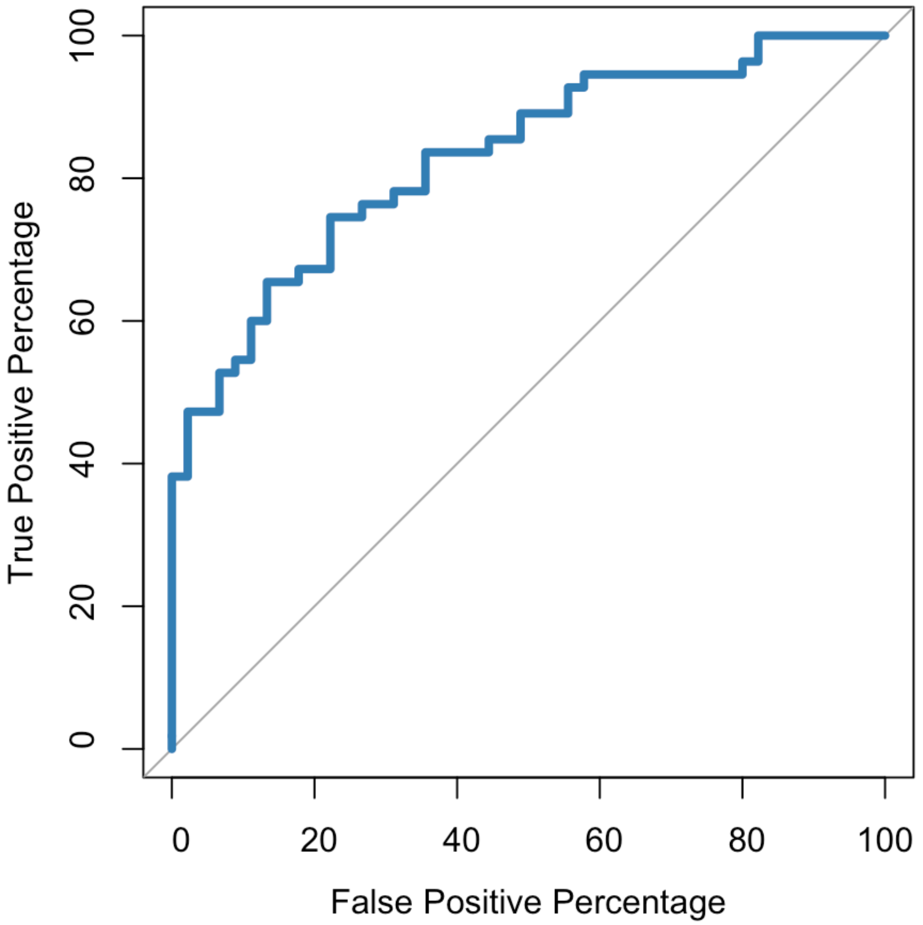
```
set.seed(420)

num.samples <- 100

weight <- sort(rnorm(n=num.samples, mean = 172, sd= 29))

obese <- ifelse(test = (runif(n=num.samples) < (rank(weight)/100)),yes = 1,no = 0)
plot(x=weight, y = obese)
glm.fit <- glm(obese ~ weight, family = binomial)
lines(weight, glm.fit$fitted.values)
```



```
par(pty = "s")
roc(obese, glm.fit$fitted.values, plot = TRUE, legacy.axes = TRUE, percent = TRUE, xl
ab = "False Positive Percentage", ylab = "True Positive Percentage", col = "#377eb8",
lwd = 4)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```



```
##
## Call:
## roc.default(response = obese, predictor = glm.fit$fitted.values,     percent = TRU
E, plot = TRUE, legacy.axes = TRUE, xlab = "False Positive Percentage",     ylab = "T
rue Positive Percentage", col = "#377eb8", lwd = 4)
##
## Data: glm.fit$fitted.values in 45 controls (obese 0) < 55 cases (obese 1).
## Area under the curve: 82.91%
```

```
roc.info <- roc(obese, glm.fit$fitted.values, legacy.axes = TRUE)
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

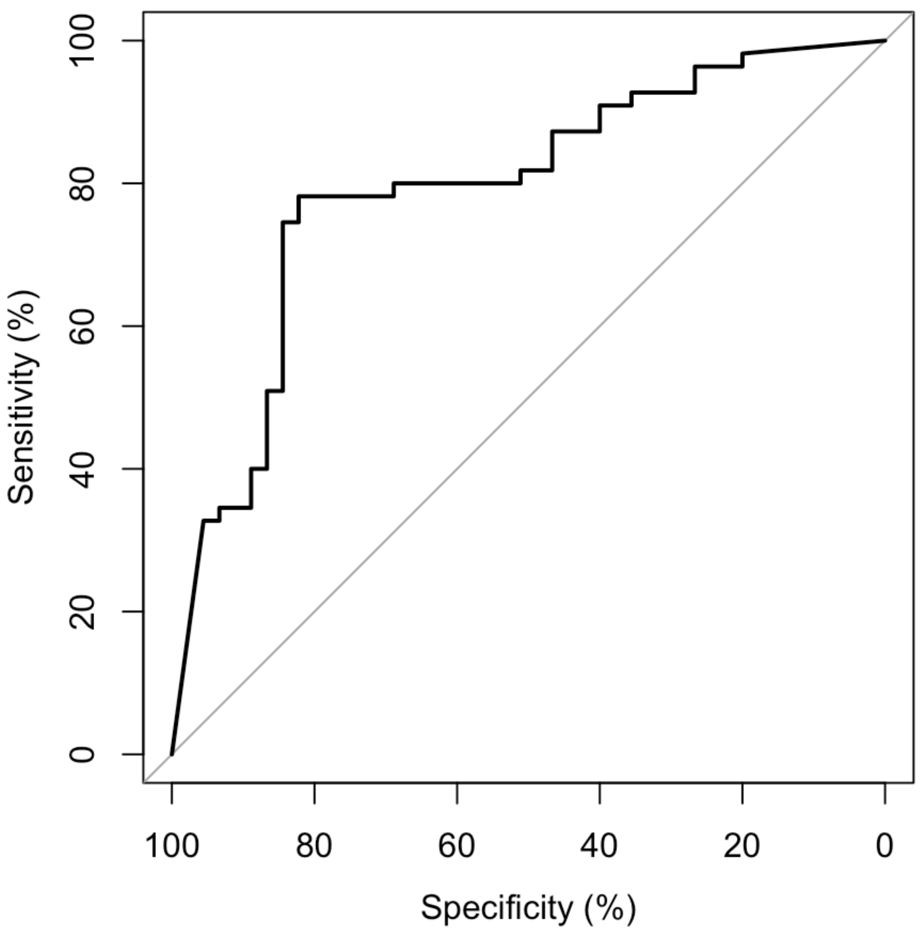
```
roc.df <- data.frame(tpp = roc.info$sensitivities*100, fpp = (1 - roc.info$specificit
ies)*100, thresholds = roc.info$thresholds)
roc.df[roc.df$tpp > 60 & roc.df$tpp < 80,]
```

```
##         tpp      fpp thresholds
## 42 78.18182 35.55556  0.5049310
## 43 78.18182 33.33333  0.5067116
## 44 78.18182 31.11111  0.5166680
## 45 76.36364 31.11111  0.5287933
## 46 76.36364 28.88889  0.5429351
## 47 76.36364 26.66667  0.5589494
## 48 74.54545 26.66667  0.5676342
## 49 74.54545 24.44444  0.5776086
## 50 74.54545 22.22222  0.5946054
## 51 72.72727 22.22222  0.6227449
## 52 70.90909 22.22222  0.6398136
## 53 69.09091 22.22222  0.6441654
## 54 67.27273 22.22222  0.6556705
## 55 67.27273 20.00000  0.6683618
## 56 67.27273 17.77778  0.6767661
## 57 65.45455 17.77778  0.6802060
## 58 65.45455 15.55556  0.6831936
## 59 65.45455 13.33333  0.6917225
## 60 63.63636 13.33333  0.6975300
## 61 61.81818 13.33333  0.6982807
```

```
rf.model <- randomForest(factor(obese) ~ weight)
roc(obese, rf.model$votes[,1], percent = TRUE, plot = TRUE)
```

```
## Setting levels: control = 0, case = 1
```

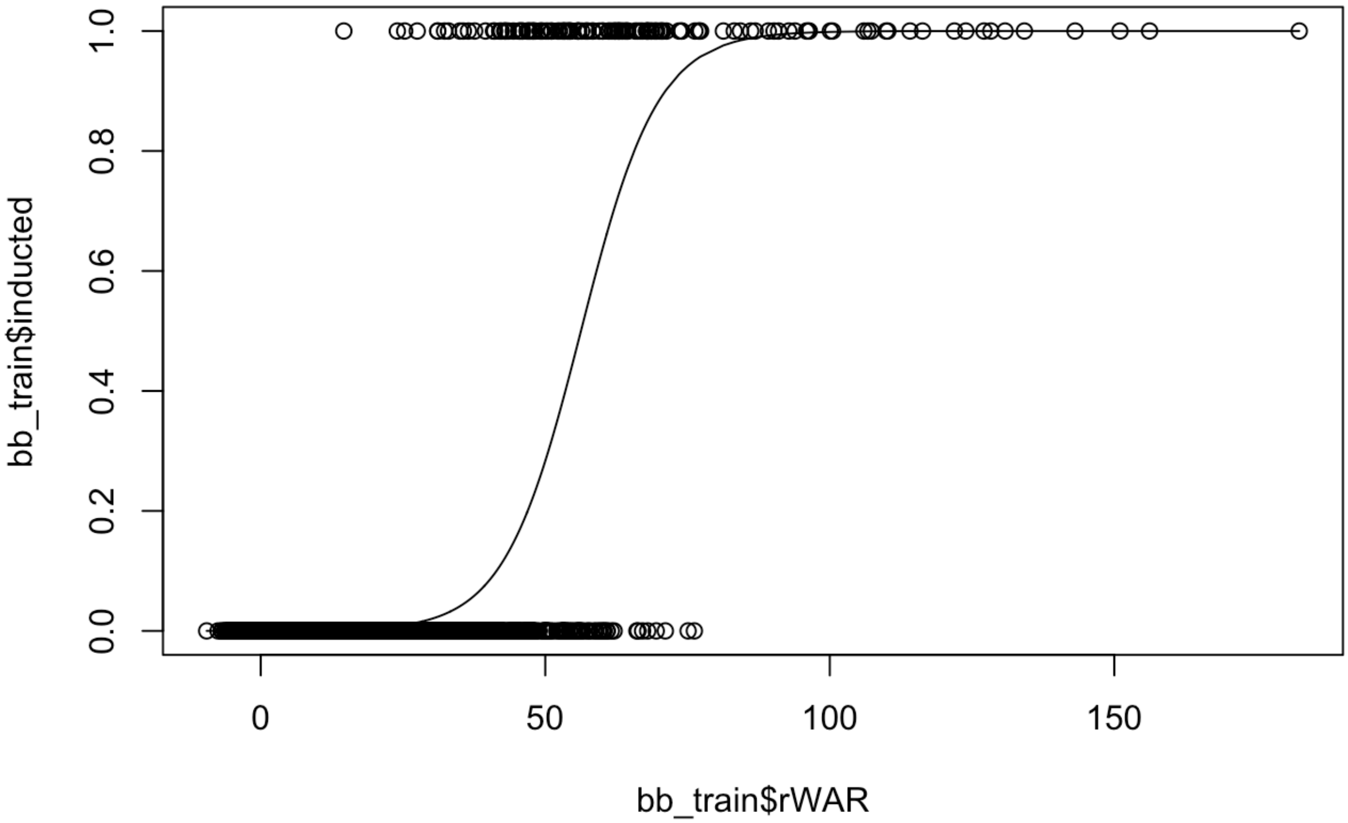
```
## Setting direction: controls > cases
```



```
##
## Call:
## roc.default(response = obese, predictor = rf.model$votes[, 1],     percent = TRUE,
plot = TRUE)
##
## Data: rf.model$votes[, 1] in 45 controls (obese 0) > 55 cases (obese 1).
## Area under the curve: 79.29%
```

Apply ROC Examples

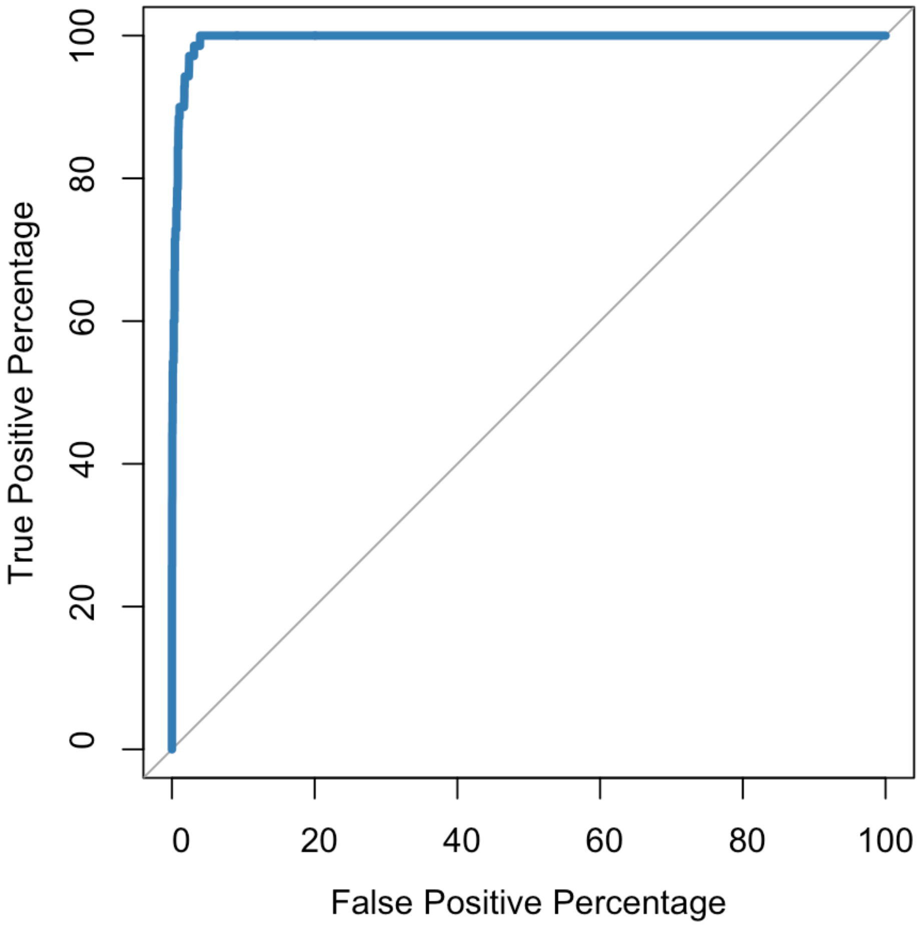
```
bb_train <- bb_train[order(bb_train$rWAR),]
plot(x = bb_train$rWAR, y = bb_train$inducted)
mod2 <- glm(bb_train$inducted ~ bb_train$rWAR, family = binomial)
lines(bb_train$rWAR, mod2$fitted.values)
```



```r
get_prob <- function(coefs, input){
  prob_new <- (exp(sum(coefs * cbind(1, input)))) / (1 + exp(sum(coefs * cbind(1, input))))
  names(prob_new) <- NULL
  return(prob_new)
}

par(pty = "s")
roc(bb_test$inducted, apply(bb_test[,"rWAR"], 1, get_prob, coefs = mod2$coefficients), plot = TRUE, legacy.axes = TRUE, percent = TRUE, xlab = "False Positive Percentage", ylab = "True Positive Percentage", col = "#377eb8", lwd = 4)
```

```
## Setting levels: control = FALSE, case = TRUE
```

```
## Setting direction: controls < cases
```



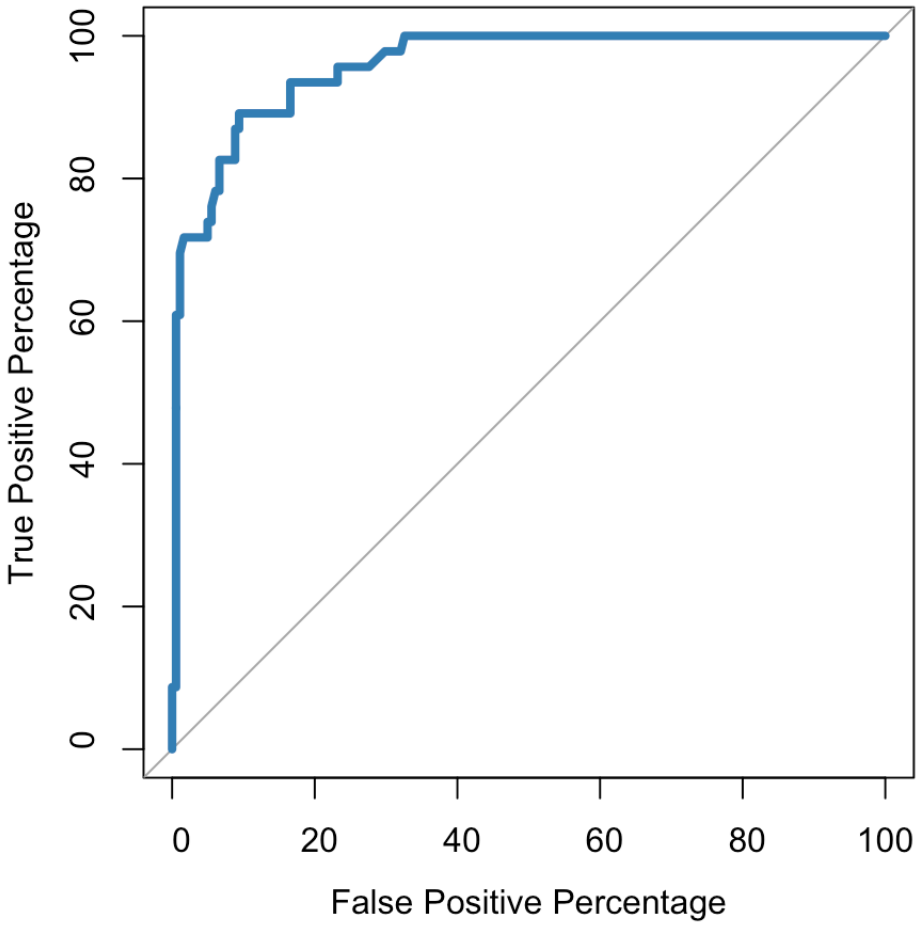
```
##
## Call:
## roc.default(response = bb_test$inducted, predictor = apply(bb_test[,     "rWAR"],
1, get_prob, coefs = mod2$coefficients), percent = TRUE,     plot = TRUE, legacy.axes
= TRUE, xlab = "False Positive Percentage",     ylab = "True Positive Percentage", co
l = "#377eb8", lwd = 4)
##
## Data: apply(bb_test[, "rWAR"], 1, get_prob, coefs = mod2$coefficients) in 4381 con
trols (bb_test$inducted FALSE) < 70 cases (bb_test$inducted TRUE).
## Area under the curve: 99.54%
```

Recreating Model

```
# The model used only position players that were considered by the Hall of Fame panel (not including historical players)
ml_mod <- baseball %>% filter(!is.na(baseball$firstYearHOF) & position != "P" & baseball$votedBy != "Old Timers" & str_sub(baseball$playerID, -2, -1) != "99")
ml_mod$debDec <- round(as.numeric(format(ml_mod$debut, "%Y")), -1)
indeces = sort(sample(nrow(ml_mod), nrow(ml_mod)*.7))
train <- ml_mod[indeces,]
test <- ml_mod[-indeces,]

rf.1 <- randomForest(factor(inducted) ~ bats + throws + BB + G.B + goldGloves + MVPs + H + HR + R + RBI + SB + A + PO + E + debDec + firstYearHOF + position + team + allStarApps + OBP + SLG + BA + BAPost + GPost + WSW + WSL + steriods, data = train)

par(pty = "s")
roc(test$inducted, predict(rf.1, test, type = "prob")[,1], plot = TRUE, legacy.axes = TRUE, percent = TRUE, xlab = "False Positive Percentage", ylab = "True Positive Percentage", col = "#377eb8", lwd = 4)
```

```
## Setting levels: control = FALSE, case = TRUE
```

```
## Setting direction: controls > cases
```



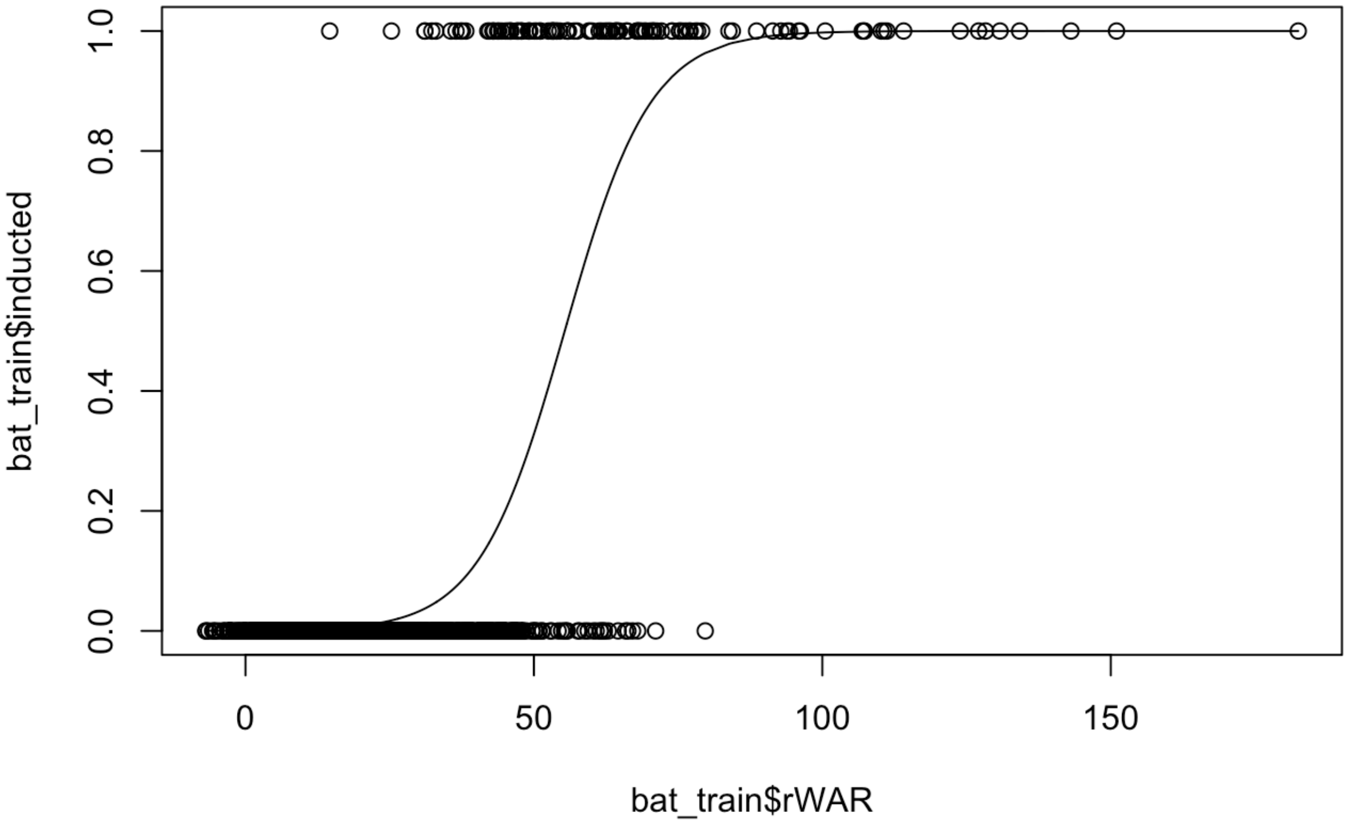
```
##
## Call:
## roc.default(response = test$inducted, predictor = predict(rf.1,     test, type = "
prob")[, 1], percent = TRUE, plot = TRUE, legacy.axes = TRUE,     xlab = "False Posit
ive Percentage", ylab = "True Positive Percentage",     col = "#377eb8", lwd = 4)
##
## Data: predict(rf.1, test, type = "prob")[, 1] in 181 controls (test$inducted FALSE
) > 46 cases (test$inducted TRUE).
## Area under the curve: 95.8%
```

Summary Statistics of HOF Inductees

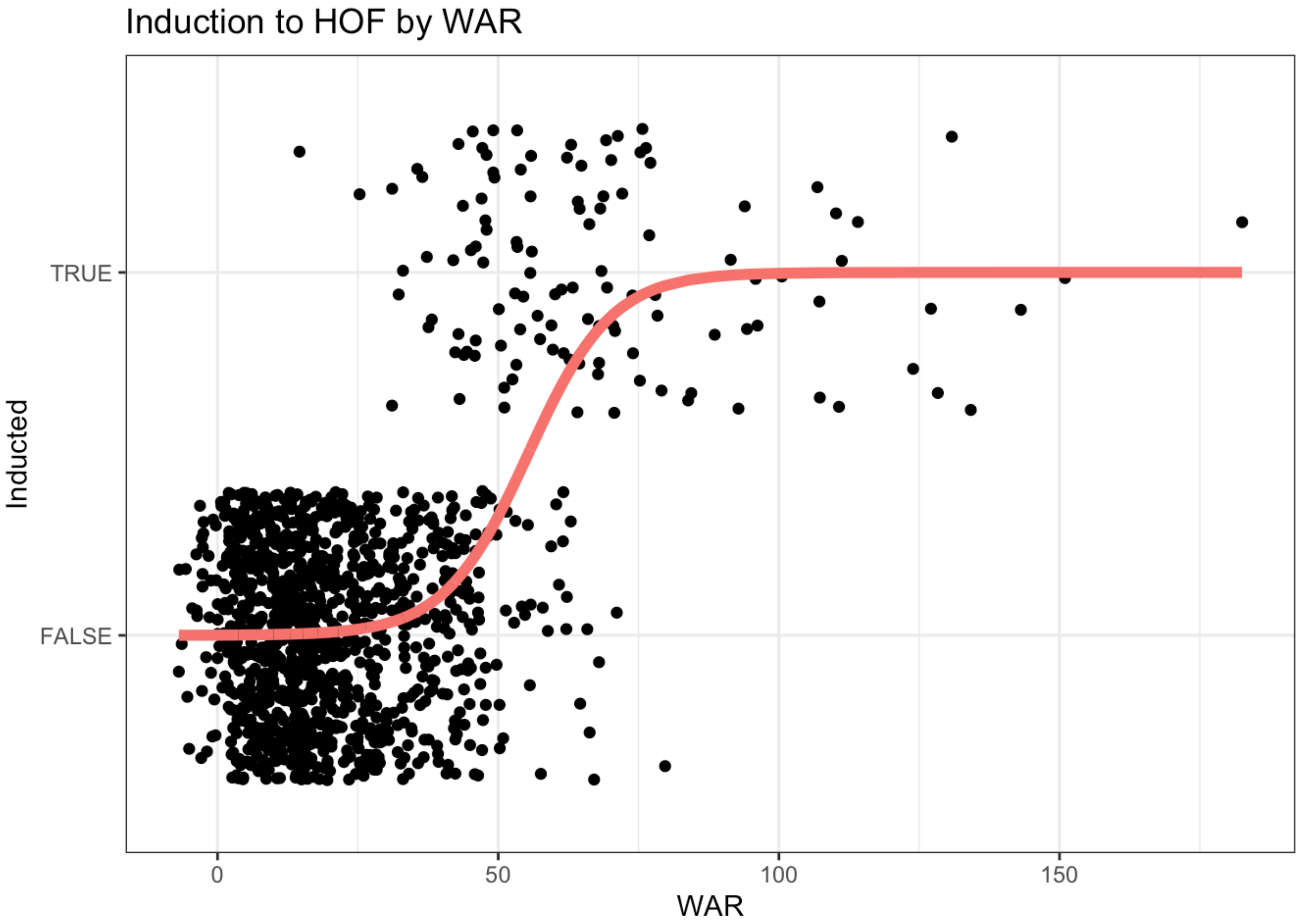
```
HOF_batters <- baseball %>% filter(inducted & position != "P" & votedBy != "Negro League" & str_sub(playerID, -2, -1) != "99")
AB_IQR <- summary(HOF_batters$AB)[5] - summary(HOF_batters$AB)[2]
low_outliers_AB <- summary(HOF_batters$AB)[2] - 1.5*AB_IQR

batters <- baseball %>% filter(position != "P" & votedBy != "Negro League" & str_sub(playerID, -2, -1) != "99" & AB >= low_outliers_AB & finalGame < as.Date("2004-01-01"))
paste0(sum(batters$inducted), " batters, about ", round(sum(batters$inducted)/dim(batters)[1]*100, 0), "% of all qualified batters are HOF inductees")
```

```
## [1] "146 batters, about 10% of all qualified batters are HOF inductees"
```

```
indeces <- sort(sample(nrow(batters), nrow(batters)*.8))
bat_train <- batters[indeces,]
bat_test <- batters[-indeces,]

mod1 <- glm(bat_train$inducted ~ bat_train$rWAR, family = "binomial")
plot(bat_train$rWAR, bat_train$inducted)
lines(sort(bat_train$rWAR), sort(mod1$fitted.values))
```



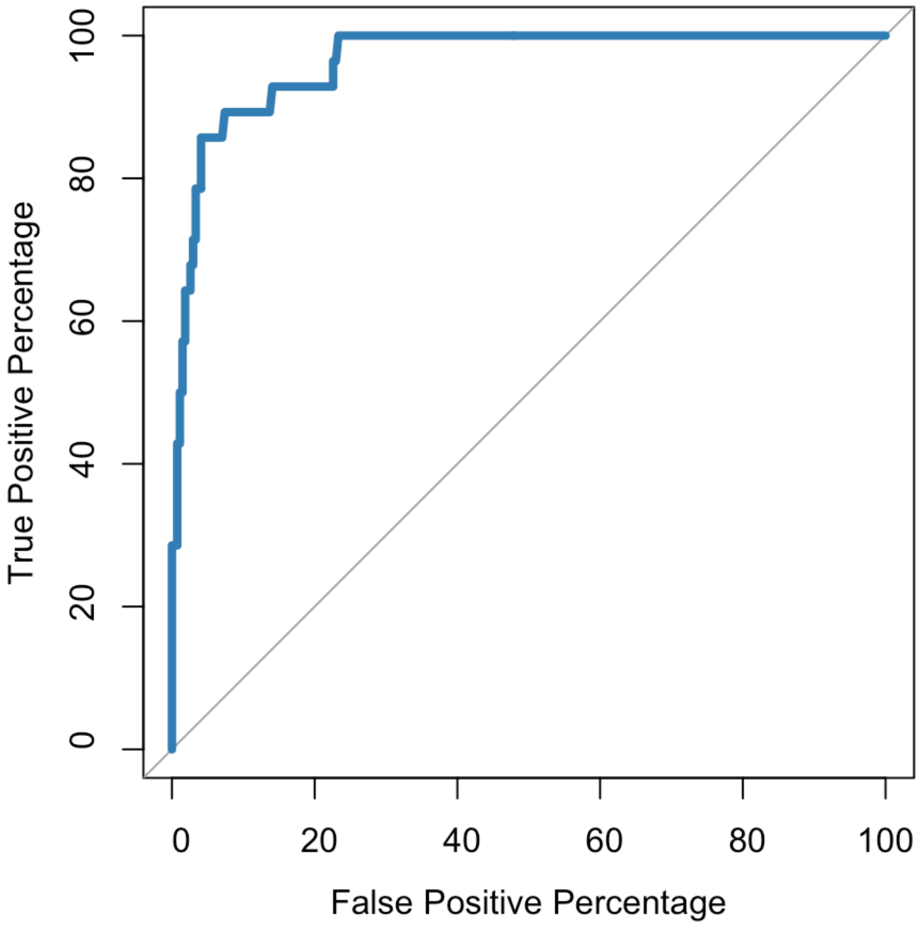
```
ggplot(bat_train, aes(rWAR, inducted)) + geom_jitter() + theme_bw() + geom_line(aes(s
ort(rWAR), sort(mod1$fitted.values) + 1, color = "red"), size = 2) + labs(title = "In
duction to HOF by WAR", x = "WAR", y = "Inducted") + theme(legend.position = "none")
```



```
par(pty = "s")
roc(bat_test$inducted, apply(bat_test[,"rWAR"], 1, get_prob, coefs = mod1$coefficient
s), plot = TRUE, legacy.axes = TRUE, percent = TRUE, xlab = "False Positive Percentag
e", ylab = "True Positive Percentage", col = "#377eb8", lwd = 4)
```

```
## Setting levels: control = FALSE, case = TRUE
```

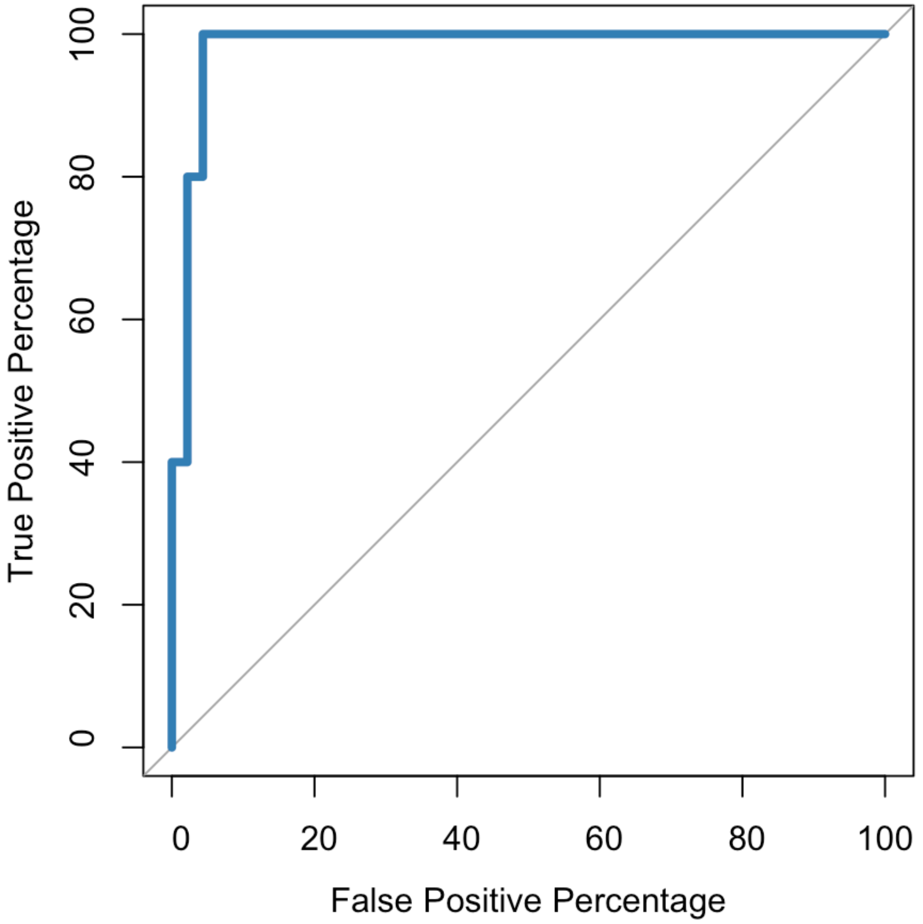
```
## Setting direction: controls < cases
```



```
##
## Call:
## roc.default(response = bat_test$inducted, predictor = apply(bat_test[,     "rWAR"]
, 1, get_prob, coefs = mod1$coefficients), percent = TRUE,     plot = TRUE, legacy.ax
es = TRUE, xlab = "False Positive Percentage",     ylab = "True Positive Percentage",
col = "#377eb8", lwd = 4)
##
## Data: apply(bat_test[, "rWAR"], 1, get_prob, coefs = mod1$coefficients) in 270 con
trols (bat_test$inducted FALSE) < 28 cases (bat_test$inducted TRUE).
## Area under the curve: 96.46%
```

```
# Early Baseball 1871-1900
eb_bat_train <- bat_train %>% filter(debut < as.Date("1899-01-01"))
eb_bat_test <- bat_test %>% filter(debut < as.Date("1899-01-01"))
mod2.eb <- glm(inducted ~ R, data = eb_bat_train, family = "binomial")
preds <- rep(0, dim(eb_bat_test)[1])
for (i in 1:length(preds)) {
  preds[i] <- get_prob(mod2.eb$coefficients, eb_bat_test[i, c("R")])
}
par(pty = "s")
roc(eb_bat_test$inducted, preds, plot = TRUE, legacy.axes = TRUE, percent = TRUE, xla
b = "False Positive Percentage", ylab = "True Positive Percentage", col = "#377eb8",
lwd = 4)
```

```
## Setting levels: control = FALSE, case = TRUE
## Setting direction: controls < cases
```



```
##
## Call:
## roc.default(response = eb_bat_test$inducted, predictor = preds,     percent = TRUE
, plot = TRUE, legacy.axes = TRUE, xlab = "False Positive Percentage",     ylab = "Tr
ue Positive Percentage", col = "#377eb8", lwd = 4)
##
## Data: preds in 46 controls (eb_bat_test$inducted FALSE) < 5 cases (eb_bat_test$ind
ucted TRUE).
## Area under the curve: 98.26%
```

```
summary(mod2.eb)
```

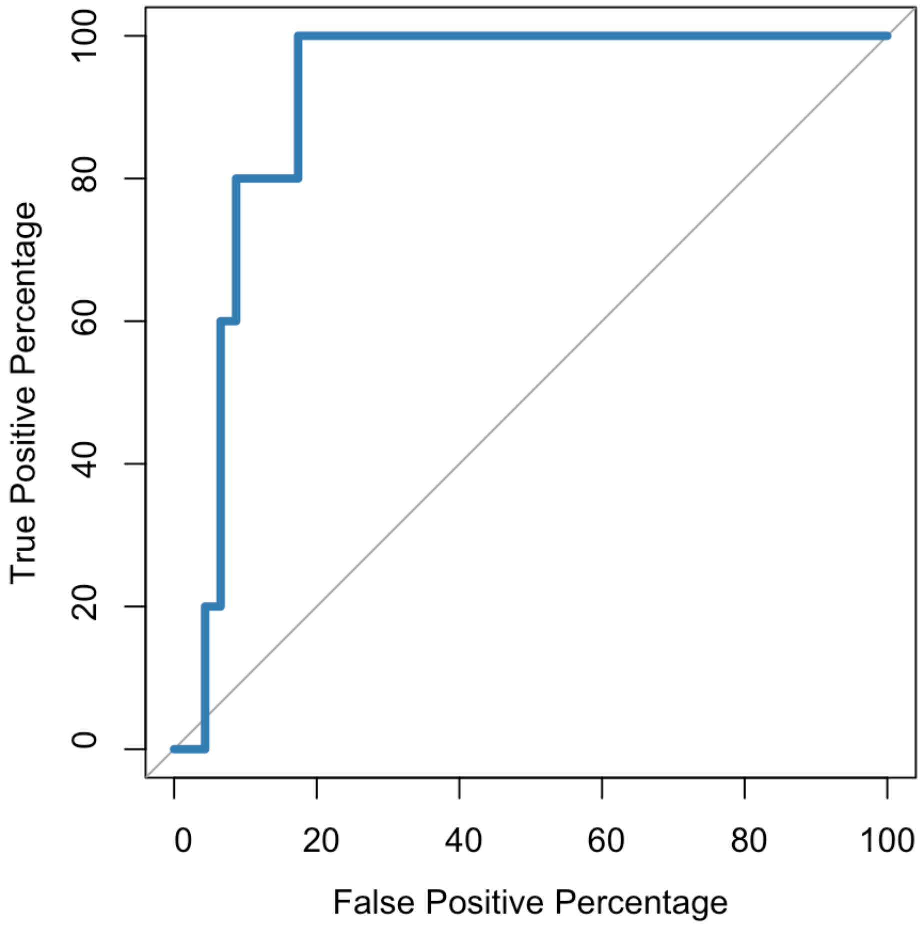
```
##
## Call:
## glm(formula = inducted ~ R, family = "binomial", data = eb_bat_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7676  -0.3496  -0.1981  -0.1225   2.6911
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -7.0895559  1.0806297  -6.561 5.36e-11 ***
## R            0.0051258  0.0009028   5.677 1.37e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 136.464  on 175  degrees of freedom
## Residual deviance:  80.539  on 174  degrees of freedom
## AIC: 84.539
##
## Number of Fisher Scoring iterations: 6
```

```
# Deadball Era 1901-1919
db_bat_train <- bat_train %>% filter(debut > as.Date("1899-01-01") & debut < as.Date(
"1917-01-01"))
db_bat_test <- bat_test %>% filter(debut < as.Date("1899-01-01") & debut < as.Date("1
917-01-01"))
mod2.db <- glm(inducted ~ G.B, data = db_bat_train, family = "binomial")
summary(mod2.db)
```

```
##
## Call:
## glm(formula = inducted ~ G.B, family = "binomial", data = db_bat_train)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.47387  -0.20655  -0.09617  -0.04320   2.40937
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -12.149344   2.293461  -5.297 1.17e-07 ***
## G.B           0.005907   0.001211   4.879 1.07e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 118.587  on 167  degrees of freedom
## Residual deviance:  46.732  on 166  degrees of freedom
## AIC: 50.732
##
## Number of Fisher Scoring iterations: 7
```

```
preds <- rep(0, dim(db_bat_test)[1])
for (i in 1:length(preds)) {
  preds[i] <- get_prob(mod2.db$coefficients, eb_bat_test[i, c("G.B")])
}
par(pty = "s")
roc(db_bat_test$inducted, preds, plot = TRUE, legacy.axes = TRUE, percent = TRUE, xla
b = "False Positive Percentage", ylab = "True Positive Percentage", col = "#377eb8",
lwd = 4)
```

```
## Setting levels: control = FALSE, case = TRUE
## Setting direction: controls < cases
```



```
##
## Call:
## roc.default(response = db_bat_test$inducted, predictor = preds,     percent = TRUE
, plot = TRUE, legacy.axes = TRUE, xlab = "False Positive Percentage",     ylab = "Tr
ue Positive Percentage", col = "#377eb8", lwd = 4)
##
## Data: preds in 46 controls (db_bat_test$inducted FALSE) < 5 cases (db_bat_test$ind
ucted TRUE).
## Area under the curve: 91.3%
```

```
# BB + G.B + goldGloves + MVPs + H + HR + R + RBI + SB + A + PO + E + OBP + SLG + BA
+ BAPost + GPost + WSW + WSL + steriods, data = db_bat_train
```