

```

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:66: The name tf.get_def
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:541: The name tf.placeh
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:4432: The name tf.randc
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:2041: The name tf.nn.fv
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:148: The name tf.placeh
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:4267: The name tf.nn.ma

Downloading data from https://github.com/fchollet/deep-learning-models/releases/download/v0.4/xception\_weights\_tf\_dim\_order\_83689472/83683744 [=====] - 2s 0us/step
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:216: The name tf.is_var
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:223: The name tf.varia

=====
Experimento_XCEPTION-Dropout_4
Experimento saltado
=====
Experimento_XCEPTION-Dropout_5
Experimento saltado
=====
Experimento_XCEPTION-Dropout_4b
experimento = Experimento_XCEPTION-Dropout_4b
model = <keras.engine.training.Model object at 0x7f26f8f92fd0>
samples_per_class = 1000
number_of_classes = 102
optimizador = Adam
clasificador = XCEPTION-5
dropout = 0.4
batch_size = 128
epochs = 20
run_experiment = True
-----
Creando sub-conjunto de datos con 102 clases y 1000 muestras por clase
number_of_classes: 102
Sub-conjunto con 102 clases creado.
Cantidad de muestras: 9145
Creando datos de train, validate y test ...
Datos de train, validate y test creados.

Split de Entrenamiento, Validación y prueba: 6401, 1372, 1372
Número de clases: 102
Número de muestras: 1000
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:3733: calling dropout (
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate = 1 - keep_prob`.
Usando Adam
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/optimizers.py:793: The name tf.train.Optimizer is dep
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:3576: The name tf.log i
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tensorflow_core/python/ops/math_grad.py:1424: where (from t
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.assig
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:1020: The name tf.assig

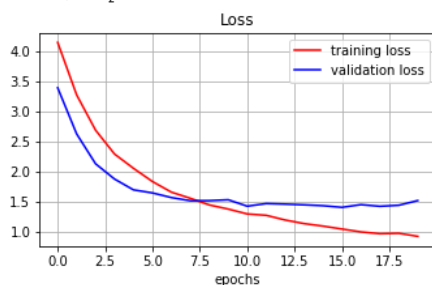
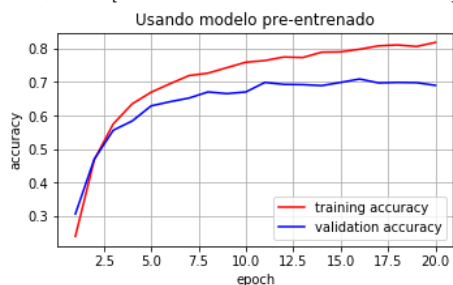
Train on 6401 samples, validate on 1372 samples
Epoch 1/20
6401/6401 [=====] - 18s 3ms/step - loss: 4.1412 - acc: 0.2393 - val_loss: 3.3875 - val_acc: 0.306
Epoch 2/20
6401/6401 [=====] - 8s 1ms/step - loss: 3.2702 - acc: 0.4679 - val_loss: 2.6221 - val_acc: 0.4701
Epoch 3/20
6401/6401 [=====] - 8s 1ms/step - loss: 2.6833 - acc: 0.5751 - val_loss: 2.1251 - val_acc: 0.5561
Epoch 4/20
6401/6401 [=====] - 8s 1ms/step - loss: 2.2871 - acc: 0.6349 - val_loss: 1.8714 - val_acc: 0.5838
Epoch 5/20
6401/6401 [=====] - 8s 1ms/step - loss: 2.0497 - acc: 0.6697 - val_loss: 1.6917 - val_acc: 0.6290
Epoch 6/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.8302 - acc: 0.6952 - val_loss: 1.6409 - val_acc: 0.6414
Epoch 7/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.6537 - acc: 0.7194 - val_loss: 1.5637 - val_acc: 0.6523
Epoch 8/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.5517 - acc: 0.7266 - val_loss: 1.5108 - val_acc: 0.6706
Epoch 9/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.4411 - acc: 0.7432 - val_loss: 1.5127 - val_acc: 0.6655
Epoch 10/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.3727 - acc: 0.7593 - val_loss: 1.5270 - val_acc: 0.6706
Epoch 11/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.2917 - acc: 0.7643 - val_loss: 1.4204 - val_acc: 0.6990
Epoch 12/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.2691 - acc: 0.7749 - val_loss: 1.4637 - val_acc: 0.6931
Epoch 13/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.1910 - acc: 0.7733 - val_loss: 1.4549 - val_acc: 0.6924
Epoch 14/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.1315 - acc: 0.7893 - val_loss: 1.4432 - val_acc: 0.6895
Epoch 15/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.0896 - acc: 0.7902 - val_loss: 1.4285 - val_acc: 0.6990
Epoch 16/20

```

```

6401/6401 [=====] - 8s 1ms/step - loss: 1.0412 - acc: 0.7982 - val_loss: 1.4014 - val_acc: 0.7092
Epoch 17/20
6401/6401 [=====] - 8s 1ms/step - loss: 0.9936 - acc: 0.8082 - val_loss: 1.4461 - val_acc: 0.6975
Epoch 18/20
6401/6401 [=====] - 8s 1ms/step - loss: 0.9635 - acc: 0.8110 - val_loss: 1.4188 - val_acc: 0.6990
Epoch 19/20
6401/6401 [=====] - 8s 1ms/step - loss: 0.9701 - acc: 0.8063 - val_loss: 1.4359 - val_acc: 0.6983
Epoch 20/20
6401/6401 [=====] - 8s 1ms/step - loss: 0.9197 - acc: 0.8185 - val_loss: 1.5140 - val_acc: 0.6902

```



Exactitud en subconjunto de test:
Predict loss: 1.5047658879972408
Predict accuracy: 0.6581632653061225

Exactitud en todo el dataset:
Predict loss: 1.3147914073791531
Predict accuracy: 0.7094587206449451

```

=====
Experimento_XCEPTION-Dropout_5b
experimento = Experimento_XCEPTION-Dropout_5b
model = <keras.engine.training.Model object at 0x7f26f8f92fd0>
samples_per_class = 1000
number_of_classes = 102
optimizador = Adam
clasificador = XCEPTION-5
dropout = 0.5
batch_size = 128
epochs = 20
run_experiment = True

```

Número de clases: 102
Número de muestras: 1000

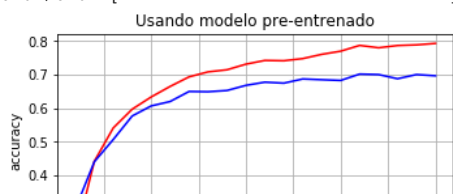
Usando Adam

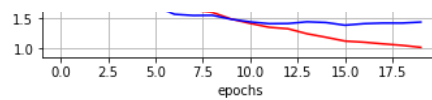
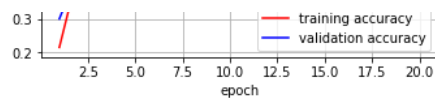
Train on 6401 samples, validate on 1372 samples

```

Epoch 1/20
6401/6401 [=====] - 11s 2ms/step - loss: 4.1834 - acc: 0.2156 - val_loss: 3.4469 - val_acc: 0.300
Epoch 2/20
6401/6401 [=====] - 8s 1ms/step - loss: 3.3983 - acc: 0.4416 - val_loss: 2.7250 - val_acc: 0.4402
Epoch 3/20
6401/6401 [=====] - 8s 1ms/step - loss: 2.8424 - acc: 0.5409 - val_loss: 2.2930 - val_acc: 0.5066
Epoch 4/20
6401/6401 [=====] - 8s 1ms/step - loss: 2.4467 - acc: 0.5968 - val_loss: 1.9260 - val_acc: 0.5765
Epoch 5/20
6401/6401 [=====] - 8s 1ms/step - loss: 2.1814 - acc: 0.6330 - val_loss: 1.7882 - val_acc: 0.6064
Epoch 6/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.9769 - acc: 0.6644 - val_loss: 1.6990 - val_acc: 0.6195
Epoch 7/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.8009 - acc: 0.6929 - val_loss: 1.5631 - val_acc: 0.6494
Epoch 8/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.6379 - acc: 0.7077 - val_loss: 1.5428 - val_acc: 0.6487
Epoch 9/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.5948 - acc: 0.7140 - val_loss: 1.5456 - val_acc: 0.6523
Epoch 10/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.4845 - acc: 0.7307 - val_loss: 1.4813 - val_acc: 0.6676
Epoch 11/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.4135 - acc: 0.7418 - val_loss: 1.4381 - val_acc: 0.6771
Epoch 12/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.3505 - acc: 0.7408 - val_loss: 1.4092 - val_acc: 0.6742
Epoch 13/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.3259 - acc: 0.7472 - val_loss: 1.4116 - val_acc: 0.6866
Epoch 14/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.2449 - acc: 0.7599 - val_loss: 1.4398 - val_acc: 0.6844
Epoch 15/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.1872 - acc: 0.7693 - val_loss: 1.4278 - val_acc: 0.6822
Epoch 16/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.1256 - acc: 0.7864 - val_loss: 1.3840 - val_acc: 0.7012
Epoch 17/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.1070 - acc: 0.7794 - val_loss: 1.4094 - val_acc: 0.6997
Epoch 18/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.0802 - acc: 0.7861 - val_loss: 1.4191 - val_acc: 0.6873
Epoch 19/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.0562 - acc: 0.7880 - val_loss: 1.4187 - val_acc: 0.6997
Epoch 20/20
6401/6401 [=====] - 8s 1ms/step - loss: 1.0235 - acc: 0.7924 - val_loss: 1.4353 - val_acc: 0.6961

```





Exactitud en subconjunto de test:
 Predict loss: 1.432564815348856
 Predict accuracy: 0.6676384837043529

Exactitud en todo el dataset:
 Predict loss: 1.267198227749839
 Predict accuracy: 0.7131765992801785

Resultados XCEPTION

Lote 1

[Ver PDF con el output de pruebas del lote 1](#)

Las gráficas usando **rmsprop** muestran un overfitting mientras que con **Adam** parecieran que se puede mejorar con mas epocas, usaremos estos datos mas adelante.

Experimento	Muestras*Clase	Optimizador	Clasificador	Batch Size	epocas	Tiempo Entrenamiento	Exac. Test	Exact. Full	Loss Test	Loss Full
XCEPTION-1	30	rmsprop	XCEPTION-1	128	10	35	56%	48%	2.84	3.39
XCEPTION-2	30	rmsprop	XCEPTION-2	128	10	35	61%	55%	2.83	2.79
XCEPTION-3	30	rmsprop	XCEPTION-3	128	10	35	41%	34%	5.47	5.71
XCEPTION-4	30	Adam	XCEPTION-1	128	10	36	51%	42%	2.44	2.86
XCEPTION-5	30	Adam	XCEPTION-2	128	10	37	54%	50%	2.04	2.38
XCEPTION-6	30	Adam	XCEPTION-3	128	10	38	59%	54%	1.79	2.13

Lote 2

[Ver PDF con el output de pruebas del lote 2](#)

Experimento	Muestras*Clase	Optimizador	Clasificador	Batch Size	epocas	Tiempo Entrenamiento	Exac. Test	Exact. Full	Loss Test	Loss Full
XCEPTION-7	100	rmsprop	XCEPTION-1	128	10	72	64%	64%	2.35	2.26
XCEPTION-8	100	rmsprop	XCEPTION-2	128	10	73	67%	66%	2.33	2.24
XCEPTION-9	100	rmsprop	XCEPTION-3	128	10	74	65%	66%	2.67	2.46
XCEPTION-10	100	Adam	XCEPTION-1	128	10	74	68%	66%	1.49	1.60
XCEPTION-11	100	Adam	XCEPTION-2	128	10	75	69%	69%	1.32	1.43
XCEPTION-12	100	Adam	XCEPTION-3	128	10	76	69%	69%	1.27	1.33

Lote 3

[Ver PDF con el output de pruebas del lote 3](#)

Experimento	Muestras*Clase	Optimizador	Clasificador	Batch Size	epocas	Tiempo Entrenamiento	Exac. Test	Exact. Full	Loss Test	Loss Full
XCEPTION-8a	100	rmsprop	XCEPTION-2	128	4	45	67%	66%	1.78	1.72
XCEPTION-8b	100	rmsprop	XCEPTION-2	128	8	70	68%	66%	1.96	2.10
XCEPTION-8c	100	rmsprop	XCEPTION-2	128	14	107	66%	65%	2.70	2.67
XCEPTION-11a	100	Adam	XCEPTION-3	64	15	114	71%	70%	1.34	1.45
XCEPTION-11b	100	Adam	XCEPTION-3	128	15	115	71%	71%	1.25	1.30
XCEPTION-11c	100	Adam	XCEPTION-3	256	15	102	71%	71%	1.23	1.28

En este lote de pruebas queria verificar que la red con optimizador **rmsprop** no iba a mejorar con mas épocas y efectivamente llega a un aproximado de precisión de 68% en mis experimentos.

Por otro lado creo que con Adam las curvas se ven mas prometedoras, en este caso queria verificar si habia algún cambio significativo cambiando el **batch size** pero como se observa no causa grandes cambios, por lo que lo dejaremos en 128 como el resto de los experimentos.

Lote 4

[Ver PDF con el output de pruebas del lote 4](#)

La red mas prometedora es la basada en el optimizador **Adam** pero las gráficas sugieren que no va a mejorar entrenandola mas épocas. Vamos a probar con 20 epocas y también ver como se comporta con todo el juego de datos.

Experimento	Muestras*Clase	Optimizador	Clasificador	Batch Size	epocas	Tiempo Entrenamiento	Exac. Test	Exact. Full	Loss Test	Loss Full
XCEPTION-12a	100	Adam	XCEPTION-3	128	20	110	70%	71%	1.35	1.36
XCEPTION-13a	1000	Adam	XCEPTION-3	128	15	115	65%	70%	1.60	1.39