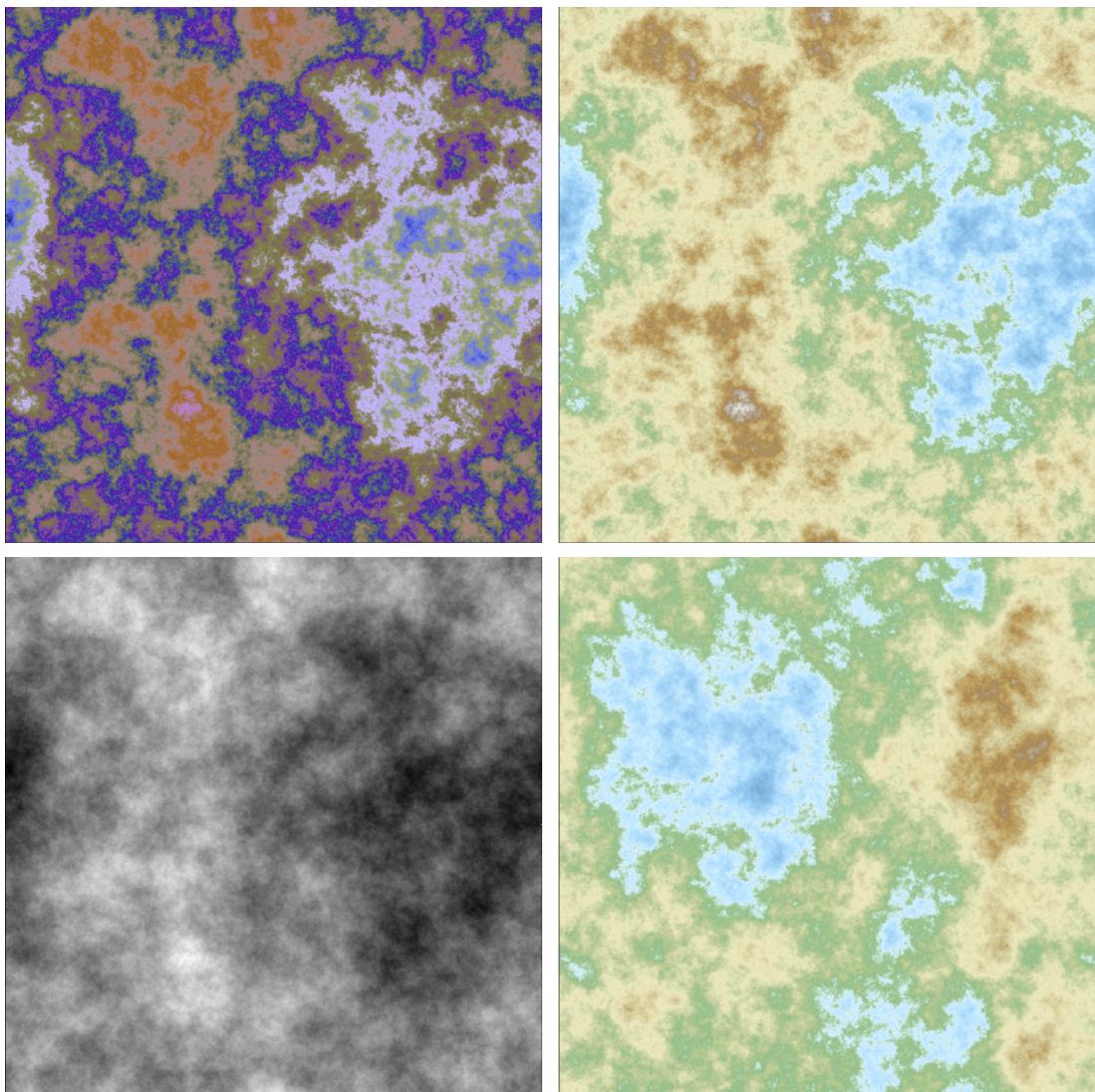


---

# Synthetic Data and Generative AI



# Preface

This book covers the foundations of machine learning, with modern approaches to solving complex problems and the systematic generation and use of synthetic data. Emphasis is on scalability, automation, testing, optimizing, and interpretability (explainable AI). For instance, regression techniques – including logistic and Lasso – are presented as a single method, without using advanced linear algebra. There is no need to learn 50 versions when one does it all and more. Confidence regions and prediction intervals are built using parametric bootstrap, without statistical models or probability distributions. Models (including generative models and mixtures) are mostly used to create rich synthetic data to test and benchmark various methods.

Topics covered include clustering and classification, GPU machine learning, ensemble methods including an original boosting technique, elements of graph modeling, deep neural networks, auto-regressive and non-periodic time series, Brownian motions and related processes, simulations, interpolation, random numbers, natural language processing (smart crawling, taxonomy creation and structuring unstructured data), computer vision (shapes generation and recognition), curve fitting, cross-validation, goodness-of-fit metrics, feature selection, curve fitting, gradient methods, optimization techniques and numerical stability.

Chapters 12 and 13 focus on synthetic data applications: fractal-like terrain generation with the diamond-square algorithm, and synthetic star clusters evolving over time and bound by gravity. The latter provides great insights to explore the past and future of our universe or studying collision graphs. It also allows you to explore alternative universes, for instance with negative masses. Chapters 14 and 16 are more advanced and may be skipped in introductory classes. The former focuses on point process applications, while the later focuses on applications a machine learning methods to discover new insights in a famous mathematical conjecture: the Riemann Hypothesis. Section 16.7.2 illustrates the use of copulas to produce synthetic data, applied to a well-known insurance dataset.

Methods are accompanied by enterprise-grade Python code, replicable datasets and visualizations, including data animations (gifs, videos, even sound done in Python). The code uses various data structures and library functions sometimes with advanced options. It constitutes a Python tutorial in itself, and an introduction to scientific computing. Some data animations and chart enhancements are done in R. The code, datasets, spreadsheets and data visualizations are also on GitHub, spread across the following repositories: [Machine Learning](#), [Point Processes](#), [Visualizations](#), and [Experimental Math](#). Chapters are mostly independent from each other, allowing you to read in random order. A glossary, index and numerous cross-references make the navigation easy and unify all the chapters.

The style is very compact, getting down to the point quickly, and suitable to business professionals eager to learn a lot of useful material in a limited amount of time. Jargon and arcane theories are absent, replaced by simple English to facilitate the reading by non-experts, and to help you discover topics usually made inaccessible to beginners. While state-of-the-art research is presented in all chapters, the prerequisites to read this book are minimal: an analytic professional background, or a first course in calculus and linear algebra. The original presentation avoids all unnecessary math and statistics, yet without eliminating advanced topics. Finally, this book is the main reference for my course on intuitive machine learning. For details about the classes, see [here](#).

## About the Author

Vincent Granville is a pioneering data scientist and machine learning expert, co-founder of Data Science Central (acquired by a publicly traded company in 2020), founder of [MLTechniques.com](#), former VC-funded executive, author and patent owner. Vincent's past corporate experience includes Visa, Wells Fargo, eBay, NBC, Microsoft, and CNET.



Vincent is also a former post-doc at Cambridge University, and the National Institute of Statistical Sciences (NISS). He published in *Journal of Number Theory*, *Journal of the Royal Statistical Society* (Series B), and *IEEE Transactions on Pattern Analysis and Machine Intelligence*. He is also the author of multiple books, available [here](#). He lives in Washington state, and enjoys doing research on stochastic processes, dynamical systems, experimental math and probabilistic number theory.

# Contents

<b>List of Figures</b>	<b>9</b>
<b>List of Tables</b>	<b>11</b>
<b>1 Machine Learning Cloud Regression and Optimization</b>	<b>12</b>
1.1 Introduction: circle fitting . . . . .	12
1.1.1 Previous versions of my method . . . . .	13
1.2 Methodology, implementation details and caveats . . . . .	14
1.2.1 Solution, R-squared and backward compatibility . . . . .	14
1.2.2 Upgrades to the model . . . . .	15
1.3 Case studies . . . . .	16
1.3.1 Logistic regression, two ways . . . . .	16
1.3.2 Ellipsoid and hyperplane fitting . . . . .	17
1.3.2.1 Curve fitting: 250 examples in one video . . . . .	17
1.3.2.2 Confidence region for the fitted ellipse: application to meteorite shapes . . . . .	18
1.3.2.3 Python code . . . . .	19
1.3.3 Non-periodic sum of periodic time series: ocean tides . . . . .	25
1.3.3.1 Numerical instability and how to fix it . . . . .	26
1.3.3.2 Python code . . . . .	27
1.3.4 Fitting a line in 3D, unsupervised clustering, and other generalizations . . . . .	28
1.3.4.1 Example: confidence region for the cluster centers . . . . .	29
1.3.4.2 Exact solution and caveats . . . . .	30
1.3.4.3 Comparison with K-means clustering . . . . .	31
1.3.4.4 Python code . . . . .	33
1.4 Connection to synthetic data: meteorites, ocean tides . . . . .	35
<b>2 A Simple, Robust and Efficient Ensemble Method</b>	<b>36</b>
2.1 Introduction . . . . .	36
2.2 Methodology . . . . .	37
2.2.1 How hidden decision trees (HDT) work . . . . .	37
2.2.2 NLP case study: summary and findings . . . . .	38
2.2.3 Parameters . . . . .	39
2.2.4 Improving the methodology . . . . .	39
2.3 Implementation details . . . . .	39
2.3.1 Correcting for bias . . . . .	39
2.3.1.1 Time-adjusted scores . . . . .	40
2.3.2 Excel spreadsheet . . . . .	40
2.3.3 Python code and dataset . . . . .	40
2.4 Model-free confidence intervals and perfect nodes . . . . .	44
2.4.1 Interesting asymptotic properties of confidence intervals . . . . .	44
<b>3 Gentle Introduction to Linear Algebra</b>	<b>46</b>
3.1 Power of a matrix . . . . .	46
3.2 Examples, generalization, and matrix inversion . . . . .	47
3.2.1 Example with a non-invertible matrix . . . . .	48
3.2.2 Fast computations . . . . .	48
3.2.3 Square root of a matrix . . . . .	48
3.3 Application to machine learning problems . . . . .	49
3.3.1 Markov chains . . . . .	49
3.3.2 Time series: auto-regressive processes . . . . .	49

3.3.3	Linear regression . . . . .	49
3.4	Mathematics of auto-regressive time series . . . . .	50
3.4.1	Simulations: curious fractal time series . . . . .	50
3.4.1.1	White noise: Fréchet, Weibull and exponential cases . . . . .	51
3.4.1.2	Illustration . . . . .	51
3.4.2	Solving Vandermonde systems: a numerically stable method . . . . .	52
3.5	Math for Machine Learning: Must-Read Books . . . . .	53
<b>4</b>	<b>The Art of Visualizing High Dimensional Data</b>	<b>54</b>
4.1	Introduction . . . . .	54
4.2	Applications . . . . .	55
4.2.1	Spatial time series . . . . .	55
4.2.2	Prediction intervals in any dimensions . . . . .	55
4.2.3	Supervised classification of an infinite dataset . . . . .	56
4.2.3.1	Machine learning perspective . . . . .	56
4.2.3.2	Six challenging problems . . . . .	57
4.2.3.3	Mathematical background: the Riemann Hypothesis . . . . .	57
4.2.3.4	Partial solutions to the six challenging problems . . . . .	58
4.2.4	Algorithms with chaotic convergence . . . . .	59
4.3	Python code . . . . .	59
4.3.1	Path simulation . . . . .	59
4.3.2	Visual convergence analysis in 2D . . . . .	62
4.3.3	Supervised classification . . . . .	63
4.4	Visualizations . . . . .	66
<b>5</b>	<b>Fast Classification and Clustering via Image Convolution Filters</b>	<b>69</b>
5.1	Introduction . . . . .	69
5.2	Generating the synthetic data . . . . .	70
5.2.1	Simulations with logistic distribution . . . . .	70
5.2.2	Mapping the raw observations onto an image bitmap . . . . .	71
5.3	Classification and unsupervised clustering . . . . .	71
5.3.1	Supervised classification based on convolution filters . . . . .	72
5.3.2	Clustering based on histogram equalization . . . . .	72
5.3.3	Fractal classification: deep neural network analogy . . . . .	73
5.3.4	Generalization to higher dimensions . . . . .	74
5.3.5	Towards a very fast implementation . . . . .	74
5.4	Python code . . . . .	75
5.4.1	Fractal classification . . . . .	76
5.4.2	GPU classification and clustering . . . . .	78
5.4.3	Home-made graphic library . . . . .	80
<b>6</b>	<b>Shape Classification via Explainable AI</b>	<b>83</b>
6.1	Introduction . . . . .	83
6.2	Mathematical foundations . . . . .	83
6.3	Shape signature . . . . .	84
6.3.1	Weighted centroid . . . . .	84
6.3.2	Computing the signature . . . . .	85
6.3.3	Example . . . . .	86
6.4	Shape comparison . . . . .	86
6.4.1	Shape classification . . . . .	87
6.5	Application . . . . .	87
6.6	Exercises . . . . .	88
<b>7</b>	<b>Synthetic Data, Interpretable Regression, and Submodels</b>	<b>89</b>
7.1	Introduction . . . . .	89
7.2	Synthetic data sets and the spreadsheet . . . . .	90
7.2.1	Correlation structure . . . . .	90
7.2.2	Standardized regression . . . . .	91
7.2.3	Initial conditions . . . . .	91
7.2.4	Simulations and Excel spreadsheet . . . . .	91
7.3	Damping schedule and convergence acceleration . . . . .	92
7.3.1	Spreadsheet implementation . . . . .	92

7.3.2	Interpretable regression with no overfitting . . . . .	93
7.3.3	Adaptive damping . . . . .	93
7.4	Performance assessment on synthetic data . . . . .	93
7.4.1	Results . . . . .	94
7.4.2	Distribution-free confidence intervals . . . . .	96
7.4.2.1	Parametric bootstrap . . . . .	97
7.5	Feature selection . . . . .	97
7.5.1	Combinatorial approach . . . . .	97
7.5.2	Stepwise approach . . . . .	98
7.6	Conclusion . . . . .	99
<b>8</b>	<b>From Interpolation to Fuzzy Regression</b>	<b>101</b>
8.1	Introduction . . . . .	101
8.2	Original version . . . . .	102
8.3	Full, non-linear model in higher dimensions . . . . .	102
8.3.1	Geometric proximity, weights, and numerical stability . . . . .	103
8.3.2	Predicted values and prediction intervals . . . . .	103
8.3.3	Illustration, with spreadsheet . . . . .	104
8.3.3.1	Output fields . . . . .	105
8.4	Results . . . . .	105
8.4.1	Performance assessment . . . . .	105
8.4.2	Visualization . . . . .	106
8.4.3	Amplitude restoration . . . . .	106
8.5	Exercises . . . . .	107
8.6	Python source code and datasets . . . . .	108
<b>9</b>	<b>Detecting Subtle Departures from Randomness</b>	<b>112</b>
9.1	Introduction . . . . .	112
9.2	Pseudo-random numbers . . . . .	113
9.2.1	Strong pseudo-random numbers . . . . .	113
9.2.1.1	New test of randomness for PRNGs . . . . .	114
9.2.1.2	Theoretical background: the law of the iterated logarithm . . . . .	114
9.2.1.3	Connection to the Generalized Riemann Hypothesis . . . . .	114
9.2.2	Testing well-known sequences . . . . .	115
9.2.2.1	Reverse-engineering a pseudo-random sequence . . . . .	116
9.2.2.2	Illustrations . . . . .	117
9.3	Python code . . . . .	119
9.3.1	Fixes to the faulty random function in Python . . . . .	119
9.3.2	Prime test implementation to detect subtle flaws in PRNG's . . . . .	119
9.3.3	Special formula to compute 10 million digits of $\sqrt{2}$ . . . . .	122
9.4	Military-grade PRNG Based on Quadratic Irrationals . . . . .	125
9.4.1	Fast algorithm rooted in advanced analytic number theory . . . . .	125
9.4.2	Fast PRNG: explanations . . . . .	126
9.4.3	Python code . . . . .	126
9.4.4	Computing a digit without generating the previous ones . . . . .	128
9.4.5	Security and comparison with other PRNGs . . . . .	128
9.4.5.1	Important comments . . . . .	128
9.4.6	Curious application: a new type of lottery . . . . .	129
<b>10</b>	<b>Some Unusual Random Walks</b>	<b>130</b>
10.1	Symmetric unbiased constrained random walks . . . . .	130
10.1.1	Three fundamental properties of pure random walks . . . . .	130
10.1.2	Random walks with more entropy than pure random signal . . . . .	131
10.1.2.1	Applications . . . . .	131
10.1.2.2	Algorithm to generate quasi-random sequences . . . . .	132
10.1.2.3	Variance of the modified random walk . . . . .	132
10.1.3	Random walks with less entropy than pure random signal . . . . .	133
10.2	Related stochastic processes . . . . .	134
10.2.1	From Brownian motions to clustered Lévy flights . . . . .	134
10.2.2	Integrated Brownian motions and special auto-regressive processes . . . . .	135
10.3	Python code . . . . .	136
10.3.1	Computing probabilities and variances attached to $S_n$ . . . . .	136

10.3.2 Path simulations . . . . .	137
<b>11 Empirical Optimization with Divergent Fixed Point Algorithm</b>	<b>139</b>
11.1 Introduction . . . . .	139
11.1.1 The problem, with illustration . . . . .	140
11.2 Non-converging fixed-point algorithm . . . . .	141
11.2.1 Trick leading to intuitive solution . . . . .	141
11.2.2 Root detection: method and parameters . . . . .	141
11.2.3 Case study: factoring a product of two large primes . . . . .	142
11.3 Generalization with synthetic random functions . . . . .	142
11.3.1 Example . . . . .	144
11.3.2 Connection to the Poisson-binomial distribution . . . . .	145
11.3.2.1 Location of next root: guesstimate . . . . .	145
11.3.2.2 Integer sequences with high density of primes . . . . .	145
11.3.3 Python code: finding the optimum . . . . .	146
11.4 Smoothing highly chaotic curves . . . . .	147
11.4.1 Python code: smoothing . . . . .	147
11.5 Connection to synthetic data: random functions . . . . .	150
<b>12 Synthetic Terrain Generation and AI-generated Art</b>	<b>151</b>
12.1 Introduction . . . . .	151
12.2 Terrain generation and the evolutionary process . . . . .	153
12.2.1 Morphing and non-linear palette operations . . . . .	153
12.2.2 The diamond-square algorithm . . . . .	153
12.2.3 The evolutionary process . . . . .	154
12.2.4 Finding optimum parameters . . . . .	154
12.2.5 Mimicking real terrain: the synthesis step . . . . .	154
12.3 Python code . . . . .	155
12.3.1 Producing data videos with four sub-videos in parallel . . . . .	155
12.3.2 Main program . . . . .	156
12.4 AI-generated art with 3D contours . . . . .	160
12.4.1 Python code using Matplotlib . . . . .	161
12.4.2 Python code using Plotly . . . . .	162
12.4.3 Tips to quickly solve new problems . . . . .	163
<b>13 Synthetic Star Cluster Generation with Collision Graphs</b>	<b>164</b>
13.1 Introduction . . . . .	164
13.2 Model parameters and simulation results . . . . .	165
13.2.1 Explanation of color codes . . . . .	165
13.2.2 Detailed description of top parameters . . . . .	165
13.2.3 Interesting parameter sets . . . . .	166
13.3 Analysis of star collisions and collision graph . . . . .	167
13.3.1 Weighted directed graphs: visualization with NetworkX . . . . .	168
13.3.2 Interesting findings: how the universe got started . . . . .	168
13.4 Animated data visualizations . . . . .	169
13.5 Python code and computational issues . . . . .	170
13.5.1 Simulating the real and synthetic universes . . . . .	170
13.5.2 Visualizing collision graphs . . . . .	174
<b>14 Perturbed-Lattice Point Process: Inference, Nearest Neighbor Graph</b>	<b>176</b>
14.1 Perturbed lattices: definition and properties . . . . .	176
14.1.1 Point counts distribution . . . . .	177
14.1.2 Periodicity and amplitude of point count expectations . . . . .	177
14.1.3 Testing the independence of point counts . . . . .	178
14.1.3.1 Results and Interpretation . . . . .	179
14.1.3.2 About the Spreadsheet . . . . .	180
14.2 Cluster processes and nearest neighbor graphs . . . . .	180
14.2.1 Synthetic, semi-rigid cluster structures . . . . .	180
14.2.2 Python code to generate cluster processes . . . . .	182
14.2.3 References on cluster processes . . . . .	182
14.2.4 Superimposed perturbed lattices: an alternative to mixture models . . . . .	183
14.2.4.1 Hexagonal lattice, nearest neighbors . . . . .	184

14.2.4.2 Exercises: nearest neighbor graphs, size of connected components . . . . .	185
14.2.4.3 Python code to compute connected components . . . . .	186
14.3 Statistical inference for point processes . . . . .	188
14.3.1 Estimation of Core Parameters . . . . .	188
14.3.1.1 Intensity . . . . .	189
14.3.1.2 Scaling factor . . . . .	189
14.3.1.3 Alternative estimation method . . . . .	189
14.3.2 Spatial statistics, nearest neighbors, clustering . . . . .	190
14.3.2.1 Inference for two-dimensional processes . . . . .	190
14.3.2.2 Other possible tests . . . . .	190
14.3.2.3 Rayleigh test . . . . .	191
14.3.2.4 Exercises . . . . .	192
14.4 Special topics . . . . .	193
14.4.1 Minimum contrast estimation and explainable AI . . . . .	193
14.4.2 Model identifiability, hard-to-detect patterns . . . . .	194
14.4.2.1 Stochastic residues . . . . .	194
14.4.3 Hidden model and random permutations . . . . .	194
14.4.4 Retrieving the $F$ distribution . . . . .	196
14.4.4.1 Theoretical values obtained by simulations . . . . .	196
14.4.4.2 Retrieving $F$ from the interarrival times distribution . . . . .	197
14.4.5 Record distances between an observed point and its vertex . . . . .	197
14.4.5.1 Distribution of records . . . . .	198
14.4.5.2 Distribution of arrival times for records . . . . .	199
<b>15 New Perspective on the Riemann Hypothesis</b>	<b>200</b>
15.1 Introduction . . . . .	200
15.1.1 Key concepts and terminology . . . . .	201
15.1.2 Orbits and holes . . . . .	201
15.1.3 Industrial Applications . . . . .	201
15.2 Euler products . . . . .	202
15.2.1 Finite Euler Products . . . . .	202
15.2.1.1 Generalization using Dirichlet characters . . . . .	203
15.2.2 Infinite Euler products . . . . .	204
15.2.2.1 Special products . . . . .	204
15.2.2.2 Probabilistic properties and conjectures . . . . .	205
15.3 Finite Dirichlet series and generalizations . . . . .	206
15.3.1 Finite Dirichlet series . . . . .	206
15.3.2 Non-trivial cases with infinitely many primes and a hole . . . . .	208
15.3.2.1 Sums of two cubes, or cuban primes . . . . .	208
15.3.2.2 Primes associated to elliptic curves . . . . .	208
15.3.2.3 Analytic continuation, convergence, and functional equation . . . . .	209
15.3.2.4 Hybrid Dirichlet-Taylor series . . . . .	209
15.3.3 Riemann Hypothesis with cosines replaced by wavelets . . . . .	210
15.3.4 Riemann Hypothesis for Beurling primes . . . . .	211
15.3.5 Stochastic Euler products . . . . .	212
15.4 Exercises . . . . .	213
15.5 Python code . . . . .	216
15.5.1 Computing the orbit of various Dirichlet series . . . . .	216
15.5.2 Creating videos of the orbit . . . . .	219
<b>16 Miscellaneous Topics</b>	<b>222</b>
16.1 The sound that data makes . . . . .	222
16.1.1 From data visualizations to videos to data music . . . . .	222
16.1.2 References . . . . .	223
16.1.3 Python code . . . . .	223
16.2 Data videos and enhanced visualizations in R . . . . .	224
16.2.1 Cairo library to produce better charts . . . . .	224
16.2.2 AV library to produce videos . . . . .	225
16.3 Dual confidence regions . . . . .	226
16.3.1 Case study . . . . .	226
16.3.2 Standard confidence region . . . . .	226

16.3.3	Dual confidence region . . . . .	227
16.3.4	Simulations . . . . .	227
16.3.5	Original problem with minimum contrast estimators . . . . .	228
16.3.6	General shape of confidence regions . . . . .	229
16.4	Fast feature selection based on predictive power . . . . .	230
16.4.1	How cross-validation works . . . . .	231
16.4.2	Measuring the predictive power of a feature . . . . .	231
16.4.3	Efficient implementation . . . . .	232
16.5	Natural language processing: taxonomy creation . . . . .	233
16.5.1	Designing a keyword taxonomy . . . . .	233
16.5.2	Fast clustering algorithm for keyword data . . . . .	234
16.5.2.1	Computational complexity . . . . .	234
16.5.2.2	Smart crawling of the whole Internet and a bit of graph theory . . . . .	234
16.6	Automated detection of outliers and number of clusters . . . . .	236
16.6.1	Black-box elbow rule to detect outliers . . . . .	236
16.7	New developments in explainable AI and synthetic data . . . . .	237
16.7.1	Sensitivity analysis, bias reduction and other uses of synthetic data . . . . .	238
16.7.2	Using copulas to generate synthetic data . . . . .	238
16.7.2.1	The insurance dataset: Python code and results . . . . .	239
16.8	Advice to beginners . . . . .	241
16.8.1	Getting started and learning how to learn . . . . .	241
16.8.1.1	Getting help . . . . .	241
16.8.1.2	Beyond Python . . . . .	242
16.8.2	Automated data cleaning and exploratory analysis . . . . .	242
16.8.3	Example of simple analysis: marketing attribution . . . . .	243
16.8.4	Upcoming books and courses . . . . .	243
<b>Glossary</b>		<b>244</b>
<b>Bibliography</b>		<b>247</b>
<b>Index</b>		<b>252</b>

# List of Figures

1.1	Fitted ellipse (blue), given the training set (red) distributed around a partial arc . . . . .	18
1.2	Confidence region in blue, $n = 30$ training set points; 50 training sets (left) vs 150 (right) . . . . .	19
1.3	Three non-periodic time series made of periodic terms (see section 15.2.2.1) . . . . .	25
1.4	Training set (red), validation set (orange), fitted curve (blue) and model (gray) . . . . .	26
1.6	Biased confidence region for $(\theta_A^*, \theta_B^*)$ ; same example as in Figure 1.5; true value is $(0.5, 1.0)$ . . . . .	29
1.5	Finding the two centers $\theta_A^*, \theta_B^*$ in sample 39; $n = 1000$ . . . . .	30
1.7	Challenging mixture, requiring $p_A = 3, p_B = 1$ to identify the two cluster centers . . . . .	31
2.1	Output from the Excel version of HDT . . . . .	41
3.1	AR models, classified based on the types of roots of the characteristic polynomial . . . . .	52
4.1	Scatterplot observations vs. predicted values, with prediction intervals (in any dimension) . . . . .	66
4.2	Comets orbiting the sun: simulation . . . . .	66
4.3	Comets orbiting the sun: snapshot in time . . . . .	67
4.4	Three orbits of $\eta(\sigma + it)$ : $\sigma = 0.5$ (red), 0.75 (blue) and 1.25 (yellow) . . . . .	67
4.5	Sample orbit points of $\eta(\sigma + it)$ : $\sigma = 0.5$ (red), 0.75 (blue) and 1.25 (yellow) . . . . .	67
4.6	Sample orbit points of $\eta(\sigma + it)$ : $\sigma = 0.5$ (red), 0.75 (blue) and 1.25 (yellow) . . . . .	68
4.7	Raw orbit points of $\eta(\sigma + it)$ : $\sigma = 0.5$ (red), 0.75 (blue) and 1.25 (yellow) . . . . .	68
4.8	Convergence of partial sums of $\eta(z)$ , for six $z = \sigma + it$ in the complex plane . . . . .	68
5.1	Special interlacing of 4 lattice processes with $s = 0$ . . . . .	71
5.2	Classification of left dataset; $s = 0.15, w = 10$ . One loop (middle) vs 3 (right). . . . .	72
5.3	Clustering of left dataset; $s = 0.15$ , 3 loops, $w = 10$ (middle) vs 20 (right). . . . .	73
5.4	Classification ( $w = 10$ ) and clustering ( $w = 20$ ); $s = 0.05$ , three loops. . . . .	73
5.5	Fractal classification, $s = 0.15$ . Loop 6, 250 and 400. . . . .	74
5.6	Fractal classification, $s = 0.05$ .Loop: 6 and 60. . . . .	74
5.7	Fast (left) vs standard method (right), 3 loops, $s = 0.15, w = 10$ . . . . .	75
5.8	Fast method, $s = 0.05, w = 20$ . Three loops (middle), one loop (right). . . . .	75
6.1	Comparing two shapes . . . . .	84
6.2	Weighted centroid, shape signature . . . . .	85
6.3	Weight function used in Figure 6.2 . . . . .	86
6.4	Another interesting shape . . . . .	87
7.1	Regression coefficients oscillating when using adaptive damping . . . . .	94
7.2	Convergence of regression coefficients (left) and distribution of residual error (right) . . . . .	95
7.3	Goodness-of-fit: training set (right) versus validation set (left) . . . . .	95
8.1	Fuzzy regression with prediction intervals, original version, 1D . . . . .	102
8.2	Fuzzy regression with prediction intervals, full model, 2D . . . . .	104
8.3	Scatterplots: median vs weighted method, on validation (left) vs training set (right) . . . . .	106
8.4	Dirichlet eta function (real part, bottom) and interpolation error (top) . . . . .	108
9.1	Orbit of $L(z, \chi)$ at $\sigma = \frac{1}{2}$ , with $0 < t < 200$ and $\chi = \chi_4$ (left) versus pseudo-random $\chi$ (right) . .	115
9.2	$L_3^*(n)$ test statistic for four sequences: Python[200] and SQRT[90,91] fail . . . . .	117
9.3	$ L_3(n) $ test statistic for four sequences: Python[200] and SQRT[90,91] fail . . . . .	117
9.4	Correlations are computed on sequences consisting of 300 binary digits . . . . .	129
10.1	Typical path $S_n$ with $0 \leq n \leq 50,000$ for four types of random walks . . . . .	131

10.2	$\delta_n = 1 - \text{Var}[S_{n+1}] + \text{Var}[S_n]$ for four types of random walks, with $0 \leq n \leq 5000$	132
10.3	Same as Figure 10.2, using a more aesthetic but less meaningful chart type	133
10.4	Clustered Brownian process	135
10.5	AR models, classified based on the types of roots of the characteristic polynomial	136
11.1	Function $f(b)$ as a better alternative to $g(b)$ in Figure 11.2. Root at $b = 3083$ .	140
11.2	Function $g(b) = 2 - \cos(2\pi b) - \cos(2\pi a/b)$ , with $a = 3083 \times 7919$ .	140
11.3	Transformed function $f_3$ , amplifying the root at $b = 3083$ .	141
11.4	Signal strength $\rho_n$ , first 130 fixed-point iterations; $n = 31$ leads to a root.	144
11.5	$(b_n, \rho_n)$ plot. Yellow and orange dots linked to roots.	144
11.6	Signal strength $\rho_n$ , first 130 fixed-point iterations; $n = 87$ leads to a root.	144
11.7	Random function from section 11.3.1, with root at $b = 5646$ .	147
12.1	Six frames from the terrain video, each one containing four images	152
12.2	Contour plot, 3D mixture model, produced with Plotly	160
12.3	Same as Figure 12.2, produced with Matplotlib	161
13.1	Collisions graph for the biggest star eater (star 47) in video 7	168
13.2	Summary statistics for the whole collision structure: the X axis represents the time	169
13.3	Snapshots of universe 4 (left) and universe 7 (right)	170
14.1	Period and amplitude of $\phi_\tau(t)$ ; here $\tau = 1, \lambda = 1.4, s = 0.3$	178
14.2	A new test of independence (R-squared version)	178
14.3	Radial cluster process ( $s = 0.2, \lambda = 1$ ) with centers in blue; zoom in on the left	181
14.4	Radial cluster process ( $s = 2, \lambda = 1$ ) with centers in blue; zoom in on the left	181
14.5	Manufactured marble lacking true lattice randomness (left)	181
14.6	Four superimposed Poisson-binomial processes: $s = 0$ (left), $s = 5$ (right)	184
14.7	Rayleigh test to assess if a point distribution matches that of a Poisson process	192
14.8	Realization of a 5-interlacing with $s = 0.15$ and $\lambda = 1$ : original (left), modulo $2/\lambda$ (right)	195
14.9	Locally random permutation $\sigma$ ; $\tau(k)$ is the index of $X_k$ 's closest neighbor to the right	195
14.10	Each arrow links a point (blue) to its vertex (red): $s = 0.2$ (left), $s = 1$ (right)	198
14.11	Distance between a point and its vertex ( $\lambda = s = 1$ )	199
15.1	Three orbits ( $\sigma = 0.5, 0.75, 1.25$ ) with finite Euler product: $P = \{2, 3\}$ (left) vs $\{2, 3, 5\}$ (right)	203
15.2	Distance between orbit and location $(c, 0)$ depending on $t$ on the X-axis	205
15.3	Distance between orbit and location $(c, 0)$ depending on $t$ on the X-axis	205
15.4	Distance between orbit and location $(c, 0)$ depending on $t$ on the X-axis	205
15.5	Four orbits where the “hole” (repulsion basin) is apparent	207
15.6	Three orbits with “hole” closer to the origin, showing impact of $\beta > \frac{1}{2}$ and larger $n$	207
15.7	Orbit of Dirichlet eta $\eta(z)$ when cosines are replaced by other periodic functions	211
16.1	Data linked to the melody: red curve for note frequencies, blue curve for note durations	223
16.2	R plot before Cairo (left), and after (right)	224
16.3	Intermediate (left) and last frame (right) of the video	225
16.4	Example of 90% dual confidence region for $(p, q)$	227
16.5	Minimum contrast estimation for $(\lambda, s)$ using $(p, q)$ as proxy stats	228
16.6	Non-elliptic confidence regions with various confidence levels	229
16.7	Elbow rule (right) finds $m = 3$ clusters in Brownian motion (left)	237

# List of Tables

1.1	Estimated ellipse parameters vs true values ( $n = 30$ ), for shape in Figure 1.2 . . . . .	19
1.2	First and last step of <code>curve_fitting</code> , approaching the model. . . . .	27
1.3	MSE for different methods and $\theta$ s, same data set as in Figure 1.5 . . . . .	32
1.4	MSE for different methods and $\theta$ s, same data set as in Figure 1.7 . . . . .	32
2.1	List of potential features to use in the model . . . . .	37
2.2	Statistics for selected HDT nodes (Excel version) . . . . .	40
2.3	Order of magnitude for the expectation and standard deviation of the range $R_n$ . . . . .	44
3.1	Characteristic polynomials used in the simulations . . . . .	51
7.1	Regression coefficients and performance metrics $r, s$ based on methodology . . . . .	96
7.2	Correlation matrix . . . . .	96
7.3	Best performance given $m$ (number of features) . . . . .	97
7.4	Feature comparison table (top 32 feature combinations) . . . . .	99
7.5	Feature comparison table (bottom 31 feature combinations) . . . . .	100
8.1	$R$ -squared $\rho^2$ and slope $\beta$ , on training and validation sets, median vs weighted . . . . .	106
9.1	$L_3^*(n)$ , for various sequences ( $n = 20,000$ ); “Fail” means failing the prime test . . . . .	118
11.1	High $\rho_n$ at iterations $n = 31$ and $n = 127$ points to roots 3083 and 7919 . . . . .	143
13.1	Description of top parameters used in the star cluster simulator . . . . .	166
13.2	Eight selected parameter sets covering various situations . . . . .	167
14.1	Variance attached to $F_s$ , as a function of $s$ . . . . .	177
14.2	Poisson process ( $s = \infty$ ) versus $s = 39.85$ . . . . .	197
16.1	Extract of the mapping table used to recover $(\lambda, s)$ from $(p, q)$ . . . . .	229
16.2	Eight bins: 2 features ( $A, B$ ) times 2 outcomes (Good/Bad) . . . . .	231
16.3	Amount of data collected at each level, when crawling the Internet . . . . .	235
16.4	Comparing real data with two different synthetic copies . . . . .	239

# Glossary

Autoregressive process	Auto-correlated time series, as described in section 3.4. Time-continuous versions include Gaussian processes and Brownian motions, while random walks are a discrete example; two-dimensional versions exist. These processes are essentially integrated white noise. See pages 49, 97, 135
Binning	Feature binning consists of aggregating the values of a feature into a small number of bins, to avoid overfitting and reduce the number of nodes in methods such as naive Bayes, neural networks, or decision trees. Binning can be applied to two or more features simultaneously. I discuss optimum binning in this book. See pages 37, 73, 231
Boosted model	Blending of several models to get the best of each one, also referred to as ensemble methods. The concept is illustrated with hidden decision trees in this book. Other popular examples are gradient boosting and AdaBoost. See pages 36, 244
Bootstrapping	A data-driven, model-free technique to estimate parameter values, to optimize goodness-of-fit metrics. Related to resampling in the context of cross-validation. In this book, I discuss parametric bootstrap on synthetic data that mimics the actual observations. See pages 15, 96, 192, 244
Confidence Region	A confidence region of level $\gamma$ is a 2D set of minimum area covering a proportion $\gamma$ of the mass of a bivariate probability distribution. It is a 2D generalization of confidence intervals. In this book, I also discuss dual confidence regions – the analogous of credible regions in Bayesian inference. See pages 12, 15, 18, 20, 29, 189, 190, 226, 229
Cross-validation	Standard procedure used in bootstrapping, and to test and validate a model, by splitting your data into training and validation sets. Parameters are estimated based on training set data. An alternative to cross-validation is testing your model on synthetic data with known response. See pages 15, 37, 93, 99, 167, 231, 244
Decision trees	A simple, intuitive non-linear modeling techniques used in classification problems. It can handle missing and categorical data, as well as a large number of features, but requires appropriate feature binning. Typically one blends multiple binary trees each with a few nodes, to boost performance. See pages 36, 37, 39, 41, 244, 245
Dimension reduction	A technique to reduce the number of features in your dataset while minimizing the loss in predictive power. The most well known are principal component analysis and feature selection to maximize goodness-of-fit metrics. See pages 12, 16, 245, 246
Empirical distribution	Cumulative frequency histogram attached to a statistic (for instance, nearest neighbor distances), and based on observations. When the number of observations tends to infinity and the bin sizes tend to zero, this step function tends to the theoretical cumulative distribution function of the statistic in question. See pages 16, 96, 113, 176, 179, 185, 191, 196, 198, 210, 238
Ensemble methods	A technique consisting of blending multiple models together, such as many decision trees with logistic regression, to get the best of each method and outperform each method taken separately. Examples include boosting, bagging, and AdaBoost. In this book, I discuss hidden decision trees. See pages 36, 83, 244
Explainable AI	Automated machine learning techniques that are easy to interpret are referred to as interpretable machine learning or explainable artificial intelligence. As much as possible, the methods discussed in this book belong to that category. The goal is to design black-box systems less likely to generate unexpected results with unintended consequences. See pages 13, 35, 69, 74, 83, 90, 139, 155, 193, 237

Feature selection	Features – as opposed to the model response – are also called independent variables or predictors. Feature selection, akin to <b>dimensionality reduction</b> , aims at finding the minimum subset of variables with enough <b>predictive power</b> . It is also used to eliminate redundant features and find <b>causality</b> (typically using <b>hierarchical Bayesian models</b> ), as opposed to mere correlations. Sometimes, two features have poor predictive power when taken separately, but provide improved predictions when combined together. See pages 12, 15, 37, 94, 97, 222, 230, 244, 246
Generative model	Bayesian Gaussian mixtures ( <b>GMM</b> ) combined with kernel density estimation and the <b>EM algorithm</b> is a classic modeling tool. In this book, I used <b><i>m</i>-interlacings</b> instead. Generative adversarial networks ( <b>GAN</b> ) work as follows: the generator creates new observations and the discriminator tests whether the new observations are statistically indistinguishable from training set data. When this goal is achieved, the new observations is your synthetic data. In this book, new observations are generated with <b>parametric bootstrap</b> instead. See pages 35, 52, 99, 150, 151, 153, 160, 167, 246
Goodness-of-fit	A <b>model fitting</b> criterion or metric to assess how a model or sub-model fits to a dataset, or to measure its <b>predictive power</b> on a <b>validation set</b> . Examples include <b>R-squared</b> , Chi-squared, Kolmogorov-Smirnov, error rate such as false positives and other metrics discussed in this book. See pages 15, 56, 93, 94, 231, 244, 246
Gradient methods	Iterative optimization techniques to find the minimum or maximum of a function, such as the <b>maximum likelihood</b> . When there are numerous local minima or maxima, use <b>swarm optimization</b> . Gradient methods (for instance, stochastic gradient descent or Newton's method) assume that the function is differentiable. If not, other techniques such as <b>Monte Carlo simulations</b> or the <b>fixed-point algorithm</b> can be used. Constrained optimization involves using <b>Lagrange multipliers</b> . See pages 15, 31, 55, 89
Graph structures	Graphs are found in <b>decision trees</b> , in <b>neural networks</b> (connections between <b>neurons</b> ), in <b>nearest neighbors methods</b> (NN graphs), in <b>hierarchical Bayesian models</b> , and more. See pages 70, 74, 168, 234, 235
Hyperparameter	An hyperparameter is used to control the learning process: for instance, the dimension, the number of features, parameters, layers (neural networks) or clusters (clustering problem), or the width of a filtering window in image processing. By contrast, the values of other parameters (typically node weights in <b>neural networks</b> or regression coefficients) are derived via training. See pages 29, 56, 70, 75, 101, 154, 245
Link function	A link function maps a nonlinear relationship to a linear one so that a linear model can be fit, and then mapped back to the original form using the inverse function. For instance, the <b>logit link function</b> is used in <b>logistic regression</b> . Generalizations include <b>quantile</b> functions and inverse <b>sigmoids</b> in <b>neural network</b> to work with additive (linear) parameters. See pages 13, 16, 245
Logistic regression	A generalized linear <b>regression</b> method where the binary response (fraud/non-fraud or cancer/non-cancer) is modeled as a probability via the logistic link function. Alternatives to the iterative maximum likelihood solution are discussed in this book. See pages 16, 33, 36, 40, 244, 245
Neural network	A blackbox system used for predictions, optimization, or pattern recognition especially in computer vision. It consists of layers, neurons in each layer, <b>link functions</b> to model non-linear interactions, parameters (weights associated to the connections between neurons) and <b>hyperparameters</b> . Networks with several layers are called <b>deep neural networks</b> . Also, <b>neurons</b> are sometimes called nodes. See pages 69, 73, 75, 83, 101, 244, 245
NLP	Natural language processing is a set of techniques to deal with unstructured text data, such as emails, automated customer support, or webpages downloaded with a crawler. The example discussed in section 16.5 deals with creating a keyword taxonomy based on parsing Google search results pages. See pages 36, 233
Numerical stability	This issue occurring in unstable optimization problems typically with multiple minima or maxima, is frequently overlooked and leads to poor predictions or high volatility. It is sometimes referred to as <b>ill-conditioned problems</b> . I explain how to fix it in several examples in this book, for instance in section 3.4.2. Not to be confused with numerical precision. See pages 12, 14, 59

Overfitting	Using too many unstable parameters resulting in excellent performance on the <a href="#">training set</a> , but poor performance on future data or on the <a href="#">validation set</a> . It typically occurs with numerically unstable procedures such as regression (especially polynomial regression) when the training set is not large enough, or in the presence of <a href="#">wide data</a> (more features than observations) when using a method not suited to this situation. The opposite is underfitting. See pages <a href="#">15</a> , <a href="#">92</a> , <a href="#">101</a> , <a href="#">238</a> , <a href="#">244</a> , <a href="#">246</a>
Predictive power	A metric to assess the <a href="#">goodness-of-fit</a> or performance of a model or subset of features, for instance in the context of <a href="#">dimensionality reduction</a> or <a href="#">feature selection</a> . Typical metrics include <a href="#">R-squared</a> , or <a href="#">confusion matrices</a> in classification. See pages <a href="#">38</a> , <a href="#">40</a> , <a href="#">44</a> , <a href="#">230</a> , <a href="#">232</a> , <a href="#">237</a> , <a href="#">245</a>
R-squared	A <a href="#">goodness-of-fit</a> metric to assess the predictive power of a model, measured on a <a href="#">validation set</a> . Alternatives include adjusted R-squared, mean absolute error and other metrics discussed in this book. See pages <a href="#">12</a> , <a href="#">15</a> , <a href="#">35</a> , <a href="#">56</a> , <a href="#">90</a> , <a href="#">93</a> , <a href="#">95</a> , <a href="#">97</a> , <a href="#">104</a> , <a href="#">245</a> , <a href="#">246</a>
Random number	Pseudo-random numbers are sequences of binary digits, usually grouped into blocks, satisfying properties of independent Bernoulli trials. In this book, the concept is formally defined, and strong pseudo-number generators are built and used in computer-intensive simulations. See pages <a href="#">29</a> , <a href="#">112</a> , <a href="#">119</a> , <a href="#">236</a>
Regression methods	I discuss a unified approach to all regression problems in chapter <a href="#">1</a> . Traditional techniques include linear, logistic, Bayesian, polynomial and <a href="#">Lasso regression</a> (to deal with numerical instability and <a href="#">overfitting</a> ), solved using optimization techniques, <a href="#">maximum likelihood</a> methods, linear algebra ( <a href="#">eigenvalues</a> and <a href="#">singular value decomposition</a> ) or stepwise procedures. See pages <a href="#">12</a> , <a href="#">13</a> , <a href="#">15</a> , <a href="#">16</a> , <a href="#">19</a> , <a href="#">27</a> , <a href="#">36</a> , <a href="#">40</a> , <a href="#">46</a> , <a href="#">49</a> , <a href="#">52</a> , <a href="#">56</a> , <a href="#">89</a> , <a href="#">95</a> , <a href="#">101</a> , <a href="#">108</a> , <a href="#">245</a> , <a href="#">246</a>
Supervised learning	Techniques dealing with labeled data ( <a href="#">classification</a> ) or when the response is known ( <a href="#">regression</a> ). The opposite is <a href="#">unsupervised learning</a> , for instance <a href="#">clustering</a> problems. In-between, you have <a href="#">semi-supervised learning</a> and <a href="#">reinforcement learning</a> (favoring good decisions). The technique described in chapter <a href="#">1</a> fits into unsupervised regression. <a href="#">Adversarial learning</a> is testing your model against extreme cases intended to make it fail, to build better models. See pages <a href="#">246</a>
Synthetic data	Artificial data simulated using a <a href="#">generative model</a> , typically a <a href="#">mixture model</a> , to enrich existing datasets and improve the quality of <a href="#">training sets</a> . Called <a href="#">augmented data</a> when blended with real data. See pages <a href="#">12</a> , <a href="#">13</a> , <a href="#">15</a> , <a href="#">17</a> , <a href="#">27</a> , <a href="#">29</a> , <a href="#">33</a> , <a href="#">35</a> , <a href="#">48</a> , <a href="#">52</a> , <a href="#">55</a> , <a href="#">69</a> , <a href="#">70</a> , <a href="#">75</a> , <a href="#">88</a> , <a href="#">94</a> , <a href="#">105</a> , <a href="#">125</a> , <a href="#">131</a> , <a href="#">139</a> , <a href="#">153</a> , <a href="#">160</a> , <a href="#">167</a> , <a href="#">227</a> , <a href="#">236</a> , <a href="#">238</a> , <a href="#">244</a>
Tensor	Matrix generalization with three or more dimensions. A matrix is a two-dimensional tensor. A triple summation with three indices is represented by a three-dimensional tensor, while a double summation involves a standard matrix. See pages <a href="#">69</a> , <a href="#">74</a>
Training set	Dataset used to train your model in <a href="#">supervised learning</a> . Typically, a portion of the training set is used to train the model, the other part is used as <a href="#">validation set</a> . See pages <a href="#">13</a> , <a href="#">15</a> , <a href="#">17</a> , <a href="#">20</a> , <a href="#">29</a> , <a href="#">36</a> , <a href="#">40</a> , <a href="#">56</a> , <a href="#">72</a> , <a href="#">88</a> , <a href="#">95</a> , <a href="#">101</a> , <a href="#">105</a> , <a href="#">167</a> , <a href="#">231</a> , <a href="#">244</a> , <a href="#">246</a>
Validation set	A portion of your <a href="#">training set</a> , typically 20%, used to measure the actual performance of your predictive algorithm outside the training set. In cross-validation and bootstrapping, the training and validation sets are split into multiple subsets to get a better sense of variations in the predictions. See pages <a href="#">15</a> , <a href="#">27</a> , <a href="#">41</a> , <a href="#">56</a> , <a href="#">93</a> , <a href="#">101</a> , <a href="#">167</a> , <a href="#">231</a> , <a href="#">238</a> , <a href="#">239</a> , <a href="#">244</a> , <a href="#">245</a> , <a href="#">246</a>

# Bibliography

- [1] Weighted percentiles using numpy. *Forum discussion*, 2020. StackOverflow [\[Link\]](#). 101
- [2] Jan Ackmann et al. Machine-learned preconditioners for linear solvers in geophysical fluid flows. *Preprint*, pages 1–19, 2020. arXiv:2010.02866 [\[Link\]](#). 93
- [3] Noga Alon and Joel H. Spencer. *The Probabilistic Method*. Wiley, fourth edition, 2016. [\[Link\]](#). 185
- [4] José M. Amigó, Roberto Dale, and Piergiulio Tempesta. A generalized permutation entropy for random processes. *Preprint*, pages 1–9, 2012. arXiv:2003.13728 [\[Link\]](#). 196
- [5] Luc Anselin. *Point Pattern Analysis: Nearest Neighbor Statistics*. The Center for Spatial Data Science, University of Chicago, 2016. Slide presentation [\[Link\]](#). 183
- [6] Adrian Baddeley. Spatial point processes and their applications. In Weil W., editor, *Stochastic Geometry. Lecture Notes in Mathematics*, pages 1–75. Springer, Berlin, 2007. [\[Link\]](#). 182
- [7] David Bailey and Richard Crandall. Random generators and normal numbers. *Experimental Mathematics*, 11, 2002. Project Euclid [\[Link\]](#). 128
- [8] N. Balakrishnan and C.R. Rao (Editors). *Order Statistics: Theory and Methods*. North-Holland, 1998. 185, 199
- [9] Christopher Beckham and Christopher Pal. A step towards procedural terrain generation with GANs. *Preprint*, pages 1–5, 2017. arXiv:1707.03383 [\[Link\]](#). 152
- [10] Rabi Bhattacharya and Edward Waymire. *Random Walk, Brownian Motion, and Martingales*. Springer, 2021. 130
- [11] Barbara Bogacka. *Lecture Notes on Time Series*. 2008. Queen Mary University of London [\[Link\]](#). 49
- [12] B. Bollobas and P. Erdős. Cliques in random graphs. *Mathematical Proceedings of the Cambridge Philosophical Society*, 80(3):419–427, 1976. [\[Link\]](#). 186
- [13] Miklos Bona. *Combinatorics of Permutations*. Routledge, second edition, 2012. 196
- [14] Peter Borwein, Stephen K. Choi, and Michael Coons. Completely multiplicative functions taking values in  $\{-1, 1\}$ . *Transactions of the American Mathematical Society*, 362(12):6279–6291, 2010. [\[Link\]](#). 204
- [15] Peter Borwein and Michael Coons. Transcendence of power series for some number theoretic functions. *Proceedings of the American Mathematical Society*, 137(4):1303–1305, 2009. [\[Link\]](#). 206
- [16] Oliver Bröker and Marcus J. Groteb. Sparse approximate inverse smoothers for geometric and algebraic multigrid. *Applied Numerical Mathematics*, 41(1):61–80, 2002. 90
- [17] H. M. Bui and M. B. Milinovich. Gaps between zeros of the Riemann zeta-function. *Quarterly Journal of Mathematics*, 69(2):402–423, 2018. [\[Link\]](#). 216
- [18] Bartłomiej Błaszczyzyn and Dhandapani Yogeshwaran. Clustering and percolation of point processes. *Preprint*, pages 1–20, 2013. Project Euclid [\[Link\]](#). 182
- [19] Bartłomiej Błaszczyzyn and Dhandapani Yogeshwaran. On comparison of clustering properties of point processes. *Preprint*, pages 1–26, 2013. arXiv:1111.6017 [\[Link\]](#). 182
- [20] Bartłomiej Błaszczyzyn and Dhandapani Yogeshwaran. Clustering comparison of point processes with applications to random geometric models. *Preprint*, pages 1–44, 2014. arXiv:1212.5285 [\[Link\]](#). 182
- [21] Oliver Chikumbo and Vincent Granville. Optimal clustering and cluster identity in understanding high-dimensional data spaces with tightly distributed points. *Machine Learning and Knowledge Extraction*, 1(2):715–744, 2019. 237
- [22] Keith Conrad. *L-functions and the Riemann Hypothesis*. 2018. 2018 CTNT Summer School [\[Link\]](#). 115, 201, 204, 209
- [23] Noel Cressie. *Statistic for Spatial Data*. Wiley, revised edition, 2015. 182

- [24] D.J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes*. Springer, second edition, 2002. Volume 1 – Elementary Theory and Methods. [134](#)
- [25] D.J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes*. Springer, second edition, 2014. Volume 2 – General Theory and Structure. [134](#)
- [26] Tilman M. Davies and Martin L. Hazelton. Assessing minimum contrast parameter estimation for spatial and spatiotemporal log-Gaussian Cox processes. *Statistica Neerlandica*, 67(4):355–389, 2013. [228](#)
- [27] Marc Deisenroth, A. Faisal, and Cheng Soon Ong. *Mathematics for Machine Learning*. Cambridge University Press, 2020. [\[Link\]](#). [53](#)
- [28] Harold G. Diamond and Wen-Bin Zhang. *Beurling Generalized Numbers*. American Mathematical Society, 2016. Mathematical Surveys and Monographs, Volume 213 [\[Link\]](#). [116](#), [212](#)
- [29] D.J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes – Volume I: Elementary Theory and Methods*. Springer, second edition, 2013. [183](#)
- [30] D.J. Daley and D. Vere-Jones. *An Introduction to the Theory of Point Processes – Volume II: General Theory and Structure*. Springer, second edition, 2014. [183](#)
- [31] David Coupier (Editor). *Stochastic Geometry: Modern Research Frontiers*. Wiley, 2019. [193](#)
- [32] Ding-Geng Chen (Editor), Jianguo Sun (Editor), and Karl E. Peace (Editor). *Interval-Censored Time-to-Event Data: Methods and Applications*. Chapman and Hall/CRC, 2012. [184](#)
- [33] Khaled Emam, Lucy Mosquera, and Richard Hoptroff. *Practical Synthetic Data Generation*. O'Reilly, 2020. [99](#)
- [34] Paul Erdős and Alfréd Rényi. On the evolution of random graphs. In *Publication of the Mathematical Institute of the Hungarian Academy of Sciences*, volume 5, pages 17–61, 1960. [\[Link\]](#). [186](#)
- [35] Achim Zeileis et al. Colorspace: A toolbox for manipulating and assessing colors and palettes. *Preprint*, pages 1–45, 2019. arXiv:1903.06490 [\[Link\]](#) [\[R Library\]](#). [152](#)
- [36] Arash Farahmand. *Math 55 Lecture Notes*. 2021. University of Berkeley [\[Link\]](#). [47](#), [53](#)
- [37] W. Feller. On the Kolmogorov-Smirnov limit theorems for empirical distributions. *Annals of Mathematical Statistics*, 19(2):177–189, 1948. [\[Link\]](#). [185](#), [192](#)
- [38] Nikos Frantzikinakis. Ergodicity of the Liouville system implies the Chowla conjecture. *Preprint*, pages 1–41, 2016. arXiv [\[Link\]](#). [206](#)
- [39] P. M. Gauthier. Approximating the Riemann zeta-function by polynomials with restricted zeros. *Canadian Mathematical Bulletin*, 62(3):475–478, 2018. [\[Link\]](#). [216](#)
- [40] P. A. Van Der Geest. The binomial distribution with dependent Bernoulli trials. *Journal of Statistical Computation and Simulation*, pages 141–154, 2004. [\[Link\]](#). [131](#)
- [41] Stamatia Giannarou and Tania Stathaki. Shape signature matching for object identification invariant to image transformations and occlusion. 2007. ResearchGate [\[Link\]](#). [84](#)
- [42] Minas Gjoka, Emily Smith, and Carter Butts. Estimating clique composition and size distributions from sampled network data. *Preprint*, pages 1–9, 2013. arXiv:1308.3297 [\[Link\]](#). [186](#)
- [43] B.V. Gnedenko and A. N. Kolmogorov. *Limit Distributions for Sums of Independent Random Variables*. Addison-Wesley, 1954. [135](#)
- [44] Manuel González-Navarrete and Rodrigo Lambert. Non-markovian random walks with memory lapses. *Preprint*, pages 1–14, 2018. arXiv [\[Link\]](#). [130](#)
- [45] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. [\[Link\]](#). [53](#)
- [46] Vincent Granville. Estimation of the intensity of a Poisson point process by means of nearest neighbor distances. *Statistica Neerlandica*, 52(2):112–124, 1998. [\[Link\]](#). [183](#)
- [47] Vincent Granville. *Applied Stochastic Processes, Chaos Modeling, and Probabilistic Properties of Numeration Systems*. MLTechniques.com, 2018. [\[Link\]](#). [116](#)
- [48] Vincent Granville. *Stochastic Processes and Simulations: A Machine Learning Perspective*. MLTechniques.com, 2022. [\[Link\]](#). [51](#), [59](#), [135](#), [145](#), [176](#), [177](#), [178](#), [180](#), [184](#), [186](#), [212](#), [216](#), [229](#), [243](#)
- [49] Vincent Granville, Mirko Krivanek, and Jean-Paul Rasson. Simulated annealing: A proof of convergence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16:652–656, 1996. [72](#)
- [50] Vincent Granville and Richard L Smith. Disaggregation of rainfall time series via Gibbs sampling. *NISS Technical Report*, pages 1–21, 1996. [\[Link\]](#). [107](#)
- [51] Kristen Grauman. Shape matching. 2008. University of Texas, Austin [\[Link\]](#). [87](#)
- [52] Hui Guo et al. Eyes tell all: Irregular pupil shapes reveal gan-generated faces. *Preprint*, pages 1–7, 2021. arXiv:2109.00162 [\[Link\]](#). [238](#)

- [53] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*. O'Reilly, third edition, 2023. [35](#)
- [54] Radim Halíř and Jan Flusser. Numerically stable direct least squares fitting of ellipses. *Preprint*, pages 1–8, 1998. [\[Link\]](#). [17](#), [19](#)
- [55] Peter Hall. *Introduction to the theory of coverage processes*. Wiley, 1988. [193](#)
- [56] Adam J. Harper. Moments of random multiplicative functions, II: High moments. *Algebra and Number Theory*, 13(10):2277–2321, 2019. [\[Link\]](#). [112](#), [212](#)
- [57] Adam J. Harper. Moments of random multiplicative functions, I: Low moments, better than squareroot cancellation, and critical multiplicative chaos. *Forum of Mathematics, Pi*, 8:1–95, 2020. [\[Link\]](#). [112](#), [114](#), [212](#)
- [58] Adam J. Harper. Almost sure large fluctuations of random multiplicative functions. *Preprint*, pages 1–38, 2021. arXiv [\[Link\]](#). [114](#), [206](#), [212](#)
- [59] K. Hartmann, J. Krois, and B. Waske. *Statistics and Geospatial Data Analysis*. Freie Universität Berlin, 2018. E-Learning Project SOGA [\[Link\]](#). [179](#)
- [60] D. R. Heath-Brown. Primes represented by  $x^3 + 2y^3$ . *Acta Mathematica*, 186:1–84, 2001. [\[Link\]](#). [208](#)
- [61] T. W. Hilberdink and M. L. Lapidus. Beurling Zeta functions, generalised primes, and fractal membranes. *Preprint*, pages 1–31, 2004. arXiv [\[Link\]](#). [115](#), [116](#), [212](#)
- [62] Christian Hill. *Learning Scientific Programming with Python*. Cambridge University Press, 2016. [\[Link\]](#). [19](#)
- [63] Robert V. Hogg, Joseph W. McKean, and Allen T. Craig. *Introduction to Mathematical Statistics*. Pearson, eighth edition, 2016. [\[Link\]](#). [53](#)
- [64] Zhiqiu Hu and Rong-Cai Yang. A new distribution-free approach to constructing the confidence region for multiple parameters. *PLOS One*, pages 1–13, 2013. [\[Link\]](#). [227](#)
- [65] Peter Humphries. The distribution of weighted sums of the Liouville function and Pólya's conjecture. *Preprint*, pages 1–33, 2011. arXiv [\[Link\]](#). [213](#)
- [66] Timothy D. Johnson. Introduction to spatial point processes. *Preprint*, page 2008. NeuroImaging Statistics Oxford (NISOx) group [\[Link\]](#)[\[Mirror\]](#). [183](#)
- [67] Chigozie Kelechi. Towards efficiency in the residual and parametric bootstrap techniques. *American Journal of Theoretical and Applied Statistics*, 5(5), 2016. [\[Link\]](#). [97](#)
- [68] Denis Kojevnikov, Vadim Marmer, and Kyungchul Song. Limit theorems for network dependent random variables. *Journal of Econometrics*, 222(2):419–427, 2021. [\[Link\]](#). [183](#)
- [69] Samuel Kotz, Tomasz Kozubowski, and Krzysztof Podgorski. *The Laplace Distribution and Generalizations: A Revisit with Applications to Communications, Economics, Engineering, and Finance*. Springer, 2001. [197](#)
- [70] Faraj Lagum. *Stochastic Geometry-Based Tools for Spatial Modeling and Planning of Future Cellular Networks*. PhD thesis, Carleton University, 2018. [\[Link\]](#). [182](#)
- [71] Günther Last and Mathew Penrose. *Lectures on the Poisson Process*. Cambridge University Press, 2017. [182](#)
- [72] Yuk-Kam Lau, Gerald Tenenbaum, and Jie Wu. On mean values of random multiplicative functions. *Proceedings of the American Mathematical Society*, 142(2):409–420, 2013. [\[Link\]](#). [112](#), [114](#)
- [73] Jing Lei et al. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113:1094–1111, 2018. [\[Link\]](#). [97](#)
- [74] G. Last M.A. Klatt and D. Yogeshwaran. Hyperuniform and rigid stable matchings. *Random Structures and Algorithms*, 2:439–473, 2020. [\[Link\]](#)[\[PowerPoint\]](#). [182](#)
- [75] Jorge Mateu, Frederic P Schoenberg, and David M Diez. On distances between point patterns and their applications. *Preprint*, pages 1–29, 2010. [\[Link\]](#). [183](#)
- [76] Natarajan Meghanathan. Distribution of maximal clique size of the vertices for theoretical small-world networks and real-world networks. *Preprint*, pages 1–20, 2015. arXiv:1508.01668 [\[Link\]](#). [186](#)
- [77] Masahiro Mine. Probability density functions attached to random Euler products for automorphic L-functions. *Preprint*, pages 1–38, 2020. arXiv [\[Link\]](#). [212](#), [213](#)
- [78] Christoph Molnar. *Interpretable Machine Learning*. ChristophMolnar.com, 2022. [\[Link\]](#). [97](#), [237](#)
- [79] Marc-Andreas Muendler. Linear difference equations and autoregressive processes. 2000. University of Berkeley [\[Link\]](#). [49](#)

- [80] V. Kumar Murty. Seminar on Fermat's last theorem. In *Canadian Mathematical Society – Conference Proceedings*, volume 17, Toronto, Canada, 1995. [Link]. 209
- [81] Peter Mörters and Yuval Peres. *Brownian Motion*. Cambridge University Press, 2010. Cambridge Series in Statistical and Probabilistic Mathematics, Volume 30 [Link]. 130, 134
- [82] Jesper Møller. Introduction to spatial point processes and simulation-based inference. In *International Center for Pure and Applied Mathematics (Lecture Notes)*, Lomé, Togo, 2018. [Link][Mirror]. 183, 196, 228
- [83] Jesper Møller and Rasmus P. Waagepetersen. *An Introduction to Simulation-Based Inference for Spatial Point Processes*. Springer, 2003. 183
- [84] Jesper Møller and Rasmus P. Waagepetersen. *Statistical Inference and Simulation for Spatial Point Processes*. CRC Press, 2007. 183
- [85] S. Ghosh N., Miyoshi, and T. Shirai. Disordered complex networks: energy optimal lattices and persistent homology. *Preprint*, pages 1–44, 2020. arXiv:2009.08811. 176
- [86] Saralees Nadarajah. A modified Bessel distribution of the second kind. *Statistica*, 67(4):405–413, 2007. [Link]. 197
- [87] Hasan Nasab, Mahdi Tavana, and Mohsen Yousefu. A new heuristic algorithm for the planar minimum covering circle problem. *Production and Manufacturing Research*, pages 142–155, 2014. [Link]. 193
- [88] Guillermo Navas-Palencia. Optimal binning: mathematical programming formulation. *Preprint*, pages 1–21, 2020. arXiv:2001.08025 [Link]. 37
- [89] Nathan Ng. Large gaps between the zeros of the Riemann zeta function. *Journal of Number Theory*, 128(3):509–556, 2007. [Link]. 216
- [90] Yosihiko Ogata. Cluster analysis of spatial point patterns: posterior distribution of parents inferred from offspring. *Japanese Journal of Statistics and Data Science*, 3:367–390, 2020. 182
- [91] Fred Park. Shape descriptor / feature extraction techniques. 2011. UCI iCAMP 2011 [Link]. 84
- [92] Yuval Peres and Allan Sly. Rigidity and tolerance for perturbed lattices. *Preprint*, pages 1–20, 2020. arXiv:1409.4490 [Link]. 176, 182
- [93] Carl Rasmussen and Christopher Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006. [Link]. 52
- [94] Kamron Saniee. A simple expression for multivariate Lagrange interpolation. *SIAM Undergraduate Research Online*, 2007. SIURO [Link]. 103
- [95] Karl Sigman. Notes on the Poisson process. New York NY, 2009. IEOR 6711: Columbia University course [Link]. 182
- [96] Joshua Snoke et al. General and specific utility measures for synthetic data. *Journal of the Royal Statistical Society Series A*, 181:663–688, 2018. arXiv:1604.06651 [Link]. 35
- [97] Luuk Spreeuwiers. *Image Filtering with Neural Networks: Applications and Performance Evaluation*. PhD thesis, University of Twente, 1992. 73
- [98] J. Michael Steele. Le Cam's inequality and Poisson approximations. *The American Mathematical Monthly*, 101(1):48–54, 1994. [link]. 145
- [99] Dietrich Stoyan, Wilfrid S. Kendall, Sung Nok Chiu, and Joseph Mecke. *Stochastic Geometry and Its Applications*. Wiley, 2013. 193
- [100] E.C. Titchmarsh and D.R. Heath-Brown. *The Theory of the Riemann Zeta-Function*. Oxford Science Publications, second edition, 1987. 58, 115, 201
- [101] Chris Tofallis. Fitting equations to data with the perfect correlation relationship. *Preprint*, pages 1–11, 2015. Hertfordshire Business School Working Paper[Link]. 13
- [102] D. Umbach and K.N. Jones. A few methods for fitting circles to data. *IEEE Transactions on Instrumentation and Measurement*, 52(6):1881–1885, 2003. [Link]. 14, 17
- [103] D. A. Vaccari and H. K. Wang. Multivariate polynomial regression for identification of chaotic time series. *Mathematical and Computer Modelling of Dynamical Systems*, 13(4):1–19, 2007. [Link]. 17
- [104] Remco van der Hofstad. *Random Graphs and Complex Networks*. Cambridge University Press, 2016. [Link]. 185
- [105] Yu Vizilter and Sergey Zheltov. Geometrical correlation and matching of 2D image shapes. 2012. ResearchGate [Link]. 86
- [106] Luyao Wang and Hai Cheng. Pseudo-random number generator based on logistic chaotic system. *Entropy*, 21, 2019. [Link]. 128

- [107] Mingguang Wu, Yanjie Sun, and Yaqian Li. Adaptive transfer of color from images to maps and visualizations. *Cartography and Geographic Information Science*, pages 289–312, 2021. [\[Link\]](#). 152
- [108] Lan Wu, Yongcheng Qi, and Jingping Yang. Asymptotics for dependent Bernoulli random variables. *Statistics and Probability Letters*, pages 455–463, 2012. [\[Link\]](#). 130
- [109] Oren Yakir. Recovering the lattice from its random perturbations. *Preprint*, pages 1–18, 2020. arXiv:2002.01508 [\[Link\]](#). 182
- [110] Ruqiang Yan, Yongbin Liub, and Robert Gao. Permutation entropy: A nonlinear statistical measure for status characterization of rotary machines. *Mechanical Systems and Signal Processing*, 29:474–484, 2012. 196
- [111] Shaohong Yan, Aimin Yang, et al. Explicit algorithm to the inverse of Vandermonde matrix. In *2009 International Conference on Test and Measurement*, 2009. IEEE [\[Link\]](#). 47
- [112] D. Yogeshwaran. Geometry and topology of the boolean model on a stationary point processes : A brief survey. *Preprint*, pages 1–13, 2018. Researchgate [\[Link\]](#). 183
- [113] Tonglin Zhang. A Kolmogorov-Smirnov type test for independence between marks and points of marked point processes. *Electronic Journal of Statistics*, 8(2):2557–2584, 2014. 178

# Index

- $\alpha$ -compositing, 57  
 $m$ -interlacing, 71, 183, 191, 194, 245
- A/B testing, 243  
AdaBoost, 36, 244  
additive number theory, 208, 215  
adversarial learning, 238, 246  
agent-based modeling, 151, 164  
AI art, 160, 238  
algebraic number, 117  
algorithmic bias, 238  
analytic continuation, 202, 209  
analytic function, 115  
anisotropy, 169, 190, 196  
anti-aliasing, 55, 59, 219, 224  
association rule, 232  
attraction (point process), 181  
attraction basin, 58  
attractor distribution, 135, 185, 191, 198  
augmented data, 35, 88, 151, 153, 246  
auto-correlation, 49, 116  
auto-regressive process, 49, 135, 244
- Bailey–Borwein–Plouffe formulas, 117  
Bayesian classification, 74  
Bayesian inference, 44
  - hierarchical models, 245
  - naive Bayes, 232Bernoulli trials, 226  
Berry-Esseen inequality, 113  
Bessel function, 197  
Beurling primes, 116, 212  
binning, 231
  - optimum binning, 37, 244binomial distribution, 191  
bisection method (root finding), 142  
boosted trees, 239  
bootstrapping, 15, 96, 192, 238
  - percentile method, 101boundary effect, 178, 179, 184, 185, 189–191, 195  
Brownian motion, 51, 130, 134, 151, 152, 167, 236, 244
  - Lévy flight, 135Brun's theorem, 208
- Cauchy distribution, 135  
Cauchy-Riemann equations, 115  
causality, 245  
Cayley-Hamilton theorem, 47  
CDF regression, 17  
censored data, 184  
central limit theorem, 135, 176, 226
- chaotic dynamical system, 151, 167  
character
  - principal, 204characteristic function, 197  
characteristic polynomial, 47, 49, 50, 136  
Chebyshev's bias (prime numbers), 115, 204  
checksum, 242  
Chi-squared test, 178  
Chowla conjecture, 206  
classification, 246  
clique (graph theory), 186  
cluster process, 180, 184, 191  
clustering, 246  
Collatz conjecture, 123  
collision graph, 167  
color model
  - RGB, 55, 225
  - RGBA, 55, 56, 76, 225color opacity, 161  
color transparency, 20, 153, 225  
complex random variable, 112, 212  
computational complexity, 125, 233  
computer vision, 12, 83  
confidence band, 192  
confidence interval, 44, 189, 244  
confidence level, 226, 229  
confidence region, 15, 29, 189, 190, 226, 238
  - dual region, 44, 227, 244conformal map, 14  
confusion matrix, 231, 246  
connected components, 168, 184–186, 190, 191, 234  
contour level, 160, 229  
contour plot, 229  
convergence
  - abscissa, 204, 209
  - absolute, 201, 202
  - alternating series, 202
  - conditional, 114, 204
  - Dirichlet test, 202convergence acceleration, 59  
convex linear combination, 108  
convolution of distributions, 197  
copula, 99, 238  
counting measure, 177  
covariance matrix, 90, 226  
covering (stochastic), 193  
covering problem, 192  
credible interval, 44  
credible region (Bayesian), 227, 244  
cross-validation, 15, 167, 231

cuban primes, 208  
 curve fitting, 26  
 data video, 151, 160  
 decision tree, 36  
 Dedekind zeta function, 115, 212  
 deep neural network, 73, 245  
 dense set (topology), 206  
 density estimation, 183  
 diamond-square algorithm, 152  
 Diehard tests of randomness, 114  
 dimensionality reduction, 16  
 Dirichlet character, 115, 116, 203, 208  
     modulo 4, 204, 209, 210  
 Dirichlet eta function, 216  
 Dirichlet functional equation, 115, 209, 210  
 Dirichlet series, 112  
 Dirichlet's theorem, 115, 204, 206, 210  
 Dirichlet- $L$  function, 115, 203, 210  
 dissimilarity metric, 233  
 distributed architecture, 230  
 distribution  
     Cauchy, 135  
     Fréchet, 51, 135  
     Gaussian, 226  
     generalized logistic, 90, 180  
     Hotelling, 227  
     Laplace, 197  
     logistic, 16  
     Lévy, 135  
     modified Bessel, 197  
     Poisson-binomial, 145, 199  
     Poisson-exponential, 176  
     Rademacher, 112, 113  
     Rayleigh, 191, 192, 198  
     Weibull, 51, 135, 191  
 domain of attraction, 185  
 dot product, 14  
 dummy variable, 36  
 dyadic map, 116  
 dynamical systems, 116, 185  
     chaotic systems, 151, 167  
     dyadic map, 116  
     ergodicity, 116  
     logistic map, 116  
     shift map, 116  
     stochastic, 152  
 edge effect (statistics), 184  
 eigenvalue, 13, 52, 90, 246  
     power iteration, 92  
 elbow rule, 139, 184, 191, 236  
 elliptic curve, 208  
 EM algorithm, 35, 245  
 empirical distribution, 16, 96, 176, 179, 185, 191, 196,  
     198, 210, 238  
     multivariate, 113  
 empirical quantiles, 101  
 ensemble methods, 36, 83, 239  
 entropy, 170, 196, 232  
 equidistribution modulo 1, 119  
 equilibrium distribution, 152  
 Erdős-Rényi model, 186  
 ergodicity, 116, 152, 189, 191, 197  
 Euler product, 112, 202, 209  
     random, 212  
 Euler's transform, 216  
 evolutionary process, 152  
 experimental design, 243  
 experimental math, 56, 200  
 explainable AI, 13, 35, 74, 83, 90, 139, 155, 193, 237  
 exploratory analysis, 242  
 exponential decay, 40  
 exponential sums, 209  
 extrapolation, 108  
 extreme value theory, 135, 198  
 feature attribution, 237  
 feature importance, 237  
 feature selection, 15, 97, 230  
 Fermat's last theorem, 209  
 fixed-point algorithm, 59, 89, 139, 245  
 flag vector, 232, 243  
 Fourier transform, 197  
 fractal dimension, 51  
 fractional part function, 118  
 Frobenius norm, 90  
 Fruchterman and Rheingold algorithm, 234  
 Fréchet distribution, 51, 135  
 fuzzy classification, 56  
 Gamma function, 51, 135  
 GAN (generative adversarial networks), 35, 245  
 Gaussian circle problem, 215  
 Gaussian distribution, 226  
 Gaussian mixture, 35, 70  
 Gaussian primes, 115, 211  
 Gaussian process, 49, 244  
 general linear model, 13  
 generalized linear model, 13, 48  
 generalized logistic distribution, 90, 190  
 generative adversarial networks, 35, 155, 238, 245  
 generative AI, 151, 164  
 generative model, 35, 52, 99, 150, 151, 153, 160, 167,  
     246  
 geostatistics, 102  
 Glivenko-Cantelli theorem, 210  
 GMM (Gaussian mixture model), 35, 245  
 Goldbach's conjecture, 208  
 goodness-of-fit, 56, 231  
 GPU-based clustering, 71  
 gradient (optimization), 139  
 gradient boosting, 244  
 gradient operator, 15  
 graph, 184  
     collision graph, 167  
     connected components, 168, 190, 234  
     directed, 168  
     edge, 184  
     Fruchterman-Reingold, 168  
     nearest neighbor graph, 186, 190  
     node, 184, 186

random graph, 185  
     random nearest neighbor graph, 185  
     tree, 168  
     undirected, 184–186, 191  
     vertex, 184  
 graph database, 234  
 graph theory, 184  
 GraphViz, 168  
 greedy algorithm, 215  
 grid search, 139, 154

half-tone (music), 222  
 Hartman–Wintner theorem, 130  
 hash table, 126, 170, 195, 232, 233  
     sparse, 232  
 Hausdorff distance, 87  
 Hellinger distance, 238  
 hexagonal lattice, 184  
 hidden decision trees, 36, 37, 244  
 hidden layer, 73  
 hidden process, 176, 194, 198  
 hierarchical clustering, 73, 233  
 Hilbert primes, 211  
 histogram equalization, 71, 73  
 Hoeffding inequality, 133  
 homogeneity (point process), 145, 183  
 Hotelling distribution, 227  
 Hurst exponent, 51  
 hyperparameter, 29, 56, 103, 154

identifiability, 194, 196  
 ill-conditioned problem, 26, 52, 92, 245  
 image segmentation, 73  
 imputation (missing values), 238  
 index  
     index discrepancy, 196  
 intensity (stochastic process), 176, 183, 190  
 interarrival times, 134, 176, 185, 189, 196  
     standardized, 197  
 interlaced processes, 183  
 Internet of Things, 176  
 inverse distance weighting, 104  
 inverse square law, 164  
 iterated logarithm, 113, 114, 130  
 Itô integral, 52

K-means clustering, 31, 32  
 key-value pair, 37, 232  
 Kolmogorov-Smirnov test, 113, 178, 185  
 Kronecker's theorem, 206, 214

Lagrange interpolation, 52  
 Lagrange multiplier, 15, 245  
 Laplace distribution, 197  
 Lasso regression, 15, 246  
 lattice, 182  
     perturbed lattice, 176  
     shifted, 184  
     stretched, 184  
 law of the iterated logarithm, 113, 114, 130, 206, 212  
 Le Cam's theorem, 145, 177  
 least absolute residuals, 101

link function, 13, 16  
 Liouville function, 203, 214  
 log-polar map, 14  
 logistic distribution, 16, 183  
 logistic map, 116  
 logistic regression, 16  
     unsupervised, 33  
 logit function, 245  
 Lévy distribution, 135  
 Lévy flight, 135

Map-reduce, 230  
 marketing attribution, 243  
 Markov chain, 49  
     MCMC, 112  
 Mathematica, 229  
 MaxCliqueDyn algorithm, 186  
 maximum likelihood estimation, 228, 245, 246  
 mean squared error, 15, 30  
 medoid, 31  
 Mersenne twister, 29, 116, 119, 132  
 Mertens function, 203  
 minimum contrast estimation, 154, 193, 196, 228  
 mixture model, 29, 45, 152, 160, 176, 183, 184, 191, 229, 246  
     blending, 152  
 model fitting, 56, 245  
 model identifiability, 15  
 modulus (complex number), 136, 202  
 Monte Carlo simulations, 112, 245  
 morphing (computer vision), 151  
 moving average, 141  
 multiplicative function  
     completely multiplicative, 112, 114, 203, 204, 215  
     Rademacher, 112  
 Möbius function, 203

N-body problem, 164  
 n-gram (NLP), 233  
 naive Bayes, 232, 244  
 natural language processing, 36, 233  
 nearest neighbor interpolation, 101, 104  
 nearest neighbors, 176, 186, 192, 245  
     nearest neighbor distances, 190–192, 194, 198  
     nearest neighbor graph, 190

NetworkX, 168  
 neural network, 73  
     hidden layer, 73  
     hyperparameter, 75  
     neuron, 73, 245  
     seq2seq, 150  
     sparse, 69  
     very deep, 73  
 Newton's method, 139  
 node (decision tree), 37, 239, 244  
     perfect node, 44  
     usable node, 38  
 normal number, 113, 206, 210  
     strongly normal, 114  
 numerical stability, 47

Omega function, 203, 209

order statistics, 198  
 ordinary least squares, 49, 101  
 outliers, 198, 236  
 overfitting, 15, 196, 238, 244  
 palette, 151, 225  
 parametric bootstrap, 20, 29, 35, 97, 192, 238, 244, 245  
 partial least squares, 13  
 path (graph theory), 184  
 percentile bootstrap, 101  
 permutation
 

- entropy, 196
- random permutation, 195

 perturbed lattices, 176  
 Plotly, 160  
 point count distribution, 177, 190, 193  
 point process
 

- attractive, 191
- cluster process
  - Matérn, 182
  - Neyman-Scott, 182
- non-homogeneous, 145, 183
- perturbed lattice process, 182
- radial, 183
- renewal process, 182
- repulsive, 181

 Poisson point process, 134, 145, 176, 190  
 Poisson-binomial distribution, 145, 176, 199  
 Poisson-exponential distribution, 176  
 positive semidefinite (matrix), 48, 91  
 power iteration, 92  
 preconditioning, 92  
 prediction interval, 15, 96, 101  
 predictive power, 37, 44, 231, 232, 237  
 prime test (of randomness), 114, 125, 131  
 principal component analysis, 47, 237, 244  
 probability generating function, 131  
 proxy space, 229  
 pseudo-inverse matrix, 48  
 pseudo-random numbers, 132, 236
 

- combined generators, 128
- congruential generator, 119
- Diehard tests, 114, 126
- Mersenne twister, 119, 132, 153
- prime test, 114, 131
- strongly random, 114, 117
- TestU01, 114

 Pólya conjecture, 205  
 quadratic irrational, 116, 119, 125  
 quantile, 227, 245
 

- empirical, 101, 238
- weighted, 101

 quantile function, 99, 176, 180, 191  
 quantile regression, 15  
 R-squared, 15, 35, 154  
 Rademacher distribution, 113  
 Rademacher function, 112, 206, 212
 

- random, 114

 random function, 144  
 random graph, 185, 186  
 random multiplicative function, 112
 

- Rademacher, 114

 random permutation, 195  
 random variable
 

- complex, 112

 random walk, 130, 154, 244
 

- first hitting time, 131, 134
- zero crossing, 130

 Rayleigh distribution, 191, 192, 198  
 Rayleigh test, 191  
 records, 198  
 regression splines, 13  
 regular expression, 233, 242  
 reinforcement learning, 246  
 rejection sampling, 239  
 renewal process, 182  
 repulsion (point process), 181, 192  
 repulsion basin, 202  
 resampling, 96, 192  
 Riemann Hypothesis, 107
 

- Generalized, 114, 204, 208, 210

 Riemann zeta function, 112, 115, 210, 212  
 root mean squared error, 56  
 scaling factor, 190, 198  
 seed (random number generator), 126, 153, 239  
 semi-supervised learning, 246  
 shape signature, 84  
 Shapley value, 237  
 Shepard's method, 104  
 shift map, 116  
 sigmoid function, 245  
 simplex, 213  
 singular value decomposition, 13, 246  
 singularity, 170  
 six degrees of separation, 235  
 Sklar's theorem, 238  
 smoothing parameter, 103  
 spatial statistics, 102, 182  
 spectral domain, 152  
 square root (matrix), 48, 91  
 square-free integer, 113, 126, 206  
 stable distribution, 135, 153, 197  
 state space, 152  
 stationary distribution, 52  
 stationary process, 49, 135, 152, 167, 178, 183, 190  
 stepwise regression, 98  
 stochastic convergence, 152  
 stochastic function, 51  
 stochastic geometry, 193  
 stochastic process, 176  
 stochastic residues, 194  
 stop word (NLP), 233  
 stretching (point process), 184  
 superimposition (point processes), 183  
 supervised classification, 71  
 surface plot, 160  
 swarm optimization, 27, 245  
 synthetic data, 13, 27, 29, 52, 88, 90, 125, 131, 139, 153, 160, 167, 200, 227, 238

synthetic metric, 232  
Tarjan's algorithm, 234  
tensor, 74  
text normalization, 233  
Theil-Sen estimator, 101  
time series, 50  
    auto-regressive, 50, 135  
    disaggregation, 107  
    Hurst exponent, 51  
    non-periodic, 25  
total least squares, 13  
training set, 101, 167, 231  
transcendental number, 117  
transformer, 73, 150  
tree (graph theory), 168  
twin primes, 208  
  
universality property, 202, 206, 208  
unsupervised clustering, 71  
unsupervised learning, 33, 246  
  
validation set, 15, 56, 101, 167, 231, 238, 239  
Vandermonde matrix, 47, 52  
vertex, 176, 184, 185, 198  
video compression  
    FFmpeg, 55, 59  
  
Waring's problem, 208  
Watts and Strogatz model, 235  
Weibull distribution, 51, 135, 191, 198  
weighted least squares, 13  
weighted quantiles, 101  
weighted regression, 16  
white noise, 27, 49, 135, 244  
wide data, 246  
  
XOR operator, 119