EXAM May 24 - June 7, 2024

The take-home exam comprises 2 questions with several subtasks each. Read through the entire exam before you start and think carefully about code-sharing between tasks to save time on execution.

It is a good strategy to attempt to answer all tasks since I will give partial marks. In addition, several of the tasks are interlinked.

Putting random results without motivation is not a good strategy. What I look for in your exams is a clear insight and motivation as to what you do to adress each task. If you do "random" things, or, even worse, "throw everything at the problem" and don't explain why you think this answers the questions, then this results in no marks. So please make sure you clearly state **why** you use a method/strategy/pipeline.

You should work alone for this exam. The reports will be checked with Urkund. You can discuss coding issues or bugs with other students *sparingly*, but don't discuss how to intepret figures or tables.

"Can I use ChatGPT?" It's been my experience that, while ChatGPT can be helpful, it is also overconfident in providing answers that are in fact wrong. It takes time to check the output from ChatGPT and, depending on your own skillset, this may result in an overall waste of time rather than a saving time, especially if you want to connect multiple tasks. Your main task is to analyze and *interpret* the results.

So I would say that you should use ChatGPT with caution and only for simple coding questions like generating plots, proposed functions calls (which may contain bugs), skeleton for coding pipelines, etc. Don't use ChatGPT for generating text.

GOOD LUCK!

- The report should not be longer than **1 page of text + max 1/2 page conclusion/main summary per sub-task (sub-tasks = a,b,c etc)**.
- You do not have to discuss the methods and assumptions *unless* this pertains to a result you present - i.e. you don't need to write a "lab report" with introduction, methods, analysis and conclusions. Here, it's only analysis and conclusions that I am interested in. However, **you may need to refer to assumptions to explain why your results look like they do**.
- Figures and Tables: **maximum of 4 figures and 2 tables per sub-task** (figures can contain panels but don't go overboard - I need to be able to read axes etc so tiny figures will not be appreciated).
- **Make sure you provide captions** for the tables and figures - these should clearly state what the figure/table contains as well as a 1-2 conclusion statement about what can be learnt from the figure/table. **Refer to the figures and tables in the text!**

- **MOTIVATE YOUR ANSWERS!**
- **Don't contradict your results!** If something is unexpected say so and explain why you thought you would see something different.
- Be careful about using buzzwords, guessing what I want you to write etc - just go directly from your results.
- Please think about your presentation. Spell check, grammar check. "The arm-length check": meaning, skim through the report as it appears on the screen - does it look nice and neat or is it messy with mis-alignments, cut-off sentences, plots that are tiny etc?
- **No cut and paste code mixed with the text.**
- No, jupyter notebooks are not accepted as a report. Preferably use latex (overleaf) for neat reports.
- Yes, **you can use bulletpoints to present your findings**. This could be especially useful for the subtask conclusion.
- Stress your main findings in the 1/2 page subtask conclusion.
- Make sure you put the figures with each subtask after the subtask discussion and *before* the next subtask so I don't have to scroll back and forth between tasks when I grade.

IMPORTANT

Wherever the exam task contains a question ("Does X differ from Y?", "Is A performing better than B") it is *not sufficient* to answer "Yes" or "No" - you need to explain how you arrived to this conclusion, which results provide you with these answers, as well as a statement trying explain *why* the results are what they are. Likewise, when a task is phrased as "Can you do X to get Y?" I, of course, mean for you to attempt to answer this question by doing something, not just hypothesizing about it.

To answer the questions it is for the most part inadequate to perform the task once on the full data set. Make use of re-sampling techniques to support your findings.

Data set 1

For the first part of the exam question you will use a data set with 1866 observations and 6 features. The data set is contained in a tab-del file called "Fish.txt" posted on canvas. The data set contains information about different species of fish, such as weight, height and width. It also contains length measures of three types called L1, L2 and L3 that measure length from the "nose" to the beginning of the tail (L1) blue line in image below), nose to the notch of the tail (L2, yellow) and full length (L3, red).

The last column of the data matrix contains a 7-class label vector (figure examples below are *not* representative of abolute length since they are rescaled to the same size, but gives you a sense of the the L1,L2,L3 relationship).

For all subtasks, make sure you motivate your answers and provide a take-home message summarizing your main conclusions.

Question 1 (50p)

1a (20p)

Train multiple classifiers (at least 6, of varying degrees of complexity and character (parametric, nonparametric, linear, nonlinear,...) on multiple training-test splits of the data. Compare classification performance.

Pay particular attention to the following;

- imbalance between classes, how you handle this
- classification accuracy, specificity and sensitivity overall, and at the class level.

Be careful about using the full data set for any training/validation steps. For example, exploration and dimension reduction, if you choose to pursue this approach, should be done of the training data only.

Make sure to discuss:

Overall performance and how you assess this

Are all or only some classes well separated

Class imbalance, modeling assumptions, training metrics

The full training pipeline you set up and motivate all your choices

1b (10p)

Which set of features are optimal classification performance? Does this differ for different classifiers in 1a? Does this differ between the classes?

Does this vary between observations, e.g., between correctly labeled and the mislabeled observations?

Use at least 3 different methods to answer the above questions. Discuss the stability of selection and the confidence you can attach to your estimate of the number of features needed or features selected, and how you assessed this.

Make sure to discuss:

Method comparisons

Interpretation of the results - why are these features important?

Be clear in how you attach confidence to your estimates

1c (10p)

This data set is low-dimensional. Turn it into a high-dimensional data set by adding simulated features to the data set.

(i) Perform a simulation study where you add more and more features that are unrelated to the class labels. How does this impact classification performance and the feature selection task?

(ii) Perform a simulation study where you add more and more features that are correlated with the original feautures. How does this impact classification performance and the feature selection task?

You can use a subset of classification methods (2-3) and feature selection techniques here (1-2) from 1a and 1b.

Make sure to discuss/investigate:

Classfication performance deterioration as a function of the proportion of noise features

Impact on feature selection

1d (10p)

Revist task 1(a) and consider if observations can be classified with confidence as belonging to one class or if set prediction is more appropriate for some observations.

Is there any evidence that there are mislabeled observations in the data set? Explain and demonstrate.

Make sure to discuss:

How you quantify how many observations can be classified with confidence, and which sets of predictions are allocated to which observations

If you could identify mislabeled observations in the original data

Data set 2

For this question you will use a variant of the MNIST digits data that I have created/manipulated in various ways. The data set is quite large with 50000 28*28 raster scan images. The first column of the data matrix is the label.

Remark: Because the data set is so large, you will have to think carefully how to approach the problem. There are R and python packages that provide big-N variants of the methods from class - you are free to use any package you like as long as you can explain what they do.

We have also talked about running models on batches of data, which you can do for both classification and clustering. For classification, you can aggregate the models from different subsets of data as any ensemble method. For clustering, this is a bit more tricky. To answer the question 2c you don't necessarily need to aggregate. If you still want to aggregate, one common approach is to cluster batches of data with a large number of clusters and then use cluster representatives to aggregate. Think kmeans with many clusters applied to each batch - a.k.a. local clustering. Take the centroids from each batch and cluster these - a.k.a. global clustering. Now you know how to assign cluster labels to all the data since the local clusters "inherit" the labels from their cluster centroid.

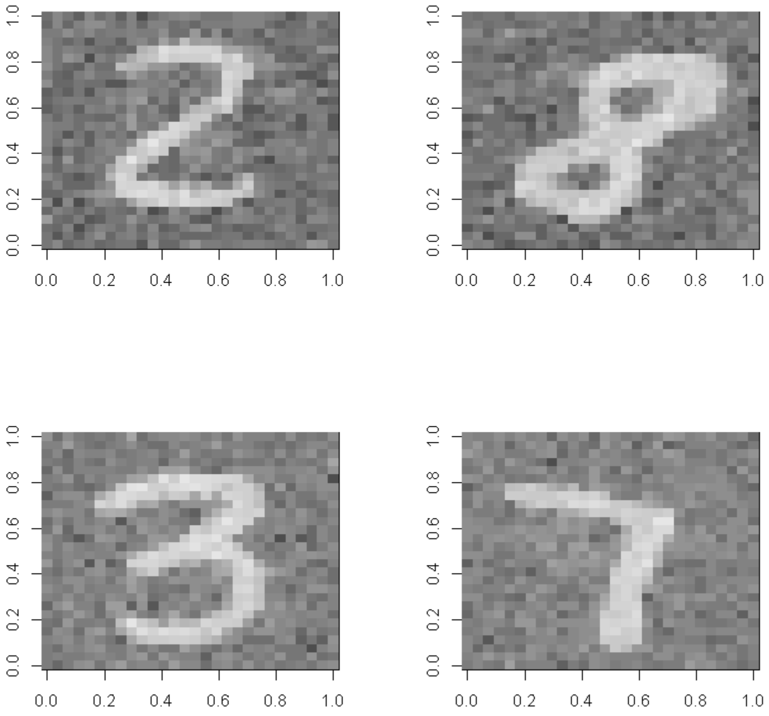
In the end, you are limited by the computing resources you have and you can obtain answers to the questions below without having to resort to high-performance computing.

In [1]:

```
MM<-read.csv("MyMNIST.csv")
```

In [2]:

```
par(mfrow=c(2,2))
ss<-sample(seq(1,50000),4)
for (kk in (1:length(ss))) {
  a<-t(matrix(as.numeric(MM[ss[kk],-1]),28,28,byrow=T))[,28:1]
  image(a,col=gray.colors(33)) }
```



Question 2 (50p)

2a(10p)

Use linear and nonlinear dimension reduction techniques (e.g. SVD, sparseSVD, kernelPCA, NMF, AE, tSNE) and filtering (testing) to explore the data set.

Make sure to discuss:

How the results differ between methods

How the dimension reduction techniques shed light on class separation, or lack thereof

How the dimension reduction techniques provide insight into the regions of the images that separate the classes

How you handled the large number of observations

2b(20p)

Investigate classification performance as a function of training sample size on this data set. Make sure to use a wide range of classification methods (at least 5) and discuss what aspects of the data makes some methods perform worse than others.

Make sure to discuss:

How you adressed the dimensionality of the data set

Specific challenges in classifying this data set, looking at class distributions, potential mislabeleing, noise level etc

Method comparisons

Discuss how sample size impacts classification performance and relative performance between the methods.

2c(20p)

Can clustering retrieve the all/some of the classes?

Investigate at least 3 clustering methods.

Make sure to discuss:

How you decide on the number of clusters

How you utilize the massive amounts of data available to you

Which classes are retrivable and which are not, and why you think this is the case