Homework04

Contents


```
set.seed(231)
data = read.csv('data/titanic.csv')
data$survived = factor(data$survived, levels = c('Yes', 'No'))
data$pclass = factor(data$pclass)
```

Question 1

```
data_split = initial_split(data, prop = 0.70,
                           strata = survived)
titanic_train = training(data_split)
titanic_test = testing(data_split)
```

```
print(c(dim(titanic_train), dim(titanic_test)))
```

```
## [1] 623  12 268  12
```

Question 2

```
cv_folds <- vfold_cv(titanic_train, v = 10)
```

Question 3

Cross Validation is a methodology to find the best parameter of models and reduce prediction error.

At first, we divide the train set into 10 pieces(fold). The size of pieces are depend on the size of train set. This size indicates K. So this case is 10-fold cross validation. An one set of 10 folds is used as test set and the rest as a train. Then, make a model with 9 folds under the certain parameter value and predict the rest test folds. Repeat this for all remaining 9 folds. Calculate the mean MSE(or RMSE, accuracy, etc) under the certain parameter. We can draw a plot of parameters and scores and find the best parameter with the lowest MSE.

It is more effective to make a model with the parameters found in this way, rather than simply fit and test with the entire train set. Bootstrap could be used with the entire train set. This method does not divide train set as CV, change the order of observation in it and make a difference.

Question 4

```
#Recipe
titanic_recipe = recipe(survived ~ pclass + sex + age + sib_sp + parch + fare,
                        data = titanic_train) %>%
        step_impute_linear(age) %>%
        step_dummy(all_nominal_predictors()) %>%
        step_interact(terms = ~ starts_with("sex"):fare + age:fare)


#Logistic
log_reg = logistic_reg() %>%
        set_engine("glm") %>%
        set_mode("classification")

log_wkflow = workflow() %>%
        add_model(log_reg) %>%
        add_recipe(titanic_recipe)

log_fit = fit(log_wkflow, titanic_train)


#LDA
lda_mod = discrim_linear() %>%
        set_engine("MASS") %>%
        set_mode("classification")

lda_wkflow = workflow() %>%
        add_model(lda_mod) %>%
        add_recipe(titanic_recipe)

lda_fit = fit(lda_wkflow, titanic_train)


#QDA
qda_mod = discrim_quad() %>%
        set_mode("classification") %>%
        set_engine("MASS")

qda_wkflow = workflow() %>%
        add_model(qda_mod) %>%
        add_recipe(titanic_recipe)

qda_fit = fit(qda_wkflow, titanic_train)
```

3 models x 10 folds = 30 models

Question 5

```
log_res = log_wkflow %>%
        fit_resamples(resamples = cv_folds,
                      metrics = metric_set(recall, precision,
                                           accuracy, sens, spec, roc_auc),
                      control = control_resamples(save_pred = TRUE))
```

```r
lda_res = lda_wkflow %>%
        fit_resamples(resamples = cv_folds,
                      metrics = metric_set(recall, precision,
                                           accuracy, sens, spec, roc_auc),
                      control = control_resamples(save_pred = TRUE))

qda_res = qda_wkflow %>%
        fit_resamples(resamples = cv_folds,
                      metrics = metric_set(recall, precision,
                                           accuracy, sens, spec, roc_auc),
                      control = control_resamples(save_pred = TRUE))
```

Question 6

```r
collect_metrics(log_res)
```

```
## # A tibble: 6 x 6
##   .metric   .estimator  mean     n std_err .config
##   <chr>     <chr>      <dbl> <int>   <dbl> <chr>
## 1 accuracy  binary     0.804    10  0.0176 Preprocessor1_Model1
## 2 precision binary     0.782    10  0.0364 Preprocessor1_Model1
## 3 recall    binary     0.687    10  0.0266 Preprocessor1_Model1
## 4 roc_auc   binary     0.843    10  0.0195 Preprocessor1_Model1
## 5 sens      binary     0.687    10  0.0266 Preprocessor1_Model1
## 6 spec      binary     0.873    10  0.0242 Preprocessor1_Model1
```

```r
collect_metrics(lda_res)
```

```
## # A tibble: 6 x 6
##   .metric   .estimator  mean     n std_err .config
##   <chr>     <chr>      <dbl> <int>   <dbl> <chr>
## 1 accuracy  binary     0.794    10  0.0162 Preprocessor1_Model1
## 2 precision binary     0.769    10  0.0252 Preprocessor1_Model1
## 3 recall    binary     0.662    10  0.0303 Preprocessor1_Model1
## 4 roc_auc   binary     0.842    10  0.0198 Preprocessor1_Model1
## 5 sens      binary     0.662    10  0.0303 Preprocessor1_Model1
## 6 spec      binary     0.872    10  0.0168 Preprocessor1_Model1
```

```r
collect_metrics(qda_res)
```

```
## # A tibble: 6 x 6
##   .metric   .estimator  mean     n std_err .config
##   <chr>     <chr>      <dbl> <int>   <dbl> <chr>
## 1 accuracy  binary     0.785    10  0.0221 Preprocessor1_Model1
## 2 precision binary     0.858    10  0.0314 Preprocessor1_Model1
## 3 recall    binary     0.536    10  0.0293 Preprocessor1_Model1
## 4 roc_auc   binary     0.835    10  0.0205 Preprocessor1_Model1
## 5 sens      binary     0.536    10  0.0293 Preprocessor1_Model1
## 6 spec      binary     0.936    10  0.0164 Preprocessor1_Model1
```

The logistic model performed the best. The mean accuracy is 0.82 and std is 0.01. It indicates, the accuracy of 10 folds is all close to the mean.

Question 7

```
log_fit = fit(log_wkflow, titanic_train)
```

Question 8

```
predict(log_fit, new_data = titanic_test, type = "prob")
```

```
## # A tibble: 268 x 2
##    .pred_Yes .pred_No
##        <dbl>    <dbl>
##  1    0.113    0.887
##  2    0.139    0.861
##  3    0.901    0.0988
##  4    0.795    0.205
##  5    0.522    0.478
##  6    0.0919   0.908
##  7    0.247    0.753
##  8    0.618    0.382
##  9    0.269    0.731
## 10    0.619    0.381
## # ... with 258 more rows
```

```
log_acc = augment(log_fit, new_data = titanic_test) %>%
  accuracy(truth = survived, estimate = .pred_class)

log_acc
```

```
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy binary         0.821
```

Question 9

$Q = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \hat{y}_i = \beta$

$Q = \sum_{i=1}^{n}(y_i - \beta)^2$

$\leftrightarrow \frac{dQ}{d\beta} = -2\sum_{i=1}^{n}(y_i - \beta) = 0$

$\leftrightarrow \sum_{i=1}^{n} y_i - n\beta = 0$

$\leftrightarrow n\beta = \sum_{i=1}^{n} y_i$

$\leftrightarrow \hat{\beta} = \frac{\sum_{i=1}^{n} y_i}{n} = \bar{y}$

Question 10

In the LOOCV, we make models with the same data points n times. This data points are affecting models n times repeatedly. If there is an outlier, it will make all individual β values in models stand out in a similar direction. Therefore, a covariance between β occurs.

Of course, we don't need to have outliers in the data to explain this. This is just to explain how data points in LOOCV are not completely independent, so they would be in a model that does not make much difference.