

ESTRATTA

Prova teorico-pratica B

Parte 1. Teorica

La/Il candidata/o sempre all'interno di questo Notebook, risponda alle seguenti quattro domande relative ai quattro argomenti di cui all'Art.7 comma 1 del Bando:

- 1) Tecniche di analisi dati e, in particolare, elaborazione di immagini
- 2) Progettazione e sviluppo di software scientifico in Python. ¶
- 3) Gestione di codici in ambienti condivisi (es. Git, GitHub, GitLab).
- 4) Revisione e definizione dei requisiti software in contesti collaborativi internazionali.

Theoretical-Practical Test B

Part 1. Theoretical Section

The candidate, within this Notebook, should answer the following four questions related to the four topics listed in Article 7, Paragraph 1 of the Call for Applications:

- 1) Data analysis techniques, and in particular, image processing.
- 2) Design and development of scientific software in Python.
- 3) Management of code in shared environments (e.g., Git, GitHub, GitLab).
- 4) Review and definition of software requirements in international collaborative contexts.

Domanda 1)

Si supponga di avere un numero N di immagini bidimensionali di uno stesso campo, ma non esattamente coincidenti e di dimensione non uguale. Si assuma anche che la dimensione dei pixel non sia uguale per le varie immagini. Definire la struttura di una pipeline in Python che:

- Definisce la parte in comune alle N immagini
- Faccia la media delle varie immagini nella parte in comune.

Question 1)

Suppose you have N two-dimensional images of the same field, but they are not perfectly aligned and do not have the same dimensions. Assume also that the pixel size is different among the various images.

Define the structure of a Python pipeline that:

determines the region common to all N images; computes the average of the images over the common region.

Risposta 1)

Domanda 2)

Si descriva come strutturare un software Python destinato all'elaborazione e visualizzazione di dati sperimentali. In particolare, discutere la suddivisione del progetto in componenti riutilizzabili, la gestione degli input e degli output, l'impiego di ambienti virtuali e le procedure di testing e validazione del codice.

Question 2)

Describe how to structure a Python software application intended for the processing and visualization of experimental data. In particular, discuss the division of the project into reusable components, the management of input and output operations, the use of virtual environments, and the procedures for testing and validating the code.

Risposta 2)

Domanda 3)

Si descriva come organizzare il lavoro di sviluppo in un progetto software gestito con Git, nel caso in cui più persone contribuiscano contemporaneamente al codice. In particolare, discutere l'uso di rami separati per lo sviluppo di nuove funzionalità, le modalità di aggiornamento e integrazione con il ramo principale, e le strategie per individuare e risolvere eventuali conflitti di merge.

Question 3)

Describe how to organize development work in a software project managed with Git when multiple people contribute to the code simultaneously. In particular, discuss the use of separate branches for the development of new features, the procedures for updating and integrating changes with the main branch, and the strategies for identifying and resolving potential merge conflicts.

Risposta 3)

Domanda 4)

In un progetto software sviluppato in collaborazione, le specifiche iniziali possono cambiare durante il ciclo di vita del progetto. Si descriva come gestire in modo ordinato i requisiti software, dalla loro raccolta e documentazione fino al tracciamento delle modifiche. Discutere inoltre come mantenere la coerenza tra requisiti, codice sviluppato e documentazione, facendo riferimento a strumenti quali issue tracker, sistemi di versionamento e procedure di code review.

Question 4)

In a collaboratively developed software project, the initial specifications may change during the project lifecycle. Describe how to manage software requirements in an organized manner, from their collection and documentation to the tracking of subsequent changes. Furthermore, discuss how to maintain consistency between requirements, the developed code, and the documentation, making reference to tools such as issue trackers, version control systems, and code review procedures.

Risposta 4)

Parte 2. Pratica

La/Il candidata/o sviluppi una routine in linguaggio Python in ambiente Jupyter Notebook, per l'analisi di un'immagine astronomica fornita dalla Commissione in formato FITS, denominata: `image.fits`

La routine dovrà essere sviluppata utilizzando esclusivamente librerie Python standard per l'analisi scientifica dei dati, quali `numpy`, `matplotlib` (dopo avere importato `matplotlib` si utilizzi il comando `%matplotlib inline` per la corretta visualizzazione delle immagini), `astropy` e `photutils`. Durante lo svolgimento della prova è consentito l'utilizzo delle funzionalità di documentazione integrate nell'ambiente di programmazione Python (`docstring`), accessibili localmente e senza necessità di connessione alla rete. In particolare, i candidati potranno consultare la documentazione delle funzioni e delle librerie disponibili mediante i comandi standard dell'ambiente Jupyter Notebook o del linguaggio Python (per esempio `help(plt.imshow)`)

La/Il candidata/o dovrà:

- 1) leggere il file `image.fits` e visualizzarne le informazioni principali;
- 2) produrre una visualizzazione dell'immagine con una scala di contrasto adeguata;
- 3) Estrarre dall'header del file fits alcuni parametri, quali `CRVAL1` (coordinate del centro in x) e `CRVAL2` (coordinate del centro in y) e delle dimensioni del pixel in x e y (`CD1_1` e `CD2_2`) e calcolare le dimensioni dell'immagine in x e y in gradi. Scrivere questi dati in un file testo di output (`output.txt`);
- 4) Descrivere le caratteristiche dell'immagine mediante analisi dell'istogramma dei flussi (a tale scopo, non considerare eventuali pixel anomali); definire l'RMS ed il flusso medio nell'immagine. Scrivere questi dati nel file testo di output precedente (`output.txt`);
- 5) Analizzare la parte dell'istogramma a bassi flussi, trovare il massimo dell'istogramma e definire l'RMS ed il fondo medio. Scrivere questi dati nel file testo di output precedente (`output.txt`);
- 6) Determinare la percentuale di pixel dell'immagine che hanno un flusso maggiore di 5 sigma più della media. Scrivere questo dato nel file testo di output precedente (`output.txt`);
- 7) Identificare alcune sorgenti con alto S/N, ma non saturate, e definire il profilo di intensità in x e in y. Definire la larghezza a metà altezza del profilo (in pixel e, successivamente, in secondi d'arco. Riferirsi all'Header del file fits). Effettuare un fit gaussiano bi-dimensionale di una sorgente con buon S/N e determinare la posizione in RA e DEC con i relativi errori;

8) (da elaborare in alternativa o ad integrazione del punto precedente): Selezionare una sotto immagine di dimensione 1000x1000 pixel, trovare le sorgenti (stelle) presenti in questo campo, con flusso superiore a 5 sigma, eliminare tutte queste sorgenti per ottenere un'immagine del fondo.

La valutazione terrà conto della correttezza dell'approccio, della chiarezza del codice, della capacità di utilizzare Python per l'analisi di immagini astronomiche, della gestione del fondo cielo, della qualità della documentazione e della consapevolezza delle limitazioni del metodo implementato.

Part 2. Practical Section

The candidate shall develop a Python routine in a Jupyter Notebook environment for the analysis of an astronomical image provided by the Examination Committee in FITS format, named `image.fits`.

The routine must be developed using only standard Python libraries for scientific data analysis, such as NumPy, Matplotlib (after importing Matplotlib, use the command `%matplotlib inline` for proper image display), Astropy, and Photutils.

During the examination, candidates may use the documentation features integrated into the Python programming environment (docstrings), accessible locally and without requiring an internet connection. In particular, candidates may consult the documentation of available functions and libraries using the standard commands provided by Jupyter Notebook or the Python language (for example, `help(plt.imshow)`).

The candidate shall:

1. Read the file `image.fits` and display its main information.
2. Produce a visualization of the image using an appropriate contrast scale.
3. Extract from the FITS header several parameters, including **CRVAL1** (x-coordinate of the image center), **CRVAL2** (y-coordinate of the image center), and the pixel scale values in x and y (**CD1_1** and **CD2_2**). Compute the image size in the x and y directions in degrees. Write these quantities to a text output file (`output.txt`).
4. Describe the characteristics of the image through an analysis of the flux histogram (excluding any anomalous pixels). Determine the **RMS** and the **mean flux** of the image. Write these values to the previously defined output text file (`output.txt`).
5. Analyze the low-flux region of the histogram, find the histogram peak, and determine the **RMS** and the **mean background level**. Write these values to the same output text file (`output.txt`).
6. Determine the percentage of image pixels whose flux is greater than **5 σ above the mean value**. Write this result to the output text file (`output.txt`).
7. Identify several sources with high **signal-to-noise ratio (S/N)** that are not saturated, and determine their intensity profiles along the x and y directions. Measure the **full width at half maximum (FWHM)** of the profiles, first in pixels and then in arcseconds (using the information contained in the FITS header). Perform a **two-dimensional Gaussian fit** to a source with good S/N and determine its **Right Ascension (RA)** and **Declination (Dec)**, together with the associated uncertainties.
8. *(To be carried out either as an alternative to, or in addition to, the previous point):* Select a **1000 × 1000 pixel** sub-image, detect the sources (stars) present in this field with fluxes above **5 σ** , and remove all detected sources in order to obtain an image containing only the background.

The evaluation will take into account the correctness of the approach, the clarity of the code, the ability to use Python for astronomical image analysis, the handling of the sky background, the quality of the documentation, and the awareness of the limitations of the implemented method.

In []: 1