Applied Machine Learning

Feature Selection: Introduction

Learning goals

- Reasons for Feature Selection
- Types of Feature Selection

Recap

WHAT IS FEATURE ENGINEERING?

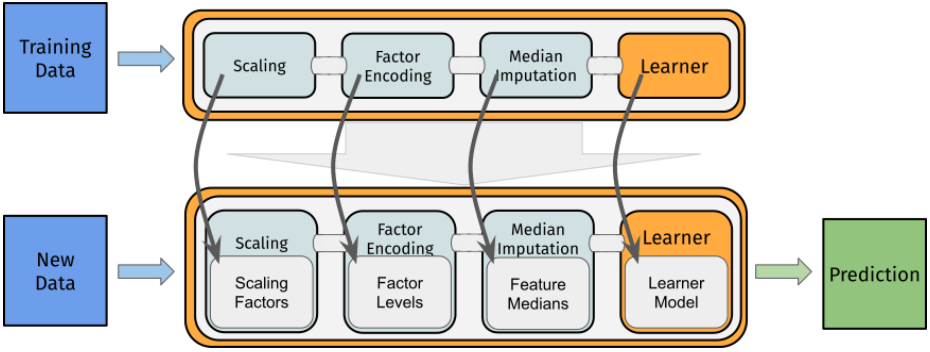
Feature engineering is on the intersection of **data cleaning**, **feature creation** and **feature selection**.

The goal is to solve common difficulties in data science projects, like

- skewed/weird feature distributions,
- (high cardinality) categorical features,
- functional (temporal) features,
- missing observations,
- high dimensional data,
- ...

and improve model performance (or make the data readable by the model in the first place).

TRAIN-TEST LEAKAGE



TYPES OF FEATURE ENGINEERING (SO FAR)

- numerical encoding of categorical features
- feature transformations
- imputation

Feature Selection

MOTIVATION

- Naive view:
  - More features $\rightarrow$ more information $\rightarrow$ discriminant power $\uparrow$
  - Model is not harmed by irrelevant features since their parameters can simply be estimated as 0.
- In practice, irrelevant and redundant features can "confuse" learners (see **curse of dimensionality**) and worsen performance.
- Example: In linear regression, R^2 is monotonically increasing in p, but adding irrelevant features leads to overfitting (capturing noise).

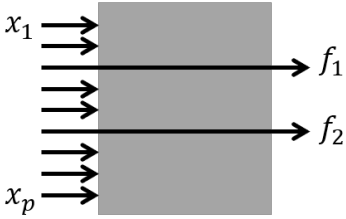
SIZE OF DATASETS

Many new forms of technical measurements and connected data leads to availability of extremely high-dimensional data sets.

- **Classical setting**: Up to around 10^2 features, feature selection might be relevant, but benefits often negligible.
- **Datasets of medium to high dimensionality**: At around 10^2 to 10^3 features, classical approaches can still work well, while principled feature selection helps in many cases.
- **High-dimensional data**: 10^3 to 10^9 or more features. Examples: micro-array / gene expression data and text categorization (bag-of-words features). If we also have few observations, scenario is called $p \gg n$.

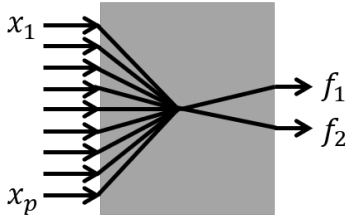
FEATURE SELECTION VS. EXTRACTION

Feature selection



- Creates a subset of original features **x** by selecting $\tilde{p} < p$ features ***f***.
- Retains information on selected individual features.

Feature extraction



- Maps p features in **x** to $\tilde{p}$ extracted features ***f***.
- Info on individual features can be lost through (non-)linear combination.

REASONS FOR FEATURE SELECTION

- improved predictive performance, since we reduce overfitting on irrelevant features,
- robust models that do not rely on noisy features,
- simpler models that are easier to interpret,
- faster model fitting, e.g. for model updates,
- faster prediction, and
- no need to collect potentially expensive features.

GOALS OF FEATURE SELECTION

Single Feature Selection Problem: identifying a minimal-size subset of the variables that is optimally predictive for an outcome variable T of interest.

Multiple Feature Selection Problem: Let $\mathcal{S}$ be the solution to the single feature selection problem (called the reference solution). The solution $\mathcal{M}$ to the multiple feature selection problem consists of all minimal-size sets $\mathcal{S}_i \in \mathcal{F}$ that are statistically equivalent to $\mathcal{S}$.

- Important for knowledge discovery (know all possible solutions)
- We will not delve into this any further

TYPES OF FEATURE SELECTION METHODS

In rest of the chapter, we introduce different types of methods for FS:

- Filters: evaluate relevance of features using statistical properties such as correlation with target variable
- Wrappers: use a model to evaluate subsets of features
- Embedded methods: integrate FS directly into specific model - we look at them in their dedicated chapters (e.g., CART, L_0, L_1)

Example: embedded method (Lasso) regularizing model params with $L1$ penalty enables “automatic" feature selection:

$$\mathcal{R}_{\text{reg}}(\boldsymbol{\theta}) = \mathcal{R}_{\text{emp}}(\boldsymbol{\theta}) + \lambda \|\boldsymbol{\theta}\|_1 = \sum_{i=1}^{n} \left(y^{(i)} - \boldsymbol{\theta}^\top \mathbf{x}\right)^2 + \lambda \sum_{j=1}^{p} |\theta_j|$$