EPFL, Autumn 2025
Yoav Zemel
Guillaume Obozinski

Ho Yun

Statistical Machine Learning

Exercise sheet 6

Exercise 6.1 (Multiclass Logistic Regression) In this exercise we derive a multiclass generalization of logistic regression. We shall assume that the input variable X is a vector in $\mathbb{R}^p$ as before, but the output variable Y is a vector of the form $y = (y_1, \ldots, y_K) \in \{0,1\}^K$ with $\sum_{k=1}^K y_k = 1$. Thus, the vector y such that $y_k = 1$ for $k = m$ and 0 otherwise, corresponds to the class m. It follows that the probability of X being in class m given $X = x$ is $\mathbb{P}(Y_m = 1|X = x)$, where $Y = (Y_1, \ldots, Y_K)$.

(a) Let $w_1, \ldots, w_K \in \mathbb{R}^p$ be K vectors of parameters each associated with the corresponding class. Construct a conditional model for $Y = y|X = x$ such that $\mathbb{P}(Y_k = 1|X = x) \propto \exp(w_k^\top x)$. In particular, find $\mathbb{P}(Y_k = 1|X = x)$.

Solution: If $\mathbb{P}(Y_k = 1|X = x) = c\exp(w_k^\top x)$, then we can work out c from

$$1 = \sum_{k=1}^K \mathbb{P}(Y_k = 1|X = x) = c\sum_{k=1}^K \exp(w_k^\top x)$$

Thus,

$$\mathbb{P}(Y = y|X = x) = \frac{\exp\left(\sum_{k=1}^K y_k w_k^\top x\right)}{\sum_{j=1}^K \exp(w_j^\top x)}$$

and in particular,

$$\mathbb{P}(Y_k = 1|X = x) = \frac{\exp(w_k^\top x)}{\sum_{j=1}^K \exp(w_j^\top x)}$$

(b) Show that when $K = 2$, the proposed model is equivalent to logistic regression, except that the model is over-parameterized, and therefore w_1 and w_2 are not identifiable. Is this a problem?

Solution: If $K = 2$,

$$\mathbb{P}(Y_1 = 1|X = x) = \frac{\exp(w_1^\top x)}{\sum_{j=1}^2 \exp(w_j^\top x)} = \sigma((w_1 - w_2)^\top x)$$

which matches binary logistic regression for $w = w_1 - w_2$. The overparameterization is not a problem because we don't care about estimating the parameters themselves, but just the predictive model here.

(c) Show that the model is still overparametrized if $K > 2$ and that one can impose the constraint $\sum_{k=1}^K w_k = 0$.

Solution: Let $\overline{w} = \frac{1}{K}\sum_{k=1}^K w_k$ and assume the original model then

$$\mathbb{P}(Y = y|X = x, \{w_j\}_{j=1}^K) = \frac{\exp\left(\sum_{k=1}^K y_k w_k^\top x\right)\exp(-\overline{w}^\top x)}{\sum_{j=1}^K \exp(w_j^\top x)\exp(-\overline{w}^\top x)} = \frac{\exp\left(\sum_{k=1}^K y_k (w_k - \overline{w})^\top x\right)}{\sum_{j=1}^K \exp((w_j - \overline{w})^\top x)},$$

because $\sum_{k=1}^K y_k = 1$. In other words,

$$\mathbb{P}(Y = y|X = x, \{w_j\}_{j=1}^K) = \mathbb{P}(Y = y|X = x, \{w_j - \overline{w}\}_{j=1}^K)$$

Thus, replacing every w_j with $\tilde{w}_j = w_j - \overline{w}$ yields the same model.

(d) Express $\mathbb{P}(Y_k = 1|Y_k + Y_j = 1, X = x)$, or alternatively, derive the log-odds between two classes. What is the shape of $\{x \mid \mathbb{P}(Y_k = 1|X = x) = \mathbb{P}(Y_j = 1|X = x)\}$? Deduce that the region of space where class k is most likely is a polyhedron.

Solution: Notice that,

$$\begin{aligned}\mathbb{P}(Y_k = 1|Y_k + Y_j = 1, X = x) &= \tfrac{\mathbb{P}(Y_k=1|X=x)}{\mathbb{P}(Y_k+Y_j=1|X=x)} = \tfrac{\mathbb{P}(Y_k=1|X=x)}{\mathbb{P}(Y_k=1|X=x)+\mathbb{P}(Y_j=1|X=x)} \\ &= \tfrac{\exp(w_k^\top x)}{\exp(w_k^\top x)+\exp(w_j^\top x)} = \sigma\big((w_k - w_j)^\top x\big)\end{aligned}$$

Therefore,

$$\mathbb{P}(Y_k = 1|X = x) \geq \mathbb{P}(Y_j = 1|X = x) \quad \Leftrightarrow \quad (w_k - w_j)^\top x \geq 0$$

so the region in which $\mathbb{P}(Y_k = 1|X = x) \geq \mathbb{P}(Y_j = 1|X = x)$ is the half-space $\{x \mid w_k^\top x \geq w_j^\top x\}$. Since, $\mathbb{P}(Y_k = 1|X = x) \geq \max_{j\neq k} \mathbb{P}(Y_j = 1|X = x)$ if and only if $\mathbb{P}(Y_k = 1|X = x) \geq \mathbb{P}(Y_j = 1|X = x)$ for every $j \neq k$ the region where class k is most likely is the set $\{x \mid w_k^\top x \geq \max_{j\neq k} w_j^\top x\} = \cap_{j\neq k}\{x \mid w_k^\top x \geq w_j^\top x\}$ which is an intersection of half-space and thus a polyhedron.

(e) Assume that we have a sample $\{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$ with $(x^{(i)} \in \mathbb{R}^p$ and $y^{(i)}$ an indicator vector. Write the negative (conditional) log-likelihood of the sample and show that it can be interpreted as the empirical risk associated with a loss function that you will specify $\ell : \mathbb{R}^K \times \{0, 1\}^K \to \mathbb{R}$ applied to a predictor $f(x)$ of the form $f(x) = (f_1(x), \ldots, f_K(x))$ with $f_k(x) = w_k^\top x$.

Solution: The negative log-likelihood is equal to $n\widehat{\mathcal{R}}_n(W)$ with

$$\widehat{\mathcal{R}}_n(W) = -\tfrac{1}{n}\sum_{i=1}^n \Big[\sum_{k=1}^K y_k^{(i)} w_k^\top x^{(i)} - \log\Big(\sum_{k=1}^K \exp(w_k^\top x^{(i)})\Big)\Big]$$

The corresponding loss is $\ell(a, y) = -y^\top a + \log\big(\sum_{k=1}^K e^{a_k}\big)$. Note that $-\ell(a, y)$ is the log-likelihood of multinomial variable y as a function of its canonical parameter. We have thus parameterized the canonical parameter of the exponential family corresponding to the multinomial model as a linear function of x.

(f) How would you apply Tikhonov regularization to the corresponding empirical risk?

Solution: Just solve

$$\min_W \widehat{\mathcal{R}}_n(W) + \lambda \sum_{k=1}^K \|w_k\|_2^2$$

The bias induced by the regularization is to pull all vectors w_k towards 0 which is implicitly “pulling" the probabilities towards uniform probabilities over classes.

(g) Since the model is over-parameterized, instead of using the strategy proposed in (c), one could propose to just set $w_K = 0$. Prove that this would yield an equivalent model.

Solution: As in (c), notice that

$$\mathbb{P}(Y = y|X = x, \{w_j\}_{j=1}^K) = \mathbb{P}(Y = y|X = x, \{w_j - w_K\}_{j=1}^K)$$

Thus, every model with the parameters $W = \{w_j\}_{j=1}^K$ is equivalent to one with the parameters $\{\tilde{w}_j\}_{j=1}^K$, where $\tilde{w}_j = w_j - w_K$ and thus the last parameter $\tilde{w}_K = 0$.

(h) Now, if we use Tikhonov regularization, why is the option to set $w_K = 0$ not such a good idea?

Solution: By (g) this would be equivalent to solving

$$\min_W \widehat{\mathcal{R}}_n(W) + \lambda \sum_{k=1}^K \|w_k - w_K\|_2^2$$

In that case the bias induced by the regularization brings all class vectors towards class K which is likely to bias specifically towards that class, and there does not seem that there should be any reason to do this in general.

(i) Why is the option proposed in (c) better? If we regularize with Tikhonov regularization and don't enforce the constraint $\sum_{k=1}^K w_k = 0$, what happens?

Solution: The option proposed in (c) is more symmetric. If we regularize and don't enforce the constrain the regularization should implicitly leads to a set of parameters with 0 mean, because of the decomposition of variance formula:

$$\sum_{k=1}^K \|w_k\|_2^2 = \sum_{k=1}^K \|w_k - \overline{w}\|_2^2 + K\|\overline{w}\|_2^2,$$

shows that decreasing $\|\overline{w}\|_2^2$ decrease Tikhonov regularization. As a matter of fact even without regularization, if the parameters w_k are initial set to 0, and any first or second order descent algorithms are used to minimize the empirical risk the iterates will be such that $\overline{w} = 0$ throughout because the partial gradient of the likelihood is such that $\sum_{k=1}^K \frac{\partial \widehat{\mathcal{R}}_n(W)}{w_k} = 0$.

Practical exercises

Exercise 6.2 (Implementation of the LDA and QDA algorithms and comparison with logistic regression) The files `classificationA.train`, `classificationB.train` and `classificationC.train` contain samples of data (x_i, y_i) where $x_i \in \mathbb{R}^2$ and $y_i \in \{0, 1\}$ (each line of each file contains the 2 components of x_i then y_i.). The goal of this exercise is to implement linear classification methods and to test them on the three data sets.

(a) For each data set (A,B,C) represent graphically the training data as a point cloud in $\mathbb{R}^2$ using different markers for the two classes using `ggplot`,

```
#Input
inp <- scan("classificationA.train", list(x1=0,x2=0,y=0))
inp <- data.frame(x1=inp$x1,x2=inp$x2,y=inp$y)
#Base Plot
G <- ggplot(data = inp, mapping = aes(x = x1, y = x2,
    color = as.factor(y)))+ geom_point()
```

(b) Apply LDA and compute the MLE estimates for all the parameters. Plot the classification boundary for LDA for each data set by completing the following `R` code by entering correct values for `b` and `S` which are determined by the fact that the boundary is given by $\texttt{w}[1]\texttt{x1} + \texttt{w}[2]\texttt{x2} + \texttt{b} = 0$.

```
#LDA Boundary Parameters
b <- << ENTER b HERE >>
w <- << ENTER w HERE >>

#LDA Boundary Function: x2 = (- x1*w[1] + b)/w[2]
LDAcurve <- function(x) {
    (- x*w[1] + b)/w[2]
}
#Plot with LDA boundary
Graph_LDA <- G
    + stat_function(fun = LDAcurve, color = "black")
```

(c) On a separate figure, plot the classification boundary for QDA, on top of the data for each data set by completing the following code. Because we do not have a handy expression for the graph of the boundary, say $\texttt{f}(\texttt{x1}, \texttt{x2}) = 0$, we shall draw it as a contour of $\texttt{f}(\texttt{x1}, \texttt{x2}) = \texttt{z}$. Enter the expression from $\texttt{f}(\texttt{x1}, \texttt{x2})$ below.

```
#QDA Contour
cont_QDA <- curve3d(<< ENTER FUNCTION f(x1,x2) HERE >>,
from = c(-6,-6), to = c(6,6), n=c(100,100),
sys3d="none")
dimnames(cont_QDA$z) <- list(cont_QDA$x,cont_QDA$y)
M_QDA <- reshape2::melt(cont_QDA$z)
#Plot with QDA boundary
Graph_QDA <- G
```

```
+ geom_contour(data=M_QDA,
aes(x=Var1,y=Var2,z=value),
breaks=0,linejoin = "round",colour="black")
```

(d) Run logistic regression using the `glm` function in `R`, and make again a similar plot with the data and the decision boundary using `stat_function` in `ggplot`.

```
#Logistic Regression
logres <- glm(y ~ x1 + x2, data = inp, family = binomial)
summary(logres)$coef

#Logistic Regression Coefficients
m1 <- summary(logres)$coef[[2,1]] #Coefficient of x1
m2 <- summary(logres)$coef[[3,1]] #Coefficient of x2
mc <- summary(logres)$coef[[1,1]] #Constant term

#Logistic Regression Boundary Function: f(x) =  -(mc + m1 * x1)/m2
logcurve <- function(x) {
<< ENTER CODE HERE >>
}
```

Are the coefficients very large? If so, why?

Solution: The coefficients are large when the data is linearly separable, because under such conditions, the values of parameters w and b given by maximum likelihood estimation are infinity.

(e) Do the same visualizations on the three testing data sets.

(f) Compute the misclassification error of all three methods on all the three training sets and their corresponding testing sets. Which method performs better and why?
Solution:

Percentage Misclassification Error						
Method	Train A	Test A	Train B	Test B	Train C	Test C
LDA	1.3	2	3	4.1	5.5	4.2
QDA	0.6	2	1.3	2	5.25	3.8
Log. Reg.	0	3.4	2	4.3	4	2.3

Notice that logistic regression overfits in the case of dataset A. This is because the data is linearly separable.
Case A: Both the LDA and QDA perform better than logistic regression. This is sobecause the data has been generated by a LDA model. Additionally, the training error of QDA is less than that of LDA because every LDA is also a QDA.
Case B: QDA beats the other two methods. This is because the data has been generated by the a QDA model.
Case C: On visulizing the data, one notices that there are three clusters of points instead of just two. This can not happen for a QDA or LDA. For this reason, the logistic regression model performs the best.