ML HW4

Date: 09/24/2025
Week: 4
Author: Alvin B. Lin
Student ID: 112652040

Problem 1: Programming assignment

Please first download the dataset: O-A0038-003.xml. This dataset comes from the Central Weather Bureau's observation platform: "Temperature Distribution – Hourly Temperature Observation Analysis Gridded Data".

Dataset Description:

- Each grid point represents a temperature observation value (°C).
- Invalid value: -999.
- Resolution of latitude/longitude: 0.03 degrees in both directions.
- The lower-left corner grid point coordinate:
  - Longitude: 120.00°E
  - Latitude: 21.88°N
- Data layout:
  - Longitude increases first (67 values in each row), then latitude increases (total 120 rows).
  - Therefore, the data forms a 67 $\times$ 120 numerical grid.

Questions

(1) Data Transformation

Convert the original dataset into two supervised learning datasets:

(a) Classification Dataset

Format: (`Longitude, Latitude, Label`)

- If the temperature value = -999 (invalid), then Label = 0.
- If the temperature value is valid, then Label = 1.

(b) Regression Dataset

Format: (`Longitude, Latitude, Value`)

- Only keep valid temperature values (remove all -999).
- Value = corresponding temperature (°C).

(2) Model Training

Using the two datasets from (1), train one simple machine learning model each:

- Classification Model:

  Use (Longitude, Latitude) to predict whether a grid point is valid (0 or 1).
- Regression Model:

  Use (Longitude, Latitude) to predict the corresponding temperature value.

Project:

(1) The data frames is made at **section 2B** in the code, here are the head aspect of them:

```
--- 2B. Generating Pandas DataFrames (Label 0/1) ---

Classification Dataset (Head):
   Longitude  Latitude  Label
0     120.00     21.88      0
1     120.03     21.88      0
2     120.06     21.88      0
3     120.09     21.88      0
4     120.12     21.88      0
Classification Dataset Shape: (8040, 3)

Regression Dataset (Head):
   Longitude  Latitude  Value
0     120.84     21.94   28.1
1     120.72     21.97   28.6
2     120.75     21.97   28.6
3     120.78     21.97   27.8
4     120.81     21.97   26.5
Regression Dataset Shape: (3495, 3)
```

(2) (a) For the categorical part, we need some **Basic setups:**

- The training set and the testing set compose 80% and 20% of the dataset.
- Two hypothesis functions are assumed:

  $$h_1(x, y) = ax^2 + by^2 + cxy + dx + ey + f$$

  and one with logistic regression,

  $$h_2(x, y) = \sigma(h_1(x, y)) = \sigma(ax^2 + by^2 + cxy + dx + ey + f).$$

  We aim to compare which do better.
- For the hypothesis without logistic regression, we use **SVM (LinearSVC)** method to find the coefficients; while we use **liblinear** solver to solve the one with logistic regression. Both have `max_iter = 5000`.
- To train a single function is regarded difficult, so I separate the whole dataset into four regions, with boundaries `LON_SPLIT_1` = 121.50, `LON_SPLIT_2` = 120.77 and `LAT_SPLIT` = 24.0. For which:

$$
(\text{lat}, \text{lon}) \in \text{Region} \begin{cases} 1, & \text{lat} \in [120.00,\ 121.50), \text{lon} \in (24.00,\ 25.45] \\ 2, & \text{lat} \in [121.50,\ 121.98], \text{lon} \in (24.00,\ 25.45] \\ 3 & \text{lat} \in [120.00,\ 120.77), \text{lon} \in [21.88,\ 24.00] \\ 4 & \text{lat} \in [120.77,\ 121.98], \text{lon} \in [21.88,\ 24.00] \end{cases}
$$

And in each region, we train an independent quadratic curve.

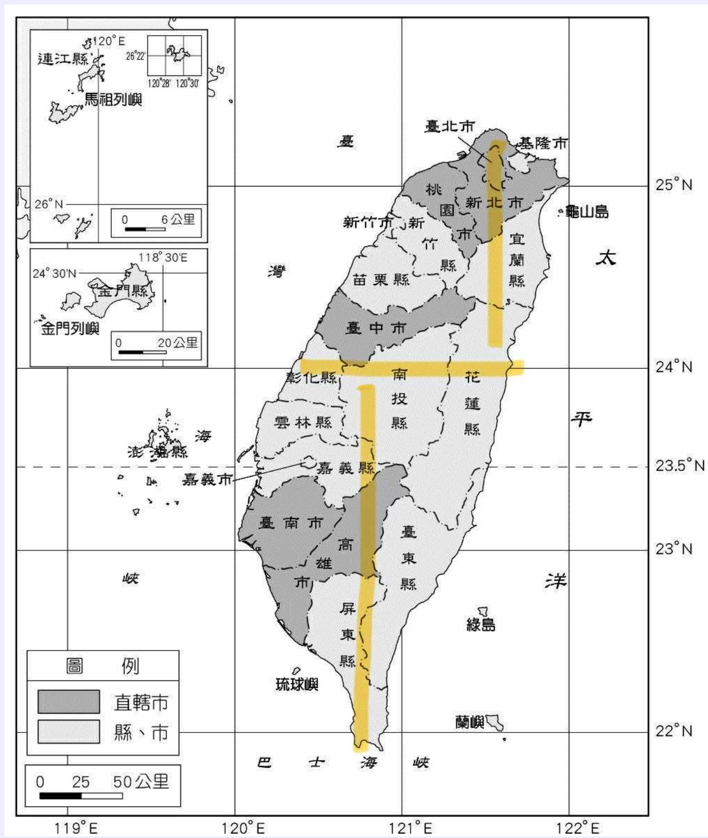


Figure 1: Illustration of region grouping

Results

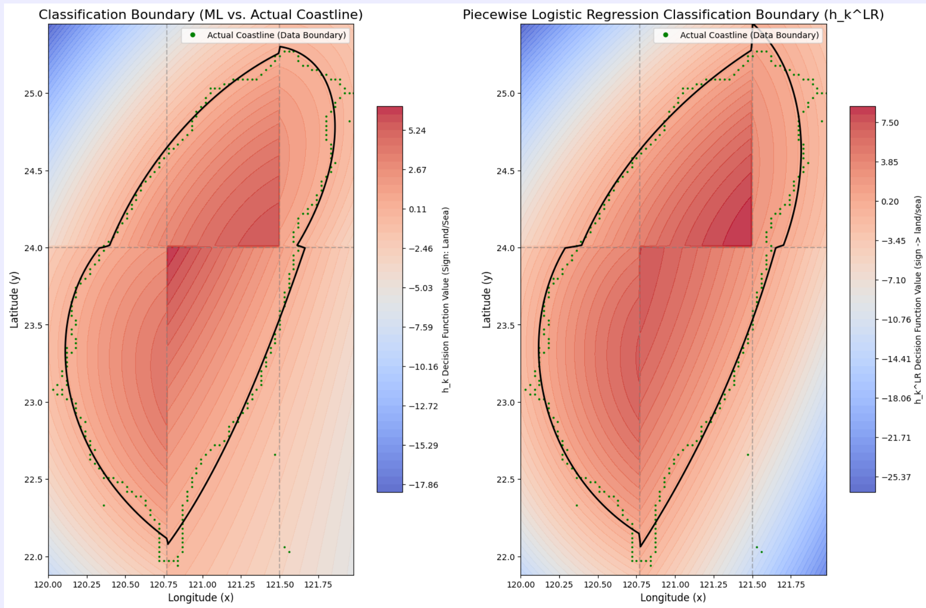
Both methods seem to have a very good categorical result, the frading is as follows:

```
--- Classification Test Results (h^LR vs. h^SVC) ---
1. Piecewise Logistic Regression (h^LR) Accuracy: 0.9596
2. Piecewise LinearSVC (h^SVC) Accuracy: 0.9714
```

The `accuracy_score` is given in `sklearn` for which compares how many data is in right position. The score shows that the hypothesis function **without logistic regression** performs slightly better.

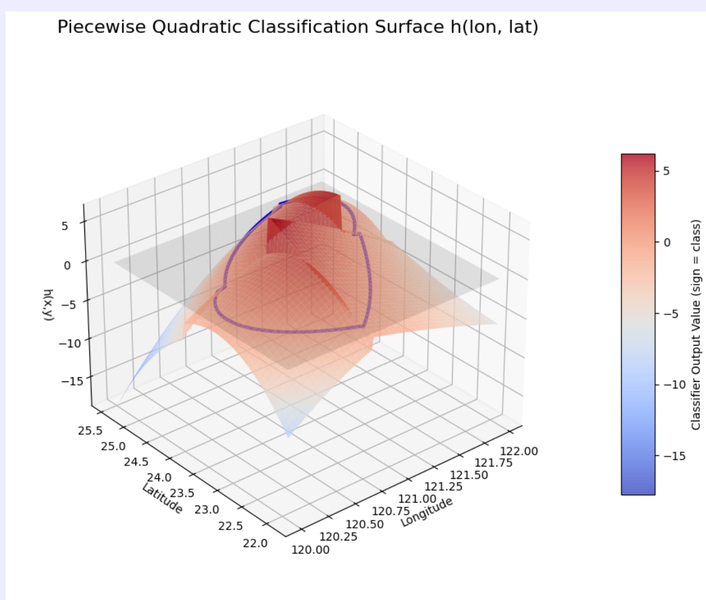
Visualisation

I've plot the boundary of the intersection with $z = 0$, or $h(\mathrm{lon}, \mathrm{lat}) = 0$ and compare to the actual coastline of the mainland Taiwan (dotted line), we get the figure like this:



Inside the contour, we will flag the label 1 because it is on the land; in contrast, 0 when outside the contour, because it is determined in the ocean area, for which the temperature is not measurable.

For the left one is the **non-logistic-regression** one while the right one is **logistic-regressioned**. Both have done a good job predicting the **label**, the contours almost fit the coastline. A 3D version result can also be seen below.



(b) In this part, we also need some setups:

- The training set and the testing set compose 80% and 20% of the dataset.
- This time hypothesis function has **only one**, in order to ensure the **smoothness** of the temperature.
- The hypothesis function is assumed cubic and

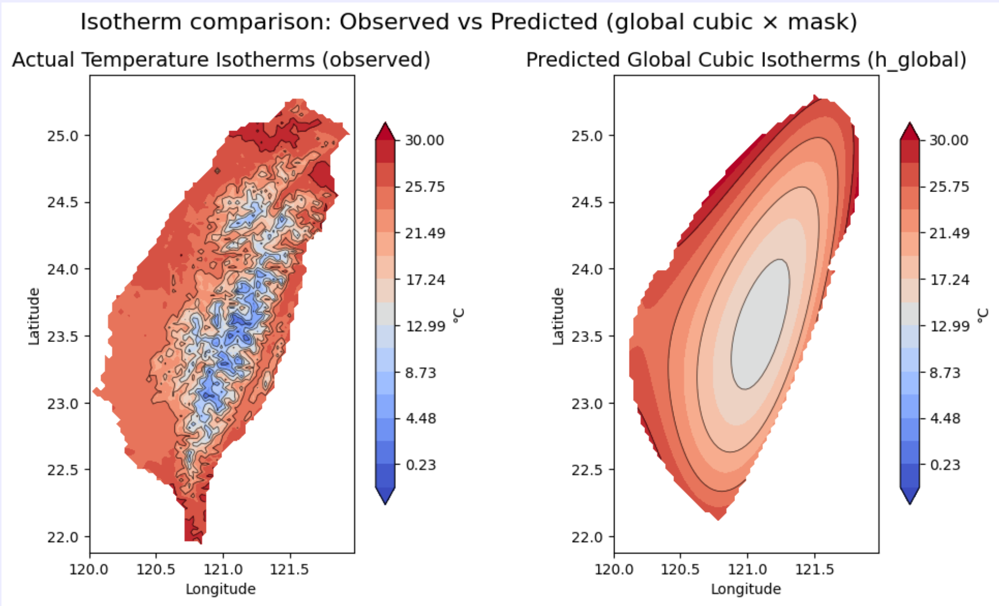
$$g_{\text{global}}(x, y) = ax^3 + by^3 + cx^2y + dxy^2 + ex^2 + fy^2 + hxy + jx + ky + l$$

- We use **SVM (LinearSVC)** method to find the coefficients, and the `max_iter` is set 5000.

Result

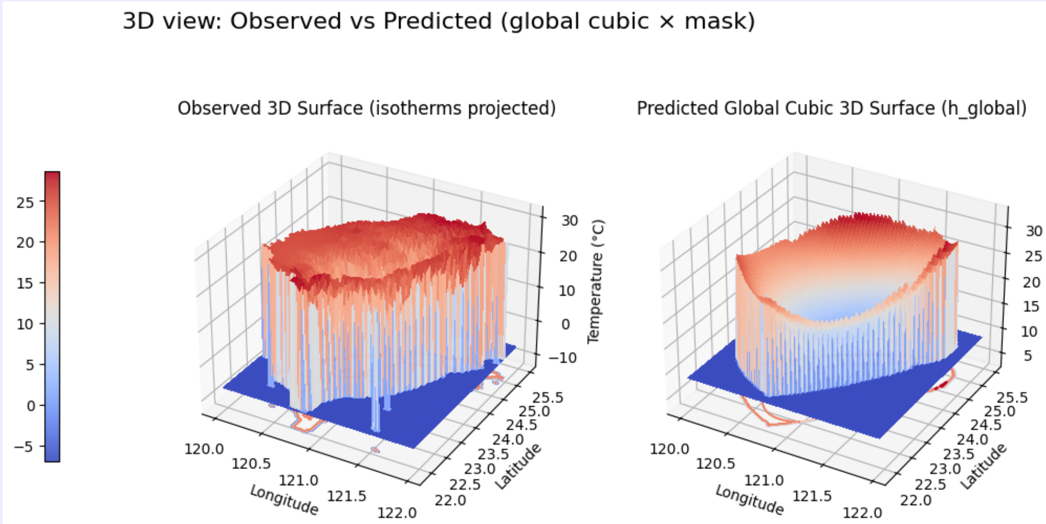
```
--- 8. Training Global Cubic Regression Model (g_global) ---
Global Cubic Regression trained with 2796 points.
Global Test RMSE for g_global: 4.4450 °C
```

The measured **root mean square error (RMSE)** is 4.4450°C, which is a little bit high; the result shows that maybe degree 3 polynomial is still not enough to fit the temperature distribution. **Visualisation** For this part, we show the **isotherm** in 2D and 3D to compare with the real data:



The 2D isotherms shows that the real-world temperature distribution is fairly **complex**, especially the mountain part. Our hypothesis function is too smooth so that we cannot do a good prediction on temperature.

I'm thinking of whether using $\sin\cdot$, $\cos\cdot$ is a better option, even though the trig functions have more **drastic volatility**, but they often occurs at the **"cliff"** of the function, seems still not fitting the distribution pattern.



The 3D version shows that not only we perform bad at the mountain part, the boundary/edge of the island has bad performance too.

Ways to improve

We can see that, although we have trained well on the categorical part, it still have some distance to the contour, maybe next time we can try to separate into more regions or using higher dimensional polynomial to obtain the better results.

As for the temperature part, I think our method is not optimal, our model without logistic regresion is somehow "not nonlinear" enough, making the temperature function too smooth, fail to predict.

My Question 1: Function Selection

In the report part, we see that temperature near the mountain area seems to have drastic volatilities, but at the boundary, the distribution is smooth. Is there a good candidate for hypothesis function? Is $x \sin \frac{1}{x};\ x \to 0$ an option?