EXAM IN STATISTICAL MACHINE LEARNING
STATISTISK MASKININLÄRNING

DATE AND TIME: June 8, 2022, 8.00–13.00

RESPONSIBLE TEACHER: Dave Zachariah

NUMBER OF PROBLEMS: 5

AIDING MATERIAL: Calculator, mathematical handbook

PRELIMINARY GRADES:	grade 3	23 points
	grade 4	33 points
	grade 5	43 points

Some general instructions and information:

- Your solutions should be given in English.
- Only write on one page of the paper.
- Write your exam code and a page number on all pages.
- Do not use a red pen.
- Use separate sheets of paper for the different problems (i.e. the numbered problems, 1–5).

With the exception of Problem 1, all your answers must be clearly motivated! A correct answer without a proper motivation will score zero points!

Good luck!

1. i.
   ii.
   iii.
   iv.
   v.
   vi.
   vii.
   viii.
   ix.
   x.

2. (a) For model (1), we have

$$X = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix},$$

and the estimate $\hat{\alpha}_0$ which minimizes the MSE is found via the normal equations

$$\hat{\alpha}_0 = (X^\mathsf{T} X)^{-1} X^\mathsf{T} \mathbf{y} = -\frac{1}{3}.$$

For model (2), we have

$$X = \begin{bmatrix} 1 & -1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix},$$

and hence

$$\hat{\beta} = \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \end{bmatrix} = (X^\mathsf{T} X)^{-1} X^\mathsf{T} \mathbf{y} = \begin{bmatrix} -\frac{1}{3} \\ -\frac{1}{2} \end{bmatrix}.$$

(b) Model (2) is more flexible than model (1), and model (2) will therefore always be able to adapt to training data (i.e., low MSE) *at least as good* as model (1).

(c) In leave-one-out cross validation, we make a loop which runs from $j = 1$ to $j = n$, and in our problem $n = 3$. For each value of j, we estimate the parameters using all data points except for number j, and then compute the prediction y_j and note the squared error. Start with model (1):

$j = 1$: $X = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \Rightarrow \hat{\alpha}_0 = -\frac{1}{2}, \text{MSE} = (\hat{y}_1 - y_1)^2 = (-\frac{1}{2} - 0)^2 = \frac{1}{4}$

$j = 2$: $X = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \Rightarrow \hat{\alpha}_0 = -\frac{1}{2}, \text{MSE} = (\hat{y}_2 - y_2)^2 = (-\frac{1}{2} - 0)^2 = \frac{1}{4}$

$j = 3$: $X = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow \hat{\alpha}_0 = 0, \text{MSE} = (\hat{y}_3 - y_3)^2 = (0 + 1)^2 = 1$

which gives an average MSE $\frac{1}{2}$.

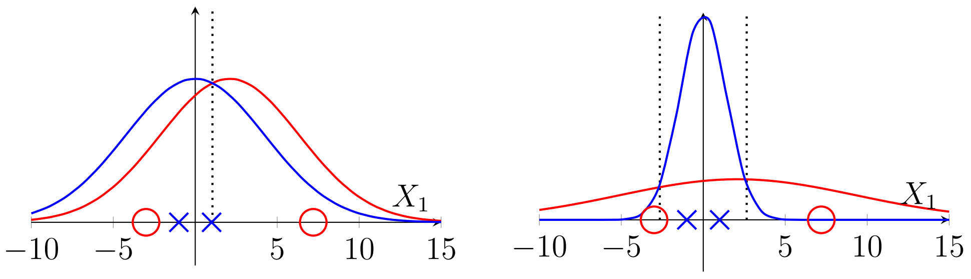
For model (2), we get in a similar fashion

$j = 1$: $X = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \Rightarrow \hat{\beta} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \text{MSE} = (\hat{y}_1 - y_1)^2 = (-1-0)^2 = 1$

$j = 2$: $X = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \Rightarrow \hat{\beta} = \begin{bmatrix} -\frac{1}{2} \\ -\frac{1}{2} \end{bmatrix}, \text{MSE} = (\hat{y}_2 - y_2)^2 = (-\frac{1}{2}-0)^2 = \frac{1}{4}$

$j = 3$: $X = \begin{bmatrix} 1 & -1 \\ 1 & 0 \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow \hat{\beta} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \text{MSE} = (\hat{y}_3 - y_3)^2 = (0+1)^2 = 1$

which gives an average MSE of $\frac{3}{4}$. Thus model (1) performs the best in this sense.



(a) LDA

(b) QDA

Figure 1: Graphical illustration of discriminant analysis (LDA and QDA) on Problem 3a. LDA has one decision boundary. 3 out of 4 training points are correctly classified. QDA has two decision boundaries. All training points are correctly classified.

3. (a)
   - (i) LDA - 25%: $\mu_0 = (-3+7.2)/2 = 2.1$, $\mu_1 = (-1+1)/2 = 0$ and $\pi_0 = \pi_1 = 1/2$. Hence, the decision boundary will pass at $(\mu_0 + \mu_1)/2 = 1.05$ $\rightarrow$ The first three points will be classified as 1 and the forth as 0 $\rightarrow$ 25% missclassification error. See also Figure 1a.
   - (iv) kNN-1 - 0%: kNN with $k = 1$ always gives zero training error (unless you don't have multiple training data point with the exact same input but different outputs.
   - (v) kNN-3 - 50%: Regardless of x, the two points at -1 and 1 will always belong to the group of the three closest neighbors $\rightarrow$ all points are classified as 1 $\rightarrow$ 50% missclassification error.
   - (vi) Tree - 25%: The best binary split you can achieve is between -3 and -1 or between 1 and 7.2. Both leading to one point that is missclassified $\rightarrow$ 25% missclassification error.
   - (iii) LogReg - 50%: Logistic regression is a linear classifier and can hence not classify all points correctly (since they are not linearly separable). We already have two classifiers with 25% $\rightarrow$ LogReg gives 50% missclassification error (the decision boundary will be just left of the point 1).
   - (vi) QDA - 0 %: QDA is a nonlinear classifier and can potentially achieve 0% missclassification error, which is the only option left. See also Figure 1b.

   (b)
   - (iii) The data is linearly separable. Hence, training logistic

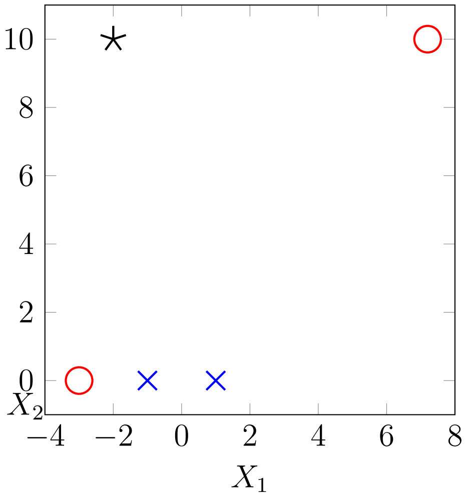


Figure 2: The training data in Problem 3b (x and o) together with the test data point in Probelm 3c.

regression with maximum likelihood will fit the data exactly and all training points are classified correctly.

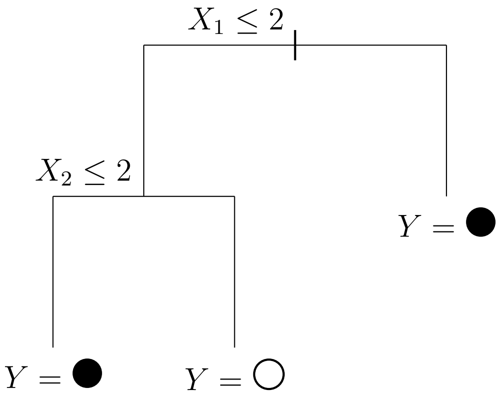
- (v) k-NN with $k = 3$ will still classify all training data points as 1 since the two data points at [-1,0] and [1,0] will be in the group of the three closest points for all the locations of the four training points, see Figure 2.
- (vi) Since three of the training points have the same X_2-value, there is no possible split to be made (neither with X_1 nor X_2) that would classify all training points correctly. Hence, no improvement.

(c) The distance between the test point and the training points is

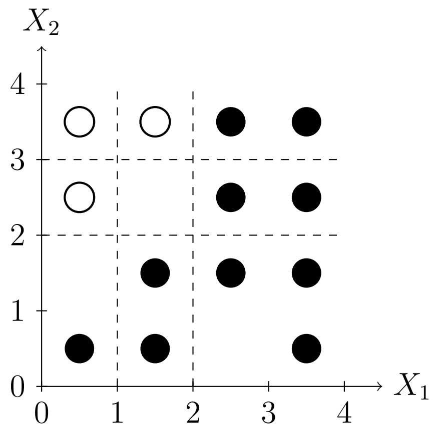
Y	X_1	X_2	$\|X - X^*\|$
0	-3	0	$\sqrt{10^2 + 1^2}$
1	-1	0	$\sqrt{10^2 + 1^2}$
1	1	0	$\sqrt{10^2 + 3^2}$
0	7.2	10	9.2

The point at [0, 7.2], [-3, 0] and [-1, 0] will be the closest. Hence, X^* will be classified as 0.

4. (a) A tree with two splits can attain zero training error (which is clearly optimal), e.g.,

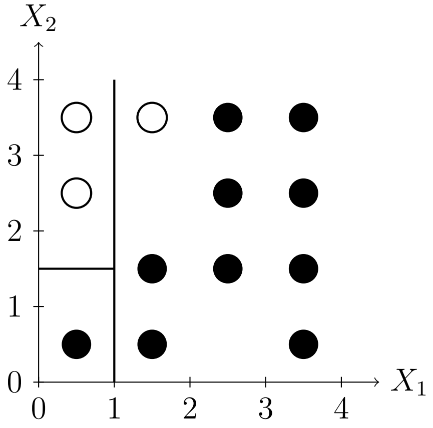


(b) For the first split, there are four candidate splits, at $X_1 = 1$, $X_1 = 2$, $X_2 = 2$ or $X_2 = 3$, sketched as dashed lines below (other splits are also possible, but clearly suboptimal):



The misclassification error for these candidate splits are, respectively, 2, 3, 3, 3. Thus the minimum misclassification error is obtained by splitting at $X_1 = 1$, i.e. we create the two regions $R_1 = \{X : X_1 \leq 1\}$ and $R_2 = \{X : X_1 > 1\}$.

For the second split, a similar argument shows that the optimal split is to split region R_1 at $X_2 = 1.5$, resulting in the following partitioning of the input space:



The misclassification training error of the resulting tree is 1, and the suboptimality of the solution comes from the fact that recursive binary splitting is a greedy procedure. That is, we do not take the second split into account when deciding how to make the first split.

(c) For the two candidate split points we get (where 0 correspond to the circles and 1 correspond to the solid discs)

	N_1	$\hat{p}_{10}$	$\hat{p}_{11}$	N_2	$\hat{p}_{20}$	$\hat{p}_{21}$	Q_1	Q_2	$C(T)$
i	3	2/3	1/3	10	1/10	9/10	0.444	0.18	3.13
ii	6	1/2	1/2	7	0	1	1/2	0	3

Thus, the Gini index prefers the second option, to split at $X_1 = 2$ (which is in fact the optimal split among all possibilities according to the Gini index). The benefit with this is that it produces one pure node (region R_2 consists only of solid discs), which can be beneficial if we intend to further grow the tree, as the region R_1 then will be split further. Indeed, for this toy example a tree with two splits which is grown using the Gini index will result in an optimal tree with zero misclassification loss. Compared to using misclassification error as a splitting criterion, the Gini index has a tendency to favor pure nodes.

5. (a) In K-fold cross validation the train data is divided into K equally sized sub sets. K different instances of the same model are trained. The kth instance is trained on all training data except the kth subset of the training data. Each model instance is then validated on the subset of data that was not used for training for that particular instance. The validation results for the different instances are then averaged. This procedure gives an indication of how well the model generalize to new unseen data (even tough it is not necessary a good approximation of the generalization error).

   K-fold cross validation is used to select between competing models, carry out input selection, and tune hyperparameters.

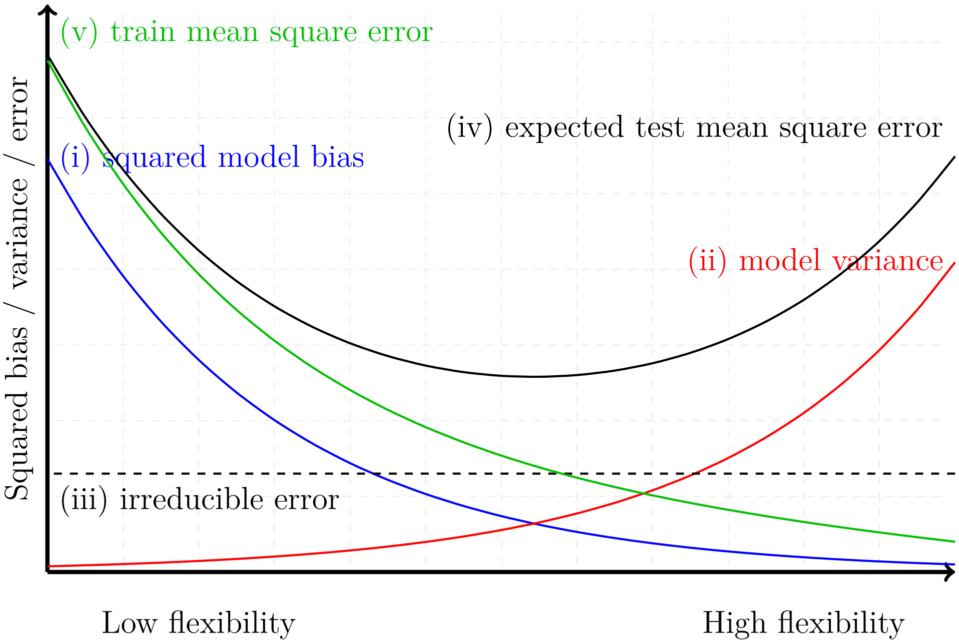
   (b) i. The model bias can be thought of as the error caused by the simplifying assumptions built into the model. Therefore, a model with low flexibility has a high bias and the bias decreases as the flexibility of the model increases.

   ii. 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.

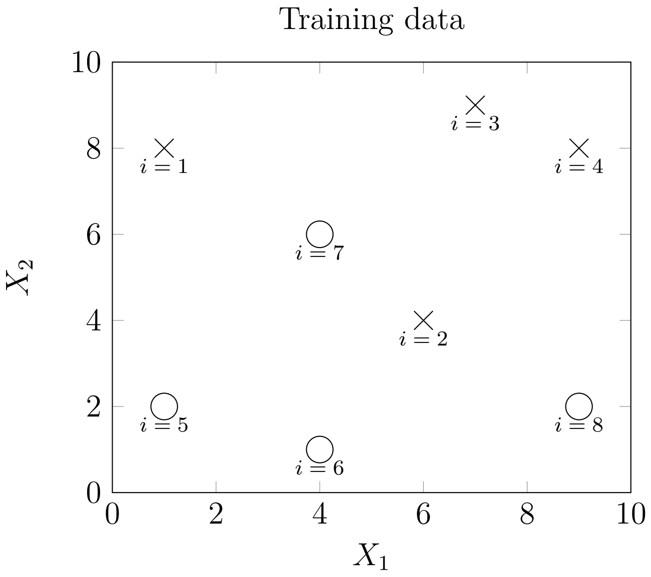
   iii. The irreducible error is the error term that has an impact on the output but is not explainable through the input variables. This error term cannot be reduced even though we would have a very good model trained on an infinite amount of data. Specifically, the irreducible error does not depend on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

   iv. 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.

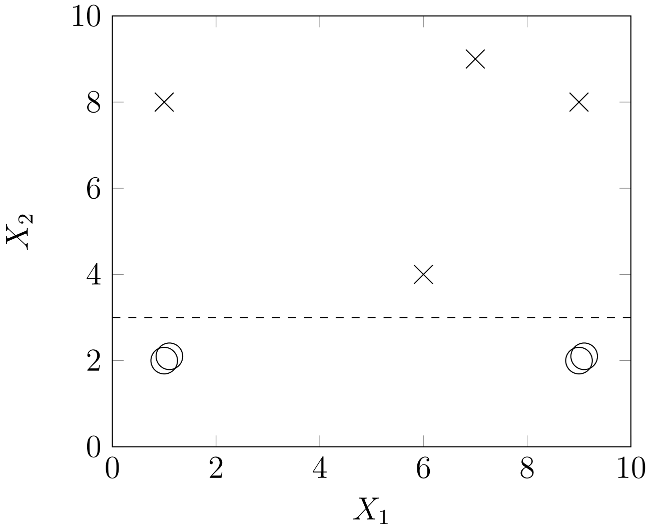
   v. The expected test mean-squared error is the expected mean-squared error of for a new unseen data point. This is a measure of the generalization performance of the model. The expected test mean-squared error is the sum of the squared model bias, the model variance and the irreducible error. Hence, 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.



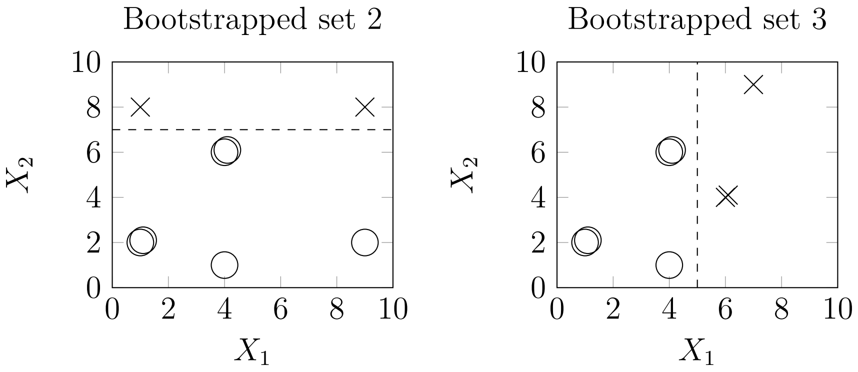
6. (a) The training data is illustrated in the following plot:



(b) Bootstraped dataset 1 looks as

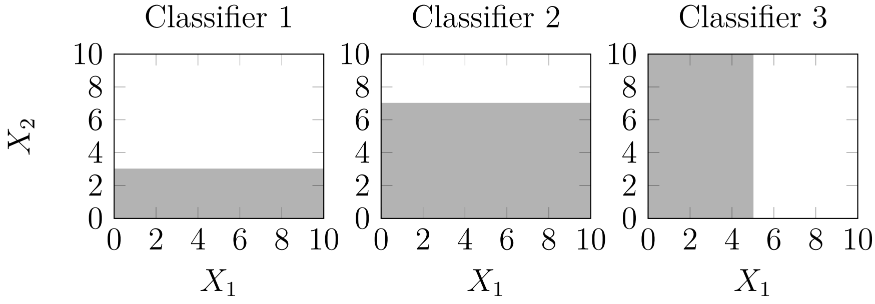


Only splits at $X_2 = r$ where $2 < r < 4$ gives zero missclassification error. We choose to split at $X_2 = 3$. Similar plots for bootstrapped dataset 2 and 3 are given below:

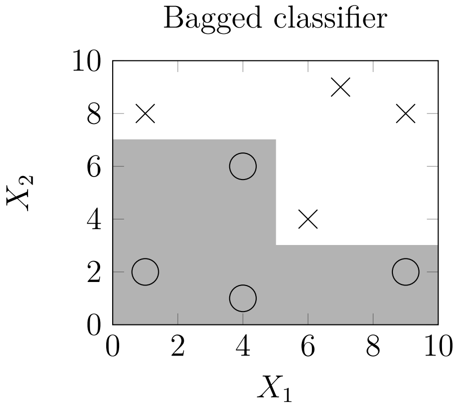


For the second dataset only splits at $X_2 = r$, $6 < r < 8$ gives zero misclassification error. We choose to split at $X_2 = 7$. For the third dataset only splits at $X_1 = r$, $4 < r < 6$ gives zero misclassification error. We choose to split at $X_1 = 5$.

(c) The decision boundary for each classifier becomes (gray: $Y = 0$ (circle), white: $Y = 1$ (cross))



which, with a majority vote, gives the final decision boundary



Note that in contrast to each ensemble member, the final bagged classifier manages to classify all data points correctly.