Bias-variance decomposition

Linear models for regression

Francesco Corona

Bias-variance decomposition

Linear models for regression

Bias-variance decomposition

Maximum likelihood, or least squares, can lead to severe over-fitting

- if complex models are trained
  using data sets of limited size

Limiting the number of basis functions to avoid over-fitting has the side effect of limiting the flexibility of the model to capture interesting trends in the data

Regularisation terms can control over-fitting for models with many parameters

- How to determine a suitable value
  for the regularisation coefficient λ?

Bias-variance decomposition (cont.)

Remark

Seeking the solution that minimises the regularised error function with respect to both the weight vector $\mathbf{w}$ and the regularisation coefficient λ is clearly not the right approach since this leads to the unregularised solution with $\lambda = 0$

Bias-variance decomposition (cont.)

Over-fitting is an unfortunate property of maximum likelihood

- It does not arise when we marginalise over parameters in a Bayesian setting

It is instructive to first consider a frequentist viewpoint of model complexity

- **bias-variance trade-off**

We introduce the concept only in the context of linear basis function models

Bias-variance decomposition (cont.)

When we discussed decision theory for regression problems, the decision stage consists of choosing a specific estimate $y(\mathbf{x})$ of the target t for each input $\mathbf{x}$

We can do this using a loss $L\Big(t, y(\mathbf{x})\Big)$, so that the average/expected loss is

$$\mathbb{E}[L] = \int \int L\Big(t, y(\mathbf{x})\Big) p(\mathbf{x}, t) d\mathbf{x} dt$$

Various loss functions for regression lead to a corresponding optimal prediction

- once we are given the conditional density $p(t|\mathbf{x})$

Bias-variance decomposition (cont.)

A common loss function in regression problems is the **squared loss function**

$$L(t, y(\mathbf{x})) = \Big(y(\mathbf{x}) - t\Big)^2 \qquad \Longrightarrow \qquad \mathbb{E}[L] = \int\int \Big(y(\mathbf{x}) - t\Big)^2 p(\mathbf{x}, t) d\mathbf{x} dt$$

Squared loss function (decision theory) $\neq$ sum-of-squares error function (ML)

Squared loss function

$$L(t, y(\mathbf{x})) = \Big(y(\mathbf{x}) - t\Big)^2$$

- Optimal prediction $h(\mathbf{x})$ is given by conditional expectation $\mathbb{E}[t|\mathbf{x}]$

$$h(\mathbf{x}) = \mathbb{E}[t|\mathbf{x}] = \int t p(t|\mathbf{x}) dt \quad (1)$$

Bias-variance decomposition (cont.)

We also obtained: $\mathbb{E}[L] = \int \big(y(\mathbf{x}) - \mathbb{E}[t|\mathbf{x}]\big)^2 p(\mathbf{x}) d\mathbf{x} + \int \big(\mathbb{E}[t|\mathbf{x}] - t\big)^2 p(\mathbf{x}) d\mathbf{x}$

- It is minimised when $y(\mathbf{x})$, in the first term, equals $\mathbb{E}[t|\mathbf{x}]$

The second term is independent of $y(\mathbf{x})$, arises from the noise ε

- The variance of the distribution of t, averaged over $\mathbf{x}$
- It is the intrinsic variability of the target variable
- The minimum achievable value of the expected loss

Bias-variance decomposition (cont.)

The expected squared loss function can be written also in another form

$$\mathbb{E}[L] = \int \Big(y(\mathbf{x}) - h(\mathbf{x})\Big)^2 p(\mathbf{x}) d\mathbf{x} + \int \int \Big(h(\mathbf{x}) - t\Big)^2 p(\mathbf{x}, t) d\mathbf{x} dt \qquad (2)$$

With an infinite supply of data and unlimited computational resources

- we could find the regression function $h(\mathbf{x})$ to any accuracy

In practice, we only have a data set $\mathcal{D}$ with a finite number N of points

- $h(\mathbf{x})$ is not know exactly

Bias-variance decomposition (cont.)

If we model $h(\mathbf{x})$ using a parametric function $y(\mathbf{x}, \mathbf{w})$ with parameter vector $\mathbf{w}$

- the uncertainty in our model is expressed through a posterior distribution over $\mathbf{w}$ (Bayesian perspective)

A frequentist treatment makes a point estimate of $\mathbf{w}$ based on the data set $\mathcal{D}$

- the uncertainty of this estimate is expressed through a large number of data sets each of size N and each drawn independently from distribution $p(t, \mathbf{x})$

For any set $\mathcal{D}$, we learn our algorithm and get a prediction function $y(\mathbf{x}; \mathcal{D})$

- Different data sets, different functions
- Different functions, different values of the squared loss

The performance of a learning algorithm is assessed by averaging over sets

Bias-variance decomposition (cont.)

$$\mathbb{E}[L] = \int \Big(y(\mathbf{x}) - h(\mathbf{x})\Big)^2 p(\mathbf{x}) d\mathbf{x} + \int \int \Big(h(\mathbf{x}) - t\Big)^2 p(\mathbf{x}, t) d\mathbf{x} dt$$

Consider the integrand of first term of the expected squared loss, it becomes

$$\Big(y(\mathbf{x}; \mathcal{D}) - h(\mathbf{x})\Big)^2 \tag{3}$$

for a particular data set $\mathcal{D}$ and it has to be averaged over the ensemble of sets

Before taking its expectation wrt $\mathcal{D}$, add and subtract quantity $\mathbb{E}_{\mathcal{D}}[y(\mathbf{x}; \mathcal{D})]$

$$\Big(y(\mathbf{x}; \mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x}; \mathcal{D})] + \mathbb{E}_{\mathcal{D}}[y(\mathbf{x}; \mathcal{D})] - h(\mathbf{x})\Big)^2$$

Bias-variance decomposition (cont.)

Expanding, we obtain

$$\begin{aligned}\Big(y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] + \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\Big)^2 \\ = \Big(y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\Big)^2 + \Big(\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\Big)^2 \\ + 2\Big(y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\Big)\Big(\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\Big) \qquad (4)\end{aligned}$$

And taking the expectation with respect to $\mathcal{D}$, it gives

$$\mathbb{E}_{\mathcal{D}}\Big[\Big(y(\mathbf{x};\mathcal{D}) - h(\mathbf{x})\Big)^2\Big] = \Big(\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\Big)^2 + \mathbb{E}_{\mathcal{D}}\Big[\Big(y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\Big)^2\Big]$$

Bias-variance decomposition (cont.)

$$\mathbb{E}_{\mathcal{D}}\Big[\Big(y(\mathbf{x};\mathcal{D}) - h(\mathbf{x})\Big)^2\Big] = \underbrace{\Big(\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\Big)^2}_{(\text{bias})^2} + \underbrace{\mathbb{E}_{\mathcal{D}}\Big[\Big(y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\Big)^2\Big]}_{\text{variance}} \quad (5)$$

The expected squared difference between $y(\mathbf{x};\mathcal{D})$ and regression function $h(\mathbf{x})$ can be expressed as the sum of two terms

- The first term, squared **bias**, represents the extent to which the average prediction over all data sets differs from the desired regression function
- The second term, **variance**, measures the extent to which the solutions for individual data sets vary around their average, and hence measures the extent to which function $y(\mathbf{x};\mathcal{D})$ is sensitive to the particular data set

We shall provide some intuition to support these definitions

Bias-variance decomposition (cont.)

$$\mathbb{E}_{\mathcal{D}}\Big[\Big(y(\mathbf{x};\mathcal{D})-h(\mathbf{x})\Big)^2\Big] = \Big(\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]-h(\mathbf{x})\Big)^2+\mathbb{E}_{\mathcal{D}}\Big[\Big(y(\mathbf{x};\mathcal{D})-\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\Big)^2\Big]$$

Expected squared difference between $y(\mathbf{x};\mathcal{D})$ and the regression function $h(\mathbf{x})$

- when considering only a single input value $\mathbf{x}$

Substituting in $\mathbb{E}[L] = \int (y(\mathbf{x}) - h(\mathbf{x}))^2 p(\mathbf{x}) d\mathbf{x} + \int\int (h(\mathbf{x}) - t)^2 p(\mathbf{x}, t) d\mathbf{x} dt$

$$\text{expected loss} = (\text{BIAS})^2 + \text{VARIANCE} + \text{noise} \quad (6)$$

$$(\text{BIAS})^2 = \int \Big(\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\Big)^2 p(\mathbf{x}) d\mathbf{x} \quad (7)$$

$$\text{VARIANCE} = \int \mathbb{E}_{\mathcal{D}}\Big[\Big(y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\Big)^2\Big] p(\mathbf{x}) d\mathbf{x} \quad (8)$$

$$\text{noise} = \int\int \Big(h(\mathbf{x}) - t\Big)^2 p(\mathbf{x}, t) d\mathbf{x} dt \quad (9)$$

Bias-variance decomposition (cont.)

We decomposed the expected loss into (integrated) bias, (integrated) variance and a constant noise term, but our goal is the same: We want to minimise it

There is a trade-off between bias and variance:

- flexible models will have low bias and high variance
- rigid models will have high bias and low variance

$$(\text{BIAS})^2 = \int \Big(\mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})] - h(\mathbf{x})\Big)^2 p(\mathbf{x})d\mathbf{x}$$

$$\text{VARIANCE} = \int \mathbb{E}_{\mathcal{D}}\Big[\Big(y(\mathbf{x};\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(\mathbf{x};\mathcal{D})]\Big)^2\Big] p(\mathbf{x})d\mathbf{x}$$

The model with optimal predictive capability is the one with the best balance

Bias-variance decomposition (cont.)

Example

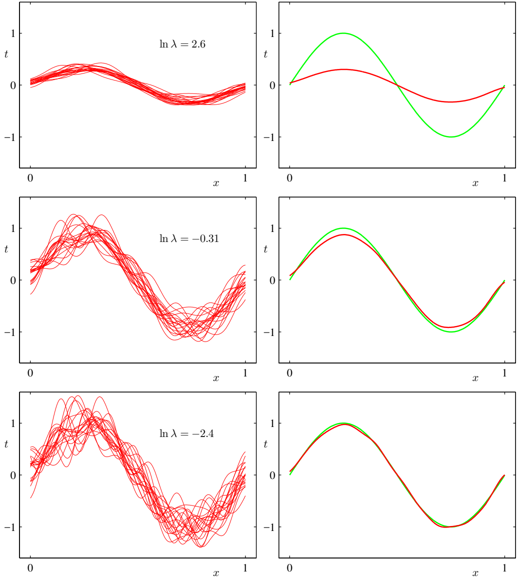
As an example, we consider the usual data from a sinusoidal function

- $l = 1, \dots, L$ datasets $\mathcal{D}^{(l)}$, each with $N = 25$ points, $L = 100$
- The points of each $\mathcal{D}^{(l)}$ are iid from $h(x) = \sin(2\pi x)$

For each $\mathcal{D}^{(l)}$, we fit a model with 24 Gaussian basis ($M = 25$ parameters)

- We minimised the regularised error $\dfrac{1}{2}\sum_{n=1}^{N}\Big(t_n - \mathbf{w}^T\phi(\mathbf{x}_n)\Big)^2 + \dfrac{\lambda}{2}\mathbf{w}^T\mathbf{w}$
- The resulting parameter vector is $\mathbf{w} = \Big(\lambda\mathbf{I} + \boldsymbol{\Phi}^T\boldsymbol{\Phi}\Big)^{-1}\boldsymbol{\Phi}^T\mathbf{t}$
- We use $\mathbf{w}^{(l)}$ to get a predictive function $y^{(l)}$

All this, for different values of the regularisation parameter λ



- Large λ (left), low variance but high bias
- Small λ (right), low bias but high variance

Bias-variance decomposition (cont.)

In this case, averaging many solutions turned out to be a beneficial procedure

$$\overline{y}(x) = \frac{1}{L}\sum_{l=1}^{L} y^{(l)}(x) \qquad \rightsquigarrow \mathbb{E}_{\mathcal{D}}[y(x;\mathcal{D})] \tag{10}$$

The integrated[1] squared bias and the integrated variance are given by

$$(\text{BIAS})^2 = \frac{1}{N}\sum_{n=1}^{N}\Big(\overline{y}(x_n) - h(x_n)\Big)^2$$
$$\rightsquigarrow \int \Big(\mathbb{E}_{\mathcal{D}}[y(x;\mathcal{D})] - h(x)\Big)^2 p(x)dx \tag{11}$$

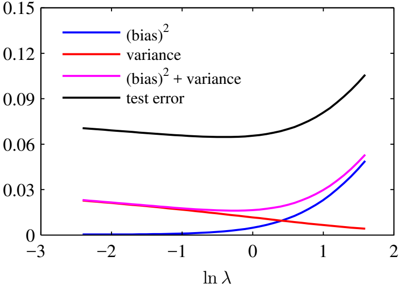
$$\text{VARIANCE} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{L}\sum_{l=1}^{L}\Big(y^{(l)}(x_n) - \overline{y}(x_n)\Big)^2$$
$$\rightsquigarrow \int \mathbb{E}_{\mathcal{D}}\Big[\Big(y(x;\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(x;\mathcal{D})]\Big)^2\Big] p(x)dx \tag{12}$$

[1] Integration over x weighted by the distribution $p(x)$ is approximated by a finite sum over points draw from that distribution

Bias-variance decomposition (cont.)

Plot of squared bias and variance, together with their sum

- Also shown is the average test set error for a test set size of 1000 points



Minimum $(BIAS)^2 + VARIANCE$ occurs around a value $\ln \lambda = -0.31$

It is close to the value that gives the minimum error on the test data