Introduction to Machine Learning

Monday, 8th March 2021, 09:30 - 17:00 (GMT)
Tuesday, 9th March 2021, 09:30 - 17:00 (GMT)
Wednesday, 10th March 2021, 09:30 - 17:00 (GMT)

Instructors:	Irina Mohorianu, Christopher Penfold, Adi Steif, Eleanor Williams, Manik Garg.
Helpers:	Soumya Banergee, Sergio Martinez Cuesta, Simon Koplev
Timetable:	https://training.csx.cam.ac.uk/bioinformatics/event/3732957

Day 1

Session 1	Machine learning and its applications in research
Session 2	Data types and partitioning
Session 3	Introduction to CARET, an R-based machine learning framework
Lunch break	*Around 1.00pm*
Session 4	Dimensionality Reduction
Session 5	Clustering
Session 6	Review and questions

Slides:
Introduction to Machine Learning
Machine Learning Data Partitioning

Day 2

Session 1	Nearest Neighbours
Session 2	Decision Trees and Random Forests
Session 3	Support Vector Machines
Lunch break	*Around 12.30pm*
Session 1	Exercises on Classifiers
Session 2	Use case applying the Day 2 methods
Session 3	Review and questions

Slides:

Day 3

Session 1	Linear models
Session 2	Logistic regression
Lunch break	*Around 1.00pm*
Session 3	Artificial Neural Networks
Session 4	Use case applying the above methods
Session 5	Review, questions and resources for further study

Slides:

Links

- **Please connect to the course via this Zoom link**: https://us02web.zoom.us/j/86288832934?pwd=NHdNdVRpU0NsU1pLNEpHWWNBWEIvdz09
- **Course notes website**: https://cambiotraining.github.io/intro-machine-learning/
- **Course Materials github**: https://github.com/cambiotraining/intro-machine-learning
- **Caret package:** https://topepo.github.io/caret/index.html
- **Course participant introductions**: https://docs.google.com/document/d/1_F4chNo48sxjMjReAKetcn4NU3u887kcjJR7W42hjTM/edit?usp=sharing
- **Course feedback survey**: https://www.surveymonkey.co.uk/r/DDR786L
  We would really appreciate it if you could share your thoughts with us regarding these sessions. We are interested in your opinions, how you feel the experience has benefited you and how it could be improved.
  If you could find a few minutes to complete a short survey at the end of the last session it would really help us in improving the training we can deliver.

Further reading:
An Introduction to Statistical Learning
Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani
https://www.dropbox.com/s/i26vfrmyszjwfyk/An%20Introduction%20To%20Statistical%20Learning%20with%20Applications%20in%20R%20%28ISLR%20Sixth%20Printing%29.pdf?dl=0

COVID-19 Paper https://www.frontiersin.org/articles/10.3389/fpubh.2020.00357/full
COVID-19 Data https://github.com/Atharva-Peshkar/Covid-19-Patient-Health-Analytics

Further actions:

- Sign up to our mailing list to get notifications of upcoming courses from the Bioinformatics Training Facility

Recordings

Recordings for each day's teaching will be posted here shortly after the zoom call closes. These recordings are intended for course attendees only for private study only. Please do not share these links. The links will continue to work for around a month after the course but you may download the videos for a permanent copy if you wish.

Day1

https://us02web.zoom.us/rec/share/Ebnd3e7t_OYdb3-TIbK_hoW0wrtWZct9JXjDVFxtKKsrm1ssCJOTqlYjbppis7HD.vbcduBXew0qroXbr Passcode: JKGf6+QE

Day2

https://us02web.zoom.us/rec/share/-fxY7qPyvB1i_LTpvorX3d2yOLbyhto0QpUi2eLIV_NO02vV_GA0QTFf3TTHhLOS.O21HkMBqq3Bqg20l Passcode: NYwJA!8?

Day3

https://us02web.zoom.us/rec/share/fc91MySshfJbBKeISb_milL3x8EPfSQwA7_IT7-LrhgcOs_wHGrCPYwykuAsyT0Q.nYi0sFaj8pp7GhBX Passcode: 5GvBqJ?*

Course netiquette

We are expecting a large number of participants in this session. We suggest everyone follows these few simple rules for the course to run as smoothly as possible.

Questions

If you have any questions/problems that you would like to share and are applicable to the whole class please write them below. A tutor will answer your question.

Write your question after the last one you can see in this document and write your name.

DAY 1

1. **<Helena><I didn't quite understand the circular graph showing the different approaches eg model, evaluation and optimisation - because I don't know what all the terms mena, is there somewhere i can read more about it?>**
   <Simon><The main point with the graph is that there are a variety of approaches in machine learning and statistics. It can be quite difficult, if not impossible, to have a full understanding of all of these. Yet there are useful similarities that one can use to think about them. For example, they normally all have (different) abstract representation and optimization methods and knowing this, it may be easier to understand what different methodologies aim to do and how the associated learning algorithms work.**>**
   <Eleanor> There is a little more information here (https://jrodthoughts.medium.com/the-five-tribes-of-machine-learning-c74d702e88da) about the different groups of ML algorithms. It will hopefully make more sense after this course so don't worry too much for now.

2. **<Wojciech><what is the difference between a hyperparameter and a "regular" parameter?>**
   <Simon><An example of a regular parameter is coefficients in a regression model, which one tries to optimize to best fit the response variable. In contrast, a hyperparameter determines something about the model that is fitted. For example,

the number and types of polynomial terms to include (x^2, x^3, etc). The point that we'll keep coming back to in the coming days is that often one would like to also optimize the hyperparameter, but one has to be careful in the separation of training and test data so as to not overfit this choice of hyperparameter. Other examples of hyperparameters include k in kth nearest neighbors or the architecture of feed forward artificial neural networks -- number of nodes/neurons, number of layers, and the activation function such as sigmoid or tanh.

Slightly more formally, and for parametric models only, the regular parameters can be thought of as making up a hypothesis function h(**θ**) where **θ** is a vector or regular parameters to be optimized. In the regression example, this would be the function one is trying to fit. The hyperparameters, in contrast, would then change the hypothesis space by altering this parametrization. In some ways it's a subtle difference but one which is made for practical reasons; typically the theory for how to fit the hypothesis function only works for the regular parameters.**>**

3. **<Wojciech><Can we also define what exactly preprocessing and dimensionality reduction are? These terms are widely used, but not always clearly defined>**
   <Eleanor><We will have full lectures about each of these topics today, preprocessing in the next lecture and dimensionality reduction straight after lunch. This current lecture is just giving an overview of ML and the different components and how they fit together but all the formal definitions will come in the following lectures so I will leave those for now but if you have any questions after the lectures then let us know!**>**

4. **Emma B><**I have genomic copy number profiles (which look like a series of hills and troughs) which I would like to assess the shape of in terms of a particular output (rather than particular peaks etc) - is this possible - is this maybe how the neighbour approach works? **>**
   <Irina>Trying the nearest neighbour approach could work if the data is labelled. Otherwise perhaps try a simulated annealing, or an approach based on genetic algorithms to identify optimums - what is the target question for your dataset, the identification of minimum and maximums or the selection and characterisation of patterns?
   Emma B - The selection and characterisation of patterns...

5. **<Wojtek><I would like to discuss the selection of the right ML methodology (e.g. NN, genetic algorithms, decision tree / forest etc.). Currently it's more of an art than science. Same applies to details of the selected methodology (e.g. the architecture of a NN). It strikes me as odd and suboptimal that we don't use some sort of preliminary AI methodology to help us choose (and subsequently fine-tune) them. Could we discuss this pls?>**
   <Soumya Banerjee><You raise a very good point. Firstly, the space of problems is quite large. For example, an algorithm that recommends clothes to buy is different from an algorithm that classifies clusters of galaxies. The data are quite different. And

one has to try different approaches before one can find the right one. You do raise a good point. There are some ML approaches that can automate some of these steps. For example, there is a package called *TPOT (https://github.com/EpistasisLab/tpot)*, that can automate data dimensionality reduction, creation of classifiers, etc. It picks the right classifier and picks the right dimensionality reduction (by essentially trying some intelligent combination of these). There is also *auto-keras* (https://autokeras.com/).This will fit the right deep learning/ANN to the data. However humans still have to spend some time with the data, cleaning it up, doing data munging, etc**. >**
<Simon**><**I think this is an excellent idea for a research topic in machine learning. Although some solutions do exist, like the ones Soumya mentions above, it is still largely an unsolved problem. Fortunately, this means that there is still work to be done -- an automatic statistician is most likely decades away. I suspect that the reason for this is that it's hard to generalize rules for well-performing approaches across data types and research fields. In addition, the space of available machine learning methods is quite large. Lastly, caveats of different methods often include aspects such as computational scalability and model interpretability, in addition to the predictive performance. Put differently, I think it's hard to capture and evaluate all these tradeoffs automatically, and for now we're stuck with human cognition often being the best solution. Yet it's interesting to think about. What kind of data would one need to train a machine learning model to predict the best approach to a given data set? Maybe a supervised formulation with the input being data (this would be difficult as different domains have very different types of data) or maybe even text descriptions of some sort and the labels/predictor variables being the best approaches?**>**

6. **<**Natalie Wallis**><Are the hyperparameters something you assign initially or create during the ML process? Bit confused by this term>**
<Soumya Banerjee**><**The hyperparameters term is a bit confusing. Essentially these are parameters that describe parameters. Hyperparameters are something that are estimated from the data. This will be discussed later. Hyperparameters are estimated in a process called cross-validation.
Expanding on the hyperparameter concept: Say a model has two parameters (slope of a line). A hyperparameter(s) can be the mean of the slope. **>**
<Natalie Wallis**> Thank you!**

7. **<**Natalie Wallis**><Do normalisation and scaling have distinct definitions? If so what>**
<Eleanor**><**Scaling is transforming your data to fit in a certain scale, for example so all points fit between 0 and 1. This means if you drew a density plot of your data before and after scaling, the shape would be exactly the same, just the x-axis range would be different. Normalisation makes a bigger change to your data by actually changing the distribution, for example to fit a normal distribution. You could be normalising to fit a particular distribution, or to make multiple bits of data comparable, for example different samples which you want to directly compare but have different distributions. You can look at scaling as a type of normalisation (the linear scaling

Irina showed is called min-max normalisation) but generally normalisation would mean you are making more of a change to the distribution. Other types of normalisation include Z-score normalisation (standardisation) and quantile normalisation.**>**

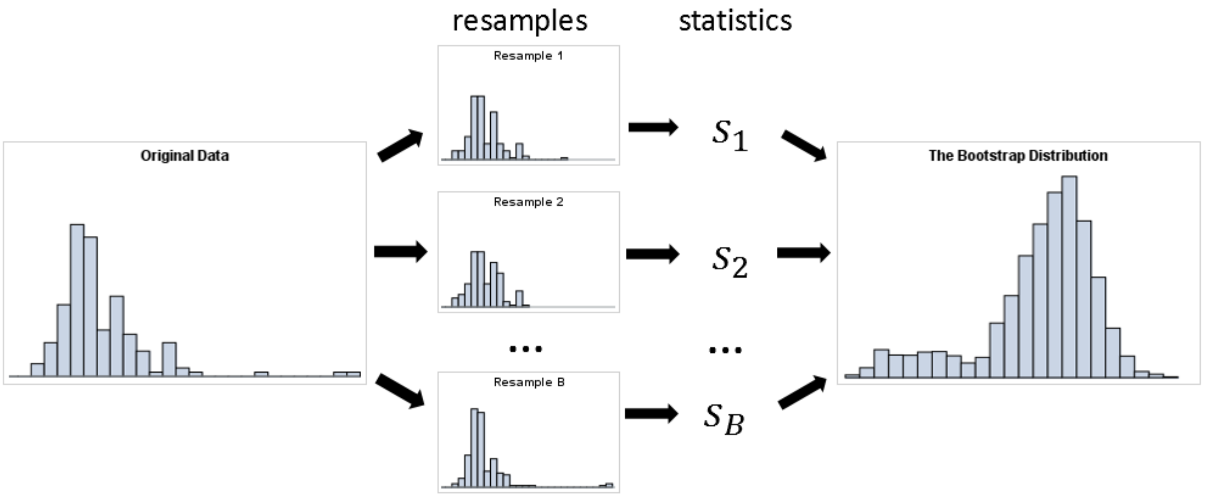
8. **<Livia><How would you accommodate for the test-training scedule when you have a small dataset and splitting in different sets might not be possible?>**
   <Eleanor><When you have a small dataset and can't afford to be using too much of the set for development/validation and test, you can use k-fold cross-validation or even leave-one-out-cross-validation (https://machinelearningmastery.com/loocv-for-evaluating-machine-learning-algorithms/). These are quite computationally expensive for large sets but for small sets, you can assess whether you are overfitting (which is a big risk with small datasets) by looking at the distributions of training and test accuracy. The general set-up here is that you use all but 1 of the examples (or 10% if 10-fold cross-validation) to train the model then test it on the remaining example(s). When you have small datasets, it is also a good idea to focus on more simple models and focus on fewer (better) features. Sometimes people also use synthetic samples but there are lots of dangers associated with that too. **>**

9. **<Bruno><May upweighting influence the splitting in training, validation and test datasets?>**
   <Simon><Upweighting shouldn't change the splitting of the data. But, it does affect the training of the model such that the underrepresented class gets a higher weight during training. The idea is that the resulting model should then perform better on the underrepresented class when predicting on unseen data -- as quantified when evaluating performance on the test data set.

   If trying to address the issue of class imbalance issue with upsampling or downsampling (rather than upweighting) then this should be done (only) on the training data set subsequently to splitting the data.

   In addition, it may make sense to change the model performance metric if you have class imbalance. Maybe you would like to invoke a higher cost on false negatives if your positive rate is very low and you really don't want to miss any positives. It can be tricky to clearly think about this and there's a topic known as statistical decision theory that may help. But, importantly, it's a different (but overlapping) issue than being able to fit your model well to a data set where one class of observations is underrepresented.**>**

10. **<John><Was bootstrapping defined?>**
    <Soumya Banerjee><Bootstrapping, simply put, is when you repeatedly sample from a distribution. Say you have just got data. And it does not look normal. Or you cannot assume it is normal. How do you determine things like mean, confidence intervals, etc?
    https://blogs.sas.com/content/iml/files/2018/12/bootstrapSummary.png



You essentially repeatedly sample from this distribution.
>Thanks
<Eleanor> <There is a nice summary of bootstrapping in section 5.2 in Introduction to Statistical Learning, linked above. This is also a nice walkthrough https://machinelearningmastery.com/a-gentle-introduction-to-the-bootstrap-method/.
>
Thank you

11. **<Name><Are the dev set and the validation set the same?>**
<Eleanor><Yes, here they are used interchangeably. The naming is very confusing and people use both versions across ML. You can usually tell from context what people mean but here they are both used to refer to the set where you optimise hyperparameters.**>**

12. **<Lisa M><How do I subset my data if there are only a few cases per class (10-13) and I would like to maintain the distribution of classes within the dev/test sets? Can I use bootstrapping in this case? (I am sorry if this has been said in the last few minutes of the last section, - my internet connection broke down.)>**
<Simon><Cross-validation or leave-one-out cross validation can be useful if you have relatively few samples and/or cases per class. Is this what you mean by bootstrapping? (see Q10 above). In this way, chance splits with few cases are averaged out. However, cross-validation is not equivalent to a leave out test data set, so if you absolutely need a proper test data set you may have to consider collecting more data. The issue with low sample sizes and class imbalance is that you get high variance in the test performance metric. One can adjust the training-test split percentage, reducing the performance metric variance at the expense of model training.

Maybe more to the point, the createDataPartition() caret function specifically aims to maintain the class distribution in the data partitions. But you, of course, need sufficient samples for this to succeed.**>**
<Lisa M> What I meant by “bootstrapping” was something related to the “cross-validation by bootstrapping” option offered by caret, I guess. However, as this

requires repeated subsampling of the original training data, I am not sure to what extent this option is useful for small data sets/low numbers of samples assigned to a certain class.

<Simon**>**<OK -- I think I understand. Unfortunately, I'm not too familiar with this approach. I know that one has to be careful in not overinterpreting the resulting performance metrics. The main idea is that you can keep resampling your data with replacement in order to estimate better the performance metric. At the end of the day, though, you don't actually get more sample information than the 10-13 cases you mentioned, so I don't think it'll fundamentally solve your problem. It may be a good way to assess the distribution of your performance metric and give you an indication of whether this distribution is too broad, which would suggest that you'd have to collect more data, or, alternatively, maybe the estimated performance metric distribution is good enough for your purposes.**>**

13. **<John><Do the axes in PCA always have to be perpendicular?>**
    <Simon**>**<Yes, by definition. Firstly, it's geometrically easier to think about vector spaces of this type. It's like establishing a smaller coordinate system; a plane in 3D space for instance. Similarly to how normal coordinate systems are perpendicular it makes specifying points in this space more convenient (imagine the trouble in plotting points in a xy plot if your x and y axis weren't perpendicular!). Secondly, and more importantly, if you were to pick one principal component at a time (which is how many PCA algorithms work) and you didn't pick one that was perpendicular to the previous components, there would always be a perpendicular solution capturing the same or more of the variance in the data.**>**
    Ok. makes sense i guess as only two variables are being plotted
14. **<antoine><How do we know which of the dimension correspond to which of the features?>**
    <Simon**>**<One can look at the 'loadings' on each principal component quantifying how much and in which direction each variable contributes. The loadings can be visualized in what is sometimes referred to as 'biplots'.**>**

15. **<Wojciech from chat><(in simple terms and in brief) - how is this explanation of variance by PCA explained?>**
    <Simon**>**<I'm not sure I fully understand the question. For PCA, variance means overall variance, or sum of variance across all variables. So when a principal component captures the most variance it means that most summed variance in the data is reflected in the points projected onto the given principal component. In this way, variance is sort of equivalent to information in the data (but also includes noise). The variance explained metric can then be interpreted as the proportion of the original data captured in the reduced dimensional space.**>**

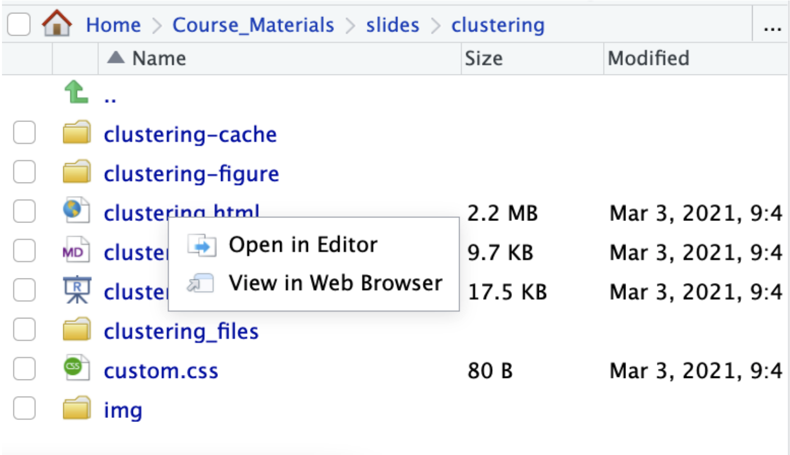
16. **<Ekim><Question> How does R deal with the missing values in a dataframe? I tried both techniques to reduce dimensionality (PCA, T-distributed), but the missing values can't be ignored and is causing error.**
    <Simon**>**<Short answer is that PCA and tSNE does not allow missing values. You'd have to perform some sort of 'imputation' before running dimensionality reduction;

that is fill in missing values using predictive modeling. More simply, it may be defensible to impute missing values to mean values, but you'd have to know your data well to convince yourself that this doesn't introduce artifacts.**>**
Ah right...is there a way to add an argument that tells R to just ignore the missing values? I tried `na.rm` but didn't work...
<Simon><Unfortunately not (there should be though!). You could omit samples with missing values using na.omit() depending on how many samples you're left with.**>**
Thank you!!

17. **<Ran Li><Is it possible to introduce UMAP which is another population Dimension reduction method? I think UMAP is more popular now than tSNE in the RNA-seq analysis. Does the distance in UMAP mean anything?>**
<Soumya><A good explanation of UMAP with code is here: https://pair-code.github.io/understanding-umap/   There may not be time in today's session to cover this. **>**
<Manik> Thanks Ran! Will see if it could be included in the next run. May be instead of tSNE (depending on if not a lot of people found it useful :))?

18. **<Ekim><Question> A general question about Bioinformatics classes: will the Github pages + this Google doc be taken down at some point?**
<Soumya><The github pages and Google docs will remain available along with the code and materials on github.**>**

19. **<Natalie Wallis><Are there slides available for clustering (they are not in slides folder)?>**
<Cathy><Adi's presentation is based on the Clustering section (4) in the course material (https://cambiotraining.github.io/intro-machine-learning/clustering.html#introduction)**>**
<Manik> You can also find them in Course_Materials>slides>clustering>clustering.html
<Natalie Wallis><clustering.html is not readable and none of the notes are on the clustering section in the course material - only the figures>
<Manik>That's interesting. Does it work when you select View in Web Browser? Anyways, as Cathy mentioned everything is in the course notes :)



Name	Size	Modified
..		
clustering-cache		
clustering-figure		
clustering.html	2.2 MB	Mar 3, 2021, 9:4
cluste	9.7 KB	Mar 3, 2021, 9:4
cluste	17.5 KB	Mar 3, 2021, 9:4
clustering_files		
custom.css	80 B	Mar 3, 2021, 9:4
img		

<Natalie Wallis> **That worked opening it through R - thank you. But the notes being shown on the screen aren't on the course materials - it's OK anyway will leave it.**

<Manik> **Cheers! Glad it worked.**

<Adi> **The notes are in html format as Cathy mentioned so you should be able to open them in a web browser.**

20. **<Name><Why in lots of the code does it specify things using [,1:2]?>**

    <Soumya><It means all rows and the 1st and 2nd column**> why only the first and second columns? Arent the columns all different cell samples? (in the exercise we did)**

    <Manik> If you look at the dataset, for example, blobs dataset, you would see that the first two columns have values in them while the last column is the label of the cluster or the ground-truth of each sample.

    Oh i see! So the third column is kind of the output we would test against?

    <Manik> Yes you would compare the results of your clustering algorithm against this ground truth label. That is why it needs to be omitted from training.

    Thank you, I understand now!

    <Adi> Yes, in this particular example we have ground truth labels, but please note that labels are not required for unsupervised clustering. You can cluster data without labels and verify whether the differences between clusters are meaningful experimentally. Classification is a different topic covered later, which is useful if you want to learn labels based on training data and apply them to new data.

21. **<fxq><What does "anisotropic" data mean?>**

    <Simon><Data that is not distributed spherically. Or put differently data with high covariance, although the complete answer is a bit more complicated than that. In contrast, isotropic data would be if your data is distributed according to a multivariate Gaussian distribution with the identity covariance matrix (so zero covariance and equal variance for each dimension).**>**

    Thanks.

22. **<Aleksandra><Is there a good (statistical?) method to see whether there is internal structure in the data and whether it is justified to cluster it, rather than just plotting and eyeballing?>**

    <Simon><The methods that I know are all of the type that tries to cluster the data first and then analyzes various metrics such as within cluster sum of squares to address whether clusters do exist. Personally I like the Gap statistic rather than the elbow method as it is less subjective. Both are better than plotting clusters on tSNE plots, though, which will show clusters when there aren't any (as you may have alluded to).**>**

23. **<Helena><When using the elbow plot to pick k, do you just decide based on intuition which is the best value to use?>**

    <Simon><Occasionally there will be an 'obvious' answer, such as k=3 in the course material for k-means. However, more often in practice it is less clear and there oftentime is an element of subjective evaluation in deciding the number of clusters to use. Often some clusters are more interesting. So, even though for a particular dataset the true number of clusters is higher it can even be advisable to suppress clusters if they are not meaningful. For example, if you are working with single-cell RNA-seq data and your true cell types are all weakly subdivided based on cell cycle state (even though you tried to adjust for this); these clusters are real but not necessarily interesting and it would be advisable, and defensible, to ignore them, focussing on the more obvious cell type clusters. Put differently, the elbow plot is one method to guide you in making this decision but it rarely, on its own, give you a definite answer.**>**

24. **<**Aleksandra**><Useful question in the zoom chat from Bruno (copying it here to have a record): Is there any (clustering) method using both distance and density?>**
    <Manik**>**Thanks! As a note, Adi mentioned probability distribution based methods which indirectly employ density and distance measures as a part of the probability distribution.
    <Adi**>** Yes, as mentioned this class of models is known as probabilistic Mixture Models. A popular one is the Gaussian Mixture Model or GMM, but you can define a mixture of any probability distribution.

25. **<**AD Hay from chat**><For choosing k, when would you identify the "elbow" (as you did in the exercise) and when would you choose based off of a silhouette plot?>**
    <Adi from chat**><**This is up to you to decide - there are various metrics used to assess clustering performance and they have different pros and cons (as do the methods themselves)**.>**

26. **<**Sebastian from chat**><How much bias is introduced by the K choice? There could be a population of cancer cells that is spread over all organs or there could be two populations in an organ that differ more between each other than to cells in another organ. Can I check for the probability that I would miss that?>**
    <Adi**><**This really depends on the features that you are using. Remember clustering is unsupervised and "exploratory" so you could detect different clusters within your data but the difference may not be biologically meaningful. If you detect let's say two different cancer cell clusters, you could then follow up by characterizing the difference between them (for example, apply differential expression analysis). It could be that they separate based on something like cell cycle or sequencing coverage, which is true structure in the data but may not be meaningful based on your scientific aims. So ultimately you need to decide what is meaningful and then follow this up with experimental validation.**>**

27. **<**Ran Li**><I think I still need clarification about why we can't use UMAP to do further clustering as UMAP stores distance and preserve the global structure.**

For example, if from the umap, cluster 1 is nearer to cluster 2 in the UMAP than cluster 3, can I say cluster 1 has closer relationship to cluster 2 than 3. >
<Adi**><**UMAP is considered better than tSNE at preserving global structure, but it still does a non-linear embedding and distorts distances, so there is always a risk that you are clustering based on features that are not actually within your dataset. As mentioned, if the dataset is high-dimensional it is best to reduce the dimensionality with PCA and cluster based on that, then use UMAP for visualization.**>**

28. **<**Amir Hay**><Do you have recommendations for clustering with large amounts of data? E.g. CLARA?>**
<Adi**><**If you have a large number of features, reducing the dimensionality with PCA prior to clustering will help. If you have a very large number of datapoints, doing some sort of bootstrapping (fitting based on subsets of the data) will help, which I believe is what CLARA does.**>**

29. **<**Ko**><What does the "set.seed" function do in the DBSCAN? , e.g.exercise 4.8.3>**
<Manik**>**set.seed() is a R function which allows you to reproduce the same result over and over again. For example, if you are randomly splitting your data in two groups, set.seed will make sure that you get the exact same split everytime you run it.

30. **<**Helena**><With the image segmentation example, when performing the k-means clustering I don't quite understand why we are selecting r g and b from the imgDF - underlined part in code**

```
res <- foreach(
  i=k,
  .options.multicore=list(set.seed=FALSE)) %dopar%
kmeans(imgDF[,c("r", "g", "b")], i, nstart=50))
```

>
<Adi**><**Image segmentation is not a really common application of clustering algorithms, but what we are doing here is essentially ignoring the locations of the pixels (treating the image as an unorganized collection of pixels) and clustering pixels by their color (RGB value). So the numerical values in the red green and blue channels are the features, and we then compute distances between them. This works surprisingly well for this H&E slide where the colors alone distinguish different parts of the image really well. Of course, for proper image segmentation the relative location of pixels within an image is important.**>**

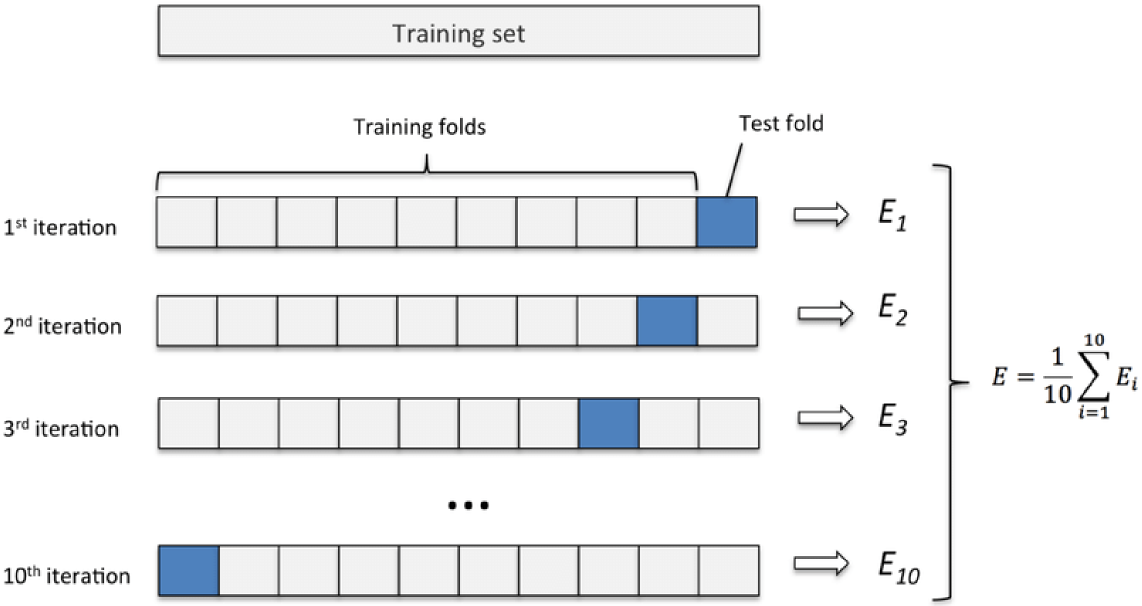
DAY 2

31. **<**Aleksandra**><Why is the accuracy of the kNN lower than what was shown when we ran the cross-validation? Is it because cross-validation splits the training dataset only and does not touch the test data?>**
<Eleanor**><**My guess would be that the lower accuracy comes from a couple of different sources. Firstly, the cross-validation accuracy is an average across the 10

iterations so there is actually a range of cross-val accuracies and our test accuracy is probably just on a lower range of those. Secondly, we chose our k based on the training set so it's possible the k chosen is best for the training data but not necessarily for the full set. There will always be some small differences between the training and test sets (although we try our best to minimise them) so you will often end up with slightly lower test accuracy. Here the difference is quite small so it is not too concerning but if you had a bigger difference you might need to be worried about how robust your results are.>

<Aleksandra><I see, thanks! Just to be sure, because I am a little confused: when we run cross-validation, we sub-split the training set into new "train" and "test" sets. We do this 10 times with different splitting, choose the hyperparameter that gives us best accuracy, and then at the end test the chosen hyperparameter on the initial independent test dataset. Is this right?>

<Eleanor> <Yes exactly, the naming all gets a bit confusing but that is the right set-up. So essentially at the very beginning we split into training and test and forget about the test set for a while (we can call it the 'unseen' set for now). Then we focus on the training set and split it up, like this:



Into 'test folds' and try some different hyperparameter values and report the average accuracy across all the folds to choose the best one. Once we're happy with our hyperparameter, we can find out what our actual unbiased performance is using the test/'unseen' set. We use that only once using the hyperparameter(s) we picked.

<Aleksandra><Thanks! And if the accuracy turns out to be bad with the unseen test set, are we allowed to go back and tune the hyperparameters again?>

<Eleanor> So if you get to the end and find your performance is bad on the unseen set then you'll want to go back and work out why that was the case. Maybe you didn't split training and test well (so uneven class distributions or something else weird) or maybe the model just isn't well suited so you're over-fitting to the training set. Cross-validation is normally quite good at preventing overfitting too much to the training data but problems can creep in. I would say going back and just trying again blindly won't fix the problem but you can change your approach for sure. You don't want the performance on the test set to influence your decision-making too much because it's meant to be unbiased so you won't want to iterate too many times but you can get a sense of how well your approach works using it.

32. **<Aleksandra><How do we choose the distance metric (Euclidean or other) in kNN and the evaluation criterion (e.g. information gain) for decision trees?>**
<Soumya Banerjee><This is very application specific. It depends on what kind of data you have. It could be a distance metric that is good for gene expression data, is not so good for data from retail stores (say if you are building a recommendation system based on purchases done in a retail store). This is very much the big part of what a modern data scientist does. Over some time you can build up some intuition. Some text from the recommended textbook gives some more clarification.
https://www.statlearning.com/s/ISLR-Seventh-Printing-xwa7.pdf

Page 396 - 398

Thus far, the examples in this chapter have used Euclidean distance as the dissimilarity measure. But sometimes other dissimilarity measures might be preferred. For example, correlation-based distance considers two observations to be similar if their features are highly correlated, even though the observed values may be far apart in terms of Euclidean distance. This is

For instance, consider an online retailer interested in clustering shoppers based on their past shopping histories. The goal is to identify subgroups of similar shoppers, so that shoppers within each subgroup can be shown items and advertisements that are particularly likely to interest them. Suppose the data takes the form of a matrix where the rows are the shoppers and the columns are the items available for purchase; the elements of the data matrix indicate the number of times a given shopper has purchased a given item (i.e. a 0 if the shopper has never purchased this item, a 1 if the shopper has purchased it once, etc.)
What type of dissimilarity measure should be used to cluster the shoppers? If Euclidean distance is used, then shoppers who have bought very few items overall (i.e. infrequent users of the online shopping site) will be clustered together. This may not be desirable.
On the other hand, if correlation-based distance is used, then shoppers with similar preferences (e.g. shoppers who have bought items A and B but never items C or D) will be clustered together, even if some shoppers with these preferences are higher-volume shoppers than others. Therefore, for this application, correlation-based distance may be a better choice.
>

33. **<Bruno><Does Decision tree suffer for unequal classes representation?>**
<Eleanor><Yes unbalanced classes can cause problems in Decision Trees as they will tend to perform better for the over-represented classes (as the cost for predicting

badly on the smaller classes will be smaller). To overcome this, you can use subsampling or oversampling or actually change your cost function to penalise misclassification for smaller classes more to counteract the differences. I think you can actually inform the function of the imbalance, possibly using the prior parameter but you can read more in the rpart documentation https://cran.r-project.org/web/packages/rpart/rpart.pdf **>**

34. **<Bruno><Is it possible to define a minimum number of samples for Random Forests? >**
    <Eleanor><You can define the number of trees in the Random Forest (as a hyperparameter), which corresponds to how many samples of the data you are using. The number of trees should be chosen through cross-validation usually. You can use the ntree parameter (https://cran.r-project.org/web/packages/randomForest/randomForest.pdf). There are lots of other parameters you can tweak too to change attributes of the forest.**>**

35. **<Lisa M><Is there a way to get information on which decisions/questions were linked to specific variables with a high importance in the random forest? Something comparable to a decision tree?>**
    <Soumya><You can visualize single decision trees in the random forest. You can also do this in a package called *party* in R and in scikit-learn in python. **>**
    <Irina> the identity of the selected/ discriminative features is more important/ informative than the actual thresholds that are used. E.g. for the iris dataset the petal features were used a lot, with different thresholds. However the threshold values were not informative.

36. **<Aleksandra><I am confused about how the SVM, or even the linear support vector classifier works. Is the initial line constructed using all the data, and then refined using only the support vectors (whose number we specify using the C parameter)? Or is it constructed in the first place using only the support vectors (are they then somehow randomly picked)? >**
    <Irina>The initial hyperplane is optimised using the whole dataset (all points); next, the support vectors are determined; for the later stages only the support vectors will be used to determine the class of a new measurement. The C parameter influences the number of support vectors; however the number of support vectors is a consequence of the choice of C, the actual number of support vectors is not specified. The support vectors are chosen by the model, they are not randomly picked.

37. **<Fanchao><How do these machine learning methods deal with multi-collinearity between predictors? How to appropriately interpret the predictors selected by these machine learning methods when the multi-collinearity exists, e.g., which predictor is more important?>**
    <Soumya Banerjee><Very good question. The short answer is: you have to detect multi-collinearity and then make adjustments to your model. For example, if age and

location are correlated consider lumping them into one variable, or consider only one variable.
From page 243 of the ISLR book,
https://www.statlearning.com/s/ISLR-Seventh-Printing-xwa7.pdf

When we perform the lasso, ridge regression, or other regression procedures in the high-dimensional setting, we must be quite cautious in the way that we report the results obtained. In Chapter 3, we learned about multicollinearity, the concept that the variables in a regression might be correlated with each other. In the high-dimensional setting, the multicollinearity problem is extreme: any variable in the model can be written as a linear combination of all of the other variables in the model. Essentially, this means that we can never know exactly which variables (if any) truly are predictive of the outcome, and we can never identify the best coefficients for use in the regression. At most, we can hope to assign large regression coefficients to variables that are correlated with the variables that truly are predictive of the outcome.
For instance, suppose that we are trying to predict blood pressure on the basis of half a million SNPs, and that forward stepwise selection indicates that 17 of those SNPs lead to a good predictive model on the training data. It would be incorrect to conclude that these 17 SNPs predict blood pressure more effectively than the other SNPs not included in the model. There are likely to be many sets of 17 SNPs that would predict blood pressure just as well as the selected model. If we were to obtain an independent data set and perform forward stepwise selection on that data set, we would likely obtain a model containing a different, and perhaps even non-overlapping, set of SNPs.
This does not detract from the value of the model obtained— for instance, the model might turn out to be very effective in predicting blood pressure on an independent set of patients, and might be clinically useful for physicians. But we must be careful not to overstate the results obtained, and to make it clear that what we have identified is simply one of many possible models for predicting blood pressure, and that it must be further validated on independent data sets.
>
<Fanchao><Thanks for your answers.

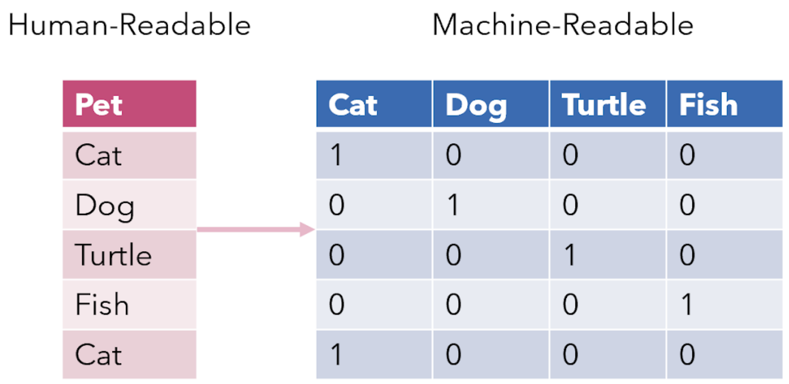
1. **If the aim is to select the strongest predictor among a wide range of related predictors, e.g. select the strongest predictor for disease from hundreds of metabolites, which method do you suggest?**
2. **If the aim is to look at how strong one predictor is associated with the disease, independently of the other predictor, how to deal with the collinearity between the two related predictors?>**

<Soumya Banerjee><In the first instance, a simple model (linear or logistic regression: which will be covered tomorrow. Then try selecting a ‘simpler’ model by trying to select the most important variables. This can be in many ways. One can perform LASSO, which is a way of selecting variables that are the most important

(simplifying here). Definitely read the last few chapters of the ISLR book (linked above, pages 243). Chapter 6, Linear model selection and regularization>
<Irina> you might want to look at several predictors, which combined might explain the variance in the output variable. Also, evaluate the correlation between predictors before concluding that they are independent.

38. **<Helena C from chat><What is kappa?>**
<Irina>Kappa = Cohen's kappa = Cohen's Kappa takes values between -1 and 1; a value of zero indicates no agreement between the observed and predicted classes, while a value of one shows perfect concordance of the model prediction and the observed classes. If the prediction is in the opposite direction of the truth, a negative value will be obtained, but large negative values are rare in practice

39. **<Aleksandra><If we have integer-valued features, should we suspect that they are actually categorical variables? And if they are, should we worry about the distances between them or them being smaller/larger don't actually have a meaning?>**
<Soumya><One should try to find out if they are categorical or not. For example, if a variable is integer valued, it could also refer to (say) number of family members in a household. If it is indeed categorical, then, one should encode it (for example, one hot encoding: see below). This will make sure all the categories are equidistant from each other.
https://miro.medium.com/max/3758/1*O_pTwOZZLYZabRjw3Ga21A.png



Human-Readable

Pet
Cat
Dog
Turtle
Fish
Cat

Machine-Readable

Cat	Dog	Turtle	Fish
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1
1	0	0	0

>

40. **<Bruno F from chat>Are we supposing a linear relation between features and outcome variable?**
<Eleanor><In kNN, we are not assuming too much about the relationship between features and their outcome variable. This is one of the strengths of this kind of 'non-parametric' approach. We are hoping that similar points have similar target variable values as then our model will be well-suited but unlike in linear regression, we are not assuming anything about the form of f where X is the vector of features for one example and f(X) gives the outcome value for that example.**>**

41. **<John><How do you find the outliers in your dataset easily?>**

<Manik>The most straightforward way is to plot your features one by one and check if there is any outlier. You can also calculate summary statistics such as mean, median, variance and standard deviation for each feature to check if some feature has a huge difference between mean (affected by the presence of outliers) and median values (robust to outlier values) or have huge standard deviation.
<John>Thanks. But in the example given there is an outlier where the predicted value differed a lot from observed. Given that there are many data points with similar observed, i am not sure how i would find that individual data point in the original dataset.
<Manik> I see that your question is about tracing back an outlier to the dataset. The easiest way in this case would be to find the point having the x-y coordinates as in your plot. Does this help? You can look at the data matrix used for making the final figure/conclusion and trace the point corresponding to those values.
<John> Thanks. Just wanted to check there wasn't any easier way.
<Manik>Hmm I think some packages also let you hover over the points and see the sample name/co-ordinates, etc. I have asked other tutors as well if they have any other easier solution in mind.
<Soumya> < I see there is a package in R also (I have not used it; so cannot vouch for it)

https://cran.r-project.org/web/packages/OutlierDetection/OutlierDetection.pdf

And some more statistical tests and tips for outliers

https://statsandr.com/blog/outliers-detection-in-r/
>
<John> Thank you.

42. **<Bruno><Probably I've missed something. In the train function, what is the seed supposed to be?>**
<Eleanor><So we set lots of seeds so that all the randomised parts are reproducible. Here:
set.seed(42)
seeds <- vector(mode = "list", length = 26)
for(i in 1:25) seeds[[i]] <- sample.int(1000, 10)
seeds[[26]] <- sample.int(1000,1)
We are using an initial seed (42) to generate a series of seeds (because here the generation of seeds is also random) which will be used throughout training. You can run the whole thing with no seeds and it will work but you might end up with slightly different results. We then use this big seeds object within the train control parameter. 42 here is just a choice (/Hitchhiker's Guide to the Galaxy reference) and you can use another value if you fancy but it won't match up to the notes then. Does that help?
Yes, I've got it. thanks>
<Aleksandra><Following up from this - why do we choose random seeds smaller than/equal to 1000 with sample.int(1000,10)?>

<**Manik**>This allows a wider choice for selecting random seeds. If you are only sampling 25 numbers, maybe giving a limit of up to 100 numbers to choose from would be alright as well.

<Aleksandra><But do we have to specify a limit at all?>

<**Manik**> I am afraid this is how the function sample.int() works. You mention out of 1000 numbers, give me 10. If you are using another function which might say, just give me 10 random, different numbers, that would be fine too.

<Aleksandra><Oh I see, I am stupid, I did not read the help page till the end. This makes sense, thanks!>

<**Manik**> No worries! :) You can also manually create your own list of random numbers too. These functions just make life easier

<Eleanor> There is more info about setting the seeds in the caret documentation https://cran.r-project.org/web/packages/caret/caret.pdf :

an optional set of integers that will be used to set the seed at each resampling iteration. This is useful when the models are run in parallel. A value of NA will stop the seed from being set within the worker processes while a value of NULL will set the seeds using a random set of integers. Alternatively, a list can be used. The list should have B+1 elements where B is the number of resamples. The first B elements of the list should be vectors of integers of length P where P is the number of subsets being evaluated (including the full set). The last element of the list only needs to be a single integer (for the final model). See the Examples section below.

43. **<Aleksandra><Is there a simple way to replace correlated features with mean/median rather than excluding them? I tried to look at the documentation for caret's preProcessing, but didn't see anything like that in there.>**

    <**Irina**> I will answer first from a computational angle. Using the object generated from the correlogram you can identify which features are highly correlated (they will have |corr| > 0.75 in the respective matrix, |a| stands in for absolute value e.g. |7| = 7 and |-7| = 7). Then using a subset of these features (the selection can be performed per columns), you apply the mean or median on rows (using the apply function). I did see an automated function to do this - I'll try to look it up.

44. **<Alex><Can we download the .RMD files for later use? Sorry if I missed that.>**

    <**Soumya**><Yes. It is all available on github. You can install all the packages on your desktop/laptop. The data is also available on github.
    https://github.com/cambiotraining/intro-machine-learning
    >
    Thanks

45. **<Bruno><Following from question 43: instead of using median/mean of highly correlated features, is it possible to substitute them with PCA components?>**

    <**Soumya**><Yes definitely this is one approach that can be tried. This comes under the topic of semi-supervised regression. It is also called supervised PCA.
    https://web.stanford.edu/~hastie/Papers/spca_JASA.pdf
    And

https://en.wikipedia.org/wiki/Principal_component_regression
>
Thanks
<Irina**>** Yes, it is an option to summarise the features using PCAs, however, remember that the PCAs are a linear combination of all features, or a selection a features i.e. in PC1 you might have some of the highly correlated features, but clothes as well. Also, in a supervised setup it is often preferred not to work on the PCA summaries, but on the individual, or (manually curated) summarised features, since these are easier to interpret.

46. **<Aleksandra><A somewhat "abstract" question I suppose: is there a way to visualise or magine how the SVM works in the case of regression? For kNN I imagine (if we only think about 2d, i.e. 2 predictors) that some "surface" is constructed across the whole plane, which specifies the predicted values depending on the coordinate of the point on the plane. For kNN, this surface is generated taking into account values of the k nearest neighbours. Is there a way to imagine something similar for SVM? Sorry for the clumsy wording, I hope you get what I'm trying to explain! >**
<Tutor><Answer>

47. **<Name><How did she get the ggpairs to open as a new window PDF? Mine just says null device 1 ??>**
<Eleanor**><**You need to use something like this
pdf("all_info_covid.pdf", width=20, height=20)
ggpairs(covid.data, ggplot2::aes(colour = result, alpha = 0.4))
dev.off()
But you'll want to change covid.data to what you've called the dataset. That will then save it to a pdf and if you open it in a new window it should look like mine. If that doesn't work let me know and I can have a look**>**

48. **<Alex><how would you go about subsetting the results in order to reduce model biases? As Eleanor mentioned in the discussion of COVID dataset analysis.>**
<Eleanor**><**One idea is to subsample the set of result 0 points multiple times and test your models on all the subsamples combined with the result 1 points. You might want to check if your subsample is representative using some form of statistical test (e.g. chi-squared). Here are some more practical ideas of how to deal with the unbalanced sets
https://www.analyticsvidhya.com/blog/2016/03/practical-guide-deal-imbalanced-classification-problems/. **>**

DAY 3

Please add your write-up of exercise 10: Linear and Logistic Regression below as advised by Chris. And any questions that you may have, please keep the numbering.

49. **<Name><what is happening here???>**

    ```
    lrfit <- train(y~., data=data.frame(x=Xs,y=D[25:nrow(D),geneindex]), method = "lm")
    predictedValues<-predict(lrfit)

    summary(lrfit)

    lrfit2 <- train(y~., data=data.frame(x=Xs,y=D[1:24,geneindex]), method = "lm")
    predictedValues2<-predict(lrfit2)
    ```

    <Simon><The predict function, by default, returns the values of the fitted line evaluated at the input provided at training. It can be useful for evaluating the fit. However, it's more true to its function name when using a fitted model to predict values based on new, unseen data points. For example, see the following line in section 10:

    ```
    prob <- predict(mod_fit, newdata=data.frame(x = Dpred$Time, y = as.factor(Dpred$Class)), type="prob")
    ```

    >

50. **<Name><How do you see the SSE of your curve?>**
    <Simon><One way to use SSE in the context of machine learning would be to use to optimize hyperparameters, such as which polynomial degree to use. In contrast to performance metric from classification, though, it's less meaningful to report the SSE because the magnitude of the error is hard to compare across regression problems.**>**

51. **<Name><In order to verify the "quality" of the inferred lm model, should we also consider the residuals distribution? >**
    <Simon><I think it's safe to assume that the errors are normally distributed in this exercise at least. Often the log transform (which has been applied to this data) helps ensure approximately normal errors for gene expression data. But it's a good idea to do in general and essential to be aware of model assumptions when applying regression and how your particular data may deviate from this.**>**

52. **<Livia><What is the benefit of running a logistic regression over grouping like knn?>**
    <Simon><They are two quite different classification models, so there may be characteristics of your data that make one perform better than the other. k-nearest neighbor is an example of a non-parametric model wheraes logistic regression is parametric. For very large datasets this means that the computational cost of training and prediction will be different. For knn there is no training time but evaluation is more costly for instance. So if you have millions of data points, for instance, this may be quite an important practical distinction. Furthermore, one might be interested in looking at the fitted parameters of the logistic model to assess which variables are important for the prediction. This type of analysis wouldn't be possible with knn, but, on the other hand, one may gain insights for a particular prediction considering which

k samples gave rise to a particular prediction. In addition, logistic regression has a direct probabilistic interpretation in the predictions; so when predictions on new observations will have a [0, 1] range which is interpreted as the class prediction and confidence in that prediction. One could average the classes seen in the k nearest neighbors to assess confidence but it's theoretically less well founded. Lastly, there are important differences in how one would try and control overfitting and some may be better suited to your data; logistic regression, like other types of regression, you may want to include a coefficient shrinkage hyperparameter (also known as regularization). In contrast, knn relies on specifying the k which is a discrete choice of hyperparameter.**>**

53. **<Livia>when would you choose to run a linear modelling with machine learning (instead of a 'normal' linear regression models) is it solely to predict values in the future?>**
    <Simon><To my mind there is little difference between 'machine learning' linear regression and 'normal' linear regression. In general (and a bit overly simplistic), machine learning approaches tend to have more emphasis on the predictive aspect of the modeling and on controlling overfitting. In addition, they tend to fit more complicated models using larger dataset. Actually, the main motivation for introducing logistic regression here, in the course, is that it's a useful conceptual building block for understand artificial neural networks, which we'll discuss after lunch, and which can be thought of as fitting consecutive layers of logistic regression models. **>**

54. **<Aleksandra (group 4)><Within the Arabidopsis gene expression dataset, we focussed on the time series of AT1G14920 expression during the infection time course. It exhibited a curious S-shape pattern with quite a sharp downregulation between T = 20 and 30 hrs, potentially also with a peak before the downregulation.**

    I first fitted a linear model to the time series, as well as polynomials of degrees 2-5, 10 and 20, using the "lm" method within the train() function of the caret package (with seed = 23). Analysis of the RMSE values revealed that the best fit to the data was polynomial of degree 10. However, visual inspection suggested that overfitting to the data was likely.

    Next, I split the dataset into "train" and "test" sets with ratio 60:40, using the caret package function createDataPartition() (seed = 678). This ratio was chosen to maximise the amount of training data points (16) while still keeping a reasonable number in the test set (8). The fitting was repeated on the training dataset and then evaluated on the test dataset. Indeed, RMSE analysis indicated that the best fit was achieved with a polynomial of degree 3. Inspecting the plot of the fit indicated that this polynomial was the first one that could recreate the wave-like pattern of the data (i.e. linear and quadratic fits were not able to do that), while higher-degree polynomials resulted in overfitting, especially towards the end of the time series.>

<Chris><This is a very nice write up. Only a few things I would add/edit here would be to be explicit about where the data comes from. In this case the microarray (including version) and how it was normalised, and perhaps include package versions of the packages. With these instructions I'd be happy to chase this up and try to repeat myself.>
<Sergio><Thumbs up>

55. **<John><A bit of a less technical question: I am a complete novice at ML, and given my day job, unlikely to be able to master it. Do you have any advice about the best way to go about finding collaborators who have the right skill set in machine learning? I am interesting in imaging of translucent structures in the eye and suspect that machine learning is going to be the solution to this.>**
    <Simon><I would recommend reaching out to computer science institutes either here or elsewhere.
    https://www.cst.cam.ac.uk/
    http://mlg.eng.cam.ac.uk/
    If you can formulate your research problem first in terms introduced in this course -- classification or regression, unsupervised or supervised learning, and why it matters, you should fare well in finding collaborators as interesting and unique datasets are quite motivating.**>**

56. **<fxq><This virtual environment seems to have been created with Singularity / Docker. Is there a chance you could provide the Dockerfile for this? I might be interested to build this on my own machine for learning.>**
    <Simon><I've asked Paul if he has the Dockerfile -- it is potentially available but not readily so. After some discussion, the recommendation is to install the software locally isntead. Although the Keras and TensorFlow dependencies are trickier to install than normal R packages available on CRAN, it doesn't take too long and there are guides available online:
    https://github.com/statsmaths/kerasR
    In addition, installation instruction will be added to the course website soon -- see section 13.3 further reading for installation instructions.**>**

57. **<Bruno><Can you explain what does Epoch mean?>**
    <Simon><Maybe the Keras documentation is helpful
    https://keras.io/getting_started/faq/#what-do-sample-batch-and-epoch-mean :

- **Epoch**: an arbitrary cutoff, generally defined as "one pass over the entire dataset", used to separate training into distinct phases, which is useful for logging and periodic evaluation. When using `validation_data` or `validation_split` with the `fit` method of Keras models, evaluation will be run at the end of every **epoch**. Within Keras, there is the ability to add callbacks specifically designed to be run at the end of an **epoch**. Examples of these are learning rate changes and model checkpointing (saving).

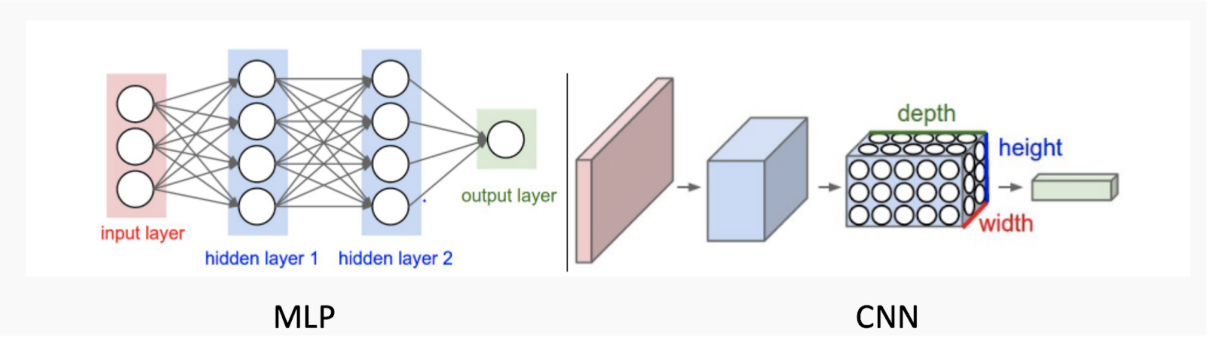
So, it determines how often, during training, the performance is evaluated and printed to the R console.**>**

58. **<Sebastian><How can I view single images from the huge array? Which R command would do that?>**
    <Simon><For example, to render the 10th image in the multidimensional array: grid.raster(allX[10,,,], interpolate=FALSE, width = 0.5)**>**

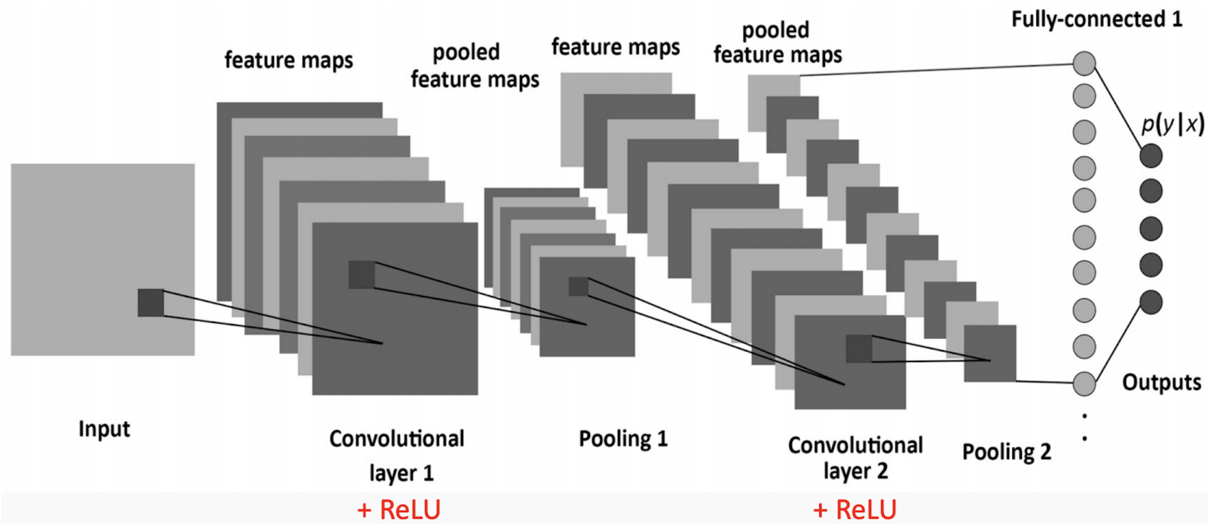
59. **<Aleksandra><Organisational question: is it possible to download the course book? What I found online suggests that I need to clone the repo and build the R book - could you help me with that? It would be great to keep the course materials, they are great!>**
    <Simon><You should be able to clone the repository with
    $ git clone https://github.com/cap76/intro-machine-learning-2019B.git
    In addition, the course book is permanently available and won't disappear after the course.**>**
    <Sergio><The course website https://cambiotraining.github.io/intro-machine-learning/ will continue operative after the course**>**

60. **<Bart><In a convolutional neural network, when we have 2 layers of filters, are all the second filter applied to the outputs of all first filters? Or are the specific 2nd filters applied to specific 1st filters?>**
    <Simon><Unfortunately, I'm not an expert on convolutional neural networks, but I do think the former seems more accurate. It may be a bit more complicated than that though. This seems like a good introduction:
    https://towardsdatascience.com/simple-introduction-to-convolutional-neural-networks-cdf8d3077bac

    Containing 2 diagrams that may be helpful in thinking about multiple layers of convolutional neural networks:



There's also a diagram for pooling:



>

<Chris**><**The architecture and the way networks work are fully definable, although if you go into TensorFlow. In our implementation it would be the second. One way to check would be to look at the network itself e.g., if I type:

mod

(for a model I've called mod) it will print out the dimensions of the individual layers, so you should be able get a better interpretation of what's going on with our specific default keras operations.**>**

61. **<**Jennifer from chat**><If you are not sure what type of network to use is there a way to test the "fit" so to speak of different ones for your data?>**
    <Simon**><**You could test the classification performance on a test data set, comparing the performance of both types of neural networks and different architectures. However, since training can be quite costly it may be computational expensive to train multiple models like this.**>**

62. **<**Name**><Question>**
    <Tutor**><Answer>**

63. **<**Name**><Question>**
    <Tutor**><Answer>**

64. **<**Name**><Question>**
    <Tutor**><Answer>**

65. **<**Name**><Question>**
    <Tutor**><Answer>**