Assignment Project Exam Help

Predictive Analytics

Week 4: Model Selection and Estimation I

https://powcoder.com

Semester 2, 2018

Add WeChat powcoder

Discipline of Business Analytics, The University of Sydney Business School

Week 4: Model Selection

1. Model Selection and Evaluation
2. The bias-variance decomposition
3. KNN bias-variance decomposition
4. Cross validation
5. The bias-variance derivation details (optional)

Reading: Chapter 2.1, 2.2 of ISL.

Exercise questions: Chapter 2.4 of ISL, Q1, Q3, Q5, Q6 and Q7.(a). Only focus on “regression” if the questions contain “classification” content.

Model Selection and Evaluation

Model Selection and Evaluation

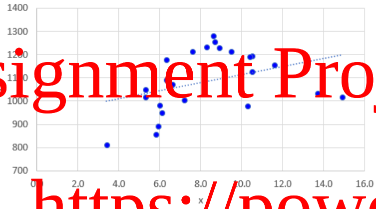
Model Selection: estimate the performance of different models in order to choose the (approximate) best one. We select the model that is estimated to have the best predictive ability.

Model Evaluation: after chosen the "different" model, estimate its test error (generalisation error) on new data.

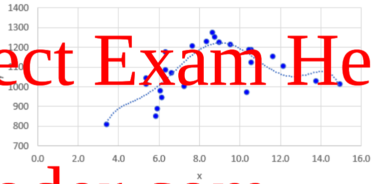
- **Training set**: for exploratory data analysis, model building, model estimation, etc.

- **Validation set**: for appropriate model selection.
- **Test set**: for model evaluation.

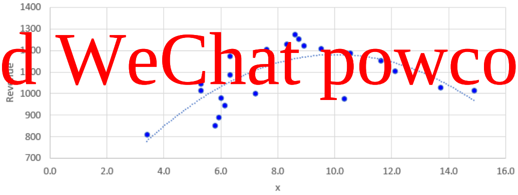
Underfitting and Overfitting



Underfitting: High bias, low variance



Overfitting: High variance, low bias



Best model

Why is overfitting bad?

- Low training error, high generalization error
- Poor predictive performance
- Overreacts to minor fluctuations in training data

How to address overfitting:

- Drop some features
  - Model selection algorithm
  - Manually select features to keep
- Regularization
  - Keep all features, reduce the magnitude/values of parameters. More details in later sections.

The bias-variance trade-off (key concept)

- Increasing model complexity brings higher flexibility and therefore lower bias. However, this comes at a cost of higher variance: there is higher sample variability when estimating complex models. Overfitting can be a problem.

- Decreasing model complexity leads to lower variance. However, simpler models may not be sufficiently flexible to capture the underlying patterns in the data, leading to higher bias. Underfitting can be a problem.

The bias-variance trade-off

To review, we use the training data to estimate the additive error model

$$Y = f(X) + \varepsilon$$

leading to an estimator $\widehat{f}(x_0)$ for the regression function at given input point x_0, $f(x_0)$.

Examples

Linear regression. Adding predictors increases model complexity. Least squares estimates have high variability when the number of inputs is large, but excluding relevant predictors leads to bias.

Review: Lecture 2

Examples

KNN regression. Reducing the number of neighbours increases model complexity. Closer neighbours means lower bias. However, averaging fewer observations increases variance.

Review: Lecture 3

Approaches to model selection

Train, validation and test sets. Validation set is used for selecting the optimal model complexity.

Resampling methods. Estimate generalisation performance by generating multiple splits of the training data.

Analytical criteria. Use analytical results to penalise training performance to account for overfitting.

Training, validation, and test split

In the **validation set** approach, we **randomly** split the training data into a training set, a validation set and a test set. We select the model with the best predictive performance in the validation set.

Typically, we use 50-80% of the data for the training set.

Training, validation, and test split

1. Estimate different models on the training data.
2. Predict the observations in the validation set.

3. Select the model with best validation set performance.
4. **Re-estimate** the selected model by combining the training and validation sets.

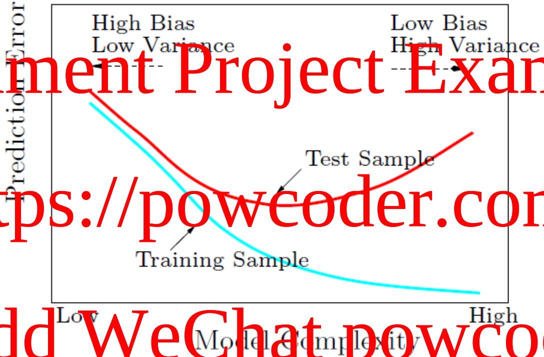
5. Predict the test data with the selected model.

Validation set

The validation set approach has serious limitations when the size of the training data is not large. The model may not have enough data to train on, and there may not be enough cases in the validation set to reliably estimate generalisation performance.

We turn instead to resampling methods.

Learning curve



We cannot use training set to select the best model. The model minimize the loss of this training data set, not necessarily minimize the loss of the new date sets.

Diagnosing learning curve

Suppose your training loss is low, while validation/test loss is high.

- Underfitting or overfitting problem?

Suppose your training loss is high, while validation/test loss is also high.

- Underfitting or overfitting problem?

Diagnosing learning curve

Underfitting: training error is high, validation error is **slightly** $>$ training error;

Overfitting: training error is low, validation error is **significantly** $>$ training error.

The bias-variance decomposition

Expected prediction error

- Consider again the additive error model:

$$Y = f(X) + \varepsilon,$$

where we assume that $E(\varepsilon) = 0$ and $\text{Var}(\varepsilon) = \sigma^2$.

- In the previous section, we treated $\widehat{f}(\cdot)$ as given since our objective was to estimate the test error. Now, we discuss the fundamental problem of choosing a method to learn a predictive function $\widehat{f}(\cdot)$.

Expected prediction error

We define the **expected prediction error** for a new input point $X = \boldsymbol{x}_0$ as

$$\text{Err}(\boldsymbol{x}_0) = E\left[\left(Y_0 - \widehat{f}(\boldsymbol{x}_0)\right)^2 | X = \boldsymbol{x}_0\right],$$

where $Y_0 = f(\boldsymbol{x}_0) + \varepsilon_0$. The expectation is over ε_0 and the training sample, i.e. over the sampling distribution of $\widehat{f}(\cdot)$. The EPE is a expected loss.

Note that this is different from the generalisation error, where $\widehat{f}(\boldsymbol{x}_0)$ is an estimate (not an estimator), and the expectation is over the population $P(X, Y)$.

Expected prediction error decomposition

We can write the expected prediction error as:

$$
\begin{aligned}
\text{Err}(\boldsymbol{x}_0) &= E\left[\left(Y_0 - \widehat{f}(\boldsymbol{x}_0)\right)^2 | X = \boldsymbol{x}_0\right] \\
&= E\left[\left(f(\boldsymbol{x}_0) + \varepsilon - \widehat{f}(\boldsymbol{x}_0)\right)^2 | X = \boldsymbol{x}_0\right] \\
&= \sigma^2 + E\left[\left(f(\boldsymbol{x}_0) - \widehat{f}(\boldsymbol{x}_0)\right)^2 | X = \boldsymbol{x}_0\right] \\
&= \text{Irreducible error} \ + \ \text{Reducible error}
\end{aligned}
$$

Expected prediction error decomposition

$$\text{Err}(\boldsymbol{x}_0) = \sigma^2 + E_{\mathcal{D}}\left[(f(\boldsymbol{x}_0) - \widehat{f}(\boldsymbol{x}_0))^2 | X = \boldsymbol{x}_0\right]$$

$$= \text{Irreducible error} + \text{Reducible error}$$

- The first term is the variance of the response around its true mean $f(x_0)$. We cannot avoid this source of error, and it puts an upper bound on the accuracy of the prediction.
- In choosing a method, our concern is the reducible error: we want to minimise the estimation error $E_{\mathcal{D}}\left[(f(\boldsymbol{x}_0) - \widehat{f}(\boldsymbol{x}_0))^2 | X = \boldsymbol{x}_0\right]$.
- Here, $E_{\mathcal{D}}(\cdot)$ is used to emphasise that the expectation is over the training data. This notation might be omitted later for simplicity purpose.

The bias-variance trade-off

We can show that (see last optional section of this slides):

$$
\begin{aligned}
E(f(\boldsymbol{x}_0) - \widehat{f}(\boldsymbol{x}_0))^2 &= \Big[E(\widehat{f}(\boldsymbol{x}_0)) - f(\boldsymbol{x}_0)\Big]^2 + E([\widehat{f}(\boldsymbol{x}_0) - E(\widehat{f}(\boldsymbol{x}_0))]^2) \\
&= \mathsf{Bias}^2(\widehat{f}(\boldsymbol{x}_0)) + \mathsf{Var}(\widehat{f}(\boldsymbol{x}_0)) \\
&= \mathsf{Bias}^2 + \mathsf{Variance}
\end{aligned}
$$

The bias-variance trade-off

$$E(f(\boldsymbol{x}_0) - \widehat{f}(\boldsymbol{x}_0))^2 = \mathsf{Bias}^2\left(\widehat{f}(\boldsymbol{x}_0)\right) + \mathsf{Var}\left(\widehat{f}(\boldsymbol{x}_0)\right)$$

- We would like our model to be flexible enough to be able to approximate (possibly) complex relationships between Y and X.

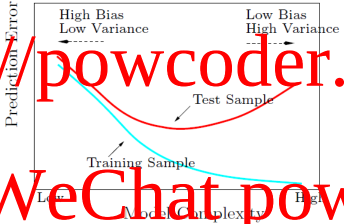
- Typically, the more complex we make the model, the better its approximation capabilities, which translates into lower bias.

- On the other hand, increasing model complexity leads to higher variance. This is due to the larger (effective) number of parameters to estimate.

- Hence, we would like to find the optimal (problem specific) model complexity that minimises our expected loss.

The bias-variance trade-off

Increasing model complexity will always reduce the training error, but there is an optimal level of complexity that minimises the test error.



How the plots will be looked like if we fix the model complexity and change x-axis from "model complexity" to "size of the data"?

Just because a model is more “realistic”, it does not mean that it will have higher predictive accuracy. All models are approximations, and our task is to find the most accurate one for our purposes in a data-driven way.

Model selection

- **Model selection** is a set of methods (such as cross validation) that allow us to choose the right model among options of different complexity. It will be a fundamental part of our methodology.

- Similarly to model evaluation, model selection methods are concerned with estimating the generalisation error. However, it is important not to confuse these two steps, which have different goals.

KNN bias-variance decomposition

Remember the following expression for the expected prediction error:

$$\begin{aligned}\mathrm{Err}(x_0) &= \mathrm{E}\left[\left(Y_0 - \widehat{f}(\boldsymbol{x}_0)\right)^2 \middle| X = \boldsymbol{x}_0\right] \\ &= \sigma^2 + \mathsf{Bias}^2\left(\widehat{f}(\boldsymbol{x}_0)\right) + \mathsf{Var}\left(\widehat{f}(\boldsymbol{x}_0)\right)\end{aligned}$$

The Bias-Variance Decomposition: kNN example

- For kNN regression: Suppose $\boldsymbol{x}_0$ is a test data point, its k nearest neighbours are $\{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_k\}$ with responses $\{y_1, y_2, ..., y_k\}$. Then the model prediction is

$$\widehat{f}(\boldsymbol{x}_0) = \frac{1}{k} \sum_{\boldsymbol{x}_i \in N_k(\boldsymbol{x}_0, \mathcal{D})} y_i.$$

- Then the test error

$$\begin{aligned}\text{Err}(\mathbf{x}_0) &= \sigma^2 + \text{Bias}^2(\widehat{f}(\boldsymbol{x}_0)) + \text{Var}(\widehat{f}(\boldsymbol{x}_0)) \\ &= \sigma^2 + \left[E(\widehat{f}(\boldsymbol{x}_0)) - f(\boldsymbol{x}_0)\right]^2 + E([\widehat{f}(\boldsymbol{x}_0) - E(\widehat{f}(\boldsymbol{x}_0))]^2)\end{aligned}$$

The Bias-Variance Decomposition: kNN example

- First we check, noting $y_l = f(\boldsymbol{x}_l) + \varepsilon_l$

$$E(\widehat{f}(\boldsymbol{x}_0)) = E\left[\frac{1}{k}\sum_{l=1}^{k} y_l\right] = \frac{1}{k}\sum_{l=1}^{k} E[f(\boldsymbol{x}_l) + \varepsilon_l]$$

$$= \frac{1}{k}\sum_{l=1}^{k} f(\boldsymbol{x}_l)$$

- For the third term, we have

$$\widehat{f}(\boldsymbol{x}_0) - E(\widehat{f}(\boldsymbol{x}_0)) = \frac{1}{k}\sum_{l=1}^{k} y_l - \frac{1}{k}\sum_{l=1}^{k} f(\boldsymbol{x}_l) = \frac{1}{k}\sum_{l=1}^{k} \varepsilon_l$$

hence, noting the independence of ε_l

$$\mathrm{Var}(\widehat{f}(\boldsymbol{x}_0)) = E\left[\left(\frac{1}{k}\sum_{l=1}^{k} \varepsilon_l\right)^2\right] = \frac{1}{k^2}(\sigma^2 + \cdots + \sigma^2) = \frac{1}{k}\sigma^2$$

The Bias-Variance Decomposition: kNN example

- Finally we have

$$\begin{aligned}\mathrm{Err}(\boldsymbol{x}_0) =& E[(Y - \widehat{f}(\boldsymbol{x}_0))^2 | X = \boldsymbol{x}_0] \\ =& \sigma^2 + \left[f(\boldsymbol{x}_0) - \frac{1}{k}\sum_{\ell=1}^{k} f(\boldsymbol{x}_\ell) \right]^2 + \frac{\sigma^2}{k}\end{aligned}$$

where we need the true model values $f(\boldsymbol{x}_0)$ and $f(\boldsymbol{x}_l)$ (on k neighbours of $\boldsymbol{x}_0$).

Bias-variance decomposition

$$\text{Err}(\boldsymbol{x}_0) = \sigma^2 + \left[f(\boldsymbol{x}_0) - \frac{1}{k} \sum_{\ell=0}^{k} f(\boldsymbol{x}_\ell) \right]^2 + \frac{\sigma^2}{k}$$

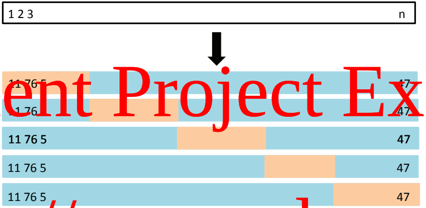
- The model complexity decreases with the number of neighbours k.
- With a small k, the bias will be relatively small since the regression function evaluated at the neighbours $f(x_\ell)$ will be close to $f(x_0)$. However, a small k means that we averaging only a few observations, leading to high variance ($\frac{\sigma^2}{k}$ is high).
- As we increase k we reduce the variance, at the cost of higher bias.

Cross validation

Cross validation

Cross validation methods are based multiple random training/validation set splits. Unlike in the validation set approach, each observation gets a turn at being predicted.

K-fold cross-validation (key concept)



The idea of K-fold cross validation is simple:

1. We randomly split the training sample into K **folds** of roughly equal size.
2. For each fold $k \in \{1, \ldots, K\}$, we estimate the model on all other folds combined, and use k as the validation set.
3. The cross validation error is the average error across the K validation sets.

5-fold and 10-fold CV. $K = 5$ or $K = 10$ folds are common choices for cross validation.

Leave one out cross validation (key concept). If we set $K = N$, this is called leave one out cross validation, or **LOOCV**. For each observation i, we train the model on all other observations, and predict i.

Algorithm Leave one out CV for regression

for i=1:N **do**
    Assign observation i to the validation set.
    Assign observations $1, \ldots, i-1, i+1 \ldots, N$ to the training set $\mathcal{D}_{-i}$.
    Estimate the model using the training set $\mathcal{D}_{-i}$
    Compute the prediction $\widehat{f}^{-i}(\boldsymbol{x}_i)$.
    Compute the squared error $(y_i - \widehat{f}^{-i}(\boldsymbol{x}_i))^2$.
end for

Compute the leave-one-out MSE:

$$\text{MSE}_{\text{CV}} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{f}^{-i}(\boldsymbol{x}_i))^2$$

LOOCV vs K-fold cross-validation

LOOCV. Approximately unbiased estimator of the expected prediction error. However, it can have high variance in some settings (since the training sets are very similar for every prediction) and a high computational cost (except in special cases).

K-fold. Lower computational cost and may have lower variance. However, it is subject to bias since the training sets are smaller than N.

Cross validation: recommendations

One standard deviation rule. Pick the simplest model within one standard deviation of the model with the lowest cross validated errors.

Choice of K. There are no general guidelines for choosing K since the trade-off between variance, bias, and computational cost is highly context specific. The variance of LOOCV tends to be relatively low with stable estimators such as linear regression.

Many predictors. When there are many predictors, pre-screening based on the entire training set may result in misleading CV.

Leave one out CV for linear regression

For a linear regression estimated by OLS, we can use a shortcut compute the leave-out-one errors without having to re-estimate the model. Given the OLS fitted values

$$\widehat{\boldsymbol{y}} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y} = \boldsymbol{H}\boldsymbol{y},$$

we can show that the leave-one-out MSE is

$$\text{MSE}_{\text{CV}} = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \widehat{f}^{-i}(\boldsymbol{x}_i)\right)^2 = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{y_i - \widehat{f}(\boldsymbol{x}_i)}{1 - H_{ii}}\right]^2,$$

where H_{ii} is the ith diagonal element of the hat matrix $\boldsymbol{H}$.

Generalised cross validation

The previous method applies to many situations in which we have fitted values of the type

$$\widehat{\boldsymbol{y}} = \boldsymbol{S}\boldsymbol{y}.$$

The **generalised cross validation** method approximates the leave one out MSE as

$$\mathsf{MSE}_{\mathsf{GCV}} = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{y_i - \widehat{f}(\boldsymbol{x}_i)}{1 - \mathrm{tr}(\boldsymbol{S})/N}\right]^2,$$

where $\mathrm{tr}(\boldsymbol{S})$ is the trace of $\boldsymbol{S}$ (the sum of the elements in its diagonal). GCV can be computationally convenient in some settings.

The bias-variance derivation details (optional)

- We first focus on the squared error loss function. Assume the true model $Y = f(X) + \epsilon$ where $E(\epsilon) = 0$ and $\text{Var}(\epsilon) = \sigma^2$
- For a test point $\mathbf{x}_0$, first we look at

$$
\begin{aligned}
(Y - \widehat{f}(\mathbf{x}_0))^2 \\
&= \Big[Y - f(\mathbf{x}_0) + f(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0) + E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0) - \widehat{f}(\mathbf{x}_0)\Big]^2 \\
&= (Y - f(\mathbf{x}_0))^2 + (f(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0))^2 + (E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0) - \widehat{f}(\mathbf{x}_0))^2 \\
&\quad + 2(Y - f(\mathbf{x}_0))(f(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0)) + 2(Y - f(\mathbf{x}_0))(E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0) - \widehat{f}(\mathbf{x}_0 \\
&\quad + 2(f(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0))(E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0) - \widehat{f}(\mathbf{x}_0))
\end{aligned}
$$

- We calculate its test error as, noting that $Y - f(\mathbf{x}_0) = \epsilon$,

$$
\begin{aligned}
&\mathsf{Err}(\mathbf{x}_0) \\
&\quad= E_{Y,\mathcal{D}}[(Y - \widehat{f}(\mathbf{x}_0))^2 | X = \mathbf{x}_0] \\
&\quad= E_{Y,\mathcal{D}}[\epsilon^2] + E_{Y,\mathcal{D}}[(f(\mathbf{x}_0) - E_{\mathcal{D}} f(\mathbf{x}_0))^2] \\
&\qquad + E_{Y,\mathcal{D}}[(E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0) - \widehat{f}(\mathbf{x}_0))^2] \\
&\qquad + 2E_{Y,\mathcal{D}}[(\epsilon)(f(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0))] + 2E_{Y,\mathcal{D}}[(\epsilon)(E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0) - \widehat{f}(\mathbf{x}_0))] \\
&\qquad + 2E_{Y,\mathcal{D}}[(f(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0))(E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0) - \widehat{f}(\mathbf{x}_0))]
\end{aligned}
$$

- Clearly

$$E_{Y,\mathcal{D}}[\epsilon^2] = E_\epsilon[\epsilon^2] = \mathsf{Var}(\epsilon) = \sigma^2$$

- There is no randomness in $(f(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0))^2$, so

$$E_{Y,\mathcal{D}}[(f(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0))^2] = (f(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0))^2$$

called the **Squared Bias**

The Bias-Variance Decomposition

- And **the Variance**

$$E_{Y,\mathcal{D}}[(E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0)-\widehat{f}(\mathbf{x}_0))^2] = E_{\mathcal{D}}[(E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0)-\widehat{f}(\mathbf{x}_0))^2] = \mathrm{Var}(\widehat{f}(\mathbf{x}_0))$$

- There is no randomness in $(f(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0))$, hence

$$E_{Y,\mathcal{D}}[(\epsilon)(f(\mathbf{x}_0)-E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0))] = E(\epsilon)(f(\mathbf{x}_0)-E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0)) = 0$$

- With independence, we have

$$E_{Y,\mathcal{D}}[(\epsilon)(E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0)-\widehat{f}(\mathbf{x}_0))] = E(\epsilon)E_{\mathcal{D}}[(E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0)-\widehat{f}(\mathbf{x}_0))] = 0$$

- Similarly

$$E_{Y,\mathcal{D}}[(f(\mathbf{x}_0)-E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0))(E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0)-\widehat{f}(\mathbf{x}_0))] = 0$$

The Bias-Variance Decomposition

- Then we have the following decomposition

$$\text{Err}(\mathbf{x}_0) = \sigma_\epsilon^2 + \text{Bias}^2(\widehat{f}(\mathbf{x}_0)) + \text{Var}(\widehat{f}(\mathbf{x}_0))$$

- The first term is irreducible error. This exists due to the nature. We cannot make it smaller through any modelling
- the second term is the squared bias, which is defined by

$$\text{Bias}^2(\widehat{f}(\mathbf{x}_0)) = (f(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0))^2$$

- And the last term is the variance (the expected squared deviation of the estimated $\widehat{f}(\mathbf{x}_0)$ around its mean, i.e.,

$$\text{Var}(\widehat{f}(\mathbf{x}_0)) = E_{\mathcal{D}}[(E_{\mathcal{D}}\widehat{f}(\mathbf{x}_0) - \widehat{f}(\mathbf{x}_0))^2]$$

where $E_{\mathcal{D}}$ is the "average" over all the training data.

The Bias-Variance Decomposition: OLS regression example

- For OLS with the estimated model $\widehat{f}_{ols}(\mathbf{x}_0) = \widehat{\boldsymbol{\beta}}^T \mathbf{x}_0 = \mathbf{x}_0^T \widehat{\boldsymbol{\beta}}$, we have

$$
\begin{aligned}
E_{\mathcal{D}} \widehat{f}_{ols}(\mathbf{x}_0) =& E_{\mathcal{D}} \left[\mathbf{x}_0^T \widehat{\boldsymbol{\beta}} \right] = \mathbf{x}_0^T E_{\mathcal{D}}[(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}] \\
=& \mathbf{x}_0^T (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T E_{\mathcal{D}}[\mathbf{y}] \\
=& \mathbf{x}_0^T (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T E_{\mathcal{D}}[\mathbf{f} + \boldsymbol{\epsilon}] \\
=& \mathbf{x}_0^T (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T \mathbf{f}. \quad \text{(Note that } E(\epsilon) = 0)
\end{aligned}
$$

where $\mathbf{f} = [f(\mathbf{x}_1), f(\mathbf{x}_2), ..., f(\mathbf{x}_N)]^T$ the vector of true model values at training data and $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, ..., \epsilon_N]^T$ given by $y_i = f(\mathbf{x}_i) + \epsilon_i$ ($i = 1, 2, ..., N$).

The Bias-Variance Decomposition: OLS regression example

- Hence the Bias square is

$$\text{Bias}^2 = \left(f(\mathbf{x}_0) - \mathbf{x}_0^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{f}\right)^2$$

- Denote
$\mathbf{h}(\mathbf{x}_0) = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{x}_0 = [h_1(\mathbf{x}_0), h_2(\mathbf{x}_0), \ldots, h_N(\mathbf{x}_0)]^T$

The Bias-Variance Decomposition: OLS regression example

- For the variance term, we have

$$
\begin{aligned}
\text{Var}(\widehat{f}(\mathbf{x}_0)) &= E_{\mathcal{D}}[(\widehat{f}_{ols}(\mathbf{x}_0) - E_{\mathcal{D}}\widehat{f}_{ols}(\mathbf{x}_0))^2] \\
&= E_{\mathcal{D}}\left[\left(\mathbf{h}(\mathbf{x}_0)^T\mathbf{y} - \mathbf{h}(\mathbf{x}_0)^T\mathbf{f}\right)^2\right] \\
&= E_{\mathcal{D}}\left[(\mathbf{h}(\mathbf{x}_0)^T\boldsymbol{\epsilon})^2\right] \\
&= E_{\mathcal{D}}[(h_1(\mathbf{x}_0)\epsilon_1 + h_2(\mathbf{x}_0)\epsilon_2 + \cdots + h_N(\mathbf{x}_0)\epsilon_N)^2]
\end{aligned}
$$

- According to assumptions, $E_{\mathcal{D}}(\epsilon^2) = \sigma^2$ and all ϵ_i's are independent of each other. We have

$$\text{Var}(\widehat{f}(\mathbf{x}_0)) = h_1(\mathbf{x}_0)\sigma^2 + h_2(\mathbf{x}_0)\sigma^2 + \cdots + h_N(\mathbf{x}_0)\sigma^2 = \|\mathbf{h}(\mathbf{x}_0)\|^2\sigma^2$$

- Finally we have

$$\text{Err}(\mathbf{x}_0) = \sigma^2 + (f(\mathbf{x}_0) - \mathbf{h}(\mathbf{x}_0)^T\mathbf{f})^2 + \|\mathbf{h}(\mathbf{x}_0)\|^2\sigma^2$$

Review questions

- How does model selection relate to the bias-variance trade-off?

- What is a validation set? How is it different from a test set?
- What is K-Fold cross validation? Describe how it works.

- What is the one standard deviation rule?