

OCFit - návod

Pavol Gajdoš

2019
verzia 0.1.3

Obsah

1	Úvod	4
2	Potrebné knižnice	4
3	Inštalácia	4
4	FitLinear a FitQuad	5
4.1	Inicializácia	5
4.2	Dostupné funkcie	5
4.2.1	FitLinear	5
4.2.2	FitQuad	6
4.2.3	FitRobust	6
4.2.4	FitMCMC	6
4.2.5	InfoMCMC	7
4.2.6	Epoch	8
4.2.7	PhaseCurve	8
4.2.8	CalcErr	9
4.2.9	CorrectErr	9
4.2.10	AddWeight	10
4.2.11	Summary	10
4.2.12	Plot	10
4.2.13	PlotRes	12
4.2.14	SaveOC	14
4.2.15	SaveRes	14
4.3	Ukážka	14
5	OCFit	16
5.1	Inicializácia	16
5.2	Nastavenie parametrov pre fitovanie	16
5.3	Práca s váhami bodov	17
5.4	Všeobecné poznámky k práci s triedou OCFit	17
5.5	Dostupné funkcie	18
5.5.1	AvailableModels	18
5.5.2	ModelParams	18
5.5.3	LiTE	18
5.5.4	AgolInPlanet	19
5.5.5	AgolInPlanetLin	20
5.5.6	AgolExPlanet	21
5.5.7	AgolExPlanetLin	22
5.5.8	LiTE3	23
5.5.9	LiTE3Quad	23
5.5.10	LiTE34	24
5.5.11	LiTE34Quad	25

5.5.12	Apsidal	26
5.5.13	Model	27
5.5.14	KeplerEQ	28
5.5.15	KeplerEQMarkley	28
5.5.16	Epoch	28
5.5.17	PhaseCurve	29
5.5.18	FitGA	29
5.5.19	InfoGA	30
5.5.20	FitMCMC	31
5.5.21	InfoMCMC	32
5.5.22	CalcErr	32
5.5.23	CorrectErr	33
5.5.24	AddWeight	33
5.5.25	Amplitude	33
5.5.26	Summary	34
5.5.27	Chi2	34
5.5.28	MassFun	34
5.5.29	AbsoluteParam	35
5.5.30	ParamsApsidal	35
5.5.31	Plot	35
5.5.32	PlotRes	37
5.5.33	Save	39
5.5.34	Load	39
5.5.35	SaveModel	39
5.5.36	SaveRes	40
5.6	Ukážka	41
6	OCFitLoad	42
6.1	Inicializácia	42
	Literatúra	43

1 Úvod

Knižnica OCFit obsahuje 4 triedy, ktoré slúžia na analýzu a fitovanie O-C diagramov. Tento návod slúži ako stručný sumár dostupných funkcií a ich základný popis.

V prípade použitia tejto knižnice na vedecké účely, citujte, prosím, prácu Gajdoš & Parimucha (2019), kde je aj podrobnejší popis fungovania a implementácie fitovacích funkcií a použitých modelov.

2 Potrebné knižnice

Program funguje v Pythone 2.7 a 3 s nasledovnými knižnicami:

- numpy - testované na verzii 1.10.2
- matplotlib - testované na verzii 1.5.0
- PyAstronomy - testované na verzii 0.9.0
- pymc - testované na verzii 2.3.6
- ga.py - vlastná knižnica, ktorá je potrebná na fitovanie pomocou genetických algoritmov
- info_ga.py - vlastná knižnica na analýzu fitovania pomocou genetických algoritmov
- info_mc.py - vlastná knižnica na analýzu fitovania pomocou Monte Carlo metódy

3 Inštalácia

Inštalácia je možná zo zdrojového kódu alebo použitím inštalačného súboru (iba pre OS Windows). Nasledujúci postup sa týka inštalácie zo zdrojového kódu. Vyextrahujte súbory a prejdite do vytvoreného priečinka. Spustením skriptu `setup.py` sa vykoná inštalácia:

```
python setup.py install
```

alebo pomocou pip

```
pip install ocfit
```

4 FitLinear a FitQuad

Triedy slúžiace na fitovanie lineárneho a kvadratického trendu O-C diagrame spôsobeného prenosom hmoty. Možný je tiež samotný výpočet O-C z napozorovaných časov miním ($O \equiv t_O$). Taktiež možno triedu FitLinear použiť na odstránenie lineárneho trendu v dátach a fitovať len kvadratický trend, vid' Ukážka.

Funkcie a parametre oboch tried sú takmer identické. V prípade, že niektorá funkcia je iba v jednej triede je to zdôraznené.

4.1 Inicializácia

OCFit.**FitLinear**(t, t0, P, oc = None, err = None)

OCFit.**FitQuad**(t, t0, P, oc = None, err = None)

Na inicializáciu triedy je potrebné zadať časy miním (O) a lineárnu efemeridu zákrytov, t.j. čas základného minima a periódu. Ak sú vypočítané hodnoty O-C je ich možné zadať ako ďalší parameter, ak nie automaticky sa vypočítajú na základe zadaných časov miním. Tieto hodnoty je možné ihneď použiť v iných výpočtoch, sú uložené v premennej **oc**. Taktiež je možné zadať chyby určenia miním (a teda aj O-C). Prípadne, ak sú určené váhy jednotlivých bodov, je možné ako chybu zadať prevrátenú hodnotu váhy.

Parametre:

t: numpy.array alebo list

Napozorované časy miním (O).

t0: float

Čas základného minima.

P: float

Pôvodná perióda systému.

oc: numpy.array alebo list, voliteľné

Určené hodnoty O-C v dňoch.

err: numpy.array alebo list, voliteľné

Chyby určenia hodnôt O-C, resp. časov O.

4.2 Dostupné funkcie

4.2.1 FitLinear

OCFit.FitLinear.**FitLinear**()

Základná funkcia na fitovanie lineárneho trendu v O-C diagrame. **Táto funkcia je dostupná iba pre triedu FitLinear.** Nové hodnoty lineárnej efemeridy sú zapísané do premennej `params`; chyby sú v premenných `params_err`. Taktiež sa vypočítajú nové časy miním (C) uložené v premennej `tC` a nové hodnoty O-C uložené v premennej `new_oc`.

Výstup: `numpy.array`

Nové hodnoty O-C pre určenú lineárnu efemeridu v dňoch.

4.2.2 FitQuad

`OCFit.FitQuad.FitQuad()`

Základná funkcia na fitovanie kvadratického trendu v O-C diagrame. **Táto funkcia je dostupná iba pre triedu FitQuad.** Nové hodnoty Q a lineárnej efemeridy sú zapísané do premennej `params`; chyby sú v premenných `params_err`. Taktiež sa vypočítajú nové časy miním (C), uložené v premennej `tC`.

Výstup: `numpy.array`

Nové hodnoty O-C pre určené Q a lineárnu efemeridu v dňoch.

4.2.3 FitRobust

`OCFit.FitLinear.FitRobust(n_iter = 10)`

`OCFit.FitQuad.FitRobust(n_iter = 10)`

Funkcia na fitovanie lineárneho / kvadratického trendu v O-C diagrame využívajúca robustnú regresiu. Túto funkcia je vhodné použiť, ak je v dátach veľa odletujúcich bodov. Detaily v Mikulášek & Zejda (2013). Nové hodnoty fitovaných parametrov sú zapísané do premennej `params`; chyby sú v premenných `params_err`. Taktiež sa vypočítajú nové časy miním (C), uložené v premennej `tC`.

Parametre:

`n_iter`: integer, voliteľné

Počet iterácii robustnej regresie.

Výstup: `numpy.array`

Nové hodnoty O-C.

4.2.4 FitMCMC

`OCFit.FitLinear.FitMCMC(n_iter, limits, steps, fit_params = None, burn = 0, binn = 1, visible = True, db = None)`

`OCFit.FitQuad.FitMCMC(n_iter, limits, steps, fit_params = None, burn = 0, binn = 1, visible = True, db = None)`

Funkcia na fitovanie lineárneho / kvadratického trendu v O-C diagrame využívajúca metódu Monte Carlo (MC). Spustenie fitovania je možné iba, ak sú vypočítané štartovacie hodnoty parametrov. Tie je možné získať s fitovania pomocou funkcie FitLinear / FitQuad alebo FitRobust. Nové hodnoty fitovaných parametrov sú zapísané do premennej **params**; chyby sú v premenných **params_err**. Taktiež sa vypočítajú nové časy minim (C), uložené v premennej **tC**.

Detailnejší popis metódy Monte Carlo je uvedený v časti venovanej triede OCFit (podkapitola 5.5.20).

Parametre:

n_iter: float

Počet MC krokov.

limits: dictionary

Hranice fitovaných parametrov.

steps: dictionary

Kroky fitovaných parametrov.

fit_params: list

Zoznam parametrov, ktoré sa budú fitovať. Východzia hodnota ['P', 't0'] resp. ['Q', 'P', 't0'].

burn: float, voliteľné

Počet MC krokov, ktoré budú odstránené.

bin: float, voliteľné

Veľkosť 1 bloku pri binovaní.

db: string, voliteľné

Názov súboru pre uloženie MC vzorkovania.

visible: boolean, voliteľné

Zobrazenie priebehu fitovania a prehľadu výsledkov z **pymc**.

Výstup: numpy.array

Nové hodnoty O-C.

4.2.5 InfoMCMC

OCFit.FitLinear.**InfoMCMC**(db, eps = False, geweke = False)

OCFit.FitQuad.**InfoMCMC**(db, eps = False, geweke = False)

Funkcia slúži na detailnú analýzu MC vzorkovania získaného pomocou FitMCMC. Pre každý z fitovaných parametrov sa vygeneruje graf zobrazujúci priebeh fitu (trace) a spoločný graf s priebehom fitu a histogramom, tiež sa vytvorí súbor so stručnými informáciami o vzorkovaní a získaných hodnotách parametra a ich chybách. Je možné vytvoriť aj grafy na Geweke analýzu (detaily v Geweke (1992)).

Taktiež sa vytvoria spoločné grafy pre všetky fitované parametre: histogramy, rozptyly (deviance), korelačné grafy a 2D histogramy zobrazujúce intervaly spoľahlivosti. Na 2D histogramy sa využíva funkcia `ConfInt`, zvyšné sa generujú pomocou triedy `TraceAnalysis` z `PyAstronomy`.

Posledným výstupom je tabuľka korelačných koeficientov medzi jednotlivými fitovanými parametrami.

Parametre:

db: string

Názov súboru s uloženým MC vzorkovaním.

eps: boolean, voliteľné

Uloženie grafov aj vo formáte **eps**.

geweke: boolean, voliteľné

Geweke analýza.

4.2.6 Epoch

`OCFit.FitLinear.Epoch()`

`OCFit.FitQuad.Epoch()`

Výpočet epochy jednotlivých napozorovaných miním. Táto funkcia sa volá počas inicializácie triedy. Získané epochy sa ukladajú do premennej **epoch**.

Výstup: `numpy.array`

Vypočítané epochy.

4.2.7 PhaseCurve

`OCFit.FitLinear.PhaseCurve(P, t0, plot = False)`

`OCFit.FitLinear.PhaseCurve(P, t0, plot = False)`

Výpočet fázovej krivky na základe zadanej lineárnej efemeridy. Výslednú fázovú krivku je možné zobraziť na grafe.

Parametre:

P: float

Periódá.

t0: float

Čas základného minima.

plot: boolean, voliteľné

Vykresľovať aj graf alebo len výpočet fázovej krivky.

Výstup: numpy.array, numpy.array

Fáza a prislúchajúca hodnota O-C.

4.2.8 CalcErr

OCFit.FitLinear.**CalcErr()**

OCFit.FitQuad.**CalcErr()**

Výpočet chýb dát na základe nafitovaných parametrov. Funkciu nemožno použiť bez toho, aby dáta boli pred tým fitované. Pred tým je možné dáta nafitovať pomocou funkcie FitLinear / FitQuad alebo FitRobust. Všetkým bodom sa priradí rovnaká chyba. Veľkosť chyby sa určí tak, aby hodnota redukovaná chyba štvorcov χ_r^2 bola rovná 1.

Funkciu je vhodné použiť, ak dáta vykazujú iba lineárny / kvadratický trend. Následne pri opätovnom fitovaní sa získajú lepšie hodnoty chýb parametrov. Funkciu je potrebné použiť pred fitovaním metódou MCMC pomocou funkcie FitMCMC, ak neboli zadané chyby vstupných dát.

Výstup: numpy.array

Vypočítané chyby.

4.2.9 CorrectErr

OCFit.FitLinear.**CorrectErr()**

OCFit.FitQuad.**CorrectErr()**

Znormovanie zadaných chýb dát na základe nafitovaných parametrov. Funkciu nemožno použiť bez toho, aby dáta boli pred tým fitované a aby boli pri inicializácii chyby zadané. Pred tým je možné dáta nafitovať pomocou funkcie FitLinear / FitQuad alebo FitRobust. Veľkosť chyby sa určí tak, aby hodnota redukovaná chyba štvorcov χ_r^2 bola rovná 1. Pri vykresľovaní grafov sa použijú pôvodné hodnoty chýb.

Funkciu je vhodné použiť, ak dáta vykazujú iba lineárny / kvadratický trend. Následne pri opätovnom fitovaní sa získajú lepšie hodnoty chýb parametrov. Funkciu je vhodné použiť pred fitovaním metódou MCMC pomocou funkcie FitMCMC, ak sú chyby vstupných dát zle určené (podhodnotené alebo nadhodnotené).

Výstup: numpy.array

Vypočítané chyby.

4.2.10 AddWeight

OCFit.FitLinear.**AddWeight**(w)

OCFit.FitQuad.**AddWeight**(w)

Pridanie váh bodov a následné znormovanie chýb na základe nafitovaných parametrov. Funkciu nemožno použiť bez toho, aby dáta boli pred tým fitované. Pred tým je možné dáta nafitovať pomocou funkcie FitLinear / FitQuad alebo FitRobust. Veľkosť chyby sa určí tak, aby hodnota redukovaná chyba štvorcov χ_r^2 bola rovná 1.

Funkciu je vhodné použiť, ak dáta vykazujú iba lineárny / kvadratický trend. Následne pri opätovnom fitovaní sa získajú lepšie hodnoty chýb parametrov. Funkciu je potrebné použiť pred fitovaním metódou MCMC pomocou funkcie FitMCMC.

Parametre:

w: numpy.array

Váhy bodov.

Výstup: numpy.array

Vypočítané chyby.

4.2.11 Summary

OCFit.FitLinear.**Summary**(name = None)

OCFit.FitQuad.**Summary**(name = None)

Prehľad hodnôt fitovaných parametrov a ich chýb. Výpis je možné uložiť do súboru alebo vypísať na obrazovku.

Parametre:

name: string, voliteľné

Názov výstupného súboru.

4.2.12 Plot

OCFit.FitLinear.**Plot**(name = None, no_plot = 0, no_plot_err = 0, eps = False, oc_min = True, time_type = 'JD', offset = 2400000, trans = True, title = None, epoch = False, min_type = False, weight = None, trans_weight = False, bw = False, double_ax = False, fig_size = None)

OCFit.FitQuad.**Plot**(name = None, no_plot = 0, no_plot_err = 0, eps = False, oc_min = True, time_type = 'JD', offset = 2400000, trans = True, title = None, epoch = False, min_type = False, weight = None, trans_weight = False, bw = False, double_ax = False, fig_size = None)

Vykreslenie pôvodných hodnôt O-C a lineárneho / kvadratického fitu. Graf je možné uložiť do súboru **png** (prípadne aj do **eps**) alebo ho len zobraziť. Veľmi odletujúce body alebo body s veľkou chybou sa dajú odstrániť z grafu. Hodnoty O-C sa môžu prepočítať na minúty alebo sa použijú pôvodné hodnoty v dňoch. V menovke **x-ovej** možno nastaviť rôzny typ používaného juliánskeho dátumu. Dátum je možné pretransformovať odčítaním zadaného offsetu. Ak bol dátum už naozaj preškálovaný, je potrebné zakázať transformáciu x-ových hodnôt (**trans = False**), ale hodnotu použitého posunu je vhodné zadať kvôli výpisu v menovke osi. Taktiež je možné vykresliť x-ovú os v epoche, prípadne v juliánskom dátume aj v epoche súčasne. V prípade, ak sú k dispozícii O-C primárnych aj sekundárnych minim, je ich možné rozlíšiť pomocou parametra **min_type = True**. Primárne minimá sú znázornené plným krúžkom a sekundárne prázdny. Ak sú k dispozícii váhy bodov, dajú sa zvýrazniť aj na grafe ich nastavením do **weight**. Váhy sú rozdelené do kategórii: 0-3, 3-5, 5-8, 8-10. V prípade, že zadané váhy nie sú z tohto intervalu, je potrebné ich preškálovanie zadaním **trans_weight = True**. Ak boli na začiatku nastavené chyby dát a zadajú sa váhy do **weight**, vykreslí sa graf na základe zadaných váh bez errorbarov.

Parametre:

name: string, voliteľné

Názov súboru (bez prípony), do ktorého sa uloží graf.

no_plot: integer, voliteľné

Počet najviac odletujúcich bodov, ktoré sa v grafe nevykreslia.

no_plot_err: integer, voliteľné

Počet bodov s najväčšou zadanou chybou, ktoré sa v grafe nevykreslia.

eps: boolean, voliteľné

Uloženie grafu aj vo formáte **eps**. (Je potrebné zadať názov do premennej **name**!)

oc_min: boolean, voliteľné

Hodnoty O-C previesť z dní na minúty.

time_type: string, voliteľné

Formát dátumu (JD, HJD, BJD).

offset: float, voliteľné

Posun dátumu kvôli skráteniu hodnôt na x-ovej osi.

trans: boolean, voliteľné

Pretransformovať aj x-ové hodnoty o offset.

title: string, voliteľné

Názov grafu.

epoch: boolean, voliteľné

x-ová os v epoche.

min_type: boolean, voliteľné
Rozlišovanie primárneho a sekundárneho minima.

weight: numpy.array, voliteľné
Váhy bodov.

trans_weight: boolean, voliteľné
Preškáľovanie váh.

bw: boolean, voliteľné
Čierno-biely graf.

double_ax: boolean, voliteľné
Dve x-ové osi - čas + epocha.

fig_size: tuple, voliteľné
Iná veľkosť grafu.

4.2.13 PlotRes

OCFit.FitLinear.**PlotRes**(name = None, no_plot = 0, no_plot_err = 0, eps = False, oc_min = True, time_type = 'JD', offset = 2400000, trans = True, title = None, epoch = False, min_type = False, weight = None, trans_weight = False, bw = False, double_ax = False, fig_size = None)

OCFit.FitQuad.**PlotRes**(name = None, no_plot = 0, no_plot_err = 0, eps = False, oc_min = True, time_type = 'JD', offset = 2400000, trans = True, title = None, epoch = False, min_type = False, weight = None, trans_weight = False, bw = False, double_ax = False, fig_size = None)

Vykreslenie nových hodnôt O-C spočítaných pre určené hodnoty fitovaných parametrov. Graf je možné uložiť do súboru **png** (prípadne aj do **eps**) alebo ho len zobrazieť. Veľmi odľutujúce body alebo body s veľkou chybou sa dajú odstrániť z grafu. Hodnoty O-C sa môžu prepočítať na minúty alebo sa použijú pôvodné hodnoty v dňoch. V menovke x-ovej možno nastaviť rôzny typ používaného juliánskeho dátumu. Dátum je možné pretransformovať odčítaním zadaného offsetu. Ak bol dátum už naozaj preškáľovaný, je potrebné zakázať transformáciu x-ových hodnôt (**trans = False**), ale hodnotu použitého posunu je vhodné zadať kvôli výpisu v menovke osi. Taktiež je možné vykresliť x-ovú os v epoche. V prípade, ak sú k dispozícii O-C primárnych aj sekundárnych miním, je ich možné rozlíšiť pomocou parametra **min_type = True**. Primárne minimá sú znázornené plným krúžkom a sekundárne prázdny. Ak sú k dispozícii váhy bodov, dajú sa zvýrazniť aj na grafe ich nastavením do **weight**. Váhy sú rozdelené do kategórii: 0-3, 3-5, 5-8, 8-10. V prípade, že zadané váhy nie sú z tohto intervalu, je potrebné ich preškáľovanie zadaním

`trans_weight = True`. Ak boli na začiatku nastavené chyby dát a zadajú sa váhy do `weight`, vykreslí sa graf na základe zadaných váh bez errorbarov.

Parametre:

name: string, voliteľné

Názov súboru (bez prípony), do ktorého sa uloží graf.

no_plot: integer, voliteľné

Počet najviac odletujúcich bodov, ktoré sa v grafe nevykreslia.

no_plot_err: integer, voliteľné

Počet bodov s najväčšou zadanou chybou, ktoré sa v grafe nevykreslia.

eps: boolean, voliteľné

Uloženie grafu aj vo formáte `eps`. (Je potrebné zadať názov do premennej `name!`)

oc_min: boolean, voliteľné

Hodnoty O-C previesť z dní na minúty.

time_type: string, voliteľné

Formát dátumu (JD, HJD, BJD).

offset: float, voliteľné

Posun dátumu kvôli skráteniu hodnôt na x-ovej osi.

trans: boolean, voliteľné

Pretransformovať aj x-ové hodnoty o `offset`.

title: string, voliteľné

Názov grafu.

epoch: boolean, voliteľné

x-ová os v epoche.

min_type: boolean, voliteľné

Rozlišovanie primárneho a sekundárneho minima.

weight: numpy.array, voliteľné

Váhy bodov.

trans_weight: boolean, voliteľné

Preškáľovanie váh.

bw: boolean, voliteľné

Čierno-biely graf.

double_ax: boolean, voliteľné

Dve x-ové osi - čas + epocha.

fig_size: tuple, voliteľné

Iná veľkosť grafu.

4.2.14 SaveOC

OCFit.FitLinear.**SaveOC**(name, weight = None)

OCFit.FitQuad.**SaveOC**(name, weight = None)

Uloženie pôvodných hodnôt O-C vypočítaných na základe zadanej lineárnej efemeridy do súboru. Tvar súboru je: čas minima O, epocha, hodnota O-C, chyba (ak bola zadaná) alebo váha (ak bola zadaná).

Parametre:

name: string

Názov výstupného súboru.

weight: numpy.array, voliteľné

Váhy bodov.

4.2.15 SaveRes

OCFit.FitLinear.**SaveRes**(name, weight = None)

OCFit.FitQuad.**SaveRes**(name, weight = None)

Uloženie reziduí (nových hodnôt O-C) vypočítaných na základe hodnôt fitovaných parametrov do súboru. Tvar súboru je: čas minima O, epocha, hodnota O-C, chyba (ak bola zadaná) alebo váha (ak bola zadaná).

Parametre:

name: string

Názov výstupného súboru.

weight: numpy.array, voliteľné

Váhy bodov.

4.3 Ukážka

Ukážka práce s oboma triedami s nasimulovanými dátami. Dáta zo súboru je vhodné načítavať pomocou funkcie `numpy.loadtxt`.

```

from OCFit import FitQuad, FitLinear
import numpy as np

#vygenerovanie dat
E = np.arange(0, 100, 1) #epocha
#simulovanie napozorovanych casov
t = 1e-5*E**2+15*E + 1540.4 + np.random.normal(scale = 0.01, size = E.shape)
err = 0.01*np.ones(E.shape) #chyby urcenia casov

#pouzitie FitLinear na prvotny odhad linearnej efemeridy
#inicializacia s pouzitim odhadu (povodnych hodnot) linearnej efemeridy
#pri dlhsom casovom rozsahu merani je potrebne zadat periodu, ktora je blizka
#spravnej hodnote, inak dojde k pomiesaniu primarnych a sekundarnych minim
lin = FitLinear(t, 1540, 15.01, err = err)
oc = lin.oc #O-C na zaklade povodnej linearnej efemeridy
lin.FitLinear() #linearny fit
lin.FitRobust() #fitovanie robustnou regresiou

lin.Summary() #prehlad parametrov

err = lin.CorrectErr() #znormovanie chyb
oc = lin.FitRobust() #fitovanie robustnou regresiou

lin.Summary() #prehlad parametrov

#vykreslenie grafou
#graf povodnych O-C s fitom, bez transformovanie hodnot na x-ovej osi
lin.Plot(trans = False)
#graf rezidui O-C a fitom, bez transformovanie hodnot na x-ovej osi
lin.PlotRes(trans = False)

#pouzitie triedy FitQuad
#ako odhad linearnej efemeridy su pozite vysledky z FitLinear
#taktiez su fitovane len novovypocitane hodnoty O-C
quad = FitQuad(t, lin.t0, lin.P, oc = oc, err = err)
oc = quad.oc #O-C na zaklade linearnej efemeridy
quad.FitQuad() #kvadraticky fit
quad.FitRobust() #fitovanie robustnou regresiou

quad.Summary() #prehlad parametrov

#vykreslenie grafou
#graf povodnych O-C s fitom, bez transformovanie hodnot na x-ovej osi
quad.Plot(trans = False)
#graf rezidui O-C a fitom, bez transformovanie hodnot na x-ovej osi
quad.PlotRes(trans = False)

```

5 OCFit

Trieda na fitovanie O-C diagramov podľa rôznych modelov: LiTE pre 3. prípadne 4. teleso, prenos hmoty a LiTE pre 3. prípadne 4. teleso, stáčanie priamky apsid a modelov O-C na základe Agol et al. (2005).

5.1 Inicializácia

OCFit.**OCFit**(t, oc, err = None)

Na inicializáciu triedy je potrebné zadať časy miním (O) a hodnoty O-C. Hodnoty O-C je možné určiť pomocou tried FitLinear a FitQuad. Taktiež je možné zadať chyby určenia O-C.

Parametre:

t: numpy.array alebo list

Napozorované časy miním (O).

oc: numpy.array alebo list

Určené hodnoty O-C v dňoch.

err: numpy.array alebo list, voliteľné

Chyby určenia hodnôt O-C.

5.2 Nastavenie parametrov pre fitovanie

OCFit.**OCFit**.?

Niektoré parametre modelu a fitovania je nutné nastaviť samostatne po inicializácii. Ide hlavne o hranice a kroky parametrov, hodnoty fixovaných parametrov, výber parametrov na fitovanie a model O-C, ktorý sa bude fitovať.

Parametre:

limits: dictionary

Hranice fitovaných parametrov. Pre každý parameter je potrebné zadať list s dolnou a hornou hranicou. Pomenovanie parametrov je uvedené pri konkrétnom modeli.

steps: dictionary

Kroky fitovaných parametrov. Pre každý parameter je potrebné zadať veľkosť kroku. Pomenovanie parametrov je uvedené pri konkrétnom modeli.

fit_params: list

Zoznam parametrov, ktoré sa budú fitovať. Pomenovanie parametrov je uvedené pri konkrétnom modeli.

params: dictionary, voliteľné

Hodnoty zafixovaných parametrov. Pomenovanie parametrov je uvedené pri konkrétnom modeli.

model: string, voliteľné

Nastavenie modelu na fitovanie O-C. Pomenovanie modelu je uvedené pri samotnej modelovej funkcii. Štandardný model je LiTE3.

5.3 Práca s váhami bodov

Pokiaľ je to možné, je lepšie pracovať priamo s chybami jednotlivých bodov a nie s váhami! V prípade práce s váhami (w) je možné chyby vypočítať ako $1/w$. Častokrát sú takto vypočítané chyby výrazne nadhodnotené.

Pri práci s váhami bodov je vhodné postupovať podľa tohto postupu:

1. pri inicializácii zadať chyby ako $1/w$
2. fitovanie pomocou FitGA
3. znormovať chyby použitím funkcie AddWeight
4. fitovanie metódou MC funkciou FitMCMC
5. pri ukladaní dát funkciou SaveRes zadať váhy cez parameter `weight`
6. pri vykresľovaní grafov zadať váhy cez parameter `weight`, a príp. použiť aj parameter `trans.weight = True`

5.4 Všeobecné poznámky k práci s triedou OCFit

Základom úspešného fitovania je zvoliť vhodný model. Je dobré minimalizovať počet parametrov modelu a tiež počet fitovaných parametrov. To znamená nepoužívať model s veľkým množstvom parametre, pričom väčšina z nich je fixovaná, ak to nie je naozaj nevyhnutné. Použitím takéhoto komplikovaného modelu sa zbytočne predlžuje čas výpočtu a niekedy aj znižuje kvalita získaných výsledkov. Preto je niekedy vhodné pracovať s rezíduami a nie s pôvodnými dátami, napr. pri fitovaní O-C diagramu ovplyvnením LiTE od 3. aj 4. telesa. Prípadne najprv odstrániť lineárny / kvadratický trend pomocou funkcií FitLinear a FitQuad a až následne fitovať komplikovanejšie zmeny. Je taktiež zbytočné fitovať parameter, ktorý má takmer nulovú hodnotu, bez akéhokoľvek vplyvu na tvar modelovej krivky, a jeho hodnota je na úrovni chyby jeho určenia. Dobrým príkladom je kvadratický člen Q pri modeloch LiTE3Quad a LiTE34Quad. V prípade, že nie je v dátach kvadratický, ale len lineárny člen, je vhodné ho fixovať na 0. Jeho odstránením z fitovaných parametrov sa spresní fitovanie a je tiež možné znížiť počet iterácií pri fitovaní.

Druhým kľúčovým bodom je nastaviť vhodné intervaly pre fitovanie. Vo väčšine prípadov môžu byť dostatočne široké. Ich zúženie však zvýši presnosť fitovania a zníži počet potrebných iterácií. V prípade excentricity e a dĺžky pericentra w je možné nastaviť celý logicky možný interval, t.j. od 0 po 1 pre excentricity a od 0 po 2π pre dĺžku pericentra. Pri iných parametroch (ako

perióda 3. telesa P3) sa dá hodnota odhadnúť zo samotného priebehu O-C diagramu. Naopak v prípade lineárnej efemeridy t_0 a P je potrebné zvoliť dostatočne malý interval, keďže tvar O-C diagramu je veľmi závislý od správnej voľby lineárnej efemeridy. Veľmi široký interval častokrát vedie k úplne nesprávnym výsledkom fitovania. V závislosti od konkrétneho O-C diagramu by šírka vyhľadávacích intervaloch pre lineárnu efemeridu nemala prekročiť 1-2 hodiny. Najlepšie je sa, pokiaľ je to možné, fitovaniu komplikovanejšieho modelu spolu s lineárnou efemeridou vyhnúť.

Pre úspešné fitovanie je potrebné nastaviť dostatočný počet iterácií jednotlivých fitovacích algoritmov. Pri použití genetických algoritmov (GA) by malo byť dostatočné nastaviť počet generácií aj veľkosť jednej generácie na $\gtrsim 100 - 200 \times \text{počet fitovaných parametrov}$. Ako nejaké univerzálne nastavenie sa dá použiť počet aj veľkosť generácií na 1000. Pre prvotné otestovanie fitovania pomocou GA stačí nastaviť oba parametre na 100. V prípade metódy Monte Carlo sa potrebný počet iterácií líši v závislosti od kvality vstupných dát, presnosti štartovacích hodnôt a typu modelu, zväčša sa pohybuje od 10^5 do 10^7 . Univerzálna hodnota by mohla byť 10^6 . Častokrát je potrebné odstrániť aspoň 10^3 prvých hodnôt a použiť binovanie s veľkosťou 10 – 100. Na prvotný odhad stačí použiť 1000 iterácií bez odstraňovania hodnôt a s jednotkovým binovaním.

5.5 Dostupné funkcie

5.5.1 AvailableModels

OCFit.OCFit.**AvailableModels()**

Zobrazenie dostupných modelov O-C. Názvy jednotlivých modelov sú uložené aj v premennej availableModels.

5.5.2 ModelParams

OCFit.OCFit.**ModelParams**(model = None, allModels = False)

Zobrazenie parametrov vybraného modelu O-C alebo všetkých dostupných modelov. Ak sa nezadá žiaden parameter, zobrazia sa parametre modelu nastaveného v premennej model.

Parametre:

model: string, voliteľné

Názov vybraného modelu.

allModels: boolean, voliteľné

Zobrazenie parametrov všetkých dostupných modelov.

5.5.3 LiTE

OCFit.OCFit.**LiTE**(t, a_sin_i3, e3, w3, t03, P3)

Funkcia na výpočet O-C na základe modelu LiTE vplyvom 3. telesa. Detaily v Irwin (1952). Nejde o funkciu pre fitovanie modelu, ale len o pomocnú funkciu!

Parametre:

t: numpy.array

Časy (O), pre ktoré sa počítajú O-C.

a.sin.i3: float

Hodnota $a \sin i3$ v AU.

e3: float

Excentricita dráhy 3. telesa.

w3: float

Dĺžka periastra dráhy 3. telesa v radiánoch.

t03: float

Čas prechodu 3. telesa periastrom.

P3: float

Periódá 3. telesa v dňoch.

Výstup: numpy.array

Hodnoty O-C určené na základe modelu LiTE.

5.5.4 AgolInPlanet

OCFit.OCFit.**AgolInPlanet**(t, P, a, w, e, mu3, r3, w3, t03, P3)

Funkcia na výpočet O-C na základe modelu perturbácií spôsobených vnútornou planétou podľa práce Agol et al. (2005, rovnica (12)). Pre použitie modelu nastavte `model = 'AgolInPlanet'`. Názvy parametrov na fitovanie, určenie hraníc, krokov a hodnôt parametrov sú rovnaké ako názvy vstupných parametrov (okrem času t). Podrobný popis ako nastaviť tieto hodnoty je uvedený v časti 5.2.

Parametre:

t: numpy.array

Časy (O), pre ktoré sa počítajú O-C.

P: float

Periódá tranzitujúcej exoplanéty v dňoch.

a: float

Veľká poloos dráhy tranzitujúcej exoplanéty v AU.

w: float

Dĺžka periastra tranzitujúcej exoplanéty v radiánoch.

e: float

Excentricita dráhy tranzitujúcej exoplanéty.

mu3: float

Redukovaná hmotnosť 3. telesa (vnútornej planéty).

r3: float

Polomer dráhy 3. telesa v AU.

w3: float

Dĺžka periastra 3. telesa v radiánoch.

t03: float

Čas prechodu 3. telesa periastrom.

P3: float

Periódna 3. telesa v dňoch.

Výstup: numpy.array

Hodnoty O-C určené na základe modelu.

5.5.5 AgolInPlanetLin

OCFit.OCFit.**AgolInPlanetLin**(t, t0, P, a, w, e, mu3, r3, w3, t03, P3)

Funkcia na výpočet O-C na základe modelu perturbácií spôsobených vnútornou planétou podľa práce Agol et al. (2005, rovnica (12)) v kombinácii s lineárnym modelom O-C. Pred použitím tohto modelu je potrebné spustiť funkciu Epoch. Pre použitie modelu nastavte `model = 'AgolInPlanetLin'`. Názvy parametrov na fitovanie, určenie hraníc, krokov a hodnôt parametrov sú rovnaké ako názvy vstupných parametrov (okrem času t). Podrobný popis ako nastaviť tieto hodnoty je uvedený v časti 5.2.

Parametre:

t: numpy.array

Časy (O), pre ktoré sa počítajú O-C.

t0: float

Čas základného tranzitu.

P: float

Periódna tranzitujúcej exoplanéty v dňoch.

a: float

Veľká poloos dráhy tranzitujúcej exoplanéty v AU.

w: float

Dĺžka periastra tranzitujúcej exoplanéty v radiánoch.

e: float

Excentricita dráhy tranzitujúcej exoplanéty.

mu3: float

Redukovaná hmotnosť 3. telesa (vnútornej planéty).

r3: float

Polomer dráhy 3. telesa v AU.

w3: float

Dĺžka periastra 3. telesa v radiánoch.

t03: float

Čas prechodu 3. telesa periastrom.

P3: float

Periódka 3. telesa v dňoch.

Výstup: numpy.array

Hodnoty O-C určené na základe modelu.

5.5.6 AgolExPlanet

OCFit.OCFit.**AgolExPlanet**(t, P, mu3, e3, t03, P3)

Funkcia na výpočet O-C na základe modelu perturbácií spôsobených vonkajšou planétou podľa práce Agol et al. (2005, rovnica (25)). Pre použitie modelu nastavte `model = 'AgolExPlanet'`. Názvy parametrov na fitovanie, určenie hraníc, krokov a hodnôt parametrov sú rovnaké ako názvy vstupných parametrov (okrem času t). Podrobný popis ako nastaviť tieto hodnoty je uvedený v časti 5.2.

Parametre:

t: numpy.array

Časy (O), pre ktoré sa počítajú O-C.

P: float

Periódka tranzitujúcej exoplanéty v dňoch.

mu3: float

Redukovaná hmotnosť 3. telesa (vonkajšej planéty).

e3: float

Excentricita dráhy 3. telesa.

t03: float

Čas prechodu 3. telesa periastrom.

P3: float

Periódá 3. telesa v dňoch.

Výstup: numpy.array

Hodnoty O-C určené na základe modelu.

5.5.7 AgolExPlanetLin

OCFit.OCFit.**AgolExPlanetLin**(t, t0, P, mu3, e3, t03, P3)

Funkcia na výpočet O-C na základe modelu perturbácií spôsobených vonkajšou planétou podľa práce Agol et al. (2005, rovnica (25)) v kombinácii s lineárnym modelom O-C. Pred použitím tohto modelu je potrebné spustiť funkciu Epoch. Pre použitie modelu nastavte `model = 'AgolExPlanetLin'`. Názvy parametrov na fitovanie, určenie hraníc, krokov a hodnôt parametrov sú rovnaké ako názvy vstupných parametrov (okrem času t). Podrobný popis ako nastaviť tieto hodnoty je uvedený v časti 5.2.

Parametre:

t: numpy.array

Časy (O), pre ktoré sa počítajú O-C.

t0: float

Čas základného tranzitu.

P: float

Periódá tranzitujúcej exoplanéty v dňoch.

mu3: float

Redukovaná hmotnosť 3. telesa (vonkajšej planéty).

e3: float

Excentricita dráhy 3. telesa.

t03: float

Čas prechodu 3. telesa periastrom.

P3: float

Periódá 3. telesa v dňoch.

Výstup: numpy.array

Hodnoty O-C určené na základe modelu.

5.5.8 LiTE3

OCFit.OCFit.**LiTE3**(t, a_sin_i3, e3, w3, t03, P3)

Funkcia na výpočet O-C na základe modelu LiTE vplyvom 3. telesa. Detaily v Irwin (1952). Pre použitie modelu nastavte `model = 'LiTE3'`. Názvy parametrov na fitovanie, určenie hraníc, krokov a hodnôt parametrov sú rovnaké ako názvy vstupných parametrov (okrem času t). Podrobný popis ako nastaviť tieto hodnoty je uvedený v časti 5.2.

Parametre:

t: numpy.array

Časy (O), pre ktoré sa počítajú O-C.

a_sin_i3: float

Hodnota $a \sin i_3$ v AU.

e3: float

Excentricita dráhy 3. telesa.

w3: float

Dĺžka periastra dráhy 3. telesa v radiánoch.

t03: float

Čas prechodu 3. telesa periastrom.

P3: float

Periódá 3. telesa v dňoch.

Výstup: numpy.array

Hodnoty O-C určené na základe modelu.

5.5.9 LiTE3Quad

OCFit.OCFit.**LiTE3Quad**(t, t0, P, Q, a_sin_i3, e3, w3, t03, P3)

Funkcia na výpočet O-C na základe modelu LiTE vplyvom 3. telesa v kombinácii s kvadratickým modelom O-C. Detaily v Irwin (1952). Pred použitím tohto modelu je potrebné spustiť funkciu Epoch. Pre použitie modelu nastavte `model = 'LiTE3Quad'`. Názvy parametrov na fitovanie, určenie hraníc, krokov a hodnôt parametrov sú rovnaké ako názvy vstupných parametrov (okrem času t). Podrobný popis ako nastaviť tieto hodnoty je uvedený v časti 5.2.

Parametre:

t: numpy.array

Časy (O), pre ktoré sa počítajú O-C.

t0: float

Čas základného minima.

P: float

Periódá zákrytovej dvojhviezdy v dňoch.

Q: float

Kvadratický člen.

a_sin_i3: float

Hodnota $a \sin i_3$ v AU.

e3: float

Excentricita dráhy 3. telesa.

w3: float

Dĺžka periastra dráhy 3. telesa v radiánoch.

t03: float

Čas prechodu 3. telesa periastrom.

P3: float

Periódá 3. telesa v dňoch.

Výstup: numpy.array

Hodnoty O-C určené na základe modelu.

5.5.10 LiTE34

OCFit.OCFit.**LiTE34**(t, a_sin_i3, e3, w3, t03, P3, a_sin_i4, e4, w4, t04, P4)

Funkcia na výpočet O-C na základe modelu LiTE vplyvom 3. telesa a 4. telesa. Details v Irwin (1952). Pre použitie modelu nastavte `model = 'LiTE34'`. Názvy parametrov na fitovanie, určenie hraníc, krokov a hodnôt parametrov sú rovnaké ako názvy vstupných parametrov (okrem času t). Podrobný popis ako nastaviť tieto hodnoty je uvedený v časti 5.2.

Parametre:

t: numpy.array

Časy (O), pre ktoré sa počítajú O-C.

a_sin_i3: float

Hodnota $a \sin i_3$ v AU.

e3: float

Excentricita dráhy 3. telesa.

w3: float

Dĺžka periastra dráhy 3. telesa v radiánoch.

t03: float

Čas prechodu 3. telesa periastrom.

P3: float

Periódá 3. telesa v dňoch.

a_sin_i4: float

Hodnota $a \sin i_4$ v AU.

e4: float

Excentricita dráhy 4. telesa.

w4: float

Dĺžka periastra dráhy 4. telesa v radiánoch.

t04: float

Čas prechodu 4. telesa periastrom.

P4: float

Periódá 4. telesa v dňoch.

Výstup: numpy.array

Hodnoty O-C určené na základe modelu.

5.5.11 LiTE34Quad

OCFit.OCFit.**LiTE34Quad**(t, t0, P, Q, a_sin_i3, e3, w3, t03, P3, a_sin_i4, e4, w4, t04, P4)

Funkcia na výpočet O-C na základe modelu LiTE vplyvom 3. telesa a 4. telesa v kombinácii s kvadratickým modelom O-C. Detaily v Irwin (1952). Pred použitím tohto modelu je potrebné spustiť funkciu Epoch. Pre použitie modelu nastavte `model = 'LiTE34Quad'`. Názvy parametrov na fitovanie, určenie hraníc, krokov a hodnôt parametrov sú rovnaké ako názvy vstupných parametrov (okrem času t). Podrobný popis ako nastaviť tieto hodnoty je uvedený v časti 5.2.

Parametre:

t: numpy.array

Časy (O), pre ktoré sa počítajú O-C.

t0: float

Čas základného minima.

P: float

Periódá zákrytovej dvojhviezdy v dňoch.

Q: float

Kvadratický člen.

a_sin_i3: float

Hodnota $a \sin i_3$ v AU.

e3: float

Excentricita dráhy 3. telesa.

w3: float

Dĺžka periastra dráhy 3. telesa v radiánoch.

t03: float

Čas prechodu 3. telesa periastrom.

P3: float

Periódá 3. telesa v dňoch.

a_sin_i4: float

Hodnota $a \sin i_4$ v AU.

e4: float

Excentricita dráhy 4. telesa.

w4: float

Dĺžka periastra dráhy 4. telesa v radiánoch.

t04: float

Čas prechodu 4. telesa periastrom.

P4: float

Periódá 4. telesa v dňoch.

Výstup: numpy.array

Hodnoty O-C určené na základe modelu.

5.5.12 Apsidal

OCFit.OCFit.**Apsidal**(t, t0, P, w0, dw, e, min_type)

Funkcia na výpočet O-C na základe modelu stáčania priamky apsid. Detaily v Giménez & Bastero (1995). Pred použitím tohto modelu je potrebné spustiť funkciu Epoch. Pre použitie modelu nastavte `model = 'Apsidal'`. Názvy parametrov na fitovanie, určenie hraníc, krokov a hodnôt parametrov sú rovnaké ako názvy vstupných parametrov (okrem času `t` a typu minima `min_type`). Podrobný popis ako nastaviť tieto hodnoty je uvedený v časti 5.2.

Parametre:

t: numpy.array

Časy (O), pre ktoré sa počítajú O-C.

t0: float

Čas základného minima.

P: float

Periódá zákrytovej dvojhviezdy v dňoch.

w0: float

Počiatočná dĺžka periastra v radiánoch.

dw: float

Uhlová rýchlosť priamky apsíd v radiánoch za periódu.

e: float

Excentricita dráhy.

min_type: numpy.array

Typ minima (0 = primárne, 1 = sekundárne).

Výstup: numpy.array

Hodnoty O-C určené na základe modelu.

5.5.13 Model

OCFit.OCFit.**Model**(t = None, params = None, min_type = None)

Funkcia na výpočet zvoleného modelu O-C na základe zadanej sady parametrov v zadanom čase. Jednotlivé parametre sú zadávané cez dictionary, pričom ich názvy zodpovedajú názvom parametrov nastaveného modelu (viď jednotlivé modely). Ak sa niektorý z parametrov tejto funkcie nezadá, použije sa miesto neho prislúchajúci parameter nastavený v samotnej triede.

Parametre:

t: numpy.array alebo float, voliteľné

Časy, pre ktoré sa má vypočítať model.

params: dictionary, voliteľné

Sada parametrov modelu.

min_type: numpy.array alebo float, voliteľné

Typ minima (0 = primárne, 1 = sekundárne).

Výstup: numpy.array

Hodnoty O-C určené na základe modelu.

5.5.14 KeplerEQ

OCFit.OCFit.**KeplerEQ**(M, e, eps = 1e-10)

Funkcia slúži na riešenie Keplerovej rovnice. Na riešenie sa používa Newton-Raphsonova iteračná metóda. Ako štartovacia hodnota sa využíva štartovací vzťah S_9 z práce Odell & Gooding (1986). Riešenie konverguje pre všetky hodnoty e a M . Pre $e \gtrsim 0.9$ sa výpočtový čas začína výrazne predlžovať, preto sa v takýchto prípadoch využíva funkcia KeplerEQMarkley.

Parametre:

M: numpy.array alebo float

Hodnoty strednej anomálie v radiánoch.

e: float

Excentricita dráhy.

eps: float, voliteľné

Požadovaná presnosť riešenia.

Výstup: numpy.array alebo float

Hodnoty excentrickej anomálie v radiánoch.

5.5.15 KeplerEQMarkley

OCFit.OCFit.**KeplerEQMarkley**(M, e)

Funkcia slúži na riešenie Keplerovej rovnice. Na riešenie sa využíva neiteratívny algoritmus z práce Markley (1995). Výpočtový čas je vždy približne rovnaký nezávisiac na zadaných hodnotách e a M . Pre $e \gtrsim 0.9$ je výpočet rýchlejší ako pri použití funkcie KeplerEQ, ale pre menšie hodnoty excentricity je lepšie používať funkciu KeplerEQ.

Parametre:

M: numpy.array alebo float

Hodnoty strednej anomálie v radiánoch.

e: float

Excentricita dráhy.

Výstup: numpy.array alebo float

Hodnoty excentrickej anomálie v radiánoch.

5.5.16 Epoch

OCFit.OCFit.**Epoch**(t0, P, t = None)

Výpočet epochy jednotlivých napozorovaných miním. Túto funkcia je potrebné použiť pri fitovaní modelu obsahujúceho lineárny alebo kvadratický model, alebo modelu stáčania priamky

apsíd, prípadne pri vykresľovaní grafov s x-ovou osou v epoche. Získané epochy sa ukladajú do premennej `epoch`.

Parametre:

t0: float

Čas základného minima.

P: float

Periódá.

t: numpy.array alebo float, voliteľné

Časy miním.

Výstup: numpy.array

Vypočítané epochy.

5.5.17 PhaseCurve

OCFit.FitLinear.**PhaseCurve**(P, t0, plot = False)

Výpočet fázovej krivky na základe zadanej periódy. Výslednú fázovú krivku je možné zobrazíť na grafe.

Parametre:

P: float, voliteľné

Periódá.

t0: float, voliteľné

Čas základného minima.

plot: boolean, voliteľné

Vykresľovať aj graf alebo len výpočet fázovej krivky.

Výstup: numpy.array, numpy.array

Fáza a prislúchajúca hodnota O-C.

5.5.18 FitGA

OCFit.OCFit.**FitGA**(generation, size, mut = 0.5, SP = 2, plot_graph = False, visible = True, n_thread = 1, db = None)

Fitovanie O-C pomocou genetických algoritmov (GA). Na použitie tejto funkcie je potrebné nastaviť intervaly a kroky jednotlivých parametrov (detaily v časti 5.2). Fitovanie pomocou GA slúži hlavne na získanie prvotného odhadu hodnôt jednotlivých parametrov, ktorý sa následne dá použiť pri fitovaní pomocou MCMC funkciou FitMCMC.

Pri fitovaní je potrebné nastaviť vhodný počet generácií a veľkosť jednej generácie v premenných `generation`, resp. `size`. Rady, ako ich nastaviť, sú v časti 5.4. Celkový počet vykonaných výpočtov modelu je `generation × size`. Fitovanie je možné vylepšiť správnou voľbou pravdepodobnosti mutácií a hodnoty selekčného tlaku.

Získaný súbor jedincov je možné uložiť do súboru a následne ho použiť na ďalšiu analýzu, napr. pomocou funkcie `InfoGA`.

Podrobnejšie informácie o použití genetických algoritmov je možné nájsť v Hartmann & Rieger (2002) a Weise (2011).

Parametre:

generation: integer

Počet generácií.

size: integer

Veľkosť jednej generácie, t.j. počet jedincov v jednej generácii.

mut: float, voliteľné

Pravdepodobnosť mutácií (v intervale od 0 do 1).

SP: float, voliteľné

Hodnota selekčného tlaku (details v Razali & Geraghty (2011)).

plot_graph: boolean, voliteľné

Vykreslenie grafu s priebehom χ^2 chyby v jednotlivých generáciách.

visible: boolean, voliteľné

Zobrazenie priebehu fitovania.

n.thread: integer, voliteľné

Počet vlákien pre multithreading.

db: string, voliteľné

Názov súboru pre uloženie fitovania.

Výstup: dictionary

Určené hodnoty parametrov.

5.5.19 InfoGA

`OCFit.OCFit.InfoGA(db, eps = False)`

Funkcia slúži na detailnú analýzu výsledkov GA získaného pomocou `FitGA`. Pre každý z fitovaných parametrov sa vygeneruje graf zobrazujúci hodnoty daného parametra počas jednotlivých generácií.

Taktiež sa vytvorí graf s priebehom χ^2 chyby v jednotlivých generáciách a súbor so sumárnymi informáciami o fitovaní.

Parametre:

db: string

Názov súboru s uloženým fitovaním.

eps: boolean, voliteľné

Uloženie grafov aj vo formáte **eps**.

5.5.20 FitMCMC

OCFit.OCFit.**FitMCMC**(n_iter, burn = 0, binn = 1, visible = True, db = None)

Fitovanie O-C pomocou metódy Markov chain Monte Carlo (MCMC). Na použitie tejto funkcie je nevyhnutné mať nainštalovanú knižnicu **pymc** (Patil et al., 2010). Spustenie fitovania je možné iba, ak sú zadané štartovacie hodnoty parametrov. Tie je možné získať s fitovania pomocou funkcie **FitGA** alebo ručným zadaním do **params** (detaily v časti 5.2).

Podobne ako pri fitovaní pomocou **FitGA** je potrebné zadať počet krokov fitovania, tu na to slúži parameter **n_iter**. Rady, aký počet iterácií nastaviť, sú v časti 5.4. Okrem počtu iterácií je niekedy potrebné nastaviť aj ďalšie parametre pre MCMC. V prípade, že nie sú štartovacie hodnoty blízko optimálnych hodnôt, ku ktorým konverguje MC, je potrebné zadať pomocou **burn** počet MC krokov, ktoré sa odstraňujú. Slúži to na to, aby sa na výpočet stredných hodnôt a ich chýb použili iba hodnoty po dosiahnutí rovnovážneho stavu. Ďalším problémom môže byť veľká korelácia medzi po sebe idúcimi MC krokmi. Tú je možné zredukovať zbinovaním vygenerovanej postupnosti MC krokov. Veľkosť 1 bloku sa zadá do parametra **binn**.

Získané MC vzorkovanie je možné uložiť do súboru a následne ho použiť na ďalšiu analýzu, napr. pomocou funkcie **InfoMCMC**.

Ak neboli pri inicializácii triedy načítané chyby dát, je potrebné ich vypočítať pomocou funkcie **CalcErr**. Prípadne, ak sa pracuje s váhami bodov je ich nutné previesť na chyby pomocou **AddWeight**. V prípade, že fitovanie pomocou **FitGA** dáva dobré výsledky, ale fitovanie cez MCMC ich výrazne zhorší, je problém v nadhodnotených chybách dát. Je ich možné skorigovať použitím **CorrectErr**.

Podrobnejšie informácie o metóde MCMC je možné nájsť v Brooks et al. (2011).

Parametre:

n_iter: integer

Počet MC krokov.

burn: integer, voliteľné

Počet MC krokov, ktoré budú odstránené.

binn: integer, voliteľné

Veľkosť 1 bloku pri binnovaní.

visible: boolean, voliteľné

Zobrazenie priebehu fitovania a prehľadu výsledkov z `pymc`.

db: string, voliteľné

Názov súboru pre uloženie MC vzorkovania.

Výstup: dictionary, dictionary

Určené hodnoty parametrov a ich chyby.

5.5.21 InfoMCMC

`OCFit.OCFit.InfoMCMC(db, eps = False, geweke = False)`

Funkcia slúži na detailnú analýzu MCMC vzorkovania získaného pomocou `FitMCMC`. Pre každý z fitovaných parametrov sa vygeneruje graf zobrazujúci priebeh fitu (trace) a spoločný graf s priebehom fitu a histogramom, tiež sa vytvorí súbor so stručnými informáciami o vzorkovaní a získaných hodnotách parametra a ich chybách. Je možné vytvoriť aj grafy na Geweke analýzu (detaily v Geweke (1992)).

Taktiež sa vytvoria spoločné grafy pre všetky fitované parametre: histogramy, rozptyly (deviance), korelačné grafy a 2D regióny spoľahlivosti. Na 2D regióny spoľahlivosti sa využíva funkcia `ConfInt`, zvyšné sa generujú pomocou triedy `TraceAnalysis` z `PyAstronomy`.

Posledným výstupom je tabuľka korelačných koeficientov medzi jednotlivými fitovanými parametrami.

Parametre:

db: string

Názov súboru s uloženým MC vzorkovaním.

eps: boolean, voliteľné

Uloženie grafov aj vo formáte `eps`.

geweke: boolean, voliteľné

Geweke analýza.

5.5.22 CalcErr

`OCFit.OCFit.CalcErr()`

Výpočet chýb dát na základe nafitovaných parametrov. Funkciu nemožno použiť bez toho, aby dáta boli pred tým fitované. Všetkým bodom sa priradí rovnaká chyba. Veľkosť chyby sa určí tak, aby hodnota redukovaná chyba štvorcov χ_r^2 bola rovná 1.

Funkciu je potrebné použiť pred fitovaním metódou MCMC pomocou funkcie `FitMCMC`, ak neboli zadané chyby vstupných dát. Pred tým je možné dáta nafitovať genetickými algorit-

mami pomocou funkcie FitGA. Predpokladá sa, že aktuálne hodnoty parametrov modelu dobre zodpovedajú reálnym dátam.

Výstup: numpy.array

Vypočítané chyby.

5.5.23 CorrectErr

OCFit.OCFit.**CorrectErr**()

Znormovanie zadaných chýb dát na základe nafitovaných parametrov. Funkciu nemožno použiť bez toho, aby dáta boli pred tým fitované a aby boli pri inicializácii chyby zadané. Veľkosť chyby sa určí tak, aby hodnota redukovaná chyba štvorcov χ_r^2 bola rovná 1. Pri vykresľovaní grafov sa použijú pôvodné hodnoty chýb.

Funkciu je vhodné použiť pred fitovaním metódou MCMC pomocou funkcie FitMCMC, ak sú chyby vstupných dát zle určené (podhodnotené alebo nadhodnotené). Pred tým je možné dáta nafitovať genetickými algoritmami pomocou funkcie FitGA. Predpokladá sa, že aktuálne hodnoty parametrov modelu dobre zodpovedajú reálnym dátam.

Výstup: numpy.array

Vypočítané chyby.

5.5.24 AddWeight

OCFit.OCFit.**AddWeight**(w)

Pridanie váh bodov a následné znormovanie chýb na základe nafitovaných parametrov. Funkciu nemožno použiť bez toho, aby dáta boli pred tým fitované. Veľkosť chyby sa určí tak, aby hodnota redukovaná chyba štvorcov χ_r^2 bola rovná 1.

Funkciu je potrebné použiť pred fitovaním metódou MCMC pomocou funkcie FitMCMC. Pred tým je možné dáta nafitovať genetickými algoritmami pomocou funkcie FitGA. Predpokladá sa, že aktuálne hodnoty parametrov modelu dobre zodpovedajú reálnym dátam.

Parametre:

w: numpy.array

Váhy bodov.

Výstup: numpy.array

Vypočítané chyby.

5.5.25 Amplitude

OCFit.OCFit.**Amplitude**()

Funkcia na výpočet amplitúdy O-C na základe použitého modelu a nastavených parametrov. Ak bola použitá funkcia `FitMCMC`, vypočítajú sa aj chyby určenia amplitúdy. Získané hodnoty sa uložia do premennej `paramsMore`, v prípade chýb sa uložia do `paramsMore_err`.

Výstup: dictionary

Vypočítaná amplitúda a jej chyba.

5.5.26 Summary

`OCFit.OCFit.Summary(name = None)`

Prehľad hodnôt fitovaných parametrov a ich chýb. Výpis je možné uložiť do súboru alebo vypísať na obrazovku.

Parametre:

name: string, voliteľné

Názov výstupného súboru.

5.5.27 Chi2

`OCFit.OCFit.Chi2(params)`

Funkcia na výpočet chyby štvorcov pre zadanú sadu parametrov a nastavený model.

Parametre:

params: dictionary

Hodnoty jednotlivých parametrov modelu.

Výstup: float

Vypočítaná chyba štvorcov.

5.5.28 MassFun

`OCFit.OCFit.MassFun()`

Výpočet funkcie hmotnosti pre modely založené na LiTE. Ak bola použitá funkcia `FitMCMC`, vypočítajú sa aj chyby jej určenia. Získané hodnoty sa uložia do premennej `paramsMore`, v prípade chýb sa uložia do `paramsMore_err`.

Výstup: dictionary

Vypočítaná funkcia hmotnosti a jej chyba.

5.5.29 AbsoluteParam

OCFit.OCFit.**AbsoluteParam**(M, i = 90, M_err = 0, i_err = 0)

Výpočet hmotnosti tretieho telesa a veľkých poloosí pre modely založené na LiTE a Agolových modeloch. Ak bola použitá funkcia FitMCMC, vypočítajú sa aj chyby ich určenia. Získané hodnoty sa uložia do premennej `paramsMore`, v prípade chýb sa uložia do `paramsMore_err`.

Parametre:

M: float

Hmotnosť zákrytovej dvojhviezdy.

i: float

Sklon dráhy 3. telesa.

M_err: float

Chyba určenia hmotnosti dvojhviezdy.

i_err: float

Chyba určenia sklonu dráhy.

Výstup: dictionary

Vypočítané hodnoty parametrov a ich chyby.

5.5.30 ParamsApsidal

OCFit.OCFit.**ParamsApsidal**()

Výpočet ďalších parametrov sústavy pri použití modelu stáčania priamky apsíd. Ak bola použitá funkcia FitMCMC, vypočítajú sa aj chyby ich určenia. Získané hodnoty sa uložia do premennej `paramsMore`, v prípade chýb sa uložia do `paramsMore_err`.

Výstup: dictionary

Vypočítané hodnoty parametrov a ich chyby.

5.5.31 Plot

OCFit.OCFit.**Plot**(name = None, no_plot = 0, no_plot_err = 0, params = None, eps = False, oc_min = True, time_type = 'JD', offset = 2400000, trans = True, title = None, epoch = False, min_type = False, weight = None, trans_weight = False, model2 = False, with_res = False, bw = False, double_ax = False, legend = None, fig_size = None)

Vykreslenie pôvodných hodnôt O-C a krivky O-C na základe fitu. Je možné vykresliť modelovú krivku aj pre iné parametre. Parametre sa zadávajú do `params`, ide o dictionary s parametrami modelu. Prípadne je možné zobrazíť modelovú krivku na základe zadaných aj modelových parametrov, a to nastavením `model2 = True`. V tom prípade je zadaný model červený a fitovaný zelený.

Graf je možné uložiť do súboru `png` (prípadne aj do `eps`) alebo ho len zobraziť. Veľmi odletujúce body alebo body s veľkou chybou sa dajú odstrániť z grafu. Hodnoty O-C sa môžu prepočítať na minúty alebo sa použijú pôvodné hodnoty v dňoch. V menovke x-ovej možno nastaviť rôzny typ používaného juliánskeho dátumu. Dátum je možné pretransformovať odčítaním zadaného offsetu. Ak bol dátum už naozaj preškálovaný, je potrebné zakázať transformáciu x-ových hodnôt (`trans = False`), ale hodnotu použitého posunu je vhodné zadať kvôli výpisu v menovke osi. Taktiež je možné vykresliť x-ovú os v epoche. V prípade, ak sú k dispozícii O-C primárnych aj sekundárnych míním, je ich možné rozlíšiť pomocou parametra `min_type = True`. Primárne minimá sú znázornené plným krúžkom a sekundárne prázdny. Ak sú k dispozícii váhy bodov, dajú sa zvýrazniť aj na grafe ich nastavením do `weight`. Váhy sú rozdelené do kategórii: 0-3, 3-5, 5-8, 8-10. V prípade, že zadané váhy nie sú z tohto intervalu, je potrebné ich preškálovanie zadaním `trans_weight = True`. Ak boli na začiatku nastavené chyby dát a zadajú sa váhy do `weight`, vykreslí sa graf na základe zadaných váh bez errorbarov.

Parametre:

name: string, voliteľné

Názov súboru (bez prípony), do ktorého sa uloží graf.

no_plot: integer, voliteľné

Počet najviac odletujúcich bodov, ktoré sa v grafe nevykreslia.

no_plot_err: integer, voliteľné

Počet bodov s najväčšou zadanou chybou, ktoré sa v grafe nevykreslia.

params: dictionary, voliteľné

Parametre modelu.

eps: boolean, voliteľné

Uloženie grafu aj vo formáte `eps`.

oc_min: boolean, voliteľné

Hodnoty O-C previesť z dní na minúty.

time_type: string, voliteľné

Formát dátumu (JD, HJD, BJD).

offset: float, voliteľné

Posun dátumu kvôli skráteniu hodnôt na x-ovej osi.

trans: boolean, voliteľné

Pretransformovať aj x-ové hodnoty o offset.

title: string, voliteľné

Názov grafu.

epoch: boolean, voliteľné

x-ová os v epoche.

min_type: boolean, voliteľné

Rozlišovanie primárneho a sekundárneho minima.

weight: numpy.array, voliteľné

Váhy bodov.

trans_weight: boolean, voliteľné

Preškáľovanie váh.

model2: boolean, voliteľné

Vykreslenie modelovej krivky pre zadané aj fitované parametre.

bw: boolean, voliteľné

Čierno-biely graf.

double_ax: boolean, voliteľné

Dve x-ové osi - čas + epocha.

with_res: boolean, voliteľné

Spoločný graf aj s rezíduami.

legend: list, voliteľné

Zoznam menoviek pre legendu (zadajte "", ak sa nemá zobrazovať menovka; 2. model zadaný cez **params** je posledný).

fig_size: tuple, voliteľné

Iná veľkosť grafu.

5.5.32 PlotRes

OCFit.OCFit.**PlotRes**(name = None, no_plot = 0, no_plot_err = 0, params = None, eps = False, oc_min = True, time_type = 'JD', offset = 2400000, trans = True, title = None, epoch = False, min_type = False, weight = None, trans_weight = False, bw = False, double_ax=False, fig_size = None)

Vykreslenie rozdielu medzi pôvodnými hodnotami O-C a O-C spočítanými na základe fitu. Je možné vykresliť rozdiel pre modelovú krivku spočítanú aj pre iné parametre. Parametre sa zadávajú do **params**, ide o dictionary s parametrami modelu.

Graf je možné uložiť do súboru **png** (prípadne aj do **eps**) alebo ho len zobraziť. Veľmi odletujúce body alebo body s veľkou chybou sa dajú odstrániť z grafu. Hodnoty O-C sa môžu prepočítať na minúty alebo sa použijú pôvodné hodnoty v dňoch. V menovke x-ovej možno nastaviť rôzny typ používaného juliánskeho dátumu. Dátum je možné pretransformovať odčítaním zadaného offsetu. Ak bol dátum už naozaj preškáľovaný, je potrebné zakázať transformáciu x-ových hodnôt (**trans** = False), ale hodnotu použitého posunu je vhodné zadať kvôli výpisu v menovke osi. Taktiež je možné vykresliť x-ovú os v epoche. V prípade, ak sú k dispozícii O-C primárnych aj sekundárnych miním, je ich možné rozlíšiť pomocou parametra **min_type** = True.

Primárne minimá sú znázornené plným krúžkom a sekundárne prázdny. Ak sú k dispozícii váhy bodov, dajú sa zvýrazniť aj na grafe ich nastavením do `weight`. Váhy sú rozdelené do kategórii: 0-3, 3-5, 5-8, 8-10. V prípade, že zadané váhy nie sú z tohto intervalu, je potrebné ich preškálovanie zadaním `trans_weight = True`. Ak boli na začiatku nastavené chyby dát a zadajú sa váhy do `weight`, vykreslí sa graf na základe zadaných váh bez errorbarov.

Parametre:

name: string, voliteľné

Názov súboru (bez prípony), do ktorého sa uloží graf.

no_plot: integer, voliteľné

Počet najviac odletujúcich bodov, ktoré sa v grafe nevykreslia.

no_plot_err: integer, voliteľné

Počet bodov s najväčšou zadanou chybou, ktoré sa v grafe nevykreslia.

params: dictionary, voliteľné

Parametre modelu.

eps: boolean, voliteľné

Uloženie grafu aj vo formáte `eps`.

oc_min: boolean, voliteľné

Hodnoty O-C previesť z dní na minúty.

time_type: string, voliteľné

Formát dátumu (JD, HJD, BJD).

offset: float, voliteľné

Posun dátumu kvôli skráteniu hodnôt na x-ovej osi.

trans: boolean, voliteľné

Pretransformovať aj x-ové hodnoty o offset.

title: string, voliteľné

Názov grafu.

epoch: boolean, voliteľné

x-ová os v epoche.

min_type: boolean, voliteľné

Rozlišovanie primárneho a sekundárneho minima.

weight: numpy.array, voliteľné

Váhy bodov.

trans_weight: boolean, voliteľné

Preškáľovanie váh.

bw: boolean, voliteľné

Čierno-biely graf.

double_ax: boolean, voliteľné

Dve x-ové osi - čas + epocha.

fig_size: tuple, voliteľné

Iná veľkosť grafu.

5.5.33 Save

OCFit.OCFit.**Save**(path)

Uloženie parametrov triedy OCFit do súboru. Na opätovné načítanie je možné použiť triedu OCFitLoad alebo funkciu Load.

Parametre:

path: string

Názov súbor, do ktorého sa uložia parametre triedy.

5.5.34 Load

OCFit.OCFit.**Load**(path)

Načítanie parametrov triedy OCFit uložených funkciou Save zo súboru. Na načítanie je možné použiť aj triedu OCFitLoad, ktorej jediný vstupný názov súboru. Obsahuje všetky funkcie triedy OCFit.

Parametre:

path: string

Názov súbor s uloženými parametrami.

5.5.35 SaveModel

OCFit.OCFit.**SaveModel**(name, E_min = None, E_max = None, n = 1000, params = None, t0 = None, P = None)

Uloženie modelovej krivky O-C spočítanými na základe fitu do súboru. Je možné vypočítať modelovú krivku aj pre iné parametre. Parametre sa zadávajú do **params**, ide o dictionary s parametrami modelu. Tvar súboru je: čas minima, epocha, modelové O-C .

Parametre:

name: string

Názov výstupného súboru.

E_min: float, voliteľné

Minimálna epocha (ak nezadaná, použije sa minimálna epocha zadaných dát).

E_max: float, voliteľné

Maximálna epocha (ak nezadaná, použije sa maximálna epocha zadaných dát).

n: integer, voliteľné

Počet bodov na výpočet modelových O-C.

params: dictionary, voliteľné

Parametre modelu.

t0: float, voliteľné

Čas základného minima (potrebné zadať, ak t0 nie je v modeli).

P: float, voliteľné

Periódka systému (potrebné zadať, ak t0 nie je v modeli).

5.5.36 SaveRes

OCFit.OCFit.**SaveRes**(name, params = None, t0 = None, P = None, weight = None)

Uloženie reziduí medzi pôvodnými hodnotami O-C a O-C spočítanými na základe fitu do súboru. Je možné vypočítať rozdiel pre modelovú krivku spočítanú aj pre iné parametre. Parametre sa zadávajú do **param**, ide o dictionary s parametrami modelu. Tvar súboru je: čas minima O, epocha, nová hodnota O-C (reziduá), chyba (ak bola zadaná) alebo váha (ak bola zadaná).

Parametre:

name: string

Názov výstupného súboru.

params: dictionary, voliteľné

Parametre modelu.

t0: float, voliteľné

Čas základného minima (potrebné zadať, ak t0 nie je v modeli).

P: float, voliteľné

Periódka systému (potrebné zadať, ak t0 nie je v modeli).

weight: numpy.array, voliteľné

Váhy bodov.

5.6 Ukážka

```
from OCFit import OCFit, FitLinear
import numpy as np

#vygenerovanie dat
E = np.arange(0, 100, 1) #epocha
#simulovanie napozorovanych casov LiTE s amplitudou 72 min. a periodou 1125 d
P = 15
t0 = 1540
t = P*E + t0 + 0.05*np.sin(2*np.pi/(5*P)*E)
    + np.random.normal(scale = 0.01, size = E.shape)
err = 0.01*np.ones(E.shape) #chyby urcenia casov

#pouzitie FitLinear na vypočet O-C
#inicializacia s pouzitim odhadu (povodnych hodnot) perody a efemeridy
#pri dlhsom casovom rozsahu merani je potrebne zadat periodu, ktora je blizka
#spravnej hodnote, inak dojde k pomiesaniu primarnych a sekundarnych minim
lin = FitLinear(t, t0, P, err = err)
oc = lin.oc #O-C na zaklade povodnej perody a efemeridy

#inicializacia triedy OCFit a O-C vypocitaniami pomocou FitLinear
fit=OCFit(t,oc,err = err)
fit.Epoch(t0,P) #vypocet epochy

#nastavenie modelu a parametrov pre fitovanie
fit.model='LiTE3'
fit.fit_params=['a_sin_i3', 'e3', 'w3', 't03', 'P3']
fit.limits={'a_sin_i3': [5, 10], 'e3': [0, 1], 'w3': [0, 2*np.pi],
            't03': [1540, 3000], 'P3': [900, 1300]}
fit.steps={'a_sin_i3': 1e-2, 'e3': 1e-2, 'w3': 1e-2, 't03': 10, 'P3': 10}

#fitovanie pomocou GA bez vypisu priebehu fitovania
fit.FitGA(100,100,visible=False)
#sumar vysledkov po GA
fit.Summary()

#fitovanie pomocou MC bez vypisu priebehu fitovania
fit.FitMCMC(1e3,visible=False)
#sumar vysledkov po MC
fit.Summary()

#vykreslenie grafu
#graf povodnych O-C s fitom, bez transformovanie hodnot na x-ovej osi,
#spolu s reziduami a osov aj v epoche
fit.Plot(trans = False,with_res=True,double_ax=True)
```

6 OCFitLoad

Trieda na načítanie parametrov triedy OCFit uložených jej funkciou Save zo súboru. Obsahuje všetky funkcie triedy OCFit.

6.1 Inicializácia

OCFit.**OCFitLoad**(path)

Na inicializáciu triedy je potrebné zadať iba názov súboru, v ktorom sú uložené parametre triedy OCFit.

Parametre:

path: string

Názov súbor s uloženými parametrami.

Literatúra

Agol, E. et al., 2005, MNRAS, 359, 567

Brooks, S. et al. 2011. Handbook of Markov Chain Monte Carlo. CRC Press.

Gajdoš, P., Parimucha, Š., 2019, OEJV, 197, 71

Geweke J., 1992, in Bernardo J. M. et al., eds, Evaluating the accuracy of sampling-based approaches to calculating posterior moments. Clarendon Press, Oxford, p. 169

Giménez, A., Bastero, M., 1995, ApSS, 226, 99

Hartmann, A. T., Rieger, H. 2002. Optimization Algorithms in Physics. Berlin: Wiley-VCH.

Irwin, J. B., 1952, ApJ, 116, 211

Markley, F. L., 1995, CeMDA, 63, 101

Mikulášek, Z., Zejda, M., 2013, Úvod do studia proměnných hvězd, munipress

Odell, A. W., Gooding, R. H., 1986, CeMec, 38, 307

Patil, A., Huard, D., Fonnesbeck, C. J. 2010. J Stat Softw., 35, 1.

Razali, N. M., Geraghty, J. 2011. Proc. of the World Congress on Eng., II, 1134.

Weise, T. 2011. Global Optimization Algorithm - Theory and Application. 3rd Ed. Germany: online. <http://www.it-weise.de/projects/bookNew.pdf>.