The Elements of Machine Learning, WS 2020/2021
Prof. Dr. Jilles Vreeken and Prof. Dr. Isabel Valera
FINAL EXAM, FEBRUARY 25, 2021, SOLUTION SHEET

PROBLEM 1 (ERRORS, ERRORS EVERYWHERE) (10 points)

(a) Consider Figure 1. Which of the following three options correctly describes what is happening in the figure. (1 point)

i) As we increase flexibility, variance starts to increase. The bias decreases more rapidly than the increase in variance, hence causing a downward trend until *Flexibility* = 6. After that, the decrease in bias becomes smaller than the increase in variance, resulting in the upward trend in the curve.

ii) As we increase flexibility, bias starts to increase. The variance decreases more rapidly than the increase bias, hence causing a downward trend until *Flexibility* = 6. After that, the decrease in variance becomes smaller than the increase in bias, resulting in the upward trend in the curve.

iii) As we increase flexibility, both bias and variance decrease until *Flexibility* = 6. After that, bias approaches zero, but variance starts to increase due to over fitting, thereby causing the overall rising trend in the curve.

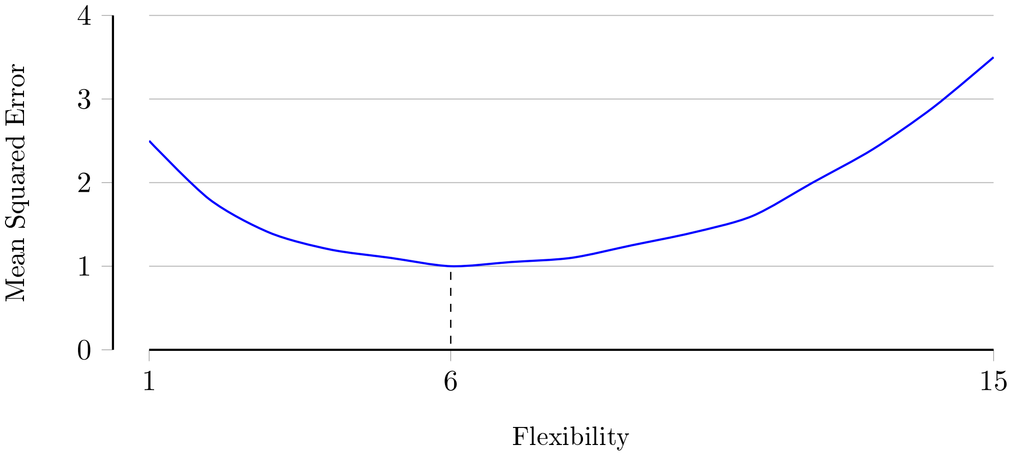


Figure 1: Test MSE for an un unknown data set.

(b) Explain, for each of the three settings below, what will happen in terms of bias and variance when we make the proposed change to the learning procedure. (3 points)

1) Changing the maximum depth of a decision tree from 10 to 2,

2) Replacing the LDA classifier with the QDA classifier,

3) When fitting a hexa-spline (i.e. polynomial spline of degree 6) enforcing continuity up to the 2nd, instead of up to the 5th derivative.

(c) Consider that we have infinite patience and training data. Is it *in general* possible to achieve the perfect predictor that obtains 0 test error? Explain why (not). (1 point)

(d) Rank the following approaches from the one that over-estimates generalization error *least* to the one that over-estimates generalization error *most*. Explain why your first ranked method over-estimates less than the second ranked method, the second ranked method less than the third ranked method, and so on. (3 points)

- validation set,
- leave-one-out cross-validation (LOOCV),
- k-fold cross-validation (CV),
- bagging.

(e) Explain *in your own words* what a confidence interval is, what a prediction interval is, and how these two are different. (2 points)

Solution.

(a) The correct answer is i). This is the bias-variance trade-off. Typically increased model complexity results in a decrease in bias and and increase in variance. At some point the bias cannot be reduced any further since it cannot drop below 0. There is however no fixed ceiling for the variance, which is why the prediction error increases again after a certain flexibility.

(b) 1) Reducing the depth and in turn the number of splits results in decreased flexibility, which means the bias goes up and variance down.

2) LDA assumes a common covariance matrix for all classes, whereas QDA assumes that each class has its own covariance matrix. Hence, bias goes down and variance up.

3) Enforcing continuity for fewer derivatives allows for a less smooth fit and hence more flexibility, which results in bias going down and variance up.

(c) This is not possible due to irreducible error. In the *general* case the response is not perfectly determined by the available and measured predictors that make up the data. This can be the result of measuring inaccuracies and unmeasured (or immeasurable) influences. Hence 0 test error cannot be achieved.

(d) 1) validation set: independent assessment, no bias on training data, variance only dependent on size of validation set.

2) k-fold cross-validation (CV) – test on all data points once, so variance is relatively low, but large overlap in training data causes bias.

3) k-bagging – random resampling version of CV, might test on all data points once, but large expected overlap in training data causes higher bias.

4) leave-one-out cross-validation (LOOCV) – tests on all data points once, but extreme overlap in training data causes higher bias than k-CV.

(e) Both confidence and prediction intervals quantify uncertainty. A k% confidence interval is the range around the parameter value estimated on the sample, for which we are k% confident that it contains the true parameter value for the entire population (e.g., uncertainty about the *average* sales over *all* cities). That means if we compute this interval for many sampled datasets we expect k% of those intervals to contain the true value. Given infinite samples the interval will converge to this true value.

A k% prediction interval is the range around the prediction for specific sample, for which we are k% confident that it contains the true response (e.g., uncertainty about sales in *a particular* city) . That means if we compute this interval for many datasets we expect k% of those intervals to contain the true response for that sample. Prediction intervals are wider than confidence intervals because they take into account the error in our estimate of the underlying function $f(X)$, and the uncertainty introduced due to the irreducible error, ϵ. Therefore the prediction interval cannot converge to a single value even with infinite samples size (ref. p. 82 ISLR).

PROBLEM 2 (REGRESSION) **(10 points)**

Please assist a group of experts with analyzing the 5 data points they obtained through a highly expensive experiment involving deep quantum entanglement and other buzzwords. The key objective is to predict response variable Y given one or more predictors X.

Expert 1 is convinced that X_1 is the key to predicting Y, and asks you to analyze the data in Table 1 using linear regression.

X_1	Y
2.3	15
2.7	14
3.8	16
3.9	15
4.6	24

Table 1: Observations for predictor variable X_1 and target variable Y.

Recall that simple linear regression takes the form $Y = \beta_1 \mathbf{X_1} + \beta_0$, but that it is often convenient to formulate it as $\mathbf{X}\beta = \mathbf{Y}$ with $\beta = \begin{bmatrix}\beta_1 \\ \beta_0\end{bmatrix}$ and $\mathbf{X} = [\mathbf{X_1}; \mathbb{1}]$.

(a) Using the following convenient approximation,

$$\left(\mathbf{X}^{\mathbf{T}}\,\mathbf{X}\right)^{-1} = \begin{bmatrix}0.3 & -1 \\ -1 & 3.6\end{bmatrix},$$

find β_1 and β_0. Explain the *reasoning* behind each step. (3 points)

(b) Expert 1 insists on using an unbiased linear estimator. Using the exact $\left(\mathbf{X}^{\mathbf{T}}\,\mathbf{X}\right)^{-1}$ under which conditions can you guarantee that the estimated β_0 and β_1 will have the smallest Standard Error? (1 point)

Expert 2 does not like X_1 at all, and instead claims that predictor X_2 is linearly related with the response variable. As evidence they show you the plot given in Figure 2 on page 5.

(c) State the common name of this plot, and describe what it is useful for. (1 point)

(d) Is Expert 2 correct in their claim? Explain why (not). (1 point)

Expert 3 is more inclusive and considers it possible that not just X_1, or X_2, but rather that any or all of $X_1, X_2, \ldots, X_{42}$ are useful for predicting Y. They kindly provide data (not shown) over all 42 predictors for the same observations as given in Table 1.

(e) Describe why you can no longer use the same general approach to determine the linear regression coefficients as you could to help Expert 1. (1 point)

Finally, the head of research unit, Expert ∞, wants to know which from $X_1, \ldots, X_{42}$ are the most relevant predictors for Y and suggests that to find out you should use ridge regression. An intern points out that this approach may have its pitfalls.

(f) Give one reason in favor of using lasso over ridge regression, and another reason why to favor ridge regression over lasso. (2 points)

(g) Both ridge regression and the lasso require you to choose a value for λ. Describe how in this case you would choose a suitable value for this parameter. (1 point)

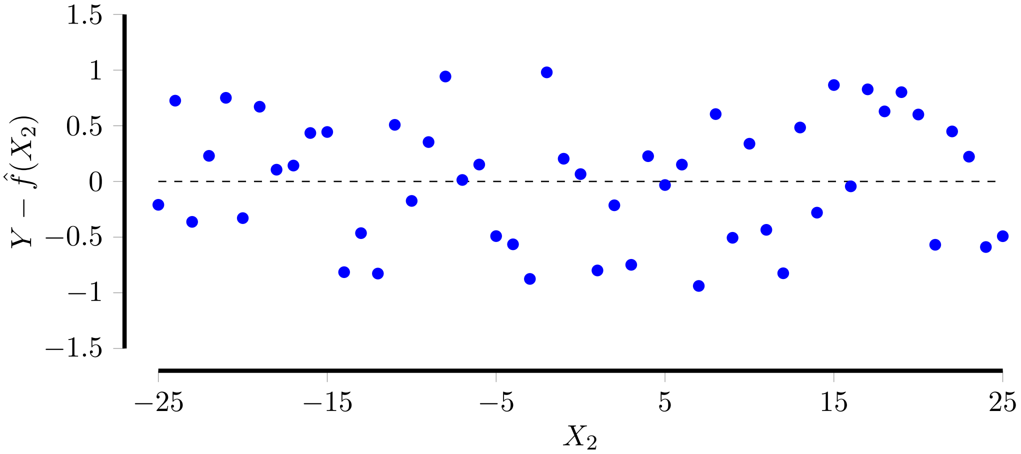


Figure 2: Plot given to you by Expert 2

Solution.

(a) (i) Matrix inversion is not possible if the matrix does not have full column rank, for example when there are more columns than rows in the matrix (i.e., more features than observations). (1 P)

(ii) $(\mathbf{X}^T\mathbf{X})^{-1}$ positive definite and invertible since $\mathbf{X}$ has full column rank, which means we can obtain a unique solution for β by formulating (1 P)

$$\begin{aligned}\mathbf{X}\beta &= Y\\ \mathbf{X}^T\mathbf{X}\beta &= \mathbf{X}^TY\\ \beta &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^TY\end{aligned}$$

(iii) We are given $(\mathbf{X}^T\mathbf{X})^{-1}$. Thus, we first multiply by $\mathbf{X}^T$:

$$\begin{aligned}&(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\\ &= \begin{bmatrix}0.3 & -1\\ -1 & 3.6\end{bmatrix}\begin{bmatrix}2.3 & 2.7 & 3.8 & 3.9 & 4.6\\ 1 & 1 & 1 & 1 & 1\end{bmatrix}\\ &= \begin{bmatrix}-0.31 & -0.19 & 0.14 & 0.17 & 0.38\\ 1.3 & 0.9 & -0.2 & -0.3 & -1\end{bmatrix}\end{aligned}$$

Now we multiply this matrix by Y to get:

$$\begin{aligned}&\left((\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\right)Y\\ &= \begin{bmatrix}-0.31 & -0.19 & 0.14 & 0.17 & 0.38\\ 1.3 & 0.9 & -0.2 & -0.3 & -1\end{bmatrix}\begin{bmatrix}15\\ 14\\ 16\\ 15\\ 24\end{bmatrix}\\ &= \begin{bmatrix}6.6\\ 0.4\end{bmatrix} = \begin{bmatrix}\beta_1\\ \beta_0\end{bmatrix}\end{aligned}$$

We allow for numerical errors in the final answer as long as the formulation is correct.

(b) The estimates for β_0 and β_1 would have the smallest standard error if the relationship between X_1 and Y follows the Gauss-Markov assumptions, i.e., the error terms in the additive linear model are random with expected value zero, homoscedastic and uncorrelated with each other.

(c) The plot is called *residual plot*. It plots the residual error (the difference between predicted and observed value) against the predicted values or predictor variable. We can use this to check for correctness of our modelling assumptions and problems in the data: non-zero-mean residual error (indicates biased model class), trends in residual distribution (indicates non-linearity), non-equal distribution of residuals over X (funnel or cone shape, heteroscedasticity of errors), outliers (can indicate deficiency in model or data), high-leverage points (inclusion/removal has large impact on regression line).

(d) They are correct. The residual errors are zero-centered over X and have no discernible trend, indicating the modelling assumptions are likely correct (for the given data). There is no obvious heteroscedasticity and there are no apparent outliers or high-leverage points.

(e) When $p > n$ the matrix is no longer invertible since it is no longer of full rank, meaning that the problem is underdetermined and has infinitely many solutions. Hence we cannot use the same technique as above to obtain estimates for β.

(f) Unlike Ridge Regression, Lasso drives coefficients to 0, meaning that some predictors drop out entirely. This makes it easier to interpret which predictors are informative for Y. One the other hand Ridge Regression will not drive coefficients of predictors that are highly correlated to 0 and hence allow us to recover equally-good predictors.

(g) We can find a suitable value for λ through cross-validation. Since it is a continuous variable we select a grid of possible λ values and compute the cross-validation error for each. Since there are so few points, LOOCV in particular is a suitable choice for the given data.

PROBLEM 3 (CLASSIFICATION) (10 points)

The research team now considers a classification problem in which they want to predict whether a cat in a box is alive ($Y = 1$) or not ($Y = 0$).

Expert 1 advocates they should use Decision Trees, as these permit interpretable models.

(a) When growing a decision tree, we iteratively split the n data points of a node t over two successor nodes t'_1 and t'_2. Show that the classification error of a decision tree never increases when we do so. For simplicity, you may consider the class label to be binary, i.e., $p(c_0) + p(c_1) = 1$. (2 points)

Expert 2 mumbles something about variance, and argues to use a Support Vector Machine instead. Recall that the Support Vector Machine is defined as follows.

$$\begin{aligned}
\underset{\beta_0,\ldots,\beta_p,\ \xi_1,\ldots,\xi_N}{\text{maximize}} \quad & M \\
\text{subject to} \quad & \|\beta\| = 1 \\
& \xi_i \geq 0 \\
& y_i f(x_i) \geq M(1 - \xi_i) \qquad \text{for } i = 1, \ldots, N \\
& \sum_{i=1}^{N} \xi_i \leq C
\end{aligned}$$

(b) Explain the purpose of variable C. Describe how bias and variance change when C is increased. (2 points)

Expert ∞ says they should not waste time with tweaking parameters and instead immediately go for the Bayes Classifier because it is *ideal*. The intern looks panicked.

(c) Give the Bayes Classifier. Explain in what sense it is ideal, and why we do not use it in practice so often. (1 point)

Expert 3 remarks that cats have nine lives, and that the classification problem hence involves not two, but rather $K = 10$ classes. Let $f_k(x)$ denote the density function of X, i.e $\Pr(X = x \mid Y = k)$, for the observation that comes from the kth class. Recall, according to *Bayes' Theorem*,

$$\Pr(Y = k \mid X = x) \ \propto \ \pi_k \cdot f_k(x) \ .$$

(d) What is π_k in the above equation? How would you calculate π_k for a given data set? (1 point)

(e) Assume that f_k is provided. How can you now use Bayes' theorem to predict the class label given the training data. (1 Point)

The intern figured that x is univariate and always positive. Moreover, they strongly suspect that it follows an exponential distribution $\mathcal{E}(\lambda_k)$ with distinct λ_k for each of the k classes, where

$$\mathcal{E}(x; \lambda_k) = \lambda_k \cdot e^{-\lambda_k x} .$$

(f) Derive the discriminant function. (2 points)

(g) Is the above derived discriminant function linear in terms of x? Why (not)? (1 point)

Solution.

(a) Assume that in node t there are n_0 items of class c_0 and n_1 items of class c_1. Further assume $n_1 \geq n_0$ so that n_0 elements are misclassified. After splitting, denote the children by t^0, t^1 and assume that there are k_0, k_1 elements of classes c_0, c_1 respectively in node t^0. We have to distinguish three different cases:

- $k_0 \geq k_1$. Then in t^0 we have k_1 misclassified items. Furthermore $n_1 - k_1 \geq n_0 - k_0$ so that in t^1 we misclassify $n_0 - k_0$ items. In total, $n_0 - k_0 + k_1 \leq n_0$ elements are misclassified.
- $k_1 > k_0$. Therefore in t^0 we have k^0 misclassified elements. Denote $d_n := n_1 - n_0, d_k := k_1 - k_0$. Then we have to distinguish the following cases:
  - $d_n \geq d_k$: Then $n_1 - k_1 \geq n_0 - k_0$ and thus in t^1 $n_0 - k_0$ elements are misclassified, for a total of n_0.
  - $d_k > d_n$: In this case $n_1 - k_1$ elements are misclassified in t^1. The total is therefore $n_1 - k_1 + k_0 = n_1 - d_k < n_1 - d_n = n_0$.

We note that the total number of misclassified elements does not increase in any of the cases.

(b) C specifies the total amount of slack, that is the total amount of distance we permit all points together to be within the margin or even on the wrong side of the hyperplane (misclassified training data). For a small C we seek narrow margins that a rarely violated and therefore fit the data tightly, which means low bias but high variance. If we allow no violations, by setting $C = 0$, we have the max-margin classifier, which only exists for linearly separable problems. Setting a larger C means we allow more violations, resulting in increased bias but reduced variance. For $\lim_{C\to\infty}$ the variance approaches zero.

(c) The Bayes Classifier is given by $\arg\max_k P(Y = k \mid X = x)$. It is ideal because it has the lowest possible error rate of all classifiers. However, we usually know neither the true $P(X \mid Y)$ nor $P(Y)$, and hence have to make assumptions to approximate these.

(d) π_k is the prior probability $P(Y = k)$ for observing an instance of class k. We can approximate $P(Y = k)$ by counting the instances of class k in our data and dividing it by the total number of instances in our data.

(e) Given that $P(Y = k \mid X = x) \propto \pi_k \cdot f_k(x)$. We plug-in our calculated π_k and the given $f_k(x)$ in to the equation of the Bayes Theorem. By taking $\arg\max_k$ of the product $\pi_k \cdot f_k(x)$ we can approximate the Bayes optimal decision.

(f) We derive the discriminant function $\text{argmax}_k \;\; p_k(x)$ by plugging in the given $f_k(x)$:

$$\begin{aligned}
\text{argmax}_k \;\; p_k(x) &= \text{argmax}_k \;\; \pi_k f_k(x) \\
&= \text{argmax}_k \;\; \pi_k \; \lambda_k e^{-\lambda_k x} \\
&= \text{argmax}_k \;\; \log(\pi_k) + \log\left(\lambda_k e^{-\lambda_k x}\right) \\
&= \text{argmax}_k \;\; \log(\pi_k) + \log(\lambda_k) + \log\left(e^{-\lambda_k x}\right) \\
&= \text{argmax}_k \;\; \log(\pi_k) + \log(\lambda_k) - \lambda_{\mathbf{k}}\mathbf{x}
\end{aligned}$$

(g) This classifier is linear in terms of x because x only appears in the linear expression $\lambda_{\mathbf{k}}\mathbf{x}$.

PROBLEM 4 (BEYOND LINEAR REGRESSION) **(10 points)**

The cat escaped from the box. The research team is now investigating how the size of a pet (X_1) relates to how loud it is (Y) as measured in decibel. Prior analysis shows that this relationship is non-linear.

Expert 1 suggests to model the relationship between X_1 and Y using regression splines, as these allow us to easily check and/or control the degrees of freedom of the model.

(a) How many degrees of freedom does a regression spline have if we use polynomials of degree $d = 4$, have $K = 10$ knots, and require the spline to be continuous at the knots, but do not care about the continuity of the derivatives. Explain your answer. (1 point)

Expert ∞ proclaims that to improve generalization they should use *unnatural* cubic splines. These are plain cubic splines with polynomials of degree $d = 10$ at the boundaries, where at each knot we enforce continuity up to and including the second derivative.

(b) How many degrees of freedom has an unnatural cubic spline with $K = 10$ knots? (1 point)

(c) Will the unnatural cubic regression spline achieve better generalization than a regular cubic spline? Explain why (not)? (2 points)

Expert 2 suggests they should use local regression using a uniform weight function (kernel) over the k points closest to the query point x_0.

(d) In the worst case, how many different local models would we have to fit if we have n training points and are asked to make m independent predictions? (2 points)

Expert 3 tells the intern that PCA is just linear regression, and that PLS and gradient boosting have nothing to do with one another. The intern looks doubtful.

(e) Suppose we are given a zero centered dataset over X and Y with $\text{Var}(X) = \text{Var}(Y) = 1$. We fit a linear regression model from X to Y to obtain β_0 and β_1, respectively perform PCA over X and Y to obtain the first principle component Z_1. When we now compare the directions of the vector $(1, \beta_1)$ to that of Z_1, we see that these directions are similar yet different. Explain why. (2 points)

(f) Explain how PLS and gradient boosted regression trees are similar. What is an advantage of PLS over boosting and advantage of boosting over PLS. (2 points)

Solution.

(a) Since there are $K = 10$ knots we have $K + 1 = 11$ regions. The polynomials we fit are of degree $d = 4$, which means they have $d + 1 = 5$ parameters each. This makes for $(K + 1)(d + 1) = 55$ parameters in total. By requiring the splines to be continuous we introduce one constraint per knot, for a total of $K = 10$ constraints. Hence the degree of freedoms are

$$(K + 1)(d + 1) - K = 55 - 10 = 45$$

(b) A plain regression spline with polynomials of degree d has $d + K + 1$ degrees of freedom. We replace the two cubic splines at the boundaries with polynomials of degree $d = 10$. This means we subtract the domains of freedom of two cubic splines, $2(3 + 1)$, and instead add those of two degree 10 polynomials, $2(10 + 1)$, hence the degrees of freedom are

$$3 + K + 1 - 2(3 + 1) + 2(10 + 1) = 28$$

(c) Splines already have high variance at the boundaries. By introducing more degrees of freedom, the variance will only increase further, and hence worsen the generalization. Natural splines, in order to address this variance, *reduce* the degrees of freedom at the boundaries by enforcing linearity.

(d) Assume x is univariate and $\forall i \; x_i < x_{i+1}$. Then $x_1, \dots, x_k$ is the first possible set of closest neighbors, $x_2, \dots, x_{k+1}$ the second one etc., until the last possible set $x_{n-k+1}, \dots, x_n$. This makes for a total of $n - k + 1$ different possible sets of closest neighbors to train the local model on.

Therefore, if $m <= n - k + 1$ we need to fit at most m local models, one for each prediction (fewer if some share a set of closest points). If $m > n - k + 1$ then we need $n - k + 1$ models since some predictions are based on the same local models.

(e) When maximizing the variance we minimize the sum of squared distances between the *projected* datapoint and the original datapoint, while when fitting a linear function we minimize the squared distance between the *predicted* datapoint and the original value.

(f) Boosting works by fitting a decision tree on the original data, computing the *residual* of this tree and training the next tree on that residual. This process is repeated B times.

PLS works by finding a first direction, computing the *residual* (the information not explained by the fist PLS direction) and computing the next direction based off that residual. This process is repeated M times.

Both methods iteratively improve performance by working off of the error remaining after "using" all previous predictors.

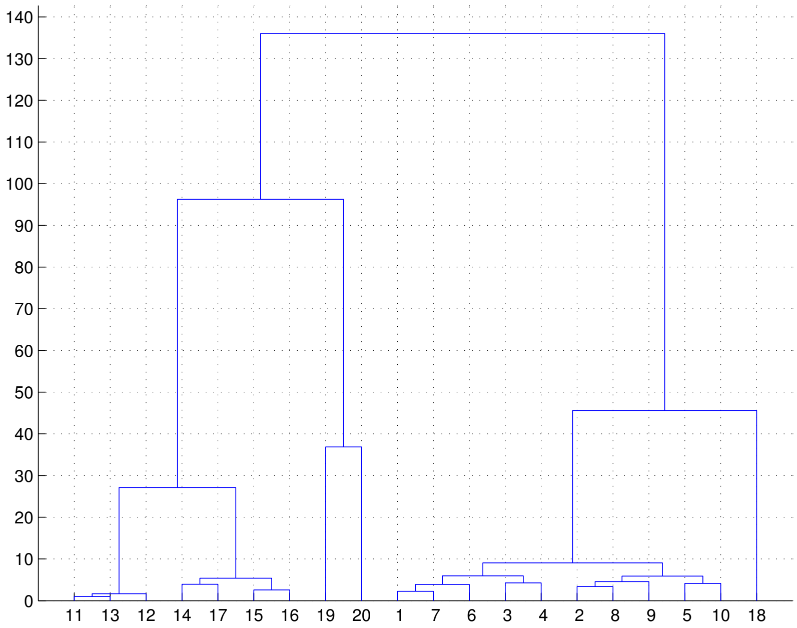
PLS can capture linear relationships while boosting can only approximate a linear relationship by a fine grained step function, but on the other hand this means boosting also works for non-linear relationships.

For Boosting we have to choose the number of splits per tree (d), the number of trees overall (B) and the shrinkage parameter (λ), whereas in PLS we only have pick the number of dimensions. Since decision trees are the basis of boosting, and its original version was applied to binary classification, it lends itself well for usage with categorical values.

PROBLEM 5 (UNSUPERVISED) **(10 points)**

Having tried and solved all possible supervised machine learning problems in their field, the experts now want to gain *insight* from their data and hence turn to unsupervised learning.

Expert 1 considers hierarchical clustering. Consider the following dendrogram.



(a) What is the clustering at $k = 3$? (1 point)

(b) Expert 1 expects that the data has 3 real clusters, and a number of possible outliers. What are the clusters and outliers? Explain your choice. (2 points)

(c) Explain the difference between single-link and complete-link hierarchical agglomerative clustering. Which weaknesses of single-link does complete-link address? (2 points)

Expert 2 does not like hierarchies, and hence instead considers k-means clustering.

(d) Show why the k-means algorithm always converges. (2 point)

(e) Is k-means is sensitive to outliers in the data? Explain why (not). (1 point)

Expert ∞ likes looking at things. While Stochastic Neighborhood Embedding (SNE) has a neat formal definition, its results tend to suffer from the 'crowding' problem. The intern suggests that changing the distribution in the lower dimensional space might solve that. Expert ∞ tells the intern to use the probability density function sketched in Figure 3.

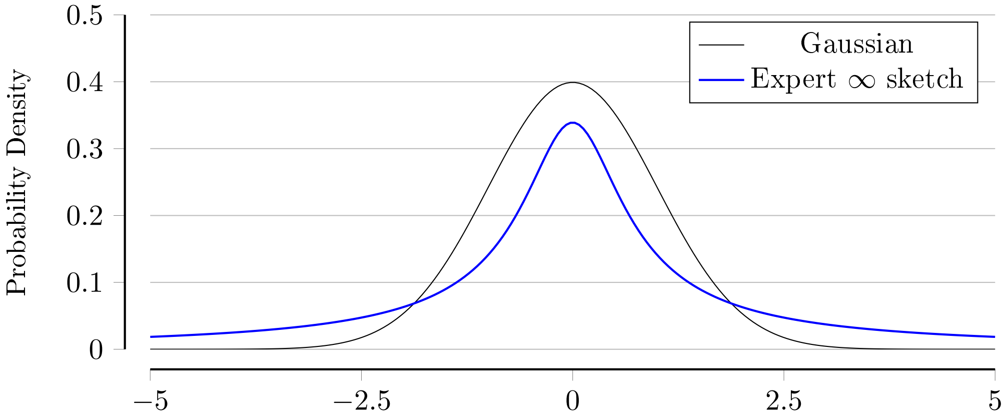


Figure 3: Probability Density Functions

(f) Consider Figure 3. Explain how and why the embeddings SNE discovers would change if we replace the Gaussian (Normal) distribution in the lower dimensional (map) space, with the sketched function. (2 points)

Solution.

1. We obtain cluster A $\{11-17\}$, Cluster B $\{19-20\}$, and Cluster C $\{1-10, 18\}$.
2. We obtain cluster A $\{11-13\}$, Cluster B $\{14-17\}$, and Cluster C $\{1-10\}$ with the outliers $\{18, 19, 20\}$. We choose the outliers such that they have the largest distance from other points, and pick the cluster such that we achieve the smallest intra-cluster distance (approximately 3 for A, 5 for B, and 9 for C). The alternative clustering $\{11-17\}, \{1, 3-4, 6-7\}, \{2, 5, 8-10\}$ would have distances of approximately 28, 6, and 6.
3. Single-link measures the distance between two clusters as the shortest distance between any pair of points drawn from the two clusters. This tends to create elongated clusters.

   Complete-link measures the distance between two clusters as the maximal distance between any pair of points drawn from the two clusters. This tends to create spherical clusters and breaks large clusters.

   Complete-link is less susceptible to noise than single-link.

4. Note that we have fewer than n^k possible clusters (for n observations and k clusters), that is we have a *finite* search space. Hence we can prove convergence by showing that subsequent states can form no cycles:

   In any iteration of the algorithm we have either:

   - The cluster assignment does not change and therefore the algorithm terminates.
   - The cluster assignment does change. Reassigning always decreases the error (see above), hence the new assignment cannot be one that has been visited in a previous iteration.

   Since the algorithm either terminates or visits a previously unseen assignment it is guaranteed to converge after at most n^k iterations.

5. The k-means algorithm is rather sensitive to outliers, due to the use of the mean as a centroid computation statistic, which is inherently sensitive to outliers. The effect may range from over-/under-estimating the true cluster center to selecting the outlier as a single cluster.

6. The steeper slopes near the peak mean that points that are close in the original space but far apart in the embedded space will be strongly penalized, whereas the lighter tails mean that points that are far away in the original space do not incur much penalization if they are close together in the embedded space. As a result, local structure in the original space will be emphasized in the embedded space: clusters will be more compact within, and further away from each other, than for the Gaussian kernel.