ML Week6 Assignment

WANG TZU YI

1 Gaussian Discriminant Analysis (GDA)

1.1 Model Description

Gaussian Discriminant Analysis (GDA) is a generative classification model that assumes the data in each class follows a multivariate Gaussian distribution. Specifically, for each class label $y \in \{0, 1\}$, we model the conditional distribution of the features as:

$$p(\vec{x}|y = i) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\vec{x} - \mu_i)^T \Sigma^{-1}(\vec{x} - \mu_i)\right)$$

where μ_i is the mean vector of class i, and Σ is the shared covariance matrix.

Using Bayes' theorem, the posterior probability is:

$$p(y = 1|\vec{x}) = \frac{p(\vec{x}|y = 1)p(y = 1)}{p(\vec{x}|y = 0)p(y = 0) + p(\vec{x}|y = 1)p(y = 1)}$$

A new sample $\vec{x}$ is classified as:

$$C(\vec{x}) = \begin{cases} 1, & \text{if } p(y = 1|\vec{x}) > 0.5 \\ 0, & \text{otherwise} \end{cases}$$

1.2 Why GDA Works for This Dataset

This dataset exhibits two classes that can be reasonably separated using Gaussian-shaped distributions in feature space. Because GDA models the full covariance of features, it can capture elliptical decision boundaries that better reflect the underlying data structure than linear models such as Logistic Regression.

2 Training and Evaluation

2.1 Training Procedure

The dataset was split into training (80%) and testing (20%) subsets. The parameters (μ_0, μ_1, Σ) were estimated via Maximum Likelihood Estimation (MLE):

$$\hat{\mu}_i = \frac{1}{N_i} \sum_{j:y_j=i} \vec{x}_j, \quad \hat{\Sigma} = \frac{1}{N} \sum_{i=1}^{N} (\vec{x}_i - \hat{\mu}_{y_i})(\vec{x}_i - \hat{\mu}_{y_i})^T$$

2.2 Performance Metric

We evaluated model performance using accuracy on the test set:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Test Samples}}$$

The mean test accuracy achieved was 82.52%.

2.3 Decision Boundary Visualization

The decision boundary of the GDA model is given by the set of points $\vec{x}$ where $p(y = 1|\vec{x}) = 0.5$. Figure **??** shows the decision regions in 2D feature space, where two areas correspond to different classes.

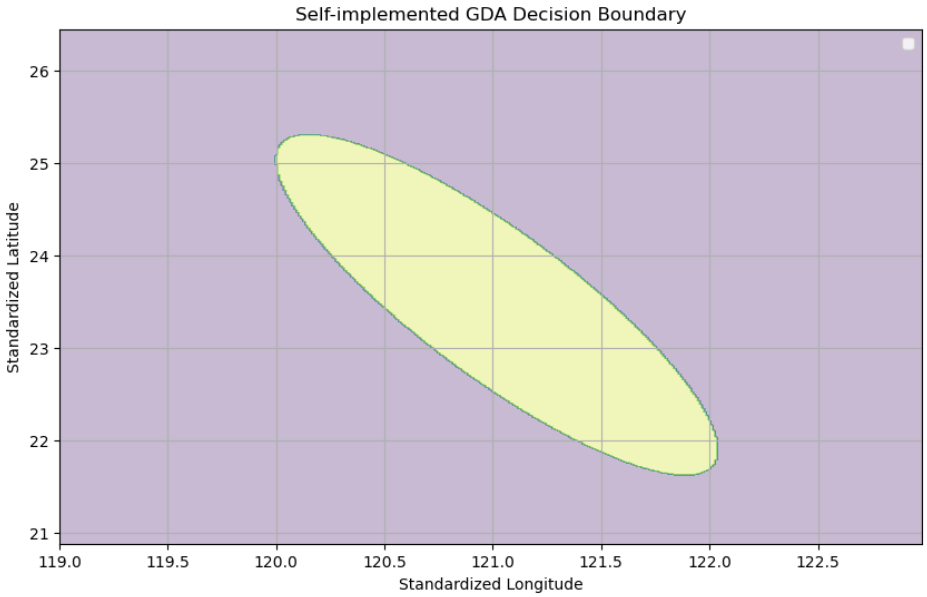

Figure 1: Decision Boundary of the GDA Classifier.

3 Combined Piecewise Regression Function

3.1 Model Definition

We define the combined model as:

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

where $C(\vec{x})$ is the GDA classifier and $R(\vec{x})$ is the regression model trained to predict a continuous variable (e.g., temperature).

3.2 Implementation

In Python, this model was implemented as follows:

```
class CombinedModel:
    def __init__(self, classifier, regressor):
        self.classifier = classifier
        self.regressor = regressor

    def predict(self, X):
        c = self.classifier.predict(X)
        r = self.regressor.predict(X)
        return np.where(c == 1, r, -999)
```

3.3 Model Behavior and Visualization

To visualize the combined model's behavior, we plotted the predicted outputs in geographical space. Points where $C(\vec{x}) = 0$ are colored in black (assigned -999), and continuous regression predictions $R(\vec{x})$ are shown with a color gradient (red–blue).

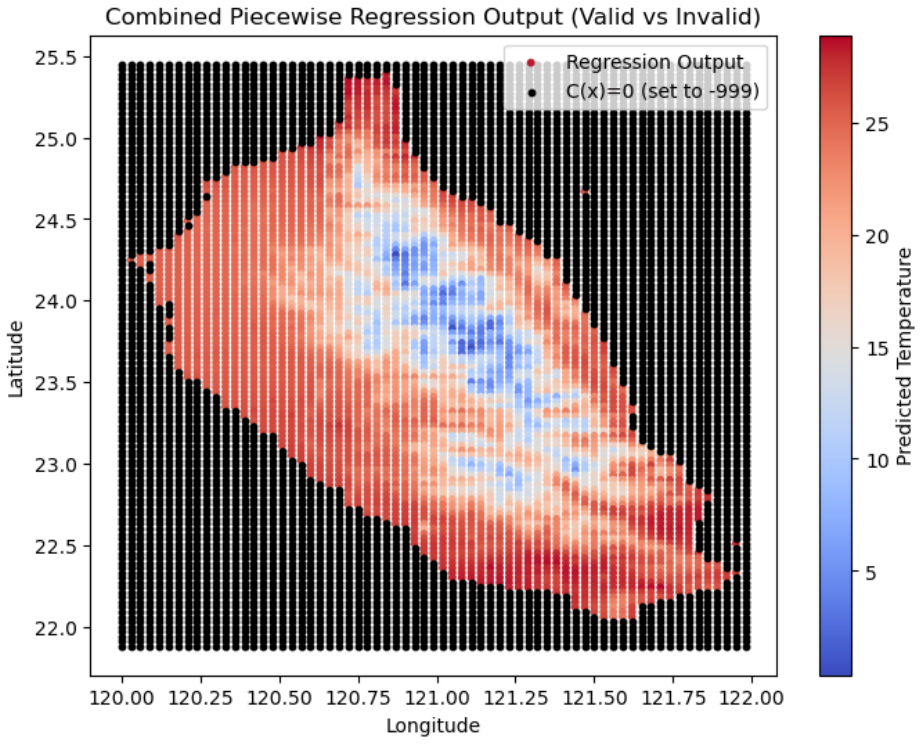

Figure 2: Combined Piecewise Regression Output. Valid regions are color-coded; invalid regions ($C(\vec{x}) = 0$) are black.

3.4 Discussion

This piecewise definition ensures the model behaves smoothly in valid regions, while clearly separating areas deemed invalid by the classifier. It is particularly useful when regression predictions should only be trusted in specific data regimes.

4 Conclusion

The GDA classifier successfully modeled the probabilistic structure of the dataset, achieving high accuracy and interpretable decision boundaries. The combined function $h(\vec{x})$ effectively integrates both classification and regression components, yielding a piecewise smooth output suitable for real-world applications involving conditional predictions.