Agent Based Modelling in Reception Classrooms

Peter Tymms

Peter Tymms is an emeritus Professor from Education at Durham University.

His main research interests include monitoring, assessment, performance indicators, ADHD, reading and research methodology. The PIPS project, which is designed to monitor the affective and cognitive progress of children through primary schools starting with a computer adaptive on-entry baseline assessment.

Can you give us an overview of the project?

Working with Professor Chris Brown, School of Education, and later Alison Clarke and Mark Turner of Advanced Research Computing, the project seeks to set up an agent based model (ABM) which can reproduce the patterns of progress that are seen in real world data. This includes maths means at the pupil and class level together with their variances.

We have real world data on pupils at the beginning and end of Reception classes at the age of 4, in England. The data come from 51,569 pupils in 2,158 classes and includes measures of maths attainment and background data for each pupil.

Did you work with a research software engineer (RSE) from the start of the project?

I started the project using NetLogo on my laptop but found that the project took a long time to run and, as an amateur, I lacked the expertise necessary to take the project to the next stage.

N8 CIR
COMPUTATIONALLY INTENSIVE RESEARCH

What are the benefits of working with an RSE?

Generally the involvement of professional expertise and understanding of systems has been vital to the project.

Alison and Mark's work helped to optimise the code for use on Durham's supercomputer which in turn enabled the analysis of more data and with shorter run times.

What tools and software did you use for your analysis?

The core simulation engine is written in Python using the Mesa ABM (Agent Based Modelling) framework. Python is more expressive than NetLogo and, with Mesa, offered good scalability across large numbers of threads, allowing us to leverage the compute resources on Hamilton 8, one of Durham's supercomputers, more effectively.

We automate several post-processing strategies with MLwiN. They allow us to run a multilevel model and calculate mean squared errors for our output data. The input data, code and documentation is open source at: **https://github.com/DurhamARC/classroom-abm**.

It is a single click pipeline on Hamilton 8, but some minor changes to the configuration settings will have to be made if users deploy the ABM on another supercomputer.

Finally, to make the model accessible and usable in training settings, the model has a web front end that is distributed through Heroku. Check it out here: **https://classroom-abm.herokuapp.com/#**.

Having worked with an RSE, will it change your approach in the future?

I will seek help earlier next time.

The knowledge of different programming languages that exists within RSE teams has been invaluable. Combined with the experience of accessing and using supercomputing platforms it has enabled the research to be scaled-up relatively quickly. Something that would have been impossible on a laptop!