

Math 342W/642/742W

Recitation – Day #3 (2.6.25)

I. Nomenclature/Terminology Review

Provide the definition/description for each of the symbols seen below.

- z_1, \dots, z_t : proximal causes – the “true” drivers of the phenomenon (*unknown*)
- t : the unknown function that exactly represents the phenomenon and takes the z ’s as inputs
- x_1, \dots, x_p : the variables/inputs that *proxy* the z ’s
- n : the number of data points in the given training/historical data
- p : the number of features/predictors
- f : the *target* function which produces the least amount of error approximating t
- \mathbb{D} : the training/historical data that the *supervised learning* will be based upon and can be expressed in two equivalent ways:
 - (i) $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$
 - (ii) $\langle X, \mathbf{y} \rangle$ where:
 - the \mathbf{x}_i ’s make up the rows of X and have p components each
 - \mathbf{y} is a column vector whose i th component is y_i
- \mathcal{H} : the set/class of candidate functions that will be considered for approximating f by the learning algorithm
- h^* : the “best” choice in \mathcal{H} that approximates f
- \mathcal{A} : the specified *learning* algorithm that requires \mathbb{D} and \mathcal{H}
- g : the “best” model produced from the learning algorithm based on the training data and the candidate set of functions, i.e., $g = \mathcal{A}(\mathbb{D}, \mathcal{H})$
- g_0 : known as the *null* model which the model that requires no training from the training data
- X : the $n \times p$ matrix made up of rows composed of the training data \mathbf{x}_i , i.e.,

$$X = \begin{bmatrix} \leftarrow \mathbf{x}_1 \rightarrow \\ \leftarrow \mathbf{x}_2 \rightarrow \\ \vdots \\ \leftarrow \mathbf{x}_n \rightarrow \end{bmatrix}$$
- \mathcal{X} : the space of all inputs
- $\mathbf{x}_{\cdot 1}, \dots, \mathbf{x}_{\cdot p}$: the column vectors of X
- $\mathbf{x}_{1\cdot}, \dots, \mathbf{x}_{n\cdot}$: the rows vectors of X
- y : the response variable, i.e., labels
- \mathbf{y} : column vector whose entries are the outputs y_i
- \mathcal{Y} : the space of all outputs
- $\mathbb{1}_a = \begin{cases} 1, & \text{if condition } a \text{ is true} \\ 0, & \text{if condition } a \text{ is false} \end{cases}$
- \mathbf{w} : a vector of p components known as *weights* which are the numerical values we are estimating given as output of the learning algorithm

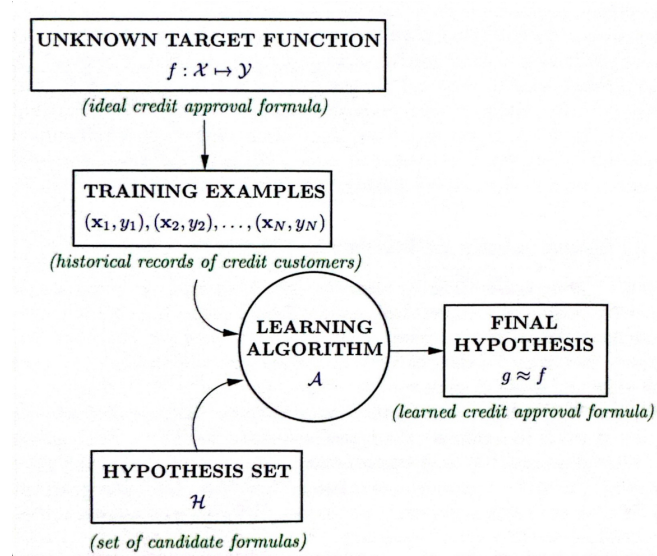
II. Three Ingredients of Supervised Learning

- (i) List the **three** ingredients that comprises *supervised learning*.
- (ii) Draw a schematic diagram/visual of this learning process.

(i) The Three Ingredients of Supervised Learning

- 1) Training Data, $\mathbb{D} = \langle X, \mathbf{y} \rangle$
- 2) Set of Candidate Functions, \mathcal{H}
- 3) Learning Algorithm, \mathcal{A} , which takes \mathbb{D} and \mathcal{H} as inputs

- (ii) Visual of supervised learning on page 4 from *Learning from Data*:



III. Types of Errors

List and mathematically express the **errors** we take into account in *supervised learning*.

Errors in Supervised Learning:

- Ignorance Error: $t(z_1, \dots, z_t) - f(x_1, \dots, x_p)$
- Misspecification Error: $f(x_1, \dots, x_p) - h^*(x_1, \dots, x_p)$
- Estimation Error: $h^*(x_1, \dots, x_p) - g(x_1, \dots, x_p)$
- Residuals (Model Errors): $e_i = y_i - \hat{y}_i = y_i - g(x_i) \in \{0, \pm 1\}$
- Total Error (TE): $\sum_{i=1}^n |e_i| = \sum_{i=1}^n \mathbb{1}_{g(x_i) \neq y_i}$
- Misclassification Error (ME): $= \frac{1}{n} \text{TE}$

IV. The Perceptron Learning Algorithm (PLA)

- (i) Who is credited for being the first to successfully implement the *perceptron* and when/where was it developed?
- (ii) What is the underlying assumption in order to successfully implement PLA?
- (iii) What are the steps/components to PLA and what is the desired result?
- (iv) What are the limitations/drawbacks of PLA?

-
- (i) In 1958, Frank Rosenblatt, a psychologist and project engineer from Cornell University, created the Mark I Perceptron which was able to recognize letters of the alphabet from a 20×20 pixel image as it iteratively *learned* the “correct” values of 400 weights.
 - (ii) The *Perceptron Learning Algorithm* (PLA) is guaranteed to converge under the assumption that the data is already linearly separable prior to the implementation of the algorithm.
 - (iii) The algorithmic steps are expressed in pseudocode below. A successful completion of PLA outputs the weights \mathbf{w} that corresponds to a hyperplane that serves as a boundary between the two sets of linearly separable data.
 - (iv) The limitations/drawbacks of PLA are:
 - relies on the linear separability of the two groups of data *a priori*
 - it produces only one of the infinitely many possibilities of hyperplanes that *separates* the data, not an “*optimal*” one
 - it is purely a computational tool that may provide insight to the correlation of the given data but it does not provide a basis for “*reasoning*”
 - in 1969, Minsky & Papert, two scientists from MIT, wrote a book on the perceptron entitled *Perceptrons: An Introduction to Computational Geometry* and pointed out a specific, and rather elementary, scenario where a single perceptron is unable to correctly separate two sets of data known as the XOR (exclusive-or) problem

Algorithm 1 Perceptron Learning Algorithm

```
Initialize the weights  $\mathbf{w}^{t=0} = \mathbf{0}^{p+1}$ 
while Total Error (TE)  $\neq 0$  do
  for  $i = 1$  to  $n$  do
    Compute  $\hat{y}_i = \mathbb{1}_{\mathbf{w} \cdot \mathbf{x}_i \geq 0}$ 
    Set  $w_0^{t=i} = w_0^{t=i-1} + (y_i - \hat{y}_i) \cdot 1$ 
    Set  $w_1^{t=i} = w_1^{t=i-1} + (y_i - \hat{y}_i) \cdot x_{i,1}$ 
     $\vdots$ 
    Set  $w_p^{t=i} = w_p^{t=i-1} + (y_i - \hat{y}_i) \cdot x_{i,p}$ 
  end for
end while
```

Notable Publications on PLA: (1) "New Navy Device Learns By Doing", *New York Times* (1958), (2) "Rival", *The New Yorker* (1958), (3) "The Design of an Intelligent Automaton", *Research Trends* (1958)