



Polynômes de Taylor-Maclaurin

© Pierre Lantagne

Enseignant retraité du Collège de Maisonneuve

Ce document est une révision de celui produit en 2007. L'objectif principal de cette feuille Maple est d'initier l'élève aux polynômes de Taylor-Maclaurin et à leur utilisation dans l'obtention d'approximations.

Bonne lecture à tous !

* Ce document Maple est exécutable avec la version 2020.1

Initialisation

```
> restart;  
> with(plots,display,setoptions):  
setoptions(axesfont=[TIMES,ROMAN,8],size=[300,300]);
```

Polynômes de Taylor-Maclaurin

Si une fonction f est dérivable une infinité de fois en $x = a$, on peut lui associer la série de Taylor-Maclaurin centrée en a :

$$T(x) = \sum_{k=0}^{\infty} f^{(k)}(a) \frac{(x-a)^k}{k!}$$

Quand $a = 0$, on parle alors de séries de Maclaurin.

En considérant les sommes partielles des termes de la série de Taylor-Maclaurin, on considère, dans ce cas, des polynômes de Taylor-Maclaurin $P_n(x)$.

$$P_n(x) = \sum_{k=0}^n \frac{f^{(k)}(a) (x-a)^k}{k!}$$

Les polynômes de Taylor-Maclaurin ont pour objet l'approximation d'une fonction dans un voisinage autour de a . Ces polynômes généralisent l'approximation de la fonction par la tangente.

Les polynômes de Taylor-Maclaurin ressemblent beaucoup à la fonction associée, la ressemblance augmentant avec le degré. Il semble alors naturel d'accepter que $P_{\infty}(x) = T(x) = f(x)$. Autrement dit, que la série de Taylor-Maclaurin converge vers la fonction utilisée pour développer la série. Malheureusement, ce n'est pas toujours le cas. Pour qu'il y ait convergence, il y a certaines conditions à satisfaire.

En fait, pour tout polynôme de Taylor-Maclaurin d'ordre n associé à une fonction f sur un intervalle centré en $x = a$, nous avons

$$f(x) = P_n(x) + R_n(x)$$

où $R_n(x)$ est appelé le reste d'ordre n de la fonction f en $x = a$. Si nous pouvons montrer que $\lim_{n \rightarrow \infty} R_n(x) = 0$, alors $T(x) = P_{\infty}(x)$ représentera la fonction f sur l'intervalle centré en $x = a$.

Le reste $R_n(x)$ est habituellement formulé sous la forme de *reste de Lagrange* ou sous la forme du *reste intégral*.

Cet aspect du reste est abordé dans un cours de calcul avancé. Pour le moment, voici quelques exemples illustrés de polynômes de Taylor-Maclaurin.

La macro-commande

La macro-commande `taylor` de la bibliothèque de base Maple nous donne le développement limité de Taylor-Maclaurin selon l'ordre k spécifié et ajoute un dernier terme de la forme $O(x - a)^k$. Ce dernier terme symbolise la somme (infinie) de tous les termes restants, soit donc en quelque sorte le reste. Cela nous indique ainsi que les termes restant sont d'un degré égal ou supérieure à k . Alors, pour obtenir avec Maple un polynôme de Taylor-Maclaurin d'ordre n , soit jusqu'à la dérivée d'ordre n pour avoir le $(n+1)$ ème termes du polynôme, il faut spécifier $k = n + 1$.

Quelques exemples

Soit la fonction f définie par $f(x) = e^x$. Développons quelques polynômes en $x = 0$.

```
> f:=x->exp(x);
```

$$f := x \mapsto e^x \quad (3.1.1)$$

Obtenons le polynôme de Taylor-Maclaurin d'ordre 0, soit P_0 .

```
> P[0]:=taylor(f(x),x=0,1):
'P[0]':P[0];
```

$$P_0 = 1 + O(x) \quad (3.1.2)$$

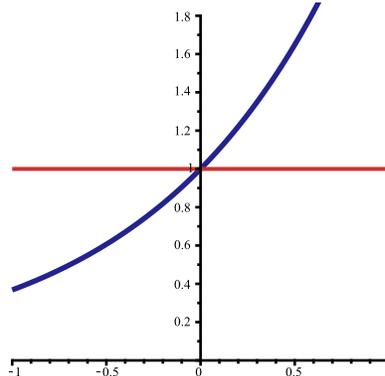
Il nous faudra donc tronquer le reste afin d'avoir le polynôme de Taylor-Maclaurin associé. Utilisons la macro-commande `convert`.

```
> 'P[0]':=convert(P[0],polynom);
```

$$P_0 = 1 \quad (3.1.3)$$

Illustrons la qualité de l'approximation de la fonction f au voisinage de $x = 0$ par le polynôme $P_0(x)$.

```
> Fonction:=plot([x,exp(x),x=-1..1],color=navy,thickness=2):
P0:=plot([x,convert(P[0],polynom),x=-1..1],color=orange):
> display(Fonction,P0,view=[-1..1,0..1.8]);
```



Graphiquement, il est clair que la qualité de l'approximation de la fonction exponentielle au voisinage de

$x = 0$ par les valeurs du polynôme P_0 dans ce voisinage laisse drôlement à désirer. Calculons tout de même l'écart entre la fonction et la valeur de P_0 en $x = 0.2$.

```
> f(0.2)-eval(P[0],x=0.2);
```

$$0.2214027580 \quad (3.1.4)$$

Obtenons le polynôme de Taylor-Maclaurin d'ordre 1 pour voir si la qualité de l'approximation va s'améliorer.

```
> P[1]:=taylor(f(x),x=0,2):
```

```
'P[1]':=P[1];
```

$$P_1 = 1 + x + O(x^2) \quad (3.1.5)$$

Il nous faudra encore tronquer le reste afin d'avoir le polynôme de Taylor-Maclaurin associé.

```
> 'P[1]':=convert(P[1],polynom);
```

$$P_1 = 1 + x \quad (3.1.6)$$

Illustrons la qualité de l'approximation de la fonction f au voisinage de $x = 0$ par le polynôme $P_1(x)$.

```
> Fonction:=plot([x,exp(x),x=-1..1],color=navy,thickness=2):
```

```
P1:=plot([x,convert(P[1],polynom),x=-1..1],color=orange):
```

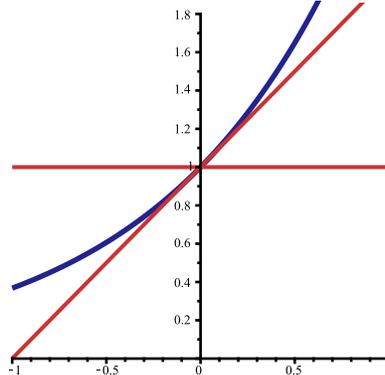
```
'P[0]':=convert(P[0],polynom);
```

```
'P[1]':=convert(P[1],polynom);
```

$$P_0 = 1$$

$$P_1 = 1 + x \quad (3.1.7)$$

```
> display(Fonction,P0,P1,view=[-1..1,0..1.8]);
```



L'approximation en $x = 0.2$ de la fonction exponentielle au voisinage de $x = 0$ par la valeur du polynôme P_1 semble qu'elle sera meilleur avec la tangente. Calculons l'écart en $x = 0.2$ entre la fonction et la valeur de P_1 .

```
> f(0.2)-eval(P[1],x=0.2);
```

$$0.0214027580 \quad (3.1.8)$$

En effet, il y a eu une amélioration dans la qualité de l'approximation.

Obtenons P_2 pour voir si la valeur de ce polynôme avec $x=0.2$ va encore améliorer l'approximation.

```
> P[2]:=taylor(f(x),x=0,3):
```

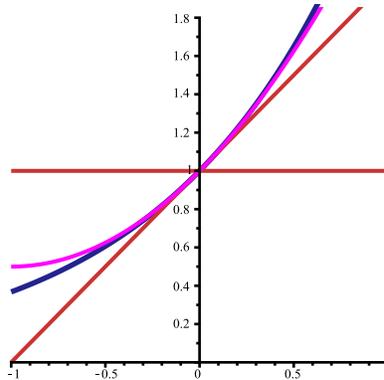
```
'P[0]':=convert(P[0],polynom);
```

```
'P[1]'=convert(P[1],polynom);
'P[2]'=convert(P[2],polynom);
P2:=plot([x,convert(P[2],polynom),x=-1..1],color=magenta);
display(Fonction,P0,P1,P2,view=[-1..1,0..1.8]);
```

$$P_0 = 1$$

$$P_1 = 1 + x$$

$$P_2 = 1 + x + \frac{1}{2}x^2$$



Au voisinage de $x = 0$, la tangente a amélioré l'approximation mais la courbure de P_2 améliore davantage l'approximation en ce voisinage. C'est beaucoup mieux. Calculons de nouveau l'écart en $x = 0.2$ mais avec P_2 .

```
> f(0.2)-eval(P[2],x=0.2);
0.0014027580 (3.1.9)
```

Alons-y avec P_3 .

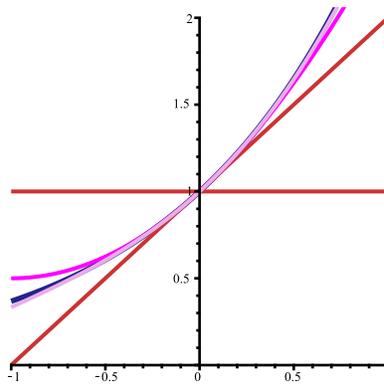
```
> P[3]:=taylor(f(x),x=0,4):
'P[0]'=convert(P[0],polynom);
'P[1]'=convert(P[1],polynom);
'P[2]'=convert(P[2],polynom);
'P[3]'=convert(P[3],polynom);
P3:=plot([x,convert(P[3],polynom),x=-1..1],color=plum):
display(Fonction,P0,P1,P2,P3,view=[-1..1,0..2]);
```

$$P_0 = 1$$

$$P_1 = 1 + x$$

$$P_2 = 1 + x + \frac{1}{2}x^2$$

$$P_3 = 1 + x + \frac{1}{2}x^2 + \frac{1}{6}x^3$$



Et ça s'améliore encore plus. Calculons de nouveau le calcul de l'écart.

```
> f(0.2)-eval(P[3],x=0.2);
```

0.0000694250 **(3.1.10)**

Ce qui confirme ce que l'on observe graphiquement: la qualité de l'approximation est encore meilleur avec P_3 .

Au-delà de P_3 , il devient difficile d'illustrer graphiquement l'augmentation de la précision de l'approximation. Voici un tableau « illustrant numériquement » l'augmentation de la qualité de l'approximation au voisinage de $x=0$ lorsque le « degré » du polynôme de Taylor-Maclaurin augmente.

```
> f:=x->exp(x):
x[0]:=0.2:
Digits:=40:
> printf(`\n          Écart en x=%1.1f entre la fonction et
l'approximation \n
          avec un polynôme de Taylor-Maclaurin d'ordre
k\n`,x[0]);
printf(`\n          |   Ordre k   |   f(%1.1f)-P[k]
(%1.1f)   |\n`,x[0],x[0]);
printf(`
=====
====\n`);
for k in [$0..10]
do
printf(`          |   %5d   |   %15.20f   |   \n`,
k,
abs(f(x[0])-eval(convert(taylor(f(x),x=0,k),polynom),x=x[0])));
od;
k:='k':
```

Écart en x=0.2 entre la fonction et l'approximation

avec un polynôme de Taylor-Maclaurin d'ordre k

Ordre k	f(0.2)-P[k](0.2)
0	1.22140275816016983392
1	0.22140275816016983392
2	0.02140275816016983392
3	0.00140275816016983392

4	0.00006942482683650059
5	0.00000275816016983392
6	0.00000009149350316725
7	0.0000000260461427837
8	0.0000000006493173868
9	0.0000000000143967519
10	0.0000000000002874045

Prenons un autre exemple. Soit la fonction f définie par $f(x) = \ln(x)$ et donnons des approximations de $\ln(1,1)$ avec des polynômes de Taylor-Maclaurin au voisinage de $x = 1$.

```
> f:=x->ln(x):
  x[0]:=1.1:
> printf(`\n
          Écart en x=%1.1f entre la fonction et
l'approximation \n
          par un polynôme de Taylor-Maclaurin d'ordre k\n`,
x[0]);
printf(`\n
      |      Ordre k      |      f(%1.1f)-P[k]
(%1.1f)  |\n`,x[0],x[0]);
printf(`
=====
===\n`);
for k in [$0..15]
do
printf(`
      |      %5d      |      %15.20f      | \n`,
k,
abs(f(x[0])-eval(convert(taylor(f(x),x=1,k),polynom),x=x[0])));
od;
k:='k':
```

```
Écart en x=1.1 entre la fonction et l'approximation
par un polynôme de Taylor-Maclaurin d'ordre k
```

Ordre k	f(1.1)-P[k](1.1)
0	0.09531017980432486004
1	0.09531017980432486004
2	0.00468982019567513996
3	0.00031017980432486004
4	0.00002315352900847329
5	0.00000184647099152671
6	0.00000015352900847329
7	0.00000001313765819338
8	0.00000000114805609234
9	0.00000000010194390766
10	0.00000000000916720345
11	0.00000000000083279655
12	0.00000000000007629436
13	0.00000000000000703898
14	0.00000000000000065333
15	0.00000000000000006095

Illustration graphique des polynômes de Taylor-Maclaurin

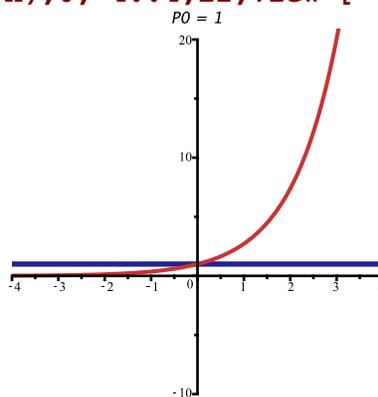
La procédure d'animation suivante illustre la « convergence » des polynômes de Taylor-Maclaurin dans un voisinage centré en $x = a$.

```
> Taylor_animation := proc(f::anything,centre::realcons,
voisinage::range, n::posint)
local a, b, k, l, t, q, p, Pn;
  l:=(rhs(voisinage)-lhs(voisinage))/2;
  a := centre - l;
  b := centre + l;
  for k to n+1 do
    t := taylor(f, x = centre, k);
    Pn := unapply(convert(t, polynom), x);
    q[k] := plot([x,Pn(x), x = a .. b] ,
      numpoints = 1000, color = navy,thickness=2,
      title=`P`||(k-1)||` = `||(convert(sort(convert(taylor(f, x =
centre, k),polynom),x),string)), titlefont=[HELVETICA, 10], args[5 ..
nargs])
    od;
    q := plots[display]([seq(q[k], k = 1 .. n+1)],insequence = true);
    p := plot([x,f, x = voisinage], color = orange,
      args[5 .. nargs]);

    plots[display]([q, p])
  end;
```

Illustrons graphiquement les polynômes de Taylor-Maclaurin centrés en $x = 0$ jusqu'à l'ordre 12 pour la fonction f définie par $f(x) = e^x$.

```
> Taylor_animation(exp(x),0,-4..4,12,view=[-4..4,-10..20]);
```



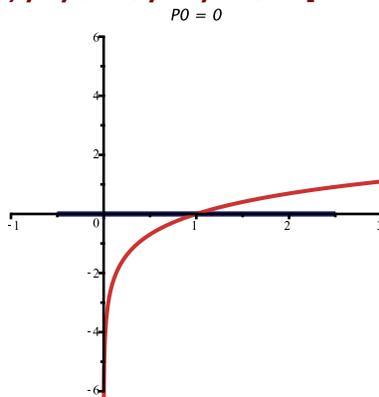
Il est clair que, pour x près de 0, l'approximation de la valeur de e^x s'améliore lorsque le « degré » des polynômes de Taylor-Maclaurin augmente. De plus, les polynômes de Taylor-Maclaurin développés en $x = 0$ semblent converger vers e^x pour n'importe quelle valeur de x , pas seulement pour des valeurs dans un voisinage immédiat de $x = 0$. Il semble donc que

$$P_{\infty} = e^x$$

mais, il faudrait le prouver.

Comme deuxième exemple, illustrons graphiquement les polynômes de Taylor-Maclaurin centrés en $x = 1$ jusqu'à l'ordre 14 pour la fonction f définie par $f(x) = \ln(x)$.

```
> Taylor_animation(ln(x),1,0..3,14,view=[-1..3,-6..6]);
```



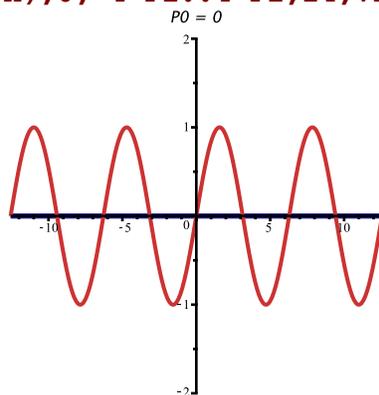
Dans cet exemple aussi, il est clair que, pour x près de 1, l'approximation de la valeur de $\ln(x)$ s'améliore lorsque le « degré » des polynômes de Taylor-Maclaurin augmente. Par contre, les polynômes de Taylor-Maclaurin développés en $x = 1$ semblent converger vers $\ln(x)$ seulement pour des x dans un voisinage immédiat de $x = 1$. Il semble que

$$P_{\infty} = \ln(x) \text{ pour } x \in]0,2],$$

mais il faudrait le prouver.

Comme dernier exemple, illustrons graphiquement les polynômes de Taylor-Maclaurin centrés en $x = 0$ jusqu'à l'ordre 24 pour la fonction f définie par $f(x) = \sin(x)$.

```
> Taylor_animation(sin(x),0,-4*Pi..4*Pi,24,view=[-4*Pi..4*Pi,-2..2]);
```



Ici aussi, les polynômes de Taylor-Maclaurin développés en $x = 0$ semblent converger vers $\sin(x)$ pour n'importe quelle valeur de x , pas seulement pour des x dans un voisinage immédiat de $x = 0$. Il semble donc que

$$P_{\infty} = \sin(x)$$

mais, il faudrait tout de même en faire la preuve.

Que remarquez-vous dans le déroulement de l'animation précédente?

```
> n:=15:
  x[0]:=0:
  P:='P':
  for k from 0 to n do
```

```
P[k]=convert(taylor(sin(x),x=x[0],k),polynom)
od;
```

$$P_0 = 0$$

$$P_1 = 0$$

$$P_2 = x$$

$$P_3 = x$$

$$P_4 = x - \frac{1}{6} x^3$$

$$P_5 = x - \frac{1}{6} x^3$$

$$P_6 = x - \frac{1}{6} x^3 + \frac{1}{120} x^5$$

$$P_7 = x - \frac{1}{6} x^3 + \frac{1}{120} x^5$$

$$P_8 = x - \frac{1}{6} x^3 + \frac{1}{120} x^5 - \frac{1}{5040} x^7$$

$$P_9 = x - \frac{1}{6} x^3 + \frac{1}{120} x^5 - \frac{1}{5040} x^7$$

$$P_{10} = x - \frac{1}{6} x^3 + \frac{1}{120} x^5 - \frac{1}{5040} x^7 + \frac{1}{362880} x^9$$

$$P_{11} = x - \frac{1}{6} x^3 + \frac{1}{120} x^5 - \frac{1}{5040} x^7 + \frac{1}{362880} x^9$$

$$P_{12} = x - \frac{1}{6} x^3 + \frac{1}{120} x^5 - \frac{1}{5040} x^7 + \frac{1}{362880} x^9 - \frac{1}{39916800} x^{11}$$

$$P_{13} = x - \frac{1}{6} x^3 + \frac{1}{120} x^5 - \frac{1}{5040} x^7 + \frac{1}{362880} x^9 - \frac{1}{39916800} x^{11}$$

$$P_{14} = x - \frac{1}{6} x^3 + \frac{1}{120} x^5 - \frac{1}{5040} x^7 + \frac{1}{362880} x^9 - \frac{1}{39916800} x^{11} + \frac{1}{6227020800} x^{13}$$

$$P_{15} = x - \frac{1}{6} x^3 + \frac{1}{120} x^5 - \frac{1}{5040} x^7 + \frac{1}{362880} x^9 - \frac{1}{39916800} x^{11} + \frac{1}{6227020800} x^{13}$$

(4.1)