

Catch probability

Tim White

2025-01-18

```
library(tidyverse)
library(rjson)
library(baseballr)
library(rvest)
```

Scrape data

We scrape individual play data for every fly ball hit to an outfielder during the 2024 season. This chunk only needs to be run once.

```
scrape_of_catch_prob_data <- function(year) {
  # Load in player IDs
  playerIDs <- read_csv(paste0("../data/of_playerIDs_", year, ".csv")) %>%
    select(player_id)

  # Scrape play-by-play data
  data <- lapply(1:nrow(playerIDs),
    function(j) {
      # Scrape data
      rawdata <- fromJSON(
        file = paste0("https://baseballsavant.mlb.com/player-services/range?playerId=",
                      playerIDs[j,], "&season=", year, "&playerType=fielder"), simplify = TRUE
      )

      # If the URL exists:
      if (length(rawdata) > 0) {

        # Change any null columns (e.g., sprint_speed) to NA
        for (k in 1:length(rawdata)) {
          rawdata[[k]][sapply(rawdata[[k]], is.null)] <- NA
        }

        # Convert raw data to tibble
        tibble(data.frame(matrix(unlist(rawdata),
          nrow = length(rawdata),
          byrow = TRUE,
          dimnames = list(1:length(rawdata),
                        names(rawdata[[1]])))), %>%
          mutate(across(c(game_pk:name_display_first_last, pos),
                       as.factor),
```

```

        across(c(stars:distance, hang_time, out:sprint_speed),
               as.numeric))
    }
}

return(tibble(do.call(rbind.data.frame, data)))
}

clean_of_catch_prob_data <- function(year) {
  of_catch_prob <- read_csv(paste0("../data/of_catch_prob_", year, "_orig.csv")) %>%
    filter(!is.na(sprint_speed)) %>%
    mutate(game_pk = as.factor(game_pk),
           play_id = as.factor(play_id),
           player = as.factor(name_display_first_last),
           pos = as.factor(pos))

  teams <- fg_batter_leaders(startseason = year, endseason = year) %>%
    mutate(team_name = as.factor(ifelse(team_name == " - - - ", "MUL", team_name))) %>%
    select(player = PlayerName, team = team_name)
  of_catch_prob_teams <- teams %>% right_join(of_catch_prob, by = "player")

  stadiums <- mlb_schedule(season = year) %>%
    mutate(game_pk = as.factor(game_pk),
           stadium = as.factor(venue_name)) %>%
    select(game_pk, stadium)
  of_catch_prob_teams_stadiums <- of_catch_prob_teams %>%
    left_join(stadiums, by = "game_pk")

  dimensions_table <- read_html("http://www.andrewclem.com/Baseball/Stadium_statistics.html") %>%
    html_elements("table") %>%
    html_table()
  dimensions_colnames <- dimensions_table[[1]] %>% janitor::row_to_names(row_number=1) %>%
    select(stadium = "Stadium (see notes)",
           wall_height_lf = "LF", wall_height_cf = "CF", wall_height_rf = "RF",
           dist_lf = "Left field", dist_lc = "Left-center", dist_cf = "Center field",
           dist_rc = "Right-center", dist_rf = "Right field") %>%
    colnames()
  dimensions_data <- dimensions_table[[2]] %>%
    select(X1, X12, X13, X14, X17, X18, X19, X20, X21) %>%
    rename_with(~dimensions_colnames) %>%
    mutate(stadium = as.factor(stadium)) %>%
    mutate(stadium = str_remove_all(stadium, "\\\\"*)) %>%
    mutate(across(contains("wall"),
                 function(col) {str_remove_all(col, "[()\\\\[\\\\]]")})) %>%
    mutate(across(contains("wall"), as.numeric)) %>%
    mutate(across(contains("dist"),
                 function(col) {str_remove_all(col, "[()\\\\[\\\\]]")})) %>%
    mutate(across(contains("dist"), as.numeric)) %>%
    mutate(stadium = fct_recode(stadium,
                               "American Family Field" = "Miller Park",
                               "Busch Stadium" = "Busch Stadium III",
                               "Great American Ball Park" = "Great American Ballpa"))
}

```

```

    "loanDepot park" = "Marlins Park",
    "Petco Park" = "PETCO Park",
    "T-Mobile Park" = "Safeco Field",
    "Truist Park" = "Truist (ex-SunTrust) Park"))
of_catch_prob_teams_stadiums_dims <- of_catch_prob_teams_stadiums %>%
  left_join(dimensions_data, by = "stadium")

return(of_catch_prob_teams_stadiums_dims)
}

of_catch_prob_2024_orig <- scrape_of_catch_prob_data("2024")
of_catch_prob_2024_orig %>% write_csv("../data/of_catch_prob_2024_orig.csv")
of_catch_prob_2024 <- clean_of_catch_prob_data("2024")
of_catch_prob_2024 %>% write_csv("../data/of_catch_prob_2024.csv")

of_catch_prob_2023_orig <- scrape_of_catch_prob_data("2023")
of_catch_prob_2023_orig %>% write_csv("../data/of_catch_prob_2023_orig.csv")
of_catch_prob_2023 <- clean_of_catch_prob_data("2023")
of_catch_prob_2023 %>% write_csv("../data/of_catch_prob_2023.csv")

```

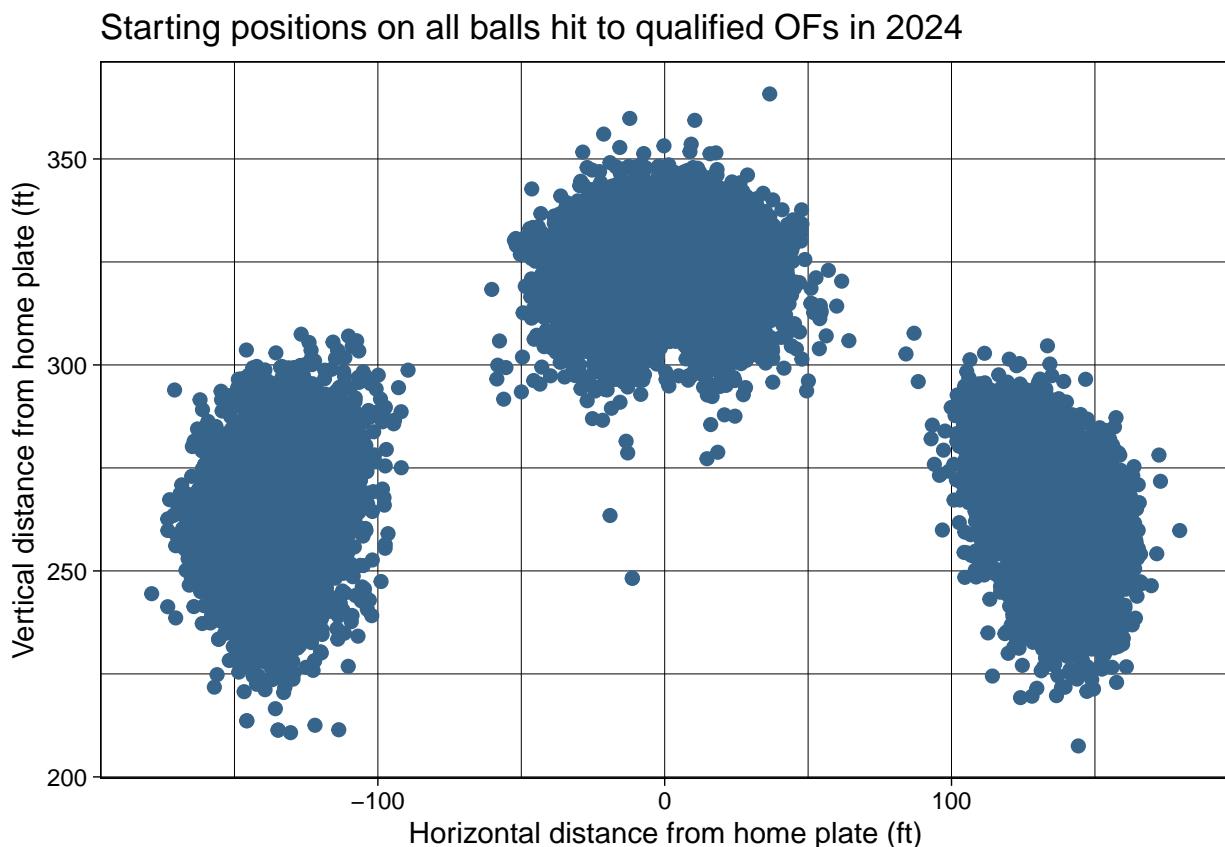
Load in data

```
of_catch_prob_2024 <- read_csv("../data/of_catch_prob_2024.csv")
of_catch_prob_2023 <- read_csv("../data/of_catch_prob_2023.csv")
```

January 2nd

```
jan2_1 <- of_catch_prob_2024 %>%
  ggplot(aes(x = start_pos_x, y = start_pos_y)) +
  geom_point(col = "steelblue4", size = 2, shape = 19) +
  labs(title = "Starting positions on all balls hit to qualified OFs in 2024",
       x = "Horizontal distance from home plate (ft)",
       y = "Vertical distance from home plate (ft)") +
  theme_linedraw()
```

```
jan2_1
```

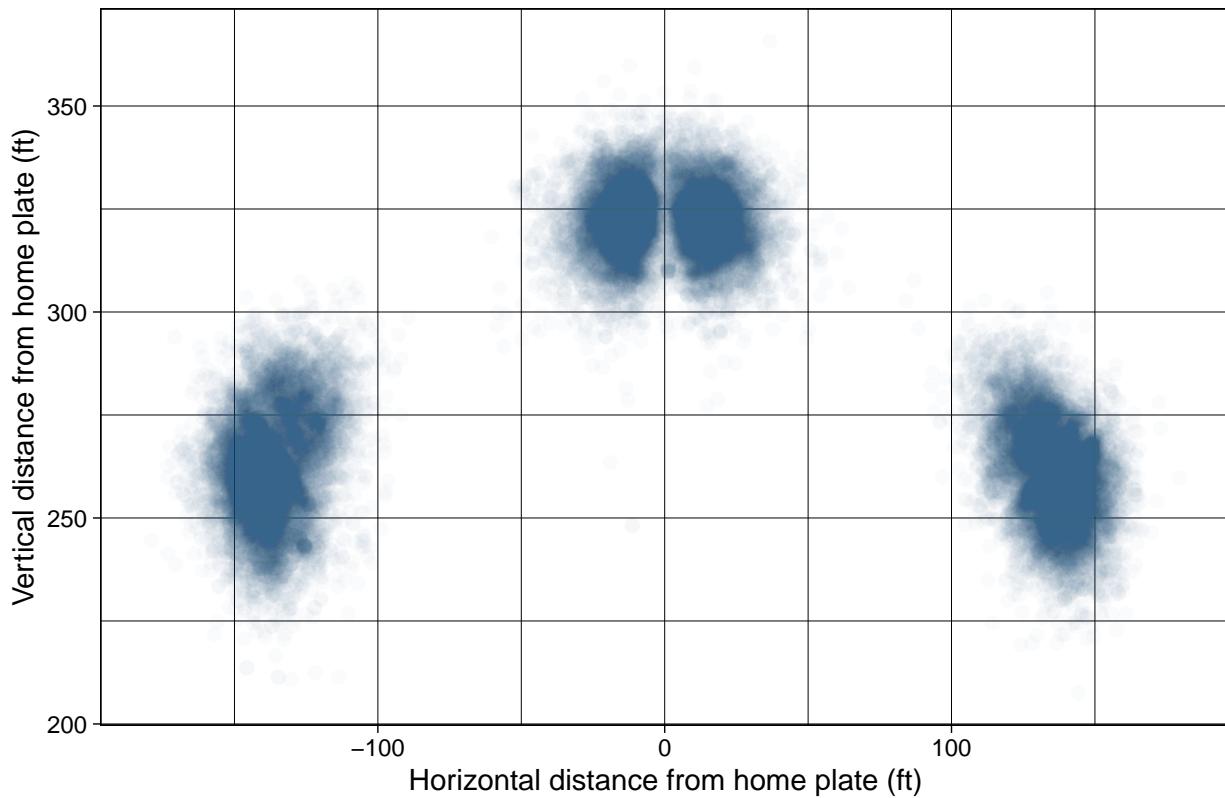


```
ggsave("../figures/jan2_1.png", plot = jan2_1)
```

```
jan2_2 <- of_catch_prob_2024 %>%
  ggplot(aes(x = start_pos_x, y = start_pos_y)) +
  geom_point(col = "steelblue4", size = 2, shape = 19, alpha = 0.025) +
  labs(title = "Starting positions on all balls hit to qualified OFs in 2024",
       x = "Horizontal distance from home plate (ft)",
       y = "Vertical distance from home plate (ft)") +
  theme_linedraw()
```

```
jan2_2
```

Starting positions on all balls hit to qualified OFs in 2024

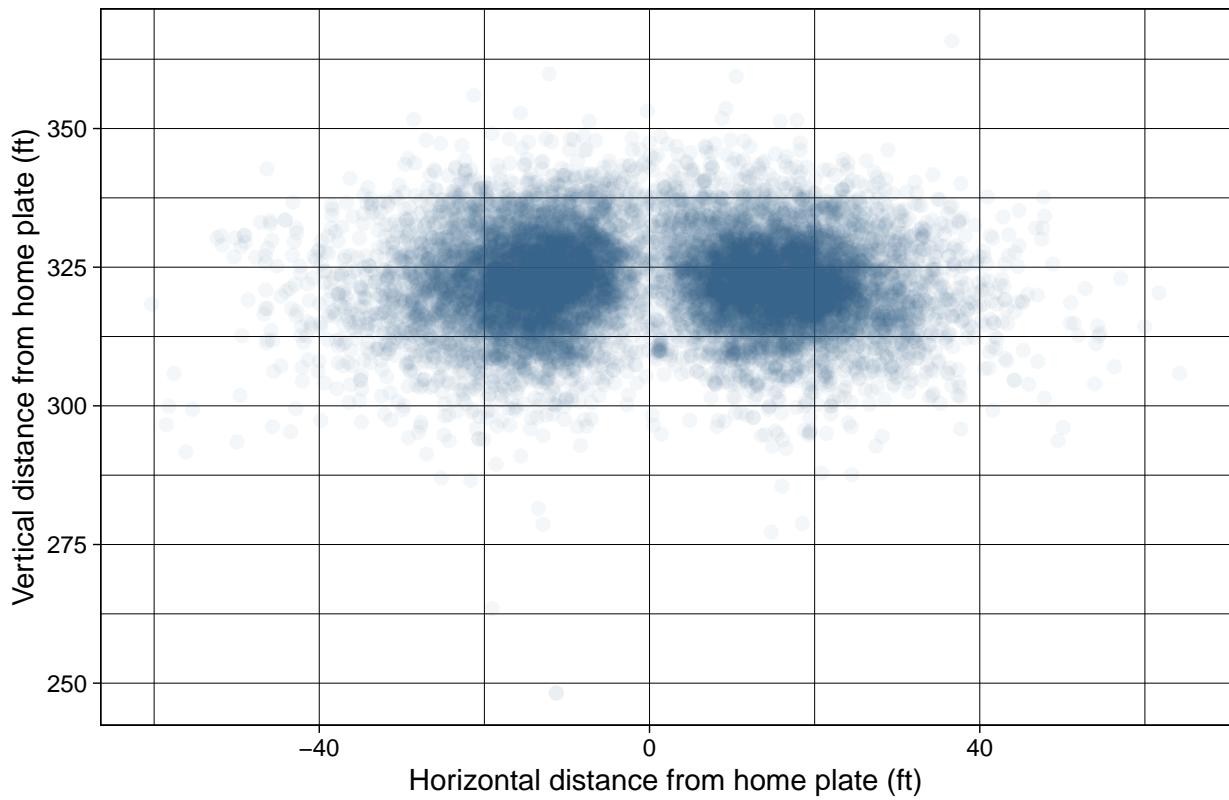


```
ggsave("../figures/jan2_2.png", plot = jan2_2)
```

```
jan2_3 <- of_catch_prob_2024 %>%
  filter(pos == 8) %>%
  ggplot(aes(x = start_pos_x, y = start_pos_y)) +
  geom_point(col = "steelblue4", size = 2, shape = 19, alpha = 0.05) +
  labs(title = "Starting positions on all balls hit to qualified CFs in 2024",
       x = "Horizontal distance from home plate (ft)",
       y = "Vertical distance from home plate (ft)") +
  theme_linedraw()
```

```
jan2_3
```

Starting positions on all balls hit to qualified CFs in 2024



```
ggsave("../figures/jan2_3.png", plot = jan2_3)
```

January 4th

```
single_game_oaa <- of_catch_prob_2024 %>%
  group_by(game_pk, player) %>%
  summarize(oaa = sum(out * (1 - catch_rate) - (1 - out) * catch_rate),
            opportunities = n(),
            catches = sum(out),
            stars5_opps = sum(stars == 5),
            stars5_catches = sum(stars == 5 & out == 1),
            stars4_opps = sum(stars == 4),
            stars4_catches = sum(stars == 4 & out == 1),
            stars3_opps = sum(stars == 3),
            stars3_catches = sum(stars == 3 & out == 1),
            stars2_opps = sum(stars == 2),
            stars2_catches = sum(stars == 2 & out == 1),
            stars1_opps = sum(stars == 1),
            stars1_catches = sum(stars == 1 & out == 1),
            stars0_opps = sum(stars == 0),
            stars0_catches = sum(stars == 0 & out == 1),
            .groups = "drop")
```

```
single_game_oaa %>%
  arrange(desc(oaa)) %>%
  head(10)
```

```
## # A tibble: 10 x 17
##   game_pk player      oaa opportunities catches stars5_opps stars5_catches
##   <dbl> <chr>     <dbl>       <int>    <dbl>       <int>       <int>
## 1 745455 Jacob Young  1.84        8       8       2       2
## 2 746097 Pete Crow-Arm~ 1.68        6       6       2       2
## 3 747064 Tyrone Taylor  1.64        6       6       2       2
## 4 745277 Julio Rodríguez 1.61        4       4       1       1
## 5 747121 Fernando Tati~  1.52        4       4       2       2
## 6 745184 Victor Scott ~  1.42        4       4       0       0
## 7 744844 Jacob Young  1.39        6       6       1       1
## 8 745716 Tyrone Taylor  1.31        3       3       1       1
## 9 745075 Pete Crow-Arm~  1.3        2       2       1       1
## 10 745713 Mickey Moniak 1.28       16      14       2       0
## # i 10 more variables: stars4_opps <int>, stars4_catches <int>,
## #   stars3_opps <int>, stars3_catches <int>, stars2_opps <int>,
## #   stars2_catches <int>, stars1_opps <int>, stars1_catches <int>,
## #   stars0_opps <int>, stars0_catches <int>
```

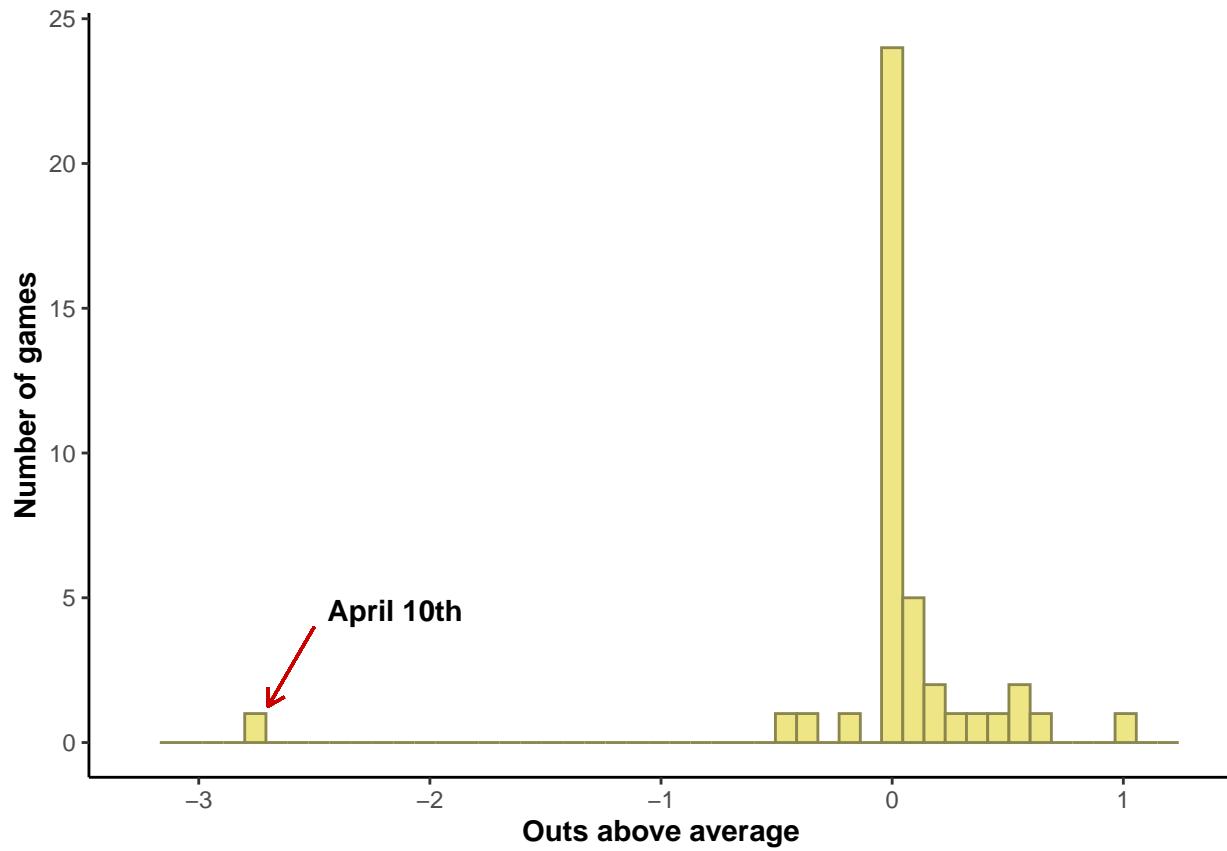
January 6th

```
single_game_oaa %>%
  arrange(oaa) %>%
  head(10)

## # A tibble: 10 x 17
##   game_pk player      oaa opportunities catches stars5_opps stars5_catches
##   <dbl> <chr>      <dbl>       <int>    <dbl>       <int>       <int>
## 1 745196 Victor Scott ~ -2.71        6       3         0         0
## 2 746262 MJ Melendez -2.57        6       2         0         0
## 3 746546 Charlie Black~ -2.34        5       1         1         0
## 4 746931 Tyler O'Neill -2.33        5       2         0         0
## 5 745708 Juan Soto     -2.18        6       2         0         0
## 6 746677 Will Benson   -2.11        6       3         1         0
## 7 745342 Luis Matos   -2.07        7       2         3         0
## 8 746481 Alec Burleson -2.04       10       6         2         0
## 9 746971 Ian Happ     -2.04        4       1         0         0
## 10 745184 Alec Burleson -1.98       2       0         0         0
## # i 10 more variables: stars4_opps <int>, stars4_catches <int>,
## # stars3_opps <int>, stars3_catches <int>, stars2_opps <int>,
## # stars2_catches <int>, stars1_opps <int>, stars1_catches <int>,
## # stars0_opps <int>, stars0_catches <int>

jan6 <- single_game_oaa %>%
  filter(player == "Victor Scott II") %>%
  ggplot() +
  geom_histogram(aes(x = oaa), bins = 50, fill = "khaki2", col = "khaki4") +
  geom_segment(x = -2.5, y = 4, xend = -2.7, yend = 1.25,
               arrow = arrow(length = unit(0.25, "cm")),
               col = "red3") +
  geom_text(x = -2.15, y = 4.55, label = "April 10th", check_overlap = TRUE, fontface = "bold")
  labs(x = "Outs above average", y = "Number of games") +
  xlim(c(-3.25, 1.25)) +
  theme_classic() + theme(axis.title = element_text(face = "bold"))

jan6
```



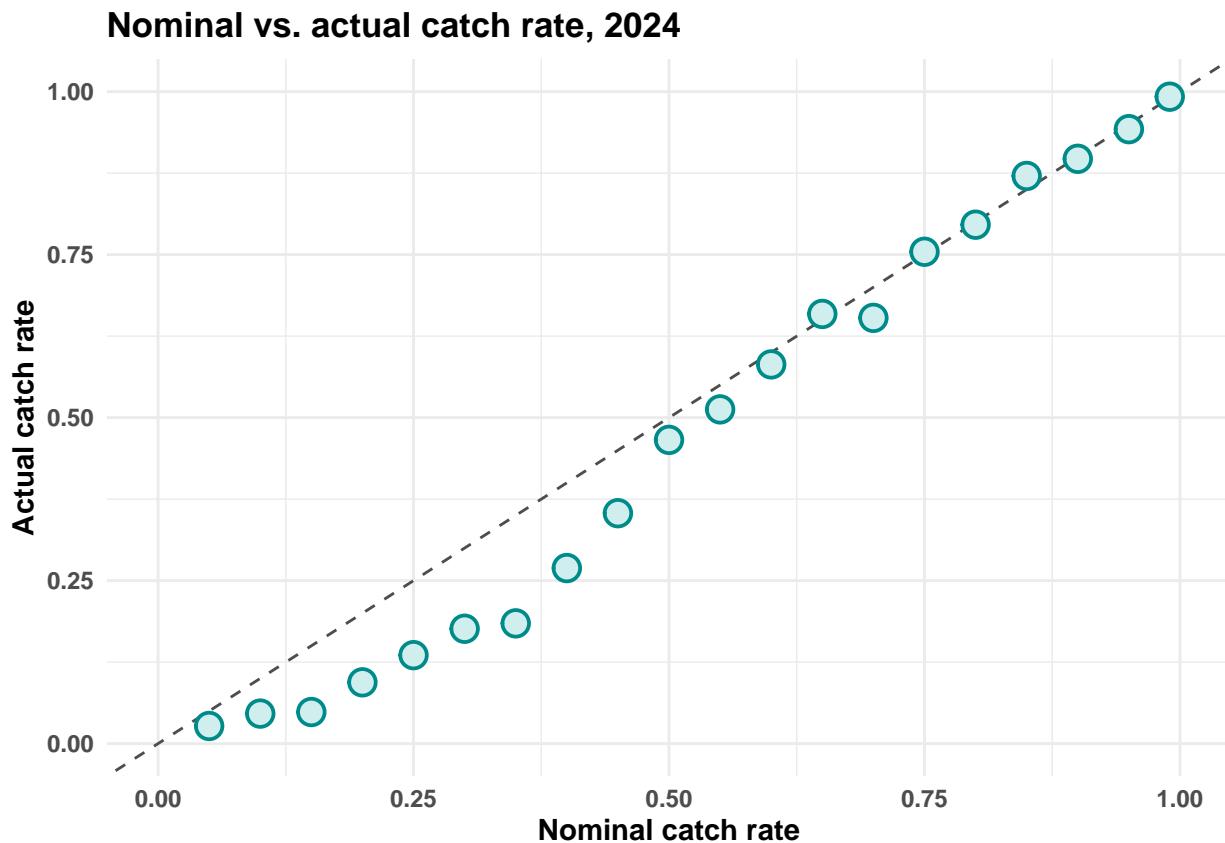
```
ggsave("../figures/jan6.png", plot = jan6, height = 4, width = 6)
```

January 7th

```
calibration_table_2024 <- of_catch_prob_2024 %>%
  group_by(catch_rate) %>%
  summarize(actual_catch_rate = mean(out)) %>%
  rename(nominal_catch_rate = catch_rate)

jan7_1 <- calibration_table_2024 %>%
  ggplot(aes(x = nominal_catch_rate, y = actual_catch_rate)) +
  geom_abline(intercept = 0, slope = 1,
              color = "gray30", linetype = "dashed", linewidth = 0.5) +
  geom_point(pch = 21, col = "cyan4", fill = "lightcyan2",
             size = 4, stroke = 1) +
  theme_minimal() +
  lims(x = c(0,1), y = c(0,1)) +
  labs(x = "Nominal catch rate", y = "Actual catch rate",
       title = "Nominal vs. actual catch rate, 2024") +
  theme(axis.title = element_text(face = "bold"),
        axis.text = element_text(face = "bold"),
        title = element_text(face = "bold"))

jan7_1
```



```
ggsave("../figures/jan7_1.png", plot = jan7_1, height = 4, width = 6)
```

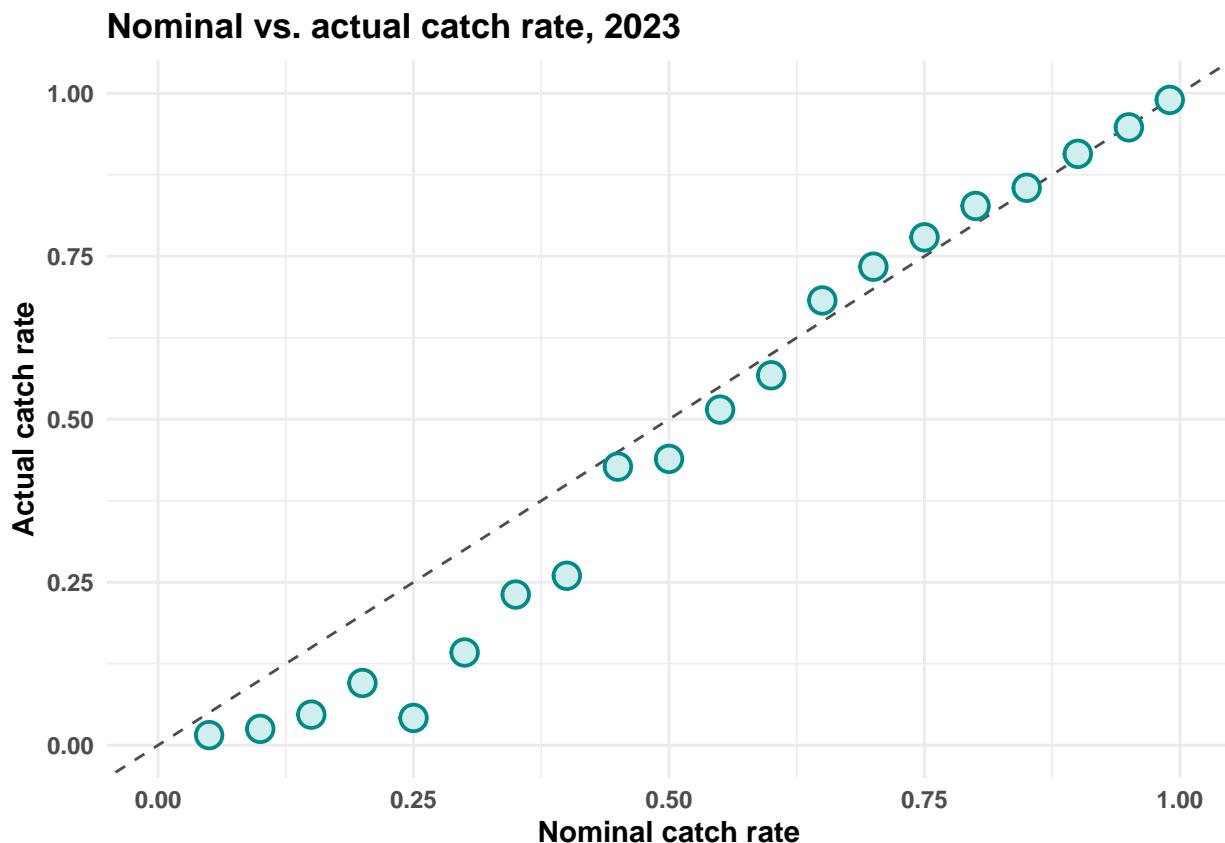
```

calibration_table_2023 <- of_catch_prob_2023 %>%
  group_by(catch_rate) %>%
  summarize(actual_catch_rate = mean(out)) %>%
  rename(nominal_catch_rate = catch_rate)

jan7_2 <- calibration_table_2023 %>%
  ggplot(aes(x = nominal_catch_rate, y = actual_catch_rate)) +
  geom_abline(intercept = 0, slope = 1,
              color = "gray30", linetype = "dashed", linewidth = 0.5) +
  geom_point(pch = 21, col = "cyan4", fill = "lightcyan2",
             size = 4, stroke = 1) +
  theme_minimal() +
  lims(x = c(0,1), y = c(0,1)) +
  labs(x = "Nominal catch rate", y = "Actual catch rate",
       title = "Nominal vs. actual catch rate, 2023") +
  theme(axis.title = element_text(face = "bold"),
        axis.text = element_text(face = "bold"),
        title = element_text(face = "bold"))

```

jan7_2



```
ggsave("../figures/jan7_2.png", plot = jan7_2, height = 4, width = 6)
```

January 8th

```
of_oaa_2024 <- read_csv("../data/of_oaa_2024.csv")

oaa_rounding_check <- of_catch_prob_2024 %>%
  mutate(catch_rate_upper = pmin(0.999999, catch_rate - 0.02),
         catch_rate_lower = pmin(0.999999, catch_rate + 0.02)) %>%
  group_by(player_id, player) %>%
  summarize(opp = n(),
            oaa_lower = round(sum(out * (1 - catch_rate_lower) - (1 - out) * catch_rate_lower)),
            oaa_estimate = sum(out * (1 - catch_rate) - (1 - out) * catch_rate),
            oaa_upper = round(sum(out * (1 - catch_rate_upper) - (1 - out) * catch_rate_upper)))
  ungroup() %>%
  left_join(of_oaa_2024, by = "player_id") %>%
  select(player_id, name = player, opp,
         oaa_lower, oaa_estimate, oaa_upper, oaa_true = oaa)

oaa_rounding_check %>%
  summarize(any(oaa_true < oaa_lower | oaa_true > oaa_upper))

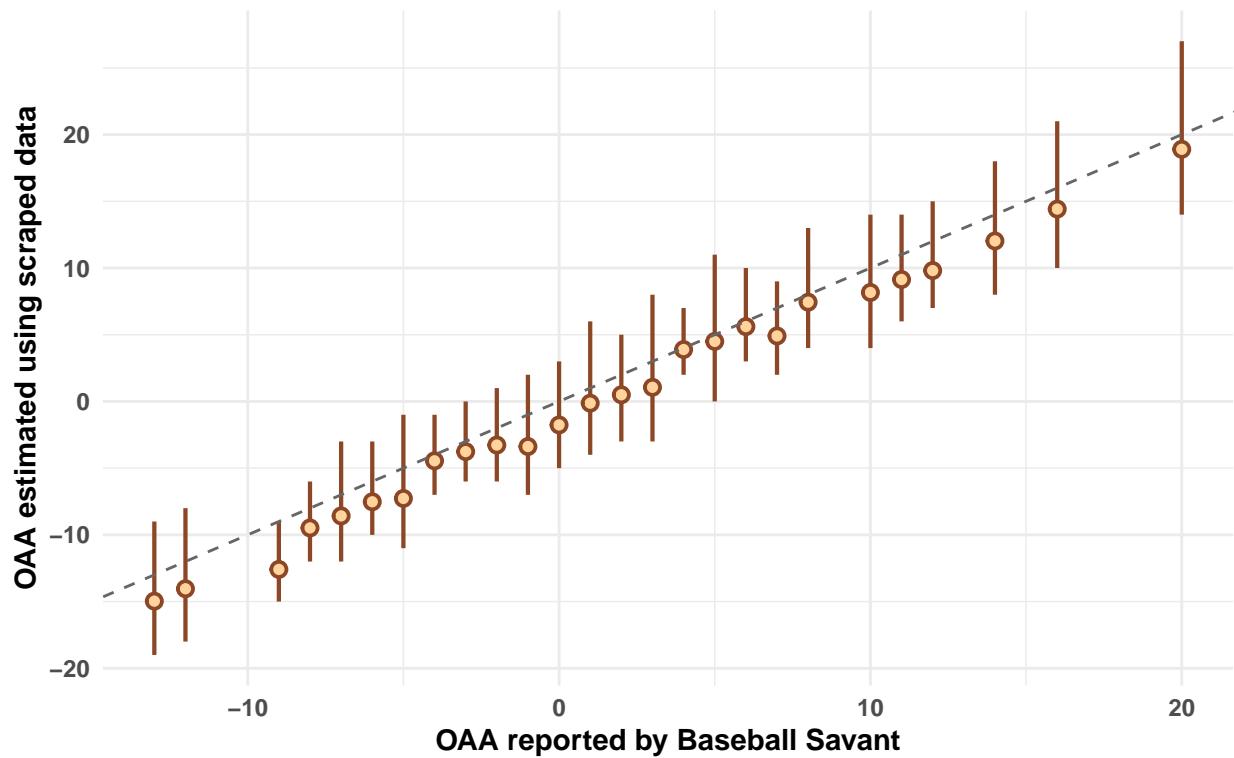
## # A tibble: 1 x 1
##   `any(oaa_true < oaa_lower | oaa_true > oaa_upper)`
##   <lgl>
##   1 FALSE

set.seed(0)
jan8 <- oaa_rounding_check %>%
  filter(opp > 162) %>%
  group_by(oaa_true) %>%
  sample_n(1) %>%
  ggplot(aes(x = oaa_true, y = oaa_estimate)) +
  geom_linerange(aes(ymin = oaa_lower, ymax = oaa_upper), col = "sienna4", linewidth = 0.75) +
  geom_point(size = 2, stroke = 1, pch = 21, fill = "burlywood1", col = "sienna4") +
  geom_abline(intercept = 0, slope = 1, linetype = "dashed", col = "gray40") +
  theme_minimal() +
  labs(x = "OAA reported by Baseball Savant",
       y = "OAA estimated using scraped data",
       title = "Actual vs. estimated OAA for selected players, 2024",
       subtitle = "Lower and upper bounds computed by changing each catch rate by +/-0.02") +
  theme(axis.title = element_text(face = "bold"),
        axis.text = element_text(face = "bold"),
        title = element_text(face = "bold"))

jan8
```

Actual vs. estimated OAA for selected players, 2024

Lower and upper bounds computed by changing each catch rate by +/-0.02

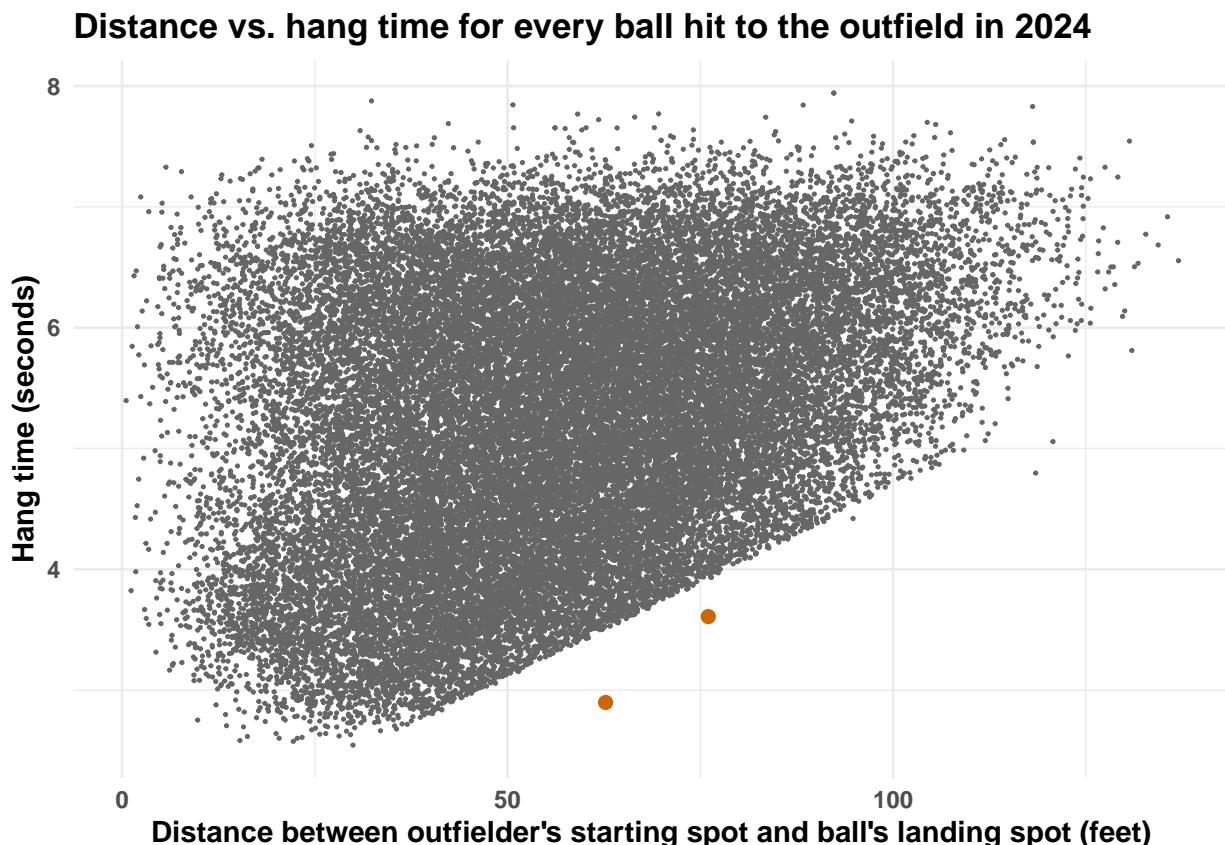


```
ggsave("../figures/jan8.png", plot = jan8, height = 4, width = 6)
```

January 9th

```
jan9 <- of_catch_prob_2024 %>%
  mutate(highlight = (hang_time < 3 & distance > 50) | (hang_time < 3.8 & distance > 75)) %>%
  ggplot(aes(x = distance, y = hang_time, col = highlight, size = highlight)) +
  geom_point() +
  scale_colour_manual(values = c("gray40", "darkorange3")) +
  scale_size_manual(values = c(0.25, 2)) +
  theme_minimal() +
  theme(legend.position = "none",
        axis.title = element_text(face = "bold"),
        axis.text = element_text(face = "bold"),
        title = element_text(face = "bold")) +
  labs(x = "Distance between outfielder's starting spot and ball's landing spot (feet)",
       y = "Hang time (seconds)",
       title = "Distance vs. hang time for every ball hit to the outfield in 2024")
```

```
jan9
```



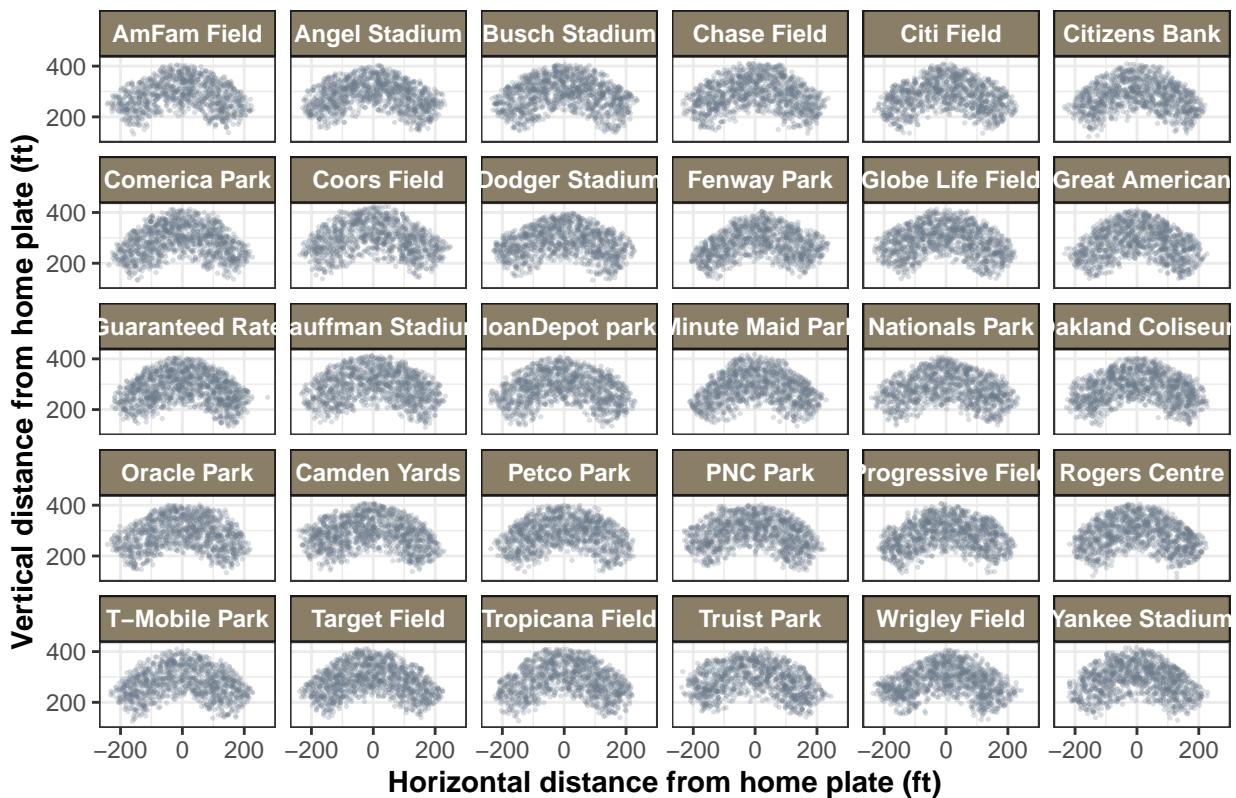
```
ggsave("../figures/jan9.png", plot = jan9, height = 4, width = 6)
```

January 17th

```
jan17_1 <- of_catch_prob_2024 %>%
  mutate(stadium = fct_recode(stadium,
    "AmFam Field" = "American Family Field",
    "Guaranteed Rate" = "Guaranteed Rate Field",
    "Camden Yards" = "Oriole Park at Camden Yards",
    "Great American" = "Great American Ball Park",
    "Citizens Bank" = "Citizens Bank Park"
  )) %>%
  ggplot(aes(x = landing_pos_x, y = landing_pos_y)) +
  geom_point(size = 0.25, alpha = 0.25, col = "slategray4") +
  facet_wrap(~stadium, scales = "fixed") +
  theme_bw() +
  labs(x = "Horizontal distance from home plate (ft)",
       y = "Vertical distance from home plate (ft)",
       title = "Landing positions of fly balls by stadium, 2024") +
  theme(axis.title = element_text(face = "bold"),
        title = element_text(face = "bold"),
        plot.subtitle = element_text(face = "italic"),
        strip.background = element_rect(fill = "wheat4", color = "gray10"),
        strip.text = element_text(face = "bold", color = "white")) +
  scale_x_continuous(breaks = c(-200, 0, 200)) +
  scale_y_continuous(breaks = c(200, 400))
```

```
jan17_1
```

Landing positions of fly balls by stadium, 2024

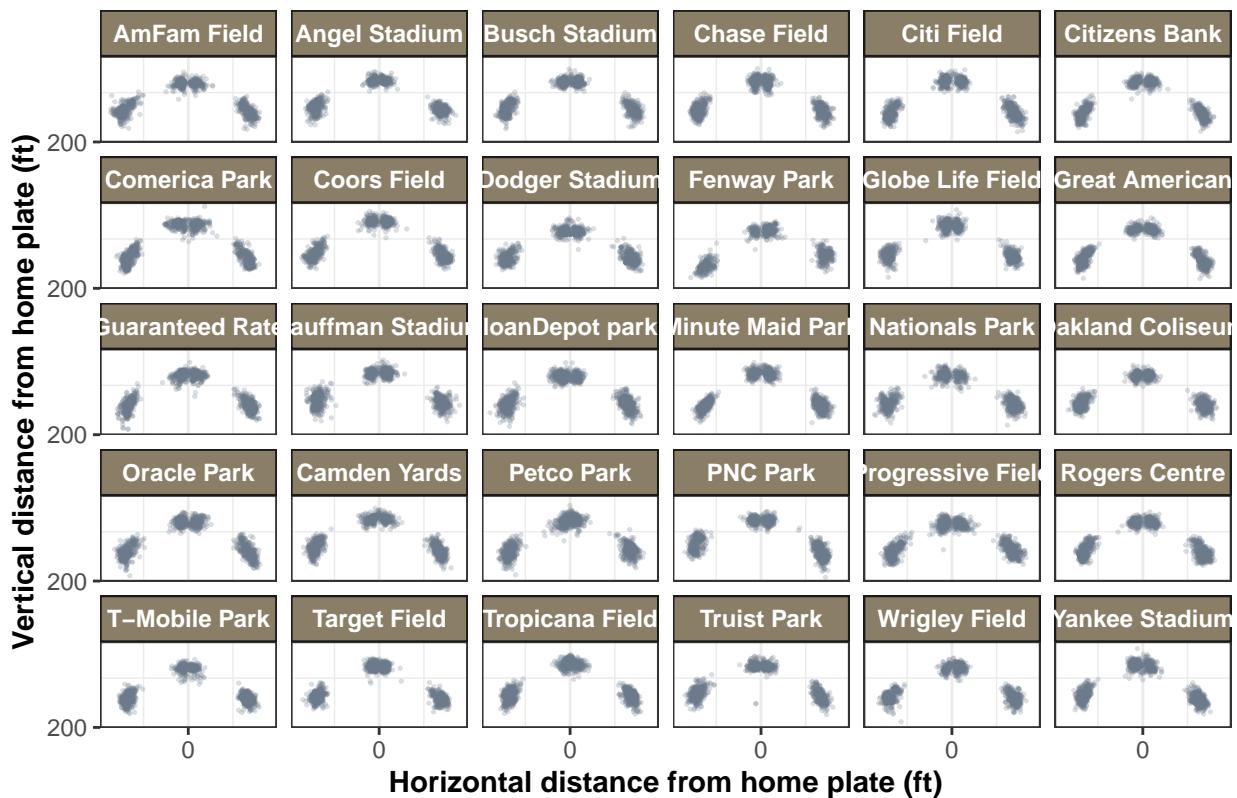


```
ggsave("../figures/jan17_1.png", plot = jan17_1, height = 6, width = 8)
```

```
jan17_2 <- of_catch_prob_2024 %>%
  mutate(stadium = fct_recode(stadium,
    "AmFam Field" = "American Family Field",
    "Guaranteed Rate" = "Guaranteed Rate Field",
    "Camden Yards" = "Oriole Park at Camden Yards",
    "Great American" = "Great American Ball Park",
    "Citizens Bank" = "Citizens Bank Park"
  )) %>%
  ggplot(aes(x = start_pos_x, y = start_pos_y)) +
  geom_point(size = 0.25, alpha = 0.25, col = "slategray4") +
  facet_wrap(~stadium, scales = "fixed") +
  theme_bw() +
  labs(x = "Horizontal distance from home plate (ft)",
       y = "Vertical distance from home plate (ft)",
       title = "Starting positions of outfielders by stadium, 2024") +
  theme(axis.title = element_text(face = "bold"),
        title = element_text(face = "bold"),
        plot.subtitle = element_text(face = "italic"),
        strip.background = element_rect(fill = "wheat4", color = "gray10"),
        strip.text = element_text(face = "bold", color = "white")) +
  scale_x_continuous(breaks = c(-200, 0, 200)) +
  scale_y_continuous(breaks = c(200, 400))
```

jan17_2

Starting positions of outfielders by stadium, 2024



```
ggsave("../figures/jan17_2.png", plot = jan17_2, height = 6, width = 8)
```

January 18th

```
stadium_dims <- of_catch_prob_2024 %>%
  distinct(stadium, .keep_all = TRUE) %>%
  select(stadium, starts_with("wall_height"), starts_with("dist_"))

pca <- prcomp(stadium_dims %>% select(-stadium), center = TRUE, scale = TRUE)

pca

## Standard deviations (1, ..., p=8):
## [1] 1.5608351 1.3872951 1.1309176 0.9586234 0.8183889 0.7235980 0.4088130
## [8] 0.2842364
##
## Rotation (n x k) = (8 x 8):
##          PC1        PC2        PC3        PC4        PC5
## wall_height_lf -0.1898228  0.62625201  0.11383912  0.12626456 -0.17220394
## wall_height_cf -0.3241882  0.11587668  0.51041136  0.19115723  0.68654340
## wall_height_rf  0.3294967 -0.16656010 -0.22990356  0.73144588  0.03023318
## dist_lf         0.5400868 -0.12444576  0.40096509  0.06355490 -0.11218946
## dist_lc         -0.1771522 -0.63331386 -0.03814045 -0.01869318  0.32002690
## dist_cf         -0.3963460 -0.32632283  0.03568282 -0.25822209 -0.37829349
## dist_rc         -0.2326441 -0.20735180  0.58578738  0.37790652 -0.48417874
## dist_rf         0.4647701 -0.02417013  0.40867766 -0.44567080  0.07450642
##          PC6        PC7        PC8
## wall_height_lf  0.24788291 -0.660375806  0.12520901
## wall_height_cf  0.21961532  0.244505757  0.07111942
## wall_height_rf  0.47518822 -0.023731803 -0.22031770
## dist_lf          0.01680321 -0.020397652  0.71744157
## dist_lc          -0.10543733 -0.667675727  0.08000882
## dist_cf          0.68733639  0.150835796  0.17373145
## dist_rc          -0.31352254 -0.002968767 -0.29015087
## dist_rf          0.28704052 -0.185984784 -0.54335153

summary(pca)

## Importance of components:
##          PC1       PC2       PC3       PC4       PC5       PC6       PC7
## Standard deviation 1.5608 1.3873 1.1309 0.9586 0.81839 0.72360 0.40881
## Proportion of Variance 0.3045 0.2406 0.1599 0.1149 0.08372 0.06545 0.02089
## Cumulative Proportion 0.3045 0.5451 0.7050 0.8198 0.90356 0.96901 0.98990
##          PC8
## Standard deviation 0.2842
## Proportion of Variance 0.0101
## Cumulative Proportion 1.0000

set.seed(18)

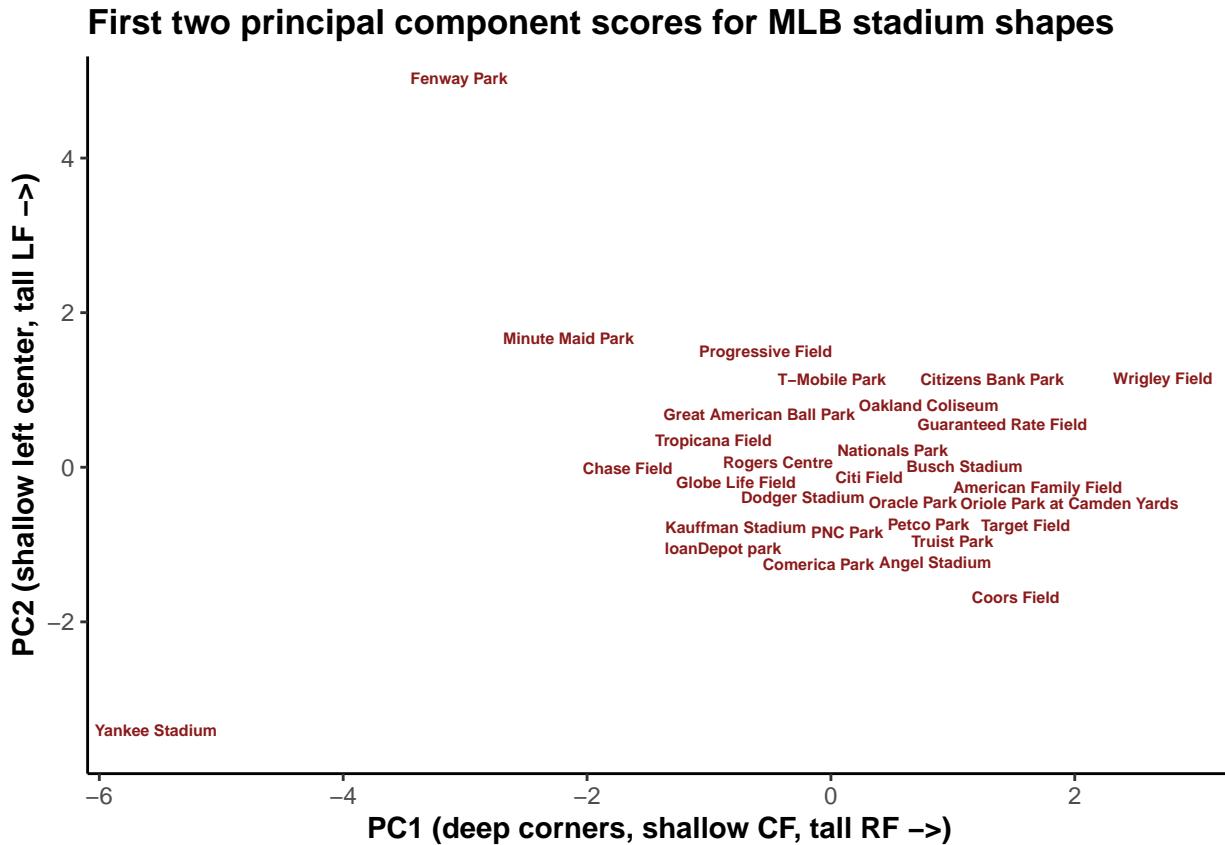
jan18 <- tibble(stadium = stadium_dims$stadium, pc1 = pca$x[,1], pc2 = pca$x[,2]) %>%
  ggplot(aes(x = pc1, y = pc2)) +
  ggrepel::geom_text_repel(aes(label = stadium), fontface = "bold", col = "firebrick4",
```

```

min.segment.length = 1,
label.padding = 0.0, box.padding = 0.05, point.padding = 0,
size = 2, max.overlaps = 30) +
theme_classic() +
labs(title = "First two principal component scores for MLB stadium shapes",
x = "PC1 (deep corners, shallow CF, tall RF ->)",
y = "PC2 (shallow left center, tall LF ->)") +
theme(axis.title = element_text(face = "bold"),
title = element_text(face = "bold"))

```

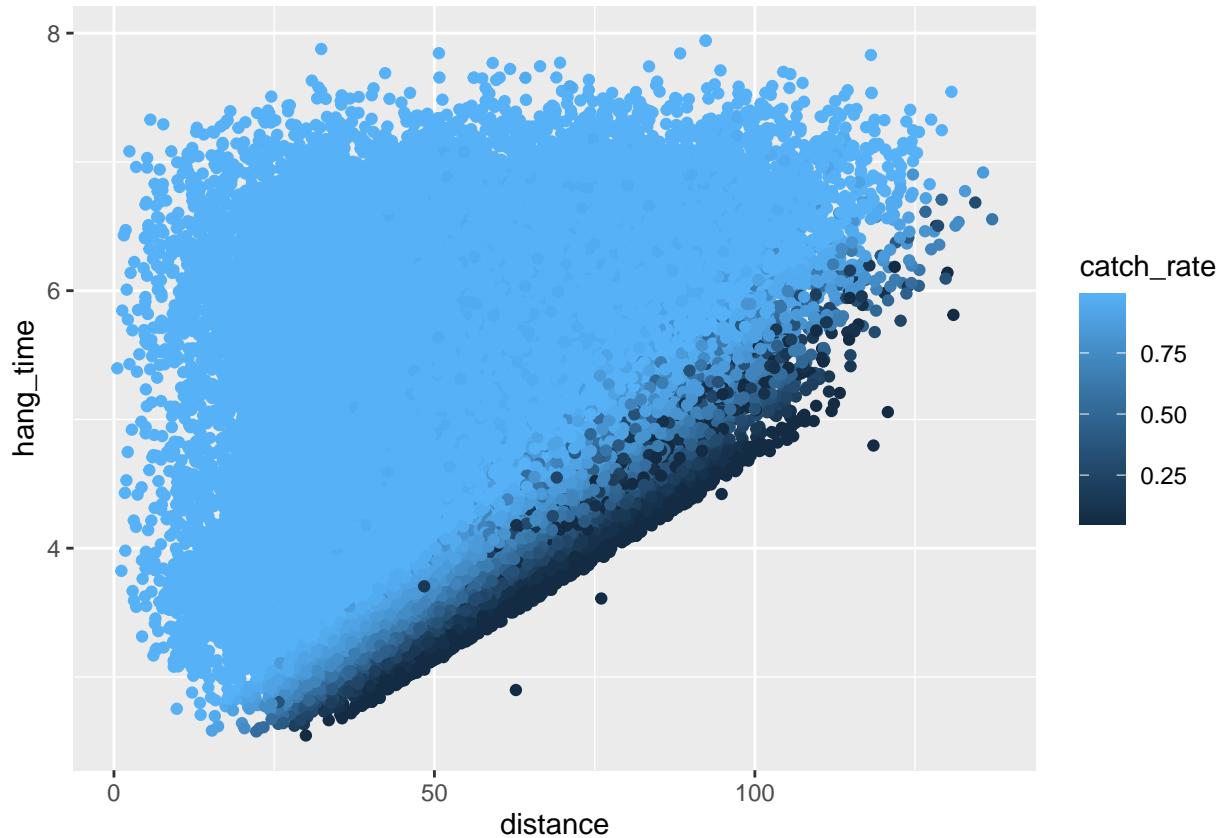
jan18



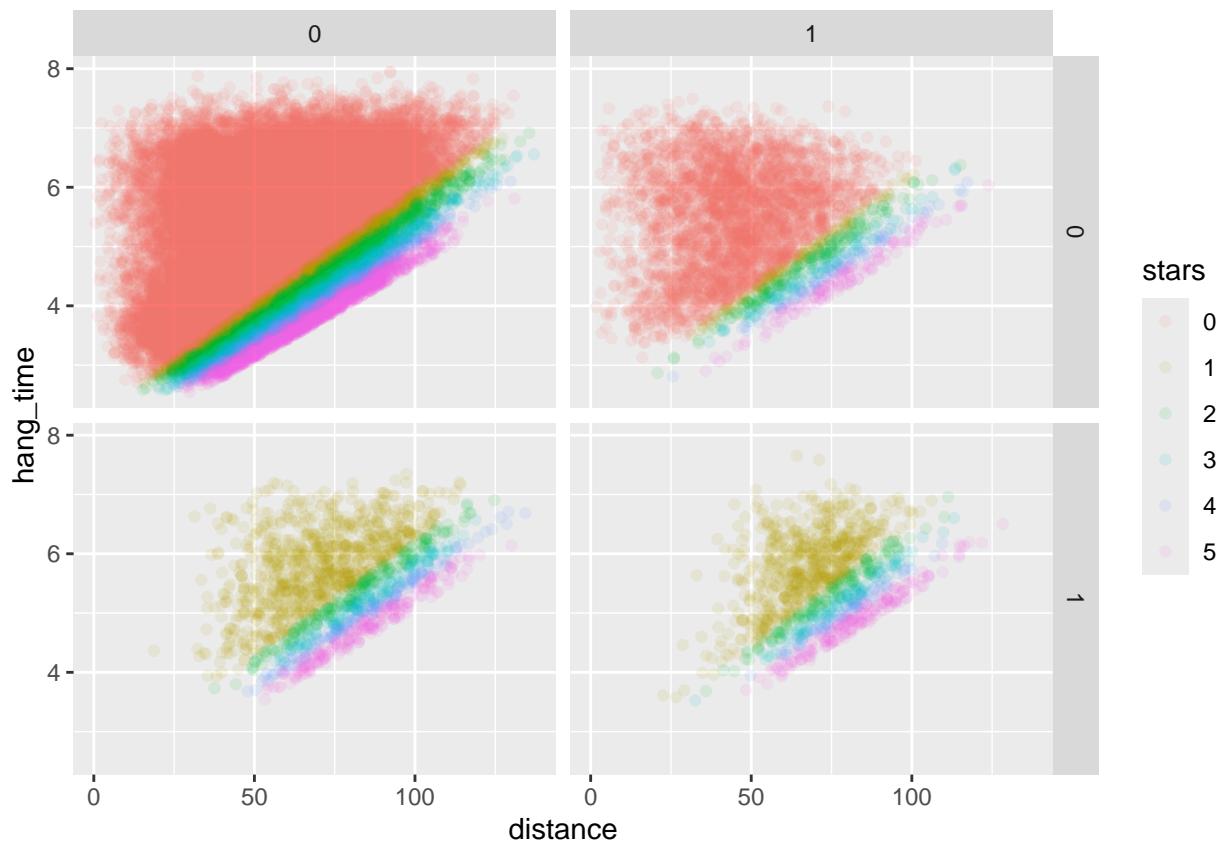
```
ggsave("../figures/jan18.png", plot = jan18, height = 4, width = 6)
```

Later

```
of_catch_prob_2024 %>%
  ggplot(aes(x = distance, y = hang_time, col = catch_rate)) +
  geom_point(alpha = 1)
```



```
of_catch_prob_2024 %>%
  filter(stars <= 5) %>%
  mutate(stars = as.factor(stars)) %>%
  ggplot(aes(x = distance, y = hang_time, col = stars)) +
  geom_point(alpha = 0.1) +
  facet_grid(rows = vars(wall), cols = vars(back))
```



```
of_catch_prob_2024 %>%
  filter(stars <= 5) %>%
  mutate(stars = as.factor(stars)) %>%
  ggplot(aes(x = distance, y = hang_time, col = catch_rate)) +
  geom_point(alpha = 0.1) +
  facet_grid(rows = vars(wall), cols = vars(back))
```

