# Lecture 4: Prediction and model selection in MLB

Skidmore College

# 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:

- $\epsilon_i \sim N(0, \sigma^2)$
- $\epsilon_i$,$\epsilon_{i'}$ 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 or p-value cutoffs (xx)
2. R-squared adjusted (x)
3. AIC
4. MAE/MSE
5. Check model assumptions

# MLB pitcher prediction

```
library(tidyverse); library(Lahman); options(digits = 4)
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

```
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)   5.10      0.0816     62.4  0.
## 2 K_rate       -9.86      0.315     -31.3  9.04e-180
## 3 BB_rate      12.6       0.722      17.5  2.51e- 64
## 4 lgIDNL       -0.170     0.0305     -5.58 2.64e-  8
## 5 BK           -0.0370    0.0179     -2.06 3.91e-  2
```

Write the estimated multiple regression model

# Which model is best?

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

```
## [1] 0.371
```

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

```
## [1] 0.3699
```

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

```
## [1] 0.3699
```

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

```
## [1] 0.369
```

# AIC

```
AIC(fit_pitcher_1)
```

```
## [1] 5146
```

```
AIC(fit_pitcher_2)
```

```
## [1] 5149
```

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))
```

1. Fit plausible models to predict future data
2. Contrast fit statistics such as AIC
3. Prediction accuracy 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

```
AIC(fit_next_yr_1)
```

```
## [1] -8422
```

```
AIC(fit_next_yr_2)
```

```
## [1] -8426
```

```
AIC(fit_next_yr_3)
```

```
## [1] -8435
```

```
AIC(fit_next_yr_4)
```

```
## [1] -8421
```

```
AIC(fit_next_yr_5)
```

```
## [1] -8421
```

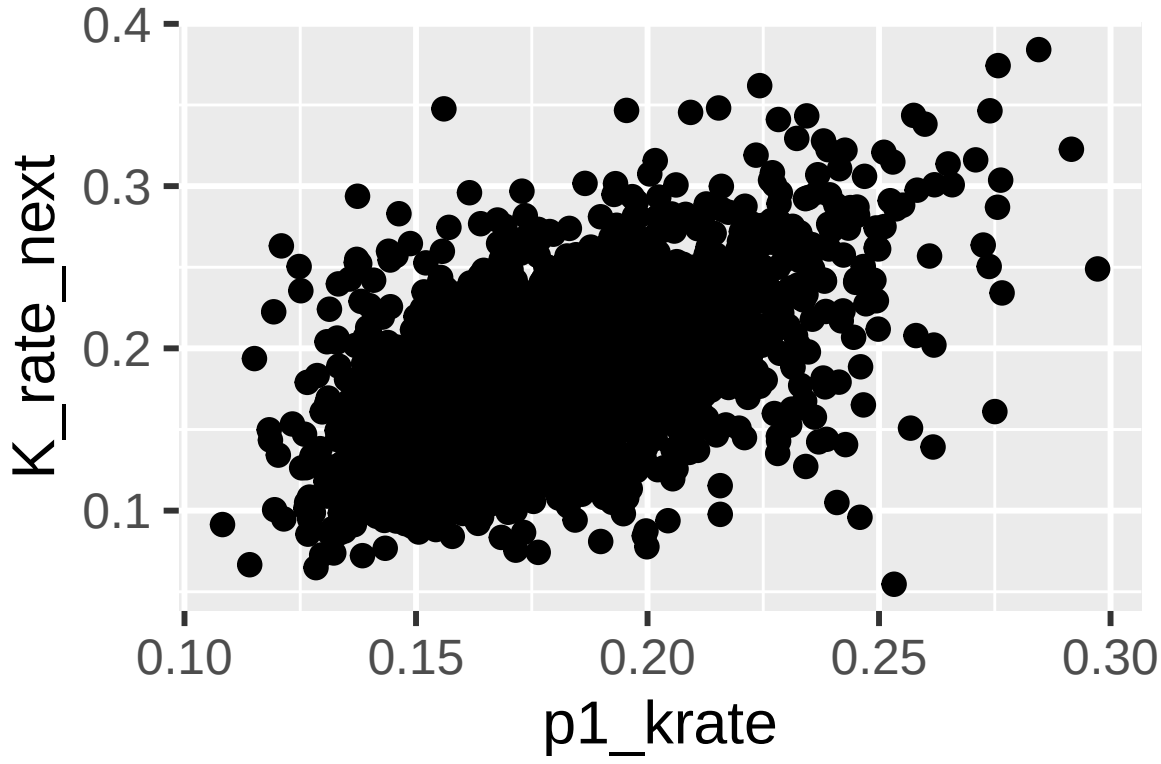
# Coding best estimates for future performance

```
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   0.1518   0.1515   0.1541   0.1516   0.1527
## 2      0.1662   0.1722   0.1718   0.1746   0.1722   0.1738
## 3      0.1992   0.1722   0.1688   0.1656   0.1725   0.1718
## 4      0.1627   0.1911   0.1944   0.1922   0.1911   0.1911
## 5      0.2034   0.1702   0.1731   0.1707   0.1701   0.1710
## 6      0.1839   0.1935   0.1872   0.1838   0.1939   0.1942
```

# Visualizations of model predictions

```
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 0.03015 0.03013 0.03007 0.03014 0.03013
```

# 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 0.001562 0.001559 0.001551 0.001562 0.001562
```