

Índice:

Prólogo	11
Prefacio	13
Introducción	17
<i>A quién va dirigido el libro</i>	17
<i>Organización del libro</i>	18
<i>Requisitos para seguir el libro</i>	20
<i>¿Por qué Keras?</i>	21
1 Ante una nueva tecnología disruptiva	23
1.1 La inteligencia artificial está cambiando nuestras vidas	25
1.2 Inteligencia artificial, Machine Learning y Deep Learning	26
<i>Inteligencia artificial</i>	27
<i>Machine Learning</i>	28
<i>Redes neuronales artificiales y Deep Learning</i>	30
1.3 ¿Por qué ahora?	33
<i>Los datos, el combustible para la inteligencia artificial</i>	33
<i>Nuevo hardware masivamente paralelo</i>	35
<i>Democratización de la computación</i>	37
<i>Un mundo open-source para la comunidad Deep Learning</i>	39
<i>Una cultura de publicación abierta</i>	41
<i>Mejoras en los algoritmos</i>	42
1.4 Preparar entorno de trabajo	42
<i>Jupyter notebook</i>	42
<i>Keras</i>	43
<i>Docker</i>	44
<i>Cloud</i>	46
2 Redes neuronales densamente conectadas	49
2.1 Caso de estudio: reconocimiento de dígitos escritos a mano	51
2.2 Perceptron	54
<i>Algoritmos de regresión</i>	54
<i>Una neurona artificial simple</i>	55
2.3 Multi-Layer Perceptron	59
<i>Softmax</i>	60
<i>Evidencia de pertenencia</i>	61
<i>Probabilidades de pertenencia</i>	65

2.4	Datos para alimentar una red neuronal	66
	<i>Conjunto de datos para entrenamiento, validación y prueba</i>	66
	<i>Precarga de los datos en Keras</i>	67
	<i>Representación de los datos en Keras</i>	67
	<i>Normalización de los datos de entrada</i>	69
2.5	Redes densamente conectadas en Keras	71
	<i>Clase Sequential en Keras</i>	71
	<i>Capas fully conectad de la red neuronal</i>	72
2.6	Pasos para implementar una red neuronal en Keras	75
	<i>Definición del modelo</i>	75
	<i>Configuración del proceso de aprendizaje</i>	75
	<i>Entrenamiento del modelo</i>	77
	<i>Evaluación del modelo</i>	78
	<i>Generación de predicciones</i>	81
3	Como se entrena una red neuronal	84
3.1	Proceso de aprendizaje de una red neuronal	86
3.2	Funciones de activación	89
	<i>Linear</i>	89
	<i>Sigmoid</i>	90
	<i>Tanh</i>	91
	<i>Softmax</i>	91
	<i>ReLU</i>	92
3.3	Elementos del backpropagation	92
	<i>Función de loss</i>	93
	<i>Optimizadores</i>	94
	<i>Gradient descent</i>	94
	<i>Stochastic Gradient Descent (SGD)</i>	95
3.4	Parametrización de los modelos	97
	<i>Diferencia entre parámetros e hiperparámetros</i>	98
	<i>Número de epochs</i>	99
	<i>Batch size</i>	99
	<i>Learning Rate</i>	100
	<i>Learning Rate Decay</i>	101
	<i>Momentum</i>	101
	<i>Inicialización de los pesos de los parámetros</i>	103
	<i>Hiperparámetros y optimizadores en Keras</i>	103
3.5	Ejemplo para una clasificación binaria	104
	<i>TensorFlow Playground</i>	105
	<i>Clasificación con una sola neurona</i>	109
	<i>Clasificación con más de una neurona</i>	110
	<i>Clasificación con varias capas</i>	112

4	Los datos y su preparación	116
4.1	Datasets para entrenar las redes <i>Deep Learning</i>	118
	<i>Datasets públicos disponibles para Deep Learning</i>	118
	<i>Datasets precargados en Keras</i>	120
	<i>Caso de estudio: Conjunto de datos IMDB</i>	121
4.2	Preprocesado de datos para redes neuronales	124
	<i>Normalización de los valores</i>	124
	<i>Gestionar la falta de algunos valores</i>	125
	<i>Vectorización</i>	126
4.3	One-hot encode	126
	<i>Ejemplo de código de one-hot encode</i>	127
	<i>Clasificación binaria con one-hot encoding</i>	128
4.4	Word Embedding	132
	<i>Limitaciones one-hot encoding</i>	132
	<i>Introducción al word embedding</i>	133
	<i>Capa embedding</i>	134
	<i>Clasificación binaria con word embedding</i>	136
	<i>Usando word embedding preentrenado</i>	138
	<i>Clasificación binaria con word embedding preentrenado</i>	140
5	Prevención del Overfitting	144
5.1	Overfitting y underfitting	145
5.2	Abordar el sobreajuste a través de curvas de validación	147
	<i>Acceso al histórico del entrenamiento</i>	150
	<i>Visualización del comportamiento del entrenamiento</i>	151
	<i>Gráfico de complejidad del modelo</i>	153
	<i>Early stopping en Keras</i>	155
5.3	Obtener mas datos	156
5.4	Reducir el tamaño de la red neuronal	157
5.5	Añadir regularización de los pesos	160
5.6	Dropout	164
6	Redes neuronales convolucional	168
6.1	Representación de la información en las neuronas artificiales	169
6.2	Componentes básicos de una red neuronal convolucional	171
	<i>Convolución</i>	172
	<i>Pooling</i>	177
6.3	Implementación de un modelo básico	179
	<i>Capa base de una red neuronal convolucional</i>	179
	<i>Un modelo simple</i>	180
	<i>Entrenamiento y evaluación del modelo</i>	183
	<i>Argumentos del método fit</i>	185

6.4	Parámetros de la capa convolucional	185
	<i>Tamaño de ventana y número de filtros</i>	185
	<i>Padding</i>	186
	<i>Strides</i>	188
7	Que hacer cuando hay pocos datos	190
7.1	Caso de estudio: modelo para clasificar perros y gatos	191
	<i>Descarga y preparación de los datos</i>	191
	<i>Creación del modelo</i>	196
	<i>Aspectos prácticos: formato HDF5 y módulo Pickle</i>	201
	<i>Validación del entrenamiento</i>	202
7.2	Data Augmentation	206
7.3	Transfer Learning	210
7.4	Feature extraction	212
	<i>Fast feature extraction sin data augmentation</i>	216
	<i>Feature extraction con Data Augmentation</i>	220
7.5	Fine-tuning	225
8	Redes neuronales recurrentes	227
8.1	Conceptos básicos de redes neuronal recurrente	228
	<i>Neurona recurrente</i>	229
	<i>Memory cell</i>	231
	<i>Backpropagation a través del tiempo</i>	232
	<i>Exploding and Vanishing Gradients</i>	233
	<i>Long-Short Term Memory</i>	234
8.2	Implementación en Keras	235
	<i>Caso de estudio: IMDB</i>	235
	<i>Capas recurrentes en Keras</i>	236
	<i>Implementación con una capa GRU</i>	237
	<i>Recurrent dropout para combatir el sobreentrenamiento</i>	240
	Clausura	247
	<i>¿Qué nos dejamos en el tintero?</i>	247
	<i>¿Moda o ha venido para quedarse?</i>	248
	<i>¿Y nosotros, los que “programamos”?</i>	250
	Agradecimientos	252
	Acerca del autor	254