7.0 About K-nearest Neighbor (KNN) Learning

- Most basic instance0-based method
- Inputs of data are numeric ones:
  - Each data point is of n-dimensions, lying in space $\mathbb{R}^n$.
  - Define "nearest" neighbors in terms of Euclidean Distance.

7.1 Euclidean Distance

Given:

- An arbitrary instance $x_i, x_j \in X = \{x_1, x_2, \cdots, x_k\}$.
  - Where x_i, x_j are n-d datas
  - $x_i = [x_{i1} \quad x_{i2} \quad \cdots \quad x_{in}]$, $x_j = [x_{j1} \quad x_{j2} \quad \cdots \quad x_{jn}]$

  Do:
- The euclidean distance between x_i and x_j is:
  - $d(x_i, x_j) = \sqrt{\sum_{r=1}^{n}(x_{ir} - x_{jf})^2}$

7.2 Output Type

7.2.1 Discrete Valued - Classification

Objective:

- Learn a discrete-valued target functions
  - of form $\mathbb{R}^n \to Y$, where
  - $Y = \{y_1, y_2, \cdots, y_s\}$ is the set of target classes

Given:

- A set of training values:
  - $X = \{x_1, x_2, \cdots, x_m\}$, where $\forall i \in [1, m], x_i \in \mathbb{R}^n$.
- A set of classes:
  - $Y = \{y_1, y_2, \cdots, y_s\}$.
- A mapping or assignments function from any training sample to a class:
  - $f : X \to Y$.

- A sample query instance x_q to be classified.
  - $x_q = [x_{q1} \quad x_{q2} \quad \cdots \quad x_{qn}] \in \mathbb{R}^n$.

  Do:
- Let $\{x_1, x_2, \cdots, x_k\}$ be k instances from training examples that's nearest to x_q.
- Output:
  - $\hat{f}(x_1) \leftarrow argmax_{y \in Y} \sum_{i=1}^{k} \delta(y, f(x_i))$, where
  - $\delta(y, f(x_i)) = \begin{cases} 1, if\ f(x_i)=y \\ 0, if\ f(x_i)\neq y \end{cases}$
  - Gives the most common value (class) from the k samples.

7.2.2 Real-Valued - Regression

Objective:

- Learn a discrete-valued target functions
  - of form $\mathbb{R}^n \to y$, where
  - $y \in \mathbb{R}$, which is a real value, i.e., a scalar.

Given:

- A set of training values:
  - $X = \{x_1, x_2, \cdots, x_m\}$, where $\forall i \in [1, m], x_i \in \mathbb{R}^n$.
- A mapping or assignments function from any training examples to a real value.
  - $f : X \to y$, where $y \in \mathbb{R}$.
- A sample query instance x_q to be classified.
  - $x_q = [x_{q1} \quad x_{q2} \quad \cdots \quad x_{qn}] \in \mathbb{R}^n$.

  Do:
- Let $\{x_1, x_2, \cdots, x_k\}$ be k instances from training examples that's nearest to x_q.
- Output:
  - $\hat{f}(x_q) \leftarrow \frac{\sum_{i=1}^{k} f(x_i)}{k}$
  - Simple mean of the values around.

7.3 Distance Weighted

- Weight the contribution
  - of each of the k neighbors
  - according to the distance to query point x_q

    - closer neighbors = greater weights

7.3.1 Discrete-Valued

- $\hat{f}(x_q) \leftarrow argmax_{y \in Y} \sum_{i=1}^{k} w_i \delta(y, f(x_i))$, where
    - $w_i = \dfrac{1}{d(x_q, x_i)^2} = \dfrac{1}{\sum_{j=1}^{n}(x_{ij} - x_{qj})^2}$

7.3.2 Real-Valued

- Each weight of the
- $\hat{f}(x_q) \leftarrow \sum_{i=1}^{k} [\dfrac{w_i}{\sum_{j=1}^{k} w_j} f(x_i)]$, where
    - $w_i = \dfrac{1}{d(x_q, x_i)^2} = \dfrac{1}{\sum_{j=1}^{n}(x_{ij} - x_{qj})^2}$