UNIVERSITÄT DES SAARLANDES

ZBI ZENTRUM FÜR BIOINFORMATIK

Programming Project

Programming with Python (for Bioinformatics)

Johanna Schmitz & Sven Rahmann

05.06.2024; Submit part I before **18.06.2024, 23:59**

Introduction

To pass the course, you have to successfully complete a multi-part programming project.

Overall goal

The goal is to analyze data from an atomic force microscope:
One examines a flat surface with "blobs" of some unknown material on it.
A thin needle is pushed down onto the surface until it hits the surface or the blob.
The surface or the material then pushes back, which results in a measurable force.
This is repeated on many locations (i, j) on the surface, $i = 0, \ldots, 127$, $j = 0, \ldots, 127$.

Data

The full data is available as a huge 1.8 GB **text** file.
It contains many (32768) measurements (two at each coordinate).
Each measurement consists of roughly 700 data points (distance d vs force f).
Initially, we provide a small sample of the file (12 measurements only).
The data file contains the numeric data (in textual form),
but also information about the measurement device and conditions.

Part I

Goal

The first goal is to extract the distance and force data from the text file for each point (i, j) on the surface (in two distinct measurement series).
The second goal is to plot each measurement as a simple graph to examine the data.

Framework

We provide a stub for a small command line application:
`python plotafm.py --textfile sample.txt --show --plotprefix curve`
Running this should create 12 PNG images whose filenames start with `curve`, e.g. `curve-0-004-000.png` is the measurement in series $s = 0$, at $i = 4$ and $j = 0$.
It will also show every plot on the screen (omit `--show` to run without interaction).
Unfortunately, we do not provide `plotafm.py`, but only an incomplete stub `plotafm-partial.py`.

Part I

Task

Finish the implementation and rename your solution to `plotafm.py`.
Commit and push the two files `sample.txt` and `plotafm.py` to your repository in the directory `project-afm/`.

How to proceed

- Examine the provided code. Note the command line interface (CLI) and its options.
- Note the two places in the program where it says `TODO`. Here, you have to work.
- Examine the provided `sample.txt` file:
  You need to make sense of the data with incomplete knowledge;
  this is very often the case in practice! Some observations are on the next slide.
- Please put some effort into generating a nice plot with axis labels, etc.
- Distance d in m should be on the x-axis; force in N on the y-axis.

Part I: Observations on the data file

- Most lines start with #;
  they contain information about the measurement, but not the measurement itself.
- The first few lines are a header.
- A new data point starts with 3 lines containing `index`, `iIndex`, `jIndex`.
  This is how you get the i and j coordinate of the point.
- Note that every (i, j) cooridnate appears **twice** in the data!
  The first apperance is for data series $s = 0$, the second one for series $s = 1$.
- The actual measurement data for point (i, j) begins below the line `# units`.
- Each measurement consists of 7 values, only the first two are of importance:
  - distance in m
  - force in N
- The number of measurement lines is given in the line `# recorded-num-points`.
- All (≈ 700) distance values should go into a numpy array d, force values into f.
- After the numerical data, a new data point starts (new `index`, `iIndex`, `jIndex`).

Part I: Example plot: `curve-1-005-000.png`