Imaging ALMA data

Preparation + continuum imaging

Dan Walker

UK ALMA Workshop 2025
The University of Manchester

ALMA
EUROPEAN ARC
ALMA Regional Centre || UK

MANCHESTER
1824

How do we actually image ALMA data?

- With an interferometer, we observe an interference pattern that is expressed by the complex visibility $V(u,v)$

- Inverse Fourier transform of $V(u,v)$ → image in (x,y)

  - Image has complicated artefacts due to incomplete sampling of uv-plane. This is the so-called '**dirty image**' — a convolution of the sky brightness and the '**dirty beam**' (point spread function).

  - Solution is to deconvolve dirty beam from dirty image to recover the true sky brightness
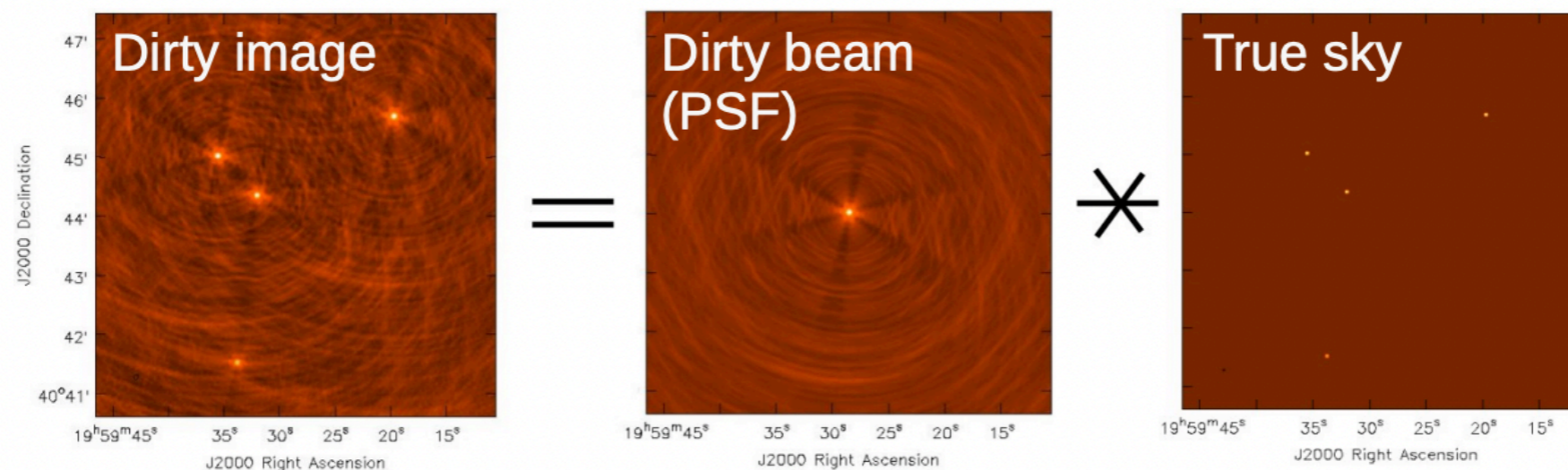
Deconvolution

- Dirty image is not ideal for science due to image artefacts

- Deconvolution aims to reconstruct the true sky brightness, but …

  - Missing information due to incomplete uv-coverage

  - Data is corrupted by noise

  - There is no unique solution

→ Aim is to find a good model of the sky brightness

Most widely used deconvolution method is the **CLEAN** algorithm (Hogbom 1974)

Basic CLEAN algorithm

Initialise a residual map (dirty map)

- Identify strongest feature in residual map as a point source

- Add point source to the clean model

- Convolve the point source with dirty beam and subtract from residual map

- If stopping criteria not reached, do next iteration

Convolve clean model with the 'clean beam' (usually a Gaussian estimated from the dirty beam) and add residual map to make the final image

Dirty image

Clean model

Cleaned image
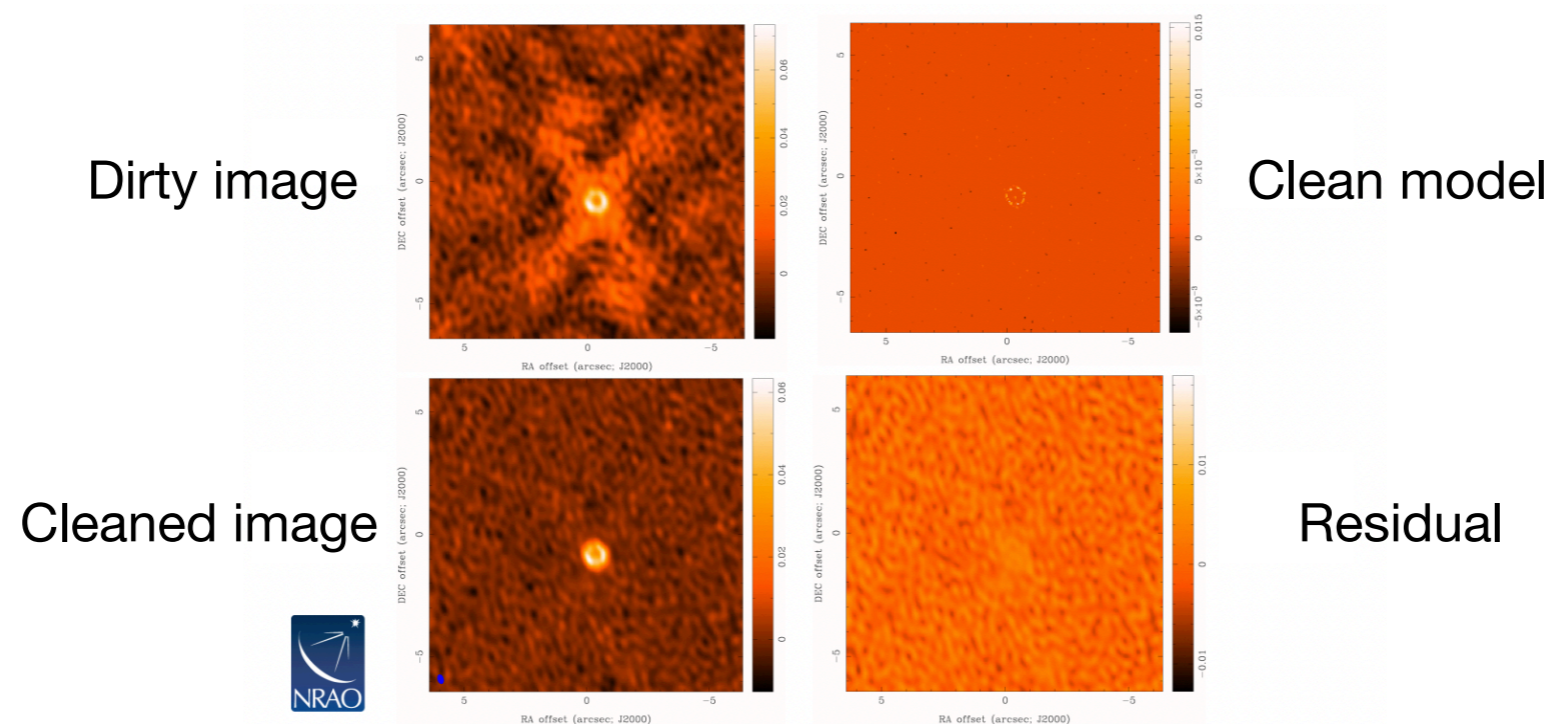
Residual

Why re-CLEAN ALMA data?

The ALMA pipeline already images your data for you, so why bother re-doing it?

- Continuum identification/subtraction may need revision

- Cleaning may not be deep enough (lots of signal still in residual)

- You may want to tweak cleaning parameters e.g. weighting scheme

- If your data are too big, the ALMA pipeline can perform various flavours of size mitigation (e.g. spectral/spatial binning, field cropping, dropping entire SPWs)

- Sometimes ALMA will deliver images in which the cleaning diverged and introduced significant artefacts. These cases *need* to be re-cleaned.

- If you have multiple 12m array configurations and/or 7m + Total Power, ALMA processes and delivers these separately — *you* will need to combine them

Hands-on Target: PN_Hb_5

- "Planetary Nebula Hubble 5"
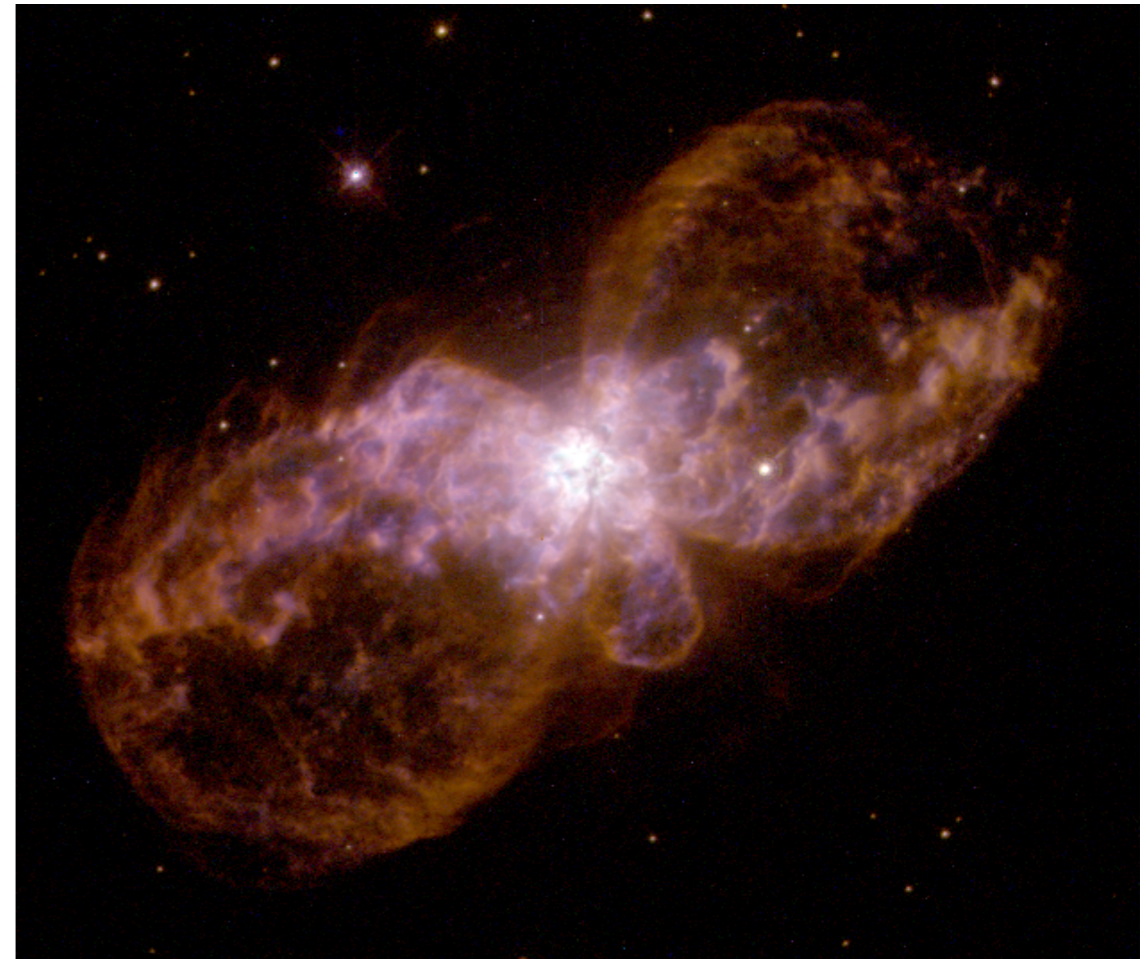
- Bi-polar planetary nebula

- Distance ~ 1.7 kpc

Image: HST (WFPC)

Credits: Bruce Balick, Vincent Icke, Garrelt Mellema, NASA

PN_Hb_5 ALMA data

- ALMA project ID: 2022.1.00401.S

- Scheduling Block (SB): PN_Hb_5_a_06_TM2

- Band 6 (representative freq. = 230.56 GHz)

- 4 spectral windows (SPWs): CO, CN, CS, HNCO

- Two different ALMA 12m observations:

  - TM1: longer baseline / higher angular resolution

  - TM2: shorter baseline / lower angular resolution

- We will use the **TM2** data only (smaller -> faster processing)

PN_Hb_5 ALMA data

uid___A002_X1003af4_Xa540.ms.split.cal

- You should have this calibrated MS if you ran `scriptForPI.py`

- Science target only calibrated MS is available <u>here</u> (~ 2.1 GB)

Imaging scripts (see meeting page)

- These are the scripts that we will walk through in this example to clean the continuum (and line data tomorrow)

Splitting out the target data

- For this particular dataset there are four science spectral windows

- The calibrated MS contains many more targets and SPWs (for e.g. calibration targets)

- To simplify things and reduce the data volume, we can split out the science target and SPWs

```
split(  vis            = filename,

        outputvis      = filename +'.target',

        field          = 'PN_Hb_5',

        spw            = '25,27,29,31',

        datacolumn     = 'corrected')
```

Note: splitting out SPWs re-indexes them. In the output file the SPWs will now be 0, 1, 2, 3

If CASA complains about having no CORRECTED datacolumn, change to 'data'

Continuum imaging

- Before imaging the continuum, we need to identify any molecular line emission — this will contaminate the continuum if we don't exclude it

  - One method is to use the frequency ranges identified by the ALMA pipeline (though this is not always ideal, be sure to check!)

  - Another method is to look at each spectral window using the CASA task `plotms`, and manually identifying the continuum ranges:

Example: SPW 0

2 prominent lines

These must be excluded for continuum imaging
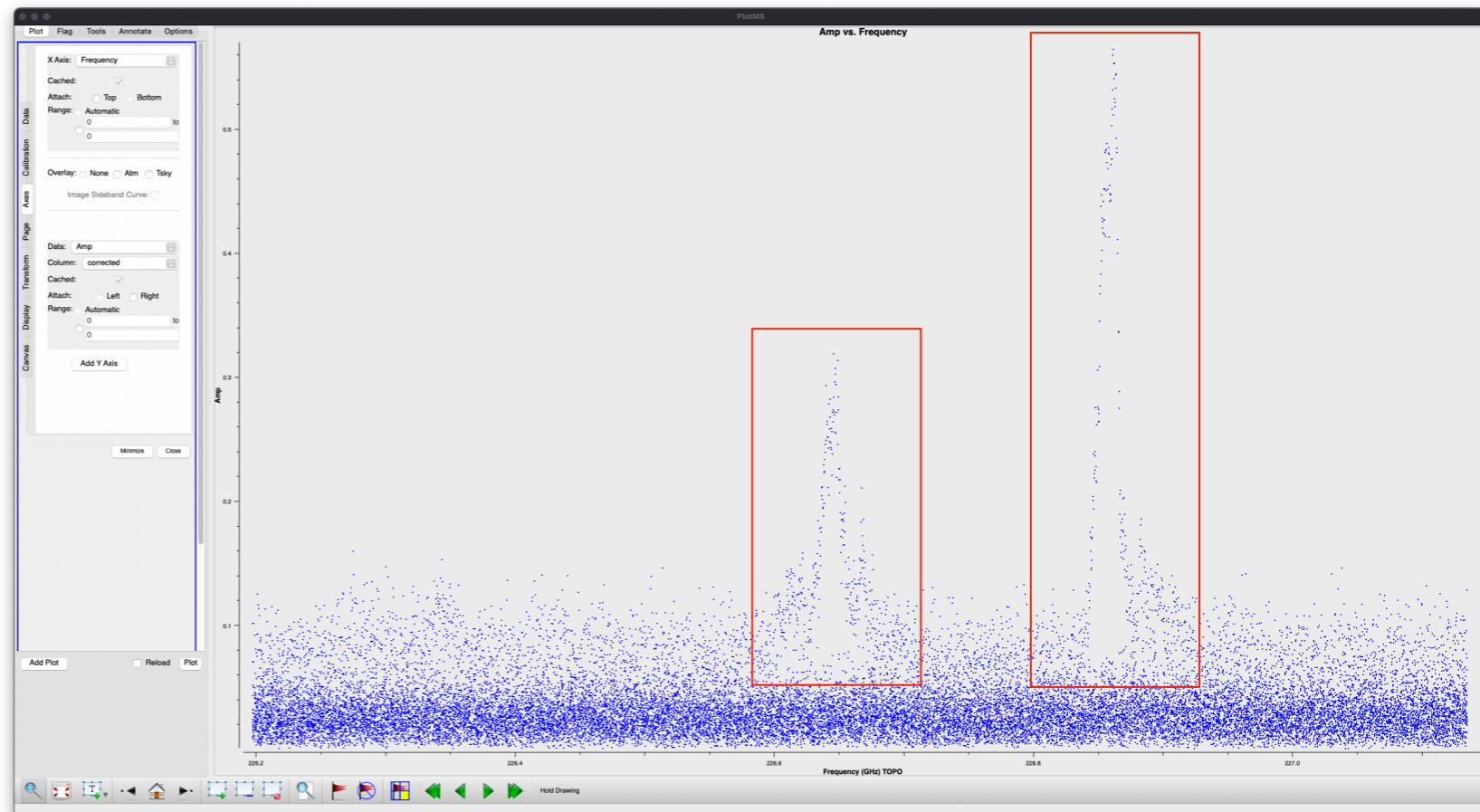
Continuum imaging

To image the data, we will use the CASA task `tclean`. We'll start by making a 'dirty image' (0 clean iterations) — this will give us a first look at the data and allow us to refine our choice of parameters. Let's take a look at some of the parameters:

```
tclean( vis          = visfile,

        imagename    = 'PN_Hb_5.cont.dirty',

        gridder      = 'mosaic',

        spw          = contchans,

        specmode     = 'cont',

        imsize       = [320, 320],

        cell         = '0.22arcsec',

        deconvolver  = 'hogbom',

        niter        = 0,

        weighting    = 'briggs',

        robust       = 0.5,

        interactive  = False)
```

Name of input measurement set

Prefix of output image files

Field name to be imaged

Channel/frequency ranges to generate continuum

Spectral mode ('cont' or 'mfs' for continuum)

Image size (2*Field of view / pixel size)

Pixel size (generally ~ $\theta/5$, θ = angular resolution)

Deconvolution algorithm to be used

Number of clean iterations (0 -> restore only residuals)

Weighting scheme to be used

Robust parameter for Briggs weighting (robust = -2 gives uniform weighting. robust = 2 gives natural weighting)

Option to clean using interactive GUI

Continuum imaging

Dirty image
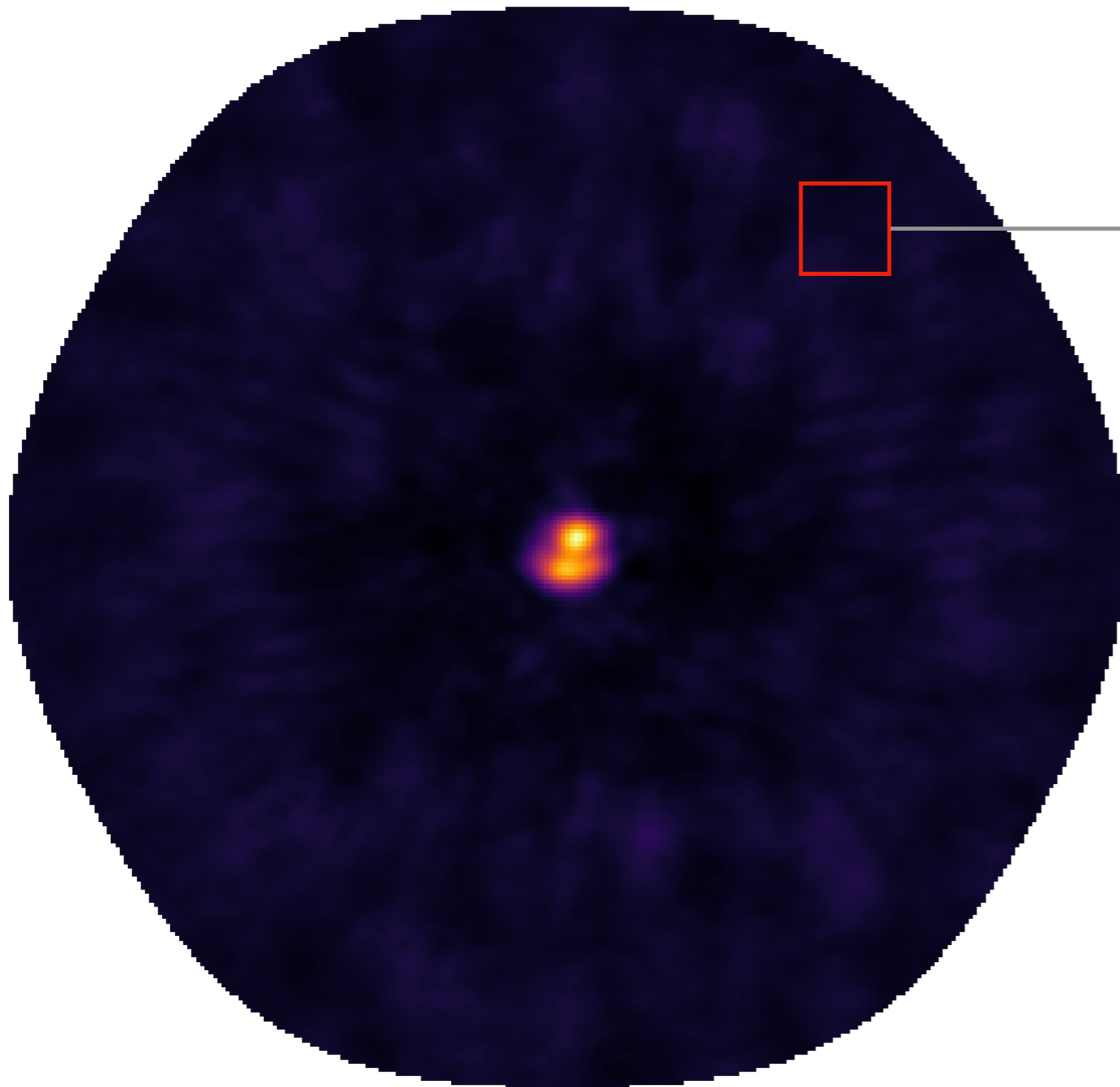
Measure RMS in a relatively blank part of the field

Use this to guide value for cleaning threshold

Typically something like 3xRMS is a good starting point

Continuum imaging

- Based on the RMS in our dirty image, we can specify a sensible cleaning threshold, typically N x RMS, where N is ~ 1-5

- Need to set number of clean iterations. If the cleaning threshold is sensible, this can be arbitrarily high, as the cleaning should stop once the threshold has been reached

  - A poor choice of cleaning parameters may lead to divergence and general weirdness

- Finally, a quick note on (auto-)masking, weighting, and primary-beam correction …

Continuum imaging (masking)

- We can use a cleaning mask to tell the algorithm where there is real emission to be cleaned. This can be done by:

  - Providing a pre-made mask as a cleaning parameter
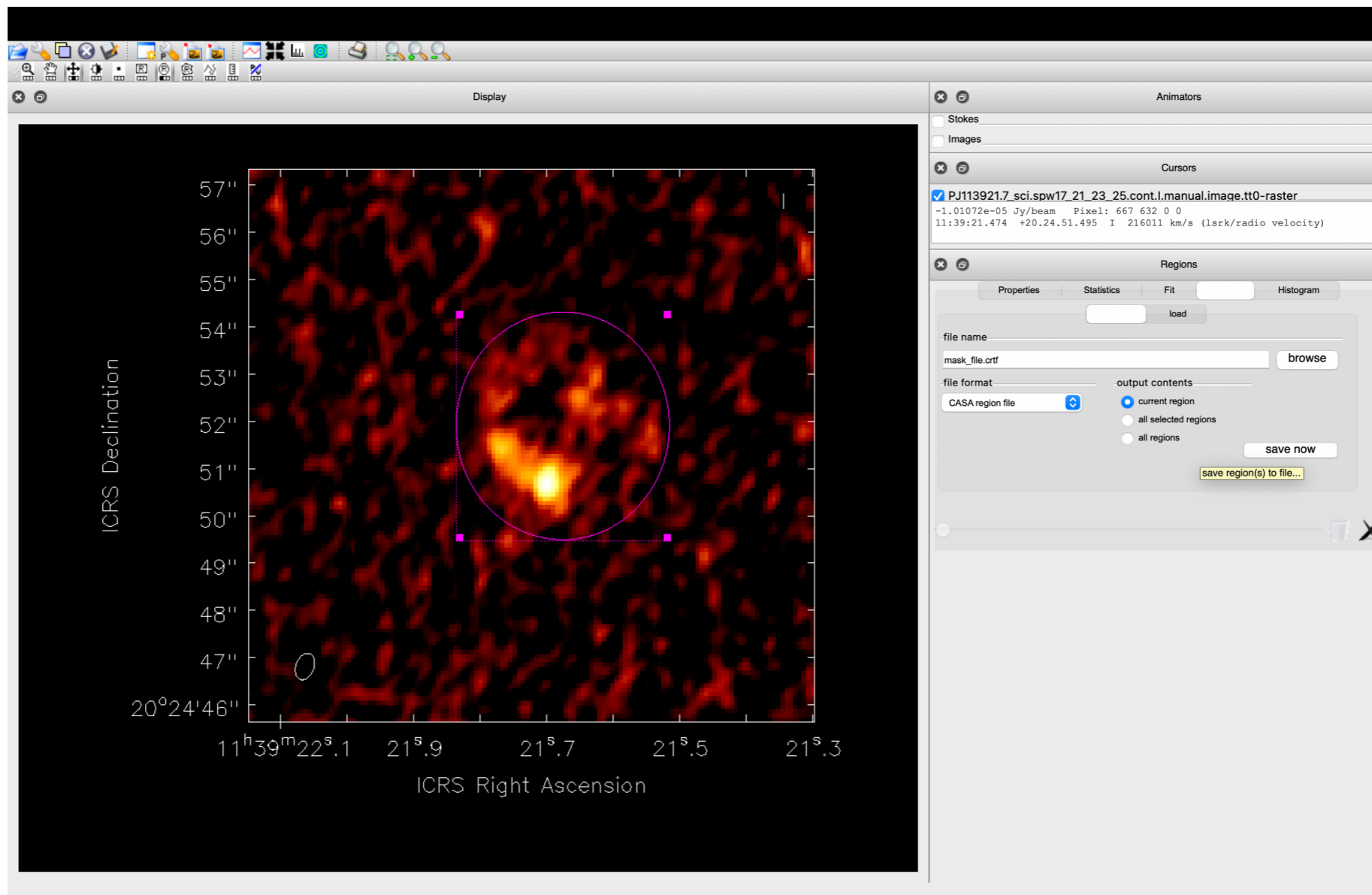
Continuum imaging (masking)

- We can use a cleaning mask to tell the algorithm where there is real emission to be cleaned. This can be done by:

  - Using the interactive cleaning GUI to manually draw a mask

Continuum imaging (masking)

- We can use a cleaning mask to tell the algorithm where there is real emission to be cleaned. This can be done by:

  - Using the built-in auto-masking functionality

  - https://casaguides.nrao.edu/index.php/Automasking_Guide

- Much more convenient, and can do a very good job

- Requires careful choice of parameters — the default parameters *typically* do a reasonable job, but can often be improved

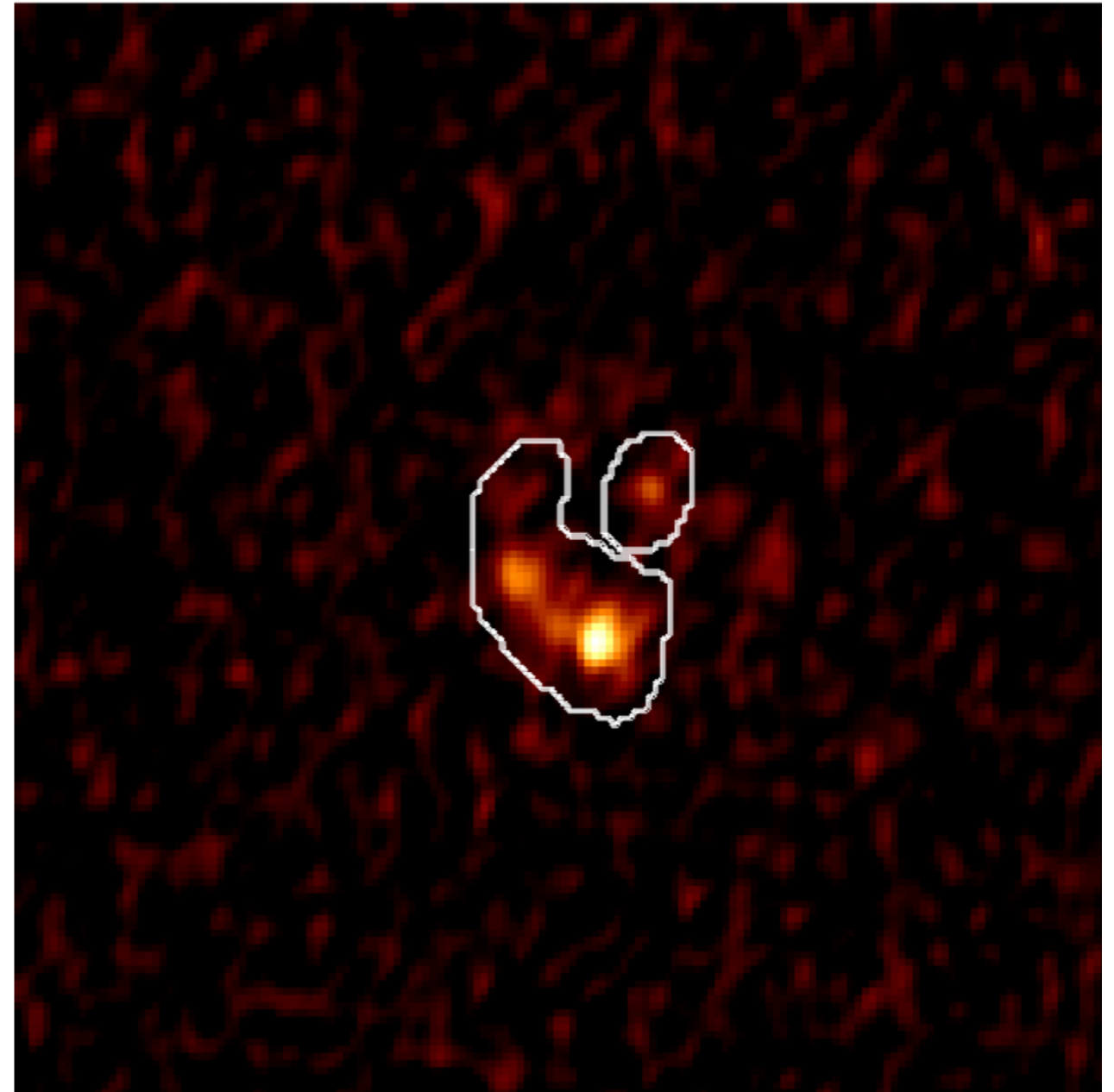- Can significantly increase computation time for large data or complex emission

Automasking
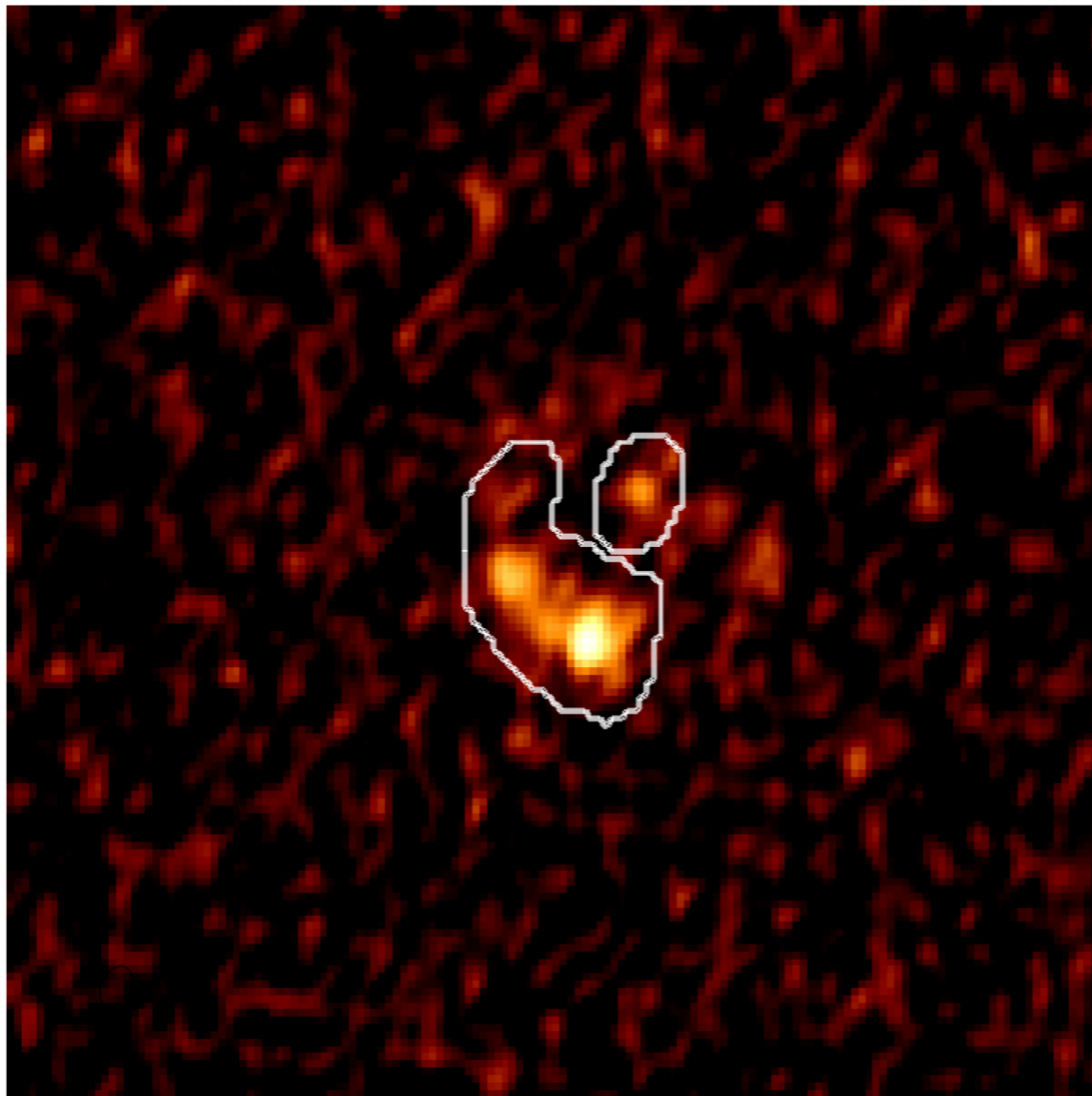
Need to be more strict with masking thresholds
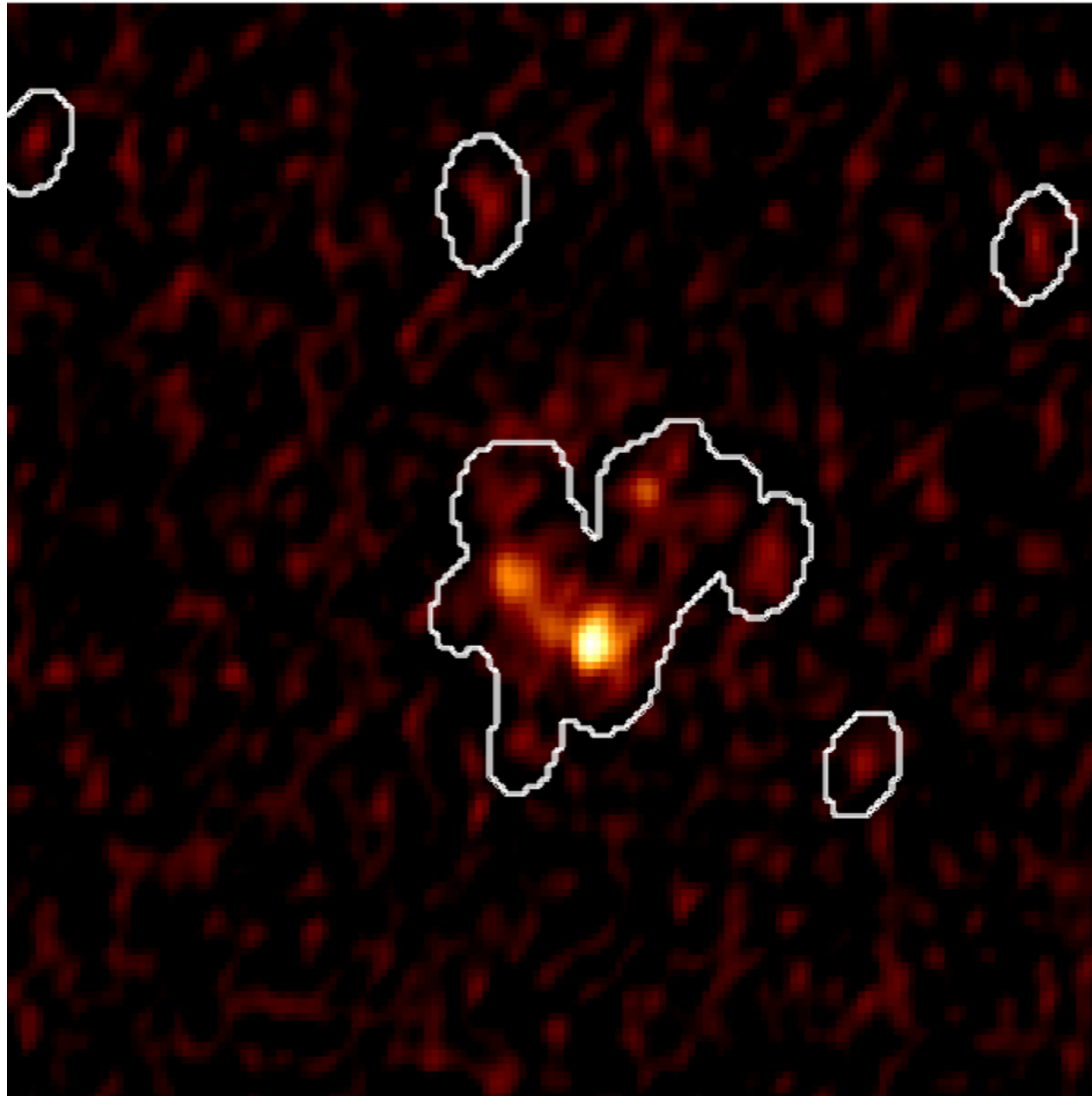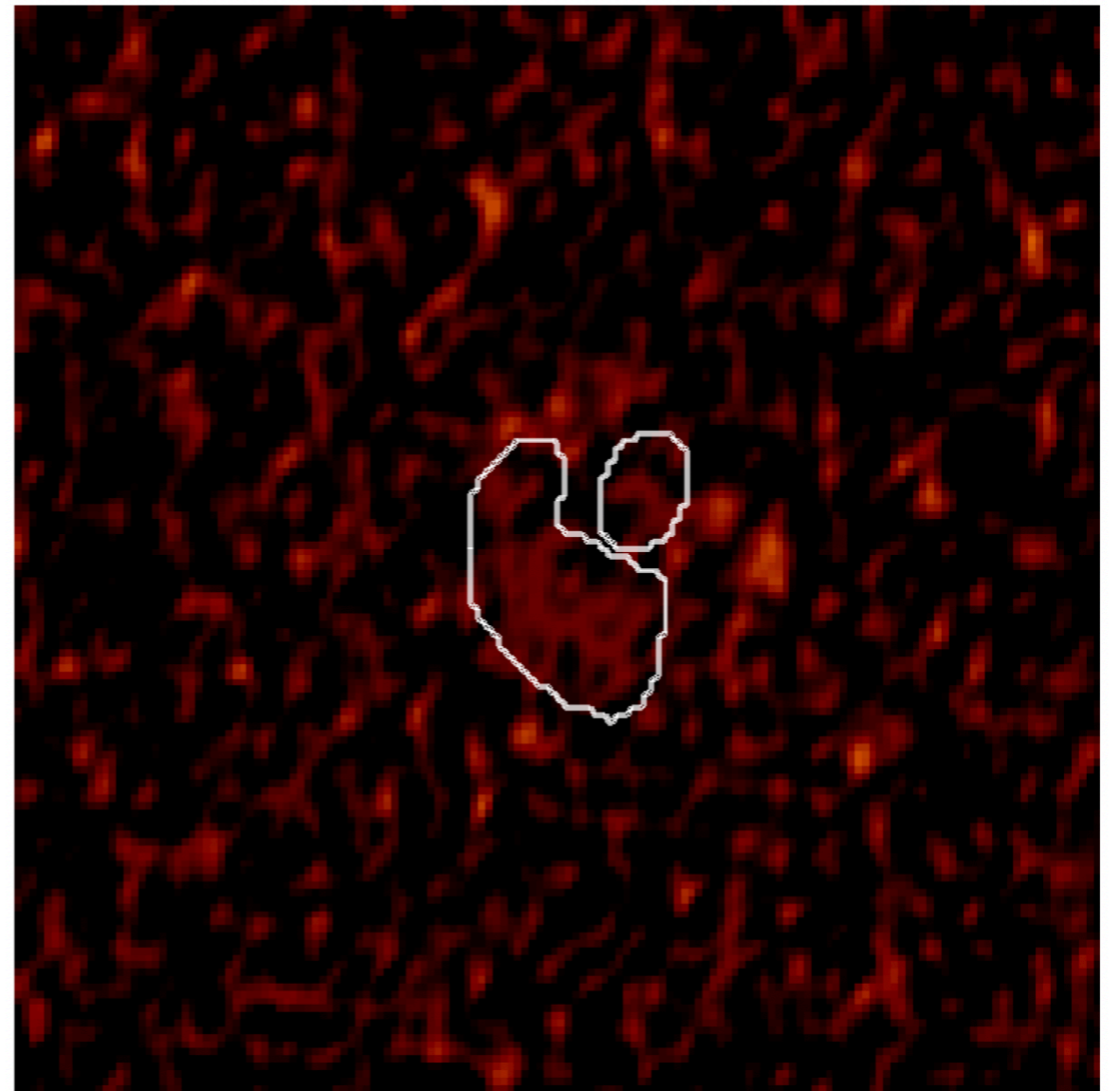
Better …

Automasking

Cleaned image (+ mask)

Residual (+mask)

Automasking parameters

`usemask = 'auto-multithresh'`

- `noisethreshold`: sets the noise threshold above which emission is masked during the initial round of mask creation

- `sidelobethreshold` = sets a threshold based on the sidelobe level, above which significant emission is masked during the initial round of mask creation

- `lownoisethreshold`: defines threshold into which the initial mask is expanded to include lower signal-to-noise regions

- `minbeamfrac`: minimum size a region must be to be retained in the mask (as a fraction of the beam size)

- `growiterations`: number of iterations per clean cycle for mask growth. A larger number will allow the mask to grow to capture fainter, more extended emission (if present), but can increase the processing time significantly

Note that either `noisethreshold` or `sidelobethreshold` is used depending on which threshold is higher.
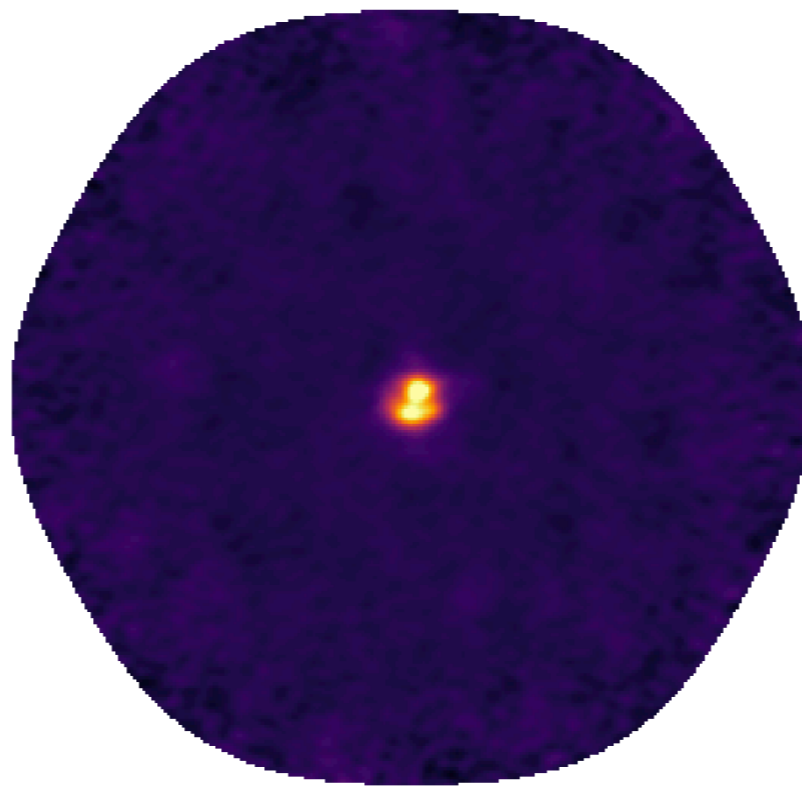
Weighting schemes

- Choice of weighting has a significant impact on the resultant image

- Trade-off between angular resolution and sensitivity

  - Need to decide what is most important for your science

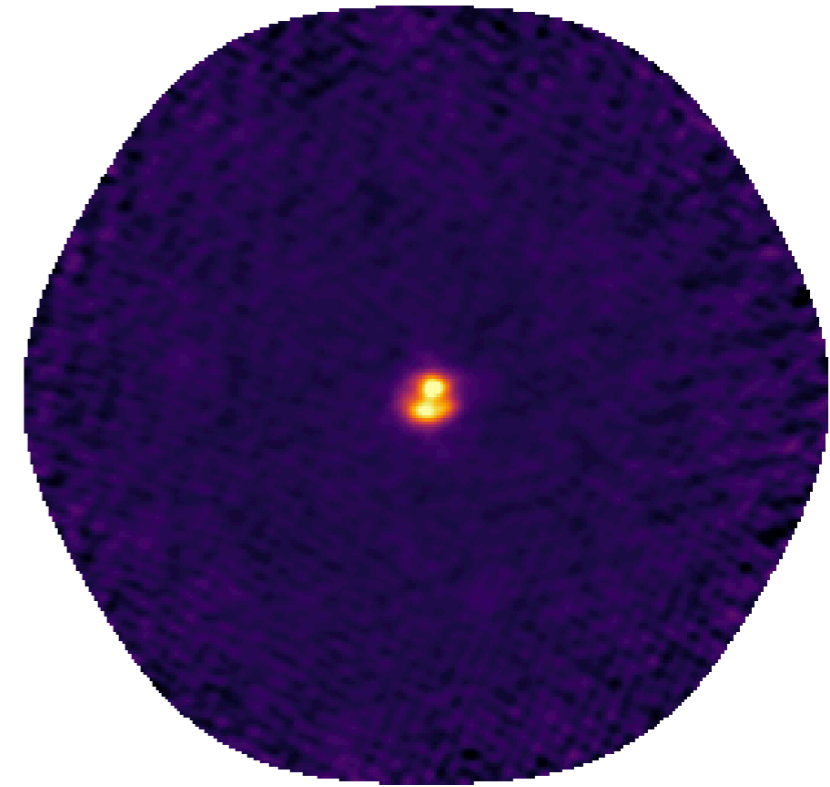 - Robust = 0.5 is often used — good compromise between the two
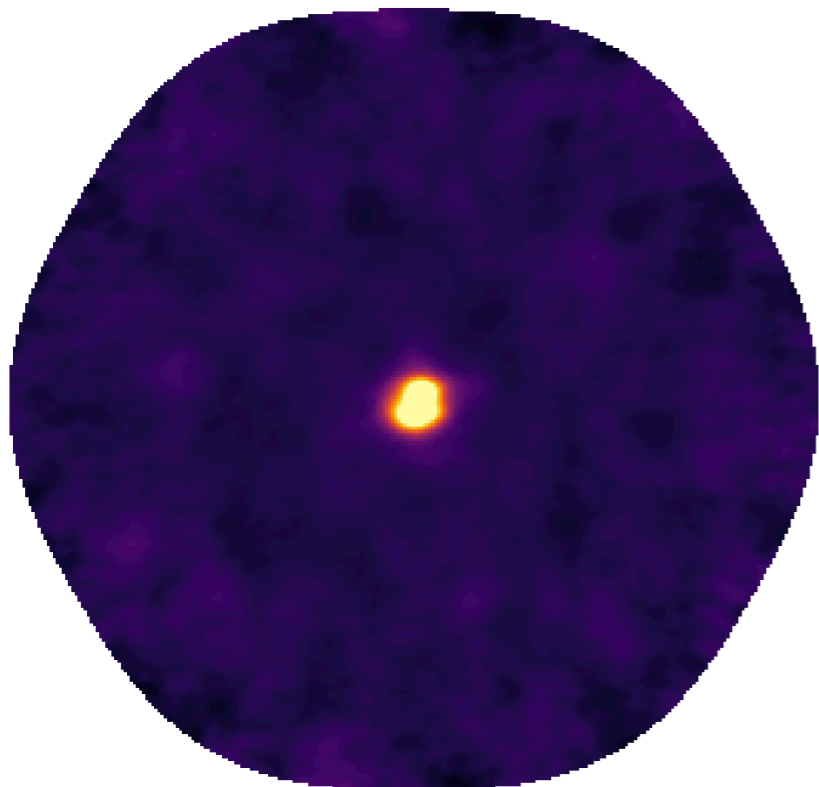
**Robust = 2 (Natural)**   **Robust = 0**   **Robust = -2 (Uniform)**

Primary beam correction

- Antenna response is not uniform across field of view — inherently noisier at the edges

- This is not accounted for during imaging, and so we need to correct for it

- This can be done in two ways:

  - Setting `pbcor=True` during cleaning. This will output two images — with and without primary beam correction

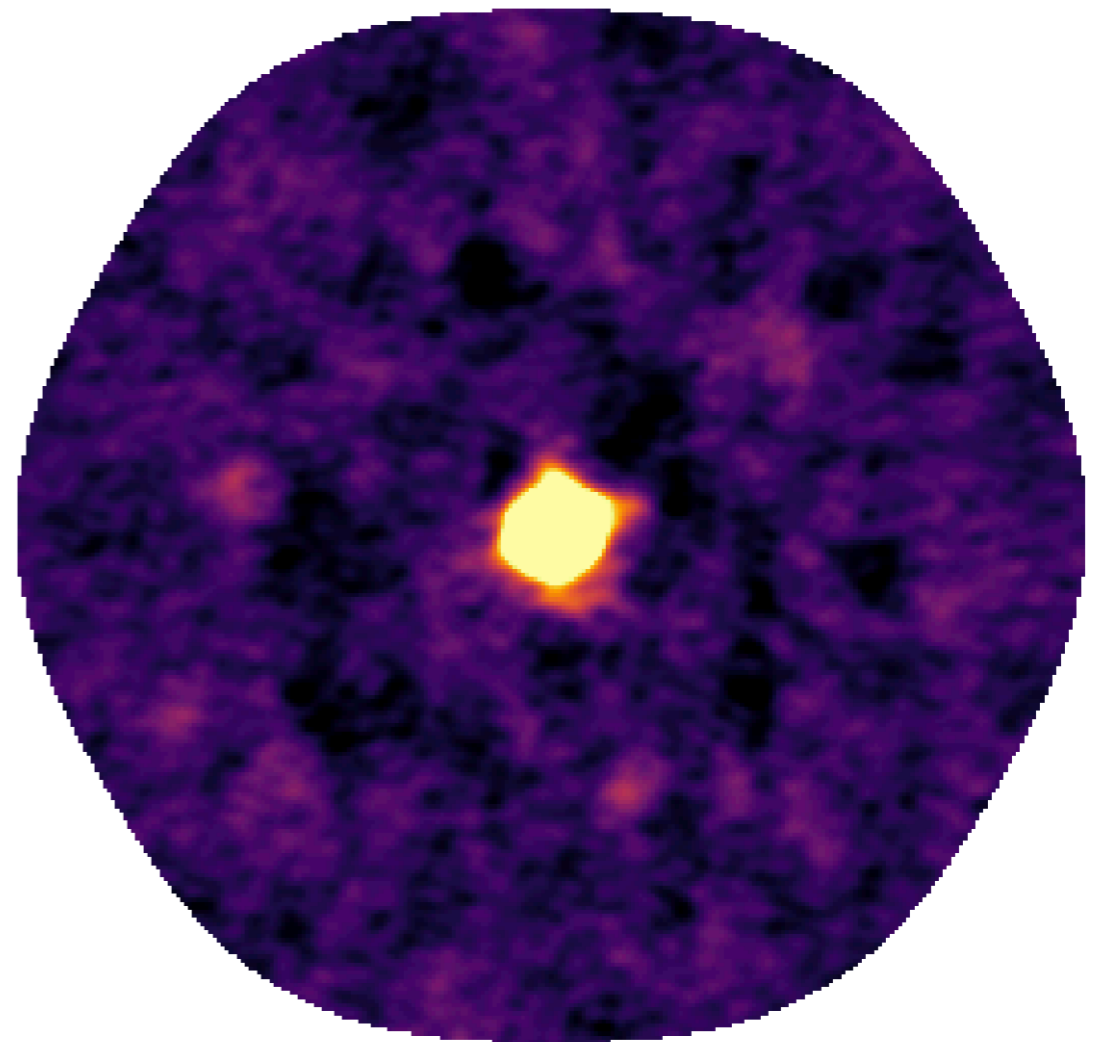 - Using the `impbcor` task post-cleaning
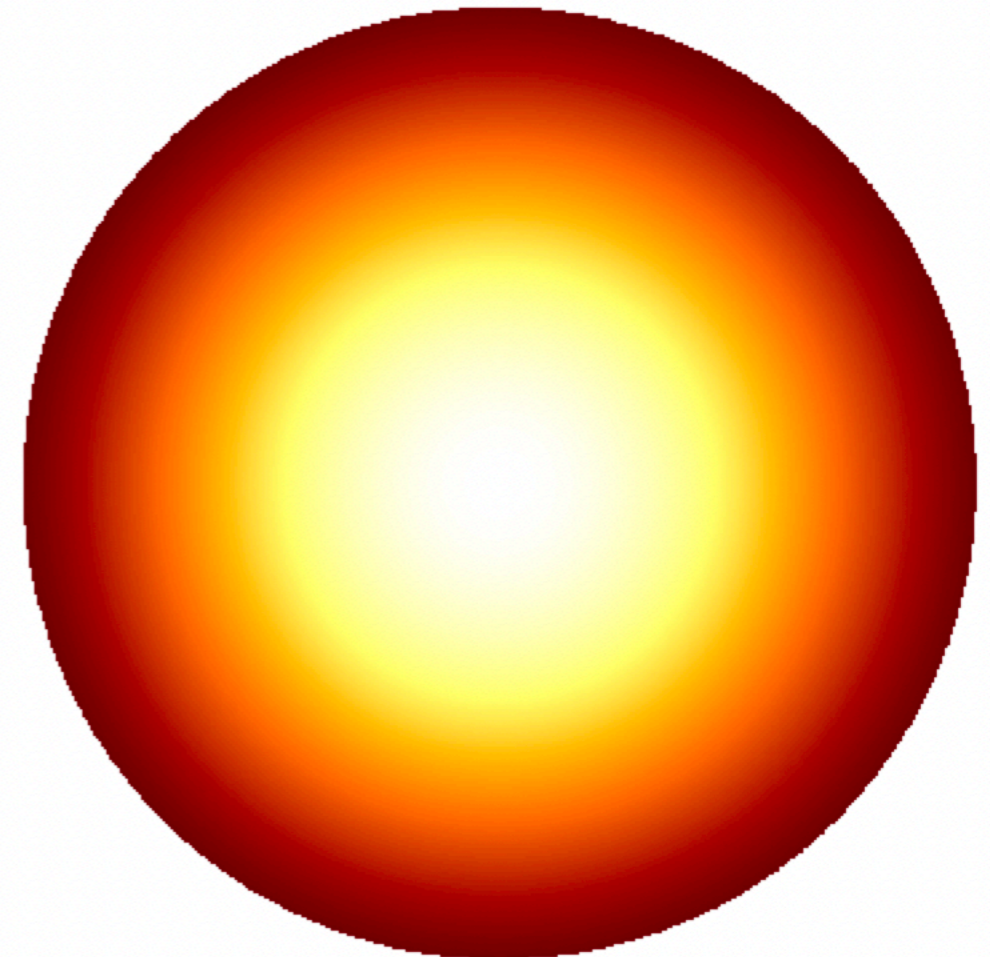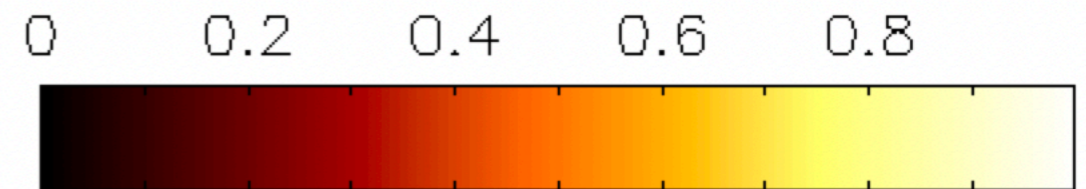
Primary beam correction

With PB-correction
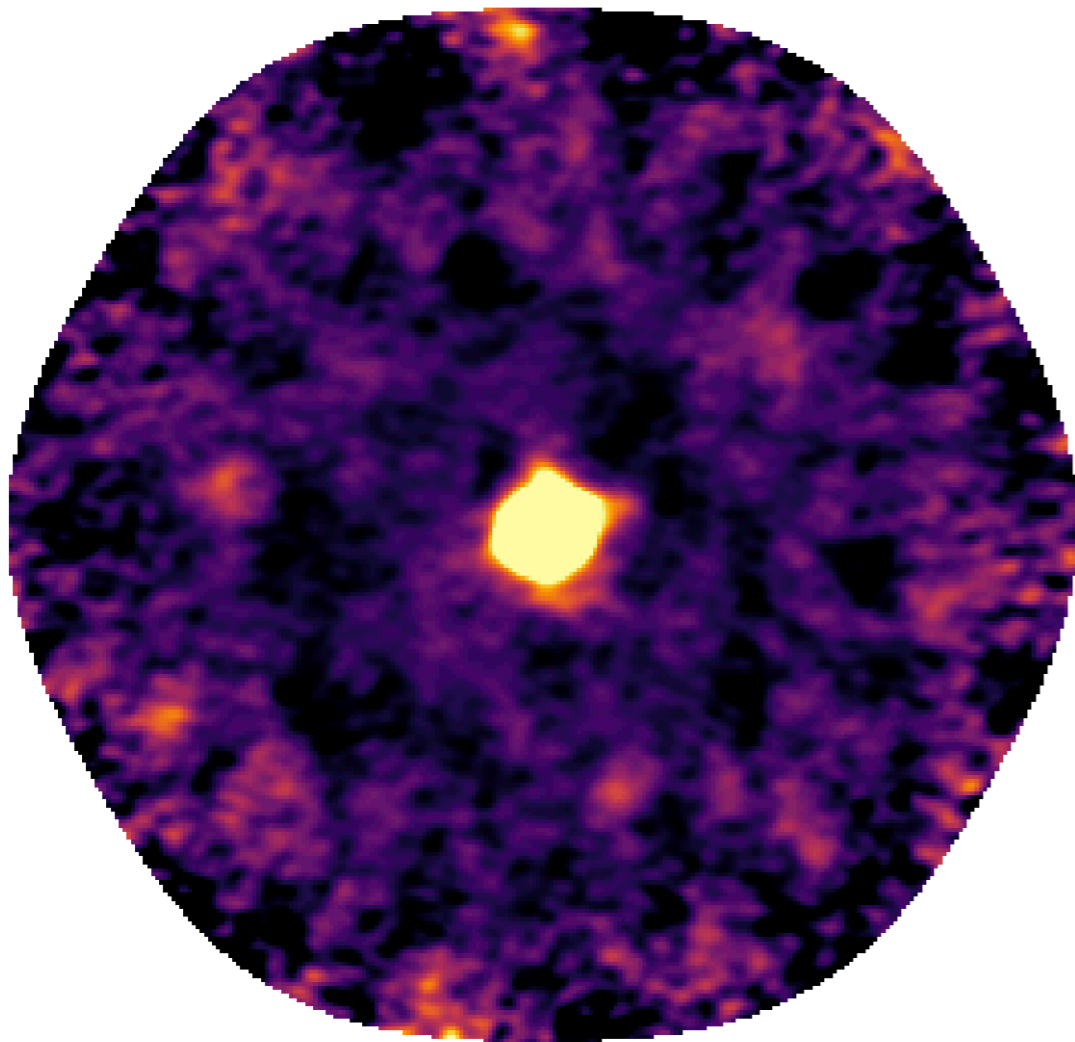
Without PB-correction

Continuum imaging

- Let's try full clean of the continuum using:

  - Cleaning threshold (based on RMS in dirty image)

  - `pbcor = True` (to correct for primary beam response)

  - Masking

    - the script has auto-masking pre-filled, but you are encouraged to experiment with manual masking, and then automasking with various parameter combinations

  - A range of robust values

  - Plus any other parameters you want to play with!