Project Assignment

The goal of this project assignment is to give students an opportunity to build and test classifier models using a real-world data. The data to be used is a part of the 2018 BRFSS Survey Data prepared by CDC. You must go to the website from which the dataset was downloaded and read the description of the dataset. The website is:

https://www.cdc.gov/brfss/annual_data/annual_2018.html

You may use any software or tools for the whole process of this project:

- You may use any software tool(s) to perform data preprocessing.
- You may use any software tool(s) for attribute selection.
- You may use any software tool(s) to build and test classification models.
- Regardless of which tool(s) you use, you must collect and submit all performance measures of all classification models as described in the deliverables section of this assignment.

You are given three files: *project-2018-BRFSS-arthritis.csv*, *project-2018-BRFSS-arthritis.arff*, and *codebook19_llcp-v2-508.pdf*.

The *project-2018-BRFSS-arthritis.csv* (or *project-2018-BRFSS-arthritis.arff*) file has the dataset for the project and it has 11933 tuples and 108 attributes. Each tuple is a person who participated in the survey and each attribute represents an answer to a survey question. The class attribute is *havarth3* and its value is either 1 or 2. The value of 1 means that the person was ever told to have some form of arthritis, rheumatoid arthritis, gout, lupus, or fibromyalgia. Otherwise, the value is 2.

The *codebook19_llcp-v2-508.pdf* has information about the dataset, including the meanings of the attributes.

Requirements

You must build and test classification models multiple times using different attribute selection methods and different classification algorithms.

First, divide the initial dataset into a training dataset and a test dataset. The training dataset must have approximately 66% of the initial dataset and the test dataset must include the remaining tuples. The class distribution in the training dataset and the test dataset must be the same as, or very close to, the class distribution in the initial dataset. Name the training dataset *project-training.csv* (or *project-training.arff*) and the test dataset *project-test.csv* (or *project-test.arff*). You may use any tool to split the dataset. You just have to make sure that the class distribution is preserved. A file named *stratified-split.pdf* is posted on Blackboard (under *Other Course Files*), which shows how to split a dataset using Weka.

The, follow the steps described below:

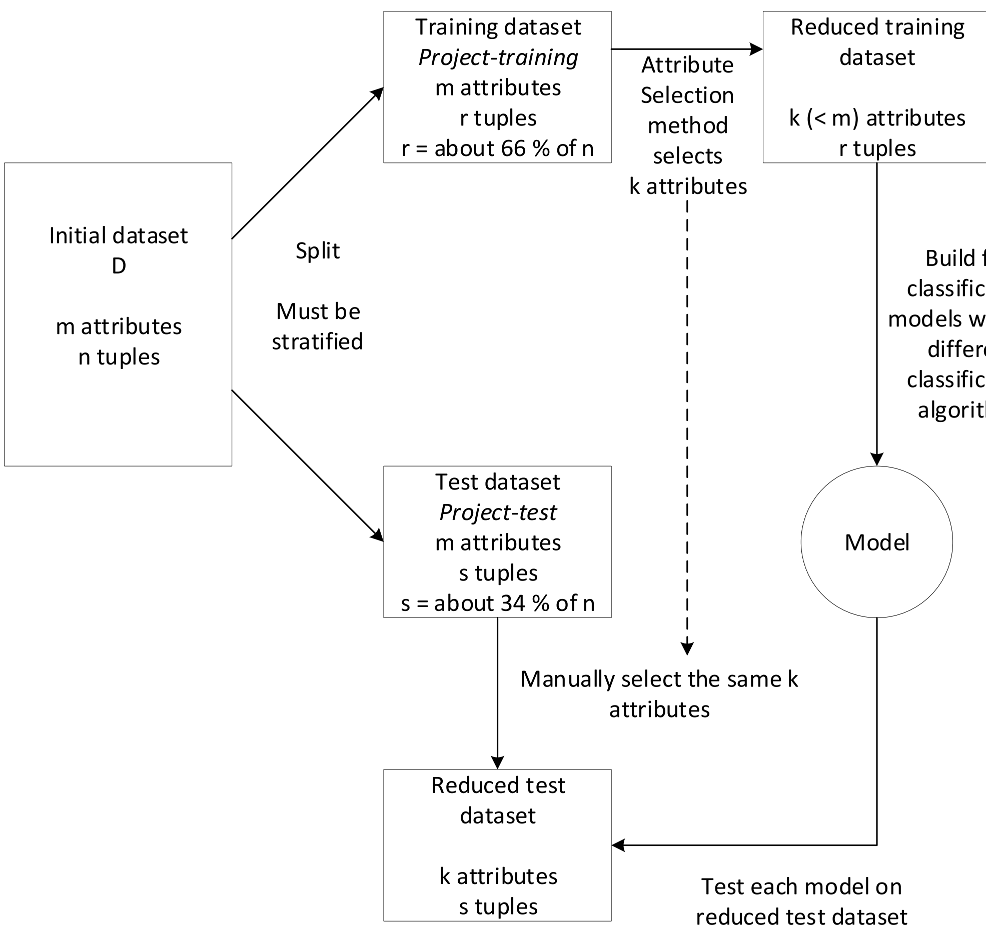
1. Apply an attribute selection method on the *project-training.csv* (or *project-training.arff*) dataset to select a subset of attributes. Let's call this *reduced training dataset*. From the *project-test.csv* (or *project-test.arff*) dataset, select only those attributes that are in the reduced training dataset. Let's call this *reduced test dataset*. So, the reduced training dataset and the reduced test dataset have exactly the same attributes.

2. Build a classifier model using a classification algorithm from the reduced training dataset and test it on the reduced test dataset. Collect the test result. Repeat this four more times with four different classification algorithms. So, you will build and test five classifier models with the same reduced training dataset and the same reduced test dataset.

Repeat the above steps four more times with four different attribute selection methods.

At the end, you will have built and tested total 25 classifier models.

One iteration of the process is illustrated below:



Note on attribute selection: Some attribute selection methods will give you a subset of attributes. In this case, you can use those selected attributes. However, some attribute selection methods will not select a subset of attributes for you. Instead, they will give you all attributes with ranks (the ranks are determined by each attribute selection method). In this case, it is your responsibility to choose a certain number of attributes.

After collecting the results of all 25 iterations, you must choose the "best" model that gave you the 'best" classification performance.

Deliverables

You must submit the following:

- Datasets
  - *project-initial.csv* (or *project-initial.arff*). This is the dataset you have after all preprocessing and before splitting.
  - *project-training.csv* (or *project-training.arff*)
  - *project-test.csv* (or *project-test.arff*)
  - *The reduced training dataset* from which your best model was built. Name this file *best-train.csv* (or *best-train.arff*)
  - *The reduced test dataset* on which your best model was tested. Name this file *best-test.csv* (or *best-test.arff*)
- Script files
  - If you used R or Python, you must submit the script files you wrote.
- Documentation
  - Name this file *LastName-FirstName-projec-report.pdf*.
  - Your documentation must include
    - Detailed description of all data preprocessing you performed.
    - For each attribute selection method you used, show:
      (1). The name of the attribute selection method and a brief description of the method.
      (2). The list of attributes that were selected by this method.
    - Names of all classifier algorithms you used and a brief description of each classification algorithm.
    - Test results of all 25 models. Each test result must include

      - Confusion matrix
      - **For each class**: TP rate, FP rate, precision, recall, F-measure, ROC area, and MCC.
      - Weighted average (over all classes) of each of the above performance measures.
      - An example is shown below:

```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    1.000     1.000     Iris-setosa
                 0.960    0.040    0.923      0.960    0.941      0.911    0.992     0.983     Iris-versicolor
                 0.920    0.020    0.958      0.920    0.939      0.910    0.992     0.986     Iris-virginica
Weighted Avg.    0.960    0.020    0.960      0.960    0.960      0.940    0.994     0.989

=== Confusion Matrix ===

  a  b  c   <-- classified as
 50  0  0 |  a = Iris-setosa
  0 48  2 |  b = Iris-versicolor
  0  4 46 |  c = Iris-virginica
```

The above is an output from Weka. It also includes PRC Area. However, the PRC Area is not required (if you use other tools).

- The name of the attribute selection method and the classifier algorithm that gave you the best performance (e.g., GainRatio attribute selection method with a decision tree algorithm gave me the best performance).
- The list of attributes that are in your *best-train.csv* (or *best-train.arff*)
- Best test result. This is the one you obtained by testing your best model on the *best-test.csv* (or *best-test.arff*) test dataset. It is one of 25 test results. This test result must include all performance measures and the confusion matrix as described above.
- Discussion:
  (1). Describe what criteria you used when you were choosing your *best model*. In other words, you need to justify why you chose that particular model as your best model. If your justification is not based on sound technical criteria, you will lose points.
  (2). List five attributes that you think are most relevant to the class attribute. You need to justify why you selected those five attributes.
  (3). What you learned from this project.
  (4). Any other observations from this project.

Submission

You must submit one archive file (such as a *zip* file or a *rar* file) that contains all dataset files and the documentation file. You may include additional files that you think are relevant to your project. Name this archive file *LastName_FirstName_project.zip* (or other appropriate extension) and submit it to Blackboard.

Grading

There is no one correct answer for this project and there is no performance threshold based on which your grade is determined. Your project will be graded based on:

- Whether all required steps are followed correctly
- Whether all required datasets and/or script files are submitted
- Whether all required components are included in your report
- Whether the justification for your best model is technically sound

- How well (meaning clearly, consistently, and unambiguously) your documentation is written.
- Whether discussion part of the report is "substantive."

Important note

- When you split the initial dataset, make sure that your training dataset and the test dataset are independent (i.e., there is no overlap between the two datasets). If they are not independent, up to 40% will be deducted
- If needed, we will build and test your best model following the description in your report. So, it is very important that your report must be written in detail so that we may be able to replicate what you did. If we cannot reproduce the performance of your best model, up to 40% will be deducted.
- Your best model may be tested on an independent test dataset. If the performance obtained from this independent dataset is different from the performance of your best model reported in your documentation, up to 40% will be deducted.