Starters and Finishers

Rachael Sanderson

6 November 2019

# *The Intended Analysis*

This initial analysis will reflect on the decline of participants within the course, to consider which steps have the steepest decline. With this analysis, ideally a further understanding will be gained about which stages of teaching attract the most participents, and so which elements need to be improved to maintain student participation and so overall cohort attainment.

# *The Applied Datasets*

The dataset used was the recorded “Step Activity” for each run of the course. This dataset considers of a start date and time for every participent (identified by a unique learner ID) for each step of the course. However, whilst the structure of the dataset is the same for each run of the course, the steps are different for runs 1 and 2. Therefore, for consistancy of the temporally based analysis, this stage of the analysis was only undertaken for runs 3-7 of the course, as these have identical step progressions through and so allow a more direct comparison. The goal of this stage was to investigate whether there are any repeated trends or clear divergences in the decline of individuals completing steps of the course, to support the business goal of having as many students successfully complete the course as possible.

# *Data Preparation*

The following function was applied to the “step activity” file for each run of the course produce a dataset for analysis. Similar functions have been applied throughout the CRISP-DM process, to produce the variety of dataframes needed within the report, with adjustments made to fit the objectives of the analysis.

QuantitiesStarting.function = function(x){ #x = file selected to analyse  
 StepActivity = x  
 Steps=c((StepActivity$week\_number)+((StepActivity$step\_number)/100)) #acknowledging data complication (3.1 matches 3.10)  
 StepActivity = cbind(StepActivity, Steps) #editing the data with the new column  
 S= unique(StepActivity$Steps, incomparables = FALSE) #list the unique steps  
 L =length(S) #How many steps are there?  
   
#making the vector of quantities for each step  
   
 total\_values = vector() #making the vector  
 for(i in 1:L){ #for loop: L = how many steps will be assessed/length of vector  
 Step= StepActivity %>% filter(Steps==S[i]) #filter: group the rows for a single step  
 Quant = nrow(Step) #how many learner IDs were collected for that step?  
 total\_values[i]= Quant} #compile a vector of how many people started each step  
  
#making the dataframe to be able to chart: Step against how many participants.  
   
DFActivity = data.frame(Step = S, Total = total\_values)  
   
 return(DFActivity)} #return: the final dataframe for the submitted course run

This function then counts how many people started each step, to produce a data frame that can then be evaluated. It can further be manipulated to identify the total finishers, or the difference between these two figures, as will be demonstrated further.

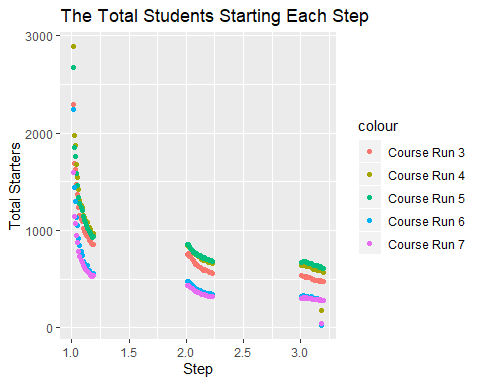
# *Analysis*

## *Part One: How many people starting at each step?*

The first analysis was to record how many people are recorded as starting at each step, to reflect on how this declines. This was considered for each course, to consider if the trends were comparable for each run. With an awareness that the number of enrolments varies each year, there was a realisation that this data needs to be standardised to acknowledge this. Therefore, the data was represented as a proportion. Within this process, the code as above was repeated, with the y axis instead being the Y value / number of course starters.

The two can be modelled as below, with exact figures (left) and proportions (right):

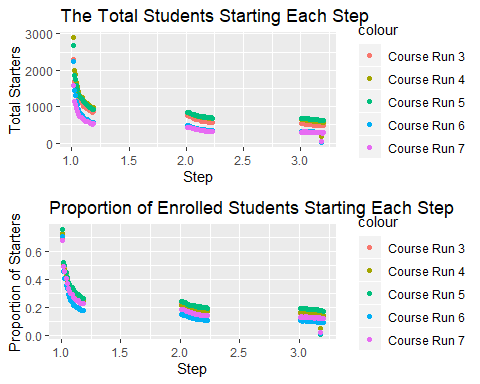
Plot = ggplot (data = CourseS3,aes (x = X, y = Total))  
CourseS3Graph =Plot +geom\_point (aes (x= Step, y = CourseS3$Total, colour = "Course Run 3"))  
CourseS4Graph = CourseS3Graph + geom\_point (aes(x = Step, y = CourseS4$Total, colour = "Course Run 4"))  
CourseS5Graph = CourseS4Graph +geom\_point (aes (x= Step, y = CourseS5$Total, colour = "Course Run 5"))  
CourseS6Graph = CourseS5Graph + geom\_point (aes(x = Step, y = CourseS6$Total, colour = "Course Run 6"))  
CourseS7Graph = CourseS6Graph + geom\_point (aes(x = Step, y = CourseS7$Total, colour = "Course Run 7"))  
CourseS8Graph = CourseS7Graph +labs(title="The Total Students Starting Each Step",  
 x ="Step", y = "Total Starters")  
CourseS8Graph

 Alternatively, a graph of each quantity of starters as a proportion of the number of people enrolled on the course can be constructed:

StartProp = ggplot (data = CourseS3,aes (x = X, y = Total))  
CourseSP3Graph =StartProp +geom\_point (aes (x= Step, y = CourseS3$Total/(Cohort\_Summaries$Entries[3]), colour = "Course Run 3"))  
CourseSP4Graph = CourseSP3Graph + geom\_point (aes(x = Step, y = CourseS4$Total/(Cohort\_Summaries$Entries[4]), colour = "Course Run 4")) #.....

The two can be modelled as below, with exact figures (left) and proportions (right):

grid.arrange(CourseS8Graph, CourseSP8Graph)



When comparing these graphs, the trends are very similar, with a gradual decline in starters at each step, and a gradual slowing of this decline as the steps progress. However, the selection of this data makes the model flawed; just because an individual started a step, does not mean that they completed it, therefore this analyisis cannot sufficiently meet the objectives of the exploration.

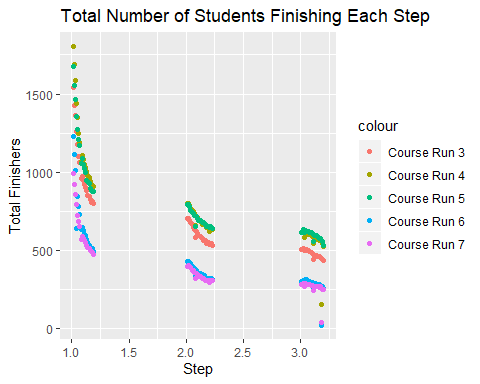
## *Part Two: How many are finishing at each step?*

The data was re-examined to consider the additional column of ‘last\_completed\_at’. The assumption was made that when this ‘completed’ column is empty, it refers to the people who *did not finish a step* and so would have not completed the course, despite starting that section. If a step has a lower number of finishers, it is potential responsible for a higher number of dropouts - something that the educational programme wants to reduce. The data frame of finishers was produced in the same way as for the starters, with the addition of a single line of code at the start:

StepActivity = cyber.security.7.step.activity  
 Filter = StepActivity %>% filter(last\_completed\_at !="")  
 StepActivity=data.frame(Filter)

This code removes the rows that are empty in the “last\_completed\_at” to produce a data sample that only contains the people that are recorded as finishing each stage. The following graph was hence produced:

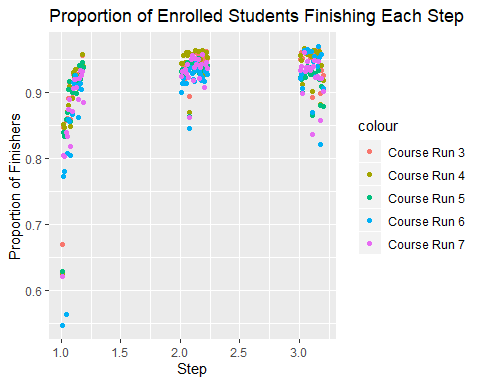
Finishers = ggplot (data =CourseF3,aes (x = X, y = Total))  
CourseF3Graph =Finishers +geom\_point (aes (x= Step, y = CourseF3$Total, colour = "Course Run 3"))  
CourseF4Graph = CourseF3Graph + geom\_point (aes(x = Step, y = CourseF4$Total, colour = "Course Run 4"))  
CourseF5Graph = CourseF4Graph +geom\_point (aes (x= Step, y = CourseF5$Total, colour = "Course Run 5"))  
CourseF6Graph = CourseF5Graph + geom\_point (aes(x = Step, y = CourseF6$Total, colour = "Course Run 6"))  
CourseF7Graph = CourseF6Graph + geom\_point (aes(x = Step, y = CourseF7$Total, colour = "Course Run 7"))  
CourseF8Graph = CourseF7Graph +labs(title="Total Number of Students Finishing Each Step",  
 x ="Step", y = "Total Finishers")  
  
CourseF8Graph



However, this model is of limited benefit to the analysis, as it fails to consider the impact of varying enrolement levels for each year. To negate this, the Y axis was adapted to represent the number of finishers as a proportion of the number of starters, dividing the finishing total by the starting total;

FinishProp = ggplot (data = CourseF3,aes (x = Step, y = Proportion\_Finishers)) #outlining graph parameters  
CourseFP3Graph = FinishProp + geom\_point (aes (x= Step, y = CourseF3$Total/CourseS3$Total, colour = "Course Run 3")) #adding each layer as previously - conntinue until CourseFP8

The final graph produced can be seen below;



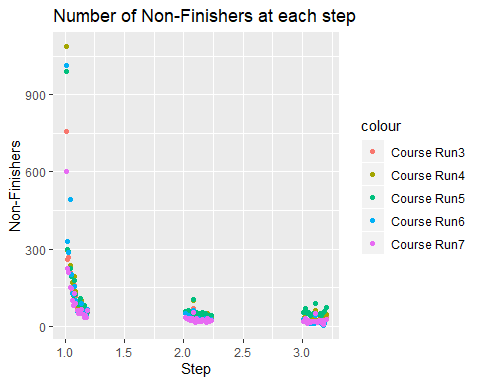
The graph shows limited correlation for each week, with each grouping being fairly clustered. There does not seem to be a particular step with a higher proportion of non-finishers throughout every run of the course, although noticably the cluster for stage 1 has a more linear trend than stages 2 and 3, implying that more people tend to leave a step incomplete when they are first introduced to the content in the earlier stages, with the cohort settling as they are embedded in the course. The model itself is clear and so easy to read, with the proportion suitably allowing for variations in sample size. In reflecting on the difference in the chart when the proportion of finishers is considered, the analysis led to the consideration of both starters and finishers.

## *Part Three: Contrasting Starters and Finishers*

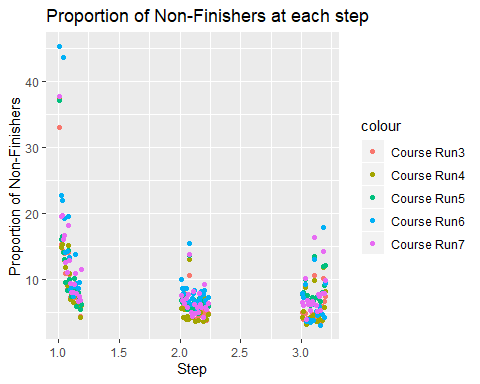
The analysis focused on the difference between the two columns, and so began with the following data frame;

DifferenceDF = data.frame(Step = CourseS3$Step,   
 Difference3 = (CourseS3$Total-CourseF3$Total),  
 Difference4 = (CourseS4$Total-CourseF4$Total),  
 Difference5 = (CourseS5$Total-CourseF5$Total),  
 Difference6 = (CourseS6$Total-CourseF6$Total),  
 Difference7 = (CourseS7$Total-CourseF7$Total))

To plot the following graph;



The previous analysis stage showed the benefits of including information about the number of starters, so the graph was adjusted to show proportions accordingly;



The graph of the percentages reflects a similar scatter to the number of finishers, which implies consistancy in the methods used to produce the models. However as a visualisation it is more effective. The difference calculated is the number of people that do not finish a step: this is the quantity that the educational provider wishes to reduce, hence meeting the business goals. However, in this graph the higher values are the ones that demand attention as the steps with the largest proportion of students dropping out. Therefore, as a final model to reflect on the proportion of students that do not complete each step, it is the most useful.

### *Discrepencies: What does it mean to “Finish”?*

Returning to the finishers graph (see below), an anomoly was revealed, as certain steps noticably have less finishers than a step after it, implying that ‘not completing’ a step may not mean completing the course; this was investigated further.

