Support Vector Machines

James Caldwell

2024-04-21

Required R packages and Directories

```r
dir_data= 'https://mdporter.github.io/SYS6018/data/' # data directory
library(knitr)      # for nicer printing of tables with kable
library(e1071)      # for SVM
library(tidymodels) # for modeling and evaluation functions
library(tidyverse)  # functions for data manipulation
```

COMPAS Recidivism Prediction

A recidivism risk model called COMPAS was the topic of a ProPublica article (https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing/) on ML bias. Because the data and notebooks used for article was released on github (https://github.com/propublica/compas-analysis), we can also evaluate the prediction bias (i.e., calibration).

This code will read in the *violent crime* risk score and apply the filtering used in the analysis (https://github.com/propublica/compas-analysis/blob/master/Compas%20Analysis.ipynb).

```
library(tidyverse)
df = read_csv("https://raw.githubusercontent.com/propublica/compas-analysis/master/compas-scores-two-years-violent.csv")

risk = df %>%
  filter(days_b_screening_arrest <= 30) %>%
  filter(days_b_screening_arrest >= -30) %>%
  filter(is_recid != -1) %>%
  filter(c_charge_degree != "O") %>%
  filter(v_score_text != 'N/A') %>%
  transmute(
    age, age_cat,
    charge = ifelse(c_charge_degree == "F", "Felony", "Misdemeanor"),
    race,
    sex,
    priors_count = priors_count...15,
    score = v_decile_score,              # the risk score {1,2,...,10}
    outcome = two_year_recid...53        # outcome {1 = two year recidivate}
  )
```

The `risk` data frame has the relevant information for completing the problems.

Problem 1: COMPAS risk score

a. Risk Score and Probability (table)

Assess the predictive bias in the COMPAS risk scores by evaluating the probability of recidivism, e.g. estimate $\Pr(Y = 1 \mid \text{Score} = x)$. Use any reasonable techniques (including Bayesian) to estimate the probability of recidivism for each risk score.

Specifically, create a table (e.g., data frame) that provides the following information:

- The COMPASS risk score.
- The point estimate of the probability of recidivism for each risk score.
- 95% confidence or credible intervals for the probability (e.g., Using normal theory, bootstrap, or Bayesian techniques).

Indicate the choices you made in estimation (e.g., state the prior if you used Bayesian methods).

```
print(head(risk))
```

```
## # A tibble: 6 × 8
##     age age_cat         charge      race        sex    priors_count score outcome
##   <dbl> <chr>           <chr>       <chr>       <chr>         <dbl> <dbl>   <dbl>
## 1    69 Greater than 45 Felony      Other       Male              0     1       0
## 2    34 25 - 45         Felony      African-Am… Male              0     1       1
## 3    44 25 - 45         Misdemeanor Other       Male              0     1       0
## 4    43 25 - 45         Felony      Other       Male              3     3       0
## 5    39 25 - 45         Misdemeanor Caucasian   Fema…             0     1       0
## 6    27 25 - 45         Felony      Caucasian   Male              0     4       0
```

```
summary(risk)
```

```
##       age          age_cat             charge              race          
##  Min.   :18.00   Length:4020        Length:4020        Length:4020       
##  1st Qu.:26.00   Class :character   Class :character   Class :character  
##  Median :33.00   Mode  :character   Mode  :character   Mode  :character  
##  Mean   :35.74                                                           
##  3rd Qu.:44.00                                                           
##  Max.   :83.00                                                           
##      sex             priors_count        score           outcome      
##  Length:4020        Min.   : 0.000   Min.   : 1.000   Min.   :0.0000  
##  Class :character   1st Qu.: 0.000   1st Qu.: 1.000   1st Qu.:0.0000  
##  Mode  :character   Median : 1.000   Median : 3.000   Median :0.0000  
##                     Mean   : 2.446   Mean   : 3.265   Mean   :0.1622  
##                     3rd Qu.: 3.000   3rd Qu.: 5.000   3rd Qu.:0.0000  
##                     Max.   :38.000   Max.   :10.000   Max.   :1.0000  
```

```
# Group data by COMPASS risk score and calculate point estimate of probability of recidivism
recidivism_prob <- risk %>%
  group_by(score) %>%
  summarise(
    recidivism_rate = mean(outcome == 1),
    n = n()
  )

bootstrap_ci <- function(data, score_col, outcome_col, n_bootstrap = 50, confidence_level = 0.95) {
  bootstrapped_props <- data %>%
    group_by({{ score_col }}) %>%
    summarise(
      prob_lower_ci = mean({{ outcome_col }} == 1) - 1.96 * sqrt(mean({{ outcome_col }} == 1) * (1 - mean({{ outcome_col }}
== 1)) / n()),
      prob_upper_ci = mean({{ outcome_col }} == 1) + 1.96 * sqrt(mean({{ outcome_col }} == 1) * (1 - mean({{ outcome_col }}
== 1)) / n())
    )
  return(bootstrapped_props)
}

# Calculate bootstrap confidence intervals for probability of recidivism
bootstrap_ci <- bootstrap_ci(risk, score, outcome)

# Combine point estimates and confidence intervals into a single dataframe
recidivism_table <- inner_join(recidivism_prob, bootstrap_ci, by = "score")

print(recidivism_table, n = Inf)
```

```
## # A tibble: 10 × 5
##    score recidivism_rate     n prob_lower_ci prob_upper_ci
##    <dbl>           <dbl> <int>         <dbl>         <dbl>
##  1     1          0.0619  1340        0.0490        0.0748
##  2     2          0.0932   622        0.0704        0.116
##  3     3          0.171    543        0.140         0.203
##  4     4          0.176    408        0.139         0.213
##  5     5          0.185    325        0.142         0.227
##  6     6          0.283    300        0.232         0.334
##  7     7          0.315    203        0.251         0.379
##  8     8          0.434    122        0.346         0.522
##  9     9          0.564    110        0.471         0.656
## 10    10          0.468     47        0.325         0.611
```

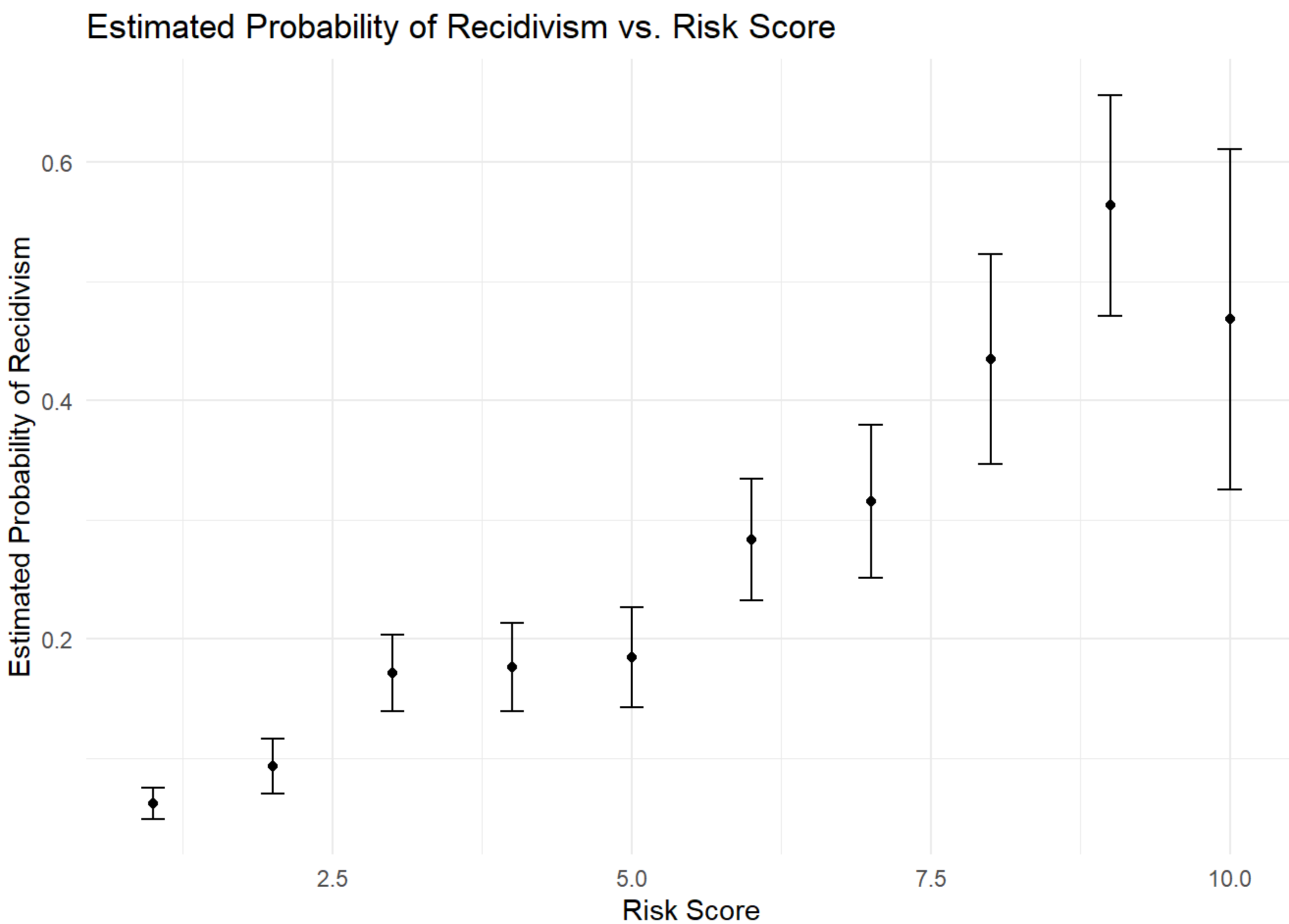
Choices made in estimation: boostrap chosen for CI's, recividism rate is average of each score category group.

b. Risk Score and Probability (plot)

Make a plot of the risk scores and corresponding estimated probability of recidivism.

- Put the risk score on the x-axis and estimate probability of recidivism on y-axis.
- Add the 95% confidence or credible intervals calculated in part a.
- Comment on the patterns you see.

```
ggplot(recidivism_table, aes(x = score, y = recidivism_rate)) +
  geom_point() +  # Add points for the estimated probabilities
  geom_errorbar(aes(ymin = prob_lower_ci, ymax = prob_upper_ci), width = 0.2) +  # Add error bars for confidence intervals
  labs(x = "Risk Score", y = "Estimated Probability of Recidivism") +  # Axis labels
  ggtitle("Estimated Probability of Recidivism vs. Risk Score") +  # Title
  theme_minimal()
```



The general trend shown here make sense: a higher risk score correlates with a higher probability of recidivism. It also makes sense that the CI's grow with the risk score, since it's easier (and more likely) to predict that someone won't be a repeat offender rather than that they will. One thing that doesn't make a lot of sense to me is how my probability drops for risk scores of 10. It is possible that there's an error in my code. If there isn't an error, this could be explained by errors in the scoring system defaulting to a value of 10 for certain people or that individuals with really high risk scores may tend to be more cautious of repeat offenses due to past experiences.

c. Risk Score and Probability (by race)

Repeat the analysis, but this time do so for every race. Produce a set of plots (one per race) and comment on the patterns.

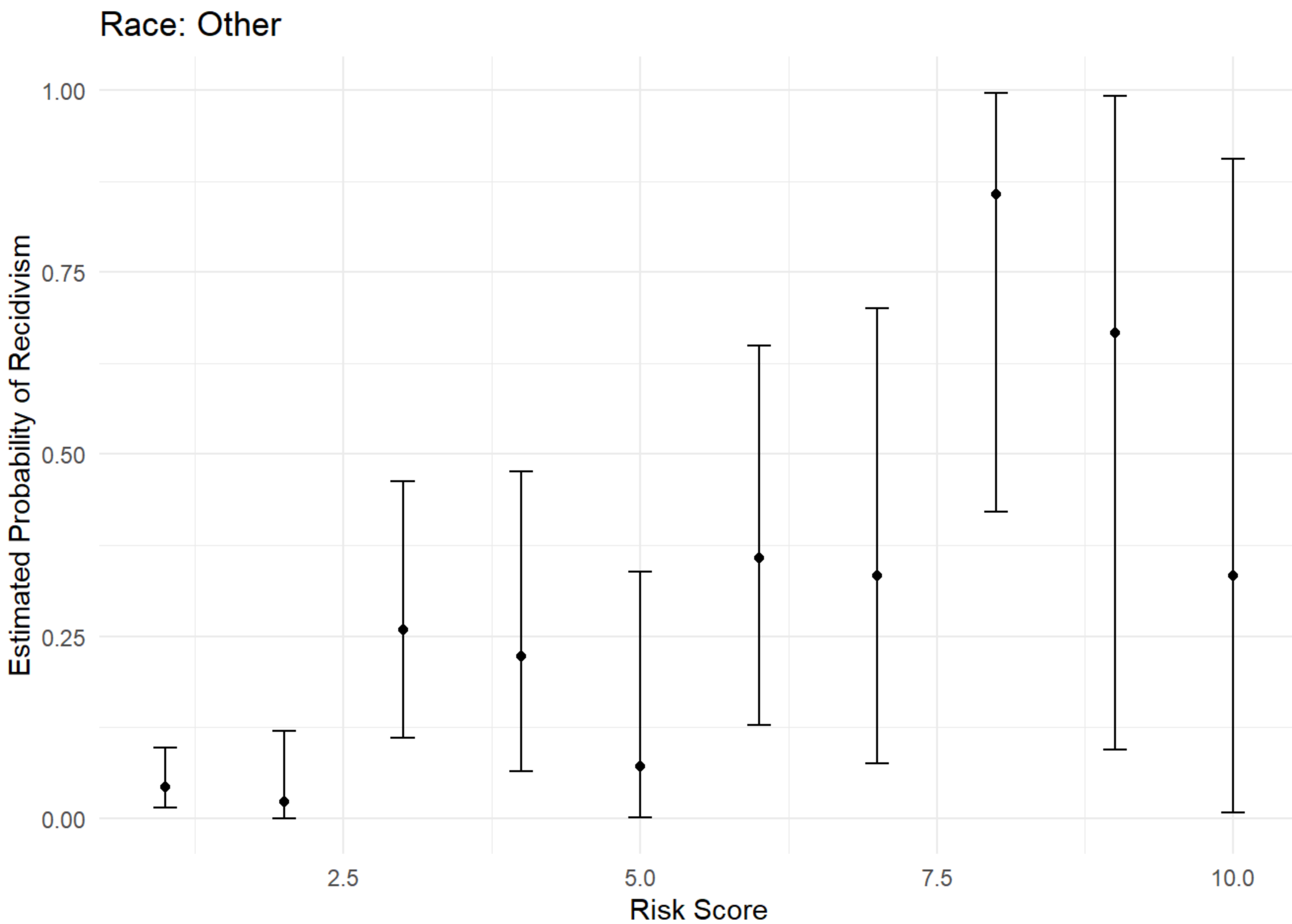
```r
# Create a list to store the plots
plots <- list()

# Get unique race categories
race_categories <- unique(risk$race)

# Loop through each race category
for (race_i in race_categories) {
  # Subset the data for the current race
  # race_data <- filter(risk, race == race)
  race_data <- filter(risk, risk$race == race_i)
  # print(head(race_data))

  # Estimate probability of recidivism and confidence intervals for this race
  recidivism_prob <- race_data %>%
    group_by(score) %>%
    summarise(recidivism_rate = mean(outcome == 1),  # Probability of recidivism
              n = n(),
              lower_ci = binom.test(sum(outcome == 1), n, conf.level = 0.95)$conf.int[1],  # Lower bound of CI
              upper_ci = binom.test(sum(outcome == 1), n, conf.level = 0.95)$conf.int[2])  # Upper bound of CI
  # Same function as part a
  bootstrap_ci <- function(data, score_col, outcome_col, n_bootstrap = 50, confidence_level = 0.95) {
    bootstrapped_props <- data %>%
      group_by({{ score_col }}) %>%
      summarise(
        prob_lower_ci = mean({{ outcome_col }} == 1) - 1.96 * sqrt(mean({{ outcome_col }} == 1) * (1 - mean({{ outcome_col
}} == 1)) / n()),
        prob_upper_ci = mean({{ outcome_col }} == 1) + 1.96 * sqrt(mean({{ outcome_col }} == 1) * (1 - mean({{ outcome_col
}} == 1)) / n())
      )
    return(bootstrapped_props)
  }

  bootstrap_ci_i <- bootstrap_ci(race_data, score, outcome)

  recidivism_table <- inner_join(recidivism_prob, bootstrap_ci_i, by = "score")

  print(race_i)
  # Print the recidivism table for the current race
  print(recidivism_table, n = Inf)

  # Create the plot for the current race
  p <- ggplot(recidivism_table, aes(x = score, y = recidivism_rate)) +
```

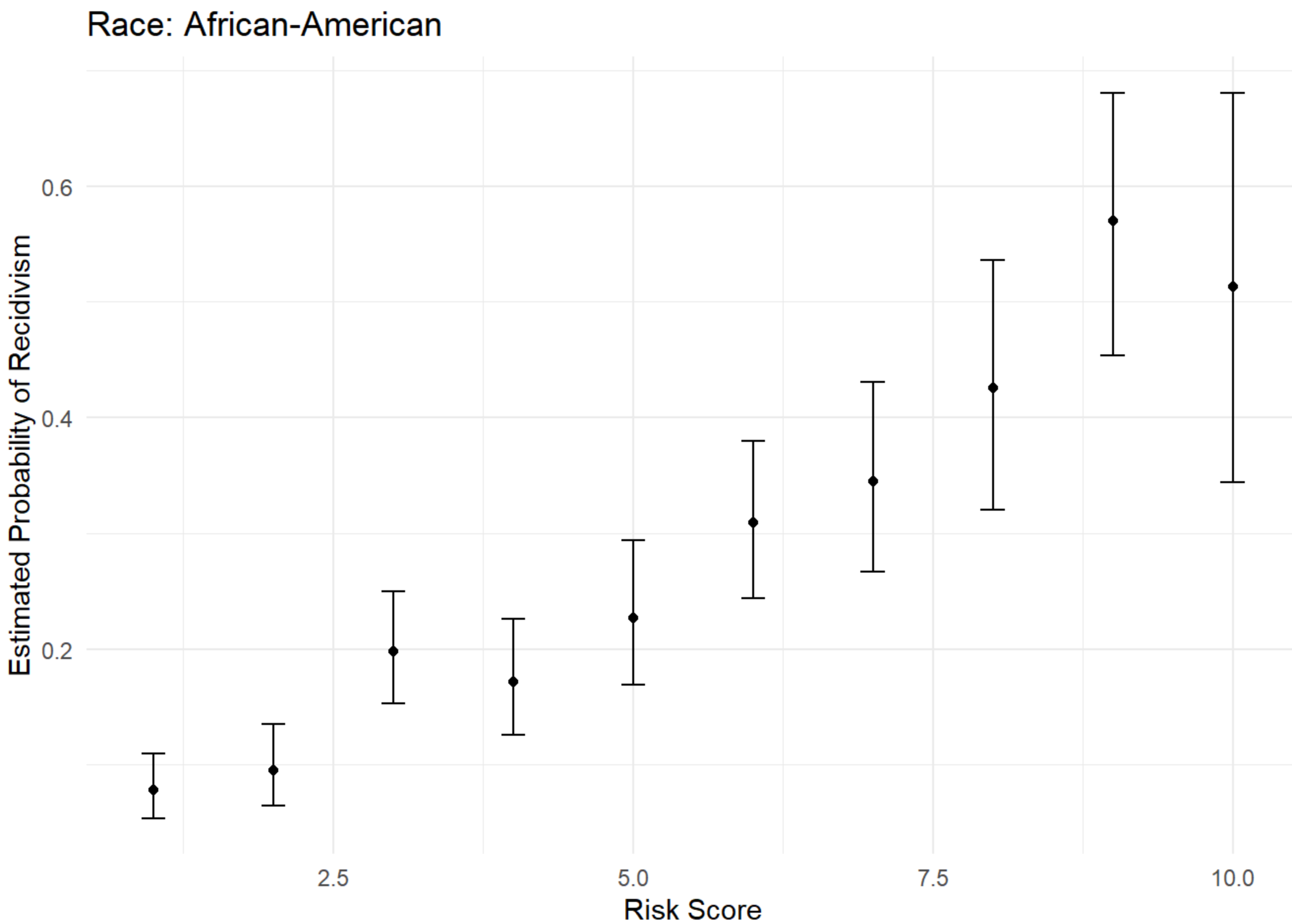
```
    geom_point() +  # Add points for the estimated probabilities
    geom_errorbar(aes(ymin = lower_ci, ymax = upper_ci), width = 0.2) +  # Add error bars for confidence intervals
    labs(x = "Risk Score", y = "Estimated Probability of Recidivism") +  # Axis labels
    ggtitle(paste("Race:", race_i)) +  # Title with current race
    theme_minimal()  # Use a minimal theme for the plot

  # Store the plot in the list
  plots[[race_i]] <- p
}
```

```
## [1] "Other"
## # A tibble: 10 × 7
##    score recidivism_rate     n lower_ci upper_ci prob_lower_ci prob_upper_ci
##    <dbl>           <dbl> <int>    <dbl>    <dbl>         <dbl>         <dbl>
##  1     1          0.0431   116 0.0141     0.0977       0.00614        0.0801
##  2     2          0.0227    44 0.000575   0.120       -0.0213         0.0668
##  3     3          0.259     27 0.111      0.463        0.0940         0.425
##  4     4          0.222     18 0.0641     0.476        0.0302         0.414
##  5     5          0.0714    14 0.00181    0.339       -0.0635         0.206
##  6     6          0.357     14 0.128      0.649        0.106          0.608
##  7     7          0.333      9 0.0749     0.701        0.0253         0.641
##  8     8          0.857      7 0.421      0.996        0.598          1.12
##  9     9          0.667      3 0.0943     0.992        0.133          1.20
## 10    10          0.333      3 0.00840    0.906       -0.200          0.867
## [1] "African-American"
## # A tibble: 10 × 7
##    score recidivism_rate     n lower_ci upper_ci prob_lower_ci prob_upper_ci
##    <dbl>           <dbl> <int>    <dbl>    <dbl>         <dbl>         <dbl>
##  1     1          0.0785   395   0.0539    0.110         0.0520         0.105
##  2     2          0.0952   294   0.0642    0.135         0.0617         0.129
##  3     3          0.198    278   0.153     0.250         0.151          0.245
##  4     4          0.172    233   0.126     0.226         0.123          0.220
##  5     5          0.227    185   0.169     0.294         0.167          0.287
##  6     6          0.309    191   0.244     0.380         0.243          0.374
##  7     7          0.345    139   0.267     0.431         0.266          0.424
##  8     8          0.425     87   0.320     0.536         0.321          0.529
##  9     9          0.570     79   0.453     0.681         0.460          0.679
## 10    10          0.514     37   0.344     0.681         0.352          0.675
## [1] "Caucasian"
## # A tibble: 10 × 7
##    score recidivism_rate     n lower_ci upper_ci prob_lower_ci prob_upper_ci
##    <dbl>           <dbl> <int>    <dbl>    <dbl>         <dbl>         <dbl>
##  1     1          0.0624   657   0.0452   0.0837         0.0439        0.0809
##  2     2          0.0814   221   0.0490   0.126          0.0454        0.118
##  3     3          0.135    193   0.0899   0.191          0.0865        0.183
##  4     4          0.198    126   0.133    0.279          0.129         0.268
##  5     5          0.138     94   0.0757   0.225          0.0685        0.208
##  6     6          0.25      72   0.155    0.366          0.150         0.350
##  7     7          0.283     46   0.160    0.435          0.152         0.413
##  8     8          0.3       20   0.119    0.543          0.0992        0.501
##  9     9          0.5       24   0.291    0.709          0.300         0.700
## 10    10          0.333      6   0.0433   0.777         -0.0439        0.711
## [1] "Hispanic"
```

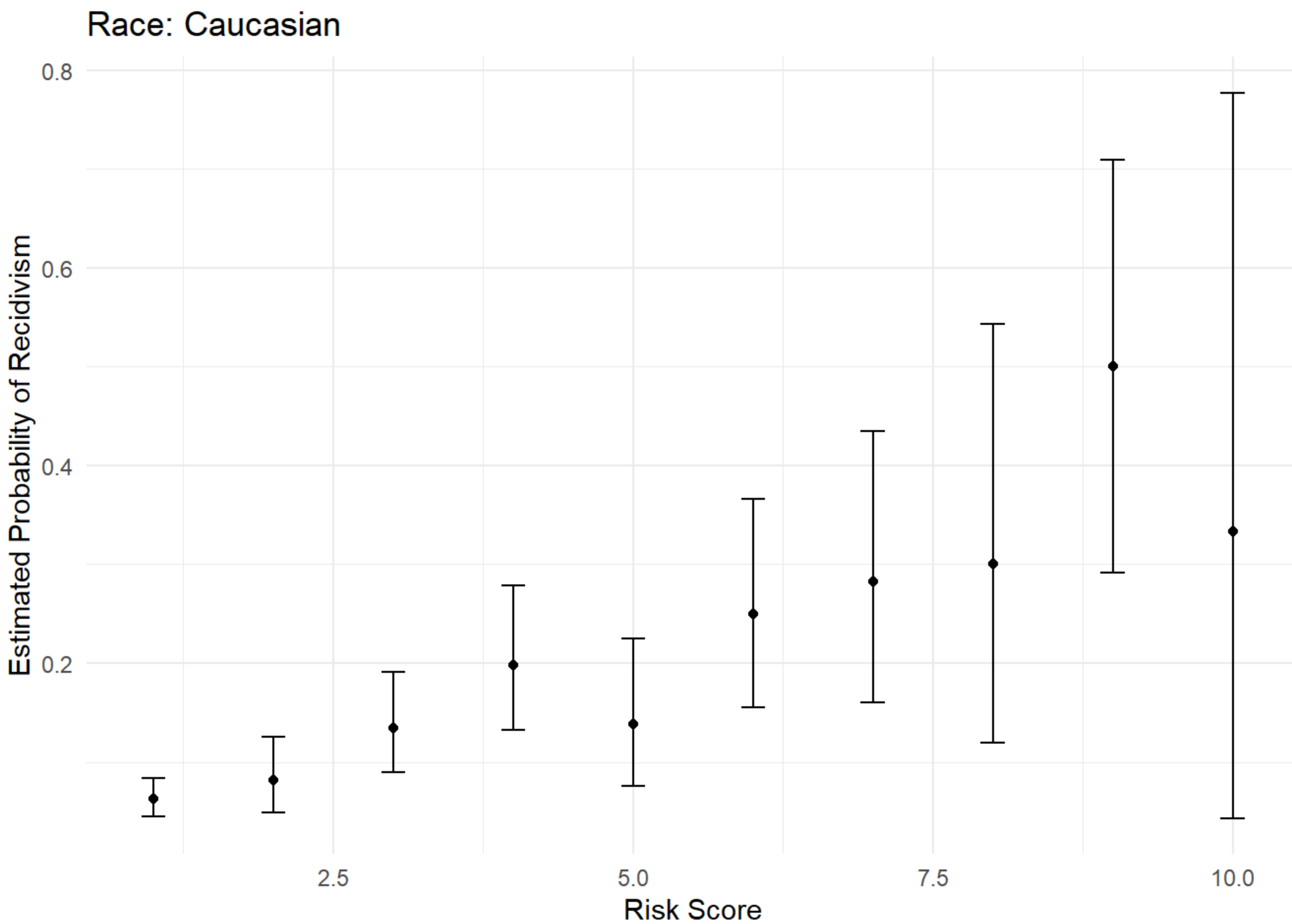
```
## # A tibble: 10 × 7
##    score recidivism_rate     n lower_ci upper_ci prob_lower_ci prob_upper_ci
##    <dbl>           <dbl> <int>    <dbl>    <dbl>         <dbl>         <dbl>
##  1     1          0.0387   155   0.0143   0.0823       0.00834        0.0691
##  2     2          0.183     60   0.0952   0.304        0.0854         0.281
##  3     3          0.114     44   0.0379   0.246        0.0199         0.207
##  4     4          0.12      25   0.0255   0.312       -0.00738        0.247
##  5     5          0.0968    31   0.0204   0.258       -0.00730        0.201
##  6     6          0.1       20   0.0123   0.317       -0.0315         0.231
##  7     7          0          9   0        0.336        0              0
##  8     8          0.429      7   0.0990   0.816        0.0620         0.795
##  9     9          0.667      3   0.0943   0.992        0.133          1.20
## 10    10          0          1   0        0.975        0              0
## [1] "Asian"
## # A tibble: 7 × 7
##   score recidivism_rate     n lower_ci upper_ci prob_lower_ci prob_upper_ci
##   <dbl>           <dbl> <int>    <dbl>    <dbl>         <dbl>         <dbl>
## 1     1             0      15   0         0.218          0              0
## 2     2             0       2   0         0.842          0              0
## 3     3             0       1   0         0.975          0              0
## 4     4             0       4   0         0.602          0              0
## 5     5             1       1   0.0250    1              1              1
## 6     6             0.5     2   0.0126    0.987         -0.193          1.19
## 7     8             1       1   0.0250    1              1              1
## [1] "Native American"
## # A tibble: 5 × 7
##   score recidivism_rate     n lower_ci upper_ci prob_lower_ci prob_upper_ci
##   <dbl>           <dbl> <int>    <dbl>    <dbl>         <dbl>         <dbl>
## 1     1               0     2   0         0.842             0             0
## 2     2               0     1   0         0.975             0             0
## 3     4               0     2   0         0.842             0             0
## 4     6               0     1   0         0.975             0             0
## 5     9               1     1   0.0250    1                 1             1
```

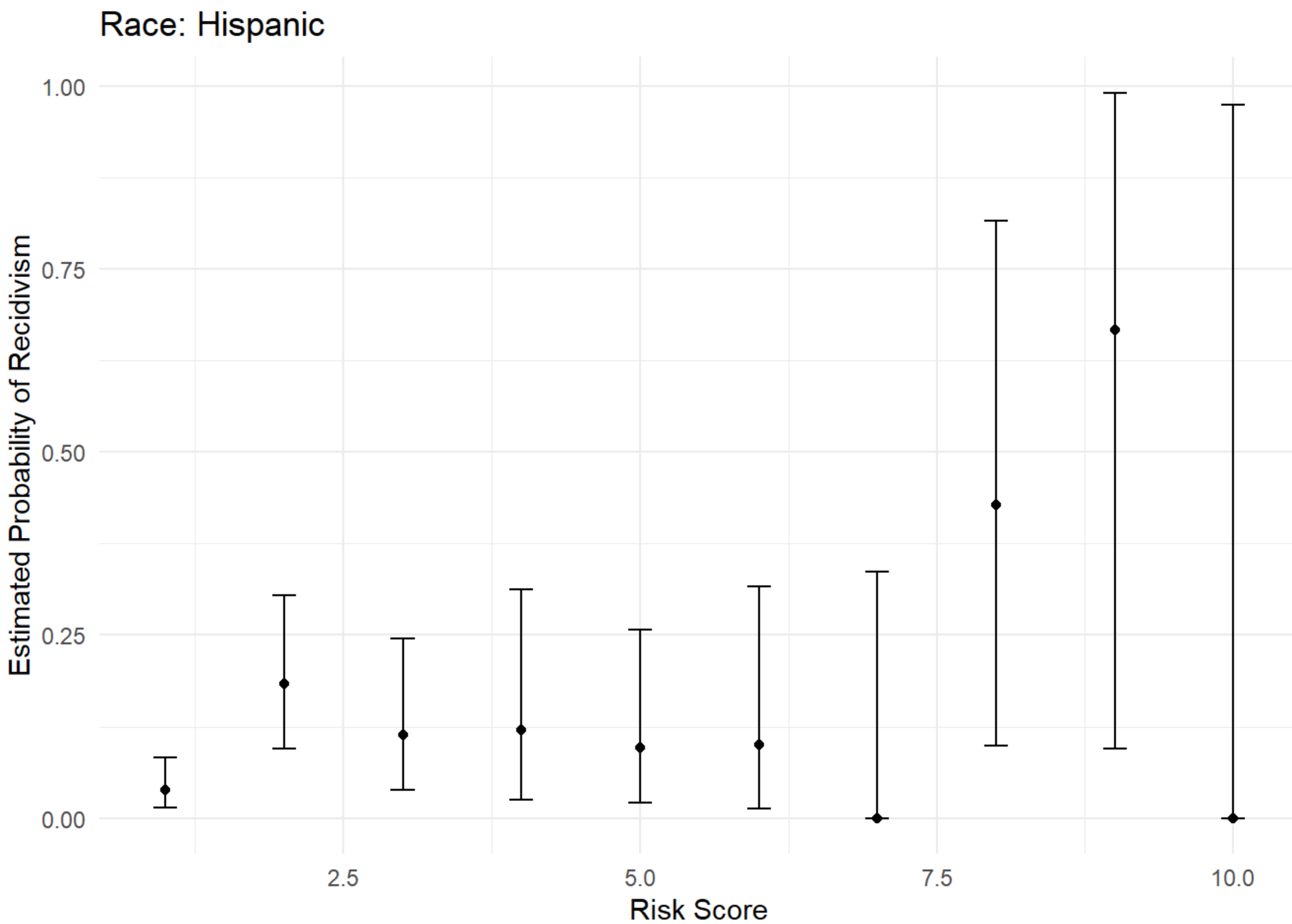
```r
# Print or display the plots
for (i in seq_along(plots)) {
  print(plots[[i]])
}
```

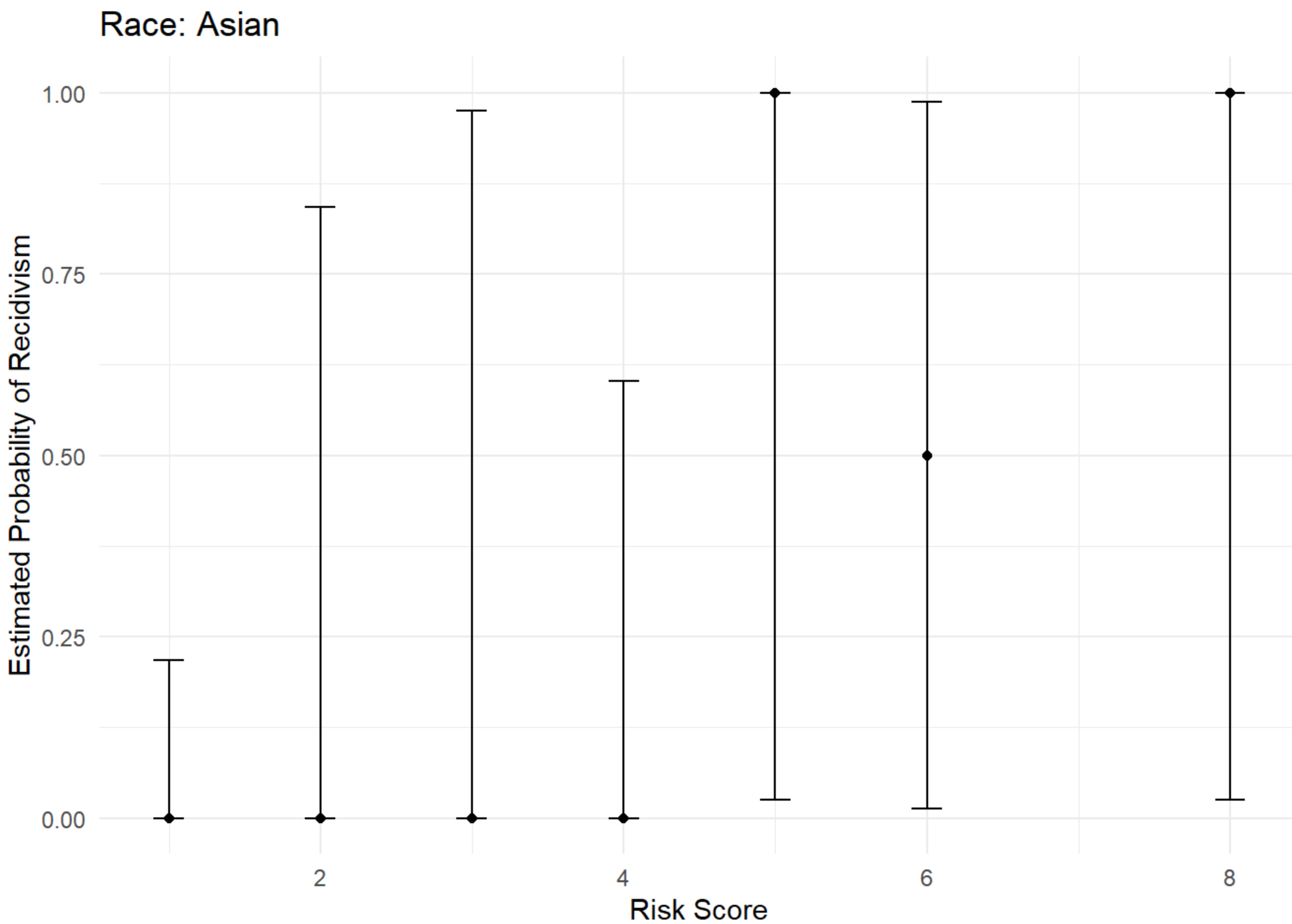

Race: Other
Estimated Probability of Recidivism
1.00
0.75
0.50
0.25
0.00
2.5
5.0
7.5
10.0
Risk Score

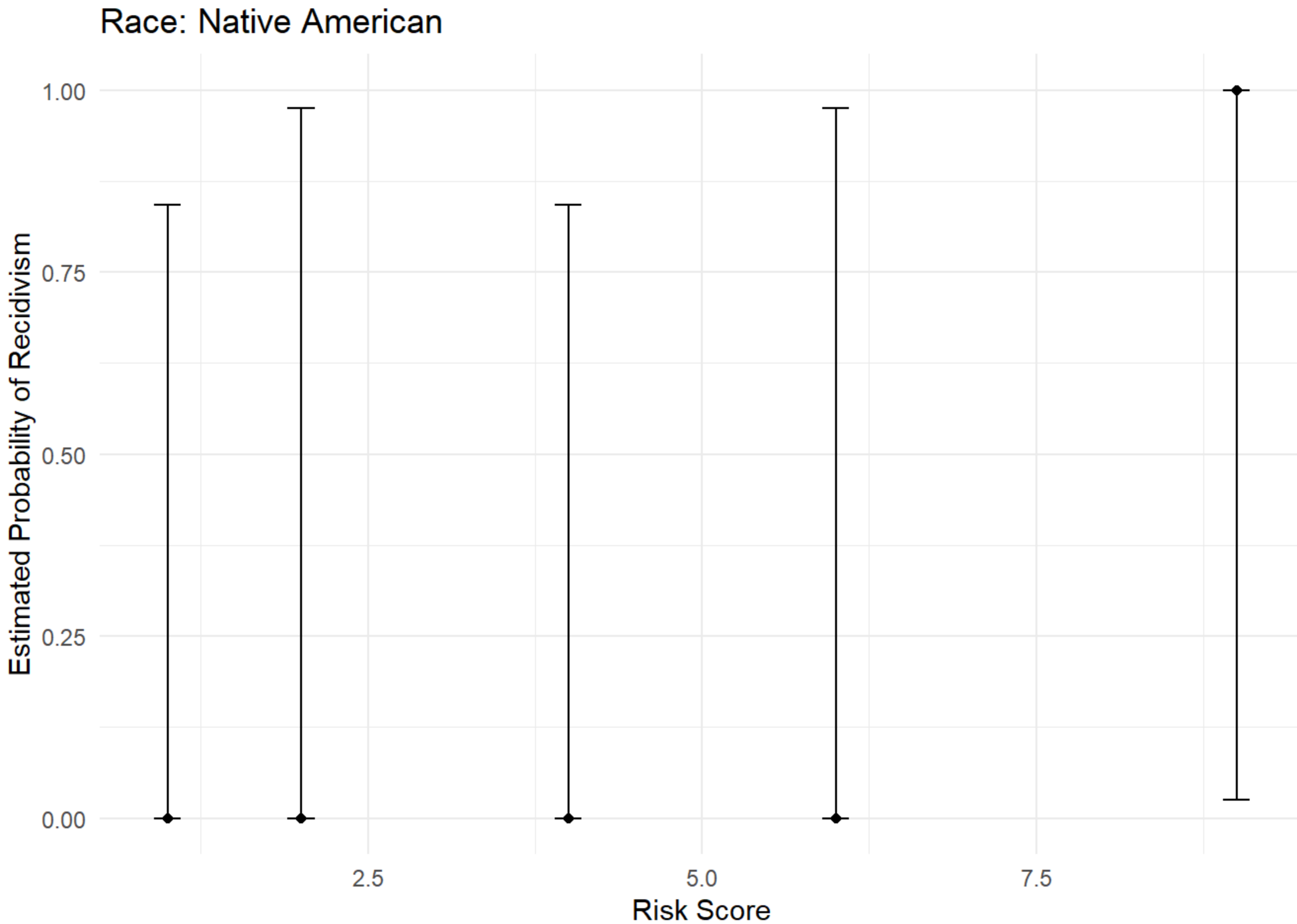


Race: African-American
Estimated Probability of Recidivism
0.6
0.4
0.2
2.5
5.0
7.5
10.0
Risk Score


Race: Caucasian
Estimated Probability of Recidivism
0.8
0.6
0.4
0.2
2.5
5.0
7.5
10.0
Risk Score


Race: Hispanic
Estimated Probability of Recidivism
1.00
0.75
0.50
0.25
0.00
2.5
5.0
7.5
10.0
Risk Score


Race: Asian
Estimated Probability of Recidivism
1.00
0.75
0.50
0.25
0.00
2
4
6
8
Risk Score



The data for asians and native americans make sense since the sample size is small.

As in part b, one thing that doesn't make a lot of sense to me is how my probability drops for risk scores of 10. It is possible that there's an error in my code. If there isn't an error, this could be explained by errors in the scoring system defaulting to a value of 10 for certain people or that individuals with really high risk scores may tend to be more cautious of repeat offenses due to past experiences.

It does makes sense that the CI's grow with the risk score, since it's easier (and more likely) to predict that someone won't be a repeat offender rather than that they will.

It's interesting that the CI for african americans is the lowest at high risk scores. This make sense though since that group has a higher rate of recidivism.

Some datapoints show a probability of 1 or 0 or a number that doesn't align with the trend, but those are usually points that have a low (or even a single) data points for generating that probability for the group.

d. ROC Curves

Use the raw COMPAS risk scores to make a ROC curve for each race.

- Are the best discriminating models the ones you expected?
- Are the ROC curves helpful in evaluating the COMPAS risk score?

::: {.callout-note title="Solution"}

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
# Create a list to store the ROC curves
roc_curves <- list()

# Loop through each race category
for (race_i in race_categories) {
  # Subset the data for the current race
  race_data <- filter(risk, risk$race == race_i)
  # Create ROC curve for the current race
  roc_curve <- roc(outcome ~ score, data = race_data)
  # Store the ROC curve in the list
  roc_curves[[race_i]] <- roc_curve
}
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

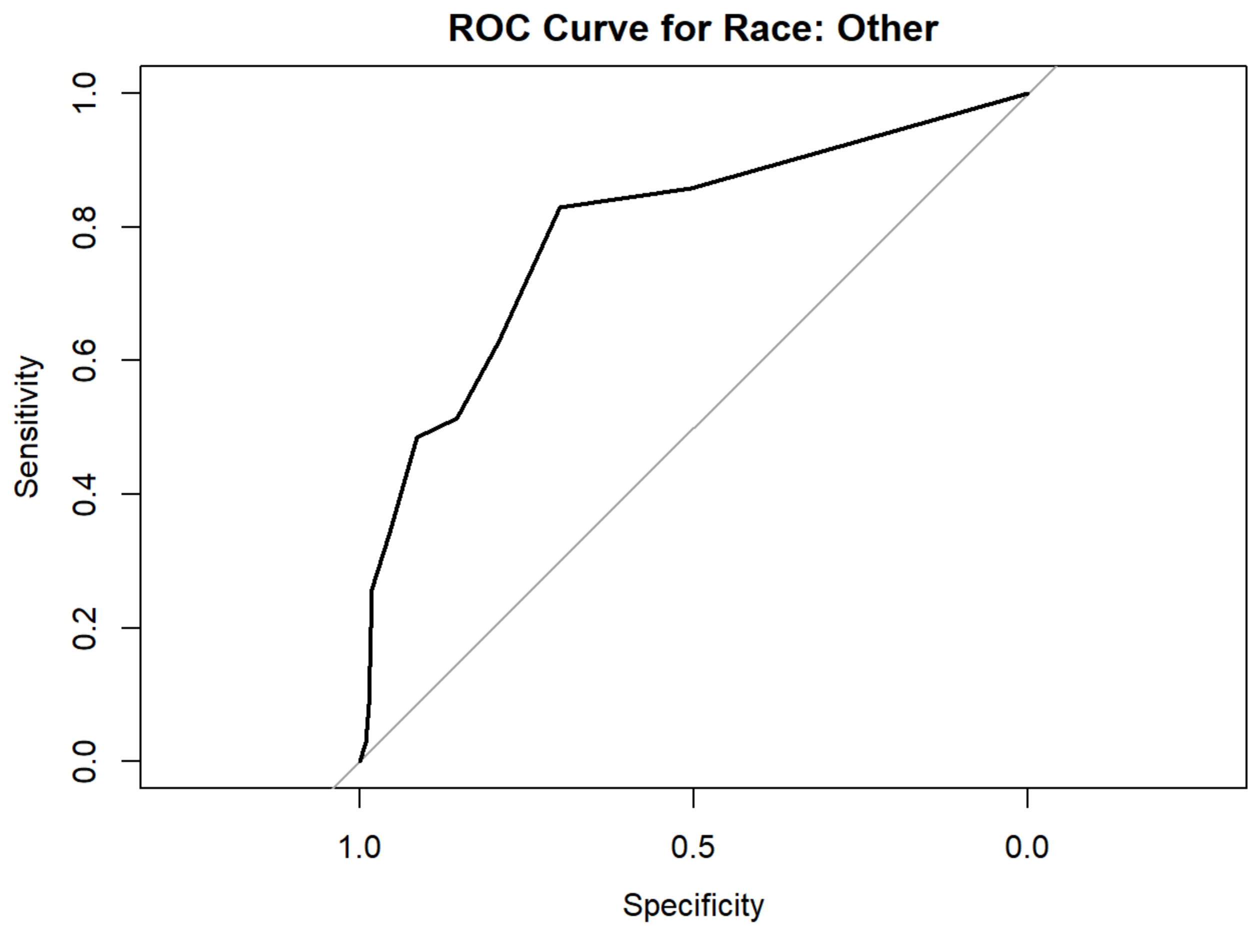
```
## Setting levels: control = 0, case = 1
```

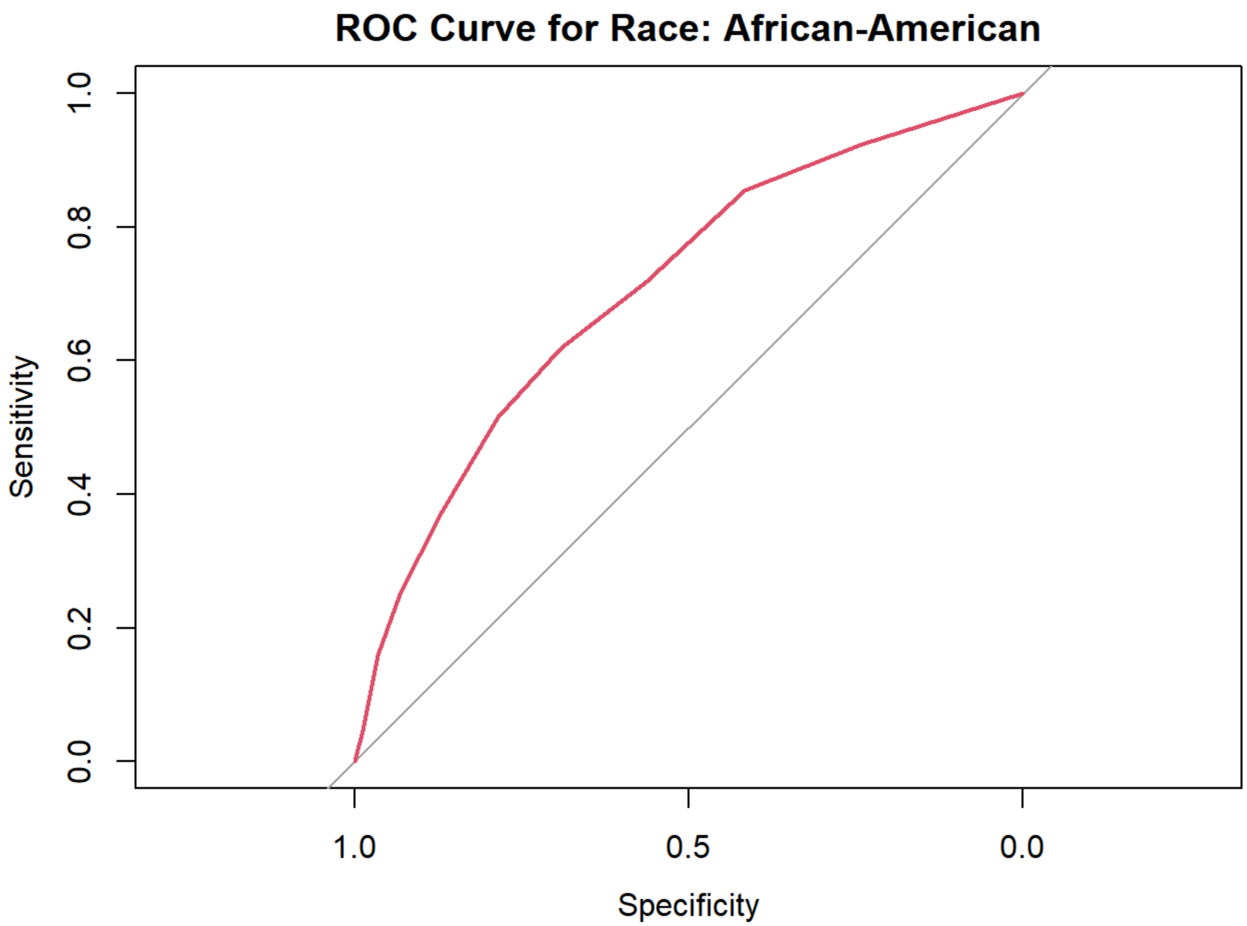
```
## Setting direction: controls < cases
```

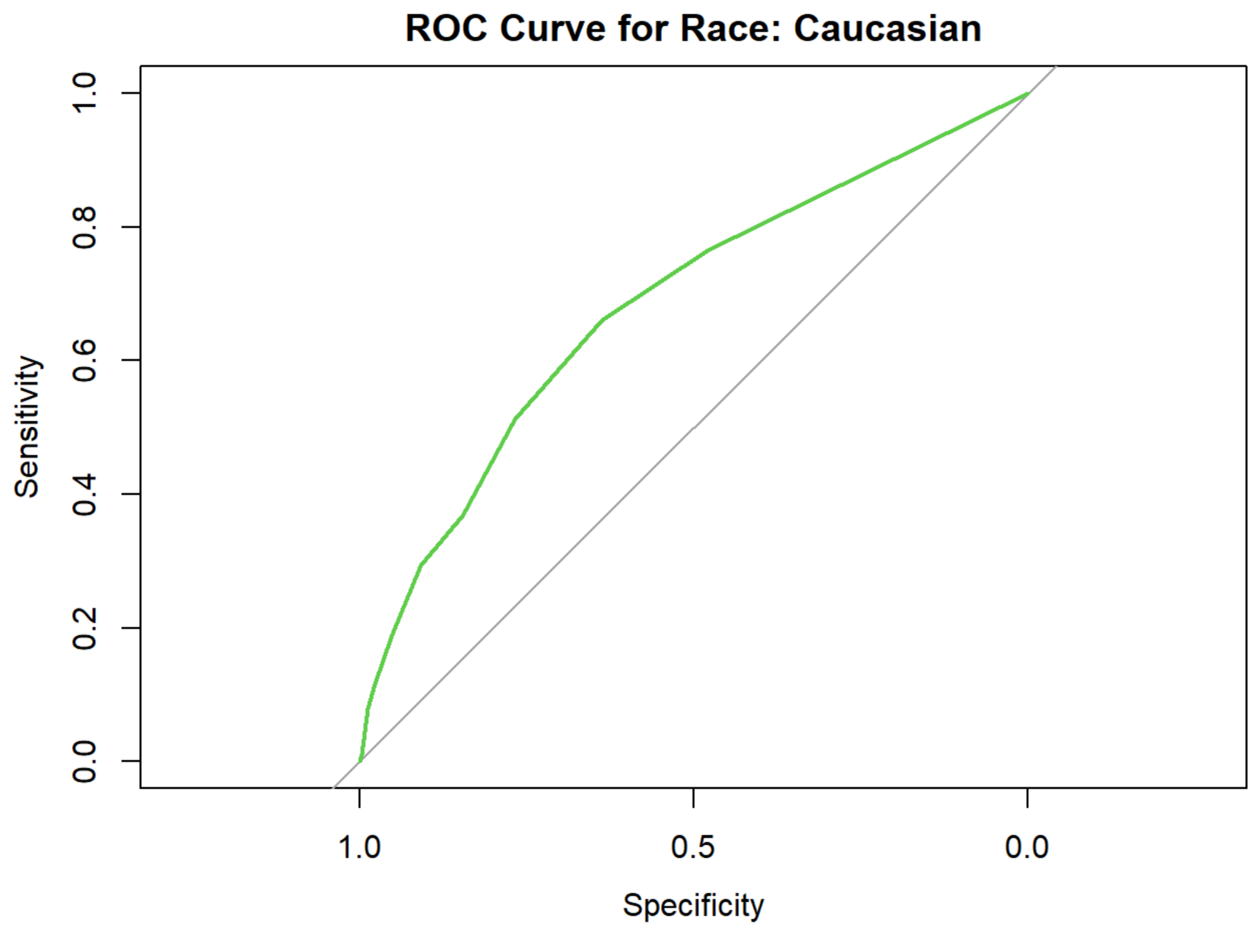
```
## Setting levels: control = 0, case = 1
```

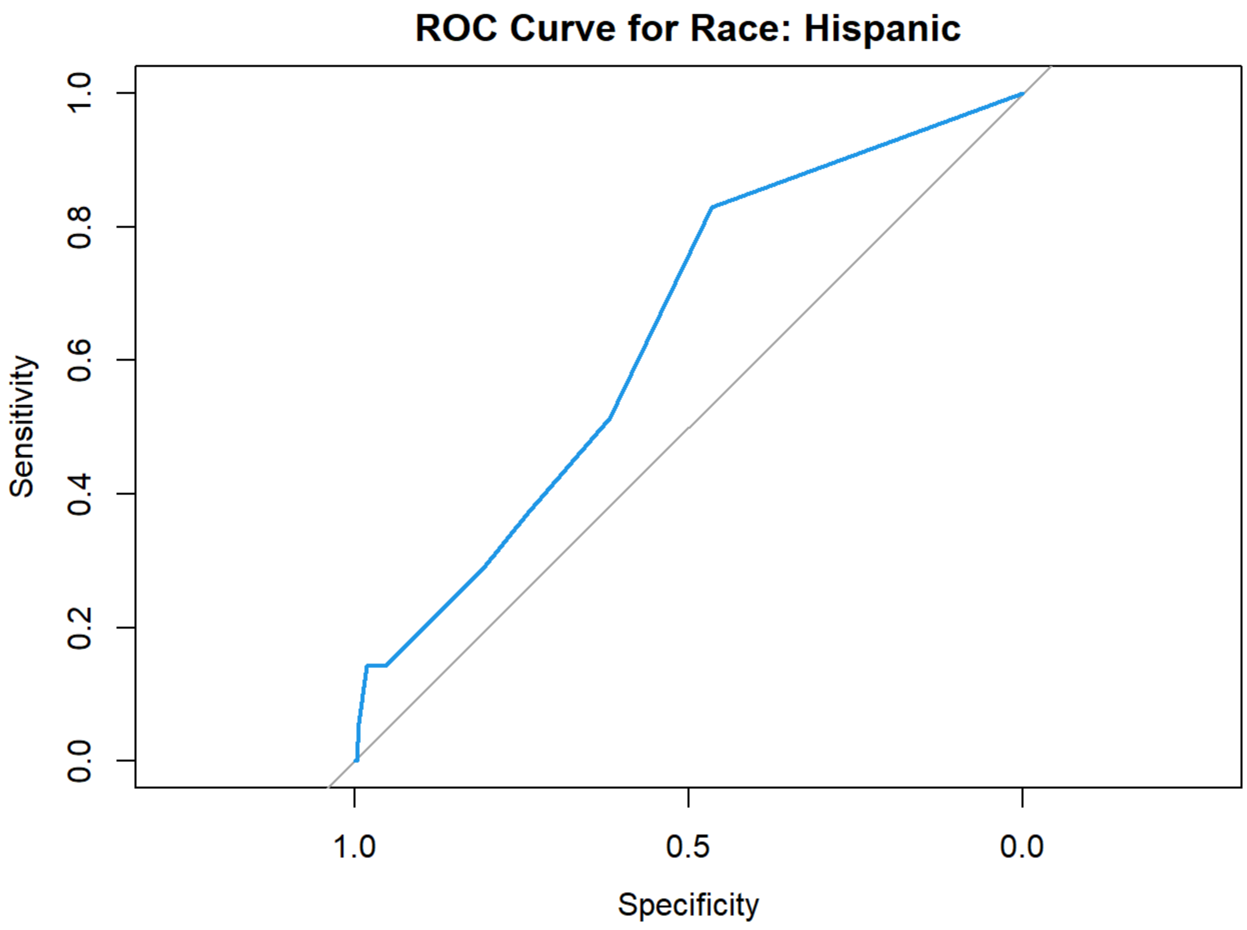
```
## Setting direction: controls < cases
```

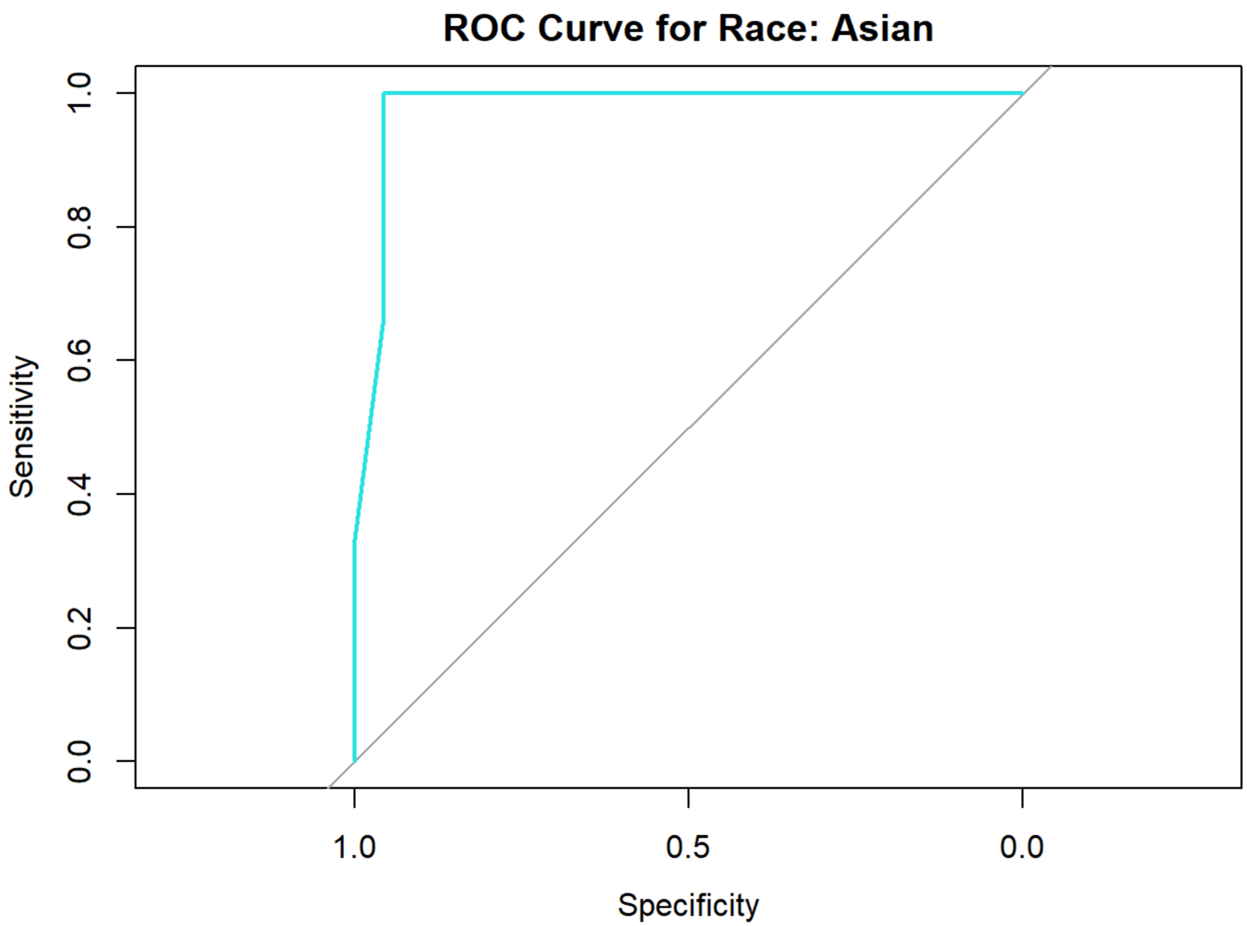
```
# Plot the ROC curves for each race
for (i in seq_along(roc_curves)) {
  plot(roc_curves[[i]], main = paste("ROC Curve for Race:", race_categories[i]), col = i)
}
```

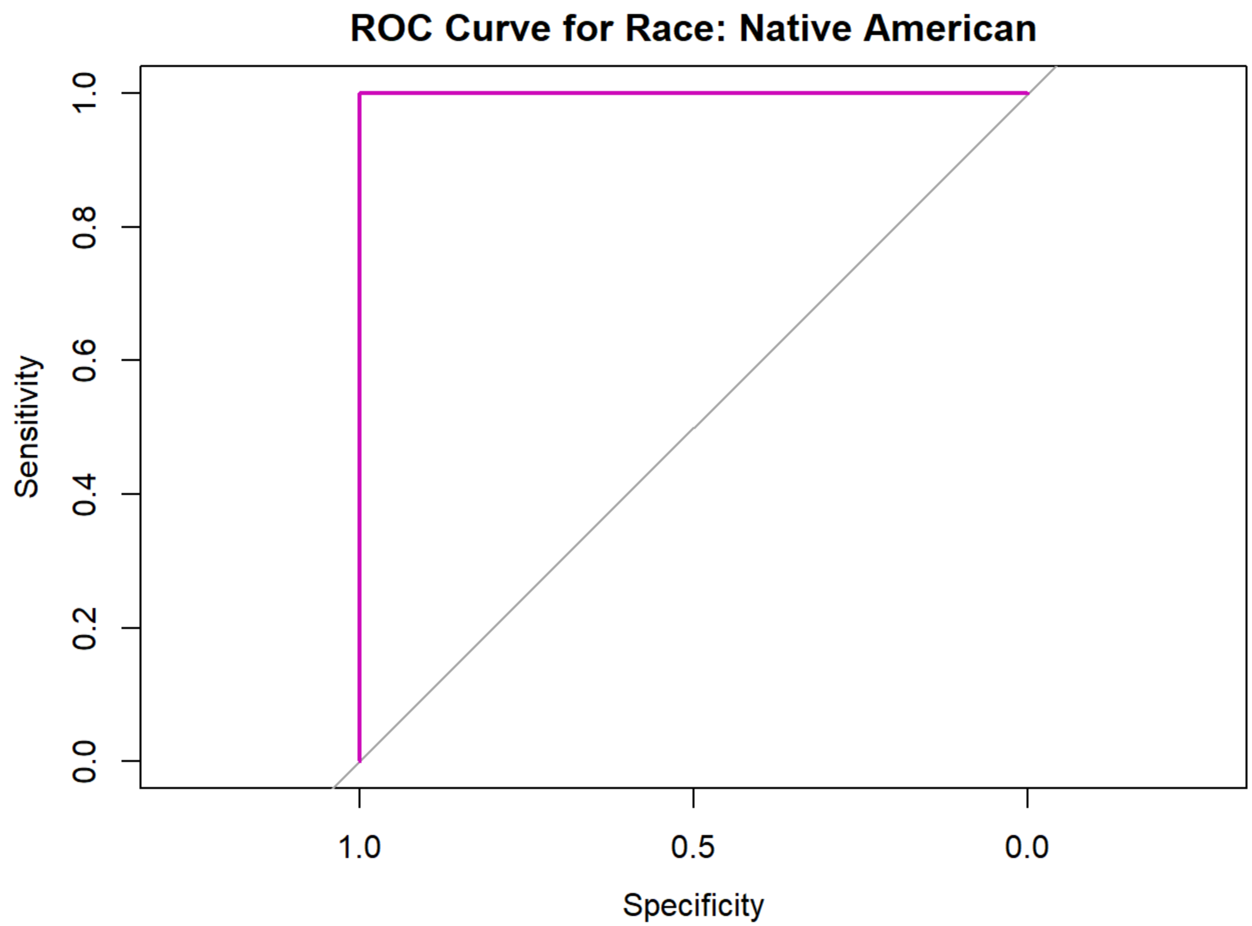

ROC Curve for Race: Other
Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
Specificity
1.0
0.5
0.0


ROC Curve for Race: African-American
Sensitivity
Specificity
1.0
0.8
0.6
0.4
0.2
0.0
1.0
0.5
0.0


ROC Curve for Race: Caucasian
Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
Specificity
1.0
0.5
0.0


ROC Curve for Race: Hispanic
Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
1.0
0.5
0.0
Specificity


ROC Curve for Race: Asian
Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
1.0
0.5
0.0
Specificity




```
# Print or display the plots
for (i in seq_along(roc_curves)) {
  print(roc_curves[[i]])
}
```

```
##
## Call:
## roc.formula(formula = outcome ~ score, data = race_data)
##
## Data: score in 220 controls (outcome 0) < 35 cases (outcome 1).
## Area under the curve: 0.7917
##
## Call:
## roc.formula(formula = outcome ~ score, data = race_data)
##
## Data: score in 1514 controls (outcome 0) < 404 cases (outcome 1).
## Area under the curve: 0.7083
##
## Call:
## roc.formula(formula = outcome ~ score, data = race_data)
##
## Data: score in 1285 controls (outcome 0) < 174 cases (outcome 1).
## Area under the curve: 0.6826
##
## Call:
## roc.formula(formula = outcome ~ score, data = race_data)
##
## Data: score in 320 controls (outcome 0) < 35 cases (outcome 1).
## Area under the curve: 0.6413
##
## Call:
## roc.formula(formula = outcome ~ score, data = race_data)
##
## Data: score in 23 controls (outcome 0) < 3 cases (outcome 1).
## Area under the curve: 0.9783
##
## Call:
## roc.formula(formula = outcome ~ score, data = race_data)
##
## Data: score in 6 controls (outcome 0) < 1 cases (outcome 1).
## Area under the curve: 1
```

Interpretation:

The asian and native american curves show very high differentiation for recidivism using the scores. The sample size for these groups are very small though, similar to as if the training data is very small for creating a model. Interestingly, the the AUCs for african american > caucasian > hispanic. This was not something I was expecting. Though it’s not significantly different, the scores seem to be the best at differentiating recidivism for african americans. This makes sense though, because even though there are more caucasian datapoints, african americans have a higher recidivism rate from the data given. The hispanic ROC curve is also interesting, but again, there aren't a lot of datapoints for hispanics with higher risk scores. Are the ROC curves helpful in evaluating the COMPAS risk score? I feel like it is difficult to draw conclusions using the ROC curves because the sample size, sample distribution for each race, and even judiciary prejudices affect so much of the data before it gets to creating a ROC curve. I feel like ROC curves can help justify decisions made elsewhere, but evaluating the risk scores based solely on the ROC curves seems a bit too presumptuous.