Descriptive Statistics Review and dplyr
Outline for today

Better know a sport: Pete Rose

Review of multiple linear regression

Manipulating data with dplyr

Worksheet 5
Review
Regression

Regression is a method of using one variable to predict the value of a second variable.

In linear regression we fit a line to the data, called the regression line.

$$\hat{y} = a + b \cdot x$$
The residual at a data value is the difference between the observed ($y$) and predicted value ($\hat{y}$) of the response variable.

\[
\text{Residual} = \text{Observed} - \text{Predicted} = y - \hat{y}
\]
Measuring goodness of fit

Residual

We can measure how well the line fits the data using the mean squared error (MSE):

$$\text{Residual} = \text{Observed} - \text{Predicted} = y - \hat{y}$$

![Graph showing the relationship between runs and batting average with residual highlighted.]

Least square regression line minimizes the MSE.
Multiple regression

Multiple regression makes predictions ($\hat{y}$) using multiple variables

Batting average:
$$BA = \frac{[(1) \cdot 1B + (1) \cdot 2B + (1) \cdot 3B + (1) \cdot HR]/AB}$$

Slugging percentage:
$$Slug = \frac{[(1) \cdot 1B + (2) \cdot 2B + (3) \cdot 3B + (4) \cdot HR]/AB}$$

Optimal statistic:
$$OPT = w_1 \cdot BB + w_2 \cdot HBP + w_3 \cdot 1B + w_4 \cdot 2B + w_5 \cdot 3B + w_6 \cdot HR + w_0$$
What are the optimal weights?

We can find the optimal weights \((w_i \text{'s})\) by minimizing the MSE:

\[
\hat{y} = w_1 \cdot BB + w_2 \cdot HBP + w_3 \cdot 1B + w_4 \cdot 2B + w_5 \cdot 3B + w_6 \cdot HR + w_0
\]

> ln.fit <- lm(R ~ BB + HBP + H + X2B + X3B + HR, data = team.batting.162)

The R model ln.fit contains the weights \(w_i \text{'s}\) for making predictions \(\hat{y}\)
What is the best statistic we can create?

Matthew came up with the MBS (Matthew’s batting statistic):

```
linear_model <- lm(R~BB+H+X2B+X3B+HR+BRA+OBP+OPS+Ghome+attendance+AB, data = team.batting.162)

RMSE:
  sqrt(mean(linear_model$residuals^2))
  23.49
```

Did anyone do better?

Who thinks MBS is the best statistic ever?
Overfitting

One problem with our optimal statistic is that we used the same data to fit our model (find the $w_i$’s) as we did to evaluate whether it was a good fit.

Thus it is possible that our weights we found we too tailored to the data at hand and our estimate of

Fitting a model to precisely to the data at hand in such a way that it does not generalize to new data is called overfitting.
Overfitting

One should always use different data when fitting and evaluating a model!
Cross-validation is a method for assessing the goodness of a model in a way that can avoid overfitting.

What we do is build the model on one set of data, called the **training set**
- i.e., find the coefficients on one set of data

Then we evaluate whether the model fits well on a second set of data, called the **test set**

If the model is truly good, we should get good predictions on the test set
- i.e., a small RMSE on the test set
Matthew’s Batting Statistic (MBS)

Randomly split the data:
• ½ if the data is in the training set
• ½ of the data is in the test set

Fit the model using the training data for the MBS:
• ln.model <- lm(R ~ BB+H+X2B+ ..., data = training.data)

Make predictions on the test data
• predicted.yhats <- predict(ln.model, newdata = test.data)
• cross.validated.RMSE <- sqrt(mean((predicted.yhats - test.data$R)^2))

MSE for predictions made using the same training data x’s and y’s: 23.18

MSE for predictions made on the test data x’s and y’s: 24.63
Manipulating data with dplyr

R packages add additional functions to R
- require(‘package.name’)

**dplyr** is a very useful package for manipulating data frames
- require(‘dplyr’)

There are several very useful functions in the dplyr package including:
- filter()
- select()
- mutate()
- arrange()
- group_by()
- summarize()

All these functions take a data frame as input and return a data frame as output
filter()

The filter() function returns a subset of the cases
  • i.e., subset of the rows of a data frame

Example:
  • all.data <- get.Lehman.batting.data()
  • red.sox.data <- filter(all.data, teamID == "BOS")

Question:
  • How could we get all players who had exactly 300 PA?
  • max.300PA <- filter(all.data, PA == 300)
select()

The `select()` function returns a subset of the variables
  • i.e., subset of the columns of a data frame

Example:
  • `all.data <- get.Lehman.batting.data()`
  • `data.hits.and.walks <- select(all.data, H, BB)`

Question:
  • How could we only home runs and doubles?
  • `data.hits.and.walks <- select(all.data, HR, X2B)`
mutate()

The mutate() function adds new variables to a data frame from variables that are already in the data frame
  • i.e., creates new columns from old columns

Example:
  • data.with.1B <- mutate(all.data, X1B = H – X2B – X3B - HR)

Question:
  • How can we add BRA (which is OBP * OPS) to our data frame?
  • data.with.BRA <- mutate(all.data, BRA = OBP * Slug)
arrange()

The arrange() function can be used to sort the cases in a data frame in order
   • i.e., arrange the values of a variable from smallest to largest

Example:
   • all.data <- get.Lehman.batting.data()
   • data.ordered.by.HR <- arrange(all.data, desc(HR))

Question:
   • How could order data by the fewest strike outs?
   • data.ordered.by.SO <- arrange(all.data, desc(SO))
group_by()

The `group_by()` function assigns categorical variables to groups

- by itself it does nothing, but it is useful in conjunction with the `summarize()` function as described on the next slide

Example:

- `data.team.grouped <- group_by(all.data, teamID)`

Question:

- How can group data by year?
- `data.year.grouped <- group_by(all.data, yearID)`
summarize()

The `summarize()` function reduces the data based on the grouping assigned by the `group_by()` function
  • i.e., it takes many cases and create summary statistics from these cases separately for each grouping.

Example:
  • `data.team.grouped <- group_by(all.data, teamID)`
  • `team.total.hits <- summarize(data.team.grouped, sum(H, na.rm = TRUE))`

Question:
  • How can we get the total number of hits as a function of the year?
    • `data.year.grouped <- group_by(all.data, yearID)`
    • `summarize(data.year.grouped, sum(H, na.rm = TRUE))`
Summary of some of what we have learned about descriptive statistics

**Descriptive statistics:** median, mean, standard deviation, percentiles, five number summary, range, interquartile range, z-scores, correlation

**Plots:** bar plots, pie charts, histograms, box plots, scatter plots

**Regression:** linear regression, multiple regression, residuals, RMSE, overfitting

A lot about baseball and analyzing data in R
Please get started on this worksheet early, some of the questions on the worksheet might be challenging!