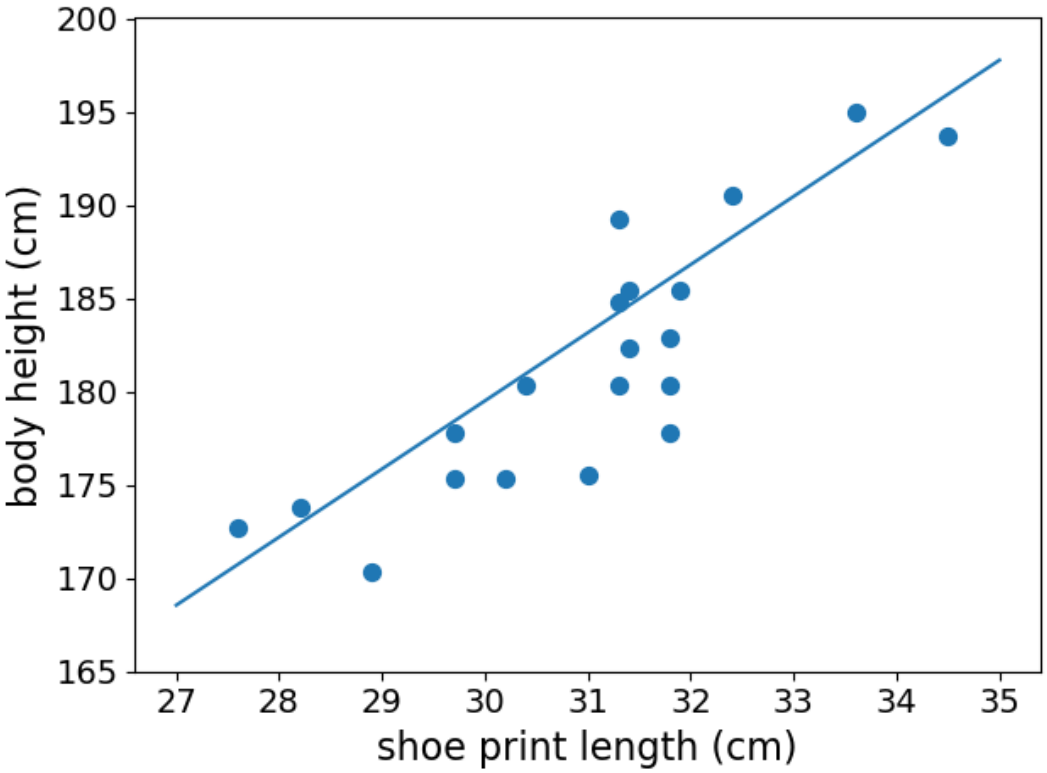
CS5228 – Tutorial 7

Classification & Regression III (Linear Models)

1. **Linear Regression.** The figure below shows the CSI example from the lecture, where we try to predict the height of a suspect given the length of a shoe print we found. The blue line represents the best fit of a Linear Regression Model.



(a) The most commonly used loss function L for Linear Regression is the Means Squared Error (MSE):

$$L = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$$

What makes MSE a suitable loss function for Linear Regression, and what part of the formula might cause "issues"?

Solution:

- The MSE loss intuitively captures what makes a good or bad solution – that is, on average, the line should be close to all data points.
- Squaring the residuals ensures that each individual loss is positive, no matter if the the try value y is above or below the predicted value $\hat{y}$
- The squaring is mathematically convenient when it comes to calculating the derivative of L.
- However, the squaring also potentially over-emphasizes larger residuals making Linear Regression quite sensitive to outliers.

(b) After training a Linear Regression model over a dataset, how can we use the result to identify if there is indeed some linear relationship between the input features and the output values.

Solution:

- We can't. Linear Regression only gives us a linear mapping from the input features to the output values, independent if there is indeed any meaningful linear relationship.
- We therefore have to assess before applying Linear Regression if there is indeed a linear relationship (e.g., using correlation.)

(c) Let's assume our data features strong linear relationships between the input features in output values, and we train a Linear Regression Model. Now we want to predict the values for unseen data points. In which case (i.e., for which type of data points) we might have to be very cautious when interpreting the predicted values?

Solution:

- As long as a data point is in the range of the training data (*interpolation*) we are generally on the safe side.
- For any data points outside the range of the data (*extrapolation*), we may not guarantee if the predicted values are meaningful:
  - The data points might not be meaningful. For example, a negative shoe print size does not make sense, but we can give it to the Linear Model and get some result.
  - Outside the range of our training data, the assumption of a linear relationship might no longer hold.

Note that this result might vary when considering more than this single feature.

(d) For our CSI example, could we train a Linear Regression model that only requires the coefficient of our single feature *length* and does not require the bias weight?

Solution: As, we can train a model without an explicit consideration of the bias

- We know that the regression line will go through the mean of our feature *length* and output *height*.
- If we standardize our data before training, we know that the regression line will go through the origin $(0, 0)$.
- We therefore already know that the bias/intercept will be 0.

(e) In the lecture, we only skimmed over the calculation of the derivative of loss function L. Given the matrix notation of L

$$L = \frac{1}{n}\|X\theta - y\|^2$$

find $\frac{\partial L}{\partial \theta}$!

Solution:

With Matrix Calculus: Firstly, let's ignore the $\frac{1}{n}$ to have less to write. We can do this because this constant will be irrelevant once we set $\frac{\partial L}{\partial \theta} = 0$. So we start with

$$L = \|X\theta - y\|^2$$

Now, $\|\overrightarrow{x}\|^2 = \sum_{i=1}^{n} x_i^2$ is the squared vector norm, summing up the squares of all vector elements. This allows us to rewrite L as follows:

$$L = (X\theta - y)^T(X\theta - y)$$

Note that with $\overrightarrow{x} = (X\theta - y)$ both formulations are indeed equivalent.

Now we can apply various matrix algebra rules to further simplify the formula. Applying $(A - B)^T = A^T - B^T$ (with A, B being vectors/matrices), we can write

$$L = ((X\theta)^T - y^T)(X\theta - y)$$

We can now multiply both parts to get rid of the product of sums.

$$L = (X\theta)^T X\theta - (X\theta)^T y - y^T(X\theta) + y^T y$$

We can now apply the rule $(AB)^T = B^T A^T$. Also note that $y^T(X\theta) = (X\theta)^T y$. y and $X\theta$ are just vectors, so multiplying them either way yields the same results.

$$L = \theta^T X^T X\theta - 2(X\theta)^T y + y^T y$$

This is as much as we can simplify L. Now we can calculate $\frac{\partial L}{\partial \theta}$. Since we have a sum of 3 parts, we can look at all parts individually. Let's start at the back. With respect to θ, $y^T y$ is a constants, so

$$\frac{\partial y^T y}{\partial \theta} = 0$$

For the middle part, we can apply the rule from matrix calculus stating that $\frac{\partial A\vec{x}}{\partial \vec{x}} = A^T$. As such,

$$\frac{\partial 2(X\theta)^T y}{\partial \theta} = 2X^T y$$

The first part is a bit less obvious and might actually require writing out the individual equations. However, at the end on can apply the following rule $\frac{\partial \vec{x}^T \vec{x}}{\partial \vec{x}} = 2\vec{x}$, giving us

$$\frac{\partial \theta^T X^T X\theta}{\partial \theta} = 2X^T X\theta$$

Putting everything together we finally get:

$$\frac{\partial L}{\partial \theta} = 2X^T X\theta - 2X^T y$$

If we want, we could factor out $2X^T$ to get the formula we saw in the lecture:

$$\frac{\partial L}{\partial \theta} = 2X^T(X\theta - y)$$

Just note that here we omitted the $\frac{1}{n}$ while we kept it for the formula on the lecture slides.

Lastly, to get the Normal Equation, setting $\frac{\partial L}{\partial \theta} = 0$ and solving for θ is comparatively straightforward, and this step we already have covered in the lecture.

Without Matrix Calculus: Again, we first ignore the $\frac{1}{n}$. In this case, we start with:

$$L = \sum_{i=1}^{n} (\theta^T x_i - y_i)^2 = \sum_{i=1}^{n} \left(\sum_{j=0}^{d} \theta_j x_i^{(j)} - y_i \right)^2$$

Now we can directly start calculating the derivative $\frac{\partial L}{\partial \theta_k}$. For this, we need the chain rule.

$$\frac{\partial L}{\partial \theta_k} = 2 \sum_{i=1}^{n} x_i^k \left(\sum_{j=0}^{d} \theta_j x_i^{(j)} - y_i \right)$$

We can simplify this, by vectorizing the sum in the parentheses – basically the opposite of what we have done in the beginning.

$$\frac{\partial L}{\partial \theta_k} = 2 \sum_{i=1}^{n} x_i^k \left(\theta^T x_i - y_i\right)$$

We now have the partial derivative with respect to a single θ_i. We can now vectorize it further to get the partial derivative with respect to θ.

$$\frac{\partial L}{\partial \theta} = 2X^T(X\theta - y)$$

Unsurprisingly, this brings us to the same derivative as above. Converting the sum into matrix form might not be that intuitive. If you doubt this step, you can makes this yourself clear by actually going through an example – e.g., with $i = 3$ and $d = 3$ – to see the equations are equivalent

2. **Logistic Regression (optional).** We have seen that Linear Regression and Logistic Regression are very similar. Simply speaking, Logistic Regression uses the basic linear signal $\theta^T x_i$ and puts it through a function f that maps this signal to a value of range $[0, 1]$ so that the final value can be interpreted as a probability. The Sigmoid $\sigma(z)$ function is most commonly used to implement f:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

With $\theta^T x_i$ being our linear signal, we can calculate the predicted value $\hat{y}_i$ for data sample x_i as follows:

$$\hat{y}_i = \frac{1}{1 + e^{\theta^T x_i}}$$

This ultimately brought us to the *Cross-Entropy Loss*, the loss function for Logistic Regression (binary classification):

$$\begin{aligned} L &= -\frac{1}{n} \sum_{i=1}^{n} [y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i)] \\ &= -\frac{1}{n} \sum_{i=1}^{n} \left[y_i \log \frac{1}{1 + e^{\theta^T x_i}} + (1 - y_i) \log \left(1 - \frac{1}{1 + e^{\theta^T x_i}}\right) \right] \end{aligned}$$

Well, find $\frac{\partial L}{\partial \theta}$! :)

Solution: If you recall from the lecture, the basic setup for both Linear and Logistic Regression is $\hat{y} = h_\theta(x) = f(\theta^T x)$. With Linear Regression, function f was simply the identity function so we got $\hat{y} = h_\theta(x) = \theta^T x$. In the case of Logistic Regression we use the Sigmoid as function f to squeeze all values between 0 and 1, hence $\hat{y} = h_\theta(x) = \sigma(\theta^T x)$.

Step 1 – Calculate the derivative of $\sigma(z)$. From above, $\sigma(z)$ is given as

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

To find the derivative $\frac{d(\sigma(z))}{dz}$ we need to apply both the quotient and chain rule:

$$\frac{d(\sigma(z))}{dz} = \frac{0 \cdot (1 + e^{-z}) - (1) \cdot (e^{-z} \cdot (-1))}{(1 + e^{-z})^2}$$

This immediately simplifies to

$$\frac{d(\sigma(z))}{dz} = \frac{(e^{-z})}{(1 + e^{-z})^2}$$

Now we can apply some mathematical trickery to further transform this equation. While it first looks like we are making the equation more complicated, you will see the usefulness at the end. Let's just add and subtract 1 from the numerator

$$\frac{d(\sigma(z))}{dz} = \frac{1 + (e^{-z}) - 1}{(1 + e^{-z})^2}$$

With this we can split the quotient into

$$\frac{d(\sigma(z))}{dz} = \frac{1 + (e^{-z})}{(1 + e^{-z})^2} - \frac{1}{(1 + e^{-z})^2} = \frac{1}{(1 + e^{-z})} - \frac{1}{(1 + e^{-z})^2}$$

By pulling out $\frac{1}{1+e^{-z}}$ we get

$$\frac{d(\sigma(z))}{dz} = \frac{1}{1 + e^{-z}} \left(1 - \frac{1}{1 + e^{-z}}\right)$$

which in turn is the same as

$$\frac{d(\sigma(z))}{dz} = \sigma(z)(1 - \sigma(z))$$

This makes the Sigmoid function so convenient in practice, since we can calculate the gradient at a position z by simply calculating $\sigma(z)$ and combining the result according to the formula above.

Step 2 – Calculating partial derivatives $\frac{\partial L}{\partial \theta_i}$. With $\hat{y} = h_\theta(x) = \sigma(\theta^T x)$, we can write our loss function L as follows:

$$L = \frac{1}{n}\sum_{i=1}^{n}\left[y_i \log \sigma(\theta^T x_i) + (1-y_i)\log\left(1-\sigma(\theta^T x_i)\right)\right]$$

With this we can directly calculate $\frac{\partial L}{\partial \theta_i}$. I good start is to recall that $\frac{d}{dx}\log f(x) = \frac{1}{f(x)}\frac{d}{dx}f(x)$. A second thing to consider is to apply the chain rule to $\sigma(\theta^T x_i)$ since now the input for σ is here a function of θ. Now, the derivative $\frac{d(\theta^T x_i)}{d\theta_j} = x_{ij}$, i.e., the j-the feature of data sample x_i. This is easy to see when you write the vector product $\theta^T x_i$ as the sum of the vector elements and then calculate the derivative. All of this applied to our calculation this yields:

$$\frac{\partial L}{\partial \theta_j} = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{y}{\sigma(\theta^T x_i)}\sigma(\theta^T x_i)(1-\sigma(\theta^T x))x_{ij} + \frac{1-y}{1-\sigma(\theta^T x_i)}(-1)\sigma(\theta^T x_i)(1-\sigma(\theta^T x_i))x_{ij}\right]$$

No this looks a bit hefty at first, but notice that both parts of the inner sum have the same factor $\sigma(\theta^T x)(1-\sigma(\theta^T x))x_{ij}$ which we can factor out, yielding

$$\frac{\partial L}{\partial \theta_j} = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{y}{\sigma(\theta^T x_i)} - \frac{1-y}{1-\sigma(\theta^T x_i)}\right]\sigma(\theta^T x_i)(1-\sigma(\theta^T x))x_{ij}$$

Note how the (-1) flipped the sign between the two inner terms. If we bring the 2 terms in the brackets to the same denominator, we get:

$$\frac{\partial L}{\partial \theta_i} = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{y - y\sigma(\theta^T x_i) - (\sigma(\theta^T x_i) - y\sigma(\theta^T x_i)}{\sigma(\theta^T x_i)(1-\sigma(\theta^T x_i))}\right]\sigma(\theta^T x_i)(1-\sigma(\theta^T x_i))x_{ij}$$

$$\frac{\partial L}{\partial \theta_j} = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{y - \sigma(\theta^T x_i)}{\sigma(\theta^T x_i)(1-\sigma(\theta^T x_i))}\right]\sigma(\theta^T x_i)(1-\sigma(\theta^T x_i))x_{ij}$$

We can see that we can now cancel quite a lot of terms:

$$\frac{\partial L}{\partial \theta_j} = \frac{1}{n}\sum_{i=1}^{n}(y - \sigma(\theta^T x_i))x_{ij}$$

And with this, we are basically done. The last step is again to convert is partial derivative for one θ_j to the partial derivative for vector θ, which gives us the gradient we saw in the lecture:

$$\frac{\partial L}{\partial \theta} = \frac{1}{n}X^T(\sigma(\theta^T X) - y)$$

where $\sigma(\theta^T X)$ is simply our $h_\theta(X)$:

$$\frac{\partial L}{\partial \theta} = \frac{1}{n} X^T (h_\theta(X) - y)$$