[Project Code: PSSVM]

Pulsar Star Classification using Support Vector Machines

Project Duration : 26-Feb-2023 ~~ 18-Mar-2023
Submission Information : (via) CSE-Moodle

Objective:
Pulsars are a rare type of Neutron star that produce radio emission detectable here on Earth. They are of considerable scientific interest as probes of space-time, the inter-stellar medium, and states of matter. Each pulsar produces a slightly different emission pattern, which varies slightly with each rotation. Thus a potential signal detection known as a 'candidate', is averaged over many rotations of the pulsar, as determined by the length of an observation. In the absence of additional info, each candidate could potentially describe a real pulsar. However in practice almost all detection is caused by radio frequency interference (RFI) and noise, making legitimate signals hard to find.

Machine learning tools are now being used to automatically label pulsar candidates to facilitate rapid analysis. Classification systems in particular are being widely adopted, which treat the candidate data sets as binary classification problems. Here the legitimate pulsar examples are a minority positive class, and spurious examples the majority negative class. In this project, your task is to develop a support vector machine classifier to classify pulsar stars as legitimate or spurious.

Your Tasks:

1. *Building a SVM Classifier*
    a. **Pre-processing the data**:
        i. Randomly pick 80% of the data as a training set and the rest as a test set.
        ii. Normalize each feature of the dataset to have zero mean and unit variance. Note that while normalizing the features, their mean and variance should be computed over the train split only. Once, the mean and variance is computed using only the train split, you normalize the test split using the mean and variance computed over the train split.
    b. **Training the model**:
        i. Note that training requires solving the dual optimization problem. To solve the dual optimization problem you can use any python packages like: CVXOPT or Scipy.optimize.minimize
        ii. Implement the following three kernels : linear, quadratic, radial basis function
    c. **Making predictions:** Write a function that takes new datapoint as input and predicts the class
    d. **Evaluation:** Finally, you should generate results on the given data and compare its results with the sklearn module (sklearn.svm)

2. *Hyper-parameter Tuning*
    In Support Vector Machines, the Gamma and C hyper-parameters control the model's flexibility and generalization performance.
    a. Choose 5 values of Gamma and 5 values of C. (Justify your choice in report). Tune the hyper-parameters by doing a grid search on all combinations of (kernel, gamma, C).
    b. Report test set accuracy for all the 3x5x5=75 combinations in a tabular form.

3. *Visualization*
    a. Consider the model (with best hyper-parameters) and plot the the decision boundary and the support vectors, on both train & test set. (You may use suitable python packages for this task)

4. *Report*
    a. Prepare a concise report or manual, spanning 2 to 3 pages, that details the results of your work, along with your observations and explanations. Be sure to include a clear and thorough overview of the methods used, the results obtained, and your interpretation of the findings.

Dataset: The dataset contains samples of pulsar candidates collected during the High Time Resolution Universe Survey (South). It has around 17898 instances with 8 continuous attributes. The target attribute is “Class” which can be legitimate(1) or spurious(0).

Data Filename: **pulsar_star_dataset.csv**
(Please note that the dataset may contain missing values. To handle these missing values, you should use appropriate techniques and clearly explain your methods in the report)

Dataset description:
The first four attributes are simple statistics obtained from the integrated pulse profile (folded profile). This is an array of continuous variables that describe a longitude-resolved version of the signal that has been averaged in both time and frequency. The remaining four variables are similarly obtained from the DM-SNR curve. These are summarized below:

1. Mean of the integrated profile.
2. Standard deviation of the integrated profile.
3. Excess kurtosis of the integrated profile.
4. Skewness of the integrated profile.
5. Mean of the DM-SNR curve.
6. Standard deviation of the DM-SNR curve.
7. Excess kurtosis of the DM-SNR curve.
8. Skewness of the DM-SNR curve.
9. Class

Submission Details: (to be submitted in CSE-Moodle, **by one representative of the group**)

1. ZIPPED folder containing code (with comments) and the dataset files
2. Report (in pdf format)

Submission Guidelines:

1. You may use one of the following languages: C / C++ / Java / Python.
2. Your program should run on a Linux Environment.
3. Your program should be standalone and should not use any special purpose library for Machine Learning. (Apart from numpy, pandas and the one’s mentioned in tasks). And, you can use libraries for other purposes, such as formatting and visualization of data.
4. You should submit the program file and a README file with instructions to run the code.
5. You should name your file as <GroupNo_ProjectCode.extension>.
   (e.g., *Group99_PSSVM.zip* for code-distribution and *Group99_PSSVM.pdf* for report)
6. The submitted program file *should* have the following header comments:
   # Group Number
   # Roll Numbers : Names of members (listed line wise)
   # Project Number
   # Project Title
7. Submit through CSE-MOODLE only.
   Link to our Course page: https://moodlecse.iitkgp.ac.in/moodle/course/view.php?id=508

You should not use any code available on the Web. Submissions found to be plagiarized or having used ML libraries (except for parts where specifically allowed) will be awarded zero marks.

For any questions about the assignment, contact the following TA:
Abhinav Bohra (Email: abhinavbohra09@gmail.com)