Correction - TP 7

Statistiques descriptives bivariées

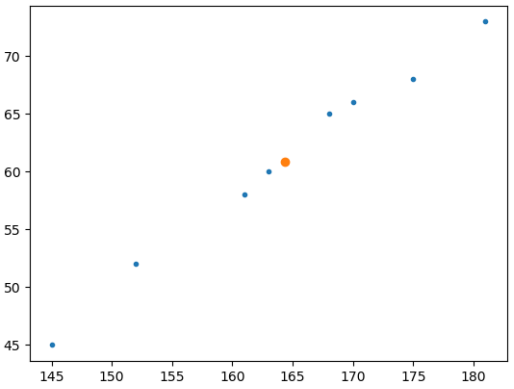
Exercice 1

1. A priori, le poids dépend de la taille donc la variable explicative est X et la variable à expliquer est Y.

2. Voici les instructions pour tracer le nuage de points et le point moyen :

```
#Tracer du nuage de points
x = np.array([161, 170, 152, 181, 163, 145, 168, 175])
y = np.array([58, 66, 52, 73, 60, 45, 65, 68])
plt.plot(x, y, ".")

#Tracer du point moyen
plt.plot(np.mean(x), np.mean(y), "o")

plt.show()
```

On obtient le tracer suivant :

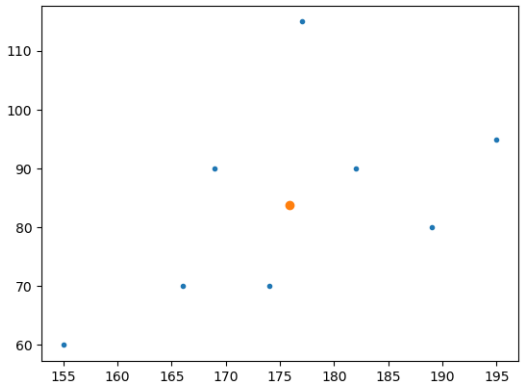


On remarque qu'il y a une forte corrélation linéaire entre la taille X et le poids Y, c'est-à-dire qu'il y a une relation du type $Y \simeq aX + b$.

3. Voici les instructions pour tracer le nuage de points et le point moyen :

```
#Tracer du nuage de points
x = np.array([169, 195, 177, 182, 166, 155, 189, 174])
y = np.array([90, 95, 115, 90, 70, 60, 80, 70])
plt.plot(x, y, ".")

#Tracer du point moyen
plt.plot(np.mean(x), np.mean(y), "o")

plt.show()
```

On obtient le tracer suivant :



Il est difficile de voir apparaitre une fonction qui relie X et Y. On remarque qu'il y a tout de même une variation selon une certaine direction oblique, on a donc $Y \simeq aX + b$ mais cette corrélation linéaire est faible.

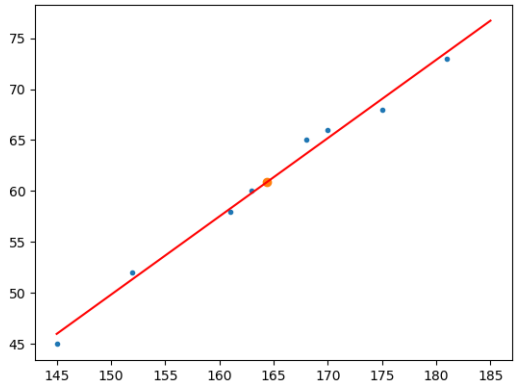
4. Aux USA, il y a un pourcentage important de personnes au comportement alimentaire déréglé... Ceci explique la faible corrélation linéaire entre la taille et le poids.

Exercice 2

1. • Pour l'exemple du Japon :

```python
#Tracer du nuage de points
x = np.array([161, 170, 152, 181, 163, 145, 168, 175])
y = np.array([58, 66, 52, 73, 60, 45, 65, 68])
plt.plot(x, y, ".")

#Tracer du point moyen
plt.plot(np.mean(x), np.mean(y), "o")

# Coefficients a et b de la droite de régression
s = np.mean(x.*y)-np.mean(x)*np.mean(y)
a = s/np.var(x)
b = np.mean(y)-a*np.mean(x)

# Coefficient de corrélation linéaire
r = s/(np.std(x)*np.std(y))
print(r)

#Tracer de la droite de régression linéaire
def f(t):
    return(a*t+b)

t = np.linspace(145,185,100)
plt.plot(t, f(t), color="red")

plt.show()
```

On obtient le tracer suivant :



On utilise l'équation de la droite de régression linéaire :

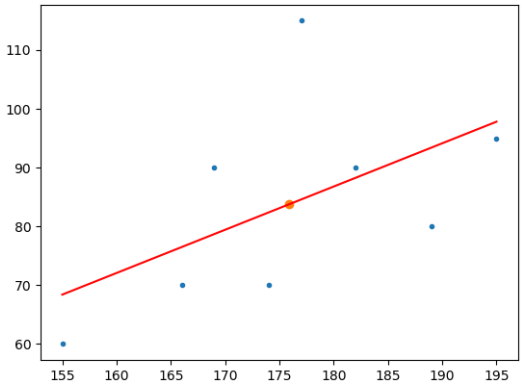
```
>>> f(180)
    72.873447
```

On peut ainsi estimer qu'un individu mesurant 1m80 pèse environ $72,9$ kg. Étant donné que la corrélation linéaire entre X et Y est forte, cette analyse prédictive parait pertinente.

- Pour l'exemple des USA :

```
#Tracer du nuage de points
x = np.array([169, 195, 177, 182, 166, 155, 189, 174])
y = np.array([90, 95, 115, 90, 70, 60, 80, 70])
plt.plot(x, y, ".")

#Tracer du point moyen
plt.plot(np.mean(x), np.mean(y), "o")

# Coefficients a et b de la droite de régression
s = mean(x.*y)-mean(x)*mean(y)
a = s/np.var(x)
b = np.mean(y)-a*np.mean(x)

# Coefficient de corrélation linéaire
r = s/(np.std(x)*np.std(y))
print(r)

#Tracer de la droite de régression linéaire
def f(t):
    return(a*t+b)

t = np.linspace(155,195,100)
plt.plot(t, f(t), color="red")

plt.show()
```

On obtient le tracer suivant :



On utilise l'équation de la droite de régression linéaire :

```
>>> f(180)
    86.783676
```

On peut ainsi estimer qu'un individu mesurant 1m80 pèse environ 86,8 kg. Étant donné que la corrélation linéaire entre X et Y est faible, cette analyse prédictive ne parait pas pertinente.

2. (a) Dans les deux cas, on obtient un coefficient directeur a de signe positif ($a = 0,76$ dans le cas du Japon et $a = 0,73$ dans le cas des USA). Ceci s'interprète par le fait que les caractères X et Y ont tendance à varier dans le même sens. Cela parait conforme à l'intuition, le poids et la taille aillant tendance à varier effectivement dans le même sens.

   (b) Dans le cas du Japon, $r_{x,y} = 0.9953155$. La corrélation linéaire entre les séries X et Y est donc forte. Cela correspond à ce qu'on s'attendait étant donné l'allure du nuage de points.

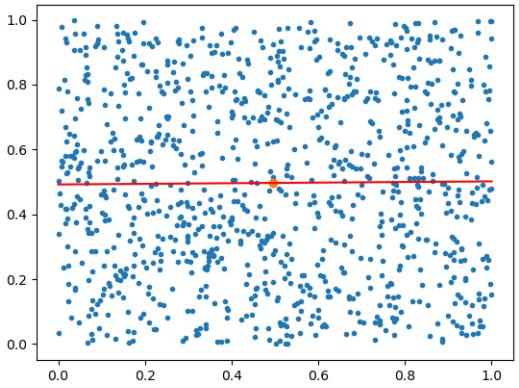
   Dans le cas des USA, $r_{x,y} = 0.5419821$. La corrélation linéaire entre les séries X et Y est donc faible. Cela correspond à ce qu'on s'attendait étant donné l'allure du nuage de points.

Exercice 3

1. On utilise le code suivant :

```
#Tracer du nuage de points
x = rd.random(1000)
y = rd.random(1000)
plt.plot(x, y, ".")

#Tracer du point moyen
plt.plot(np.mean(x), np.mean(y), "o")

#Tracer de la droite de régression linéaire
s = mean(x.*y)-mean(x)*mean(y)
a = s/np.var(x)
b = np.mean(y)-a*np.mean(x)

def f(t):
    return(a*t+b)

t = np.linspace(0,1,10)
plt.plot(t, f(t), color="red")

plt.show()

# Coefficient de corrélation linéaire
r = s/(np.std(x)*np.std(y))
print(r)
```

On obtient le graphe suivant :



La droite de régression linéaire est quasi horizontale. Elle ne semble pas du tout adaptée pour approximer notre nuage de points, il ne semble pas y avoir de corrélation linéaire entre les deux séries statistiques.

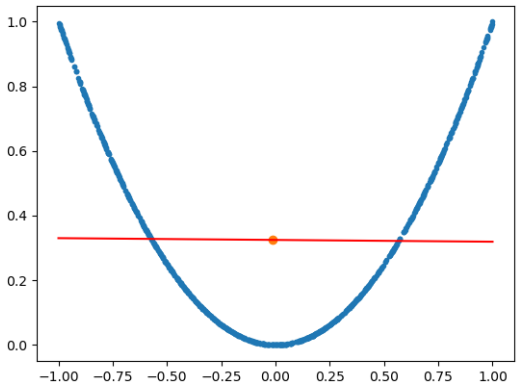
Le coefficient de corrélation linéaire obtenu est `0.0181605`. Il confirme l'indépendance linéaire qu'on avait constaté auparavant. Il explique également pourquoi la droite de régression linéaire est horizontale. Rappelons en effet que le coefficient directeur de cette droite fait intervenir la covariance, qui dans notre cas est voisin de 0.

Tous ces résultats ne sont pas étonnants : les séries statistiques `x` et `y` ont été choisies de manière indépendante l'une de l'autre, la fonction `rd.random` générant des réalisations indépendantes de la loi $\mathcal{U}([0,1])$. Elles sont en particulier linéairement indépendantes, ce qui implique que leur coefficient de corrélation linéaire est proche de 0.

2. On adapte le code précédent :

```
#Tracer du nuage de points
x = rd.uniform(-1, 1, 1000)
y = x**2
plt.plot(x, y, ".")

#Tracer de la droite de régression linéaire
s = mean(x.*y)-mean(x)*mean(y)
a = s/np.var(x)
b = np.mean(y)-a*np.mean(x)

def f(t):
    return(a*t+b)

t = np.linspace(-1,1,10)
plt.plot(t, f(t), color="red")

plt.show()

# Coefficient de corrélation linéaire
r = s/(np.std(x)*np.std(y))
print(r)
```

On obtient le graphe suivant :



La droite de régression linéaire est là aussi quasi horizontale (ce qui indique un coefficient de corrélation linéaire proche de 0) et ne semble également pas du tout adaptée pour approximer notre nuage de points. Le coefficient de corrélation linéaire obtenu le confirme : il est de `-0.0056861`.

On peut conclure que les séries statistiques `x` et `y` sont linéairement indépendantes. Mais attention à ne pas conclure à l'indépendance de ces séries : elles ne le sont clairement pas puisque `y = x**2`.

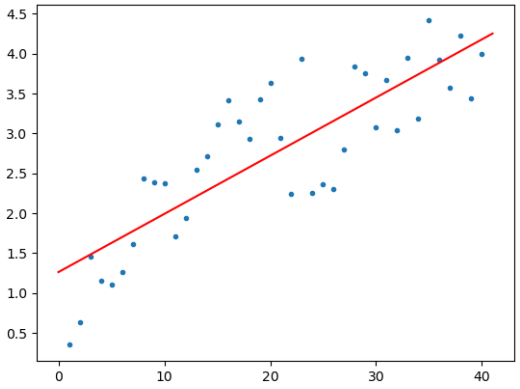
Exercice 4

1. et 2. On utilise le code suivant :

```
#Tracer du nuage de points
x = np.arange(1, 41)
y = np.log(x)+rd.uniform(-1, 1, 40)
plt.plot(x, y, ".")

#Tracer de la droite de régression linéaire
s = mean(x.*y)-mean(x)*mean(y)
a = s/np.var(x)
b = np.mean(y)-a*np.mean(x)

def f(t):
    return(a*t+b)

t = np.linspace(0,41,10)
plt.plot(t, f(t), color="red")

plt.show()

# Coefficient de corrélation linéaire
r = s/(np.std(x)*np.std(y))
print(r)
```

On obtient `0.7537838` comme coefficient de corrélation linéaire, bien en deçà de la barre des 0.9. La corrélation linéaire entre les deux séries statistiques n'est donc pas bonne.

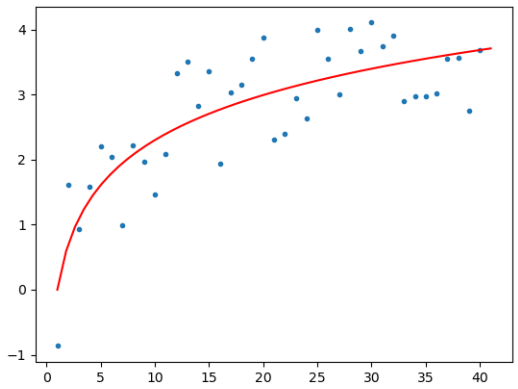
On obtient la représentation suivante du nuage de points et de la droite de régression linéaire associée :



3. Avec le code :

```
#Tracer du nuage de points
x = np.arange(1, 41)
y = np.log(x)+rd.uniform(-1, 1, 40)
plt.plot(x, y, ".")

#Courbe du log
t = linspace(1,41,50)
plt.plot(t, np.log(t), color="red")
```

on obtient :



Le nuage de point se superpose effectivement bien avec la courbe représentative du logarithme.

4. En exécutant les commandes

```
//Coefficient de corrélation linéaire
z = np.log(x)
r = s/(np.std(z)*np.std(y))
print(r)
```

on obtient `0.8377479` comme coefficient de corrélation linéaire. C'est proche des 0.9, ce qui confirme une forte corrélation linéaire entre `log(x)` et `y`.

Exercice 5

1. Avec le code :

```
x = np.array([15.85, 15.7, 15.39, 15.32, 14.85])
y = np.array([230, 280, 360, 410, 525])
s = np.mean(x*y)-np.mean(x)*np.mean(y)
r = s/(np.std(x)*np.std(y))
print(r)
```

on obtient un coefficient de corrélation linéaire de `-0.9951611`.

2. Ces deux séries statistiques sont donc fortement corrélées linéairement. Pourtant, il semble bien difficile d'envisager un lien rationnel entre le nombre de morts sur les routes et les importations de citrons mexicains !
