Continuous Assignment for

**Programming For Big Data**

**B8IT105**

Date of Submission: 26th November 2017

**CA4 - Perform Analysis on a 5000 line Dataset**

Ciara M Quinn

10358638

Table of Contents

[1. Overview 3](#_Toc499471099)

[1.1 Outline 3](#_Toc499471100)

[2. Process 3](#_Toc499471101)

[2.1 Input File 3](#_Toc499471102)

[2.2 Python Processing 3](#_Toc499471103)

[2.3 Output 4](#_Toc499471104)

[3. Analysis 4](#_Toc499471105)

[3.1 Overview 4](#_Toc499471106)

[3.1.1 Most Frequent User 5](#_Toc499471107)

[3.1.2 Comments 6](#_Toc499471108)

[3.1.3 Busy Periods 7](#_Toc499471109)

[4. Conclusion 8](#_Toc499471110)

[Appendix A 9](#_Toc499471111)

# Overview

## Outline

The assignment outline is as follows:

* Assignment 4 is based on transforming a large dataset in text format - over 5000 lines of text.
* You will need to scrub (clean) the data and place it into the relevant holder/container objects.
* Once in these objects you will see that there are 422 different sets of commit objects.
* So your task will be to analyse these 422 objects that are in a list and come up with 3 interesting statistical pieces of information for this dataset with supporting evidence of "interestingness'
* You code for calculating the analysis should be documented and tested.
* Test should be in a separate file runnable from the command line.
* Your statistical analytics conclusions should be in a word document explaining in approximately 500 words the information that you have gleamed from the dataset.
* You will be required to submit your code via GitHub along with all documentation and tests.
* The deadline is the 26th November 2017 on Moodle @ 23:55.

# Process

## Input File

The data we had to analyse had been supplied in .html format. The first thing I did was to save in into a text file so I could work with it. The Input File contains a list of commit actions that were taken against a system, including details of what the change was, who made it and when etc.

All my files are available in Appendix A.

## Python Processing

Next I built a python file to read in the input file. This identifies the separator between blocks of data in the file and then picks specific values within the block to read in (each block of data is structured the same way). I didn’t read in all of the variables, just the following ones which I thought would be most useful:

1. Reference
2. Author
3. Date
4. Timestamp
5. Lines\_Updated
6. Author\_Comment

I also created a Python file to run some simple tests on the data. These tests included:

* Checking that there were 422 objects.
* Checking that there were over 5,000 lines in the file.
* Seeing what user made different commit statements.

## Output

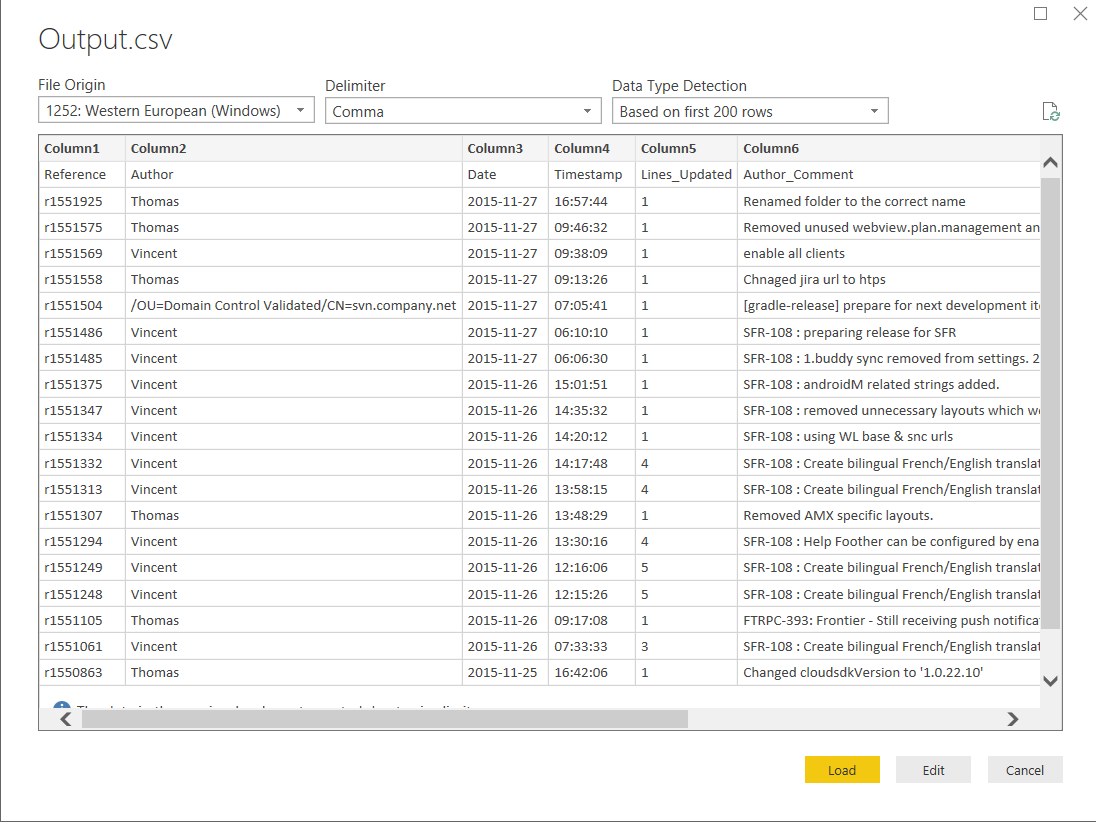
The Python files generated a .csv which included, row by row, each of the 6 variables I had pulled back on the 422 commit objects in the Input File.

This .csv can be pulled into a range of Data Analysis tools, so I was now ready to start analysing the output.

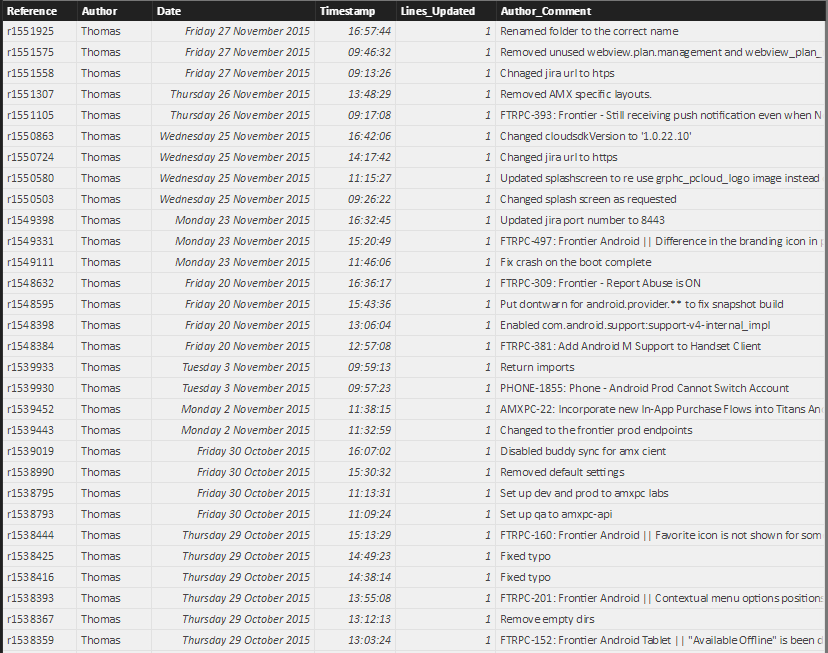
# Analysis

## Overview

I’ve decided to analyse the data in Power BI, which is an opportunity to learn how to use this tool. Importing it I can see that there is an entry in the Author Name column which looks like an update being made by the system and not a user, so I’ll rename that ‘System’ so it’s easier to report on.



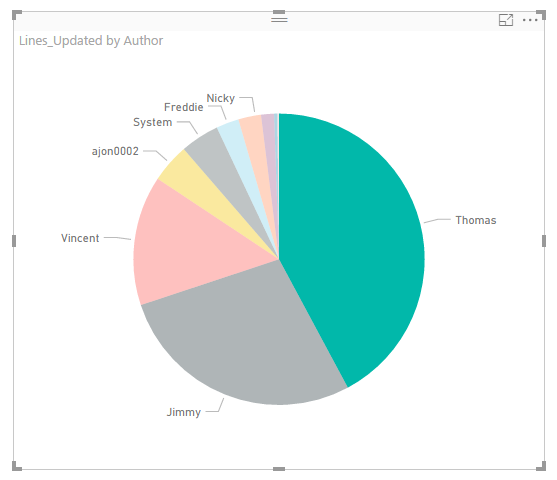
Here’s my data then in Power BI after I’ve imported it and replaced that system comment with a user called ‘System’.



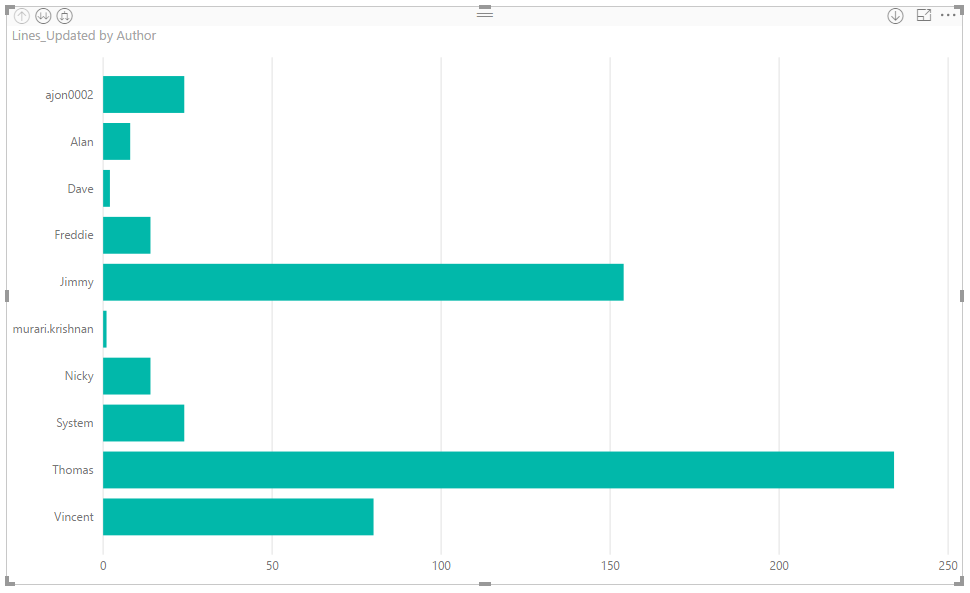
I’m now going to run some visualisations on the data and report on what I find most interesting.

## Most Frequent User

The author with the greatest number of comments in the time period is Thomas, with 234 comments. Murari Krishnan only made one comment in the time period (the comment field is not blank, so these are all the updates to the system). From the pie chart you can see that Thomas has made nearly half of all commits in this time period.

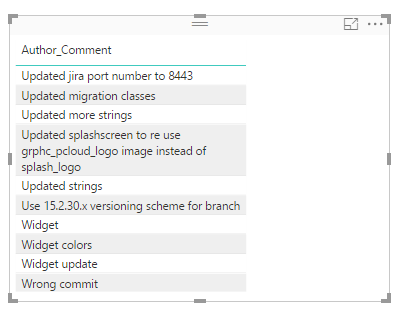


Here’s a bar chart containing the same information:



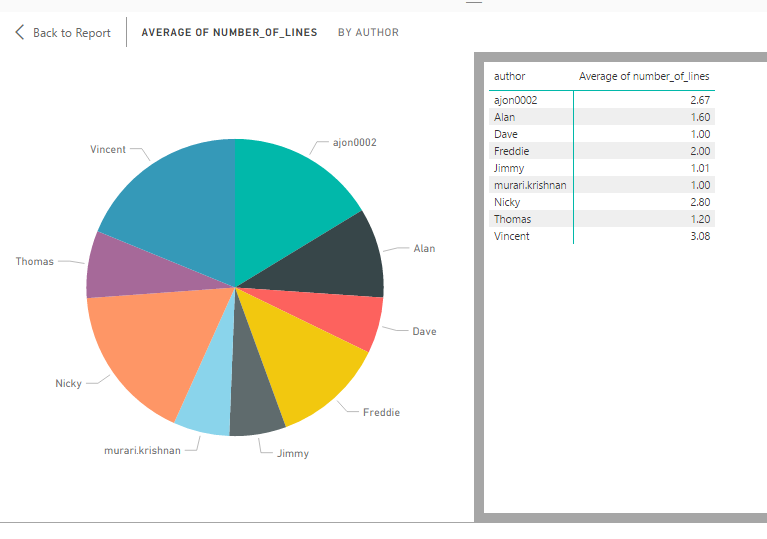
## Comments

Here’s the top 10 comments. The ‘Comment’ field in the system is obviously a free format text field, but the same items are added over time.



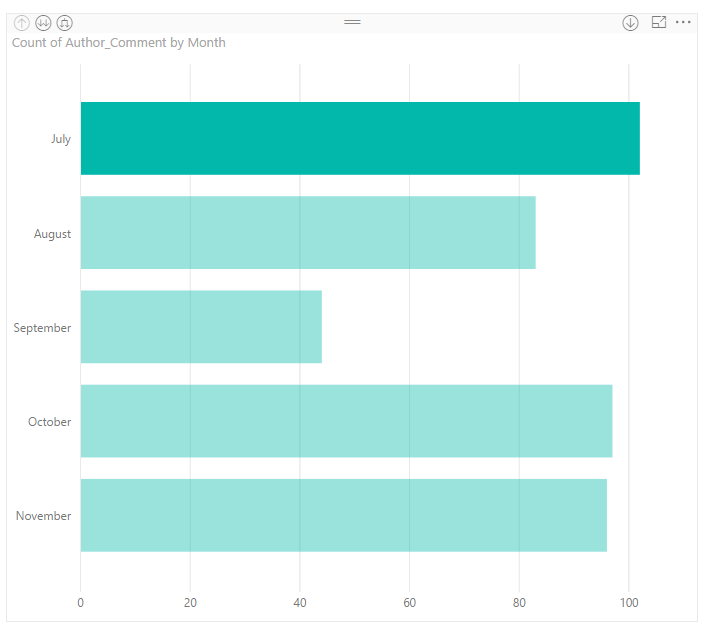
If you look at the two busiest users, Thomas and Jimmy, you can see that they make the same updates time after time. Thomas has the following comment 24 times *“[gradle-release] prepare for next development iteration”* whereas Jimmy made the following comment 10 times “*Lint Fix*”.

When we look at the average number of lines being updated during each commit we see that Vincent actually updates more lines at 3.08 than Thomas does at 1.20.

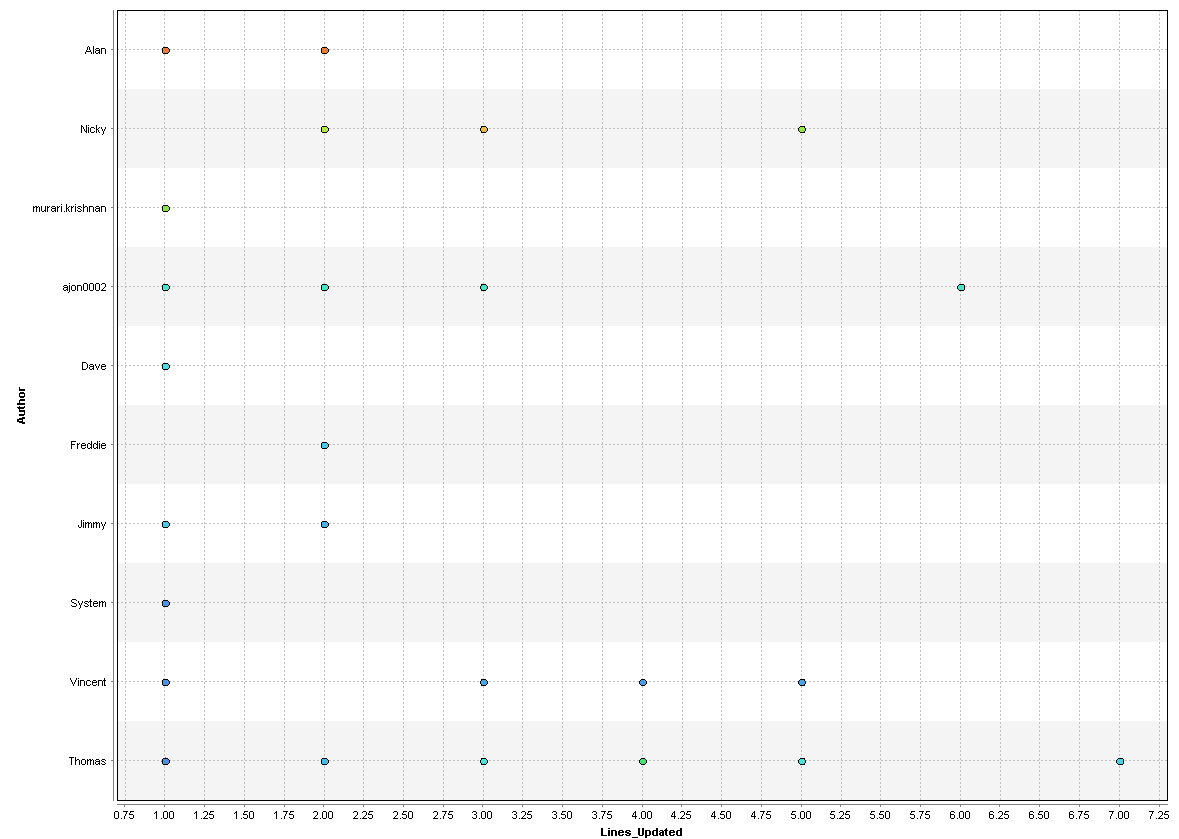


## Busy Periods

This graph shows when most of the commits were made to the database. We can see we are only looking at a 5 month period here with the most commits being made in July (102) and the least in September (44).



I had a quick look at the data in RapidMiner and could easily see what time of day the users were most likely to be updating the system. Alan works in the morning, system updates are made at 1am and Thomas is mostly likely to work throughout the day (he has the greatest spread of updates).



# Conclusion

This is a simple data set showing details of changes made to a system by various users over a set time period. I cleaned up the initial file and identified the key information I wanted to retrieve from it using Python and then did some high level analysis of the data using Power BI and RapidMiner. As required by the outline I’ve identified 3 interesting pieces of information about the data and have uploaded all my files online. They’re available at: <https://github.com/quingle/CA4.git>

# Appendix A

**Input File**



**Python Files**

*File to analyse data:*



*File to test data:*



**Output File:**

****