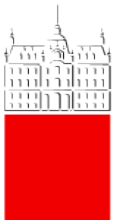
Univerza *v Ljubljani*
Fakulteta za *matematiko in fiziko*



Bayesianska regresija

6. naloga pri Opazovalni Astrofiziki

Avtor: Marko Urbanč (28191096)
Predavatelj: prof. dr. Janez Kos

3.9.2023

Kazalo

1 Uvod

Do sedaj smo se tekom izobraževanja v glavnem srečevali z Klasičnim oz. Frequentističnim pristopom k statistiki. Razlika je v tem kako se pri vsaki uporablja repeticija. V Frequentističnem pristopu se uporablja repeticija v smislu, da se izvede poskus večkrat. S tem smo torej fiksirali parametre, ki nas zanimajo in zdaj repliciramo podatke. V Bayesianskem pristopu pa se uporablja repeticija v smislu, da so podatki fiksni in da posledično repliciramo parametre. To je še posebaj uporabno, recimo v astrofizikalnem kontekstu, kjer ne moramo replicirati poskusov, lahko pa repliciramo parametre.

1.1 Bayesov izrek

Osnova Bayesovega pristopa je Bayesov izrek, ki pravi

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}\,, \tag{1}$$

kjer je $P(A|B)$ pogojna verjetnost, da se zgodi A ob pogojih B. $P(B|A)$ je pogojna verjetnost, da se zgodi B ob pogojih A. $P(A)$ in $P(B)$ sta verjetnosti, da se zgodi A in da se zgodi B. Načeloma naj bi bralec to spoznal že tekom gimnazijske matematike. Iz tega izreka sledi Bayesianska inferenca, ki je osnova Bayesovega pristopa k statistiki. Pravzaprav so stvari samo malo preimenovane poglejmo še enkrat Bayesov izrek, tokrat v kontekstu inference

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)}\,. \tag{2}$$

Tukaj H pravimo **hipoteza** katere verjetnost nas zanima. Nanj vplivajo naši zbrani podatki. Običajno imamo v bistvu hipotez več in te medseboj tekmujejo v tem, katera je bolj verjetna. Naši novi podatki so E. Pravimo jim tudi **dokazi** (angl. evidence). $P(H)$ je **a priori** verjetnost (angl. **prior** probability) in je verjetnost za hipotezo H preden upoštevamo naše nove podatke E. $P(E|H)$ je **verjetnost podatkov** (angl. likelihood), ki predstavlja verjetnost, da se pojavijo naši podatki E ob pogojih H. $P(H|E)$ je **a posteriori** verjetnost (angl. **posterior** probability). To je tisto kar nas zanima.

1.2 Bayesianska regresija

Pristop Bayesianske regresije deluje tako, da imamo neko hipotezo H in nek set podatkov. Naredimo model za naše podatke tako, da postavimo priorje za parametre modela, glede na to kaj že vemo o danem problemu. Nato definiramo porazdelitev za verjetnost podatkov, kateri pokažemo del podatkov. S tem izračunamo posteriorje za parametre modela. Obstaja rek, ki pravi

Yesterday's posterior is today's prior.

Točno to naredimo. Za nove priorje postavimo posteriorje iz prejšnjega koraka. Nato ponovimo postopek dokler ne dobimo zadovoljivega rezultata. Ko smo zadovoljni s parametri lahko izračunamo tudi posterior porazdelitev.

2 Naloga

Naloga je sestavljena iz dveh delov. Prvi del od nas zahteva da pri danih podatkih za razliko med pričakovano in izmerjeno svetlostjo supernov v odvisnosti od rdečega oremika naredimo model, kjer meritve parametriziramo z ortogonalnimi polinomi. Potem naloga poda dve hipotezi za model. Prva ima predpis

$$\Delta m(z) = 0\,, \tag{3}$$

in predstavlja hipotezo, da naše vesolje **ni** prazno. Druga hipoteza ima predpis

$$\Delta m(z) = -0.27z - 0.14z^2\,, \tag{4}$$

in predstavlja hipotezo, da je naše vesolje kritično in polno barionske snovi.

Drugi del od nas zahteva, da na dane podatke meritev porazdelitve števila zvezd po masah z uporabo Bayesianskega pristopa prileţemo krivuljo

$$\frac{\mathrm{d}N}{\mathrm{d}m} = Am^{-\alpha}\left[1 - e^{\left(-\frac{m}{m_p}\right)^{\beta}}\right]\,, \tag{5}$$

kjer je A skalirna konstanta, α parameter, ki opisuje obnašanje masne funkcije pri velikih masah in favorizira $\alpha = 2.42$, β parameter, ki opisuje obnašanje masne funkcije pri majhnih masah in m_p parameter, kjer se en režim masne funkcije spremeni v drugo oz. kjer ima masna funkcija maksimum.

3 Opis reševanja

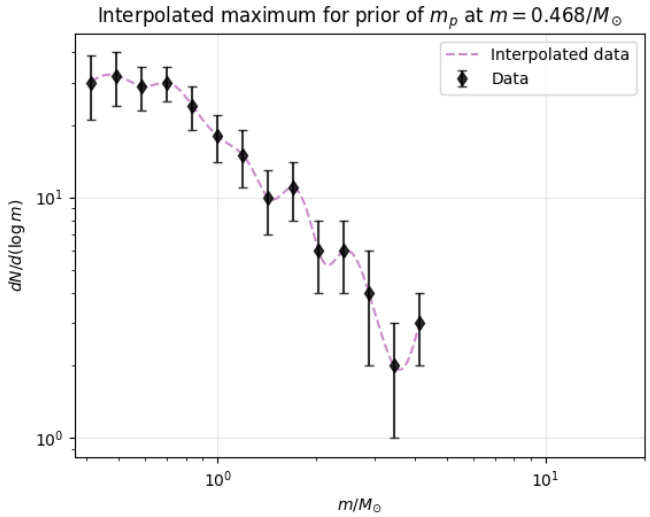
Po predolgem premoru od zadnje rešene naloge, sem se reševanja lotil v Pythonu. Za reševanje sem uporabil knjižnico `pymc` [1] (ne `pymc3`, ki je stara, ampak v5.7.2). Ta je namenjena ravno Bayesianski regresiji. V dodatno pomoč so bile še knjižnice `numpy`, `scipy` in `matplotlib`.
Za zelo lep prikaz parametrov in njihovih porazdelitev sem uporabil knjižnico `corner`.

3.1 Supernove

Zdelo se mi je smiselno, da naprej napravim model, ki se bo v Bayezianskem smislu najbolje prilegal dobljenim podatkom in ga nato testiram proti hipotezama. Smiselno bi bilo modele primerjati glede na njihov **Bayesov faktor**, ki je definiran kot

$$B_{12} = \frac{P(D|M_1)}{P(D|M_2)}\,, \tag{6}$$

kjer je $P(D|M_i)$ verjetnost podatkov D ob pogojih modela M_i. Tako da sem šel v to smer. Za osnovni sem izbral polinom 2. reda. Izkazalo se je, da je bilo to čisto zadovoljivo. Njegove parametre sem inicializiral kot normalno porazdeljene slučajne spremenljivke s povprečjem 0 in varianco 1. Verjetnost podatkov sem izračunal kot normalno porazdelitev s povprečjem modela (torej naš polinom), opazovanimi podatki in variancami opazovanih podatkov. Se mi zdi, da je bilo



Slika 1: Interpolirani podatki in maksimum.

ugibanje, da so vsi parametri normalno porazdeljeni kar upravičeno.

Ustvaril sem še en model, zdaj za teoretično krivuljo (3). V njem sem naredil samo en prior, ki je bil normalno porazdeljen s povprečjem 0 in varianco 1. Ta prior direktno predstavlja vrednost izmerjene razlike med magnitudami, tako da je to tudi naše središče za normalno porazdelitev verjetnosti podatkov. Zopet tudi za opazovanimi podatki in napakami opazovanih podatkov.

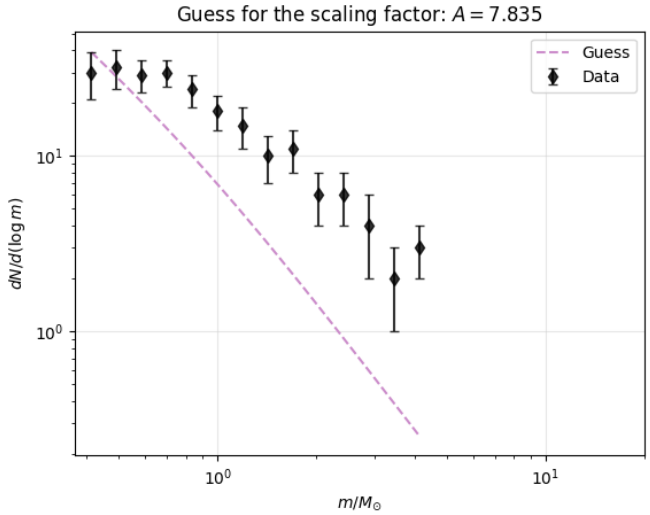
Za konec sem naredil še model za teoretično krivuljo (4). Ampak sem potem ugotovil, da sem malo smešen, saj je to v bistvu že isti model kot osnovni model. V tem primeru se mi je zdelo smiselno računati residume.

Modele sem vzorčil vsakega z 4000 vzorci na jedro z 4 jedri. Vzorčil sem tudi posterior porazdelitev in logaritem verjetnosti podatkov iz katerega sem izračunal Bayesov faktor.

3.2 Masna funkcija

Za masno funkcijo sem naredil model, ki je bil sestavljen iz treh priorjev, α pa sem fiksiral na 2.42. Ker nisem imel nobene ideje kako bi lahko določil priorje za A, β in m_p, sem si narisal nekaj pomožnih grafov. Na grafu 1 sem vrednosti med točkami interpoliral s kvadratnim polinomom in nato izračunal maksimum. S tem bi lahko vsaj približno uganil, kje se nahaja m_p, ker naj bi bil na mestu maksimuma.
Za skalirni faktor A sem vzel razmerje

$$A_{\text{guess}} = \left(\frac{m_{\max}}{N_{\max}}\right)^{-\beta}, \tag{7}$$



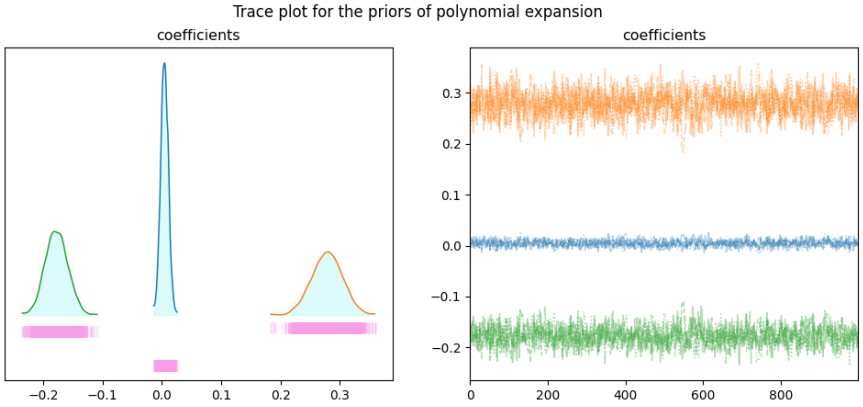
Slika 2: Ugibanje skalirnega faktorja.

pravzaprav brez kakšnega dobrega razloga. Poskušal sem vrednosti in kombinacje parametrov, dokler ni približno zadelo skalo.
Za β nisem imel ideje, kako bi ga lahko na pameten način ocenil, ampak sem sklepal, da mora biti verjetno nekako kot α. Vse parametre sem inicializiral kot enakomerno porazdeljene naključne spremenljivke. Območja za vsako sem poskušal oceniti iz grafov in tudi iterativno izbolševal z večkratnim zaganjanjem modela. Izkazalo se je, da je bilo najbolše če sem po vzorčenju tega modela ustvaril nov, posodobljen model, kjer sem uporabil posteriorje iz prejšnjega modela kot nove priorje in tokrat privzel, da so normalno porazdeljeni okoli svojih srednjih vrednosti. Rezultati so predstavljeni v naslednjem poglavju.

4 Rezultati

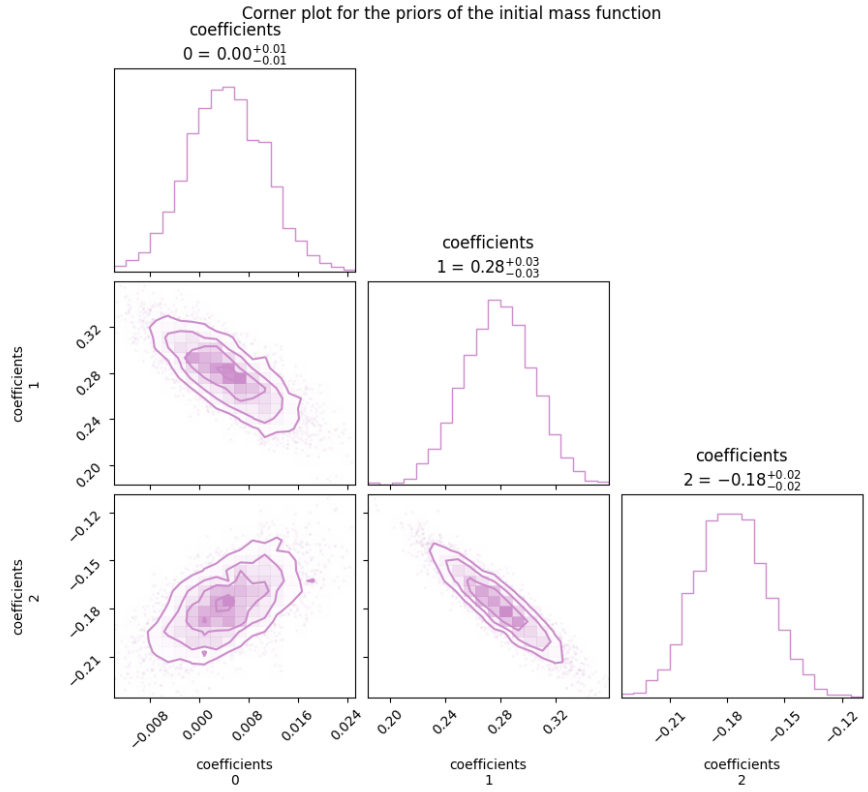
4.1 Supernove

Prvo si je smiselno pogledati, kako so parametri modela tekom vzorčenja konvergirali. To je prikazano na sliki 3.



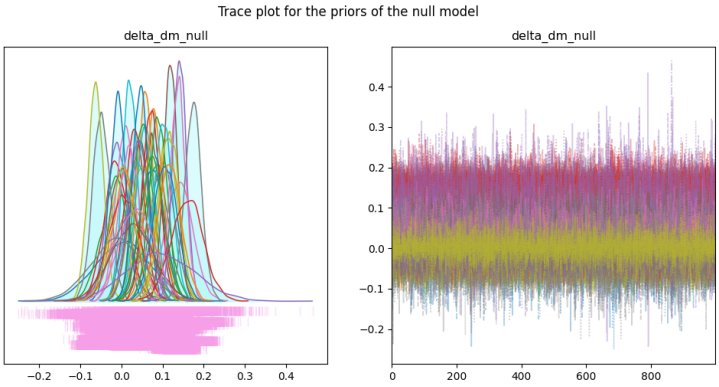
Slika 3: Konvergenca parametrov modela.

Vidimo, da so se parametri lepo ustalili. Če pogledamo še porazdelitve parametrov, ki so prikazane na sliki 4, vidimo, da so porazdelitve lepo simetrične in da so tudi medsebojno neodvisne.



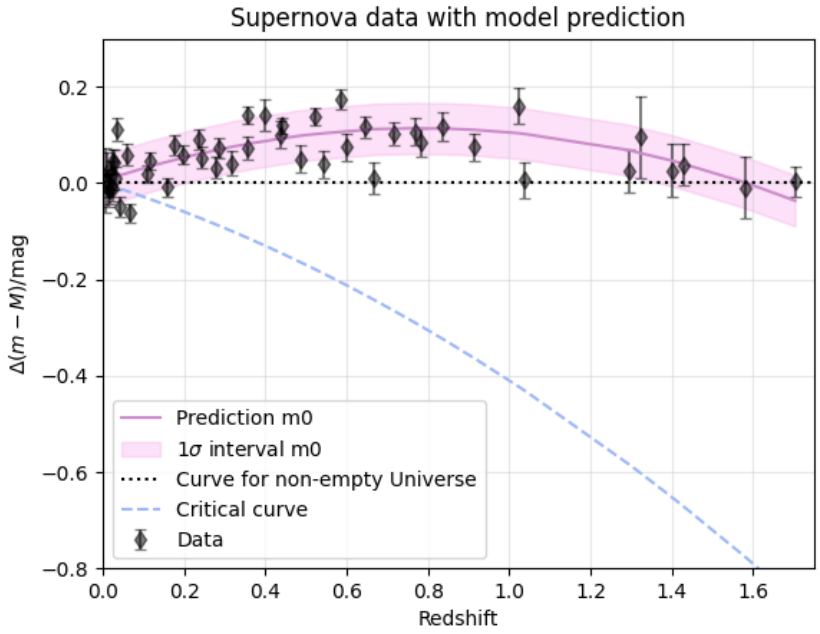
Slika 4: Porazdelitve parametrov modela.

Poglejmo si še enake stvari za model, ki predstavlja hipotezo (3). Na sliki 5



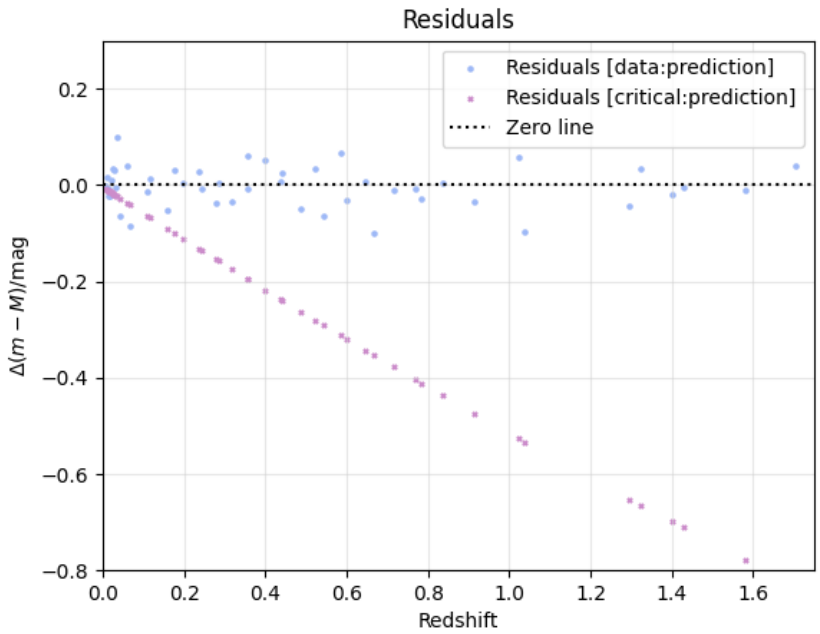
Slika 5: Konvergenca parametrov modela.

Kot pričakovano, dobimo množico gaussovk centriranih okoli 0. Porazdelitve parametrov nima smisla prikazovati, saj so vse enake. Okay, poglejmo si potem kaj napoveduje naš model. Na sliki 6 je prikazan model in podatki. Hkrati pa tudi obe hipotezi. Model se zelo lepo prilega podatkom.



Slika 6: Napoved modela.

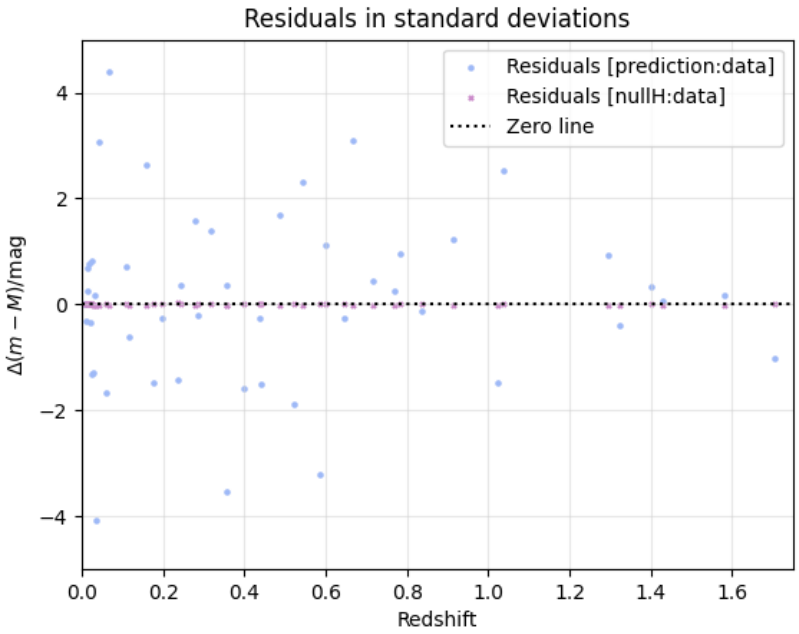
Poglejmo si še residume. Na sliki 7 so prikazani residumi med podatki in napovedjo našega osnovnega modela in residumi za model med to isto napovedjo in krivuljo, ki predstavlja hipotezo (4).



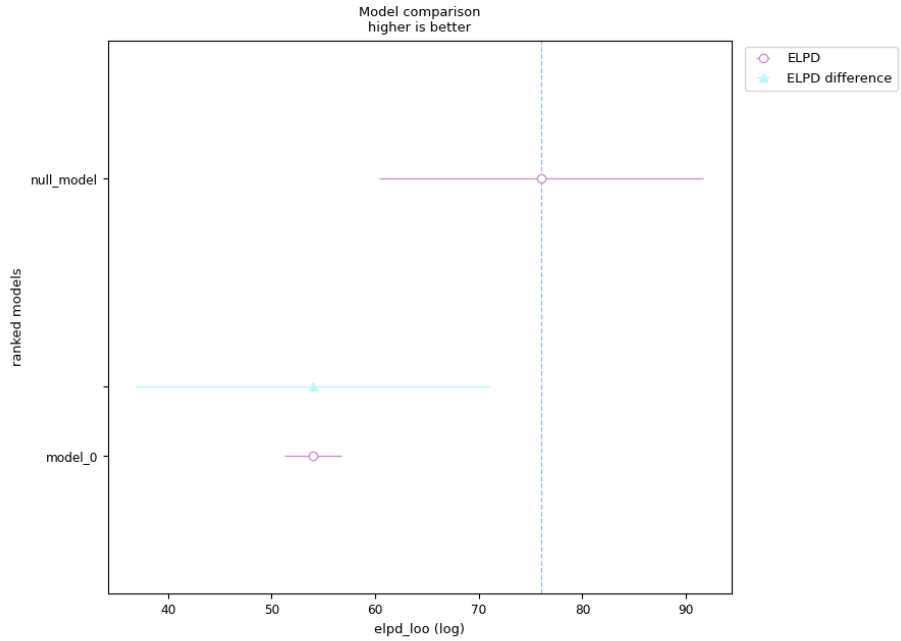
Slika 7: Residumi za osnovni model in hipotezo (4).

Vidimo, da so residumi za hipotezo (4) veliko večji kot za osnovni model. To je tudi pričakovano, saj je hipoteza (4) zelo slaba napoved za naše podatke. Naša osnoven model se pa zelo lepo prilega podatkom in residumi so ustrezno majhni. Smiselno bi bilo pogledati še za naš null model, torej tisti, ki predstavlja hipotezo (3). Na sliki 8 so prikazani residumi za ta model. Modre točke predstavljajo residume za napoved našega osnovnega modela (glede na podatke), vijolčne pa za napoved null modela (glede na podatke). Vidimo, da so residumi za null model praktično enaki 0. To je seveda pričakovano. Null model je namreč tisti, ki predstavlja hipotezo, da naše vesolje ni prazno in se mi zdi, da ima ta hipoteza že veliko dokazov za seboj. Hkrati lahko tudi vidimo, da krivulja (3) najbolje opisuje naše podatke. To je vidno že s prostim očesom.

Kot zanimivost sem narisal še primerjavo modelov glede na njihovo pričakovano vrednost logoritma napovedne gostote (angl. expected log predictive density, ELPD). To je prikazano na sliki 9.
Vidimo, da je ELPD za null model največji, kar je seveda pričakovano. To je še en dokaz, da je null model najboljši model za naše podatke. Čisto na koncu pa še z uporabo enačbe (6) izračunamo Bayesov faktor med null modelom in osnovnim modelom. Ta je enak $B = 700 \pm 20$. To je zelo velik Bayesov faktor, kar pomeni, da lahko zelo veliko zanesljivostjo rečemo, da naše vesolje ni prazno.



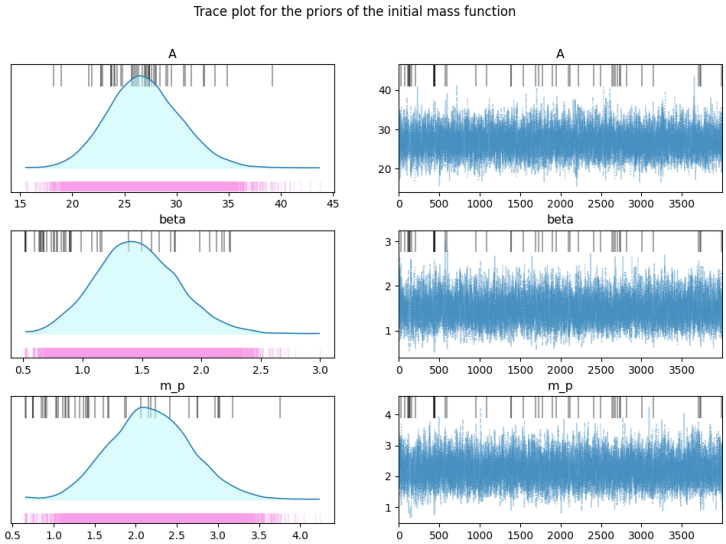
Slika 8: Residumi za null model.



Slika 9: ELPD primerjalni plot za modele.

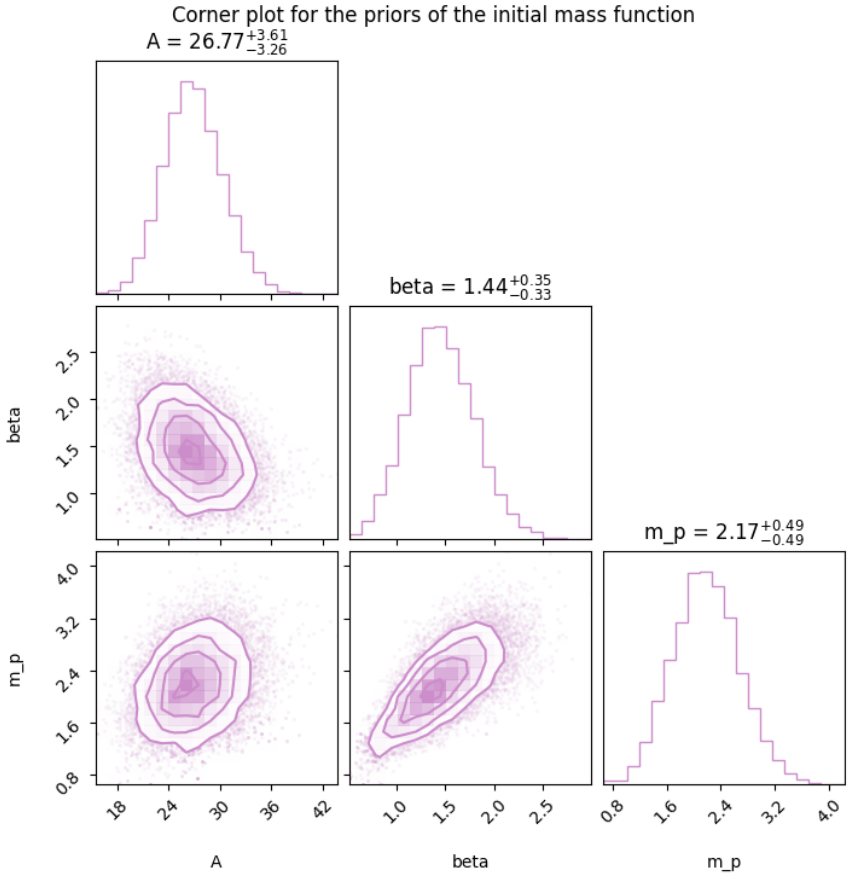
4.2 Masna funkcija

Poglejmo si še rezultate za masno funkcijo. Na sliki 10 je prikazana konvergenca parametrov modela. Vidimo, da so se parametri razmeroma lepo ustalili.



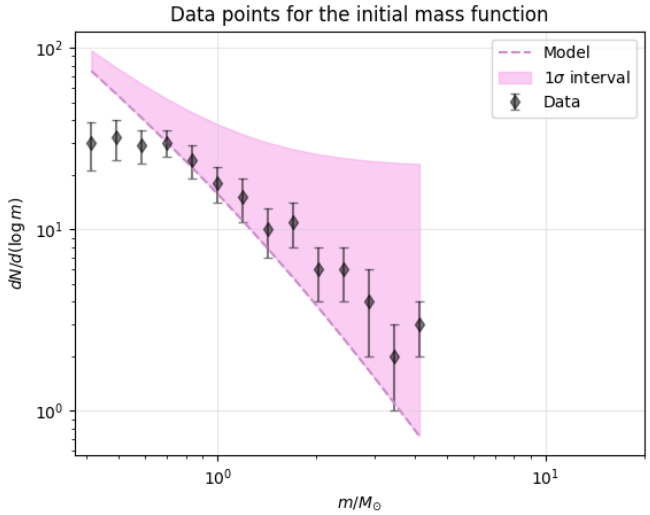
Slika 10: Konvergenca parametrov modela.

Poglejmo si še porazdelitve parametrov. Te so prikazane na sliki 11.



Slika 11: Porazdelitve parametrov modela.

Recimo, da so porazdelitve še kar simetrične in neodvisne. S tem dobimo družino masnih funkcij, ki so prikazane na sliki 12.



Slika 12: Masna funkcija.

5 Komentarji in izboljšave

Trenutno praktično speedrunam zaključek študija, tako da je definitivno prostor za izboljšave.
Ena super ideja bi bila si pogledati kakšne običajne statistične teste za dobljene krivulje. Recimo kakšen χ^2 test. Definitely bi tudi posvetil več časa 2. nalogi in ugoravljanju boljšega načina za to "družino masnih funkcij", ker trenutno 1 σ območje zelo naraste ko se masa veča.

Knjižnica `pymc` je zelo uporabna, ampak ima nekoliko learning curve in je potrebno veliko branja dokumentacije, da ugotoviš kako stvari delujejo. Da ne govorimo o tem kakšne nočne more se dogajajo sicer behind the scenes z podatkovnimi tipi (np.ndarray, pandas.DataFrame, xarray.DataArray, pytensor.tensor.TensorVariable, ...) in kako se te stvari med seboj ne nujno vedno marajo.

No funny today. Only work.

Literatura

[1] Oriol Abril-Pla, Virgile Andreani, Colin Carroll, Larry Dong, Christopher J. Fonnesbeck, Maxim Kochurov, Ravin Kumar, Junpeng Lao, Christian C. Luhmann, Osvaldo A. Martin, and et al. Pymc: A modern, and comprehensive probabilistic programming framework in python. *PeerJ Computer Science*, 9, 2023.