linear algebra for AI and ML

(September-23)

LS data fitting problem:

$x^{(1)}, \ldots, x^{(N)} \in \mathbb{R}^n$ and $y^{(1)}, \ldots, y^{(N)} \in \mathbb{R}$

$f : \mathbb{R}^n \to \mathbb{R}$

$f(x) = y \quad \forall x \in \mathbb{R}^n$

$\hat{f} : \mathbb{R}^n \to \mathbb{R}$

$f_i : \mathbb{R}^n \to \mathbb{R} \quad \forall i = 1, 2, \ldots, p$

for $x \in \mathbb{R}^n$ basis: $\{f_1, f_2, \ldots, f_p\}$

parameters: $\alpha_1, \alpha_2, \ldots, \alpha_p$.

linear in α_i's $\rightarrow \hat{f} = \alpha_1 f_1 + \ldots + \alpha_p f_p$

$$\min_{\alpha_1, \ldots, \alpha_p} \sum_{i=1}^{N} \left(y^{(i)} - \hat{f}(x^{(i)})\right)^2$$

$$= \min_{\alpha_1, \ldots, \alpha_p} \|r^d\|_2^2$$

r^d = residual vector

$$y^d = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(N)} \end{bmatrix}; \quad \hat{y}^d = \begin{bmatrix} \hat{f}(x^{(1)}) \\ \vdots \\ \hat{f}(x^{(N)}) \end{bmatrix}$$

LS classification:

$x^{(1)}, \ldots, x^{(N)}$ and $y^{(1)}, y^{(2)}, \ldots, y^{(N)}$: labels/categorical

$f : \mathbb{R}^n \to \mathbb{R}$

$f(x) = y$

only two values

TRUE/FALSE

YES/NO

0/1

-1/1

binary/two classes/

Boolean classification.

$f : \mathbb{R}^n \longrightarrow \{-1, 1\}$

Find $\hat{f} : \mathbb{R}^n \longrightarrow \{-1, 1\}$

$\hat{y} = \hat{f}(x)$

Ex: spam detection/ Disease detection

MNIST $\quad 0, 1, 2, \ldots, 9$ $\quad x^{(1)}, x^{(2)}, \ldots, x^{(N)} \in \mathbb{R}^{784\times1}$

$n = 784$

1 or not $\quad N =$ no. of images.

$\{1\}$ $\quad \{0, 2, 3, 4, \ldots, 9\}$ $\quad y^{(1)}, y^{(2)}, \ldots, y^{(N)}$: labels

$\uparrow$ class 1 $\quad \uparrow$ class 2

$n = 784 \quad (28 \times 28)$

$f : \mathbb{R}^n \longrightarrow \mathbb{R}$

$x \longmapsto \{-1, 1\}$

$\uparrow$ image

$f(x) = -1$ if $x \in \{0, 2, \ldots, 9\}$

$= 1$ if $x = 1$

$\hat{f} : \mathbb{R}^n \longrightarrow \mathbb{R}$ $\quad \hat{y}^{(i)}$ – <u>as close to $y^{(i)}$ as possible</u>.

prediction error: r^d $\underbrace{\hat{y}^{(i)} - y^{(i)} = r^{(i)}}$ $i = 1, 2, \cdots, N$

Confusion matrix:

- True positive : $y = +1$ and $\hat{y} = +1$
- True negative : $y = -1$ and $\hat{y} = -1$
- False positive : $\underbrace{y = -1}$ and $\underbrace{\hat{y} = +1}$
- $\boxed{\text{False negative}}$: $\underbrace{y = +1}$ and $\underbrace{\hat{y} = -1}$

Prediction

$$\begin{array}{lccc} \text{outcome} & \hat{y} = +1 & \hat{y} = -1 & \\ y = +1 & N_{tp} & \boxed{N_{fn}} & N_p \\ y = -1 & \boxed{N_{fp}} & N_{tn} & N_n \\ & & & N \end{array}$$

error rate : $\dfrac{N_{fp} + N_{fn}}{N}$

true positive rate : $\dfrac{N_{tp}}{N_p}$

Specificity / true negative : $\dfrac{N_{tn}}{N_n}$

precision : $\dfrac{N_{tp}}{N_{tp} + N_{fp}}$

We first carry out the ordinary LS fitting of the outcomes.

choose basis f^{n} : $f_1, f_2, \dots, f_p$ $\leftarrow$ $\tilde{f}_1$

parameters: $\alpha_1, \alpha_2, \dots, \alpha_p$

Define: $\tilde{f} = \alpha_1 f_1 + \alpha_2 f_2 + \dots + \alpha_p f_p$

$$\boxed{\min_{\alpha_1, \alpha_2, \dots, \alpha_p} \sum_{i=1}^{N} \left(y^{(i)} - \tilde{f}(x^{(i)})\right)^2}$$

Define $\underbrace{\hat{f}(x) = \operatorname{sign}\big(\underbrace{\tilde{f}(x)}_{a}\big)}$

for any $a \in \mathbb{R}$

$\operatorname{sign}(a) = +1$ if $a \geq 0$

$\operatorname{sign}(a) = -1$ if $a < 0$

$$\hat{f}(x) = \text{sign}\,(\underbrace{x^T\beta + v}_{})$$

$$\tilde{f}(x^{(i)}) = 0.8 \qquad \hat{f}(x^{(i)}) = +1$$

$$\tilde{f}(x^{(j)}) = \underline{\underline{0.02}} \qquad \hat{f}(x^{(j)}) = +1$$

LS multi-class classification:

$x^{(1)}, x^{(2)}, \dots, x^{(N)}$ and $y^{(1)}, y^{(2)}, \dots, y^{(N)}$

$f : \mathbb{R}^n \to \{1, 2, \dots, K\}$ where K is the number of classes.

A multi-class classifier (model)

$$\hat{f} : \mathbb{R}^n \longrightarrow \{1, 2, \dots, K\}$$

Given $x \in \mathbb{R}^n$; $\hat{f}(x) \in \{1, 2, \dots, K\}$

Prediction

$$\begin{array}{c|c}
 & \begin{matrix} \hat{1} & \hat{2} & \cdots & \hat{K} \end{matrix} \\
\text{output} & \\
\begin{matrix} 1 \\ 2 \\ \vdots \\ K \end{matrix} & \begin{bmatrix} \square & * & & * \\ & \square & & \\ & & & \\ & & & \square \end{bmatrix}
\end{array}$$

cofusion matrix

2-class classifier $\longrightarrow$ K-class classifier.

$$\hat{f}(x) = \underset{k=1,2,..,K}{\operatorname{argmax}} \tilde{f}_k(x)$$

where $\tilde{f}_k(x)$ is the least squares regression model for label k against the others.

where $[K] = \{1, .., k\}$

$$\tilde{f}_1(x) \qquad \{1\} \;,\; [K] \setminus \{1\}$$

$$\tilde{f}_2(x) \qquad \{2\} \;,\; [K] \setminus \{2\}$$

$$\vdots$$

$$\tilde{f}_K(x) \qquad \{K\} \;,\; [K] \setminus K$$

$K = 3$

$\{1, 2, 3\} = [K] = [3]$

$\tilde{f}_1(x) = -0.7 \leftarrow$ does <u>NOT</u> belong to $\{1\}$

$\tilde{f}_2(x) = 0.2 \leftarrow$ belongs to $\{2\}$

$\tilde{f}_3(x) = 0.8 \leftarrow$ belongs to $\{3\}$

$\hat{f}(x) = 3 = \operatorname{argmax}\{-0.7, 0.2, 0.8\}$

MNIST : 10 Labels $\{0, 1, 2, \dots, 9\}$

$$\hat{f}(x) = \operatorname*{argmax}_{k=1,2,\dots,10} \left(\tilde{f}_k(x)\right)$$

$\hat{f}_k(x) = \operatorname{sign}(\tilde{f}_k(x))$ where $\tilde{f}_k(x) = x^T \beta_k + v_k$

$$x \in \mathbb{R}^n$$

$$\downarrow$$

$$\begin{bmatrix} \\ \\ \\ \\ \end{bmatrix} \in \mathbb{R}^{n^*} \qquad n^* > n$$

$$\tilde{f}(x) = \underline{\alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_n x_n} + \underline{\alpha_{n+1}}$$

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

$$\tilde{f}(x) = \alpha_1 f_1(x) + \cdots + \alpha_p f_p(x)$$

$$f_1\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = x_1 \; ; \quad f_2\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = x_2, \; \ldots, \; f_n\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = x_n,$$

$$f_{n+1}\begin{pmatrix} \alpha_1 \\ \vdots \\ x_n \end{pmatrix} = 1$$

$$\boxed{p = n+1}$$