UNIVERSITA' DEGLI STUDI DI NAPOLI FEDERICO II

Scuola Politecnica e delle Scienze di Base
Corso di Laurea Magistrale in Ingegneria Informatica

Progetto di Impianti di Elaborazione

Anno Accademico 2023/2024

Professori
Prof. Domenico Cotroneo
Prof. Pietro Liguori
Prof. Stefano Rosiello

Studenti
Letizia Arena (M63001513)
Leonardo Ferrara (M63001517)
Carmine Palmese (M63001624)

Contents

Capacity Test **4**
Setup sperimentale . . . 4
Descrizione dei test . . . 5
Grafici e analisi . . . 6
Analisi del Bottleneck . . . 6
Fairness Index . . . 7
Test 1 - Allocazione fair . . . 8
Test 2 - Allocazione unfair . . . 8
Test 3 - Allocazione unfair con utenti variabili . . . 9

PCA e Clustering **10**
Analisi del dataset . . . 10
Caratterizzazione PCA+Clustering . . . 19
Caratterizzazione Clustering-only . . . 25

Workload Characterization **26**
Scopo del test . . . 26
Setup del test . . . 26
Caratterizzazione Workload sintetico . . . 29
Caratterizzazione HL . . . 29
Caratterizzazione LL . . . 38
Caratterizzazione LL' . . . 47
Test di ipotesi . . . 50

DoE **53**
Progettazione esperimento . . . 53
Misurazioni e analisi dataset . . . 54
Analisi di importanza . . . 55
Analisi di significatività . . . 56
Normalità . . . 57
Omoschedasticità . . . 57
ANOVA (Test di Kruskal-Wallis) . . . 58

Regression **60**
EXP 1 . . . 60
Stima lineare . . . 60
Controllo della normalità . . . 62
Test di Mann-Kendall . . . 64
Stimatore di Theil-Sen . . . 67
EXP 2 . . . 68
Stima lineare . . . 68
Controllo della normalità . . . 70
Test di Mann-Kendall . . . 72
Stimatore di Theil-Sen . . . 75
OS1 . . . 76
Stima lineare . . . 76
Controllo della normalità . . . 81
Test di Mann-Kendall . . . 85
Stimatore di Theil-Sen . . . 89
OS2 . . . 90
Stima lineare . . . 91
Controllo della normalità . . . 95
Test di Mann-Kendall . . . 99

Stimatore di Theil-Sen . . . 103
OS3 . . . 104
Stima lineare . . . 105
Controllo della normalità . . . 109
Test di Mann-Kendall . . . 113
Stimatore di Theil-Sen . . . 117
Confronto tra i trend individuati nei dataset . . . 118
VMres1 . . . 118
Stima lineare . . . 119
Controllo della normalità . . . 119
Test di Mann-Kendall . . . 120
Stimatore di Theil-Sen . . . 121
Failure prediction . . . 122
VMres2 . . . 122
Stima lineare . . . 123
Controllo della normalità . . . 123
Test di Mann-Kendall . . . 124
Stimatore di Theil-Sen . . . 125
Failure prediction . . . 126
VMres3 . . . 126
Stima lineare . . . 127
Controllo della normalità . . . 127
Test di Mann-Kendall . . . 128
Stimatore di Theil-Sen . . . 129
Failure prediction . . . 130
Confronto tra i risultati ottenuti . . . 130

Benchmark **131**
Misurazioni e raccolta dei dati . . . 131
Verifica dati campionari e test visivi . . . 132
Confronto delle distribuzioni: 10000 . . . 134
Confronto delle distribuzioni: 100000 . . . 135
Confronto delle distribuzioni: 1000000 . . . 137
Test statistico . . . 138
Test di normalità . . . 139
Test di omoschedasticità . . . 141
Test di Wilcoxon per 10000 bodies . . . 143
Two sample t-test unpooled per 10000 bodies . . . 144
Two sample t-test unpooled per 100000 bodies . . . 146
Two sample t-test unpooled per 1000000 bodies . . . 146
Conclusione . . . 147

FFDA **148**
Mercury . . . 148
Sensitivity Analysis . . . 149
Reliability e TTF . . . 150
Sensitivity Analysis per categoria . . . 153
Sensitivity Analysis per nodo . . . 160
BlueGene . . . 173
Sensitivity Analysis . . . 173
Reliability e TTF . . . 174
Sensitivity Analysis per compute card (opzionale) . . . 178
Sensitivity Analysis per nodo . . . 187
Confronti e considerazioni . . . 200

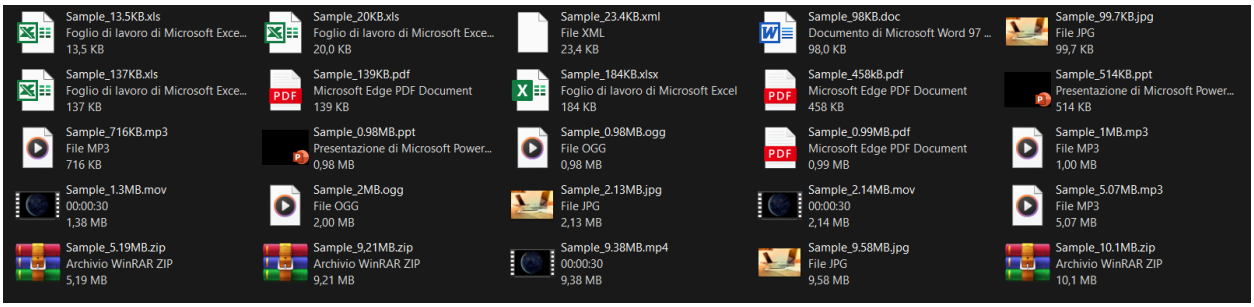
RBD & Fault Tree **201**
Esercizio 1 201
Esercizio 2 204
Esercizio 3 207
Esercizio 4 209
Prima coppia 209
Seconda coppia 210
Terza coppia 212
Quarta coppia 213
Esercizio 5 214
Punto A 215
Punto B 215
Punto C 217
Punto D 217

Capacity Test

Setup sperimentale

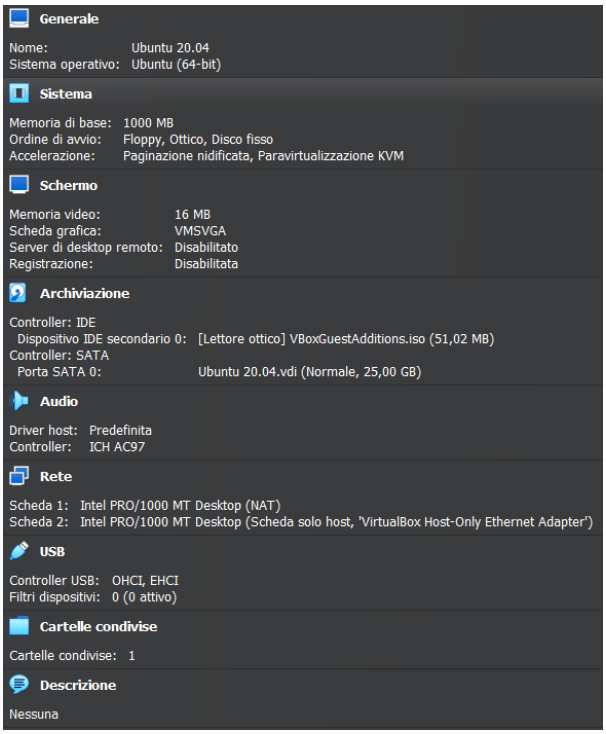
Lo scopo del seguente elaborato è stato quello di caratterizzare la **massima capacità** di un Web Server. A tale scopo, è stata impostata una piattaforma sperimentale su **macchina virtuale** che ospita un sistema operativo **Linux Ubuntu 20.04**, il quale a sua volta ha in esecuzione un servizio **Apache2** che funge da gestore del web server.

Per simulare le richieste di un determinato **carico** sul web server, sono state predisposte una serie di **risorse multimediali** che variano in base alla dimensione. Di seguito sono riportate le risorse in questione:



Si noti che le risorse hanno dimensioni che variano da pochi kiloByte a una decina di MegaByte. La **caratterizzazione della capacity** è un test che richiede un certo numero di queste risorse per valutare la stabilità del sistema a fronte di un tasso di richieste sempre crescente.

Di seguito sono riportate le specifiche tecniche della macchina virtuale su cui è ospitato il web server:



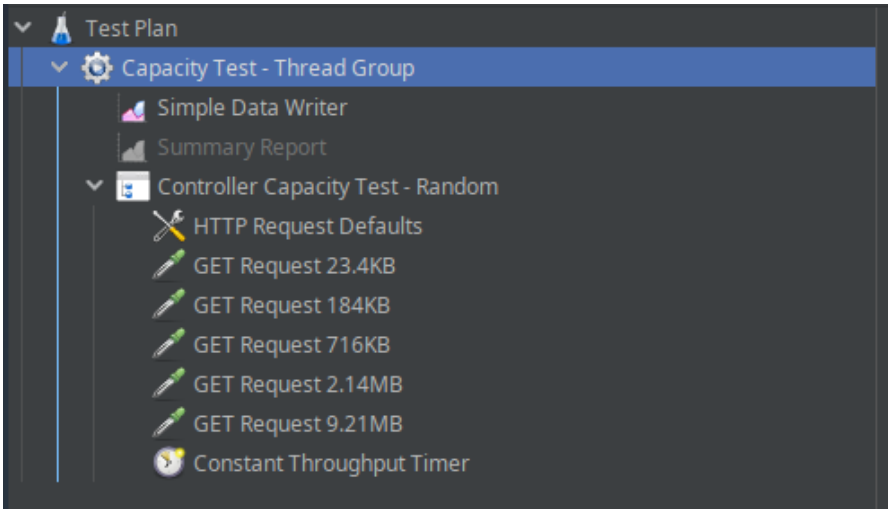
Per l'esecuzione dei test, è stato utilizzato il software **Apache JMeter 5.6.2**, eseguito su una macchina Host con le seguenti caratteristiche:

Processore	AMD Ryzen 7 5825U - 2.00GHz
RAM	8 GB
Sistema Operativo	Windows 11

Descrizione dei test

Il test si compone principalmente di un singolo **Thread Group**. Il gruppo è stato impostato con le seguenti caratteristiche di esecuzione:

- **Numero di thread (user):** 50.
- **Ramp-up period:** 25 secondi.
- **Duration:** 300 secondi.



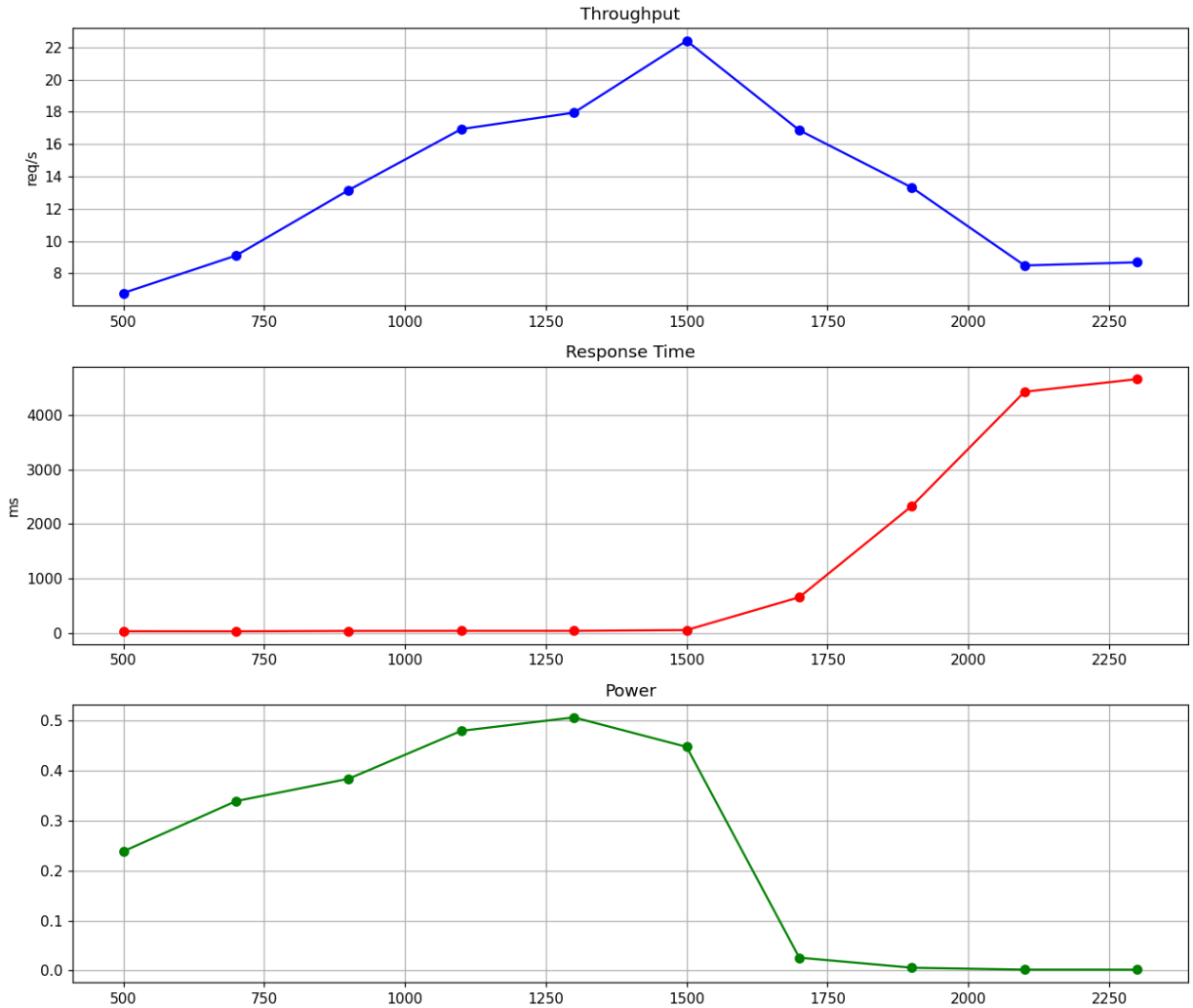
All'interno del gruppo è presente un **Random Controller**, in cui sono state inserite diverse richieste HTTP di risorse, fatte variare per dimensione. Il controller regola il tasso di richieste attraverso un **Constant Throughput Timer** che indica il numero di richieste da effettuare al minuto; quest'ultimo parametro è quello che è stato fatto variare per caratterizzare la capacity del sistema.

Al fine di ottenere dei report esplicativi dell'esecuzione del test, è stato utilizzato un **Simple Data Writer**.

Si è deciso di effettuare **3 iterazioni** per ogni tasso di richieste impostato e di considerare poi i valori di interesse come media sulle iterazioni.

Per quanto riguarda la raccolta di informazioni a **basso livello** (per l'analisi del bottleneck di prestazioni) è stato utilizzato il tool **vmstat**, fatto eseguire per tutta la durata del test sulla macchina virtuale su cui gira il web server. Infine, i report ad alto livello e quelli a basso livello sono stati processati attraverso uno script apposito.

Grafici e analisi



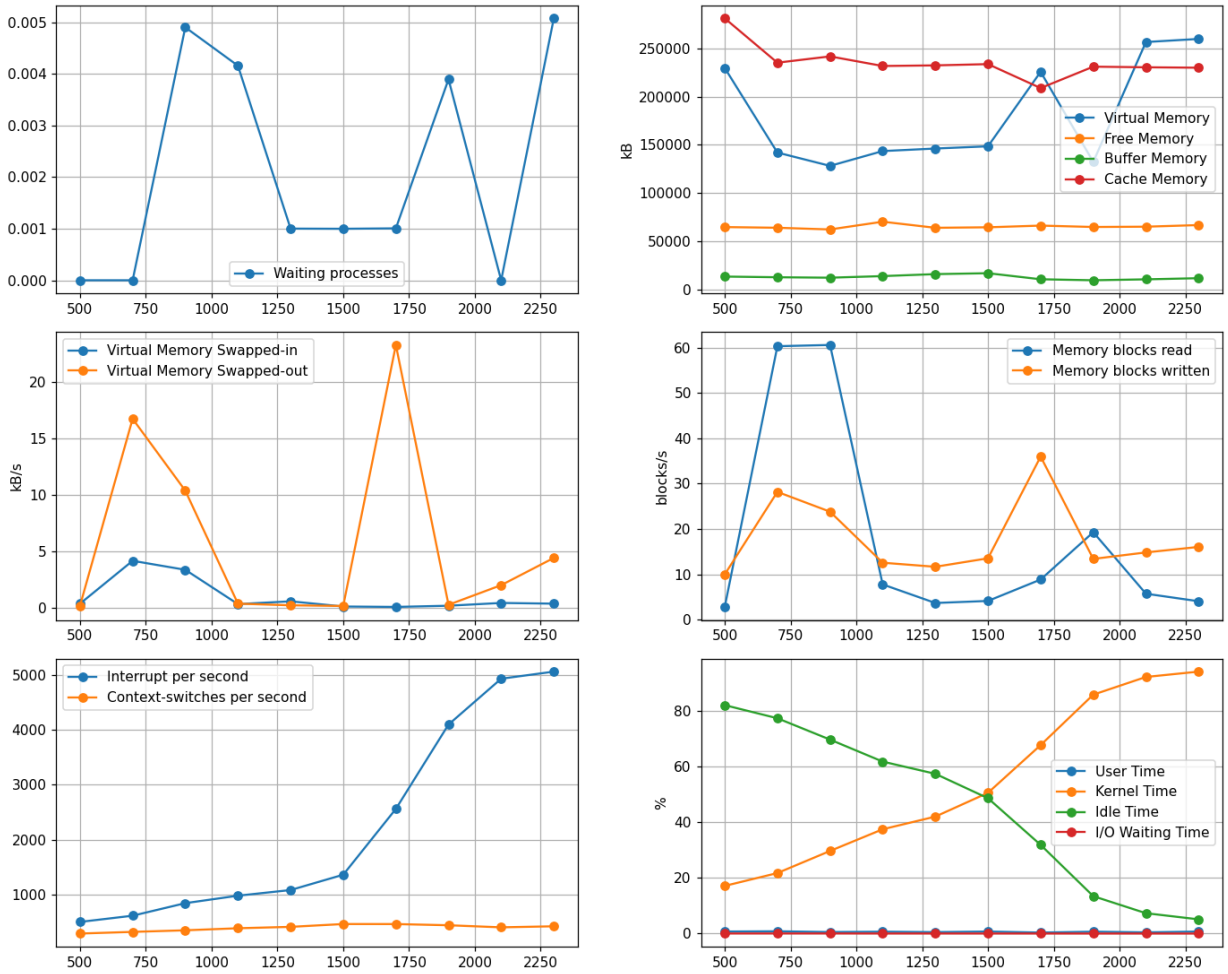
A valle dell'esecuzione dello script, sono stati ottenuti i seguenti grafici, che rappresentano rispettivamente il **throughput**, ovvero il tasso di richieste che il sistema ha potuto servire, il **response time**, ovvero la media dei tempi di risposta delle richieste per quel carico, e la **power**, che non è altro che un valore riassuntivo dei primi due parametri, in particolare:

$$\text{Power} = \frac{\text{Throughput}}{\text{Response Time}}$$

Si hanno quindi i dati sufficienti per caratterizzare **Knee Capacity** e **Usable Capacity**. La Knee Capacity è il punto di **massima power**, che con buona approssimazione si può dire trovarsi al punto di carico 1100. La Usable Capacity è il punto dopo il quale la power comincia drasticamente a decrescere, e lo si trova al punto di carico 1500. Di conseguenza, un buon range di utilizzo del server si trova nell'insieme $[1100, 1500]$ di richieste al minuto.

Analisi del Bottleneck

Si è notato che dopo un carico di 1500 richieste al minuto, il sistema non riesce più a gestire le richieste e ne serve molto meno di quelle che effettivamente dovrebbe servire. Il motivo di questo calo di performance può essere analizzato attraverso un grafico dell'**utilizzo delle risorse a basso livello**.



Si può notare come con l'aumentare del carico richiesto aumenti considerevolmente il **kernel time** rispetto all'**idle time** della CPU. In effetti, essendo Apache WebServer un servizio di sistema Linux, questo era un effetto prevedibile. In particolare si nota come, a partire dal carico 1900, il kernel time supera considerevolmente il 90% dell'utilizzo di CPU.

Poiché le risorse come **memoria virtuale** e **utilizzo del disco** si trovano a intervalli di valori perfettamente nella norma (la memoria virtuale ad esempio non supera mai i 250MB, come anche la cache), si può dire con una buona confidenza che il bottleneck è proprio nell'utilizzo della CPU.

Fairness Index

Per il calcolo del **Fairness Index**, è stato creato un setup di test per la valutazione di vari carichi di richieste, su thread group diversi che eseguono in parallelo e richiedono risorse di dimensioni diverse. In particolare, i thread richiedono rispettivamente:

- una risorsa **LOW** (~10 KB)
- una risorsa **MID** (~1 MB)
- una risorsa **HIGH** (~5 MB)

Sono stati quindi effettuati tre test, in cui sono stati fatti variare i seguenti parametri:

- **1° Test:** numero di utenti costante, CTT **decrescente** al crescere della dimensione delle risorse richieste
  - **TG1:** risorsa LOW, CTT 1100 (10 utenti)
  - **TG2:** risorsa MID, CTT 800 (10 utenti)
  - **TG3:** risorsa HIGH, CTT 500 (10 utenti)

- **2° Test:** numero di utenti costante, CTT **crescente** al crescere della dimensione delle risorse richieste
  - **TG1:** risorsa LOW, CTT 500 (10 utenti)
  - **TG2:** risorsa MID, CTT 800 (10 utenti)
  - **TG3:** risorsa HIGH, CTT 1100 (10 utenti)
- **3° Test:** numero di utenti **crescente** e CTT **decrescente** al crescere della dimensione delle risorse
  - **TG1:** risorsa LOW, CTT 1100 (5 utenti)
  - **TG2:** risorsa MID, CTT 800 (10 utenti)
  - **TG3:** risorsa HIGH, CTT 500 (15 utenti)

I test sono stati condotti per 5 minuti l'uno, e per ogni test sono state effettuate **tre ripetizioni**. Da queste, è stato quindi calcolato il **fairness index** per ogni test:

$$f(x_1, x_2, ..., x_n) = \frac{\left(\sum_{i=1}^{n} x_i\right)^2}{n \sum_{i=1}^{n} x_i^2}$$

dove x_i è il **normalized throughput**, che si calcola come $x_i = \frac{\text{Throughput}}{\text{Required Throughput}}$. È stato a tale scopo realizzato uno script in Python in grado di leggere i risultati di un **Summary Report** di Jmeter e di calcolare il Fairness Index richiesto.

Test 1 - Allocazione fair

In questa situazione di traffico, si è simulata un'allocazione di throughput che fosse **inversamente proporzionale** alla dimensione della richiesta. Questo per mettere a disposizione più banda per quegli utenti che fanno richieste più "leggere", e fornirne meno agli utenti che fanno richieste più "pesanti". Questa situazione ci ha permesso di ottenere un fairness index pari a:

$$x_1 = 0.8635, \quad x_2 = 0.6916, \quad x_3 = 0.3732$$
$$f(x_1, x_2, x_3) = 0.9092$$

Si prenderà quindi questa situazione come riferimento per studiare le successive, in cui le allocazioni delle bande saranno gestite diversamente. Nel complesso si nota come effettivamente i valori dei throughput normalizzati non siano poi così alti, quindi il sistema non è comunque riuscito a gestire la mole totale di richieste. Tuttavia, anche in questa condizione, è comunque riuscito a ripartire in modo "fair" le richieste ai thread group, in quanto effettivamente il primo thread group è quello a throughput complessivo maggiore, mentre il terzo ne riceve di meno.

Test 2 - Allocazione unfair

Questo test è fondamentalmente l'inverso di quello di prima, in quanto l'allocazione di banda è stata effettuata in modo **direttamente proporzionale** alla dimensione della richiesta. Qui gli utenti che fanno richieste più pesanti avranno a disposizione più banda, al contrario di quelli che fanno richieste più leggere. Questo ci ha permesso di ottenere:

$$x_1 = 0.8418, \quad x_2 = 0.5237, \quad x_3 = 0.0785$$
$$f(x_1, x_2, x_3) = 0.7027$$

Si può notare bene come questa situazione sia **peggiore** di quella precedente, in cui si è infatti ottenuto un Fairness Index più alto. Qui il problema è stato principalmente legato al thread group 3, che è quello che aveva il throughput nominale più alto, e non è infatti riuscito a sostenerlo. Si noti inoltre che il secondo thread group non ha subito particolari variazioni nel suo throughput (ce lo si poteva aspettare visto che il suo CTT è rimasto invariato) e che quello che ha giovato più di questa situazione è il primo, che si è avvicinato molto all'ammontare di richieste previsto.

Test 3 - Allocazione unfair con utenti variabili

In questo test si riprende il setup del secondo test, con la differenza che è stato fatto **variare il numero degli utenti** per thread group. In questo modo, si è simulata sì una situazione in cui la banda allocata fosse "unfair" rispetto alla dimensione della risorsa richiesta, ma allo stesso tempo gli utenti interessati a fare richieste pesanti sono di più, e quelli interessati a fare richieste leggere sono di meno. Il risultato ottenuto è quindi:

$$x_1 = 0.9391, \quad x_2 = 0.4934, \quad x_3 = 0.1686$$
$$f(x_1, x_2, x_3) = 0.7406$$

Avendo "bilanciato" la situazione precedente, si ottengono risultati (anche se di poco) migliori. Questo perché probabilmente non c'è più il flooding di richieste di bassa dimensione, ed è stata data più priorità per le richieste più grandi che andavano in starvation nel caso precedente.

PCA e Clustering

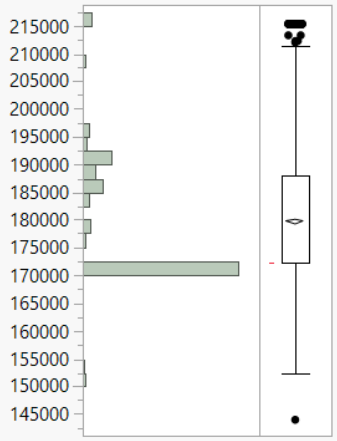
Analisi del dataset

Lo scopo di questo esercizio è stato quello di analizzare l'effetto delle tecniche di PCA e Clustering su un dataset di misurazioni su un dato processore. In particolare, tramite queste tecniche è stato possibile caratterizzare il workload per ottenere un **workload sintetico**. Un esempio di qualche riga del dataset è riportato di seguito:

	'VmPeak'	'VmSize'	'VmHWM'	'VmRSS'	'VmPTE'	'Threads'	'MemFree'	'Buffers'	'Cached'	'Active'	'Inactive'	'Dirty'	'Writeback'	'AnonPages'	'Mapped'
1	144012	126424	14224	14224	108	9	5621452	29064	334600	0	131152	300072	5621452	8159224	156
2	152272	150216	24508	23708	160	9	5610188	29144	334640	0	139780	299928	5610188	8159224	40
3	152272	150216	24628	24588	160	15	5609652	29240	334640	0	140764	300020	5609652	8159224	144
4	152272	150216	24628	24500	160	35	5610232	29336	334640	0	140624	300116	5610232	8159224	176
5	152272	150216	24628	24580	160	29	5610032	29432	334640	0	140704	300212	5610032	8159224	20
6	152272	150216	24628	24500	160	29	5609896	29528	334640	0	140624	300308	5609896	8159224	44
7	152272	150216	24628	24592	160	30	5609648	29632	334640	0	140720	300412	5609648	8159224	24
8	152272	150216	24628	24540	160	24	5609812	29728	334640	0	140668	300508	5609812	8159224	48
9	152272	150216	24628	24524	160	24	5609688	29824	334640	0	140656	300604	5609688	8159224	12
10	152272	150216	24628	24572	160	17	5609380	29920	334640	0	140700	300696	5609380	8159224	172
11	152272	150216	24628	24576	160	10	5609436	30020	334636	0	140700	300792	5609436	8159224	132
12	152272	150216	24628	23824	160	9	5609824	30116	334640	0	140020	300828	5609824	8159224	56
13	152272	150216	24628	24576	160	62	5609152	30212	334640	0	140776	300924	5609152	8159224	24
14	152272	150216	24696	24420	160	54	5609324	30308	334640	0	140620	301012	5609324	8159224	56
15	152272	150216	24696	24492	160	35	5608864	30404	334632	0	140684	301116	5608864	8159224	20
16	152272	150216	24696	24568	160	18	5609256	30500	334640	0	140768	301212	5609256	8159224	36
17	152272	150216	24696	23824	160	9	5609576	30596	334640	0	140024	301308	5609576	8159224	140
18	152272	150216	24696	23824	160	9	5608956	30700	334640	0	140024	301412	5608956	8159224	172
19	152272	150216	25172	25104	160	63	5607780	30796	334644	0	141300	301504	5607780	8159224	20
20	152272	150216	25172	25108	160	55	5607348	30900	334636	0	141312	301612	5607348	8159224	52
21	152272	150216	25172	25108	160	38	5606988	30996	334648	0	141312	301712	5606988	8159224	20
22	152272	150216	25172	25108	160	25	5606960	31092	334648	0	141316	301804	5606960	8159224	48
23	152272	150216	25172	25108	160	12	5607072	31188	334640	0	141320	301888	5607072	8159224	12
24	152272	150216	25172	25108	160	9	5607220	31284	334648	0	141320	301992	5607220	8159224	176
25	152272	150216	25172	25108	160	54	5606612	31388	334640	0	141324	302096	5606612	8159224	20
26	152272	150216	25172	25108	160	44	5606568	31484	334648	0	141324	302184	5606568	8159224	44
27	152272	150216	25172	25108	160	27	5606696	31580	334648	0	141320	302288	5606696	8159224	24
28	152272	150216	25172	25108	160	18	5606776	31684	334640	0	141324	302384	5606776	8159224	56
29	152272	150216	25172	25108	160	9	5606600	31780	334648	0	141324	302488	5606600	8159224	24
30	152272	150216	25172	25104	160	60	5605928	31876	334648	0	141316	302584	5605928	8159224	32

Il tool che è stato usato per l'analisi dati e l'esecuzione di test statistici è il software **JMP**. In primo luogo, per un'analisi superficiale, sono state valutate le **distribuzioni campionarie** di ogni colonna del dataset, per valutare la presenza di colonne che possono essere eliminate ed eventuali outlier che avrebbero compromesso le performance globali della caratterizzazione.

'VmPeak'



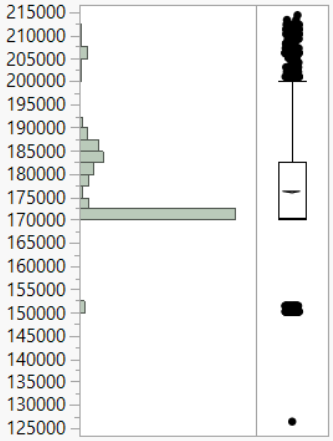
Quantili

100.0%	massimo	215352
99.5%		215352
97.5%		215352
90.0%		192384
75.0%	quartile	188104
50.0%	mediana	172392
25.0%	quartile	172392
10.0%		172388
2.5%		170344
0.5%		152272
0.0%	minimo	144012

Statistiche di riepilogo

Media	179734,08
Dev std	11874,06
Errore std della media	168,00859
Media superiore al 95%	180063,45
Media inferiore al 95%	179404,71
N	4995

'VmSize'



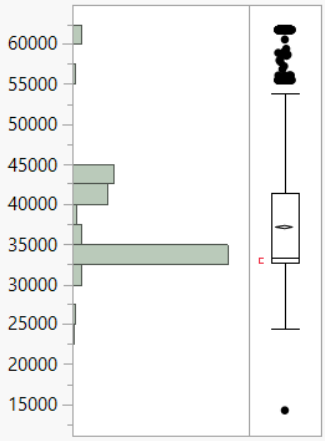
Quantili

100.0%	massimo	214328
99.5%		211256
97.5%		206136
90.0%		186968
75.0%	quartile	182604
50.0%	mediana	170340
25.0%	quartile	170340
10.0%		170340
2.5%		170084
0.5%		150216
0.0%	minimo	126424

Statistiche di riepilogo

Media	176085,69
Dev std	10118,342
Errore std della media	143,16657
Media superiore al 95%	176366,36
Media inferiore al 95%	175805,02
N	4995

'VmHWM'



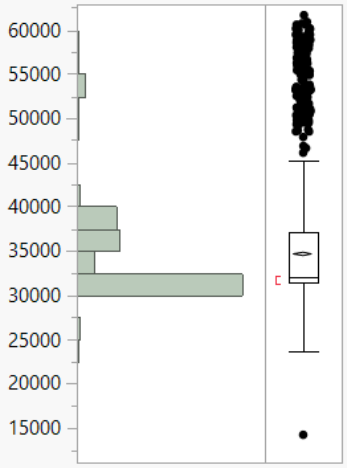
Quantili

100.0%	massimo	61784
99.5%		61784
97.5%		61784
90.0%		43804
75.0%	quartile	41320
50.0%	mediana	33228
25.0%	quartile	32696
10.0%		32696
2.5%		31948
0.5%		25172
0.0%	minimo	14224

Statistiche di riepilogo

Media	37201,467
Dev std	7092,2243
Errore std della media	100,34939
Media superiore al 95%	37398,196
Media inferiore al 95%	37004,738
N	4995

'VmRSS'



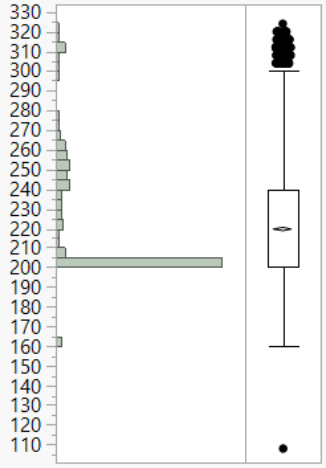
Quantili

100.0%	massimo	61644
99.5%		58892,08
97.5%		53204
90.0%		38848
75.0%	quartile	37040
50.0%	mediana	32116
25.0%	quartile	31508
10.0%		31260
2.5%		30768
0.5%		25104
0.0%	minimo	14224

Statistiche di riepilogo

Media	34611,465
Dev std	5521,5417
Errore std della media	78,125465
Media superiore al 95%	34764,625
Media inferiore al 95%	34458,305
N	4995

'VmPTE'



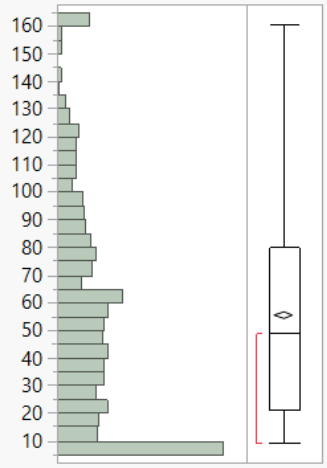
Quantili

100.0%	massimo	324
99.5%		316
97.5%		312
90.0%		260
75.0%	quartile	240
50.0%	mediana	200
25.0%	quartile	200
10.0%		200
2.5%		200
0.5%		160
0.0%	minimo	108

Statistiche di riepilogo

Media	219,49149
Dev std	31,760632
Errore std della media	0,4493879
Media superiore al 95%	220,37249
Media inferiore al 95%	218,61049
N	4995

'Threads'



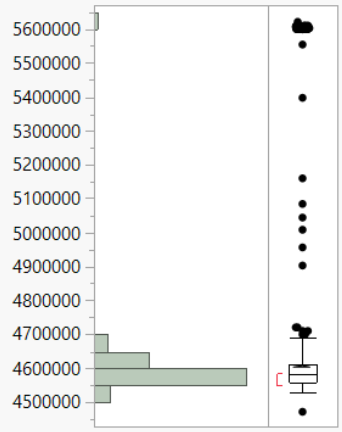
Quantili

100.0%	massimo	161
99.5%		161
97.5%		161
90.0%		114
75.0%	quartile	80
50.0%	mediana	49
25.0%	quartile	21
10.0%		9
2.5%		9
0.5%		9
0.0%	minimo	9

Statistiche di riepilogo

Media	55,411011
Dev std	39,758918
Errore std della media	0,5625574
Media superiore al 95%	56,51387
Media inferiore al 95%	54,308152
N	4995

'MemFree'



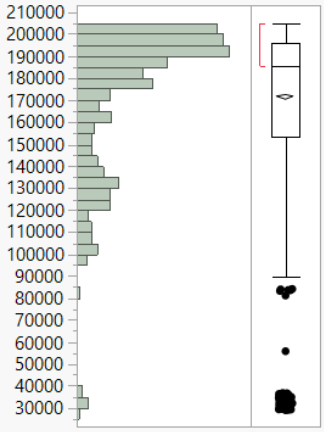
Quantili

100.0%	massimo	5621452
99.5%		5606779,68
97.5%		4668108
90.0%		4645228,8
75.0%	quartile	4614508
50.0%	mediana	4579904
25.0%	quartile	4560000
10.0%		4552982,4
2.5%		4538724
0.5%		4529951,84
0.0%	minimo	4471580

Statistiche di riepilogo

Media	4604927,2
Dev std	135078,85
Errore std della media	1911,2593
Media superiore al 95%	4608674,1
Media inferiore al 95%	4601180,3
N	4995

'Buffers'



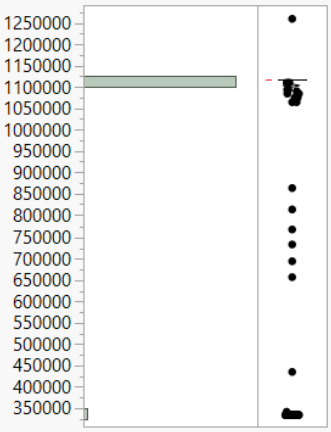
Quantili

100.0%	massimo	204720
99.5%		204628
97.5%		204305,2
90.0%		201898,4
75.0%	quartile	196088
50.0%	mediana	185660
25.0%	quartile	152884
10.0%		122860
2.5%		99520
0.5%		31385,92
0.0%	minimo	29064

Statistiche di riepilogo

Media	171523,37
Dev std	34377,309
Errore std della media	486,41183
Media superiore al 95%	172476,95
Media inferiore al 95%	170569,79
N	4995

'Cached'



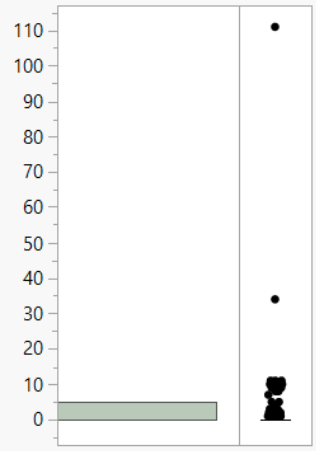
Quantili

100.0%	massimo	1259568
99.5%		1118492
97.5%		1118448
90.0%		1118236
75.0%	quartile	1118040
50.0%	mediana	1117816
25.0%	quartile	1117332
10.0%		1117152
2.5%		1117028
0.5%		334644
0.0%	minimo	334600

Statistiche di riepilogo

Media	1104280,2
Dev std	100745,13
Errore std della media	1425,4643
Media superiore al 95%	1107074,8
Media inferiore al 95%	1101485,7
N	4995

'Active'



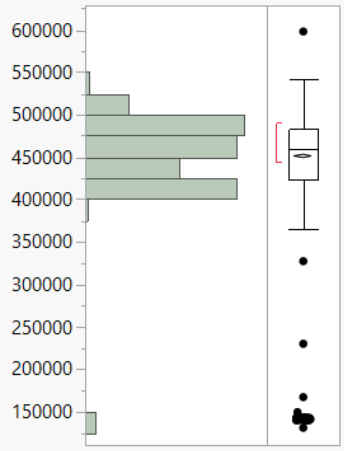
Quantili

100.0%	massimo	111
99.5%		2
97.5%		0
90.0%		0
75.0%	quartile	0
50.0%	mediana	0
25.0%	quartile	0
10.0%		0
2.5%		0
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

Media	0,0602603
Dev std	1,7102287
Errore std della media	0,0241984
Media superiore al 95%	0,1076997
Media inferiore al 95%	0,0128208
N	4995

'Inactive'



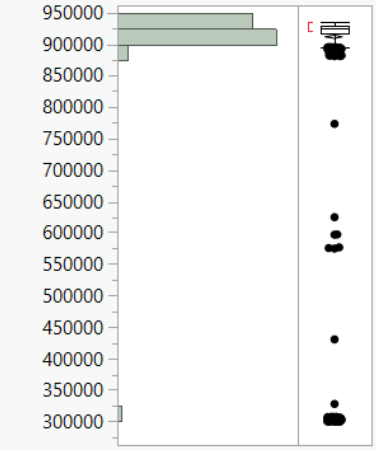
Quantili

100.0%	massimo	597900
99.5%		525456,8
97.5%		520905,2
90.0%		497421,6
75.0%	quartile	482356
50.0%	mediana	458420
25.0%	quartile	422876
10.0%		411564
2.5%		407624
0.5%		141316
0.0%	minimo	131152

Statistiche di riepilogo

Media	451589,16
Dev std	51842,636
Errore std della media	733,53245
Media superiore al 95%	453027,2
Media inferiore al 95%	450151,11
N	4995

'Dirty'



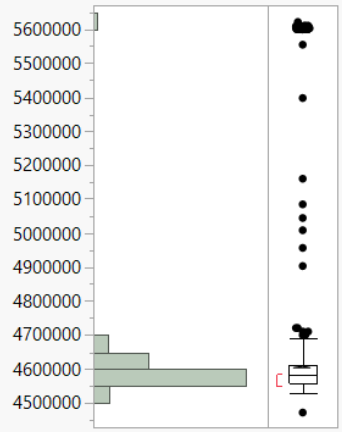
Quantili

100.0%	massimo	935160
99.5%		934988,16
97.5%		934005,2
90.0%		933269,6
75.0%	quartile	929092
50.0%	mediana	923452
25.0%	quartile	915548
10.0%		908098,4
2.5%		893132
0.5%		302093,92
0.0%	minimo	299928

Statistiche di riepilogo

Media	911238,26
Dev std	80205,174
Errore std della media	1134,84
Media superiore al 95%	913463,04
Media inferiore al 95%	909013,48
N	4995

'Writeback'



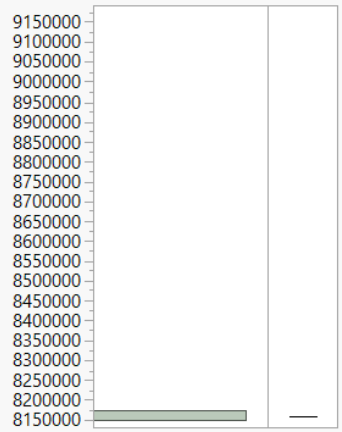
Quantili

100.0%	massimo	5621452
99.5%		5606779,68
97.5%		4668108
90.0%		4645228,8
75.0%	quartile	4614508
50.0%	mediana	4579904
25.0%	quartile	4560000
10.0%		4552982,4
2.5%		4538724
0.5%		4529951,84
0.0%	minimo	4471580

Statistiche di riepilogo

Media	4604927,2
Dev std	135078,85
Errore std della media	1911,2593
Media superiore al 95%	4608674,1
Media inferiore al 95%	4601180,3
N	4995

'AnonPages'



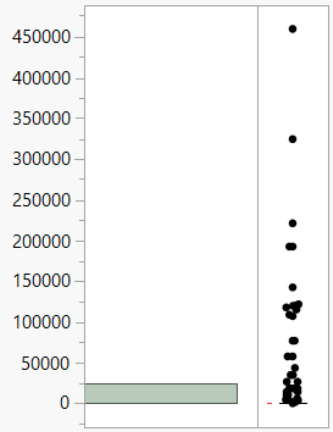
Quantili

100.0%	massimo	8159224
99.5%		8159224
97.5%		8159224
90.0%		8159224
75.0%	quartile	8159224
50.0%	mediana	8159224
25.0%	quartile	8159224
10.0%		8159224
2.5%		8159224
0.5%		8159224
0.0%	minimo	8159224

Statistiche di riepilogo

Media	8159224
Dev std	0
Errore std della media	0
Media superiore al 95%	8159224
Media inferiore al 95%	8159224
N	4995

'Mapped'



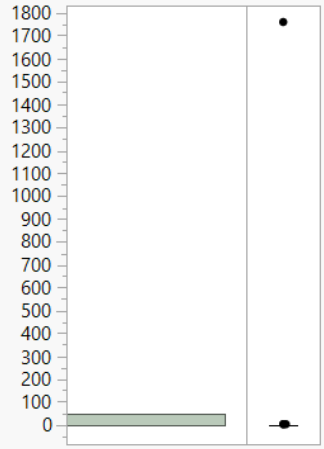
Quantili

100.0%	massimo	459632
99.5%		18645,679999996
97.5%		176
90.0%		168
75.0%	quartile	156
50.0%	mediana	144
25.0%	quartile	36
10.0%		24
2.5%		12
0.5%		4
0.0%	minimo	0

Statistiche di riepilogo

Media	718,69469
Dev std	10883,318
Errore std della media	153,99037
Media superiore al 95%	1020,5834
Media inferiore al 95%	416,80595
N	4995

'Slab'



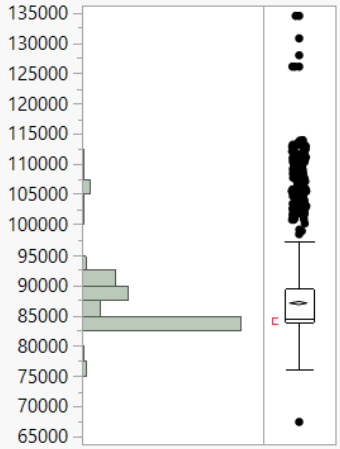
Quantili

100.0%	massimo	1760
99.5%		0
97.5%		0
90.0%		0
75.0%	quartile	0
50.0%	mediana	0
25.0%	quartile	0
10.0%		0
2.5%		0
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

Media	0,3547548
Dev std	24,902772
Errore std della media	0,3523546
Media superiore al 95%	1,0455245
Media inferiore al 95%	-0,336015
N	4995

'PageTables'



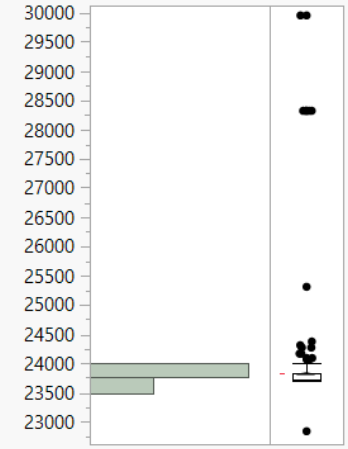
Quantili

100.0%	massimo	134504
99.5%		111752,4
97.5%		105548
90.0%		91196
75.0%	quartile	89384
50.0%	mediana	84460
25.0%	quartile	83844
10.0%		83592
2.5%		83100
0.5%		77363,92
0.0%	minimo	67408

Statistiche di riepilogo

Media	87022,282
Dev std	5759,2002
Errore std della media	81,488145
Media superiore al 95%	87182,035
Media inferiore al 95%	86862,53
N	4995

'Committed_AS'



Quantili

100.0%	massimo	29956
99.5%		23996
97.5%		23848
90.0%		23848
75.0%	quartile	23848
50.0%	mediana	23844
25.0%	quartile	23740
10.0%		23740
2.5%		23740
0.5%		23740
0.0%	minimo	22840

Statistiche di riepilogo

Media	23823,205
Dev std	196,93276
Errore std della media	2,7864434
Media superiore al 95%	23828,667
Media inferiore al 95%	23817,742
N	4995

Quantili

100.0%	massimo	120004
99.5%		117700,08
97.5%		115900
90.0%		114096
75.0%	quartile	113588
50.0%	mediana	110896
25.0%	quartile	110064
10.0%		109290,4
2.5%		108880
0.5%		27812
0.0%	minimo	26160

Statistiche di riepilogo

Media	110233,5
Dev std	10835,772
Errore std della media	153,31764
Media superiore al 95%	110534,07
Media inferiore al 95%	109932,93
N	4995

Quantili

100.0%	massimo	8028
99.5%		7608
97.5%		7320
90.0%		7272
75.0%	quartile	7248
50.0%	mediana	7208
25.0%	quartile	7208
10.0%		7204
2.5%		7196
0.5%		7040
0.0%	minimo	6908

Statistiche di riepilogo

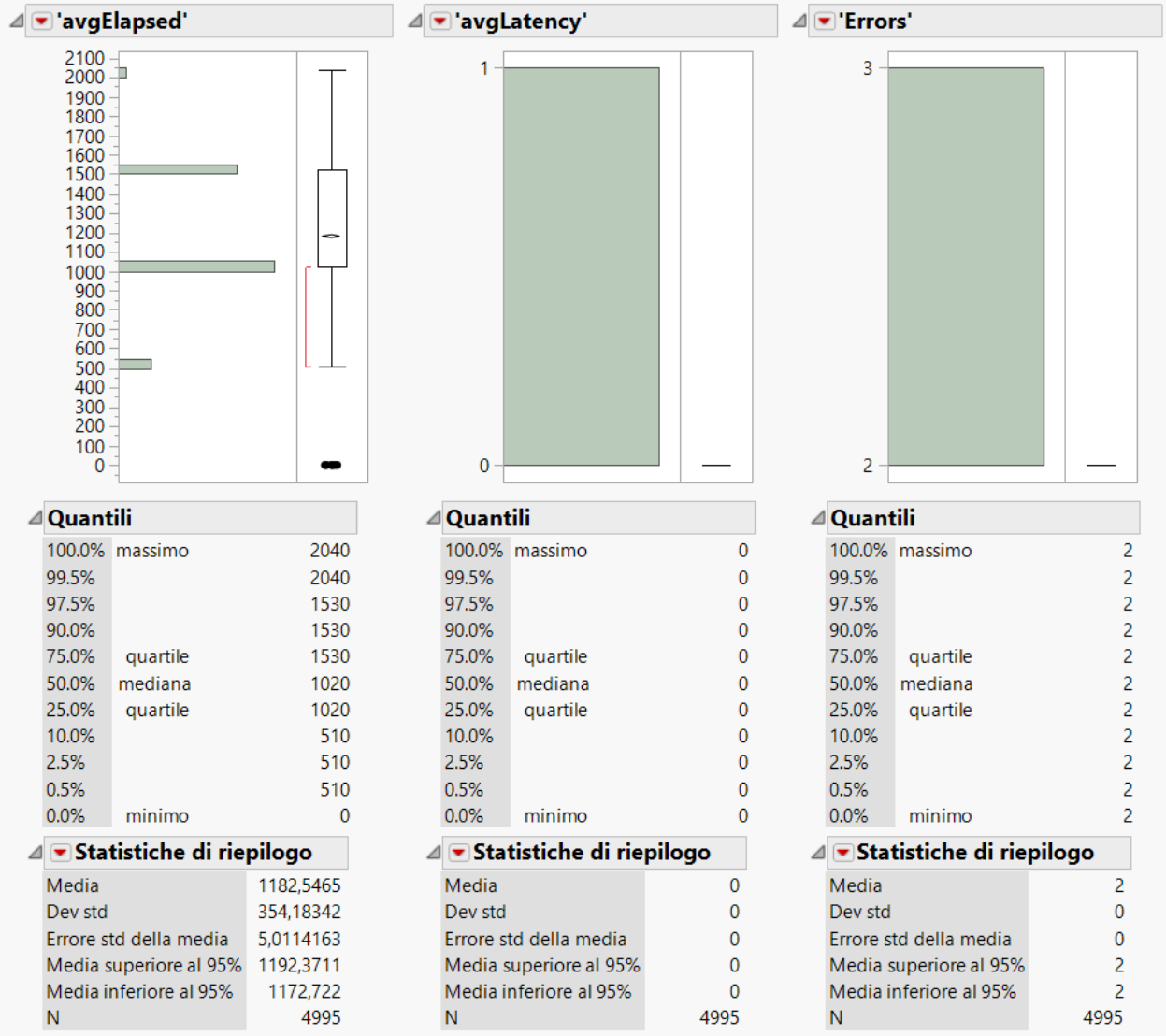
Media	7229,206
Dev std	60,805537
Errore std della media	0,8603504
Media superiore al 95%	7230,8927
Media inferiore al 95%	7227,5193
N	4995

Quantili

100.0%	massimo	366024
99.5%		352506,96
97.5%		347830,4
90.0%		333956,8
75.0%	quartile	329184
50.0%	mediana	317408
25.0%	quartile	315704
10.0%		314600
2.5%		313236
0.5%		294966,8
0.0%	minimo	268820

Statistiche di riepilogo

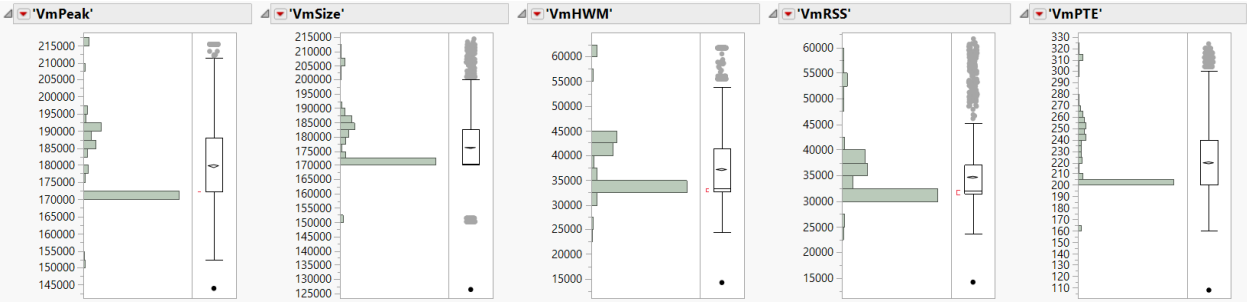
Media	321952,28
Dev std	9769,4942
Errore std della media	138,23064
Media superiore al 95%	322223,27
Media inferiore al 95%	321681,29
N	4995



Sono state notate alcune colonne ('AnonPages', 'Errors', 'AvgLatency') con valori prevalentemente costanti, per cui sono state valutate come statisticamente irrilevanti e sono state eliminate dal dataset per semplicità.

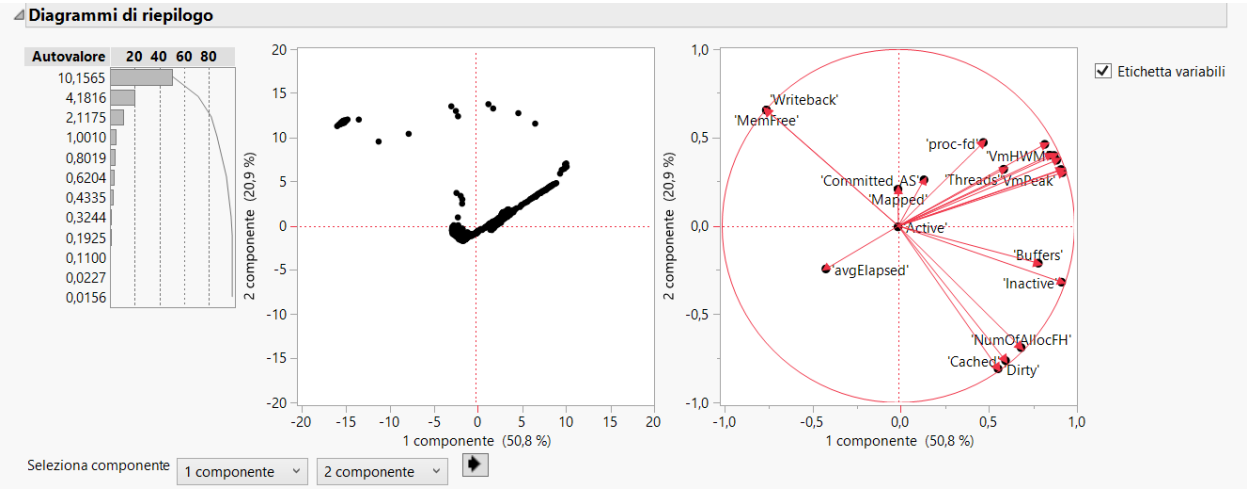
Si noti inoltre che la colonna 'Slab' presenta solo due valori diversi da 0, '1760' alla riga 90 e '4' alla riga 4963: a seguito di un'analisi dei valori corrispondenti nelle altre colonne, si è notato che questi non sono valori rilevanti all'interno dell'intero record, per cui si è deciso di eliminare queste due righe considerandole come **errori di misura**.

Si noti infine la presenza di considerevoli **outlier** in corrispondenza della prima riga: si è ipotizzato quindi che questa corrisponda al momento di startup del server, per cui è stato deciso di eliminarla.



Caratterizzazione PCA+Clustering

A seguito dell'analisi e pulizia del dataset si può procedere con PCA e clustering.



Sempre tramite JMP, è stata effettuata l'analisi degli **autovalori della matrice di covarianza**, e tramite il **test di Bartlett** sono state estratte le **componenti principali** con le relative **percentuali cumulative**.

Autovalori

Numero	Autovalore	Percentuale	20 40 60 80	Percentuale cumulativa	Chi-quadrato	DF	Prob>ChiQu
1	10,1565	50,783		50,783	331439	186,275	<,0001*
2	4,1816	20,908		71,690	280655	185,441	<,0001*
3	2,1175	10,588		82,278	246253	173,602	<,0001*
4	1,0010	5,005		87,283	220878	158,303	<,0001*
5	0,8019	4,009		91,292	207037	141,781	<,0001*
6	0,6204	3,102		94,394	191588	125,639	<,0001*
7	0,4335	2,168		96,562	174003	110,800	<,0001*
8	0,3244	1,622		98,184	155531	96,879	<,0001*
9	0,1925	0,963		99,147	131179	83,862	<,0001*
10	0,1100	0,550		99,697	103774	71,340	<,0001*
11	0,0227	0,114		99,810	66755,5	60,220	<,0001*
12	0,0156	0,078		99,888	57028,5	49,994	<,0001*
13	0,0104	0,052		99,940	47187,6	40,550	<,0001*
14	0,0082	0,041		99,981	36706,7	32,171	<,0001*
15	0,0019	0,009		99,990	15231,5	24,388	<,0001*
16	0,0012	0,006		99,996	8535,74	18,017	<,0001*
17	0,0008	0,004		100,000	583,617	12,461	<,0001*
18	0,0000	0,000		100,000	.	7,725	.
19	0,0000	0,000		100,000	.	3,829	.

Principal Components	Variance
3 PCA	0.8228
4 PCA	0.8728
5 PCA	0.9129
6 PCA	0.9439

È stato quindi possibile fare una selezione delle componenti principali in base ai valori di percentuali di varianza spiegata più significativi.

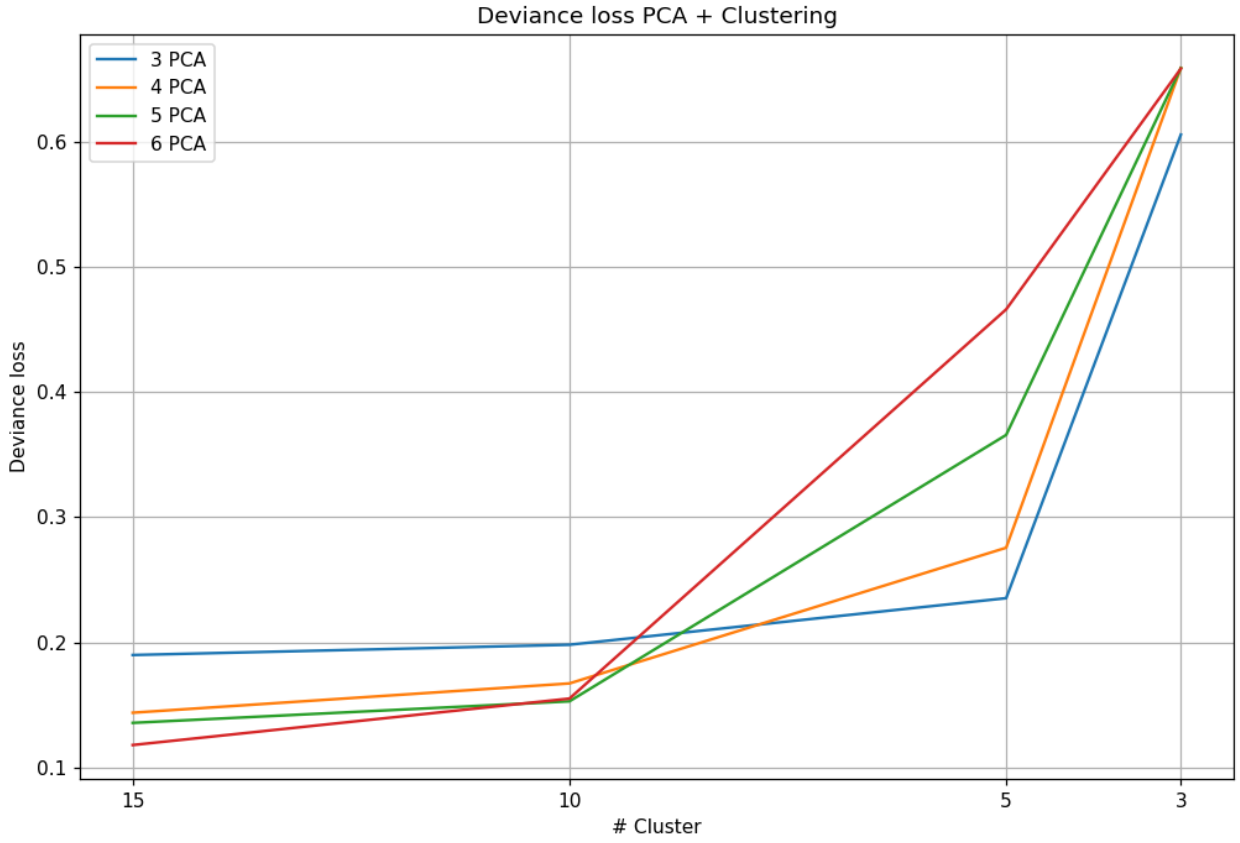
Poiché si voleva decidere quale coppia (numero di PCA e numero di cluster) fosse la scelta ottimale, sono stati considerati più valori di PCA su cui poi sono stati definiti vari valori di cluster. I valori di PCA scelti sono riportati di lato.

Per quanto riguarda la scelta del numero di cluster, si è optato per un approccio metodico su ogni numero di componenti scelte. In particolare, si è optato per analizzare la varianza persa su **15, 10, 5 e 3 cluster** per ogni numero di componenti principali. Di seguito sono riportati i risultati ottenuti:

	Intra-Cluster Deviance	Inter-Cluster Deviance	Total Deviance Loss	Area Cost
PCA: 3, Cluster: 15	0.0155	0.9845	0.1899	45
PCA: 3, Cluster: 10	0.0253	0.9747	0.1981	30
PCA: 3, Cluster: 5	0.0706	0.9294	0.2353	15
PCA: 3, Cluster: 3	0.5206	0.4793	0.6056	9
PCA: 4, Cluster: 15	0.0190	0.9810	0.1438	60

	Intra-Cluster Deviance	Inter-Cluster Deviance	Total Deviance Loss	Area Cost
PCA: 4, Cluster: 10	0.0459	0.9541	0.1672	40
PCA: 4, Cluster: 5	0.1700	0.8300	0.2756	20
PCA: 4, Cluster: 3	0.6091	0.3910	0.6588	12
PCA: 5, Cluster: 15	0.0532	0.9468	0.1357	75
PCA: 5, Cluster: 10	0.0720	0.9280	0.1529	50
PCA: 5, Cluster: 5	0.3051	0.6949	0.3657	25
PCA: 5, Cluster: 3	0.6261	0.3739	0.6587	15
PCA: 6, Cluster: 15	0.0656	0.9344	0.1180	90
PCA: 6, Cluster: 10	0.1049	0.8951	0.1551	60
PCA: 6, Cluster: 5	0.4342	0.5658	0.4660	30
PCA: 6, Cluster: 3	0.6379	0.3621	0.6583	18

È inoltre stato riportato un grafico che mostrasse l'andamento della perdita di devianza per ogni numero di componenti principali (su curve diverse).

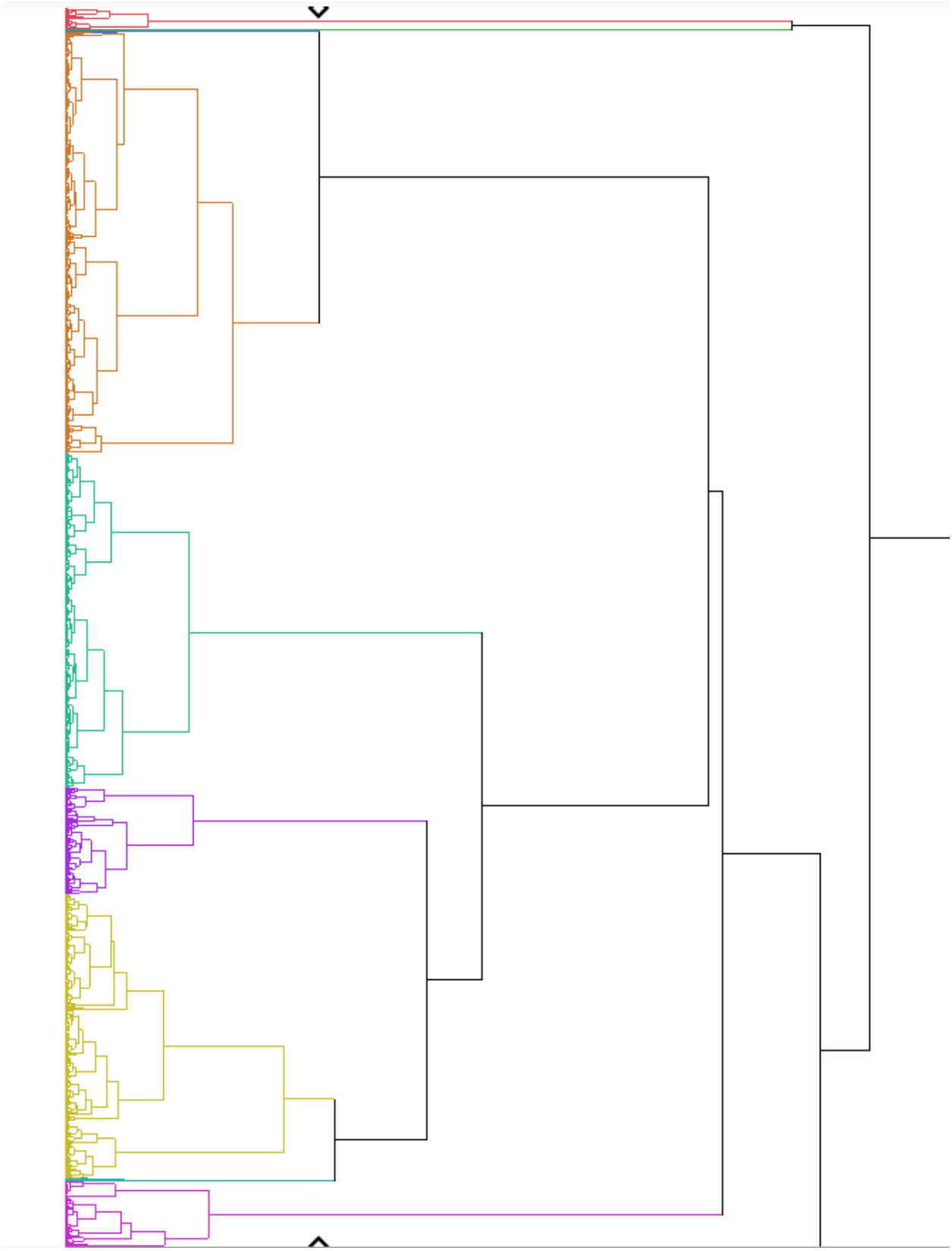


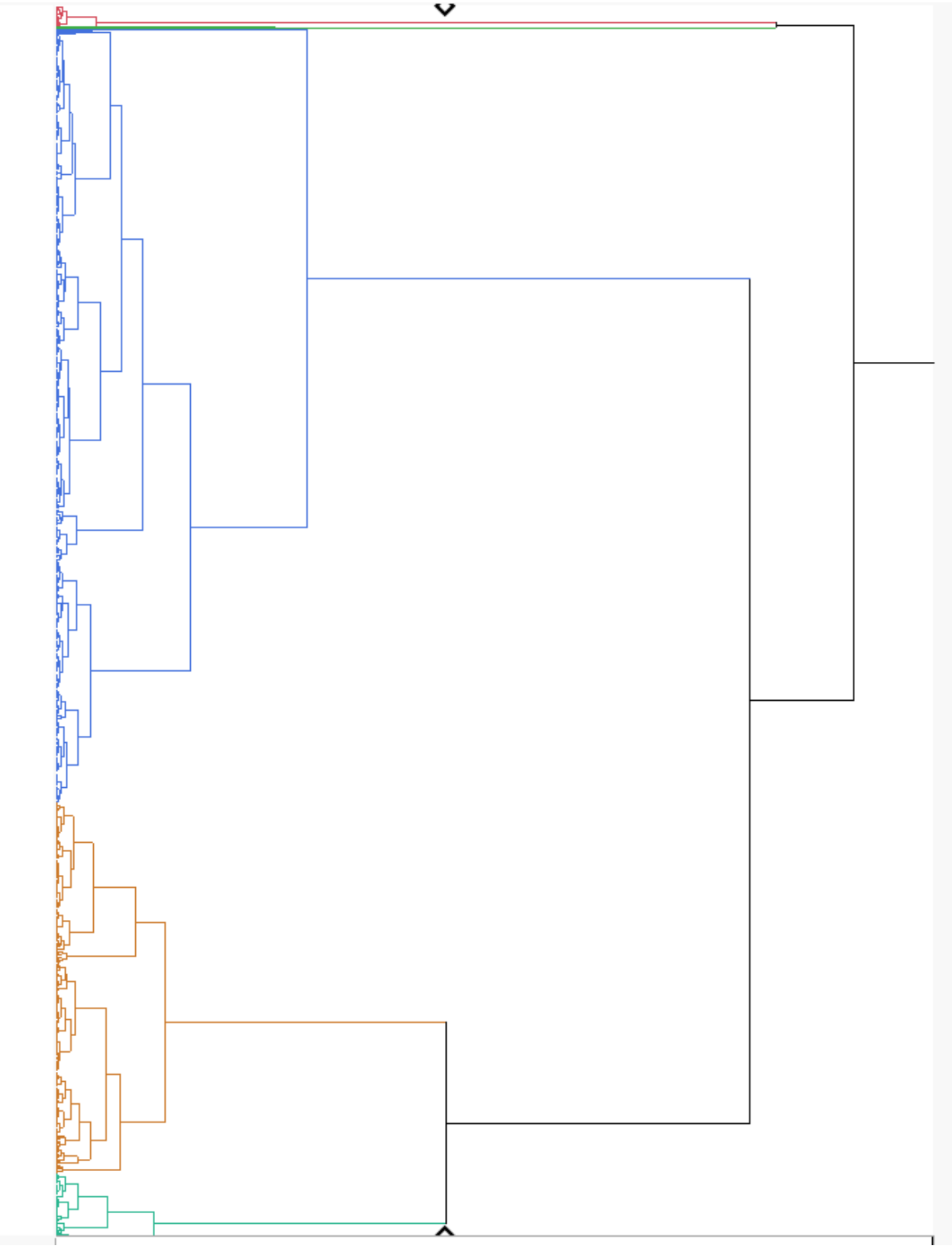
Si può notare che, al fine di caratterizzare un workload sintetico senza perdere troppa devianza, un buon numero di cluster che funziona su tutti i numeri di PC è **10 cluster**; in particolare, si potrebbe scegliere in questo caso di prendere 5 o 4 componenti principali. Tuttavia, se ci fosse necessità di avere meno cluster

(come ad esempio per **limiti di costo**), si potrebbe optare per 3 componenti su 5 cluster, perdendo però un considerevole 10% di varianza in più rispetto al caso a 10 cluster.

Potrebbe essere interessante analizzare l'andamento del clustering anche nei range intermedi a quelli considerati, per fare fine-tuning sulla scelta dei parametri di sintesi e ottimizzare la riduzione di dimensionalità.

Si mostra di seguito un esempio di **clustering post-PCA**, sia con 10 cluster e 5 componenti, sia con 5 cluster e 3 componenti, rappresentati tramite **dendrogrammi:**





Una volta ottenuti i cluster, può essere estratto un **campione significativo per ogni cluster**, in modo da poter effettuare eventuali analisi solo su una versione ridotta dell'intero dataset. L'estrazione del campione

può essere eseguita secondo varie politiche:

- Si potrebbe scegliere un **elemento a caso** all'interno del cluster.
- Si potrebbe scegliere il **centroide del cluster**.
- Si potrebbe scegliere l'**elemento più vicino al centroide** del cluster.

Non è possibile dire a priori quale scelta sia migliore rispetto alle altre, questo varia molto in base allo scopo delle successive analisi. In genere, scegliere un elemento a caso è adatto quando non importa **quanto** un certo campione appartenga a un cluster, mentre se si vuole analizzare il comportamento degli elementi più caratteristici di un cluster conviene sceglierli vicino al centroide.

Caratterizzazione Clustering-only

A fine di comparazione, è stata ripetuta la caratterizzazione senza però considerare l'intervento della PCA. Questo perché ciò che si vuole determinare è se ci sia un effettivo beneficio nell'evitare la riduzione di dimensionalità dei dati a monte di un'operazione di clustering.

Anche stavolta, sono state effettuate misure a 15, 10, 5 e 3 cluster, riportando poi le varianze intra-cluster e inter-cluster, insieme alla devianza totale persa.

	Intra-Cluster Deviance	Inter-Cluster Deviance	Total Deviance Loss
3 Cluster	0.0582	0.9418	0.0582
5 Cluster	0.0454	0.9546	0.0454
10 Cluster	0.0136	0.9864	0.0136
15 Cluster	0.0129	0.9871	0.0129

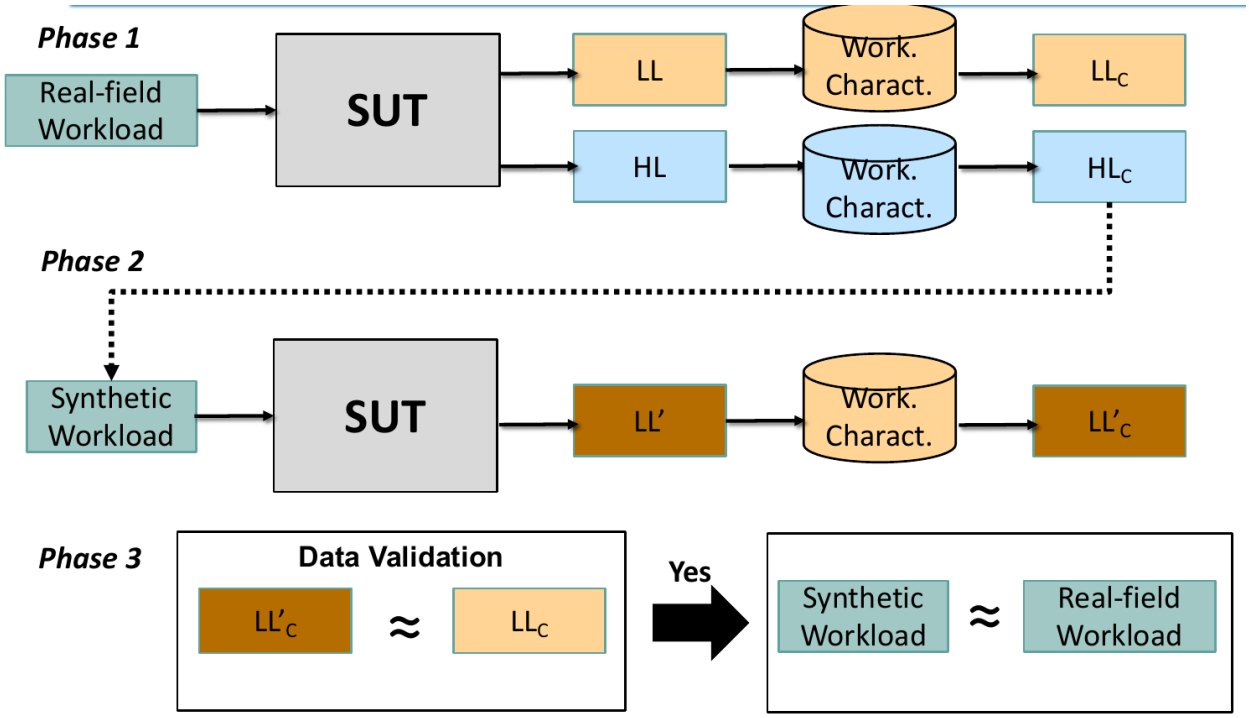
Com'era prevedibile, la devianza totale persa è considerevolmente più bassa del caso con PCA. Tuttavia, in questo caso **restano tutte le colonne del dataset**, con le eventuali correlazioni incrociate che esse portano con sé e con una quantità di dati dunque molto superiore.

Questo potrebbe non essere un problema, sia nei casi in cui le colonne sono effettivamente indipendenti, sia nei casi in cui importa solo ottenere degli esemplari significativi per ogni "gruppo" di dati separabile, e in cui magari è di interesse mantenere le informazioni ad alto livello delle feature.

Workload Characterization

Scopo del test

Lo scopo di questo esercizio è quello di ottenere una buona **caratterizzazione sintetica** di un certo **workload reale**, effettuando una valutazione statistica di parametri a **basso livello** e parametri ad **alto livello**. Di seguito è illustrata una procedura schematizzata di ciò che si andrà a fare:



In primo luogo verrà quindi simulato un **workload reale** sul Web Server che è stato analizzato precedentemente. Da questo verranno estratti due report:

- il report delle **misurazioni ad alto livello HL**: questo report racchiude informazioni sui tempi di risposta, sui byte trasferiti durante una richiesta, il tempo di avvio delle richieste, ecc.
- il report delle **risorse a basso livello LL:** questo report racchiude informazioni sulla memoria utilizzata, la percentuale di CPU utilizzata, i blocchi trasferiti da e su disco, ecc.

In seguito, verranno adoperate le tecniche di caratterizzazione viste precedentemente (PCA e Clustering) per ottenere un **workload sintetico**, sia per i dati a basso livello che per quelli ad alto livello. Con i dati ad alto livello verrà generato un workload di test sintetico per sollecitare nuovamente il sistema, ottenendo quindi nuovamente dei report a basso livello. Si dovrà poi effettuare su questi ultimi lo **stesso processo di caratterizzazione** che è stato usato per quelli reali, confrontandoli poi con opportuni **test di ipotesi** per verificare l'effettiva **equivalenza statistica** tra il workload reale e quello sintetico.

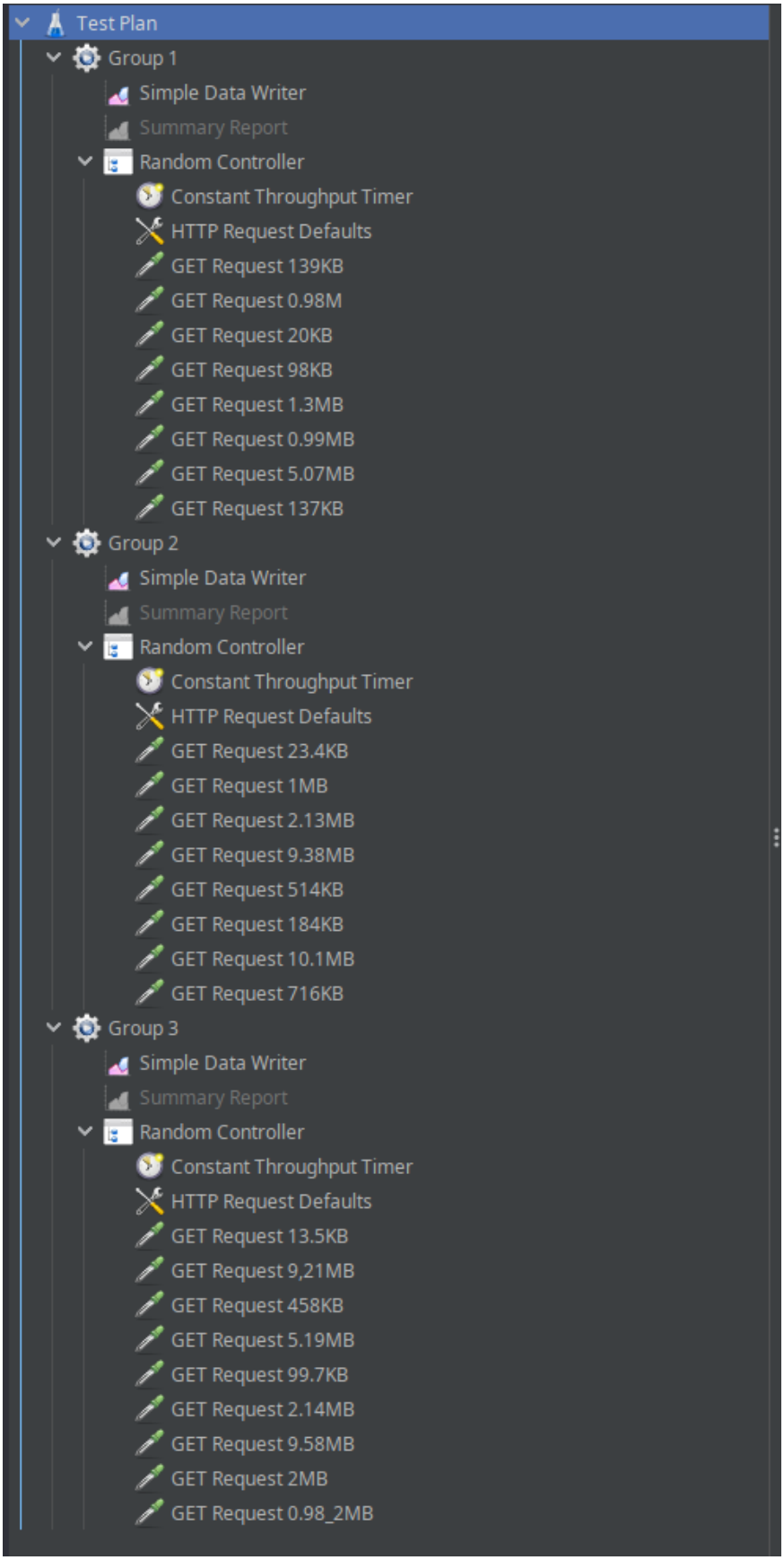
Setup del test

Al fine di realizzare la procedura sopra descritta, è stato realizzato un apposito modello di test tramite Apache Jmeter.

In particolare, dal set di risorse disponibili sul Web Server, si è ottenuta una ripartizione aleatoria di richieste (una per ogni risorsa) in tre **thread group** diversi. In particolare, la suddivisione è stata fatta secondo un criterio di ordini di grandezza delle risorse, il cui elenco completo è riportato di seguito:

- **Risorse Low**
  - Sample_13.5KB.xls

  - Sample_20KB.xls
  - Sample_23.4KB.xml
  - Sample_98KB.doc
  - Sample_99.7KB.jpg
- **Risorse Mid-Low**
  - Sample_137KB.xls
  - Sample_139KB.pdf
  - Sample_184KB.xlsx
  - Sample_458KB.pdf
  - Sample_514KB.ppt
- **Risorse Mid**
  - Sample_716KB.mp3
  - Sample_0.98MB.ppt
  - Sample_0.98_2MB.ogg
  - Sample_0.99MB.pdf
  - Sample_1MB.mp3
- **Risorse Mid-High**
  - Sample_1.3MB.mov
  - Sample_2MB.ogg
  - Sample_2.13MB.jpg
  - Sample_2.14MB.mov
  - Sample_5.07MB.mp3


Test Plan
Group 1
Simple Data Writer
Summary Report
Random Controller
Constant Throughput Timer
HTTP Request Defaults
GET Request 139KB
GET Request 0.98M
GET Request 20KB
GET Request 98KB
GET Request 1.3MB
GET Request 0.99MB
GET Request 5.07MB
GET Request 137KB
Group 2
Simple Data Writer
Summary Report
Random Controller
Constant Throughput Timer
HTTP Request Defaults
GET Request 23.4KB
GET Request 1MB
GET Request 2.13MB
GET Request 9.38MB
GET Request 514KB
GET Request 184KB
GET Request 10.1MB
GET Request 716KB
Group 3
Simple Data Writer
Summary Report
Random Controller
Constant Throughput Timer
HTTP Request Defaults
GET Request 13.5KB
GET Request 9,21MB
GET Request 458KB
GET Request 5.19MB
GET Request 99.7KB
GET Request 2.14MB
GET Request 9.58MB
GET Request 2MB
GET Request 0.98_2MB

- **Risorse High**
  - Sample_5.19MB.zip
  - Sample_9.21MB.zip
  - Sample_9.38MB.mp4
  - Sample_9.58MB.jpg
  - Sample_10.1MB.zip

Le risorse ripartite nei thread group sono visibili nell'immagine sopra riportata. I thread group sono stati impostati per eseguire **sequenzialmente**, e **non in parallelo**, per evitare strane dipendenze tra le richieste tra gruppi diversi e evitare di saturare il bottleneck del sistema (la CPU).

Per ogni thread group sono stati impostati i seguenti parametri di esecuzione:

- **Numero di thread (user):** 15
- **Ramp-up period:** 15 secondi
- **Duration:** 300 secondi

Il **Constant Throughput Timer** per ognuno dei gruppi è stato impostato a 1300, che è un valore intermedio tra 1100 (Knee Capacity) e 1500 (Usable Capacity).

È stato quindi fatto partire un test completo, per una durata totale di **15 minuti** di misurazioni. Da queste si sono quindi ricavati i **report ad alto livello** tramite il **Simple Data Writer** di JMeter, e i **report a basso livello** tramite **vmstat** eseguito sulla macchina virtuale che ospita il Web Server.

Caratterizzazione Workload sintetico

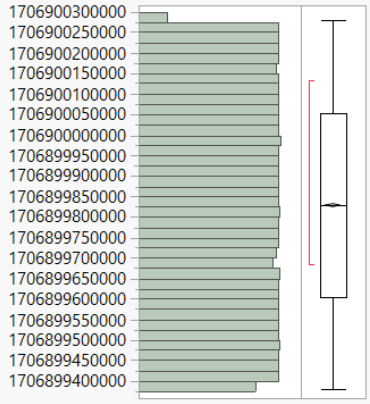
Caratterizzazione HL

Sia per la caratterizzazione ad alto livello che per quella a basso livello è stato usato il software JMP. Nel caso dell'analisi ad alto livello HL, si riporta innanzitutto una visione dei dati grezzi:

	timeStamp	elapsed	label	responseCode	responseMessage	threadName	dataType	success	failureMessage	bytes	sentBytes	grpThreads	allThreads	URL	Latency	IdleTime	Connect
1	1,70663e+12	19	GET Request 30k	200	OK	Group 1 1-1	text	true		34202	0	1	3	http:/...	16	0	0
2	1,70663e+12	6	GET Request 9.4k	200	OK	Group 1 1-1	text	true		9987	0	1	3	http:/...	5	0	0
3	1,70663e+12	1	GET Request 100k	200	OK	Group 1 1-1	bin	true		12070	0	1	3	http:/...	1	0	0
4	1,70663e+12	4	GET Request 30k	200	OK	Group 1 1-1	text	true		34201	0	1	3	http:/...	3	0	0
5	1,70663e+12	6	GET Request 500k	200	OK	Group 1 1-1	text	true		512810	0	1	3	http:/...	1	0	0
6	1,70663e+12	1	GET Request 30k	200	OK	Group 1 1-1	text	true		34201	0	1	3	http:/...	1	0	0
7	1,70663e+12	5	GET Request 30k	200	OK	Group 1 1-1	text	true		34201	0	1	3	http:/...	5	0	0
8	1,70663e+12	3	GET Request 30k	200	OK	Group 1 1-1	text	true		34201	0	1	3	http:/...	3	0	0
9	1,70663e+12	1	GET Request 30k	200	OK	Group 1 1-1	text	true		34201	0	1	3	http:/...	1	0	0
10	1,70663e+12	1	GET Request 100k	200	OK	Group 1 1-1	bin	true		12070	0	1	3	http:/...	1	0	0
11	1,70663e+12	1	GET Request 9.4k	200	OK	Group 1 1-1	text	true		9987	0	1	3	http:/...	1	0	0
12	1,70663e+12	7	GET Request 500k	200	OK	Group 1 1-1	text	true		512810	0	1	3	http:/...	1	0	0
13	1,70663e+12	1	GET Request 30k	200	OK	Group 1 1-1	text	true		34201	0	1	3	http:/...	1	0	0
14	1,70663e+12	1	GET Request 9.4k	200	OK	Group 1 1-1	text	true		9987	0	1	3	http:/...	1	0	0
15	1,70663e+12	1	GET Request 9.4k	200	OK	Group 1 1-1	text	true		9987	0	1	3	http:/...	1	0	0
16	1,70663e+12	3	GET Request 100k	200	OK	Group 1 1-1	bin	true		12070	0	1	3	http:/...	3	0	0
17	1,70663e+12	1	GET Request 100k	200	OK	Group 1 1-1	bin	true		12070	0	1	3	http:/...	1	0	0
18	1,70663e+12	6	GET Request 500k	200	OK	Group 1 1-1	text	true		512810	0	1	3	http:/...	2	0	0
19	1,70663e+12	1	GET Request 100k	200	OK	Group 1 1-1	bin	true		12070	0	1	3	http:/...	1	0	0
20	1,70663e+12	4	GET Request 100k	200	OK	Group 1 1-1	bin	true		12070	0	1	3	http:/...	4	0	0
21	1,70663e+12	22	GET Request 500k	200	OK	Group 1 1-1	text	true		512810	0	1	3	http:/...	9	0	0
22	1,70663e+12	1	GET Request 100k	200	OK	Group 1 1-1	bin	true		12070	0	1	3	http:/...	1	0	0
23	1,70663e+12	1	GET Request 100k	200	OK	Group 1 1-1	bin	true		12070	0	1	3	http:/...	1	0	0
24	1,70663e+12	6	GET Request 500k	200	OK	Group 1 1-1	text	true		512810	0	1	3	http:/...	1	0	0
25	1,70663e+12	7	GET Request 500k	200	OK	Group 1 1-1	text	true		512810	0	1	3	http:/...	1	0	0
26	1,70663e+12	0	GET Request 9.4k	200	OK	Group 1 1-1	text	true		9987	0	1	3	http:/...	0	0	0
27	1,70663e+12	8	GET Request 500k	200	OK	Group 1 1-1	text	true		512810	0	1	3	http:/...	1	0	0
28	1,70663e+12	9	GET Request 500k	200	OK	Group 1 1-1	text	true		512810	0	1	3	http:/...	1	0	0
29	1,70663e+12	0	GET Request 100k	200	OK	Group 1 1-1	bin	true		12070	0	1	3	http:/...	0	0	0
30	1,70663e+12	1	GET Request 100k	200	OK	Group 1 1-1	bin	true		12070	0	1	3	http:/...	1	0	0
31	1,70663e+12	1	GET Request 9.4k	200	OK	Group 1 1-1	text	true		9987	0	1	3	http:/...	1	0	0
32	1,70663e+12	1	GET Request 30k	200	OK	Group 1 1-1	text	true		34201	0	1	3	http:/...	1	0	0

Per cui è stata poi effettuata una preventiva analisi delle **distribuzioni campionarie**:

timeStamp



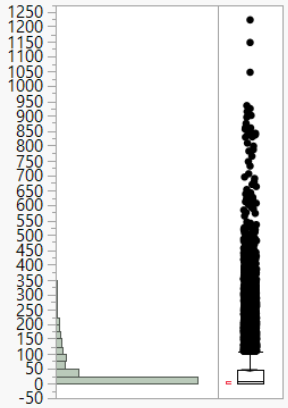
Quantili

100.0%	massimo	1706900279986
99.5%		1706900275522,6
97.5%		1706900257527,2
90.0%		1706900190125
75.0%	quartile	1706900054966
50.0%	mediana	1706899830419
25.0%	quartile	1706899604079
10.0%		1706899469095
2.5%		1706899401652
0.5%		1706899384030
0.0%	minimo	1706899379750

Statistiche di riepilogo

Media	1,707e+12
Dev std	260144,01
Errore std della media	1862,6912
Media superiore al 95%	1,707e+12
Media inferiore al 95%	1,707e+12
N	19505

elapsed



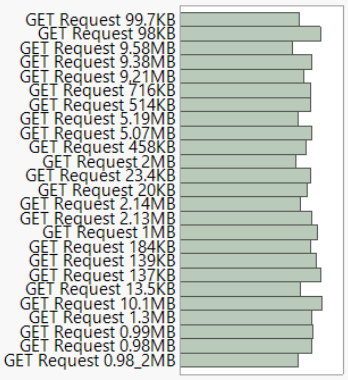
Quantili

100.0%	massimo	1223
99.5%		461
97.5%		298
90.0%		138
75.0%	quartile	45
50.0%	mediana	10
25.0%	quartile	2
10.0%		1
2.5%		1
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

Media	45,657216
Dev std	84,841491
Errore std della media	0,6074847
Media superiore al 95%	46,847938
Media inferiore al 95%	44,466494
N	19505

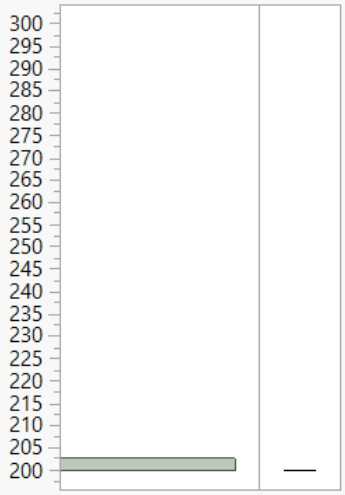
label



Frequenze

Livello	Conteggio	Prob
GET Request 0.98_2MB	714	0,03661
GET Request 0.98MB	802	0,04112
GET Request 0.99MB	813	0,04168
GET Request 1.3MB	805	0,04127
GET Request 10.1MB	861	0,04414
GET Request 13.5KB	734	0,03763
GET Request 137KB	849	0,04353
GET Request 139KB	823	0,04219
GET Request 184KB	794	0,04071
GET Request 1MB	839	0,04301
GET Request 2.13MB	797	0,04086
GET Request 2.14MB	735	0,03768
GET Request 20KB	776	0,03978
GET Request 23.4KB	795	0,04076
GET Request 2MB	702	0,03599
GET Request 458KB	762	0,03907
GET Request 5.07MB	798	0,04091
GET Request 5.19MB	712	0,03650
GET Request 514KB	793	0,04066
GET Request 716KB	794	0,04071

responseCode



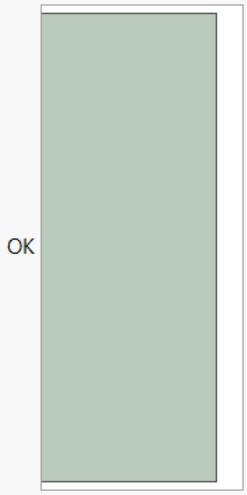
Quantili

100.0%	massimo	200
99.5%		200
97.5%		200
90.0%		200
75.0%	quartile	200
50.0%	mediana	200
25.0%	quartile	200
10.0%		200
2.5%		200
0.5%		200
0.0%	minimo	200

Statistiche di riepilogo

Media	200
Dev std	0
Errore std della media	0
Media superiore al 95%	200
Media inferiore al 95%	200
N	19505

responseMessage



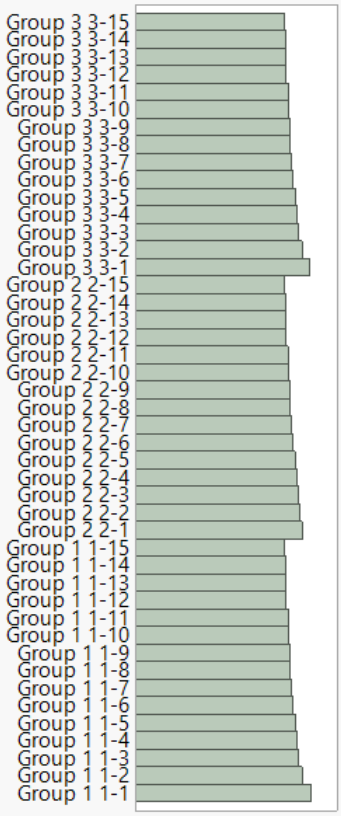
Frequenze

Livello	Conteggio	Prob
OK	19505	1,00000
Totale	19505	1,00000

N mancanti 0

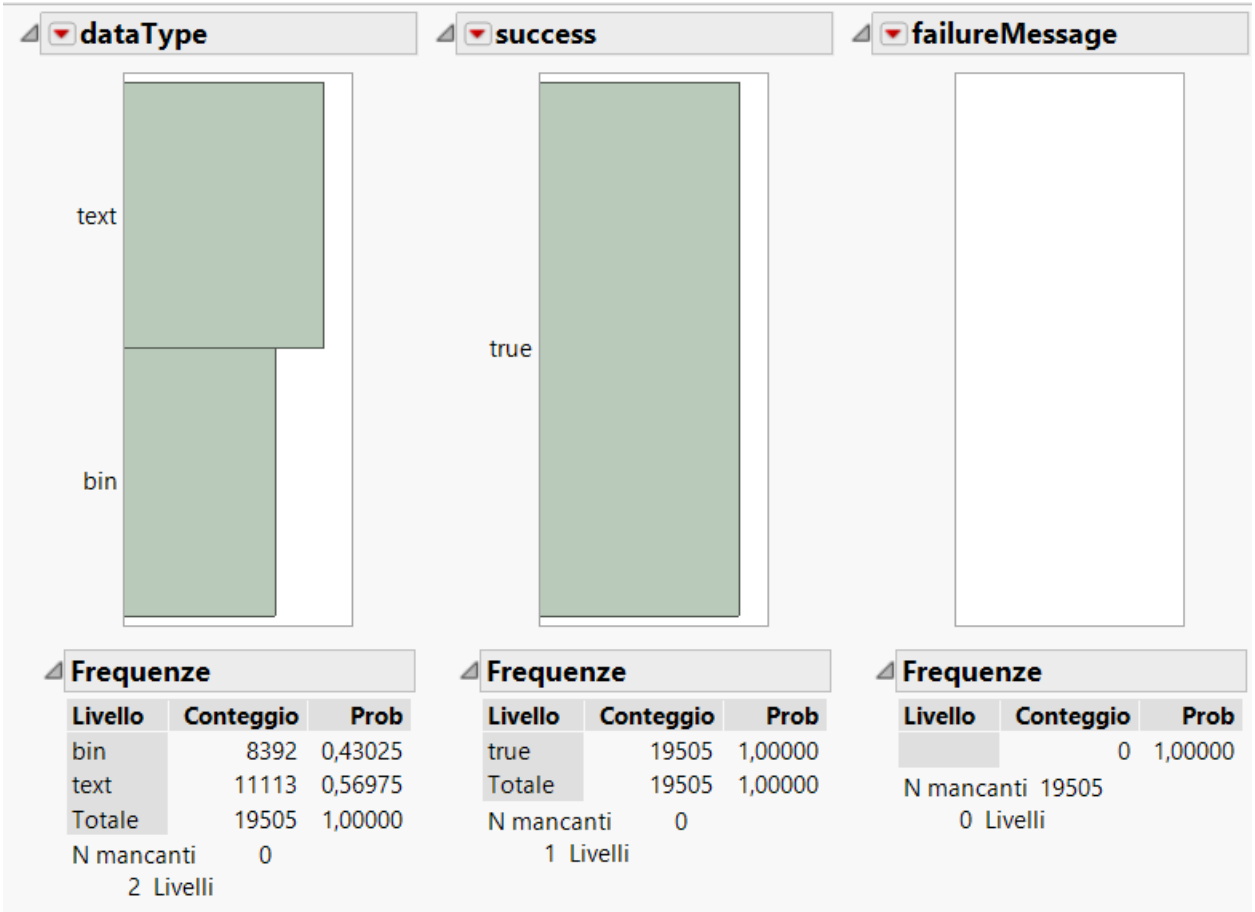
1 Livelli

threadName



Frequenze

Livello	Conteggio	Prob
Group 1 1-1	485	0,02487
Group 1 1-2	463	0,02374
Group 1 1-3	452	0,02317
Group 1 1-4	446	0,02287
Group 1 1-5	440	0,02256
Group 1 1-6	435	0,02230
Group 1 1-7	432	0,02215
Group 1 1-8	429	0,02199



Frequenze

Livello	Conteggio	Prob
bin	8392	0,43025
text	11113	0,56975
Totale	19505	1,00000

N mancanti 0

2 Livelli

Frequenze

Livello	Conteggio	Prob
true	19505	1,00000
Totale	19505	1,00000

N mancanti 0

1 Livelli

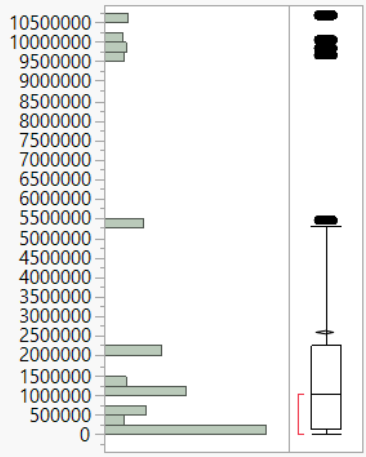
Frequenze

Livello	Conteggio	Prob
	0	1,00000

N mancanti 19505

0 Livelli

bytes



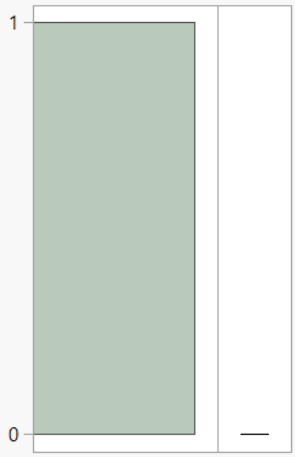
Quantili

100.0%	massimo	10679929
99.5%		10679928
97.5%		10679928
90.0%		9840788
75.0%	quartile	2247497
50.0%	mediana	1033238
25.0%	quartile	143081
10.0%		24288
2.5%		14126
0.5%		14126
0.0%	minimo	14125

Statistiche di riepilogo

Media	2597737,4
Dev std	3537121,3
Errore std della media	25326,605
Media superiore al 95%	2647379,8
Media inferiore al 95%	2548095,1
N	19505

sentBytes



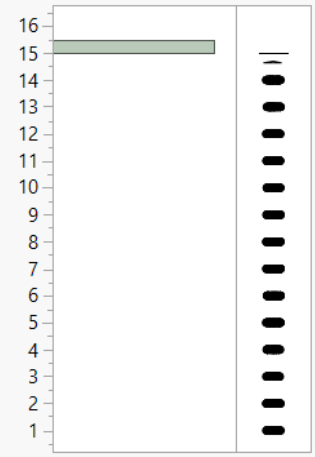
Quantili

100.0%	massimo	0
99.5%		0
97.5%		0
90.0%		0
75.0%	quartile	0
50.0%	mediana	0
25.0%	quartile	0
10.0%		0
2.5%		0
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

Media	0
Dev std	0
Errore std della media	0
Media superiore al 95%	0
Media inferiore al 95%	0
N	19505

grpThreads



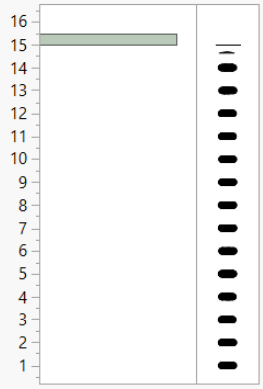
Quantili

100.0%	massimo	15
99.5%		15
97.5%		15
90.0%		15
75.0%	quartile	15
50.0%	mediana	15
25.0%	quartile	15
10.0%		15
2.5%		8
0.5%		2
0.0%	minimo	1

Statistiche di riepilogo

Media	14,646911
Dev std	1,7838959
Errore std della media	0,0127731
Media superiore al 95%	14,671947
Media inferiore al 95%	14,621875
N	19505

allThreads



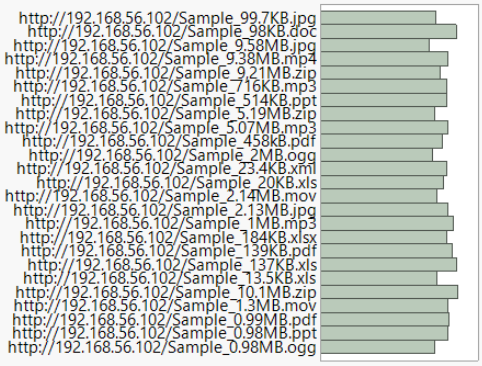
Quantili

100.0%	massimo	15
99.5%		15
97.5%		15
90.0%		15
75.0%	quartile	15
50.0%	mediana	15
25.0%	quartile	15
10.0%		15
2.5%		8
0.5%		2
0.0%	minimo	1

Statistiche di riepilogo

Media	14,646911
Dev std	1,7838959
Errore std della media	0,0127731
Media superiore al 95%	14,671947
Media inferiore al 95%	14,621875
N	19505

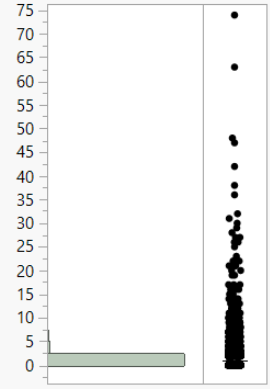
URL



Frequenze

Livello	Conteggio	Prob
http://192.168.56.102/Sample_0.98MB.ogg	714	0,03661
http://192.168.56.102/Sample_0.98MB.ppt	802	0,04112
http://192.168.56.102/Sample_0.99MB.pdf	813	0,04168
http://192.168.56.102/Sample_1.3MB.mov	805	0,04127
http://192.168.56.102/Sample_10.1MB.zip	861	0,04414
http://192.168.56.102/Sample_13.5KB.xls	734	0,03763
http://192.168.56.102/Sample_137KB.xls	849	0,04353
http://192.168.56.102/Sample_139KB.pdf	823	0,04219
http://192.168.56.102/Sample_184KB.xlsx	794	0,04071
http://192.168.56.102/Sample_1MB.mp3	839	0,04301
http://192.168.56.102/Sample_2.13MB.jpg	797	0,04086
http://192.168.56.102/Sample_2.14MB.mov	735	0,03768
http://192.168.56.102/Sample_20KB.xls	776	0,03978
http://192.168.56.102/Sample_23.4KB.xml	795	0,04076
http://192.168.56.102/Sample_2MB.ogg	702	0,03599
http://192.168.56.102/Sample_458kB.pdf	762	0,03907
http://192.168.56.102/Sample_5.07MB.mp3	798	0,04091
http://192.168.56.102/Sample_5.19MB.zip	712	0,03650
http://192.168.56.102/Sample_514KB.ppt	793	0,04066
http://192.168.56.102/Sample_716KB.mp3	794	0,04071

Latency

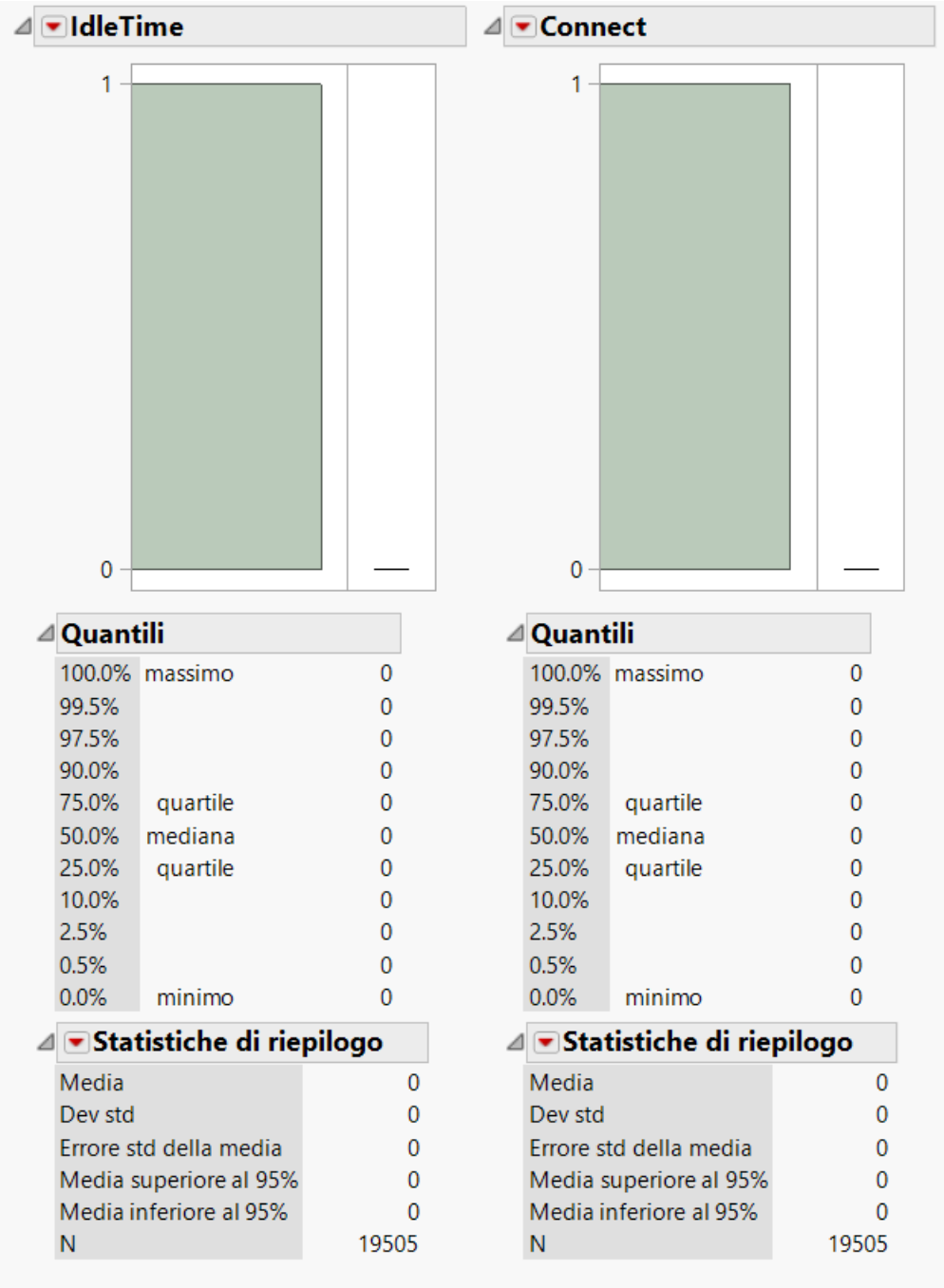


Quantili

100.0%	massimo	74
99.5%		9
97.5%		3
90.0%		2
75.0%	quartile	1
50.0%	mediana	1
25.0%	quartile	1
10.0%		0
2.5%		0
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

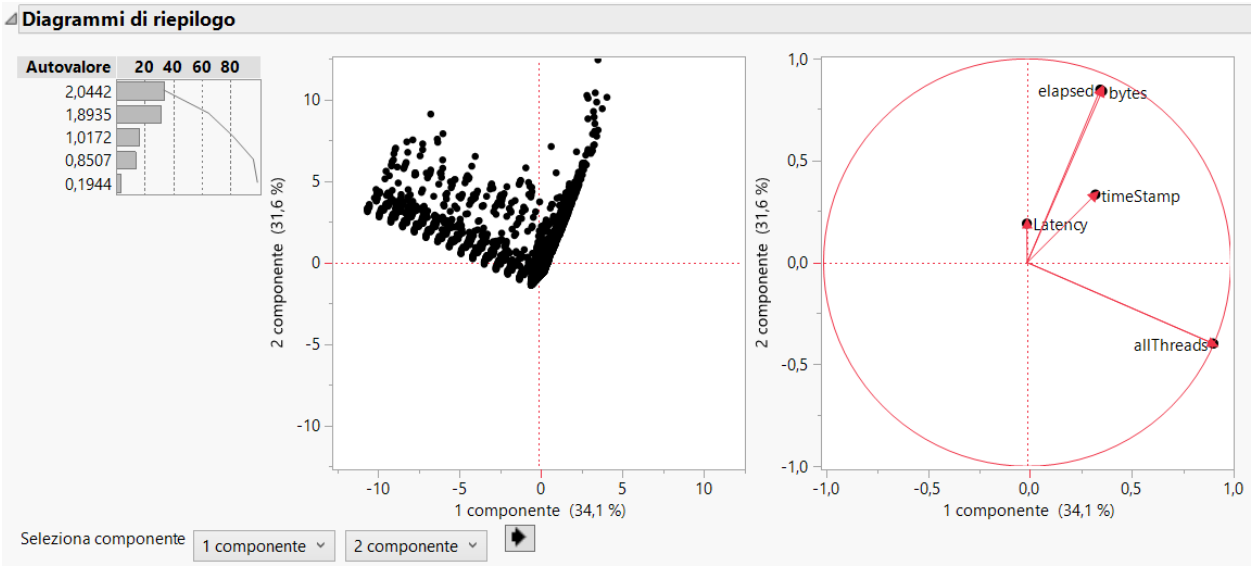
Media	1,1693412
Dev std	1,6312811
Errore std della media	0,0116803
Media superiore al 95%	1,1922357
Media inferiore al 95%	1,1464467
N	19505



Si effettua quindi una pulizia dei dati e si tolgono le colonne non significative:

- ResponseMessage, ResponseCode, Success, FailureMessage, SentBytes, IdleTime e Connect poiché mantengono sempre lo stesso valore.
- DataType perché rappresenta un'informazione superflua.
- URL perché è possibile ottenere le stesse informazioni in maniera più leggibile con Label.

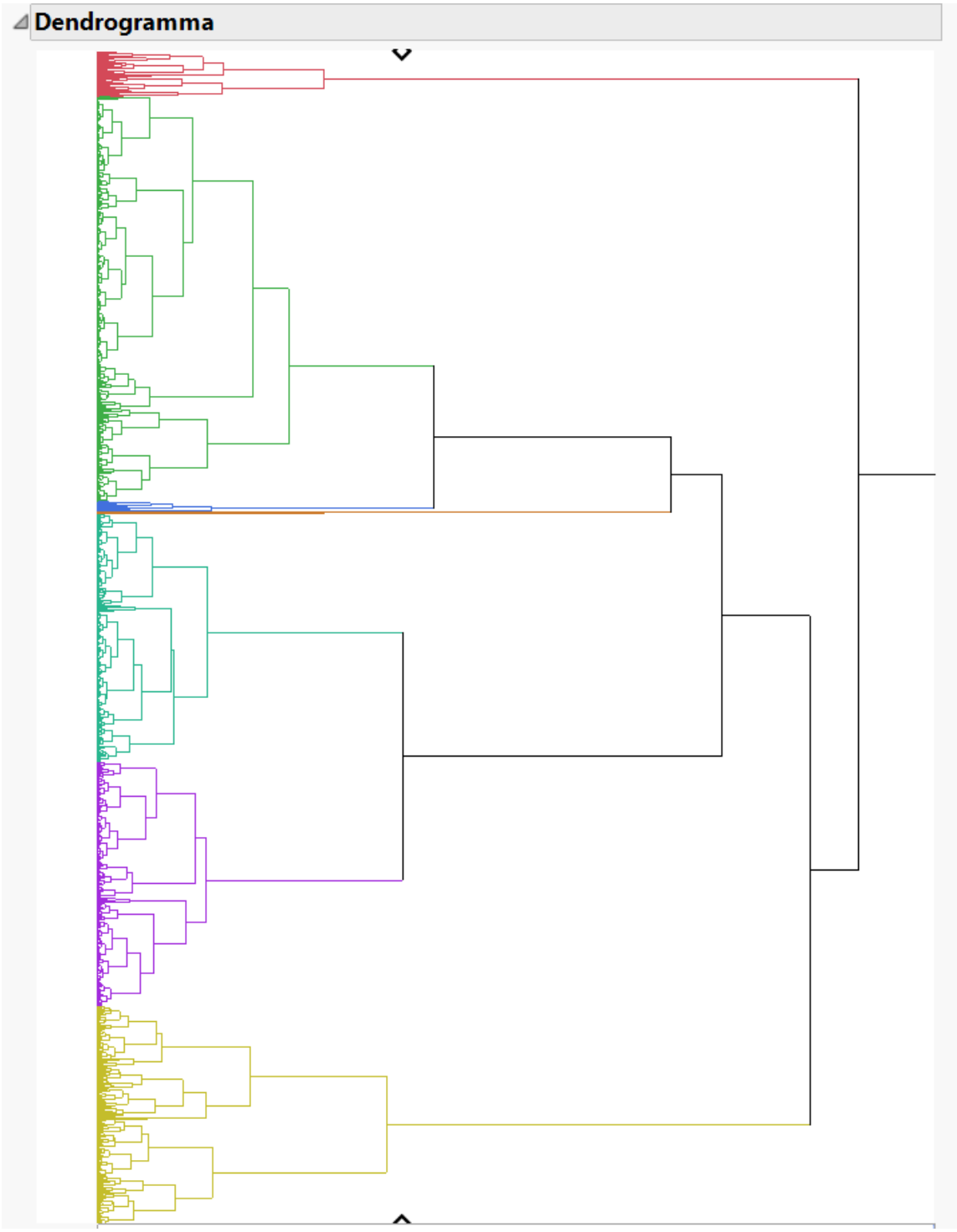
Dopo la pulizia delle colonne, è stato possibile effettuare un'**analisi PCA**, e anche stavolta sono stati analizzati **autovalori della matrice di covarianza** e **test di Bartlett** per ottenere le percentuali cumulative.



Autovalori

Numero	Autovalore	Percentuale	20 40 60 80	Percentuale cumulativa	Chi-quadrato	DF	Prob>ChiQu
1	2,0442	34,070		34,070	8368,44	14,141	<,0001*
2	1,8935	31,559		65,628	.	11,641	.
3	1,0172	16,953		82,582	.	9,231	.
4	0,8507	14,179		96,760	.	5,567	.
5	0,1944	3,240		100,000	.	2,749	.

È stato scelto di selezionare **4 componenti principali**. A valle di questa scelta, è stata avviata la procedura di clustering per scegliere il numero ottimale di cluster da prendere in considerazione per costruire il workload sintetico.



Si è optato per un numero di **8 cluster**, che ci porta in una situazione di devianza persa (considerando anche la PCA) di:

Intra-Cluster Deviance	Inter-Cluster Deviance	Total Deviance Loss (PCA + Clustering)
0.2029	0.7971	0.2287

Conservando quindi una varianza di circa il 77,13% rispetto a quella totale (prima di PCA e clustering). Si è provveduto quindi ad analizzare la **tabella di contingenza** tra cluster e label di richieste effettuate, per estrarre solo quelle più presenti all'interno dei cluster:

Cluster	Conteggio % del totale % di colonne % di righe	GET Request 0.98_2 MB	GET Request 0.98MB	GET Request 0.99MB	GET Request 1.3MB	GET Request 10.1MB	GET Request 13.5KB	GET Request 137KB	GET Request 139KB	GET Request 184KB	GET Request 1MB	GET Request 2.13MB	GET Request 2.14MB	GET Request 20KB	GET Request 23.4KB	GET Request 2MB	GET Request 458KB	GET Request 5.07MB	GET Request 5.19MB	GET Request 514KB	GET Request 716KB	GET Request 9,21MB	GET Request 9.38MB	GET Request 9.58MB	GET Request 98KB	GET Request 99.7KB	Totale
	1	32	29	26	34	32	31	40	25	27	27	26	37	33	17	28	41	36	25	27	28	28	33	26	36	18	742
		0,16	0,15	0,13	0,17	0,16	0,16	0,21	0,13	0,14	0,14	0,13	0,19	0,17	0,09	0,14	0,21	0,18	0,13	0,14	0,14	0,14	0,17	0,13	0,18	0,09	3,80
		4,48	3,62	3,20	4,22	3,72	4,22	4,71	3,04	3,40	3,22	3,26	5,03	4,25	2,14	3,99	5,38	4,51	3,51	3,40	3,53	3,75	4,10	3,80	4,23	2,50	
		4,31	3,91	3,50	4,58	4,31	4,18	5,39	3,37	3,64	3,64	3,50	4,99	4,45	2,29	3,77	5,53	4,85	3,37	3,64	3,77	3,77	4,45	3,50	4,85	2,43	
	2	1	767	778	754	0	0	801	792	76	102	85	3	736	88	5	0	753	0	98	95	0	0	0	807	0	6741
		0,01	3,93	3,99	3,87	0,00	0,00	4,11	4,06	0,39	0,52	0,44	0,02	3,77	0,45	0,03	0,00	3,86	0,00	0,50	0,49	0,00	0,00	0,00	4,14	0,00	34,56
		0,14	95,64	95,69	93,66	0,00	0,00	94,35	96,23	9,57	12,16	10,66	0,41	94,85	11,07	0,71	0,00	94,36	0,00	12,36	11,96	0,00	0,00	0,00	94,72	0,00	
		0,01	11,38	11,54	11,19	0,00	0,00	11,88	11,75	1,13	1,51	1,26	0,04	10,92	1,31	0,07	0,00	11,17	0,00	1,45	1,41	0,00	0,00	0,00	11,97	0,00	
	3	2	4	7	11	7	3	8	6	8	11	11	5	7	16	4	0	8	3	9	5	4	8	2	9	2	160
		0,01	0,02	0,04	0,06	0,04	0,02	0,04	0,03	0,04	0,06	0,06	0,03	0,04	0,08	0,02	0,00	0,04	0,02	0,05	0,03	0,02	0,04	0,01	0,05	0,01	0,82
		0,28	0,50	0,86	1,37	0,81	0,41	0,94	0,73	1,01	1,31	1,38	0,68	0,90	2,01	0,57	0,00	1,00	0,42	1,13	0,63	0,54	0,99	0,29	1,06	0,28	
		1,25	2,50	4,38	6,88	4,38	1,88	5,00	3,75	5,00	6,88	6,88	3,13	4,38	10,00	2,50	0,00	5,00	1,88	5,63	3,13	2,50	5,00	1,25	5,63	1,25	
	4	2	0	0	1	4	1	0	0	1	1	2	2	0	1	1	2	0	0	4	2	6	2	1	0	0	33
		0,01	0,00	0,00	0,01	0,02	0,01	0,00	0,00	0,01	0,01	0,01	0,01	0,00	0,01	0,01	0,01	0,00	0,00	0,02	0,01	0,03	0,01	0,01	0,00	0,00	0,17
		0,28	0,00	0,00	0,12	0,46	0,14	0,00	0,00	0,13	0,12	0,25	0,27	0,00	0,13	0,14	0,26	0,00	0,00	0,50	0,25	0,80	0,25	0,15	0,00	0,00	
		6,06	0,00	0,00	3,03	12,12	3,03	0,00	0,00	3,03	3,03	6,06	6,06	0,00	3,03	3,03	6,06	0,00	0,00	12,12	6,06	18,18	6,06	3,03	0,00	0,00	
	5	30	2	2	5	0	32	0	0	662	678	661	22	0	655	23	38	0	0	632	645	0	0	0	0	49	4136
		0,15	0,01	0,01	0,03	0,00	0,16	0,00	0,00	3,39	3,48	3,39	0,11	0,00	3,36	0,12	0,19	0,00	0,00	3,24	3,31	0,00	0,00	0,00	0,00	0,25	21,20
		4,20	0,25	0,25	0,62	0,00	4,36	0,00	0,00	83,38	80,81	82,94	2,99	0,00	82,39	3,28	4,99	0,00	0,00	79,70	81,23	0,00	0,00	0,00	0,00	6,82	
		0,73	0,05	0,05	0,12	0,00	0,77	0,00	0,00	16,01	16,39	15,98	0,53	0,00	15,84	0,56	0,92	0,00	0,00	15,28	15,59	0,00	0,00	0,00	0,00	1,18	
	6	647	0	0	0	0	667	0	0	20	20	12	656	0	18	638	681	0	0	23	19	0	0	0	0	650	4051
		3,32	0,00	0,00	0,00	0,00	3,42	0,00	0,00	0,10	0,10	0,06	3,36	0,00	0,09	3,27	3,49	0,00	0,00	0,12	0,10	0,00	0,00	0,00	0,00	3,33	20,77
		90,62	0,00	0,00	0,00	0,00	90,87	0,00	0,00	2,52	2,38	1,51	89,25	0,00	2,26	90,88	89,37	0,00	0,00	2,90	2,39	0,00	0,00	0,00	0,00	90,40	
		15,97	0,00	0,00	0,00	0,00	16,47	0,00	0,00	0,49	0,49	0,30	16,19	0,00	0,44	15,75	16,81	0,00	0,00	0,57	0,47	0,00	0,00	0,00	0,00	16,05	
	7	0	0	0	0	799	0	0	0	0	0	0	0	0	0	0	0	1	6	0	0	154	745	179	0	0	1884
		0,00	0,00	0,00	0,00	4,10	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,03	0,00	0,00	0,79	3,82	0,92	0,00	0,00	9,66
		0,00	0,00	0,00	0,00	92,80	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,13	0,84	0,00	0,00	20,62	92,55	26,17	0,00	0,00	
		0,00	0,00	0,00	0,00	42,41	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,05	0,32	0,00	0,00	8,17	39,54	9,50	0,00	0,00	
	8	0	0	0	0	19	0	0	0	0	0	0	10	0	0	3	0	0	678	0	0	555	17	476	0	0	1758
		0,00	0,00	0,00	0,00	0,10	0,00	0,00	0,00	0,00	0,00	0,00	0,05	0,00	0,00	0,02	0,00	0,00	3,48	0,00	0,00	2,85	0,09	2,44	0,00	0,00	9,01
		0,00	0,00	0,00	0,00	2,21	0,00	0,00	0,00	0,00	0,00	0,00	1,36	0,00	0,00	0,43	0,00	0,00	95,22	0,00	0,00	74,30	2,11	69,59	0,00	0,00	
		0,00	0,00	0,00	0,00	1,08	0,00	0,00	0,00	0,00	0,00	0,00	0,57	0,00	0,00	0,17	0,00	0,00	38,57	0,00	0,00	31,57	0,97	27,08	0,00	0,00	
	Totale	714	802	813	805	861	734	849	823	794	839	797	735	776	795	702	762	798	712	793	794	747	805	684	852	719	19505
		3,66	4,11	4,17	4,13	4,41	3,76	4,35	4,22	4,07	4,30	4,09	3,77	3,98	4,08	3,60	3,91	4,09	3,65	4,07	4,07	3,83	4,13	3,51	4,37	3,69	

Da questi valori è possibile quindi caratterizzare il **workload sintetico**, che viene costruito a partire dalla richiesta più presente per ogni cluster, e quindi:

- **Cluster 1:** 458KB (41)
- **Cluster 2:** 98KB (807)
- **Cluster 3:** 23.4KB (16)
- **Cluster 4:** 9.21MB (6)
- **Cluster 5:** 1MB (678)
- **Cluster 6:** 458KB (681)
- **Cluster 7:** 10.1MB (799)
- **Cluster 8:** 5.19MB (678)

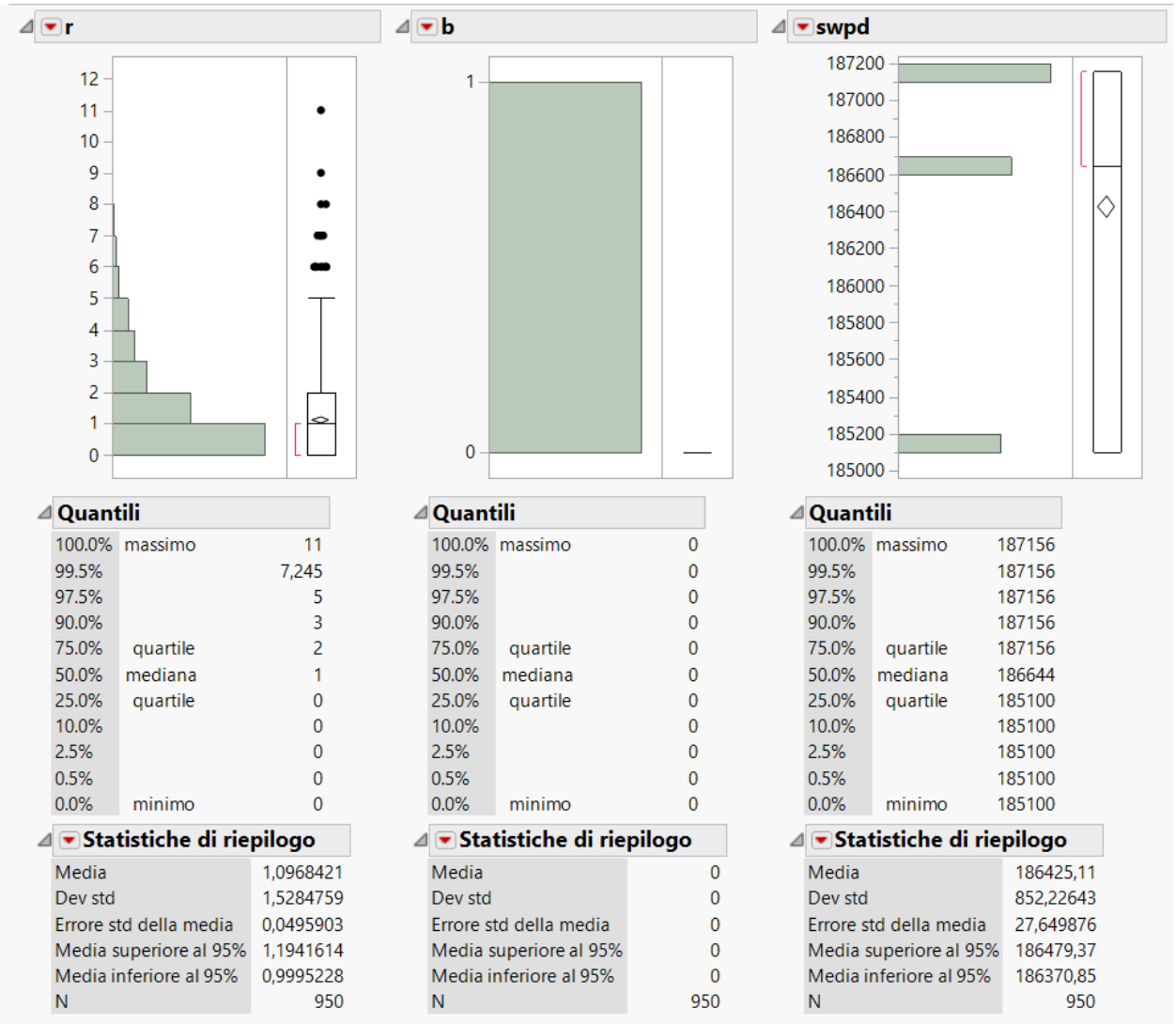
È stato quindi costruito il workload sintetico su JMeter basato su queste richieste, con l'ipotesi che avrebbe dato risultati a basso livello statisticamente simili a quelli della sollecitazione reale.

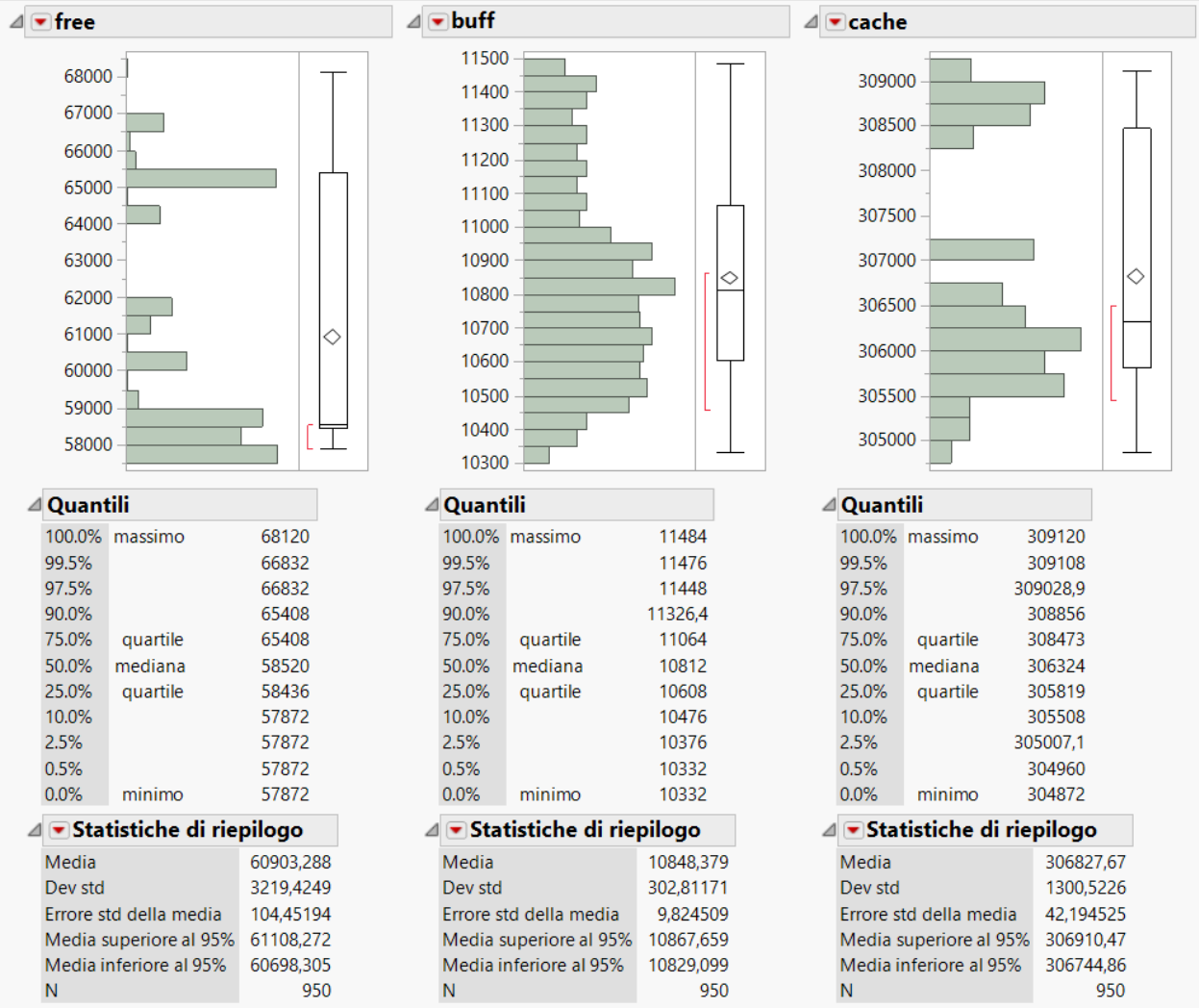
Caratterizzazione LL

Per verificare la similitudine tra le variazioni delle risorse a basso livello tra workload reale e sintetico, è necessario innanzitutto definire un **processo di caratterizzazione** per i dati a basso livello, che così come verrà effettuato su quelli reali, andrà ripetuto anche su quelli sintetici. È stato quindi caricato il dataset ottenuto dalle misurazioni *vmstat* in JMP:

	r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa	st
1	0	0	185100	61232	10332	308296	4	34	212	53	1506	394	1	36	64	0	0
2	0	0	185100	61232	10332	308308	0	0	0	0	181	668	0	1	99	0	0
3	0	0	185100	61232	10332	308308	0	0	0	0	130	401	1	2	97	0	0
4	1	0	185100	61232	10332	308308	0	0	0	0	96	229	0	0	100	0	0
5	1	0	185100	61232	10332	308308	0	0	0	0	296	1194	0	1	99	0	0
6	0	0	185100	61232	10332	308308	0	0	0	0	90	240	0	0	100	0	0
7	0	0	185100	61232	10344	308308	0	0	0	36	93	218	0	0	100	0	0
8	0	0	185100	61232	10344	308308	0	0	0	0	101	254	0	0	100	0	0
9	1	0	185100	60560	10344	308420	0	0	120	0	198	364	1	2	97	0	0
10	0	0	185100	60112	10344	308432	0	0	0	0	636	411	0	19	81	0	0
11	1	0	185100	60112	10344	308436	0	0	0	0	636	440	0	19	81	0	0
12	1	0	185100	59664	10344	308436	0	0	0	0	966	437	0	34	66	0	0
13	2	0	185100	59440	10352	308440	0	0	0	640	812	406	0	25	75	0	0
14	0	0	185100	59440	10352	308444	0	0	0	0	1040	407	0	37	63	0	0
15	1	0	185100	59440	10352	308444	0	0	0	0	631	369	0	19	81	0	0

E come di consueto, sono state analizzate le distribuzioni campionarie:





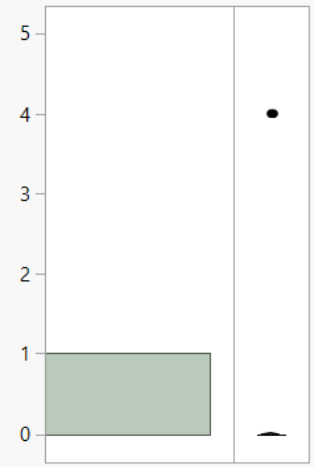
Quantili

		free	buff	cache
100.0%	massimo	68120	11484	309120
99.5%		66832	11476	309108
97.5%		66832	11448	309028,9
90.0%		65408	11326,4	308856
75.0%	quartile	65408	11064	308473
50.0%	mediana	58520	10812	306324
25.0%	quartile	58436	10608	305819
10.0%		57872	10476	305508
2.5%		57872	10376	305007,1
0.5%		57872	10332	304960
0.0%	minimo	57872	10332	304872

Statistiche di riepilogo

	free	buff	cache
Media	60903,288	10848,379	306827,67
Dev std	3219,4249	302,81171	1300,5226
Errore std della media	104,45194	9,824509	42,194525
Media superiore al 95%	61108,272	10867,659	306910,47
Media inferiore al 95%	60698,305	10829,099	306744,86
N	950	950	950

si



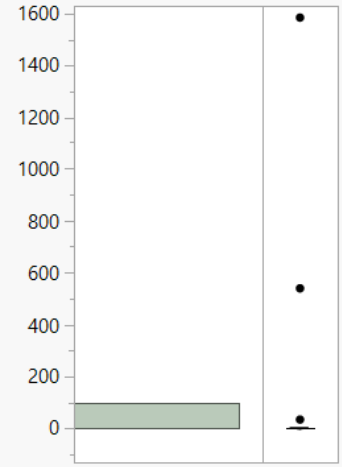
Quantili

100.0%	massimo	4
99.5%		0
97.5%		0
90.0%		0
75.0%	quartile	0
50.0%	mediana	0
25.0%	quartile	0
10.0%		0
2.5%		0
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

Media	0,0084211
Dev std	0,1834359
Errore std della media	0,0059514
Media superiore al 95%	0,0201006
Media inferiore al 95%	-0,003258
N	950

so



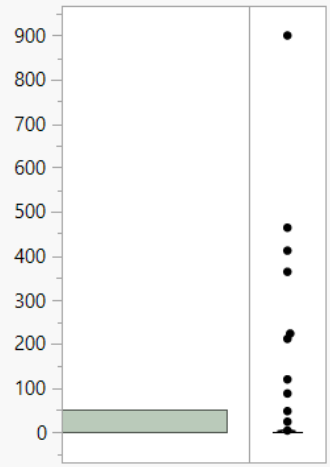
Quantili

100.0%	massimo	1584
99.5%		0
97.5%		0
90.0%		0
75.0%	quartile	0
50.0%	mediana	0
25.0%	quartile	0
10.0%		0
2.5%		0
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

Media	2,2715789
Dev std	54,288287
Errore std della media	1,7613446
Media superiore al 95%	5,7281594
Media inferiore al 95%	-1,185001
N	950

bi



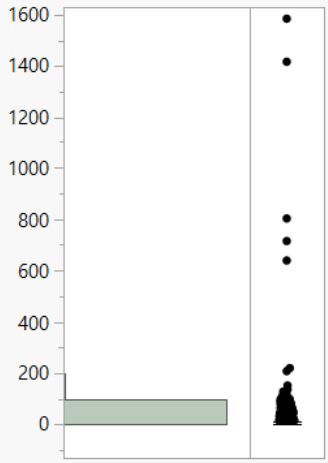
Quantili

100.0%	massimo	900
99.5%		258,3
97.5%		0
90.0%		0
75.0%	quartile	0
50.0%	mediana	0
25.0%	quartile	0
10.0%		0
2.5%		0
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

Media	3,0105263
Dev std	38,941283
Errore std della media	1,263422
Media superiore al 95%	5,4899502
Media inferiore al 95%	0,5311024
N	950

bo



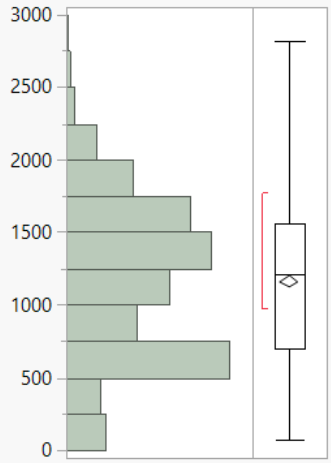
Quantili

100.0%	massimo	1584
99.5%		658,62
97.5%		84
90.0%		32
75.0%	quartile	5
50.0%	mediana	0
25.0%	quartile	0
10.0%		0
2.5%		0
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

Media	13,887368
Dev std	82,110574
Errore std della media	2,6640188
Media superiore al 95%	19,115417
Media inferiore al 95%	8,6593197
N	950

in



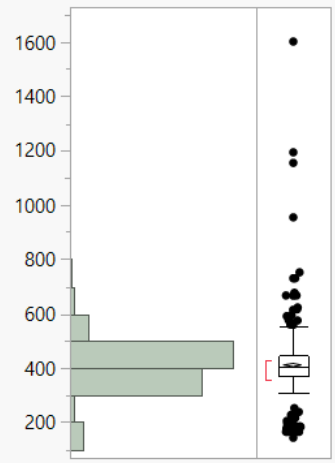
Quantili

100.0%	massimo	2819
99.5%		2528,7
97.5%		2156
90.0%		1859,9
75.0%	quartile	1560,5
50.0%	mediana	1213
25.0%	quartile	694,25
10.0%		508,1
2.5%		84
0.5%		73,51
0.0%	minimo	69

Statistiche di riepilogo

Media	1162,6821
Dev std	545,84431
Errore std della media	17,709528
Media superiore al 95%	1197,4365
Media inferiore al 95%	1127,9277
N	950

cs



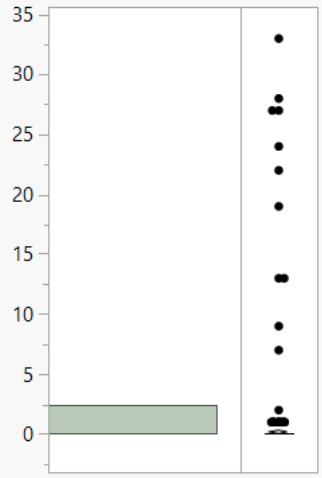
Quantili

100.0%	massimo	1601
99.5%		802,49
97.5%		551,45
90.0%		484,9
75.0%	quartile	446,25
50.0%	mediana	408,5
25.0%	quartile	371
10.0%		346
2.5%		178,775
0.5%		167
0.0%	minimo	145

Statistiche di riepilogo

Media	410,08947
Dev std	93,338746
Errore std della media	3,0283088
Media superiore al 95%	416,03243
Media inferiore al 95%	404,14652
N	950

us



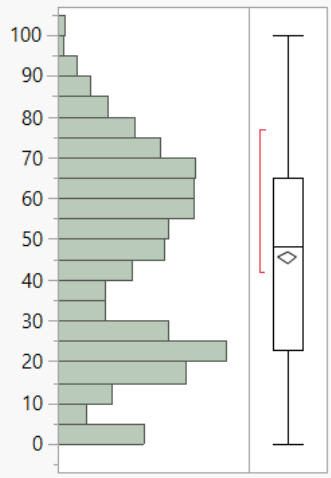
Quantili

100.0%	massimo	33
99.5%		24,735
97.5%		1
90.0%		0
75.0%	quartile	0
50.0%	mediana	0
25.0%	quartile	0
10.0%		0
2.5%		0
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

Media	0,2505263
Dev std	2,3359505
Errore std della media	0,0757882
Media superiore al 95%	0,3992582
Media inferiore al 95%	0,1017944
N	950

sy



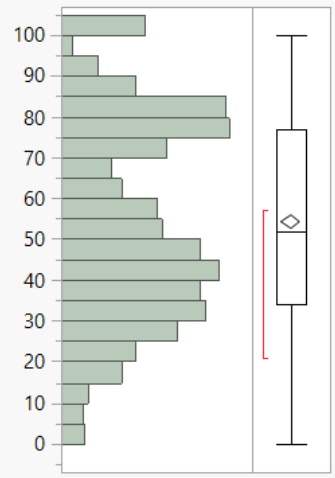
Quantili

100.0%	massimo	100
99.5%		99,245
97.5%		88
90.0%		76
75.0%	quartile	65
50.0%	mediana	48
25.0%	quartile	23
10.0%		14
2.5%		0
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

Media	45,525263
Dev std	24,535922
Errore std della media	0,7960504
Media superiore al 95%	47,087486
Media inferiore al 95%	43,963041
N	950

id

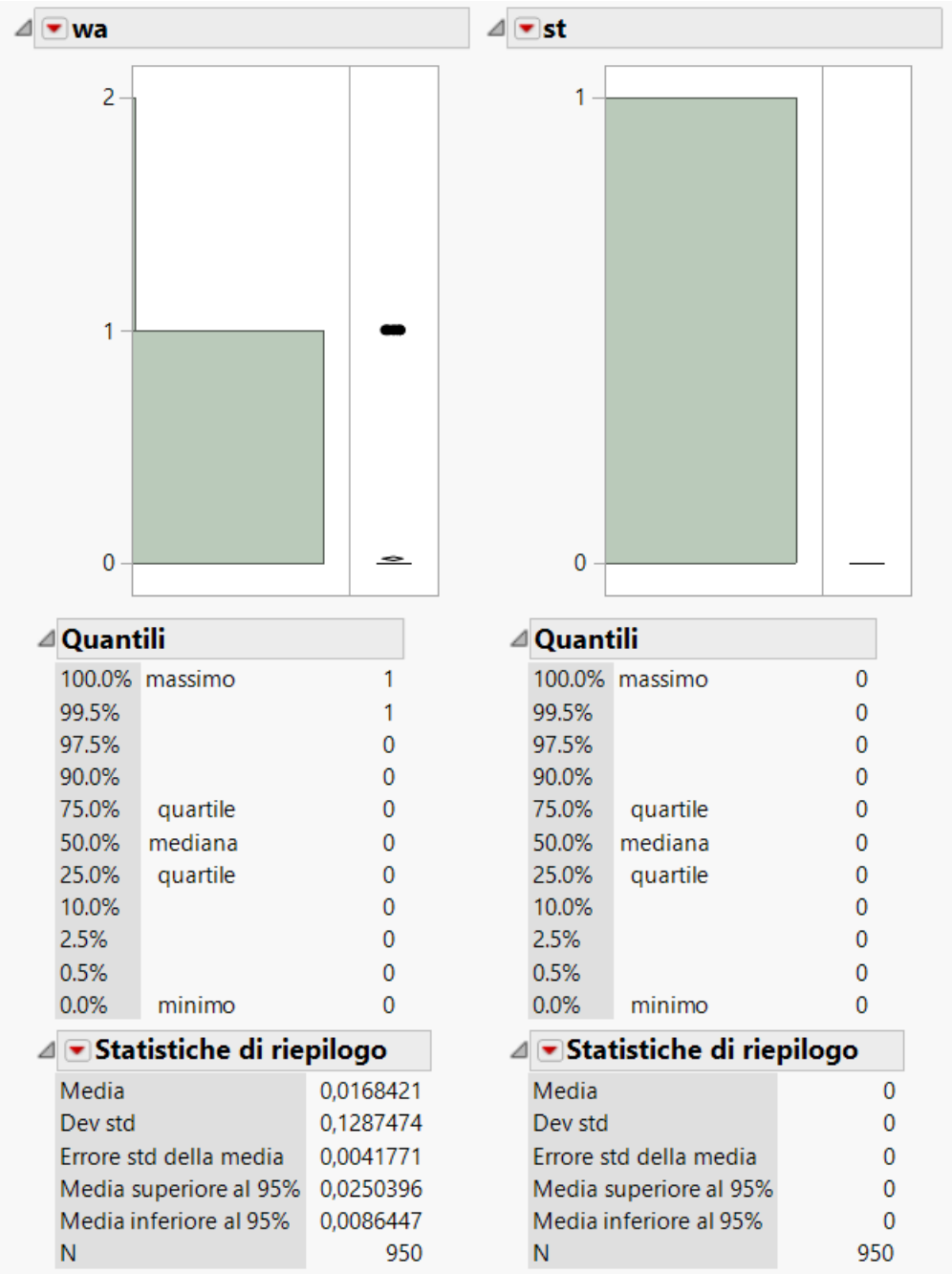


Quantili

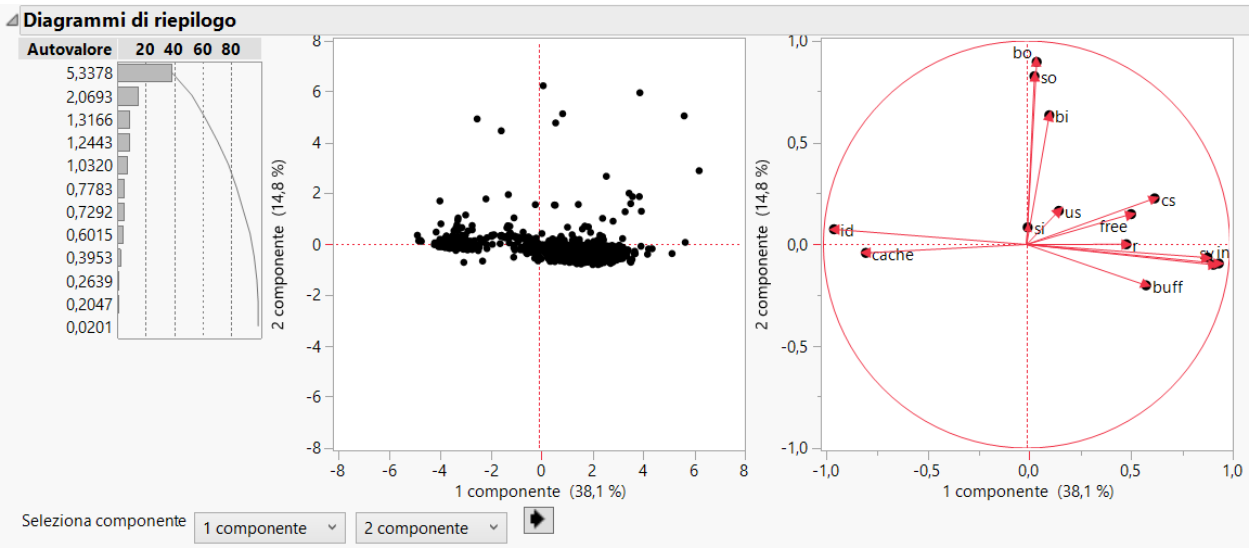
100.0%	massimo	100
99.5%		100
97.5%		100
90.0%		85,9
75.0%	quartile	77
50.0%	mediana	52
25.0%	quartile	34
10.0%		23
2.5%		9,775
0.5%		0
0.0%	minimo	0

Statistiche di riepilogo

Media	54,210526
Dev std	24,85723
Errore std della media	0,806475
Media superiore al 95%	55,793207
Media inferiore al 95%	52,627846
N	950



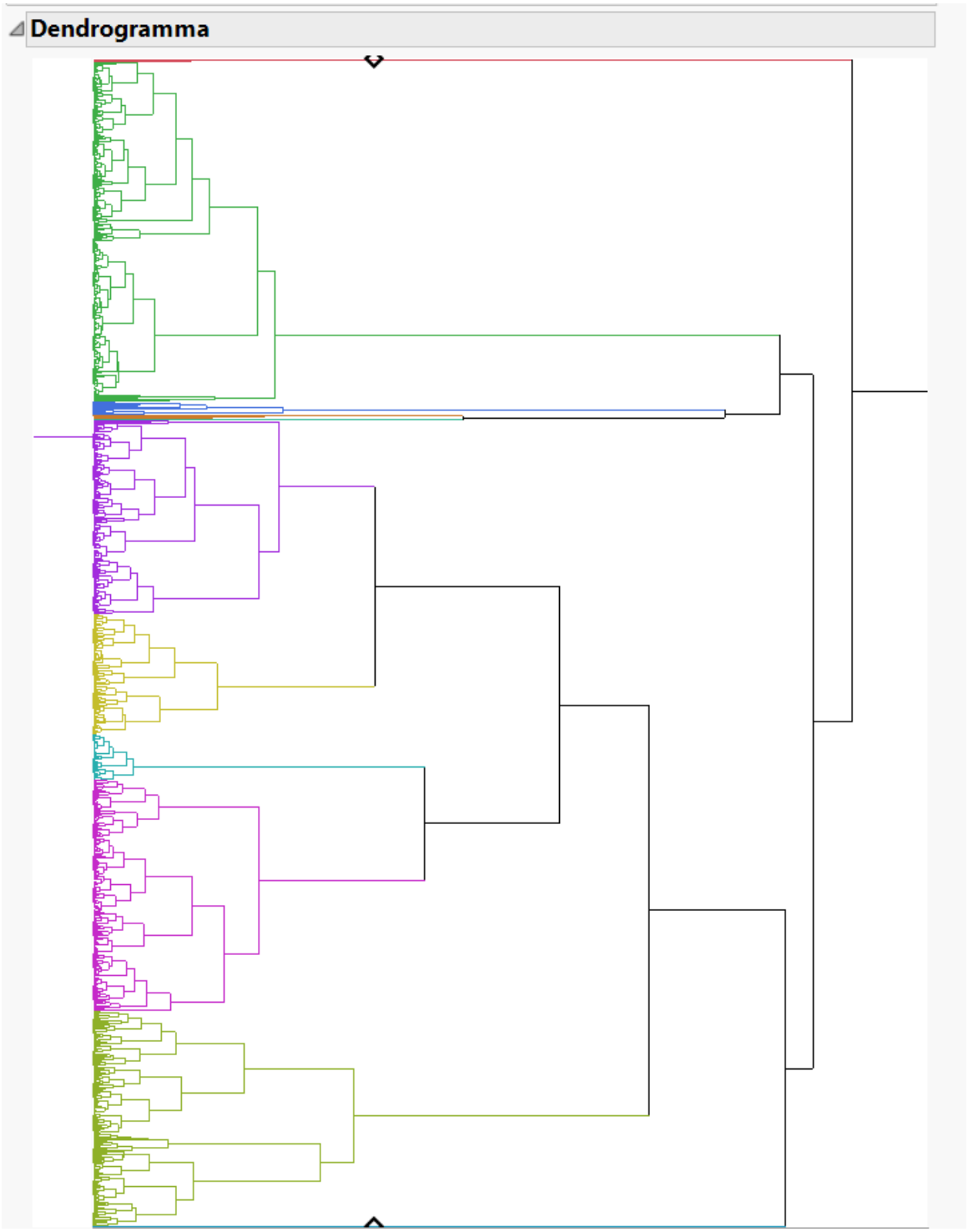
Le colonne eliminate sono **b**, **wa** e **st**, in quanto praticamente costanti. Si è provveduto quindi all'analisi PCA:



Autovalori

Numero	Autovalore	Percentuale	20 40 60 80	Percentuale cumulativa	Chi-quadrato	DF	Prob>ChiQu
1	5,3378	38,127		38,127	19555,4	90,385	<,0001*
2	2,0693	14,781		52,908	16284,6	86,041	<,0001*
3	1,3166	9,404		62,312	15205,1	76,143	<,0001*
4	1,2443	8,888		71,200	14634,8	65,617	<,0001*
5	1,0320	7,371		78,571	13917,6	55,796	<,0001*
6	0,7783	5,559		84,131	13211,0	46,392	<,0001*
7	0,7292	5,208		89,339	12649,7	37,670	<,0001*
8	0,6015	4,296		93,635	11844,4	29,981	<,0001*
9	0,3953	2,824		96,459	10817,0	23,211	<,0001*
10	0,2639	1,885		98,343	9873,77	17,159	<,0001*
11	0,2047	1,462		99,805	8818,22	11,718	<,0001*
12	0,0201	0,143		99,949	4952,34	7,253	<,0001*
13	0,0072	0,051		100,000	3994,23	3,562	<,0001*
14	0,0000	0,000		100,000	.	0,982	.

Qui si è optato per un valore di **7 componenti principali**, in quanto una percentuale cumulativa dell'89% è stata ritenuta sufficiente per gli scopi dell'analisi. Stavolta, il numero di clustering che è stato scelto è di **11 cluster**:



di cui si riportano i parametri di devianza ottenuti:

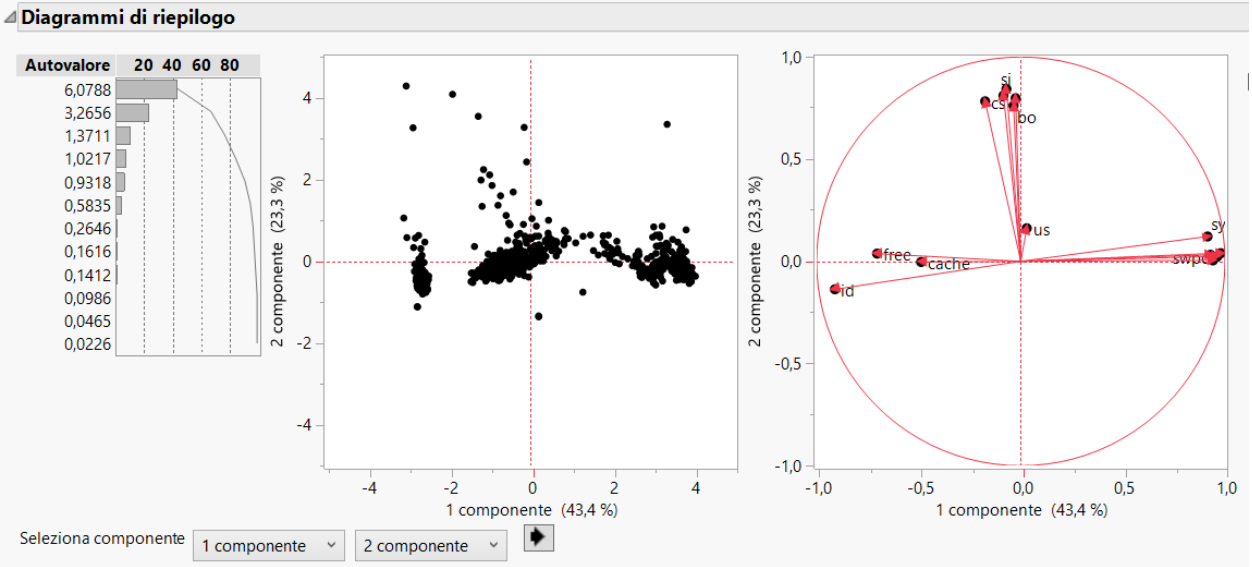
Intra-Cluster Deviance	Inter-Cluster Deviance	Total Deviance Loss (PCA + Clustering)
0.1613	0.8387	0.2507

Per ogni cluster, sono stati quindi selezionati in **maniera casuale** dei campioni da utilizzare per il test di ipotesi della verifica della somiglianza tra i dati reali e quelli sintetici. Si riportano quindi i dati selezionati, uno per ogni cluster:

Principale1	Principale2	Principale3	Principale4	Principale5	Principale6	Principale7
-1,47756882	4,467555974	3,7924958	4,371621253	20,04408234	-5,518952302	-0,916649109
-3,477515316	0,00445111	0,460993922	0,349584306	-0,241184033	-0,229372848	0,236057012
3,689898479	1,88019301	8,994065945	-5,016411129	-0,772130668	1,232919955	-4,211873173
5,721291904	5,054605181	5,58458792	3,813526359	1,039834977	4,905284716	5,586533723
5,893783812	27,12092347	1,62314844	3,779513127	1,615088692	6,171004714	10,10625
0,556423515	-0,350370446	-0,284903068	0,300537765	0,074969172	0,386517516	-0,520200766
1,2085072	-0,706638957	-0,799634652	1,267359137	0,076524423	1,114063394	-0,320841205
-1,12606105	0,465925482	-0,655676119	-2,876521987	0,364752164	-0,896796722	0,831234625
-0,442878626	0,136646326	-1,502371153	-1,278209941	0,551011662	0,809241787	0,193190617
3,182180439	-0,178288442	0,246906122	-0,33519213	-0,214894452	-0,790108111	0,327793959
0,819642974	28,50423102	-10,99207726	1,960676359	-5,81858709	-6,520302032	-11,38896958

Caratterizzazione LL'

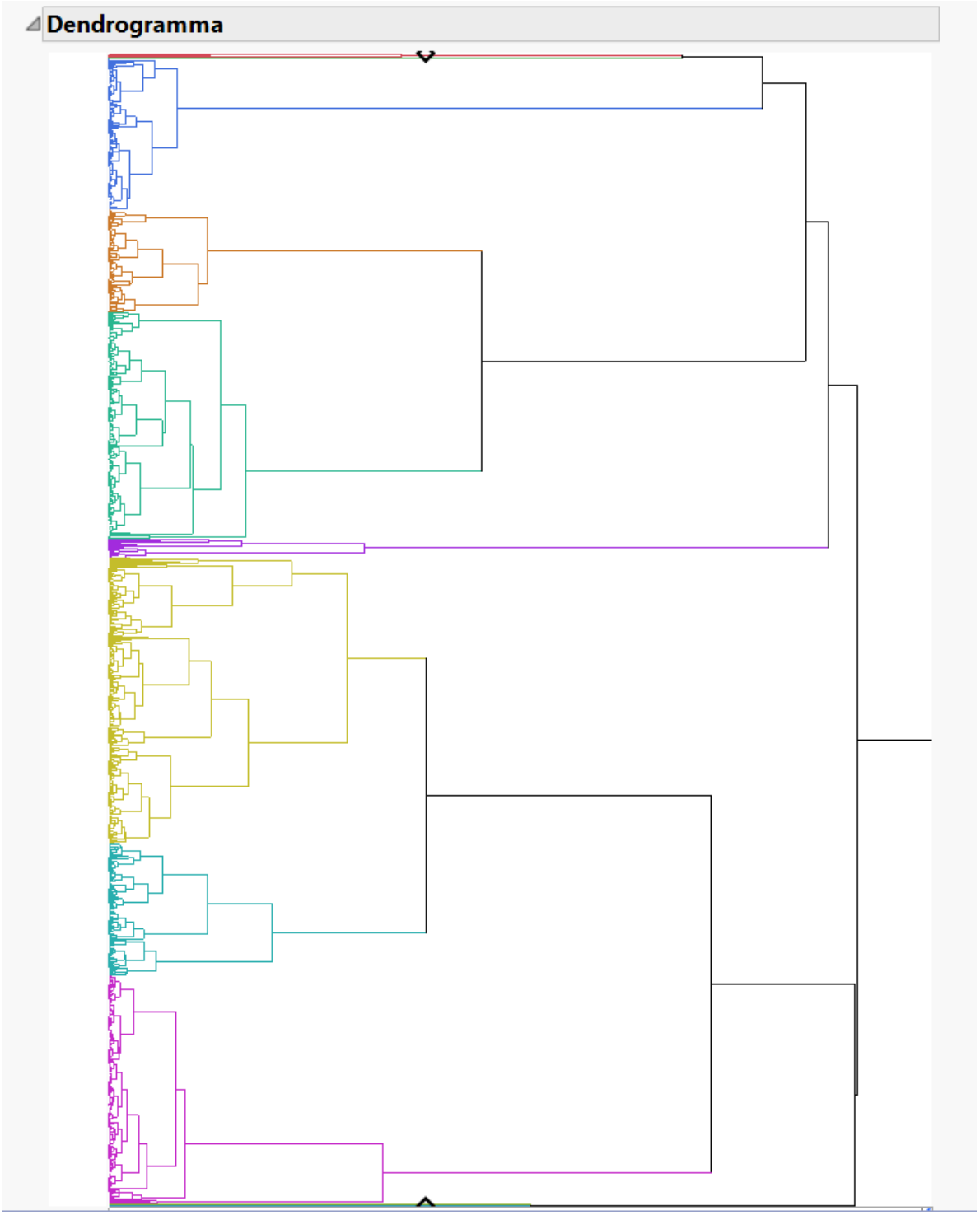
Dopo aver sollecitato nuovamente il sistema, stavolta con il **workload sintetico**, sono stati raccolti semplicemente i report dei dati a basso livello, per poter effettuare i test statistici necessari a verificarne la somiglianza. Come nel caso del workload reale, sono state eliminate le stesse colonne superflue, e si è provveduto a effettuare PCA e Clustering allo stesso modo in cui sono stati effettuati nella controparte reale.



Autovalori

Numero	Autovalore	Percentuale	20 40 60 80	Percentuale cumulativa	Chi-quadrato	DF	Prob>ChiQu
1	6,0788	43,420		43,420	25445,7	90,260	<,0001*
2	3,2656	23,326		66,746	21237,3	87,339	<,0001*
3	1,3711	9,794		76,540	17845,3	79,449	<,0001*
4	1,0217	7,298		83,838	16372,0	68,413	<,0001*
5	0,9318	6,655		90,493	14968,4	57,985	<,0001*
6	0,5835	4,168		94,661	12783,3	48,705	<,0001*
7	0,2646	1,890		96,551	10701,7	39,448	<,0001*
8	0,1616	1,155		97,706	9719,18	30,996	<,0001*
9	0,1412	1,008		98,714	9110,15	23,392	<,0001*
10	0,0986	0,704		99,418	8193,01	16,941	<,0001*
11	0,0465	0,332		99,750	7044,59	11,378	<,0001*
12	0,0226	0,161		99,912	6276,94	6,780	<,0001*
13	0,0124	0,088		100,000	5731,01	3,470	<,0001*
14	0,0000	0,000		100,000	0,000	0,986	0,9995

Qui si scelgono quindi 7 componenti, stavolta con una percentuale di varianza cumulativa più alta, come c'era da aspettarselo in quanto le informazioni generate dovrebbero essere in quantità minore rispetto al workload reale, per cui ci sarà meno perdita a parità di componenti.



È stato poi effettuato il clustering a 11 cluster, con dei valori di devianza che vengono riportati di seguito:

Intra-Cluster Deviance	Inter-Cluster Deviance	Total Deviance Loss (PCA + Clustering)
0.0974	0.9026	0.1286

E nuovamente, qui è stato selezionato un campione per ogni cluster (sempre scelto casualmente) in modo da ottenere un dataset ridotto su cui effettuare i test di ipotesi necessari.

Principale1	Principale2	Principale3	Principale4	Principale5	Principale6	Principale7
-4,238309461	22,27435534	-0,249150799	3,144440549	-1,649941269	-3,081869411	-8,877868669
-3,963582604	21,69599764	4,110717637	8,200256635	-15,81563258	6,53248613	2,984613088
-2,665438426	-0,213356593	-1,725939003	1,080583572	0,252559426	-0,01081881	-0,281145992
2,553289626	-0,120363429	-1,362004678	0,900365311	0,098535719	-0,333201265	0,435755565
3,647871022	-0,219306896	-0,365969167	-0,204071651	-0,099084302	0,562873506	-0,049283022
3,215008357	0,859393892	2,210770731	2,633067878	2,777124796	1,448095425	-0,525290169
-0,254251103	-0,17783034	0,903775232	-0,626757874	-0,279640988	-0,297997108	0,139864354
-0,274603067	0,128050939	0,02789563	0,403575187	-0,025803882	-1,141636712	0,423590483
-2,736975266	-0,625335924	0,637164781	-0,857035047	-0,180720845	0,719475909	-0,002950256
-2,595503218	38,24589558	-3,794011695	-5,121106778	2,1086069	2,507844954	5,776947788
1,713965439	15,81455578	-8,864489431	-11,88038873	13,085768	-0,842779163	-0,677311741

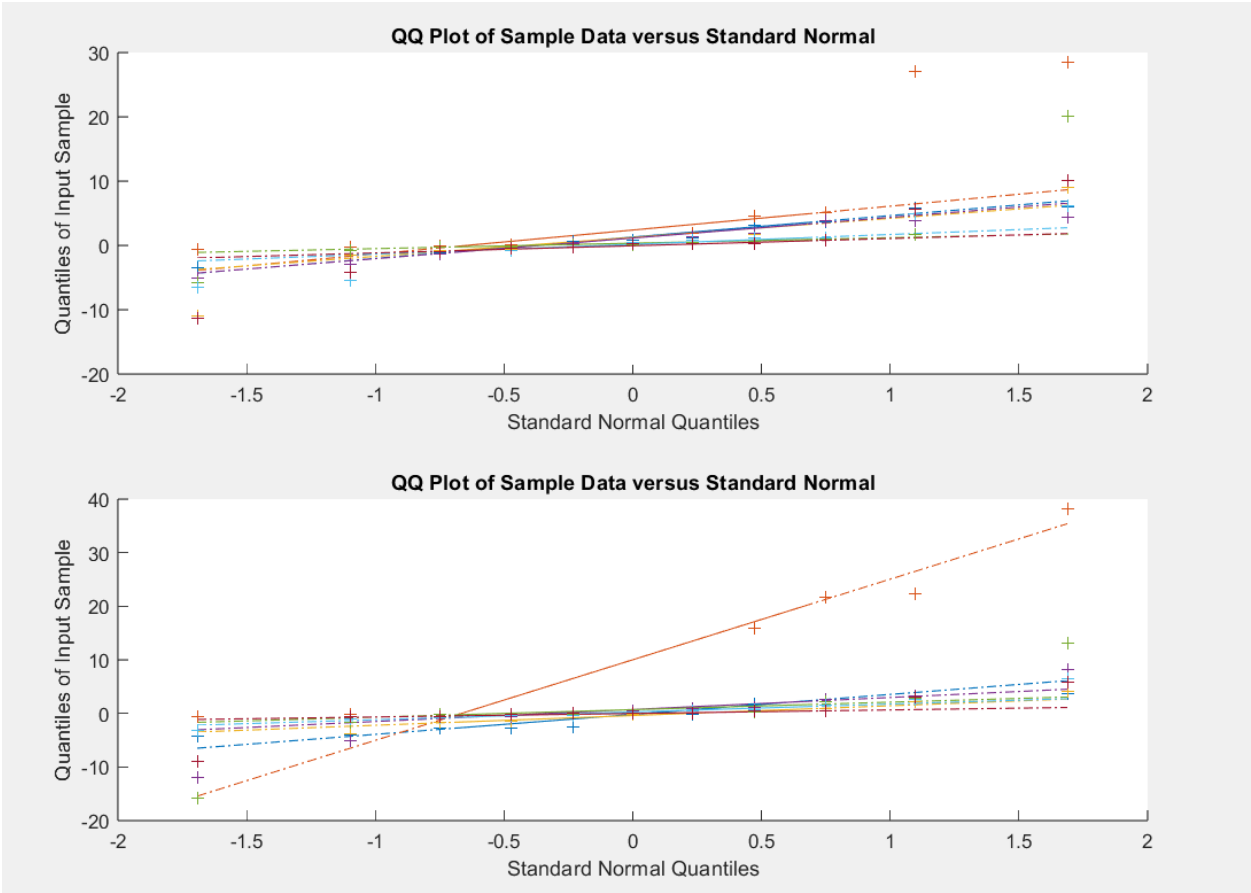
Test di ipotesi

Per verificare di aver ottenuto un workload sintetico abbastanza rappresentativo rispetto al workload reale, si è reso necessario effettuare una serie di **test di ipotesi** per confrontare le caratterizzazioni a basso livello reali e sintetiche.

A tale scopo, è stato utilizzato uno script Matlab apposito:

```
clear all; close all; clc;

%% Acquisizione dei dati:
real=xlsread("C:\Users\letiz\OneDrive\Desktop\Impianti di elaborazione\WC\reale.xlsx"); %workload reale
synthetic=xlsread("C:\Users\letiz\OneDrive\Desktop\Impianti di elaborazione\WC\sintetico.xlsx"); %workl
N=size(real,2); %indice del numero di colonne

%% Distribuzione normale: test visivo
figure();
subplot(2,1,1);
qqplot(real);
subplot(2,1,2);
qqplot(synthetic);

%% Distribuzione normale: test di Kolmogorov-Smirnov
[h_ks_real,p_ks_real] = kstest(real);
[h_ks_synth,p_ks_synth] = kstest(synthetic);

%% Test di ipotesi non parametrico: test di Wilcoxon
p_wilc = zeros(1,N);
h_wilc = zeros(1,N);
for i= 1:N
    [p_wilc(i), h_wilc(i)] = ranksum(synthetic(:,i), real(:,i));
end
```

Mentre questo viene commentato, vengono valutati anche i suoi output. Innanzitutto, c'è la lettura dei file in cui sono stati salvati i valori per-cluster precedentemente salvati delle componenti su LL e LL'. Dopodiché, è stato visualizzato un **qq-plot** per entrambe le distribuzioni, per testare in maniera visiva la normalità dei dati:



Come è facile notare, questi dati **non risultano normali**. Ciò che si può fare per appurarlo matematicamente (e con una buona confidenza) è quello di utilizzare un test di ipotesi chiamato **Test di Kolmogorov-Smirnov**, definito come segue: ****

$$H_0: \text{ il vettore dei dati viene da una distribuzione normale standard}$$
$$H_1: \text{ il vettore dei dati non viene da una distribuzione normale standard}$$

In particolare, il risultato del test è 1 se **il test rigetta l'ipotesi nulla** a un livello di significatività del 5%, e 0 altrimenti. Come si può notare, in questo caso si ottiene 1 per entrambi i vettori di valori:

h_ks_real	1
h_ks_synth	1

e quindi, è possibile affermare che **il test rigetta l'ipotesi nulla** e quindi i dati **non sono normali**. Di conseguenza, per confrontare i due vettori, è necessario utilizzare un **test non parametrico**, in particolare si è deciso di utilizzare il **Test di Wilcoxon**, anche detto **Test *ranksum***. Il test è definito come:

$$H_0: \text{ i dati sono sample di distribuzioni continue con le stesse mediane}$$
$$H_1: \text{ i dati sono sample di distribuzioni con mediane diverse}$$

La decisione si basa sul risultato del test che, per ogni componente principale (ogni colonna), se è 0 indica che **non si può rigettare l'ipotesi nulla** a un livello di significatività del 5%.

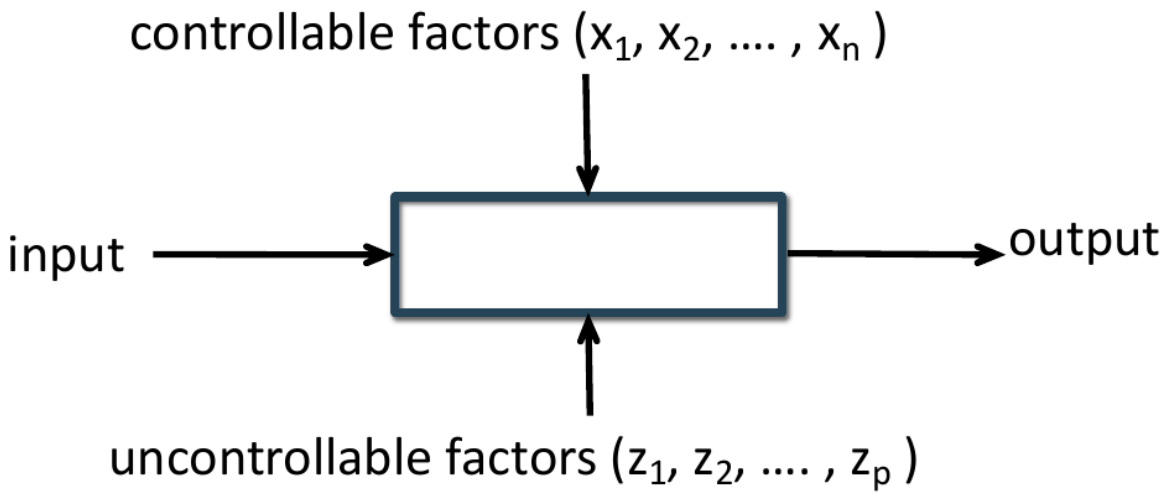
h_wilc	[0,0,0,0,0,0,0]

Come si può vedere, ogni componente principale riporta valore nullo, e quindi si può dire che **i dati provengono da distribuzioni con la stessa mediana**.

DoE

Progettazione esperimento

Si definisce un **esperimento** come un test in cui si identificano dei **fattori controllabili** rispetto all'insieme dei fattori che cambiano la relazione input/output di uno specifico sistema.



Scegliere i fattori su cui effettuare un esperimento è cruciale, in quanto prendere in considerazione un insieme di fattori da far variare vuol dire considerare anche le **interazioni tra di essi**. È quindi importante definire bene **cosa** si vuole misurare, in termini di:

- **Response variable.** Il valore su cui vogliamo controllare l'**influenza** dei vari fattori.
- **Fattori.** Vanno scelti in base a ciò che si vuole valutare.
- **Livelli dei fattori.** I vari valori che i fattori possono assumere.
- **Replicazione dell'esperimento.** Quante volte ripetere l'esperimento affinché dia risultati soddisfacenti.

Nel caso in esame, il nostro sistema è il **Web Server** che è stato precedentemente caratterizzato. Si vuole quindi studiare l'evoluzione del **tempo di risposta** al variare dei parametri di **dimensione di pagina** e **intensità (throughput richiesto)**.

Per i fattori in esame sono stati scelti vari livelli, che vengono riportati di seguito:

- **Page Type:**
  - **Low:** 20KB.
  - **Mid-Low:** 184KB.
  - **Mid:** 716KB.
  - **Mid-High:** 2.13MB.
  - **High:** 5.07MB.
- **Intensity:**
  - **25% della Usable Capacity:** 375.
  - **75% della Usable Capacity:** 1125.

I livelli sono stati scelti in base ai vari range delle risorse disponibili sul Web Server (che sono ovviamente le stesse adoperate per la caratterizzazione tramite **Capacity Test**) e in base alla **Usable Capacity** (sempre caratterizzata precedentemente) che si è osservata intorno alle 1500 richieste al minuto.

Essendo il numero dei fattori pari a 2, e il numero dei livelli pari a 5 per il Page Type e a 2 per l'Intensity, si potrebbe pensare di effettuare un **Full Factorial Design**, senza perdere informazione sull'interazione tra i fattori. Nello specifico, il numero di test **totali** da eseguire, scegliendo di effettuare 5 ripetizioni per ogni test, è:

$$(2 \cdot 5) \cdot 5 = 50$$

Si è scelto di eseguire ogni test per 1 minuto, per un tempo totale di misurazioni di 50 minuti. Per effettuare i test, è stato adoperato nuovamente il tool **Jmeter**, ottenendo le informazioni sul **tempo di risposta medio** dal tool **Summary Report**.

Misurazioni e analisi dataset

Una volta effettuati tutti i test, si è ottenuto il **dataset delle misurazioni** riportato di seguito:

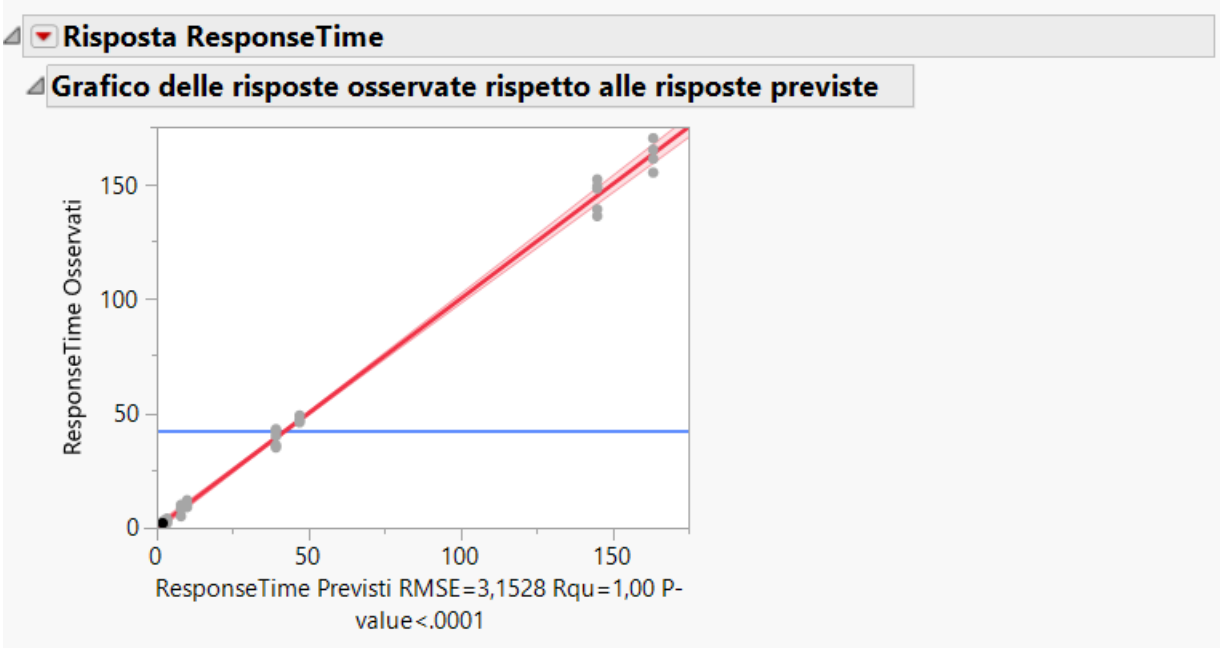
Intensity	Page Type	Response Time
375	Low	2
375	Low	2
375	Low	2
375	Low	2
375	Low	2
1125	Low	2
1125	Low	3
1125	Low	2
1125	Low	2
1125	Low	2
375	Mid-Low	3
375	Mid-Low	2
375	Mid-Low	2
375	Mid-Low	3
375	Mid-Low	3
1125	Mid-Low	4
1125	Mid-Low	3
1125	Mid-Low	4
1125	Mid-Low	2
1125	Mid-Low	4
375	Mid	5
375	Mid	8
375	Mid	9
375	Mid	10
375	Mid	8
1125	Mid	10
1125	Mid	9
1125	Mid	12
1125	Mid	9
1125	Mid	10
375	Mid-High	43
375	Mid-High	40
375	Mid-High	36
375	Mid-High	35
375	Mid-High	42
1125	Mid-High	47
1125	Mid-High	46
1125	Mid-High	46
1125	Mid-High	49
1125	Mid-High	47
375	High	139

Intensity	Page Type	Response Time
375	High	149
375	High	152
375	High	136
375	High	148
1125	High	170
1125	High	155
1125	High	165
1125	High	161
1125	High	165

La struttura tabellare è stata generata tramite JMP con lo strumento di **Generazione Piano Personalizzato DOE**, in cui è stato possibile definire i fattori e i loro livelli associandogli un label e una relazione d'ordine.

Analisi di importanza

Per cominciare l'analisi dei risultati ottenuti, è stata generata una **stima di modello** a partire dal piano DOE generato precedentemente. La stima è una regressione lineare con minimizzazione del **Mean Square Error** ed **enfasi su leverage degli effetti**. Di seguito è riportato il grafico della regressione effettuata:



Sono inoltre riportati l'**indice di determinazione** R^2 (insieme alla sua versione corretta) e i valori di **varianza spiegata** sia per il modello di regressione, sia per i fattori:

Riepilogo della stima

R-quadro	0,997646
R-quadro corretto	0,997116
Scarto quadratico medio	3,152777
Media della risposta	42,24
Osservazioni (o somme pesate)	50

Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	9	168479,52	18719,9	1883,294
Errore	40	397,60	9,9	**Prob > F**
C. totale	49	168877,12		<,0001*

Stime dei parametri

Test degli effetti

Origine	Nparm	DF	Somma dei quadrati	Rapporto F	Prob > F
Intensity	1	1	426,32	42,8893	<,0001*
PageType	4	4	167469,32	4212,005	<,0001*
Intensity*PageType	4	4	583,88	14,6851	<,0001*

Da questi dati, è possibile ottenere varie informazioni sulla stima effettuata. Innanzitutto, l'indice di determinazione pari a 0.9976 indica un'ottima resa in termini di stima, da cui ci si dovrebbe aspettare un errore sufficientemente basso, che in effetti è ciò che si nota nell'**analisi della varianza**. È quindi possibile determinare l'**importanza** dell'errore e dei vari fattori, di seguito riportate:

$$\frac{SSE}{SST} = \frac{397.60}{168877.12} = 0.24\% \implies \frac{SSR}{SST} = 99.76\%$$

$$\frac{SS_{Intensity}}{SST} = \frac{426.32}{168877.12} = 0.25\%$$

$$\frac{SS_{PageType}}{SST} = \frac{167469.32}{168877.12} = 99.17\%$$

$$\frac{SS_{Interaction}}{SST} = \frac{583.88}{168877.12} = 0.35\%$$

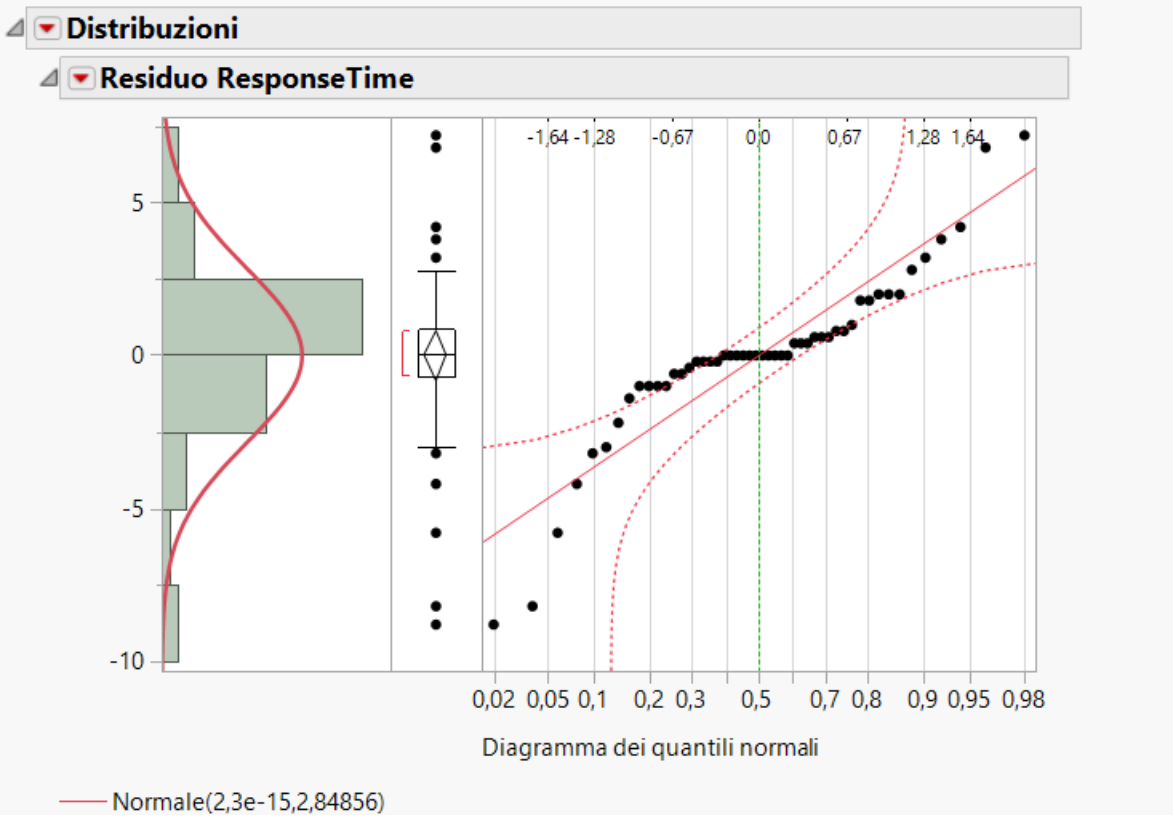
È quindi possibile vedere come il **PageType** sia il **fattore più importante**. Tuttavia, è da ricordare che l'Importanza **non è un fattore statistico**, ed è quindi da prendere solo come **parametro soggettivo**, in quanto quella che deve essere realmente controllata è la **Significatività**.

Analisi di significatività

Per valutare la **significatività statistica dei fattori**, è necessario applicare un metodo detto **ANOVA (Analysis Of Variance)**, la cui implementazione varia in base alle proprietà di **normalità** e **omoschedasticità** della distribuzione dei **residui**, i quali non sono altro che gli errori sulla stima del modello.

Normalità

Per valutare la **normalità** dei residui, si è pensato di utilizzare sia un metodo visivo che un test di ipotesi chiamato **Test di Shapiro-Wilk**, in quanto è adatto per dimensioni di sample inferiori a 2000.



Dal test visivo, un **plot quantile-quantile**, è già possibile appurare la non-normalità dei dati. Per ulteriore conferma di questa ipotesi, è riportato il test W di Shapiro-Wilk:

Test della bontà di adattamento

Test W di Shapiro-Wilk

W	Prob<W
0,883358	0,0001*

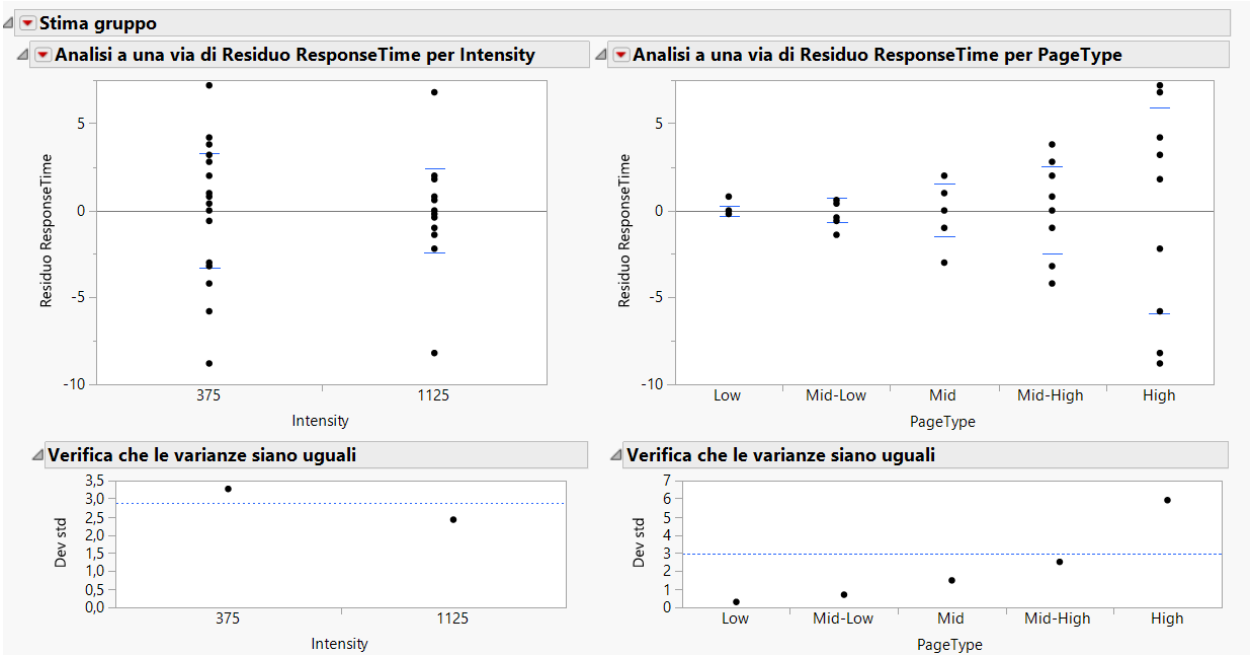
Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

Siccome la probabilità (il **p-value**) calcolato è estremamente basso, **si rigetta l'ipotesi nulla che i dati vengano da una distribuzione normale**.

Omoschedasticità

Essendo stata valutata la non normalità dei residui, sarebbe già possibile optare per un'analisi tramite **test di Kruskal-Wallis** o **test di Wilcoxon**. Tuttavia, per un'analisi più approfondita è stato deciso di valutare comunque l'**omoschedasticità** dei dati.

Questo è stato fatto innanzitutto tramite un test visivo, mostrando un grafico delle varianze dei residui rispetto ai due fattori e attivando un grafico di verifica dell'uguaglianza delle varianze.



Visivamente è già chiaro che, i residui dell'intensity **sembrano omoschedastici**, mentre quelli del Page Type sembrano **eteroscehdastici**. Questo è confermabile attraverso vari testi di ipotesi, che indicano quindi le due nature della varianza che sono state ipotizzate con i test visivi.

Livello	Conteggio	Dev std	Diff assoluta media rispetto alla media	Diff assoluta media rispetto alla mediana
375	25	3,271085	2,096000	2,096000
1125	25	2,422120	1,360000	1,360000

Test	Rapporto F	Num DF	Den DF	p-value
O'Brien[.5]	0,8635	1	48	0,3574
Brown-Forsythe	1,3457	1	48	0,2518
Levene	1,3457	1	48	0,2518
Bartlett	2,0915	1	.	0,1481
Test F bilaterale	1,8239	24	24	0,1481

Test di Welch

ANOVA di Welch verifica l'uguaglianza delle medie, ammette deviazioni standard non uguali

Rapporto F	Num DF	Den DF	Prob > F
0,0000	1	44,235	1,0000

Test t
0,0000

Livello	Conteggio	Dev std	Diff assoluta media rispetto alla media	Diff assoluta media rispetto alla mediana
Low	10	0,298142	0,160000	0,160000
Mid-Low	10	0,699206	0,600000	0,520000
Mid	10	1,490712	1,000000	1,000000
Mid-High	10	2,512192	1,880000	1,880000
High	10	5,921711	5,000000	4,640000

Test	Rapporto F	Num DF	Den DF	Prob > F
O'Brien[.5]	9,5995	4	45	<,0001*
Brown-Forsythe	8,6464	4	45	<,0001*
Levene	17,1380	4	45	<,0001*
Bartlett	16,8718	4	.	<,0001*

Test di Welch

ANOVA di Welch verifica l'uguaglianza delle medie, ammette deviazioni standard non uguali

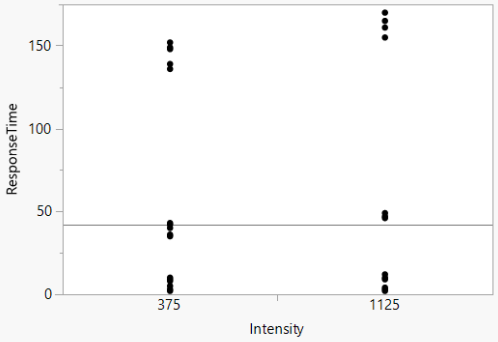
Rapporto F	Num DF	Den DF	Prob > F
0,0000	4	19,593	1,0000

ANOVA (Test di Kruskal-Wallis)

Poiché il **test di Kruskal-Wallis** si adatta sia al caso omoschedastico che a quello eteroschedastico (con spread di varianza non troppo spinti), è stato applicato su entrambe le distribuzioni della response variable. Si nota infatti che, nel caso dell'Intensity, il test di Kruskal-Wallis indica che **non si può rigettare l'ipotesi di non significatività**, mentre per il Page Type **si può rigettare l'ipotesi di non significatività**.

Stima gruppo

Analisi a una via di ResponseTime per Intensity



Test di Wilcoxon/Kruskal-Wallis (somme dei ranghi)

Livello	Conteggio	Somma degli score	Score atteso	Media degli score	(Media-Media0)/Std0
375	25	589,500	637,500	23,5800	-0,929
1125	25	685,500	637,500	27,4200	0,929

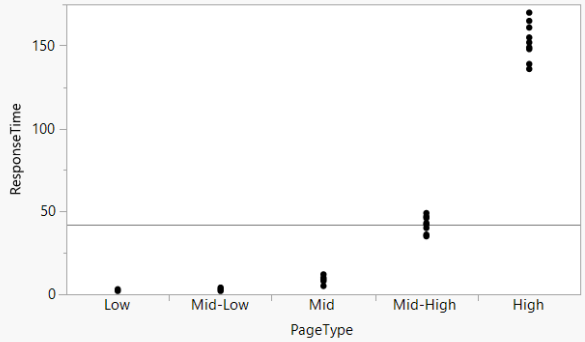
Test a due campioni, approssimazione normale

S	Z	Prob>|Z|
685,5	0,92884	0,3530

Test a una variabile, approssimazione chi-quadrato

Chi-quadrato	DF	Prob>ChiQu
0,8810	1	0,3479

Analisi a una via di ResponseTime per PageType



Test di Wilcoxon/Kruskal-Wallis (somme dei ranghi)

Livello	Conteggio	Somma degli score	Score atteso	Media degli score	(Media-Media0)/Std0
Low	10	73,500	255,000	7,3500	-4,424
Mid-Low	10	136,500	255,000	13,6500	-2,884
Mid	10	255,000	255,000	25,5000	0,000
Mid-High	10	355,000	255,000	35,5000	2,432
High	10	455,000	255,000	45,5000	4,876

Test a una variabile, approssimazione chi-quadrato

Chi-quadrato	DF	Prob>ChiQu
46,3565	4	<,0001*

Questo conclude l'analisi, in quanto si è dimostrato che il fattore realmente significativo è la dimensione delle pagine, mentre l'Intensity non ha una vera influenza sulla variabile di response, per cui si può dire che in realtà non è significativa.

Si potrebbero validare i risultati ottenuti dal test di Kruskal-Wallis per Page Type con un **Test di Friedman**, che si adatta meglio in contesti di eteroschedasticità.

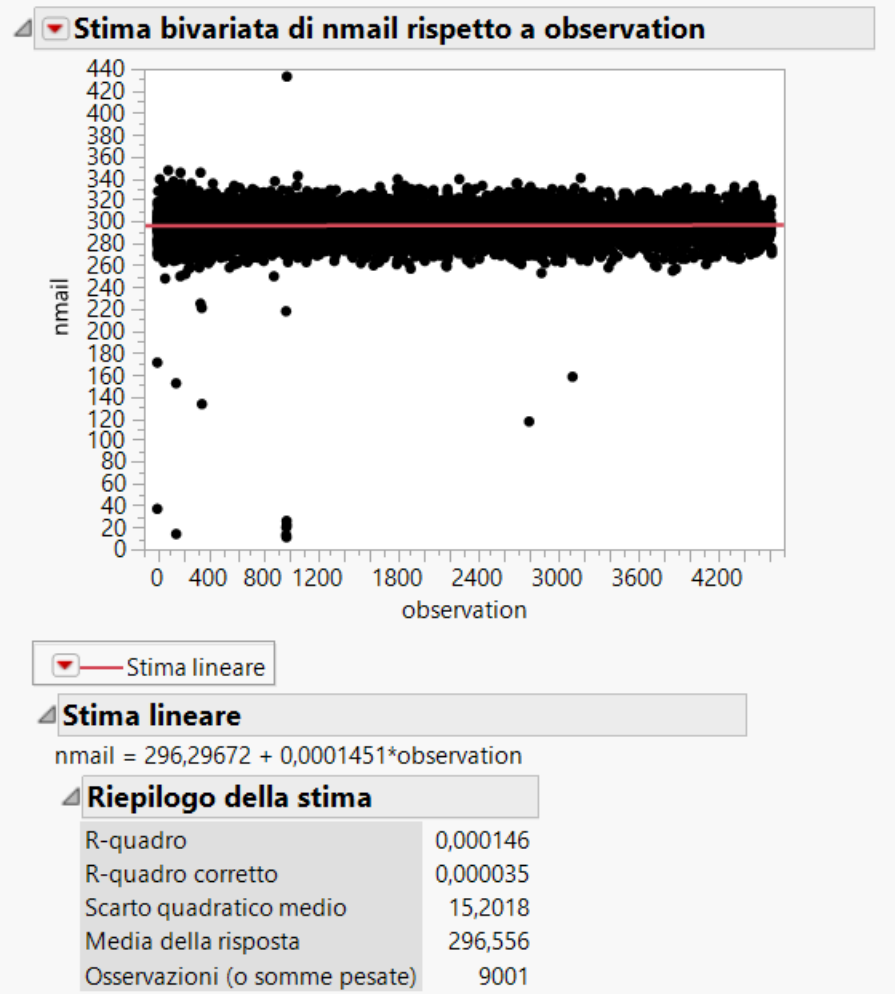
Regression

EXP 1

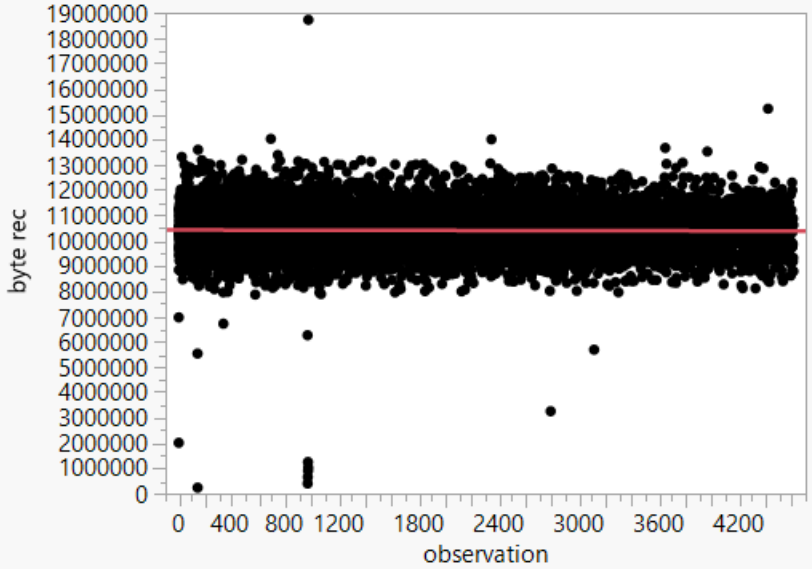
Per l'analisi di questo dataset si considerano *nmail*, *byte rec* e *byte sent* come variabili di risposta, mentre *observation* è considerata variabile di predizione.

Stima lineare

Per ottenere una stima lineare, è stato utilizzato il tool JMP utilizzando la funzione **"Stima Y (nmail, byte rec e byte sent) rispetto ad X (observation)"**.



Stima bivariata di byte rec rispetto a observation



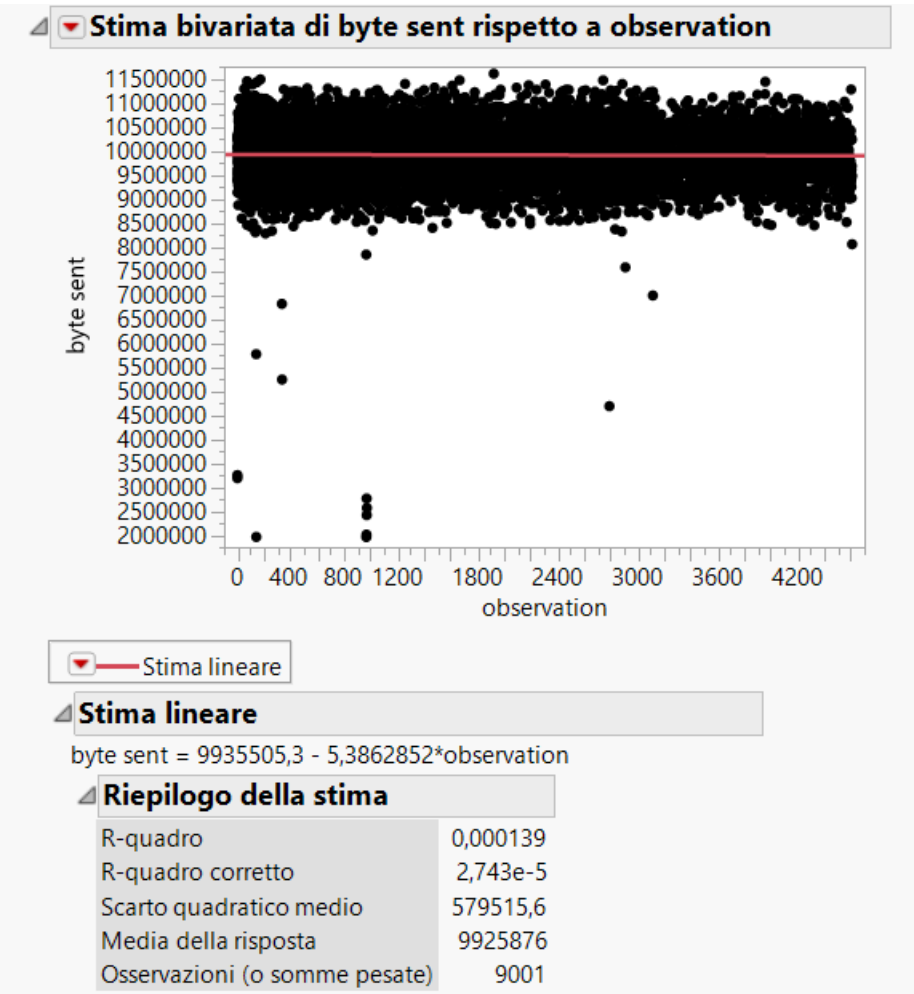
Stima lineare

Stima lineare

byte rec = 10432157 - 9,6329789*observation

Riepilogo della stima

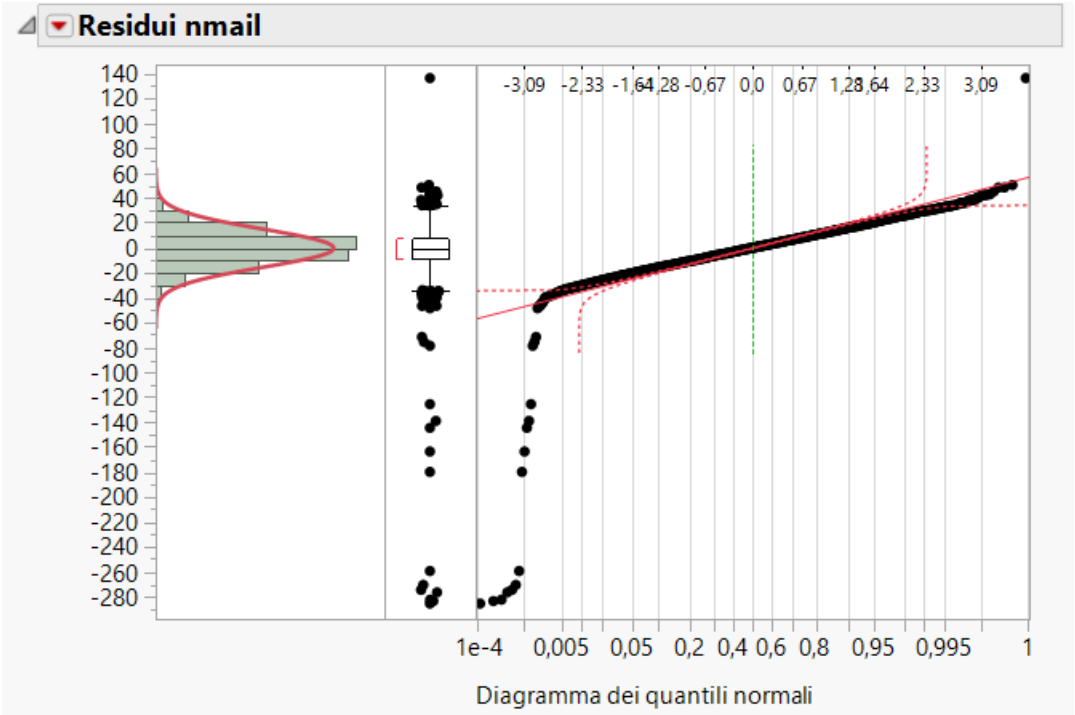
R-quadro	0,000179
R-quadro corretto	6,834e-5
Scarto quadratico medio	910638,4
Media della risposta	10414935
Osservazioni (o somme pesate)	9001

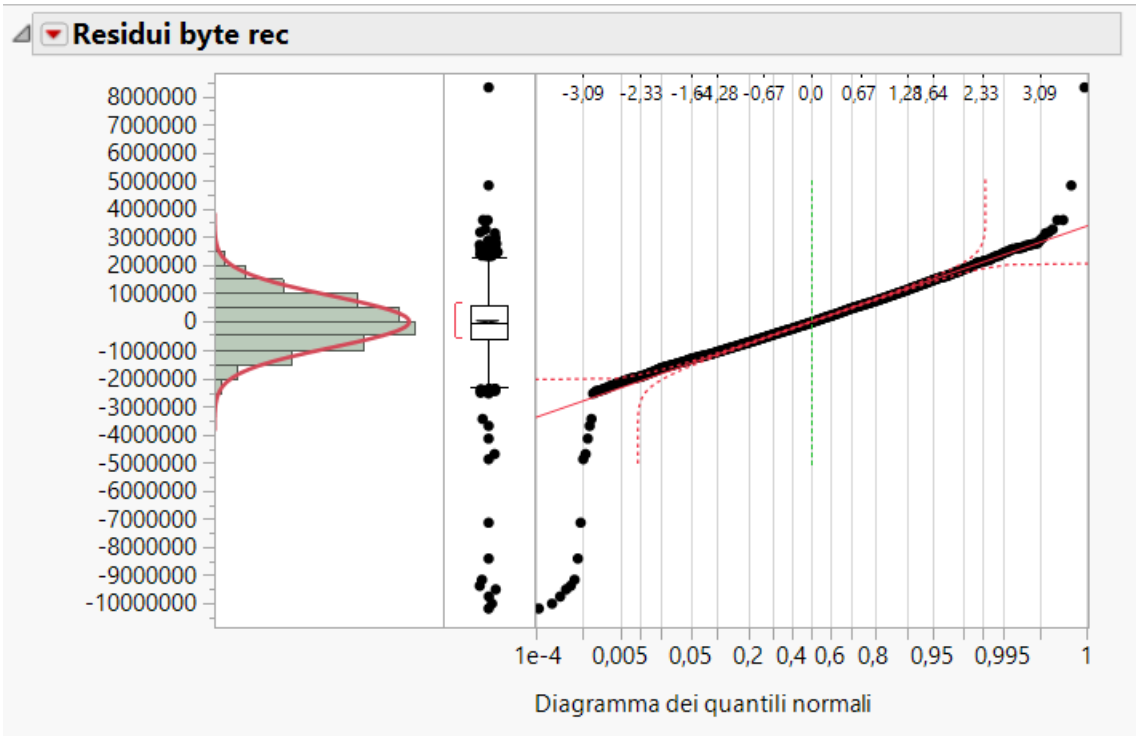


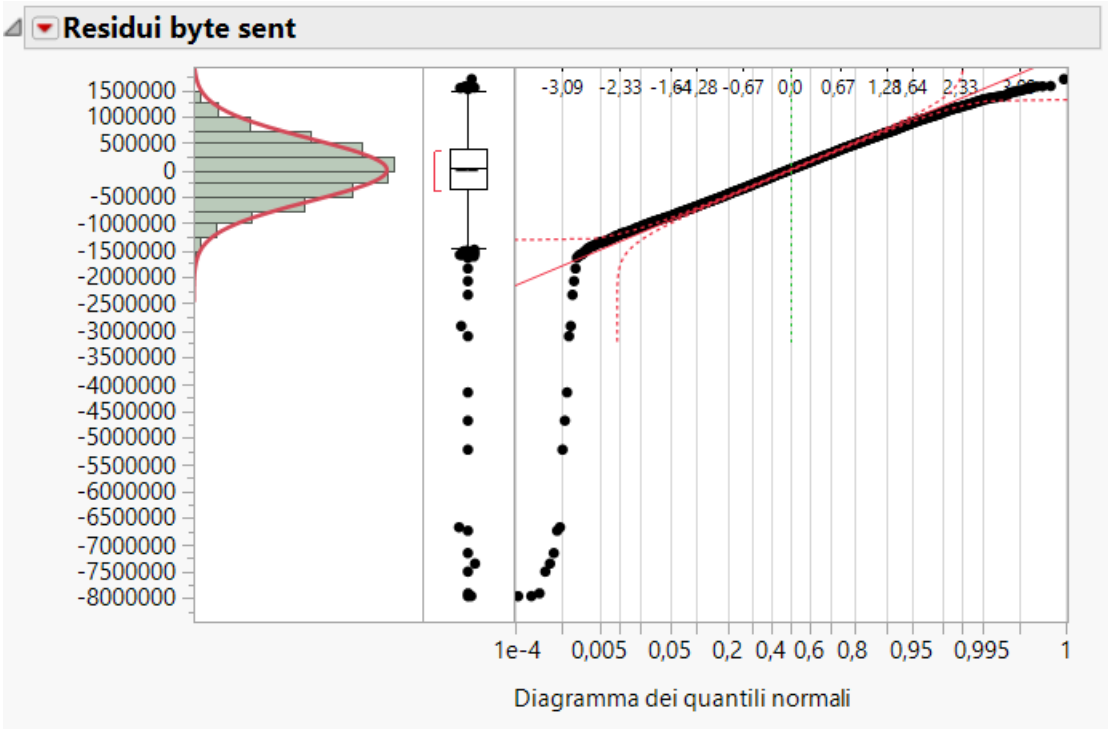
Si può notare che in tutti e tre i casi R^2 è un valore estremamente piccolo, per cui è possibile affermare che **i dati non sono lineari**.

Controllo della normalità

Si procede a fare un test visivo della normalità dei residui dei dati attraverso QQPlot, necessario per la scelta del test di regressione da eseguire in seguito


Residui nmail
140
120
100
80
60
40
20
0
-20
-40
-60
-80
-100
-120
-140
-160
-180
-200
-220
-240
-260
-280
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
1e-4 0,005 0,05 0,2 0,4 0,6 0,8 0,95 0,995 1
Diagramma dei quantili normali


Residui byte rec
8000000
7000000
6000000
5000000
4000000
3000000
2000000
1000000
0
-1000000
-2000000
-3000000
-4000000
-5000000
-6000000
-7000000
-8000000
-9000000
-10000000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
1e-4 0,005 0,05 0,2 0,4 0,6 0,8 0,95 0,995 1
Diagramma dei quantili normali



A partire da questi plot è possibile determinare che le distribuzioni **non sono quindi normali**.

Quanto osservato è ulteriormente supportato se si analizza la bontà di adattamento delle tre distribuzioni, calcolata mediante il **test di KSL** in quanto il numero di campioni della distribuzione supera i 2000 elementi:

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,056916	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

nmail

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,022067	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

byte rec

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,020797	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

byte sent

Avendo dimostrato la non-normalità dei dati, viene ritenuto inutile testare la loro omoschedasticità, per cui si procede direttamente ad effettuare un test di regressione lineare non parametrico: il test di Mann-Kendall.

Test di Mann-Kendall

Questo test è utilizzato per determinare se una serie temporale ha un trend monotonamente crescente o decrescente quando i dati non sono normali né lineari, a patto però che si abbia assenza di autocorrelazione.

Questo metodo fornisce due tipi di informazione:

- Il **Kendall Tau**, che misura la monotonia dei dati. Il τ varia tra -1 e 1; un valore prossimo a ± 1 indica una forte associazione tra le due variabili, che sarà positiva o negativa (correlazione o correlazione inversa) a seconda del segno, mentre un valore prossimo allo 0 indica assenza di relazione tra le variabili.
- La **significatività**, che rappresenta la soglia entro cui l'ipotesi di assenza di trend è accettata: il trend è statisticamente significativo se il p-value è minore di 0.05.

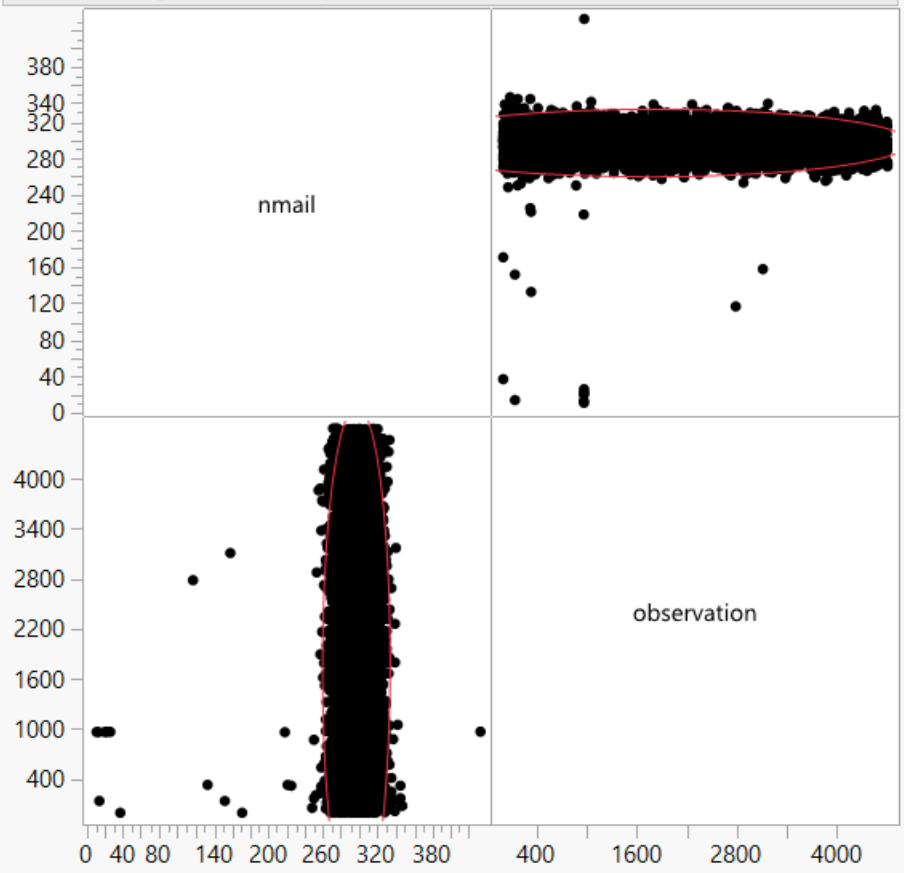
Si presentano dunque i risultati ottenuti a seguito del test:

Correlazioni

	nmail	observation
nmail	1,0000	0,0121
observation	0,0121	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

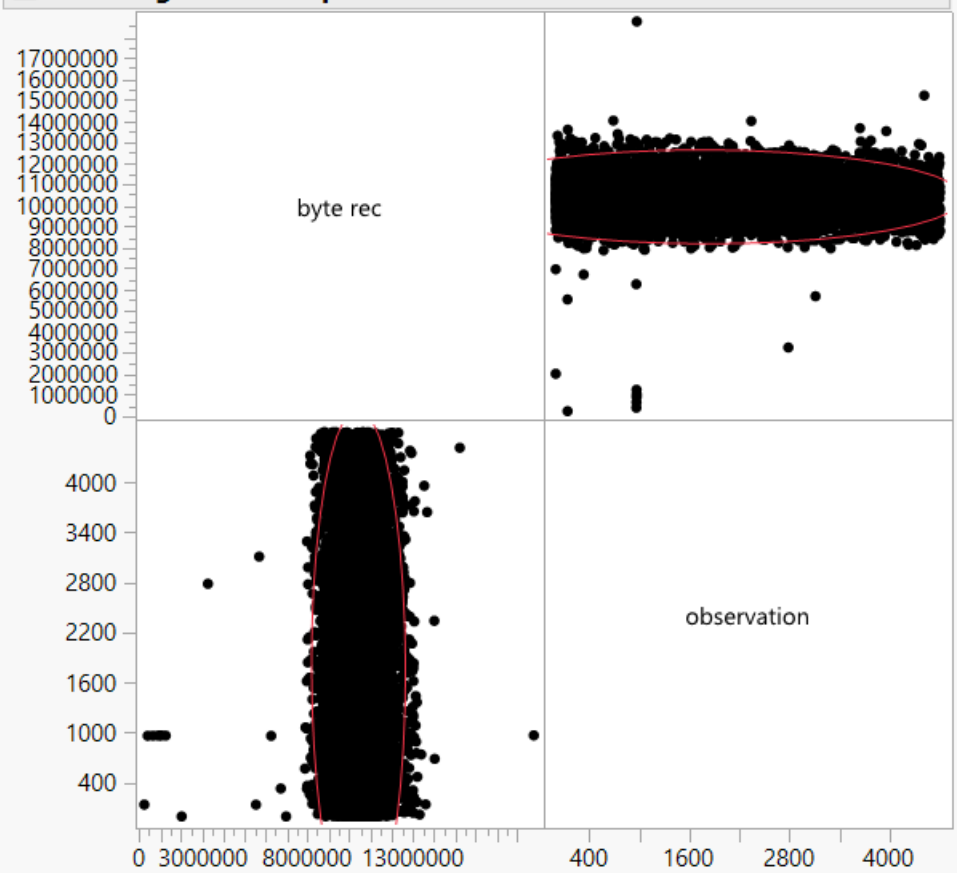
Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
observation	nmail	-0,0004	0,9509	

Correlazioni

	byte rec	observation
byte rec	1,0000	-0,0134
observation	-0,0134	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
observation	byte rec	-0,0122	0,0834	

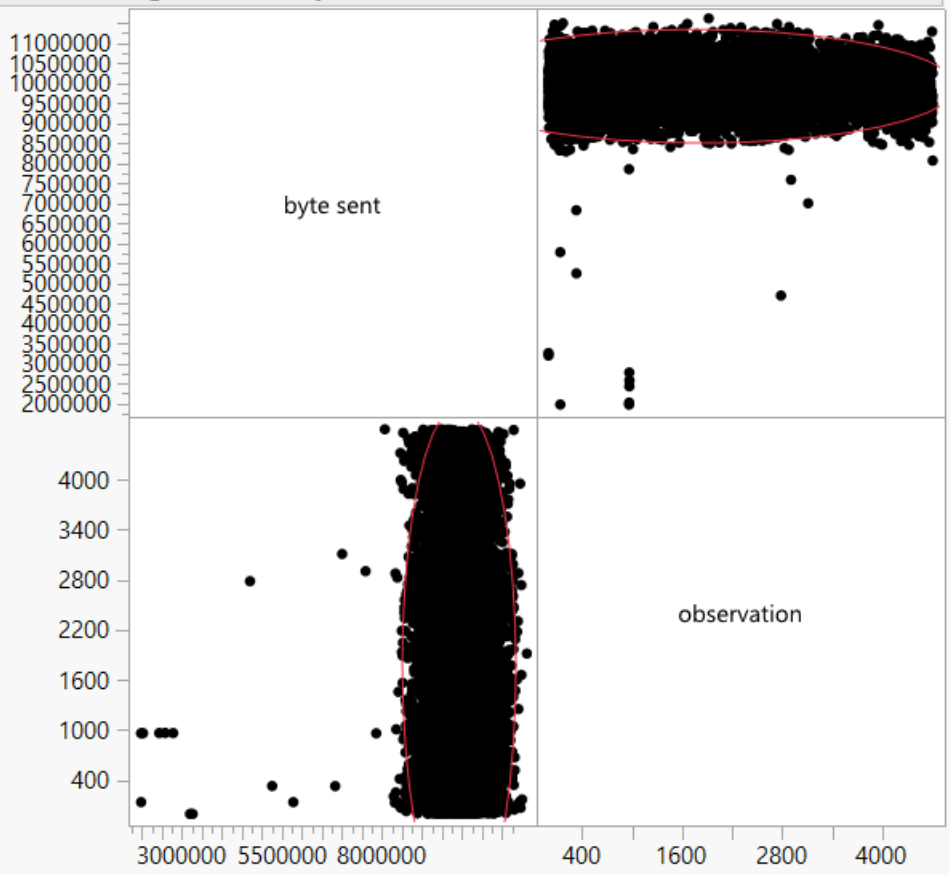
Multivariato

Correlazioni

	byte sent	observation
byte sent	1,0000	-0,0118
observation	-0,0118	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
observation	byte sent	-0,0146	0,0383*	

Si osserva quindi che *byte sent* è l'unica variabile a presentare trend, in quanto il p-value è minore di 0.05. Non è possibile invece rigettare l'ipotesi nulla (assenza di trend monotono) negli altri due casi.

Stimatore di Theil-Sen

Una volta dimostrato che la variabile *byte sent* presenta un trend, si ha interesse nell'individuare la pendenza e l'intercetta della sua retta di regressione.

Per farlo si utilizza lo **stimatore di Theil-Sen**, che valuta la pendenza della retta come la mediana di tutte le pendenze calcolate tra ogni coppia di punti della serie. È possibile ottenere i risultati desiderati utilizzando la funzione "*theilslopses*" della libreria SciPy:

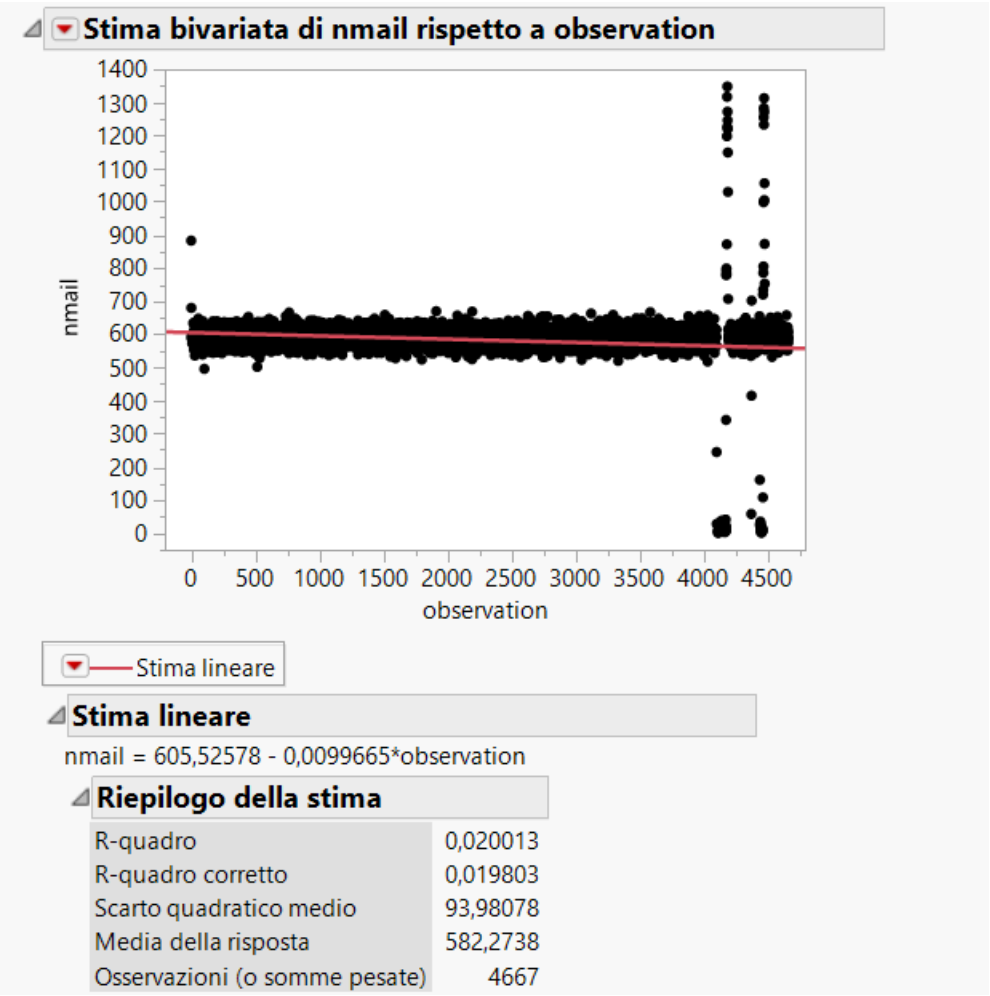
Pendenza	-9.667
Intercetta	9954543.874
Intervallo di confidenza superiore	-0.517
Intervallo di confidenza inferiore	-18.762

L'intervallo di confidenza scelto è quello di default del 95%: osservando quindi che l'intervallo di confidenza non include lo 0 è possibile affermare con una confidenza del 95% che la pendenza della retta sia diversa da 0.

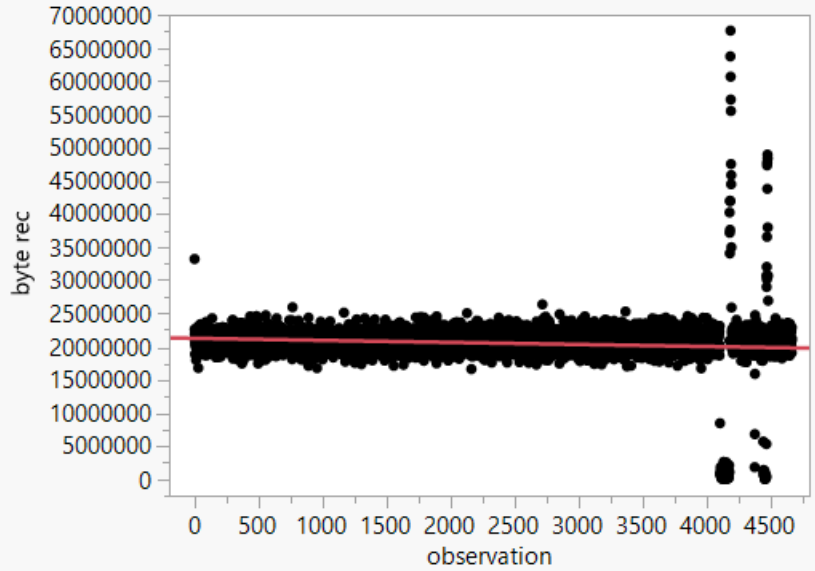
EXP 2

Per l'analisi di questo dataset si considerano *nmail*, *byte rec* e *byte sent* come variabili di risposta, mentre *observation* è considerata variabile di predizione.

Stima lineare



Stima bivariata di byte rec rispetto a observation



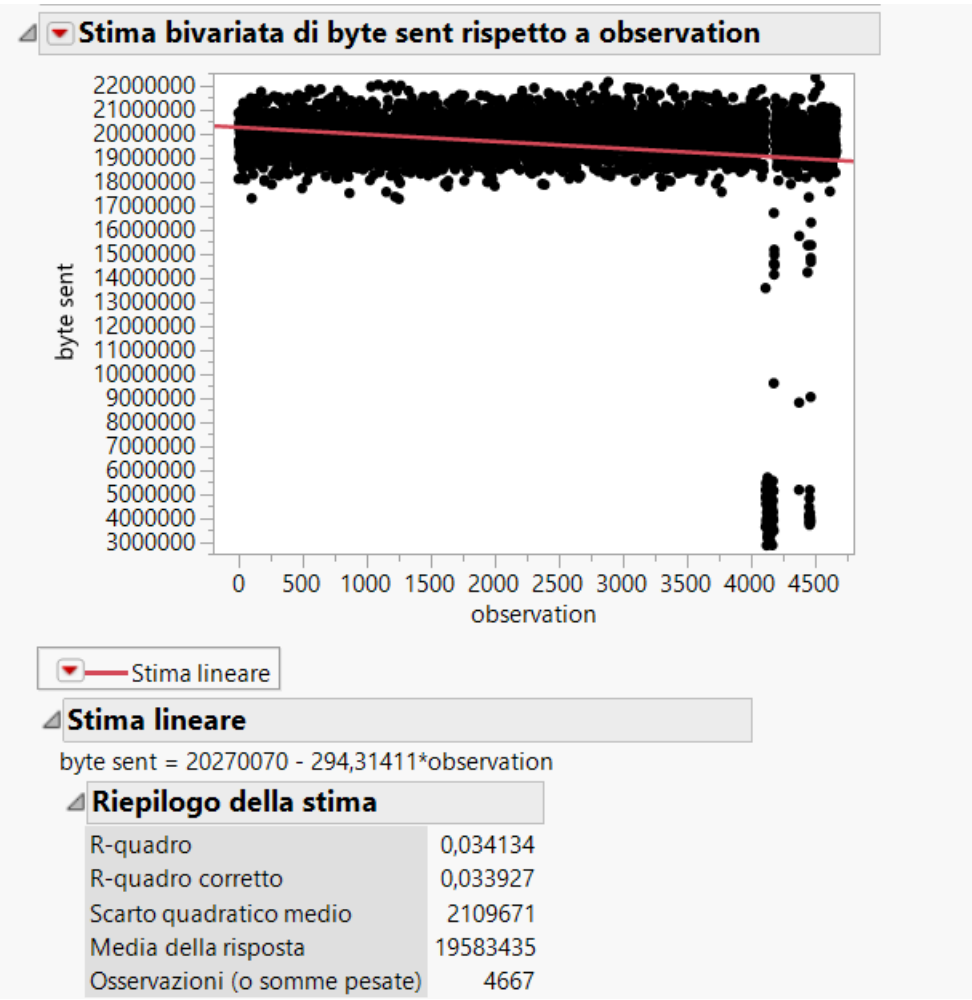
Stima lineare

Stima lineare

byte rec = 21263600 - 308,91128*observation

Riepilogo della stima

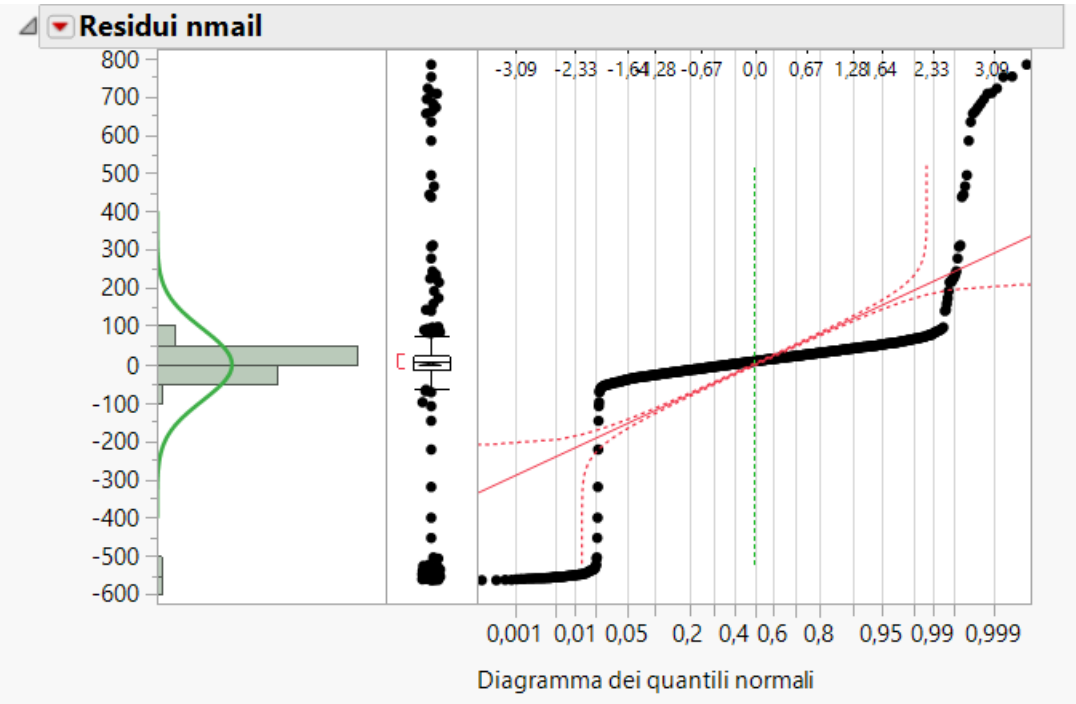
R-quadro	0,012592
R-quadro corretto	0,01238
Scarto quadratico medio	3686215
Media della risposta	20542910
Osservazioni (o somme pesate)	4667

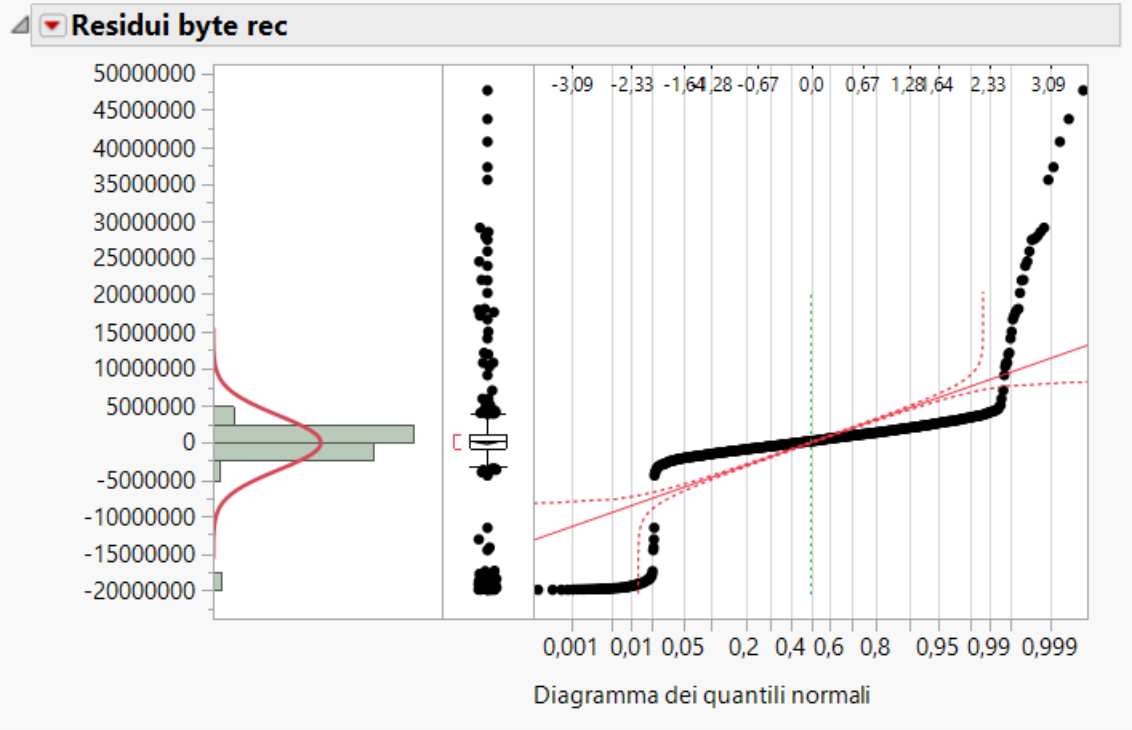


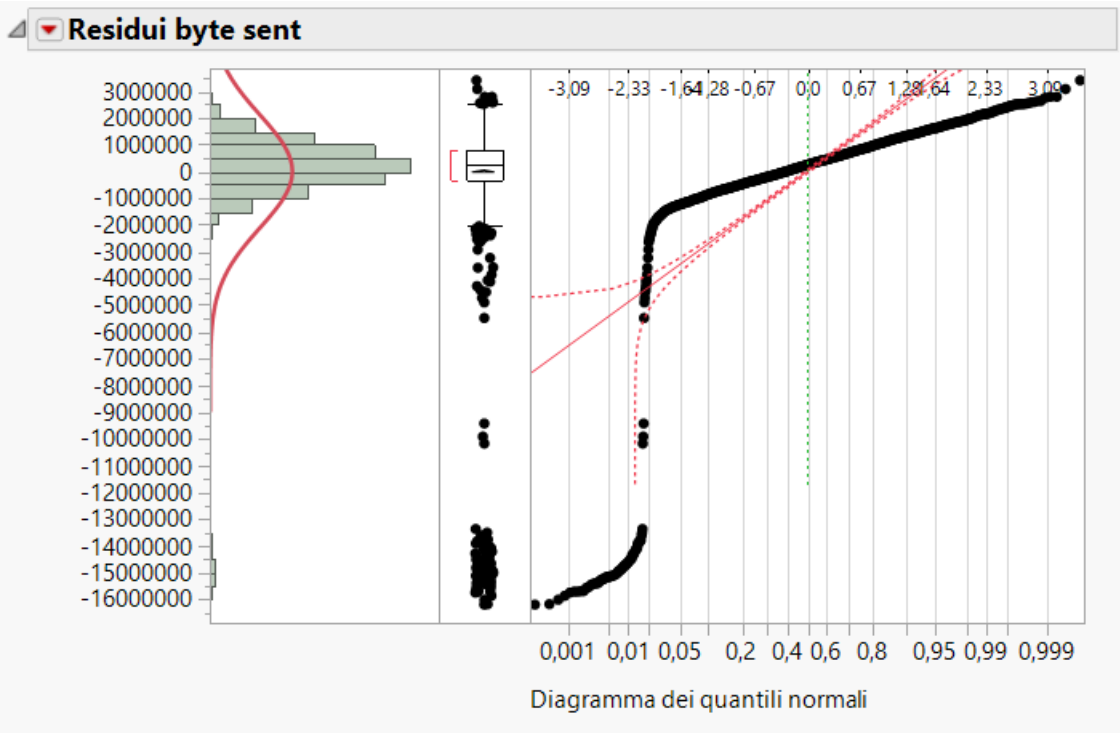
Vale anche in questo caso quanto osservato per EXP1.

Controllo della normalità

Si procede a fare un test visivo della normalità dei residui dei dati attraverso QQPlot


Residui nmail
800
700
600
500
400
300
200
100
0
-100
-200
-300
-400
-500
-600
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali


Residui byte rec
50000000
45000000
40000000
35000000
30000000
25000000
20000000
15000000
10000000
5000000
0
-5000000
-10000000
-15000000
-20000000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali



A partire da questi plot è possibile determinare che le distribuzioni **non sono quindi normali**.

Quanto osservato è ulteriormente supportato se si analizza la bontà di adattamento delle tre distribuzioni, calcolata mediante il **test di KSL** in quanto il numero di campioni della distribuzione supera i 2000 elementi:

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,295627	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

nmail

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,241107	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

byte rec

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,242236	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

byte sent

Avendo dimostrato la non-normalità dei dati, viene ritenuto inutile testare la loro omoschedasticità, per cui si procede direttamente ad effettuare il test di Mann-Kendall.

Test di Mann-Kendall

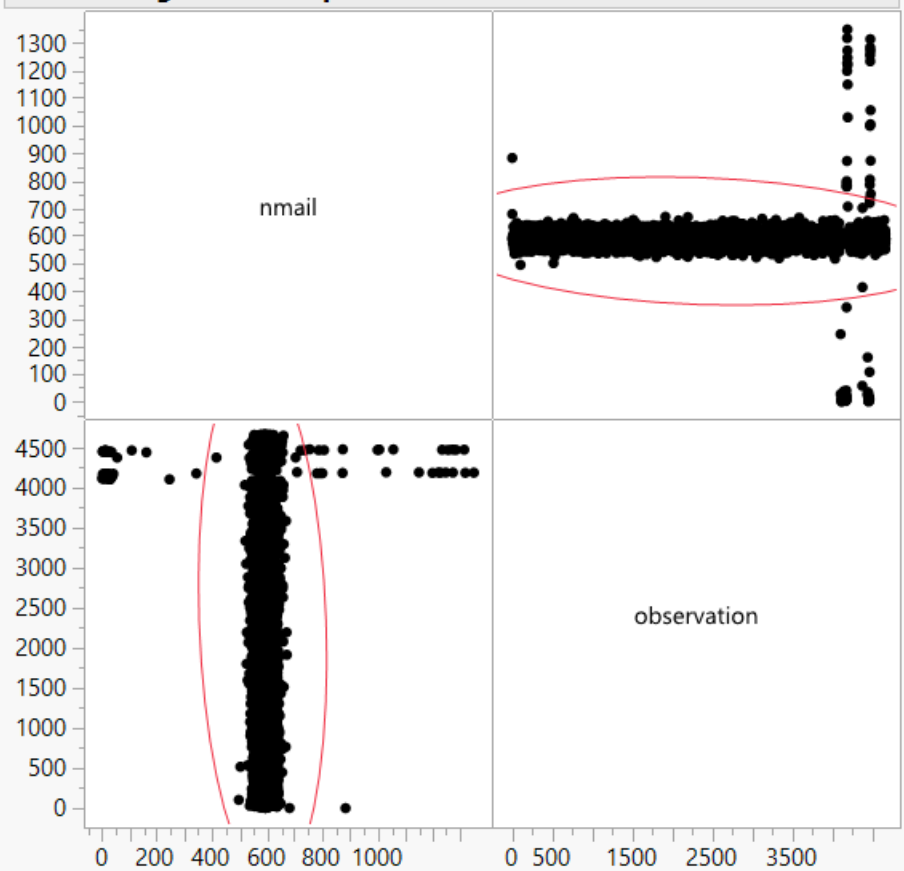
Ripetendo il test sul secondo dataset si ottengono i seguenti risultati:

Correlazioni

	nmail	observation
nmail	1,0000	-0,1415
observation	-0,1415	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

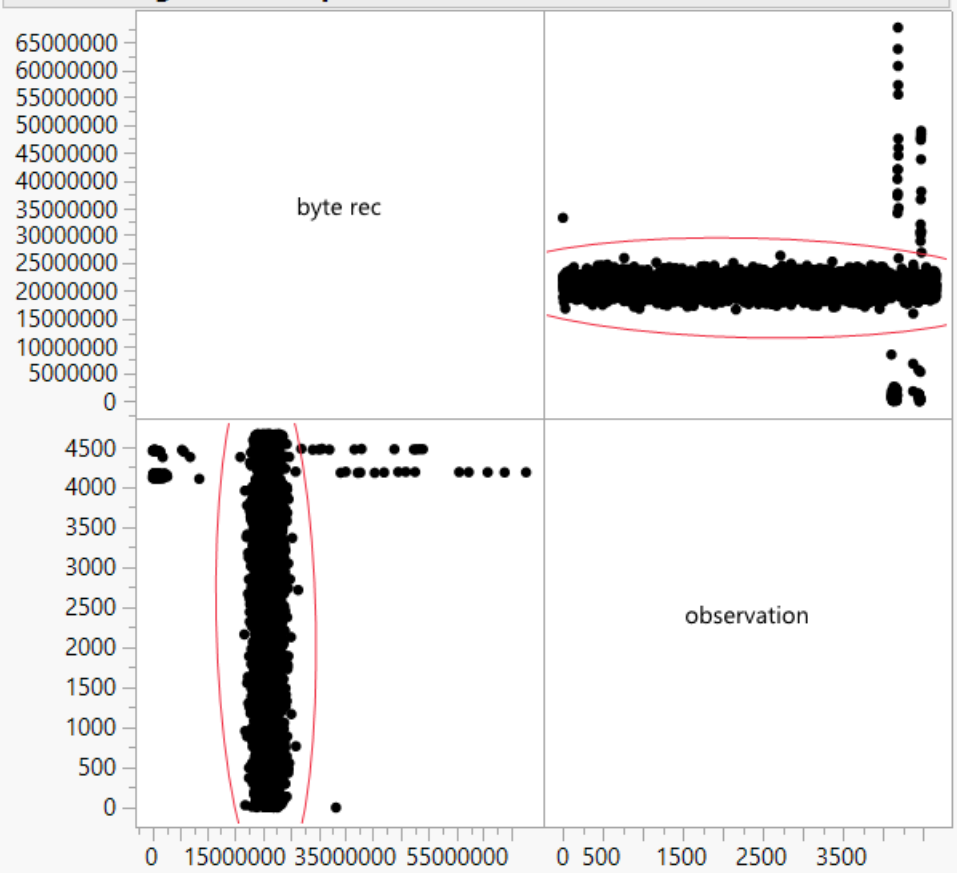
Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
observation	nmail	-0,0242	0,0138*	

Correlazioni

	byte rec	observation
byte rec	1,0000	-0,1122
observation	-0,1122	1,0000

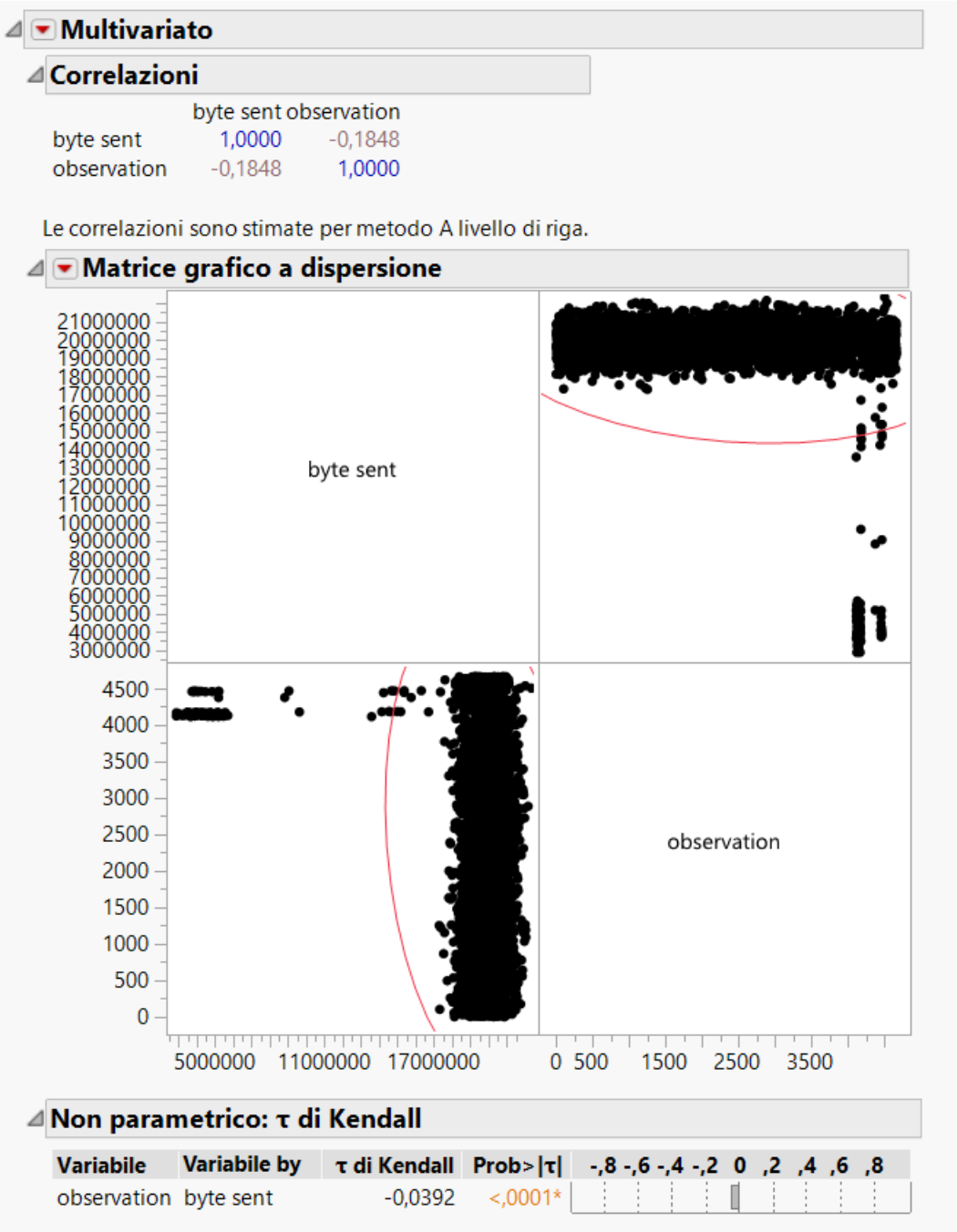
Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
observation	byte rec	-0,0316	0,0012*	



Multivariato

Correlazioni

	byte sent	observation
byte sent	1,0000	-0,1848
observation	-0,1848	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione

Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
observation	byte sent	-0,0392	<,0001*	

In questo caso è possibile osservare la presenza di trend statisticamente significativi per tutte e tre le variabili, in quanto in tutti i casi il p-value si presenta minore della soglia indicata.

Stimatore di Theil-Sen

Si applica quindi la procedura indicata precedentemente per tutte le variabili:

	nmail	byte rec	byte sent
Pendenza	-0.001	-48.294	-34.063
Intercetta	592.394	20890131.991	19903064.101
Intervallo di confidenza superiore	0.0	-18.993	-17.374

	nmail	byte rec	byte sent
Intervallo di confidenza inferiore	-0.001	-77.548	-50.673

Per *byte sent* e *byte rec* valgono le considerazioni viste per EXP 1; *nmail* è invece l'unica variabile il cui intervallo di confidenza include lo 0, per cui è possibile affermare con una probabilità del 95% che la pendenza della retta sia uguale a 0.

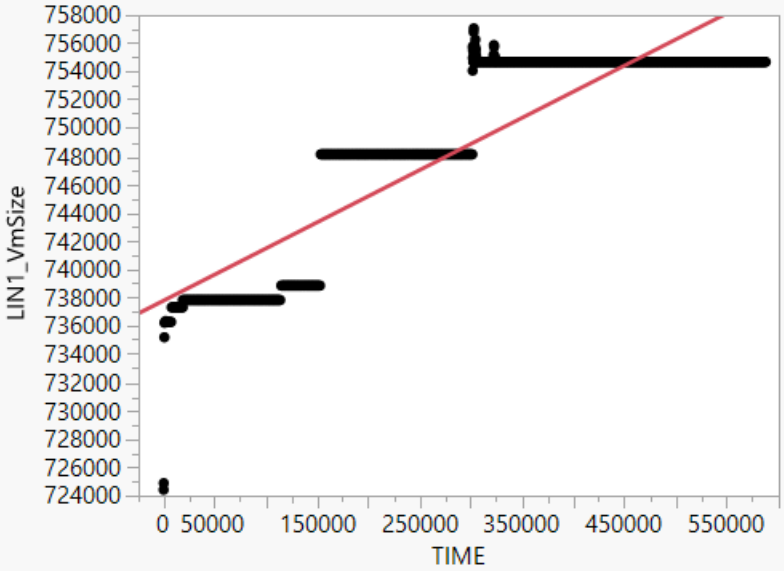
OS1

Per l'analisi di questo dataset si considerano *LIN1_VmSize, LIN1_VmData, LIN1_RSS, LIN1_byte_letti_I_O* e *LIN1_byte_letti_I_O1* come variabili di risposta, mentre *TIME* è considerata variabile di predizione.

Stima lineare

Anche in questo caso si parte dal valutare la stima lineare delle variabili:

Stima bivariata di LIN1_VmSize rispetto a TIME



Stima lineare

Stima lineare

LIN1_VmSize = 737797,88 + 0,036964*TIME

Riepilogo della stima

R-quadro	0,819413
R-quadro corretto	0,819377
Scarto quadratico medio	2950,291
Media della risposta	748678,6
Osservazioni (o somme pesate)	4907

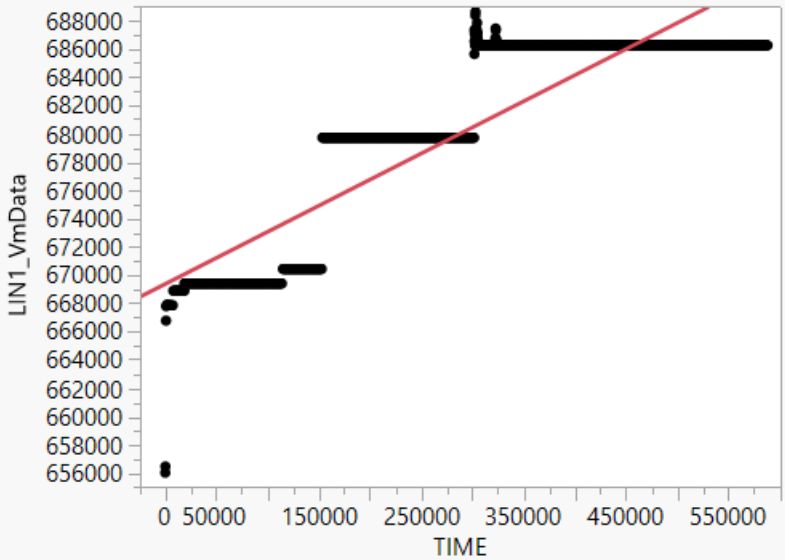
Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	1,9373e+11	1,937e+11	22256,48
Errore	4905	4,2694e+10	8704218,3	**Prob > F**
C. totale	4906	2,3642e+11		<,0001*

Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	737797,88	84,22102	8760,3	<,0001*
TIME	0,036964	0,000248	149,19	<,0001*

Stima bivariata di LIN1_VmData rispetto a TIME



Stima lineare

Stima lineare

LIN1_VmData = 669393,9 + 0,0369639*TIME

Riepilogo della stima

R-quadro	0,819416
R-quadro corretto	0,819379
Scarto quadratico medio	2950,265
Media della risposta	680274,6
Osservazioni (o somme pesate)	4907

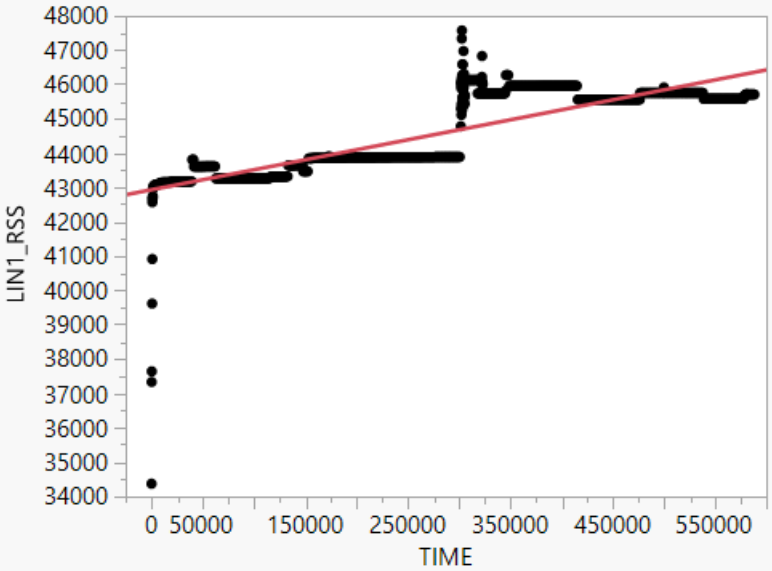
Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	1,9372e+11	1,937e+11	22256,80
Errore	4905	4,2693e+10	8704065,6	**Prob > F**
C. totale	4906	2,3642e+11		<,0001*

Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	669393,9	84,22028	7948,1	<,0001*
TIME	0,0369639	0,000248	149,19	<,0001*

Stima bivariata di LIN1_RSS rispetto a TIME



Stima lineare

Stima lineare

LIN1_RSS = 42935,798 + 0,0058185*TIME

Riepilogo della stima

R-quadro	0,755357
R-quadro corretto	0,755307
Scarto quadratico medio	562,9853
Media della risposta	44648,53
Osservazioni (o somme pesate)	4907

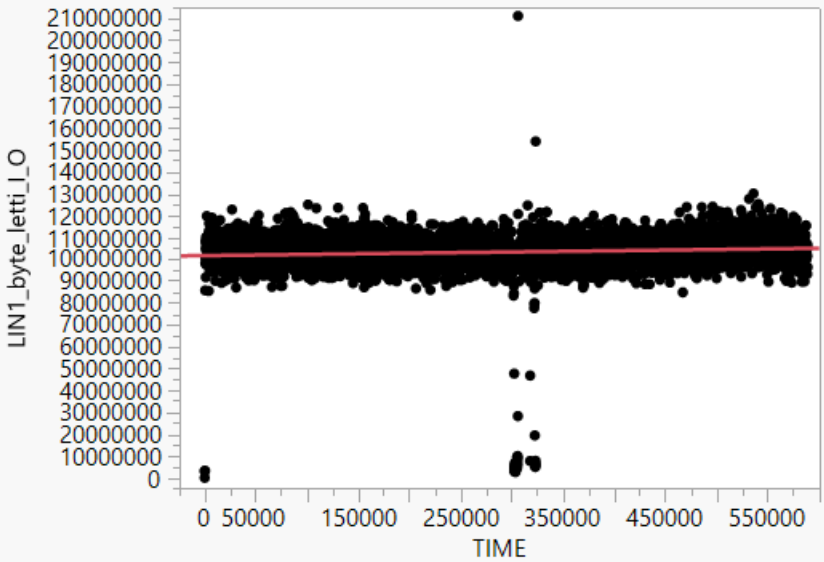
Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	4800113628	4,8001e+9	15144,59
Errore	4905	1554651546	316952,4	**Prob > F**
C. totale	4906	6354765174		<,0001*

Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	42935,798	16,07136	2671,6	<,0001*
TIME	0,0058185	4,728e-5	123,06	<,0001*

Stima bivariata di LIN1_byte_letti_I_O rispetto a TIME



Stima lineare

Stima lineare

LIN1_byte_letti_I_O = 101738601 + 5,5173708*TIME

Riepilogo della stima

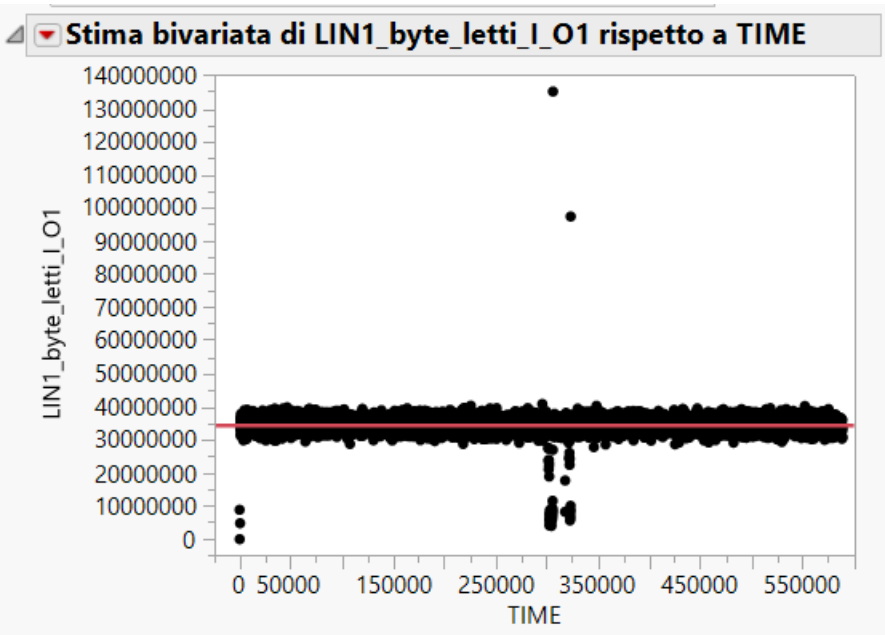
R-quadro	0,006948
R-quadro corretto	0,006746
Scarto quadratico medio	11214618
Media della risposta	1,034e+8
Osservazioni (o somme pesate)	4907

Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	4,3161e+15	4,316e+15	34,3182
Errore	4905	6,1689e+17	1,258e+14	**Prob > F**
C. totale	4906	6,2121e+17		<,0001*

Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	101738601	320140,1	317,79	<,0001*
TIME	5,5173708	0,941824	5,86	<,0001*



Stima lineare

Stima lineare

LIN1_byte_letti_I_O1 = 34245222 + 0,1574532*TIME

Riepilogo della stima

R-quadro	5,357e-5
R-quadro corretto	-0,00015
Scarto quadratico medio	3657269
Media della risposta	34291570
Osservazioni (o somme pesate)	4907

Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	3,5151e+12	3,515e+12	0,2628
Errore	4905	6,5607e+16	1,338e+13	Prob > F
C. totale	4906	6,5611e+16		0,6082

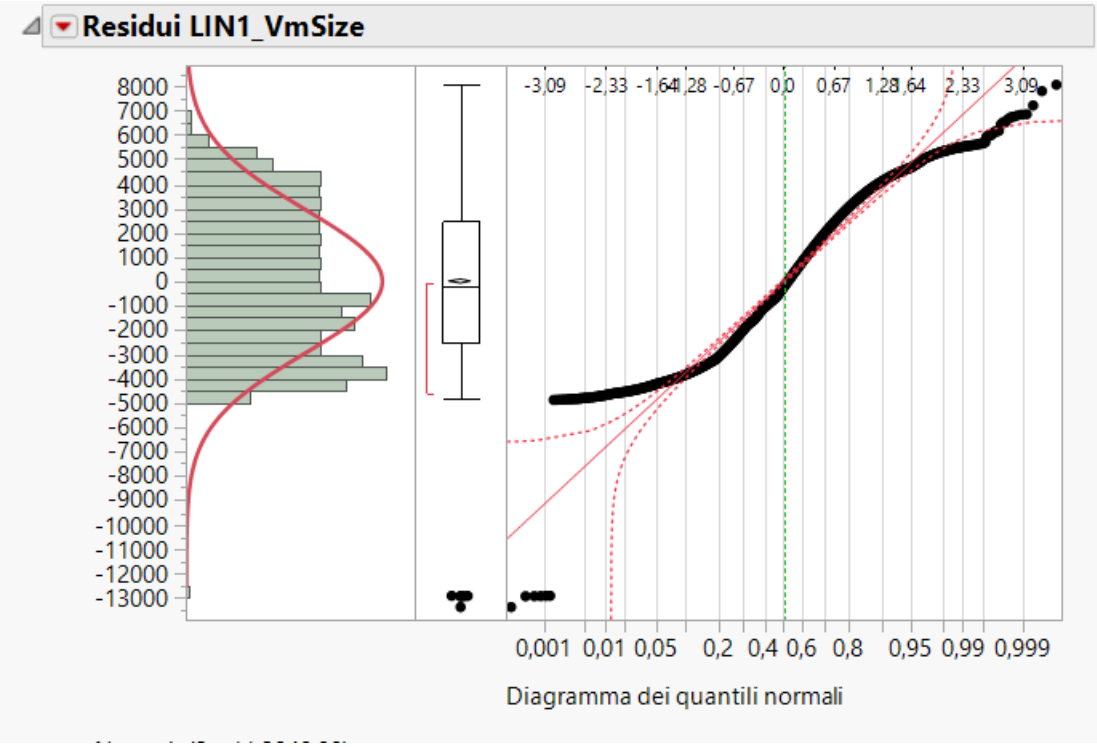
Stime dei parametri

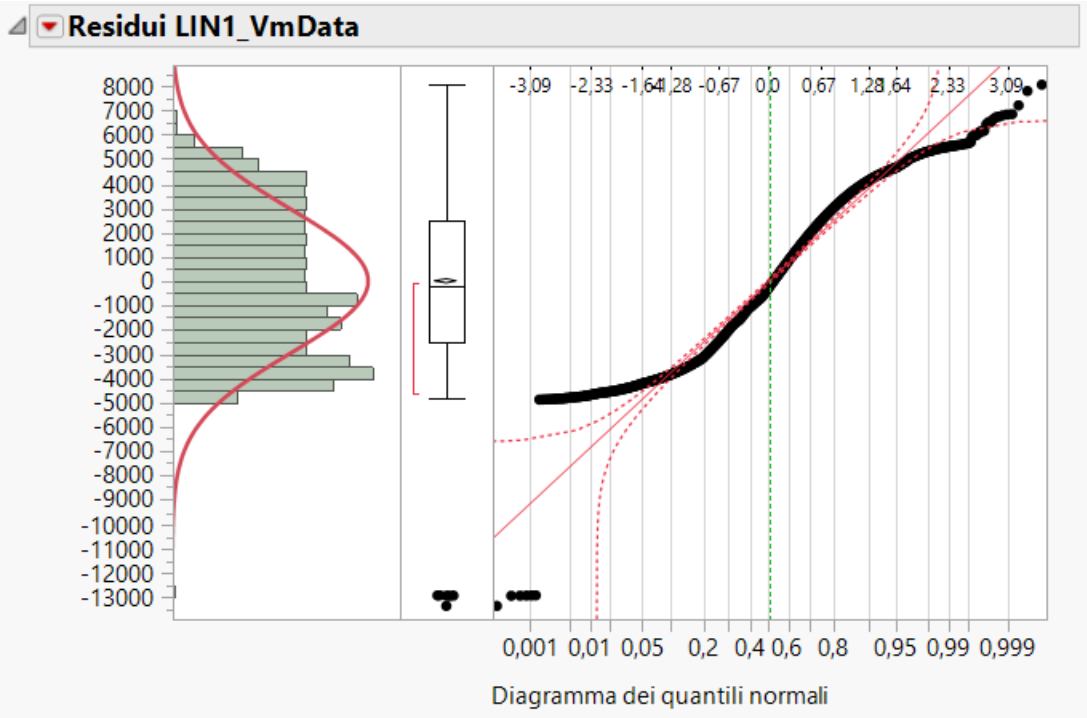
Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	34245222	104402,9	328,01	<,0001*
TIME	0,1574532	0,307144	0,51	0,6082

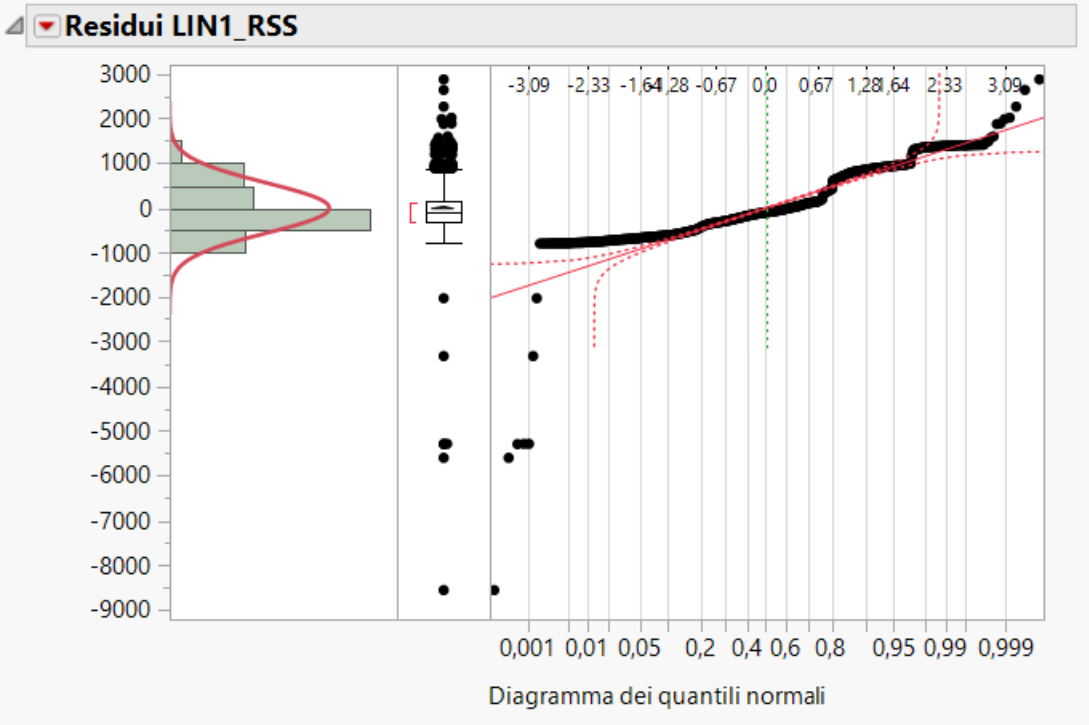
Si osserva quindi che *LIN1_VmSize, LIN1_VmData, LIN1_RSS* hanno un R^2 piuttosto alto ($0.75 < R^2 < 0.82$, per cui *TIME* spiega almeno il 75% della variazione su queste variabili), mentre *LIN1_byte_letti_I_O* e *LIN1_byte_letti_I_O1* presentano valori prossimi allo 0.

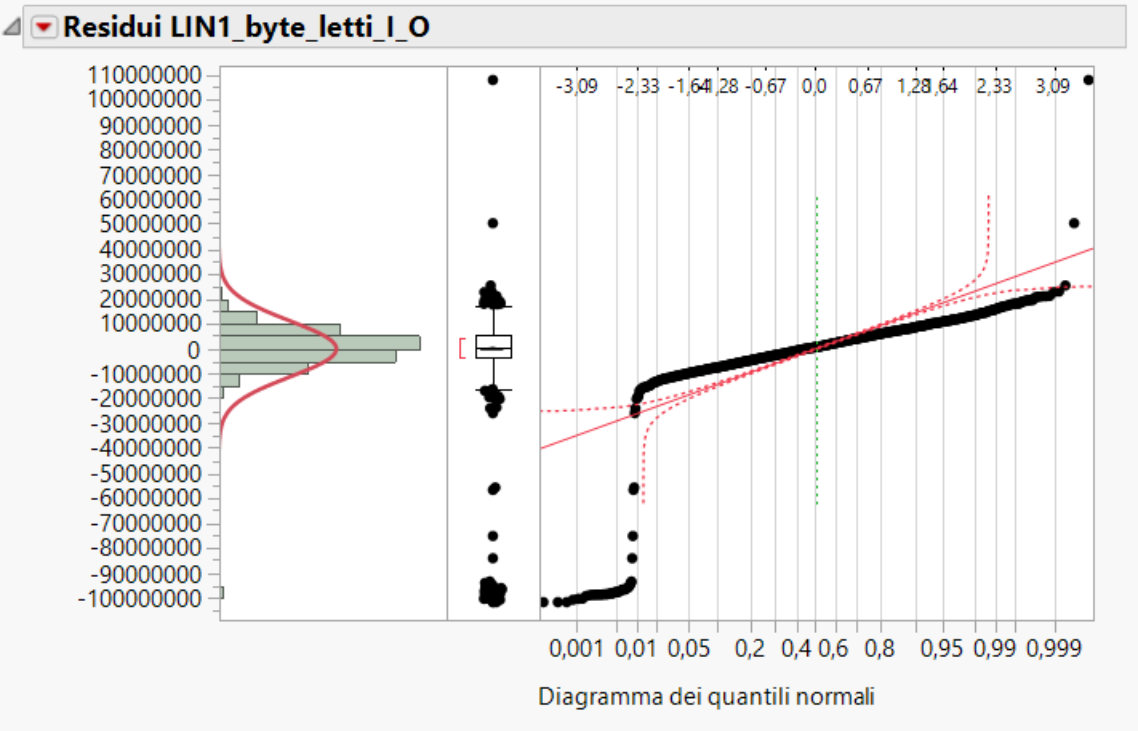
Controllo della normalità

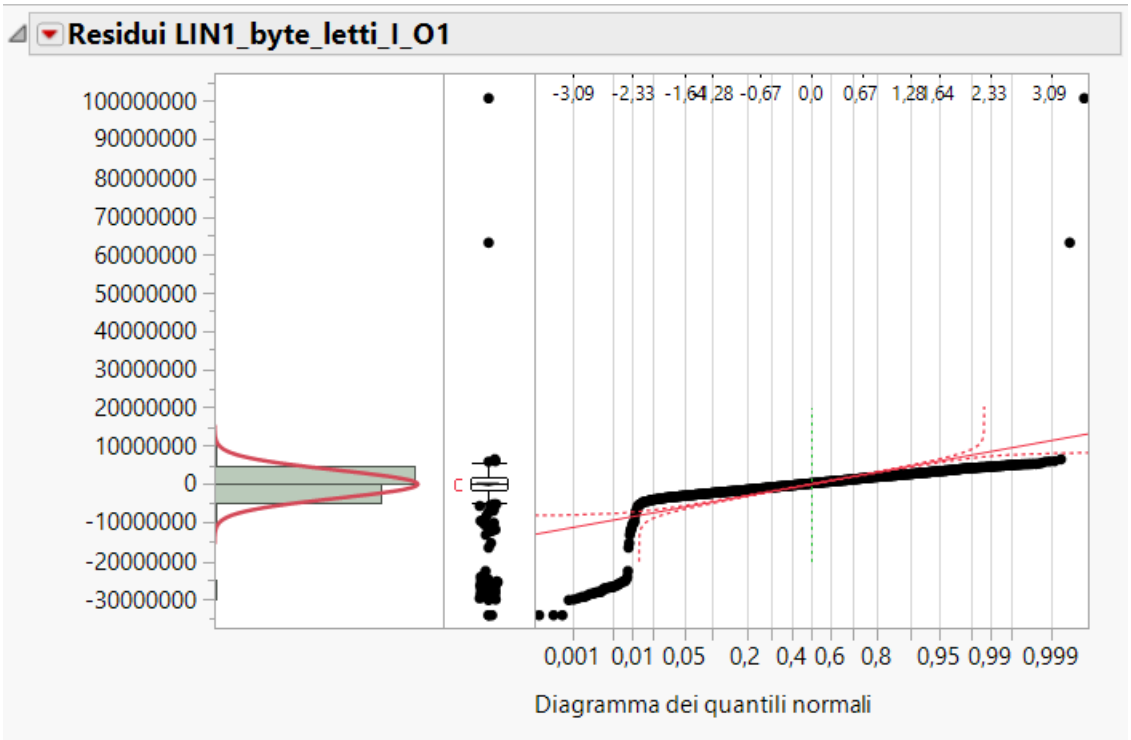
Si procede quindi a fare un test visivo della normalità dei residui dei dati attraverso QQPlot, necessario per la scelta del test di regressione da eseguire in seguito.


Residui LIN1_VmSize
8000
7000
6000
5000
4000
3000
2000
1000
0
-1000
-2000
-3000
-4000
-5000
-6000
-7000
-8000
-9000
-10000
-11000
-12000
-13000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali


Residui LIN1_VmData
8000
7000
6000
5000
4000
3000
2000
1000
0
-1000
-2000
-3000
-4000
-5000
-6000
-7000
-8000
-9000
-10000
-11000
-12000
-13000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali


Residui LIN1_RSS
3000
2000
1000
0
-1000
-2000
-3000
-4000
-5000
-6000
-7000
-8000
-9000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali


Residui LIN1_byte_letti_I_O
110000000
100000000
90000000
80000000
70000000
60000000
50000000
40000000
30000000
20000000
10000000
0
-10000000
-20000000
-30000000
-40000000
-50000000
-60000000
-70000000
-80000000
-90000000
-100000000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali



Già dal test visivo è possibile affermare che **nessuna delle variabili ha distribuzione normale**, risultato supportato anche dal test di KSL:

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,056120	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN1_VmSize

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,056122	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN1_VmData

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,143244	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN1_RSS

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,155254	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN1_byte_letti_I_O

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,165616	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN1_byte_letti_I_O1

Test di Mann-Kendall

Avendo dimostrato la non-normalità dei dati, viene ritenuto inutile testare la loro omoschedasticità, per cui si procede direttamente ad effettuare il test di Mann-Kendall in modo da determinare se la serie temporale ha un trend monotonamente crescente o decrescente.

Si presentano dunque i risultati ottenuti a seguito del test:

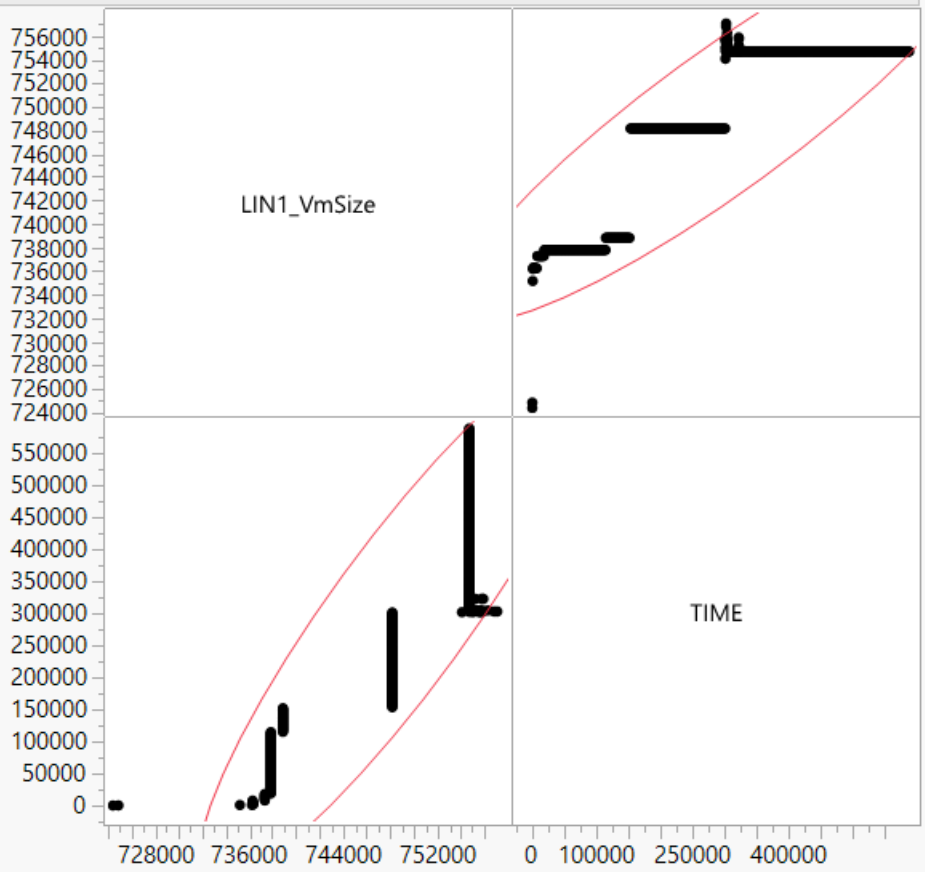
Multivariato

Correlazioni

	LIN1_VmSize	TIME
LIN1_VmSize	1,0000	0,9052
TIME	0,9052	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN1_VmSize	0,8076	<,0001*	

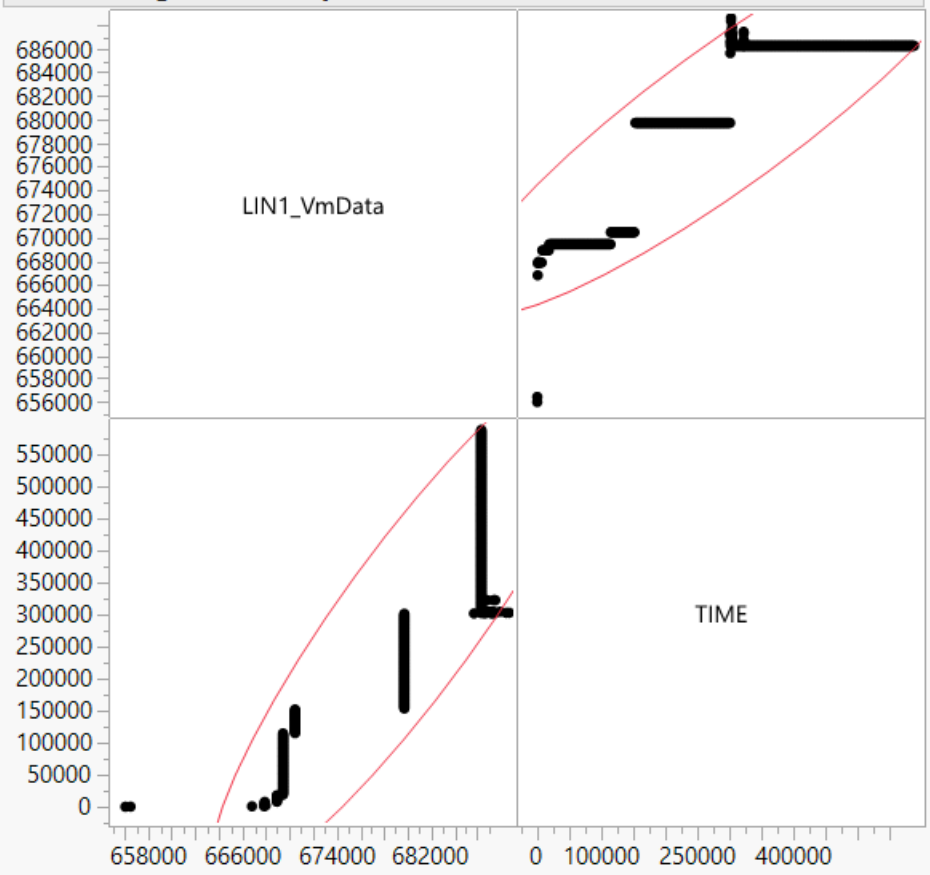
Multivariato

Correlazioni

	LIN1_VmData	TIME
LIN1_VmData	1,0000	0,9052
TIME	0,9052	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN1_VmData	0,8076	<,0001*	

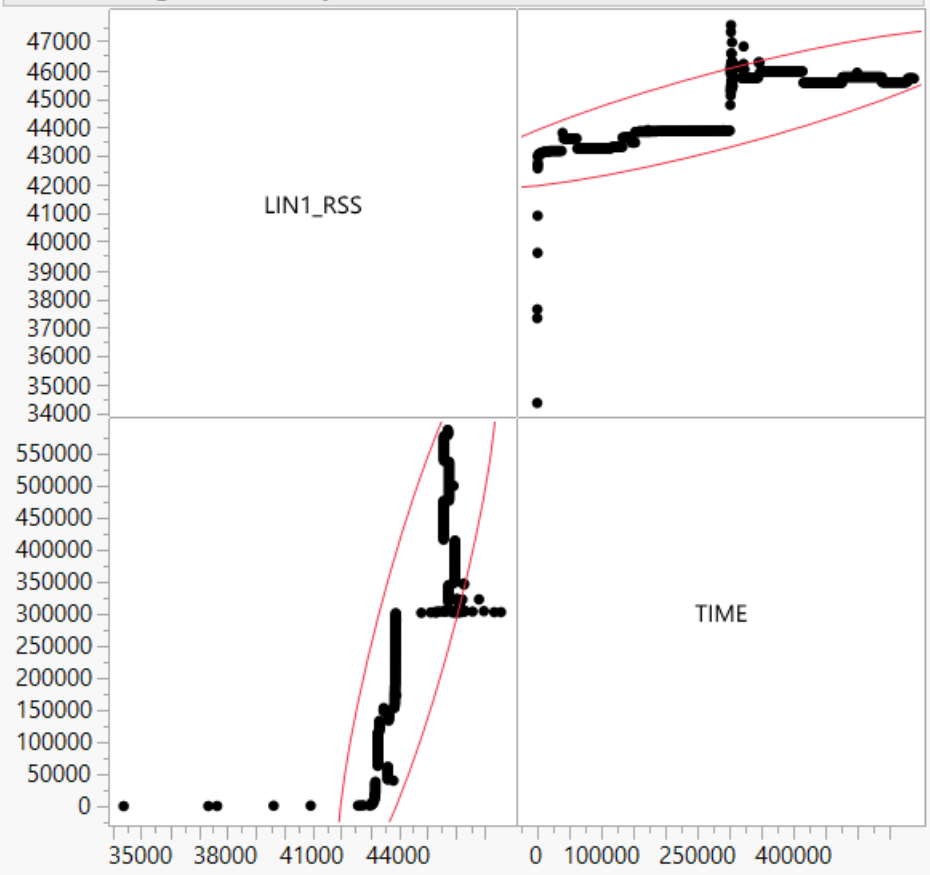
Multivariato

Correlazioni

	LIN1_RSS	TIME
LIN1_RSS	1,0000	0,8691
TIME	0,8691	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN1_RSS	0,6827	<,0001*	

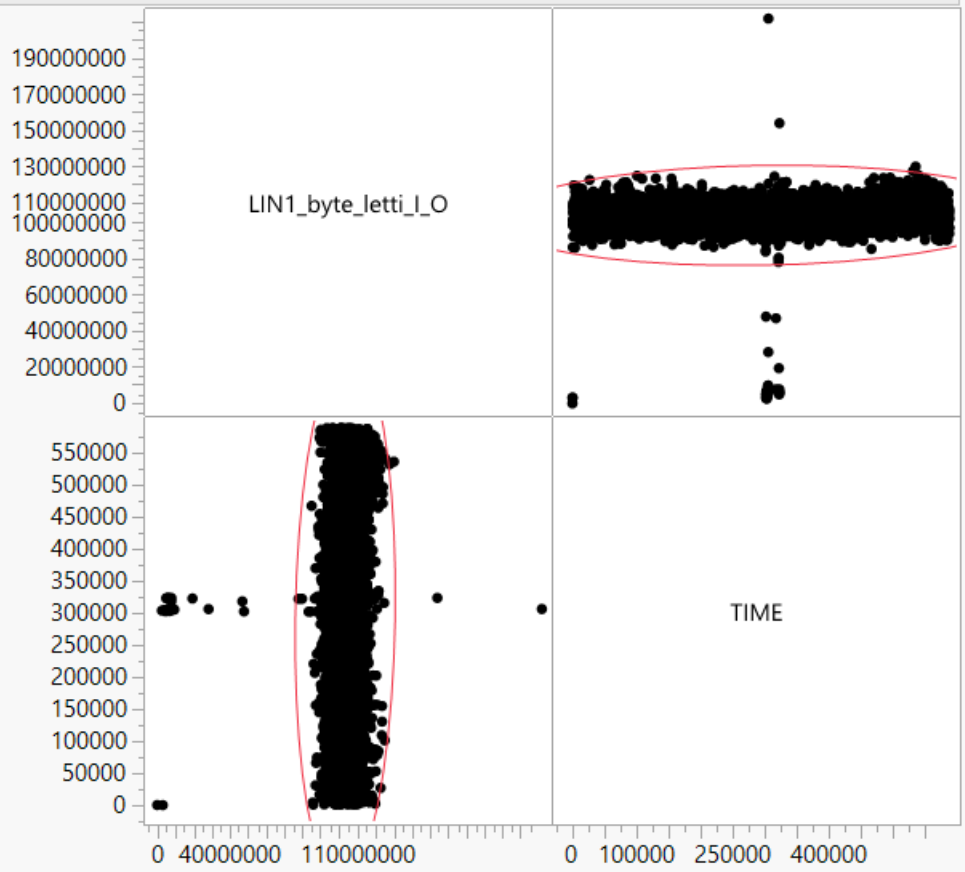
Multivariato

Correlazioni

	LIN1_byte_letti_I_O	TIME
LIN1_byte_letti_I_O	1,0000	0,0834
TIME	0,0834	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN1_byte_letti_I_O	0,0823	<,0001*	

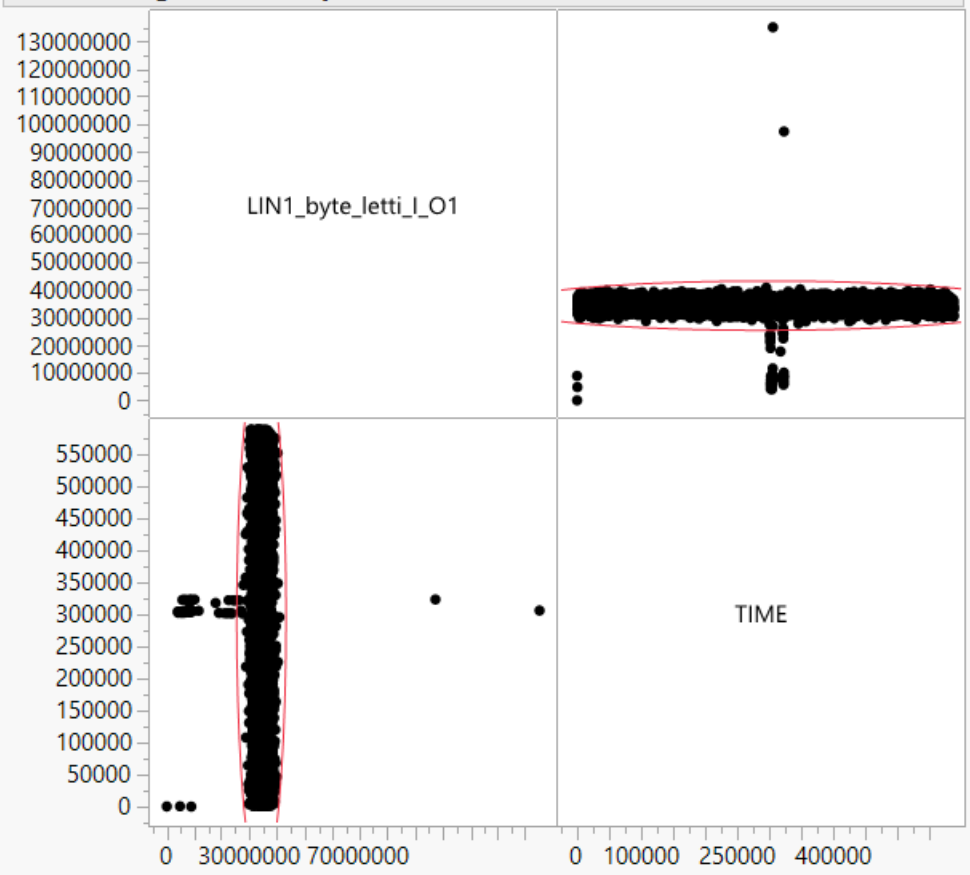
Multivariato

Correlazioni

	LIN1_byte_letti_I_O1	TIME
LIN1_byte_letti_I_O1	1,0000	0,0073
TIME	0,0073	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN1_byte_letti_I_O1	-0,0017	0,8555	

In questo caso è possibile osservare la presenza di trend statisticamente significativi per le prime 4 variabili (*LIN1_VmSize, LIN1_VmData, LIN1_RSS* e *LIN1_byte_letti_I_O*), in quanto il p-value si presenta minore della soglia indicata, ma non per la quinta (*LIN1_byte_letti_I_O1*).

Stimatore di Theil-Sen

Si applica quindi la procedura indicata precedentemente per le prime quattro variabili:

	LIN1_VmSize	LIN1_VmData	LIN1_RSS	LIN1_byte_letti_I_O
Pendenza	0.032	0.031	0.005	4.738

	LIN1_VmSize	LIN1_VmData	LIN1_RSS	LIN1_byte_letti_I_O
Intercetta	738854.395	670450.395	42428.807	102763229.503
Intervallo di confidenza superiore	0.032	0.032	0.005	5.805
Intervallo di confidenza inferiore	0.031	0.031	0.005	3.674

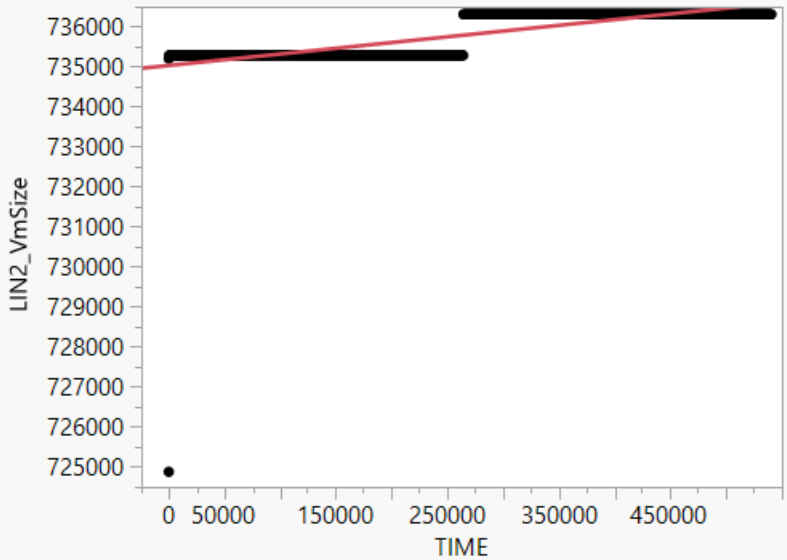
Nessuna delle variabili ha intervallo di confidenza che include lo 0, per cui è possibile affermare con una probabilità del 95% che le pendenze delle rette siano diverse da 0.

OS2

Per l'analisi di questo dataset si considerano *LIN2_VmSize, LIN2_VmData, LIN2_RSS, LIN2_byte_letti_sec* e *LIN2_byte_scritti_sec* come variabili di risposta, mentre *TIME* è considerata variabile di predizione.

Stima lineare

Stima bivariata di LIN2_VmSize rispetto a TIME



Stima lineare

Stima lineare

LIN2_VmSize = 735030,96 + 0,0028618*TIME

Riepilogo della stima

R-quadro	0,693318
R-quadro corretto	0,69325
Scarto quadratico medio	297,4945
Media della risposta	735805,4
Osservazioni (o somme pesate)	4511

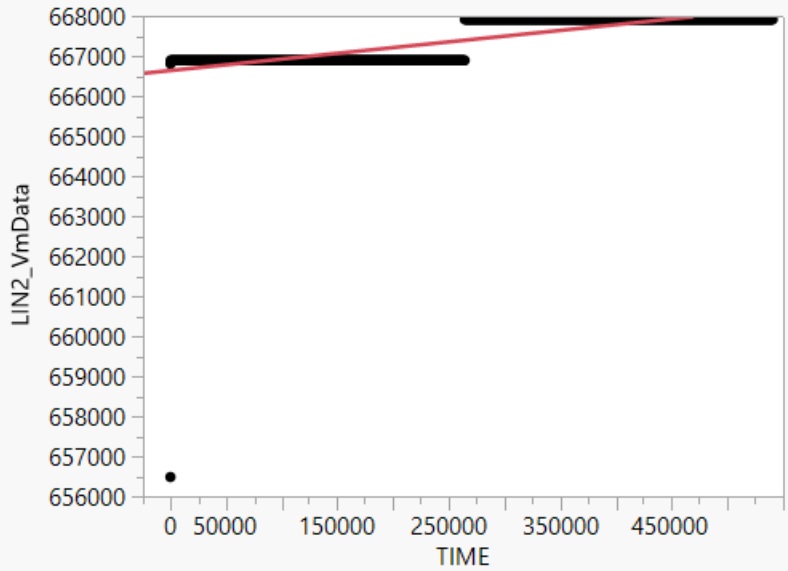
Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	902157946	902157946	10193,53
Errore	4509	399059871	88502,965	**Prob > F**
C. totale	4510	1301217817		<,0001*

Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	735030,96	8,857278	82986	<,0001*
TIME	0,0028618	2,835e-5	100,96	<,0001*

Stima bivariata di LIN2_VmData rispetto a TIME



Stima lineare

Stima lineare

LIN2_VmData = 666654,96 + 0,0028618*TIME

Riepilogo della stima

R-quadro	0,693318
R-quadro corretto	0,69325
Scarto quadratico medio	297,4945
Media della risposta	667429,4
Osservazioni (o somme pesate)	4511

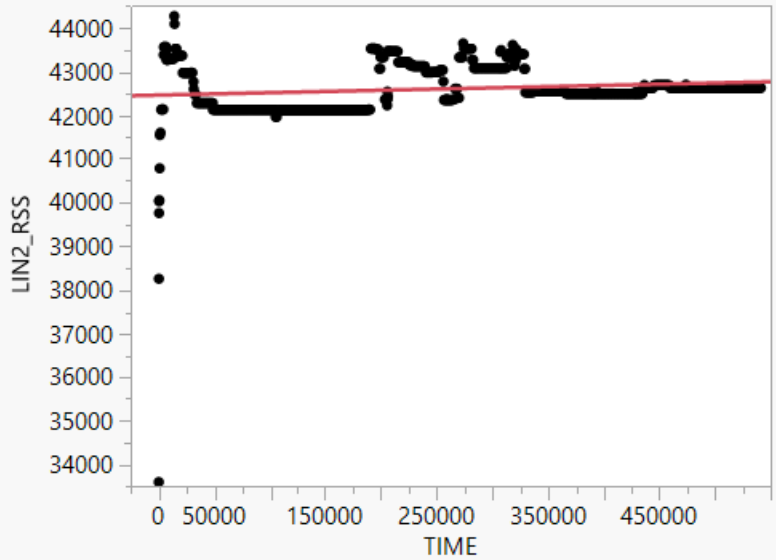
Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	902157946	902157946	10193,53
Errore	4509	399059871	88502,965	**Prob > F**
C. totale	4510	1301217817		<,0001*

Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	666654,96	8,857278	75266	<,0001*
TIME	0,0028618	2,835e-5	100,96	<,0001*

Stima bivariata di LIN2_RSS rispetto a TIME



Stima lineare

Stima lineare

LIN2_RSS = 42463,363 + 0,0005531*TIME

Riepilogo della stima

R-quadro	0,033845
R-quadro corretto	0,033631
Scarto quadratico medio	461,8856
Media della risposta	42613,03
Osservazioni (o somme pesate)	4511

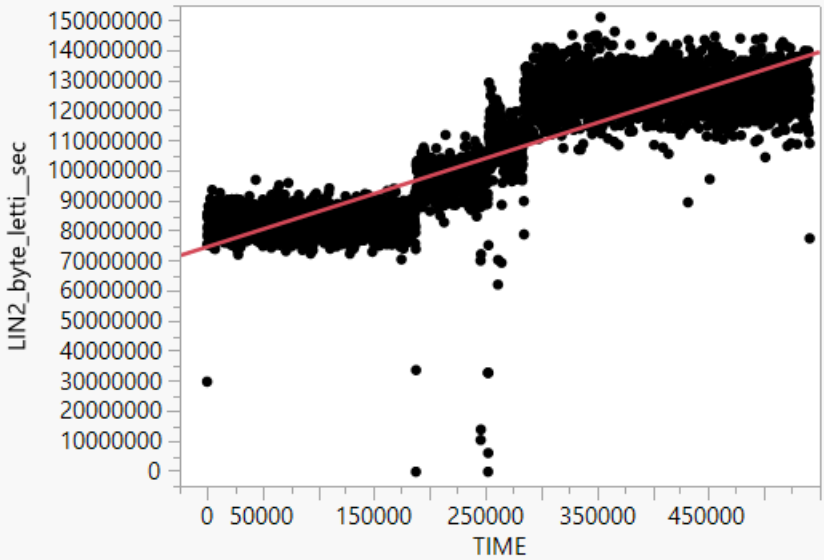
Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	33697546	33697546	157,9536
Errore	4509	961942305	213338,28	**Prob > F**
C. totale	4510	995639851		<,0001*

Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	42463,363	13,75168	3087,9	<,0001*
TIME	0,0005531	0,000044	12,57	<,0001*

Stima bivariata di LIN2_byte_letti__sec rispetto a TIME



Stima lineare

Stima lineare

LIN2_byte_letti__sec = 74822011 + 117,96378*TIME

Riepilogo della stima

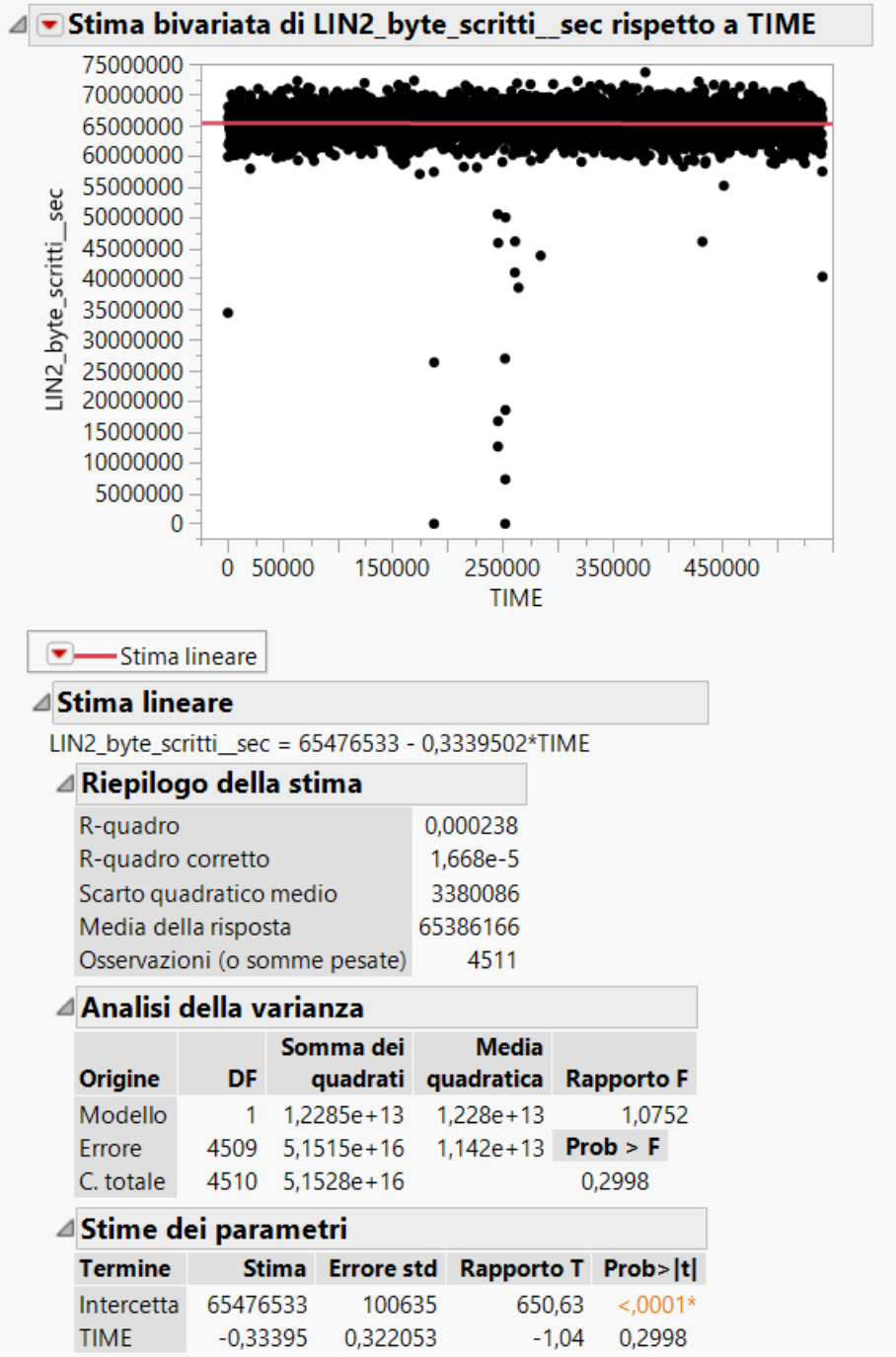
R-quadro	0,760394
R-quadro corretto	0,760341
Scarto quadratico medio	10349942
Media della risposta	1,067e+8
Osservazioni (o somme pesate)	4511

Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	1,5328e+18	1,533e+18	14309,39
Errore	4509	4,8301e+17	1,071e+14	**Prob > F**
C. totale	4510	2,0159e+18		<,0001*

Stime dei parametri

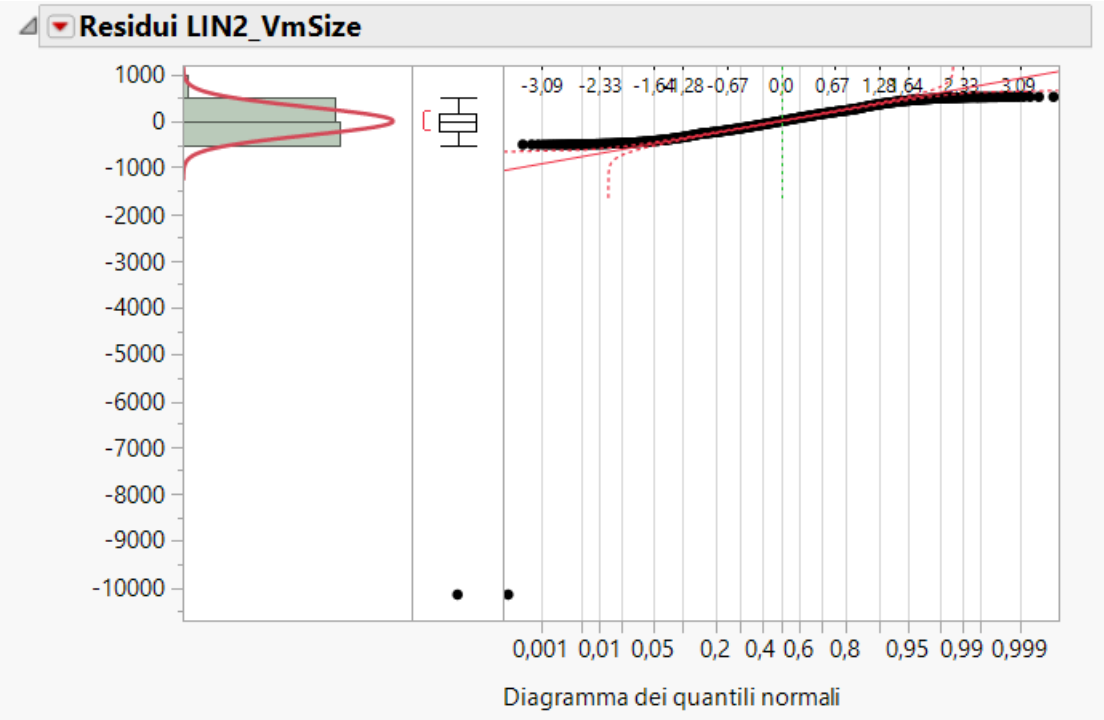
Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	74822011	308148	242,81	<,0001*
TIME	117,96378	0,986139	119,62	<,0001*

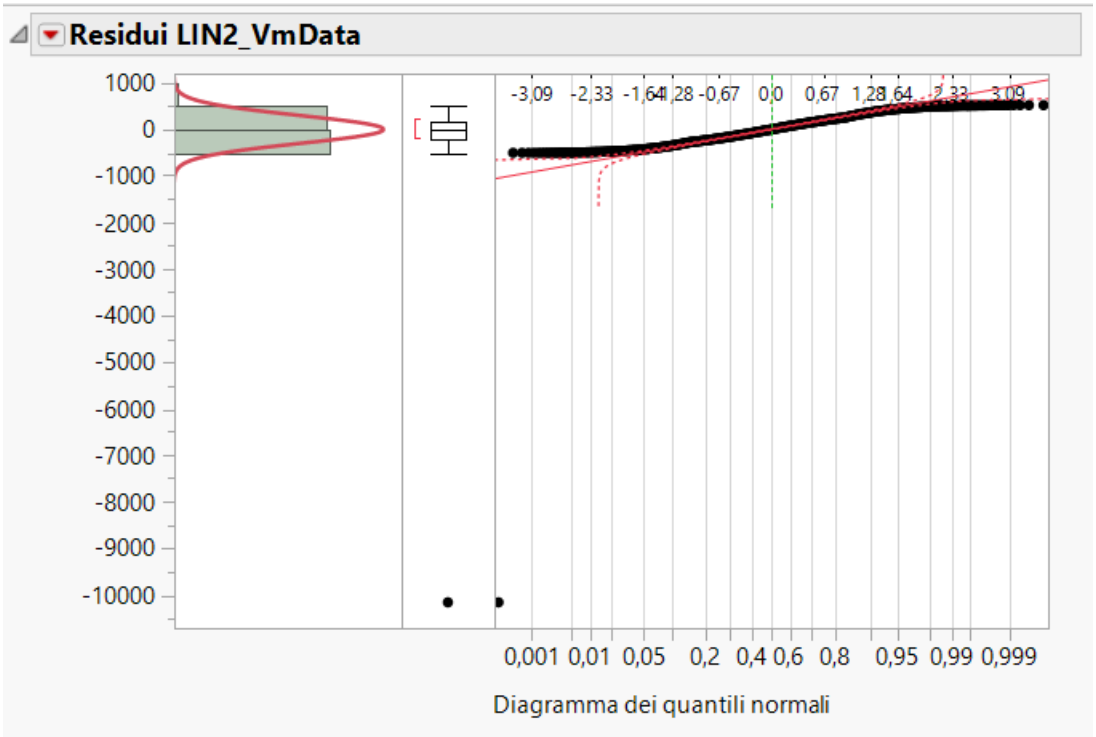


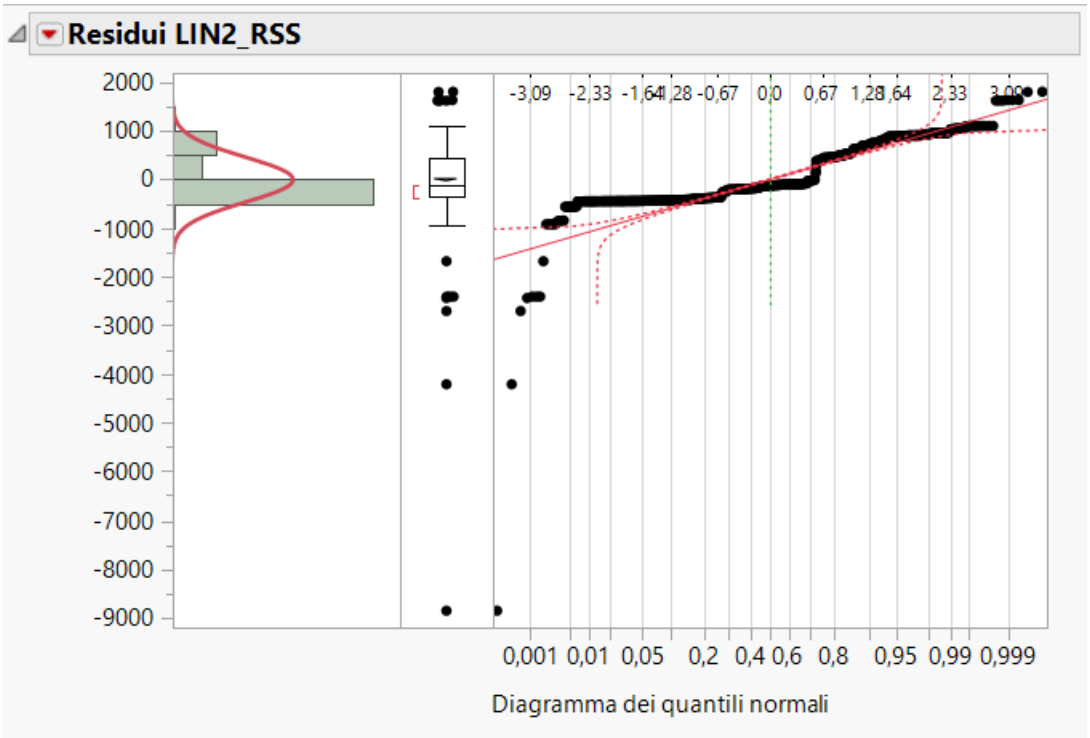
Si osserva quindi che *LIN2_VmSize, LIN2_VmData, LIN2_byte_letti_sec* hanno un R^2 piuttosto alto ($0.69 < R^2 < 0.77$, per cui *TIME* spiega almeno il 69% della variazione su queste variabili), mentre *LIN2_RSS* e *LIN2_byte_scritti_sec* presentano valori prossimi allo 0.

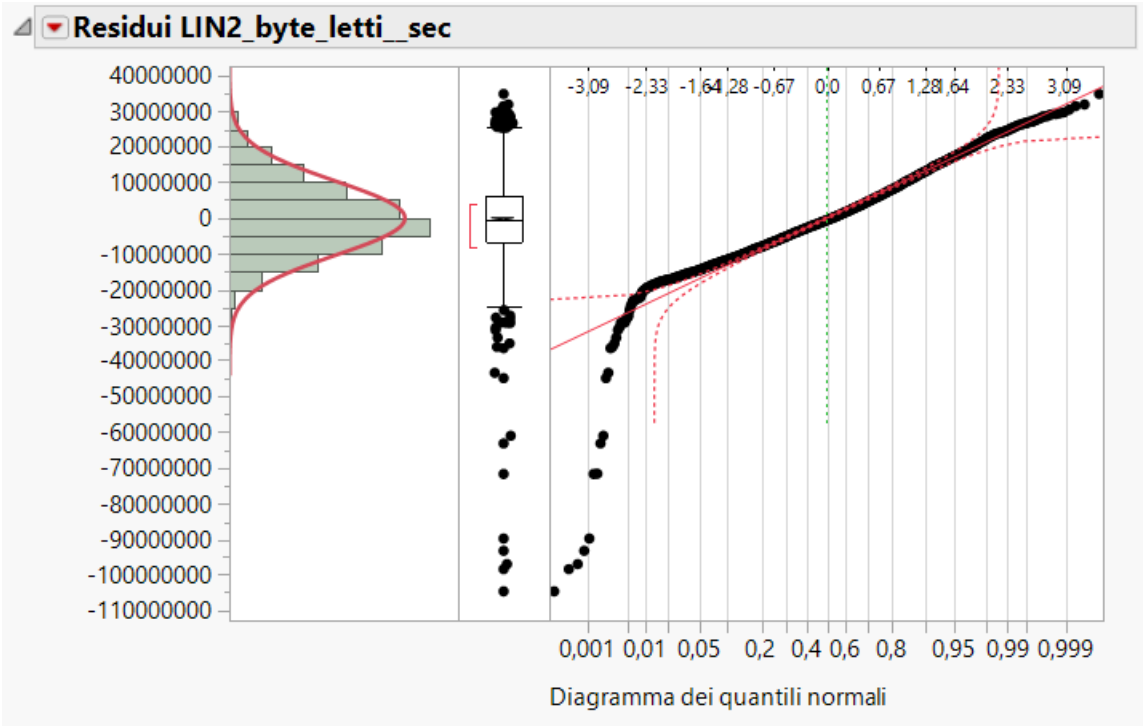
Controllo della normalità

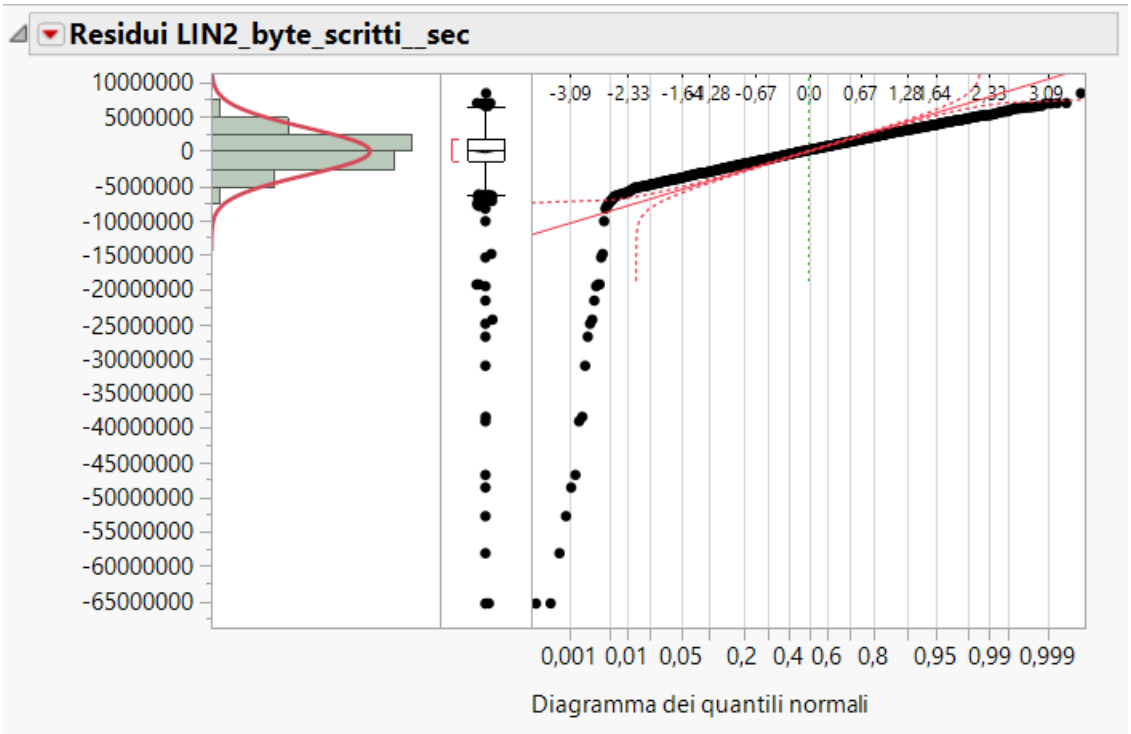
Si procede quindi a fare un test visivo della normalità dei residui dei dati attraverso QQPlot


Residui LIN2_VmSize
1000
0
-1000
-2000
-3000
-4000
-5000
-6000
-7000
-8000
-9000
-10000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali


Residui LIN2_VmData
1000
0
-1000
-2000
-3000
-4000
-5000
-6000
-7000
-8000
-9000
-10000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali


Residui LIN2_RSS
2000
1000
0
-1000
-2000
-3000
-4000
-5000
-6000
-7000
-8000
-9000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali


Residui LIN2_byte_letti__sec
40000000
30000000
20000000
10000000
0
-10000000
-20000000
-30000000
-40000000
-50000000
-60000000
-70000000
-80000000
-90000000
-100000000
-110000000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali



Già dal test visivo è possibile affermare che **nessuna delle variabili ha distribuzione normale**, risultato supportato anche dal test di KSL:

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,045035	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN2_VmSize

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,045035	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN2_VmData

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,251864	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN2_RSS

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,034420	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN2_byte_letti_sec

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,098410	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN2_byte_scritti_sec

Test di Mann-Kendall

Avendo dimostrato la non-normalità dei dati, viene ritenuto inutile testare la loro omoschedasticità, per cui si procede direttamente ad effettuare il test di Mann-Kendall in modo da determinare se la serie temporale ha un trend monotonamente crescente o decrescente.

Si presentano dunque i risultati ottenuti a seguito del test:

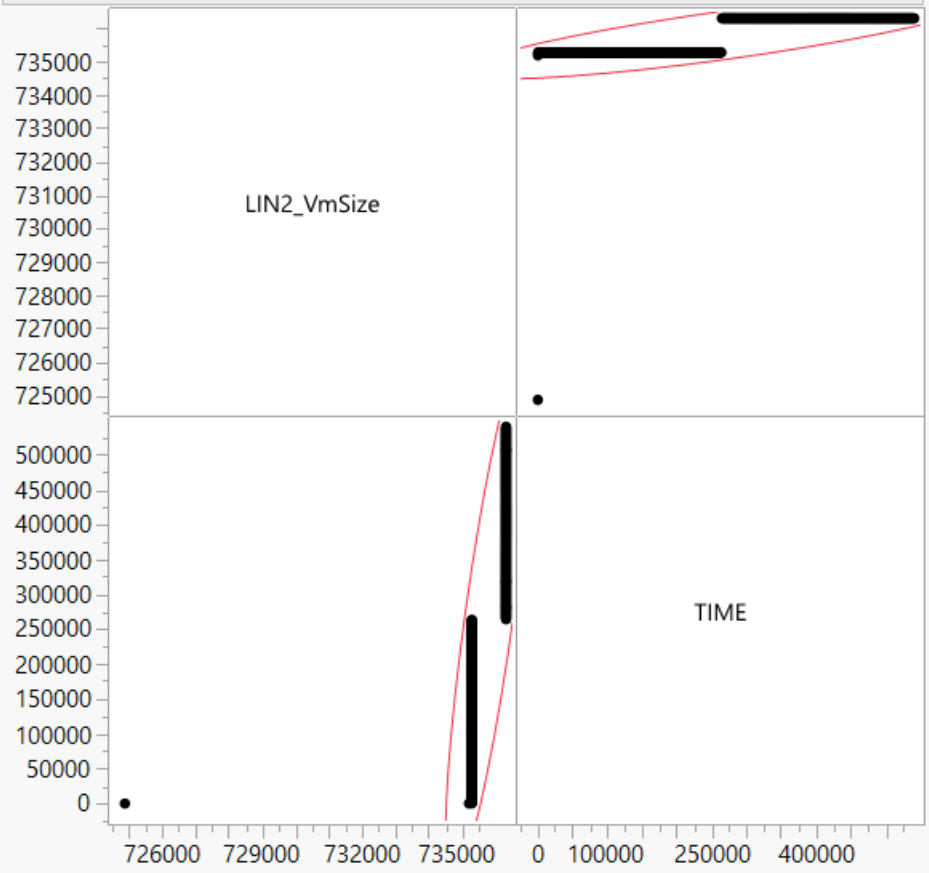
Multivariato

Correlazioni

	LIN2_VmSize	TIME
LIN2_VmSize	1,0000	0,8327
TIME	0,8327	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN2_VmSize	0,7065	<,0001*	

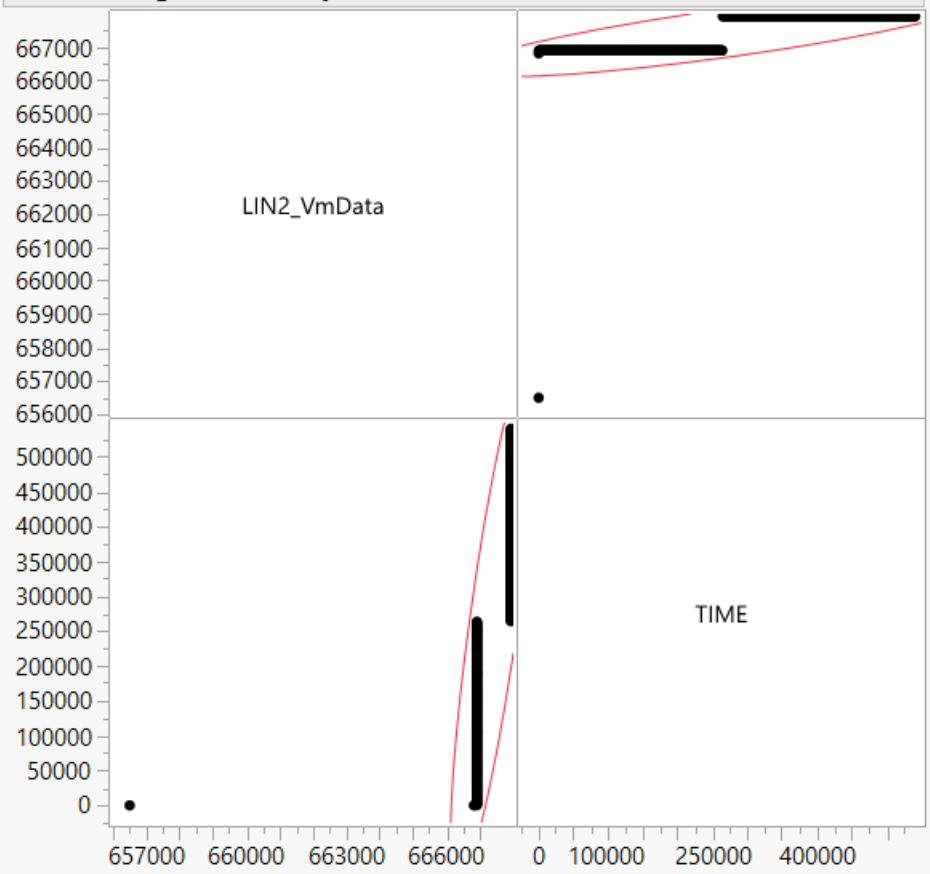
Multivariato

Correlazioni

	LIN2_VmData	TIME
LIN2_VmData	1,0000	0,8327
TIME	0,8327	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN2_VmData	0,7065	<,0001*	

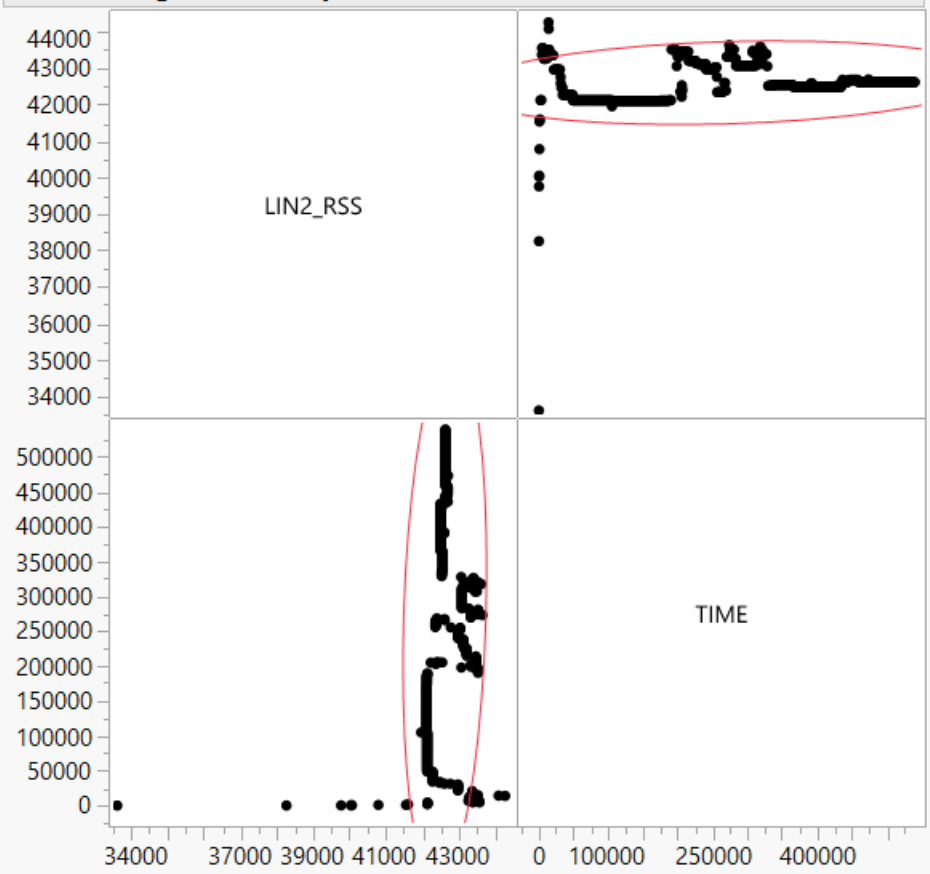
Multivariato

Correlazioni

	LIN2_RSS	TIME
LIN2_RSS	1,0000	0,1840
TIME	0,1840	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN2_RSS	0,1788	<,0001*	

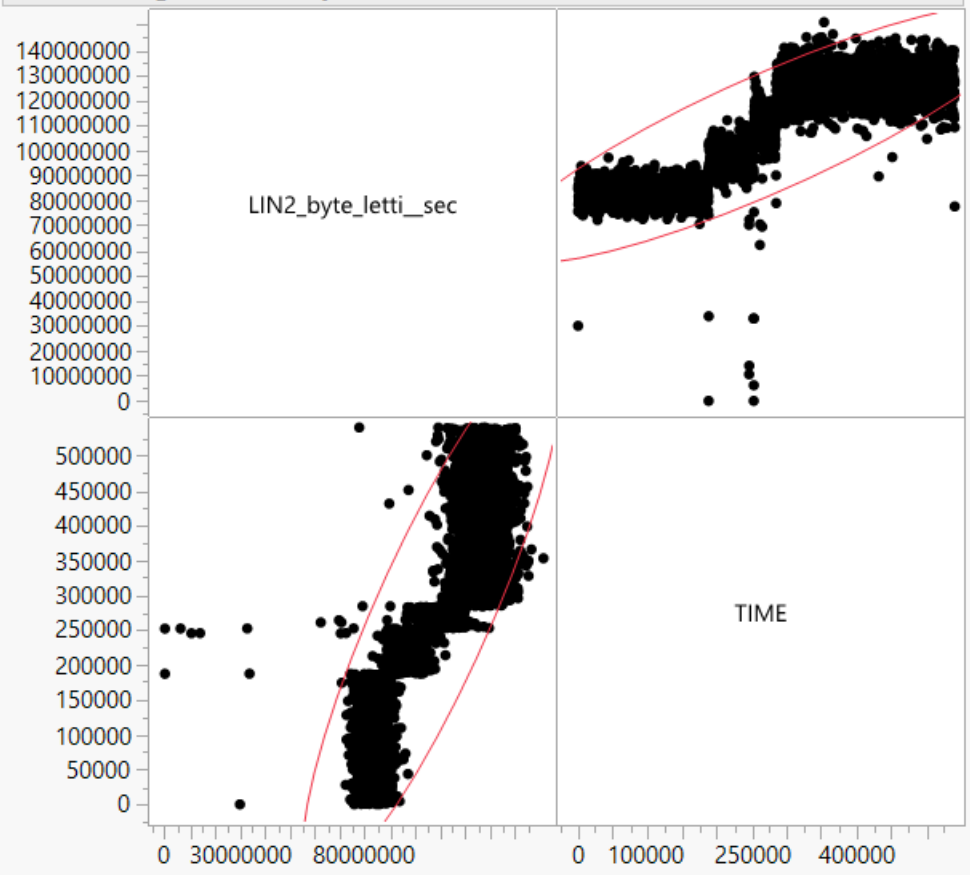
Multivariato

Correlazioni

	LIN2_byte_letti__sec	TIME
LIN2_byte_letti__sec	1,0000	0,8720
TIME	0,8720	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN2_byte_letti__sec	0,6259	<,0001*	

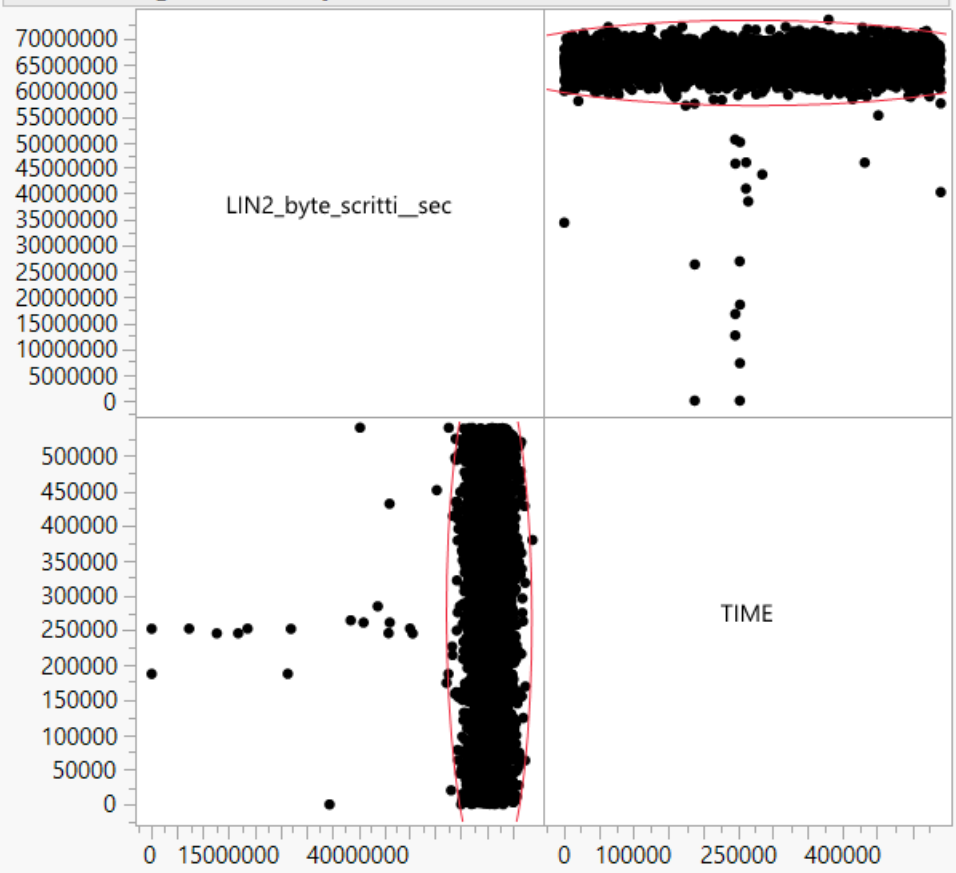
Multivariato

Correlazioni

	LIN2_byte_scritti__sec	TIME
LIN2_byte_scritti__sec	1,0000	-0,0154
TIME	-0,0154	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN2_byte_scritti__sec	-0,0210	0,0342*	

In questo caso i p-value indicano che tutte le variabili hanno trend statisticamente significativi.

Stimatore di Theil-Sen

Si applica quindi la procedura indicata precedentemente a tutte le variabili:

	LIN2_VmSize	LIN2_VmData	LIN2_RSS	LIN2_byte_letti_sec	LIN2_byte_scritti_sec
Pendenza	0.002	0.002	0.001	114.554	-0.489
Intercetta	735790.376	667414.376	42384.855	77733384.238	65693615.576

	LIN2_VmSize	LIN2_VmData	LIN2_RSS	LIN2_byte_letti_sec	LIN2_byte_scritti_sec
Intervallo di confidenza superiore	0.002	0.002	0.001	116.377	-0.036
Intervallo di confidenza inferiore	0.0	0.0	0.001	112.734	-0.939

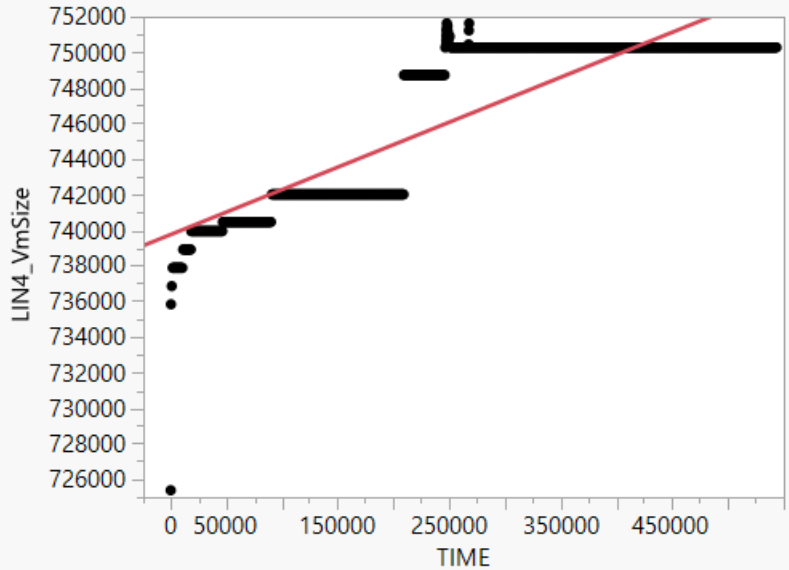
Si osserva che le variabili *LIN2_RSS, LIN2_byte_letti_sec* e *LIN2_byte_scritti_sec* hanno intervallo di confidenza che esclude lo 0, per cui è possibile affermare con una probabilità del 95% che le pendenze delle rette siano diverse da 0; mentre per le variabili *LIN2_VmSize* e *LIN2_VmData* è possibile affermare il contrario.

OS3

Per l'analisi di questo dataset si considerano *LIN4_VmSize, LIN4_VmData, LIN4_RSS, LIN4_byte_letti_sec* e *LIN4_byte_scritti_sec* come variabili di risposta, mentre *TIME* è considerata variabile di predizione.

Stima lineare

Stima bivariata di LIN4_VmSize rispetto a TIME



Stima lineare

Stima lineare

LIN4_VmSize = 739753,89 + 0,0252467*TIME

Riepilogo della stima

R-quadro	0,777384
R-quadro corretto	0,777334
Scarto quadratico medio	2124,77
Media della risposta	746628,1
Osservazioni (o somme pesate)	4539

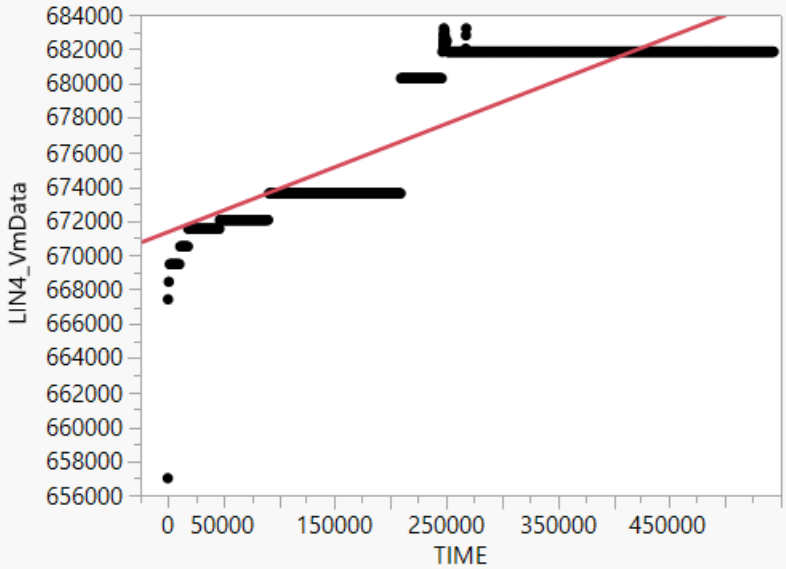
Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	7,1527e+10	7,153e+10	15843,34
Errore	4537	2,0483e+10	4514649,3	**Prob > F**
C. totale	4538	9,201e+10		<,0001*

Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	739753,89	63,06526	11730	<,0001*
TIME	0,0252467	0,000201	125,87	<,0001*

Stima bivariata di LIN4_VmData rispetto a TIME



Stima lineare

Stima lineare

LIN4_VmData = 671349,94 + 0,0252466*TIME

Riepilogo della stima

R-quadro	0,777395
R-quadro corretto	0,777346
Scarto quadratico medio	2124,687
Media della risposta	678224,1
Osservazioni (o somme pesate)	4539

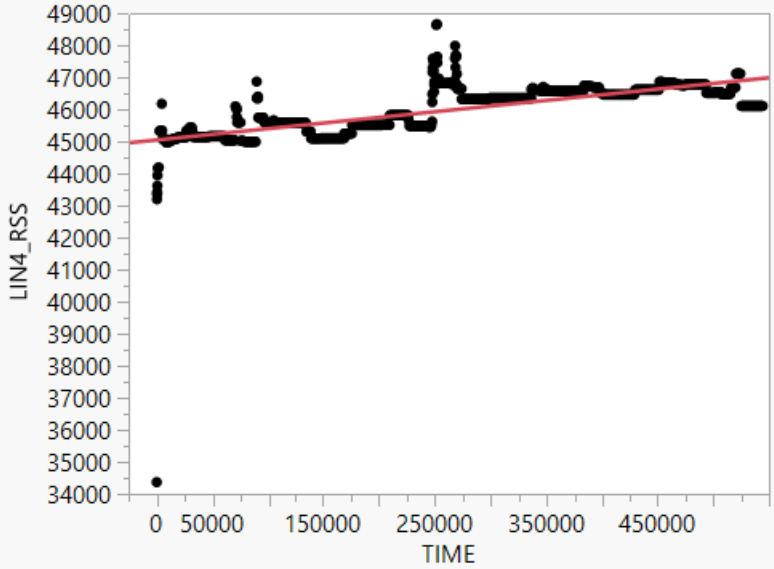
Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	7,1526e+10	7,153e+10	15844,41
Errore	4537	2,0481e+10	4514294,9	**Prob > F**
C. totale	4538	9,2008e+10		<,0001*

Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	671349,94	63,06278	10646	<,0001*
TIME	0,0252466	0,000201	125,87	<,0001*

Stima bivariata di LIN4_RSS rispetto a TIME



Stima lineare

Stima lineare

LIN4_RSS = 45054,086 + 0,0035302*TIME

Riepilogo della stima

R-quadro	0,612477
R-quadro corretto	0,612392
Scarto quadratico medio	441,6245
Media della risposta	46015,3
Osservazioni (o somme pesate)	4539

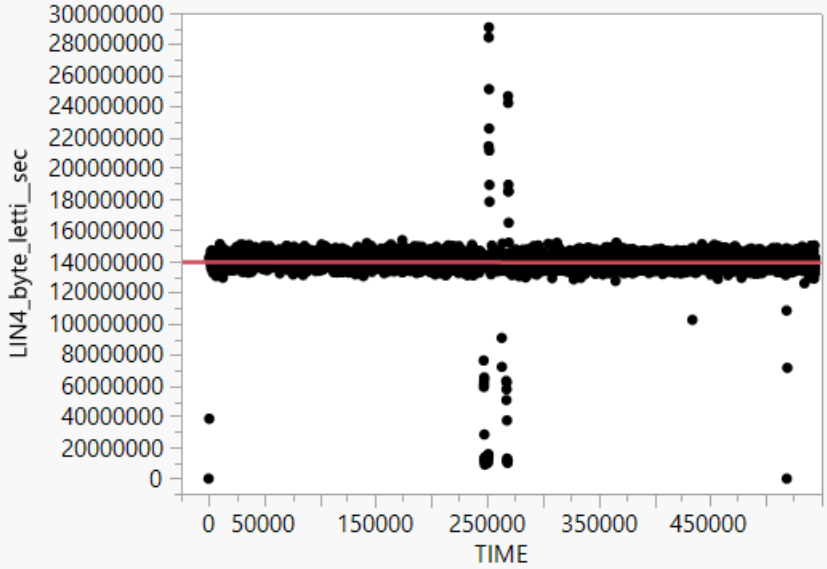
Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	1398515701	1,3985e+9	7170,690
Errore	4537	884861222	195032,23	Prob > F
C. totale	4538	2283376923		<,0001*

Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	45054,086	13,10785	3437,2	<,0001*
TIME	0,0035302	4,169e-5	84,68	<,0001*

Stima bivariata di LIN4_byte_letti__sec rispetto a TIME



Stima lineare

Stima lineare

LIN4_byte_letti__sec = 139632951 - 0,6896483*TIME

Riepilogo della stima

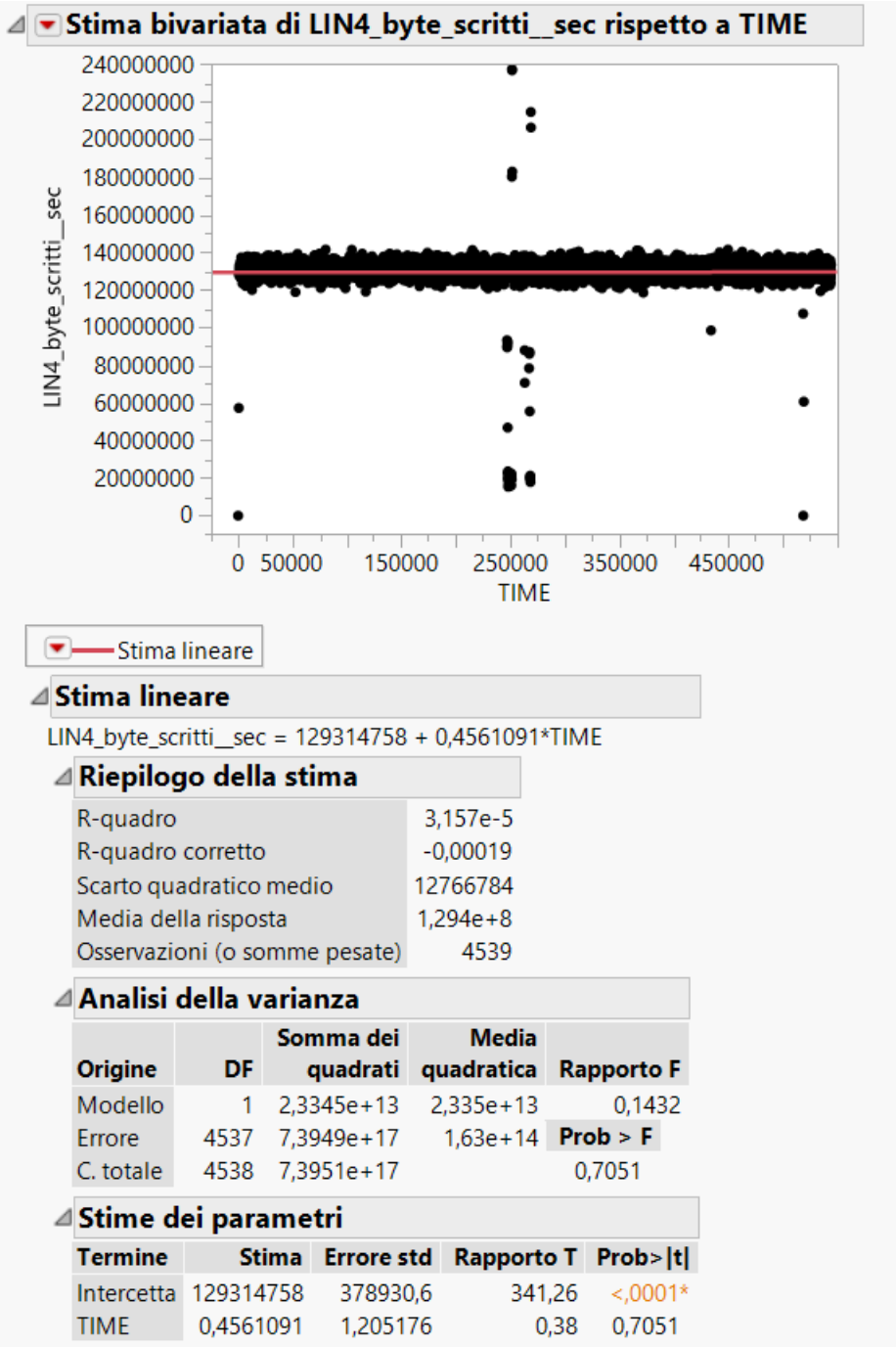
R-quadro	5,045e-5
R-quadro corretto	-0,00017
Scarto quadratico medio	15269052
Media della risposta	1,394e+8
Osservazioni (o somme pesate)	4539

Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	5,3372e+13	5,337e+13	0,2289
Errore	4537	1,0578e+18	2,331e+14	**Prob > F**
C. totale	4538	1,0578e+18		0,6323

Stime dei parametri

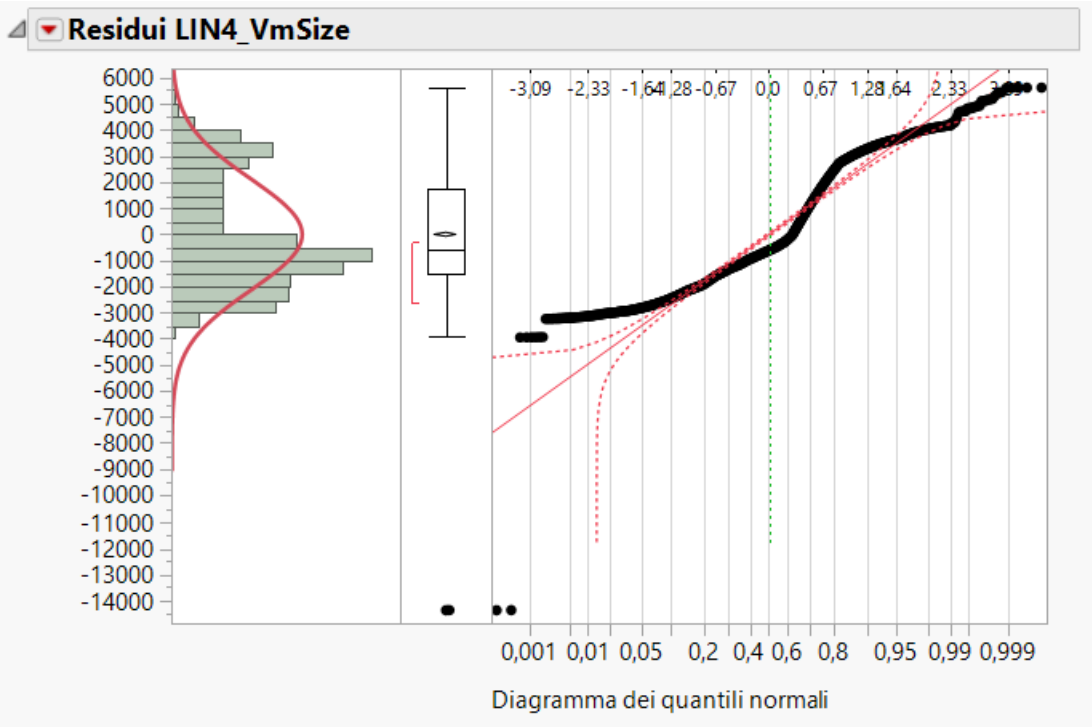
Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	139632951	453200,3	308,10	<,0001*
TIME	-0,689648	1,441389	-0,48	0,6323

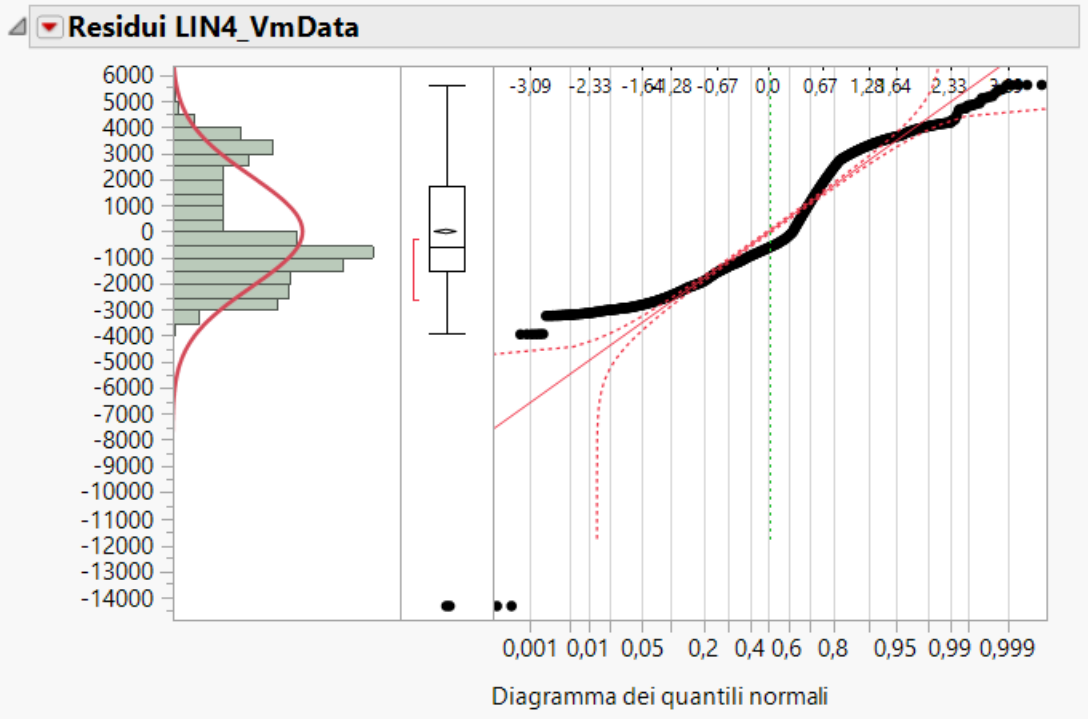


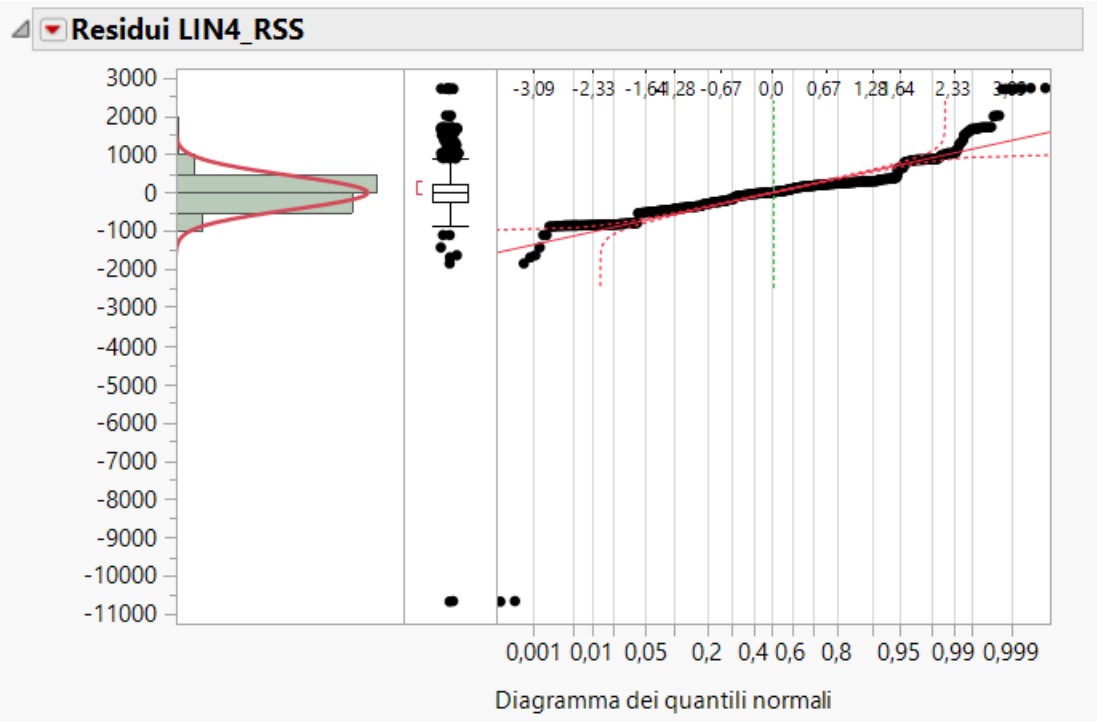
Si osserva quindi che *LIN4_VmSize, LIN4_VmData, LIN4_RSS* hanno un R^2 piuttosto alto ($0.60 < R^2 < 0.78$, per cui *TIME* spiega almeno il 60% della variazione su queste variabili), mentre *LIN4_byte_letti_sec* e *LIN4_byte_scritti_sec* presentano valori prossimi allo 0.

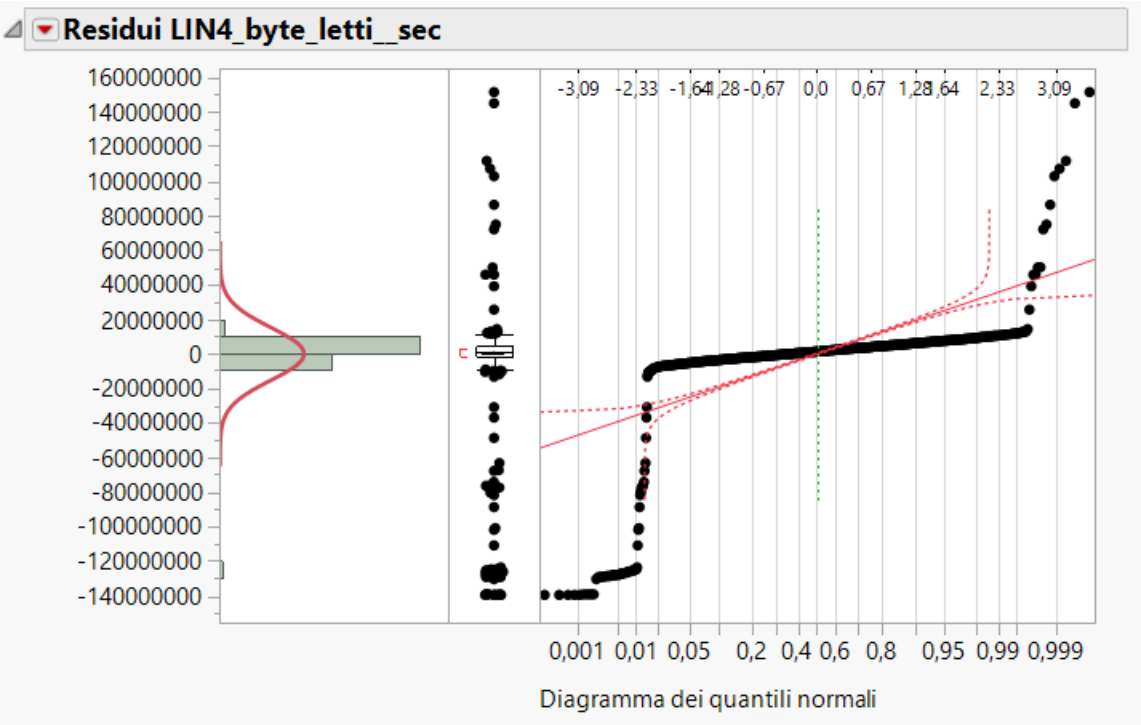
Controllo della normalità

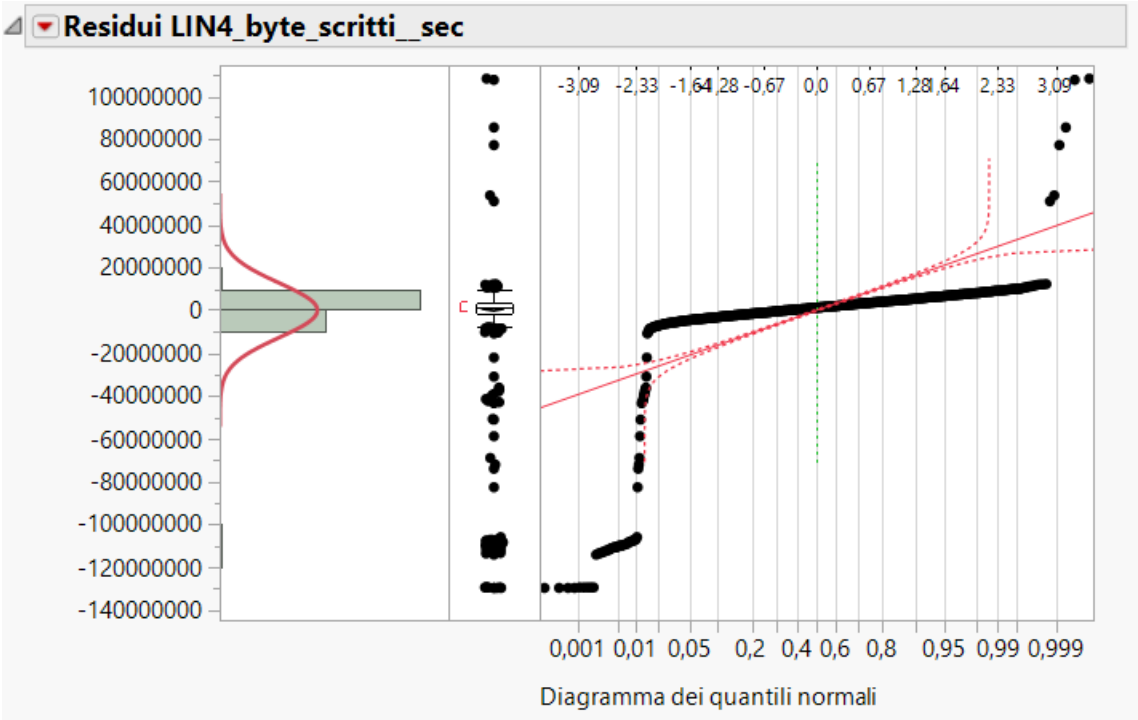
Si procede quindi a fare un test visivo della normalità dei residui dei dati attraverso QQPlot


Residui LIN4_VmSize
6000
5000
4000
3000
2000
1000
0
-1000
-2000
-3000
-4000
-5000
-6000
-7000
-8000
-9000
-10000
-11000
-12000
-13000
-14000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali


Residui LIN4_VmData
6000
5000
4000
3000
2000
1000
0
-1000
-2000
-3000
-4000
-5000
-6000
-7000
-8000
-9000
-10000
-11000
-12000
-13000
-14000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali


Residui LIN4_RSS
3000
2000
1000
0
-1000
-2000
-3000
-4000
-5000
-6000
-7000
-8000
-9000
-10000
-11000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali


Residui LIN4_byte_letti__sec
160000000
140000000
120000000
100000000
80000000
60000000
40000000
20000000
0
-20000000
-40000000
-60000000
-80000000
-100000000
-120000000
-140000000
-3,09 -2,33 -1,64 -1,28 -0,67 0,0 0,67 1,28 1,64 2,33 3,09
0,001 0,01 0,05 0,2 0,4 0,6 0,8 0,95 0,99 0,999
Diagramma dei quantili normali



Già dal test visivo è possibile affermare che **nessuna delle variabili ha distribuzione normale**, risultato supportato anche dal test di KSL:

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,133591	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN4_VmSize

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,133599	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN4_VmData

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,155442	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN4_RSS

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,317205	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN4_byte_letti_sec

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,311102	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

LIN4_byte_scritti_sec

Test di Mann-Kendall

Avendo dimostrato la non-normalità dei dati, viene ritenuto inutile testare la loro omoschedasticità, per cui si procede direttamente ad effettuare il test di Mann-Kendall in modo da determinare se la serie temporale ha un trend monotonamente crescente o decrescente.

Si presentano dunque i risultati ottenuti a seguito del test:

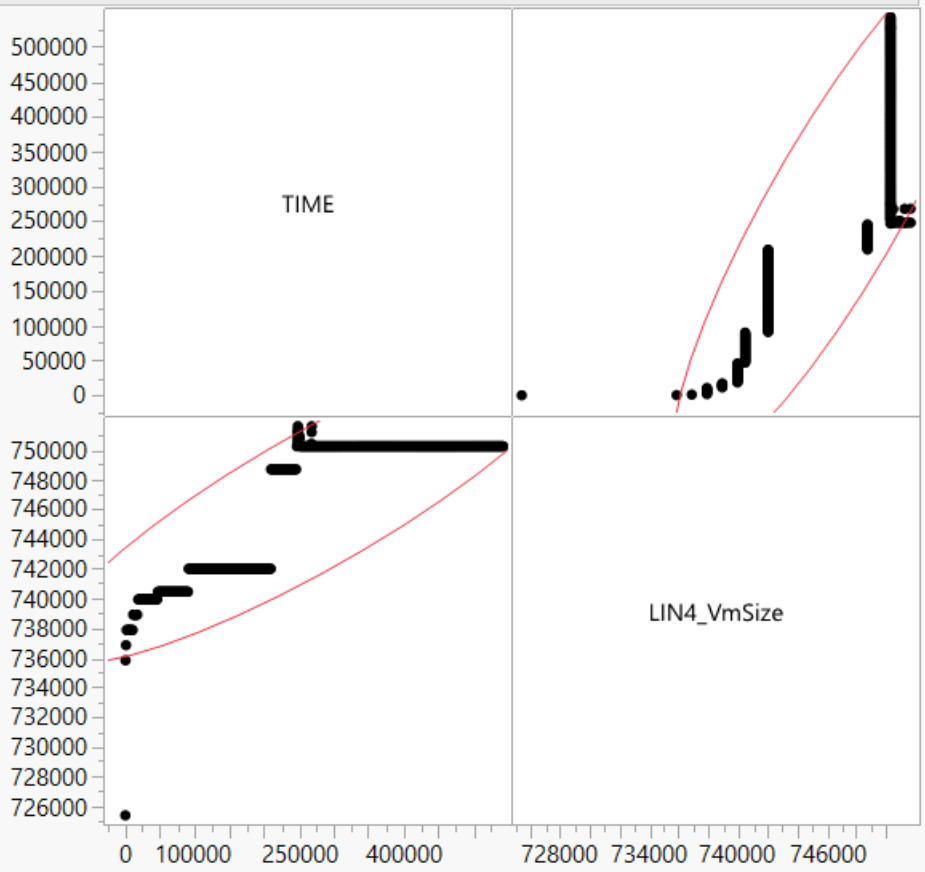
Multivariato

Correlazioni

	TIME	LIN4_VmSize
TIME	1,0000	0,8817
LIN4_VmSize	0,8817	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
LIN4_VmSize	TIME	0,7832	<,0001*	

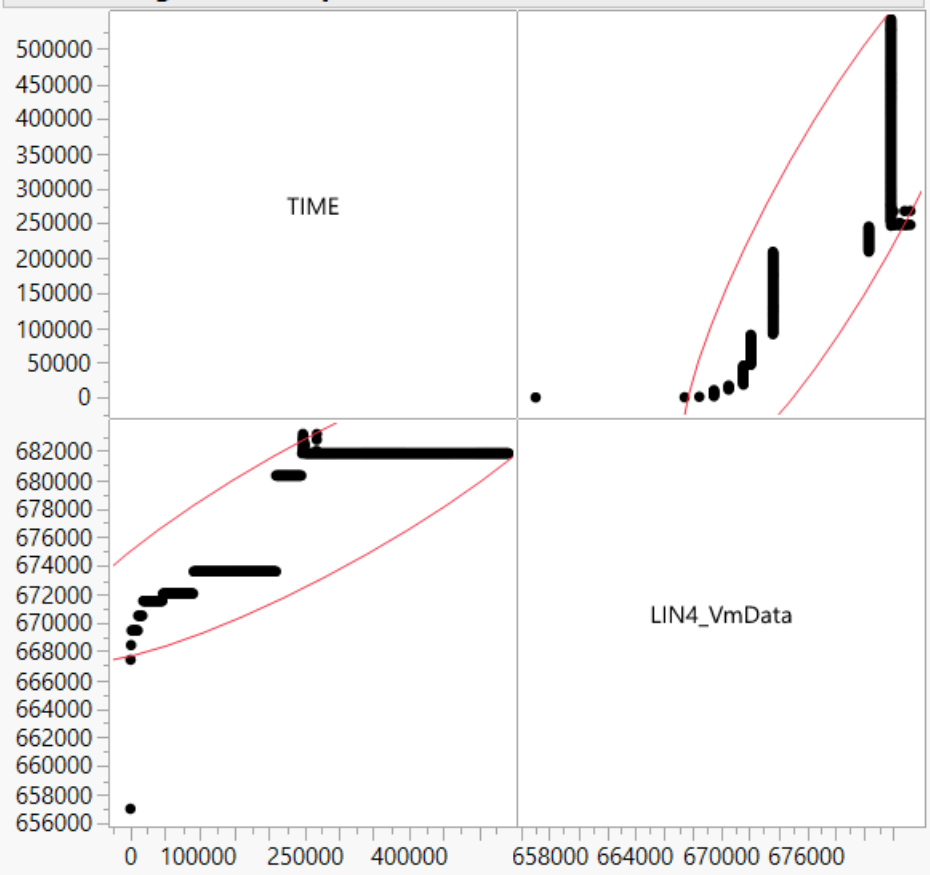
Multivariato

Correlazioni

	TIME	LIN4_VmData
TIME	1,0000	0,8817
LIN4_VmData	0,8817	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
LIN4_VmData	TIME	0,7832	<,0001*	

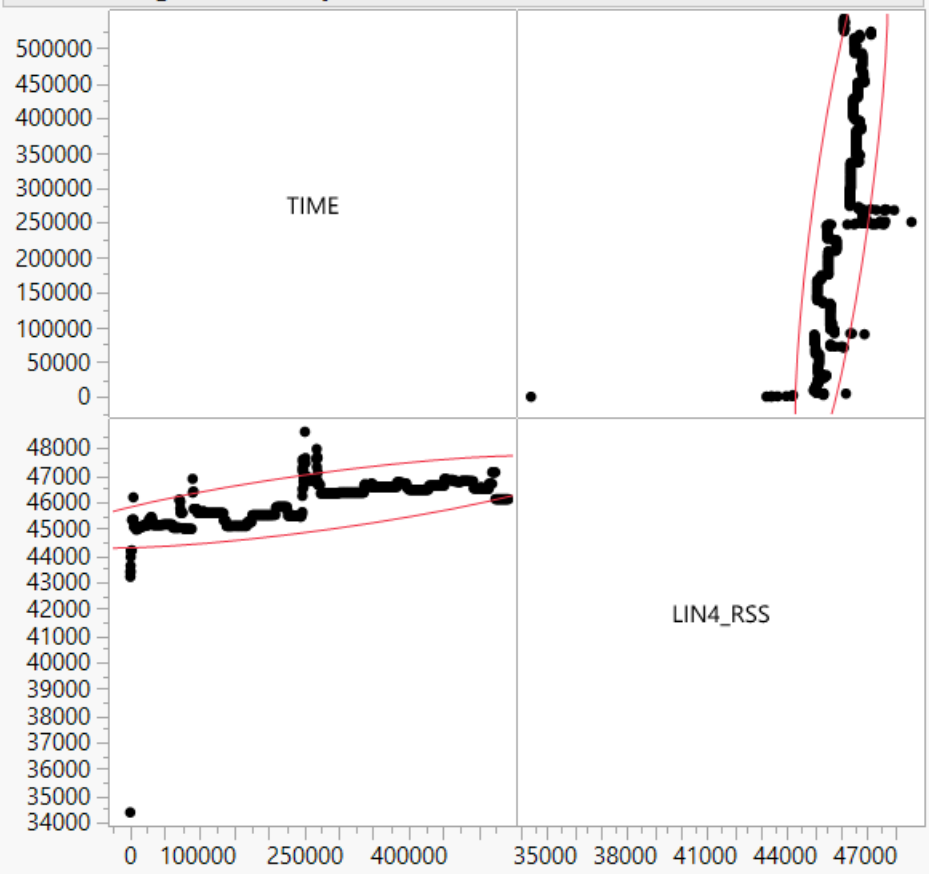
Multivariato

Correlazioni

	TIME	LIN4_RSS
TIME	1,0000	0,7826
LIN4_RSS	0,7826	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
LIN4_RSS	TIME	0,5938	<,0001*	

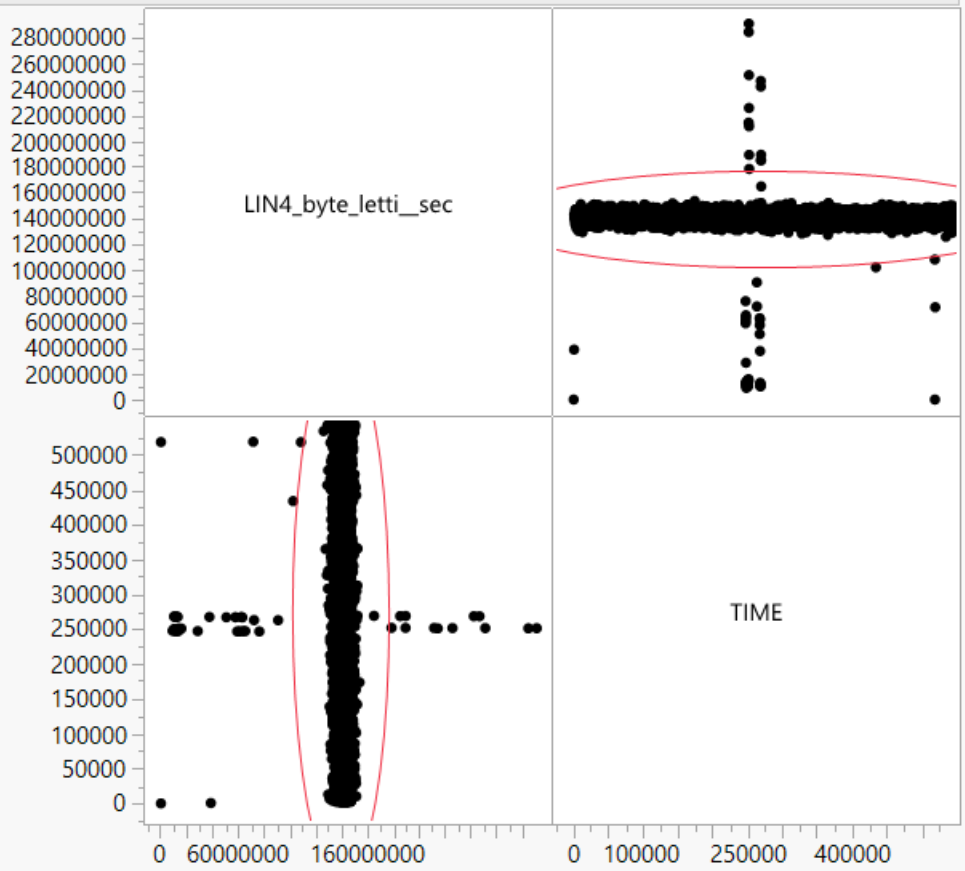
Multivariato

Correlazioni

	LIN4_byte_letti__sec	TIME
LIN4_byte_letti__sec	1,0000	-0,0071
TIME	-0,0071	1,0000

Vi sono 3 valori mancanti. Le correlazioni sono stimate per metodo REML.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN4_byte_letti__sec	-0,0506	<,0001*	

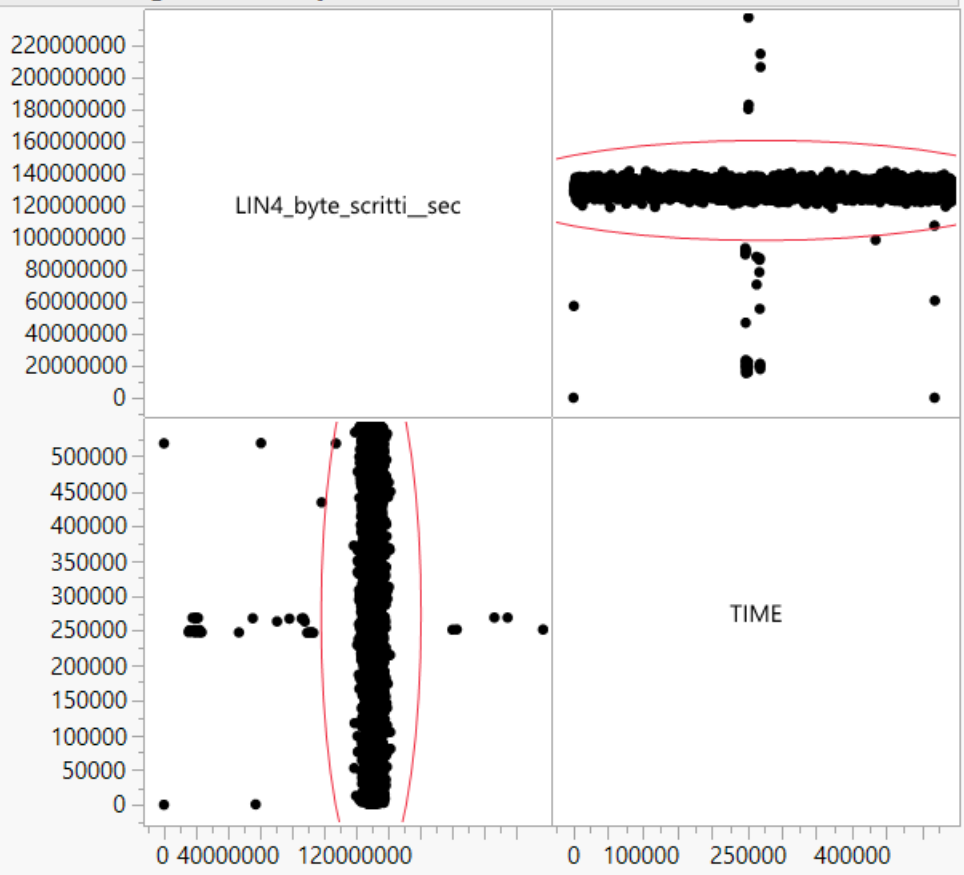
Multivariato

Correlazioni

	LIN4_byte_scritti__sec	TIME
LIN4_byte_scritti__sec	1,0000	0,0056
TIME	0,0056	1,0000

Vi sono 3 valori mancanti. Le correlazioni sono stimate per metodo REML.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
TIME	LIN4_byte_scritti__sec	-0,0171	0,0841	

In questo caso è possibile osservare la presenza di trend statisticamente significativi per le prime 4 variabili (*LIN4_VmSize, LIN4_VmData, LIN4_RSS* e *LIN4_byte_letti_sec*), in quanto il p-value si presenta minore della soglia indicata, ma non per la quinta (*LIN4_byte_scritti_sec*).

Stimatore di Theil-Sen

Si applica quindi la procedura indicata precedentemente per le prime quattro variabili:

	LIN4_VmSize	LIN4_VmData	LIN4_RSS	LIN4_byte_letti_sec
Pendenza	0.022	0.022	0.003	-1.891

	LIN4_VmSize	LIN4_VmData	LIN4_RSS	LIN4_byte_letti_sec
Intercetta	744374.599	675970.599	45386.471	141291080.771
Intervallo di confidenza superiore	0.022	0.022	0.004	-1.168
Intervallo di confidenza inferiore	0.021	0.021	0.003	-2.618

Nessuna delle variabili ha intervallo di confidenza che include lo 0, per cui è possibile affermare con una probabilità del 95% che le pendenze delle rette siano diverse da 0.

Confronto tra i trend individuati nei dataset

I dataset analizzati sono relativi all'occupazione di memoria in una macchina virtuale: nel caso di os2 tutte le variabili presentano trend, mentre per os1 e os3 solo le prime 4.

È possibile dunque confrontare i trend ottenuti nei diversi dataset rispetto alla stessa variabile, in particolare:

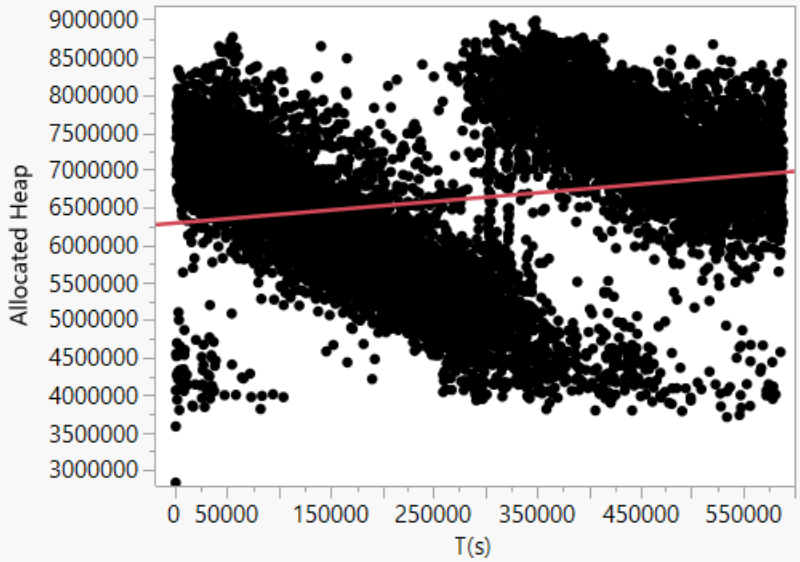
- Per **VMSize** e **VMData** si ha che in tutti e tre i casi la pendenza è positiva, anche se piuttosto bassa: nel caso di os2 si ha addirittura la probabilità che questa sia in realtà pari a 0. In generale è possibile però affermare che queste variabili crescano con il passare del tempo, per cui si nota un utilizzo crescente della memoria in generale.
- Analogo è il caso per **VMRSS**, per cui è possibile fare le stesse considerazioni e dedurre che si ha un trend crescente anche per l'utilizzo dello spazio di archiviazione.
- Per quanto riguarda **Byte_letti_sec** si nota che il trend cambia a seconda del dataset: si osserva infatti un trend positivo per os2 (con pendenza anche piuttosto elevata) e negativo invece per os3; impossibile invece fare confronti sulla variabile **Byte_scritti_sec** in quanto nel caso di os2 non presenta trend statisticamente significativi, e non è possibile dunque individuarne pendenza e intercetta.

VMres1

Per l'analisi di questo dataset si considera *Allocated Heap* come variabile di risposta, mentre *T(s)* è considerata variabile di predizione.

Stima lineare

Stima bivariata di Allocated Heap rispetto a T(s)



Stima lineare

Stima lineare

Allocated Heap = 6296499,4 + 1,1503066*T(s)

Riepilogo della stima

R-quadro	0,033711
R-quadro corretto	0,033613
Scarto quadratico medio	1045362
Media della risposta	6634621
Osservazioni (o somme pesate)	9799

Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	3,735e+14	3,735e+14	341,7907
Errore	9797	1,0706e+16	1,093e+12	**Prob > F**
C. totale	9798	1,1079e+16		<,0001*

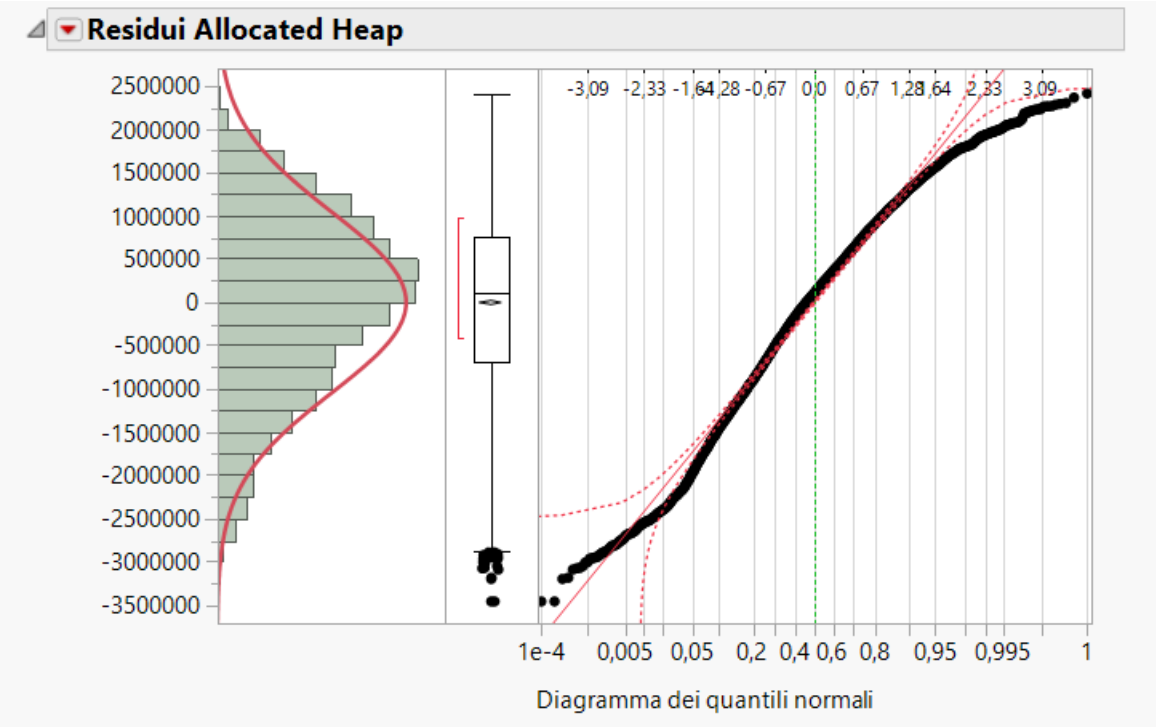
Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	6296499,4	21118,97	298,14	<,0001*
T(s)	1,1503066	0,062221	18,49	<,0001*

Si osserva il valore R^2 della stima lineare è prossimo allo 0.

Controllo della normalità

Si procede quindi a fare un test visivo della normalità dei residui dei dati attraverso QQPlot



Già dal test visivo è possibile affermare che **nessuna delle variabili ha distribuzione normale**, risultato supportato anche dal test di KSL:

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,044176	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

Test di Mann-Kendall

Avendo dimostrato la non-normalità dei dati, viene ritenuto inutile testare la loro omoschedasticità, per cui si procede direttamente ad effettuare il test di Mann-Kendall in modo da determinare se la serie temporale ha un trend monotonamente crescente o decrescente.

Si presentano dunque i risultati ottenuti a seguito del test:

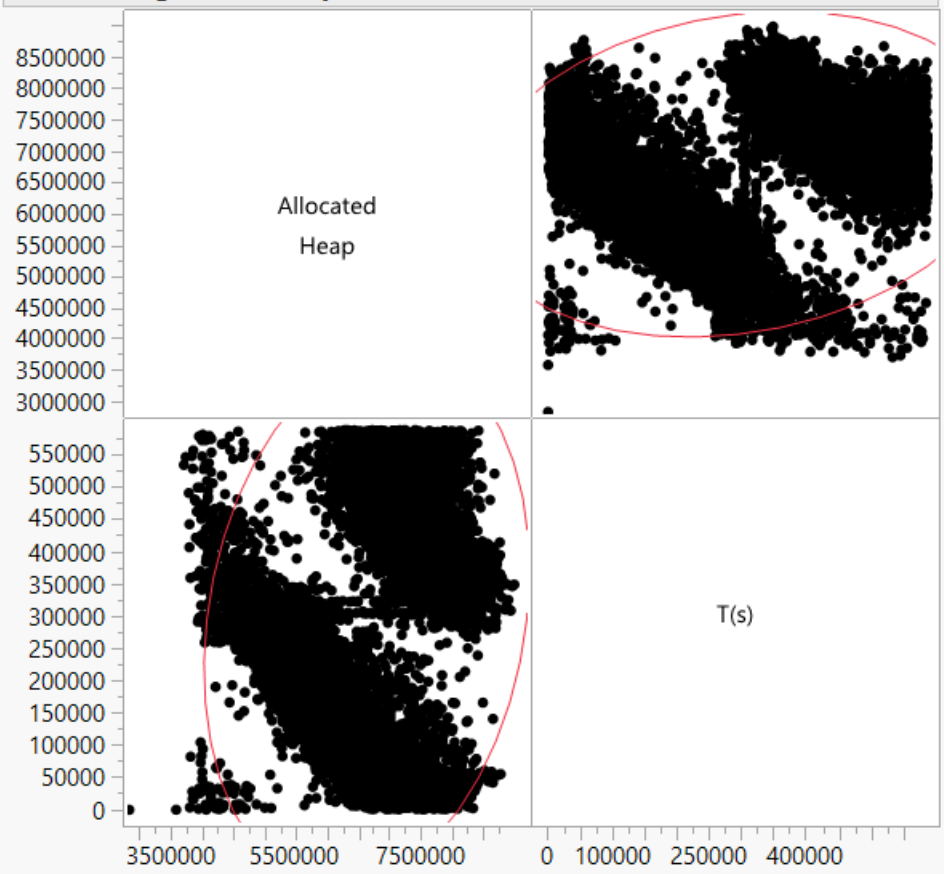
Multivariato

Correlazioni

	Allocated Heap	T(s)
Allocated Heap	1,0000	0,1836
T(s)	0,1836	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
T(s)	Allocated Heap	0,0700	<,0001*	

Il p-value è minore della soglia indicata, per cui possibile osservare la presenza di trend statisticamente significativi per questa variabile.

Stimatore di Theil-Sen

Si applica quindi la procedura indicata precedentemente:

Pendenza	0.691
Intercetta	6562275.462

Intervallo di confidenza superiore	0.818
Intervallo di confidenza inferiore	0.564

Osservando quindi che l'intervallo di confidenza non include lo 0 è possibile affermare con una probabilità del 95% che la pendenza della retta sia diversa da 0.

Failure prediction

Una volta stimato il trend relativo all'*Allocated Heap* è possibile definire **il tempo di saturazione della memoria heap***;* supponendo che il suo limite massimo sia di 1 GByte, il tempo di saturazione (e dunque il tempo per arrivare al fallimento) può essere calcolato come segue.

Si considera in primo luogo la retta stimata:

$$\begin{aligned} \text{Allocated Heap} &= m + b \cdot T(s) = \\ &= 6562275.46 + (0.69 \pm 0.13) \cdot T(s) \end{aligned}$$

Si pone Allocated Heap = 1 GB = 1073741824 B e si calcola la formula inversa per trovare il corrispettivo $T(s)$:

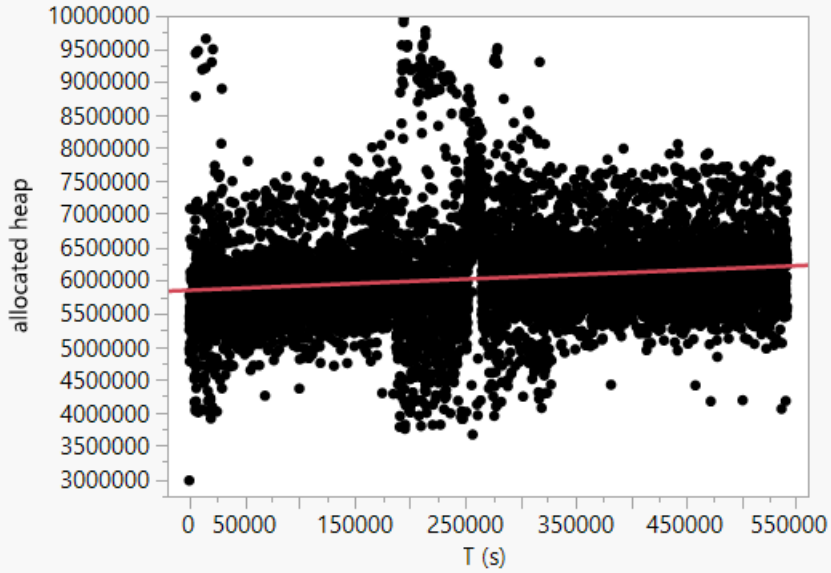
$$\begin{aligned} T_{\text{failure}}(s) &= \frac{1073741824 - 6562275.46}{(0.69 \pm 0.13)} = \\ &= (1546637027 \pm 277369603)s = (49 \pm 8.8) \text{ anni} \end{aligned}$$

VMres2

Per l'analisi di questo dataset si considera *Allocated Heap* come variabile di risposta, mentre *T(s)* è considerata variabile di predizione.

Stima lineare

Stima bivariata di allocated heap rispetto a T (s)



Stima lineare

Stima lineare

allocated heap = 5850817,4 + 0,6750272*T (s)

Riepilogo della stima

R-quadro	0,024828
R-quadro corretto	0,02472
Scarto quadratico medio	661296,9
Media della risposta	6033541
Osservazioni (o somme pesate)	9024

Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	1,0045e+14	1,005e+14	229,7037
Errore	9022	3,9454e+15	4,373e+11	**Prob > F**
C. totale	9023	4,0459e+15		<,0001*

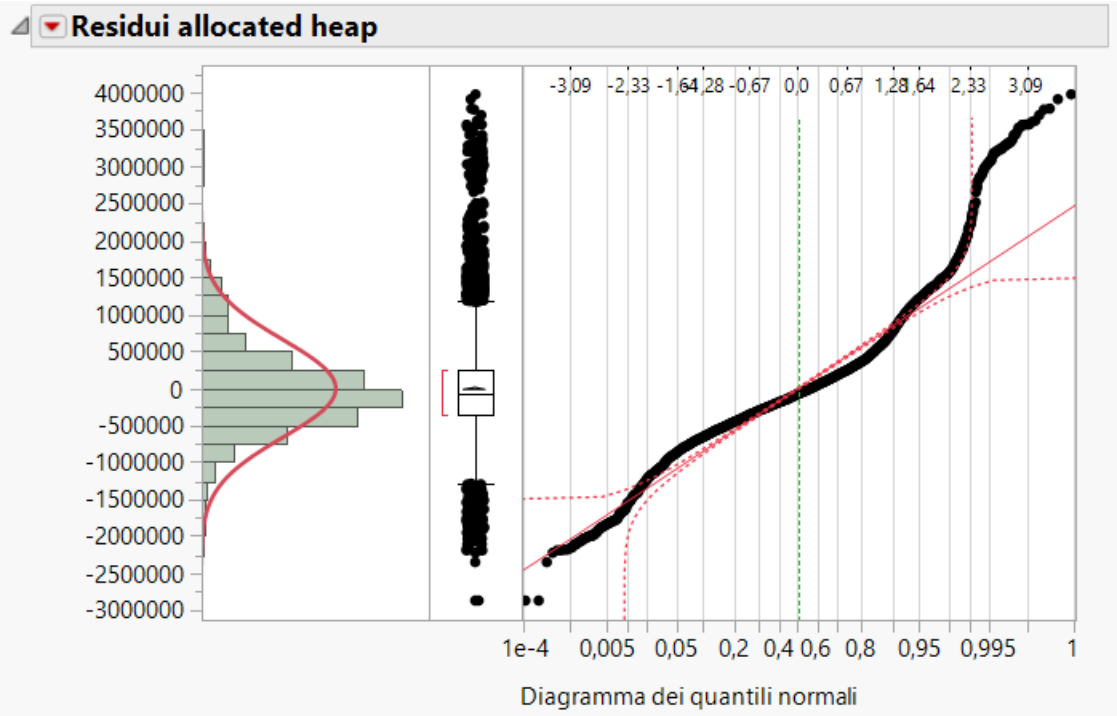
Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	5850817,4	13921,65	420,27	<,0001*
T (s)	0,6750272	0,044539	15,16	<,0001*

Si osserva il valore R^2 della stima lineare è prossimo allo 0.

Controllo della normalità

Si procede quindi a fare un test visivo della normalità dei residui dei dati attraverso QQPlot



Già dal test visivo è possibile affermare che **nessuna delle variabili ha distribuzione normale**, risultato supportato anche dal test di KSL:

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,100369	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

Test di Mann-Kendall

Avendo dimostrato la non-normalità dei dati, viene ritenuto inutile testare la loro omoschedasticità, per cui si procede direttamente ad effettuare il test di Mann-Kendall in modo da determinare se la serie temporale ha un trend monotonamente crescente o decrescente.

Si presentano dunque i risultati ottenuti a seguito del test:

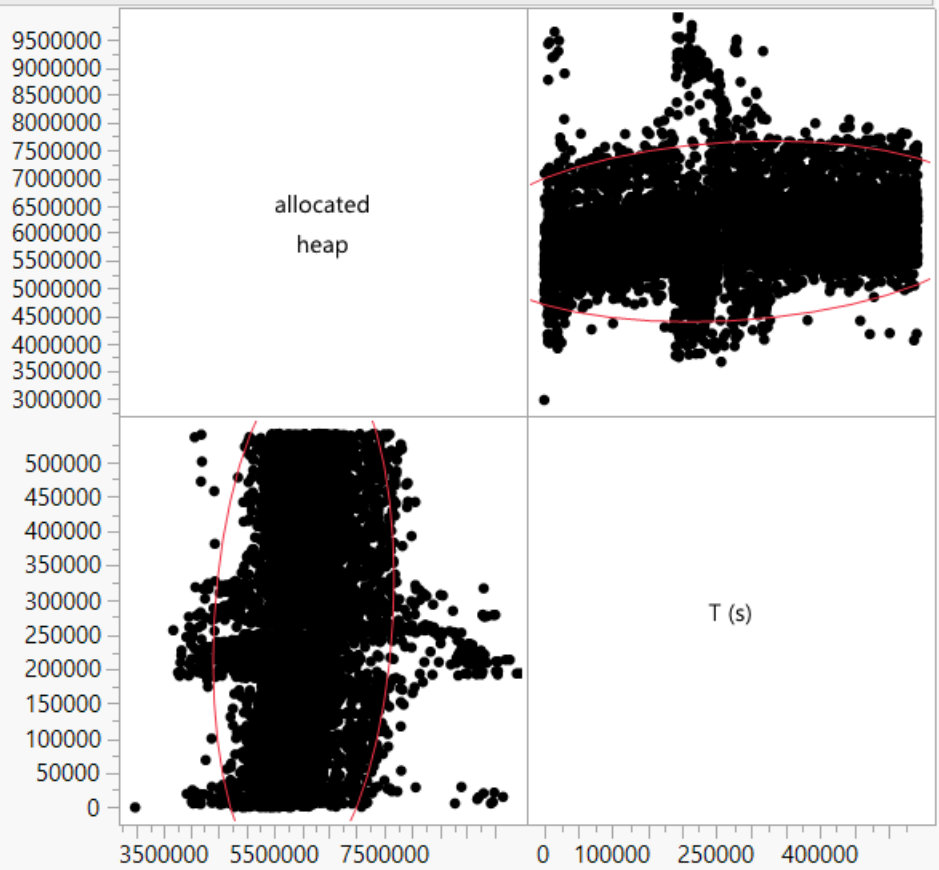
Multivariato

Correlazioni

	allocated heap	T (s)
allocated heap	1,0000	0,1576
T (s)	0,1576	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
T (s)	allocated heap	0,1378	<,0001*	

Il p-value è minore della soglia indicata, per cui possibile osservare la presenza di trend statisticamente significativi per questa variabile.

Stimatore di Theil-Sen

Si applica quindi la procedura indicata precedentemente:

Pendenza	0.671
Intercetta	5780180.563

Intervallo di confidenza superiore	0.737
Intervallo di confidenza inferiore	0.605

Osservando quindi che l'intervallo di confidenza non include lo 0 è possibile affermare con una probabilità del 95% che la pendenza della retta sia diversa da 0.

Failure prediction

Una volta stimato il trend relativo all'*Allocated Heap* è possibile definire **il tempo di saturazione della memoria heap***;* supponendo che il suo limite massimo sia di 1 GByte, il tempo di saturazione (e dunque il tempo per arrivare al fallimento) può essere calcolato come segue.

Si considera in primo luogo la retta stimata:

$$\begin{aligned} \text{Allocated Heap} &= m + b \cdot T(s) = \\ &= 5780180.56 + (0.67 \pm 0.07) \cdot T(s) \end{aligned}$$

Si pone Allocated Heap = 1 GB = 1073741824 B e si calcola la formula inversa per trovare il corrispettivo $T(s)$:

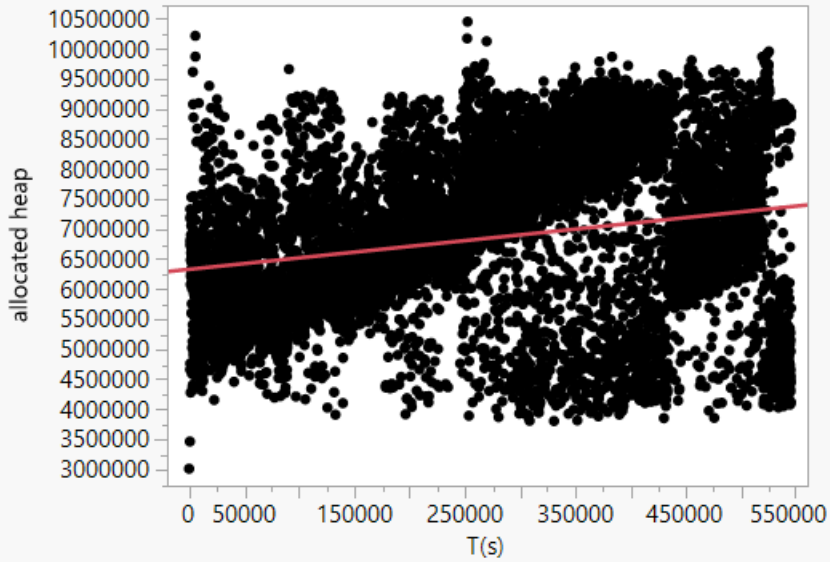
$$\begin{aligned} T_{\text{failure}}(s) &= \frac{1073741824 - 5780180.56}{(0.67 \pm 0.07)} = \\ &= (1593972602 \pm 168372331)s = (50.5 \pm 5.3) \text{ anni} \end{aligned}$$

VMres3

Per l'analisi di questo dataset si considera *Allocated Heap* come variabile di risposta, mentre *T(s)* è considerata variabile di predizione.

Stima lineare

Stima bivariata di allocated heap rispetto a T(s)



Stima lineare

Stima lineare

allocated heap = 6345728,9 + 1,9133287*T(s)

Riepilogo della stima

R-quadro	0,052082
R-quadro corretto	0,051977
Scarto quadratico medio	1285303
Media della risposta	6867436
Osservazioni (o somme pesate)	9090

Analisi della varianza

Origine	DF	Somma dei quadrati	Media quadratica	Rapporto F
Modello	1	8,2488e+14	8,249e+14	499,3227
Errore	9088	1,5013e+16	1,652e+12	**Prob > F**
C. totale	9089	1,5838e+16		<,0001*

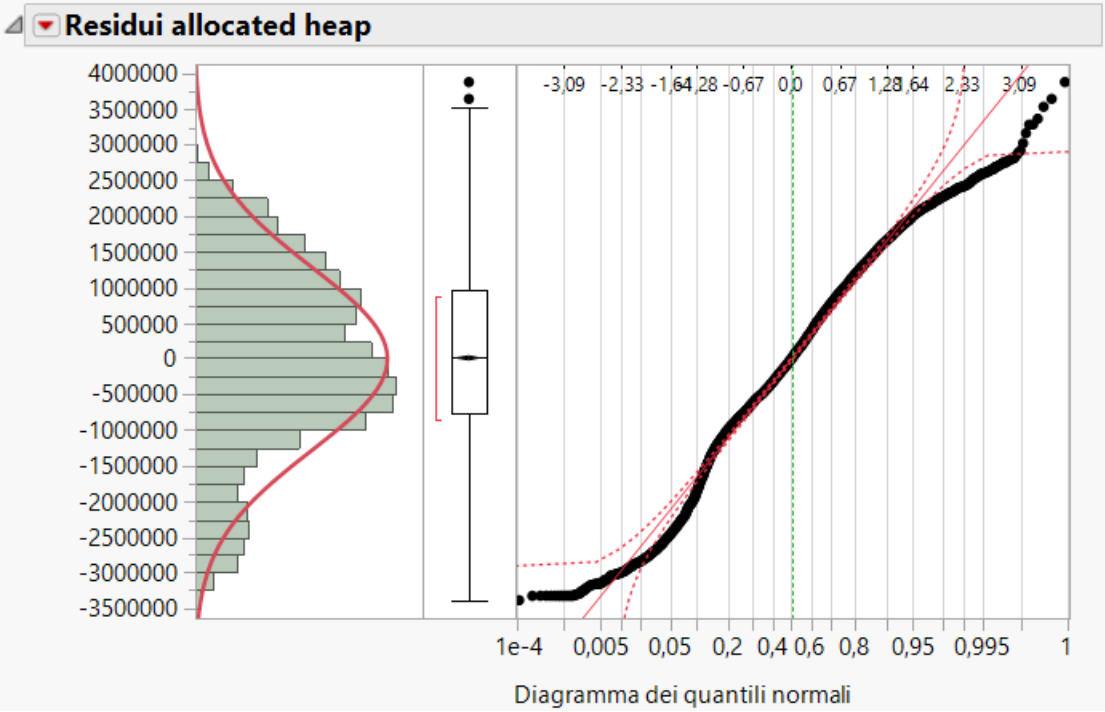
Stime dei parametri

Termine	Stima	Errore std	Rapporto T	Prob>|t|
Intercetta	6345728,9	26959,86	235,38	<,0001*
T(s)	1,9133287	0,085625	22,35	<,0001*

Si osserva il valore R^2 della stima lineare è prossimo allo 0.

Controllo della normalità

Si procede quindi a fare un test visivo della normalità dei residui dei dati attraverso QQPlot



Già dal test visivo è possibile affermare che **nessuna delle variabili ha distribuzione normale**, risultato supportato anche dal test di KSL:

Test della bontà di adattamento

Test di KSL

D		Prob>D
0,030198	<	0,0100*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

Test di Mann-Kendall

Avendo dimostrato la non-normalità dei dati, viene ritenuto inutile testare la loro omoschedasticità, per cui si procede direttamente ad effettuare il test di Mann-Kendall in modo da determinare se la serie temporale ha un trend monotonamente crescente o decrescente.

Si presentano dunque i risultati ottenuti a seguito del test:

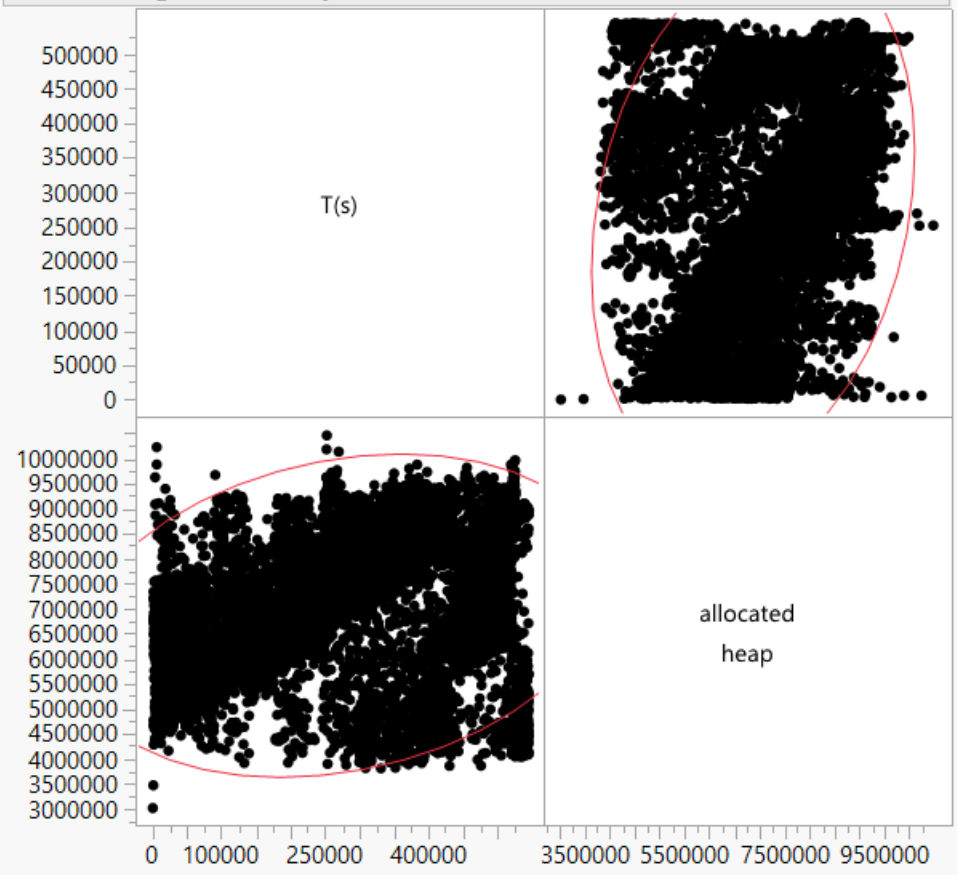
Multivariato

Correlazioni

	T(s)	allocated heap
T(s)	1,0000	0,2282
allocated heap	0,2282	1,0000

Le correlazioni sono stimate per metodo A livello di riga.

Matrice grafico a dispersione



Non parametrico: τ di Kendall

Variabile	Variabile by	τ di Kendall	Prob>|τ|	-,8 -,6 -,4 -,2 0 ,2 ,4 ,6 ,8
allocated heap	T(s)	0,2031	<,0001*	

Il p-value è minore della soglia indicata, per cui possibile osservare la presenza di trend statisticamente significativi per questa variabile.

Stimatore di Theil-Sen

Si applica quindi la procedura indicata precedentemente:

Pendenza	2.903
Intercetta	6041585.556

Intervallo di confidenza superiore	3.092
Intervallo di confidenza inferiore	2.715

Osservando quindi che l'intervallo di confidenza non include lo 0 è possibile affermare con una probabilità del 95% che la pendenza della retta sia diversa da 0.

Failure prediction

Una volta stimato il trend relativo all'*Allocated Heap* è possibile definire **il tempo di saturazione della memoria heap***;* supponendo che il suo limite massimo sia di 1 GByte, il tempo di saturazione (e dunque il tempo per arrivare al fallimento) può essere calcolato come segue.

Si considera in primo luogo la retta stimata:

$$\text{Allocated Heap} = m + b \cdot T(s) = \ = 6041585.55 + (2.9 \pm 0.19) \cdot T(s)$$

Si pone Allocated Heap = 1 GB = 1073741824 B e si calcola la formula inversa per trovare il corrispettivo $T(s)$:

$$T_{\text{failure}}(s) = \frac{1073741824 - 6041585.55}{(2.9 \pm 0.19)} = \ = (368172496 \pm 24225634)s = (11.7 \pm 0.8) \text{ anni}$$

Confronto tra i risultati ottenuti

Si osserva dunque che mentre VMres1 e VMres2 hanno risultati piuttosto simili, il tempo di failure di VMres3 è considerevolmente minore: questo è dovuto ad una pendenza molto più elevata (al pari dell'intercetta) della retta del trend rispetto alle altre due, per cui la memoria heap tende a riempirsi con una velocità maggiore.

Dovendo scegliere la VM che massimizza il tempo di failure, considerando anche l'errore al 95% dell'intervallo di confidenza, la soluzione più opportuna sarebbe quindi VMres2.

Benchmark

Lo scopo di questo homework è stato quello di rilevare statisticamente una differenza di performance tra due macchine differenti. È stato quindi effettuato un **benchmark** su entrambi i sistemi, nello specifico è stato effettuato un **benchmark di simulazione n-body**. Questo benchmark modella il moto di un certo numero di corpi celesti in un sistema gravitazionale.

Nel corso dell'homework sono stati seguiti i seguenti passaggi:

- **Misurazioni e raccolta dei dati:** è stato eseguito *n-body* su entrambi i processori con stessi parametri di avvio. In particolare sono state effettuate, per ogni test, **5** ripetizioni, e il numero di test eseguiti per ogni dimensione di corpi è stato **31**, in modo da ottenere un numero significativo di dati per valutare la **dimensione campionaria**.
- **Verifica dati campionari e test visivi:** sono stati calcolati i valori di media e deviazione standard campionari dei **tempi di esecuzione** per entrambi i processori, e di conseguenza sono stati eseguiti dei test visivi per valutare l'intersezione degli intervalli di confidenza e la normalità delle distribuzioni.
- **Test statistico:** per effettuare un confronto statisticamente significativo, sono stati effettuati dei test statistici per valutare la differenza tra i due sistemi.

Sono di seguito riportate le caratteristiche tecniche delle due macchine su cui è stato effettuato il benchmark, ponendo attenzione al fatto che entrambe le procedure di misurazione avvenissero sullo **stesso sistema operativo**:

	Laptop	Desktop
CPU	AMD Ryzen 5 3500U (4 core / 8 threads) - 2.4GHz	Amd Ryzen 7 5700G (8 core / 16 threads) - 3.8GHz
RAM	(8 GB + 4 GB) DDR4 - 2400 MT/s	(8 GB + 8 GB) DDR4 - 3600 MT/s
Sistema operativo	Linux Kubuntu 22.04	Linux Kubuntu 22.04

Misurazioni e raccolta dei dati

Per la misurazione dei tempi di esecuzione del benchmark su entrambe le macchine, è stato realizzato uno script bash per l'automatizzazione del processo:

```
cd ~/Dropbox/UNI/Impianti/Esercitazioni/Benchmark/nbody
~/Dropbox/UNI/Impianti/Esercitazioni/Benchmark/nbody/launch_nbody.sh -r 5 -n $1 >> benchmark_{$2}.txt

N_ITERATIONS=$((N_ITERATIONS+1))

if [ N_ITERATIONS < 32 ]; then
  reboot
fi
```

Che è stato inserito in esecuzione automatica all'avvio della sessione utente di Linux (in *.bashrc*). Lo script non è solo di scopo pratico, ma anche per garantire l'effettiva **indipendenza delle misurazioni**, in quanto viene eseguito senza l'intervento di un utente e garantisce un riavvio della macchina per ogni misurazione. In poche parole, le misurazioni risultano **indipendenti e identicamente distribuite**.

Questo permette di poter applicare il **Teorema del Limite Centrale**, che garantisce che se vengono prelevati *abbastanza campioni*, la **distribuzione della media** dei campioni si avvicinerà sempre di più a una **distribuzione normale**. Una regola empirica prevederebbe di prelevare almeno **30 campioni** per poter effettuare le stime corrette, ragion per cui è stato deciso di prelevarne **31**.

Verifica dati campionari e test visivi

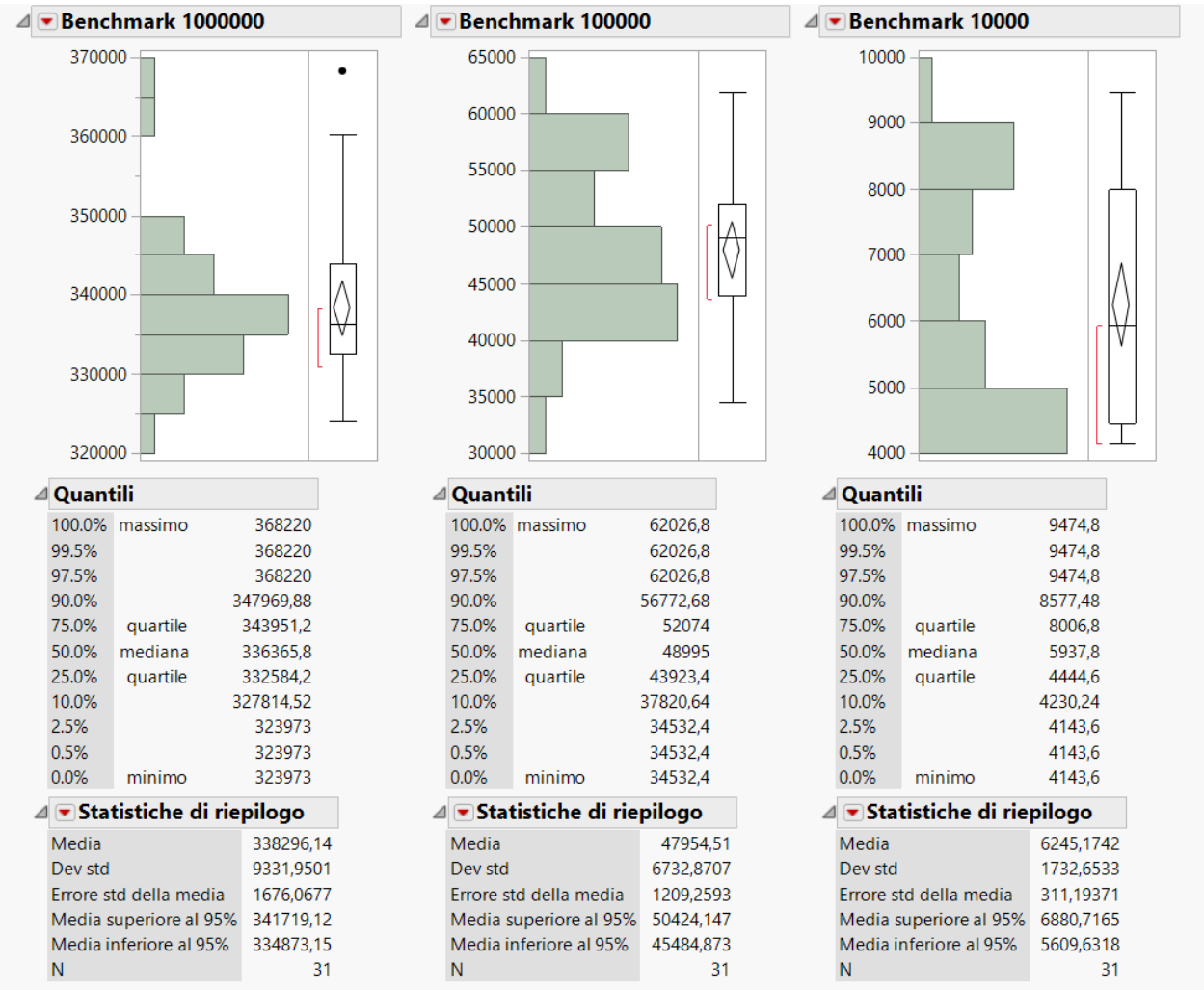
I 31 dati raccolti dei tempi di esecuzione del benchmark sono da considerarsi un **pre-campione**, in quanto ci è utile per valutare l'**effettiva dimensione campionaria** per approssimare media e varianza del campione con quelli della popolazione a un certo intervallo di confidenza (e quindi a una certa percentuale di **errore massimo**). In particolare, ci si vuole assicurare che:

$$|\overline{x} - \mu| < E, \quad E \text{ errore}$$

sia vero su un intervallo di confidenza del 95%. Si calcola quindi la dimensione campionaria, in questo modo:

$$n = \left(\frac{z_{\alpha/2} \cdot s}{E}\right)^2$$

dove s è la **deviazione standard campionaria**, e $z_{\alpha/2} = 1.96$ corrispondente al valore prelevato dalla **z-table**. Di seguito si riportano le distribuzioni, sia per il sistema laptop che per quello desktop, insieme alle relative tabelle in cui sono contenute le dimensioni campionarie calcolate:



Benchmark 1000000

Quantili

100.0%	massimo	368220
99.5%		368220
97.5%		368220
90.0%		347969,88
75.0%	quartile	343951,2
50.0%	mediana	336365,8
25.0%	quartile	332584,2
10.0%		327814,52
2.5%		323973
0.5%		323973
0.0%	minimo	323973

Statistiche di riepilogo

Media	338296,14
Dev std	9331,9501
Errore std della media	1676,0677
Media superiore al 95%	341719,12
Media inferiore al 95%	334873,15
N	31

Benchmark 100000

Quantili

100.0%	massimo	62026,8
99.5%		62026,8
97.5%		62026,8
90.0%		56772,68
75.0%	quartile	52074
50.0%	mediana	48995
25.0%	quartile	43923,4
10.0%		37820,64
2.5%		34532,4
0.5%		34532,4
0.0%	minimo	34532,4

Statistiche di riepilogo

Media	47954,51
Dev std	6732,8707
Errore std della media	1209,2593
Media superiore al 95%	50424,147
Media inferiore al 95%	45484,873
N	31

Benchmark 10000

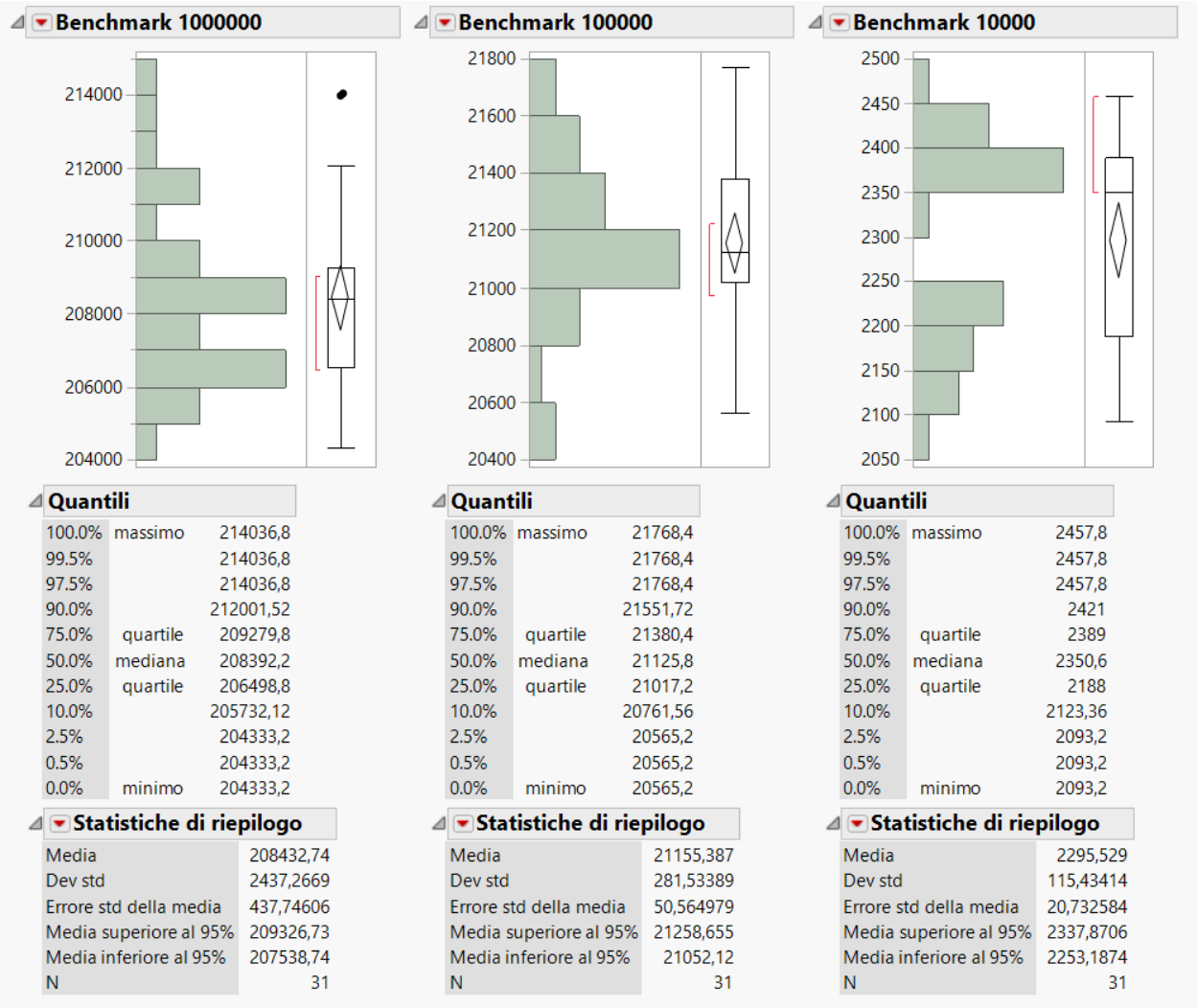
Quantili

100.0%	massimo	9474,8
99.5%		9474,8
97.5%		9474,8
90.0%		8577,48
75.0%	quartile	8006,8
50.0%	mediana	5937,8
25.0%	quartile	4444,6
10.0%		4230,24
2.5%		4143,6
0.5%		4143,6
0.0%	minimo	4143,6

Statistiche di riepilogo

Media	6245,1742
Dev std	1732,6533
Errore std della media	311,19371
Media superiore al 95%	6880,7165
Media inferiore al 95%	5609,6318
N	31

Distribuzioni benchmark laptop



Distribuzioni benchmark desktop

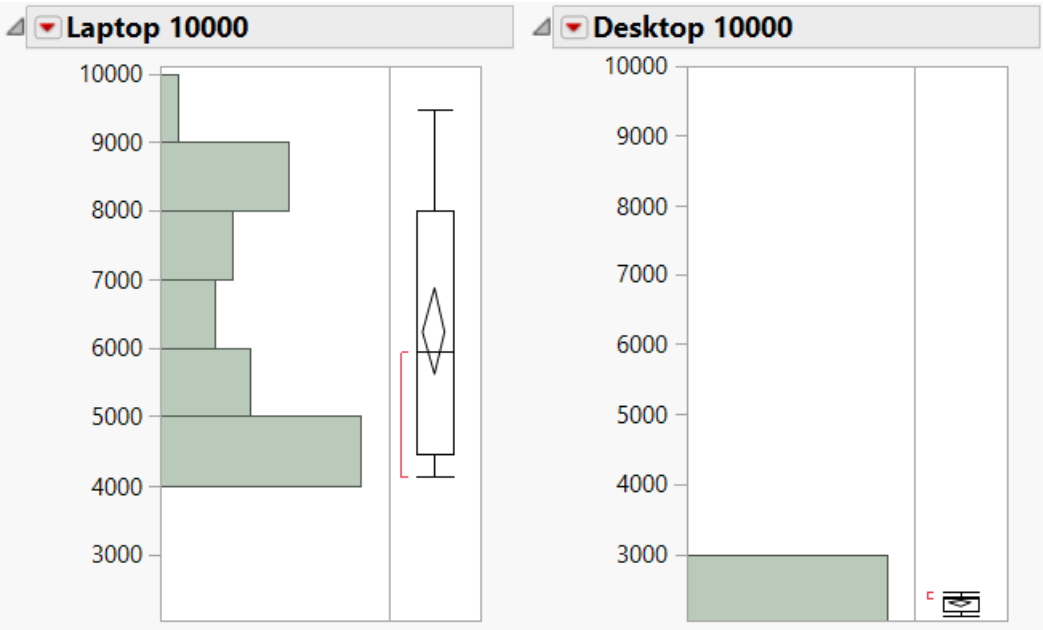
Laptop	10000	100000	1000000
5%	119	31	2
8%	47	23	1
10%	30	15	1

Desktop	10000	100000	1000000
5%	4	1	1
8%	2	1	1
10%	1	1	1

È stato scelto di rappresentare i valori campionari per vari valori dell'errore di stima, e in particolare sono riportati i valori al 5%, 8% e 10% di errore.

Si è optato per un errore al 10%, in quanto non richiede misurazioni aggiuntive ai campioni già misurati. Nel caso in cui si fosse optato per l'errore all'8%, ci sarebbero dovute essere **16 misurazioni supplementari** per il benchmark da 10000 sul laptop.

Confronto delle distribuzioni: 10000



Intervalli di confidenza

Parametro	Stima	CI inferiore	CI superiore	1-Alfa
Media	6245,174	5609,632	6880,717	0,950
Dev std	1732,653	1384,584	2315,991	0,950

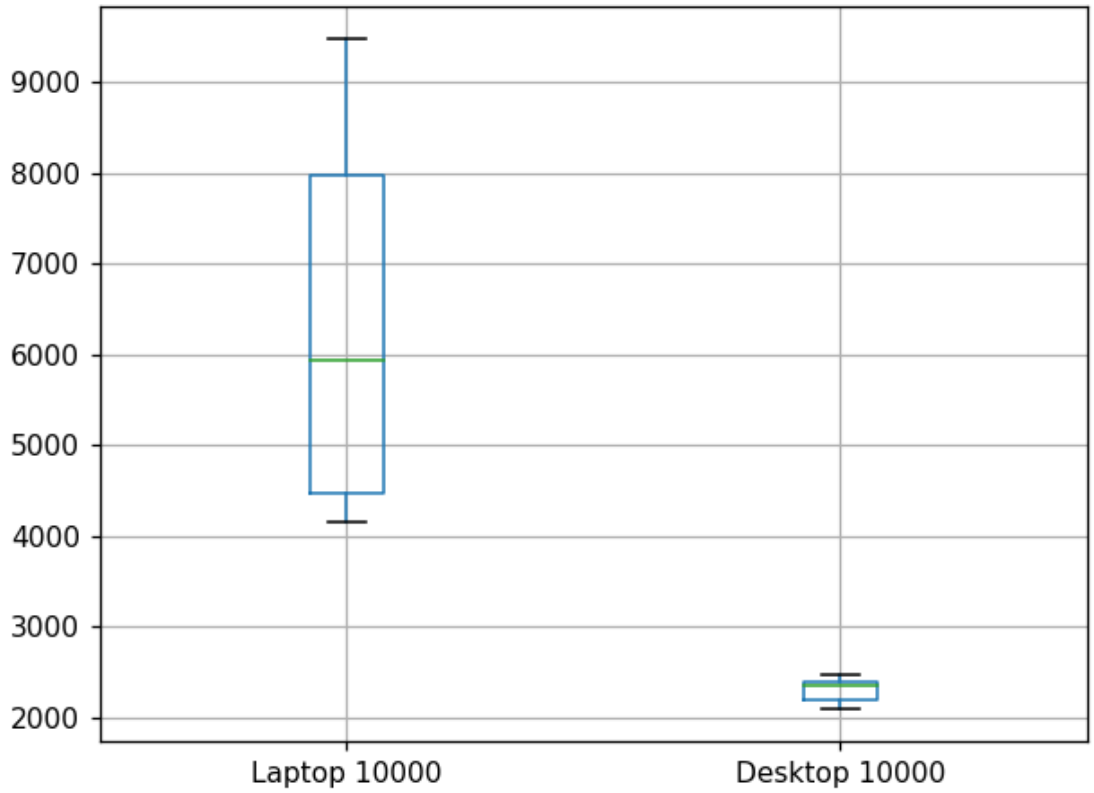
10000 Laptop

Intervalli di confidenza

Parametro	Stima	CI inferiore	CI superiore	1-Alfa
Media	2295,529	2253,187	2337,871	0,950
Dev std	115,4341	92,2448	154,2977	0,950

10000 Desktop

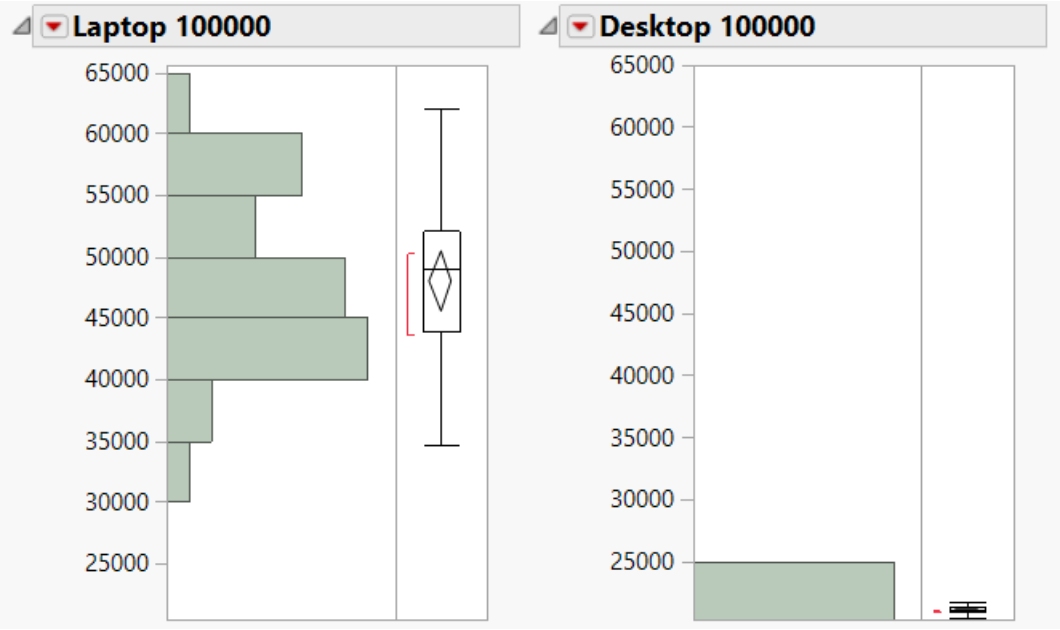
Sia per questo caso che per i prossimi due valori di simulazione, sono stati visualizzati dei grafici rappresentativi per valutare **visivamente** la differenza tra le due distribuzioni. Nel caso di simulazione di **10000 corpi**, si può già vedere una netta differenza tra gli **intervalli di confidenza** delle due distribuzioni, che risultano **fortemente disgiunti**. Un dettaglio di questa analisi è visibile nei boxplot qui riportati in dettaglio:



Il test visivo ci suggerisce, anche se è praticamente possibile affermarlo, che **i due campioni vengono da popolazioni diverse**.

Questa cosa è riscontrabile anche per gli altri valori di numero di corpi da simulare, per cui sono semplicemente riportati graficamente ma senza commento.

Confronto delle distribuzioni: 100000



Intervalli di confidenza

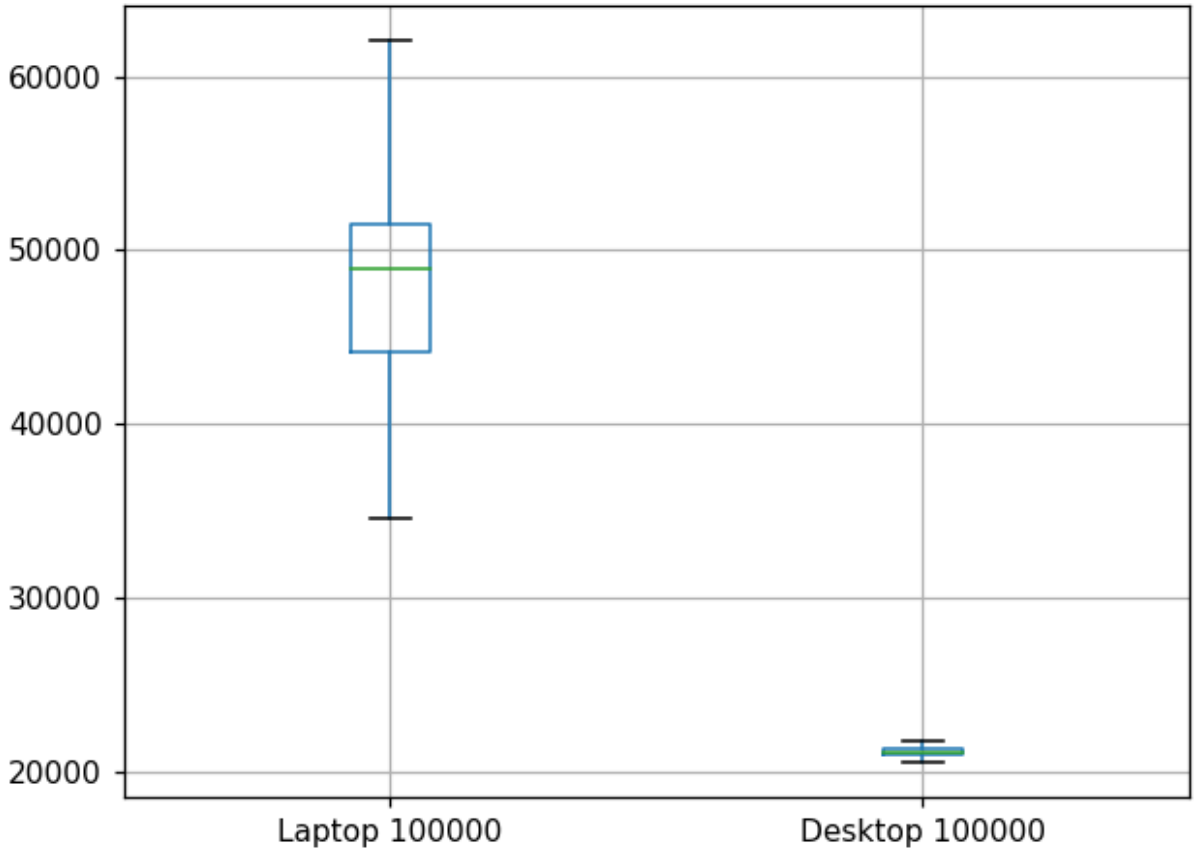
Parametro	Stima	CI inferiore	CI superiore	1-Alfa
Media	47954,51	45484,87	50424,15	0,950
Dev std	6732,871	5380,317	8999,649	0,950

100000 Laptop

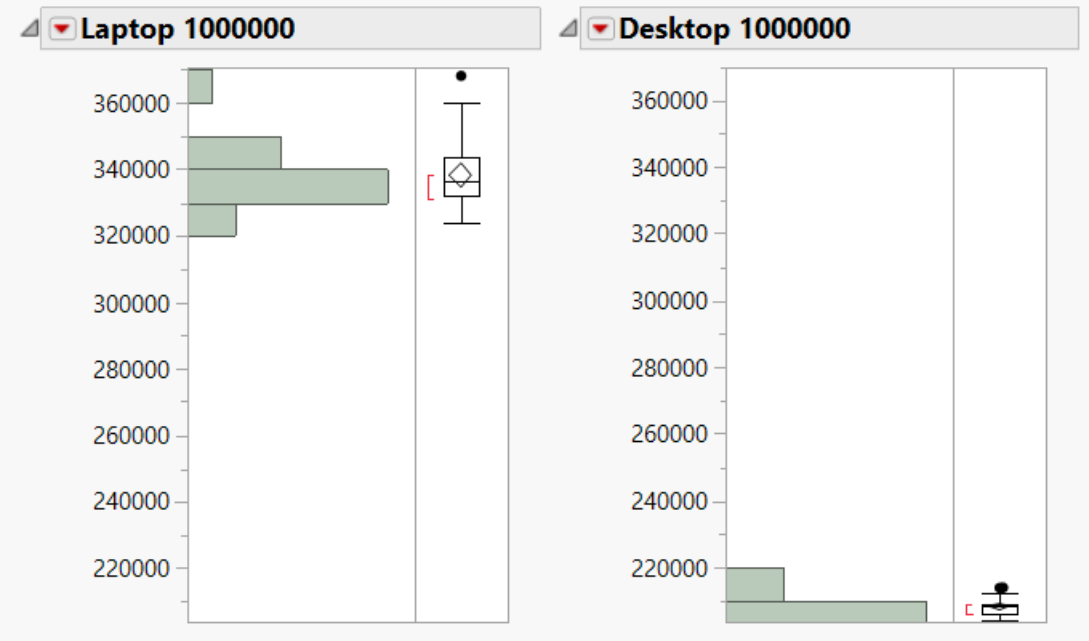
Intervalli di confidenza

Parametro	Stima	CI inferiore	CI superiore	1-Alfa
Media	21155,39	21052,12	21258,65	0,950
Dev std	281,5339	224,9771	376,3188	0,950

100000 Desktop



Confronto delle distribuzioni: 1000000



Intervalli di confidenza

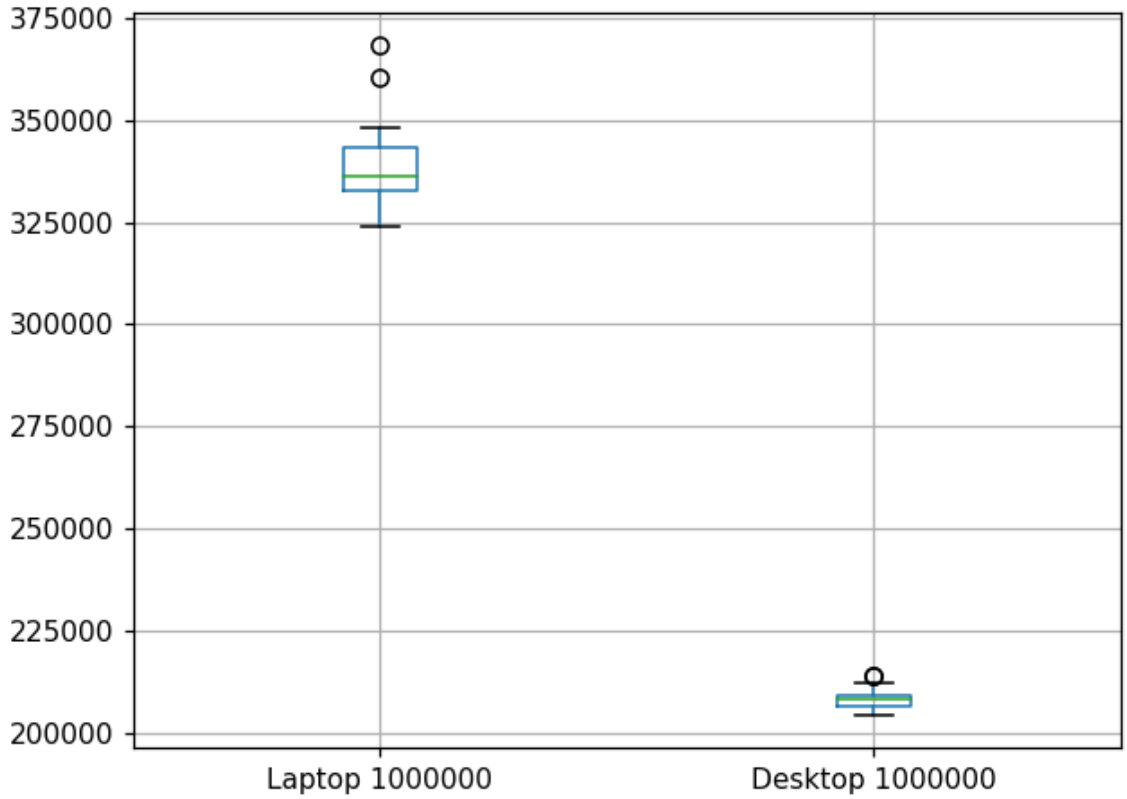
Parametro	Stima	CI inferiore	CI superiore	1-Alfa
Media	338296,1	334873,1	341719,1	0,950
Dev std	9331,95	7457,273	12473,77	0,950

1000000 Laptop

Intervalli di confidenza

Parametro	Stima	CI inferiore	CI superiore	1-Alfa
Media	208432,7	207538,7	209326,7	0,950
Dev std	2437,267	1947,649	3257,83	0,950

1000000 Desktop



Test statistico

Anche se non è pienamente necessario, visto il test visivo delle medie e degli intervalli di confidenza, sono stati effettuati dei **testi di ipotesi statistica** per valutare l'effettiva differenza tra le due distribuzioni di dati. Di seguito è riportata una **tabella di test parametrici** utilizzabili nel caso in esame:

Z -test		
One-sample z-test	$z_0 = \frac{\bar{x} - \mu_0}{\sigma/\sqrt{n}}$	(**Normal population** **or** ***n* large**) **and** σ known.
Two-sample z-test	$z = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$	**Normal population** **and** independent observations **and** σ_1 and σ_2 are known
One-sample t-test	$t = \frac{\bar{x} - \mu_0}{\frac{S}{\sqrt{n}}}$	(**Normal population** **or** ***n* large**) **and** σ unknown
Paired t-test	$t = \frac{\bar{x}_1 - \bar{x}_2}{S_d/\sqrt{n}}$	(**Normal population** of differences **or** ***n* large**) **and** σ unknown
Two-sample t-test (pooled)	$t = \frac{\bar{x}_1 - \bar{x}_2}{S_p\sqrt{1/n_1 + 1/n_2}}$	(**Normal pop.** **or** $n_1 + n_2 > 40$) **and** independent observations **and** $\sigma_1 = \sigma_2$ unknown *S_p is the pooled std*
Two-sample t-test (unpooled)	$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{s_1^2/n_1 + s_2^2/n_2}}$	(**Normal pop.** **or** $n_1 + n_2 > 40$) **and** independent observations **and** $\sigma_1 \neq \sigma_2$ both unknown

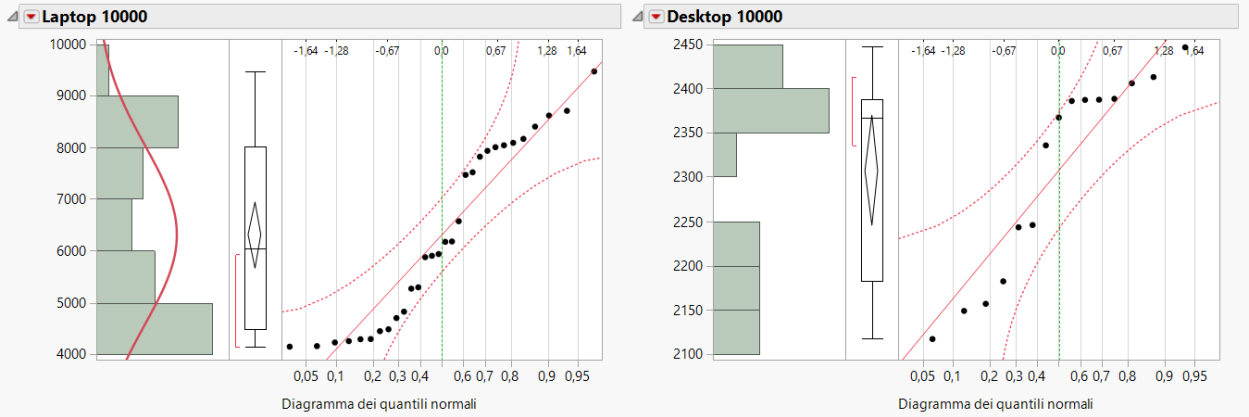
È chiaro che, visto che la varianza è sconosciuta per entrambe le distribuzioni, sarà necessario adoperare un

tipo di **t-test**. In presenza di dati non normali, è inoltre possibile ricorrere a test non parametrici, come il **test di Wilcoxon**.

Si verificano quindi innanzitutto le varie proprietà di normalità dei dati misurati.

Test di normalità

Si noti che per effettuare il test di normalità si è scelto di utilizzare 30 campioni per Latptop 10000 e 15 per tutti gli altri.



Test della bontà di adattamento

Test W di Shapiro-Wilk

W	Prob<W
0,899393	0,0081*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

Laptop 10000

Test della bontà di adattamento

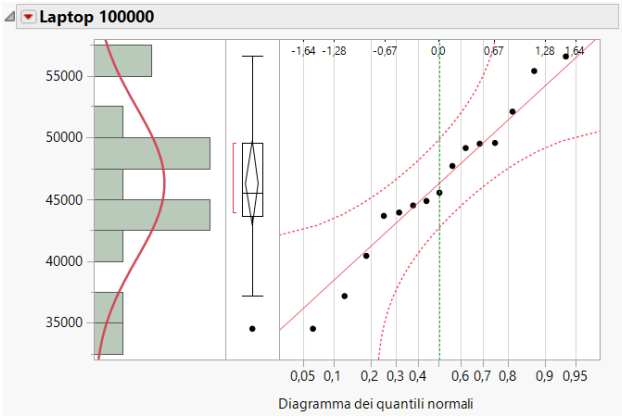
Test W di Shapiro-Wilk

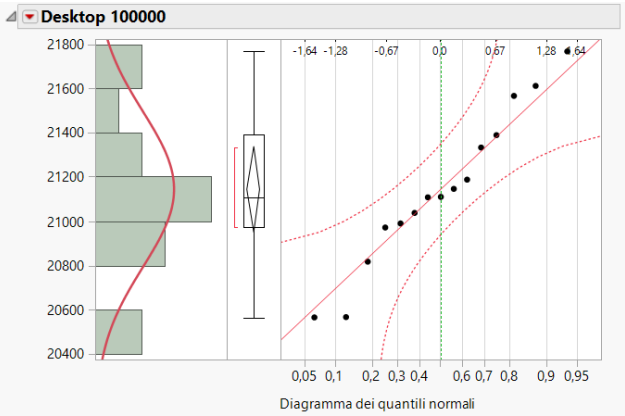
W	Prob<W
0,864738	0,0283*

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

Desktop 10000

Si può osservare dal test di Shapiro-Wilk che le distribuzioni per 10000 corpi non sono normali. Questo potrebbe indurre a pensare di utilizzare un test di Wilcoxon per valutare la similarità tra le distribuzioni, tuttavia con il numero di campioni scelto (30+15=45>40) è anche possibile effettuare un **test parametrico a due sample (t-test unpooled)**.





Test della bontà di adattamento

Test W di Shapiro-Wilk

W	Prob<W
0,975170	0,9259

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

Laptop 100000

Test della bontà di adattamento

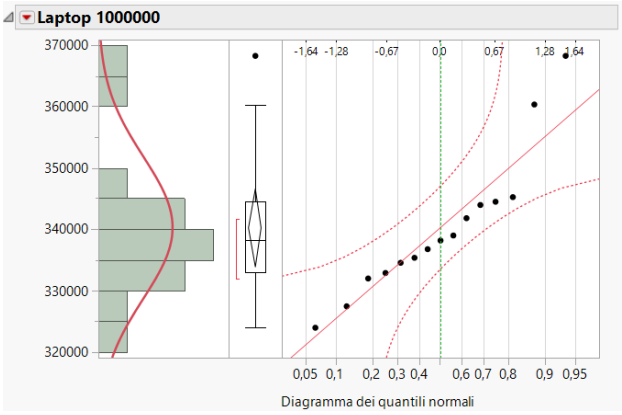
Test W di Shapiro-Wilk

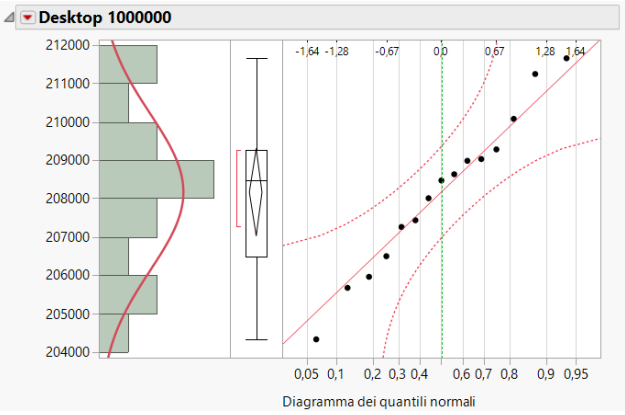
W	Prob<W
0,966121	0,7971

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

Desktop 100000

Si può osservare dal test di Shapiro-Wilk che le distribuzioni per 100000 corpi sono normali.





Test della bontà di adattamento

Test W di Shapiro-Wilk

W	Prob<W
0,902523	0,1040

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

Laptop 1000000

Test della bontà di adattamento

Test W di Shapiro-Wilk

W	Prob<W
0,983362	0,9874

Nota: Ho = i dati provengono dalla distribuzione Normale. I p-value bassi rifiutano Ho.

Desktop 1000000

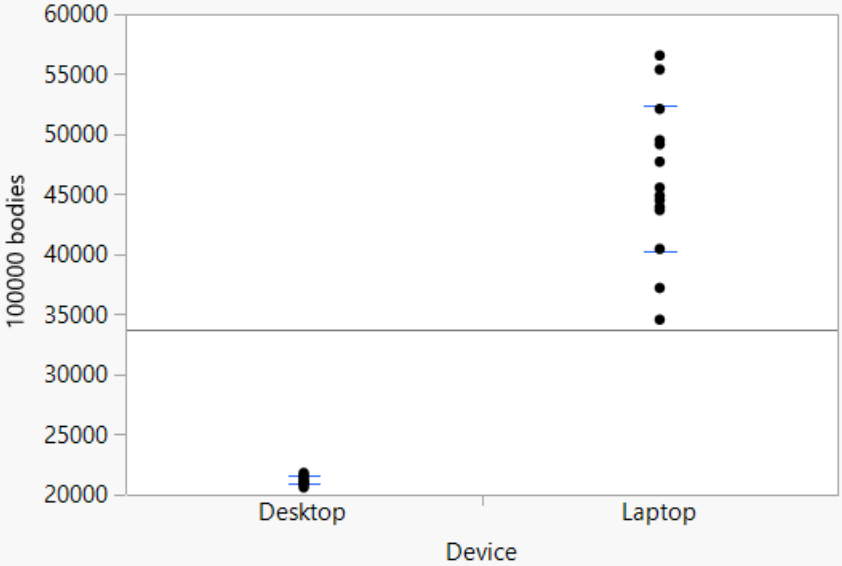
Anche in questo caso, il test di Shapiro-Wilk indica che le distribuzioni per 1000000 di corpi sono normali.

Per valutare l'effettivo test da utilizzare è necessario effettuare un **test di omoschedasticità**.

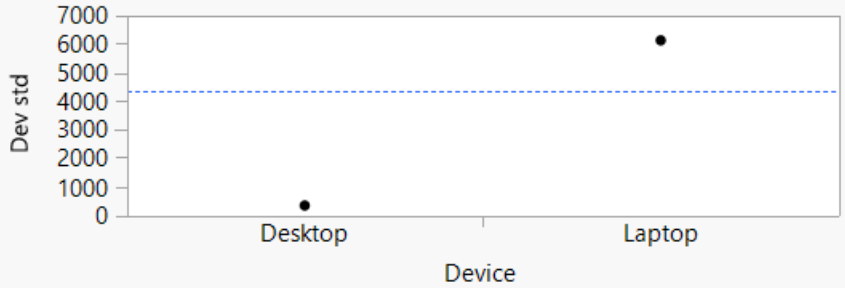
Test di omoschedasticità

Avendo ottenuto che le distribuzioni per 100000 e per 1000000 di corpi sono normali, è necessario andarne a verificare l'omoschedasticità, in modo da poter individuare quale test sia più adatto:

Analisi a una via di 100000 bodies per Device

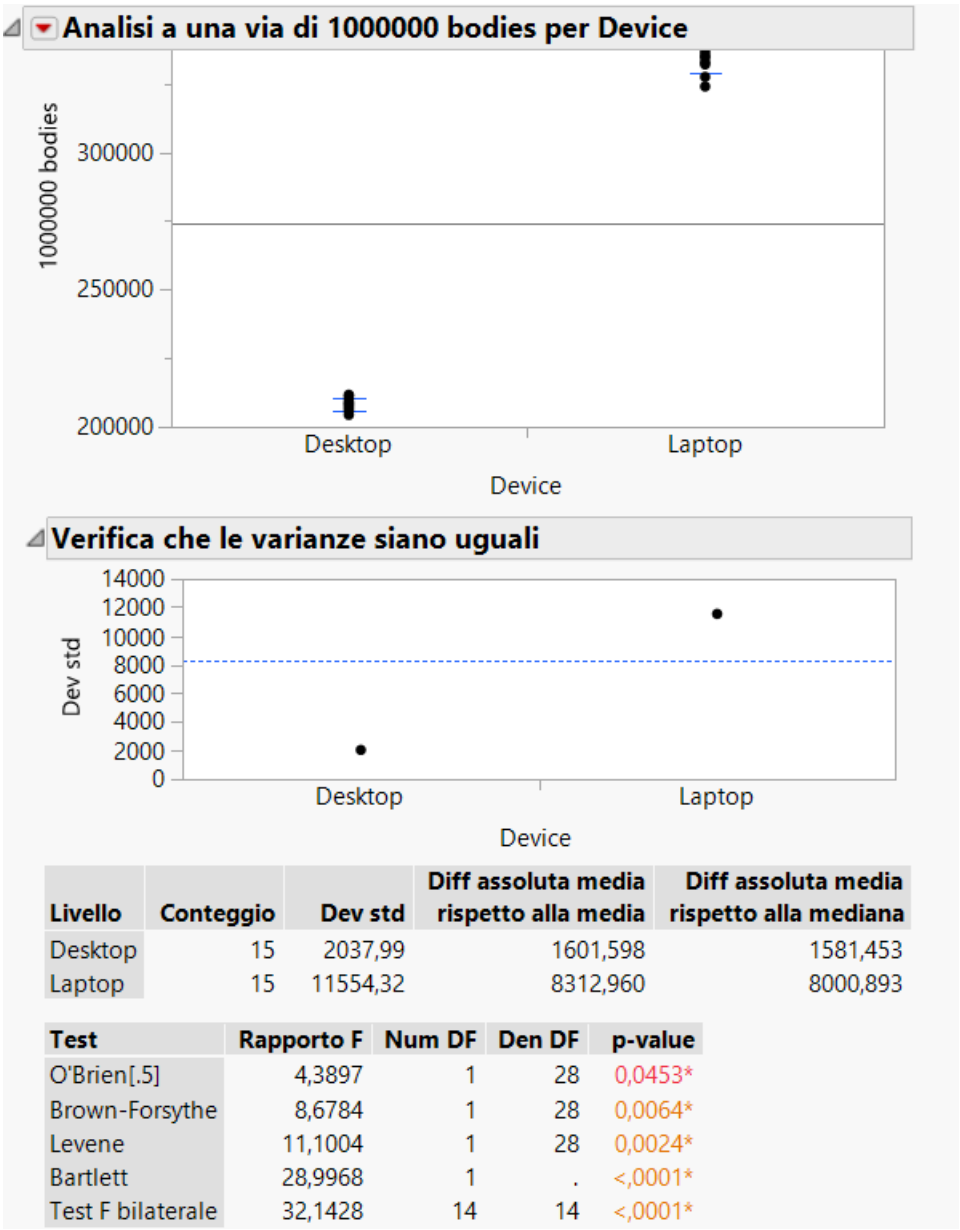


Verifica che le varianze siano uguali



Livello	Conteggio	Dev std	Diff assoluta media rispetto alla media	Diff assoluta media rispetto alla mediana
Desktop	15	351,391	265,556	263,253
Laptop	15	6125,774	4766,674	4715,707

Test	Rapporto F	Num DF	Den DF	p-value
O'Brien[.5]	8,4663	1	28	0,0070*
Brown-Forsythe	20,6855	1	28	<,0001*
Levene	22,9725	1	28	<,0001*
Bartlett	58,6243	1	.	<,0001*
Test F bilaterale	303,9077	14	14	<,0001*



Livello	Conteggio	Dev std	Diff assoluta media rispetto alla media	Diff assoluta media rispetto alla mediana
Desktop	15	2037,99	1601,598	1581,453
Laptop	15	11554,32	8312,960	8000,893

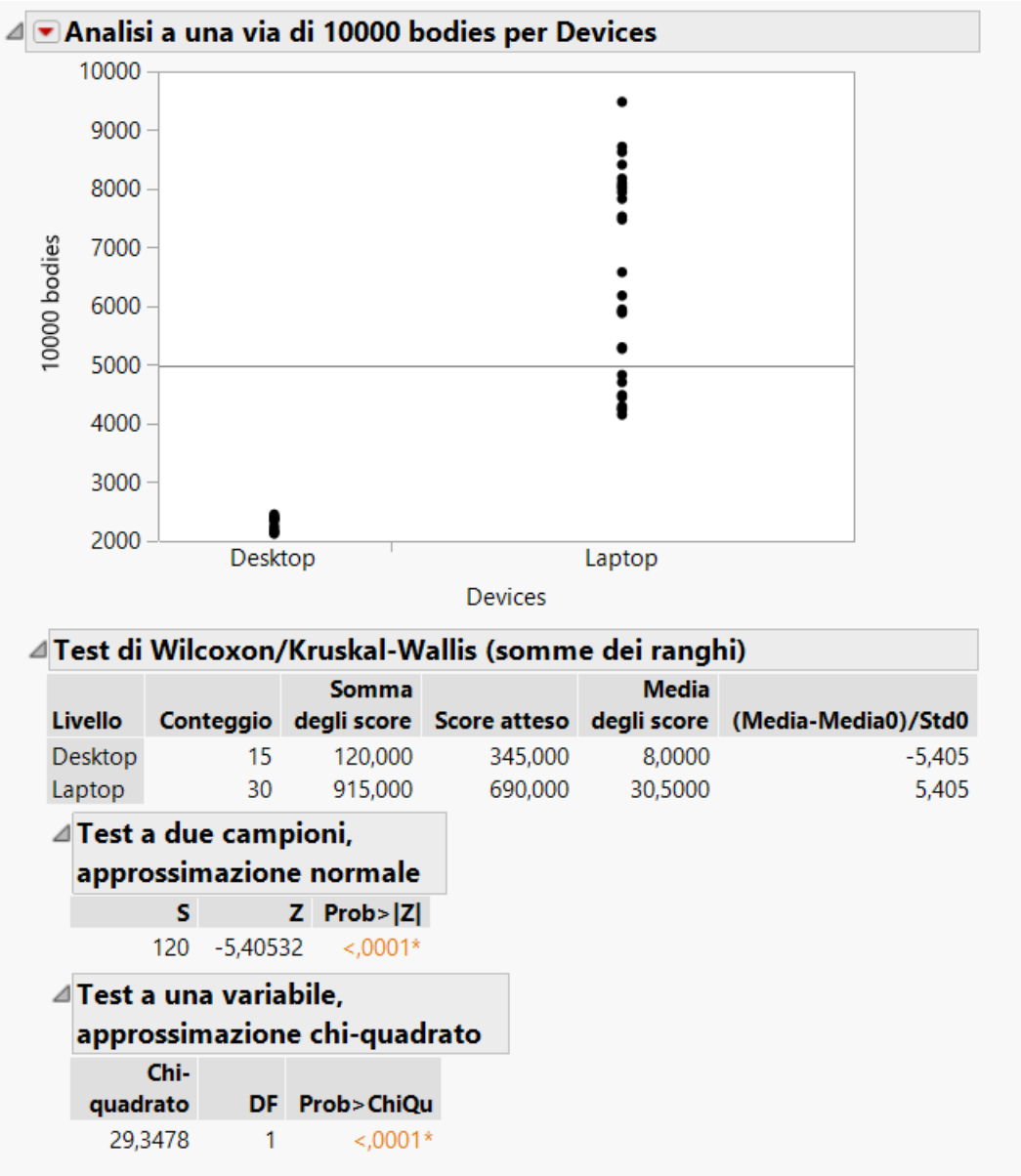
Test	Rapporto F	Num DF	Den DF	p-value
O'Brien[.5]	4,3897	1	28	0,0453*
Brown-Forsythe	8,6784	1	28	0,0064*
Levene	11,1004	1	28	0,0024*
Bartlett	28,9968	1	.	<,0001*
Test F bilaterale	32,1428	14	14	<,0001*

Si può osservare che, sia visivamente che tramite gli opportuni test riportati, i dati dei 100000 corpi e quelli per il milione di corpi **non sono omoschedastici**. Per loro sarà quindi opportuno effettuare un test two-sample **unpooled**.

La discussione sull'omoschedasticità dei dati per il benchmark a 10000 corpi è trattata nel paragrafo successivo per completezza.

Test di Wilcoxon per 10000 bodies

Data la non-normalità dei dati per le distribuzioni a 10000 corpi, è stato utilizzato **il test non parametrico di Wilcoxon**:



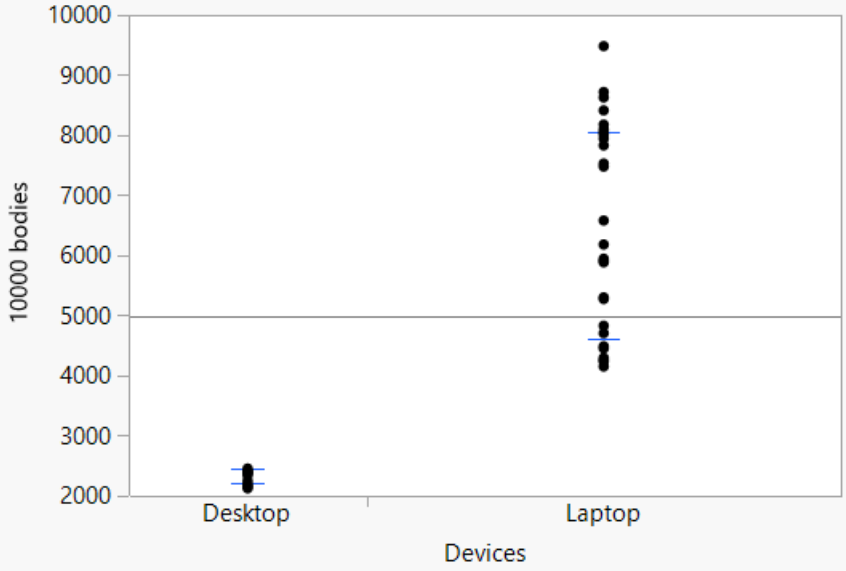
Si nota quindi che il p-value è minore di 0.05, per cui è possibile affermare che c'è una differenza significativa nelle due distribuzioni: questo risultato supporta dunque quanto già ottenuto a partire dal test visivo, e conferma una differenza sostanziale dei due device dal punto di vista prestazionale.

Two sample t-test unpooled per 10000 bodies

Nonostante le distribuzioni non siano normali, è in realtà comunque possibile applicare un two sample t-test per 10000 bodies: si ha infatti che il numero di campioni utilizzati per Laptop (n_1) è pari a 30, mentre quello per Desktop (n_2) è pari a 15. La loro somma $n_1 + n_2 = 45$ è quindi maggiore di 40, per cui viene rispettata la condizione alternativa alla normalità per l'utilizzo del t-test.

Si valuta quindi anche in questo caso l'omoschedasticità per determinare se il test debba essere pooled o unpooled, e, una volta osservato che le varianze sono ineguali, si procede con il test unpooled:

Analisi a una via di 10000 bodies per Devices

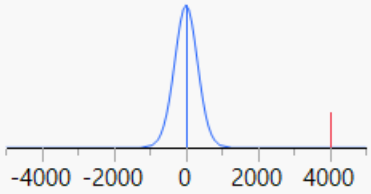


Test t

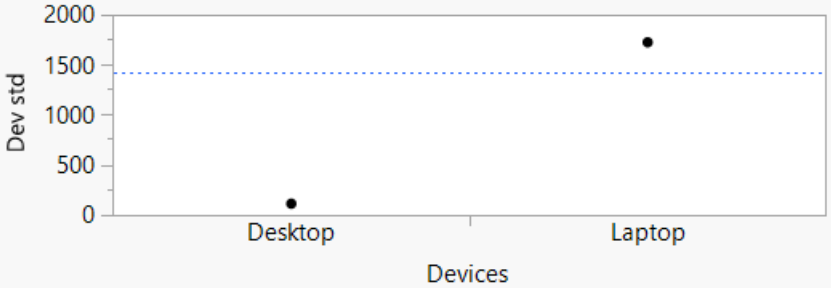
Laptop-Desktop

Assumendo varianze ineguali

Differenza	4002,05	Rapporto T	12,65891
Diff err std	316,14	DF	29,49323
Diff CL superiore	4648,17	Prob > |t|	<,0001*
Diff CL inferiore	3355,93	Prob > t	<,0001*
Confidenza	0,95	Prob < t	1,0000



Verifica che le varianze siano uguali



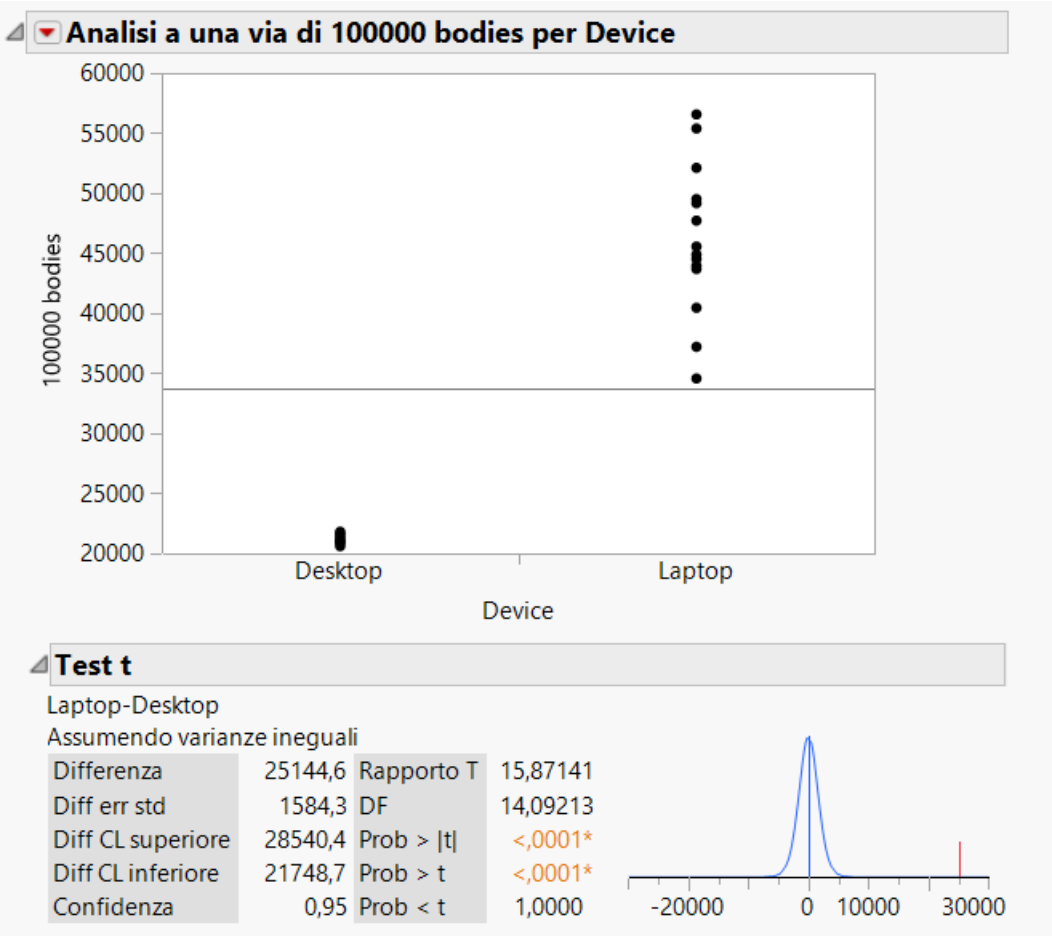
Livello	Conteggio	Dev std	Diff assoluta media rispetto alla media	Diff assoluta media rispetto alla mediana
Desktop	15	112,700	100,101	92,333
Laptop	30	1724,245	1521,710	1504,140

Test	Rapporto F	Num DF	Den DF	p-value
O'Brien[.5]	24,2306	1	43	<,0001*
Brown-Forsythe	42,0671	1	43	<,0001*
Levene	51,7977	1	43	<,0001*
Bartlett	57,9334	1	.	<,0001*
Test F bilaterale	234,0735	29	14	<,0001*

Il risultato ottenuto è coerente con quanto osservato con il test di Wilcoxon e conferma che le distribuzioni provengono da popolazioni differenti.

Two sample t-test unpooled per 100000 bodies

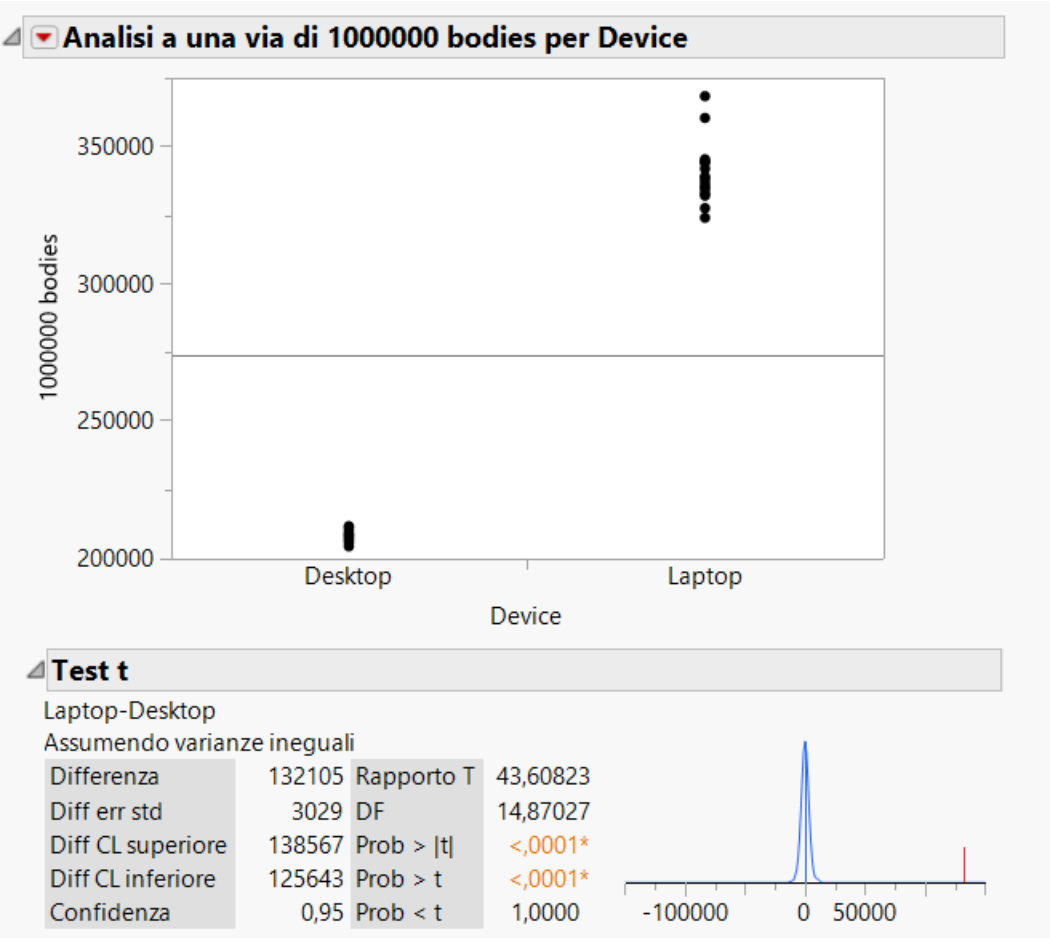
Avendo verificato l'eteroschedasticità si ottiene che il test di ipotesi da effettuare nel caso di 100000 bodies è il **two sample t-test unpooled** (caso popolazione normale, osservazioni indipendenti e $\sigma_1 \neq \sigma_2$ entrambe sconosciute). Si riporta di seguito il risultato ottenuto:



Il risultato ottenuto conferma che le distribuzioni provengono da popolazioni differenti.

Two sample t-test unpooled per 1000000 bodies

Anche in questo caso il test di ipotesi da effettuare è il **two sample t-test unpooled**: si riporta di seguito il risultato ottenuto:



Il risultato ottenuto conferma che le distribuzioni provengono da popolazioni differenti.

Conclusione

Analizzando i risultati ottenuti a valle dell'analisi, è possibile affermare che **le macchine sono diverse da un punto di vista di prestazioni**.

Questo non è sorprendente, vista la considerevole differenza hardware tra i due sistemi: le due macchine hanno processori nettamente diversi, tra l'altro di due **generazioni diverse**, che contribuisce ad accentuare le differenze in termini prestazionali.

Un'eventuale accentuazione della differenza di performance è chiaramente anche l'**architettura** dei processori, uno per laptop e l'altro per desktop, che lavorano quindi anche a diversi range di consumo energetico e quindi di efficienza di calcolo (chiaramente andando a pagarlo con maggiori consumi per unità di tempo).

FFDA

Mercury

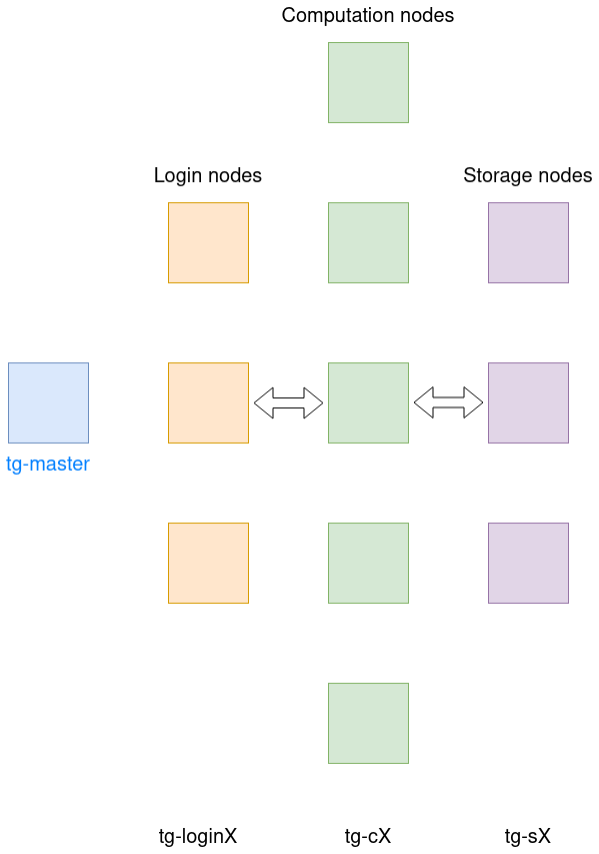
Il **cluster Mercury** è un cluster costituito da nodi IBM.

In particolare, come si può vedere dalla figura riportata lateralmente, il cluster è composto da un architettura a **3 livelli**, più un nodo di **gestione** chiamato **tg-master**.

I livelli dell'architettura del cluster sono:

- **Layer di login:** nodi *tg-loginX*
- **Layer di calcolo:** nodi *tg-cX*
- **Layer di storage:** nodi *tg-sX*

Il valore X è un numero intero che corrisponde all'identificativo del nodo all'interno del suo layer. Le interconnessioni tra i layer e i nodi sono gestite da un'architettura **Myrinet**, e ogni nodo ha in esecuzione un sistema con sistema operativo **RedHat 9.0**.



Architettura schematica di Mercury

I valori che verranno analizzati sono stati ottenuti tramite il ***syslog* daemon**. Inoltre, il log è stato **filtrato** riportando quindi voci **anonime** e che comprendono solo **errori fatali**. In particolare, si avrà a che fare con **6 categorie principali di errore:**

- gli errori **DEV** dei dispositivi connessi al sistema (ad esempio, fallimenti di dispositivi PCI)
- gli errori **MEM** che indicano errori sui dispositivi di memoria (come la RAM)
- gli errori **NET**, che sono errori di connessione (generalmente **connection down**)
- gli errori **I-O**, che sono generalmente errori di lettura/scrittura di **storage**
- gli errori **PRO**, che sono errori legati ai processori

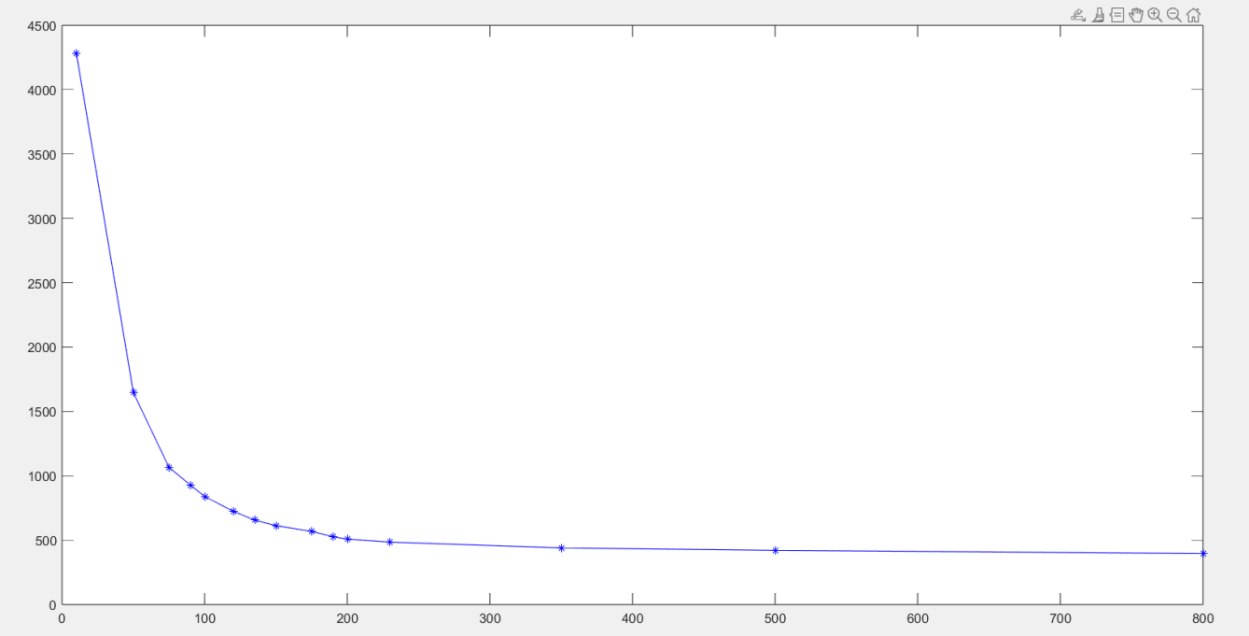
- tutti gli altri errori, raggruppati sotto la categoria **OTH**

Sensitivity Analysis

Il primo passo da fare nell'analisi è quello di trovare una **Coalescence Window** per le voci nel log. La dimensione della coalescence window va cercata in modo da ottimizzare il **raggruppamento di fallimenti** nelle stesse finestre temporali.

Lo scopo è quello di raggruppare tutte le voci che possono essere collegate allo stesso **problema**, visto che questo può presentarsi molteplici volte e può propagarsi con più errori diversi, e spesso su più categorie.

Per fare ciò, è stato utilizzato lo script *tupleCount_func_CWIN.sh*, che data una lista di **valori di tentativo** per la dimensione della coalescing window (da ora in poi CWIN), effettua il processo di **tupling** per contare il numero di tuple (gruppi) che si genereranno con la dimensione di CWIN corrispondente. Da una serie di misurazioni si ottiene:



Per scegliere un valore adatto per la CWIN, una regola empirica che si può seguire è quella di scegliere un punto **subito dopo** la knee della curva. Seguendo questa regola, è stata scelta una CWIN di **200s**, ottenendo un numero totale di **508 tuple**.

Da questa scelta, è stato quindi eseguito lo script *tupling_with_CWIN.sh* specificando il valore scelto al passo precedente, ed è stato quindi ottenuto il log diviso in tuple, insieme ai **tempi di inter-arrivo** tra una tupla e l'altra, le **lunghezze di tupla** e gli **starting point**.

Tramite uno script Python costruito per lo scopo, sono stati inoltre calcolati i valori di **troncamenti** e **collisioni** per il tupling effettuato. Si è quindi trovato che, con una dimensione temporale di 200s, si hanno:

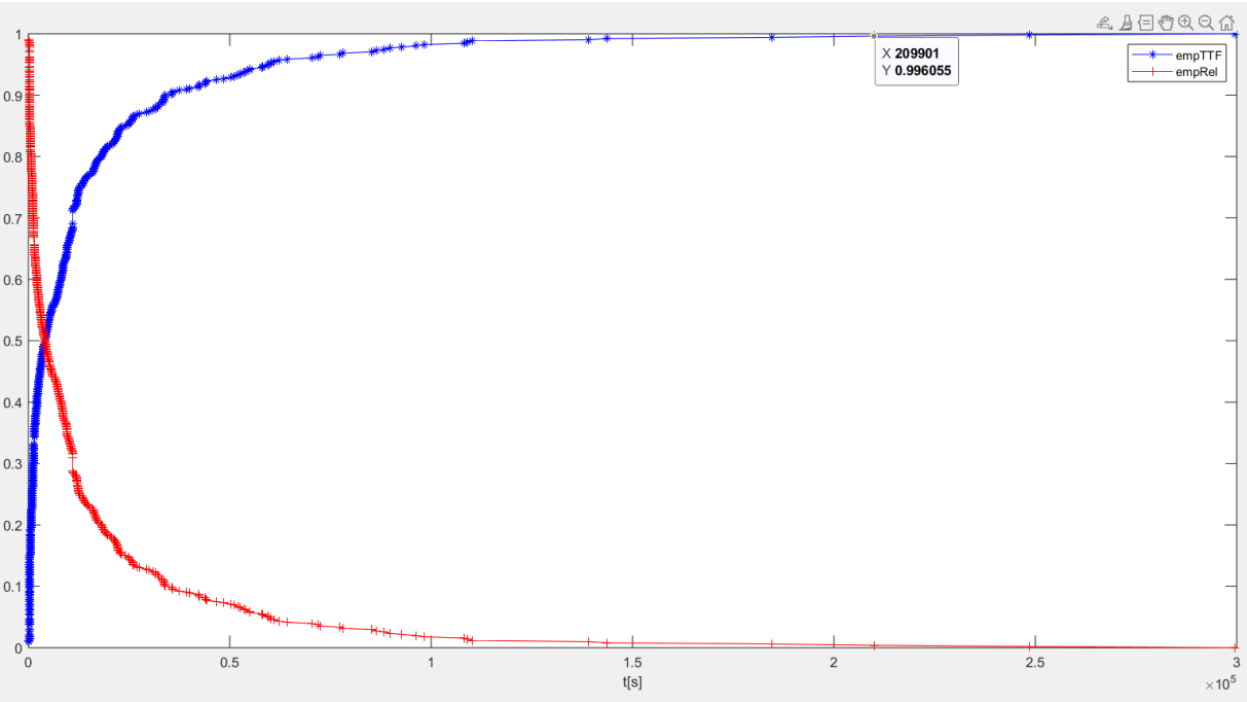
- **Troncamenti:** 233 (~45.87% delle tuple)
- **Collisioni:** 47 (~9.25% delle tuple)

Nonostante il numero di troncamenti sembri elevato, si osserva che una dimensione più ampia della finestra non fornirebbe ulteriori benefici, in quanto il numero di tuple rimarrebbe sostanzialmente lo stesso. Inoltre, scegliendo valori minori per la finestra, aumenterebbero il numero di troncamenti, sebbene diminuiscano il numero di collisioni; possiamo considerare questa scelta come il miglior compromesso ottenibile tra collisioni e troncamenti.

Reliability e TTF

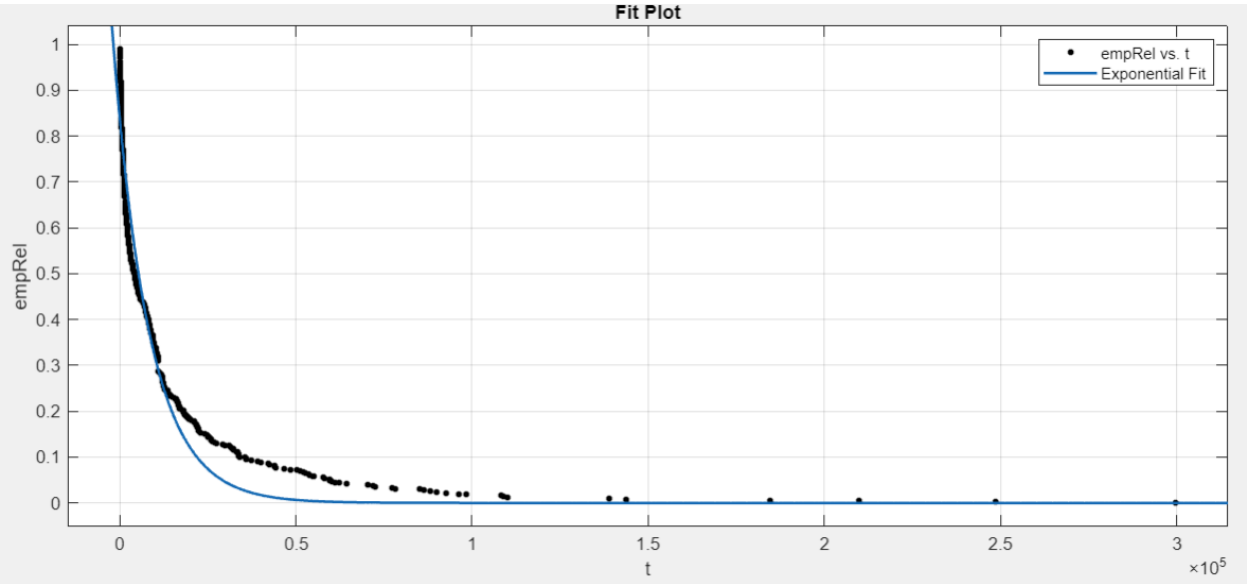
È quindi possibile stimare una forma per la **reliability** del sistema, intesa come la **probabilità che un sistema non fallisca** in un certo intervallo di tempo. Avendo i **tempi di inter-arrivo** precedentemente calcolati col tupling, è possibile quindi costruire una **Reliability empirica**, corredata del suo opposto: il **Time to Failure empirico**.

Si ottengono questi valori tramite degli **script Matlab**:



Per cui sarebbe utile calcolare un **modello regressivo** che si adatti al meglio a queste stime empiriche. Per fare ciò, è stato utilizzato il **cftool** di Matlab, su cui sono stati testati tre modelli diversi di regressione: **esponenziale**, **iper-esponenziale** e **funzione di Weibull**.

Di seguito si riportano i grafici dei modelli di regressione ottenuti tramite il tool di Matlab:

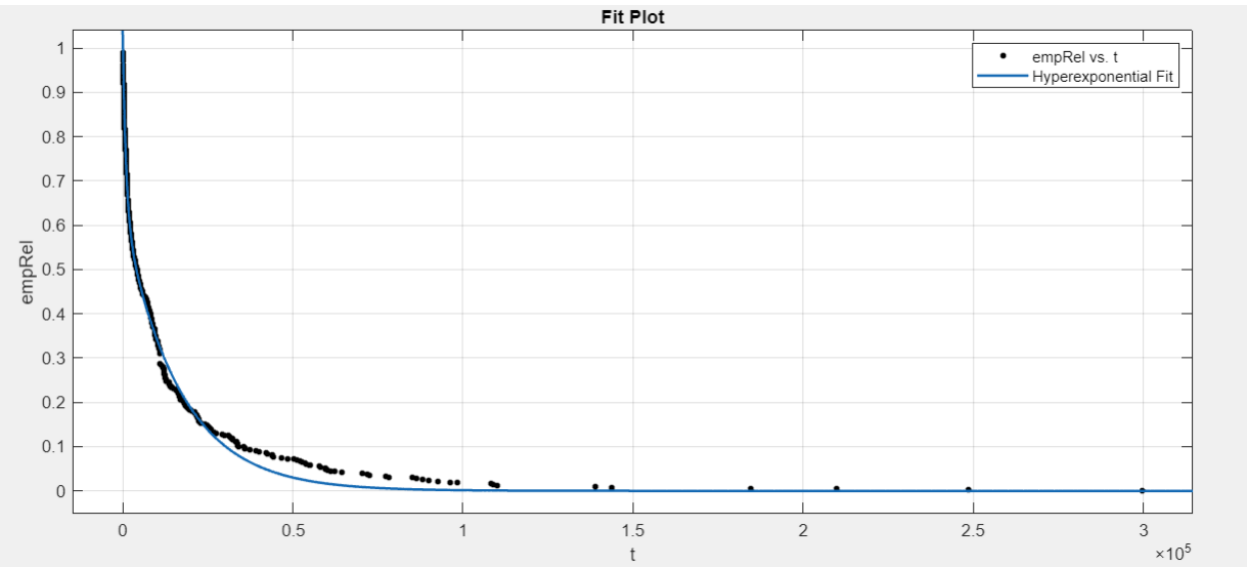


Questo modello si basa sul fitting di una funzione esponenziale di questo tipo:

$$f(x) = \begin{cases} ae^{bx} & \text{se } x \geq 0 \\ 0 & \text{se } x < 0 \end{cases}$$

dove **a** e **b** sono i **parametri da stimare**. La stima è stata effettuata automaticamente dal tool tramite il **Trust-Region algorithm**, e di seguito vengono riportati i parametri con i loro intervalli di confidenza al 95%:

	Valore	Limite inferiore	Limite superiore
a	0.8386	0.8283	0.8489
b	-0.0001	-0.0001	-0.0001

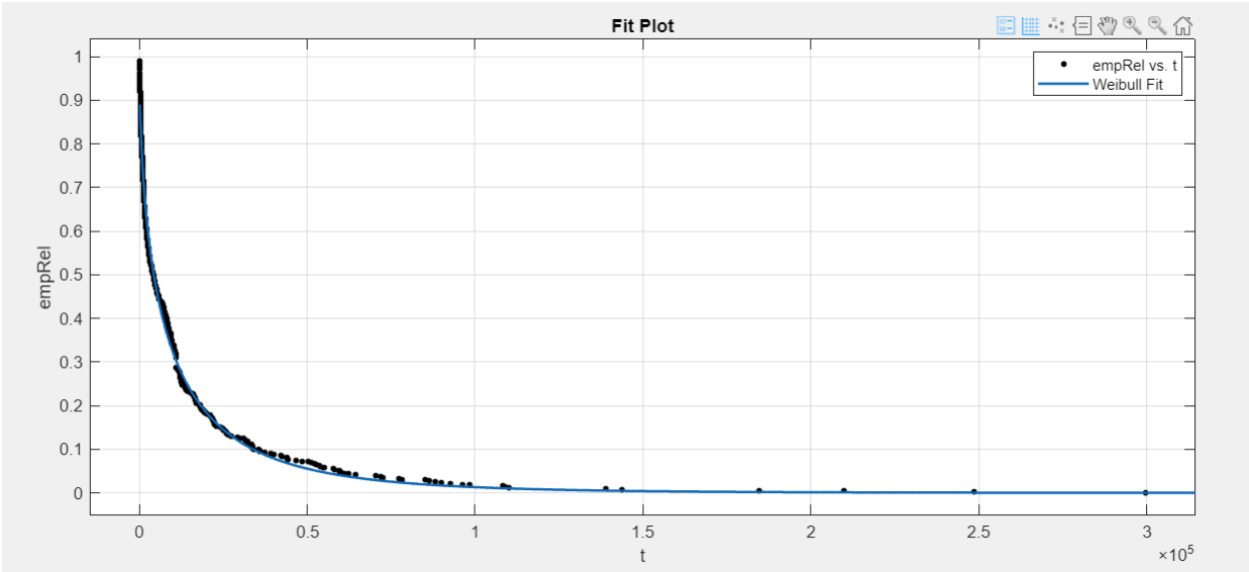


Questo modello si basa sul fitting di una funzione iper-esponenziale:

$$f(x) = \begin{cases} ae^{bx} + ce^{dx} & \text{se } x \geq 0 \\ 0 & \text{se } x < 0 \end{cases}$$

Con lo stesso setup del fit precedente, sono stati ottenuti i seguenti valori:

	Valore	Limite inferiore	Limite superiore
a	0.3922	0.3824	0.4020
b	-0.0010	-0.0011	-0.0010
c	0.6297	0.6207	0.6388
d	-0.0001	-0.0001	-0.0001



Questo modello si basa sul fitting di una funzione di Weibull:

$$f(x) = \begin{cases} e^{-bx^a} & \text{se } x \geq 0 \\ 0 & \text{se } x < 0 \end{cases}$$

Per cui, nuovamente, sono stati ottenuti i seguenti valori:

	Valore	Limite inferiore	Limite superiore
a	0.5823	0.5748	0.5899
b	0.0053	0.0050	0.0056

Da questi tre modelli sono stati riportati i valori di R^2 e di **varianza persa** nella regressione:

Modello	R-quadro	R-quadro corretto	SSE
Esponenziale	0.9543	0.9533	1.7471
Iper-esponenziale	0.99612	0.99609	0.14572
Weibul	0.99179	0.99177	0.3079

Sia visivamente che dai valori raccolti per R^2 e SSE, si può già dire che il **modello esponenziale non è adatto** al fitting della Reliability empirica, mentre sembrano già più consistenti gli altri due modelli. Per capire qual è meglio scegliere, si può effettuare un **test di Kolmogorov-Smirnov** per capire quale funzione si presta meglio a rappresentare i dati empirici. Tramite Matlab, sono stati effettuati questi test, ottenendo:

Modello	h	p-value	k
Esponenziale	1	3.3444*10^-5	0.1524
Iper-esponenziale	0	0.6666	0.0427
Weibul	0	0.0724	0.0837

Come c'era da aspettarselo, il test **rigetta l'ipotesi nulla** nel caso del fit esponenziale, ma al contrario non è possibile rigettare l'ipotesi nulla negli altri due casi. È stato quindi scelto di utilizzare il modello con **p-value**

maggiore, in quanto massimizza la probabilità di non rigettare l'ipotesi nulla. Per cui, la funzione scelta come rappresentativa della Reliability è quella iper-esponenziale:

$$R_{\text{Mercury}}(t) = \begin{cases} 0.3922e^{-0.001x} + 0.6297e^{-0.0001x} & \text{se } t \geq 0 \\ 0 & \text{se } t < 0 \end{cases}$$

Si può quindi pensare di calcolare il **Mean Time To Failure MTTF** come:

$$MTTF = \int_0^{+\infty} R_{\text{Mercury}}(t)dt = 6689.2s$$

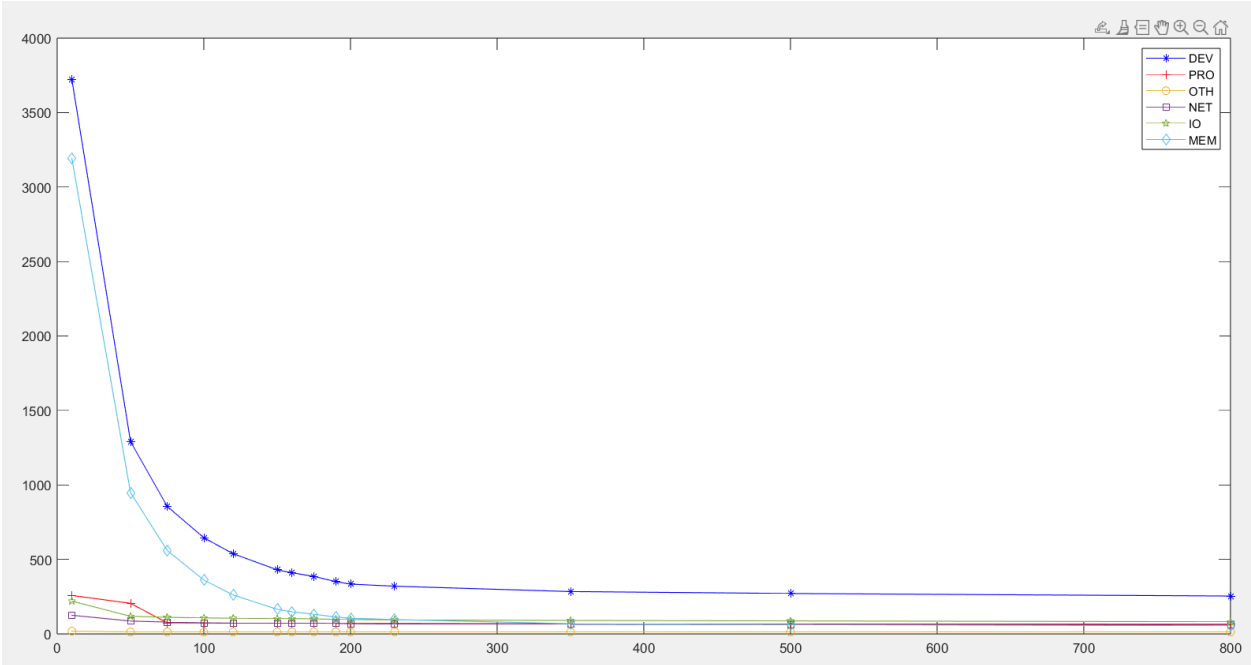
e quindi mediamente il sistema fallirà dopo **1 ora e 50 minuti** circa.

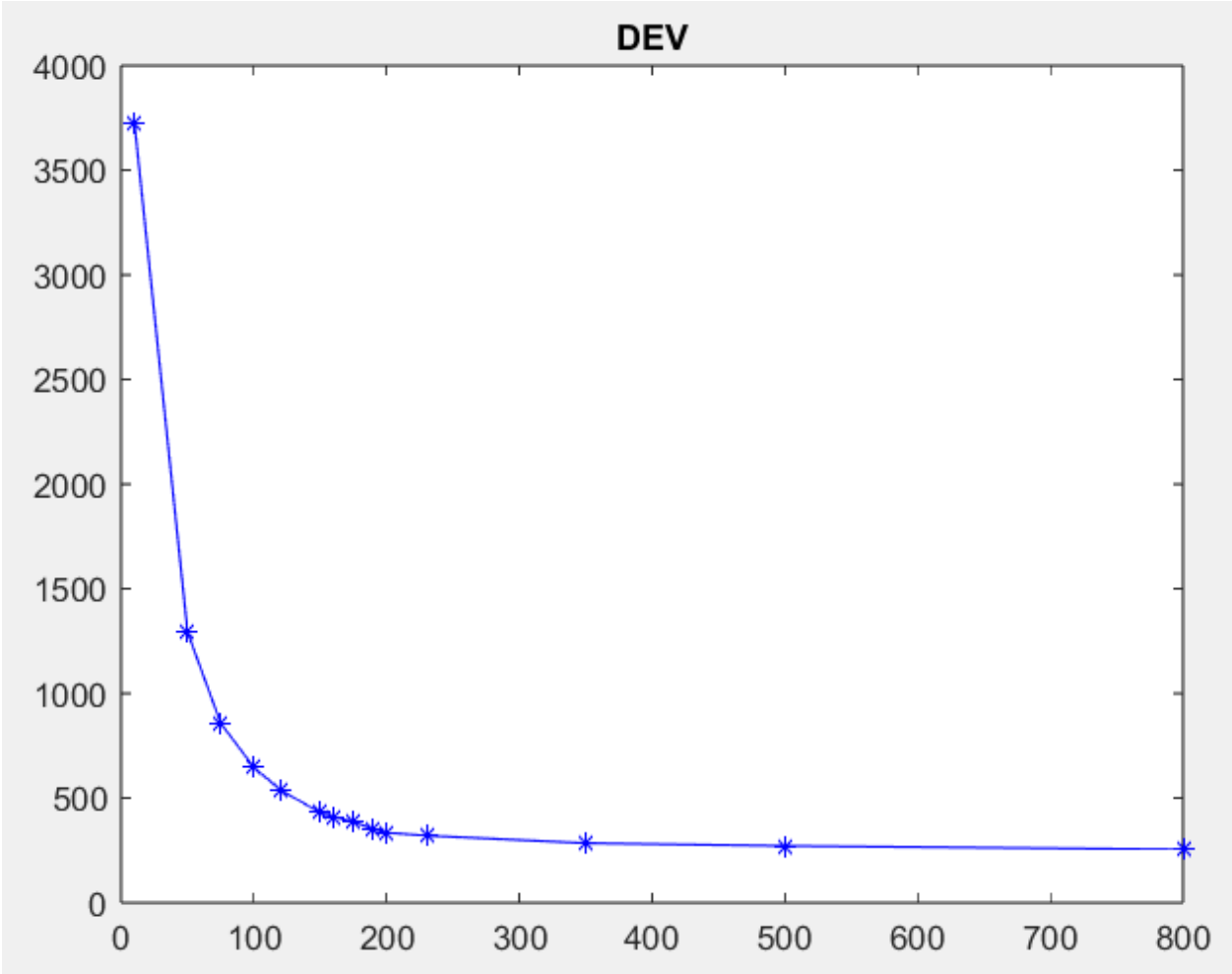
Avendo effettuato un'analisi di sensitività globale, si procederà ad effettuarla anche **per categoria di errore** e **per nodo**. Per non doverla effettuare su tutte le categorie e tutti i nodi, è stato generato un report descrittivo (tramite *logStatistics.sh*) del log del cluster, in cui le categorie dei nodi sono riportate per **numero decrescente di errori**:

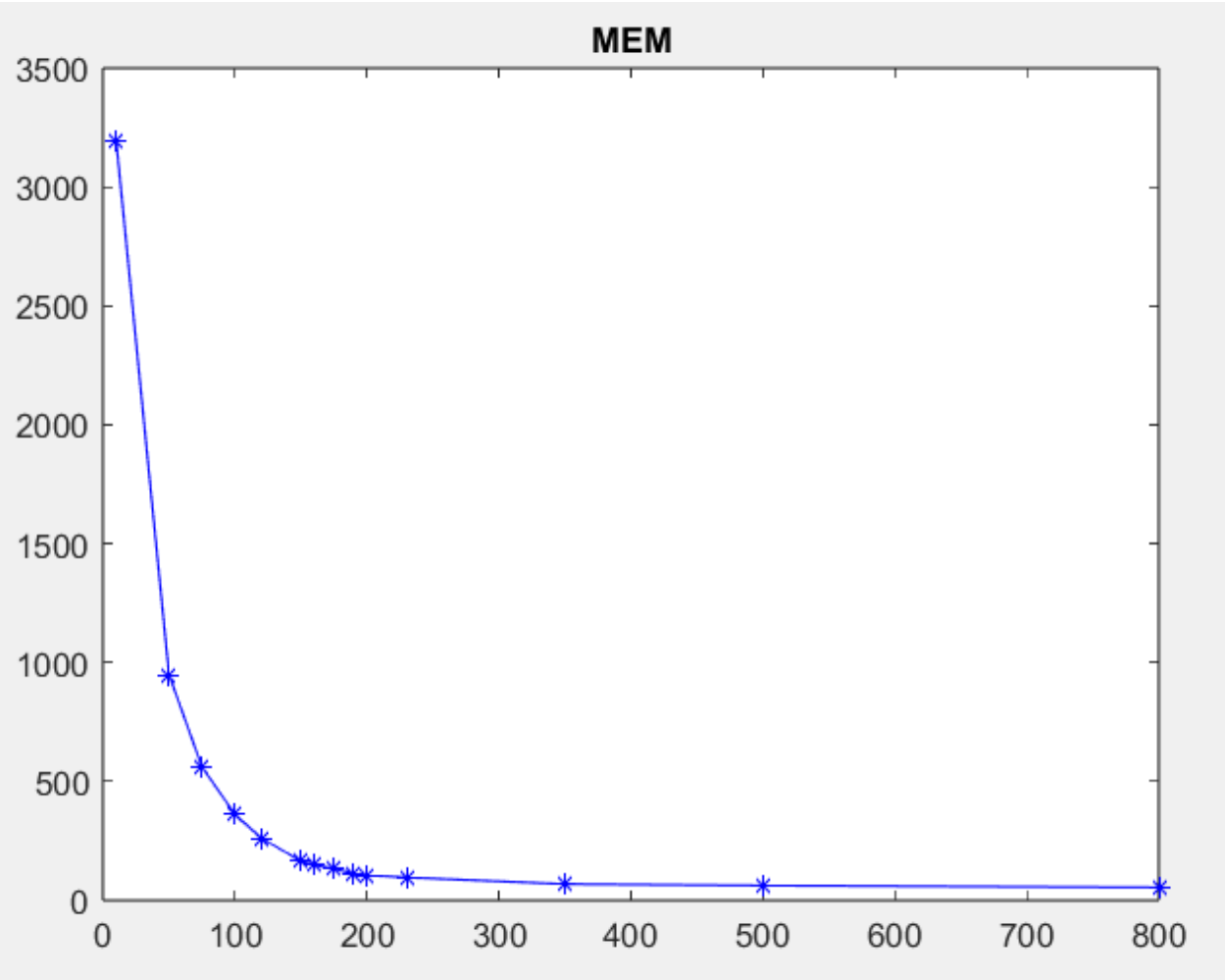
```
== Mercury Error Log ==

== Total error entries ==
80854

== Breakup by CATEGORY ==
DEV 57248
MEM 12819
I-O 5547
NET 3702
PRO 1504
OTH 34

== Breakup by NODE* ==
tg-c401 62340
tg-master 4098
tg-c572 4030
tg-s044 3224
tg-c238 1273
tg-c242 1067
tg-c648 643
tg-login3 382
tg-c117 268
tg-c669 267
  * only the 10 most occurring nodes are reported
```

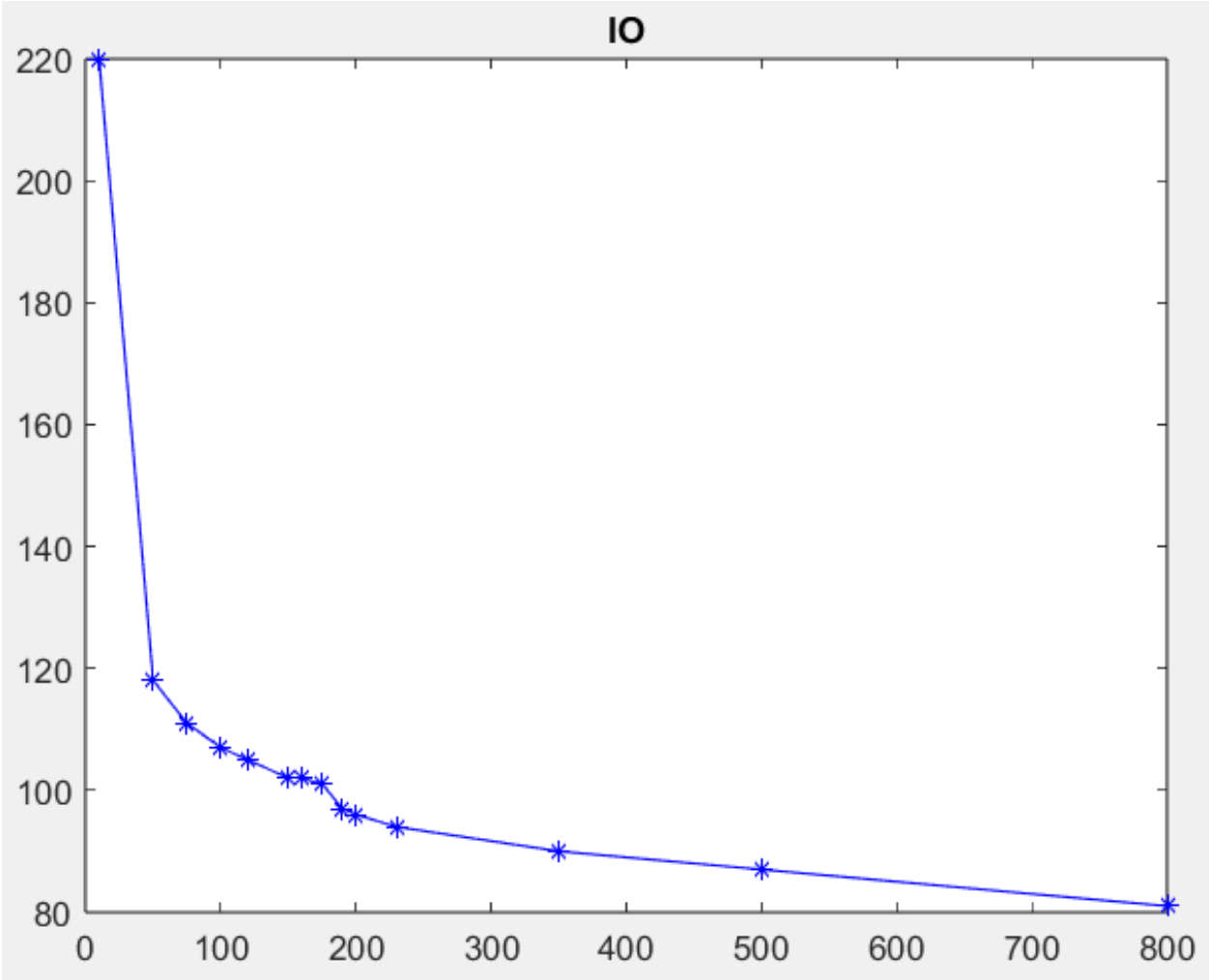
Sensitivity Analysis per categoria

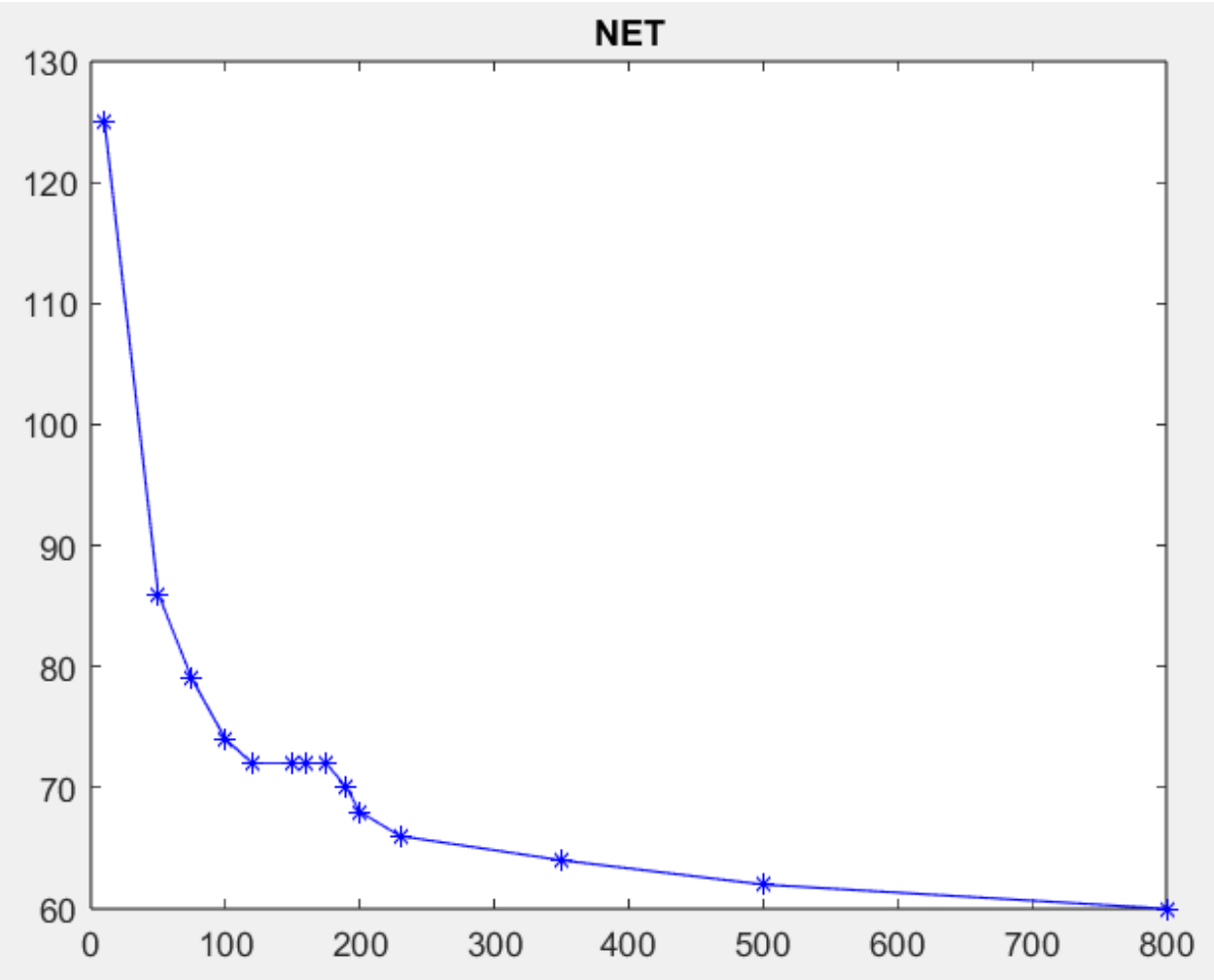
L'**analisi per categoria** è stata effettuata considerando il log riportato al passo precedente, ed effettuando un **filtraggio** tramite lo script *filter.sh* su **ogni categoria** riportata nel log. Per ognuna di queste, è stato effettuato un **tuple Count** come nel passo precedente è stato fatto per l'intero log, in modo da poter stimare le **CWIN** per ogni categoria. I dati ottenuti sono stati riportati tramite grafici:

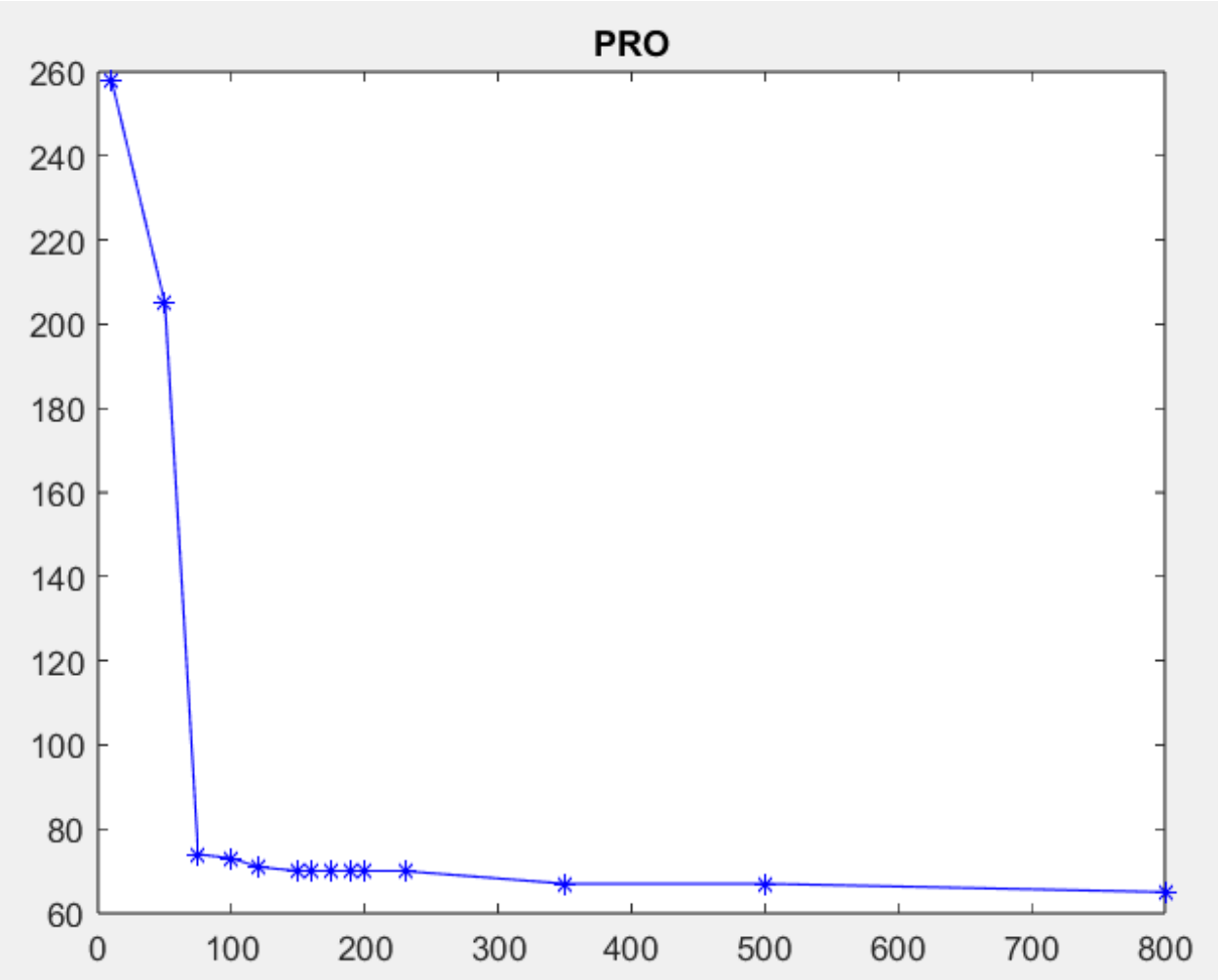

DEV
PRO
OTH
NET
IO
MEM
4000
3500
3000
2500
2000
1500
1000
500
0
0
100
200
300
400
500
600
700
800

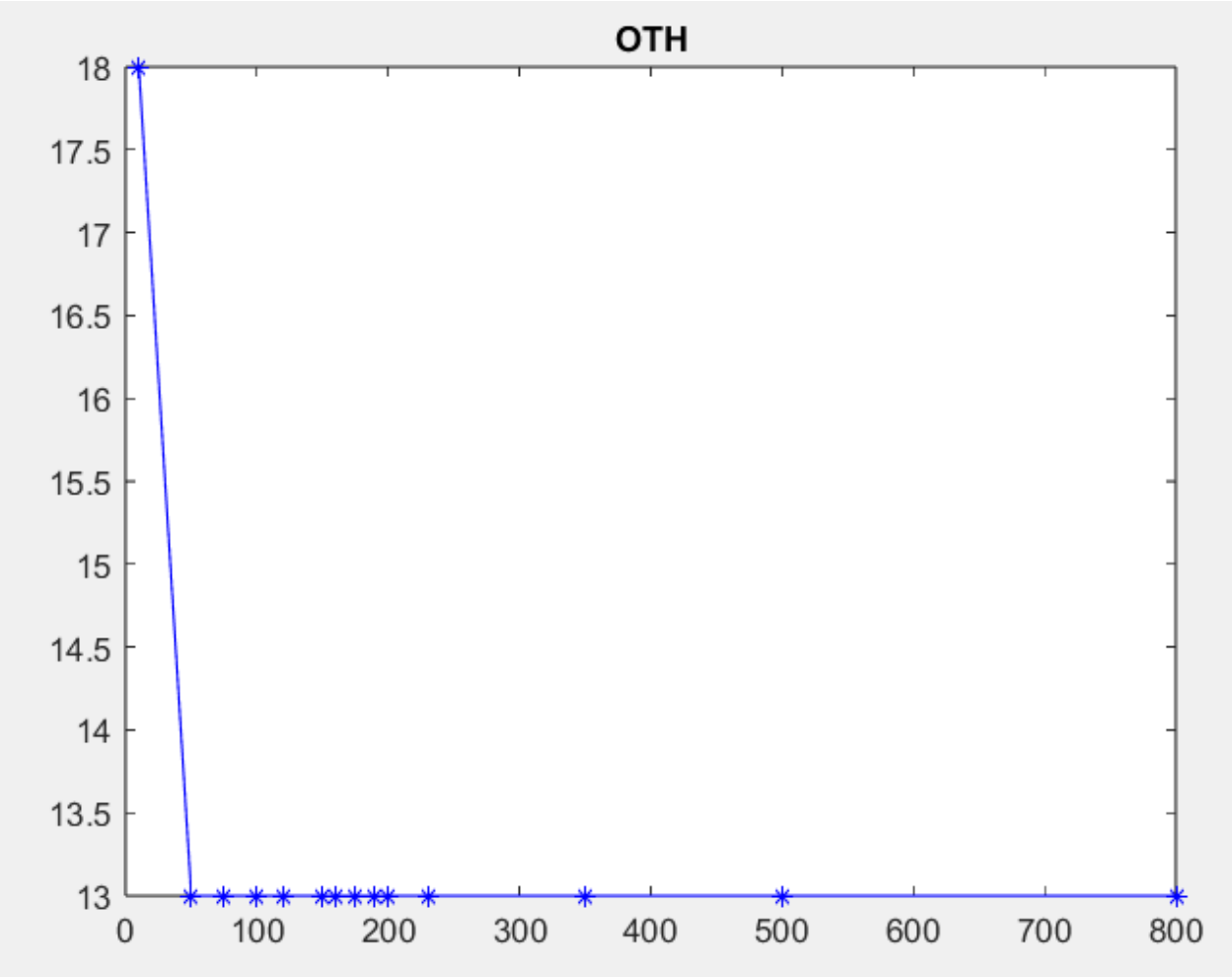

DEV
4000
3500
3000
2500
2000
1500
1000
500
0
0
100
200
300
400
500
600
700
800


MEM
3500
3000
2500
2000
1500
1000
500
0
0
100
200
300
400
500
600
700
800


IO
220
200
180
160
140
120
100
80
0
100
200
300
400
500
600
700
800


NET
130
120
110
100
90
80
70
60
0
100
200
300
400
500
600
700
800


PRO
260
240
220
200
180
160
140
120
100
80
60
0
100
200
300
400
500
600
700
800

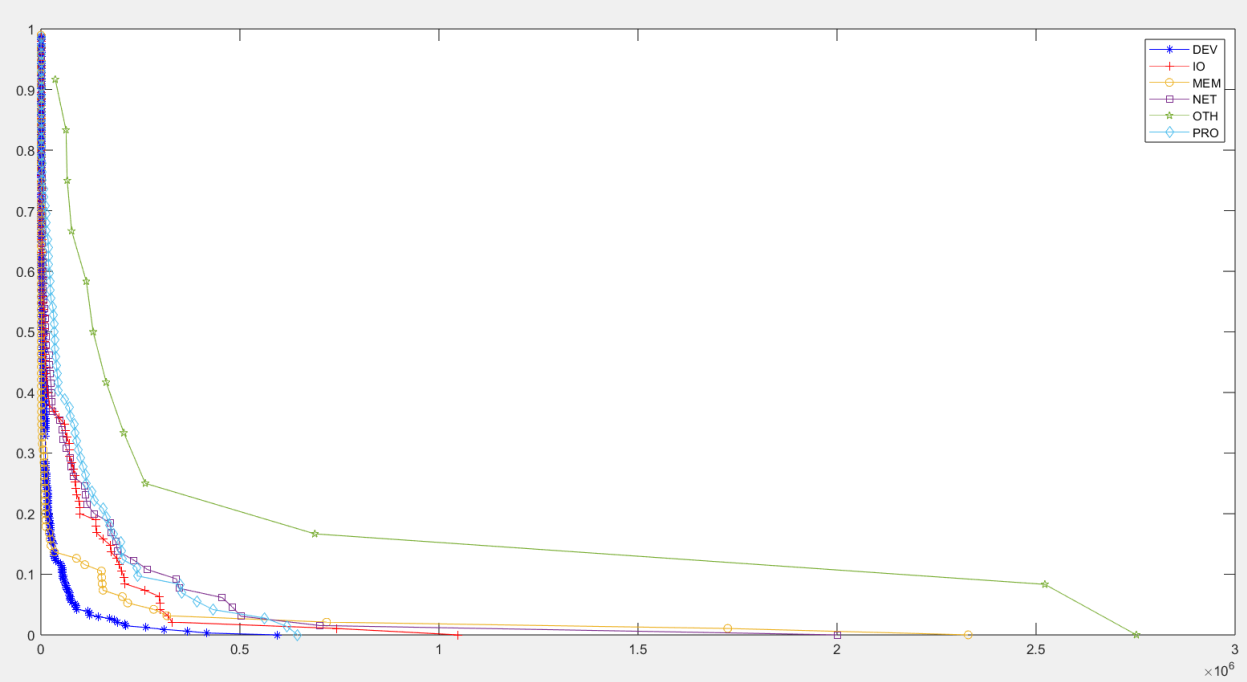


Node	CWIN
DEV	200
MEM	230
I-O	200
NET	230
PRO	100
OTH	75

Come è possibile vedere, la CWIN (scelta col metodo empirico definito precedentemente) ha una **mediana** di 200, che corrisponde con la scelta che è stata fatta per caratterizzare il tupling del sistema globale.

Da questi valori, è quindi possibile calcolare la **reliability empirica** per ogni categoria, in modo da poter fare delle considerazioni su quale di queste sia l'effettivo **bottleneck** del sistema.

Si riporta quindi il grafico delle reliability:

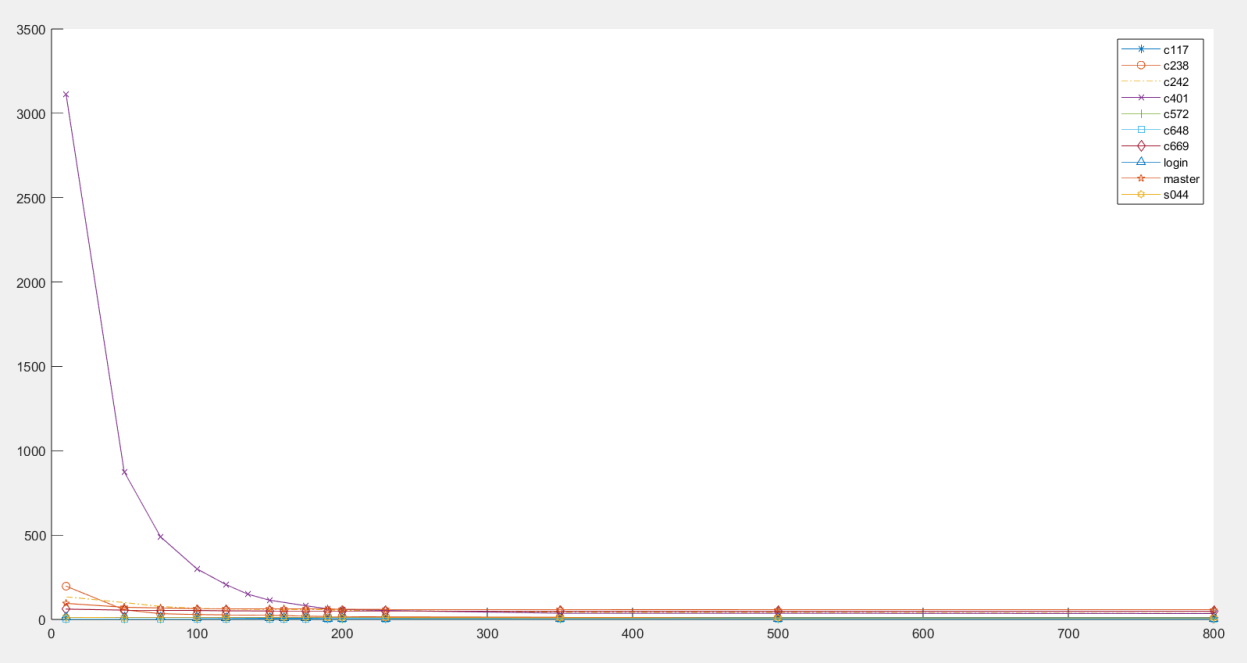


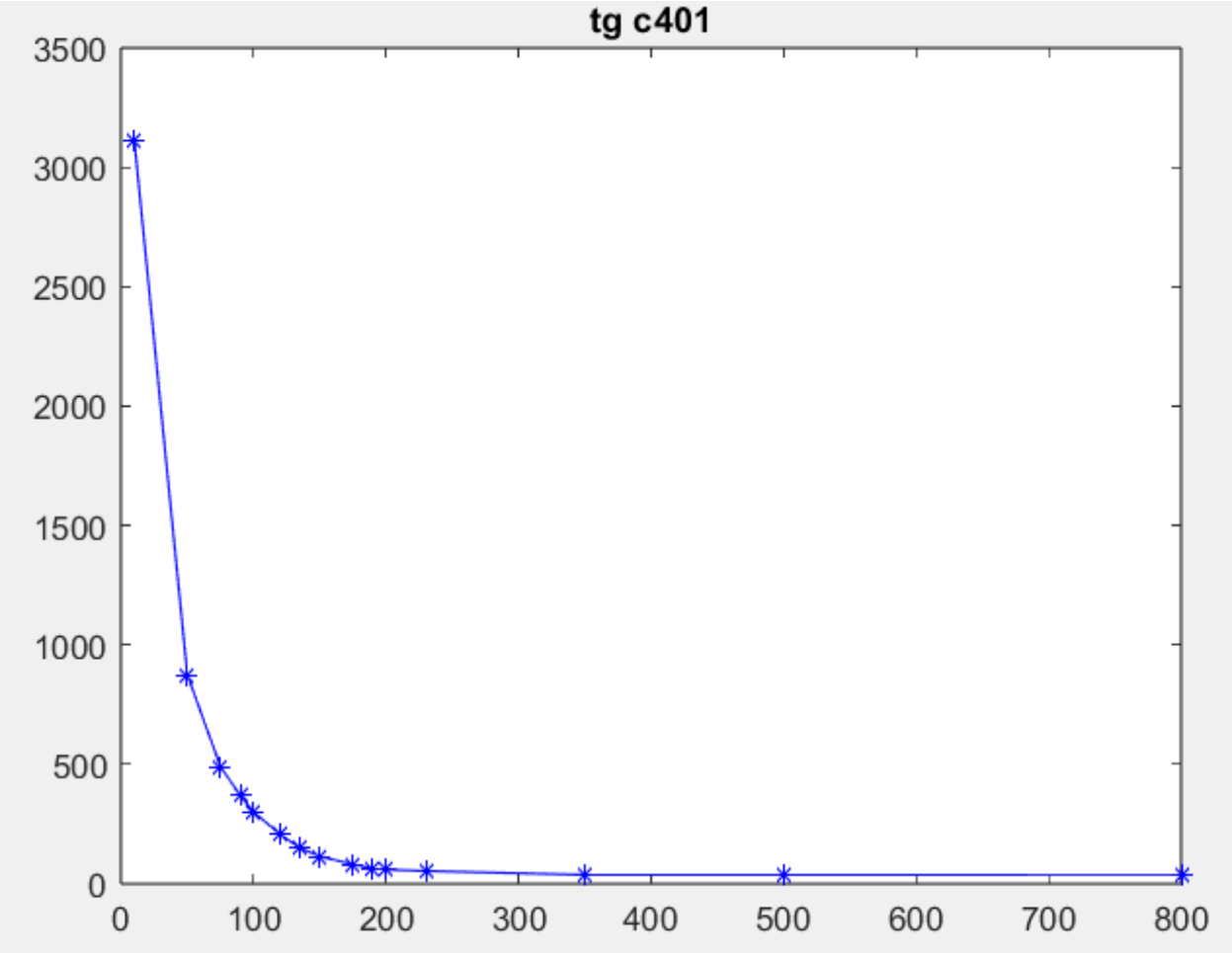
Visivamente, la categoria più **unreliable** sembra essere quella dei failure relativi ai **dispositivi** (la curva DEV). Sembrano poi venire, in ordine, la categoria **MEM** e la categoria **I-O**, mentre le categorie **NET** e **PRO** sono comparabili. Si può quindi dire che la categoria che contribuisce di più ai failure è proprio quella degli errori di dispositivi connessi al sistema, che potrebbe essere considerata il **bottleneck** di questo sistema, anche se visivamente non c'è un distacco netto rispetto alle altre.

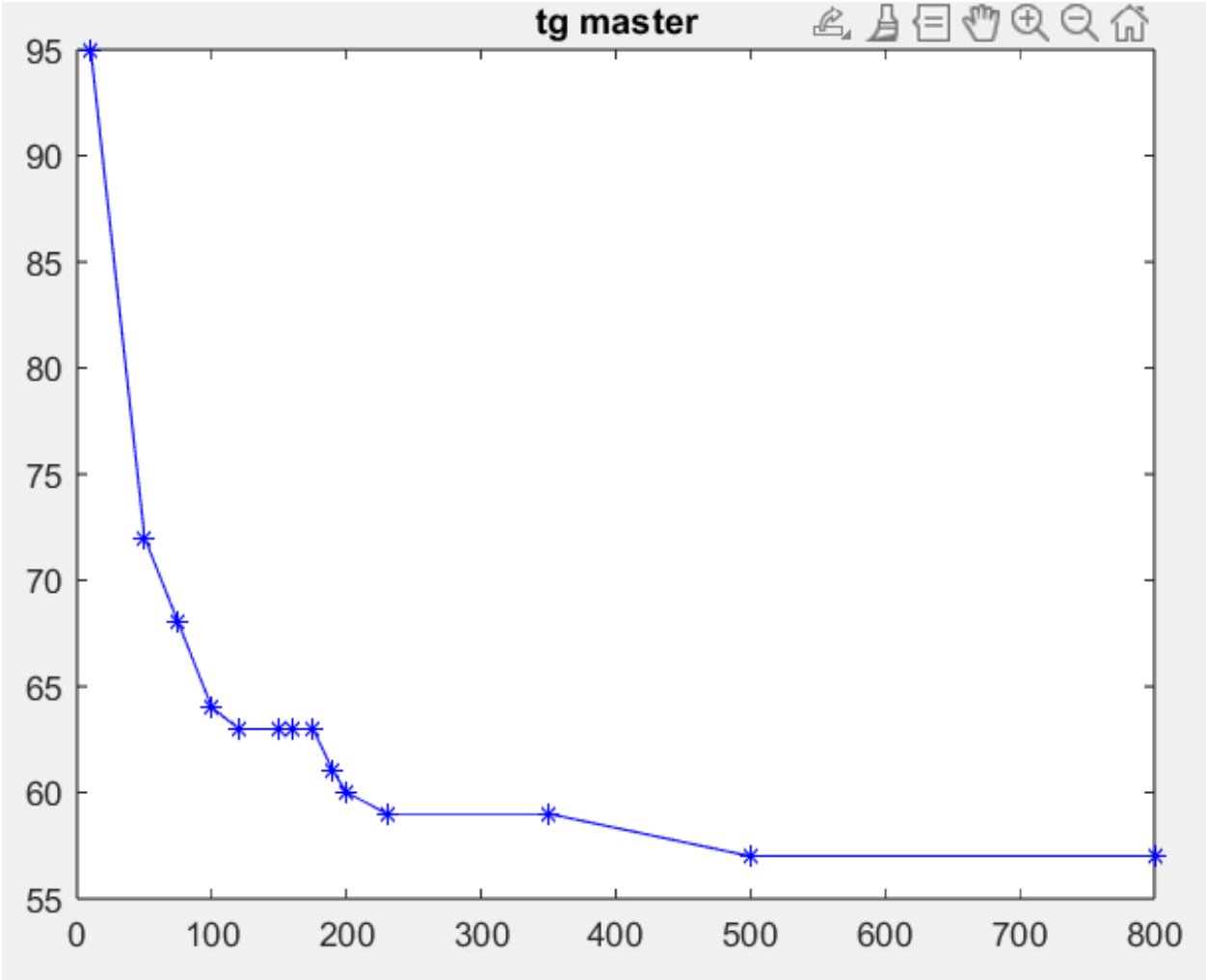
Per la categoria **OTH** forse non sono a disposizione sufficienti dati per ottenere una buona analisi di reliability, e inoltre e una categoria comprensiva di errori generici non specificamente schedati, per cui è difficile fare considerazioni.

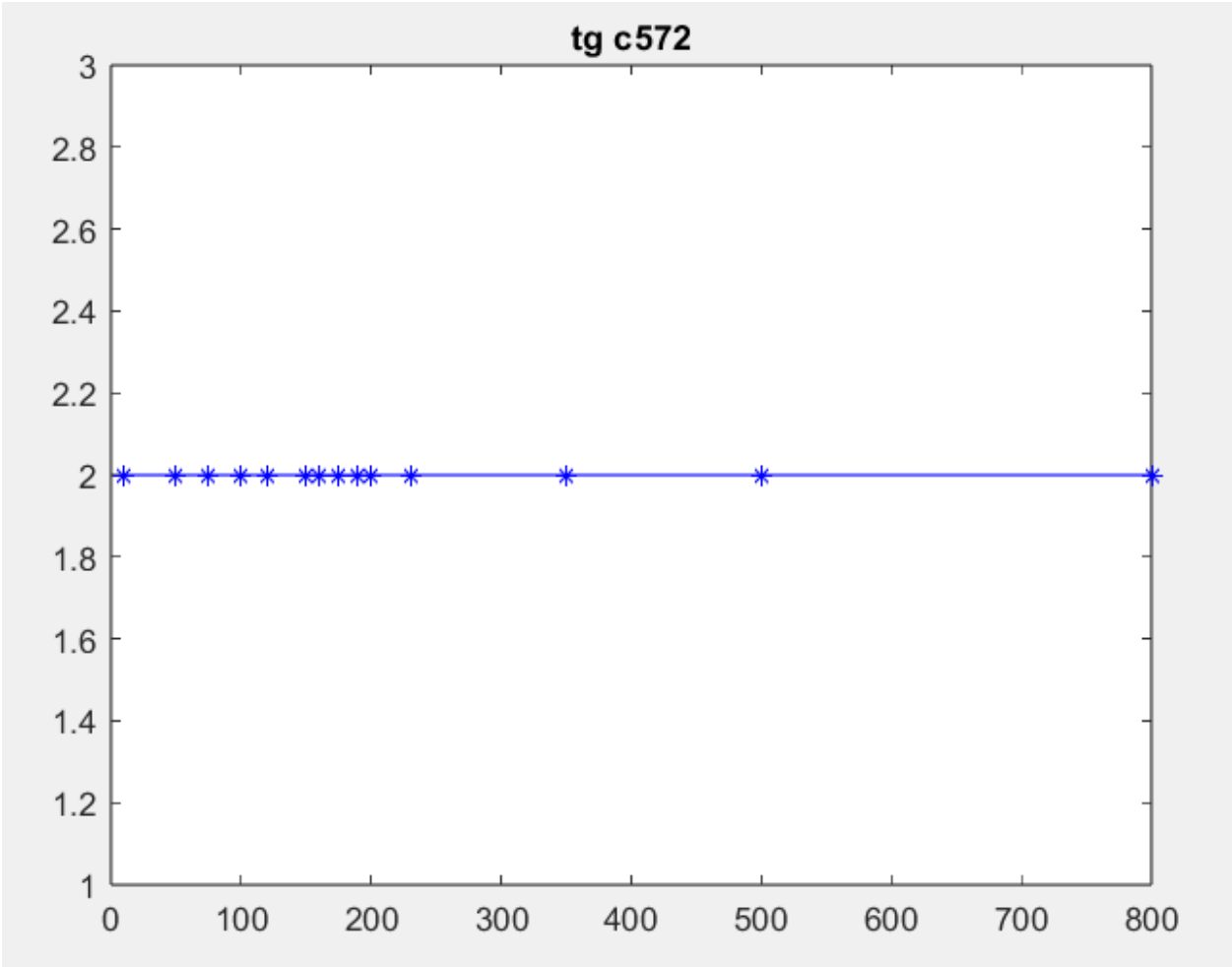
Sensitivity Analysis per nodo

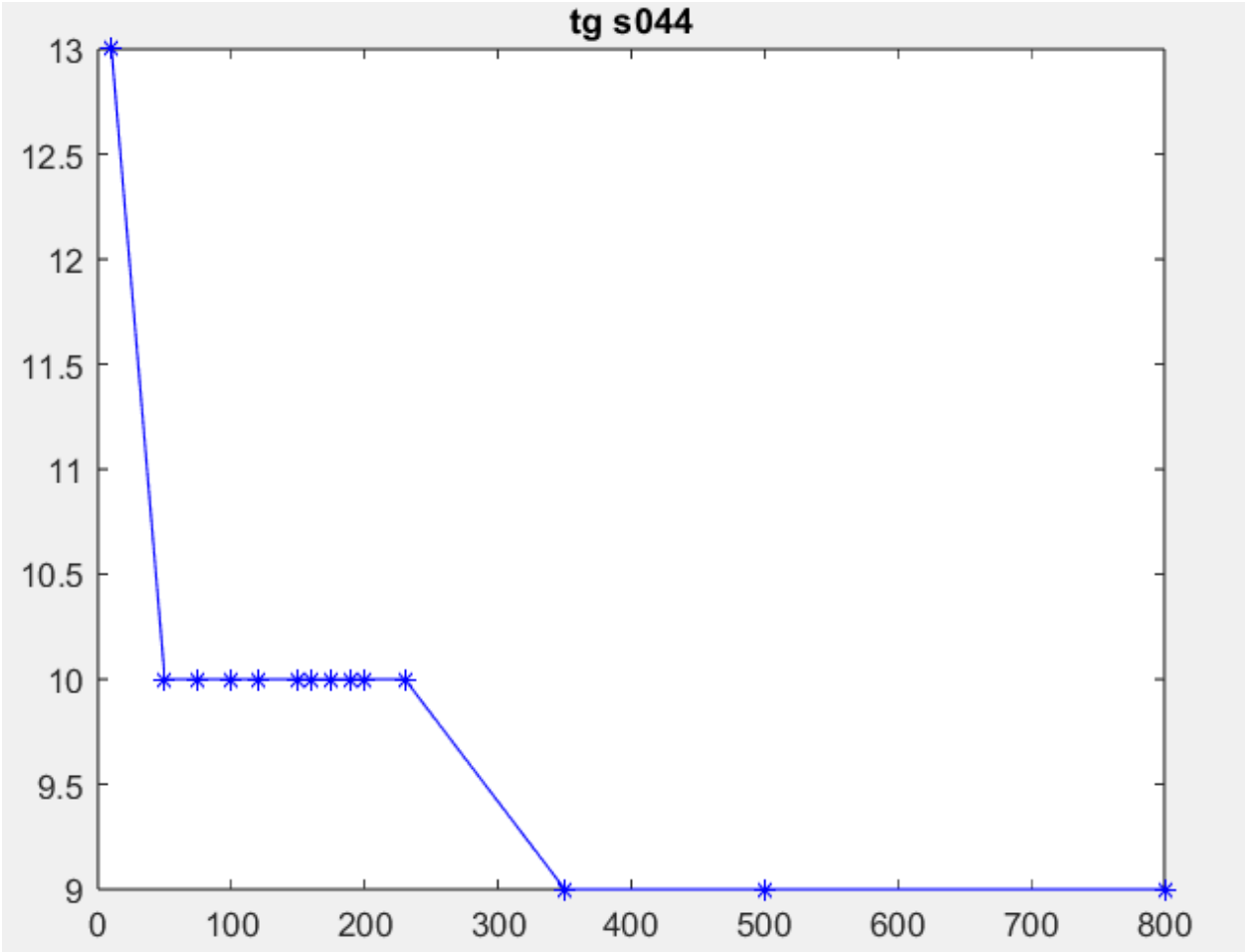
Come nel caso precedente, anche qui è stato effettuato il filtraggio e il tuple count dei **singoli nodi** del sistema. Non sono stati considerati tutti i nodi, ma solo i **10 con più errori** (come riportato nel log).

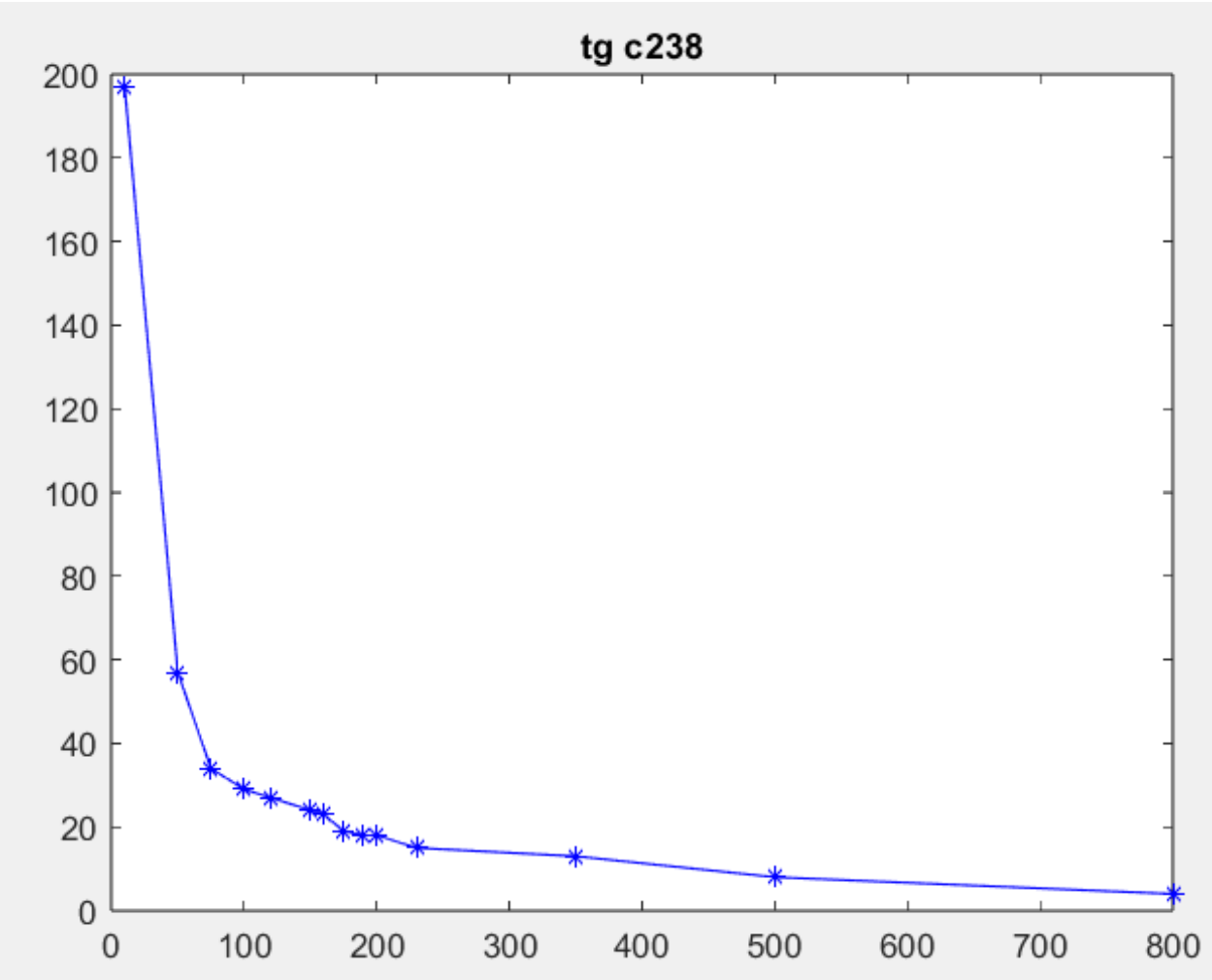

3500
3000
2500
2000
1500
1000
500
0
0
100
200
300
400
500
600
700
800
c117
c238
c242
c401
c572
c648
c669
login
master
s044

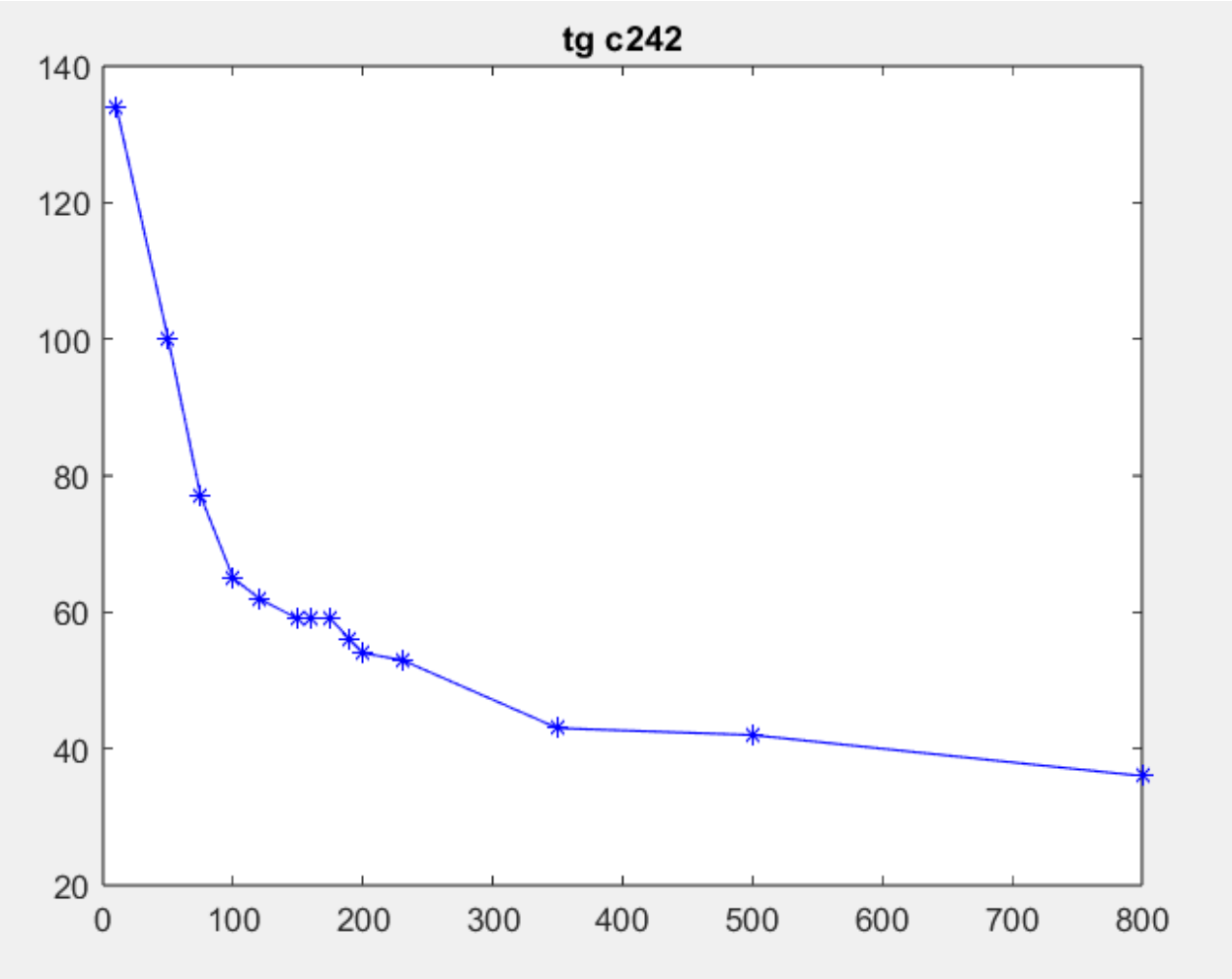

tg c401
3500
3000
2500
2000
1500
1000
500
0
0
100
200
300
400
500
600
700
800

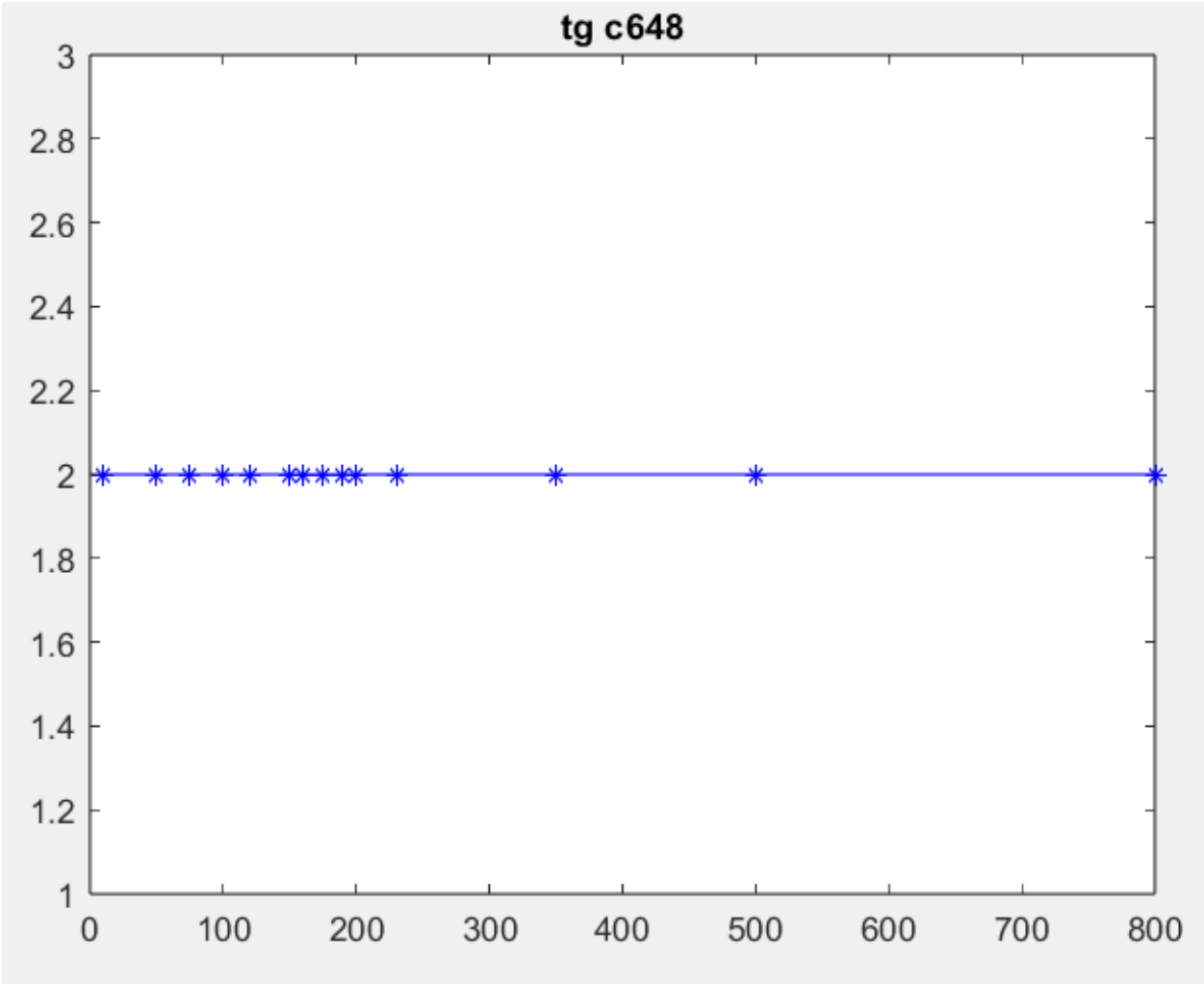

tg master
95
90
85
80
75
70
65
60
55
0
100
200
300
400
500
600
700
800

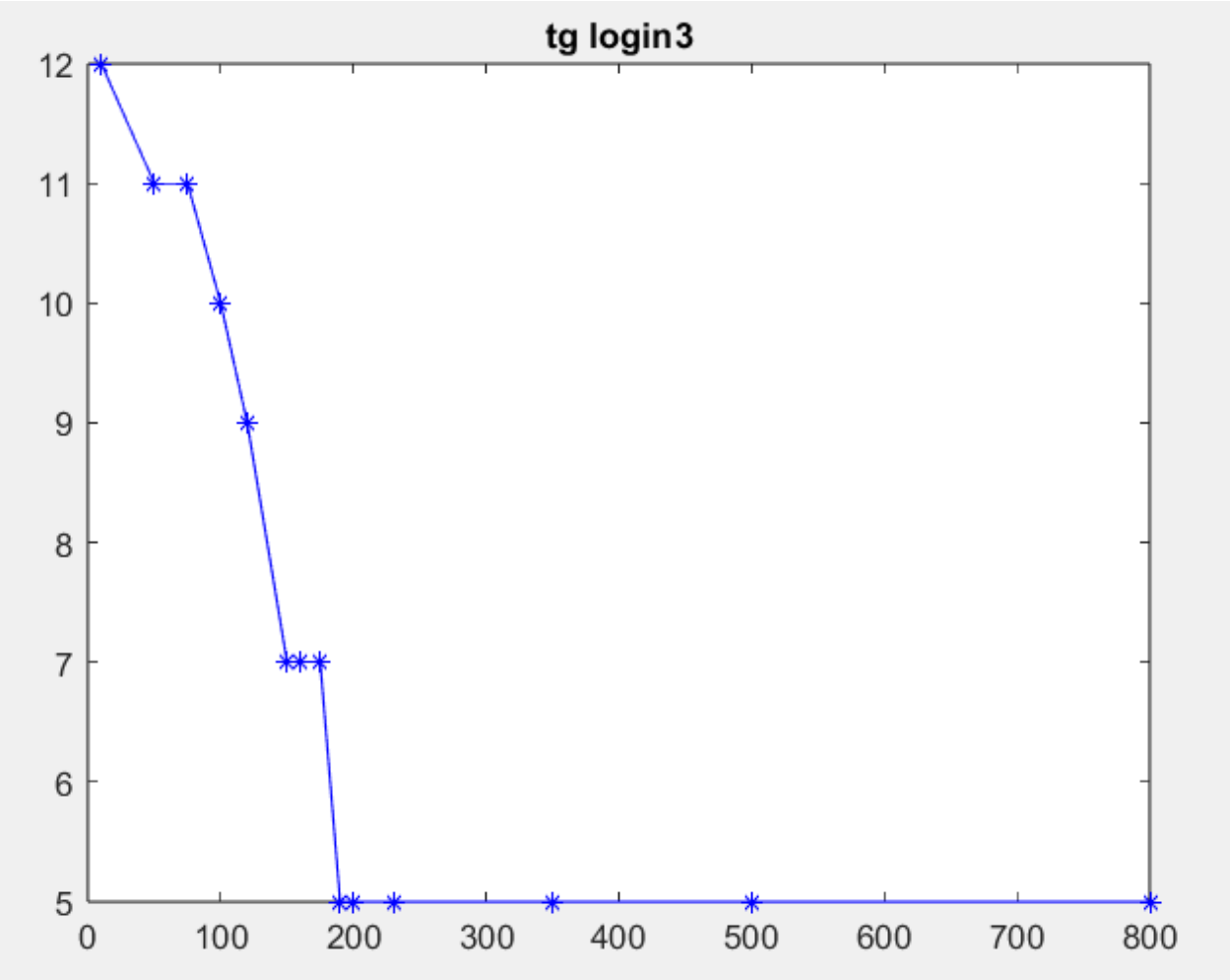

tg c572
3
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
1.2
1
0
100
200
300
400
500
600
700
800

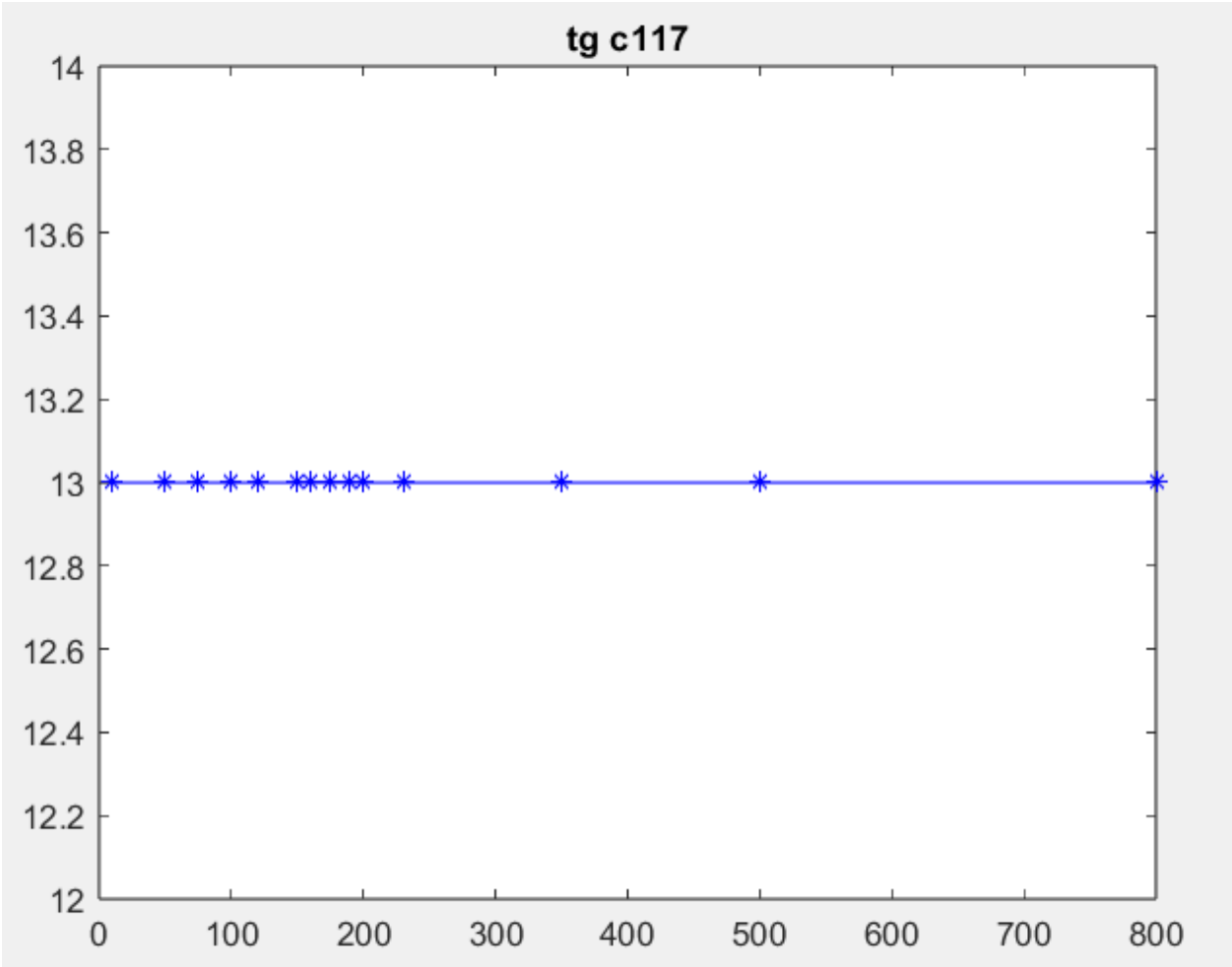

tg s044
13
12.5
12
11.5
11
10.5
10
9.5
9
0
100
200
300
400
500
600
700
800

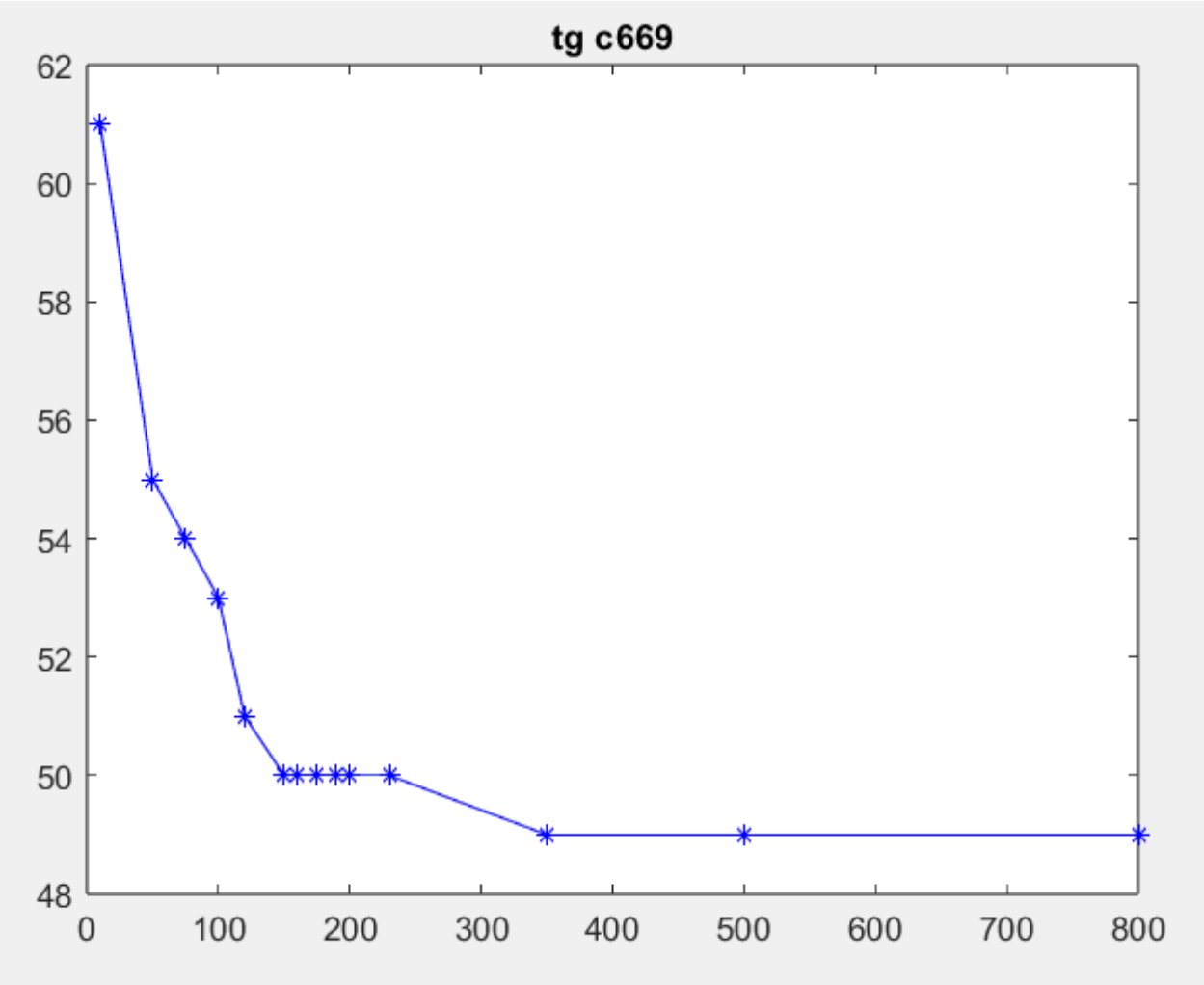

tg c238
200
180
160
140
120
100
80
60
40
20
0
0
100
200
300
400
500
600
700
800


tg c242
140
120
100
80
60
40
20
0
100
200
300
400
500
600
700
800


tg c648
3
2.8
2.6
2.4
2.2
2
1.8
1.6
1.4
1.2
1
0
100
200
300
400
500
600
700
800


tg login3
12
11
10
9
8
7
6
5
0
100
200
300
400
500
600
700
800


tg c117
14
13.8
13.6
13.4
13.2
13
12.8
12.6
12.4
12.2
12
0
100
200
300
400
500
600
700
800

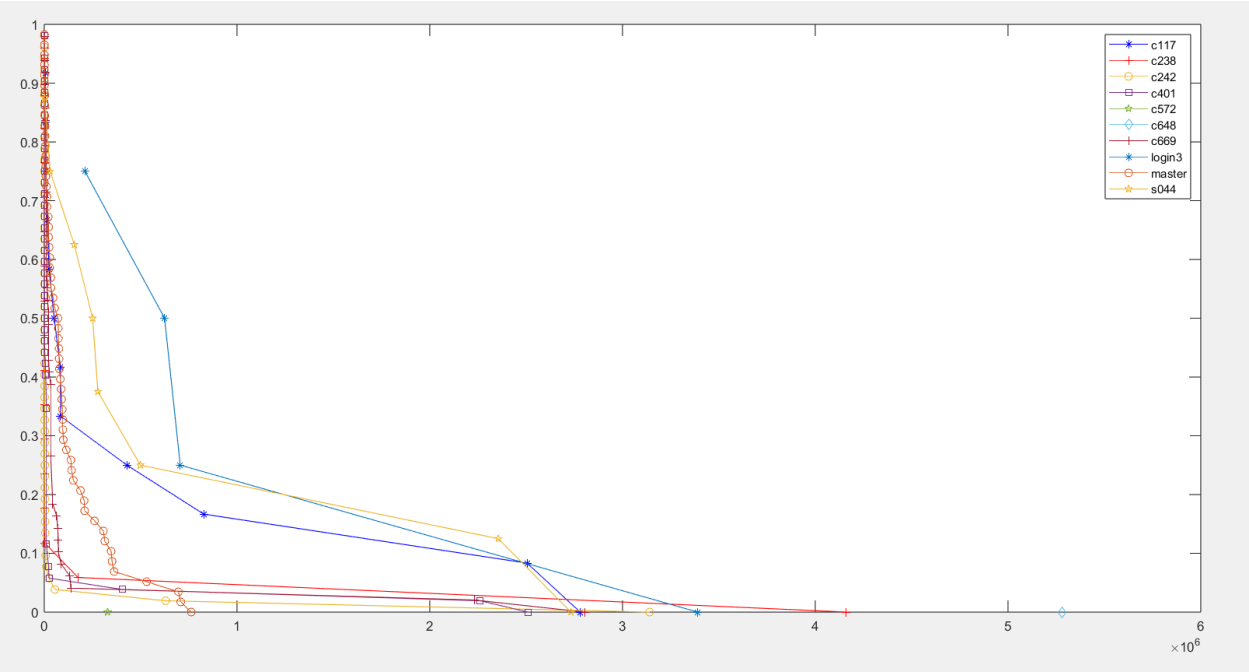


Node	CWIN
tg-c401	230
tg-master	230
tg-c572	indifferente
tg-s044	350
tg-c238	190
tg-c242	230
tg-c648	indifferente
tg-login3	200
tg-c117	indifferente
tg-c669	160

In questo caso, si nota che vi sono dei nodi per cui il tupling **non ha effetto**, per cui la scelta della finestra CWIN è praticamente indifferente (saranno dei log di errore molto distanti nel tempo, per cui vengono inseriti sempre nelle stesse tuple).

Per loro, ai fini di effettuare l'analisi di Reliability, è quindi possibile scegliere una CWIN arbitraria, che è stata selezionata come la **mediana** delle CWIN scelte per i nodi per cui aveva senso farlo. La mediana vale **230**.

Si riportano quindi nuovamente i grafici delle reliability empiriche:



Qui si nota che vi sono dei nodi particolarmente **unreliable**. In primis si notano una serie di nodi con una knee che si trova agli inizi del grafico temporale, e questi sono: **c242, c669, c238, c401, c572**. Inoltre, si nota che il nodo **tg-master** ha una reliability **praticamente nulla** già prima del milione di secondi. Questo è imputabile al fatto che la **propagazione di failure da più nodi** porti il nodo master, che li gestisce, ad essere estremamente unreliable.

Una considerazione ulteriore che si può fare è che i nodi più proni a fallimenti sono proprio quelli di **calcolo**, che si ipotizzano essere quelli con un carico di elaborazioni molto più intensivo rispetto agli altri (storage e login). Si ritrova in effetti un nodo di login (**login3**) all'interno dei 10 con più failure come anche un nodo di storage (**s044**).

È possibile quindi effettuare un report per i nodi di calcolo unreliable riportati poco prima con l'aggiunta di quello di login e di quello di storage per avere più comprensione del motivo per cui falliscono.

```
== tg-master ==

== Total error entries ==
4098

== Breakup by CATEGORY ==
NET 3639
I-O 452
OTH 3
DEV 3
MEM 1
```

```
== tg-c242 ==

== Total error entries ==
1067

== Breakup by CATEGORY ==
DEV 918
MEM 149
```

```
== tg-c238 ==

== Total error entries ==
1273

== Breakup by CATEGORY ==
DEV 1071
MEM 197
NET 3
I-O 2
```

```
== tg-c669 ==

== Total error entries ==
267

== Breakup by CATEGORY ==
DEV 257
MEM 10
```

```
== tg-c401 ==

== Total error entries ==
62340

== Breakup by CATEGORY ==
DEV 50782
MEM 11558
```

Come riportato anche nell'analisi precedente, la maggior causa di errori è proprio provocata da errori di device PCI e da errori di memoria. Inoltre, il nodo master riporta un **gran numero di errori sulla rete**. Dando una rapida occhiata al log, è possibile notare che questi errori sono principalmente del tipo:

```
1168099806 tg-master NET setting node state to down
1168099806 tg-master NET setting node state to down
1168099806 tg-master NET setting node state to down
1168099806 tg-master NET setting node state to down
1168099806 tg-master NET setting node state to down
1168099806 tg-master NET setting node state to down
1168099806 tg-master NET setting node state to down
1168099807 tg-master NET setting node state to down
```

Per cui, come conferma dell'ipotesi precedente, il nodo master fallisce perché i nodi che gestisce falliscono.

Si riportano adesso le statistiche anche per **login-3** e **s044**:

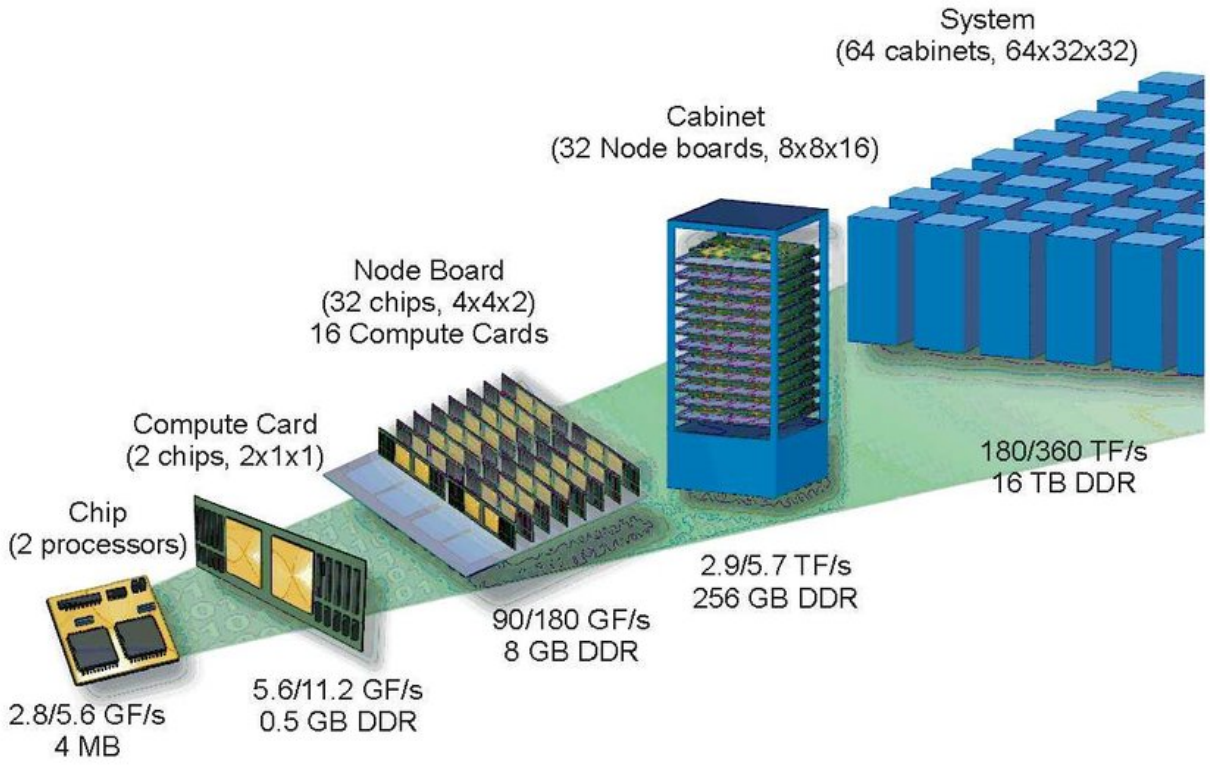
```
== tg-login3 ==

== Total error entries ==
382

== Breakup by CATEGORY ==
I-O 381
NET 1
```

```
== tg-s044 ==

== Total error entries ==
```

```
3224

== Breakup by CATEGORY ==
I-O 3208
OTH 14
NET 2
```

BlueGene

Il **cluster BlueGene** ha un'architettura ben diversa da quella di Mercury, oltre ad essere un cluster **molto più grande**.



Architettura di BlueGene

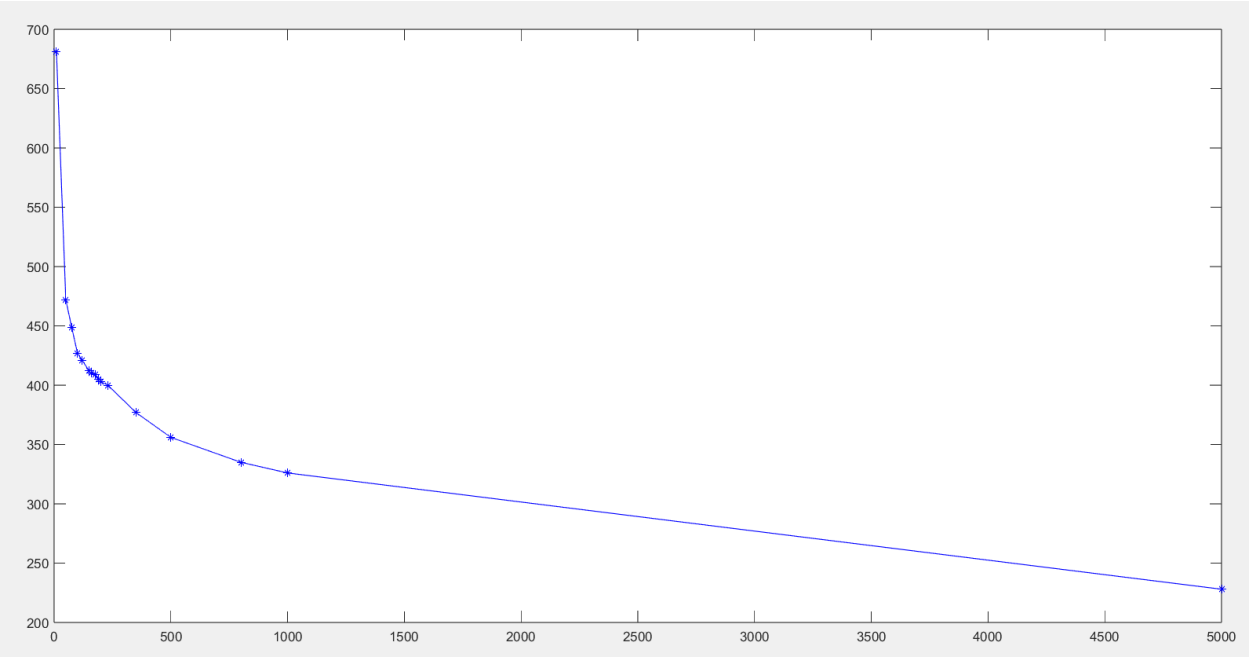
È una struttura realizzata **a livelli**, nel senso che ad alto livello abbiamo una struttura di **cabine** chiamate **rack**. In ogni rack sono presenti molte **node board** (o anche **midplane**), e ognuna di esse è costituita da un certo numero di **nodi**.

Ad esempio, **R27-M0-NA** identifica il nodo 10, nel midplane 0 del rack 27. Ognuno di questi nodi ha 16 **compute card** che vanno da J2 a J17 e i nodi N0, N4, N8 e NC hanno una I/O card aggiuntiva (J18). Ogni compute card possiede **2 compute chip** che si identificano con U01 e U11.

Per questo sistema non sono previste delle categorie di errore esplicite, in quanto ogni errore è espresso tramite un testo libero all'interno della voce del log.

Sensitivity Analysis

Come fatto precedentemente per Mercury, anche qui si è provveduto a trovare una CWIN adatta, basandosi sul metodo empirico di trovare il knee della curva:



Qui si è optato per una scelta di un valore di **230s** per la finestra di coalescenza, ottenendo un totale di **400 tuple**.

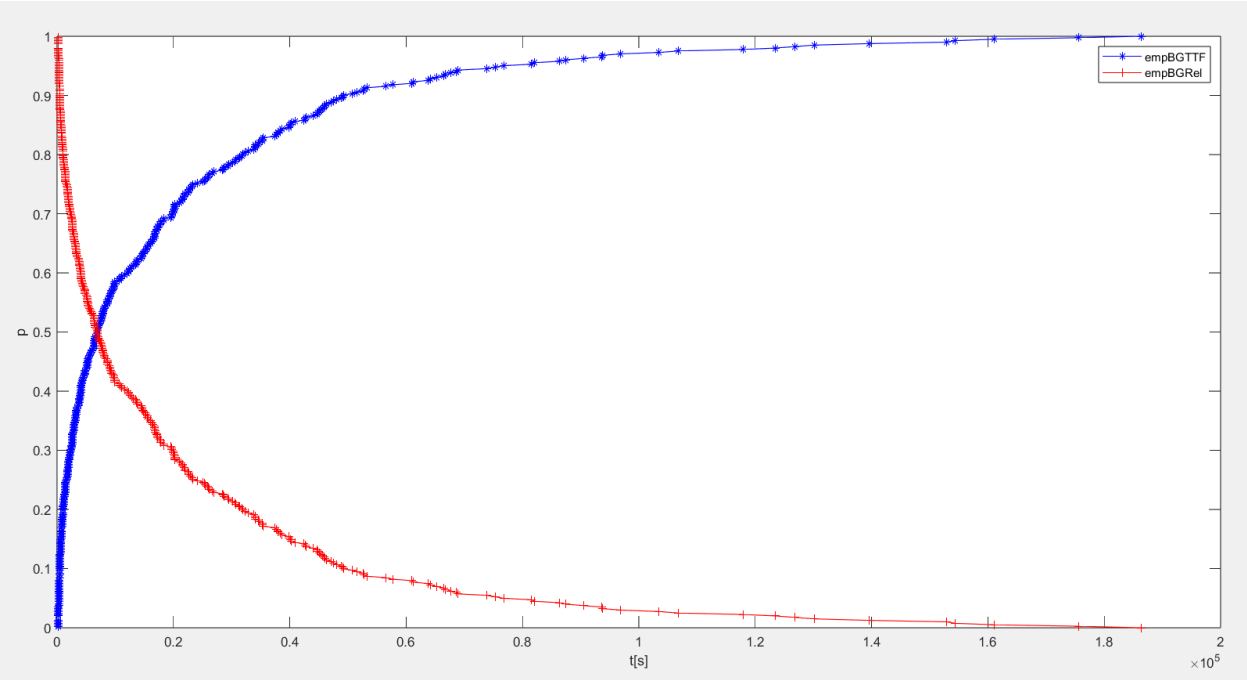
Ancora una volta, sono stati quindi calcolati i valori di **troncamenti e collisioni** per poter valutare la bontà della scelta:

- **Troncamenti:** 21 (5.25% delle tuple)
- **Collisioni:** 266 (66.5% delle tuple)

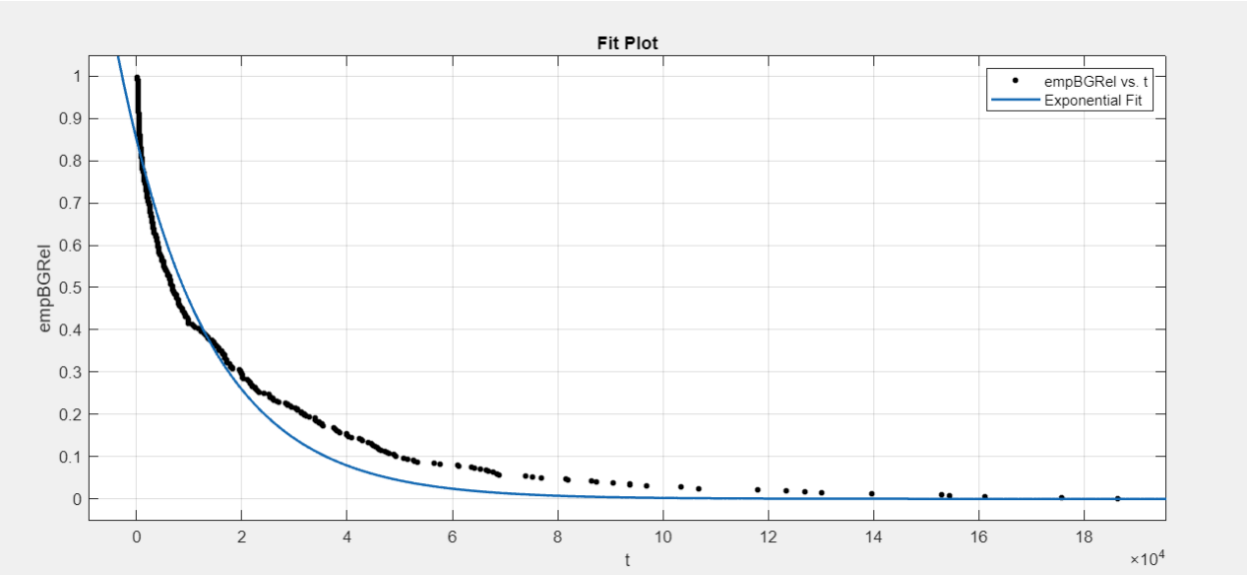
Anche qui, andando avanti con la finestra temporale, il numero di collisioni **non farebbe altro che aumentare**, e al diminuire della finestra aumenterebbero i troncamenti (con un potenziale decremento delle collisioni). Tuttavia, visto che il numero di tuple varia poco attorno al knee della tupla, si è ritenuto che questo fosse un buon risultato.

Reliability e TTF

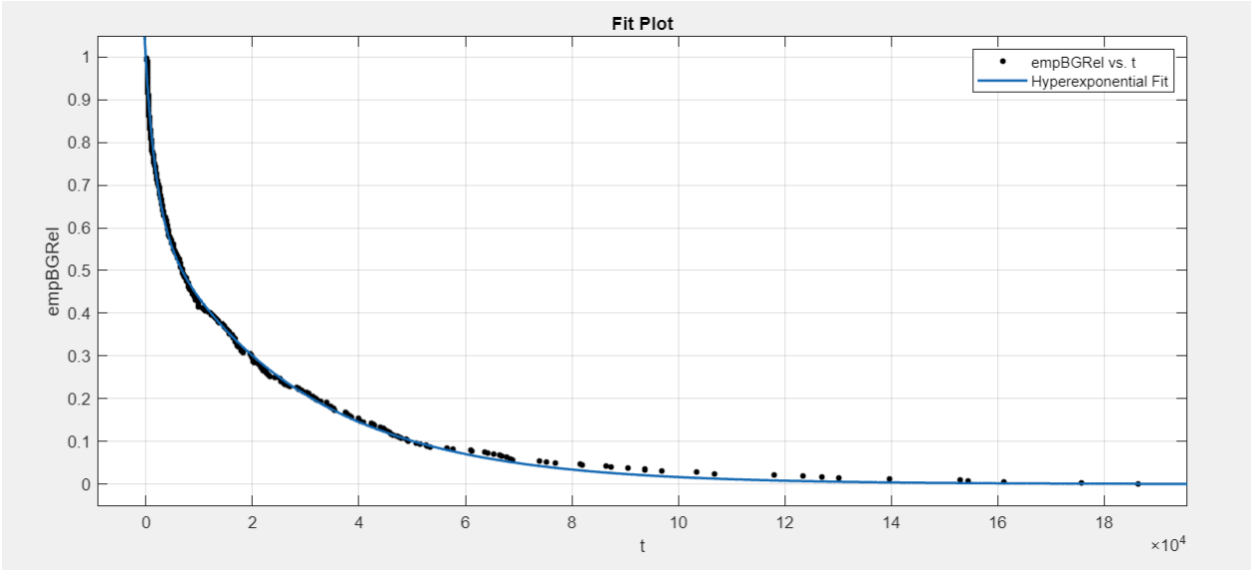
Dalla scelta della coaelscing window, si è proceduto a calcolare la reliability empirica attraverso i tempi di interarrivo delle tuple:



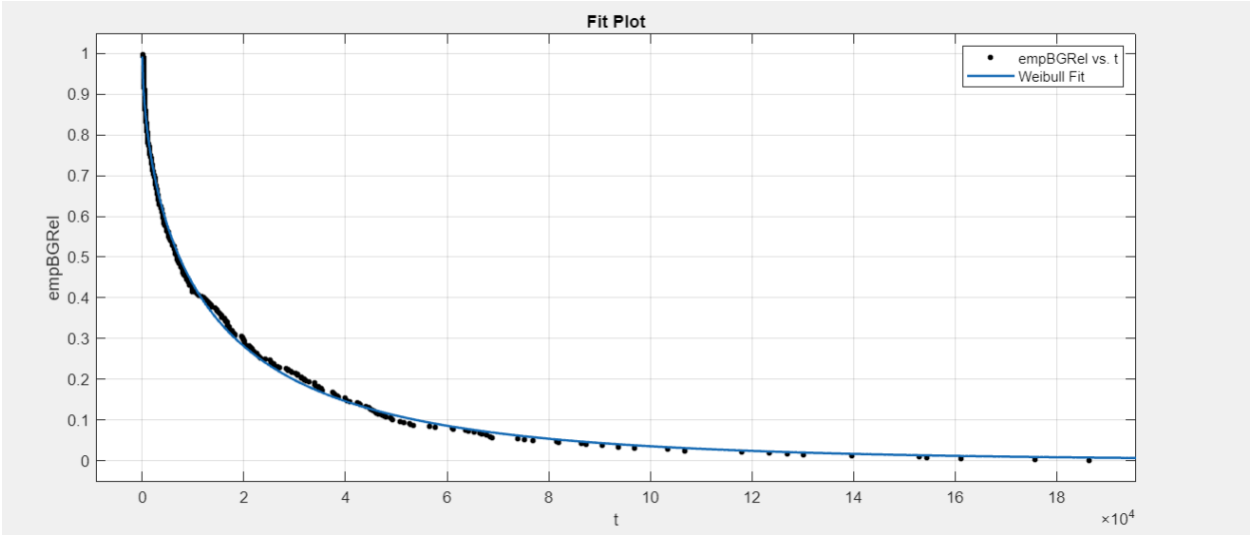
E anche qui, sono stati effettuati dei fit regressivi agli stessi modelli di funzione definiti precedentemente:



	Valore	Limite inferiore	Limite superiore
a	0.8507	0.8396	0.8618
b	-0.0001	-0.0001	-0.0001



	Valore	Limite inferiore	Limite superiore
a	0.3675	0.3595	0.3755
b	-0.0005	-0.0005	-0.0005
c	0.6222	0.6139	0.6305
d	-3.828*10^-5	-3.913*10^-5	-3.742*10^-5



	Valore	Limite inferiore	Limite superiore
a	0.6007	0.5944	0.6069
b	0.0033	0.0031	0.0035

Da questi tre modelli sono stati riportati i valori di R^2 e di **varianza persa** nella regressione:

Modello	R-quadro	R-quadro corretto	SSE
Esponenziale	0.9563	0.9562	1.4006
Iper-esponenziale	0.9978	0.9978	0.0692
Weibul	0.99179	0.9956	0.1412

E, come prima, sono stati effettuati **test di Kolmogorov-Smirnov** per decidere la funzione da utilizzare per la rappresentazione e il calcolo di MTTF:

Modello	h	p-value	k
Esponenziale	1	3.3507*10^-4	0.1472
Iper-esponenziale	0	0.8954	0.0406
Weibul	0	0.2631	0.0711

Anche in questo caso la funzione **iper-esponenziale** risulta essere la più rappresentativa, per cui possiamo esprimere la Reliability del sistema come:

$$R_{\text{BlueGene}}(t) = \begin{cases} 0.3675e^{-0.0005x} + 0.6222 & \text{se } t \geq 0 \\ 0 & \text{se } t < 0 \end{cases}$$

Ottenendo quindi un'espressione per il Mean Time To Failure:

$$MTTF = \int_0^{+\infty} R_{\text{BlueGene}}(t)dt = 21475s$$

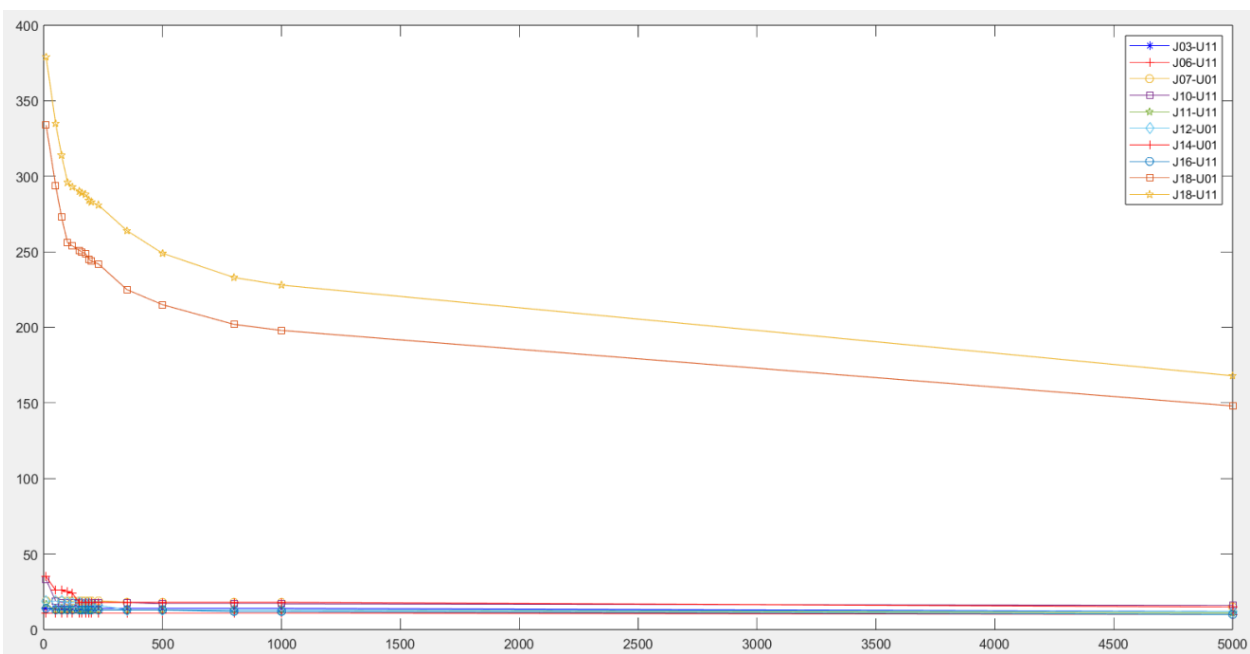
che corrispondono quindi a circa **6 ore di lavoro** affinché avvenga un failure.

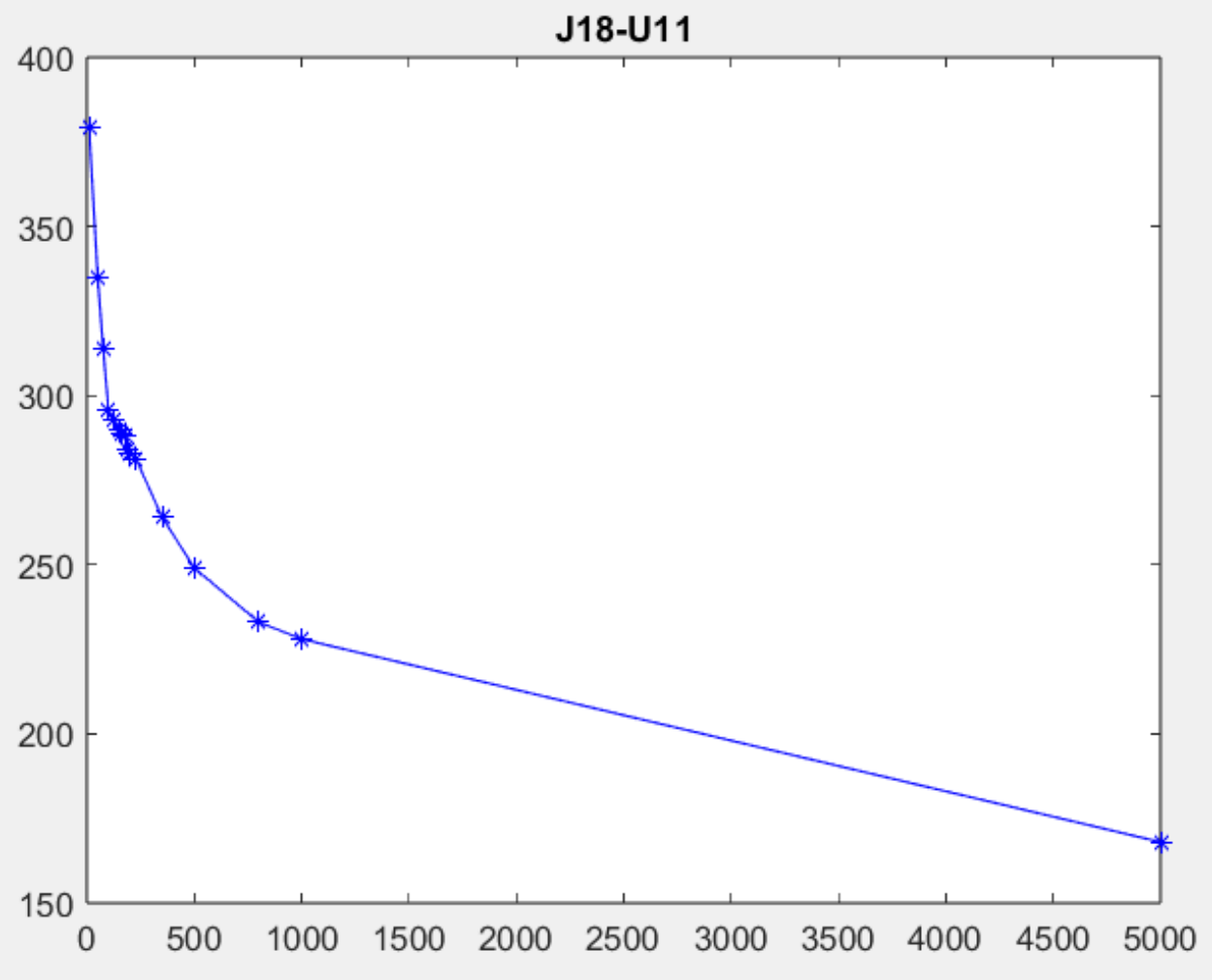
Si riportano anche qui le statistiche ottenute con lo script *logStatistics.sh* per l'analisi della sensitivity per nodo:

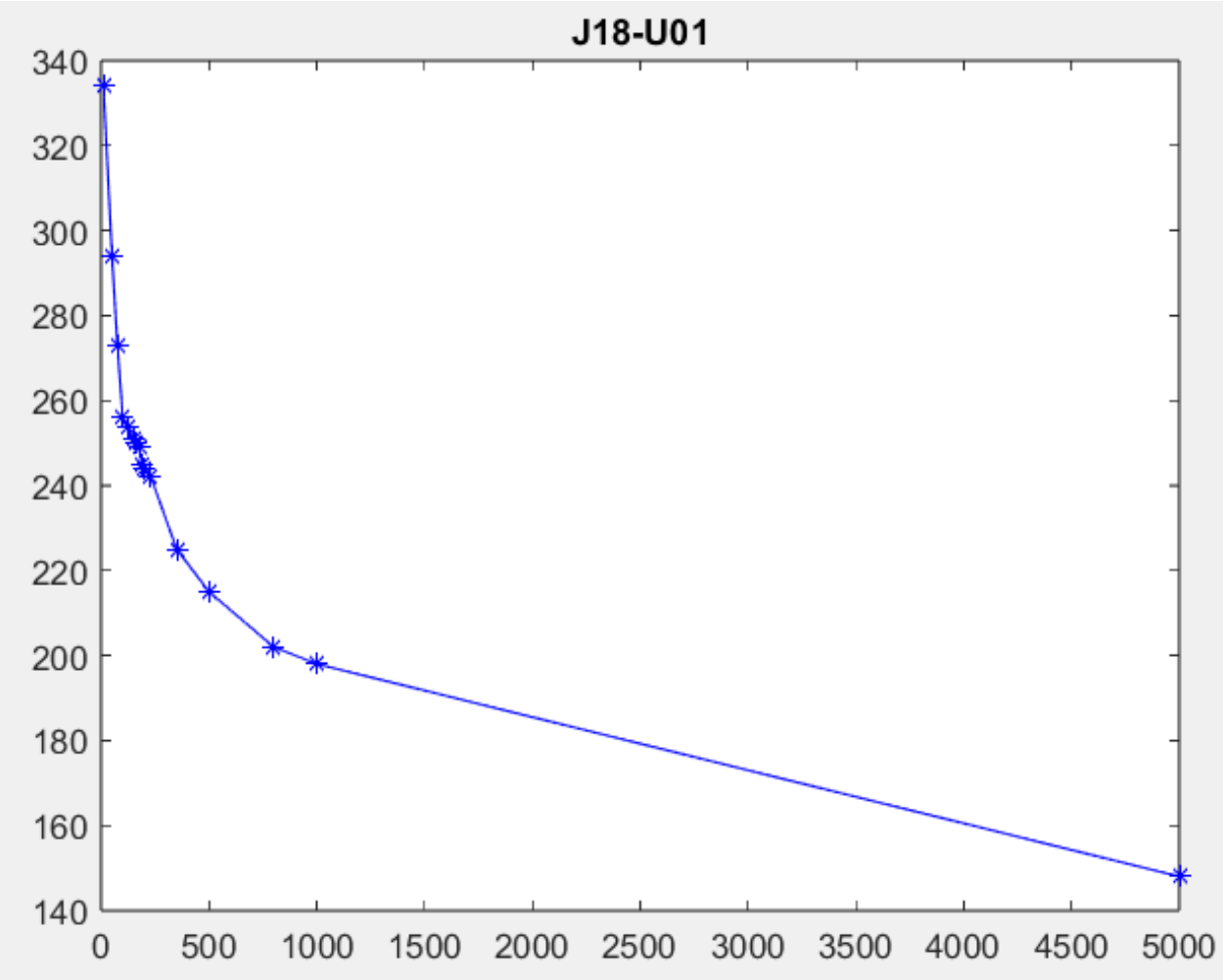
```
== BlueGene Error Log ==

== Total error entries ==
125624

== Breakup by COMPUTE CARD ==
J18-U11 50055
J18-U01 49932
J14-U01 2257
J12-U01 1877
J07-U01 1780
J10-U11 1333
J03-U11 1020
J16-U11 973
J06-U11 960
J11-U11 888
J17-U11 824
J09-U01 808
J16-U01 753
J09-U11 746
J08-U01 741
J03-U01 701
```

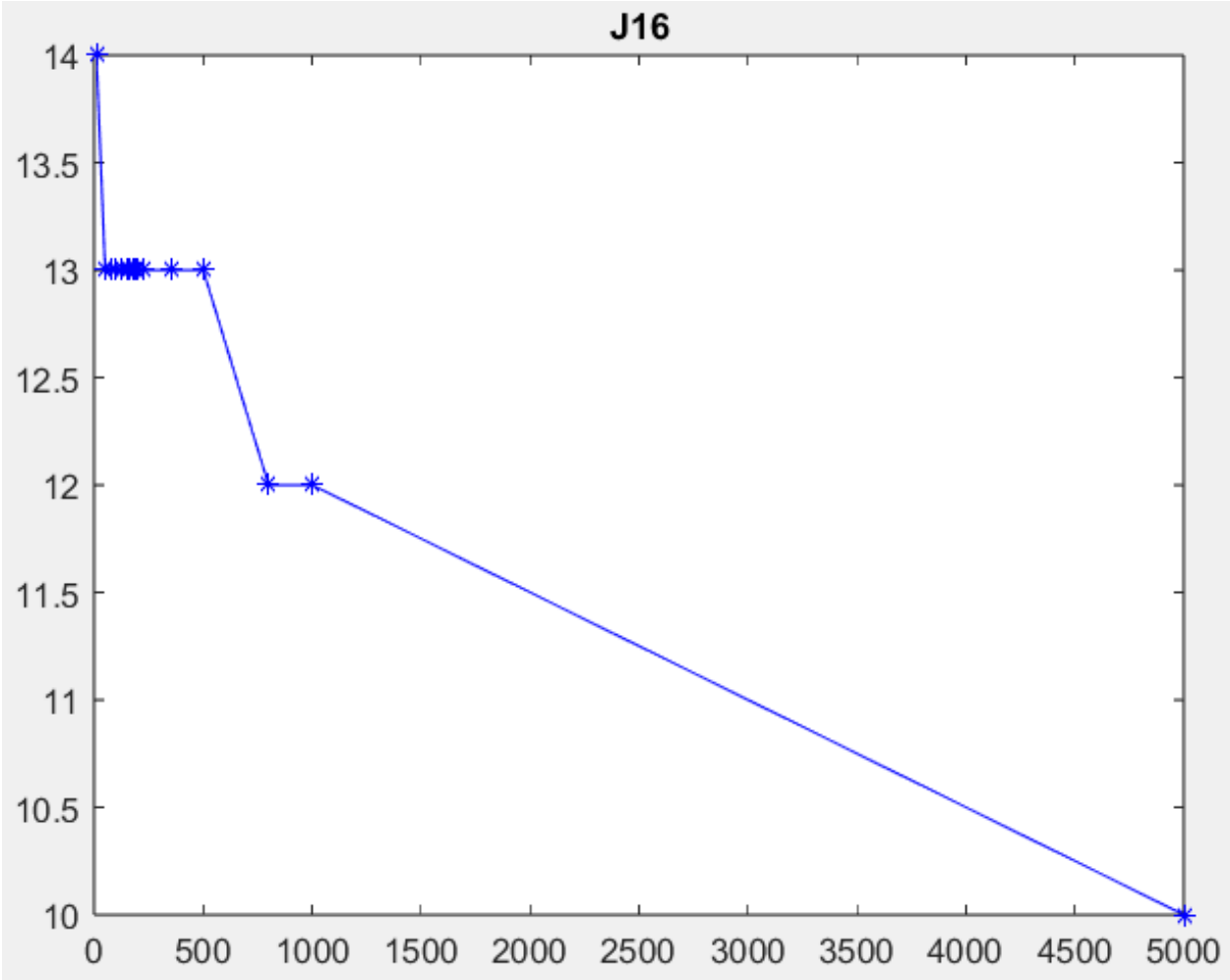
```
J11-U01 700
J05-U11 670
J04-U01 655
J13-U01 647
J15-U01 633
J17-U01 602
J12-U11 596
J04-U11 593
J13-U11 563
J08-U11 560
J10-U01 554
J02-U01 524
J05-U01 456
J15-U11 454
J02-U11 449
J14-U11 445
J07-U11 445
J06-U01 430

== Breakup by NODE* ==
R71-M0-N4 1716
R12-M0-N0 1563
R63-M0-N2 976
R03-M1-NF 960
R63-M0-N0 791
R36-M1-N0 788
R62-M0-N4 515
R63-M0-NC 460
R63-M0-N8 454
R63-M0-N4 452
  * only the 10 most occurring nodes are reported
```

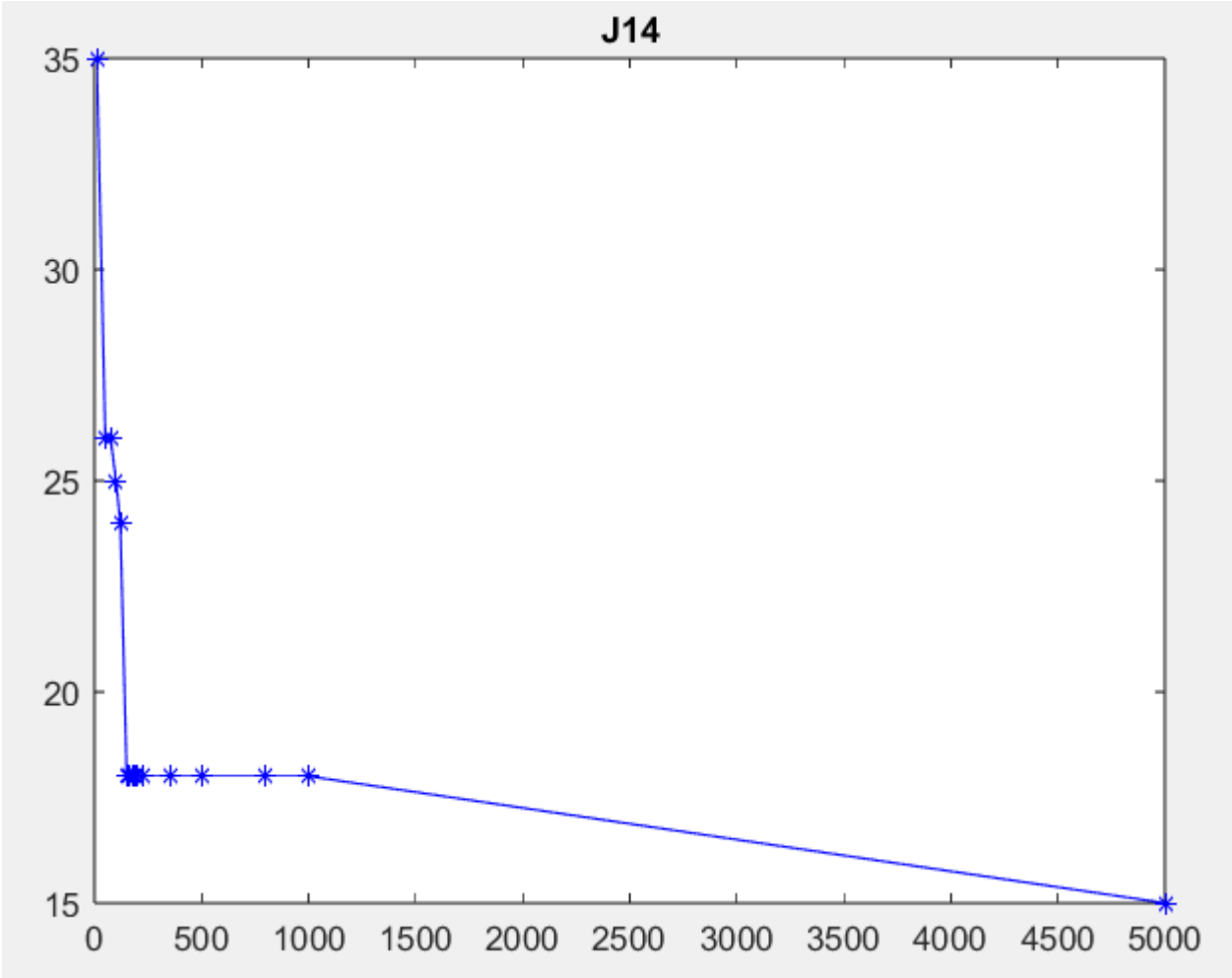
Sensitivity Analysis per compute card (opzionale)

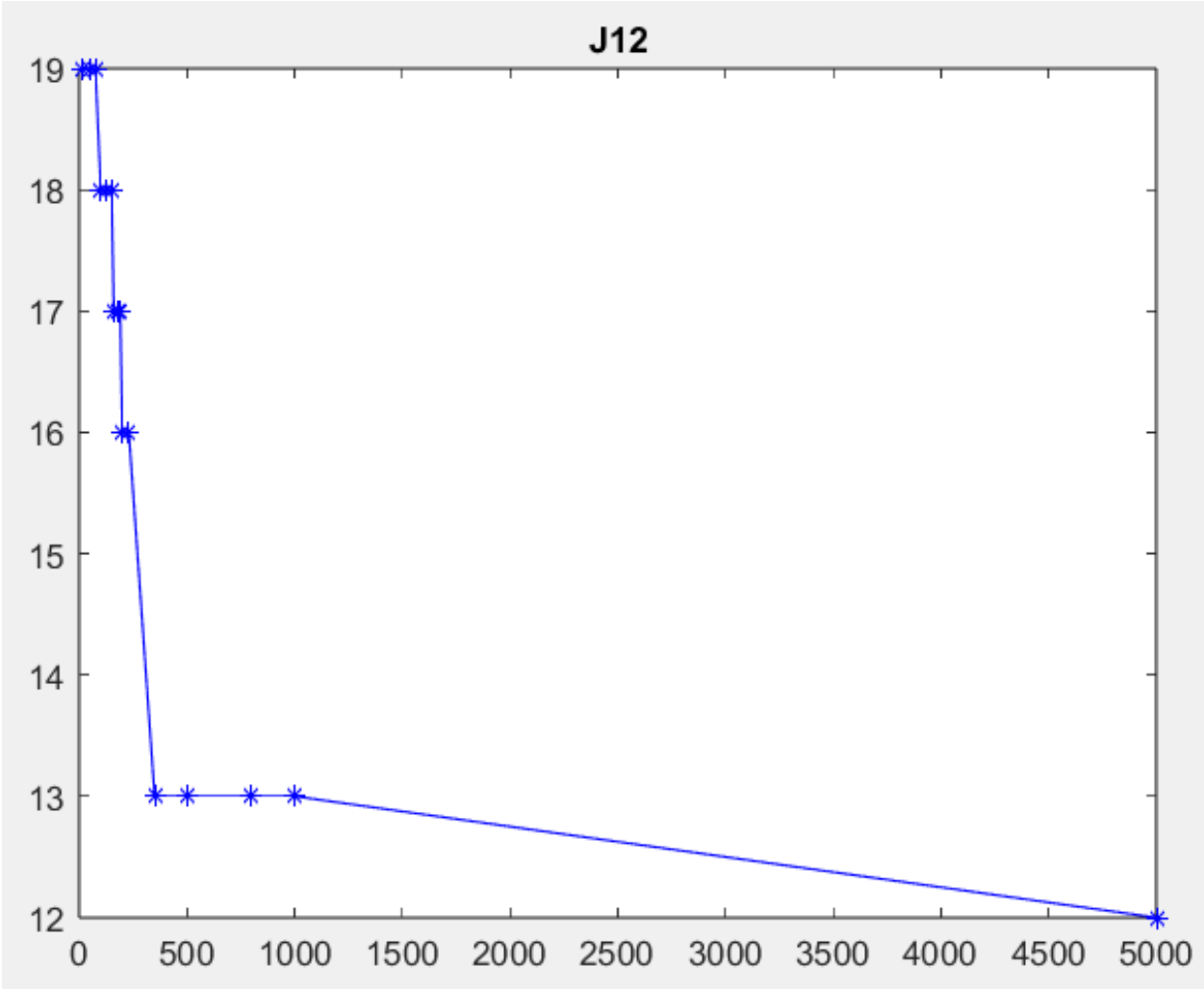


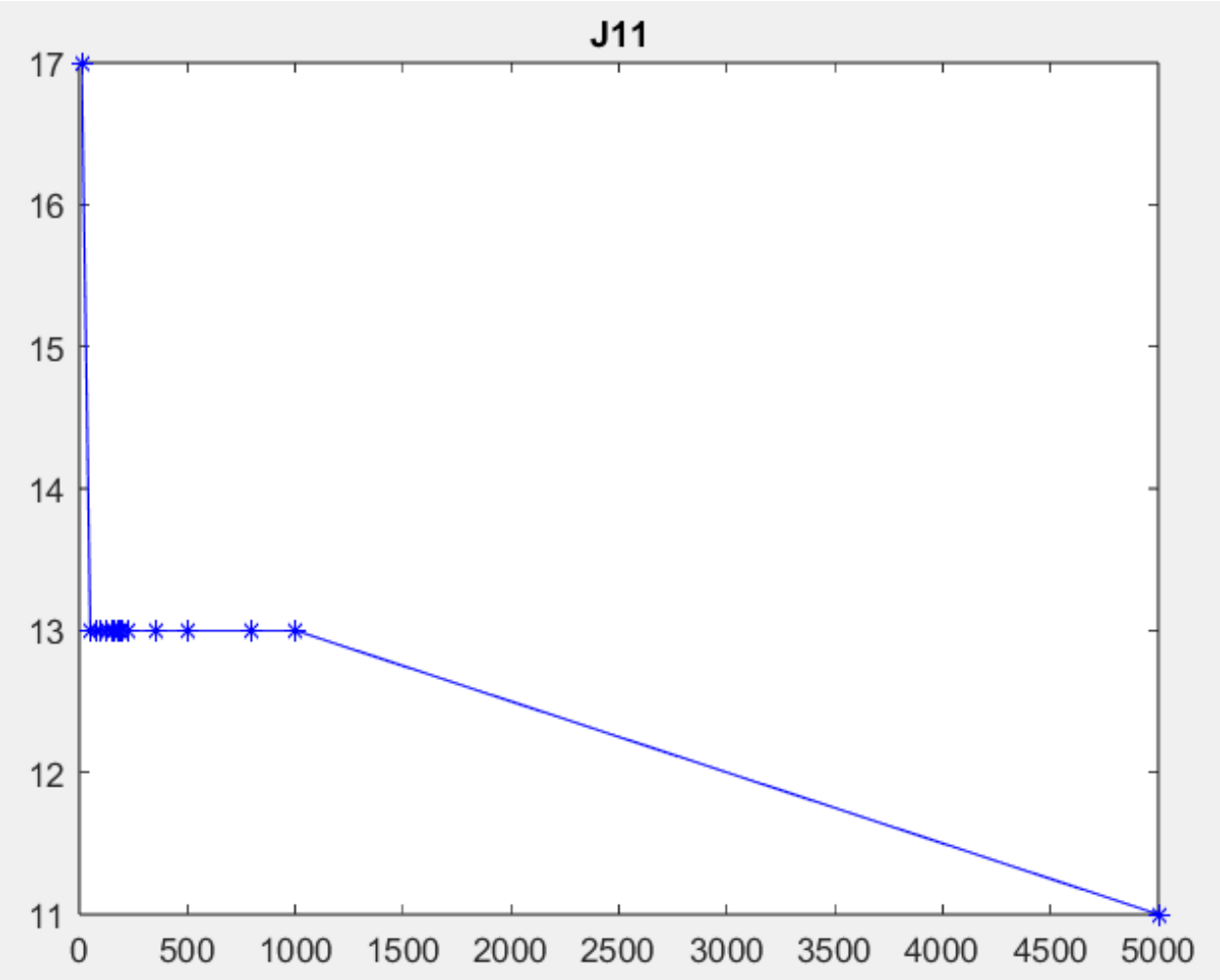

J18-U11
400
350
300
250
200
150
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000

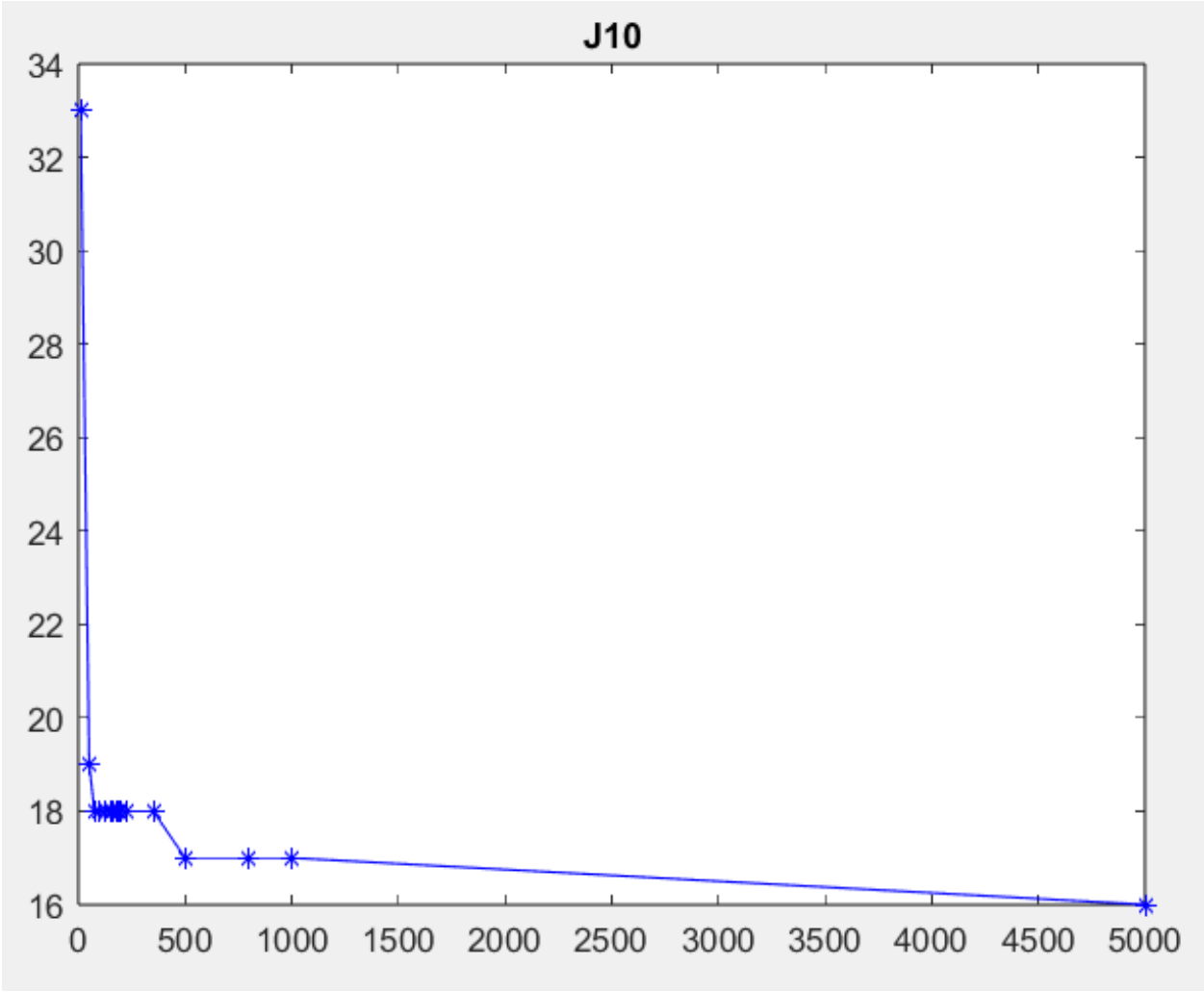

J18-U01
340
320
300
280
260
240
220
200
180
160
140
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000

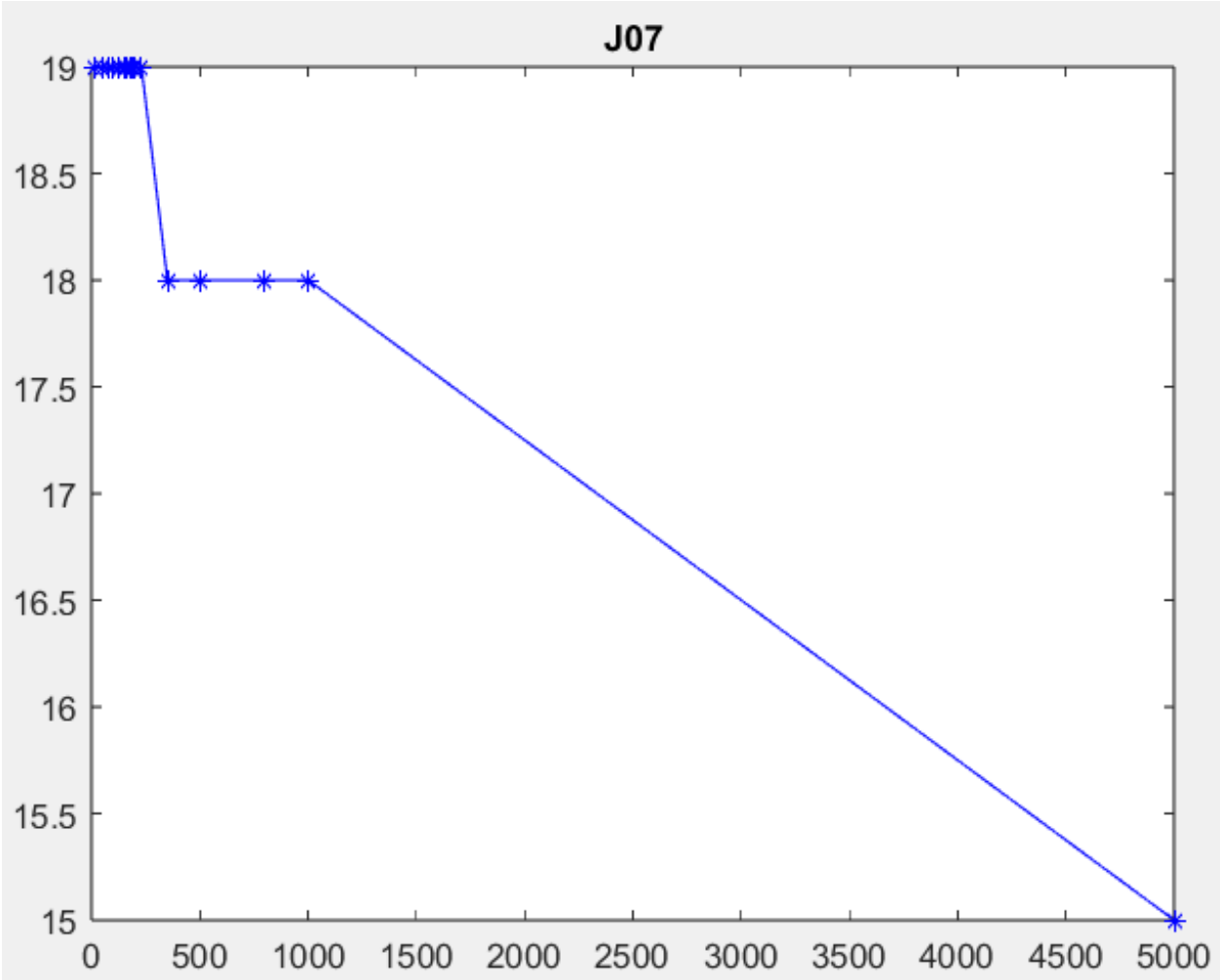

J16
14
13.5
13
12.5
12
11.5
11
10.5
10
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000


J14
35
30
25
20
15
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000


J12
19
18
17
16
15
14
13
12
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000


J11
17
16
15
14
13
12
11
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000

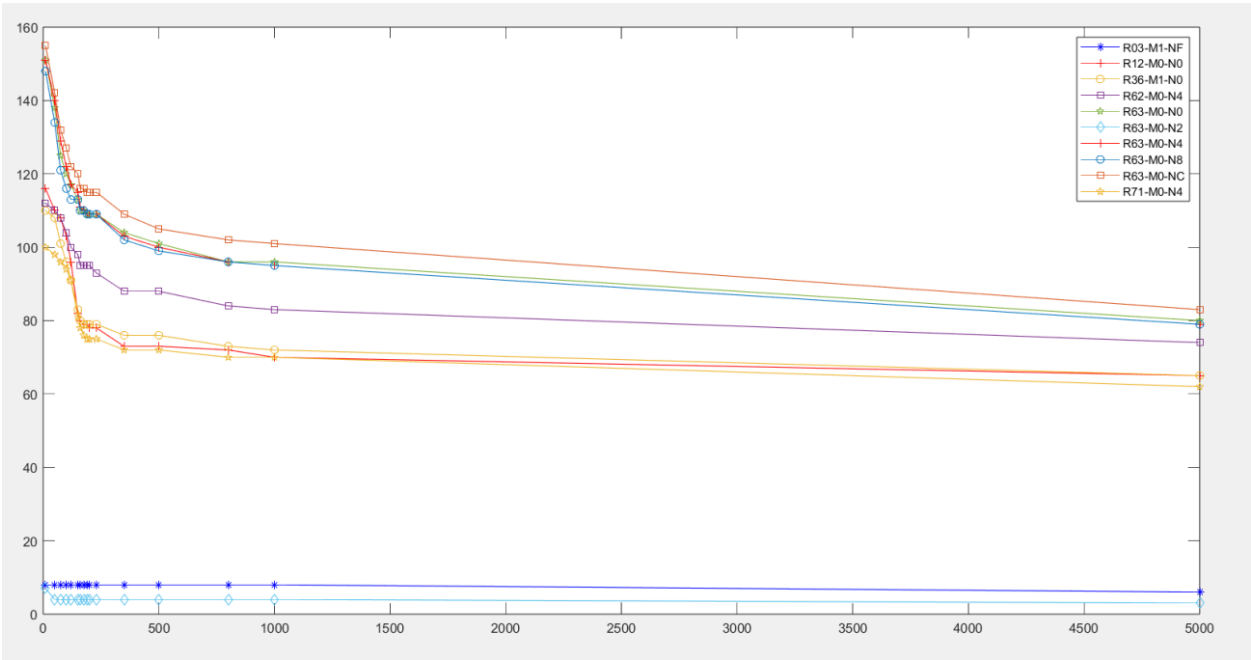

J10
34
32
30
28
26
24
22
20
18
16
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000

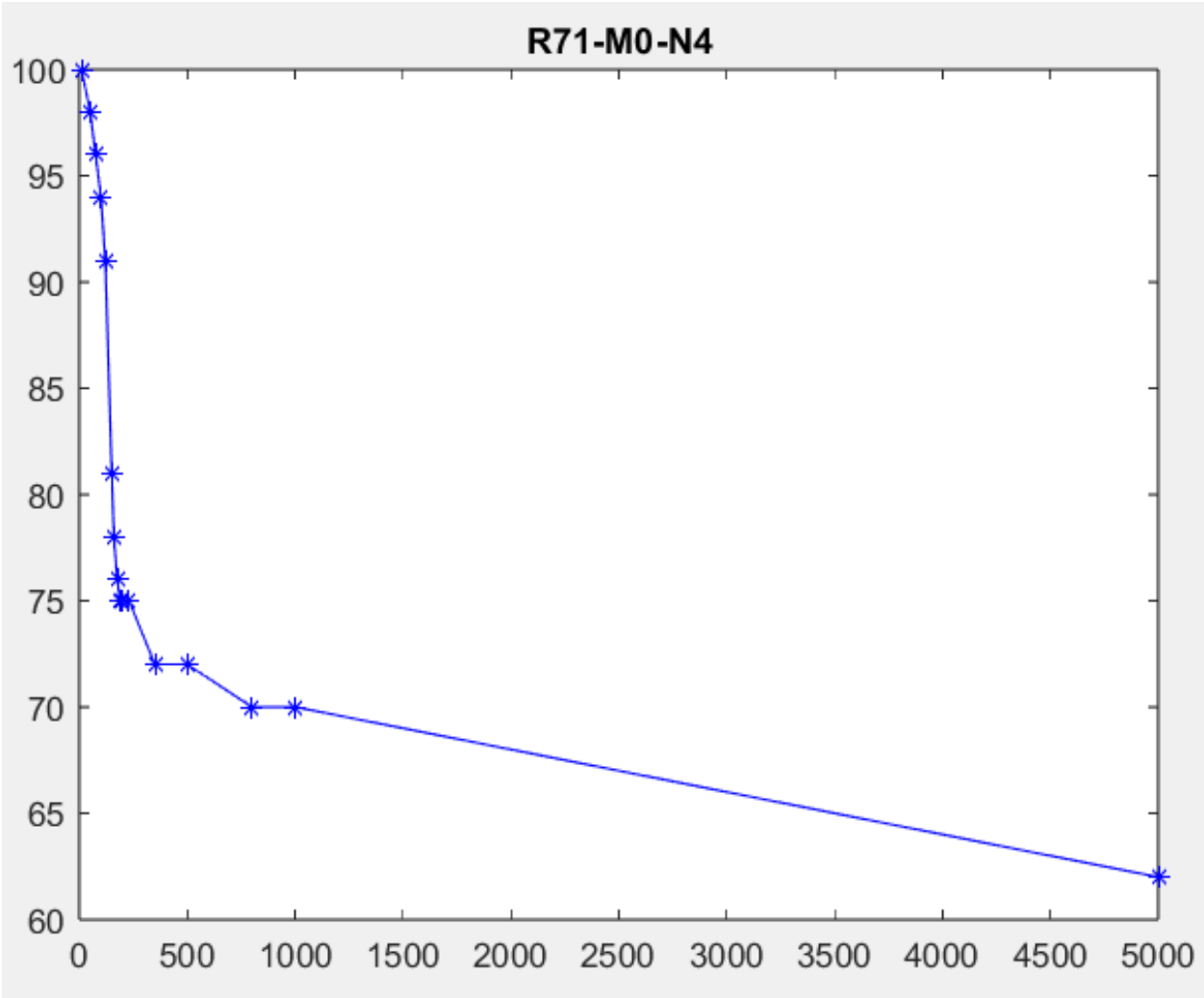


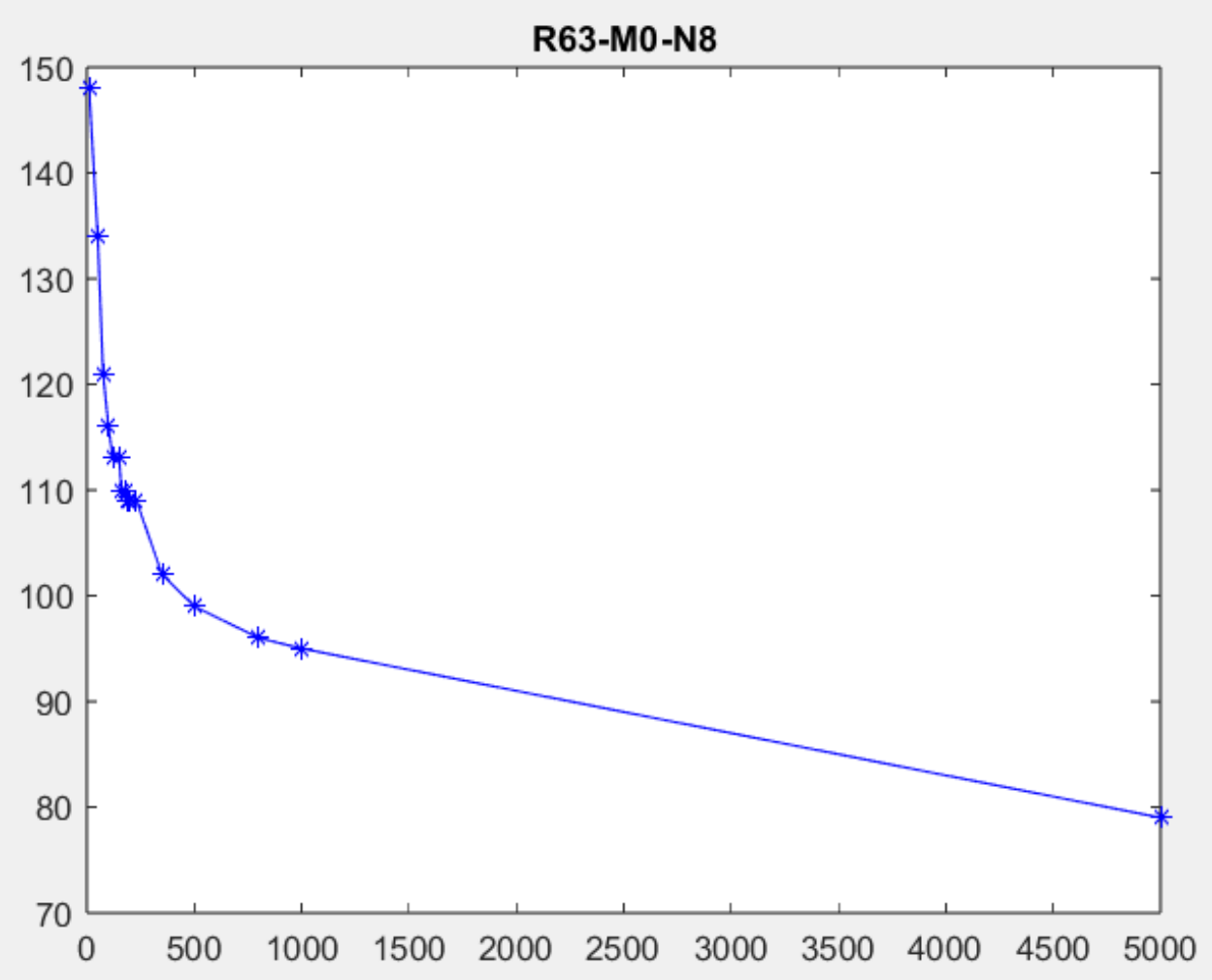
Node	CWIN
J18-U11	800
J18-U01	800
J16-U11	50
J14-U01	160
J12-U01	350
J11-U11	75
J10-U11	100
J07-U01	350

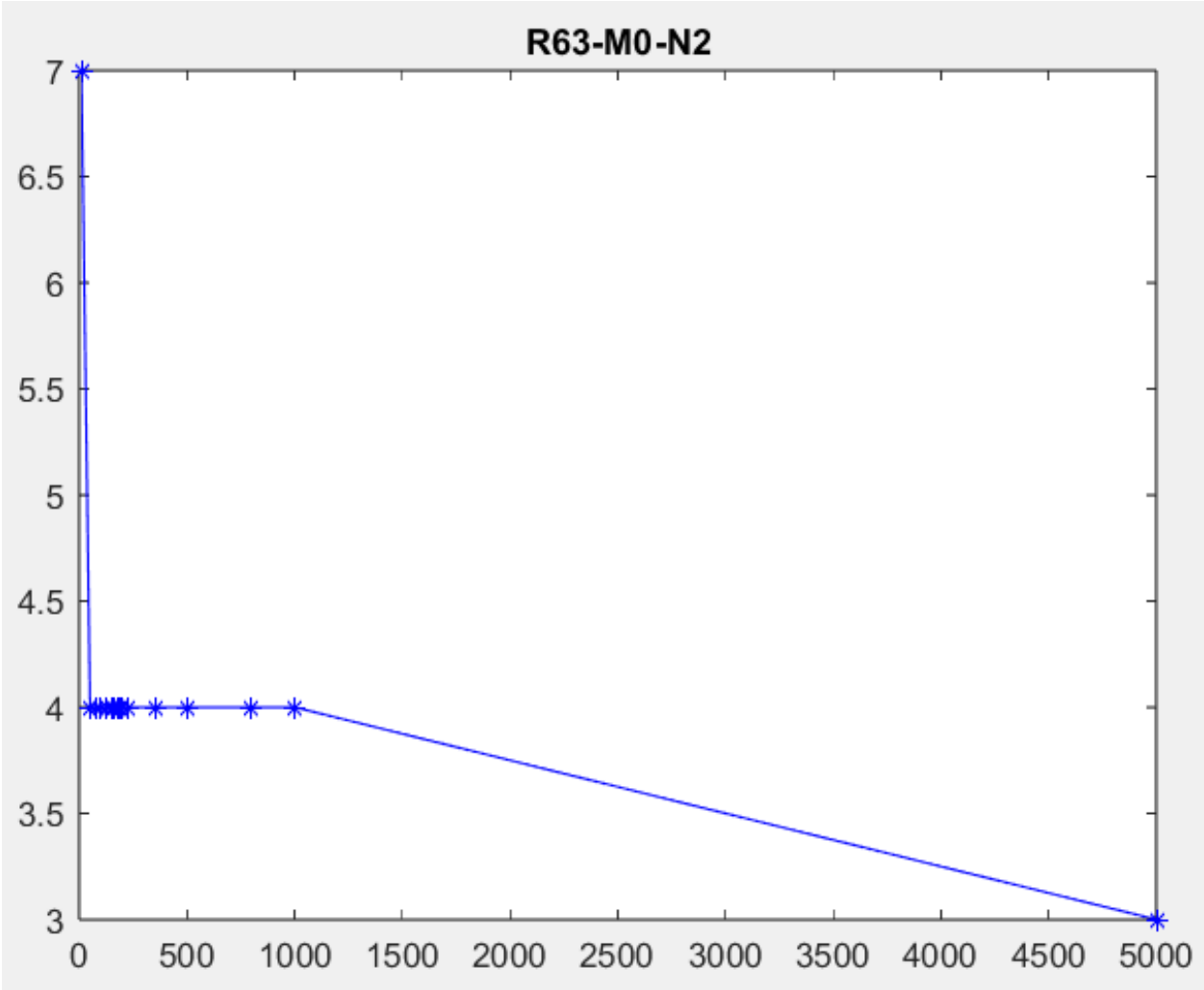
Mediana: 750

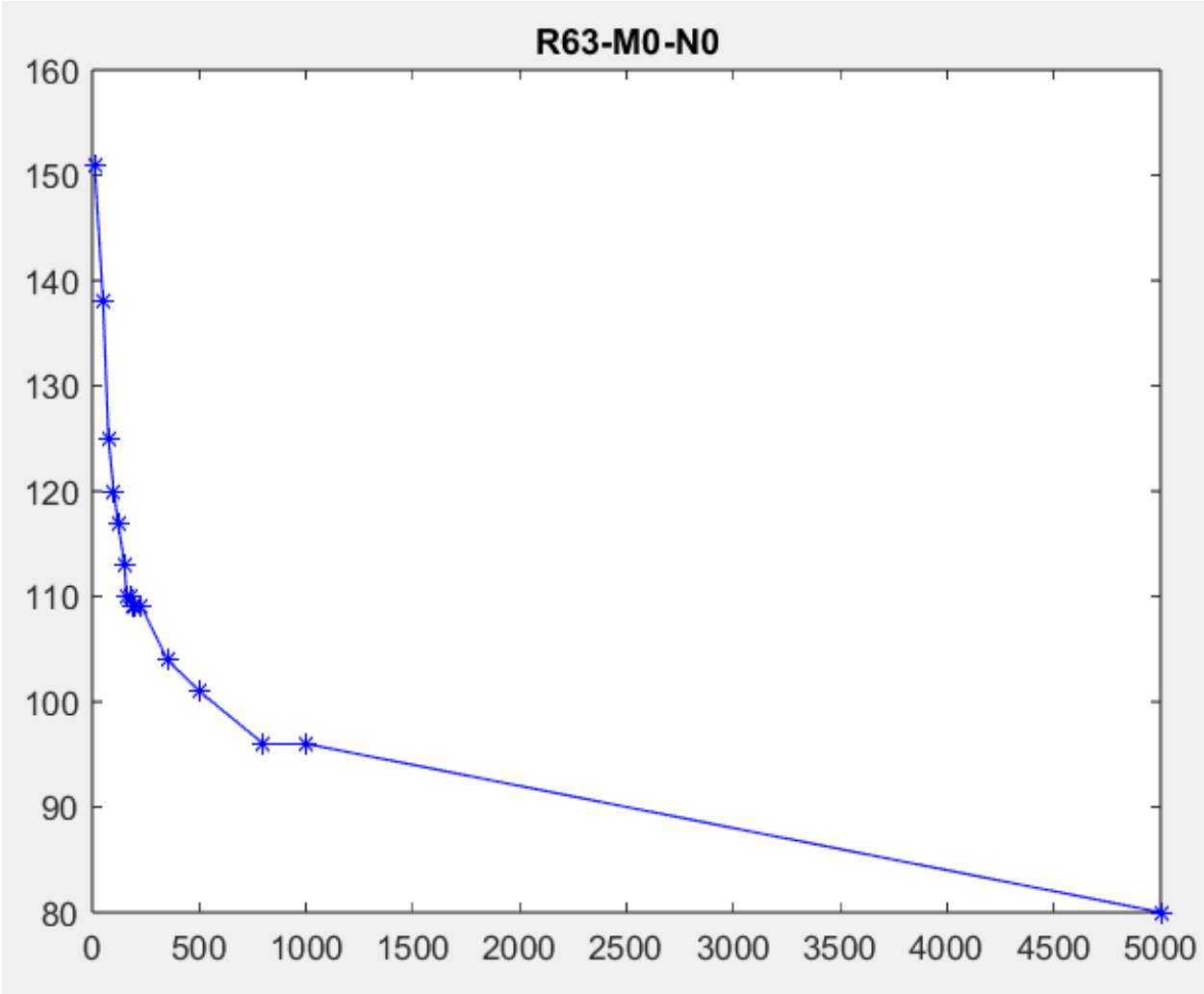
Sensitivity Analysis per nodo

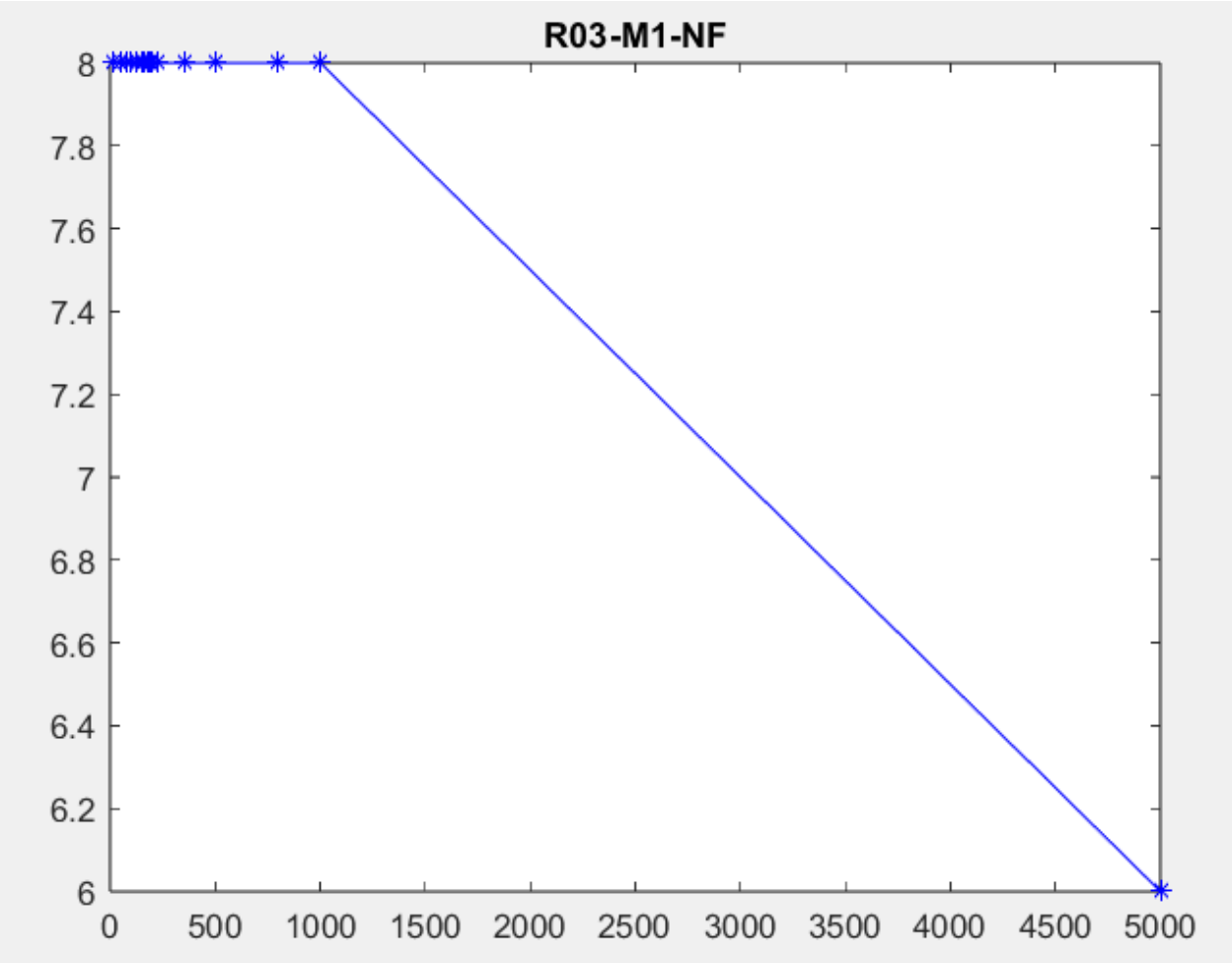


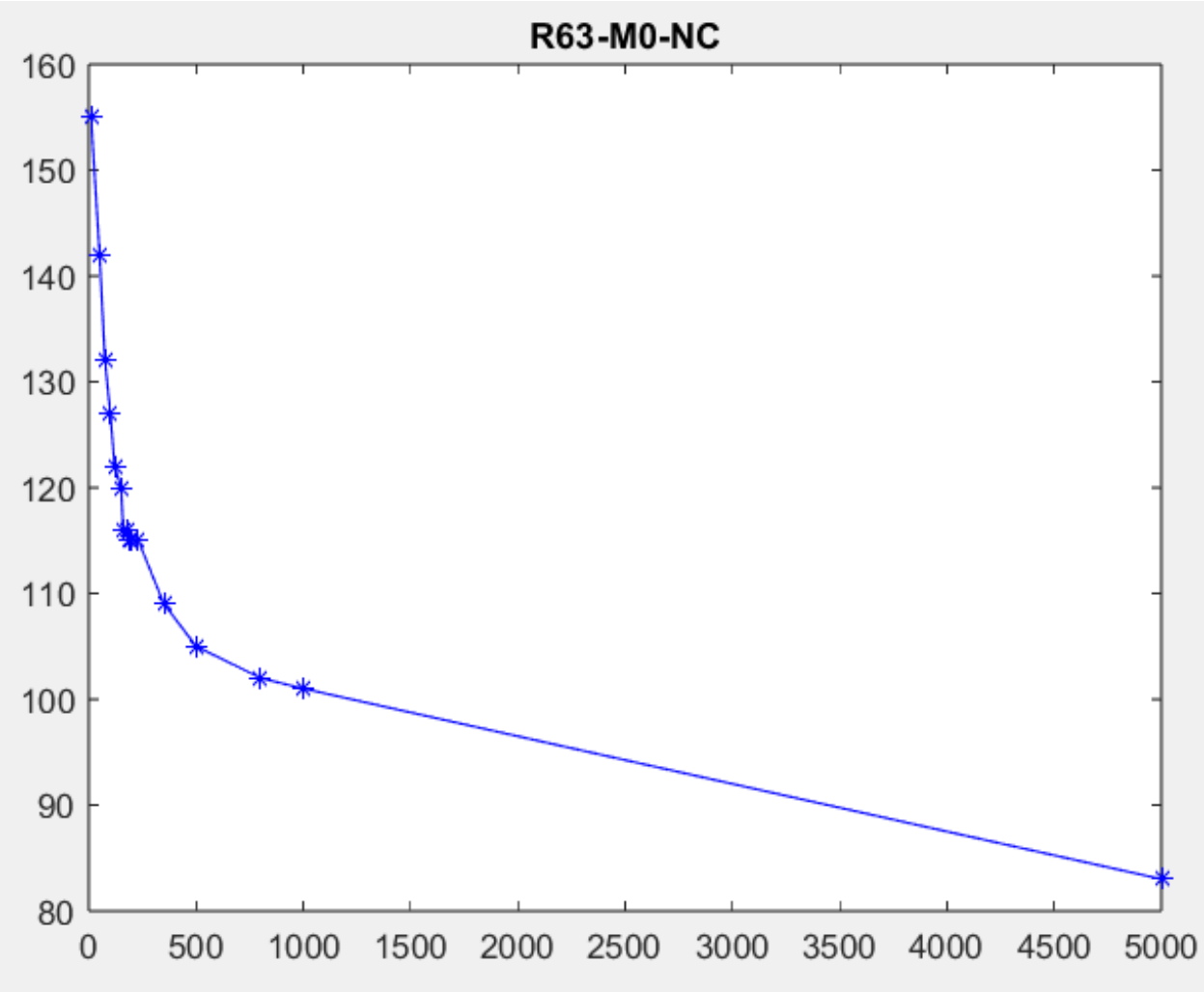

R71-M0-N4
100
95
90
85
80
75
70
65
60
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000

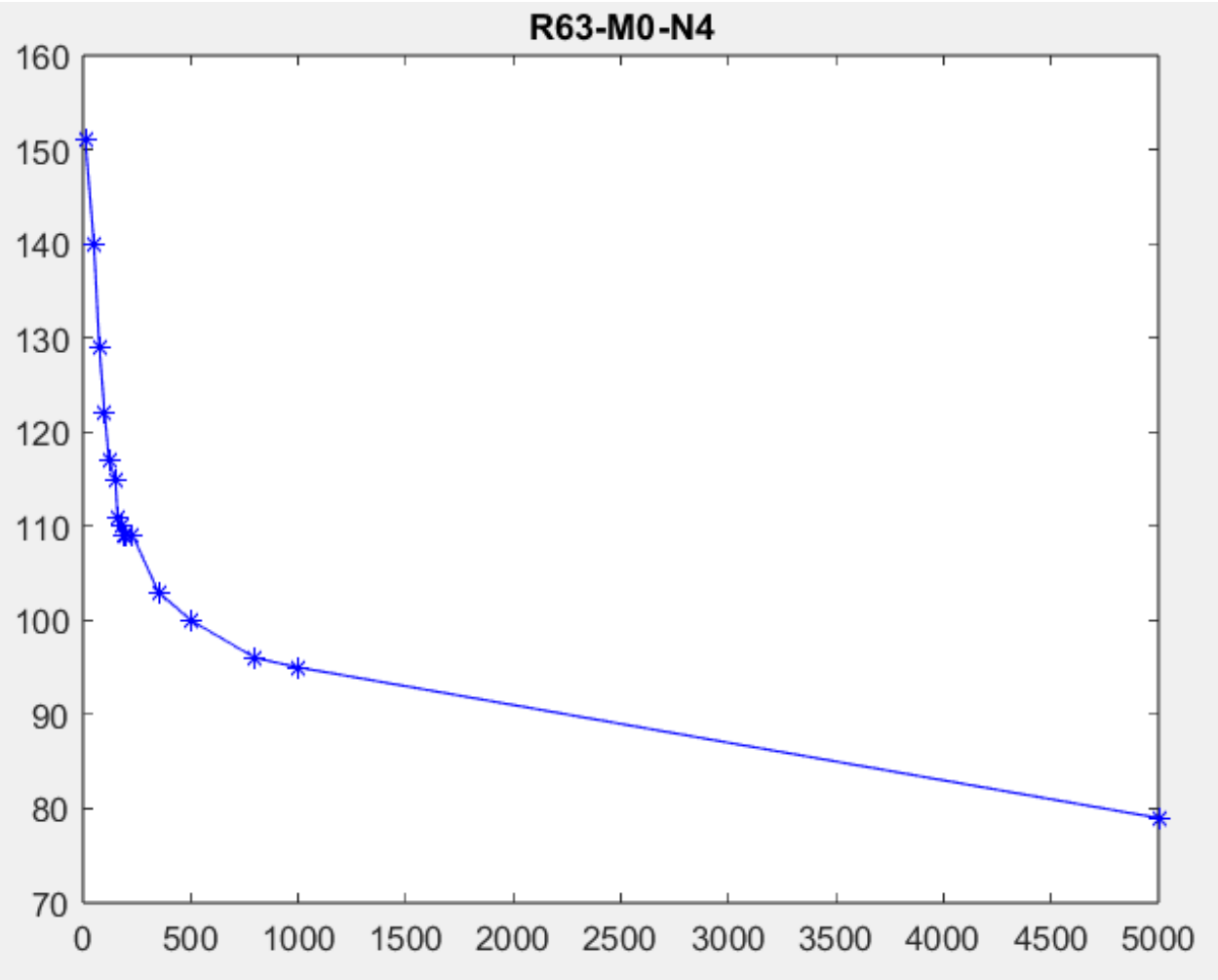

R63-M0-N8
150
140
130
120
110
100
90
80
70
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000

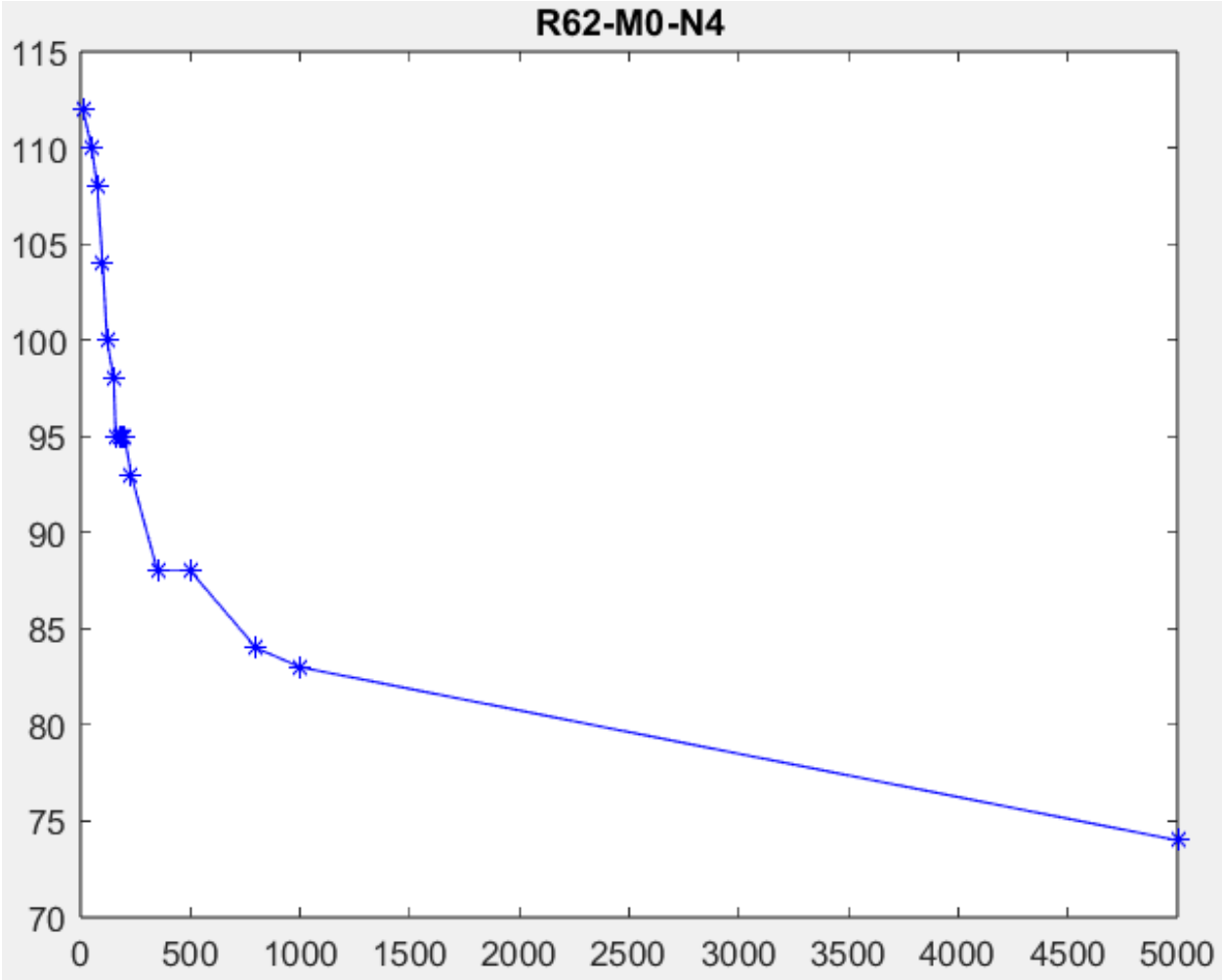

R63-M0-N2
7
6.5
6
5.5
5
4.5
4
3.5
3
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000

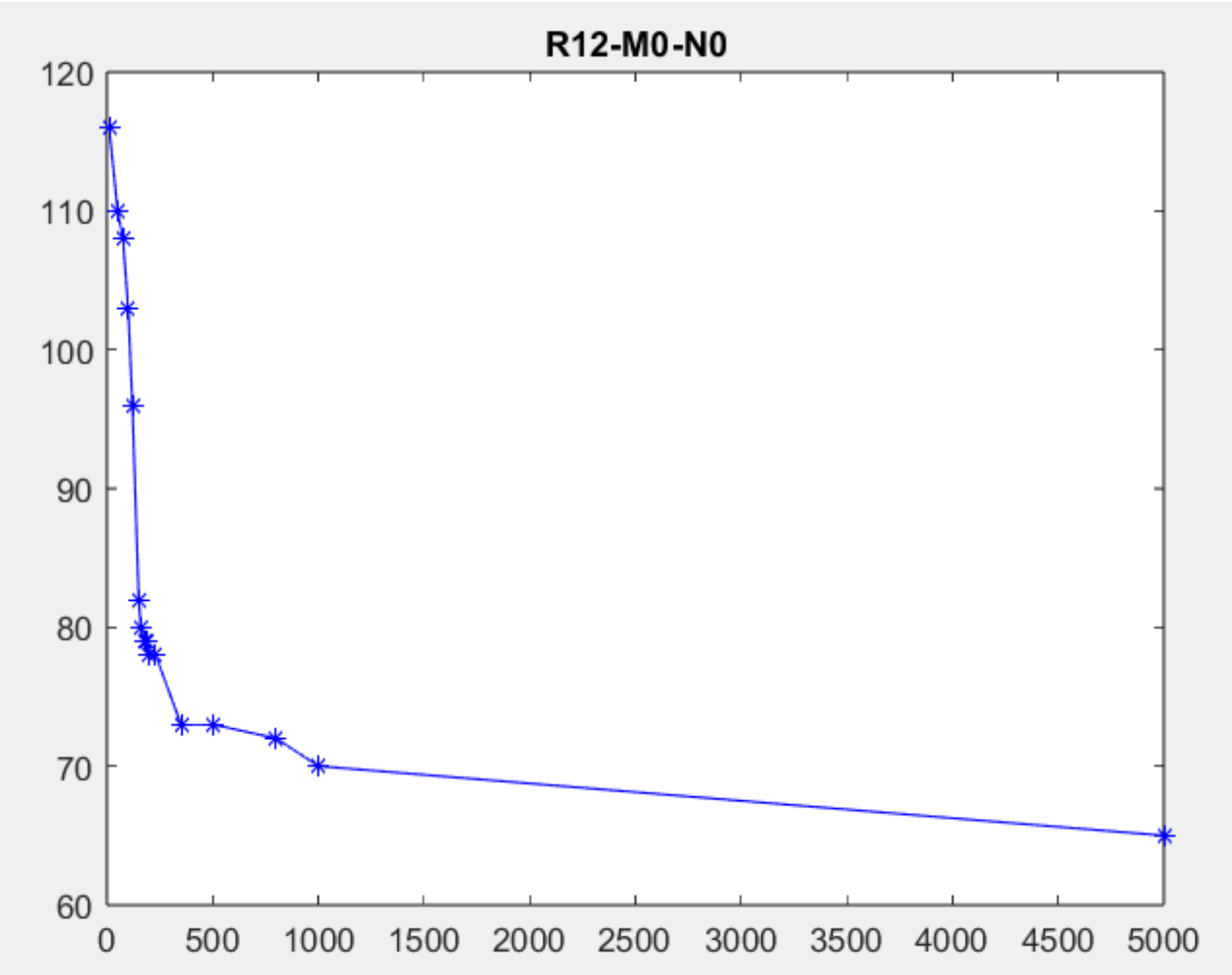

R63-M0-N0
160
150
140
130
120
110
100
90
80
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000

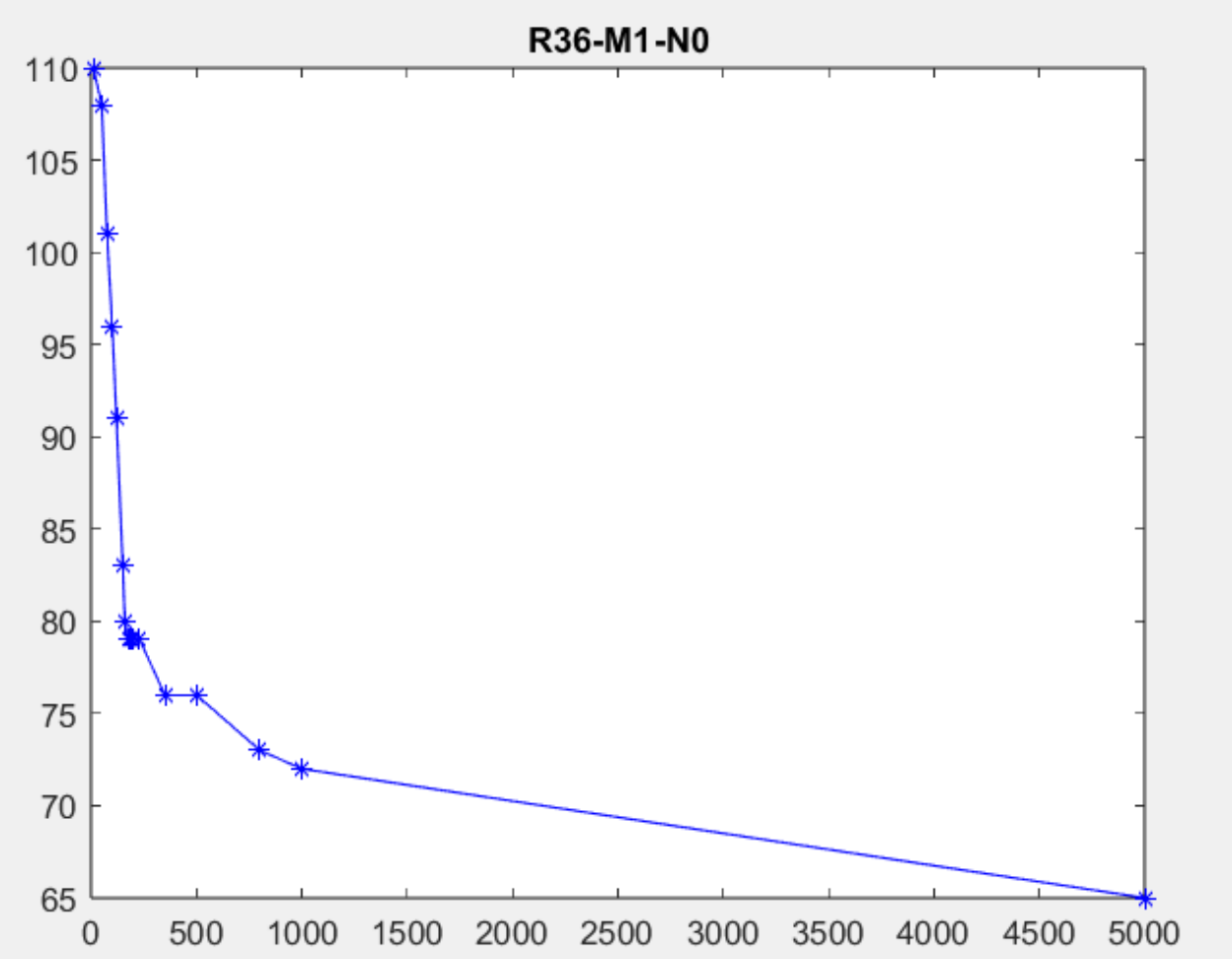

R03-M1-NF
8
7.8
7.6
7.4
7.2
7
6.8
6.6
6.4
6.2
6
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000


R63-M0-NC
160
150
140
130
120
110
100
90
80
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000


R63-M0-N4
160
150
140
130
120
110
100
90
80
70
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000


R62-M0-N4
115
110
105
100
95
90
85
80
75
70
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000


R12-M0-N0
120
110
100
90
80
70
60
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000

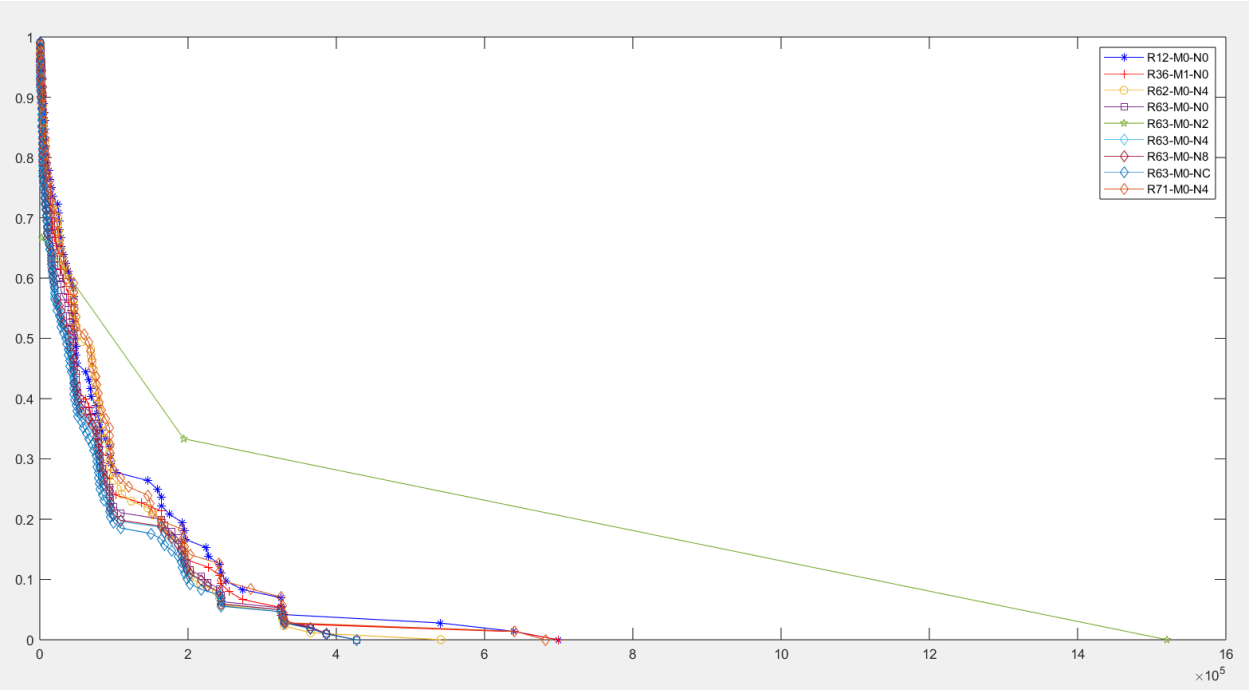


Node	CWIN
R71-M0-N4	350
R63-M0-NC	350
R63-M0-N8	350
R63-M0-N4	350
R63-M0-N2	50
R63-M0-N0	800
R62-M0-N4	350
R36-M1-N0	350
R12-M0-N0	350

Come è possibile vedere, la mediana delle CWIN scelte è di **350s**. Si discosta un po' dal valore scelto in fase di conteggio delle tuple, tuttavia non ci si trova così distanti quindi non è particolarmente strano, considerando che sono stati tralasciati comunque molti nodi dall'analisi.

Da questi valori, è quindi possibile calcolare la **reliability empirica** per ogni nodo, in modo da poter fare delle considerazioni su quali di questi siano gli effettivi **bottleneck** del sistema.

Si riporta quindi il grafico delle reliability:



Da questo grafico non si rilevano dei **bottleneck specifici**, tuttavia è comunque possibile notare qualcosa di molto particolare. La maggior parte dei nodi seguono la **stessa forma di reliability**, che si notino essere **tutti i nodi con la card di I/O aggiuntiva**. L'unico nodo che in effetti si discosta da questo andamento (e che è quello che ha la curva effettivamente maggiore delle altre) è il nodo **R63-M0-N2**, che per l'appunto è un nodo senza la card di I/O aggiuntiva.

È possibile verificare questa cosa ottenendo i report dei log filtrati per nodo:

```
== R12-M0-N0 ==

== Total error entries ==
1563

== Breakup by CATEGORY ==
J12-U01 1365
J18-U11 101
J18-U01 97
```

```
== R36-M1-N0 ==

== Total error entries ==
788

== Breakup by CATEGORY ==
J16-U11 585
J18-U11 103
J18-U01 100
```

```
== R62-M0-N4 ==

== Total error entries ==
515

== Breakup by CATEGORY ==
```

```
J13-U01 261
J18-U11 98
J18-U01 94
...
== R63-M0-N0 ==

== Total error entries ==
791

== Breakup by CATEGORY ==
J18-U11 132
J18-U01 131
J11-U11 118
J11-U01 118
...
== R63-M0-N2 ==

== Total error entries ==
976

== Breakup by CATEGORY ==
J11-U11 118
J11-U01 118
J10-U11 118
J09-U11 118
J09-U01 118
J08-U11 118
...
== R63-M0-N4 ==

== Total error entries ==
452

== Breakup by CATEGORY ==
J18-U11 130
J18-U01 130
...
== R12-M0-N8 ==

== Total error entries ==
454

== Breakup by CATEGORY ==
J18-U11 132
J18-U01 128
J10-U01 8
...
== R36-M1-NC ==

== Total error entries ==
460
```

```
== Breakup by CATEGORY ==
J18-U11 135
J18-U01 133
J17-U11 6
...
```

Come si può notare, ogni nodo che possiede le schede aggiuntive per l'I/O ha **sempre** degli errori ad esse relative ai primi 3 posti degli errori trovati, ordinati per numero di errori decrescenti.

Si può quindi ipotizzare che effettivamente ci sia una grandissima componente di correlazione tra l'unreliability e queste schede aggiuntive, ovvero un **bottleneck dovuto agli errori di I/O**.

Confronti e considerazioni

Dalle analisi effettuate, è possibile giungere a una serie di considerazioni interessanti, sia dal punto di vista dell'analisi interna dei due sistemi sia dal punto di vista di metterli effettivamente a confronto.

Innanzitutto, si vede come, dall'analisi per categorie e nodi, **non sia possibile scegliere la stessa coalescence window** tra nodi o categorie diverse, in quanto presentano parametri di Reliability molto diversi. Di conseguenza, la CWIN calcolata per il sistema complessivo non è un valore ottimale per **ogni** nodo o categoria.

Si sono analizzate inoltre le relazioni tra la **system reliability** e la **reliability dei nodi**, ponendo attenzione a quali fossero i nodi più influenti alla reliability del sistema anche nel senso di **bottleneck**, intesi come nodi che contribuiscono di più al **numero di fallimenti globale**.

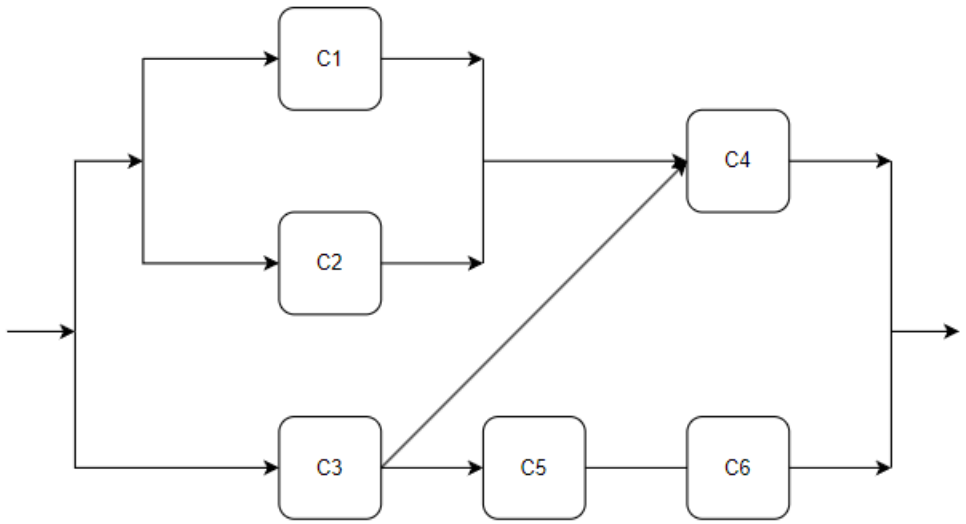
Inoltre, è stato possibile notare dei **pattern per nodi funzionali simili**, sia per quanto riguarda i nodi di Mercury (come ad esempio i nodi di calcolo che avevano tutti comportamenti simili, discostandosi molto da quelli di storage) che per quanto riguarda i nodi di BlueGene (in cui si è notato come i nodi con la scheda aggiuntiva di I/O mostrassero forme molto simili della curva di reliability). Addizionalmente si sono notati anche **pattern tra categorie di errori e nodi**, ad esempio su Mercury si sono notate relazioni tra gli errori DEV e i nodi di calcolo, così come gli errori I-O e i nodi di storage.

Infine, tramite l'analisi del **Mean Time To Failure**, è possibile dire che, nel complesso, **il sistema BlueGene è più reliable del sistema Mercury**, poiché possiede un MTTF maggiore (di circa 3 ore in più).

RBD & Fault Tree

Esercizio 1

Il primo esercizio chiede di trovare la Reliability e il Mean Time To Failure di un sistema descritto dal seguente diagramma:



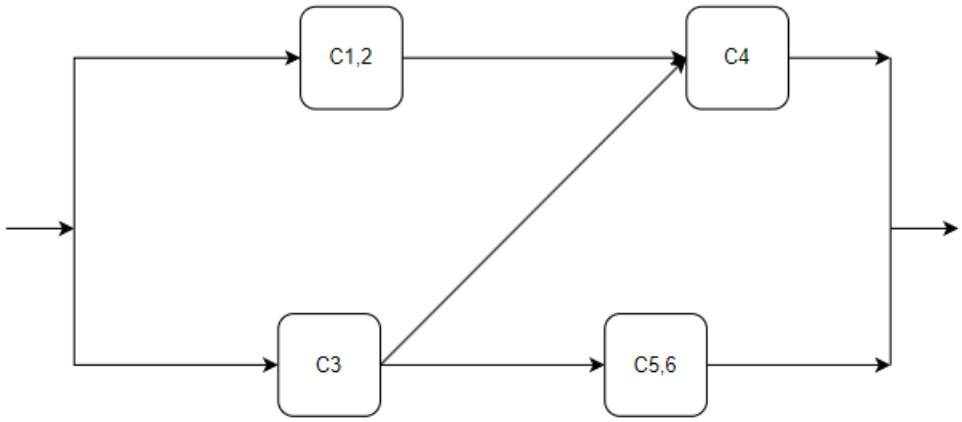
I componenti sono tutti identici e falliscono casualmente con il medesimo failure rate λ.

Da un'analisi preliminare è possibile subito individuare il parallelo tra i componenti 1 e 2 e la serie tra 5 e 6. Applicando le formule seguenti:

$$R_{1,2} = 1 - \prod_{i=1}^{2}(1 - R_i) = 1 - (1 - R_1)(1 - R_2)$$

$$R_{5,6} = \prod_{i=5}^{6} R_i = R_5 R_6$$

Si giunge pertanto allo schema seguente:



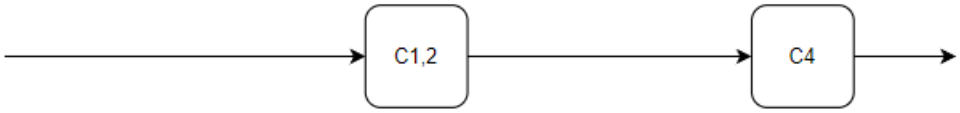
A questo punto vi è la necessità di effettuare un primo "Conditioning", andando a valutare due casistiche: si assumerà che nel primo caso, il componente condizionato funzioni (trattandolo come un cortocircuito), mentre nel secondo caso si assumerà che esso non funzioni (trattandolo come un circuito aperto). Si è deciso di applicare il conditioning al componente 3.

Nel primo caso, si ottiene il seguente circuito:

Individuato il parallelo tra $C_{5,6}$ e C_4, la Reliability del circuito è:

$$R_{4,5,6} = 1 - (1 - R_4)(1 - R_5 R_6)$$

Nel secondo caso, si ottiene il seguente circuito:



La cui Reliability complessiva è:

$$R_{1,2,4} = R_{1,2}R_4 = (1-(1-R_1)(1-R_2))R_4$$

Quindi, la Reliability del sistema originario è:

$$R_{sys} = R_3 P(\text{Il sistema funziona}|C_3 \text{ funziona}) + (1-R_3)P(\text{Il sistema funziona}|C_3 \text{ non funziona}) = R_3 R_{4,5,6} + (1-R_3)R_{1,2,4}$$

Che equivale a:

$$\begin{aligned}
R_{sys} &= R_3(1-(1-R_4)(1-R_5R_6)) + (1-R_3)(1-(1-R_1)(1-R_2))R_4 \\
&= R(1-(1-R)(1-R^2)) + (1-R)(1-(1-R)^2)R \\
&= R(1-(1-R-R^2+R^3)) + (1-R)(1-(1-2R+R^2))R \\
&= R(R+R^2-R^3) + (1-R)(2R-R^2)R \\
&= R^2+R^3-R^4+(2R-2R^2-R^2+R^3)R \\
&= R^2+R^3-R^4+2R^2-3R^3+R^4 \\
&= -2R^3+3R^2
\end{aligned}$$

Dal momento che tutti i componenti hanno una distribuzione dei failure esponenziale, la Reliability del sistema può anche essere espressa come:

$$R_{sys} = -2e^{-3\lambda t} + 3e^{-2\lambda t}$$

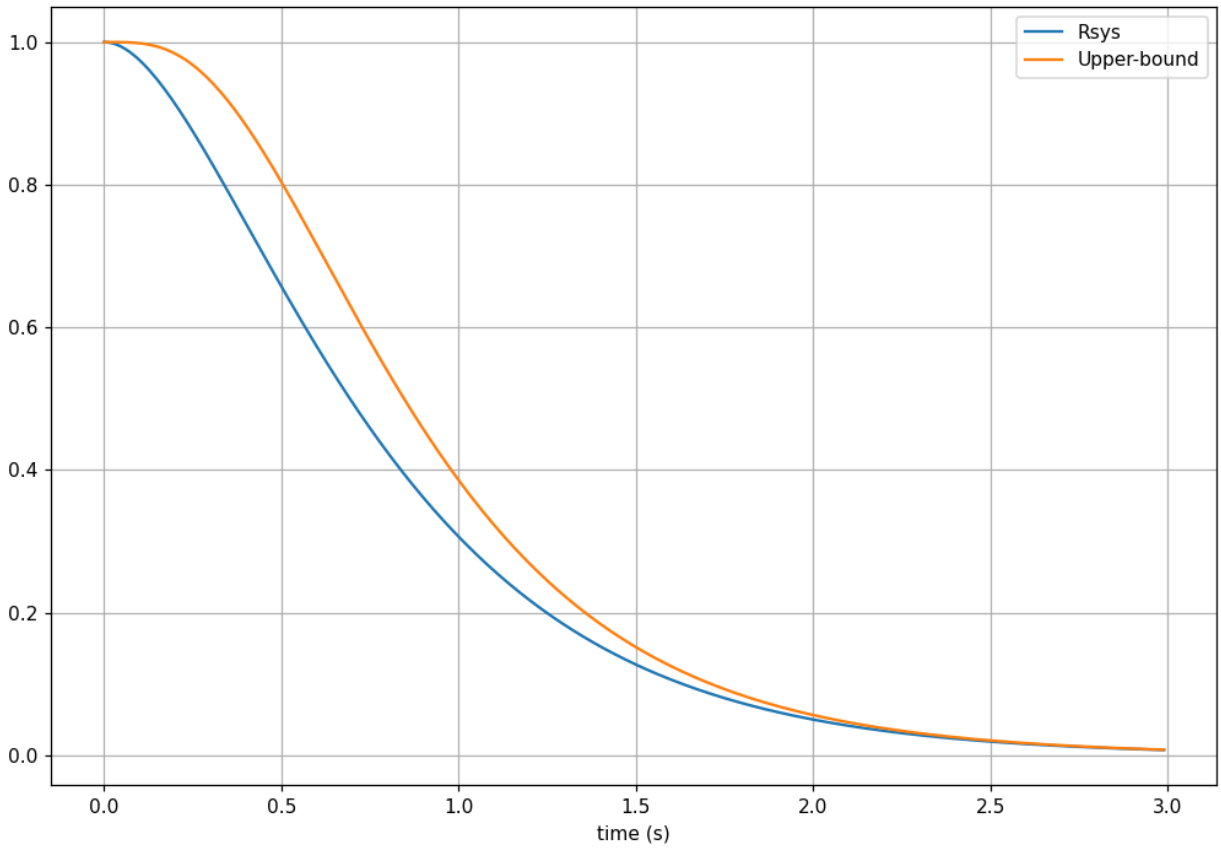
Si passa al calcolo del Mean Time To Failure, integrando la Reliability calcolata poc'anzi:

$$MTTF = \int_0^\infty R_{sys} \cdot dt = \int_0^\infty (-2e^{-3\lambda t} + 3e^{-2\lambda t}) \cdot dt = \left[\frac{2}{3\lambda}e^{-3\lambda t} - \frac{3}{2\lambda}e^{-2\lambda t}\right]_0^\infty = \frac{3}{2\lambda} - \frac{2}{3\lambda} = \frac{9-4}{6\lambda} = \frac{5}{6\lambda}$$

Viene fornita in calce l'applicazione del teorema dell'upper-bound, per validare quanto calcolato in precedenza:

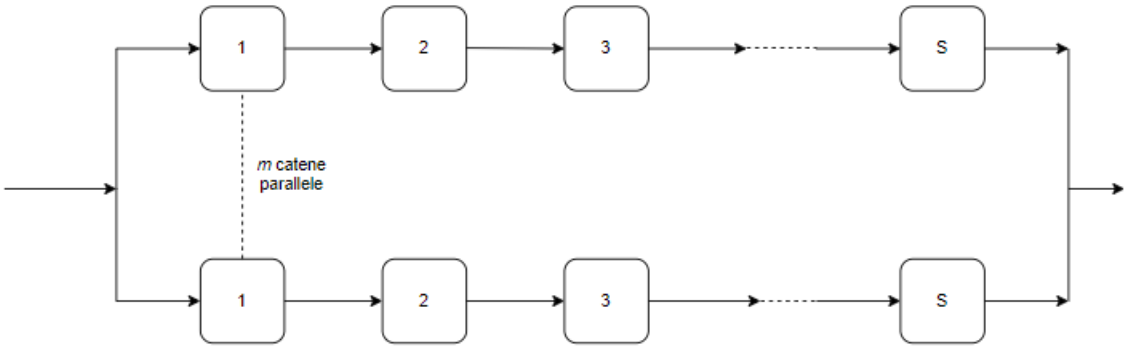
$$\begin{aligned}
R_{sys} &\leq 1 - \prod_i (1 - R_{\text{path } i}) \\
&= 1-(1-R^2)^3(1-R^3) \\
&= 1-(1-R^6+3R^4-3R^2)(1-R^3) \\
&= 1-(1-R^3-R^6+R^9+3R^4-3R^7-3R^2+3R^5) \\
&= 3R^2+R^3-3R^4-3R^5+R^6+3R^7-R^9
\end{aligned}$$

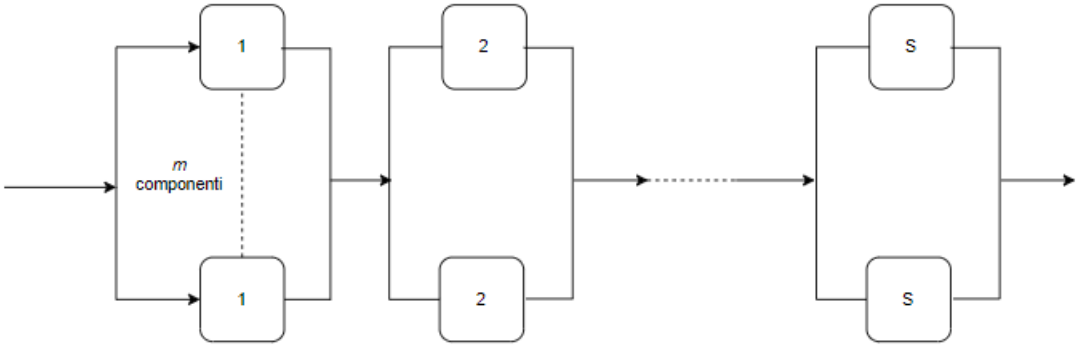
Dal grafico della Reliability calcolata e dell'upper-bound, si può validare il risultato ottenuto, in quanto è effettivamente al di sotto della curva upper-bound:



Esercizio 2

Il secondo esercizio chiede in principio di trovare le espressioni per le Reliability di due sistemi:





Il primo sistema origina da m catene parallele di s componenti; la Reliability individuale di ognuna di esse è:

$$R_{chain}(t) = \prod_{i=1}^{s} R_i = R^s$$

Mentre quella complessiva è data dal parallelo tra le m catene:

$$R_{sys1}(t) = 1 - \prod_{i=1}^{m}(1 - R_{chain,i}) = 1 - (1 - R^s)^m$$

Il secondo sistema origina dalla serie di s paralleli tra m componenti; la Reliability di ciascun parallelo è:

$$R_{block}(t) = 1 - \prod_{i=1}^{m}(1 - R) = 1 - (1 - R)^m$$

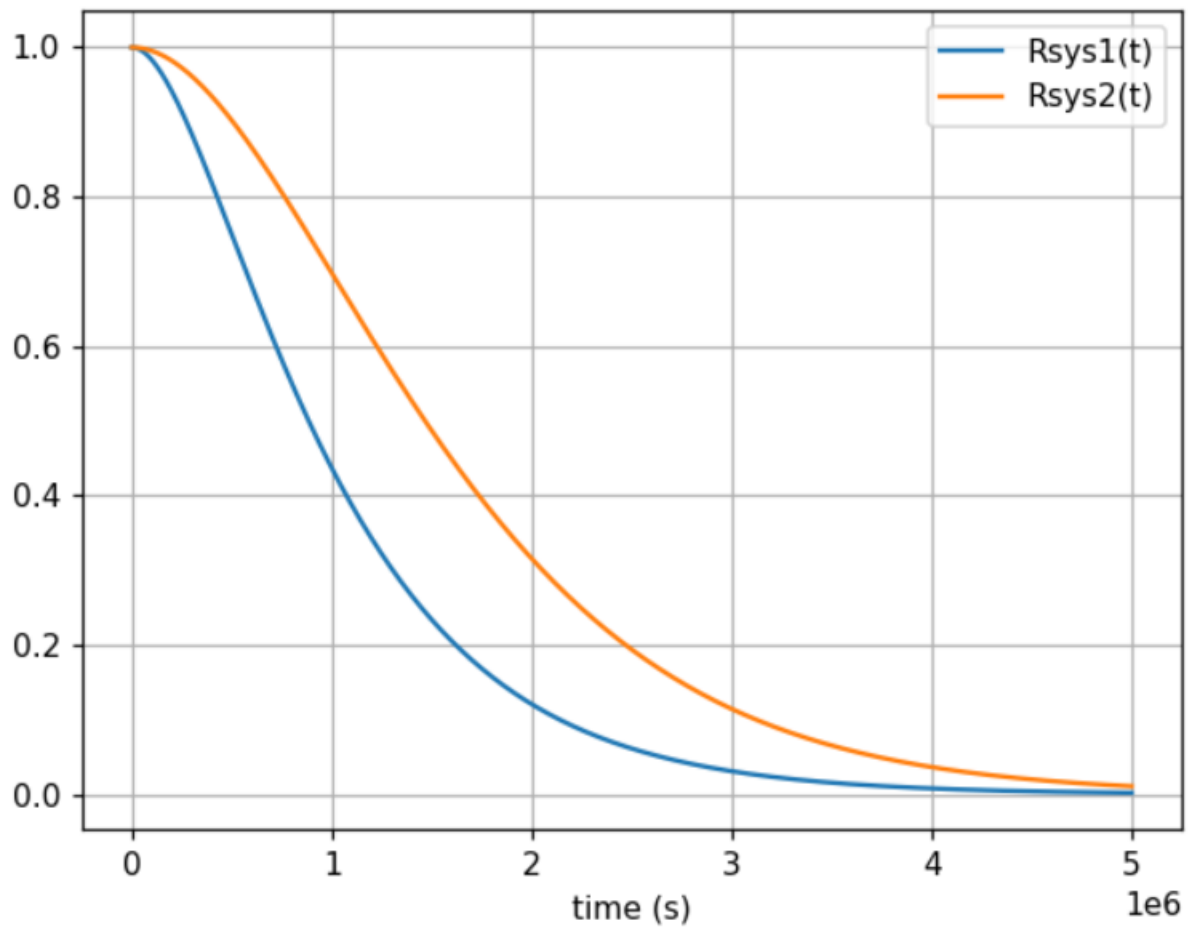
Mentre quella complessiva è data dalla serie tra i blocchi:

$$R_{sys2}(t) = \prod_{i=1}^{s} R_{block,i} = (1 - (1 - R)^m)^s$$

Posto $m = 2$ e $s = 4$, calcoliamo le Reliability ricavate poc'anzi:

$$R_{sys1}(t) = 1 - (1 - R^4)^2 = 1 - (1 - 2R^4 + R^8) = -R^8 + 2R^4$$
$$R_{sys2}(t) = (1 - (1 - R)^2)^4 = R^8 - 8R^7 + 24R^6 - 32R^5 + 16R^4$$

Possiamo sfruttare questi risultati per graficare le Reliability, assumendo il failure rate pari a $\lambda = \frac{1}{MTTF}$:



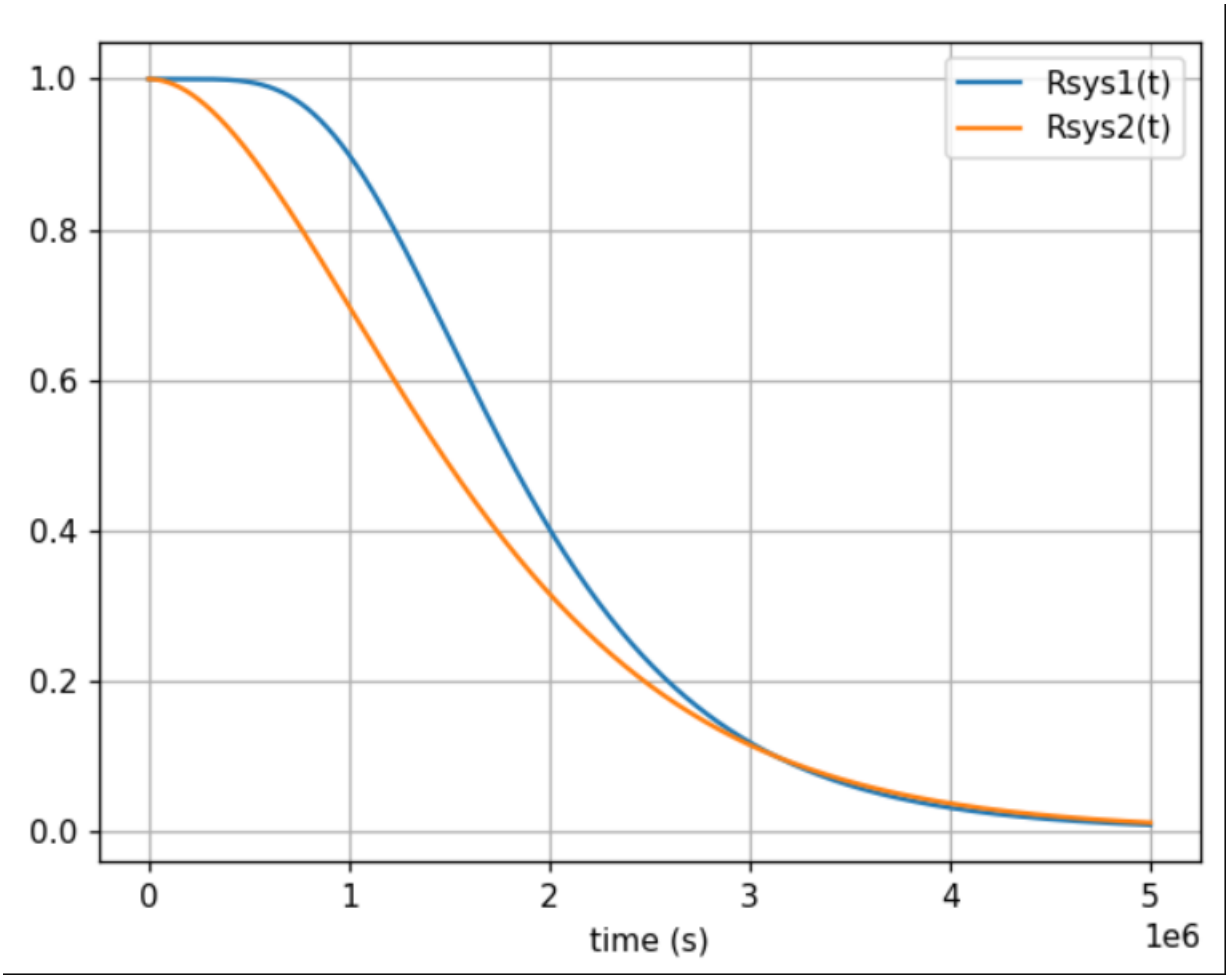
Si può osservare dai grafici che è il secondo sistema a offrire una maggiore Reliability. Ci si pone come obiettivo l'allineamento delle Reliability dei due sistemi. A parità di componenti, abbiamo bisogno di più catene in parallelo per migliorare la Reliability del primo sistema. Per farlo calcoliamo i valori di Reliability per il valore di MTTF assegnato:

$$R_{sys1}(MTTF) = 1 - (1 - e^{-4})^2 = 0.036 = 3.6\%$$
$$R_{sys2}(MTTF) = (1 - (1 - e^{-1})^2)^4 = 0.13 = 13\%$$

Dobbiamo calcolare il numero di catene parallele (m) che ci permetta di ottenere nel primo sistema la stessa Reliability del secondo.

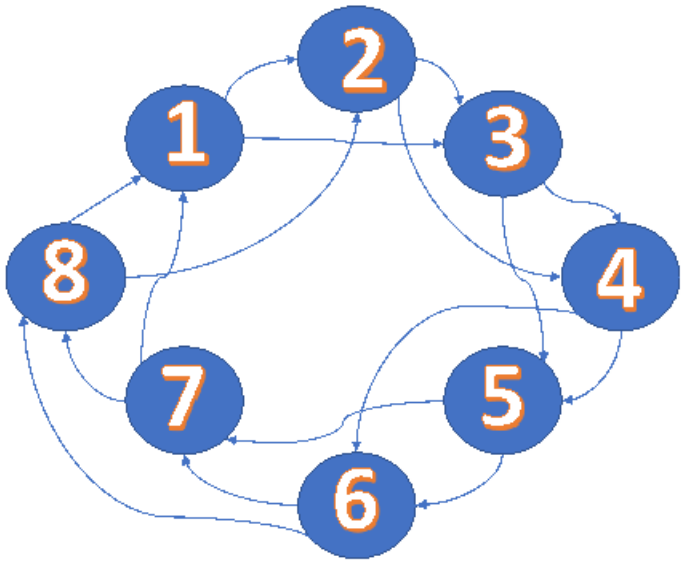
$$R_{sys1} = R_{sys2} \iff 1 - (1 - R^4)^m = 0.13 \iff m \approx 8$$

Lo possiamo apprezzare anche graficamente, osservando che all'istante $2.88 \cdot 10^6 s (= 800h)$ le due curve coincidono:



Esercizio 3

In questo esercizio viene chiesto di derivare la Reliability del sistema seguente:



In questa "skip-ring" network, l'operatività è garantita se almeno 4 nodi su 8 (e.g. 1,3,5,7) operano correttamente, purché non si guastino due nodi consecutivi; si ha dunque a che fare con un sistema "M-out-of-N", e la sua Reliability sarà, assumendo Reliability dei nodi tutte uguali e pari a R_m :

$$R_{MN}(t) = \sum_{i=0}^{N-M} \binom{N}{i} R_m^{N-i}(1-R_m)^i$$

Dove N è il numero di nodi e M il numero di nodi fuori uso; come è stato già fatto notare, non tutte le possibili combinazioni danno però luogo ad un sistema funzionante, e dovranno essere estromesse. Bisogna, a questo punto, valutare quale sia la probabilità che il sistema fallisca al variare del numero di nodi guasti:

Nodi guasti	Combinazioni non funzionanti	Combinazioni totali	P(n nodi falliscono)	P(sistema fallisce | n nodi falliscono)
0	0	1	R_m^8	0
1	0	8	$8R_m^7(1-R_m)$	0
2	8	28	$28R_m^6(1-R_m)^2$	$\frac{8}{28}$
3	40	56	$56R_m^5(1-R_m)^3$	$\frac{40}{56}$
4	68	70	$70R_m^4(1-R_m)^4$	$\frac{68}{70}$
5	Tutte	56	$56R_m^3(1-R_m)^5$	1
6	Tutte	28	$28R_m^2(1-R_m)^6$	1
7	Tutte	8	$8R_m(1-R_m)^7$	1
8	Tutte	0	$(1-R_m)^8$	1

Questo permette, in prima battuta, di calcolare l'Unreliability del sistema; verrà sfruttato il teorema della probabilità totale:

$$F(t) = \sum_{i=0}^{8} P(\text{Sistema in fallimento}|i \text{ nodi in fallimento})\text{P}(i \text{ nodi in fallimento})$$

Osservando che per $i = 0, 1$ non si verifica il fallimento del sistema, a causa della particolare topologia della rete, basta combinare i risultati precedenti:

$$F(t) = 8R_m^6(1-R_m)^2+40R_m^5(1-R_m)^3+68R_m^4(1-R_m)^4+56R_m^3(1-R_m)^5+28R_m^2(1-R_m)^6+8R_m(1-R_m)^7+(1-R_m)^8$$

La Reliability del sistema sarà il complementare di questa probabilità:

$$R(t) = 1 - F(t)$$

Assumendo un failure rate $\lambda = 0.005$, si troverà con i risultati ottenuti finora la Reliability del sistema dopo 48 ore. Poiché $R_m(t) = e^{-0.005t}$, si ottiene:

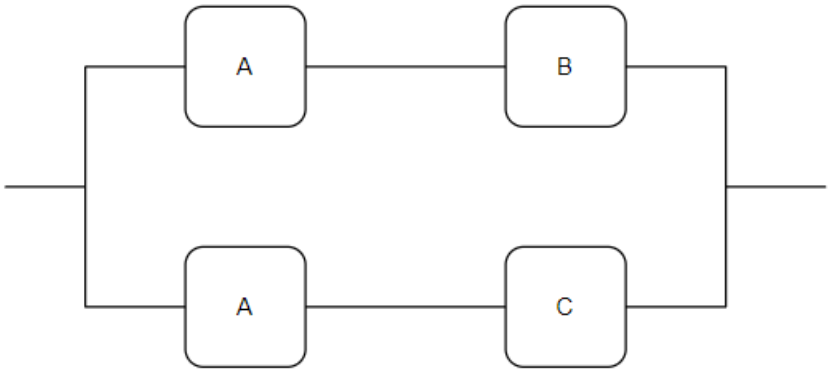
$$R(48h) \approx 0.7289$$

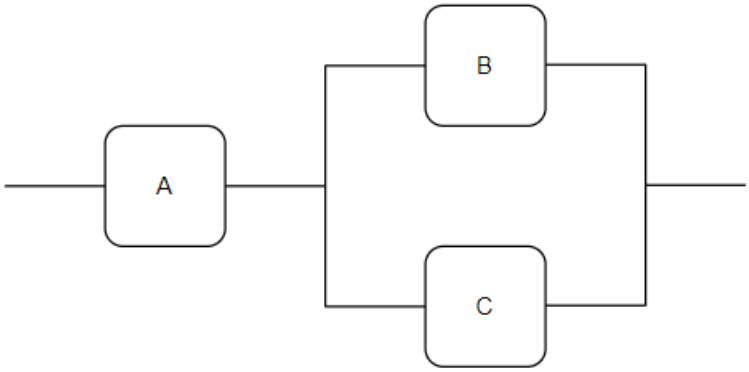
Esercizio 4

Si deve comparare la Reliability di 4 sistemi, assumendo una distribuzione dei failure esponenziale e i seguenti MTTF:

- $MTTF_A = 500h$.
- $MTTF_B = 9000h$.
- $MTTF_C = 10000h$.

Prima coppia





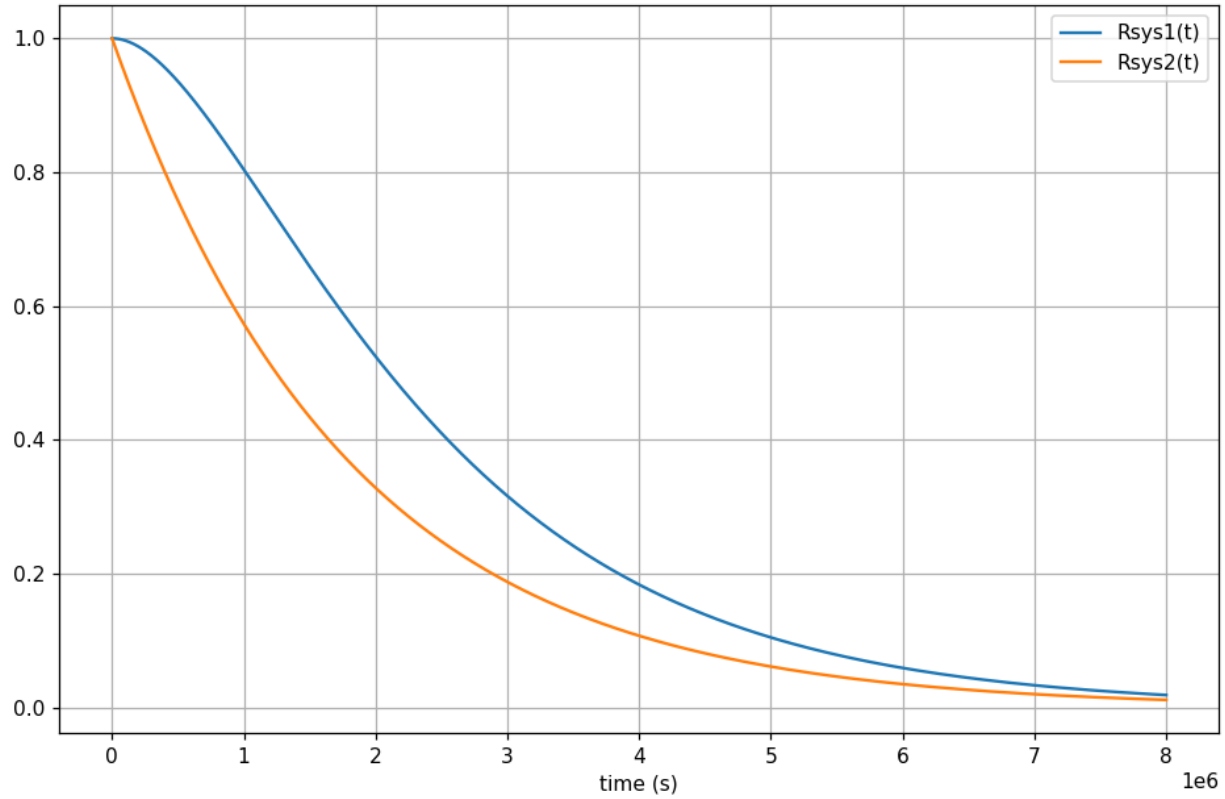
Le espressioni per le Reliability dei due sistemi sono:

$$R_{sys1}(t) = 1 - (1 - R_A R_B)(1 - R_A R_C)$$
$$R_{sys2}(t) = (1 - (1 - R_B)(1 - R_C))R_A$$

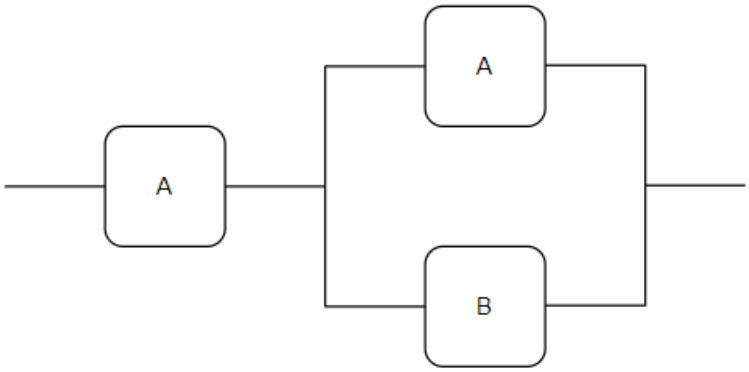
In ragione delle assunzioni fatte in principio si ottengono

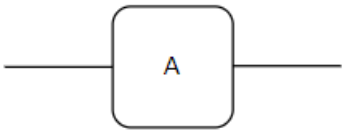
$$R_{sys1}(t) = 1 - (1 - e^{-(\frac{t}{MTTF_A} + \frac{t}{MTTF_B})})(1 - e^{-(\frac{t}{MTTF_A} + \frac{t}{MTTF_C})})$$
$$R_{sys2}(t) = (1 - (1 - e^{-\frac{t}{MTTF_B}})(1 - e^{\frac{t}{MTTF_C}}))(e^{-\frac{t}{MTTF_A}})$$

Si deduce dai seguenti grafici che è maggiore la Reliability del **primo** sistema:



Seconda coppia





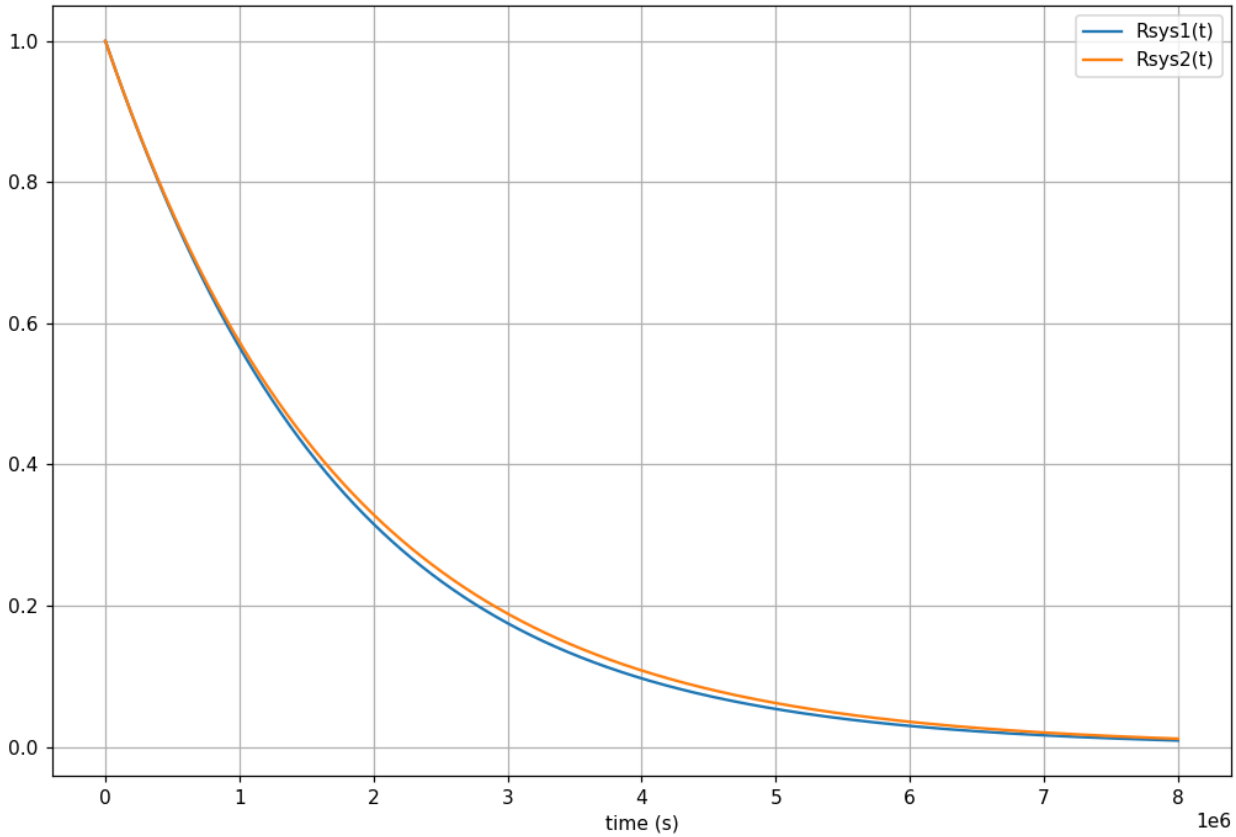
Le espressioni per le Reliability dei due sistemi sono:

$$R_{sys1}(t) = (1 - (1 - R_B)(1 - R_A))R_A$$
$$R_{sys2}(t) = R_A$$

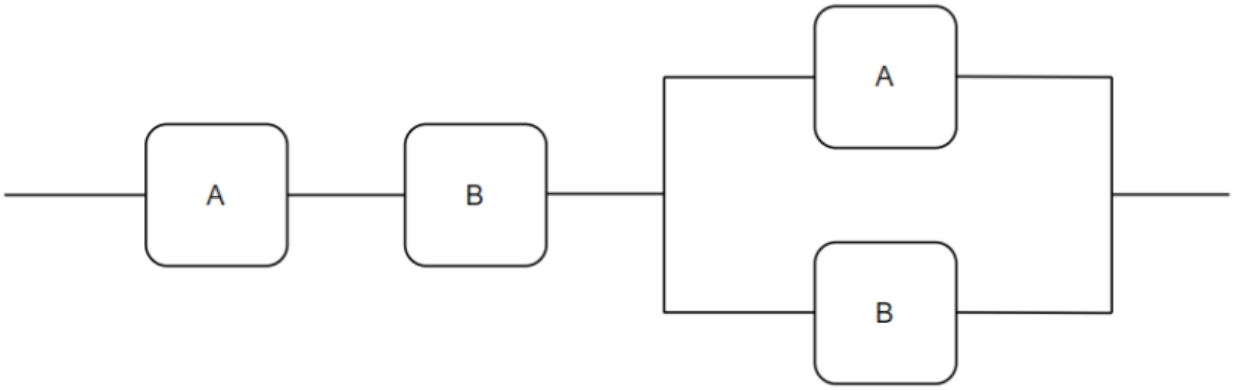
In ragione delle assunzioni fatte in principio si ottengono:

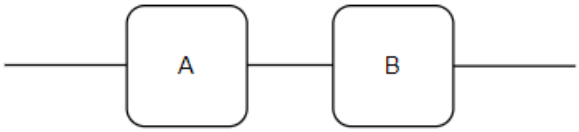
$$R_{sys1}(t) = (1 - (1 - e^{-\frac{t}{MTTF_B}})(1 - e^{\frac{t}{MTTF_A}}))(e^{-\frac{t}{MTTF_A}})$$
$$R_{sys2}(t) = e^{-\frac{t}{MTTFA}}$$

Si deduce dai seguenti grafici che è maggiore la Reliability del **secondo** sistema:



Terza coppia





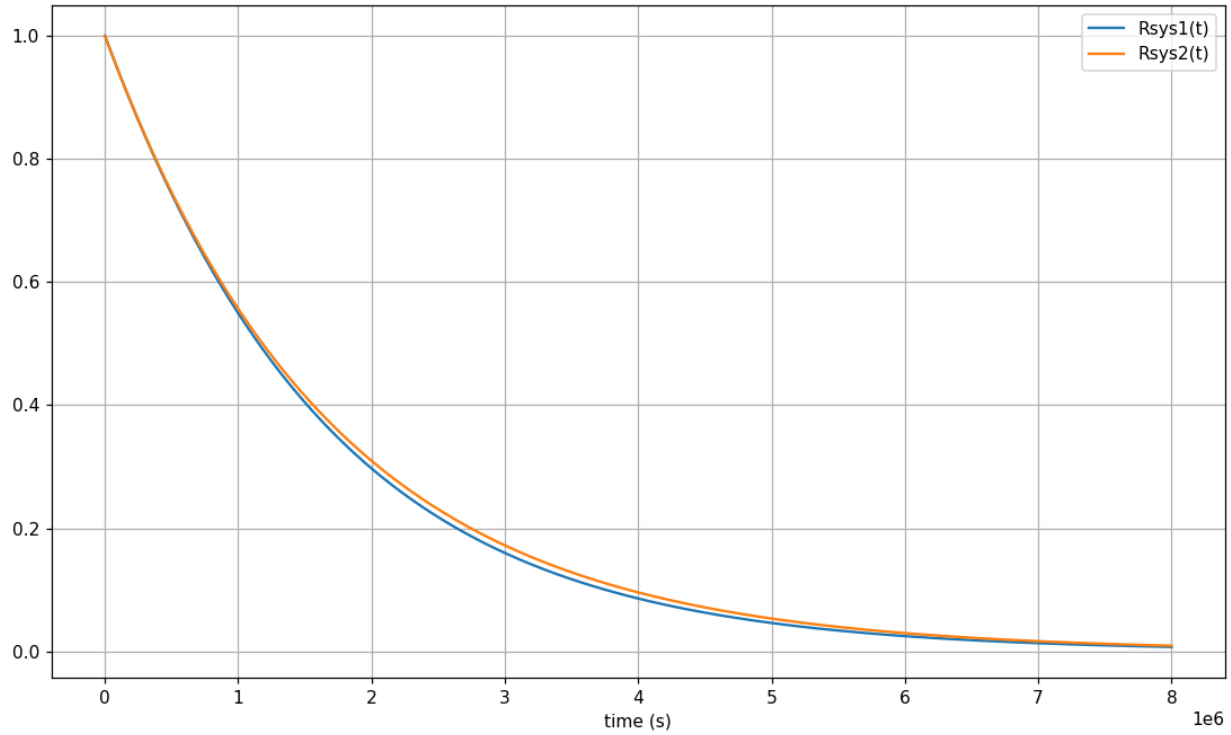
Le espressioni per le Reliability dei due sistemi sono:

$$R_{sys1}(t) = (1 - (1 - R_B)(1 - R_A))(R_A R_B)$$
$$R_{sys2}(t) = R_A R_B$$

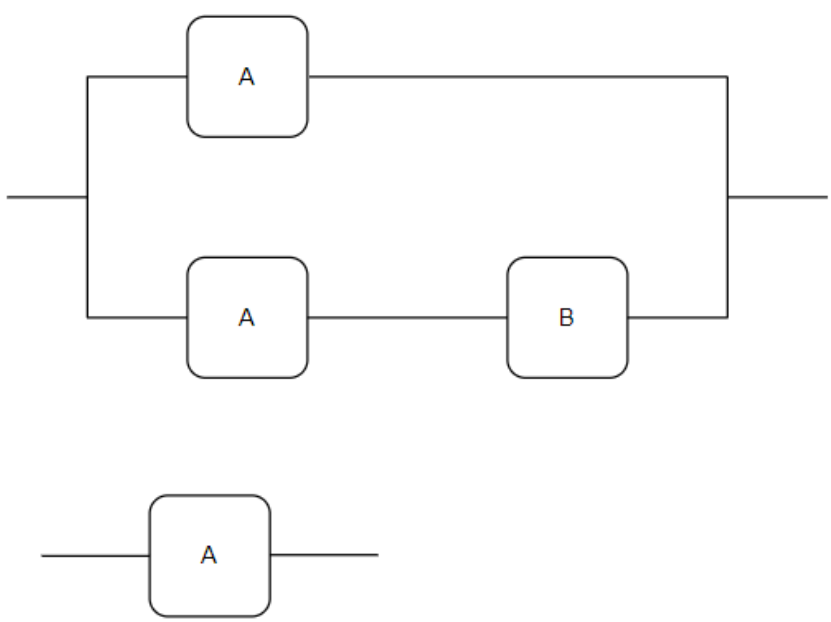
In ragione delle assunzioni fatte in principio si ottengono:

$$R_{sys1}(t) = (1 - (1 - e^{-\frac{t}{MTTF_B}})(1 - e^{\frac{t}{MTTF_A}}))(e^{-(\frac{t}{MTTF_A} + \frac{t}{MTTF_B})})$$
$$R_{sys2}(t) = e^{-(\frac{t}{MTTF_A} + \frac{t}{MTTF_B})}$$

Si deduce dai seguenti grafici che è maggiore la Reliability del **secondo** sistema:



Quarta coppia



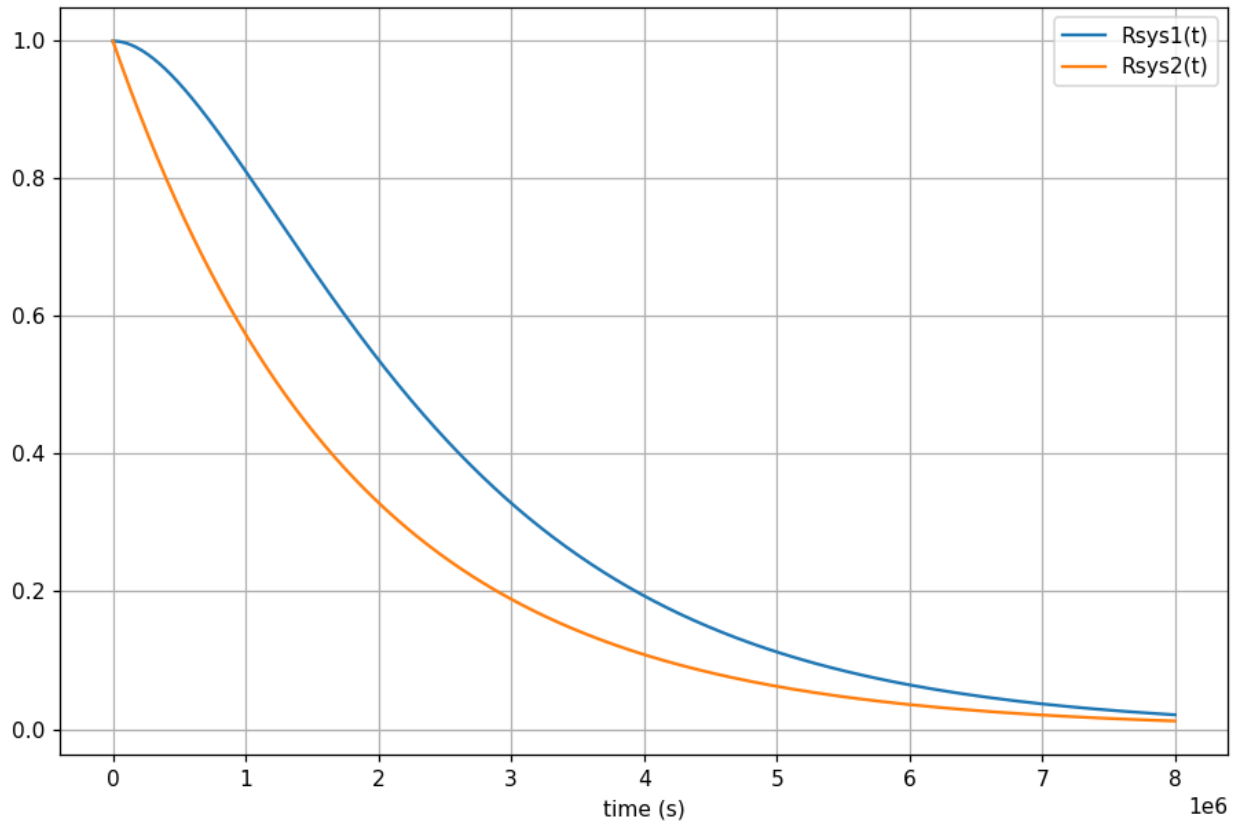
Le espressioni per le Reliability dei due sistemi sono:

$$R_{sys1}(t) = 1 - (1 - R_A R_B)(1 - R_A)$$
$$R_{sys2}(t) = R_A$$

In ragione delle assunzioni fatte in principio si ottengono:

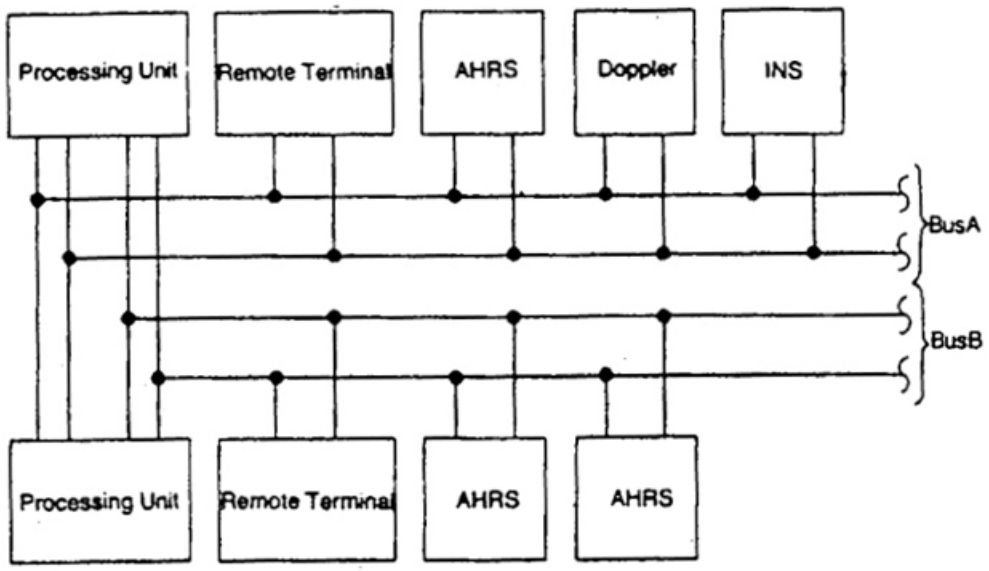
$$R_{sys1}(t) = 1 - (1 - e^{-(\frac{t}{MTTF_A} + \frac{t}{MTTF_B})})(1 - e^{\frac{t}{MTTF_A}})$$
$$R_{sys2}(t) = e^{-(\frac{t}{MTTF_A})}$$

Si deduce dai seguenti grafici che è maggiore la Reliability del **primo** sistema:



Esercizio 5

Viene assegnato il seguente apparato di pilotaggio di un elicottero:

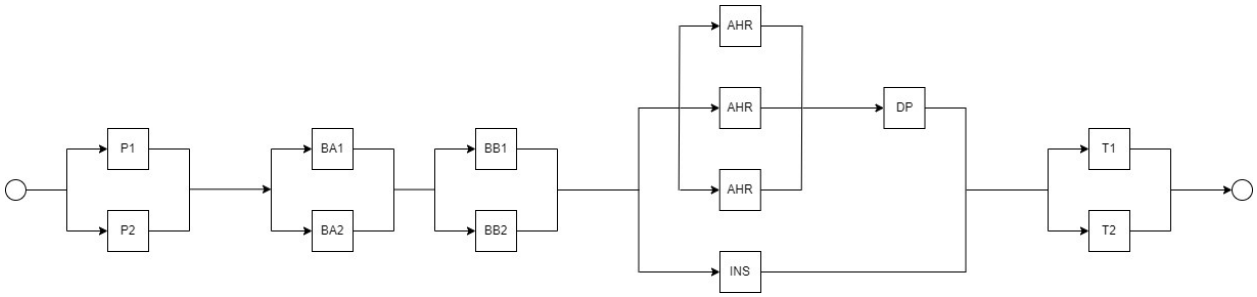


Verrà nel seguito utilizzata la seguente simbologia:

Simbolo	Significato
Px	Processing Unit
BAx	Bus A
BBx	Bus B
AHR	Altitude Heading and Reference System
DP	Doppler
INS	Inertial Navigation System
Tx	Remote Terminal

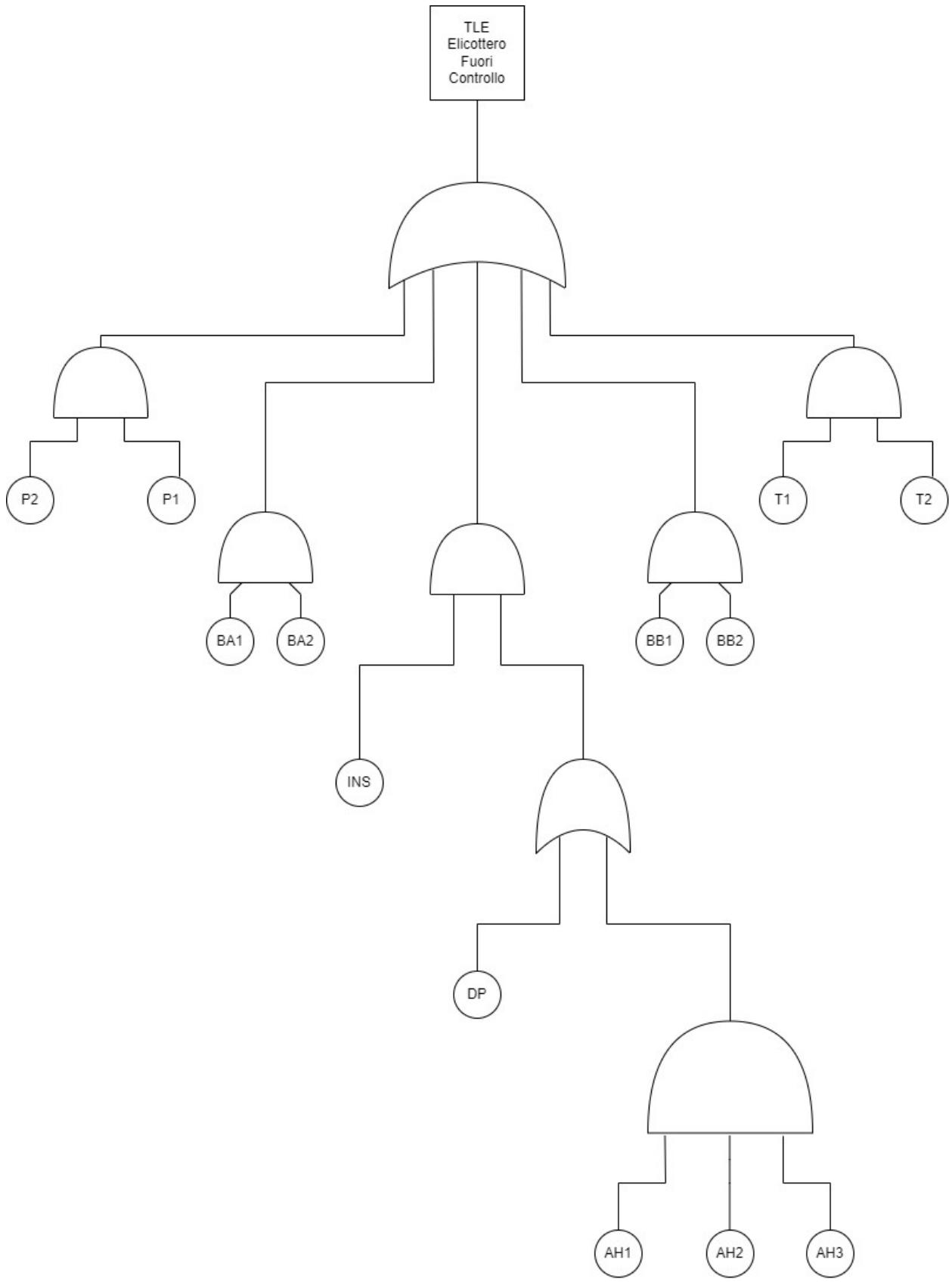
Punto A

Viene chiesto di realizzarne un Reliability Block Diagram:



Punto B

Viene richiesto di realizzare un Fault Tree per il sistema in esame; sapendo che ad ogni parallelo corrisponde una porta AND e ad ogni serie una porta OR, si ottiene il seguente schema:


TLE
Elicottero
Fuori
Controllo
P2
P1
T1
T2
BA1
BA2
BB1
BB2
INS
DP
AH1
AH2
AH3

Si devono trovare i minimal cut set, ovvero i percorsi a lunghezza minima che determinano il fallimento del sistema. Per farlo, viene considerata la funzione di uscita:

$$\Phi(x) = (P_1 \wedge P_2) \vee (BA_1 \wedge BA_2) \vee (BB_1 \wedge BB_2) \vee (INS \wedge (DP \vee (AH_1 \wedge AH_2 \wedge AH_3))) \vee (T_1 \wedge T_2)$$

I minimal cut set sono pertanto i seguenti:

- $(P_1 \wedge P_2)$.
- $(BA_1 \wedge BA_2)$.
- $(BB_1 \wedge BB_2)$.
- $(INS \wedge (DP \vee (AH_1 \wedge AH_2 \wedge AH_3)))$.
- $(T_1 \wedge T_2)$.

Punto C

Viene chiesto di calcolare la Reliability del sistema dopo un'ora di viaggio, assumendo distribuzione esponenziale dei failure e perfetta coverage dei guasti, e usando i seguenti MTTF:

Equipaggiamento	MTTF(h)
Processing Unit	10000
Remote Terminal	4500
AHRS	2000
INS	2000
Doppler	500
Bus	60000

$$R(t) = (1-(1-R_P)^2)(1-(1-R_B)^2)^2(1-(1-R_{INS})(1-R_D(1-(1-R_{AHRS})^3))(1-(1-R_T)^2)$$

Ricavata l'espressione per la Reliability, si può ottenere il suo valore dopo un'ora di volo:

$$R(1h) \approx 0.99999$$

Il sistema offre 5 nines di Reliability.

Punto D

Viene adesso chiesto di incorporare un fattore di coverage incognito nella detezione e correzione di errori relativi alle processing unit; si deve incorporare questo fattore in serie ad un'unità di processing, modificando la formula e risolvendo la seguente equazione:

$$R(1h) = (1-(1-R_P)(1-R_P \cdot c))(1-(1-R_B)^2)^2(1-(1-R_{INS})(1-R_D(1-(1-R_{AHRS})^3))(1-(1-R_T)^2) = 0.99999$$
$$\iff c = 0.910079$$

Si riesce quindi ad individuare il 91% dei guasti nel sistema in esame.