Enhancing Stellar Classification with Active Learning and Machine Learning Models

CSE623 – Machine Learning Theory and Practice

Winter Semester 2025

Weekly Report1 – 22/02/2025

Group Members:

Riya Mahendra

Kishan Malaviya

Yogeshkumar Patel

The task of literature Review and dataset evaluation have been performed this week.

Paper:

El-Kholy, R. I., & Hayman, Z. M. (2025). Optimised of SDSS- IV MaStar spectra for stellar classification using supervised models. *Astronomy & Astrophysics, 693*, A300. https://doi.org/10.1051/0004-6361/202451309

Literature Review:

The paper explores the application of Active Learning (AL) algorithms to optimize the sampling of stellar spectra for classification tasks using supervised machine learning models. The primary goal of the study is to reduce the size of the training dataset required for stellar classification while maintaining or improving model performance. The authors used the MaStar Stellar Library form the Sloan Digital Survey (SDSS) Data Release 17 (DR17). This library contains gigh-quality empirical stellar spectra covering a wide range of stellar parameters, such as effective temperature (Teff), surface gravity (log g), and iron metallicity ([Fe/H]). The dataset is highly imbalanced, which has been addressed through several strategies, both in the preprocessing stage and during the application of AL algorithms. The author also applied quality cuts to ensure the reliability of the data, and they used Value-added catalog (VAC) from SDSS DR17, which provides stellar parameter measurements derived using different methods. The final dataset was used to classify stars into different categories based on stellar parameters, with the class distribution showing significant imbalance.

Methods:

The initial stage is preprocessing, which consists of four steps- 1. Feature Selection, where specific absorption lines (e.g., Ca II K, H $\delta\delta$, Fe I, etc.) have been selected that are known to be sensitive to stellar parameters. 2. Data Splitting, where the datasets were split into training and testing sets. Stratification was used during the split to ensure that the class imbalance was preserved in both sets. 3. Scaling, features were scaled using min-max normalization. 4.

Dimensionality Reduction, where Principal Component Analysis (PCA) was applied to further reduce the dimensionality of the data.

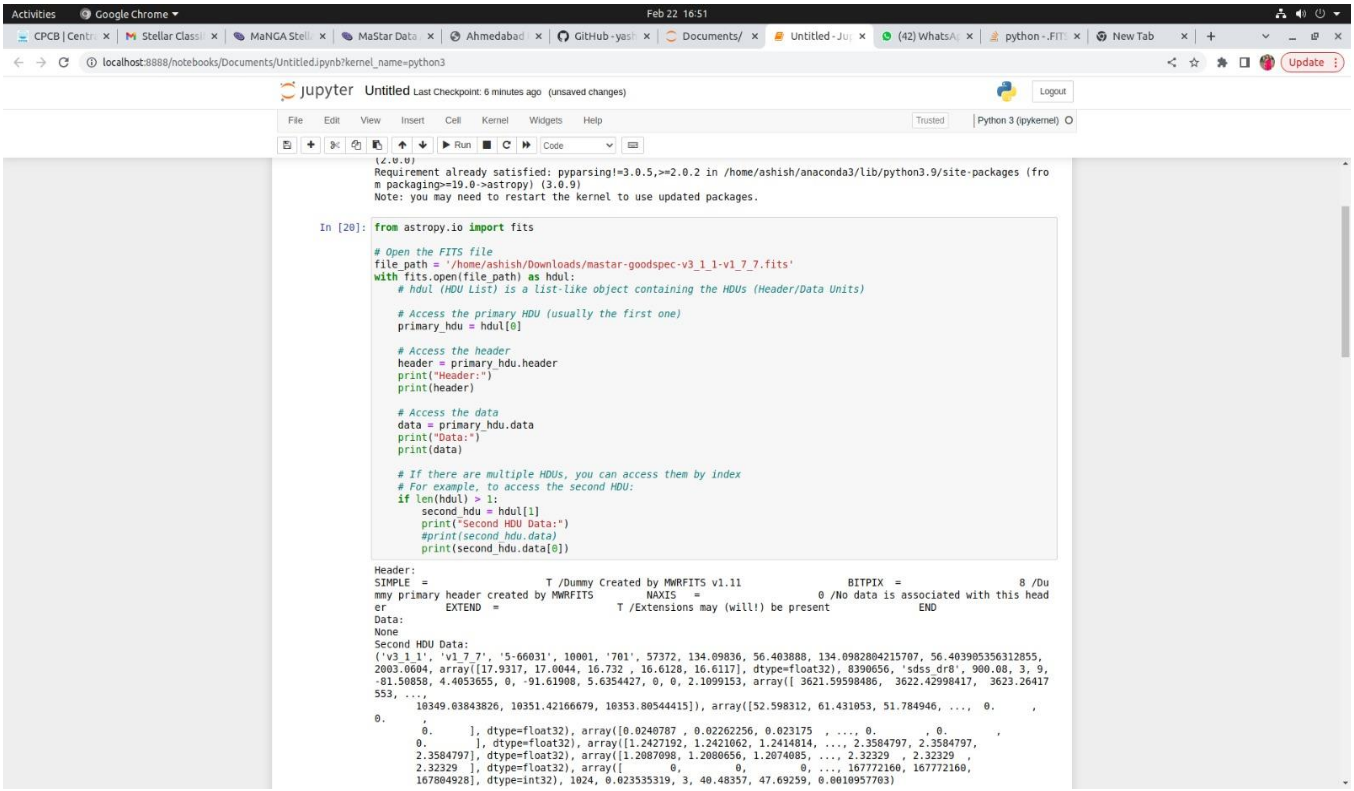
The second stage is use of AL algorithms. Here, author used pool-based AL to iteratively select the most informative instances from the unlabelled data pool. They tested six different AL strategies, divided into two categories:

- **Uncertainty Sampling**: This includes strategies based on classification uncertainty, classification margin, and classification entropy. These strategies select instances where the model is most uncertain about the predicted class.
- **Query by Committee (QBC)**: This includes strategies based on vote entropy, consensus entropy, and maximum disagreement. These strategies use a committee of models to select instances where the models disagree the most.

The third stage is applying the Machine Learning Models. In this authors used the three models: K-Nearest Neighbours (KNN), Random Forest (RF), and Gradient Boosting (GB). They also used an ensemble model that combined the predictions of these models. The models were trained on the samples selected by the AL algorithms, and their performance was compared to models trained on randomly sampled data. To evaluate the performance of the models, authors used several metrics such as accuracy, sensitivity, specificity, area under the curve (AUC), and Matthew's Correlation Coefficient (MCC).

We understand that Active Learning is a powerful approach for reducing the labelling cost in supervised machine learning tasks. The authors demonstrated that AL can select the most informative instances for labeling, leading to better model performance with fewer training samples. The use of feature selection and dimensionality reduction techniques also played a crucial role in improving the efficiency of the models.

Dataset Evaluation in python:



```
(2.0.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /home/ashish/anaconda3/lib/python3.9/site-packages (fro
m packaging>=19.0->astropy) (3.0.9)
Note: you may need to restart the kernel to use updated packages.
```

```
In [20]: from astropy.io import fits

         # Open the FITS file
         file_path = '/home/ashish/Downloads/mastar-goodspec-v3_1_1-v1_7_7.fits'
         with fits.open(file_path) as hdul:
             # hdul (HDU List) is a list-like object containing the HDUs (Header/Data Units)

             # Access the primary HDU (usually the first one)
             primary_hdu = hdul[0]

             # Access the header
             header = primary_hdu.header
             print("Header:")
             print(header)

             # Access the data
             data = primary_hdu.data
             print("Data:")
             print(data)

             # If there are multiple HDUs, you can access them by index
             # For example, to access the second HDU:
             if len(hdul) > 1:
                 second_hdu = hdul[1]
                 print("Second HDU Data:")
                 #print(second_hdu.data)
                 print(second_hdu.data[0])
```

```
Header:
SIMPLE  =                    T /Dummy Created by MWRFITS v1.11                BITPIX  =                    8 /Du
mmy primary header created by MWRFITS        NAXIS   =                    0 /No data is associated with this head
er           EXTEND  =                    T /Extensions may (will!) be present                 END
Data:
None
Second HDU Data:
('v3_1_1', 'v1_7_7', '5-66031', 10001, '701', 57372, 134.09836, 56.403888, 134.0982804215707, 56.403905356312855,
2003.0604, array([17.9317, 17.0044, 16.732 , 16.6128, 16.6117], dtype=float32), 8390656, 'sdss_dr8', 900.08, 3, 9,
-81.50858, 4.4053655, 0, -91.61908, 5.6354427, 0, 0, 2.1099153, array([ 3621.59598486,  3622.42998417,  3623.26417
553, ...,
       10349.03843826, 10351.42166679, 10353.80544415]), array([52.598312, 61.431053, 51.784946, ...,  0.        ,
0.        ,
        0.        ], dtype=float32), array([0.0240787 , 0.02262256, 0.023175  , ..., 0.        , 0.        ,
       0.        ], dtype=float32), array([1.2427192, 1.2421062, 1.2414814, ..., 2.3584797, 2.3584797,
       2.3584797], dtype=float32), array([1.2087098, 1.2080656, 1.2074085, ..., 2.32329  , 2.32329  ,
       2.32329  ], dtype=float32), array([        0,         0,         0, ..., 167772160, 167772160,
       167804928], dtype=int32), 1024, 0.023535319, 3, 40.48357, 47.69259, 0.0010957703)
```

For the next week:

1. We will work on feature selection part. In which we will investigate the absorption lines used by the authors for feature selection and understand their significance in stellar classification.