Bias-variance trade-off, model selection and cross validation

5.1 **Bias-variance trade-off—understand the concept: regression**

You have access to a European database of 1 000 000 individual trees of various types which include the following entries:

- Tree type (birch, pine, aspen etc.). In total 98 different classes.
- Age
- Height
- Circumference (at 1 meter height)
- Geographical coordinate of the position of the tree
- Vegetation type (openwoodland, mixedwood, highland, wet coniferous etc.)

All parts of Europe are well represented in the data base.

Consider a regression problem where you want to model the age of a tree based on its height and circumference. We use a linear regression model with two input variables

$$y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \epsilon,$$

where the input variables represent the height and the circumference, and the output is the age.

(a) What causes the bias of the model? Do you think the bias is high or low?

(b) What causes the variance of the model? Do you think the variance is high or low?

(c) What causes the irreducible error of the model?

(d) Where do you see the biggest improvement potential of the model (reducing bias, variance or irreducible error) and how would you go about improving it?

5.2 **Bias-variance trade-off—understand the concept: classification**

Consider the same data base as in Exercise 5.1. Now you consider a classification problem where you model the tree class as the output and the geographical coordinates as the input. We use a k-nearest neighbor model with $k = 1$.

(a) What causes the bias and variance of the model? Do you think the bias and the variance is high or low, respectively?

(b) How could you trade variance for some bias (or vice versa depending on your answer to the previous question) to reduce the mean square error without changing the set of input variables?

Note that we have not presented any formal definition of model bias and model variance for classification problems. However, we can still reason about the concepts in the same manner as we did for the regression setting.

5.3 **Bias and variance when estimating a constant with ridge regression**

To illustrate the concept of bias and variance, let us consider the very simple case of estimating the constant 1 as a linear regression problem. Assume that we have one data sample y_1 ($n = 1$) from

$$y = f_0(\mathbf{x}) + \epsilon, \qquad f_0(\mathbf{x}) = 1,$$

where ϵ has mean 0 and variance σ^2. We use linear regression with only a θ_0-term,

$$y = \theta_0 + \epsilon,$$

where we learn θ_0 using *ridge regression* with regularization parameter λ. Since this problem is so simple that it has no inputs $\mathbf{x}$, the distribution $p(\mathbf{x})$ does not matter.

(a) What is the closed-form solution for $\widehat{\theta}_0$, as a function of the training data y_1 and the regularization parameter λ? What is $\widehat{y}_\star(\mathbf{x}_\star; \mathcal{T})$?

(b) What is the average trained model $\bar{f}(\mathbf{x}_\star) \triangleq \mathbb{E}_\mathcal{T}\left[\widehat{y}_\star(\mathbf{x}_\star; \mathcal{T})\right]$? The expectation operator $\mathbb{E}_\mathcal{T}$ is an expectation over all random variations in the training data.

(c) What is the squared bias $\mathbb{E}_\star\left[(\bar{f}(\mathbf{x}_\star) - f_0(\mathbf{x}_\star))^2\right]$? The expectation operator $\mathbb{E}_\star$ is here an expectation over the test input $\mathbf{x}_\star \sim p(\mathbf{x})$.

(d) What is the variance $\mathbb{E}_\star\left[\mathbb{E}_\mathcal{T}\left[(\widehat{y}(\mathbf{x}_\star; \mathcal{T}) - \bar{f}(\mathbf{x}_\star))^2\right]\right]$?

(e) What is the irreducible error $\mathbb{E}[\epsilon^2]$?

(f) What is $\bar{E}_{\text{new}} = \mathbb{E}_\mathcal{T}\left[\mathbb{E}_\star\left[(\widehat{y}(\mathbf{x}_\star; \mathcal{T}) - y_\star)^2\right]\right]$ for this problem?

(g) For which value of the regularization parameter λ is $\bar{E}_{\text{new}}$ minimized? What does it tell us about the bias-variance trade-off for this (simple) problem?

5.4 Model selection

Suppose that you collect $n = 200$ observations of a single variable x and its single output y. You then go ahead and fit the linear regression model

$$y = \theta_0 + \theta_1 x + \epsilon \tag{5.1}$$

to the first half of your data (the training data), as well as a cubic polynomial

$$y = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \epsilon. \tag{5.2}$$

(a) Suppose that the true relationship between x and y is linear. Which of the models, (5.1) or (5.2), will be able to fit your training data the best? That is, which model gives you the smallest E_{train}?

(b) You now consider the second half of the data (the validation data), which you did not use for training. Using your two previously trained models, which one will be able to predict y in the validation data the best? That is, which model gives you the smallest $E_{\text{hold-out}}$?

(c) Consider (a) and (b) again, but suppose that the true relationship is not linear. Which model will have smallest E_{train} and $E_{\text{hold-out}}$, respectively?

5.5 **Model flexibility**

Consider a regression problem $y = f(x) + \epsilon$. Assume (as is often the case in practice) that we have access to a family of models with different levels of flexibility. Make a sketch of the typical shapes of the following curves, as the model flexibility goes from low to high:

(i) The squared model bias.

(ii) The model variance.

(iii) The irreducible error, $\text{Var}(\epsilon)$.

(iv) The training mean-squared error.

(v) The expected test mean-squared error.

Think about the shape of each curve, as well as how they relate to each other.

5.6 **Bias and variance when learning a linear function for a quadratic relationship**

As a slightly more involved illustration of the concept of bias and variance, let us consider the case of learning a linear function from data that actually is generated by a quadratic model. Assume that the distribution over data $p(x, y)$ is (implicitly) defined by

$$y = f_0(x), \qquad f_0(x) = x^2, \qquad x \sim \mathcal{U}[-1, 1],$$

from which you randomly observe $n = 2$ independent data points. Those two data point become our training data $\mathcal{T}$. To simplify the calculations, we have restricted ourselves to a problem with no noise ϵ; the only randomness in the problem is for which two input samples x_1 and x_2 we happen to learn about $f_0(x)$ by observing y_1 and y_2.

From the training data with two samples, learn a linear regression model,

$$y = \theta_0 + \theta_1 x + \epsilon,$$

where we assume ϵ is Gaussian.

(a) What is the closed-form solution for $\widehat{y}(x_\star, \mathcal{T})$, as a function of the inputs in the training data x_1, x_2?

(b) What is the average trained model $\bar{f}(x_\star) \triangleq \mathbb{E}_{\mathcal{T}}\left[\widehat{y}_\star(x_\star; \mathcal{T})\right]$?

(c) What is the squared bias $\mathbb{E}_\star\left[(\bar{f}(x_\star) - f_0(x_\star))^2\right]$?

(d) What is the variance $\mathbb{E}_\star\left[\mathbb{E}_{\mathcal{T}}\left[(\widehat{y}(x_\star; \mathcal{T}) - \bar{f}(x_\star))^2\right]\right]$?

(e) What is $\bar{E}_{\text{new}} = \mathbb{E}_{\mathcal{T}}\left[\mathbb{E}_\star\left[(\widehat{y}(x_\star; \mathcal{T}) - y_\star)^2\right]\right]$ for this problem? What would $\bar{E}_{\text{new}}$ be if true relationship had been $f_0(x) = x$ (instead of x^2)?

5.7 For leave-one-out cross validation (or equivalently k-fold cross validation with $k = n$) the cross validation error $E_{\text{k-fold}}$ for ridge regression actually has a closed-form solution

$$E_{\text{k-fold}} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\widehat{y}_i - y_i}{1 - [\mathbf{H}(\lambda)]_{ii}}\right)^2, \tag{5.3}$$

where $\widehat{y}_i$ is the prediction of y_i when the model is learned from *all* n data points (no data point is left out), λ the regularization parameter and $[\mathbf{H}(\lambda)]_{ii}$ is element (i, i) of the matrix $\mathbf{H}(\lambda) = \mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I)^{-1}\mathbf{X}^\mathsf{T}$.

In this problem, we will let $\mathbf{x}_i^\mathsf{T}$ denote the entire ith row of $\mathbf{X}$.

(a) Let $\mathbf{X}_{-i}$ denote the matrix $\mathbf{X}$ where row i is removed, and $\mathbf{y}_{-i}$ is the column vector $\mathbf{y}$ with element i removed. Show that

$$\begin{aligned}
\mathbf{X}_{-i}^\mathsf{T}\mathbf{X}_{-i} &= \mathbf{X}^\mathsf{T}\mathbf{X} - \mathbf{x}_i\mathbf{x}_i^\mathsf{T}, \\
\mathbf{X}_{-i}^\mathsf{T}\mathbf{y}_{-i} &= \mathbf{X}^\mathsf{T}\mathbf{y} - \mathbf{x}_i y_i, \text{ and} \\
[\mathbf{H}(\lambda)]_{ii} &= \mathbf{x}_i^\mathsf{T}(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I)^{-1}\mathbf{x}_i
\end{aligned}$$

(b) Using the results from (a) and a special case of the matrix inversion lemma

$$(\mathbf{A} - \mathbf{v}\mathbf{v}^\mathsf{T})^{-1} = \mathbf{A}^{-1} + \frac{\mathbf{A}^{-1}\mathbf{v}\mathbf{v}^\mathsf{T}\mathbf{A}^{-1}}{1 - \mathbf{v}^\mathsf{T}\mathbf{A}^{-1}\mathbf{v}},$$

show that

$$\widehat{\boldsymbol{\theta}}_{-i} = \widehat{\boldsymbol{\theta}} + \frac{1}{1 - [\mathbf{H}(\lambda)]_{ii}}(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I)^{-1}\mathbf{x}_i(\widehat{y}_i - y_i).$$

Here $\widehat{\boldsymbol{\theta}}_{-i}$ are the parameters learned from all data points except i, and $\widehat{\boldsymbol{\theta}}$ the parameters learned from all data.

Hint: Start from the ridge regression expression for $\widehat{\boldsymbol{\theta}}_{-i}$ as a function of $\mathbf{X}_{-i}$, $\mathbf{y}_{-i}$ and λ.

(c) Use your result from (b) to derive eq. (5.3), starting from

$$E_{\text{k-fold}} = \frac{1}{n}\sum_{i=1}^{n}\left(\widehat{\boldsymbol{\theta}}_{-i}^\mathsf{T}\mathbf{x}_i - y_i\right)^2.$$

(d) Describe (in a few sentences) what eq. (5.3) can be used for in practice.

Solutions

5.1 (a) The bias can be thought of as the error caused by the simplifying assumptions built into the model. In this example, we use a very simple model, which only takes the height and circumference of a tree into account. For a certain class of trees it will most likely do a systematic error in estimating the age, resulting in a possibly high model bias.

(b) The variance is about the stability of the model in response to new training examples. Assume you randomly would split the data set in two halves and estimate the linear regression model for each of the two halves. Each of them would probably get roughly the same estimated parameter $\widehat{\boldsymbol{\theta}}$ and hence also the same predicted output for a new output $f(x_*, \widehat{\boldsymbol{\theta}})$ since the data set is big in comparison to the complexity of the model. Hence, the variance is probably fairly low.

(c) The irreducible error is the error that we cannot reduce even though we would have a very good model trained on an infinite amount of data. This is based on the notion that there are individual age variations amongst the trees that cannot be explained based only on the input variables (features) in the data base. Another source for the irreducible error is the measurement error when measuring the age, height, circumference, etc.

(d) The main problem with the model is probably the high bias. This can be reduced by including more input variables present in the database such as tree class and vegetation type.

5.2 (a) In this model we will classify a new tree according to the class of the closest tree in the training data. This is highly dependent on the selection of the training data. If we would split the data set in two halves and make a k-nearest neighbor model with $k = 1$ for each of these two data sets, it is likely that we would get very different decision boundaries for the two models since we base the predictions on a single training data point. This means that we have a high variance in the model. As for the bias: whether or not this is high or low depends on whether we think that the geographic location alone is sufficiently informative for determining the tree type. If this is the case, then the bias is low, since the 1-NN model can describe very flexible mappings (in this case from "location" to "tree type"). If, however, there is relevant information about the tree type available in the features not used in the model, then this can be viewed as a bias due to under-modeling of the "true" input-output relationship.

(b) A way to reduce the variance would be to increase k such that the classification does not depend on a single data point, but rather on a group of trees in a neighborhood. However, this also increases the bias (since we make a simplifying assumption) and the most common tree class will be favored. For example, in the limit where $k = 1\,000\,000$, all test point would be classified according to the most common class in the data base causing a huge model bias.

5.3 (a) In general, ridge regression is $\widehat{\boldsymbol{\theta}} = \left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda\mathbf{I}\right)^{-1}\mathbf{X}^\mathsf{T}\mathbf{y}$, which for our problem gives

$$\widehat{\theta}_0 = \frac{y_1}{1+\lambda}.$$

We have, for this very simple problem, that $\widehat{y}_\star(\mathbf{x}_\star; \mathcal{T}) = \widehat{\theta}_0$.

(b) We have

$$\bar{f}(\mathbf{x}_\star; \mathcal{T}) = \mathbb{E}_\mathcal{T}\left[\widehat{y}_\star(\mathbf{x}_\star; \mathcal{T})\right] = \mathbb{E}_\mathcal{T}\left[\frac{y_1}{1+\lambda}\right] = \mathbb{E}_\mathcal{T}\left[\frac{1+\epsilon}{1+\lambda}\right] = \frac{1}{1+\lambda}(1 + \underbrace{\mathbb{E}_\mathcal{T}\left[\epsilon\right]}_{0}) = \frac{1}{1+\lambda}$$

(c) The squared bias is

$$\mathbb{E}_\star\left[(\bar{f}(\mathbf{x}_\star) - f_0(\mathbf{x}_\star))^2\right] = \mathbb{E}_\star\left[\left(\frac{1}{1+\lambda} - 1\right)^2\right] = \left(\frac{1}{1+\lambda} - 1\right)^2 = \left(\frac{\lambda}{1+\lambda}\right)^2$$

(d) The variance is

$$\begin{aligned}\mathbb{E}_\star\left[\mathbb{E}_\mathcal{T}\left[(\widehat{y}(\mathbf{x}_\star; \mathcal{T}) - \bar{f}(\mathbf{x}_\star))^2\right]\right] &= \mathbb{E}_\star\left[\mathbb{E}_\mathcal{T}\left[\left(\frac{1+\epsilon}{1+\lambda} - \frac{1}{1+\lambda}\right)^2\right]\right] = \mathbb{E}_\star\left[\mathbb{E}_\mathcal{T}\left[\left(\frac{\epsilon}{1+\lambda}\right)^2\right]\right] \\ &= \frac{1}{(1+\lambda)^2}\mathbb{E}_\star\left[\mathbb{E}_\mathcal{T}\left[\epsilon^2\right]\right] = \frac{\sigma^2}{(1+\lambda)^2}\end{aligned}$$

(e) The irreducible error is $\mathbb{E}[\epsilon^2] = \sigma^2$.

(f) From the lecture notes, we have that $\bar{E}_{\text{new}}$ = squared bias + variance + irreducible error, hence

$$\bar{E}_{\text{new}} = \left(\frac{\lambda}{1+\lambda}\right)^2 + \frac{\sigma^2}{(1+\lambda)^2} + \sigma^2.$$

(g) By differentiating $\bar{E}_{\text{new}}$ with respect to λ, we get

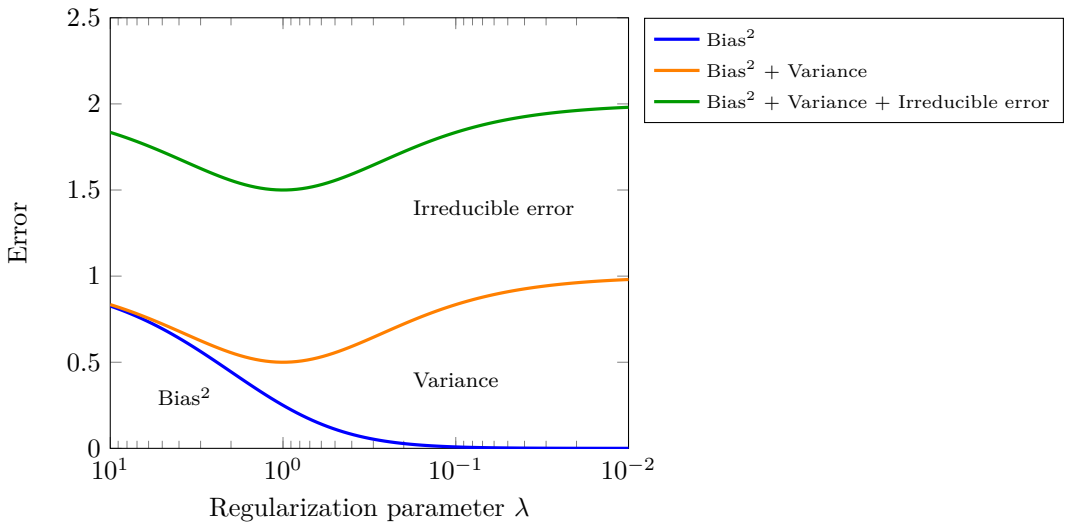
$$\frac{\partial}{\partial\lambda}\bar{E}_{\text{new}} = \frac{2(\lambda-\sigma^2)}{(1+\lambda)^3}.$$

We conclude that for this particular problem, $\bar{E}_{\text{new}}$ is minimized when $\lambda = \sigma^2$. In terms of bias-variance trade-off, we can see that for this problem

i. There is no bias when $\lambda = 0$, but the variance peaks at σ^2

ii. There is no variance when $\lambda \to \infty$, but the bias peaks at 1

iii. The smallest expected new data error, $\bar{E}_{\text{new}}$, is achieved at neither of those extremes, but at $\lambda = \sigma^2$. For $\sigma^2 = 1$, we can make the following plot:



5.4 (a) Since (5.2) has more flexibility than (5.1) (note that if you set $\theta_2 = \theta_3 = 0$ in (5.2), you get (5.1)), it will be able to fit to the training data *at least as good* as (5.1). Thus, E_{train} for (5.2) is $\leq$ than E_{train} for (5.1).

(b) (5.2) will most likely overfit to the training data, since the model is more flexible than the true relationship between the input and output. Thus, $E_{\text{hold-out}}$ for (5.2) is likely to be $\geq$ than $E_{\text{hold-out}}$ for (5.1).

(c) The argument for (a) is still applicable in the training case, i.e., E_{train} for (5.2) is $\leq$ than E_{train} for (5.1). For $E_{\text{hold-out}}$, we cannot tell unless we have more information about the true relationship between the input and the output.

5.5 (a) The model bias can be thought of as the error caused by the simplifying assumptions built into the model. Therefore, a model with low flexibility has a high bias and the bias decreases as the flexibility of the model increases.

(b) The model variance is about the stability of the model in response to new training examples. For a model with low flexibility the variance is low since the model would not change much in response to new training data. As the flexibility increases the model becomes more adaptive to new data and the variance increases.

(c) The irreducible error is the error term that has an impact on the output but is not explainable through the input variables. This error term cannot be reduced even though we would have a very good model trained on an infinite amount of data. Specifically, the irreducible error does not depend on the model flexibility and is constant in this respect.

(d) The training mean-squared error is a measure of how well our training data is described by our model. This is the cost function that we typically minimize in a regression problem. As a consequence, if the flexibility of the model increases, this error is reduced since we can achieve a lower value of the cost function and fit our training data better.

(e) The expected test mean-squared error is the expected mean-squared error of for a new unseen data point. This is a measure of the generalization performance of the model. The expected test mean-squared error is the sum of the squared model bias, the model variance and the irreducible error. Hence, it will be large both for low flexibility and high flexibility, but typically has a minimum in between where we find the optimal bias–variance trade-off.

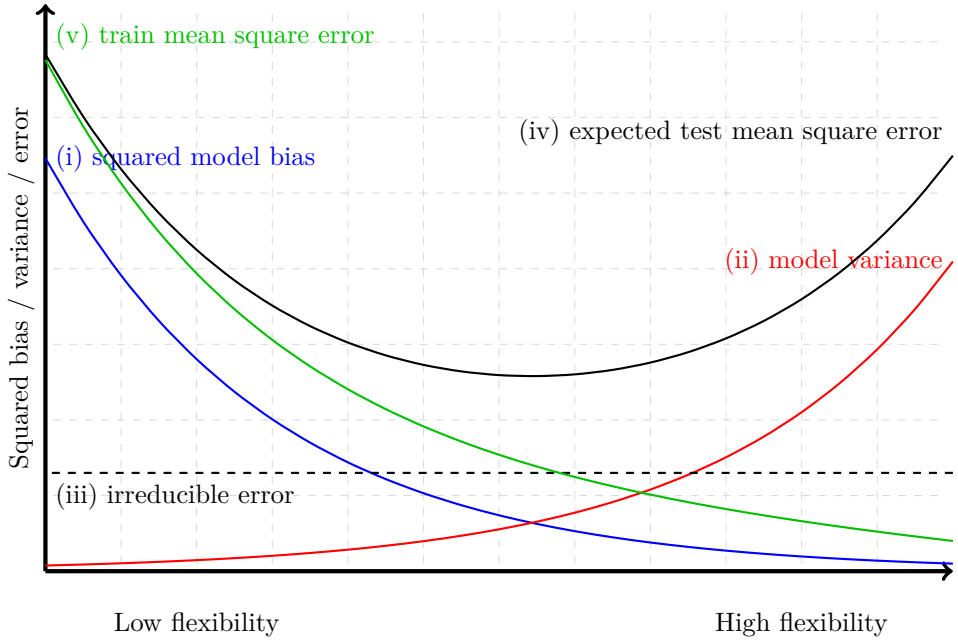


Figure 5.1. The optimal bias–variance trade–off.

5.6 (a) By denoting $\mathbf{y} = [x_1^2 \; x_2^2]^\mathsf{T}$, one way to find the expression is to analytically solve $\mathbf{X}\widehat{\boldsymbol{\theta}} = \mathbf{y}$,

$$\begin{bmatrix} 1 & x_1 \\ 1 & x_2 \end{bmatrix} \begin{bmatrix} \widehat{\theta}_0 \\ \widehat{\theta}_1 \end{bmatrix} = \begin{bmatrix} x_1^2 \\ x_2^2 \end{bmatrix} \Rightarrow \cdots \Rightarrow \begin{bmatrix} \widehat{\theta}_0 \\ \widehat{\theta}_1 \end{bmatrix} = \begin{bmatrix} -x_1 x_2 \\ x_1 + x_2 \end{bmatrix}.$$

This gives

$$\widehat{y}(x_\star; \mathcal{T}) = \widehat{\theta}_0 + \widehat{\theta}_1 x_\star = -x_1 x_2 + (x_1 + x_2) x_\star.$$

(b) Since the only random element of the training data is its two input samples x_1, x_2, which both have a uniform distribution on $[-1, 1]$, the average trained model $\bar{f}(x)$ is

$$\bar{f}(x_\star) = \mathbb{E}_{\mathcal{T}}\left[\widehat{y}(x_\star; \mathcal{T})\right] = \iint \widehat{y}(x_\star; x_1, x_2) \underbrace{p(x_1, x_2)}_{\frac{1}{4} \text{ on } [-1,1]^2} dx_1 dx_2 = \frac{1}{2} \cdot \frac{1}{2} \int_{-1}^{1} \int_{-1}^{1} \widehat{y}(x_\star; x_1, x_2) dx_1 dx_2 =$$

$$= \frac{1}{4} \int_{-1}^{1} \int_{-1}^{1} -x_1 x_2 + (x_1 + x_2) x_\star dx_1 dx_2 = 0$$

(c) The squared bias is

$$\mathbb{E}_\star\left[(\bar{f}(x_\star) - f_0(x_\star))^2\right] = \mathbb{E}_\star\left[(0 - x_\star^2)^2\right] = \frac{1}{2} \int_{-1}^{1} x_\star^4 dx_\star = \frac{1}{5}.$$

(d) The variance is

$$\begin{aligned}
\mathbb{E}_\star\left[\mathbb{E}_\mathcal{T}\left[(\widehat{y}(x_\star;\mathcal{T})-\bar{f}(x_\star))^2\right]\right] &= \mathbb{E}_\star\left[\mathbb{E}_\mathcal{T}\left[(x_1x_2-(x_1+x_2)x_\star)^2\right]\right] \\
&= \mathbb{E}_\star\left[\mathbb{E}_\mathcal{T}\left[x_1^2x_2^2\right]-2x_\star\mathbb{E}_\mathcal{T}\left[x_1^2x_2^2(x_1+x_2)\right]+x_\star^2\mathbb{E}_\mathcal{T}\left[(x_1+x_2)^2\right]\right] \\
&= \mathbb{E}_\mathcal{T}\left[x_1^2x_2^2\right]-2\underbrace{\mathbb{E}_\star\left[x_\star\right]}_{0}\mathbb{E}_\mathcal{T}\left[x_1^2x_2^2(x_1+x_2)\right]+\mathbb{E}_\star\left[x_\star^2\right]\mathbb{E}_\mathcal{T}\left[(x_1+x_2)^2\right] \\
&= \frac{1}{4}\int_{-1}^{1}\int_{-1}^{1}x_1^2x_2^2dx_1dx_2+\left(\frac{1}{2}\int_{-1}^{1}x_\star^2dx_\star\right)\left(\frac{1}{4}\int_{-1}^{1}\int_{-1}^{1}(x_1+x_2)^2dx_1dx_2\right) \\
&= \frac{1}{9}+\frac{1}{3}\cdot\frac{2}{3}=\frac{1}{3}.
\end{aligned}$$

(e) Since there is no irreducible error, we have that $\bar{E}_{\text{new}}$ = squared bias + variance (this can be thoroughly derived using the definition of $\bar{E}_{\text{new}}$),

$$\bar{E}_{\text{new}}=\frac{1}{5}+\frac{1}{3}=\frac{8}{15}.$$

This is not due to any noise (there is no noise in this problem!), but only the fact that the model $(\theta_0+\theta_1x)$ can not describe the 'reality' $(f_0(x)=x^2)$ perfectly well. If the 'reality' had been $f_0(x)=x$, there would have been no bias or variance (because there is no noise), and hence $\bar{E}_{\text{new}}=0$.

5.7 (a) We have

$$\mathbf{X}_{-i}^\mathsf{T}\mathbf{X}_{-i}=\sum_{j=1,j\neq i}^{n}\mathbf{x}_j\mathbf{x}_j^\mathsf{T}=\left(\sum_{j=1}^{n}\mathbf{x}_j\mathbf{x}_j^\mathsf{T}\right)-\mathbf{x}_i\mathbf{x}_i^\mathsf{T}=\mathbf{X}^\mathsf{T}\mathbf{X}-\mathbf{x}_i\mathbf{x}_i^\mathsf{T}$$

and similarly

$$\mathbf{X}_{-i}^\mathsf{T}\mathbf{y}_{-i}=\sum_{j=1,j\neq i}^{n}\mathbf{x}_jy_j=\left(\sum_{j=1}^{n}\mathbf{x}_jy_j^\mathsf{T}\right)-\mathbf{x}_iy_i=\mathbf{X}^\mathsf{T}\mathbf{y}-\mathbf{x}_iy_i.$$

For matrices $\mathbf{A}\in\mathbb{R}^{n\times n}$ and $i\in\{1,\ldots,n\}$ we can express element $[\mathbf{A}]_{ii}$ as $[\mathbf{A}]_{ii}=\mathbf{e}_i^\mathsf{T}\mathbf{A}\mathbf{e}_i$, where $\mathbf{e}_i\in\mathbb{R}^n$ is the ith unit vector (its ith element is 1 and all other entries are 0). Thus we get

$$\begin{aligned}
[\mathbf{H}(\lambda)]_{ii} &= \mathbf{e}_i^\mathsf{T}\mathbf{H}(\lambda)\mathbf{e}_i=\mathbf{e}_i^\mathsf{T}\left(\mathbf{X}(\mathbf{X}^\mathsf{T}\mathbf{X}+\lambda I)^{-1}\mathbf{X}^\mathsf{T}\right)\mathbf{e}_i \\
&= \left(\mathbf{e}_i^\mathsf{T}\mathbf{X}\right)(\mathbf{X}^\mathsf{T}\mathbf{X}+\lambda I)^{-1}\left(\mathbf{X}^\mathsf{T}\mathbf{e}_i\right)=\mathbf{x}_i^\mathsf{T}(\mathbf{X}^\mathsf{T}\mathbf{X}+\lambda I)^{-1}\mathbf{x}_i,
\end{aligned}$$

where we used that $\mathbf{X}^\mathsf{T}\mathbf{e}_i=\mathbf{x}_i$.

(b) From the ridge regression expression we know that the parameters $\widehat{\boldsymbol{\theta}}_{-i}$ learned from all data points except i are given by

$$\widehat{\boldsymbol{\theta}}_{-i}=\left(\mathbf{X}_{-i}^\mathsf{T}\mathbf{X}_{-i}+\lambda I\right)^{-1}\mathbf{X}_{-i}^\mathsf{T}\mathbf{y}_{-i}.$$

Using the first two equations from (a) we get

$$\widehat{\boldsymbol{\theta}}_{-i}=\left(\mathbf{X}^\mathsf{T}\mathbf{X}-\mathbf{x}_i\mathbf{x}_i^\mathsf{T}+\lambda I\right)^{-1}(\mathbf{X}^\mathsf{T}\mathbf{y}-\mathbf{x}_iy_i). \tag{5.4}$$

Applying the matrix inversion lemma with $\mathbf{A}=\mathbf{X}^\mathsf{T}\mathbf{X}+\lambda I$ and $\mathbf{v}=\mathbf{x}_i$ and using the third equation from (a) yields

$$\begin{aligned}
\left(\mathbf{X}^\mathsf{T}\mathbf{X}-\mathbf{x}_i\mathbf{x}_i^\mathsf{T}+\lambda I\right)^{-1} &= \left(\mathbf{X}^\mathsf{T}\mathbf{X}+\lambda I\right)^{-1}+\frac{\left(\mathbf{X}^\mathsf{T}\mathbf{X}+\lambda I\right)^{-1}\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X}+\lambda I\right)^{-1}}{1-\mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X}+\lambda I\right)^{-1}\mathbf{x}_i} \\
&= \left(\mathbf{X}^\mathsf{T}\mathbf{X}+\lambda I\right)^{-1}+\frac{\left(\mathbf{X}^\mathsf{T}\mathbf{X}+\lambda I\right)^{-1}\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X}+\lambda I\right)^{-1}}{1-[\mathbf{H}(\lambda)]_{ii}}.
\end{aligned}$$

If we plug this result in eq. (5.4) and use the third equation from (a) we get

$$\begin{aligned}
\widehat{\boldsymbol{\theta}}_{-i} &= \left(\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1} + \frac{\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}}{1 - [\mathbf{H}(\lambda)]_{ii}} \right) (\mathbf{X}^\mathsf{T}\mathbf{y} - \mathbf{x}_i y_i) \\
&= \left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{X}^\mathsf{T}\mathbf{y} + \frac{1}{1 - [\mathbf{H}(\lambda)]_{ii}}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{X}^\mathsf{T}\mathbf{y} \\
&\quad - \left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{x}_i \left(1 + \frac{\mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{x}_i}{1 - [\mathbf{H}(\lambda)]_{ii}} \right) y_i \\
&= \left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{X}^\mathsf{T}\mathbf{y} + \frac{1}{1 - [\mathbf{H}(\lambda)]_{ii}}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{x}_i\mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{X}^\mathsf{T}\mathbf{y} \\
&\quad - \frac{1}{1 - [\mathbf{H}(\lambda)]_{ii}}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{x}_i y_i \\
&= \left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{X}^\mathsf{T}\mathbf{y} \\
&\quad + \frac{1}{1 - [\mathbf{H}(\lambda)]_{ii}}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{x}_i \left(\mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{X}^\mathsf{T}\mathbf{y} - y_i \right).
\end{aligned}$$

The parameters $\widehat{\boldsymbol{\theta}}$ learned from all data points satisfy

$$\widehat{\boldsymbol{\theta}} = \left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{X}^\mathsf{T}\mathbf{y},$$

and hence the model learned from all data points predicts the value

$$\widehat{y}_i = \widehat{\boldsymbol{\theta}}^\mathsf{T}\mathbf{x}_i = \mathbf{x}_i^\mathsf{T}\widehat{\boldsymbol{\theta}} = \mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{X}^\mathsf{T}\mathbf{y}$$

for the ith data point. Thus we obtain

$$\widehat{\boldsymbol{\theta}}_{-i} = \widehat{\boldsymbol{\theta}} + \frac{1}{1 - [\mathbf{H}(\lambda)]_{ii}}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{x}_i \left(\widehat{y}_i - y_i\right).$$

(c) Using the result from (b) we get

$$\begin{aligned}
E_{\text{k-fold}} &= \frac{1}{n}\sum_{i=1}^{n}\left(\widehat{\boldsymbol{\theta}}_{-i}^\mathsf{T}\mathbf{x}_i - y_i\right)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(\left(\widehat{\boldsymbol{\theta}} + \frac{1}{1 - [\mathbf{H}(\lambda)]_{ii}}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{x}_i\left(\widehat{y}_i - y_i\right)\right)^\mathsf{T}\mathbf{x}_i - y_i\right)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(\left(\widehat{\boldsymbol{\theta}}^\mathsf{T} + \frac{1}{1 - [\mathbf{H}(\lambda)]_{ii}}\mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\left(\widehat{y}_i - y_i\right)\right)\mathbf{x}_i - y_i\right)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(\widehat{\boldsymbol{\theta}}^\mathsf{T}\mathbf{x}_i + \frac{1}{1 - [\mathbf{H}(\lambda)]_{ii}}\mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{x}_i\left(\widehat{y}_i - y_i\right) - y_i\right)^2.
\end{aligned}$$

Since $\widehat{y}_i = \widehat{\boldsymbol{\theta}}^\mathsf{T}\mathbf{x}_i$ and from (a) we know that $[\mathbf{H}(\lambda)]_{ii} = \mathbf{x}_i^\mathsf{T}\left(\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda I\right)^{-1}\mathbf{x}_i$, we obtain

$$\begin{aligned}
E_{\text{k-fold}} &= \frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}_i + \frac{1}{1 - [\mathbf{H}(\lambda)]_{ii}}[\mathbf{H}(\lambda)]_{ii}\left(\widehat{y}_i - y_i\right) - y_i\right)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\widehat{y}_i - y_i}{1 - [\mathbf{H}(\lambda)]_{ii}}\right)^2.
\end{aligned}$$

(d) Ideally, we would like to choose regularization parameter λ in ridge regression such that the new data error of the resulting trained model is minimized. Unfortunately, we can not usually compute the new data error. The cross validation error $E_{\text{k-fold}}$ provides an approximation of the new data error, and hence can be used to select

a "good" λ. However, performing k-fold cross validation can be computationally expensive and time consuming, since usually it requires training the model from scratch and evaluating its performance c times. In particular, performing leave-one-out cross validation with $c = n$ might not be feasible. In ridge regression, however, eq. (1) allows us to train the model only once on the full dataset and to compute the leave-one-out cross validation error from its prediction errors without having to retrain the model n times on all subsets of our dataset with $n - 1$ samples.