

UNIVERSITÀ DEGLI STUDI DI SALERNO

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE ED
ELETTRICA E MATEMATICA APPLICATA

Corso di Statistica Applicata

Analisi di Regressione di un dataset

Autor:
Adinolfi Teodoro
Amato Emilio
Bove Antonio
Ferrara Grazia

FISCIANO, 04/02/2022

INDICE

1 Premessa	4
1.1 I regressori	4
1.2 Nomenclatura	4
2 Analisi dei dati	5
2.1 Sintesi delle informazioni in forma tabellare	5
2.2 Istogrammi	7
2.3 Grafici a gradini	10
2.4 Indici di tendenza centrale (o di posizione)	12
2.5 Indici di dispersione	14
2.6 Box-Plot	15
3 Analisi delle relazioni tra le variabili	17
3.1 Analisi di correlazione	17
3.2 Scatter Plot	20
4 Definizione del modello statistico	22
4.1 Analisi delle relazioni tra le variabili tramite regressione polinomiale	22
4.2 Scelta del modello di regressione	22
4.3 Evoluzione dei parametri rilevanti	30
5 Stima dei parametri del modello	32
5.1 Metodo dei "Minimi Quadrati"	32
5.2 Stima degli intervalli di confidenza	34
5.3 Stima della v.a. errore	36
6 Analisi del modello	37
6.1 Calcolo del coefficiente di determinazione	37
6.2 Grafici diagnostici - Analisi dei residui	38
6.2.1 Verifica omoschedasticità	39
6.2.2 Verifica normalità	40
6.2.3 Verifica linearità	42
6.2.4 Verifica indipendenza	43

7 Convalida del modello	45
7.1 Scelta del modello tramite regressione stepwise	47
7.2 Confronto tra modelli	50
7.3 Conclusioni	51

CAPITOLO 1

PREMESSA

1.1 I regressori

Il dataset si presenta composto da 6 regressori

- **X1**: indice standardizzato e centrato relativo alla velocità della CPU
- **X2**: indice standardizzato e centrato relativo alla dimensione dell' HD
- **X3**: indice standardizzato e centrato relativo al numero di processi SW
- **X4**: indice standardizzato e centrato relativo all' aging SW
- **X5**: indice standardizzato e centrato relativo alle prestazioni della scheda audio
- **X6**: indice standardizzato e centrato relativo alle prestazioni della RAM

La variabile dipendente Y , è invece la variabile di output ed è relativa alle prestazioni SW del calcolatore.

1.2 Nomenclatura

Nelle pagine seguenti faremo riferimento ai diversi regressori mediante la seguente nomenclatura

- **X1**: cpu
- **X2**: hardDisk
- **X3**: processi
- **X4**: aging
- **X5**: scheda audio
- **X6**: ram

CAPITOLO 2

ANALISI DEI DATI

Il dataset fornito, si presenta come un insieme di variabili di tipo **quantitativo continuo**, ossia che assumono valori all'interno del campo dei numeri reali. La numerosità campionaria è $n = 100$. Per la rappresentazione delle variabili si è scelto di utilizzare le più comuni **tecniche di statistica descrittiva**. I risultati delle analisi verranno presentati nei successivi paragrafi.

2.1 Sintesi delle informazioni in forma tabellare

Dall'analisi delle informazioni contenute nel dataset, una volta classificate le variabili come quantitative continue, si è proceduto con la definizione in forma tabellare della distribuzione di frequenza e della distribuzione di frequenza cumulata. Per ottenere una distribuzione di frequenza per una variabile continua, è opportuno suddividere l'intervallo dei valori che la variabile può assumere (tra il minimo ed il massimo) in **classi**, ossia sotto-intervalli dell'intervalle di definizione. Si sono individuate le classi in accordo alla relazione empirica $k = 1 + 3.3 \times \log_{10}(N)$, si sono definiti i limiti superiori e inferiori ed il centro di classe, per ogni classe. Effettuate queste operazioni, si è proceduto con il calcolo delle varie distribuzioni di frequenza.

Tutti i dati ricavati, sono sintetizzati nelle seguenti tabelle.

Frequenze Aging					Frequenze Audio						
	Centro di classe	Absolute	Relative	Cumulate	Relative Cumulate		Centro di classe	Absolute	Relative	Cumulate	Relative Cumulate
[-1.73,-1.31)	-1.52	12	0.12	12	0.12	[-1.65,-1.25)	-1.45	14	0.14	14	0.14
[-1.31,-0.897)	-1.1	10	0.1	22	0.22	[-1.25,-0.851)	-1.05	13	0.13	27	0.27
[-0.897,-0.481)	-0.69	19	0.19	41	0.41	[-0.851,-0.449)	-0.65	8	0.08	35	0.35
[-0.481,-0.065)	-0.27	5	0.05	46	0.46	[-0.449,-0.0483)	-0.25	10	0.1	45	0.45
[-0.065,0.351)	0.14	13	0.13	59	0.59	[0.0483,0.353)	0.15	15	0.15	60	0.6
[0.351,0.767)	0.56	12	0.12	71	0.71	[0.353,0.754)	0.55	13	0.13	73	0.73
[0.767,1.18)	0.97	14	0.14	85	0.85	[0.754,1.16)	0.95	9	0.09	82	0.82
[1.18,1.6]	1.39	15	0.15	100	1	[1.16,1.56]	1.36	18	0.18	100	1

Frequenze CPU					Frequenze Hard Disk						
	Centro di classe	Absolute	Relative	Cumulate	Relative Cumulate		Centro di classe	Absolute	Relative	Cumulate	Relative Cumulate
[-1.66, -1.22)	-1.44	14	0.14	14	0.14	[-1.59, -1.16)	-1.38	16	0.16	16	0.16
[-1.22, -0.786)	-1	10	0.1	24	0.24	[-1.16, -0.738)	-0.95	12	0.12	28	0.28
[-0.786, -0.351)	-0.57	18	0.18	42	0.42	[-0.738, -0.313)	-0.53	13	0.13	41	0.41
[-0.351, 0.0841)	-0.13	11	0.11	53	0.53	[-0.313, 0.112)	-0.1	11	0.11	52	0.52
[0.0841, 0.519)	0.3	15	0.15	68	0.68	[0.112, 0.538)	0.32	17	0.17	69	0.69
[0.519, 0.954)	0.74	12	0.12	80	0.8	[0.538, 0.963)	0.75	10	0.1	79	0.79
[0.954, 1.39)	1.17	9	0.09	89	0.89	[0.963, 1.39)	1.18	11	0.11	90	0.9
[1.39, 1.82]	1.61	11	0.11	100	1	[1.39, 1.81]	1.6	10	0.1	100	1

Frequenze Processi					Frequenze RAM						
	Centro di classe	Absolute	Relative	Cumulate	Relative Cumulate		Centro di classe	Absolute	Relative	Cumulate	Relative Cumulate
[-1.72, -1.29)	-1.5	9	0.09	9	0.09	[-1.7, -1.3)	-1.5	12	0.12	12	0.12
[-1.29, -0.866)	-1.08	18	0.18	27	0.27	[-1.3, -0.893)	-1.09	11	0.11	23	0.23
[-0.866, -0.44)	-0.65	13	0.13	40	0.4	[-0.893, -0.488)	-0.69	15	0.15	38	0.38
[-0.44, -0.014)	-0.23	12	0.12	52	0.52	[-0.488, -0.0833)	-0.29	11	0.11	49	0.49
[-0.014, 0.412)	0.2	8	0.08	60	0.6	[0.0833, 0.321)	0.12	8	0.08	57	0.57
[0.412, 0.838)	0.62	11	0.11	71	0.71	[0.321, 0.726)	0.52	9	0.09	66	0.66
[0.838, 1.26)	1.05	16	0.16	87	0.87	[0.726, 1.13)	0.93	17	0.17	83	0.83
[1.26, 1.69]	1.48	13	0.13	100	1	[1.13, 1.54]	1.33	17	0.17	100	1

Per ottenere queste tabelle, prima di tutto si è utilizzata la funzione `seq()`. La funzione in questione, permette di ricavare delle sequenze di intervalli tra due valori limite (che in questo caso sono il valore massimo ed il valore minimo registrati nel dominio dei valori assunti da ciascun regressore). `seq()` prende in input il valore minimo ed il valore massimo dell'intervallo ed il numero di sotto-intervalli che si desiderano ottenere che, stando al risultato della relazione empirica di cui sopra, sono 8. Tramite la funzione `cut()`, è stato poi possibile dividere l'intervallo in sotto-intervalli, ottenendo le classi di modalità. `cut()` prende in input il regressore e l'output della funzione `seq()`, e restituisce le classi di modalità.

Come è possibile notare, la prima colonna contiene il centro di classe, ossia il valore determinato dalla semi-somma dei limiti superiore ed inferiore di ogni classe. Al fine di determinare tali valori, si è utilizzata la funzione `rollmean()` che, preso in input il valore restituito dalla `seq()`, fornisce il centro di classe. La funzione `table()` applicata all'output della `cut()`, restituisce le frequenze assolute e, se si divide tale quantità per la numerosità campionaria, si ottengono le frequenze relative per ogni classe. Analogamente, la funzione `cumsum()` restituisce le frequenze cumulate e, dividendo tale quantità per la numerosità campionaria, si ottengono le frequenze relative cumulate.

È possibile notare, come le tabelle forniscono una sintesi dell'intera informazione fornita dai dati.

2.2 Istogrammi

In alternativa alla rappresentazione tabellare di una distribuzione di frequenza, è possibile ricorrere ad una rappresentazione grafica.

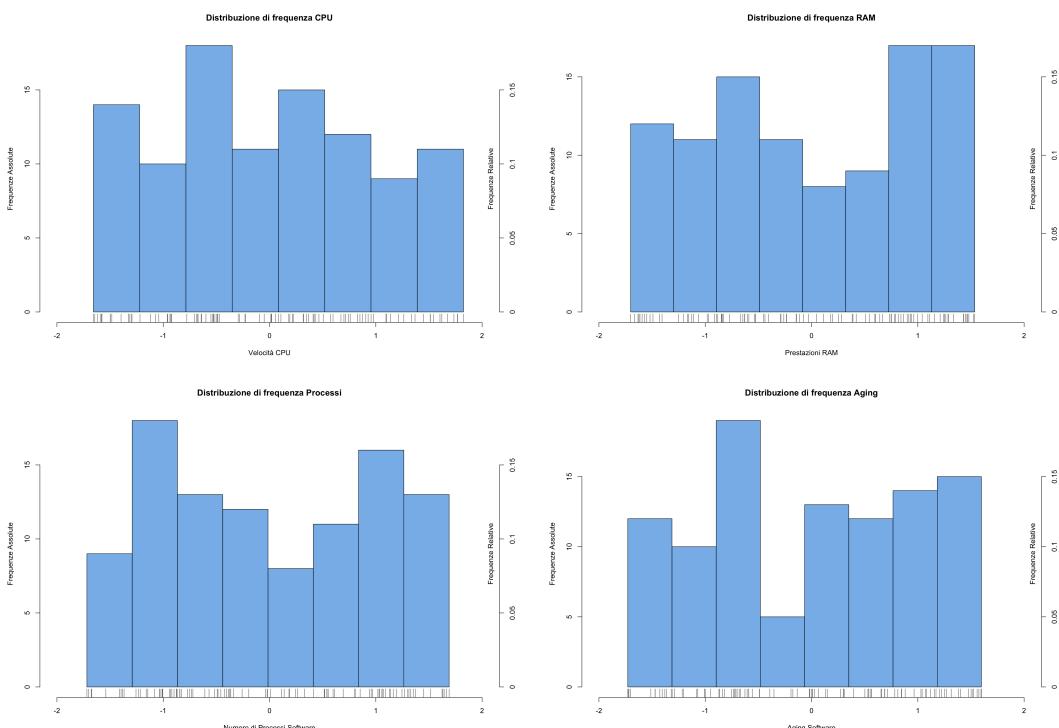
Le modalità di rappresentazione grafica di una distribuzione di frequenza, variano a seconda della tipologia di variabile in considerazione. Per le variabili di tipo quantitativo continuo, la rappresentazione grafica da adottare è un **istogramma**.

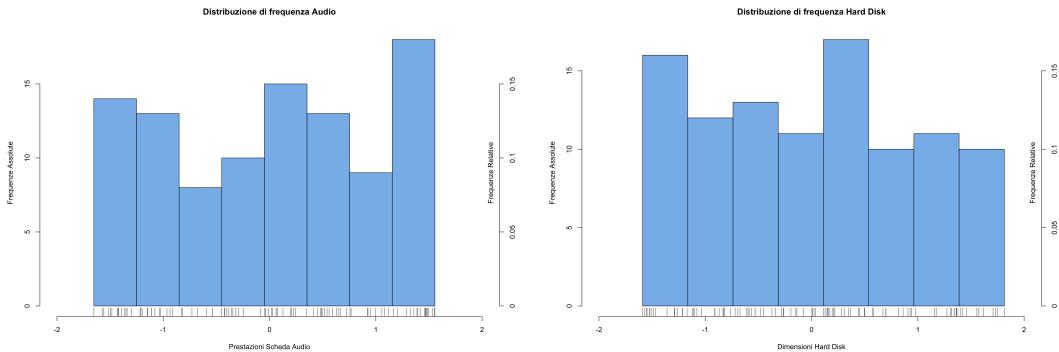
Un istogramma è un grafico che, alle classi di modalità di una variabile continua, fa corrispondere un rettangolo di area pari alla frequenza delle unità statistiche appartenenti a quella classe.

La suddivisione in classi adottata è la medesima del caso precedente.

In particolare l'output del comando `seq()` è stato passato come parametro all'attributo '`'breaks'`' del comando `hist()` (comando che permette di ottenere un istogramma), il quale, tra le tante cose, prende in input un vettore di valori che sono gli estremi di ciascuna classe di modalità. Di default, la funzione `hist()`, restituisce un istogramma avente sull'asse delle ordinate le frequenze assolute. Si è deciso di affiancare sul lato destro dell'istogramma un ulteriore asse riportante le frequenze relative.

Inoltre, si è scelto di inserire dei ticks (segni verticali, in grigio) sull'asse delle ascisse utilizzando il comando `rug()`, in corrispondenza delle modalità osservate, in modo da preservare la localizzazione dei dati originaria. Di seguito sono riportati gli istogrammi relativi ai vari regressori ottenuti come descritto sopra.





Dal punto di vista teorico, non è propriamente corretto utilizzare due assi delle ordinate sullo stesso grafico. Tuttavia, sulla base del fatto che nelle rappresentazioni grafiche si possono usare sia le frequenze relative, sia le frequenze assolute, perché la forma del grafico non cambia, ma cambia solo la scala delle ordinate, si è deciso di usare ambedue gli assi per ottenere maggiore chiarezza, completezza e coerenza, anche rispetto alle tabelle precedentemente esposte.

In seguito, si è sfruttata l'opzione di `hist()` che consente di ottenere l'istogramma considerando come scala per le ordinate le **densità**, `'freq=FALSE'`. In particolare, ciò ha permesso di tracciare un grafico di tipo **istogramma perequato**.

L'istogramma perequato, è una distribuzione continua della distribuzione di frequenza della variabile X e fornisce informazioni più nette e regolari, che sono particolarmente utili per effettuare confronti tra distribuzioni.

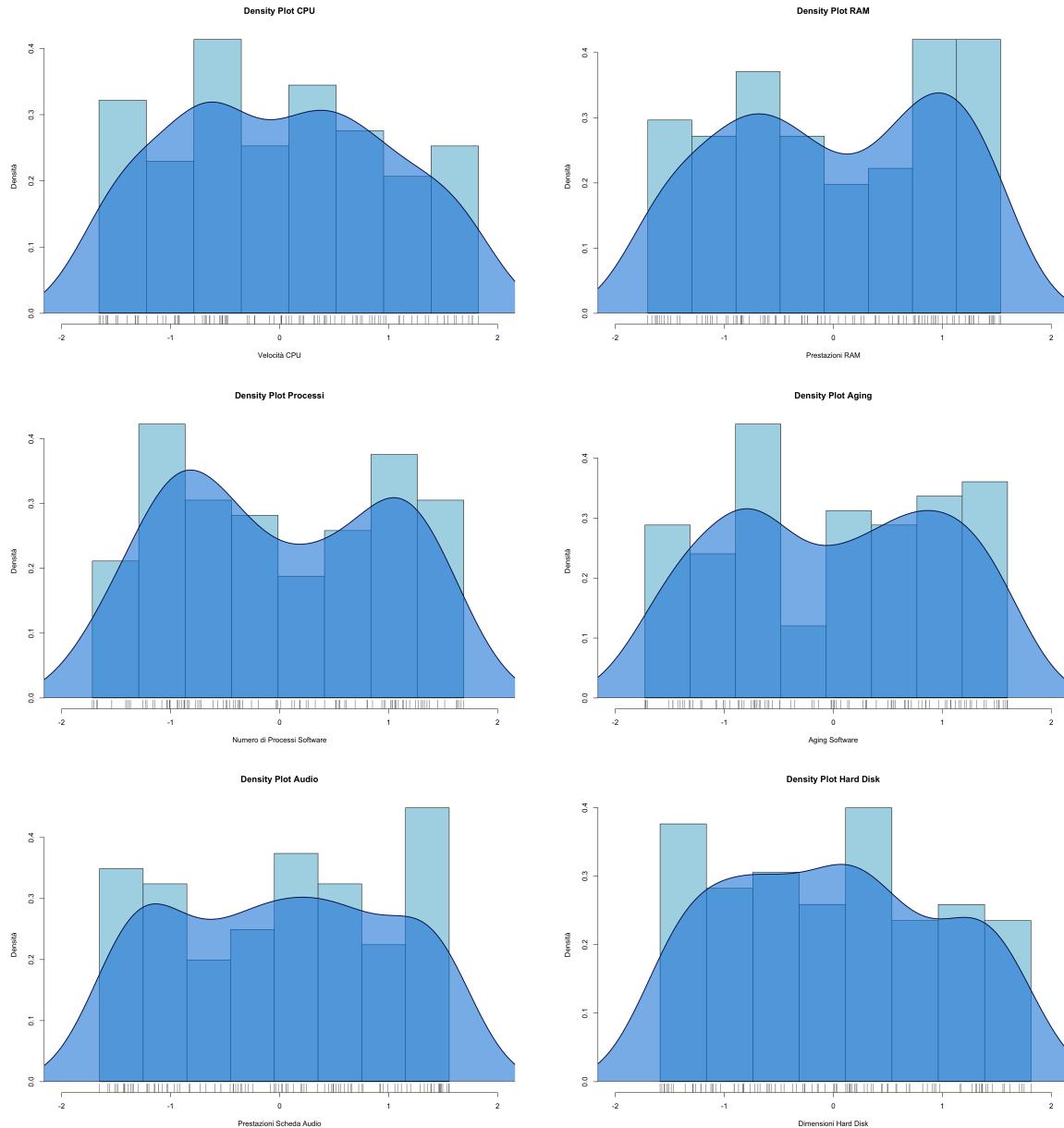
L'istogramma perequato (*kernel histogram*) si costruisce sostituendo ad ogni osservazione x_i di una variabile quantitativa X una funzione simmetrica (di modesta variabilità) centrata sul valore osservato x_i . Poi, si considera come rappresentazione finale l'area sottesa da tutte queste mini-funzioni.

Dopo aver fatto il plot dell'istogramma classico, come descritto in precedenza, si è sovrapposto un istogramma perequato utilizzando la funzione `lines()` e passandole come argomento `density()`, che a sua volta, prende come parametro il regressore di cui va a considerare la densità. Dopodichè, si è utilizzata la funzione `polygon()`, che prende come argomento `density()`, per riempire l'area sottesa dalla curva dell'istogramma perequato. Inoltre, anche in questo caso, sono stati sovrapposti al grafico dei `ticks` in corrispondenza dell'asse delle ascisse, utilizzando la funzione `rug()`, per delinare la distribuzione originaria dei dati.

Questi plot sono anche noti come **density plot**, in particolare sono distribuzioni di una variabile numerica che utilizza un istogramma perequato per mostrare la pdf della variabile. In R, la funzione `density()` calcola i valori stimati di densità. Il suo valore di ritorno viene utilizzato per la costruzione di un density plot.

È possibile notare come sommando l'area di ciascun rettangolo, calcolata come il prodotto tra l'ampiezza della classe di modalità e l'altezza, si ottenga esattamente 1.

Si riportano i grafici così ottenuti.



Si evidenzia come, dalla disposizione dei ticks rispetto alle varie classi di modalità, i dati risultino distribuiti in maniera generalmente omogenea per ogni classe.

Gli istogrammi dei regressori *cpu*, *ram*, *processi* ed *aging* presentano due picchi, mentre *audio* ed *hard disk*, ne hanno tre. Inoltre, *ram*, *processi* ed *aging* hanno una valle abbastanza accentuata tra i picchi e situata circa al centro della distribuzione.

2.3 Grafici a gradini

Le frequenze relative cumulate, calcolate come il rapporto tra le frequenze cumulate e la numerosità campionaria, sono i valori della **funzione di ripartizione empirica**.

La funzione di ripartizione empirica, calcolata nel valore x_0 è la funzione che associa ad ogni valore reale x_0 la frazione delle unità che sono minori o uguali (cioè non superiori) ad x_0 .

Dopo aver ordinato le modalità di X dal valore minimo al valore massimo, la funzione di ripartizione, è ottenuta cumulando progressivamente, al crescere di x , le frequenze relative.

$F(x)$ è una funzione definita per ogni x che varia da $-\infty$ a $+\infty$. Essa aumenta di un gradino pari ad $\frac{1}{n}$ per ogni valore osservato della popolazione; al variare di x , tale funzione cresce di $\frac{1}{n}$ ogni volta che la modalità è presente, e rimane costante quando essa è assente.

La funzione di ripartizione è sempre non decrescente e, nel caso delle frequenze relative cumulate, è compresa tra 0 ed 1.

Inoltre, poichè è stata definita per valori minori o uguali dell'argomento x , essa è continua sulla destra, cioè rispetto a valori che stanno alla destra del valore rispetto al quale viene calcolata.

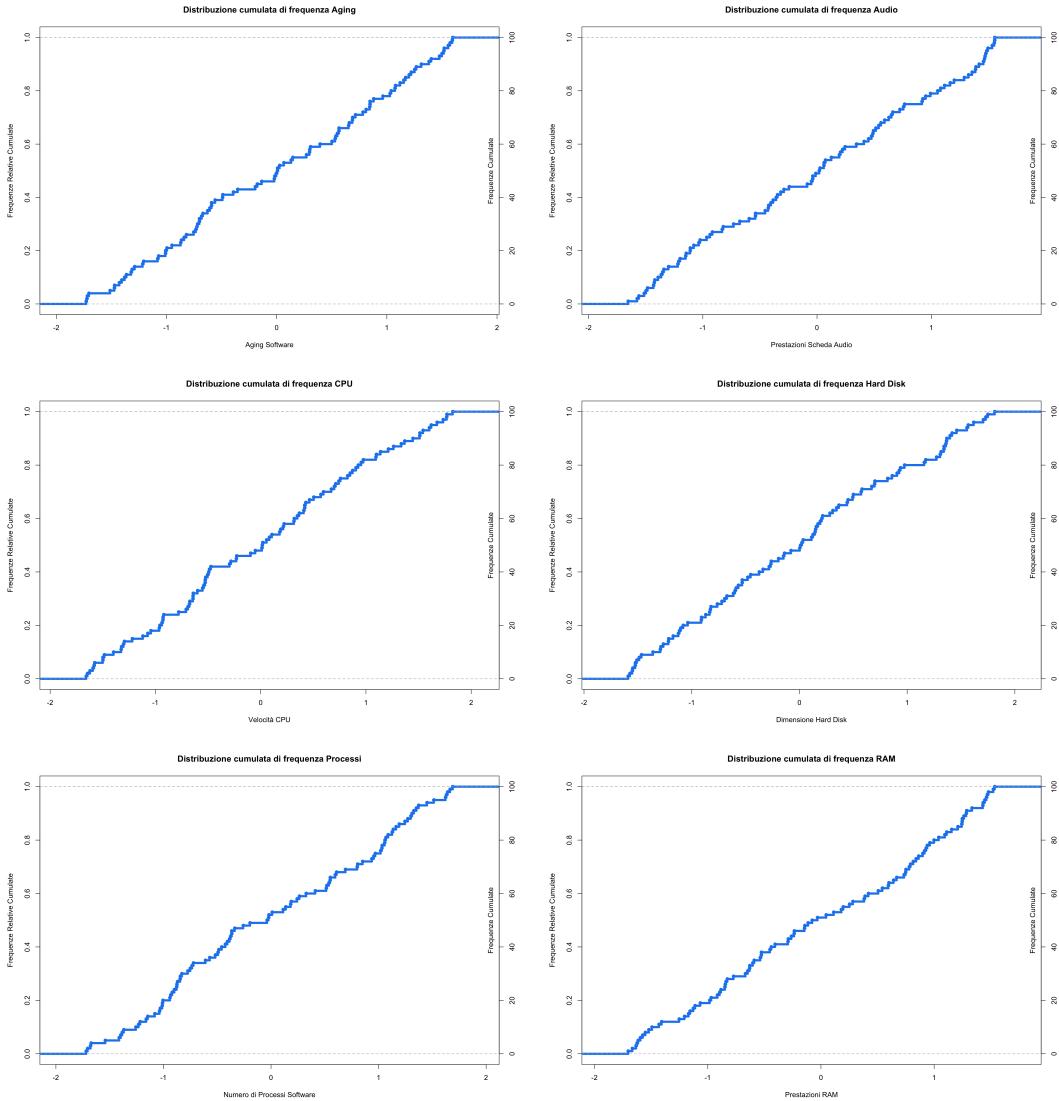
L'istogramma della distribuzione cumulata di frequenza si presenta come un **grafico a gradini**, ottenuto riportando, in corrispondenza di ciascuna classe, il valore della frequenza cumulata fino a tale classe.

R permette di calcolare la funzione di ripartizione cumulativa empirica dei dati tramite il comando `ecdf()` - *Empirical cumulative distribution function* applicato alla variabile di interesse.

Si può utilizzare la funzione `plot()` per ottenere una rappresentazione grafica della `ecdf`, e la funzione `lines()` che prende in input l'output restituito da `ecdf()` e congiunge i punti tramite una linea. In questo modo, di fatti, viene fuori un grafico a gradini.

Anche in questo caso, si è scelto di riportare, oltre all'asse delle frequenze relative cumulate, che viene messo di default, un ulteriore asse con le frequenze cumulate, sempre per una questione di coerenza con quanto riportato nelle tabelle delle frequenze.

Si riportano i grafici così ottenuti:



Come si può notare dai grafici, se si fa riferimento alla scala delle frequenze relative cumulate, i valori che la funzione assume sono compresi tra 0 ed 1, mentre nel caso delle frequenze cumulate, tra 0 e 100.

Ricordiamo infatti, che la numerosità campionaria n è di 100 elementi, per cui le frequenze relative cumulate saranno la centesima parte di quelle cumulate e, in riferimento al caso esaminato nel paragrafo precedente, le frequenze relative, saranno la centesima parte di quelle assolute.

2.4 Indici di tendenza centrale (o di posizione)

Tra le tipologie di indici sintetici che di solito vengono utilizzati, abbiamo gli **indici di tendenza centrale**. Essi hanno lo scopo di indicare intorno a quali valori tendono ad addensarsi i valori della caratteristica di interesse.

Gli **indici di posizione** sono stati introdotti per sintetizzare una pluralità di informazioni (modalità e frequenze) in un solo numero, in modo da poter effettuare confronti nel tempo, nello spazio o in circostanze differenti.

- La **media** è la somma di tutte le modalità di una variabile quantitativa, diviso per il numero delle unità statistiche. Essa rappresenta il *baricentro* delle distribuzioni di frequenza.

Molto importante è ricordare che questo indice **tende a dare eccessivo peso** ai valori estremi anche se questi sono poco numerosi.

- La **mediana** è la modalità dell'unità statistica che occupa il posto centrale nella distribuzione delle osservazioni ordinate (è quel valore della variabile rispetto al quale metà dei valori osservati, ordinati in senso crescente, risultano minori e l'altra metà maggiori).

Al contrario della media, questo indice non è influenzato dai valori estremi, ma l'ordinamento dei dati può essere molto oneroso se la numerosità campionaria è grande e non si presta a manipolazioni algebriche.

- La **moda** di una distribuzione di frequenza è la modalità a cui corrisponde la massima frequenza, assoluta o relativa.

Tipicamente si può assumere come il valore più rappresentativo della popolazione o del campione.

	Prestazioni SW	Velocità CPU	Dimensioni HD	Numero processi SW	Aging SW	Prestazioni scheda audio	Prestazioni RAM
Min.	8.9	-1.66	-1.59	-1.72	-1.73	-1.65	-1.7
1st Qu.	32.79	-0.73	-0.83	-0.88	-0.82	-0.94	-0.85
Median	39.63	0.02	0.02	-0.03	0.01	0.02	-0.05
Mean	38.72	0	0	0	0	0	0
3rd Qu.	44.9	0.78	0.83	0.98	0.85	0.81	0.9
Max.	59.54	1.82	1.81	1.69	1.6	1.56	1.54

La **moda** è stata calcolata prendendo nell'ambiente di sviluppo la tabella delle frequenze per ciascuna variabile in esame e considerando il valore con frequenza maggiore. Siccome ciascun valore ha frequenza unitaria, la moda non è significativa e dunque non è stata riportata in tabella.

Tra i risultati restituiti dal `summary()`, ci sono il **primo** e **terzo quartile**, che sono rispettivamente il valore della variabile in esame tale che al di sotto di esso vi siano $\frac{n}{4}$ delle unità statistiche (25%) ed il valore della variabile in esame tale che al di sotto di esso vi siano $\frac{3n}{4}$ delle unità statistiche (75%).

Il **minimo** ed il **massimo** sono chiaramente il valore minimo ed il valore massimo assunti dalla variabile in esame all'interno del dataset.

2.5 Indici di dispersione

Gli **indici di dispersione**, hanno lo scopo di dare una misura della variabilità nelle osservazioni e, quindi, di dare indicazioni sulla tendenza di tali valori a differire tra loro.

Per studiare la diversità che si osserva nei fenomeni reali, sono stati introdotti gli indici di dispersione.

- L'**escursione campionaria (o range)**, nonostante sia molto semplice da calcolare, in questo caso non è particolarmente significativa in quanto risulta attendibile solo nel caso vi sia un numero piccolo di campioni (al più 10, e non è questo il caso essendone presenti 100).

Esso è influenzato anche da un solo valore atipico (perchè molto basso o molto alto), il che rende l'indice vulnerabile ad errori e/o situazioni eccezionali. Talvolta, per ovviare a questo problema, si preferisce esaminare la variazione tra i quartili intermedi rispetto a quelli estremi (IQR, che sarà trattato nel successivo paragrafo).

- La **varianza**, rappresenta l'indice più importante per la misura della variabilità di una distribuzione. Essa è la media degli scarti al quadrato, pertanto è sempre positiva ed, al suo crescere, cresce la variabilità della distribuzione.
- La **deviazione standard (o scarto quadratico medio)**, è definita come la radice quadrata della varianza.
- Il **coefficiente di variazione**, è definito come rapporto tra la deviazione standard e media aritmetica. Esso, al contrario di varianza e deviazione standard che sono indici assoluti e quindi dipendono dall'unità di misura, è un indice relativo, svincolato dall'unità di misura della variabile.

Il coefficiente di variazione è utile per confrontare la variabilità di due fenomeni espressi in unità di misura non confrontabili. Esso ha significato solo se la media è maggiore di 0 e non è questo il caso in quanto la media è esattamente pari a 0 (tranne per quanto riguarda le *prestazioni software* del calcolatore dove è circa 0.2473318). Come è facilmente intuibile, il coefficiente di variazione diventa infinitamente grande al tendere della media a 0.

	Prestazioni SW Calcolatore	Velocità CPU	Dimensioni HD	Numero processi SW	Aging SW	Prestazioni scheda audio	Prestazioni RAM
Range	50.64	3.48	3.4	3.41	3.33	3.21	3.24
Varianza	91.72	1	1	1	1	1	1
Deviazione Standard	9.58	1	1	1	1	1	1

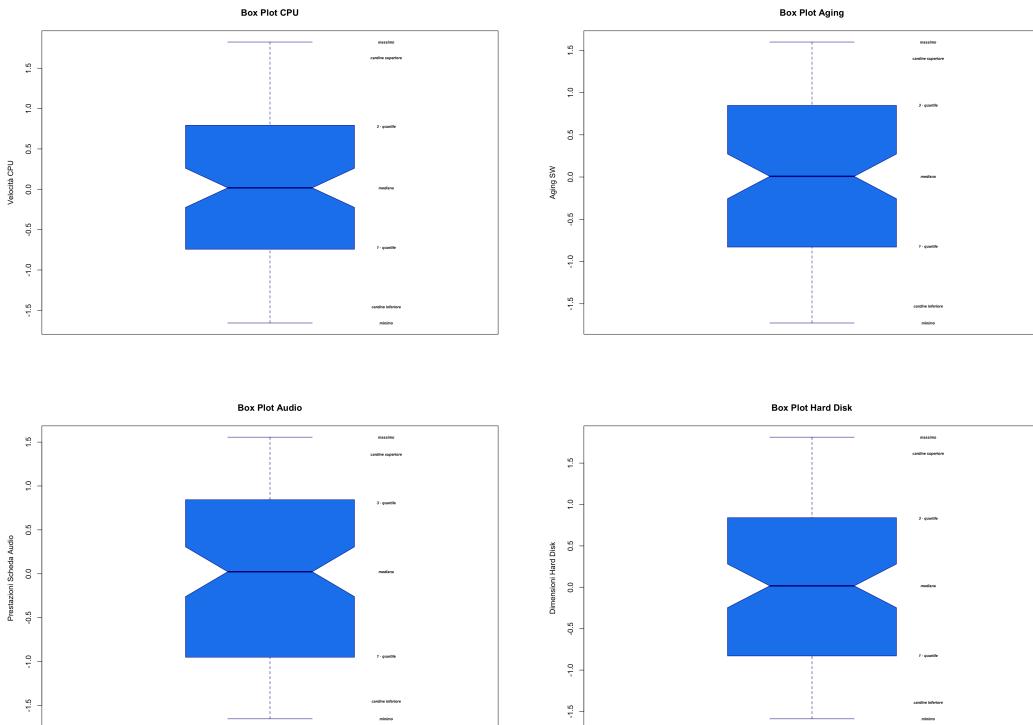
2.6 Box-Plot

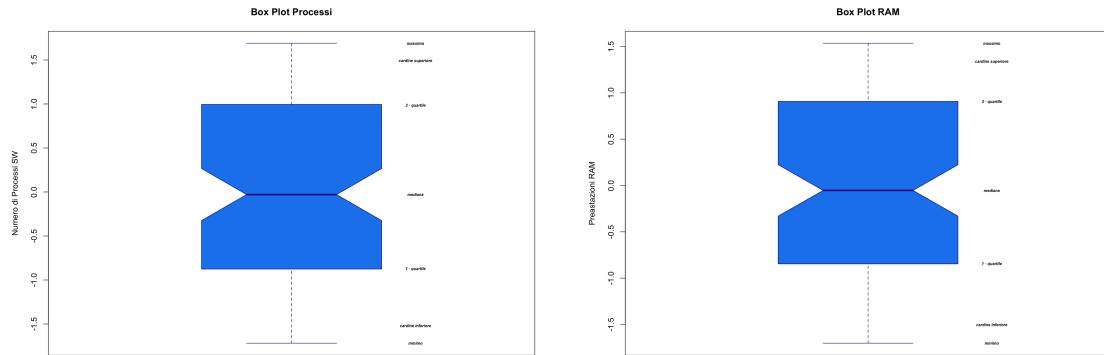
Un **Box-Plot** mostra la distribuzione dei dati per una variabile continua. Può essere utilizzato come strumento visivo per la verifica della normalità o per identificare possibili *outlier* (che sono più estremi della variazione attesa).

- La **linea centrale** nella scatola rappresenta la *mediana* dei dati. La metà dei dati si trova sopra questo valore, l'altra metà sotto. Se i dati sono simmetrici, la mediana è al centro della scatola, altrimenti sarà più vicina alla parte superiore o a quella inferiore della scatola.
- La **parte inferiore e superiore della scatola** mostrano il 25° e il 75° percentile, anche detti 1° e 3° quartile. La lunghezza della scatola è la differenza tra i due quartili e si chiama **range interquartile (IQR)**.

Il campo di variazione interquartile (IQR), è definito come la differenza tra il terzo ed il primo quartile, cioè $IQR(X) = Q_3 - Q_1$. L'IQR è basato sulle informazioni espresse dalla metà della popolazione.

- Le linee che si estendono a partire dalla scatola sono chiamate **baffi**. Essi rappresentano la variazione dei dati attesi e si estendono per 1.5 volte dall'IQR dalla parte superiore e inferiore della scatola. Se invece i dati non arrivano fino alla fine dei baffi, essi si estendono fino ai valori di dati minimi e massimi. Inoltre, se i dati ricadono sopra o sotto la fine dei baffi, sono rappresentati come punti denominati *outliers*.





I boxplot qui riportati, sono stati ottenuti utilizzando l'apposita funzione `boxplot()`, e passandole di volta in volta i vari regressori. Inoltre, si è scelto di sovrapporre al grafico delle etichette in corrispondenza di *mediana*, *primo quartile*, *terzo quartile*, *valore minimo*, *valore massimo*, *cardine superiore* e *cardine inferiore* per ottenere maggiore chiarezza e precisione.

Tutti i regressori presentano una simmetria dei dati, in quanto possiamo notare come la mediana sia posta esattamente al centro della scatola. Questo trova riscontro nei dati emersi dal `summary()` in precedenza dove i valori sono molto prossimi allo 0.

La prima cosa che si evince guardando i garfici ottenuti, è che non sono presenti outliers, in nessuno dei casi in esame.

Come conseguenza, si ha che gli estremi dei baffi corrispondono al valore minimo ed al valore massimo registrati per ciascuna variabile.

CAPITOLO 3

ANALISI DELLE RELAZIONI TRA LE VARIABILI

3.1 Analisi di correlazione

Il coefficiente di correlazione rappresenta una misura quantitativa del “grado di associazione” tra due variabili. Si noti che tale coefficiente misura il grado di associazione **lineare**, tale approccio infatti permette di individuare sicuramente alcune delle relazioni che legano la variabile dipendente a quelle non dipendenti, ma non basta ad escludere possibili relazioni **non lineari** fra le variabili in esame.

	Prestazioni SW Calcolatore	Velocità CPU	Dimensioni HD	Numero processi SW	Aging SW	Prestazioni scheda audio	Prestazioni RAM
Prestazioni SW Calcolatore	1	0.46	0.17	0.22	-0.48	0.12	0.16
Velocità CPU	0.46	1	0.18	-0.07	-0.04	0.17	-0.03
Dimensioni HD	0.17	0.18	1	0.04	-0.14	0.07	0.11
Numero processi SW	0.22	-0.07	0.04	1	-0.03	0	0.06
Aging SW	-0.48	-0.04	-0.14	-0.03	1	0.15	-0.05
Prestazioni scheda audio	0.12	0.17	0.07	0	0.15	1	0.05
Prestazioni RAM	0.16	-0.03	0.11	0.06	-0.05	0.05	1

La tabella qui riportata è stata ottenuta utilizzando il comando `corr()`.

Il coefficiente di correlazione R è un numero compreso tra -1 e 1 .

Per l'interpretazione dei coefficienti riportati in tabella, si è tenuto conto del fatto che quanto più essi sono prossimi ad 1 in valore assoluto, tanto più il grado di associazione tra le variabili è elevato. Viceversa, quanto più sono prossimi a 0, tanto più il grado di associazione è basso. Inoltre, valori positivi di R indicano una correlazione positiva (Y cresce se X cresce), mentre valori negativi indicano una correlazione negativa (Y decresce se X cresce).

In definitiva, il valore $R = 1$ indica perfetta correlazione positiva, il valore $R = -1$ indica perfetta correlazione negativa (o anti-correlazione), mentre il valore $R = 0$ indica completa assenza di correlazione.

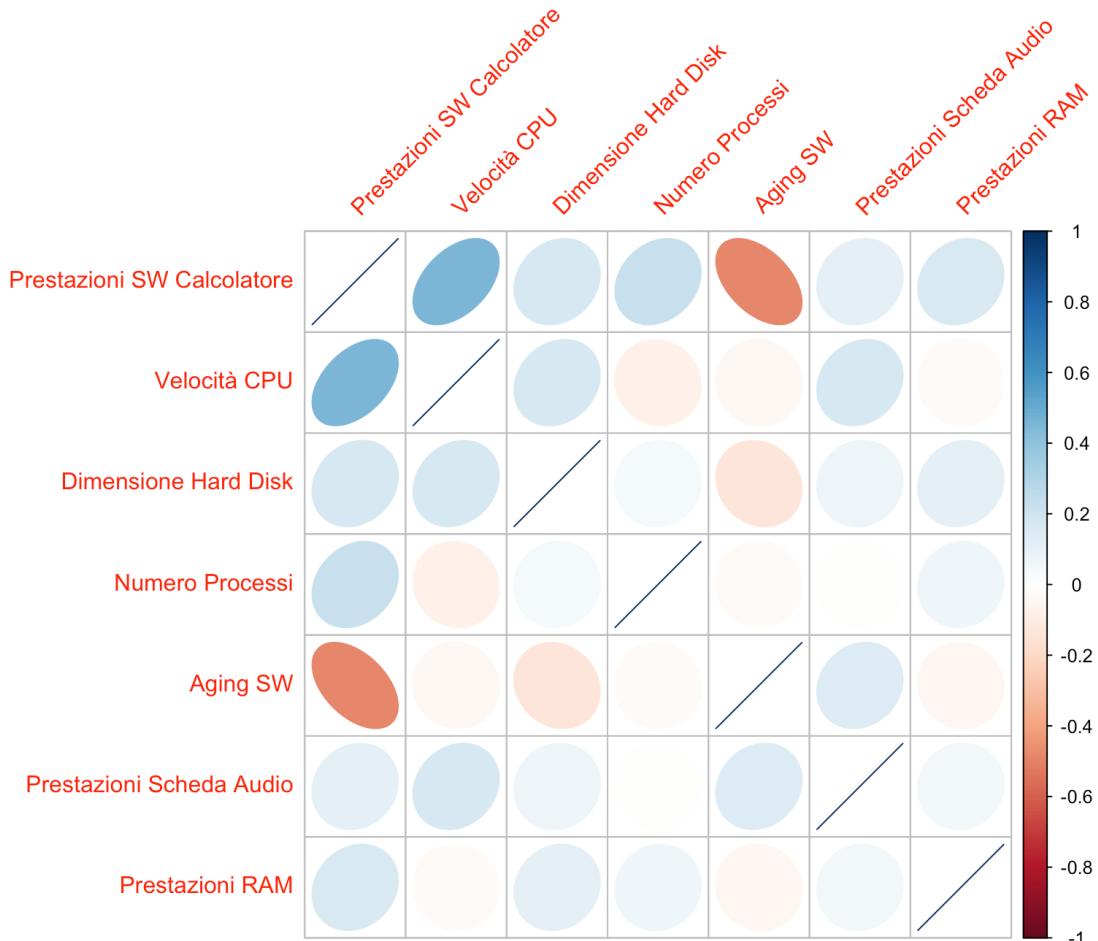
Dunque, seguendo questi parametri, è emerso che:

- c'è una **moderata correlazione positiva** tra le prestazioni software del calcolatore e la velocità della CPU,
- c'è una **moderata correlazione negativa** tra le prestazioni software del calcolatore e l'aging del software,
- c'è una **debole correlazione positiva** tra le prestazioni software del calcolatore ed il numero di processi software in esecuzione,
- c'è una **molto debole correlazione positiva** tra le prestazioni software del calcolatore e le dimensioni dell'Hard Disk,
- c'è una **molto debole correlazione positiva** tra le prestazioni software del calcolatore e le prestazioni della RAM,
- altrove vi è all'incirca **assenza di correlazione**.

Ad una prima analisi, dal punto di vista delle implicazioni reali che i componenti rappresentati dai regressori possono avere tra loro, quanto emerso in merito alla correlazione tra le prestazioni del calcolatore ed il regressore *cpu*, è risultato calzante con un'eventuale implicazione reale tra questi due fattori. Anche l'anti-correlazione tra le prestazioni del calcolatore ed il regressore *aging*, trova effettivo riscontro in una situazione reale. Lo stesso vale per il regressore *processi*, dove chiaramente si può ben immaginare che il numero di processi in esecuzione possa influenzare in qualche modo le prestazioni di un calcolatore. La più debole correlazione con *hard disk*, è risultata non particolarmente calzante a quanto potrebbe accadere in una situazione reale, al contrario di quella con *ram*.

Queste osservazioni, per quanto possano essere immediate ed intuitive, non necessariamente rispecchiano il caso in esame. Bisogna infatti tenere conto della sostanziale differenza che sussiste tra il concetto di *correlazione* ed il concetto di *causa-effetto*. Un valore elevato del coefficiente di correlazione non implica necessariamente che una variabile è la causa della variazione osservata dell'altra, e viceversa. Spesso può essere presente una causa comune non osservata che determina la variazione di entrambe le variabili studiate.

Per avere un supporto grafico, rispetto ai risultati ottenuti dall'analisi di correlazione e riportati in tabella, si è deciso di utilizzare utilizzare un **Correlation Plot**, ottenuto con il comando `corrplot()`.



Per la corretta interpretazione del *correlation plot* si ritiene opportuno specificare che le **ellissi** individuano la direzione del legame di correlazione (*correlate* - se schiacciate verso destra - *anti-correlate* - se schiacciate verso sinistra) e il **colore** è segnale della forza di tale legame secondo la scala indicata a fianco.

I risultati ottenuti, come atteso, rispecchiano in pieno quanto visto in precedenza, infatti sono presenti due ellissi in celeste scuro schiacciate verso destra che indicano una rilevante correlazione positiva tra le *prestazioni del calcolatore* ed il regressore *cpu*, ci sono anche due ellissi in arancio schiacciate verso sinistra, che indicano una rilevante correlazione negativa tra *prestazioni del calcolatore* ed il regressore *aging*, ci sono due ellissi in celeste chiaro schiacciate verso destra che rilevano una debole correlazione positiva tra *prestazioni del calcolatore* ed il regressore *processi* ed infine ci sono due ellissi in azzurro chiaro che rilevano una molto debole correlazione positiva tra *prestazioni del calcolatore* ed il regressore *hard disk* e tra *prestazioni del calcolatore* ed il regressore *ram*.

3.2 Scatter Plot

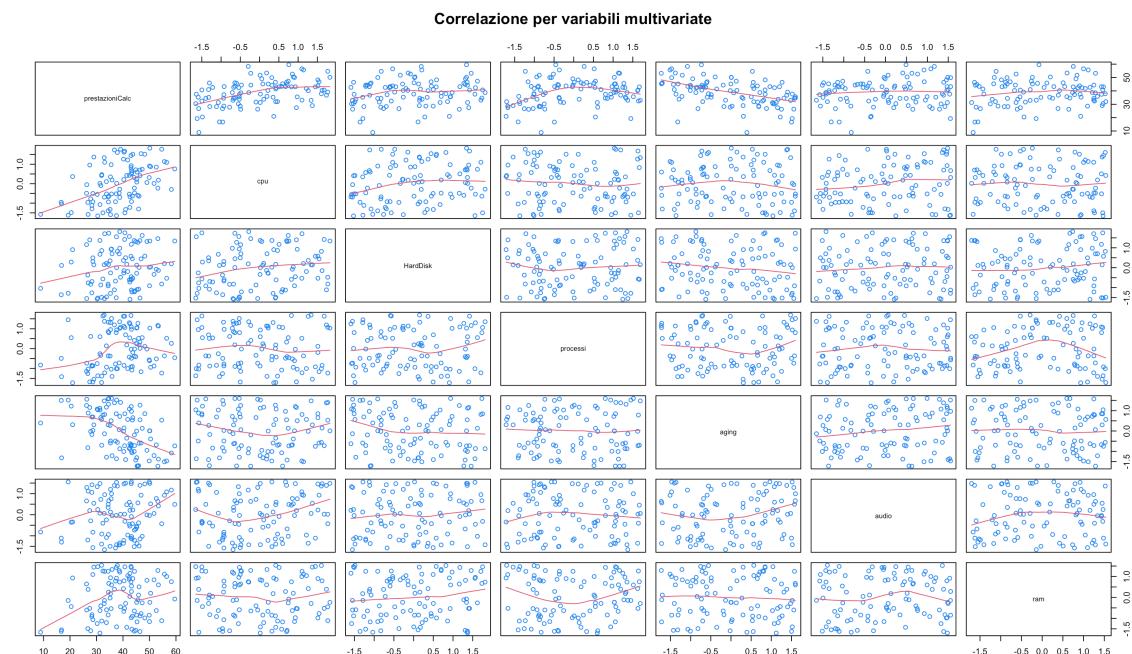
Per eseguire un'analisi preliminare dei dati che rilevi un legame tra due variabili quantitative, si utilizzano i **diagrammi di correlazione** per ciascuna coppia di variabili in esame. Il diagramma di correlazione prevede il tracciamento di due assi ortogonali con opportune scale che permettano la rappresentazione delle due variabili.

Se si vuole studiare la dipendenza della variabile Y dalla variabile X è consuetudine utilizzare l'asse verticale per la Y e quello orizzontale per la X . Nel grafico per variabili multivariate ottenuto tramite il comando `pairs()` (che prende in input tutte le variabili indipendenti (regressori) e la variabile dipendente, e produce una matrice di **scatter plot**), è possibile avere contezza visiva di entrambe le versioni. Questo risulta particolarmente utile in quanto un andamento potrebbe essere più accentuato in una versione, piuttosto che in un'altra, sebbene comunque si precisa che il coefficiente di correlazione resti il medesimo.

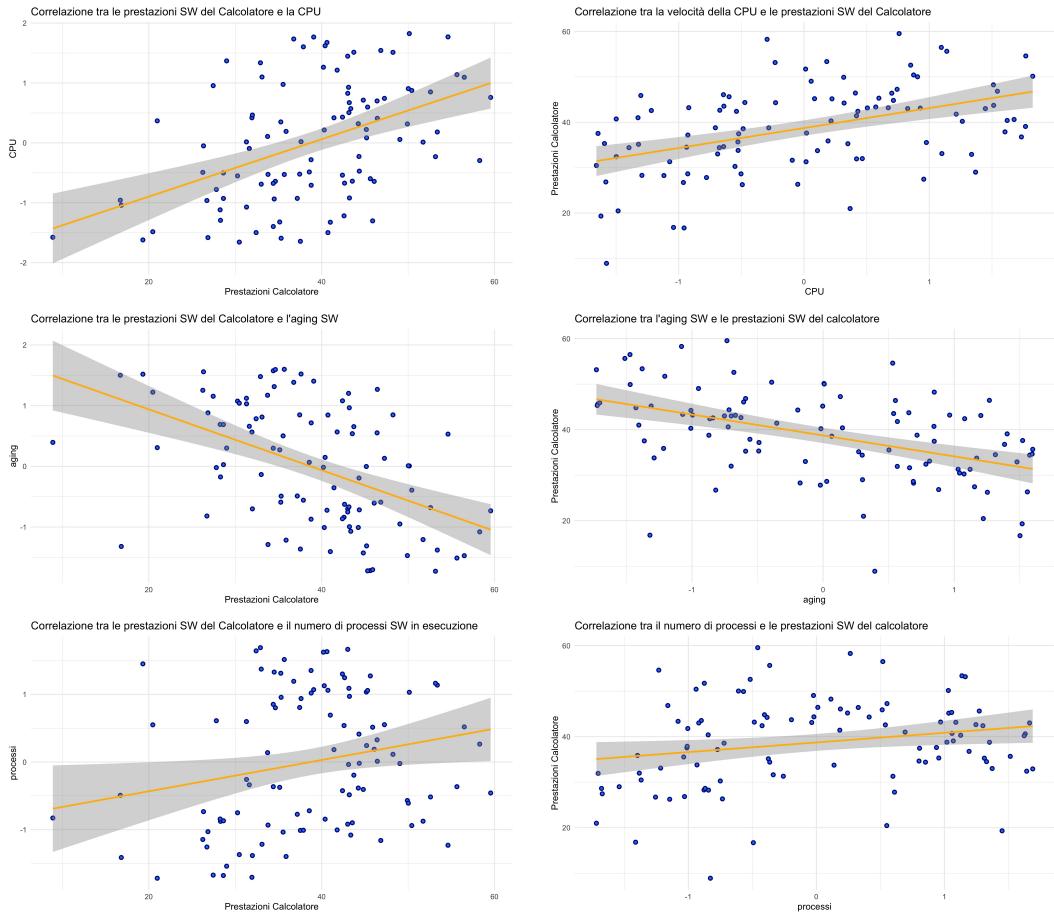
Si è scelto di sovrapporre agli Scatter Plot una **curva regolare**, che interpola al meglio la linea di tendenza tra ascisse ed ordinate, evidenziandone quindi, ancora di più, l'andamento.

I risultati attesi a partire da quanto emerso dall'analisi di correlazione effettuata in precedenza, sono i seguenti:

- una relazione lineare crescente che leggi **prestazioni software del calcolatore e velocità della CPU**,
- una relazione lineare descrescente che leggi **prestazioni software del calcolatore e l'aging del software**,
- gli andamenti delle altre variabili aventi trend sempre meno approssimabili con un andamento lineare per via del loro basso coefficiente di correlazione lineare.



Dando uno sguardo d'insieme ai risultati, si possono notare dei grafici che hanno andamenti più significativi di altri. Di seguito si riportano delle *macro* dei grafici in questione, ottenute con `ggplot()` che presentano in giallo una retta che cerca di seguire l'andamento della distribuzione dei punti, ed in grigio una sezione corrispondente all'intervallo di confidenza al 95%.



Per le stesse motivazioni esposte in precedenza, i grafici sono stati riportati in due forme diverse: in quelli a sinistra troviamo il regressore sulle ordinate e la variabile dipendente (prestazioni software) sulle ascisse, a destra invece sulle ordinate troviamo la variabile dipendente, e sulle ascisse il regressore.

I primi due grafici, evidenziano una marcata correlazione lineare positiva tra la variabile dipendente ed il regressore *cpu*. I secondi due grafici, evidenziano una marcata anti-correlazione tra la variabile dipendente ed il regressore *aging*. Gli ultimi due grafici evidenziano una moderata correlazione positiva (più evidente nel primo, che nel secondo) tra la variabile dipendente ed il regressore *processi*. Anche in questo caso il risultato ottenuto era atteso e risulta coerente con quanto emerso fin ora.

CAPITOLO 4

DEFINIZIONE DEL MODELLO STATISTICO

4.1 Analisi delle relazioni tra le variabili tramite regressione polinomiale

L'**analisi di regressione** viene utilizzata per stimare il valore atteso delle grandezze di interesse in funzione di una combinazione lineare dei livelli delle variabili misurate in grado di spiegarne l'andamento.

L'obiettivo di questa sezione di analisi è stimare il modo in cui i vari regressori influenzano la variabile indipendente.

4.2 Scelta del modello di regressione

Per la scelta del **modello di regressione multipla** che descrive la variabile dipendente in funzione delle sei variabili esplicative disponibili, sono stati presi in considerazione a poco a poco tutti i campi del dataset, ponendo l'attenzione in primo luogo sulle sole relazioni lineari rispetto ai regressori.

L'individuazione del modello si è basata in principio sulla regressione lineare multipla, mediante cui è stata valutata la risposta del modello all'aggiunta dei vari regressori tramite il comando `lm()` di R, analizzando per ognuno dei modelli ottenuti, i valori del *p*-value associati ad ogni regressore per avere la certezza che fossero statisticamente rilevanti.

Il comando `lm()` di R viene utilizzato per analizzare modelli lineari (*linear models*).

L'output del comando `lm()` presenta vari campi a cui, di volta in volta, si farà riferimento per prendere decisioni in merito all'evolversi del modello. I campi in questione sono:

- *Coefficients*: i coefficienti sono costanti sconosciute che rappresentano l'intercetta e i valori che moltiplicano i vari regressori considerati nel modello.
- *Estimate*: questa colonna del campo *Coefficients* contiene diverse righe, la prima rappresenta il valore atteso dell'intercetta e gli altri, i valori attesi degli altri coefficienti.
- *t value*: questa colonna del campo *Coefficients* rappresenta il valore della statistica t, ossia dà un'indicazione di quanto il coefficiente possa considerarsi diverso da 0 e quindi di quanto il regressore a cui è associato sia rilevante ai fini della descrizione del modello. Più esso è maggiore di 0, e più si rigetta con forza l'ipotesi nulla che il coefficiente sia uguale a 0, ed è possibile affermare che il regressore è rilevante.
- $Pr(>|t|)$: questa colonna del campo *Coefficients* rappresenta il p-value. Più esso è piccolo e minore è la probabilità che la relazione tra il regressore e la variabile indipendente sia dovuta al caso. Fissato un livello di significatività, che da questo punto in poi si considererà essere 0.05, si valuta il p-value rispetto a questa soglia. Più esso è piccolo rispetto alla soglia stabilita e più si rigetta con forza l'ipotesi nulla.
- *Residual standard error*: è la deviazione standard dei residui, più è piccolo e migliore è la qualità del modello ottenuto.
- *Multiple R-squared*: è la misura di R^2 per i modelli che hanno più regressori. Il coefficiente di determinazione R^2 in questione, è definito come il rapporto tra l'SQR (variabilità imputata alla regressione) e l'SQTOT (variabilità totale). Generalmente, maggiore è l' R^2 e meglio il modello rappresenta i dati. Aggiungendo regressori al modello, l' R^2 aumenta sempre.
- *Adjusted R-squared*: Aggiunge delle penalità all'inserimento dei regressori nel modello. Aiuta a trovare un equilibrio tra il modello più parsimonioso e quello che rappresenta meglio i dati. Se si ha una grande differenza tra Multiple ed Adjusted R^2 , si è andati in overfitting.
- *F-statistic*: è un indicatore di se esiste o meno una relazione tra i regressori e la variabile di risposta. Più il suo valore è lontano da 1 e meglio è. Quanto grande il suo valore debba essere dipende sia dalla numerosità campionaria, che dal numero di regressori. Se la numerosità campionaria è grande, una *F-statistic* poco più grande di 1 è già sufficiente per rigettare l'ipotesi nulla e quindi poter asserire che effettivamente esiste una relazione tra i regressori. Al contrario, se la numerosità campionaria è piccola per poter giungere alla medesima conclusione è necessario avere un valore di *F-statistic* più elevato.

Il primo modello analizzato prende in considerazione soltanto il regressore *cpu*:

$$Y = \beta_0 + \beta_1 \text{cpu} + \epsilon \quad (4.1)$$

I risultati dell'analisi di tale modello sono stati i seguenti:

```

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 38.7223    0.8552  45.280 < 2e-16 ***
cpu          4.3967    0.8595   5.116 1.55e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.552 on 98 degrees of freedom
Multiple R-squared:  0.2108, Adjusted R-squared:  0.2027 
F-statistic: 26.17 on 1 and 98 DF,  p-value: 1.555e-06

```

Come è possibile notare dal risultato qui riportato, sia il *p*-value che la *F*-statistic danno conferma della capacità esplicativa di tale regressore rispetto al modello. Si è poi aggiunto il regressore *Hard Disk*, ottenendo:

$$Y = \beta_0 + \beta_1 \text{cpu} + \beta_2 \text{HardDisk} + \epsilon \quad (4.2)$$

I risultati dell'analisi di tale modello sono stati i seguenti:

```

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 38.7223    0.8548  45.301 < 2e-16 ***
cpu          4.2329    0.8733   4.847 4.76e-06 ***
HardDisk     0.9107    0.8733   1.043      0.3    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.548 on 97 degrees of freedom
Multiple R-squared:  0.2195, Adjusted R-squared:  0.2034 
F-statistic: 13.64 on 2 and 97 DF,  p-value: 6.025e-06

```

L'aggiunta di *Hard Disk* non ha apportato variazioni significative: sebbene l' R^2 sia aumentato (si ricorda che esso aumenta all'aggiunta di ciascun regressore), il suo basso *p*-value, l'aumento del *residual standard error* e la diminuzione della *F*-statistic, sono indici che esso non sia particolarmente rappresentativo del modello. Si è però deciso di mantenere tale regressore in quanto si voleva analizzare l'evoluzione del modello fino ad ottenerne uno che contenesse tutti i regressori.

Il terzo modello considerato ha visto l'inserimento del regressore *processi*, ottenendo:

$$Y = \beta_0 + \beta_1 \text{cpu} + \beta_2 \text{HardDisk} + \beta_3 \text{processi} + \epsilon \quad (4.3)$$

L'aggiunta di tale regressore ha dimostrato che esso è in grado di spiegare significativamente l'andamento della variabile dipendente, avendo un *p*-value inferiore rispetto alla soglia fissata. Infatti dall'analisi risulta:

```

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 38.7223   0.8242  46.980 < 2e-16 ***
cpu          4.4253   0.8448   5.238 9.57e-07 ***
HardDisk     0.7777   0.8434   0.922  0.35879    
processi    2.3995   0.8317   2.885  0.00483 **  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.242 on 96 degrees of freedom
Multiple R-squared:  0.2818,    Adjusted R-squared:  0.2593 
F-statistic: 12.55 on 3 and 96 DF,  p-value: 5.404e-07

```

Il quarto modello considerato presenta in più del precedente il regressore *aging*, il quale ha portato ad avere:

$$Y = \beta_0 + \beta_1 cpu + \beta_2 HardDisk + \beta_3 processi + \beta_4 aging + \epsilon \quad (4.4)$$

Analogamente al caso precedente, *aging* è in grado di spiegare parte dell'andamento della variabile dipendente. Questo è sicuramente dimostrato dal fatto che il suo *p*-value è prossimo allo 0 e l' R^2 è aumentato in modo rilevante. Risulta infatti:

```

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 38.7223   0.7019  55.165 < 2e-16 ***
cpu          4.3692   0.7195   6.073 2.58e-08 ***
HardDisk     0.1850   0.7248   0.255  0.79907    
processi    2.3053   0.7084   3.254  0.00158 **  
aging       -4.3563   0.7126  -6.113 2.15e-08 *** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.019 on 95 degrees of freedom
Multiple R-squared:  0.4845,    Adjusted R-squared:  0.4628 
F-statistic: 22.33 on 4 and 95 DF,  p-value: 5.125e-13

```

Come atteso, i regressori *aging* e *cpu*, fin ora, sono quelli risultati maggiormente significativi. Questo risultato può essere giustificato da quanto emerso dall'analisi effettuata in precedenza sugli Scatter Plot, infatti si era evinto come questi ultimi fossero in relazione lineare con la variabile dipendente.

Si è a questo punto considerata l'aggiunta del regressore *audio*, ottenendo un modello del tipo:

$$Y = \beta_0 + \beta_1 cpu + \beta_2 HardDisk + \beta_3 processi + \beta_4 aging + \beta_5 audio + \epsilon \quad (4.5)$$

Tale regressore si è rivelato **non essere significativo ai fini del modello**, infatti:

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	38.7223	0.6974	55.526	< 2e-16 ***
cpu	4.1903	0.7247	5.782	9.58e-08 ***
HardDisk	0.1162	0.7215	0.161	0.87236
processi	2.2960	0.7039	3.262	0.00154 **
aging	-4.5314	0.7176	-6.315	8.89e-09 ***
audio	1.0813	0.7216	1.499	0.13732

Signif. codes:	0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1			

Residual standard error: 6.974 on 94 degrees of freedom
 Multiple R-squared: 0.4966, Adjusted R-squared: 0.4698
 F-statistic: 18.54 on 5 and 94 DF, p-value: 8.921e-13

Il sesto modello considerato è quello completo di tutti i regressori. Questo ha visto l'aggiunta del regressore *ram*. Si è pervenuti ad:

$$Y = \beta_0 + \beta_1 cpu + \beta_2 HardDisk + \beta_3 processi + \beta_4 aging + \beta_5 audio + \beta_6 ram + \epsilon \quad (4.6)$$

I risultati ottenuti sono stati i seguenti:

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	38.72225	0.68938	56.170	< 2e-16 ***
cpu	4.26009	0.71745	5.938	4.94e-08 ***
HardDisk	-0.02224	0.71743	-0.031	0.97534
processi	2.22894	0.69680	3.199	0.00189 **
aging	-4.47633	0.71000	-6.305	9.59e-09 ***
audio	1.00622	0.71452	1.408	0.16239
ram	1.25239	0.70085	1.787	0.07720 .

Signif. codes:	0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1			

Residual standard error: 6.894 on 93 degrees of freedom
 Multiple R-squared: 0.5133, Adjusted R-squared: 0.4819
 F-statistic: 16.35 on 6 and 93 DF, p-value: 9.076e-13

Gli alti *p*-value dei regressori *Hard Disk* ed *audio*, hanno portato ad asserire che i regressori risultati convincenti ai fini della determinazione del modello sono stati:

- **CPU**
- **Processi**
- **Aging**
- **Ram**

Mentre per *cpu* ed *aging* si ha un'ottima risposta da parte del modello (*p*-value particolarmente bassi) e per i processi una buona risposta, per quanto riguarda *ram*, questa relazione deve essere maggiormente approfondita per capire se tale regressore possa o meno essere scartato.

Alla luce di quanto emerso si è quindi iniziato ad approfondire il modello di regressione lineare completo del tipo:

$$Y = \beta_0 + \beta_1 \text{cpu} + \beta_2 \text{processi} + \beta_3 \text{aging} + \beta_4 \text{ram} + \epsilon \quad (4.7)$$

considerando i suddetti regressori. Si riporta di seguito il risultato della relativa analisi di regressione:

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	38.7223	0.6893	56.174	< 2e-16 ***
cpu	4.4356	0.6952	6.380	6.41e-09 ***
processi	2.2360	0.6960	3.213	0.0018 **
aging	-4.3159	0.6944	-6.215	1.36e-08 ***
ram	1.3143	0.6952	1.891	0.0617 .

Signif. codes:	0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1			

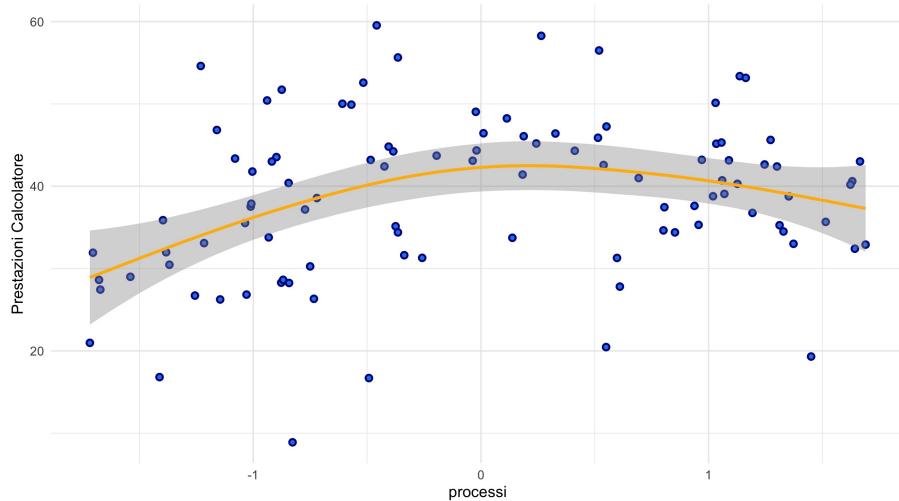
Residual standard error: 6.893 on 95 degrees of freedom

Multiple R-squared: 0.5029, Adjusted R-squared: 0.482

F-statistic: 24.03 on 4 and 95 DF, p-value: 9.495e-14

È possibile notare come l'indice R^2 sia di poco inferiore rispetto a quello ottenuto dal modello comprensivo di tutti i regressori.

In seguito, si è ipotizzato un possibile modello di regressione polinomiale. In particolare, considerando anche i risultati dell'analisi di correlazione fra *prestazioni software del calcolatore* e *processi* (indice di correlazione lineare = 0.22143) e tenuto in conto che lo Scatter Plot ad esso associato può meglio essere approssimato da una relazione quadratica, come si evince dal seguente grafico:



Il suo andamento **"pseudo-parabolico"** suggerisce una **possibile relazione non lineare**.

Si ricorda che all'interno dei modelli lineari rientra anche la regressione polinomiale, ossia quella in cui compaiono alcuni regressori con grado uguale o superiore a 2.

Il modello in questione continua ad essere lineare nei parametri β_1, \dots, β_n .

Si è deciso dunque di ipotizzare un modello di regressione del tipo:

$$Y = \beta_0 + \beta_1 \text{cpu} + \beta_2 \text{processi} + \beta_3 \text{processi}^2 + \beta_4 \text{aging} + \beta_5 \text{ram} + \epsilon \quad (4.8)$$

Dall' analisi è emerso infatti:

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 43.6865   0.8974  48.680 < 2e-16 ***
cpu          4.8358   0.5674   8.523 2.52e-13 ***
processi    2.5169   0.5666   4.442 2.43e-05 ***
I(processi^2) -5.0144  0.7086  -7.077 2.62e-10 ***
aging        -3.9869  0.5657  -7.047 3.01e-10 ***
ram           2.4811   0.5881   4.219 5.65e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.597 on 94 degrees of freedom
Multiple R-squared:  0.6757,    Adjusted R-squared:  0.6584 
F-statistic: 39.17 on 5 and 94 DF,  p-value: < 2.2e-16
```

Si può notare come l'indice R^2 sia maggiore rispetto a quelli precedenti, il *residual standard error* sia diminuito e la *F-statistic* permetta di rigettare con forza l'ipotesi nulla ed asserire che il modello ottenuto descriva bene il set di dati. Oltretutto, grazie all'aggiunta di processi^2 , il regressore legato alla RAM ha acquisito grande rilevanza statistica.

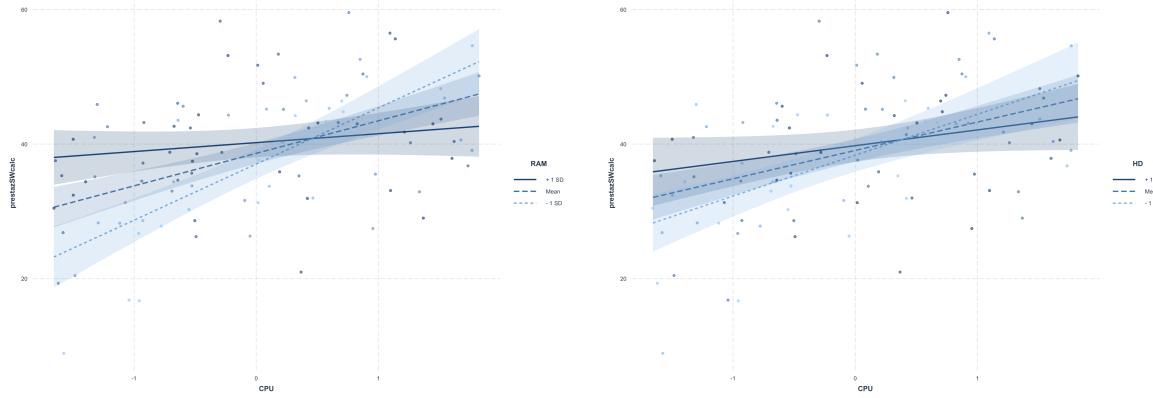
L'ultima fase dell' analisi condotta ha cercato di evidenziare possibili **interazioni** fra variabili.

In regressione, un effetto di interazione esiste quando l'effetto di una variabile indipendente su una variabile dipendente cambia, secondo il valore assunto da una o più altre variabili indipendenti.

Per rilevare la presenza o assenza di interazioni tra le variabili indipendenti, si utilizzano gli **interactions plot**. Questi grafici presentano la variabile dipendente sull'asse delle ordinate ed una variabile indipendente sull'asse delle ascisse. Vengono tracciate varie linee per i diversi livelli di una variabile indipendente potenzialmente interagente con quella presente sull'asse delle ascisse. Se le linee prodotte sono parallele, si può escludere l'ipotesi di interazione tra le variabili indipendenti considerate, in caso contrario, si può concludere che esse siano interagenti.

In R, si sono ottenute tutte le possibili combinazioni di interazioni tra le variabili dipendenti, utilizzando per ciascuna la funzione *interaction_plot()*. Questa funzione prende in input la variabile di risposta e le due variabili di cui si sta considerando l'interazione. Le linee tracciate sul grafico rappresentano la risposta della variabile dipendente al variare della variabile dipendente posta sulle ascisse (*cpu*) in relazione al valore fissato per l'altra variabile dipendente (nel primo caso *ram*, nel secondo *hardDisk*). Si è proceduto all'interpretazione dei grafici ottenuti come descritto in

precedenza ed in particolare due grafici presentavano delle intersezioni tra le linee che erano particolarmente marcate: *cpu:ram* e *cpu:hardDisk*. Si riportano di seguito i due grafici in questione.



Si è deciso inoltre, di escludere l'interazione tra *cpu* ed *hard disk*, sebbene questa appaia significativa, in quanto avrebbe implicato l'inserimento del regressore *hard disk* risultato in precedenza non statisticamente rilevante per il modello.

Si è deciso di non avanzare ipotesi in merito ad interazioni di ordine superiore al secondo in quanto queste sarebbero potute essere di difficile interpretazione.

L'interazione individuata come significativa è stata pertanto quella fra *cpu* e *ram*. Sulla base di tale osservazione, si è giunti al modello finale:

$$Y = \beta_0 + \beta_1 \text{cpu} + \beta_2 \text{ram} + \beta_3 \text{cpu} * \text{ram} + \beta_4 \text{processi} + \beta_5 \text{processi}^2 + \beta_6 \text{aging} + \epsilon \quad (4.9)$$

la cui analisi produce il seguente risultato :

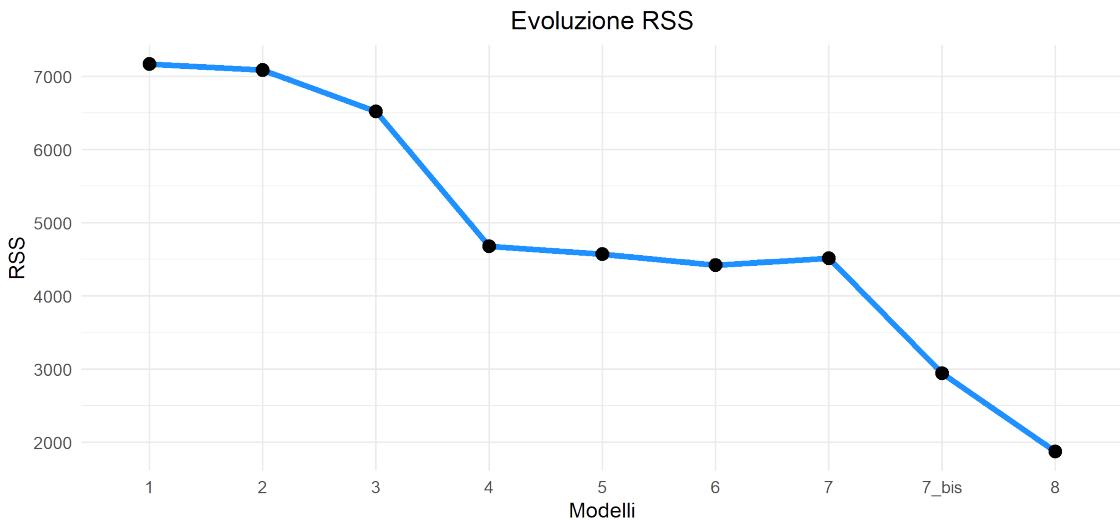
```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 43.6273   0.7203  60.572 < 2e-16 ***
cpu          5.1754   0.4577  11.307 < 2e-16 ***
ram          2.4386   0.4720   5.166 1.36e-06 ***
processi     1.8369   0.4642   3.957 0.000148 ***
I(processi^2) -5.0523   0.5687  -8.884 4.68e-14 ***
aging        -4.2156   0.4551  -9.263 7.43e-15 ***
cpu:ram      -3.5206   0.4838  -7.277 1.07e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.492 on 93 degrees of freedom
Multiple R-squared:  0.7933,    Adjusted R-squared:  0.78 
F-statistic: 59.5 on 6 and 93 DF,  p-value: < 2.2e-16
```

Tutti i regressori hanno *p*-value significativo, per cui sono statisticamente rilevanti ai fini della descrizione del set di dati. Inoltre, rispetto al caso precedente la *F-statistic* ha subito un incremento non trascurabile che permette di poter rigettare l'ipotesi nulla con ancora maggior certezza ed asserire che il modello ottenuto rappresenta bene il dataset in esame.

4.3 Evoluzione dei parametri rilevanti

Di seguito sono riportate rispettivamente le evoluzioni ottenute dall'analisi dell'*RSS*, Residual Sum of Squares e dell'*R²*, coefficiente di determinazione multipla. Si può vedere come, sulla base degli indici dei modelli riportati, le rispettive evoluzioni evidenzino la correttezza delle osservazioni fatte fin'ora.



L'*RSS*, Residual Sum of Squares, è la devianza dei residui $Dev(\hat{e}) = \sum_{i=1}^n (\hat{e}_i)^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2$. Esso misura di quanto differisce il modello di regressione dal set di dati dopo l'individuazione di un determinato modello. Per questo motivo, il desiderio generale è quello di individuare un *RSS* molto basso nel modello finale.

Il **grafico dell'evoluzione dell'*RSS***, ha un andamento generalmente decrescente, se non per un breve tratto.

In particolare si parte da un valore di *RSS* molto alto (circa 7000) corrispondente al modello contenente unicamente il regressore *cpu*.

L'andamento per i primi 4 modelli (ottenuti inserendo nell'ordine *hard disk*, *processi*, *aging* al regressore iniziale *cpu*) è di tipo esponenziale decrescente.

In seguito, inserendo il regressore *audio* nel modello, si ha un leggero abbassamento del grafico in linea retta e lo stesso vale allo step successivo per quanto riguarda l'inserimento del regressore *ram*.

Togliendo il regressore *audio*, l'*RSS* aumenta leggermente, provocando un leggero innalzamento del grafico.

Decresce significamente considerando all'interno del modello il regressore *processi*².

Decresce ulteriormente, ma in maniera meno significativa, all'inserimento del regressore *cpu:ram*, che ha portato al raggiungimento del modello finale con un *RSS* inferiore a 2000.



L'indice di determinazione multipla R^2 è definito da $R^2 = \frac{Dev(\hat{y})}{Dev(y)} = 1 - \frac{Dev(\hat{e})}{Dev(y)} = \frac{SQTOT - SQE}{SQTOT}$, ossia è pari al rapporto tra la variabilità totale e la variabilità non controllata imputabile alle unità sperimentali, e la variabilità totale stessa. L' R^2 è un indicatore della bontà di adattamento dei dati del modello di regressione e può assumere valori compresi tra 0 ed 1, pertanto si desidera che esso sia il più alto possibile nel modello finale (si trascurano eventuali problemi di overfitting).

Il **grafico dell'evoluzione dell'indice di determinazione R^2** ha un andamento generalmente crescente, se non per un breve tratto.

Esso parte da un valore iniziale di poco superiore a 0.2, corrispondente al modello contenente il solo regressore *cpu*.

L'andamento per i primi 4 modelli (ottenuti inserendo nell'ordine *hard disk*, *processi* ed *aging* al regessore iniziale *cpu*) è di tipo esponenziale crescente.

Dopodichè, c'è un leggero innalzamento dovuto all'inserimento del regressore *audio* all'interno del modello, lo stesso vale per *ram* allo step successivo.

Togliendo il regressore *audio*, l' R^2 si abbassa di poco, e questo è l'unico tratto decrescente dell'intero grafico.

L'inserimento di *processi*² comporta un aumento consistente dell' R^2 .

Infine, si ha ancora una volta un aumento, meno significativo del precedente, considerando l'interazione *cpu:ram* che ha portato alla determinazione del modello finale, con un valore di R^2 di circa 0.8.

I valori ottenuti, sia di *RSS* che di R^2 risultano essere soddisfacenti e si può stimare che il modello a cui si è pervenuti spieghi bene l'80% circa dei dati.

CAPITOLO 5

STIMA DEI PARAMETRI DEL MODELLO

Per la stima dei coefficienti, nella regressione lineare polinomiale multipla, così come nella regressione lineare semplice, si utilizza il **metodo dei minimi quadrati**.

L'adozione di tale metodo è giustificata dal fatto che, sulla base dell'ipotesi sulla natura dell'errore ($\epsilon \sim N(0, \sigma^2)$), esso fornisce degli stimatori dei parametri, i quali, nell'ambito della classe degli stimatori lineari sono gli unici ad essere *non distorti* ed a *minima varianza*.

Sotto l' ipotesi di normalità della variabile aleatoria ϵ , le stime ai minimi quadrati coincidono con le stime che si ottengono applicando il **metodo della Massima Verosimiglianza**.

5.1 Metodo dei "Minimi Quadrati"

Con questo metodo si assegnano a $\beta_0, \beta_1, \dots, \beta_p$ quei valori b_0, b_1, \dots, b_p che rendono minima la quantità (ai fini della stima si usano i simboli y_1, \dots, y_n per indicare i valori osservati della variabile Y)

$$SQE = \sum_{i=1}^n (y_i - b_0 - b_1 x_{i1} - b_2 x_{i2} - \dots - b_p x_{ip})^2 \quad (5.1)$$

funzione delle $p + 1$ variabili b_0, b_1, \dots, b_p .

Per determinare il minimo di una funzione di più variabili, basta risolvere il sistema di equazioni che si ottiene uguagliando a zero le sue derivate parziali prime. Il calcolo in questione risulta più agevole mediante l'utilizzo dell'algebra delle matrici.

I risultati di questa operazione si possono sintetizzare nella seguente *proposizione*.

Si considerino il vettore dei valori osservati della variabile risposta

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad (5.2)$$

e la matrice

$$X = \begin{bmatrix} 1 & x_{11} & x_{12} & \dots & x_{1p} \\ 1 & x_{21} & x_{22} & \dots & x_{2p} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & x_{n1} & x_{n2} & \dots & x_{np} \end{bmatrix} \quad (5.3)$$

dei valori osservati delle variabili esplicative. Allora, la stima ai minimi quadrati del vettore

$$\beta = [\beta_0, \beta_1, \dots, \beta_n]^T \quad (5.4)$$

dei coefficienti di regressione del modello, è data da

$$b = (X^T X)^{-1} X^T y \quad (5.5)$$

I valori calcolati utilizzando quest'approccio sono stati i seguenti:

```
[,1]
intercetta 43.627261
cpu          5.175387
ram          2.438585
cpu_ram     -3.520613
aging        -4.215586
processi     1.836872
processi_2   -5.052338
```

in accordo con quelli restituiti dal comando *lm()*:

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 43.6273   0.7203  60.572 < 2e-16 ***
cpu          5.1754   0.4577  11.307 < 2e-16 ***
ram          2.4386   0.4720   5.166 1.36e-06 ***
aging        -4.2156   0.4551  -9.263 7.43e-15 ***
processi     1.8369   0.4642   3.957 0.000148 ***
I(processi^2) -5.0523   0.5687  -8.884 4.68e-14 ***
cpu:ram      -3.5206   0.4838  -7.277 1.07e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

5.2 Stima degli intervalli di confidenza

Per gli intervalli di confidenza nella regressione multipla, si segue la stessa logica utilizzata per la pendenza (β_1) in un modello a singolo regressore, sapendo che la statistica pivot $T = \frac{\hat{\beta}_j - \beta_j}{SE(\hat{\beta}_j)}$ è distribuita come una *t di Student* con $\nu = n - p - 1$ gradi di libertà.

Dato che lo stimatore $\hat{\beta}_0$ è una variabile aleatoria Normale con valore atteso β_0 , nel caso in cui il parametro σ sia noto, l'intervallo di confidenza su β_0 si ricava sulla base del fatto che

$$Z = \frac{\hat{\beta}_0 - \beta_0}{\sqrt{Var(\hat{\beta}_0)}} = \frac{\hat{\beta}_0 - \beta_0}{\sigma \sqrt{\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}}} \quad (5.6)$$

è una variabile aleatoria *Normale Standard*.

Nel caso più frequente nelle applicazioni in cui il parametro σ è *incognito*, indicando con $S = \sqrt{MSQE}$ lo stimatore di tale parametro, si può fare riferimento al fatto che

$$T = \frac{\hat{\beta}_0 - \beta_0}{S \sqrt{\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}}} \quad (5.7)$$

è una quantità pivot che ha una distribuzione di tipo *Student*.

Fissato il livello di confidenza $1 - \alpha$, si ha $Pr\{t_{\alpha/2; \nu} < T \leq t_{1-\alpha/2; \nu}\} = 1 - \alpha$, da cui segue che l'intervallo di confidenza sul parametro β_0 è dato da

$$Pr\left\{ \hat{\beta}_0 - t_{1-\alpha/2; \nu} S \sqrt{\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}} < \beta_0 \leq \hat{\beta}_0 + t_{1-\alpha/2; \nu} S \sqrt{\frac{1}{n} + \frac{\bar{x}^2}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}} \right\} = 1 - \alpha \quad (5.8)$$

Per lo stimatore $\hat{\beta}_1$, essendo che è una variabile aleatoria Normale con valore atteso β_1 , nel caso in cui il parametro σ sia noto, l'intervallo di confidenza su β_1 si ricava sulla base del fatto che

$$Z = \frac{\hat{\beta}_1 - \beta_1}{\sqrt{Var(\hat{\beta}_1)}} = \frac{\hat{\beta}_1 - \beta_1}{\sigma \sqrt{\frac{1}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}}} \quad (5.9)$$

è una variabile aleatoria *Normale Standard*.

Nel caso più frequente nelle applicazioni in cui il parametro σ è *incognito*, indicando con $S = \sqrt{MSQE}$ lo stimatore di tale parametro, si può fare riferimento al fatto che

$$T = \frac{\hat{\beta}_1 - \beta_1}{S \sqrt{\frac{1}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}}} \quad (5.10)$$

è una quantità pivot che ha una distribuzione di tipo *Student*.

Fissato il livello di confidenza $1 - \alpha$, si ha $Pr\{t_{\alpha/2;\nu} < T \leq t_{1-\alpha/2;\nu}\} = 1 - \alpha$, da cui segue che l'intervallo di confidenza sul parametro β_1 è dato da

$$Pr\left\{ \hat{\beta}_1 - t_{1-\alpha/2;n-p-1} S \sqrt{\frac{1}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}} < \beta_1 \leq \hat{\beta}_1 + t_{1-\alpha/2;n-p-1} S \sqrt{\frac{1}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}} \right\} = 1 - \alpha \quad (5.11)$$

Si ricorda che per $n > 30$ si ha che $t_{1-\alpha/2} \sim z_{1-\alpha/2}$

Sono di seguito riportati i risultati ottenuti:

	2.5%	97.5%
<i>Intercept</i>	42.74	44.52
<i>CPU</i>	4.28	6.07
<i>RAM</i>	1.54	3.34
<i>Aging</i>	-5.11	-3.32
<i>Processi</i>	0.94	2.73
<i>I(Processi^2)</i>	-6.13	-3.98
<i>CPU:RAM</i>	-4.45	-2.59

Si può notare come questi si discostino di poco dai valori riportati utilizzando il comando *confint()* sul risultato ottenuto da *lm()*:

	2.5%	97.5%
<i>Intercept</i>	42.2	45.06
<i>CPU</i>	4.27	6.08
<i>RAM</i>	1.5	3.38
<i>Aging</i>	-5.12	-3.31
<i>Processi</i>	0.92	2.76
<i>I(Processi^2)</i>	-6.18	-3.92
<i>CPU:RAM</i>	-4.48	-2.56

La **verifica** del modello stimato è possibile, per ciascun parametro, sottponendo a verifica l'ipotesi $H_0 : \beta_j = 0$ contro l'ipotesi alternativa $H_1 : \beta_j \neq 0$ utilizzando una variabile casuale di Student con $n - (p + 1)$ gradi di libertà.

In modo alternativo, ma equivalente, se l'intervallo di confidenza per un parametro β_j include lo 0, non si può rifiutare l'ipotesi H_0 che quel parametro sia nullo.

Se non si può rifiutare H_0 , la variabile corrispondente X_j non possiede capacità esplicativa per la risposta Y .

Dalla stima degli intervalli di confidenza effettuata nel paragrafo precedente, a supporto delle altre analisi condotte, possiamo dimostrare come le variabili corrispondenti X_j associate ai coefficienti stimati posseggano capacità esplicative per la risposta Y , dal momento in cui in nessun intervallo di confidenza stimato è incluso il valore 0.

5.3 Stima della v.a. errore

Una volta stimati i parametri del modello di regressione, tuttavia, per definire completamente il modello, rimane da stimare il parametro σ^2 che rappresenta la varianza della variabile aleatoria ϵ .

Tale parametro è in relazione con l'incertezza associata alla stima del valore atteso della variabile aleatoria Y ed è intuitivamente evidente che la quantità

$SQE = \sum_{i=1}^n (y_i - \hat{y}_i)^2$, divisa per un opportuno numero di gradi di libertà, rappresenta una stima non distorta del parametro σ^2 .

Pertanto, nel caso di un modello lineare generale con k variabili indipendenti e quindi con $k + 1$ parametri (per la presenza della costante β_0), se n è il numero di osservazioni sperimentali, il numero di gradi di libertà per la stima di σ^2 risulta pari a $\nu = n - (k + 1)$. Nel caso in esame, fissata la quantità SQE , e fissato il numero di gradi di libertà, la stima della varianza della variabile aleatoria ϵ ha esibito il seguente risultato:

$$Var(\epsilon) = \frac{SQE}{n - (k + 1)} = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{93} = 20,17835 \quad (5.12)$$

Il numeratore della relazione sopra riportata corrisponde anche alla *Devianza dei residui*, $Dev(\hat{\epsilon})$, indicata anche con l'acronimo *RSS* (Residual Sum of Squares), già incontrata in precedenza.

CAPITOLO 6

ANALISI DEL MODELLO

6.1 Calcolo del coefficiente di determinazione

Il **coefficiente di determinazione** considerato precedentemente è stato ricavato direttamente a partire dal comando `lm()` di R. Tuttavia, questo può essere calcolato direttamente a partire dai dati.

Ricordando infatti che

$$R^2 = \frac{SQR}{SQTOT} = \frac{SQTOT - SQE}{SQTOT} \quad (6.1)$$

dove:

- **SQTOT** è rappresentativo della variabilità totale dell'esperimento,
- **SQE** è rappresentativo della variabilità non controllata,
- **SQR** è rappresentativo della variabilità dovuta alla regressione.

Ricordando che le espressioni per il calcolo tali quantità sono le seguenti:

$$SQTOT = \sum_{i=1}^n (Y_i - \bar{Y})^2 \quad SQE = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (6.2)$$

Il valore di R^2 così calcolato è pari a 0.7933 e coincide con i valori restituiti dal comando `lm()`.

6.2 Grafici diagnostici - Analisi dei residui



Dopo aver stimato il modello di regressione, è necessario **verificare che siano valide le ipotesi di base** tramite opportuni test-statistici. Si riportano di seguito le suddette ipotesi:

- $Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip} + \epsilon_i, i = 1, 2, \dots, n$ (**Linearità**)
- $E(\epsilon_i) = 0, i = 1, 2, \dots, n$
- $Var(\epsilon_i) = \sigma^2 < +\infty, i = 1, 2, \dots, n$ (**Omoschedasticità**)
- $Corr(\epsilon_i, \epsilon_j) = 0, \forall i \neq j = 1, 2, \dots, n$ (**Indipendenza**)
- le variabili esplicative $X_j, j = 1, 2, \dots, p < n$ sono note senza errore, sono in numero $p < n$ e nessuna variabile esplicativa può essere espressa mediante una combinazione lineare delle altre.

In primo luogo, **verifichiamo che la media degli errori non sia significativamente diversa da zero** calcolando l'intervallo di confidenza sul valore atteso dei residui.

L'esito di tale test è stato il seguente:

- `lower_conf_int = -0.72`
- `upper_conf_int = 0.72`

il che comporta che la media degli errori non è significativamente diversa da 0, essendo compresa nell'intervallo $[-0.72, 0.72]$.

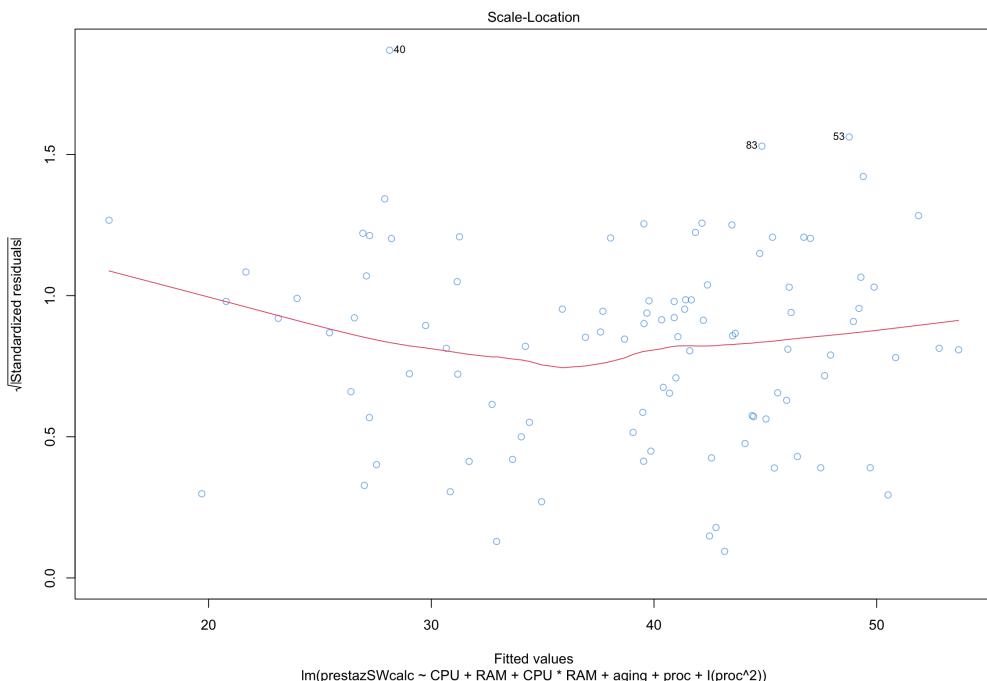
Un'ulteriore analisi che può essere effettuata prevede un'assunzione più forte sull'origine della variabile aleatoria errore ed è la seguente:

- **Normalità** (la distribuzione dei residui è di tipo gaussiano).

6.2.1 Verifica omoschedasticità

L'omoschedasticità è l'ipotesi di varianza costante dei residui.

Per verificare l'**omoschedasticità** si è scelto di utilizzare uno *Scale-Location plot*, che rende particolarmente agevole questo tipo di analisi. Esso riporta sull'asse delle ordinate la radice quadrata dei residui standardizzati (invece di tracciare semplicemente i residui), e sull'asse delle ascisse i valori stimati a partire dal modello.



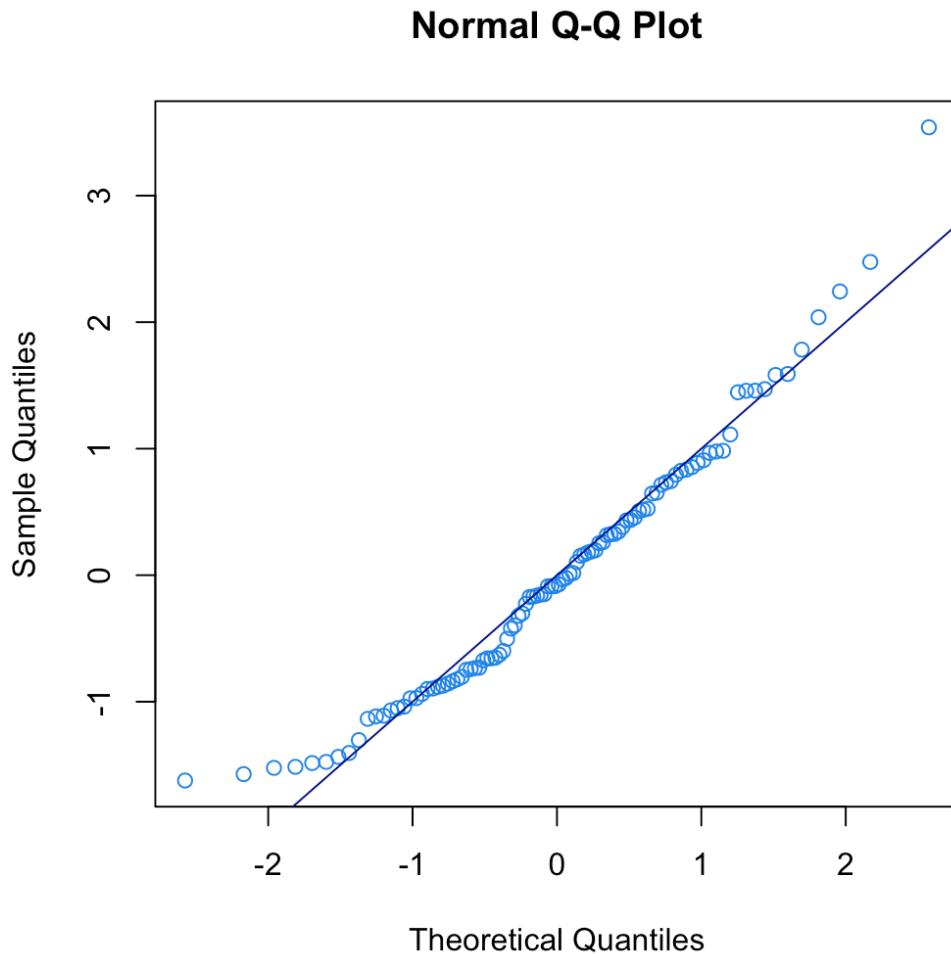
Dal grafico possiamo fare due particolari osservazioni. In primo luogo, verifichiamo che la linea rossa sia approssimativamente orizzontale. In secondo luogo, verifichiamo che la distribuzione dei punti non mostri alcun pattern evidente. In altre parole, i residui dovrebbero essere casualmente sparsi intorno alla linea rossa. Ciascuna delle due osservazioni è sufficiente ad asserire che si tratta di omoschedasticità.

In questo caso entrambe le condizioni sembrano essere rispettate, per cui è possibile assumere che la condizione di omoschedasticità sia soddisfatta.

6.2.2 Verifica normalità

La normalità migliora sensibilmente tutte le proprietà finite ed asintotiche degli stimatori per i parametri del modello. Questo requisito è auspicabile, anche se non vincolante, ai fini delle proprietà degli stimatori dei minimi quadrati.

Per verificare la **normalità**, si ricorre al QQ-plot dei residui. Se gli errori seguono una distribuzione gaussiana, i punti del grafico dovrebbero concentrarsi intorno ad una retta a 45°.



È evidente dal grafico qui riportato che quanto richiesto dall'ipotesi di normalità sia rispettato, infatti i punti presenti nel QQ-Plot sono ben disposti lungo la retta a 45°.

Inoltre è opportuno verificare la normalità anche con un test statistico appropriato, come, ad esempio, il *test di Shapiro-Wilk*.

Il **test di Shapiro-Wilk** è uno dei test più potenti per la verifica della normalità. La statistica W può assumere valori da 0 a 1. Qualora il valore della statistica W sia troppo piccolo, il test rifiuta l'ipotesi nulla che i valori campionari siano distribuiti come una variabile casuale normale.

Il risultato ottenuto è:

```
Shapiro-Wilk normality test  
data: residui  
W = 0.96365, p-value = 0.007387
```

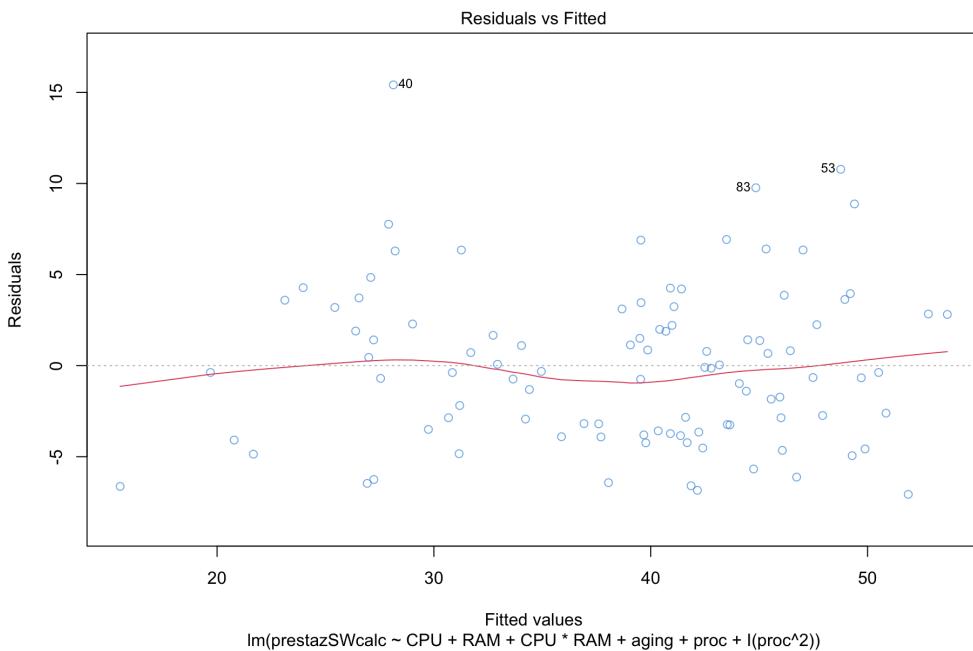
Da quanto emerso dal test, essendo che il valore del *p*-value è < 0.05, pertanto si rifiuta l'ipotesi di normalità.

C'è da sapere che il test era originariamente limitato ad una dimensione campionaria $n \leq 50$. Successivi studi portarono ad evidenziare come le approssimazioni dei pesi utilizzate dall'algoritmo fossero inadeguate se $n \geq 50$. Dunque, al fine di poter evidenziare la normalità della distribuzione degli errori, si ricorre all'utilizzo di metodi grafici i quali possono essere istogrammi, Box-Plot o Q-Q Plot.

6.2.3 Verifica linearità

La linearità richiede che la relazione tra le variabili indipendenti e la variabile dipendente, debba essere lineare.

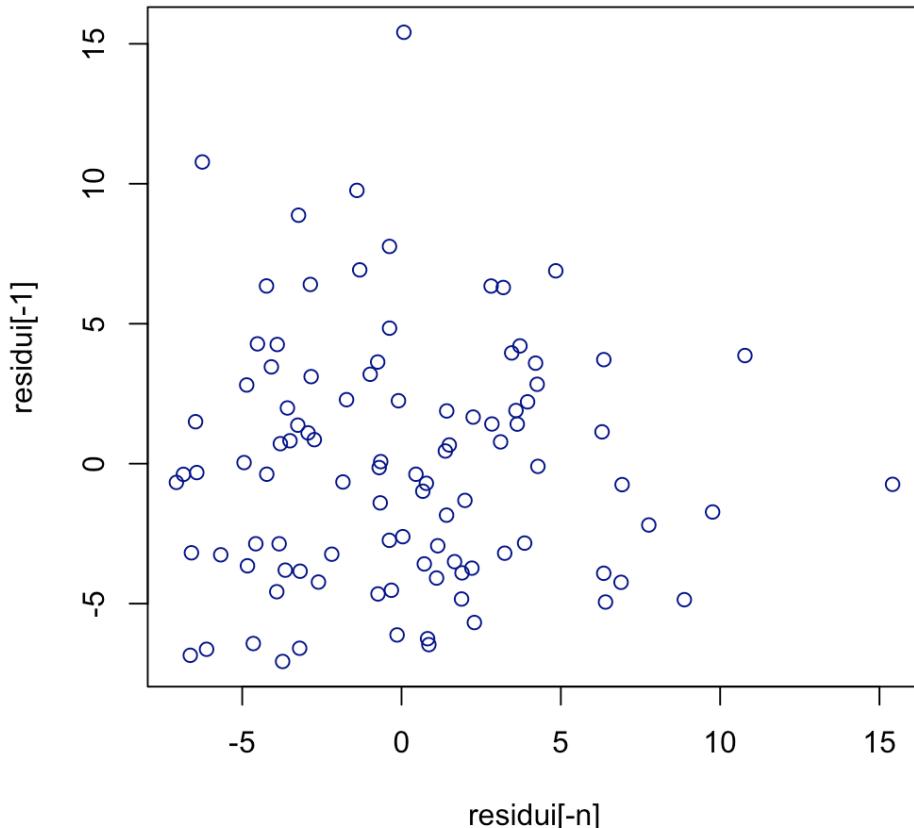
Per verificare la **linearità** occorre tracciare il grafico dei residui (*ordinata*) "verso" i valori previsti (*ascissa*). Secondo l'ipotesi di linearità, i punti dovrebbero distribuirsi in modo casuale intorno allo 0.



In questo grafico è riportata una linea orizzontale tratteggiata in corrispondenza dei residui con media zero. La linea rossa, invece, è una linea di tendenza. Se la linea rossa segue quasi pedissequamente quella tratteggiata, come in questo caso, allora l'ipotesi di linearità è verificata.

6.2.4 Verifica indipendenza

Per verificare l'**assenza di correlazione** si può tracciare il grafico dei residui (ordinata) "contro" i residui precedenti (ascissa) che non dovrebbe rilevare alcun pattern evidente.



A supporto, sappiamo che la **statistica di Durbin-Watson** è un test statistico utilizzato per rilevare la presenza di autocorrelazione dei residui in un'analisi di regressione. Il valore della statistica di Durbin-Watson è sempre compreso tra 0 e 4:

- un valore di 2 indica che non è presente alcuna autocorrelazione,
- valori piccoli che emergono dal test indicano che i residui successivi sono, in media, vicini in valore l'uno all'altro, o correlati positivamente,
- valori grandi indicano che i residui successivi sono, in media, molto differenti in valore l'uno dall'altro, o correlati negativamente.

L'esito del test ha dimostrato l'assenza di autocorrelazione, producendo un risultato pari a

$$DW = 1.9829 \quad (6.3)$$

Di seguito è riportato l'output:

```
Durbin-Watson test

data: modello
DW = 1.9829, p-value = 0.4739
alternative hypothesis: true autocorrelation is greater than 0
```

CAPITOLO 7

CONVALIDA DEL MODELLO

Per corroborare il modello di regressione individuato nei capitoli precedenti, si è deciso di confrontarlo con un nuovo modello ottenuto mediante l'uso di algoritmi per la selezione delle variabili come:

- **Stepwise selection**
- **Backward selection**
- **Forward selection**

L'**obiettivo** di tali algoritmi è quello di includere nel modello solo quelle variabili esplicative che apportano un contributo significativo alla variazione della variabile di risposta (corrispondente alle *prestazioni software del calcolatore* nel nostro caso).

Si noti che in generale, incrementando il numero dei regressori inseriti nel modello, la *devianza dei residui* tende a diminuire (*RSS*). Si deve anche considerare che alcune *variabili apparentemente poco rilevanti* potrebbero risultare statisticamente significative e quindi venire incluse nel modello anche per fattori dovuti al caso. Viceversa, *variabili esplicative logicamente fondamentali*, potrebbero risultare statisticamente non significative ed essere così escluse dal modello.

Gli algoritmi presentati necessitano tutti di un criterio sulla base del quale confrontare due diversi modelli e stabilire quali variabili esplicative utilizzare. Il problema **precedentemente evidenziato** viene risolto mediante l'introduzione dei seguenti indici: l'**Asymptotic Information Criterion (AIC) di Akaike** oppure il **Bayesian Information Criterion (BIC) di Schwarz**, entrambi basati sulla funzione di log-verosimiglianza. Le loro espressioni sono le seguenti:

$$AIC = -2l(\hat{\theta}) + 2(p + 1) \quad BIC = -2l(\hat{\theta}) + (p + 1) \log(n) \quad (7.1)$$

L'AIC è un indice numerico che permette di stimare la quantità di informazione persa in un modello cercando di coniugare l'esigenza di avere un'elevata bontà di

adattamento con quella di favorirne la semplicità. **Dati due modelli M1 ed M2 è da preferirsi quello avente l'indice AIC più basso.**

Vediamo ora nel dettaglio il funzionamento dei sopracitati algoritmi per la selezione delle variabili:

1. **Backward elimination:** si parte considerando il modello che include tutte le variabili a disposizione. Si rimuove la variabile che, se rimossa, porterebbe ad un abbassamento dell'AIC.

Utilizzando il comando `step()`, si è ottenuto:

```
Start: AIC=392.87
prestazioniCalc ~ cpu + HardDisk + processi + aging + audio +
                    ram

      Df Sum of Sq    RSS    AIC
- HardDisk  1     0.05 4419.8 390.87
<none>          4419.7 392.87
- audio     1     94.25 4514.0 392.98
- ram       1     151.75 4571.5 394.24
- processi  1     486.28 4906.0 401.30
- cpu       1    1675.60 6095.3 423.01
- aging     1    1889.01 6308.7 426.45
```

Il che significa che l'unico regressore che consiglia di eliminare dal modello la Backward Elimination è *HardDisk*, in quanto è l'unico a trovarsi al di sopra della riga con `<none>`. È emerso dunque che il modello consigliato è costituito dai regressori *audio*, *ram*, *processi*, *cpu* e *aging*.

2. **Forward selection** si parte da un modello senza alcun regressore, ad ogni iterazione si aggiunge la variabile indipendente che porta il modello ad un AIC più basso se questa è presente, altrimenti si considera il modello ottenuto.

Alla prima iterazione, si ottiene il seguente risultato:

```
Start: AIC=452.87
prestazioniCalc ~ 1

      Df Sum of Sq    RSS    AIC
+ aging     1   2094.08 6986.6 428.66
+ cpu       1   1913.79 7166.8 431.21
+ processi  1   445.25 8635.4 449.85
+ HardDisk  1   276.79 8803.8 451.78
+ ram       1   236.95 8843.7 452.23
<none>          9080.6 452.87
+ audio     1   125.91 8954.7 453.48
```

Aggiungere uno qualsiasi dei regressori presenti sopra la riga indicata come <none> migliorerebbe il modello, mentre quelli al di sotto andrebbero a peggiorarlo.

```
Step: AIC=390.87
prestazioniCalc ~ aging + cpu + processi + ram + audio

Df Sum of Sq    RSS    AIC
<none>             4419.8 390.87
+ HardDisk  1  0.045667 4419.7 392.87
:
```

Ciò che si conviene fare è fermarsi all'iterazione che presenta <none> in cima ed eliminare tutti i regressori sottostanti. In questo caso l'unico regressore eliminato dalle iterazioni è stato *HardDisk*, e dunque il modello conveniente secondo la Forward Selection prevede i regressori *aging*, *cpu*, *processi*, *ram* e *audio*.

7.1 Scelta del modello tramite regressione stepwise

La **Stepwise regression** è una combinazione dei due criteri precedenti. La selezione dei regressori da includere nel modello avviene come nella forward selection. Aggiungendo successivamente una nuova variabile, i coefficienti di regressione delle variabili già incluse potrebbero risultare singolarmente non significativi a causa della forte correlazione con la nuova variabile. Pertanto, dopo l'inserimento di ciascuna variabile, il modello viene riconsiderato per verificare se vi è qualche variabile da eliminare (come nella backward elimination). Ad una prima iterazione si ottiene:

```
Start: AIC=452.87
prestazioniCalc ~ 1

Df Sum of Sq    RSS    AIC
+ aging      1  2094.08 6986.6 428.66
+ cpu        1  1913.79 7166.8 431.21
+ processi   1   445.25 8635.4 449.85
+ HardDisk   1   276.79 8803.8 451.78
+ ram        1   236.95 8843.7 452.23
<none>                  9080.6 452.87
+ audio      1   125.91 8954.7 453.48
```

Da quanto risulta, il miglior regressore da aggiungere al modello tenendo conto del desiderio di abbassare il livello di AIC è *aging*.

Step: AIC=428.66
 prestazioniCalc ~ aging

	Df	Sum of Sq	RSS	AIC
+ cpu	1	1775.33	5211.2	401.34
+ processi	1	396.21	6590.4	424.82
+ audio	1	329.03	6657.5	425.83
+ ram	1	172.26	6814.3	428.16
<none>			6986.6	428.66
+ HardDisk	1	107.45	6879.1	429.11
- aging	1	2094.08	9080.6	452.87

Alla seconda iterazione, viene consigliata l'aggiunta del regressore *cpu* e si sconsiglia di aggiungere il regressore *HardDisk*, cosa che resta vera fino all'ultima iterazione, dove si ottiene:

Step: AIC=390.87
 prestazioniCalc ~ aging + cpu + processi + ram + audio

	Df	Sum of Sq	RSS	AIC
<none>			4419.8	390.87
- audio	1	94.32	4514.1	390.98
- ram	1	152.97	4572.7	392.27
+ HardDisk	1	0.05	4419.7	392.87
- processi	1	486.82	4906.6	399.32
- cpu	1	1723.20	6143.0	421.79
- aging	1	1922.26	6342.0	424.98

In conclusione il criterio di Stepwise Regression consiglia di annettere al modello i regressori: *aging*, *cpu*, *processi*, *ram*, *audio*.

Tutti e tre gli algoritmi in esame hanno fornito lo stesso risultato, ossia hanno assicurato che i regressori più rappresentativi del modello sono *aging*, *cpu*, *processi*, *ram* e *audio*.

Tuttavia, da un'ulteriore analisi di regressione si è appurato che questo modello non sia il migliore per aderenza ai dati, rispetto al campione in esame. Si riporta di seguito il risultato che si è ottenuto:

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	38.7223	0.6857	56.471	< 2e-16 ***
aging	-4.4733	0.6996	-6.394	6.20e-09 ***
cpu	4.2563	0.7031	6.054	2.88e-08 ***
processi	2.2280	0.6924	3.218	0.00177 **
ram	1.2500	0.6930	1.804	0.07448 .
audio	1.0050	0.7096	1.416	0.15999

Signif. codes:	0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1			

Residual standard error: 6.857 on 94 degrees of freedom
 Multiple R-squared: 0.5133, Adjusted R-squared: 0.4874
 F-statistic: 19.83 on 5 and 94 DF, p-value: 1.915e-13

Il valore di R^2 non risulta sufficientemente soddisfacente ed inoltre la relazione che intercorre fra la variabile dipendente ed i processi, sembra essere ben descritta da una relazione quadratica. Pertanto, a seguito di varie iterazioni, si era giunti al seguente modello:

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	43.6273	0.7203	60.572	< 2e-16 ***
cpu	5.1754	0.4577	11.307	< 2e-16 ***
ram	2.4386	0.4720	5.166	1.36e-06 ***
processi	1.8369	0.4642	3.957	0.000148 ***
I(processi^2)	-5.0523	0.5687	-8.884	4.68e-14 ***
aging	-4.2156	0.4551	-9.263	7.43e-15 ***
cpu:ram	-3.5206	0.4838	-7.277	1.07e-10 ***

Signif. codes:	0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1			

Residual standard error: 4.492 on 93 degrees of freedom
 Multiple R-squared: 0.7933, Adjusted R-squared: 0.78
 F-statistic: 59.5 on 6 and 93 DF, p-value: < 2.2e-16

Il suddetto modello è sicuramente più aderente al campione in esame rispetto al modello lineare comprendente solo i regressori risultati significativi per gli algoritmi di regressione. Tuttavia, si può notare come i regressori considerati in questo modello siano tutti tra quelli risultati significativi a seguito dell'applicazione dei vari algoritmi esaminati in precedenza.

Infatti, applicando l'algoritmo di regressione stepwise al modello finale si ottiene:

```

Start: AIC=307.2
prestazioniCalc ~ cpu * ram + aging + processi + I(processi^2) +
cpu + ram

          Df Sum of Sq    RSS    AIC
<none>                 1876.6 307.20
- processi      1     315.96 2192.5 320.76
- cpu:ram       1     1068.45 2945.0 350.27
- I(processi^2) 1     1592.76 3469.3 366.66
- aging         1     1731.24 3607.8 370.57

```

Questo significa che tutti i regressori presenti nel modello sono significativi al fine di descrivere il caso in esame. Inoltre, tutti i metodi fin ora utilizzati, sono stati condotti mediante il criterio *AIC*.

Il regressore *audio* non compare nel modello in quanto il suo ρ -value è superiore al livello di significatività e quindi non aderente ed inoltre, anche aggiungendolo, viene chiaramente rimosso dall'algoritmo di regressione *stepwise*.

```

Step: AIC=307.2
prestazioniCalc ~ cpu + ram + aging + processi + I(processi^2) +
cpu:ram

          Df Sum of Sq    RSS    AIC
<none>                 1876.6 307.20
+ audio      1     1.20 1875.4 309.14
- processi   1     315.96 2192.5 320.76
- cpu:ram    1     1068.45 2945.0 350.27
- I(processi^2) 1     1592.76 3469.3 366.66
- aging      1     1731.24 3607.8 370.57

```

7.2 Confronto tra modelli

Per confrontare due o più modelli di regressione che differiscono per il numero di variabili esplicative inserite si usa l'**ANOVA**, che mette in evidenza se le variabili in più o in meno di un modello rispetto all'altro apportano o meno un contributo significativo nello spiegare la variabile di risposta.

Risultato di tale analisi è quanto segue

```

Analysis of Variance Table

Model 1: prestazioniCalc ~ aging + cpu + processi + ram + audio
Model 2: prestazioniCalc ~ cpu * ram + aging + processi + I(processi^2) +
cpu + ram
  Res.Df    RSS Df Sum of Sq    F    Pr(>F)
  1     94 4419.8
  2     93 1876.6  1    2543.2 126.04 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Come si evince facilmente, l'aggiunta dell'interazione $cpu * ram$ e di $processi^2$ apporta un contributo significativo (***) nello spiegare la variabile di risposta, che nel nostro caso ricordiamo essere *prestazioni software del calcolatore*.

Per verificare l'importanza o meno del regressore *audio* in questo modello, potremmo fare un ulteriore test.

Si ottiene:

```
Analysis of Variance Table

Model 1: prestazioniCalc ~ cpu * ram + aging + processi + I(processi^2) +
          audio + cpu + ram
Model 2: prestazioniCalc ~ cpu * ram + aging + processi + I(processi^2) +
          cpu + ram
Res.Df   RSS Df Sum of Sq    F Pr(>F)
1     92 1875.4
2     93 1876.6 -1   -1.2026 0.059 0.8086
```

In questo caso, al contrario del caso precedente, l'aggiunta del regressore *audio* non apporta alcun contributo nello spiegare la variabile di risposta.

A parità di R^2 , si ritiene ancora il modello trovato in precedenza come il migliore per aderenza al campione in esame.

7.3 Conclusioni

Il modello finale a cui si è giunti, è il seguente:

$$Y = 43.63 + 5.18cpu + 2.44ram - 3.52cpu * ram + 1.84processi - 5.05processi^2 - 4.22aging + \epsilon$$

$$R^2 = 0.7933$$

$$Var(\epsilon) = 20.17835$$

$$RSS = 1876.586$$

