1. Record Nr. UNINA9910300755003321

Autore	Michelucci Umberto
Titolo	Applied Deep Learning : A Case-Based Approach to Understanding Deep Neural Networks / / by Umberto Michelucci
Pubbl/distr/stampa	Berkeley, CA : , : Apress : , : Imprint : Apress, , 2018
ISBN	9781484237908 1484237900
Edizione	[1st ed. 2018.]
Descrizione fisica	1 online resource (425 pages)
Disciplina	006.31
Soggetti	Artificial intelligence Python (Computer program language) Open source software Computer programming Big data Artificial Intelligence Python Open Source Big Data
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di contenuto	Chapter 1: Introduction -- Chapter 2: Single Neurons -- Chapter 3: Fully connected Neural Network with more neurons -- Chapter 4: Neural networks error analysis -- Chapter 5: Dropout technique -- Chapter 6: Hyper parameters tuning -- Chapter 7: Tensorflow and optimizers (Gradient descent, Adam, momentum, etc.) -- Chapter 8: Convolutional Networks and image recognition -- Chapter 9: Recurrent Neural Networks -- Chapter 10: A practical COMPLETE example from scratch (put everything together) -- Chapter 11: Logistic regression implement from scratch in Python without libraries. .
Sommario/riassunto	Work with advanced topics in deep learning, such as optimization algorithms, hyper-parameter tuning, dropout, and error analysis as well as strategies to address typical problems encountered when training deep neural networks. You’ll begin by studying the activation

functions mostly with a single neuron (ReLu, sigmoid, and Swish), seeing how to perform linear and logistic regression using TensorFlow, and choosing the right cost function. The next section talks about more complicated neural network architectures with several layers and neurons and explores the problem of random initialization of weights. An entire chapter is dedicated to a complete overview of neural network error analysis, giving examples of solving problems originating from variance, bias, overfitting, and datasets coming from different distributions. Applied Deep Learning also discusses how to implement logistic regression completely from scratch without using any Python library except NumPy, to let you appreciate how libraries such as TensorFlow allow quick and efficient experiments. Case studies for each method are included to put into practice all theoretical information. You'll discover tips and tricks for writing optimized Python code (for example vectorizing loops with NumPy). You will: Implement advanced techniques in the right way in Python and TensorFlow Debug and optimize advanced methods (such as dropout and regularization) Carry out error analysis (to realize if one has a bias problem, a variance problem, a data offset problem, and so on) Set up a machine learning project focused on deep learning on a complex dataset.