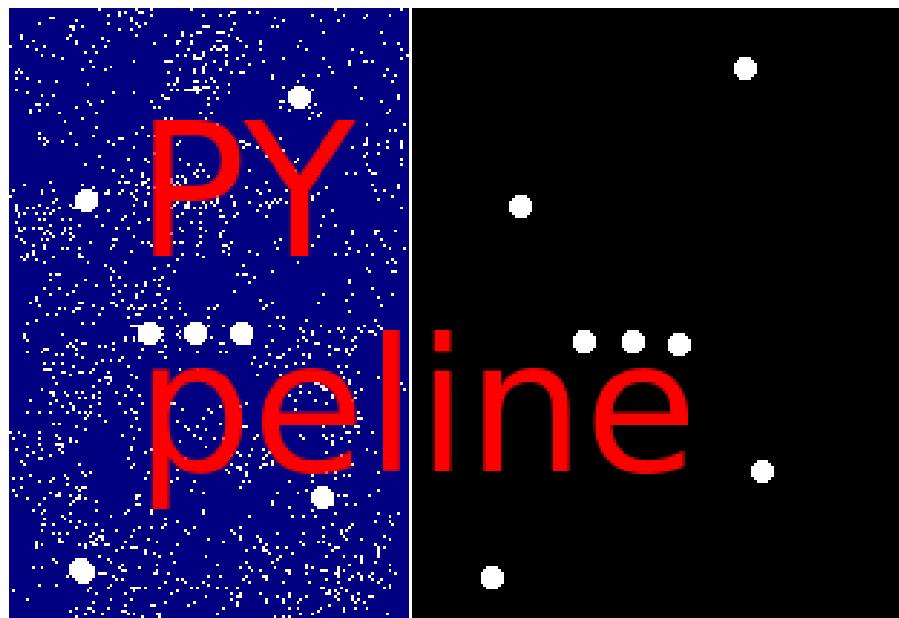


Manual



Autor: Matheus Bernini Peron

Email: matheus.b.peron@gmail.com

Link GitHub: <https://github.com/MathBern/PYpeline>

INTRODUÇÃO

A extração de informações científicas dependem de um bom processo de redução dos dados observacionais crus obtidos, atualmente, com os CCDs. Os elementos que separam a imagem crua e a informação científica buscada são os ruídos envolvidos na detecção dos fótons; são, eles geralmente:

- ruído devido ao processo de leitura do CCD
- ruído devido a emissão térmica dos próprios instrumentos
- ruído devido a sensibilidade diferenciada a radiação de cada (grupo de) pixel do CCD

Há diversas técnicas e métodos de implementá-las que permitem reduzir esses erros e extrair a ciência relevante. A técnica que será exposta e implementada, com certas modificações, pelo PYipeline será o Procedimento Padrão IRAF (Tody 1992).

Correção de Bias:

- 1 – Cria-se uma imagem, denominada masterbias, a partir da combinação pela mediana¹ das imagens de bias feitas pelo(a) observador(a).
- 2 – Subtrai-se o masterbias de todas as outras imagens (ciência, dark e flat)

Correção de Dark Current²:

- ! - É necessário exista antes o masterbias.
- 1 – Subtrai-se de cada imagem de dark o masterbias
- 2 – Cria-se uma imagem, denominada masterdark, com o dark mediano a partir das várias imagens de dark.
- 3 - Subtrai-se o masterdark das imagens de ciência e de flat

Correção e Normalização de Flat:

- !É necessário que exista masterbias e masterdark (este último se houver)
- 1 – Subtrai-se de cada imagem de flat o masterbias e o masterdark, este último se houver
- 2 – Obtém-se o valor médio de todos os pixels para uma imagem e divide-se a imagem por este valor obtido.
- 3 – Cria-se uma imagem mediana, masterflat, a partir das imagens de flat já normalizados

Correção de Imagens de Ciência³:

- !É necessário que exista masterbias, masterflat e masterdark
- 1 – Subtrai-se de cada imagem de ciência o masterbias e o masterdark, se houver.
- 2 – Multiplica-se cada imagem de ciência pelo masterflat
- A – Se necessário, combina-se as imagens de ciência para uma imagem principal.

Análise Fotométrica dos dados:

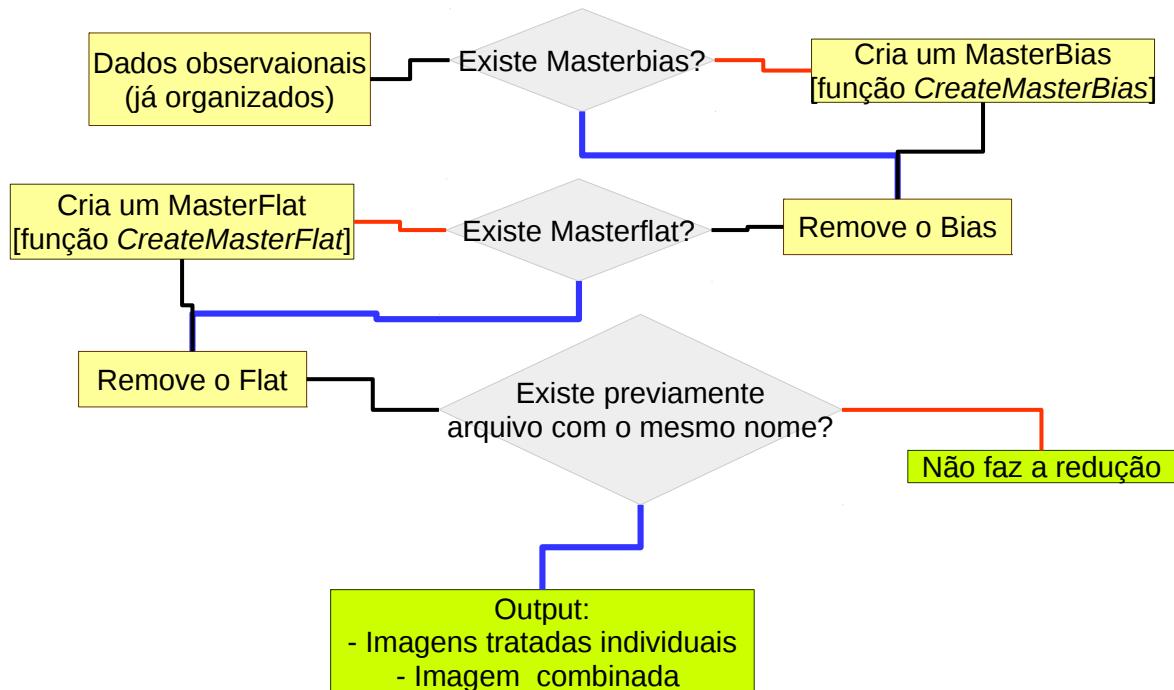
Essa parte está a critério do(a) cientista e depende do objetivo buscado. O PYipeline não faz esse trabalho em sua versão atual.

1 Através da mediana elimina-se a contaminação devido a raios cósmicos, que possuem altíssimo número de contagens e provocariam um aumento artificial do nível médio caso a combinação seja pela média.

2 Não incluído no Pypeline, já que atualmente, o ruído térmico não é significante.

3 Não incluído no Pypeline, já que o código se propõe apenas a fazer a redução primária.

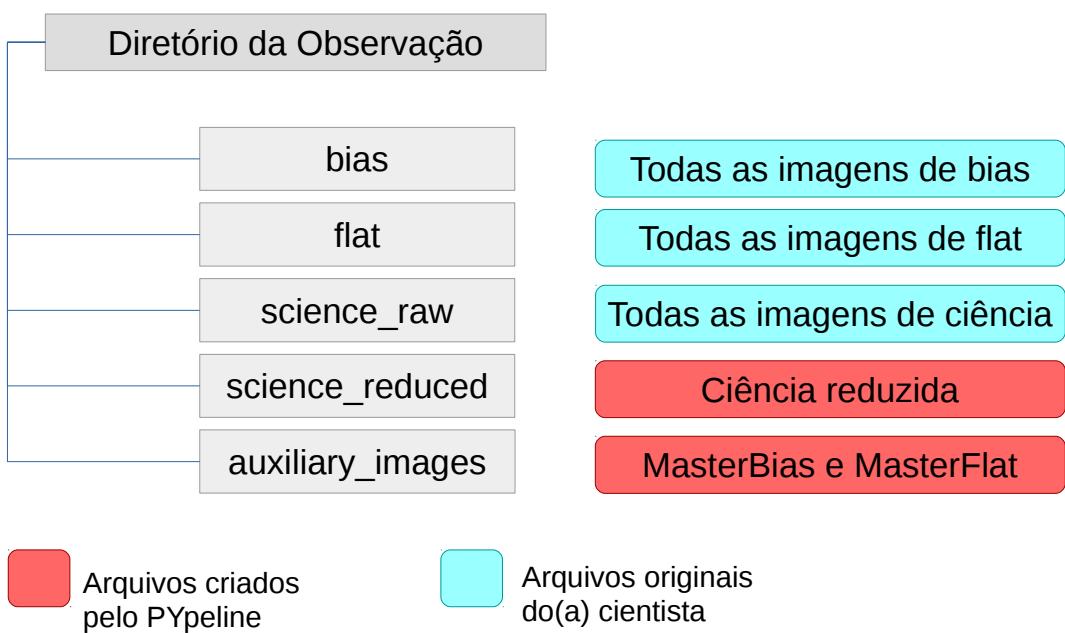
O PYpline tem por objetivo fazer a remoção do bias e do flat apenas através da sua função principal *ReduceCompletely*, seguindo o fluxograma abaixo.



DEPENDÊNCIAS E PRÉ-REQUISITOS

O PYpline é um código escrito em Python 3.6 (portanto, recomenda-se que o(a) usuário(a) utilize versões posteriores ao Python 3.5), e depende de alguns pacotes para funcionar corretamente (ver o Readme.md). Eles tem funções que tem por objetivo facilitar o trabalho numérico dos dados (Numpy, Numba), gerenciamento de arquivos (Os, Glob) e leitura dos arquivos FITS (Astropy).

Além das dependências de pacotes, o PYpline depende de uma organização de diretórios específica para funcionar corretamente (figura abaixo).



Cada diretório deve conter as respectivas imagens, de acordo com sua função. Assim não importa o nome dos arquivos, desde que estejam nas pastas corretas.

EXEMPLOS E VERIFICAÇÕES DA QUALIDADE DA REDUÇÃO

Obtendo os dados

Os dados utilizados nos exemplos e nos testes pertencem ao colaborador Walter S. Martins Filho e podem ser baixados através do link: <https://www.dropbox.com/s/4u5qmtv1b154tzz/xo2b.zip?dl=0> .

Os diretórios não estão organizados de acordo com a descrição da seção anterior, cabendo ao usuário organiza-los.

Exemplo

Na pasta *Examples*, há um arquivo demonstrando a utilização das funções.

Testes

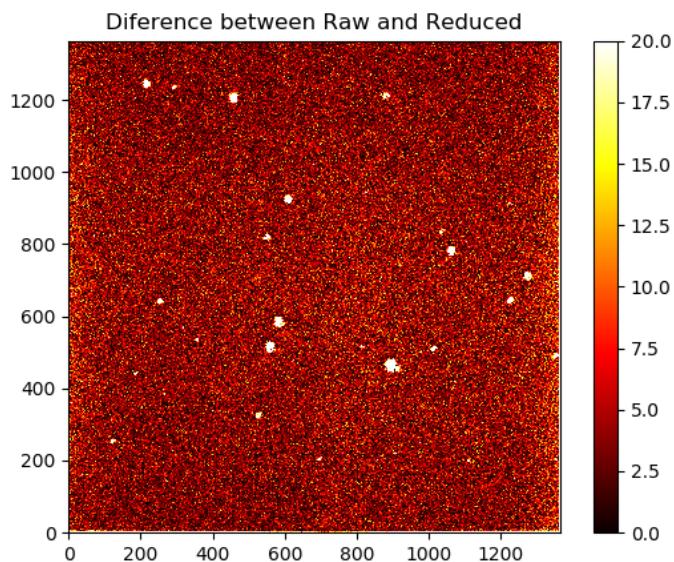
Antes de executar o arquivo de teste, faça primeiro o exemplo!

O PYpline tem embutido nas funções CreateMasterBias e CreateMasterFlat pequenos testes que permitem averiguar a qualidade das imagens de normalização. No entanto, outros testes que o(a) cientista queira fazer não estão incluídos no pacote, devendo ele(a) elaborar por si.

O teste do masterflat é feito fazendo uma média da imagem. Se o valor calculado estiver no intervalo (0.9, 1.1), considera-se que a função foi bem sucedida e o masterflat está próprio.

O teste do masterbias é igual ao do masterbias, tirando a média. Se esta for menor que 30, considera-se que o masterbias tem boa qualidade.

Na pasta *Tests*, existe um arquivo que realiza um teste da qualidade geral da redução mostrando a diferença de uma determinada imagem reduzida com sua progenitora crua (figura abaixo). Como é possível ver, existem regiões mais claras e mais escuras, revelando as diferenças entre as imagens cruas e as imagens tratadas.



CONSIDERAÇÕES FINAIS

O código não se propõe um ser um pipeline de ponta, porém já é capaz de fazer a redução básica para as necessidades científicas modernas. Em suas novas versões, mais funções serão

incorporadas, melhorando o código.