Title: Feature engineering

URL: https://en.wikipedia.org/wiki/Feature_engineering

PageID: 46207323

Categories: Category:Data analysis, Category:Machine learning

-----

Supervised learning

Unsupervised learning

Semi-supervised learning

Self-supervised learning

Reinforcement learning

Meta-learning

Online learning

Batch learning

Curriculum learning

Rule-based learning

Neuro-symbolic AI

Neuromorphic engineering

Quantum machine learning

Classification

Generative modeling

Regression

Clustering

Dimensionality reduction

Density estimation

Anomaly detection

Data cleaning

AutoML

Association rules

Semantic analysis

Structured prediction

Feature engineering

Feature learning

Learning to rank

Grammar induction

Ontology learning

Multimodal learning

Apprenticeship learning

Decision trees

Ensembles Bagging Boosting Random forest

Bagging

Boosting

Random forest

k -NN

Linear regression

Naive Bayes

Artificial neural networks

Logistic regression

Perceptron

Relevance vector machine (RVM)

Support vector machine (SVM)

BIRCH

CURE

Hierarchical

k -means

Fuzzy

Expectation–maximization (EM)

DBSCAN

OPTICS

Mean shift

Factor analysis

CCA

ICA

LDA

NMF

PCA

PGD

t-SNE

SDL

Graphical models Bayes net Conditional random field Hidden Markov

Bayes net

Conditional random field

Hidden Markov

RANSAC

k -NN

Local outlier factor

Isolation forest

Autoencoder

Deep learning

Feedforward neural network

Recurrent neural network LSTM GRU ESN reservoir computing

LSTM

GRU

ESN

reservoir computing

Boltzmann machine Restricted

Restricted

GAN

Diffusion model

SOM

Convolutional neural network U-Net LeNet AlexNet DeepDream

U-Net

LeNet

AlexNet

DeepDream

Neural field Neural radiance field Physics-informed neural networks

Neural radiance field

Physics-informed neural networks

Transformer Vision

Vision

Mamba

Spiking neural network

Memtransistor

Electrochemical RAM (ECRAM)

Q-learning

Policy gradient

SARSA

Temporal difference (TD)

Multi-agent Self-play

Self-play

Active learning

Crowdsourcing

Human-in-the-loop

Mechanistic interpretability

RLHF

Coefficient of determination

Confusion matrix

Learning curve

ROC curve

Kernel machines

Bias–variance tradeoff

Computational learning theory

Empirical risk minimization

Occam learning

PAC learning

Statistical learning

VC theory

Topological deep learning

AAAI

ECML PKDD

NeurIPS

ICML

ICLR

IJCAI

ML

JMLR

Glossary of artificial intelligence

List of datasets for machine-learning research List of datasets in computer vision and image processing

List of datasets in computer vision and image processing

Outline of machine learning

v

t

e

Feature engineering is a preprocessing step in supervised machine learning and statistical modeling which transforms raw data into a more effective set of inputs. Each input comprises several attributes, known as features. By providing models with relevant information, feature engineering significantly enhances their predictive accuracy and decision-making capability.

Beyond machine learning, the principles of feature engineering are applied in various scientific fields, including physics. For example, physicists construct dimensionless numbers such as the Reynolds number in fluid dynamics , the Nusselt number in heat transfer , and the Archimedes number in sedimentation . They also develop first approximations of solutions, such as analytical solutions for the strength of materials in mechanics.

Clustering

One of the applications of feature engineering has been clustering of feature-objects or sample-objects in a dataset. Especially, feature engineering based on matrix decomposition has been extensively used for data clustering under non-negativity constraints on the feature coefficients. These include Non-Negative Matrix Factorization (NMF), Non-Negative Matrix-Tri Factorization (NMTF), Non-Negative Tensor Decomposition/Factorization (NTF/NTD), etc. The non-negativity constraints on coefficients of the feature vectors mined by the above-stated algorithms yields a part-based representation, and different factor matrices exhibit natural clustering properties. Several extensions of the above-stated feature engineering methods have been reported in literature, including orthogonality-constrained factorization for hard clustering, and manifold learning to overcome inherent issues with these algorithms.

Other classes of feature engineering algorithms include leveraging a common hidden structure across multiple inter-related datasets to obtain a consensus (common) clustering scheme. An example is Multi-view Classification based on Consensus Matrix Decomposition (MCMD), which mines a common clustering scheme across multiple datasets. MCMD is designed to output two types of class labels (scale-variant and scale-invariant clustering), and:

is computationally robust to missing information,

can obtain shape- and scale-based outliers,

and can handle high-dimensional data effectively.

Coupled matrix and tensor decompositions are popular in multi-view feature engineering.

## Predictive modelling

Feature engineering in machine learning and statistical modeling involves selecting, creating, transforming, and extracting data features. Key components include feature creation from existing data, transforming and imputing missing or invalid features, reducing data dimensionality through methods like Principal Components Analysis (PCA), Independent Component Analysis (ICA), and Linear Discriminant Analysis (LDA), and selecting the most relevant features for model training based on importance scores and correlation matrices .

Features vary in significance. Even relatively insignificant features may contribute to a model. Feature selection can reduce the number of features to prevent a model from becoming too specific to the training data set (overfitting).

Feature explosion occurs when the number of identified features is too large for effective model estimation or optimization. Common causes include:

Feature templates - implementing feature templates instead of coding new features

Feature combinations - combinations that cannot be represented by a linear system

Feature explosion can be limited via techniques such as: regularization , kernel methods , and feature selection .

## Automation

Automation of feature engineering is a research topic that dates back to the 1990s. Machine learning software that incorporates automated feature engineering has been commercially available since 2016. Related academic literature can be roughly separated into two types:

Multi-relational decision tree learning (MRDTL) uses a supervised algorithm that is similar to a decision tree .

Deep Feature Synthesis uses simpler methods. [ citation needed ]

### Multi-relational decision tree learning (MRDTL)

Multi-relational Decision Tree Learning (MRDTL) extends traditional decision tree methods to relational databases , handling complex data relationships across tables. It innovatively uses selection graphs as decision nodes , refined systematically until a specific termination criterion is reached.

Most MRDTL studies base implementations on relational databases, which results in many redundant operations. These redundancies can be reduced by using techniques such as tuple id propagation.

## Open-source implementations

There are a number of open-source libraries and tools that automate feature engineering on relational data and time series:

featuretools is a Python library for transforming time series and relational data into feature matrices for machine learning.

MCMD: An open-source feature engineering algorithm for joint clustering of multiple datasets .

OneBM or One-Button Machine combines feature transformations and feature selection on relational data with feature selection techniques. [OneBM] helps data scientists reduce data exploration time allowing them to try and error many ideas in short time. On the other hand, it enables non-experts, who are not familiar with data science, to quickly extract value from their data with a little effort, time, and cost.

[OneBM] helps data scientists reduce data exploration time allowing them to try and error many ideas in short time. On the other hand, it enables non-experts, who are not familiar with data science, to quickly extract value from their data with a little effort, time, and cost.

getML community is an open source tool for automated feature engineering on time series and relational data. It is implemented in C / C++ with a Python interface. It has been shown to be at least 60 times faster than tsflex, tsfresh, tsfel, featuretools or kats.

tsfresh is a Python library for feature extraction on time series data. It evaluates the quality of the features using hypothesis testing.

tsflex is an open source Python library for extracting features from time series data. Despite being 100% written in Python, it has been shown to be faster and more memory efficient than tsfresh, seglearn or tsfel.

seglearn is an extension for multivariate, sequential time series data to the scikit-learn Python library.

tsfel is a Python package for feature extraction on time series data.

kats is a Python toolkit for analyzing time series data.

## Deep feature synthesis

The deep feature synthesis (DFS) algorithm beat 615 of 906 human teams in a competition.

## Feature stores

The feature store is where the features are stored and organized for the explicit purpose of being used to either train models (by data scientists) or make predictions (by applications that have a trained model). It is a central location where you can either create or update groups of features created from multiple different data sources, or create and update new datasets from those feature groups for training models or for use in applications that do not want to compute the features but just retrieve them when it needs them to make predictions.

A feature store includes the ability to store code used to generate features, apply the code to raw data, and serve those features to models upon request. Useful capabilities include feature versioning and policies governing the circumstances under which features can be used.

Feature stores can be standalone software tools or built into machine learning platforms.

## Alternatives

Feature engineering can be a time-consuming and error-prone process, as it requires domain expertise and often involves trial and error. Deep learning algorithms may be used to process a large raw dataset without having to resort to feature engineering. However, deep learning algorithms still require careful preprocessing and cleaning of the input data. In addition, choosing the right architecture, hyperparameters, and optimization algorithm for a deep neural network can be a challenging and iterative process.

## See also

Covariate

Data transformation

Feature extraction

Feature learning

Hashing trick

Instrumental variables estimation

Kernel method

List of datasets for machine learning research

Scale co-occurrence matrix

Space mapping

## References

## Further reading

Boehmke B, Greenwell B (2019). "Feature & Target Engineering". Hands-On Machine Learning with R . Chapman & Hall. pp. 41– 75. ISBN 978-1-138-49568-5 .

Zheng A, Casari A (2018). Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists . O'Reilly. ISBN 978-1-4919-5324-2 .

Zumel N, Mount (2020). "Data Engineering and Data Shaping". Practical Data Science with R (2nd ed.). Manning. pp. 113– 160. ISBN 978-1-61729-587-4 .

Abououf, M., Singh, S., Mizouni, R., Otrok, H. (2024), "Feature engineering and deep learning-based approach for event detection in Medical Internet of Things (MIoT)" , Internet of Things , 26 101191, Elsevier BV, doi : 10.1016/j.iot.2024.101191

Chicco D, Oneto L, Tavazzi E (December 2022). "Eleven quick tips for data cleaning and feature engineering" . PLOS Computational Biology . 18 (12): e1010718. Bibcode : 2022PLSCB..18E0718C . doi : 10.1371/journal.pcbi.1010718 . PMC 9754225 . PMID 36520712 . S2CID 254733288 . {{ cite journal }} : CS1 maint: article number as page number ( link )