Multiclass Classification

Yufei Tao

Department of Computer Science and Engineering
Chinese University of Hong Kong

Classification (Re-defined)

Let $A_1, ..., A_d$ be d **attributes**.

Define the **instance space** as $\mathcal{X} = dom(A_1) \times dom(A_2) \times ... \times dom(A_d)$ where $dom(A_i)$ represents the set of possible values on A_i.

Define the **label space** as $\mathcal{Y} = \{1, 2, ..., k\}$ (the elements in $\mathcal{Y}$ are called the **class labels**).

Each **instance-label pair** (a.k.a. **object**) is a pair $(\boldsymbol{x}, y)$ in $\mathcal{X} \times \mathcal{Y}$.

- $\boldsymbol{x}$ is a vector; we use $\boldsymbol{x}[A_i]$ to represent the vector's value on A_i $(1 \leq i \leq d)$.

Denote by $\mathcal{D}$ a probabilistic distribution over $\mathcal{X} \times \mathcal{Y}$.

Classification (Re-defined)

Goal: Given an object $(\boldsymbol{x}, y)$ drawn from $\mathcal{D}$, we want to predict its label y from its attribute values $\boldsymbol{x}[A_1], ..., \boldsymbol{x}[A_d]$.

We will find a function

$$h : \mathcal{X} \rightarrow \mathcal{Y}$$

which is referred to as a **classifier** (sometimes also called a **hypothesis**). Given an instance $\boldsymbol{x}$, we predict its label as $h(\boldsymbol{x})$.

The error of h on $\mathcal{D}$ — denoted as $err_{\mathcal{D}}(h)$ — is defined as:

$$err_{\mathcal{D}}(h) \quad = \quad \boldsymbol{Pr}_{(\boldsymbol{x},y)\sim\mathcal{D}}[h(\boldsymbol{x}) \neq y]$$

namely, if we draw an object $(\boldsymbol{x}, y)$ according to $\mathcal{D}$, what is the probability that h mis-predicts the label?

Classification

Ideally, we want to find an h to minimize $err_{\mathcal{D}}(h)$, but this in general is not possible without the precise information about $\mathcal{D}$.

Instead, We would like to learn a classifier h with small $err_{\mathcal{D}}(h)$ from a **training set** S where each object is drawn independently from $\mathcal{D}$.

Classification – Redefined

In training, we are given a sample set S of D, where each object in S is drawn independently according to D. We refer to S as the **training set**.

We would like to learn our classifier h from S.

The key difference from what we have discussed before is that the number k of classes can be anything (in binary classifications, $k = 2$). We will refer to this version of classification as **multiclass classification**.

Think: How would you adapt the decision tree method and Bayes' method to multiclass classification?

Next, assuming that every $dom(A_i)$ $(1 \leq i \leq d)$ is the real domain $\mathbb{R}$, we will extend linear classifiers and Perceptron to multiclass classification.

Linear Classification – Generalized

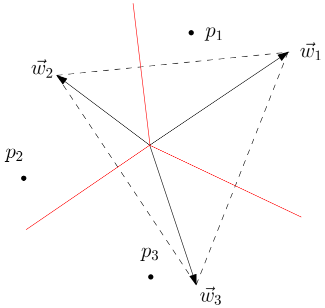
A **generalized linear classifiers** is defined by k d-dimensional vectors $\boldsymbol{w}_1, \boldsymbol{w}_2, ..., \boldsymbol{w}_k$. Given a point p in $\mathbb{R}^d$, the classifier predicts its class label as

$$\arg\max_{i \in [1,k]} \boldsymbol{w}_i \cdot \boldsymbol{p}.$$

Namely, it returns the label $i \in [1, k]$ that gives the largest $\boldsymbol{w}_i \cdot \boldsymbol{p}$.

Tie breaking: In the special case where two distinct $i, j \in [1, d]$ achieve the maximum (i.e., $\boldsymbol{w}_i \cdot \boldsymbol{p} = \boldsymbol{w}_j \cdot \boldsymbol{p}$), we can break the tie using some consistent policy, e.g., predicting the label as the smaller between i and j.

Example



Points p_1, p_2, and p_3 will be classified as label 1, 2, and 3, respectively.

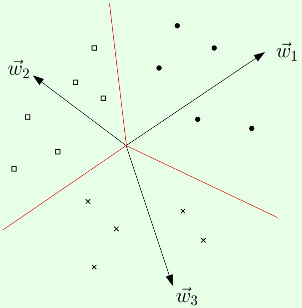
Think: What do the three red rays stand for?

A training set S is **linearly separable** if there exist $\boldsymbol{w}_1, ..., \boldsymbol{w}_d$ that

- correctly classify all the points in S;
- for every point $p \in S$ with label ℓ, $\boldsymbol{w}_\ell \cdot \boldsymbol{p} > \boldsymbol{w}_z \cdot \boldsymbol{p}$ for every $z \neq \ell$.

The set $\{\boldsymbol{w}_1, ..., \boldsymbol{w}_d\}$ is said to **separate** S.

Example:



The dots have label 1, squares label 2, and crosses label 3.

Next we will discuss an algorithm that extends the Perceptron algorithm to find a set of weight vectors to separate S, **provided that** S is linearly separable. We will refer to the algorithm as **multiclass Perceptron**.

Multiclass Perceptron

1. $\boldsymbol{w}_i \leftarrow \boldsymbol{0}$ for all $i \in [1, k]$
2. **while** there is a **violation point** $p \in S$
   /* namely, p mis-classified by $\{\boldsymbol{w}_1, ..., \boldsymbol{w}_k\}$ */
3. $\ell \rightarrow$ the **real label** of p
4. $z \rightarrow$ the **predicted label** of p
   /* $\ell \neq z$ since p is a violation point */
5. $\boldsymbol{w}_\ell \leftarrow \boldsymbol{w}_\ell + \boldsymbol{p}$
6. $\boldsymbol{w}_z \leftarrow \boldsymbol{w}_z - \boldsymbol{p}$

When $k = 2$, the above algorithm degenerates into (the conventional) Perceptron. Can you see why?

"Margin"

Let W be a set of weight vectors $\{\boldsymbol{w}_1, ..., \boldsymbol{w}_k\}$ that separates S.

Given a point $p \in S$ with label ℓ, let us define its **margin under** W as

$$margin(p \mid W) = \min_{z \neq \ell} \frac{\boldsymbol{w}_\ell \cdot \boldsymbol{p} - \boldsymbol{w}_z \cdot \boldsymbol{p}}{\sqrt{2 \sum_{i=1}^{k} |\boldsymbol{w}_i|^2}}.$$

The margin of p under W is a way to measure how "confidently" W gives p the class label ℓ. **Think:** why?

The **margin** of W equals the **smallest** margin of all points under W:

$$margin(W) = \min_{p \in S} margin(p \mid W).$$

"Margin"

Let W^* be a set of weight vectors that (i) separates S, and (ii) has the largest margin.

Define

$$\gamma = margin(W^*).$$

As before, define the **radius** of S as

$$R = \max_{p \in S} |p|.$$

Theorem: Multiclass Perceptron stops after processing at most R^2/γ^2 violation points.

This is the general version of the theorem we have already learned on (the old) Perceptron.

Let M be a $d \times k$ matrix. We use $M[i,j]$ to denote the element at the i-th row and j-th column $(1 \le i \le d, 1 \le j \le k)$.

The **Frobenius norm** of M, denoted as $|M|_F$, is:

$$|M|_F = \sqrt{\sum_{i,j} M[i,j]^2}.$$

Here is an easy way to appreciate the above norm: think of M as a (dk)-dimensional vector by concatenating all its rows; then $|M|_F$ is simply the length of that vector.

Given two $d \times k$ matrices M_1, M_2, the (matrix) **dot product** operation gives a new $d \times k$ matrix M where

$$M[i,j] = M_1[i,j] \cdot M_2[i,j].$$

Proof of the theorem on Slide 14: The algorithm maintains a set of vectors $\{\boldsymbol{w}_1, ..., \boldsymbol{w}_k\}$. Each $\boldsymbol{w}_i$ $(1 \le i \le k)$ is a $d \times 1$ vector.

Henceforth, we will regard a set of vectors $\{\boldsymbol{w}_1, ..., \boldsymbol{w}_k\}$ as a $d \times k$ matrix W, where the i-th ($i \in [1, k]$) row of W is the **transpose** of $\boldsymbol{w}_i$ (i.e., a $1 \times d$ vector).

Define t as the number of violation points.

The algorithm performs t adjustments to W. Denote by W_j ($j \in [1, t]$) as the W after the j-th adjustment. Define specially W_0 the $d \times k$ matrix with all 0's.

Denote by W^* the $d \times k$ matrix that corresponds to an optimal set of weight vectors $\{w_1^*, ..., w_d^*\}$ whose margin is γ.

Claim 1: $W^* \cdot W_t \geq \sqrt{2}t\gamma \cdot |W^*|_F$.

Proof: Consider any $j \in [1, t]$. Let p be the violation point that caused the j-th adjustment. Let ℓ be the real label of p, and z the label predicted by W_{j-1}.

Define Δ as the $d \times k$ matrix such that

- The ℓ-th row of Δ is the transpose of $\boldsymbol{p}$.
- The z-th row of Δ is the transpose of $(-1) \cdot \boldsymbol{p}$.
- All the other rows are 0.

Hence, $W_j = W_{j-1} + \Delta$, which means:

$$W^* \cdot W_j = W^* \cdot W_{j-1} + W^* \cdot \Delta.$$

We will prove $W^* \cdot \Delta \geq \sqrt{2}\gamma \cdot |W^*|_F$, which will complete the proof of Claim 1.

$$
\begin{aligned}
W^* \cdot \Delta &= \boldsymbol{w}_\ell^* \cdot \boldsymbol{p} - \boldsymbol{w}_z^* \cdot \boldsymbol{p} \\
&\geq \gamma \sqrt{2 \sum_{i=1}^{k} |w_i^*|^2} \\
&= \gamma \sqrt{2 |W*|_F^2} \\
&= \sqrt{2}\gamma \cdot |W^*|_F.
\end{aligned}
$$

□

Claim 2: $|W_t|_F^2 \le 2tR^2$.

Proof: Consider any $j \in [1, t]$. Let p be the violation point that caused the j-th adjustment. Let ℓ be the real label of p, and z the label predicted by W_{j-1}. Suppose that $W_{j-1} = \{\boldsymbol{u}_1, ..., \boldsymbol{u}_k\}$.

Since p is a violation point, we must have:

$$\boldsymbol{u}_\ell \cdot \boldsymbol{p} \le \boldsymbol{u}_z \cdot \boldsymbol{p}$$

Denote by $\boldsymbol{v}_\ell$ the new vector for class label ℓ after the update, and similarly by $\boldsymbol{v}_z$ the new vector for class label z after the update. By how the algorithm runs, we have:

$$\begin{aligned} \boldsymbol{v}_\ell &= \boldsymbol{u}_\ell + \boldsymbol{p} \\ \boldsymbol{v}_z &= \boldsymbol{u}_z - \boldsymbol{p} \end{aligned}$$

We have

$$\begin{aligned} |\boldsymbol{v}_\ell|^2 + |\boldsymbol{v}_z|^2 &= (\boldsymbol{u}_\ell + \boldsymbol{p})^2 + (\boldsymbol{u}_z - \boldsymbol{p})^2 \\ &= |\boldsymbol{u}_\ell|^2 + |\boldsymbol{u}_z|^2 + 2|\boldsymbol{p}|^2 + 2(\boldsymbol{u}_\ell \cdot \boldsymbol{p} - \boldsymbol{u}_z \cdot \boldsymbol{p}) \\ \text{(as } \boldsymbol{p} \text{ is a violation point)} &\le |\boldsymbol{u}_\ell|^2 + |\boldsymbol{u}_z|^2 + 2|\boldsymbol{p}|^2 \\ &\le |\boldsymbol{u}_\ell|^2 + |\boldsymbol{u}_z|^2 + 2R^2. \end{aligned}$$

Observe that

$$|W_j|_F^2 - |W_{j-1}|_F^2 = (|\boldsymbol{v}_\ell|^2 + |\boldsymbol{v}_z|^2) - (|\boldsymbol{u}_\ell|^2 + |\boldsymbol{u}_z|^2)$$

We therefore have

$$|W_j|_F^2 - |W_{j-1}|_F^2 \le 2R^2.$$

This completes the proof of the claim. □

Claim 3: $W^* \cdot W_t \leq |W^*|_F \cdot |W_t|_F$.

Proof: The claim follows immediately from the following general result:

Let $\boldsymbol{u}$ and $\boldsymbol{v}$ be two vectors of the same dimensionality; it always holds that $\boldsymbol{u} \cdot \boldsymbol{v} \leq |\boldsymbol{u}||\boldsymbol{v}|$.

The above is true because $\boldsymbol{u} \cdot \boldsymbol{v} = |\boldsymbol{u}||\boldsymbol{v}| \cos\theta$ where θ is the angle between the two vectors. □

By combining Claims 1-3, we have:

$$\begin{aligned}\sqrt{2}t\gamma|W^*|_F &\le |W^*|_F \cdot |W_t|_F \le |W^*|_F \cdot \sqrt{2t}R \\ \Rightarrow t &\le R^2/\gamma^2.\end{aligned}$$

This completes the proof of the theorem.