Exercise Set 3

- Submit the answer via Moodle at the latest on **Tuesday**, 12 December 2023, at 23:59.
- You can answer anonymously or write your name on the answer sheet; it is your choice.
- One person should complete the assignment, but discussions and joint-solving sessions with others are encouraged. Your final solution must, however, be your own. You are not allowed to copy ready-made solutions or solutions made by other students. You are permitted to use external sources, regular web searches included.
- You can discuss the problems in the exercise workshop.
- Your answer will be peer-reviewed by you and randomly selected other students.
- The language of the assignments is English.
- The submitted report should be in a single Portable Document Format (pdf) file.
- Answer the problems in the correct order.
- Read Moodle's general instructions and grading criteria before starting the problems.
- Main source material: ISLR_v2, Chapter 12. Please feel free to use other resources as well. ISLR_v2 refers to James, Witten, Hastie, and Tibshirani, 2021. An Introduction to Statistical Learning with Applications in R, 2nd edition. Springer.
- Notice that you can submit your answer to Moodle well before the deadline and revise it until the deadline. Therefore, please submit your solution in advance after you have solved some problems! Due to the peer review process, we cannot grant extensions to the deadline. Even though the Moodle submission may occasionally remain open after 23:59, the submission system will eventually close. If you try to submit your answers late, you will **not** get any points (including peer-review points) from this Exercise Set. You have been warned.
- Please double-check that the submitted pdf is appropriately formatted and, e.g., contains all figures. It is challenging to produce correctly formatted PDF files with Jupyter Notebooks: remember to check the created PDF. I recommend using R Markdown instead of Jupyter notebooks.
- You can solve the problems at your own pace. However, we have given a suggested schedule below. If you follow the plan, you can do the problems after we have covered the topics in class.

Suggested schedule

Please see the Moodle for the reading list and topics of the classes. Before the classes, read the respective textbook sections and only attempt to do the problems afterwards! Also, notice that the lab sections at the end of the textbook chapters contain helpful hints for many problems requiring programming.

As a general guideline, if you get stuck on a problem or a task, please do the other problems or tasks first and then return to the challenging problem. You can ask for help in Slack (channel #e3) or at the exercise workshops on 4 December and 11 December.

We recommend that you do the problems as follows:

After L9 clustering (29 Nov) and before the following lecture

Do problems 17–19. Submit your solution to Moodle; you can still update it until the deadline.

After L10 dimensionality reduction (1 Dec) and before 6 December

Do problem 20, write the learning diary (Problem 20), and submit your final solution to Moodle.

Default policy in the case of force majeure reasons

If you need a special arrangement due to a sudden illness or other force majeure reason, contact us as soon as possible. As a default policy, we assume that in such cases, you have followed the schedule recommended above up until you inform us of the force majeure reason. For example, if you contact us on the last day before the deadline, we assume you have had time to solve the problems. Therefore, please solve the problems in time and submit your answer early (you can update the answer until the deadline)!

Problem 17

[9 points]

Objectives: k-means loss and Lloyd's algorithm

In this problem, you will study the (naive) k-means algorithm (Lloyd's algorithm). You should be able to solve this problem with a pen and paper. See Section 12.4.1 of ISLR_v2.

Task a

Answer the following:

1. For what kinds of tasks can we use the k-means algorithm?
2. What are the algorithm's *inputs* and *outputs*?
3. How should you interpret the results?

Task b

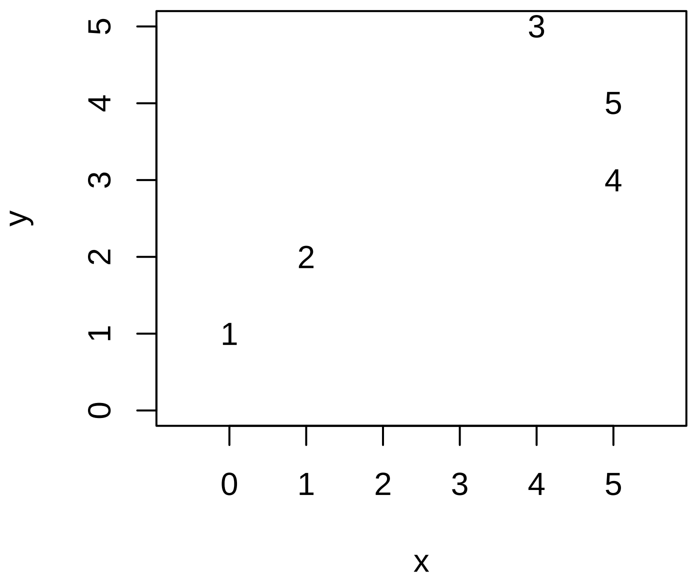
Define the objective (or cost) function the k-means algorithm tries to minimise.

What can you say about the objective function's value during the algorithm iteration?

Task c

Consider the following toy data set:

	1	2	3	4	5
x	0	1	4	5	5
y	1	2	5	3	4



Sketch a run of the (naive) k-means algorithm using $K = 2$ and initial prototype (mean) vectors $\mu_1 = (0, 2)$ and $\mu_2 = (2, 0)$. Write down the calculation procedure at each iteration and report the cluster memberships, the prototype vectors, and the value of the objective function at each iteration.

Problem 18

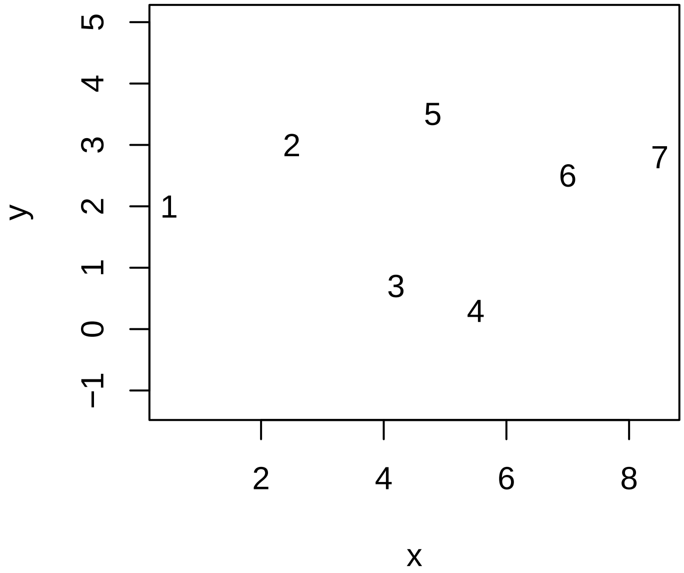
[9 points]

Objectives: Understanding hierarchical clustering algorithms

In this problem, you will apply *hierarchical clustering* on a toy data set. You should be able to solve this problem with a pen and paper. See Section 12.4.2 of ISLR_v2.

The toy data are shown below in a table and a figure.

	1	2	3	4	5	6	7
x	0.5	2.5	4.2	5.5	4.8	7.0	8.5
y	2.0	3.0	0.7	0.3	3.5	2.5	2.8



The table below shows the pairwise Euclidean distances between the data points.

	1	2	3	4	5	6	7
1	0.00	2.24	3.92	5.28	4.55	6.52	8.04
2	2.24	0.00	2.86	4.04	2.35	4.53	6.00
3	3.92	2.86	0.00	1.36	2.86	3.33	4.79
4	5.28	4.04	1.36	0.00	3.28	2.66	3.91
5	4.55	2.35	2.86	3.28	0.00	2.42	3.77
6	6.52	4.53	3.33	2.66	2.42	0.00	1.53
7	8.04	6.00	4.79	3.91	3.77	1.53	0.00

Task a

Sketch a run of the basic agglomerative hierarchical clustering algorithm using the *single link (min)* notion of dissimilarity between clusters.

Use the above figure in your answer sheet to indicate step-by-step how the algorithm forms clusters. You don't need to show detailed calculations but indicate the cost of the join you select.

Visualise the result as a dendrogram.

Task b

Repeat Task a using the *complete link (max)* dissimilarity. Compare the results.

Problem 19

[9 points]

Objectives: practical application of k-means and hierarchical clustering

For this problem, you will apply clustering on a subset of the project data (`mol.csv`). See Section 12.5.3 of ISLR_v2.

When clustering the data, omit the columns `parentspecies` and `pSat_Pa`, and scale the other variables to zero mean and unit variance unless otherwise instructed.

You should use library functions, such as `kmeans` in R and `sklearn.cluster.KMeans` in Python. Note: In R, `kmeans` uses the Hartigan-Wong algorithm by default, so set `algorithm="Lloyd"` to use Lloyd's algorithm. Python's `KMeans` uses kmeans++ as the default initialization, so set `init="random"` to use random initialisation in Task b.

Task a

Plot (or report in a table) the k-means loss as a function of the number of clusters from 1 to 20 for scaled and non-scaled data.

Should you scale the columns? How does scaling the columns affect the result?

Task b

The initialisation of k-means can affect the solution.

Cluster the data with k-means using $k = 5$ with 1000 different random initialisations, and then answer the following:

- What are the minimum and maximum k-means losses for your 1000 random initialisations?
- How many initialisations would you expect to have to obtain one reasonably good loss for this data set and number of clusters? A sufficiently good loss here is a solution with a loss within 1% of the best loss out of your 1000 losses.
- How do we deal with the effect of initialization when using k-means in practice?

Make a histogram of the losses.

Task c

(i) Cluster the data with agglomerative hierarchical clustering using single, complete, and Ward linkage. Produce their dendrograms side-by-side.

(ii) Find and report at least one interesting feature or reproduce some properties of hierarchical clustering discussed in the class or the textbook. For example, you can show differences between the linkage functions by comparing cluster sizes in different flat clusterings (`cutree` in R, `scipy.cluster.hierarchy.cut_tree` in Python).

Problem 20

[9 points]

Objectives: uses of PCA

In this problem, you will apply Principal Component Analysis (PCA) to `mol.csv`. See Sect. 12.2.4 of ISLR_v2.

When computing PCA on the data, omit the columns `parentspecies` and `pSat_Pa`, and scale the other variables to zero mean and unit variance unless otherwise instructed.

Task a

Compute and show a PCA projection of the data into two dimensions.

Indicate the parent species (column `parentspecies`) by the point's colour and shape. Remember to include a legend that indicates which colour/shape corresponds to which class.

What does the plot imply about the relationship between `parentspecies` and the other variables?

Task b

Plot (or report in a table) the proportion of variance explained (PVE), and the cumulative PVE for the principal components for scaled and unscaled data.

Why does it seem that fewer components explain a large proportion of the variance for unscaled data compared to the scaled data?

Task c

In this task, you will use ordinary least-squares linear regression to predict `log10(pSat_Pa)` using all variables except `parentspecies`. Use the random split of `mol.csv` in files `mol_train.csv` and `mol_validation.csv` we have provided.

(i) Train the model on the training set `mol_train.csv` (without dimensionality reduction) and report the RMSE on the validation set `mol_validation.csv`.

(ii) Repeat (i) after reducing the dimensionality with PCA for all dimensions $\{0, 1, \dots, p\}$, where p is the number of covariates, and report the RMSE values in a table or a plot. How does your model's performance vary with the (reduced) dimensionality? Is there an "optimal" dimensionality which gives you the best performance on the validation set?

Hint: the dimensionality zero means here that your covariates should consist only of the intercept term (i.e., you have no PCA components).

(iii) What is the smallest dimensionality that gives you a validation set RMSE that is at most 1% larger than the RMSE on dimensionality with the smallest RMSE? Argue why this dimensionality could be a better choice to learn a model than the "optimal" dimensionality you found in subtask (ii) above.

Tip: Notice that you can apply PCA on the combined training and validation sets to utilise the structure of the validation set even if you don't know the class labels; this is a simple form of *semi-supervised learning.*

Problem 21

[2 points]

Objectives: self-reflection, giving feedback on the course

Task a

Write a learning diary of the topics of lectures 9–10 and this exercise set.

Guiding questions: What did I learn? What did I not understand? Was there something relevant to other studies or (future) work? Your reply should be 1-3 paragraphs of text. You can also give feedback on the course.

Task b

Give an estimate of the hours used in solving the problems in this exercise set.