

Catch probability

Tim White

2025-01-09

```
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.

```
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)))
}

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

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

Load in data

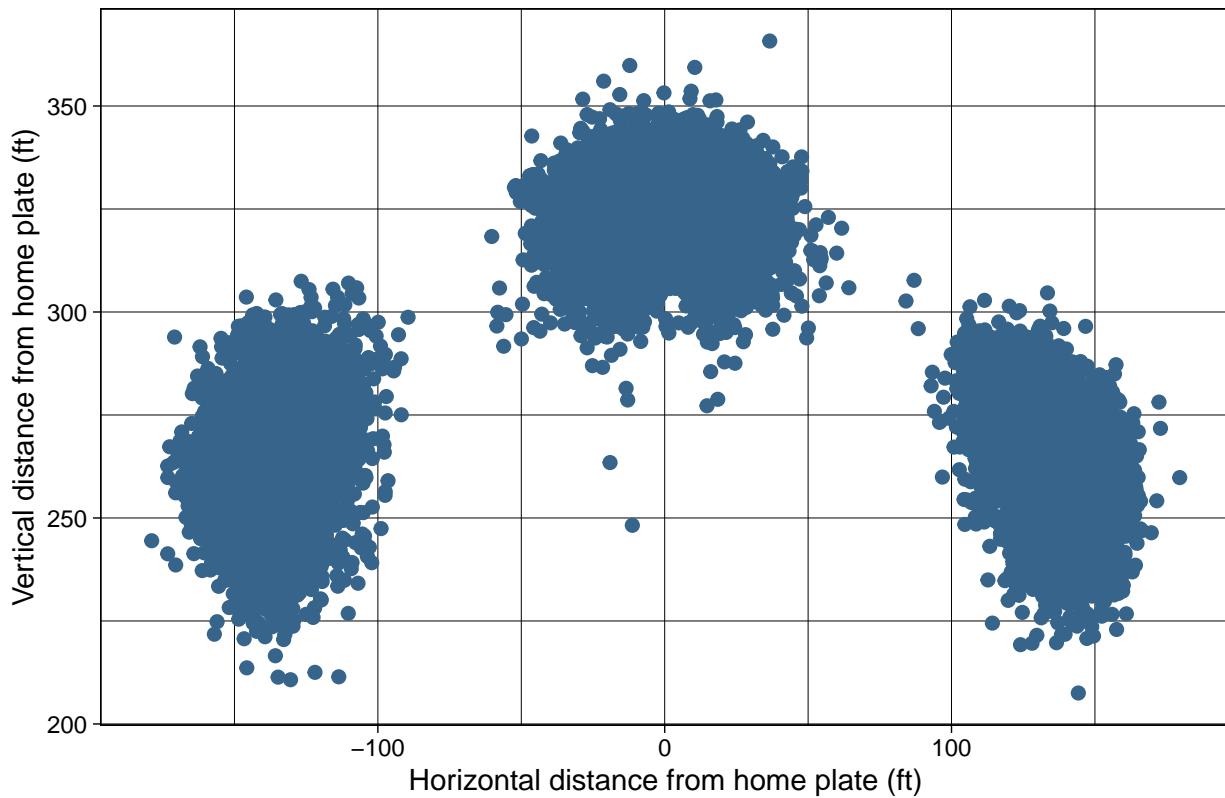
```
of_catch_prob_2024 <- read_csv("../data/of_catch_prob_2024.csv") %>%
  filter(!is.na(sprint_speed)) %>%
  mutate(game_pk = as.factor(game_pk),
         play_id = as.factor(play_id),
         name_display_first_last = as.factor(name_display_first_last),
         pos = as.factor(pos))
of_catch_prob_2023 <- read_csv("../data/of_catch_prob_2023.csv") %>%
  filter(!is.na(sprint_speed)) %>%
  mutate(game_pk = as.factor(game_pk),
         play_id = as.factor(play_id),
         name_display_first_last = as.factor(name_display_first_last),
         pos = as.factor(pos))
```

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
```

Starting positions on all balls hit to qualified OFs in 2024

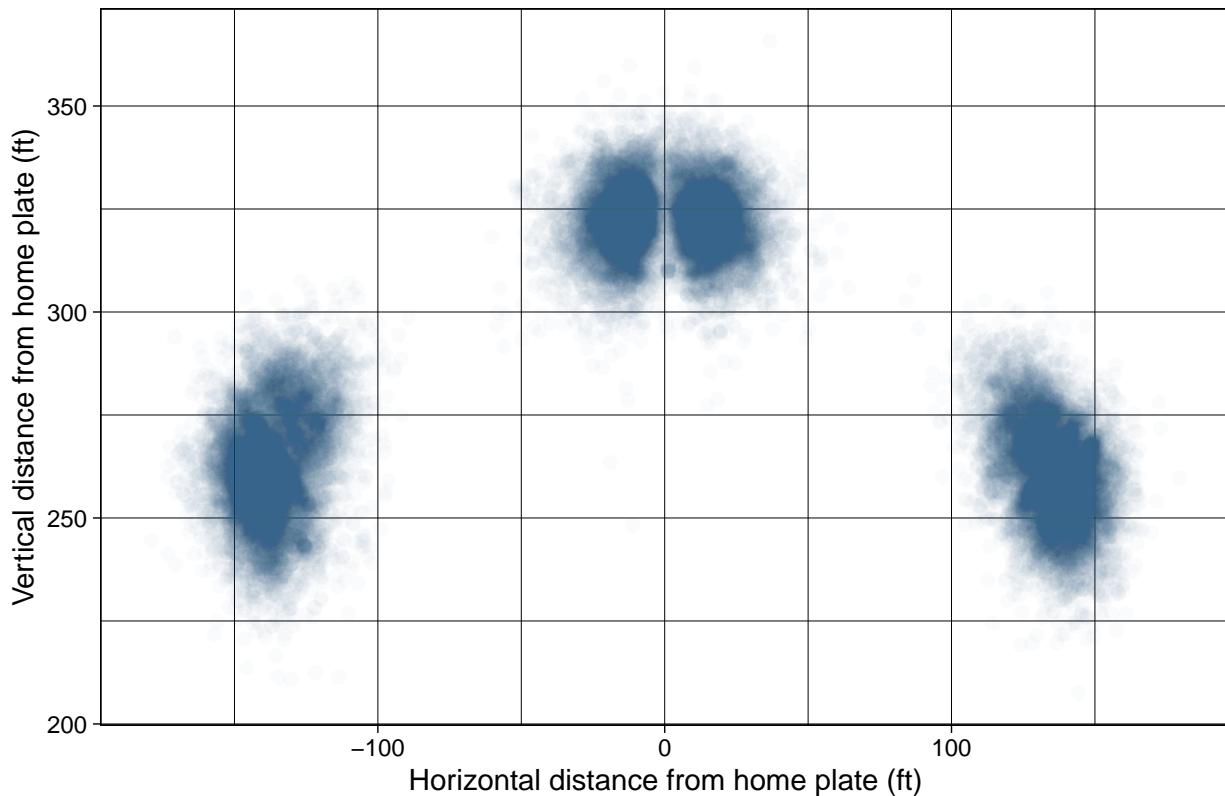


```
ggsave("../figures/jan2_1.png", plot = jan2_1)
```

```
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

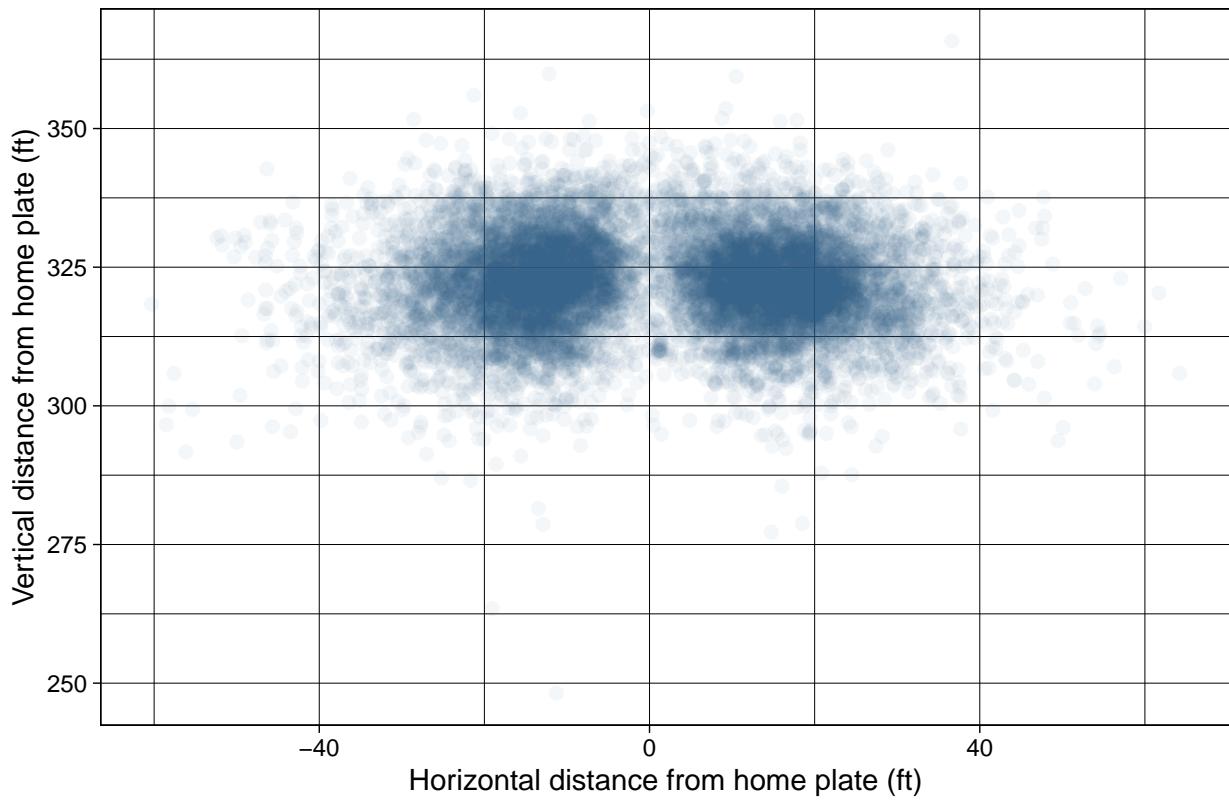


```
ggsave("../figures/jan2_2.png", plot = jan2_2)
```

```
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, 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
##   <fct>    <fct>          <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 Bruján         1.25         2       2           1
## 10 745270  Julio Rodríguez      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>
```

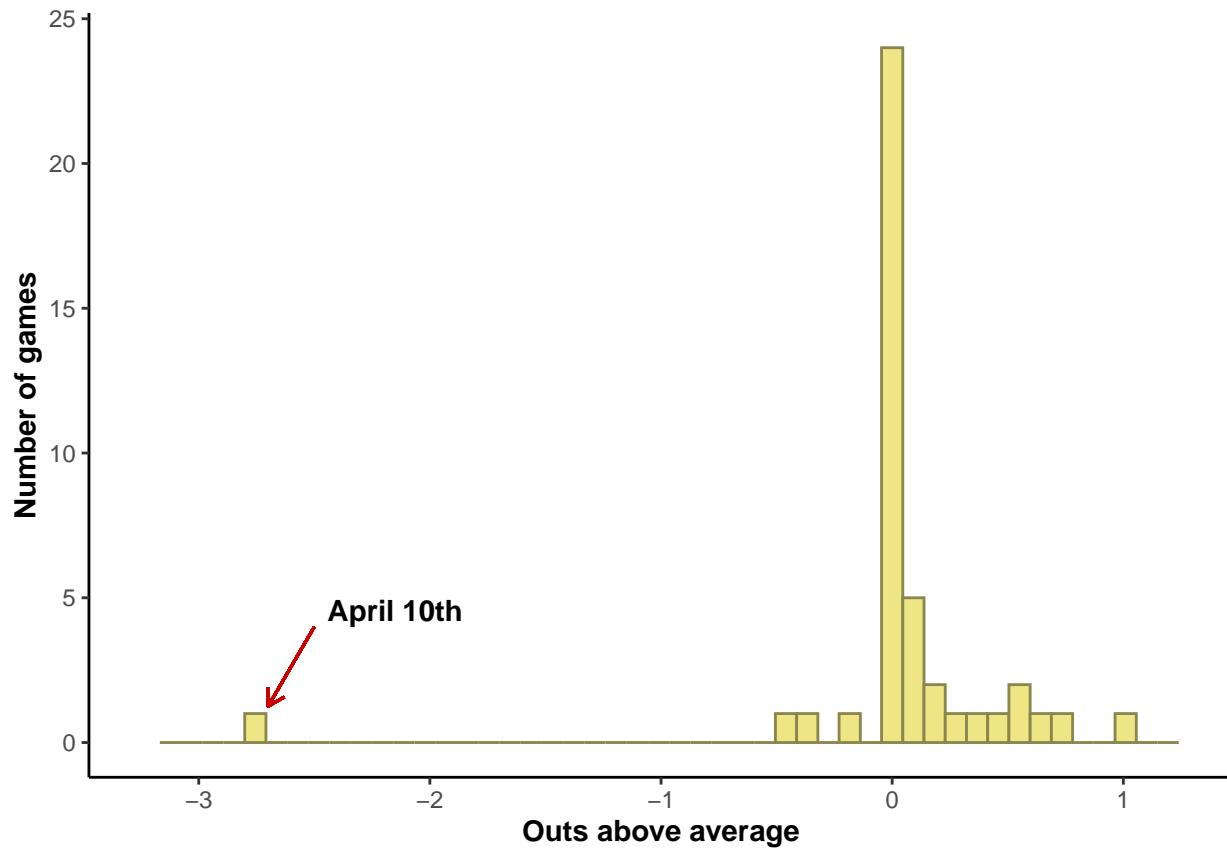
January 6th

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

## # A tibble: 10 x 17
##   game_pk name_display_first_last   oaa opportunities catches stars5_opps
##   <dbl>   <fct>           <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
```



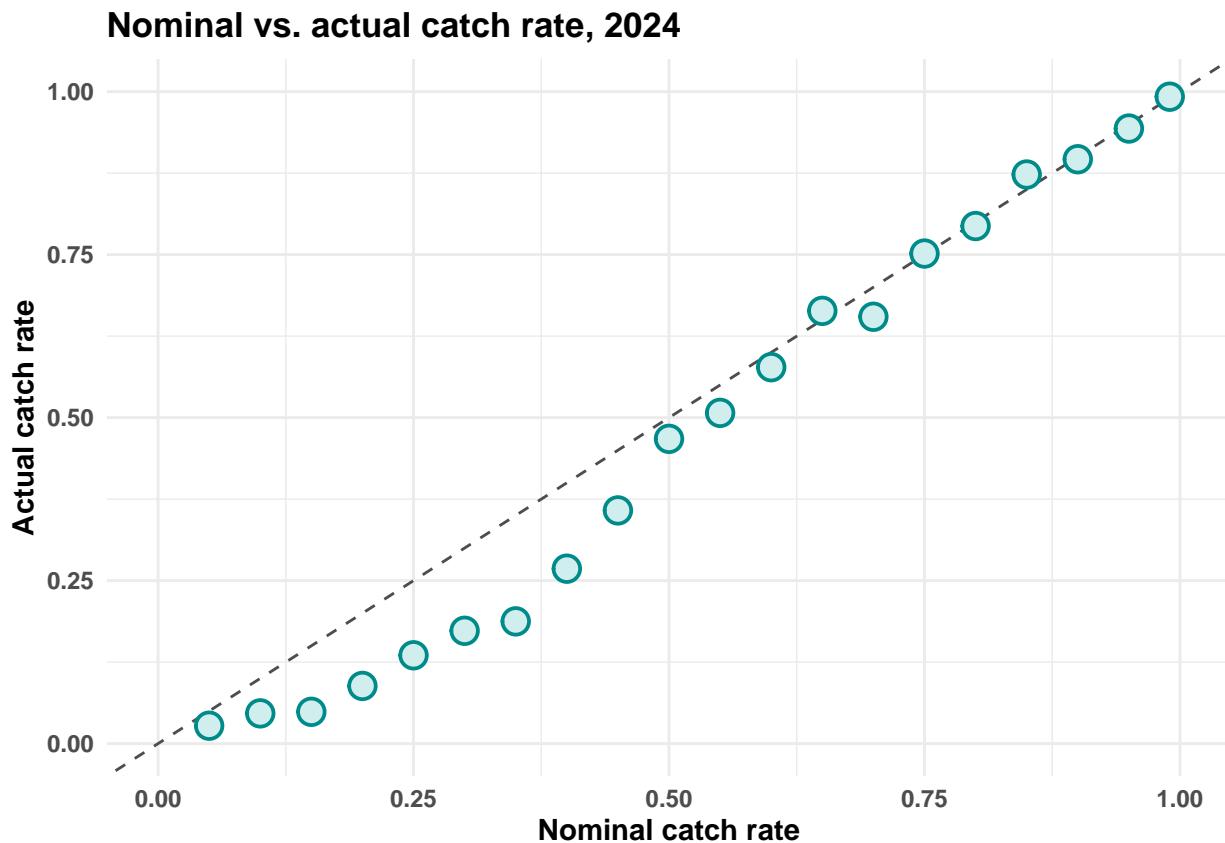
```
ggsave("../figures/jan6.png", plot = jan6, height = 4, width = 6)
```

January 7th

```
calibration_table_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)
```

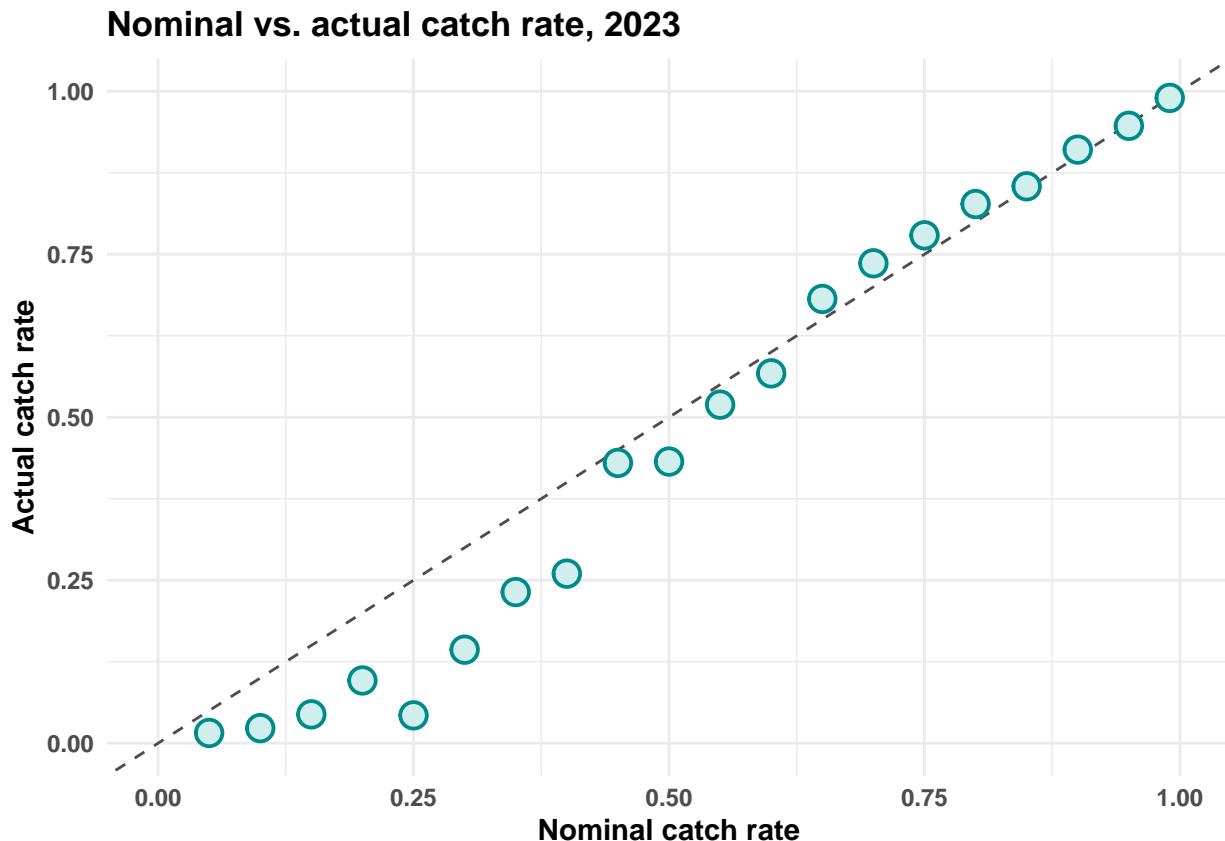
```

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, name_display_first_last) %>%
  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 = name_display_first_last, opp,
         oaa_lower, oaa_estimate, oaa_upper, oaa_true = oaa)

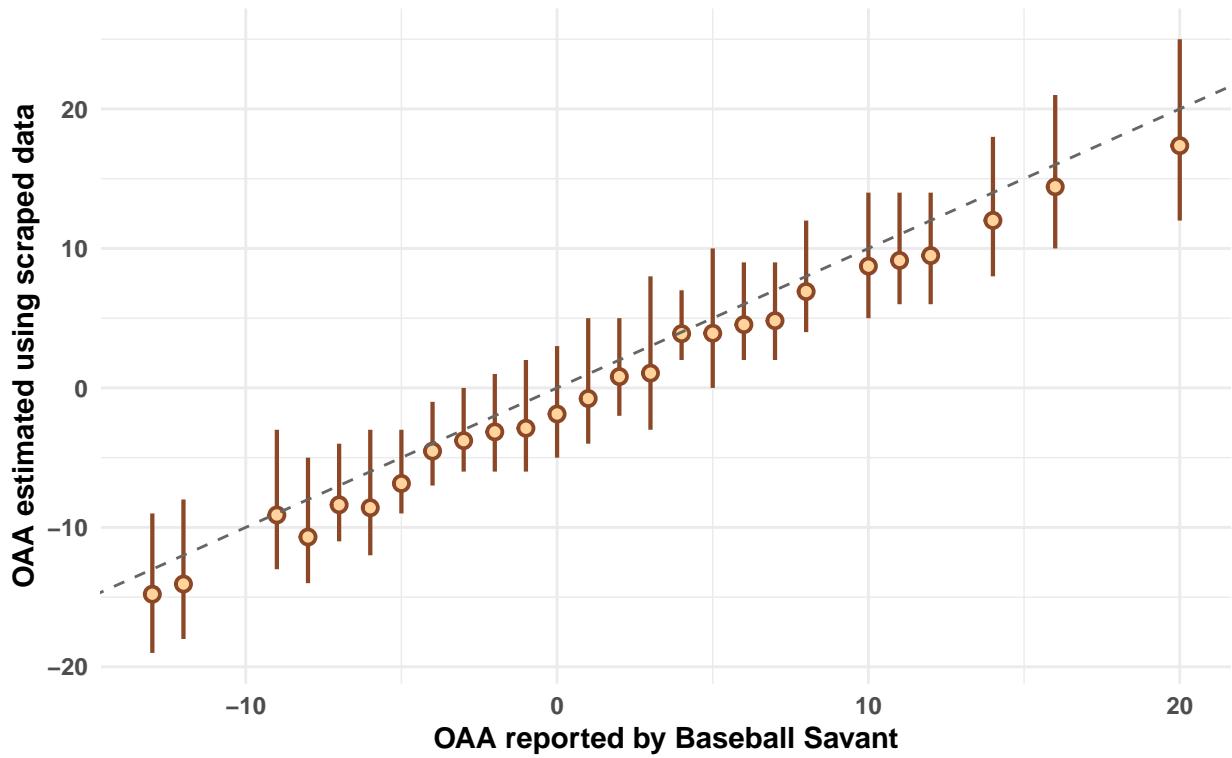
oaa_rounding_check %>%
  summarize(any(oaa_true < oaa_lower | oaa_true > oaa_upper))

## # 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

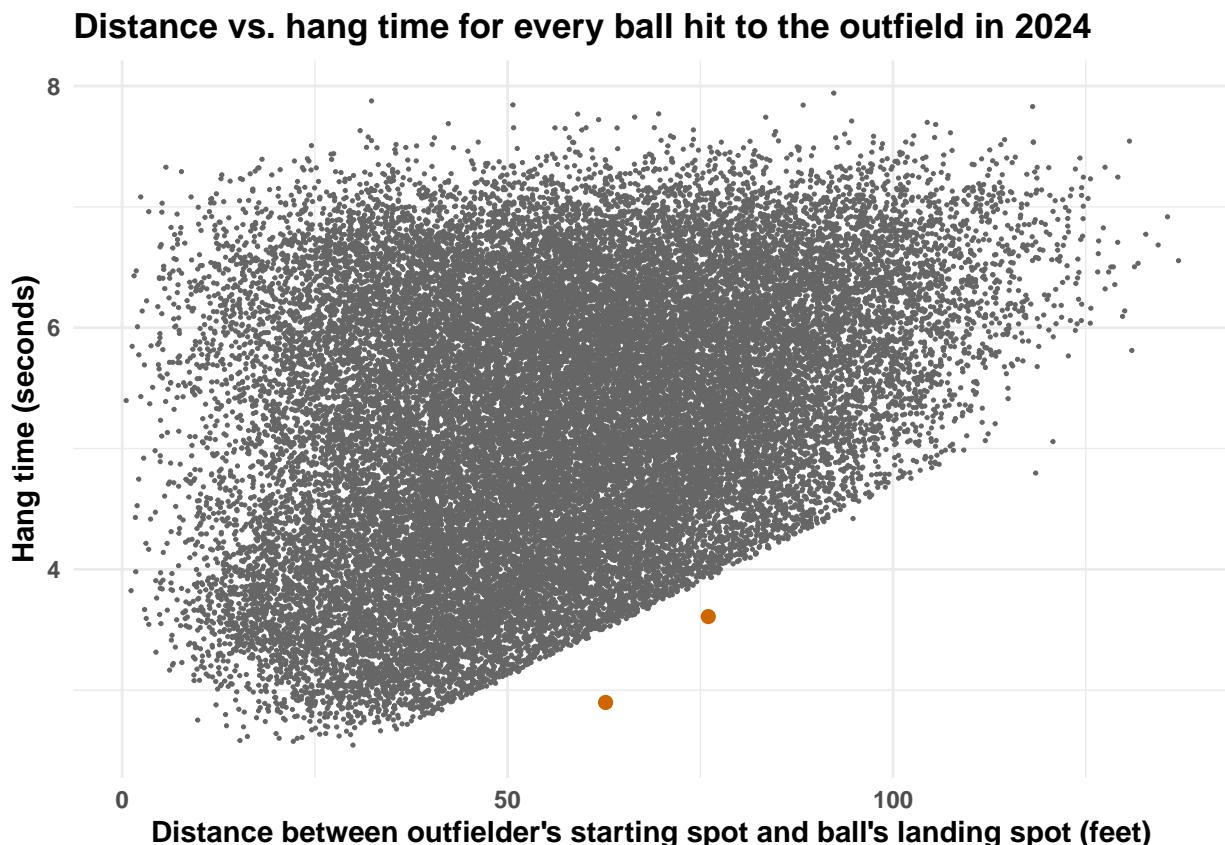


```
ggsave("../figures/jan8.png", plot = jan8, height = 4, width = 6)
```

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")
```

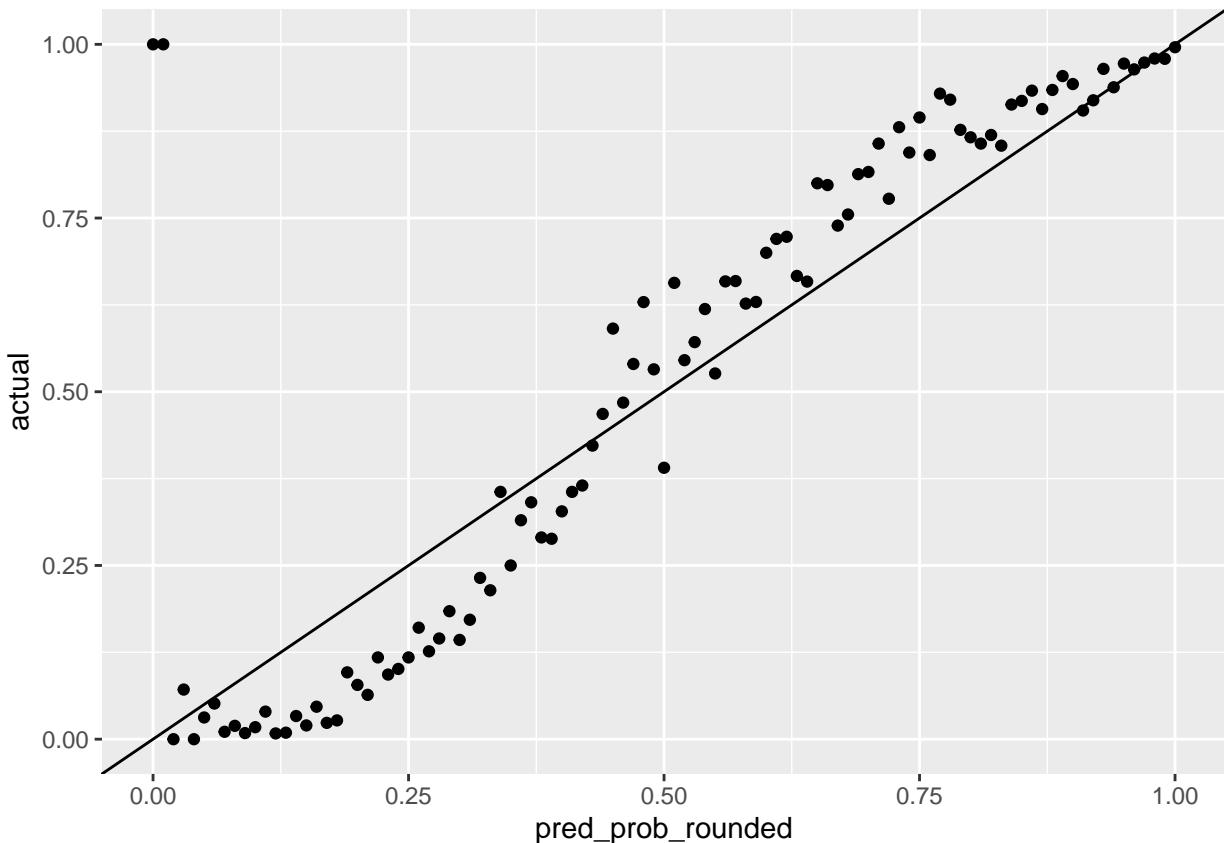
```
jan9
```



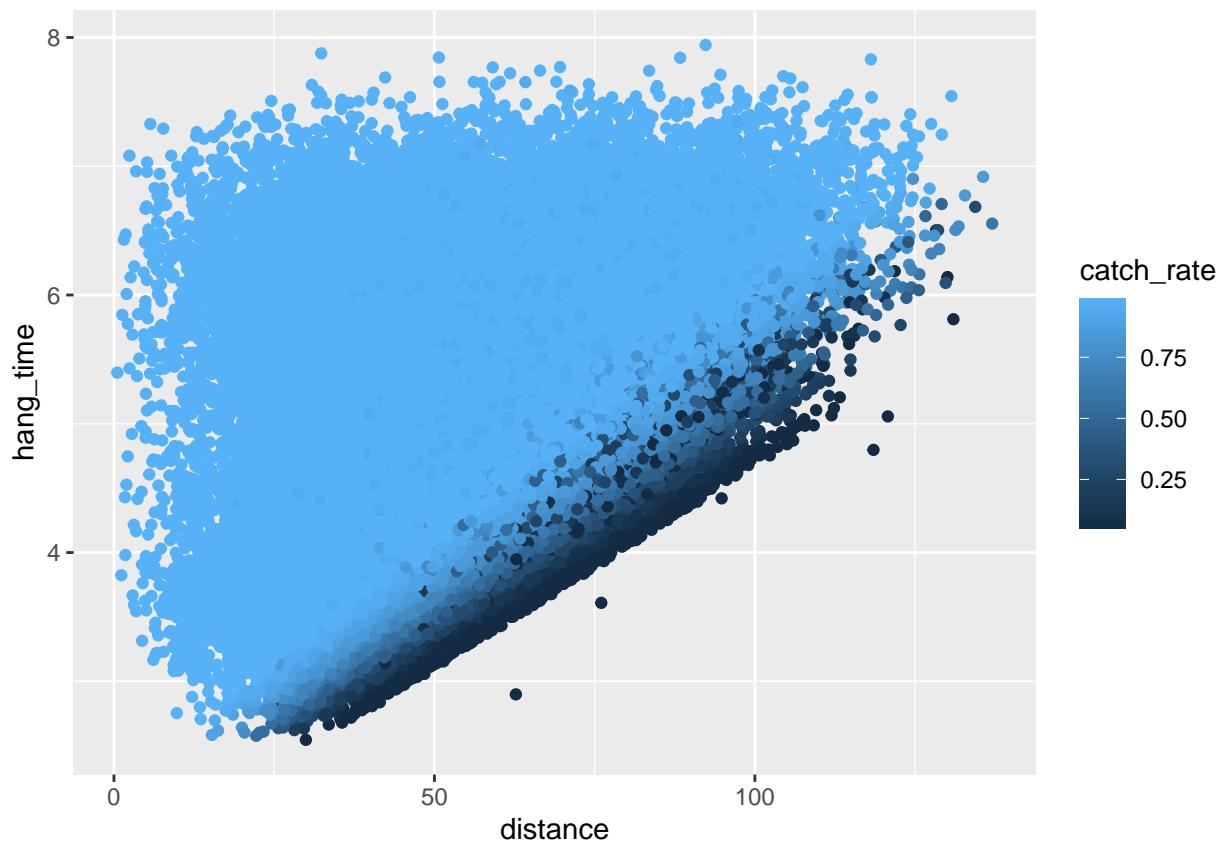
```
ggsave("../figures/jan9.png", plot = jan9, height = 4, width = 6)
```

Later

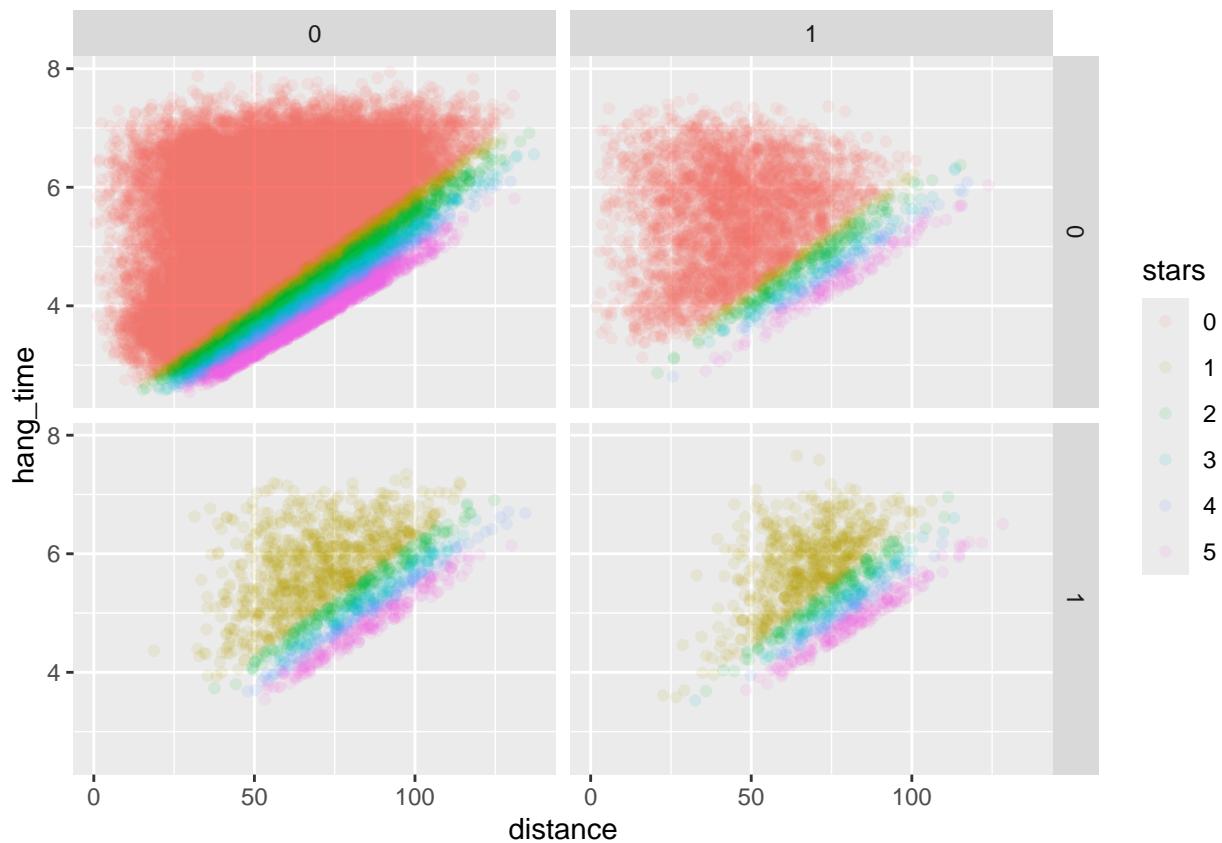
```
mod <- glm(as.factor(out) ~ hang_time + distance + as.factor(wall) + as.factor(back) + as.factor(pos) +  
  
tibble(pred_prob = predict(mod, type = "response")) %>%  
  mutate(pred_prob_rounded = round(pred_prob, digits = 2)) %>%  
  bind_cols(of_catch_prob_2024) %>%  
  group_by(pred_prob_rounded) %>%  
  summarize(actual = mean(out)) %>%  
  ggplot(aes(x = pred_prob_rounded, y = actual)) + geom_point() + geom_abline()
```



```
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))
```

