

# Catch probability

Tim White

```
rSys.Date()
```

```
library(tidyverse)
library(rjson)
```

## 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.

```
# Outfielders' player IDs
playerIDs <- read_csv("../data/of_playerIDs_2024.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=2024&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
      }

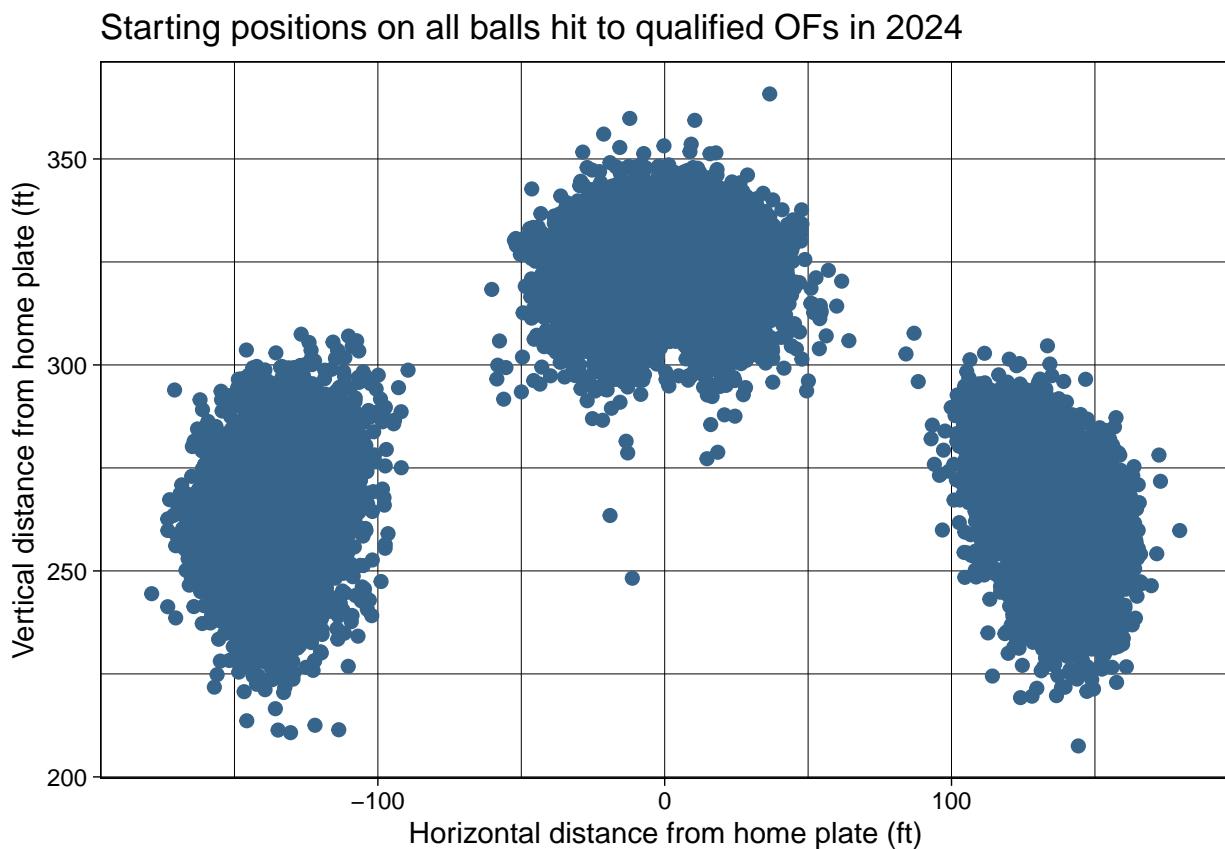
      # Convert raw data to tibble
      tibble(data.frame(matrix(unlist(rawdata),
                                nrow = length(rawdata),
                                byrow = TRUE,
                                dimnames = list(1:length(rawdata),
                                                names(rawdata[[1]])))),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

```
)  
  
data_final <- tibble(do.call(rbind.data.frame, data))  
  
# Write tibble to csv  
data_final %>% write_csv("../data/of_catch_prob_2024.csv")  
  
# Load in data  
data_final <- read_csv("../data/of_catch_prob_2024.csv")
```

## January 2nd

```
jan2_1 <- data_final %>%
  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
```

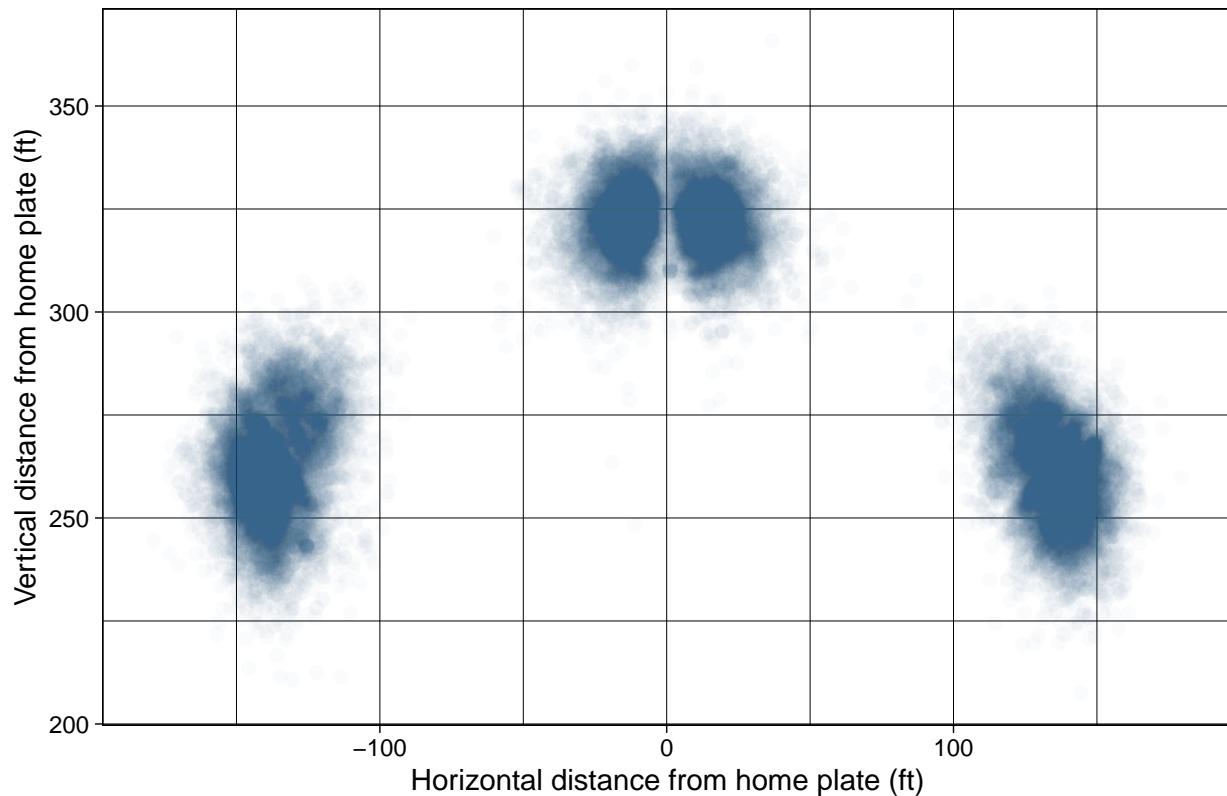


```
ggsave("../figures/jan2_1.png", plot = jan2_1)
```

```
jan2_2 <- data_final %>%
  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

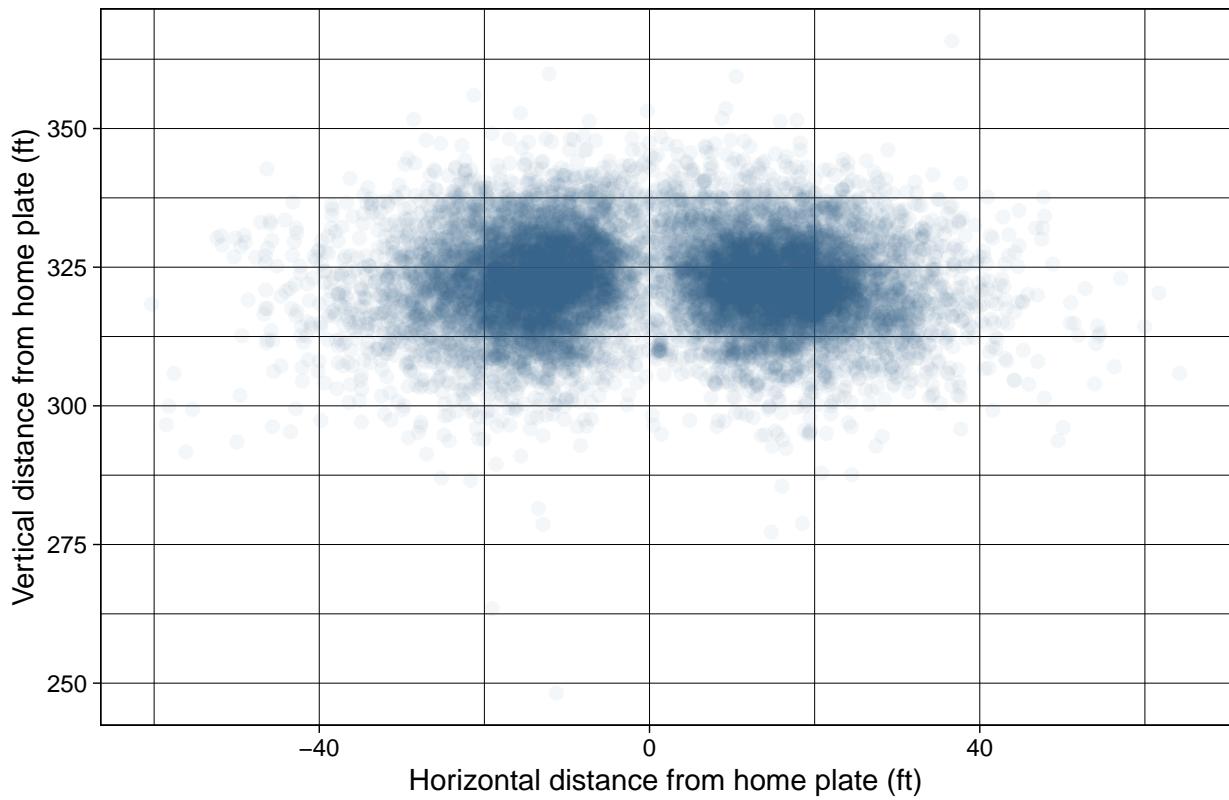


```
ggsave("../figures/jan2_2.png", plot = jan2_2)
```

```
jan2_3 <- data_final %>%
  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 <- data_final %>%
  group_by(game_pk, name_display_first_last) %>%
  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 name_display_first_last    oaa opportunities catches stars5_opps
##   <dbl> <chr>              <dbl>       <int>     <dbl>       <int>
## 1 746097 Pete Crow-Armstrong 1.68        6       6         2
## 2 745277 Julio Rodriguez 1.61        4       4         1
## 3 744844 Jacob Young 1.39        6       6         1
## 4 745716 Tyrone Taylor 1.31        3       3         1
## 5 745075 Pete Crow-Armstrong 1.3        2       2         1
## 6 745498 Michael A. Taylor 1.27        4       4         1
## 7 746101 Pete Crow-Armstrong 1.27        5       4         1
## 8 744997 Garrett Hampson 1.25        3       3         0
## 9 746030 Vidal Bruj  n 1.25        2       2         1
## 10 745270 Julio Rodriguez 1.22        5       5         1
## # i 11 more variables: stars5_catches <int>, 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>
```

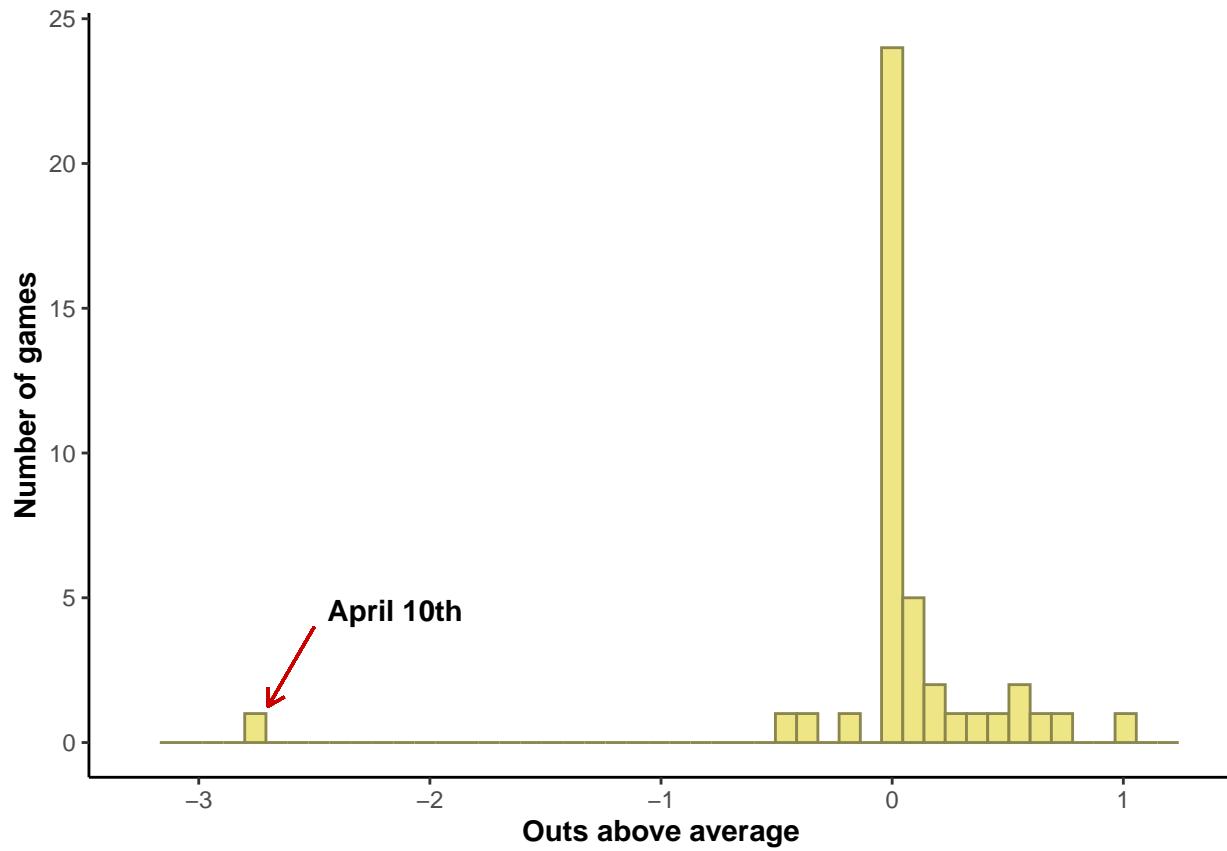
## January 6th

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

## # A tibble: 10 x 17
##   game_pk name_display_first_last   oaa opportunities catches stars5_opps
##   <dbl> <chr>                <dbl>      <int>    <dbl>      <int>
## 1 745196 Victor Scott II       -2.71        6       3       0
## 2 746262 MJ Melendez         -2.57        6       2       0
## 3 746546 Charlie Blackmon    -2.34        5       1       1
## 4 746931 Tyler O'Neill       -2.33        5       2       0
## 5 746677 Will Benson         -2.11        6       3       1
## 6 745342 Luis Matos          -2.07        7       2       3
## 7 746971 Ian Happ             -2.04        4       1       0
## 8 745300 Ramón Laureano     -1.98        2       0       0
## 9 746546 Jack Suwinski       -1.97        6       2       1
## 10 746525 Ceddanne Rafaela   -1.94        5       2       1
## # i 11 more variables: stars5_catches <int>, 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(name_display_first_last == "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
```



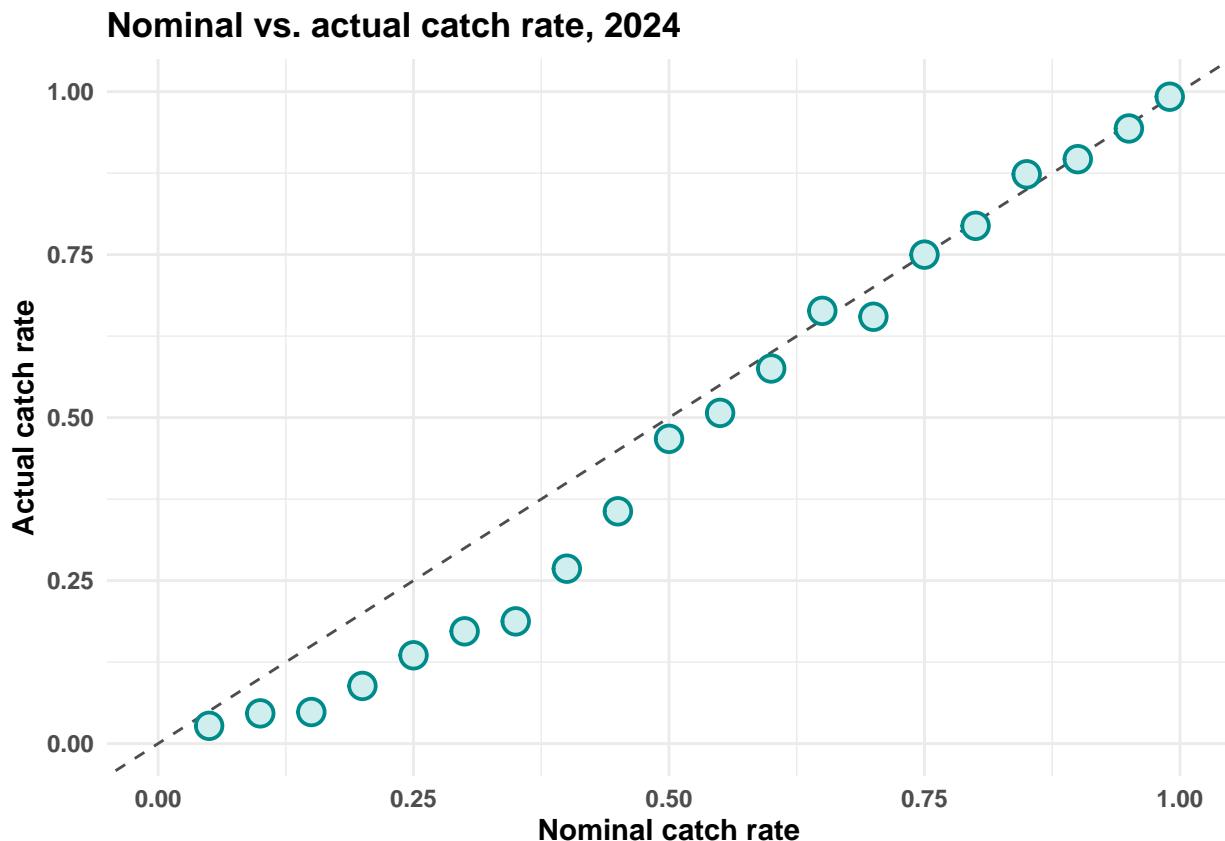
```
ggsave("../figures/jan6.png", plot = jan6, height = 4, width = 6)
```

## January 7th

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

jan7 <- calibration_table %>%
  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"))

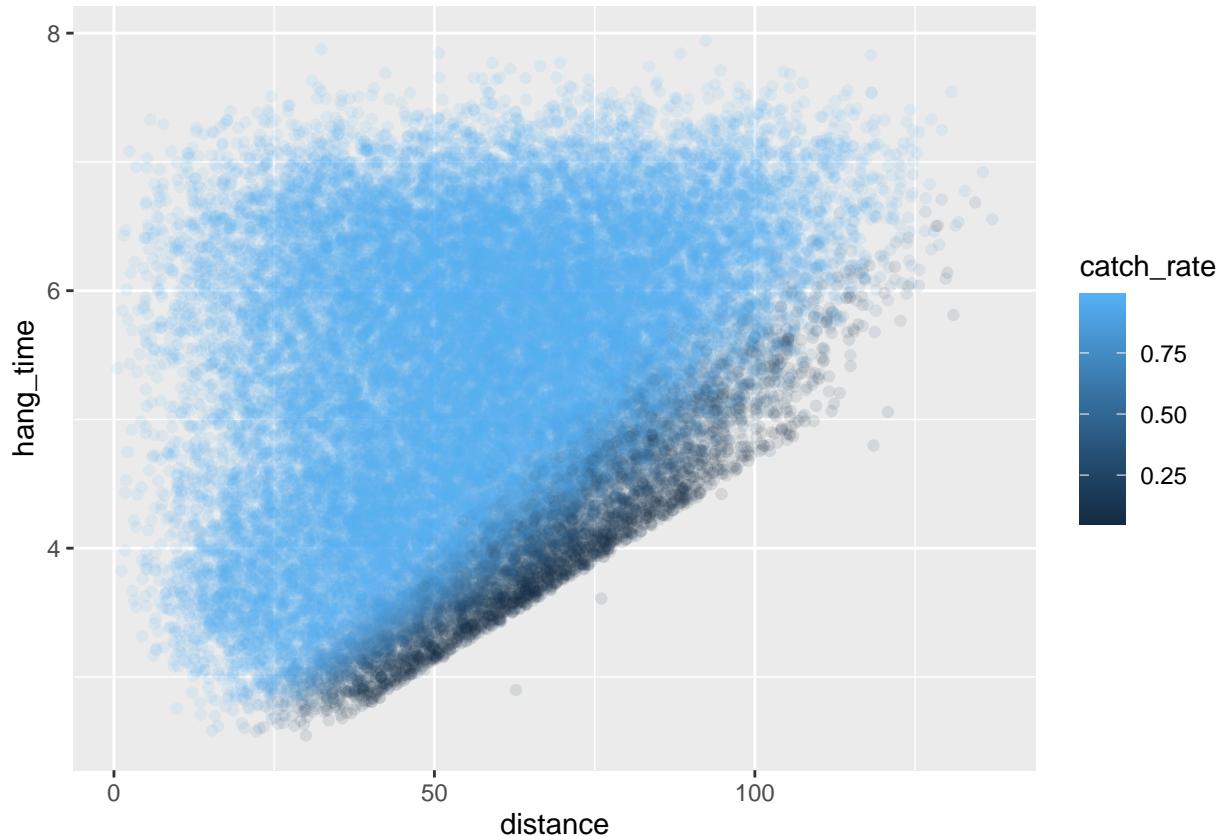
jan7
```



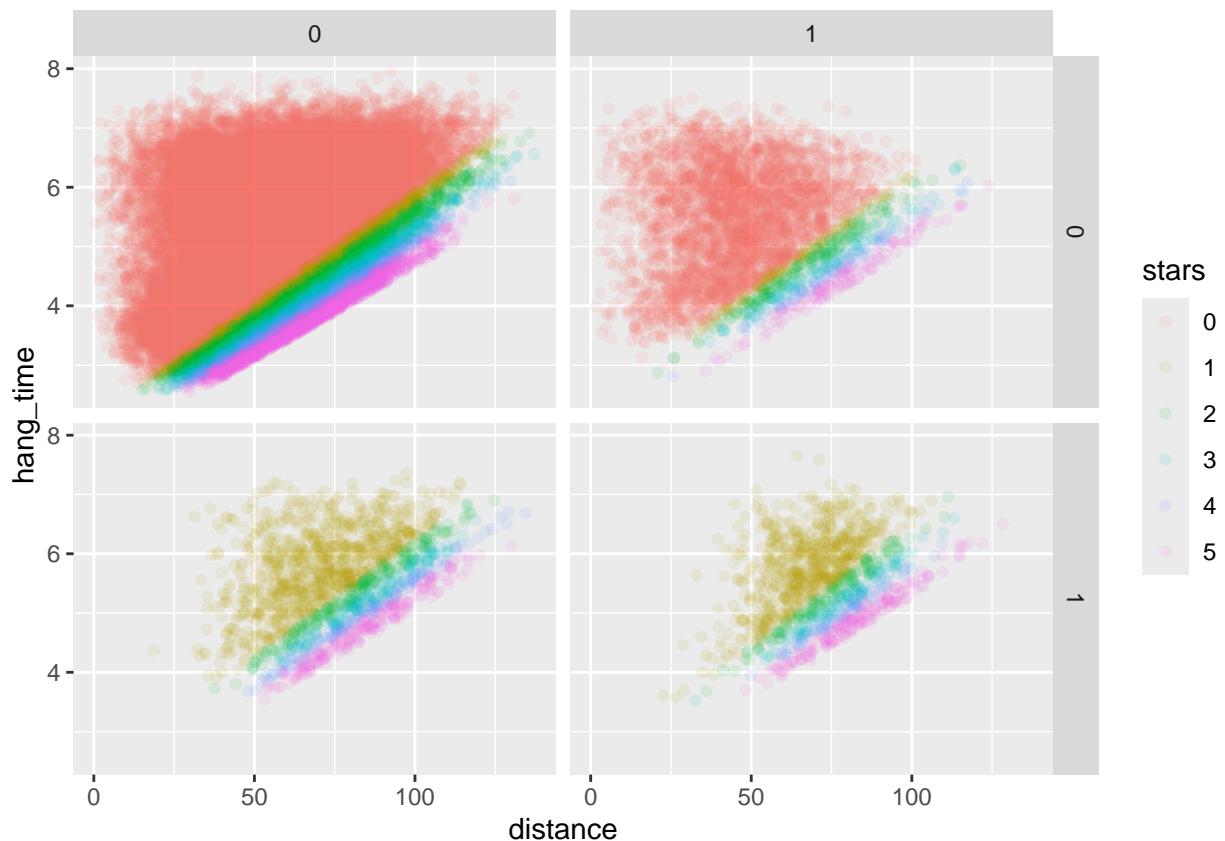
```
ggsave("../figures/jan7.png", plot = jan7, height = 4, width = 6)
```

## Later

```
data_final %>%
  ggplot(aes(x = distance, y = hang_time, col = catch_rate)) +
  geom_point(alpha = 0.1)
```



```
data_final %>%
  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))
```



```
data_final %>%
  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))
```

