CS-453 Machine Learning

Nazar Khan

Department of Computer Science
University of the Punjab

3. Linear Regression

Regression

- We study the problem of *regression*.
  - Predict *continuous* target variable(s) t given input variables vector x.
- Given training data $\{(\mathrm{x}_1, t_1), \ldots, (\mathrm{x}_N, t_N)\}$, learn a function $y(\mathrm{x}, \mathrm{w})$ that maps the inputs to the targets.
- Regression corresponds to finding the optimal parameters w^*.

Linear Regression

- The simplest regression model is *linear regression*.
- Linear in parameters w and linear in inputs x.

$$y(\mathrm{x}, \mathrm{w}) = \mathrm{w}^T \mathrm{x} = w_0 + w_1 x_1 + \cdots + w_D x_D$$

- Parameter w_0 accounts for a fixed offset in the data and is called the *bias* parameter.
- To incorporate bias, we have increased the dimensionality of x from D to $D+1$ by appending a 1 before it.
- This makes our input vector $\mathrm{x} \in \mathbb{R}^{D+1}$ and parameter vector $\mathrm{w} \in \mathbb{R}^{D+1}$.

Linear Regression

- Linear models are significantly limited for practical problems – especially for high dimensional inputs.
- However, they have nice analytical properties and they form the foundation for more sophisticated machine learning approaches.

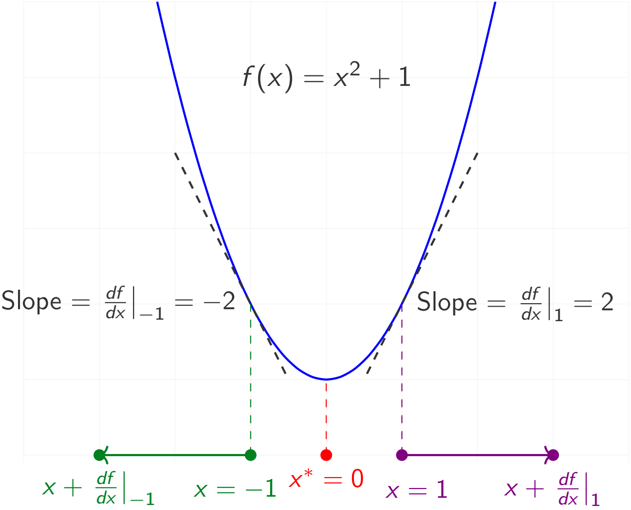
Linear Regression

- A more powerful model is linear in parameters w but non-linear in inputs x.

$$y(\mathrm{x}, \mathrm{w}) = \mathrm{w}^T \boldsymbol{\phi}(\mathrm{x}) = w_0\phi_0(\mathrm{x}) + w_1\phi_1(\mathrm{x}) + \cdots + w_M\phi_M(\mathrm{x})$$

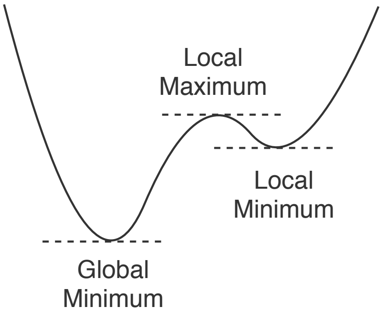
- $\phi_0(\mathrm{x})$ is usually set to 1 to make w_0 the bias parameter.
- Note that now $\mathrm{w} \in \mathbb{R}^{M+1}$ where M is not necessarily equal to D.
- The input x-space is non-linearly mapped to ϕ-space and learning takes place in this new ϕ-space.
- While the learning remains linear, the learned mapping is actually non-linear in x-space.

Minimization



What is the slope/derivative/gradient at the minimizer $x^* = 0$?

Minimization
Local vs. Global Minima



- *Stationary point*: where derivative is 0.
- A stationary point can be a minimum or a maximum.
- A minimum can be local or global. Same for maximum.

Linear Regression

- Error function of a regression model is

$$E(\mathrm{w}) = \frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathrm{w}^T\phi(\mathrm{x_n})\}^2$$

- Derivative with respect to w is

$$\frac{d}{d\mathrm{w}}E(\mathrm{w}) = \sum_{n=1}^{N}\{t_n - \mathrm{w}^T\phi(\mathrm{x_n})\}\phi(\mathrm{x_n})^T$$

- At the minimiser w^*, the gradient must be equal to 0

$$\left.\frac{d}{d\mathrm{w}}E(\mathrm{w})\right|_{\mathrm{w}^*} = 0$$

Linear Regression

- Equating gradient to the 0 vector

$$\sum_{n=1}^{N} t_n \phi(\mathrm{x_n})^T - \mathrm{w}^{*T} \left(\sum_{n=1}^{N} \phi(\mathrm{x_n}) \phi(\mathrm{x_n})^T \right) = 0 \quad (1)$$

$$\implies \mathrm{w}^{*T} = \left(\sum_{n=1}^{N} t_n \phi(\mathrm{x_n})^T \right) \left(\sum_{n=1}^{N} \phi(\mathrm{x_n}) \phi(\mathrm{x_n})^T \right)^{-1}$$

Linear Regression

- To convert to a pure matrix-vector notation without summations, let us define the following $N \times M$ matrix

$$\boldsymbol{\Phi} = \begin{bmatrix} \phi_0(x_1) & \phi_1(x_1) & \cdots & \phi_{M-1}(x_1) \\ \phi_0(x_2) & \phi_1(x_2) & \cdots & \phi_{M-1}(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_N) & \phi_1(x_N) & \cdots & \phi_{M-1}(x_N) \end{bmatrix}$$

  known as the *design matrix*.

Linear Regression

- It can be verified that the second term in Equation (1) $\sum_{n=1}^{N} \phi(\mathrm{x_n})\phi(\mathrm{x_n})^T = \mathbf{\Phi}^T\mathbf{\Phi}$. **(Verify this.)**
- By placing the target values in a vector $\mathbf{t} = (t_1, \ldots, t_N)^T$ we can also write the first term as $\mathbf{\Phi}^T\mathbf{t}$. **(Verify this.)**
- Now we can solve for the optimal weights as

$$\mathrm{w}^* = \underbrace{(\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T}_{\mathbf{\Phi}^\dagger}\mathbf{t}$$

- The $M \times N$ matrix $\mathbf{\Phi}^\dagger$ is known as the *Moore-Penrose pseudo-inverse* or simply *pseudo-inverse* of matrix $\mathbf{\Phi}$.
- It is a generalisation of matrix inverse to non-square matrices.
- For a square, invertible matrix $\mathbf{\Phi}$, it can be verified that $\mathbf{\Phi}^\dagger = \mathbf{\Phi}^{-1}$. **(Verify this.)**

Linear Regression
Regularisation

- Error function for regularised linear regression is

$$E(\mathrm{w}) = \frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathrm{w}^T\phi(\mathrm{x_n})\}^2 + \frac{\lambda}{2}\|\mathrm{w}\|^2$$

  where λ is the *regularisation coefficient* that controls the trade-off between fitting and regularisation.
- This is also known as *regularised least squares*.
- Such regularisation is also called *weight decay* or *parameter shrinkage* because it encourages weight/parameter values to remain close to 0.
- Regularisation allows more complex models to be trained on small datasets without severe over-fitting.
- However, parameter λ needs to be set appropriately.

Linear Regression
Regularised

- Optimal solution to regularised linear regression is

$$\mathsf{w}^* = (\lambda \mathsf{I} + \mathbf{\Phi}^T \mathbf{\Phi})^{-1} \mathbf{\Phi}^T \mathbf{t}$$

Linear Regression
Multivariate targets

- For the case of multivariate target vectors $\mathbf{t}_n \in \mathbb{R}^K$, we are interested in the multivariate mapping $\mathbf{y}(\mathbf{x}, \mathbf{W}) = \mathbf{W}^T \boldsymbol{\Phi}(\mathbf{x})$.
- Column k of the $M \times K$ matrix $\mathbf{W}$ determines the mapping from $\phi(\mathbf{x})$ to the k_{th} output component.
- The optimal solution given training data $\{\mathbf{x}_n, \mathbf{t}_n\}_{n=1}^N$ can be computed as

$$\mathbf{W}^* = \boldsymbol{\Phi}^\dagger \mathbf{T}$$

where $\mathbf{T} = \begin{bmatrix} \mathbf{t}_1^T \\ \vdots \\ \mathbf{t}_N^T \end{bmatrix}$ is the $N \times K$ matrix of target vectors.