MATH 462 LECTURE NOTES: FEATURE REGRESSION

ADAM M. OBERMAN

The formulas above looks complicated at first glance. However, there is a geometrical interpretation in terms of projection. https://en.wikipedia.org/wiki/Projection_matrix

1. Vector dataset notation

Here we want to emphasize the fact that feature vectors are functions of the data.

So let x be a datapoint in an abstract data domain $\mathcal{X}$. In particular, we do not think of X as being a vectors space, since, for generic data, we do not have a notion of $x_1 + cx_2$.

We write a dataset

$$S_m = \{(x_1, y_1), \dots, (x_m, y_m)\}$$

in matrix form as

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}, \quad X \in \mathcal{X}^{m \times 1}$$

and

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}, \quad Y \in \mathbb{R}^{m \times 1}$$

However, we assume that we have vector features, $f : \mathcal{X} \to \mathbb{R}^d$ written as a column vector,

$$f(x) = \begin{bmatrix} f^1(x) \\ \vdots \\ f^d(x) \end{bmatrix}$$

so that

$$f(x)^\top = \left[f^1(x), \dots, f^d(x)\right], \quad f(x) \in \mathbb{R}^{d \times 1}$$

is a row vector.

Note: since x is a column vector, the function $f(x) = x$ is also column vector, so we can think of *a vector-valued function as a vector of functions.*

Date: November 14, 2023.

1.1. **Data Matrix.** We are going to define, following the convention in [DFO20, Chapter 9] *Convention. Note, if data/features are a column vector, then a matrix of data/features needs to be* $m \times d$

Define

$$F=f(X)^{\top}=\left[\begin{array}{c} f(x_1)^{\top} \\ \vdots \\ f(x_m)^{\top} \end{array}\right]=\left[\begin{array}{ccc} f^1(x_1) & \cdots & f^d(x_1) \\ \vdots & & \vdots \\ f^1(x_m) & \ldots & f^d(x_m) \end{array}\right], \quad F \in \mathbb{R}^{m \times d}$$

Thus we also have

$$F^{\top}=f(X)=\left[\begin{array}{ccc} f^1(x_1) & \cdots & f^1(x_m) \\ \vdots & & \vdots \\ f^d(x_1) & \ldots & f^d(x_m) \end{array}\right], \quad F \in \mathbb{R}^{d \times m}$$

Let $w \in \mathbb{R}^d$ be a column vector,

$$w=\left[\begin{array}{c} w_1 \\ \vdots \\ \vdots \\ w_d \end{array}\right]$$

Then, we can write,

$$h(x)=f(x)^{\top} w$$

Linear functions of the features

$$\mathcal{H}=\{h: x \rightarrow \mathbb{R} \mid h(x)=f(x)^{\top} w, w \in \mathbb{R}^d\}$$

Write, in vector notation

$$\begin{aligned} H &= h(X)=F w \\ E &= H-Y \end{aligned}$$

The mean squared loss

$$L(h, S)=\frac{1}{m} \sum_{i=1}^m (h(x_i)-y_i)^2$$

Can be written in vector notation as

$$L(h, S)=E^{\top} E=\|H-Y\|_2^2=\|F w-Y\|_2^2$$

Using the linear functions, we express the loss as a function of w,

$$L(w)=\frac{1}{m} \sum_{i=1}^m (f(x_i) \cdot w-y_i)^2 \tag{1}$$

1.2. **Gradients and Jacobians.** *For a scalar function* $h(x)$, *we make* $\nabla_x h$ *a row vector. This way the jacobian of a vector valued function* f *is a matrix, where each row is* $\nabla_x f^i$

Definition 1.1 (Jacobian)**.** The collection of all first-order partial derivatives of a vector-valued function $\boldsymbol{f}: \mathbb{R}^n \rightarrow \mathbb{R}^d$ is called the Jacobian. The Jacobian $\boldsymbol{J}$ is an $d \times n$ matrix, which we define and arrange as follows:

$$\boldsymbol{J} = \nabla_{\boldsymbol{x}} \boldsymbol{f} = \frac{\mathrm{d}\boldsymbol{f}(\boldsymbol{x})}{\mathrm{d}\boldsymbol{x}} = \begin{bmatrix} \frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial x_1} & \cdots & \frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial x_n} \end{bmatrix}$$

$$\boldsymbol{J} = \begin{bmatrix} \frac{\partial f_1(\boldsymbol{x})}{\partial x_1} & \cdots & \frac{\partial f_1(\boldsymbol{x})}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_d(\boldsymbol{x})}{\partial x_1} & \cdots & \frac{\partial f_d(\boldsymbol{x})}{\partial x_n} \end{bmatrix},$$

where

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix},$$

so that

$$J(i,j) = \frac{\partial f_i}{\partial x_j}.$$

As a special case of (5.58), a function $f : \mathbb{R}^n \to \mathbb{R}^1$, which maps a vector $\boldsymbol{x} \in \mathbb{R}^n$ onto a scalar (e.g., $f(\boldsymbol{x}) = \sum_{i=1}^n a_i x_i$), possesses a Jacobian that is a row vector (matrix of dimension $1 \times n$).

Example 1.2. Let $f(x) = a^\top x = x^\top a$ be linear. Note, x is a column vector, and a is a column vector. Then $\nabla_x f(x) = a^\top$ is a row vector. However, we define $\nabla_a f(x) = x$ to be a *column* vector, to be consistent with the notion that a vector of functions is a column vector. Thus

$$\begin{aligned} f(x) &= a^\top x && \text{scalar function} \\ \nabla_x f(x) &= a^\top && \text{row vector (gradient in } x) \\ \nabla_a f(x) &= x && \text{column vector (vector of functions)} \end{aligned}$$

Example 1.3. Now when w is the variable, and x is a parameter, let $L(w,x) = (w^\top x - y)^2/2$. Then

$$\begin{aligned} L(w,x) &= (w^\top x - y)^2/2 && \text{scalar function of } w \\ \nabla_w L &= (w^\top x - y)x^\top && \text{row vector (gradient in } w) \\ \nabla_x L &= (w^\top x - y)w && \text{column vector (vector of functions of } w) \end{aligned}$$

1.3. **Feature regression minimizer.** Going back to the vector notation for the loss, (1),

$$L(w) = \frac{1}{m} \sum_{i=1}^m \left(f(x_i) \cdot w - y_i \right)^2$$

Taking a derivative

$$\frac{\partial L}{\partial w_j}(w) = \frac{1}{m} \sum_{i=1}^m \left(f(x_i) \cdot w - y_i \right) \left(2 f^j \left(x_i \right) \right)$$

Now for the gradient (which is a *row* vector),

$$\nabla_w L\left(w\right) = \frac{2}{m} \sum_{i=1}^m (f(x_i) \cdot w - y_i) f(x_i)^\top$$

in vector notation, this is

$$\frac{2}{m}E^\top F^\top = \frac{2}{m}F^\top(Fw - Y)$$

Thus, in vector notation

$$\nabla_w L(w) = \frac{2}{m}F^\top(Fw - Y)$$

So the mimizer satisfies the normal equation

$$F^\top F w = F^\top Y \tag{2}$$

Which means

$$w = (F^\top F)^{-1} F^\top y$$

Then the function values are

$$h = Fw = F(F^\top F)^{-1} F^\top y$$

and for new function values,

$$h(x) = f(x) \cdot w = f(x) \cdot (F^\top F)^{-1} F^\top y$$

2. Functional notation

2.1. **Inner product of functions.** Let H be a vector space of functions, with an inner product $(f, g) = (f, g)_H$.

We are now going to write the regression problem as a projection problem in function space. Given $y \in H$, and given functions $f_1, \dots f_d$, let

$$V = \operatorname{span}\{f_1, \dots f_d\}$$

Let

$$h = Proj_V(y) = \arg\min_{f \in V} \|f - y\|_H^2$$

Since h is the projection, for each basis element $f_i \in V$, we have

$$(h, f_i) = (y, f_i)$$

Write $h = \sum w_j f_j$. Then

$$\left(\sum w_i f_i, f_j\right) = (y, f_j), \quad \forall j$$

or

$$\sum_i w_i (f_i, f_j) = (y, f_j)$$

Define $M_{ij} = (f', f^j)$ and $b_j = (y, f_j)$. Then equation becomes

$$Mw = b$$

This makes more intuitive sense to me than the vector way.

If, instead, we find an orthonormal basis of V, say, $e_1, \dots, e_d$, then the equations become

$$\left(\sum w_i e_i, e_j\right) = (y, e_j), \quad \forall j$$

or

$$w_i = (y, e_i), \quad \forall i$$

Leading to the projection

$$h = \sum_i (y, e_i) e_i$$

2.2. **Analysis of the minimizer.** Define the vector space $V = V(X)$ to be m-dimensional vectors, regarded as functions

$$V = \{f : X \to \mathbb{R} \mid f_i = f(x_i)\}$$

Define an inner product on V by

$$(h, g)_X = \frac{1}{m} \sum_{i=1}^{m} g(x_i)h(x_i)$$

Then the normal equation can be interpreted as follows

$$(F^\top F)_{ij} = (f^i, f^j)_X, \quad (F^\top Y)_j = (y, f^j)_X$$

and

$$h(X) = w \cdot f(X)$$

is the projection of Y onto the span of F.

we also have the following result.

The error is orthogonal to the solution

Theorem 2.1. *Let Y be given as above. Let $H = Fw$ be the solution of the normal equation. Let $E = H - Y$ be the error. Then the error is orthogonal to the solution,*

$$(E, H)_X = 0$$

Proof. From the normal equation, (2)

$$F^\top (Fw - Y) = 0$$

multiply on the right by $w^\top$, to obtain

$$w^\top F^\top (Fw - Y) = 0$$

rewrite this as

$$(H, E) = 0$$

as desired. □

References

[DFO20] Marc Peter Deisenroth, A Aldo Faisal, and Cheng Soon Ong. *Mathematics for machine learning*. Cambridge University Press, 2020.