



手搓算法， 從想像到實作

高三 AI 選修期末報告

陳俊智 黃麒翰

Database: BMI DB & IRIS DB

Jan, 8, 2025



內容概要

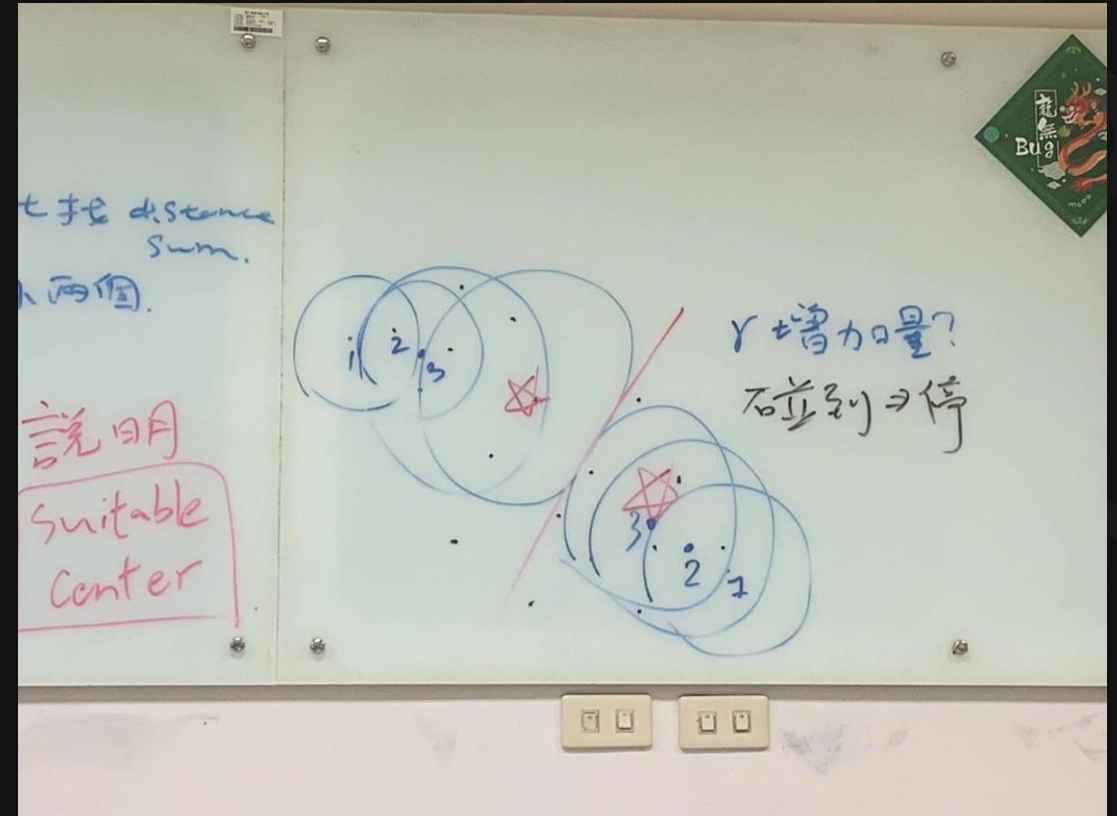
- > 1. 講述兩種自己想出的演算法 + 小規模測試
- > 2. 針對 2 種不同演算法的問題總結 (簡短)
- > 3. 資料集的介紹
- > 4. 針對資料集所做出的改變
- > 5. 兩種演算法運用在資料集上的結果
- > 6. 總結兩種演算法的優劣



算法的介紹-方法一

> 課堂上最初的想法

- > 1. 隨機取兩點
- > 2. 以兩點為圓心畫圓
- > 3. 圓的半徑不斷增加，直到兩圓相交
- > 4. 圓內的所有點座標取平均數，得到新的中心點
- > 5. 再以兩中心點繼續重複 2~4
- > 6. 得到最終的中心點
- > 7. 最後每個資料點較靠近某中心點的為一類







問題？

- 1.不知何時停止
- 2.程式撰寫困難

新的演算邏輯

- > 1. 隨機取兩點
- > 2. 以兩點畫兩圓，半徑為兩點的距離除以 2
- > 3. 圓內的所有點座標取平均數，得到新的中心點
- > 4. 不斷重複 2~3
- > 5. 重複到一個上限 (本次設定 50 次)
- > 6. 得到最終的中心點
- > 7. 最後每個資料點較靠近某中心點的為一類





小規模測試

結果



不穩定...



算法的介紹-方法二

- > 1. 隨機取兩點 A 與 B
- > 2. 找和 A 與 B 最接近的點 a 與 b
- > 3. A 和 a 座標取終點得 M，B 和 b 座標取終點得 N (N 和 M 為中心點)
- > 4. N 成為新的 A，M 成為新的 B
- > 5. 刪除點 A、B、a、b
- > 6. 重複執行 2~5，直到所有點用完
- > 7. 以最後得到的 N 與 M 作為最終的中心點，距離較近的點分為一組





小規模測試

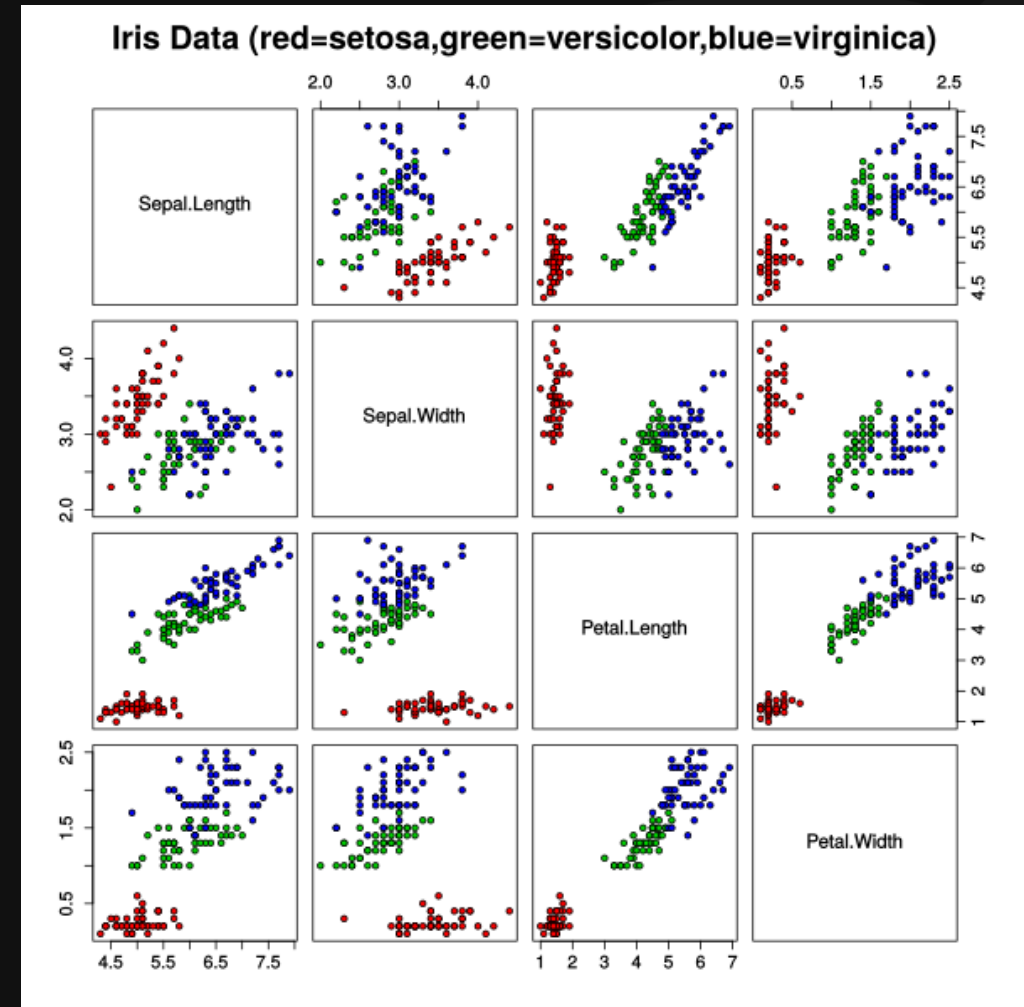
結果

1. 結果不穩定
2. 迭代次數會過多
3. 兩中心點有機率重疊

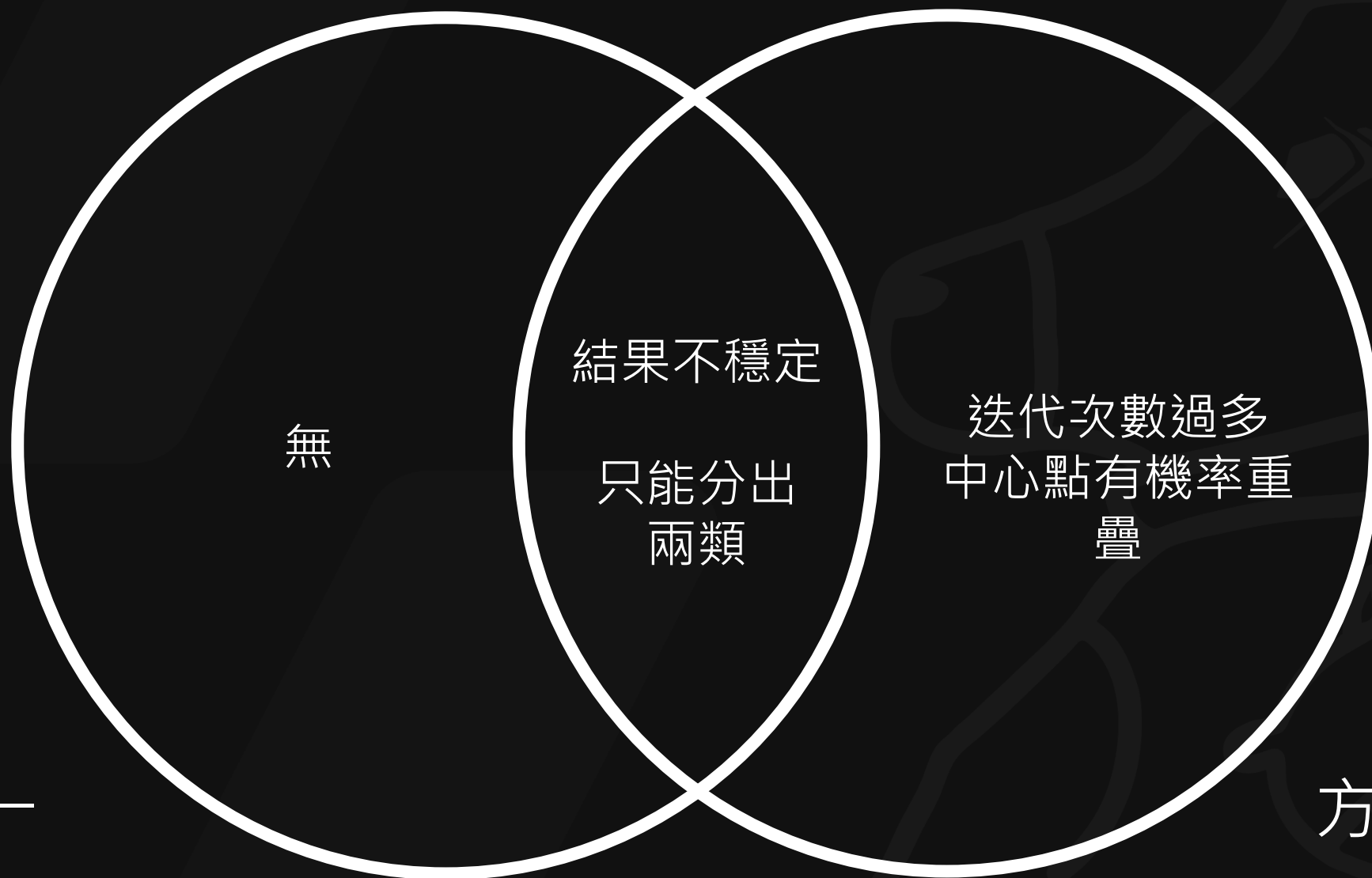


Database: IRIS

- 目標：分類不同的鳶尾花品種。
- 品種：三種不同的鳶尾花品種：
 - 山鳶尾 (Iris setosa)
 - 變色鳶尾 (Iris versicolor)
 - 維吉尼亞鳶尾 (Iris virginica)
- 特徵：四個特徵的測量值
 - 萼片長度 (Sepal Length)
 - 萼片寬度 (Sepal Width)
 - 花瓣長度 (Petal Length)
 - 花瓣寬度 (Petal Width)
- 樣本分佈：150 個樣本，每個品種各 50 個。



面臨問題



方法一

方法二

解決方法

分類結果不穩定

隨機取點 修掉
改為
由左下及右上取初始點

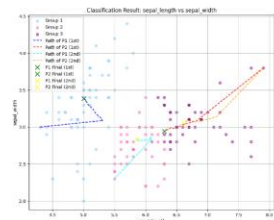
只能分兩類

分類完之後
再進行二次分類

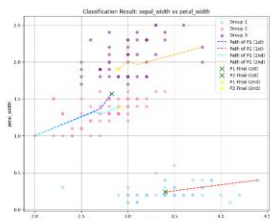
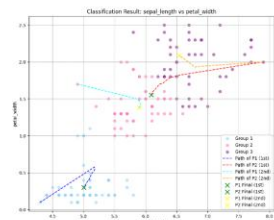
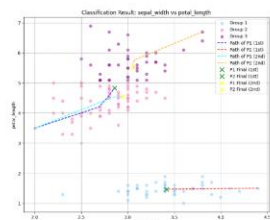
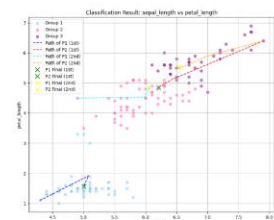
實際運作

方法一 ~對答案

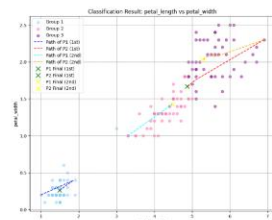
Sepal.Length



Sepal.Width

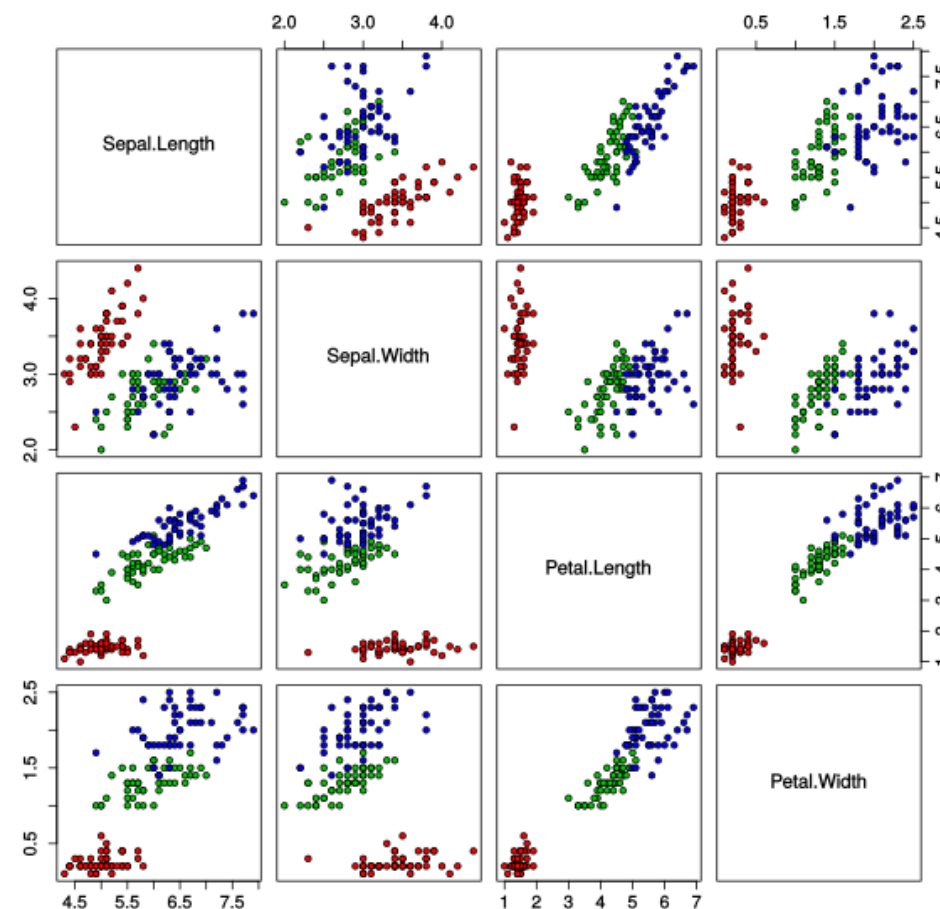


Petal.Length



Petal.Width

Iris Data (red=setosa,green=versicolor,blue=virginica)



方法一 ~成功率一覽

	SL vs SW	SL vs PL	SL vs PW	SW vs PL	SW vs PW	PL vs PW
Group_1	88%	94%	93%	98%	100%	98%
Group_2	71%	76%	78%	84%	92%	92%
Group_3	75%	92%	83%	98%	96%	96%
Average	78%	87.3%	84.6%	93.3%	96%	95.3%

> S : Sepal (花萼)

> P : Petal (花瓣)

> W : Width (寬度)

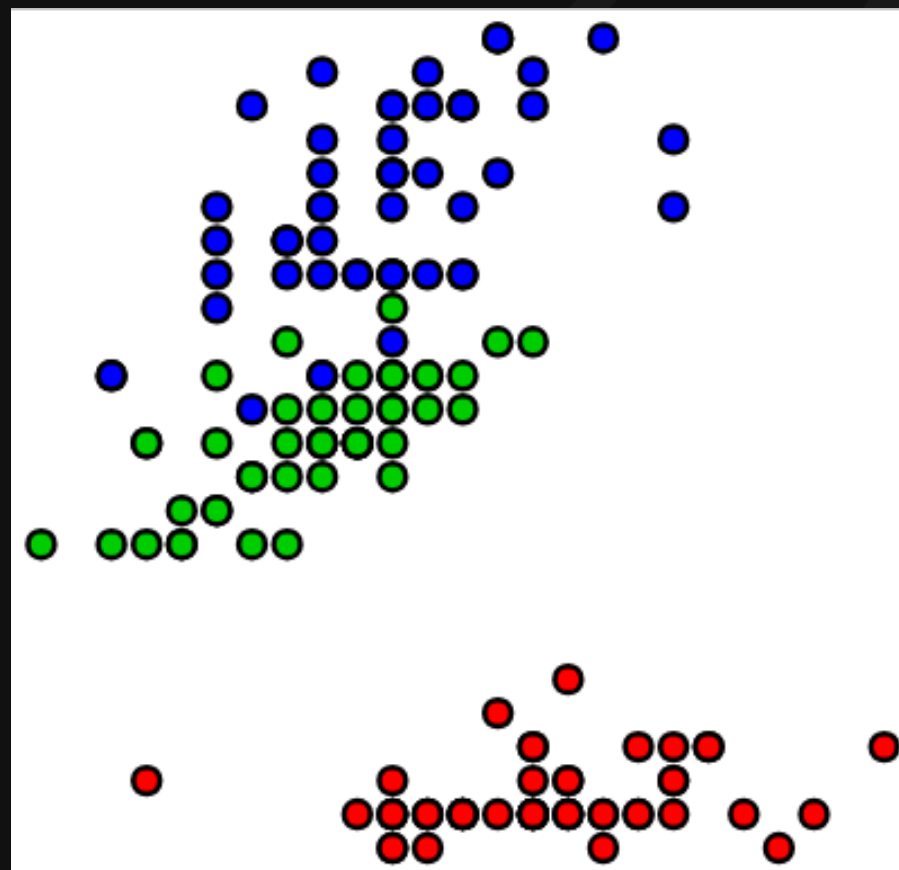
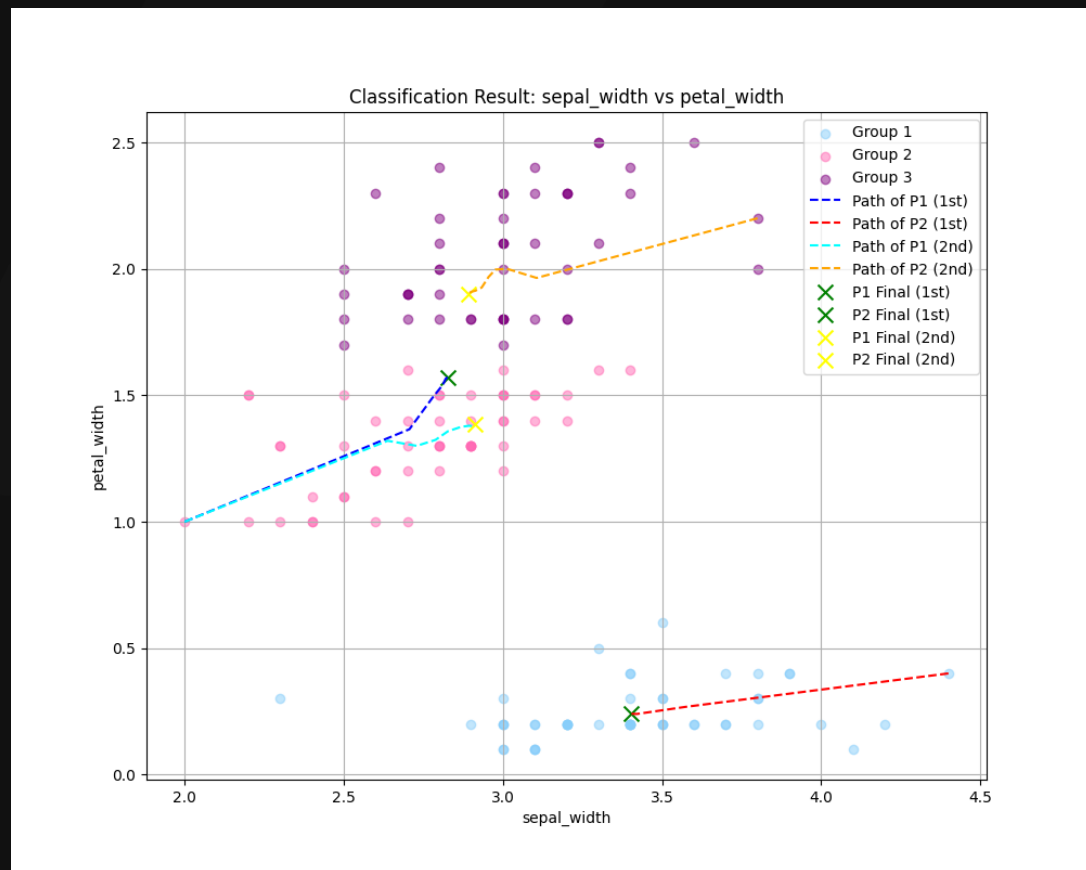
> L : Length (長度)

最好的一個分類

算法一

	Group1	Group2	Group3	Average
Rate	100%	92%	96%	96%

原始資料

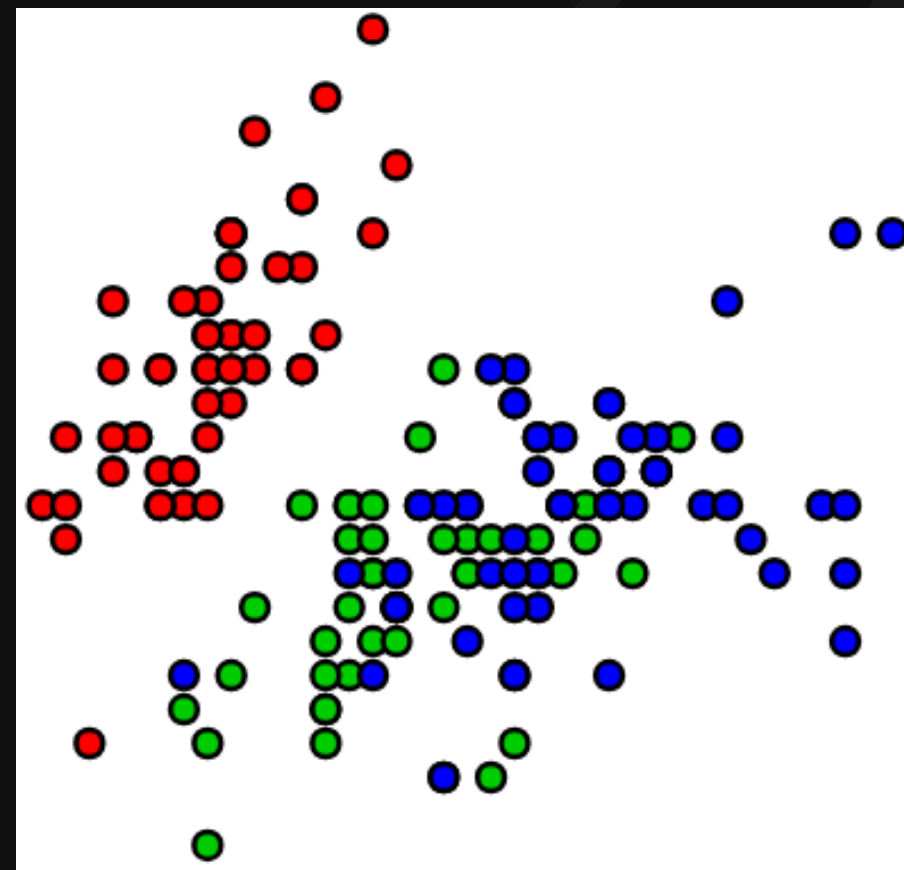
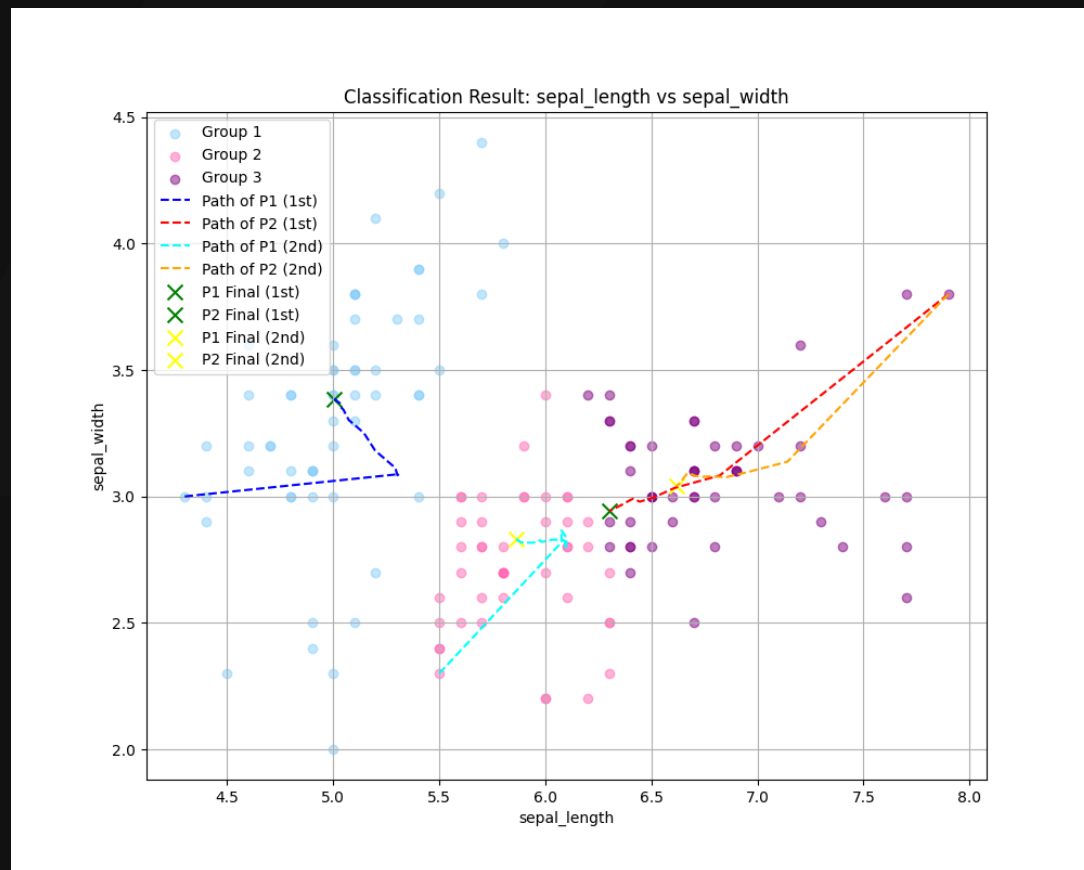


最爛的一個分類

算法一

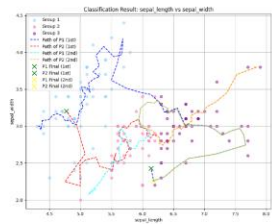
	Group1	Group2	Group3	Average
Rate	88%	71%	75%	78%

原始資料

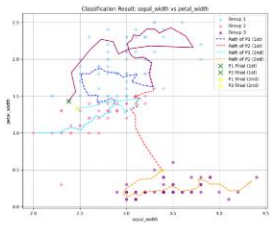
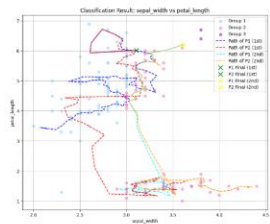
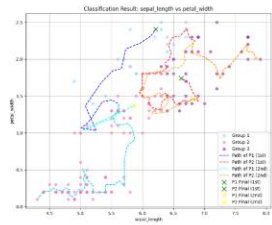
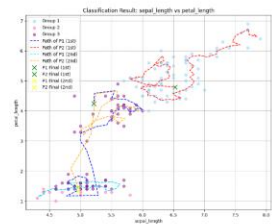


方法二 ~對答案

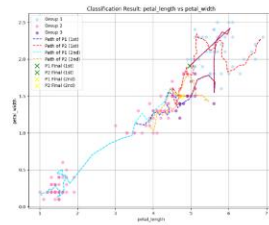
Sepal.Length



Sepal.Width

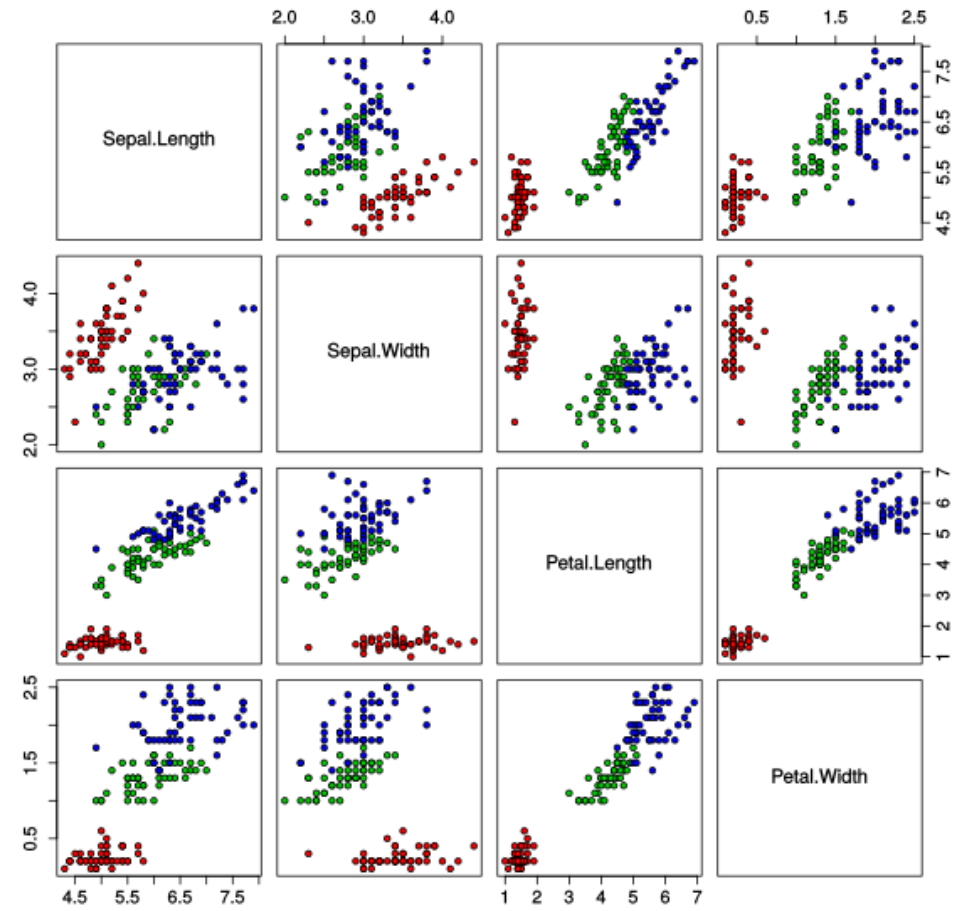


Petal.Length



Petal.Width

Iris Data (red=setosa,green=versicolor,blue=virginica)



方法二 ~成功率一覽

	SL vs SW	SL vs PL	SL vs PW	SW vs PL	SW vs PW	PL vs PW
Group_1	89%	67%	85%	51%	80%	100%
Group_2	71%	100%	53%	62%	95%	52%
Group_3	71%	58%	69%	100%	100%	69%
Average	77%	75%	69%	71%	91.6%	73.6%

> S : Sepal (花萼)

> P : Petal (花瓣)

> W : Width (寬度)

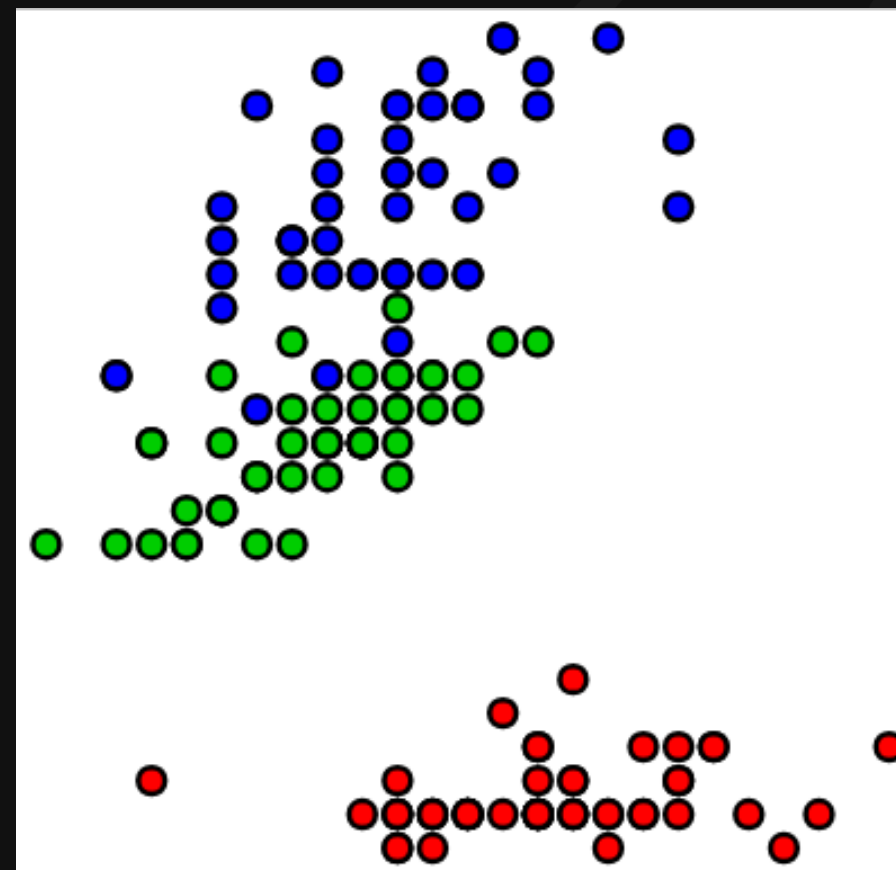
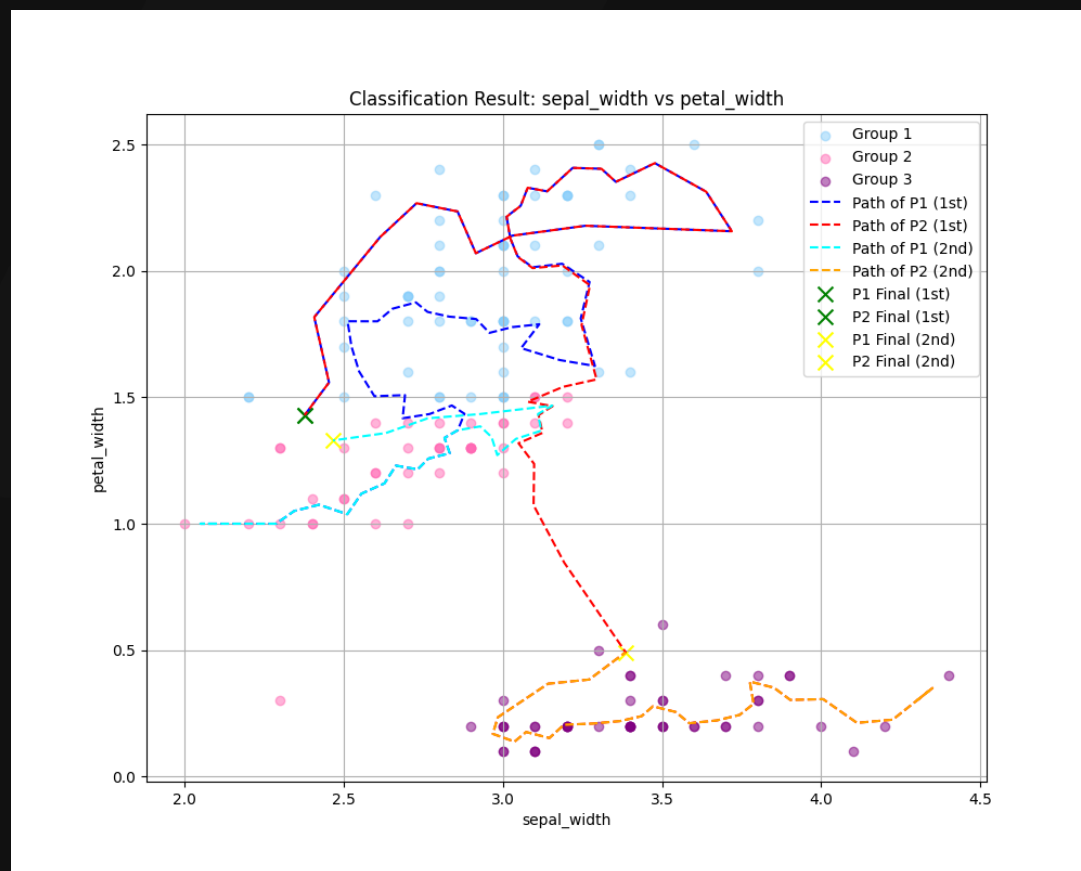
> L : Length (長度)

最好的一個分類

算法二

	Group1	Group2	Group3	Average
Rate	80%	95%	100%	91.6%

原始資料

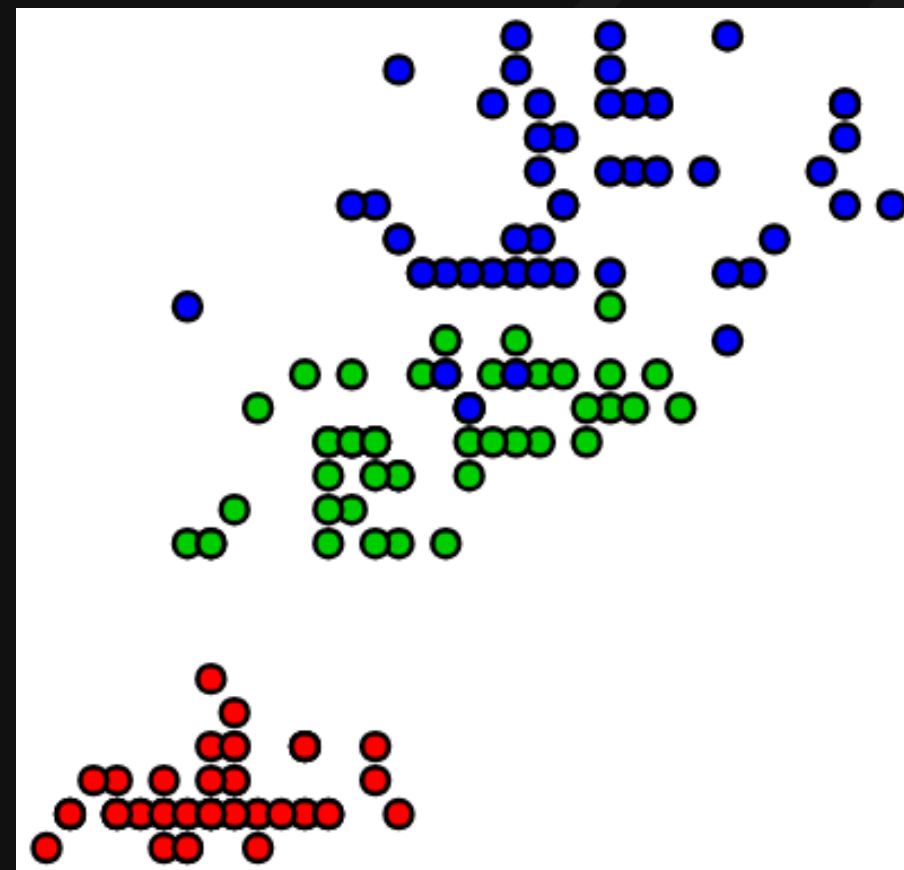
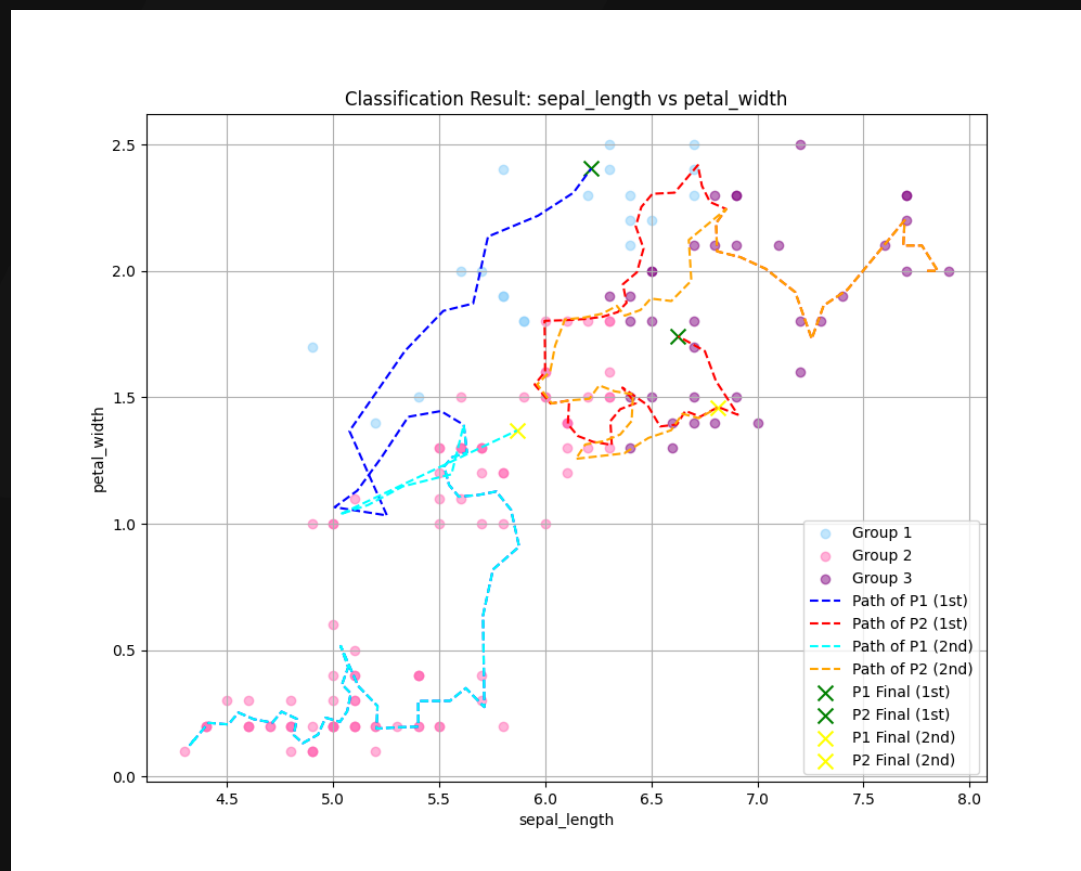


最爛的一個分類

算法二

	Group1	Group2	Group3	Average
Rate	85%	53%	69%	69%

原始資料



結果討論

優劣比較

方法一比較好 (完勝)



	SL vs SW	SL vs PL	SL vs PW	SW vs PL	SW vs PW	PL vs PW
Method_1	78%	87.3%	84.6%	93.3%	96%	95.3%
Method_2	77%	75%	69%	71%	91.6%	73.6%
Method (1-2)	+1%	+12.3%	+15.6%	+22.3%	+5.4%	+21.7%



原因推測

影響分類成效的因素

方法一

資料的分散程度

方法二

資料的數量
資料的分散程度
剛開始的取點位置



方法一真的好嗎？

影響方法一成效的其他因素

剛開始的取點位置

圓的半徑

未來的改良方向

- > 1. 圓的半徑可以考慮在迭代的過程中做變化
- > 2. 多次的隨機取點 (With 隨機森林)





報告結束