TERM 1

TERM 2

1. TERM 1 RECAP– PROGRAMMING & STATS

* Recap of Python fundamentals – this was no problem.
* Recap of statistics – this was more time consuming as it is a weaker area for me.

**Probability**

* Number successes/total possibilities

**Conditional probability**

* Bayes Rule e.g. if I choose A, what is the probability it came from bag B?
* Probability of A given B – full details of Bayes Rule formula in notebook.

**Descriptive Statistics – measures of spread**

* Mean: Count/total
* Median: Middle value of sorted list
* Mode: Most frequently occurring
* Std: Variance from the mean
* Q1: Median of all values below the mean
* Q3: Median of all values above the mean
* IQR: Q3-Q1

**T-Test – statistical test to compare 2 means**

* We use the T test to look for significant differences in data – it is easy to see that there are differences but you can’t easily see if these are due to outliers in the data without the t test.
* T test gives us a critical value: the P value.
* We want to know if the result is due to random chance or not.
* If the P value is <0.05, there is <5% chance that this is random, so >=95% confidence that this is significant.
* If the P value was 0.52, it means in 52% of cases, it could be due to chance which is not good enough!
* T Tests are either
* 1 tailed (= or ≠) or 2 tailed (>= or <=). We usually use 2 tailed.
* Paired or non paired (paired if same population at different times)

**T Test by hand (very time consuming)**

* Doing the calculations (e.g. means, size, stds)
* Set up hypotheses
  + H0 - µ1- µ2 = 0 (no difference btw sample means)
  + H1 - µ1- µ2 ≠0 (2 tailed test. 1 tailed would test < or > 0. Usually best to do 2 tailed)
* Calculate the T Stat: Gives us the P value e.g. if >0.05 we would not reject H0.

**T Test in Excel**

* For each sample, calculate AVERAGE(range), COUNT(range), STDEV(range)
* P Value = TTEST(range1, range2, #tails, type)
* Type is paired(1), equal variance (2), unequal variance (3)
* T Test in Excel

**T Test in Python**

From statsmodels.stats.weightstats import ttest\_ind

Ttest = ttest\_ind(array1, array2, alternative = ‘two sided’, usevar=’pooled’)

Or

From scipy.stats import ttest\_ind

From scipy.stats import ttest\_rel (if paired samples)

Ttest = ttest\_ind(1,2,equal\_var = True) or Ttest\_paired = ttest\_rel(1,2)

This will give us the t-stat, p value and variance.

PROJECT

The project concerned taking 2 samples of people who had taken the Stroop test, and looking to see if the difference in the samples was significant or due to chance.

It was carried out in Python.

E:\My Documents\Paula\DATA ANALYST NANO DEGREE\TERM 2\Stroop

* Test a Perceptual Phenomenon-Paula Thur.html

REVIEW FEEDBACK/REFLECTION

Ttest assumptions.hrml and Stats project feedback.doc give extra help on understanding what the ttest is about and what we can deduce from it.

1. EXPLORATORY DATA ANALYSIS – R STUDIO

* EDA is often the 1st stage of data analysis
* Quick understanding of data through visualisations and statistical models.
  + Variable distributions – histograms.
  + Relationship between variables – scatter plots.
  + Understand the source of data.
* Be curious and sceptical. Decide what insights/questions you want to examine. Look for oddities.
* R is used for statistical programming and visualisations, exploration. R Studio is easy to install – from CRAN!

LEARNING – a process for analysing data with R

1. Read in the file: setwd(‘…’)/getwd()/mydata = read.csv(‘file.csv’)
2. Check the class: class(dataset)
3. View dimensions dim(dataset)
4. Look at the columns in the dataset – names(dataset)
5. Dig deeper and look at structure - str(dataset)
6. Load dplyr – allows use of select, filter, arrange, rename, mutate, summarize)
   1. Note install.packages("dplyr”) at command line.
7. Get stats on relevant variables - summary(data)
8. View the data – head/tail(dataset)
9. Visualise the data
   1. Hist(data$var)
   2. Qplot – histograms/scatter plots/line plots/frequency polygons
   3. Ggplot
10. View plots side by side – library(GridExtra)/grid.arrange(p1,p2,ncol)
11. Alter/restructure data with tidyr – make long (gather) or wide (spread)
12. Look at correlations cor\_test(x,y)

Example scatter plot which focuses on 13-90 year olds and adds a mean line:

ggplot(aes(x=age,y=friend\_count),data = pf) +

xlim(13,90) +

geom\_point(alpha=0.05,position=position\_jitter(h=0),color=’orange’) +

coord\_trans(y=’sqrt’) +

geom\_line(stat=’summary’,fun\_y=mean)

PROJECT

Apply EDA techniques using R to explore a selected data set (White Wine) for distributions, outliers, and anomalies.

RMD file contains analysis, final plots, summary and reflection.

HTML file knitted from RMD file (knitr)

E:/My Documents/Paula/DATA ANALYST NANO DEGREE/TERM 2/EDA/EDA project/

* whitewineEDA2.html
* wineQualityWhites.csv
* whitewineEDA2.rmd
* EDA project review.docx

REVIEW FEEDBACK/REFLECTION

* Easy to produce analysis on a dataset with R Studio.
* Might not be so easy to do a lot of data wrangling – might need to be cleansed in Python.
* Visualisations can be produced quickly and easily.
* Can produce presentations easily by knitting to html.
* The RMD file can be structured with code comment blocks so the file is a stream of consciousness. Also good practice to give each code block a label.

1. DATA WRANGLING – PYTHON

This generally must be done before analysis, visualising and building models. The process:

* Gather data
  + Download files from the internet programmatically using the requests and os libraries; requests.get(url) and read files into a dataframe.
  + Use APIs to scrape twitter, Instagram, wikipedia etc. Each API will have its own Python libraries, e.g. wptools for Wikipedia, tweepy for twitter. Most data from APIs is in JSON format which can be read into JSON objects which are like Dictionaries.
  + Use the Beautiful Soup library if there is no machine readable API.
  + There are libraries to allow you to connect to a database to get data.
* Assess data (ensure you follow formulaic process to make this simpler)
  + Quality (missing values and inconsistency = dirty data)
  + Tidiness (each variable in a column, each obs in a row, each obs unit in a table)
  + First assess visually – view in spreadsheet or table – and note the issues.
  + Then assess programmatically – types, counts, nulls, values, duplicates – for tidiness rules.
* Clean data
  + Programmatically – define in words, code and test each issue found.
  + Ensure you copy the dataframe before doing any cleansing.

PROJECT

Wrangle WeRateDogs Twitter data to create interesting analyses and visualizations using python. The project supports the three stages of the data wrangling process and

the work culminated with storing, analysing and visualising the cleansed data.

E:/My Documents/Paula/DATA ANALYST NANO DEGREE/TERM 2/ DataWrangling/Project

* wrangle\_report.pdf
* act\_report.pdf – the actual code and comments.
* Wrangling review.doc

REVIEW FEEDBACK/REFLECTION

This felt like it should have been a fairly simple and quick project but it was very intensive and the cleansing could have gone on for much longer. I think you need to be clear about how much cleaning is really necessary as there would always be more to do.

There are more succinct ways to cleanse – better use of functions and better use of Jupyter functionality.

Also important to note that actually gathering the data can be a big, time-consuming job!

1. DATA STORYTELLING – TABLEAU

This was the Explanatory phase of the data analysis process. It concerned presenting insights in a focused way, using the best visualisation techniques to engage the audience, using polished visualisations.

Visualisations should

* Have high data ink ratio – there should not be superfluous information that is not part of the main message and remove chart junk.
* Use simple colours – perhaps just one colour that is colour blind friendly (blue/orange) or just black. Only use a different colour if you want to make something stand out.
* Not necessary to use 3D visuals.

When presenting findings start with a question, then deeper questions. Use visuals to help.

Tableau allows you to first create visuals on a work sheet, drag worksheets to a dashboard, and add dashboards in order to create a datastory.

Visuals can be interactive – use hovers and interactive filters.

PROJECT

The task was to use Tableau to create an explanatory data visualisation from the Baseball dataset to communicate which factors influenced the performance of batters.

I firstly quickly visualised the variables (input and response variables) then looked at the response variables more closely and ended with a conclusion.

A big part of this project was sharing with others and getting feedback then providing new iterations of the presentation.

E:\My Documents\Paula\DATA ANALYST NANO DEGREE\TERM 2\Visualisation\Tableau\Project

* Data\_Story\_Writeup.pdf

REVIEW FEEDBACK/REFLECTION

* Tableau Project review.doc

I really liked Tableau as a tool and couldn’t believe how simple it was to produce something really effective for an audience.

I think Tableau would be used to create a final interactive product for users once preliminary data analysis had been performed – e.g. EDA, wrangling, etc.