Knowledge Discovery & Data Mining

Dimensionality Reduction: Feature Selection

Instructor: Yong Zhuang

yong.zhuang@gvsu.edu

Dimension Reduction

Dimension Reduction: It's a process that reduces the number of random variables under consideration by obtaining a set of principal variables that retain the most important information in the data while discarding the redundant or less important features.

Feature extraction: Transforms data into a set of new features.

- **Method:** PCA, Kernel PCA, Stochastic neighbor embedding, Autoencoders, ….
- **Advantages:** The newly derived features can capture essential information in fewer dimensions.

Feature selection: Selects a subset of the most relevant features for model construction.

- **Method:** Filter methods, wrapper methods, embedded methods.
- **Advantages:** Enhances model interpretability, discards irrelevant or redundant features.

Outline

- Feature selection
  - Filter
  - Wrapper
  - Embedded

Feature Selection

Feature selection, also known as **variable selection**, **attribute selection** or **variable subset selection**, is the process of selecting a subset of relevant features (variables, predictors) for use in model construction. Feature selection techniques are used for these reasons:

Removing / reduce effect of irrelevant data	Removing redundant data	Reducing dimensionality	Increasing learning accuracy	Improving result Interpretability

Feature Selection vs. Feature Extraction

Feature Selection	Feature Extraction
Selects a subset of relevant features from the original set of features.	Extracts a new set of features that are more informative and compact.
Reduces the dimensionality of the feature space and simplifies the model.	Captures the essential information from the original features and represents it in a lower-dimensional feature space.
Can be categorized into filter, wrapper, and embedded methods.	Can be categorized into linear and nonlinear methods.
Requires domain knowledge and feature engineering.	Can be applied to raw data without feature engineering.
Can improve the model's interpretability and reduce overfitting.	Can improve the model performance and handle nonlinear relationships.
May lose some information and introduce bias if the wrong features are selected.	May introduce some noise and redundancy if the extracted features are not informative.

Features Selection Methods

Filter	Wrapper	Embedded
Information gain	Forward selection	Regularization
Chi-square test	Backward elimination	Tree-based methods
Fisher score	Bi-directional elimination	
Correlation coefficient	Exhaustive selection	
Mutual information	Recursive feature elimination	
Variance threshold		

Filter Method

Filter methods select features from the dataset irrespective of the use of any predictive models. They are based only on general characteristics such as correlation with the feature to be predicted. Filtering methods suppress the least interesting features. Other features will be part of a classification or regression model used to classify or predict data. These methods are particularly efficient in terms of computation time and are robust to overfitting.

Filter Method: Fisher Score

Suppose we have a binary class label y (i.e., whether or not the customer will buy a computer). Intuitively, the feature x (e.g., income) is strongly correlated with the class label y if (1) the average income of all customers who buy a computer is significantly different from the average income of all customers who do not buy a computer, (2) all customers who buy a computer share similar income, and (3) all customers who do not buy a computer share similar income. Formally, Fisher score is defined as follows:

$$s = \frac{\sum_{j=1}^{c} n_j (\mu_j - \mu)^2}{\sum_{j=1}^{c} n_j \sigma_j^2}, \tag{7.1}$$

where c is the total number of classes ($c = 2$ in our example), n_j is the number of training tuples in class j, μ_j and σ_j^2 are the mean value and variance of feature x among all tuples that belong to class j, respectively, and μ is the mean value of feature x among all training tuples. Therefore a feature x would have a high Fisher score if the following conditions hold. First, the *class-specific mean* values $u_j (j = 1, ..., c)$ are dramatically different from each other (e.g., a large numerator of the Fisher score in Eq. (7.1)). Intuitively, this implies that on average, the feature values from different classes are quite different from each other. Second, the *class-specific variance* σ_j^2 is small (e.g., a small denominator of the Fisher score in Eq. (7.1)). This indicates that, within a class, different training tuples share similar feature values.

Filter Method: Fisher Score

Example. We are given 10 training tuples in Fig. 7.2(a), each of which is represented by two attributes (attribute A and attribute B) and a binary class label (+ vs. −). We want to use Fisher scores to decide which attribute is more correlated with the class label.

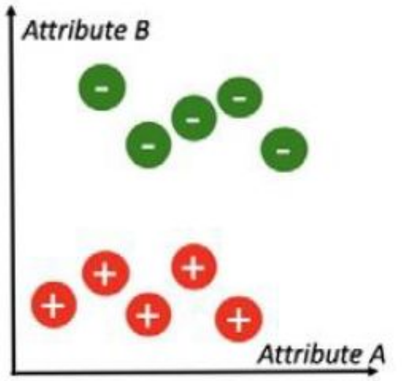
tuple index	1	2	3	4	5	6	7	8	9	10
Attribute A	1	2	3	4	5	2	3	4	5	6
Attribute B	1.0	0.9	0.8	1.2	1.1	5.1	4.8	4.9	5.2	5.0
Class label	+	+	+	+	+	−	−	−	−	−

(a) Training tuples

$$s = \frac{\sum_{j=1}^{c} n_j (\mu_j - \mu)^2}{\sum_{j=1}^{c} n_j \sigma_j^2}$$

$S_A = 0.125$

$S_B = 200$



(b) Scatter-plot

Filter Method: Variance Threshold

Variance Threshold (sklearn.feature_selection.VarianceThreshold) is a basic feature selection method that removes features whose variance falls below a certain threshold. By its default setting, it eliminates all features with zero variance, which means those features that have the same value across all samples. For instance, consider a dataset with boolean features; if we aim to discard features that are either always true (one) or false (zero) in more than 80% of the samples, we can utilize this method. The variance of boolean features, which are essentially Bernoulli random variables, is given by

$$Var(X) = p(1-p)$$

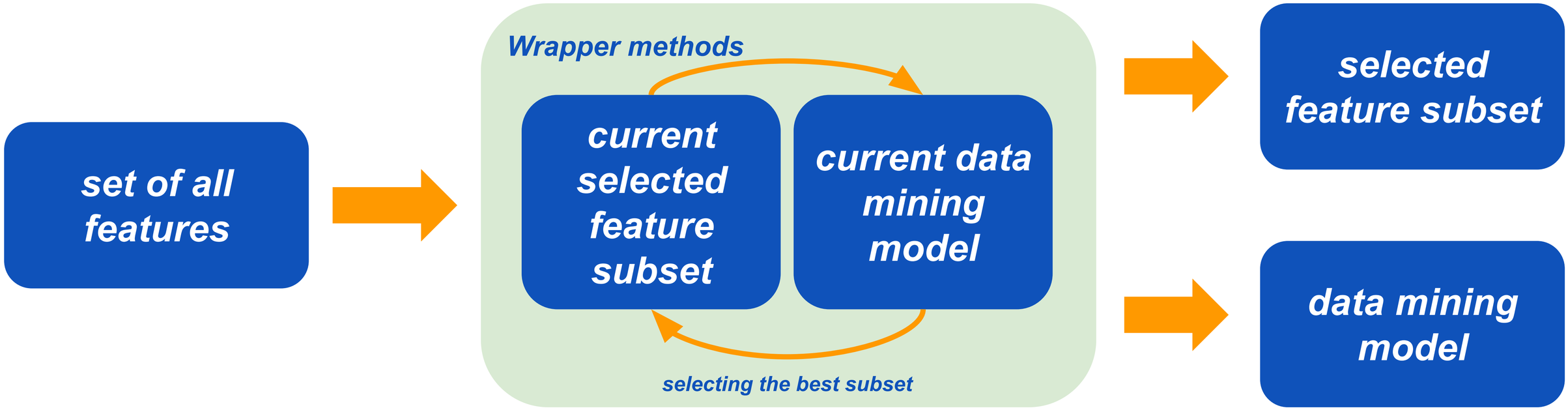
where p is the proportion of ones (or zeros).

- We often operate on the presumption that features with greater variance may contain more useful information.
- This method does not account for the relationship between different features or between features and target variables.

Wrapper Method

Wrapper methods, also referred to as greedy algorithms train the algorithm by using a subset of features in an iterative manner. Based on the conclusions made from training in prior to the model, addition and removal of features takes place.

The main advantage of wrapper methods over the filter methods is that they provide an optimal set of features for training the model, thus resulting in better accuracy than the filter methods but are computationally more expensive.



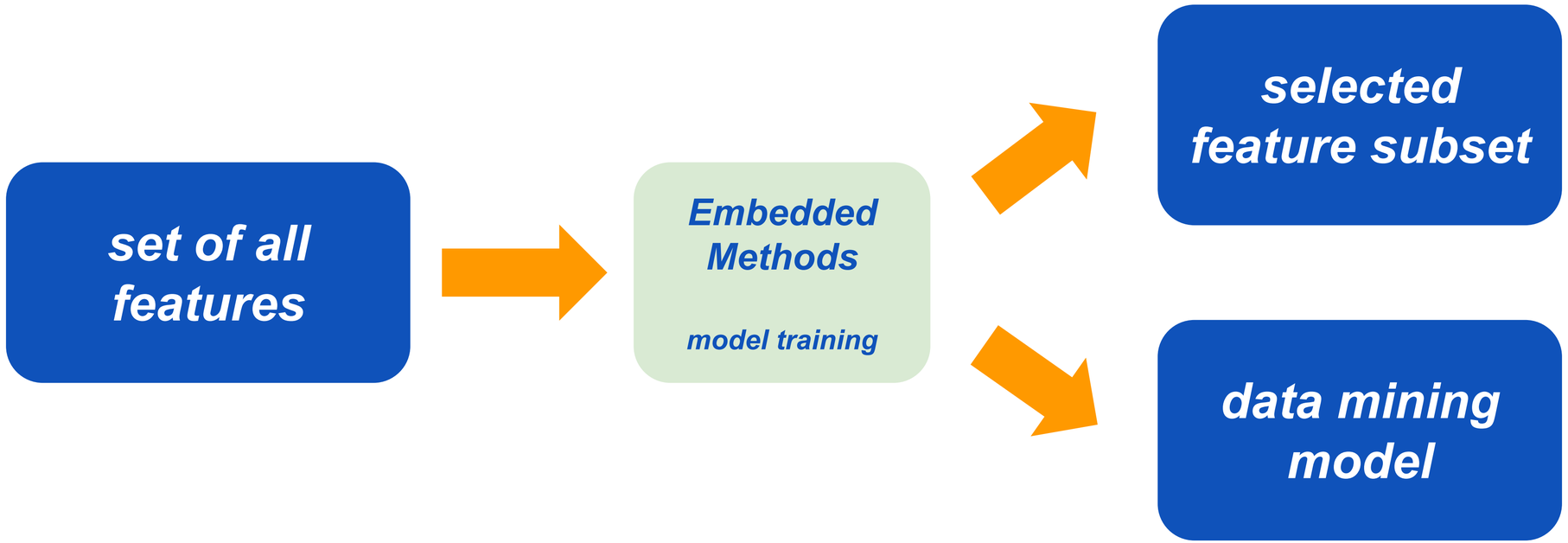
Wrapper Method: Forward Selection

Forward Selection (sklearn.feature_selection.SequentialFeatureSelector), a special case of sequential feature selection, is a greedy search algorithm that attempts to find the “optimal” feature subset by iteratively selecting features based on the classifier performance.

This method is an iterative approach where we initially start with an empty set of features and keep adding a feature which best improves our model after each iteration. The stopping criterion is till the addition of a new variable does not improve the performance of the model

Embedded Method

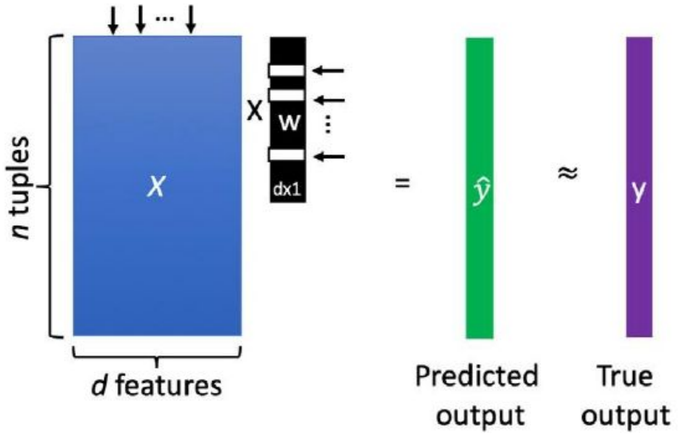
In **embedded methods**, the feature selection algorithm is blended as part of the learning algorithm, thus having its own built-in feature selection methods. Embedded methods encounter the drawbacks of filter and wrapper methods and merge their advantages. These methods are faster like those of filter methods and more accurate than the filter methods and take into consideration a combination of features as well.



Embedded Method: Lasso Regression

L1 Regularization, also known as Lasso Regression (sklearn.linear_model.Lasso), is a type of linear regression that includes a regularization term. The regularization term encourages simpler models by penalizing features with larger coefficient values.

$$\hat{L}(w) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda\|w\|_1 = \frac{1}{2}\sum_{i=1}^{n}(y_i - w^T x_i)^2 + \lambda\sum_{j=0}^{d}|w_j|$$



L1 Regularization

As λ increases, more coefficients fall to zero.

Summary

- Feature selection
  - Filter
  - Wrapper
  - Embedded