Feature Selection and Feature Engineering for dimensionality reduction

Dimensionality reduction could be done by both feature selection methods as well as feature engineering methods.

Feature selection is the process of identifying and selecting relevant features for your sample.

Feature engineering is manually generating new features from existing features, by applying some transformation or performing some operation on them.

There are a lot of tools could employ to aid feature selection.

1. **Heatmaps that show the correlation between features is a good idea.**
2. **Just visualizing the relationship between the features and the target variable by plotting each feature against the target variable.**

Few programmatic methods for feature selection from the popular machine learning library sci-kit learn, namely,

1. Variance Threshold and
2. Univariate selection.

Variance Threshold is a baseline approach to feature selection. As the name suggests, it drops all features where the variance along the column does not exceed a threshold value. The premise is that a feature which doesn't vary much within itself, has very little predictive power.

Univariate Feature Selection uses statistical tests to select features.

- Univariate describes a type of data which consists of observations on only a single characteristic or attribute.
- Univariate feature selection examines each feature individually to determine the strength of the relationship of the feature with the response variable.
- Some examples of statistical tests that can be used to evaluate feature relevance are **Pearson Correlation, Maximal information coefficient, Distance correlation, ANOVA and Chi-square**. Chi-square is used to find the relationship between categorical variables and Anova is preferred when the variables are continuous.

Scikit-learn exposes feature selection routines likes
SelectKBest,
SelectPercentile or
GenericUnivariateSelect

- as objects that implement a transform method based on the score of anova or chi2 or mutual information.

Sklearn offers
f_regression
mutual_info_regression

as the scoring functions for regression and
f_classif and
mutual_info_classif for classification.

feature selection techniques

1. Filter Methods
2. Wrapper Methods

Filter Methods

Filter Methods considers the relationship between features and the target variable to compute the importance of features.

Scikit learn provides the **Selecting K best** features using F-Test.

sklearn.feature_selection.f_regression

For Classification tasks

sklearn.feature_selection.f_classif

Sklearn offers feature selection with Mutual Information for regression and classification tasks.

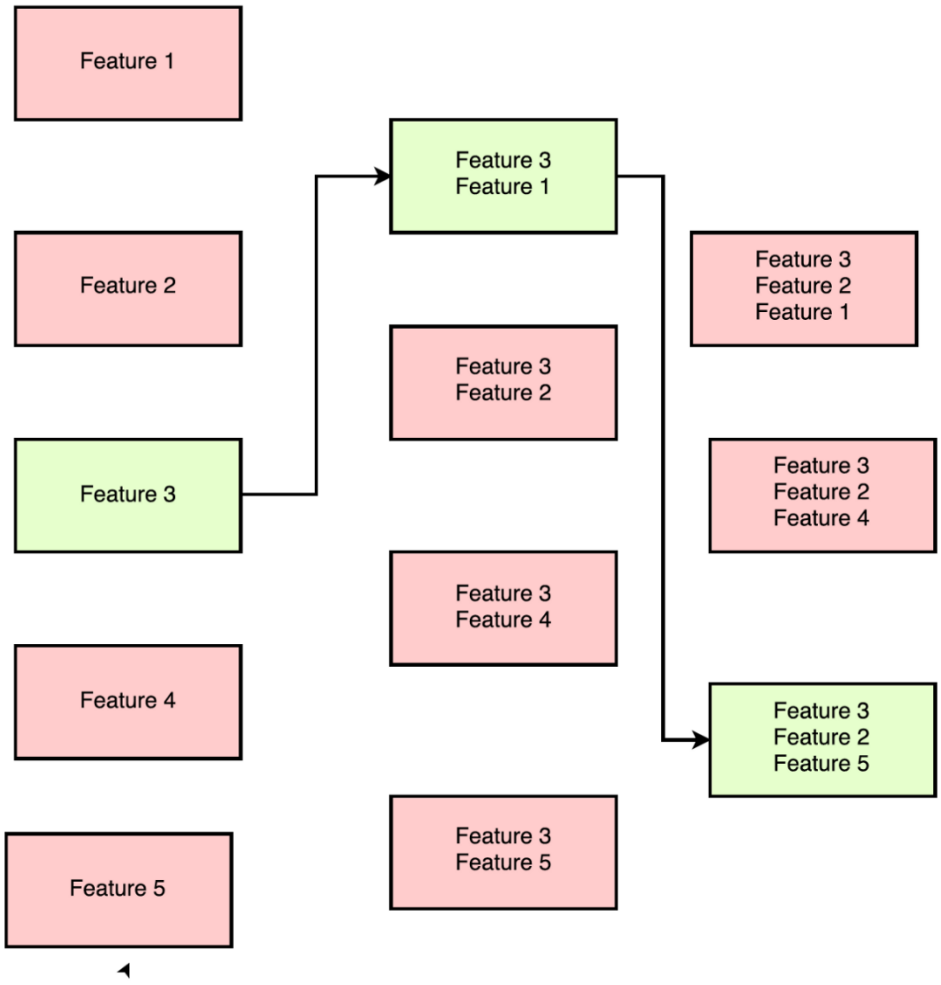
sklearn.feature_selection.mututal_info_regression
sklearn.feature_selection.mututal_info_classif

Wrapper Methods

Wrapper Methods generate models with a subsets of feature and gauge their model performances.

Forward Search

This method allows you to search for the best feature w.r.t model performance and add them to your feature subset one after the other.



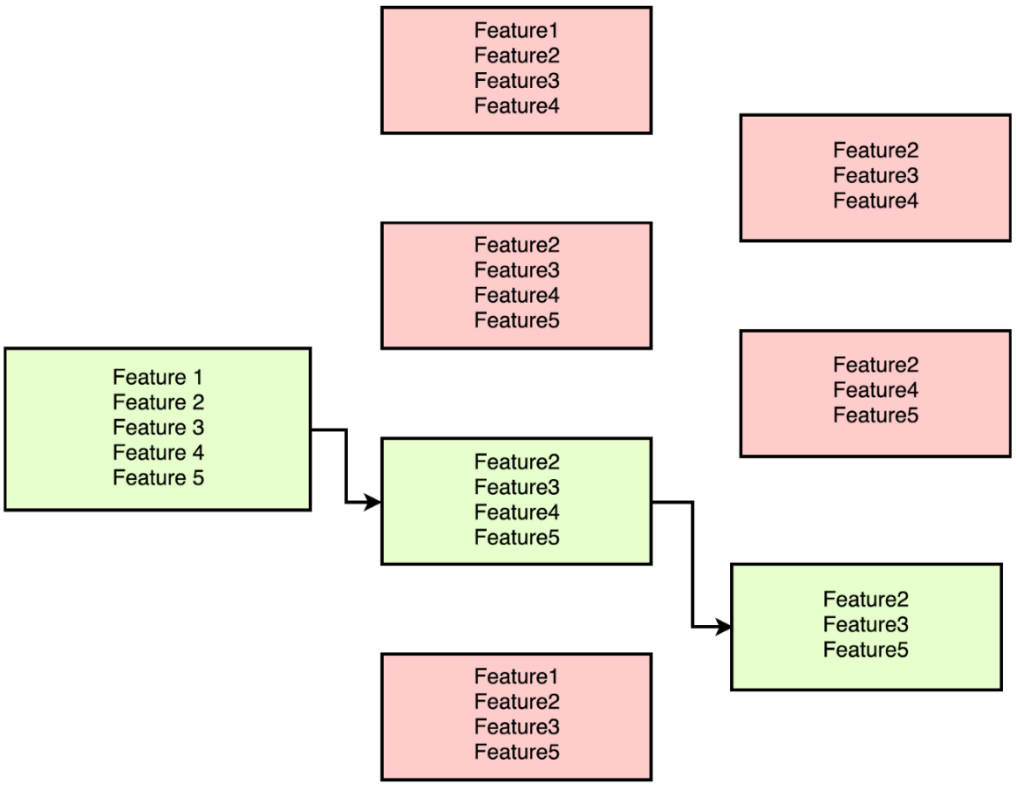
For data with n features,

->On first round ‘n’ models are created with individual feature and the best predictive feature is selected.

->On second round, ‘n-1’ models are created with each feature and the previously selected feature.

->This is repeated till a best subset of ‘m’ features are selected.

Recursive Feature Elimination



For data with n features,

->On first round ‘n-1’ models are created with combination of all features except one. The least performing feature is removed

-> On second round ‘n-2’ models are created by removing another feature.

Wrapper Methods promises you a best set of features with a extensive greedy search.

But the main drawbacks of wrapper methods is the sheer amount of models that needs to be trained. It is computationally very expensive and is infeasible with large number of features.