Decision Theory

Marko Sahan

March 13, 2019

1 Introduction to Decision Theory

Decision process is complicated set of actions that living being makes for satisfying its needs. We want to apply same concept for inanimate thing such as computers. Before constructing some theory we must define some terms with which we will work.

Assuming that we want to solve classification problem. For simplicity we will work with binary classification problem. We want to find such solution that will assign for each input value its class. Moreover, we want to make classification error as small as possible. Considering that input data $\mathbf{x} \in \mathbf{X} \subset \mathbb{R}^n$ and $\mathbf{y} \in \mathbf{Y}$, where $\mathbf{x}$ is an input vectors of size n and $\mathbf{y}$ is its labels assigned to the data from $\mathbf{X}$. Each value from $\mathbf{Y}$ is one or zero $\mathbf{y} \in \{0, 1\}$ that represents first or second class.

In order to make a classification with respect to some input data we must provide an action or in other words decision. Let $a \in \mathcal{A}$ is an action and $\mathcal{A}$ is an action space. Of-course we do not want to make random decisions. We want to make decisions with respect to some metrics that can tell us how good our decision is. Thus, we will introduce a loss function L. From the previous text it is obvious that loss function will be dependent on action $a \in \mathcal{A}$. Furthermore, from the definition of the loss function it must be also dependent on a parameter. Let $\theta \in \Theta$ is parameters' vector and Θ is parameters space. As a result loss function L can be represented as

$$L = L(\theta, a). \tag{1}$$

We will continue construction of the decision theory on the example of Support Vector Machine (SVM) method. For simplicity lets consider linearly separable dataset. From the theoretical perspective SVM constructs hyperplane in high dimensional space that separates two classes. In this case our decision is a hyperplane that will separate two classes from each other. Equation of the hyperplane can be written as $f(\mathbf{x}, \mathbf{w}, b) = \mathbf{w}^T\mathbf{x} + b$ where $\mathbf{w} \in \mathbb{R}^n$ is a set of hyperplane parameters and $b \in \mathbb{R}$ is a bias . As a result, action space is represented as $(\mathbb{R}^n, \mathbb{R}) = \mathcal{A}$ and as a consequence tuple $(\mathbf{w}, b) \in \mathcal{A}$. From this knowledge we can define $\mathbf{X}$ as a parameter space and $\mathbf{Y}$ is the set of possible outcomes (*sample space*) where $\mathbf{y} = \mathbf{y}(\mathbf{x})$. Considering updated definitions loss function (1) can be rewritten as

$$L = L(\mathbf{x}, \mathbf{w}, b). \tag{2}$$

Following task is to understand how good is out action (hyperplane estimation) with respect to the dataset. We can choose different types of the loss functions such as cross entropy or hinge loss, etc. The most basic approach for SVM method is hinge loss function which is defined as

$$L(\mathbf{x}, \mathbf{w}, b) = \max(0, 1 - \mathbf{y}(\mathbf{x})\hat{\mathbf{y}}(\mathbf{x}, \mathbf{w}, b)) \tag{3}$$

where $\hat{\mathbf{y}}(\mathbf{x}, \mathbf{w}, b) = \mathbf{w}^T\mathbf{x} + b$.

If we had available data $\mathbf{X}$ and it its labels $\mathbf{Y}$ we would not have to construct all this theory because all labels would be known and no classification problem must be solved. However, in most cases we would have little discrete subset $\tilde{\mathbf{X}} \subset \mathbf{X}$ and $\tilde{\mathbf{Y}} \subset \mathbf{Y}$. On the basis of $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$ we want to come up with a decision that will help us to label unlabeled data. In terms of SVM method we want to find such hyperplane that will label input values as a first class if it is above the hyperplane and as a second class if it is below the hyperplane. At this point very important assumption will be introduced. In order to find an optimal hyperplane we assume that the data $\tilde{\mathbf{X}}$ and its labels $\tilde{\mathbf{Y}}$ fully describe dataset $\mathbf{X}$ and $\mathbf{Y}$. Moreover we want to consider $\mathbf{x} \in \mathbf{X}$ as a random variable with probability distribution $p(\mathbf{x})$. We will also assume that $\forall i \in \{1, .., N\},\ \mathbf{x}_i \in \tilde{\mathbf{X}}$ is independent identically distributed. With the usage of the data $\tilde{\mathbf{X}}$, probability distribution $p(\mathbf{x})$ can be approximated as

$$p(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} \delta(\mathbf{x} - \mathbf{x}_i) \tag{4}$$

where $\delta(\mathbf{x} - \mathbf{x}_i)$ is Dirac delta function which is centered in x_i.

Definition 1.1. If $\pi^*(\theta)$ is believed probability distribution of θ at the time of decision making, the *Bayesian expected loss* of an action a is

$$\rho(\pi^*, a) = \mathbb{E}_{\pi^*}[L(\theta, a)], \tag{5}$$

$$= \int_{\Theta} L(\theta, a) dF^{\pi^*}(\theta) \tag{6}$$

Using (5) we can evaluate expected loss function for SVM as follows

$$\begin{aligned}
\mathbb{E}_{\pi^*} L &= \int_{\mathbf{X}} L(\mathbf{x}, \mathbf{w}, b) p(\mathbf{x}) d(\mathbf{x}), \\
&= \int_{\mathbf{X}} \max(0, 1 - \mathbf{y}(\mathbf{x}) \hat{\mathbf{y}}(\mathbf{x}, \mathbf{w}, b)) \frac{1}{N} \sum_{i=1}^{N} \delta(\mathbf{x} - \mathbf{x}_i) d(\mathbf{x}), \\
&= \frac{1}{N} \sum_{i=1}^{N} \max(0, 1 - \mathbf{y}(\mathbf{x}_i) \hat{\mathbf{y}}(\mathbf{x}_i, \mathbf{w}, b))
\end{aligned}$$

where $\mathbf{y}(\mathbf{x}_i) = \tilde{\mathbf{y}}_i$ and $\hat{\mathbf{y}}(\mathbf{x}_i) = \hat{\mathbf{y}}_i$. Expect loss function for SVM can be written as

$$\rho(\mathbf{x}_i, \mathbf{w}, b) = \frac{1}{N} \sum_{i=1}^{N} \max(0, 1 - \mathbf{y}_i \mathbf{w}^T \mathbf{x}_i + b) \tag{7}$$