**Progress Update – 13 July 2022**

**Action: Updated the interaction sequence specification**

An interaction sequence has the following structure:

* 0 or 1 to indicate that an interactive slide has not / has been completed.
* N, F, P to indicate that a problem slide has no submission, resulted in a failed submission, or a resulted in a successful submission. (0 1 2)
* **Example I:** 1 1 1 P P 1 1 1 P P is an interaction sequence for a module with 10 slides. It indicates that the student completed all interactive slides (represented by 1s) and successfully passed all four problems in the module (represented by Ps).
* **Example II:** 0 0 1 F F 0 0 0 P P is an interaction sequence for a module with 10 slides. It indicates that the student did not complete first two slides, attempted but did not pass the first two problems, did not complete the next three slides, and passed the last two problems.

**Action: Extended the interaction sequence to the entire module**

Last time, the interaction sequence was restricted to the first set of problems. Now, the interaction sequence has been extended to cover the entire problem (two sets of problems).

**Action: Extended the analysis across all modules with interactive slides**

Note: The following modules have **interactive slides**

["w1p1", "w1p2", "w2p1", "w2p2", "w3p1", "w3p2", "w4p1", "w4p2", "w5p1", "w5p2"]

Across all problems investigated, the most common **outcome** is (P, P, P, P) followed by (N, N, N, N). That is, students either complete all problems successfully or do not attempt any problems at all.

Across all problems investigated, the most common interaction **sequences** belonged to three groups:

* All 0s besides Ps for problem slides – students who did not complete any of the interactive slides but passed all problems
* All 0s besides Ns for problem students – students who did not complete any interactive slides or problems.
* Mostly (or all) 1s for interactive slides and Ps for problem slides – students who completed most (or all) interactive slides and passed all problems.

Note: Other outcomes (those that contributed less than 5% to the total count) were aggregated into a separate category called ‘Other’.

Double check if other courses have interactive slides – would be interesting to explore. Expects the beginner course to beginner to other courses esp.

**Action: Extracted features and conduct correlative analysis**

I attempted to extract some features from the interaction sequences. Namely, I extracted:

1. Number of 1s – number of slides completed
2. Number of Ps – number of problem passed (0 – 4)
3. Number of Fs – number of slides failed
4. Number of Ns – number of slides not attempted.

I did a correlative analysis (correlation matrix) of pairs of variables, and found no strong correlations, indicating that they are all largely independent. The problem however with this type of analysis is that it is on discrete rather than continuous data, and the number of 1s, Ps, Fs, Ns is always capped. Ideally, what I wanted to see was relationship between slide completion (number of 1s) and number of slides passed (number of Ps).

Predict: Given this sequence, what will be the next state.

**Next steps**

* **Answer the following questions**
  + **How commonly do students follow the same interaction sequence** (e.g. there are three common interaction sequences identified previously. Do students follow the same sequence most of the time? Another way of saying this is how consistent is student behaviour?)
  + **What are some other types of encoding that are more descriptive e.g. capture slide completion order?**
  + **Without order we can say that they completed most of the slides. This type of descriptive characteristics – average completion.**
  + **Summary of the sequence**
* **Extract outcomes and use interaction strings to predict outcomes**

**Prediction (Monday)**

**Prediction can be done in different ways –** we can encode previous sequence and predict the next sequence/value. There are other temporal models that are useful – Hidden Markov Models can be used to predict the next state. In Sophia’s thesis, we saw how many students go from one state to another.

**How to formulate prediction task:** we have standard machine learning methods and HMM. For standard ML models, we use the sequence (10 previous values) and we predict the next one.   
We have score associated with each module, and overall score at the end of the course.   
  
**Bryn’s idea:** Predict engagement – who will drop out and who will continue.   
We could predict final mark, probability of dropping out – we could ignore problems and just take slides. Based on the sequence of slides, can we predict the final mark (sum of the problem slide marks – score out of 400). We could discretise this – HD, D, CR, PS. It might be difficult to predict exact value, but we could predict the range. Firstly, we need to look at final mark distribution – what’s the histogram – and then decide how to discretise. There is no final exam score.

**Hypothesis:** Students who complete all slides will do well, the outliers will be students who don’t complete slides and do well.

**Another prediction task:** Can we predict performance on the last task (most complicated), given performance on all previous modules (interaction sequence – includes tasks and slides/slides only/tasks only) – look at all past interaction (not just the modules interaction sequence) and predict outcome on given module.

**Task:** Can we identify key tasks/modules – more difficult/comprehensive tasks – important milestone (find another word). We try to predict the performance at the end based on all the interaction sequences in key tasks/modules (Week 3). We need to get more information – we need to a longer history.

The above task: For a given milestone (e.g. Week 3), take the previous interaction sequences (including the tasks – the whole sequence) (Wk 0 – Wk 2), and then we predict the outcomes of Week 3 (P P P P). What would we predict P, N, F, and then predict marks.

We can discretise this – high distinction, distinction/credit, fail – look at distribution of marks and set the intervals.

Week 3 looks good because it’s in the middle. And then look at Week 5.

**LOOK INTO ADVANCED COURSE**

**TASKS:**

1. Beginner’s course – continue with this at the moment. Look at all modules and identify two milestone tasks e.g. Week 3 & 5.
2. Look at distribution of the marks for Week 3 and 5. Look at proper discretisation to create different groups of students (HD/D, /CR/P, F).
3. Do the prediction task
   1. Use the previous sequence and predict the score.
   2. Length of string is not an issue.
4. Do the above for the advanced course.

How will we apply a NN? We need to encode this information in a suitable way.

We have nominal variables – P, F, N, 0, 1 – treat them as nominal. We apply a neural network to nominal data. We can apply DT as well.

We could also apply temporal neural networks. Firstly, we need to think about how to represent the data, and then we apply different models. Send Irena document/finish it. Will think again about representation and how to use for prediction task.

On Holidays til July 26 – skipping two weeks – send an update next Tuesday – extract data and send couple of feature vectors.  
  
Q: Will we have the same sequence being mapped to different outcomes? Investigate.

Send Irena – dimensionality of the data for the first 3 weeks (how many interactive slides we have), how many problems, how many students (~30 slides/length of 30) – send Irena feature vectors and examples when I finish extraction. Irena will think about sequential modules – we need to take into account temporal aspect of data. Two part thesis – slide completion first, and second part – temporal information taken into account.

Set up the prediction process, and predict outcomes based on interaction sequences for the binary encoding example only. Do it for the individual problem, module and challenge and compare. ☐

# Thesis Ideas

**Thesis workflow**

**Data extraction**

* Explain the dataset: instances, features, which features are important (by literature and CFS)
* Explain the process to extract and clean the data. In particular, explain which variables of the dataset were used to work out interaction sequences.
* Explain any other processes performed on the data e.g. aggregation, de-aggregation
* Explain what variables were removed
* Note: relate discussion of the dataset to literature

**Data transformation**

* Explain how the chosen data variables were transformed/encoded into interaction sequences.
* Explain the **different encodings**, and their strengths and weaknesses e.g. time complexity of the different algorithms.

**Feature extraction**

* Explain how features are extracted from the interaction sequences, and relate the selection of these features back to literature.
* Explain the positives and negatives of **different features** and metrics.
* Perform an analysis on feature correlation.

**Prediction**

* Predict student performance with different metrics (e.g. final exam mark, final course mark, module mark, outcome).
* Use different prediction methods: clustering, Naive Bayes, Decision Trees etc. – look into different methods.
* **Other prediction tasks**
  + Using early behaviour to predict later behaviour
  + Predicting at risk behaviour
  + Think about other ML tasks

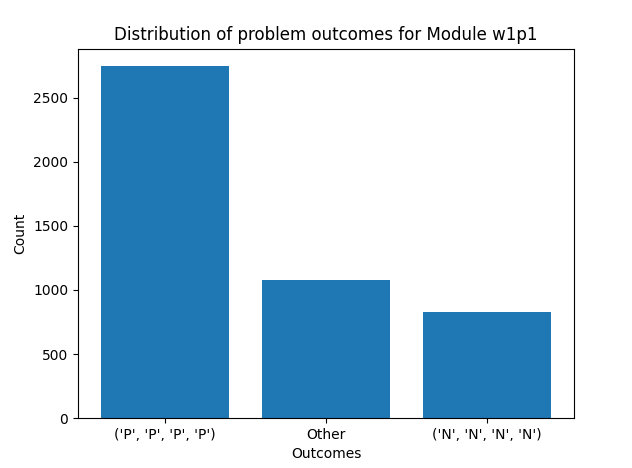
**Evaluation and comparison**

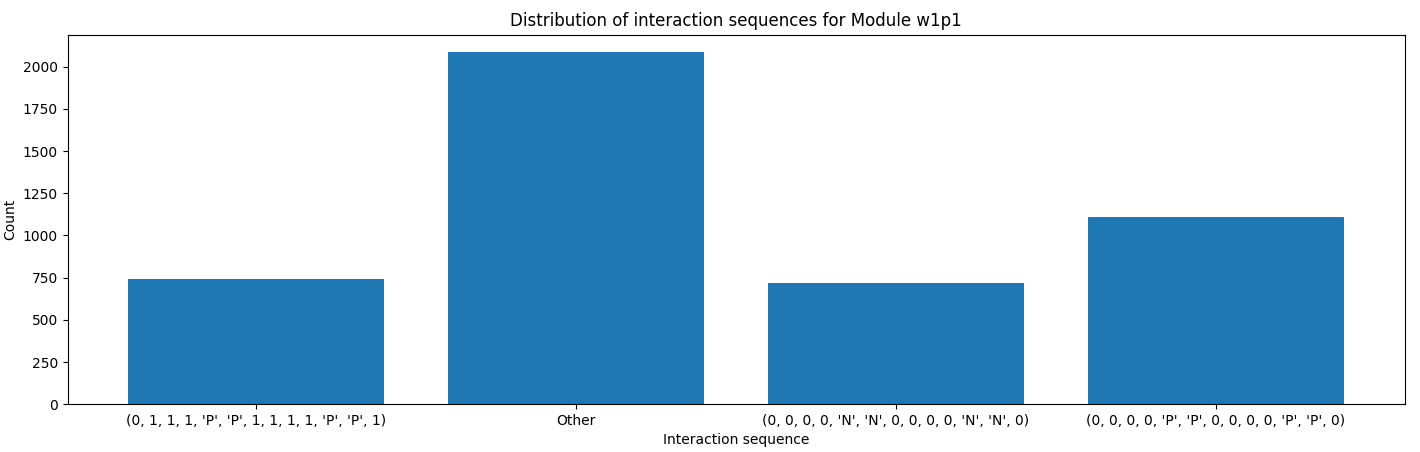
* Compare
  + Different encodings
  + Different features extracted
  + Different performance metrics
  + Different prediction methods
  + Different datasets and problems

**Note:** Any analysis by module can be aggregated later

USEFUL IDEAS

* <https://peterroelants.github.io/posts/rnn-implementation-part02/>





Notes

* This kind of task does lend itself to aggregating data. We don’t want a situation where we have to consider every single situation, because we don’t have enough data for all situations. Like logistical regression, you chunk your data down – different ways of doing that.
* Only thing that’s useful about first slide is narrate button. There is some processing to do to identify whether a slide is actually a useful slide/discriminating slide for the data. If a small proportion of people actually complete it – does that slide have much predictive power on its own. If not, probably don’t include.
* This says something about models that you could apply
  + Naive Bayes – flexible, deal with categorical data and numerical data.
  + Often outperforms surprisingly well.
* 2 ways
  + Preprocessing
  + Naive Bayes that places less weight on slides
* One reason why prefer preprocessing
  + Don’t want first slide of all modules to have less weight, just want first slide of that module
  + Two steps to the process
    - For a module, what are the important slides to consider. Chuck away any slide for which only a small fraction of people took part in that – and relate that back to context e.g. narrative slide
    - This is why we do a per module prediction task (then we might do a global task).
    - **AIES CONFERENCE – DEADLINE IN OCTOBER – TRY TO SUBMIT A PAPER**
    - **DO IT BY EMAIL NEXT WEEK –**
    - **MAKE NOTES AVAILABLE**

