VICTORIA UNIVERSITY OF WELLINGTON
TE HERENGA WAKA

AIML 231/DATA 302 — Week 6

Dimensionality Reduction and Feature Selection

Dr Bach Hoai Nguyen

School of Engineering and Computer Science

Victoria University of Wellington

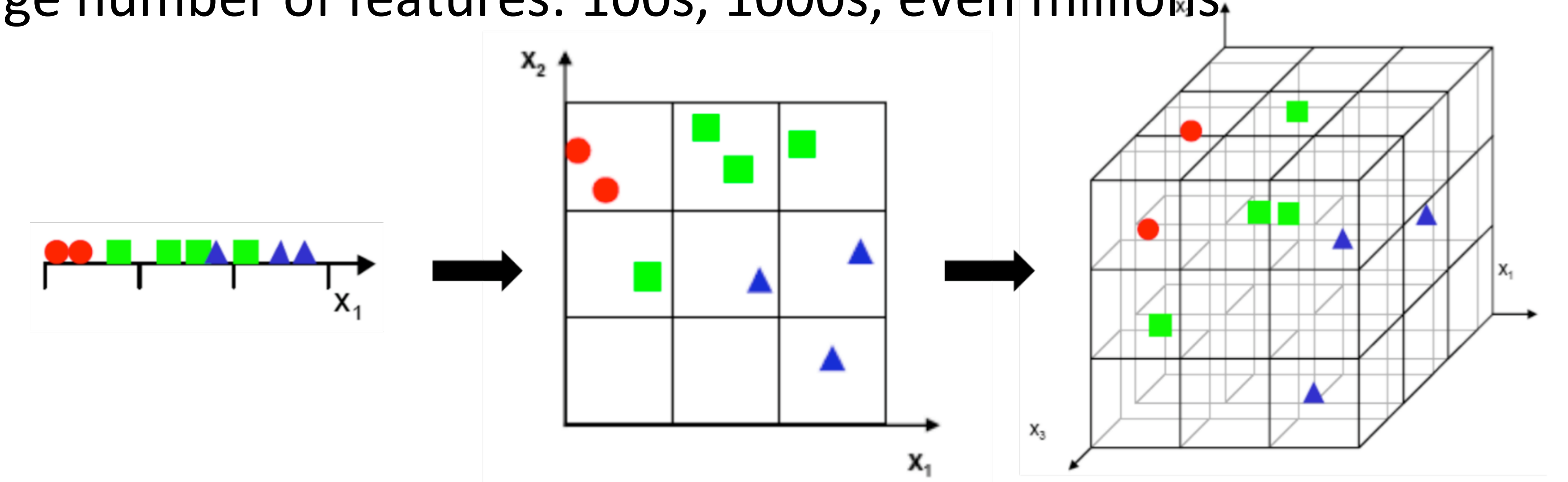
Bach.Nguyen@vuw.ac.nz

Week Overview

- Introduction of Data Preparation/Preprocessing
  - What data preparation include
  - Why data preparation
  - Types of data preparation techniques
- Data Preparation Techniques
  - Training vs Testing, k-fold cross validation
  - Categorical Data Encoding
  - Normalisation
  - Discretisation
- Dimensionality Reduction
  - Feature Selection
  - Feature Construction

Why Dimensionality Reduction?

- **"Curse of dimensionality"**
  - Large number of features: 100s, 1000s, even millions



3 samples per region 1 sample per region 1/3 sample per region

  - Data density decreases exponentially with dimensionality ☹

- Irrelevant features: no information for learning task
- Redundant features: same information as other features
- time, memory, and money

Feature Selection and Feature Construction

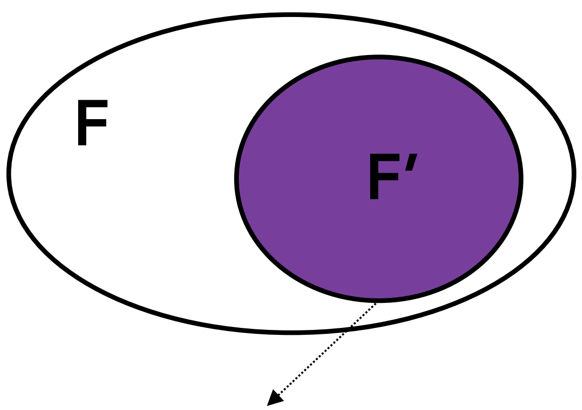
Feature selection (FS)

- Select a subset of relevant features to achieve similar or better performance than using all features

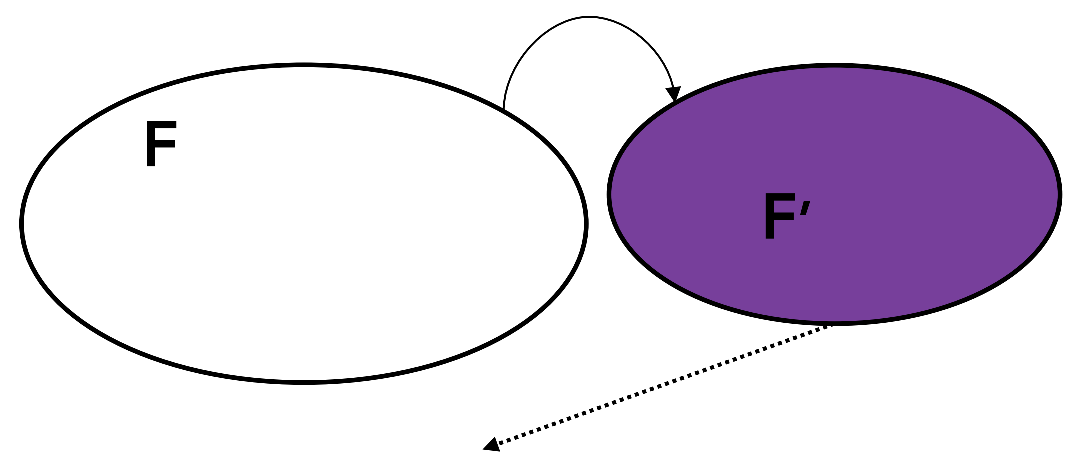
Feature construction (FC)

- Build new high-level features using the original features to improve the classification performance

$\mathbf{F} = \{f_1, f_2, \ldots, f_n\}$



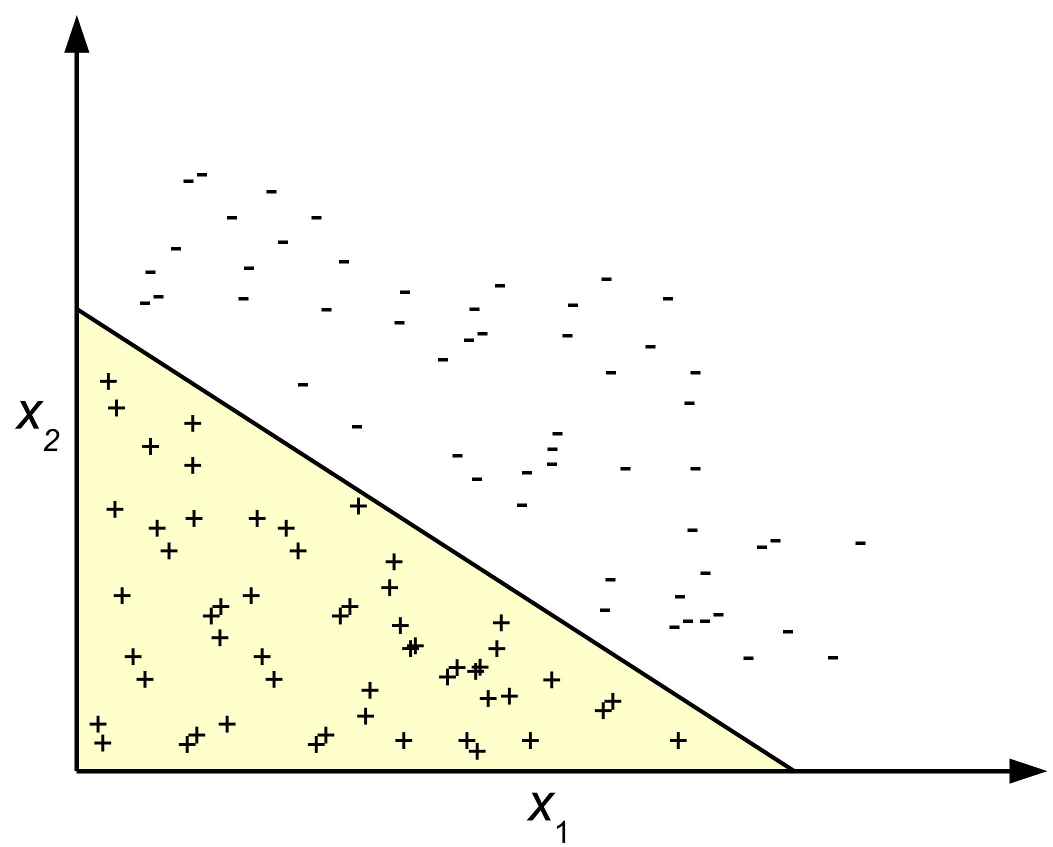
$\mathbf{F'} = \{f_{i1}, f_{i2}, \ldots, f_{im}\}$
$(m<n)$



$\mathbf{F'} = \{g_1(f_1, \ldots, f_n), g_2(f_1, \ldots, f_n), \ldots, g_m(f_1, \ldots, f_n)\}$

Feature Selection Challenges

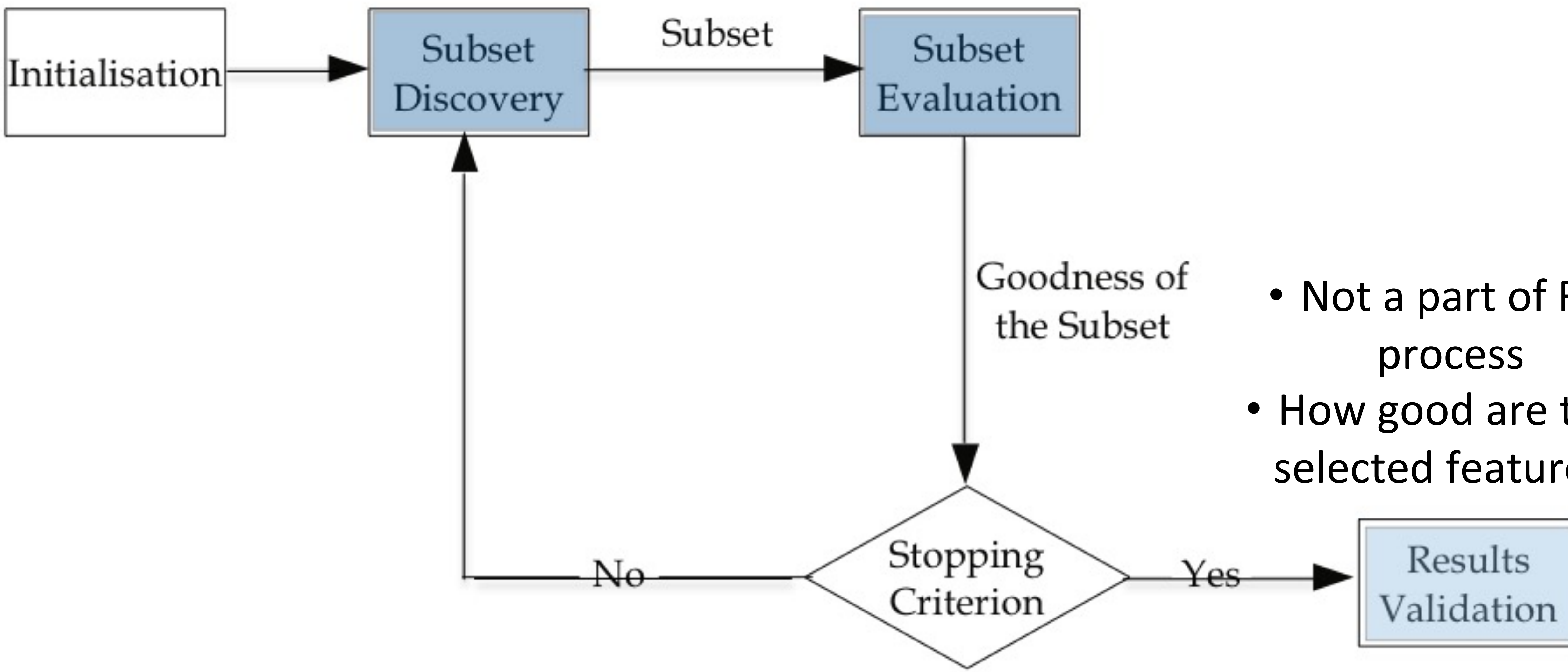
- Large (exponential) search space (2^n *possible feature subsets*– *n* is the number of features)
- Complex feature interactions:
  - Top relevant features can be redundant as they provide the same information about the class label
  - Weakly relevant features can be strongly relevant together – complementary features



Overall FS System

Generate new feature subsets

Evaluate the goodness of the generated subsets



- Not a part of FS process
- How good are the selected features?

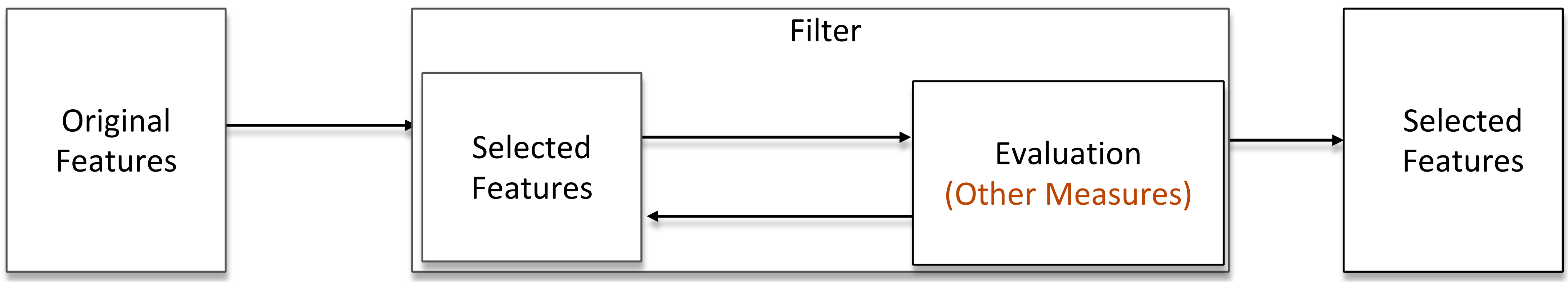
- Number of iterations
- Number of features
- ...

Filter, Wrapper and Embedded FS (1)

- Based on the evaluation component

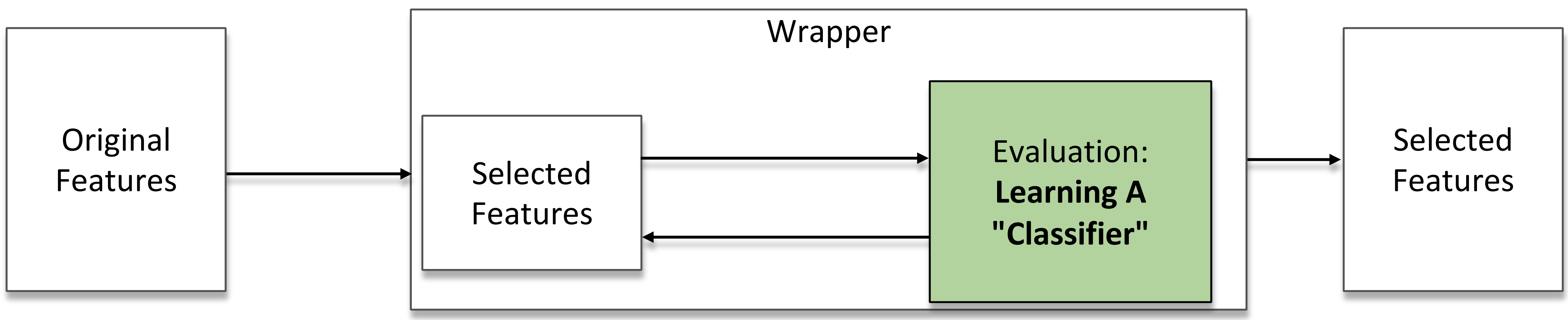
Filter

Uses existing measures, no learning algorithm



Wrapper

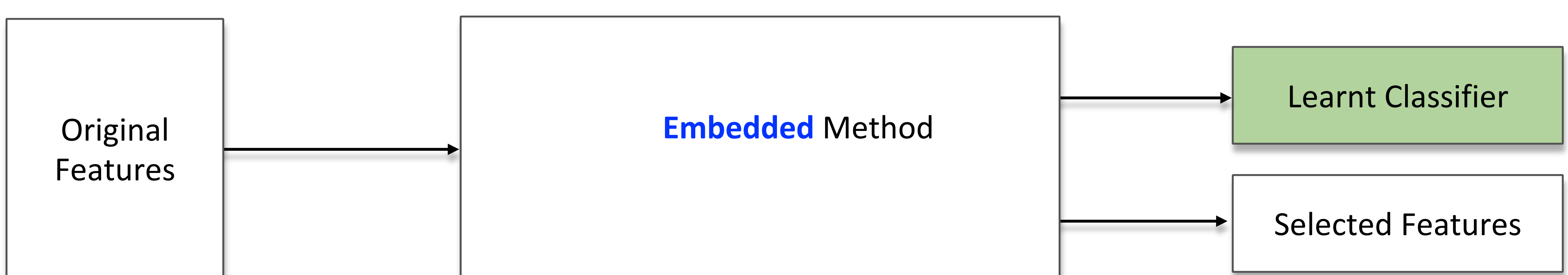
Uses learning performance, train ML models **many times**



Embedded

Train ML model **once**

Select features based on the learned model



Filter, Wrapper and Embedded FS (2)

- Filter FS: can use the following metrics to calculate the relevance between features and labels
  - Mutual information
  - Pearson correlation
  - Relief score

- Embedded FS examples:
  - Decision tree: features used in the tree are considered selected features
  - Linear SVM: features with larger weights are considered important features

Filter, Wrapper and Embedded FS (3)

- Wrapper FS: can use any classification algorithm to evaluate the feature subset
  - Divide the training set into sub-training set and sub-test set
  - Train the "wrapped" classification algorithm on the sub-training set
  - Examine the obtained classifier on the sub-test set
  - The performance on the sub-test set can be used as the quality of the feature subsets
  - Can apply K-fold cross validation to split the training set but it is more time-consuming

Feature Ranking vs Subset Selection

- Based on the Search Mechanism

- **Feature ranking**:
  - Evaluate features individually
  - Rank features and select top-ranked features
  - Simple, efficient
  - Ignore feature interactions (can select redundant features)

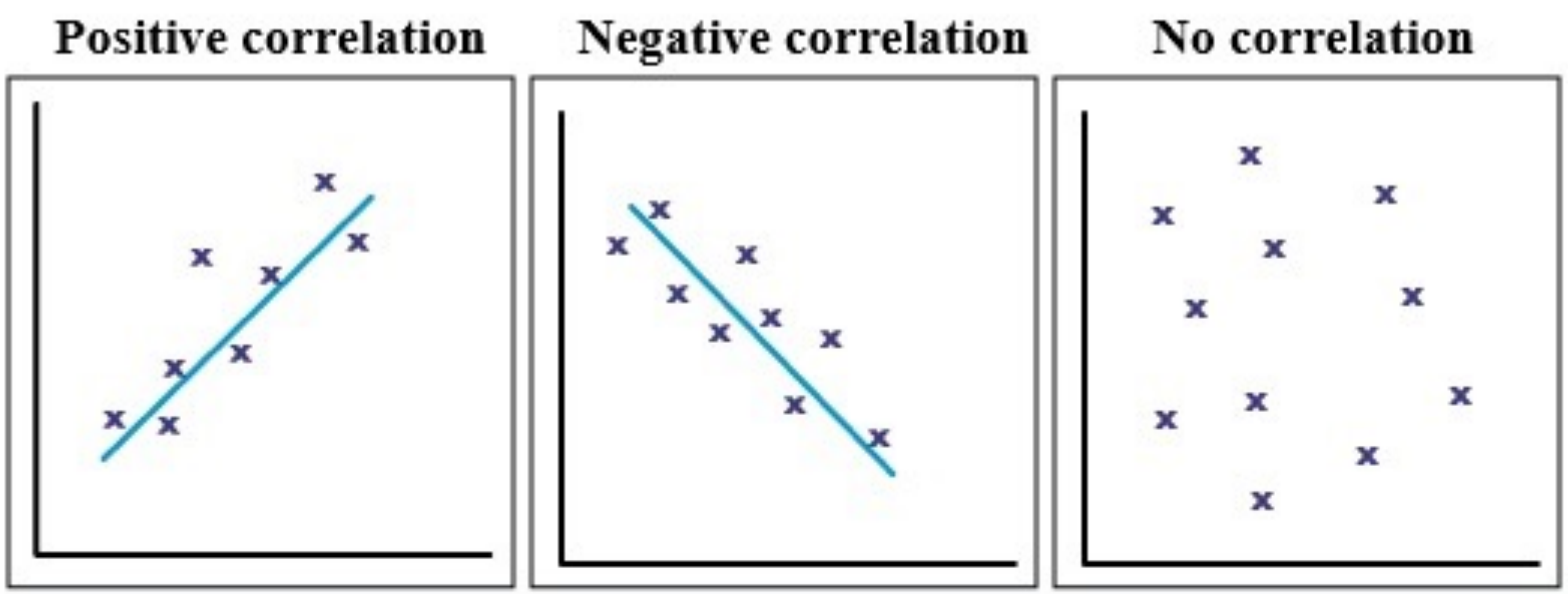
- **Feature subset selection:**
  - Evaluate the whole feature subset
  - Often an iterative process to improve the feature subset
  - Sequential feature selection is an example
  - Consider feature interactions, usually better performance
  - More complicated search, usually more expensive than ranking

Univariate FS - Correlation based methods

Univariate methods measure correlation between each input feature and the target variable/class label

- Pearson correlation: between -1 and 1
- Two variables move in the same direction/opposite directions, then have a positive correlation/negative correlation
- Rank features based on the absolute values of feature correlation
- The higher the correlation, the better the feature

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{(n-1)s_x s_y}$$



- sklearn.feature_selection.r_regression

Some Other Filter Measures for FS (1)

- Mutual information - measures the reduction in uncertainty for one variable given a known value of the other variable, measures mutual dependency
- I(X; Y) measures the common information between two X and Y.
- Rank features based on the mutual information between each feature and the class label
- The higher the mutual information, the more relevant the feature

- Use sklearn.feature_selection.mutual_info_classif

  sklearn.feature_selection.mutual_info_regression

Some Other Filter Measures for FS (2)

- Spearman: for continuous features/variable, nonlinear correlation, use scipy.stats.spearmanr
- ANOVA: between continuous feature and discreate label

  use sklearn.feature_selection.f_classif

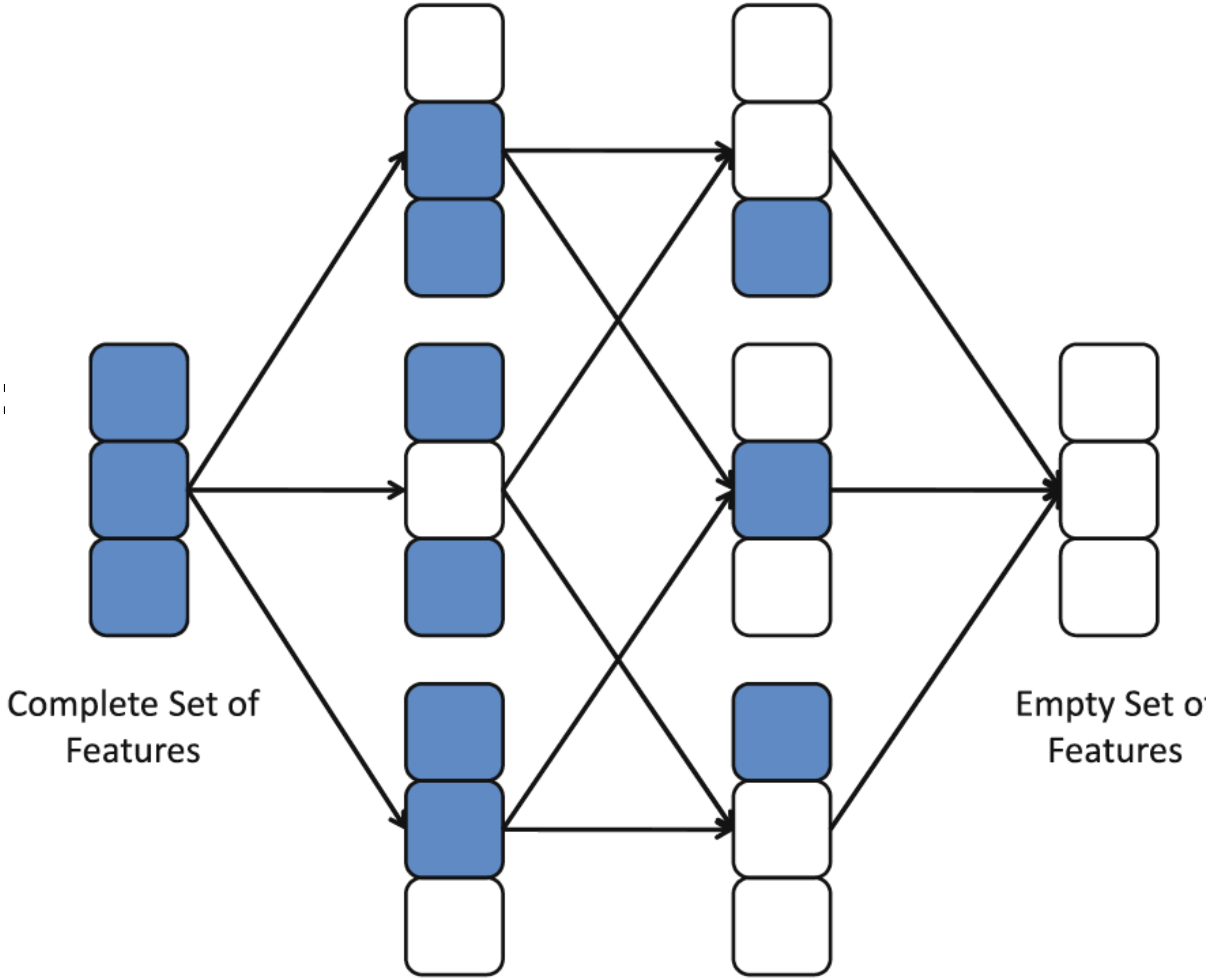
Subset Selection: Sequential Search

- Sequential Forward Feature Selection (SFFS):
  - starting from an empty set of features
  - sequentially add the feature X that results in the highest objective value when combined with the current set
  - stop when a pre-defined number of features is selected
  - works best when the optimal subset has a small number of features

- Sequential Backward Feature Selection (SBFS):
  - starting from the full set
  - sequentially remove the feature X that results in the highest objective value
  - stop when a pre-defined number of features is selected
  - works best when the optimal subset has a large number of features

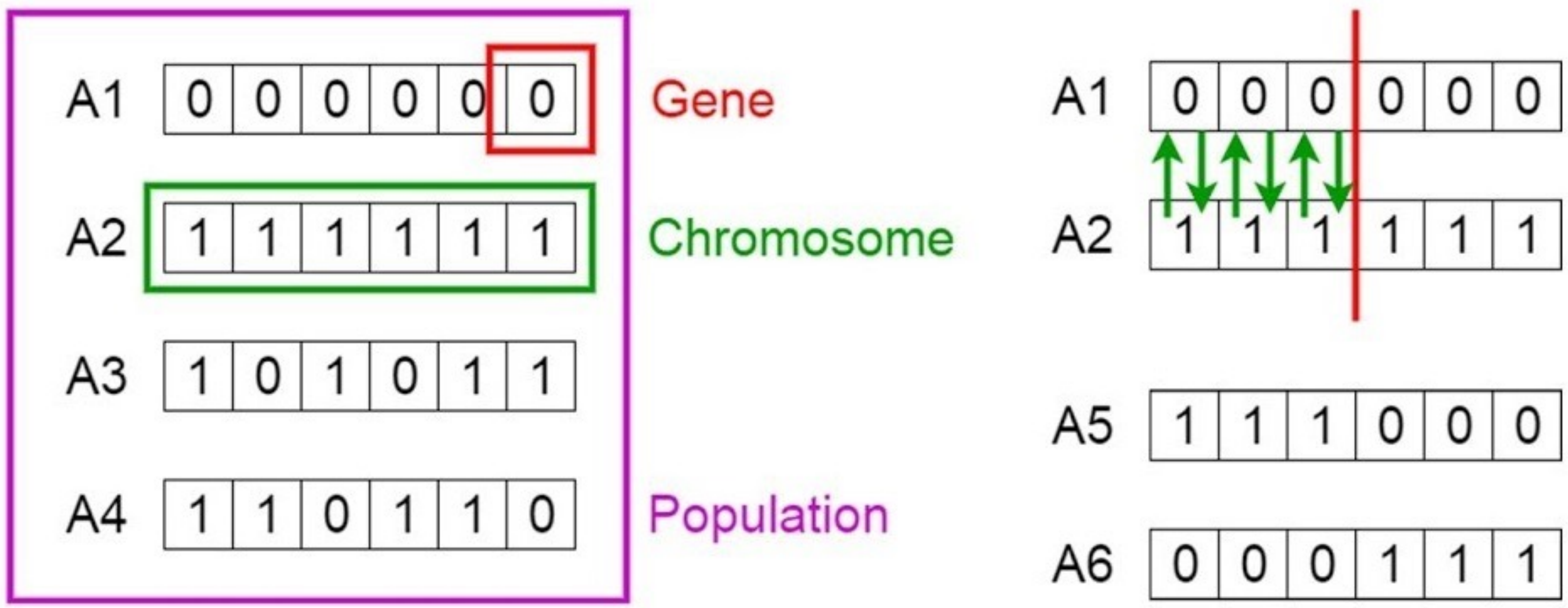
Subset Selection Illustration

Search Space:



More advanced FS Methods

- Genetic Algorithm for FS



- Particle Swarm Optimization for feature selection