Python for Science and Engg: Statistics

FOSSEE

Department of Aerospace Engineering
IIT Bombay

25 September, 2010
Day 1, Session 3

Outline

1. Computing mean
2. Processing voluminous data
   - Data processing
   - Dictionaries
   - Visualizing data
   - Obtaining statistics

Value of acceleration due to gravity?

- We already have pendulum.txt
- We know that $T = 2\pi\sqrt{\frac{L}{g}}$
- So $g = \frac{4\pi^2 L}{T^2}$
- Calculate “g” - acceleration due to gravity for each pair of L and T
- Hence calculate mean “g”

Acceleration due to gravity - “g”. . .

```
In []: g_list = []
In []: for line in open('pendulum.txt'):
  ....     point = line.split()
  ....     L = float(point[0])
  ....     t = float(point[1])
  ....     g = 4 * pi * pi * L / (t * t)
  ....     g_list.append(g)
```

Mean “g” - Classical method

```
In []: total = 0
In []: for g in g_list:
 ....:     total += g
 ....:

In []: g_mean = total / len(g_list)
In []: print 'Mean: ', g_mean
```

Mean “g” - Slightly improved method

```
In []: g_mean = sum(g_list) / len(g_list)
In []: print 'Mean: ', g_mean
```

Mean “g” - One liner

```
In []: g_mean = mean(g_list)
In []: print 'Mean: ', g_mean
```

10 m

Outline

1. Computing mean

2. Processing voluminous data
   - Data processing
   - Dictionaries
   - Visualizing data
   - Obtaining statistics

More on data processing

We have a huge data file–180,000 records.
How do we do *efficient* statistical computations, i.e. find mean, median, standard deviation etc; draw pie charts?

Structure of the file

Understanding the structure of sslc1.txt

- Each line in the file has a student’s details(record)
- Each record consists of fields separated by ’;’

```
A;015162;JENIL T P;081;060;77;41;74;333;P;;
```

Structure of the file . . .

```
A;015163;JOSEPH RAJ S;083;042;47;AA;72;244;;;
```

Each record consists of:

- Region Code
- Roll Number
- Name
- Marks of 5 subjects: SLang, Flang Maths, Science, Social
- Total marks
- Pass/Fail (P/F)
- Withheld (W)

15 m

Statistical Analysis: Problem statement

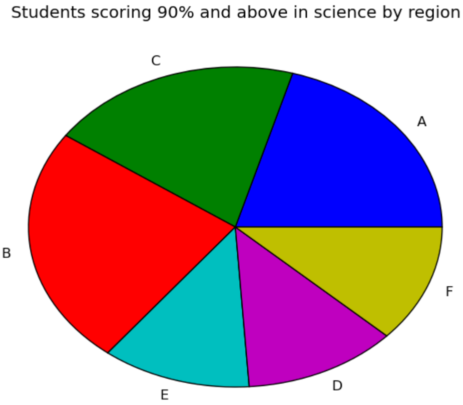
1. Read the data supplied in the file *sslc1.txt* and carry out the following:

   a Draw a pie chart representing proportion of students who scored more than 90% in each region in Science.

   b Print mean, median and standard deviation of math scores for all regions combined.

Problem statement: explanation

a. Draw a pie chart representing proportion of students who scored more than 90% in each region in Science.



Machinery Required

- File reading
- Parsing
- Dictionaries
- Arrays
- Statistical operations

Outline

1 Computing mean

2 Processing voluminous data
- Data processing
- Dictionaries
- Visualizing data
- Obtaining statistics

File reading and parsing . . .

Reading files line by line is the same as we had done with the pendulum example.

```
for record in open('sslc1.txt'):
    fields = record.split(';')
```

Outline

1. Computing mean
2. Processing voluminous data
   - Data processing
   - Dictionaries
   - Visualizing data
   - Obtaining statistics

Dictionaries: Introduction

- Lists index using integers
  Recall `p = [2, 3, 5, 7]` and
  `p[1]` is equal to `3`
- Dictionaries index using strings

Dictionaries . . .

```
In []: d = {'png'  : 'image file',
      'txt'  : 'text file',
      'py'  : 'python code',
      'java': 'bad code',
      'cpp': 'complex code'}

In []: d['txt']
Out[]: 'text file'
```

Dictionaries . . .

```
In []: 'py' in d
Out[]: True

In []: 'jpg' in d
Out[]: False
```

Dictionaries . . .

```
In []: d.keys()
Out[]: ['cpp', 'py', 'txt', 'java', 'png']

In []: d.values()
Out[]: ['complex code', 'python code',
        'text file', 'bad code',
        'image file']
```

25 m

Inserting elements into dictionary

d[key] = value

```
In []: d['bin'] = 'binary file'
In []: d
Out[]:
{'bin': 'binary file',
 'cpp': 'complex code',
 'java': 'bad code',
 'png': 'image file',
 'py': 'python code',
 'txt': 'text file'}
```

Getting back to the problem

Let our dictionary be:

```
science = {}
```

- Keys will be region codes
- Values will be the number students who scored more than 90% in that region in Science

Sample *science* dictionary

{'A': 729, 'C': 764, 'B': 1120,'E': 414, 'D': 603, 'F': 500}

Building parsed data . . .

```
science = {}

for record in open('sslc1.txt'):
    fields = record.split(';')

    region_code = fields[0].strip()
```

Building parsed data . . .

```
if region_code not in science:
    science[region_code] = 0

score_str = fields[6].strip()

score = int(score_str) if \
    score_str != 'AA' else 0

if score > 90:
    science[region_code] += 1
```

Building parsed data . . .

```
print science
print science.keys()
print science.values()
```

Outline

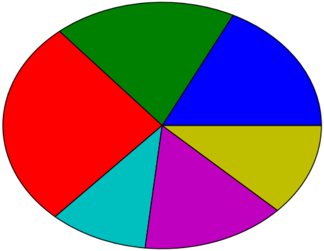
1 Computing mean

2 Processing voluminous data

- Data processing
- Dictionaries
- Visualizing data
- Obtaining statistics

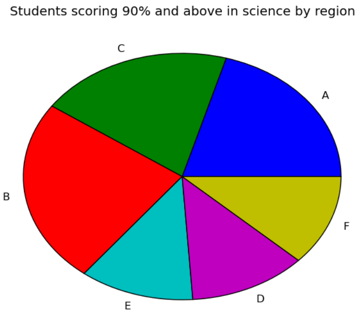
Pie Chart

```
pie(science.values())
```



Pie chart

```
pie(science.values(),
    labels = science.keys())
title('Students scoring 90% and above
      in science by region')
savefig('science.png')
```



Problem statement

b. Print mean, median and standard deviation of math scores for all regions combined.

Building data for statistics

```
math_scores = []

for record in open('sslc1.txt'):
    fields = record.split(';')

    score_str = fields[5].strip()
    score = int(score_str) if \
      score_str != 'AA' else 0

    math_scores.append(score)
```

Outline

1. Computing mean
2. Processing voluminous data
   - Data processing
   - Dictionaries
   - Visualizing data
   - Obtaining statistics

Obtaining statistics

```
print 'Mean: ', mean(math_scores)

print 'Median: ', median(math_scores)

print 'Standard Deviation: ',
                std(math_scores)
```

45 m

Obtaining statistics: efficiently!

```
math_array = array(math_scores)

print 'Mean: ', mean(math_array)

print 'Median: ', median(math_array)

print 'Standard Deviation: ',
                  std(math_array)
```

50 m

What tools did we use?

- Dictionaries for storing data
- Facilities for drawing pie charts
- Efficient array manipulations
- Functions for statistical computations - mean, median, standard deviation