

Calcolo per l'astronomia - Progetto 1: stima della costante di Hubble partendo dalle curve di luce di 19 stelle cefeidi osservate in altrettante galassie

Michele Costa

Novembre 2020

0 Step iniziale: indagine dei dati a disposizione

0.1 Scopo e dati utilizzati

Nello step iniziale è stata effettuata un'indagine dei dati a disposizione, è stato cioè letto il file *gal_vel.txt* e sono stati salvati il numero delle cefeidi e i nomi delle loro galassie di appartenenza. Per evitare di dover aprire il file in un momento successivo sono stati salvati anche gli altri dati contenuti in esso, ovvero la velocità di recessione di ogni galassia e il relativo errore.

0.2 Subroutines e algoritmi

Contarighe

Per leggere qualsiasi file è necessario conoscerne in precedenza il numero delle righe. La subroutine *contarighe* prende come parametri il percorso del file da leggere *catname*, il numero di caratteri dello stesso ovvero l'intero *lencat* e l'intero *nrighe*, al quale verrà assegnato il numero di righe.

1 Primo step: calcolo periodi di pulsazione e magnitudine apparente media

1.1 Scopo e dati utilizzati

Lo step 1 ha come scopo finale il calcolo del periodo di pulsazione della cefeide presa in esame, della sua magnitudine apparente media e dei relativi errori. Si ha a disposizione, per ogni stella, un file riportante la magnitudine apparente misurata in diversi giorni giuliani.

1.2 Analisi del problema fisico

Per calcolare il periodo di pulsazione della cefeide si osserva la sua curva di luce: se si è in grado di stabilire la coordinata temporale di tutti i punti aventi uno stesso valore della magnitudine apparente si può facilmente risalire al periodo di pulsazione della stella, vista la natura periodica delle curve di luce. Dato che da una singola curva è possibile estrarre più valori del periodo viene calcolato il valor medio dello stesso, che ha come errore associato l'errore standard. Per quanto riguarda invece la magnitudine apparente media si è calcolato il valore medio di questa grandezza in un numero intero di periodi. Anche in questo caso l'errore associato sarà l'errore standard.

1.3 Subroutines e algoritmi

Per ottenere i risultati richiesti è stato realizzato un ciclo che per ogni cefeide legge i dati riguardanti le osservazioni, li ordina temporalmente, effettua un'interpolazione tramite spline cubica e infine calcola il periodo, la magnitudine media e i relativi errori. Si è fatto ricorso alle seguenti subroutines:

Ordinamento

Dopo aver letto i dati riguardanti la curva di luce della singola cefeide è stato necessario ordinarli temporalmente, senza però compromettere l'associazione giorno giuliano–magnitudine apparente. La subroutine *ordinamento* ha come parametri l'intero *ndati* e i vettori reali *giorni* e *magapp* e si occupa di ordinare entrambi i vettori in modo crescente basandosi però unicamente sui valori del primo.

Spline

È stato poi necessario procedere con l'interpolazione: i file forniti infatti presentano un numero insufficiente di dati per poter calcolare con precisione il periodo di pulsazione. Si è ricorsi quindi a un'interpolazione tramite spline cubica per stimare l'andamento della curva di luce anche in punti dove non si possiedono dati sperimentali. È stata pertanto realizzata la subroutine *spline*: essa ha come parametri i vettori reali *x* e *y*, l'intero *ndati* e i vettori reali *x_f_i* e *f_i* e si occupa di stimare l'andamento della curva di luce anche dove non si hanno dati sperimentali. L'algoritmo implementato per la spline cubica funziona nel seguente modo: vengono inizialmente costruiti la matrice tridiagonale dei coefficienti delle derivate seconde, avente dimensione $[ndati-2, ndati-2]$ e il vettore dei termini noti, di dimensione $[ndati-2]$.

$$\begin{bmatrix} r_1 & g_1 & 0 & \cdot & \cdot \\ e_2 & r_2 & g_2 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & 0 & e_{n-2} & r_{n-2} & g_{n-2} \\ \cdot & \cdot & 0 & e_{n-1} & r_{n-1} \end{bmatrix} \begin{bmatrix} c_1 \\ \cdot \\ \cdot \\ \cdot \\ c_{n-1} \end{bmatrix}$$

$$\begin{cases} e_i = x_i - x_{i-1} \\ r_i = 2 \cdot (x_{i+1} - x_{i-1}) \\ g_i = x_{i+1} - x_i \\ c_i = \frac{6 \cdot (y_{i+1} - y_i)}{x_{i+1} - x_i} + \frac{6 \cdot (y_{i-1} - y_i)}{x_i - x_{i-1}} \end{cases}$$

La matrice viene risolta attraverso l'algoritmo di Thomas: essendo una matrice tridiagonale questo metodo può essere implementato ed è molto più efficiente a confronto con la classica eliminazione di Gauss in termini di numero di operazioni eseguite. Il vettore delle soluzioni (le derivate seconde), viene poi completato aggiungendo i valori 0 all'inizio e alla fine. Viene poi realizzato il vettore contenente le ascisse per le quali cercare il valore della funzione interpolata (*x_f_i*), ordinate che vanno dal primo valore del giorno per il quale si hanno valori sperimentali all'ultimo, con passo pari a $\frac{1}{100}$ di giorno giuliano e, attraverso la formula dimostrata a lezione, viene calcolato il valore della magnitudine apparente in essi (*f_i*). Si ha, per un generico $x_f_i \mid x_f_i \in]x_{i-1}, x_i[$:

$$f_i(x_f_i) = \frac{sol_{i-1}}{6(x_i - x_{i-1})}(x_i - x_f_i)^3 + \frac{sol_i}{6(x_i - x_{i-1})}(x_f_i - x_{i-1})^3 + \\ + \left[\frac{y_{i-1}}{(x_i - x_{i-1})} - \frac{sol_{i-1}(x_i - x_{i-1})}{6} \right] (x_i - x_f_i) + \left[\frac{y_i}{(x_i - x_{i-1})} - \frac{sol_i(x_i - x_{i-1})}{6} \right] (x_f_i - x_{i-1})$$

Thomas

La subroutine *thomas* è stata realizzata per risolvere le matrici tridiagonali. Essa ha come parametri la matrice a coefficienti reali *mat* di dimensione $[n,n]$, il vettore dei termini noti *tnot* di dimensione $[n]$, l'intero *n* e il vettore delle soluzioni *sol*. Il sistema viene risolto in due step: nel primo la matrice viene trasformata in una matrice triangolare superiore

$$\begin{bmatrix} a_1 & b_1 & 0 & \cdot & \cdot \\ c_2 & a_2 & b_2 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & 0 & c_{n-2} & a_{n-2} & b_{n-2} \\ \cdot & \cdot & 0 & c_{n-1} & a_{n-1} \end{bmatrix} \begin{bmatrix} d_1 \\ \cdot \\ \cdot \\ \cdot \\ d_{n-1} \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & b'_1 & 0 & \cdot & \cdot \\ 0 & 1 & b'_2 & 0 & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & 0 & 1 & b'_{n-2} \\ \cdot & \cdot & \cdot & 0 & 1 \end{bmatrix} \begin{bmatrix} d'_1 \\ \cdot \\ \cdot \\ \cdot \\ d'_{n-1} \end{bmatrix}$$

$$\begin{cases} b'_i = \frac{b_i}{a_i} & i = 1 \\ b'_i = \frac{b_i}{a_i - c_i \cdot b'_{i-1}} & i = 2, \dots, n-1 \end{cases} \quad \begin{cases} d'_i = \frac{d_i}{a_i} & i = 1 \\ d'_i = \frac{d_i - c_i \cdot d'_{i-1}}{a_i - c_i \cdot b'_{i-1}} & i = 2, \dots, n-1 \end{cases}$$

nel secondo vengono ottenute le soluzioni del sistema attraverso la back-substitution.

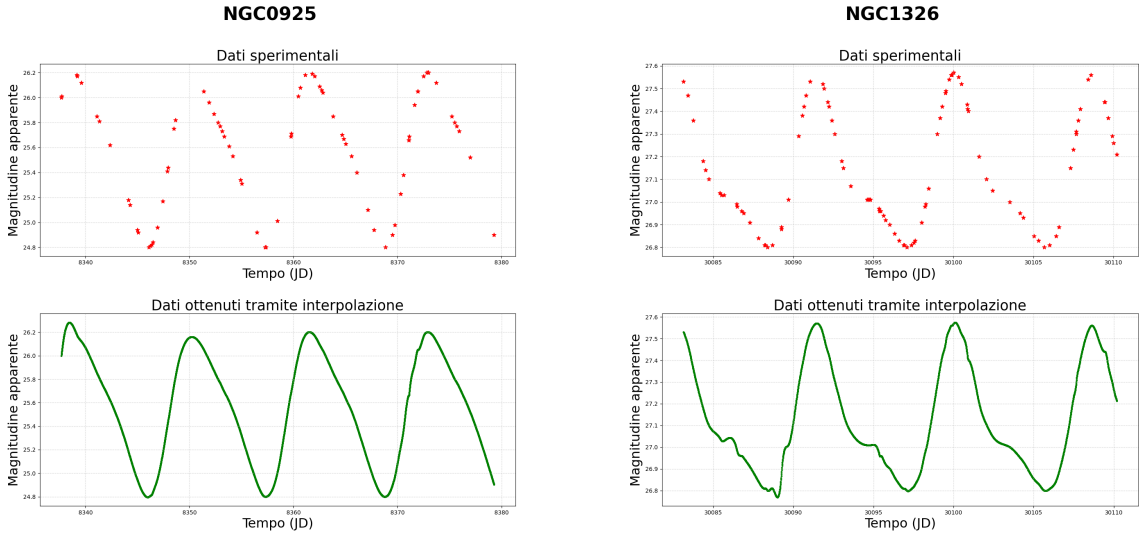
$$\begin{cases} sol_i = d'_i & i = n \\ b'_i = d'_i - b'_i \cdot sol_{i+1} & i = n-1, \dots, 1 \end{cases}$$

Trovaperiodiimag

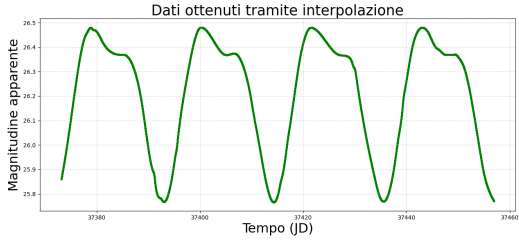
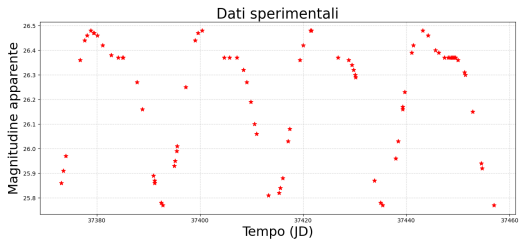
La funzione *trovaperiodiimag* ha il compito di calcolare il periodo di pulsazione della cefeide, la sua magnitudine apparente media e gli errori da associare a queste due grandezze. Essa ha come parametri i vettori reali x e y di dimensione $[np]$, l'intero np e la stringa *nome*. Per ottenere i risultati richiesti è stato implementato un algoritmo che trova i punti di intersezione della curva di luce con una retta orizzontale passante per *medy*, ovvero il valore medio tra il massimo assoluto e il minimo assoluto della curva. Ciò è stato fatto cercando tutti quei valori della magnitudine apparente minori di *medy* tali per cui il valore presente nella posizione immediatamente successiva del vettore fosse maggiore di *medy*. La coordinata temporale di questi punti viene poi salvata in un apposito vettore e i periodi calcolati come differenza tra i valori presenti in posizione n e quelli in posizione $n+2$ (con n che varia tra 1 e *intersez-2*, dove *intersez* è la dimensione del vettore). Siccome da una curva di luce vengono estratti *intersez-2* valori del periodo ne si calcola la media aritmetica e vi si associa l'errore standard, ovvero $\frac{\sigma}{\sqrt{\text{intersez}-2}}$. Per trovare la magnitudine apparente media invece si è trovato l'intervallo temporale corrispondente al maggior numero di periodi estrapolabili dal grafico e si è calcolato il valore medio di questa grandezza all'interno dello stesso. Anche qui l'errore associato è l'errore standard. I risultati ottenuti vengono poi salvati nel file *periodi.e_magapp.txt*

1.4 Risultati

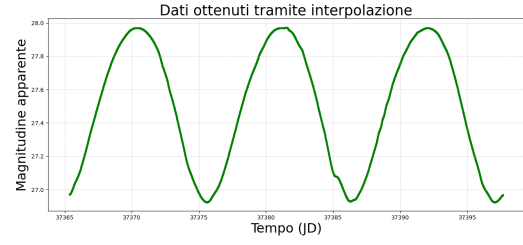
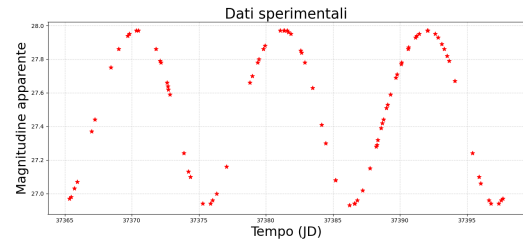
Si ottiene dalla spline un numero molto maggiore di dati che ci permette di stimare con maggior precisione l'andamento della curva di luce: di seguito vengono riportate le immagini che confrontano le curve di luce di ogni cefeide prima e dopo l'interpolazione. Questi grafici identificano in definitiva le stelle prese in esame come delle stelle cefeidi, ovvero stelle che possono essere usate come candele standard.



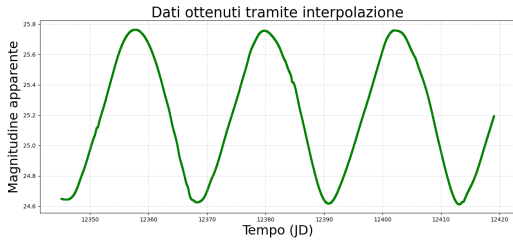
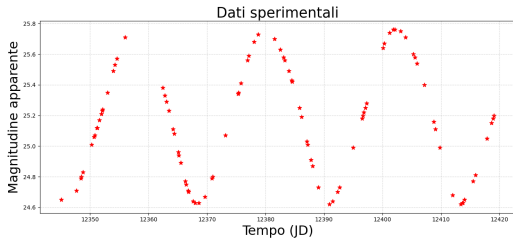
NGC1365



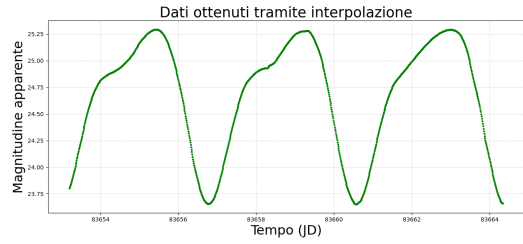
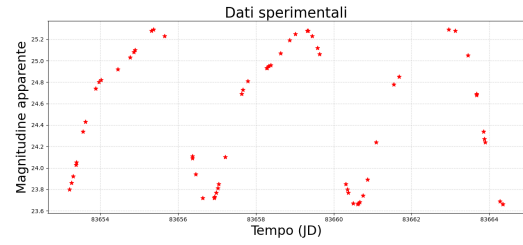
NGC1425



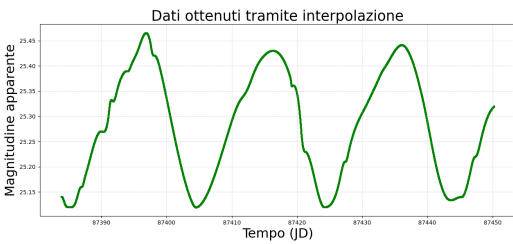
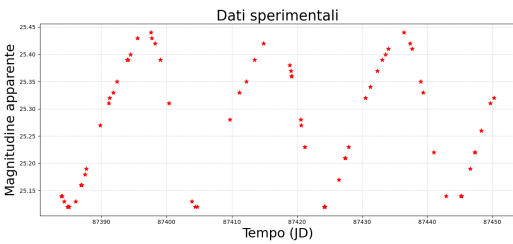
NGC2090



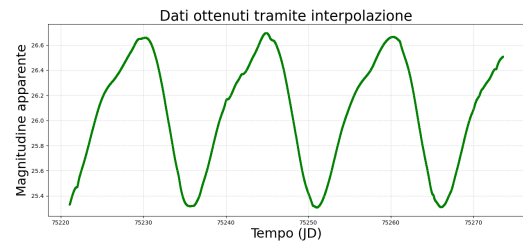
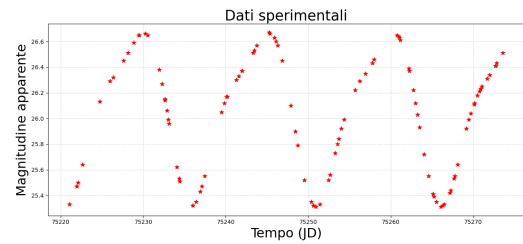
NGC2403



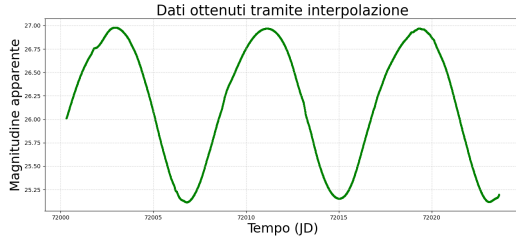
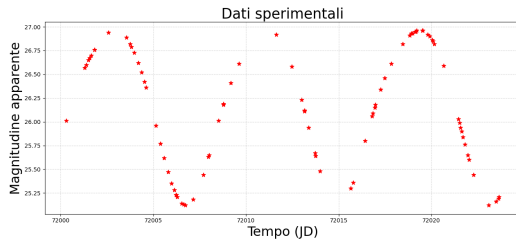
NGC2541



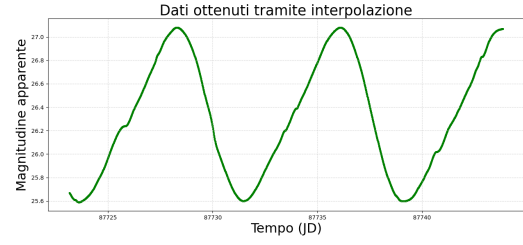
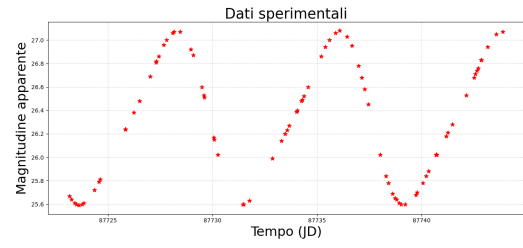
NGC3198



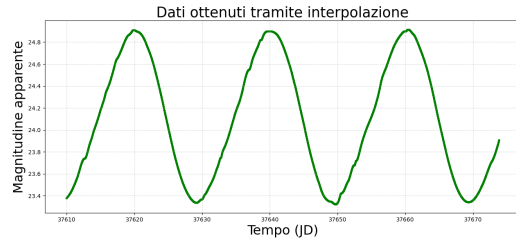
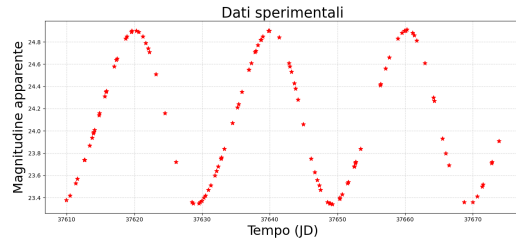
NGC3351



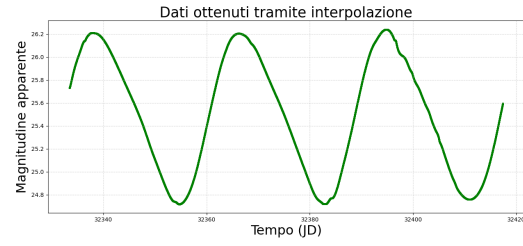
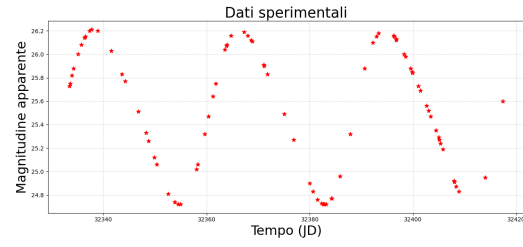
NGC3368



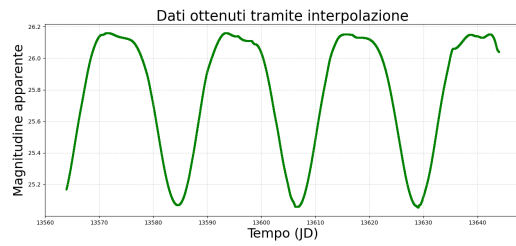
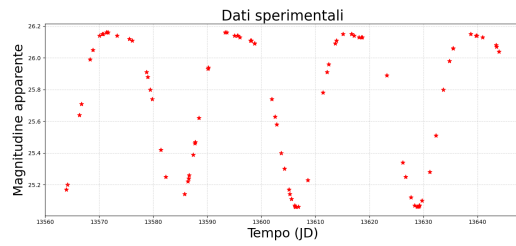
NGC3621



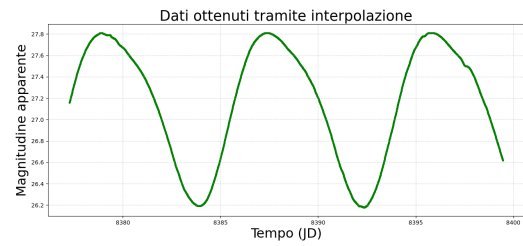
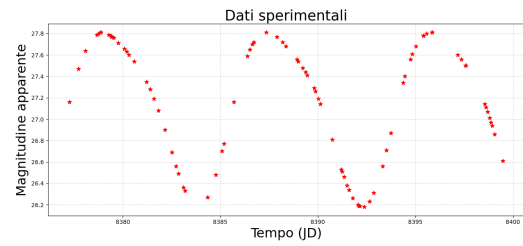
NGC4321



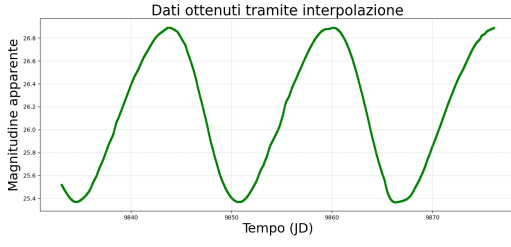
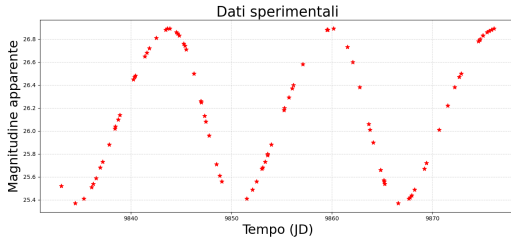
NGC4496



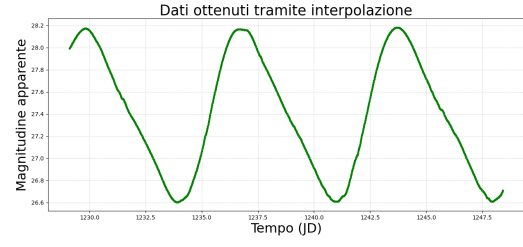
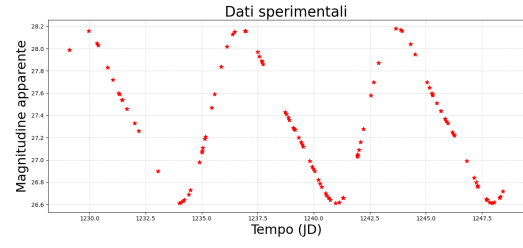
NGC4535



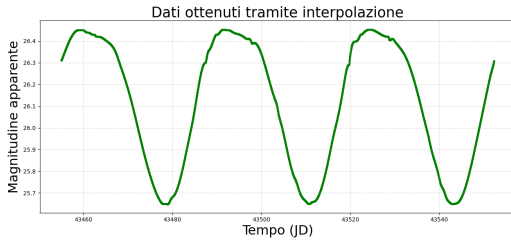
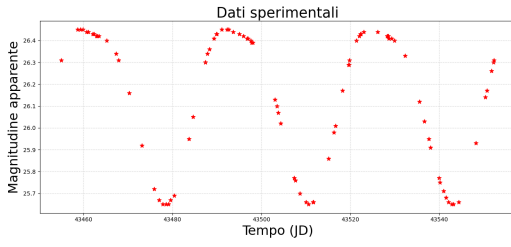
NGC4536



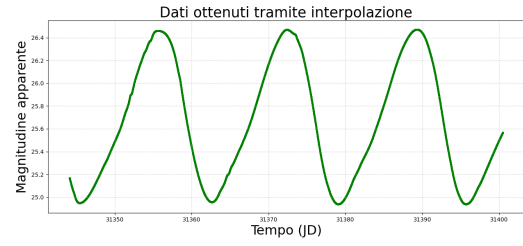
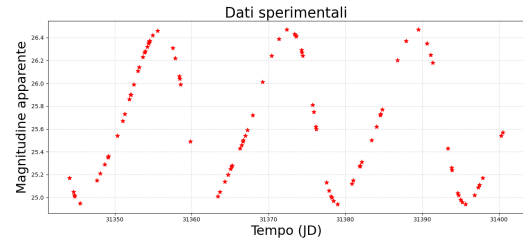
NGC4548



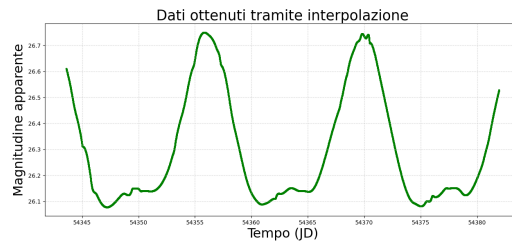
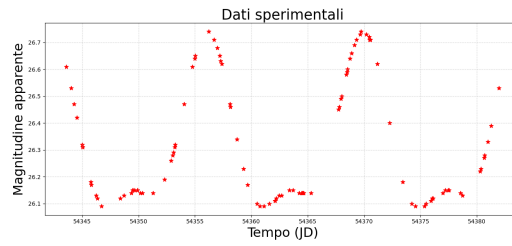
NGC4639



NGC4725



NGC7331



Come ci si aspettava, le immagini delle curve di luce a seguito della spline sono delle funzioni periodiche, i cui massimi e minimi hanno valori molto simili tra loro. Vengono riportati anche, nella seguente tabella, i periodi di pulsazione di ogni cefeide e la magnitudine apparente media.

Nome	Periodo (gg)	m_v media
NGC0925	11.37 ± 0.01	25.537 ± 0.022
NGC1326	8.64 ± 0.01	27.114 ± 0.024
NGC1365	21.34 ± 0.09	26.204 ± 0.017
NGC1425	10.74 ± 0.02	27.499 ± 0.025
NGC2090	22.42 ± 0.02	25.224 ± 0.017
NGC2403	3.80 ± 0.01	24.648 ± 0.085
NGC2541	19.47 ± 0.12	25.292 ± 0.004
NGC3198	15.13 ± 0.01	26.066 ± 0.027
NGC3351	8.21 ± 0.01	26.141 ± 0.047
NGC3368	7.74 ± 0.03	26.322 ± 0.029
NGC3621	20.13 ± 0.04	24.120 ± 0.021
NGC4321	28.19 ± 0.02	25.495 ± 0.014
NGC4496	22.17 ± 0.01	25.748 ± 0.026
NGC4535	8.43 ± 0.01	27.098 ± 0.023
NGC4536	16.16 ± 0.05	26.155 ± 0.025
NGC4548	6.99 ± 0.01	27.393 ± 0.034
NGC4639	32.42 ± 0.03	26.199 ± 0.006
NGC4725	16.67 ± 0.02	25.707 ± 0.020
NGC7331	13.82 ± 0.03	26.327 ± 0.020

2 Secondo step: derivazione dei coefficienti C_1 e C_2 della relazione periodo-magnitudine assoluta

2.1 Scopo e dati utilizzati

Lo step 2 ha come scopo il calcolo dei coefficienti C_1 e C_2 che compaiono nella formula che lega il periodo di pulsazione di una cefeide alla sua magnitudine assoluta. Sono stati utilizzati i dati presenti nel file *ceph_catalog.txt*, ovvero il periodo di pulsazione e la magnitudine assoluta media di un gruppo di 29 cefeidi.

2.2 Analisi del problema fisico

La relazione tra periodo di pulsazione e magnitudine apparente è:

$$M = C_1 \cdot \text{Log}(P) + C_2$$

Se si ha a disposizione un set di cefeidi per il quale M e $\text{Log}(P)$ sono noti è possibile, attraverso un fit lineare, ricavare i coefficienti C_1 e C_2 della retta che meglio approssima l'andamento dei dati in nostro possesso.

2.3 Subroutines e algoritmi utilizzati

Fitlin

Per effettuare questa operazione è stata realizzata la subroutine *fitlin*: essa ha come parametri l'intero nd , i vettori reali x , y e *coeff* e si occupa perl'appunto di calcolare i sopracitati coefficienti. Per prima cosa vengono costruiti la matrice e il vettore dei termini noti associati al sistema lineare:

$$\begin{cases} N \cdot C_2 + \sum x \cdot C_1 = \sum y \\ \sum x \cdot C_2 + \sum x^2 \cdot C_1 = \sum x \cdot y \end{cases}$$

Ovvero:

$$\begin{bmatrix} N & \sum x \\ \sum x & \sum x^2 \end{bmatrix} \begin{bmatrix} C_2 \\ C_1 \end{bmatrix} = \begin{bmatrix} \sum y \\ \sum x \cdot y \end{bmatrix}$$

Poi, una volta ottenuti questi elementi, si chiama la subroutine *gauss* per risolvere il sistema.

Gauss

La subroutine *gauss* ha il compito di risolvere i sistemi lineari: essa ha come parametri l'intero nd , la matrice a coefficienti reali a di dimensioni $[nd, nd]$ e i vettori reali c e sol di dimensioni $[nd]$. In primo luogo la matrice data in input viene trasformata in una matrice triangolare superiore attraverso l'algoritmo della forward elimination (viene modificato ovviamente anche il vettore dei termini noti):

$$\begin{bmatrix} a_{1,1} & \cdot & a_{1,nd} \\ \cdot & \cdot & \cdot \\ a_{nd,1} & \cdot & a_{nd,nd} \end{bmatrix} \begin{bmatrix} c_1 \\ \cdot \\ c_{nd} \end{bmatrix} \Rightarrow \begin{bmatrix} a_{1,1} & \cdot & a_{1,nd} \\ \cdot & \cdot & \cdot \\ 0 & \cdot & a'_{nd,nd} \end{bmatrix} \begin{bmatrix} c_1 \\ \cdot \\ c'_{nd} \end{bmatrix}$$

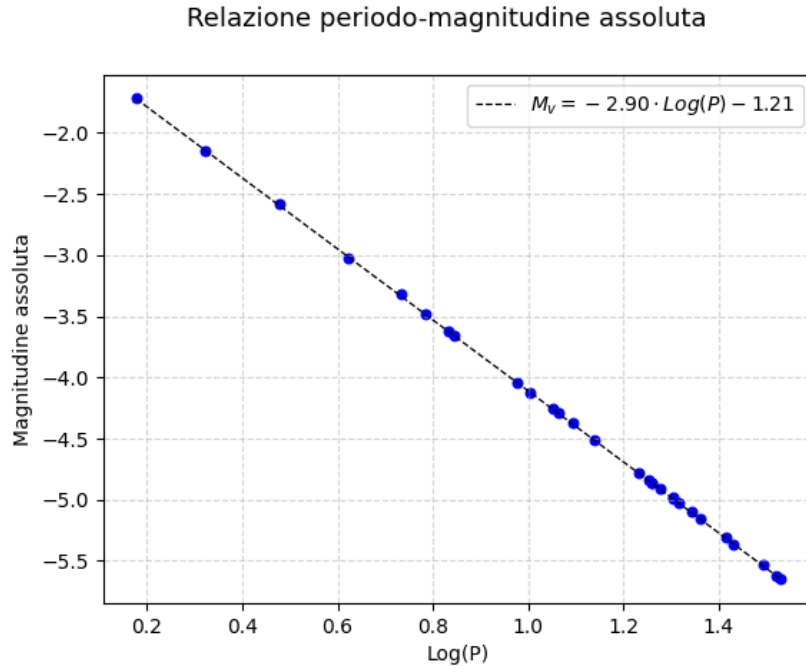
Si passa poi alla back-substitution per trovare le soluzioni del sistema lineare:

$$sol_{nd} = \frac{c'_{nd}}{a'_{nd,nd}} \quad sol_i = \frac{c'_i - \sum_{j=i+1}^{nd} a'_{i,j} \cdot sol_j}{a'_{i,i}}$$

e si ottiene quindi il vettore delle soluzioni.

2.4 Risultati

I coefficienti trovati valgono rispettivamente $C_1 = -2.90$ e $C_2 = -1.21$



3 Terzo step: calcolo della distanza della cefeide

3.1 Scopo e dati utilizzati

Lo scopo finale di questo step è la derivazione della distanza delle cefeidi. Si ha a disposizione per ogni stella il suo periodo di pulsazione e la magnitudine apparente media salvati nel file *periodi.txt*, inoltre si conoscono i coefficienti C_1 e C_2 precedentemente calcolati.

3.2 Analisi del problema fisico

Conoscendo il periodo di pulsazione delle cefeidi e i coefficienti C_1 e C_2 è possibile ricavare la magnitudine assoluta di ognuna di esse attraverso la formula

$$M = C_1 \cdot \text{Log}(P) + C_2 \quad \Delta M = C_1 \cdot \frac{\Delta P}{P}$$

e quindi, conoscendo ora la loro magnitudine assoluta, si può ricavare la distanza attraverso la relazione:

$$d = 10^{0.2 \cdot (M-m)+1} \quad \Delta d = 0.2 \cdot \ln 10 \cdot 10^{0.2 \cdot (M-m)+1} \cdot (\Delta M + \Delta m)$$

3.3 Subroutines e algoritmi

Calcdist

La subroutine *calcdist* si occupa appunto di calcolare le distanze delle varie cefeidi: essa ha come parametri l'intero *ngal* e il vettore dei coefficienti reali *coeff*. In primo luogo vengono letti i dati contenuti nel file *periodi.txt*, in seguito vengono calcolati i logaritmi dei periodi, gli errori su di essi e attraverso la formula sopracitata le magnitudini assolute col relativo errore. Trovate queste ultime è possibile calcolare le distanze e le loro incertezze, valori che vengono salvati sul file *magass_e_dist.txt*

3.4 Risultati

Si ottengono così le magnitudini assolute e le distanze di ogni cefeide, che vengono riportate nella seguente tabella:

Nome	M_v media	Distanza (Mpc)
NGC0925	-4.267 ± 0.003	9.14 ± 0.10
NGC1326	-3.922 ± 0.003	16.11 ± 0.20
NGC1365	-5.060 ± 0.012	17.90 ± 0.24
NGC1425	-4.196 ± 0.005	21.82 ± 0.31
NGC2090	-5.123 ± 0.003	11.73 ± 0.11
NGC2403	-2.887 ± 0.008	3.21 ± 0.14
NGC2541	-4.945 ± 0.018	11.15 ± 0.11
NGC3198	-4.627 ± 0.002	13.76 ± 0.18
NGC3351	-3.857 ± 0.004	9.99 ± 0.23
NGC3368	-3.783 ± 0.011	10.50 ± 0.19
NGC3621	-4.987 ± 0.006	6.63 ± 0.08
NGC4321	-5.411 ± 0.002	15.18 ± 0.11
NGC4496	-5.108 ± 0.001	14.83 ± 0.19
NGC4535	-3.891 ± 0.003	15.77 ± 0.19
NGC4536	-4.710 ± 0.009	14.89 ± 0.23
NGC4548	-3.655 ± 0.004	16.20 ± 0.28
NGC4639	-5.587 ± 0.003	21.94 ± 0.09
NGC4725	-4.749 ± 0.003	12.34 ± 0.13
NGC7331	-4.513 ± 0.006	14.72 ± 0.18

4 Quarto step: stima della costante di Hubble

4.1 Scopo e dati utilizzati

Lo scopo di questo step è il calcolo della costante di Hubble. Si hanno a disposizione le distanze delle 19 cefeidi calcolate nello step precedente e salvate nel file *magass_e_dist.txt* e le velocità di recessione delle galassie a cui appartengono, oltre che l'errore associato ad entrambe.

4.2 Analisi del problema fisico

La legge di Hubble-Lemaître sancisce che:

$$H_0 = \frac{v_{rec}}{d} \quad \Delta H_0 = \left(\frac{\Delta v_{rec}}{v_{rec}} + \frac{\Delta d}{d} \right) \cdot H_0$$

É quindi possibile per ogni cefeide calcolare la costante di Hubble associata, per poi fare una media pesata sugli errori. Si avrà quindi:

$$\overline{H_0} = \frac{\sum w \cdot H_0}{\sum w} \quad \Delta \overline{H_0} = \sqrt{\frac{1}{\sum w}} \quad \text{dove } w = \frac{1}{(\Delta H_0)^2}$$

4.3 Subroutines e algoritmi

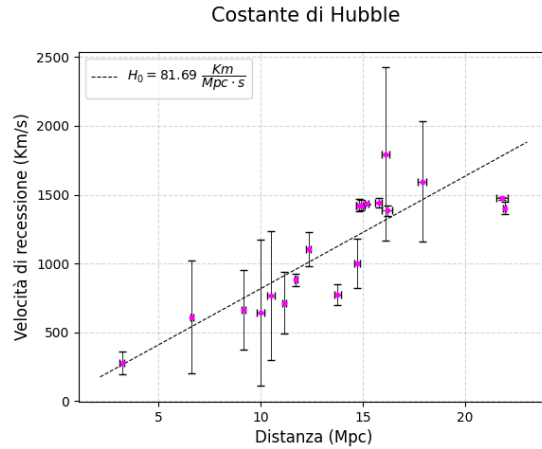
Calchubble

Per eseguire le operazioni discusse nello step precedente è stata realizzata la funzione *calchubble*: questa funzione ha come parametri l'intero *ngal*, i vettori reali *vrec* e *errvrec* e si occupa di calcolare la media pesata sugli errori della costante di Hubble e la relativa incertezza e salvarle rispettivamente in *H0* e *EH0*. Per prima cosa viene letto il file dove sono state salvate le distanze e i loro errori, in seguito viene calcolata, per ogni cefeide, la costante di Hubble associata con il suo errore e salvata nel file *cost_hubble_e_err.txt*. Una volta fatto ciò è possibile ricavare la media pesata delle stesse e l'errore associato attraverso la formula sopra riportata.

4.4 Risultati

Si riportano nella tabella sottostante i valori di H_0 ottenuti per ogni stella:

Nome	$H_0 \left(\frac{Km}{Mpc \cdot s} \right)$
NGC0925	72.6 ± 32.5
NGC1326	111.4 ± 40.5
NGC1365	89.1 ± 25.6
NGC1425	67.5 ± 1.3
NGC2090	75.2 ± 4.5
NGC2403	86.6 ± 30.3
NGC2541	64.0 ± 20.5
NGC3198	56.1 ± 6.3
NGC3351	64.3 ± 54.8
NGC3368	73.1 ± 46.1
NGC3621	91.9 ± 63.1
NGC4321	94.4 ± 0.9
NGC4496	96.0 ± 4.1
NGC4535	91.6 ± 3.3
NGC4536	95.6 ± 4.2
NGC4548	85.4 ± 3.8
NGC4639	63.9 ± 2.3
NGC4725	89.4 ± 10.8
NGC7331	67.9 ± 13.0
media pesata	81.7 ± 0.6



5 Quinto step: stima dei parametri Ω_M e Ω_Λ in universi dalla geometria euclidea, sferica e iperbolica

5.1 Scopo e dati utilizzati

In questo step si vogliono calcolare le combinazioni dei parametri Ω_M e Ω_Λ che permettono di ottenere un'età dell'universo pari a 13.8 ± 0.1 Gy. Viene utilizzata la costante H_0 ricavata in precedenza.

5.2 Analisi del problema fisico

L'età dell'universo è ricavabile dalla relazione:

$$t_0 = \int_0^{t_0} \delta t = \frac{1}{H_0} \int_0^\infty \frac{\delta z}{(1+z) \cdot E(z)^{\frac{1}{2}}}$$

$$E(z) = \Omega_M(1+z)^3 + (1 - \Omega_M - \Omega_\Lambda)(1+z)^2 + \Omega_\Lambda$$

Da cui deriva:

$$t_0 \cdot H_0 = \int_0^\infty \frac{\delta z}{(1+z) \cdot E(z)^{\frac{1}{2}}}$$

$$t_0 \cdot H_0 = \int_0^1 \frac{\delta z}{(1+z) \cdot E(z)^{\frac{1}{2}}} + \int_0^1 \frac{1}{z^2} \frac{\delta z}{(1+\frac{1}{z}) \cdot E(\frac{1}{z})^{\frac{1}{2}}}$$

É stata operata questa trasformazione in quanto la formula originale richiede la risoluzione di un integrale improprio, che per poter essere stimato attraverso il calcolo numerico necessita di essere trasformato in integrale proprio. Una volta calcolati il membro di sinistra e la sua incertezza è possibile passare alla risoluzione dell'integrale. Sono quindi state accettate tutte le combinazioni di Ω_M e Ω_Λ tali per cui il valore del membro di destra fosse confrontabile con quello di sinistra (entro l'errore). Si distinguono in seguito tre casi:

- 1) $\Omega_M + \Omega_\Lambda = 1 \quad \Rightarrow \quad$ universo con geometria euclidea
- 2) $\Omega_M + \Omega_\Lambda < 1 \quad \Rightarrow \quad$ universo con geometria iperbolica
- 3) $\Omega_M + \Omega_\Lambda > 1 \quad \Rightarrow \quad$ universo con geometria sferica

5.3 Subroutines e algoritmi

Per affrontare questo problema si è proceduto nel seguente modo:

- 1) vengono effettuate le necessarie conversioni: l'età dell'universo viene espressa in secondi e la costante di Hubble in Hertz. I due risultati sono stati moltiplicati tra loro ed è stato propagato l'errore sul prodotto.
- 2) attraverso un ciclo sono stati forniti alla funzione *quadgauss* tutte le combinazioni possibili dei parametri Ω_M e Ω_Λ (i valori di questi parametri vanno da 0 a 1 e si è scelto di farli variare con passo di 0.01)
- 3) la funzione *quadgauss* calcola il valore dei due integrali presenti nella formula, valori che vengono in seguito sommati tra loro.
- 4) viene confrontato il risultato ottenuto con il valore di $t_0 \cdot H_0$: se la somma dei due integrali ricade entro l'errore su questa grandezza i valori di Ω_M e Ω_Λ vengono considerati accettabili e vengono salvati, a seconda del caso, in un appropriato file.

Quadgauss

Nel programma si è scelto di implementare come metodo risolutivo per gli integrali la quadratura di Gauss utilizzando la formula di Gauss-Legendre a 4 punti. La subroutine *quadgauss* ha come parametri le variabili reali f , a , b , Ω_{MegaM} , Ω_{MegaL} e la variabile *ris*. Dato che per ogni integrale vale:

$$I \simeq c_0 f(x_0) + c_1 f(x_1) + c_2 f(x_2) + c_3 f(x_3)$$

questa subroutine si occupa di determinare il valore dell'integrale calcolando in primis i coefficienti $c_0 \dots c_3$ e in seguito, operando l'opportuno cambio di variabile, i punti $x_0 \dots x_3$. Viene quindi calcolato infine l'integrale sommando i quattro termini tra loro.

Integranda1 e integranda2

Queste due funzioni sono completamente analoghe tra loro e si occupano di calcolare il valore della funzione che contengono. Esse prendono in input il punto in cui calcolare la funzione, ovvero il reale x e i due parametri reali Ω_M e Ω_Λ . La prima contiene la funzione integranda del primo termine, la seconda la funzione integranda del secondo. Queste due funzioni vengono utilizzate dalla subroutine *quadgauss*.

5.4 Risultati

Si ottengono, nei diversi casi, molteplici valori per i parametri Ω_M e Ω_Λ che vengono salvati nei file *univ-geom-euclid.txt*, *univ-geom-iperb.txt* e *univ-geom-sfer.txt*. Di seguito è graficato l'andamento dell'integrale e evidenziati i risultati accettabili:

