Machine Learning Exercise Sheet 05

Linear Classification

Exercise sheets consist of two parts: In-class exercises and homework. The in-class exercises will be solved and discussed during the tutorial. The homework is for you to solve at home and further engage with the lecture content. There is no grade bonus and you do not have to upload any solutions. Note that the order of some exercises might have changed compared to last year's recordings.

In-class Exercises

Multi-Class Classification

Problem 1: Consider a generative classification model for C classes defined by class probabilities $\mathrm{p}(y = c) = \boldsymbol{\pi}_c$ and general class-conditional densities $\mathrm{p}(\boldsymbol{x} \mid y = c, \boldsymbol{\theta}_c)$ where $\boldsymbol{x} \in \mathbb{R}^D$ is the input feature vector and $\boldsymbol{\theta} = \{\boldsymbol{\theta}_c\}_{c=1}^C$ are further model parameters. Suppose we are given a training set $\mathcal{D} = \{(\boldsymbol{x}^{(n)}, y^{(n)})\}_{n=1}^N$ where $y^{(n)}$ is a binary target vector of length C that uses the 1-of-C (one-hot) encoding scheme, so that it has components $y_c^{(n)} = \delta_{ck}$ if pattern n is from class $y = k$. Assuming that the data points are i.i.d., show that the maximum-likelihood solution for the class probabilities $\boldsymbol{\pi}$ is given by

$$\pi_c = \frac{N_c}{N}$$

where N_c is the number of data points assigned to class c.

Linear Discrimant Analysis

Problem 2: Using the same classification model as in the previous question, now suppose that the class-conditional densities are given by Gaussian distributions with a *shared* covariance matrix, so that

$$\mathrm{p}(\boldsymbol{x} \mid y = c, \boldsymbol{\theta}) = \mathrm{p}(\boldsymbol{x} \mid \boldsymbol{\theta}_c) = \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_c, \boldsymbol{\Sigma}).$$

Show that the maximum likelihood estimate for the mean of the Gaussian distribution for class c is given by

$$\boldsymbol{\mu}_c = \frac{1}{N_c} \sum_{\substack{n=1 \\ y^{(n)}=c}}^{N} \boldsymbol{x}^{(n)}$$

which represents the mean of the observations assigned to class c.

Similarly, show that the maximum likelihood estimate for the shared covariance matrix is given by

$$\boldsymbol{\Sigma} = \sum_{c=1}^{C} \frac{N_c}{N} \boldsymbol{S}_c \quad \text{where} \quad \boldsymbol{S}_c = \frac{1}{N_c} \sum_{\substack{n=1 \\ y^{(n)}=c}}^{N} (\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_c)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_c)^{\mathrm{T}}.$$

Problem 1: Consider a generative classification model for C classes defined by class probabilities $\mathrm{p}(y = c) = \pi_c$ and general class-conditional densities $\mathrm{p}(\boldsymbol{x} \mid y = c, \boldsymbol{\theta}_c)$ where $\boldsymbol{x} \in \mathbb{R}^D$ is the input feature vector and $\boldsymbol{\theta} = \{\boldsymbol{\theta}_c\}_{c=1}^C$ are further model parameters. Suppose we are given a training set $\mathcal{D} = \{(\boldsymbol{x}^{(n)}, y^{(n)})\}_{n=1}^N$ where $y^{(n)}$ is a binary target vector of length C that uses the 1-of-C (one-hot) encoding scheme, so that it has components $y_c^{(n)} = \delta_{ck}$ if pattern n is from class $y = k$. Assuming that the data points are i.i.d., show that the maximum-likelihood solution for the class probabilities $\boldsymbol{\pi}$ is given by

$$\pi_c = \frac{N_c}{N}$$

where N_c is the number of data points assigned to class c.

$p(y^{(i)} = 1 \mid \pi) = \pi_c \qquad p(x \mid y = c, \theta_c)$

$p(x, y \mid \pi, \theta) = \prod_{n=1}^{N} p(x^{(n)} \mid y^{(n)}, \theta) \cdot p(y^{(n)} \mid \pi)$ ← one-hot

$= \prod_{n=1}^{N} \prod_{c=1}^{C} \left[p(x^{(n)} \mid y_c^{(n)}, \theta_c)\, p(y_c^{(n)} \mid \pi_c) \right]^{y_c^{(n)}}$

$1 \cdot 1 \cdot 1 \cdot 1 \cdot \cdots \cdot p(x^{(n)} \mid y_c^{(n)} = 1, \theta_c) \cdot p(y_c^{(n)} = 1 \mid \pi_c)$ ← no π_c

log-likelihood

$\log p(x, y \mid \pi, \theta) = \sum_{n=1}^{N} \sum_{c=1}^{C} y_c^{(n)} \log \pi_c + \text{const} \quad \text{w.r.t. } \pi_c$

$\sum_{c=1}^{C} \pi_c = 1$

$\nabla f(x,y) = \lambda \nabla g(x,y)$

$\nabla f(x,y) - \lambda \nabla g(x,y) = 0$

$\max \quad \underbrace{\log p(x, y \mid \pi, \theta)}_{f} - \lambda \underbrace{\left[\sum_{c=1}^{C} \pi_c - 1 \right]}_{g} = 0.$

$\sum_{n=1}^{N} \frac{y_c^{(n)}}{\pi_c} - \lambda \overset{!}{=} 0. \qquad \sum_{c=1}^{C} \pi_c - 1 \overset{!}{=} 0.$

$\pi_c = \frac{\sum_{n=1}^{N} y_c^{(n)}}{\lambda} = \frac{N_c}{\lambda}. \qquad \sum_{c=1}^{C} \frac{N_c}{\lambda} \overset{!}{=} 1.$

$\lambda = \sum_{c=1}^{C} N_c = N.$

$\pi_c = \frac{N_c}{\lambda} = \frac{N_c}{N}$

Problem 2: Using the same classification model as in the previous question, now suppose that the class-conditional densities are given by Gaussian distributions with a *shared* covariance matrix, so that

$$\mathrm{p}(\boldsymbol{x} \mid y = c, \boldsymbol{\theta}) = \mathrm{p}(\boldsymbol{x} \mid \boldsymbol{\theta}_c) = \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_c, \boldsymbol{\Sigma}).$$

Show that the maximum likelihood estimate for the mean of the Gaussian distribution for class c is given by

$$\boldsymbol{\mu}_c = \frac{1}{N_c} \sum_{\substack{n=1 \\ y^{(n)}=c}}^{N} \boldsymbol{x}^{(n)}$$

which represents the mean of the observations assigned to class c.

Similarly, show that the maximum likelihood estimate for the shared covariance matrix is given by

$$\boldsymbol{\Sigma} = \sum_{c=1}^{C} \frac{N_c}{N} \boldsymbol{S}_c \quad \text{where} \quad \boldsymbol{S}_c = \frac{1}{N_c} \sum_{\substack{n=1 \\ y^{(n)}=c}}^{N} (\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_c)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_c)^{\mathrm{T}}.$$

$p(y|x) = p(x|y) \cdot p(y)$ $\quad p(y_c = 1) = \lambda_c$

$p(x|y=1, \theta_c) = \mathcal{N}(x \mid \mu_c, \Sigma)$

$= \frac{1}{\sqrt{(2\pi)^D \cdot |\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu_c)^T \Sigma^{-1} (x-\mu_c)\right)$

$\log p(y|x) = \sum_{n=1}^{N} \sum_{c=1}^{C} y_c^{(n)} \cdot (\log \lambda_c \cdot p(x, \theta_c))$

$= \sum_{n=1}^{N} \sum_{c=1}^{C} y_c^{(n)} \left[\log \lambda_c + \left(-\frac{1}{2}\right)\left[(D \log(2\pi) + \log|\Sigma|) + (x-\mu_c)^T \Sigma^{-1} (x-\mu_c)\right]\right]$

$= \sum_{n=1}^{N} \sum_{c=1}^{C} -\frac{1}{2} y_c^{(n)} \cdot \left[-2\log \lambda_c + D \log(2\pi) + \log|\Sigma| + (x-\mu_c)^T \Sigma^{-1} (x-\mu_c)\right]$

$(x^T \Sigma^{-1} - \mu_c^T \Sigma^{-1})(x - \mu_c)$

$x^T \Sigma^{-1} x - 2x^T \Sigma^{-1} \mu_c$

$- \cancel{\mu_c^T \Sigma^{-1} x} + \mu_c^T \Sigma^{-1} \mu_c$

$\nabla_{\mu_c} \sum_{n=1}^{N} -\frac{1}{2} y_c^{(n)} \left(-2x^T \Sigma^{-1} + 2\Sigma^{-1} \mu_c\right) = 0$

$\sum_{n=1}^{N} y_c^{(n)} \cdot 2\Sigma^{-1} \cdot \mu_c = \sum_{n=1}^{N} y_c^{(n)} \cdot 2x^T \Sigma^{-1}$

$\mu_c = \frac{\sum_{n=1}^{N} y_c^{(n)} \cdot 2x^T \cancel{\Sigma^{-1}}}{\sum_{n=1}^{N} y_c^{(n)} \cdot \cancel{2\Sigma^{-1}}} = \frac{1}{N_c} \left(\sum_{\substack{n=1 \\ y_c^{(n)}=1}}^{N} x^{(n)}\right)$

② $Tr(ABC) = Tr(BCA)$ $\quad Tr(a) = a$.

$(x-\mu_c)^T \Sigma^{-1} (x-\mu_c) = Tr\left[\Sigma^{-1}(x-\mu_c)(x-\mu_c)^T\right]$.

$\log|A| = -\log|A^{-1}|$

$\frac{d\log P}{d\Sigma} = \frac{d\log P}{d\Sigma^{-1}} \frac{d\Sigma^{-1}}{d\Sigma}$

$\frac{d\,Tr(AB)}{dA} = B^T$

$\frac{d\log|x|}{dx} = (x^{-1})^T$

$$\nabla_{\Sigma^{-1}} = -\frac{1}{2}\sum_{n=1}^{N}\sum_{c=1}^{C} y_c^{(n)}\left[-\log|A^{-1}| + Tr\left[\Sigma^{-1}(x-\mu_c)(x-\mu_c)^T\right]\right]$$

$$= -\frac{1}{2}\sum_{n=1}^{N}\sum_{c=1}^{C} y_c^{(n)}\left(-\Sigma^T + (x-\mu_c)(x-\mu_c)^T\right) = 0.$$

$$= -N\Sigma^T + \sum_{n=1}^{N}\sum_{c=1}^{C} y_c^{(n)}(x-\mu_c)(x-\mu_c)^T = 0.$$

$$\Sigma^T = \frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C} y_c^{(n)}(x-\mu_c)(x-\mu_c)^T$$

$$\Sigma = \frac{1}{N}\sum_{\substack{n=1 \\ y_c^{(n)}=1}}^{N}(x-\mu_c)^T(x-\mu_c)$$

Problem 1: Consider a generative classification model for C classes defined by class probabilities $\mathrm{p}(y = c) = \boldsymbol{\pi}_c$ and general class-conditional densities $\mathrm{p}(\boldsymbol{x} \mid y = c, \boldsymbol{\theta}_c)$ where $\boldsymbol{x} \in \mathbb{R}^D$ is the input feature vector and $\boldsymbol{\theta} = \{\boldsymbol{\theta}_c\}_{c=1}^C$ are further model parameters. Suppose we are given a training set $\mathcal{D} = \{(\boldsymbol{x}^{(n)}, y^{(n)})\}_{n=1}^N$ where $y^{(n)}$ is a binary target vector of length C that uses the 1-of-C (one-hot) encoding scheme, so that it has components $y_c^{(n)} = \delta_{ck}$ if pattern n is from class $y = k$. Assuming that the data points are i.i.d., show that the maximum-likelihood solution for the class probabilities $\boldsymbol{\pi}$ is given by

$$\pi_c = \frac{N_c}{N}$$

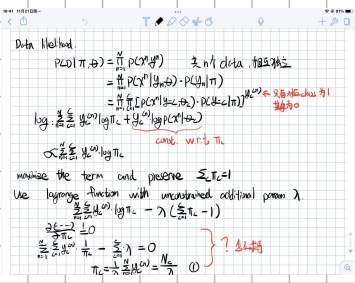
where N_c is the number of data points assigned to class c.

Data likelihood

$$p(X|\Theta) = P(D|\{\pi_c, \theta_c\}_{c=1}^C) = \prod_{n=1}^N p(x^n, y^n)$$

$$= \prod_{n=1}^N p(x^n|y^n, \Theta)\, p(y^n|\pi)$$

$$= \prod_{n=1}^N \prod_{c=1}^C \left[p(x^n|y^n=c, \theta_c) \cdot p(y^n=c|\pi) \right]^{y_c^n}$$



对于 $N=1$ 时

$y^n = (0\ 0\ 0\ 0\ \cdots\ 1\ 0)$

$\therefore [\cdots]^{y_c^n}$ 只能 y^n 有效

$= \begin{cases} y^n=1 \text{ 时 } P(x|\theta) \\ y^n=\text{other } =1 \end{cases}$

$$\pi^* = \arg\max_{\pi} \log p(X|\theta) = \sum_{n=1}^N \sum_{c=1}^C \left(y_c^n \cdot \log \pi_c + C \right)$$

assume $\sum_c \pi_c = 1$

Lagrange

$$\mathcal{L} : \sum_{n=1}^N \sum_{c=1}^C (y_c^n \cdot \log \pi_c) + C - \lambda(\Sigma \pi_c - 1)$$

$$\frac{\partial \mathcal{L}}{\partial \pi_c} = \frac{1}{\pi_c} \cdot y_c^{(n)} - \lambda = 0 \Rightarrow \pi_c = \frac{1}{\lambda} \sum_{n=1}^N y_c^{(n)}$$

$$= \frac{N_c}{\lambda}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \Sigma \pi_c = 1$$

$$\sum_{c=1}^C \frac{N_c}{\lambda} = 1$$

$$\sum_{c=1}^C N_c = \lambda = N$$

$$\boxed{\pi_c = \frac{N_c}{N}}$$

$\mathcal{L} = A \cdot \sum_{n=1}^N B_n$

$$\frac{\partial \mathcal{L}}{\partial B_n} = \frac{\partial A(B_1 + B_2 + B_3 + \cdots + B_n)}{\partial B_n}$$

$n=1$ 时 $\frac{\partial \mathcal{L}}{\partial B_1} = A$ 与求导结果无关

$n=2$ 时 $\frac{\partial \mathcal{L}}{\partial B_2} = A$

$n=n$ 时 $\frac{\partial \mathcal{L}}{\partial B_n} = A$

Problem 2: Using the same classification model as in the previous question, now suppose that the class-conditional densities are given by Gaussian distributions with a *shared* covariance matrix, so that

$$\mathrm{p}(\boldsymbol{x} \mid y = c, \boldsymbol{\theta}) = \mathrm{p}(\boldsymbol{x} \mid \boldsymbol{\theta}_c) = \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_c, \boldsymbol{\Sigma}).$$

Show that the maximum likelihood estimate for the mean of the Gaussian distribution for class c is given by

$$\boldsymbol{\mu}_c = \frac{1}{N_c} \sum_{\substack{n=1 \\ y^{(n)}=c}}^{N} \boldsymbol{x}^{(n)}$$

which represents the mean of the observations assigned to class c.

Similarly, show that the maximum likelihood estimate for the shared covariance matrix is given by

$$\boldsymbol{\Sigma} = \sum_{c=1}^{C} \frac{N_c}{N} \boldsymbol{S}_c \quad \text{where} \quad \boldsymbol{S}_c = \frac{1}{N_c} \sum_{\substack{n=1 \\ y^{(n)}=c}}^{N} (\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_c)(\boldsymbol{x}^{(n)} - \boldsymbol{\mu}_c)^{\mathrm{T}}.$$

Thus $\boldsymbol{\Sigma}$ is given by a weighted average of the sample covariances of the data associated with each class, in which the weighting coefficients N_c/N are the prior probabilities of the classes.

data likelihood

$\log p(D \mid \{\pi_c, \theta_c\}_{c=1}^{C}) = \sum_{n=1}^{N} \sum_{c=1}^{C} \left(y_c^n \cdot \log \pi_c + y_c^n \cdot \log p(x \mid y=c, \theta) \right)$

$= \sum_{n=1}^{N} \sum_{c=1}^{C} y_c^n \cdot \left(-\frac{D}{2} \log 2\pi - \frac{1}{2} \log |\Sigma| - \frac{1}{2} (x-\mu_c)^T \Sigma^{-1} (x-\mu_c) + \log \pi_c \right)$

$= \sum_{n=1}^{N} \sum_{c=1}^{C} -\frac{1}{2} y_c^n \cdot \left(D \cdot \log 2\pi + \log |\Sigma| + (x-\mu_c)^T \Sigma^{-1} (x-\mu_c) - 2 \log \pi_c \right)$

$\frac{\partial -}{\partial \mu_c} = \sum_{n=1}^{N} y_c^n \cdot \left(\Sigma^{-1} (x^n - \mu_c) \right) = 0$

$\sum_{n=1}^{N} y_c^n \Sigma^{-1} x^n = \sum_{n=1}^{N} y_c^n \Sigma^{-1} \mu_c = \mu_c \sum_{n=1}^{N} y_c^n$

$\mu_c = \frac{\sum_{n=1}^{N} y_c^n x^n}{\sum_{n} y_c^n} = \frac{1}{N_c} \sum_{\substack{n=1 \\ y^n=c}}^{N} x^n$

$\mathrm{Tr}[(x-\mu_c)^T \Sigma^{-1} (x-\mu_c)] = \mathrm{Tr}[\Sigma^{-1} (x-\mu_c)^T (x-\mu_c)]$

$\frac{\partial \mathrm{Tr}(AB)}{\partial A} = B^T \quad A = -\log |A|^{-1}$

$= \sum_{n=1}^{N} \sum_{c=1}^{C} -\frac{1}{2} y_c^n \cdot \left(D \log 2\pi + \log |\Sigma| + \mathrm{Tr}[\Sigma^{-1} (x-\mu_c)^T (x-\mu_c)] - 2 \log \pi_c \right)$

$\frac{\partial -}{\partial \Sigma} = \sum_{n=1}^{N} \sum_{c=1}^{C} -\frac{1}{2} y_c^n \left(-\log |\Sigma^{-1}| + \mathrm{Tr}[\Sigma^{-1} (x-\mu_c)^T (x-\mu_c)] \right)$

$\frac{\partial \log |X|}{\partial X} = (X^{-1})^T$

$= -\frac{1}{2} \sum_{n=1}^{N} \sum_{c=1}^{C} y_c^n \left(-\Sigma^T + (x^n - \mu_c)^T (x^n - \mu_c) \right) = 0$

$\Sigma^T = \frac{1}{\sum_{n=1}^{N} \sum_{c=1}^{C} y_c^n} \sum_{n=1}^{N} \sum_{c=1}^{C} \left(y_c^n (x^n - \mu_c)^T (x - \mu_c) \right)$

$\frac{1}{N} \sum_{n=1}^{N} \sum_{c=1}^{C} y_c^n (x^n - \mu_c)(x - \mu_c)^T$

Thus $\mathbf{\Sigma}$ is given by a weighted average of the sample covariances of the data associated with each class, in which the weighting coefficients N_c/N are the prior probabilities of the classes.

Homework

Linear classification

Problem 3: We want to create a generative binary classification model for classifying *non-negative* one-dimensional data. This means, that the labels are binary ($y \in \{0, 1\}$) and the samples are $x \in [0, \infty)$.

We assume uniform class probabilities

$$\mathrm{p}(y = 0) = \mathrm{p}(y = 1) = \frac{1}{2}.$$

As our samples x are non-negative, we use exponential distributions (and not Gaussians) as class conditionals:

$$\mathrm{p}(x \mid y = 0) = \mathrm{Expo}(x \mid \lambda_0) \qquad \text{and} \qquad \mathrm{p}(x \mid y = 1) = \mathrm{Expo}(x \mid \lambda_1),$$

where $\lambda_0 \neq \lambda_1$. Assume, that the parameters λ_0 and λ_1 are known and fixed.

a) Suppose you are given an observation x. What is the name of the posterior distribution $\mathrm{p}(y \mid x)$? You only need to provide the name of the distribution (e.g., "normal", "gamma", etc.), not estimate its parameters.

b) What values of x are classified as class 1? (As usual, we assume that the classification decision is $\hat{y} = \arg\max_k \mathrm{p}(y = k \mid x)$)

Problem 4: Let $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}$ be a linearly separable dataset for 2-class classification, i.e. there exists a vector $\boldsymbol{w}$ such that $\operatorname{sign}\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}\right)$ separates the classes. Show that the maximum likelihood parameter $\boldsymbol{w}$ of a logistic regression model has $\|\boldsymbol{w}\| \to \infty$. Assume that $\boldsymbol{w}$ contains the bias term.

How can we modify the training process to prefer a $\boldsymbol{w}$ of finite magnitude?

Problem 5: Show that the softmax function is equivalent to a sigmoid in the 2-class case.

Problem 6: Show that the derivative of the sigmoid function $\sigma(a) = (1 + e^{-a})^{-1}$ can be written as

$$\frac{\partial \sigma(a)}{\partial a} = \sigma(a)\left(1 - \sigma(a)\right).$$

Problem 7: Give a basis function $\phi(x_1, x_2)$ that makes the data in the example below linearly separable (crosses in one class, circles in the other).

3 $f(x)=\begin{cases}\lambda e^{-\lambda x} & x\geq 0\\ 0 & x<0\end{cases}$

① $p(y)\sim Ber(\cdot)$

$p(y|x)\sim Ber$

② $p(y=1|x)>p(y=0|x)$

$\Rightarrow \frac{p(y=1|x)}{p(y=0|x)} \overset{!}{>} 1 \quad \log \frac{p(y=1|x)}{p(y=0|x)} \overset{!}{>} 0$

$= \log \frac{p(x|y=1)p(y=1)}{p(x|y=0)p(y=0)}$

$= \log \frac{p(x|y=1)=\lambda_1 e^{-\lambda_1 x}}{p(x|y=0)=\lambda_0 e^{-\lambda_0 x}}$

$= \log \frac{\lambda_1}{\lambda_0} + \lambda_0 x - \lambda_1 x \overset{!}{>} 0 = \log 1$

$(\lambda_0-\lambda_1)x > \log\lambda_0 - \log\lambda_1$

$x > \frac{\log\lambda_0 - \log\lambda_1}{\lambda_0-\lambda_1} \Rightarrow \begin{cases} x\in \left(\frac{\log\lambda_0-\log\lambda_1}{\lambda_0-\lambda_1}, \infty\right) & \lambda_0>\lambda_1 \\ x\in [0, \ ) & \lambda_0<\lambda_1 \end{cases}$

---

4 $E(w) = -\sum_{i=1}^{N} (y_i \cdot \log \sigma(w^T x_i) + (1-y_i)\log(1-\sigma(w^T x_i)))$ $\quad \sigma(a)=\frac{1}{1+e^{-a}}$

$w^* = \text{argmin } E(w) = \frac{\partial E}{\partial w} = -\sum_{i=1}^{N} \left(y_i \cdot \frac{1}{\sigma(w^T x_i)} \cdot \frac{\partial \sigma(w^T x_i)}{\partial w} + (1-y_i)\left(-\frac{\partial \sigma(w^T x_i)}{\partial w}\right)\right)$

$= -\sum_{i=1}^{N} \left(y_i (1+e^{-w^T x_i}) \cdot (1+e^{-w^T x_i})^{-2} \cdot e^{-w^T x_i} x_i + (1-y_i)(1+(1+e^{-w^T x_i})^{-2} \cdot e^{-w^T x_i})\right)$

$= -\sum_{i=1}^{N} (1+$

$E(w) = -\sum_{i=1}^{N} (y_i \cdot \log \sigma(w^T x_i) + (1-y_i)\log(1-\sigma(w^T x_i)))$

set $w^T x_i > 0$ $y_i = 1$ $\quad w^T x_i < 0$, $y_i = 0$

use scaling $\lambda \gg 0$ make the negative log-likelihood smaller

$\lim_{\alpha\to\infty} E(\lambda w) = -\sum_{i=1}^{N} \left(\lim \log \sigma(\lambda w^T x_i) + \lim \log\left(1-\sigma(\alpha w^T x_i)\right)\right)$

$\log\sigma(\lambda w^T x_i) > 0$, $\log(1-\sigma(\alpha w^T x_i)) < 0$

$= 0$ smallest $\Rightarrow E(w) \in [0, \infty)$

我们可以看到E是一个凸函数，因为log是凹的，σ在a<0时是凸的，在a>0时是凹的。所以logσ(a)在a>0时是凹的，log（1-σ(a)）在a<0时是凹的。

如果一个凸函数达到了它的最小值，那么它就有一个唯一的最小值。我们知道，当λ→∞时，E趋向于它的最小值，所以E不可能有一个有限的最小值，它的所有最小值只在极限中实现。由此可见，任何损失最小化问题的解决方案都具有无限的规范。

由于E是凸的，并且在某些方向上趋向于0的极限，我们可以通过添加任何实现其最小值的凸项，如w T w或类似形式的权重正则化，将最小值移动到finite向量空间。

softmax function: $\sigma(x)_i = \frac{\exp(x_i)}{\sum_{k=1}^{K} \exp(x_k)}$

LDA:

sigmoid $\sigma(a) = \frac{1}{1+\exp(-a)}$

$a = \log \frac{p(x|y=1)\,p(y=1)}{p(x|y=0)\,p(y=0)}$

$= w^T x + w_0$

$= \frac{1}{1+\exp(-w^T x - w_0)}$

$\frac{1}{\sigma(a)} = 1 + \frac{1}{\exp(w^T x + w_0)}$

$\sigma(a) = \frac{\exp(w^T x + w_0)}{1+\exp(w^T x + w_0)} = \frac{\exp(w_0 + w_1 x_1 + \dots)}{\dots} = \frac{\dots}{\dots}$

softmax $\sigma(x_i) = \frac{\exp(w_1^T x)}{\exp(w_1^T x) + \exp(w_0^T x)}$

$= \frac{1}{1 + \exp(w_0^T x - w_1^T x)}$

$= \frac{1}{1+\exp(-(w_1 - w_0)^T x)}$

$= \sigma(\hat{w}^T x)$

$\hat{w} = w_1 - w_0$

我们可以从中得出的一个结论是，如果我们有C类的参数向量w c，逻辑回归模型是不可识别的。这意味着在每个向量w c := w c + τ中加入一个常数τ∈R D将导致相同的逻辑回归模型。我们可以通过增加一个约束条件w 1 = 0来解决这个问题，这就是我们在二元分类中使用sigmoid（而不是2类softmax）时隐含的做法。

6. $\frac{\partial \sigma(a)}{\partial a} = -\frac{1}{(1+e^{-a})^2} \cdot (-e^{-a})$

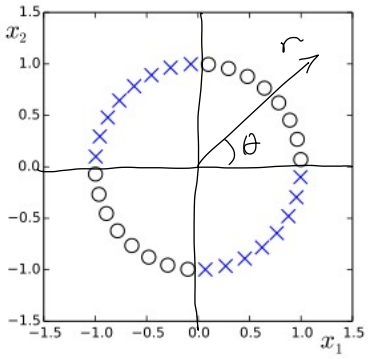
$= \sigma^2(a) \cdot \left(\frac{1}{\sigma(a)} - 1\right)$

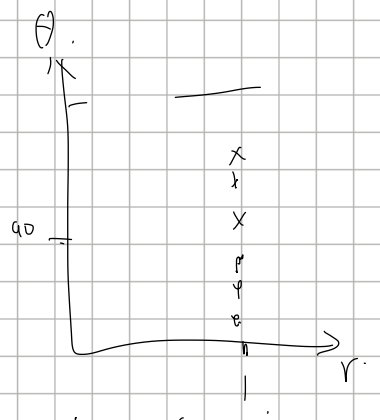
$= \frac{1}{(1+e^{-a})} \cdot \frac{e^{-a}}{(1+e^{-a})}$

$= \sigma(a) \cdot \frac{1+e^{-a}-1}{(1+e^{-a})}$

$= \sigma(a)(1-\sigma(a))$

7





$\phi(x) = (\theta)$

$= \arctan\left(\frac{\theta}{2}\right)$

$(x - \frac{\theta}{2}k) \in (0, \theta)$

$\phi(x) = x_1 x_2$

一个例子是φ (x) = x 1 x 2，这使得数据可以通过超平面w = (1)来分离，因为圆圈将被映射到正实数，而十字架则去了负数，也就是说，如果x是一个圆圈，w T x > 0，否则w T x < 0。

8

Naive Bayes

Problem 8: In 2-class classification the decision boundary Γ is the set of points where both classes are assigned equal probability,

$$\Gamma = \{\boldsymbol{x} \mid \mathrm{p}(y=1 \mid \boldsymbol{x}) = \mathrm{p}(y=0 \mid \boldsymbol{x})\}\,.$$

Show that Naive Bayes with Gaussian class likelihoods produces a quadratic decision boundary in the 2-class case, i.e. that Γ can be written with a quadratic equation of $\boldsymbol{x}$,

$$\Gamma = \left\{\boldsymbol{x} \mid \boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}^{\mathrm{T}}\boldsymbol{x} + c = 0\right\},$$

for some $\boldsymbol{A}$, $\boldsymbol{b}$ and c.

As a reminder, in Naive Bayes we assume class prior probabilities

$$\mathrm{p}(y=0) = \pi_0 \quad \text{and} \quad \mathrm{p}(y=1) = \pi_1$$

and class likelihoods

$$\mathrm{p}(\boldsymbol{x} \mid y=c) = \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$$

with per-class means $\boldsymbol{\mu}_c$ and *diagonal* (because of the feature independence) covariances $\boldsymbol{\Sigma}_c$.

$$\frac{p(y=1 \mid x)}{p(y=0 \mid x)} = 1$$

$$\frac{p(x \mid y=1) \cdot p(y=1)}{p(x \mid y=0)\, p(y=0)} = 1$$

$$\log p(x|y=1) + \log \pi_1 - \log p(x|y=0) - \log \pi_0 = \log 1 = 0$$

$$\log \mathcal{N}(x|\mu_1, \Sigma_1) - \log \mathcal{N}(x|\mu_0, \Sigma_0) + \log \frac{\pi_1}{\pi_0} = 0$$

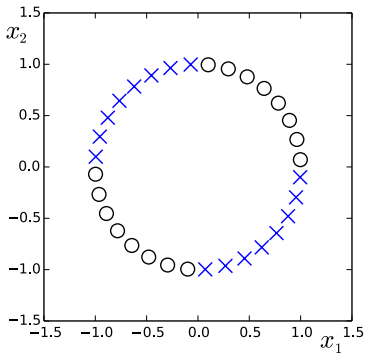
$$= -\frac{1}{2}\log (2\pi)^D |\Sigma_1| - \frac{1}{2}(x-\mu_1)^T \cdot \Sigma_1^{-1}(x-\mu_1) + \frac{1}{2}\log (2\pi)^D |\Sigma_0| + \frac{1}{2}(x-\mu_0)^T \Sigma_0^{-1}(x-\mu_0) + \log \frac{\pi_1}{\pi_0} = 0$$

$$= (x^T \Sigma_0^{-1} - \mu_0^T \Sigma_0^{-1})(x-\mu_0) - (x^T \Sigma_1^{-1} - \mu_1^T \Sigma_1^{-1})(x-\mu_1) + 2\log \frac{\pi_1}{\pi_0} + \underbrace{\log (2\pi)^D \cdot |\Sigma_0| - \log (2\pi)^D \cdot |\Sigma_1|}_{=0}$$

$$= \underline{x^T \Sigma_0^{-1} x} - 2\underline{x^T \Sigma_0^{-1} \cdot \mu_0} + \mu_0^T \cdot \Sigma_0^{-1} \cdot \mu_0 - \underline{x^T \Sigma_1^{-1} x} + \underline{2 x^T \Sigma_1^{-1} \cdot \mu_1} - \mu_1^T \Sigma_1^{-1} \mu_1 + 2\log \frac{\pi_1}{\pi_0} + \cdots$$

$$= \underbrace{x^T (\Sigma_0^{-1} - \Sigma_1^{-1}) x}_{A} + \underbrace{x^T (2\Sigma_1^{-1}\mu_1 - 2\Sigma_0^{-1}\mu_0)}_{b} + \underbrace{\left(\mu_0^T \Sigma_0^{-1} \mu_0 - \mu_1^T \Sigma_1^{-1} \mu_1 + 2\log \frac{\pi_1}{\pi_0} + \log (2\pi)^D |\Sigma_0| - \log (2\pi)^D |\Sigma_1|\right)}_{c}$$

for $\Sigma_0 = \Sigma_1$, $A = \vec{0}$. Linear boundary



Naive Bayes

Problem 8: In 2-class classification the decision boundary Γ is the set of points where both classes are assigned equal probability,

$$\Gamma = \{\boldsymbol{x} \mid \mathrm{p}(y = 1 \mid \boldsymbol{x}) = \mathrm{p}(y = 0 \mid \boldsymbol{x})\}\,.$$

Show that Naive Bayes with Gaussian class likelihoods produces a quadratic decision boundary in the 2-class case, i.e. that Γ can be written with a quadratic equation of $\boldsymbol{x}$,

$$\Gamma = \left\{\boldsymbol{x} \mid \boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}^{\mathrm{T}}\boldsymbol{x} + c = 0\right\},$$

for some $\boldsymbol{A}$, $\boldsymbol{b}$ and c.

As a reminder, in Naive Bayes we assume class prior probabilities

$$\mathrm{p}(y = 0) = \boldsymbol{\pi}_0 \quad \text{and} \quad \mathrm{p}(y = 1) = \boldsymbol{\pi}_1$$

and class likelihoods

$$\mathrm{p}(\boldsymbol{x} \mid y = c) = \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$$

with per-class means $\boldsymbol{\mu}_c$ and *diagonal* (because of the feature independence) covariances $\boldsymbol{\Sigma}_c$.