Project Assignment

The goal of this project assignment is to give students an opportunity to build and test classifier models using 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

The **primary objective** of this project is for you to (1) research attribute selection methods, (2) apply attribute selection methods to your data, and (3) successfully apply the classification models you have learned in class. The **secondary objective** of this project is to expose you to the complexities of working with messy data. This is very common in many academic research and corporate research work. The intention is for you to experience some of the real-world challenges. However, please do not spend too much time on preprocessing and attribute selection. It is not expected for you to solve *all* of the real-world challenges in this project.

You may use any combination of 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.

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*. A *havarth3* value of 1 means that the person was ever told to have some form of arthritis, rheumatoid arthritis, gout, lupus, or fibromyalgia. Otherwise, the *havarth3* 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. You should do the following:

- Apply pre-processing to your data as you see fit (this could be scalers, encoding techniques, feature generation methods, instance-reduction techniques, missing data handling techniques, or any other pre-processing you feel would be appropriate). There are MANY issues in this data that pre-processing can fix. You do not need to fix ALL issues but should address some. Do **not** do attribute selection in this step (that will come later in this project). Explain why you do the pre-processing that you choose to do.

- 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. You may use any tool to split the dataset, but you need 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.

Save your pre-processed data as *yourname-project-initial.csv* (or *.arff* or *.jmp*). Then split this data into your train and test and save each as *yourname-project-training.csv* (or *.arff* or *.jmp*) and the test dataset as *yourname-project-test.csv* (or *.arff* or *.jmp*).

Next, do the following:

- 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 should have the same attributes. Note that attribute selection MUST remove some attributes.

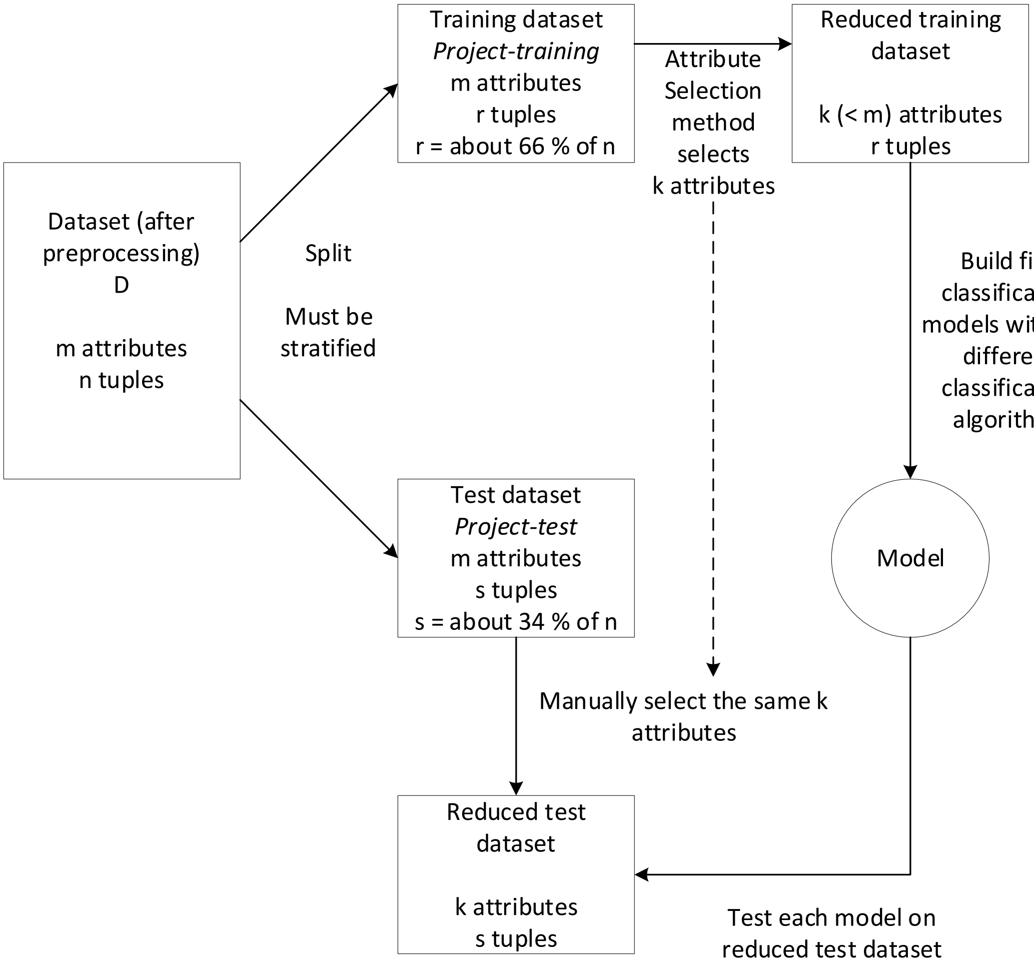
- Build a classifier model using a classification algorithm from the reduced training dataset and test it on the reduced test dataset. Collect the test result. Repeat this for four additional times so you have explored five different classification algorithms. So, you will build and test five classifier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

- Repeat these two steps for four additional times so you end up with results for five different attribute selection methods.

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.

At the end, you will have built and tested a total of 25 classifier models (5 classification algorithms times 5 attribute selection methods). You may use the same classification algorithms within each attribute selection method if you wish. After collecting the results of all 25 iterations, you must choose the "best" model that gave you the 'best" classification performance.

Each of your 25 models should generally follow the process illustrated on the next page (you only need to pre-process and test-train split the data once):

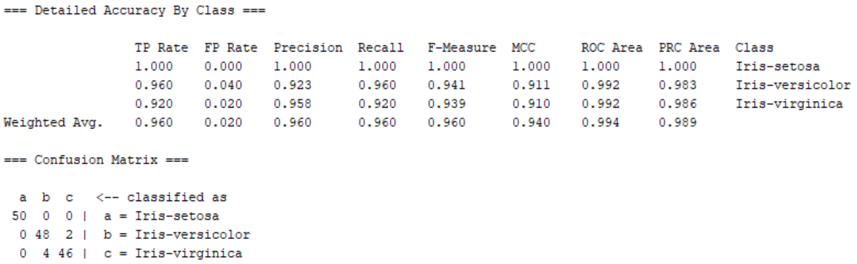


Deliverables

You must submit the following:

- Datasets
  - *yourname-project-initial.csv* (or *.arff* or *.jmp*). This is the dataset you have after all preprocessing and before splitting.
  - *yourname-project-training.csv* (or *.arff* or *.jmp*)
  - *yourname-project-test.csv* (or *.arff* or *.jmp*)
  - *The reduced training dataset* from which your best model was built. Name this file *yourname-best-train.csv* (or *.arff* or *.jmp*)
  - *The reduced test dataset* on which your best model was tested. Name this file *yourname-best-test.csv* (or *.arff* or *.jmp*)
- Script files
  - If you used R or Python, you must submit the script files you wrote.
- Documentation
  - Submit a Word or PDF report. The filename should include your name.
  - Your document should read like a report/paper. It must include

- A description of all data preprocessing you performed and why.
- Names of all attribute selection methods you used, a brief description of each method, and the list of attributes selected by the method.
- Names of all classifier algorithms you used and a brief description of each classification algorithm.
- A summary of what you learned from building the 25 models. In the main part of your report you don't have to discuss all 25 models but you should mention the most interesting ones. You should also provide detail on all 25 models in an Appendix.
- A description of what criteria you used when you were choosing your *best model*. If your justification is not based on sound technical criteria, you will lose points.
- 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). This includes:
    - 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 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 table is an output from Weka. It also includes PRC Area. However, the PRC Area is not required (if you use other tools).

- Some reflection on what you've learned:
  (1). List the five attributes that you think are most relevant to the class attribute. You need to justify why you selected those five attributes.
  (2). What you learned from this project.
  (3). Any other observations from this project.
- Reminder: Provide the test results for all 25 models (including the best one) in an **Appendix**. You must include all the metrics requested above (with the best result).

Submission

Upload the Word/PDF report. Separately, combine all remaining files into an archive file (such as a *zip* file or a *rar* file). This will include all dataset files and any additional files that you think are relevant to your project. Again, be sure all files you create contain your name in the filename.

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
- Your ability to explain what you've done and why
- 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 notes

- This assignment is not (usually) accepted late. In the event of a true emergency, your facilitator or the professor can grant an exception, but exceptions are rare. As this assignment is due in the last week of the course, keep in mind that a late submission nearly guarantees you will have to take an Incomplete grade and complete an Incomplete Grade Contract with the professor.
- 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 (e.g., we may select a different 34% split). If the performance obtained from this independent dataset is substantially different from the performance of your best model reported in your documentation, up to 40% will be deducted.