# Linear Regression

## Useful Formulas

- Pseudoinverse of matrix $\Phi$: $\Phi_p = \left(\Phi^T \Phi\right)^{-1} \Phi^T$

## Questions

1. What is the role of basis functions in linear regression?

   Basis functions allow us to represent a non-linear function of the input variables with a function which is linear in the weights.

2. Can an algorithm doing linear regression learn only linear functions of the inputs?

   No, the learned function is linear in the weights but does not need to be linear in the input variables.

3. When can we solve the linear regression problem exactly (with 0 error)? Why is it not a good idea to do so?

   When the number of parameters is the same as the number of points in the data set. Normally, we want much fewer parameters than data points.

4. What is the error we want to minimize when doing linear regression?

   The sum of squares error: $E(\boldsymbol{X}) = \frac{1}{2}\sum_{i=1}^{N}\left(\boldsymbol{\Phi}_i^T \boldsymbol{w} - t_i\right)^2$ , where **X** is the dataset, $\boldsymbol{\Phi}_i$ is the vector of the basis functions evaluated on point i, $t_i$ is the desired value for point i (target), and **w** is the vector of weights to optimise.

5. What is the least-squares solution? How is it affected by outliers?

   The least-squares solution uses the pseudo-inverse of the matrix **Φ** of the basis functions evaluated on the data set, and is defined as: $w = \left(\boldsymbol{\Phi}^T \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^T \boldsymbol{t}$ where $\boldsymbol{\Phi}_p = \left(\boldsymbol{\Phi}^T \boldsymbol{\Phi}\right)^{-1} \boldsymbol{\Phi}^T$ is the pseudo-inverse of **Φ** (the full derivation is in the slides). Since the least-squares solution minimises the average error on all the points, outliers affect the average strongly by pushing it towards their value.

6. How can we find the least-squares solution when there are too many points to compute the pseudoinverse efficiently?

   We can perform stochastic gradient descent on the error point by point, updating the current weight vector according to: $\boldsymbol{w}_{k+1}=\boldsymbol{w}_k-\eta\nabla E_i=\boldsymbol{w}_k-\eta(\boldsymbol{\Phi}_i^T\boldsymbol{w}_k-t_i)\boldsymbol{\Phi}_n$ (full derivation in the slides).

7. What are the bias and the variance for a supervised learning problem?

The bias is an error of the regression (or the classifier) which on average converges towards something away from the desired value. The variance is the dependency of the model on the data set, so that with different training sets we get different regressed functions (or classifiers). The variations of the different functions (or classifiers) is captured by the variance.

8. What is the link between the error on the validation set increasing with training, and the bias/variance decomposition?

The bias/variance decomposition shows us that the expected error has three components: the bias, the variance, and the noise in the data. The noise is an intrinsic property of the data set and training does nothing about it. On the other hand, training decreases the bias, making the average estimate increasingly correct. The total expected error, however, does not change, therefore the reduction of the bias has to happen at the expense of something else: the variance. Therefore, the model becomes increasingly dependent on the particular data points used for training (which increases the variance) and loses generalization.

9. Given the dataset: <-1, -0.5>, <0,1.1>, <1,3.8>, <2,8.8>, find the least-squares solution for the function: $y(x,\boldsymbol{w})=w_0+w_1x$

First, we need to create the matrix of the coefficients for the linear system. The first column of the matrix is the value of the first basis function on the points. The first basis function, that is what is multiplied by $w_0$ is the constant 1. The second basis function is the function x:

$$\boldsymbol{\Phi}=\begin{bmatrix}1 & -1\\ 1 & 0\\ 1 & 1\\ 1 & 2\end{bmatrix}$$

Then, we need to compute the pseudo inverse of $\boldsymbol{\Phi}$ :

$$\boldsymbol{\Phi}^T\boldsymbol{\Phi}=\begin{bmatrix}1 & 1 & 1 & 1\\ -1 & 0 & 1 & 2\end{bmatrix}\begin{bmatrix}1 & -1\\ 1 & 0\\ 1 & 1\\ 1 & 2\end{bmatrix}=\begin{bmatrix}4 & 2\\ 2 & 6\end{bmatrix}$$

$$(\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}=\begin{bmatrix}0.3 & -0.1\\ -0.1 & 0.2\end{bmatrix}$$

and lastly:

$$\boldsymbol{\Phi}_p=\begin{bmatrix}0.4 & 0.3 & 0.2 & 0.1\\ -0.3 & -0.1 & 0.1 & 0.3\end{bmatrix}\;.$$

We can now use the psudo inverse to compute the optimal vector of weights:

$$\boldsymbol{w}=\boldsymbol{\Phi}_p\boldsymbol{t}=\begin{bmatrix}0.4 & 0.3 & 0.2 & 0.1\\ -0.3 & -0.1 & 0.1 & 0.3\end{bmatrix}\begin{bmatrix}-0.5\\ 1.1\\ 3.8\\ 8.8\end{bmatrix}=\begin{bmatrix}1.77\\ 3.06\end{bmatrix},$$

where the vector t is the vector of the values of the function over the points in the dataset (the last element of each vector in the dataset).

10. Given the dataset: <-1, 0.78>, <0,1>, <1,1.22>, <2,1.52>, find the least-squares solution for

$$y(x,\boldsymbol{w})=w_0+w_1 e^{\frac{(x+1)^2}{20}}$$

All the steps are illustrated before, here I will just compute the final vectors for your reference:

$$\boldsymbol{\Phi}=\begin{bmatrix}1 & 1\\ 1 & 1.05\\ 1 & 1.22\\ 1 & 1.57\end{bmatrix}$$

$$\boldsymbol{w}=\boldsymbol{\Phi}_p\boldsymbol{t}=\begin{bmatrix}1.52 & 1.22 & 0.19 & -1.93\\ -1.05 & -0.80 & 0.05 & 1.81\end{bmatrix}\begin{bmatrix}0.78\\ 1\\ 1.22\\ 1.52\end{bmatrix}=\begin{bmatrix}-0.30\\ 1.18\end{bmatrix}$$

11. Given the dataset: <-1, 1.6>, <0,0.95>, <1,1.2>, <2,1.9>, find the least-squares solution for the function: $y(x,\boldsymbol{w})=w_0+w_1\frac{1}{1+e^{-(x+1)}}$

$$\boldsymbol{\Phi}=\begin{bmatrix}1 & 0.5\\ 1 & 0.73\\ 1 & 0.88\\ 1 & 0.95\end{bmatrix}$$

$$\boldsymbol{w}=\boldsymbol{\Phi}_p\boldsymbol{t}=\begin{bmatrix}1.96 & 0.48 & -0.49 & -0.94\\ -2.23 & -0.29 & 0.97 & 1.56\end{bmatrix}\begin{bmatrix}1.6\\ 0.95\\ 1.2\\ 1.9\end{bmatrix}=\begin{bmatrix}1.22\\ 0.28\end{bmatrix}$$