

# SNT - TP Python 1 : Modélisation d'un réseau social



Pour commencer, exécutez le script suivant pour importer deux modules qui nous seront nécessaires pendant tout le TP.

```
[1] # importation du module "matplotlib" pour créer des graphiques
import matplotlib.pyplot as plt
# importation du module "networkx" pour simuler un réseau social
import networkx as nx
```

## 1 Introduction

Exécuter plusieurs fois le script Python ci-dessous (sans chercher pour le moment à en comprendre le code) :

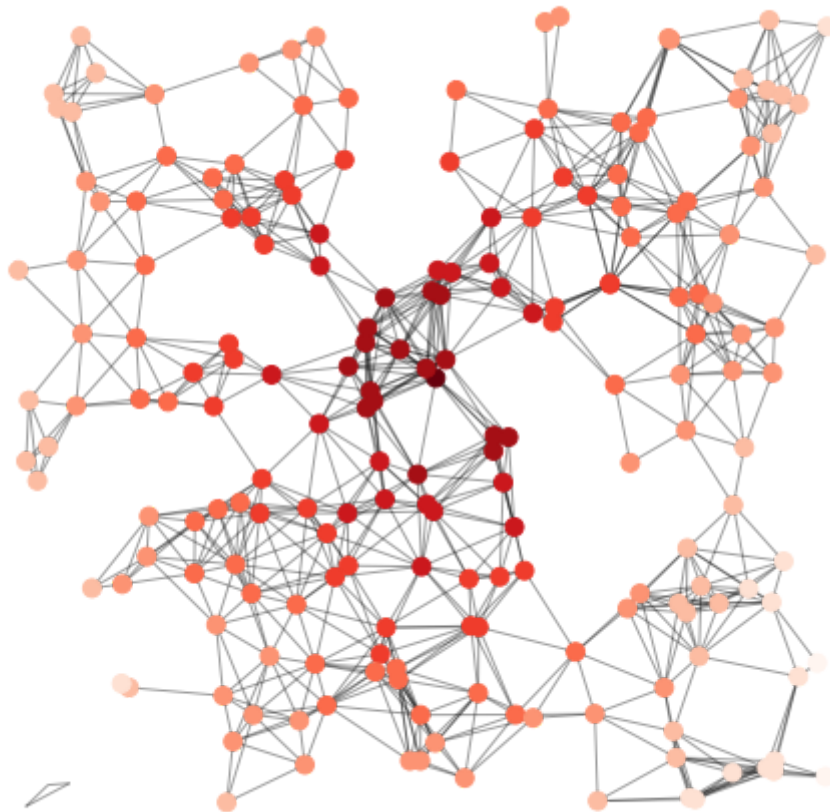
```
[5] G = nx.random_geometric_graph(200, 0.125)
pos = nx.get_node_attributes(G, 'pos')
dmin = 1
ncenter = 0
for n in pos:
    x, y = pos[n]
    d = (x - 0.5)**2 + (y - 0.5)**2
    if d < dmin:
```

```

ncenter = n
dmin = d
p = dict(nx.single_source_shortest_path_length(G, ncenter))
plt.figure(figsize=(8, 8))
nx.draw_networkx_edges(G, pos, nodelist=[ncenter], alpha=0.4)
nx.draw_networkx_nodes(G, pos, nodelist=list(p.keys()),
                        node_size=80,
                        node_color=list(p.values()),
                        cmap=plt.cm.Reds_r)

plt.xlim(-0.05, 1.05)
plt.ylim(-0.05, 1.05)
plt.axis('off')
plt.show()

```



**Exercice 1 :** Expliquer pourquoi la figure générée nous permet de modéliser un réseau social.

---> votre réponse dans cette cellule...

## 2 Découverte du module networkx

Simulons un réseau social avec le module Python networkx.

Lire puis exécuter plusieurs fois le code suivant :

```
[6] # création d'un graphe vide non orienté appelé "monReseau"
monReseau = nx.Graph()

monReseau.add_node("Audrey")
monReseau.add_node("Nicolas")
monReseau.add_node("Alexis")
monReseau.add_node("Hélène")

# dessine (draw) et affiche (show) le graphe "monReseau"
nx.draw(monReseau, with_labels=True)
plt.show()

print("Nombre de personnes =", monReseau.number_of_nodes())
```

MatplotlibDeprecationWarning: pyplot.hold is deprecated.

Future behavior will be consistent with the long-time default:  
plot commands add elements without first clearing the  
Axes and/or Figure.

b = plt.ishold()

C:\ProgramData\Anaconda3\lib\site-packages\networkx\drawing\nx\_pylab.py:138:

MatplotlibDeprecationWarning: pyplot.hold is deprecated.

Future behavior will be consistent with the long-time default:  
plot commands add elements without first clearing the  
Axes and/or Figure.

plt.hold(b)

C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\\_\_init\_\_.py:917:

UserWarning: axes.hold is deprecated. Please remove it from your  
matplotlibrc and/or style files.

warnings.warn(self.msg\_depr\_set % key)

C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\rcsetup.py:152:

UserWarning: axes.hold is deprecated, will be removed in 3.0

warnings.warn("axes.hold is deprecated, will be removed in 3.0")

**Exercice 2 :** Que fait le code de la ligne 6 : `monReseau.add_node("Alexis")` ?

---> votre réponse dans cette cellule...

Lire puis exécuter plusieurs fois le code suivant :

```
[7] monReseau = nx.Graph()

monReseau.add_node("Audrey")
monReseau.add_node("Nicolas")
monReseau.add_node("Alexis")
monReseau.add_node("Hélène")

monReseau.add_edge("Audrey", "Nicolas")
monReseau.add_edge("Audrey", "Alexis")
monReseau.add_edge("Nicolas", "Alexis")
monReseau.add_edge("Hélène", "Audrey")
monReseau.add_edge("Hélène", "Nicolas")

nx.draw(monReseau, with_labels=True)
plt.show()

print("Nombre de personnes =", monReseau.number_of_nodes())
print("Nombre de liens =", monReseau.number_of_edges())
```

Future behavior will be consistent with the long-time default:  
plot commands add elements without first clearing the  
Axes and/or Figure.

b = plt.ishold()

C:\ProgramData\Anaconda3\lib\site-packages\networkx\drawing\nx\_pylab.py:138:  
MatplotlibDeprecationWarning: pyplot.hold is deprecated.

Future behavior will be consistent with the long-time default:  
plot commands add elements without first clearing the  
Axes and/or Figure.

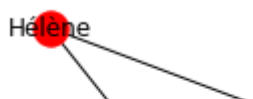
plt.hold(b)

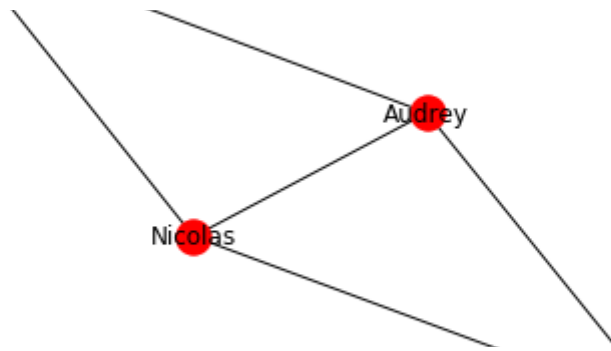
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\\_\_init\_\_.py:917:  
UserWarning: axes.hold is deprecated. Please remove it from your  
matplotlibrc and/or style files.

warnings.warn(self.msg\_depr\_set % key)

C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\rcsetup.py:152:  
UserWarning: axes.hold is deprecated, will be removed in 3.0

warnings.warn("axes.hold is deprecated, will be removed in 3.0")





**Exercice 3 :** Que fait le code de la ligne 8 : `monReseau.add_edge("Audrey","Nicolas")` ?

---> *votre réponse dans cette cellule...*

**Exercice 4 :** Quelles sont les deux personnes qui ne sont pas directement en relation ? Que faudrait-il faire pour les mettre en relation ?

---> *votre réponse dans cette cellule...*

### 3 Un graphe orienté

Lire puis exécuter plusieurs fois le code suivant, qui est presque le même que le code précédent :

```
[8] monReseau = nx.DiGraph() # création d'un graphe orienté

monReseau.add_node("Audrey")
monReseau.add_node("Nicolas")
monReseau.add_node("Alexis")
monReseau.add_node("Hélène")

monReseau.add_edge("Audrey", "Nicolas") # lien de Audrey vers
Nicolas
monReseau.add_edge("Audrey", "Alexis")
monReseau.add_edge("Nicolas", "Alexis")
```

```

monReseau.add_edge("Hélène", "Audrey")
monReseau.add_edge("Hélène", "Nicolas")
monReseau.add_edge("Alexis", "Audrey")

nx.draw(monReseau, with_labels=True)
plt.show()

print("Nombre de personnes =", monReseau.number_of_nodes())
print("Nombre de liens =", monReseau.number_of_edges())

```

Future behavior will be consistent with the long-time default:  
 plot commands add elements without first clearing the  
 Axes and/or Figure.

```
b = plt.ishold()
```

C:\ProgramData\Anaconda3\lib\site-  
 packages\networkx\drawing\nx\_pyplot.py:138:

MatplotlibDeprecationWarning: pyplot.hold is deprecated.

Future behavior will be consistent with the long-time default:  
 plot commands add elements without first clearing the  
 Axes and/or Figure.

```
plt.hold(b)
```

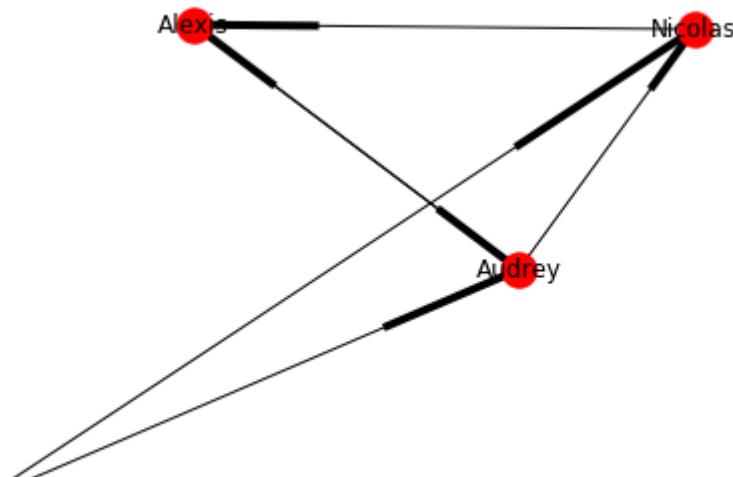
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\\_\_init\_\_.py:917:  
 UserWarning: axes.hold is deprecated. Please remove it from your  
 matplotlibrc and/or style files.

```
warnings.warn(self.msg_depr_set % key)
```

C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\rcsetup.py:152:

UserWarning: axes.hold is deprecated, will be removed in 3.0

```
warnings.warn("axes.hold is deprecated, will be removed in 3.0")
```



**Exercice 5 :** Qu'est-ce qui a changé par rapport au graphe précédent ?

---> votre réponse dans cette cellule...

**Exercice 6 :** Pourquoi la relation entre Audrey et Alexis est-elle différente des autres ?  
Pourquoi voit-on 5 segments mais 6 liens ?

---> votre réponse dans cette cellule...

**Exercice 7 :** Ce dernier graphe est dit **orienté**, par opposition au graphe du paragraphe précédent qui est dit **non orienté**. Parmi Facebook, Twitter, Instagram, WhatsApp, Snapchat, citer un réseau social orienté et un réseau social non orienté.

---> votre réponse dans cette cellule...

## 4 Mon petit réseau social

**Exercice 8 :** Choisir six prénoms de votre choix et simuler graphiquement un réseau social **non orienté** entre ces personnes, à l'aide du code appris précédemment.

```
[ ] # votre code ici  
...  
pass
```

## 5 Un réseau social où tout le monde est amis

Lire puis exécuter plusieurs fois le programme suivant.

```
[9] monRezo = nx.Graph()
```

```

Liste = ["Ronaldo", "Messi", "Totti", "Salah", "Mbappé", "Neymar"]

for a in Liste:
    monRezo.add_node(a)

for b in Liste:
    for c in Liste:
        monRezo.add_edge(b, c)

nx.draw(monRezo, with_labels = True)
plt.show()

print("Nombre de personnes =", monRezo.number_of_nodes())
print("Nombre de liens =", monRezo.number_of_edges())

```

Future behavior will be consistent with the long-time default:  
plot commands add elements without first clearing the  
Axes and/or Figure.

b = plt.ishold()

C:\ProgramData\Anaconda3\lib\site-packages\networkx\drawing\nx\_pylab.py:138:

MatplotlibDeprecationWarning: pyplot.hold is deprecated.

Future behavior will be consistent with the long-time default:  
plot commands add elements without first clearing the  
Axes and/or Figure.

plt.hold(b)

C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\\_\_init\_\_.py:917:

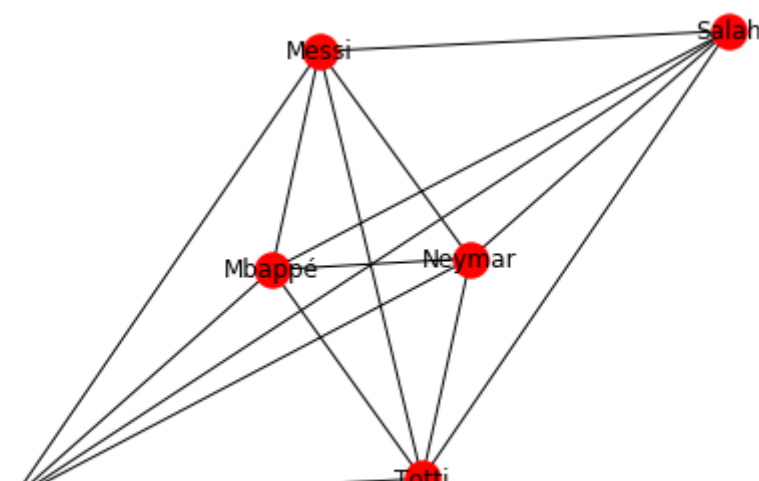
UserWarning: axes.hold is deprecated. Please remove it from your  
matplotlibrc and/or style files.

warnings.warn(self.msg\_depr\_set % key)

C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\rcsetup.py:152:

UserWarning: axes.hold is deprecated, will be removed in 3.0

warnings.warn("axes.hold is deprecated, will be removed in 3.0")



**Exercice 9 :** Expliquer ce que font les lignes 8, 9 et 10 du programme.



---> votre réponse dans cette cellule...

## 6 Un réseau social plus ou moins social

La fonction `random()` renvoie un nombre réel au hasard entre 0 et 1.

Lire puis exécutez plusieurs fois de suite le programme suivant :

```
[10] from random import random

monRezo = nx.Graph()

Liste = ["Ronaldo", "Messi", "Totti", "Salah", "Pogba", "Neymar"]
p = 0.3

for a in Liste:
    monRezo.add_node(a)

for b in Liste:
    for c in Liste:
        if random() < p:
            monRezo.add_edge(b, c)

nx.draw(monRezo, with_labels = True)
plt.show()

print("Nombre de personnes =", monRezo.number_of_nodes())
print("Nombre de liens =", monRezo.number_of_edges())
```

```
Future behavior will be consistent with the long-time default:
plot commands add elements without first clearing the
Axes and/or Figure.
```

```
b = plt.ishold()
```

```
C:\ProgramData\Anaconda3\lib\site-
packages\networkx\drawing\nx_pylab.py:138:
```

```
MatplotlibDeprecationWarning: pyplot.hold is deprecated.
```

```
Future behavior will be consistent with the long-time default:
plot commands add elements without first clearing the
Axes and/or Figure.
```

```
plt.hold(b)
```

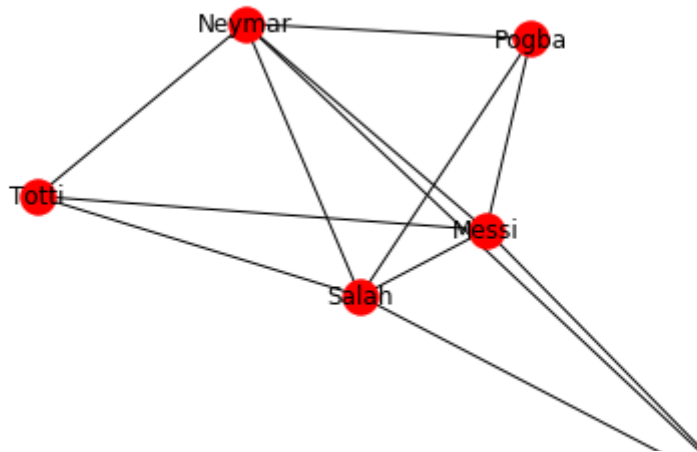
```
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\__init__.py:917:
```

```
UserWarning: axes.hold is deprecated. Please remove it from your
matplotlibrc and/or style files.
```

```
warnings.warn(self.msg_depr_set % key)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\matplotlib\__init__.py:152:
```

```
C:\ProgramData\Anaconda3\env\site-packages\matplotlib\rcsetup.py:152:
UserWarning: axes.hold is deprecated, will be removed in 3.0
  warnings.warn("axes.hold is deprecated, will be removed in 3.0")
```



**Exercice 10 :** Par défaut  $p = 0.3$  (ligne 6 du programme). Donnez à  $p$  la valeur  $0.1$  puis exécutez plusieurs fois le programme. Donnez à  $p$  la valeur  $0.9$  puis exécutez plusieurs fois le programme. Que constatez-vous ? Pourquoi ?

---> votre réponse dans cette cellule...

## 7 D'autres réseaux ?

**Bonus :** Nous venons de simuler un réseau social avec le module Python `networkx`. Quels autres types de réseaux pourrait-on modéliser avec `networkx` ?

---> votre réponse dans cette cellule...

---