Linear Programming-Based Classifier for Flower Species

Arryan Kanodia and Jiya Maheshwari

IE 411 – Large-Scale Linear Optimization, Fall 2024

December 6, 2024

1 Introduction

This project aims to design a linear classifier to distinguish between two species of flowers using data provided in `flowers_data.csv`. Each flower sample is represented by two features that are the length of the sepal and the length of the petal, and is labeled as species A (denoted as $z_i = 1$) or species B (denoted by $z_i = -1$). Our aim is to identify parameters $w = (w_1, w_2)$ and b that define a decision boundary

$$w_1 y_1 + w_2 y_2 - b = 0$$

to separate the two classes.

Because the data is not linearly separable in our case, we incorporate hinge loss variables to accommodate misclassifications and margin violations. We also apply a L_1 norm penalty on the weight vector w to encourage a simpler and potentially more generalizable model.

This approach transforms the classification task into a linear optimization problem. By solving a suitable Linear Program (LP), we obtain parameters that balance classification accuracy and model complexity.

2 Problem Setting and Data

The data set consists of $n = 40$ observations. Each observation i includes:

$$y_{i1} \text{ (sepal length)}, \quad y_{i2} \text{ (petal length)}, \quad z_i \in \{+1, -1\}.$$

A classifier ideally assigns species A if $w_1 y_{i1} + w_2 y_{i2} - b \geq 0$ and species B otherwise. To create a margin, we would prefer:

$$z_i(w_1 y_{i1} + w_2 y_{i2} - b) \geq 1.$$

However, since the data are not perfectly separable, we introduce hinge loss variables u_i that measure the degree of violation.

3 Linear Programming Formulation

Decision Variables

- $w_1, w_2 \in \mathbb{R}$: Components of the linear classifier's weight vector.
- $b \in \mathbb{R}$: The bias (intercept) term.
- $u_i \geq 0$: Hinge loss variables, one per data point i.
- $abs\_w_1, abs\_w_2 \geq 0$: Auxiliary variables for the models $|w_1|$ and $|w_2|$.

Objective Function

We aim to minimize the sum of hinge losses plus a scaled L_1 penalty on w:

$$\min_{w,b,u,abs\_w} \sum_{i=1}^{n} u_i \; + \; \frac{1}{10}(abs\_w_1 + abs\_w_2).$$

The term $\sum_{i=1}^{n} u_i$ penalizes misclassifications or points within the margin. The L_1 penalty $\frac{1}{10}(abs\_w_1 + abs\_w_2)$ encourages simplicity by controlling the magnitude of w.

Constraints

1. **Hinge Loss Constraints:** For each $i = 1, \ldots, n$:

$$u_i \geq 1 - z_i(w_1 y_{i1} + w_2 y_{i2} - b).$$

If a point is correctly classified with sufficient margin, u_i can be zero. Otherwise, u_i becomes positive to account for the margin violation.

2. **Absolute Value Constraints:** To linearize the L_1 norm, we impose:

$$abs\_w_1 \geq w_1, \quad abs\_w_1 \geq -w_1,$$

$$abs\_w_2 \geq w_2, \quad abs\_w_2 \geq -w_2.$$

These ensure $abs\_w_j = |w_j|$.

3. **Non-Negativity Constraints:**

$$u_i \geq 0, \; abs\_w_j \geq 0 \quad \forall i, j.$$

The resulting formulation is a Linear Program that can be solved with a standard LP solver like Gurobi.

4 Implementation Details

Data Loading

We use `pandas` to load the data from `flowers_data.csv`, given to us in the Project, extracting arrays for (y_{i1}, y_{i2}) and z_i.

Model Setup

Using Python and the Gurobi interface:

1. Define variables $w_1, w_2, b, u_i, abs\_w_1, abs\_w_2$.
2. Set the objective function: $\sum u_i + \frac{1}{10}(abs\_w_1 + abs\_w_2)$.
3. Add hinge loss constraints and absolute value constraints.

Optimization and Solution

We call `m.optimize()` to solve the Linear Program. Gurobi returns optimal values of w_1, w_2, b, and u_i. We then use the `matplotlib` library to plot:

- Data points: blue for $z_i = 1$ (species A) and red for $z_i = -1$ (species B).
- The decision boundary is defined as $w_1 x_1 + w_2 x_2 - b = 0$.

5 Results and Discussion

After solving, we obtain a linear boundary and associated hinge losses. While some misclassification may remain, the solution provides a reasonable compromise between accuracy and simplicity.

Optimal Values Example

Suppose the solver returns:

$$w_1^* = 0.4698, \quad w_2^* = -1.72195, \quad b^* = -5.72165.$$

These values are an example; actual results depend on the given dataset. With these parameters, the decision boundary is:

$$0.4698\, y_1 - 1.72195\, y_2 + 5.72165 = 0.$$

Hinge Losses

If a data point lies within or on the wrong side of the margin, the corresponding u_i is positive. Minimizing $\sum u_i$ pushes the boundary to correctly classify as many points with a margin as possible.

6 Conclusion

In this project, we formulated a linear classification problem as a Linear Program. We incorporated the hinge loss variables to handle non-separable data and used an L_1 penalty on w for regularization. Solving this Linear Program with Gurobi provided us with a reasonable linear classifier that balances accuracy and simplicity.

This approach demonstrates how optimization methods can be effectively applied to machine learning problems, offering a clear framework for handling misclassifications and controlling model complexity.