Feature Engineering

Prior Knowledge Simplifies Your Model

The more relevant prior knowledge you have, the simpler the model can be.

Applications

Prior knowledge, such as domain knowledge, can be used to

- Define the problem more clearly
- Filter out unnecessary features
- Simplify feature engineering, e.g., combining power and time into total energy used
- Locate anomalies

Encoding

Encode the features into numerical values for the model to process.

Methods

Categorical Data Encoding

- Binary Encoding
- One-hot Encoding
- Numerical Encoding

Datetime

- **Disintegration**
  Using year, month, hour, second, etc as features

Feature Crossing

Introduce higher order features to make the model more linearly separatable.

Methods

- Create x^2, x^3 from x

Scaling

Scale the data to different ranges.

Methods

- Rescale Based on Location and Spread
- **MinMax**
  Scale data into a specific range

Combining Features

Combine several features into one so that the new feature bears more relevant information.

Sparse Categorical Data

Some categorical data values do not have a large number of counts. Combining these low count values into one might be helpful.

Normalization

For example, if a feature has a very high variance and we are working on a clustering method, it is easier if we normalize the data, e.g., log.

Using Statistical Results as Features

Methods

- Use the Average of Several Features

Extract Values from Texts

Methods

- **TFIDF**
  For simple text manipulations, one could use TFIDF to extract some important words as features.

Location, Variability, Skewness and Kurtosis

Methods

Fix the Skewness

- Box Cox transform

Feature Selection

Remove Redundant Features

What are Redundant Features

- Noisy features
- Features that are highly correlated to or duplicate of some other features

Methods

- **Only Include Useful Features**
  Feature selection using domain knowledge, or feature selection algorithms.
- Remove High Correlated Features