

## Feature Engineering

Made Easy in 3 Methods







#### Feature Engineering: Method 1/3

# Feature Transformation

Feature transformation involves modifying or converting the existing features to enhance their usefulness for analysis.

- Data Cleaning: Ensure data quality by handling missing values, outliers, and inconsistencies.
- Feature Scaling: Normalize numerical features to a common scale (e.g., using techniques like Min-Max scaling or Standardization).
  - Feature Encoding: Convert categorical variables into numerical representations (e.g., one-hot encoding, label encoding).



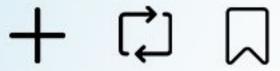


#### Feature Engineering: Method 2/3

### Feature Selection

Feature selection is the process of choosing a subset of the original pool of features that are most relevant and informative.

- Backward Feature Elimination (BFE): Starts training the model with all the features then removes one at a time to see how it affects the model's performance. Identifies the most important features by eliminating the less useful ones.
- Forward Feature Selection (FFS): Contrary to BFE, It adds one feature
   at a time to see how it improves the model's performance.
   Helps finding the most relevant features by including only the
   ones that contribute the most.
  - Random Forest Regressor: A Machine Learning algorithm that selects important features by building decision trees on different subsets of features.





## Feature Extraction

Feature extraction involves deriving useful and meaningful features from existing data.

- Principal Component Analysis (PCA): Transform
  high-dimensional data into a lower-dimensional
  representation while preserving important information.
- Featuretools: Automated feature engineering library that extracts insightful features from structured and time-series data.
- Simple Statistical Aggregations: Compute summary statistics (e.g., mean, median, standard deviation) for numerical variables or derive new features from existing ones.

