

Contents

	<i>Preface</i>	<i>page xi</i>
1	Introduction	1
	1.1 Teaching a computer to distinguish cats from dogs	1
	1.1.1 The pipeline of a typical machine learning problem	5
	1.2 Predictive learning problems	6
	1.2.1 Regression	6
	1.2.2 Classification	9
	1.3 Feature design	12
	1.4 Numerical optimization	15
	1.5 Summary	16
	Part I Fundamental tools and concepts	19
2	Fundamentals of numerical optimization	21
	2.1 Calculus-defined optimality	21
	2.1.1 Taylor series approximations	21
	2.1.2 The first order condition for optimality	22
	2.1.3 The convenience of convexity	24
	2.2 Numerical methods for optimization	26
	2.2.1 The big picture	27
	2.2.2 Stopping condition	27
	2.2.3 Gradient descent	29
	2.2.4 Newton's method	33
	2.3 Summary	38
	2.4 Exercises	38
3	Regression	45
	3.1 The basics of linear regression	45
	3.1.1 Notation and modeling	45
	3.1.2 The Least Squares cost function for linear regression	47
	3.1.3 Minimization of the Least Squares cost function	48

3.1.4	The efficacy of a learned model	50
3.1.5	Predicting the value of new input data	50
3.2	Knowledge-driven feature design for regression	51
3.2.1	General conclusions	54
3.3	Nonlinear regression and ℓ_2 regularization	56
3.3.1	Logistic regression	56
3.3.2	Non-convex cost functions and ℓ_2 regularization	59
3.4	Summary	61
3.5	Exercises	62
4	Classification	73
4.1	The perceptron cost functions	73
4.1.1	The basic perceptron model	73
4.1.2	The softmax cost function	75
4.1.3	The margin perceptron	78
4.1.4	Differentiable approximations to the margin perceptron	80
4.1.5	The accuracy of a learned classifier	82
4.1.6	Predicting the value of new input data	83
4.1.7	Which cost function produces the best results?	84
4.1.8	The connection between the perceptron and counting costs	85
4.2	The logistic regression perspective on the softmax cost	86
4.2.1	Step functions and classification	87
4.2.2	Convex logistic regression	89
4.3	The support vector machine perspective on the margin perceptron	91
4.3.1	A quest for the hyperplane with maximum margin	91
4.3.2	The hard-margin SVM problem	93
4.3.3	The soft-margin SVM problem	93
4.3.4	Support vector machines and logistic regression	95
4.4	Multiclass classification	95
4.4.1	One-versus-all multiclass classification	96
4.4.2	Multiclass softmax classification	99
4.4.3	The accuracy of a learned multiclass classifier	103
4.4.4	Which multiclass classification scheme works best?	104
4.5	Knowledge-driven feature design for classification	104
4.5.1	General conclusions	106
4.6	Histogram features for real data types	107
4.6.1	Histogram features for text data	109
4.6.2	Histogram features for image data	112
4.6.3	Histogram features for audio data	115
4.7	Summary	117
4.8	Exercises	118

Part II	Tools for fully data-driven machine learning	129
5	Automatic feature design for regression	131
5.1	Automatic feature design for the ideal regression scenario	131
5.1.1	Vector approximation	132
5.1.2	From vectors to continuous functions	133
5.1.3	Continuous function approximation	134
5.1.4	Common bases for continuous function approximation	135
5.1.5	Recovering weights	140
5.1.6	Graphical representation of a neural network	140
5.2	Automatic feature design for the real regression scenario	141
5.2.1	Approximation of discretized continuous functions	142
5.2.2	The real regression scenario	142
5.3	Cross-validation for regression	146
5.3.1	Diagnosing the problem of overfitting/underfitting	149
5.3.2	Hold out cross-validation	149
5.3.3	Hold out calculations	151
5.3.4	k -fold cross-validation	152
5.4	Which basis works best?	155
5.4.1	Understanding of the phenomenon underlying the data	156
5.4.2	Practical considerations	156
5.4.3	When the choice of basis is arbitrary	156
5.5	Summary	158
5.6	Exercises	158
5.7	Notes on continuous function approximation	165
6	Automatic feature design for classification	166
6.1	Automatic feature design for the ideal classification scenario	166
6.1.1	Approximation of piecewise continuous functions	166
6.1.2	The formal definition of an indicator function	168
6.1.3	Indicator function approximation	170
6.1.4	Recovering weights	170
6.2	Automatic feature design for the real classification scenario	171
6.2.1	Approximation of discretized indicator functions	171
6.2.2	The real classification scenario	172
6.2.3	Classifier accuracy and boundary definition	178
6.3	Multiclass classification	179
6.3.1	One-versus-all multiclass classification	179
6.3.2	Multiclass softmax classification	180
6.4	Cross-validation for classification	180
6.4.1	Hold out cross-validation	182
6.4.2	Hold out calculations	182

6.4.3	<i>k</i> -fold cross-validation	184
6.4.4	<i>k</i> -fold cross-validation for one-versus-all multiclass classification	187
6.5	Which basis works best?	187
6.6	Summary	188
6.7	Exercises	189
7	Kernels, backpropagation, and regularized cross-validation	195
7.1	Fixed feature kernels	195
7.1.1	The fundamental theorem of linear algebra	196
7.1.2	Kernelizing cost functions	197
7.1.3	The value of kernelization	197
7.1.4	Examples of kernels	199
7.1.5	Kernels as similarity matrices	201
7.2	The backpropagation algorithm	202
7.2.1	Computing the gradient of a two layer network cost function	203
7.2.2	Three layer neural network gradient calculations	205
7.2.3	Gradient descent with momentum	206
7.3	Cross-validation via ℓ_2 regularization	208
7.3.1	ℓ_2 regularization and cross-validation	209
7.3.2	Regularized <i>k</i> -fold cross-validation for regression	210
7.3.3	Regularized cross-validation for classification	211
7.4	Summary	212
7.5	Further kernel calculations	212
7.5.1	Kernelizing various cost functions	212
7.5.2	Fourier kernel calculations – scalar input	214
7.5.3	Fourier kernel calculations – vector input	215
	Part III Methods for large scale machine learning	217
8	Advanced gradient schemes	219
8.1	Fixed step length rules for gradient descent	219
8.1.1	Gradient descent and simple quadratic surrogates	219
8.1.2	Functions with bounded curvature and optimally conservative step length rules	221
8.1.3	How to use the conservative fixed step length rule	224
8.2	Adaptive step length rules for gradient descent	225
8.2.1	Adaptive step length rule via backtracking line search	226
8.2.2	How to use the adaptive step length rule	227
8.3	Stochastic gradient descent	229
8.3.1	Decomposing the gradient	229
8.3.2	The stochastic gradient descent iteration	230
8.3.3	The value of stochastic gradient descent	232

		233
	8.3.4 Step length rules for stochastic gradient descent	233
	8.3.5 How to use the stochastic gradient method in practice	234
8.4	Convergence proofs for gradient descent schemes	235
	8.4.1 Convergence of gradient descent with Lipschitz constant fixed step length	236
	8.4.2 Convergence of gradient descent with backtracking line search	236
	8.4.3 Convergence of the stochastic gradient method	238
	8.4.4 Convergence rate of gradient descent for convex functions with fixed step length	239
8.5	Calculation of computable Lipschitz constants	241
8.6	Summary	243
8.7	Exercises	243
9	Dimension reduction techniques	245
	9.1 Techniques for data dimension reduction	245
	9.1.1 Random subsampling	245
	9.1.2 K -means clustering	246
	9.1.3 Optimization of the K -means problem	249
	9.2 Principal component analysis	250
	9.2.1 Optimization of the PCA problem	256
	9.3 Recommender systems	256
	9.3.1 Matrix completion setup	257
	9.3.2 Optimization of the matrix completion model	258
	9.4 Summary	259
	9.5 Exercises	260
Part IV Appendices		263
A	Basic vector and matrix operations	265
	A.1 Vector operations	265
	A.2 Matrix operations	266
B	Basics of vector calculus	268
	B.1 Basic definitions	268
	B.2 Commonly used rules for computing derivatives	269
	B.3 Examples of gradient and Hessian calculations	269
C	Fundamental matrix factorizations and the pseudo-inverse	274
	C.1 Fundamental matrix factorizations	274
	C.1.1 The singular value decomposition	274
	C.1.2 Eigenvalue decomposition	276
	C.1.3 The pseudo-inverse	277

D Convex geometry	278
D.1 Definitions of convexity	278
D.1.1 Zeroth order definition of a convex function	278
D.1.2 First order definition of a convex function	279
<i>References</i>	280
<i>Index</i>	285