MVA: Convex Optimization (2020/2021) **Homework 3**

Lasso Regression

Lecturer: *A. d'Aspremont* *(November 15, 2020)*

Solution by Clément Bonnet

1 Dual Problem of LASSO

Let $\lambda > 0$.

$$\underset{w \in \mathrm{R}^d}{\text{minimize}} \quad \frac{1}{2}\|Xw - y\|_2^2 + \lambda\|w\|_1 \iff \begin{array}{ll} \underset{w \in \mathrm{R}^d, z \in \mathrm{R}^n}{\text{minimize}} & \frac{1}{2}\|z\|_2^2 + \lambda\|w\|_1 \\ \text{subject to} & z = Xw - y \end{array}$$

Let L be the Lagrangian function associated to the minimization problem.

$$\mathrm{L}(z, w, v) = \frac{1}{2}\|z\|_2^2 + \lambda\|w\|_1 + v^T(z - Xw + y)$$

$\mathrm{L}(z, w, v)$ is a quadratic form with respect to z. Its minimum is attained for $z = -v$.

$$\min_z \mathrm{L}(z, w, v) = -\frac{1}{2}v^T v + v^T y + \lambda\|w\|_1 - v^T Xw$$

Let $g(v)$ be defined as:

$$g(v) = \min_{z,w} \mathrm{L}(z, w, v) = \min_w \min_z \mathrm{L}(z, w, v)$$

Let f be the L^1-norm and $f^\star$ be its dual, defined by $f^\star(y) = \max_x(y^T x - f(x))$. One can compute the value of $f^\star$.

$$f^\star(y) = \max_x(y^T x - \|x\|_1)$$

$$f^\star(y) = \begin{cases} +\infty & \text{if } \|y\|_\infty > 1 \\ 0 & \text{if } \|y\|_\infty \leq 1 \end{cases}$$

Coming back to the original problem,

$$\begin{aligned} g(v) &= \min_w \left[-\frac{1}{2}v^T v + v^T y + \lambda\|w\|_1 - v^T Xw \right] \\ &= -\frac{1}{2}v^T v + v^T y + \min_w \left[\lambda\|w\|_1 - v^T Xw\right] \\ &= -\frac{1}{2}v^T v + v^T y - \max_w \left[(X^T v)^T w - \lambda\|w\|_1\right] \\ &= -\frac{1}{2}v^T v + v^T y - \lambda f^\star\left(\frac{X^T v}{\lambda}\right) \end{aligned}$$

$$g(v) = \min_{z,w} \mathrm{L}(z, w, v) = \begin{cases} -\infty & \text{if } \|X^T v\|_\infty > \lambda \\ -\frac{1}{2}v^T v + v^T y & \text{if } \|X^T v\|_\infty \leq \lambda \end{cases}$$

Therefore, one can derive the dual problem of LASSO:

$$\max_p g(v) \iff \begin{array}{ll} \underset{v\in\mathbb{R}^n}{\text{maximize}} & -\frac{1}{2}v^Tv + v^Ty \\ \text{subject to} & \|X^Tv\|_\infty \leq \lambda \end{array}$$

$$\iff \boxed{\begin{array}{ll} \underset{v\in\mathbb{R}^n}{\text{minimize}} & \frac{1}{2}v^Tv + y^Tv \\ \text{subject to} & \begin{cases} X^Tv \preceq \lambda.\mathbf{1} \\ -X^Tv \preceq \lambda.\mathbf{1} \end{cases} \end{array}}$$

$$\iff \begin{array}{ll} \underset{v\in\mathbb{R}^n}{\text{minimize}} & v^TQv + p^Tv \\ \text{subject to} & Av \preceq b \end{array}$$

With $\begin{cases} Q &= \frac{1}{2}\mathrm{I} \\ p &= y \\ A &= \begin{pmatrix} X^T \\ -X^T \end{pmatrix} \\ b &= \lambda.\mathbf{1} \end{cases}$. It is indeed a quadratic program (QP).

2 Barrier Method

Initial quadratic program:

$$\begin{array}{ll} \underset{v\in\mathbb{R}^n}{\text{minimize}} & f(v) = v^TQv + p^Tv \\ \text{subject to} & Av \preceq b \end{array}$$

After including the log-barrier, the function f_t to minimize now becomes:

$$f_t(v) = tv^TQv + tp^Tv - \sum_{k=1}^m \ln(b_k - (Av)_k)$$

One can derive its gradient ∇f_t:

$$\nabla f_t(v)_i = 2t(Qv)_i + tp_i + \sum_{k=1}^m \frac{A_{k,i}}{b_k - (Av)_k}$$
$$\nabla f_t(v) = 2tQv + tp + A^T\gamma(v)$$

With $\gamma(v)_k = \left[b_k - (Av)_k\right]^{-1}$.
Finally, the hessian matrix follows from differentiating the gradient.

$$\nabla^2 f_t(v)_{i,j} = 2tQ_{i,j} + \sum_{k=1}^m \frac{A_{k,i}A_{k,j}}{(b_k - (Av)_k)^2}$$
$$\nabla^2 f_t(v) = 2tQ + A^TDA$$

With $D = diag(\alpha_1, \cdots, \alpha_m)$ and $\alpha_k = \left[b_k - (Av)_k\right]^{-2}$.
From these equations, one can implement the log-barrier method to solve the quadratic program. The code is provided in functions `centering_step(Q,p,A,b,t,v0,eps)` and `barr_method(Q,p,A,b,v0,eps)`.

3 Numerical Results

For $n = 100$ and $d = 200 > n$, $X \in \mathbb{R}^{n\times d}$ and $w \in \mathbb{R}^d$ were generated randomly. Observations y were obtained from adding a nose to the regression on X, $y = Xw + \epsilon$ with $\epsilon \sim \mathrm{N}(0, 0.1^2)$. The results

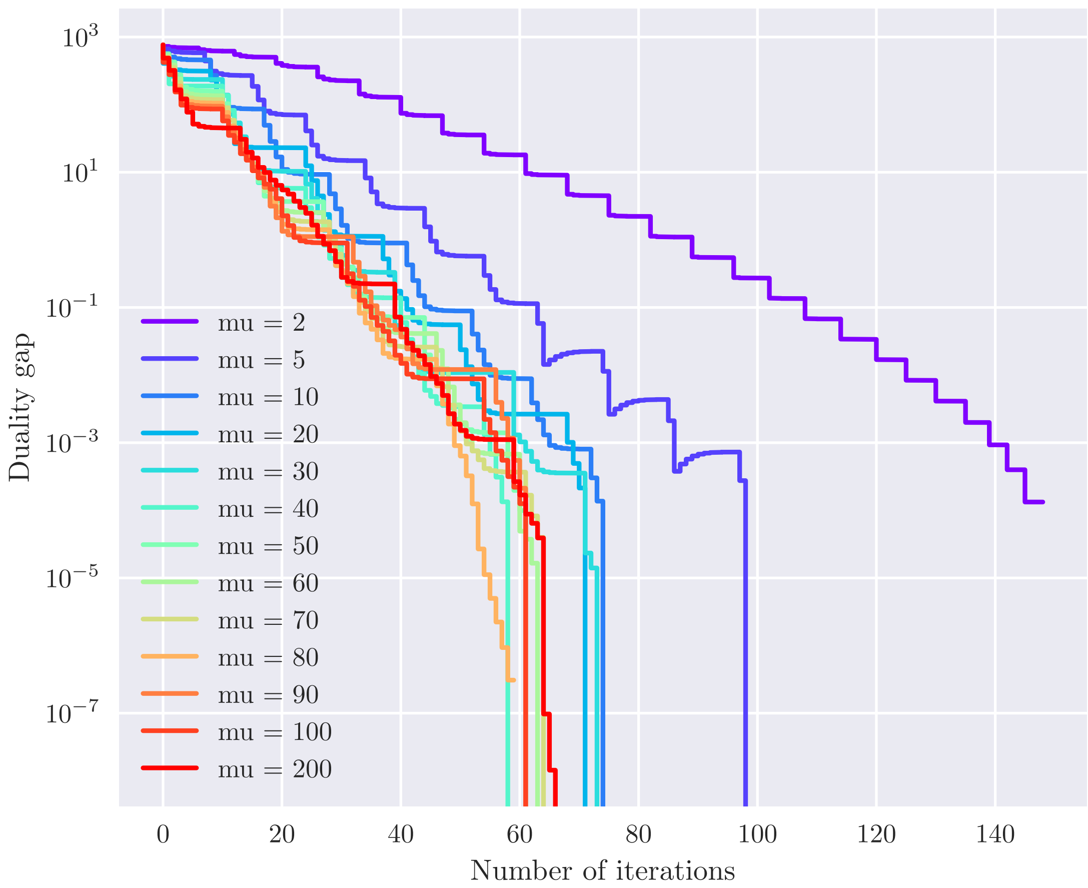

Figure 1: Duality gap with respect to the number of Newton steps.

are displayed in figure 1. Although the optimum value of μ seems to depend on the dimension d, for our experiment, one can observe that the algorithm converges in a fewer number of Newton steps with $\boxed{\mu \approx 80}$.
Once the optimal solution $v^\star$ of the dual problem is found, one can recover the optimal solution $w^\star$ of the primal using the KKT conditions.

$$Xw^\star = y - v^\star$$

Errors in recovering the true w are displayed in figure 2. One can see that the value of μ does not affect the recovered w.

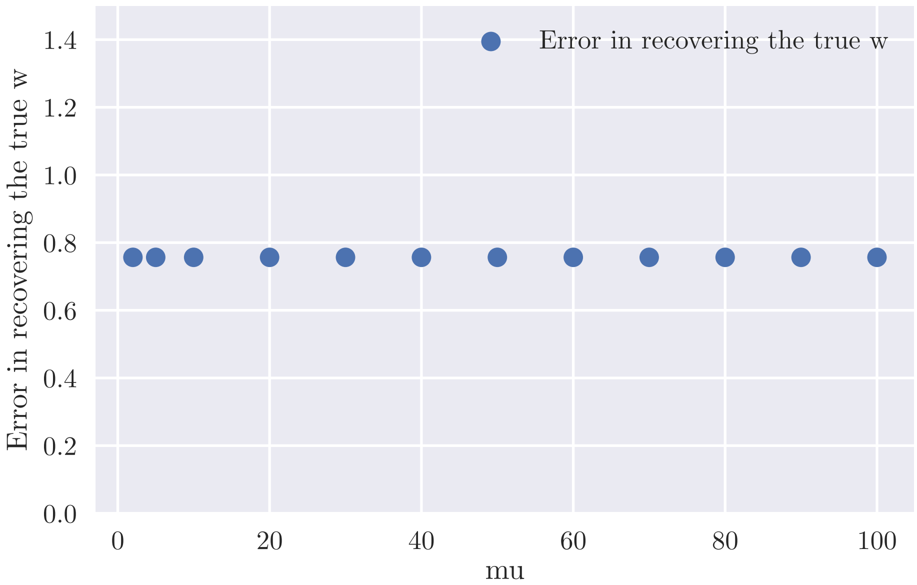

Figure 2: Recovering w does not depend on μ.