

# Machine Learning Assignment 2

409410005 鍾天睿

## Execution description

我使用 `np.linspace` 產生等距的資料。如

```
x = np.linspace(0, 1, num=m)
y = np.sin(2 * np.pi * x) + np.random.normal(loc=0, scale=0.2, size=m)
```

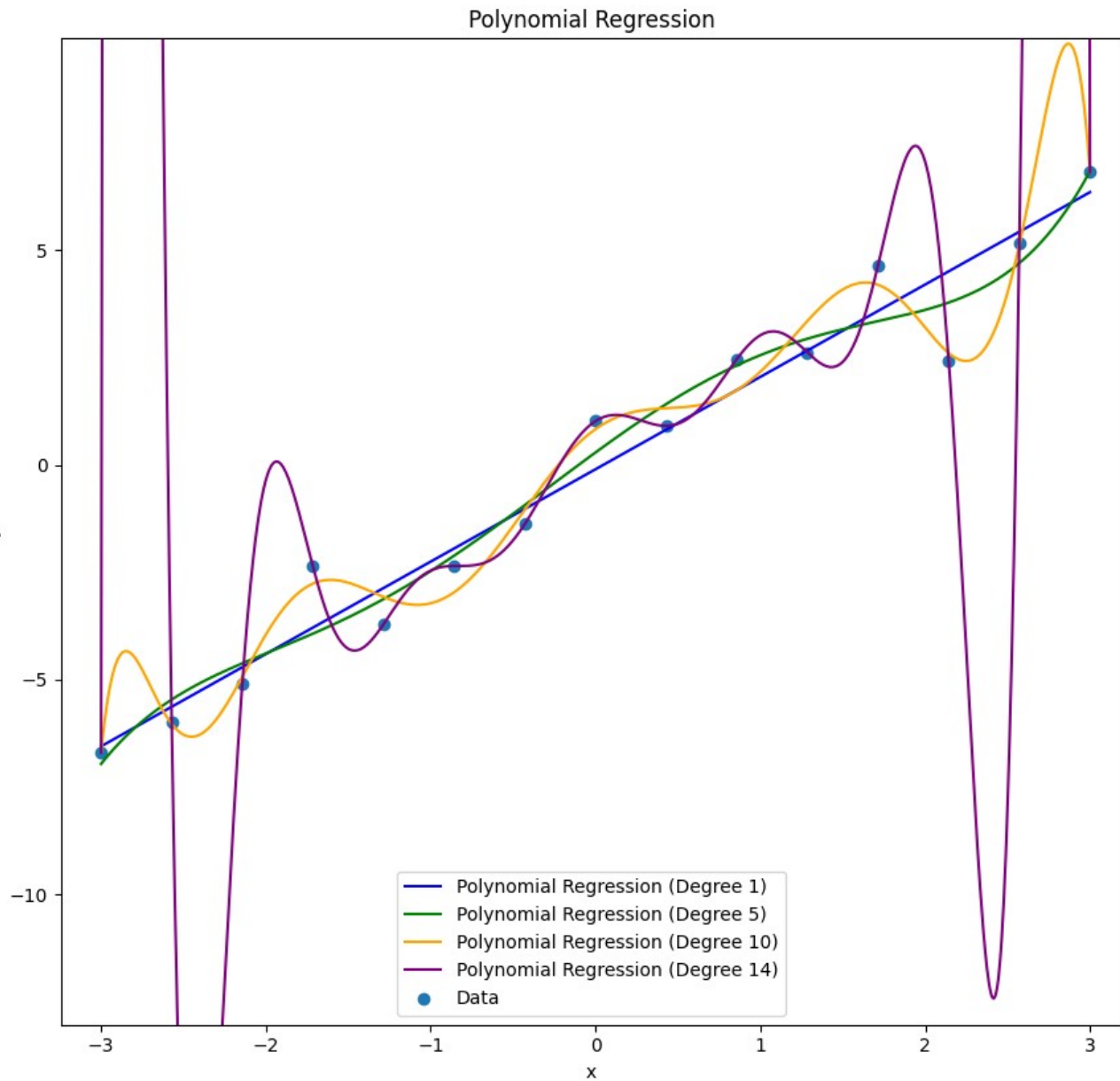
擬合的部份，我繼承了 `sk-learn` 的 `base class` 以便於使用如 `cross-val` 的工具，然後自己實做了擬合的程式。實做可以在 `my_model.py` 中找到。

畫圖的部份使用 `matplotlib`。繪製函數的時候使用 1000 個點繪製，並非指使用 `m` 個點，以更好的展示擬合的函數。

執行結果的部份由於可能沒設定好 `seed` 等原因，與程式實際執行有極小的落差，但是由於差異極小，不影響作業的意義，我就不花時間再貼上一次了。

# Experimental results

## 線性資料 (p1-p3)



### Linear Model

Training Error (MSE): 0.723

CV Error (MSE): 0.915

### Degree 5

Training Error (MSE): 0.550

CV Error (MSE): 2.454

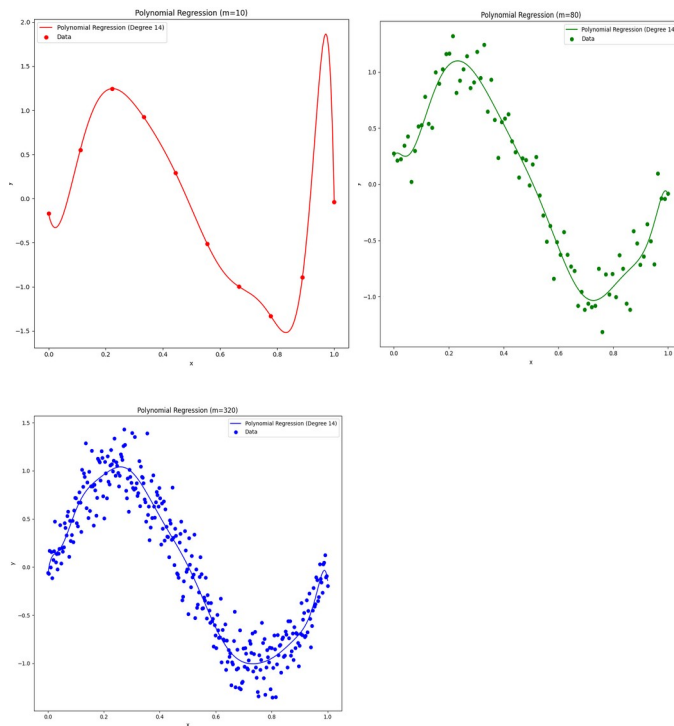
## Degree 10

Training Error (MSE): 0.172  
CV Error (MSE): 6800.166

## Degree 14

Training Error (MSE): 0.000  
CV Error (MSE): 100974690.738

## Sin 資料，15D，不同資料大小



## m=10

Training Error (MSE): 0.  
CV Error (MSE): 220.339

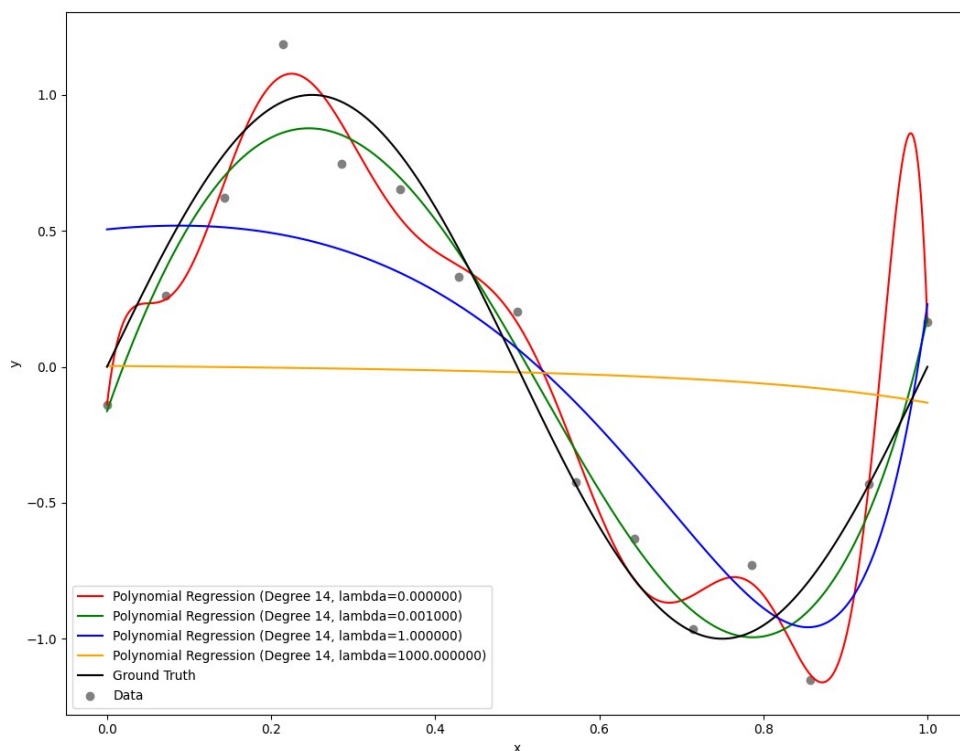
## m=80

Training Error (MSE): 0.030  
CV Error (MSE): 0.051

## m=320

Training Error (MSE): 0.046  
CV Error (MSE): 0.039

## Sin 資料+模型正規化



### Lambda = 0 (沒有正規化)

Training Error (MSE): 0.007  
CV Error (MSE): 1372.791

### Lambda = 0.001

Training Error (MSE): 0.022  
CV Error (MSE): 0.455

### Lambda = 1.

Training Error (MSE): 0.109  
CV Error (MSE): 0.212

### Lambda = 1000.

Training Error (MSE): 0.415  
CV Error (MSE): 0.400

## Conclusion

1. 參數量大的模型，可以更好的 fit 複雜的函數。
2. 當資料數量不變時，模型參數越多，越容易發生過度擬合。

3. 使用正規化技巧可以有效的改善過度擬合的問題。但是正規化的係數過高時，則會影響模型擬合的效果。
4. 相同大小模型，資料比數越多，越困難擬合模型，故 training loss 較高，但由於更不容易發生 over fitting，所以 valid loss 可能比較低。

## Discussion

1. 使用 scikit-learn 的 cross validate 工具時，要事先將資料 shuffle，否則資料按原始排序取樣，將嚴重影響驗證可信度。
2. 本作業程式碼部份參考 chatGPT、GitHub Copilot 產生的程式碼。自動產生的程式碼雖然看起來完美，但實際上可能暗藏嚴重 bug (如 1. 提到的問題)，仍然需要謹慎的檢查、調試代碼，以及閱讀文件。