

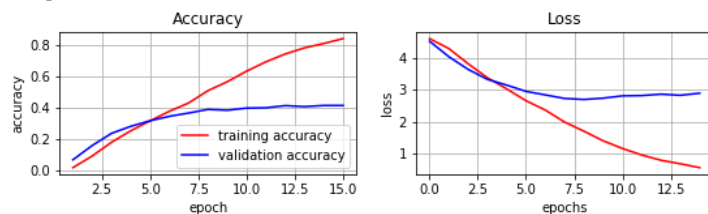
```

=====
exp = Experimento_1_1
modelo = 1
samples_per_class = 100
opti = Adam
batch = 32
epochs = 15
run = True
-----
Creando juego de datos ...
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:66: The name tf.get_defaul
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:541: The name tf.placeholder
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:4432: The name tf.random_
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:4267: The name tf.nn.max_
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:148: The name tf.placeholder
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:3733: calling dropout (fr
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate = 1 - keep_prob`.
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/optimizers.py:793: The name tf.train.Optimizer is depre
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:3576: The name tf.log is

Entrenando ...
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tensorflow_core/python/ops/math_grad.py:1424: where (from ter
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:1033: The name tf.assign_
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:1020: The name tf.assign
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:216: The name tf.is_varia
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:223: The name tf.variable

Tiempo de entrenamiento: ** 89.0 **

```



```

Evaluando ...
Exactitud validate: 41.46 %, Test: 41.46 %
Tiempo de ejecución del experimento: 90.0
=====

```

```

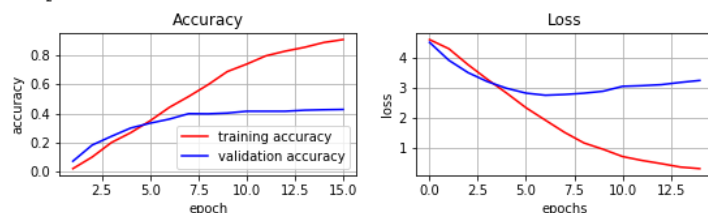
exp = Experimento_1_2
modelo = 2
samples_per_class = 100
opti = Adam
batch = 32
epochs = 15
run = True
-----

```

```

Entrenando ...
Tiempo de entrenamiento: ** 139.0 **

```



```

Evaluando ...
Exactitud validate: 42.71 %, Test: 43.12 %
Tiempo de ejecución del experimento: 141.0
=====

```

```

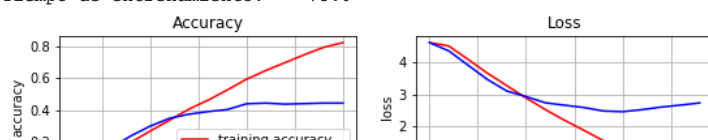
exp = Experimento_1_3
modelo = 3
samples_per_class = 100
opti = Adam
batch = 32
epochs = 15
run = True
-----

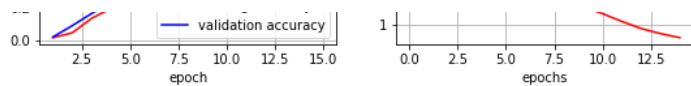
```

```

Entrenando ...
Tiempo de entrenamiento: ** 73.0 **

```



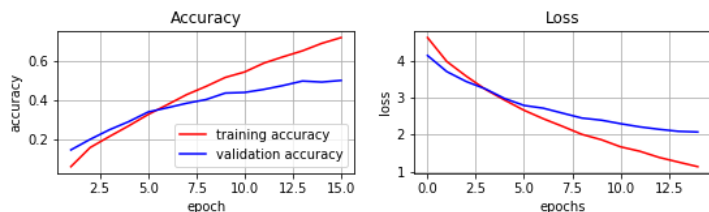


Evaluando ...
 Exactitud validate: 44.38 %, Test: 48.44 %
 Tiempo de ejecución del experimento: 74.0

```
=====
exp = Experimento_1_4
modelo = 4
samples_per_class = 100
opti = Adam
batch = 32
epochs = 15
run = True
```

 WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:2041: The name tf.nn.fuse

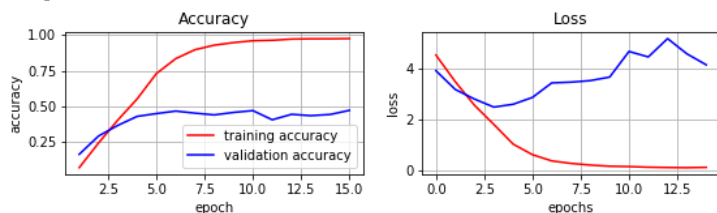
Entrenando ...
 Tiempo de entrenamiento: ** 44.0 **



Evaluando ...
 Exactitud validate: 50.1 %, Test: 52.08 %
 Tiempo de ejecución del experimento: 45.0

```
=====
exp = Experimento_2_1
modelo = 1
samples_per_class = 100
opti = rmsprop
batch = 32
epochs = 15
run = True
```

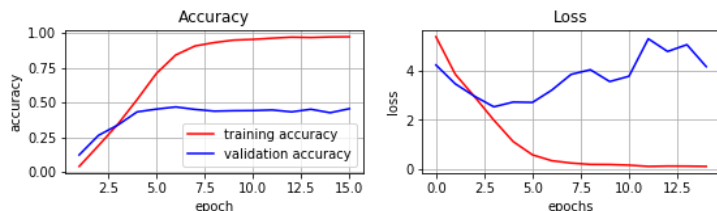
 Entrenando ...
 Tiempo de entrenamiento: ** 74.0 **



Evaluando ...
 Exactitud validate: 46.98 %, Test: 47.92 %
 Tiempo de ejecución del experimento: 75.0

```
=====
exp = Experimento_2_2
modelo = 2
samples_per_class = 100
opti = rmsprop
batch = 32
epochs = 15
run = True
```

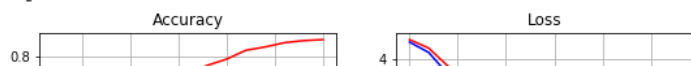
 Entrenando ...
 Tiempo de entrenamiento: ** 131.0 **

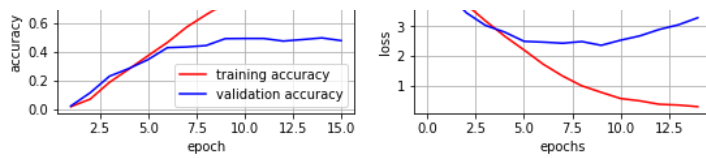


Evaluando ...
 Exactitud validate: 45.52 %, Test: 47.81 %
 Tiempo de ejecución del experimento: 132.0

```
=====
exp = Experimento_2_3
modelo = 3
samples_per_class = 100
opti = rmsprop
batch = 32
epochs = 15
run = True
```

 Entrenando ...
 Tiempo de entrenamiento: ** 71.0 **





Evaluando ...

Exactitud validate: 47.71 %, Test: 53.44 %

Tiempo de ejecución del experimento: 72.0

=====

exp = Experimento_2_4

modelo = 4

samples_per_class = 100

opti = rmsprop

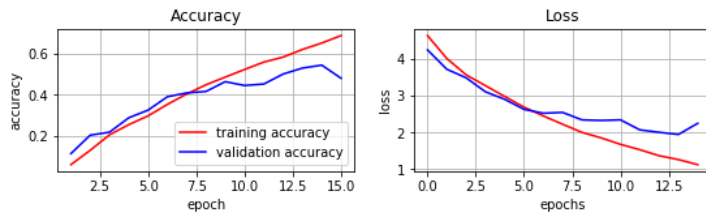
batch = 32

epochs = 15

run = True

Entrenando ...

Tiempo de entrenamiento: ** 43.0 **



Evaluando ...

Exactitud validate: 48.12 %, Test: 52.6 %

Tiempo de ejecución del experimento: 43.0

=====

exp = Experimento_3_1

modelo = 1

samples_per_class = 100

opti = Adam

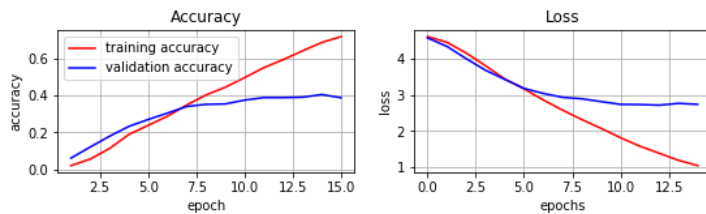
batch = 64

epochs = 15

run = True

Entrenando ...

Tiempo de entrenamiento: ** 69.0 **



Evaluando ...

Exactitud validate: 38.75 %, Test: 41.04 %

Tiempo de ejecución del experimento: 70.0

=====

exp = Experimento_3_2

modelo = 2

samples_per_class = 100

opti = Adam

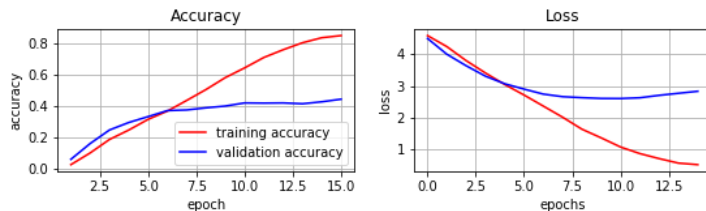
batch = 64

epochs = 15

run = True

Entrenando ...

Tiempo de entrenamiento: ** 122.0 **



Evaluando ...

Exactitud validate: 44.48 %, Test: 45.31 %

Tiempo de ejecución del experimento: 124.0

=====

exp = Experimento_3_3

modelo = 3

samples_per_class = 100

opti = Adam

batch = 64

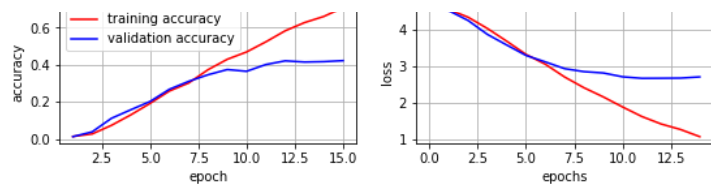
epochs = 15

run = True

Entrenando ...

Tiempo de entrenamiento: ** 69.0 **





Evaluando ...

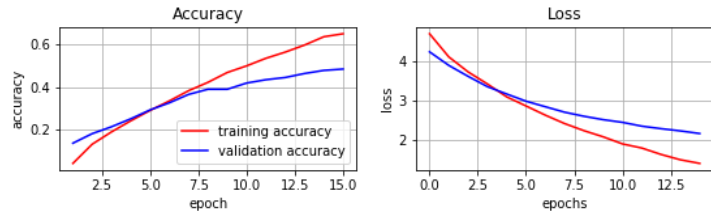
Exactitud validate: 42.29 %, Test: 44.69 %

Tiempo de ejecución del experimento: 70.0

```
=====
exp = Experimento_3_4
modelo = 4
samples_per_class = 100
opti = Adam
batch = 64
epochs = 15
run = True
```

Entrenando ...

Tiempo de entrenamiento: ** 41.0 **



Evaluando ...

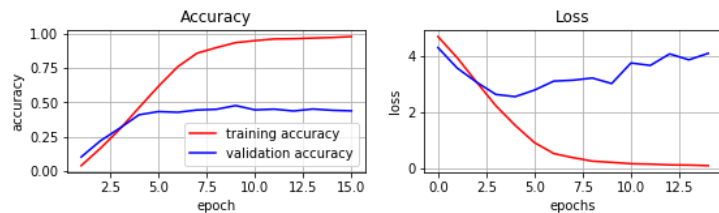
Exactitud validate: 48.54 %, Test: 49.69 %

Tiempo de ejecución del experimento: 42.0

```
=====
exp = Experimento_4_1
modelo = 1
samples_per_class = 100
opti = rmsprop
batch = 64
epochs = 15
run = True
```

Entrenando ...

Tiempo de entrenamiento: ** 66.0 **



Evaluando ...

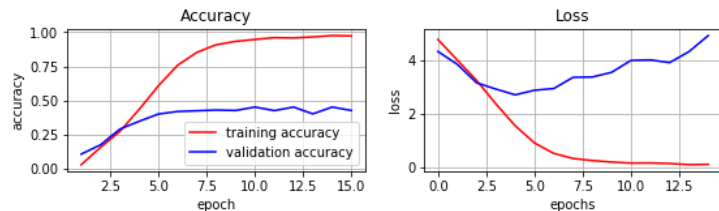
Exactitud validate: 43.85 %, Test: 46.56 %

Tiempo de ejecución del experimento: 67.0

```
=====
exp = Experimento_4_2
modelo = 2
samples_per_class = 100
opti = rmsprop
batch = 64
epochs = 15
run = True
```

Entrenando ...

Tiempo de entrenamiento: ** 117.0 **



Evaluando ...

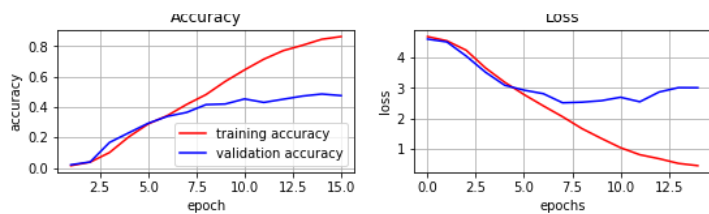
Exactitud validate: 42.92 %, Test: 44.06 %

Tiempo de ejecución del experimento: 118.0

```
=====
exp = Experimento_4_3
modelo = 3
samples_per_class = 100
opti = rmsprop
batch = 64
epochs = 15
run = True
```

Entrenando ...

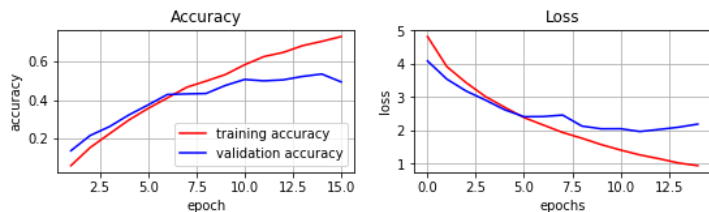
Tiempo de entrenamiento: ** 68.0 **



Evaluando ...
Exactitud validate: 47.6 %, Test: 50.31 %
Tiempo de ejecución del experimento: 69.0

```
=====
exp = Experimento_4_4
modelo = 4
samples_per_class = 100
opti = rmsprop
batch = 64
epochs = 15
run = True
```

Entrenando ...
Tiempo de entrenamiento: ** 40.0 **



Evaluando ...
Exactitud validate: 49.38 %, Test: 50.42 %
Tiempo de ejecución del experimento: 41.0
Tiempo de ejecución todo el batch: 1315.0

Resultados Lote 1

Experimento	Muestras*Clase	Optimizador	Modelo	Batch Size	epocas	Tiempo Entrenamiento	Exac. val.	Exact. test
Experimento_1_1	100	Adam	1	32	15	89	41.6%	41.6%
Experimento_1_2	100	Adam	2	32	15	139	42.71%	43.12%
Experimento_1_3	100	Adam	3	32	15	73	44.38%	48.44%
Experimento_1_4	100	Adam	4	32	15	44	50.10%	52.08%
Experimento_2_1	100	rmsprop	1	32	15	74	46.98%	47.92%
Experimento_2_2	100	rmsprop	2	32	15	131	45.52%	47.81%
Experimento_2_3	100	rmsprop	3	32	15	71	47.71%	53.44%
Experimento_2_4	100	rmsprop	4	32	15	43	48.12%	52.60%
Experimento_3_1	100	Adam	1	64	15	69	38.75%	41.04%
Experimento_3_2	100	Adam	2	64	15	122	44.48%	45.31%
Experimento_3_3	100	Adam	3	64	15	69	42.29%	44.69%
Experimento_3_4	100	Adam	4	64	15	41	48.54%	49.69%
Experimento_4_1	100	rmsprop	1	64	15	66	43.85%	46.56%
Experimento_4_2	100	rmsprop	2	64	15	117	42.92%	44.06%
Experimento_4_3	100	rmsprop	3	64	15	68	47.6%	50.31%
Experimento_4_4	100	rmsprop	4	64	15	40	49.38%	50.42%

Al parecer el batch_size=32 es mejor que el 64, e identificamos los modelos 3 y 4 como los mejores. Hacemos unas pruebas con estos y batch_size=128 a ver que ocurre.

5.4.2 Lote 2

Double-click (or enter) to edit

```
1 model = models.Sequential()
2 print("Input dimensions: ",X_train.shape[1:])
3
4 #model.add(layers.Conv2D(32, (3, 3), input_shape=X_train.shape[1:]))
5 model.add(layers.Conv2D(32, (3, 3),use_bias=False, input_shape=X_train.shape[1:]))
6 model.add(layers.BatchNormalization()) #####
7 model.add(layers.Activation('relu'))
```