SKETCH of the SOLUTIONS
STK2100 - spring 2024

Exercise 1 kNN

a The training error is always 0 because the estimate of a point is the point itself. It is the most extreme case of overfitting: the model fits perfectly the training data, but it is not really generalisable, as it does not only model the structural part of the relationship between response and covariate(s), but also the noise.

b The curse of dimensionality is the definition of that phenomenon in which increasing the number of dimensions a method's performance declines substantially. It mainly affects non-parametric approaches, as they highly rely on the empirical distribution of the data. The formula, that describes the median distance between the point of interest (here the origin) and the first training point helps to understand why it happens: increasing the number of dimensions p on the right hand side results in an exponential increase in the distance (the argument $(1 - (1/2)^{(1/n)})$ is smaller than 1, so a decrease in the exponent $1/p$ increases the values). Since the estimation is based on points more and more distant form the one of interest, the estimate declines. Note that increasing the number of observations helps in keeping the distance small, but one needs a lot of observations to keep the distance comparable when the number of dimensions increases.

c The 0.632 bootstrap approach is a version of bootstrap in which the bootstrap error is mixed with the training error in order to have a better estimate of the test error. It is known, indeed, that the former (bootstrap error) tends to overestimate the test error, while the latter (training error) tends to underestimate it. The weight 0.632 comes from the probability of having a single observation in the bootstrap sample.

In the case of small k in KNN, the 0.632 bootstrap may not perform well because the overfitting is so high that the training error is too small and the method ends up to underestimate the test error. In order to solve the issue, one can implement 0.632+ bootstrap (NOTE: 0.632+ is only one possibility, other suggestions were accepted in the correction).

Exercise 2 Regression

a As in the textbook,

$$\begin{aligned} E_{\mathcal{P}}[Y - f^*(x)]^2 &= E_{\mathcal{P}}[Y - f_{\text{ag}}(x) + f_{\text{ag}}(x) - f^*(x)]^2 \\ &= E_{\mathcal{P}}[Y - f_{\text{ag}}(x)]^2 + E_{\mathcal{P}}[f^*(x) - f_{\text{ag}}(x)]^2 \\ &\geqslant E_{\mathcal{P}}[Y - f_{\text{ag}}(x)]^2. \end{aligned}$$

The extra error on the right-hand side comes from the variance of $f^*(x)$ around its mean $f_{\text{ag}}(x)$. Therefore true population aggregation never increases mean squared error.

b For a large number of times B:

1. Generate a bootstrap sample;
2. Fit a tree on the bootstrapped data, each time repeating:
   (a) Select randomly $m \leqslant p$ covariates;
   (b) Split the node into two child nodes using the best covariate/split-point among the m possibilities;

   until a stopping criterion (e.g., k observations per node) is reached.

Aggregate the results of all trees.

The passage that helps improving the performance with respect to bagging is the 2.(a), where a subset of the available covariates is chosen before performing each tree split. This helps reducing the correlation.

c When the number of relevant variables remains constant and the number of noise variables increases, at each tree split there is a higher chance that only noise covariates are selected among the $m < p$ variables. Therefore, increasing the number of noise covariates means that there is a higher chance that the trees that form the random forest are in large part based on pure noise.

The probability of selecting a relevant variable among the $m = \sqrt{p}$ ones is based on the hypergeometric distribution. In particular, the probability to select a relevant covariate is 1 - the probability of selecting only noise ones. Therefore:

(2, 5) $p = 2 + 5 \rightarrow m = \lfloor\sqrt{7}\rfloor = 2$, and $p = 1 - \frac{\binom{2}{0}\binom{5}{2}}{\binom{7}{2}} \approx 0.52$;

(2, 25) $p = 2 + 25 \rightarrow m = \lfloor\sqrt{27}\rfloor = 5$, and $p = 1 - \frac{\binom{2}{0}\binom{25}{5}}{\binom{27}{5}} \approx 0.34$;

(2, 50) $p = 2 + 50 \rightarrow m = \lfloor\sqrt{52}\rfloor = 7$, and $p = 1 - \frac{\binom{2}{0}\binom{50}{7}}{\binom{53}{7}} \approx 0.25$;

(2, 100) $p = 2 + 100 \to m = \lfloor\sqrt{102}\rfloor = 10$, and $p = 1 - \frac{\binom{2}{0}\binom{10}{2}}{\binom{100}{10}} \approx 0.19$;

(2, 150) $p = 2 + 150 \to m = \lfloor\sqrt{152}\rfloor = 12$, and $p = 1 - \frac{\binom{2}{0}\binom{150}{12}}{\binom{152}{12}} \approx 0.15$.

Exercise 3 Classification

a The probability of getting the disease is

$$\pi = \frac{\exp\{\beta_0 + \beta_1 \times \text{pregnant} + \beta_2 \times \text{glucose}\}}{1 + \exp\{\beta_0 + \beta_1 \times \text{pregnant} + \beta_2 \times \text{glucose}\}}$$

If we insert `pregnant` = 2 and `glucose` = 160, knowing that $\beta_0 \approx -5.751$, $\beta_1 \approx 0.123$, and $\beta_2 \approx 0.037$, then

$$\pi = \frac{\exp\{-5.751 + 0.123 \times 2 + 0.037 \times 160\}}{1 + \exp\{-5.751 + 0.123 \times 2 + 0.037 \times 160\}} \approx 0.602$$

Setting $\pi = 0.5$ and `glucose` $= x$,

$$\begin{aligned} 0.5 &\geqslant \frac{\exp\{-5.751 + 0.246 + 0.037x\}}{1 + \exp\{-5.751 + 0.246 + 0.037x\}} \\ \log 1 &\geqslant -5.505 + 0.037x \\ 5.505 &\geqslant 0.037x \\ x &\leqslant 148.78 \end{aligned}$$

b The fact that `pressure` is not significant in the sparser model but is significant in the more complex is related to the correlation between the variables. In this particular example it seems that it is correlated to `triceps`: these two covariates may provide similar information, that for some reasons in model B is captured by `triceps`, in model C by `pressure`. Another reason may be that `pressure` becomes significant to “balance” (interaction) the effect of `pedigree`, that has a strong effect (NOTE: surely the reason is NOT that `pressure` is correlated to a covariate present in C and not in B, otherwise it would be the other way around, namely `pressure` significant in model B, not significant in model C).

From the figure, we can say that the best model is model `C`, has it has the largest value of the area under the curve, i.e., it is the farthest from the random guess. The model is fine, but there it seems space for improvements, as the curve in the figure is far from the top left corner, where the optimal solution is. For example, from the R output it seems that model C contains variables that are not relevant, and may be therefore removed from the model.

c The prediction is “positive” in this case as well: it is a result of the path “right” ($180 > 127.5$), “left” ($28 < 29.95$), “right” ($180 > 145.5$).

The most left and most right splits have both children with the same prediction because in both cases the estimated probabilities to be positive are smaller (left case) or larger (right case) of the threshold used to split between positive and negative (most probably 0.5), but different from each other. The tree, indeed, provides an estimate of the probability, not directly the response.

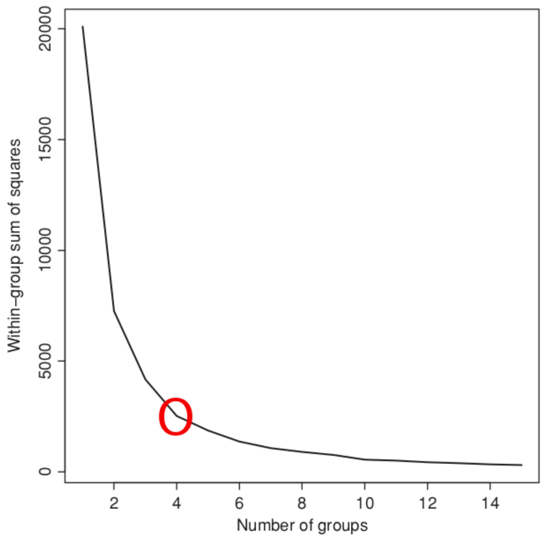
Exercise 4 Clustering

a The K-means method is a clustering approach based on the Euclidean distance. It groups the statistical units $x_1, \dots, x_n$ into K clusters, where K is a pre-specified number, based on the distance to the clusters means m_k. It performs the clustering by applying iteratively two steps:

- allocation: each statistical units x_i, $i = 1, \dots, n$ is allocated to the group k by finding $k = \text{argmin}_k ||x_i - m_k||_2^2$;
- update: each m_k, $k = 1, \dots, K$ is updated as the arithmetic means of all statistical units belonging to the cluster k;

The procedure is repeated until the results stabilise.

If the value of K is not known in advance, it can be found through the so-called “elbow-rule”: the final discrepancy is plotted against the number of clusters, and the best number of clusters K is chosen as the largest value for which there is a noticeable decrease in the discrepancy.


Within−group sum of squares
0
5000
10000
15000
20000
2
4
6
8
10
12
14
Number of groups
O