

## Inteligencia Artificial - Práctica 5

### Eduardo Gimeno 721615

Perceptrón						
Función de activación	Algoritmo de entrenamiento	Función de coste	Épocas	Training Time	Accuracy Train (%)	Accuracy Test (%)
Sigmoid	SGD	categorical_crossentropy	20	19.089 s	90.00	90.68
Sigmoid	RMSprop	categorical_crossentropy	20	20.042 s	93.22	92.73
Sigmoid	Adam	categorical_crossentropy	20	21.004 s	93.27	92.74
Softmax	SGD	categorical_crossentropy	20	19.626 s	90.16	90.82
Softmax	RMSprop	categorical_crossentropy	20	20.584 s	93.22	92.70
Softmax	Adam	categorical_crossentropy	20	21.261 s	93.27	92.78
Sigmoid	SGD	mean_squared_error	20	16.447 s	77.15	78.20
Sigmoid	RMSprop	mean_squared_error	20	17.163 s	92.38	92.38
Sigmoid	Adam	mean_squared_error	20	17.780 s	92.35	92.34
Softmax	SGD	mean_squared_error	20	16.657 s	76.42	77.82
Softmax	RMSprop	mean_squared_error	20	17.582 s	93.78	93.20
Softmax	Adam	mean_squared_error	20	18.207 s	94.04	93.16
Sigmoid	Adam	categorical_crossentropy	30	31.566 s	93.46	92.77
Softmax	Adam	categorical_crossentropy	30	32.249 s	93.44	92.72
Sigmoid	RMSprop	mean_squared_error	30	25.696 s	92.58	92.53
Softmax	RMSprop	mean_squared_error	30	26.084 s	94.06	93.17

Se han probado todas las combinaciones posibles teniendo en cuenta como función de activación sigmoid o softmax, como algoritmo de entrenamiento SGD, RMSprop y Adam y como función de coste categorical\_crossentropy y mean\_squared\_error.

Una vez realizadas las distintas pruebas, se han seleccionado los cuatro mejores resultados y se ha probado a aumentar el número de épocas para obtener un posible mejor resultado, pero no ha habido mejora.

De esta red se obtiene como mejor resultado la siguiente combinación, como función de activación, softmax, algoritmo de entrenamiento, RMSprop y como función de coste mean\_squared\_error. Obteniendo un 93,20 % de accuracy.

En ningún momento ha sido aplicada ninguna técnica de corrección de sobreajuste.

Como base para construir las siguientes redes se han tomado las combinaciones que generan los dos mejores resultados.

Perceptrón multi-nivel con una capa oculta										
Capa oculta		Capa salida								
Función de activación	Neuronas	Función de activación	Neuronas	Función de coste	Algoritmo de entrenamiento	Épocas	Training Time	Accuracy Train (%)	Accuracy Test (%)	
Sigmoid	10	Sigmoid	10	Categorical_crossentropy	Adam	20	22.319 s	92.94	92.54	
Sigmoid	500	Sigmoid	10	Categorical_crossentropy	Adam	20	197.278 s	99.69	97.94	
Relu	10	Sigmoid	10	Categorical_crossentropy	Adam	20	22.438 s	93.86	93.16	
Relu	500	Sigmoid	10	Categorical_crossentropy	Adam	20	196.220 s	99.79	98.09	
Sigmoid	10	Softmax	10	Categorical_crossentropy	Adam	20	22.728 s	93.51	92.85	
Sigmoid	500	Softmax	10	Categorical_crossentropy	Adam	20	196.424 s	99.69	97.98	
Relu	10	Softmax	10	Categorical_crossentropy	Adam	20	22.896 s	94.24	93.50	
Relu	500	Softmax	10	Categorical_crossentropy	Adam	20	197.611 s	99.59	97.78	
Sigmoid	10	Sigmoid	10	Mean_squared_error	RMSprop	20	18.410 s	92.09	91.86	
Sigmoid	500	Sigmoid	10	Mean_squared_error	RMSprop	20	158.527 s	99.21	98.09	
Relu	10	Sigmoid	10	Mean_squared_error	RMSprop	20	18.060 s	92.63	92.59	
Relu	500	Sigmoid	10	Mean_squared_error	RMSprop	20	161.530 s	99.58	98.43	
Sigmoid	10	Softmax	10	Mean_squared_error	RMSprop	20	18.591 s	93.56	93.19	
Sigmoid	500	Softmax	10	Mean_squared_error	RMSprop	20	160.237 s	99.36	97.98	
Relu	10	Softmax	10	Mean_squared_error	RMSprop	20	18.291 s	94.67	93.75	
Relu	500	Softmax	10	Mean_squared_error	RMSprop	20	161.000 s	99.72	98.18	
Relu	250	Sigmoid	10	Categorical_crossentropy	Adam	20	102.805 s	99.77	97.88	
Sigmoid	250	Sigmoid	10	Mean_squared_error	RMSprop	20	88.368 s	98.88	97.80	
Relu	250	Sigmoid	10	Mean_squared_error	RMSprop	20	87.517 s	99.47	98.14	
Relu	250	Softmax	10	Mean_squared_error	RMSprop	20	87.816 s	99.64	98.00	
Relu	250	Sigmoid	10	Categorical_crossentropy	Adam	40	98.525 s	99.78	98.05 (98.11)	Earllystop(19)
							219.039 s	99.83	98.28	Dropout(0.2)
Relu	250	Softmax	10	Mean_squared_error	RMSprop	40	147.370 s	99.70	98.14 (98.02)	Earllystop(34)
							187.101 s	99.68	98.17	Dropout(0.2)

Para la construcción de esta red se utiliza como base las combinaciones que generan los mejores resultados de la red anterior.

Se realizan todas las combinaciones posibles partiendo de dicha base y siguiendo el siguiente proceso para la nueva capa añadida.

En primer lugar se prueba con 10 neuronas en dicha capa, se ha utilizado este valor al igual que en la capa de salida, y en segundo lugar se prueba con 500 neuronas, se realiza una prueba de extremos.

Una vez realizadas todas las combinaciones pertinentes, se seleccionan los cuatro mejores resultados y se pasa a realizar una segunda fase las combinaciones que generan dichos resultados. Se reduce el número de neuronas de 500 a la mitad con el fin de reducir el tiempo de entrenamiento y

que no sea tan elevado, ya que en la siguiente red que se va a construir se añadirá una capa más. Al finalizar esta prueba se seleccionan los dos mejores resultados, en función de la pérdida que han tenido en accuracy, es decir, las dos combinaciones que menos pérdida han tenido en accuracy al reducir el número de neuronas en la capa oculta.

Por último, se incrementa el número de épocas a 40 para obtener mejores resultados. Además se aplican las técnicas de corrección de sobreajuste earlystop y dropout.

Se obtiene como mejor resultado la combinación, función de activación de la capa oculta relu, de la capa de salida sigmoid, función de coste categorical\_crossentropy y algoritmo de entrenamiento Adam, con un 98,20 % de accuracy.

Esta red servirá de base para la última red ha desarrollar.

Perceptrón multi-nivel con dos capas ocultas											
Capa oculta 1		Capa oculta 2		Capa salida							
Función de activación	Neuronas	Función de activación	Neuronas	Función de activación	Neuronas	Función de coste	Algoritmo de entrenamiento	Épocas	Training Time	Accuracy Train (%)	Accuracy Test (%)
Sigmoid	10	Relu	250	Sigmoid	10	Categorical_crossentropy	Adam	40	75.309 s	96.55	95.03
Sigmoid	500	Relu	250	Sigmoid	10	Categorical_crossentropy	Adam	40	544.263 s	99.77	98.08
Relu	10	Relu	250	Sigmoid	10	Categorical_crossentropy	Adam	40	75.028 s	97.47	95.94
Relu	500	Relu	250	Sigmoid	10	Categorical_crossentropy	Adam	40	540.463 s	99.76	98.28
Relu	250	Relu	250	Sigmoid	10	Categorical_crossentropy	Adam	40	286.393 s	99.77	98.05
Relu	250	Relu	250	Sigmoid	10	Categorical_crossentropy	Adam	60	447.463 s	99.81	98.33 (98.31)

Para la construcción de esta red, se utiliza como base la mejor combinación obtenida en la red anterior.

Partiendo de dicha base, se prueban todas las combinaciones posibles siguiendo la estrategia de la red anterior, en primer lugar con 10 neuronas en la nueva capa oculta y en segundo lugar con 500, además se mantienen las 40 épocas con las que se obtuvo el mejor resultado en la red anterior.

Se selecciona el mejor resultado y se reduce el número de neuronas de la nueva capa oculta a la mitad para reducir el tiempo de entrenamiento.

Por último se incrementan el número de épocas a 60 para mejorar el resultado y aplicando la técnica de dropout para reducir el sobreajuste, se utiliza esta técnica porque dio los mejores resultados en la red anterior.

Finalmente se obtiene como mejor combinación:

- Función de activación
  - Capa oculta 1: relu
  - Capa oculta 2: relu
  - Capa de salida: sigmoid
- Función de coste: categorical\_crossentropy
- Algoritmo de entrenamiento: Adam
- Épocas: 60
- Técnica sobreajuste: Dropout(0.2)
- Accuracy: 98.33 %

Red Convolucional	
Accuracy Test (%)	99.16

En último lugar se prueba la red convolucional dada, obteniendo el mejor resultado de la práctica, un 99.16 de accuracy, además del mayor tiempo de entrenamiento, 15 minutos aproximadamente.

Con esta red se obtiene el mejor resultado ya que es un tipo de red la cual cada una de sus capas aprende un nivel de abstracción, son usadas comúnmente para la categorización de objetos.

Si el objetivo de esta práctica fuese que la mejor red desarrollada fuese utilizada para un propósito real, se elegiría esta última ya que es, con diferencia, la que mejor accuracy proporciona para este tipo de problema, podría tener como desventaja su tiempo de entrenamiento pero si tuviese un gran uso con un 99.16 % tardaría más en aparecer un fallo que con un 98.33 % (red multi-nivel con dos capas ocultas).

En apartados anteriores se ha buscado reducir el tiempo de entrenamiento y luego recuperar accuracy introduciendo más épocas.