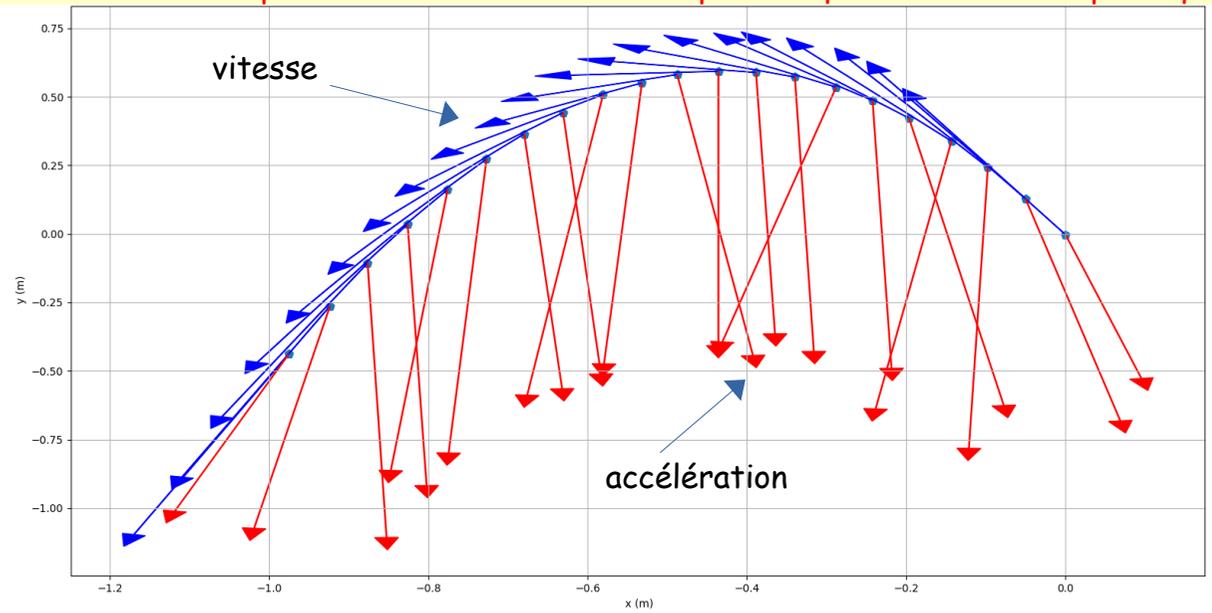
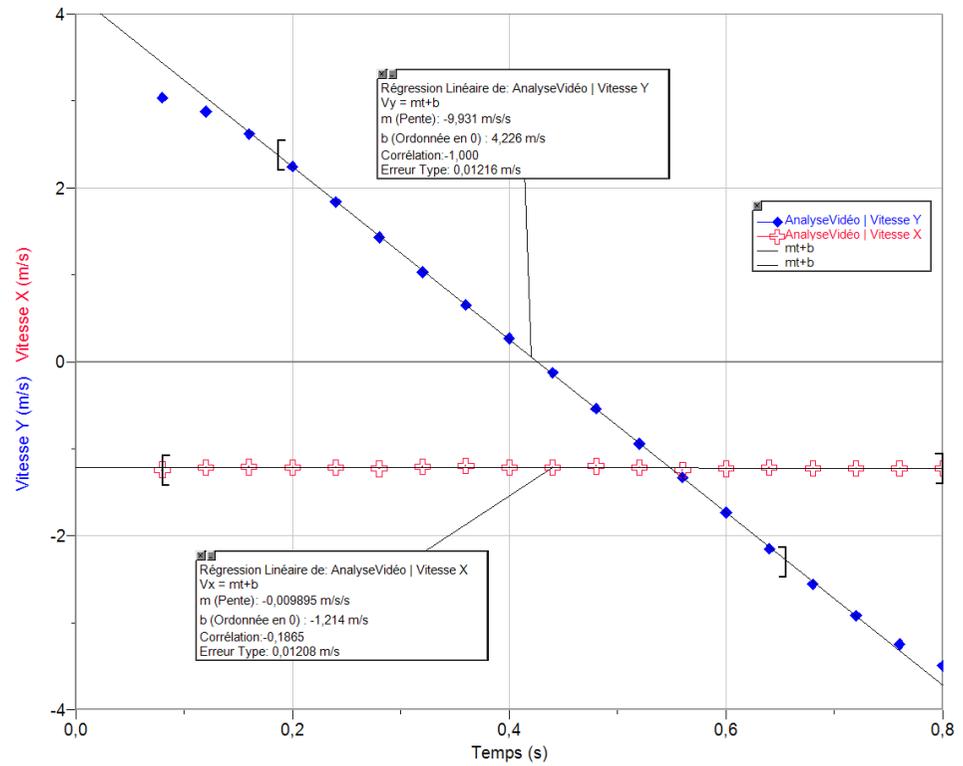


TRACÉ DES VECTEURS VITESSE ET ACCÉLÉRATION EN PYTHON à partir de l'étude expérimentale de la chute libre parabolique de la boule de pétanque.

Tracé des vecteurs par Python (programme ci-après) :



Tracé et modélisation des composantes de la vitesse :



```

47 def vitesse(position,date):
48     n = len(date)
49     d = np.zeros(n,'d')
50     for i in range(1,n-1):
51         d[i] = (position[i+1]-position[i-1])/(date[i+1]-date[i-1])
52         d[0] = (position[1]-position[0])/(date[1]-date[0])
53         d[n-1] = (position[n-1]-position[n-2])/(date[n-1]-date[n-2])
54     return d
55
56 vitesseX = vitesse(abscisse,temps)
57 vitesseY = vitesse(ordonnee,temps)
58
59 def acceleration(vitesse,date):
60
61
62
63
64
65
66
67
68
69 #graphique
70 plt.figure ('Position, vitesse et accélération')
71 plt.clf()
72 plt.grid(True)
73 plt.title('Position, vitesse et accélération')
74 plt.xlabel(" x (m)")
75 plt.ylabel(" y (m)")
76 q = np.arange(len(temps))
77 plt.scatter(abscisse,ordonnee,marker='p',linewidth=3)
78 echelle_v=0.15
79 echelle_a=0.1
80 for i in q :
81     plt.arrow(abscisse[i],ordonnee[i],echelle_v*vitesseX[i],echelle_v*vitesseY[i], head_width=0.03, color="b")
82
83 plt.show()

```

- 1) Pourquoi le tracé des composantes horizontale et verticale de la vitesse est-il bien en accord avec ce que l'on observe sur le tracé des vecteurs vitesse ?

Nous constatons que les vecteurs représentant la vitesse possèdent au début une composante verticale positive car ils pointent vers le haut. Au cours du temps, les vecteurs se rapprochent de l'horizontale puis ils pointent vers le bas. Donc la composante verticale ne cesse de diminuer depuis une composante positive vers des valeurs de plus en plus négatives.

Cela est compatible avec le graphique représentant les variations au cours du temps de la composante verticale de la vitesse : une droite affine de coefficient directeur négatif et d'ordonnée à l'origine positive.

En ce qui concerne la composante horizontale de la vitesse, le graphique représentant l'évolution temporelle de cette composante est une constante négative. Cela est compatible avec l'observation sur le premier tracé : les vecteurs sont dirigés vers les abscisses décroissantes et peuvent très bien avoir des composantes horizontales constantes.

2) Quel commentaire peut-on faire sur les vecteurs accélération tracés ici ?

Les vecteurs représentant l'accélération sont globalement verticaux et vers le bas. À cause des erreurs de mesure sur les positions, amplifiées par les calculs de vitesse puis d'accélération, ils ne correspondent pas tout-à-fait à ce que l'on attendait : vertical, vers le bas, tous égaux au vecteur intensité de la pesanteur \vec{g} .

3) Par analogie avec la définition de la fonction « vitesse », compléter les lignes 60 à 66 pour définir la fonction « acceleration ». Indication : la commande `np.zeros(N,'nom')` génère un tableau avec N zéros dans chaque case (de 0 à N-1) ; cette commande vient de la bibliothèque numpy (alias np ici).

Par analogie, on a :

```
60     n=len(date)
61     d=np.zeros(n,'d')
62     for i in range(1,n-1):
63         d[i]=(vitesse[i+1]-vitesse[i-1])/(date[i+1]-date[i-1])
64         d[0]=(vitesse[1]-vitesse[0])/(date[1]-date[0])
65         d[n-1]=(vitesse[n-1]-vitesse[n-2])/(date[n-1]-date[n-2])
66     return d
```

4) Toujours en vous inspirant de ce qui est fait pour la vitesse, compléter les lignes 67 et 68 pour faire calculer les composantes horizontale et verticale de l'accélération, notées `accelerationX` et `accelerationY`.

On procède comme pour passer de la position à la vitesse aux lignes 56 et 57 :

```
67     accelerationX=acceleration(vitesseX, temps)
68     accelerationY=acceleration(vitesseY, temps)
```

5) Quel est l'effet de la ligne 77 ?

Commande qui génère un graphique avec des points non-reliés avec en abscisse les valeurs de la liste « abscisse » et en ordonnée les valeurs de la liste « ordonnées » Ces listes ont dû être définies avant dans le programme.

6) Quel est l'effet de la boucle des lignes 80 et 81 ? Indication : la commande `np.arange(N)` de la ligne 76 génère un tableau avec N valeurs : 0, 1, ..., N-1. Origine : bibliothèque numpy (alias np ici).

Cette boucle trace les vecteurs vitesse à chacune des dates.

7) Compléter la ligne 82 pour obtenir également le tracé des vecteurs accélération.

Par analogie avec la ligne précédente, on a :

```
82 plt.arrow(abscisse[i],ordonnee[i],echelle_a*accelerationX[i],echelle_a*accelerationY[i],head_width=0.03,color="r")
```