Often, feature selection and dimensionality reduction are grouped together (like here in this article). While both methods are used for reducing the number of features in a dataset, there is an important difference.Feature selection is simply selecting and excluding given features **without changing** them.Dimensionality reduction **transforms** features into a lower dimension.

Feature Selection

- Remove features with missing values
- Remove features with low variance
- Remove highly correlated features
- Univariate feature selection
- Recursive feature elimination
- Feature selection using SelectFromModel

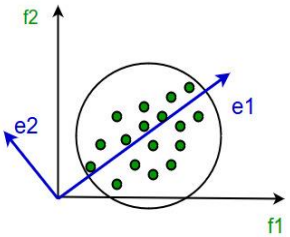
Dimensionality Reduction

- Principal Component Analysis (PCA)
- Linear Discriminant Analysis (LDA)
- Generalized Discriminant Analysis (GDA)

Dimensionality reduction may be both linear or non-linear, depending upon the method used. The prime linear method, called Principal Component Analysis, or PCA, is discussed below.

Principal Component Analysis

This method was introduced by Karl Pearson. It works on a condition that while the data in a higher dimensional space is mapped to data in a lower dimension space, the variance of the data in the lower dimensional space should be maximum.



It involves the following steps:

- Construct the covariance matrix of the data.
- Compute the eigenvectors of this matrix.
- Eigenvectors corresponding to the largest eigenvalues are used to reconstruct a large fraction of variance of the original data.

Hence, we are left with a lesser number of eigenvectors, and there might have been some data loss in the process. But, the most important variances should be retained by the remaining eigenvectors.

Advantages of Dimensionality Reduction

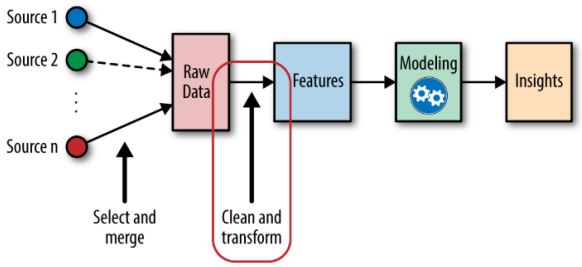
- It helps in data compression, and hence reduced storage space.
- It reduces computation time.
- It also helps remove redundant features, if any.

Disadvantages of Dimensionality Reduction

- It may lead to some amount of data loss.
- PCA tends to find linear correlations between variables, which is sometimes undesirable.
- PCA fails in cases where mean and covariance are not enough to define datasets.
- We may not know how many principal components to keep- in practice, some thumb rules are applied

Feature Engineering Definition

Feature engineering refers to the process of using domain knowledge to select and transform the most relevant variables from raw data when creating a predictive model using machine learning or statistical modeling. The goal of feature engineering and selection is to improve the performance of machine learning (ML) algorithms.



FAQs

What is Feature Engineering?

The feature engineering pipeline is the preprocessing steps that transform raw data into features that can be used in machine learning algorithms, such as predictive models. Predictive models consist of an outcome variable and predictor variables, and it is during the feature

engineering process that the most useful predictor variables are created and selected for the predictive model. Automated feature engineering has been available in some machine learning software since 2016. Feature engineering in ML consists of four main steps: Feature Creation, Transformations, Feature Extraction, and Feature Selection.

Feature engineering consists of creation, transformation, extraction, and selection of features, also known as variables, that are most conducive to creating an accurate ML algorithm. These processes entail:

- Feature Creation: Creating features involves identifying the variables that will be most useful in the predictive model. This is a subjective process that requires human intervention and creativity. Existing features are mixed via addition, subtraction, multiplication, and ratio to create new derived features that have greater predictive power.
- Transformations: Transformation involves manipulating the predictor variables to improve model performance; e.g. ensuring the model is flexible in the variety of data it can ingest; ensuring variables are on the same scale, making the model easier to understand; improving accuracy; and avoiding computational errors by ensuring all features are within an acceptable range for the model.
- Feature Extraction: Feature extraction is the automatic creation of new variables by extracting them from raw data. The purpose of this step is to automatically reduce the volume of data into a more manageable set for modeling. Some feature extraction methods include cluster analysis, text analytics, edge detection algorithms, and principal components analysis.
- Feature Selection: Feature selection algorithms essentially analyze, judge, and rank various features to determine which features are irrelevant and should be removed, which features are redundant and should be removed, and which features are most useful for the model and should be prioritized.

Steps in Feature Engineering

The art of feature engineering may vary among data scientists, however steps for how to perform feature engineering for most machine learning algorithms include the following:

- Data Preparation: This preprocessing step involves the manipulation and consolidation of raw data from different sources into a standardized format so that it can be used in a model. Data preparation may entail data augmentation, cleaning, delivery, fusion, ingestion, and/or loading.
- Exploratory Analysis: This step is used to identify and summarize the main characteristics in a data set through data analysis and investigation. Data science experts use data visualizations to better understand how best to manipulate data

sources, to determine which statistical techniques are most appropriate for data analysis, and for choosing the right features for a model.

- Benchmark: Benchmarking is setting a baseline standard for accuracy to which all variables are compared. This is done to reduce the rate of error and improve a model's predictability. Experimentation, testing and optimizing metrics for benchmarking is performed by data scientists with domain expertise and business users.

Examples of Feature Engineering

Feature engineering determines the success of failure of a predictive model, and determines how comprehensible the model will be to humans. Advanced feature engineering is at the heart of the Titanic Competition, a popular feature engineering example developed by Kaggle Fundamentals, an online community of data scientists and subsidiary of Google LLC. This project challenges competitors to predict which passengers survived the sinking of the Titanic.

Each Kaggle competition provides a training data set to train the predictive model, and a testing data set to work with. The Titanic Competition also provides information about passengers onboard the Titan