Assignment 6_2

April 23, 2023

Gabriel Avinaz

Week 6

4/17/23

0.1 Assignment 6.2

0.1.1 Assignment 6.2.a

[ ]:
```
from keras.datasets import cifar10
from keras import layers, models, optimizers
from keras.preprocessing.image import ImageDataGenerator
from keras.utils import to_categorical
from keras.callbacks import CSVLogger
import matplotlib.pyplot as plt
```

[ ]:
```
(x_train, y_train), (x_test, y_test) = cifar10.load_data()
print(x_train.shape == (50000, 32, 32, 3))
print(x_test.shape == (10000, 32, 32, 3))
print(y_train.shape == (50000, 1))
print(y_test.shape == (10000, 1))

num_classes = 10
y_train = to_categorical(y_train, num_classes)
y_test = to_categorical(y_test, num_classes)


x_train = x_train / 255
x_test = x_test / 255
```

```
True
True
True
True
```

[ ]:
```
model = models.Sequential()

# example of a 3-block vgg style architecture
```

```
model.add(layers.Conv2D(32, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same', input_shape=(32, 32, 3)))
model.add(layers.Conv2D(32, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same'))
model.add(layers.Conv2D(64, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(128, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same'))
model.add(layers.Conv2D(128, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Flatten())
model.add(layers.Dense(128, activation='relu', kernel_initializer='he_uniform'))
model.add(layers.Dense(10, activation='softmax'))

model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 32, 32, 32)        896

 conv2d_1 (Conv2D)           (None, 32, 32, 32)        9248

 max_pooling2d (MaxPooling2D  (None, 16, 16, 32)       0
 )

 conv2d_2 (Conv2D)           (None, 16, 16, 64)        18496

 conv2d_3 (Conv2D)           (None, 16, 16, 64)        36928

 max_pooling2d_1 (MaxPooling  (None, 8, 8, 64)         0
 2D)

 conv2d_4 (Conv2D)           (None, 8, 8, 128)         73856

 conv2d_5 (Conv2D)           (None, 8, 8, 128)         147584

 max_pooling2d_2 (MaxPooling  (None, 4, 4, 128)        0
 2D)

 flatten (Flatten)           (None, 2048)              0
```

```
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 conv2d (Conv2D)              (None, 32, 32, 32)        896

 conv2d_1 (Conv2D)            (None, 32, 32, 32)        9248

 max_pooling2d (MaxPooling2D  (None, 16, 16, 32)        0
 )

 conv2d_2 (Conv2D)            (None, 16, 16, 64)        18496

 conv2d_3 (Conv2D)            (None, 16, 16, 64)        36928

 max_pooling2d_1 (MaxPooling  (None, 8, 8, 64)          0
 2D)

 conv2d_4 (Conv2D)            (None, 8, 8, 128)         73856

 conv2d_5 (Conv2D)            (None, 8, 8, 128)         147584

 max_pooling2d_2 (MaxPooling  (None, 4, 4, 128)         0
 2D)

 flatten (Flatten)            (None, 2048)              0

 dense (Dense)                (None, 128)               262272

 dense_1 (Dense)              (None, 10)                1290

=================================================================
Total params: 550,570
Trainable params: 550,570
Non-trainable params: 0
_________________________________________________________________
```

[ ]:
```
model.compile(loss='categorical_crossentropy', optimizer=optimizers.
↪RMSprop(learning_rate=0.001), metrics=['acc'])
```

[ ]:
```
train_datagen = ImageDataGenerator()
test_datagen = ImageDataGenerator()

train_datagen.fit(x_train)
test_datagen.fit(x_test)

train_generator = train_datagen.flow(x_train, y_train, batch_size=64)
```

```
validation_generator = test_datagen.flow(x_test, y_test, batch_size=64)

for data_batch, labels_batch in train_generator:
    print('data batch shape:', data_batch.shape)
    print('labels batch shape:', labels_batch.shape)
    break
```

```
data batch shape: (64, 32, 32, 3)
labels batch shape: (64, 10)
```

[ ]:
```
csv_logger = CSVLogger('results/cifar10_model.log')
history = model.fit(train_generator, steps_per_epoch=64, epochs=120,␣
 ↪validation_data=(validation_generator), validation_steps=64,␣
 ↪callbacks=csv_logger)
```

```
Epoch 1/120
64/64 [==============================] - 5s 62ms/step - loss: 2.2633 - acc:
0.2068 - val_loss: 2.0676 - val_acc: 0.2383
Epoch 2/120
64/64 [==============================] - 5s 85ms/step - loss: 1.8967 - acc:
0.3088 - val_loss: 1.6619 - val_acc: 0.3855
Epoch 3/120
64/64 [==============================] - 4s 65ms/step - loss: 1.6811 - acc:
0.3904 - val_loss: 1.5661 - val_acc: 0.4514
Epoch 4/120
64/64 [==============================] - 5s 78ms/step - loss: 1.6069 - acc:
0.4214 - val_loss: 1.6355 - val_acc: 0.4133
Epoch 5/120
64/64 [==============================] - 5s 84ms/step - loss: 1.5241 - acc:
0.4558 - val_loss: 1.6828 - val_acc: 0.4082
Epoch 6/120
64/64 [==============================] - 6s 88ms/step - loss: 1.4715 - acc:
0.4780 - val_loss: 1.3736 - val_acc: 0.5110
Epoch 7/120
64/64 [==============================] - 5s 78ms/step - loss: 1.3625 - acc:
0.5122 - val_loss: 1.3298 - val_acc: 0.5171
Epoch 8/120
64/64 [==============================] - 6s 93ms/step - loss: 1.3181 - acc:
0.5266 - val_loss: 1.2369 - val_acc: 0.5522
Epoch 9/120
64/64 [==============================] - 5s 81ms/step - loss: 1.2588 - acc:
0.5481 - val_loss: 1.3541 - val_acc: 0.5085
Epoch 10/120
64/64 [==============================] - 5s 70ms/step - loss: 1.2080 - acc:
0.5652 - val_loss: 1.2554 - val_acc: 0.5461
Epoch 11/120
64/64 [==============================] - 5s 72ms/step - loss: 1.1851 - acc:
```

```
0.5784 - val_loss: 1.2337 - val_acc: 0.5642
Epoch 12/120
64/64 [==============================] - 6s 96ms/step - loss: 1.1369 - acc:
0.6011 - val_loss: 1.2068 - val_acc: 0.5776
Epoch 13/120
64/64 [==============================] - 6s 96ms/step - loss: 1.0900 - acc:
0.6165 - val_loss: 1.1316 - val_acc: 0.5955
Epoch 14/120
64/64 [==============================] - 6s 90ms/step - loss: 1.0407 - acc:
0.6250 - val_loss: 1.0655 - val_acc: 0.6201
Epoch 15/120
64/64 [==============================] - 6s 88ms/step - loss: 1.0302 - acc:
0.6316 - val_loss: 1.0537 - val_acc: 0.6313
Epoch 16/120
64/64 [==============================] - 6s 94ms/step - loss: 0.9975 - acc:
0.6528 - val_loss: 0.9714 - val_acc: 0.6487
Epoch 17/120
64/64 [==============================] - 6s 96ms/step - loss: 0.9723 - acc:
0.6597 - val_loss: 1.1227 - val_acc: 0.6079
Epoch 18/120
64/64 [==============================] - 5s 86ms/step - loss: 0.9009 - acc:
0.6897 - val_loss: 1.0961 - val_acc: 0.6147
Epoch 19/120
64/64 [==============================] - 7s 115ms/step - loss: 0.9282 - acc:
0.6689 - val_loss: 1.0917 - val_acc: 0.6177
Epoch 20/120
64/64 [==============================] - 6s 95ms/step - loss: 0.8788 - acc:
0.7013 - val_loss: 0.9496 - val_acc: 0.6709
Epoch 21/120
64/64 [==============================] - 7s 107ms/step - loss: 0.8451 - acc:
0.7036 - val_loss: 1.0012 - val_acc: 0.6658
Epoch 22/120
64/64 [==============================] - 6s 95ms/step - loss: 0.8690 - acc:
0.6982 - val_loss: 1.0061 - val_acc: 0.6418
Epoch 23/120
64/64 [==============================] - 7s 107ms/step - loss: 0.8438 - acc:
0.7153 - val_loss: 0.9056 - val_acc: 0.6897
Epoch 24/120
64/64 [==============================] - 7s 105ms/step - loss: 0.8266 - acc:
0.7136 - val_loss: 1.0209 - val_acc: 0.6543
Epoch 25/120
64/64 [==============================] - 6s 98ms/step - loss: 0.7874 - acc:
0.7278 - val_loss: 1.0455 - val_acc: 0.6396
Epoch 26/120
64/64 [==============================] - 7s 114ms/step - loss: 0.7886 - acc:
0.7354 - val_loss: 0.8841 - val_acc: 0.6975
Epoch 27/120
64/64 [==============================] - 6s 93ms/step - loss: 0.7789 - acc:
```

```
0.7268 - val_loss: 0.8590 - val_acc: 0.7056
Epoch 28/120
64/64 [==============================] - 6s 93ms/step - loss: 0.7564 - acc:
0.7402 - val_loss: 1.0212 - val_acc: 0.6548
Epoch 29/120
64/64 [==============================] - 6s 100ms/step - loss: 0.7442 - acc:
0.7400 - val_loss: 0.8220 - val_acc: 0.7136
Epoch 30/120
64/64 [==============================] - 6s 93ms/step - loss: 0.7185 - acc:
0.7520 - val_loss: 0.8292 - val_acc: 0.7161
Epoch 31/120
64/64 [==============================] - 5s 85ms/step - loss: 0.6924 - acc:
0.7627 - val_loss: 0.8430 - val_acc: 0.7073
Epoch 32/120
64/64 [==============================] - 6s 91ms/step - loss: 0.7074 - acc:
0.7578 - val_loss: 0.8922 - val_acc: 0.6956
Epoch 33/120
64/64 [==============================] - 6s 86ms/step - loss: 0.6929 - acc:
0.7585 - val_loss: 0.8787 - val_acc: 0.7007
Epoch 34/120
64/64 [==============================] - 5s 84ms/step - loss: 0.6749 - acc:
0.7688 - val_loss: 0.8674 - val_acc: 0.7080
Epoch 35/120
64/64 [==============================] - 5s 81ms/step - loss: 0.6439 - acc:
0.7769 - val_loss: 0.8420 - val_acc: 0.7161
Epoch 36/120
64/64 [==============================] - 6s 87ms/step - loss: 0.6213 - acc:
0.7817 - val_loss: 0.8175 - val_acc: 0.7327
Epoch 37/120
64/64 [==============================] - 5s 83ms/step - loss: 0.6173 - acc:
0.7896 - val_loss: 0.8537 - val_acc: 0.7195
Epoch 38/120
64/64 [==============================] - 5s 78ms/step - loss: 0.6144 - acc:
0.7848 - val_loss: 0.8500 - val_acc: 0.7163
Epoch 39/120
64/64 [==============================] - 6s 91ms/step - loss: 0.5935 - acc:
0.7891 - val_loss: 0.8733 - val_acc: 0.7173
Epoch 40/120
64/64 [==============================] - 5s 83ms/step - loss: 0.5662 - acc:
0.8066 - val_loss: 0.9193 - val_acc: 0.7031
Epoch 41/120
64/64 [==============================] - 5s 80ms/step - loss: 0.5756 - acc:
0.8031 - val_loss: 0.9028 - val_acc: 0.7007
Epoch 42/120
64/64 [==============================] - 5s 84ms/step - loss: 0.5718 - acc:
0.8030 - val_loss: 0.8406 - val_acc: 0.7305
Epoch 43/120
64/64 [==============================] - 5s 80ms/step - loss: 0.5586 - acc:
```

```
0.8132 - val_loss: 0.8409 - val_acc: 0.7236
Epoch 44/120
64/64 [==============================] - 5s 78ms/step - loss: 0.5095 - acc:
0.8293 - val_loss: 0.9236 - val_acc: 0.7180
Epoch 45/120
64/64 [==============================] - 5s 81ms/step - loss: 0.5589 - acc:
0.8069 - val_loss: 0.8715 - val_acc: 0.7197
Epoch 46/120
64/64 [==============================] - 6s 87ms/step - loss: 0.5504 - acc:
0.8169 - val_loss: 0.7935 - val_acc: 0.7361
Epoch 47/120
64/64 [==============================] - 6s 88ms/step - loss: 0.5188 - acc:
0.8242 - val_loss: 0.8025 - val_acc: 0.7380
Epoch 48/120
64/64 [==============================] - 5s 86ms/step - loss: 0.4962 - acc:
0.8362 - val_loss: 0.8194 - val_acc: 0.7349
Epoch 49/120
64/64 [==============================] - 5s 86ms/step - loss: 0.5195 - acc:
0.8242 - val_loss: 0.8045 - val_acc: 0.7395
Epoch 50/120
64/64 [==============================] - 5s 84ms/step - loss: 0.4854 - acc:
0.8333 - val_loss: 0.8922 - val_acc: 0.7339
Epoch 51/120
64/64 [==============================] - 5s 81ms/step - loss: 0.4817 - acc:
0.8291 - val_loss: 0.9027 - val_acc: 0.7173
Epoch 52/120
64/64 [==============================] - 5s 83ms/step - loss: 0.4812 - acc:
0.8384 - val_loss: 0.8365 - val_acc: 0.7468
Epoch 53/120
64/64 [==============================] - 6s 87ms/step - loss: 0.4686 - acc:
0.8433 - val_loss: 0.8419 - val_acc: 0.7427
Epoch 54/120
64/64 [==============================] - 5s 83ms/step - loss: 0.4901 - acc:
0.8340 - val_loss: 0.8311 - val_acc: 0.7397
Epoch 55/120
64/64 [==============================] - 5s 85ms/step - loss: 0.4609 - acc:
0.8491 - val_loss: 0.8933 - val_acc: 0.7202
Epoch 56/120
64/64 [==============================] - 5s 81ms/step - loss: 0.4691 - acc:
0.8406 - val_loss: 0.8474 - val_acc: 0.7329
Epoch 57/120
64/64 [==============================] - 5s 84ms/step - loss: 0.4545 - acc:
0.8442 - val_loss: 0.8345 - val_acc: 0.7471
Epoch 58/120
64/64 [==============================] - 6s 91ms/step - loss: 0.4503 - acc:
0.8416 - val_loss: 0.9017 - val_acc: 0.7229
Epoch 59/120
64/64 [==============================] - 5s 85ms/step - loss: 0.4443 - acc:
```

```
0.8503 - val_loss: 0.8941 - val_acc: 0.7327
Epoch 60/120
64/64 [==============================] - 5s 83ms/step - loss: 0.4311 - acc:
0.8513 - val_loss: 0.7915 - val_acc: 0.7476
Epoch 61/120
64/64 [==============================] - 5s 80ms/step - loss: 0.4302 - acc:
0.8535 - val_loss: 0.8231 - val_acc: 0.7546
Epoch 62/120
64/64 [==============================] - 5s 77ms/step - loss: 0.4212 - acc:
0.8540 - val_loss: 0.8168 - val_acc: 0.7471
Epoch 63/120
64/64 [==============================] - 5s 83ms/step - loss: 0.4336 - acc:
0.8547 - val_loss: 0.7885 - val_acc: 0.7556
Epoch 64/120
64/64 [==============================] - 5s 77ms/step - loss: 0.3902 - acc:
0.8713 - val_loss: 0.8667 - val_acc: 0.7373
Epoch 65/120
64/64 [==============================] - 6s 89ms/step - loss: 0.3847 - acc:
0.8694 - val_loss: 0.8655 - val_acc: 0.7380
Epoch 66/120
64/64 [==============================] - 6s 89ms/step - loss: 0.3957 - acc:
0.8677 - val_loss: 0.8617 - val_acc: 0.7334
Epoch 67/120
64/64 [==============================] - 5s 83ms/step - loss: 0.3584 - acc:
0.8774 - val_loss: 0.9219 - val_acc: 0.7456
Epoch 68/120
64/64 [==============================] - 5s 80ms/step - loss: 0.3939 - acc:
0.8726 - val_loss: 0.7842 - val_acc: 0.7471
Epoch 69/120
64/64 [==============================] - 5s 79ms/step - loss: 0.3614 - acc:
0.8814 - val_loss: 0.9192 - val_acc: 0.7461
Epoch 70/120
64/64 [==============================] - 5s 85ms/step - loss: 0.3404 - acc:
0.8835 - val_loss: 0.8463 - val_acc: 0.7449
Epoch 71/120
64/64 [==============================] - 5s 80ms/step - loss: 0.3863 - acc:
0.8679 - val_loss: 0.8525 - val_acc: 0.7444
Epoch 72/120
64/64 [==============================] - 6s 88ms/step - loss: 0.3566 - acc:
0.8779 - val_loss: 0.7862 - val_acc: 0.7539
Epoch 73/120
64/64 [==============================] - 5s 82ms/step - loss: 0.3343 - acc:
0.8940 - val_loss: 0.8560 - val_acc: 0.7517
Epoch 74/120
64/64 [==============================] - 6s 89ms/step - loss: 0.3411 - acc:
0.8835 - val_loss: 0.7482 - val_acc: 0.7644
Epoch 75/120
64/64 [==============================] - 6s 88ms/step - loss: 0.3355 - acc:
```

```
0.8875 - val_loss: 0.9808 - val_acc: 0.7380
Epoch 76/120
64/64 [==============================] - 5s 85ms/step - loss: 0.3382 - acc:
0.8889 - val_loss: 0.8622 - val_acc: 0.7566
Epoch 77/120
64/64 [==============================] - 5s 84ms/step - loss: 0.3125 - acc:
0.8904 - val_loss: 0.9771 - val_acc: 0.7441
Epoch 78/120
64/64 [==============================] - 5s 83ms/step - loss: 0.3412 - acc:
0.8892 - val_loss: 0.9391 - val_acc: 0.7375
Epoch 79/120
64/64 [==============================] - 5s 85ms/step - loss: 0.3115 - acc:
0.8940 - val_loss: 0.9177 - val_acc: 0.7505
Epoch 80/120
64/64 [==============================] - 5s 84ms/step - loss: 0.3363 - acc:
0.8853 - val_loss: 0.8984 - val_acc: 0.7429
Epoch 81/120
64/64 [==============================] - 6s 88ms/step - loss: 0.3209 - acc:
0.8911 - val_loss: 0.8771 - val_acc: 0.7571
Epoch 82/120
64/64 [==============================] - 5s 85ms/step - loss: 0.3377 - acc:
0.8911 - val_loss: 0.9011 - val_acc: 0.7563
Epoch 83/120
64/64 [==============================] - 5s 81ms/step - loss: 0.3167 - acc:
0.8997 - val_loss: 0.8849 - val_acc: 0.7588
Epoch 84/120
64/64 [==============================] - 5s 81ms/step - loss: 0.3344 - acc:
0.9001 - val_loss: 0.9187 - val_acc: 0.7522
Epoch 85/120
64/64 [==============================] - 5s 83ms/step - loss: 0.2839 - acc:
0.9038 - val_loss: 0.9603 - val_acc: 0.7480
Epoch 86/120
64/64 [==============================] - 5s 77ms/step - loss: 0.3059 - acc:
0.9053 - val_loss: 0.8682 - val_acc: 0.7498
Epoch 87/120
64/64 [==============================] - 5s 85ms/step - loss: 0.2845 - acc:
0.9053 - val_loss: 0.9808 - val_acc: 0.7554
Epoch 88/120
64/64 [==============================] - 6s 91ms/step - loss: 0.3001 - acc:
0.8994 - val_loss: 1.0050 - val_acc: 0.7566
Epoch 89/120
64/64 [==============================] - 6s 87ms/step - loss: 0.2986 - acc:
0.8992 - val_loss: 0.9269 - val_acc: 0.7581
Epoch 90/120
64/64 [==============================] - 5s 86ms/step - loss: 0.2904 - acc:
0.9009 - val_loss: 0.9744 - val_acc: 0.7456
Epoch 91/120
64/64 [==============================] - 6s 88ms/step - loss: 0.2812 - acc:
```

```
0.9041 - val_loss: 1.2949 - val_acc: 0.7139
Epoch 92/120
64/64 [==============================] - 6s 88ms/step - loss: 0.2690 - acc:
0.9075 - val_loss: 1.0947 - val_acc: 0.7388
Epoch 93/120
64/64 [==============================] - 6s 100ms/step - loss: 0.2797 - acc:
0.9097 - val_loss: 0.8920 - val_acc: 0.7522
Epoch 94/120
64/64 [==============================] - 6s 86ms/step - loss: 0.2887 - acc:
0.9006 - val_loss: 1.0111 - val_acc: 0.7393
Epoch 95/120
64/64 [==============================] - 6s 88ms/step - loss: 0.2759 - acc:
0.9126 - val_loss: 0.9614 - val_acc: 0.7507
Epoch 96/120
64/64 [==============================] - 5s 84ms/step - loss: 0.2597 - acc:
0.9177 - val_loss: 1.0302 - val_acc: 0.7332
Epoch 97/120
64/64 [==============================] - 6s 86ms/step - loss: 0.2862 - acc:
0.9041 - val_loss: 1.0637 - val_acc: 0.7363
Epoch 98/120
64/64 [==============================] - 5s 81ms/step - loss: 0.2822 - acc:
0.9087 - val_loss: 0.9163 - val_acc: 0.7705
Epoch 99/120
64/64 [==============================] - 5s 80ms/step - loss: 0.2686 - acc:
0.9119 - val_loss: 0.9484 - val_acc: 0.7581
Epoch 100/120
64/64 [==============================] - 5s 72ms/step - loss: 0.2810 - acc:
0.9084 - val_loss: 0.9531 - val_acc: 0.7649
Epoch 101/120
64/64 [==============================] - 5s 76ms/step - loss: 0.2775 - acc:
0.9116 - val_loss: 1.0852 - val_acc: 0.7473
Epoch 102/120
64/64 [==============================] - 5s 79ms/step - loss: 0.2580 - acc:
0.9187 - val_loss: 1.0781 - val_acc: 0.7454
Epoch 103/120
64/64 [==============================] - 6s 96ms/step - loss: 0.2558 - acc:
0.9128 - val_loss: 1.1520 - val_acc: 0.7324
Epoch 104/120
64/64 [==============================] - 5s 83ms/step - loss: 0.2558 - acc:
0.9165 - val_loss: 0.9451 - val_acc: 0.7500
Epoch 105/120
64/64 [==============================] - 6s 101ms/step - loss: 0.2774 - acc:
0.9111 - val_loss: 1.0412 - val_acc: 0.7651
Epoch 106/120
64/64 [==============================] - 6s 87ms/step - loss: 0.2541 - acc:
0.9148 - val_loss: 1.0956 - val_acc: 0.7451
Epoch 107/120
64/64 [==============================] - 5s 86ms/step - loss: 0.2699 - acc:
```

```
0.9148 - val_loss: 0.9393 - val_acc: 0.7463
Epoch 108/120
64/64 [==============================] - 7s 105ms/step - loss: 0.2556 - acc:
0.9209 - val_loss: 1.1127 - val_acc: 0.7461
Epoch 109/120
64/64 [==============================] - 6s 101ms/step - loss: 0.2741 - acc:
0.9136 - val_loss: 1.0853 - val_acc: 0.7427
Epoch 110/120
64/64 [==============================] - 5s 85ms/step - loss: 0.2527 - acc:
0.9211 - val_loss: 1.0489 - val_acc: 0.7515
Epoch 111/120
64/64 [==============================] - 6s 90ms/step - loss: 0.2588 - acc:
0.9165 - val_loss: 0.9446 - val_acc: 0.7544
Epoch 112/120
64/64 [==============================] - 5s 82ms/step - loss: 0.2502 - acc:
0.9170 - val_loss: 1.0890 - val_acc: 0.7654
Epoch 113/120
64/64 [==============================] - 5s 83ms/step - loss: 0.2409 - acc:
0.9207 - val_loss: 1.1248 - val_acc: 0.7566
Epoch 114/120
64/64 [==============================] - 6s 86ms/step - loss: 0.2605 - acc:
0.9143 - val_loss: 1.0767 - val_acc: 0.7429
Epoch 115/120
64/64 [==============================] - 5s 82ms/step - loss: 0.2446 - acc:
0.9243 - val_loss: 1.2310 - val_acc: 0.7607
Epoch 116/120
64/64 [==============================] - 5s 82ms/step - loss: 0.2617 - acc:
0.9160 - val_loss: 1.0091 - val_acc: 0.7581
Epoch 117/120
64/64 [==============================] - 6s 87ms/step - loss: 0.2581 - acc:
0.9180 - val_loss: 0.9574 - val_acc: 0.7671
Epoch 118/120
64/64 [==============================] - 5s 83ms/step - loss: 0.2475 - acc:
0.9221 - val_loss: 1.1926 - val_acc: 0.7473
Epoch 119/120
64/64 [==============================] - 6s 87ms/step - loss: 0.2466 - acc:
0.9214 - val_loss: 1.2225 - val_acc: 0.7563
Epoch 120/120
64/64 [==============================] - 5s 84ms/step - loss: 0.2290 - acc:
0.9270 - val_loss: 1.2916 - val_acc: 0.7620
```

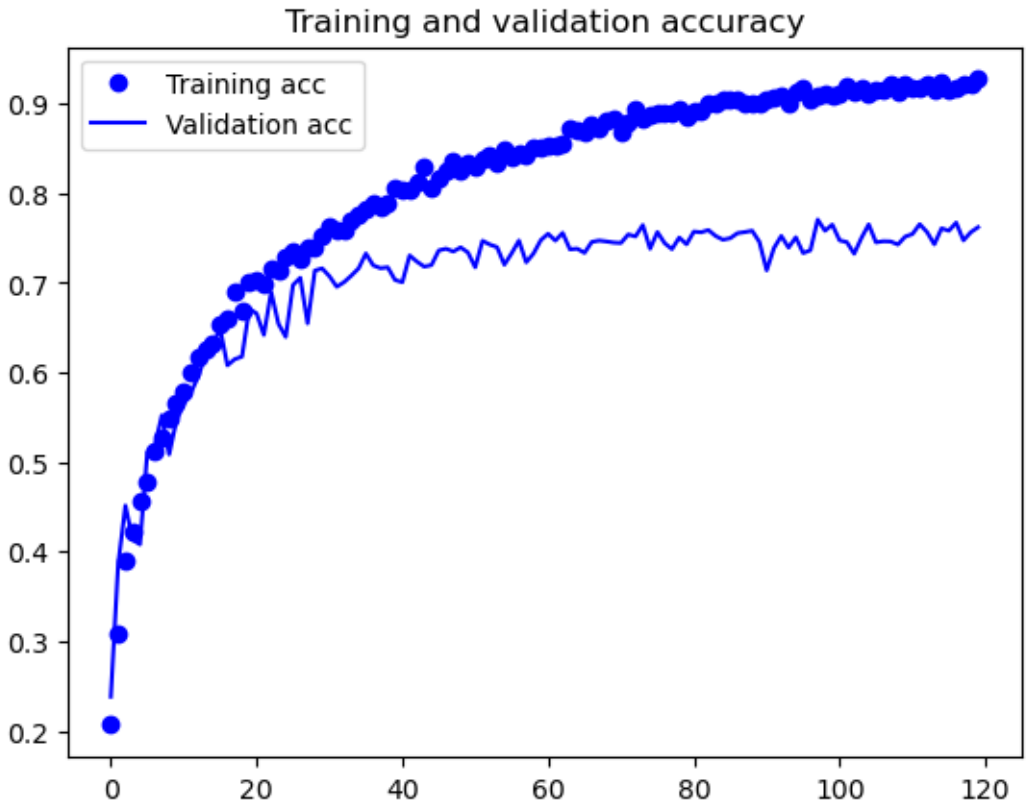
[ ]:
```
model.save('results/cifar10_model.h5')
```

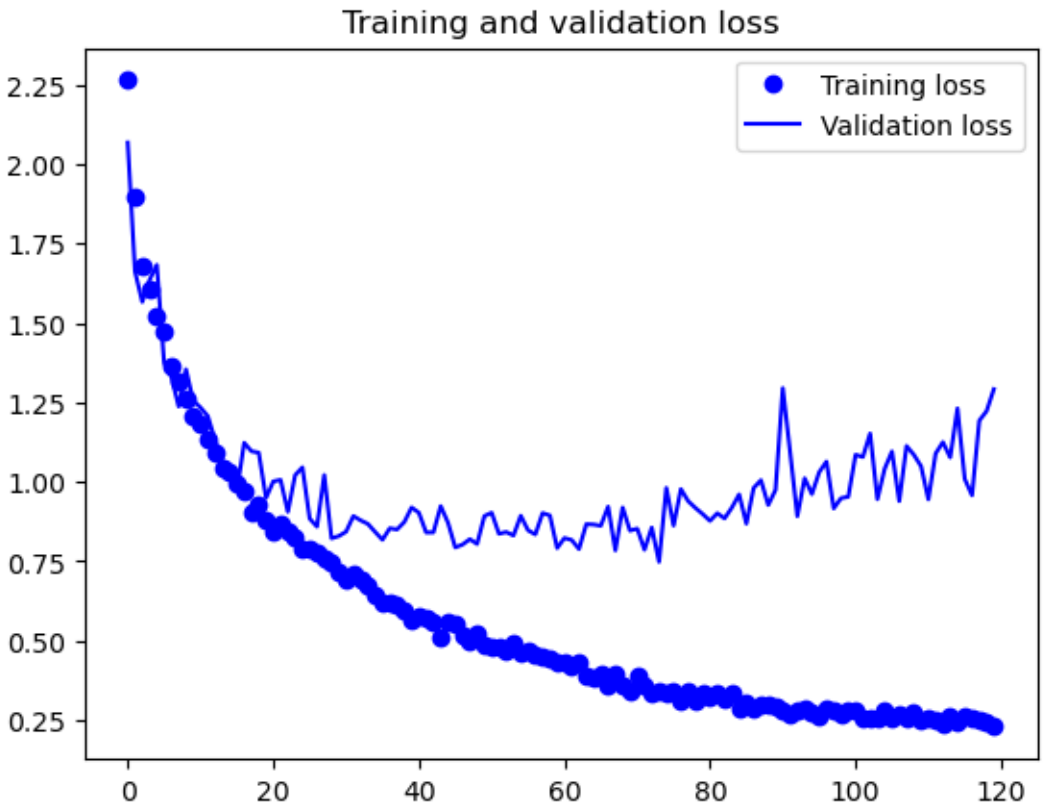
[ ]:
```
acc = history.history['acc']
val_acc = history.history['val_acc']
loss = history.history['loss']
val_loss = history.history['val_loss']
```

```
epochs = range(len(acc))

plt.plot(epochs, acc, 'bo', label='Training acc')
plt.plot(epochs, val_acc, 'b', label='Validation acc')
plt.title('Training and validation accuracy')
plt.legend()
plt.savefig("results/cifar10_model_acc.png")
plt.figure()

plt.plot(epochs, loss, 'bo', label='Training loss')
plt.plot(epochs, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.legend()
plt.savefig("results/cifar10_model_loss.png")
plt.show()
```





0.1.2 Assignment 6.2.b

[ ]:
```
(x_train, y_train), (x_test, y_test) = cifar10.load_data()
print(x_train.shape == (50000, 32, 32, 3))
print(x_test.shape == (10000, 32, 32, 3))
print(y_train.shape == (50000, 1))
print(y_test.shape == (10000, 1))

num_classes = 10
y_train = to_categorical(y_train, num_classes)
y_test = to_categorical(y_test, num_classes)
```

```
True
True
True
True
```

[ ]:
```
model = models.Sequential()

model = models.Sequential()
```

```
model.add(layers.Conv2D(32, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same', input_shape=(32, 32, 3)))
model.add(layers.BatchNormalization())
model.add(layers.Conv2D(32, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Dropout(0.2))
model.add(layers.Conv2D(64, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same'))
model.add(layers.BatchNormalization())
model.add(layers.Conv2D(64, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Dropout(0.3))
model.add(layers.Conv2D(128, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same'))
model.add(layers.BatchNormalization())
model.add(layers.Conv2D(128, (3, 3), activation='relu',␣
 ↪kernel_initializer='he_uniform', padding='same'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Dropout(0.4))
model.add(layers.Flatten())
model.add(layers.Dense(128, activation='relu', kernel_initializer='he_uniform'))
model.add(layers.Dropout(0.5))
model.add(layers.Dense(10, activation='softmax'))


model.summary()

model.compile(loss='categorical_crossentropy', optimizer=optimizers.
 ↪RMSprop(learning_rate=0.001), metrics=['acc'])
```

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_6 (Conv2D)           (None, 32, 32, 32)        896

 batch_normalization (BatchN  (None, 32, 32, 32)       128
 ormalization)

 conv2d_7 (Conv2D)           (None, 32, 32, 32)        9248

 max_pooling2d_3 (MaxPooling  (None, 16, 16, 32)       0
 2D)

 dropout (Dropout)           (None, 16, 16, 32)        0
```

```
conv2d_8 (Conv2D)           (None, 16, 16, 64)        18496

batch_normalization_1 (Batc  (None, 16, 16, 64)       256
hNormalization)

conv2d_9 (Conv2D)           (None, 16, 16, 64)        36928

_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_6 (Conv2D)           (None, 32, 32, 32)        896

batch_normalization (BatchN  (None, 32, 32, 32)       128
ormalization)

conv2d_7 (Conv2D)           (None, 32, 32, 32)        9248

max_pooling2d_3 (MaxPooling  (None, 16, 16, 32)       0
2D)

dropout (Dropout)           (None, 16, 16, 32)        0

conv2d_8 (Conv2D)           (None, 16, 16, 64)        18496

batch_normalization_1 (Batc  (None, 16, 16, 64)       256
hNormalization)

conv2d_9 (Conv2D)           (None, 16, 16, 64)        36928

max_pooling2d_4 (MaxPooling  (None, 8, 8, 64)         0
2D)

dropout_1 (Dropout)         (None, 8, 8, 64)          0

conv2d_10 (Conv2D)          (None, 8, 8, 128)         73856

batch_normalization_2 (Batc  (None, 8, 8, 128)        512
hNormalization)

conv2d_11 (Conv2D)          (None, 8, 8, 128)         147584

max_pooling2d_5 (MaxPooling  (None, 4, 4, 128)        0
2D)

dropout_2 (Dropout)         (None, 4, 4, 128)         0

flatten_1 (Flatten)         (None, 2048)              0
```

```
 dense_2 (Dense)              (None, 128)               262272

 dropout_3 (Dropout)          (None, 128)               0

 dense_3 (Dense)              (None, 10)                1290

=================================================================
Total params: 551,466
Trainable params: 551,018
Non-trainable params: 448
_________________________________________________________________
```

[ ]:
```
train_datagen = ImageDataGenerator(
            rescale=1./255,
            rotation_range=40,
            width_shift_range=0.2,
            height_shift_range=0.2,
            shear_range=0.2,
            zoom_range=0.2,
            horizontal_flip=True)

test_datagen = ImageDataGenerator(rescale=1./255)

train_datagen.fit(x_train)
test_datagen.fit(x_test)

train_generator = train_datagen.flow(x_train, y_train, batch_size=64)

validation_generator = test_datagen.flow(x_test, y_test, batch_size=64)

for data_batch, labels_batch in train_generator:
    print('data batch shape:', data_batch.shape)
    print('labels batch shape:', labels_batch.shape)
    break
```

```
data batch shape: (64, 32, 32, 3)
labels batch shape: (64, 10)
```

[ ]:
```
csv_logger = CSVLogger('results/cifar10_model_augmented.log')
history = model.fit(train_generator, steps_per_epoch=64, epochs=120,␣
 ↪validation_data=(validation_generator), validation_steps=64,␣
 ↪callbacks=csv_logger)
```

```
Epoch 1/120
64/64 [==============================] - 8s 100ms/step - loss: 2.5987 - acc:
0.1233 - val_loss: 2.2408 - val_acc: 0.1531
Epoch 2/120
64/64 [==============================] - 7s 104ms/step - loss: 2.2879 - acc:
```

```
0.1453 - val_loss: 2.2575 - val_acc: 0.1772
Epoch 3/120
64/64 [==============================] - 6s 99ms/step - loss: 2.2277 - acc:
0.1663 - val_loss: 2.1114 - val_acc: 0.2622
Epoch 4/120
64/64 [==============================] - 6s 96ms/step - loss: 2.2226 - acc:
0.1555 - val_loss: 2.0919 - val_acc: 0.2185
Epoch 5/120
64/64 [==============================] - 6s 93ms/step - loss: 2.1985 - acc:
0.1626 - val_loss: 2.0392 - val_acc: 0.2537
Epoch 6/120
64/64 [==============================] - 6s 89ms/step - loss: 2.1709 - acc:
0.1711 - val_loss: 1.9590 - val_acc: 0.2747
Epoch 7/120
64/64 [==============================] - 6s 89ms/step - loss: 2.1277 - acc:
0.1946 - val_loss: 1.9386 - val_acc: 0.2859
Epoch 8/120
64/64 [==============================] - 6s 100ms/step - loss: 2.1086 - acc:
0.1973 - val_loss: 1.9024 - val_acc: 0.2886
Epoch 9/120
64/64 [==============================] - 6s 99ms/step - loss: 2.0785 - acc:
0.2080 - val_loss: 1.8696 - val_acc: 0.2852
Epoch 10/120
64/64 [==============================] - 7s 105ms/step - loss: 2.0796 - acc:
0.2134 - val_loss: 1.9034 - val_acc: 0.2837
Epoch 11/120
64/64 [==============================] - 6s 94ms/step - loss: 2.0333 - acc:
0.2180 - val_loss: 1.8248 - val_acc: 0.3105
Epoch 12/120
64/64 [==============================] - 7s 104ms/step - loss: 2.0176 - acc:
0.2444 - val_loss: 1.7182 - val_acc: 0.3311
Epoch 13/120
64/64 [==============================] - 6s 96ms/step - loss: 2.0262 - acc:
0.2478 - val_loss: 1.7975 - val_acc: 0.3105
Epoch 14/120
64/64 [==============================] - 6s 92ms/step - loss: 1.9381 - acc:
0.2578 - val_loss: 1.9106 - val_acc: 0.2891
Epoch 15/120
64/64 [==============================] - 6s 101ms/step - loss: 1.9674 - acc:
0.2556 - val_loss: 1.8659 - val_acc: 0.2842
Epoch 16/120
64/64 [==============================] - 7s 101ms/step - loss: 1.9270 - acc:
0.2710 - val_loss: 1.7837 - val_acc: 0.3240
Epoch 17/120
64/64 [==============================] - 6s 97ms/step - loss: 1.8888 - acc:
0.2824 - val_loss: 1.7248 - val_acc: 0.3486
Epoch 18/120
64/64 [==============================] - 6s 97ms/step - loss: 1.8689 - acc:
```

```
0.3100 - val_loss: 1.7429 - val_acc: 0.3257
Epoch 19/120
64/64 [==============================] - 6s 94ms/step - loss: 1.8640 - acc:
0.3076 - val_loss: 1.7534 - val_acc: 0.3391
Epoch 20/120
64/64 [==============================] - 6s 89ms/step - loss: 1.8092 - acc:
0.3293 - val_loss: 1.6144 - val_acc: 0.4028
Epoch 21/120
64/64 [==============================] - 6s 98ms/step - loss: 1.8243 - acc:
0.3228 - val_loss: 1.6353 - val_acc: 0.3743
Epoch 22/120
64/64 [==============================] - 7s 102ms/step - loss: 1.8092 - acc:
0.3279 - val_loss: 1.5401 - val_acc: 0.4043
Epoch 23/120
64/64 [==============================] - 6s 93ms/step - loss: 1.7805 - acc:
0.3496 - val_loss: 1.5517 - val_acc: 0.4207
Epoch 24/120
64/64 [==============================] - 6s 96ms/step - loss: 1.7309 - acc:
0.3579 - val_loss: 1.5767 - val_acc: 0.4241
Epoch 25/120
64/64 [==============================] - 6s 99ms/step - loss: 1.7183 - acc:
0.3711 - val_loss: 1.6139 - val_acc: 0.4116
Epoch 26/120
64/64 [==============================] - 6s 89ms/step - loss: 1.7317 - acc:
0.3684 - val_loss: 1.6516 - val_acc: 0.4397
Epoch 27/120
64/64 [==============================] - 6s 93ms/step - loss: 1.7146 - acc:
0.3743 - val_loss: 1.4786 - val_acc: 0.4495
Epoch 28/120
64/64 [==============================] - 7s 102ms/step - loss: 1.6859 - acc:
0.3938 - val_loss: 1.6923 - val_acc: 0.4275
Epoch 29/120
64/64 [==============================] - 6s 95ms/step - loss: 1.7057 - acc:
0.3943 - val_loss: 2.0006 - val_acc: 0.3862
Epoch 30/120
64/64 [==============================] - 6s 100ms/step - loss: 1.6839 - acc:
0.3999 - val_loss: 1.5169 - val_acc: 0.4475
Epoch 31/120
64/64 [==============================] - 6s 92ms/step - loss: 1.6362 - acc:
0.4143 - val_loss: 1.6571 - val_acc: 0.4209
Epoch 32/120
64/64 [==============================] - 6s 98ms/step - loss: 1.6421 - acc:
0.4092 - val_loss: 1.5103 - val_acc: 0.4463
Epoch 33/120
64/64 [==============================] - 6s 93ms/step - loss: 1.6375 - acc:
0.3992 - val_loss: 1.5191 - val_acc: 0.4558
Epoch 34/120
64/64 [==============================] - 6s 99ms/step - loss: 1.6302 - acc:
```

```
0.4094 - val_loss: 1.3421 - val_acc: 0.5110
Epoch 35/120
64/64 [==============================] - 6s 99ms/step - loss: 1.6514 - acc:
0.4067 - val_loss: 1.3984 - val_acc: 0.4937
Epoch 36/120
64/64 [==============================] - 6s 92ms/step - loss: 1.6027 - acc:
0.4177 - val_loss: 1.4571 - val_acc: 0.4971
Epoch 37/120
64/64 [==============================] - 6s 100ms/step - loss: 1.6085 - acc:
0.4268 - val_loss: 1.4759 - val_acc: 0.4968
Epoch 38/120
64/64 [==============================] - 6s 99ms/step - loss: 1.5992 - acc:
0.4280 - val_loss: 1.4941 - val_acc: 0.4678
Epoch 39/120
64/64 [==============================] - 6s 91ms/step - loss: 1.6166 - acc:
0.4226 - val_loss: 1.6181 - val_acc: 0.4402
Epoch 40/120
64/64 [==============================] - 7s 102ms/step - loss: 1.5832 - acc:
0.4399 - val_loss: 1.5107 - val_acc: 0.5005
Epoch 41/120
64/64 [==============================] - 6s 92ms/step - loss: 1.5913 - acc:
0.4309 - val_loss: 1.4363 - val_acc: 0.4868
Epoch 42/120
64/64 [==============================] - 6s 98ms/step - loss: 1.5615 - acc:
0.4507 - val_loss: 1.3950 - val_acc: 0.4968
Epoch 43/120
64/64 [==============================] - 6s 92ms/step - loss: 1.5662 - acc:
0.4404 - val_loss: 1.4530 - val_acc: 0.4719
Epoch 44/120
64/64 [==============================] - 6s 95ms/step - loss: 1.5480 - acc:
0.4492 - val_loss: 1.4069 - val_acc: 0.5112
Epoch 45/120
64/64 [==============================] - 6s 100ms/step - loss: 1.5465 - acc:
0.4492 - val_loss: 1.5383 - val_acc: 0.4912
Epoch 46/120
64/64 [==============================] - 6s 93ms/step - loss: 1.5651 - acc:
0.4553 - val_loss: 1.4056 - val_acc: 0.5156
Epoch 47/120
64/64 [==============================] - 6s 97ms/step - loss: 1.5200 - acc:
0.4597 - val_loss: 1.4553 - val_acc: 0.5176
Epoch 48/120
64/64 [==============================] - 6s 99ms/step - loss: 1.5568 - acc:
0.4541 - val_loss: 1.9721 - val_acc: 0.4243
Epoch 49/120
64/64 [==============================] - 6s 97ms/step - loss: 1.5012 - acc:
0.4700 - val_loss: 1.3766 - val_acc: 0.5076
Epoch 50/120
64/64 [==============================] - 7s 110ms/step - loss: 1.5222 - acc:
```

```
0.4536 - val_loss: 1.4215 - val_acc: 0.5208
Epoch 51/120
64/64 [==============================] - 7s 104ms/step - loss: 1.5499 - acc:
0.4460 - val_loss: 1.7816 - val_acc: 0.3994
Epoch 52/120
64/64 [==============================] - 7s 109ms/step - loss: 1.5239 - acc:
0.4568 - val_loss: 1.8406 - val_acc: 0.4543
Epoch 53/120
64/64 [==============================] - 7s 105ms/step - loss: 1.5051 - acc:
0.4729 - val_loss: 1.5639 - val_acc: 0.4807
Epoch 54/120
64/64 [==============================] - 8s 120ms/step - loss: 1.4896 - acc:
0.4731 - val_loss: 1.3137 - val_acc: 0.5432
Epoch 55/120
64/64 [==============================] - 7s 112ms/step - loss: 1.4819 - acc:
0.4778 - val_loss: 1.4515 - val_acc: 0.5000
Epoch 56/120
64/64 [==============================] - 8s 120ms/step - loss: 1.5098 - acc:
0.4734 - val_loss: 1.6315 - val_acc: 0.4749
Epoch 57/120
64/64 [==============================] - 7s 112ms/step - loss: 1.5010 - acc:
0.4666 - val_loss: 1.5014 - val_acc: 0.5134
Epoch 58/120
64/64 [==============================] - 7s 112ms/step - loss: 1.4903 - acc:
0.4688 - val_loss: 1.2874 - val_acc: 0.5635
Epoch 59/120
64/64 [==============================] - 7s 114ms/step - loss: 1.4939 - acc:
0.4807 - val_loss: 1.4909 - val_acc: 0.5164
Epoch 60/120
64/64 [==============================] - 8s 122ms/step - loss: 1.4960 - acc:
0.4763 - val_loss: 1.2617 - val_acc: 0.5479
Epoch 61/120
64/64 [==============================] - 7s 116ms/step - loss: 1.4551 - acc:
0.4961 - val_loss: 1.3506 - val_acc: 0.5503
Epoch 62/120
64/64 [==============================] - 6s 99ms/step - loss: 1.4796 - acc:
0.4817 - val_loss: 1.4481 - val_acc: 0.4927
Epoch 63/120
64/64 [==============================] - 7s 114ms/step - loss: 1.4816 - acc:
0.4881 - val_loss: 1.2728 - val_acc: 0.5527
Epoch 64/120
64/64 [==============================] - 7s 112ms/step - loss: 1.4575 - acc:
0.4907 - val_loss: 1.3622 - val_acc: 0.5291
Epoch 65/120
64/64 [==============================] - 7s 113ms/step - loss: 1.4924 - acc:
0.4695 - val_loss: 1.2861 - val_acc: 0.5444
Epoch 66/120
64/64 [==============================] - 7s 110ms/step - loss: 1.4873 - acc:
```

```
0.4834 - val_loss: 1.2144 - val_acc: 0.5813
Epoch 67/120
64/64 [==============================] - 7s 109ms/step - loss: 1.4479 - acc:
0.5027 - val_loss: 1.2684 - val_acc: 0.5510
Epoch 68/120
64/64 [==============================] - 7s 101ms/step - loss: 1.4456 - acc:
0.4963 - val_loss: 1.5547 - val_acc: 0.5278
Epoch 69/120
64/64 [==============================] - 7s 114ms/step - loss: 1.4755 - acc:
0.4951 - val_loss: 1.4644 - val_acc: 0.5234
Epoch 70/120
64/64 [==============================] - 7s 113ms/step - loss: 1.4801 - acc:
0.4841 - val_loss: 1.5893 - val_acc: 0.4839
Epoch 71/120
64/64 [==============================] - 7s 104ms/step - loss: 1.4484 - acc:
0.4878 - val_loss: 1.2841 - val_acc: 0.5601
Epoch 72/120
64/64 [==============================] - 8s 117ms/step - loss: 1.4715 - acc:
0.4897 - val_loss: 1.0982 - val_acc: 0.6086
Epoch 73/120
64/64 [==============================] - 7s 114ms/step - loss: 1.4493 - acc:
0.4983 - val_loss: 1.2617 - val_acc: 0.5732
Epoch 74/120
64/64 [==============================] - 7s 115ms/step - loss: 1.4302 - acc:
0.5020 - val_loss: 1.7029 - val_acc: 0.4646
Epoch 75/120
64/64 [==============================] - 7s 113ms/step - loss: 1.4508 - acc:
0.4880 - val_loss: 2.1206 - val_acc: 0.4280
Epoch 76/120
64/64 [==============================] - 7s 116ms/step - loss: 1.4500 - acc:
0.4915 - val_loss: 1.5074 - val_acc: 0.4961
Epoch 77/120
64/64 [==============================] - 7s 113ms/step - loss: 1.4588 - acc:
0.4910 - val_loss: 1.4485 - val_acc: 0.5168
Epoch 78/120
64/64 [==============================] - 6s 100ms/step - loss: 1.4478 - acc:
0.5000 - val_loss: 1.4873 - val_acc: 0.4814
Epoch 79/120
64/64 [==============================] - 7s 111ms/step - loss: 1.4384 - acc:
0.5054 - val_loss: 1.4425 - val_acc: 0.5012
Epoch 80/120
64/64 [==============================] - 7s 110ms/step - loss: 1.4556 - acc:
0.4912 - val_loss: 1.3942 - val_acc: 0.5264
Epoch 81/120
64/64 [==============================] - 7s 105ms/step - loss: 1.4362 - acc:
0.5067 - val_loss: 1.2071 - val_acc: 0.5852
Epoch 82/120
64/64 [==============================] - 8s 117ms/step - loss: 1.4045 - acc:
```

```
0.5115 - val_loss: 1.3410 - val_acc: 0.5457
Epoch 83/120
64/64 [==============================] - 7s 111ms/step - loss: 1.4250 - acc:
0.4980 - val_loss: 1.2626 - val_acc: 0.5737
Epoch 84/120
64/64 [==============================] - 7s 104ms/step - loss: 1.4163 - acc:
0.5076 - val_loss: 1.9864 - val_acc: 0.4658
Epoch 85/120
64/64 [==============================] - 7s 103ms/step - loss: 1.4357 - acc:
0.5061 - val_loss: 1.6623 - val_acc: 0.5037
Epoch 86/120
64/64 [==============================] - 7s 112ms/step - loss: 1.3797 - acc:
0.5154 - val_loss: 1.2719 - val_acc: 0.5876
Epoch 87/120
64/64 [==============================] - 7s 114ms/step - loss: 1.4289 - acc:
0.5005 - val_loss: 1.9632 - val_acc: 0.4536
Epoch 88/120
64/64 [==============================] - 7s 114ms/step - loss: 1.4529 - acc:
0.5078 - val_loss: 1.5751 - val_acc: 0.5154
Epoch 89/120
64/64 [==============================] - 7s 107ms/step - loss: 1.4116 - acc:
0.5237 - val_loss: 1.5682 - val_acc: 0.4919
Epoch 90/120
64/64 [==============================] - 8s 119ms/step - loss: 1.4121 - acc:
0.5105 - val_loss: 1.4829 - val_acc: 0.5208
Epoch 91/120
64/64 [==============================] - 7s 114ms/step - loss: 1.4113 - acc:
0.5168 - val_loss: 1.4731 - val_acc: 0.5239
Epoch 92/120
64/64 [==============================] - 7s 105ms/step - loss: 1.4065 - acc:
0.5154 - val_loss: 1.2604 - val_acc: 0.5598
Epoch 93/120
64/64 [==============================] - 8s 119ms/step - loss: 1.4043 - acc:
0.5176 - val_loss: 1.4643 - val_acc: 0.5195
Epoch 94/120
64/64 [==============================] - 7s 115ms/step - loss: 1.4180 - acc:
0.5107 - val_loss: 1.3259 - val_acc: 0.5461
Epoch 95/120
64/64 [==============================] - 7s 116ms/step - loss: 1.3847 - acc:
0.5288 - val_loss: 1.1709 - val_acc: 0.5957
Epoch 96/120
64/64 [==============================] - 10s 150ms/step - loss: 1.3994 - acc:
0.5164 - val_loss: 1.3218 - val_acc: 0.5688
Epoch 97/120
64/64 [==============================] - 7s 110ms/step - loss: 1.4158 - acc:
0.5220 - val_loss: 1.2182 - val_acc: 0.5803
Epoch 98/120
64/64 [==============================] - 7s 111ms/step - loss: 1.3990 - acc:
```

```
0.5129 - val_loss: 1.3458 - val_acc: 0.5356
Epoch 99/120
64/64 [==============================] - 7s 113ms/step - loss: 1.4061 - acc:
0.5156 - val_loss: 1.1850 - val_acc: 0.5847
Epoch 100/120
64/64 [==============================] - 7s 104ms/step - loss: 1.4151 - acc:
0.5142 - val_loss: 1.1220 - val_acc: 0.6077
Epoch 101/120
64/64 [==============================] - 7s 109ms/step - loss: 1.3809 - acc:
0.5239 - val_loss: 1.4128 - val_acc: 0.5442
Epoch 102/120
64/64 [==============================] - 8s 119ms/step - loss: 1.3659 - acc:
0.5300 - val_loss: 1.9252 - val_acc: 0.4722
Epoch 103/120
64/64 [==============================] - 7s 109ms/step - loss: 1.3629 - acc:
0.5212 - val_loss: 1.6247 - val_acc: 0.5173
Epoch 104/120
64/64 [==============================] - 7s 113ms/step - loss: 1.3679 - acc:
0.5227 - val_loss: 1.3615 - val_acc: 0.5293
Epoch 105/120
64/64 [==============================] - 7s 116ms/step - loss: 1.3809 - acc:
0.5237 - val_loss: 1.1247 - val_acc: 0.6033
Epoch 106/120
64/64 [==============================] - 7s 114ms/step - loss: 1.3632 - acc:
0.5269 - val_loss: 1.1917 - val_acc: 0.5852
Epoch 107/120
64/64 [==============================] - 7s 104ms/step - loss: 1.4125 - acc:
0.5159 - val_loss: 1.3837 - val_acc: 0.5457
Epoch 108/120
64/64 [==============================] - 8s 120ms/step - loss: 1.3899 - acc:
0.5249 - val_loss: 1.2885 - val_acc: 0.5774
Epoch 109/120
64/64 [==============================] - 8s 119ms/step - loss: 1.3558 - acc:
0.5273 - val_loss: 1.3207 - val_acc: 0.5469
Epoch 110/120
64/64 [==============================] - 7s 110ms/step - loss: 1.3759 - acc:
0.5269 - val_loss: 1.2735 - val_acc: 0.5901
Epoch 111/120
64/64 [==============================] - 7s 113ms/step - loss: 1.3976 - acc:
0.5122 - val_loss: 1.7071 - val_acc: 0.5122
Epoch 112/120
64/64 [==============================] - 7s 114ms/step - loss: 1.3648 - acc:
0.5454 - val_loss: 1.5533 - val_acc: 0.5266
Epoch 113/120
64/64 [==============================] - 7s 112ms/step - loss: 1.3366 - acc:
0.5349 - val_loss: 1.1038 - val_acc: 0.6257
Epoch 114/120
64/64 [==============================] - 7s 110ms/step - loss: 1.3928 - acc:
```

```
0.5261 - val_loss: 1.3721 - val_acc: 0.5708
Epoch 115/120
64/64 [==============================] - 7s 115ms/step - loss: 1.3632 - acc:
0.5342 - val_loss: 1.3411 - val_acc: 0.5515
Epoch 116/120
64/64 [==============================] - 7s 114ms/step - loss: 1.3337 - acc:
0.5420 - val_loss: 1.4405 - val_acc: 0.5454
Epoch 117/120
64/64 [==============================] - 8s 123ms/step - loss: 1.3996 - acc:
0.5225 - val_loss: 1.4868 - val_acc: 0.5186
Epoch 118/120
64/64 [==============================] - 8s 117ms/step - loss: 1.3659 - acc:
0.5222 - val_loss: 1.2366 - val_acc: 0.5889
Epoch 119/120
64/64 [==============================] - 7s 115ms/step - loss: 1.3869 - acc:
0.5220 - val_loss: 1.2808 - val_acc: 0.5881
Epoch 120/120
64/64 [==============================] - 8s 118ms/step - loss: 1.3812 - acc:
0.5222 - val_loss: 1.6954 - val_acc: 0.4998
```

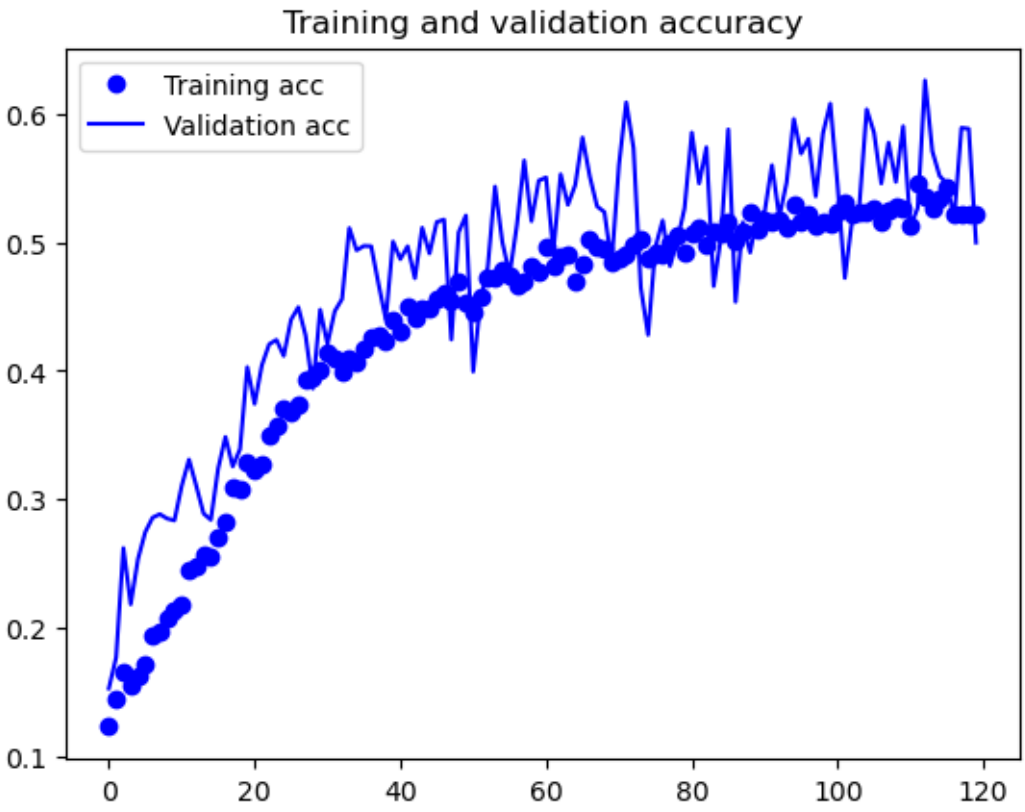
[ ]:
```
model.save('results/cifar10_model_augmented.h5')
```

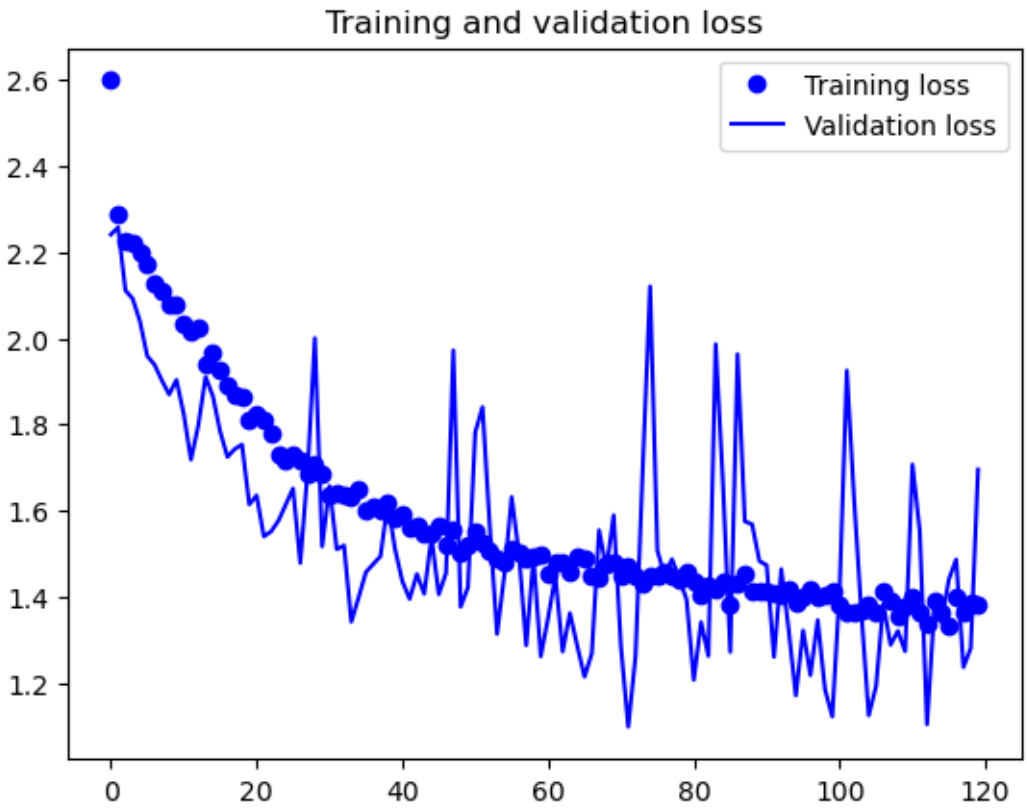
[ ]:
```
acc = history.history['acc']
val_acc = history.history['val_acc']
loss = history.history['loss']
val_loss = history.history['val_loss']

epochs = range(len(acc))

plt.plot(epochs, acc, 'bo', label='Training acc')
plt.plot(epochs, val_acc, 'b', label='Validation acc')
plt.title('Training and validation accuracy')
plt.legend()
plt.savefig("results/cifar10_model_acc_augmented.png")
plt.figure()

plt.plot(epochs, loss, 'bo', label='Training loss')
plt.plot(epochs, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.legend()
plt.savefig("results/cifar10_model_loss_augmented.png")
plt.show()
```


Training and validation accuracy
Training acc
Validation acc
0.6
0.5
0.4
0.3
0.2
0.1
0
20
40
60
80
100
120


Training and validation loss
Training loss
Validation loss
2.6
2.4
2.2
2.0
1.8
1.6
1.4
1.2
0
20
40
60
80
100
120