COMP9417 Machine Learning and Data Mining

Lecture(s): Regression
Content: Review; Questions on topics in lecture

Review

R1

Obtain the partial derivatives with respect to one value, when

$$f(x, y) = a_1 x^2 y^2 + a_4 xy + a_5 x + a_7$$

R2

When

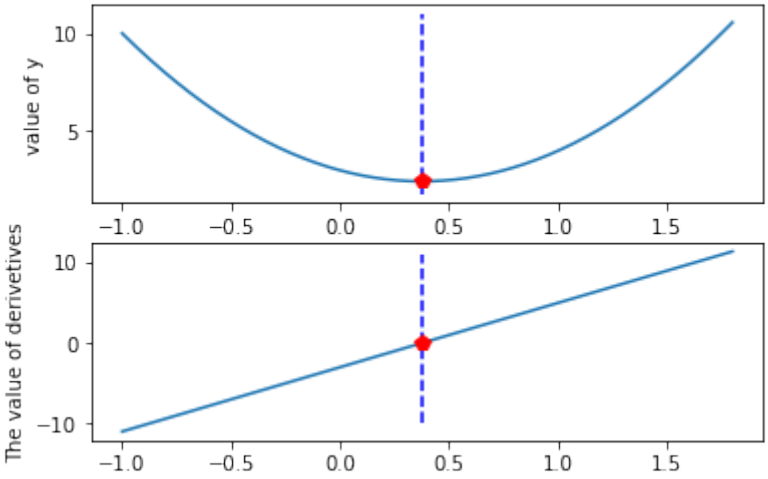
$$f(x, y) = a_1 x^2 y^2 + a_2 x^2 y + a_3 x y^2 + a_4 xy + a_5 x + a_6 y + a_7$$

what will $\frac{\partial f(x,y)}{\partial x}$ and $\frac{\partial f(x,y)}{\partial y}$ be?

R3

Do you recall how to solve optimization problems for Quadratic function? For example, $y = 4x^2 - 3x + 3$. Is the solution a minimum or maximum?

Hint:



R4

What is the *loss function* for linear regression?

Q1

Go through this derivation and complete the exercise at the end of it.

A *univariate linear regression model* is a linear equation $y = a + bx$. Learning such a model requires fitting it to a sample of training data so as to minimize the error function $\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(y_i - (w_0 + w_1 x_i))^2$. To find the best parameters a and b that minimize this error function we need to find the error *gradients* $\frac{\partial \mathcal{L}}{\partial w_0}$ and $\frac{\partial \mathcal{L}}{\partial w_1}$. So we need to derive these expressions by taking partial derivatives, set them to zero, and solve for w_0 and w_1.

First we write the loss function for the univariate linear regression $y = w_0 + w_1 x$ as

$$\mathcal{L} \;=\; \mathcal{L}(w_0, w_1) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (w_0 + w_1 x_i))^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - w_0 - w_1 x_i)^2$$

At a minimum of $\mathcal{L}$, the partial derivatives with respect to w_0, w_1 should be zero. We will start with taking the partial derivative of $\mathcal{L}$ with respect to w_0:

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial w_0} &= \frac{\partial}{\partial w_0}\frac{1}{n}\sum_{i=1}^{n}(y_i - w_0 - w_1 x_i)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial w_0}(y_i - w_0 - w_1 x_i)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n} -2(y_i - w_0 - w_1 x_i) \\
&= -2\left[\frac{1}{n}\sum_{i=1}^{n} y_i - w_0\frac{1}{n}\sum_{i=1}^{n} 1 - w_1\frac{1}{n}\sum_{i=1}^{n} x_i\right] \\
&= -2\left[\overline{y} - w_0 - w_1\overline{x}\right],
\end{aligned}$$

where we have introduced the notation $\overline{f}$ to mean the sample average of f, i.e. $\overline{f} = \frac{1}{m}\sum_{j=1}^{m} f_j$, where m is the length of f. Now, we equate this to zero and solve for w_0 to get

$$-2\left[\overline{y} - w_0 - w_1\overline{x}\right] \;=\; 0 \implies w_0 = \overline{y} - w_1\overline{x}$$

Note, we have not actually solved for w_0 yet, since our expression depends on w_1, which we must also optimise over.

Taking the partial derivative of $\mathcal{L}$ with respect to w_1:

$$
\begin{aligned}
\frac{\partial \mathcal{L}}{\partial w_1} &= \frac{\partial}{\partial w_1} \frac{1}{n} \sum_{i=1}^{n} (y_i - w_0 - w_1 x_i)^2 \\
&= \frac{1}{n} \sum_{i=1}^{n} -2x_i (y_i - w_0 - w_1 x_i) \\
&= -2 \left[\frac{1}{n} \sum_{i=1}^{n} x_i y_i - w_0 \frac{1}{n} \sum_{i=1}^{n} x_i - w_1 \frac{1}{n} \sum_{i=1}^{n} x_i^2 \right] \\
&= -2 \left[\overline{xy} - w_0 \overline{x} - w_1 \overline{x^2} \right],
\end{aligned}
$$

Now, we equate this to zero and solve for w_1 to get

$$
-2 \left[\overline{xy} - w_0 \overline{x} - w_1 \overline{x^2} \right] = 0 \implies w_1 = \frac{\overline{xy} - w_0 \overline{x}}{\overline{x^2}}
$$

We now have an expression for w_0 in terms of w_1, and an expression for w_1 in terms of w_0. These are known as the Normal Equations. In order to get an explicit solution for w_0, w_1, we can plug w_0 into w_1 and solve:

$$
\begin{aligned}
w_1 &= \frac{\overline{xy} - w_0 \overline{x}}{\overline{x^2}} \\
&= \frac{\overline{xy} - (\overline{y} - w_1 \overline{x}) \overline{x}}{\overline{x^2}} \\
&= \frac{\overline{xy} - \overline{x}\ \overline{y} + w_1 \overline{x}^2}{\overline{x^2}}
\end{aligned}
$$

Rearranging and solving for w_1 gives us

$$
w_1 = \frac{\overline{xy} - \overline{x}\ \overline{y}}{\overline{x^2} - \overline{x}^2}
$$

So now we have an explicit solution for the regression parameters w_0 and w_1, and so we are done.

Exercise: To make sure you know the process, try to solve the following loss function for linear regression with a version of "$L2$" regularization, in which we add a penalty that penalizes the size of w_1. Let $\lambda > 0$ and consider the regularised loss

$$
\mathcal{L}(w_0, w_1) = \frac{1}{n} \sum_{i=1}^{n} (y_i - (w_0 + w_1 x_i))^2 + \lambda w_1^2
$$

Q2

An intuitive understanding of the *regression coefficient* w_1 for univariate regression is that it defined as:

$$\frac{\text{covariance of } x \text{ and } y}{\text{variance of } x}$$

and a straightforward, if inefficient, way to compute this is:

$$w_1 = \frac{\Sigma(x - \bar{x})(y - \bar{y})}{\Sigma(x - \bar{x})^2}$$

where $\bar{v}$ represents the sample-mean of the values in the dataset for variable v. Once w_1 is obtained we can find $w_0 = \bar{y} - w_1\bar{x}$. Apply this method to determine the linear regression equation $y = w_0 + w_1 x$ for the small dataset below.

x	y
3	13
6	8
7	11
8	2
11	6

However, the same univariate regression can be written in matrix notation as

$$\widehat{\mathbf{w}} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}$$

(this expression is used for multivariate linear regression, but it also applies to univariate linear regression using homogeneous coordinates).

We can see that this expression essentially is the variance of x represented by $(\mathbf{X}^{\mathrm{T}}\mathbf{X})$) for which we take the inverse, multiplied by the covariance of x and y – in other words, it is the same expression as we had before.

Now apply this expression to derive the vector of estimated coefficients, $\widehat{\mathbf{w}}$ to the dataset above. First, you will need to recall the definition of the inverse of a 2×2 matrix, available at many places on the web, e.g., `http://mathworld.wolfram.com/MatrixInverse.html`.

For a 2×2 matrix (why is $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ a 2×2 matrix?) it is possible (and possibly instructive) to calculate out the matrix operations by hand, including the inversion, but you will find it easier to put the x and y data into a matrix and vector representation and do the calculation in NumPy (or some alternative such as Matlab).

You should, of course, find the same values using both methods, and this example is sufficiently simple to intuitively see what the coefficients should be.