Lecture 15 Notes

2024-03-19

Loading the data

```
require(tidyverse)
```

```
## Loading required package: tidyverse
```

```
## Warning: package 'tidyverse' was built under R version 4.3.2
```

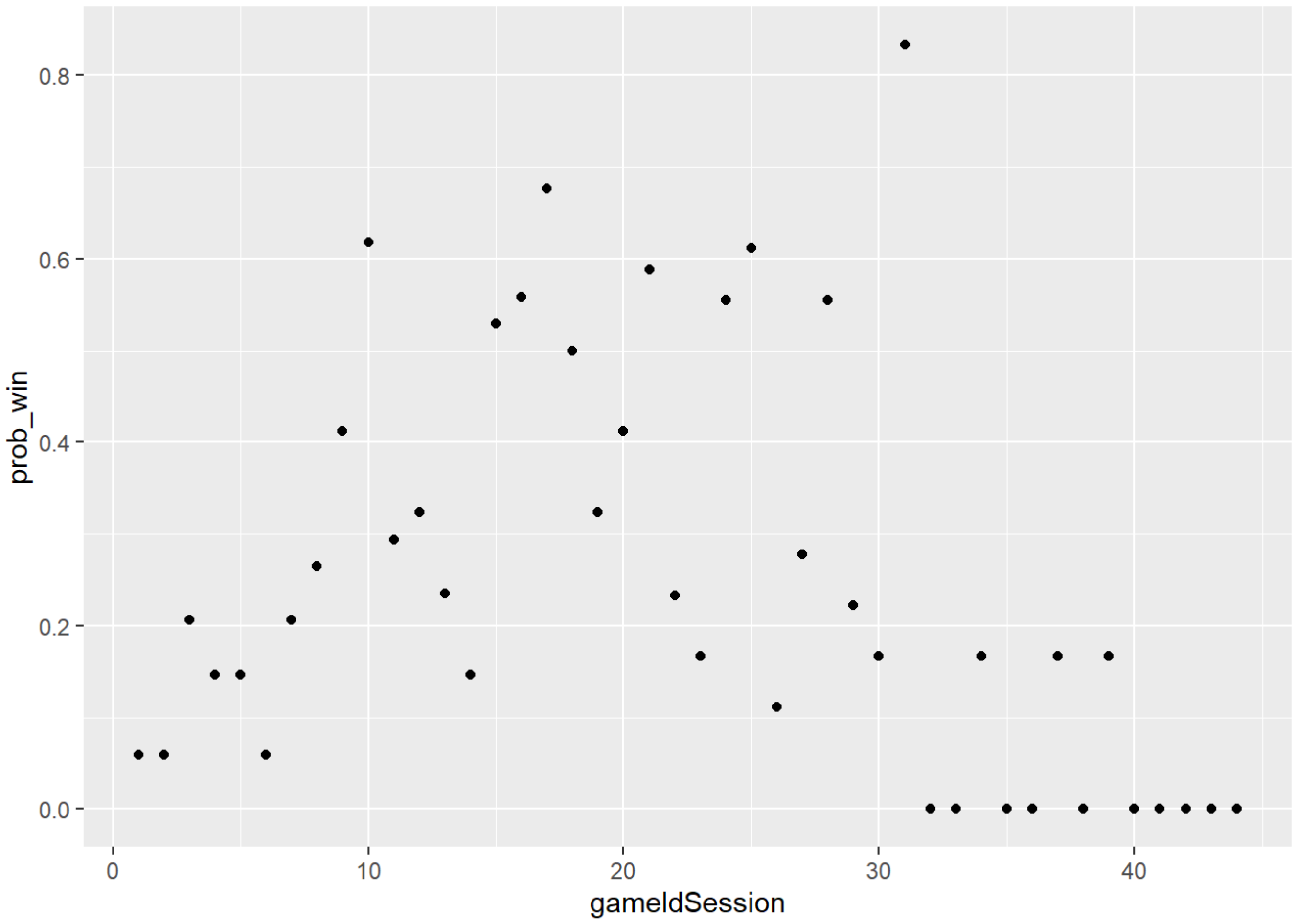
```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ dplyr     1.1.2     ✔ readr     2.1.4
## ✔ forcats   1.0.0     ✔ stringr   1.5.0
## ✔ ggplot2   3.4.4     ✔ tibble    3.2.1
## ✔ lubridate 1.9.2     ✔ tidyr     1.3.0
## ✔ purrr     1.0.1
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts t
o become errors
```

```
fn <- read_rds('https://github.com/jbisbee1/DS1000_S2024/raw/main/data/fn_cleaned_final.
rds')
```

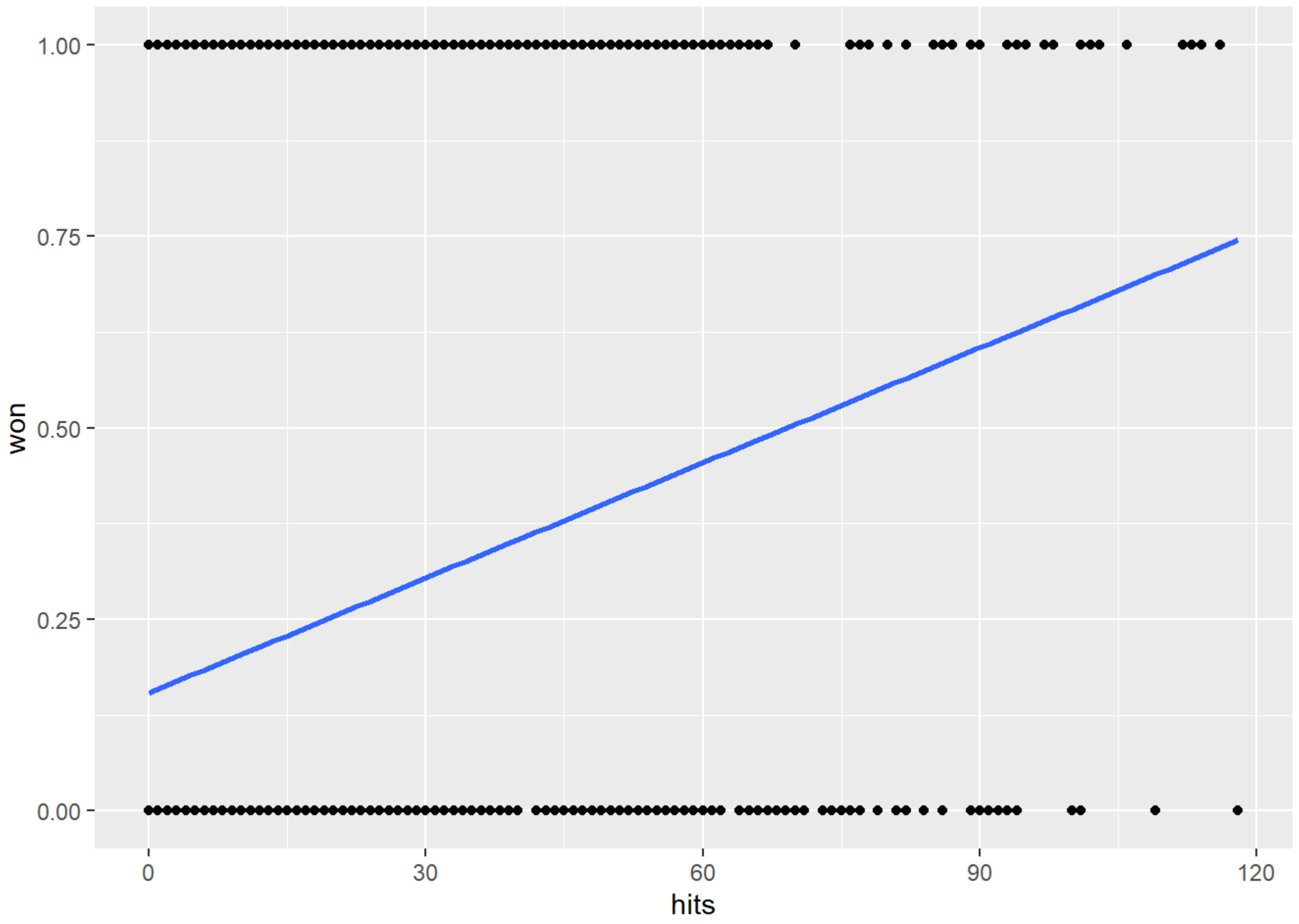
Multivariate visualization

```
fn %>%
  group_by(gameIdSession) %>%
  summarise(prob_win = mean(won)) %>%
  ggplot(aes(x = gameIdSession,
             y = prob_win)) +
  geom_point()
```

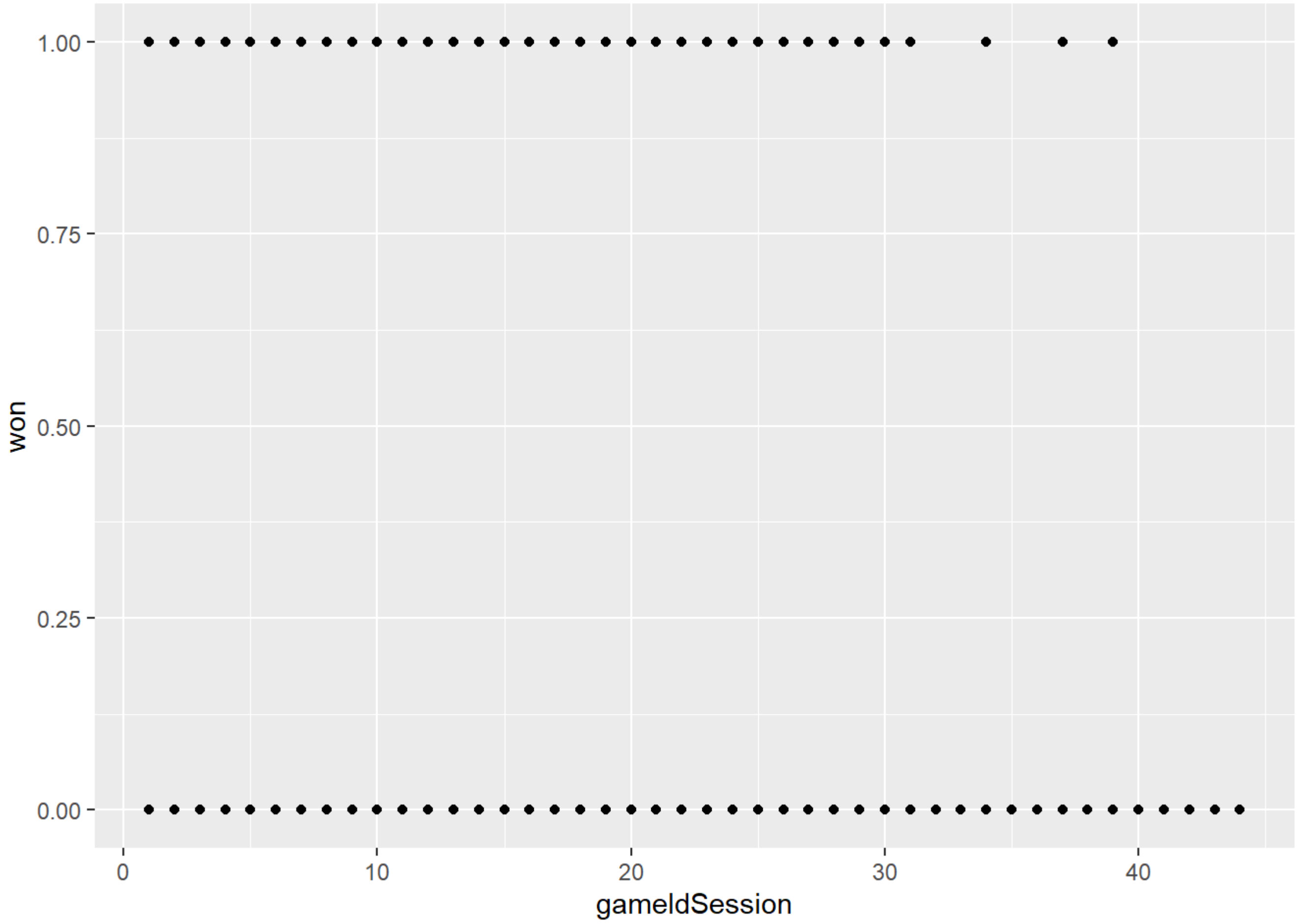


```
fn %>%
  ggplot(aes(x = hits,
             y = won)) +
  geom_point() +
  geom_smooth(method = 'lm',se = F)
```

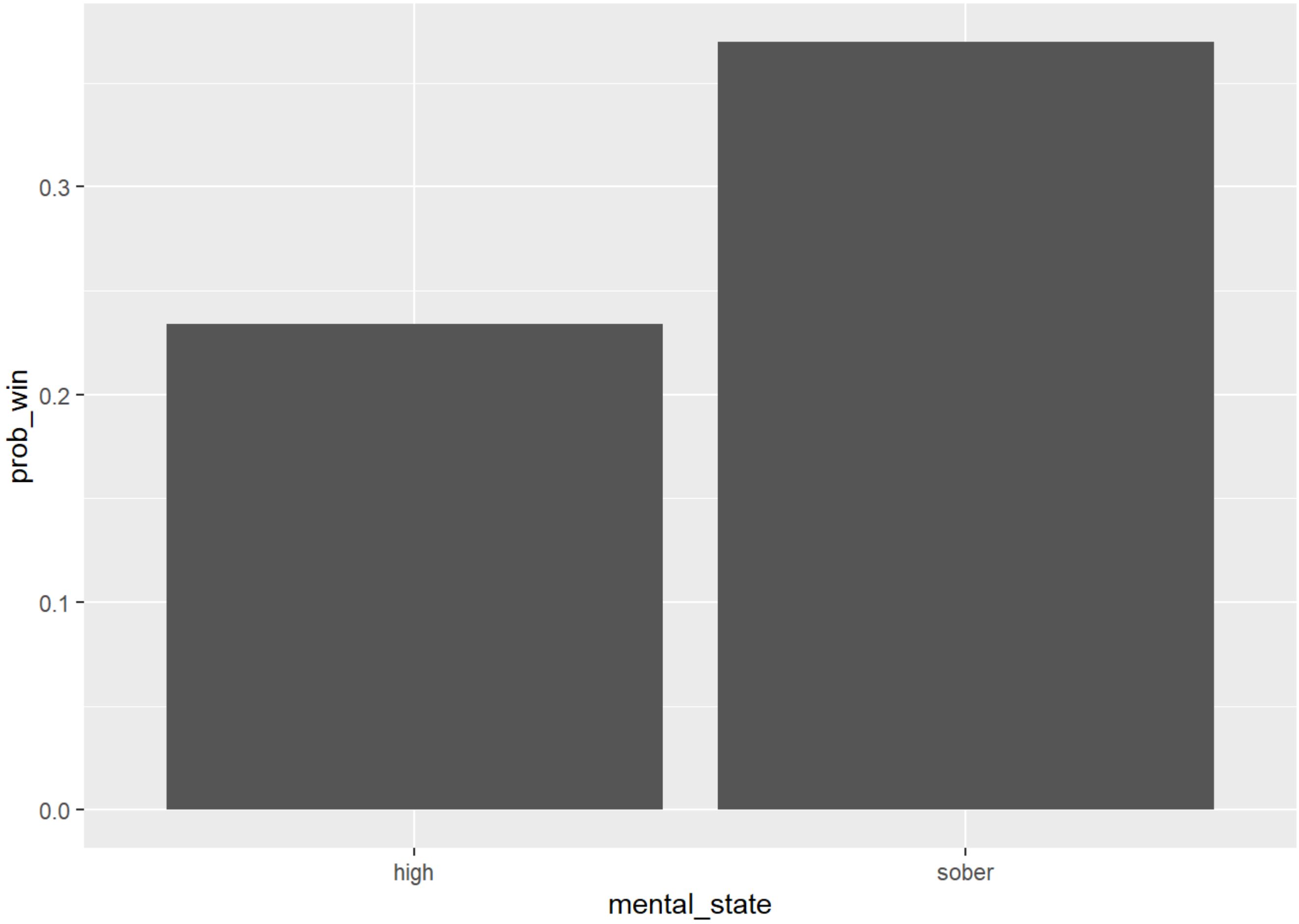
```
## `geom_smooth()` using formula = 'y ~ x'
```



```
fn %>%
  ggplot(aes(x = gameIdSession,
             y = won)) +
  geom_point()
```



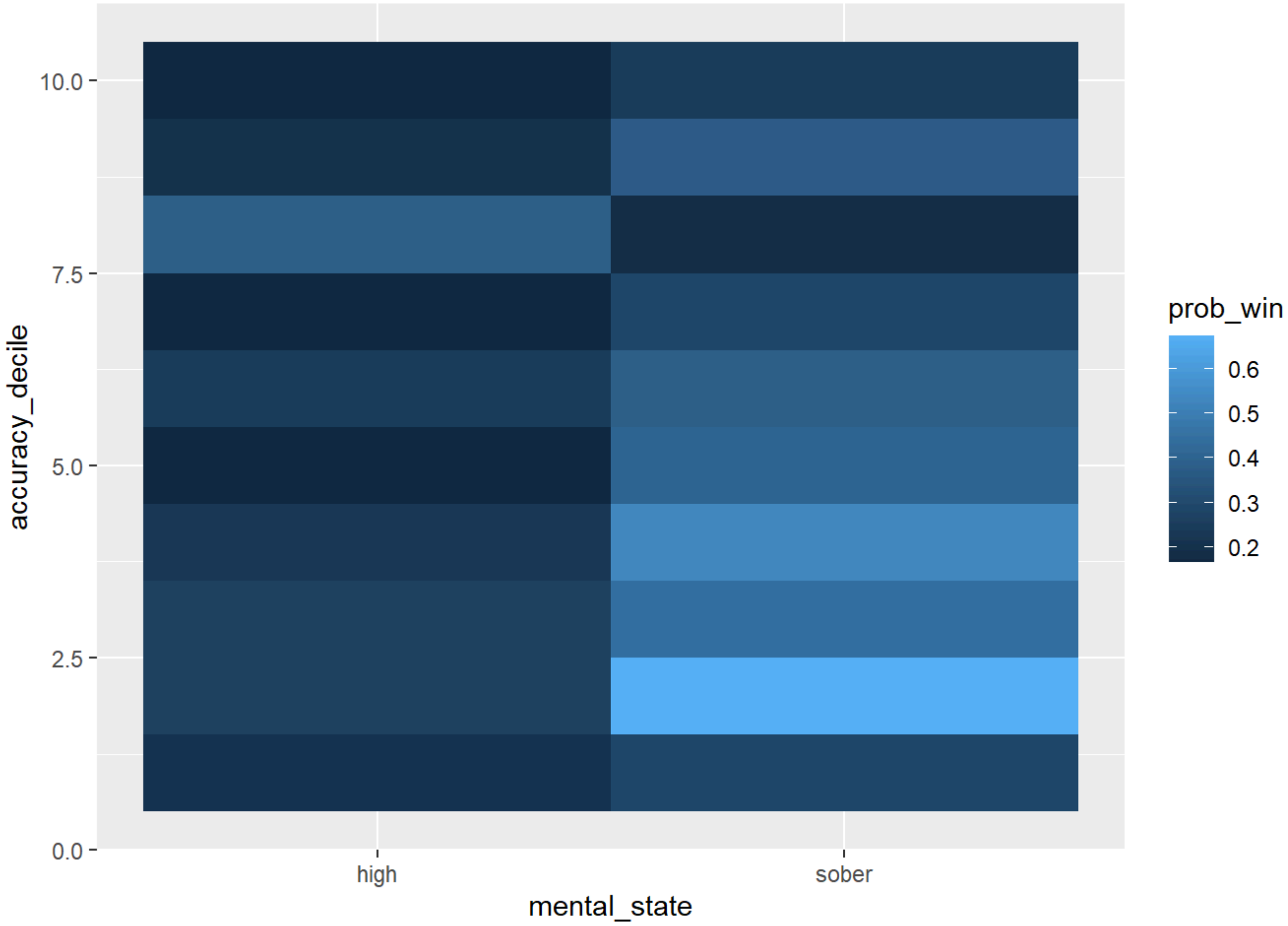
```
fn %>%
  group_by(mental_state) %>%
  summarise(prob_win = mean(won)) %>%
  ggplot(aes(x = mental_state,
             y = prob_win)) +
  geom_bar(stat = 'identity')
```



Heatmaps

```
# ntile()
fn %>%
  mutate(accuracy_decile = ntile(accuracy,n = 10)) %>%
  # select(accuracy,accuracy_decile)
  group_by(accuracy_decile,mental_state) %>%
  summarise(prob_win = mean(won)) %>%
  ggplot(aes(x = mental_state,
             y = accuracy_decile,
             fill = prob_win)) +
  geom_tile()
```

```
## `summarise()` has grouped output by 'accuracy_decile'. You can override using
## the `.groups` argument.
```



Predicting wins / losses

```
fn %>%
  mutate(accuracy_decile = ntile(accuracy,n = 10)) %>%
  group_by(accuracy_decile,mental_state) %>%
  mutate(prob_win = mean(won)) %>%
  select(accuracy_decile,mental_state,won,prob_win) %>%
  mutate(pred_win = ifelse(prob_win > .5,1,0)) %>%
  group_by(won,pred_win) %>%
  summarise(nGames = n()) %>%
  group_by(won) %>%
  mutate(totGames = sum(nGames)) %>%
  ungroup() %>%
  mutate(proportion = nGames / totGames)
```

```
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## # A tibble: 4 × 5
##     won pred_win nGames totGames proportion
##   <dbl>    <dbl>  <int>    <int>      <dbl>
## 1     0        0    632      666     0.949
## 2     0        1     34      666     0.0511
## 3     1        0    239      291     0.821
## 4     1        1     52      291     0.179
```

```
(632 + 52) / (957)
```

```
## [1] 0.7147335
```

Running a linear regression

```
m <- lm(formula = won ~ damage_to_players + mental_state + hits,data = fn)

summary(m)
```

```
##
## Call:
## lm(formula = won ~ damage_to_players + mental_state + hits, data = fn)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8004 -0.2891 -0.1557  0.3694  1.0229
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -1.779e-02  2.883e-02  -0.617    0.537
## damage_to_players  5.087e-04  5.485e-05   9.276  < 2e-16 ***
## mental_statesober  1.094e-01  2.749e-02   3.979 7.43e-05 ***
## hits              -1.023e-03  8.837e-04  -1.157    0.247
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4229 on 953 degrees of freedom
## Multiple R-squared:  0.1584, Adjusted R-squared:  0.1557
## F-statistic: 59.78 on 3 and 953 DF,  p-value: < 2.2e-16
```

```
fn %>%
  mutate(prob_win = predict(m)) %>%
  select(won,prob_win) %>%
  mutate(pred_win = ifelse(prob_win > .45,1,0)) %>%
  group_by(won,pred_win) %>%
  summarise(nGames = n()) %>%
  group_by(won) %>%
  mutate(totGames = sum(nGames)) %>%
  ungroup() %>%
  mutate(proportion = nGames / totGames) %>%
  mutate(overall_accuracy = (sum((won == pred_win)*nGames) / sum(nGames)))
```

```
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
```

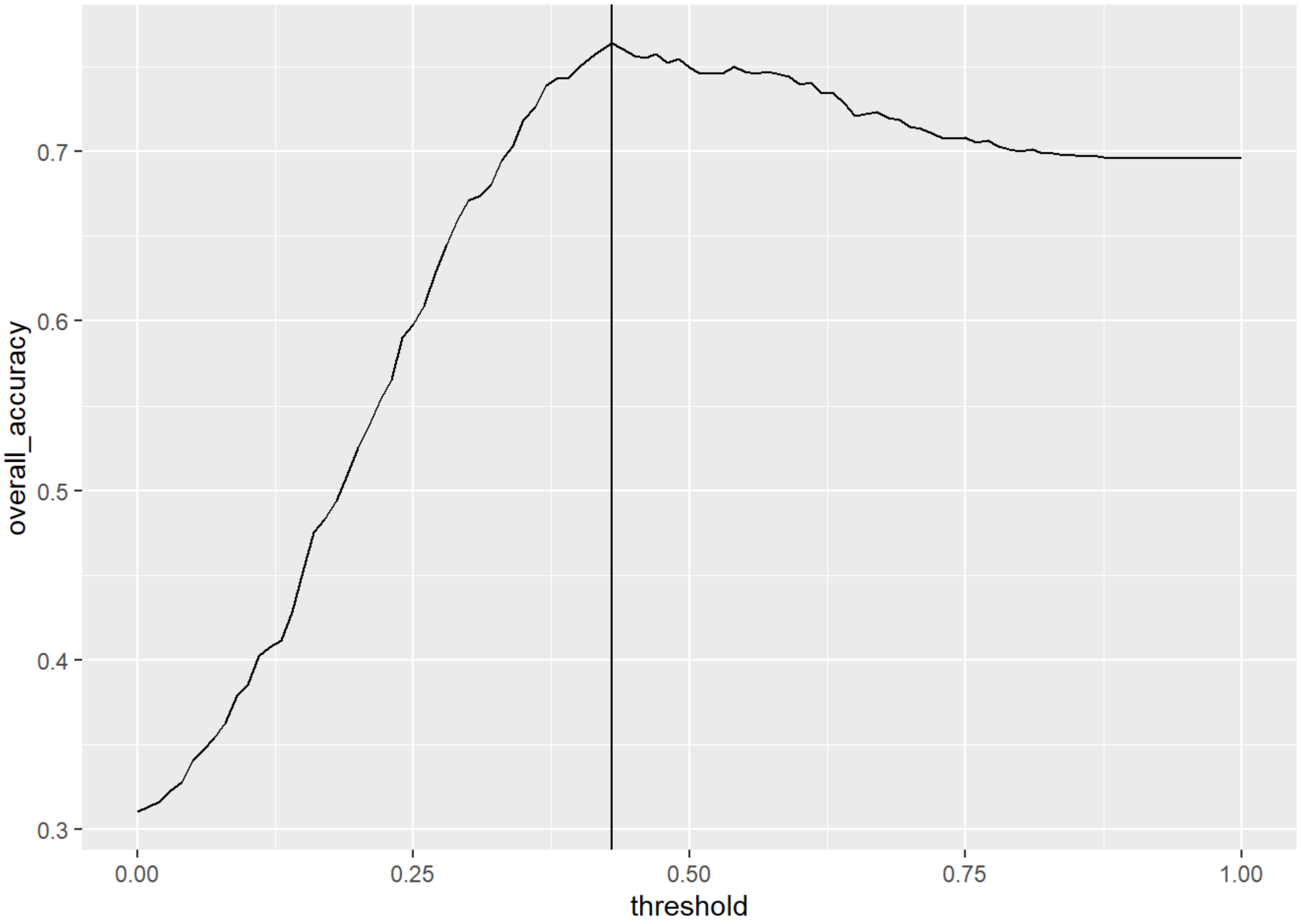
```
## # A tibble: 4 × 6
##     won pred_win nGames totGames proportion overall_accuracy
##   <dbl>    <dbl>  <int>    <int>      <dbl>            <dbl>
## 1     0        0    598      666      0.898            0.757
## 2     0        1     68      666      0.102            0.757
## 3     1        0    165      291      0.567            0.757
## 4     1        1    126      291      0.433            0.757
```

Sensitivity vs Specificity

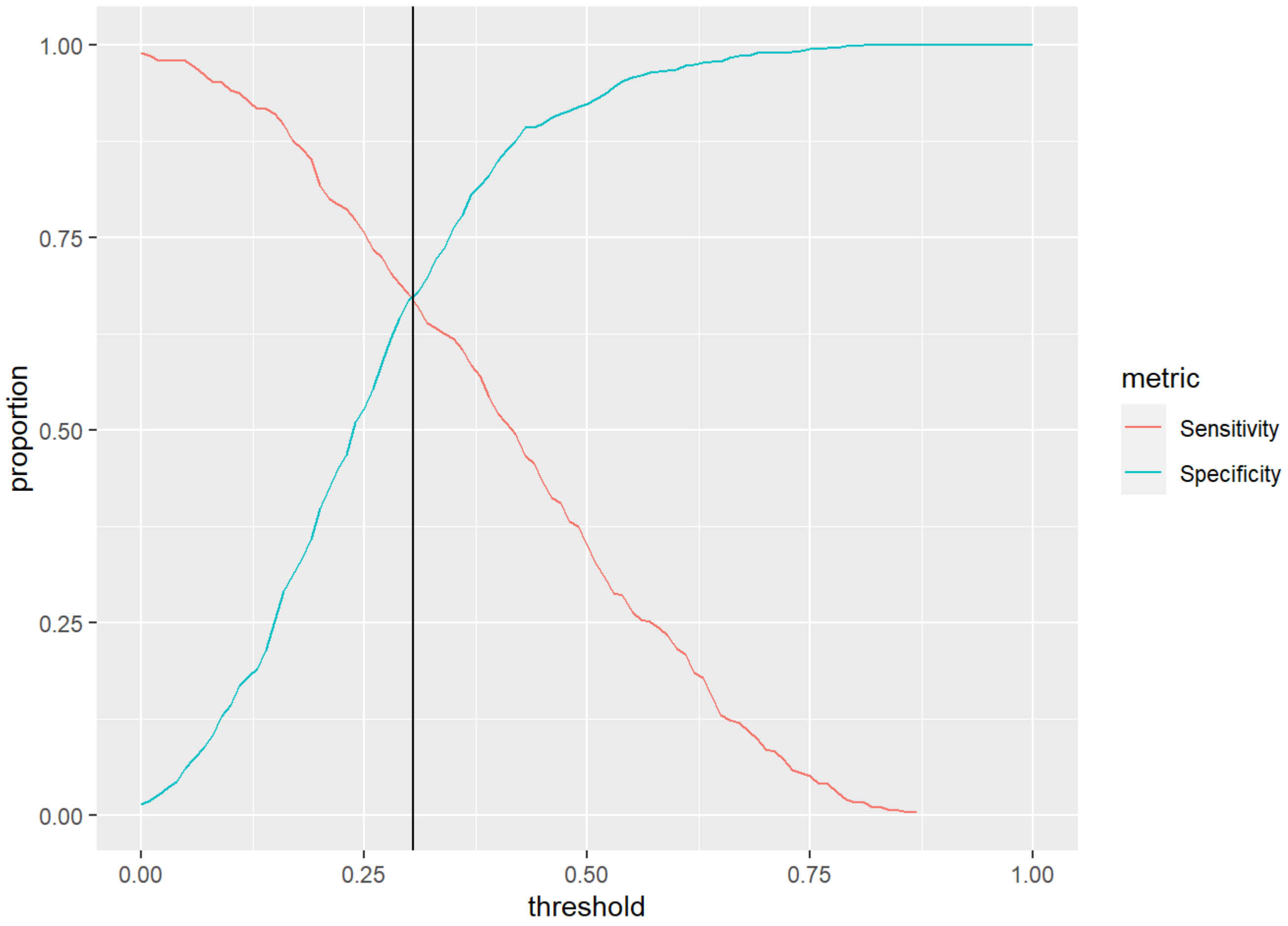
```
threshRes <- NULL
for(thresh in seq(0,1,by = .01)) {
  threshRes <- threshRes %>%
    bind_rows(fn %>%
  mutate(prob_win = predict(m)) %>%
  select(won,prob_win) %>%
  mutate(pred_win = ifelse(prob_win > thresh,1,0)) %>%
  group_by(won,pred_win) %>%
  summarise(nGames = n()) %>%
  group_by(won) %>%
  mutate(totGames = sum(nGames)) %>%
  ungroup() %>%
  mutate(proportion = nGames / totGames) %>%
  mutate(overall_accuracy = (sum((won == pred_win)*nGames) / sum(nGames))) %>%
    mutate(threshold = thresh))
}
```

```
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
## `summarise()` has grouped output by 'won'. You can override using the `.groups`
## argument.
```

```
threshRes %>%
  ggplot(aes(x = threshold,
             y = overall_accuracy)) +
  geom_line() +
  geom_vline(xintercept = .43)
```



```
# Sensitivity vs specificity
threshRes %>%
  mutate(metric = ifelse(won == 0 & pred_win == 0,'Specificity',
                         ifelse(won == 1 & pred_win == 1,'Sensitivity',NA))) %>%
  drop_na(metric) %>%
  ggplot(aes(x = threshold,
             y = proportion,
             color = metric)) +
  geom_line() +
  geom_vline(xintercept = .305)
```


1.00
0.75
0.50
0.25
0.00
proportion
0.00
0.25
0.50
0.75
1.00
threshold
metric
Sensitivity
Specificity