Dylan Webb
4/13/2021

Bayesian Analysis in Golf Tournament Prediction

Golf is a popular sport worldwide, with large tournaments occurring on an almost weekly basis in the United States. To correctly anticipate the winners of these tournaments, I created a random forest prediction model using Python earlier this year. My model successfully predicts eight likely candidates to win any upcoming PGA tournament, but usually fails to anticipate which of these eight will perform the best. To improve my weekly predictions, I employed Bayesian analysis to compare posterior-predictive distributions for the eight golfers of interest.

Methods

To perform Bayesian analysis on my eight golfers, I first scraped the past two and a half years of scoring data for all PGA tournaments directly from https://www.pgatour.com/stats.html. Due to the discrete nature of the data, I chose the Gamma-Poisson to model a golfer's score. In golf, the score is measured in strokes, or the number of times a golfer hits the ball. Course difficulty varies, and a tournament winner's score may range from 260 to 280 strokes. To remove this variation, I standardized the scoring data to be number of strokes made above the winner's score (the winner has a standardized score of 0). Thus, the parameter of the likelihood, θ, represents a specific golfer's average number of strokes above the winner's score across many tournaments.

Next, I checked the assumptions necessary for the Gamma-Poisson. Because a golfer's ability varies over time, his scores may not be independent and identically distributed. To control this effect, I included only the ten most recent scores for each golfer. The Poisson distribution also assumes equal mean and variance. For the standardized scores, however, variance was almost triple the mean, as seen in Figure 1 below. Due to this high variability, the final credible intervals may be narrower than they ought to be, overstating the strength of my inference on θ.

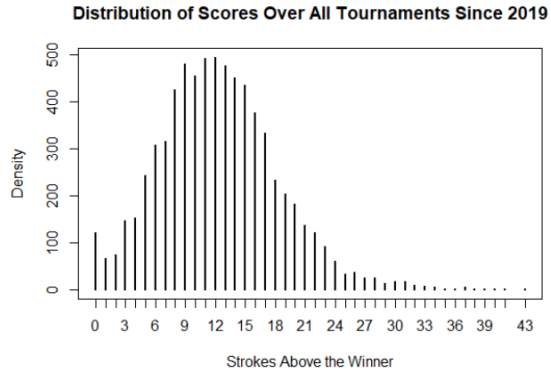

Figure 1. The distribution of all standardized scores since 2019. μ = 12.35 and σ^2 = 35.19

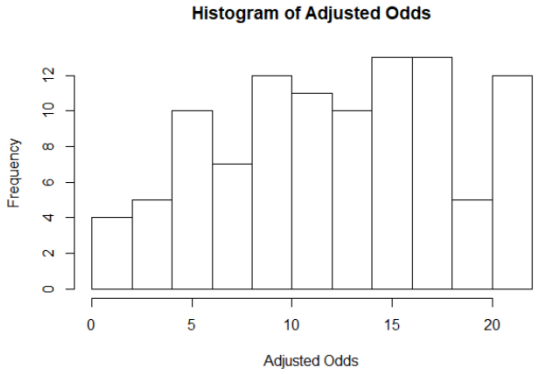

Figure 2. The distribution of adjusted betting odds for a single tournament. μ = 12.64 and σ^2 = 32.35

To define the prior distribution for the Gamma-Poisson, I needed an informed estimate of each golfer's ability. Because betting odds represent golf experts' a priori estimates of a golfer's current ability, with lower odds indicating a higher potential of winning, they seemed like an effective choice for the prior. Problematically, betting odds are on the scale of the thousands. By taking the square root and scaling, I was able to transform the betting odds for the Honda Classic (retrieved from https://www.vegasinsider.com/golf/odds/futures/ on 3/23) to follow a similar distribution as the standardized scores, as seen in Figure 2 above.

Following this preparation, I modeled each of the eight golfers' score using a Gamma-Poisson. To give the betting odds more weight in the posterior, I used a given golfer's adjusted betting odds multiplied by three as the prior shape, and three as the prior rate. I then added the golfer's ten most recent scores (standardized as described above) to the shape, and added ten to the rate, to obtain the posterior distribution. Finally, I calculated the maximum density value of $y_{new}|y_{obs}$ in the posterior-predictive distribution to predict the golfer's likely score in the Honda Classic.

Results

The predicted scores as well as the predicted ordering for all eight golfers are compared with the actual results of the Honda Classic below (scores are standardized).

	Sungjae Im	Lee Westwood	Keegan Bradley	Brendan Steele	Shane Lowry	Matthew NeSmith	Alex Noren	Keith Mitchell
Predicted Score	7	9	10	10	10	10	12	14
Actual Score	7	Cut	11	6	13	13	14	15

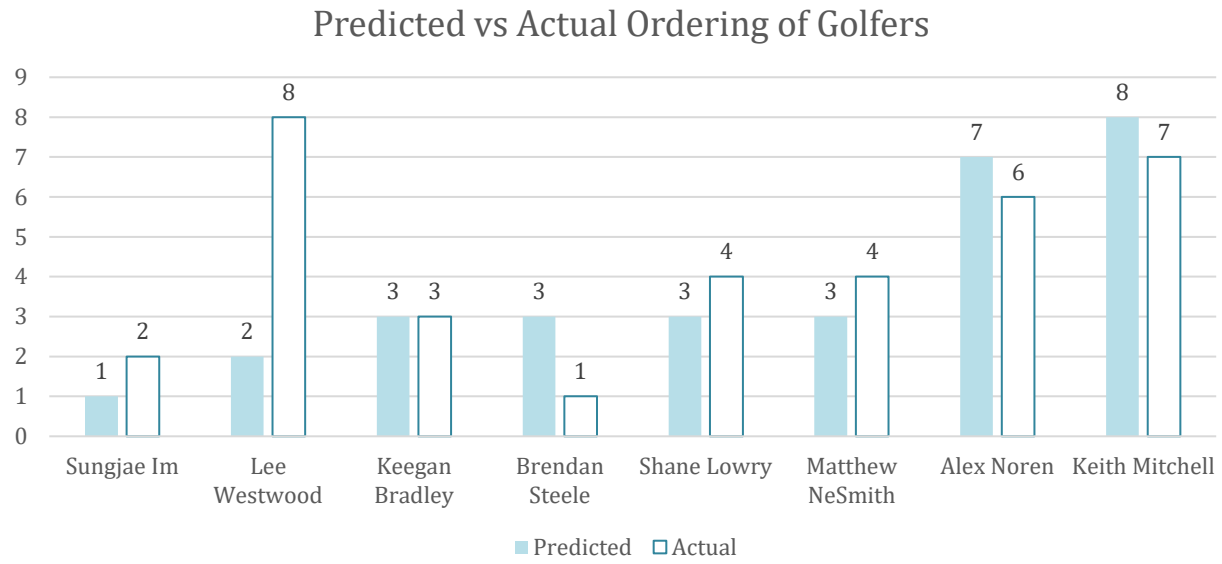

Figure 3. Bar chart of predicted vs actual ordering of Golfers in the Honda Classic

The predicted scores and predicted ordering for all eight golfers were surprisingly accurate, with only Lee Westwood deviating wildly from his predicted order, as seen in Figure 3. Unfortunately, Westwood was cut halfway through the tournament, so nothing can be said about his actual score—the fact that he placed last despite his strong betting odds is not so surprising considering the width of his associated interquartile range, as seen at continuation.

For brevity, I will only examine the full results of my analysis for the players with the top four predicted scores (Bradley and Steele were higher on the list produced by the random forest and are therefore included over Lowry or NeSmith). Below are the summary statistics for each of the four golfers' ten most recent scores (y_{obs}) along with their adjusted betting odds score.

	Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum	Adjusted Odds
Sungjae Im	4	7	10	9.5	11.75	16	1
Lee Westwood	1	3	13	11.56	19	26	2
Brendan Steele	2	9	12	12.5	13	26	5
Keegan Bradley	4	8	10.5	11.4	15.5	20	6

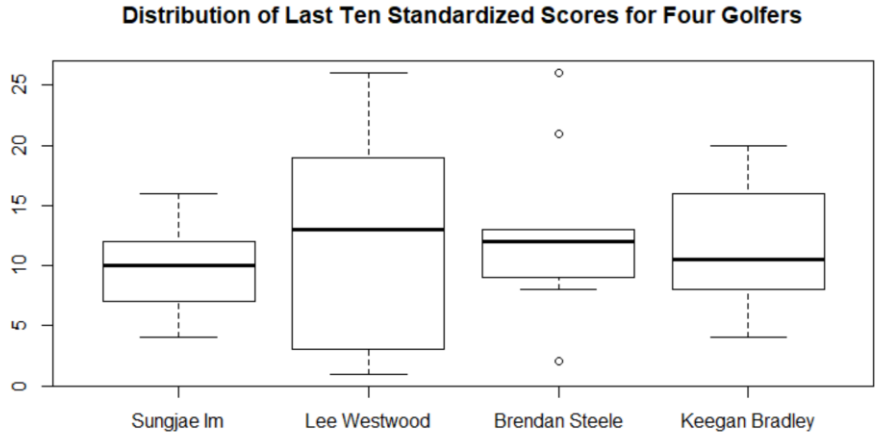

Figure 4. Side by side boxplots for (y_{obs}) for all four golfers.

As we can see in the summary statistics and Figure 4, Sungjae Im has the lowest mean for y_{obs}, the narrowest range of scores (while Westwood had the widest), and the lowest adjusted betting odds score, suggesting his average number of strokes above the winner to be the lowest. Indeed, the prior and posterior distributions (calculated as described in methods) reflect this trend:

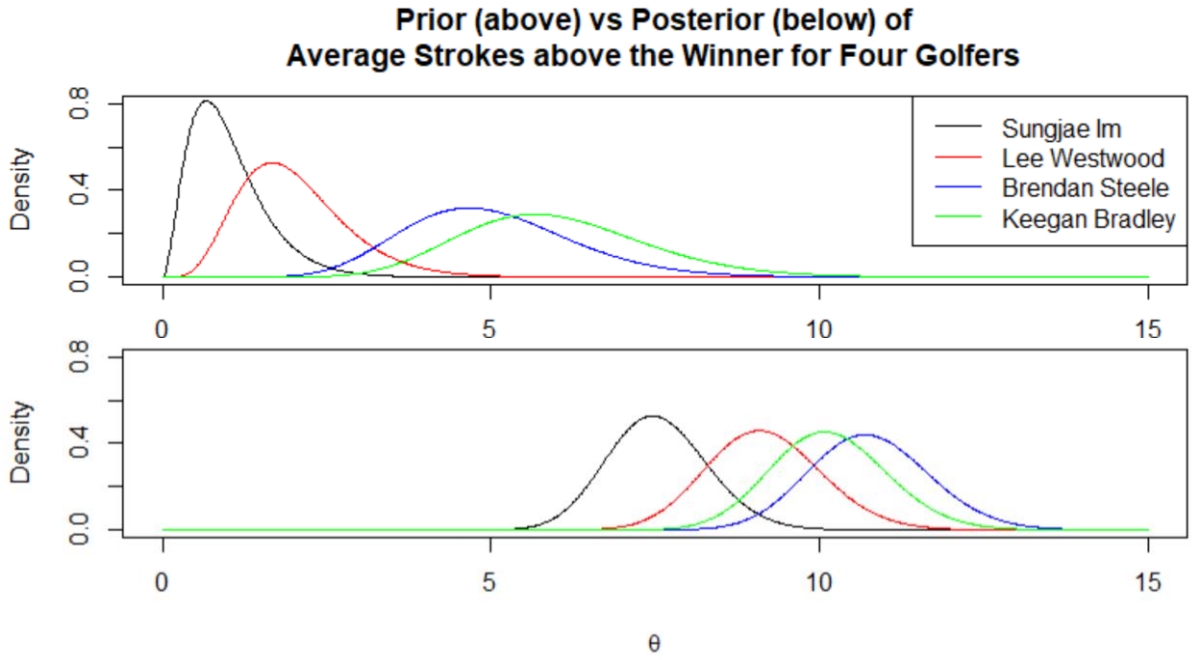

Figure 5. Comparison of the prior (top) and posterior (bottom) distributions for all four golfers

In Figure 5, we observe that adding the ten most recent scores shifts each golfer's posterior distribution farther to the right, but only switches the order of Brendan Steele and Keegan Bradley, reinforcing the validity of using betting odds as a prior. The 95% credible intervals for each golfer's average score (as number of strokes above the winner) are reported below:

Credible Intervals for Sungjae Im:
The prior probability that Im's average score lies between 0.2062 and 2.4082 is .95
The posterior probability that Im's average score lies between 6.1201 and 9.1024 is .95

Credible Intervals for Lee Westwood:
The prior probability that Westwood's average score lies between 0.7340 and 3.8894 is .95
The posterior probability that Westwood's average score lies between 7.5339 and 10.9572 is .95

Credible Intervals for Brendan Steele:
The prior probability that Steele's average score lies between 2.7985 and 7.8299 is .95
The posterior probability that Steele's average score lies between 9.0593 and 12.6248 is .95

Credible Intervals for Keegan Bradley:
The prior probability that Bradley's average score lies between 3.5560 and 9.0729 is .95
The posterior probability that Bradley's average score lies between 8.4956 and 11.9577 is .95

We note that Sungjae Im has a markedly better posterior distribution than Brendan Steele with hardly any overlap, giving strong evidence that Im's average score is higher than Steele's.

To predict each golfer's likely score in the upcoming tournament, I computed the posterior-predictive distributions, and then compared Sungjae Im's distribution to those of the other golfers using 95% predictive intervals and calculating the area underneath the curve to the left of 0 (indicating Im would tie or win).

Predictive interval comparing Sungjae Im and Lee Westwood:
The predicted probability that the difference in Im and Westwood's scores in the Honda Classic lies between -10 and 7 is .95 and the predicted probability Im's score is equal or lower is 0.6030

Predictive interval comparing Sungjae Im and Brendan Steele:
The predicted probability that the difference in Im and Steele's scores in the Honda Classic lies between -12 and 5 is .95 and the predicted probability Im's score is equal or lower is 0.7298

Predictive interval comparing Sungjae Im and Keegan Bradley:
The predicted probability that the difference in Im and Bradley's scores in the Honda Classic lies between -11 and 6 is .95 and the predicted probability Im's score is equal or lower is 0.6868

These predictive interval results are illustrated graphically in Figures 6-11 below.

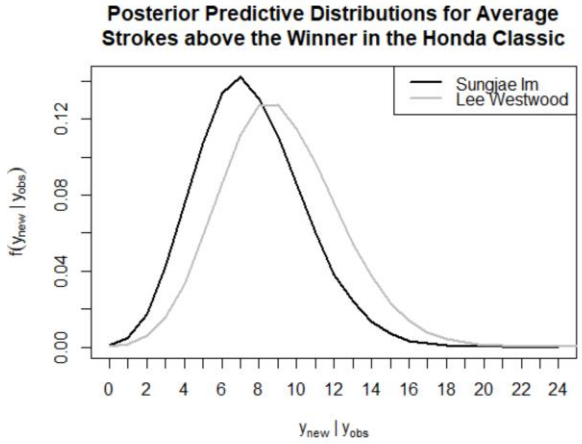

Figure 6. Comparison of posterior-predictive distributions for Im (black) and Westwood (gray).

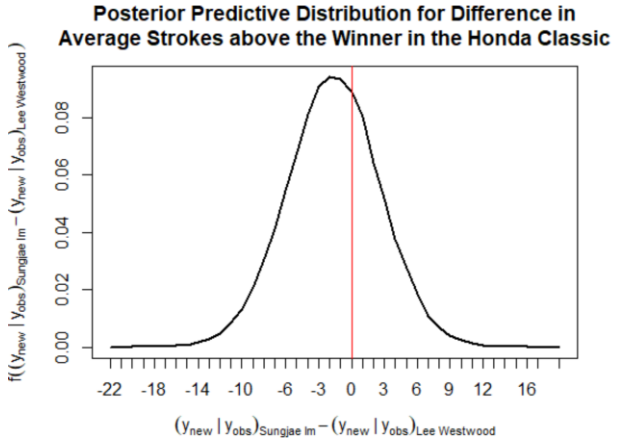

Figure 7. Posterior-predictive for difference between Im and Westwood's average strokes.

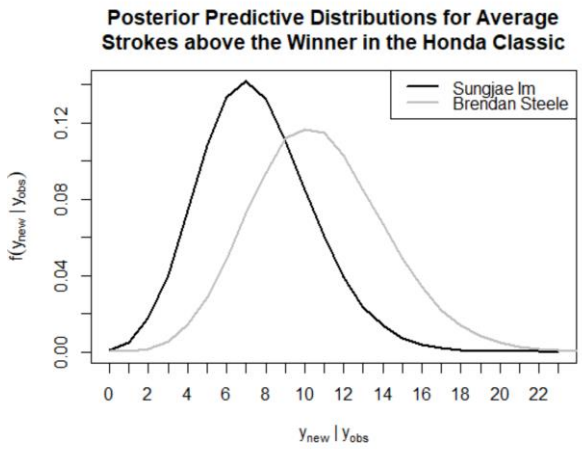

Figure 8. Comparison of posterior-predictive distributions for Im (black) and Steele (gray).

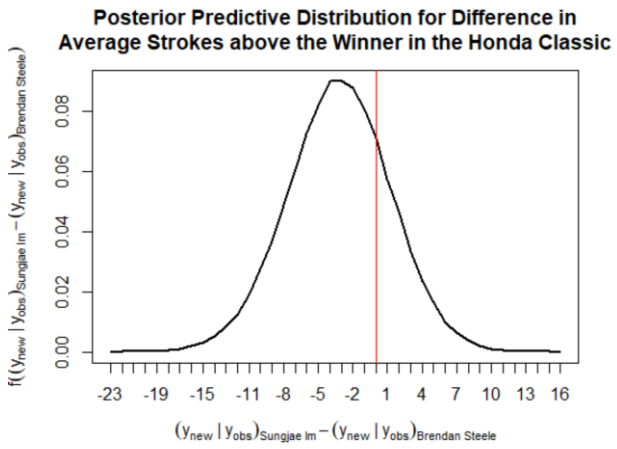

Figure 9. Posterior-predictive for difference between Im and Steele's average strokes.

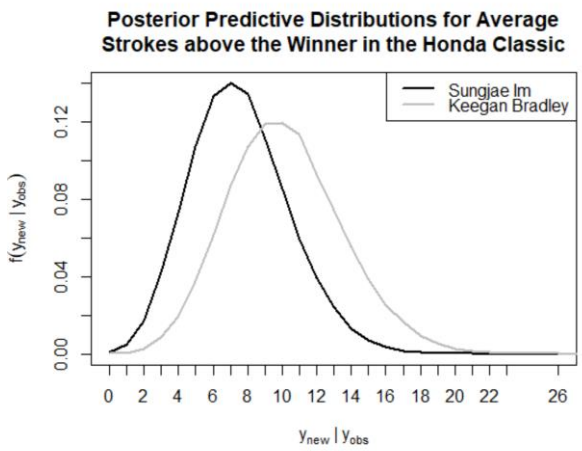

Figure 10. Comparison of posterior-predictive distributions for Im (black) and Bradley (gray).

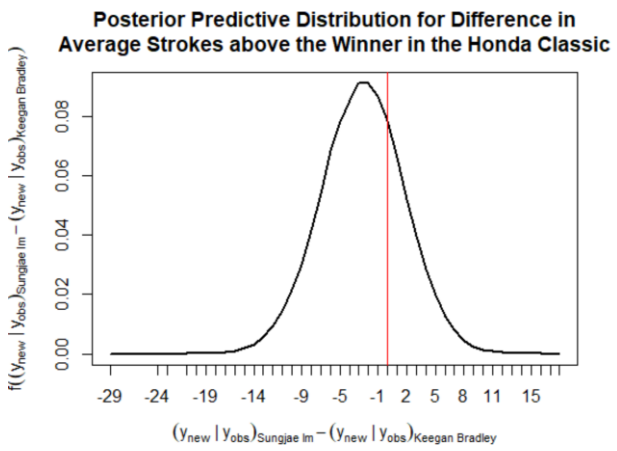

Figure 11. Posterior-predictive for difference between Im and Bradley's average strokes.

The posterior-predictive distributions are much wider than the posterior distributions due to the greater uncertainty inherent in examining a single tournament. Although all three predictive intervals favor Sungjae Im, 0 is included in every interval and the predictive probability that he wins against any of the other three golfers is never greater than 0.75, allowing for no certain prediction of Im's success. Due to their width, the predictive intervals captured the actual differences in the players' scores in the Honda Classic (excluding Lee Westwood, who was cut).

Discussion

Overall, the model was successful in predicting the relative ordering of the given golfers and demonstrates the validity of the Gamma-Poisson in modeling golf tournaments. This success is partly due to the golf's unique property that competing players do not affect each other's scores—an analysis of this type may not yield similar results among other sports.

Despite its success, my analysis had a few shortcomings as well. One such drawback is that the predictive intervals were too wide to confidently predict a given golfer's success. Although adding more observed data would narrow these intervals, data from before 2019 has less bearing on current ability and could negatively impact the predictions. Another shortcoming was the inaccuracy of the high betting odds favoring Lee Westwood. This could be improved by taking the average of betting odds from multiple sources rather than the single source I used.

Dylan Webb
4/13/2021

Appendices

Appendix A – Data Scraping Code from Random Forest Model

See golfdatahandler.py at https://github.com/dylanwebbc/GolfPredictionModel

Appendix B – Gamma-Poisson Analysis in R

```
#Import data files
golf <- read.csv("golf.csv")
prediction.rf <- read.csv("prediction_rf.csv", fileEncoding="UTF-8-BOM")



#Create yobs vector from golf.csv
yobs <- c()
for (name in prediction.rf$Name) {
  individual.data <- subset(golf, Name == name, select = c(Name, Score))
  yobs <- rbind(yobs, tail(individual.data, 10))
}



#Compare mean and variance of golf scores and adjusted bettings odds
plot(table(golf$Score), xlab = "Strokes Above the Winner", ylab = "Density",
     main = "Distribution of Scores Over All Tournaments Since 2019")

odds.all <- read.csv("odds.csv", fileEncoding = "UTF-8-BOM")
adjusted.odds <- as.integer(sqrt(odds.all$Odds)/6 - 5)
hist(adjusted.odds, breaks = 10, main = "Histogram of Adjusted Odds", xlab =
"Adjusted Odds")



#Define predictive distribution for gamma-poisson
pred.dist <- function(a, b, ynew) {

  lp <- -log(factorial(ynew)) + a*log(b) - lgamma(a) +
    lgamma(ynew + a) - (ynew + a)*log(b + 1)

  exp(lp)
}



#Calculate predicted score for each player
PredictedScore <- c()
prior.params <- c()
post.params <- c()
for (name in prediction.rf$Name) {
  individual.data <- subset(yobs, Name == name, select = Score)
```

```
  a <- 3*as.integer(sqrt(prediction.rf$Odds[match(name, prediction.rf$Name)])
/6 - 5)
  b <- 3
  prior.params <- cbind(prior.params, rbind(a, b))

  astar <- a + sum(individual.data)
  bstar <- b + length(t(individual.data))
  post.params <- cbind(post.params, rbind(astar, bstar))

  PredictedScore <- rbind(PredictedScore,
                          which.max(pred.dist(astar, bstar, 0:30)) - 1)
}



#Combine prediction results into one dataframe
prediction.gp <- data.frame(PredictedScore)
prediction.gp$Name <- prediction.rf$Name
prediction.gp <- prediction.gp[order(PredictedScore),]
print(prediction.gp)

predicted.winner <- levels(droplevels(prediction.gp[1, "Name"]))

predicted.winner.a <- prior.params[1, which(prediction.rf$Name == predicted.w
inner)]
predicted.winner.b <- prior.params[2, which(prediction.rf$Name == predicted.w
inner)]

predicted.winner.astar <- post.params[1, which(prediction.rf$Name == predicte
d.winner)]
predicted.winner.bstar <- post.params[2, which(prediction.rf$Name == predicte
d.winner)]



#Print summary statistics for each player and plot
boxplot.data <- c()
for (name in prediction.gp[1:4, "Name"]) {
  individual.data <- subset(yobs, Name == name, select = Score)$Score
  index <- match(name, prediction.rf$Name)
  print(name)
  print(summary(individual.data))
  cat("Adjusted odds:", prior.params[1, index][[1]]/3, "\n\n")
  boxplot.data <- cbind(boxplot.data, individual.data)
}

boxplot(boxplot.data, names = prediction.gp[1:4, "Name"],
        main = "Distribution of Last Ten Standardized Scores for Four Golfers
")
```

```
#Plot priors
th <- seq(0, 15, length.out = 1001)
plot(th, dgamma(th, shape = predicted.winner.a, rate = predicted.winner.b),
     type = 'l', ylim = c(0,0.8), xlab = expression(theta), ylab = "Density",
     main = "Prior (above) vs Posterior (below) of\nAverage Strokes above the
Winner for Four Golfers")
cols = c("red", "blue", "green")
for (name in prediction.gp[2:4, "Name"]) {
  index <- match(name, prediction.rf$Name)
  lines(th, dgamma(th, shape = prior.params[1, index], rate = prior.params[2,
index]),
        type = 'l', col = cols[match(name, prediction.gp$Name) - 1])
}
legend(x = "topright", legend = prediction.gp[1:4, "Name"],
       col = c("black", "red", "blue", "green"), lty = 1:1)



#Plot posteriors
th <- seq(0, 15, length.out = 1001)
plot(th, dgamma(th, shape = predicted.winner.astar, rate = predicted.winner.b
star),
     type = 'l', ylim = c(0,0.8), xlab = expression(theta), ylab = "Density",
main = "")
cols = c("red", "blue", "green")
for (name in prediction.gp[2:4, "Name"]) {
  index <- match(name, prediction.rf$Name)
  lines(th, dgamma(th, shape = post.params[1, index], rate = post.params[2, i
ndex]),
        type = 'l', col = cols[match(name, prediction.gp$Name) - 1])
}



#Calculate prior and posterior credible intervals
for (name in prediction.gp[1:4, "Name"]) {
  index <- match(name, prediction.rf$Name)
  cat("\n\n", "Credible Intervals for", levels(droplevels(prediction.rf[index
, "Name"])))
  cat("\n", "Prior:", (qgamma(c(.025, .975), shape = prior.params[1, index],
prior.params[2, index])))
  cat("\n", "Posterior:", (qgamma(c(.025, .975), shape = post.params[1, index
], post.params[2, index])))
}



#Calculate prediction intervals and plot predicted difference in strokes
J = 100000
```

```
for (name in prediction.gp[2:4, "Name"]) {
  index <- match(name, prediction.rf$Name)

  predicted.winner.dist <- rpois(J, rgamma(J, shape = predicted.winner.astar,
                                           rate = predicted.winner.bstar))
  player.dist <- rpois(J, rgamma(J, shape = post.params[1, index],
                                 rate = post.params[2, index]))

  cat("\n", predicted.winner, "vs", levels(droplevels(prediction.rf[index, "N
ame"])), "\n")
  print(mean(predicted.winner.dist < player.dist))
  print(quantile(predicted.winner.dist - player.dist, probs = c(0.025, 0.975)
))

  xeq <- bquote(y[new]~"|"~y[obs])
  yeq <- bquote(f(.(xeq)))
  plot(table(predicted.winner.dist)/J, type = 'l',
       xlab = xeq, ylab = yeq,
       main = "Posterior Predictive Distributions for Average\nStrokes above
the Winner in the Honda Classic")
  lines(table(player.dist)/J, type = 'l', col = "gray")
  legend(x = "topright", legend = c(predicted.winner, name),
         col = c("black", "gray"), lty = 1:1, lwd = 2)

  eq <- bquote((y[new]~"|"~y[obs]))
  xeq <- bquote(.(eq)[.(predicted.winner)]-.(eq)[.(name)])
  yeq <- bquote(f(.(xeq)))
  plot(table(predicted.winner.dist - player.dist)/J, type = 'l',
       xlab = xeq, ylab = yeq,
       main = "Posterior Predictive Distribution for Difference in\nAverage S
trokes above the Winner in the Honda Classic")
  abline(v = 0, col = "red")
}
```