Fundamentals of Machine Learning for Earth Science
April 20 - May 4, 2023

Questions & Answers Part 1

Please type your questions in the Question Box. We will try our best to get to all your questions. If we don't, feel free to email trainer 1 (email 1) or trainer 2 (email 2).

Question 1: What counts as "very large numbers" for the comment about R vs. Python - millions, billions, trillions?
Answer 1: By very large numbers, I mean "mathematically large", for instance, R provides a package called "gmp", that can help to calculate the following case: n=10^250, n^2=?; instead of returning "inf", R with gmp can provide the correct answer of 10^500. Besides this minor "advantage", the most important strength/advantage of R is its capability of building statistical models and vast packages for statistical computing.

Question 2: Are there any case studies where a hybrid method is used? Are data-driven and model drives applied at the same time? And at which level can we use each one in the same process?
Answer 2: There are many cases where a hybrid method is used. Some of these techniques are called physics-informed models, and/or knowledge-informed models. There are other cases where data-driven models are used for one task, and model-driven is applied for a downstream task (and vice versa). There are no specifics on where to use one or the other. If you have the data available to train a data-driven model, you can always rely on it, and validate with some model-driven data. You could also generate synthetic training data from the model-driven method to then train the data-driven model. You could even use some of the equations from the model-driven method to use as a loss function (learning function) of your data-driven model.
There are also cases where people use the ML method to do feature selection or parameter tuning for the data driven model. This takes advantage of the potential speed of the ML method while retaining the potential explainability of the physics-based model.
Among many domains, this paper summarized some progress on the hybrid method in climate science: https://doi.org/10.3389/fams.2023.1133226

Question 3: Hi, how is the tradeoff between speed and accuracy determined for the models shown in this figure (slide 36)? How do you define accuracy here?
Answer 3: Slide 36 has a general simplification of speed vs. accuracy. The Accuracy can be any metric on your end. It would be the metric you want your model to improve on. Many simple algorithms are very fast.

Question 4: Can we use the Random Forest model to find crop water use?
Answer 4: In general you can train an ML model to find anything that you have training data for. So if you have data that describe crop water use that you can use as training data, then you can at least "try" to model it with a set of predictors. Your degrees of success will be determined by the quality of your training and the relative use of your predictors to map the crop water use.

Question 5: How do you choose the number of clusters?
Answer 5: It will depend on the number of classes or clusters you are trying to find. K-means: Depends on your question. The software can set a limit.

Question 6: Could you please explain how K-means is unsupervised?
Answer 6: K-means is not dependent on training labels, it simply looks for "similar" spectral features in the dataset and groups them into a similar class. However it doesn't tell you "what is the feature" only that there is a feature at that location.

Question 7: Why does Random Forest not work well while doing cross-validation?
Answer 7: Cross-validation is a second step in the evaluation of a model. RF can be used here. It is just downsized, it can only classify well on data it has already seen.

Question 8: When we use algorithms such as Random Forest or Tree Decision and we have different inputs for the model, is it advisable to do a preliminary analysis to avoid collinearity, or is the algorithm able to manage it?
Answer 8: It is always good to do an exploratory analysis of features and data. You can find what features are similar enough to see how your model will be classifying/learning. In algorithms such as Decision Trees it is always good to avoid collinearity.

Question 9: I have three questions: 1. What are the possible solutions if the number of samples of the classified classes is unequal? 2. What is the overfitting problem? 3. If I have different features, how can I study the feature importance?
Answer 9: 1) Several ways, you could drop some features from the data in a class you have the most. You could increase the number of observations (synthetic, data augmentation). It is always best to try to balance your classes if you can. Neural Networks have functions to handle that balance.
2) Overfitting is where the model learns to reproduce the training data very well but does not produce accurate results when run on data it has not seen before.
3) Many ML models (particularly tree based models) will give you a ranking of the input features in terms of their importance to the outcome. This can be more difficult with Neural Nets but new Python packages are becoming available to assist in this area.

Question 10: Do you have any additional machine learning references that would be helpful to read/use (particularly on the wide variety of algorithms and their uses)?
Answer 10:
General ML:
Courses: https://www.coursera.org/specializations/machine-learning-introduction (free without certificate)
Books: https://sebastianraschka.com/blog/2022/ml-pytorch-book.html
https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/
Deep Learning: https://d2l.ai/ (book w/ code examples)

Question 11: When I try to change the runtime type, the Hardware Accelerator is already set to GPU and the GPU class is Standard (can purchase more for premium GPUs). Is that correct?
Answer 11: Yes, re Assignment #1. You do not need to purchase. Make sure to follow the instructions above the cells.

Question 12: What do you think are the downsides of machine learning as regards land change science, knowing that natural and anthropogenic activities are dynamic?
Answer 12: Many downsides are the data we have available, how much data we can input into our model. Sometimes features that are extracted.

Presumably you are referring to remote sensing land “cover” change. The downsides are more about the limitations of remote sensing than the limitations of the ML. If you have appropriate inputs you can identify the features you want on the landscape.

Question 13: I am trying to work with atmospheric phenomenon forecasts and predictions. What should guide my choice of training data - reanalysis data or forecasts of numerical weather prediction models? Can we also take real time observations available for a short period of time for verification of our model results?
Answer 13: It will depend on the temporal and spatial resolution how temporal- or spatial-specific your model will be. Real time: yes, you can also have training data that is closer to real time (minute, hours, 10-day forecasts). Even real time data has uncertainty as well, so keep that in mind.

Question 14: What is the most suitable machine learning model for predicting earthquake hazard potential, and what are some methods to detect if the model is overfitting the data? How can we say that the division of the training set, validation set is the optimum? What are some techniques to mitigate the risk of false positives and false negatives in the model's predictions?
Answer 14: There is no simple answer to this. It simply depends on your input data available, your skill level with developing algorithms, and your available software. If you need a quick answer on limited resources it’s probably best to start with simple methods like tree models and see if that works for you.

Question 15: Can supervised classification performed in Image Classification Software such as ERDAS imagine or QGIS-Semi Automatic classification be considered as machine learning techniques?
Answer 15: Things like Maximum Likelihood is not a ML classifier but each of those software do have some ML methods available.

Question 16: In terms of raster datasets, does samples mean the number of labeled pixels of the class we are interested in?
Answer 16: With no other context… I will say yes that is probably correct.

Question 17: What would be the difference between using Sentinel-2 (10m) and MODIS (250m) for surface water modeling with ML?

Answer 17: In terms of the steps in generating the model, very little. In either case you need to collect training labels and input predictors (presumably the spectral bands from the instrument), train the model and then apply it to the data. At 10m spatial resolution Sentinel does have fewer bands so your false positives and false negatives would likely be greater. In addition, with finer spatial resolution you have more features that can cause confusion with water such as cloud shadows, terrain shadows, and different types of dark ground features like dark soil, burns, etc.

Question 18: When working with satellite data, is there a specific way to split, train, and test a dataset?
Answer 18: The general practice from machine learning is to have a 80/20/20 or 70/30/30 split of training, testing and validation data. The volume of this data will be heavily dependent on the application, problem, and model being used. If your satellite data is in array or tabular format, you can always use the scikit-learn split function to create the split for you in a stratified way.

Question 19: What is the strategy to prepare and annotate a dataset when they're not available? How much is enough data and how do you judge its quality?
Answer 19: There are many tools and ways of annotating data when there is none available. The first step would be to try unsupervised learning methods to try to gather features from your data to label. If the method is not efficient, you can use your domain knowledge to create the training dataset. Our rule is to always create training data in batches, and re-train our models as we increase the number of training samples. This will allow you to monitor your model performance without spending too much time labeling. The quality of your training data will depend on your domain knowledge and expertise. You will be the one to judge the quality of the training data base on the science properties of your problem.

Question 20: When we looked at the k-means clustering example on the notebook, was the goal of that exercise extracting a binary variable from the raster image that identified water and no water pixels? Doesn't the raster already have that information? I was confused about what the goal is with applying this method to the raster image.
Answer 20: In this example, the idea was to extract clusters out of the data to try to classify the pixels from the raster into water and not water. We used 4 clusters to increase the variance of our clusters trying to improve our prediction of water and

not-water pixels. In many other cases, you can use these initial clusters as a first step to label data for training, without having to draw polygons or identify the features pixel-by-pixel. The raster does not have the information of which pixels are water and not water, they only include the surface reflectance bands, thus we want to produce a water map.

Question 21: For time-series data (such as data from InSAR), which type of models are suitable?
Answer 21: This truly depends on your science question more than it depends on the data.

Question 22: Are there models that can support both continuous and categorical questions? For example, if someone wanted to quantify forest stem volumes, but then also identify the component of those stems that are from live versus dead trees, would they have to run two separate models for each task or is there one that can do both?
Answer 22: There could be some ways to get this in one shot but most often this would be done in two steps. There are ways of creating multi-branch models where you feed the model the same input, and then in the multi-branch portion you can output several predictions of different types based on the last activation functions of the model. A simple example can be seen here:
https://towardsdatascience.com/building-a-multi-output-convolutional-neural-network-with-keras-ed24c7bc1178

Question 23: The biggest challenge in machine learning is data engineering. Are we working on, or do we have efforts on, creating benchmark datasets (for example, remote sensing for oil spill monitoring, etc.)?
Answer 23: There are big efforts towards creating benchmark datasets for many applications in the machine learning field related to Earth Science. Many of these have been geared towards classification applications, thus you would need to research online to see if there are any related to oil spill monitoring. Some places you can look at are github, huggingface, and even journals where there are mentions of these benchmark datasets for several applications. SpaceNet challenge has some disaster response datasets that could get you started as well.

Question 24: You mentioned in the Colab that there is a binary band or multi-band option (i.e., open water, river, etc.). Does the user select which band they want, or is this decided by the algorithm?
Answer 24: This was just an example for the exercise portion where if your training data had multiple classes you could choose between merging similar classes into water and not water, or by having multiple classes to predict on where these classes are different types of water.

Question 25: How do you define the hyper parameter space of Random Forest while tuning?
Answer 25: In session 3 we discuss automatic hyperparameter tuning. The overall idea to define the hyper parameter space is to select the available hyperparameters from your library (in this case scikit-learn), and to specify a high and lower bound for numerical parameters such as number of trees, and a list of options for discrete parameters such as the split criteria (Gini or Entropy in our classification example). You will use those with packages such as optuna to create the grid space for optimization.