Machine Learning, 2024 Spring
Assignment 6

Name: Zhou Shouchen

Student ID: 2021533042

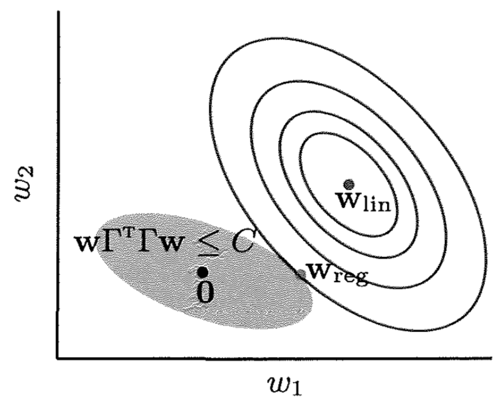
Notice

Plagiarizer will get 0 points.
LaTeXis highly recommended. Otherwise you should write as legibly as possible.

Problem 1 In this problem, you will investigate the relationship between the soft order constraint and the augmented error. The regularized weight $\mathbf{w}_{\text{reg}}$ is a solution to

$$\begin{gathered}\min E_{\text{in}}(\mathbf{w}) \\ \text{subject to } \mathbf{w}^{\mathrm{T}}\Gamma^{\mathrm{T}}\Gamma\mathbf{w} \leq C\end{gathered}$$

(a) If $\mathbf{w}_{\text{lin}}^{\mathrm{T}}\Gamma^{\mathrm{T}}\Gamma \mathrm{w}_{\text{lin}} \leq C$, then what is $\mathbf{w}_{\text{reg}}$?
(b) If $\mathbf{w}_{\text{lin}}^{\tau}\Gamma^{\top}\Gamma\mathbf{w}_{\text{lin}} > C$, the situation is illustrated below,



The constraint is satisfied in the shaded region and the contours of constant E_{in} are the ellipsoids (why ellipsoids?). What is $\mathbf{w}_{\text{reg}}^{\mathrm{T}}\Gamma^{\mathrm{T}}\Gamma\mathbf{w}_{\text{reg}}$?
(c) Show that with

$$\lambda_C = -\frac{1}{2C}\mathbf{w}_{\text{reg}}^{\mathrm{T}}\nabla E_{\text{in}}(\mathbf{w}_{\text{reg}})$$

$\mathbf{w}_{\text{reg}}$ minimizes $E_{\text{in}}(\mathbf{w}) + \lambda_C\mathbf{w}^{\mathrm{T}}\Gamma^{\mathrm{T}}\Gamma\mathbf{w}$. [Hint: use the previous part to solve for $\mathbf{w}_{\text{reg}}$ as an equality constrained optimization problem using the method of Lagrange multipliers.]
(d) Show that the following hold for λ_C :
(i) If $\mathrm{w}_{\text{lin}}^{\mathrm{T}}\Gamma^{T}\Gamma\mathbf{w}_{\text{lin}} \leq C$ then $\lambda_C = 0$ ($\mathbf{w}_{\text{lin}}$ itself satisfies the constraint).
(ii) If $\mathrm{w}_{\text{lin}}^{\mathrm{T}}\Gamma^{\mathrm{T}}\Gamma\mathrm{w}_{\text{lin}} > C$, then $\lambda_C > 0$ (the penalty term is positive).
(iii) If $\mathrm{w}_{\text{lin}}^{\mathrm{T}}\Gamma^{\mathrm{T}}\Gamma\mathrm{w}_{\text{lin}} > C$, then λ_C is a strictly decreasing function of C. [Hint: show that $\frac{d\lambda_C}{dC} < 0$ for $C \in \left[0, \mathbf{w}_{\text{lin}}^{T}\Gamma^{T}\Gamma\mathbf{w}_{\text{lin}}\right]\right]$.

Solution
(a) Since $\mathbf{w}_{\text{lin}}^{\mathrm{T}}\Gamma^{\mathrm{T}}\Gamma\mathrm{w}_{\text{lin}} \leq C$, which has already suitable for the constrains, so $\mathbf{w}_{\text{reg}} = \mathbf{w}_{\text{lin}}$.
(b) 1. Firstly, we can prove that the contours of constant E_{in} are the ellipsoids.
We can set $E_{\text{in}}(\mathbf{w})$ to be the L_2 loss, i.e.

$$E_{\text{in}}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n - y_n\right)^2 = \frac{1}{N}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2$$

where $\mathbf{X}$ is the matrix of feature vectors, $\mathbf{y}$ is the vector of labels, and N is the number of data points, $\mathbf{x}_n$ is the feature vector of the n-th data point and y_n is the label.

Consider the $\mathbf{w}$ in $2D$ dimensional case, where each feature vector has a dummy feature $x_2 = 1$, so $\mathbf{w} = [w_1, w_2]$, and $\mathbf{x}_n = [x_n, 1]$. Then the $E_{\text{in}}(\mathbf{w})$ can be written as

$$
\begin{aligned}
E_{\text{in}}(\mathbf{w}) &= \frac{1}{N}\|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 \\
&= \frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n - y_n\right)^2 \\
&= \frac{1}{N}\sum_{n=1}^{N}\left(w_1 x_n + w_2 - y_n\right)^2 \\
&= \frac{1}{N}\sum_{n=1}^{N}(w_1^2 x_n^2 + w_2^2 + 2w_1 w_2 x_n + C(w_1, w_2, x_n, y_n))
\end{aligned}
$$

Since we want to get the contours of constant E_{in} , so we just need to focus on the quadratic form of E_{in} , so $C(w_1, w_2, x_n, y_n)$ can be ignored.
Let $A = \frac{1}{N}\sum_{n=1}^{N} x_n^2$, $B = \frac{1}{N}\sum_{n=1}^{N} x_n$,$C = 1$.
So

$$
E_{\text{in}}(\mathbf{w}) = A \cdot w_1^2 + 2B \cdot w_1 w_2 + C \cdot w_2^2 + \frac{1}{N}\sum_{n=1}^{N} C(w_1, w_2, x_n, y_n)
$$

$$
\begin{aligned}
AC - B^2 &= \frac{1}{N}\sum_{n=1}^{N} x_n^2 \cdot 1 - \left(\frac{1}{N}\sum_{n=1}^{N} x_n\right)^2 \\
&= \frac{1}{N}\sum_{n=1}^{N} x_n^2 - \frac{1}{N^2}\left(\sum_{n=1}^{N} x_n\right)^2 \\
&= \frac{1}{N^2}\left[N\sum_{n=1}^{N} x_n^2 - \left(\sum_{n=1}^{N} x_n\right)^2\right] \\
&= \frac{1}{N^2}\left[(N-1)\sum_{n=1}^{N} x_n^2 - \sum_{1 \le i < j \le N} x_i x_j\right] \\
&= \frac{1}{N^2}\left[\sum_{1 \le i < j \le N} (x_i - x_j)^2\right] \\
&\ge 0
\end{aligned}
$$

Since we can remove the duplicate data, i.e. make sure $x_i \neq x_j$, so $AC - B^2 > 0$, so the contours of constant E_{in} are the ellipsoids.

2. Secondly, from the figure of contours, we can find that the minimum of E_{in} which also fit the constraint is on the boundary of the constraint, i.e. the intersection of the objective function's contour and the constraint.
So we can find that

$$
\mathbf{w}_{\text{reg}}^{\mathrm{T}}\Gamma^{\mathrm{T}}\Gamma\mathbf{w}_{\text{reg}} = C
$$

So above all, the contours of constant E_{in} are the ellipsoids, and $\mathbf{w}_{\text{reg}}^{\mathrm{T}}\Gamma^{\mathrm{T}}\Gamma\mathbf{w}_{\text{reg}} = C$.

(c) We can apply Lagrange multipliers to solve the problem.
The Lagrangian function is:

$$
L(\mathbf{w}, \lambda) = E_{\text{in}}(\mathbf{w}) + \lambda(\mathbf{w}^{\mathrm{T}}\Gamma^{\mathrm{T}}\Gamma\mathbf{w} - C)
$$

And its gradient is:

$$\nabla_{\mathbf{w}} L(\mathbf{w}, \lambda) = \nabla E_{\text{in}}(\mathbf{w}) + 2\lambda \Gamma^{\mathrm{T}} \Gamma \mathbf{w}$$

To minimize $E_{\text{in}}(\mathbf{w}) + \lambda \mathbf{w}^{\mathrm{T}} \Gamma^{\mathrm{T}} \Gamma \mathbf{w}$, we need to make sure that the gradient of $L(\mathbf{w}, \lambda)$ is zero, so we have

$$\begin{aligned}
\nabla_{\mathbf{w}} L(\mathbf{w}, \lambda) = 0 &\Rightarrow \nabla E_{\text{in}}(\mathbf{w}) + 2\lambda \Gamma^{\mathrm{T}} \Gamma \mathbf{w} = 0 \\
&\Rightarrow 2\lambda \Gamma^{\mathrm{T}} \Gamma \mathbf{w} = -\nabla E_{\text{in}}(\mathbf{w}) \\
&\Rightarrow 2\lambda \mathbf{w}^{\mathrm{T}} \Gamma^{\mathrm{T}} \Gamma \mathbf{w} = -\mathbf{w}^{\mathrm{T}} \nabla E_{\text{in}}(\mathbf{w}) \\
&\Rightarrow \lambda = -\frac{1}{2\mathbf{w}^{\mathrm{T}} \Gamma^{\mathrm{T}} \Gamma \mathbf{w}} \mathbf{w}^{\mathrm{T}} \nabla E_{\text{in}}(\mathbf{w})
\end{aligned}$$

1. If $\mathbf{w}_{\text{reg}}$ is a solution fits the primal feasibility, i.e. $\mathbf{w}_{\text{lin}} \Gamma^{T} \Gamma \mathbf{w}_{\text{lin}} \leq C$
Then we have

$$\mathbf{w}_{\text{reg}} = \mathbf{w}_{\text{lin}}$$

Since its the primal optimal points, so we also have

$$\nabla E_{\text{in}}(\mathbf{w}_{\text{reg}}) = \nabla E_{\text{in}}(\mathbf{w}_{\text{lin}}) = 0$$

Then we also have

$$\lambda = 0$$

2. If $\mathbf{w}_{\text{lin}} \Gamma^{T} \Gamma \mathbf{w}_{\text{lin}} > C$, then we have

$$\mathbf{w}_{\text{reg}} \Gamma^{T} \Gamma \mathbf{w}_{\text{reg}} = C$$

So we have

$$\lambda = -\frac{1}{2C} \mathbf{w}_{\text{reg}}^{T} \nabla E_{\text{in}}(\mathbf{w}_{\text{reg}})$$

So combine the above two situations, we can get that with

$$\lambda = -\frac{1}{2C} \mathbf{w}_{\text{reg}}^{T} \nabla E_{\text{in}}(\mathbf{w}_{\text{reg}})$$

$\mathbf{w}_{\text{reg}}$ minimizes $E_{\text{in}}(\mathbf{w}) + \lambda \mathbf{w}^{\mathrm{T}} \Gamma^{\mathrm{T}} \Gamma \mathbf{w}$.

(d) The KKT conditions of the optimization problem are:

$$\begin{cases}
\mathbf{w}^{T} \Gamma^{T} \Gamma \mathbf{w} \leq C & \text{(primal feasibility)} \\
\lambda \geq 0 & \text{(dual feasibility)} \\
\lambda(\mathbf{w}^{T} \Gamma^{T} \Gamma \mathbf{w} - C) = 0 & \text{(complementary slackness)} \\
\nabla_{\mathbf{w}} L(\mathbf{w}, \lambda) = 0 & \text{(zero gradient of Lagrangian of } \mathbf{w}\text{)}
\end{cases} \tag{1}$$

(i) From the complementary slackness, we have

$$\lambda(\mathbf{w}^{T} \Gamma^{T} \Gamma \mathbf{w} - C) = 0$$

so if $\mathbf{w}_{\text{lin}}^{T} \Gamma^{T} \Gamma \mathbf{w}_{\text{lin}} \leq C$, i.e. $\mathbf{w}_{\text{weg}} = \mathbf{w}_{\text{lin}}$, we have

$$\mathbf{w}_{\text{weg}}^{T} \Gamma^{T} \Gamma \mathbf{w}_{\text{weg}} - C \neq 0$$

then $\lambda = 0$.
From (c)'s 1. , we can also verify that $\lambda_C = 0$.

(ii) From the dual feasibility, we have

$$\lambda \geq 0$$

And if $\mathbf{w}_{\text{lin}}^{T} \Gamma^{T} \Gamma \mathbf{w}_{\text{lin}} > C$, From (c)'s 2. , we can also get that

$$\lambda_C = -\frac{1}{2C} \mathbf{w}_{\text{reg}}^{T} \nabla E_{\text{in}}(\mathbf{w}_{\text{reg}})$$

We can use contradiction to prove that $\lambda > 0$:
Suppose $\lambda_C = 0$, then form KKT condition's zero gradient of Lagrangian of $\mathbf{w}$, we can get that

$$\nabla_{\mathbf{w}} L(\mathbf{w}, \lambda) = \nabla E_{\text{in}}(\mathbf{w}) + 2\lambda \Gamma^{\mathrm{T}} \Gamma \mathbf{w} = 0$$

Since $\lambda_C = 0$, so we can get that

$$\nabla E_{\text{in}}(\mathbf{w}) = 0$$

i.e. $\mathbf{w}_{\text{reg}} = \mathbf{w}_{\text{lin}}$.
However, we have known that $\mathbf{w}_{\text{lin}}^T \Gamma^T \Gamma \mathbf{w}_{\text{lin}} > C$, so its impossible.
So it contradicts.
i.e. it is not possible that $\lambda_C = 0$.

So above all, we have proved that $\lambda_C > 0$.

(iii) From (ii), we have get that

$$\lambda_C = -\frac{1}{2C} \mathbf{w}_{\text{reg}}^T \nabla E_{\text{in}}(\mathbf{w}_{\text{reg}}) > 0$$

When $C \to 0$, we can get that: $C \to 0 \Leftrightarrow \mathbf{w} \to \mathbf{0}$.
Since $C = \mathbf{w}^T \Gamma^T \Gamma \mathbf{w} = o(\|\mathbf{w}\|^2)$, and

$$E_{\text{in}}(\mathbf{w}) = \frac{1}{N} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2$$

$$w^T \nabla E_{\text{in}}(\mathbf{w}) = w^T \frac{2}{N} \mathbf{X}^T (\mathbf{X}\mathbf{w} - \mathbf{y}) = o(\|\mathbf{w}\|^2 + \|\mathbf{w}\|) = o(\|\mathbf{w}\|)$$

So

$$\lim_{C \to 0} \lambda_C = \lim_{\mathbf{w} \to \mathbf{0}} \frac{o(\|\mathbf{w}\|)}{o(\|\mathbf{w}\|^2)} = +\infty$$

And when $C > 0$, we have:

$$\frac{d\lambda_C}{dC} = \frac{1}{2C^2} \mathbf{w}_{\text{reg}}^T \nabla E_{\text{in}}(\mathbf{w}_{\text{reg}}) = -\frac{1}{C} \lambda_C < 0$$

So above all, λ_C is a strictly decreasing function for $C \in \left[0, \mathbf{w}_{\text{lin}}^T \Gamma^T \Gamma \mathbf{w}_{\text{lin}}\right]$.

Problem 2 [The Lasso algorithm] Rather than a soft order constraint on the squares of the weights, one could use the absolute values of the weights:

$$\min E_{\text{in}}(\mathbf{w})$$
$$\text{subject to } \sum_{i=0}^{d} |w_i| \leq C$$

The model is called the lasso algorithm.
(a) Formulate and implement this as a quadratic program.
(b) What is the augmented error and discuss the algorithm for solving it. You can solve this problem using iterative soft-thresholding algorithm or a gradient projection method and present your pseudocode.

Solution
(a) We can set $E_{\text{in}}(\mathbf{w})$ to be the L_2 loss, i.e.

$$E_{\text{in}}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n - y_n\right)^2 = \frac{1}{N}\|\mathbf{X}\mathbf{w}-\mathbf{y}\|_2^2$$

where $\mathbf{X}$ is the matrix of feature vectors, $\mathbf{y}$ is the vector of labels, and N is the number of data points, $\mathbf{x}_n$ is the feature vector of the n-th data point and y_n is the label.

For the objective function, we can rewrite it as:

$$\begin{aligned}\min_{\mathbf{w},\mathbf{t}} E_{\text{in}}(\mathbf{w}) &= \frac{1}{N}\|\mathbf{X}\mathbf{w}-\mathbf{y}\|_2^2 \\ &= \frac{1}{N}(\mathbf{X}\mathbf{w}-\mathbf{y})^T(\mathbf{X}\mathbf{w}-\mathbf{y}) \\ &= \frac{1}{N}(\mathbf{w}^T\mathbf{X}^T\mathbf{X}\mathbf{w} - 2\mathbf{w}^T\mathbf{X}^T\mathbf{y} + \mathbf{y}^T\mathbf{y})\end{aligned}$$

Which is a quadratic function of $\mathbf{w},\mathbf{t}$, and the definition of $\mathbf{t}$ is as followed.
For the constrains, we can slack the variables by letting $|w_i| \leq t_i$ for $i = 1, 2, \cdots, d$, and $t_i \geq 0$. Then we can rewrite the constrain as:

$$\begin{aligned}\sum_{i=0}^{d} t_i &\leq C \\ t_i &\geq 0, & i = 1, 2, \cdots, d \\ -t_i &\leq w_i \leq t_i, & i = 1, 2, \cdots, d\end{aligned}$$

Which are the linear constrains for $\mathbf{w},\mathbf{t}$.
Since the optimization problem is a quadratic programming problem with linear constrains, so it is a quadratic programming.

(b) The augmented error is defined as:

$$E_{\text{aug}}(\mathbf{w}) = E_{\text{in}}(\mathbf{w}) + \lambda\sum_{i=0}^{d}|w_i| = E_{\text{in}}(\mathbf{w}) + \lambda\|\mathbf{w}\|_1$$

where λ and is the hyperparameter.
Since L_1 norm is not differentiable, so we cannot simply use the gradient methods, but we can use the iterative soft-thresholding algorithm to solve it.

Define the regularization term to be $h(\mathbf{x}) = \|\mathbf{x}\|_1$.
The L_1 regularization's proximal term is $\text{prox}_{\lambda h}(\mathbf{x}) = \arg\min_{\mathbf{z}}\left\{\frac{1}{2}\|\mathbf{z}-\mathbf{x}\|_2^2 + \lambda h(\mathbf{z})\right\}$.
Since the proximal term is seperatable, so we can decompose into item by item optimization with

soft-thresholding.
i.e.

$$(\text{prox}_{\lambda h}(\mathbf{x}))_i = \psi_{\text{st}}(x_i, \lambda)$$

where ψ_{st} is the soft-thresholding function.
Then we analyze the soft-thresholding function:

$$\psi_{\text{st}}(x, \lambda) = \arg\min_{z_i} \left\{ \frac{1}{2}(z_i - x_i)^2 + \lambda |z_i| \right\}$$

- If $z_i \geq 0$, then $\arg\min_{z_i} \left\{ \frac{1}{2}z_i^2 + (\lambda - x_i)z_i + \frac{1}{2}x_i^2 \right\}$, which is a simple quadratic function.
  1. If $x_i \geq \lambda$, then $z_i = x_i - \lambda \geq 0$
  2. If $x_i < \lambda$, then $z_i = 0$

- If $z_i < 0$, then $\arg\min_{z_i} \left\{ \frac{1}{2}z_i^2 - (\lambda + x_i)z_i + \frac{1}{2}x_i^2 \right\}$, which is also a simple quadratic function.
  1. If $x_i \leq -\lambda$, then $z_i = x_i + \lambda \leq 0$
  2. If $x_i > -\lambda$, then $z_i = 0$

So combine all these cases together, we can get the soft-thresholding function:

$$\psi_{\text{st}}(x_i, \lambda) = \begin{cases} x_i - \lambda, & x_i > \lambda \\ 0, & |x_i| \leq \lambda \\ x_i + \lambda, & x_i < -\lambda \end{cases}$$

So with the soft-thresholding function, we can solve the Lasso problem by applying the proximal gradient method.

The pseudocode is shown in Algorithm 1.

Algorithm 1 Proximal Gradient Method for Lasso Problem

for $t = 0, 1, 2, \cdots$ **do**
    $\mathbf{w}^{(t+1)} \leftarrow \text{prox}_{\eta_t \lambda h}(\mathbf{w}^{(t)} - \eta_t \nabla E_{\text{in}}(\mathbf{w}^{(t)}))$
end for

Problem 3 Similar to problem 3 in assignment 3 and assignment 4, you need to use the SUV dataset to implement (using Python or MATLAB) the L_1/L_2 regularization (penalty/augmented).

- Present your code.
- Present the path plots of L_1 and L_2 regularization. (Notice: you need to mark the selected value of the regular parameter)
- Analyze the weight difference between L_1 and L_2 regularization. (Notice: you need to describe the similarities and differences between the solutions of path plots)
- If you only want to build a model that contains 2 variables, which two features would you choose?

Solution

1. The code and the method to run the code are all in the folder 'code'.

2. The path plots of L_1 and L_2 regularization are shown in Figure 1 and Figure 2. Where the feature 'Age' is seperate as: $0-20, 20-26, 26-30, 30-40, 40-50$, others.
And feature 'EstimatedSalary' is seperate as: $0-19500, 19500-40000, 40000-60000, 60000-80000, 80000-100000, 100000-130000, 130000-145000$, others.
All features are normalize before training.

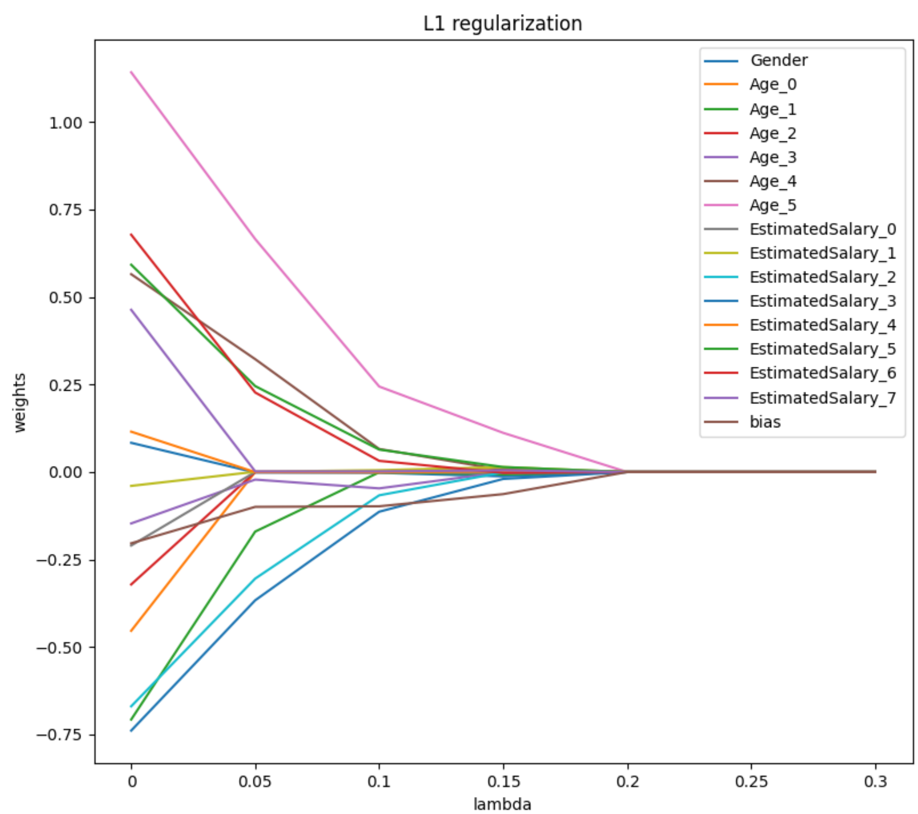


Figure 1: Path plot of L_1 regularization

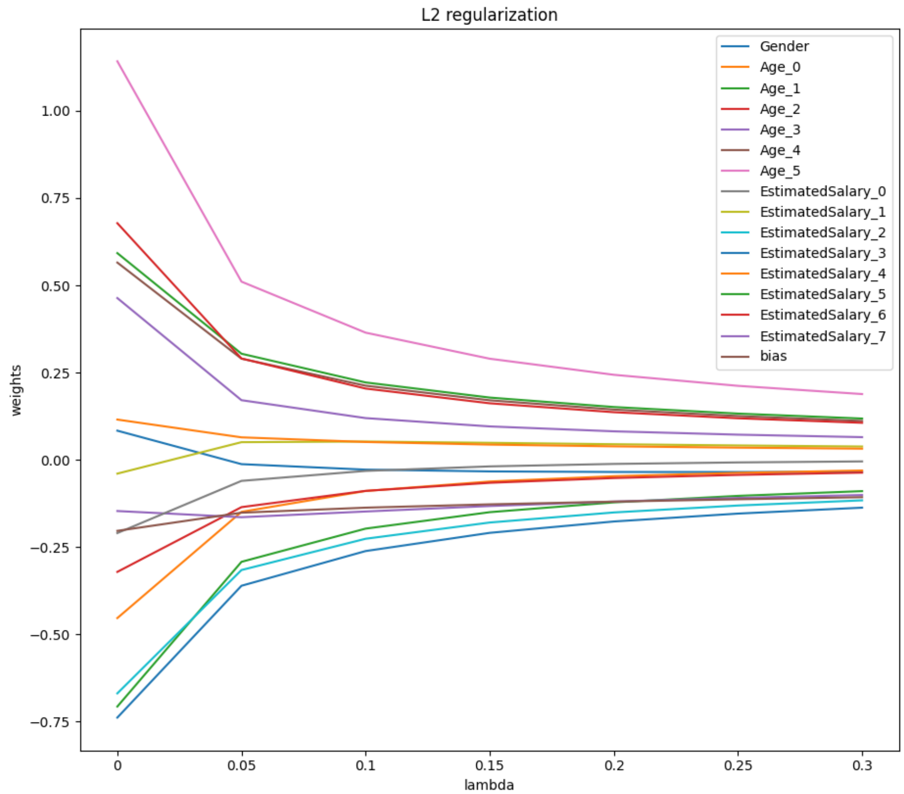


Figure 2: Path plot of L_2 regularization

3. We can see that both L_1 and L_2 regularization has the smaller weight as the regularization parameter λ increases. But the difference is that L_1 regularization can make some weights to be zero, which means that L_1 regularization can do feature selection. But L_2 regularization can only make the weights smaller, but not zero.

4. If only want to build a model that contains 2 variables, For L_1 regularization, 'Age_5' and 'bias' are chosen; for L_2 regularization, 'Age_5' and 'EstimatedSalary_3' are chosen.

This is because from the path plots, we can see that the weights of these two features have the largest absolute value when the regularization parameter λ gets bigger.