

Neural Networks

In this assignment you will complete a variety of tasks related to binary classification with neural networks. The dataset that we will be using is related to criminal justice and deals specifically with parole violations.

Deliverable: All of your work for this assignment should be done in an R Markdown document. Knit your document into a Word file and submit the Word file as the deliverable for this assignment.

Libraries: For this assignment you will need the following libraries: tidyverse, caret, and nnet.

Before beginning the assignment tasks, you should read-in the data for the assignment into a data frame called `parole`. **Carefully** convert the `male`, `race`, `state`, `crime`, `multiple.offenses`, and `violation` variables to factors. Recode (rename) the factor levels of each of these variables according to the description of the variables provided in the `ParoleData.txt` file (located with the assignment on Canvas).

Note: You did this before. I would encourage you to re-use your code.

Task 1: Split the data into training and testing sets. Your training set should have 70% of the data. Use a random number (`set.seed`) of 12345.

Task 2: Create a neural network to predict parole violation. Use a size of 12 (corresponding roughly to the number of variables, including dummy variables) and a decay rate of 0.1. Use `caret` to implement 10-fold k-fold cross-validation. Use a random number seed of 1234. To suppress all of the text describing model convergence, add the command: `trace = FALSE` after `verbose = FALSE`.

Task 3 Use your model from Task 2 to develop predictions on the training set. Use `caret`'s `confusionMatrix` function to evaluate the model quality. Comment on the model quality.

Task 4: Create a neural network to predict parole violation. Use a grid to search sizes 1 through 12 (by 1) and decay rates of 0.1 to 0.5 (by 0.1). Use `caret` to implement 10-fold k-fold cross-validation. Use a random number seed of 1234. To suppress all of the text describing model convergence, add the command: `trace = FALSE` after `verbose = FALSE`. **Note: This model make take some time to run! Be patient, particularly if you are using an older computer.**

Task 5: Use your model from Task 4 to develop predictions on the training set. Use `caret`'s `confusionMatrix` function to evaluate the model quality. Comment on the model quality.

Task 6: Use your model from Task 2 to develop predictions on the testing set. Use the `confusionMatrix` command to assess and comment on the quality of the model.

Task 7: Use your model from Task 4 to develop predictions on the testing set. Use the `confusionMatrix` command to assess and comment on the quality of the model.

Task 8: Comment on whether there appears to be overfitting in one or both of your models from Tasks 2 and 4.