

Conclusion : la linéarité de la courbe semble vérifiée. La modélisation par une loi de Beer-Lambert semble pertinente.

Test de modèle

La modélisation du nuage de points par une fonction affine est réalisée automatiquement au moyen de la fonction `polyfit`.

```
#trouver le meilleur polynome du premier degre qui colle aux points experimentaux
mod=np.polyfit(C,A,1)
print('mod :', mod)
```

```
mod : [6.17394528e+01 6.03855565e-03]
```

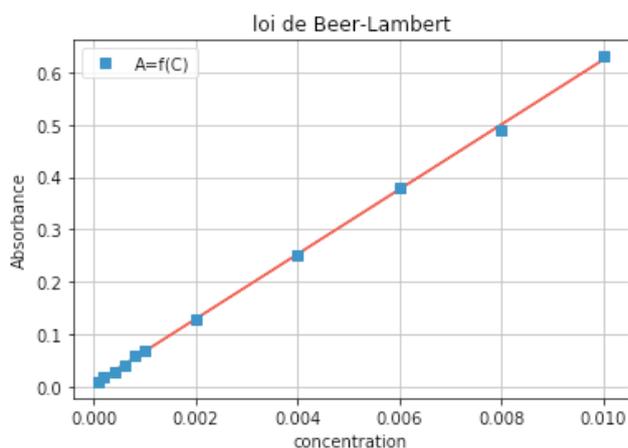
Remarque : il est également possible de faire tracer aux élèves une liste de fonctions affines à l'aide d'une boucle `for`, pour vérifier que l'une d'elles correspond mieux que les autres et ainsi illustrer les critères de recherche de fonction modèle.

Un tableau des valeurs de la fonction modèle est ensuite créé par construction d'une liste pas à pas, grâce à la fonction `append`.

```
#creation tableau vide
model=[]

#calcul pour toutes les valeurs de t des y proposés par le modèle
for i in C :
    model.append(mod[0]*i+mod[1])

#tracer la courbe
plt.plot(C,model,'r-')
plt.grid()
plt.plot(C,A,'s',label="A=f(C)")
plt.xlabel("concentration")
plt.ylabel("Absorbance")
plt.legend()
plt.title("loi de Beer-Lambert")
plt.show()
```



Conclusion : La courbe modèle obtenue automatiquement par `polyfit` semble assez correctement reproduire l'évolution la tendance des points de mesure, ce qui illustre la pertinence de la loi de Beer-Lambert ici. Le coefficient d'absorption molaire peut alors être évalué à partir des paramètres de la courbe modèle.

Retrouvez éduscol sur :

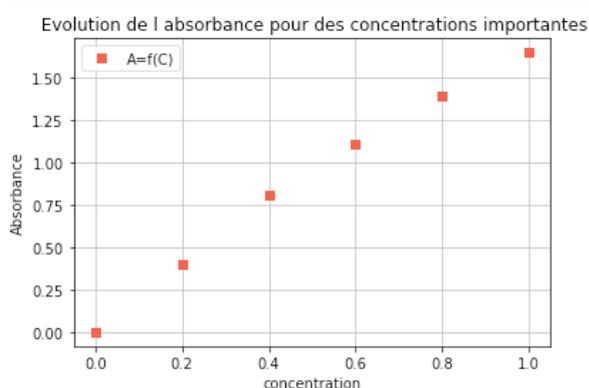


Mise en défaut pour des concentrations plus élevées

L'expérience, et son exploitation sont répétées avec des concentrations plus élevées.

```
C0=1
c2=np.array([0,C0/5,C0/2.5,C0/1.67,C0/1.25,C0])
A2=np.array([0,0.40,0.81,1.11,1.39,1.65])

#tracer la courbe
plt.grid()
plt.plot(c2,A2,'rs',label="A=f(C)")
plt.xlabel("concentration")
plt.ylabel("Absorbance")
plt.legend()
plt.title(" Evolution de l absorbance pour des concentrations importantes")
plt.show()
```



L'évolution n'est plus linéaire. Pour matérialiser l'écart des points expérimentaux à un modèle linéaire, une modélisation linéaire est opérée d'abord à partir des 4 premiers points, puis seulement à partir des 3 premiers points.

```
# on extrait d'abord un tableau des valeurs voulues
c3=c2[:-2]
A3=A2[:-2]
mod3=np.polyfit(c3,A3,1)
model3=[]

#un deuxième tableau
c4=c2[:-3]
A4=A2[:-3]
mod4=np.polyfit(c4,A4,1)
model4=[]

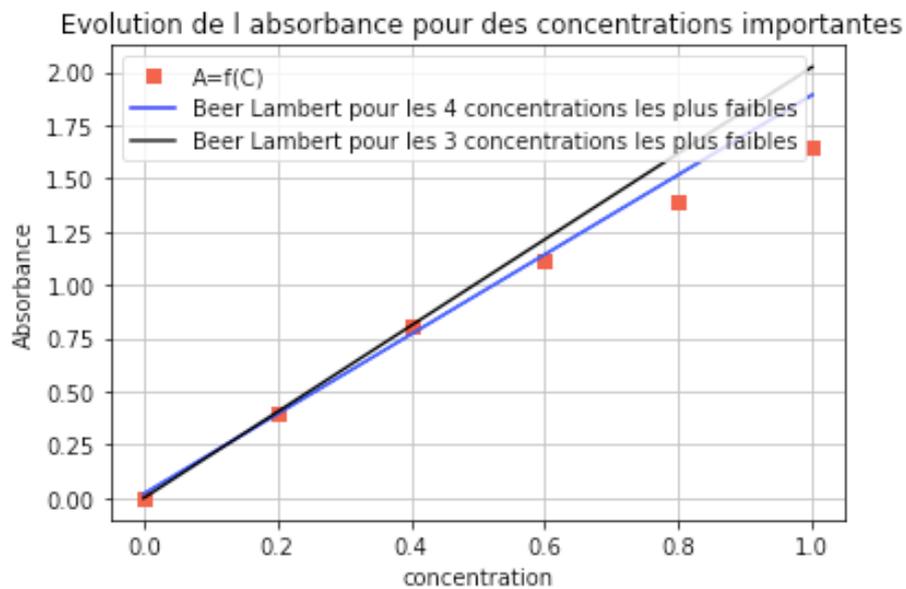
#on définit les fonctions modèles à partir de ces tableaux
for i in c2 :
    model3.append(mod3[0]*i+mod3[1])
for i in c2 :
    model4.append(mod4[0]*i+mod4[1])

plt.grid()
plt.plot(c2,A2,'rs',label="A=f(C)")
plt.plot(c2,model3,'b-',label="Beer Lambert pour les 4 concentrations les plus faibles")
plt.plot(c2,model4,'k-',label="Beer Lambert pour les 3 concentrations les plus faibles")

plt.xlabel("concentration")
plt.ylabel("Absorbance")
plt.legend()
plt.title(" Evolution de l absorbance pour des concentrations importantes")
plt.show()
```

Retrouvez éduscol sur :





Conclusion : l'invalidité de la loi de Beer-Lambert pour des concentrations élevées est ici démontrée.

Retrouvez éduscol sur :

