EPFL | MGT-483 : Optimal Decision Making | Project 1
Robust Regression

Anastasia Sveshnikova, Erick Maraz, Zhecho Mitev

April 2021

Question 1

It can be noticed that the outlier (red point) caused the MSE predictor to obtain a slope heavily influenced by the oulier. This behaviour is expected since the an outlier would cause a significantly big squared error. MSE is usually applied because of its nice mathematical interpretation of bias-variance trade-off.

$$MSE(\hat{\theta}) = Variance(\hat{\theta}) + (Bias(\hat{\theta}, \theta))^2 \tag{1}$$

Such formulation allows for understanding of the generality of the model. If the variance is too low and the bias too high, then the model is oversimplified. On the contrary, if variance is high and bias - low then there is overfitting as the model is too complex. Although the MAE predictor does not have such mathematical meaning, in our case it handled better the outlier because the error in this case is not squared and the outlier has not influenced greatly the overall prediction.

Question 2

1. The optimization problem presented by Section 2 is clearly a non-linear one because of the absolute value function that is used for calculating both the mean absolute error and the norm of $\boldsymbol{\theta}$. By using the auxiliary variables $\boldsymbol{u}$ and $\boldsymbol{v}$ and including four additional constraints, we can rewrite the original problem as a linear program.

$$\begin{aligned} \min_{\substack{\boldsymbol{v}, \boldsymbol{\theta} \in \mathbb{R}^D \\ \boldsymbol{u} \in \mathbb{R}^N}} \quad & \frac{1}{N} \sum_{i=1}^{N} u_i + \lambda \sum_{i=1}^{N} v_i \\ \text{s.t.} \quad & y_i - \boldsymbol{\theta}^\top \boldsymbol{x}_i \le u_i \quad \forall i = 1, \ldots, N \\ & -y_i + \boldsymbol{\theta}^\top \boldsymbol{x}_i \le u_i \quad \forall i = 1, \ldots, N \\ & \theta_i \le v_i \quad \forall i = 1, \ldots, D \\ & -\theta_i \le v_i \quad \forall i = 1, \ldots, D \end{aligned} \tag{2}$$

where $\boldsymbol{x}_i \in \mathbb{R}^D$, $\boldsymbol{y} \in \mathbb{R}^N$.

2. Implementing the linear program with the *cvxpy* library shows us that the minimum value for our problem is 120.5 when $\lambda = 0.5$. Due to the fact that the Lasso regression produces sparse vectors, all of the entries of the optimal θ are all zero except for the bias (intercept) weight. This hints that the value of λ might be too high and we must try with lower values, which is exactly the goal of task 3. Ror reference, the MAE for the training set is 79.9 and 80.3 for the test set.

3. In this task we split the dataset into a training set which contains 75% of the diabetes data points and the other 25% are located in the validation set. The logarithmically spread values of λ are in the range $[10^{-5}, 10^{-1}]$. Therefore, the first value is very small and it will probably have no regularization effect to the problem. We see that the training error is 42.446 and the validation error is 46.516 (Almost the same results appear if no regularization is applied). Gradually increasing λ makes the validation error go down and reaches a minimum at approximately $\lambda = 0.005$, where training error is 44.29 and validation - 45.154. We note that even though the training error has become higher, the validation MAE has dropped by 1.4, which is an expected effect when using Lasso. Thus, we prove that Lasso regression can perform better on unseen data, compared to a non-regularized regression method. However, increasing λ further than 0.005 makes the model oversimplified and the validation error increases again, reaching 60.8 for $\lambda = 0.1$. This shows the importance of the hyperparameter tuning technique. Using $\lambda = 0.005$ on our initial test set outputs a result of 45.758, which is significantly lower than the 80.3, obtained when $\lambda = 0.5$.

Question 3

X is the feasible set of the problem. As the original variables x_1 and x_2 are binary (1 or 0) and $x_1 + x_2 \leq 1$, X originally contains 3 (x_1, x_2) points: ((0,1), (0,0), (1,0)) (Figure 1, A). To form a convex hull, we have to transform this set into a convex set which can be represented with the following linear program:

$$\begin{aligned} \min_{x_1,x_2} \quad & -x_1 - 2x_2 \\ \text{s.t.} \quad & x_1 + x_2 \leq 1 \\ & \boldsymbol{x_1} \geq 0 \\ & \boldsymbol{x_2} \geq 0 \end{aligned} \tag{3}$$

The resulting convex hull is plotted on Figure 1, B. This figure has 3 vertices, that correspond to the points in the original set X: ((0,1), (0,0), (1,0)). The increase of cost vector is denoted as c. Point B (0, 1) is the optimal BFS of the linear program.

Question 4

The optimal decision variables are $x_1 = 0$, $x_2 = 1$. This solution is binary for both variables. In fact, all basic feasible solutions of the resulting convex hull are binary. This follows from the definition of the convex hull and its construction: all the points $x_i\theta_i$ of the resulting convex hull are supposed to lie between 2 points of the initial arbitrary set, therefore, the vertices will always correspond to the points of the initial set. In our case, as the initial variables were binary we can only find binary BFS in the convex hull as well.

Question 5

1. Our decision variables that we can adjust to minimize our loss is amount of money (x_i) that we are going to hide in every hiding spot I. The restrictions that we have is the total amount of money (X), capacity of every hiding spot C_i and obvious restriction that we cannot hide negative amount of money in any of the spots.

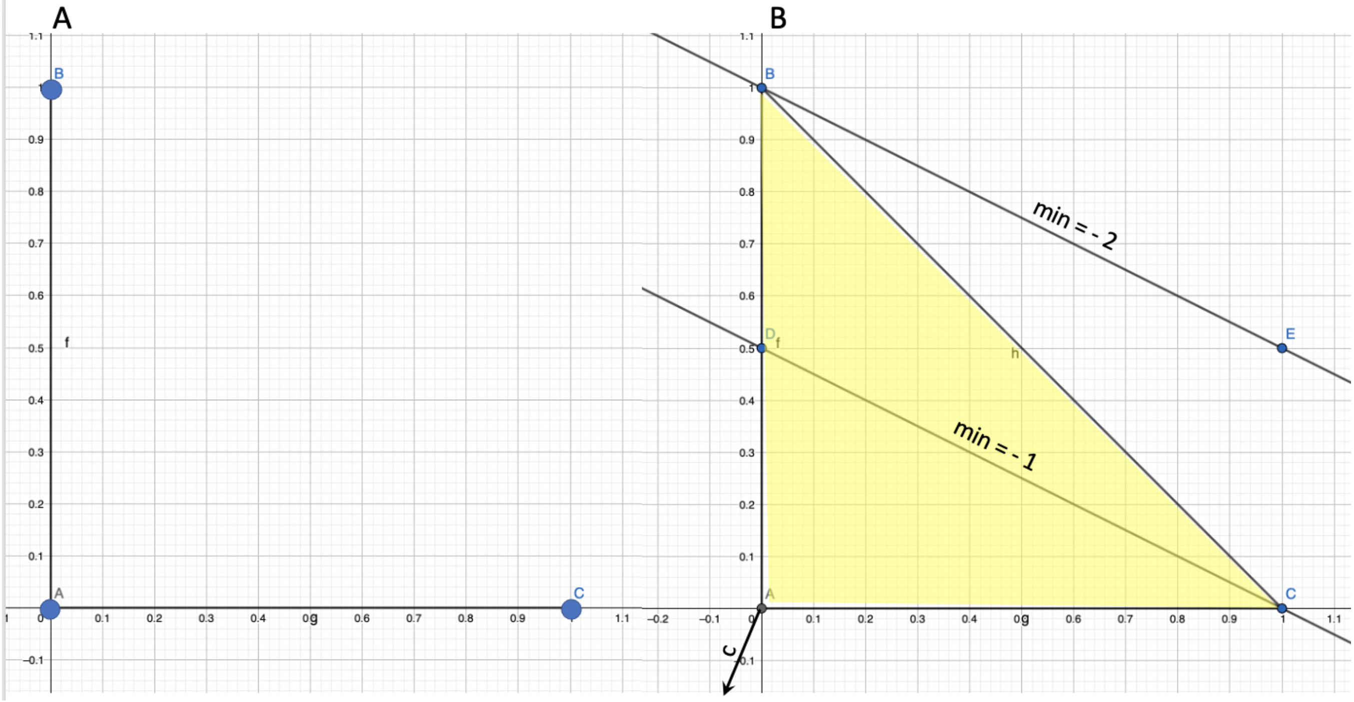

Figure 1

$$
\begin{aligned}
\min_{\boldsymbol{x}\in X} \quad & \sum_{i=1}^{I} x_i z_i p_i \\
\text{s.t.} \quad & \sum_{i=1}^{I} x_i \leq 10000 \\
& x_i \leq C_i \quad \forall i = 1, \dots, I \\
& x_i \geq 0 \quad \forall i = 1, \dots, I
\end{aligned}
\tag{4}
$$

2. The decision variable of the thief is amount of time (z_i) he is going to search every spot i of I set. The restrictions the thief has is the maximum amount of time (T), difficulty of searching every hiding spot p_i and obvious restriction that amount of time cannot be negative.

$$
\begin{aligned}
\max_{\boldsymbol{z}\in Z} \quad & \sum_{i=1}^{I} x_i z_i p_i \\
\text{s.t.} \quad & \sum_{i=1}^{I} z_i \leq T \\
& z_i \leq \frac{1}{p_i} \quad \forall i = 1, \dots, I \\
& z_i \geq 0 \quad \forall i = 1, \dots, I
\end{aligned}
\tag{5}
$$

3. Combining the decision variables and feasible sets for points 1. and 2. we get:

$$
\begin{aligned}
\min_{\boldsymbol{x}\in X} \quad & max_{\boldsymbol{z}\in Z} \sum_{i=1}^{I} x_i z_i p_i \\
\text{s.t.} \quad & \sum_{i=1}^{I} x_i = 10000 \\
& \sum_{i=1}^{I} z_i \leq T \\
& x_i \leq C_i \quad \forall i = 1, \dots, I \\
& z_i \leq \frac{1}{p_i} \quad \forall i = 1, \dots, I \\
& z_i \geq 0 \quad \forall i = 1, \dots, I \\
& x_i \geq 0 \quad \forall i = 1, \dots, I
\end{aligned} \tag{6}
$$

4. The dual of the problem with the assumption that x_i is constant corresponds to the dual of the problem (4) (decision of the thief). Transposing the matrix of the coefficients, and transforming coefficients of the variables in the optimisation problem $(x_i z_i)$ into the constraints and constraints into coefficients of the dual optimisation problem we get:

$$
\begin{aligned}
\min_{\boldsymbol{y}\in Y} \quad & \sum_{i=1}^{I} \frac{1}{p_i} y_i + T y_{I+1} \\
\text{s.t.} \quad & y_i + y_{I+1} \leq x_i p_i \quad \forall i = 1, \dots, I \\
& y_i \leq 0 \quad \forall i = 1, \dots, I+1
\end{aligned} \tag{7}
$$

Overall optimization problem turns into:

$$
\begin{aligned}
\min_{\boldsymbol{x}\in X, \boldsymbol{y}\in Y} \quad & \sum_{i=1}^{I} \frac{1}{p_i} y_i + T y_{I+1} \\
\text{s.t.} \quad & y_i + y_{I+1} \geq x_i p_i \quad \forall i = 1, \dots, I \\
& \sum_{i=1}^{I} x_i = 10000 \\
& x_i \leq C_i \quad \forall i = 1, \dots, I \\
& x_i \geq 0 \quad \forall i = 1, \dots, I \\
& y_i \geq 0 \quad \forall i = 1, \dots, I+1
\end{aligned} \tag{8}
$$

5. Our decision on hiding the money depends on the difficulty of the hiding place and its capacity. The smaller p corresponds to a better hiding place as the thief would spend more time searching it. Therefore, our hiding strategy is to locate the biggest amount of money in places with largest difficulty (smallest p). Dual variable of thief represents the best amount of money the thief can find in each place in the best case. Therefore, the worst amount of money we can lose is (400 + 500 + 1500 + 2400) = 4800 dollars.

place	capacity	p_i	optimal amount of money hidden	best amount of money thief can find
1	2000	0.2	2000	400
2	1000	0.5	1000	500
3	3000	0.5	3000	1500
4	5000	0.6	4000	2400
5	5000	0.8	0	0
6	10000	0.9	0	0

Question 6

The feasible set of the adversarial problem is defined as follows:

$$Z = \left\{ z \in \mathbb{R}^n : \sum_{i=1}^{N} z_i = k, z_i \in \{0,1\}, \forall i \in [N] \right\} \tag{9}$$

To construct the convex hull of set Z, first, we need to pick vectors in the same dimension : $\mathbb{R}^n$. Each vector should also satisfy the condition that the sum of its elements is equal to k. This is due to the fact that all z vectors belong to a single affine hyperplane, which can be defined as $\sum_{i=1}^{N} a_i z_i = k$, where $\boldsymbol{a}$ is a non-zero vector (see definition of affine hyperplane) and $a_i \in \boldsymbol{a}$. It is easy to see that the set Z belongs to this hyperplane as replacing θ with the $\mathbf{1}$ vector, will give us exactly the definition of set Z. Therefore, the linear combination of all vectors in set Z, which is its convex hull, must also be entirely defined in the same affine hyperplane. Thus we define $Conv(Z)$ as a candidate convex hull and we will explain why $Conv(Z)$ is indeed the convex hull of set Z.

$$Conv(Z) = \left\{ x \in \mathbb{R}^n : \sum_{i=1}^{N} x_i = k, x_i \geq 0, x_i \leq 1, \forall i \in [N] \right\} \tag{10}$$

To prove that $Conv(Z)$ can produce every point in the convex hull of Z , we use the fact that by definition the convex hull is produced linear combinations of the elements in Z. $Conv(Z)$ can produce every possible linear combination because we ca define each vector $x = \sum_{j=1}^{M} \theta \odot z_j$, where $z_j \in Z$, $\sum_{j=1}^{M} \theta_j = 1$, $\theta_j \geq 0$ and $M = |Z|$. Since the sum of the elements of θ equals 1 and we make an element wise multiplication between each vector z_j and θ, we can be sure that the condition $\sum_{i=1}^{N} x_i = k$ holds. Moreover , since $\theta_j \geq 0$ and $\theta_j \leq 1$, the conditions $x_i \geq 0$ and $x_i \leq 1$ will hold as well. Therefore, we can conclude that the convex hull of Z is a subset of $Conv(Z)$.

Now, we only need to prove the other direction - Conv(Z) is a subset of the convex hull. This can be proven by contradiction. Let $x^{'} \in Conv(Z)$, but not in the convex hull of Z. Due to the fact that $x^{'}$ is not in the convex hull, there must exist at least one element of $x^{'}$ which is bigger than 1 or smaller than 0. However, this is impossible, because we implicitly condition $x_i \geq 0$ and $x_i \leq 1$ in our definition. Therefore, we reach a contradiction and it is impossible that the convex hull of Z contain a point outside of $Conv(Z)$. That allows us to conclude that $Conv(Z)$ is a correct formulation of the convex hull of set Z

Question 7

1. By replacing the feasible set for the convex hull (10), we obtain the problem:

$$\begin{aligned} \min_{\boldsymbol{\theta}\in\mathbb{R}^D} \quad & \max_{\boldsymbol{z}\in\mathbb{R}^N} \quad \frac{1}{k}\sum_{i=1}^{N} z_i|y_i - \boldsymbol{\theta}^T\boldsymbol{x}_i| + \lambda\|\theta\|_1 \\ \text{s.t.} \quad & \sum_{i=1}^{N} z_i = k \\ & \quad z_i \leq 1 \quad \forall i = 1,\ldots,N \\ & \quad z_i \geq 0 \quad \forall i = 1,\ldots,N \end{aligned} \tag{11}$$

Recall that $\max f(x) = -\min -f(x)$ and note that the regularization term is just a constant for the inner problem. Therefore, Problem (11) can be written as:

$$\begin{aligned} \min_{\boldsymbol{\theta}\in\mathbb{R}^D} \quad & \lambda\|\boldsymbol{\theta}\|_1 - \min_{\boldsymbol{z}\in\mathbb{R}^N} \quad -\frac{1}{k}\sum_{i=1}^{N} z_i|y_i - \boldsymbol{\theta}^T\boldsymbol{x}_i| \\ \text{s.t.} \quad & \sum_{i=1}^{N} z_i = k \\ & \quad z_i \leq 1 \quad \forall i = 1,\ldots,N \\ & \quad z_i \geq 0 \quad \forall i = 1,\ldots,N \end{aligned} \tag{12}$$

Now, we solve the inner minimization problem of Problem (12). Using matrix notation, the primal is:

$$\begin{aligned} \min_{\boldsymbol{y}\in\mathbb{R}^N} \quad & \begin{bmatrix} -\frac{1}{k}\left|\boldsymbol{y} - X\boldsymbol{\theta}\right| \\ \mathbf{0} \end{bmatrix}^\top \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{s} \end{bmatrix} \\ \text{s.t.} \quad & \begin{bmatrix} \mathbb{I} & \mathbb{I} \\ \mathbf{1}^\top & \mathbf{0}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{s} \end{bmatrix} = \begin{bmatrix} \mathbf{1} \\ k \end{bmatrix} \\ & \begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{s} \end{bmatrix} \geq \mathbf{0} \end{aligned} \tag{13}$$

Where **1** and **0** are a column vectors, $\mathbb{I}$ is the identity matrix of the proper size and $\boldsymbol{s}$ are the slack variables.

The dual of Problem (11) is then:

$$\begin{aligned} \max_{\boldsymbol{p}\in\mathbb{R}^{N+1}} \quad & \begin{bmatrix} \mathbf{1} \\ k \end{bmatrix}^\top \boldsymbol{p} \\ \text{s.t.} \quad & \begin{bmatrix} \mathbb{I} & \mathbf{1} \\ \mathbb{I} & \mathbf{0} \end{bmatrix} \boldsymbol{p} \leq \begin{bmatrix} -\frac{1}{k}\left|\boldsymbol{y} - X\boldsymbol{\theta}\right| \\ \mathbf{0} \end{bmatrix} \end{aligned} \tag{14}$$

Replacing:

$$\boldsymbol{p} = \begin{bmatrix} -\beta \\ -\alpha \end{bmatrix}$$

Problem (14) can be now reformulated as:

$$
\begin{aligned}
\max_{\boldsymbol{\beta}\in\mathbb{R}^N,\,\alpha\in\mathbb{R}} \quad & -\mathbf{1}^\top\boldsymbol{\beta} - \alpha k \\
\text{s.t.} \quad & -\begin{bmatrix}\mathbb{I} & \mathbf{1}\end{bmatrix}\begin{bmatrix}\boldsymbol{\beta}\\ \alpha\end{bmatrix} \leq -\frac{1}{k}\left|\boldsymbol{y} - X\boldsymbol{\theta}\right| \\
& -\boldsymbol{\beta} \leq \mathbf{0}
\end{aligned}
\tag{15}
$$

By replacing the Problem (16) in Problem (12), we obtain:

$$
\begin{aligned}
\min_{\substack{\boldsymbol{\theta}\in\mathbb{R}^D\\ \boldsymbol{\beta}\in\mathbb{R}^N,\,\alpha\in\mathbb{R}}} \quad & \mathbf{1}^\top\boldsymbol{\beta} + \alpha k + \lambda\left\|\boldsymbol{\theta}\right\|_1 \\
\text{s.t.} \quad & \begin{bmatrix}\mathbb{I} & \mathbf{1}\end{bmatrix}\begin{bmatrix}\boldsymbol{\beta}\\ \alpha\end{bmatrix} \geq \frac{1}{k}\left|\boldsymbol{y} - X\boldsymbol{\theta}\right| \\
& \boldsymbol{\beta} \geq \mathbf{0}
\end{aligned}
\tag{16}
$$

Finally, we express the norm 1 as inequalities:

$$
\begin{aligned}
\min_{\substack{\boldsymbol{\theta}\in\mathbb{R}^D\\ \boldsymbol{v}\in\mathbb{R}^N\\ \boldsymbol{\beta}\in\mathbb{R}^N,\,\alpha\in\mathbb{R}}} \quad & \mathbf{1}^\top\boldsymbol{\beta} + \alpha k + \lambda\boldsymbol{v} \\
\text{s.t.} \quad & \begin{bmatrix}\mathbb{I} & \mathbf{1}\end{bmatrix}\begin{bmatrix}\boldsymbol{\beta}\\ \alpha\end{bmatrix} \geq \frac{1}{k}\left(\boldsymbol{y} - X\boldsymbol{\theta}\right) \\
& \begin{bmatrix}\mathbb{I} & \mathbf{1}\end{bmatrix}\begin{bmatrix}\boldsymbol{\beta}\\ \alpha\end{bmatrix} \geq -\frac{1}{k}\left(\boldsymbol{y} - X\boldsymbol{\theta}\right) \\
& \boldsymbol{\theta} \leq \boldsymbol{v} \\
& -\boldsymbol{\theta} \leq \boldsymbol{v} \\
& \boldsymbol{\beta} \geq \mathbf{0}
\end{aligned}
\tag{17}
$$

Our approach to find the values of z was to sort all the absolute errors (with the optimal values found in Problem (17) in increasing order and then choosing the first k indices. The MAE found with this z values and the one found with Problem (17) is the same.

2. Table 1 shows the results found using cross validation and robust cross validation. It can be seen that there is a slight improvement in the MAE (test set) using robust cross validation. Notice that the lambda shown was chosen by iterating over log-spaced possible values and picking the one that achieved the best MAE (validation set).

Estimator	λ	MAE	MAE	MAE
		Train set	Validation set	Test set
MAE+Lasso (standard cross validation)	0.007197	45.741	44.988	46.574
MAE+Lasso (robust cross validation)	0.003393	54.207	8.886	44.934

Table 1: Mean absolute error for the diabetes dataset using the estimators: MAE+Lasso with **standard cross validation** and MAE+Lasso with **robust cross validation**

3. On the other hand, it can be seen that using robust cross validation, the MAE (validation set) improved significantly and the MAE (train set) got worse. This behaviour was expected since we picked to train the model with the portion of the data set that would get the worst possible MAE.

Question 8

The selected dataset represents data collected in two different schools regarding the student's performance in studying, student's alcohol consumption and number of background factors related to family, social background, personal characteristics of the student and motivation in studying. The proposed task is to predict the grade of the student in mathematics based on the background factors. At the following web page, one can access the full dataset and check all variables: `https://www.kaggle.com/uciml/student-alcohol-consumption`

In order to assess the benefits of Lasso and robust cross validation we trained three models with the student dataset and presented the results in Table 2. Although, we believe that the worst case estimator (robust cross validation) might be too conservative, the MAE (test set) found is in this case very close to the one found using standard cross validation. This suggest that fitting a model by minimizing the MAE (which handles better the outliers than the MSE) and choosing the worst possible train dataset should provide a model close to the best possible estimator.

Estimator	λ	MAE Train set	MAE Validation set	MAE Test set
MAE		0.130	0.163	0.183
MAE+Lasso (standard cross validation)	0.0016	0.131	0.157	0.178
MAE+Lasso (robust cross validation)	0.12	0.202	0.019	0.177

Table 2: Mean absolute error for the alcohol dataset using the estimators: MAE, MAE+Lasso with **standard cross validation** and MAE+Lasso with **robust cross validation**

.

Question 9

A presentation is prepared to explain our results.