


- Due Jun 7, 2023 by 11:59pm
- Points 100
- Submitting a file upload
- Available until Jun 10, 2023 at 11:59pm

This assignment was locked Jun 10, 2023 at 11:59pm.

A3 Decision Tree and Naïve Cross-validation

Instructions

Packages required: Install Rmarkdown, C50, e1071, caret, rminer, matrixStats and knitr packages.

Input file - CD_additional_modified.csv (<https://utah.instructure.com/courses/882385/files/147060909?wrap=1>) 
(https://utah.instructure.com/courses/882385/files/147060909/download?download_frd=1) (different from the input file in A1 and A2) **and data dictionary - CD_metadata.xlsx** (<https://utah.instructure.com/courses/882385/files/147060897/download?wrap=1>), (the same as the data dictionary in A1 and A2).

The target variable is still y.

Submission: Submit two files – A3_yourLastname_yourFirstname.Rmd, an R Markdown file, and **A3_yourLastname_yourFirstname.html**, HTML output file rendered from rendering A3_yourLastname_yourFirstname.Rmd. **(Please do not compress these two files into one zip file. Submit them as separate files.)**

Note: If there is any discrepancy in the task descriptions in the assignment page and the short rubric descriptions, please follow the task descriptions in the assignment page.

Task I (95 points)

(5 points) Create A3_yourLastname_yourFirstname.Rmd that includes code chunks to meet the requirements specified per chunk below. Use the RMD header to include assignment title, author name – you and the file creation date. Also include header specifications to generate a table of contents and section numbers of your code chunks. I suggest that you set your chunk_output_type to console. (See week 3 tutorials for examples of the RMD header specification.)

Create code chunks to meet the following requirements.

1. (10 points) Code Chunk 1 - Set up, Data import, and Preparation

A. Package loading, and data import. Set the working directory to the directory where your rmarkdown program file resides in rstudio using getwd() and setwd(). For example,

```
mydir <- getwd()
```

```
setwd(mydir)
```

Now that you are familiar with the variables in the input data, feel free to load character variables as factors in read.csv(). Show the overall structure and summary of the data frame that keeps the data from the input file.

B. Partition this data frame for simple hold-out evaluation – 70% for training and the other 30% for testing.

C. Show the distributions (in percentages) of the target variable in the whole input data frame, the train set and the test set.

2. (20 points) Code Chunk 2 - Simple Decision Tree Training and Testing

(Note: Here, you are repeating the code for the same task in Assignment 2 in order to examine the models and performance results for a different data set):

A. (10 points) Train a C5.0 model using the default setting. Show information about this model and the summary of the model. Do not plot the tree at this point because the tree might be too complex. Generate and compare this model's confusion matrices and classification evaluation metrics in testing and training sets

B. (10 points) Explore reducing the tree complexity by lowering CF levels. In the code, select a CF level of your choice to train and test another C5.0 model. Plot the tree. Generate and compare this model's confusion matrices and classification evaluation metrics in testing and training sets

3. (30 points) Code Chunk 3 - Simple Naïve Bayes Model Training and Testing

A. (15 points) Train a naive Bayes model using the training set from 1. Show information about this model. Generate and compare this model's confusion matrices and classification evaluation metrics in testing and training sets

B. (15 points) Explore removing one predictor for building naive Bayes models for this requirement so as to exam the impact of the removal of a predictor. In the code, decide on which predictor to be removed from the data sets for training and testing another naive Bayes model that could improve the true positive rate of the "yes" class of the target variable y . Train and apply this new model. Generate and compare this model's confusion matrices and classification evaluation metrics in testing and training sets

4. (10 points) Code Chunk 4 - Create a Named Cross-validation Function – `cv_function` (you can reuse the `cv_function` in week 3's tutorial, e.g., [CV Titanic Tutorial.Rmd \(https://utah.instructure.com/courses/882385/files/147060889/download?wrap=1\)](https://utah.instructure.com/courses/882385/files/147060889/download?wrap=1). https://utah.instructure.com/courses/882385/files/147060889/download?download_frd=1)

A. This function uses several arguments – a data frame, the target variable, classification algorithm, seed value, the number of folds, and a set of classification metrics (without including confusion matrix output).

B. It generates and displays the overall accuracy, and precision, true positive rate and f-measure of each class of the target variable of the model built for each fold.

C. The function should also generate the mean values and standard deviations of each performance metric over all of the folds.

D. Use `kable()` to show the performance metrics by fold and their mean values and standard deviations.

5. (20 points) Code Chunk 5 - 5-fold and 10-fold C5.0 and naive Bayes evaluation performance with `cv_function` (you can use the `cv_function` from the tutorial code)

A. Use the *data frame that keeps the entire set of input data* to evaluate C5.0 and naive Bayes models by 5-fold as well as 10-fold cross-validation evaluations.

Task II: Reflections (5%)

What have you learned from building each of these models and the modeling impact of your adjustments to the hyperparameters or dataset? If you were explaining the results of these models to a supervisor what would you say about them? Attempt to do more than just state facts here, interpret the results. Coding is great, interpretation of output is even more important. Discuss each model.

Write at least 150 words.

Suggestions for each chunk:

- Add some simple descriptive text in the text area before the code chunk.
- Add a name or description of each code chunk in `{r}`. Be sure that you *allow code and output from executing code to be included in the file from rendering* `A3_yourLastname_yourFirstname.Rmd`.
- Feel free to add comment lines with the requirement item numbers (e.g., # 3.A or # 3.B) to your code cell to help TAs and instructors easily identify your code that addresses a particular requirement.

Render `A3_yourLastname_yourFirstname.Rmd` to HTML output. You can click on the "Knit html" button above the source code pane in rstudio.

Criteria		Ratings		Pts
Task I Rmd header, chunks and formatting.		5 pts Full Marks	0 pts No Marks	5 pts
1.1.A Package loading, and data import. Show the overall structure and summary of the CD_prediction data frame		4 pts Full Marks	0 pts No Marks	4 pts
1.1.B Partition the CD_prediction– 70% for training and the other 30% for testing.		4 pts Full Marks	0 pts No Marks	4 pts
1.1.C Show the distributions (in percentages) of the target variable in the train set and the test set.		2 pts Full Marks	0 pts No Marks	2 pts
2.2.A.i Train a C5.0 model using the default setting. Show information about this model and the summary of the model.		6 pts Full Marks	0 pts No Marks	6 pts
2.2.A.ii Generate and compare this model's confusion matrices and classification evaluation metrics in testing and training sets		4 pts Full Marks	0 pts No Marks	4 pts
2.2.B.i Train a simplified C5.0 model by changing CF and Plot the tree.		6 pts Full Marks	0 pts No Marks	6 pts
2.2.B.ii Generate and compare this model's confusion matrices and classification evaluation metrics in testing and training sets.		4 pts Full Marks	0 pts No Marks	4 pts
2.3.A.i Train a naive Bayes model. Show information about this model.		11 pts Full Marks	0 pts No Marks	11 pts
2.3.A.ii Generate and compare this model's confusion matrices and classification evaluation metrics in testing and training sets.		4 pts Full Marks	0 pts No Marks	4 pts
2.3.B.i Select and remove one predictor to improve the true positive rate of the “yes” class of the target variable y.		6 pts Full Marks	0 pts No Marks	6 pts
2.3.B.ii Train and apply this new model.		5 pts Full Marks	0 pts No Marks	5 pts
2.3.B.iii Generate and compare this model's confusion matrices and classification evaluation metrics in testing and training sets.		4 pts Full Marks	0 pts No Marks	4 pts

Criteria	Ratings		Pts
<p>2.4</p> <p>Create a Named Cross-validation Function – cv_function.</p>	<p>10 pts Full Marks</p>	<p>0 pts No Marks</p>	10 pts
<p>2.5.A.i</p> <p>Use the CD_prediction dataset to evaluate C5.0 model by 5-fold as well as 10-fold cross-validation evaluations.</p>	<p>10 pts Full Marks</p>	<p>0 pts No Marks</p>	10 pts
<p>2.5.A.ii</p> <p>Use the CD_prediction dataset to evaluate naive Bayes models by 5-fold as well as 10-fold cross-validation evaluations.</p>	<p>10 pts Full Marks</p>	<p>0 pts No Marks</p>	10 pts
<p>2.6</p> <p>What have you learned from building each of these models and the modeling impact of your adjustments to the hyperparameters or dataset? If you were explaining the results of these models to a supervisor what would you say about them? Attempt to do more than just state facts here, interpret the results. Coding is great, interpretation of output is even more important. Discuss each model.</p> <p>Write at least 150 words.</p>	<p>5 pts Full Marks</p>	<p>0 pts No Marks</p>	5 pts
Total Points: 100			