ML AI Assignment

Shubhang Periwal 19201104

4/19/2020

```
library(keras)
```

```
## Warning: package 'keras' was built under R version 3.6.3
```

```
tensorflow::tf$random$set_seed(0)
library(tfruns)
```

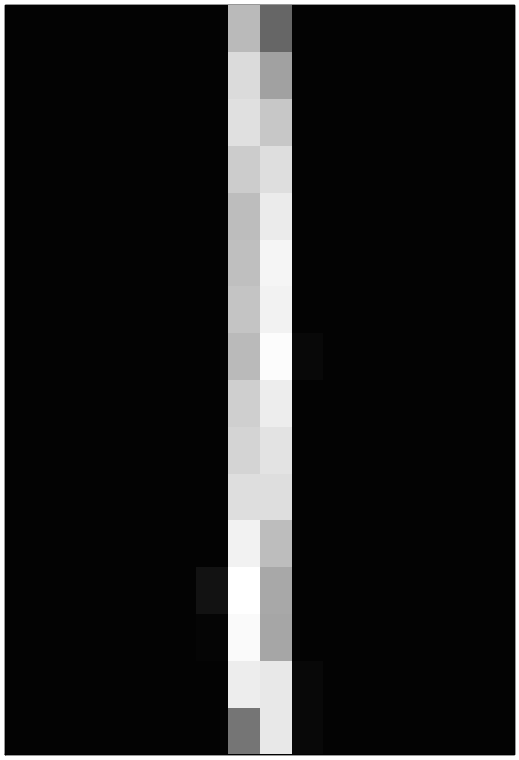
```
## Warning: package 'tfruns' was built under R version 3.6.3
```

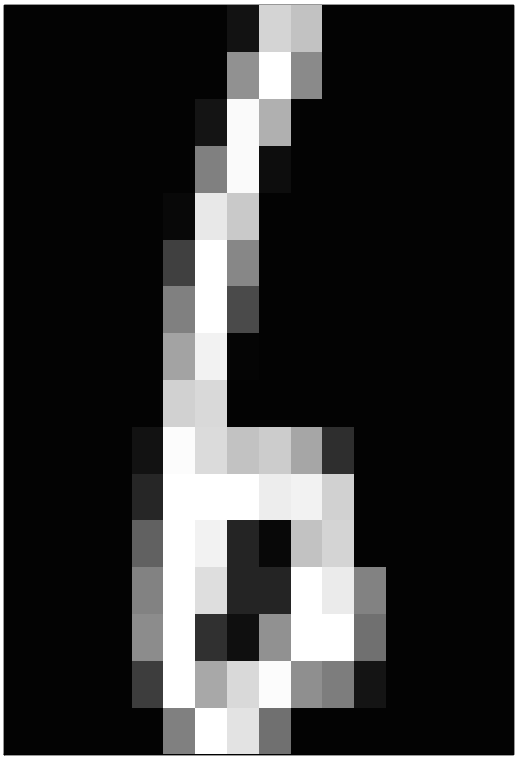
```
library(reticulate)
```

```
## Warning: package 'reticulate' was built under R version 3.6.3
```

```
library(jsonlite)
 load("data_usps_digits.RData") #loading the data into R
```

```
# helper function
# to plot the digit
plot_digit <- function(index, data) {
tmp <- (-data + 1) / 2 # to convert back to original
z <- matrix( data = as.numeric(data[index, 256:1]), 16, 16 )
image(z[16:1,1:16], col = gray((1:100)/100),
xaxt = "n", yaxt = "n")
}

# plot few example digits
par(mfrow = c(1,2), mar = rep(1.5, 4))
plot_digit(14, x_train)
plot_digit(900, x_train)
```





```
#preprocessing
# Convert y to categorical using one-hot encoding
y_train <- to_categorical(y_train, num_classes = 10)
y_test <- to_categorical(y_test , num_classes = 10)
# convert x_train and x_test from data frame to matrix for valid network input
x_train <- as.matrix(x_train)
x_test <- as.matrix(x_test)

# normalizen x(input) to 0-1
range_norm <- function(x, a = 0, b = 1) {
( (x - min(x)) / (max(x) - min(x)) )*(b - a) + a }

x_train <- apply(x_train, 2, range_norm)
x_test <- apply(x_test, 2, range_norm)
range(x_train)
```

```
## [1] 0 1
```

```
range(x_test)
```

```
## [1] 0 1
```

```
#coverting the x datasets to matrices:
x_train<-as.matrix(x_train)
x_test<-as.matrix(x_test)
```

```r
# split the test data in two halves: one for validation
# and the other for actual testing
val <- sample(1:nrow(x_test), 1000) # there are 10000 images in x_test
test <- setdiff(1:nrow(x_test), val)
x_val <- x_test[val,]
y_val <- y_test[val,]
x_test <- x_test[test,]
y_test <- y_test[test,]
# need these later
N <- nrow(x_train)
V <- ncol(x_train)
```

```r
#setting a grid of values for the flags/hyperparameters of interest:
hdlayer1 <- c(128,64,256)
dropout1 <- c(0,0.1,0.3)
hdlayer2 <- c(64,32)
dropout2 <- c(0,0.2)
hdlayer3 <- c(64,32,16,8)
dropout3 <- c(0,0.1,0.3)
```

```r
# run ----------------------------------------------------------------
runs <- tuning_run("assignment3config.R",
                     runs_dir = "runs_assignment",
                     flags = list(
                    hdlayer_1 = hdlayer1,
                    dropout_1 = dropout1,
                    hdlayer_2 = hdlayer2,
                    dropout_2 = dropout2,
                    hdlayer_3 = hdlayer3,
                    dropout_3 = dropout3
                       ),
                   sample = 0.2)
```

```
## 432 total combinations of flags (sampled to 86 combinations)

## Training run 1/86 (flags = list(256, 0, 32, 0, 8, 0.3))

## Using run directory runs_assignment/2020-04-20T08-34-15Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)
```

```
## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'

##
## Run completed: runs_assignment/2020-04-20T08-34-15Z

## Training run 2/86 (flags = list(256, 0.1, 32, 0.2, 32, 0.1))

## Using run directory runs_assignment/2020-04-20T08-34-57Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'

##
## Run completed: runs_assignment/2020-04-20T08-34-57Z

## Training run 3/86 (flags = list(64, 0, 64, 0, 8, 0))

## Using run directory runs_assignment/2020-04-20T08-35-25Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'
```

```
##
## Run completed: runs_assignment/2020-04-20T08-35-25Z

## Training run 4/86 (flags = list(128, 0.3, 64, 0.2, 32, 0.1))

## Using run directory runs_assignment/2020-04-20T08-36-09Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'

##
## Run completed: runs_assignment/2020-04-20T08-36-09Z

## Training run 5/86 (flags = list(256, 0, 32, 0, 8, 0))

## Using run directory runs_assignment/2020-04-20T08-36-37Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'

##
## Run completed: runs_assignment/2020-04-20T08-36-37Z
```

```
## Training run 6/86 (flags = list(64, 0, 32, 0, 8, 0))

## Using run directory runs_assignment/2020-04-20T08-37-02Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'

##
## Run completed: runs_assignment/2020-04-20T08-37-02Z

## Training run 7/86 (flags = list(128, 0.1, 32, 0, 64, 0.1))

## Using run directory runs_assignment/2020-04-20T08-37-30Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'

##
## Run completed: runs_assignment/2020-04-20T08-37-30Z

## Training run 8/86 (flags = list(128, 0, 64, 0.2, 16, 0.1))
```

```
## Using run directory runs_assignment/2020-04-20T08-38-03Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +     .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'

##
## Run completed: runs_assignment/2020-04-20T08-38-03Z

## Training run 9/86 (flags = list(256, 0.3, 32, 0, 64, 0.1))

## Using run directory runs_assignment/2020-04-20T08-38-34Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +     .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'

##
## Run completed: runs_assignment/2020-04-20T08-38-34Z

## Training run 10/86 (flags = list(64, 0, 64, 0.2, 32, 0.1))

## Using run directory runs_assignment/2020-04-20T08-39-11Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-39-11Z

## Training run 11/86 (flags = list(128, 0, 64, 0.2, 8, 0.1))

## Using run directory runs_assignment/2020-04-20T08-39-52Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-39-52Z

## Training run 12/86 (flags = list(256, 0.3, 32, 0.2, 16, 0.3))

## Using run directory runs_assignment/2020-04-20T08-40-40Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-40-40Z

## Training run 13/86 (flags = list(128, 0.1, 32, 0, 8, 0))

## Using run directory runs_assignment/2020-04-20T08-41-18Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-41-18Z

## Training run 14/86 (flags = list(256, 0.1, 64, 0, 16, 0))

## Using run directory runs_assignment/2020-04-20T08-42-09Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-42-09Z

## Training run 15/86 (flags = list(256, 0.3, 64, 0.2, 16, 0.1))

## Using run directory runs_assignment/2020-04-20T08-42-43Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-42-43Z

## Training run 16/86 (flags = list(128, 0.1, 64, 0.2, 8, 0.1))

## Using run directory runs_assignment/2020-04-20T08-43-12Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-43-12Z

## Training run 17/86 (flags = list(256, 0.1, 64, 0.2, 64, 0.1))

## Using run directory runs_assignment/2020-04-20T08-43-56Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-43-56Z

## Training run 18/86 (flags = list(256, 0.3, 32, 0, 32, 0))

## Using run directory runs_assignment/2020-04-20T08-44-33Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-44-33Z

## Training run 19/86 (flags = list(64, 0.1, 64, 0.2, 64, 0.3))

## Using run directory runs_assignment/2020-04-20T08-45-02Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-45-02Z

## Training run 20/86 (flags = list(64, 0.3, 64, 0.2, 16, 0.1))

## Using run directory runs_assignment/2020-04-20T08-45-52Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-45-52Z

## Training run 21/86 (flags = list(256, 0, 32, 0.2, 8, 0))

## Using run directory runs_assignment/2020-04-20T08-46-24Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-46-24Z

## Training run 22/86 (flags = list(256, 0.3, 64, 0, 16, 0.1))

## Using run directory runs_assignment/2020-04-20T08-46-53Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-46-53Z

## Training run 23/86 (flags = list(64, 0.1, 32, 0, 16, 0.1))

## Using run directory runs_assignment/2020-04-20T08-47-26Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-47-26Z

## Training run 24/86 (flags = list(256, 0.1, 64, 0, 64, 0.1))

## Using run directory runs_assignment/2020-04-20T08-47-55Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-47-55Z

## Training run 25/86 (flags = list(128, 0.3, 32, 0, 16, 0.1))

## Using run directory runs_assignment/2020-04-20T08-48-25Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-48-25Z

## Training run 26/86 (flags = list(64, 0.1, 64, 0.2, 8, 0))

## Using run directory runs_assignment/2020-04-20T08-48-51Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-48-51Z

## Training run 27/86 (flags = list(64, 0.1, 32, 0, 16, 0))

## Using run directory runs_assignment/2020-04-20T08-49-24Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-49-24Z

## Training run 28/86 (flags = list(256, 0.1, 32, 0.2, 8, 0))

## Using run directory runs_assignment/2020-04-20T08-49-53Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-49-53Z

## Training run 29/86 (flags = list(64, 0.3, 32, 0, 64, 0.1))

## Using run directory runs_assignment/2020-04-20T08-50-31Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-50-31Z

## Training run 30/86 (flags = list(64, 0.3, 32, 0.2, 16, 0))

## Using run directory runs_assignment/2020-04-20T08-50-59Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-50-59Z

## Training run 31/86 (flags = list(128, 0.3, 64, 0.2, 64, 0))

## Using run directory runs_assignment/2020-04-20T08-51-42Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-51-42Z

## Training run 32/86 (flags = list(64, 0.3, 64, 0, 8, 0.1))

## Using run directory runs_assignment/2020-04-20T08-52-21Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-52-21Z

## Training run 33/86 (flags = list(128, 0.1, 32, 0, 16, 0))

## Using run directory runs_assignment/2020-04-20T08-53-06Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-53-06Z

## Training run 34/86 (flags = list(128, 0, 32, 0, 64, 0.1))

## Using run directory runs_assignment/2020-04-20T08-54-00Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-54-00Z

## Training run 35/86 (flags = list(128, 0.1, 64, 0, 32, 0.3))

## Using run directory runs_assignment/2020-04-20T08-54-18Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-54-18Z

## Training run 36/86 (flags = list(64, 0.3, 32, 0, 64, 0))

## Using run directory runs_assignment/2020-04-20T08-54-43Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-54-43Z

## Training run 37/86 (flags = list(128, 0, 32, 0, 64, 0))

## Using run directory runs_assignment/2020-04-20T08-55-19Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-55-19Z

## Training run 38/86 (flags = list(256, 0.3, 32, 0, 8, 0.3))

## Using run directory runs_assignment/2020-04-20T08-55-47Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-55-47Z

## Training run 39/86 (flags = list(64, 0.1, 32, 0.2, 16, 0))

## Using run directory runs_assignment/2020-04-20T08-56-17Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-56-17Z

## Training run 40/86 (flags = list(64, 0, 32, 0, 32, 0.3))

## Using run directory runs_assignment/2020-04-20T08-57-13Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-57-13Z

## Training run 41/86 (flags = list(64, 0, 64, 0.2, 8, 0.1))

## Using run directory runs_assignment/2020-04-20T08-57-46Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-57-46Z

## Training run 42/86 (flags = list(64, 0.1, 32, 0.2, 64, 0))

## Using run directory runs_assignment/2020-04-20T08-58-10Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-58-10Z

## Training run 43/86 (flags = list(256, 0, 32, 0.2, 32, 0.1))

## Using run directory runs_assignment/2020-04-20T08-59-06Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-59-06Z

## Training run 44/86 (flags = list(64, 0.1, 32, 0.2, 8, 0))

## Using run directory runs_assignment/2020-04-20T08-59-30Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T08-59-30Z

## Training run 45/86 (flags = list(256, 0, 64, 0.2, 64, 0))

## Using run directory runs_assignment/2020-04-20T09-00-31Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-00-31Z

## Training run 46/86 (flags = list(128, 0.1, 64, 0.2, 8, 0))

## Using run directory runs_assignment/2020-04-20T09-01-18Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-01-18Z

## Training run 47/86 (flags = list(64, 0.3, 32, 0.2, 8, 0.3))

## Using run directory runs_assignment/2020-04-20T09-02-16Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-02-16Z

## Training run 48/86 (flags = list(128, 0.1, 32, 0, 8, 0.1))

## Using run directory runs_assignment/2020-04-20T09-03-02Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-03-02Z

## Training run 49/86 (flags = list(64, 0.1, 32, 0, 8, 0))

## Using run directory runs_assignment/2020-04-20T09-03-27Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-03-27Z

## Training run 50/86 (flags = list(64, 0, 64, 0, 64, 0))

## Using run directory runs_assignment/2020-04-20T09-04-02Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-04-02Z

## Training run 51/86 (flags = list(128, 0, 64, 0.2, 64, 0.3))

## Using run directory runs_assignment/2020-04-20T09-04-32Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-04-32Z

## Training run 52/86 (flags = list(128, 0, 64, 0, 64, 0))

## Using run directory runs_assignment/2020-04-20T09-05-23Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-05-23Z

## Training run 53/86 (flags = list(128, 0, 32, 0.2, 8, 0))

## Using run directory runs_assignment/2020-04-20T09-05-54Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-05-54Z

## Training run 54/86 (flags = list(64, 0.3, 64, 0, 32, 0.3))

## Using run directory runs_assignment/2020-04-20T09-06-30Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-06-30Z

## Training run 55/86 (flags = list(128, 0.1, 64, 0, 64, 0))

## Using run directory runs_assignment/2020-04-20T09-07-12Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-07-12Z

## Training run 56/86 (flags = list(128, 0.3, 64, 0, 32, 0))

## Using run directory runs_assignment/2020-04-20T09-07-40Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-07-40Z

## Training run 57/86 (flags = list(64, 0.1, 32, 0, 64, 0.1))

## Using run directory runs_assignment/2020-04-20T09-08-07Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-08-07Z

## Training run 58/86 (flags = list(128, 0.1, 64, 0.2, 8, 0.3))

## Using run directory runs_assignment/2020-04-20T09-08-50Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-08-50Z

## Training run 59/86 (flags = list(128, 0.3, 32, 0.2, 32, 0.1))

## Using run directory runs_assignment/2020-04-20T09-09-54Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-09-54Z

## Training run 60/86 (flags = list(128, 0, 64, 0, 32, 0.1))

## Using run directory runs_assignment/2020-04-20T09-10-26Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-10-26Z

## Training run 61/86 (flags = list(64, 0.3, 32, 0.2, 64, 0.1))

## Using run directory runs_assignment/2020-04-20T09-10-53Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-10-53Z

## Training run 62/86 (flags = list(128, 0.1, 32, 0, 32, 0.3))

## Using run directory runs_assignment/2020-04-20T09-11-27Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-11-27Z

## Training run 63/86 (flags = list(128, 0.3, 32, 0.2, 32, 0))

## Using run directory runs_assignment/2020-04-20T09-12-02Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-12-02Z

## Training run 64/86 (flags = list(64, 0.1, 64, 0.2, 8, 0.3))

## Using run directory runs_assignment/2020-04-20T09-12-57Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-12-57Z

## Training run 65/86 (flags = list(256, 0.1, 64, 0, 32, 0.1))

## Using run directory runs_assignment/2020-04-20T09-14-01Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-14-01Z

## Training run 66/86 (flags = list(128, 0.1, 64, 0.2, 16, 0.1))

## Using run directory runs_assignment/2020-04-20T09-14-29Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-14-29Z

## Training run 67/86 (flags = list(128, 0.3, 64, 0, 8, 0.3))

## Using run directory runs_assignment/2020-04-20T09-15-01Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-15-01Z

## Training run 68/86 (flags = list(64, 0.3, 32, 0, 64, 0.3))

## Using run directory runs_assignment/2020-04-20T09-15-35Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-15-35Z

## Training run 69/86 (flags = list(128, 0.3, 32, 0.2, 8, 0))

## Using run directory runs_assignment/2020-04-20T09-16-23Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-16-23Z

## Training run 70/86 (flags = list(64, 0.3, 64, 0, 64, 0.3))

## Using run directory runs_assignment/2020-04-20T09-16-57Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-16-57Z

## Training run 71/86 (flags = list(256, 0, 32, 0.2, 64, 0.3))

## Using run directory runs_assignment/2020-04-20T09-17-21Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-17-21Z

## Training run 72/86 (flags = list(128, 0, 32, 0, 32, 0))

## Using run directory runs_assignment/2020-04-20T09-17-53Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-17-53Z

## Training run 73/86 (flags = list(256, 0, 32, 0, 32, 0.3))

## Using run directory runs_assignment/2020-04-20T09-18-16Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-18-16Z

## Training run 74/86 (flags = list(64, 0.3, 32, 0, 8, 0))

## Using run directory runs_assignment/2020-04-20T09-18-42Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-18-42Z

## Training run 75/86 (flags = list(64, 0.3, 32, 0, 32, 0.1))

## Using run directory runs_assignment/2020-04-20T09-19-35Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-19-35Z

## Training run 76/86 (flags = list(256, 0.3, 32, 0, 64, 0))

## Using run directory runs_assignment/2020-04-20T09-20-36Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-20-36Z

## Training run 77/86 (flags = list(256, 0.1, 64, 0.2, 64, 0.3))

## Using run directory runs_assignment/2020-04-20T09-21-13Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-21-13Z

## Training run 78/86 (flags = list(128, 0, 32, 0.2, 32, 0.3))

## Using run directory runs_assignment/2020-04-20T09-21-51Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-21-51Z

## Training run 79/86 (flags = list(128, 0.3, 32, 0.2, 16, 0.1))

## Using run directory runs_assignment/2020-04-20T09-22-41Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-22-41Z

## Training run 80/86 (flags = list(256, 0.3, 64, 0, 8, 0.1))

## Using run directory runs_assignment/2020-04-20T09-23-16Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-23-16Z

## Training run 81/86 (flags = list(64, 0.1, 32, 0, 64, 0))

## Using run directory runs_assignment/2020-04-20T09-23-47Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-23-47Z

## Training run 82/86 (flags = list(64, 0.1, 32, 0.2, 16, 0.3))

## Using run directory runs_assignment/2020-04-20T09-24-38Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-24-38Z

## Training run 83/86 (flags = list(64, 0.3, 64, 0, 16, 0.3))

## Using run directory runs_assignment/2020-04-20T09-25-24Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-25-24Z

## Training run 84/86 (flags = list(64, 0.3, 64, 0.2, 64, 0.1))

## Using run directory runs_assignment/2020-04-20T09-25-59Z
```

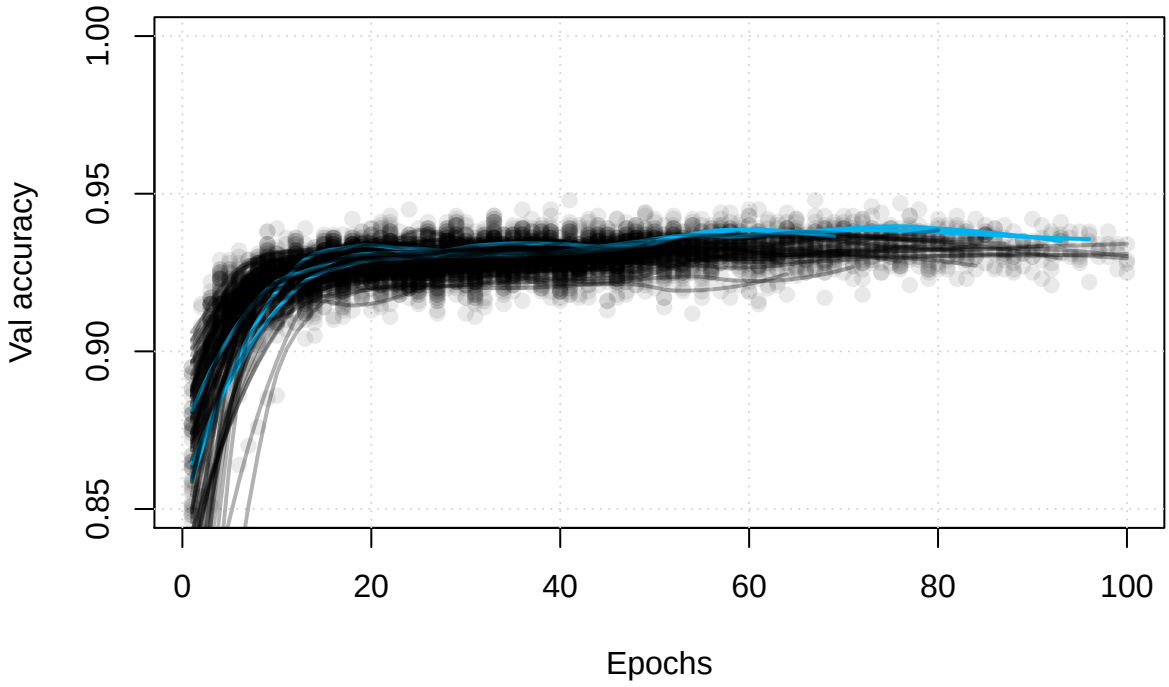
```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-25-59Z

## Training run 85/86 (flags = list(64, 0, 32, 0, 16, 0.1))

## Using run directory runs_assignment/2020-04-20T09-26-56Z

##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-26-56Z

## Training run 86/86 (flags = list(256, 0.1, 64, 0.2, 8, 0.3))

## Using run directory runs_assignment/2020-04-20T09-27-27Z
```

```
##
## > library(keras)
##
## > FLAGS <- flags(flag_integer("hdlayer_1", 256), flag_numeric("dropout_1",
## +     0.2), flag_integer("hdlayer_2", 64), flag_numeric("dropout_2",
## +    .... [TRUNCATED]
##
## > model <- keras_model_sequential() %>% layer_dense(units = FLAGS$hdlayer_1,
## +     input_shape = ncol(x_train), activation = "relu", name = "layer_1" .... [TRUNCATED]
##
## > fit <- model %>% fit(x = x_train, y = y_train, validation_data = list(x_val,
## +     y_val), epochs = 100, batch_size = 64, verbose = 1, callbacks =  .... [TRUNCATED]
##
## > score <- model %>% evaluate(x_test, y_test, verbose = 0)

## Warning in value[[3L]](cond): Error occurred resetting tf graph:
## AttributeError: module 'tensorflow' has no attribute 'reset_default_graph'


##
## Run completed: runs_assignment/2020-04-20T09-27-27Z
```

```r
#Determing the optimal configuration for the data
#Extracting values from the stored runs

read_metrics <- function(path, files =NULL)
{
path <- paste0(path, "/")
if(is.null(files)) files <- list.files(path)
n <- length(files)
out <- vector("list", n)
for(i in 1:n) {
dir <- paste0(path, files[i], "/tfruns.d/")
out[[i]] <- jsonlite::fromJSON(paste0(dir, "metrics.json"))
out[[i]]$flags <- jsonlite::fromJSON(paste0(dir, "flags.json"))
out[[i]]$evaluation <- jsonlite::fromJSON(paste0(dir,"evaluation.json"))
}
return(out)
}
#Plotting the corresponding validation learning curves
plot_learning_curve <- function(x, ylab = NULL, cols = NULL, top = 3,
span = 0.4, ...)
{
smooth_line <- function(y) {
x <- 1:length(y)
out <- predict(loess(y~x, span = span))
return(out)
}
matplot(x, ylab = ylab, xlab = "Epochs", type = "n", ...)
grid()
matplot(x, pch = 19, col = adjustcolor(cols, 0.3), add = TRUE)
tmp <- apply(x, 2, smooth_line)
tmp <- sapply(tmp, "length<-", max(lengths(tmp)))
set <- order(apply(tmp, 2, max, na.rm = TRUE), decreasing = TRUE)[1:top]
```

```
cl <- rep(cols, ncol(tmp))
cl[set] <- "deepskyblue2"
matlines(tmp, lty = 1, col = cl, lwd = 2)
}
```

```
# extract results
out <- read_metrics("runs_assignment")
# extract validation accuracy and plot learning curve
acc <- sapply(out, "[[", "val_accuracy")
plot_learning_curve(acc, col = adjustcolor("black", 0.3), ylim = c(0.85, 1),ylab = "Val accuracy", top =
```




```
res1<- ls_runs(metric_val_accuracy > 0.87, runs_dir = "runs_assignment", order = metric_val_accuracy)
res1
```

```
## Data frame: 86 x 30
##                                run_dir metric_val_accuracy eval_loss
## 1  runs_assignment/2020-04-20T09-21-51Z               0.942    0.3163
## 2  runs_assignment/2020-04-20T09-20-36Z               0.940    0.2725
## 3  runs_assignment/2020-04-20T09-18-16Z               0.940    0.3158
## 4  runs_assignment/2020-04-20T08-59-06Z               0.940    0.3105
## 5  runs_assignment/2020-04-20T08-45-02Z               0.940    0.2809
## 6  runs_assignment/2020-04-20T08-46-53Z               0.938    0.3047
## 7  runs_assignment/2020-04-20T08-38-34Z               0.938    0.2641
## 8  runs_assignment/2020-04-20T09-08-07Z               0.937    0.2785
## 9  runs_assignment/2020-04-20T09-01-18Z               0.937    0.2956
```

```
## 10 runs_assignment/2020-04-20T08-58-10Z              0.937    0.2751
##    eval_accuracy metric_loss metric_accuracy metric_val_loss
## 1         0.9533      0.1198          0.9927          0.3914
## 2         0.9543      0.1190          0.9915          0.3344
## 3         0.9444      0.1368          0.9930          0.3645
## 4         0.9474      0.1448          0.9888          0.3700
## 5         0.9464      0.1074          0.9905          0.3318
## 6         0.9494      0.1667          0.9840          0.3599
## 7         0.9563      0.1247          0.9893          0.3421
## 8         0.9464      0.0961          0.9941          0.3497
## 9         0.9454      0.1189          0.9914          0.3476
## 10        0.9513      0.1036          0.9918          0.3462
## # ... with 76 more rows
## # ... with 23 more columns:
## #   flag_hdlayer_1, flag_dropout_1, flag_hdlayer_2, flag_dropout_2,
## #   flag_hdlayer_3, flag_dropout_3, samples, batch_size, epochs,
## #   epochs_completed, metrics, model, loss_function, optimizer,
## #   learning_rate, script, start, end, completed, output, source_code,
## #   context, type
```

```
res1 <- res1[,c(2,4,8:11)]
res1[1:10,]
```

```
## Data frame: 10 x 6
##    metric_val_accuracy eval_accuracy flag_hdlayer_1 flag_dropout_1
## 1                0.942        0.9533            128            0.0
## 2                0.940        0.9543            256            0.3
## 3                0.940        0.9444            256            0.0
## 4                0.940        0.9474            256            0.0
## 5                0.940        0.9464             64            0.1
## 6                0.938        0.9494            256            0.3
## 7                0.938        0.9563            256            0.3
## 8                0.937        0.9464             64            0.1
## 9                0.937        0.9454            128            0.1
## 10               0.937        0.9513             64            0.1
##    flag_hdlayer_2 flag_dropout_2
## 1              32            0.2
## 2              32            0.0
## 3              32            0.0
## 4              32            0.2
## 5              64            0.2
## 6              64            0.0
## 7              32            0.0
## 8              32            0.0
## 9              64            0.2
## 10             32            0.2
```