

```
In [48]: from sklearn.datasets import load_diabetes
import autograd.numpy as np
import pandas as pd
```

```
In [97]: #1.
X, y = load_diabetes(return_X_y=True)
```

```
In [114]: def loss_fun(y_true, y_pred):
    error = y_true - y_pred
    resolt = []
    for x in error:
        resolt.append(0.5*x**2 if x<=delta else 0.5*(abs(x)-0.5*delta))
    resolt = np.array(resolt)
    return(resolt)
def empirical_risk(theta, X, y):
    X = np.column_stack((np.ones_like(y), X))
    y_pred = X @ theta.T
    return np.mean(loss_fun(y, y_pred))
```

```
In [115]: from scipy.optimize import minimize
from autograd import grad
theta_0 = np.zeros(1+10) #需加 1 個截距項。
grad_fun = grad(empirical_risk)
result = minimize(empirical_risk, theta_0, (X, y), method='CG', jac=grad_fun)
result
```

```
Out[115]: fun: 52.201178248802904
jac: array([ 0.00747043, -0.01042827,  0.00117886, -0.01410306, -0.00939635,
           -0.01974502, -0.01957315,  0.01470534, -0.02270541, -0.02674988,
           -0.0014914 ])
message: 'Desired error not necessarily achieved due to precision loss.'
nfev: 75
nit: 1
njev: 68
status: 2
success: False
x: array([ 5.59796937e+01, -2.04354360e-14,  7.46831125e-15, -4.50829874e-14,
          7.29533533e-15, -5.17106963e-15,  7.23464202e-15, -2.54911883e-14,
          2.17949660e-14, -2.14611528e-14, -1.92033619e-14])
```

```
In [109]: from sklearn.linear_model import HuberRegressor
X, y = load_diabetes(return_X_y=True, as_frame=True)
huber = HuberRegressor(alpha=0)
huber.fit(X,y)
#coefficient
print("coefficient:", huber.coef_)
#截距
print("\nintercept:", huber.intercept_)
print("\n此model為:\nY = {}".format(huber.intercept_), end = "")
for i, name in enumerate(X.columns):
    print("\n + {a} X_{b}".format(a=huber.coef_[i], b=name), end = "")
```

```
coefficient: [ -32.67227245 -280.43454285  506.99137825  339.50673741 -616.97855764
  329.84406703 -18.2761147  113.40296118  739.7939154  40.2337496 ]
```

```
intercept: 151.27013716490694
```

此model為:

```
Y = 151.27013716490694
+ -32.67227245273352 X_age
+ -280.43454284576984 X_sex
+ 506.991378252603 X_bmi
+ 339.50673740909076 X_bp
+ -616.9785576406506 X_s1
+ 329.84406703316563 X_s2
+ -18.27611470047831 X_s3
+ 113.40296117987923 X_s4
+ 739.7939153950564 X_s5
+ 40.23374959731776 X_s6
```

```
C:\Users\user\.conda\envs\datascience\lib\site-packages\sklearn\linear_model\_huber.py:296: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.
```

```
Increase the number of iterations (max_iter) or scale the data as shown in:
https://scikit-learn.org/stable/modules/preprocessing.html (https://scikit-learn.org/stable/modules/preprocessing.html)
self.n_iter_ = _check_optimize_result("lbfgs", opt_res, self.max_iter)
```

```
In [ ]:
```