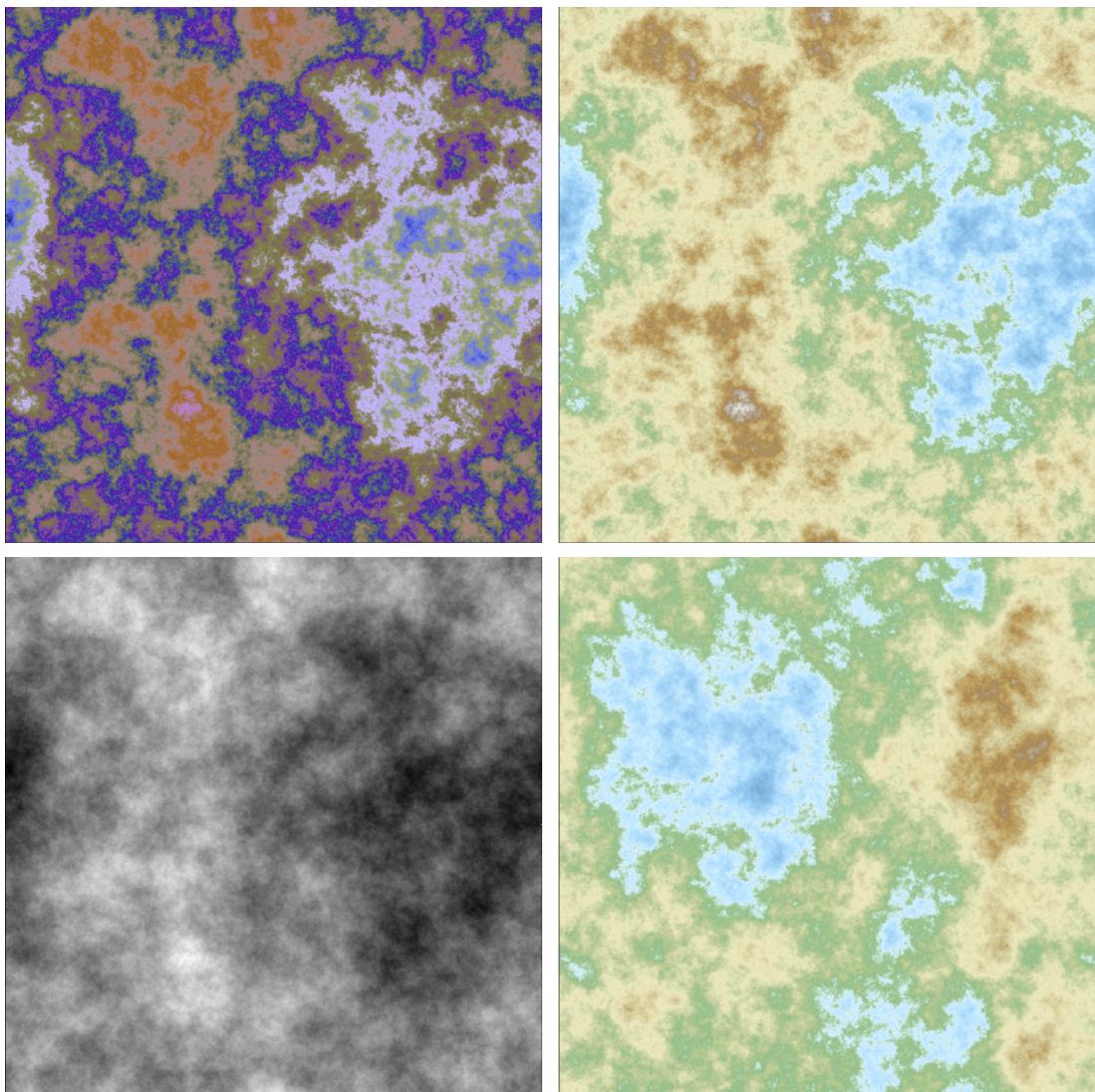

Synthetic Data and Generative AI



Preface

This book was first started in December 2022 and has been revised and augmented multiple times since the first writing. It now covers all the modern techniques on the subject as well as efficient proprietary methods developed by the author, with real industry use cases and Python implementations. The goal is to quickly help you pick up the right tool and run the code on your own dataset, in very little time. Yet the author provides enough background so that the reader understands all the aspects and interconnections of the methods involved, their strengths and weaknesses, potential enhancements, rule of thumbs, and best practices. Research-level material is also present throughout the book, and explained in simple English.

What is synthetic data and why use it?

Synthetic data is more than simulations, mimicking real data, fake (gibberish) data or noise injection to add variations to real data. It is defined by its usage and purpose. Four broad areas include:

- Data augmentation to produce richer training sets for predictive modeling; it leads to more robust predictions and reduced overfitting. For instance, to produce a better version of ChatGPT or better detection of cancer from medical images or tabular data.
- Generation of diversified data to test and benchmark machine learning algorithms, to identify their limits or to understand and improve black-box systems. Sensitivity testing fits in this category.
- Increasing security and compliance with data protection laws by strongly anonymizing data (especially for data sharing purposes), as well as reduction of algorithm bias impacting minorities.
- Data re-balancing in the presence of small segments (fraud / non-fraud, minority group), and smart data imputation. It is also useful in the presence of small samples with many features, when the data is difficult to obtain: for instance, clinical trials.

The data can be tabular (transactional), time series, graphs or consisting of images, videos, sound, text, spatial information or the result of agent-based systems. The goal is to identify and reproduce the structure (such as the autocorrelation function, shape, or correlation structure) rather than replicating the original data itself. In some instances (benchmarking), no real data is even needed.

Several techniques can be used for synthetization: GAN (generative adversarial networks), GMM (Gaussian mixture models) and other statistical models, interpolation, parametric noise with a target correlation structure, and more. Many metrics are available to assess the quality, be it cross-validation, ROC curves, statistical summaries, or Hellinger and related distances. All this material is reviewed in this book. In particular, chapter 10 discusses a GAN with replicable output especially designed for synthetization, illustrated on tabular data.

Book contents and target audience

This book covers the foundations of generative models and data synthetization. Emphasis is on scalability, automation, testing, optimizing, and interpretability (explainable AI). Models (including GMM, GAN and copulas) are often used to create rich synthetic data, augment real data, or to test and benchmark various methods. Many machine learning algorithms are revisited, simplified, unified, and generalized. 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: it shows another usage of synthetization, with an application to meteorites shapes, for instance when the goal is to classify these celestial bodies.

With a focus on applications, synthetization and simulations, the book also covers 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, strong 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.

Chapter 10 illustrates the use of copulas to produce synthetic data, applied to a well-known insurance dataset. It also features both GAN (generative adversarial networks) and copulas applied to an health industry data set, comparing results and showing how both methods can be blended for better synthetization and predictions, or even for data compression. Agent-based modeling and GIS applications are also covered, with interpolation techniques used for synthetization: fractal-like terrain generation with the diamond-square algorithm, disaggregation of ocean tides time series, and geospatial interpolation of temperatures in the Chicago area.

Image and video generation include star clusters evolving over time and bound by gravity, providing potential scenarios about the past and future of our universe, or to synthesize collision graphs. It also allows you to explore alternative universes, for instance with negative masses. Chapters 16 and 17 are more advanced and may be skipped in introductory classes. The former focuses on point processes as a simple alternative to GMM. The later features synthetic multiplicative functions to discover new insights about a famous mathematical conjecture: the Riemann Hypothesis.

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 solid introduction to scientific programming. The code, datasets, spreadsheets and data visualizations are also on GitHub, [here](#). 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. 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 synthetic data and generative AI.

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

List of Figures	10
List of Tables	12
1 Machine Learning Cloud Regression and Optimization	13
1.1 Introduction: circle fitting	13
1.1.1 Previous versions of my method	14
1.2 Methodology, implementation details and caveats	15
1.2.1 Solution, R-squared and backward compatibility	15
1.2.2 Upgrades to the model	16
1.3 Case studies	17
1.3.1 Logistic regression, two ways	17
1.3.2 Ellipsoid and hyperplane fitting	18
1.3.2.1 Curve fitting: 250 examples in one video	18
1.3.2.2 Confidence region for the fitted ellipse: application to meteorite shapes	19
1.3.2.3 Python code	20
1.3.3 Non-periodic sum of periodic time series: ocean tides	26
1.3.3.1 Numerical instability and how to fix it	27
1.3.3.2 Python code	28
1.3.4 Fitting a line in 3D, unsupervised clustering, and other generalizations	29
1.3.4.1 Example: confidence region for the cluster centers	30
1.3.4.2 Exact solution and caveats	31
1.3.4.3 Comparison with K-means clustering	32
1.3.4.4 Python code	34
1.4 Connection to synthetic data: meteorites, ocean tides	36
2 A Simple, Robust and Efficient Ensemble Method	37
2.1 Introduction	37
2.2 Methodology	38
2.2.1 How hidden decision trees (HDT) work	38
2.2.2 NLP case study: summary and findings	39
2.2.3 Parameters	40
2.2.4 Improving the methodology	40
2.3 Implementation details	40
2.3.1 Correcting for bias	40
2.3.1.1 Time-adjusted scores	41
2.3.2 Excel spreadsheet	41
2.3.3 Python code and dataset	41
2.4 Model-free confidence intervals and perfect nodes	45
2.4.1 Interesting asymptotic properties of confidence intervals	45
3 Gentle Introduction to Linear Algebra – Synthetic Time Series	47
3.1 Power of a matrix	47
3.2 Examples, generalization, and matrix inversion	48
3.2.1 Example with a non-invertible matrix	49
3.2.2 Fast computations	49
3.2.3 Square root of a matrix	49
3.3 Application to machine learning problems	50
3.3.1 Markov chains	50
3.3.2 Time series: auto-regressive processes	50
3.3.3 Linear regression	51

3.4	Mathematics of auto-regressive time series	51
3.4.1	Simulations: curious fractal time series	52
3.4.1.1	White noise: Fréchet, Weibull and exponential cases	52
3.4.1.2	Illustration	52
3.4.2	Solving Vandermonde systems: a numerically stable method	53
3.5	Math for Machine Learning: Must-Read Books	54
4	Image and Video Generation	55
4.1	Introduction	55
4.2	Applications	56
4.2.1	Spatial time series	56
4.2.2	Prediction intervals in any dimensions	56
4.2.3	Supervised classification of an infinite dataset	57
4.2.3.1	Machine learning perspective	57
4.2.3.2	Six challenging problems	58
4.2.3.3	Mathematical background: the Riemann Hypothesis	58
4.2.3.4	Partial solutions to the six challenging problems	59
4.2.4	Algorithms with chaotic convergence	60
4.3	Python code	60
4.3.1	Path simulation	60
4.3.2	Visual convergence analysis in 2D	63
4.3.3	Supervised classification	64
4.4	Visualizations	67
5	Synthetic Clusters and Alternative to GMM	70
5.1	Introduction	70
5.2	Generating the synthetic data	71
5.2.1	Simulations with logistic distribution	71
5.2.2	Mapping the raw observations onto an image bitmap	72
5.3	Classification and unsupervised clustering	72
5.3.1	Supervised classification based on convolution filters	73
5.3.2	Clustering based on histogram equalization	73
5.3.3	Fractal classification: deep neural network analogy	74
5.3.4	Generalization to higher dimensions	75
5.3.5	Towards a very fast implementation	75
5.4	Python code	76
5.4.1	Fractal classification	77
5.4.2	GPU classification and clustering	79
5.4.3	Home-made graphic library	81
6	Shape Classification and Synthetization via Explainable AI	84
6.1	Introduction	84
6.2	Mathematical foundations	84
6.3	Shape signature	85
6.3.1	Weighted centroid	85
6.3.2	Computing the signature	86
6.3.3	Example	87
6.4	Shape comparison	87
6.4.1	Shape classification	88
6.5	Application	88
6.6	Exercises	89
7	Synthetic Data, Interpretable Regression, and Submodels	90
7.1	Introduction	90
7.2	Synthetic data sets and the spreadsheet	91
7.2.1	Correlation structure	91
7.2.2	Standardized regression	92
7.2.3	Initial conditions	92
7.2.4	Simulations and Excel spreadsheet	92
7.3	Damping schedule and convergence acceleration	93
7.3.1	Spreadsheet implementation	93
7.3.2	Interpretable regression with no overfitting	94
7.3.3	Adaptive damping	94

7.4	Performance assessment on synthetic data	94
7.4.1	Results	95
7.4.2	Distribution-free confidence intervals	97
7.4.2.1	Parametric bootstrap	98
7.5	Feature selection	98
7.5.1	Combinatorial approach	98
7.5.2	Stepwise approach	99
7.6	Conclusion	100
8	From Interpolation to Fuzzy Regression	102
8.1	Introduction	102
8.2	Original version	103
8.3	Full, non-linear model in higher dimensions	103
8.3.1	Geometric proximity, weights, and numerical stability	104
8.3.2	Predicted values and prediction intervals	104
8.3.3	Illustration, with spreadsheet	105
8.3.3.1	Output fields	106
8.4	Results	106
8.4.1	Performance assessment	106
8.4.2	Visualization	107
8.4.3	Amplitude restoration	107
8.5	Exercises	108
8.6	Python source code and datasets	109
9	New Interpolation Methods for Synthetization and Prediction	113
9.1	First method	113
9.1.1	Example with infinite summation	114
9.1.2	Applications: ocean tides, planet alignment	115
9.1.3	Problem in two dimensions	116
9.1.4	Spatial interpolation of the temperature dataset	117
9.2	Second method	119
9.2.1	From unstable polynomial to robust orthogonal regression	120
9.2.2	Using orthogonal functions	120
9.2.3	Application to regression	120
9.3	Python code	121
9.3.1	Time series interpolation	121
9.3.2	Geospatial temperature dataset	124
9.3.3	Regression with Fourier series	127
10	Synthetic Tabular Data: Copulas vs enhanced GANs	129
10.1	Sensitivity analysis, bias reduction and other uses of synthetic data	130
10.2	Using copulas to generate synthetic data	130
10.2.1	The insurance dataset: Python code and results	131
10.2.2	Potential improvements	133
10.3	Synthetization: GAN versus copulas	134
10.3.1	Parameterizing the copula quantiles combined with gradient descent	134
10.3.2	Feature clustering to break a big problem into smaller ones	134
10.4	Deep dive into generative adversarial networks (GAN)	135
10.4.1	Open source libraries and references	135
10.4.2	Synthesizing medical data with GAN	136
10.4.2.1	Hyperparameters	137
10.4.2.2	GAN: Main steps	138
10.4.3	Initial results	139
10.4.4	Fine-tuning the hyperparameters	140
10.4.5	Enhanced GAN: methodology and results	140
10.5	Comparing GANs with the copula method	141
10.5.1	Conclusion: getting the best out of copulas and GAN	143
10.6	Data synthetization explained in one picture	143
10.7	Python code: GAN to synthesize medical data	144
10.7.1	Classification problem	144
10.7.2	GAN method	145
10.7.3	GAN Evaluation and post-classification	147

11 High Quality Random Numbers for Data Synthetization	149
11.1 Introduction	149
11.2 Pseudo-random numbers	150
11.2.1 Strong pseudo-random numbers	150
11.2.1.1 New test of randomness for PRNGs	151
11.2.1.2 Theoretical background: the law of the iterated logarithm	151
11.2.1.3 Connection to the Generalized Riemann Hypothesis	151
11.2.2 Testing well-known sequences	152
11.2.2.1 Reverse-engineering a pseudo-random sequence	153
11.2.2.2 Illustrations	154
11.3 Python code	156
11.3.1 Fixes to the faulty random function in Python	156
11.3.2 Prime test implementation to detect subtle flaws in PRNG's	156
11.3.3 Special formula to compute 10 million digits of $\sqrt{2}$	159
11.4 Military-grade PRNG Based on Quadratic Irrationals	162
11.4.1 Fast algorithm rooted in advanced analytic number theory	162
11.4.2 Fast PRNG: explanations	163
11.4.3 Python code	163
11.4.4 Computing a digit without generating the previous ones	165
11.4.5 Security and comparison with other PRNGs	165
11.4.5.1 Important comments	166
11.4.6 Curious application: a new type of lottery	166
12 Some Unusual Random Walks	167
12.1 Symmetric unbiased constrained random walks	167
12.1.1 Three fundamental properties of pure random walks	167
12.1.2 Random walks with more entropy than pure random signal	168
12.1.2.1 Applications	168
12.1.2.2 Algorithm to generate quasi-random sequences	169
12.1.2.3 Variance of the modified random walk	169
12.1.3 Random walks with less entropy than pure random signal	170
12.2 Related stochastic processes	171
12.2.1 From Brownian motions to clustered Lévy flights	171
12.2.2 Integrated Brownian motions and special auto-regressive processes	172
12.3 Python code	173
12.3.1 Computing probabilities and variances attached to S_n	173
12.3.2 Path simulations	174
13 Divergent Optimization Algorithm and Synthetic Functions	176
13.1 Introduction	176
13.1.1 The problem, with illustration	177
13.2 Non-converging fixed-point algorithm	178
13.2.1 Trick leading to intuitive solution	178
13.2.2 Root detection: method and parameters	178
13.2.3 Case study: factoring a product of two large primes	179
13.3 Generalization with synthetic random functions	179
13.3.1 Example	181
13.3.2 Connection to the Poisson-binomial distribution	182
13.3.2.1 Location of next root: guesstimate	182
13.3.2.2 Integer sequences with high density of primes	182
13.3.3 Python code: finding the optimum	183
13.4 Smoothing highly chaotic curves	184
13.4.1 Python code: smoothing	184
13.5 Connection to synthetic data: random functions	187
14 Synthetic Terrain Generation and AI-generated Art	188
14.1 Introduction	188
14.2 Terrain generation and the evolutionary process	190
14.2.1 Morphing and non-linear palette operations	190
14.2.2 The diamond-square algorithm	190
14.2.3 The evolutionary process	191
14.2.4 Finding optimum parameters	191

14.2.5	Mimicking real terrain: the synthesis step	191
14.3	Python code	192
14.3.1	Producing data videos with four sub-videos in parallel	192
14.3.2	Main program	193
14.4	AI-generated art with 3D contours	197
14.4.1	Python code using Matplotlib	198
14.4.2	Python code using Plotly	199
14.4.3	Tips to quickly solve new problems	200
15	Synthetic Star Cluster Generation with Collision Graphs	201
15.1	Introduction	201
15.2	Model parameters and simulation results	202
15.2.1	Explanation of color codes	202
15.2.2	Detailed description of top parameters	202
15.2.3	Interesting parameter sets	203
15.3	Analysis of star collisions and collision graph	204
15.3.1	Weighted directed graphs: visualization with NetworkX	205
15.3.2	Interesting findings: how the universe got started	205
15.4	Animated data visualizations	206
15.5	Python code and computational issues	207
15.5.1	Simulating the real and synthetic universes	207
15.5.2	Visualizing collision graphs	211
16	Perturbed Lattice Point Process: Alternative to GMM	213
16.1	Perturbed lattices: definition and properties	213
16.1.1	Point counts distribution	214
16.1.2	Periodicity and amplitude of point count expectations	214
16.1.3	Testing the independence of point counts	215
16.1.3.1	Results and Interpretation	216
16.1.3.2	About the Spreadsheet	217
16.2	Cluster processes and nearest neighbor graphs	217
16.2.1	Synthetic, semi-rigid cluster structures	217
16.2.2	Python code to generate cluster processes	219
16.2.3	References on cluster processes	219
16.2.4	Superimposed perturbed lattices: an alternative to mixture models	220
16.2.4.1	Hexagonal lattice, nearest neighbors	221
16.2.4.2	Exercises: nearest neighbor graphs, size of connected components	222
16.2.4.3	Python code to compute connected components	223
16.3	Statistical inference for point processes	225
16.3.1	Estimation of Core Parameters	225
16.3.1.1	Intensity	226
16.3.1.2	Scaling factor	226
16.3.1.3	Alternative estimation method	226
16.3.2	Spatial statistics, nearest neighbors, clustering	227
16.3.2.1	Inference for two-dimensional processes	227
16.3.2.2	Other possible tests	227
16.3.2.3	Rayleigh test	228
16.3.2.4	Exercises	229
16.4	Special topics	230
16.4.1	Minimum contrast estimation and explainable AI	230
16.4.2	Model identifiability, hard-to-detect patterns	231
16.4.2.1	Stochastic residues	231
16.4.3	Hidden model and random permutations	231
16.4.4	Retrieving the F distribution	233
16.4.4.1	Theoretical values obtained by simulations	233
16.4.4.2	Retrieving F from the interarrival times distribution	234
16.4.5	Record distances between an observed point and its vertex	234
16.4.5.1	Distribution of records	235
16.4.5.2	Distribution of arrival times for records	236
17	Synthetizing Multiplicative Functions in Number Theory	237
17.1	Introduction	237

17.1.1	Key concepts and terminology	238
17.1.2	Orbits and holes	238
17.1.3	Industrial Applications	238
17.2	Euler products	239
17.2.1	Finite Euler Products	239
17.2.1.1	Generalization using Dirichlet characters	240
17.2.2	Infinite Euler products	241
17.2.2.1	Special products	241
17.2.2.2	Probabilistic properties and conjectures	242
17.3	Finite Dirichlet series and generalizations	243
17.3.1	Finite Dirichlet series	243
17.3.2	Non-trivial cases with infinitely many primes and a hole	245
17.3.2.1	Sums of two cubes, or cuban primes	245
17.3.2.2	Primes associated to elliptic curves	245
17.3.2.3	Analytic continuation, convergence, and functional equation	246
17.3.2.4	Hybrid Dirichlet-Taylor series	246
17.3.3	Riemann Hypothesis with cosines replaced by wavelets	247
17.3.4	Riemann Hypothesis for Beurling primes	248
17.3.5	Stochastic Euler products	249
17.4	Exercises	250
17.5	Python code	253
17.5.1	Computing the orbit of various Dirichlet series	253
17.5.2	Creating videos of the orbit	256
18	Text, Sound Generation and Other Topics	259
18.1	Sound generation: let your data sing!	259
18.1.1	From data visualizations to videos to data music	259
18.1.2	References	260
18.1.3	Python code	260
18.2	Data videos and enhanced visualizations in R	261
18.2.1	Cairo library to produce better charts	261
18.2.2	AV library to produce videos	262
18.3	Dual confidence regions	263
18.3.1	Case study	263
18.3.2	Standard confidence region	263
18.3.3	Dual confidence region	264
18.3.4	Simulations	264
18.3.5	Original problem with minimum contrast estimators	265
18.3.6	General shape of confidence regions	266
18.4	Fast feature selection based on predictive power	267
18.4.1	How cross-validation works	268
18.4.2	Measuring the predictive power of a feature	268
18.4.3	Efficient implementation	269
18.5	NLP: taxonomy creation and text generation	270
18.5.1	Designing a keyword taxonomy	270
18.5.2	Fast clustering algorithm for keyword data	271
18.5.2.1	Computational complexity	271
18.5.2.2	Smart crawling of the whole Internet and a bit of graph theory	272
18.6	Automated detection of outliers and number of clusters	273
18.6.1	Black-box elbow rule to detect outliers	273
18.7	Advice to beginners	274
18.7.1	Getting started and learning how to learn	274
18.7.1.1	Getting help	275
18.7.1.2	Beyond Python	275
18.7.2	Automated data cleaning and exploratory analysis	275
18.7.3	Example of simple analysis: marketing attribution	276
Glossary		277
Bibliography		280
Index		285

List of Figures

1.1	Fitted ellipse (blue), given the training set (red) distributed around a partial arc	19
1.2	Confidence region in blue, $n = 30$ training set points; 50 training sets (left) vs 150 (right)	20
1.3	Three non-periodic time series made of periodic terms (see section 17.2.2.1)	26
1.4	Training set (red), validation set (orange), fitted curve (blue) and model (gray)	27
1.6	Biased confidence region for (θ_A^*, θ_B^*) ; same example as in Figure 1.5; true value is $(0.5, 1.0)$	30
1.5	Finding the two centers θ_A^*, θ_B^* in sample 39; $n = 1000$	31
1.7	Challenging mixture, requiring $p_A = 3, p_B = 1$ to identify the two cluster centers	32
2.1	Output from the Excel version of HDT	42
3.1	AR models, classified based on the types of roots of the characteristic polynomial	53
4.1	Scatterplot observations vs. predicted values, with prediction intervals (in any dimension)	67
4.2	Comets orbiting the sun: simulation	67
4.3	Comets orbiting the sun: snapshot in time	68
4.4	Three orbits of $\eta(\sigma + it)$: $\sigma = 0.5$ (red), 0.75 (blue) and 1.25 (yellow)	68
4.5	Sample orbit points of $\eta(\sigma + it)$: $\sigma = 0.5$ (red), 0.75 (blue) and 1.25 (yellow)	68
4.6	Sample orbit points of $\eta(\sigma + it)$: $\sigma = 0.5$ (red), 0.75 (blue) and 1.25 (yellow)	69
4.7	Raw orbit points of $\eta(\sigma + it)$: $\sigma = 0.5$ (red), 0.75 (blue) and 1.25 (yellow)	69
4.8	Convergence of partial sums of $\eta(z)$, for six $z = \sigma + it$ in the complex plane	69
5.1	Special interlacing of 4 lattice processes with $s = 0$	72
5.2	Classification of left dataset; $s = 0.15, w = 10$. One loop (middle) vs 3 (right).	73
5.3	Clustering of left dataset; $s = 0.15$, 3 loops, $w = 10$ (middle) vs 20 (right).	74
5.4	Classification ($w = 10$) and clustering ($w = 20$); $s = 0.05$, three loops.	74
5.5	Fractal classification, $s = 0.15$. Loop 6, 250 and 400.	75
5.6	Fractal classification, $s = 0.05$.Loop: 6 and 60.	75
5.7	Fast (left) vs standard method (right), 3 loops, $s = 0.15, w = 10$	76
5.8	Fast method, $s = 0.05, w = 20$. Three loops (middle), one loop (right).	76
6.1	Comparing two shapes	85
6.2	Weighted centroid, shape signature	86
6.3	Weight function used in Figure 6.2	87
6.4	Another interesting shape	88
7.1	Regression coefficients oscillating when using adaptive damping	95
7.2	Convergence of regression coefficients (left) and distribution of residual error (right)	96
7.3	Goodness-of-fit: training set (right) versus validation set (left)	96
8.1	Fuzzy regression with prediction intervals, original version, 1D	103
8.2	Fuzzy regression with prediction intervals, full model, 2D	105
8.3	Scatterplots: median vs weighted method, on validation (left) vs training set (right)	107
8.4	Dirichlet eta function (real part, bottom) and interpolation error (top)	109
9.1	Interpolating the real part of $\zeta(\frac{1}{2} + it)$ based on orange points	114
9.2	Tides at Dublin (5-min data), with 80 mins between interpolating nodes	117
9.3	Temperature data: interpolation with my method (observed values at dots)	118
9.4	My method: round dots represent observed values, “+” are interpolated	118
9.5	Temperature dataset: interpolation using ordinary kriging	119
10.1	Synthetic versus real data, produced by SDV GAN + copula	135

10.2	Loss function (in orange) for 10^4 successive epochs; enhanced GAN on the right plot	137
10.3	Summary statistics, medical dataset (synth 1 and 2 correspond to GAN)	138
10.4	Correlation matrix, real vs synthetic: GAN (synth 2) and copula-based	139
10.5	Copulas superior to GANs (synth 1, 2) to capture correlations in real data	139
10.6	Real data (left), copula (middle) and GAN synthetization (right)	141
10.7	Loss function (orange) and distance (grey), circle dataset	142
10.8	Data synthetization: general schema	144
11.1	Orbit of $L(z, \chi)$ at $\sigma = \frac{1}{2}$, with $0 < t < 200$ and $\chi = \chi_4$ (left) versus pseudo-random χ (right) . .	152
11.2	$L_3^*(n)$ test statistic for four sequences: Python[200] and SQRT[90,91] fail	154
11.3	$ L_3(n) $ test statistic for four sequences: Python[200] and SQRT[90,91] fail	154
11.4	Correlations are computed on sequences consisting of 300 binary digits	165
12.1	Typical path S_n with $0 \leq n \leq 50,000$ for four types of random walks	168
12.2	$\delta_n = 1 - \text{Var}[S_{n+1}] + \text{Var}[S_n]$ for four types of random walks, with $0 \leq n \leq 5000$	169
12.3	Same as Figure 12.2, using a more aesthetic but less meaningful chart type	170
12.4	Clustered Brownian process	172
12.5	AR models, classified based on the types of roots of the characteristic polynomial	173
13.1	Function $f(b)$ as a better alternative to $g(b)$ in Figure 13.2. Root at $b = 3083$	177
13.2	Function $g(b) = 2 - \cos(2\pi b) - \cos(2\pi a/b)$, with $a = 3083 \times 7919$	177
13.3	Transformed function f_3 , amplifying the root at $b = 3083$	178
13.4	Signal strength ρ_n , first 130 fixed-point iterations; $n = 31$ leads to a root.	181
13.5	(b_n, ρ_n) plot. Yellow and orange dots linked to roots.	181
13.6	Signal strength ρ_n , first 130 fixed-point iterations; $n = 87$ leads to a root.	181
13.7	Random function from section 13.3.1, with root at $b = 5646$	184
14.1	Six frames from the terrain video, each one containing four images	189
14.2	Contour plot, 3D mixture model, produced with Plotly	197
14.3	Same as Figure 14.2, produced with Matplotlib	198
15.1	Collisions graph for the biggest star eater (star 47) in video 7	205
15.2	Summary statistics for the whole collision structure: the X axis represents the time	206
15.3	Snapshots of universe 4 (left) and universe 7 (right)	207
16.1	Period and amplitude of $\phi_\tau(t)$; here $\tau = 1, \lambda = 1.4, s = 0.3$	215
16.2	A new test of independence (R-squared version)	215
16.3	Radial cluster process ($s = 0.2, \lambda = 1$) with centers in blue; zoom in on the left	218
16.4	Radial cluster process ($s = 2, \lambda = 1$) with centers in blue; zoom in on the left	218
16.5	Manufactured marble lacking true lattice randomness (left)	218
16.6	Four superimposed Poisson-binomial processes: $s = 0$ (left), $s = 5$ (right)	221
16.7	Rayleigh test to assess if a point distribution matches that of a Poisson process	229
16.8	Realization of a 5-interlacing with $s = 0.15$ and $\lambda = 1$: original (left), modulo $2/\lambda$ (right)	232
16.9	Locally random permutation σ ; $\tau(k)$ is the index of X_k 's closest neighbor to the right	232
16.10	Each arrow links a point (blue) to its vertex (red): $s = 0.2$ (left), $s = 1$ (right)	235
16.11	Distance between a point and its vertex ($\lambda = s = 1$)	236
17.1	Three orbits ($\sigma = 0.5, 0.75, 1.25$) with finite Euler product: $P = \{2, 3\}$ (left) vs $\{2, 3, 5\}$ (right) .	240
17.2	Distance between orbit and location $(c, 0)$ depending on t on the X-axis	242
17.3	Distance between orbit and location $(c, 0)$ depending on t on the X-axis	242
17.4	Distance between orbit and location $(c, 0)$ depending on t on the X-axis	242
17.5	Four orbits where the “hole” (repulsion basin) is apparent	244
17.6	Three orbits with “hole” closer to the origin, showing impact of $\beta > \frac{1}{2}$ and larger n	244
17.7	Orbit of Dirichlet eta $\eta(z)$ when cosines are replaced by other periodic functions	248
18.1	Data linked to the melody: red curve for note frequencies, blue curve for note durations	260
18.2	R plot before Cairo (left), and after (right)	261
18.3	Intermediate (left) and last frame (right) of the video	262
18.4	Example of 90% dual confidence region for (p, q)	264
18.5	Minimum contrast estimation for (λ, s) using (p, q) as proxy stats	265
18.6	Non-elliptic confidence regions with various confidence levels	266
18.7	Elbow rule (right) finds $m = 3$ clusters in Brownian motion (left)	274

List of Tables

1.1	Estimated ellipse parameters vs true values ($n = 30$), for shape in Figure 1.2	20
1.2	First and last step of <code>curve_fitting</code> , approaching the model.	28
1.3	MSE for different methods and θ s, same data set as in Figure 1.5	33
1.4	MSE for different methods and θ s, same data set as in Figure 1.7	33
2.1	List of potential features to use in the model	38
2.2	Statistics for selected HDT nodes (Excel version)	41
2.3	Order of magnitude for the expectation and standard deviation of the range R_n	45
3.1	Characteristic polynomials used in the simulations	52
7.1	Regression coefficients and performance metrics r, s based on methodology	97
7.2	Correlation matrix	97
7.3	Best performance given m (number of features)	98
7.4	Feature comparison table (top 32 feature combinations)	100
7.5	Feature comparison table (bottom 31 feature combinations)	101
8.1	R-squared ρ^2 and slope β , on training and validation sets, median vs weighted	107
10.1	Comparing real data with two different synthetic copies	132
10.2	Correlations on 9D circle dataset: real vs copula and GAN	142
11.1	$L_3^*(n)$, for various sequences ($n = 20,000$); “Fail” means failing the prime test	155
13.1	High ρ_n at iterations $n = 31$ and $n = 127$ points to roots 3083 and 7919	180
15.1	Description of top parameters used in the star cluster simulator	203
15.2	Eight selected parameter sets covering various situations	204
16.1	Variance attached to F_s , as a function of s	214
16.2	Poisson process ($s = \infty$) versus $s = 39.85$	234
18.1	Extract of the mapping table used to recover (λ, s) from (p, q)	266
18.2	Eight bins: 2 features (A, B) times 2 outcomes (Good/Bad)	268
18.3	Amount of data collected at each level, when crawling the Internet	272

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 50, 98, 172
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 38, 74, 268
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 37, 277
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 16, 97, 229, 277
Confidence Region	A confidence region of level γ is a 2D set of minimum area covering a proportion γ 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 13, 16, 19, 21, 30, 226, 227, 263, 266
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 16, 38, 94, 100, 138, 204, 268, 277
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 37, 38, 40, 42, 277, 278
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 13, 17, 278, 279
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 17, 97, 121, 130, 150, 213, 216, 222, 228, 233, 235, 247
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 37, 84, 277

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 14, 36, 70, 75, 84, 91, 129, 141, 176, 192, 230
Feature selection	Features – as opposed to the model response – are also called independent variables or predictors. Feature selection, akin to dimensionality reduction , aims at finding the minimum subset of variables with enough predictive power . It is also used to eliminate redundant features and find causality (typically using hierarchical Bayesian models), as opposed to mere correlations. Sometimes, two features have poor predictive power when taken separately, but provide improved predictions when combined together. See pages 13, 16, 38, 95, 98, 259, 267, 277, 279
Generative model	Bayesian Gaussian mixtures (GMM) combined with kernel density estimation and the EM algorithm is a classic modeling tool. In this book, I used <i>m</i>-interlacings instead. Generative adversarial networks (GAN) 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. New observations can also be generated via parametric bootstrap . See pages 36, 53, 100, 143, 187, 188, 190, 197, 204, 279
Goodness-of-fit	A model fitting criterion or metric to assess how a model or sub-model fits to a dataset, or to measure its predictive power on a validation set . Examples include R-squared , Chi-squared, Kolmogorov-Smirnov, error rate such as false positives and other metrics discussed in this book. See pages 16, 57, 94, 95, 268, 277, 279
Gradient methods	Iterative optimization techniques to find the minimum or maximum of a function, such as the maximum likelihood . When there are numerous local minima or maxima, use swarm optimization . Gradient methods (for instance, stochastic gradient descent or Newton's method) assume that the function is differentiable. If not, other techniques such as Monte Carlo simulations or the fixed-point algorithm can be used. Constrained optimization involves using Lagrange multipliers . See pages 16, 32, 56, 90
Graph structures	Graphs are found in decision trees , in neural networks (connections between neurons), in nearest neighbors methods (NN graphs), in hierarchical Bayesian models , and more. See pages 71, 75, 205, 271, 272
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 neural networks or regression coefficients) are derived via training. See pages 30, 57, 71, 76, 102, 136, 191, 278
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 logit link function is used in logistic regression . Generalizations include quantile functions and inverse sigmoids in neural network to work with additive (linear) parameters. See pages 14, 17, 278
Logistic regression	A generalized linear regression 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 17, 34, 37, 41, 277, 278
Neural network	A blackbox system used for predictions, optimization, or pattern recognition especially in computer vision. It consists of layers, neurons in each layer, link functions to model non-linear interactions, parameters (weights associated to the connections between neurons) and hyperparameters . Networks with several layers are called deep neural networks . Also, neurons are sometimes called nodes. See pages 70, 74, 76, 84, 102, 277, 278

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 18.5 deals with creating a keyword taxonomy based on parsing Google search result pages. Text generation is referred to as NLG or natural language generation , using large language models (LLM). See pages 37, 270
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 ill-conditioned problems . 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 13, 15, 60
Overfitting	Using too many unstable parameters resulting in excellent performance on the training set , but poor performance on future data or on the validation set . 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 wide data (more features than observations) when using a method not suited to this situation. The opposite is underfitting. See pages 16, 93, 102, 130, 136, 277, 279
Predictive power	A metric to assess the goodness-of-fit or performance of a model or subset of features, for instance in the context of dimensionality reduction or feature selection . Typical metrics include R-squared , or confusion matrices in classification. See pages 39, 41, 45, 129, 267, 269, 278
R-squared	A goodness-of-fit metric to assess the predictive power of a model, measured on a validation set . Alternatives include adjusted R-squared, mean absolute error and other metrics discussed in this book. See pages 13, 16, 36, 57, 91, 94, 96, 98, 105, 278, 279
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 30, 149, 156, 273
Regression methods	I discuss a unified approach to all regression problems in chapter 1. Traditional techniques include linear, logistic, Bayesian, polynomial and Lasso regression (to deal with numerical instability and overfitting), solved using optimization techniques, maximum likelihood methods, linear algebra (eigenvalues and singular value decomposition) or stepwise procedures. See pages 13, 14, 16, 17, 20, 28, 37, 41, 47, 51, 53, 57, 90, 96, 102, 109, 278, 279
Supervised learning	Techniques dealing with labeled data (classification) or when the response is known (regression). The opposite is unsupervised learning , for instance clustering problems. In-between, you have semi-supervised learning and reinforcement learning (favoring good decisions). The technique described in chapter 1 fits into unsupervised regression. Adversarial learning is testing your model against extreme cases intended to make it fail, to build better models. See pages 279
Synthetic data	Artificial data simulated using a generative model , typically a mixture model , to enrich existing datasets and improve the quality of training sets . Called augmented data when blended with real data. See pages 13, 14, 16, 18, 28, 30, 34, 36, 49, 53, 56, 70, 71, 76, 89, 95, 106, 113, 119, 130, 149, 162, 168, 176, 190, 197, 204, 264, 273, 277
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 70, 75
Training set	Dataset used to train your model in supervised learning . Typically, a portion of the training set is used to train the model, the other part is used as validation set . See pages 14, 16, 18, 21, 30, 37, 41, 57, 73, 89, 96, 102, 106, 204, 268, 277, 279
Validation set	A portion of your training set , 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 16, 28, 42, 57, 94, 102, 130, 204, 268, 277, 278, 279

Bibliography

- [1] Weighted percentiles using numpy. *Forum discussion*, 2020. StackOverflow [\[Link\]](#). 102
- [2] Jan Ackmann et al. Machine-learned preconditioners for linear solvers in geophysical fluid flows. *Preprint*, pages 1–19, 2020. arXiv:2010.02866 [\[Link\]](#). 94
- [3] Noga Alon and Joel H. Spencer. *The Probabilistic Method*. Wiley, fourth edition, 2016. [\[Link\]](#). 222
- [4] José M. Amigó, Roberto Dale, and Piergiulio Tempesta. A generalized permutation entropy for random processes. *Preprint*, pages 1–9, 2012. arXiv:2003.13728 [\[Link\]](#). 233
- [5] Luc Anselin. *Point Pattern Analysis: Nearest Neighbor Statistics*. The Center for Spatial Data Science, University of Chicago, 2016. Slide presentation [\[Link\]](#). 220
- [6] Insaf Ashrapov. Tabular gans for uneven distribution. *Preprint*, pages 1–11, 2020. arXiv:2010.00638 [\[Link\]](#). 136
- [7] Adrian Baddeley. Spatial point processes and their applications. In Weil W., editor, *Stochastic Geometry. Lecture Notes in Mathematics*, pages 1–75. Springer, Berlin, 2007. [\[Link\]](#). 219
- [8] David Bailey and Richard Crandall. Random generators and normal numbers. *Experimental Mathematics*, 11, 2002. Project Euclid [\[Link\]](#). 166
- [9] N. Balakrishnan and C.R. Rao (Editors). *Order Statistics: Theory and Methods*. North-Holland, 1998. 222, 236
- [10] Christopher Beckham and Christopher Pal. A step towards procedural terrain generation with GANs. *Preprint*, pages 1–5, 2017. arXiv:1707.03383 [\[Link\]](#). 189
- [11] Rabi Bhattacharya and Edward Waymire. *Random Walk, Brownian Motion, and Martingales*. Springer, 2021. 167
- [12] Barbara Bogacka. *Lecture Notes on Time Series*. 2008. Queen Mary University of London [\[Link\]](#). 50
- [13] B. Bollobas and P. Erdős. Cliques in random graphs. *Mathematical Proceedings of the Cambridge Philosophical Society*, 80(3):419–427, 1976. [\[Link\]](#). 223
- [14] Miklos Bona. *Combinatorics of Permutations*. Routledge, second edition, 2012. 233
- [15] Ali Borji. Pros and cons of GAN evaluation measures: New developments. *Preprint*, pages 1–35, 2021. arXiv:2103.09396 [\[Link\]](#). 136
- [16] 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. [\[Link\]](#). 241
- [17] 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. [\[Link\]](#). 243
- [18] Oliver Bröker and Marcus J. Groteb. Sparse approximate inverse smoothers for geometric and algebraic multigrid. *Applied Numerical Mathematics*, 41(1):61–80, 2002. 91
- [19] H. M. Bui and M. B. Milinovich. Gaps between zeros of the Riemann zeta-function. *Quarterly Journal of Mathematics*, 69(2):402–423, 2018. [\[Link\]](#). 253
- [20] Bartłomiej Błaszczyzyn and Dhandapani Yogeshwaran. Clustering and percolation of point processes. *Preprint*, pages 1–20, 2013. Project Euclid [\[Link\]](#). 219
- [21] Bartłomiej Błaszczyzyn and Dhandapani Yogeshwaran. On comparison of clustering properties of point processes. *Preprint*, pages 1–26, 2013. arXiv:1111.6017 [\[Link\]](#). 219
- [22] Bartłomiej Błaszczyzyn and Dhandapani Yogeshwaran. Clustering comparison of point processes with applications to random geometric models. *Preprint*, pages 1–44, 2014. arXiv:1212.5285 [\[Link\]](#). 219
- [23] 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. 274

- [24] Keith Conrad. *L-functions and the Riemann Hypothesis*. 2018. 2018 CTNT Summer School [Link]. 152, 238, 241, 246
- [25] Noel Cressie. *Statistic for Spatial Data*. Wiley, revised edition, 2015. 219
- [26] 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. 171
- [27] 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. 171
- [28] 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. 265
- [29] Marc Deisenroth, A. Faisal, and Cheng Soon Ong. *Mathematics for Machine Learning*. Cambridge University Press, 2020. [Link]. 54
- [30] Harold G. Diamond and Wen-Bin Zhang. *Beurling Generalized Numbers*. American Mathematical Society, 2016. Mathematical Surveys and Monographs, Volume 213 [Link]. 153, 249
- [31] 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. 220
- [32] 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. 220
- [33] David Coupier (Editor). *Stochastic Geometry: Modern Research Frontiers*. Wiley, 2019. 230
- [34] 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. 221
- [35] Khaled Emam, Lucy Mosquera, and Richard Hoptroff. *Practical Synthetic Data Generation*. O'Reilly, 2020. 100
- [36] 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]. 223
- [37] 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]. 189
- [38] Arash Farahmand. *Math 55 Lecture Notes*. 2021. University of Berkeley [Link]. 49, 54
- [39] W. Feller. On the Kolmogorov-Smirnov limit theorems for empirical distributions. *Annals of Mathematical Statistics*, 19(2):177–189, 1948. [Link]. 222, 229
- [40] Alvaro Figueira and Bruno Vaz. Survey on synthetic data generation, evaluation methods and GANs. *New Insights in Machine Learning and Deep Neural Networks*, 2022. MDPI [Link]. 136
- [41] Nikos Frantzikinakis. Ergodicity of the Liouville system implies the Chowla conjecture. *Preprint*, pages 1–41, 2016. arXiv [Link]. 243
- [42] P. M. Gauthier. Approximating the Riemann zeta-function by polynomials with restricted zeros. *Canadian Mathematical Bulletin*, 62(3):475–478, 2018. [Link]. 253
- [43] P. A. Van Der Geest. The binomial distribution with dependent Bernoulli trials. *Journal of Statistical Computation and Simulation*, pages 141–154, 2004. [Link]. 168
- [44] Stamatia Giannarou and Tania Stathaki. Shape signature matching for object identification invariant to image transformations and occlusion. 2007. ResearchGate [Link]. 85
- [45] 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]. 223
- [46] B.V. Gnedenko and A. N. Kolmogorov. *Limit Distributions for Sums of Independent Random Variables*. Addison-Wesley, 1954. 172
- [47] Manuel González-Navarrete and Rodrigo Lambert. Non-markovian random walks with memory lapses. *Preprint*, pages 1–14, 2018. arXiv [Link]. 167
- [48] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. [Link]. 54
- [49] 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]. 220
- [50] Vincent Granville. *Applied Stochastic Processes, Chaos Modeling, and Probabilistic Properties of Numeration Systems*. MLTechniques.com, 2018. [Link]. 153
- [51] Vincent Granville. *Stochastic Processes and Simulations: A Machine Learning Perspective*. MLTechniques.com, 2022. [Link]. 52, 60, 172, 182, 213, 214, 216, 217, 221, 223, 249, 253, 267
- [52] 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. 73

- [53] Vincent Granville and Richard L Smith. Disaggregation of rainfall time series via Gibbs sampling. *NISS Technical Report*, pages 1–21, 1996. [\[Link\]](#). 108
- [54] Kristen Grauman. Shape matching. 2008. University of Texas, Austin [\[Link\]](#). 88
- [55] Hui Guo et al. Eyes tell all: Irregular pupil shapes reveal gan-generated faces. *Preprint*, pages 1–7, 2021. arXiv:2109.00162 [\[Link\]](#). 129
- [56] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*. O'Reilly, third edition, 2023. 36
- [57] Radim Halir and Jan Flusser. Numerically stable direct least squares fitting of ellipses. *Preprint*, pages 1–8, 1998. [\[Link\]](#). 18, 20
- [58] Peter Hall. *Introduction to the theory of coverage processes*. Wiley, 1988. 230
- [59] Adam J. Harper. Moments of random multiplicative functions, II: High moments. *Algebra and Number Theory*, 13(10):2277–2321, 2019. [\[Link\]](#). 149, 249
- [60] 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. [\[Link\]](#). 149, 151, 249
- [61] Adam J. Harper. Almost sure large fluctuations of random multiplicative functions. *Preprint*, pages 1–38, 2021. arXiv [\[Link\]](#). 151, 243, 249
- [62] K. Hartmann, J. Krois, and B. Waske. *Statistics and Geospatial Data Analysis*. Freie Universität Berlin, 2018. E-Learning Project SOGA [\[Link\]](#). 216
- [63] D. R. Heath-Brown. Primes represented by $x^3 + 2y^3$. *Acta Mathematica*, 186:1–84, 2001. [\[Link\]](#). 245
- [64] Markus Herdin. Correlation matrix distance, a meaningful measure for evaluation of non-stationary MIMO channels. *Proc. IEEE 61st Vehicular Technology Conference*, pages 1–5, 2005. [\[Link\]](#). 136
- [65] T. W. Hilberdink and M. L. Lapidus. Beurling Zeta functions, generalised primes, and fractal membranes. *Preprint*, pages 1–31, 2004. arXiv [\[Link\]](#). 152, 153, 249
- [66] Christian Hill. *Learning Scientific Programming with Python*. Cambridge University Press, 2016. [\[Link\]](#). 20
- [67] Robert V. Hogg, Joseph W. McKean, and Allen T. Craig. *Introduction to Mathematical Statistics*. Pearson, eighth edition, 2016. [\[Link\]](#). 54
- [68] 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. [\[Link\]](#). 264
- [69] Peter Humphries. The distribution of weighted sums of the Liouville function and Pólya's conjecture. *Preprint*, pages 1–33, 2011. arXiv [\[Link\]](#). 250
- [70] Timothy D. Johnson. Introduction to spatial point processes. *Preprint*, page 2008. NeuroImaging Statistics Oxford (NISOx) group [\[Link\]](#)[\[Mirror\]](#). 220
- [71] Chigozie Kelechi. Towards efficiency in the residual and parametric bootstrap techniques. *American Journal of Theoretical and Applied Statistics*, 5(5), 2016. [\[Link\]](#). 98
- [72] Denis Kojevnikov, Vadim Marmer, and Kyungchul Song. Limit theorems for network dependent random variables. *Journal of Econometrics*, 222(2):419–427, 2021. [\[Link\]](#). 220
- [73] Samuel Kotz, Tomasz Kozubowski, and Krzysztof Podgorski. *The Laplace Distribution and Generalizations: A Revisit with Applications to Communications, Economics, Engineering, and Finance*. Springer, 2001. 234
- [74] Faraj Lagum. *Stochastic Geometry-Based Tools for Spatial Modeling and Planning of Future Cellular Networks*. PhD thesis, Carleton University, 2018. [\[Link\]](#). 219
- [75] Günther Last and Mathew Penrose. *Lectures on the Poisson Process*. Cambridge University Press, 2017. 219
- [76] 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. [\[Link\]](#). 149, 151
- [77] Gary R. Lawlor. A l'Hospital's rule for multivariable functions. *Preprint*, pages 1–13, 2013. arXiv:1209.0363 [\[Link\]](#). 114
- [78] Jing Lei et al. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113:1094–1111, 2018. [\[Link\]](#). 98
- [79] Hui Liu et al. A new model using multiple feature clustering and neural networks for forecasting hourly PM_{2.5} concentrations. *Engineering*, 6:944–956, 2020. [\[Link\]](#). 134

- [80] Mario Lucic et al. Are GANs created equal? a large-scale study. *Proc. NeurIPS Conference*, pages 1–10, 2018. [\[Link\]](#). 136
- [81] G. Last M.A. Klatt and D. Yogeshwaran. Hyperuniform and rigid stable matchings. *Random Structures and Algorithms*, 2:439–473, 2020. [\[Link\]](#)[\[PowerPoint\]](#). 219
- [82] Jorge Mateu, Frederic P Schoenberg, and David M Diez. On distances between point patterns and their applications. *Preprint*, pages 1–29, 2010. [\[Link\]](#). 220
- [83] 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 [\[Link\]](#). 223
- [84] Masahiro Mine. Probability density functions attached to random Euler products for automorphic L-functions. *Preprint*, pages 1–38, 2020. arXiv [\[Link\]](#). 249, 250
- [85] Christoph Molnar. *Interpretable Machine Learning*. ChristophMolnar.com, 2022. [\[Link\]](#). 98, 129
- [86] Marc-Andreas Muendler. Linear difference equations and autoregressive processes. 2000. University of Berkeley [\[Link\]](#). 50
- [87] V. Kumar Murty. Seminar on Fermat’s last theorem. In *Canadian Mathematical Society – Conference Proceedings*, volume 17, Toronto, Canada, 1995. [\[Link\]](#). 246
- [88] Peter Mörters and Yuval Peres. *Brownian Motion*. Cambridge University Press, 2010. Cambridge Series in Statistical and Probabilistic Mathematics, Volume 30 [\[Link\]](#). 167, 171
- [89] Jesper Møller. Introduction to spatial point processes and simulation-based inference. In *International Center for Pure and Applied Mathematics (Lecture Notes)*, Lomé, Togo, 2018. [\[Link\]](#)[\[Mirror\]](#). 220, 233, 265
- [90] Jesper Møller and Rasmus P. Waagepetersen. *An Introduction to Simulation-Based Inference for Spatial Point Processes*. Springer, 2003. 220
- [91] Jesper Møller and Rasmus P. Waagepetersen. *Statistical Inference and Simulation for Spatial Point Processes*. CRC Press, 2007. 220
- [92] S. Ghosh N., Miyoshi, and T. Shirai. Disordered complex networks: energy optimal lattices and persistent homology. *Preprint*, pages 1–44, 2020. arXiv:2009.08811. 213
- [93] Saralees Nadarajah. A modified Bessel distribution of the second kind. *Statistica*, 67(4):405–413, 2007. [\[Link\]](#). 234
- [94] 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. [\[Link\]](#). 230
- [95] Guillermo Navas-Palencia. Optimal binning: mathematical programming formulation. *Preprint*, pages 1–21, 2020. arXiv:2001.08025 [\[Link\]](#). 38
- [96] Nathan Ng. Large gaps between the zeros of the Riemann zeta function. *Journal of Number Theory*, 128(3):509–556, 2007. [\[Link\]](#). 253
- [97] Sergey I. Nikolenko. *Synthetic Data for Deep Learning*. Springer, 2021. 136
- [98] 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. 219
- [99] Fred Park. Shape descriptor / feature extraction techniques. 2011. UCI iCAMP 2011 [\[Link\]](#). 85
- [100] Yuval Peres and Allan Sly. Rigidity and tolerance for perturbed lattices. *Preprint*, pages 1–20, 2020. arXiv:1409.4490 [\[Link\]](#). 213, 219
- [101] Carl Rasmussen and Christopher Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006. [\[Link\]](#). 53
- [102] Alfred R.Osborne. Multidimensional Fourier series. *International Geophysics*, 97:115–145, 2010. [\[Link\]](#). 121
- [103] Kamron Sanee. A simple expression for multivariate Lagrange interpolation. *SIAM Undergraduate Research Online*, 2007. SIURO [\[Link\]](#). 104
- [104] Mahesh Shivanand and all. Fitting random regression models with Legendre polynomial and B-spline to model the lactation curve for Indian dairy goat of semi-arid tropic. *Journal of Animal Breeding and Genetics*, pages 414–422, 2022. [\[Link\]](#). 121
- [105] Karl Sigman. Notes on the Poisson process. New York NY, 2009. IEOR 6711: Columbia University course [\[Link\]](#). 219
- [106] 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 [\[Link\]](#). 36

- [107] Luuk Spreeuwiers. *Image Filtering with Neural Networks: Applications and Performance Evaluation*. PhD thesis, University of Twente, 1992. [74](#)
- [108] J. Michael Steele. Le Cam's inequality and Poisson approximations. *The American Mathematical Monthly*, 101(1):48–54, 1994. [\[link\]](#). [182](#)
- [109] Dietrich Stoyan, Wilfrid S. Kendall, Sung Nok Chiu, and Joseph Mecke. *Stochastic Geometry and Its Applications*. Wiley, 2013. [230](#)
- [110] E.C. Titchmarsh and D.R. Heath-Brown. *The Theory of the Riemann Zeta-Function*. Oxford Science Publications, second edition, 1987. [59](#), [152](#), [238](#)
- [111] Chris Tofallis. Fitting equations to data with the perfect correlation relationship. *Preprint*, pages 1–11, 2015. Hertfordshire Business School Working Paper[\[Link\]](#). [14](#)
- [112] 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. [\[Link\]](#). [15](#), [18](#)
- [113] 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. [\[Link\]](#). [18](#)
- [114] Remco van der Hofstad. *Random Graphs and Complex Networks*. Cambridge University Press, 2016. [\[Link\]](#). [222](#)
- [115] Yu Vizilter and Sergey Zheltov. Geometrical correlation and matching of 2D image shapes. 2012. ResearchGate [\[Link\]](#). [87](#)
- [116] Fengyun Wang and all. Bivariate Fourier-series-based prediction of surface residual stress fields using stresses of partial points. *Mathematics and Mechanics of Solids*, 2018. [\[Link\]](#). [121](#)
- [117] Luyao Wang and Hai Cheng. Pseudo-random number generator based on logistic chaotic system. *Entropy*, 21, 2019. [\[Link\]](#). [166](#)
- [118] 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. [\[Link\]](#). [189](#)
- [119] Lan Wu, Yongcheng Qi, and Jingping Yang. Asymptotics for dependent Bernoulli random variables. *Statistics and Probability Letters*, pages 455–463, 2012. [\[Link\]](#). [167](#)
- [120] Lei Xu and Kalyan Veeramachaneni. Synthesizing tabular data using generative adversarial networks. *Preprint*, pages 1–12, 2018. arXiv:1811.11264 [\[Link\]](#). [136](#)
- [121] Oren Yakir. Recovering the lattice from its random perturbations. *Preprint*, pages 1–18, 2020. arXiv:2002.01508 [\[Link\]](#). [219](#)
- [122] 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. [233](#)
- [123] Shaohong Yan, Aimin Yang, et al. Explicit algorithm to the inverse of Vandermonde matrix. In *2009 International Conference on Test and Measurement*, 2009. IEEE [\[Link\]](#). [48](#)
- [124] D. Yogeshwaran. Geometry and topology of the boolean model on a stationary point processes : A brief survey. *Preprint*, pages 1–13, 2018. Researchgate [\[Link\]](#). [220](#)
- [125] 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. [215](#)
- [126] Changgang Zheng et al. Reward-reinforced generative adversarial networks for multi-agent systems. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 6:479–488, 2021. arXiv:2103.12192 [\[Link\]](#). [141](#)

Index

- α -compositing, 58
 m -interlacing, 72, 220, 228, 231, 278
- A/B testing, 276
activation function, 137
AdaBoost, 37, 277
Adam gradient descent, 138
additive number theory, 245, 252
adversarial learning, 129, 279
agent-based modeling, 188, 201
AI art, 130, 197
algebraic number, 154
algorithmic bias, 130
analytic continuation, 239, 246
analytic function, 152
anisotropy, 206, 227, 233
anti-aliasing, 56, 60, 256, 261
association rule, 269
attraction (point process), 218
attraction basin, 59
attractor distribution, 172, 222, 228, 235
augmented data, 36, 89, 188, 190, 279
auto-correlation, 50, 153
auto-regressive process, 50, 172, 277
- Bailey–Borwein–Plouffe formulas, 154
Bayesian classification, 75
Bayesian inference, 45
 hierarchical models, 278
 naive Bayes, 269
Bernoulli trials, 263
Berry-Esseen inequality, 150
Bessel function, 234
Beurling primes, 153, 249
binning, 268
 optimum binning, 38, 277
binomial distribution, 228
bisection method (root finding), 179
boosted trees, 131
bootstrapping, 16, 97, 130, 229
 percentile method, 102
boundary effect, 215, 216, 221, 222, 226–228, 232
Brownian motion, 52, 167, 171, 188, 189, 204, 273, 277
 Lévy flight, 172
Brun’s theorem, 245
- Cauchy distribution, 172
Cauchy-Riemann equations, 152
causality, 278
Cayley-Hamilton theorem, 48
CDF regression, 18
- censored data, 221
central limit theorem, 172, 213, 263
chaotic dynamical system, 188, 204
character
 principal, 241
characteristic function, 234
characteristic polynomial, 48, 50–52, 173
ChatGPT, 270
Chebyshev’s bias (prime numbers), 152, 241
checksum, 276
Chi-squared test, 215
Chowla conjecture, 243
classification, 279
clique (graph theory), 223
cluster process, 217, 221, 228
clustering, 279
Collatz conjecture, 160
collision graph, 204
color model
 RGB, 56, 262
 RGBA, 56, 57, 77, 262
color opacity, 198
color transparency, 21, 190, 262
complex random variable, 149, 249
computational complexity, 162, 270
computer vision, 13, 84
confidence band, 229
confidence interval, 45, 226, 277
confidence level, 264, 266
confidence region, 16, 30, 130, 226, 227, 263
 dual region, 45, 264, 277
conformal map, 15
confusion matrix, 268, 279
connected components, 205, 221–223, 227, 228, 271
contour level, 197, 266
contour plot, 266
convergence
 abscissa, 241, 246
 absolute, 113, 238, 239
 alternating series, 239
 conditional, 113, 151, 241
 Dirichlet test, 239
convergence acceleration, 60
convex linear combination, 109
convolution of distributions, 234
copula, 100, 130, 143
 Frank, 130, 136
 Gaussian, 130
correlation matrix distance, 136, 141, 143
cosine distance, 271

counting measure, 214
 covariance matrix, 91, 263
 covering (stochastic), 230
 covering problem, 229
 credible interval, 45
 credible region (Bayesian), 264, 277
 critical line (number theory), 114
 cross-validation, 16, 138, 204, 268
 cuban primes, 245
 curse of dimensionality, 116
 curve fitting, 27

 data video, 188, 197
 decision tree, 37
 decorrelate, 143
 decorrelation, 140, 143
 Dedekind zeta function, 152, 249
 deep neural network, 74, 137, 278
 dense set (topology), 243
 density estimation, 220
 diamond-square algorithm, 189
 Diehard tests of randomness, 151
 dimensionality reduction, 17
 Dirichlet character, 152, 153, 240, 245
 modulo 4, 241, 246, 247
 Dirichlet eta function, 253
 Dirichlet functional equation, 152, 246, 247
 Dirichlet series, 149
 Dirichlet's theorem, 152, 241, 243, 247
 Dirichlet- L function, 152, 240, 247
 disaggregation, 115
 discrete Fourier series, 120
 discrete orthogonal functions, 120
 dissimilarity metric, 271
 distributed architecture, 267
 distribution
 Cauchy, 172
 Fréchet, 52, 172
 Gaussian, 263
 generalized logistic, 91, 217
 Hotelling, 264
 Laplace, 234
 logistic, 17
 Lévy, 172
 modified Bessel, 234
 Poisson-binomial, 182, 236
 Poisson-exponential, 213
 Rademacher, 149, 150
 Rayleigh, 228, 229, 235
 Weibull, 52, 172, 228
 domain of attraction, 222
 dot product, 15
 dummy variable, 37
 dummy variables, 137
 dyadic map, 153
 dynamical systems, 153, 222
 chaotic systems, 188, 204
 dyadic map, 153
 ergodicity, 153
 logistic map, 153
 shift map, 153

 stochastic, 189
 edge effect (statistics), 221
 eigenvalue, 14, 53, 91, 279
 power iteration, 93
 elbow rule, 176, 221, 228, 273
 elliptic curve, 245
 EM algorithm, 36, 134, 278
 empirical distribution, 17, 97, 121, 130, 213, 216, 222,
 228, 233, 235, 247
 multivariate, 150
 empirical quantiles, 102, 143
 ensemble methods, 37, 84, 131
 entropy, 207, 233, 269
 epoch, 137, 140
 equidistribution modulo 1, 155
 equilibrium distribution, 189
 Erdős-Rényi model, 223
 ergodicity, 153, 189, 226, 228, 234
 Euler product, 149, 239, 246
 random, 249
 Euler's transform, 253
 evolutionary process, 189
 experimental design, 276
 experimental math, 57, 237
 explainable AI, 14, 36, 75, 84, 91, 129, 143, 176, 192,
 230
 exploratory analysis, 275
 exponential decay, 41
 exponential sums, 246
 extrapolation, 109
 extreme value theory, 172, 235

 feature attribution, 129
 feature clustering, 134, 140, 143
 feature importance, 129
 feature selection, 16, 98, 267
 Fermat's last theorem, 246
 fixed-point algorithm, 60, 90, 176, 278
 flag vector, 269, 276
 Fourier series, 120
 Fourier transform, 234
 fractal dimension, 52
 fractional part function, 155
 Frobenius norm, 91
 Fruchterman and Rheingold algorithm, 272
 Fréchet distribution, 52, 172
 fuzzy classification, 57

 Gamma function, 52, 172
 GAN (generative adversarial networks), 36, 129, 134,
 143, 192, 213, 278
 Gaussian circle problem, 252
 Gaussian distribution, 263
 Gaussian mixture model
 see GMM, 36
 Gaussian primes, 152, 248
 Gaussian process, 50, 277
 general linear model, 14
 generalized linear model, 14, 49
 generalized logistic distribution, 91, 227

generative adversarial networks
 see GAN, 36
 generative AI, 188, 201
 generative model, 36, 53, 100, 187, 188, 190, 197, 204, 279
 geostatistics, 103
 GIS, 117
 Glivenko-Cantelli theorem, 247
 GMM (Gaussian mixture model), 70, 71, 133, 134, 143, 213, 278
 Goldbach's conjecture, 245
 goodness-of-fit, 57, 268
 GPU-based clustering, 72
 gradient (optimization), 176
 gradient boosting, 277
 gradient operator, 16
 Gram-Schmidt orthogonalization, 120
 graph, 221
 collision graph, 204
 connected components, 205, 227, 271
 directed, 205
 edge, 221
 Fruchterman-Reingold, 205
 nearest neighbor graph, 223, 227
 node, 221, 223
 random graph, 222
 random nearest neighbor graph, 222
 tree, 205
 undirected, 221–223, 228
 vertex, 221
 graph database, 272
 graph theory, 221
 GraphViz, 205
 greedy algorithm, 113, 252
 grid search, 176, 191
 half-tone (music), 259
 Hartman–Wintner theorem, 167
 hash table, 163, 207, 232, 269, 270
 sparse, 270
 Hausdorff distance, 88
 Hellinger distance, 130, 143
 Hermite polynomials, 120
 hexagonal lattice, 221
 hidden decision trees, 37, 38, 277
 hidden layer, 74
 hidden process, 214, 231, 235
 hierarchical clustering, 74, 270
 Hilbert primes, 248
 histogram equalization, 72, 74
 Hoeffding inequality, 170
 homogeneity (point process), 182, 220
 Hotelling distribution, 264
 Hurst exponent, 52
 hyperparameter, 30, 57, 104, 191
 hyperparameters, 136
 identifiability, 231, 233
 ill-conditioned problem, 27, 53, 93, 279
 image segmentation, 74
 imputation (missing values), 130
 index
 index discrepancy, 233
 intensity (stochastic process), 213, 220, 227
 interarrival times, 171, 213, 222, 226, 233
 standardized, 234
 interlaced processes, 220
 Internet of Things, 213
 inverse distance weighting, 105
 inverse square law, 201
 iterated logarithm, 150, 151, 167
 Itô integral, 53
 K-means clustering, 32, 33
 key-value pair, 38, 269
 Kolmogorov-Smirnov test, 130, 150, 215, 222
 kriging, 113
 Kronecker's theorem, 243, 251
 Lagrange interpolation, 53
 Lagrange multiplier, 16, 278
 Laplace distribution, 234
 large language models, 270, 279
 Lasso regression, 16, 279
 latent variables, 137
 lattice, 219
 perturbed lattice, 213
 shifted, 221
 stretched, 221
 law of the iterated logarithm, 150, 151, 167, 243, 249
 Le Cam's theorem, 182, 214
 learning rate, 136, 141, 143
 least absolute residuals, 102
 LightGBM, 136, 143
 link function, 14, 17
 Liouville function, 240, 251
 LLM (large language model), 270, 279
 log-polar map, 15
 logistic distribution, 17, 220
 logistic map, 153
 logistic regression, 17
 unsupervised, 34
 logit function, 278
 loss function, 137, 140, 143
 Lévy distribution, 172
 Lévy flight, 172
 Map-reduce, 267
 marketing attribution, 276
 Markov chain, 50
 MCMC, 149
 Mathematica, 266
 MaxCliqueDyn algorithm, 223
 maximum likelihood estimation, 265, 278, 279
 mean squared error, 16, 31
 medoid, 32
 Mersenne twister, 30, 153, 156, 169
 Mertens function, 240
 minimum contrast estimation, 191, 230, 233, 265
 mixture model, 30, 46, 189, 197, 220, 221, 228, 266, 279
 blending, 189

model fitting, 57, 278
 model identifiability, 16
 modulus (complex number), 173, 239
 Monte Carlo simulations, 149, 278
 morphing (computer vision), 188
 moving average, 178
 multidimensional Fourier series, 121
 multiple root, 114
 multiplicative function
 completely multiplicative, 149, 151, 240, 241, 252
 Rademacher, 149
 Möbius function, 240
 N-body problem, 201
 n-gram (NLP), 270
 naive Bayes, 269, 277
 natural language generation, 270, 279
 natural language processing, 37, 270
 nearest neighbor interpolation, 102, 105
 nearest neighbors, 213, 223, 229, 278
 nearest neighbor distances, 227–229, 231, 235
 nearest neighbor graph, 227
 NetworkX, 205
 neural network, 74
 activation function, 137
 epoch, 137
 hidden layer, 74
 hyperparameter, 76
 neuron, 74, 137, 278
 seq2seq, 187
 sparse, 70
 very deep, 74
 Newton’s method, 176
 NLG (natural language generation), 270, 279
 node (decision tree), 38, 131, 277
 perfect node, 45
 usable node, 39
 node (interpolation), 114
 normal number, 150, 243, 247
 strongly normal, 151
 numerical stability, 48
 Omega function, 240, 246
 order statistics, 235
 ordinary least squares, 51, 102, 120
 orthogonal function, 120
 Otsuka–Ochiai coefficient, 271
 outliers, 235, 273
 overfitting, 16, 130, 136, 233, 277
 palette, 188, 262
 parametric bootstrap, 21, 30, 36, 98, 130, 229, 277, 278
 partial derivative, 114
 partial least squares, 14
 path (graph theory), 221
 percentile bootstrap, 102
 permutation
 entropy, 233
 random permutation, 232
 perturbed lattices, 213
 Plotly, 197
 point count distribution, 214, 227, 230
 point process
 attractive, 228
 cluster process
 Matérn, 219
 Neyman–Scott, 219
 non-homogeneous, 182, 220
 perturbed lattice process, 219
 radial, 220
 renewal process, 219
 repulsive, 218
 Poisson point process, 171, 182, 213, 227
 Poisson-binomial distribution, 182, 213, 236
 Poisson-exponential distribution, 213
 positive semidefinite (matrix), 49, 92
 power iteration, 93
 preconditioning, 93
 prediction interval, 16, 97, 102
 predictive power, 38, 45, 129, 268, 269
 prime test (of randomness), 151, 162, 168
 principal component analysis, 49, 129, 277
 probability generating function, 168
 proxy space, 266
 pseudo-inverse matrix, 49
 pseudo-random numbers, 169, 273
 combined generators, 166
 congruential generator, 156
 Diehard tests, 151, 163
 Mersenne twister, 156, 169, 190
 prime test, 151, 168
 strongly random, 151, 154
 TestU01, 151
 Pólya conjecture, 242
 quadratic irrational, 153, 156, 162
 quantile, 264, 278
 empirical, 102, 130
 weighted, 102
 quantile function, 100, 121, 213, 217, 228
 quantile regression, 16
 R-squared, 16, 36, 191
 Rademacher distribution, 150
 Rademacher function, 149, 243, 249
 random, 151
 random function, 181
 random graph, 222, 223
 random multiplicative function, 149
 Rademacher, 151
 random permutation, 232
 random variable
 complex, 149
 random walk, 167, 191, 277
 first hitting time, 168, 171
 zero crossing, 167
 Rayleigh distribution, 228, 229, 235
 Rayleigh test, 228
 records, 235
 regression splines, 14
 regular expression, 270, 276
 reinforcement learning, 141, 143, 279

rejection sampling, 131
 ReLU function, 137
 renewal process, 219
 repulsion (point process), 218, 229
 repulsion basin, 239
 resampling, 97, 229
 Riemann Hypothesis, 108, 114
 Generalized, 151, 241, 245, 247
 Riemann zeta function, 114, 149, 152, 247, 249
 root mean squared error, 57

 scaling factor, 227, 235
 SDV (Python library), 135
 seed (random number generator), 131, 136, 140, 163,
 190
 semi-supervised learning, 279
 shape signature, 85
 Shapley value, 129
 Shepard's method, 105
 shift map, 153
 sigmoid function, 137, 278
 simplex, 250
 singular value decomposition, 14, 279
 singularity, 207
 six degrees of separation, 272
 Sklar's theorem, 130
 smoothing parameter, 104
 spatial statistics, 103, 219
 spectral domain, 189
 spline regression, 121
 square root (matrix), 49, 92, 140
 square-free integer, 150, 163, 243
 stable distribution, 172, 190, 234
 state space, 189
 stationary distribution, 53
 stationary process, 50, 172, 189, 204, 215, 220, 227
 stepwise regression, 99
 stochastic convergence, 189
 stochastic function, 52
 stochastic geometry, 230
 stochastic gradient descent, 137
 stochastic process, 213
 stochastic residues, 231
 stop word (NLP), 270
 stretching (point process), 221
 Sturm-Liouville theory, 120
 superimposition (point processes), 220
 supervised classification, 72
 surface plot, 197
 SVD (Python library), 143
 swarm optimization, 28, 278
 synthetic data, 14, 28, 30, 53, 89, 91, 113, 119, 130,
 149, 162, 168, 176, 190, 197, 204, 237, 264
 synthetic metric, 269

 TabGAN (Python library), 136
 Tarjan's algorithm, 271
 tensor, 75
 TensorFlow, 136
 text normalization, 270
 Theil-Sen estimator, 102

 time series, 51
 auto-regressive, 52, 172
 disaggregation, 108
 Hurst exponent, 52
 non-periodic, 26
 total least squares, 14
 training set, 102, 204, 268
 transcendental number, 154
 transformer, 74, 187
 tree (graph theory), 205
 twin primes, 245

 universality property, 240, 243, 245
 unsupervised clustering, 72
 unsupervised learning, 34, 279

 validation set, 16, 57, 102, 130, 204, 268
 Vandermonde matrix, 48, 53
 vertex, 213, 221, 222, 235
 video compression
 FFmpeg, 56, 60

 Waring's problem, 245
 Watts and Strogatz model, 272
 Weibull distribution, 52, 172, 228, 235
 weighted least squares, 14
 weighted quantiles, 102
 weighted regression, 17
 white noise, 28, 50, 172, 277
 wide data, 121, 279

 XOR operator, 156