

```

#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Thu Feb 24 09:40:42 2022

@author: raphaelbailly
"""

import numpy as np

np.random.seed(0)

X = np.random.randn(10,3)
Y = np.random.randn(10,2)

def grad(f, epsilon = 10**(-7)):
    def gradf(x,t):
        n = x.shape[0]
        y = []
        for i in range(n):
            ei = np.zeros(n,float)
            ei[i] = epsilon
            y.append((f(x+ei,t)-f(x,t))/epsilon)
        return(np.array(y))
    return(gradf)

def regression_square(param, data):
    X = data[0]
    Y = data[1]
    A = param[:6]
    b = param[6:]
    A = A.reshape(3,2)
    square_error = 0
    for i in range(X.shape[0]):
        square_error += ((Y[i]-X[i]@A-b)**2).sum()
    return(square_error)

def desc_grad(f, N, param0, data, mu = 0.1):
    param = param0
    gradf = grad(f)
    for i in range(N):
        param = param - mu*gradf(param, data)
    return(param)

A = np.random.randn(3,2)
b = np.random.randn(2)

a = A.reshape(6)
param0 = np.hstack((a,b))

data = [X,Y]

param_optimal = desc_grad(regression_square, 100,
                           param0, data, mu = 0.01)

Aoptimal = param_optimal[:6]
boptimal = param_optimal[6:]
Aoptimal = Aoptimal.reshape(3,2)

```

```
print(Aoptimal)
print(boptimal)
```

```
Xmean = X.mean(axis = 0)
Ymean = Y.mean(axis = 0)
Xbar = X-Xmean
Ybar = Y-Ymean
Aopt = np.linalg.inv(Xbar.T@Xbar)@Xbar.T@Ybar
bopt = Ymean - Xmean@Aopt
```

```
print(Aopt)
print(bopt)
```