L2ande - TP maximisation d'une fonction sous contraintes

28 avril 2021

Partie 1 : maximisation d'une fonction ℝ→ℝ sur un intervalle [a,b]

Habituellement on forme le **tableau de variation de la fonction** (qui est une allure du graphe très épuré avec explicités les points de changement de variation).

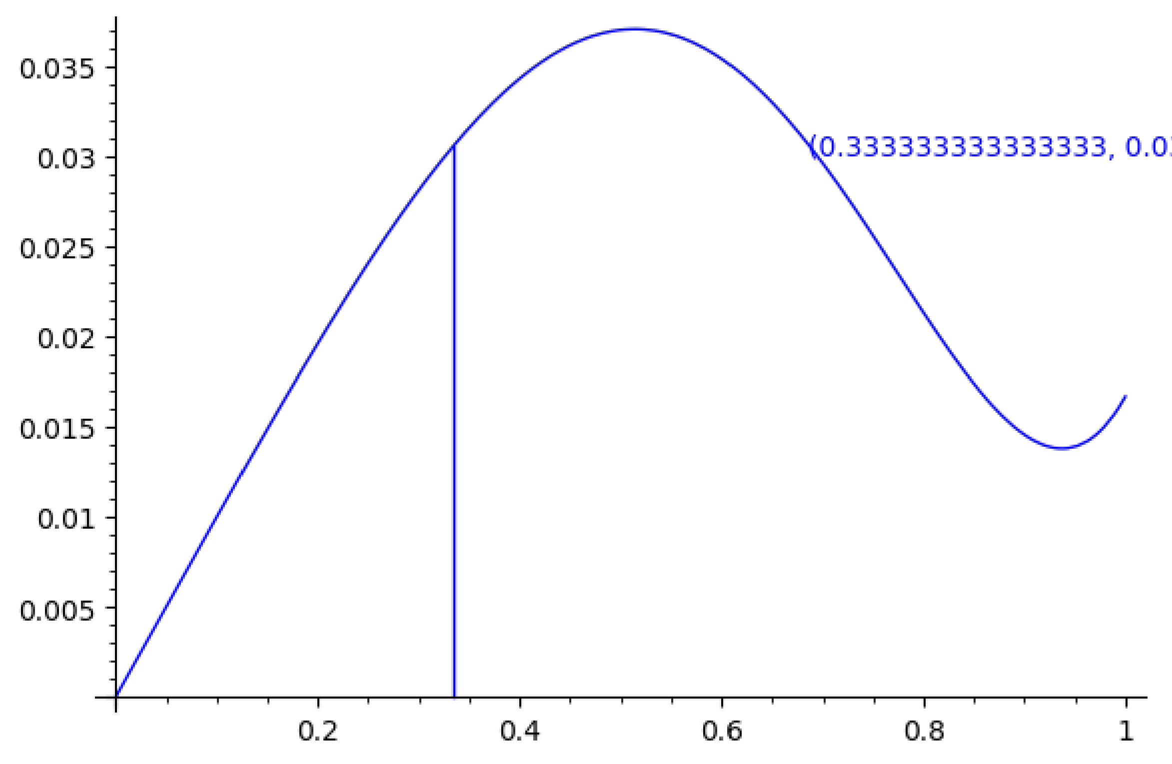
On accède à la représentation graphique du tableau de variation en dessinant le graphe de la fonction (représentation approchée et non démontrée a priori : on ne connaît pas la certification de l'algorithme Sagemath qui dessine le graphe !). Ne manque qu'un traqueur des points sur le graphe pour avoir une approximation numérique à la précision souhaitée des points d'intérêt.

✍ Exemple : $g : [0,1] \to \mathbb{R},\ x \mapsto x^6/6 - x^4/4 + x/10$. Donner une valeur à 10^{-3} près du x où g atteint son maximum en vous aidant du dessin mobile ci-dessous.

In [1]:
```
a,b,ymin=0.0,1.0,0
@interact
def go(g=1/6*x^6-1/4*x^4+1/10*x,x0=slider(a,b,(b-a)/100,default = a*2/3+b*1/3)):#a=input_box(0, width=8),b=input_box(1, width=8),x0=slider(a,b,(b-a)/100.n(digits=2),default = a*2/3+b*1/3)
    p1=plot(g,(x,a,b))
    p2=line([(x0,ymin),(x0,g(x0))])
    show(p1+p2+text(str((x0,g(x0))),(b,g(x0))))
```

g 1/6*x^6 - 1/4*x^4 + 1/10*x

x0 0.33



Les points d'intérêt du tableau de variation sont les extrémités a, b et les $x \in [a,b]$ où la dérivée de g s'annule.

La dérivée s'obtient par l'instruction `g.derivative()` ou mieux `diff(g(x),x)` (on précise la variable par rapport à laquelle on dérive la fonction).

Les x où une fonction h s'annule s'obtiennent formellement par l'instruction `solve(h(x)==0,x)` (mais il n'y a pas toujours de réponse explicite), numériquement par l'instruction `solve(h(x)==0,x,to_poly_solve='force')` (notamment)

✍ Comparer les points d'annulation obtenus pour $x^5 - x^3 + 1/10$ avec le graphe de la primitive g plus bas.

In [2]:
```
@interact
def _(h=selector([x^5-x^3+1/10,2*x^5-x^3+1,cos(x)^2-1,(cos(x)^2-1)*(x^5-x^3+1/10)],buttons=True),explicit=checkbox(False),polysolve=selector(['force',True,False])):
      print(solve(h==0,x,explicit_solutions=explicit,to_poly_solve=polysolve))
#solve?
```

h x^5 - x^3 + 1/10 2*x^5 - x^3 + 1 cos(x)^2 - 1 1/10*(10*x^5 - 10*x…

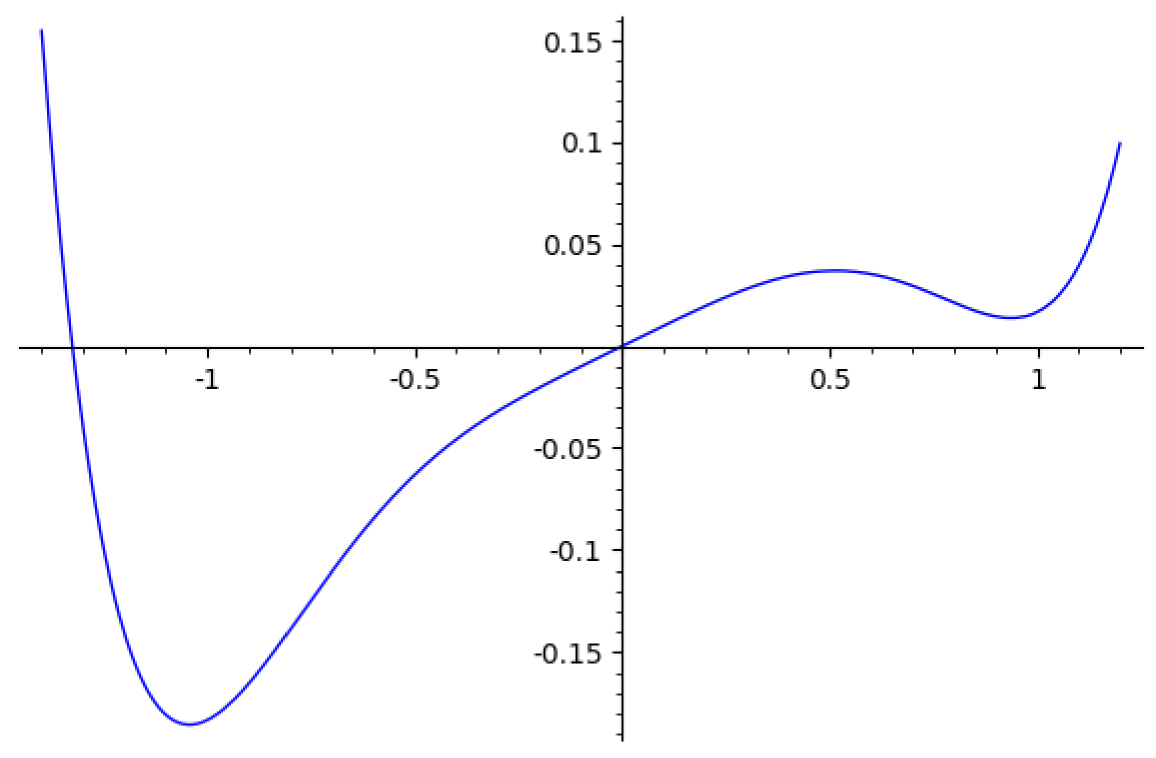
☐ explicit

polysolve force

```
[x == -1.043122035360069, x == 0.5141808475141808, x == 0.9373188405797102, x == (-0.2041887771764046 - 0.3965088043695821*I), x == (-0.2041887771764046 + 0.3965088043695821*I)]
```

In [3]:
```
g(x)=1/6*x^6-1/4*x^4+1/10*x
plot(g,-1.4,1.2)
```

Out[3]:



L'étude du tableau de variation se généralise mal aux fonctions à plusieurs variable alors que la méthode des multiplicateurs de Lagrange s'applique en toute dimension.

Maximisation de g sous la contrainte $x \geq a$ avec les multiplicateurs de Lagrange:

Si $g : [a, +\infty[\to \mathbb{R}$ est maximal en x alors nécessairement le gradiant de g en x (c'est à dire la dérivée de g en x) est nul ou bien $x = a$ et le gradiant pointe vers l'extérieur de $[a, +\infty[$ (c'est à dire est négatif).

On réécrit la contrainte sous la forme $a - x \leq 0$ puis on forme l'expression $dxL(x,k) = \frac{\partial}{\partial x}(g(x) - k \times (a - x))$. Les points d'intérêt sont les (x,k) annulant en même temps dxL et $k \times (a - x)$. On retient ceux pour lesquels $k \geq 0$. Si on en trouve plusieurs, on compare la valeur de g en les x trouvés (pourvu qu'ils soient en nombre fini). Si on sait que g atteint son maximum sur le domaine $[a, +\infty[$ alors les x trouvés pour lesquels g est maximal sont des maximums de g.

Exemple : g comme ci-dessus et $a = 0$.

In [4]:
```
a=0
dxL(x,k)=diff(g(x)-k*(a-x),x)
s=solve([dxL(x,k)==0,k*(a-x)==0],[x,k])
for i in range(len(s)):
    print(i+1,s[i])
```

```
1 [x == 0.5141808475141808, k == 0.0]
2 [x == (-0.2041887771764046 + 0.3965088043695821*I), k == 0.0]
3 [x == (-0.2041887771764046 - 0.3965088043695821*I), k == 0.0]
4 [x == 0.9373188405797102, k == 0.0]
5 [x == -1.043122035360069, k == 0.0]
6 [x == 0, k == (-1/10)]
```

Les deuxième et troisième solutions sont des nombres complexes, hors sujet.
L'avant dernière solution ne satisfait pas la contrainte $x \geq a$.
La dernière solution a un $k < 0$ donc est disqualifiée pour la recherche du maximum.
Reste la première et la quatrième.

In [5]:
```
for i in [0,3]: # l'indentation commence à 0 dans Python !
    x=s[i][0].rhs()
    print(x,g(x))
```

```
0.5141808475141808 0.03702358492805572
0.9373188405797102 0.013786623659479413
```

On élimine ainsi le deuxième candidat. On n'en déduit pas que g atteint son maximum en le premier candidat car de fait g n'est pas majorée sur l'intervalle $[0, +\infty[$.

✍ Refaire les calculs et la discussion pour les contraintes simultanée $(x \geq 0 \text{ et } x \leq 1)$. Cette fois g atteint un maximum sous les contraintes $x \geq 0$ et $x \leq 1$: g est continue et le domaine $[0,1]$ est fermé et borné ("compact"). On a cependant également besoin de croire que Sagemath rend tous les points d'intérêt sans omission pour conclure la discussion.

✍ Comment adapte t-on la discussion pour la recherche du minimum de g sous contrainte ?

Partie 2 : maximisation d'une fonction ℝ^2→ℝ sur un domaine défini par un ensemble d'inégalité.

On considère la fonction d'utilité et les contraintes de l'exercice 2 de la feuille de TD 2 :

$U(x,y) = (x+2)(x+3y)$ sous les contraintes $x \geq 0$, $y \geq 0$, $5x + 3y \leq r$ ($r \geq 0$ paramètre fixé)

On réécrit les contraintes : $c_i(x,y) \leq 0$ pour $i = 1,2,3$, où $c_1(x,y) = -x$, $c_2(x,y) = -y$ et $c_3(x,y) = 5x + 3y - r$.

Le Lagrangien s'écrit $L(x,y,k,l,m) = U(x,y) - k \times c_1(x,y) - l \times c_2(x,y) - m \times c_3(x,y)$.

Les points d'intérêt sont les (x,y,k,l,m) annulant à la fois le gradiant $\left(\frac{\partial}{\partial x}L(x,y,k,l,m), \frac{\partial}{\partial y}L(x,y,k,l,m)\right)$ et $k \times c_1(x,y)$, $l \times c_2(x,y)$, $m \times c_3(x,y)$.
On retient ceux pour lesquels les contraintes sont satisfaites ($c_i(x,y) \leq 0$) et $k, l, m \geq 0$.

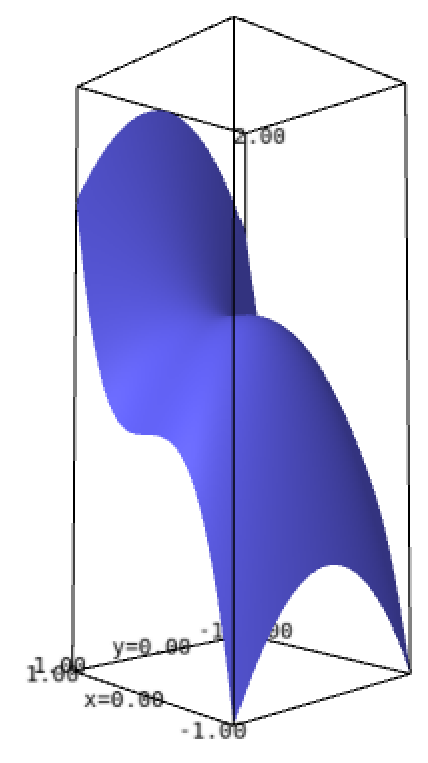
Rq : La convention habituelle pour le Lagrangien est d'écrire $L(x,y,k,l,m) = U(x,y) + k \times c_1(x,y) + l \times c_2(x,y) + m \times c_3(x,y)$ ce qui revient à changer les multiplicateur k, l, m dans les calculs comme dans la discussion qui s'en suit en leurs opposés.

Si U est continue et le domaine déterminé par les contraintes $c_i(x,y) \leq 0$ est fermé borné alors U atteint son maximum sous contrainte. Si aucun des points d'intérêt n'est omis par Sagemath, on peut mener la discussion jusqu'à sa conclusion.

✍ Comment chercherait on le minimum d'une fonction ?

Un exemple de représentation graphique, calcul de gradiant et de recherche de points d'intérêt

In [6]:
```
f(x,y)=2*x^3-y^2
c(x,y)=x-1
show(plot3d(f,[-1,1],[-1,1]))
show("gradiant",(diff(f(x,y),x),diff(f(x,y),y)))
show("points d'intérêt",solve([diff(f(x,y)-k*(x-1),x)==0,diff(f(x,y)-k*(x-1),y)==0,k*(x-1)==0],[x,y,k]))
```



gradiant $(6x^2, -2y)$

points d'intérêt $[[x = 0, y = 0, k = 0], [x = 1, y = 0, k = 6]]$

✍ Faire les calculs et la discussion pour la fonction d'utilité et les contraintes rappelées ci-dessus pour $r = 20$ puis pour $r = 5$.