Homework4

Steven Black

3/3/2019

IS 677: Introduction to Data Science Spring 2019

Homework Assignment 4 (Due: March 10, 2019, midnight EST)

Use the MNIST data that comes with Keras to answer the questions below.

Install keras

```
# install.packages("keras", "reticulate")
library(keras)
# install_keras()
```

Get MINST data

```
mnist <- dataset_mnist()
train_images <- mnist$train$x
train_labels <- mnist$train$y
test_images <- mnist$test$x
test_labels <- mnist$test$y
```

Shape MINST for training

```
train_images_reshaped <- array_reshape(train_images, c(60000, 28 * 28))
train_images_reshaped <- train_images_reshaped / 255

test_images_reshaped <- array_reshape(test_images, c(10000, 28 * 28))
test_images_reshaped <- test_images_reshaped / 255

train_labels_categorical <- to_categorical(train_labels)
test_labels_categorical <- to_categorical(test_labels)
```

1. Plot image #50 in the training set. (10 points)

```
plot(as.raster(train_images[50,,], max = 255))
```

2. Plot Image #50 in the test set. (10 points)

```
plot(as.raster(test_images[50,,], max = 255))
```

3. How many 6's are there in the training images? (10 points)

```
# easy solution
table(train_labels)[7]
```

```
##    6
## 5918
```

```
network <- keras_model_sequential() %>%
  layer_dense(units = 512, activation = "relu", input_shape = c(28 * 28)) %>%
  layer_dense(units = 10, activation = "softmax")

network %>% compile(
  optimizer = "rmsprop",
```

```
  loss = "categorical_crossentropy",
  metrics = c("accuracy")
)

network %>% fit(train_images_reshaped, train_labels_categorical, epochs = 5, batch_size = 128)

network %>% evaluate(test_images_reshaped, test_labels_categorical)
```

```
## $loss
## [1] 0.06615047
##
## $acc
## [1] 0.9801
```

```
train_image_predictions <- network %>% predict_classes(train_images_reshaped)
table(train_image_predictions)[7]
```

```
##    6
## 5888
```

4. How many 6's are there in the test images? (10 points)

```
# easy answer
table(test_labels)[7]
```

```
##   6
## 958
```

```
test_image_predictions <- network %>% predict_classes(test_images_reshaped)
table(test_image_predictions)[7]
```

```
##   6
## 944
```

5. Add a second dense hidden layer identical to the first layer and retrain the network. (40 points)

```
network <- keras_model_sequential() %>%
  layer_dense(units = 512, activation = "relu", input_shape = c(28 * 28)) %>%
  layer_dense(units = 512, activation = "relu", input_shape = c(28 * 28)) %>%
  layer_dense(units = 10, activation = "softmax")

network %>% compile(
  optimizer = "rmsprop",
  loss = "categorical_crossentropy",
  metrics = c("accuracy")
)

network %>% fit(train_images_reshaped, train_labels_categorical, epochs = 5, batch_size = 128)

network %>% evaluate(test_images_reshaped, test_labels_categorical)
```

```
## $loss
```

```
## [1] 0.08209576
##
## $acc
## [1] 0.9807
```

6. What benefits did you notice in adding an additional dense layer? (10 points)

Accuracy increased from `0.9802` to `0.9821`.

7. What costs did you notice in adding the additional dense layer? (10 points)

The loss increased from `0.06629773` to `0.07555452`.