Module 4: Alternative approaches

2-step Semi-supervised Approach

i) Regress the surrogate on the features with penalized least square to get the direction of beta.

```
x <- all_x %>% select(starts_with("health") | starts_with('
S <- ehr_data$main_ICDNLP

# Step 1
beta.step1 <- adaptive_lasso_fit(
  y = S, # surrogate
  x = x, # all X
  family = "gaussian",
  tuning = "cv"
)

# Features selected
names(beta.step1[abs(beta.step1) > 0])[-1]
```

```
##   [1] "COD6"   "COD8"   "COD10"  "NLP5"   "NLP7"   "NLP1
##   [6] "NLP24"  "NLP28"  "NLP31"  "NLP33"  "NLP44"  "NLP
```

2-step Semi-supervised Approach

i) Regress the surrogate on the features with penalized least square to get the direction of beta.

(ii) Regress the outcome on the linear predictor to get the intercept and multiplier for the beta.

```
# linear predictor without intercept
bhatx <- linear_model_predict(beta = beta.step1, x = as.mat

# Step 2
step2 <- glm(train_y ~ bhatx[train_data$patient_id] + S[tra
  health_count[train_data$patient_id])
beta_step2 <- coef(step2)
beta_step2
```

```
##                          (Intercept)                 bhatx[train_d
##                           0.80766869
##              S[train_data$patient_id] health_count[train_d
##                           0.14334940
```

ROC

```
plot(roc(test_y, y_hat.lasso),
  print.auc = TRUE, main = "n_training = 106 (60%)"
)
plot(roc(test_y, y_hat.alasso),
  print.auc = TRUE, col = "red", add = TRUE, print.auc.y =
)
plot(roc(test_y, y_hat.ss),
  print.auc = TRUE, col = "green", add = TRUE, print.auc.y
)
plot(roc(test_y, y_hat.phecap),
  print.auc = TRUE, col = "blue", add = TRUE, print.auc.y =
)
legend(0, 0.3,
  legend = c("LASSO", "ALASSO", "PheCAP", "Two-Step"),
  col = c("black", "red", "blue", "green"),
  lty = 1, cex = 0.8
)
```

Model Evaluation

```
start <- Sys.time()
auc_twostep <- validate_ss(
  dat = labeled_data, nsim = 600,
  n.train = c(50, 70, 90),
  beta = beta.step1,
  S = S,
  x = x
)
end <- Sys.time()
end - start
```

```
## Time difference of 26.92371 secs
```

```
par(mfrow = c(1,3))
# Compare with Previous method
boxplot(cbind(auc_supervised, auc_phecap, auc_twostep)
%>% select(starts_with("n=50")),
ylim = c(0.5, 1), names = c("LASSO", "ALASSO","PheCAP", "Tw
```