

Machine Learning Exercise Sheet 04

Linear Regression

Exercise sheets consist of two parts: homework and in-class exercises. You solve the homework exercises on your own or with your registered group and upload it to Moodle for a possible grade bonus. The in-class exercises will be solved and explained during the tutorial. You do not have to upload any solutions of the in-class exercises.

In-class Exercises

Problem 1: Assume that we are given a dataset, where each sample x_i and regression target y_i is generated according to the following process

$$x_i \sim \text{Uniform}(-10, 10)$$

$$y_i = ax_i^3 + bx_i^2 + cx_i + d + \epsilon_i, \quad \text{where} \quad \epsilon_i \sim \mathcal{N}(0, 1) \quad \text{and} \quad a, b, c, d \in \mathbb{R}.$$

The 3 regression algorithms below are applied to the given data. Your task is to say what the bias and variance of these models are (low or high). Provide a 1-2 sentence explanation to each of your answers.

a) Linear regression

Bias: high. Variance: low.

A straight line cannot capture a degree 3 polynomial (thus underfitting the data).

b) Polynomial regression with degree 3

Bias: low. Variance: low.

The model is same as the data generating process. We can achieve a good fit.

c) Polynomial regression with degree 10

Bias: low. Variance: high.

Since we are using a polynomial regression with a degree much higher compared to the data generating process, the model will overfit the data.

Problem 2: Given is a training set consisting of samples $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)^T$ with respective regression targets $\mathbf{y} = (y_1, y_2, \dots, y_N)^T$ where $\mathbf{x}_i \in \mathbb{R}^D$ and $y_i \in \mathbb{R}$.

Upload a single PDF file with your homework solution to Moodle by 17.11.2021, 23:59 CET. We recommend to typeset your solution (using L^AT_EX or Word), but handwritten solutions are also accepted. If your handwritten solution is illegible, it won't be graded and you waive your right to dispute that.

Alice fits a linear regression model $f(\mathbf{x}_i) = \mathbf{w}^T \mathbf{x}_i$ to the dataset using the closed form solution for linear regression (normal equations).

Bob has heard that by transforming the inputs \mathbf{x}_i with a vector-valued function Φ , he can fit an alternative function, $g(\mathbf{x}_i) = \mathbf{v}^T \Phi(\mathbf{x}_i)$, using the same procedure (solving the normal equations). He decides to use a linear transformation $\Phi(\mathbf{x}_i) = \mathbf{A}^T \mathbf{x}_i$, where $\mathbf{A} \in \mathbb{R}^{D \times D}$ has full rank.

- a) Show that Bob's procedure will fit the same function as Alice's original procedure, that is $f(\mathbf{x}) = g(\mathbf{x})$ for all $\mathbf{x} \in \mathbb{R}^D$ (given that \mathbf{w} and \mathbf{v} minimize the training set error).

Alice uses the normal equation directly and obtains

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

Bob fits the model to the transformed data and obtains

$$\begin{aligned} \mathbf{v}^* &= ((\mathbf{X}\mathbf{A})^T (\mathbf{X}\mathbf{A}))^{-1} (\mathbf{X}\mathbf{A})^T \mathbf{y} \\ &= (\mathbf{A}^T \mathbf{X}^T \mathbf{X} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{X}^T \mathbf{y} & (\dagger) \\ &= \mathbf{A}^{-1} (\mathbf{X}^T \mathbf{X})^{-1} (\mathbf{A}^T)^{-1} \mathbf{A}^T \mathbf{X}^T \mathbf{y} & (\ddagger) \\ &= \mathbf{A}^{-1} (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} = \mathbf{A}^{-1} \mathbf{w}^*. \end{aligned}$$

(Note that Φ transforms the column vectors \mathbf{x}_i via $\mathbf{A}^T \mathbf{x}_i$ but \mathbf{X} contains the transposed observations \mathbf{x}_i^T as rows. Therefore the transformed feature matrix is $\mathbf{X}\mathbf{A}$)

Now it is immediate to see that

$$g(\mathbf{x}_i) = \mathbf{v}^{*T} \Phi(\mathbf{x}_i) = \mathbf{w}^{*T} \mathbf{A}^{-T} \mathbf{A}^T \mathbf{x}_i = \mathbf{w}^{*T} \mathbf{x}_i = f(\mathbf{x}_i).$$

- b) Can Bob's procedure lead to a lower training set error than Alice's if the matrix \mathbf{A} is not invertible? Explain your answer.

Any weights \mathbf{v}^* Bob finds are also feasible for Alice by letting $\mathbf{w} = \mathbf{A}\mathbf{v}^*$. Therefore Bob can only access a subset of the parameter space and cannot achieve a lower loss value than Alice. (It could still be equal but it cannot be better.)

Note that we are only talking about training error in this example, not test error. Bob might manage to find a model that generalizes better than Alice's, but Alice will always be able to fit the training data at least as well as Bob.

Problem 3: See Jupyter notebook `inclass_04_notebook.ipynb`.