Project Report

Problem

The Sample Analysis at Mars (SAM) instrument is a key part of NASA's Curiosity rover mission to explore the red planet. SAM's primary goal is to detect and analyse organic molecules. This rover collects data using a method of chemical analysis—gas chromatography–mass spectrometry (GCMS)—performed on soil and rock samples.
The goal of this project is to detect specific organic (and inorganic) compounds in the sample, like aromatic, hydrocarbons, and others. This can be achieved by analysing the GCMS data using data analysis techniques and building machine learning models to predict the presence of different compounds in the sample.

Approach

The presence of a compound in a sample depends on the presence of its constituent compounds and molecules. In each data sample, the intensity values of ions of different mass-to-charge ratios (m/z) are recorded at different times. The intensity values represent the relative number of compounds of a particular m/z ratio. Thus by detecting peaks in the intensity values, we can get the m/z values of the constituents and eventually find whether a compound is present.

Solution

The idea behind the primary solution is to detect peaks in the data. Initial steps in this approach involved compressing the data into valuable features by applying mean and max at different data segments. After min-max scaling, these features are fed to a simple logistic regression model for training and validation. The predicted values showed very low positive but good negative class precision. To refine the model and extract more relevant information (without loss), I calculated the extreme values of time, mass, and intensity from the entire dataset. Based on this, I grouped time and mass into values ranging from 0 to 2700 and 0 to 1400, respectively.
Since each training data is now a matrix of 2700*1400 values, training using simple logistic regression models was inefficient and time-consuming. For processing this large amount of data in the form of matrices, I used Convolutional Neural Network (CNN). These are generally used for image classification problems, where they help in extracting features like edges and curves. The model used convolutional layers and max pooling layers to detect data peaks, followed by fully connected layers with 'relu' activation for faster convergence. The output layer uses a 'sigmoid' activation to predict nine independent probability values. This architecture effectively captured the relevant features and was able to predict with good precision.

Performance

During the testing, the model showed validation loss of 0.16 (approx) and training loss of 0.12 (approx). The classification reports showed around 96% precision for negative classes, 90% precision for positive classes. The model gave slightly better performance for classes with more positive examples than the classes with less positive examples.