Recruitment task
ASTRO - BEAM

October 12, 2024

Overview

Your task is to carry out various data analysis and signal processing methods to demonstrate your abilities as a potential member of the ASTRO project. This exercise should also be used as a learning tool for python, jupyter notebooks, git, the basics of radio astronomy and machine learning. To make the most of this, ensure your code is readable and perform necessary revisions as you enhance your abilities.

In this task you will:

- Simulate radio frequency signals
- Perform smoothing algorithms and calibrations
- Detect characteristics of signals
- Upsample your signals
- Find suitable fits
- Evaluate your techniques
- Provide clear and meaningful visualizations along the way.

Once you complete these steps, you'll need to repeat the process using different methods of your choosing. Then, compare the results with your original approach. It does not matter which results are better.

Your work must be presented in a well-organized **Jupyter notebook**. Use markdown cells to summarize your code, add comments within the code to explain your decisions and plot graphs that build a cohesive image of your actions.

The python libraries that you decide to use are up to you but here are some recommendations:

- **numpy** for numerical analysis

- **pandas** for data analysis
- **matplotlib** for data visualization
- **scipy** for fitting.

Disclaimer: Every bold text is a hyperlink.

Lastly, upload your notebook to Github and update it as you progress. Git is an essential version control tool, especially when working in a team. Personally, I use Github Desktop and the VSCode IDE, but feel free to choose tools that suit you best.

1 Simulating a radio signal

In radio astronomy, an observation produces a signal that consists of a background, noise and potential spectral lines from radio sources. These observations also have a duration, which depends on the type of instrument and the brightness of the source. So, the signal is a function of Relative Power (Arbitary unit for the purpose of this task) over Frequency (MHz). During the observation we can assume that the background and spectral lines stay the same but the noise changes each time segment. In order to simulate our first signal follow these steps:

1. Decide how many samples to use for your signal
2. Define the region of the frequency domain for your signal. It must contain 1420 MHz and have a bandwidth of < 5 MHz.
3. Create a **background signal**. This signal can have any shape you want but avoid using a straight line for this task. This background will be reused on every signal.
4. Add **gaussian noise** to your background. The noise should be random for each time segment.
5. Repeat for a number of time segments you provide in order to create a **"waterfall".** This duration of time will be constant for every signal.
6. Visualise the waterfall as a **heatmap** where Relative Power is represented by color.
7. Change the signal accordingly until it looks real.
8. Remember to visualize each step with its corresponding titles, legends and axes

This signal will be called ”off” because it does not contain any spectral lines. We will use it to remove the background from future signals. Moving forward always create a new signal when applying each step instead of overwriting a signal you have labeled.

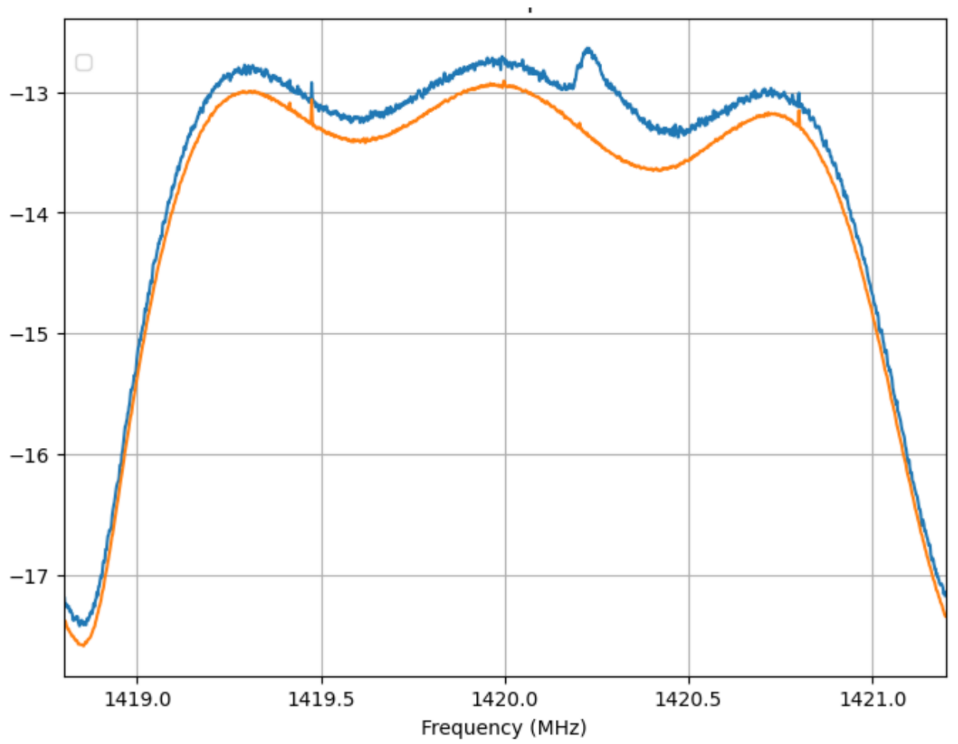
Next, create an ”on” signal by following the same steps. but this time introduce a spectral line at 1420 MHz. This is called the **hydrogen line or 21 centimeter line**. Simulate a signal containing only a **spectral line shape** at 1420 MHz for each time segment. Label it ”spectral” The spectral line should differentiate from the baseline (background + noise).

Finally, create the same spectral line shape but with twice the samples. Label it ”spectral2”. We will use it later to evaluate your upsampling.

2 Signal Smoothing

2.1 Averaging over time

Our first instance of signal smoothing is finding the average signal over time for both the ”off” and ”on” signals. You must be left with two two-dimensional signals:



Blue represents ”on” and orange represents ”off”. In your case the signals should be noisier and overlap with each other.

2.2 Removing Background

The "off" signal can be used to remove the background from the "on" signal. This is achieved by dividing $\frac{on}{off}$. This is an easy case of background calibration that we will call on/off calibration. Subtracting off from on (on-off) is another case of calibration that can be used. The final signal will be labeled "calibrated".

2.3 Moving average

To take it a step further, apply a **moving average** on the "calibrated" signal. Select a odd number N. N should be a small percentage of the total samples in the signal. Each point in the calibrated signal should be smoothed by averaging it with the $\frac{N-1}{2}$ values before and after it. Be mindful of edge cases, and ensure you don't reuse already averaged values when performing the moving average. Label this resulting signal as "smooth".

3 Spectral Line Detection

Identify the frequency where the spectral line peaks without assuming it corresponds to the highest power value. The peak should be near 1420 MHz.

4 Upsampling

Use the "smooth" signal and **upsample** it using **linear interpolation**. Label this signal "smooth2". Perform the following steps for both "smooth" and "smooth2"

5 Spectral Line Fitting

Find and calculate the best fit for the spectral line depending on how you decided to simulate it, if the data from the previous step is useful you can use it. This step should result in two signals labeled "fit" and "fit2"

6 Error Estimation

Compare the "fit" and "fit2" signals with "spectral" and "spectral2" accordingly by providing graphs and evaluation metrics.

7 Your turn

Now that you are comfortable with the fundamentals of working with radio astronomy data it is your turn to try more advanced methods. Follow all the above steps while making the following changes:

- Use a different noise type in step 1
- Use a different background calibration/subtraction method in step 2.2 without the use of the "off" signal
- Use a different smoothing algorithm in step 2.3
- Use a different interpolation in step 4
- Optional: Experiment either by changing your signals or by changing techniques you already decided to use.

Have fun!