Lecture 4: Prediction and model selection in MLB

Skidmore College

Intro/review

```r
library(tidyverse); library(Lahman); options(digits = 4)
set.seed(0)
Pitching %>%
  select(playerID, yearID, W, L, H, BB, SO, BFP, ERA) %>%
  sample_n(5)
```

```
##    playerID yearID  W  L   H BB  SO  BFP  ERA
## 1 seguidi01   1964  8 17 219 94 155  947 4.56
## 2 wagnery01   2006  3  3  36 15  20  141 4.70
## 3 johnske01   1949  0  1  29 35  18  160 6.42
## 4 burtoja01   2012  3  2  41 16  55  245 2.18
## 5 newcodo01   1950 19 11 258 75 130 1101 3.70
```

Preliminary questions

Write code to

1. Filter pitchers with at least 500 batters faced in a season that came in the year 2000 or after
2. Make a new variable, `bb_rate`, to represent the percent of batters faced that each pitcher walks
3. Identify the players with the most wins in a season in the data set
4. Identify the players with the most total wins across the data set
5. Find the team whose pitchers allowed the most home runs between 2010 and 2019

Multivariate regression

Model:

$$y_i = \beta_0 + \beta_1 * x_{i1} + \beta_2 * x_{i2} + \ldots + \beta_{p-1} * x_{i,p-1} + \epsilon_i$$

Assumptions:

- $\epsilon_i \sim N(0, \sigma^2)$
- ϵ_i,$\epsilon_{i'}$ independent for all i, i'
- Linear relationship between y and x

Estimated model:

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 * x_{i1} + \hat{\beta}_2 * x_{i2} + \ldots + \hat{\beta}_{p-1} * x_{i,p-1}$$

How to pick the best model

0. Scatter plots
1. R-squared, R-squared adjusted, p-value cutoffs (x)
2. AIC
3. MAE/MSE
4. Check model assumptions

MLB pitcher prediction

```r
Pitching <- Pitching %>%
  filter(yearID >= 2000, BFP >= 500) %>%
  mutate(K_rate = SO/BFP,
         BB_rate = BB/BFP,
         HR_rate = HR/BFP,
         FIP = ((13*HR) + 5*(H - HR)  + 3*(BB + HBP) - 2*SO)/(IPouts))

fit_pitcher_1 <- lm(ERA ~ K_rate + BB_rate + lgID + BK, data = Pitching
fit_pitcher_2 <- lm(ERA ~ K_rate + BB_rate + lgID, data = Pitching)
```

Write the multiple regression model:

MLB pitcher prediction

```r
library(broom)
tidy(fit_pitcher_1)  ### alternatively, use summary(fit.pitcher)
```

```
## # A tibble: 5 x 5
##   term        estimate std.error statistic   p.value
##   <chr>          <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)   5.05      0.0807     62.6  0.
## 2 K_rate       -9.38      0.305     -30.8  4.96e-176
## 3 BB_rate      12.4       0.718      17.2  8.01e- 63
## 4 lgIDNL       -0.182     0.0302     -6.01 2.09e-  9
## 5 BK           -0.0319    0.0177     -1.80 7.23e-  2
```

Write the estimated multiple regression model

Which model is best?

```
summary(fit_pitcher_1)$r.squared
```

```
## [1] 0.3566
```

```
summary(fit_pitcher_2)$r.squared
```

```
## [1] 0.3557
```

```
summary(fit_pitcher_1)$adj.r.squared
```

```
## [1] 0.3555
```

```
summary(fit_pitcher_2)$adj.r.squared
```

```
## [1] 0.3549
```

AIC

```r
AIC(fit_pitcher_1)
```

```
## [1] 5425
```

```r
AIC(fit_pitcher_2)
```

```
## [1] 5427
```

What is AIC?

What does AIC say about these two models?

Setting up next year

```
Pitching <- Pitching %>%
  arrange(playerID, yearID) %>%
  mutate(K_rate_next = lead(K_rate, 1))
```

Why predict next year?

Steps to model selection

1. Fit plausible models
2. Contrast AIC, pick lowest performing model. If different models have similar AICs, err on the side of parsimony
3. Consider prediction errors using MSE and MAE

Step 1: fit plausible models

```
fit_next_yr_1 <- lm(K_rate_next ~ K_rate, data = Pitching)
fit_next_yr_2 <- lm(K_rate_next ~ K_rate + HR_rate, data = Pitching)
fit_next_yr_3 <- lm(K_rate_next ~ K_rate + HR_rate + lgID, data = Pitch
fit_next_yr_4 <- lm(K_rate_next ~ K_rate + FIP, data = Pitching)
fit_next_yr_5 <- lm(K_rate_next ~ K_rate + BB_rate, data = Pitching)
```

Step 2: AIC to get started

```r
AIC(fit_next_yr_1)
```

```
## [1] -8705
```

```r
AIC(fit_next_yr_2)
```

```
## [1] -8711
```

```r
AIC(fit_next_yr_3)
```

```
## [1] -8723
```

```r
AIC(fit_next_yr_4)
```

```
## [1] -8703
```

```r
AIC(fit_next_yr_5)
```

```
## [1] -8707
```

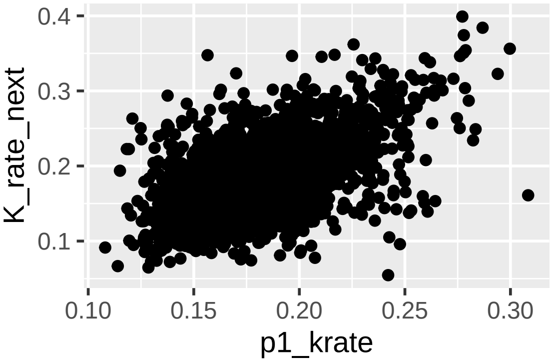
Coding best estimates for future performance

```r
Pitching <- Pitching %>%
  mutate(p1_krate = predict(fit_next_yr_1, Pitching),
         p2_krate = predict(fit_next_yr_2, Pitching),
         p3_krate = predict(fit_next_yr_3, Pitching),
         p4_krate = predict(fit_next_yr_4, Pitching),
         p5_krate = predict(fit_next_yr_5, Pitching))
head(Pitching) %>% select(K_rate_next, p1_krate:p5_krate)
```

```
##   K_rate_next p1_krate p2_krate p3_krate p4_krate p5_krate
## 1      0.1662   0.1522   0.1519   0.1547   0.1522   0.1543
## 2      0.1662   0.1729   0.1725   0.1756   0.1729   0.1764
## 3      0.1992   0.1729   0.1690   0.1655   0.1730   0.1720
## 4      0.1627   0.1921   0.1961   0.1938   0.1921   0.1921
## 5      0.2034   0.1709   0.1744   0.1718   0.1709   0.1726
## 6      0.1839   0.1946   0.1872   0.1834   0.1947   0.1961
```

Visualizations of model predictions

```r
ggplot(data = Pitching, aes(p1_krate, K_rate_next)) +
  geom_point()
```



Metrics for accuracy

```
Pitching %>%
  filter(!is.na(K_rate_next)) %>%
  summarise(mae_p1 = mean(abs(p1_krate - K_rate_next)),
            mae_p2 = mean(abs(p2_krate - K_rate_next)),
            mae_p3 = mean(abs(p3_krate - K_rate_next)),
            mae_p4 = mean(abs(p4_krate - K_rate_next)),
            mae_p5 = mean(abs(p5_krate - K_rate_next)))
```

```
##    mae_p1  mae_p2  mae_p3  mae_p4 mae_p5
## 1 0.03055 0.03054 0.03048 0.03055 0.0305
```

Metrics for accuracy

```
Pitching %>%
  filter(!is.na(K_rate_next)) %>%
  summarise(mse_p1 = mean((p1_krate - K_rate_next)^2),
            mse_p2 = mean((p2_krate - K_rate_next)^2),
            mse_p3 = mean((p3_krate - K_rate_next)^2),
            mse_p4 = mean((p4_krate - K_rate_next)^2),
            mse_p5 = mean((p5_krate - K_rate_next)^2))
```

```
##     mse_p1   mse_p2   mse_p3   mse_p4   mse_p5
## 1 0.001617 0.001612 0.001603 0.001617 0.001615
```