# Assignment 1, FYS-2021,
# Logistic Regression Analysis on Spotify Data

Truls Larsen

UiT - Norges Arktiske Universitet

September 8, 2024

## Problem 1: Data Preparation and Visualization

### 1a: Dataset Overview

The Spotify dataset consists of **232,725** songs, each with **18** features representing various musical attributes. The dataset includes a wide range of genres, e.g. Pop and Classical.

### 1b: Data Filtering and Labeling

The Spotfy dataset the dataset was filtered to include only songs categorized as 'Pop' or 'Classical'. After filtering:

- 'Pop' samples/songs labeled as 1: **9,386**.
- 'Classical' samples/songs labeled as 0: **9,256**.

### 1c: Data Splitting

The dataset is the filtered to only select the fatures features 'liveness' and 'loudness', for the classification task. And the dataset was split into training (80%) and test (20%) sets with from bulit in sklearn model in python, with maintaining an even distribution of genres in both sets with and picking random samples with within the class in each set.

| Class | Genre | Training samples | Test samples |
|---|---|---|---|
| Class 1: '1' | Pop | 7508 | 1878 |
| Class 2: '0' | Classical | 7405 | 1851 |

### 1d: Feature Visualization

To visualize the distribution of the selected features, a scatter plot of 'liveness' vs 'loudness' was created for the two genres. The samples from the two classes are overlapping and it could be difficult to optimize 100% correct linear classifier.

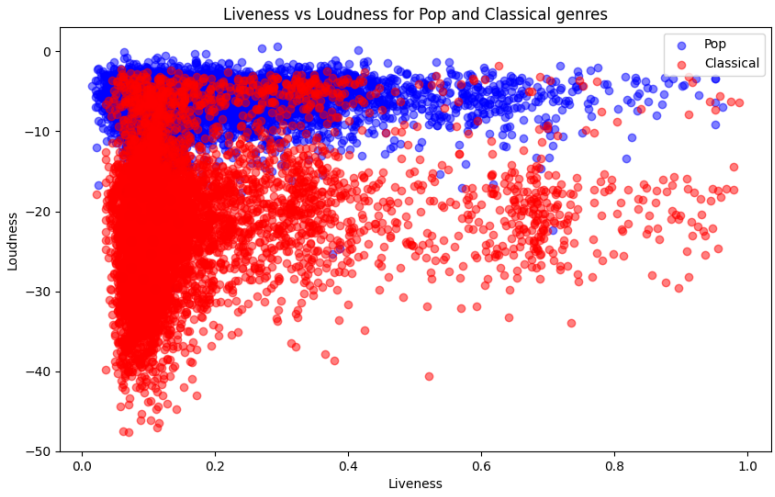


Figure 1: Scatter plot of Liveness vs Loudness for Pop and Classical genres.

# Problem 2: Logistic Regression Model

## 2a: Model Training

Implementing a logistic discriminent classifier with a stochastic gradient decent.

$$\hat{y} = \begin{cases} 1, & \text{if } \sigma(z) \geq 0.5 \\ 0, & \text{if } \sigma(z) < 0.5 \end{cases} \tag{1}$$

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{2}$$

$$z = w_1 \cdot x_1 + w_2 \cdot x_2 + b \tag{3}$$

Where: $\hat{y}$: predicted y, $\sigma$: is the Sigmoid function, $z$: , $x_i$: features, $w_i$: weights, $\epsilon$: constant to avoid $log(0)$.

The model uses the following predefined constants:

- learning rate: $lr = 0.001$
- number of epochs: 10
- $w_1 = [0, 0]$.
- $\epsilon = 10^{-8}$, and does not affect the the outcome to much.

Cross entrypy function for computing teh loss L:

$$L = -\sum_{i=1}^{m} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \tag{4}$$

A logistic regression model was trained using stochastic gradient descent. The model parameters were optimized over 10 iterations with a learning rate of 0.001. The training process involved calculating gradients, updating weights, and minimizing the cost function.

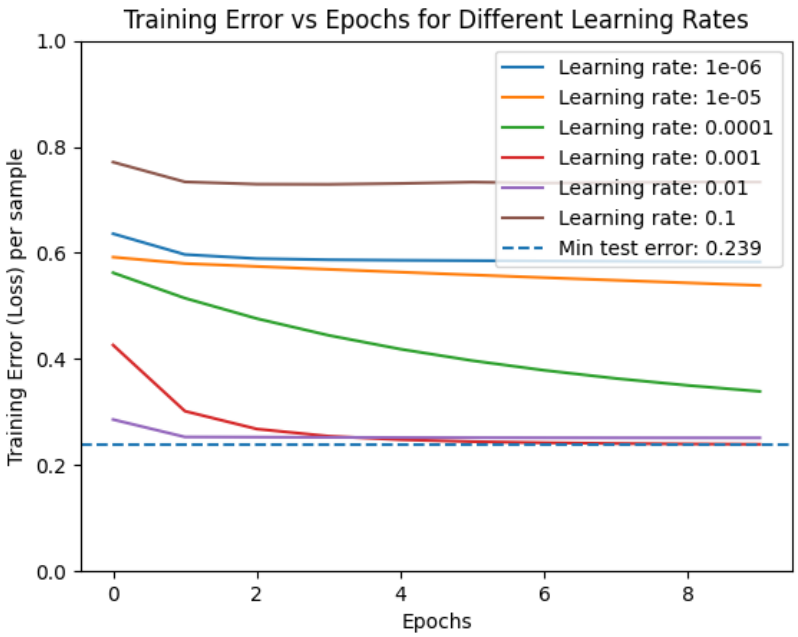


Figure 2: Learning rates for 10 epochs

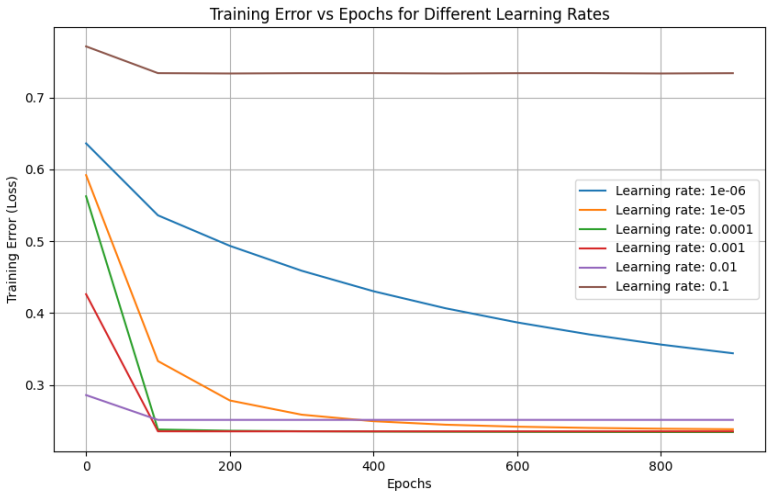


Figure 3: Learning rates for 1000 epochs, with 100 epoch resolution

## 2b: Test Set Evaluation

The trained model was evaluated on the test set, achieving an accuracy of **92.28%**. This result indicates that the model was able to classify the majority of the songs correctly.

## 2c: Decision Boundary Visualization

The decision boundary given by w and b from the regression model.

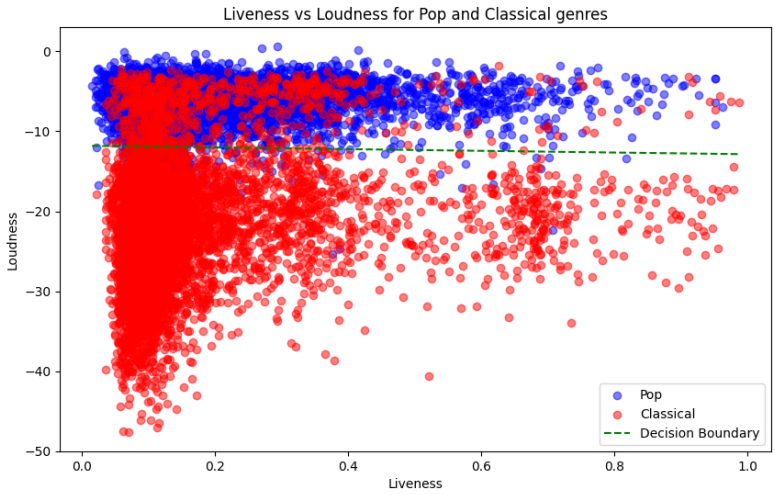


Figure 4: Decision Boundary on the Liveness vs Loudness plot.

# Problem 3: Results and Analysis

## 3a: Confusion Matrix

The confusion matrix was manually calculated to provide a detailed breakdown of the model's performance on the test set. This matrix reveals the number of true positives, true negatives, false positives, and false negatives.

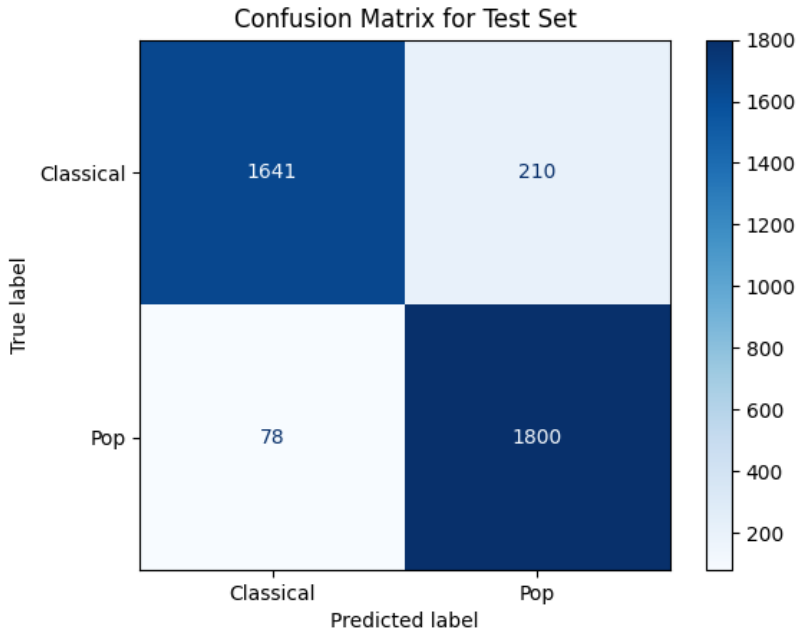


Figure 5: Confusion Matrix for the Test Set.

### 3b: Performance Analysis

While the accuracy metric gives an overall sense of the model's performance, the confusion matrix offers deeper insights. It highlights the specific areas where the model performed well and where it struggled. For example, the model's ability to correctly classify 'Pop' songs as 'Pop' and 'Classical' songs as 'Classical' is evident from the high values along the diagonal of the matrix.

# References


[1] Alpaydin, Ethem. *Introduction to Machine Learning*. MIT Press, 2014.

[2] Lecture notes and slides from Canvas in course FYS-2021.

[3] Spotify Dataset from Canvas in course FYS-2021.

[4] Python Documentation, *Numpy*, *Pandas*, *Matplotlib*. Available at: `https://numpy.org/doc/`, `https://pandas.pydata.org/pandas-docs/stable/`, `https://matplotlib.org/stable/contents.html`