Lecture 5:

Model selection
Gradient descent
The LASSO
The LASSO and gradient descent

Lecture 5

Section 1: Model selection

Main question of the section is: how to select an optimal value of α. The same type of problem arises in polynomial regression as well: how do we select an optimal degree d of a polynomial?

Idea: choose such a value of a hyperparameter that minimizes the error (cost).

Let $\mathcal{D}$ be a probability distribution on $\mathbb{R}^d \times \mathbb{R}^m$ and the input data is built from independent copies $(\bar{x}^{(i)}, \bar{y}^{(i)})$ chosen from the distribution $\mathcal{D}$.

$$E(f) := \mathbb{E}_{\bar{x},\bar{y}}\left[\ell(\bar{y}, f(\bar{x}))\right] = \int \rho(\bar{x},\bar{y})\, \ell(\bar{y}, f(\bar{x}))\, d\bar{x}\, d\bar{y}$$

$\rho(\bar{x},\bar{y})$: ?

$E(f)$: Population risk, Expected risk, Expected error

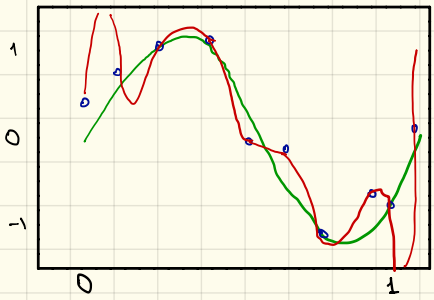
ℓ: cost function (MSE, MAE, etc.)

$$L_S(f) := \frac{1}{|S|} \sum_{(\bar{x},\bar{y}) \in S} \ell(\bar{y}, f(\bar{x})) \xrightarrow{|S| \to \infty} E(f)$$

$L_S(f)$: empirical risk

Main problem: f is a function of a training set and thus training error (the one computed over a training set) is not representative.

Example:



The error is zero in this case, but the regression is not optimal.

Thus we need to validate/check the model on a different data.

Idea: split the training data into proper training data and a validation data. Usually 80/20.

$$S = S_t \cup S_v$$

S_t: training, S_v: validation

$$f = f_{w_t}, \quad w_t = \arg\min_w L_t(f_w) \quad \text{where}$$

$$L_t(f) := \frac{1}{|S_t|} \sum_{x,y \in S_t} \ell(y, f(x))$$

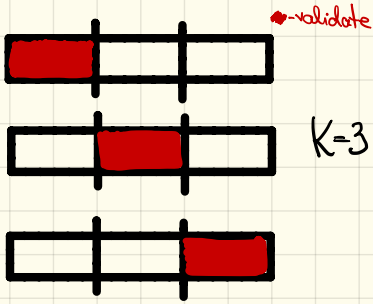
$$\text{Error}: \quad L_v(f) := \frac{1}{|S_v|} \sum_{x,y \in S_v} \ell(y, f(x)) \to \min$$

hyperparameters

K-fold cross-validation

Another possible validation idea: repeat the above several times in such a way that all data is used for both: training and validation but at different times.

- Randomly partition data into K groups
- Train the model K times, leaving 1 group for validation
- Average the result



1. Split the data S into training data S_t and a validation data S_v.
2. Train the model.
3. Calculate the validation error.
4. Minimise the validation error over all possible values of hyperparameters.

Example 1: Ridge regression - selection of a parameter α.

$$W_\alpha = \arg\min_{W \in \mathbb{R}^{(d+1)\times m}} \frac{1}{2} \| \underset{t\times(d+1)}{X_t} W - \underset{t\times m}{Y_t} \|^2 + \frac{\alpha}{2} \|W\|^2$$

$$\hat{\alpha} = \arg\min_{\alpha} \frac{1}{2} \| \underset{v\times(d+1)}{X_v} W^\alpha - \underset{v\times m}{Y_v} \|^2$$

Example 2: Polynomial regression - selection of a parameter d.

$$W_d = \arg\min_{W \in \mathbb{R}^{(d+1)\times m}} \frac{1}{2} \| \Phi_d(X_t) W - Y_t \|^2$$

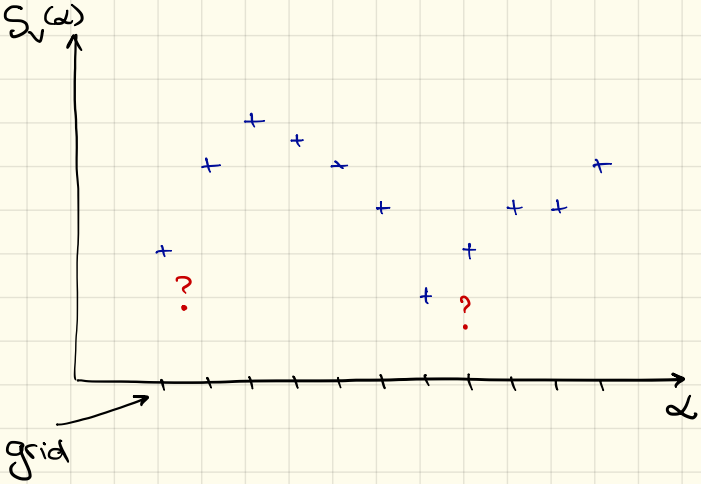
$$\hat{d} = \arg\min_{d} \frac{1}{2} \| \Phi_d(X_v) W_d - Y_v \|^2$$

? How can we solve such a bi-level optimisation problem?

1. Analytically it my be too complicated.
2. Grid search (evaluate the function of the interest - validation error - at the points of a grid).
3. Random search (evaluate the function of our interest at a random points).
4. Gradient-based optimisation (iterative procedure to approach the minimum)
5. Bayesian optimisation
6. Evolutionary optimisation

In this module we are mainly using grid search.

The method consist of:
- defining a grid of values of hyper parameter;
- evaluating the validation error for every value of a hyper parameter of the grid;
- finding such a $\hat{\lambda}$ that minimises the error.



- Where the minimum is?
- Whether it is attained at one of the grid points?
- Can we be sure that this is a global minimum and not the local one?

Advantages:
1. Universality
2. Easy realisation

Disadvantages
1. Computationally infeasible
2. No guarantee of getting the minimum point.

Section 2: The LASSO

Previously we have seen how the ridge regression can help to prevent overfitting.

$$W_\alpha = \arg\min_W \{ L(W) + R(W) \}$$

($L(W)$: regression; $R(W)$: regularisation)

In this section we discuss another regularisation of a standard regression problem. Namely we discuss the LASSO - Least Absolute Shrinkage and Selection Operator.

Def. The LASSO problem is a minimisation problem of the form

$$W_\alpha = \arg\min_W \frac{1}{2}\|\Phi(X)W - Y\|^2 + \alpha\|W\|_1$$

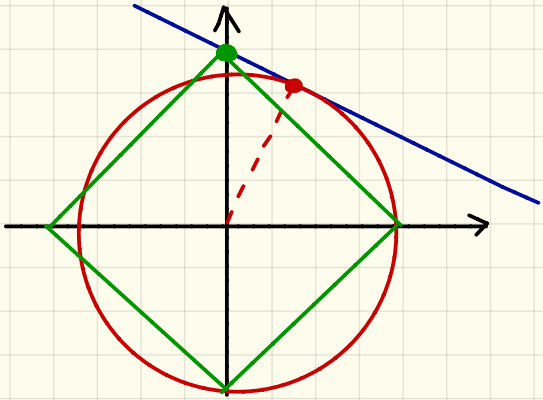
where $\|W\|_1 = \sum_{i,j} |W_j^{(i)}|$

What is the motivation of introducing 1-norm?

$\|v\|_2^2 = \sum_{i=1}^{N} v_i^2 \rightarrow$ fix $\Rightarrow \|v\|_1 \rightarrow$ min when v is sparse

Example:

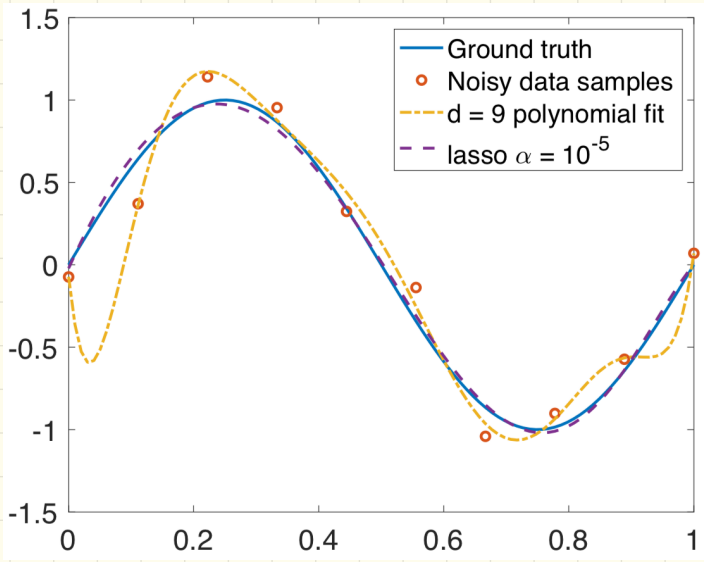
Consider a linear regression with 1 data point: (x,y).



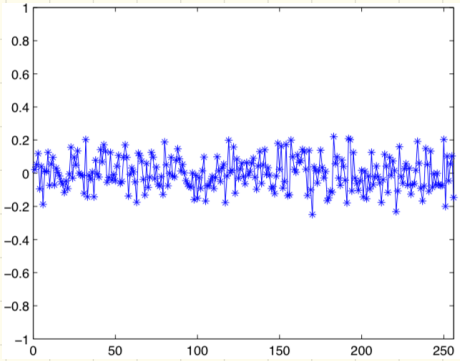
1°) $\min(w_0^2 + w_1^2)$ subject to $y = w_0 + w_1 x$

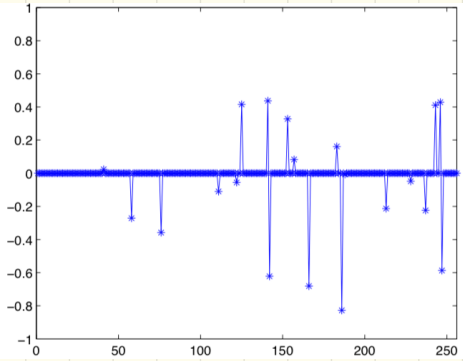
2°) $\min(|w_0| + |w_1|)$ subject to $y = w_0 + w_1 x$

Example:



Example:





$\|\cdot\|_1 = 20.061$ $\quad\quad$ $\|\cdot\|_1 = 6.293$

$\|\cdot\|_2 = 1.5431$ $\quad\quad$ $\|\cdot\|_2 = 1.747$

Therefore, the LASSO problem reconstructs solutions with only few non-zero elements. This leads to implicit parameters reduction and thus we obtain the simpler model (occam's razor).

Problem: 1-norm is not differentiable.

We need another method to solve the problem!

Section 3: Gradient descent

Let us consider a problem of minimisation

$$\hat{w} = \arg\min_{w} E(w)$$

Def. The gradient descent method is an iterative procedure of the form

$$w^{k+1} = w^k - \tau \nabla E(w^k)$$

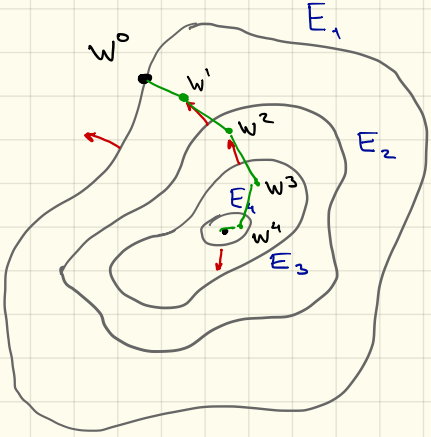
for the energy E, an initial value w^0 and a step-size parameter τ.

$$w^1 = w^0 - \tau \nabla E(w^0)$$

$$w^2 = w^1 - \tau \nabla E(w^1)$$

$$w^3 = w^2 - \tau \nabla E(w^2)$$

$$\vdots$$



$E_1 > E_2 > E_3 > E_4$

Example Zero order polynomial regression

$$E(w) = \frac{1}{2S} \sum_{i=1}^{S} (w - y_i)^2 \Rightarrow \nabla E(w) = w - \frac{1}{S} \sum_{i=1}^{N} y_i$$

$$w^{k+1} = (1-\tau) w^k + \frac{\tau}{S} \sum_{i=1}^{S} y_i$$

what happens for $\tau = 1$?

$$\left(w^{k+1} - \frac{1}{S} \sum_{i=1}^{S} y_i \right) = (1-\tau) \left(w^k - \frac{1}{S} \sum_{i=1}^{S} y_i \right)$$

$0 < \tau < 2 \Rightarrow w^k \to \frac{1}{S} \sum_{i=1}^{S} y_i$

Example Linear regression

$$E(w) = \frac{1}{2s} \|Xw - y\|^2$$

$$\nabla E(w) = \frac{1}{s} X^T (Xw - y)$$

$$w^{k+1} = w^k - \frac{\tau}{s} X^T (X w^k - y)$$

$$= \left(1 - \frac{\tau}{s} X^T X\right) w^k + \frac{\tau}{s} X^T y$$

How should we select the parameter τ to guarantee the convergence?

Idea: we should guarantee that the energy decrease!

$$E(w^{k+1}) = \frac{1}{2s} \|Xw^k - \frac{\tau}{s} XX^T X w^k + \frac{\tau}{s} XX^T y - y\|^2$$

$$= \frac{1}{2s} \|\left(I - \frac{\tau}{s} XX^T\right) \cdot (Xw^k - y)\|^2$$

$$\leq \frac{1}{2s} \|I - \frac{\tau}{s} XX^T\|^2 \cdot \|Xw^k - y\|^2$$

$$= \|I - \frac{\tau}{s} XX^T\|^2 \cdot E(w^k)$$

< 1 for small τ

How to guarantee a convergence in general case?

Theorem: Let $E: C \subset \mathbb{R}^m \to \mathbb{R}$ be a convex energy function such that $G(w) := \frac{1}{2\tau}\|w\|^2 - E(w)$ is also convex. Let $\hat{w}$ be a global minimiser of E. Then:

1. For any k: $E(w^{k+1}) \leq E(w^k)$
2. There exist $C > 0$ such that $E(w^k) - E(\hat{w}) \leq C/k$
3. As a consequence $E(w^k) \to E(\hat{w})$

Def.: A continuously differentiable function $E: C \subset \mathbb{R}^m \to \mathbb{R}$ is called L-smooth if

$$\|\nabla E(u) - \nabla E(v)\| \leq L\|u - v\|, \quad \forall u, v \in C.$$

Lemma. Let $E: C \subset \mathbb{R}^m \to \mathbb{R}$ be an γ_τ-smooth function. Then

$$G_\tau(w) = \frac{1}{2\tau}\|w\|^2 - E(w)$$

is convex on C.

Example

$$E(w) = \frac{1}{2s}\sum_{j=1}^{s}(w - y_j)^2$$

$$\nabla E(w) = w - \frac{1}{s}\sum_{j=1}^{s} y_j$$

$$\|\nabla E(u) - \nabla E(v)\| = \|u - v\|$$

$$\Rightarrow E \text{ is } 1\text{-smooth } (\tau = 1)$$

What if E is not convex?

Then the gradient descent would end up at some local minima.

What if E is convex but there is no such a τ that G_τ is convex?

Then we need to adapt our algorithm. When defining w^{k+1} one needs to check $E(w^{k+1}) \le E(w^k)$. If not, then decrease the value of τ.

Section 4: The LASSO and gradient descent

How to use a gradient descent to solve the LASSO?

The main difficulty to overcome is non-differentiability of the absolute value function. The trick is to replace the absolute value with a differentiable function.

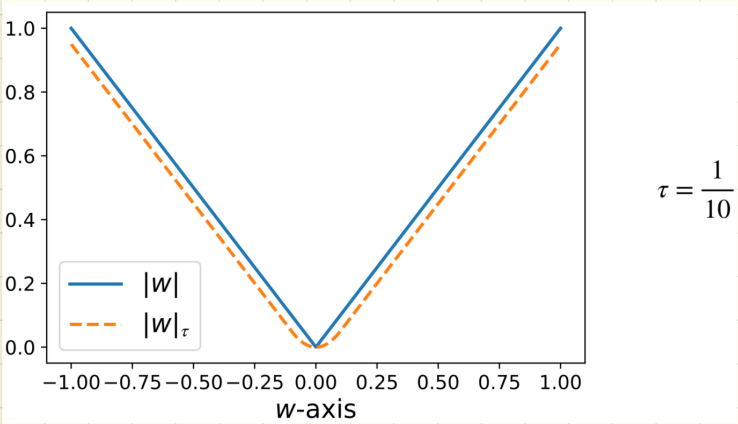
$$|x| = \max_{p \in [-1,1]} xp$$

$$|x|_\tau := \max_{p \in [-1,1]} xp - \frac{\tau}{2} p^2, \quad \tau > 0$$

↑ small perturbation

$|x|_\tau$ - ?

- $|x| \le \tau \Rightarrow |x|_\tau = \frac{x^2}{2\tau}$
- $|x| > \tau \Rightarrow |x|_\tau = |x| - \frac{\tau}{2}$



$\tau = \frac{1}{10}$

Properties of the smoothing:

1. The smooth absolute value is differentiable.

$$\frac{d}{dw}|w|_\tau = \begin{cases} 1, & w > \tau \\ \frac{1}{\tau} w, & |w| \le \tau \\ -1, & w < -\tau \end{cases}$$

2. The smooth absolute value is close to abs.

$$|w_\tau| \le |w| \le |w_\tau| + \frac{\tau}{2}.$$

$$w_\alpha^\tau = \arg\min_w \frac{1}{2}\|\Phi(X)w - y\|^2 + \alpha H_\tau(w)$$

$$H_\tau(w) := \sum_{i=0}^{d} |w_i|_\tau .$$

Can we use normal equation? - No, the system is not linear anymore.
Can we use gradient descent? - Yes, with minor upgrade.

$$E_\tau(w) = \frac{1}{2}\|Xw - y\|^2 + \alpha H_\tau(w)$$

$$w^{k+1} = w^k - \tau\left[X^T(Xw^k - y) + \alpha \nabla H_\tau(w^k)\right]$$

! in practice

forward-splitting

$$w^{k+1/2} = w^k - \frac{\tau}{\alpha} X^T(Xw^k - y)$$

$$w^k = w^{k+1/2} - \tau \nabla H_\tau(w^{k+1/2})$$

Remark: $w - \tau \nabla |w_\tau| = \begin{cases} w - \tau, & w \ge \tau \\ 0 \\ w + \tau, & w \le -\tau \end{cases} =: \text{soft}_\tau(w)$

soft-tresholding

$$w_j^{k+1} = \text{soft}_\tau\left[\left(w^k - \frac{\tau}{\alpha} X^T(Xw^k - y)\right)_j\right]$$

This is known as ISTA (iterative soft-tresholding algorithm) and is a special case of proximal gradient descent

$$w^{k+1} = (1 + \tau \partial R)^{-1}\left(w^k - \tau \nabla L(w^k)\right)$$

proximal map → $(1+\tau\partial R)^{-1}(z) := \arg\min_w \left\{\frac{1}{2}\|w-z\|^2 + \tau R(w)\right\}$

$$L = \frac{1}{2d}\|Xw - y\|^2$$

$$R = \tau |x|$$

Section 5 Proximal gradient descent

Recap: The LASSO problem consist of

$$w_\alpha = \arg\min_w \left\{ \frac{1}{2}\|Xw - y\|^2 + \alpha H_\tau(w) \right\}$$

where $H_\tau(w) = \sum_{j=0}^{d} |w_j|_\tau$, and $|x|_\tau = \begin{cases} |x| - \frac{\tau}{2}, & |x| > \tau \\ \frac{1}{2\tau}|x|^2, & |x| \le \tau \end{cases}$

Gradient descent in this case has a form

$$w^{k+1} = w^k - \tau \left[X^\top (Xw^k - y) + \alpha \nabla H_\tau(w^k) \right]$$

Alternative: proximal gradient descent

Suppose we want to minimise $E(w) = L(w) + R(w)$

where

1. L is convex, continuously differentiable
2. R is proper, convex, lower-semi continuous

and has a simple proximal map, i.e.

proximal map

$$\text{prox}_{\tau R}(z) := \arg\min_x \left[\frac{1}{2}\|x - z\|^2 + \tau R(x) \right]$$

↑ distance to z

is easy to compute

Def. The proximal gradient descent method is an iterative procedure of the form

$$w^{k+1} = \text{prox}_{\tau R}\left(w^{(k)} - \tau \nabla L(w^{(k)}) \right)$$

for the energy $E(w) = L(w) + R(w)$

If additionally to the above R is differentiable

$$\nabla \left(\frac{1}{2}\|x-z\|^2 + \tau R(x) \right) = x - z + \tau \nabla R(x) = 0$$

$$w^{k+1} + \tau \nabla R(w^{k+1}) - \left(w^k - \tau \nabla L(w^k) \right) = 0$$

$$\boxed{w^{k+1} = w^k - \tau \left(\nabla L(w^k) + \nabla R(w^{k+1}) \right)}$$

↑ implicit - explicit gradient method

Example: $R(w) = \frac{1}{2}\|w\|^2$

$$\text{prox}_{\tau R}(z) = \arg\min_x \left\{ \underbrace{\frac{1}{2}\|x-z\|^2 + \frac{\tau}{2}\|x\|^2}_{E(x)} \right\}$$

$$\nabla E(x) = x - z + \tau x = 0$$

$$\boxed{\hat{x} = \frac{1}{1+\tau} \cdot z}$$

$$\left(1 + \tau \, \partial \, \tfrac{1}{2}\|\cdot\|^2\right)^{-1}(z) = \frac{z}{1+\tau}$$

Example:

$$R(w) = \begin{cases} 0, & w \in C \\ \infty, & w \notin C \end{cases}$$

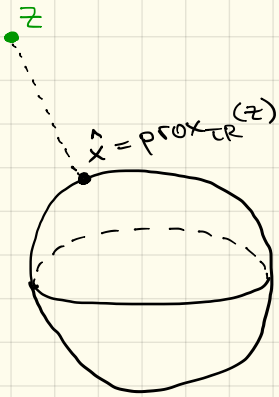
↑ convex set

Remark: This corresponds to a minimisation problem with constraint $w \in C$.

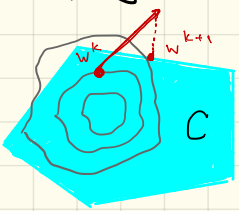
$$\text{prox}_{\tau R}(z) = \arg\min_x \left[\frac{1}{2}\|x-z\|^2 + \tau R(x) \right]$$

$$= \arg\min_{x \in C} \frac{1}{2}\|x-z\|^2$$

↑ projection onto convex set



$$w^{k+1} = \text{proj}_C\left(w^k - \tau \nabla L(w^k)\right)$$



Example: $C = \{x \in \mathbb{R} : x \geq 0\}$

$\text{proj}_C(z) = \max(0, z)$ [Exercise]

$w^{k+1} = \max(0,\ w^k - \tau \nabla L(w^k))$

↑ ensures that the iterate stays positive.

What is the motivation of proximal gradient descent?

$$w^{k+1} = \arg\min_w \{ L(w) + R(w) + D_J(w, w^k) \}$$

where $J = \frac{1}{2\tau}\|w\|^2 - L(w)$

$$D_J(u, v) = J(u) - J(v) - \langle \nabla J(v), u - v \rangle$$

↑ analogue of $\frac{1}{2} J''(v) \cdot (u-v)^2$ in $d = 1$

$$w^{k+1} = \arg\min_w \{ L(w) + R(w) + \tfrac{1}{2\tau} D_{\|\cdot\|^2}(w, w^k) - D_L(w, w^k) \}$$

$$= \arg\min_w \{ L(w^k) - \langle \nabla L(w^k), w - w^k \rangle + R(w) + \tfrac{1}{2\tau}(\|w\|^2 - \|w^k\|^2 - 2\langle w^k, w - w^k \rangle) \}$$

$$= \arg\min_w \{ \tau R(w) + \tfrac{1}{2}\|w - w^k\|^2 - \tau \langle \nabla L(w^k), w - w^k \rangle \}$$

$$= \arg\min_w \{ \tfrac{1}{2}\|w - (w^k - \tau \nabla L(w^k))\|^2 + \tau R(w) \}$$

How do we use proximal gradient descent to solve the LASSO problem?

$$\hat{w}_\alpha = \arg\min_w \left\{ \underbrace{\frac{1}{2}\|Xw - y\|^2}_{L(w)} + \underbrace{\alpha \|w\|_1}_{R(w)} \right\}$$

$\text{prox}_{\tau R}(z) = ?$

$$\left[\left(I + \tau \alpha \partial \|\cdot\|_1 \right)^{-1} (z) \right]_j = \begin{cases} z_j - \tau\alpha, & z_j > \tau\alpha \\ 0, & |z_j| \leq \tau\alpha \\ z_j + \tau\alpha, & z_j < -\tau\alpha \end{cases}$$

(Assignment)

$$w_\alpha^{k+1} = \left(I + \tau \alpha \partial \|\cdot\|_1 \right)^{-1} \left[\left(I - \tau X^T X \right) w^k + \tau X^T y \right]$$