# 機器學習報告

## 結果

## Iris data set

- 決策樹

  準確度：0.948 (取 10 次平均)

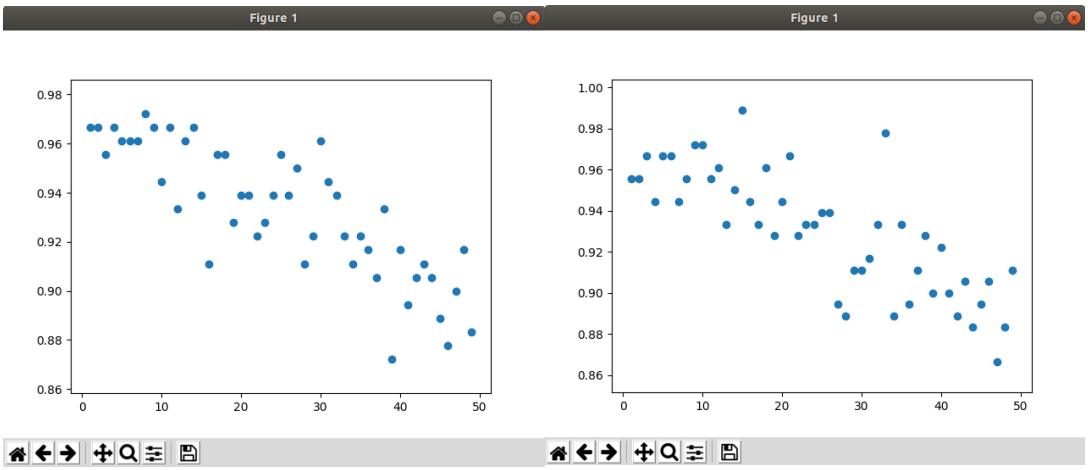
  設定：先用 PCA 降到 2 維，再建立決策樹。我沒有建立 random forest，因為我記得在第 1 份作業，也有用 random forest 分析 Iris data set，準確率沒有顯著差別

- kNN

  準確度：0.968 (取 10 次平均)

  設定：先用 PCA 降到 2 維，再尋找 5 個最近的點(k = 5)

  關於不同 k 值的效果(我只試 1~50，因為一個 target 只有 50 筆資料)




  我搜尋最佳 k 值的方法有把計算 4 次精準度取平均，然而隨機誤差太大了，很難比較最好的 k，所以我用 sklearn 默認的 k 值 k = 5

- Naive Bayes

  準確度：0.951 (取 10 次平均)

  設定：就只是 Naive Bayes，沒有 normalize，也沒有 PCA。因為 feature 是連續型，所以就用常態分佈 (Gaussian) 來估計 feature 值的分佈

- 比較：kNN 比 Naive Bayes 好，Naive Bayes 比決策樹好

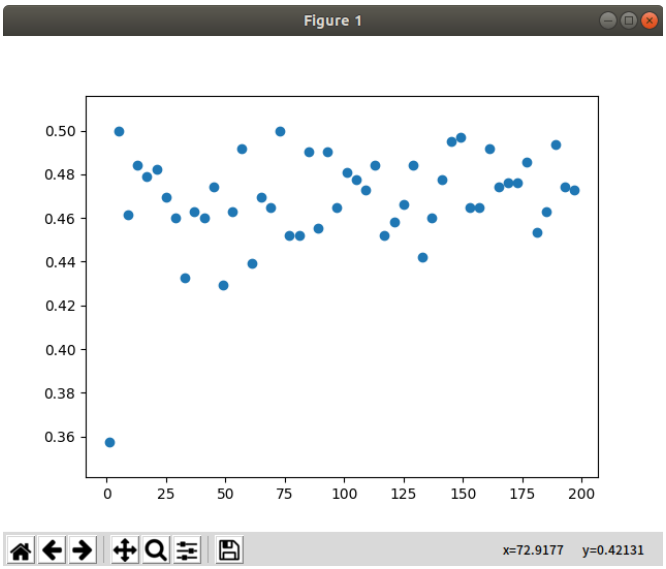
## forest fire data set

- 決策樹

  準確度：0.353 (取 10 次平均)

- kNN

  尋找適合的 k 值，我只嘗試 k=1+4n 的值

  我將 k 值設為 100，得到的精準度約 0.487 (取 10 次平均)

- Naive Bayes

  準確度：0.214 (取 10 次平均)

- 結果，Naive Bayes 非常的糟，推測是因為 feature 不夠像常態分佈，而我用的是 Gaussian Naive Bayes 導致。即使最好的 kNN，也不到 50% 精準度，這可能是因為 target 的分佈太不平均造成的。

# 使用函式庫

numpy、sklearn、matplotlib (畫圖)

# 使用語言

Python 3

# 程式解釋

我的程式可分為 `read.py`、`decision.py`、`knn.py`、`bayes.py` 和 `run.sh`。

- `read.py`：讀取 csv 檔案，然後就可以機器學習了！不過我有對資料做預處理
  - 對 Iris data set：把花的名稱轉成整數
  - 對 Forest Fires Data Set：丟棄前 4 個屬性(X 座標、Y 座標、月份、星期幾)，然後把火災範圍區分成 6 種 class (0, 0~1, 1~10, 10~100, 100~1000, 1000 以上)
- `decision.py`：用決策樹 (decision tree) 模型做預測。我使用的函式庫 sklearn 建立決策樹的方法是 CART

- `knn.py`：用 kNN (k nearest neighbor) 模型做預測，此外還可以尋找最好的 k 值並畫圖
- `bayes.py`：用 naive bayes 模型做預測
- `run.sh`：安裝程式需要用到的套件

# 資料來源

Iris data set: https://archive.ics.uci.edu/ml/datasets/Iris

Forest Fire Data Set: https://archive.ics.uci.edu/ml/datasets/Forest+Fires