POLITECHNIKA POZNAŃSKA

Poznan University of Technololgy

Introduction to Artificial Intelligence

Theory and algorithms

Dariusz Brzezinski

Agenda

- AI and ML definitions
- Basic machine learning algorithms
  - Linear and logistic regression
  - K-nearest neighbors
  - K-Means
  - Decision trees
  - Naive Bayes
  - Neural Networks
- Explainability

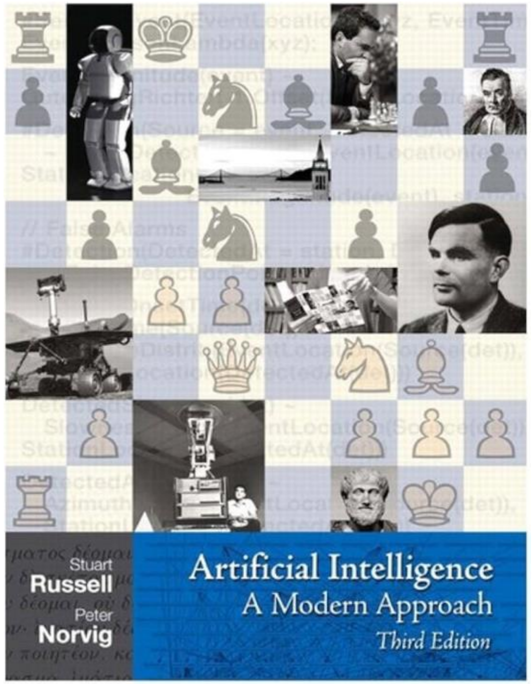
What is Artificial Intelligence (AI)?

- *The science and engineering of making **intelligent machines** (McCarthy)*
- *The theory and development of computer **systems able to perform tasks normally requiring human intelligence**, such as visual perception, speech recognition, decision-making, and translation between languages (Oxford dictionary)*
- *Field of science dealing with **solving non-algorithmizable problems** (Duch)*

Which intelligence?

- Human intelligence is a broad term encompassing
  - associations, metaphors, and analogies
  - common sense
  - conceptual frameworks
  - creativity
  - empathy
  - …
- Some research promote the terms **Machine Intelligence** and **Computational Intelligence**

Definition of AI

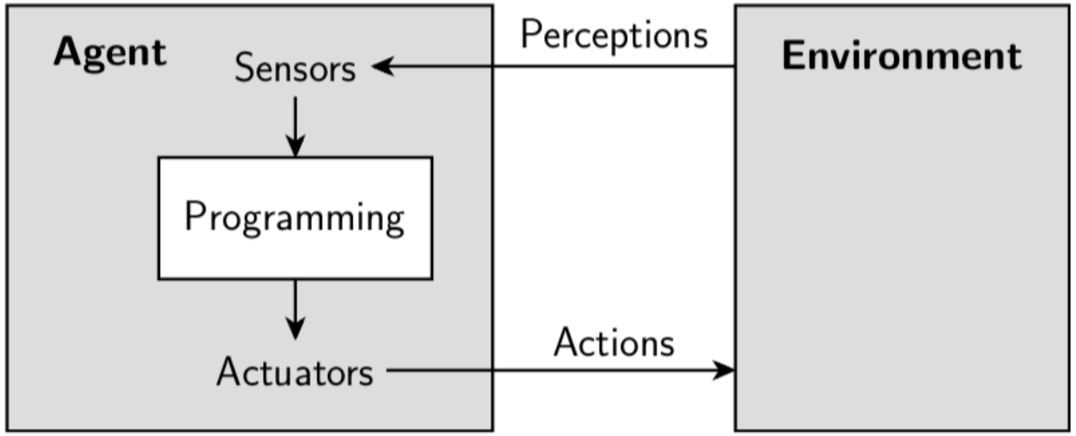


- Human intelligence is a broad term encompassing
  - associations, metaphors, and analogies
  - common sense
  - conceptual frameworks
  - creativity
  - empathy
  - …
- Some research promote the terms **Machine Intelligence** and **Computational Intelligence**
- Russell and Norvig agree that AI must be defined in terms of *acting* and not *thinking*. They promote the concept of an **intelligent agent**.

Intelligent behavior

- Act rationally to optimize some goals
- Apply knowledge to solve current situation
- Handle complex problems
- Manage missing or uncertain data
- Process and manipulate symbols
- Learn from experiences and surroundings
- Adapt to new situations and tasks

Intelligent behavior



An agent **perceives** its environment through sensors and **acts** on the environment through actuators.

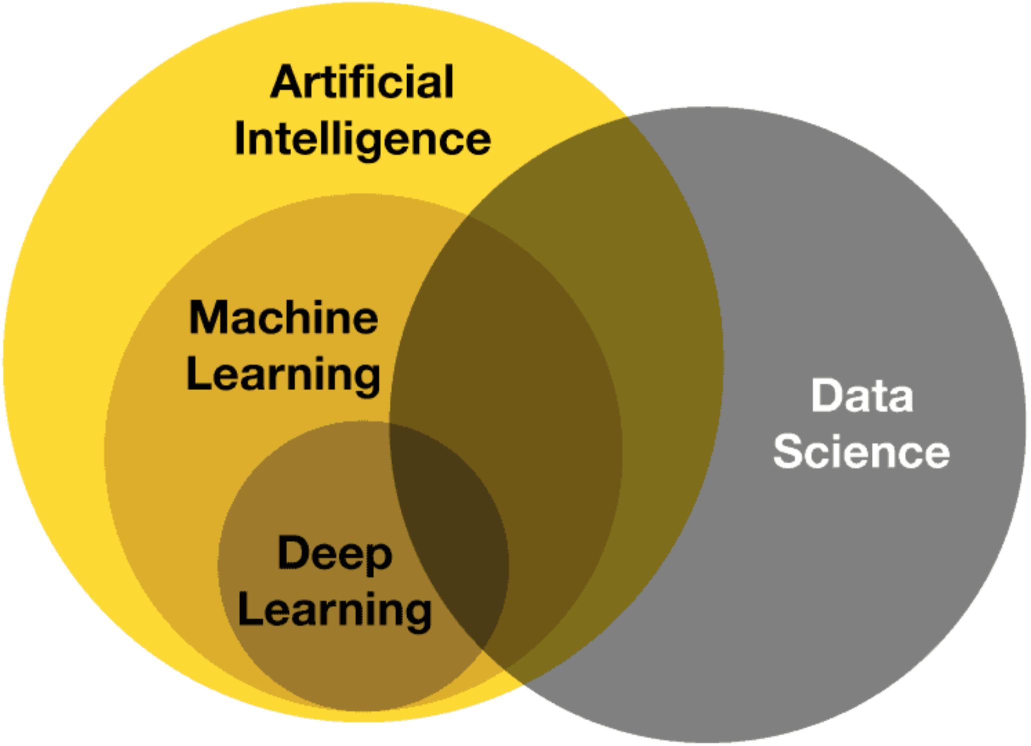
	Human	**Robot**
Sensors	Eyes, ears, nose, …	Cameras, mics, lidars, …
Actuators	Hands, legs, mouth, …	Grippers, motors, speaker, …

Different levels of AI

- (Narrow) Artificial Intelligence
  - Intelligent systems for specialized tasks (e.g., speech recognition, image recognition, sentiment analysis, predicting loan risk, planning robot moves)
  - Rational agents that work without figuring out how human reasoning works
- Artificial General Intelligence
  - broadly intelligent, context-aware machines solving general problems
- Superintelligence
  - would outperform humans at nearly every thinking task

Back to our terminological confusion

- Artificial intelligence (AI, AGI)
- **Machine learning** (ML)
- Deep learning (DL)
- Data mining
- Knowledge discovery
- Data science
- Big data
- Statistical data analysis



Machine learning

***Machine learning** is a subfield of artificial intelligence dedicated to algorithms that improve themselves and make predictions based on data. The term is often used interchangeably with artificial intelligence.*

Various applications:

- Internet search engines
- Recommendation systems
- Face recognition
- Voice recognition
- Spam filtering
- Credit risk assessment

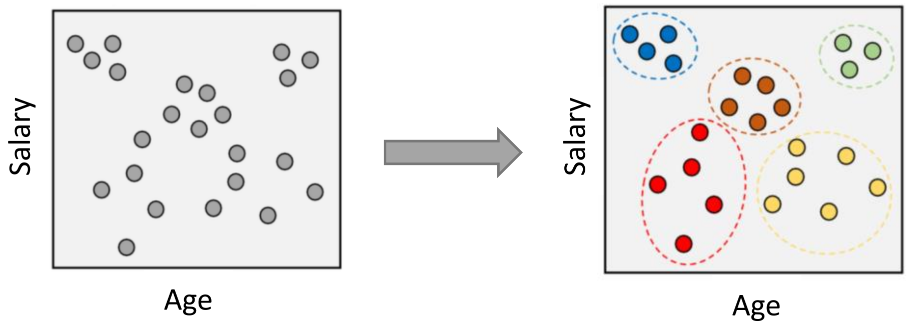
When to use machine learning?

1. **There is a pattern in the studied problem**
2. **We cannot model the pattern mathematically***
3. **We have data on the problem**

Yaser Abu-Mostafa
Learning from Data

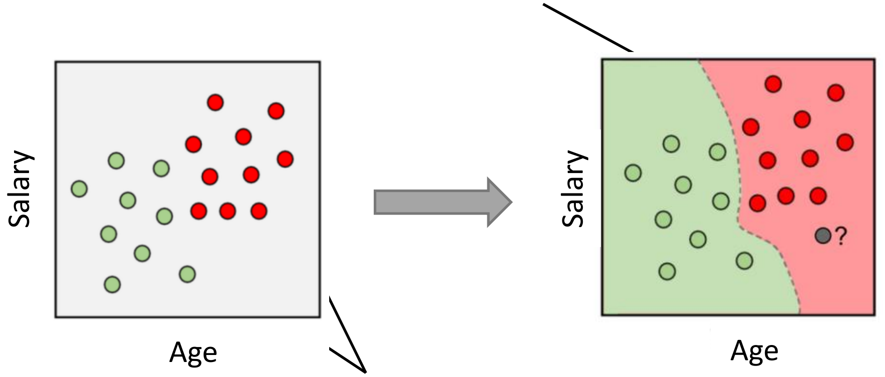
Machine learning

- **Unsupervised** (without a teacher)
  - Clustering
  - Association mining



- **Supervised** (with a teacher)
  - Classification
  - Regression
  - Reinforcement learning

prediction model

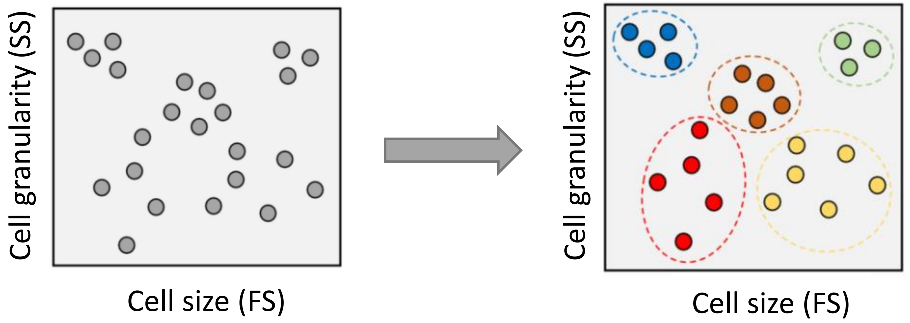


labeled training data

new data

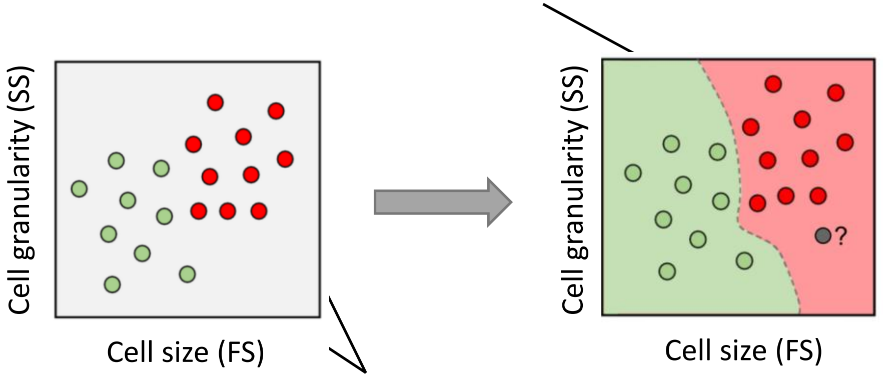
Machine learning

- **Unsupervised** (without a teacher)
  - Clustering
  - Association mining



- **Supervised** (with a teacher)
  - Classification
  - Regression
  - Reinforcement learning

prediction model



labeled training data

new data

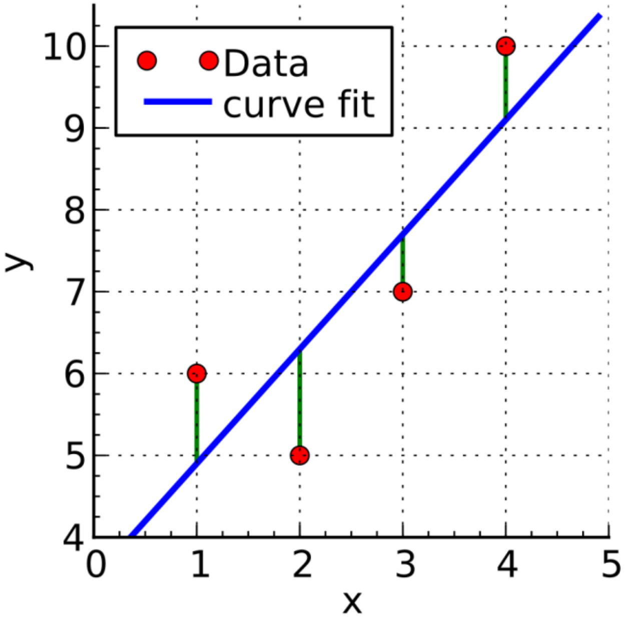
But how do you create a prediction model?

Prediction model = ML Algorithm + Hyperparameters + Data

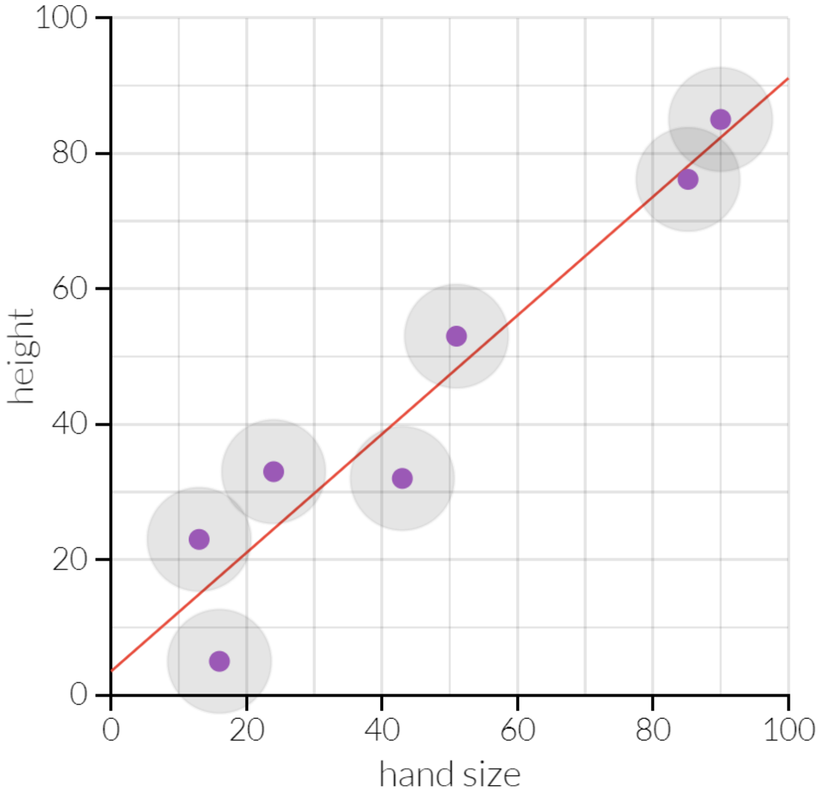
- **Algorithms discover patterns** in data and generalize tchem
- Algorithms usually have **hyperparameters** that **have to be tuned**
- **You always need** some form of **data**
- Many approaches:
  - Probabilistic models (bayesian methods)
  - Distance-based learning (nearest neighbors)
  - Symbolic learning (trees, rules)
  - Function approximation (regression, neural nets)

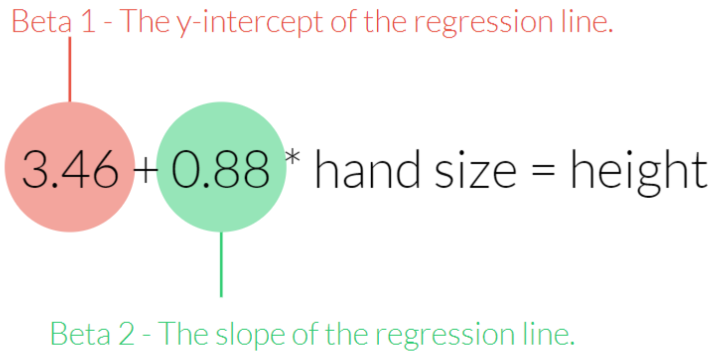
Linear regression

- Simple model for predicting continuous values
- General idea: Fit a line to the data to minimize average error
- What kind of error?
    - Typically mean squared error (L2, OLS)
    - Could be mean absolute error (L1, quantile regression)
    - Huber loss (robust regression)
- Commonly used in medicine
- Interpretable results: $y = b_0 + b_1 x_1 + \ldots$



Linear regression





Beta 1 - The y-intercept of the regression line.

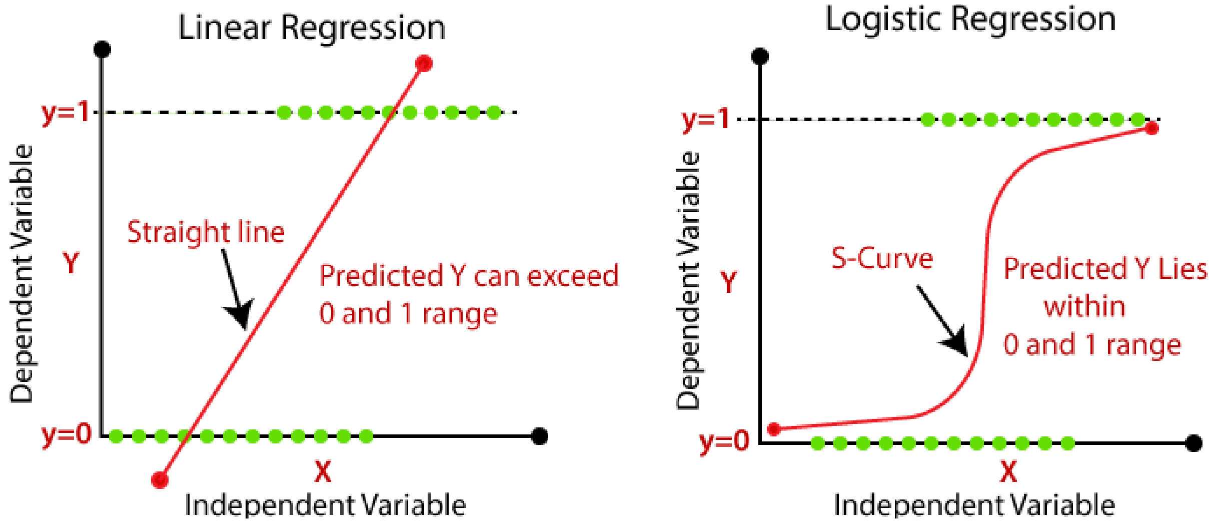
3.46 + 0.88 * hand size = height

Beta 2 - The slope of the regression line.

https://setosa.io/ev/ordinary-least-squares-regression/

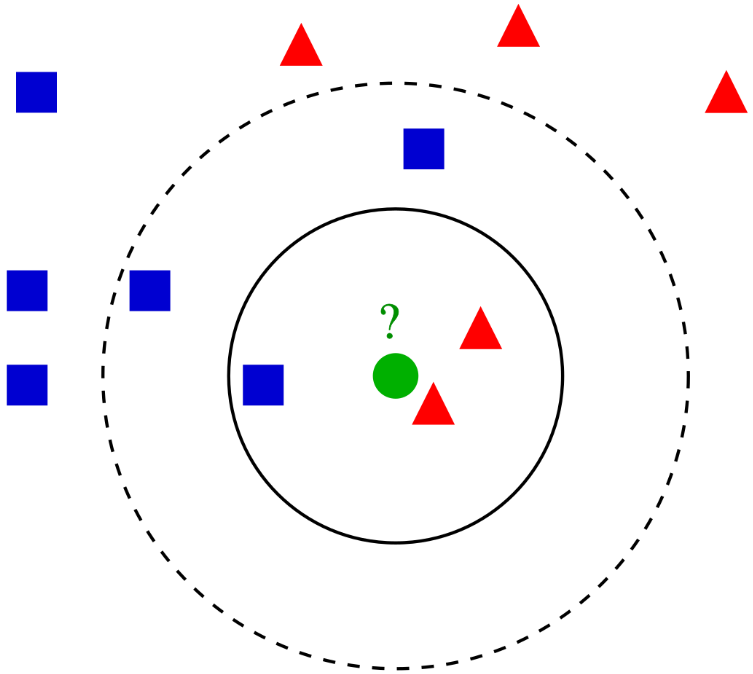
Logistic regression

- Linear model for predicting categorical values
- Sigmoid function instead of linear function
- Returns values from 0 to 1
- Interpretable model
- $\log(y/(1-y)) = b_0 + b_1 x_1 + \ldots$

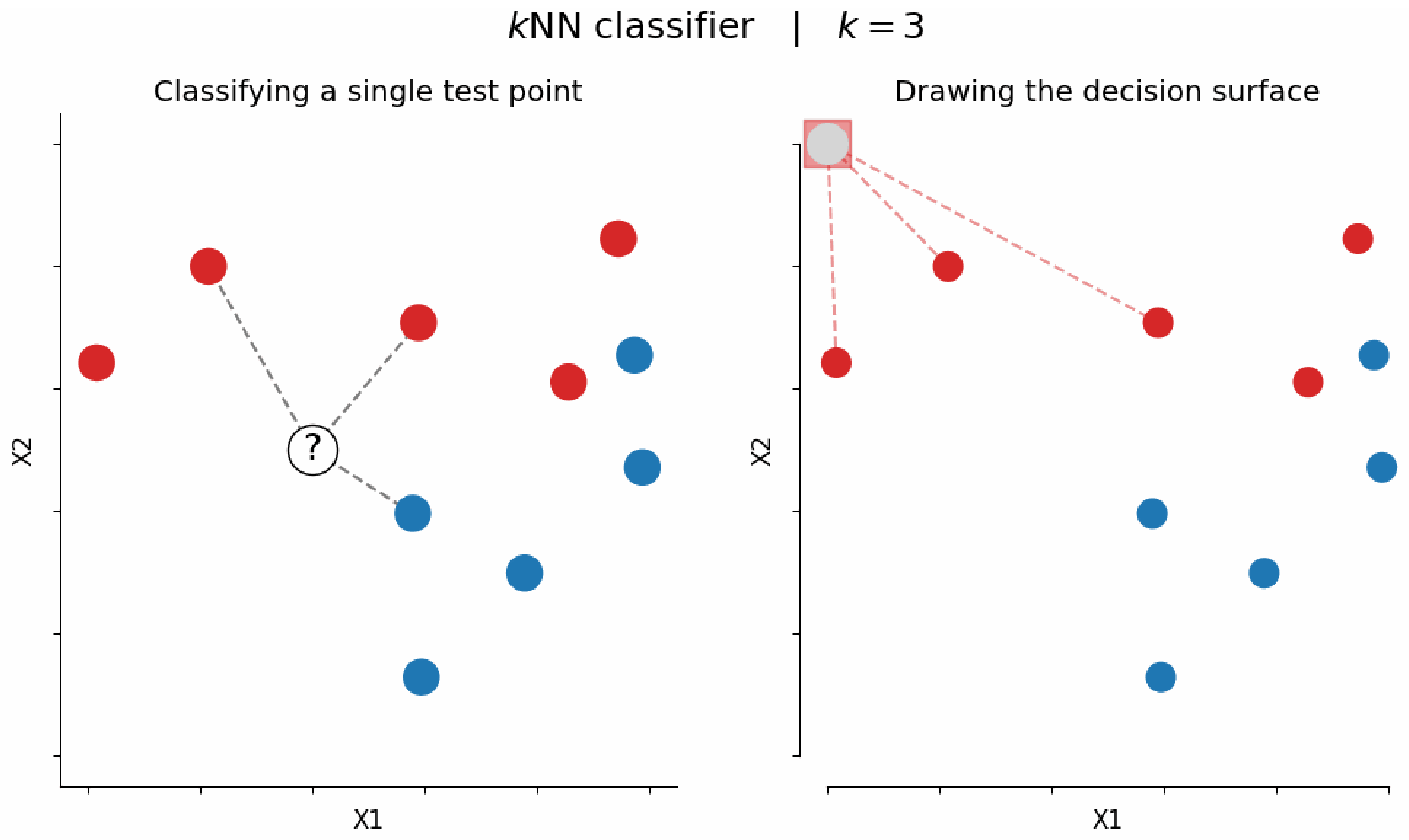


K-nearest neighbors

- Predictions based on similarity
- Base prediction on the most class common class of *k* nearest neighbors
- For regression average target values of *k* nearest neighbors
- Requires storing the training data
- *k* is a hyperparameter
- Needs a distance function
- All attributes must be on the same scale!

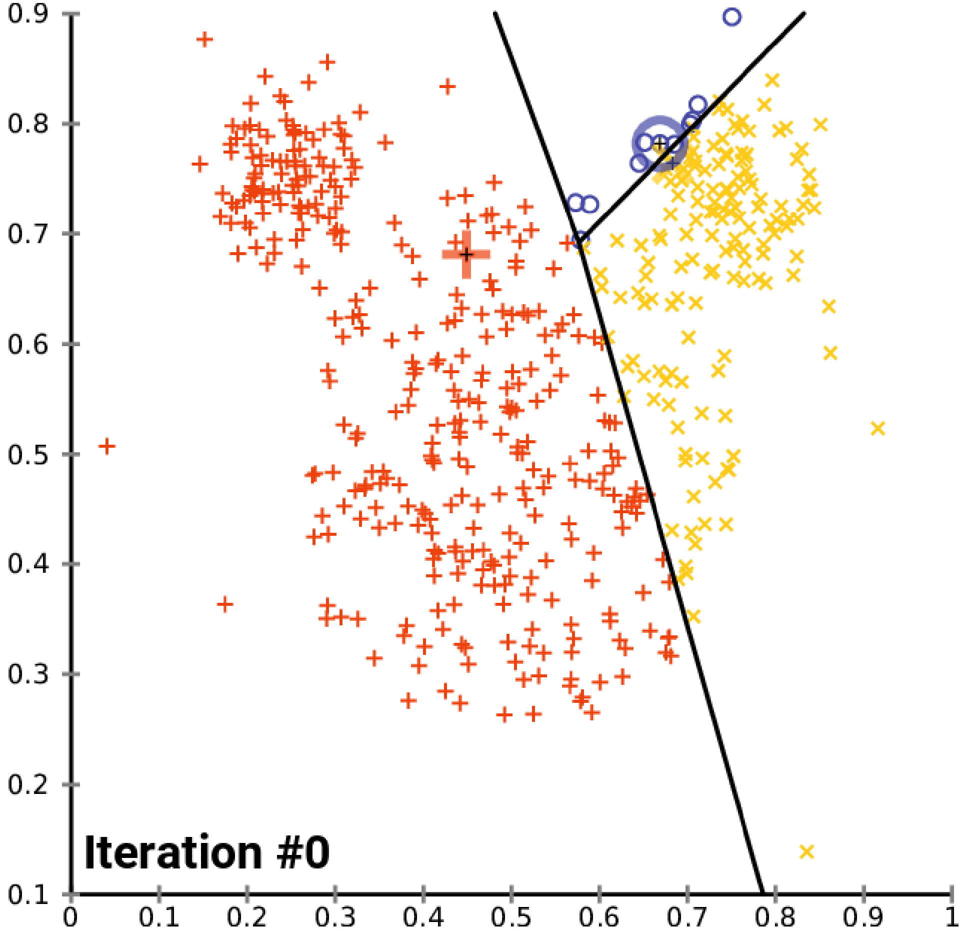


K-nearest neighbors



K-means

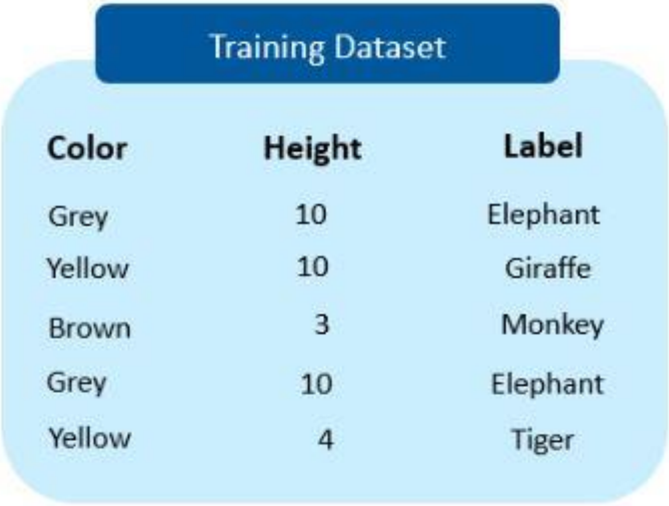
- Distance based approaches are also used for **clustering**
- Clustering = similar examples should be in the same cluster, dissimilar in different clusters
- K-means is an algorithm that finds *k* clusters that minimize the sum of squared witihin cluster distances
- All attributes must be on the same scale!



Decision trees

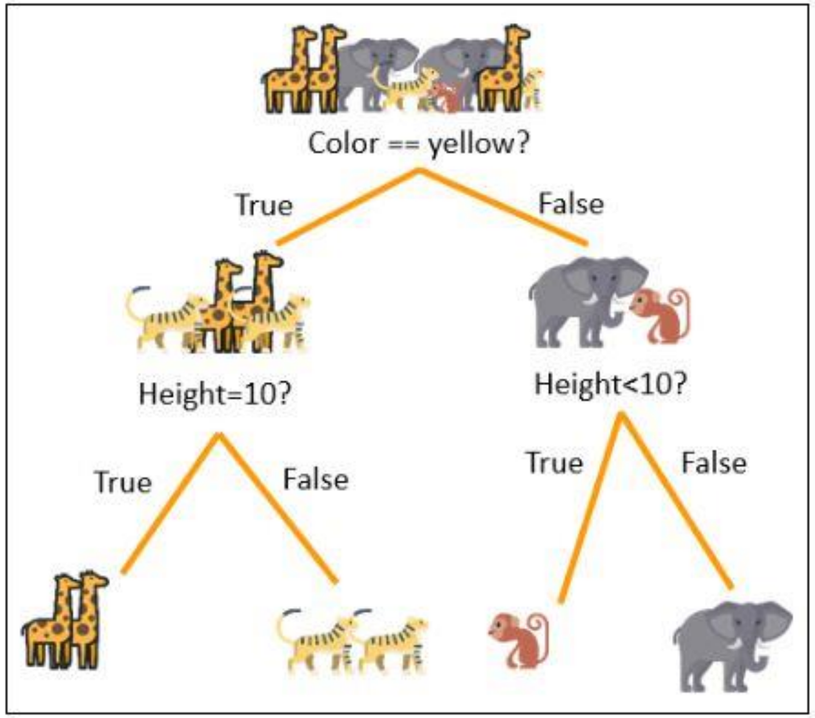
Representing knowledge as a tree:

- Each internal node tests an attribute
- Each branch corresponds to an attribute value
- Each leaf node assigns a classification



Training Dataset

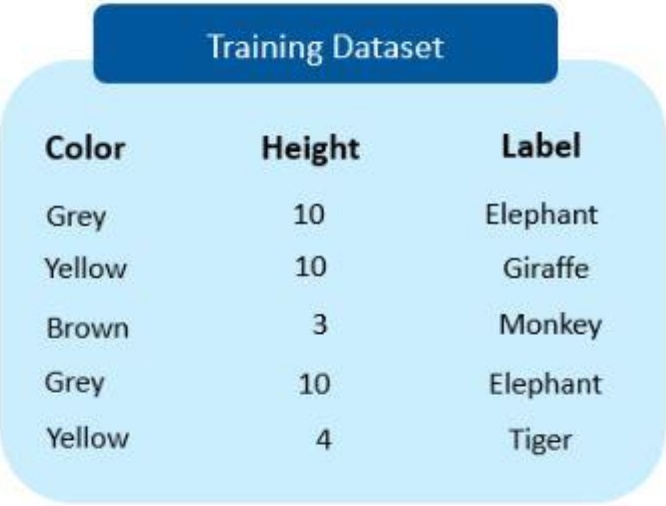
Color	Height	Label
Grey	10	Elephant
Yellow	10	Giraffe
Brown	3	Monkey
Grey	10	Elephant
Yellow	4	Tiger



Decision trees

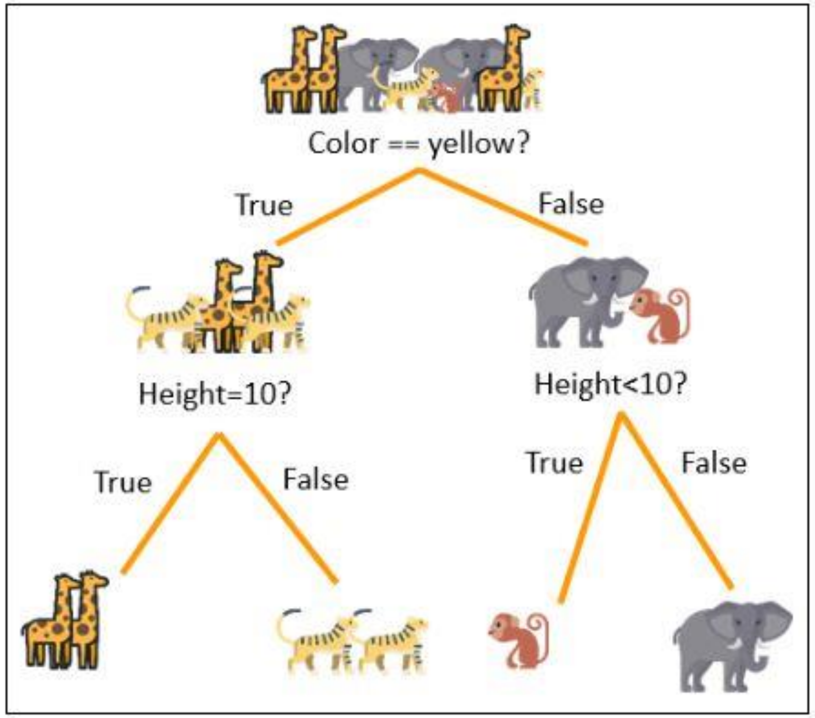
Representing knowledge as a tree:

- Each internal node tests an attribute
- Each branch corresponds to an attribute value
- Each leaf node assigns a classification



Training Dataset

Color	Height	Label
Grey	10	Elephant
Yellow	10	Giraffe
Brown	3	Monkey
Grey	10	Elephant
Yellow	4	Tiger



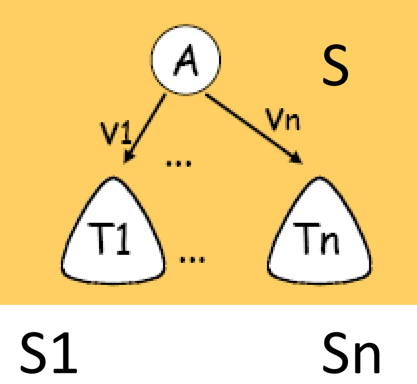
Decision trees

- Decision trees are created in (recursive) steps
- At start, all training examples S are at the root of the tree.
- **If** all examples from S belong to the same class Kj

  then label the node / leaf with Kj

  else

  - select the „best” attribute A
  - divide S into S_1, … , S_n according to values v_1, … , v_n of attribute A
  - Recursively build subtrees T_1, … , T_n for S_1, … , S_n
  - Stop if all nodes are labeled leaves



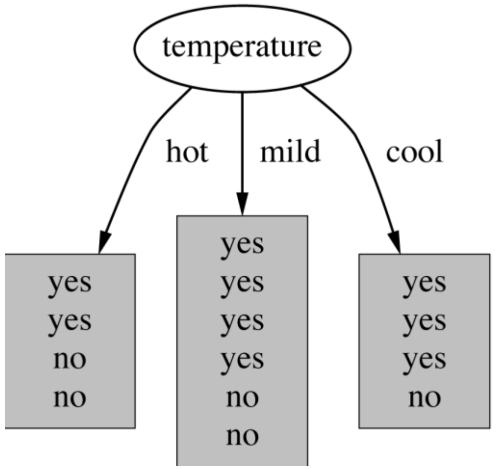
Decision trees

- Example dataset: Golf (Play or not?)

Outlook	Temperature	Humidity	Windy	Play?
sunny	hot	high	false	No
sunny	hot	high	true	No
overcast	hot	high	false	Yes
rain	mild	high	false	Yes
rain	cool	normal	false	Yes
rain	cool	normal	true	No
overcast	cool	normal	true	Yes
sunny	mild	high	false	No
sunny	cool	normal	false	Yes
rain	mild	normal	false	Yes
sunny	mild	normal	true	Yes
overcast	mild	high	true	Yes
overcast	hot	normal	false	Yes
rain	mild	high	true	No

Decision trees

- Example dataset: Golf (Play or not?)
- What constitutes a good split?

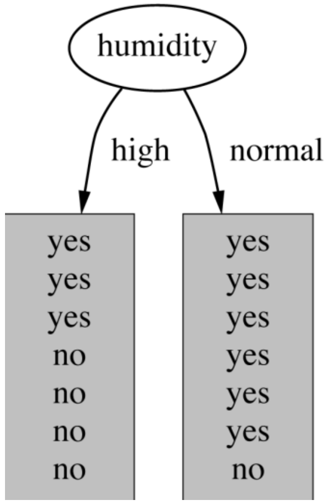


temperature

hot: yes, yes, no, no

mild: yes, yes, yes, yes, no, no

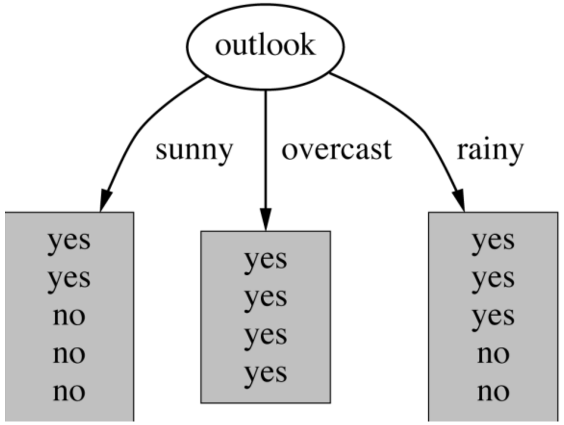
cool: yes, yes, yes, no



humidity

high: yes, yes, yes, no, no, no, no

normal: yes, yes, yes, yes, yes, yes, no

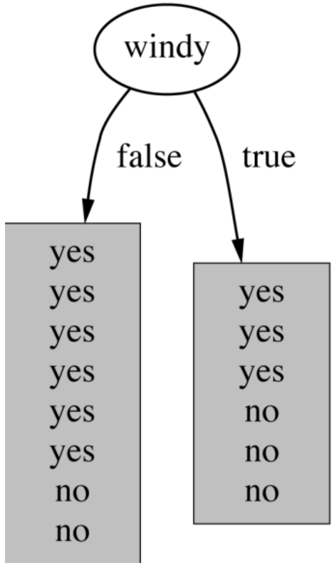


outlook

sunny: yes, yes, no, no, no

overcast: yes, yes, yes, yes

rainy: yes, yes, yes, no, no



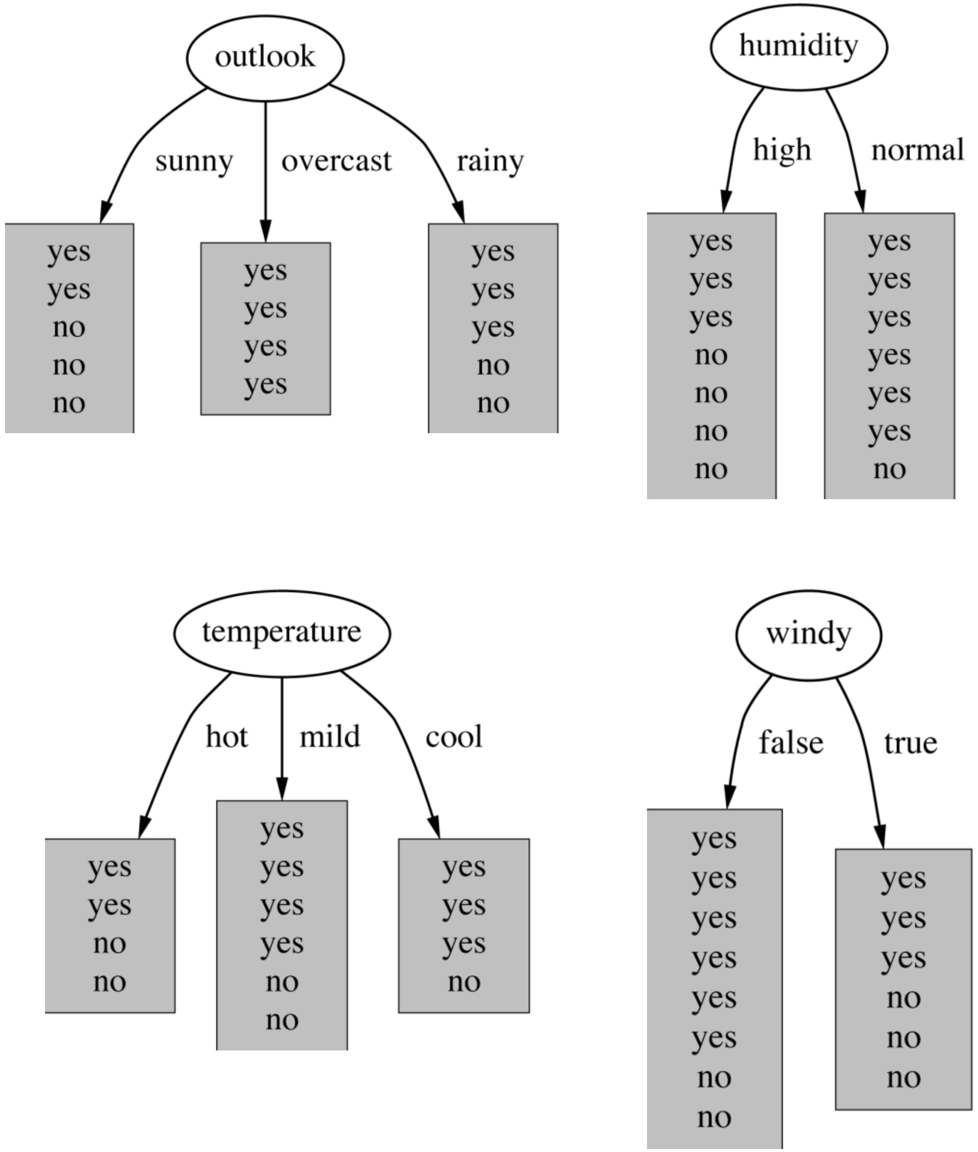
windy

false: yes, yes, yes, yes, yes, yes, no, no

true: yes, yes, yes, no, no, no

Decision trees

- Example dataset: Golf (Play or not?)
- What constitutes a good split?
- $\mathit{Entropy}(S) = -\sum_{i=1..n} p_i * \log_2 p_i$
- $\mathit{Entropy}(S,A) = \sum_{j=1..k} |S_j|/|S| * \mathit{Ent}(S_j)$
- $\mathit{InfoGain}(S,A) = \mathit{Ent}(S) - \mathit{Ent}(S|A)$
- $\mathit{InfoGainRatio}(S,A) = \mathit{InfoGain}(S,A) / \mathit{Ent}(A)$

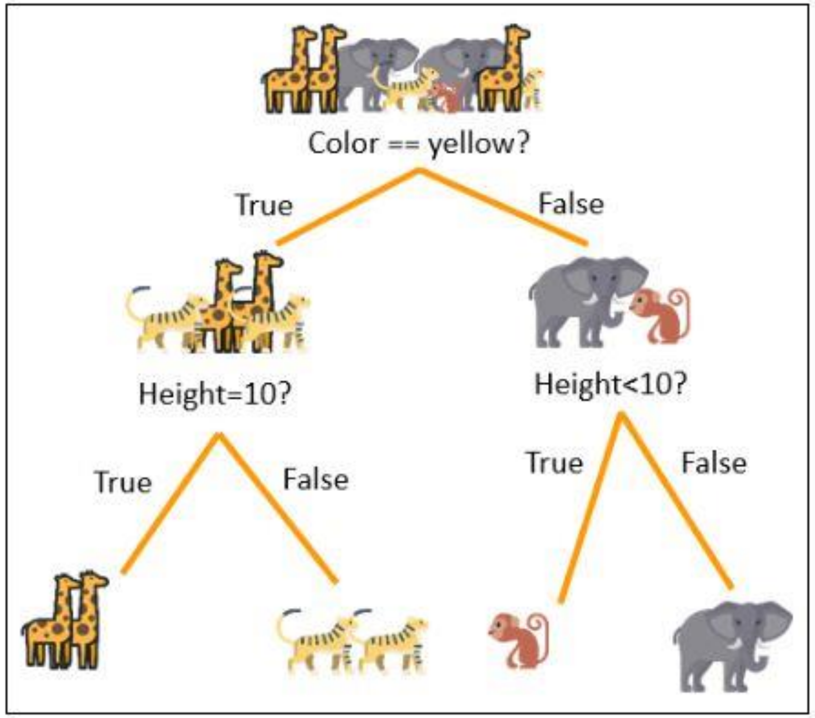


Rules

- Decision trees can be read as rules
- Decision rules:

  IF *Conditions* **THEN** *Class*

- One needs lists of rules to apply
- Dedicated algorithms in addition to tree interpretations
- Rules can also be created based on associations



Association rules

- Market basket analysis
- {Cereal, Milk} → Bread [sup=5%, conf=80%]
- *80% of customers who buy cereal and milk also buy bread and 5% of customers buy all these products together*
- Applications
    - Product placement in stores
    - Recommender systems
    - Rule discovery in life sciences

TID	Produce
1	MILK, BREAD, EGGS
2	BREAD, SUGAR
3	BREAD, CEREAL
4	MILK, BREAD, SUGAR
5	MILK, CEREAL
6	BREAD, CEREAL
7	MILK, CEREAL
8	MILK, BREAD, CEREAL, EGGS
9	MILK, BREAD, CEREAL

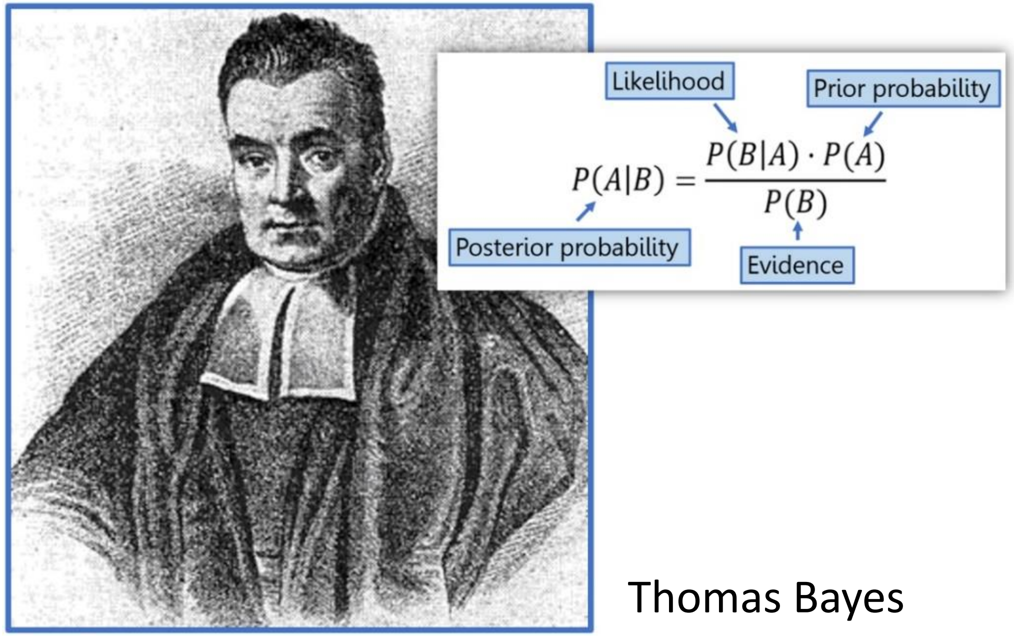
Bayesian theorem

- Given training data X, posteriori probability of a class C, P(C|X) follows the Bayes theorem

$$P(C|X) = \frac{P(X \wedge C)}{P(X)} = \frac{P(X|C)P(C)}{P(X)}$$

- Maximum posteriori (MAP):
  - Choose the most probable hypothesis under the given conditions

$$C_{MAP} = \arg\max_{C_i \in CD} P(C = C_i)P(X|C = C_i)$$



Likelihood

Prior probability

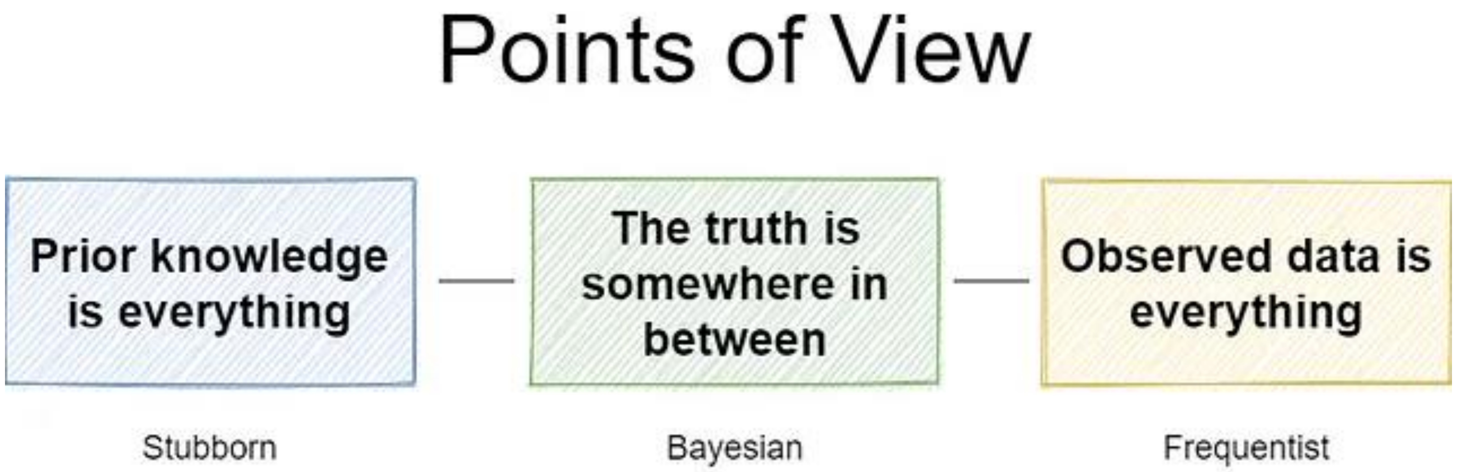
$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

Posterior probability

Evidence

Thomas Bayes

Bayesian models



Points of View

Prior knowledge is everything — The truth is somewhere in between — Observed data is everything

Stubborn

Bayesian

Frequentist

Naive Bayes classifier

- Assume conditional independence of attribute values

$$P(X|C = C_i) = \prod_{j=1}^{m} P(A_j = x_j|C = C_i)$$

where $X = (A_1 = x_1, A_2 = x_2, \dots, A_m = x_m)$

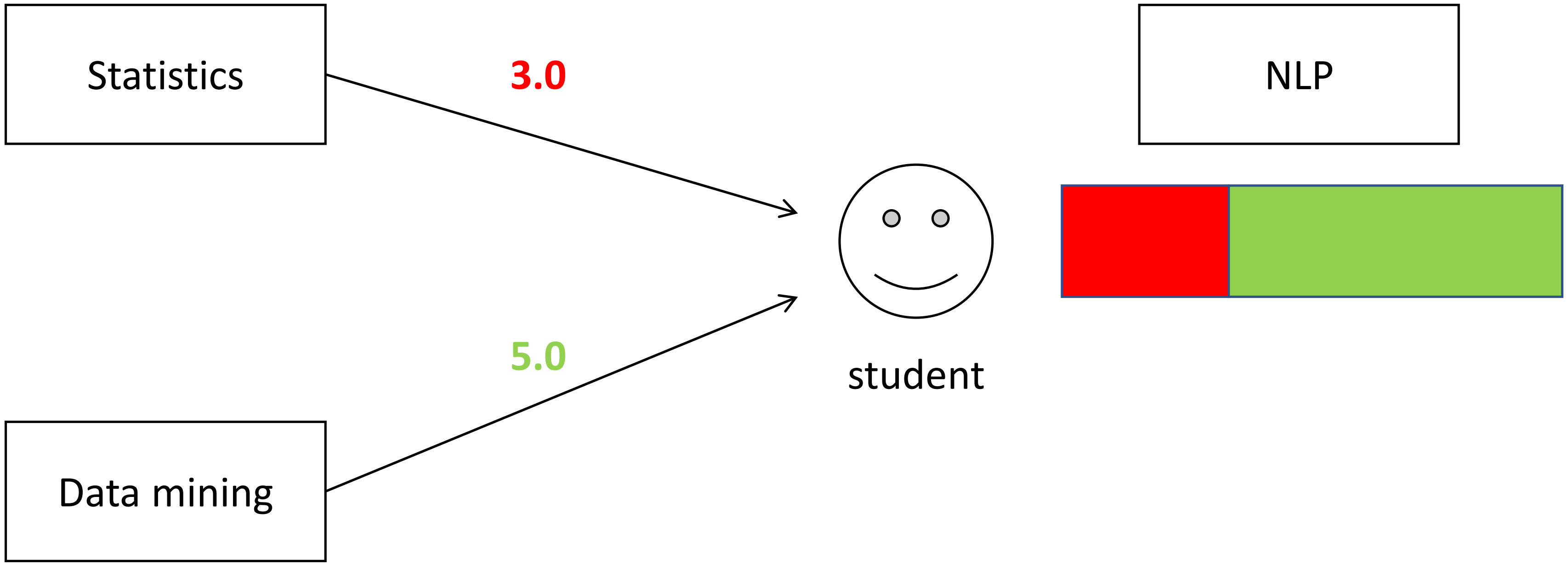
- Following the MAP approach we get

$$C_{MAP} = \arg\max_{C_i \in CD} P(C = C_i) \prod_{j=1}^{m} P(A_j = x_j|C = C_i)$$

- Now probabilities can be estimated by counts

$$P(C = C_i) = \frac{s_i}{N} \text{ and } P\left(A_j = x_j \middle| C = C_i\right) = \frac{s_{ij}}{s_i}$$

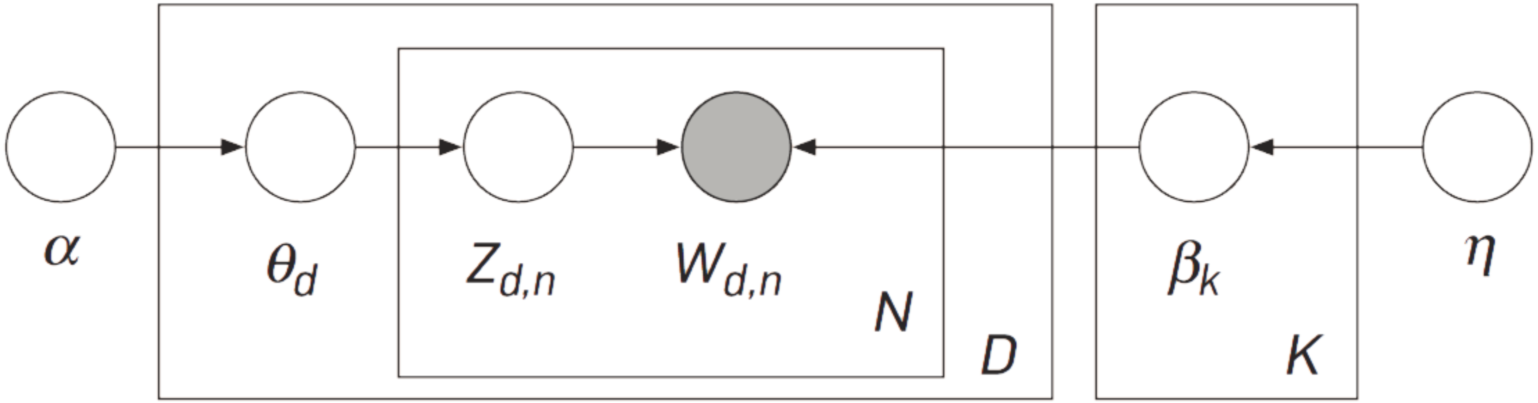
Bayesian networks



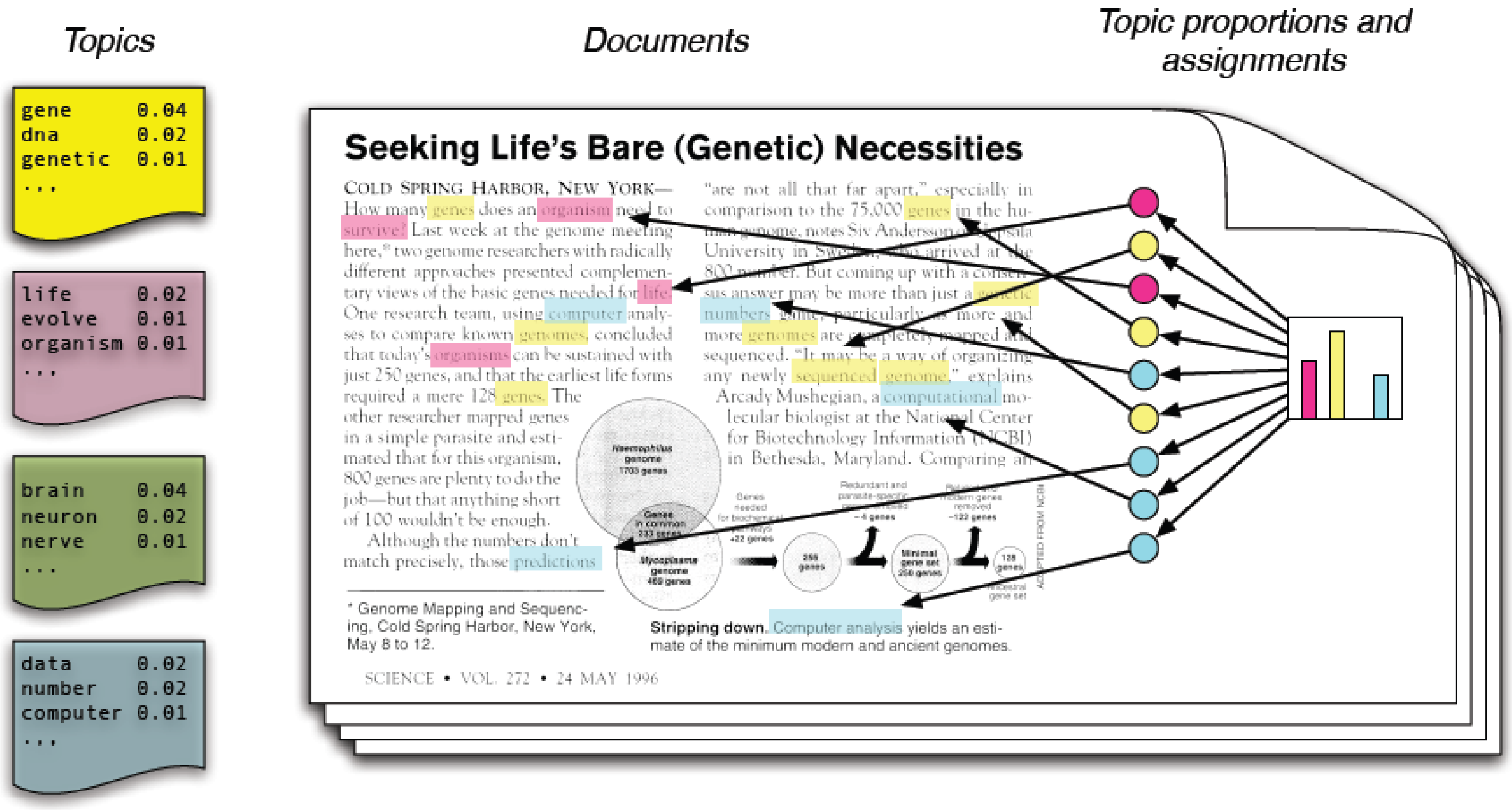
Bayesian networks (topic modeling)

$$p(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D})$$

$$= \prod_{i=1}^{K} p(\beta_i) \prod_{d=1}^{D} p(\theta_d) (\prod_{n=1}^{N} p\big(z_{d,n} | \theta_d\big) p(w_{d,n} | \beta_{1:K}, z_{d,n}))$$



Bayesian networks (topic modeling)


Topics
Documents
Topic proportions and assignments
gene 0.04
dna 0.02
genetic 0.01
...
life 0.02
evolve 0.01
organism 0.01
...
brain 0.04
neuron 0.02
nerve 0.01
...
data 0.02
number 0.02
computer 0.01
...
Haemophilus genome 1703 genes
Genes in common 233 genes
Mycoplasma genome 469 genes
Genes needed for biochemical pathways +22 genes
256 genes
Redundant and parasite-specific genes removed −4 genes
Minimal gene set 250 genes
Related and modern genes removed −122 genes
128 genes
Ancestral gene set
ADAPTED FROM NCBI

Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

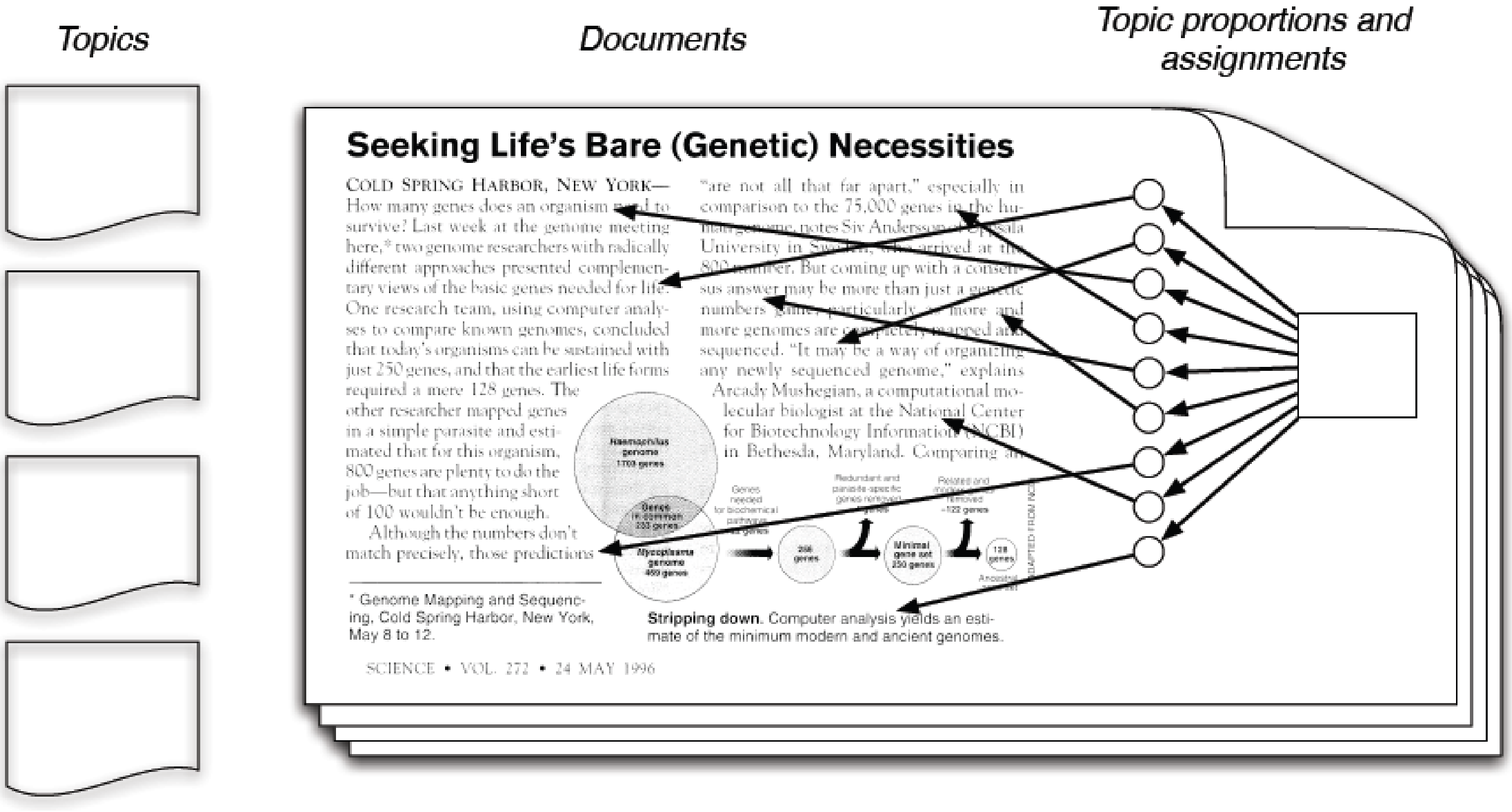
Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

SCIENCE • VOL. 272 • 24 MAY 1996

Bayesian networks (topic modeling)


Topics
Documents
Topic proportions and assignments
Haemophilus genome 1703 genes
Genes in common 233 genes
Mycoplasma genome 469 genes
Genes needed for biochemical pathways
256 genes
Redundant and parasite-specific genes removed
Minimal gene set 250 genes
Related and modern genes removed
–122 genes
128 genes
Ancestral

Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK— How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

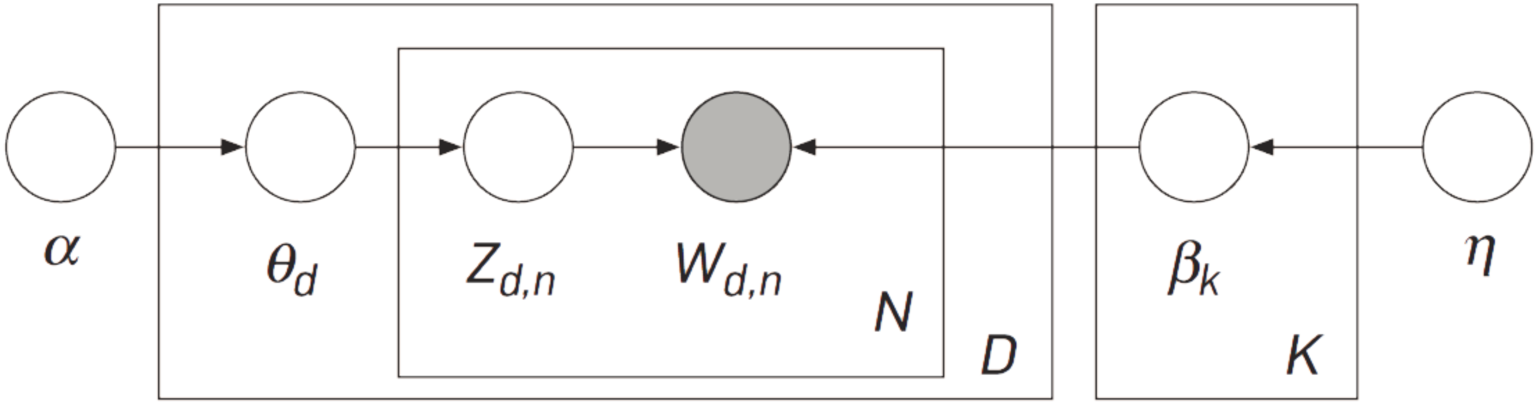
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

SCIENCE • VOL. 272 • 24 MAY 1996

Bayesian networks (topic modeling)

$$p(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D})$$

$$= \prod_{i=1}^{K} p(\beta_i) \prod_{d=1}^{D} p(\theta_d)(\prod_{n=1}^{N} p\big(z_{d,n}|\theta_d\big)p(w_{d,n}|\beta_{1:K}, z_{d,n}))$$

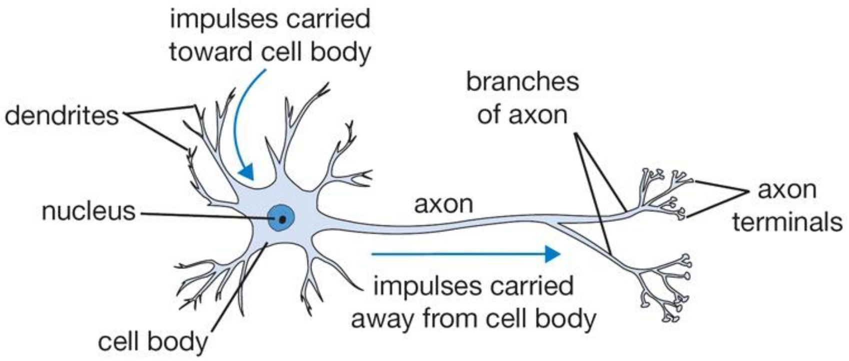


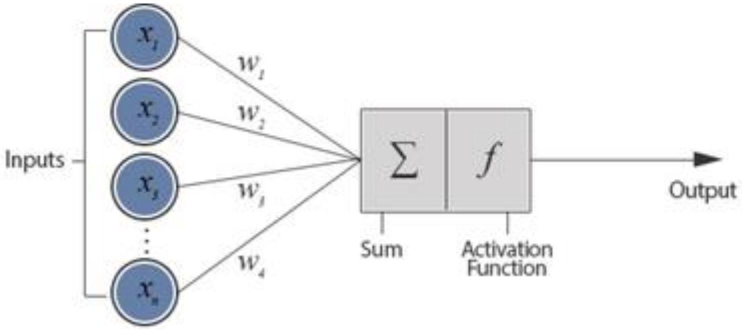
Artificial Neural Networks (ANN)

- A neural network is a machine learning algorithm
- The basic idea of training on data is the same as in decision trees or k-NN
- The main difference is the modularity
- To train a neural network you have to define:
  - **Architecture** (number, type, and arrangement of neurons)
  - **Loss function** (what we optimize)
  - **Optimizer** (algorithm that steer the training)
- If you define the above, you can use the net as any other classifier

Architecture

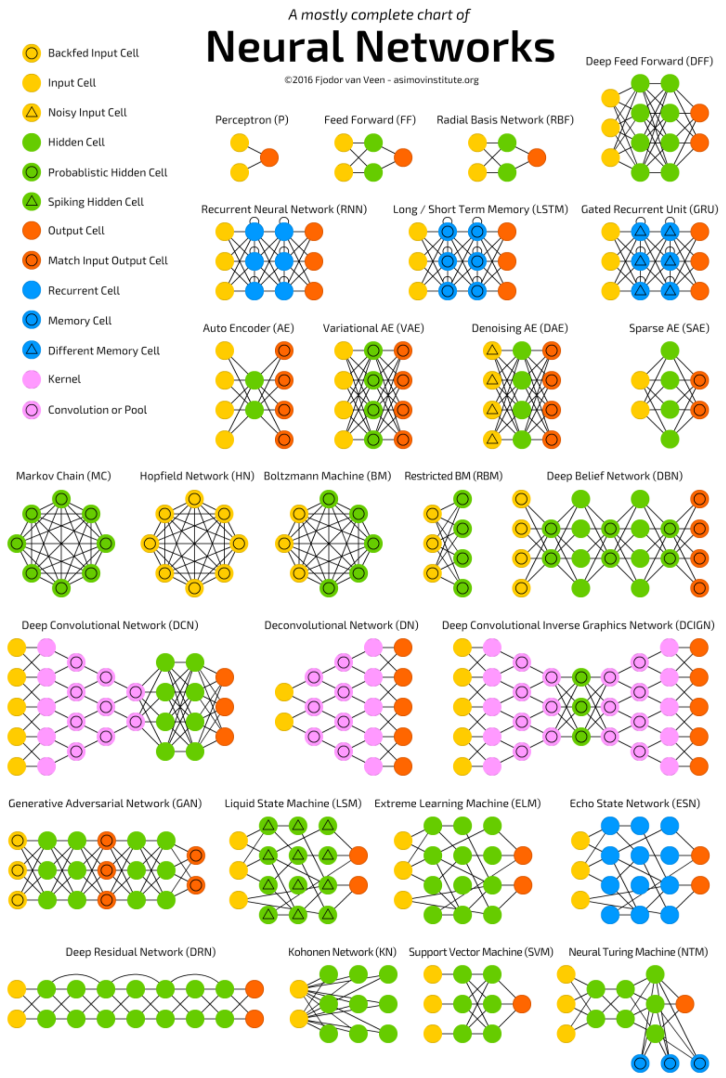
- The basic building block of ANN is a **neuron**
- An artificial neuron is a signal processor:
  - Vector of numbers as **input**
  - Each input is multiplied by a **weight** and summed into an **activation**
  - The activation is then transformed by a (non-linear) **activation function**
  - **Outputs** the result





Architecture

- The architecture defines the tasks the neural netowrk can solve
- A good neural netowrk architecture can transform raw data into features
- Representation learning
- End-to-end learning
- People reuse architectures from others to solve their own problems



A mostly complete chart of
Neural Networks
©2016 Fjodor van Veen - asimovinstitute.org
Backfed Input Cell
Input Cell
Noisy Input Cell
Hidden Cell
Probablistic Hidden Cell
Spiking Hidden Cell
Output Cell
Match Input Output Cell
Recurrent Cell
Memory Cell
Different Memory Cell
Kernel
Convolution or Pool
Perceptron (P)
Feed Forward (FF)
Radial Basis Network (RBF)
Deep Feed Forward (DFF)
Recurrent Neural Network (RNN)
Long / Short Term Memory (LSTM)
Gated Recurrent Unit (GRU)
Auto Encoder (AE)
Variational AE (VAE)
Denoising AE (DAE)
Sparse AE (SAE)
Markov Chain (MC)
Hopfield Network (HN)
Boltzmann Machine (BM)
Restricted BM (RBM)
Deep Belief Network (DBN)
Deep Convolutional Network (DCN)
Deconvolutional Network (DN)
Deep Convolutional Inverse Graphics Network (DCIGN)
Generative Adversarial Network (GAN)
Liquid State Machine (LSM)
Extreme Learning Machine (ELM)
Echo State Network (ESN)
Deep Residual Network (DRN)
Kohonen Network (KN)
Support Vector Machine (SVM)
Neural Turing Machine (NTM)

Loss function

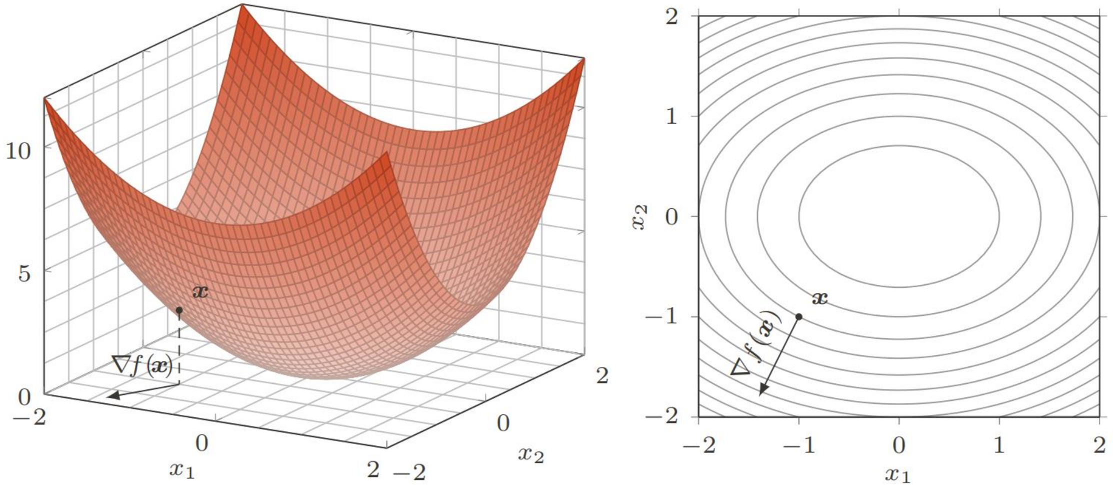
- Loss functions have to be differentiable
- Typical loss function:
  - **MAE, MSE, MAPE**
    - For regression
  - **Categorical cross entropy** (aka softmax)
    - Combined with softmax activation (activation on an entire vector (sic!) of numbers)
    - Class probabilities sum up to 1
  - **Binary cross entropy** (aka sigmoid)
    - For multilabel classification
    - Class probabilities do not have to sum up to 1

Optimizer

- Iteratively updates the input weights

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \nabla f(\boldsymbol{x}_k)$$

- The loss function’s gradient ∇ shows the direction of optimization
- How to define update speed α_k (**learning rate**)
- Multiple heuristics:
  - SGD, SGD + Momentum,
  - Adagrad, RMSprop,
  - Adam, Adadelta

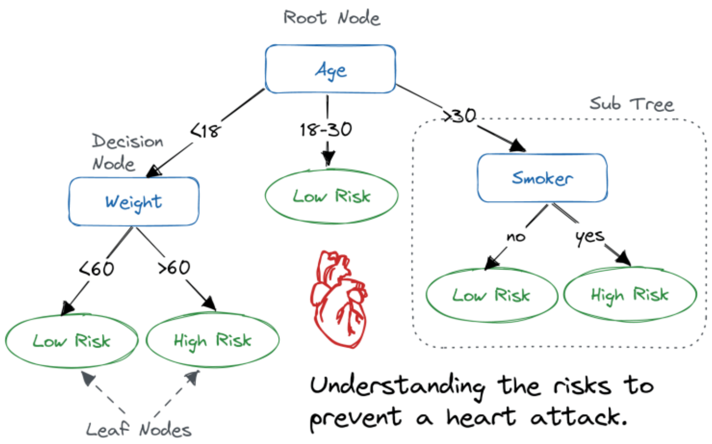


Explainability

- Good predictive performance is not everything
- Many machine learning models are **black boxes**
- When using AI systems that impact our lives **we want to be able to explain model predictions**
  - Bank loans
  - Court case management
  - Medical diagnoses
  - Treatment selection

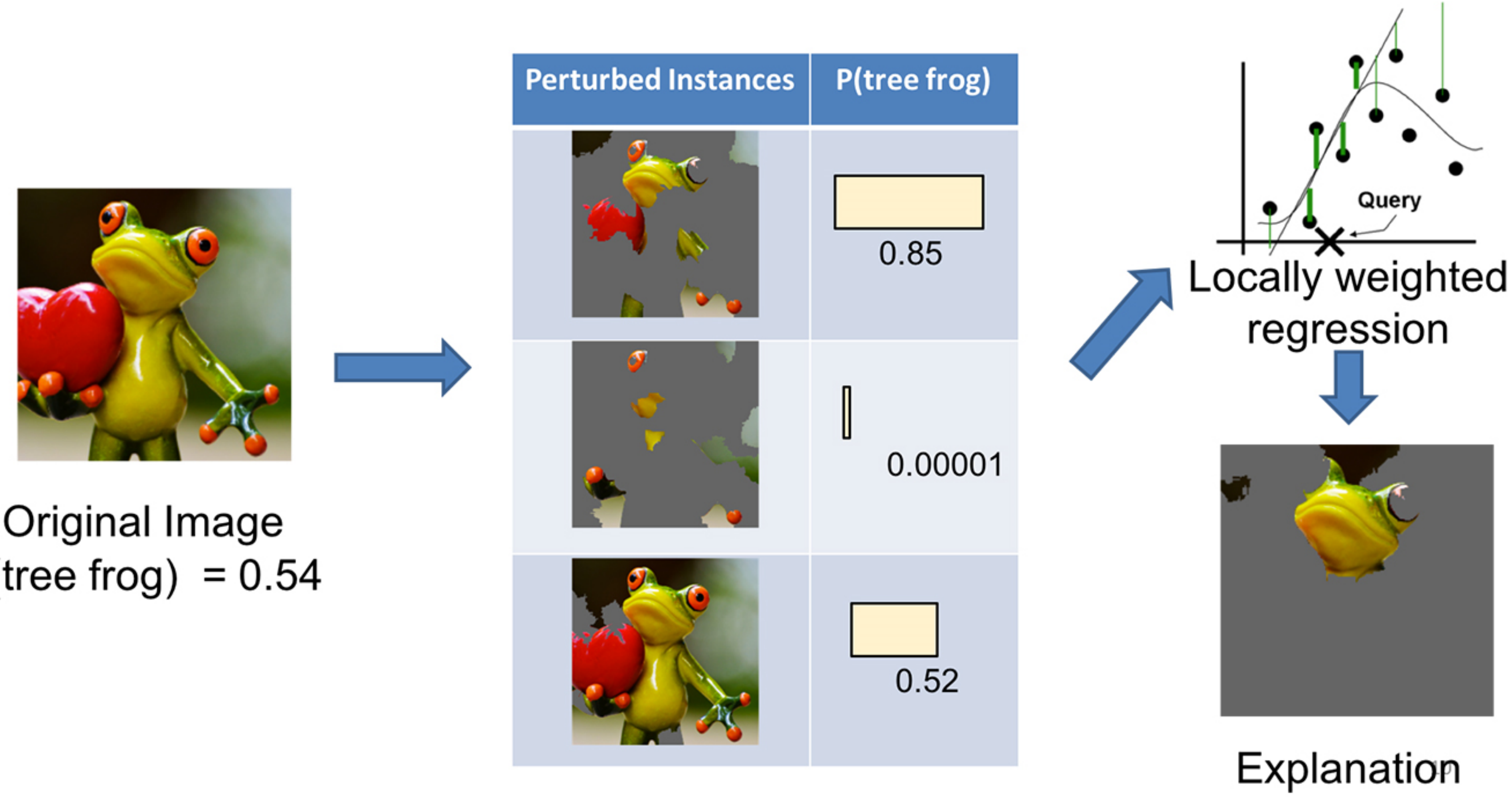
Explainability

- Some models (e.g. rules or decision trees) are interpretable by nature
- In recent years, several model agnostic explainability algorithms have been developed, which work for any machine learning model
- General idea: modify (perturb) examples and see how it affects the prediction
- Popular model agnostic methods:
  - LIME
  - SHAP



Molnar, C. (2020). *Interpretable machine learning*

Model agnostic methods (LIME)



Original Image
P(tree frog) = 0.54

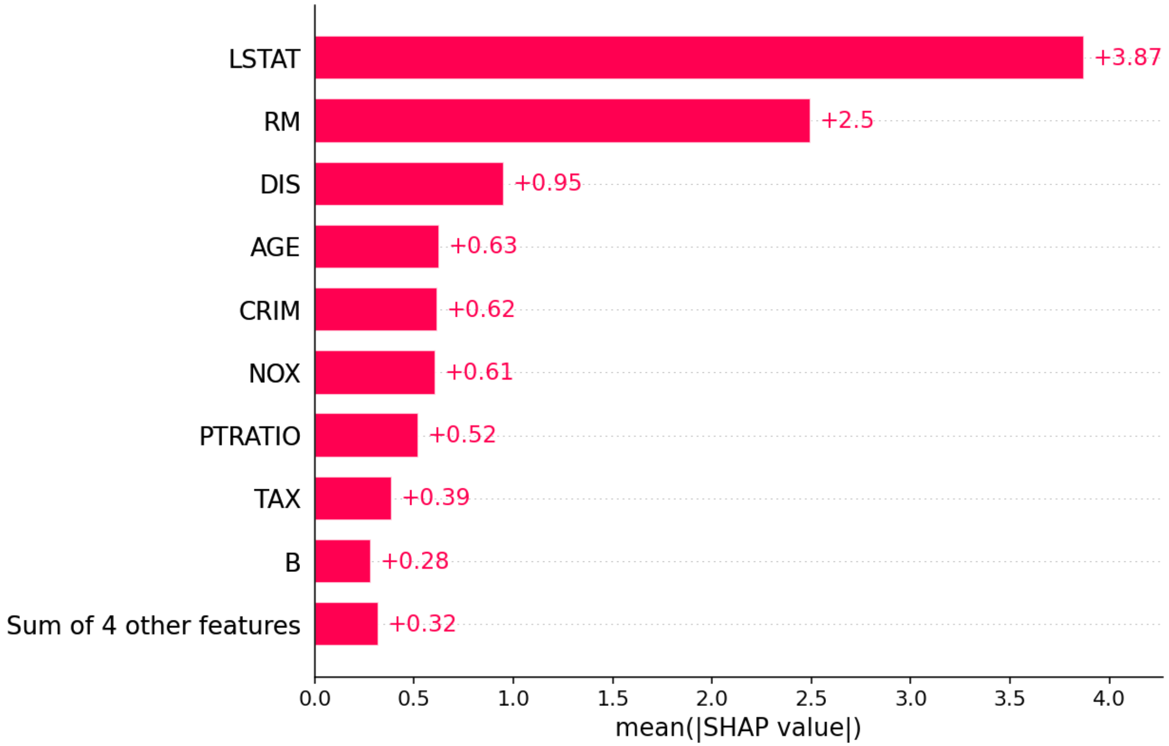
Perturbed Instances	P(tree frog)
	0.85
	0.00001
	0.52

Locally weighted regression

Explanation

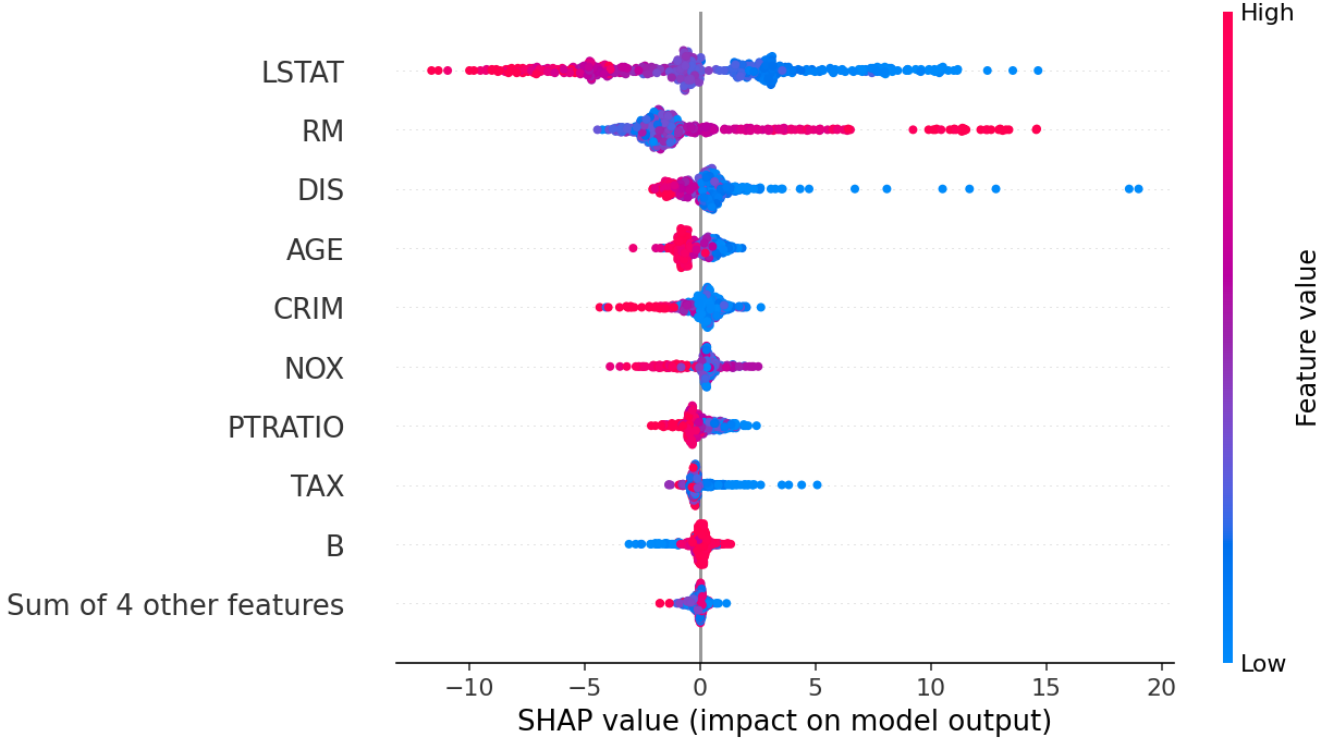
Feature importance plot

Questions answered: *Which features are (on average) the most important when making predictions?*



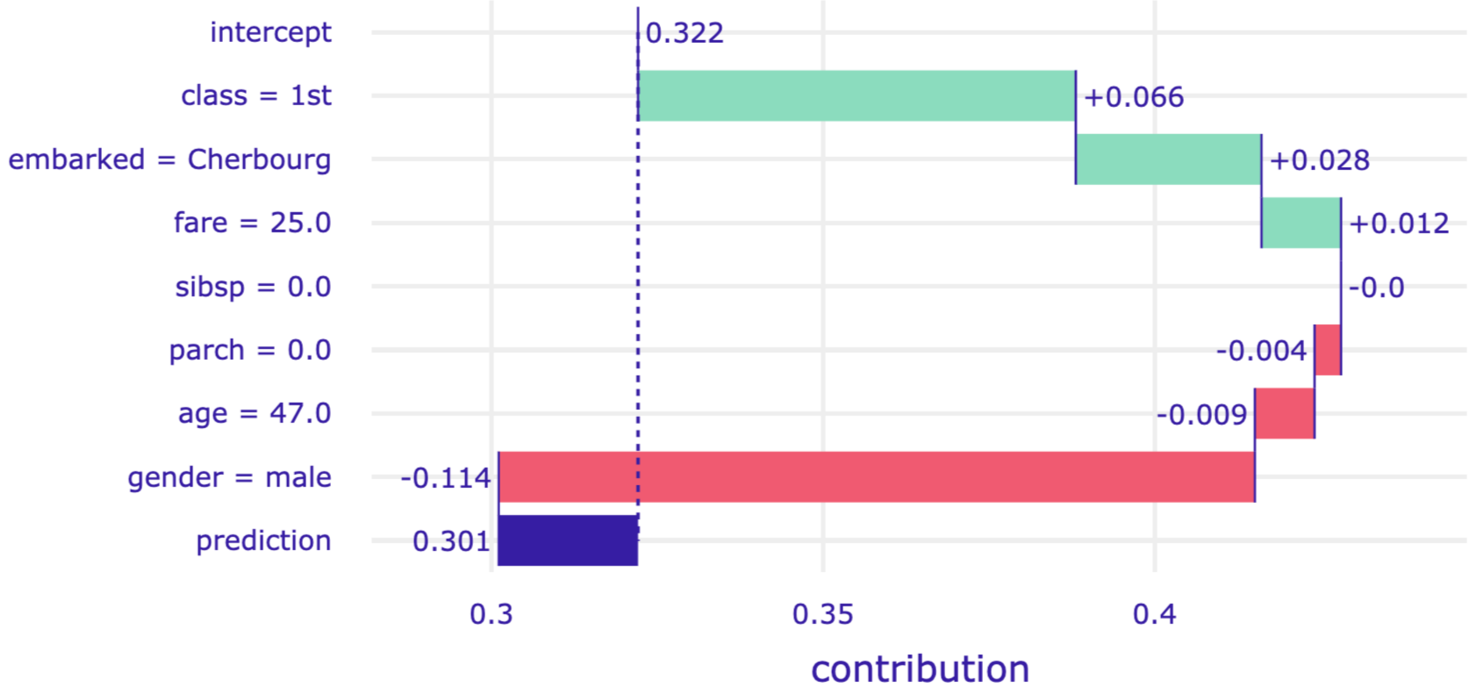
SHAP summaries

Questions answered: *What were the feature importances for each prediction?*



Contribution plots (waterfall chart)

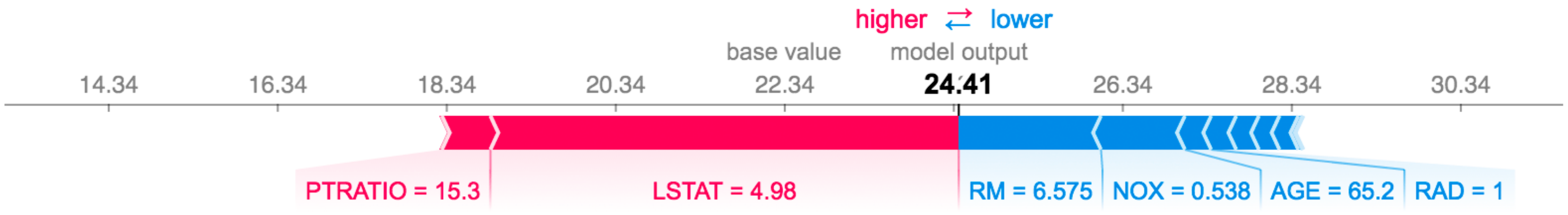
Questions answered: *What contributed to a given prediction?*



Baniecki, H. et al. (2021). Dalex: responsible machine learning with interactive explainability and fairness in python. *Journal of Machine Learning Research*, *22*(214), 1-7.

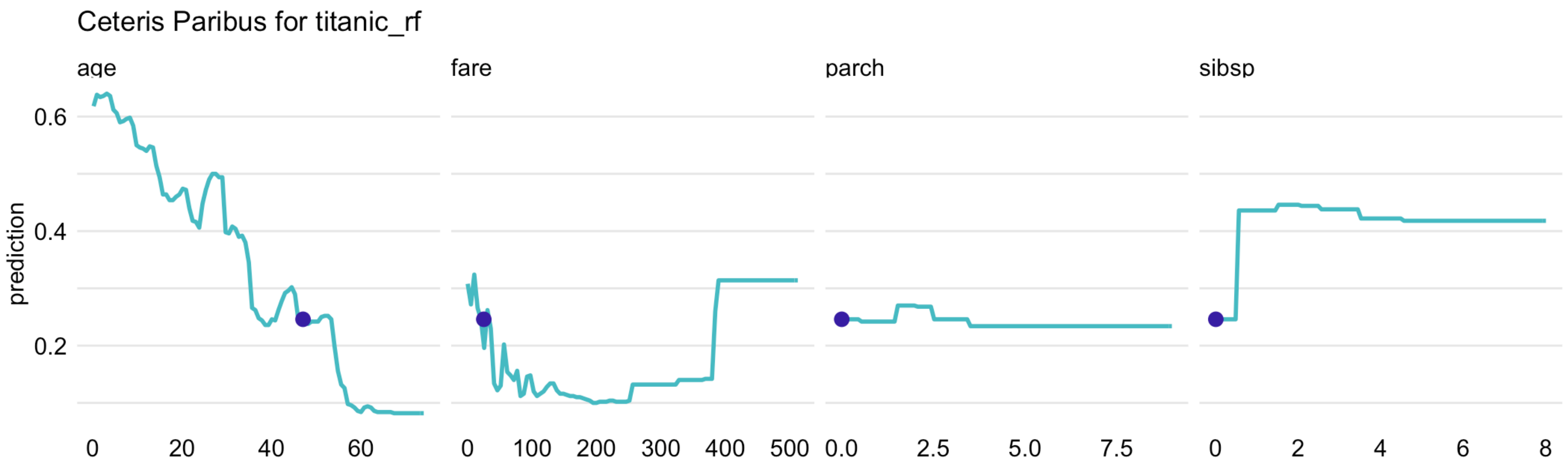
Contribution plots (force plot)

Questions answered: *What contributed to a given prediction?*



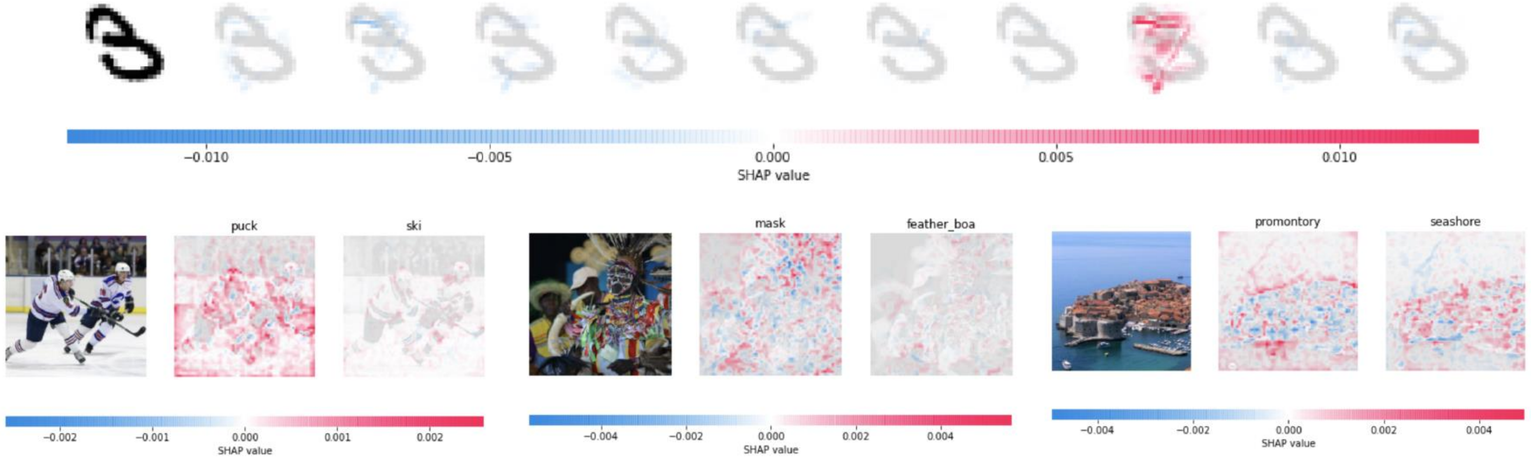
Ceteris paribus profile

Questions answered: *How a change of a feature value would affect a given prediction?*



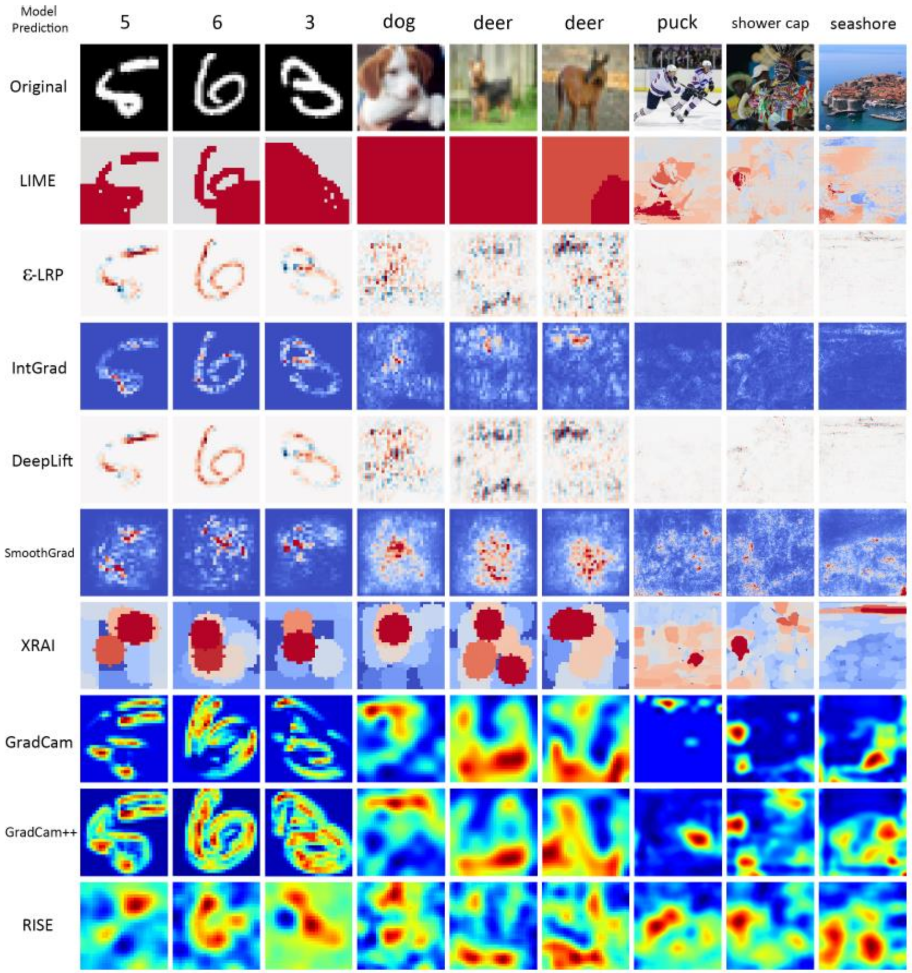
Saliency maps

Question answered: *Which pixels contributed to the prediction?*

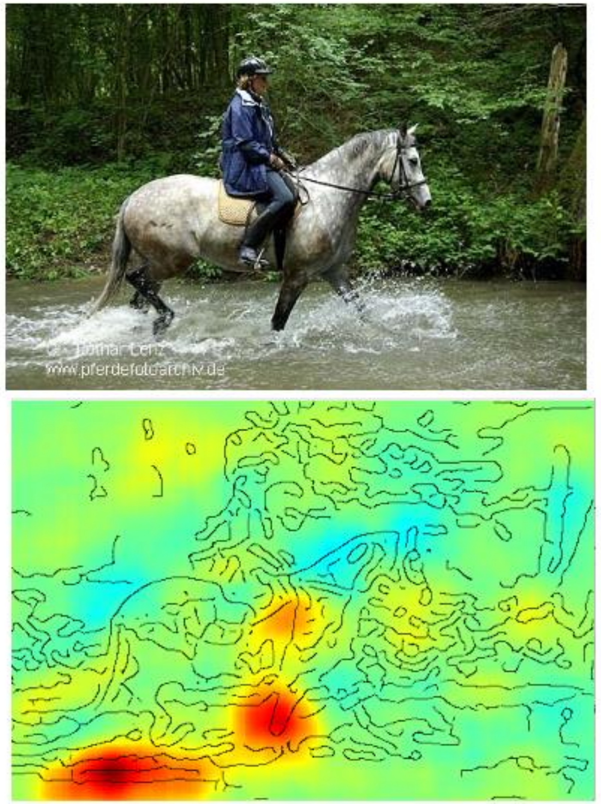


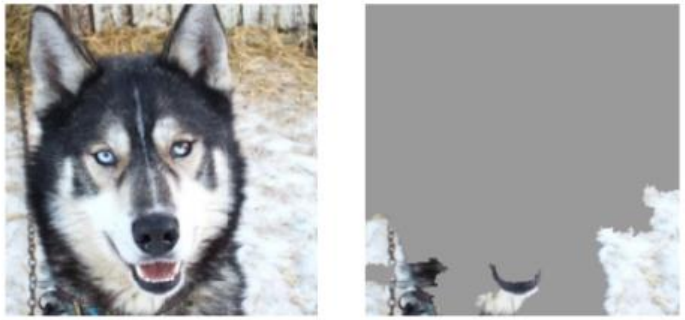
Bodria, F., Giannotti, F., Guidotti, R., Naretto, F., Pedreschi, D., & Rinzivillo, S. (2023). Benchmarking and survey of explanation methods for black box models. *Data Mining and Knowledge Discovery*, *37*(5), 1719-1778.

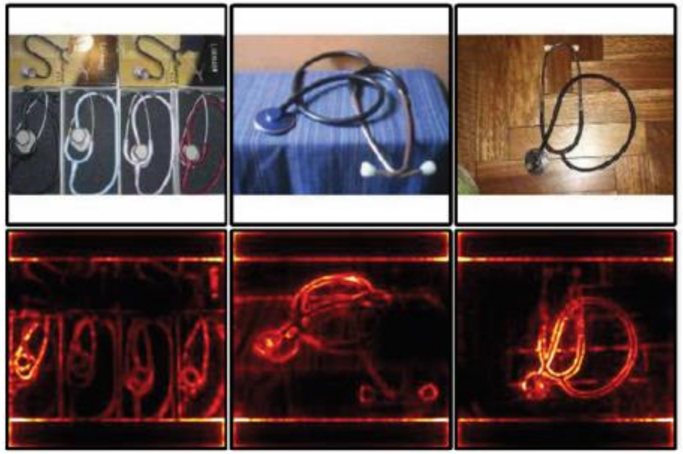
Saliency maps

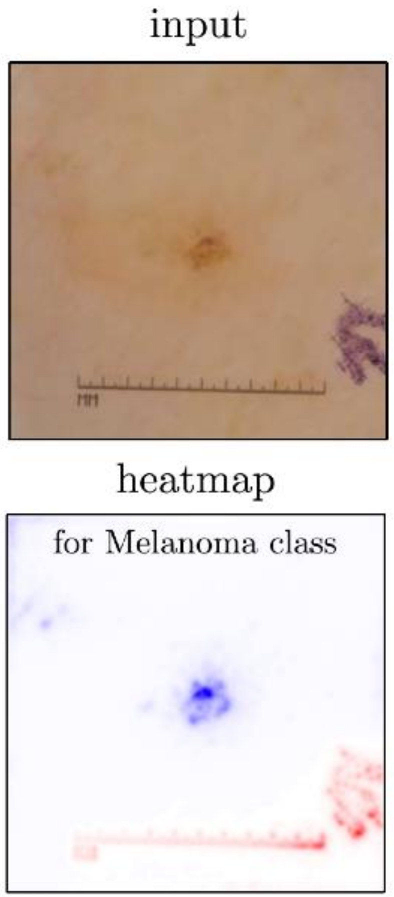


Success stories

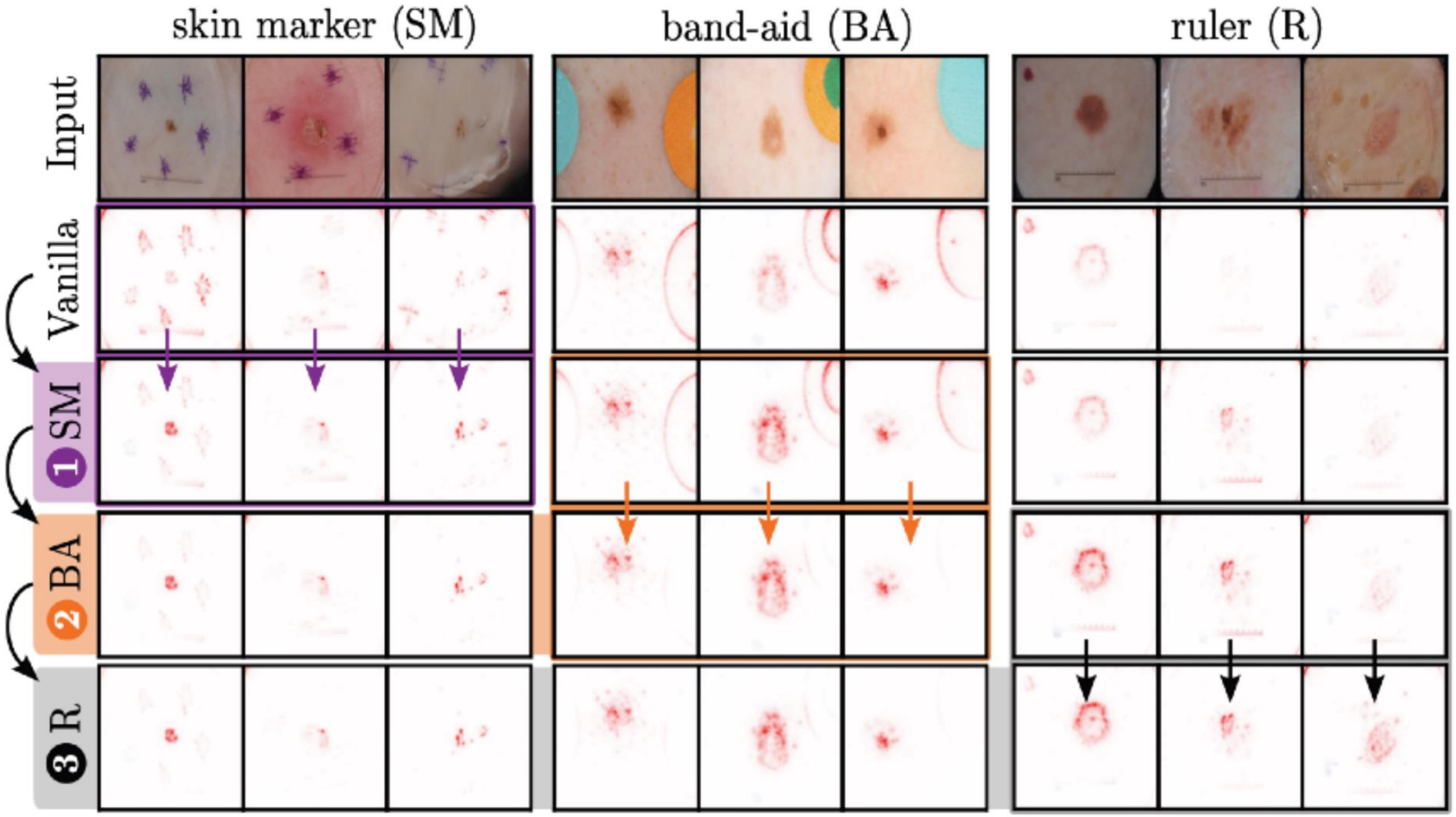



(a) Husky classified as wolf (b) Explanation





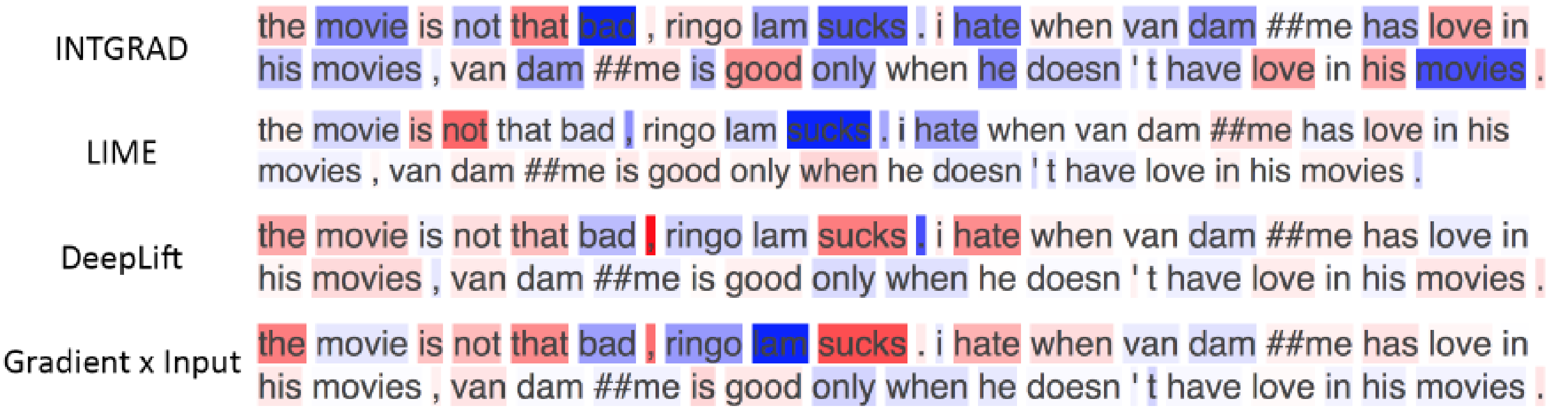
Related topic: machine unlearning



Pahde, F., Dreyer, M., Samek, W., & Lapuschkin, S. (2023). Reveal to revise: An explainable ai life cycle for iterative bias correction of deep models. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 596-606). Cham: Springer Nature Switzerland.

Sentence highlighting

Question answered: *Which words contributed to the prediction?*



INTGRAD

the movie is not that bad , ringo lam sucks . i hate when van dam ##me has love in his movies , van dam ##me is good only when he doesn ' t have love in his movies .

LIME

the movie is not that bad , ringo lam sucks . i hate when van dam ##me has love in his movies , van dam ##me is good only when he doesn ' t have love in his movies .

DeepLift

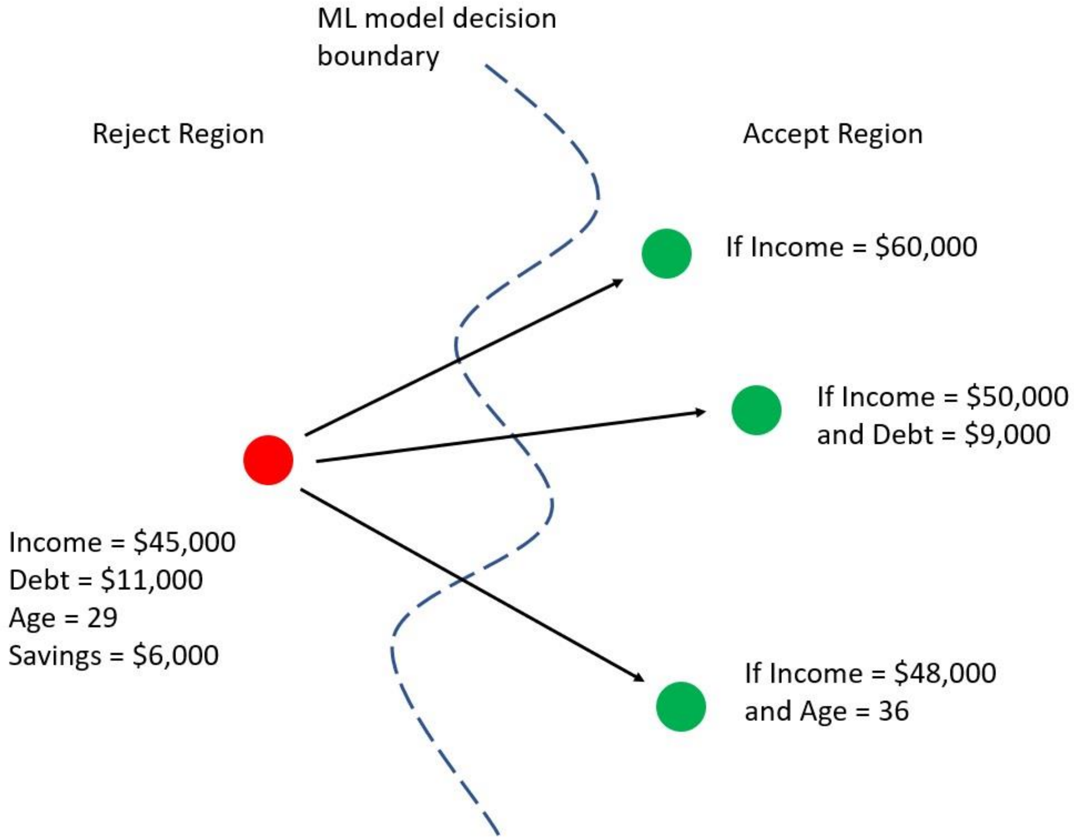
the movie is not that bad , ringo lam sucks . i hate when van dam ##me has love in his movies , van dam ##me is good only when he doesn ' t have love in his movies .

Gradient x Input

the movie is not that bad , ringo lam sucks . i hate when van dam ##me has love in his movies , van dam ##me is good only when he doesn ' t have love in his movies .

Counterfactual explanations

- **Counterfactual explanations** – describe causal relations between changes in conditions and changes in decisions
- In machine learning – the smallest object modification that changes the prediction
- *If the client earned $60 000 anually instead of $45 000, they'd get the loan*



Dandl, S., Molnar, C., Binder, M., & Bischl, B. (2020, August). Multi-objective counterfactual explanations. In *International Conference on Parallel Problem Solving from Nature* (pp. 448-469). Cham: Springer International Publishing.

Summary

- **Artificial intelligence** (machine intelligence) is usually defined as **intelligent actions** rather than thinking
- **Machine learning** is the basic building block of AI
- Predictive models are **patterns discovered by algorithms from data**
- Many algorithms for finding patterns (**probability, function approximation, rules, distance**)
- **Explainability** plays an important role in AI applications
- **Explaining model predictions ≠ explaining phenomenon**