Bias Variance Decomposition

Decomposing the generalization error

Mostly based on the following resources:

- Lecture by Kilian Weinberger
- This blog-entry with a good notation

Preliminaries

- P is a joint probability distribution with the training data pairs $(x_i, y_i) \sim P(X, Y)$, $x_i \in \mathcal{X}, y_i \in \mathcal{Y}, i = 1, \ldots, N$ and the training dataset $\mathcal{D} = \{(x_i, y_i) | i = 1, \ldots, N\}$.
- $\mathcal{A}$ an algorithm (e.g.:Decision Trees,Random Forest,Neural Networks etc).
- $h_{\mathcal{D}}$ is an classifier/regressor that we did get by training $\mathcal{A}$ on dataset $\mathcal{D}$.
- The expected label $\bar{y}$ of x is given by $\bar{y}(x) = \mathbb{E}_{y|x}[y] = \int\limits_y y \, P(y|x) \, \mathrm{d}y$.

Supervised learning is usually concerned with the the conditional estimation $P(Y = y | X = x)$.

- **Classification**: Given $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y} = \mathbb{R}^p \times \{1, \ldots, C\}$, for $i = 1, \ldots, N$ and C classes,

  for any new x, we want to estimate

  $\underset{y}{\operatorname{argmax}} \, P(Y = y | X = x)$.

- **Regression**: Given $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y} = \mathbb{R}^p \times \{1, \ldots, C\}$, for $i = 1, \ldots, N$,

  for any new x we want to estimate

  $\mathbb{E}[Y | X = x]$.

The expected test error of a classifier/regressor

Intuitively, to calculate the expected test error of a classifier $h_{\mathcal{D}}$, we have to repeat the following process a large number of times:

1. we take a couple (x, y) from the distribution P (for sure we have to choose (x, y) that don't belong to the dataset $\mathcal{D}$).
2. we calculate the squared error in this case .
3. we record the result obtained.

At the end, all we need to do is to calculate the average of the results that we recorded before, formally :

$\mathbb{E}_{(x,y) \sim P}([h_{\mathcal{D}}(x) - y)^2])$

For more precision, the expected test error of a classifier could be given as follow:

$$\mathbb{E}_{(x,y)\sim P}([h_{\mathcal{D}}(x)-y)^2]) = \int\limits_x \int\limits_y [h_{\mathcal{D}}(x)-y]^2 p(x,y)\,\mathrm{d}x\,\mathrm{d}y$$

The expected classifier/regressor

As we said before,the classifier $h_{\mathcal{D}}$ is a result of combination between two components:

1. The algorithm of classification $\mathcal{A}$.
2. The dataset $\mathcal{D}$ that we used to feet this algorithm.

$$h_{\mathcal{D}} = \mathcal{A}(\mathcal{D})$$

We can see that we could build many classifiers $(h_1, h_2, h_3, \dots)$ using different datasets $(\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3, \dots)$ who are formed from the same distribution P, so we define the expected classifier as follow:

$$\bar{h} = \mathbb{E}_{\mathcal{D}\sim P^n}[\mathcal{A}(\mathcal{D})] = \int\limits_{\mathcal{D}} h_{\mathcal{D}}\; p(\mathcal{D})\,\mathrm{d}\mathcal{D}$$

The output of the expected classifier when we give him the vector x as input would be calculated with the formula:

$$\bar{h}(x) = \int\limits_{\mathcal{D}} h_{\mathcal{D}}(x)\; p(\mathcal{D})\,\mathrm{d}\mathcal{D}$$

The expected error of the algorithm $\mathcal{A}$

In a simple way, we can calculate this value by repeating this process a large number of times:

1. we build a dataset $\mathcal{D}$ from the distribution P.
2. we get the classifier $h_{\mathcal{D}}$ by training the algorithm $\mathcal{A}$ using the dataset $\mathcal{D}$.
3. we use the process described before to calculate the expected test error for this particular classifier.
4. we record the result obtained.

At the end of this process, all we need to do is to calculate the average of the results that we recorded before and finally we can say that we did get value wanted,formally:

$$\mathbb{E}_{\substack{(x,y)\sim P\\ \mathcal{D}\sim P^n}}[(h_{\mathcal{D}}(x)-y)^2] = \int\limits_{\mathcal{D}} \int\limits_y \int\limits_x [h_{\mathcal{D}}(x)-y]^2\; p(x,y)\; p(\mathcal{D})\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}\mathcal{D}$$

Simplifying $\mathbb{E}_{\substack{(x,y)\sim P\\ \mathcal{D}\sim P^n}}$ to $\mathbb{E}_{\mathbf{x},y,D}$ we get:

$$\begin{aligned}\mathbb{E}_{\mathbf{x},y,D}\left[[h_D(\mathbf{x})-y]^2\right] &= \mathbb{E}_{\mathbf{x},y,D}[[\underbrace{(h_D(\mathbf{x})-\bar{h}(\mathbf{x}))}_{a}+\underbrace{(\bar{h}(\mathbf{x})-y)}_{b}]^2] \\ &= \mathbb{E}_{\mathbf{x},D}[\underbrace{(h_D(\mathbf{x})-\bar{h}(\mathbf{x}))^2}_{a^2}] + \mathbb{E}_{\mathbf{x},y,D}[\underbrace{2\,(h_D(\mathbf{x})-\bar{h}(\mathbf{x}))\,(\bar{h}(\mathbf{x})-y)}_{2ab}] + \mathbb{E}_{\mathbf{x},y}[\underbrace{(\bar{h}(\mathbf{x})-y)^2}_{b^2}]\end{aligned}$$

The middle term of the above equation is 0 as we show below

$$\begin{aligned}\mathbb{E}_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x})-\bar{h}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x})-y\right)\right] &= \mathbb{E}_{\mathbf{x},y}\left[\mathbb{E}_D\left[h_D(\mathbf{x})-\bar{h}(\mathbf{x})\right]\left(\bar{h}(\mathbf{x})-y\right)\right]\\ &= \mathbb{E}_{\mathbf{x},y}\left[\left(\mathbb{E}_D\left[h_D(\mathbf{x})\right]-\bar{h}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x})-y\right)\right]\\ &= \mathbb{E}_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-\bar{h}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x})-y\right)\right]\\ &= \mathbb{E}_{\mathbf{x},y}\left[0\right]\\ &= 0\end{aligned}$$

Returning to the earlier expression, we're left with the variance and another term

$$\mathbb{E}_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x})-y\right)^2\right] = \mathbb{E}_{\mathbf{x},D}\Big[\underbrace{\left(h_D(\mathbf{x})-\bar{h}(\mathbf{x})\right)^2}_{\text{Variance}}\Big] + \mathbb{E}_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-y\right)^2\right]$$

Recall from the preliminaries:

- The expected label $\bar{y}$ of x is given by $\bar{y}(x) = \mathbb{E}_{y|x}[y] = \int\limits_y y\,P(y|x)\,\mathrm{d}y$.

Now we can also break down the second term in the above equation as follows:

$$\begin{aligned}\mathbb{E}_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-y\right)^2\right] &= \mathbb{E}_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)+\left(\bar{y}(\mathbf{x})-y\right)^2\right]\\ &= \mathbb{E}_{\mathbf{x},y}\Big[\underbrace{\left(\bar{y}(\mathbf{x})-y\right)^2}_{\text{Noise}}\Big] + \mathbb{E}_{\mathbf{x}}\Big[\underbrace{\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)^2}_{\text{Bias}^2}\Big] + 2\,\mathbb{E}_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)\left(\bar{y}(\mathbf{x})-y\right)\right]\end{aligned}$$

The third term in the equation above is 0, as we show below

$$\begin{aligned}\mathbb{E}_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)\left(\bar{y}(\mathbf{x})-y\right)\right] &= \mathbb{E}_{\mathbf{x}}\left[\mathbb{E}_{y|\mathbf{x}}\left[\bar{y}(\mathbf{x})-y\right]\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)\right]\\ &= \mathbb{E}_{\mathbf{x}}\left[\left(\bar{y}(\mathbf{x})-\mathbb{E}_{y|\mathbf{x}}\left[y\right]\right)\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)\right]\\ &= \mathbb{E}_{\mathbf{x}}\left[\left(\bar{y}(\mathbf{x})-\bar{y}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)\right]\\ &= \mathbb{E}_{\mathbf{x}}\left[0\right]\\ &= 0\end{aligned}$$

This gives us the decomposition of expected test error as follows

$$\mathbb{E}_{\mathbf{x},y,D}\Big[\underbrace{\left(h_D(\mathbf{x})-y\right)^2}_{\text{Expected Test Error}}\Big] = \mathbb{E}_{\mathbf{x},D}\Big[\underbrace{\left(h_D(\mathbf{x})-\bar{h}(\mathbf{x})\right)^2}_{\text{Variance}}\Big] + \mathbb{E}_{\mathbf{x},y}\Big[\underbrace{\left(\bar{y}(\mathbf{x})-y\right)^2}_{\text{Noise}}\Big] + \mathbb{E}_{\mathbf{x}}\Big[\underbrace{\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)^2}_{\text{Bias}^2}\Big]$$

Variance: Captures how much your **classifier changes if you train on a different training set**. How "over-specialized" is your classifier to a particular training set (overfitting)? If we have the best possible model for our training data, how far off are we from the average classifier?

Bias: What is the **inherent error that you obtain from your classifier** even with infinite training data? This is due to your classifier being "biased" to a particular kind of solution (e.g. linear classifier). In other words, bias is inherent to your model.

Noise: How big is the **data-intrinsic noise**? This error measures ambiguity due to your data distribution and feature representation. You can never beat this, it is an aspect of the data.