Computing for mathematics handout 8 - Sharing and publishing worksheets, writing to files, the individual coursework and Class test feedback.

Lecturer: Vince Knight

Office: M1.30

email: knightva@cf.ac.uk

Office hours: Thursday 1300-1500

What you have learnt this week:

- You learnt about the mathematical topic of linear algebra;

- You saw how to make relevant calculation in Sage:

    - Determinant
    - Inverse

Sharing files

Sage makes it very easy to share worksheets. If you know someones username, you can click on 'Share' (towards the top right) and add lists of usernames of people who want to give access to your worksheet.

You can also Publish worksheets. This is how I get the solutions to you. Clicking on 'Publish' puts your worksheet on the internet (viewable by everyone). On a published worksheet (like the solutions) you can click on 'Edit this.' (top left) to add a copy of the sheet to your own Sage account.

Writing to file

One difficulty that was reported to me was writing data to file. Let's take a quick look at the following example (revisiting the field example from Handout 5):

Let us say that we want to write to file the dimensions of the profitable fields but also this value (let's pretend it's important):

$$\int_0^h x^w \, dx$$

Where h and w are the heights and widths of the field.

```
fields = [[4,5],
          [6,2],
          [1,7],
          [8,2],
          [4,1],
          [7,2],
          [8,2],
          [9,1],
          [10,56],
          [83,15],
          [4,1],
          [53,2]]

class Field():
    """
    A class for our field
```

```
    """
    def __init__(self, x, y):
        self.width = x
        self.height = y

    def profitable(self):
        return self.width * self.height >= 50

    def measure(self):
        return integrate(x ^ self.width, x, 0, self.height)  # This is a new measure that we cr

fields = [Field(f[0], f[1]) for f in fields]

lstforoutput = []
for f in fields:
    if f.profitable():
        lstforoutput.append([f.height, f.width, f.measure()])
```

Now let's write our data to file (we saw how to do this in the second sheet).

```
import csv
f = open("fieldoutput.csv", "w")
csvwrtr = csv.writer(f)

for row in lstforoutput:
    csvwrtr.writerow(row)

f.close()
```

The individual coursework

The individual coursework is now available (**due in week 11**): http://drvinceknight.github.io/Computing_for_mathematics/Individ

It requires you to use LaTeX (how mathematicians write mathematics). Take a look at the 9th Lab sheet if you want to get started (in the meantime you can get going with the required coding).

Class test

Here are some summary statistics about the class test marks:

- Total mark:

    - mean 56.23
    - median: 55.00
    - max: 100.00

- q1 mark:

    - mean: 90.09
    - median: 100.00
    - max: 100.00

- q2 mark:

    - mean: 42.79
    - median: 36.67
    - max: 100.00

- q3 mark:

  - mean: 24.81
  - median: 16.67
  - max: 100.00

- Documentation mark:

  - mean: 89.23
  - median: 100.00
  - max: 100.00

Percentage of marks above given mark:

- 0.85 of scripts $\geq 40\%$
- 0.71 of scripts $\geq 50\%$
- 0.38 of scripts $\geq 60\%$
- 0.22 of scripts $\geq 70\%$
- 0.05 of scripts $\geq 80\%$
- 0.01 of scripts $\geq 90\%$

A picture of the distribution of the marks:

What you should do next:

- **Start the next sheet**: make sure you spend time working on the sheet **BEFORE** the labs.
- **Start the coursework**
- Contribute to the wiki.
- To make the best use of the lab sessions turn up having finished your sheets;
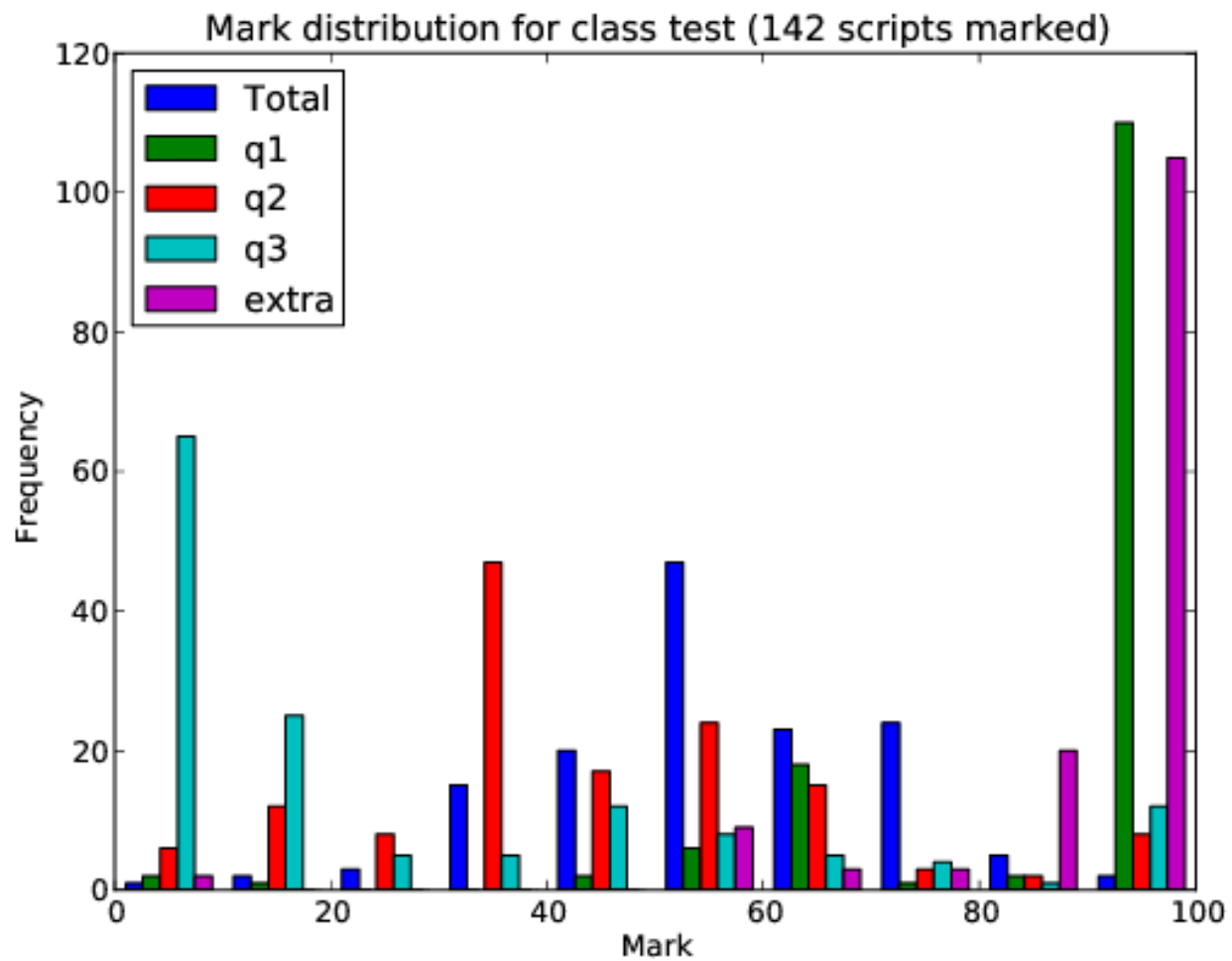- If anything is still unclear **please** come and see me during office hours.

Figure 1: