National University of Singapore
School of Computing
CS3244: Machine Learning
Week 05 - Tutorial - 04

Bias, Variance, and Overfitting

Colab Notebook:[1] Bias, Variance, and Overfitting

1. **Overfitting**: Briefly answer the following questions:

   (a) When (the number of data points / noise / target complexity) increases, is overfitting less likely to occur?

   (b) Assume $\mathcal{H}$ is fixed and we increase the complexity of f. Will deterministic noise in general (but not necessarily in all cases) go up or down? How about stochastic noise? Is there a higher or lower tendency to overfit?

   (c) Assume f is fixed and we increase the complexity of $\mathcal{H}$. Will deterministic noise in general (but not necessarily in all cases) go up or down? How about stochastic noise? Is there a higher or lower tendency to overfit?

2. **Validation**: We will be using a strategy called Cross-Validation(CV) to measure the performance(This will be discussed in detail in the week 06 lecture). k-fold CV can be described as follows. The training dataset is divided into k groups. Then, there would be k number of training iterations and validation performance measures. In each iteration, $k-1$ number of groups of data is used to train the model and the remaining group is used to measure the validation performance. In every iteration, the validation group is different. Finally, the final performance is the average over k validation performance measures of each iteration.

   You are deciding on a regularization parameter $\mathcal{H}$ for your linear regression model. You perform 10-fold CV on your training data for the following values of $\mathcal{C}$ and get the following graph (Figure 1):

   (a) What does each blue and green point represent? How are they calculated?

   (b) What do the blue and green shaded areas represent? How are they calculated?

   (c) What should you select as your $\mathcal{C}$ value?

3. **Bias and Variance**: Assume $y = f(x) + \epsilon$ where $\mathrm{E}(\epsilon) = 0$ and $\mathrm{Var}(\epsilon) = \sigma^2$. Using squared-error loss, the expected prediction error of a regression fit $\hat{f}(x)$ at an input point $x = x_0$ can be written as the sum of Irreducible Error, Bias^2 and Variance. For the k-nearest regression fit, the error can be expressed as:

[1] The notebooks are shared as additional exercises. They will be neither tested in exams nor discussed during the tutorial. However, the solutions will be provided.

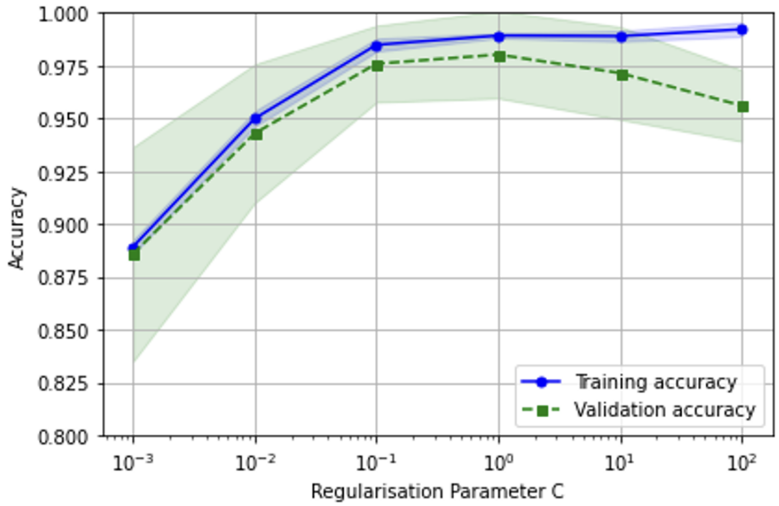

Figure 1: Validation Curve: (Source: Python Machine Learning by Sebastian Raschka)

$$
\begin{aligned}
Err(x_0) &= \mathrm{E}[(y - \hat{f}_k(x_0))^2 | x_0)] \\
&= \sigma^2 + (f(x_0) - \frac{1}{k}\sum_{i=1}^{k} f(x_i))^2 + \frac{\sigma^2}{k}.
\end{aligned}
$$

where $x_i (i = 1, ..., k)$ are the k nearest data points. We assume that these neighbors are fixed, for simplicity.

(a) Derive the above equation by calculating bias and variance.

(b) Describe how you would chose an optimal value of K using the above equations.

(c) In the training set, each sample follows the assumption of $y_i = f(x_i) + \epsilon_i$. Choose all the right choices about this assumption:

- A. ϵ_i comes from a Gaussian distribution
- B. all ϵ_i have the same mean
- C. all y_i have the same mean
- D. all y_i have the same variance

4. [**[2]] **Support Vector Machines** The figure 2 shows two hyper-planes $\mathcal{H}_1$ and $\mathcal{H}_2$. The red points have $y_i = 1$ and the blue points have $y_i = -1$. The blue vector is a hyperplane passing through the origin and has the from $\theta^\top x + b = 0$. $\vec{\theta}$ is normal to the plane. Show that the distance between the two hyperplanes is $d = \frac{2}{||\theta||}$ where $||\cdot||$ denotes the argument's norm.

[2]This is offered as an optional exercise to have a better theoretical understanding.

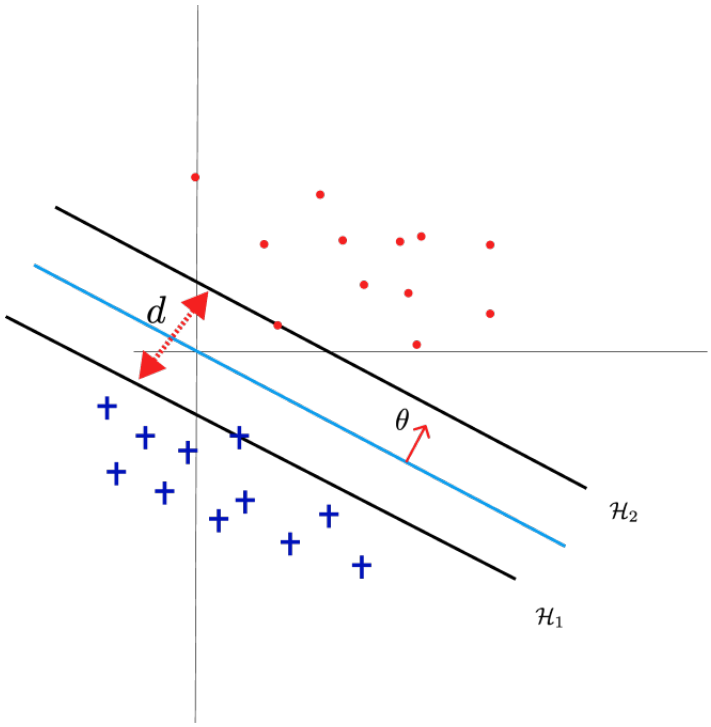

Figure 2: Support Vector Machines