

Darrel Hankerson
Greg A. Harris
Peter D. Johnson, Jr.
Auburn University

Introduction to
Information
Theory
and
Data
Compression



CRC Press

Boca Raton Boston New York Washington, D.C. London

Contents

Preface	v
Part I: Information Theory	1
1 Elementary Probability	1
1.1 Introduction	1
1.2 Events	3
1.3 Conditional probability	7
1.4 Independence	11
1.5 Bernoulli trials	13
1.6 An elementary counting principle	15
1.7* On drawing without replacement	17
1.8 Random variables and expected, or average, value	18
1.9 The law of large numbers	22
2 Information and Entropy	25
2.1 Systems of events	25
2.2 Information	28
2.3 Entropy	32
2.4 Information and entropy	35
3 Channels and Channel Capacity	37
3.1 Discrete memoryless channels	37
3.2 Transition probabilities and binary symmetric channels	40
3.3 Input frequencies	42
3.4 Channel capacity	46
3.5* Proof of Theorem 3.4.1	54
4 Coding Theory	57
4.1 Encoding and decoding	57
4.2 Prefix-condition codes and the Kraft-McMillan inequality	61
4.3 Average code word length and Huffman's algorithm	65
4.3.1 The validity of Huffman's algorithm	71
4.4 Optimizing the input frequencies	75
4.5 Error correction, maximum likelihood decoding, nearest code word decoding, and reliability	79
4.6 Shannon's Noisy Channel Theorem	89
4.7 Error correction with binary symmetric channels and equal source frequencies	94

Part II: Data Compression	99
5 Lossless Data Compression by Replacement Schemes	99
5.1 Replacement via encoding scheme	100
5.2 Review of the prefix condition	103
5.3 How to choose an encoding scheme	106
5.3.1 Shannon's method	107
5.3.2 Fano's method	110
5.3.3 Huffman's algorithm	111
5.4 The Noiseless Coding Theorem and Shannon's bound	114
6 Arithmetic Coding	119
6.1 Pure zeroth-order arithmetic coding: dfwld	120
6.1.1 Rescaling while encoding	124
6.1.2 Decoding	126
6.2 What's good about dfwld coding: the compression ratio	131
6.3 What's bad about dfwld coding and some ways to fix it	136
6.3.1 Supplying the source word length	136
6.3.2 Computation	137
6.3.3 Must decoding wait until encoding is completed?	139
6.4 Implementing arithmetic coding	142
6.4.1 Use of integer arithmetic	145
6.4.2 Implementation and performance issues	147
6.5 Notes	153
7 Higher-order Modeling	155
7.1 Higher-order Huffman encoding	156
7.2 The Shannon bound for higher-order encoding	160
7.3 Higher-order arithmetic coding	165
7.4 Statistical models, statistics, and the possibly unknowable truth	167
8 Adaptive Methods	171
8.1 Adaptive Huffman encoding	172
8.1.1 Compression and readjustment	175
8.1.2 Higher-order adaptive Huffman encoding	176
8.2 Maintaining the tree in adaptive Huffman encoding: the method of Knuth and Gallager	178
8.2.1 Gallager's method	181
8.2.2 Knuth's algorithm	182
8.3 Adaptive arithmetic coding	185
8.4 Interval and recency rank encoding	187
8.4.1 Interval encoding	187
8.4.2 Recency rank encoding	190

9	Dictionary Methods	195
9.1	LZ77 (sliding window) schemes	196
9.1.1	An LZ77 implementation	198
9.1.2	Case study: GNU zip	201
9.2	The LZ78 approach	203
9.2.1	The LZW variant	206
9.2.2	Case study: Unix <i>compress</i>	208
9.3	Notes	210
10	Transform Methods and Image Compression	211
10.1	Transforms	212
10.2	Periodic signals and the Fourier transform	215
10.2.1	The Fourier transform and compression: an example	221
10.3	The cosine and sine transforms	232
10.3.1	A general orthogonal transform	236
10.3.2	Summary	236
10.4	Two-dimensional transforms	239
10.4.1	The 2D Fourier, cosine, and sine transforms	241
10.4.2	Matrix expressions for 2D transforms	244
10.5	An application: JPEG image compression	246
10.6	A brief introduction to wavelets	257
10.6.1	2D Haar wavelets	261
10.7	Notes	266
	Appendices	269
A	JPEGtool User's Guide	269
A.1	Using the tools	270
A.2	Reference	280
A.3	Obtaining Octave	285
B	Source Listing for LZRW1-A	287
C	Resources, Patents, and Illusions	299
C.1	Resources	299
C.2	Data compression and patents	301
C.3	Illusions	304
D	Notes on the Exercises	309
	Bibliography	323
	Index	327