DA5401 A5: Visualizing Data Veracity Challenges in Multi-Label Classification

Objective: This assignment aims to deepen your understanding of the challenges in real-world machine learning, specifically in **multi-label classification**, by utilizing advanced non-linear dimensionality reduction techniques such as **t-SNE** and **Isomap**. You will visually inspect the data for issues such as noisy labels, outliers, and hard-to-learn data points, sparking curiosity about **data veracity** in a biological context.

---

1. Problem Statement

You are a data scientist analyzing gene expression data. You have been given the **Yeast Dataset**, where each data point (instance) represents an experiment, and the features are gene expression levels. The target is a set of **14 functional categories (labels)** to which the gene product may belong (multi-label classification). This dataset, despite being standardized, can still exhibit data veracity issues:

1. **Noisy/Ambiguous Labels:** Genes whose functions span multiple categories or are misclassified.
2. **Outliers:** Experiments with highly unusual gene expression profiles.
3. **Hard-to-Learn Samples:** Data points lying in regions where functional categories are thoroughly mixed.

Your task is to apply t-SNE and Isomap to the feature vectors to visually expose these data quality issues, thereby understanding the challenges a classifier would face.

You will submit a Jupyter Notebook with your complete code, visualizations, and a plausible story that explains your findings. The notebook should be well-commented, reproducible, and easy to follow.

Dataset:

- **Yeast Dataset:** The feature matrix X and the binary multi-label matrix Y (with 14 labels) are standard files available from the **Mulan Repository** or other machine learning data repositories.
- **Download Link (Example Source - use the text files):** MULAN Repository - Yeast Data (Look for `yeast.arff` and the corresponding label file, or a pre-converted CSV/NumPy format).

---

2. Tasks

Part A: Preprocessing and Initial Setup [10 points]

1. **Data Loading [2]:** Load the feature matrix X (86 features) and the multi-label target matrix Y (14 labels).
2. **Dimensionality Check:** Report the initial number of features and the number of data points.

3. **Label Selection for Visualization [5]:** To simplify the visualization (since 14 colors can be overwhelming), create a new target variable for coloring that represents the **two most frequent single-label classes** and the **most frequent multi-label combination**. Assign an "Other" category to the rest. This approach creates a simple, distinct categorical index for coloring the plots.
4. **Scaling [3]:** Explain why scaling is crucial before applying distance-based dimensionality reduction techniques. Apply **Standardization** to the feature matrix X.

Part B: t-SNE and Veracity Inspection [20 points]

1. **t-SNE Implementation [5]:** Apply **t-Distributed Stochastic Neighbor Embedding (t-SNE)** to the scaled feature matrix X to reduce it to 2 dimensions. Experiment with the **perplexity** hyperparameter (e.g., 5, 30, 50) and note how the visualization changes. Justify your final choice of perplexity.
2. **Visualization [5]:**
    - Create a 2D scatter plot of the final t-SNE coordinates.
    - Color each data point according to the categorical index you created in Part A.
3. **Veracity Inspection [10]:** Analyze the resulting plot and visually identify regions corresponding to:
    - **Noisy/Ambiguous Labels [4]:** Points where one color is deeply embedded within a cluster of a different color.
    - **Outliers [3]:** Isolated points or tiny, distant clusters. Hypothesize what these unusual expression patterns might represent.
    - **Hard-to-Learn Samples [3]:** Areas where functional category colors are thoroughly mixed. Explain why a simple classifier would likely struggle in these regions.

Part C: Isomap and Manifold Learning [20 points]

1. **Isomap Implementation [5]:** Apply **Isomap** to the scaled feature matrix X, reducing it to 2 dimensions. Explain the fundamental difference between Isomap and t-SNE in terms of how they preserve data structure (global vs. local).
2. **Visualization [5]:** Create a 2D scatter plot of the Isomap coordinates, using the same coloring scheme.
3. **Comparison and Curvature [10]:**
    - Compare the Isomap visualization to the t-SNE visualization. Which one is better at revealing the **global structure** of the gene expression data? [5]
    - Discuss the concept of the **data manifold**. Does the Isomap plot suggest a highly curved or complex manifold? How does the complexity of this manifold relate to the difficulty of classification? [5]

---

3. Submission Guidelines

- The assignment is due in **1 week**.
- Submit a single Jupyter Notebook with all your code, visualizations, and answers to the conceptual questions in markdown cells.
- Ensure all code is clean, readable, and reproducible.

Evaluation Criteria:

- Correct and justified application of t-SNE and Isomap.
- Quality and clarity of visualizations.
- Insightful analysis and correct identification of data veracity issues from the plots.
- Demonstrated understanding of the theoretical differences between t-SNE and Isomap.
- Thoughtful discussion on how data veracity issues, revealed by visualization, impact classification model performance.

Good luck!