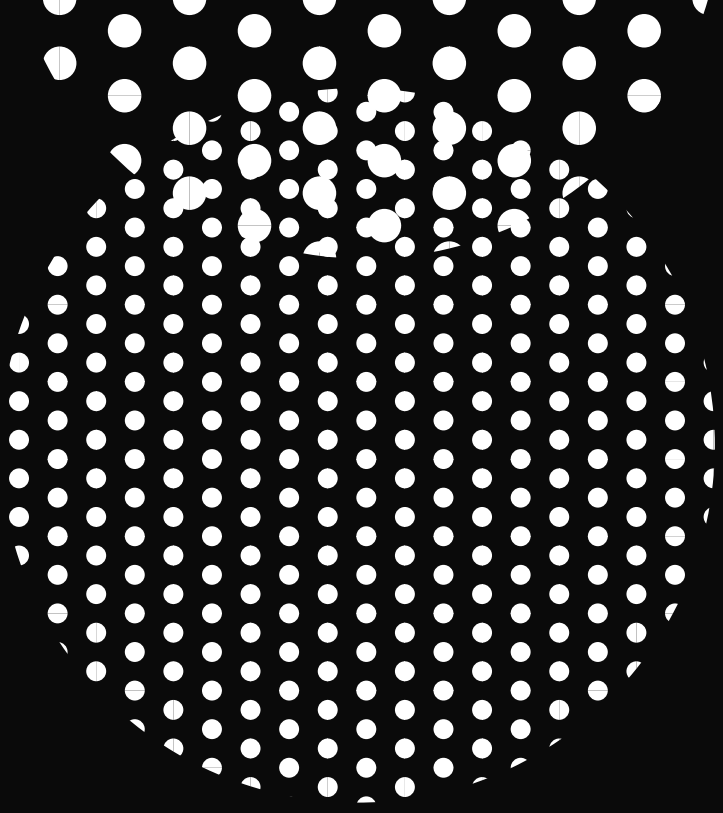


Multi-Class Classification

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.

1-hot:

C=5 →
1 → 0 0 0 0 1
2 → 0 0 0 1 0
3 → 0 0 1 0 0
⋮

$$\delta_{ij} = \begin{cases} 1 & \text{if } i=j \\ 0 & \text{otherwise} \end{cases}$$

The data likelihood given the parameters $\{\pi_c, \theta_c\}_{c=1}^C$ is

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

and so the data log-likelihood is given by

$$\log p(D \mid \{\pi_c, \theta_c\}_{c=1}^C) = \sum_{n=1}^N \sum_{c=1}^C y_c^{(n)} \log \pi_c + \text{const w.r.t. } \pi_c$$

In order to maximize the log likelihood with respect to π_c we need to preserve the constraint $\sum_c \pi_c = 1$. For this we use the method of Lagrange multipliers where we introduce λ as an unconstrained additional parameter and find a local extremum of the unconstrained function

$$\sum_{n=1}^N \sum_{c=1}^C y_c^{(n)} \log \pi_c - \lambda\left(\sum_{c=1}^C \pi_c - 1\right)$$

This function is a sum of concave terms in π_c as well as λ and is therefore itself concave in these variables.

We can find the extremum by finding the root of the derivatives. Setting the derivative with respect to π_c equal to zero, we obtain

$$\pi_c = \frac{1}{\lambda} \sum_{n=1}^N y_c^{(n)} = \frac{N_c}{\lambda}$$

Setting the derivative with respect to λ equal to zero, we obtain the original constraint

$$\sum_{c=1}^C \pi_c = 1 \quad \text{where we can now plug in the previous result } \pi_c = \frac{N_c}{\lambda}$$

and obtain $\lambda = \sum_c N_c = N$. Plugging this in turn into the expression for π_c we obtain $\pi_c = \frac{N_c}{N}$

$$p(D \mid \pi, \theta) = \prod_{n=1}^{N} p(x^n, y^n)$$

$$= \prod_{n=1}^{N} p(x^{(n)} \mid y_n, \theta) \cdot p(y_n \mid \pi)$$

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

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

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

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.

We begin by writing out the data log-likelihood

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

Then we plug in the definition of the multivariate Gaussian

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

and simplify.

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

Derivative with respect to μ_c : $\sum_{n=1}^{N} y_c^{(n)} \Sigma^{-1} (x^{(n)} - \mu_c)$, set to 0 and solve for μ_c:

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

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.

Bernoulli. y can only take values in $\{0, 1\}$, so obviously Bernoulli is the only possible answer.

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

Sample x is classified as class 1 if $p(y=1 \mid x) > p(y=0 \mid x)$. This is the same as saying $\frac{p(y=1 \mid x)}{p(y=0 \mid x)} \overset{!}{\geq} 1$ // $\frac{p(y=0 \mid x)}{p(y=1 \mid x)} \overset{!}{\geq} 0$

We begin by simplifying the left hand side

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

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

$\log(\lambda_1 \exp(-\lambda_1 x)) - \log(\lambda_0 \exp(-\lambda_0 x))$ (:) $= \log \frac{\lambda_1 \exp(-\lambda_1 x)}{\lambda_0 \exp(-\lambda_0 x)}$

$\log \lambda_1 - \lambda_1 x - \log \lambda_0 + \lambda_0 x$

$= \log \frac{\lambda_1}{\lambda_0} + \lambda_0 x - \lambda_1 x = \log \frac{\lambda_1}{\lambda_0} + (\lambda_0 - \lambda_1) x$

$\log \frac{\lambda_1}{\lambda_0} + \lambda_0 x - \lambda_1 x$

To figure out which x are classified as class 1, we need to solve for x.

$$\log \frac{\lambda_1}{\lambda_0} + (\lambda_0 - \lambda_1) x > \log 1 \Leftrightarrow (\lambda_0 - \lambda_1) x > -\log \frac{\lambda_1}{\lambda_0} = \log \lambda_0 - \log \lambda_1$$

We have to be careful, because if $(\lambda_0 - \lambda_1) < 0$, dividing by it will flip the inequality sign. Hence the answer is:

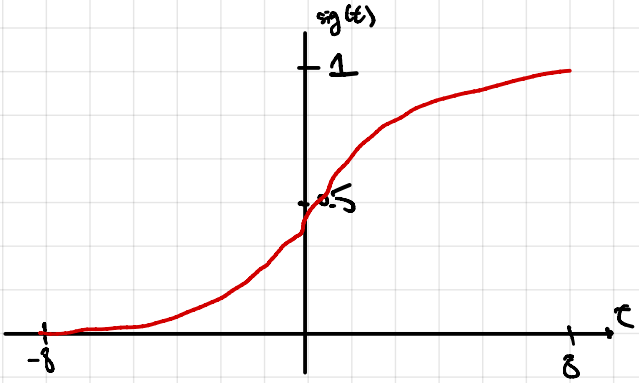
$$\begin{cases} x \in \left(\dfrac{\log \lambda_0 - \log \lambda_1}{\lambda_0 - \lambda_1}, \infty \right) & \text{if } \lambda_0 > \lambda_1 \\ x \in \left[0, \dfrac{\log \lambda_0 - \log \lambda_1}{\lambda_0 - \lambda_1} \right) & \text{otherwise} \end{cases}$$

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 sign $(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x})$ 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?

In logistic regression, we model the posterior distribution as

$$y_i \mid x \sim \text{Bernoulli}\left(\sigma(w^T x_i)\right) \quad \text{where} \quad \sigma(a) = \frac{1}{1+\exp(-a)}$$



We fit the logistic regression model by choosing the parameter w that maximizes the data log-likelihood or alternatively minimizes the negative log-likelihood which expands to

$$E(w) = -\log p(y \mid w, X) = -\sum_{i=1}^{N} y_i \log \sigma(w^T x_i) + (1-y_i)\log(1-\sigma(w^T x_i))$$

We assumed that the data-set is linearly separable, so by definition there is a $\tilde{w}$ such that $\tilde{w}^T x_i > 0$ if $y_i = 1$ and $\tilde{x}^T x_i < 0$ if $y_i = 0$

Scaling this separator $\tilde{w}$ by a factor $\lambda \gg 0$ makes the negative log-likelihood smaller and smaller. To see this, we compute the limit

$$\lim_{\lambda \to \infty} E(\lambda \tilde{w}) = -\left(\sum_{\substack{i=1 \\ y_i=1}}^{N} \log \lim_{\lambda \to \infty} \sigma(\lambda \tilde{w}^T x_i) + \sum_{\substack{i=1 \\ y_i=0}}^{N} \log\left(1 - \lim_{\lambda \to \infty} \sigma(\lambda \tilde{w}^T x_i)\right)\right) = 0$$

($\tilde{w}^T x_i > 0$ over the first sum, $\tilde{w}^T x_i < 0$ over the second)

which equals the smallest achievable value (E is the negative log of a probability, so $E(w) \in [0, \infty)$ and thus $E(w) \geq 0$)

We can see that E is convex function because log is concave and σ is convex if $a < 0$ and concave if $a > 0$. So $\log(a)$ is concave if $a > 0$ and $\log(1-\sigma(a))$ is concave if $a < 0$. It follows that E is a convex function because E is the negative sum of concave functions.

A convex function has a unique minimum if it attains its minimum value. We know that E tends towards its minimum as $\lambda \to \infty$, so E cannot have a finite minimizer and all its minima are only achieved in the limit. It follows that any solution to the loss minimization problem has infinite norm.

Because E is convex and tends towards a limit of 0 in some directions. we can move the minimum into the space of finite vectors by adding any convex term that achieves its minimum such as $w^T w$ or similar forms of weight regularization.

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

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

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

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

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

where $\hat{w} = w_1 - w_0$

One conclusion we can draw from this is that if we have C parameter vectors w_c for C classes, the logistic regression model is unidentifiable. This means that adding a constant $\tau \in \mathbb{R}^D$ to each vector $w_c := w_c + \tau$ would lead to the same logistic regression model. We can fix this issue by adding a constraint $w_1 = 0$, which is what is done implicitely when we use sigmoid in binary classification

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

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

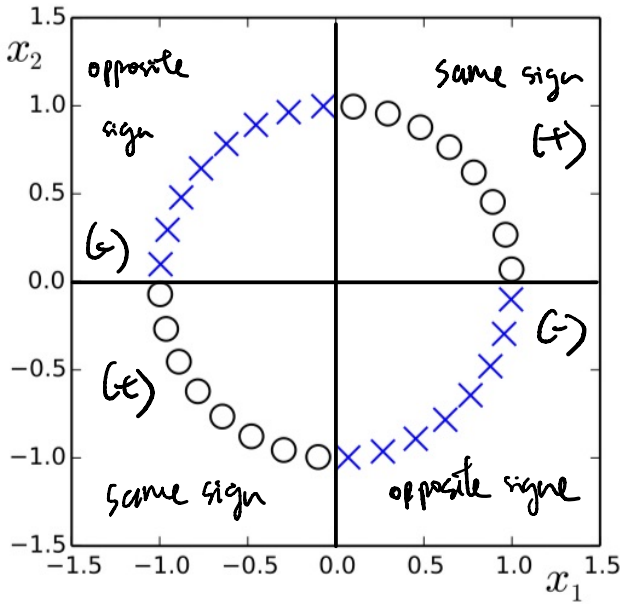
$$\frac{\partial \sigma(a)}{\partial a} = -\frac{1}{(1+e^{-a})^2} \cdot e^{-a} \cdot (-1) = \frac{1}{1+e^{-a}} \cdot \frac{e^{-a}}{1+e^{-a}} =$$

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

$$f(x) = x^{-1}$$

$$f'(x) = -1 \cdot x^{-1-1} = \frac{-1}{x^2}$$

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).



One example is $\phi(x) = x_1 x_2$ which makes the data separable by the hyperplane $w = (1)$ because the circles will be mapped to the positive real numbers while the crosses go to the negative numbers, i.e. $w^T x \geq 0$ if x is a circle and $w^T x < 0$ otherwise

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 $\boldsymbol{\Sigma}_c$.

Because $p(y=1|x) + p(y=0|x) = 1$ and we want them to be equal, we can assume that $p(y=0|x) > 0$ and rewrite $p(y=1|x) = p(y=0|x)$ as

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

$$\log 1 = \log \frac{p(y=1|x)}{p(y=0|x)} \;;\quad 0 = \log\left(\frac{p(x|y=1)\cdot p(y=1)}{p(x)} \cdot \frac{p(x)}{p(x|y=0)\cdot p(y=0)}\right)$$

$$= \log\left(p(x|y=1)\,p(y=1)\right) - \log\left(p(x|y=0)\,p(y=0)\right)$$

$$= \log N(x|\mu_1, S_1) - \log N(x|\mu_0, S_0) + \log \pi_1/\pi_0$$

$$= -\frac{1}{2}\log(2\pi)^D|S_1| - \frac{1}{2}(x-\mu_1)^T S_1^{-1}(x-\mu_1) + \frac{1}{2}\log(2\pi)^D|S_0| + \frac{1}{2}(x-\mu_0)^T S_0^{-1}(x-\mu_0)$$

$$+ \log \pi_1/\pi_0 = -\frac{1}{2}x^T S_1^{-1} x + x^T S_1^{-1}\mu_1 - \frac{1}{2}\mu_1^T S_1^{-1}\mu_1$$

$$+ \frac{1}{2}x^T S_0^{-1} x - x^T S_0^{-1}\mu_0 + \frac{1}{2}\mu_0^T S_0^{-1}\mu_0 + \frac{1}{2}\log\frac{|S_0|}{|S_1|} + \log\frac{\pi_1}{\pi_0}$$

$$= \frac{1}{2}x^T\left[S_0^{-1} - S_1^{-1}\right]x + x^T\left[S_1^{-1}\mu_1 - S_0^{-1}\mu_0\right] - \frac{1}{2}\mu_1^T S_1^{-1}\mu_1 + \frac{1}{2}\mu_0^T S_0^{-1}\mu_0 + \log\frac{\pi_1}{\pi_0} + \frac{1}{2}\log\frac{|S_0|}{|S_1|}$$

This shows that Γ is quadratic and can be alternatively be written as

$$\Gamma = \{x \mid x^T A x + b^T x + c = 0\}$$

where

$$A = \frac{1}{2}\left[\Sigma_0^{-1} - \Sigma_1^{-1}\right] \qquad b = \Sigma_1^{-1}\mu_1 - \Sigma_0^{-1}\mu_0$$

$$c = -\frac{1}{2}\mu_1^T \Sigma_1^{-1}\mu_1 + \frac{1}{2}\mu_0^T \Sigma_0^{-1}\mu_0 + \log\frac{\pi_1}{\pi_0} + \frac{1}{2}\log\frac{|\Sigma_0|}{|\Sigma_1|}$$

If both classes had the same covariance matrix ($\Sigma_0 = \Sigma_1$), A would be the zero matrix and we would obtain a linear decision boundary as we did in the lecture (also, $\log\frac{|\Sigma_0|}{|\Sigma_1|} = 0$)