

Durée de vie d'un élément radioactif et ajustement linéaire et exponentiel

En 1902, Rutherford et Soddy énoncent la loi de décroissance radioactive qui permet d'écrire que la variable aléatoire T qui, à un noyau donné, associe la durée de vie de celui-ci, admet une fonction de répartition F solution de l'équation différentielle

$$F'(t) + \lambda F(t) = \lambda$$

où λ est appelée constante de désintégration, interprétée comme l'inverse de la durée de vie moyenne d'un noyau.

On considère, à l'instant initial $t = 0$, 10000 noyaux radioactifs. La constante λ vaut ici 0,60.

Simulation de la désintégration :

À chaque étape, pour chaque atome, on simule un réel choisi de manière aléatoire entre 0 et 1, si ce nombre est inférieur à λ , l'atome meurt, sinon il garde son état précédent.

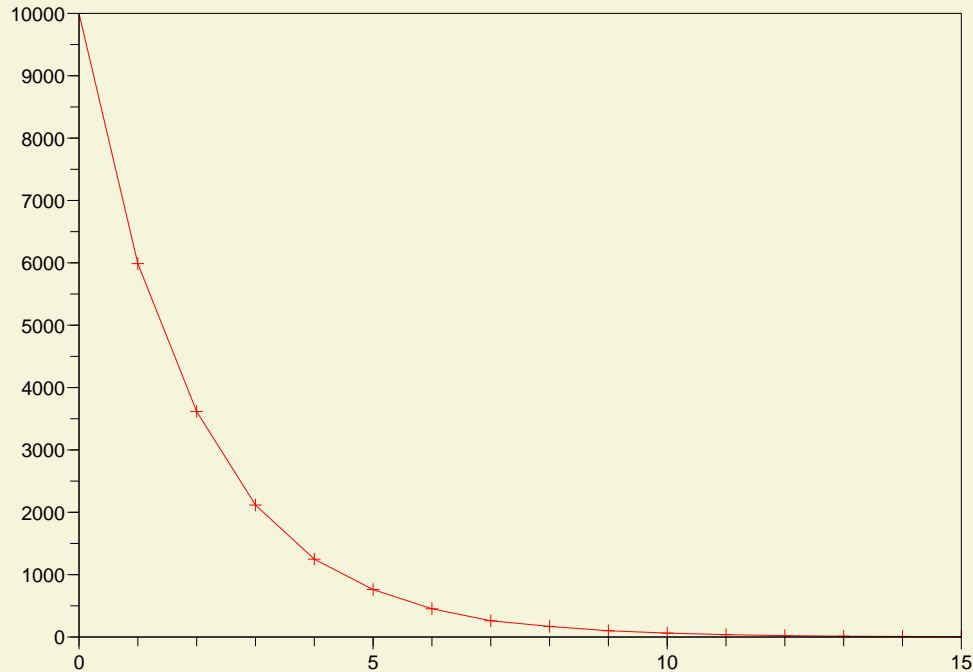
On observe alors l'évolution du nombre $N(t)$ d'atomes encore vivants en fonction du temps t et on trace cette évolution grâce au nuage de points $M(t; N(t)), t$ prenant des valeurs entières.

```
--> lambda=0.6;
--> N=15;
--> y=zeros(10000,N);
--> for i=1:10000
-->     j=1;t=rand();
-->     while t<lambda do
-->         y(i,j)=1;j=j+1;t=rand();
```



```
--> end;  
--> end;  
--> for j=1:N, somme(j)=sum(y(:,j));end  
--> temps=[0:N];  
--> t0=[10000];  
--> sommes=[t0,somme'];  
--> x=0:N;  
--> plot(temps',sommes,"r+-")
```





Le graphique nous laisse penser que l'on peut ajuster le nuage de points par une courbe exponentielle. Pour cela, nous allons dans un premier temps écrire un algorithme qui permet de déterminer une équation de la droite de régression de $\ln(\text{sommes})$ en x par la méthode des moindres carrés.

Je me suis inspiré très fortement du T-P se trouvant dans le document de Claude Gomez que l'on peut lire ici : http://www.scilab.org/lycee/index_lycee.php



Déterminons la taille du vecteur sommes.

```
--> n=length(sommes);
```

```
--> x=[0:n-1];
```

Effectuons le changement de variable qui va permettre un ajustement linéaire.

```
--> y=[log(sommes)];
```

Calculons la moyenne arithmétique des variables x et y .

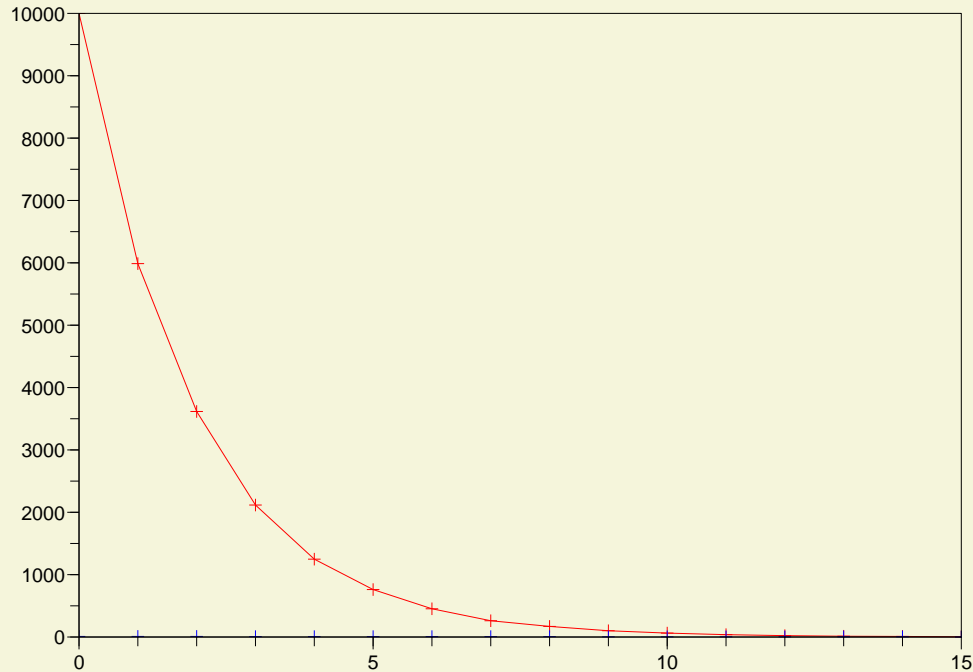
```
--> xbar=mean(x);
```

```
--> ybar=mean(y);
```

Traçons le nuage de points

```
--> plot(x,y,"+b")
```





Demandons 20 chiffres significatifs dans les calculs.

--> `format(20)`

Nous allons utiliser la méthode des moindres carrés en cherchant, parmi toutes les droites passant par le point moyen, $G(\bar{x}; \bar{y})$, la droite qui minimise la somme $\sum_{i=1}^n (y_i - (ax_i + b))^2$. Nous fixons arbitrairement la



somme des carrés égale à 20000 et le coefficient directeur `coeff` égal à 0 pour la suite du test.

```
--> sommedescarres=20000;
```

```
--> coeff=0;
```

```
Warning :redefining function: coeff
```

Pour gagner du temps on acceptera l'idée que le coefficient directeur de la droite d'ajustement est un réel compris entre -0.65 et -0.4. Dans le cas contraire on pourra appliquer le test qui suit en *balayant* beaucoup plus large.

```
--> for k=-0.65:0.00001:-0.4
```

```
--> d=y-(k*(x-xbar)+ybar);
```

```
--> E=sum(d.*d);
```

```
--> if E<sommedescarres then sommedescarres=E;coeff=k;end
```

```
--> end
```

```
--> Y=k*(x-xbar)+ybar;
```

```
--> b=ybar-xbar*coeff;
```

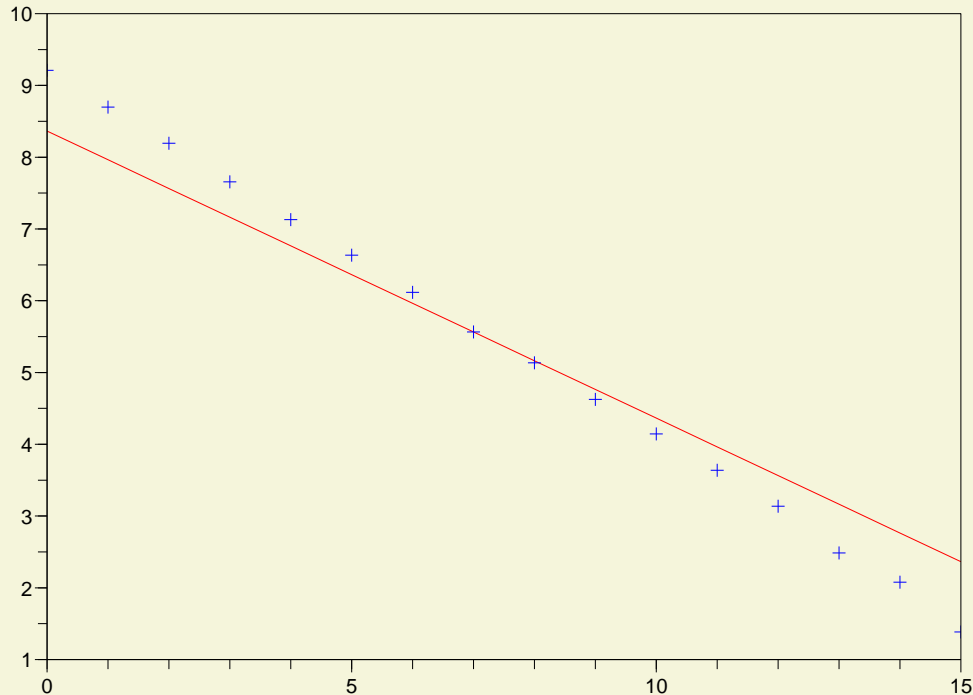
```
--> clf()
```

On peut tracer ce que l'on a trouvé comme ajustement affine.

```
--> plot(x,Y,"r")
```

```
--> plot(x,y,"+b")
```





Comparons le coefficient directeur que l'on a trouvé avec le coefficient directeur $a = \frac{\text{cov}(x;y)}{\text{var}(x)}$ que l'on doit obtenir théoriquement.

```
--> cov=sum(x.*y)/n-xbar*ybar;
--> varx=sum(x.*x)/n-xbar*xbar;
```



```
--> coeff1=cov/varx
coeff1 =
-0.51278269336269200
--> coeff
ans =
-0.51278
```

Nous pouvons maintenant tracer l'ajustement exponentiel correspondant.

```
--> ajust=exp(b)*exp(coeff*x);
--> clf()
--> plot(x,ajust,"+r")
--> plot(temps',sommes,"-b")
```



