UNIVERSITÄT
ZU KÖLN

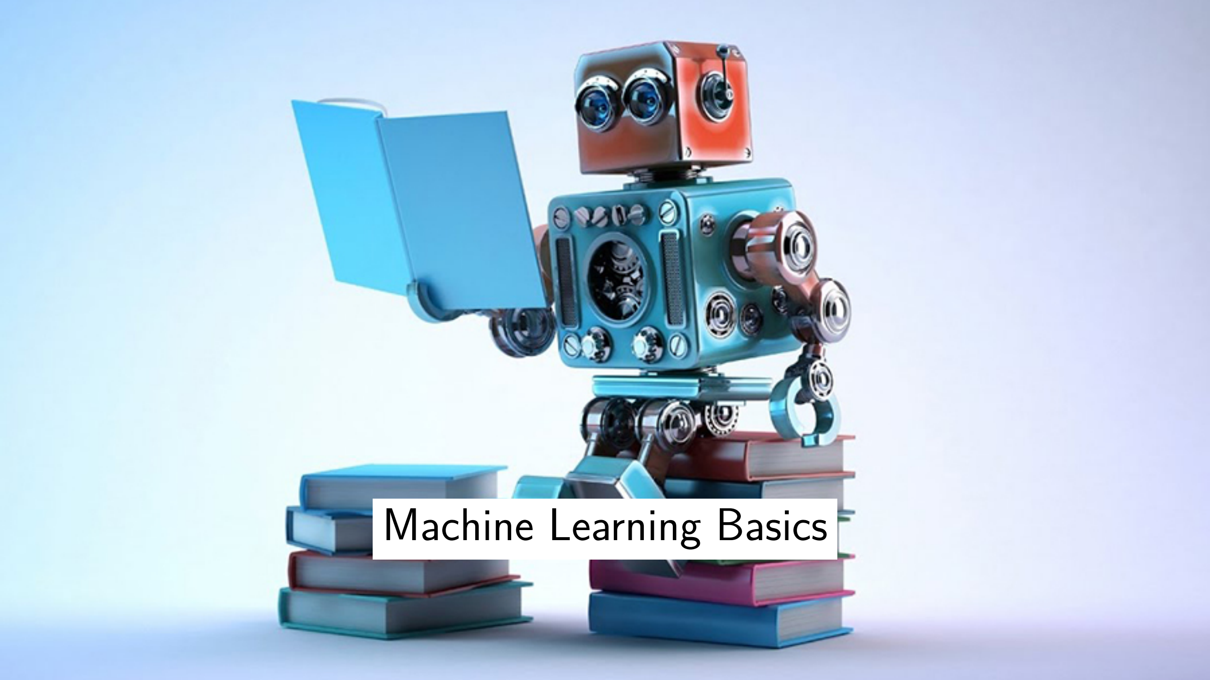
Machine Learning, Part 1

Einführung in die Informationsverarbeitung

Nils Reiter

November 2, 2023

INSTITUT FÜR
DIGITAL HUMANITIES
UNIVERSITÄT ZU KÖLN



Machine Learning Basics

Introduction

- What is machine learning?
  - Method to find patterns, hidden structures and undetected relations in data

Introduction

- What is machine learning?
  - Method to find patterns, hidden structures and undetected relations in data
- It's all around us
  - Stock market transactions
  - Search engines
  - Surveillance
  - Data-driven research & science
  - ...

Introduction

- What is machine learning?
  - Method to find patterns, hidden structures and undetected relations in data
- It's all around us
  - Stock market transactions
  - Search engines
  - Surveillance
  - Data-driven research & science
  - ...
- Why is it interesting for text analysis?
  - Big data analyses
    - Automatic prediction of phenomena
    - Canonisation, Euro-centrism
    - Statements about 1000 texts more convincing than abt 10
  - Insights into data
    - By inspecting features and making error analysis

Two Parts

Prediction Model

How do we make predictions on data instances?
(e.g., how do we assign a part of speech tag for a word?)

Learning Algorithm

How do we create a prediction model, given annotated data?
(e.g. how do we create rules for assigning a part of speech tag for a word?)

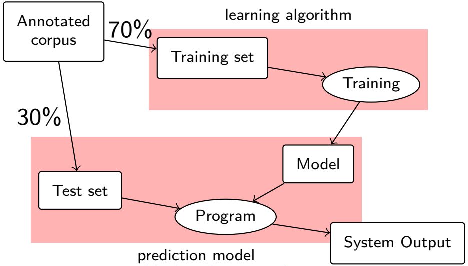
Two Parts

Prediction Model

How do we make predictions on data instances?
(e.g., how do we assign a part of speech tag for a word?)

Learning Algorithm

How do we create a prediction model, given annotated data?
(e.g. how do we create rules for assigning a part of speech tag for a word?)



Classification

- Assigning *classes* to *objects/instances/items*

Classification

- Assigning *classes* to *objects/instances/items*
  - Words → parts of speech

Classification

- Assigning *classes* to *objects/instances/items*
  - Words → parts of speech
  - Texts → genres

Classification

- Assigning *classes* to *objects/instances/items*
  - Words → parts of speech
  - Texts → genres
  - Portrait photos → name of depicted person

Classification

- Assigning *classes* to *objects/instances/items*
  - Words → parts of speech
  - Texts → genres
  - ~~Portrait photos → name of depicted person~~

Classification

- Assigning *classes* to *objects/instances/items*
  - Words → parts of speech
  - Texts → genres
  - ~~Portrait photos → name of depicted person~~
- Prediction model: Responsible for the classification

Classification

- Assigning *classes* to *objects/instances/items*
  - Words $\rightarrow$ parts of speech
  - Texts $\rightarrow$ genres
  - ~~Portrait photos $\rightarrow$ name of depicted person~~
- Prediction model: Responsible for the classification
- Many different models/algorithms available (all with variants):
  - Decision trees
  - Support vector machines
  - Naïve bayes
  - Neural networks
  - Bayesian networks
  - ...

Classification

Target classes

Classes: A finite set of categories

Examples

- Parts of speech: Noun, verb, adjective, ...
  - E.g., STTS tagset
- Argument analysis: Pro or con some claim
- Smart home: Is a person at home or not based on sensor input
- Genres: Abenteuerroman, Bildungsroman, Kriminalroman, ...
  - ⚠ But: Novels may fall in multiple classes

Classification

Target classes

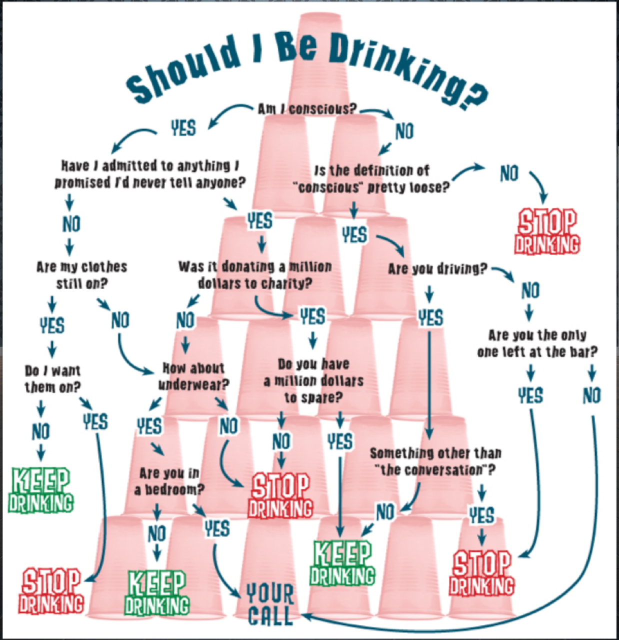
Classes: A finite set of categories

Examples

- Parts of speech: Noun, verb, adjective, ...
  - E.g., STTS tagset
- Argument analysis: Pro or con some claim
- Smart home: Is a person at home or not based on sensor input
- Genres: Abenteuerroman, Bildungsroman, Kriminalroman, ...
  - ⚠ But: Novels may fall in multiple classes

Important first step: Clearly identify classes and problem properties

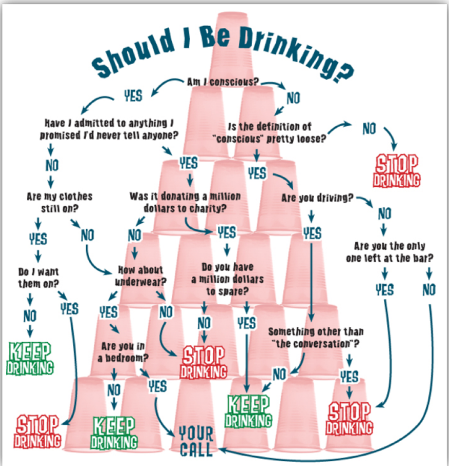
Decision Trees



Should I Be Drinking?
Am I conscious?
YES
NO
Have I admitted to anything I promised I'd never tell anyone?
Is the definition of "conscious" pretty loose?
NO
STOP DRINKING
NO
YES
YES
Are my clothes still on?
Was it donating a million dollars to charity?
Are you driving?
YES
NO
NO
YES
YES
NO
Do I want them on?
How about underwear?
Do you have a million dollars to spare?
Are you the only one left at the bar?
NO
YES
YES
NO
NO
YES
YES
NO
KEEP DRINKING
Are you in a bedroom?
STOP DRINKING
Something other than "the conversation"?
NO
YES
NO
YES
STOP DRINKING
KEEP DRINKING
YOUR CALL
KEEP DRINKING
STOP DRINKING

Decision Trees

Prediction Model – Toy Example



- What are the instances?
  - Situations we are in
    (this is not really automatisable)

Decision Trees

Prediction Model – Toy Example

- What are the instances?
  - Situations we are in (this is not really automatisable)
- What are the features?
  - Consciousness
  - Clothing situation
  - Promises made
  - Whether we are driving
  - ...

Decision Trees

Prediction Model

- Each non-leaf node in the tree represents one feature
- Each leaf node represents a class label
- Each branch at this node represents one possible feature value
  - Number of branches = number of possible values

Decision Trees

Prediction Model

- Each non-leaf node in the tree represents one feature
- Each leaf node represents a class label
- Each branch at this node represents one possible feature value
  - Number of branches $=$ number of possible values
- Make a prediction for x:
  1. Start at root node
  2. If it's a leaf node
     - assign the class label
  3. Else
     - Check node which feature is to be tested (f_i)
     - Extract $f_i(x)$
     - Follow corresponding branch
     - Go to 2

Decision Trees

Learning Algorithm (Quinlan, 1986)

- Core idea: The tree represents splits of the training data
  1. Start with the full data set D_{train} as D
  2. If D only contains members of a single class:
     - Done.
  3. Else:
     - Select a feature f_i
     - Extract feature values of all instances in D
     - Split the data set according to f_i: $D = D_v \cup D_w \cup D_u \ldots$
     - Go back to 2

Decision Trees

Learning Algorithm (Quinlan, 1986)

- Core idea: The tree represents splits of the training data
  1. Start with the full data set D_{train} as D
  2. If D only contains members of a single class:
     - Done.
  3. Else:
     - Select a feature f_i
     - Extract feature values of all instances in D
     - Split the data set according to f_i: $D = D_v \cup D_w \cup D_u \ldots$
     - Go back to 2
- Remaining question: How to select features?

Decision Trees

Feature Selection

- What is a good feature?
  - One that maximizes homogeneity in the split data set

Decision Trees

Feature Selection

- What is a good feature?
  - One that maximizes homogeneity in the split data set
- "Homogeneity"
  - Increase
    $\{✔✔✔✖\} = \{✖\} \cup \{✔✔✔\}$
  - No increase
    $\{✔✔✔✖\} = \{✔\} \cup \{✔✔✖\}$

Decision Trees

Feature Selection

- What is a good feature?
  - One that maximizes homogeneity in the split data set
- "Homogeneity"
  - Increase
    $\{✔✔✔✖\} = \{✖\} \cup \{✔✔✔\} \leftarrow$ better split!
  - No increase
    $\{✔✔✔✖\} = \{✔\} \cup \{✔✔✖\}$
- Homogeneity: Entropy/information — Shannon (1948)

Decision Trees

Feature Selection

- What is a good feature?
  - One that maximizes homogeneity in the split data set
- "Homogeneity"
  - Increase
    $\{✔✔✔✖\} = \{✖\} \cup \{✔✔✔\} \leftarrow$ better split!
  - No increase
    $\{✔✔✔✖\} = \{✔\} \cup \{✔✔✖\}$
- Homogeneity: Entropy/information — Shannon (1948)
- Rule: Always select the feature with the highest *information gain* (IG)
  - (= the highest reduction in entropy = the highest increase in homogeneity)

Entropy

Shannon (1948)

number of classes present in X

relative frequency of the class

$$H(X) \quad = \quad -\sum_{i=1}^{n} \; p(x_i) \;\; \log_b p(x_i)$$

entropy

- A metric for the uncertainty in a random variable

Entropy
Example

- How certain are we in predicting the next value?
  - “aaaaaaaaaaaaaa” – only one symbol, very certain

Entropy

Example

- How certain are we in predicting the next value?
  - "aaaaaaaaaaaaaa" – only one symbol, very certain
    - $H = -\sum_1^1 p(a) \log_2 p(a) = -1 \log_2 1 = 0$

Entropy

Example

- How certain are we in predicting the next value?
  - "aaaaaaaaaaaaaa" – only one symbol, very certain
    - $H = -\sum_1^1 p(a) \log_2 p(a) = -1 \log_2 1 = 0$
  - "abbaabbabbaaba" – two symbols, evenly distributed, 50:50

Entropy

Example

- How certain are we in predicting the next value?
  - “aaaaaaaaaaaaaa” – only one symbol, very certain
    - $H = -\sum_1^1 p(a) \log_2 p(a) = -1 \log_2 1 = 0$
  - “abbaabbabbaaba” – two symbols, evenly distributed, 50:50
    - $H = -\left(p(a) \log_2 p(a) + p(b) \log_2 p(b)\right) = ((0.5 \times -1) + (0.5 \times -1)) = 1$

Entropy

Example

- How certain are we in predicting the next value?
  - "aaaaaaaaaaaaaa" – only one symbol, very certain
    - $H = -\sum_1^1 p(a) \log_2 p(a) = -1 \log_2 1 = 0$
  - "abbaabbababbaaba" – two symbols, evenly distributed, 50:50
    - $H = -\left(p(a) \log_2 p(a) + p(b) \log_2 p(b)\right) = ((0.5 \times -1) + (0.5 \times -1)) = 1$
  - "bbabbababbbaba" – two symbols, unevenly distributed, 33:66

Entropy

Example

- How certain are we in predicting the next value?
  - "aaaaaaaaaaaaaa" – only one symbol, very certain
    - $H = -\sum_1^1 p(a)\log_2 p(a) = -1\log_2 1 = 0$
  - "abbaabbabbaaba" – two symbols, evenly distributed, 50:50
    - $H = -\left(p(a)\log_2 p(a) + p(b)\log_2 p(b)\right) = ((0.5 \times -1) + (0.5 \times -1)) = 1$
  - "bbabbababbbaba" – two symbols, unevenly distributed, 33:66
    - $H = -\left(0.333\log_2 0.333 + 0.666\log_2 0.666\right) = 0.91$

Entropy

Example

- How certain are we in predicting the next value?
  - "aaaaaaaaaaaaaa" – only one symbol, very certain
    - $H = -\sum_1^1 p(a) \log_2 p(a) = -1 \log_2 1 = 0$
  - "abbaabbababbaaba" – two symbols, evenly distributed, 50:50
    - $H = -\left(p(a) \log_2 p(a) + p(b) \log_2 p(b)\right) = ((0.5 \times -1) + (0.5 \times -1)) = 1$
  - "bbabbababbbaba" – two symbols, unevenly distributed, 33:66
    - $H = -\left(0.333 \log_2 0.333 + 0.666 \log_2 0.666\right) = 0.91$
  - "nmkfjigeoahlpdcb" – 16 symbols, very uncertain

Entropy

Example

- How certain are we in predicting the next value?
  - "aaaaaaaaaaaaaa" – only one symbol, very certain
    - $H = -\sum_1^1 p(a) \log_2 p(a) = -1 \log_2 1 = 0$
  - "abbaabbababbaaba" – two symbols, evenly distributed, 50:50
    - $H = -\left(p(a) \log_2 p(a) + p(b) \log_2 p(b)\right) = ((0.5 \times -1) + (0.5 \times -1)) = 1$
  - "bbabbababbbaba" – two symbols, unevenly distributed, 33:66
    - $H = -\left(0.333 \log_2 0.333 + 0.666 \log_2 0.666\right) = 0.91$
  - "nmkfjigeoahlpdcb" – 16 symbols, very uncertain
    - $H = -16 \times -0.25 = 4$

Entropy

Example

- How certain are we in predicting the next value?
  - "aaaaaaaaaaaaaa" – only one symbol, very certain
    - $H = -\sum_1^1 p(a) \log_2 p(a) = -1 \log_2 1 = 0$
  - "abbaabbababbaaba" – two symbols, evenly distributed, 50:50
    - $H = -(p(a) \log_2 p(a) + p(b) \log_2 p(b)) = ((0.5 \times -1) + (0.5 \times -1)) = 1$
  - "bbabbababbbaba" – two symbols, unevenly distributed, 33:66
    - $H = -(0.333 \log_2 0.333 + 0.666 \log_2 0.666) = 0.91$
  - "nmkfjigeoahlpdcb" – 16 symbols, very uncertain
    - $H = -16 \times -0.25 = 4$
- Interpretation: We need $H(X)$ bits to encode the next symbol

Entropy

Application

- Data Representation: How to represent the text “abca” in memory?
- Variant 1: Three states to distinguish
  - a = $\boxed{0}\boxed{0}$, b = $\boxed{0}\boxed{1}$, c = $\boxed{1}\boxed{0}$
  - Memory consumption: 2 bits per character

Entropy

Application

- Data Representation: How to represent the text “abca” in memory?
- Variant 1: Three states to distinguish
  - a = $\boxed{0}\boxed{0}$, b = $\boxed{0}\boxed{1}$, c = $\boxed{1}\boxed{0}$
  - Memory consumption: 2 bits per character
- Variant 2: Some symbols are more frequent than the others!
  - a = $\boxed{0}$, b= $\boxed{1}\boxed{0}$, c= $\boxed{1}\boxed{1}$
  - Memory consumption: 1.5 bits per character
  - This is the entropy of “abca” – the minimal memory consumption

Decision Trees

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log_b p(x_i)$$

Examples (with $b = 2$)

- $H(\{✔✔✔✔\}) = -\frac{4}{4}\log_2\frac{4}{4} = 0$
- $H(\{✔✔✔✖\}) = -\left(\underbrace{\frac{3}{4}\log_2\frac{3}{4}}_{✔} + \underbrace{\frac{1}{4}\log_2\frac{1}{4}}_{✖}\right) = 0.562$
- $H(\{✔✔✖✖\}) = \ldots = 0.693$

Decision Trees

Feature Selection (2)

$\{\checkmark\checkmark\checkmark\times\}$

/ \

$\{\times\}\{\checkmark\checkmark\checkmark\}$

$\{\checkmark\checkmark\checkmark\times\}$

/ \

$\{\checkmark\}\{\checkmark\checkmark\times\}$

$$\begin{aligned} H(\{\checkmark\checkmark\checkmark\times\}) &= H([3,1]) \\ &= 0.562 \\ H(\{\times\}) &= H([1]) = 0 \\ H(\{\checkmark\checkmark\checkmark\}) &= H([3]) \\ &= 0 \end{aligned}$$

$$\begin{aligned} H(\{\checkmark\checkmark\checkmark\times\}) &= H([3,1]) \\ &= 0.562 \\ H(\{\checkmark\}) &= H([1]) = 0 \\ H(\{\checkmark\checkmark\times\}) &= H([2,1]) \\ &= 0.637 \end{aligned}$$

Decision Trees

Feature Selection (3)

$$
\begin{aligned}
H(\{\checkmark\checkmark\checkmark\times\}) &= 0.562 \\
H(\{\times\}) &= 0 \\
H(\{\checkmark\checkmark\checkmark\}) &= 0
\end{aligned}
\qquad
\begin{aligned}
H(\{\checkmark\checkmark\checkmark\times\}) &= 0.562 \\
H(\{\checkmark\}) &= 0 \\
H(\{\checkmark\checkmark\times\}) &= 0.637
\end{aligned}
$$

$$
\begin{aligned}
IG(f_1) &= H(\{\checkmark\checkmark\checkmark\times\}) - \varnothing\left(H(\{\times\}), H(\{\checkmark\checkmark\checkmark\})\right) \\
&= 0.562 - 0 = 0.562 \\
IG(f_2) &= H(\{\checkmark\checkmark\checkmark\times\}) - \varnothing\left(H(\{\checkmark\}), H(\{\checkmark\checkmark\times\})\right) \\
&= 0.562 - (\frac{3}{4}0.637 + \frac{1}{4}0) \\
&= 0.562 - 0.562 - 0.477 = 0.085
\end{aligned}
$$

Example: TreeTagger

Helmut Schmid (1994). "Probabilistic part-of-speech tagging using decision trees". In: *Proceedings of the conference on New Methods in Language Processing* 12

- Web page: `https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/`
- Models for many different languages
  - Including middle High German by Echelmeyer et al. (2017)

Example: TreeTagger

Helmut Schmid (1994). "Probabilistic part-of-speech tagging using decision trees". In: *Proceedings of the conference on New Methods in Language Processing* 12

- Web page: `https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/`
- Models for many different languages
  - Including middle High German by Echelmeyer et al. (2017)
- Lexicon to provide candidates (and probabilities)
- Previous two pos tags as features for a decision tree

Summary

Decision Tree

- Classification algorithm
- Built around trees, recursive learning and prediction
- Pros
  - Highly transparent (if the number of features is not very large)
  - Reasonably fast
  - Dependencies between features can be incorporated into the model
- Cons
  - No pairwise dependencies
  - May lead to overfitting
  - Only nominal features
- Variants exist

References I

Echelmeyer, Nora/Nils Reiter/Sarah Schulz (2017). “Ein PoS–Tagger für „das” Mittelhochdeutsche”. In: *Book of Abstracts of DHd 2017*. Bern, Switzerland. DOI: 10.18419/opus-9023. URL: https://elib.uni-stuttgart.de/handle/11682/9040.

Quinlan, J. R (1986). “Induction of Decision Trees”. In: *Machine Learning* 1.1, pp. 81–106.

Schmid, Helmut (1994). “Probabilistic part-of-speech tagging using decision trees”. In: *Proceedings of the conference on New Methods in Language Processing* 12.

Shannon, Claude E. (1948). “A mathematical theory of communication”. In: *The Bell System Technical Journal* 27.3, pp. 379–423.