CS M146 - Week 3

Xinzhu Bei

xzbei@cs.ucla.edu

February 1, 2018

Overview

- Miscellaneous
  - Office Hour: 12pm – 2pm, Mon, ENG VI 386

Classification v.s. Regression

- **Classification**: In classification problems we are trying to predict a discrete number of values. The labels(y) generally comes in categorical form and represents a finite number of classes.
  Example:
  - Given an image correctly classify as containing Cats or Dogs.
  - From a given email predict whether its spam email or not.

  Types: Binary classification / Multi-Class Classification
  Algorithms: Decision Trees; Logistic Regression; K Nearest Neighbors; Linear SVC (Support vector Classifier), etc.

- **Regression**: In regression problems we trying to predict continuous valued output.
  Example: Given a size of the house predict the price(real value)
  Algorithms: Linear Regression, Regression Trees(e.g. Random Forest) Support Vector Regression (SVR), etc.

Parameter v.s. Hyperparmeter

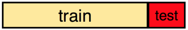
- A model **parameter** is a configuration variable that is internal to the model and whose value can be estimated from data.
  - They are required by the model when making predictions.
  - They values define the skill of the model on your problem.
  - They are estimated or learned from data.
  - They are often not set manually by the practitioner.
  - They are often saved as part of the learned model.

  Example: Attribute using to split a node in decision tree
- A model **hyperparameter** is a configuration that is external to the model and whose value cannot be estimated from data.
  - They are often used in processes to help estimate model parameters.
  - They are often specified by the practitioner.
  - They can often be set using heuristics.
  - They are often tuned for a given predictive modeling problem.

  Example: K-value in KNN.

Validation v.s. Cross Validation

Validation and Cross-Validation is used for finding the optimum hyper-parameters and thus to some extent prevent overfitting.

- **Validation** 

  We train multiple models with different hyperparameters with the help of the training set and then test the model with on the validation set. Those hyperparameters are chosen which give good performance on the validation set.

- **Cross Validation** 
  We have many models with different hyper-parameters and each model is trained against different combinations of these subsets and validated against the remaining parts. Hyperparameters giving the best results is used further. Final model is trained using these hyperparameters on the full training set.

- **Advantage** of using Cross-Validation is that we see that we are not wasting any data.

Linear Discriminant Functions

- A discriminant function based classifier is

$$y(X) = \begin{matrix} 1 & \text{if } g(X) > 0 \\ 0 & \text{Otherwise} \end{matrix} \quad (1)$$

  where g is called a discriminant function.
- A linear discriminant function is a linear combination of the components of x can be written as

$$g(x) = \mathbf{w}^T x + w_0$$

- Linear discriminant functions are going to be studied for the two-category case, multi-category case, and general case.

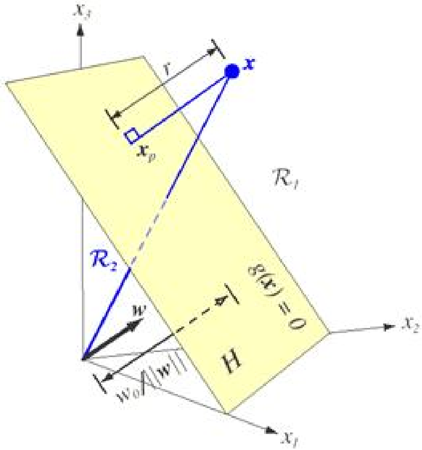
Two Category Case

- For a discriminant function, a two-category classifier implements the following decision rule: Decide Class1 if $g(x) > 0$ and Class2 if $g(x) < 0$.
- If x_1 and x_2 are both on the decision surface, then

$$\begin{aligned} \mathbf{w}^T\mathbf{x}_1 + w_0 &= \mathbf{w}^T\mathbf{x}_2 + w_0 \\ \text{or } \mathbf{w}^T(\mathbf{x}_1 - \mathbf{x}_2) &= 0 \end{aligned} \tag{2}$$

  and this shows that **w** is normal to any vector lying in the hyperplane.
- In general, the hyperplane H divides the feature space into two half-spaces: decision region $R1$ for Class1 and region $R2$ for Class2. Because $g(\mathbf{x}) > 0$ if $\mathbf{x}$ is in $R1$, it follows that the normal vector $\mathbf{w}$ points into $R1$.

Two Category Case



Express $\mathbf{x}$ as $\mathbf{x} = \mathbf{x}_p + r\frac{\mathbf{w}}{||\mathbf{w}||}$. Where $\mathbf{x}_p$ is the normal projection of $\mathbf{x}$ onto H.

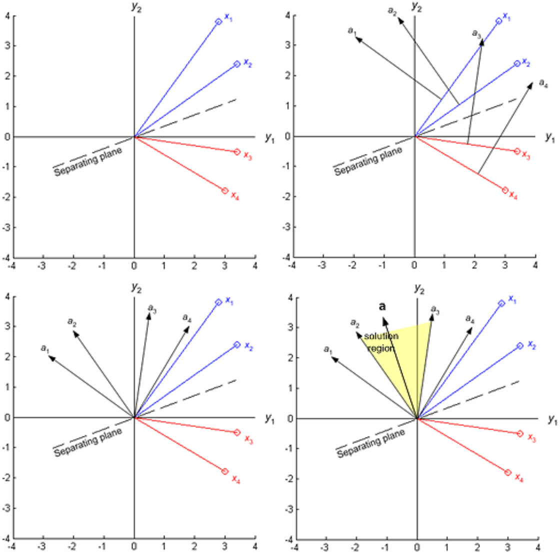
$$\mathbf{x}_p = \mathbf{x} - r\frac{\mathbf{w}}{\|\mathbf{w}\|}$$

$$g(\mathbf{x}_p) = \mathbf{w}^T\mathbf{x}_p + w_0 = 0$$

$$g(\mathbf{x}_p) = \mathbf{w}^T(\mathbf{x} - r\frac{\mathbf{w}}{\|\mathbf{w}\|}) + w_0 = 0$$

$$r = \frac{g(\mathbf{x})}{||\mathbf{w}||}$$

Two Category Case

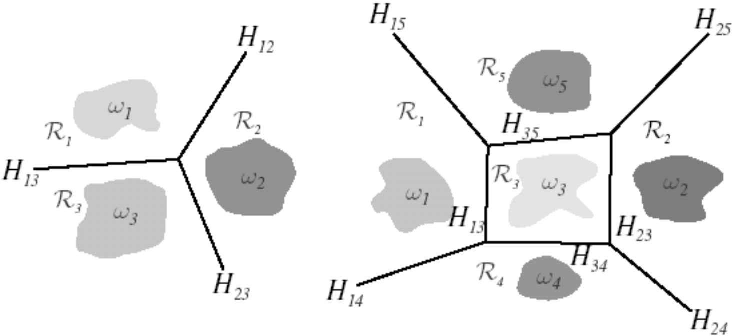


The Multicategory Case

There is more than one way to devise multicategory classifiers employing linear discriminant functions. For example, defining c linear discriminant functions

$$g_i(\mathbf{x}) = \mathbf{w}_i^T \mathbf{x} + w_{i0}, i = 1, \cdots, c$$

and assigning $\mathbf{x}$ to Class_i if $g_i(\mathbf{x}) > g_j(\mathbf{x})$. The resulting classifier is called a linear machine



Generalized Linear Discriminant Functions

$$g(\mathbf{x}) = \sum_{i=1}^{\hat{d}} a_i y_i(\mathbf{x}) \tag{3}$$

or

$$g(\mathbf{x}) = \mathbf{a}^T \mathbf{y}$$

where a is now a $\hat{d}$-dimensional weight vector and where the $\hat{d}$ functions $\mathbf{y}_i(\mathbf{x})$ can be arbitrary functions of $\mathbf{x}$.

The End