Assignment 5: Non-Linear Logistic Regression and Probabilistic Surfaces

December 2025

Objective

The goal of this assignment is to extend the concept of Logistic Regression to non-linear datasets. You will utilize feature engineering to map a 2D circular boundary into a higher-dimensional linear space and visualize the resulting classification probability as a 3D Sigmoid surface.

Dataset

Use the file dataset.csv. This dataset contains 1,000 observations with two features (x, y) and a binary `label` (0 or 1). The labels are generated with a circular boundary and include significant Gaussian noise to create class overlap.

1 Part A: 2D Data Visualization and Hypothesis

1. **Visualization:** Load the dataset and generate a 2D scatter plot. Color the points according to their `label`.
2. **Analysis:** Observe the distribution of the two classes.
   - Can a standard linear boundary of the form $z = \beta_0 + \beta_1 x + \beta_2 y$ successfully separate these classes? Justify your answer.
   - Based on the visual shape of the separation, hypothesize a mathematical form for the logit z that could represent a closed-loop boundary.

2 Part B: Feature Engineering and Model Fitting

Since the classes are separated by a circle centered near $(2, 2)$, we must transform our input space.

1. **Feature Expansion:** Create a new feature set including quadratic terms. Your model equation for the logit z should be:

$$z = \beta_0 + \beta_1 x + \beta_2 y + \beta_3 x^2 + \beta_4 y^2$$

2. **Optimization:** Implement the Log-Loss function and use `scipy.optimize.minimize` to find the optimal parameter vector $\beta = [\beta_0, \beta_1, \beta_2, \beta_3, \beta_4]$.
3. **Boundary Extraction:** The decision boundary is defined by the set of points where $P(y = 1|x, y) = 0.5$. Using your optimized weights, state the final equation of the resulting curve.

3 Part C: 2D Decision Boundary Plotting

1. **Grid Generation:** Use `np.linspace` to create a dense 100×100 grid of x and y values covering the dataset range.

2. **Contour Mapping:** Calculate the value of z for every point on this grid using your optimized β parameters.

3. **Plotting:** Overlay a **black dotted line** on your original scatter plot using `plt.contour` at the level $z = 0$.

4 Part D: 3D Sigmoid Surface Visualization

Visualize the model's predictive confidence by plotting the Sigmoid function in 3D.

1. **3D Plot:** Create a 3D surface plot where the X and Y axes represent the features and the Z axis represents the predicted probability:

$$P(y = 1) = \sigma(z) = \frac{1}{1 + e^{-z}}$$

2. **Discussion:** Examine the "steepness" of the Sigmoid hill along the circular boundary. How does the noise (overlap) in the dataset manifest in the shape of this surface? Compare this to a case where the classes were perfectly separated.

Submission Instructions

- Submit a PDF report containing the optimized coefficients β and the three required plots (2D Scatter, 2D Boundary, and 3D Sigmoid Surface).
- Submit the corresponding `.ipynb` file.