Class Activity 22

Your name here

May 14 2024

Group Activity 1

Load the `mlbench` package to get `PimaIndiansDiabetes2` dataset.

```
# Load the data - diabetes
data(PimaIndiansDiabetes2)
db <- PimaIndiansDiabetes2
db <- db %>% drop_na() %>% mutate(diabetes = fct_rev(factor(diabetes)))
db_raw <- db %>% select(glucose, insulin, diabetes)
```

a. Split the data `75-25` into training and test set using the following code.

```
set.seed(123)
```

b. Follow the steps to train a 7-NN classifier using the `tidymodels` toolkit

```
# define recipe and preprocess the data
db_recipe <-
Error: <text>:3:0: unexpected end of input
1: # define recipe and preprocess the data
2: db_recipe <-
  ^
```

```
# specify the model
db_knn_spec7 <-
Error: <text>:3:0: unexpected end of input
1: # specify the model
2: db_knn_spec7 <-
  ^
```

```
# define the workflow
db_workflow <-
Error: <text>:3:0: unexpected end of input
1: # define the workflow
2: db_workflow <-
  ^
```

```
# fit the model on the training data
db_fit <-
Error: <text>:3:0: unexpected end of input
1: # fit the model on the training data
2: db_fit <-
  ^
```

c. Classify the penguins in the `test` data frame.

```
test_features <- db_test %>% select(glucose, insulin)

# predict
db_pred <-


# combine the results
Error: <text>:9:0: unexpected end of input
7: # combine the results
8:
  ^
```

Group Activity 2

Calculate the accuracy, sensitivity, specificity, and positive predictive value (PPV) using the following confusion matrix.

```
conf_mat(db_results, truth = diabetes, estimate = predicted)
Error in eval(expr, envir, enclos): object 'db_results' not found
```

```
# your r-code
```

Extra: Code to recreate the plot in the `slides` for the `diabetes` dataset.

```
metrics_for_k <- function(k, db_train, db_test){
db_knn_spec <- nearest_neighbor(mode = "classification",
                            engine = "kknn",
                            weight_func = "rectangular",
                            neighbors = k)

db_knn_wkflow <- workflow() %>%
  add_recipe(db_recipe) %>%
  add_model(db_knn_spec)

db_knn_fit <- fit(db_knn_wkflow, data = db_train)
test_features <- db_test %>% select(glucose, insulin)
nn1_pred <- predict(db_knn_fit, test_features, type = "raw")

db_results <- db_test %>%
  select(diabetes) %>%
  bind_cols(predicted = nn1_pred)
custom_metrics <- metric_set(accuracy, sens, spec, ppv)

metrics <- custom_metrics(db_results,
               truth = diabetes,
               estimate = predicted)
metrics <- metrics %>% select(-.estimator) %>% mutate(k = rep(k,4))

return(list = metrics)
}
```

```r
# run the function and store it as "optim.results"




# plot

optim.results %>%
  ggplot(aes(x = , y = , color = ))) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  theme_minimal() +
  ggthemes::scale_color_wsj() +
  scale_x_continuous(breaks = k) +
  theme(panel.grid.minor.x = element_blank(),
        axis.text=element_text(size=6, angle = 20))+
  labs(color='Metric', y = "Estimate", x = "K")
Error: <text>:38:36: unexpected ')'
37: optim.results %>%
38:   ggplot(aes(x = , y = , color = )))
                                      ^
```