Machine Learning Course
basic track

Machine Learning
Lecture 8: Feature engineering & feature importances

Harbour.Space University
February 2020

Radoslav Neychev

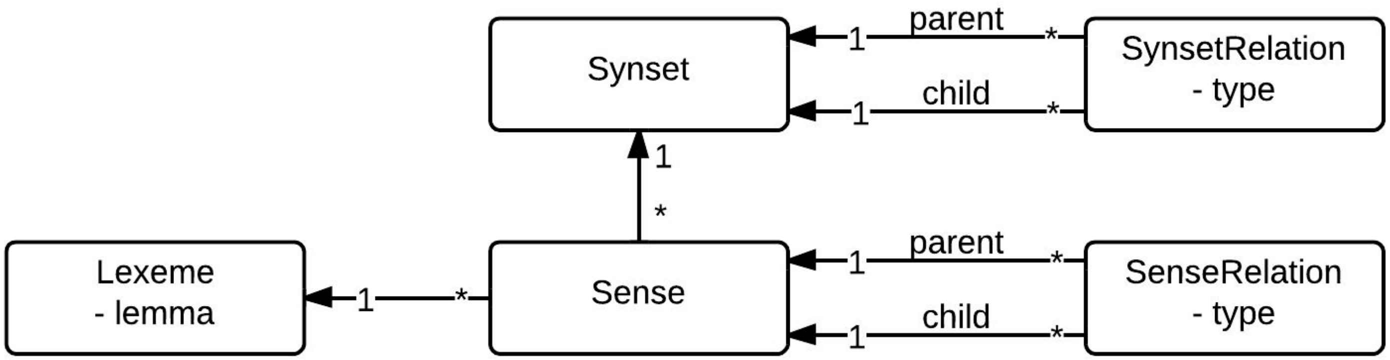
Outline

1. Word representations and categorical features
2. Missing data
3. Feature importances estimation

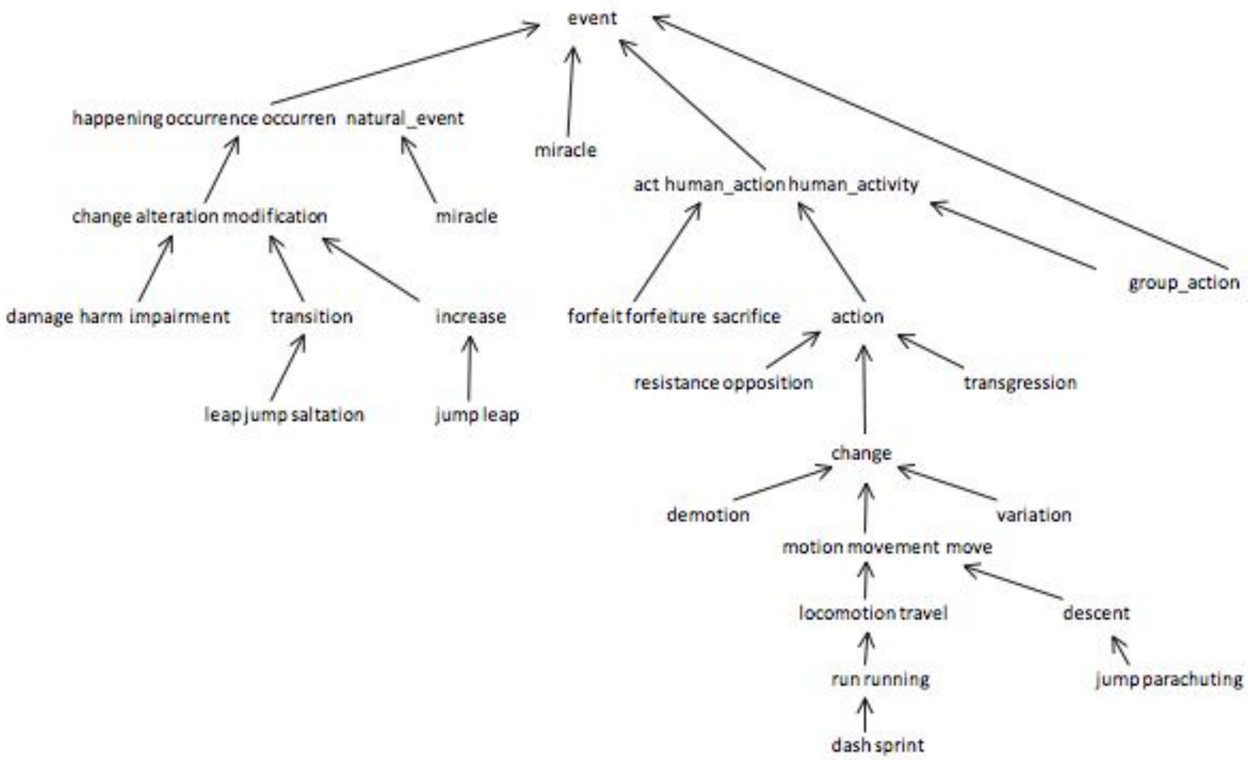
Based on ml-mipt materials by Victor Kantor and Evgeny Sokolov HSE 2018 materials

How to represent text in a computer?

Use a taxonomy like WordNet that has hypernyms (is-a) relationships and synonym sets



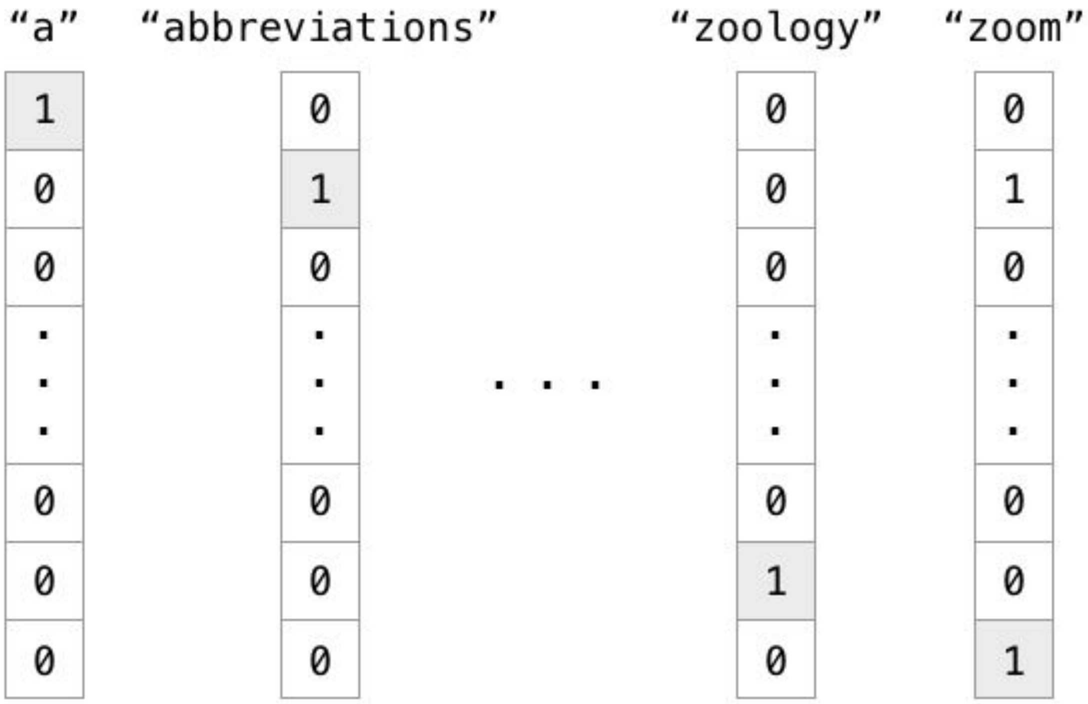
How to represent text in a computer: WordNet



Discrete representations: problems

- Missing new words
- Subjective
- Requires human labor to create and adapt
- Hard to compute accurate word similarity

Discrete representations: one-hot encoding



TF-IDF

If word is repeated in the query, it's probably important

Repetitions of query words in the document ➔ good

Rare words more important

$$s(Q,D) = \sum_{w} tf_{w,Q} \cdot \underbrace{\frac{tf_{w,D}}{tf_{w,D} + \frac{k|D|}{avg|D|}}}_{\text{Repetitions of same word less important than different words. Except in very long documents}} \cdot \log \frac{|C|}{df_w}$$

The more query words we match, the better. Σ over the vocabulary

TF - term frequency

IDF - Inversed Document Frequency

TF-IDF: make it simple

$$\mathrm{tf}(''\mathsf{this}'', d_1) = \frac{1}{5} = 0.2$$
$$\mathrm{tf}(''\mathsf{this}'', d_2) = \frac{1}{7} \approx 0.14$$

$$\mathrm{idf}(''\mathsf{this}'', D) = \log\left(\frac{2}{2}\right) = 0$$

$$\mathrm{tfidf}(''\mathsf{this}'', d_1, D) = 0.2 \times 0 = 0$$
$$\mathrm{tfidf}(''\mathsf{this}'', d_2, D) = 0.14 \times 0 = 0$$

Document 1

Term	Term Count
this	1
is	1
a	2
sample	1

Document 2

Term	Term Count
this	1
is	1
another	2
example	3

Word 'this' is not very informative

Words cooccurrences

One of the most successful ideas of statistical NLP:

“You shall know a word by the company it keeps”

(J. R. Firth 1957: 11)

Words cooccurrences

Finding N-grams in a text

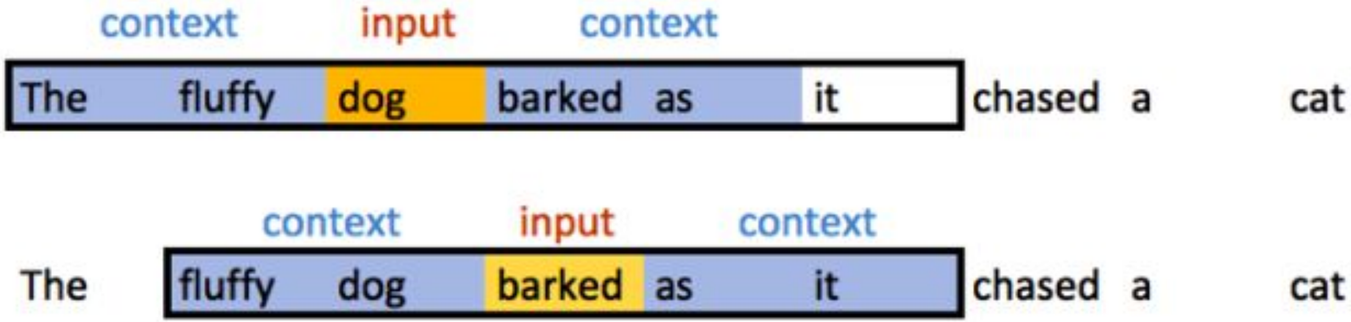
Word-document
cooccurrence matrix

Window around
each word

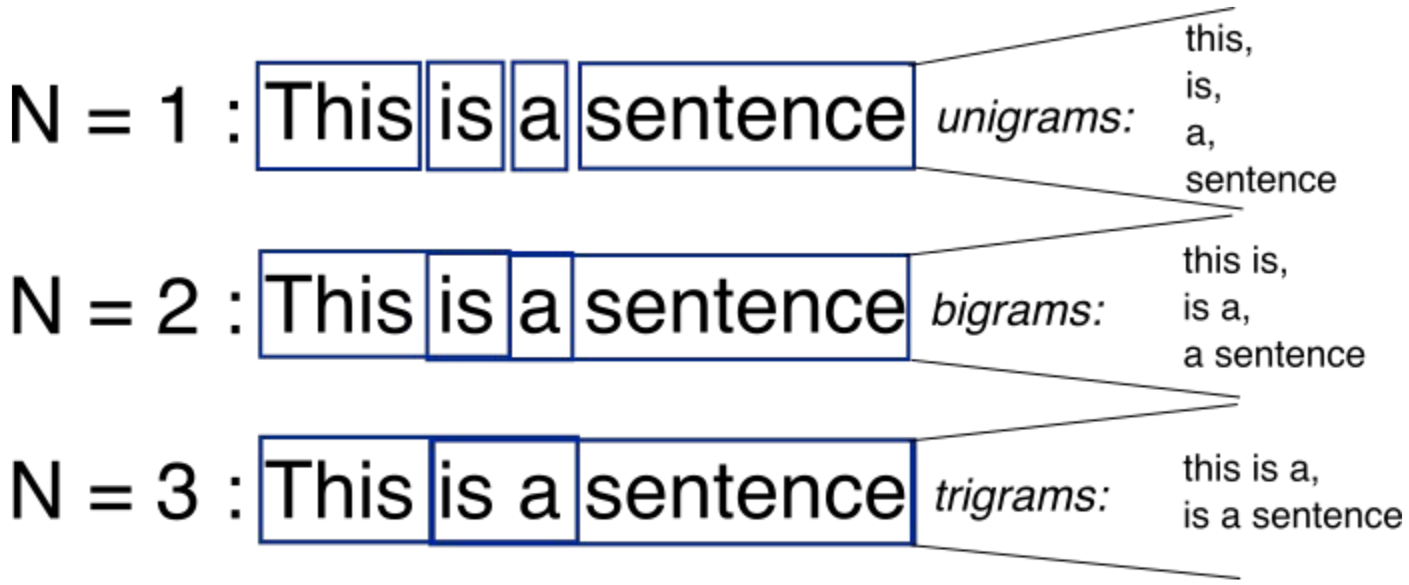
Word-document cooccurrence matrix

$$X = \begin{array}{c} \\ I \\ like \\ enjoy \\ deep \\ learning \\ NLP \\ flying \\ . \end{array} \begin{array}{c} \begin{array}{cccccccc} I & like & enjoy & deep & learning & NLP & flying & . \end{array} \\ \begin{bmatrix} 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

Words cooccurrences: sliding window



Words cooccurrences: n-grams



Cooccurrence vectors: problems

- Increase in size with vocabulary
- Very high dimensional: require a lot of storage
- Subsequent classification models have sparsity issues

↓

Models are less robust

Feature importance estimation

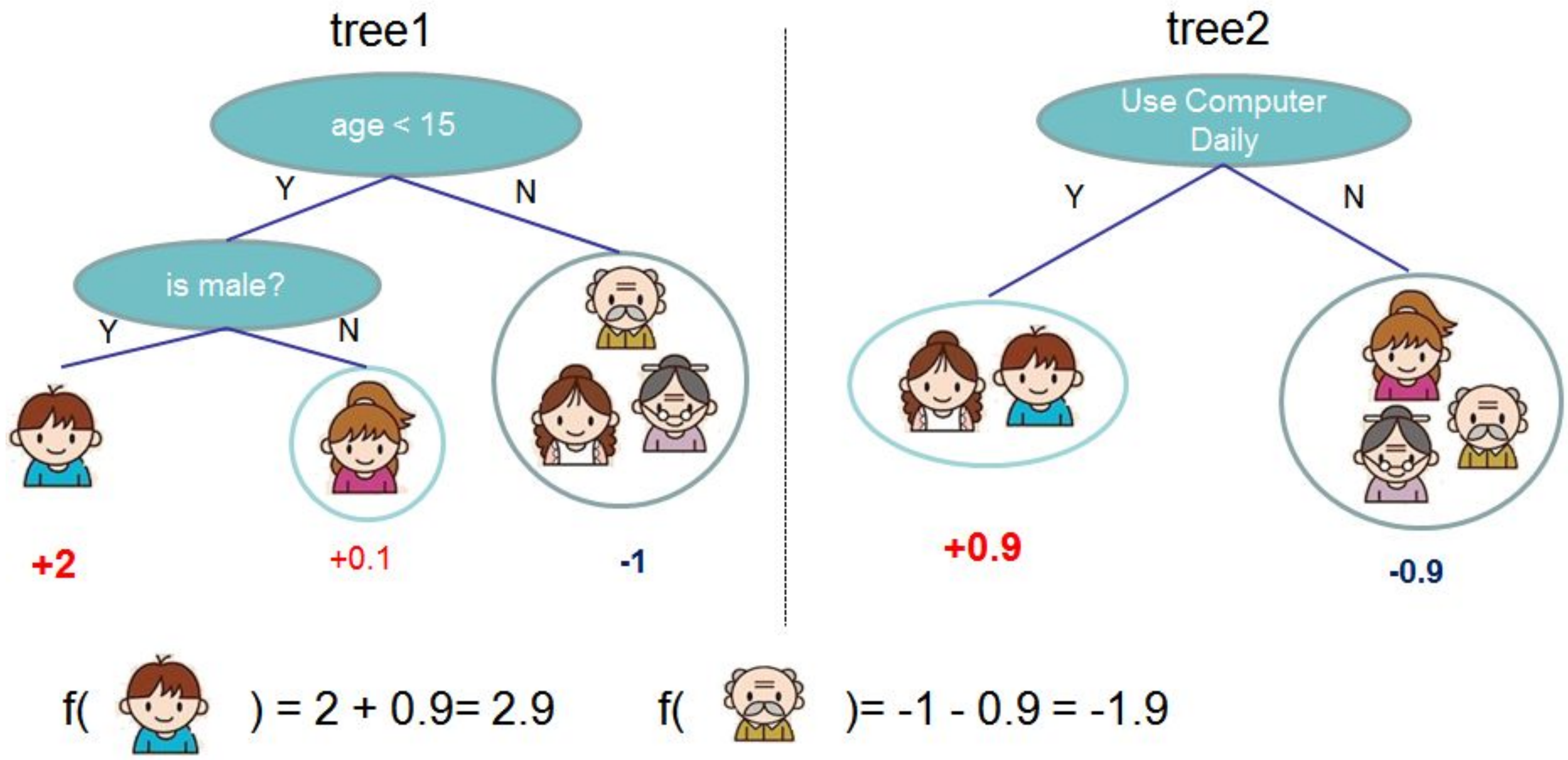


Image source: https://habr.com/ru/post/428213/

Feature importance estimation

1. Permutation importance
2. Partial Dependence Plots (PDP)
3. Tree specific:
   a. Gain
   b. Frequency (Split Count)
   c. Cover (weighted Split Count)
4. Shap

Permutation importance

Height at age 20 (cm)	Height at age 10 (cm)	...	Socks owned at age 10
182	155	...	20
175	147	...	10
...	...	...	...
156	142	...	8
153	130	...	24

Image source: https://www.kaggle.com/dansbecker/permutation-importance

Permutation importance

Height at age 20 (cm)	Height at age 10 (cm)	...	Socks owned at age 10
182	155	...	20
175	147	...	10
...	...	...	...
156	142	...	8
153	130	...	24

Train model

Observe changes caused by feature random permutations

Image source: https://www.kaggle.com/dansbecker/permutation-importance

Partial Dependence Plots

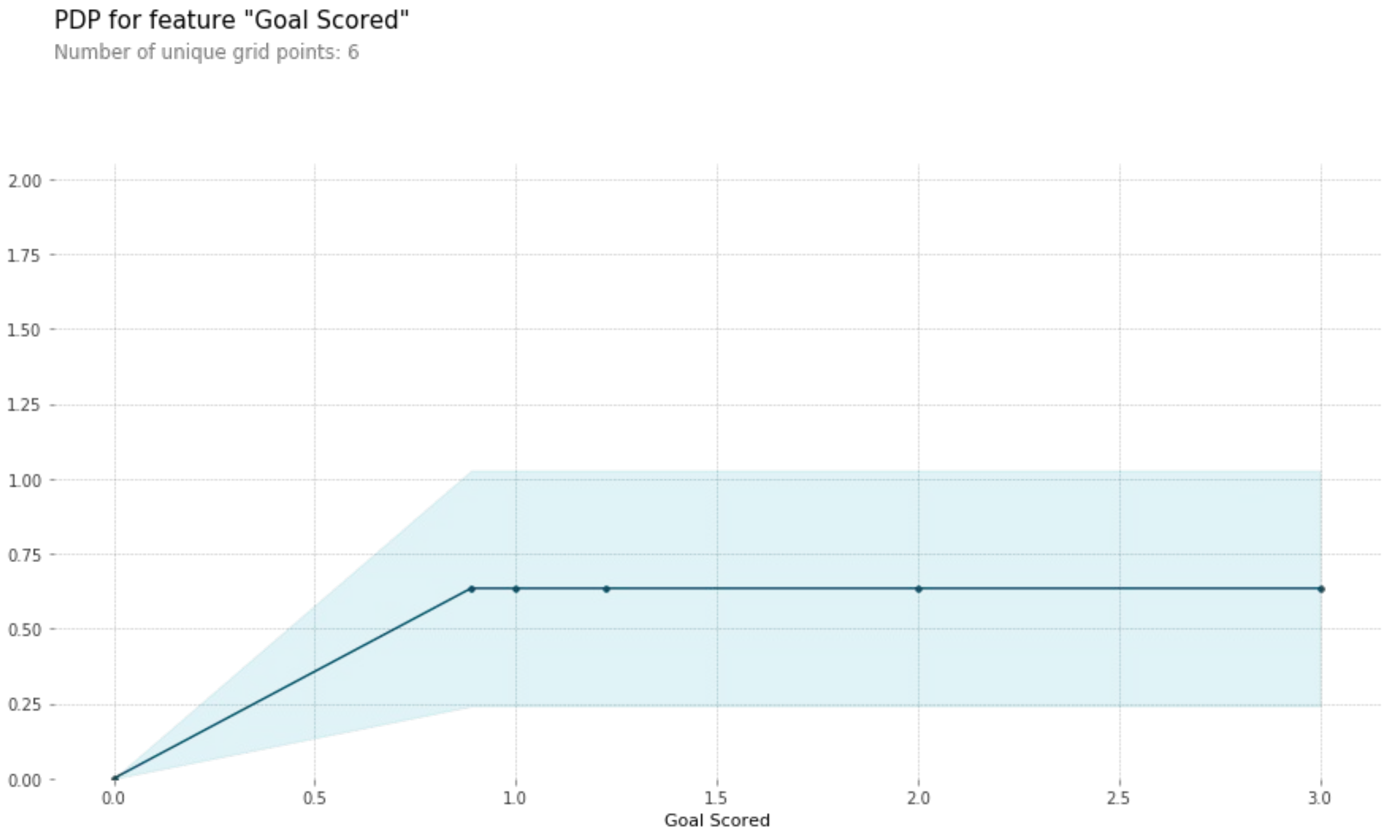


Image source: https://www.kaggle.com/dansbecker/partial-plots

Importance estimation problems

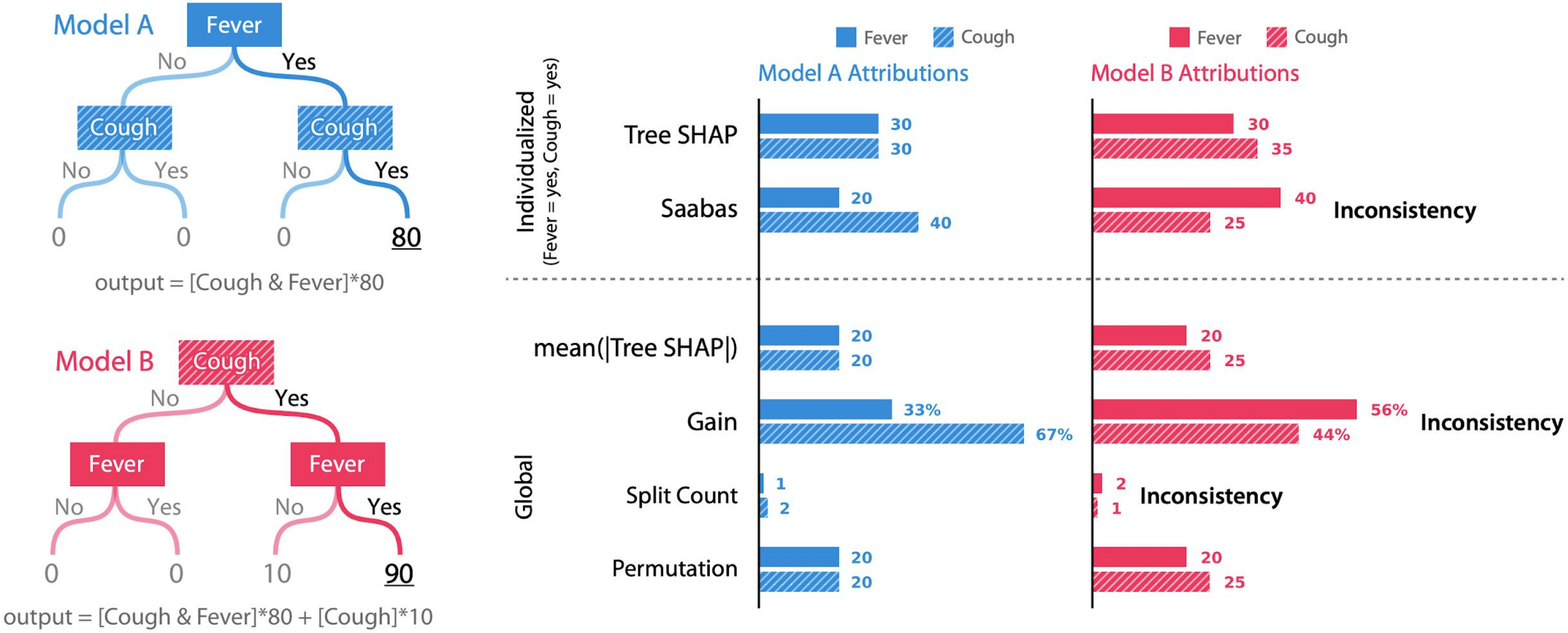


Image source: [“Consistent Individualized Feature Attribution for Tree Ensembles”] paper

Consider *i-th* feature. Shap value will be

$$\phi_i(p) = \sum_{S \subseteq N/\{i\}} \frac{|S|!(n-|S|-1)!}{n!}(p(S \cup \{i\}) - p(S))$$

where $p(S \cup \{i\})$ is model prediction on feature subset S with *i-th* feature added.

Shap values

Consider *i-th* feature. Shap value will be

$$\phi_i(p) = \sum_{S \subseteq N/\{i\}} \frac{|S|!(n - |S| - 1)!}{n!} (p(S \cup \{i\}) - p(S))$$

where $p(S \cup \{i\})$ is model prediction on feature subset S with *i-th* feature added.

SHAP values are the only consistent and locally accurate individualized feature attributions

See Consistent Individualized Feature Attribution for Tree Ensembles paper for more info

Outro

1. Remember the bias-varience decomposition
2. Consider using SHAP values to estimate feature importances.