

## Ressources nécessaires

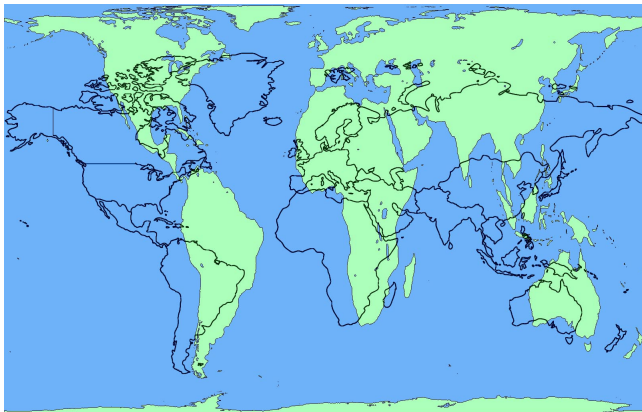
Ce cours est entièrement conçu en **Python 3**. Si vous êtes habitués à R, vous pouvez avec quelques aménagements suivre ce cours en R ; pour tous les autres nous recommandons l'installation des logiciels suivants :

- Python 3.x. Si vous n'êtes pas familiers avec l'installation de packages Python, choisissez la distribution Miniconda.
- l'IDE Pyzo. Configurez le shell pour utiliser Miniconda.
- les packages suivants (via l'IDE avec conda ou en utilisant pip) : NumPy, SciPy, matplotlib, pandas et sklearn

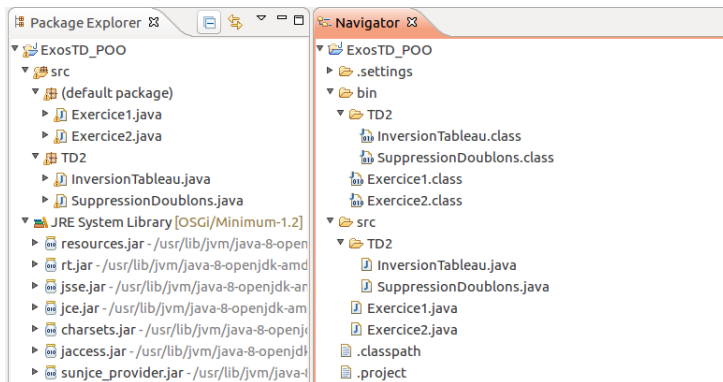
# Des représentations naturelles ?



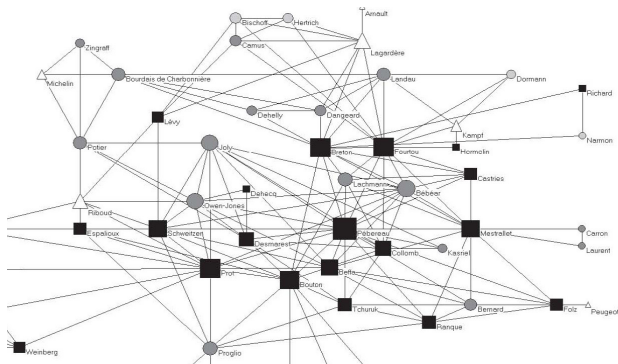
# Des représentations naturelles ?



# Des représentations naturelles ?



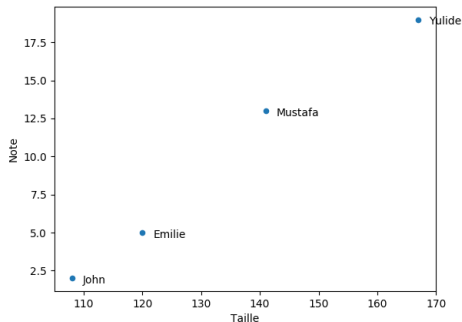
# Des représentations naturelles ?



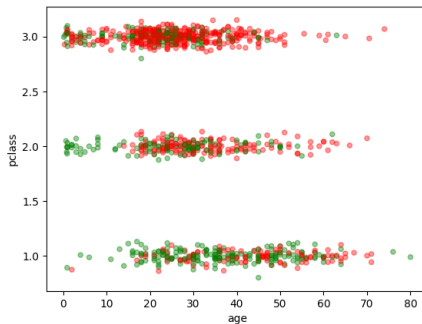
# Problème : multidimensionnalité



# Problème : multidimensionnalité



# Problème : multidimensionnalité





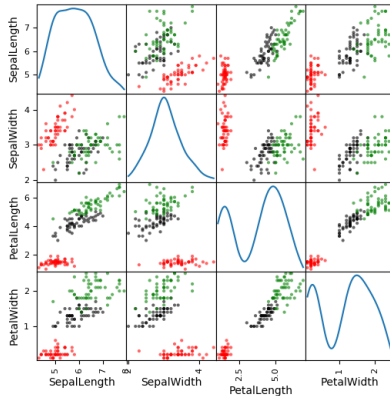
# Problème : multidimensionnalité

0.000 ;0.026 ;0.271 ;0.000 ;0.000 ;0.001 ;0.036 ;0.039 ;0.025 ;0.002 ;0.002 ;0.001 ;0.001 ;0.009 ;0.000 ;0.002 ;0.002 ;0.000 ;0.000 ;  
 0.000 ;0.053 ;0.044 ;0.004 ;0.003 ;0.004 ;0.143 ;0.005 ;0.002 ;0.007 ;0.002 ;0.007 ;0.002 ;0.003 ;0.001 ;0.002 ;0.003 ;0.003 ;0.003 ;  
 0.000 ;0.021 ;0.134 ;0.000 ;0.001 ;0.000 ;0.000 ;0.003 ;0.006 ;0.001 ;0.000 ;0.000 ;0.000 ;0.003 ;0.000 ;0.000 ;0.000 ;0.002 ;0.000 ;  
 0.000 ;0.006 ;0.046 ;0.002 ;0.001 ;0.001 ;0.055 ;0.004 ;0.001 ;0.001 ;0.001 ;0.009 ;0.038 ;0.003 ;0.001 ;0.000 ;0.002 ;0.001 ;0.025 ;  
 0.000 ;0.000 ;0.013 ;0.000 ;0.000 ;0.000 ;0.000 ;0.000 ;0.080 ;0.000 ;0.000 ;0.000 ;0.000 ;0.000 ;0.000 ;0.000 ;0.000 ;0.000 ;  
 0.000 ;0.001 ;0.057 ;0.000 ;0.000 ;0.000 ;0.018 ;0.005 ;0.009 ;0.000 ;0.000 ;0.003 ;0.000 ;0.209 ;0.000 ;0.000 ;0.001 ;0.000 ;0.000 ;  
 0.000 ;0.001 ;0.096 ;0.001 ;0.001 ;0.001 ;0.112 ;0.005 ;0.001 ;0.001 ;0.001 ;0.022 ;0.001 ;0.007 ;0.001 ;0.000 ;0.003 ;0.001 ;0.001 ;  
 0.000 ;0.066 ;0.019 ;0.000 ;0.000 ;0.047 ;0.011 ;0.010 ;0.000 ;0.001 ;0.000 ;0.001 ;0.001 ;0.001 ;0.021 ;0.064 ;0.001 ;0.010 ;0.012 ;  
 0.000 ;0.001 ;0.049 ;0.001 ;0.000 ;0.000 ;0.030 ;0.003 ;0.000 ;0.000 ;0.000 ;0.002 ;0.000 ;0.002 ;0.000 ;0.000 ;0.001 ;0.000 ;0.000 ;  
 0.000 ;0.002 ;0.252 ;0.002 ;0.001 ;0.001 ;0.031 ;0.037 ;0.001 ;0.001 ;0.001 ;0.003 ;0.001 ;0.005 ;0.000 ;0.000 ;0.002 ;0.000 ;0.000 ;  
 0.000 ;0.001 ;0.013 ;0.000 ;0.000 ;0.000 ;0.001 ;0.000 ;0.002 ;0.001 ;0.001 ;0.001 ;0.000 ;0.002 ;0.120 ;0.001 ;0.000 ;0.000 ;0.000 ;  
 0.000 ;0.007 ;0.063 ;0.004 ;0.001 ;0.000 ;0.072 ;0.005 ;0.001 ;0.002 ;0.001 ;0.015 ;0.002 ;0.002 ;0.001 ;0.000 ;0.001 ;0.197 ;0.001 ;  
 0.000 ;0.005 ;0.001 ;0.000 ;0.000 ;0.000 ;0.000 ;0.000 ;0.001 ;0.000 ;0.001 ;0.000 ;0.000 ;0.000 ;0.000 ;0.000 ;0.000 ;0.000 ;0.000 ;

# Problème : multidimensionnalité



# Scatter Matrix

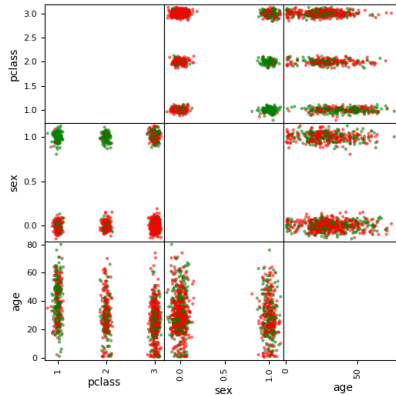


# Scatter Matrix

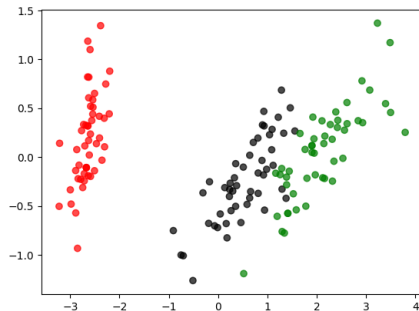
```
import pandas as pd
from matplotlib import pyplot as plt
from pandas.plotting import scatter_matrix

iris_data = pd.read_csv('./iris.csv')
colMap={"Iris-setosa":"red","Iris-virginica":"green",
        "Iris-versicolor":"black"}
colors=list(map(lambda x:colMap.get(x),iris_data.Name))
scatter_matrix(iris_data , alpha=0.6, figsize=(6, 6),
               diagonal='kde',c=colors)
plt.show()
```

# Scatter Matrix



# ACP



# ACP

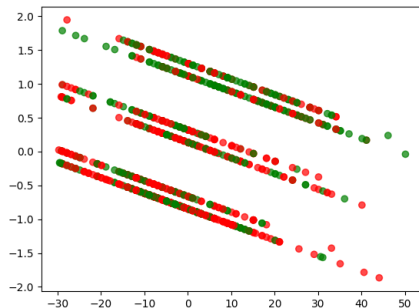


```
import matplotlib.pyplot as plt
from sklearn import datasets
from sklearn.decomposition import PCA

iris = datasets.load_iris()
X,Y = iris.data, iris.target
colMap={0:"red",1:"green",2:"black"}
colors=list(map(lambda x:colMap.get(x),Y))
X_2ev = PCA(n_components=2).fit_transform(X)
plt.scatter(X_2ev[:,0],X_2ev[:,1],alpha=0.7,c=colors)

plt.show()
```

# ACP





# ACP

