

Bibliography

- [1] DONALD HEBB. *The Organization of Behavior*.
- [2] ANDREAS KAPLAN, MICHAEL HAENLEIN. *Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence* (<https://doi.org/10.1016%2Fj.bushor.2018.08.004>).
- [3] *A Brief History of Computing* (www.alanturing.net/turing_archive/pages/Reference%20Articles/BriefHistofComp.html).
- [4] *Raspberry Pi benchmarks* (www.researchgate.net/publication/333973011_Raspberry_Pi_4B_32_Bit_Benchmarks).
- [5] *IBM704* (https://en.wikipedia.org/wiki/IBM_704).
- [6] H. A. SIMON, ALLEN NEWELL. *Heuristic Problem Solving: The Next Advance in Operations Research* (<https://doi.org/10.1287%2Fopre.6.1.1>).
- [7] H. A. SIMON. *The Shape of Automation for Men and Management*.
- [8] M. MINSKY. *Computation: Finite and Infinite Machines*.
- [9] M. MINSKY. *Meet Shaky, the First Electronic Person, Life Magazine*, pp. 58–68.

BIBLIOGRAPHY

- [10] *rmsprop optimizer description* (www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf).
- [11] S. ZHANG, A.E. CHOROMANSKA, Y. LECUN. *Deep learning with elastic averaging SGD* (<https://papers.nips.cc/paper/5761-deep-learning-with-elastic-averaging-sgd.pdf>).
- [12] D. KINGMA, J. BA. *Adam: A Method for Stochastic Optimization* (<https://arxiv.org/abs/1412.6980v8>).
- [13] I. SUTSKEVER, J. MARTENS, G. DAHL, G. HINTON. *On the importance of initialization and momentum in deep learning* (www.cs.toronto.edu/~fritz/absps/momentum.pdf).
- [14] *Auto MPG dataset* (<https://archive.ics.uci.edu/ml/datasets/auto+mpg>).
- [15] *Moore-Penrose inverse* (https://en.wikipedia.org/wiki/Moore%E2%80%93Penrose_inverse).
- [16] JUDITH A HIRSCH, LUIS M MARTINEZ. *Visual Cortical and Subcortical Receptive Fields* (https://link.springer.com/referenceworkentry/10.1007%2F978-3-540-29678-2_6348).
- [17] JUDITH A HIRSCH, LUIS M MARTINEZ. *Circuits that build visual cortical receptive fields* (<https://pubmed.ncbi.nlm.nih.gov/16309753/>).
- [18] KEVIN R. DUFFY, DAVID H. HUBEL. *Receptive field properties of neurons in the primary visual cortex under photopic and scotopic lighting conditions* (www.ncbi.nlm.nih.gov/pmc/articles/PMC2951600/).

- [19] SUMIT SAHA. *A Comprehensive Guide to Convolutional Neural Networks—the ELI5 way* (<https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>).
- [20] IRWIN SOBEL. *An Isotropic 3x3 Image Gradient Operator* (www.researchgate.net/publication/239398674_An_Isotropic_3_3_Image_Gradient_Operator).
- [21] F. MOUTARDE, G. DEVINEAU. *Deep-Learning: Introduction to Convolutional Neural Networks* (http://people.mines-paristech.fr/fabien.moutarde/ES_MachineLearning/Practical_deepLearning-convNets/convnet-notebook.html).
- [22] *Fruits 360 Kaggle dataset* (www.kaggle.com/moltean/fruits).
- [23] *Raspberry Pi home page* (www.raspberrypi.org).
- [24] *Pimoroni MLX90640 breakout* (<https://shop.pimoroni.com/products/mlx90640-thermal-camera-breakout>).
- [25] *Breakout Garden* (<https://shop.pimoroni.com/products/breakout-garden-hat-i2c-spi>).
- [26] *NOOBS* (www.raspberrypi.org/downloads/noobs/).
- [27] *Raspberry Pi OS* (www.raspberrypi.org/downloads/).
- [28] *Platypush Gitlab page* (<https://git.platypush.tech/platypush/platypush>).
- [29] *Platypush modules documentation* (<https://platypush.readthedocs.io/en/latest/>).

Index

A

Accuracy, 8, 33, 111, 113, 126
Activation function, 90, 91, 93, 94,
101, 153
Artificial intelligence, 2–6, 90
Artificial neural network, 90, 93
AutoLocation, 209, 211
Average pooling, 144, 145

B

Backends, 171, 178
Back-propagation, 6, 96–98, 102

C

categorical_crossentropy loss
function, 32, 154
Classification error, 73, 97, 111
Convolutional layer, 135, 136, 137,
142–145
Convolutional neural networks
(CNN), 134
architecture, 135
convolutional layers, 138, 140,
142, 143
features, 135
fully connected/dropout, 145

kernel/filter, 136
pooling layer, 144
recognition, 146–148
training epochs, 155
training set, 150

Cost function, 21, 23, 24, 26,
32, 42, 63, 64, 73, 75,
96–98, 101

Cross-entropy functions, 125, 126

D

Deep learning, 8, 32
Dropout technique, 146

E

Eigenvalue, 48–50
Expert system, 2, 7, 8

F

False negatives (FN), 112, 126
False positives (FP), 112, 115, 184
fit_generator method, 154

G

Grayscale thermal picture, 181

INDEX

H

- Hadamard product, 101, 139
- Hardware I²C connection, 164
- Harmonic mean, 115

I, J

- Infrared cameras, 120,
160–162, 176

K

- kernel_size parameter, 153

L

- Linear regression, 17
 - defining/training, 30–33
 - evaluation, 34–36, 38
 - gradient descent, 22–24, 26, 27
 - idea behind regression, 19–21
 - input normalization, 28, 29
 - loading/plotting
 - dataset, 18, 19
 - saving/loading, 38
- Logistic regression, 67, 69, 70, 72
 - building from scratch,
75, 76, 78–81
 - cost function, 72, 74
 - multiclass regression, 84, 85
 - non-linear boundaries, 85, 86
 - TensorFlow, 82–84
- loss function, 21, 32, 35, 57,
58, 125, 221

M

- Machine learning, 120
 - definition, 1
 - environment, 13–15, 17
 - history, 3–5
 - tools, 12, 13
- Max pooling, 144, 145
- Mean absolute error, 32, 33, 111
- Mean squared error, 32, 34, 42, 61,
72, 111, 125
- Moravec’s paradox, 7
- Multivariate linear regression,
41, 42, 44, 45
 - loading/visualizing dataset,
53, 55–59
 - PCA, 47, 49–51
 - redundant features, 45–47
 - training/test set, 52, 53

N, O

- Neural network
 - accuracy, 129
 - color space, 119
 - confusion matrix, 113
 - cost function, 106, 107
 - error metrics, 111
 - Fashion MNIST
 - dataset, 118, 122
 - guidelines, 104, 105
 - implement, 116
 - input features, 88
 - intrusions, 115
 - label, 130–132

- optical camera, 120
- output, 109, 110
- predicted values, 112
- underfit/overfit, 106
- values, 108

Normal equation, 64, 65, 67

P, Q

- PiCamera-compatible infrared camera, 174
- PiCamera-compatible optical camera, 174
- Pimoroni MLX90640 thermal camera, 162
- plot_results function, 157
- Polynomial regression, 60–63
- Pooling layer, 135, 137, 144, 145, 153, 154
- pool_size parameter, 153
- predict function, 36, 37
- Principal component analysis (PCA), 47, 49, 51, 107, 137, 146

R

- Raspberry Pi
 - automation flows, 205, 206
 - backends, 171
 - capture images, 179–181
 - classifiers, 219, 221, 224
 - communication protocol, 163
 - deploy model, 195, 197

- OpenCV, 197, 198, 201
- TensorFlow way, 201
- GPIO, 163, 166
- home surveillance system, 208–211, 213–215
- label images, 182, 184, 185
- live training, 216, 218, 219
- MLX90640, 172
- network, 168
- Pimoroni, 172
- Platpush automation
 - capture URL, 177
 - manual restart, 175
 - plugin, 174
 - web interface, 174
- preparing hardware, 161, 163
- preparing operating system, 166, 167
- Redis, 170
- semi-supervised learning, 216, 218, 219
- software, 169
- test set, 194
- training model, 185, 186
- training set, 190
- utility functions, 192
- VCC/GND, 164
- Recommender system, 10

S

- Same padding, 143, 144
- Scalar/dot product, 48
- Sigmoid function, 68, 69, 75, 101

INDEX

Simple regression model, 17
softmax activation function, 124
Software I²C connection, 165
Stochastic gradient descent (SGD),
 31, 83, 125
Supervised learning, 9
System-on-chip (SoC), 159

T

TensorFlow/Keras models, 197
TensorFlow regression models, 132

`tensorflow.trainmethod`, 216, 221
True negatives (TN), 112
True positives (TP), 112
Twilio integrations, 213

U

Unsupervised learning, 9–11

V, W, X, Y, Z

Valid padding, 143