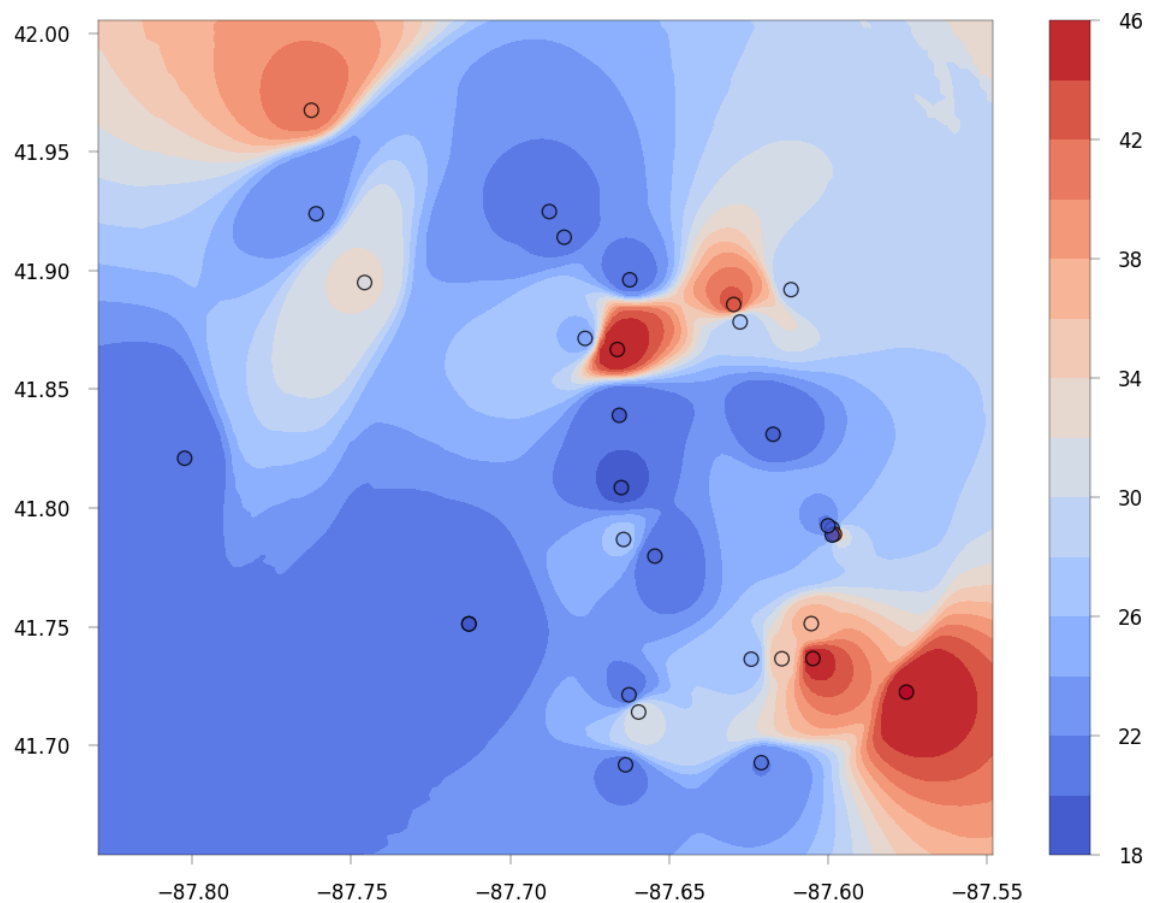

Statistical Optimization for AI and Machine Learning



Preface

This book covers optimization techniques pertaining to machine learning and generative AI, with an emphasis on producing better synthetic data with faster methods, some not even involving neural networks. NoGAN for tabular data is described in detail, along with full Python code, and case studies in healthcare, insurance, cybersecurity, education, and telecom. This low-cost technique is a game changer: it runs 1000x faster than generative adversarial networks (GAN) while consistently producing better results. Also, it leads to replicable results and auto-tuning.

Many evaluation metrics fail to detect defects in synthesized data, not because they are bad, but because they are poorly implemented: due to the complexity, the full multivariate version is absent from vendor solutions. In this book, I describe an implementation of the full version, tested on numerous examples. Known as the multivariate Kolmogorov-Smirnov distance (KS), it is based on the joint empirical distributions attached to the datasets, and works in any dimension on categorical and numerical features. Python libraries, both for NoGAN and KS, are now available and presented in this book.

A very different synthesizer also discussed, namely NoGAN2, is based on resampling, model-free hierarchical methods, auto-tuning, and explainable AI. It minimizes a particular loss function, also without gradient descent. While not based on neural networks, it nevertheless shares many similarities with GAN. Thus you can use it as a sandbox to quickly test various features and hyperparameters before adding the ones that work best, to GAN. Even though NoGAN and NoGAN2 don't use traditional optimization, gradient descent is the topic of the first chapter. Applied to data rather than math functions, there is no assumption of differentiability, no learning parameter, and essentially no math. The second chapter introduces a generic class of regression methods covering all existing ones and more, whether your data has a response or not, for supervised or unsupervised learning. I use gradient descent in this case.

One chapter is devoted to NLP, featuring an efficient technique to process large amounts of text data: hidden decision trees, presenting some similarities with XGBoost. A similar technique is used in NoGAN. Then I discuss other GenAI methods and various optimization techniques, including feature clustering, data thinning, smart grid search and more. Multivariate interpolation is used for time series and geospatial data, while agent-based modeling applies to complex systems.

Methods are accompanied by enterprise-grade Python code, also available on GitHub. Chapters are mostly independent from each other, allowing you to read in random order. The style is very compact, and suitable to business professionals with little 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.

About the author

Vincent Granville is a pioneering GenAI scientist and machine learning expert, co-founder of Data Science Central (acquired by a publicly traded company in 2020), Chief AI Scientist at MLTechniques.com, former VC-funded executive, author and patent owner – one related to LLM. 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 the author of multiple books, available [here](#), including “Synthetic Data and Generative AI” (Elsevier, 2024). Vincent lives in Washington state, and enjoys doing research on stochastic processes, dynamical systems, experimental math and probabilistic number theory. He recently launched a GenAI certification program, offering state-of-the-art, enterprise grade projects to participants. The program, based on his books, is discussed [here](#).

Contents

1	Math-free, Parameter-free Gradient Descent Algorithm	7
1.1	Introduction	7
1.2	Gradient descent and related optimization techniques	8
1.2.1	Implementation details	8
1.2.2	General comments about the methodology and parameters	11
1.2.3	Mathematical version of gradient descent and orthogonal trajectories	12
1.3	Distribution of minima and the Riemann Hypothesis	13
1.3.1	Root taxonomy	14
1.3.2	Studying root propagation with synthetic math functions	14
1.4	Python code	15
1.4.1	Contours and orthogonal trajectories	15
1.4.2	Animated gradient descent starting with 100 random points	19
2	Machine Learning Cloud Regression and Optimization	21
2.1	Introduction: circle fitting	21
2.1.1	Previous versions of my method	22
2.2	Methodology, implementation details and caveats	23
2.2.1	Solution, R-squared and backward compatibility	23
2.2.2	Upgrades to the model	24
2.3	Case studies	25
2.3.1	Logistic regression, two ways	25
2.3.2	Ellipsoid and hyperplane fitting	26
2.3.2.1	Curve fitting: 250 examples in one video	26
2.3.2.2	Confidence region for the fitted ellipse: application to meteorite shapes	27
2.3.2.3	Python code	28
2.3.3	Non-periodic sum of periodic time series: ocean tides	34
2.3.3.1	Numerical instability and how to fix it	35
2.3.3.2	Python code	36
2.3.4	Fitting a line in 3D, unsupervised clustering, and other generalizations	37
2.3.4.1	Example: confidence region for the cluster centers	38
2.3.4.2	Exact solution and caveats	39
2.3.4.3	Comparison with K-means clustering	40
2.3.4.4	Python code	42
2.4	Connection to synthetic data: meteorites, ocean tides	44
3	A Simple, Robust and Efficient Ensemble Method	45
3.1	Introduction	45
3.2	Methodology	46
3.2.1	How hidden decision trees (HDT) work	46
3.2.2	NLP case study: summary and findings	47
3.2.3	Parameters	48
3.2.4	Improving the methodology	48
3.3	Implementation details	48
3.3.1	Correcting for bias	48
3.3.1.1	Time-adjusted scores	49
3.3.2	Excel spreadsheet	49
3.3.3	Python code and dataset	49
3.4	Model-free confidence intervals and perfect nodes	53
3.4.1	Interesting asymptotic properties of confidence intervals	53

4	New Interpolation Methods for Synthetization and Prediction	55
4.1	First method	55
4.1.1	Example with infinite summation	56
4.1.2	Applications: ocean tides, planet alignment	57
4.1.3	Problem in two dimensions	58
4.1.4	Spatial interpolation of the temperature dataset	59
4.2	Second method	61
4.2.1	From unstable polynomial to robust orthogonal regression	62
4.2.2	Using orthogonal functions	62
4.2.3	Application to regression	62
4.3	Python code	63
4.3.1	Time series interpolation	63
4.3.2	Geospatial temperature dataset	66
4.3.3	Regression with Fourier series	69
5	Synthetic Tabular Data: Copulas vs enhanced GANs	71
5.1	Sensitivity analysis, bias reduction and other uses of synthetic data	72
5.2	Using copulas to generate synthetic data	72
5.2.1	The insurance dataset: Python code and results	73
5.2.2	Potential improvements	75
5.3	Synthetization: GAN versus copulas	76
5.3.1	Parameterizing the copula quantiles combined with gradient descent	76
5.3.2	Feature clustering to break a big problem into smaller ones	76
5.4	Deep dive into generative adversarial networks (GAN)	77
5.4.1	Open source libraries and references	77
5.4.2	Synthesizing medical data with GAN	78
5.4.2.1	Hyperparameters	79
5.4.2.2	GAN: Main steps	80
5.4.3	Initial results	81
5.4.4	Fine-tuning the hyperparameters	82
5.4.5	Enhanced GAN: methodology and results	82
5.4.6	Feature clustering via hierarchical clustering or connected components	83
5.5	Comparing GANs with the copula method	86
5.5.1	Conclusion: getting the best out of copulas and GAN	88
5.6	Data synthetization explained in one picture	88
5.7	Python code: GAN to synthesize medical data	89
5.7.1	Classification problem with random forests	90
5.7.2	GAN method	91
5.7.3	GAN Evaluation and post-classification	93
6	Cost-effective Generative AI with NoGAN	94
6.1	Introduction	94
6.2	Description and architecture	95
6.2.1	Binning the feature space	95
6.2.2	Data synthetization	96
6.2.3	Computing the multivariate ECDF	96
6.2.4	Evaluating the quality	97
6.3	Results and discussion	98
6.3.1	Telecom dataset	98
6.3.2	The circle dataset	99
6.4	Potential improvements	100
6.4.1	The original idea behind NoGAN	101
6.4.2	Auto-tuning the hyperparameters, missing values	101
6.5	Conclusion	101
6.6	Python implementation	102
7	Fast Model-free Synthesizer with Hierarchical Bayesian Method	110
7.1	Methodology	110
7.1.1	Base algorithm	111
7.1.2	Loss function	111
7.1.3	Hyperparemeters and convergence	112
7.1.4	Acknowledgments	113

7.2	Case studies	113
7.2.1	Synthesizing the student dataset	114
7.2.2	Synthesizing the Telecom dataset	116
7.2.3	Other case studies	117
7.2.4	Auto-tuning the hyperparameters	118
7.2.5	Evaluation with multivariate ECDF and KS distance	119
7.3	Conclusion	120
7.4	Python implementation	121
8	Three Simple Optimization Techniques	131
8.1	Feature clustering	131
8.1.1	Method and case study	132
8.2	Speed-up AI Training with Stochastic Thinning	133
8.2.1	Introduction	134
8.2.2	The abalone dataset	135
8.2.3	Stochastic thinning with fractional training sets	135
8.2.4	Linear regression	136
8.2.4.1	Conclusions	136
8.2.4.2	Potential improvement	137
8.2.5	Time series	137
8.2.6	Neural networks	138
8.2.6.1	Conclusions	139
8.2.6.2	Jump-starting neural networks with regression	140
8.2.7	Python code	140
8.2.7.1	Stochastic thinning for regression	140
8.2.7.2	Stochastic thinning for neural networks	143
8.3	Smart Grid Search for Faster Hyperparameter Tuning	145
8.3.1	Introduction to hybrid distributions	145
8.3.2	Case study: parametric copulas to synthesize data	147
8.3.2.1	Brief overview of the copula method	147
8.3.2.2	Sampling from a zeta-geometric distribution	148
8.3.3	Smart grid search: viable alternative to gradient descent	148
8.3.4	Python code	150
9	How to Fix a Failing Generative Adversarial Network	154
9.1	Context	154
9.2	GAN enhancements techniques	155
9.2.1	Linear transform to decorrelate features	155
9.2.2	WGAN with PCA transform and standardization	156
9.3	Front-end enhancements to GAN technology	156
9.4	Evaluating synthetizations	158
9.4.1	Loss function history log	158
9.4.2	Histograms: real versus synthetic data	158
9.4.3	Summary statistics: real versus synthetic data	158
9.5	Python code for home-made GAN	159
10	Miscellaneous Topics	164
10.1	Agent-based Modeling: Simulating Aggregative Processes	164
10.1.1	Introduction	164
10.1.2	Atom size distribution	165
10.1.2.1	Core algorithm: single simulation path	165
10.1.2.2	Time scale	166
10.1.2.3	Averaging across multiple simulations	167
10.1.2.4	Collision graphs	167
10.1.3	Python implementation	168
10.2	Fraud Detection and Cybersecurity: Balancing Datasets	171
10.2.1	Project description	172
10.2.2	Solution	173
10.2.3	Python code with SQL queries	173
10.3	Advances in Applied Statistical Engineering	174
10.3.1	Traditional statistics versus new machine learning approach	174
10.3.2	How to build small samples that outperform big data	176

10.3.2.1	Extrapolating confidence intervals beyond the observed data	176
10.3.2.2	Beating the laws of randomness to reduce costs by factor 10	177
10.3.3	The power of smart resampling: case study	177
10.3.4	Best Practices	178
10.3.5	Python code	178
10.4	Generating data outside the observation range	181
10.4.1	Quantile convolution	181
10.4.2	Truncated Gaussian mixtures and bias detection	182
10.4.3	Case studies	183
10.4.4	Conclusion	184
10.4.5	Python code	185
A	Open Source Python Libraries	188
B	Glossary: GAN and Tabular Data Synthetization	192
	Bibliography	196
	Index	198

Bibliography

- [1] Insaf Ashrapov. Tabular gans for uneven distribution. *Preprint*, pages 1–11, 2020. arXiv:2010.00638 [\[Link\]](#). 78
- [2] Caglar Aytakin. Neural networks are decision trees. *Preprint*, pages 1–8, 2022. arXiv:2210.05189 [\[Link\]](#). 94
- [3] Fabiola Banfi, Greta Cazzaniga, and Carlo De Michele. Nonparametric extrapolation of extreme quantiles: a comparison study. *Stochastic Environmental Research and Risk Assessment*, 36:1579–1596, 2022. [\[Link\]](#). 114, 182
- [4] Paul Beale. *Statistical Mechanics*. Academic Press, third edition, 2011. 165
- [5] Marc G. Bellemare et al. The Cramer distance as a solution to biased Wasserstein gradients. *Preprint*, pages 1–20, 2017. arXiv:1705.10743 [\[Link\]](#). 118
- [6] Anthony J Bishara, Jiexiang Li, and Thomas Nash. Asymptotic confidence intervals for the Pearson correlation via skewness and kurtosis. *British Journal of Mathematical and Statistical Psychology*, pages 165–185, 2018. [\[Link\]](#). 175
- [7] Ali Borji. Pros and cons of GAN evaluation measures: New developments. *Preprint*, pages 1–35, 2021. arXiv:2103.09396 [\[Link\]](#). 78
- [8] Wei Chen and Mark Fuge. Synthesizing designs with interpart dependencies using hierarchical generative adversarial networks. *Journal of Mechanical Design*, 141:1–11, 2019. [\[Link\]](#). 113
- [9] Fida Dankar et al. A multi-dimensional evaluation of synthetic data generators. *IEEE Access*, pages 11147–11158, 2022. [\[Link\]](#). 156
- [10] 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 [\[Link\]](#). 78
- [11] Vincent Granville. *Stochastic Processes and Simulations: A Machine Learning Perspective*. MLTechniques.com, 2022. [\[Link\]](#). 140
- [12] Vincent Granville. Generative AI: Synthetic data vendor comparison and benchmarking best practices. *Preprint*, pages 1–13, 2023. MLTechniques.com [\[Link\]](#). 96, 99, 110, 118, 156
- [13] Vincent Granville. *Gentle Introduction To Chaotic Dynamical Systems*. MLTechniques.com, 2023. [\[Link\]](#). 177
- [14] Vincent Granville. *Synthetic Data and Generative AI*. Elsevier, 2024. [\[Link\]](#). 8, 14, 35, 38, 39, 42, 53, 58, 63, 71, 73, 75, 83, 156, 167, 176
- [15] 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. 120, 139
- [16] Hui Guo et al. Eyes tell all: Irregular pupil shapes reveal gan-generated faces. *Preprint*, pages 1–7, 2021. arXiv:2109.00162 [\[Link\]](#). 71
- [17] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*. O’Reilly, third edition, 2023. 44
- [18] Radim Halir and Jan Flusser. Numerically stable direct least squares fitting of ellipses. *Preprint*, pages 1–8, 1998. [\[Link\]](#). 26, 28
- [19] 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. [\[Link\]](#). 78
- [20] Christian Hill. *Learning Scientific Programming with Python*. Cambridge University Press, 2016. [\[Link\]](#). 28
- [21] Pavel Krapivsky, Sidney Redner, and Eli Ben-Naim. *A Kinetic View of Statistical Physics*. Cambridge University Press, 2010. [\[Link\]](#). 165

- [22] Jogendra Nath Kundu et al. GAN-Tree: An incrementally learned hierarchical generative framework for multi-modal data distributions. *IEEE/CVF International Conference on Computer Vision*, pages 8190–8199, 2019. arXiv:1908.03919 [Link]. 113
- [23] Nicolas Langrené and Xavier Warin. Fast multivariate empirical cumulative distribution function with connection to kernel density estimation. *Computational Statistics & Data Analysis*, 162:1–16, 2021. [Link]. 95, 96, 119
- [24] Gary R. Lawlor. A l’Hospital’s rule for multivariable functions. *Preprint*, pages 1–13, 2013. arXiv:1209.0363 [Link]. 56
- [25] Tengyuan Liang. Estimating certain integral probability metric (IPM) is as hard as estimating under the IPM. *Preprint*, pages 1–15, 2019. arXiv:1911.00730 [Link]. 119
- [26] 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]. 76, 83, 132
- [27] Mario Lucic et al. Are GANs created equal? a large-scale study. *Proc. NeurIPS Conference*, pages 1–10, 2018. [Link]. 78
- [28] Christoph Molnar. *Interpretable Machine Learning*. ChristophMolnar.com, 2022. [Link]. 71
- [29] Michael Naaman. On the tight constant in the multivariate Dvoretzky–Kiefer–Wolfowitz inequality. *Statistics & Probability Letters*, 173:1–8, 2021. [Link]. 95, 119
- [30] Guillermo Navas-Palencia. Optimal binning: mathematical programming formulation. *Preprint*, pages 1–21, 2020. arXiv:2001.08025 [Link]. 46
- [31] Sergey I. Nikolenko. *Synthetic Data for Deep Learning*. Springer, 2021. 78
- [32] Peter Olver. *Complex Analysis and Conformal Mapping*. Preprint, 2022. University of Minnesota [Link][Mirror]. 13
- [33] Alfred R. Osborne. Multidimensional Fourier series. *International Geophysics*, 97:115–145, 2010. [Link]. 63
- [34] A Rény. On the theory of order statistics. *Acta Mathematica Academiae Scientiarum Hungaricae*, 4:191–231, 1953. 175
- [35] 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]. 63
- [36] Joshua Snoko 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]. 44
- [37] Bharath Sriperumbudur et al. On the empirical estimation of integral probability metrics. *Electronic Journal of Statistics*, pages 1550–1599, 2012. [Link]. 119
- [38] Chang Su, Linglin Wei, and Xianzhong Xie. Churn prediction in telecommunications industry based on conditional Wasserstein GAN. *IEEE International Conference on High Performance Computing, Data, and Analytics*, pages 186–191, 2022. IEEE HiPC 2022 [Link]. 100, 113, 155, 156
- [39] Chris Tofallis. Fitting equations to data with the perfect correlation relationship. *Preprint*, pages 1–11, 2015. Hertfordshire Business School Working Paper [Link]. 22
- [40] 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]. 23, 26
- [41] 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]. 26
- [42] 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]. 63
- [43] Lei Xu and Kalyan Veeramachaneni. Synthesizing tabular data using generative adversarial networks. *Preprint*, pages 1–12, 2018. arXiv:1811.11264 [Link]. 78
- [44] Jinsung Yoon et al. GAIN: Missing data imputation using generative adversarial nets. *Preprint*, pages 1–10, 2018. arXiv:1806.02920 [Link]. 113
- [45] 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]. 83

Index

- activation function, 79, 120, 154
- AdaBoost, 45
- Adam, 154
- Adam gradient descent, 80
- adversarial learning, 71
- agent-based modeling, 164
- AI art, 72
- algorithmic bias, 72
- analytic function, 13
- assignment problem, 101
- asymptotic approximation, 174
- augmented data, 44
- auto-tuning, 101, 112, 118, 120
- autocorrelation function, 178

- batch, 101, 112, 114, 120
- batch size (neural networks), 155
- Bayesian hierarchical models, 113
- Bayesian inference, 53
- Beatty sequences, 177
- binning
 - optimum binning, 46, 95
- bisection method, 148
- boosted trees, 73
- bootstrapping, 24, 72, 102, 174, 178

- categorical feature, 112
- Cauchy-Riemann equations, 13
- CDF regression, 26
- central limit theorem, 174
- collision theory, 165
- color transparency, 29
- computer vision, 21
- confidence interval, 53, 174
- confidence level, 176
- confidence region, 24, 38, 72
 - dual region, 53
- conformal map, 23
- connected components, 83, 132
- contour level, 8, 13
- contour lines, 8
- convergence
 - absolute, 55
 - conditional, 55
- convolution product, 181
- copula, 72, 89, 94, 99, 113, 121, 147, 156
 - Frank, 72, 78
 - Gaussian, 72
- correlation matrix distance, 78, 83, 88, 156
- Cramér's V, 112
- credible interval, 53

- critical line (number theory), 8, 56
- cross-validation, 24, 80, 110, 173
- curse of dimensionality, 58
- curve fitting, 35

- data distillation, 100
- decision tree, 45
- decorrelatation, 88
- decorrelation, 82, 88
- deep comparison, 156
- deep neural network, 79, 94, 120
- diffusion model, 100, 181
- dimensionality reduction, 25
- disaggregation, 57
- discrete Fourier series, 62
- discrete orthogonal functions, 62
- distribution
 - logistic, 25
- distribution function, 174
- dot product, 23
- dummy variable, 45, 79, 95, 112

- ECDF empirical distribution, 111
- eigenvalue, 22
- EM algorithm, 44, 76
- empirical density function, 119
- empirical distribution, 25, 63, 72, 94, 148, 156, 174, 176
 - multivariate, 110, 113, 119, 173
- empirical quantiles, 88, 147
- ensemble methods, 45, 73
- epoch, 79, 82, 95
- epoch (neural networks), 138, 154
- explainable AI, 22, 44, 71, 89, 95, 110, 120, 134
- exponential decay, 49

- feature attribution, 71
- feature clustering, 76, 82, 83, 88, 100, 131, 132, 134, 138
- feature importance, 71
- feature selection, 24
- Fisher transform, 175
- Fourier series, 62
- fractional training set, 134, 135

- GAN, 181
- GAN (generative adversarial networks), 44, 71, 76, 88, 94
- Gaussian mixture model, 94, 181
 - see GMM, 44
- general linear model, 22

generalized linear model, 22
 generative adversarial networks, 94, 110, 154, 181
 see GAN, 44
 generative model, 44
 geometric distribution, 146
 GIS, 59
 GMM (Gaussian mixture model), 75, 76, 88, 89
 gradient descent, 12, 101, 111, 120, 139, 149, 154, 156
 gradient operator, 24
 Gram-Schmidt orthogonalization, 62
 graph
 connected components, 83, 132
 undirected, 83, 132
 greedy algorithm, 55
 grid search, 100, 147, 148

 Hadamard product, 15, 111
 Hellinger distance, 72, 88, 89, 94, 119
 Hermite polynomials, 62
 hidden decision trees, 45, 46
 hierarchical Bayesian model, 121
 hierarchical clustering, 83, 132
 hierarchical deep resampling, 110, 113
 hierarchical GAN, 113
 holdout, 113
 holdout method, 173
 Hungarian algorithm, 101
 hybrid distribution, 146
 hyperparameter, 38, 78, 96, 112, 173

 ill-conditioned problem, 35
 imputation (missing values), 72, 113
 influential feature, 134
 influential observation, 134
 integral probability metrics, 119
 inverse transform sampling, 147, 148

 K-means clustering, 40, 41
 kernel, 181
 kernel density estimation, 181
 key-value pair, 46, 95
 Kolmogorov-Smirnov distance, 94, 97, 112, 113, 119, 156, 173
 Kolmogorov-Smirnov test, 72
 kriging, 55

 label feature, 118
 Lagrange multiplier, 24
 Lasso regression, 24
 latent variables, 79
 learning rate, 10, 78, 83, 88, 139, 154
 LightGBM, 78, 88
 link function, 22, 25
 log-polar map, 23
 logistic distribution, 25
 logistic regression, 25
 unsupervised, 42
 loss function, 79, 82, 88, 100, 101, 111, 112, 120, 154

 maximum contrast estimation, 147
 mean squared error, 24, 39
 medoid, 40

 Mersenne twister, 38
 minimum modulus principle, 13
 missing values, 101
 mixture model, 38, 54, 119
 mode collapse, 102, 156
 model identifiability, 24
 modulus (complex number), 8
 moment estimation method, 147
 multidimensional Fourier series, 63
 multinomial distribution, 96, 172
 multiple root, 56

 natural language processing, 45
 neural network
 activation function, 79
 epoch, 79
 neuron, 79
 NLP (natural language processing), 45
 node (decision tree), 46, 73
 perfect node, 53
 usable node, 47
 node (interpolation), 56, 119

 ordinary least squares, 62
 orthogonal function, 62
 orthogonal trajectory, 8
 overfitting, 24, 72, 78

 Pandas, 172
 parametric bootstrap, 29, 38, 44, 72
 partial derivative, 56
 partial least squares, 22
 partial principal component analysis, 155
 power-geometric distribution, 147
 prediction interval, 24
 predictive power, 46, 53, 71
 principal component analysis, 71, 95, 100, 113
 probability density function, 174, 181

 quantile, 95
 empirical, 72, 148
 extrapolated, 182
 quantile function, 63, 95, 114, 174, 176, 181
 quantile regression, 24

 R-squared, 24, 44
 random search, 148
 regression splines, 22
 reinforcement learning, 83, 89, 120
 rejection sampling, 73, 182
 ReLU function, 79
 resampling, 102
 Riemann Hypothesis, 7, 56
 Riemann zeta function, 8, 56, 146
 Rényi's representation, 175

 saddle point, 10, 13
 SDV (Python library), 77
 seed (random number generator), 73, 78, 82, 112, 154
 Shapley value, 71
 sigmoid function, 79
 simulated annealing, 120

- singular value decomposition, 22
- Sklar's theorem, 72
- smart grid search, 101
- softmax (activation function), 154
- spline regression, 63
- square root (matrix), 82
- steepest descent, 12
- stochastic gradient descent, 79
- stopping rule, 159
- Sturm-Liouville theory, 62
- SVD (Python library), 88
- swap, 111, 120
- swarm optimization, 36
- synthetic data, 22, 36, 38, 55, 61, 72, 147, 182

- TabGAN (Python library), 78
- tensor, 112
- TensorFlow, 78, 112
- textcolor, 44
- time series
 - non-periodic, 34
- total least squares, 22
- total variation distance, 101, 156
- training set, 95, 173
- transformer, 100, 120
- truncated Gaussian, 182

- unsupervised learning, 42

- validation set, 24, 72, 95, 98, 113, 134, 173
- vanishing gradient, 112

- Wasserstein GAN, 100, 120, 155
- Wasserstein loss, 156
- weighted least squares, 22
- weighted regression, 25
- white noise, 36
- wide data, 63, 83, 133

- zeta distribution, 146
- zeta-geometric distribution, 145