Assignment 5: Predict Recidivism Within 3 Years - Hands-on with Regularization and Ensemble Methods

8 points

April 14, 2025

Abstract

ATTENTION: this assignment should be completed individually. And I will use tools to check your codes against your peers' submissions! This assignment consists of one problem. The instructions are designed to modify the data sequentially. In other words, each step should be performed on the resulting data frame from the previous step.

Predict Recidivism Within 3 Years (8 points)

For this assignment, you will be working with recidivism forecasting challenge dataset made available by National Institute of Justice. You can find the description of variables in the codebook. The goal is to predict "Recidivism_Within_3years" variable using the other variables/features in the data. Download the dataset from **Canvas**.

Data Cleaning & Exploration

- 1. (0.1 points) Read the dataset; Remove the first column, as it is a unique identifier and not used in predicting recidivism; Remove the variables: "Recidivism_Arrest_Year1", "Recidivism_Arrest_Year2", "Recidivism_Arrest_Year3". These variables show whether recidivism occurred in year1, year2, and year3 after arrest.
- 2. (0.1 points) **Take a summary** of the data and explore the result.
- 3. (0.3 points) Which columns have missing values and what percentages of those columns have NAs? **Note: the missing values may be represented by empty strings/values.**
- 4. (0.3 points) Read the codebook carefully. **Based on the data description, how many categorical variables are there in the dataset? Convert these categorical variables to factors.**
- 5. (0.5 points) Read the codebook carefully. **Based on the data description, how many numeric variables are there in the dataset? If any numeric variables are represented as characters, convert them to numeric indices. For example, if the variable "Prior_Arrest_Episodes_Felony" has a value of "10 or more", convert it to 10 by removing the text. Similarly, for "Prison_Years" with values such as "More than 3 years", "Greater than 2 to 3 years", "1-2 years", and "Less than 1 year", convert these to numeric indices like 4, 3, 2, and 1, respectively.**
- 6. (0.1 points) The dataset has a binary variable "Training_Sample" which takes values one or zero if the sample is in the train or test sets, respectively. Split the data to train and test set based on this variable. Then remove this variable.
- 7. (0.5 points) This dataset has some missing values. Read the codebook carefully and decide about what imputation method you want to use. Don't just use a simple mean or mode imputation for all variables. Decide about data imputation based on the description of each variable and any pattern you observe in the missing values. For example, it appears that individuals

without drug tests have missing values for all drug-related variables. In such cases, you can impute the missing drug values with "zero" and create an additional indicator variable, such as "drug_imputed", which is set to "true" if an individual's drug-related variables are missing and imputed, and "false" otherwise. Refer to Chapter 13 of the required textbook "Machine Learning with R", specifically the section on "simple imputation with missing value indicators". **Hint: If you use any statistics (e.g., mean or mode) to impute missing values, make sure they are computed based on the training data only to avoid data leakage.**

Creating a Simple Benchmark

- 8. (0.7 points) Before we jump into building a machine learning model, we need to start with something simpler; that is a heuristic benchmark. Think of this as setting a basic benchmark without using ML, which will help us see if ML is really a better solution. Here's how we can do it for this project:

  We have a variable called the "Supervision_Risk_ Score_First", which is the risk level assigned to someone when they first got parole. We can split this score into three groups: "low", "medium", and "high" risk. For example, scores from 1 to 3 are "low" risk, 4 to 6 are "medium", and anything above or equal to 7 is "high" risk.

  Next, we'll look at our training data and check how many people in each risk group actually went back to committing crimes (i.e., "Recidivism_Within_3years"=true). Let's say in the past, 60% of the people with a "high" risk score ended up committing crimes again. We'll use this same percentage, 60%, as our predicted probability for any new person with a "high" risk score. For example, you can use the one-line "sample" function to do predictions to individuals in the "high" risk group of the test data.

```
predicted_labels[test_set$risk_group == "high"] <- sample(c(TRUE, FALSE),

sum(test_set$risk_group == "high"), prob = c(percentage_high, 1 - percentage_high),

replace = TRUE)
```

  Get the predictions of this benchmark model for the test data. Create a cross table (confusion matrix) of predicted test labels vs true test labels and compute precision, recall, and F1 score (Note: treat "Recidivism_Within_3years=true" as positive).

Training ML Models

After cleaning and exploring data and creating a simple benchmark, we are ready to train machine learning models to predict "Recidivism_Within_3years". We will examine two categories of models: Regularized Logistic Regression and Tree-based Ensemble Models.

Creating Regularized Logistic Regression Models

Hint: check code demo on Regularized and Ensemble Models.

- 9. (0.5 points) set.seed(2025) and train a Lasso Logistic Regression model using "glmnet" and "caret" as explained in the code demo lectures to predict the "Recidivism_Within_3years". Use 5-fold cross validation and tune the lambda parameter. Note: You do not need to worry about scaling your train or test data, "glmnet" will automatically do it for you.
- 10. (0.5 points) Get the coefficients for the best tuned model in Q9. Did Lasso shrink some of the coefficients to zero? If so, what does this mean?
- 11. (0.5 points) set.seed(2025) again and train a Ridge Logistic Regression model using 5-fold cross validation and tune lambda as you did for Lasso in Q9.
- 12. (0.5 points) set.seed(2025) again and train an Elastic Net Logistic Regression model using 5-fold cross validation and tune lambda and alpha.

Creating Tree-based Ensemble Models

Hint: check code demo on Regularized and Ensemble Models.

- 13. (0.5 point) set.seed(2025) and use "caret" package with "rf" method to train a random forest model (version 2) on the training data to predict "Recidivism_Within_3years". Use 5-fold cross validation and let caret auto-tune the model. Auto-tune means that you do not need to specify the "tuneGrid" like what you did in the Regularized Logistic Regression Models and "caret" automatically selects the optimal hyperparameters for the model by evaluating different configurations during the training process using cross-validation. (Note: use importance=T in your train method so it computes the variable importance while building the model). Be patient. This model may take a long time to train.
- 14. (0.4 points) Use caret's "varImp" function to get the variable importance for the random forest model you got in Q13. Which variables were the most predictive in the random forest model?
- 15. (0.5 point) set.seed(2025) and use "caret" package with "gbm" method to train a Gradient Boosted Tree model on the training data. Use 5-fold cross validation and let "caret" auto-tune the model.

Compare all the models

- 16. (0.5 points) "resamples" method gives a distribution of the performance measures across folds in cross validation for each tuned model. Use "resamples" method to compare the cross validation metrics of the five models you created above (LASSO, RIDGE, elastic net, random-forest, and gbm). Which models have better cross validation performance? **In a sentence or two, interpret the results**.
- 17. (0.5 points) Test all the five models on the test set, compute their precisions, recalls, and F1 scores. Compare them to the heuristic benchmark you created in Q8, do they perform better than the heuristic benchmark? Why or why not?

Responsible AI Discussion

For the following questions, consider the following hypothetical: The city of Summerfield has deployed a machine learning model powered by a K Nearest Neighbor classifier to predict recidivism for individuals leaving the prison system. This model uses features such as age, criminal history, employment status, and socio-economic background, and is trained on historical recidivism data.

- 18. (0.5 points) A KNN classifier typically makes predictions by majority vote – it looks at the k nearest neighbors and takes the most common class.

  How might relying on historical recidivism data in building this model perpetuate existing biases against marginalized groups? What long-term societal impacts could the widespread use of such a predictive system have on individuals from these groups?

- 19. (0.5 points) Assume preliminary analysis shows that the recidivism prediction model has improved the efficiency of parole decisions, leading to fewer repeat offenses. However, civil rights groups argue that the model lacks fairness and has the potential to reinforce systemic biases against certain demographics.

  Discuss a trade-off between reducing recidivism rates and ensuring fairness or equity. How might you analyze data or conduct an impact assessment to understand whether this model's benefits outweigh its costs with respect to this trade-off?

The submission must be in these formats

- A html file; You run all the code cells, get all the intermediate results and formalize your answers/analysis, then you click "preview" and this will create a html file in the same directory as your notebook. **You must submit this html file or your submission will not be graded.**

- An Rmd file which contains your R notebook.