COQUELIN Pierre
CHERON Guilhem

MVA "Kernel methods"
HW 4

Exercice 1

a) $$\min_{\beta\in\mathbb{R}^p}\left\{\sum_{i=1}^{n}\log\left(1+e^{-y_i\beta^T x_i}\right)+\lambda\|\beta\|_2^2\right\},\quad x_i\in\mathbb{R}^p,\ y_i\in\{-1,1\}$$

$$\log(1+\exp(-y_i\beta^T x_i))=\log\left(1+\exp\left(-y_i\log\left(\frac{P_\beta(Y=1|X=x_i)}{P_\beta(Y=-1|X=x_i)}\right)\right)\right)$$

$$=\log\left(1+\left(\frac{P_\beta(Y=1|X=x_i)}{P_\beta(Y=-1|X=x_i)}\right)^{-y_i}\right)$$

but, $$\left(\frac{P_\beta(Y=1|X=x_i)}{P_\beta(Y=-1|X=x_i)}\right)^{-y_i}=\begin{cases}\dfrac{P_\beta(Y=-1|X=x_i)}{P_\beta(Y=1|X=x_i)} & \text{if } y_i=1\\[2ex] \dfrac{P_\beta(Y=1|X=x_i)}{P_\beta(Y=-1|X=x_i)} & \text{if } y_i=-1\end{cases}$$

$$=\frac{P_\beta(Y=-y_i|X=x_i)}{P_\beta(Y=y_i|X=x_i)}$$

$$=\frac{1-P_\beta(Y=y_i|X=x_i)}{P_\beta(Y=y_i|X=x_i)}$$

then,

$$\log(1+\exp(-y_i\beta^T x_i))=\log\left(1+\frac{1-P_\beta(Y=y_i|X=x_i)}{P_\beta(Y=y_i|X=x_i)}\right)$$

$$=\log\left(\frac{P_\beta(Y=y_i|X=x_i)+1-P_\beta(Y=y_i|X=x_i)}{P_\beta(Y=y_i|X=x_i)}\right)$$

$$=-\log\left(P_\beta(Y=y_i|X=x_i)\right)$$

Then we have,

$$\min_{\beta\in\mathbb{R}^p}\left\{\sum_{i=1}^{n}-\log\left(P_\beta(Y=y_i|X=x_i)\right)+\lambda\|\beta\|_2^2\right\},$$ which is equivalent to $$\max_{\beta\in\mathbb{R}^p}\left\{\sum_{i=1}^{n}\log\left(P_\beta(Y=y_i|X=x_i)\right)-\lambda\|\beta\|_2^2\right\}$$

b) $\beta^T x_i = \sum_{j=1}^{n} \alpha_j \, K(x_i, x_j)$ $\Rightarrow \begin{pmatrix} \beta^T x_1 \\ \vdots \\ \beta^T x_n \end{pmatrix} = K\alpha$

$\|\beta\|_2^2 = \alpha^T K \alpha$ (Representer theorem with linear kernel)

Then we get the new problem,

$$\min_{\alpha \in \mathbb{R}^n} \left\{ R(K\alpha) + \lambda \alpha^T K \alpha \right\}$$

with $R(x) = \sum_{i=1}^{n} \log(1 + \exp(-y_i x_i))$, here we see that we find the closed and convex risk of the kernel logistic regression.

Let $\alpha^* \in \mathbb{R}^n$ be the solution of this problem.

If we choose K the kernel Gram matrix with entries $K_{ij} = x_i^T x_j$ then we get:

$$\beta^T x_i = \sum_{j=1}^{n} \alpha_j x_j^T x_i \Rightarrow \beta = \sum_{i=1}^{n} \alpha_i x_i$$

Then the solution of the first problem is:

$$\beta^* = \sum_{i=1}^{n} \alpha_i^* x_i$$

c) Rewritting the problem as:

$$\min_{\alpha \in \mathbb{R}^n, u \in \mathbb{R}^n} \left\{ R(u) + \lambda \alpha^T K \alpha \right\} \quad \text{s.t } K\alpha = u$$

Its Lagragian is,

$$L(\alpha, u, \nu) = R(u) + \lambda \alpha^T K \alpha + \nu^T (K\alpha - u)$$

Thus we have,

$$g(\nu) = \inf_{\alpha, u} L(\alpha, u, \nu) = \inf_{\alpha, u} \left\{ R(u) + \lambda \alpha^T K \alpha + \nu^T (K\alpha - u) \right\}$$

We have

$$\inf_u \left(R(u) - \nu^T u \right) = -\sup_u \left(\nu^T u - R(u) \right) = -R^*(\nu)$$

Taking the gradient w.r.t α and setting it to zero:

$$\nabla_\alpha g(\nu) = 2\lambda \alpha^T K + \nu^T K = 0$$

$$\Leftrightarrow \alpha = -\frac{\nu}{2\lambda} + c, \quad c \in \mathrm{Ker}(K) \text{ and we take } c = 0$$

Then $g(\nu) = -R^*(\nu) + \lambda \times -\frac{\nu^T}{2\lambda} K \times -\frac{\nu}{2\lambda} + \nu^T K \times -\frac{\nu}{2\lambda}$

$$= -R^*(\nu) - \frac{\nu^T K \nu}{4\lambda}$$

$$R^*(\nu) = \sup_{x \in \mathbb{R}^n} \left\{ \langle x, \nu \rangle - R(x) \right\}$$

$$= \sup_{x \in \mathbb{R}^n} \left\{ \sum_{i=1}^n x_i \nu_i - \log\left(1 + \exp(-y_i x_i)\right) \right\}$$

$$= \sum_{i=1}^n \sup_{x_i \in \mathbb{R}} \left\{ x_i \nu_i - \log\left(1 + \exp(-y_i x_i)\right) \right\}$$

$$= \sum_{i=1}^n \sup_{x_i \in \mathbb{R}} \left\{ x_i \nu_i - \ell_L(y_i, x_i) \right\}$$

$= \sum_{i=1}^n \ell_L^*(y_i, \nu_i)$, with ℓ_L which is the logistic loss and ℓ_L^* is its Fenchel-Legendre transform

We know that,

$$\ell_L^*(y_i, \nu_i) = \begin{cases} (-y_i\nu_i)\log(-y_i\nu_i) + (1+y_i\nu_i)\log(1+y_i\nu_i) & \text{if } -1 \le y_i\nu_i \le 0 \\ +\infty & \text{otherwise} \end{cases}$$

Then we obtain the dual problem

$$\max_{\nu \in \mathbb{R}^m} -R^*(\nu) - \frac{\nu^T K \nu}{4\lambda}$$

$$= \max_{\nu \in \mathbb{R}^m} \sum_{i=1}^m (y_i\nu_i)\log(-y_i\nu_i) - (1+y_i\nu_i)\log(1+y_i\nu_i) - \frac{\nu^T K \nu}{4\lambda}$$

$$\text{s.t} \quad -1 \le y_i\nu_i \le 0, \; i = 1, .., m$$

If ν^* is the solution of the dual problem, then we get $\alpha^* = -\frac{\nu^*}{2\lambda}$

d. * The objective of problem (1) is:

$$C_1(\beta) = \sum_i^n \log\left(1+e^{-y_i\beta^T x_i}\right) + \lambda \|\beta\|_2^2$$

$$= -\sum_i^n \log g(y_i \beta^T x_i) + \lambda \|\beta\|_2^2$$

With $g(x) = \frac{1}{1+\exp(-x)}$ the sigmoid function.

We have

$$\nabla_\beta \left(\log g(y_i\beta^T x_i)\right) = \frac{1}{g(y_i\beta^T x_i)} g(y_i\beta^T x_i)(1-g(y_i\beta^T x_i))(y_i x_i)$$

$$= y_i x_i \left(1-g(y_i\beta^T x_i)\right)$$

Because $\frac{\partial g(x)}{\partial x} = g(x)(1-g(x))$.

Thus the gradient of $C_1(\beta)$ is

$$\nabla_\beta C_1(\beta) = -\sum_i^n y_i x_i \left(1-g(y_i\beta^T x_i)\right) + 2\lambda\beta$$

We have

$$\frac{\partial^2 y_i x_i g(y_i\beta^T x_i)}{\partial\beta_p \partial\beta_k} = x_i(p)\left(g(y_i\beta^T x_i)(1-g(y_i\beta^T x_i)\right) x_i(k) y_i^2$$

Thus the Hessian of $C_1(\beta)$ is

$$\left(\nabla_\beta^2 C_1(\beta)\right)_{pk} = \sum_i^n x_i(p) x_i(k) \left(g(y_i\beta^T x_i)(1-g(y_i\beta^T x_i)\right)$$

$$+ 2\lambda \mathbb{1}_{\{p=k\}}$$

* The objective of problem (2) is

$$C_2(\alpha) = \sum_i^n \log\left(1+\exp\left(-y_i \sum_j^n \alpha_j K_{ij}\right)\right) + \lambda \alpha^T K \alpha$$

$$= -\sum_i^n \log\left(g\left(y_i \sum_j^n \alpha_j K_{ij}\right)\right) + \lambda \alpha^T K \alpha$$

We have

$$\frac{\partial \log g\left(y_i \sum_j \alpha_j K_{ij}\right)}{\partial \alpha_k} = \left(1 - g\left(y_i \sum_j^n \alpha_j K_{ij}\right)\right) y_i K_{ik}$$

Thus the gradient is

$$\left(\nabla_\alpha C_2(\alpha)\right)_k = -\sum_i \left(1 - g\left(y_i \sum_j^n \alpha_j K_{ij}\right)\right) y_i K_{ik} + 2\lambda (K\alpha)_k$$

And the Hessian is

$$\left(\nabla_\alpha^2 C_2(\alpha)\right)_{kp} = \sum g\left(y_i \sum_j^n \alpha_j K_{ij}\right)\left(1 - g\left(y_i \sum_j^n \alpha_j K_{ij}\right)\right) K_{ik} K_{ip} + 2\lambda K_{kp}$$

* The objective of problem (3) is

$$C_3(r) = \sum y_i r_i \log(-y_i r_i) - (1+y_i r_i)\log(1+y_i r_i) - \frac{r^T K r}{4\lambda}$$

We have

$$\frac{\partial C_3(r)}{\partial r_i} = y_i \log(-y_i r_i) + y_i - y_i \log(1+y_i r_i) - y_i - \frac{(Kr)_i}{2\lambda}$$

$$= y_i \log\left(\frac{-y_i r_i}{1+y_i r_i}\right) - \frac{(Kr)_i}{2\lambda}$$

and

$$\left(\nabla_r^2 C_3(r)\right)_{ij} = 1\!\!1_{\{i=j\}} \left(\frac{1}{y_i r_i} - \frac{1}{1+y_i r_i}\right) - \frac{K_{ij}}{2\lambda}$$

We implemented a simple Newton-Raphson algorithm and the gradient and Hessian matrix of each problem in MATLAB.

Data is generated randomly using Gaussian distributions with random covariance matrix generated using Wishart distribution. The data is split into a train set and a test set.

The parameter λ is determined using a 5-fold cross validation on the training set. For optimization problems (2) and (3) we used both a linear kernel and a Gaussian kernel with $\sigma = 10$.

Problem (3) is the only one with constraints. After a Newton-Raphson iteration, we set the variables ν_i that don't respect these constraints at the closest frontier of the constraint set.

The figures below display the objective function and prediction error in function of Newton-Raphson iterations. Each figure display the results obtained with the various methods, for given values of n and p.

We observe the following points:

- As expected, results obtained with problem (1) and (2) are the same when using a linear kernel (red and yellow curves on the figures below are superimposed).
- Optimization problem (2) needs fewer iterations to converge than problem (3)
- The highest computation cost source is the inversion of the Hessian matrix. Thus, when the number of points n is superior to the dimension p, problem (2) and (3) have a higher computation cost than problem (1). Indeed hessians of problems (2) and (3) have the same size as the Gram matrix (number of points) while hessian of problem (1) size's is p x p.

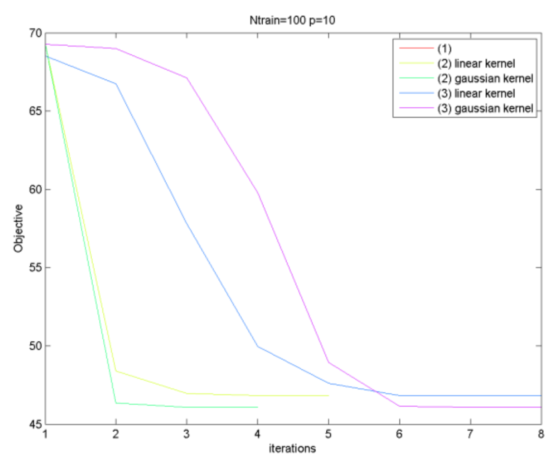

Figure 1 Objective in function of iterations for n=100, p=10

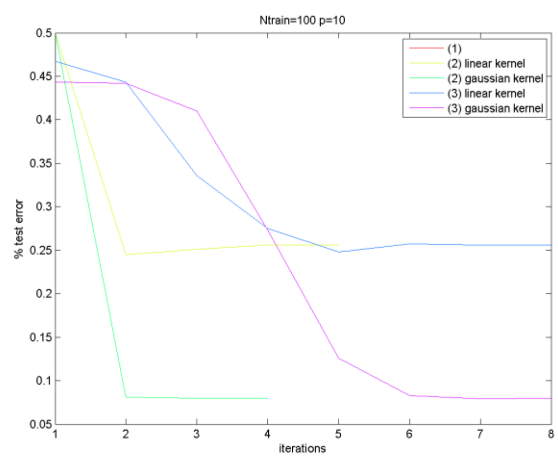

Figure 2 Percentage of error on the test set for n=100, p=10

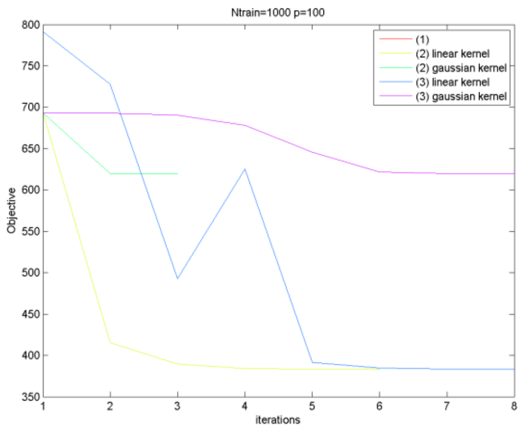

Figure 3 Objective in function of iterations for n=1000, p=100

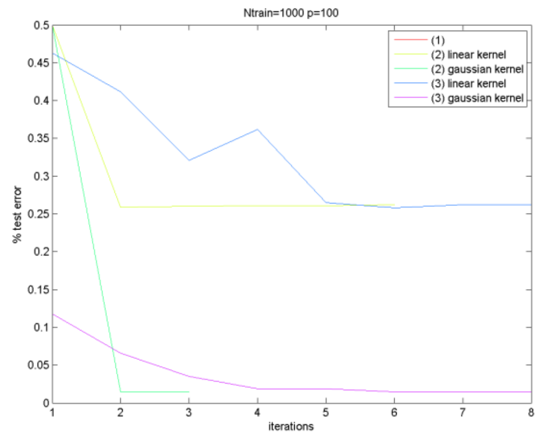

Figure 4 Percentage of error on the test set for n=1000, p=100