Assignment 1: CS 335 & CS 337

Richeek Das : 190260036

29th August 2021

1.1 CS337: Theory

$$\nabla mse(w,b) = \begin{pmatrix} \frac{\partial mse(w,b)}{\partial w} \\ \frac{\partial mse(w,b)}{\partial b} \end{pmatrix} \tag{1}$$

In our single variable case,

$$mse = \frac{1}{N}\sum_{i=1}^{N}((wx_i + b) - y_i)^2 \tag{2}$$

Therefore, its easy to calculate the gradient of this function with respect to w and b.

$$\frac{\partial mse(w,b)}{\partial w} = \frac{2}{N}\sum_{i=1}^{N}((wx_i + b) - y_i) \cdot x_i = \frac{2}{N}\sum_{i=1}^{N}(wx_i^2 + bx_i - y_i x_i) \tag{3}$$

$$\frac{\partial mse(w,b)}{\partial b} = \frac{2}{N}\sum_{i=1}^{N}((wx_i + b) - y_i) \cdot 1 = \frac{2}{N}\sum_{i=1}^{N}(wx_i + b - y_i) \tag{4}$$

1.2 CS335: Lab

c: `epochs=100, lr=1e-3`

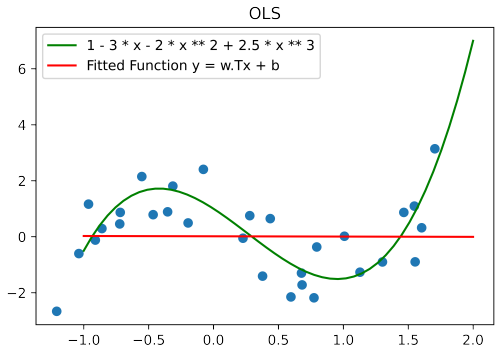

Figure 1: Single Variable Gradient Descent

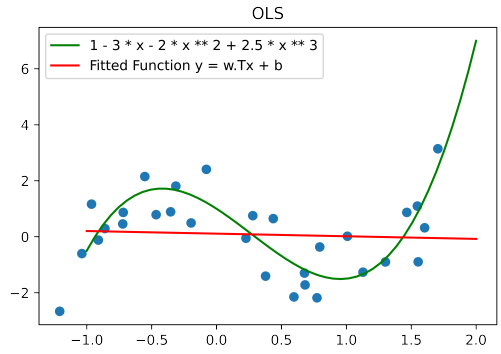

Figure 2: Single Variable Closed Form

d: No. `singlevar_grad()` can't achieve a lower training loss than `singlevar_closedform()`. Loss function or the mse we derived earlier is strictly convex and hence has a single global minima. We get the closed form expression by differentiating the mse and setting it to zero.

Given this has only a single solution, we get the global minima of mse. If the solution we obtain by doing gradient descent leads to a lower mse, it will contradict the definition of global minima.

2 OLS and Ridge Regression

2.1 CS337: Theory

a: We also need to incorporate the bias.

$$\hat{Y} = XW + b = X'W'$$

where X' has an extra column of **ones** appended to the front. So $X' \in \mathbb{R}^{N \times d+1}$. Also W' is the set of weights plus the bias term appended to the front. So $W' \in \mathbb{R}^{d+1}$. So $\hat{Y}$ turns out to be N dimensional vector as expected. **We will work with this W' and X' for the rest of this problem**.

b:

$$\begin{aligned} mse &= \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 \\ &= \frac{1}{N}(X'W' - Y)^T(X'W' - Y) \\ &= \frac{1}{N}(W'^T X'^T - Y^T)(X'W' - Y) \\ &= \frac{1}{N}\left(W'^T X'^T X'W' - Y^T X'W' - W'^T X'^T Y + Y^T Y\right) \end{aligned}$$

We know W' is a vector. So let's differentiate the above expression wrt W'

$$\frac{\partial mse}{\partial W'} = \frac{1}{N}\left(W'^T(X'^T X' + X'^T X') - Y^T X' - (X'^T Y)^T + 0\right) \tag{5}$$

$$= \frac{2}{N}\left(W'^T X'^T X' - Y^T X'\right) = \frac{2}{N}\left(W'^T X'^T - Y^T\right) X' \tag{6}$$

But, with this equation we get a $1 \times d+1$ vector. We will want a $d+1 \times 1$ vector instead for easier calculation and better looking equations. So, let's transpose this

$$\frac{\partial mse}{\partial W'} = \frac{2}{N} X'^T\left(X'W' - Y\right) \tag{7}$$

To get a **closed form**, we set this to **0**.

$$\begin{aligned} X'^T X'W' &= X'^T Y \\ \implies W' &= (X'^T X')^{-1} X'^T Y \end{aligned}$$

c: For ridge regression, mse:

$$\begin{aligned} ridge\_mse &= \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 + \lambda\|W'\|^2 \\ &= \frac{1}{N}(X'W' - Y)^T(X'W' - Y) + \lambda W'^T W' \\ &= \frac{1}{N}(W'^T X'^T - Y^T)(X'W' - Y) + \lambda W'^T W' \\ &= \frac{1}{N}\left(W'^T X'^T X'W' - Y^T X'W' - W'^T X'^T Y + Y^T Y\right) + \lambda W'^T W' \end{aligned}$$

Differentiating this wrt W' gives:

$$\frac{\partial ridge\_mse}{\partial W'} = \frac{2}{N} X'^T \left(X'W' - Y\right) + \lambda \cdot 2W' \tag{8}$$

To get a **closed form**, we set this to **0**.

$$\begin{aligned} \frac{1}{N} X'^T X'W' + \lambda W' &= \frac{1}{N} X'^T Y \\ \implies \left(\frac{1}{N} X'^T X' + \lambda I\right) W' &= \frac{1}{N} X'^T Y \\ \implies W' &= \left(\frac{1}{N} X'^T X' + \lambda I\right)^{-1} \frac{1}{N} X'^T Y \end{aligned}$$

d: For the closed form of OLS to exist, $\mathbf{X'^T X'}$ must be invertible. We know that $\mathbf{X^T X}$ is invertible if $\mathbf{X}$ is full rank. Also we know $rank(\mathbf{AB}) \leq \min(rank(\mathbf{A}), rank(\mathbf{B}))$. **Therefore, if columns of X are not linearly independent, X won't be full-rank and hence $\mathbf{X^T X}$ won't be invertible.**

Yes it can. If the closed form doesn't exist, it basically means that a unique inverse is not present. There might be multiple global minima(in the subspace spanned by the linearly independent columns of $\mathbf{X}$). If we do a gradient descent, we will end up in one of the global minima (since it moves in the direction of decreasing gradient, **and the loss function remains convex, irrespective of the columns of X being linearly dependent**) which essentially means, "converging to a solution".

2.2 CS335: Lab

b: Multivariate Least Squares Regression with Gradient Descent and Multivariate LSR with Closed Form . `epochs=100000, lr=1e-3, lambda=0.01`

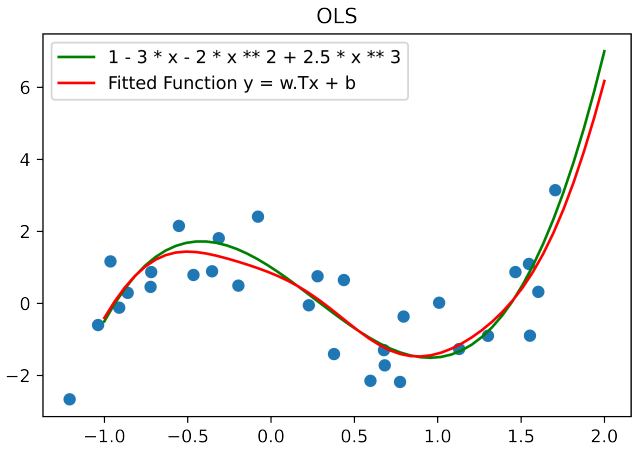

Figure 3: Multi Variable LSR with Gradient Descent

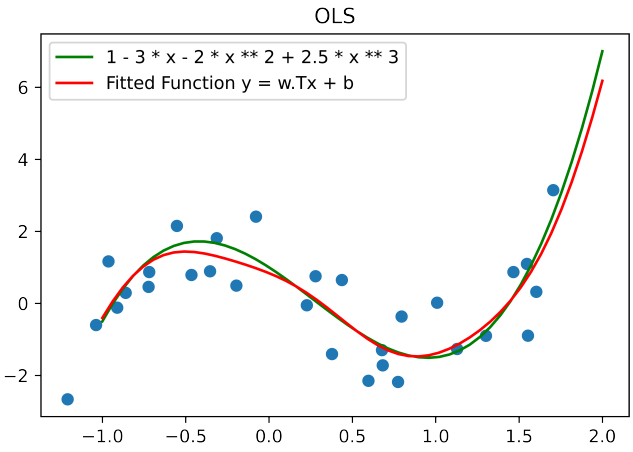

Figure 4: Multi Variable LSR with Closed Form

c: Multi variable Ridge Regression with Gradient Descent and Multi variable Ridge Regression with Closed Form. `epochs=100000, lr=1e-3, lamda=0.01`

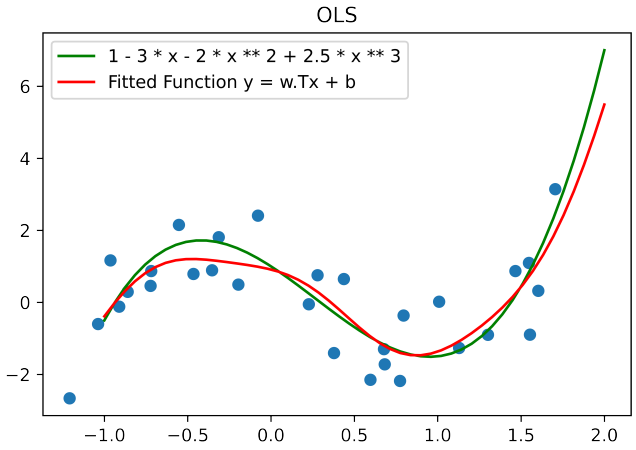

Figure 5: Multi Variable Ridge Regression with Gradient Descent

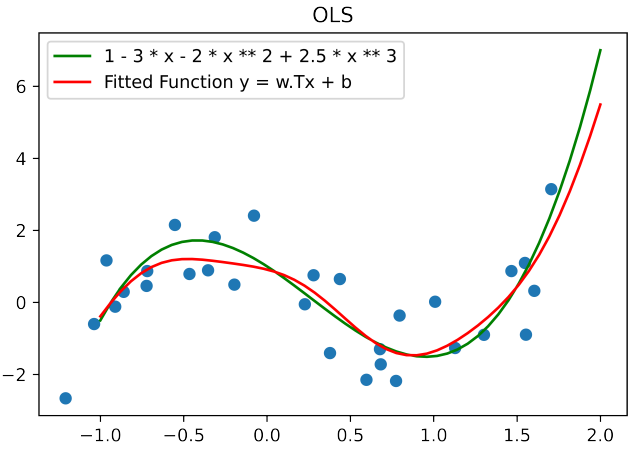

Figure 6: Multi Variable Ridge Regression with Closed Form

3 Bayesian Linear Regression

3.1 CS337: Theory

a: Expand it into gaussian expression:

$$p(w) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(w-\mu_0)^2\right) \tag{9}$$

c: Given IID samples:

$$p(D|w) = \prod_{i=1}^{N} p(y_i|x_i;w) \tag{10}$$

$$= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(y_i - wx_i)^2\right) \tag{11}$$

$$= \frac{1}{(2\pi)^{N/2}} \exp\left(-\frac{1}{2}\sum_{i=1}^{N}(y_i - wx_i)^2\right) \tag{12}$$

d: Using Bayes Theorem:

$$p(w|D) = \frac{p(D|w)p(w)}{p(D)} \tag{13}$$

$$= \frac{\frac{1}{(2\pi)^{N/2}} \exp\left(-\frac{1}{2}\sum_{i=1}^{N}(y_i - wx_i)^2\right) \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(w-\mu_0)^2\right)}{p(D)} \tag{14}$$

$$= \frac{\frac{1}{(2\pi)^{N+1/2}} \exp\left(-\frac{1}{2}\sum_{i=1}^{N}(y_i - wx_i)^2 - \frac{1}{2}(w-\mu_0)^2\right)}{p(D)} \tag{15}$$

h: Let's compare those two:

Comparing the w^2 terms in the exponent:

$$-\frac{1}{2\sigma_N^2} = -\frac{1}{2}\left(1 + \sum_{i=1}^{N} x_i^2\right) \tag{16}$$

$$\implies \sigma_N^2 = \frac{1}{1 + \sum_{i=1}^{N} x_i^2} \tag{17}$$

Comparing the w^1 terms in the exponents:

$$\frac{\mu_N}{\sigma_N^2} = \mu_0 + \sum_{i=1}^{N} x_i y_i \tag{18}$$

$$\implies \mu_N = \frac{\mu_0 + \sum_{i=1}^{N} x_i y_i}{1 + \sum_{i=1}^{N} x_i^2} \tag{19}$$

i: When we observe a lot of data, i.e $N \longrightarrow \infty$ we expect $\sum_i x_i^2 >> 1$ and $\sum_i x_i^2 >> \mu_0$ since each $x_i^2 \geq 0$ and equality occurring only at $x_i = 0$ which describes only a single data point.

Therefore,

$$\sigma_N^2 \longrightarrow \frac{1}{\sum_{i=1}^{N} x_i^2} \tag{20}$$

$$\mu_N \longrightarrow \frac{\sum_{i=1}^{N} x_i y_i}{\sum_{i=1}^{N} x_i^2} \tag{21}$$

j: We can draw some intuitive observations.

- As we see more data, our prior beliefs fade away. Our beliefs get modified a lot by the new data.
- As N increases, $\sum_{i=1}^{N} x_i^2$ increases. So the variance of posterior decreases. We can interpret this as $\longrightarrow$ our confidence in the weights based on the observed data increases.

3.2 MLE Estimate

a: MLE Estimate:

$$w^* = \arg\max_w p(D|w) = \arg\max_w \log(p(D|w)) \tag{22}$$

$$= \arg\max_w \left(-\frac{1}{2}\sum_{i=1}^{N}(y_i - wx_i)^2\right) = \arg\min_w \left(\sum_{i=1}^{N}(y_i - wx_i)^2\right) \tag{23}$$

$$\implies \frac{\partial}{\partial w}\left(\sum_{i=1}^{N}(y_i - wx_i)^2\right) = 0 \tag{24}$$

$$\implies \sum_{i=1}^{N} 2(y_i - wx_i)x_i = 0 \tag{25}$$

$$\implies w^* = \frac{\sum_{i=1}^{N} x_i y_i}{\sum_{i=1}^{N} x_i^2} \tag{26}$$

b: The w^* we obtain as the MLE, matches with the μ_∞ of the posterior of Bayesian Estimation. This is obvious because as we see more and more data, our confidence in the weights increases. $\sigma_\infty^2 \longrightarrow 0$ and w becomes a random variable with mean at w^* and variance 0.

3.3 CS335: Lab

a: Multi Variable curve approximation with Bayesian Linear Regression. σ=0.1

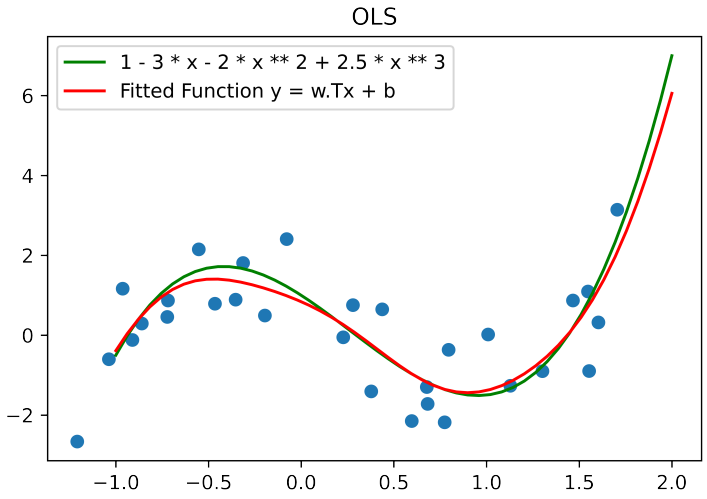

Figure 7: Multi Variable Bayesian Linear Regression

b: For OLS the closed form expression is $\mathbf{W} = (\mathbf{\Phi^T \Phi})^{-1}\mathbf{\Phi^T})\mathbf{Y}$. In this case, if the columns of $\mathbf{\Phi}$ are not linearly independent, $\mathbf{\Phi^T \Phi}$ won't be invertible.

In Bayesian Linear Regression, $\mathbf{\Sigma_N} = (\mathbf{\Sigma_0^{-1}} + \frac{\mathbf{1}}{\sigma^2}\mathbf{\Phi^T \Phi})^{-1}$. In this case, even if the columns of $\mathbf{\Phi}$ are not linearly independent, $\mathbf{\Sigma_0^{-1}} + \frac{\mathbf{1}}{\sigma^2}\mathbf{\Phi^T \Phi}$ will still be invertible. It is always invertible. We will proceed to prove this:

We know two things about the aforementioned expression. $\mathbf{\Sigma_0}$ is a covariance matrix, so it is symmetric, positive semi-definite and invertible(**full-rank**). We also know that $\mathbf{\Phi^T \Phi}$ is symmetric and positive semi-definite. We will prove:

$$rank(\mathbf{A} + \mathbf{B}) \geq \max\left(rank(\mathbf{A}), rank(\mathbf{B})\right) \tag{27}$$

if $\mathbf{A}, \mathbf{B}$ are positive semi-definite matrices.

Let $\mathcal{N}(X)$ be the null space of X. Then for $v \in \mathcal{N}(X + Y)$:

$$v^T(X + Y)v = 0$$
$$\implies v^T X v + v^T Y v = 0$$

Since they are positive semidefinite matrices, we know $v^T X v \geq 0$ and $v^T Y v \geq 0$ for all v. Hence, $v^T X v = 0$ and $v^T Y v = 0$ if $v \in \mathcal{N}(X + Y)$.

Therefore,

$$\mathcal{N}(X + Y) = \mathcal{N}(X) \cap \mathcal{N}(Y)$$

So by **rank-nullity theorem**:

$$\begin{aligned} rank(X+Y) &= n - dim(\mathcal{N}(X+Y)) \\ &= n - dim(\mathcal{N}(X) \cap \mathcal{N}(Y)) \\ &\geq n - dim(\mathcal{N}(X)) \\ &= rank(X) \end{aligned}$$

Therefore, we proved $rank(X+Y) \geq rank(X)$. Similarly we can prove $rank(X+Y) \geq rank(Y)$.

Now we utilise this result! We know both $\mathbf{\Sigma_0}$ and $\mathbf{\Phi^T \Phi}$ are PSD matrices. Therefore, $rank(\mathbf{\Sigma_0^{-1}} + \frac{\mathbf{1}}{\sigma^2}\mathbf{\Phi^T \Phi}) \geq \max(rank(\mathbf{\Sigma_0^{-1}}), rank(\mathbf{\Phi^T \Phi}))$. We know, $rank(\mathbf{\Sigma_0}) = n$, where $\mathbf{\Sigma_0} \in \mathbb{R}^{n \times n}$. Therefore $rank(\mathbf{\Sigma_0^{-1}} + \frac{\mathbf{1}}{\sigma^2}\mathbf{\Phi^T \Phi}) \geq n$. Hence it's always full-rank and invertible. **(Proved.)**

4 Conclusion

We time the functions.

- `multivar_grad`: **639 ms ± 7.64 ms** for **100000 epochs**, since this gives a comparable solution with the closedform one.

  Validation loss of 0.3851

  Training Loss loss of 0.6649

- `multivar_closedform`: **274 µs ± 11.4 µs** with the dataset we have.

  Validation loss of 0.3851

  Training Loss loss of 0.6649

We see among the `multivar_grad` and `multivar_closedform` solutions `multivar_closedform` provides vastly faster and better solutions. `multivar_grad` has to run for over **100000 epochs** to get an as accurate solution as the closedform! But, there is a scalability to the gradient descent solution that the closedform lacks. As the training dataset becomes larger, it becomes almost impossible to store the $X^T X$, forget about calculating its inverse.

Verdict: For smaller datasets $\longrightarrow$ `multivar_closedform`. For larger datasets $\longrightarrow$ `multivar_grad`.

We time more functions.

- `multivar_reg_grad`: **762 ms ± 4.93 ms** for **100000 epochs**, since this gives a comparable solution with the closedform one.

  Validation loss of 0.4212

  Training Loss loss of 0.6882

- `multivar_reg_closedform`: **302 µs ± 10.6 µs** with the dataset we have.

  Validation loss of 0.4212

  Training Loss loss of 0.6881

Again the same thing. Gradient descent is much slower than the closed form. But we know that the closedform is not scalable. Once the dataset becomes larger, we can't even store $X^T X$.

Verdict: For smaller datasets $\longrightarrow$ `multivar_reg_closedform`. For larger datasets $\longrightarrow$ `multivar_reg_grad`.

We time more functions:

- `bayesion_lr`: Bayesian Linear Regression, **223 µs ± 4.15 µs** with our dataset.

  Validation loss of 0.3851

  Training Loss loss of 0.6649

We see in terms of speed: `bayesion_lr` > `multivar_closedform` > `multivar_reg_closedform` >> `multivar_grad` > `multivar_reg_grad`

Where > represents faster.