Project2_Task4

```
library(glmnet)
```

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

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

```
library(pROC)
```

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

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

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

```
library(caret)
```

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

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

```
library(tidyverse)
```

```
## -- 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
```

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

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

```
library(yardstick)
```

```
##
## 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
```

```
library(ggplot2)
library(dplyr)
```

```
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>, ...
```

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

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

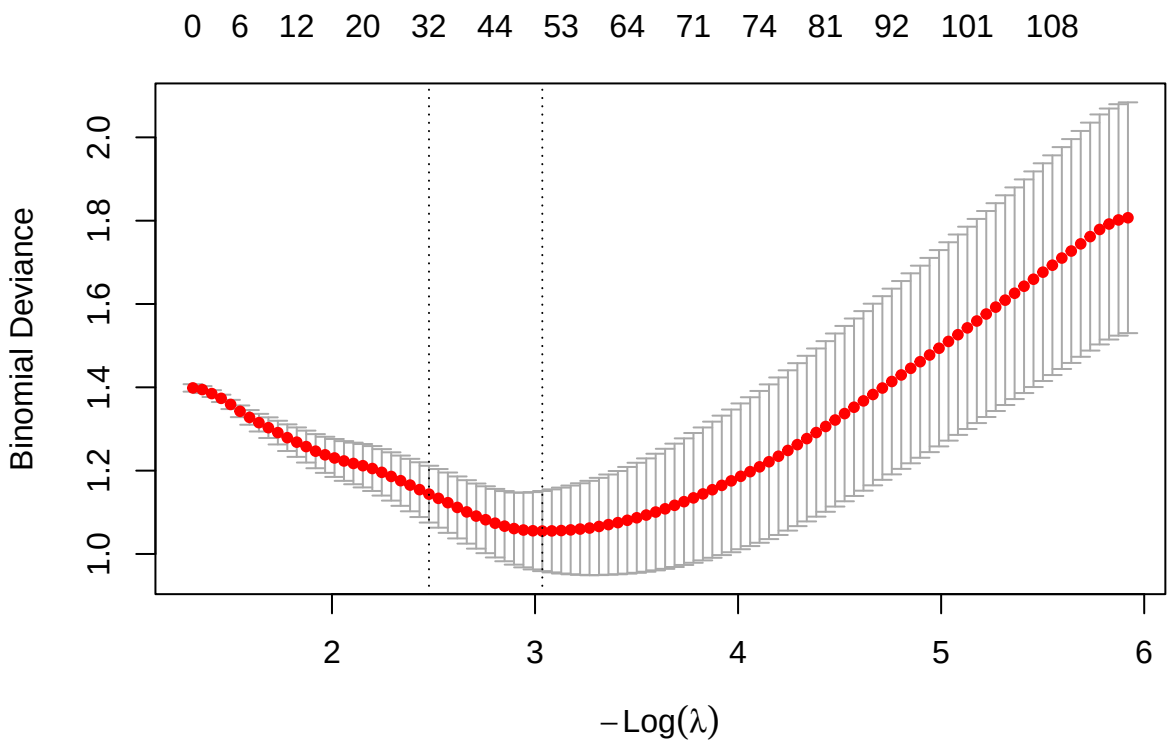
```r
set.seed(1)
partitions <- biomarker_data %>%
  initial_split(prop = 0.8)

partitions
```

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

```r
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
```

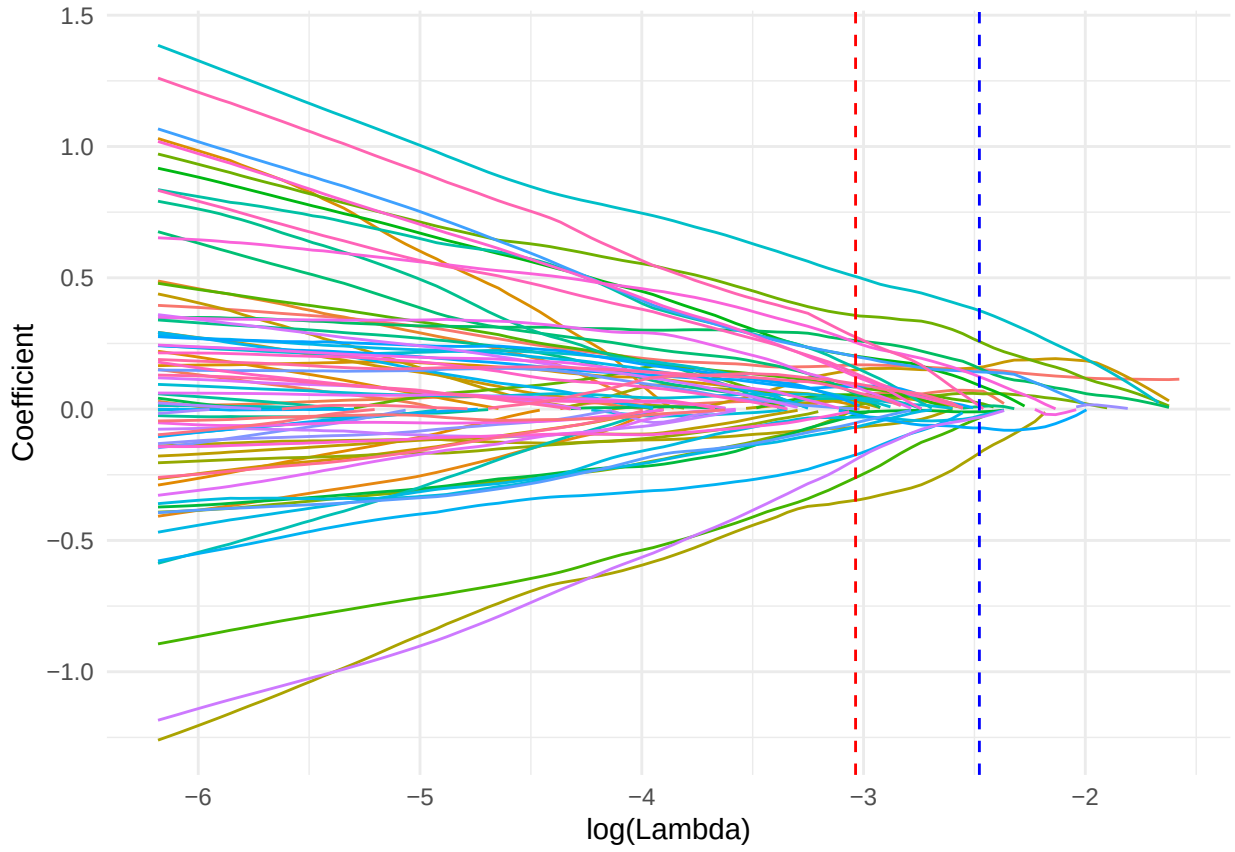
Fit lasso logistic regression (feature selection)

```r
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 = 0.77,
  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"               "HGFA"
## [13] "PYY"                       "MAPK2"
## [15] "ETHE1"                     "IL-6 sRa"
## [17] "MMP-2"                     "ENPP7"
## [19] "ENTP5"                     "Calcineurin"
## [21] "IgD"                       "Lysozyme"
## [23] "DERM"                      "hnRNP K"
## [25] "EPHB2"                     "SIG14"
```

```
## [27] "TAJ"                        "CD27"
## [29] "SRCN1"                      "Epo"
## [31] "14-3-3 protein zeta/delta" "MIG"
```

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

```
## Total: 32 proteins
```

```
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
```

```
summary(model_alt)
```

```
##
## Call:
## glm(formula = group ~ ., family = "binomial", data = train_sel)
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  2.020e+01  8.274e+04   0.000    1.000
## `IL-5`                      -6.966e-01  1.202e+05   0.000    1.000
## LPPL                         2.072e+01  5.403e+04   0.000    1.000
## CD59                        -1.554e+01  2.758e+05   0.000    1.000
## FSTL1                        9.136e+00  6.223e+04   0.000    1.000
## `CXCL16, soluble`            1.078e-01  5.148e+04   0.000    1.000
## CD30                        -1.997e+01  2.103e+05   0.000    1.000
## `Protein S`                  2.185e+01  3.149e+04   0.001    0.999
## `Kallikrein 11`             -1.406e+01  1.354e+05   0.000    1.000
## `PAI-1`                     -2.835e+01  7.229e+04   0.000    1.000
## `IGFBP-4`                    6.208e+00  4.085e+04   0.000    1.000
## `TGF-b R III`               -4.481e+00  8.620e+04   0.000    1.000
## HGFA                         1.832e+01  4.163e+04   0.000    1.000
## PYY                          1.519e+01  4.369e+04   0.000    1.000
## MAPK2                        4.429e+01  9.963e+04   0.000    1.000
## ETHE1                        3.165e+00  1.812e+05   0.000    1.000
## `IL-6 sRa`                  -4.094e+00  8.146e+04   0.000    1.000
## `MMP-2`                      3.618e+01  4.720e+04   0.001    0.999
## ENPP7                        4.960e+00  5.986e+04   0.000    1.000
## ENTP5                       -2.947e+01  3.313e+04  -0.001    0.999
## Calcineurin                  8.242e-01  7.131e+04   0.000    1.000
## IgD                          3.331e+01  1.002e+05   0.000    1.000
## Lysozyme                     1.366e+01  1.838e+05   0.000    1.000
## DERM                         1.725e+01  3.604e+04   0.000    1.000
## `hnRNP K`                    3.031e+01  1.715e+05   0.000    1.000
## EPHB2                        1.119e+01  1.414e+05   0.000    1.000
```

```
## SIG14                         2.658e+01  1.105e+05   0.000    1.000
## TAJ                           1.684e+01  5.439e+04   0.000    1.000
## CD27                         -4.674e+00  1.564e+05   0.000    1.000
## SRCN1                         2.905e+01  2.056e+05   0.000    1.000
## Epo                           6.298e+00  1.544e+05   0.000    1.000
## `14-3-3 protein zeta/delta` -8.926e-01  1.515e+05   0.000    1.000
## MIG                           1.069e+00  1.326e+05   0.000    1.000
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1.7012e+02  on 122  degrees of freedom
## Residual deviance: 1.0213e-08  on  90  degrees of freedom
## AIC: 66
##
## 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
```

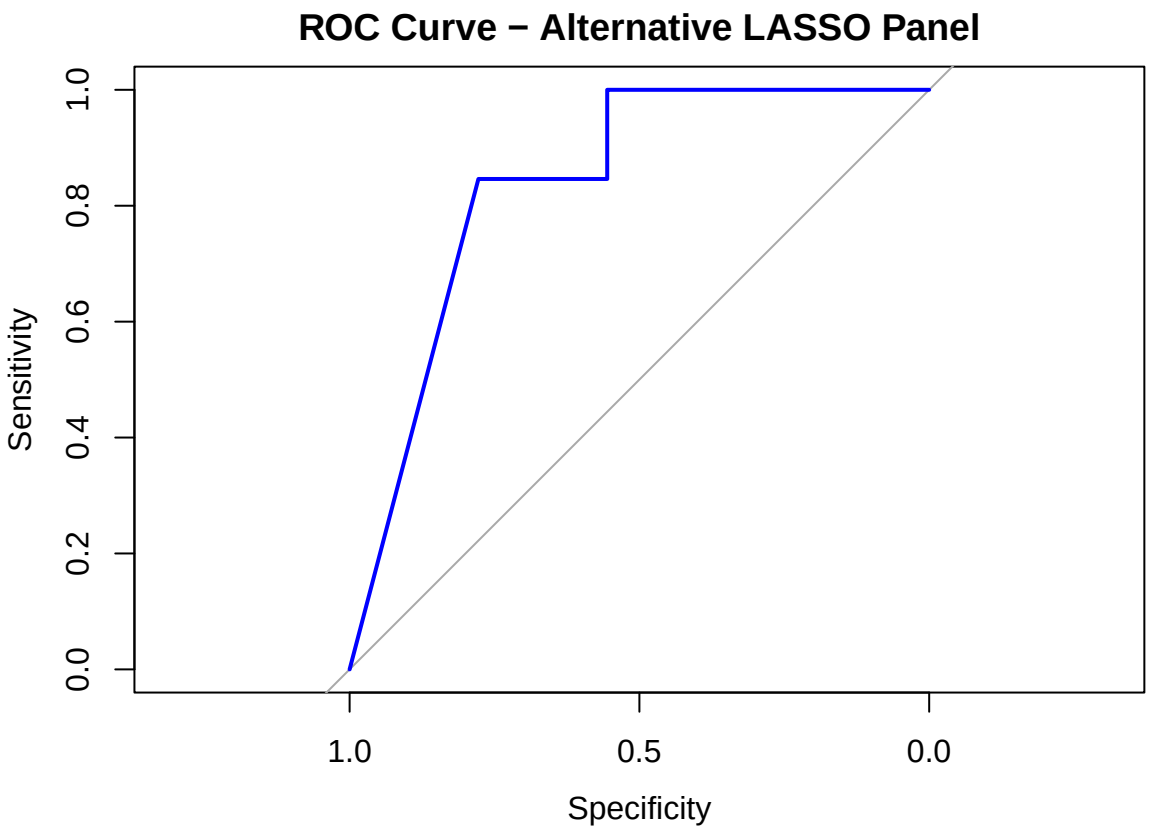
```
cat("Alternative LASSO panel AUROC:", round(auc_alt, 3), "\n")
```

```
## Alternative LASSO panel AUROC: 0.838
```

```
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