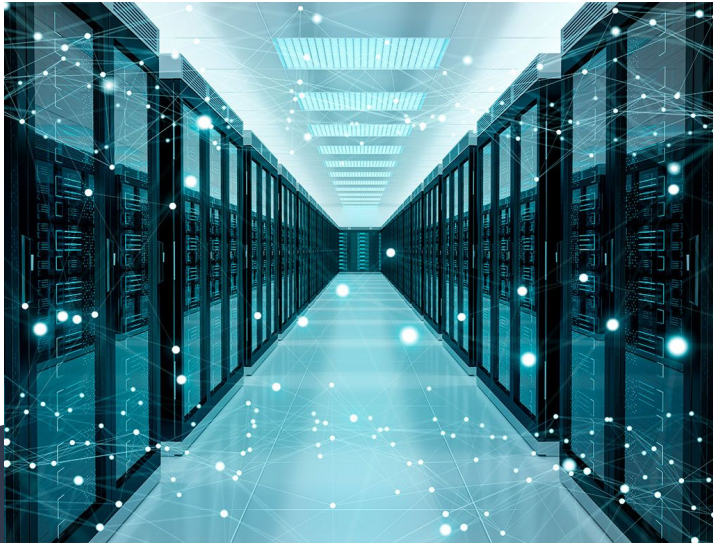


# Optimisation énergétique des datacenters en utilisant un algorithme de bandits



Stage effectué par : Alexandre Gary - M1 MAPI3  
Encadrement : Emmanuelle CLAEYS / Georges DA COSTA

# Sommaire :

- Contexte : Algorithmes de bandits
- Présentation des expériences et des données
- Résultats sur données résumées
- Résultats données détaillées

# Introduction : Algorithmes de bandits

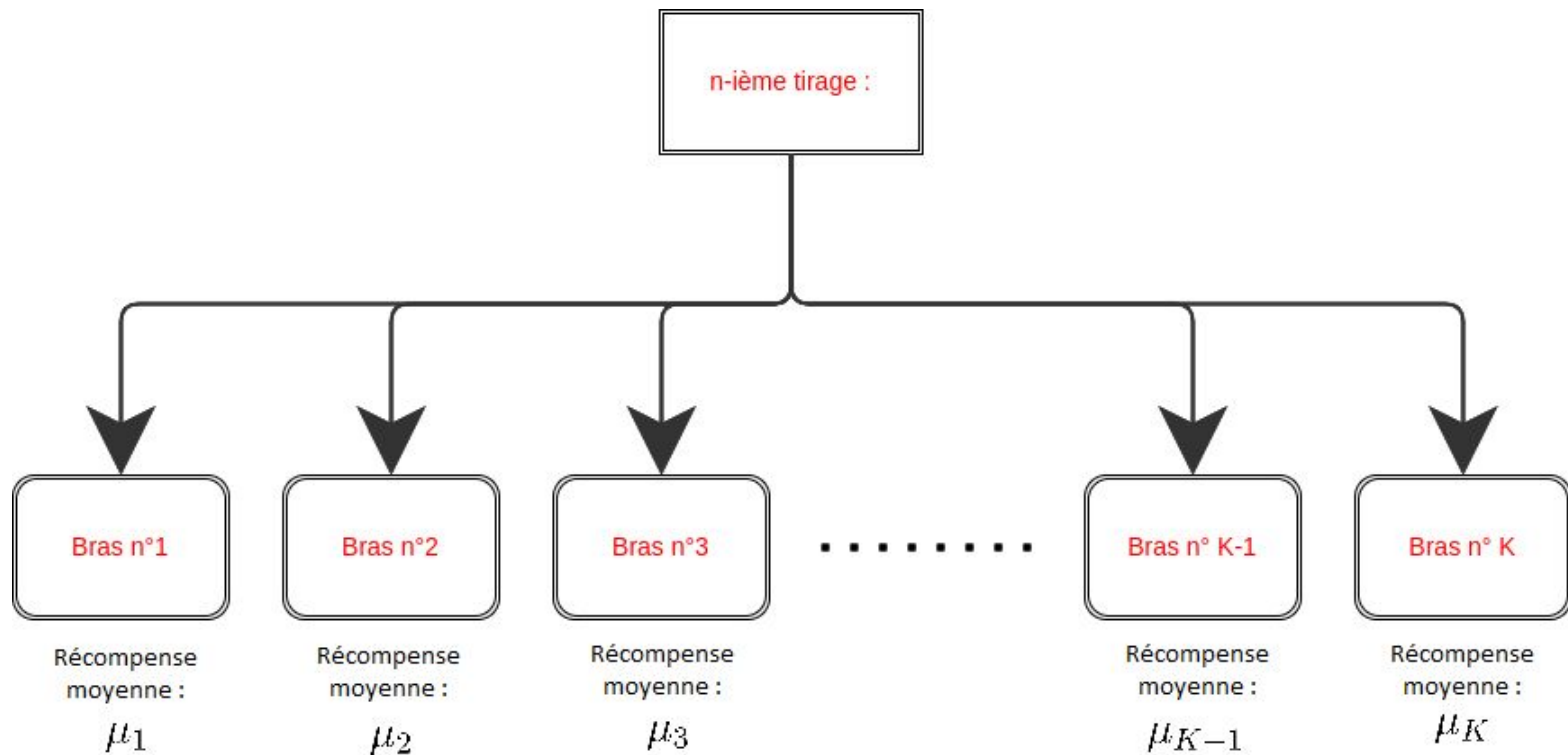
Algorithmes d'apprentissage par  
renforcement

Introduits dans les années 1960

Au départ : pour tests cliniques

Mesure résultat : notion de regret

# Algorithmes de bandits : Fonctionnement



# Allocation uniforme :

---

**Algorithme 1** Bandits : Allocation uniforme

---

**Entrée:** Tableau contenant les récompenses pour chaque bras à chaque itération, et  $K \in \mathbb{N}^*$  :  
Nombre de bras

**Sortie:** Nombre de choix de chaque bras, et estimation des récompenses moyennes

C : Liste contenant les choix à chaque itération

R : Liste contenant les récompenses moyennes obtenues pour chaque bras

Tirer successivement une fois chaque bras

Incrémenter la liste C avec chaque choix

Incrémenter la liste R avec la récompense obtenue pour chaque bras

**pour**  $n \in \{K + 1, \dots, N\}$  **faire**

    Tirer une variable aléatoire uniforme  $a$  sur  $\{1, \dots, K\}$

    Sélectionner le bras  $a$

    Stocker le choix  $a$  dans la liste C

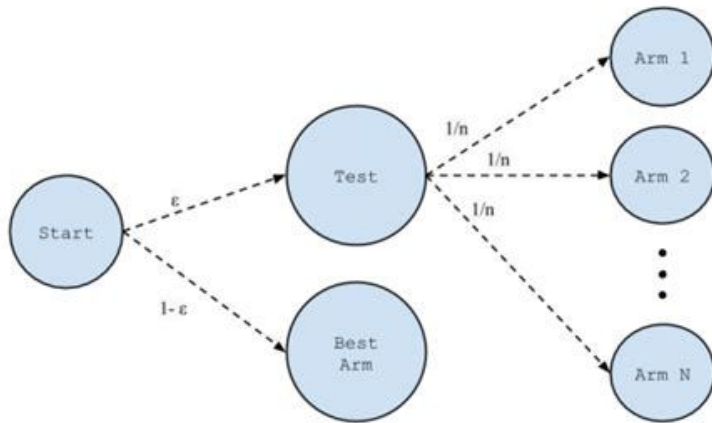
    Mettre à jour les récompenses moyennes dans la liste R

**fin pour**

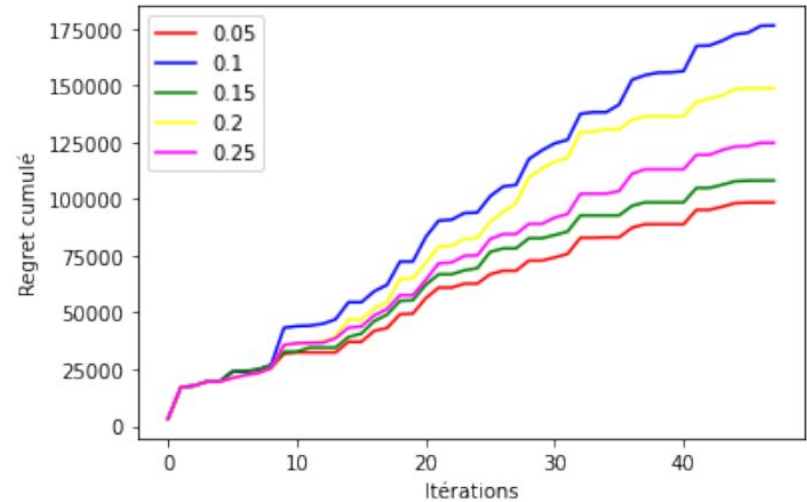
---

# Epsilon - greedy : idée générale

Schéma algorithme :



Regret pour différents epsilon :



# Epsilon – greedy : Algorithme détaillé

---

## Algorithme 2 Bandits : $\epsilon$ - greedy

---

**Entrée:** Tableau contenant les récompenses pour chaque bras à chaque itération,  $K \in \mathbb{N}^*$  :  
Nombre de bras et  $\epsilon \in [0, 1]$  : probabilité d'exploration.

**Sortie:** Nombre de choix de chaque bras, et estimation des récompenses moyennes

C : Liste contenant les choix à chaque itération

R : Liste contenant les récompenses moyennes obtenues pour chaque bras

Tirer successivement une fois chaque bras

Incrémenter la liste C avec chaque choix

Incrémenter la liste R avec la récompense obtenue pour chaque bras

**pour**  $n \in \{K + 1, \dots, N\}$  **faire**

    Tirer une variable de Bernoulli X de paramètre  $1 - \epsilon$

**si**  $X = 1$  **alors**

        Choisir le bras ayant eu les meilleurs résultats jusque-là

        Stocker le choix dans la liste C

        Mettre à jour la liste des récompenses moyennes R

**sinon**

        Choisir uniformément un bras parmi les  $K - 1$  autres

        Stocker le choix dans la liste C

        Mettre à jour la liste des récompenses moyennes R

**fin si**

**fin pour**

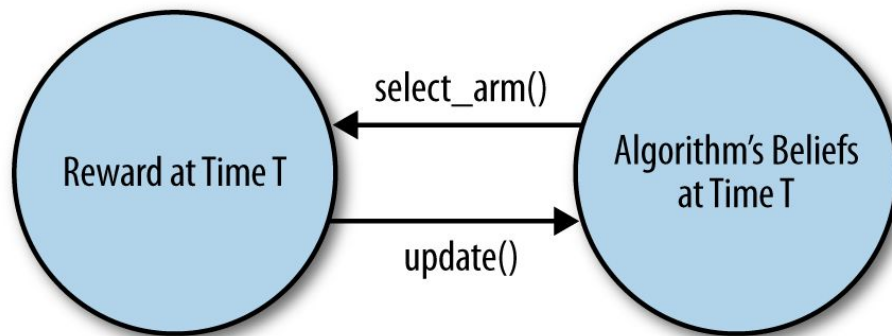
---

# Algorithme UCB : Idée générale

Intervalle de confiance :

$$IC_{b,n} = \left[ m_b \pm \alpha \sqrt{\frac{2 \log(n)}{e_b}} \right]$$

Schéma algorithme :





# Algorithme UCB : Algorithme détaillé

---

## Algorithme 3 Bandits : UCB

---

**Entrée:** Tableau contenant les récompenses pour chaque bras à chaque itération, et  $K \in \mathbb{N}^*$  :

Nombre de bras

**Sortie:** Nombre de choix de chaque bras, et estimation des récompenses moyennes

C : Liste contenant les choix à chaque itération

I : Liste contenant les intervalles de confiances obtenus pour la récompense moyenne associée à chaque bras

Incrémenter la liste C avec chaque choix

Incrémenter la liste I avec l'intervalle de confiance obtenu pour chaque bras

**pour**  $n \in \{K + 1, \dots, N\}$  **faire**

Choisir le bras  $a$  ayant la borne supérieure de l'intervalle de confiance la plus élevée

Stocker le choix dans la liste C

Mettre à jour la liste des intervalles de confiance I

**fin pour**

Récupérer les récompenses moyennes pour chaque bras à partir de la liste des intervalles de confiance I

---

# Algorithme LINUCB : Algorithme détaillé

---

## Algorithme 4 Bandits : LINUCB

---

**Entrée:** Tableau contenant les récompenses pour chaque bras à chaque itération, et  $K \in \mathbb{N}^*$  :

Nombre de bras

**Sortie:** Nombre de choix de chaque bras, et estimation des récompenses moyennes

C : Liste contenant les choix à chaque itération

I : Liste contenant les intervalles de confiances obtenus pour la récompense moyenne associée à chaque bras

Incrémenter la liste C avec chaque choix

Incrémenter la liste I avec l'intervalle de confiance obtenu pour chaque bras

**pour**  $n \in \{K + 1, \dots, N\}$  **faire**

Création du contexte en utilisant une régression linéaire

Choisir le bras  $a$  ayant les meilleurs résultats potentiels en fonction de la régression effectuée

Stocker le choix dans la liste C

Mettre à jour la liste des intervalles de confiance I

**fin pour**

Récupérer les récompenses moyennes pour chaque bras à partir de la liste des intervalles de confiance I

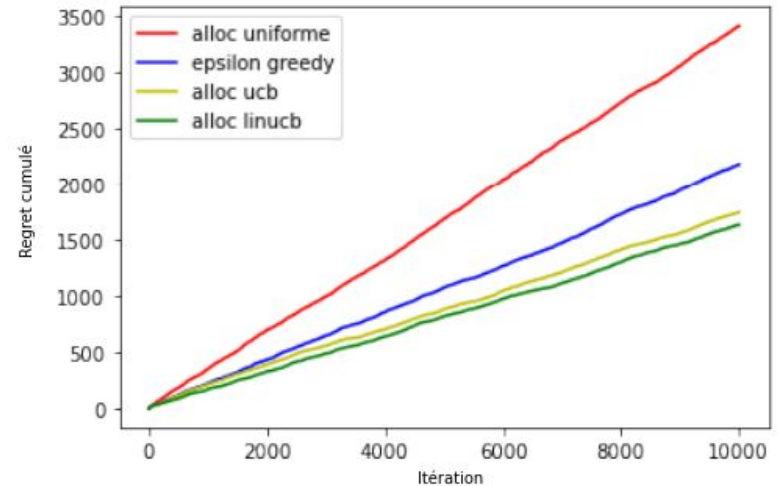
---

# Notion de regret :

Formule regret :

$$R_n = n\mu^* - \sum_{t=1}^n \mu_{I_t}$$

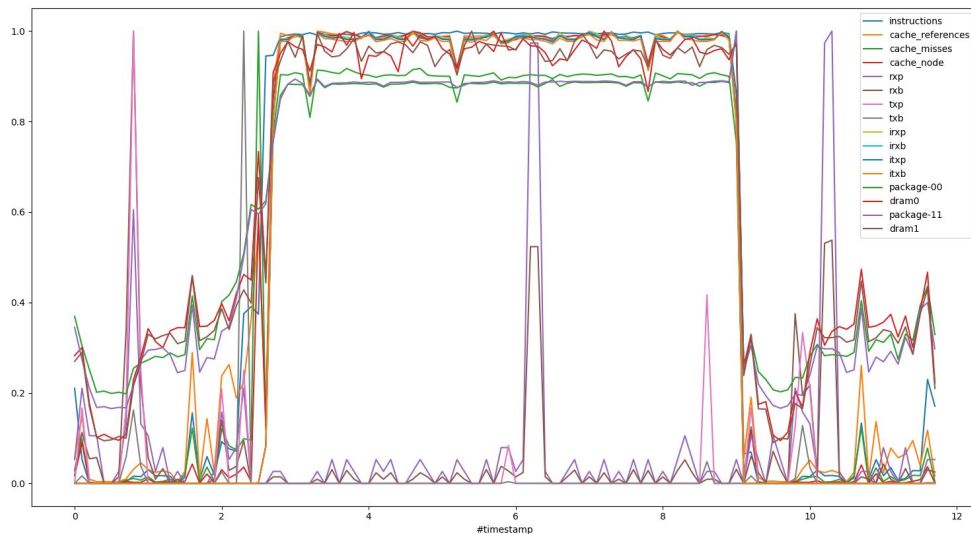
Comparaison regret :



# Contexte : expériences réalisées / objectif

8 applications exécutées sur serveurs

Objectif : choisir fréquence optimale  
du processeur sans connaître le nom  
de l'application



# Présentation données avec noms :

Contexte :

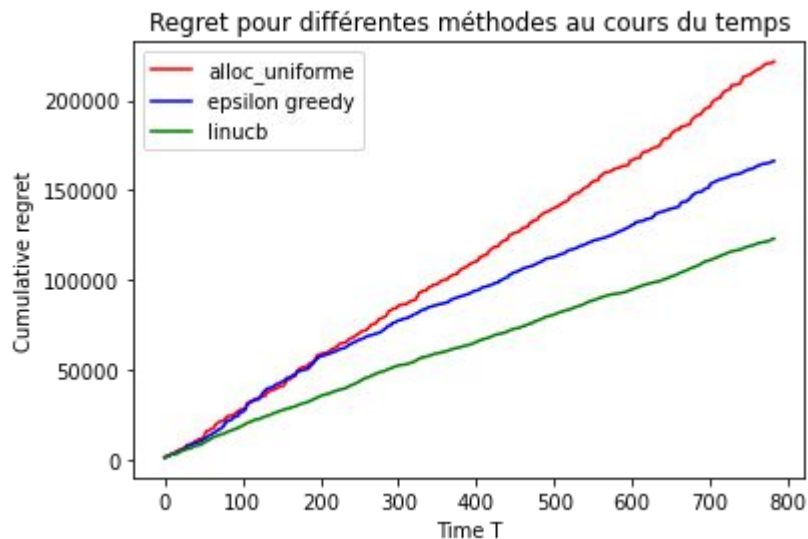
	bt	cg	ep	ft	is	lu	mg	sp	duration
1	0	0	0	0	0	1	0	0	28.63
2	0	0	0	0	1	0	0	0	15.86
3	0	0	0	1	0	0	0	0	11.73
4	0	0	0	0	1	0	0	0	15.86
5	0	0	0	0	0	0	0	1	27.80

Récompenses :

	ref_energy	ref_energy.1	ref_energy.2	ref_energy.3	ref_energy.4	ref_energy.5
1	-4638.06	-4382.08	-4131.74	-4039.200	-3973.810	-3859.20
2	-2394.86	-2330.20	-2337.40	-2277.600	-2319.840	-2371.07
3	-1865.07	-1773.90	-1754.62	-1674.860	-1648.115	-1579.30
4	-2339.35	-2360.56	-2342.05	-2248.775	-2217.605	-2236.80
5	-5309.80	-5177.16	-5075.00	-4908.980	-4811.100	-4788.72

# Mise en oeuvre sur données avec noms :

Tableau récapitulatif :



	Frequence optimale	Consommation optimale	Gain moyen
bt	2.0	3331.671713	2897.604678
cg	2.1	1026.840410	204.548272
ep	2.0	190.948250	843.054201
ft	2.0	1074.902362	753.132021
is	2.0	1897.548867	764.851028
lu	2.3	1907.508985	2265.119612
mg	2.4	5852.418687	181.052242
sp	2.2	3411.396701	927.945845



# Présentation données détaillées :

Données détaillées pour une expérience :

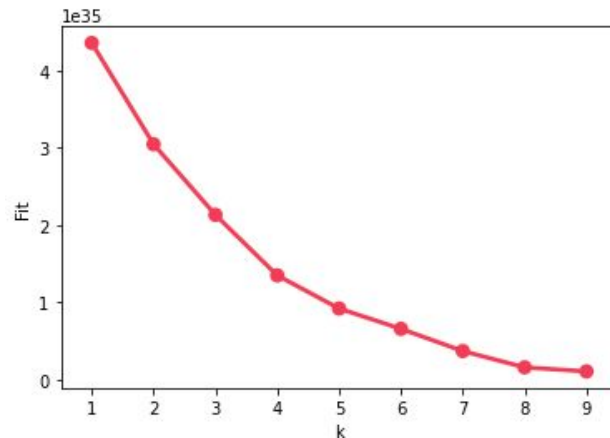
instructions	cache_references	cache_misses	cache_node	rxp	rxb	txp	txb	irxp	irxb	itxp	
7.126050e+09	2.063024e+07	1.068034e+07	6.790390e+06	5.897119	477.176955	0.506173	264.473251	18262.230453	8.380612e+06	18262.152263	8.38059
7.120380e+09	2.039575e+07	1.062264e+07	6.761612e+06	5.599174	429.239669	0.148760	119.648760	18166.024793	8.336804e+06	18166.004132	8.33680
7.100371e+09	2.039909e+07	1.061428e+07	6.753175e+06	5.565574	426.286885	0.184426	122.409836	18175.151639	8.341428e+06	18174.954918	8.34105
7.141135e+09	2.043908e+07	1.062542e+07	6.751956e+06	5.698347	448.190083	0.322314	138.764463	18165.913223	8.336802e+06	18165.979339	8.33681
4.928587e+09	3.244595e+07	5.439335e+06	4.103259e+06	1.838235	259.367647	1.382353	569.926471	76533.529412	3.552697e+07	76524.588235	3.55222
...	...	...	...	...	...	...	...	...	...	...	...
4.708899e+09	7.206684e+07	1.421589e+07	1.065412e+07	1.811594	141.192029	0.434783	233.840580	6306.420290	3.035007e+06	6306.108696	3.03478
3.635065e+09	2.254729e+07	1.014367e+07	6.673391e+06	1.364000	786.856000	0.636000	176.648000	35175.864000	1.811652e+07	34690.348000	1.78653
3.709421e+09	2.253363e+07	1.013092e+07	6.660942e+06	1.365462	778.650602	0.598394	264.718876	34950.465863	1.799941e+07	34954.321285	1.80012
3.632964e+09	2.242635e+07	1.032884e+07	6.890166e+06	1.318725	759.254980	0.581673	261.529880	34477.760956	1.775560e+07	34738.932271	1.78908
3.683452e+09	2.248119e+07	1.034567e+07	6.844611e+06	1.286853	179.031873	0.490040	257.422311	34510.035857	1.777199e+07	34728.621514	1.78851

# Clustering :

Données pour clustering :

	hostname	fullname	nproc	duration	startTime	endTime	fmin	fmax
0	graoully-1.nancy.grid5000.fr	bt-C-64	64	40.57	1637601498	1637601609	1200000	1200000
1	graoully-1.nancy.grid5000.fr	bt-C-64	64	37.76	1637601626	1637601734	1200000	1300000
2	graoully-1.nancy.grid5000.fr	bt-C-64	64	35.64	1637601750	1637601855	1200000	1400000
3	graoully-1.nancy.grid5000.fr	bt-C-64	64	33.67	1637601870	1637601974	1200000	1500000
4	graoully-1.nancy.grid5000.fr	bt-C-64	64	32.12	1637601990	1637602092	1200000	1600000

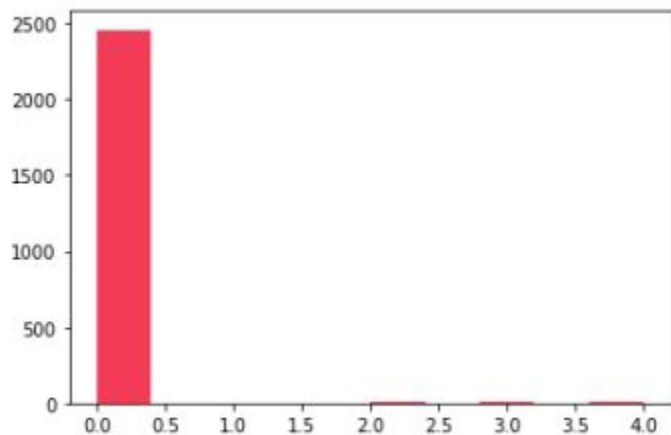
Méthode du coude :



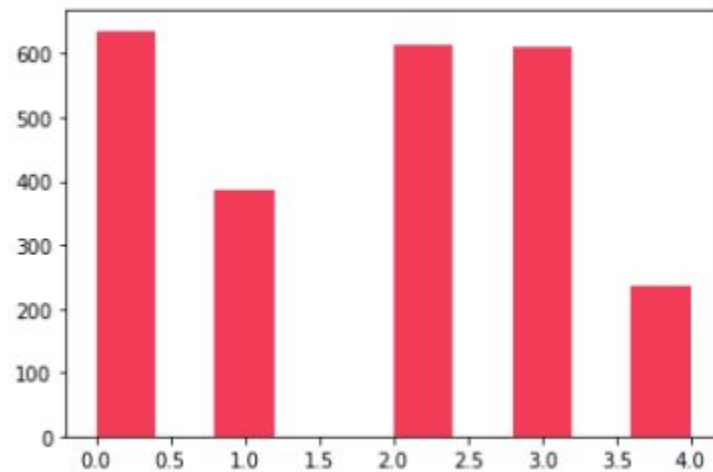


# Comparaison moyennes / médianes :

Répartition avec moyennes :



Répartition avec médianes :



## Recherche meilleure combinaison :

[illegible]

# Résultats dans le cas optimal :

	Frequence optimale	Consommation optimale	Gain moyen
0	2.3	14565.016882	9556.939688
1	1.9	17810.923460	3715.426509
2	2.3	9945.570908	8623.261680
3	2.3	2089.162050	4583.371741
4	2.0	10993.019935	5118.458999