COS40007 Artificial Intelligence for Engineering

Week 4 Studio Activities

ILO	Understand the Evaluation Measure of Machine Learning model.
Aim	• Learn how to find different evaluation measures of ML models • Learn how to compare ML models • Learn how to build AI from ML
Resources	Books: 1. Prosise, Jeff. Applied machine learning and AI for engineers. " O'Reilly Media, Inc.", 2022. 2. Raschka, Sebastian, Yuxi Hayden Liu, and Vahid Mirjalili. Machine Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python. Packt Publishing Ltd, 2022. Web Resources: 1. https://scikit-learn.org/stable/modules/model_evaluation.html
Requirements for to be marked as complete	Demonstrate the table of outcomes to your tutor

Disclaimer: The dataset used in this studio was originally collected for a funded research project by the Australian Meat Processor Corporation. The dataset here is used solely for educational purposes and can only be used to complete activities for this studio. By any means, this dataset is not shareable with others or in any public domain.

Dataset: We will use the same dataset we used in Studio 3. If you have not completed Studio 3, please return and complete Studio 3. Otherwise, download the dataset from Studio 3.

Evaluation metrics

Check this link: https://scikit-learn.org/stable/modules/model_evaluation.html

Several evaluation measures have been used to compare different ML models. The evaluation typically depends on the target problem. A common evaluation measure when you solve classification problems is accuracy, precision, recall, and F1 score, as well as a confusion matrix to visualise the evaluation.

Studio Activity 1: Group formation for Design project
The overview and the list of topics for the design project are now available on Canvas. If you have not already reviewed the design project specification document, please read it. This is available under Module> Design Project.

You must form a group of 5 (6 in exceptional cases) to complete the project. If you already have a group to form, this is great. Otherwise, if you need help with group formation, please get assistance from your tutor. Once you have finalised your group, please submit the form. The form link can be found in the design project specification document.

Studio Activity 2: Classification Report
Check this function: https://scikit-learn.org/stable/modules/model_evaluation.html#classification-report

Now, go back to your Studio 3 outcome. In Activity 2, you measured the performance of the ML model for train-test split and cross-validation using only accuracy. In Activity 7, we measured the accuracy of different ML models using the original dataset. Now, let us expand our evaluation outcome. We will conduct this evaluation exercise using original data only for the train-test split (y_test and y_pred).

Now, compute the evaluation outcome of 4 models (SVM, SGD, RandomForest and MLP) using a classification report and answer the following questions.

1) Which model has the best F1 score in terms of weighted average? Is this different from the accuracy that you obtained in Studio 3?
2) Which model has the best precision value
3) Which model has the best recall value
4) Which model has the best accuracy value
5) Which class has poor performance across all models? How did you determine this?

Studio Activity 3: Confusion Matrix

A good way to understand your evaluation outcome is to visualise them in a confusion matrix. You will also get an idea of the class distribution of your test set. This also helps you to understand true positives, false positives, true negatives and false negative classes.

Now plot the confusion matrix of 4 models. From your observation, where are the false positives coming from? How you can fix it?

Studio Activity 4: Class balancing: Undersampling and oversampling

This dataset has a class imbalance issue, so that the model may overfit the majority class. You can fix this issue using undersampling and oversampling. If you look at the distribution of classes, you will see that we have the majority of samples from class 2 and a few from class 1. So, let us fix the issue in our dataset.

Now generate a resample version of the dataset by

1) Oversampling the minority classes using SMOTE
   And
2) Undersampling Majority class using Tomeklinks

Now, train and test your best classifier with the resampled version of the dataset. Computer the classification report and visualise the confusion matrix. What did you observe?

Repeat the above process, but now adjust class weights (e.g., 0.2 for class 2, 0.5 for class 1 and 0.3 for class 0)

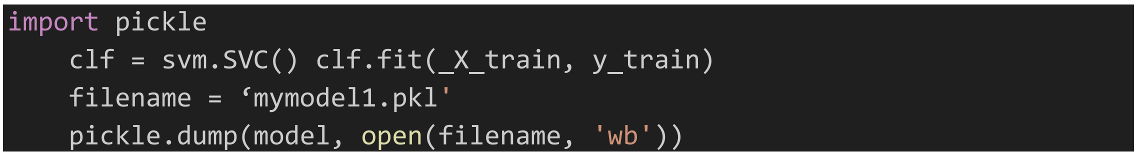
Which approach resulted in better evaluation performance?

Studio Activity 5: Saving ML model

This is very similar to Studio 3 activity. You need to create an SVM model using data only from 3 workers (*w1, w2* and *w3*) with the following criteria. Furthermore, keep w4 data aside.

1) Merge *w1,w2,w3* data
2) Use undersampling and oversampling to fix the class distribution
3) Use train-test split (70/30)
4) Generate classification report
5) Visualise confusing matrix
6) Now save the model that you generated

Sample code



```
import pickle
    clf = svm.SVC() clf.fit(_X_train, y_train)
    filename = ‘mymodel1.pkl'
    pickle.dump(model, open(filename, 'wb'))
```

Do the similar thing as above, but now generate and save a RandomForest model (as mymodel2.pkl)

Studio Activity 6: AI from ML

In Activity 4, you trained ML models using data from 3 workers. Now, you want to build AI using the trained model for a new worker. So, the model has never seen data from the worker before because we have not used the data from train and validation. Say we want to use the model in real-life deployment settings. That is, assume your model is running on some machine, and the worker's activity data comes in real-time every second. You want to use your model to get the data and find the activity that the worker is doing. This then becomes AI. Now, we will use w4 data to test our first AI deployment.

1. First, you need to load the model that you saved. You can do this using pickle

```
model = pickle.load(open(filename, 'rb'))
```

2. Now load the w4.csv data in a data frame and separate feature (X) and class (Y).

3. Now write a for loop that initialises a counter starting from 0

   i. Iteratively read 1 line from X and Y (say x_i,y_i for i-th iteration)
   ii. Apply model to predict class y_pred_i = model.predict(x_i)
   iii. If s y_pred_i matches with y_i, increment the counter.

So, at the end of the loop, the value in the counter will be the number of correct classifications you obtained for w4 data.

If you divide the counter by the length of Y, then you will basically find the accuracy for the w4 data.

4. Now repeat the above the model2 (random forest) and measure the accuracy value again for the model2.

Which model has given you better performance? Have you observed different results that you obtained in the model evaluation (using a classification report) or different?

Studio Activity 7: Understand ML model construction

Sometimes, you want to understand how your ML model is constructed and how your model reaches a decision point. This provides an explainability of your outcome prediction. A decision tree is a simple classifier to visualise such a measure.

1) Using the data generated in Activity 3 after sampling, get the data with 10 features using mutual_info_classif — scikit-learn 1.5.1 documentation.
2) Now build a Decision Tree classifier using 10 selected features and measure the accuracy (use train-test split)
3) Save your model
4) Load the model
5) Plot the tree using plot_tree
6) Plot tree using export_text

Recommended readings

Check the following links to visualise other ML models that you used before.

1) Support Vector Machine
2) Stochastic Gradient Decent
3) Random Forest

Next Steps:
The assessment Task for Week 4 can now be attempted and submitted via Canvas.