Mathematics for Machine Learning (70015)

Lecture 1.1 - Linear Regression Intro

Linear regression aims to provide a solution to the supervised learning problem; we are given a dataset of N examples of inputs and expected outputs, with a goal of predicting the correct output for a new input. Examples of this include image classification (such as digit classification) and translation. A curve fitting problem in 1-dimension has an input space $\in \mathbb{R}$, and an output space $\in \mathbb{R}$.

To tackle this problem mathematically, we need to first describe the curve fitting problem mathematically. As each input is associated with a single output, this is equivalent to a function in mathematics. We are given a dataset of N pairs, of inputs and outputs, where $\boldsymbol{x_n} \in \mathcal{X}$, which is usually $\mathbb{R}^D$ and $y_n \in \mathcal{Y}$ (usually $\mathbb{R}$ in this case); $\{(\boldsymbol{x_n}, y_n)\}_{n=1}^N$. The goal is to find a function that maps from the input space to the output space **well**; $f : \mathcal{X} \to \mathcal{Y}$.

We need to first find candidates for functions that can perform the predictions. Functions need to be parameterised, such that some numbers $\boldsymbol{\theta}$ map to a function. From here, we need to pick the 'best' function, thus requiring us to define what good and bad functions are. A good function has the property $f(\boldsymbol{x_i}, \boldsymbol{\theta^*}) \approx y_i$; the output of the function closely matches the outputs of the training points. This can be defined with a **loss function**, for example;

$$L(\boldsymbol{\theta}) = \sum_{i=1}^{N} (y_i - f(\boldsymbol{x_i}, \boldsymbol{\theta}))^2$$

Therefore, a good function is chosen by minimising the loss; $\boldsymbol{\theta^*} = \text{argmin}_{\boldsymbol{\theta}} L(\boldsymbol{\theta})$.

Lecture 1.2 - Scalar Differentiation

We can plot the loss against the parameters for a function. This raises two questions; how to change the parameter to make the loss smaller and how we know if we can't get a better loss. The derivative is defined as the limit of the difference quotient (as usual);

$$f'(x) = \frac{\mathrm{d}f}{\mathrm{d}x} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

Several examples of this are as follows;

$f(x)$	$f'(x)$
x^n	nx^{n-1}
$\sin(x)$	$\cos(x)$
$\tanh(x)$	$1 - \tanh^2(x)$
$e^x = \exp(x)$	e^x
$\log(x)$	$\frac{1}{x}$

There are also the following rules which combine the basic functions;

- sum rule — describes the derivative of sum of two functions

$$(f(x) + g(x))' = f'(x) + g'(x)$$

- product rule — similarly, for multiplication

$$(f(x)g(x))' = f'(x)g(x) + f(x)g'(x)$$

- chain rule describes how to differentiate functions that are composed

$$(g \circ f)'(x) = (g(f(x)))' = g'(f(x))f'(x)$$

- quotient rule describes division, special case of the product rule

$$\left(\frac{f(x)}{g(x)}\right)' = \frac{f'(x)g(x) - f(x)g'(x)}{(g(x))^2}$$

This tells us how to change the input in the function; the gradient tells us how much the output changes based on an increase in the input. We first compute the derivative function at a point to find the point's gradient. If the gradient is negative, this tells us the function will decrease if we increase the input (hence increase to minimise), and vice versa; decrease for positive gradients - this is the idea behind gradient descent.

Similarly, we know we are done (at a minimum for the loss function) when there is nothing that can be done to lower the output, hence the gradient must be 0. However, this isn't sufficient to tell us that we have reached a minimum, as a maximum also has a gradient of zero. A minimum has a decreasing function followed by an increasing function; hence the gradient of the gradient (second derivative) is positive.

However, this only gives us a local minima. We should be concerned with getting stuck in a local minima (rather than a global minima) when dealing with non-convex functions. Working through a simple example, with a linear regression problem (aiming to find an optimal a) - the final step takes the second derivative to verify we have a minimum;

$$\begin{aligned}
f(x) &= a \cdot x \\
L(a) &= \sum_{n=1}^{N} (f(x_n) - y_n)^2 \\
\frac{\mathrm{d}L}{\mathrm{d}a} &= \sum_{n=1}^{N} 2(ax_n - y_n)x_n \\
&= \sum_{n=1}^{N} 2ax_n^2 - 2x_n y_n \\
&= 0 \\
2a \sum_n x_n^2 &= \sum_n 2x_n y_n \\
a &= \frac{\sum_n 2x_n y_n}{\sum_n x_n^2} \\
\frac{\mathrm{d}^2 L}{\mathrm{d}a^2} &= \sum_{n=1}^{N} 2x_n^2 \\
&\geq 0
\end{aligned}$$

Lecture 1.3 - Scalar-by-vector Differentiation

The previous example is too simple for real applications - we will need to differentiate by more parameters (vectors). Consider the following example, a polynomial, which has 4 vectors parametising it - each vector now corresponds to a single cubic polynomial;

$$f(x) = \theta_3 x^3 + \theta_2 x^2 + \theta_1 x + \theta_0$$

$$= \boldsymbol{\theta}^\top \boldsymbol{\phi}(x)$$
$$\boldsymbol{\phi}(x) = \begin{bmatrix} x^3 & x^2 & x & 1 \end{bmatrix}^\top$$

Our goal still remains to understand how a function changes with our parameter and to characterise what an optimum is for a function of a vector. Both of these change a multi-dimensional problem into many 1-dimensional problems.

We want to change it into a 1-dimensional question; instead of asking about a change to $\boldsymbol{\theta}$, we ask what happens when we move along a particular line / direction. A directional derivative is how much the function changes, when we move in a direction $\boldsymbol{v}$;

$$\nabla_{\boldsymbol{v}} L(\boldsymbol{\theta}) = \lim_{h \to 0} \frac{L(\boldsymbol{\theta} + h\boldsymbol{v}) - L(\boldsymbol{\theta})}{h}$$

The distance that we move away from the starting point ($\boldsymbol{\theta}$) is determined by **both** the norm / scale of the direction vector $\boldsymbol{v}$, as well as h. If we can understand how the function changes based on a change in **any** direction, we can fully characterise differentiation with respect to a vector.

Consider the following example, where we deal with two parameters (also notice the second equality holds as the values in violet are equal and cancel each other out);

$$\begin{aligned}
\nabla_{\boldsymbol{v}} L(\boldsymbol{\theta}) &= \lim_{h \to 0} \frac{L(\theta_1 + hv_1, \theta_2 + hv_2) - L(\theta_1, \theta_2)}{h} \\
&= \lim_{h \to 0} \underbrace{\frac{L(\theta_1 + hv_1, \theta_2 + hv_2) - L(\theta_1, \theta_2 + hv_2)}{h}}_{\text{only change in first parameter}} + \underbrace{\frac{L(\theta_1, \theta_2 + hv_2) - L(\theta_1, \theta_2)}{h}}_{\text{only change in second parameter}} \\
&= \lim_{h \to 0} \frac{L(\theta_1 + h', \theta_2 + h'\frac{v_2}{v_1}) - L(\theta_1, \theta_2 + h'\frac{v_2}{v_1})}{\frac{h'}{v_1}} + \frac{L(\theta_1, \theta_2 + h'') - L(\theta_1, \theta_2)}{\frac{h''}{v_2}} \\
&= \frac{\partial L}{\partial \theta_1} v_1 + \frac{\partial L}{\partial \theta_2} v_2
\end{aligned}$$

This means that we can find the gradient in any direction with the partial derivatives, as we chose arbitrary v_1, v_2. With a partial derivative, we change only one coordinate at a time - see the following for a function $f : \mathbb{R}^N \to \mathbb{R}$;

$$y = f(\boldsymbol{x})$$
$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix}$$
$$\frac{\partial f}{\partial x_i} = \lim_{h \to 0} \frac{f(x_1, \ldots, x_{i-1}, x_i + h, x_{i+1}, \ldots, x_N) - f(\boldsymbol{x})}{h}$$

The Jacobian vector collects all partial derivatives into a row vector;

$$\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}} = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \cdots & \frac{\partial f}{\partial x_N} \end{bmatrix} \in \mathbb{R}^{1 \times N}$$

We now want to know which direction to go in, to decrease the value of the function the most. The directional derivative can be written as the inner product;

$$\nabla_{\boldsymbol{v}} f(\boldsymbol{\theta}) = \frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{\theta}} \boldsymbol{v} = \left| \frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{\theta}} \right| |\boldsymbol{v}| \cos \beta$$

We can maximise this by making $\cos \beta$ as large as possible, since the norms of the two vectors are fixed. If we choose a unit vector $\boldsymbol{v}$, we want the largest value possible, hence $\cos \beta = 1$, so the angle between the vectors should be zero (such that $\beta = 0$). As such, we move in the direction of the Jacobian / gradient vector.

We can reuse the intuition from the 1-D case, where moving in either direction doesn't change your value - the directional derivative should be zero in **all directions** (hence the zero vector, $\mathbf{0}$). Additionally, to verify it's a minimum, we want the second directional derivative to also be positive in **all directions**.

Lecture 1.4 - Vector-by-vector Differentiation

Recall the motivating example of linear regression;

$$L(\boldsymbol{\theta}) = \sum_{n=1}^{N} (y_n - \boldsymbol{\phi}(x_n)^T \boldsymbol{\theta})^2 = ||\boldsymbol{y} - \boldsymbol{\Phi}(X)\boldsymbol{\theta}||^2$$

We could either manually take partial derivatives of L, which would be laborious, or consider it as a composition of a vector-to-vector function (the matrix multiplication) and a vector-to-scalar function (the norm of the vector squared);

$$f(\boldsymbol{g}(\boldsymbol{\theta})) \qquad f : \mathbb{R}^D \to \mathbb{R} \qquad g : \mathbb{R}^E \to \mathbb{R}^D$$

There is a multivariate chain rule, for scalars it is as follows (where f is a function of a and b, both of which are functions of t);

$$\frac{\mathrm{d}f(a(t), b(t))}{\mathrm{d}t} = \frac{\partial f}{\partial a}\frac{\mathrm{d}a}{\mathrm{d}t} + \frac{\partial f}{\partial b}\frac{\mathrm{d}b}{\mathrm{d}t}$$

This can be generalised as follows, for $\boldsymbol{g}(t) \in \mathbb{R}^D$, and both are vectors - allowing us to write it as an inner product;

$$\frac{\mathrm{d}f(\boldsymbol{g}(t))}{\mathrm{d}t} = \sum_{i=1}^{D} \frac{\partial f}{\partial g_i}\frac{\mathrm{d}g_i}{\mathrm{d}t} = \underbrace{\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{g}}}_{\text{row}} \cdot \underbrace{\frac{\mathrm{d}\boldsymbol{g}}{\mathrm{d}t}}_{\text{col}}$$

This only works as we've defined the differentiation of a function with respect to a vector as a row vector - which we will use for the remainder of the course. Similarly, the second part of the sum is the derivative of a column vector, which remains a column vector. Consider the following example, with $f : \mathbb{R}^2 \to \mathbb{R}$ and $\boldsymbol{x} : \mathbb{R} \to \mathbb{R}^2$;

$$\begin{aligned}
f(\boldsymbol{x}) &= f(x_1, x_2) \\
&= x_1^2 + 2x_2 \\
\boldsymbol{x}(t) &= \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \\
&= \begin{bmatrix} \sin(t) \\ \cos(t) \end{bmatrix} \\
\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}} &\in \mathbb{R}^{1\times 2} \\
\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} &\in \mathbb{R}^2 \\
\frac{\mathrm{d}f}{\mathrm{d}t} &= \frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}}\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} \\
&= \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix} \begin{bmatrix} \frac{\partial x_1}{\partial t} \\ \frac{\partial x_2}{\partial t} \end{bmatrix} \\
&= \begin{bmatrix} 2\sin(t) & 2 \end{bmatrix} \begin{bmatrix} \cos(t) \\ -\sin(t) \end{bmatrix} \\
&= 2\sin(t)(\cos(t) - 1)
\end{aligned}$$

A similar rule applies when differentiating with respect to a vector (note previously we only did it with respect to a scalar). Similarly, this just requires collecting all the partial derivatives, and the same chain rule applies for $\boldsymbol{g}(\boldsymbol{x}) \in \mathbb{R}^D$;

$$\frac{\partial f(\boldsymbol{g}(\boldsymbol{x}))}{\partial x_j} = \sum_{i=1}^{D} \frac{\partial f}{\partial g_i}\frac{\mathrm{d}g_i}{\mathrm{d}x_j}$$

Once collected, this becomes matrix multiplication, which can be written in vector form;

$$\frac{\mathrm{d}f(\boldsymbol{g}(\boldsymbol{x}))}{\mathrm{d}\boldsymbol{x}} = \underbrace{\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{g}}}_{\text{row}} \cdot \underbrace{\frac{\mathrm{d}\boldsymbol{g}}{\mathrm{d}\boldsymbol{x}}}_{\text{mat}}$$

Note that the matrix is the derivative of a column vector ($\boldsymbol{g}$) with respect to an input vector $\boldsymbol{x}$. We instead put the elements in each row (similar to how we did it for a vector by scalar) - the elements of $\boldsymbol{g}$ (i) are along the column, and the dimensions of the derivative (j) are along the row. If f gave a vector as an output, we'd end up with a matrix by matrix multiplication.

Consider the following example, where we have $f : \mathbb{R}^N \to \mathbb{R}^M$, $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^M$, and $\boldsymbol{x} \in \mathbb{R}^N$;

$$\begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix} = \begin{bmatrix} f_1(\boldsymbol{x}) \\ \vdots \\ f_M(\boldsymbol{x}) \end{bmatrix} = \begin{bmatrix} f_1(x_1, \ldots, x_N) \\ \vdots \\ f_M(x_1, \ldots, x_N) \end{bmatrix}$$

The collection of all partial derivatives is called a Jacobian matrix;

$$\begin{bmatrix} \frac{\mathrm{d}y_1}{\mathrm{d}\boldsymbol{x}} \\ \vdots \\ \frac{\mathrm{d}y_M}{\mathrm{d}\boldsymbol{x}} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \vdots & & \vdots \\ \frac{\partial f_M}{\partial x_1} & \cdots & \frac{\partial f_M}{\partial x_N} \end{bmatrix} \in \mathbb{R}^{M \times N}$$

In general, a function $\boldsymbol{f} : \mathbb{R}^N \to \mathbb{R}^M$ has a gradient that is a matrix $\mathbb{R}^{M \times N}$, or has the number of target dimensions (M) $\times$ the number of input dimensions (N);

$$\mathrm{d}\boldsymbol{f}[m, n] = \frac{\partial f_m}{\partial x_n}$$

Recall that for matrix multiplication, the second dimension of the first matrix must match with the first dimension of the second matrix. A function composition is constrained that the output dimension of $\boldsymbol{h}$ must be the same as the input dimension of $\boldsymbol{g}$ to compute $\boldsymbol{g}(\boldsymbol{h}(\boldsymbol{x}))$ when we have $\boldsymbol{f}(\boldsymbol{x}) = (\boldsymbol{g} \circ \boldsymbol{h})(\boldsymbol{x})$. This ensures the shapes of the chain rule will **always** work out. For $\boldsymbol{f} : \mathbb{R}^N \to \mathbb{R}^M$, $\boldsymbol{g} : \mathbb{R}^L \to \mathbb{R}^M$, and $\boldsymbol{h} : \mathbb{R}^N \to \mathbb{R}^L$, we have the following shapes;

$$\underbrace{\frac{\mathrm{d}\boldsymbol{f}}{\mathrm{d}\boldsymbol{x}}}_{M \times N} = \underbrace{\frac{\mathrm{d}\boldsymbol{g}}{\mathrm{d}\boldsymbol{h}}}_{M \times L} \underbrace{\frac{\mathrm{d}\boldsymbol{h}}{\mathrm{d}\boldsymbol{x}}}_{L \times N}$$

Consider the following example, of a matrix multiplication $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x}$ where we have $\boldsymbol{A} \in \mathbb{R}^{M \times N}$ and $\boldsymbol{x} \in \mathbb{R}^N$;

$$\begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix} = \begin{bmatrix} f_1(\boldsymbol{x}) \\ \vdots \\ f_M(\boldsymbol{x}) \end{bmatrix} \tag{1}$$

$$= \begin{bmatrix} A_{1,1}x_1 + A_{1,2}x_2 + \cdots + A_{1,N}x_N \\ \vdots \\ A_{M,1}x_1 + A_{M,2}x_2 + \cdots + A_{M,N}x_N \end{bmatrix} \tag{2}$$

$$f_i(\boldsymbol{x}) = \sum_{k=1}^{N} A_{i,k}x_k \tag{3}$$

$$\frac{\partial f_i}{\partial x_j} = \sum_k A_{i,k}\frac{\partial x_k}{\partial x_j} \tag{4}$$

$$= \sum_k A_{i,k}\delta_{k,j} \tag{5}$$

$$= A_{i,j} \tag{6}$$

Note the following steps in particular;

(3) write out each scalar in the vector with index notation by writing the sum explicitly allowing us to take the derivative of an arbitrary element in the output vector with respect to an arbitrary element in the input vector

(4) take differential operator inside by sum rule

(5) matrix value is constant, so we end up taking a partial derivative of x_k by x_j - partial derivative only changes x_j and keeps everything else constant, hence it will be zero when $k \neq j$ and 1 when $k = j$ (indicator function, δ)

(6) end up with $A_{i,j}$ as it's the only case the indicator function is non-zero

Therefore, we get the result that $\frac{\mathrm{d}f}{\mathrm{d}x} = \boldsymbol{A} \in \mathbb{R}^{M \times N}$.

Consider the following, which is similar to the loss function, where $\boldsymbol{x} \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{M \times N}, \boldsymbol{e}, \boldsymbol{y} \in \mathbb{R}^M$;

$$
\begin{aligned}
L(\boldsymbol{e}) &= \frac{1}{2}||\boldsymbol{e}||^2 \\
&= \frac{1}{2}\boldsymbol{e}^\top \boldsymbol{e} \\
\boldsymbol{e} &= \boldsymbol{y} - \boldsymbol{A}\boldsymbol{x} \\
\frac{\mathrm{d}L}{\mathrm{d}\boldsymbol{x}} &= \frac{\partial L}{\partial \boldsymbol{e}} \frac{\partial \boldsymbol{e}}{\partial \boldsymbol{x}} \\
\frac{\partial L}{\partial e_i} &= \frac{\partial}{\partial e_i} \sum_j \frac{1}{2} e_j^2 \\
&= \sum_j \frac{1}{2} 2 e_j \frac{\partial e_j}{\partial e_i} \\
&= e_i \\
\frac{\partial L}{\partial \boldsymbol{e}} &= \boldsymbol{e}^\top \\
\frac{\partial \boldsymbol{e}}{\partial \boldsymbol{x}} &= -\boldsymbol{A} \\
\frac{\partial L}{\partial \boldsymbol{x}} &= e^\top (-\boldsymbol{A}) \\
&= -(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x})^\top \boldsymbol{A}
\end{aligned}
$$

Lecture 1.5 - Hessians

We still want to get the second derivative, in order to verify that we have a minima and not a maxima. The chain rule we have in place only works with derivatives of scalars, or column vectors (when differentiating with respect to vectors). We've currently defined the vertical axis to be outputs and the horizontal axis for variables we're differentiating by. This convention no longer holds when we are taking second derivatives, we we need to take the second derivative along a row vector;

$$
\nabla_{\boldsymbol{v}} \underbrace{\left[\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{\theta}}\boldsymbol{v}\right]}_{\text{scalar}} = \underbrace{\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\theta}}\left[\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{\theta}}\boldsymbol{v}\right]}_{\text{row vector}} \boldsymbol{v}
$$

Solving this in a way that only involves taking derivatives with respect to scalars;

$$
\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{\theta}}\boldsymbol{v} = \sum_j \frac{\partial f}{\partial \theta_j} v_j \tag{1}
$$

$$
\frac{\partial}{\partial \theta_i}\left[\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{\theta}}\boldsymbol{v}\right] = \sum_j \frac{\partial}{\partial \theta_j}\frac{\partial f}{\partial \theta_j} v_j \tag{2}
$$

$$= \sum_j \frac{\partial^2 f}{\partial \theta_i \partial \theta_j} v_j \tag{3}$$

$$\nabla_{\boldsymbol{v}} \left[\frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{\theta}} \boldsymbol{v} \right] = \boldsymbol{v}^\top \boldsymbol{H} \boldsymbol{v} \tag{4}$$

This involves the following steps;

1. expand out in terms of summation components (partial derivatives multiplied by vector component)
2. take partial derivative of the scalar, using sum rule
3. obtain a row vector where we multiply against the second partial derivatives
4. the matrix $\boldsymbol{H}$ (Hessian) contains all second partial derivatives of the function f

We are at the minimum if $\boldsymbol{H}$ is positive definite ($\forall \boldsymbol{v}\ \boldsymbol{v}^\top \boldsymbol{H} \boldsymbol{v} \geq 0$) - positive eigenvalues.