AGA0414 Data Reduction

Prof. Alessandro Ederoclite

No imager is perfect

- Background noise
- Differences in illumination of the field of view
- "Sky concentration"
- Fringes
- Cosmic rays

(on the right a raw image from HST/WFPC2)

Every time you touch data, you affect them -> noise increases!

Data reduction is painful and everybody makes mistakes.

"Standard" data reduction is a myth.



I will now guide you through a full reduction of an image (the IRAF way)

If you have the VM, ds9 and IRAF in the xgterm should open easily.

If you are not running the VM (but have ds9 and IRAF installed on your computer remember):

- Launch ds9 first
- Go to the directory where your login.cl is (you probably have run mkiraf there)
- Open the xgterm; I do xgterm -sb &
- Make your xgterm big (but not too big)
- Start IRAF typing cl

My screen looks like this



login.cl

This is the configuration file of IRAF

A couple of important tips

- stdimage is the maximum size of image that IRAF will handle, my setting is:

set stdimage = imt4096

- To make sure you can use fits:

set imtype = "fits"

set imextn = "oif:imh fxf:fits,fit fxb:fxb plf:pl qpf:qp stf:hhh,??h"

Before you start, check your data

Always important: go to the directory where your data are and check them out:

ecl> pwd /Users/alessandroederoclite/iraf ecl> cd ../Desktop/USP/Lectures/UndergraduateCourses/lectures_2020A/12_datareduction/work/

A good thing is to get a list of the content of your directory

ecl> ls bias1.fits bias2.fits bias3.fits flat_R_1.fits flat_R_2.fits flat_R_3.fits sci2_R.fits



Let's do some statistics!

Let's check the properties of the images:

ecl≻	imstat *fits					
#	IMAGE	NPIX	MEAN	STDDEV	MIN	MAX
	bias1.fits	1060900	219.6	11,23	0.	3190.
	bias2.fits	1060900	222.6	13.06	0.	5094.
	bias3.fits	1060900	223.5	9,46	0.	660.
	flat_R_1.fits	1060900	10315.	1488.	0.	30248.
	flat_R_2.fits	1060900	37472.	5403.	0.	65535.
	flat_R_3.fits	1060900	37207.	5365.	0.	65535.
	sci2_R.fits	1060900	230,2	23.8	0.	6920.

Imstat is one of my favourite commands! It's very powerful but...

...with great power, comes great responsibility

We can check the parameters that a command accepts with lpar

If you want to modify the ones in parenthesis, you will have to do it ecl> lpar imstat explicitly images = "*fits" List of input images We will see how (fields = "image,npix,mean,stddev,min,max") Fields to be printed (1ower = INDEF)Lower limit for pixel values (upper = INDEF) Upper limit for pixel values (nclip = 0)Number of clipping iterations (lsigma = 3.) Lower side clipping factor in sigma (usigma = 3.) Upper side clipping factor in sigma (binwidth = 0.1)Bin width of histogram in sigma (format = yes)Format output and print column labels ? (cache = no) Cache image in memory ? (mode = "ql")

Help!

help imstat

Will give you a more complete insight on the command

(needless to say, this applies to <u>any</u> IRAF command)

```
X xaterm
IMSTATISTICS (Feb01)
                              images.imutil
                                                   IMSTATISTICS (Feb01)
NAME
    imstatistics -- compute and print image pixel statistics
USAGE
    imstatistics images
PARAMETERS
    images
        The input images or image sections for which pixel statistics
        are to be computed.
    fields = "image,npix,mean,stddev,min,max"
        The statistical quantities to be computed and printed.
    lower = INDEF
        The minimum good data limit. All pixels are above the default
        value of INDEF.
    upper = INDEF
        The maximum good data limit. All pixels are above the default
        value of INDEF.
    nclip = 0
        The maximum number of iterative clipping cycles, By default no
        clipping is performed.
    lsigma = 3.0
        The low side clipping factor in sigma.
    usigma = 3.0
        The high side clipping factor in sigma.
    binwidth = 0.1
        The width of the histogram bins used for computing the midpoint
        (estimate of the median) and the mode. The units are in sigma.
    format = yes
        Label the output columns and print the result in fixed format.
        If format is "no" no column labels are printed and the output
        is in free format.
    cache = no
        Cache the image data in memory ? This can increase the
        efficiency of the task if nclip > 0 or either of the midpt and
        mode statistics are computed.
  g=guit,d=downhalf,flsp=downfull,jlcr=downline,N=next]
```

Last on imstat

I hate that "NPIX" and I think mode and median are useful. Here's something I find more useful:

ecl≻	imstat *fits fields	s="image,m	ean,mode,m	idpt,stdde [,]	v,min,max"		
#	IMAGE	MEAN	MODE	MIDPT	STDDEV	MIN	MAX
	bias1.fits	219.6	219,5	219,5	11,23	0.	3190.
	bias2.fits	222,6	222,6	222.7	13.06	0.	5094.
	bias3.fits	223.5	222.	223,3	9,46	0.	660.
	flat_R_1.fits	10315.	10517.	10516.	1488.	0.	30248.
	flat_R_2.fits	37472.	38171.	38182.	5403.	0.	65535.
	flat_R_3.fits	37207.	37945.	37959.	5365.	0.	65535.
	_ sci2_R.fits	230,2	231.9	230.4	23.8	0.	6920.

display

To display an image in ds9 from IRAF, we use the command "display"

Ds9 has several "frames" we can use (we can show several images at the same time)

I also like to "fill" the frame (so I see the full image)

ecl> lpar display image = "bias1.fits" image to be displayed frame = 1frame to be written into (bpmask = "BPM")bad pixel mask (bpdisplay = "none") bad pixel display (noneloverlay/interpolate) (bpcolors = "red") bad pixel colors (overlay = "") overlay mask (ocolors = "green") overlay colors (erase = yes) erase frame erase unfilled area of window (border_erase = no) (select frame = yes) display frame being loaded (repeat = no)repeat previous display parameters (fill = no)scale image to fit display window (zscale = ues)display range of greylevels near median contrast adjustment for zscale algorithm (contrast = 0.25)(zrange = yes) display full image intensity range (zmask = "") sample mask (nsample = 1000)maximum number of sample pixels to use (xcenter = 0.5)display window horizontal center display window vertical center (ucenter = 0.5)(xsize = 1.)display window horizontal size (usize = 1.) display window vertical size (xmag = 1.)display window horizontal magnification (umag = 1.)display window vertical magnification (order = 0)spatial interpolator order (0=replicate, 1=linear) (z1 =)minimum greylevel to be displayed $(z^2 =)$ maximum greylevel to be displayed (ztrans = "linear") greylevel transformation (linear/log/none/user) (lutfile = "") file containing user defined look up table (mode = "ql")

display

The "complete" way to use display is:

ecl> display bias1,fits 1 fill=yes z1=184, z2=246,

The numbers underneath are the minimum and maximum which are shown in the image

Friends invoke display like:

ecl> displ bias1.fits 1 fi+ z1=18<u>4</u>. z2=246.

"=yes" can be replaced by "+" You can replace complete names with abbreviations if it's unambiguous

•					X	SAOIma	ige ds9				
File	Edit	View	Frame	Bin	Zoom	Scale	Color	Region	WCS	Analysis	Help
File Obje	ct		bias1.fits BIAS	;	1						
value	3										
Phys	ical	×			y						
Fram	e 1	×	0.114	423) y	0	•				
file	edit	view	frame	bin	zoom	scale	color	regior	n wes	s analys	is help
zoor	n in	zoom	out z	oom fit	Z00	m 1/4	zoom 1.	/2 zoo	om 1	zoom 2	zoom 4
	190	10	16 2	03	209	215	221	25	7	234 3	,
	130	18	,o 2	00	203	210	22	. 22	. /	204 0	.40

First time, it is a good practice to look at everything!

ecl> displ flat_R_1.fits 1 fi+ z1=97<u>9</u>2.322 z2=10956.

00	0.0				X	SAOIma	age ds9				
File	Edit	View	Frame		Zoom	Scale	Color	Region	WCS	Analysis	Help
File			flat_R_1	.fits						MONEMERS	1000
Obje	ct		SKY,FL	AT							16
Valu	е		1072	2.1							10000
WCS											1
Phys	ical	×	484.	555	У	977.664	4			1.1	A COLORED
Imag	8	×	1926	i.44	У	3887.4				16.0	
Fram	e 1	×	0.11	423		0	•			No. Contraction	P. (56:8842)
file	edit	view	frame	bin	zoom	scale	color	r regior	1 WC	s analys	is help
200	m in	zoom	out z	oom fit	Z00	m 1/4	zoom 1	/2 zoo	om 1	zoom 2	zoom 4
	9908	100)24 10	142	10258	1037	5 104	91 106	507 ·	10724 1	0840

ecl> displ sci2_R.fits 1 fi+ z1=203. z2=260.

File Edit View Frame Bin Zoom Scale Color Region WCS Analysis Help File Scale_R fils 0						X	SAOIma	ige ds9				
File Cbject Cbje	File	Edit	View	Fram	e Bin	Zoom	Scale	Color	Region	wcs	Analysis	Help
Object 07 Physical x Image x Frame 1 x 11423 0 file edit 1200 out 200m 0ut 200 out 200m 1/2	File			sci2_R	.fits							
Value Physical x y 0 File edit view frame bin coom scale color region wcs analysis heip zoom in zoom out zoom fit zoom 1/4 zoom 1/2 zoom 1 zoom 2 zoom 4	Objei	ct		07								
Physical mage * y 0 * Ife etit view frame 1 x 000	Value	9										
Image Frame 1 X 0.11423 0 0 Ile edit view frame bin zoom scale color region wcs analysis help zoon in zoom out zoom fit zoom 1/4 zoom 1/2 zoom 1 zoom 2 zoom 4 0 1	Phys	ical	×					_				
Frame 1 x 0.11423 0 * file edit view frame bin zoom colm cel colm vcs analysis help zoom in zoom out zoom file zoom 1/2 zoom 1/2 zoom 1 zoom 2 zoom 4	Image	3	×			ý		-				
11e edit view frame bin zoom scale color region vics analysis help zoon in zoom out zoom fit zoom fit zoom 1/2 zoom 1/2 zoom 2 zoom 4	Fram	e 1	×	0.1	1423		0	۰				
2000 ini 2000 fti	file	edit	view	frame	e bin	zoom	scale	color	region	WCS	s analys	is help
	zoor	n in	zoom	out	zoom fit	2001	m 1/4	zoom 1	/2 Z001	n 1	zoom 2	zoom 4
209 214 220 226 232 237 243 249 254												
		209	21	4	220	226	232	23	7 243	3	249 2	254

Background Noise

The "bias"

What happens if you take a "0 seconds exposure"?

Wait a second! What do you mean by "0 seconds"?

A "bias" is the readout of a CCD without collecting photons.

000				XS	SAOIma	ge ds9				
File Edi	t View	Frame	Bin	Zoom	Scale	Color F	Region	WCS	Analysis	Help
File Object Value		bias1.fits BIAS								
Physical Image Frame 1	× × ×	0.114	23	у У	0	。				
file edi	view	frame	bin	zoom	scale	color	region	wcs	analysi	s help
zoom in	zoom	out zo	om fit	zoom	1/4	zoom 1/2	zoon	n 1	zoom 2	zoom 4
190	19	96 20)3	209	215	221	227		234 2	40

The "overscan"

A region "on the side" of the image, where "empty" readouts of the electronics are made.

(these are cuts along lines)



File Edit Vie File Object	w Frame E	Bin Zoom Sc	ale Color	Region			lu.		
File Edit Vie File Object	W Frame E	sin Zoom Sc	ale Color	Region	WCC A	train lla	the second s		
File Object Volue	000176046			negion	WCS AI	nalysis He	ip		
Object Velue	000170040	0-20180903-OS	IRIS-OsirisL	ongSlitSp	ectro				
Value	1					Y			
value	8051.32								
WCS ×		Y					×	•	
Physical >	288.455	Y 1520.8	191						
image × Fromo 1 ×	0.220	Y 3036.0	102						
	0.220								l la e la
file edit	VIEW TI	rame bin	zoom	scale	color	region	wcs an:	alysis	neip
zoom in	zoom out	zoom fit	zoom 1/4	ZC	oom 1/2	zoom 1	zoom 2	2	zoom 4

The "overscan"



Bias vs Overscan

- Bias takes into account the 2D variations of noise
- Overscan measures the noise of your image

Not all CCDs have overscan (or prescan) regions.

Bias vs Overscan

	Pros	Cons
Bias	2D mapping of the noise of the CCD	Not simultaneous with observations
Overscan	Obtained together with observations	It is a 1D approximation to the noise of the CCD. Not all CCDs allow for an overscan region.

Make a "masterbias"

To get a good measure of the bias, we take several bias frames and we combine them.

For this, we need to load the imred package (if you see a ".", it's a sub-package; if you see an "@", it's a parameter file; if you see nothing else, it's a command)

ecl> r	noao artdata. astcat. astrometry.	astutil. digiphot. focas.	imre mtlo nobs	ed. ocal. solete.	nproto. observa obsuti]	atory l.	onedspec rv. surfphot).	twodspec.	
noao>	imred argus. bias. ccdred.	crutil. ctioslit. dtoi.	eche gene hydr	elle. eric. `a.	iids. irred. irs.		kpnocoud kpnoslit quadred.	de. :.	specred. vtel.	
imred)	> ccdred badpiximage ccdgroups ccdhedit	ccdinstrum ccdlist ccdmask	ent	ccdproc ccdtest. combine	c f n	darkcon Clatcon nkfring	nbine nbine gecor	mkill mkill mksky	lumcor lumflat jcor	mkskyflat setinstrument zerocombine

We want to combine bias frames

In IRAF, a bias is called "zero" (guess why?)

Hence we want to use the zerocombine command.

Let's get to know it:

ccdred> lpar zerocombine input = "bias*fits" List of zero level images to combine (output = "Zero") Output zero level name (combine = "average") Type of combine operation (reject = "minmax") Tupe of rejection (ccdtupe = "zero") CCD image type to combine (process = no) Process images before combining? (delete = no)Delete input images after combining? (clobber = no)Clobber existing output image? (scale = "none") Image scaling (statsec = "") Image section for computing statistics (nlow = 0)minmax: Number of low pixels to reject (nhigh = 1)minmax: Number of high pixels to reject (nkeep = 1)Minimum to keep (pos) or maximum to reject (neg) (molip = yes) Use median in sigma clipping algorithms? $(lsiqma = \overline{3}.)$ Lower sigma clipping factor $(hsigma = 3_{\star})$ Upper sigma clipping factor (rdnoise = "0.") ccdclip: CCD readout noise (electrons) (gain = "1.") ccdclip: CCD gain (electrons/DN) (snoise = "0.") ccdclip: Sensitivity noise (fraction) (pclip = -0.5)polip: Percentile clipping parameter (blank = 0.)Value if there are no pixels (mode = "ql")

Let's run zerocombine

ccdred> zerocombine bias*fits output=masterbias.fits ccdtype="" ccdred>

(the ccdtype = "" is a sad story)

It looks pretty dull, eh?

Let's see what it did:

ccdred≻	imstat *ias* fie	elds="image	e,mean,mode	e,midpt,st	ddev,min,ma	κ"	
#	IMAGE	MEAN	MODE	MIDPT	STDDEV	MIN	MAX
	bias1.fits	219.6	219,5	219.5	11,23	0.	3190.
	bias2.fits	222.6	222.6	222.7	13.06	0.	5094.
	bias3.fits	223.5	222.	223.3	9,46	0.	660.
ma	asterbias.fits	217.8	218.4	217.7	6,196	0.	247.

The flat field

It is related with the difference of response across the field of view.

Famous example: dust particles on filters.

Corrected by observing a "flat field" (hence the name): sky vs. dome.

It is a multiplicative effect.

Mind the division by zero!

(happens often at the borders)

							X SA	Olmage	ds9				
File Edit	View	Frame	Bin	Zo	oom S	cale	Color	Region	WCS	Analysis	Help		
File		j02-FLA	S-b20	151	101e12	08-rS	DSS-0	0-C01M5	V2.fit				
Object		SKYFLA	١T										
Value		1.029	79										
fk5	α	0:02:05	.639	5	+27:35	:35.6	Э						
Physical	Х	5634.	889	Y	4490	3.893							
Image	Х	5634.	889	Y	4493	3.893							
Frame 3	×	0.04	9		0	.000	•			Sector 10	and a state of the		
file e	dit	view	fram	9	bin	ZO	om	scale	color	regio	n wcs	analysis	help
open		s	ave			head	er		page se	tup	print		exit



Make a masterflat

Check the flats:

ccdred≻	imstat *lat* f	ields="imag	e,mean,mod	e,midpt,st	ddev,min,ma	ax"	
#	IMAGE	MEAN	MODE	MIDPT	STIDDEV	MIN	MAX
	flat_R_1.fits	10315.	10517.	10516.	1488.	0.	30248.
	flat_R_2.fits	37472.	38171.	38182.	5403.	0.	65535.
	<u>f</u> lat_R_3.fits	37207.	37945.	37959.	5365.	0.	65535.

You normally want a flat to have counts of about the half well (~35,000); in this case, only 2 fulfill this criteria. Let's make a list (the "@" lets you access a list of files.

ccdred>	ls flat_R_2,fits	s flat_R_3	.fits > fl	ats.lis			
ccdred>	imstat @flats.li	s fields=	"image,mea	n,mode,mid	pt,stddev,m	nin,max"	
#	IMAGE	MEAN	MODE	MIDPT	STDDEV	MIN	MAX
	flat_R_2.fits	37472.	38171.	38182.	5403.	0.	65535.
	flat_R_3.fits	37207.	37945.	37959.	5365.	0.	65535.

Subtract the bias

Mind you: a flat has a bias

You want to subtract it before combining the flats.

We use the almighty ccdproc

(the longer the parameter list, the more powerful the task)

ccdred> lpar ccdproc	
images = "sci2_R.fits"	List of CCD images to correct
(output = "")	List of output CCD images
(ccdtype = "")	CCD image type to correct
(max_cache = 0)	Maximum image caching memory (in Mbytes)
(noproc = no)	List processing steps only?\n
(fixpix = no)	Fix bad CCD lines and columns?
(overscan = no)	Apply overscan strip correction?
(trim = no)	Trim the image?
(zerocor = no)	Apply zero level correction?
(darkcor = no)	Apply dark count correction?
(flatcor = no)	Apply flat field correction?
(illumcor = no)	Apply illumination correction?
(fringecor = no)	Apply fringe correction?
(readcor = no)	Convert zero level image to readout correction
(scancor = no)	Convert flat field image to scan correction?\n
(readaxis = "line")	Read out axis (column line)
(fixfile = "")	File describing the bad lines and columns
(biassec = "")	Overscan strip image section
(trimsec = "")	Trim data section
(zero = "")	Zero level calibration image
(dark = "")	Dark count calibration image
(flat = "")	Flat field images
(illum = "")	Illumination correction images
(fringe = "")	Fringe correction images
(minreplace = 1.)	Minimum flat field value
(scantype = "shortscan")	Scan type (shortscanllongscan)
(nscan = 1)	Number of short scan lines\n
(interactive = no)	Fit overscan interactively?
(function = "legendre")	Fitting function
(order = 1)	Number of polynomial terms or spline pieces
(sample = "*")	Sample points to fit
(naverage = 1)	Number of sample points to combine
(niterate = 1)	Number of rejection iterations
$(10W_reject = 5,)$	Low sigma rejection factor
$(n_1gn_reject = 3,)$	High sigma rejection factor
(grow = V.) (and = "-1")	Rejection growing radius
l lmode = "dl")	

ccdred> <u>c</u>cdproc @flats.lis output=b_//@flats.lis zerocor+ zero=masterbias.fits

The "//" allows you to add some text before the name of the files in the list.

Let's check what happened:

ccdred	> imstat *]	lat* fiel	ds="image,	 mean,mode,	midpt,stdd	ev.min.ma×		
#		IMAGE	MEAN	MODE	MIDPT	STDDEV	MIN	Max
	b_flat_R_2.	fits	37255.	37926.	37933.	5403.	-229,5	65364.
	b_flat_R_3.	fits	36989.	37695.	37708.	5366.	-229,5	65364.
	flat_R_1.	fits	10315.	10517.	10516.	1488.	0.	30248.
	flat_R_2.	fits	37472.	38171.	38182.	5403.	0.	65535.
10.01	flat_R_3.	fits	37207.	37945.	37959.	5365.	0.	65535.
Error	re <u>a</u> ding ima	age flats	.lis					

Let's combine flats

You can probably guess, by now, that we will need a command called flatcombine

Here are the parameters ->

ccdred> lpar flatcombine	
input = "h //@flats.lis"	"list of flat field images to combine
(output = "Flat")	Output flat field root name
(combine = "average")	Type of combine operation
(reject = "avsigclip")	Type of rejection
(ccdtype = "")	CCD image type to combine
(process = no)	Process images before combining?
(subsets = no)	Combine images by subset parameter?
(delete = no)	Delete input images after combining?
(clobber = no)	Clobber existing output image?
(scale = "mode")	Image scaling
(statsec = "")	Image section for computing statistics
(nlow = 1)	minmax: Number of low pixels to reject
(nhigh = 1)	minmax: Number of high pixels to reject
(nkeep = 1)	Minimum to keep (pos) or maximum to reject (neg)
(mclip = yes)	Use median in sigma clipping algorithms?
(lsigma = 3.)	Lower sigma clipping factor
(hsigma = 3₊)	Upper sigma clipping factor
(rdnoise = "0.")	ccdclip: CCD readout noise (electrons)
(gain = "1.")	ccdclip: CCD gain (electrons/DN)
(snoise = "0,")	ccdclip: Sensitivity noise (fraction)
(pclip = -0,5)	pclip: Percentile clipping parameter
(blank = 1,)	Value if there are no pixels
(mode = "ql")	

Let's run it!

ccdred> <u>f</u>latcombine b_//@flats.lis output=masterflat_R.fits

As usual... let's check

ccdred> imstat	red> imstat *lat* fields="image,mean,mode,midpt,stddev,min,max"										
#	IMAGE	MEAN	MODE	MIDPT	STDDEV	MIN	MAX				
b_flat_	R_2.fits	37255.	37926.	37933.	5403.	-229,5	65364.				
b_flat_l	R_3.fits	36989.	37695.	37708.	5366.	-229,5	65364.				
flat_l	R_1.fits	10315.	10517.	10516.	1488.	0.	30248.				
flat_l	R_2.fits	37472.	38171.	38182.	5403.	0.	65535.				
flat_l	R_3.fits	37207.	37945.	37959.	5365.	0.	65535.				
Error reading	image flat	s.lis									
masterfla	t_R.fits	37122.	37710.	37688.	5383.	-229,5	65365.				

Of course, flats.lis is not an image, hence you get an error

We have few flats, so we don't get a very very good flat.

Reduce the science frame!

We are finally there!

Let's check our science frame (it's actually the field of a photometric standard star; we will see what these are soon)

ccdred>	imstat sci2_R.fit:	s fields='	"image,mea	n,mode,miu	dpt,stddev,i	min,max"	
#	IMAGE	MEAN	MÕDE	MIDPT	STDDEV	MIN	MAX
	sci2_R.fits	230,2	231,9	230.4	23.8	0.	6920,
ccdred>	displ sci2_R.fits	1 fi+					
z1=203.	z2=260.						

We will use our old friend ccdproc

•		X SAOImage ds9											
File	Edit	View	Fram	e Bin	Zoon	n Scale	Colo	r Re	gion	WCS	Analysis	Hel	р
File Object Value WCS Physi Image Frame	ct cal e e 1	××××	sci2_R 07 234 975 387 0.1	.fits 1.794 1.466 78.64 1423]] y] y	y 508.767 y 2022.74 0 •							の方法であると
file	edit	view	frame	e bin	zooi	m scal	e col	or	region	wc	s analys	sis	help
zoon	1 in	zoom	out	zoom fit	zo	om 1/4	zoom	1/2	Z00	m 1	zoom 2	200	m 4
				1	,	,		-			,		
	209	21	4	220	226	23	2 2	237	24	3	249	254	

ccdred> ccdproc sci2_R.fits output=r_sci2_R.fits zerocor+ flatcor+ \ >>> zero=masterbias.fits flat=masterflat_R.fits ERROR: f<u>l</u>oating point invalid operation

What do you mean "error"? Do you know who I am?

2 things:

- 1) The "\" is used to go to a new line
- 2) Remember I said that you may get a zero at the edges of the image? It is exactly what happened! We need to "trim" the image.

First delete the reduced image that IRAF tried to create.

ccdred> del r_sci2_R.fits ccdred> ccdproc sci2_R.fits output=r_sci2_R.fits zerocor+ flatcor+ \ zero=masterbias.fits flat=masterflat_R.fits trim+ trimsec=[100:100.100:1000] Then do the right thing and say that you want to trim the image and define the "trim section":

[x of lower left corner : x lower right corner , y upper left corner : y upper right corner]

Let's see our masterpiece

ccdred> imstat *.fits	fields="ima	ge,mean,mo	de,midpt,s	tddev,min,	max"	
# IMAGE	MEAN	MODE	MIDPT	STDDEV	MIN	MA
b_flat_R_2.fits	37255.	37926.	37933.	5403.	-229,5	65364
b_flat_R_3.fits	36989.	37695.	37708.	5366.	-229,5	65364
bias1.fits	219,6	219,5	219,5	11,23	0.	3190
bias2.fits	222,6	222,6	222.7	13.06	0.	5094
bias3.fits	223,5	222.	223.3	9,46	0.	660
flat_R_1.fits	10315.	10517.	10516.	1488.	0.	30248
flat_R_2.fits	37472.	38171.	38182.	5403.	0.	65535
flat_R_3.fits	37207.	37945.	37959.	5365.	0.	65535
masterbias.fits	218.3	217.9	218.	6,003	175.	244.
masterflat_R.fits	37910.	37931.	37927.	1059.	15881.	58180
r_sci2_R.fits	12,58	14,31	12,37	26,27	-81,83	6688
sci2_R.fits	230,2	231,9	230,4	23.8	0.	6920
ccdred> displ r_sci2_R	.fits 1 fi+					
z1=-32.14126 z2=46.335	46					

Congratulations! You have reduced your first image!

Some more steps

Fringing

This is due to the interaction of light with the coating of the CCD.

It only affects the reddest filters (and not always).

It may vary across the night.

It is (yet) another additive effect.

SAOImage ds9													
File Edit	View	Frame	Bin Z	oom Sc	ale Colo	r Rej	gion	WCS A	nalysis H	elp			
File		j02-FRIN	-b20151	203e021	8-zSDSS-0)0-C0	1V3.fi	its 🛛 📷		Strates.			196 S 184
Object		J-PLUS-0	02477										
Value		-1.190	86					52					
fk5	a	9:03:56.	564 8	+55:18	:14.98					E.S.		•	
Physical	Х	7707.1	84 Y	3324	.380				(r				
Image	Х	7707.1	84 Y	3324	.380								
Frame 3	×	0.049	9	0.	000 °					的复数距			
file ed	it	view	frame	bin	zoom	sca	ale	color	region	WCS	analys	is	help
zoom in	2	zoom out	ZO	om fit	zoom 1	/4	zo	oom 1/2	zoom	1	zoom 2	z	oom 4



Cosmic Rays

These are high energy particles which interact with our silicon.

It gets worse out of the atmosphere (e.g. HST).

They are normally removed by "averaging" several images.

There are some programs (e.g. <u>LACos</u>) which use statistics to get rid of them.



Bad pixels / columns

Some pixels do not respond linearly -> hot/cold pixels.

Some pixels actually block the movement of charge -> bad columns

Fix it by

- -) "dithering" the telescope
- -) align the images
- -) combine

It is also what the IRAF fixpix in ccdproc promises.

SAOImage ds9											
File Ed	it View	/ Frame	Bin Z	loom S	icale Colo	or Region	WCS A	Analysis	Help		
File		0001760	3460-201	80903-C	SIRIS-Osir	risLongSlitSp	ectri	<u>.</u>			
Object		J-CV-41	1								
Value								E -			
WCS	v										
Priysical	×		Y								
Frame 4	×	1.00	00	0.	000 0						
file	edit	view	frame	bin	zoom	scale	color	regio	n wos	analysi	s help
linear	log	pov	ver	sqrt	squared	asinh	sinh	hist	togram	min max	zscale
									L		
									1		
	1.1.1								1		
and the second											
	4999					1'					
Second Second						1		-			
						. Ing					
						1. A.	- 1) -			8 7 B	
						- I					
						1	l				
								4			
CHOICE BEESTING									10000000		
;	7598	8302	87	74	9119	9394	9617	98	07	9974 10)120

That's all for today

Can you do the same for the other filters?

OR

Can you do it in Python?

<u>Next time</u>: we will see how to measure the characteristics of a CCD (gain and readout noise) and we will learn to do some basic photometry (finally speaking about noise)