

# Analisi delle Prestazioni dei Supercomputer

Progetto per il corso di Statistica del Prof. Marco Romito

*Rambod Rahmani*

Corso di Laurea Magistrale in  
Artificial Intelligence and Data Engineering

22 Novembre 2020

## Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
<b>2</b>	<b>Dati</b>	<b>1</b>
2.1	Contenuto della tabella . . . . .	2
2.2	Importazione e pulizia . . . . .	2
2.3	Esplorazione e comprensione dei dati . . . . .	3
<b>3</b>	<b>Analisi</b>	<b>3</b>
3.1	Modello di Regressione Lineare . . . . .	3
3.1.1	Analisi dei Residui . . . . .	3
3.2	Modello di Regressione Esponenziale . . . . .	7
3.2.1	Analisi dei Residui . . . . .	7
<b>4</b>	<b>Conclusioni</b>	<b>7</b>

## 1 Introduzione

Lo scopo della presente analisi è quello di costruire un modello di regressione lineare multivariata per poter prevedere le prestazioni di un Supercomputer a partire dalle specifiche delle sue caratteristiche hardware. A partire dalla tabella dei dati, tramite l'utilizzo di R, sono stati valutati due modelli di regressione. In particolare, uno di regressione lineare e uno di tipo non lineare.

Per quanto riguarda il contesto applicativo ipotizzato, esistono due benchmarks essenziali che vengono usati per valutare le prestazioni di questo tipo di calcolatori, il LINPACK benchmark (una misura delle prestazioni computazionali per operazioni in virgola mobile) e il HPCG benchmark (a complemento del LINPACK, valuta le prestazioni dei sotto sistemi del calcolatore). I costruttori di tali computer forniscono di solito il parametro teorico (calcolato su carta) chiamato **Rpeak** che indica le prestazioni massime teoriche. Dato che per eseguire un test LINPACK è necessario calibrare ben oltre 18 parametri, sarebbe interessante poter stimare il valore di **Rmax** (prestazioni raggiunte nel LINPACK test) tramite un modello statistico.

## 2 Dati

La tabella dei dati è stata scaricata dal sito dell'organizzazione **TOP500**. La TOP500 mantiene una graduatoria, ordinata secondo le loro prestazioni, dei Supercomputer attualmente installati e in funzione. Tale graduatoria viene aggiornata con cadenza semestrale.

Link di download diretto: [https://www.top500.org/lists/top500/2020/11/download/TOP500\\_202011.xlsx](https://www.top500.org/lists/top500/2020/11/download/TOP500_202011.xlsx)

Credenziali di accesso:

Login: rambodrahmani@yahoo.it  
Password: GCgFH6yuZYFMeCr

## 2.1 Contenuto della tabella

La tabella dei dati contiene 37 colonne per un totale di 500 osservazioni. Per la presente analisi ho utilizzato le seguenti colonne:

- **Total Cores**: numero totale di cores;
- **Accelerator/Co-Processor Cores**: numero totale di cores del co-processore;
- **Rmax [TFlop/s]**: massime prestazioni raggiunte nel benchmark LINPACK;
- **Rpeak [TFlop/s]**: massime prestazioni teoriche;
- **HPCG [TFlop/s]**: massime prestazioni raggiunte nel benchmark HPCG (High Performance Conjugate Gradient);
- **Power [kW]**: potenza consumata;
- **Processor Speed [MHz]**: velocità processore;
- **Cores per Socket**: numero di cores per socket.

## 2.2 Importazione e pulizia

Sui dati, non è stata effettuata alcuna operazione precedente la loro importazione in R. Il file originale, in formato `.xlsx`, è stato però convertito in `.csv` per facilitare l'importazione. Prima di iniziare l'analisi, ho rimosso le colonne che ritengo che non influenzano le prestazioni di un Supercomputer ("Name", "Manufacturer", "Country", "Year" ecc...), mentre come fattore di uscita per la predizione utilizzerò il valore della colonna "Rmax".

Nelle colonne rimanenti, ci sono 351 valori mancanti in "Accelerator/Co-Processor Cores", 426 in "HPCG [TFlop/s]" e 310 in "Power [kW]". Dato che il numero di valori mancanti è elevato rispetto al totale delle 500 osservazioni, le suddette colonne sono state eliminate.

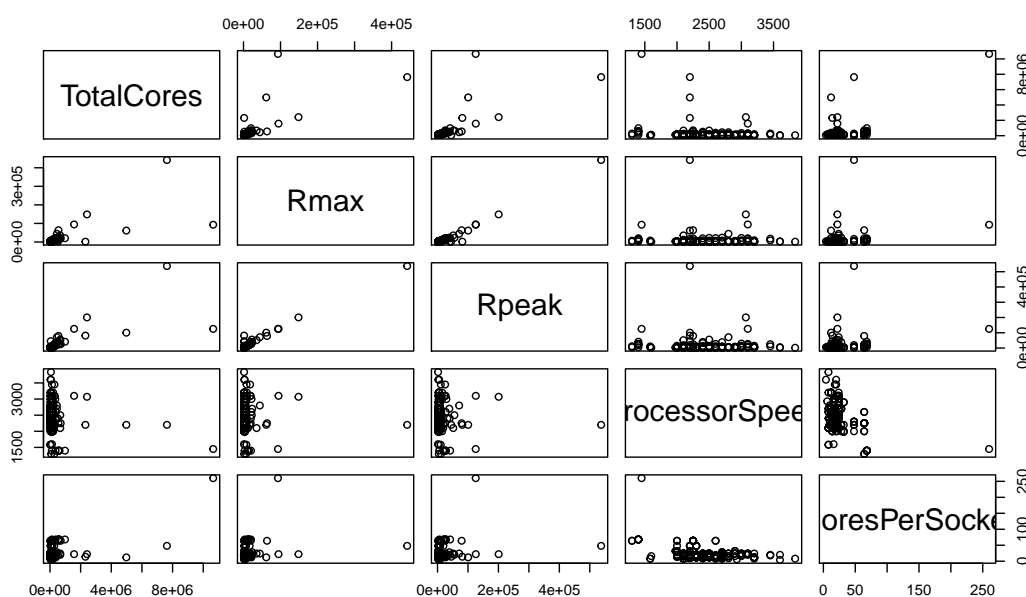


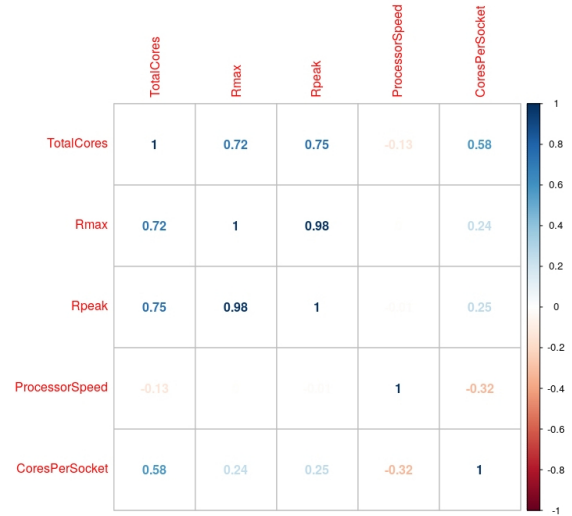
Figure 1: Diagramma di dispersione

## 2.3 Esplorazione e comprensione dei dati

Il primo approccio esplorativo dei dati è stato di tipo grafico: nonostante si possa palesemente vedere una forte correlazione tra il fattore di uscita **Rmax** e i fattori **TotalCores** e **Rpeak**, ho preferito iniziare con un modello di regressione lineare multipla usando **Rmax** come output e tutte le altre variabili come predittori. Ho poi proceduto con l'eliminazione dei fattori valutando per ogni modello ottenuto i seguenti valori:

- $R^2$ : Proporzione di Varianza Spiegata dal Modello;
- $R^2_{Adj}$ : Proporzione di Varianza Spiegata dal Modello corretto;
- **p-value** globale e dei singoli coefficienti.

Figure 2: Matrice di dispersione



## 3 Analisi

### 3.1 Modello di Regressione Lineare

Come primo passo, ho costruito 4 differenti modelli di regressione lineare valutando di volta in volta il valore di  $R^2$ ,  $R^2_{Adj}$ , **p-value**. Si può osservare un netto calo di entrambi i grafici tra i punti di ascissa 3 e 4: dunque la terza versione è la migliore delle quattro regressioni effettuate. Con una proporzione di varianza spiegata dal modello maggiore del 96% e con

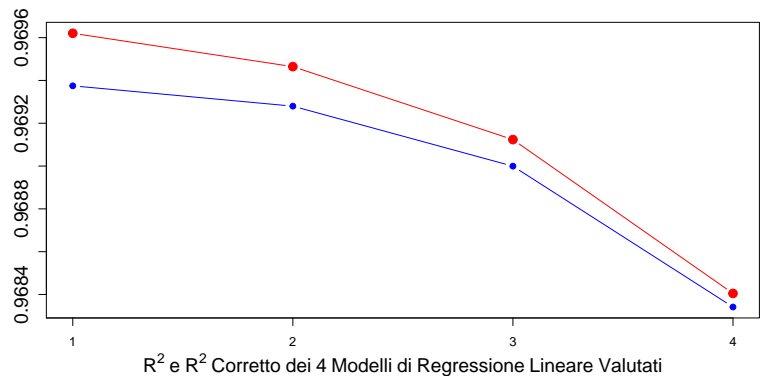
```
> summary(lm)

Call:
lm(formula = Rmax ~ Rpeak + TotalCores, data = data)

Residuals:
    Min       1Q   Median       3Q      Max
-59167    -74     478    1132   23307

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.101e+03  1.804e+02  -6.103  2.09e-09 ***
Rpeak        8.012e-01  9.450e-03  84.782  < 2e-16 ***
TotalCores   -1.392e-03  4.091e-04  -3.401  0.000724 ***
---
Signif. codes:  0   ***    0.001   **    0.01   *    0.05

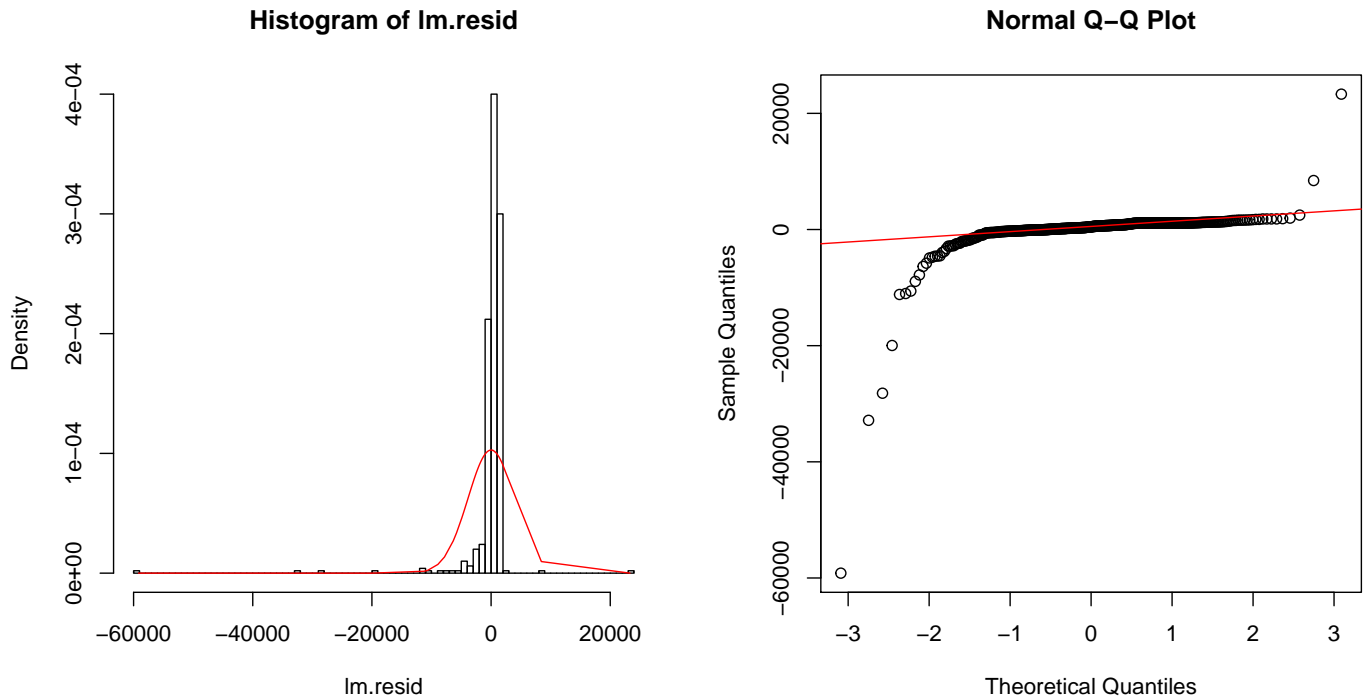
Residual standard error: 3889 on 497 degrees of freedom
Multiple R-squared:  0.9691, Adjusted R-squared:  0.969
F-statistic: 7800 on 2 and 497 DF, p-value: < 2.2e-16
```



dei p-value quasi nulli (sia per quanto riguarda i singoli coefficienti che quello globale), possiamo concludere che il modello di regressione che abbiamo ottenuto cattura buona parte del problema ed è statisticamente significativo.

#### 3.1.1 Analisi dei Residui

Soddisfatto del modello ho proceduto con l'analisi dei residui. Dal diagramma di dispersione, dall'istogramma, dal QQ-plot, dal valore della skewness e della kurtosi e dai risultati ottenuti dal Test di Shapiro-Wilk è evidente che la distribuzione dei residui è ben lontana dall'essere la Gaussiana che cerchiamo:



```
> skewness = mean(((lm.resid - mean(lm.resid)) / sd(lm.resid))^3); skewness
[1] -9.094902
> kurtosi = mean(((lm.resid - mean(lm.resid)) / sd(lm.resid))^4) - 3; kurtosi
[1] 125.4671

> shapiro.test(lm.resid)

Shapiro-Wilk normality test

data:  lm.resid
W = 0.28981, p-value < 2.2e-16
```

Per approfondire, ho analizzato la correlazione tra i predittori utilizzati, se eventualmente i residui del modello fossero correlati con uno dei predittori, un modello di regressione con tutti i predittori disponibili e un modello di regressione lineare semplice. I risultati però non sono stati comunque soddisfacenti:

```
> cor(data$Rpeak, data$TotalCores)
[1] 0.7525282
> cor(data$Rpeak, lm.resid)
[1] 1.965037e-15
> cor(data$TotalCores, lm.resid)
[1] 8.458876e-16

> # approfondimento analisi dei residui: valutazione modello di regressione lineare con tutti i fattori
> lm.1.resid = residuals(lm.1)
> shapiro.test(lm.1.resid)

Shapiro-Wilk normality test

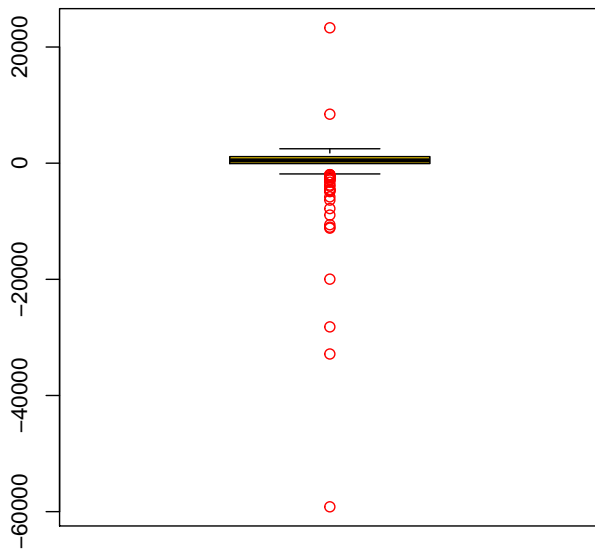
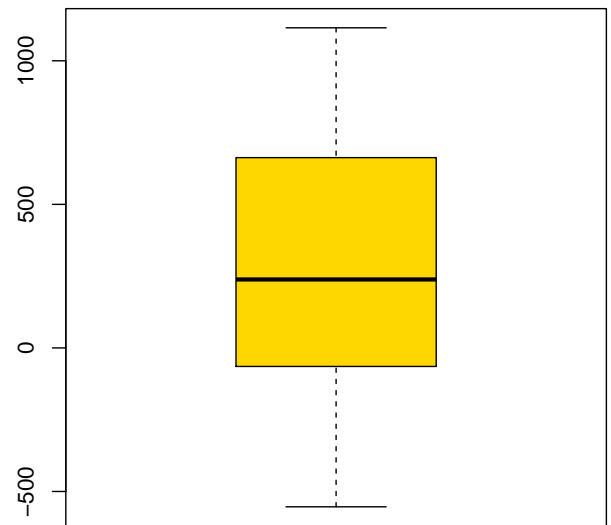
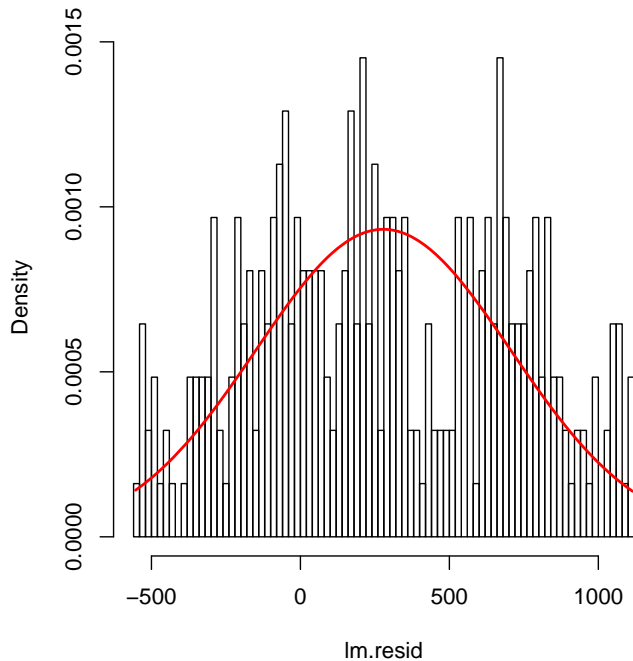
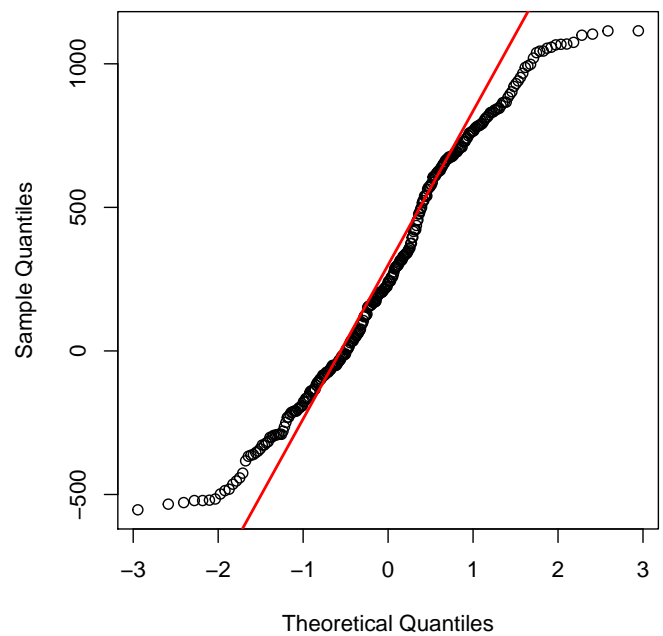
data:  lm.1.resid
W = 0.31715, p-value < 2.2e-16

> # approfondimento analisi dei residui: valutazione modello di regressione lineare semplice
> lm.4.resid = residuals(lm.4)
> shapiro.test(lm.4.resid)

Shapiro-Wilk normality test

data:  lm.4.resid
W = 0.27671, p-value < 2.2e-16
```

La mia ipotesi allora, dato che la distribuzione è comunque centrata in zero e ricorda l'andamento di una Gaussiana, è stata che fosse a causa della presenza di alcuni valori numerici eccessivamente elevati rispetto alla media. Dopo aver rimosso questo primo sottoinsieme di residui, ho notato che probabilmente fosse necessario rimuovere una parte dei residui che causavano una forte deviazione da una distribuzione Gaussiana. Così facendo ho ottenuto un modello certamente migliore ma comunque non perfetto:

**Boxplot residui iniziali****Boxplot residui dopo la rimozione degli outliers****Histogram of lm.resid****Normal Q-Q Plot**

```
> skewness = mean(((lm.resid - mean(lm.resid)) / sd(lm.resid))^3)
> skewness
[1] 0.08041151

> kurtosi = mean(((lm.resid - mean(lm.resid)) / sd(lm.resid))^4) - 3
> kurtosi
[1] -1.008218

> shapiro.test(lm.resid)

Shapiro-Wilk normality test

data:  lm.resid
W = 0.97162, p-value = 8.588e-06
```

Da notare comunque la diminuzione del valore del p-value di 10 ordini di grandezza, e il netto miglioramento dei valori di skewness e kurtosi.

### **3.2 Modello di Regressione Esponenziale**

Non contento, nella speranza di riuscire ad ottenere un modello di regressione con un valore di  $R^2$  leggermente minore, ma con una distribuzione dei residui migliore, ho provato una analisi tramite modello di regressione esponenziale.

#### **3.2.1 Analisi dei Residui**

## **4 Conclusioni**