PSTAT 131 Homework 4

Tammy Truong

2022-04-26

Question 1:

Splitting the data set
We split the data set and stratify on the variable `survived`.

```
set.seed(1004)
titanic_split <- initial_split(titanic, prop = 0.70, strata = survived)
titanic_train <- training(titanic_split)
titanic_test <- testing(titanic_split)
```

Verifying that the testing and training sets have the appropriate number of observations.

```
nrow(titanic_train)
```

```
## [1] 623
```

```
nrow(titanic_test)
```

```
## [1] 268
```

Question 2:

Folding the training set
We fold the training set and use k-fold cross validation with $k = 10$.

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

```
## #  10-fold cross-validation
## # A tibble: 10 x 2
##    splits           id
##    <list>           <chr>
##  1 <split [560/63]> Fold01
##  2 <split [560/63]> Fold02
##  3 <split [560/63]> Fold03
##  4 <split [561/62]> Fold04
##  5 <split [561/62]> Fold05
```

```
##  6 <split [561/62]> Fold06
##  7 <split [561/62]> Fold07
##  8 <split [561/62]> Fold08
##  9 <split [561/62]> Fold09
## 10 <split [561/62]> Fold10
```

Question 3:

K-fold cross-validation splits the data into k number of folds to assess model performance. It is used to select the best model for our dataset. We use k-fold cross-validation to compare the best value to our model, rather than simply fitting and testing models. If we did use the entire training set, resampling method is called the validation set approach.

Question 4:

Setting up workflows for 3 models We will be setting up workflows for the following 3 models:

- A logistic regression with the `glm` engine;
- A linear discriminant analysis with the `MASS` engine;
- A quadratic discriminant analysis with the `MASS` engine.

```
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) %>%
  step_interact(~ age:fare)
```

```
# Logistic Regression
log_reg <- logistic_reg() %>%
  set_engine("glm") %>%
  set_mode("classification")

log_wkflow <- workflow() %>%
  add_model(log_reg) %>%
  add_recipe(titanic_recipe)
```

```
# LDA
lda_mod <- discrim_linear() %>%
  set_mode("classification") %>%
  set_engine("MASS")

lda_wkflow <- workflow() %>%
  add_model(lda_mod) %>%
  add_recipe(titanic_recipe)
```

```r
# QDA
qda_mod <- discrim_quad() %>%
  set_mode("classification") %>%
  set_engine("MASS")

qda_wkflow <- workflow() %>%
  add_model(qda_mod) %>%
  add_recipe(titanic_recipe)
```

Question 5:

Fitting the models into the folded data

```r
log_fit <- log_wkflow %>%
  fit_resamples(titanic_folds)


lda_fit <- lda_wkflow %>%
  fit_resamples(titanic_folds)


qda_fit <- qda_wkflow %>%
  fit_resamples(titanic_folds)
```

Question 6:

We use `collect_metrics()` to print the mean and standard errors of the performance metric *accuracy* across all folds for each of the four models.

```r
collect_metrics(log_fit)
```

```
## # A tibble: 2 x 6
##   .metric  .estimator  mean     n std_err .config
##   <chr>    <chr>      <dbl> <int>   <dbl> <chr>
## 1 accuracy binary     0.811    10  0.0153 Preprocessor1_Model1
## 2 roc_auc  binary     0.845    10  0.0156 Preprocessor1_Model1
```

```r
collect_metrics(lda_fit)
```

```
## # A tibble: 2 x 6
##   .metric  .estimator  mean     n std_err .config
##   <chr>    <chr>      <dbl> <int>   <dbl> <chr>
## 1 accuracy binary     0.799    10  0.0181 Preprocessor1_Model1
## 2 roc_auc  binary     0.843    10  0.0156 Preprocessor1_Model1
```

```r
collect_metrics(qda_fit)
```

```
## # A tibble: 2 x 6
##   .metric  .estimator  mean     n std_err .config
##   <chr>    <chr>      <dbl> <int>   <dbl> <chr>
## 1 accuracy binary     0.790    10  0.0157 Preprocessor1_Model1
## 2 roc_auc  binary     0.844    10  0.0193 Preprocessor1_Model1
```

From above, we note that the best model is the logistic regression model. It has the lowest accuracy standard error at 0.01531636 and the highest mean accuracy at 0.8107015.

Question 7:

Now, we fit the Logistic Regression model to the entire training dataset.

```r
log_fit <- fit(log_wkflow, titanic_train)
```

Question 8:

Finally, we use `predict()`, `bind_cols()`, and `accuracy()` to assess the model's performance on the testing data.

```r
log_predict <- predict(log_fit, new_data = titanic_train, type = "prob")
log_predict <- bind_cols(log_predict, titanic_train %>% select(survived))

augment(log_fit, new_data = titanic_train) %>%
  accuracy(as.factor(survived), estimate = .pred_class)
```

```
## # A tibble: 1 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy binary         0.819
```

From the above results, we have that our model's testing accuracy is estimated to be 0.8186196. This is a high number because the average accuracy across folds was 0.810701, which shows that our model's testing accuracy is higher than the data from the 10 folds.