

Régression linéaire à base de Scikit-learn

L'objectif de cet exemple consiste à montrer comment utiliser l'API Scikit-learn, pour créer un modèle de prédiction linéaire à base d'un jeu de données .

commençons tout d'abord par importer les bibliothèques nécessaires pour créer notre modèle

```
In [8]: import numpy as np # bibliothèque très puissante dans le calcul scientifique
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt # pour visualiser nos données (Scripting layer)
# pour visualiser nos données dans le même notebook :
%matplotlib inline
```

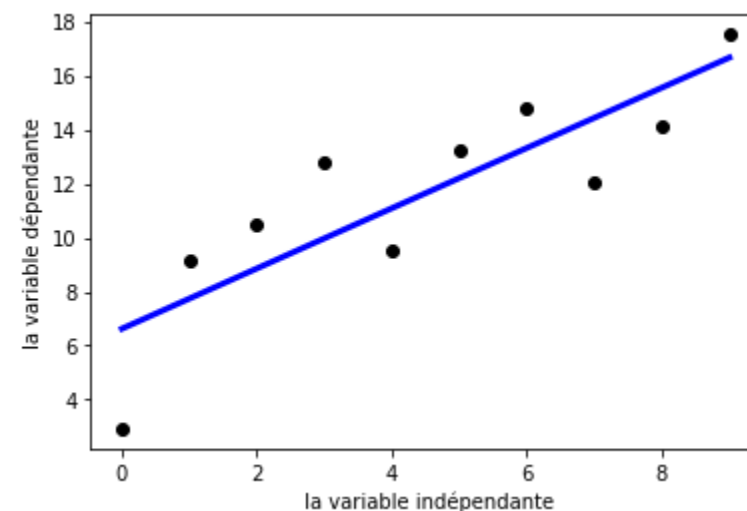
```
In [9]: # génération aléatoire de dix points
length=10
X=np.arange(length,dtype=float).reshape((length,1))
y=X+(np.random.rand(length)*10).reshape(length,1)
```

```
In [10]: #Création d'un modèle linéaire à base de X et y
reg=LinearRegression()
```

```
In [11]: #L'utilisation de la méthode .fit(X,y) pour s'ajuster aux données d'apprentissage
reg.fit(X,y)
```

```
Out[11]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

```
In [18]: #plot des dix points
plt.scatter(X,y,color='black')
#plot de modèle
plt.plot(X, reg.predict(X), color='blue', linewidth=3)
plt.xlabel(" la variable indépendante")
plt.ylabel(("la variable dépendante"))
plt.show()
```



une interprétation évidente qu'on peut donner à base de cette visualisation si jamais rencontrer avec des jeux de données réelle , c'est qu'il y a une forte corrélation linéaire positive entre la variable dépendante et la variable indépendante