Project2_Task4

```
library(glmnet)
```

```
## Warning: package 'glmnet' was built under R version 4.4.3
```

```
## Loading required package: Matrix
```

```
## Warning: package 'Matrix' was built under R version 4.4.3
```

```
## Loaded glmnet 4.1-10
```

```
library(pROC)
```

```
## Warning: package 'pROC' was built under R version 4.4.2
```

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

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

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

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.4.2
```

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 4.4.3
```

```
## Loading required package: lattice
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.4.3
```

```
## Warning: package 'purrr' was built under R version 4.4.3
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.3     v tibble    3.2.1
## v purrr     1.1.0     v tidyr     1.3.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x tidyr::expand() masks Matrix::expand()
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x purrr::lift()   masks caret::lift()
## x tidyr::pack()   masks Matrix::pack()
## x tidyr::unpack() masks Matrix::unpack()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(modelr)
library(rsample)
```

```
## Warning: package 'rsample' was built under R version 4.4.3

##
## Attaching package: 'rsample'
##
## The following object is masked from 'package:caret':
##
##     calibration
```

```r
library(yardstick)
```

```
## Warning: package 'yardstick' was built under R version 4.4.2

##
## Attaching package: 'yardstick'
##
## The following objects are masked from 'package:modelr':
##
##     mae, mape, rmse
##
## The following object is masked from 'package:readr':
##
##     spec
##
## The following objects are masked from 'package:caret':
##
##     precision, recall, sensitivity, specificity
```

```r
library(ggplot2)
```

```r
load(here::here("data", "biomarker-clean.RData"))

head(biomarker_clean)
```

```
## # A tibble: 6 x 1,319
##   group  ados    CHIP  CEBPB     NSE   PIAS4 `IL-10 Ra`  STAT3   IRF1 `c-Jun`
##   <chr> <dbl>   <dbl>  <dbl>   <dbl>   <dbl>      <dbl>  <dbl>  <dbl>   <dbl>
## 1 ASD       8  0.335   0.520 -0.554   0.650      -0.358  0.305 -0.484   0.309
## 2 ASD      21 -0.0715  1.01   3       1.28       -0.133  1.13   0.253   0.408
## 3 ASD      12 -0.406  -0.531 -0.0592  1.13        0.554 -0.334  0.287  -0.845
## 4 ASD      20 -0.102  -0.251  1.47    0.0773     -0.705  0.893  2.61   -0.372
## 5 ASD      22 -0.395  -0.536  0.0410 -0.299      -0.830  0.899  1.01   -0.843
## 6 ASD      17 -0.126   1.27  -0.892   0.239      -0.344  0.216  0.211   0.221
## # i 1,309 more variables: `Mcl-1` <dbl>, OAS1 <dbl>, `c-Myc` <dbl>,
## #   SMAD3 <dbl>, SMAD2 <dbl>, `IL-23` <dbl>, PDGFRA <dbl>, `IL-12` <dbl>,
## #   STAT1 <dbl>, STAT6 <dbl>, LRRK2 <dbl>, Osteocalcin <dbl>, `IL-5` <dbl>,
## #   GPDA <dbl>, IgA <dbl>, LPPL <dbl>, HEMK2 <dbl>, PDXK <dbl>, TLR4 <dbl>,
## #   REG4 <dbl>, `HSP 27` <dbl>, `YKL-40` <dbl>, `Alpha enolase` <dbl>,
## #   `Apo L1` <dbl>, CD38 <dbl>, CD59 <dbl>, FABPL <dbl>, `GDF-11` <dbl>,
## #   BTC <dbl>, `HIF-1a` <dbl>, S100A6 <dbl>, SECTM1 <dbl>, RSPO3 <dbl>, ...
```

```
biomarker_data <- biomarker_clean %>%
  select(-starts_with("ados"))

biomarker_data$group <- factor(biomarker_data$group)
```

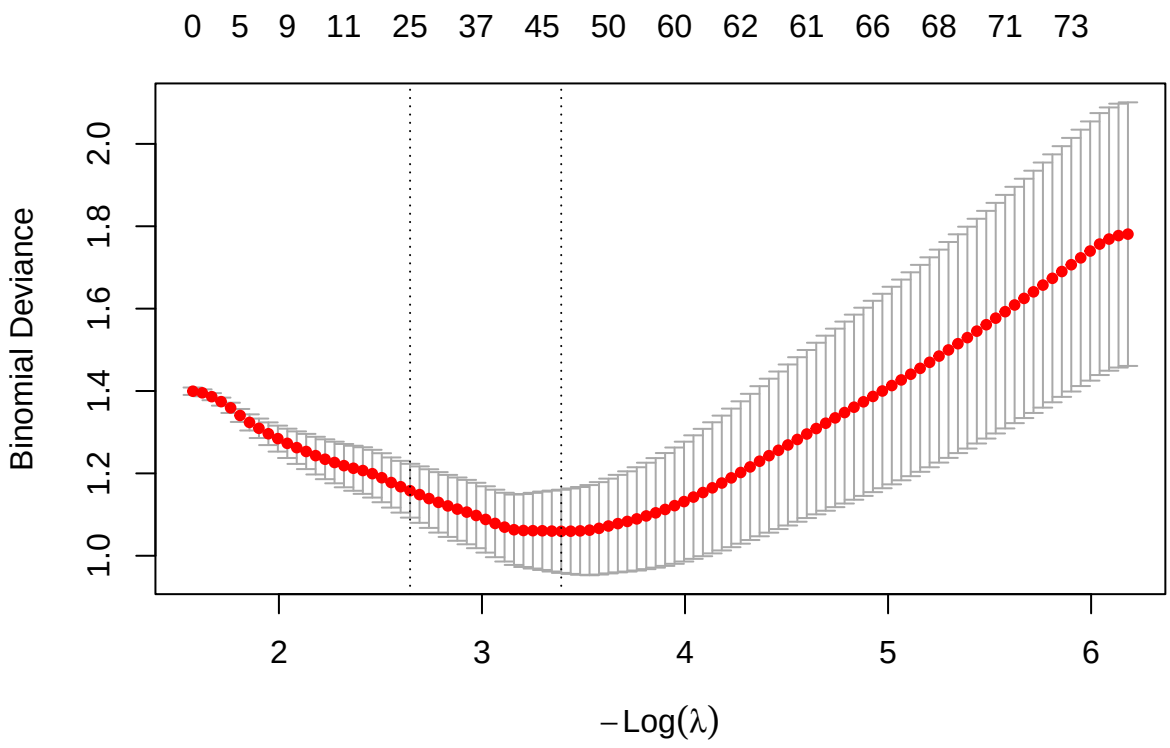
```
set.seed(1)
partitions <- biomarker_data %>%
  initial_split(prop = 0.8)

partitions
```

```
## <Training/Testing/Total>
## <123/31/154>
```

```
train_data <- training(partitions)
test_data <- testing(partitions)

X_train <- train_data[, !(names(train_data) %in% "group")]
y_train <- train_data$group
X_test <- test_data[, !(names(test_data) %in% "group")]
y_test <- test_data$group
```

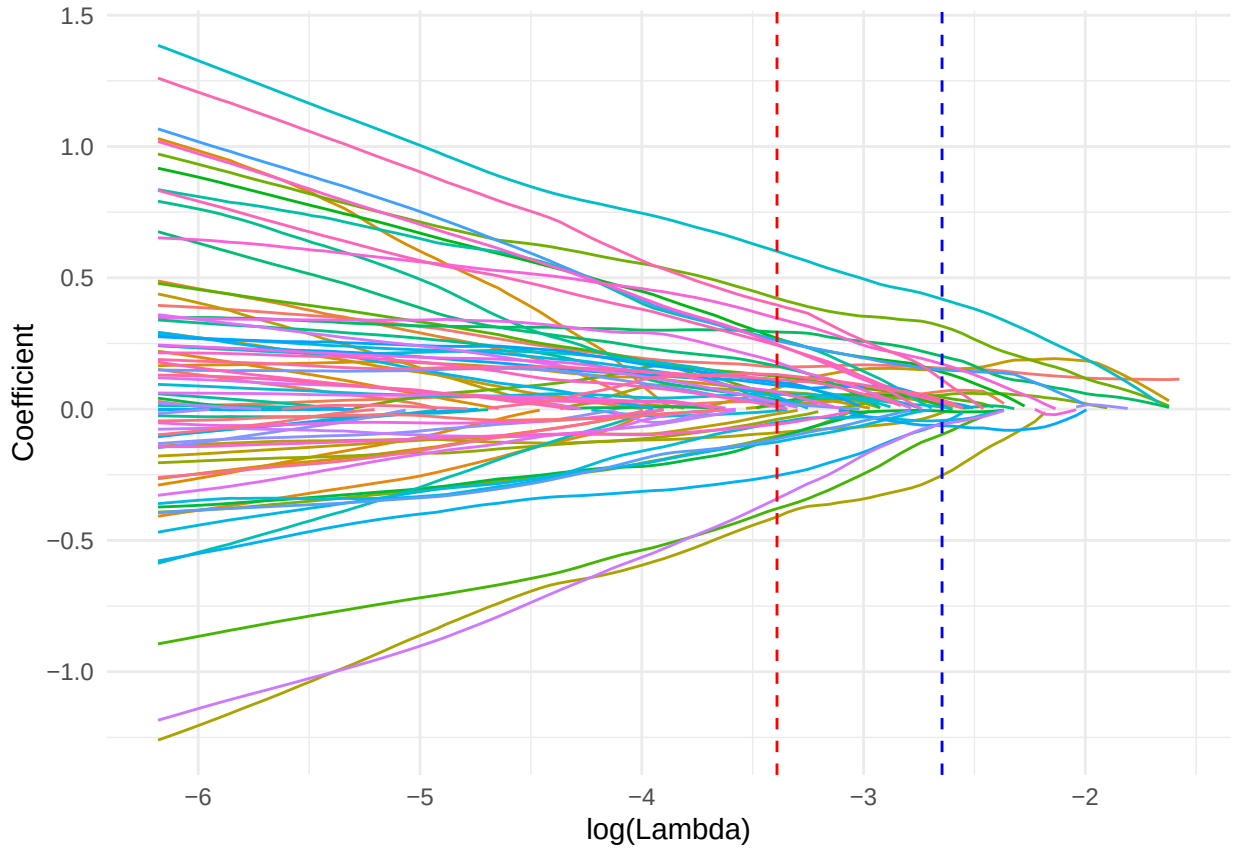
Fit lasso logistic regression (feature selection)

```
X_train_mat <- as.matrix(X_train)
X_test_mat <- as.matrix(X_test)

# 10-fold cross-validation for lambda
cv_fit <- cv.glmnet(
  X_train_mat, y_train,
  family = "binomial",
  alpha = 1,
  nfolds = 10,
  type.measure = 'deviance')

plot(cv_fit)
```




```
lambda_min <- cv_fit$lambda.min
lambda_1se <- cv_fit$lambda.1se
```

```
# LASSO estimates
fit <- glmnet(X_train, y_train, family = "binomial")
fit_df <- tidy(fit)

ggplot(fit_df, aes(x = log(lambda), y = estimate, color = term)) +
  geom_line() +
  theme_minimal() +
  labs(x = "log(Lambda)", y = "Coefficient") +
  theme(legend.position = "none") +
  geom_vline(xintercept = log(lambda_min), linetype = "dashed", color = "red") +
  geom_vline(xintercept = log(lambda_1se), linetype = "dashed", color = "blue")
```



Identify selected biomarkers

```
coef_lasso <- coef(cv_fit, s = "lambda.1se")
selected_idx <- which(coef_lasso != 0)
selected_features <- rownames(coef_lasso)[selected_idx][-1] # drop intercept

cat("Selected proteins:\n")
```

```
## Selected proteins:
```

```
print(selected_features)
```

```
##  [1] "IL-5"                      "LPPL"
##  [3] "CD59"                      "FSTL1"
##  [5] "CXCL16, soluble"           "CD30"
##  [7] "Protein S"                 "Kallikrein 11"
##  [9] "PAI-1"                     "IGFBP-4"
## [11] "TGF-b R III"               "MAPK2"
## [13] "ETHE1"                     "ENPP7"
## [15] "ENTP5"                     "Calcineurin"
## [17] "IgD"                       "Lysozyme"
## [19] "DERM"                      "EPHB2"
## [21] "SIG14"                     "CD27"
## [23] "SRCN1"                     "Epo"
## [25] "14-3-3 protein zeta/delta"
```

```r
cat("Total:", length(selected_features), "proteins\n")
```

```
## Total: 25 proteins
```

```r
train_sel <- train_data[, c(selected_features, "group")]
test_sel <- test_data[, c(selected_features, "group")]

model_alt <- glm(group ~ ., data = train_sel, family = "binomial")
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```r
summary(model_alt)
```

```
##
## Call:
## glm(formula = group ~ ., family = "binomial", data = train_sel)
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                    55.940  23669.931   0.002    0.998
## `IL-5`                        -46.123  71162.118  -0.001    0.999
## LPPL                          -84.702 125153.785  -0.001    0.999
## CD59                          -20.415 216172.036   0.000    1.000
## FSTL1                        -118.402  51318.745  -0.002    0.998
## `CXCL16, soluble`              11.455  63530.453   0.000    1.000
## CD30                         -167.353  75807.335  -0.002    0.998
## `Protein S`                   102.215  19889.193   0.005    0.996
## `Kallikrein 11`              -120.062  44052.925  -0.003    0.998
## `PAI-1`                       -74.997  57526.885  -0.001    0.999
## `IGFBP-4`                     149.466 101298.427   0.001    0.999
## `TGF-b R III`                  95.546 111313.949   0.001    0.999
## MAPK2                         101.498  91065.769   0.001    0.999
## ETHE1                          11.187  50154.792   0.000    1.000
## ENPP7                          48.704  57310.282   0.001    0.999
## ENTP5                         -81.751  77205.639  -0.001    0.999
## Calcineurin                    90.691  76721.568   0.001    0.999
## IgD                           188.693  43835.986   0.004    0.997
## Lysozyme                        9.836  21698.180   0.000    1.000
## DERM                          144.403  68146.822   0.002    0.998
## EPHB2                          42.780 120605.049   0.000    1.000
## SIG14                         132.396  79516.162   0.002    0.999
## CD27                           64.095  65640.669   0.001    0.999
## SRCN1                         195.162  58311.120   0.003    0.997
## Epo                            93.227  83966.213   0.001    0.999
## `14-3-3 protein zeta/delta`  -140.063  56447.881  -0.002    0.998
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1.7012e+02  on 122  degrees of freedom
```

```
## Residual deviance: 1.1556e-07  on  97  degrees of freedom
## AIC: 52
##
## Number of Fisher Scoring iterations: 25
```

```
probs <- predict(model_alt, newdata = test_sel, type = "response")
preds <- ifelse(probs > 0.5, 1, 0)
```

```
roc_obj <- roc(y_test, probs)
```

```
## Setting levels: control = ASD, case = TD
```

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

```
auc_alt <- auc(roc_obj)
accuracy <- mean(preds == as.numeric(as.character(y_test)))
```

```
## Warning in mean(preds == as.numeric(as.character(y_test))): NAs introduced by
## coercion
```

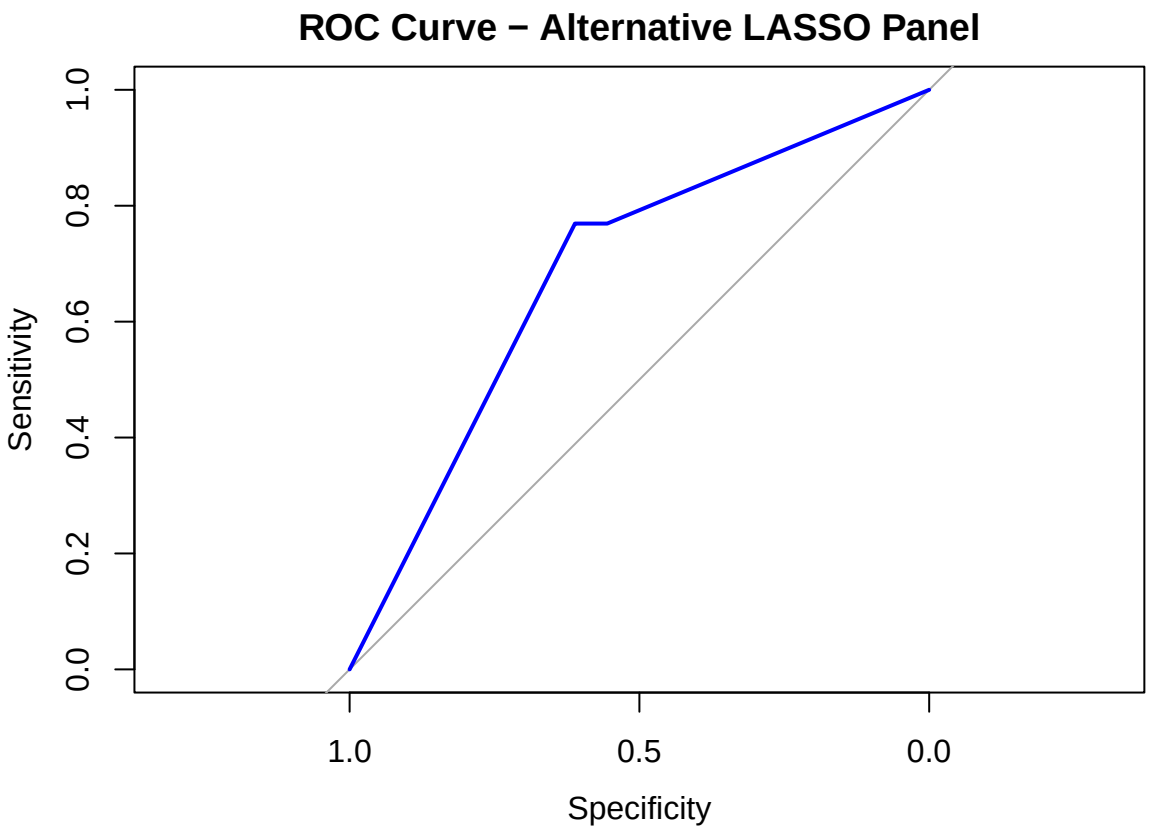
```
cat("Alternative LASSO panel AUROC:", round(auc_alt, 3), "\n")
```

```
## Alternative LASSO panel AUROC: 0.684
```

```
cat("Accuracy:", round(accuracy, 3), "\n")
```

```
## Accuracy: NA
```

```
# ROC plot
plot(roc_obj, col = "blue", main = "ROC Curve - Alternative LASSO Panel")
```


ROC Curve − Alternative LASSO Panel
Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
1.0
0.5
0.0
Specificity