Probabilities and statistics for machine learning and data science

Tiphaine Viard

2021 – 2022

Goals and outline

Brush up on the parts of probabilities and statistics you will need

Probabilities: Random variables (and their types), conditional probabilities, Bayes' theorem, "naive" Bayes

Statistics: Mean, median, mode and standard deviation, distributions

Machine learning: Prepare your data, common pitfalls, bias-variance trade-off

Give technical and conceptual tools to use in your daily practice

Random variables

Quantify some uncertainty on a population

- **Discrete**: dice roll, coin flip, number of people, etc.
- **Continuous**: height, waiting time, etc.

Random variable $\neq$ realization

Random variables' realizations follow a **distribution**, $X \sim P(X)$

- Cumulative $(P(X \geq x))$ or not $(P(X = x))$
- Mass function vs density functions
- $\sum_X P(X) = 1$, $P(X) > 0$
- Independence: $A \perp\!\!\!\perp B \Leftrightarrow P(A \cap B) = P(A)P(B)$

Conditional probability

The probability that A happens knowing B has happened is:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

e.g. Prob. that it rains given that the ground is wet

Be careful, in general, $P(B|A) \neq P(A|B)$

Bayes' theorem

So, what is the link between $P(A|B)$ and $P(B|A)$?

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

This is one of the key theorems of machine learning and data science

The Naive Bayes classifier

Suppose we want to classify email as **legitimate** or **spam**. Given an email, described by features $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ we want to **classify** it into K classes $\{c_1, \ldots, c_K\}$.

$$p(\boldsymbol{x}|c_k) = \frac{p(c_k|\boldsymbol{x})p(c_k)}{p(\boldsymbol{x})} = \frac{1}{p(\boldsymbol{x})} p(c_k) \prod_i p(x_i|c_k)$$

Trick: assume $\forall i, j < n, x_i$ independent x_j

$$\hat{y} = \arg\max_k p(\boldsymbol{x}|c_k)$$

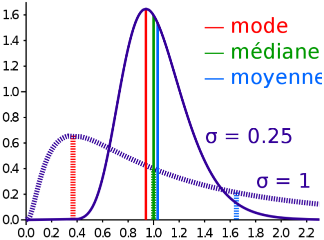
Easy to implement, scalable, performs surprisingly well

The basics: mean, median, mode

For a series of points X:

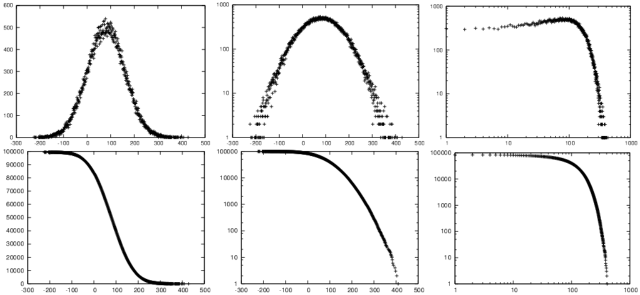
- Mean: $\frac{1}{|X|}\sum_i x_i$, the average value
- Median: the value separating X in two subsets of equal size
- Mode: the most frequent value in X
- Standard deviation: root mean square of distances to mean

When Billie Eilish walks into a coffee shop, the mean income soars...



Source: *Wikipedia*

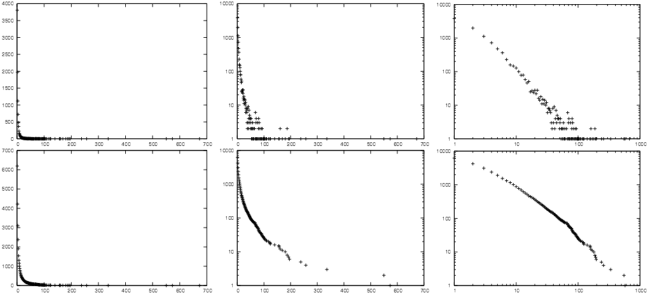
Common data distributions



Be careful: many tools have assumptions on the underlying distribution!

Source: *T. Viard*

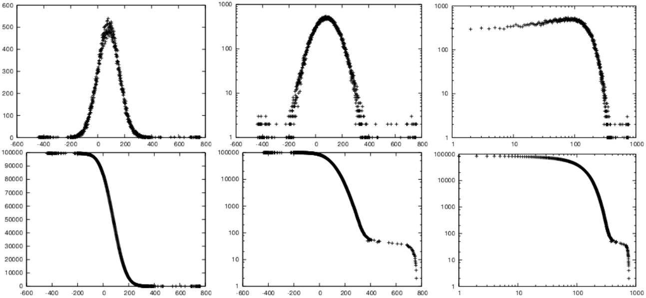
Common data distributions



Be careful: many tools have assumptions on the underlying distribution!

Source: *T. Viard*

Common data distributions

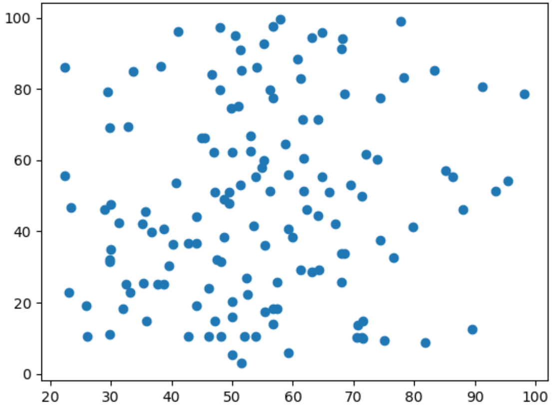


Be careful: many tools have assumptions on the underlying distribution!

Source: *T. Viard*

Always plot your data

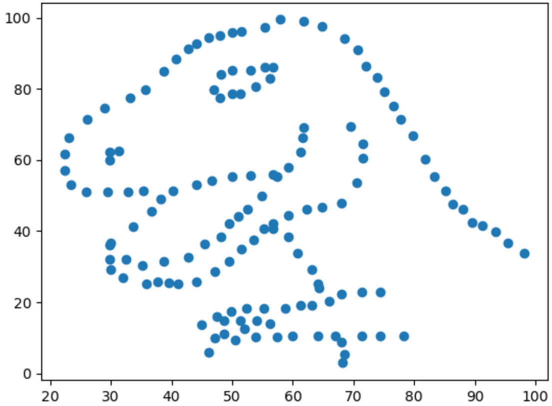
Dataset: $|X| = 142$ points, $\mu_1(X) = 54.22(16.76)$, $\mu_2(X) = 47.83(26.93)$



Source: *The Datasaurus dataset, A. Cairo, 2016*

Always plot your data

Dataset: $|X| = 142$ points, $\mu_1(X) = 54.22(16.76)$, $\mu_2(X) = 47.83(26.93)$



Source: *The Datasaurus dataset, A. Cairo, 2016*

What is machine learning?

Perform a **task** with a clear (=formalized) objective

Supervised vs **unsupervised**

- (U) Find subgroups of interest
- (U) Detect anomalies
- (S) Predict a price, the weather, autocomplete text
- (S) Classify documents into an ontology

Splitting your dataset

Splitting is a way to use all your data for a machine learning task

Three different splits:

- Train: to understand the data;
- Validation: to see if you're doing well;
- Test: to see how you truly generalize

Is chronological order important? Is categorical representativity important? Which properties will you be breaking?

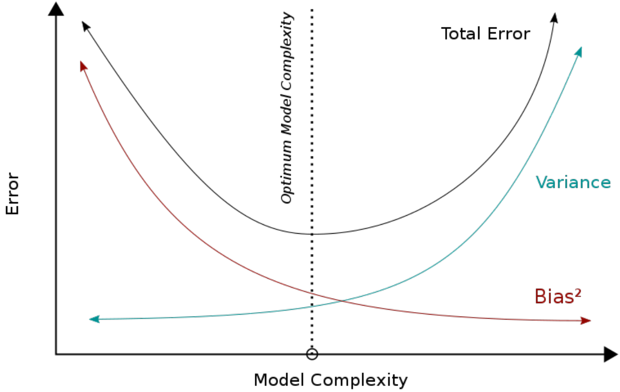
Cross-validation: swap the splits and apply your algorithm

Evaluate performance on average (please report deviations!)

Is my validation dataset good?

Underfitting and overfitting

The **bias-variance** trade off: how to learn well, but also generalize?



Source: *Wikipedia*

The many shapes of bias

Social vs **statistical**, **Model** vs **data**

- "The data isn't good enough!"
- Good average performance $\neq$ good local performance
- Data can be noisy, statistically biased
- But also socially biased
- Generalization vs stochastic parrotting
- What is lost when fitting?
- How do we adapt to error?