Temple University

SEI x Phillies 2024 Hackathon:

Estimating Strike Probabilities Through Umpire Tendencies

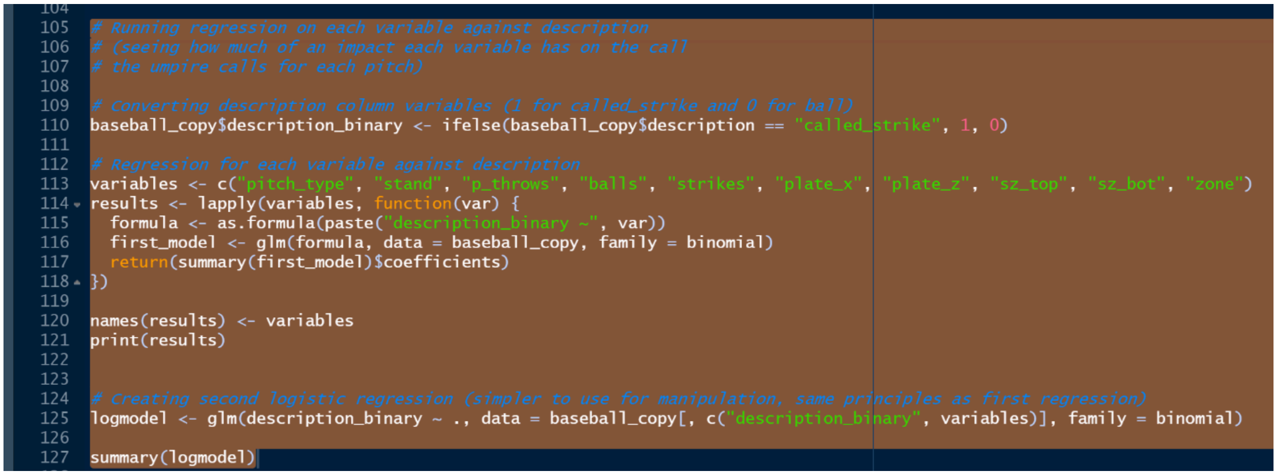
CODE INSTRUCTIONS

Kale Wiley, Gibson Hurst, Michael Mucciolo, & Cara Fredericks

April 24, 2024

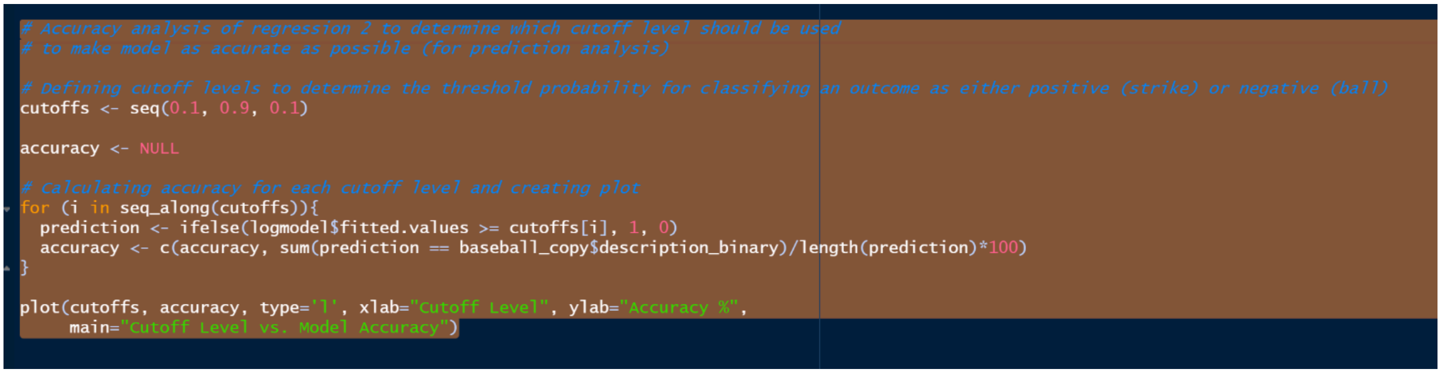
Instructions for running code:

1. Import necessary libraries:
    a. library(readxl)
    b. library(dplyr)
    c. library(tidyr)
    d. library(ggplot2)
2. Import Dataset (We kept the name the same):
    a. baseball_copy <- read_excel("baseball copy.xlsx")
3. Run regression code (highlighted chunk):



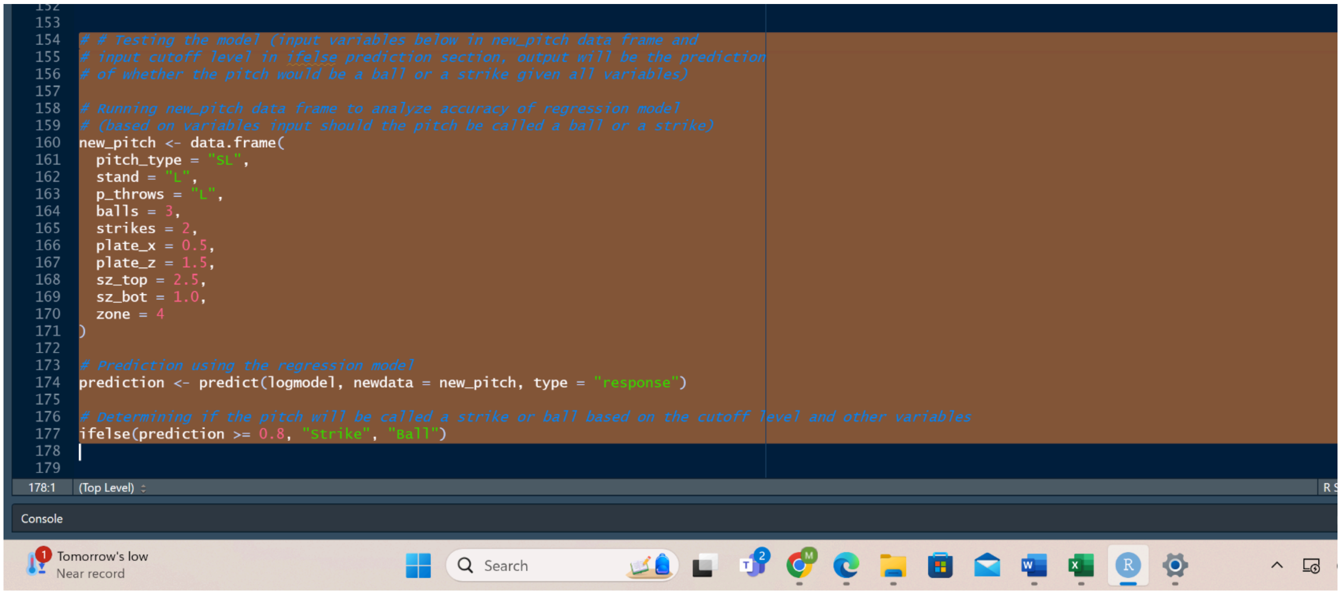
```
# Running regression on each variable against description
# (seeing how much of an impact each variable has on the call
# the umpire calls for each pitch)

# Converting description column variables (1 for called_strike and 0 for ball)
baseball_copy$description_binary <- ifelse(baseball_copy$description == "called_strike", 1, 0)

# Regression for each variable against description
variables <- c("pitch_type", "stand", "p_throws", "balls", "strikes", "plate_x", "plate_z", "sz_top", "sz_bot", "zone")
results <- lapply(variables, function(var) {
  formula <- as.formula(paste("description_binary ~", var))
  first_model <- glm(formula, data = baseball_copy, family = binomial)
  return(summary(first_model)$coefficients)
})

names(results) <- variables
print(results)


# Creating second logistic regression (simpler to use for manipulation, same principles as first regression)
logmodel <- glm(description_binary ~ ., data = baseball_copy[, c("description_binary", variables)], family = binomial)

summary(logmodel)
```

4. Run accuracy analysis:



```
# Accuracy analysis of regression 2 to determine which cutoff level should be used
# to make model as accurate as possible (for prediction analysis)

# Defining cutoff levels to determine the threshold probability for classifying an outcome as either positive (strike) or negative (ball)
cutoffs <- seq(0.1, 0.9, 0.1)

accuracy <- NULL

# Calculating accuracy for each cutoff level and creating plot
for (i in seq_along(cutoffs)){
  prediction <- ifelse(logmodel$fitted.values >= cutoffs[i], 1, 0)
  accuracy <- c(accuracy, sum(prediction == baseball_copy$description_binary)/length(prediction)*100)
}

plot(cutoffs, accuracy, type='l', xlab="Cutoff Level", ylab="Accuracy %",
     main="Cutoff Level vs. Model Accuracy")
```

5. Testing model (adjust variables however you please):



```r
# # Testing the model (input variables below in new_pitch data frame and
# input cutoff level in ifelse prediction section, output will be the prediction
# of whether the pitch would be a ball or a strike given all variables)

# Running new_pitch data frame to analyze accuracy of regression model
# (based on variables input should the pitch be called a ball or a strike)
new_pitch <- data.frame(
  pitch_type = "SL",
  stand = "L",
  p_throws = "L",
  balls = 3,
  strikes = 2,
  plate_x = 0.5,
  plate_z = 1.5,
  sz_top = 2.5,
  sz_bot = 1.0,
  zone = 4
)

# Prediction using the regression model
prediction <- predict(logmodel, newdata = new_pitch, type = "response")

# Determining if the pitch will be called a strike or ball based on the cutoff level and other variables
ifelse(prediction >= 0.8, "Strike", "Ball")

```