Tampa Bay Rays R&D Internship

Dylan Zumar (Duke University Class of 2021)

R Markdown

```
dataset <- read.csv("battedBallData.csv")
dataset %>%
  group_by(hittype) %>%
  summarise(count = n())
```

```
## # A tibble: 5 x 2
##   hittype     count
##   <fct>       <int>
## 1 fly_ball    16722
## 2 ground_ball 33239
## 3 line_drive  18166
## 4 popup        5246
## 5 U               2
```

```
dataset <- dataset %>%
  filter(hittype != "U")
```

There were two instances in the data set where the hit type was 'U' (presumably "unknown"), which I removed from the data set since there is likely little information to be gleaned from those two cases.

Exploratory Data Analysis

Creating Boxplots of Speed and Angles for each Hit Type

In order to tell the differences between system A and system B, I will make boxplots for the angles and speeds measured by A and B split up for each hit type. This will hopefully isolate the flaws in A and B.

```
speedAgb <- dataset %>%
  filter(hittype == "ground_ball") %>%
  ggplot(mapping = aes(y = speed_A)) +
  geom_boxplot()
speedAld <- dataset %>%
  filter(hittype == "line_drive") %>%
  ggplot(mapping = aes(y = speed_A)) +
  geom_boxplot()
speedAfb <- dataset %>%
  filter(hittype == "fly_ball") %>%
  ggplot(mapping = aes(y = speed_A)) +
  geom_boxplot()
speedApu <- dataset %>%
  filter(hittype == "popup") %>%
  ggplot(mapping = aes(y = speed_A)) +
  geom_boxplot()
```

```
vangleAgb <- dataset %>%
  filter(hittype == "ground_ball") %>%
```

```
  ggplot(mapping = aes(y = vangle_A)) +
  geom_boxplot()
vangleAld <- dataset %>%
  filter(hittype == "line_drive") %>%
  ggplot(mapping = aes(y = vangle_A)) +
  geom_boxplot()
vangleAfb <- dataset %>%
  filter(hittype == "fly_ball") %>%
  ggplot(mapping = aes(y = vangle_A)) +
  geom_boxplot()
vangleApu <- dataset %>%
  filter(hittype == "popup") %>%
  ggplot(mapping = aes(y = vangle_A)) +
  geom_boxplot()
```

```
speedBgb <- dataset %>%
  filter(hittype == "ground_ball") %>%
  ggplot(mapping = aes(y = speed_B)) +
  geom_boxplot()
speedBld <- dataset %>%
  filter(hittype == "line_drive") %>%
  ggplot(mapping = aes(y = speed_B)) +
  geom_boxplot()
speedBfb <- dataset %>%
  filter(hittype == "fly_ball") %>%
  ggplot(mapping = aes(y = speed_B)) +
  geom_boxplot()
speedBpu <- dataset %>%
  filter(hittype == "popup") %>%
  ggplot(mapping = aes(y = speed_B)) +
  geom_boxplot()
```

```
vangleBgb <- dataset %>%
  filter(hittype == "ground_ball") %>%
  ggplot(mapping = aes(y = vangle_B)) +
  geom_boxplot()
vangleBld <- dataset %>%
  filter(hittype == "line_drive") %>%
  ggplot(mapping = aes(y = vangle_B)) +
  geom_boxplot()
vangleBfb <- dataset %>%
  filter(hittype == "fly_ball") %>%
  ggplot(mapping = aes(y = vangle_B)) +
  geom_boxplot()
vangleBpu <- dataset %>%
  filter(hittype == "popup") %>%
  ggplot(mapping = aes(y = vangle_B)) +
  geom_boxplot()
```

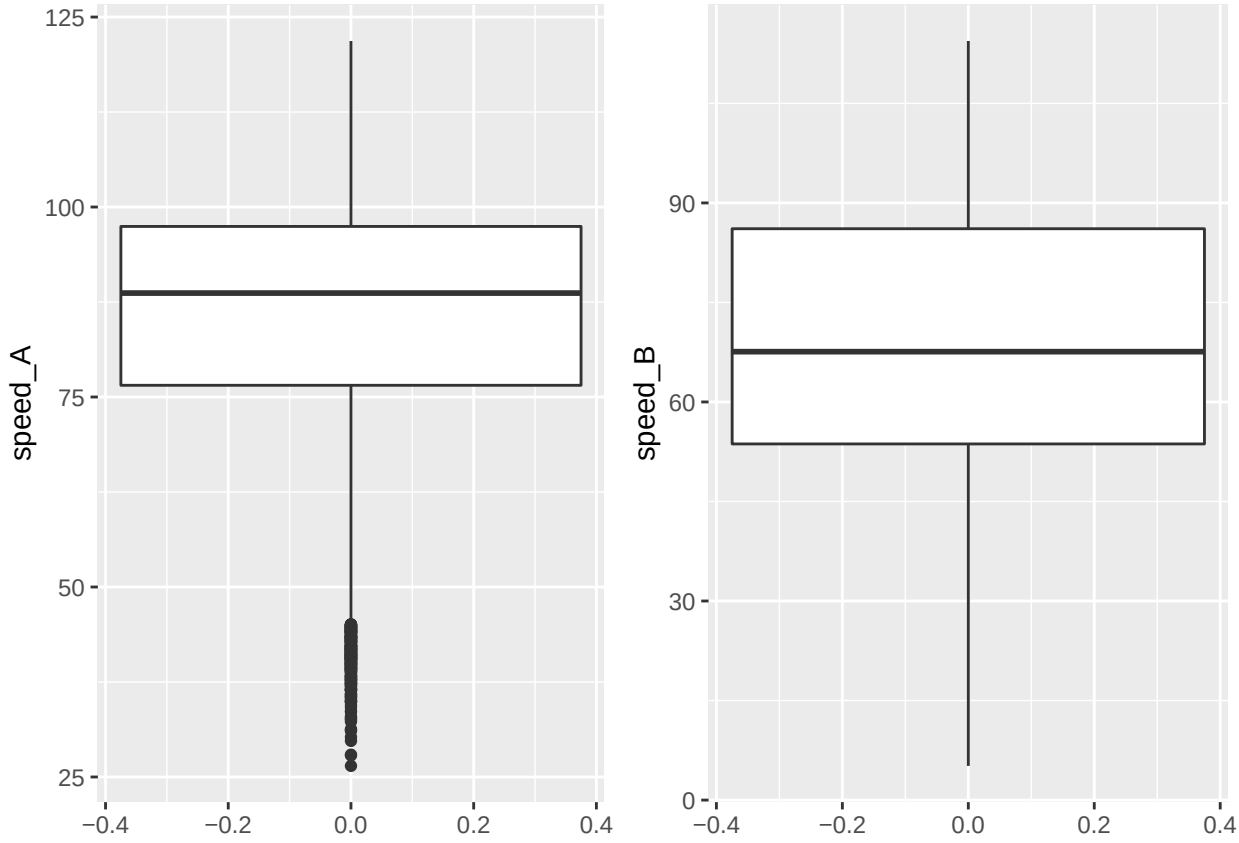
Displaying Boxplots Side-by-Side with 5-Number Summary

Ground balls

```
plot_grid(speedAgb, speedBgb)
```

```
## Warning: Removed 4591 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 681 rows containing non-finite values (stat_boxplot).
```




```
dataset %>%
  filter(hittype == "ground_ball") %>%
  summarise(minA = min(speed_A, na.rm = TRUE), q1A = quantile(speed_A, 0.25, na.rm = TRUE),
            meanA = mean(speed_A, na.rm = TRUE), q3A = quantile(speed_A, 0.75, na.rm = TRUE),
            maxA = max(speed_A, na.rm = TRUE))
```

```
##       minA      q1A   meanA      q3A     maxA
## 1 26.46182 76.54264 86.1458 97.44401 121.8475
```

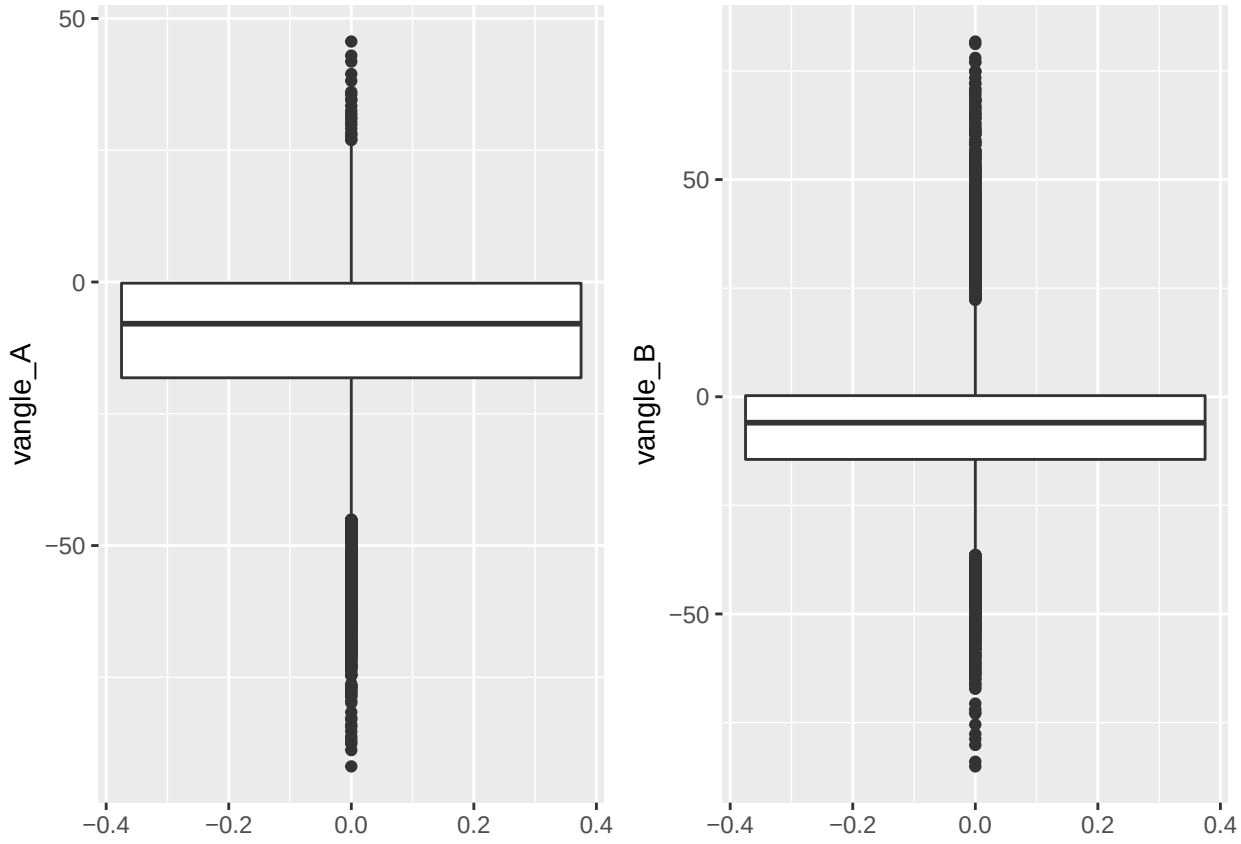
```
dataset %>%
  filter(hittype == "ground_ball") %>%
  summarise(minB = min(speed_B, na.rm = TRUE), q1B = quantile(speed_B, 0.25, na.rm = TRUE),
            meanB = mean(speed_B, na.rm = TRUE), q3B = quantile(speed_B, 0.75, na.rm = TRUE),
            maxB = max(speed_B, na.rm = TRUE))
```

```
##       minB      q1B    meanB      q3B     maxB
## 1 5.152318 53.67203 68.22395 86.10967 114.4034
```

```
plot_grid(vangleAgb, vangleBgb)
```

```
## Warning: Removed 4591 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 681 rows containing non-finite values (stat_boxplot).
```




```
dataset %>%
  filter(hittype == "ground_ball") %>%
  summarise(minA = min(vangle_A, na.rm = TRUE), q1A = quantile(vangle_A, 0.25, na.rm = TRUE),
            meanA = mean(vangle_A, na.rm = TRUE), q3A = quantile(vangle_A, 0.75, na.rm = TRUE),
            maxA = max(vangle_A, na.rm = TRUE))
```

```
##        minA       q1A     meanA        q3A     maxA
## 1 -91.89863 -18.16965 -10.76874 -0.2169038 45.63272
```

```
dataset %>%
  filter(hittype == "ground_ball") %>%
  summarise(minB = min(vangle_B, na.rm = TRUE), q1B = quantile(vangle_B, 0.25, na.rm = TRUE),
            meanB = mean(vangle_B, na.rm = TRUE), q3B = quantile(vangle_B, 0.75, na.rm = TRUE),
            maxB = max(vangle_B, na.rm = TRUE))
```

```
##        minB       q1B     meanB       q3B     maxB
## 1 -85.09093 -14.42102 -7.119327 0.2621273 81.77666
```

```
ttestspeedAgb <- as.data.frame(dataset %>%
  filter(hittype == "ground_ball") %>%
  select(speed_A))
ttestspeedBgb <- as.data.frame(dataset %>%
  filter(hittype == "ground_ball") %>%
```

```
  select(speed_B))
t.test(ttestspeedAgb, ttestspeedBgb, alternative = "two.sided", mu = 0,
       paired = FALSE, var.equal = TRUE, conf.level = 0.95)
```

```
##
##  Two Sample t-test
##
## data:  ttestspeedAgb and ttestspeedBgb
## t = 121.26, df = 61204, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  17.63217 18.21153
## sample estimates:
## mean of x mean of y
##  86.14580  68.22395
```

```
ttestvangleAgb <- as.data.frame(dataset %>%
  filter(hittype == "ground_ball") %>%
  select(vangle_A))
ttestvangleBgb <- as.data.frame(dataset %>%
  filter(hittype == "ground_ball") %>%
  select(vangle_B))
t.test(ttestvangleAgb, ttestvangleBgb, alternative = "two.sided", mu = 0,
       paired = FALSE, var.equal = TRUE, conf.level = 0.95)
```

```
##
##  Two Sample t-test
##
## data:  ttestvangleAgb and ttestvangleBgb
## t = -32.41, df = 61204, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.870110 -3.428715
## sample estimates:
##  mean of x  mean of y
## -10.768739  -7.119327
```

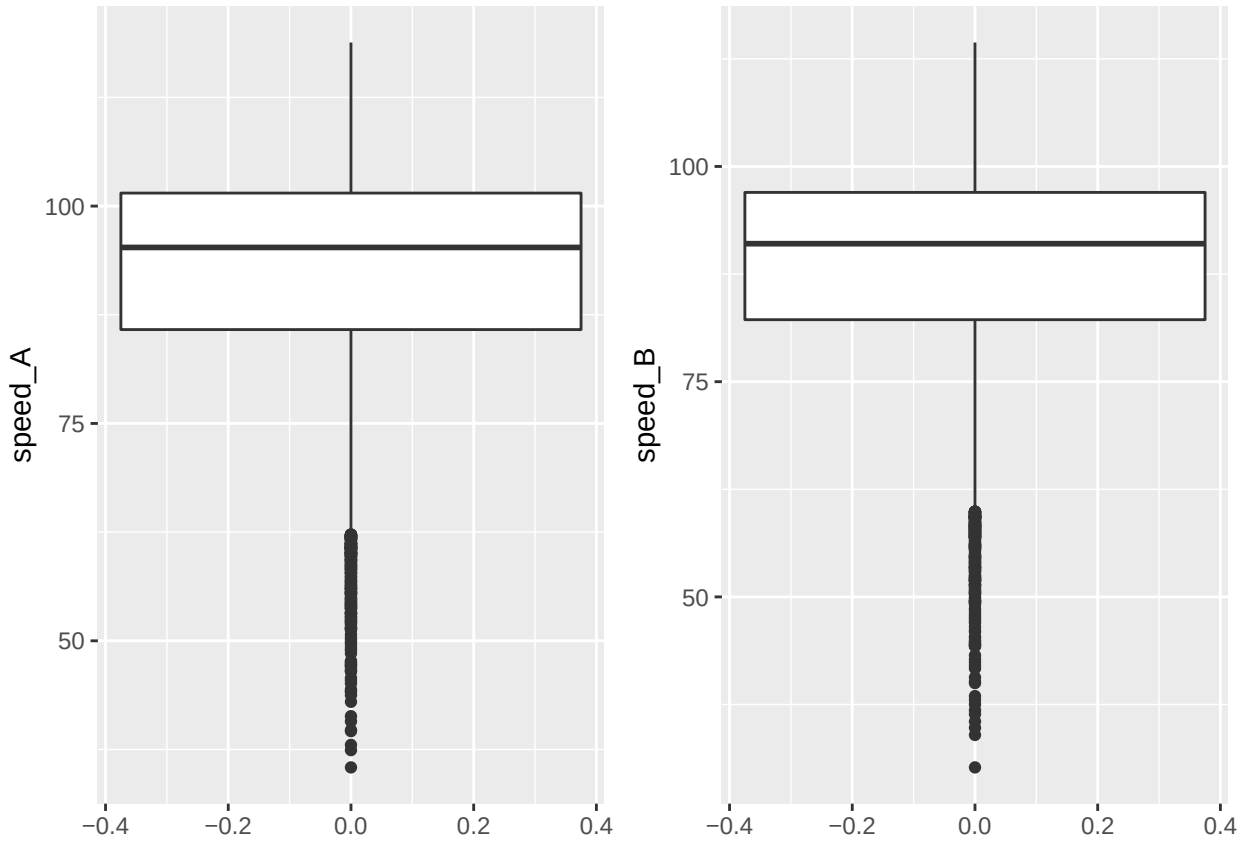
We can see that the distribution for the speed calculated by system A for ground balls is very different from the distribution for the speed calculated by system B. However, the angles calculated from the two systems are similar. Given that system A is suspected to be more accurate than system B, it appears that the speed for ground balls should be calculated using system A.

Line drives

```
plot_grid(speedAld, speedBld)
```

```
## Warning: Removed 258 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 130 rows containing non-finite values (stat_boxplot).
```




```
dataset %>%
  filter(hittype == "line_drive") %>%
  summarise(minA = min(speed_A, na.rm = TRUE), q1A = quantile(speed_A, 0.25, na.rm = TRUE), meanA
            = mean(speed_A, na.rm = TRUE), q3A = quantile(speed_A, 0.75, na.rm = TRUE), maxA =
              max(speed_A, na.rm = TRUE))
```

```
##       minA      q1A    meanA      q3A     maxA
## 1 35.44025 85.79469 92.83912 101.4946 118.8119
```

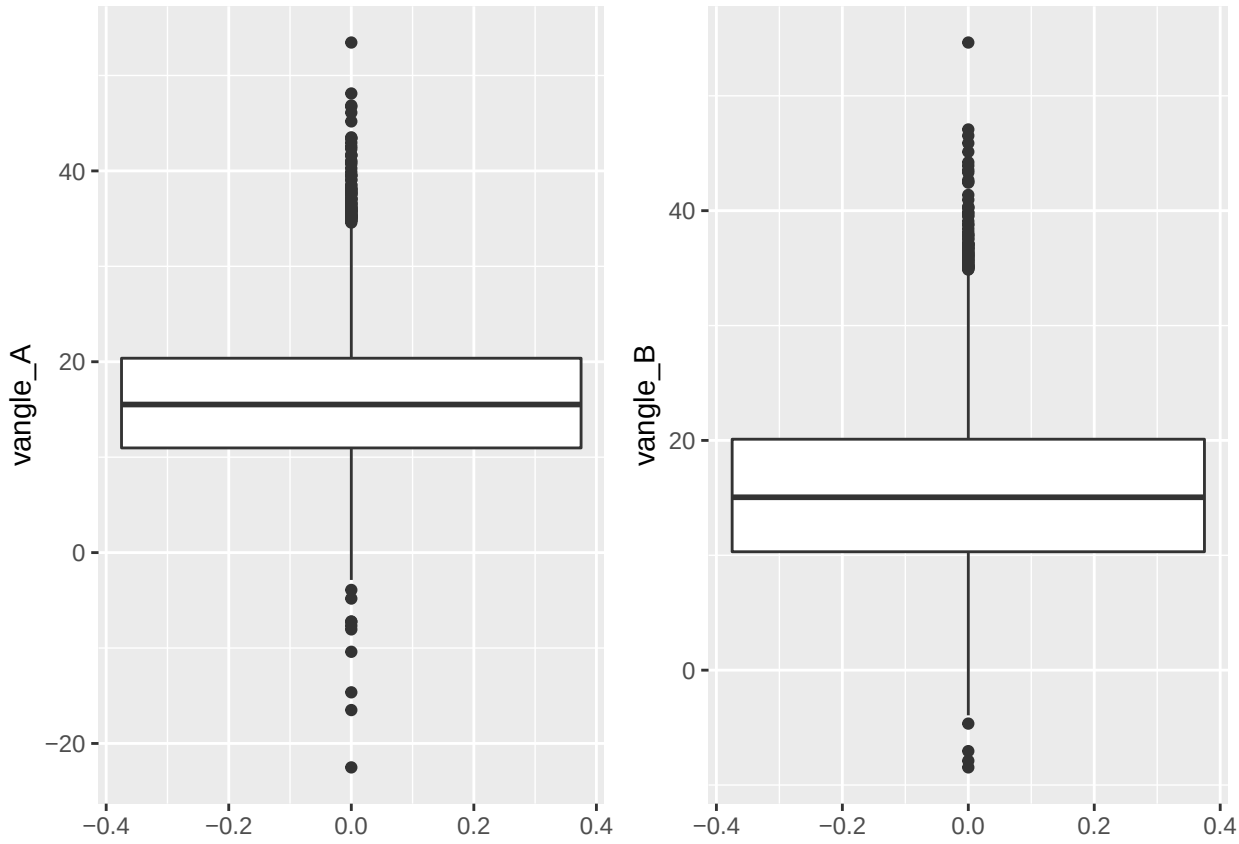
```
dataset %>%
  filter(hittype == "line_drive") %>%
  summarise(minB = min(speed_B, na.rm = TRUE), q1B = quantile(speed_B, 0.25, na.rm = TRUE), meanB
            = mean(speed_B, na.rm = TRUE), q3B = quantile(speed_B, 0.75, na.rm = TRUE), maxB =
              max(speed_B, na.rm = TRUE))
```

```
##       minB      q1B    meanB      q3B     maxB
## 1 30.19517 82.20025 88.74452 96.98768 114.3988
```

```
plot_grid(vangleAld, vangleBld)
```

```
## Warning: Removed 258 rows containing non-finite values (stat_boxplot).

## Warning: Removed 130 rows containing non-finite values (stat_boxplot).
```




```
dataset %>%
  filter(hittype == "line_drive") %>%
  summarise(minA = min(vangle_A, na.rm = TRUE), q1A = quantile(vangle_A, 0.25, na.rm = TRUE),
            meanA = mean(vangle_A, na.rm = TRUE), q3A = quantile(vangle_A, 0.75, na.rm = TRUE),
            maxA = max(vangle_A, na.rm = TRUE))
```

```
##        minA      q1A    meanA      q3A     maxA
## 1 -22.50977 10.96821 15.82258 20.37113 53.45513
```

```
dataset %>%
  filter(hittype == "line_drive") %>%
  summarise(minB = min(vangle_B, na.rm = TRUE), q1B = quantile(vangle_B, 0.25, na.rm = TRUE),
            meanB = mean(vangle_B, na.rm = TRUE), q3B = quantile(vangle_B, 0.75, na.rm = TRUE),
            maxB = max(vangle_B, na.rm = TRUE))
```

```
##        minB      q1B    meanB    q3B    maxB
## 1 -8.464513 10.30865 15.35495 20.105 54.6403
```

```
ttestspeedAld <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(speed_A))
ttestspeedBld <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(speed_B))
t.test(ttestspeedAld, ttestspeedBld, alternative = "two.sided", mu = 0,
       paired = FALSE, var.equal = TRUE, conf.level = 0.95)
```

```
##
##  Two Sample t-test
```

```
##
## data:  ttestspeedAld and ttestspeedBld
## t = 34.141, df = 35942, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  3.859532 4.329672
## sample estimates:
## mean of x mean of y
##  92.83912  88.74452
```

```
ttestvangleAld <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(vangle_A))
ttestvangleBld <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(vangle_B))
t.test(ttestvangleAld, ttestvangleBld, alternative = "two.sided", mu = 0,
       paired = FALSE, var.equal = TRUE, conf.level = 0.95)
```

```
##
##  Two Sample t-test
##
## data:  ttestvangleAld and ttestvangleBld
## t = 6.6143, df = 35942, p-value = 3.786e-11
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.3290589 0.6062099
## sample estimates:
## mean of x mean of y
##  15.82258  15.35495
```

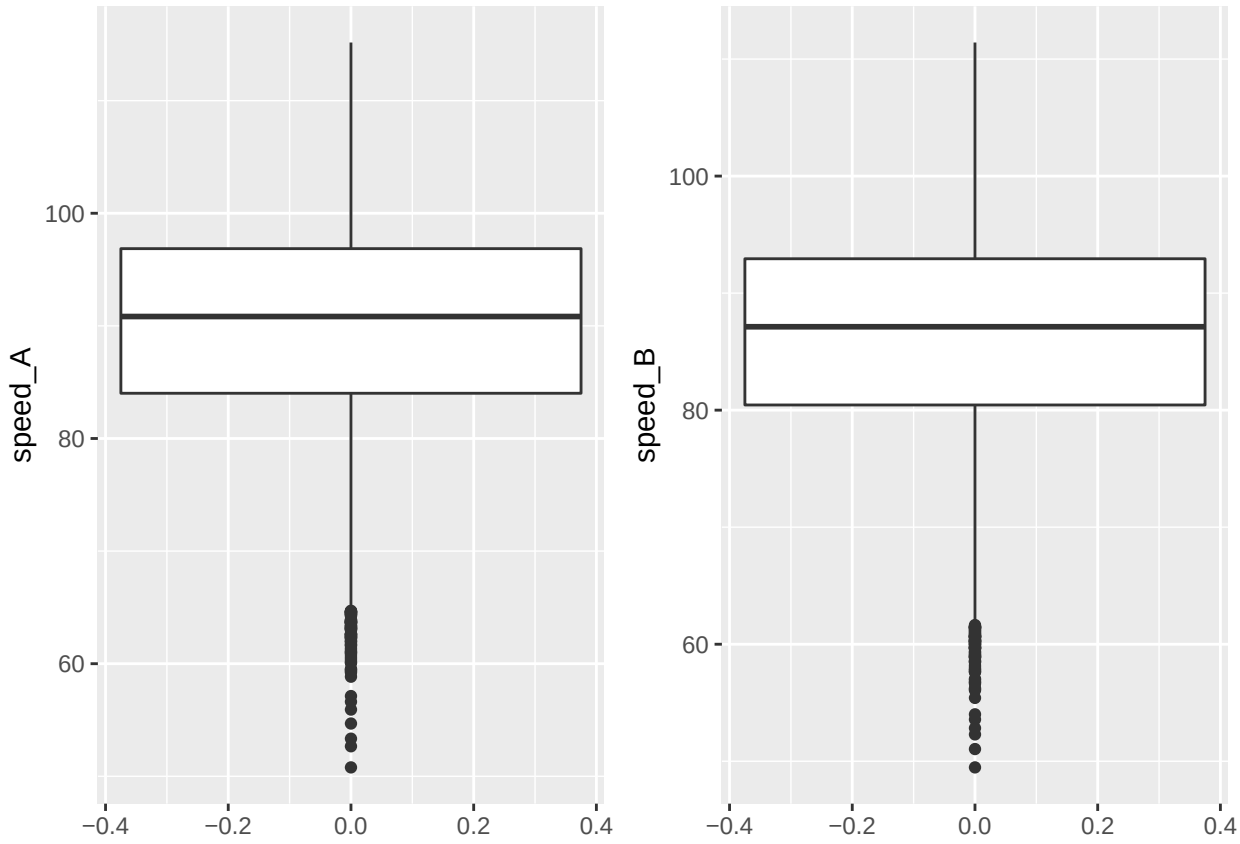
The distributions for speed and angle for line drives appear to be more or less the same for system A and system B. However, the negative angles for line drives are very unusual and unintuitive, and they should perhaps be cleaned from the data set. Alternatively, if an angle measurement for A is negative but is positive for B, then B should be used (and vice versa).

Fly balls

```
plot_grid(speedAfb, speedBfb)
```

```
## Warning: Removed 276 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 236 rows containing non-finite values (stat_boxplot).
```




```
dataset %>%
  filter(hittype == "fly_ball") %>%
  summarise(minA = min(speed_A, na.rm = TRUE), q1A = quantile(speed_A, 0.25, na.rm = TRUE), meanA
            = mean(speed_A, na.rm = TRUE), q3A = quantile(speed_A, 0.75, na.rm = TRUE), maxA =
              max(speed_A, na.rm = TRUE))
```

```
##       minA      q1A    meanA      q3A     maxA
## 1 50.79051 84.01561 90.05837 96.85953 115.1647
```

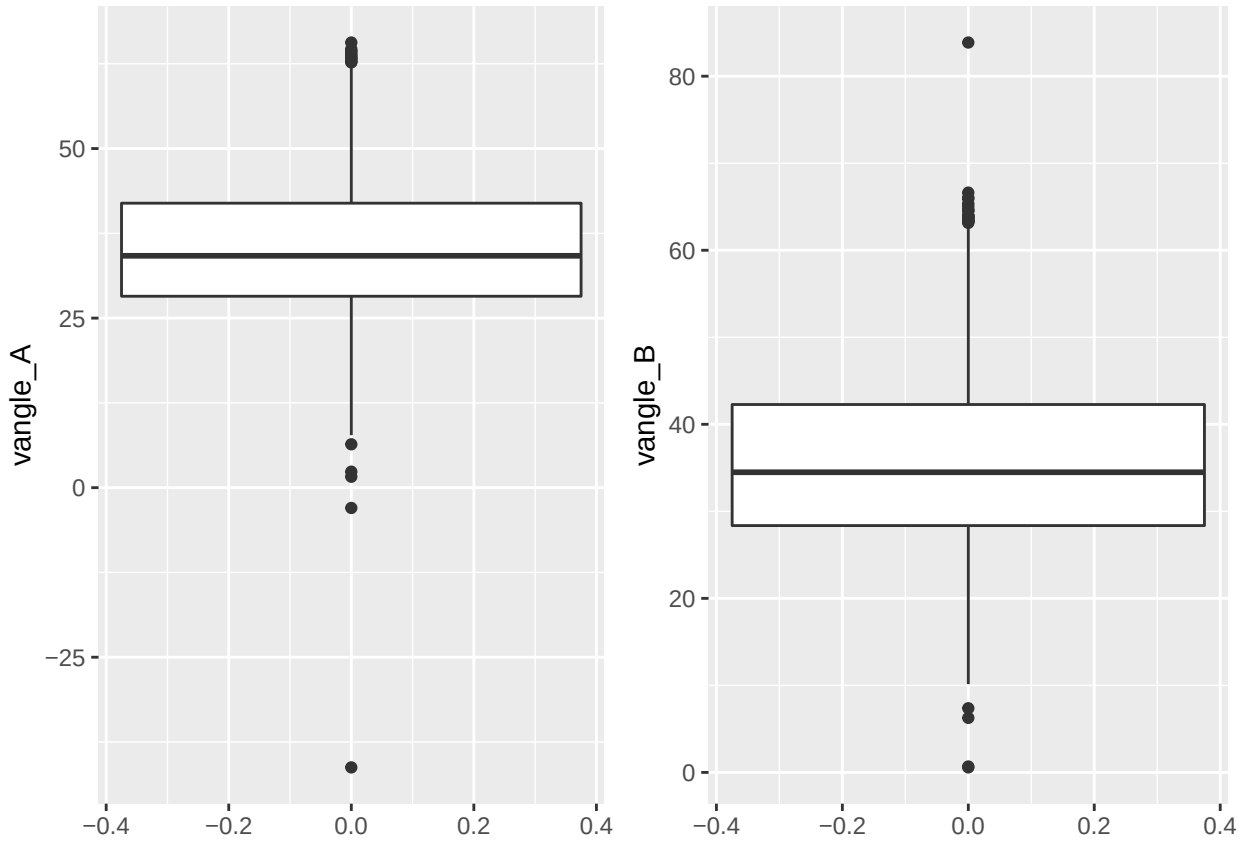
```
dataset %>%
  filter(hittype == "fly_ball") %>%
  summarise(minB = min(speed_B, na.rm = TRUE), q1B = quantile(speed_B, 0.25, na.rm = TRUE), meanB
            = mean(speed_B, na.rm = TRUE), q3B = quantile(speed_B, 0.75, na.rm = TRUE), maxB =
              max(speed_B, na.rm = TRUE))
```

```
##       minB      q1B    meanB      q3B     maxB
## 1 49.47551 80.44244 86.31298 92.93764 111.4138
```

```
plot_grid(vangleAfb, vangleBfb)
```

```
## Warning: Removed 276 rows containing non-finite values (stat_boxplot).

## Warning: Removed 236 rows containing non-finite values (stat_boxplot).
```




```
dataset %>%
  filter(hittype == "fly_ball") %>%
  summarise(minA = min(vangle_A, na.rm = TRUE), q1A = quantile(vangle_A, 0.25, na.rm = TRUE),
            meanA = mean(vangle_A, na.rm = TRUE), q3A = quantile(vangle_A, 0.75, na.rm = TRUE),
            maxA = max(vangle_A, na.rm = TRUE))
```

```
##        minA      q1A   meanA      q3A     maxA
## 1 -41.24443 28.22384 35.4001 41.94293 65.63399
```

```
dataset %>%
  filter(hittype == "fly_ball") %>%
  summarise(minB = min(vangle_B, na.rm = TRUE), q1B = quantile(vangle_B, 0.25, na.rm = TRUE),
            meanB = mean(vangle_B, na.rm = TRUE), q3B = quantile(vangle_B, 0.75, na.rm = TRUE),
            maxB = max(vangle_B, na.rm = TRUE))
```

```
##        minB      q1B    meanB      q3B     maxB
## 1 0.5774136 28.36421 35.70389 42.27492 83.87139
```

```
ttestspeedAfb <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(speed_A))
ttestspeedBfb <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(speed_B))
t.test(ttestspeedAfb, ttestspeedBfb, alternative = "two.sided", mu = 0,
       paired = FALSE, var.equal = TRUE, conf.level = 0.95)
```

```
##
##  Two Sample t-test
```

```
##
## data:  ttestspeedAfb and ttestspeedBfb
## t = 34.141, df = 35942, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  3.859532 4.329672
## sample estimates:
## mean of x mean of y
##  92.83912  88.74452
```

```
ttestvangleAfb <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(vangle_A))
ttestvangleBfb <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(vangle_B))
t.test(ttestvangleAfb, ttestvangleBfb, alternative = "two.sided", mu = 0,
       paired = FALSE, var.equal = TRUE, conf.level = 0.95)
```

```
##
##  Two Sample t-test
##
## data:  ttestvangleAfb and ttestvangleBfb
## t = 6.6143, df = 35942, p-value = 3.786e-11
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.3290589 0.6062099
## sample estimates:
## mean of x mean of y
##  15.82258  15.35495
```

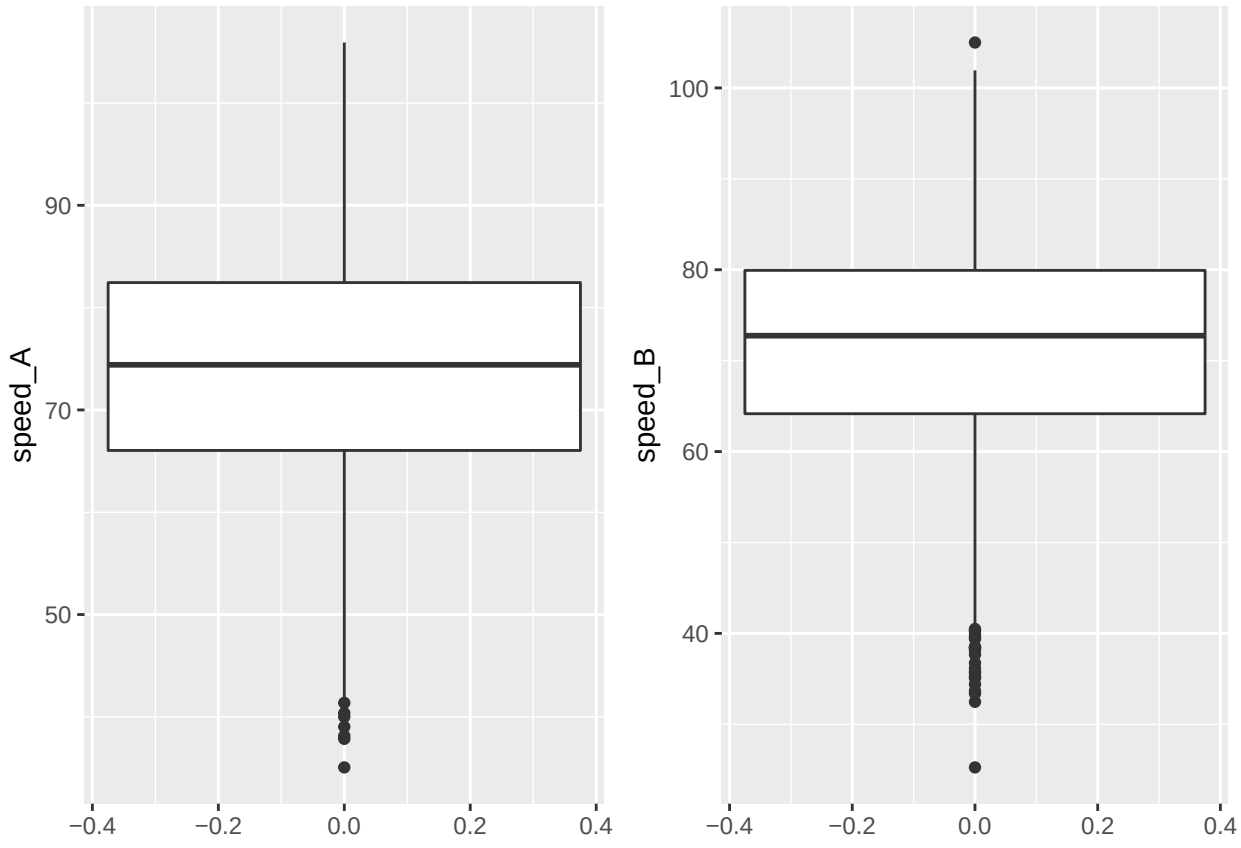
Again, the distributions appear to be similar for fly balls; however, the `vangle` outliers for system A are much more significant and should be cleaned from the data set.

Pop ups

```
plot_grid(speedApu, speedBpu)
```

```
## Warning: Removed 2446 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 355 rows containing non-finite values (stat_boxplot).
```




```
dataset %>%
  filter(hittype == "popup") %>%
  summarise(minA = min(speed_A, na.rm = TRUE), q1A = quantile(speed_A, 0.25, na.rm = TRUE), meanA
            = mean(speed_A, na.rm = TRUE), q3A = quantile(speed_A, 0.75, na.rm = TRUE), maxA =
              max(speed_A, na.rm = TRUE))
```

```
##       minA      q1A    meanA      q3A     maxA
## 1 35.06217 66.04185 73.92777 82.44511 105.9107
```

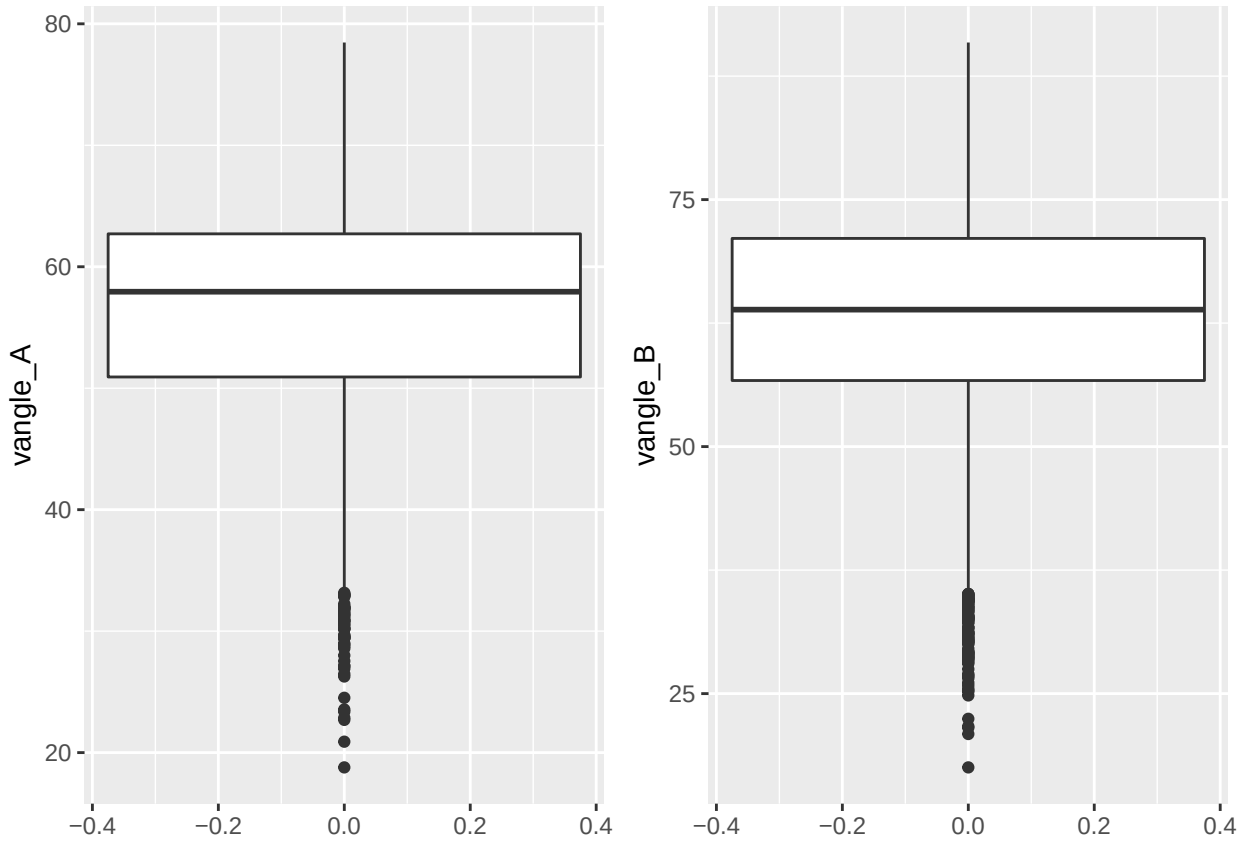
```
dataset %>%
  filter(hittype == "popup") %>%
  summarise(minB = min(speed_B, na.rm = TRUE), q1B = quantile(speed_B, 0.25, na.rm = TRUE), meanB
            = mean(speed_B, na.rm = TRUE), q3B = quantile(speed_B, 0.75, na.rm = TRUE), maxB =
              max(speed_B, na.rm = TRUE))
```

```
##      minB      q1B    meanB      q3B     maxB
## 1 25.2595 64.16481 71.73106 79.93452 104.9961
```

```
plot_grid(vangleApu, vangleBpu)
```

```
## Warning: Removed 2446 rows containing non-finite values (stat_boxplot).

## Warning: Removed 355 rows containing non-finite values (stat_boxplot).
```




```
dataset %>%
  filter(hittype == "popup") %>%
  summarise(minA = min(vangle_A, na.rm = TRUE), q1A = quantile(vangle_A, 0.25, na.rm = TRUE),
            meanA = mean(vangle_A, na.rm = TRUE), q3A = quantile(vangle_A, 0.75, na.rm = TRUE),
            maxA = max(vangle_A, na.rm = TRUE))
```

```
##       minA      q1A    meanA      q3A     maxA
## 1 18.78228 50.92506 56.11939 62.70811 78.46098
```

```
dataset %>%
  filter(hittype == "popup") %>%
  summarise(minB = min(vangle_B, na.rm = TRUE), q1B = quantile(vangle_B, 0.25, na.rm = TRUE),
            meanB = mean(vangle_B, na.rm = TRUE), q3B = quantile(vangle_B, 0.75, na.rm = TRUE),
            maxB = max(vangle_B, na.rm = TRUE))
```

```
##       minB      q1B   meanB      q3B     maxB
## 1 17.53133 56.68934 63.1098 71.07286 90.90082
```

```
ttestspeedApu <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(speed_A))
ttestspeedBpu <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(speed_B))
t.test(ttestspeedApu, ttestspeedBpu, alternative = "two.sided", mu = 0,
       paired = FALSE, var.equal = TRUE, conf.level = 0.95)
```

```
##
##  Two Sample t-test
```

```
##
## data:  ttestspeedApu and ttestspeedBpu
## t = 34.141, df = 35942, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  3.859532 4.329672
## sample estimates:
## mean of x mean of y
##  92.83912  88.74452
```

```
ttestvangleApu <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(vangle_A))
ttestvangleBpu <- as.data.frame(dataset %>%
  filter(hittype == "line_drive") %>%
  select(vangle_B))
t.test(ttestvangleApu, ttestvangleBpu, alternative = "two.sided", mu = 0,
       paired = FALSE, var.equal = TRUE, conf.level = 0.95)
```

```
##
##  Two Sample t-test
##
## data:  ttestvangleApu and ttestvangleBpu
## t = 6.6143, df = 35942, p-value = 3.786e-11
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.3290589 0.6062099
## sample estimates:
## mean of x mean of y
##  15.82258  15.35495
```

The speeds from systems A and B appear to be more or less the same; however, the angles are distinctly different. Again, the heuristic in this case would be to trust A over B, though it is possible that the launch angle simply should not be used to predict speed for popups.

Data Cleaning/Analysis

Since the t-tests for all of the above speeds/angles for each hit type provided little information, I will base the following on the results of the boxplots.

Ground balls

For ground balls, we found that the speeds calculated by B were drastically different than those calculated by A. Since we are given that system A is assumed to be more accurate than system B, when system A is known, it should be used to predict the true speed for ground balls.

```
dataset <- dataset %>%
  mutate(true_speed = case_when(
    is.na(speed_A) == TRUE ~ NA_real_,
    hittype == "ground_ball" ~ speed_A
  ))
```

When system A is NA for ground balls, our only option is to predict A using B. We will fit a linear regression using the speed and angle for B. Since the distribution of the angles for A and B appeared similar, it is likely

that vangle_B is a good predictor of speed_A, but we will initially begin with speed_B and vangle_B as predictors.

```
full_model_gb <- lm(data = subset(dataset, hittype == "ground_ball"),
                    speed_A ~ speed_B + vangle_B + speed_B*vangle_B)
partial_model_gb <- lm(data = subset(dataset, hittype == "ground_ball"),
                       speed_A ~ speed_B + vangle_B)
angle_model_gb <- lm(data = subset(dataset, hittype == "ground_ball"), speed_A ~ vangle_B)

kable(anova(partial_model_gb,full_model_gb,test="Chisq"),format="html")
```

Res.Df

RSS

Df

Sum of Sq

Pr(>Chi)

28259

2063509

NA

NA

NA

28258

1941836

1

121672.8

0

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

```
## [1] 0.6729119
```

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

```
## [1] 0.652417
```

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

```
## [1] 0.07570127
```

The nested F test shows that that the full model is the best linear regression to predict speed_A from system B. The r-squared values confirm this, and it shows that simply using vangle_B is not an accurate way to predict speed_A. Finally, we will predict the true_speed of all ground balls by fitting the full model to ground ball cases where system A is NA.

```
dataset <- dataset %>%
  mutate(speed_A = ifelse(is.na(speed_A) & hittype == "ground_ball", predict(full_model_gb),
                          speed_A))
dataset <- dataset %>%
  mutate(true_speed = ifelse(is.na(true_speed) & hittype == "ground_ball", speed_A, true_speed))
```

Line drives

For line drives, the distributions of speeds and angles for A and B appeared to be more or less similar. However, there were a number of angles that were calculated as negative, which does not make sense in the case of line drives. We will first look at the negative angle values for line drives and see if it is consistent between A and B to perform some data cleaning.

We removed all line drives that had negative angles recorded from both systems A and B because they, intuitively, do not qualify as line drives and should not be used to calculate true speed. Now, we will predict A from B and vice versa for the NAs.

```
full_model_ld <- lm(data = subset(dataset, hittype == "line_drive"),
                    speed_A ~ speed_B + vangle_B + speed_B*vangle_B)
partial_model_ld <- lm(data = subset(dataset, hittype == "line_drive"),
                       speed_A ~ speed_B + vangle_B)
angle_model_ld <- lm(data = subset(dataset, hittype == "line_drive"), speed_A ~ vangle_B)

kable(anova(partial_model_ld,full_model_ld,test="Chisq"),format="html")
```

Res.Df

RSS

Df

Sum of Sq

Pr(>Chi)

17751

22110.82

NA

NA

NA

17750

21879.02

1

231.8034

0

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

```
## [1] 0.9909117
```

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

```
## [1] 0.9908154
```

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

```
## [1] 0.08741479
```

```
full_model_ldB <- lm(data = subset(dataset, hittype == "line_drive"),
                    speed_B ~ speed_A + vangle_A + speed_A*vangle_A)
partial_model_ldB <- lm(data = subset(dataset, hittype == "line_drive"),
                       speed_B ~ speed_A + vangle_A)
```

```
angle_model_ldB <- lm(data = subset(dataset, hittype == "line_drive"), speed_B ~ vangle_A)

kable(anova(partial_model_ldB,full_model_ld,test="Chisq"),format="html")
```

```
## Warning in anova.lmlist(object, ...): models with response '"speed_A"'
## removed because response differs from model 1
```

Df

Sum Sq

Mean Sq

F value

Pr(>F)

speed_A

1

2.126568e+06

2.126568e+06

1.907103e+06

0.0000000

vangle_A

1

2.011838e+00

2.011838e+00

1.804214e+00

0.1792211

Residuals

17751

1.979374e+04

1.115077e+00

NA

NA

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

```
## [1] 0.9913765
```

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

```
## [1] 0.990778
```

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

```
## [1] 0.1144445
```

Given the nested F test and the r-squared values, the full models including the interaction prove to be the best predictors of A and B. We will now fill the NAs for line drives. When the angle is less than zero for A or

B, we will use the speed measurements from the non-zero system. When both angles are greater than zero, we will take the average of the speed measurements since they are more or less similar.

```
dataset <- dataset %>%
  mutate(speed_A = ifelse(is.na(speed_A) & hittype == "line_drive", predict(full_model_ld),
                          speed_A))
dataset <- dataset %>%
  mutate(speed_B = ifelse(is.na(speed_B) & hittype == "line_drive", predict(full_model_ldB),
                          speed_B))
dataset <- dataset %>%
  mutate(true_speed = ifelse((vangle_B < 0 | vangle_A < 0) & hittype == "line_drive", NA_real_,
                             true_speed)) %>%
  mutate(true_speed = case_when(
    hittype != "line_drive" ~ true_speed,
    hittype == "line_drive" & vangle_A < 0 ~ speed_B,
    hittype == "line_drive" & vangle_B < 0 ~ speed_A,
    hittype == "line_drive" & vangle_A >= 0 & vangle_B >= 0 ~ (speed_A + speed_B)/2,
    hittype == "line_drive" & (is.na(vangle_A) | is.na(vangle_B)) ~ (speed_A + speed_B)/2
    ))
```

Fly balls

For fly balls, the distribution of speed and angles for A and B appeared similar. However, there were two significant fly ball `vangle_A` outliers in which the angle was negative which should not be the case for fly balls. In those two instances, we will simply use system B as the predictor of `true_speed`. Otherwise, we will average `speed_A` and `speed_B` to predict `true_speed`. We we again use linear regression to fill in the NAs for `speed_A` and `speed_B`.

```
full_model_fb <- lm(data = subset(dataset, hittype == "fly_ball"),
                    speed_A ~ speed_B + vangle_B + speed_B*vangle_B)
partial_model_fb <- lm(data = subset(dataset, hittype == "fly_ball"),
                       speed_A ~ speed_B + vangle_B)
angle_model_fb <- lm(data = subset(dataset, hittype == "fly_ball"), speed_A ~ vangle_B)

kable(anova(partial_model_fb,full_model_fb,test="Chisq"),format="html")
```

Res.Df

RSS

Df

Sum of Sq

Pr(>Chi)

16216

18704.05

NA

NA

NA

16215

17945.09

1

758.9638

0

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

```
## [1] 0.987269
```

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

```
## [1] 0.9867305
```

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

```
## [1] 0.06055608
```

```
full_model_fbB <- lm(data = subset(dataset, hittype == "fly_ball"),
                     speed_B ~ speed_A + vangle_A + speed_A*vangle_A)
partial_model_fbB <- lm(data = subset(dataset, hittype == "fly_ball"),
                        speed_B ~ speed_A + vangle_A)
angle_model_fbB <- lm(data = subset(dataset, hittype == "fly_ball"), speed_B ~ vangle_A)

kable(anova(partial_model_fbB,full_model_fb,test="Chisq"),format="html")
```

```
## Warning in anova.lmlist(object, ...): models with response '"speed_A"'
## removed because response differs from model 1
```

Df

Sum Sq

Mean Sq

F value

Pr(>F)

speed_A

1

1320856.587

1.320857e+06

1185677.995

0

vangle_A

1

1250.354

1.250354e+03

1122.391

0

Residuals

16216

18064.779

1.114010e+00

NA

NA

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

```
## [1] 0.9868165
```

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

```
## [1] 0.9865205
```

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

```
## [1] 0.05688059
```

```
dataset <- dataset %>%
  mutate(speed_A = ifelse(is.na(speed_A) & hittype == "fly_ball", predict(full_model_fb),
                          speed_A))
dataset <- dataset %>%
  mutate(speed_B = ifelse(is.na(speed_B) & hittype == "fly_ball", predict(full_model_fbB),
                          speed_B))

dataset %>%
  filter(hittype == "fly_ball" & vangle_A < 0)
```

```
##   batter pitcher  hittype  speed_A   vangle_A  speed_B vangle_B true_speed
## 1     86     280 fly_ball 80.78150 -41.244432 90.93112 37.04955         NA
## 2    536     515 fly_ball 78.41118  -2.990652 78.24196 42.05891         NA
```

```
dataset <- dataset %>%
  mutate(true_speed = case_when(
    hittype == "fly_ball" & vangle_A < 0 ~ speed_B,
    hittype == "fly_ball" & (vangle_A > 0 | is.na(vangle_A)) ~ (speed_A + speed_B)/2,
    hittype != "fly_ball" ~ true_speed
  ))
```

Pop Ups

For popups, we found that `speed_A` and `speed_B` had similar distributions but `vangle_A` and `vangle_B` had very different means and distributions. Again, we should trust system A over B, so we will change `vangle_B` to `vangle_A` when possible. Though, we can still take the average of `speed_A` and `speed_B` to calculate `true_speed`. We will also use linear regression to populate the NAs for `speed_A` and `speed_B`.

```
dataset <- dataset %>%
  mutate(vangle_B = case_when(
    hittype == "popup" & is.na(vangle_A) ~ vangle_B,
    hittype == "popup" & (is.na(vangle_A) == FALSE) ~ vangle_A,
    hittype != "popup" ~ vangle_B
  ))

full_model_pu <- lm(data = subset(dataset, hittype == "popup"),
                    speed_A ~ speed_B + vangle_B + speed_B*vangle_B)
partial_model_pu <- lm(data = subset(dataset, hittype == "popup"),
                       speed_A ~ speed_B + vangle_B)
angle_model_pu <- lm(data = subset(dataset, hittype == "popup"), speed_A ~ vangle_B)
```

```
kable(anova(partial_model_pu,full_model_pu,test="Chisq"),format="html")
```

Res.Df

RSS

Df

Sum of Sq

Pr(>Chi)

2678

4912.717

NA

NA

NA

2677

4841.885

1

70.83146

0

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

```
## [1] 0.9858423
```

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

```
## [1] 0.9856352
```

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

```
## [1] 0.3021802
```

```
full_model_puB <- lm(data = subset(dataset, hittype == "popup"),
                     speed_B ~ speed_A + vangle_A + speed_A*vangle_A)
partial_model_puB <- lm(data = subset(dataset, hittype == "popup"),
                        speed_B ~ speed_A + vangle_A)
angle_model_puB <- lm(data = subset(dataset, hittype == "popup"), speed_B ~ vangle_A)

kable(anova(partial_model_fbB,full_model_fb,test="Chisq"),format="html")
```

```
## Warning in anova.lmlist(object, ...): models with response '"speed_A"'
## removed because response differs from model 1
```

Df

Sum Sq

Mean Sq

F value

Pr(>F)

speed_A

1

1320856.587

1.320857e+06

1185677.995

0

vangle_A

1

1250.354

1.250354e+03

1122.391

0

Residuals

16216

18064.779

1.114010e+00

NA

NA

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

```
## [1] 0.9860425
```

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

```
## [1] 0.9857393
```

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

```
## [1] 0.3060962
```

```
dataset <- dataset %>%
  mutate(speed_A = ifelse(is.na(speed_A) & hittype == "popup", predict(full_model_pu),
                          speed_A))
dataset <- dataset %>%
  mutate(speed_B = ifelse(is.na(speed_B) & hittype == "popup", predict(full_model_puB),
                          speed_B))

dataset <- dataset %>%
  mutate(true_speed = case_when(
    hittype == "popup" ~ (speed_A + speed_B)/2,
    hittype != "popup" ~ true_speed
  ))
```

Conclusion

First, let's look at the 5-number summary of the calculated `true_speed`:

```
dataset %>%
  filter(hittype == "ground_ball") %>%
  summarise(mingb = min(true_speed, na.rm = TRUE), q1gb = quantile(true_speed, 0.25, na.rm =
                                                                   TRUE),
            meangb = mean(true_speed, na.rm = TRUE), q3gb = quantile(true_speed, 0.75, na.rm =
                                                                    TRUE),
            maxgb = max(true_speed, na.rm = TRUE))
```

```
##      mingb   q1gb   meangb     q3gb    maxgb
## 1 26.46182 77.043 86.18843 97.20741 121.8475
```

```
dataset %>%
  filter(hittype == "line_drive") %>%
  summarise(minld = min(true_speed, na.rm = TRUE), q1ld = quantile(true_speed, 0.25, na.rm =
                                                                   TRUE),
            meanld = mean(true_speed, na.rm = TRUE), q3ld = quantile(true_speed, 0.75, na.rm =
                                                                    TRUE),
            maxld = max(true_speed, na.rm = TRUE))
```

```
##      minld     q1ld   meanld     q3ld    maxld
## 1 34.70057 83.99692 90.79122 99.17201 116.6054
```

```
dataset %>%
  filter(hittype == "fly_ball") %>%
  summarise(minfb = min(true_speed, na.rm = TRUE), q1fb = quantile(true_speed, 0.25, na.rm =
                                                                   TRUE),
            meanfb = mean(true_speed, na.rm = TRUE), q3fb = quantile(true_speed, 0.75, na.rm =
                                                                    TRUE),
            maxfb = max(true_speed, na.rm = TRUE))
```

```
##      minfb     q1fb   meanfb     q3fb    maxfb
## 1 52.19242 82.32038 88.18887 94.83531 113.2559
```

```
dataset %>%
  filter(hittype == "popup") %>%
  summarise(minpu = min(true_speed, na.rm = TRUE), q1pu = quantile(true_speed, 0.25, na.rm =
                                                                   TRUE),
            meanpu = mean(true_speed, na.rm = TRUE), q3pu = quantile(true_speed, 0.75, na.rm =
                                                                    TRUE),
            maxpu = max(true_speed, na.rm = TRUE))
```

```
##      minpu     q1pu   meanpu    q3pu    maxpu
## 1 33.76575 66.03642 72.65276 79.8856 103.9216
```

For each hit type, the distributions seem to be distinctly different, thus validating the idea that each hit type should be treated independently. It is intuitive that popups would have the lowest mean `true_speed`, as they do not leave the infield. Moreover, it makes sense that line drives would have the highest mean `true_speed`, since they are hit squarely.

In all, to predict batter performance next year, we recommend the following:

For ground balls: Use system A to measure the speed, and if A is unavailable, then use a linear regression model to predict A from B. This is because there is a large discrepancy between A and B, and we are told that system A is generally more reliable.

For line drives: Disregard any observations in which `vangle_A` and `vangle_B` are both negative, as that does not align with the characteristics of a line drive. Then, average `speed_A` and `speed_B` to calculate

`true_speed` since the measurements appear similar. If A or B is NA, then use linear regression with the known system to predict the other and average them.

For fly balls: Remove significant `vangle_A` outliers, as system A seems to occasionally malfunction in regards to angles of fly balls. In those cases, resort to using system B. Otherwise, average the speeds of systems A and B due to the fact that they are similar.

For popups: Replace `vangle_B` with `vangle_A` because they are drastically different, and we must assume A is more reliable. For speed, A and B seem to be consistent so take the average speed.

This `true_speed` system should prove to be a relatively accurate way of measuring hit speed.