

Multiclass Classification

Yufei Tao

Department of Computer Science and Engineering
Chinese University of Hong Kong

Classification (Re-defined)

Let A_1, \dots, A_d be d **attributes**.

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

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

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

- \mathbf{x} is a vector; we use $\mathbf{x}[A_i]$ 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 (\mathbf{x}, y) drawn from \mathcal{D} , we want to predict its label y from its attribute values $\mathbf{x}[A_1], \dots, \mathbf{x}[A_d]$.

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 \mathbf{x} , we predict its label as $h(\mathbf{x})$.

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

$$err_{\mathcal{D}}(h) = \Pr_{(\mathbf{x}, y) \sim \mathcal{D}}[h(\mathbf{x}) \neq y]$$

namely, if we draw an object (\mathbf{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

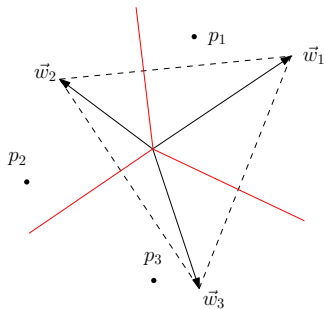
A **generalized linear classifier** is defined by k d -dimensional vectors $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k$. Given a point \mathbf{p} in \mathbb{R}^d , the classifier predicts its class label as

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

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

Tie breaking: In the special case where two distinct $i, j \in [1, d]$ achieve the maximum (i.e., $\mathbf{w}_i \cdot \mathbf{p} = \mathbf{w}_j \cdot \mathbf{p}$), 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.

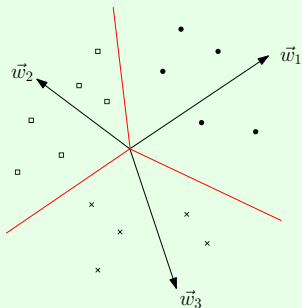
Think: What do the three red rays stand for?

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

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

The set $\{\mathbf{w}_1, \dots, \mathbf{w}_d\}$ 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. $\mathbf{w}_i \leftarrow \mathbf{0}$ for all $i \in [1, k]$
2. **while** there is a **violation point** $p \in S$
/* namely, p mis-classified by $\{\mathbf{w}_1, \dots, \mathbf{w}_k\}$ */
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. $\mathbf{w}_\ell \leftarrow \mathbf{w}_\ell + \mathbf{p}$
6. $\mathbf{w}_z \leftarrow \mathbf{w}_z - \mathbf{p}$

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

“Margin”

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

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

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

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 :

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

“Margin”

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

Define

$$\gamma = \text{margin}(W^*).$$

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 ($1 \leq i \leq d, 1 \leq j \leq k$).

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 real value that equals:

$$\sum_{1 \leq i \leq d, 1 \leq j \leq k} M_1[i, j] \cdot M_2[i, j].$$

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

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

Define t as the number of violation points.

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

Denote by W^* the $k \times d$ matrix that corresponds to an optimal set of weight vectors $\{\mathbf{w}_1^*, \dots, \mathbf{w}_k^*\}$ 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 $k \times d$ matrix such that

- The ℓ -th row of Δ is p (a $1 \times d$ vector).
- The z -th row of Δ is $(-1) \cdot p$.
- 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.

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



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

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 $W_{j-1} = \{u_1, \dots, u_k\}$.

Since p is a violation point, we must have:

$$u_\ell \cdot p \leq u_z \cdot p$$

Denote by v_ℓ the new vector for class label ℓ after the update, and similarly by v_z the new vector for class label z after the update. By how the algorithm runs, we have:

$$v_\ell = u_\ell + p$$

$$v_z = u_z - p$$

We have

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

Observe that

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

We therefore have

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

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

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

By combining Claims 1-3, we have:

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

This completes the proof of the theorem.