IFT6390-fundamentals of machine learning
Assignment 2

Jonathan Guymont, Marzieh Mehdizadeh

1 Linear and non-linear regularized regression

1.1 Linear Regression

(1) The set of parameters is $\theta = \{w_1, ..., w_d, b\}$ where $w_i \in \mathbb{R}$ for $i = 1, ..., d$ and $b \in \mathbb{R}$. The w_i are the coefficients of the different features of the inputs random variable $x_1, ..., x_d$ and b is the bias.

(2)

$$\begin{aligned}\hat{R}((x,t),f) =& \sum_{i=1}^{n}(f(\boldsymbol{x}^{(i)}) - t^{(i)})^2 \\ =& \sum_{i=1}^{n}(b + \boldsymbol{w}^\top \boldsymbol{x}^{(i)} - t^{(i)})^2\end{aligned} \tag{1}$$

(3) We rewrite the loss with matrix instead. Let $X \in \mathbb{R}^{n \times d+1}$ be the design matrix where the rows are the different training examples and the first column is a row of ones that we add for convenience. Also lets redefine the weight vector as $\boldsymbol{w}^t op = (b, w_1, ..., w_d)$. In this new formulation of the input and parameters, we have that $\boldsymbol{w}^\top \boldsymbol{x} \equiv b + \boldsymbol{w'}^t op \boldsymbol{x'}$, where "$'$" indicates the previous format. Let $\hat{Y} \in \mathbb{R}^n$ be the vector of preditions for all the training examples and $Y = (t^{(1)}, ..., t^{(n)})^\top$. We have $\hat{Y} \in \mathbb{R}^n = X\boldsymbol{w}$. It follows that $\hat{R}((x,t),f) = (\hat{Y} - Y)^\top(\hat{Y} - Y) = (X\boldsymbol{w} - Y)^\top(X\boldsymbol{w} - Y)^\top$

$$\begin{aligned}(X\boldsymbol{w} - Y)(X\boldsymbol{w} - Y)^\top =& (-Y^\top + \boldsymbol{w}^\top X^\top)(X\boldsymbol{w} - Y) \\ =& \boldsymbol{w}^\top X^\top X\boldsymbol{w} - Y^\top X\boldsymbol{w} - \boldsymbol{w}^\top X^\top Y + Y^\top Y \\ =& \boldsymbol{w}^\top X^\top X\boldsymbol{w} - 2\boldsymbol{w}^\top X^\top Y + Y^\top Y \quad (Y^\top X\boldsymbol{w} = \boldsymbol{w}^\top X^\top Y)\end{aligned} \tag{2}$$

where the last equality is true because $Y^\top X\boldsymbol{w}$ and $\boldsymbol{w}^\top X^\top Y$ are both the same scalar. And the derivative wrt w is given by

$$\frac{\partial}{\partial \boldsymbol{w}} \boldsymbol{w}^\top X^\top X\boldsymbol{w} - 2\boldsymbol{w}^\top X^\top Y + Y^\top Y = 2X^\top X\boldsymbol{w} - 2X^\top Y \tag{3}$$

Setting the derivatives to zero

$$\begin{aligned}2X^\top X\boldsymbol{w} - 2X^\top Y =& 0 \\ X^\top X\boldsymbol{w} =& X^\top Y \\ \boldsymbol{w}* =& (X^\top X)^{-1}X^\top Y\end{aligned} \tag{4}$$

(4) While $||w|| > \epsilon$ Do:

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} - \lambda \cdot 2(X^\top X \boldsymbol{w}_{t-1} - X^\top Y)$$

where ϵ is some level of tolerence beyond which we are satisfied with the solution (gradient is flat enough).

(5) The gradient can be rewriten as $2 \cdot (X^\top \hat{Y} - X^\top Y) = 2X^\top \cdot (\hat{Y} - Y) = 2X^\top \boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon}$ is the vector of errors (one for each examples). For a single point, the gradient is equal to $2\boldsymbol{x}^{(i)}\varepsilon^{(i)}$. We can see that the higher the error, the bigger the step. Also we always move in either the direction of $\boldsymbol{x}$ or in the opposite direction, and proportionnaly to the magnitude of the error.

1.2 Ridge Regression

(1) The gradient of the regulaized risk is equal to the gradient of the unregularized rist plus the gradient of the regularizer.

$$\frac{\partial}{\partial \boldsymbol{w}} \hat{R} = 2(X^\top X \boldsymbol{w}_{t-1} - X^\top Y) + 2\lambda \boldsymbol{w}$$

(2) While $||w|| > \epsilon$ Do:

$$\boldsymbol{w}_t \leftarrow \boldsymbol{w}_{t-1} - \eta \cdot 2(X^\top X \boldsymbol{w}_{t-1} - X^\top Y + \lambda \boldsymbol{w}_{t-1})$$

where ϵ is some level of tolerance beyond which we are satisfied with the solution (the gradient is flat enough).

(3) The gradient in (1) is already written in term of matrices. The regularized risk is given by

$$\hat{R} = (X\boldsymbol{w} - Y)(X\boldsymbol{w} - \boldsymbol{t})^\top + \lambda ||\boldsymbol{w}||^2$$

(4) Setting the derivative in (1) to zero

$$\begin{aligned} 2(X^\top X \boldsymbol{w} - X^\top Y) + 2\lambda \boldsymbol{w} &= 0 \\ X^\top X \boldsymbol{w} + \lambda \boldsymbol{w} &= X^\top Y \\ (X^\top X + \lambda I)\boldsymbol{w} &= X^\top Y \\ \boldsymbol{w}* &= (X^\top X + \lambda I)^{-1} X^\top Y \end{aligned} \tag{5}$$

If $\lambda = 0$ whatever N is, the empirical risk will equal to the unregularized one. The number of examples N do not affect the shape of the gradient (the gradient is always in $\mathbb{R}^d$ whatever N is).

1.3 Regression with a fixed non-linear pre-processing

(1) $\tilde{f}_k(x) = b + \sum_{i=1}^{k} w_i x^i = b + \boldsymbol{w}^\top \phi_k(x)$

(2) b is a scalar and $w_1, ..., w_k$ are also scalars and the coefficient of the polynomial terms. Therefore dimension $d = 1$ we will have $k + 1$ parameters

(3) $\phi_1(x) = x$ so this one is a linear regression.

$\tilde{f}_1(\boldsymbol{x}) = b + w_1 x_1 + w_2 x_2$

$\tilde{f}_2(\boldsymbol{x}) = b + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2 + w_5 x_1 x_2$

$\tilde{f}_3(\boldsymbol{x}) = b + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2 + w_5 x_1 x_2 + w_6 x_1^2 x_2 + w_7 x_1 x_2^2 + w_8 x_1^3 + w_9 x_2^3$
(4) For each powers $1, 2, ..., k$ we need d coefficients, i.e. dk parameters. We also need coefficient for the different combination of inputs (eg $x_1 x_2$). IF p is the degree of polynomial up to k, then one can prove the complexity by induction which it equal to:

$$\sum_{p=1}^{k} \binom{d-1+p}{p}.$$

2 Practical Part

Comment question 6: Figure 3 show the predictions of 3 polynomial regression of different degrees fitted using gradient descent. We can see that models with higher degrees seems to fit the training observations better. We can also see that it is quite the opposite when we compare the models to the ground truth $h(x)$. Table 1 show the mean square errors of the fitted models computed on both the training set and the validation set. We can see that while the loss on the training set is lower for higher degree, it is the opposite for looses computed on the test set. This was expected since the ground truth is a polynomial of degree one. Thus the models with $l > 1$ have to much capacity, which is particularly problematic given the very small training set, since in order to not overfit, the models would have to learn that $w_k = 0$ for $k > 1$.

l	train	test
1	0.5654	0.5783
2	0.5549	0.6295
5	0.2722	0.771

Table 1: Mean square error of the fitted polynomial regressions of degree l. The MSE it reported for both the training and the test set

Figure 1: Fitted ridge regression with different regularization constant

Figure 2: Effect of the regularization constant λ on the MSE loss of a simple linear regression

Figure 3: Fitted ridge regression with polynomial degrees