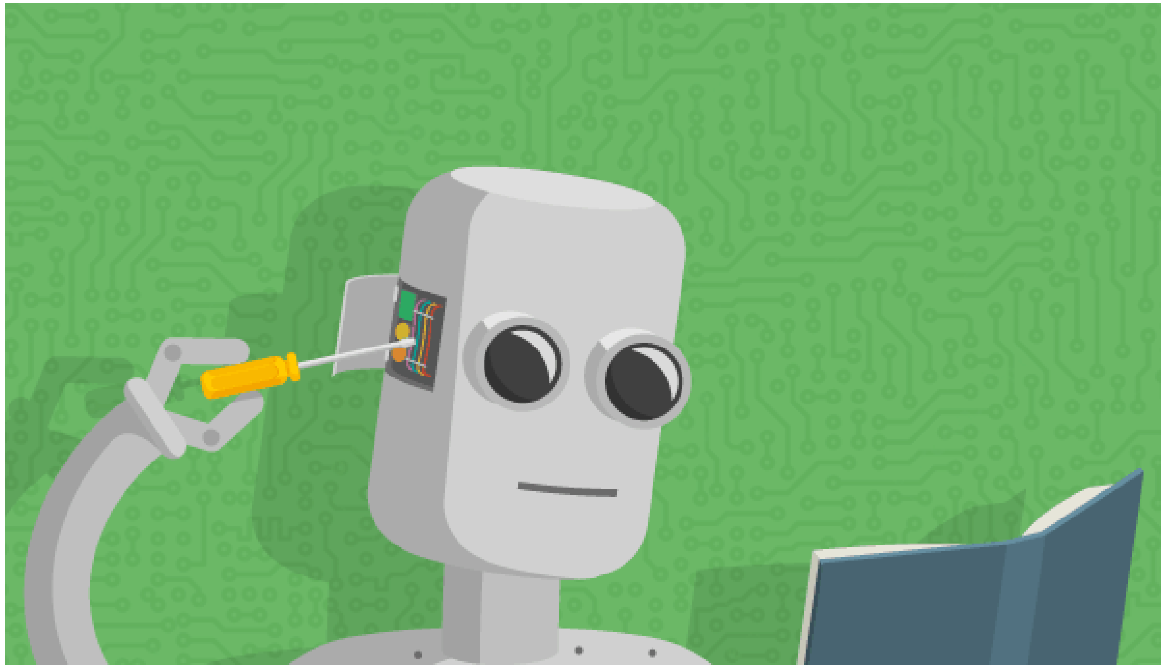
(Short) Introduction to Machine Learning

Alberto TONDA, Senior Researcher (DR)

UMR 518 MIA-PS (Applied Mathematics and Computer Science)
INRAE, AgroParisTech, Université Paris-Saclay
Institut des Systèmes Complexes, Paris-Ile-de-France

Outline

- What is machine learning
- ML as optimization
- Supervised ML
- Overfitting
- Unsupervised ML
- Issues



Machine learning (proper definition)

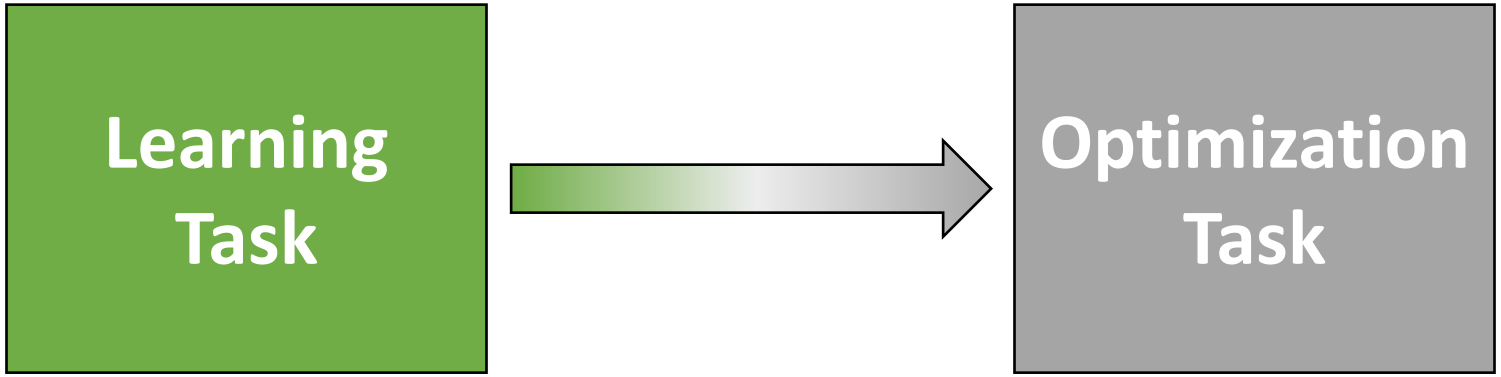
*Given a class of tasks **T**,*
*a performance measure **P**, and experience **E**,*
a machine learning algorithm improves its
*performance measured with **P**, for tasks in **T**,*
*using the experience **E***

Machine learning (proper definition)

*Given a class of tasks **T**,*
*a performance measure **P**, and experience **E**,*
*a machine learning algorithm improves its performance measured with **P**, for tasks in **T**,*
*using the experience **E***

Machine learning as optimization

- Learn a task directly from examples
  - No need for symbols, just large quantities of data
  - *Samples* (rows) and *features* (columns)
- "Dirty secret" of ML: it's mostly optimization
  - Restate **learning task** as **optimization task**
  - Solve it relying on available (training) data



Machine learning as optimization

- What does “restating” the problem mean?
  - Variables to optimize: **parameters** of the model
  - Create an **objective function** related to your learning problem
  - **Optimizing** the objective function also solves your problem
- Types of machine learning
  - **Supervised**: we have labeled data (correct answers, ground truth)
    - Common tasks: classification, regression
  - **Unsupervised**: we do not have labeled data
    - Common tasks: clustering, dimensionality reduction; also the base of advanced techniques, such as image/text generation

Vocabulary

- **Model/predictor**: one candidate solution (regressor/classifier)
- **Model parameters**
  - Values (numerical, categorical, …) *inside* the model
  - Optimized (e.g. change values) during training process
- **Samples**: rows of the dataset
- **Features**: columns of the dataset
- **Training data**: data from which we want to learn
- **Test data**: unseen data, kept aside to assess *generalization*
- **Validation data**: used during training, not for training (!)
- **Training/Fit**: optimize parameter values to fit training data

Vocabulary

- **Model hyperparameters**
  - Choices/parameters *outside* the model
  - Usually user-defined *before* training process starts
- **Capacity** (loose definition)
  - Maximum order of function that can be approximated by model
  - The more parameters, the more capacity
- **Bias**: source of errors, not enough capacity (underfitting)
- **Variance**: sensitivity to small variations in training data, too much capacity (overfitting)

Supervised machine learning: Brainstorming

- How would you construct our objective function for ML?

Supervised machine learning

- Regression
  - **Minimize** (squared/abs) difference predictions - training data
  - Way of optimizing depends on the structure of the model
  - After optimization, **R2/MSE** is usually used as a metric of quality

Features: y, x_1, x_2, x_3

Model: $\hat{y} = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_3$

Optimization task: $\mathrm{argmin}(\sum_{i=0}^{N} | \hat{y}(i) - y(i) |)$

Supervised machine learning

- Regression
  - **Minimize** (squared/abs) difference predictions - training data
  - Way of optimizing depends on the structure of the model
  - After optimization, **R2/MSE** is usually used as a metric of quality

Features: $y, \boldsymbol{x}$

Model (parameters): $\hat{y} = f(\boldsymbol{x}, \theta)$

Optimization task: $\text{argmin}(\sum_{i=0}^{N} | \hat{y}(i) - y(i) |)$

Supervised machine learning

- Regression
  - **Minimize** (squared/abs) difference predictions - training data
  - Way of optimizing depends on the structure of the model
  - After optimization, **R2/MSE** is usually used as a metric of quality

Features: $y, \boldsymbol{x}$

Sometimes called "**problem variables**" in ML

Model (parameters): $\hat{y} = f(\boldsymbol{x}, \theta)$

Optimization task: $\mathrm{argmin}(\sum_{i=0}^{N} |\, \hat{y}(i) - y(i) \,|)$

From an optimization point of view, these are **variables**!

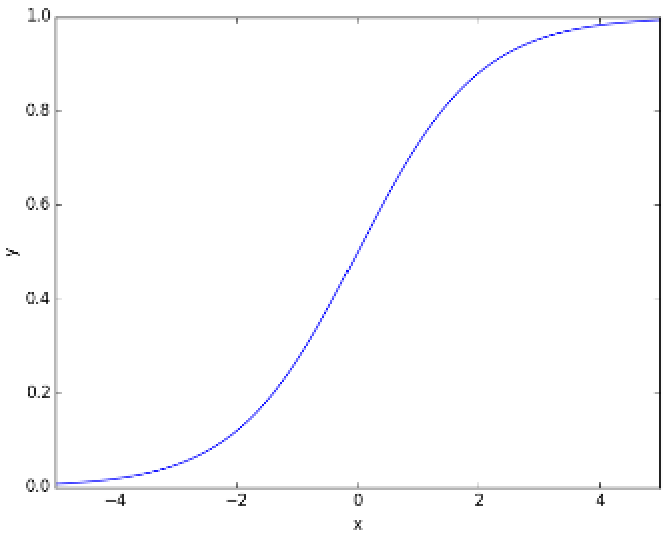
Supervised machine learning

- Classification
  - During training, sometimes treated as continuous optimization
  - E.g. interpret continuous output as *probability* (…) of class

Features: y, x_1, x_2, x_3

Model: $f(X) = \beta_0 + x_1\beta_1 + \cdots + x_n\beta_n$

$$\hat{y} = \frac{1}{1 - e^{-f(X)}}$$



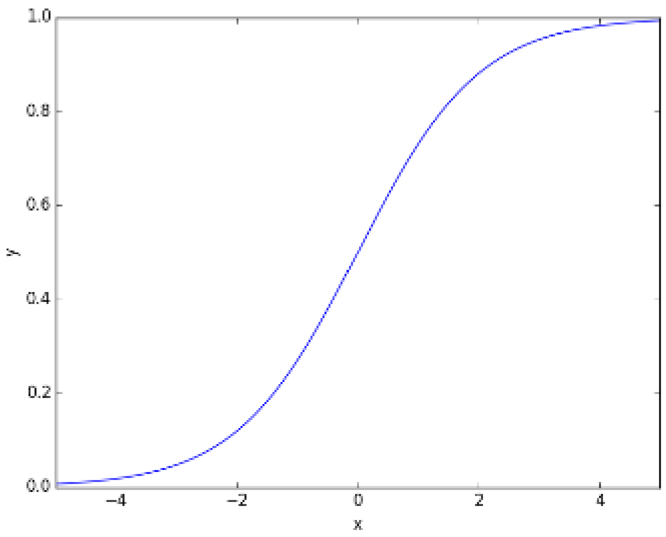
Supervised machine learning

- Classification
  - Evaluating a trained model: **accuracy,** % of correct answers
  - **F1**, **Matthew's Correlation Coefficient**, **AUC ROC** are better

Features: y, x_1, x_2, x_3

Model: $f(X) = \beta_0 + x_1\beta_1 + \cdots + x_n\beta_n$

$$\hat{y} = \frac{1}{1 - e^{-f(X)}}$$



Overfitting and regularization

- ML model has been trained on data
  - It fits the training data really well
  - It DOES NOT generalize for *unseen data*
  - The trained model captures unique properties of the training data…
  - …that **only exist for those data samples**

- How can we **evaluate overfitting**?

Image generated by AI, prompt "The concept of overfitting in machine learning as the final boss monster in a videogame"

Overfitting and regularization

- Hide part of the available data, use it only for test
- Ok, but we could be just lucky! We can do better
- **k-fold cross-validation** (k=5 or 10)
  - Divide data into k parts (splits)
  - Iterate k times
  - Each time, use k-1 splits for training
  - One split for testing
  - Obtain an **average** and a **stdev** of performance
- Large stdev usually indicates issues

Overfitting and regularization

- Optimizing for maximal fitting is not enough
  - Also need to add penalties for overfitting
  - But how?

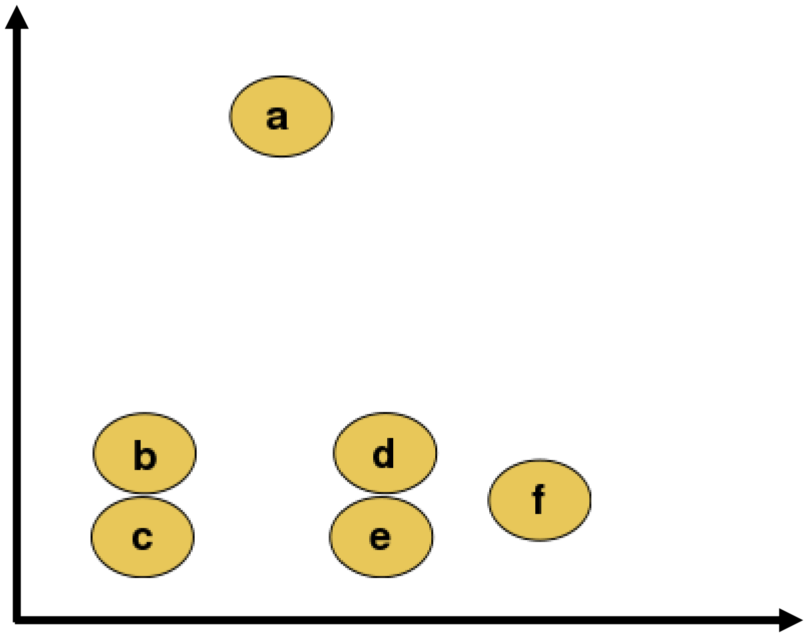
- Penalize values **correlated** to overfitting
  - In Genetic Programming, tree size
  - Linear Models, coefficient values
  - Artificial Neural Networks, weight values

Unsupervised ML: Brainstorming

- There is **no ground truth**, no labels; what can we optimize?

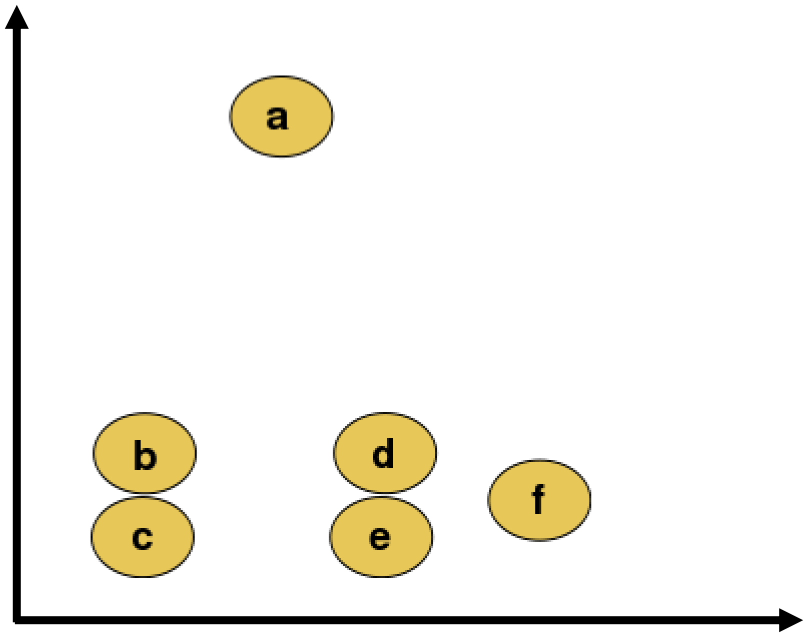
Clustering

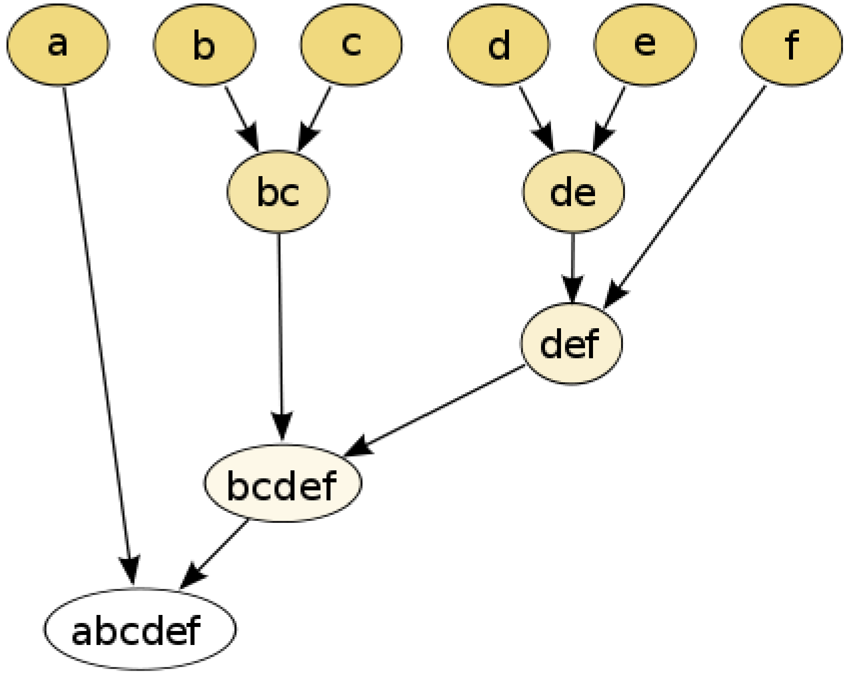
- Group together points (samples) in feature space
  - On the basis of their (Euclidean) distance (or other measure)
  - Show the user different groups (dendrogram), ask them to pick



Clustering

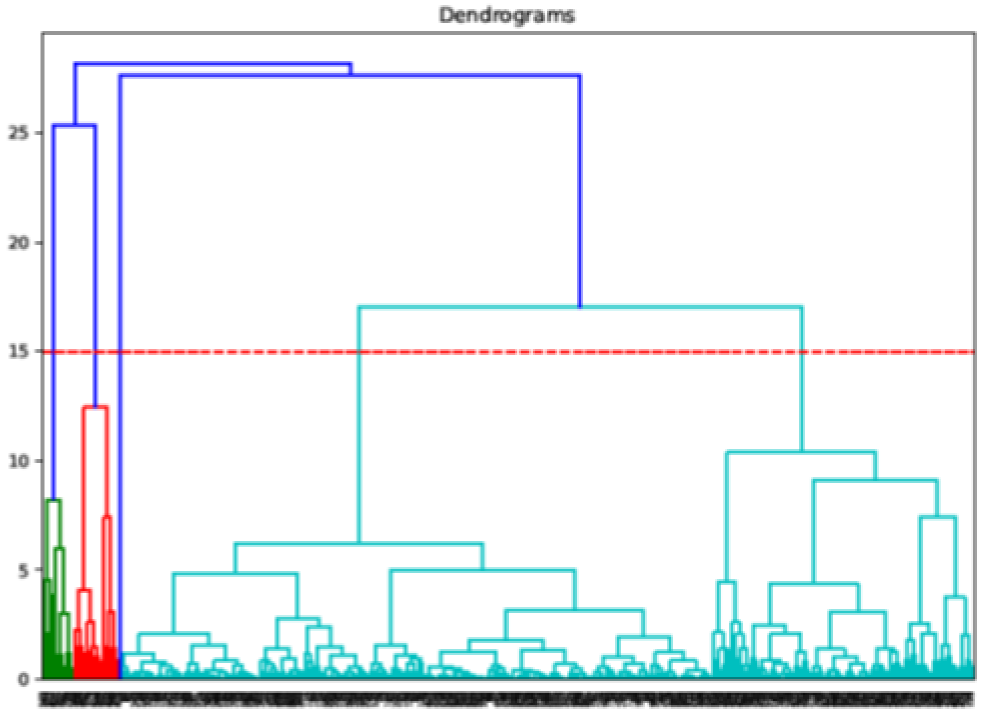
- Group together points (samples) in feature space
  - On the basis of their (Euclidean) distance (or other measure)
  - Show the user different groups (dendrogram), ask them to pick





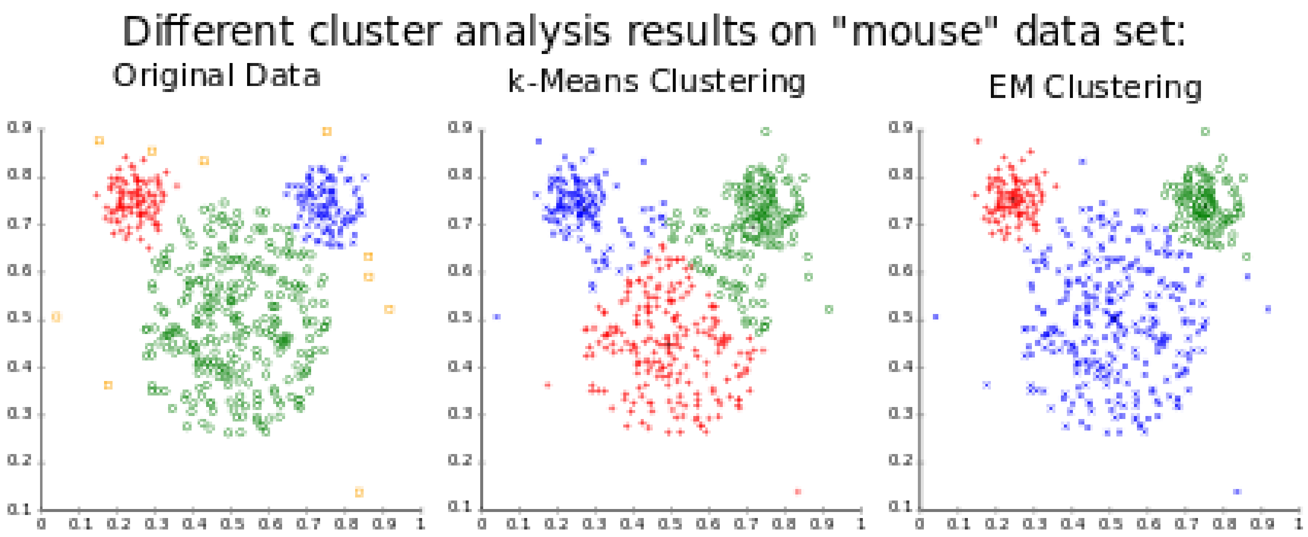
Clustering

- Group together points (samples) in feature space
  - On the basis of their (Euclidean) distance (or other measure)
  - Show the user different groups (dendrogram), ask them to pick



Clustering

- Other solutions: min inter-cluster distance, max intra-cluster
- State-of-the-art algorithms
  - Hierarchical agglomerative clustering
  - Density-Based Spatial Clustering of Applications with Noise (DBSCAN)
  - Hierarchical DBSCAN
- **Visualize results!!!**
- However, datasets are high-dimensional!



Dimensionality reduction

- Principal component analysis
  - Transform a higher-dimensional feature space in lower dimension
  - Each new dimension explains a part of the original variance
  - New dimensions are weighted sums of the original features
  - Can be framed as a maximization problem

$$\arg\max_{\|\mathbf{w}\|=1} \left\{ \sum_i \left(\mathbf{x}_{(i)} \cdot \mathbf{w}\right)^2 \right\} = \arg\max_{\|\mathbf{w}\|=1} \left\{ \|\mathbf{X}\mathbf{w}\|^2 \right\}$$

$$\hat{\mathbf{X}}_k = \mathbf{X} - \sum_{s=1}^{k-1} \mathbf{X}\mathbf{w}_{(s)} \mathbf{w}_{(s)}^{\mathsf{T}}$$

- Other technique: t-SNE (t-distributed Stochastic Neighbor Embedding)

Embeddings

- Create vector space, distances/positions have *meaning*
  - **Automatically**, starting from data
  - We don't know exactly **how the space should be**

- Optimization algorithms used vary depending on case study
- Often heuristics, but sometimes gradient descent on NNs
- Example: Word2Vec

Word2Vec

- ML always had issues with *language*
  - ML generally works well with continuous values, or sortables
  - Words are discrete, and their sequence matters
  - There is a **syntax**, but also a **semantic**
  - In general, this was more the domain of Symbolic AI
  - But few things worked! Until…
- Word2Vec is an unsupervised algorithm
  - Turns words into points in a (high-dimensional) vector space
  - Distances and displacements have meaning!

Word2Vec

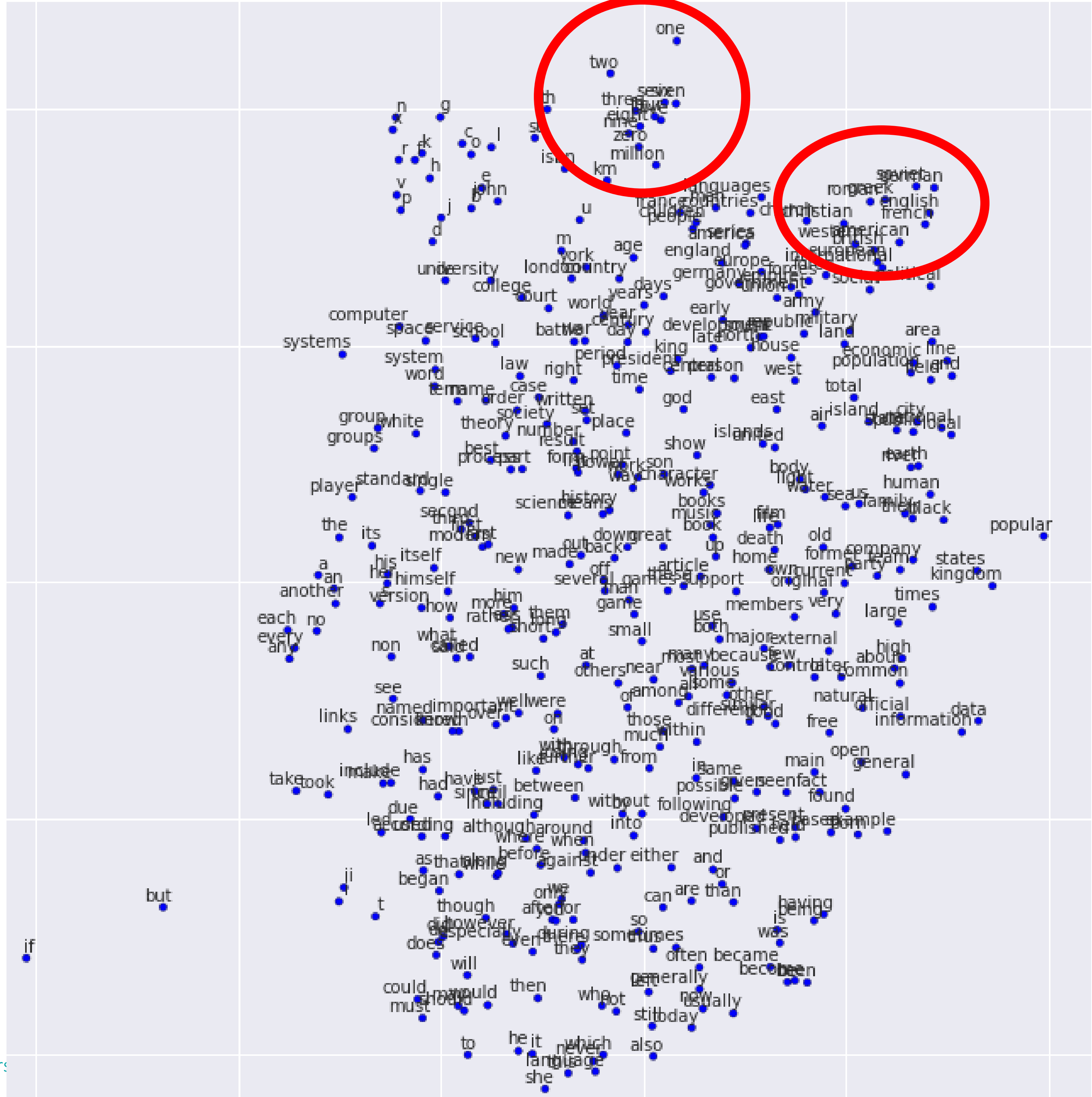
- **Input**: a considerable amount of text
- **Output**: vector space, each word corresponds to a point
- Slides a window over the text
- Reduces distance between middle word and adjacent ones
- Slides the window by one word, iterates

This is a sentence that the algorithm is analyzing…

Reduce distance between points corresponding to “sentence” and “a”, “sentence” and “is”, …

Word2Vec

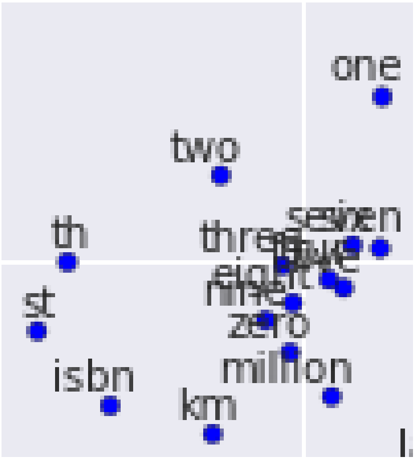
- Vector space



Word2Vec

- “French”, “British”, “American”…
  - Adjectives for nationality!
  - Nearby, you have “languages”, “countries”
  - Also, “England”, “Europe”, “International”, …

Word2Vec



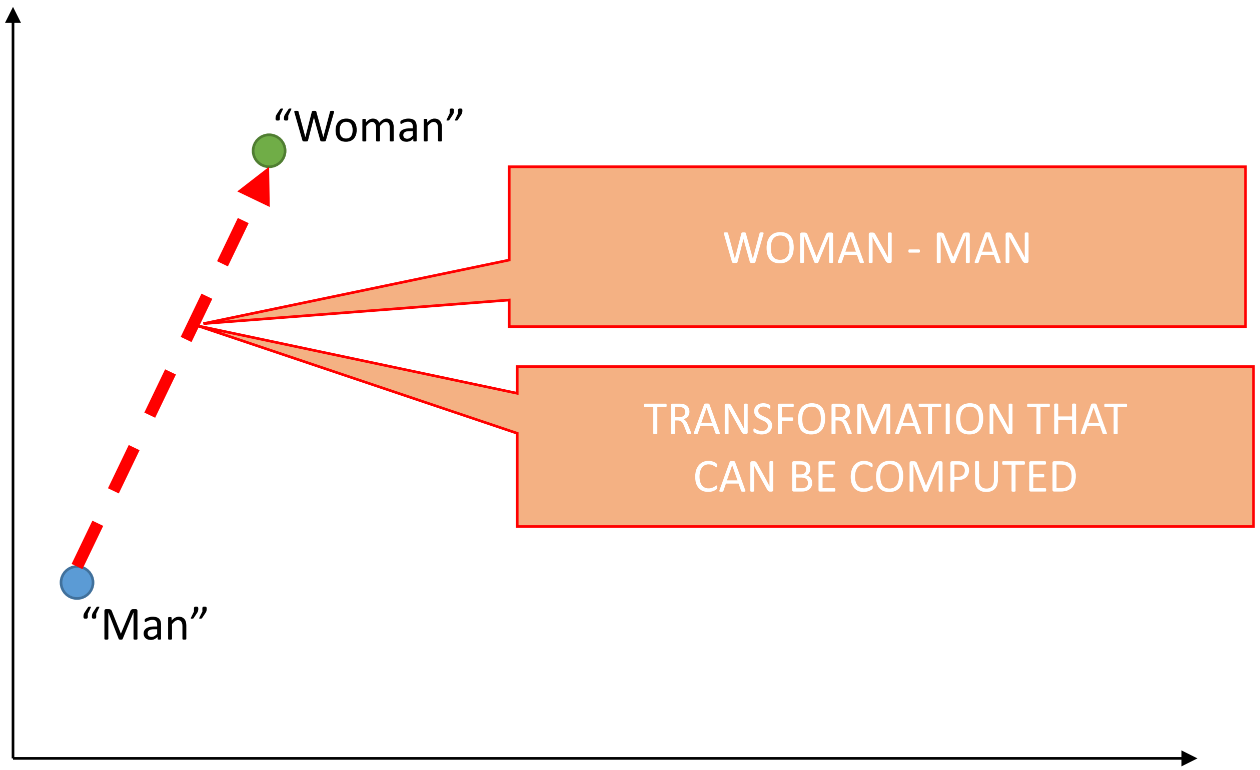
- "one", "two", "zero", "seven", "million"…
  - Numbers, quantities
  - Nearby, you have some units of measurement
  - Also "th", and "st", as in 9-th, 1-st
  - ISBN (guess usually appears nearby numbers!)

Word2Vec

- What is happening here?
  - Algorithm has **no semantic** info (**no meaning**)
  - But words with **similar meaning** are **close**
- Just by looking at the position of words in text
  - Words with similar *use* appear in same positions w.r.t. other words
  - Word2Vec captures *some* aspects of meaning
- Can we do something else with Word2Vec?

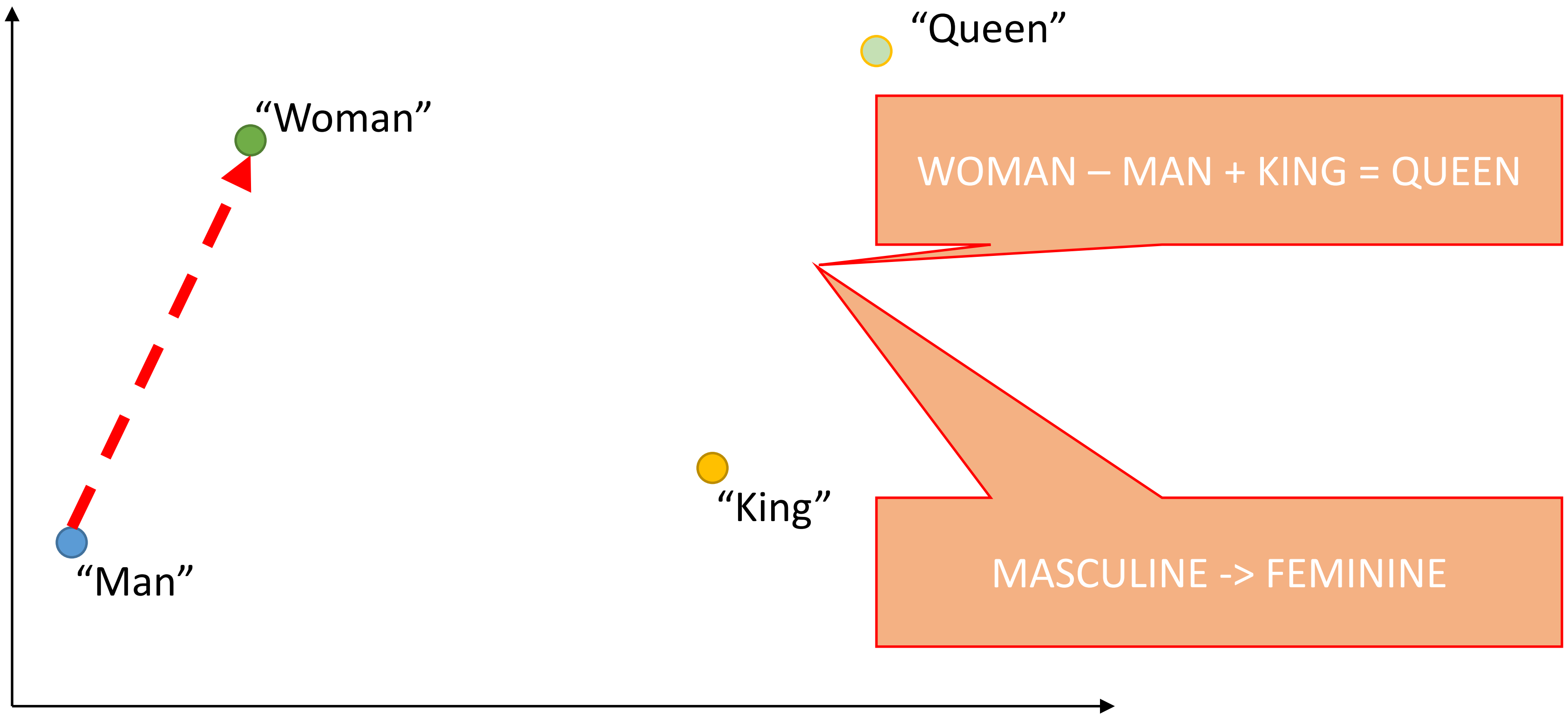
Word2Vec

- Can we do something else with Word2Vec?

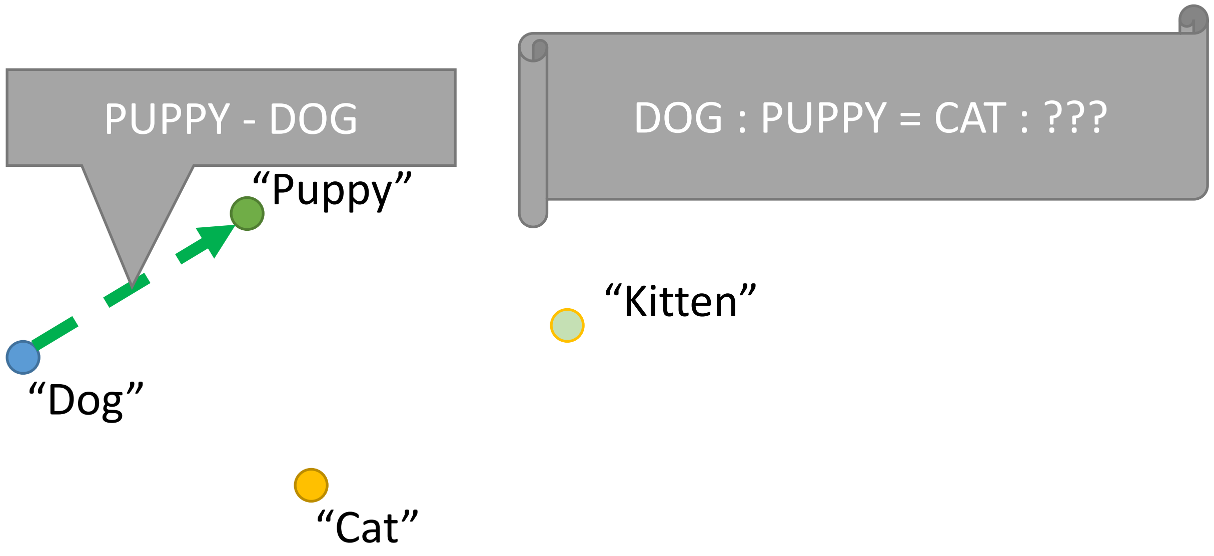


Word2Vec

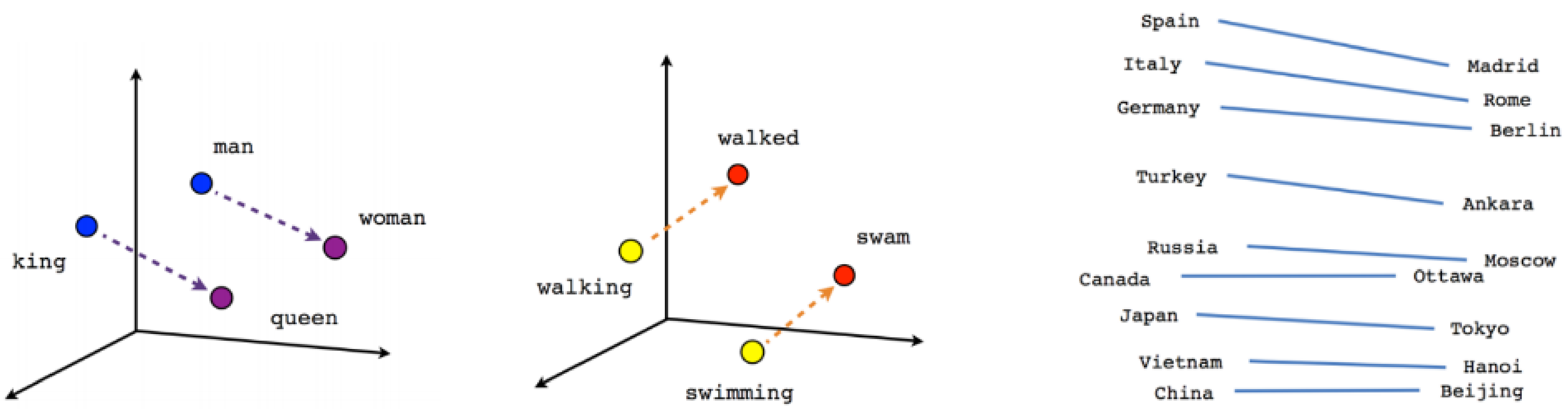
- Can we do something else with Word2Vec?



Word2Vec



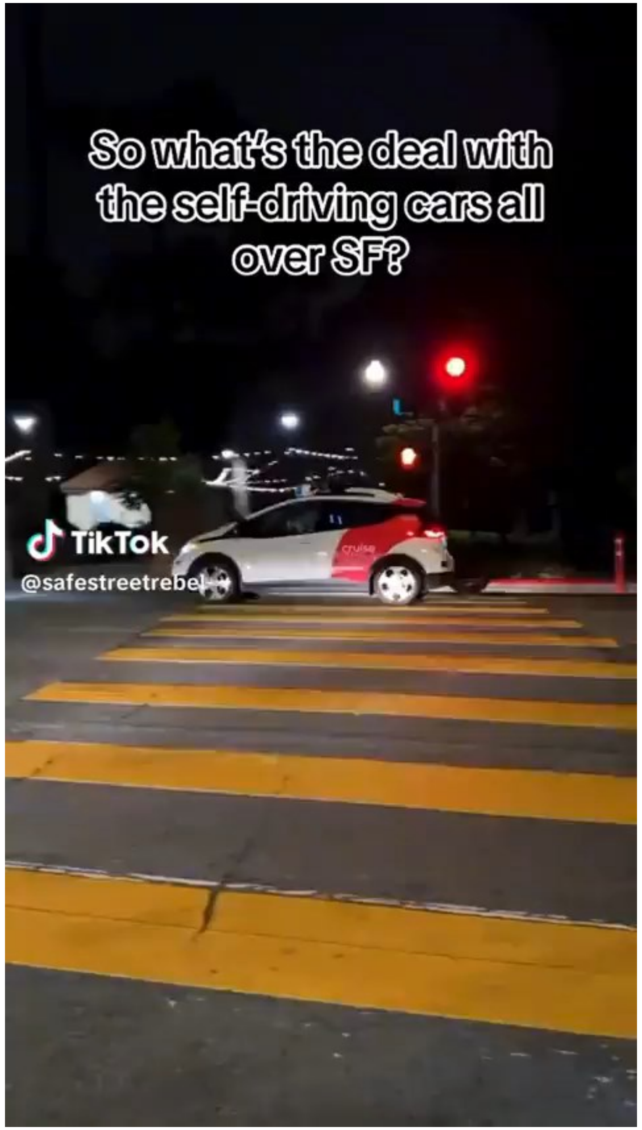
Word2Vec



Male-Female Verb tense Country-Capital

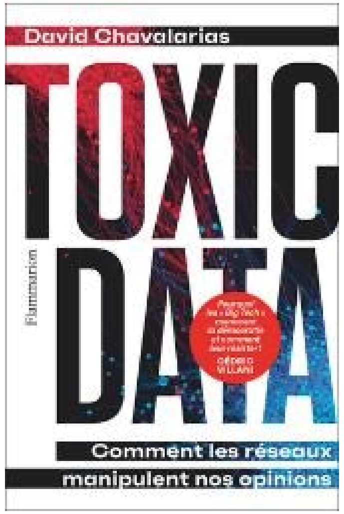
Practical issues with machine learning

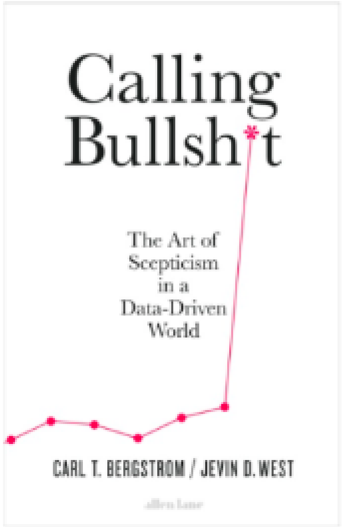
- Algorithms find *correlations,* not causal relationships
- Black-box effect
- Fragility (combination of inputs that unexpectedly produce undesired results)
- All this for large, high-capacity models

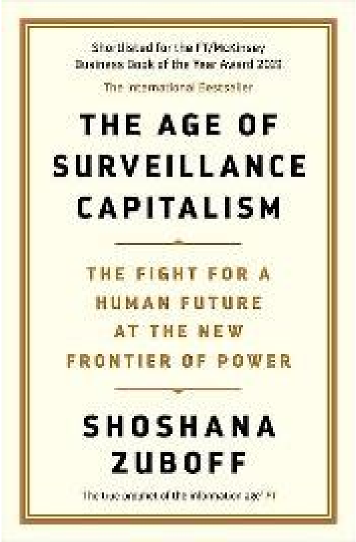


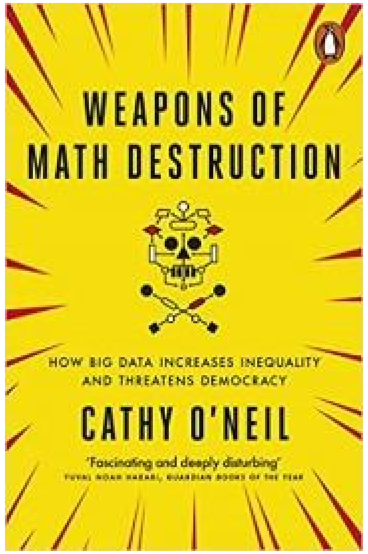
Societal issues with machine learning

- Ethical implications
- Bias in the data
- Black-box effect
- Unintended consequences
- Further reading (divulgation)
  - https://callingbullshit.org/
  - Weapons of Math Destruction
  - The Age of Surveillance Capitalism
  - Toxic Data









Practical advice (April 2025)

- Supervised ML
  - Try **all the algorithms** with default values, pick the best
  - Or run "AutoML" approaches (next set of slides)
- If you do not have the time to do that
  - For images: a convolutional neural network (CNN)
  - For text: transformer-based NNs (BERT, Llama or Word2Vec/Doc2Vec)
  - For tabular data: XGBoost, LightGBM, or Random Forest
  - For time series: …nothing really works better than other systems

Questions?

Bibliography

- James et al., *An Introduction to Statistical Learning with Applications in Python*, 2023

Images and videos: unless otherwise stated, I stole them from the Internet. I hope they are not copyrighted, or that their use falls under the Fair Use clause, and if not, I am sorry. Please don't sue me.