```
# Variable Star Astronomy
with undergraduates:
    import astropy
```

Python in Astronomy 2016

Matt Craig
Department of Physics and Astronomy
Minnesota State University Moorhead

Acknowledgements

- Current students
  - Nathan Walker
  - Laura Herzog
  - Michael Meraz
  - Connor Stotts
  - Erin Aadland
  - Andy Block
  - Elias Holte
  - Laura Maixner
  - Stefan Nelson
  - Jane Glanzer
  - Elizabeth Dougherty
- Colleagues
  - Juan Cabanela
  - Linda Winkler

Context

- undergraduate-only program
- 5 ± 3 new astronomy emphasis students/ year
  - 0.5/year go to graduate school
- primary responsibility is teaching
- prepare students for a (non-astronomy) career
- python introduced in first calculus-based physics course, continues throughout curriculum.

Constraints

MINNESOTA STATE UNIVERSITY MOORHEAD

- Some students use Windows
  - ★Anaconda python distribution.
- Cannot `from __future__ import experience`
  - ★Graphical interface to python for 1st year students.
- Need record of student work
  - ★ Traditional GUI app does not do that
- Use existing, well-supported packages
  - ★active dev + multiple devs + documentation + modularity = win!

Coding goals

MINNESOTA STATE UNIVERSITY MOORHEAD

- Start participating early
- Begin with little or no coding
- Develop coding skills over 4 years
- Introduce GitHub as early as possible
- Introduce git when and if needed
- Encourage contributions to open source projects

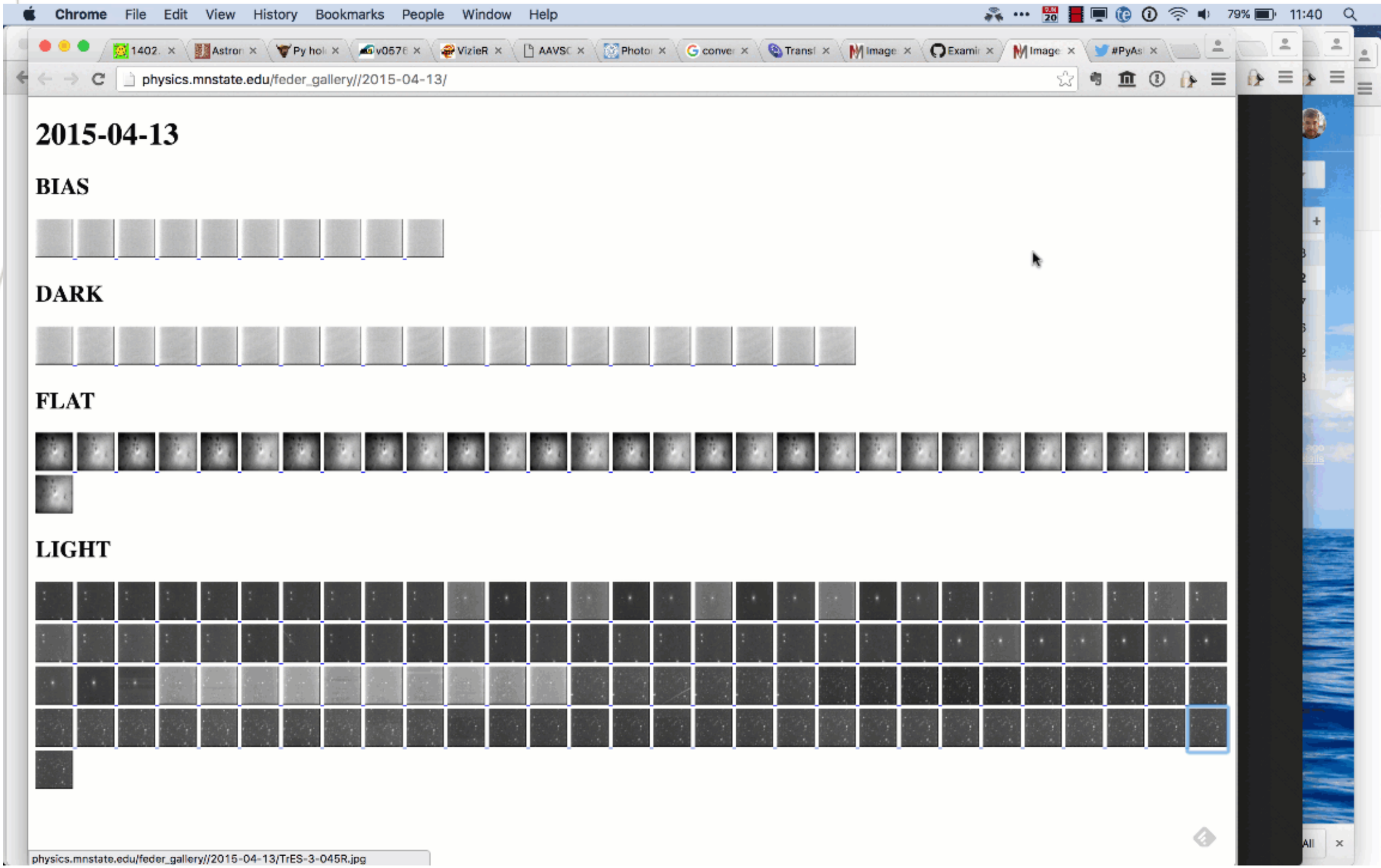
Variable Star Astronomy

MINNESOTA STATE UNIVERSITY MOORHEAD

- Look for new variable stars in fields containing known exoplanets
  - Take data
  - Add metadata
  - Review data
  - Reduce data
  - Identify sources
  - Perform aperture photometry
  - Differential ensemble photometry
  - Analyze time series

Visual review

MINNESOTA STATE UNIVERSITY MOORHEAD



Visual review

MINNESOTA STATE UNIVERSITY MOORHEAD

Examine staged data for 2015-04-13 #43

Edit | New issue

Open **feder-bot** opened this issue on Sep 7, 2015 · 0 comments

feder-bot commented on Sep 7, 2015 — feder-observatory member

Click here to edit README for this night

Image gallery for this night at:

http://physics.mnstate.edu/feder_gallery//2015-04-13

mwcraig added the **no gallery** label on Nov 5, 2015

Labels

None yet

Milestone

No milestone

Assignee

No one—assign yourself

Visual review

MINNESOTA STATE UNIVERSITY MOORHEAD

processed_images / nights / 2015-04-13-README.mc or cancel

Edit file | Preview changes | Spaces | 2 | Soft wrap

```
1  # 2015-04-13
2
3  ## Reviewed by:   your_name_here
4
5  ## Unusual images?
6
7  Are there any images that look unusual? List the file name of any unusual images for this night here, with description:
8
9  + `crazy-object-001R.fit` -- not sure what this is an image of
10 + `another-crazy-object-003B.fit` -- this looks more like a flat than a science image.
11 + `full-moon-005B.fit` -- nice satellite or plane track in here.
12
13 Delete this list if there are no unusual images.
14
15 ## Missing information?
16
17 Check these off if they are true:
18
19 - [ ] No images are missing filter information (except BIAS and DARK, which need no filter).
20 - [ ] No images are missing pointing information (RA/Dec and WCS)
21 - [ ] No images are missing object names (only applies to science images)
22 - [x] EXAMPLE checked-off box, please delete.
23
24 If any images are missing information and you have been unable to fix them please list
25 them below with a short description of the problem.
26
27 + `m34-002R.fit` -- no WCS, looks like there were maybe clouds.
28 + `m404-001B.fit` -- Not sure what object this, and googling `m404` got me a 404.
```

Commit changes

Update 2015-04-13-README.md

Add an optional extended description…

Commit directly to the `master` branch.

Create a **new branch** for this commit and start a pull request. Learn more about pull requests.

mwcraig-patch-16

Propose file change | Cancel

Data reduction

- **ccdproc 1.0** released last week!
- New features:
  - Combine images based on WCS
  - Cosmic ray removal with astroscrappy
  - Work with directories of images easily:

```
for d in dirs:
    print(d)
    ic = ImageFileCollection(d, keywords='*')
    for data, fname in ic.data(imagetyp='LIGHT', return_fname=True):
        if data.mean() > 4000.:
            print(fname)
```

Data reduction

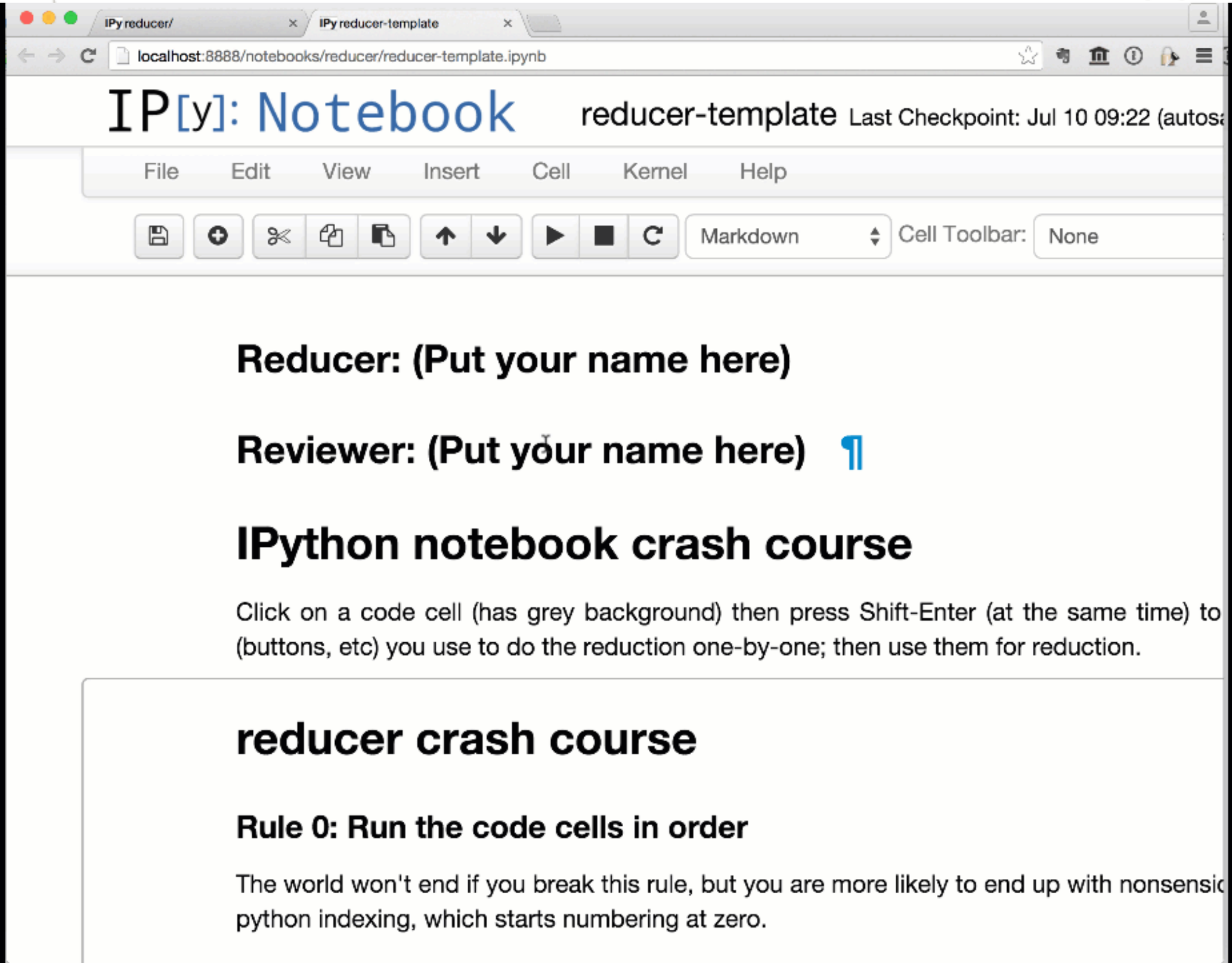
- HUGE THANKS to Steve Crawford and all of the new contributors:
- Christoph Deil
- Forrest Gasdia
- Carlos Gomez
- Hans Moritz Günther
- Nathan Heidt
- Anthony Horton
- Jennifer Karr
- Stefan Nelson
- Joe Philip Ninan
- Punyaslok Pattnaik
- Evert Rol
- William Schoenell
- Michael Seifert
- Sourav Singh
- Brigitta Sipocz
- Connor Stotts
- Ole Streicher
- Erik Tollerud
- Nathan Walker

```
with undergraduates:
```

- reducer: jupyter notebook with interactive widgets

with undergraduates:

MINNESOTA STATE UNIVERSITY MOORHEAD



`with` undergraduates:

MINNESOTA STATE UNIVERSITY
MOORHEAD

Photometry, currently

- Use AstroImageJ to:
  - Choose sources
    - Click on them once
    - Save as list
  - Perform aperture photometry with local background subtraction
    - Reject outlying pixels in annulus
- Result
  - instrumental magnitudes for night of data
- AIJ: http://arxiv.org/abs/1601.02622

Progress in python

MINNESOTA STATE UNIVERSITY MOORHEAD

Chrome File Edit View History Bookmarks People Window Help 49% 5:02

localhost:8888/notebooks/photometer.ipynb#

jupyter photometer Last Checkpoint: an hour ago (unsaved changes)

File Edit View Insert Cell Kernel Help Python 2

Code CellToolbar

```
                                description='Perform aperture photometry with',
                            value=None)
```

In [ ]:

```
fig, ax = plt.subplots(1, 1)

ccd = CCDData.read('M52-002R.fit')

ax.imshow(scale_and_downsample(ccd.data, downsample=1), cmap='gray', origin='lower')

source_list = None

def call_nathans_func(change):
    global source_list
    # print(change)
    to_call = change['new']
    if to_call is None:
        return
    # print("Calling it....")
    source_list = to_call(ccd, threshold=10, fwhm=8)
    ax.scatter(source_list['xcentroid'], source_list['ycentroid'], c='none', s=100, edgecolor='cyan')

def call_connors_func(change):
    to_call = change['new']
    if to_call is None:
        print("Not implemented")
        return
    foo = photutils_stellar_photometry(ccd, source_list, 5.0, 20.0)
    title = "Photometry done for {} sources".format(len(foo))
    ax.set_title(title)
    print(foo)

sources.observe(call_nathans_func, names='value')
photometry.observe(call_connors_func, names='value')
container.children = [sources, photometry]
container
```

Source detection photutils (@walkerna22)

Perform aperture photometry with photutils (@stottsco)

Ensemble differential photometry

- Magnitude differences to eliminate atmospheric effects
- Several comparison stars
  - Which stars should be used as comparisons?
  - How should they be combined?
- **HELP**: which approach is "best"? Is there a best?
  - Everett & Howell 2001
  - Broeg, Fernández & Neuhäuser 2005
    - lemon (on github)
  - Collins, Keilkopf & Stassun 2016
  - OSCAAR (somewhat like Broeg et al)(on github)
  - AAVSO

Solution for now

- Trial and error…
- … and gatspy
  - Insanely fast Lomb-Scargle periodogram

Solution for now

MINNESOTA STATE UNIVERSITY MOORHEAD

Chrome File Edit View History Bookmarks People Window Help

localhost:8889/notebooks/comp-star-plots-best.ipynb#

jupyter comp-star-plots-best Last Checkpoint: 02/25/2016 (unsaved changes)

File Edit View Insert Cell Kernel Help Python 2

Code CellToolbar

```
def source_column(source_number):
    col_name = 'Source-Sky_C' + str(source_number)
    return col_name

def source_error(source_number):
    col_name = 'Source_Error_C' + str(source_number)
    return col_name

def source_peak(source_number):
    col_name = 'Peak_C' + str(source_number)
    return col_name
```

Get list of soures in the AIJ table

In [4]:
```
sources = uniformize_source_names(aij_raw)
```

Set EITHER the list of known variables or the list of comparison stars

Do **NOT** set both.

The one you do NOT want to use should be set to `None`

In [ ]:
```
# List of stars you think ARE variable
known_variables = None# [1, 5, 6, 13, 14, 15, 16, 19, 20]  # Change to list of stars in the field that you know vary, or None

# List of stars you think ARE NOT variable
keepers = [10, 11, 12] # Change to list of comparisons you want to use, or None

# Check that settings are sensible

if known_variables is None and keepers is None:
    raise ValueError('You must set either known_variables or keepers to a list')

if known_variables is not None and keepers is not None:
    raise ValueError('Only ONE of known_variables and keeps can be set. '
                     'The other should be set to None.')
```

Variable?

MINNESOTA STATE UNIVERSITY MOORHEAD

Chrome File Edit View History Bookmarks People Window Help 82% 5:43

localhost:8889/notebooks/gatspy.ipynb#

jupyter gatspy Last Checkpoint: 02/25/2016 (unsaved changes)

File Edit View Insert Cell Kernel Help Python 2

Code CellToolbar

In [1]:
```
from __future__ import print_function, division

from astropy.table import Table

from gatspy.periodic import LombScargleFast, LombScargle

import matplotlib.pyplot as plt
import numpy as np

%matplotlib notebook

# use seaborn for plot styles
import seaborn; seaborn.set()
```

In [2]:
```
s = Table.read('kelt_13.csv')
t = s['BJD_TDB']
mag = s['Source-Sky_C13']
dmag = s['mag_error']
```

```
['BJD_TDB', 'Source-Sky_C13', 'mag_error']
```

In [3]:
```
good_mag = ~np.isnan(mag) & ~np.isinf(mag)#& (mag.data < 5.5) & (mag > 3.5)
good_mag &= ~(np.isnan(dmag) & np.isinf(dmag))
t = t[good_mag]
mag = mag[good_mag]
dmag = dmag[good_mag]
```

In [ ]:
```
model = LombScargleFast(Nterms=1).fit(t, mag, dmag)
periods, power = model.periodogram_auto(nyquist_factor=100, oversampling=5)
```

In [ ]:
```
fig, ax = plt.subplots()
ax.plot(periods, power)
ax.set(xlim=(0, 10), xlabel='period (days)',
       ylabel='Lomb-Scargle Power');
```

In [ ]:
```
# get range and find period
```

Image viewer

MINNESOTA STATE UNIVERSITY MOORHEAD

- Image viewer with catalog overlay
  - `astroquery` to get catalog from Vizer
  - `ccdproc`, `astropy.wcs` and `astropy.coordinates` to handle coordinates.
  - mpld3 to display
    - transition to `bqplot`?
    - `%matplotlib notebook`?

Image viewer

MINNESOTA STATE UNIVERSITY MOORHEAD

Chrome File Edit View History Bookmarks People Window Help 86% 6:24

localhost:8888/notebooks/image-viewer.ipynb#

jupyter image-viewer Last Checkpoint: Last Tuesday at 3:10 PM (unsaved changes)

File Edit View Insert Cell Kernel Help Python 2

Code CellToolbar

In [ ]:
```
from __future__ import print_function, division

import numpy as np

from astroquery.vizier import Vizier

from astropy.table import Table
from astropy.visualization import scale_image
from astropy.nddata import block_replicate
from astropy.coordinates import SkyCoord
import astropy.units as u

from ccdproc import CCDData

import matplotlib.pyplot as plt
%matplotlib inline
```

Read image

In [ ]:
```
ccd = CCDData.read('tres-3b-201R.fit', unit='adu')
print(ccd.wcs)
print(ccd.shape)
```

In [ ]:
```
# This function really should be somewhere else eventually.
def scale_and_downsample(data, downsample=4,
                         min_percent=20,
                         max_percent=99.5):

    scaled_data = scale_image(data,
                              min_percent=min_percent,
                              max_percent=max_percent)

    if downsample > 1:
        scaled_data = block_reduce(scaled_data,
                                   block_size=(downsample, downsample))
```

This week

- differential photometry
  - Which approach to comparing?
  - Selecting comparison set
  - idea: use APASS to transform to standard magnitude system frame-by-frame?
- reducer updates (help welcome)
- start turning these demos into a package
  - sneak peak on github, repo is:
    - glowing-waffle/glowing-waffles

Links

- lemon: http://lemon.readthedocs.org/en/latest/
  - end-to-end data reduction and photometry
- OSCAAR: http://oscaar.github.io/OSCAAR/
  - Focuses on exoplanet transit measurements
- gatspy: http://www.astroml.org/gatspy/
  - fast Lomb-Scargle implementation
- conda-build-all: https://github.com/SciTools/conda-build-all
  - eases the pain of building packages
- sep: http://sep.readthedocs.org/en/v0.5.x/
  - Photometry (uses internals from SExtractor)
- astroquery: http://astroquery.readthedocs.org/
  - Search a variety of online data sources from python.
- ginga: https://ejeschke.github.io/ginga/
  - Image viewer framework (and a reference viewer)
- ccdproc: http://ccdproc.readthedocs.org/en/latest/
  - Data reduction
- photutils: https://photutils.readthedocs.org/en/latest/
  - Photometry (including, but not limited to, IRAF-equivalents)
- AstroImageJ: http://www.astro.louisville.edu/software/astroimagej/
  - Very nice graphical interface with sophisticated fitting and graphing
- reducer: http://reducer.readthedocs.org/en/latest/
  - Widget-interface to ccdproc reduction
- glowing-waffles: https://github.com/glowing-waffle/glowing-waffles
  - Very much work-in-progress, examples from today will be up there by Tue, 3/22/16
- feder_image_shuffle: https://github.com/mwcraig/feder_image_shuffle
  - Among other things, makes jpeg images and gallery pages, also demonstrates interacting with Github API.
- msumastro: https://github.com/mwcraig/msumastro
  - Infrastructure for adding metadata (largely telescope specific)