Questions & Answers Part 3

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 Jordan A. Caraballo-Vega (jordan.a.caraballo-vega@nasa.gov).

Question 1: We've generated a feature importances chart using MDI and scikit learn. On the y-axis it shows "Mean decrease in impurity", running from 0.00 - 0.06. I'm unsure what these numbers mean exactly. How do I interpret this range?
Answer 1: Impurity-based feature importances works by measuring how much the overall impurity of the model (measured using our impurity metric such as Gini or entropy) decreases when a feature is used to make a split in a node in a tree in the forest. The larger the decrease in impurity, the more important that feature may be. The mean decrease is calculated by averaging the impurity decrease over all the trees in the forest.

Interpreting this is pretty straightforward: a higher value for the MDI for a corresponding feature is more important to the model (according to this method). It means the model relied more heavily on that feature to make predictions.

Gini/Entropy:
https://scikit-learn.org/stable/modules/tree.html#tree-mathematical-formulation

Permutation Importance and MDI:
https://scikit-learn.org/stable/auto_examples/inspection/plot_permutation_importance.html

Question 2: Kindly comment on how to tackle biases' and discrimination in selection of training of labeled data.
Answer 2: Basically there are some biases in all data. Very difficult to eliminate them entirely so the key thing is to recognize that they are there and do your best to explain them.

Question 3: Do you know about using genetic algorithm for model tuning and optimization? Is it sort of a random search?
Answer 3: The techniques we mention in this training are manual and automated techniques. Manual techniques, you use your domain knowledge of the algorithm and test combinations. Automated techniques then include random (you allow the optimization algorithm to pick random set of the hyperparameters), grid based (you give the optimization algorithm lists of possible parameters and iterate over those to find the best model), and bayesian (you use prior knowledge to improve upon the given parameters).

Question 4: Are the Precision, recall and F-score interchangeably used as accuracy assessment reports like Kappa, producer accuracy? Can you provide articles related to these parameters used as model evaluators in science of total environment?
Answer 4: Each one of these metrics look for different characteristics from your output. Precision will give you the accuracy of positive predictions, recall will give you the completeness of the positive predictions, F-score will give you a balance between precision and recall, while overall accuracy will simply care about the overall performance. Producer and user accuracy can be taken from the confusion metric you generate and will focus on omission error (producer's accuracy), and commission error (user's accuracy). Note that all of these metrics will give you a different view of the performance of the model allowing for more interpretability of your map errors.

Simple explanation of some metrics:
https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9
Some formal literature: https://doi.org/10.1016/j.rse.2014.02.015,
https://doi.org/10.1002/(SICI)1097-4571(199401)45:1<12::AID-ASI2>3.0.CO;2-L

Question 5: As a high school student learning ML and data analytics what are some good free and not free resources to understand ML programming better?
Answer 5:

Free course: https://www.coursera.org/specializations/machine-learning-introduction
https://developers.google.com/machine-learning
Book(s):

Machine Learning with PyTorch and Scikit-Learn (https://www.oreilly.com/library/view/machine-learning-with/9781801819312/)
Deep Learning (https://www.deeplearningbook.org/)
We wrote some materials to introduce ML concepts:

https://github.com/astg606/py_courses/tree/master/modules/machine_learning

https://www.coursera.org/learn/machine-learning-models-in-science

Interactive book w/ code samples (Deep learning specific): https://d2l.ai/

Question 6: import folium_helper. This line doesn't work, it returns the error: "ModuleNotFoundError Traceback (most recent call last) <ipython-input-9-ec8c4feabf6a> in <cell line: 31>()
29 import folium
30 from folium import plugins
---> 31 import folium_helper
32
33 from pprint import pprint

ModuleNotFoundError: No module named 'folium_helper'"
Answer 6: Note that there is a cell in the notebooks with "**!**wget https://raw.githubusercontent.com/NASAARSET/ARSET_ML_Fundamentals/main/src/folium_helper.py". This will download the folium_helper.py module. Make sure you run this cell before proceeding to run all other cells.

Question 7: The model should not detect rivers (water lines) as outputs as well? Those should be considered as water pixels, right?
Answer 7: The model is designed to detect "water". It won't matter to the model whether it is ocean, lake, or river. If you wanted to separate these types of water you would likely need a two step approach and/or some different methods.

Question 8: How do we determine whether the image produced has a false positive or not?

Answer 8: Once you produce an output map you need to validate that map using a validation dataset or manually with visual interpretation. These activities should identify false positives as well as false negatives.

Question 9: In robustness model, what do you mean by 'input can be noisy'?
Answer 9: In real world data scenarios (remote sensing data, etc.) the data is not perfect. There could be errors, inconsistencies, or outlier values in the data due to various factors such as instrument malfunctions, data acquisition errors, processing problems, etc. We refer to this data as noisy, since the data has been corrupted or contaminated, perhaps making it difficult for the model to make accurate predictions.

We want models to be “robust” to be able to still draw meaningful and accurate predictions despite the data being noisy.

Question 10: Do you know if Random Forest can also generate maps of probability that a cover is found in a specific location, i.e., instead of a regression or classification, get a map that goes from 0-1 where 1 is the highest probability of finding that type of class in a certain geographical area?
Answer 10: The Random Forest module from sckitlearn (and other libraries as well) provide the predict_proba options where instead of a categorical class, it can output a probability value from 0-1 for the observation given. The size of the geographical area will be given by the spatial resolution of your features.

Question 11: On the Assignment 1 Google Collab, when running the 3rd cell, I'm getting the error "Your session crashed for an unknown reason" and after looking at the app log I'm still having trouble identifying what caused it to crash. Is there an appropriate place to look for help with this?
Answer 11: Follow the instructions above each cell. Some of these cells will restart your kernel in order to install the GPU drivers needed for the session to run. You will need to run the cells one by one so all of the drivers can be installed and you can use the GPU version of the model. You can also download and run locally.

Question 12: Can this technique help to predict areas that will have floods and those that are likely to experience drought conditions now and in the future?
Answer 12: You could use this technique to generate a time series of maps that could be analyzed to determine whether the current conditions are a departure from “normal”

which can give you some clues to the onset of drought or flood. To be sure you would want to combine the maps with other information such as precipitation records to build confidence in the predictions.

Question 13: I may have missed it, but is it possible to use the Optuna library for XGBoost tuning, is there an example of XGBoost for regression based model construction and tuning available?
Answer 13: It is possible to use optuna with XGBoost. We do not have a colab notebook example but here are some examples from other sources:

https://xgboost.readthedocs.io/en/stable/python/survival-examples/aft_survival_demo_with_optuna.html

https://www.kaggle.com/code/alisultanov/regression-xgboost-optuna

Question 14: Is it possible to employ SHAP to eliminate some features that impact predictions negatively?
Answer 14: It is possible but not recommended. One reason why is that looking at individual explanations does not give us an accurate representation of how important that feature is to overall accuracy. While a feature may negatively affect a single prediction in one case, it may have a positive influence on other predictions.

Another reason it is not recommended is correlated features. SHAP considers each feature in isolation without taking into account the correlations between features. It is possible for a feature that has a low SHAP value on its own but may be important when combined with other features.

Question 15: For the explainability part, those SHAP results may be used to eliminate NDVI as a feature for the model?
Answer 15: See above answer. Remember, in this case we are looking at examples where the NDVI is anomalous. These cases show that NDVI may have a negative impact due to these anomalous values, however this is not indicative of the overall impact of NDVI on overall model performance. Permutation importance tells us that NDVI is important to model performance.

Question 16: With SHAP local explainability, where it was showing the influence of all of the variables, was it looking at individual pixels?
Answer 16: Correct, it was looking at a single row (pixel) of data.

Question 17: NDVI values greater than 1 (10000) due to atmospheric correction? Is it possible that the high NDVI values represent some vegetation within the areas classified as water?
Answer 17: NDVI > 1 can happen if both the “red” and “NIR” values are very low. This case results in anomalous calculations of NDVI. These can happen using Top Of Atm data as well as the surface reflectance data we used in these exercises. The values don’t happen frequently but do need to be captured in your algorithm to avoid unusual results in your output maps.

Question 18: In the example of model tuning, the range of precisions generated by different hyperparameterizations was very small - it didn't seem to be making a big difference to the precision. In your experience, maybe with more complex classifications, how much difference in model performance can we expect by tuning parameters?
Answer 18: The influence of hyperparameters can be attributed to both the dataset we are working with and the models we are using. In this case, we have small amounts of data and the models seem to pick up on an accurate decision boundary for the water/not-water problem. This leads to the performance not changing much despite changing the hyperparameters.

In a more complicated classification (or regression) problem the hyperparameters can have a greater impact on the model. Hyperparameters control model complexity, regularization, etc. All of these will help you adapt the model to the complexity of the problem. If a problem is more complex or harder for the model to capture, hyperparameters will help you perform this adaptation.

Different model architectures also are more or less affected by the hyperparameters. Models like random forests work pretty well out of the box with little change in hyperparameters. Other models such as XGBoost can be greatly affected by the hyperparameters chosen.

Question 19: Can we use samples collected with points or polygons in the Google Earth Engine to run these classifier examples? How did you select your samples?
Answer 19: The tabular points are available to you in the hugging face URL and yes you could take those into GEE to try them out there. The dataset itself is quite small so that it can produce results very quickly for the demo. You would want to add training if you are trying to produce a more generalized model. This can be done manually or with tools such as image segmentation (or other unsupervised methods) to identify some additional samples to include in your training.

Question 20: Are there auto-ML packages containing DL models?
Answer 20: Yes, there is https://github.com/automl/Auto-PyTorch, https://github.com/rafiqhasan/auto-tensorflow (a simpler form of AutoML), auto-keras and others.

Question 21: Question about the joblib library, for the last several example Jupyter notebooks, I noticed the joblib library is used to save and load models constructed. Is this more efficient (in terms of run time expense to load) way than pickle format based serialization of ML model and then de-serialization?
Answer 21: Joblib uses its own implementation of pickle with some special additions. Sklearn recommends using joblib as it is more efficient in serializing objects that contain large numpy arrays internally (as is the case for a lot of sklearn estimators). Pickle can also be used (it is often faster for smaller objects as it is implemented in pure C)

https://scikit-learn.org/stable/model_persistence.html

Question 22: Why do you reshape after the prediction? Why not before prediction?
Answer 22: The data comes in array format, we reshape into tabular format to perform the prediction using the model, the output is now in tabular format, and we then reshape into array format so we can output the predictions into a raster we can visualize.

Question 23: Can you ask the instructor to run MODIS_autoML in Session 3? There is a runtime error.

Answer 23: The notebook will take about 30 minutes to complete when training the model and doing the prediction. Make sure to restart the kernel and run all of the cells. At this time, we are not able to reproduce the error.

Question 24: Can we apply this technique to single scene satellite images from sentinel or landsat? Can we apply the autoML for classifying one scene image of sentinel data for specific location?
Answer 24: Absolutely, the method shown here is very generic. However you would need to retrain the model with spectral values from sentinel or Landsat so that it can learn the spectral response of those instruments. Though the different instruments have the same “band” names (i.e. red, green, blue, etc.) the actual sensor wavelengths and characteristics are inherently different between the instruments which will result in differences in how the model gets trained. You can apply a trained model to a different instrument (assuming it has all of the same bands available) but you shouldn’t expect perfect (or even good) results unless you do some amount of retraining.

Question 25: The notebook failed to install the dependencies for me.
Answer 25: Make sure you restart your session, and run all cells. At this time, we are not able to reproduce the error.

Question 26: How can we classify raw satellite images together with NDVI, NDWI, EVI and SAVI?
Answer 26: From the given information in this question, the basic answer is to calculate the indices and add them to the data stack (array with data in it) and then apply your algorithm to this stack. Need more info if they were looking for a different answer.

Question 27: How can we apply predictions to unlabeled data (for example, if I download data from MODIS directly)? How can I apply NDVI to these satellite images?
Answer 27: In the notebook given during this exercise, we calculate NDVI directly from the notebook. Take a look at the functions that are on the exercise and feel free to add new images to your fileList variable in the notebook so you can predict using other MODIS tiles.

Question 28: Does hyperparameter tuning lead to overfitting?

Answer 28: No, it should decrease the possibility of overfitting since you are optimizing using cross validation. Overfitting might be reached even when using hyperparameter tuning techniques if your dataset is not ideal, but is not directly related to the exercise of hyperparameter tuning.

Question 29: For operational type work where you would want to update and run a model daily, what workflow would you suggest? For example, do you commonly set the time in auto AI to generate an output?
Answer 29: You could use any number of different tools to run an algorithm on a schedule. In the linux/unix world something like a cron job could do this. In the commercial cloud something like Lambda can do this. Windows probably has some type of scheduler as well.

Question 30: How can we select reference pixel values for confusion matrix when we do not have in situ data? What is the percentage of the pixels selected for confusion matrix out of total classified pixels?
Answer 30: This can be done with a reference data set from another source or can be done by visual interpretation within a viewer. There are references for how many pixels should be selected for validation. I tend to go with the Oloffson method as the standard for validation.

https://doi.org/10.1016/j.rse.2014.02.015

Question 31: My question (7) was related to the performance of the model we saw. Visually, we could expect the model to return lines extending from the produce areas. Do you think this means we still could improve the model?
Answer 31: Yes, visually you can see we improve our model using hyperparameter tuning, but even that is not enough, thus the next step is to increase our dataset. The dataset used in these exercises is extremely small to make it manageable to run in Google Colab. A real dataset for classifying water over MODIS data will definitely have many more observations than the 1000 we used for the exercises.

Question 32: Can we use unsupervised models? As in real life scenarios we may have to classify data that is not labeled.
Answer 32: Of course unsupervised models can always be tried and one should try many different methods to optimize the results for your area. Typically the unsupervised

methods will identify that there is a “feature” but it cannot tell you what that feature is since it doesn’t have labels. I would say the model gives you an “inference”, but that inference requires further interpretation by the operator to give it meaning.

Question 33: What is the difference between python and machine learning?
Answer 33: Python is a programming language that can be used to enable machine learning. Machine learning is a discipline that covers many techniques to allow the machine to automatically learn from data, improve performance from experiences, and to predict things without being explicitly programmed.

Question 34: Which machine learning algorithms require normalized input data, and how can it be done in Python? How can the performance of a machine learning model be assessed on different landcovers, elevations, etc.?
Answer 34:

Distance-based algorithms like neural networks require normalized input data. sckitlearn for example has multiple scalers to deal with this: https://scikit-learn.org/stable/modules/preprocessing.html

Methods of evaluation will be identical to what we showed here regardless of how many different land cover classes you have represented in your outputs. You will need either a reference data set or visual interpretation to assess point based data.

Question 35: Can we use this example for crop type classification, and do you have a good example can you share?
Answer 35: Sure, the methods would be very similar. You would need to provide your own labels to train on to get crop types. I suggest reaching out to the NASA Harvest program for support in this area.