Center of Astrophysics and Related Technologies

PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE

CONICYT Ministerio de Educación Gobierno de Chile

FONDECYT Fondo Nacional de Desarrollo Científico y Tecnológico

Federica Ricci - PUC

What I would like to do: Unveiling the black hole mass - host galaxy connection in obscured accreting supermassive black hole

What I will tell you today:

NIR BASS - technical report (no exciting science yet, sorry)

BASS@Gainesville 2-7 January 2019

Motivation

#nir_obscured_bhmass

NIR measurements of obscured black holes using FIRE/Xshooter/triplespec/dr1 BASS data

General goal: build a *complete* and *unbiased* sample of AGN, thus particular effort on *obscured* AGN
Also, try to target *lower-L* AGN than previous BASS NIR data, by using large telescope (VLT, Magellan)

Why NIR?

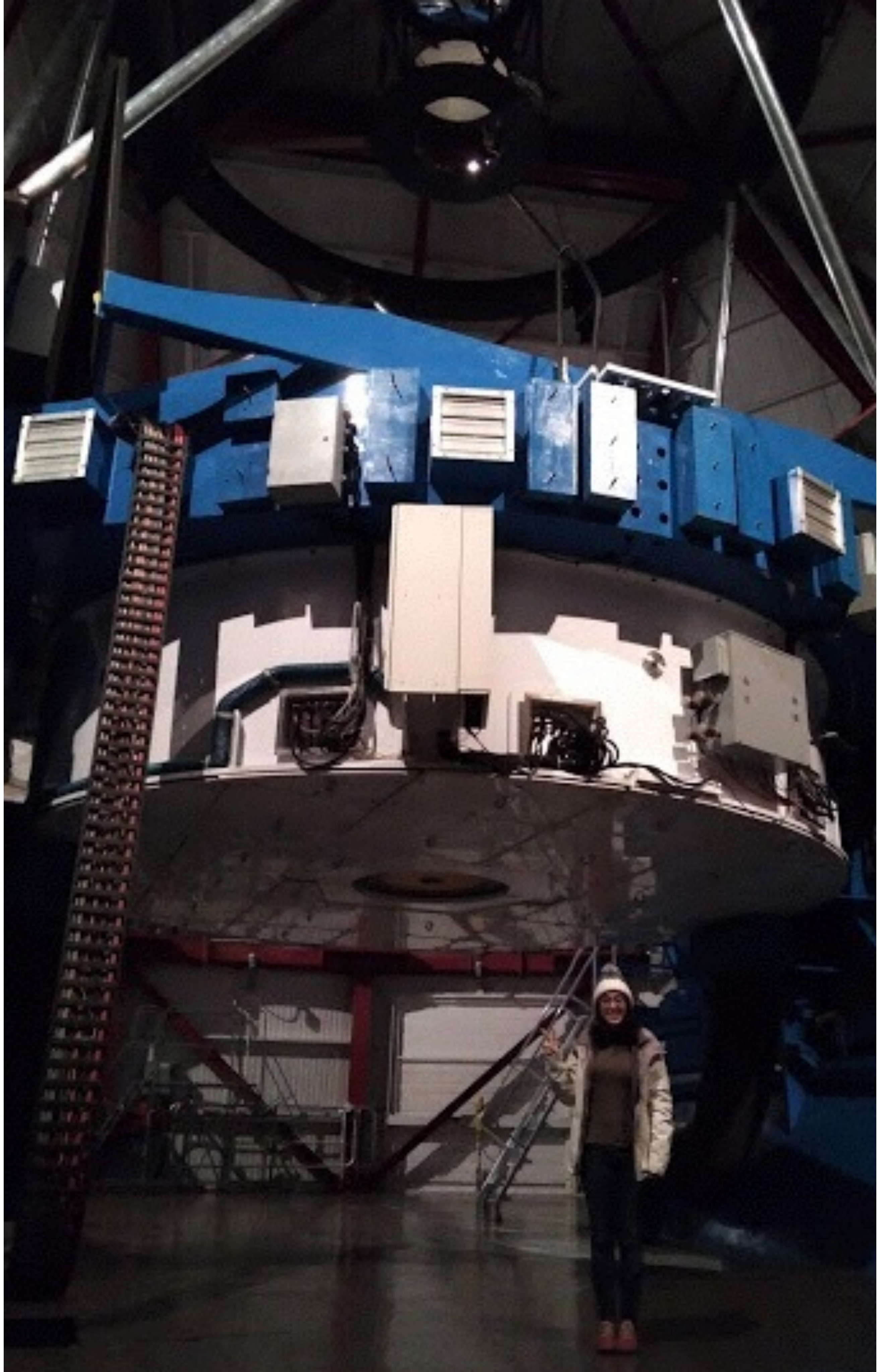
- ~10 times less sensitive to reddening than the optical
- Mbh from Ca II triplet and/or CO H- K- band heads and/or
- hidden broad line region → can complement/help the optical information (for instance disentangle between real 1.9 or 2 with outflows and understanding systematics)
- several coronal lines
- (more distant future: benchmark for JWST AGN studies)

NIR BASS DATA

- Palomar/Triplespec
  - 1 night
- VLT/Xshooter (PI Benny, ~100 obs)
- Magellan/FIRE
  - 2 nights (PI Ezequiel, 8+14 obs)

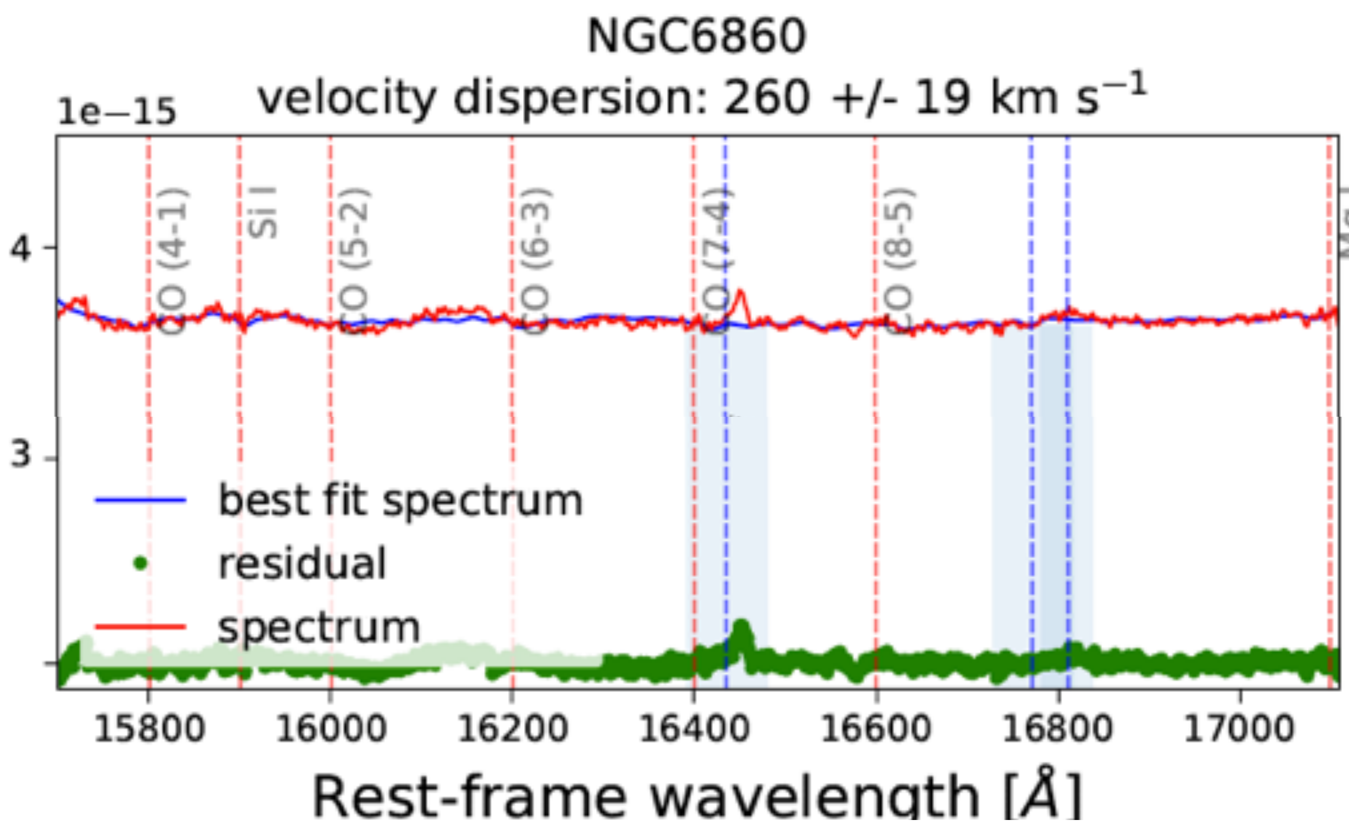
FIRE data reduction completed, even though molecfit need to be implemented

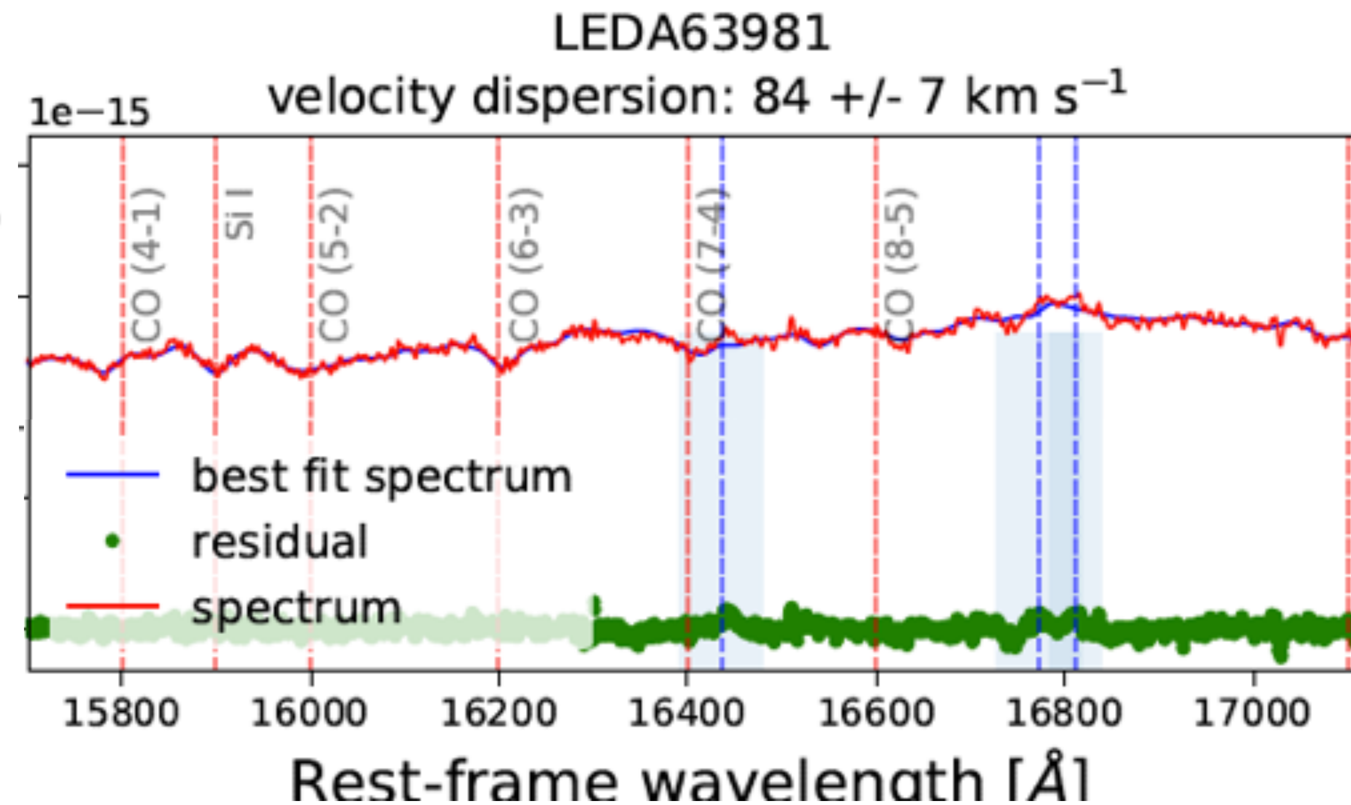
- Coming this semester, directly involved as observer:
  - 1 night 9 March (PI Federica, ~15 exp)
  - 1.5 nights 14-15 April (PI Mislav, ~20-25 exp)



DATA analysis (only FIRE at the moment)

1. **ppxf**: measure stellar velocity dispersion in the Ca T, **CO H-**/K- band heads. CO H- is the most promising, highest S/N most of the times





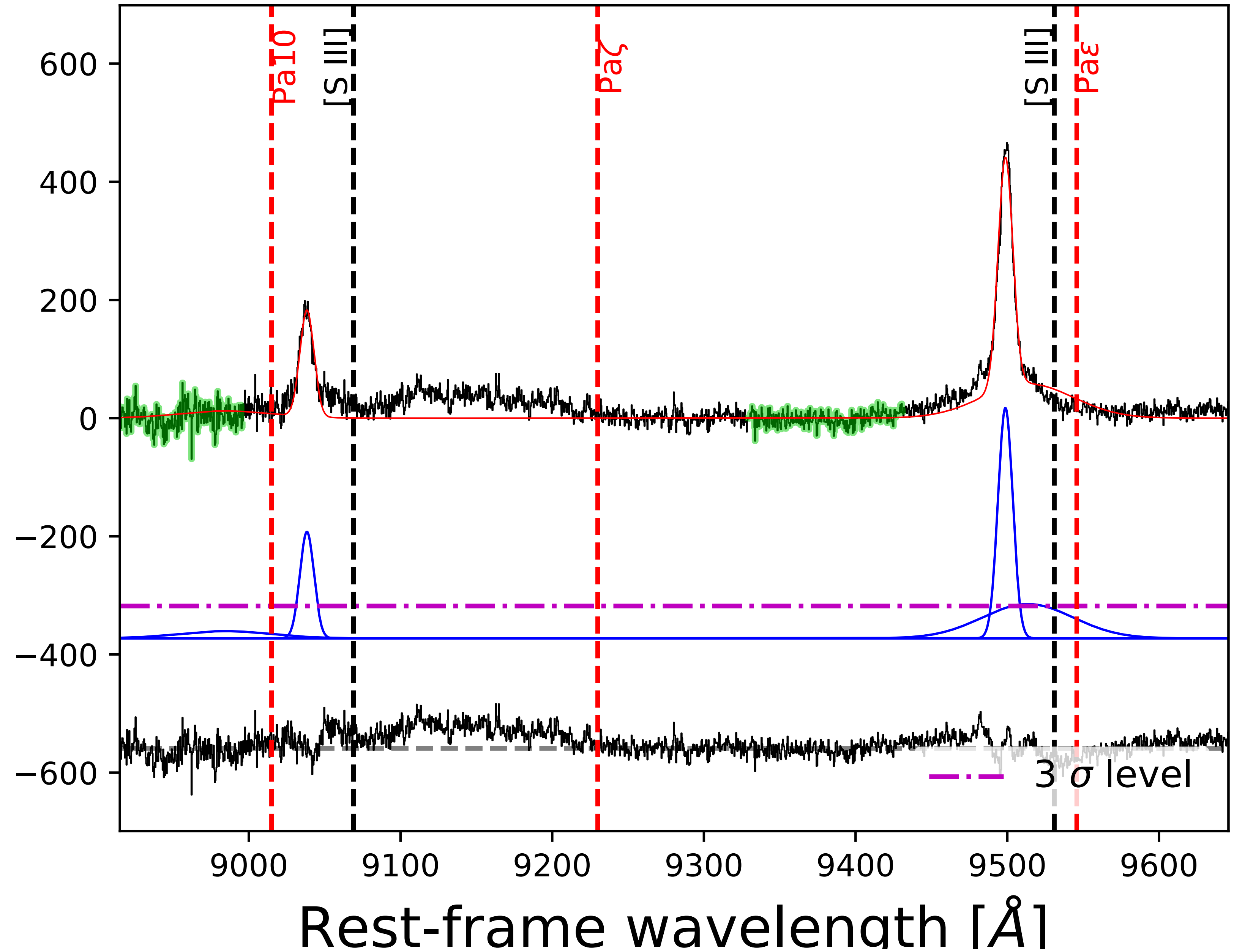
- up to now, acceptable CO H fit for 8/22
- TBDone: better fit CO H changing the masking region
- TBDone: run ppxf on CaT
- TBDone: run ppxf on CO K

DATA analysis (only FIRE at the moment)

2. pyspeckit (v 0.1.20, python 2.7.15): Emission line fitting.

The code (from Isabella Lamperti) divides the spectrum in 7 regions (e.g. P14, Pzeta, Pdelta, Pgamma, Pbeta, SiX, Palpha, Brgamma)

The first thing is constrain the NLR, thus the 1st region to fit is the one containing the SIII 9531 A



pyspeckit - CODE debugging + updates

Major updates (at the moment only Pa_zeta region of the spectrum):

1. tie the sigma of the narrow gaussians in velocity space rather than in wavelength space -> more physical

$$\sigma_N^{el,1} = \sigma_N^{SIIIb} \times \frac{\lambda^{el,1}}{\lambda_{SIIIb}}$$

2. tie the central wavelength of the narrow component to have the expected ratio (restframe) wrt the SIII 9531 rather than a linear offset in velocity -> advantage of neglecting errors on redshift (if any)!

$$\lambda_N^{el,1,FIT} = \lambda_N^{SIIIb,FIT} \times \frac{\lambda_N^{el,1}}{9531.0\mathring{A}}$$

```
for k in range(len(NIR_emis)):
    offset =NIR_emis[k][0]/ NIR_emis[1][0] #NIR_emis[k][0]- NIR_emis[1][0]
    if k==1: # Fix the width of the [SIII]a line
        if flag[7]==F and flag[6]==F:  # Fix the amplitude of the [SIII] line, if there is not the Pa_eps
            tied=tied+['p[3]', '','']
        else:
            tied=tied+['', '','']  # [SIII]b line is free
    else:
        tied=tied+['', 'p[4]*{0}'.format(offset),'(p[5]/p[4])*(p[4]*{0})'.format(offset)]
```

pyspeckit - CODE debugging + updates

Major updates (at the moment only Pa_zeta region of the spectrum):

3. for the broad lines in the region (Pa10, Pazeta, Paeps), the central wav of the narrow component is tied with the wav ratio of SIII while the broad component is tied wrt the expected wavelength ratio of the Pazeta

$$\lambda_B^{el,1,FIT} = \lambda_B^{Pa\zeta,FIT} \times \frac{\lambda^{el,1}}{\lambda^{Pa\zeta}}$$

4. the sigma of the broad component is tied in wav to the Pazeta rather than in velocity (no update wrt Isabella's version)

$$\sigma_B^{el,1,FIT}[\AA] = \sigma_B^{Pa\zeta,FIT}[\AA]$$

```
        index = len(NIR_emis)*3
        print 'flag', flag
        for k in range(len(Pa_emis)):
            offset = Pa_emis[k][0]/NIR_emis[1][0]#Pa_emis[k][0]- NIR_emis[1][0]#ratio of Paschen lines wavelength wrt Siiib
            print Pa_emis[k][1], flag[ind+2*k]
            if flag[ind+2*k]==True:
                tied=tied+['', 'p[4]*{0}'.format(offset), '(p[5]/p[4])*(p[4]*{0})'.format(offset)]
            else:
                tied=tied+['p['+str(index)+']', 'p[4]*{0}'.format(offset), '(p[5]/p[4])*(p[4]*{0})'.format(offset)]

            offset_broad = Pa_emis[k][0]/Pazeta#Pa_emis[k][0]-Pazeta#for the broad component I impose the ratio of Paschen wrt
            if flag[ind+2*k+1]== True:
                if k==1:
                    tied=tied+['', '', '']
                else:
                    if flag[4]==T:
                        tied=tied+['', 'p[16]*{0}'.format(offset_broad), 'p[17]']#'(p[17]/p[16])*(p[16]+{0})'.format(offset_br
                    else:
                        tied=tied+['', 'p[4]*{0}'.format(offset), 'p[17]']#'(p[17]/p[16])*(p[4]+{0})'.format(offset)]

            else:
                tied=tied+['p['+str(index+3)+']', 'p[4]*{0}'.format(offset), 'p[17]']#'(p[17]/p[16])*(p[4]+{0})'.format(offset
```

pyspeckit - CODE debugging + updates

Major updates (only Pa_zeta region of the spectrum):

5. fix the maximum range of velocity shift in the emission lines in velocity space rather than in wav -> up to ±500 km/s for narrow and up to ±1000 km/s for broad lines

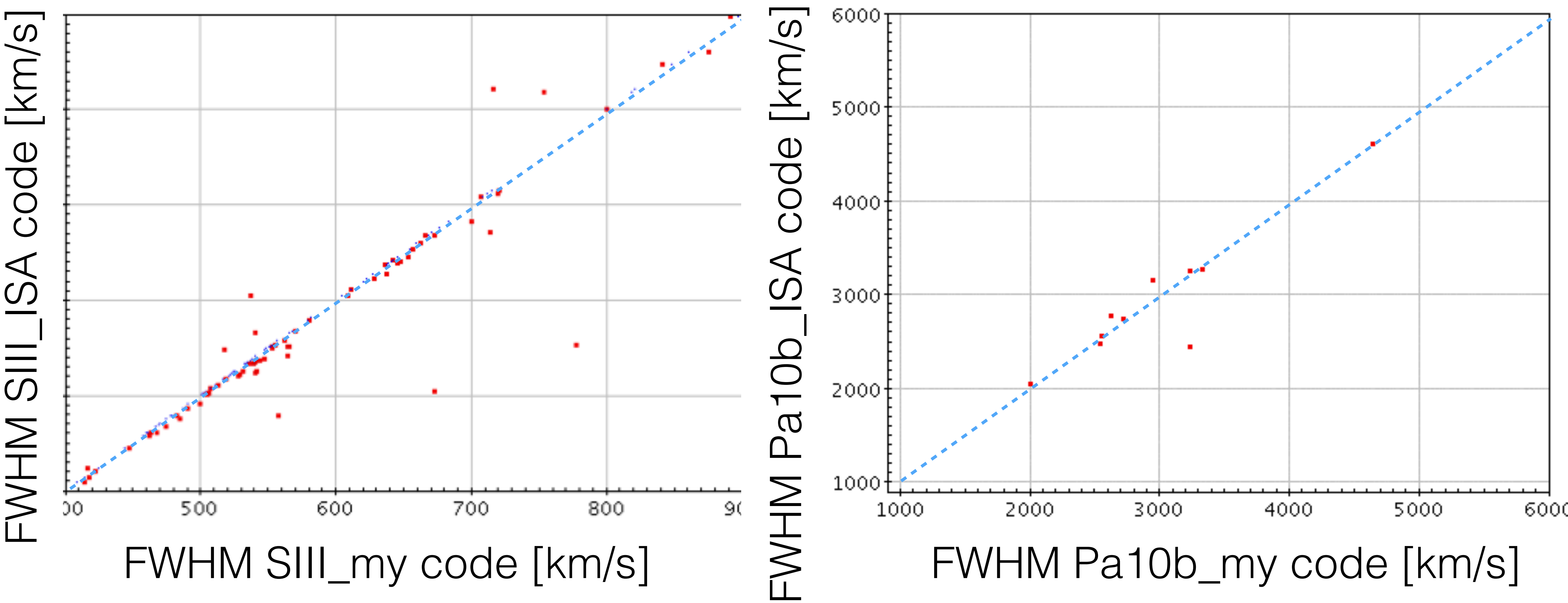
```
#maximal offset for the central wavelength
shift_v_n = 500 #km/s
shift_v_b = 1000 #km/s
shift_A_n = shift_v_n/c*wave_narr # Conversion in Angstrom


        if m < (nbr_lines-nbr_broad*2)*3:
            minp.append(element - shift_A_n) #changed from 8 A to 500 km/s
            maxp.append(element + shift_A_n)
            lmin.append(True)
            lmax.append(True)
        else:
            if p % 2 == 0:
                minp.append(element - shift_A_n)
                maxp.append(element + shift_A_n)
                lmin.append(True)
                lmax.append(True)

            else:
                # Broad component
                shift_A_b=shift_v_b/c*guesses[m]
                #print 'm=',m,' guess', guesses[m],' shift_A_b', shift_A_b
                minp.append(element - shift_A_b) #changed from 30 A to 1000 km/s
                maxp.append(element + shift_A_b)
                lmin.append(True)
                lmax.append(True)
```

pyspeckit - CODE debugging + updates

Comparison with DR1 in the Pazeta region



(near) FUTURE: What I need to do

PART I - data

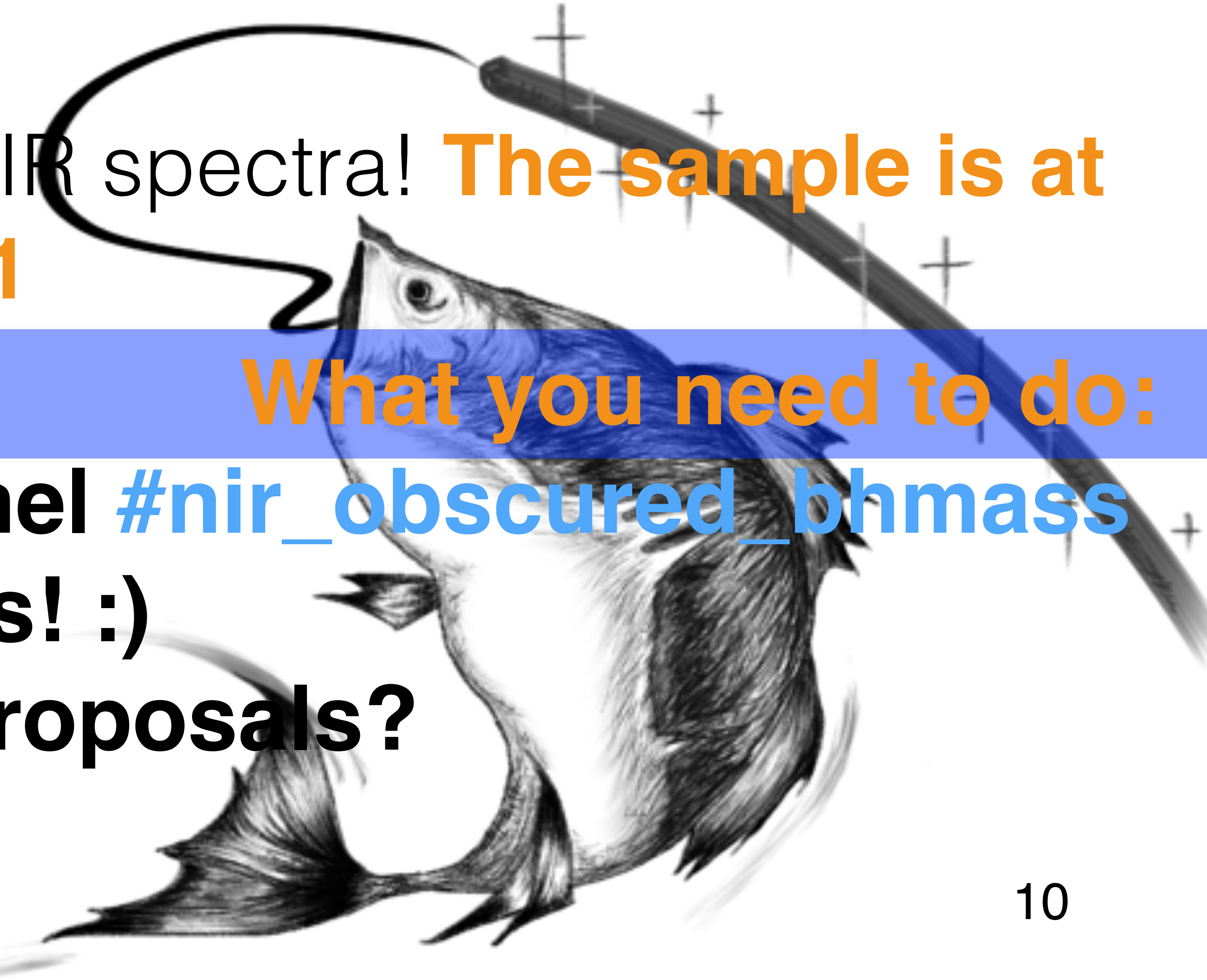
- ✳ improve the telluric correction of FIRE with molecfit
- ✳ reduce the Triplespec data
- ✳ get more data.. going to observe a couple of FIRE runs

PART II - pPXF

- ✦ (wait and) include the NIR high resolution stellar template

PART III - pyspeckit

- ‣ implement the same the fitting criteria/constraints for the other emission line regions
- ‣ fit all the emission lines in BASS NIR spectra! **The sample is at least doubled wrt BASS NIR DR1**



What you need to do:

- ✓ **Join the slack channel #nir_obscured_bhmass**
- ✓ **Give me suggestions! :)**
- ✓ **Want to talk about proposals?**