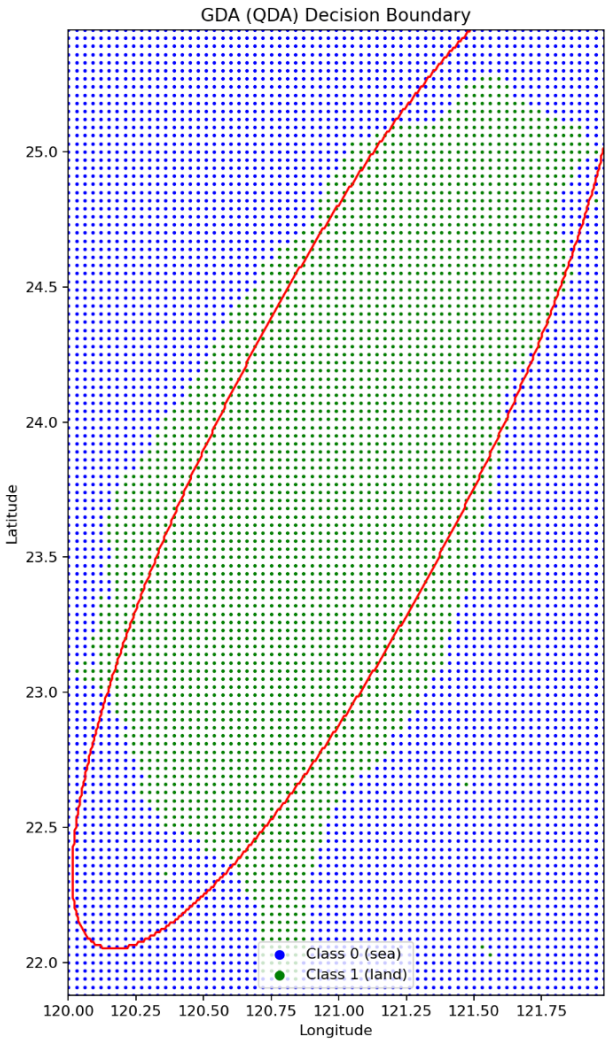
蔡翔宇的 HW6 的程式報告

1 先來看我們程式跑的成果：



2 原本在 HW4 我們使用的是 LDA 的方法，他的 Decision boundary 會事一條直線，但我們現在在一個不規則的圖形上（也就是台灣），要區分陸地與海洋的溫度不應該使用一刀切的方法，故我們在 HW6 中使用 GDA 讓我們的 Decision boundary 變成一個二次曲線，使我們更好的來判斷台灣的陸地與海洋。

3 而在這次的作業中我先是使用了兩個模型：

3.1 C(x)：分類模型 → 採用 GDA/QDA (Quadratic Discriminant Analysis)，將地點分為海(label 0) 與「陸地」(label 1)。這個模型能捕捉不同類別間的共變異矩陣差異，因此決策邊界為橢圓形。

3.2 R(x)：迴歸模型 → 針對陸地樣本（label = 1）擬合多項式 ridge regression，用經緯度預測地表溫度。

3.3 最後依照題目給定的 piecewise 定義，建構：

$$h(\mathbf{x}) = \begin{cases} R(\mathbf{x}), & \text{if } C(\mathbf{x}) = 1 \\ -999, & \text{if } C(\mathbf{x}) = 0 \end{cases}$$

這樣我們就可以在海洋區域回傳 -999 的值，並且在陸地區域輸出預測溫度，也形成分段平滑函數!而實際操作流程如下：將分類模型 $C(\mathbf{x})$ 與迴歸模型 $R(\mathbf{x})$ 結合，建立一個分段函數 $h(\mathbf{x})$，在陸地區域輸出平滑的溫度預測，在海洋區域則輸出 -999。而我的實作流程如下：

4 資料預處理（Data Preprocessing）

- 4.1 讀取中央氣象局的格點資料（XML）。資料中 -999 代表海洋無測站區，其他值則為有效的地表溫度測量。
- 4.2 將資料轉換成 DataFrame，欄位包含經度lon、緯度lat、溫度value。
- 4.3 根據value == -999將資料二分類：海洋點標記為 label = 0，陸地點標記為 label = 1。這部分資料會用於後續的分類模型訓練。

5 分類模型 C(x):Quadratic Discriminant Analysis（QDA）/GDA

- 5.1 為了讓模型能學習到真實的邊界，我使用 DA，而非 LDA 或 logistic regression。
- 5.2 GDA 允許不同類別（海 vs 陸）具有不同的共變異矩陣，因此決策邊界為二次曲線（橢圓形），非常適合台灣海岸線這種彎曲、不規則的邊界。
- 5.3 我將lon、lat標準化後，用 GDA 訓練分類器，最後在整個經緯度網格上預測每個點的類別，得到一張海／陸分類圖。

6 迴歸模型 R(x) :Polynomial Ridge Regression

- 6.1 迴歸模型只使用陸地資料（label = 1）。
- 6.2 以lon、lat作為輸入特徵，對溫度進行多項式擴展，再用 ridge regression 做擬合，以避免過擬合(overfitting)發生。
- 6.3 再來我們使用 cross-validation 選擇最佳的多項式次數與 ridge 參數 λ，最終得到一個平滑的溫度預測函數。

7 接下來開始組合函數 h(x)

- 7.1 對於每一個網格點$\mathbf{x}$：
  - 7.1.1 若分類模型$C(\mathbf{x}) = 1$，代表在陸地上，則輸出$R(\mathbf{x})$的預測溫度。
  - 7.1.2 若$C(\mathbf{x}) = 0$，代表海洋，則直接輸出 -999。
- 7.2 這樣就完成了一個分段平滑函數，能夠在整張網格上同時處理海洋與陸地。

8 結果輸出與評估

- 8.1 對 hold-out 資料計算 RMSE，檢驗迴歸準確度。
- 8.2 也對分類資料計算 accuracy，確保邊界學習效果如何。

8.3 將預測值畫在台灣網格地圖上做視覺化，方便比對真實地形與分類結果！

9 上圖顯示我們的分類模型在整個台灣網格上的預測結果：

9.1 藍色點：GDA 判定為海洋（label = 0）

9.2 綠色點：GDA 判定為陸地（label = 1）

9.3 紅色曲線：GDA 的二次決策邊界，代表海陸分界線

這張上圖清楚展示了 GDA 能學習出接近真實海岸線的橢圓形邊界，而我們的準確率約為 83.5%。

（以上不懂多詢問 ChatGPT ，但結果與報告都是自己理解完成做完的）