

```
In [1]: import sklearn.datasets
import pandas as pd
import numpy as np

X,y = sklearn.datasets.load_diabetes(return_X_y=True, as_frame=True)
```

```
In [2]: # 1.
from sklearn.linear_model import Ridge
ridge = Ridge(alpha=0.1)
ridge.fit(X,y)

#有成功 fit 就給分(能多給一些資訊最好)
#coefficient
print("coefficient:",ridge.coef_)
#截距
print("\nintercept:", ridge.intercept_)

# model
print("\n此model為:\nY = {}".format(ridge.intercept_),end = "")
for i,name in enumerate(X.columns):
    print("\n + ({}X_{}_b)".format(a=ridge.coef_[i],b=name),end = "")

coefficient: [ 1.30734895 -207.19481363 489.69108009 301.76943732 -83.46607377
-70.82809551 -188.68016351 115.7127025 443.81405412 86.74853944]

intercept: 152.1334841628964

此model為:
Y = 152.1334841628964
+ (1.3073489468055493)X_age
+ (-207.19481363269662)X_sex
+ (489.6910800850107)X_bmi
+ (301.7694373160264)X_bp
+ (-83.46607377280742)X_s1
+ (-70.82809550533649)X_s2
+ (-188.68016350975518)X_s3
+ (115.71270250415137)X_s4
+ (443.81405411972656)X_s5
+ (86.74853944271864)X_s6
```

```
In [3]: #2.
#方法一

from sklearn.linear_model import RidgeCV
ridgeCV = RidgeCV(alphas=[10**i for i in range(-10,1)]).fit(X,y)

#取得的 hyperparameter alpha
print("答案(ridgeCV) : \nalpha:",ridgeCV.alpha_)

print("\n以下答案不用show。")

#coefficient
print("\ncoefficient:",ridgeCV.coef_)

#截距
print("\nintercept:", ridgeCV.intercept_)

# model
print("\n此model為:\nY = {}".format(ridgeCV.intercept_),end = "")
for i,name in enumerate(X.columns):
    print("\n + ({}X_{}_b)".format(a=ridgeCV.coef_[i],b=name),end = "")

答案(ridgeCV) :
alpha: 0.01

以下答案不用show。

coefficient: [ -7.19945679 -234.55293001 520.58313622 320.52335582 -380.60706569
150.48375154 -78.59123221 130.31305868 592.34958662 71.1337681 ]

intercept: 152.13348416289645

此model為:
Y = 152.13348416289645
+ (-7.199456785271423)X_age
+ (-234.55293000522124)X_sex
+ (520.583136216344)X_bmi
+ (320.5233558209702)X_bp
+ (-380.6070656879183)X_s1
+ (150.48375153706456)X_s2
+ (-78.59123221333311)X_s3
+ (130.31305868099844)X_s4
+ (592.3495866188416)X_s5
+ (71.1337681034338)X_s6
```

```

In [4]: from sklearn.model_selection import GridSearchCV
# estimator可選擇 Ridge()
# Ridge()的 argument 是 alpha
parameter = {'alpha':[10**i for i in range(-10,1)]}

GCV = GridSearchCV(estimator = Ridge(), param_grid = parameter)
GCV.fit(X,y)
#取得的 hyperparameter alpha
print("答案(GCV) : \nalpha:",GCV.best_params_['alpha'])
# 也可以 GCV.best_estimator_.alpha

print("\n以下答案不用show。")

#coefficent
print("\ncoefficent:\n",GCV.best_estimator_.coef_)

#截距
print("\nintercept:", GCV.best_estimator_.intercept_)

# model
print("\n此model為:\nY = {}".format(GCV.best_estimator_.intercept_),end = "")
for i,name in enumerate(X.columns):
    print("\n + ({a})X_{b}".format(a=GCV.best_estimator_.coef_[i],b=name),end = "")

```

答案(GCV) :

alpha: 0.0001

以下答案不用show。

coefficent:

```
[ -9.96228958 -239.74191297 519.90178223 324.33049163 -783.36918479
 469.75119728 97.15076946 176.00399634 747.93655335 67.6781432 ]
```

intercept: 152.1334841628965

此model為:

```
Y = 152.1334841628965
+ (-9.96228958260985)X_age
+ (-239.7419129679896)X_sex
+ (519.9017822346113)X_bmi
+ (324.3304916311564)X_bp
+ (-783.3691847878055)X_s1
+ (469.75119727781566)X_s2
+ (97.15076946049741)X_s3
+ (176.0039963420826)X_s4
+ (747.936553349532)X_s5
+ (67.67814320057171)X_s6
```

In []: