Homework 3

Neal Marquez

October 12, 2018

```r
rm(list=ls())
library(tidyverse)
library(latex2exp)
library(boot)
library(glmnet)
library(plotROC)
library(lmtest)
library(mvtnorm)

cyDF <- read_csv("https://faculty.washington.edu/cadolph/mle/cyyoung.csv") %>%
    mutate(id=1:n())


predict.opt <- function(model, newdata, formula, sefit=NULL){
    X <- model.matrix(formula[-2], newdata)
    rez <- c(X %*% model$par)
    if(is.null(sefit)){
        return(rez)
    }
    betas <- rmvnorm(sefit, model$par, solve(model$hessian))
    preds <- X %*% t(betas)
    tibble(
        fit = rez,
        lwr = apply(preds, 1, quantile, probs=.025),
        upr = apply(preds, 1, quantile, probs=.975)
    )
}
```

Question 1

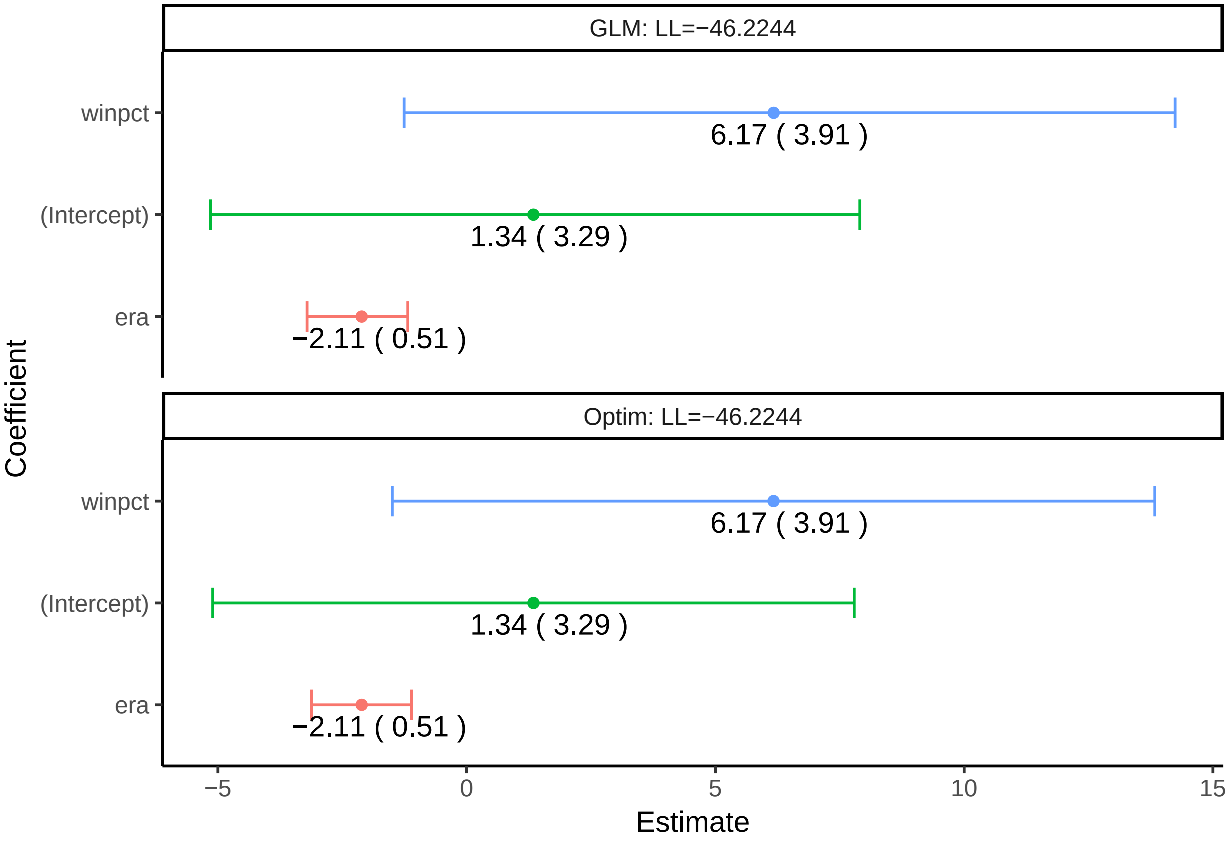
The dataset `cyyoung.csv` contains information on selected North American baseball pitchers from 1980 to 2002. Pitchers' performance can be measured in several ways: their record of games won or lost, the number of runs (points) they allowed the other team to score per game, the number of players they "struck out," the number of players they "walked," and the number of innings they pitched. At the end of the season, two pitchers (one from the American League, and one from the National League) are voted the best pitchers of the year.

Section A

Fit a logistic regression to the variable `cy` with `era` and `winpct` as the only covariates. Report the estimated parameters, their standard errors, and the log likelihood at its maximum. Perform this fit using `optim()`, then replicate the fit using `glm()`.

Fitting our model using either the canned `glm` function in `R` vis writing our own likelihood in optim provided similar results as seen in the plot below. The likelihoods for both models, the coefficients, and the standard errors each provided similar results. From this point forward I will use `optim` for all analysis.

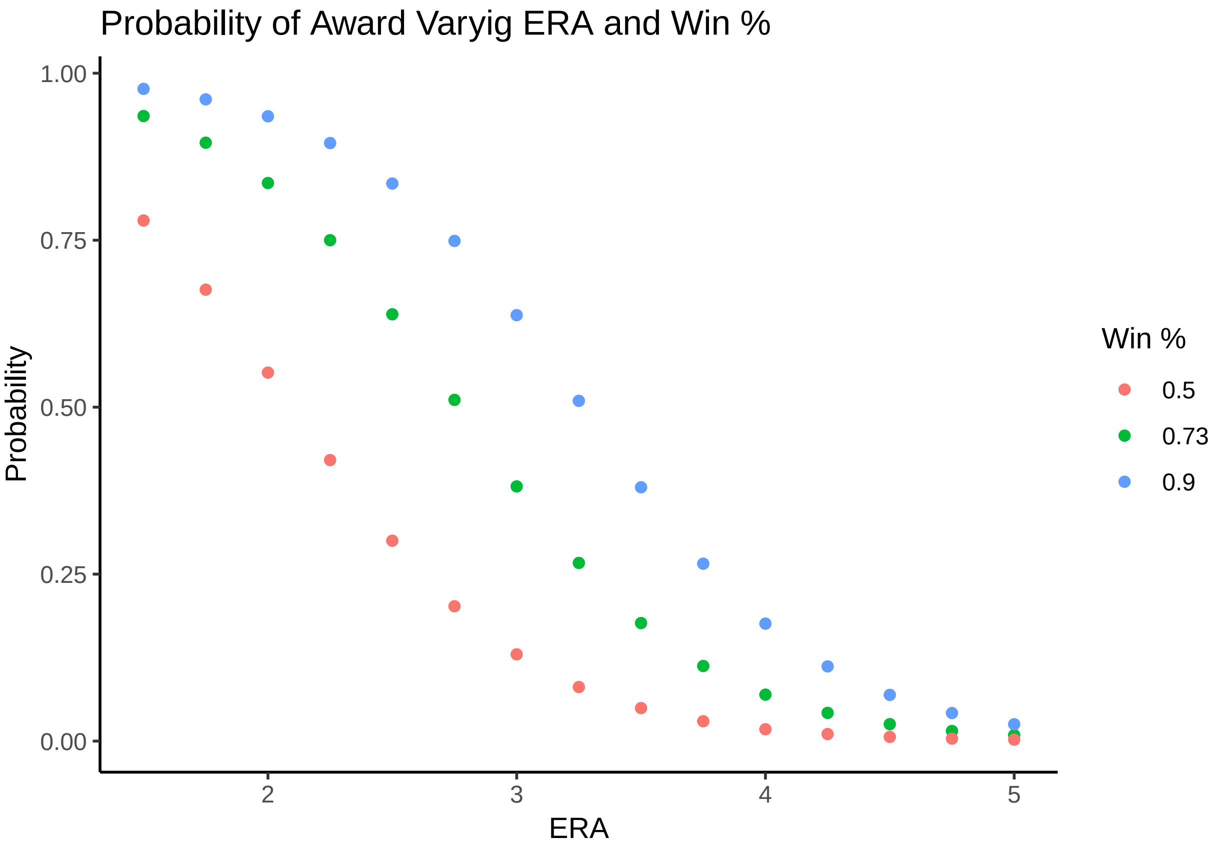
```r
evalParams <- function(params, formula, data=cyDF){
    X <- model.matrix(formula[-2], data)
    p <- inv.logit(X %*% params)
    y <- data$cy
    -sum(log((1-p)^(1-y) * p^y))
}

fog <- cy ~ era + winpct
runOPT <- optim(c(0, 0, 0), evalParams, formula=fog, hessian=T)
runGLM <- glm(fog, data=cyDF, family=binomial)
llGLM <- as.numeric(round(logLik(runGLM), 4))

vcovMat <- solve(runOPT$hessian)
se_ <- sqrt(diag(vcovMat))

bind_rows(
    tibble(coeff=names(runGLM$coefficients), est=runOPT$par, se=se_) %>%
        mutate(`97.5 %`=est + se*1.96, `2.5 %`=est - se*1.96) %>%
        mutate(Approach=paste0("Optim: LL=", round(-runOPT$value,4))),
    as_tibble(confint(runGLM)) %>%
        mutate(est=runGLM$coefficients, coeff=names(runGLM$coefficients)) %>%
        mutate(Approach=paste0("GLM: LL=", llGLM)) %>%
        mutate(se=sqrt(diag(vcov(runGLM))))) %>%
    mutate(text=paste(round(est, 2), "(", round(se, 2),")")) %>%
    ggplot(aes(x=coeff, y=est, ymin=`2.5 %`, ymax=`97.5 %`, color=coeff)) +
    geom_point() +
    geom_errorbar(width=.3) +
    facet_wrap(~Approach, nrow=2) +
    theme_classic() +
    guides(color=FALSE) +
    coord_flip() +
    labs(x="Coefficient", y="Estimate") +
    geom_text(aes(label=text, vjust=1.5, color=NULL, hjust=.4))
```



Section B

Without using a special package like `simcf`, calculate the probability a pitcher receives the Cy Young award given `era`$= \{1.50, 1.75, 2.00, \ldots, 4.75, 5.00\}$ with `winpct` held at its mean value. Now, calculate the probability again, for the same range of `era`'s, given either `winpct`$=$ 0.5 or `winpct`$=$ 0.9. You should end up with $3 \times 15 = 45$ probabilities. Plot these estimated probabilities nicely (the tile package works well for this graphic and the next problem, but for this part, even a matrix works well).

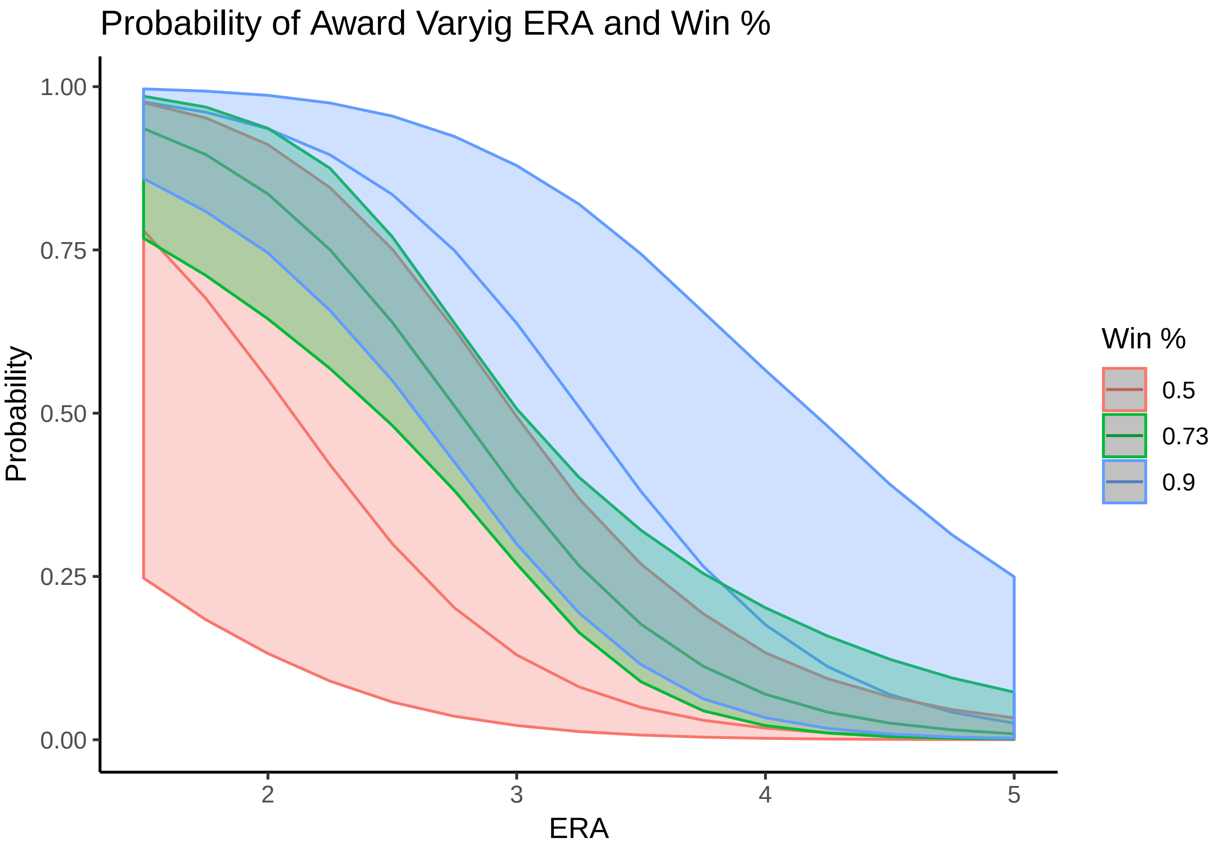
```
expand.grid(era=seq(1.5, 5, by=.25), winpct=c(mean(cyDF$winpct), .5, .9)) %>%
    mutate(prob=inv.logit(predict.opt(runOPT, newdata=., fog))) %>%
    mutate(winpct=as.factor(round(winpct, 2))) %>%
    ggplot(aes(x=era, y=prob, group=winpct, color=winpct)) +
    geom_point() +
    theme_classic() +
    labs(x="ERA", y="Probability", color="Win %") +
    ggtitle("Probability of Award Varyig ERA and Win %")
```



Section C

Calculate or simulate 95 percent confidence intervals for each of the probabilities plotted in b. (You may now use any package you wish.) Design a graphic (not a table) to incorporate these confidence intervals. Interpret your findings.

```
expand.grid(era=seq(1.5, 5, by=.25), winpct=c(mean(cyDF$winpct), .5, .9)) %>%
    cbind(predict.opt(runOPT, newdata=., fog, sefit = 10000)) %>%
    mutate_at(c("fit", "lwr", "upr"), inv.logit) %>%
    mutate(winpct=as.factor(round(winpct, 2))) %>%
    ggplot(
        aes(x=era, y=fit, group=winpct, color=winpct, ymin=lwr, ymax=upr,
            fill=winpct)) +
    geom_line() +
    geom_ribbon(alpha=.3) +
    theme_classic() +
    labs(x="ERA", y="Probability", color="Win %") +
    guides(fill=FALSE) +
    ggtitle("Probability of Award Varyig ERA and Win %")
```



Section D

Find a “better model” of `cy`. You may add other variables from the dataset, remove variables already in the model, and/or transform or any variables you wish, except `cy`. Whatever choice you make you should justify in some fashion. Fit your new model, and show whether your fit has improved using **(i)** a likelihood ratio test, **(ii)** AIC and/or BIC, **(iii)** in-sample ROC curves, **(iv)** in-sample Actual versus Predicted plots, and **(v)** cross-validation using the metric(s) of your choice.

I added in a new term for year and multiplied it alongside ERA as ERAs have improved over time and we might expect that lower eras would be more important in more recent years. The addition of two new terms was tested using the metrics above and for each test the inclusion of the new parameters was justified.

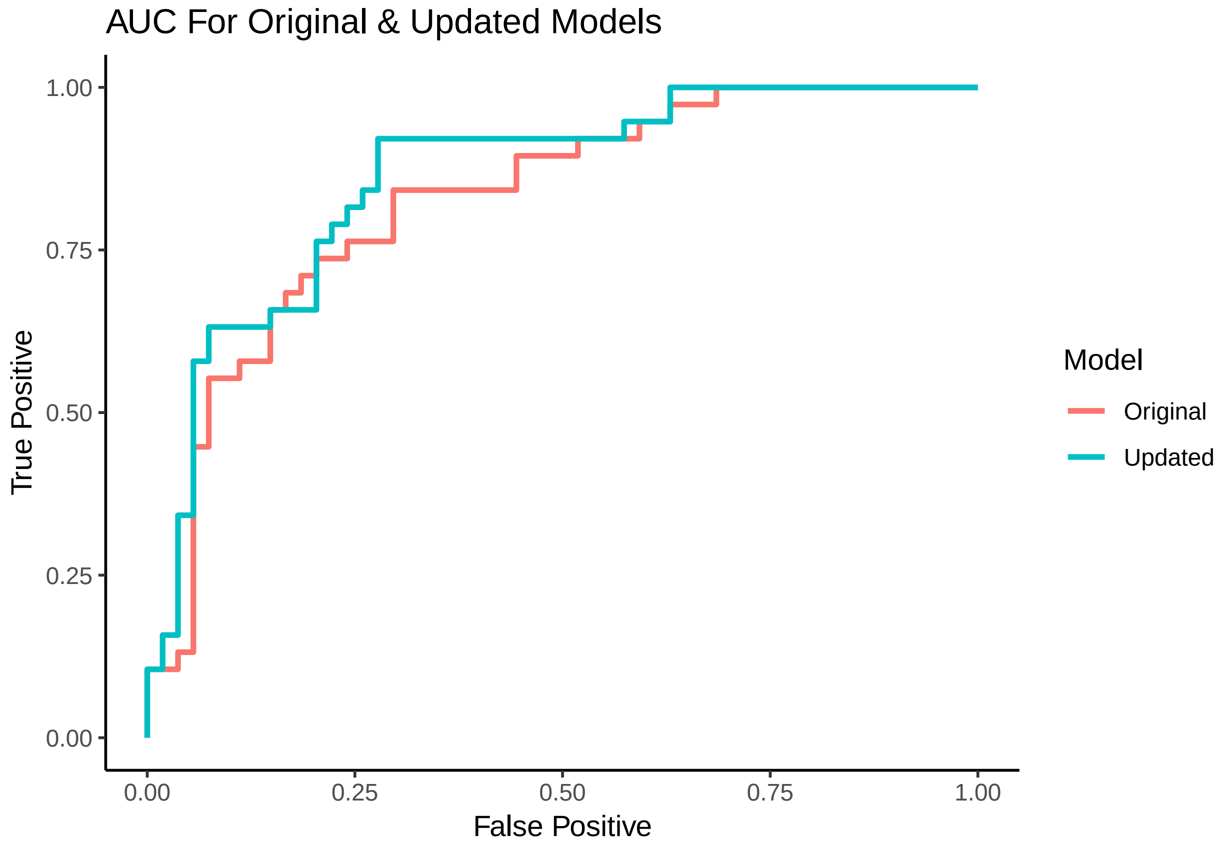
```
fupdate <- cy ~ era * year + winpct
parLength <- length(labels(terms(fupdate[-2]))) + 1
updateGLM <- glm(fupdate, data=cyDF, family=binomial)
# we set the baseline parameters to the fitted GLM data but we could get
# the same result by scling the coefficients
updateOPT <- optim(coef(updateGLM), evalParams, formula=fupdate, hessian=T)
lrtest1 <- round(
    pchisq(-2 * (-runOPT$value + updateOPT$value), 2, lower.tail=F), 4)

cat(paste(
    "Likelihood Ratio Test For Additional Parameters (p-value)",
    lrtest1, sep="\n"))
```

```
## Likelihood Ratio Test For Additional Parameters (p-value)
## 0.0327
```

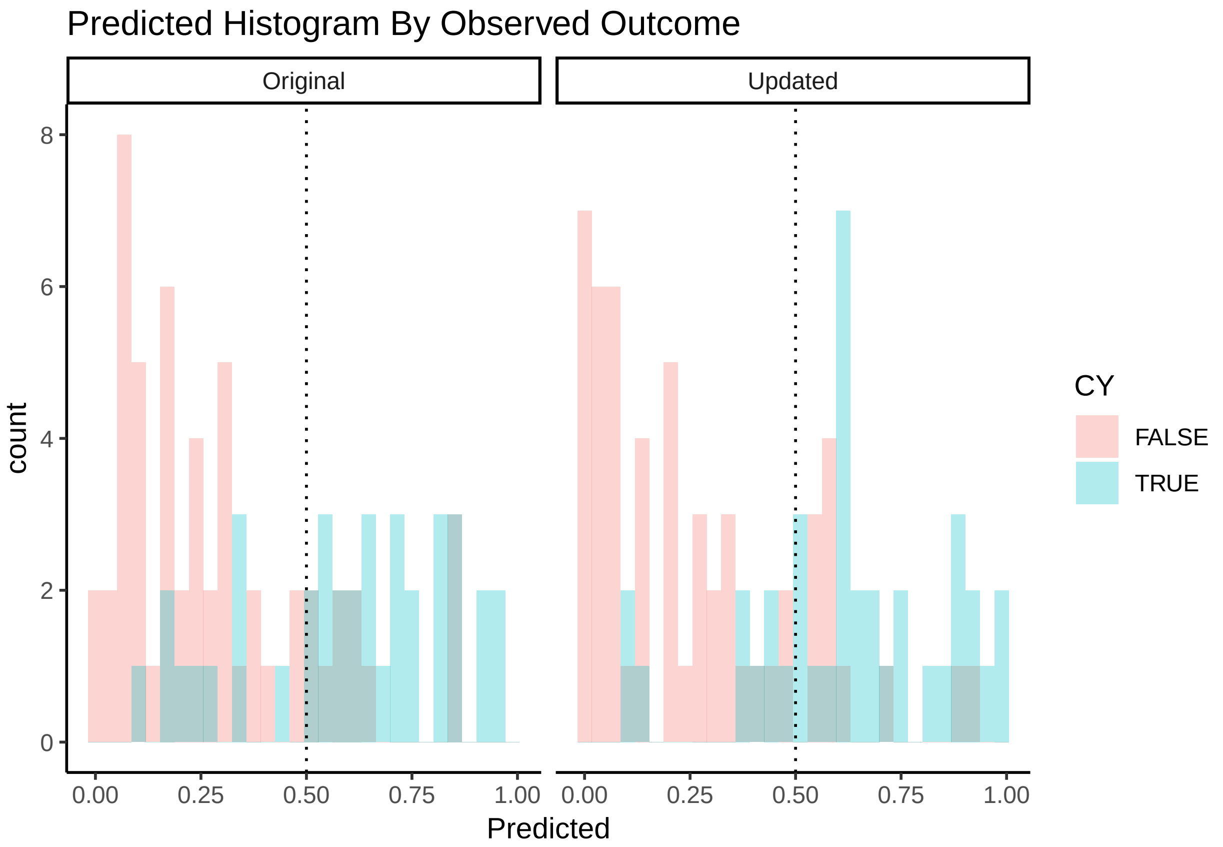
```r
AICs <- c(
    Original = round(2*(length(runOPT$par)) + 2 * runOPT$value, 4),
    Updated = round(2*(length(updateOPT$par)) + 2 * updateOPT$value, 4))

cat(paste0("AIC Calculations for model ", names(AICs), ": ", AICs, "\n"))
```

```
## AIC Calculations for model Original: 98.4487
##  AIC Calculations for model Updated: 95.6104
```

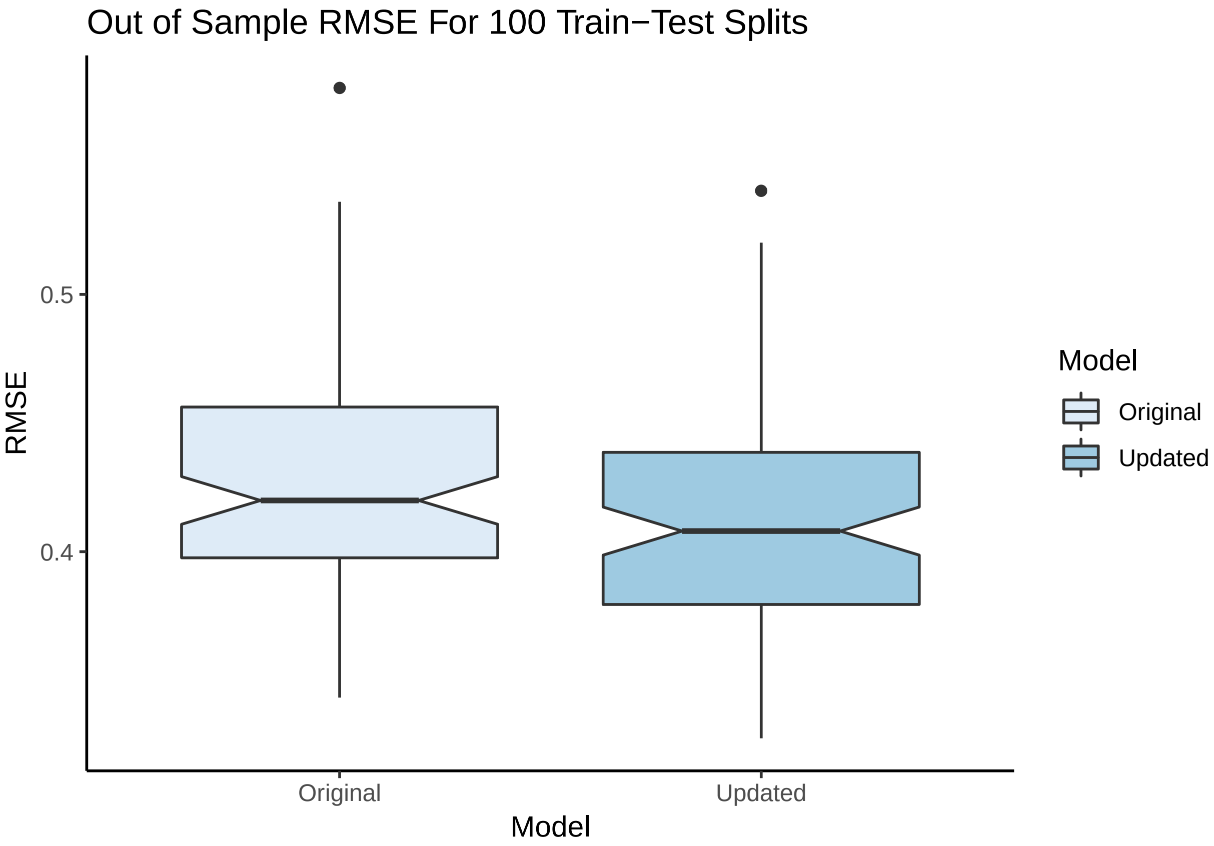
```r
predDF <- rbind(
    data.frame(
        predictor = inv.logit(predict.opt(runOPT, cyDF, fog)),
        known.truth = cyDF$cy,
        Model = "Original"),
    data.frame(
        predictor = inv.logit(predict.opt(updateOPT, cyDF, fupdate)),
        known.truth = cyDF$cy,
        Model = "Updated"))

predDF %>%
    ggplot(aes(d = known.truth, m = predictor, color = Model)) +
    geom_roc(n.cuts = 0) +
    theme_classic() +
    labs(x="False Positive", y="True Positive") +
    ggtitle("AUC For Original & Updated Models")
```



```r
predDF %>%
    mutate(CY=known.truth == 1) %>%
    ggplot(aes(x=predictor, group=CY, fill=CY)) +
    geom_histogram(alpha=.3, position="identity", bins=30) +
    theme_classic() +
    facet_wrap(~Model) +
    geom_vline(xintercept=.5, linetype=3) +
    xlab("Predicted") +
    ggtitle("Predicted Histogram By Observed Outcome")
```



```r
set.seed(12345)
M <- 100
trainSize <- .7

resultsDF <- sapply(1:M, function(x){
    trainDF <- sample_frac(cyDF, size=trainSize) %>%
        mutate(trainID=1:n())
    testDF <- select(trainDF, id, trainID) %>%
        right_join(cyDF, by="id") %>%
        filter(is.na(trainID))

    trainOGGLM <- optim(
        c(0, 0, 0), evalParams, formula=fog, data=trainDF, hessian=T)
    trainUPGLM <- optim(
        coef(updateGLM), evalParams, formula=fupdate, data=trainDF, hessian=T)

    predOG <- inv.logit(predict.opt(trainOGGLM, newdata=testDF, fog))
    predUP <- inv.logit(predict.opt(trainUPGLM, newdata=testDF, fupdate))

    res <- c(
        sqrt(sum((testDF$cy - predOG)^2) / nrow(testDF)),
        sqrt(sum((testDF$cy - predUP)^2) / nrow(testDF)))
    res}) %>%
    t %>%
    as.data.frame %>%
    rename(Original=V1, Updated=V2)

resultsDF %>%
    gather("Model", "RMSE") %>%
    ggplot(aes(x=Model, y=RMSE, fill=Model)) +
    geom_boxplot(notch=TRUE) +
    theme_classic() +
    scale_fill_brewer(palette="Blues") +
    ggtitle("Out of Sample RMSE For 100 Train-Test Splits")
```



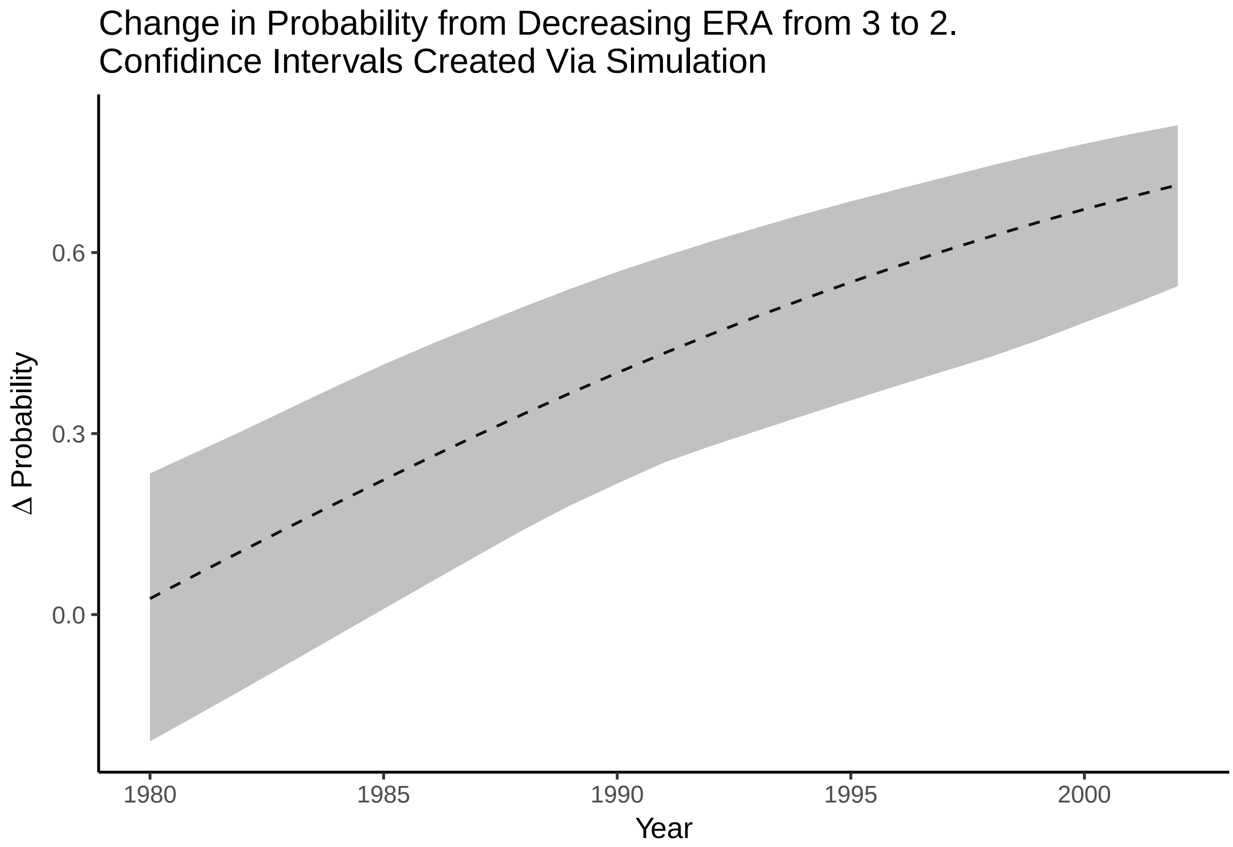
Section E

Suppose your model from d. has the following form:

$$\text{cy}_i \sim \text{Bernoulli}(\pi)$$
$$\pi_i = \text{logit}^{-1}(\alpha + z_i\gamma + \boldsymbol{x}_i\boldsymbol{\beta})$$

where z_i is a covariate of particular interest and x_i is a vector of additional covariates. We are interested in understanding how the conditional expectation of `cy` changes as $z$such that z chnages from z_{pre} to z_{post}. Then simulate the first difference in the probability (or, if you prefer, the relative risk) of receiving a Cy Young given the change in z, holding other covariates $\boldsymbol{x}$ constant (that is, make sure $\boldsymbol{x}_{\text{pre}} = \boldsymbol{x}$post). Display your results neatly in a plot or table, and be sure to show the uncertainty in these estimates.

```r
M <- 10000
betaDraws <- t(rmvnorm(M, updateOPT$par, solve(updateOPT$hessian)))

expand.grid(era=c(2, 3), year=1980:2002, winpct=mean(cyDF$winpct)) %>%
    mutate(`(Intercept)`=1, `era:year`=era*year) %>%
    select(`(Intercept)`, era, year, winpct, `era:year`) %>%
    cbind(as.matrix(.) %*% betaDraws) %>%
    gather(key="simulation", value="value", `1`:`100`) %>%
    mutate(simulation=as.numeric(simulation), value=inv.logit(value)) %>%
    arrange(simulation, year, era) %>%
    group_by(year, simulation) %>%
    summarize(diff=nth(value, 1) - nth(value, 2)) %>%
    summarize(
        mu=mean(diff),
        lwr=quantile(diff, probs=.025),
        upr=quantile(diff, probs=.975)) %>%
    ggplot(aes(x=year, y=mu, ymin=lwr, ymax=upr)) +
    geom_line(linetype=2) +
    geom_ribbon(alpha=.3) +
    theme_classic() +
    labs(y=TeX("$\\Delta$ Probability"), x="Year") +
    ggtitle(paste(
        "Change in Probability from Decreasing ERA from 3 to 2.",
        "Confidince Intervals Created Via Simulation", sep="\n"))
```



Section F

Does logistic regression offer a defensible probability model here? What assumptions of this model might be violated by the variable `cy`?

The logistic model is not an appropriate probability model as it assumes that each players chance of winning is independent when we know that only one person can win in each division every year. A more appropriate model would select one winner from a pool of applicants within a division no matter the size of the pool. We know that there can only be one winner every year and our probability model should reflect that.

Section G

Suppose the pitchers selected for inclusion in the dataset were all considered "contenders" for the Cy Young award by knowledgeable experts. Pitchers whom the experts considered unlikely to win the award were excluded. How might this fact affect your findings?

If this dataset only included contenders selected by a panel of knowledgeable experts then future observations would need to take that into consideration as well. That is future observations that were not selected by a panel would need to be evaluated differently. Furthermore, we can see that within our dataset that even though two individuals should be selected every year, this is always the case. If these players were indeed selected by knowledgeable experts it appears that they are not the only players who win the award every year.