

# Table of Contents

Preface.....	xiii
--------------	------

## Part I. The Fundamentals of Machine Learning

<b>1. The Machine Learning Landscape.....</b>	<b>3</b>
What Is Machine Learning?	4
Why Use Machine Learning?	4
Types of Machine Learning Systems	7
Supervised/Unsupervised Learning	8
Batch and Online Learning	14
Instance-Based Versus Model-Based Learning	17
Main Challenges of Machine Learning	22
Insufficient Quantity of Training Data	22
Nonrepresentative Training Data	24
Poor-Quality Data	25
Irrelevant Features	25
Overfitting the Training Data	26
Underfitting the Training Data	28
Stepping Back	28
Testing and Validating	29
Exercises	31
<b>2. End-to-End Machine Learning Project.....</b>	<b>33</b>
Working with Real Data	33
Look at the Big Picture	35
Frame the Problem	35
Select a Performance Measure	37

Check the Assumptions	40
Get the Data	40
Create the Workspace	40
Download the Data	43
Take a Quick Look at the Data Structure	45
Create a Test Set	49
Discover and Visualize the Data to Gain Insights	53
Visualizing Geographical Data	53
Looking for Correlations	56
Experimenting with Attribute Combinations	59
Prepare the Data for Machine Learning Algorithms	60
Data Cleaning	61
Handling Text and Categorical Attributes	63
Custom Transformers	65
Feature Scaling	66
Transformation Pipelines	66
Select and Train a Model	68
Training and Evaluating on the Training Set	69
Better Evaluation Using Cross-Validation	70
Fine-Tune Your Model	72
Grid Search	72
Randomized Search	75
Ensemble Methods	75
Analyze the Best Models and Their Errors	75
Evaluate Your System on the Test Set	76
Launch, Monitor, and Maintain Your System	77
Try It Out!	77
Exercises	78
<b>3. Classification.....</b>	<b>79</b>
MNIST	79
Training a Binary Classifier	82
Performance Measures	82
Measuring Accuracy Using Cross-Validation	83
Confusion Matrix	84
Precision and Recall	86
Precision/Recall Tradeoff	87
The ROC Curve	91
Multiclass Classification	93
Error Analysis	96
Multilabel Classification	100
Multioutput Classification	101

Exercises	102
<b>4. Training Models</b>	<b>105</b>
Linear Regression	106
The Normal Equation	108
Computational Complexity	110
Gradient Descent	111
Batch Gradient Descent	114
Stochastic Gradient Descent	117
Mini-batch Gradient Descent	119
Polynomial Regression	121
Learning Curves	123
Regularized Linear Models	127
Ridge Regression	127
Lasso Regression	130
Elastic Net	132
Early Stopping	133
Logistic Regression	134
Estimating Probabilities	134
Training and Cost Function	135
Decision Boundaries	136
Softmax Regression	139
Exercises	142
<b>5. Support Vector Machines</b>	<b>145</b>
Linear SVM Classification	145
Soft Margin Classification	146
Nonlinear SVM Classification	149
Polynomial Kernel	150
Adding Similarity Features	151
Gaussian RBF Kernel	152
Computational Complexity	153
SVM Regression	154
Under the Hood	156
Decision Function and Predictions	156
Training Objective	157
Quadratic Programming	159
The Dual Problem	160
Kernelized SVM	161
Online SVMs	164
Exercises	165

<b>6. Decision Trees.....</b>	<b>167</b>
Training and Visualizing a Decision Tree	167
Making Predictions	169
Estimating Class Probabilities	171
The CART Training Algorithm	171
Computational Complexity	172
Gini Impurity or Entropy?	172
Regularization Hyperparameters	173
Regression	175
Instability	177
Exercises	178
<b>7. Ensemble Learning and Random Forests.....</b>	<b>181</b>
Voting Classifiers	181
Bagging and Pasting	185
Bagging and Pasting in Scikit-Learn	186
Out-of-Bag Evaluation	187
Random Patches and Random Subspaces	188
Random Forests	189
Extra-Trees	190
Feature Importance	190
Boosting	191
AdaBoost	192
Gradient Boosting	195
Stacking	200
Exercises	202
<b>8. Dimensionality Reduction.....</b>	<b>205</b>
The Curse of Dimensionality	206
Main Approaches for Dimensionality Reduction	207
Projection	207
Manifold Learning	210
PCA	211
Preserving the Variance	211
Principal Components	212
Projecting Down to d Dimensions	213
Using Scikit-Learn	214
Explained Variance Ratio	214
Choosing the Right Number of Dimensions	215
PCA for Compression	216
Incremental PCA	217
Randomized PCA	218

Kernel PCA	218
Selecting a Kernel and Tuning Hyperparameters	219
LLE	221
Other Dimensionality Reduction Techniques	223
Exercises	224

---

## Part II. Neural Networks and Deep Learning

<b>9. Up and Running with TensorFlow.....</b>	<b>229</b>
Installation	232
Creating Your First Graph and Running It in a Session	232
Managing Graphs	234
Lifecycle of a Node Value	235
Linear Regression with TensorFlow	235
Implementing Gradient Descent	237
Manually Computing the Gradients	237
Using autodiff	238
Using an Optimizer	239
Feeding Data to the Training Algorithm	239
Saving and Restoring Models	241
Visualizing the Graph and Training Curves Using TensorBoard	242
Name Scopes	245
Modularity	246
Sharing Variables	248
Exercises	251
<b>10. Introduction to Artificial Neural Networks.....</b>	<b>253</b>
From Biological to Artificial Neurons	254
Biological Neurons	255
Logical Computations with Neurons	256
The Perceptron	257
Multi-Layer Perceptron and Backpropagation	261
Training an MLP with TensorFlow's High-Level API	264
Training a DNN Using Plain TensorFlow	265
Construction Phase	265
Execution Phase	269
Using the Neural Network	269
Fine-Tuning Neural Network Hyperparameters	270
Number of Hidden Layers	270
Number of Neurons per Hidden Layer	271
Activation Functions	272

Exercises	272
<b>11. Training Deep Neural Nets</b>	<b>275</b>
Vanishing/Exploding Gradients Problems	275
Xavier and He Initialization	277
Nonsaturating Activation Functions	279
Batch Normalization	282
Gradient Clipping	286
Reusing Pretrained Layers	287
Reusing a TensorFlow Model	287
Reusing Models from Other Frameworks	289
Freezing the Lower Layers	290
Caching the Frozen Layers	291
Tweaking, Dropping, or Replacing the Upper Layers	292
Model Zoos	292
Unsupervised Pretraining	293
Pretraining on an Auxiliary Task	294
Faster Optimizers	295
Momentum Optimization	295
Nesterov Accelerated Gradient	297
AdaGrad	298
RMSProp	300
Adam Optimization	300
Learning Rate Scheduling	303
Avoiding Overfitting Through Regularization	305
Early Stopping	305
$\ell_1$ and $\ell_2$ Regularization	305
Dropout	307
Max-Norm Regularization	309
Data Augmentation	311
Practical Guidelines	312
Exercises	313
<b>12. Distributing TensorFlow Across Devices and Servers</b>	<b>315</b>
Multiple Devices on a Single Machine	316
Installation	316
Managing the GPU RAM	319
Placing Operations on Devices	320
Parallel Execution	323
Control Dependencies	325
Multiple Devices Across Multiple Servers	325
Opening a Session	327

The Master and Worker Services	327
Pinning Operations Across Tasks	328
Sharding Variables Across Multiple Parameter Servers	329
Sharing State Across Sessions Using Resource Containers	330
Asynchronous Communication Using TensorFlow Queues	331
Loading Data Directly from the Graph	337
Parallelizing Neural Networks on a TensorFlow Cluster	344
One Neural Network per Device	344
In-Graph Versus Between-Graph Replication	345
Model Parallelism	347
Data Parallelism	349
Exercises	354
<b>13. Convolutional Neural Networks</b>	<b>355</b>
The Architecture of the Visual Cortex	356
Convolutional Layer	357
Filters	359
Stacking Multiple Feature Maps	360
TensorFlow Implementation	362
Memory Requirements	364
Pooling Layer	365
CNN Architectures	367
LeNet-5	368
AlexNet	369
GoogLeNet	371
ResNet	374
Exercises	378
<b>14. Recurrent Neural Networks</b>	<b>381</b>
Recurrent Neurons	382
Memory Cells	384
Input and Output Sequences	384
Basic RNNs in TensorFlow	386
Static Unrolling Through Time	387
Dynamic Unrolling Through Time	389
Handling Variable Length Input Sequences	389
Handling Variable-Length Output Sequences	390
Training RNNs	391
Training a Sequence Classifier	391
Training to Predict Time Series	393
Creative RNN	398
Deep RNNs	398

Distributing a Deep RNN Across Multiple GPUs	399
Applying Dropout	400
The Difficulty of Training over Many Time Steps	402
LSTM Cell	402
Peephole Connections	405
GRU Cell	406
Natural Language Processing	407
Word Embeddings	407
An Encoder–Decoder Network for Machine Translation	409
Exercises	412
<b>15. Autoencoders</b>	<b>415</b>
Efficient Data Representations	416
Performing PCA with an Undercomplete Linear Autoencoder	417
Stacked Autoencoders	419
TensorFlow Implementation	420
Tying Weights	421
Training One Autoencoder at a Time	422
Visualizing the Reconstructions	425
Visualizing Features	425
Unsupervised Pretraining Using Stacked Autoencoders	426
Denoising Autoencoders	428
TensorFlow Implementation	429
Sparse Autoencoders	430
TensorFlow Implementation	432
Variational Autoencoders	433
Generating Digits	436
Other Autoencoders	437
Exercises	438
<b>16. Reinforcement Learning</b>	<b>441</b>
Learning to Optimize Rewards	442
Policy Search	444
Introduction to OpenAI Gym	445
Neural Network Policies	448
Evaluating Actions: The Credit Assignment Problem	451
Policy Gradients	452
Markov Decision Processes	457
Temporal Difference Learning and Q-Learning	461
Exploration Policies	463
Approximate Q-Learning	464
Learning to Play Ms. Pac-Man Using Deep Q-Learning	464



Exercises	473
Thank You!	473
<b>A. Exercise Solutions</b> .....	<b>475</b>
<b>B. Machine Learning Project Checklist</b> .....	<b>501</b>
<b>C. SVM Dual Problem</b> .....	<b>507</b>
<b>D. Autodiff</b> .....	<b>511</b>
<b>E. Other Popular ANN Architectures</b> .....	<b>519</b>
<b>Index</b> .....	<b>529</b>

In 2006, Geoffrey Hinton et al. published a paper<sup>1</sup> showing how to train a deep neural network capable of recognizing handwritten digits with state-of-the-art precision (99%). They branded this technique "Deep Learning." Training a deep neural net was widely considered impossible at the time,<sup>2</sup> and most researchers had abandoned the idea since the 1990s. This paper revived the interest of the scientific community and before long many new papers demonstrated that Deep Learning was not only possible, but capable of mind-blowing achievements that no other Machine Learning (ML) technique could hope to match (with the help of tremendous computing power and great amounts of data). This enthusiasm soon extended to many other areas of Machine Learning.

Fast forward 10 years and Machine Learning has conquered the industry: it is now at the heart of much of the magic in today's high-tech products, ranking your web search results, powering your smartphone's speech recognition, and recommending videos, beating the world champion at the game of Go. Before you know it, it will be driving your car.

## Machine Learning in Your Projects

So naturally you are excited about Machine Learning and you would love to join the party!

Perhaps you would like to give your homemade robot a brain of its own? Make it recognize faces! Or learn to walk around!

<sup>1</sup> Available on Hinton's home page at <http://www.cs.toronto.edu/~hinton/>

<sup>2</sup> Despite the fact that Yann Lecun's deep convolutional neural networks had worked well for image recognition since the 1990s, although they were not as general purpose.