Linear Regression Normal Equation – Additional Results

Pawel Wocjan

January 14th, 2019

Abstract

We derive the normal equation for linear regression and show that the mean-squared-error is a convex function.

1 Notation

Let $(\boldsymbol{x}^{(1)}, y^{(1)}), \ldots, (\boldsymbol{x}^{(m)}, y^{(m)}) \in \mathbb{R}^n \times \mathbb{R}$ denote the collection of training examples. The ith training example consists of the feature vector $\boldsymbol{x}^{(i)} = (x_1^{(i)}, \ldots, x_n^{(i)})^T \in \mathbb{R}^n$ and the label $y^{(i)} \in \mathbb{R}$.

Set

$$\boldsymbol{X} = \begin{pmatrix} x_1^{(1)} & \ldots & x_n^{(1)} \\ x^{(2)} & \ldots & x_n^{(2)} \\ \vdots & \ddots & \vdots \\ x^{(m)} & \ldots & x_n^{(m)} \end{pmatrix} \in \mathbb{R}^{m \times n}$$

and

$$\boldsymbol{y} = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{pmatrix} \in \mathbb{R}^m.$$

$\boldsymbol{X}$ is called the design matrix. Its rows correspond to the feature vectors of the training examples.

2 Normal equation

To simplify the discussion, consider first the case that the bias of the linear regression model is set to 0, that is, only the weight vector $\boldsymbol{w} = (w_1, \ldots, w_n)^T \in \mathbb{R}^n$ needs to be determined.

Theorem 1 (Normal equation)**.** *The optimal weight vector* $\boldsymbol{w} = (w_1, \ldots, w_n)^T \in \mathbb{R}^n$*, that is, the one that minimizes the mean squared error is given by the formula*

$$\boldsymbol{w} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{y}.$$

This is proved in 5.1.4 Example: Linear Regression in [1]. I have included this proof with additional results so you can understand every step of the proof.

3 Additional results

We introduce some abbreviations. Let $[n] = \{1, \ldots, n\}$. Let ∂w_r denote the partial derivative operator

$$\frac{\partial}{\partial w_r}.$$

Lemma 1 (Gradient of quadratic form)**.** *Let* $\boldsymbol{A} = (a_{rs}) \in \mathbb{R}^{n \times n}$ *be an arbitrary symmetric matrix and* $\boldsymbol{w} = (w_1, \ldots, w_n)^T \in \mathbb{R}^n$ *an arbitrary column vector. Define the quadratic form* $f(\boldsymbol{w}) = \boldsymbol{w}^T \boldsymbol{A} \boldsymbol{w}$*. Its gradient is given by*

$$\nabla_{\boldsymbol{w}} f(\boldsymbol{w}) = 2\boldsymbol{A}\boldsymbol{w}\,.$$

Proof. The right hand side is the column vector whose entries are given by

$$2 \sum_{s=1}^{n} a_{rs} w_s.$$

for $r \in [n]$. This follows simply by carrying out the matrix-vector-multiplication.

The left hand side of the above equation is the column vector whose entries are the partial derivatives

$$\partial w_r f(\boldsymbol{w})$$

for $r \in [n]$. This follows from the definition of the nable operator

$$\nabla_{\boldsymbol{w}} = \begin{pmatrix} \partial w_1 \\ \vdots \\ \partial w_n \end{pmatrix}.$$

We have

$$\begin{aligned}
\partial w_r f(\boldsymbol{w}) &= \partial w_r \left(\sum_{t,s=1} w_t a_{ts} w_s \right) \\
&= \partial w_r \left(w_r^2 a_{rr} + 2 \sum_{s \neq r} w_r a_{rs} w_s \right) \\
&= 2 w_r a_{rr} + 2 \sum_{s \neq r} a_{rs} w_s \\
&= 2 \sum_{s=1}^{n} a_{rs} w_s.
\end{aligned}$$

We use that either $t = r$ and $s = r$ or $t = r$ and $s \neq r$. Otherwise the partial derivative $\partial w_r(w_t a_{ts} w_s)$ is equal to 0. We also use that $\boldsymbol{A}$ is symmetric, that is, $a_{rs} = a_{sr}$.

Lemma 2. *Let $\boldsymbol{w} = (w_1, \ldots, w_n)^T \in \mathbb{R}^n$ and $\boldsymbol{v} = (v_1, \ldots, v_n)^T \in \mathbb{R}^n$ be an arbitrary column vectors. Define the function $g(\boldsymbol{w}) = \boldsymbol{w}^T \boldsymbol{v}$. Its gradient is given by*

$$\nabla_{\boldsymbol{w}} g(\boldsymbol{w}) = \boldsymbol{v}\,.$$

Proof. This is easy. Prove it yourself.

4 Proof of normal equation

Let $y^{(i)} = \boldsymbol{w}^T \boldsymbol{x}^{(i)}$ denote the prediction of the linear regression model with weight vector $\boldsymbol{w}$ when given the ith feature vector $\boldsymbol{x}^{(i)}$.[1] Define

$$\hat{\boldsymbol{y}} = \begin{pmatrix} \hat{y}^{(1)} \\ \vdots \\ \hat{y}^{(n)} \end{pmatrix}$$

The mean squared error (MSE) is given by

$$\begin{aligned} \text{MSE} &= \frac{1}{m}\sum_{i=1}^{m}(\hat{y}^{(i)} - y^{(i)})^2 \\ &= \frac{1}{m}\|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_2^2, \end{aligned}$$

so the error increases whenever the Euclidean distance between the predictions and the targets (labels) increases.

To minimize MSE, we can simply solve for where its gradient is $\mathbf{0}$:

$$\begin{aligned} \nabla_{\boldsymbol{w}}\text{MSE} &= \mathbf{0} & (1)\\ \Rightarrow \nabla_{\boldsymbol{w}}\frac{1}{m}\|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_2^2 &= \mathbf{0} & (2)\\ \Rightarrow \frac{1}{m}\nabla_{\boldsymbol{w}}\|\boldsymbol{X}\boldsymbol{w} - \hat{\boldsymbol{y}}\|_2^2 &= \mathbf{0} & (3)\\ \Rightarrow \frac{1}{m}\nabla_{\boldsymbol{w}}(\boldsymbol{X}\boldsymbol{w} - \hat{\boldsymbol{y}})^T(\boldsymbol{X}\boldsymbol{w} - \hat{\boldsymbol{y}}) &= \mathbf{0} & (4)\\ \Rightarrow \frac{1}{m}\nabla_{\boldsymbol{w}}(\boldsymbol{w}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} - 2\boldsymbol{w}^T\boldsymbol{X}^T\boldsymbol{y} + \boldsymbol{y}^T\boldsymbol{y}) &= \mathbf{0} & (5)\\ \Rightarrow 2\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} - 2\boldsymbol{X}^T\boldsymbol{y} &= \mathbf{0} & (6)\\ \Rightarrow \boldsymbol{w} = (\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w})^{-1}\boldsymbol{X}^T\boldsymbol{y} & & (7) \end{aligned}$$

[1] Note that $\hat{y}^{(i)}$ is the inner product between the weight vector $\boldsymbol{w}$ and the feature vector $\boldsymbol{x}^{(i)}$. We also have $\hat{y}^{(i)} = (\boldsymbol{x}^{(i)})^T\boldsymbol{w}$ because the inner product is symmetric.

In eq. (5), we can use Lemma 1 with $\boldsymbol{A} = \boldsymbol{X}^T\boldsymbol{X}$ and Lemma 2 with $\boldsymbol{v} = \boldsymbol{X}^T\boldsymbol{y}$ to compute the gradient. The system of equations whose solutions is given by eq. 7 is known as normal equations.

The general case of linear regression with non-zero bias b can also be solved with the help of the normal equation. Define the augmented weight vector $\boldsymbol{w}_b = (b, w_1, \ldots, w_n)^T \in \mathbb{R}^{n+1}$ and the augmented feature vectors $\boldsymbol{x}_b^{(i)} = (1, x_1^{(i)}, \ldots, x_n^{(i)})^T \in \mathbb{R}^{n+1}$. We have

$$\hat{y}^{(i)} = \boldsymbol{w}_b^T\boldsymbol{y}_b^{(i)} = \boldsymbol{w}^T\boldsymbol{y}^{(i)} + b\,.$$

5 Proof of convexity of mean-squared-error

We now show that MSE is convex in $\boldsymbol{w}$. We can write

$$\mathrm{MSE}(\boldsymbol{w}) = \phi(\|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|),$$

where $\phi : [0, \infty) \to \mathbb{R}, \phi(x) = \frac{1}{m}x^2$ is non-decreasing and convex. Therefore, it suffices to show that $f(\boldsymbol{w}) = \|\boldsymbol{X}\boldsymbol{w} - \boldsymbol{y}\|$ is convex. This is seen by invoking the following simple lemma.

Lemma 3. *Suppose $\phi : [0, \infty) \to \mathbb{R}$ is non-decreasing and convex and $f : \mathbb{R}^n \to [0, \infty)$ is convex. For $p \in [0, 1]$ and $\boldsymbol{r}, \boldsymbol{s} \in \mathbb{R}^n$, we have*

$$\begin{aligned}
\phi(f(p\boldsymbol{r} + (1-p)\boldsymbol{s})) &\leq \phi(f(p\boldsymbol{r}) + (1-p)f(\boldsymbol{s})) \\
&\leq p\phi(f(\boldsymbol{r})) + (1-p)\phi(f(\boldsymbol{s})).
\end{aligned}$$

Hence, $\phi \circ f$ is convex.

Let $\boldsymbol{w}$ and $\tilde{\boldsymbol{w}}$ be two weight vectors. We have

$$\begin{aligned}
f(p\boldsymbol{w} + (1-p)\tilde{\boldsymbol{w}})) &= \|\boldsymbol{X}(p\boldsymbol{w} + (1-p)\tilde{\boldsymbol{w}}) - \boldsymbol{y}\| \\
&\leq \|p(X\boldsymbol{w} - y) + (1-p)(\boldsymbol{X}\tilde{\boldsymbol{w}} - \boldsymbol{y})\| \\
&\leq p\|X\boldsymbol{w} - y\| + (1-p)\|\boldsymbol{X}\tilde{\boldsymbol{w}} - \boldsymbol{y}\| \\
&= pf(\boldsymbol{w}) + (1-p)f(\tilde{\boldsymbol{w}}).
\end{aligned}$$

In the above derivation, we have used the triangle inequality $\|\boldsymbol{r} + \boldsymbol{s}\| \leq \|\boldsymbol{r}\| + \|\boldsymbol{s}\|$ and $\|\lambda\boldsymbol{r}\| = |\lambda|\|\boldsymbol{r}\|$, which hold for arbitrary $\lambda \in \mathbb{R}$ and $\boldsymbol{r}, \boldsymbol{s} \in \mathbb{R}^n$.

We see that the mean-squared-error is a convex function by combining all results.

References

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, MIT Press, 2006, `http://www.deeplearningbook.org`