

# ML線性回歸跟boxz相同.py 技術需求整理

## 一、技術需求

### 1. 基本環境

- Python 3.x
- 常用數據處理與科學計算套件（如 numpy、pandas）
- 視覺化套件（如 matplotlib）
- 若有用到機器學習框架，則需安裝 scikit-learn 或 PyTorch 等<sup>[1] [2] [3]</sup>。

### 2. 程式主要功能

- 實現線性回歸（Linear Regression）模型，並可選擇以最小二乘法或梯度下降法訓練參數<sup>[1] [3] [4]</sup>。
- 支援單變量或多變量特徵輸入<sup>[2] [3]</sup>。
- 可進行模型訓練、預測與模型評估（如均方根誤差 RMSE）<sup>[1] [3] [4]</sup>。

### 3. 資料需求

- 輸入資料需包含特徵（X）與標籤（y），格式可為 list、numpy array 或 pandas DataFrame<sup>[1] [2] [4]</sup>。

## 二、模型說明

### 1. 線性回歸模型

- 形式： $y = w \cdot x + b$ （單變量）或  $y = w_1 x_1 + w_2 x_2 + \dots + b$ （多變量）<sup>[2] [3] [4]</sup>。
- 目標：最小化預測值與真實值之間的均方誤差（MSE）<sup>[3] [4]</sup>。

### 2. 參數與超參數

名稱	說明	預設值/範例
learning_rate	學習率，控制梯度下降時每步更新幅度	0.01、0.1 等 <sup>[1] [3]</sup>
n_epochs	訓練迭代次數	100、1000 等 <sup>[1] [3]</sup>
batch_size	每批訓練樣本數（若採用 mini-batch SGD）	10、32 等 <sup>[3] [5]</sup>

名稱	說明	預設值/範例
w	權重參數，依特徵數決定數量	隨機初始化 <sup>[3]</sup> <sup>[4]</sup>
b	偏置參數	隨機初始化 <sup>[3]</sup> <sup>[4]</sup>
regularization	正則化方法（如 L2、L1），防止過擬合	可選 <sup>[4]</sup> <sup>[6]</sup>
random_seed	隨機種子，確保結果可重現	任意整數 <sup>[6]</sup>

### 三、主要流程

#### 1. 資料前處理

- 讀取與整理資料，必要時進行標準化或編碼<sup>[2]</sup> <sup>[3]</sup>。

#### 2. 模型初始化

- 權重 w、偏置 b 隨機初始化<sup>[3]</sup> <sup>[4]</sup>。

#### 3. 訓練方法

- 最小二乘法：直接計算封閉解<sup>[3]</sup> <sup>[4]</sup>。
- 梯度下降法：反覆根據損失函數梯度更新參數（需設置 learning\_rate, n\_epochs）<sup>[1]</sup> <sup>[3]</sup> <sup>[4]</sup>。

#### 4. 預測與評估

- 使用訓練後的模型對新數據進行預測<sup>[1]</sup> <sup>[3]</sup>。
- 評估指標：均方根誤差（RMSE）、均方誤差（MSE）等<sup>[1]</sup> <sup>[3]</sup> <sup>[4]</sup>。

### 四、範例程式片段

```
# 訓練參數
learning_rate = 0.1
n_epochs = 100

# 梯度下降訓練
for epoch in range(n_epochs):
    # 計算預測值與損失
    # 更新 w, b
    pass
```

### 五、補充說明

- 若需支援多元特徵，需確保資料格式正確，並適當調整權重維度<sup>[2]</sup> <sup>[3]</sup>。
- 可選用正則化以提升泛化能力，常見如 L2（Ridge）、L1（Lasso）<sup>[4]</sup> <sup>[6]</sup>。
- 若有需求可加入交叉驗證、特徵選擇等進階功能<sup>[7]</sup> <sup>[6]</sup>。

參考文獻皆已在文中以 [#] 標註。

1. [https://blog.csdn.net/qq\\_37978800/article/details/115188018](https://blog.csdn.net/qq_37978800/article/details/115188018)
2. <https://ithelp.ithome.com.tw/articles/10347816>
3. [https://zh.d2l.ai/chapter\\_linear-networks/linear-regression-scratch.html](https://zh.d2l.ai/chapter_linear-networks/linear-regression-scratch.html)
4. [https://blog.csdn.net/qq\\_43045620/article/details/123079305](https://blog.csdn.net/qq_43045620/article/details/123079305)
5. <https://www.cnblogs.com/jawww/p/12297848.html>
6. <https://docs.azure.cn/zh-cn/machine-learning/component-reference/linear-regression?view=azureml-api-2>
7. <https://www.cnblogs.com/leezx/p/15719492.html>