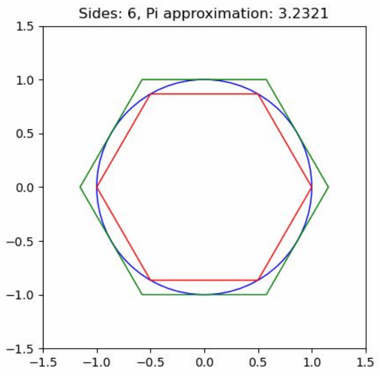
Data Science
Survival Skills

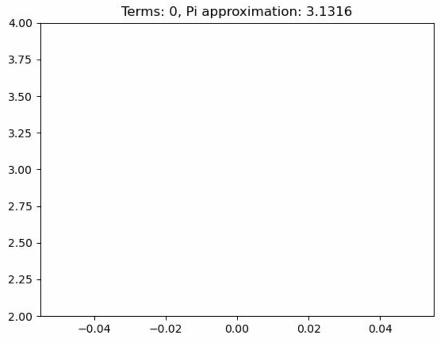
Baselines and Sanity Checks

Multiple ways to reach a goal

Geometric Method: Archimedes' Method

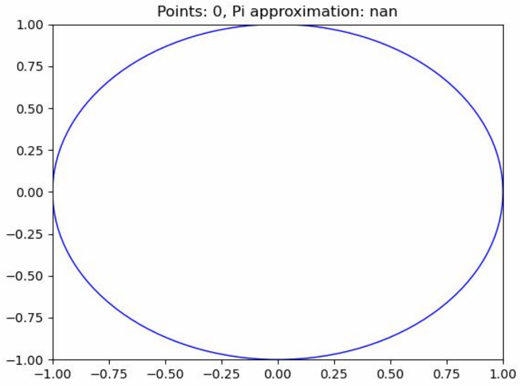


Analytical Method: Infinite Series

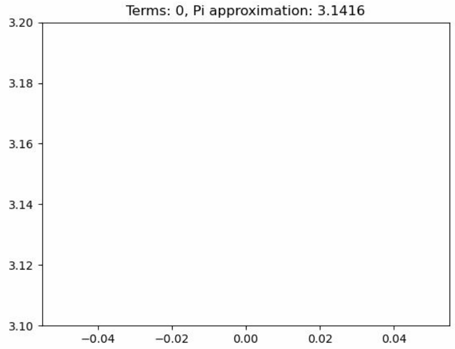


π/4 = 1 - 1/3 + 1/5 - 1/7 + 1/9 - ...

Probabilistic Method: Monte Carlo Simulation



Trigonometric Method: Machin-like Formulas



John Machin's formula
π = 16 arctan(1/5) - 4 arctan(1/239).

+ Series expansion

What can we compare?

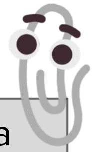
Qualitative
(e.g. Likert scale)

Quantitative

3 ms vs. 10 ms

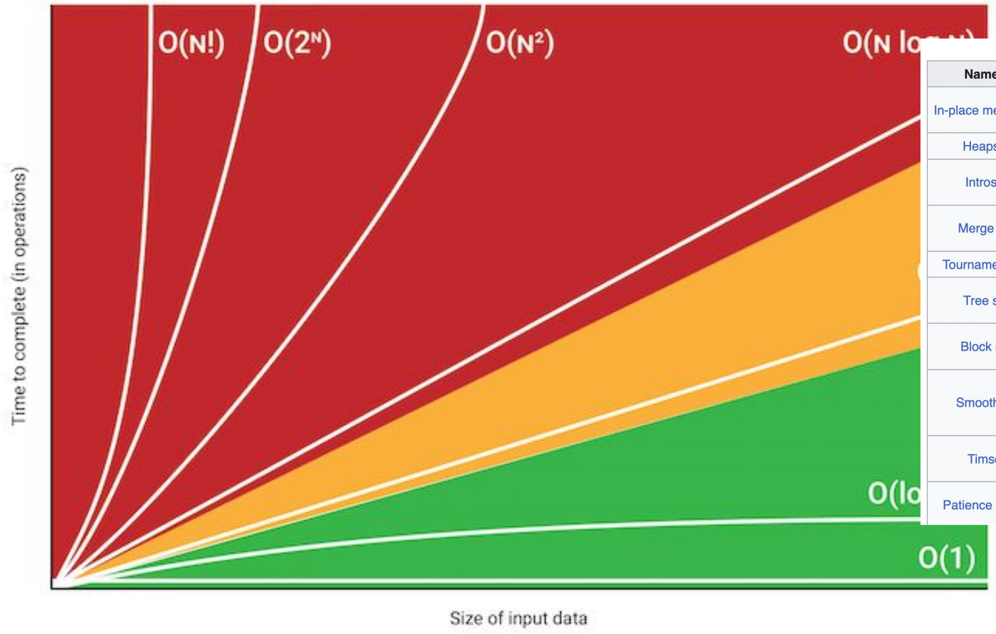
Algorithmic performance

- Error between True and Estimated Value
- Time how long it takes
- How much memory I need
- How much computation I need to perform
- ....



See lecture about data types, categories, etc

Algorithmic complexity (big O)



Sorting algorithms

Comparison sorts

Name	Best	Average	Worst	Memory	Stable	Method	Other notes
In-place merge sort	—	—	$n \log^2 n$	1	Yes	Merging	Can be implemented as a stable sort based on stable in-place merging.[5]
Heapsort	$n \log n$	$n \log n$	$n \log n$	1	No	Selection	
Introsort	$n \log n$	$n \log n$	$n \log n$	$\log n$	No	Partitioning & Selection	Used in several STL implementations.
Merge sort	$n \log n$	$n \log n$	$n \log n$	n	Yes	Merging	Highly parallelizable (up to $O(\log n)$ using the Three Hungarians' Algorithm).[6]
Tournament sort	$n \log n$	$n \log n$	$n \log n$	n[7]	No	Selection	Variation of Heapsort.
Tree sort	$n \log n$	$n \log n$	$n \log n$ (balanced)	n	Yes	Insertion	When using a self-balancing binary search tree.
Block sort	n	$n \log n$	$n \log n$	1	Yes	Insertion & Merging	Combine a block-based $O(n)$ in-place merge algorithm[8] with a bottom-up merge sort.
Smoothsort	n	$n \log n$	$n \log n$	1	No	Selection	An adaptive variant of heapsort based upon the Leonardo sequence rather than a traditional binary heap.
Timsort	n	$n \log n$	$n \log n$	n	Yes	Insertion & Merging	Makes *n-1* comparisons when the data is already sorted or reverse sorted.
Patience sorting	n	$n \log n$	$n \log n$	n	No	Insertion & Selection	Finds all the longest increasing subsequences in $O(n \log n)$.

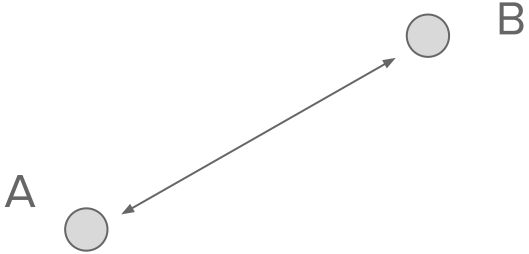
https://towardsai.net/p/programming/big-o-notation-what-is-it

Computing Pi

Method	Advantages	Disadvantages	Performance (Big O Notation)	Implementation	Intuition	Computational Effort	Lag to Correct Pi Approximation
Archimedes' Method	Conceptually simple and visual	Computationally intensive for high accuracy	$O(n^2)$ for n-sided polygon	+	++	--	-
Monte Carlo Simulation	Easy to implement; probabilistic insight	Requires many points for high accuracy; statistical variability	$O(n)$ for n points	++	+	--	-
Leibniz Formula	Precise and systematic	Slow convergence; many terms for high accuracy	$O(n)$ for n terms	++	o	-	--
Machin-like Formulas	Highly accurate; efficient computation	Requires understanding of advanced math	$O(n)$ for n terms	-	--	+	++

Metrics

Computing a distance



L2: low-dimensional space, affected by outliers, magnitude of vectors important.

L1: Grid-like patterns (Urban layout, chessboard,...) and high-dimensional space, Travel along axes and not vertically

Euclidean distance (L2 norm):

$$\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

x and y are two points in the i-th dimension

Manhattan distance (L1 norm):

$$\sum_{i=1}^{n}|x_i - y_i|$$

Minkowski distance (Lx norm):

Formula: $\left(\sum_{i=1}^{n}|x_i - y_i|^p\right)^{\frac{1}{p}}$

Generalization of Euclidean (p=2) and Manhattan (p=1) distances.

Considerations for choosing a distance metric

- **Data Dimensionality**:
  Euclidean distance can be less effective in high-dimensional spaces (curse of dimensionality), while Manhattan distance can perform better.

- **Outlier Sensitivity:**
  If your data has outliers or noise, Manhattan distance can be more robust as it is less influenced by extreme values.

- **Problem Domain:** The nature of your problem might dictate the most appropriate distance metric (e.g., Manhattan for grid-based problems).

Relation to errors

Mean Absolute Error (MAE)

- Formula: $\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$
- Where y_i is the actual value and $\hat{y}_i$ is the predicted value.

Mean Squared Error (MSE)

- Formula: $\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$

Root Mean Squared Error (RMSE)

- Formula: $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$

More distances

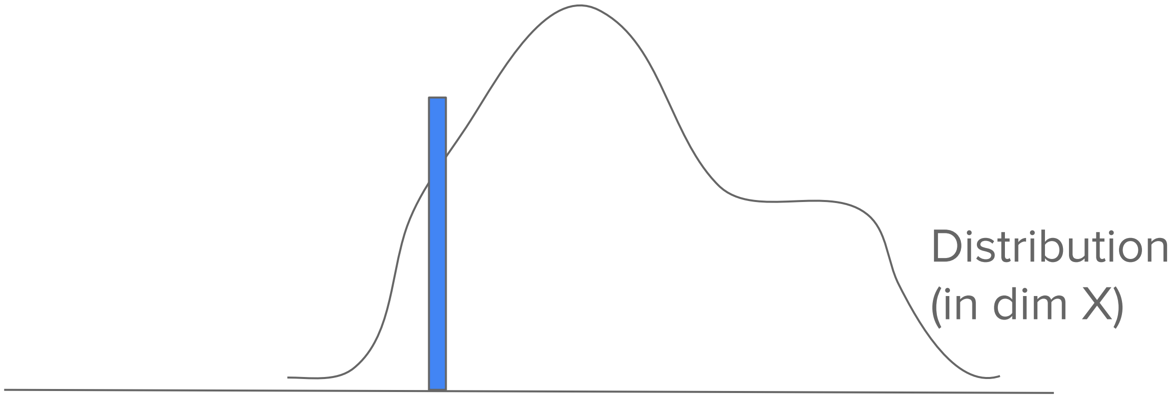
- Chebyshev Distance (max. Distance along any coordinate dimension)

- Cosine Similarity (cosine of the angle between two vectors)

- Hamming distance (# positions corresponding elements are different)

More distances

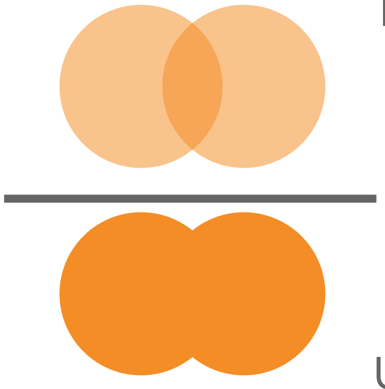
- Mahalanobis distance (Distance between a point and a distribution, considering correlations in a multi-dimensional setting)



$$D_M(x) = \sqrt{(x-\mu)^\top S^{-1}(x-\mu)}$$

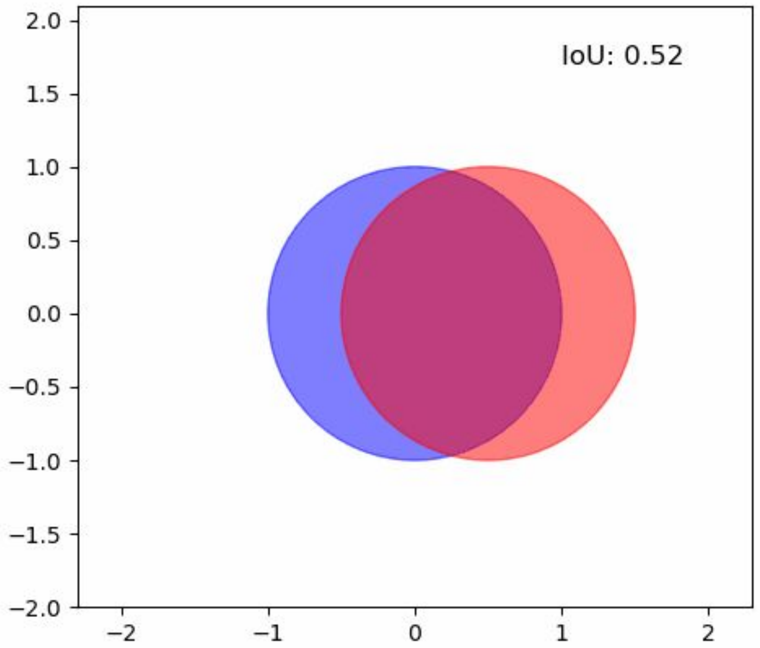
where x is the vector of observed values, μ is the mean vector, and S is the covariance matrix.

Jaccard distance



Intersection

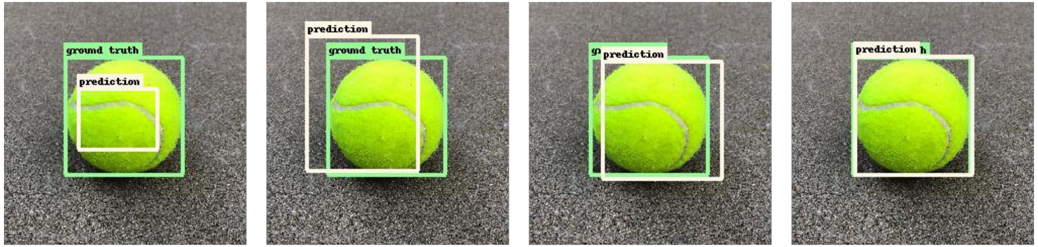
Union



Bound by 0 (no intersection)
and 1 (perfect match, I == U)

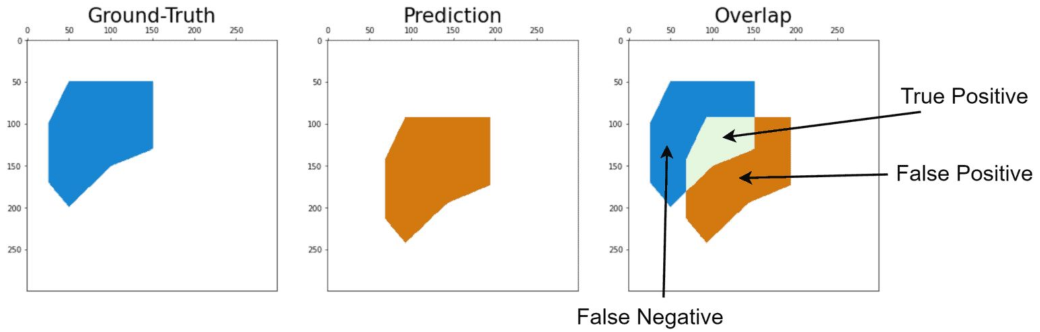
$$DSC = \frac{2|X \cap Y|}{|X| + |Y|}$$

Applications



IoU: 35% | IoU: 55% | IoU: 75% | IoU: 95%

https://apple.github.io/turicreate/docs/userguide/object_detection/advanced-usage.html



$$IoU = \frac{|G \cap P|}{|G \cup P|} = \frac{TP}{TP + FP + FN}$$

$$Dice = \frac{2 * |G \cap P|}{|G| + |P|} = \frac{2TP}{2TP + FP + FN}$$

What are TPs/FPs/...?

		Predicted condition	
	Total population = P + N	Positive (PP)	Negative (PN)
Actual condition	Positive (P)	True positive (TP)	False negative (FN)
	Negative (N)	False positive (FP)	True negative (TN)

Examples from Wikipedia

A confusion matrix

Individual Number	1	2	3	4	5	6	7	8	9	10	11	12
Actual Classification	1	1	1	1	1	1	1	1	0	0	0	0
Predicted Classification	0	0	1	1	1	1	1	1	1	0	0	0
Result	FN	FN	TP	TP	TP	TP	TP	TP	FP	TN	TN	TN

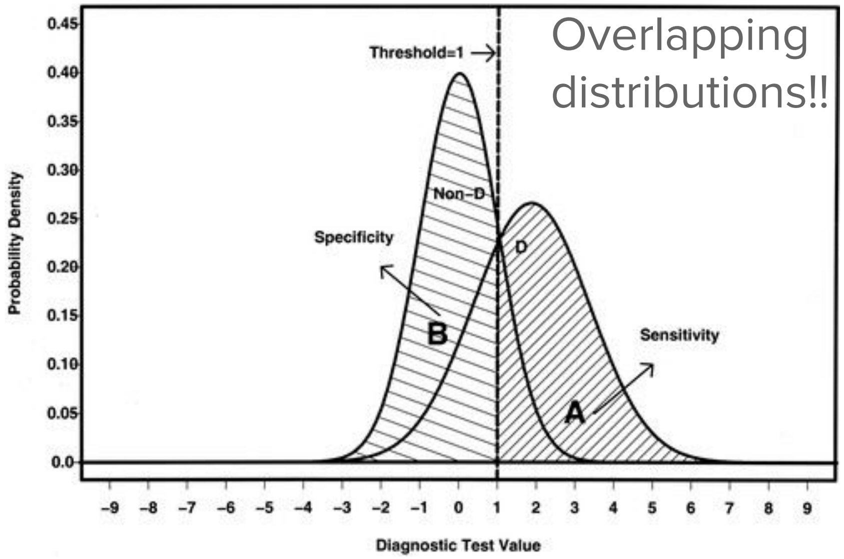
Does not matter how we achieve the prediction for now...

		Predicted condition	
	Total 8 + 4 = 12	**Cancer** 7	**Non-cancer** 5
Actual condition	**Cancer** 8	6	2
	Non-cancer 4	1	3

The confusion matrix

		Predicted condition		Sources: [22][23][24][25][26][27][28][29][30] view · talk · edit	
	Total population $= P + N$	**Predicted Positive (PP)**	**Predicted Negative (PN)**	Informedness, bookmaker informedness (BM) $= TPR + TNR - 1$	Prevalence threshold (PT) $= \frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	**Positive (P)**	**True positive (TP),** **hit**	**False negative (FN),** **type II error, miss, underestimation**	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{TP}{P} = 1 - FNR$	False negative rate (FNR), miss rate $= \frac{FN}{P} = 1 - TPR$
	Negative (N)	**False positive (FP),** **type I error, false alarm, overestimation**	**True negative (TN),** **correct rejection**	False positive rate (FPR), probability of false alarm, fall-out $= \frac{FP}{N} = 1 - TNR$	True negative rate (TNR), specificity (SPC), selectivity $= \frac{TN}{N} = 1 - FPR$
	Prevalence $= \frac{P}{P + N}$	Positive predictive value (PPV), precision $= \frac{TP}{PP} = 1 - FDR$	False omission rate (FOR) $= \frac{FN}{PN} = 1 - NPV$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$	Negative likelihood ratio (LR−) $= \frac{FNR}{TNR}$
	Accuracy (ACC) $= \frac{TP + TN}{P + N}$	False discovery rate (FDR) $= \frac{FP}{PP} = 1 - PPV$	Negative predictive value (NPV) $= \frac{TN}{PN} = 1 - FOR$	Markedness (MK), deltaP (Δp) $= PPV + NPV - 1$	Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$
	Balanced accuracy (BA) $= \frac{TPR + TNR}{2}$	F_1 score $= \frac{2 PPV \times TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN}$	Fowlkes–Mallows index (FM) $= \sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) $= \sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times FDR}$	Threat score (TS), critical success index (CSI), Jaccard index $= \frac{TP}{TP + FN + FP}$

Sensitivity and Specificity



D: Diseased,
Non-D: “Healthy”

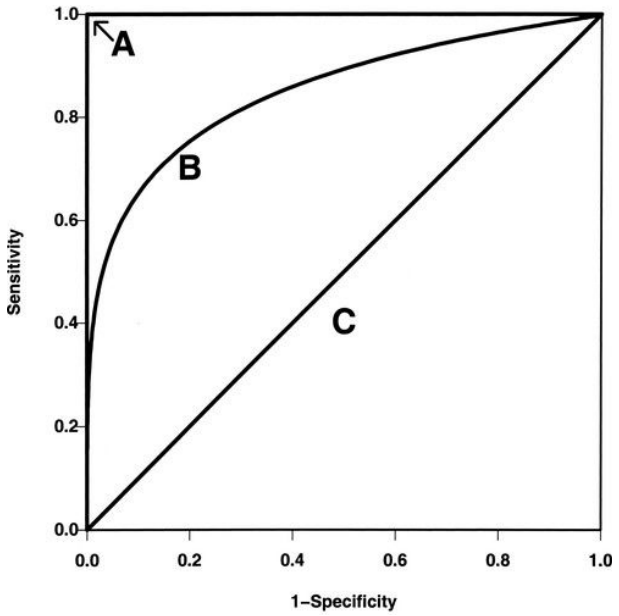
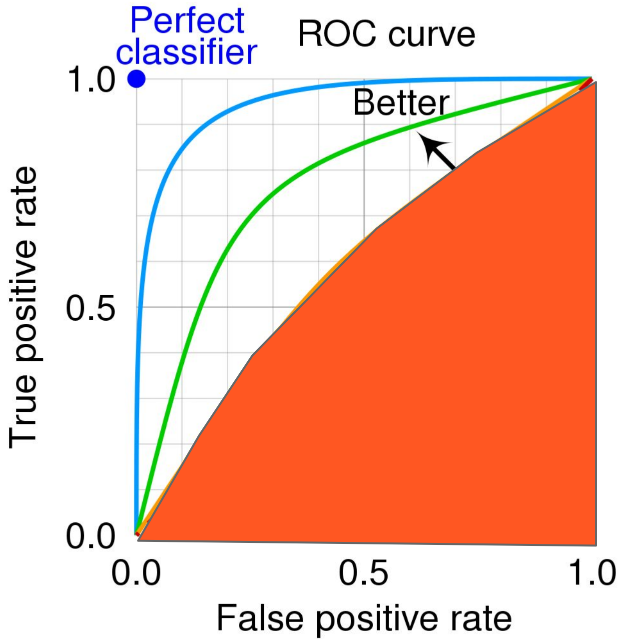


Figure 2. Three hypothetical ROC curves representing the diagnostic accuracy of the gold standard (lines A; AUC=1) on the upper and left axes in the unit square, a typical ROC curve (curve B; AUC=0.85), and a diagonal line corresponding to random chance (line C; AUC=0.5). As diagnostic test accuracy improves, the ROC curve moves toward A, and the AUC approaches 1.

https://www.ahajournals.org/doi/full/10.1161/CIRCULATIONAHA.105.594929

ROC Curve



AUC (Area under the curve).

The higher, the better;
Used to compare different classifiers

Problematic:
Noise, Class Imbalance (favoring the majority class, not informative about best threshold, scale invariance ➔ ranks predictions rather than use their absolute values)

Examples

COVID-19 pandemic:

We needed something that is very **sensitive**

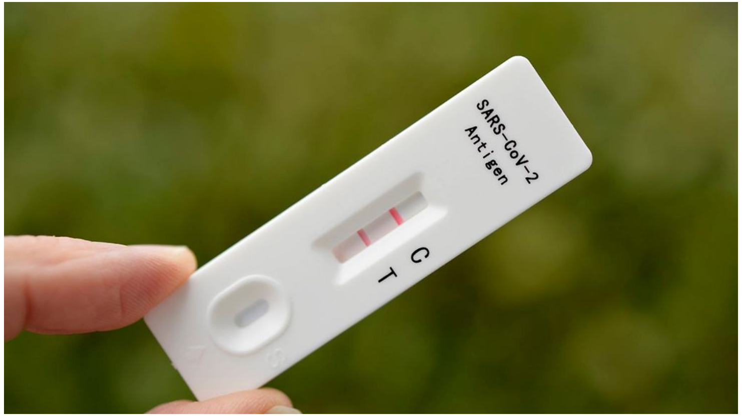


Foto: imago images/Michael Weber

Cancer diagnosis:

We need something that is very **specifi** for rare, but serious diseases:

- Treatment depends on it
- Anxiety and distress for patient, fa etc.

Recall and precision

$$\text{Precision} = \frac{TP}{TP + FP}$$

=Sensitivity

How many of the positive identified instances are really positive.

$$\text{Recall} = \frac{TP}{TP + FN}$$

How many real positive instances were correctly identified?

Sometimes it is not easy to identify REAL NEGATIVES. => Specificity cannot be determined, recall and precision better metrics

Accuracy and F1 Score

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

Where *TP* = True Positives, *TN* = True Negatives, *FP* = False Positives, and *FN* = False Negatives.

Class Imbalance: If out of 100 instances, 95 are P and 5 are N, you get 95% accuracy by assuming constantly P.

https://developers.google.com/machine-learning/crash-course/classification/accuracy

F1 Score

$$\text{F1 Score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Harmonic mean of Precision and Recall

Harmonic vs. arithmetic vs. geometric mean. Important for **average of rates**, e.g. average speed for a given distance or average resistance in parallel circuit

Baselines and Benchmarks

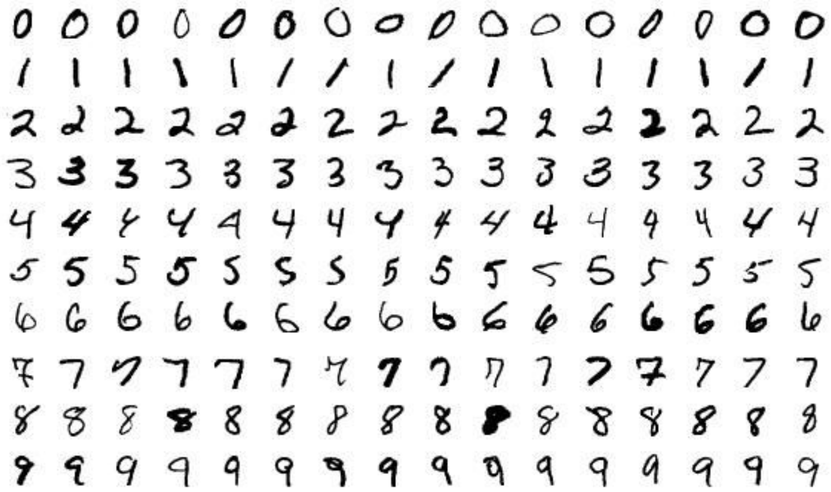
Baseline algorithm

Naive approach, e.g. computing odds

Classical math/analytic, exact solution

Established ML/DL solutions

Benchmark dataset

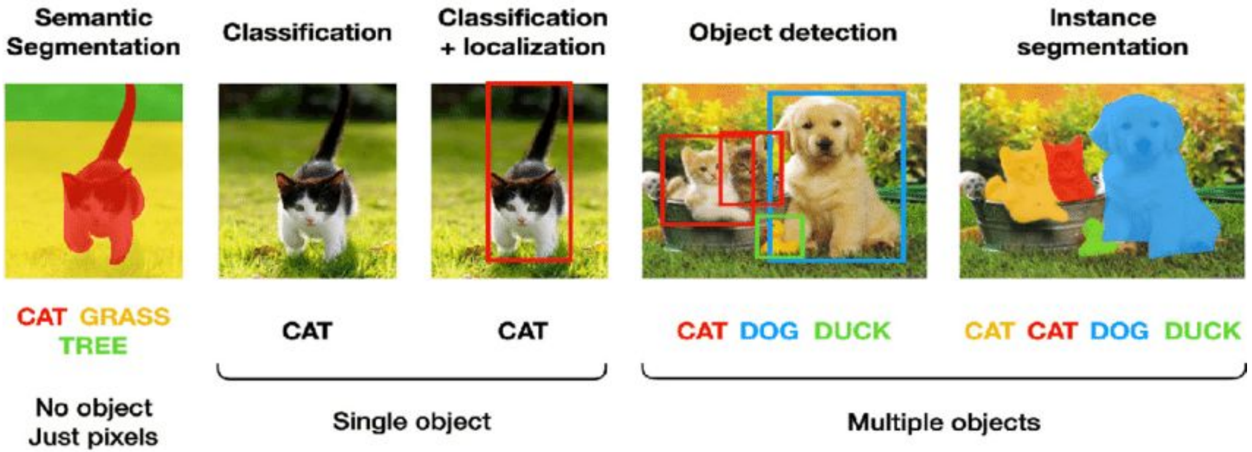


The MNIST database

Modified National Institute of Standards and Technology

60k training, 10k test,
Numbers 0-9 (10 classes)

Tasks



Audio

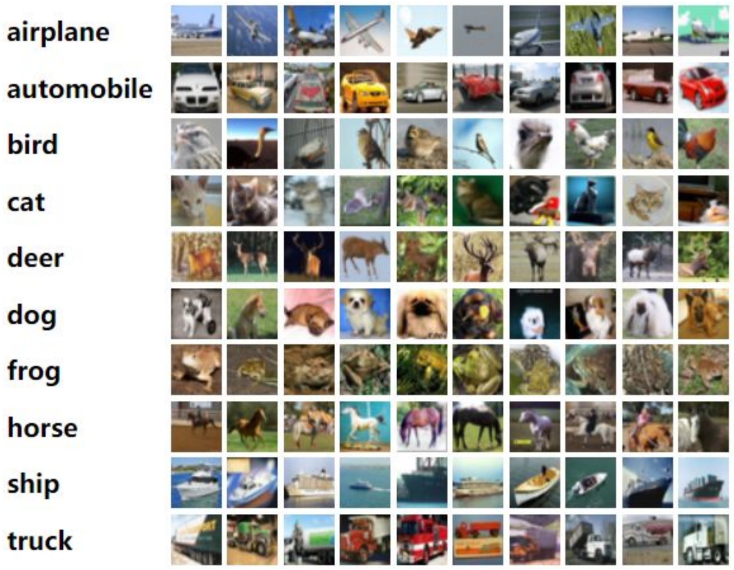
Text

Medical

....

Benchmark datasets

CIFAR-10 (10 classes)



ImageNet (15M images, 1k classes)

Cityscapes (autonomous driving, instance segmentation)

Dense annotated images from a car driving through 50 different cities.

30 classes grouped into 8 categories (flat surfaces, humans, vehicles, constructions, objects, nature, sky, and void)

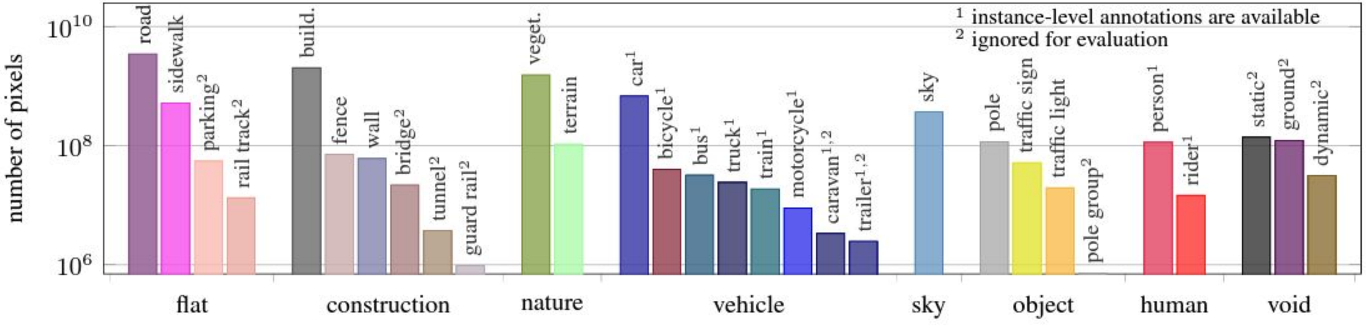
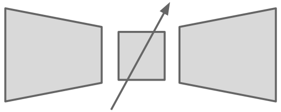
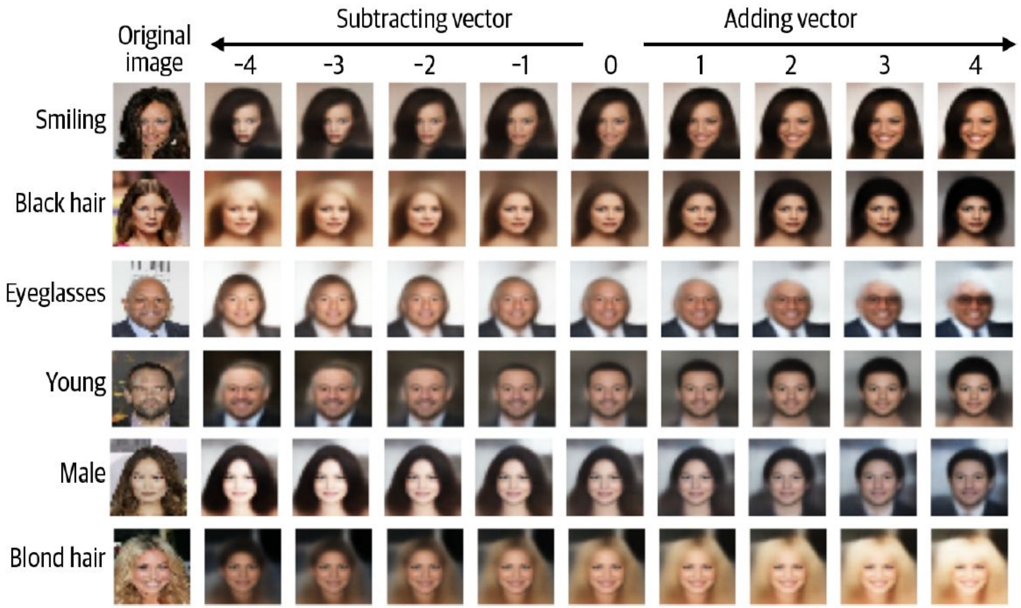


Figure 1. Number of finely annotated pixels (y-axis) per class and their associated categories (x-axis).

CelebA dataset



Multiclass labels, good for feature manipulation



David Foster, Generative Deep Learning 2023

Sanity Checks/Synthetic data

Before you work on real data, use something you know the result of.

Performance evaluation on data that you know

- You know the desired outcome
- You can check for logical errors
- Identify algorithm limitations
  - Distributions
  - Shapes
  - Dimensionality
  - Approximate big O
  - ...

Missing Data in Data Science

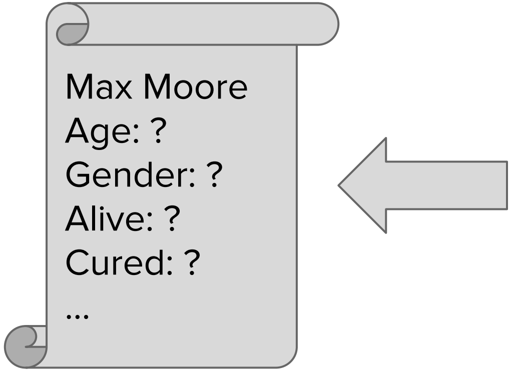
Missing data occurs when information is not stored for certain observations or features in a dataset.

We and Machine learning models require complete data to make accurate predictions. Missing data can lead to biased outcomes and models, incorrect conclusions, and reduced model performance.

Common issue in real-world datasets

Max Moore
Age: ?
Gender: ?
Alive: ?
Cured: ?
...

Key challenge



The key challenge is to **accurately estimate or handle these missing values** to **maintain the integrity and performance of the model**.

⇒ NO FREE LUNCH / ONE FITS ALL solution

Understanding the Types of Missing Data

Missing Completely at Random (MCAR):

- Definition: Data is MCAR when the probability of a data point being missing is the same for all observations. It is independent of both observed and unobserved data.
- Example: In a survey, if respondents randomly skip questions due to lack of attention, the missingness is MCAR.

Missing at Random (MAR)

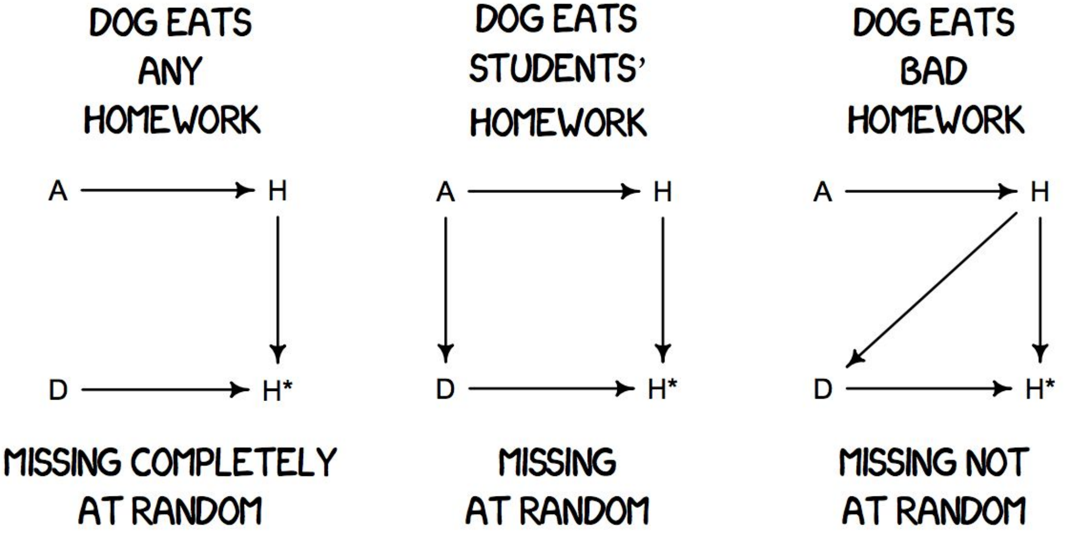
- Definition: Data is MAR when the probability of a data point being missing is related only to available information (observed data), not the missing data.
- Example: In a health survey, if younger people are less likely to report their age, the missingness of age data is MAR, as it is related to another variable in the dataset (age group).

Missing Not at Random (MNAR):

- Definition: Data is MNAR if its missingness is related to unobserved data, i.e., the reason for missingness is related to the value that's missing.
- Example: If people with higher incomes are less likely to disclose their earnings, the missingness in income data is MNAR, as it directly relates to the missing data itself.

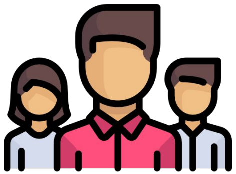
MCAR, MAR, MNAR - visually explained

H: Homework
H*: Homework with missing values
A: Attribute of student
D: Dog (missingness mechanism)



Richard McElreath
@rlmcelreath

Reasons for missing data



Technical challenges:

- Hardware or Software Malfunction
  ⇒ broken sensors
- Data Transfer
  ⇒ unstable transmission or storage

Human factors

- Individuals do not answer all questions in survey (sensitive topics, e.g. income, health)
- Mistakes in data entry: typos, omissions, wrong row/column

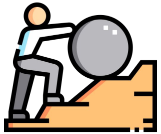
Systemic errors

- Selection Bias (way individuals/items are selecting for study ⇒ non-representative sample)
- Censoring (partially observed, time-to-event data ➔ time until a machine fails, time until a patient recovers…)

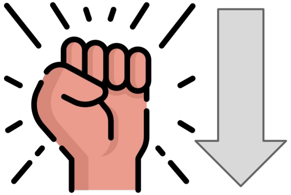
Impact of Missing Data on Models

Missing data can lead to a significant reduction in model **accuracy**, as the model might be trained on an **incomplete representation** of the problem space.

When **missing data is not random**, it can lead to biased models that do **not correctly represent the underlying population or phenomena.**

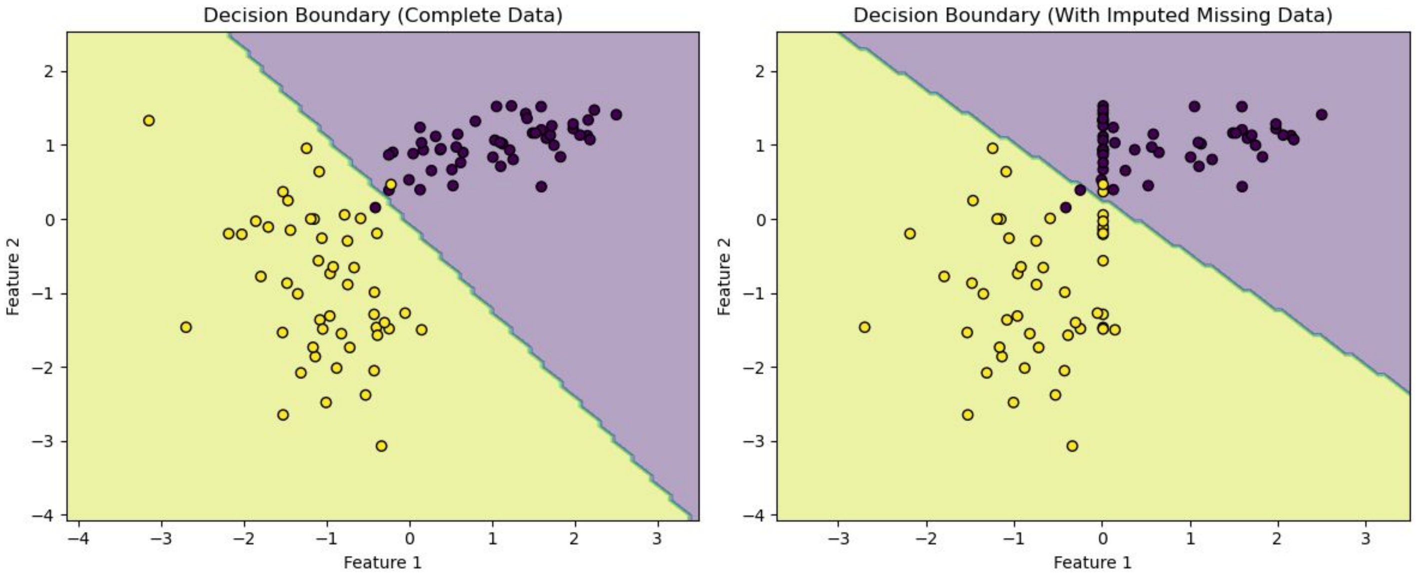


Handling missing data often **complicates the process of training and validating models**, requiring additional steps (maybe come up with data) and considerations (how to train/select the model)



Missing data can reduce the **statistical power** of a model, leading to less confident predictions and conclusions

Decision boundary differences



Strategies for Handling Missing Data

Multiple methods available: all have their strengths/weaknesses/prior assumptions

DELETION METHODS

- List-wise deletion
- Pair-wise deletion

SINGLE IMPUTATION METHODS

- Mean/Median/Mode imputation (easy, but maybe bias)
- Regression Imputation (assumes relationship between features)
- K-NN Imputation (effective for non-lin. relationships)

MULTIPLE IMPUTATION METHODS

Multivariate Imputation by Chained Equations (MICE) - uncertainty based. Should reduce bias and increases robustness, but more complex.

Imputation using DL (e.g. Autoencoders)

Deletion Methods for Handling Missing Data

Only for MCAR

Listwise Deletion (Complete Case Analysis):

Removal of **any records** (rows) from the dataset that contain **any missing values**.

pro/con:
Simple to implement. Best used when the amount of missing data is minimal and **MCAR**. However, it can significantly reduce the dataset size and lead to biased results if the missingness is not **MCAR**.

Pairwise Deletion:

Using **all available data** for each **individual analysis,** without deleting entire records. ⇒ Each analysis might use **a different subset** of the data based on availability.

pro/con:
Useful in **correlation or covariance analyses** where complete cases for pairs of variables are used. Reduces data loss compared to listwise deletion but can lead to inconsistent results across different analyses

Example

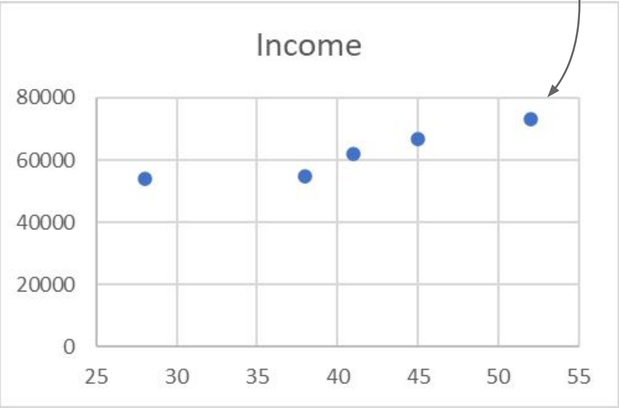
	A	B	C	D
1	Name	Age	Income	Coun
2	John Doe	28	54000	USA
3	Jane Smith		58000	UK
4	Ali Khan	33		Pakistan
5	Maria Lee	45	67000	South Korea
6	Steve Ray	52	73000	
7	Lucy Liu		61000	China
8	Omar Sy	38	55000	France
9	Emma Stone	30		USA
10	Raj Patel		50000	India
11	Ana Maria	41	62000	Brazil

List

	A	B	C	D
1	Name	Age	Income	Country
2	John Doe	28	54000	USA
	Maria Lee	45	67000	South Korea
	mar Sy	38	55000	France
5	Ana Maria	41	62000	Brazil

Pair

	A	B	C	D
1	Name	Age	Income	Country
2	John Doe	28	54000	USA
	Maria Lee	45	67000	South Korea
	Steve Ray	52	73000	
5	Omar Sy	38	55000	France
6	Ana Maria	41	62000	Brazil



Single Imputation Techniques

Mean/Median/Mode Imputation

Works for numerical data (mean, median), and for categorical data (mode) - can distort data distribution and mask the variance

MCAR

Regression Imputation

Estimate missing value based on regression model using the other variables (linear relationship!)

More accurate than mean/median/mode if there is a linear relationship between variables. However, it assumes such a relationship and can underestimate variability.

MAR

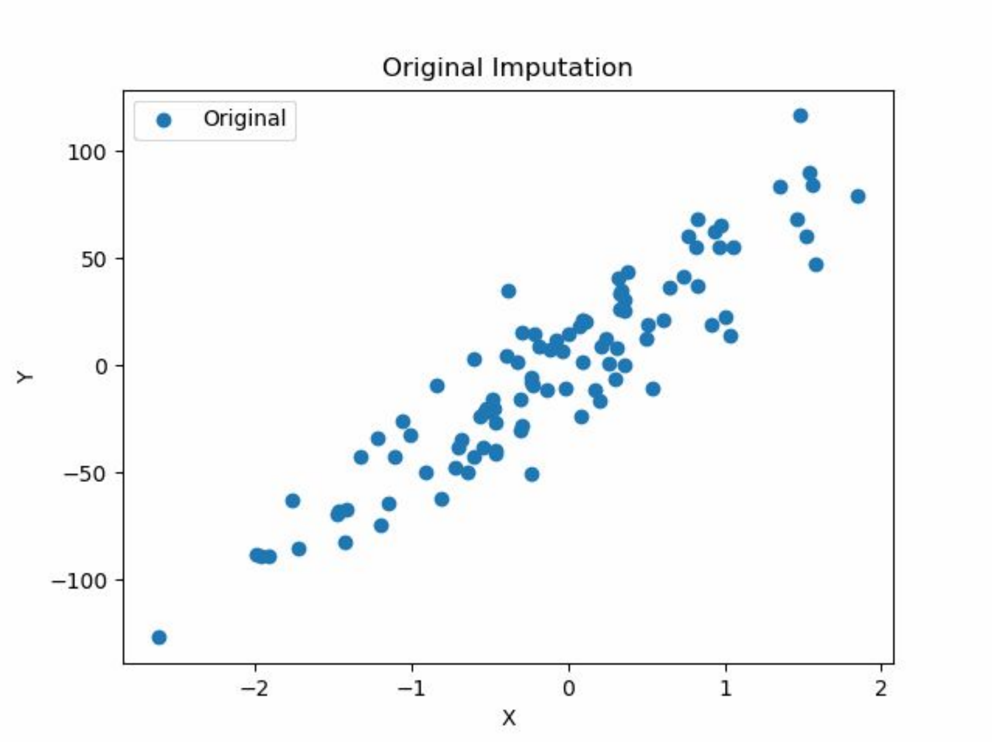
K-Nearest Neighbors (K-NN) Imputation

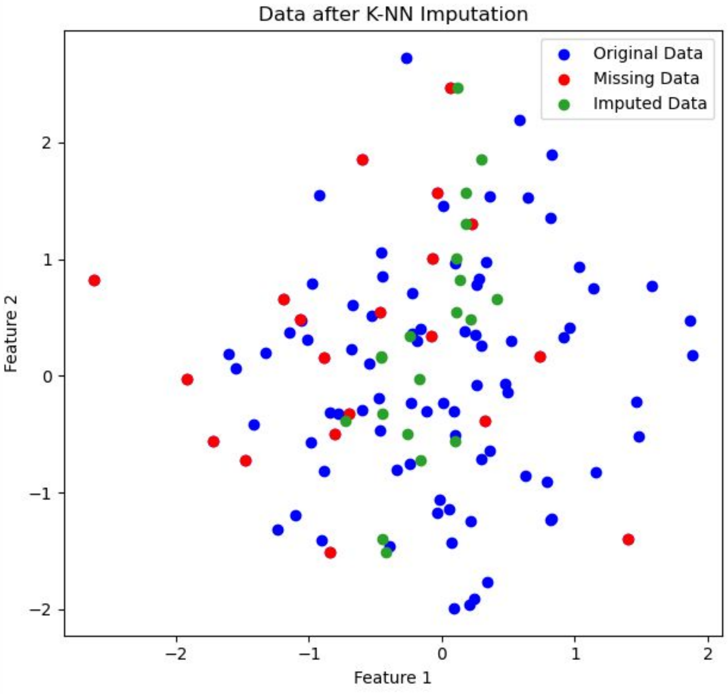
Using the **nearest neighbors**' values to impute missing data. Typically based on a **similarity** measure like **Euclidean distance**.

Effective for non-linear relationships and more complex data structures. However, it's computationally intensive and sensitive to outliers

MAR

Examples


Original Imputation
Original
100
50
0
−50
−100
Y
−2
−1
0
1
2
X

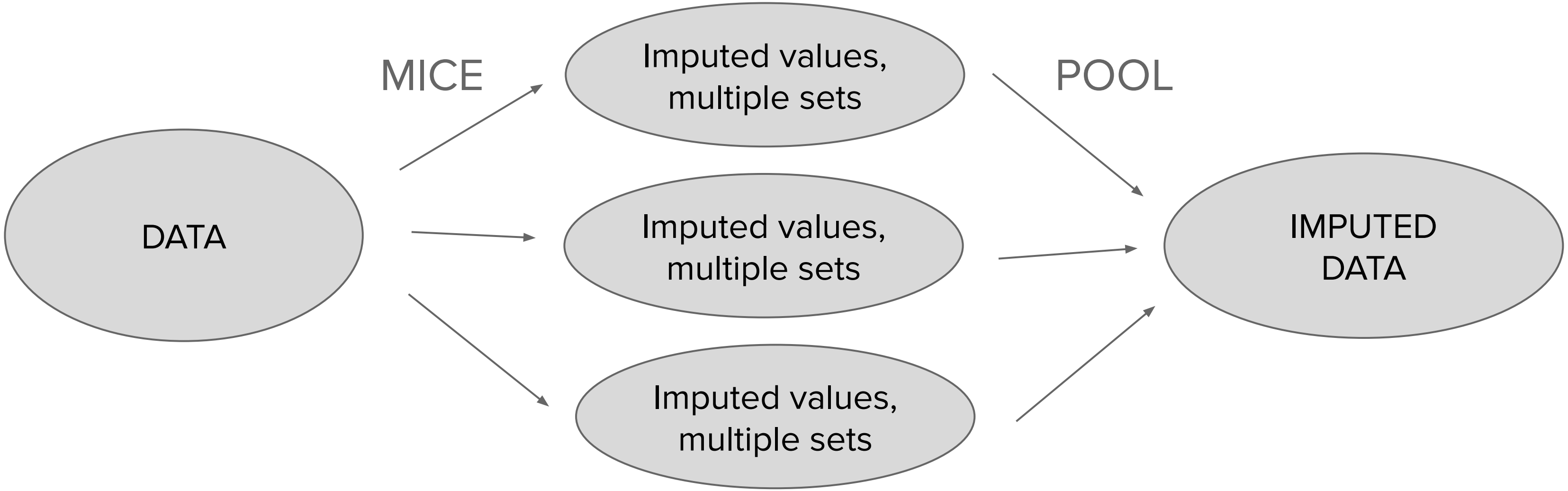

Data after K-NN Imputation
Original Data
Missing Data
Imputed Data
2
1
0
−1
−2
Feature 2
−2
−1
0
1
2
Feature 1

Multiple Imputation

Multiple imputation involves creating **several different imputed datasets** and then combining the results. This approach **accounts for the uncertainty** inherent in the imputation process.

⇒ Provides a more accurate and robust method of dealing with missing data, especially when the data is MAR or **MNAR**.

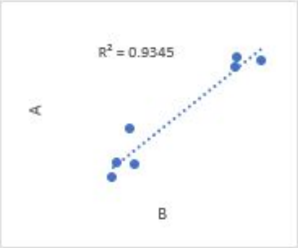
Famous algorithm: Multivariate Imputation by Chained Equations (MICE)



MICE Algorithm

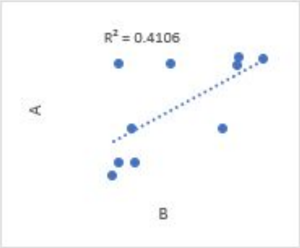
Missing data is in red. There is a strong correlation between A and B, so let's try to impute A using B and C.

A	B	C
0.93	1.40	1.53
0.24	0.46	0.76
	0.80	
0.95	1.24	1.46
0.23	0.57	
0.90		1.28
0.15	0.42	
0.47	0.54	0.63
	1.14	
0.89	1.23	1.45



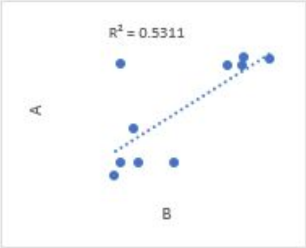
Missing data is filled in randomly. This dillutes the correlations, but allows us to impute using all available data.

A	B	C
0.93	1.40	1.53
0.24	0.46	0.76
0.90	0.80	1.53
0.95	1.24	1.46
0.23	0.57	1.28
0.90	0.46	1.28
0.15	0.42	1.53
0.47	0.54	0.63
0.47	1.14	1.28
0.89	1.23	1.45



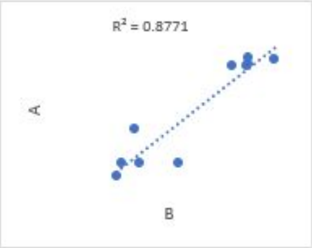
A random forest is used to predict A with B and C. Notice the correlation between A and B improved.

A	B	C
0.93	1.40	1.53
0.24	0.46	0.76
0.24	0.80	1.53
0.95	1.24	1.46
0.23	0.57	1.28
0.90	0.46	1.28
0.15	0.42	1.53
0.47	0.54	0.63
0.89	1.14	1.28
0.89	1.23	1.45



After Imputing B using A and C, we have achieved a correlation between A and B much closer to the original data.

A	B	C
0.93	1.40	1.53
0.24	0.46	0.76
0.24	0.80	1.53
0.95	1.24	1.46
0.23	0.57	1.28
0.90	1.24	1.28
0.15	0.42	1.53
0.47	0.54	0.63
0.89	1.14	1.28
0.89	1.23	1.45

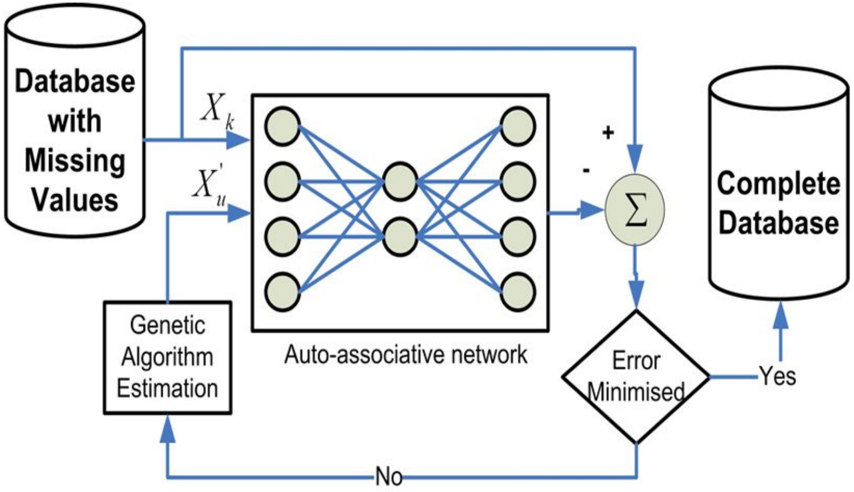


Advanced Techniques

Missing Data: A Comparison of Neural Network and Expectation Maximisation Techniques

Fulufhelo V. Nelwamondo, Shakir Mohamed and Tshilidzi Marwala

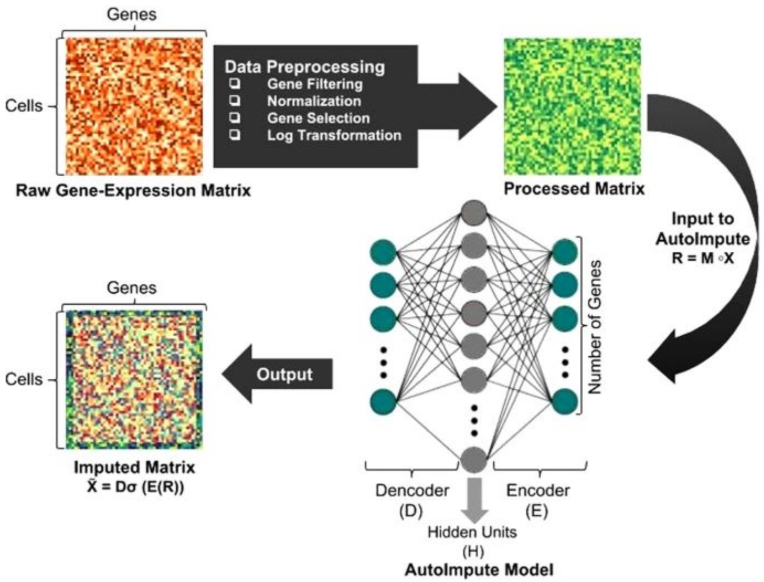
*School of Electrical and Information Engineering, University of the Witwatersrand
Private Bag 3, Wits, 2050, South Africa,
e-mail: f.nelwamondo@ee.wits.ac.za, s.mohamed@ee.wits.ac.za t.marwala@ee.wits.ac.za



Article | Open access | Published: 05 November 2018

AutoImpute: Autoencoder based imputation of single-cell RNA-seq data

Divyanshu Talwar, Aanchal Mongia, Debarka Sengupta & Angshul Majumdar



Data augmentation

Scope

Applicable across **various data types** – including tabular, audio, images, textual, and time-series data. Each data type has specific augmentation techniques that are best suited to its characteristics.

Definition and Purpose

Data Augmentation is the process of artificially expanding the size and diversity of a dataset by creating modified versions of the data points. This technique helps in **preventing overfitting, enhancing model generalization, and improving performance, especially when original data is limited or imbalanced.**

General benefits

Improves model robustness by simulating a variety of scenarios and conditions. Particularly useful in deep learning, where **large datasets are often required**

Data Augmentation in Tabular Data

Sampling with variance:

$$x_{\text{new}} = x + N(0, \sigma^2)$$

Sampling with variance involves adding a small amount of random noise to the existing data points. This technique generates new data points that are variations of the existing ones.

Synthetic Minority Over-Sampling Technique (SMOTE):

SMOTE is used primarily in the context of classification problems to **address imbalances between classes**. It generates synthetic samples for the minority class by **interpolating between existing minority instances.**

$$x_{\text{new}} = x_i + \lambda \times (x_{\text{nn}} - x_i)$$

where λ is a random number between 0 and 1.

Data Augmentation on Audio

Convolutional neural network-based cross-corpus speech emotion recognition with data augmentation and features fusion

Rashid Jahangir[1,2] · Ying Wah Teh[1] · Ghulam Mujtaba[3] · Roobaea Alroobaea[4] · Zahid Hussain Shaikh[5] · Ihsan Ali[1]

Received: 5 April 2021 / Revised: 23 December 2021 / Accepted: 22 February 2022 / Published online: 28 March 2022

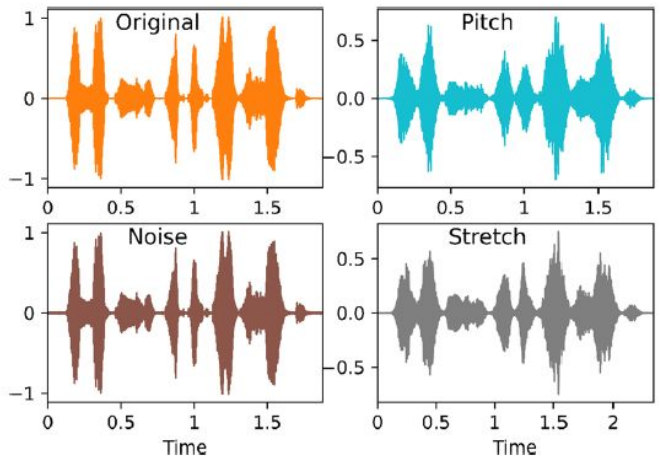


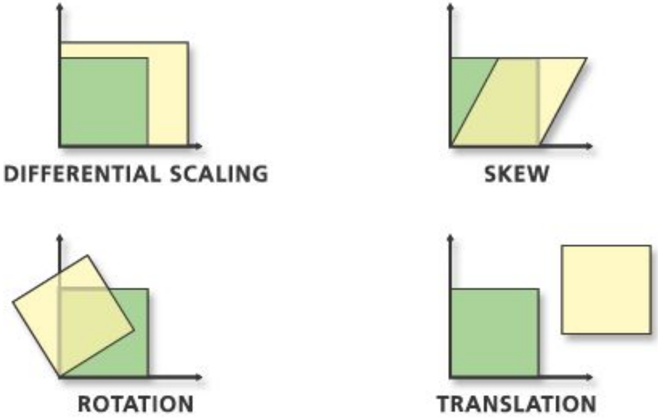
Fig. 3 Data augmentation methods applied on audio signal

+ Padding
+ Sentiment transfer
+ Speaker's identity transfer (Voice cloning)

Data Augmentation on Images

© albumentations

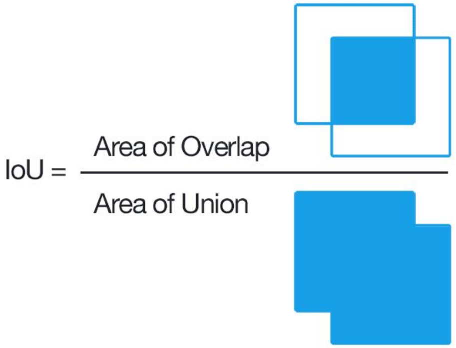
Affine transformations



https://desktop.arcgis.com/de/arcmap/latest/tools/coverage-toolbox/how-transform-works.htm

Homework

In this homework assignment, you will implement the IoU score and calculate it on some dummy data. You will then delve into the area of data augmentation with the "albumentations" library.



https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/