Report on NIJ Recidivism Forecasting Challenge

Jianye Ge
University of North Texas Health Science Center
Jianye.Ge@unthsc.edu

1. Introduction

The National Institute of Justice (NIJ) recently launched a Recidivism Forecasting Challenge that "aims to increase public safety and improve the fair administration of justice across the United States." Participants were challenged to "forecast recidivism using person- and place-based variables with the goal of improving outcomes for those serving a community supervision sentence." The data used in this challenge were from "the State of Georgia about persons released from prison to parole supervision for the period January 1, 2013 through December 31, 2015." The contestants were expected to submit forecasted likelihoods of whether individuals in the dataset recidivated within one year, two years, or three years after release.

I submitted the forecast likelihoods as a small team and was one of the winners in Year 1. This paper described how I analyzed the data in this challenge.

2. Methods

2.1. Data and variables

The data provided by NIJ included both training and test datasets. For Year 1, there are 32 variables and one binary class variable (i.e., Recidivism_Arrest_Year1). The training set has 18,028 samples, with 15,811 males and 2,217 females. The test set has 7,807 samples, with 6,857 males and 950 females.

The InfoGainAttributeEval class in WEKA version 3.8.5 [1, 2] was used to rank the variables in terms of Information Gain with respect to the class variable (i.e., Recidivism_Arrest_Year1). InfoGain(Class, Variable) = H(Class) - H(Class|Variable). Both male and female samples in the training set were used in the evaluation. Table 1 shows the InfoGain of the variables. The top 11 variables in Table 1 are the most informative variables (InfoGain>0.008), but the rest variables may still contribute to the class variable to a certain degree.

Table 1. The variables ranked by InfoGain.

InfoGain	Variable
0.021459	Prior_Arrest_Episodes_PPViolationCharges
0.020346	Prior_Arrest_Episodes_Felony
0.017596	Prior_Arrest_Episodes_Property
0.01505	Age_at_Release
0.0149	Supervision_Risk_Score_First
0.013349	Gang_Affiliated
0.012965	Prior_Arrest_Episodes_Misd
0.012561	Prior_Conviction_Episodes_Prop
0.012087	Prior_Conviction_Episodes_Misd

This resource was prepared by the author(s) using Federal funds provided by the U.S. Department of Justice. Opinions or points of view expressed are those of the author(s) and do not necessarily reflect the official position or policies of the U.S. Department of Justice.

0.009	Prison_Offense
0.008121	Prison_Years
0.004795	Condition_MH_SA
0.00463	Prior_Conviction_Episodes_Felony
0.004372	Education_Level
0.004363	Gender
0.003681	Supervision_Level_First
0.003541	Prior_Conviction_Episodes_PPViolationCharges
0.001992	Prior_Arrest_Episodes_Violent
0.001836	Prior_Arrest_Episodes_Drug
0.001703	Prior_Arrest_Episodes_DVCharges
0.001632	Prior_Revocations_Parole
0.001471	Dependents
0.001444	Prior_Conviction_Episodes_Viol
0.001388	Condition_Cog_Ed
0.001232	Residence_PUMA
0.001088	Prior_Conviction_Episodes_DomesticViolenceCharges
0.000908	Prior_Revocations_Probation
0.000731	Prior_Conviction_Episodes_Drug
0.000647	Race
0.000443	Prior_Arrest_Episodes_GunCharges
0.000219	Prior_Conviction_Episodes_GunCharges
5.98E-05	Condition_Other

2.2. Models, methods, and codes

All classifiers in WEKA were tested with the training dataset by 5 folder cross-validation (i.e., randomly picking 80% of the samples for training and using the rest 20% as test samples). LibLinear [3], LibSVM[4], and XGBoost4J [5, 6] were also used. The goal was to find the best-performing classifier. Logistic, LMT, MultiClassClassifier, and ClassificationViaRegression were the best performing classifiers in terms of Brier Score. However, sampling different subsets from the training data provided a slightly different ranking of these classifiers, which means they performed similarly for the training dataset. The missing data were automatically imputed by each classifier. All variables were used in training.

On average, these classifiers had accuracies from 69.5% to 70.2% and Brier Score from 0.187 to 0.189. ClassificationViaRegression is slightly more stable than other classifiers in terms of Brier Score (i.e., with the minimum variance) (Table 1), and thus was used in the final analysis.

Table 1. The averages and variances of the Brier Score for the top-performing classifiers.

	Logistic	LMT	ClassificationViaRegression	MultiClassClassifier
Average	0.1879	0.1894	0.1880	0.1875
Variance	7.12E-06	1.20E-05	7.06E-06	1.17E-05

This resource was prepared by the author(s) using Federal funds provided by the U.S. Department of Justice. Opinions or points of view expressed are those of the author(s) and do not necessarily reflect the official position or policies of the U.S. Department of Justice.

In ClassificationViaRegression [7] applies a regression scheme into classification. First, a dataset is derived into multiple datasets by binarizing the class (one class value for one dataset). In each dataset (or class value), the class value was assigned as 1, if this instance has this particular class value, or 0 for any other class values. Regression is then performed for each derived dataset (or class value). The classification is determined with the maximum output of the regression models of these datasets. Because this NIJ Recidivism only has two classes, the ClassificationViaRegression would perform similarly as logistic regression. The cross-validation results did confirm they had similar Brier Scores. LMT also uses a regression model in a classification tree. For a two-class dataset, as expected, the classification performance of LMT was similar to that of the logistic regression.

The following are the JAVA codes used in converting data formats, training classifier with the training dataset, and predicting the likelihoods of the samples in the test dataset.

```
//convert CSV file to ARFF file as standard input of WEKA package
CSVLoader loader = new CSVLoader();
loader.setSource(new File("TrainingYear1.csv"));));
Instances trainingdata = loader.getDataSet();
DataSink.write("TrainingYear1.arff", trainingdata);

//train classifier with the training dataset
Instances trainingdata = DataSource.read("TrainingYear1.arff");
trainingdata.setClassIndex(trainingdata.numAttributes()-1);
Classifier classifier = new ClassificationViaRegression();
classifier.buildClassifier(trainingdata);

//likelihood prediction with the test dataset
Instances testdata = DataSource.read("TestYear1.arff");
for(int i=0;i<testdata.numInstances();i++) {
        double[] dist =
                classifier.distributionForInstance(testdata.get(i));
        out.write(String.valueOf(dist[1]));
}
```

2.3. Other attempted methods

I also tried multiple ways to improve the overall performance, such as changing the classifier threshold from 0.5 to other numbers (e.g., 0.55), replicating more true cases to balance the number of false and true cases, removing certain false cases that are very close to the true cases, binarizing and normalizing the variables, and removing low InfoGain variables. Unfortunately, none of them had any substantial improvement to the Brier Scores.

3. Questions and answers

3.1. Were variables added to the data set? If so, detail the variables.
Answer: No. All samples were anonymized. It would be challenging to add variables to the existing data without knowing the identities.

3.2. What variables were constructed? How were the variables constructed?
Answer: All variables were used in training and testing, because removing variables with low InfoGain did not have substantial improvement.

3.3. Which variables were statistically significant?
Answer: As shown in Table 1, the variables were ranked by InfoGain. The top 11 variables had substantially higher InfoGains (i.e., >0.008) than the other variables.

3.4. What variables were not statistically significant? How was this handled? For example, were they dropped from the overall model?
Answer: As shown in Table 1, there are 21 variables with InfoGain values <0.005. I kept all variables in training, because with or without the low InfoGain variables did not substantial difference in terms of performance during cross-validation.

3.5. What type of model was used?
Answer: ClassificationViaRegression in WEKA was used.

3.6. Did you try other models? Were they close in performance? Not at all close?
Answer: All classifiers in WEKA [1, 2], LibLinear [3], LibSVM[4], and XGBoost4J [5, 6]. Many classifiers were very close to each other in terms of Brier Score. Sampling different subsets of training data could have different best-performing classifiers. In general, the best performing classifiers include ClassificationViaRegression, LMT, MultiClassClassifer, and Logistic. The ClassificationViaRegression was picked because it was slightly more stable than other classifiers in terms of Brier Score (~0.189).

3.7. What other evaluation metrics should have been considered/used for this Challenge? For example, using false negatives in the penalty function.
Answer: It would depend on the purposes or applications of analyzing these data. Minimizing false negatives and false positives are both meaningful in certain scenarios.

3.8. Did the 0.5 threshold affect anything? Would your team recommend a different threshold?
Answer: I did try other thresholds to classify the cases, such as 0.45, 0.55, etc. The Brier Scores were not significantly changed, or sometimes even worse. The classifiers were able to properly determine the probabilities for the samples. I would not recommend a different threshold.

3.9. Did the fact that the fairness penalty only considered false positives affect your submission?
Answer: No. Brier Score is the most important measure. I did my best to reduce Brier Score. False positive rate is related to Brier Score in certain degree.

3.10. Are there practical/applied findings that could help the field based on your work? If yes, what are they?
Answer: Through the data analysis, I found that 1) race is not a significant variable to forecast recidivism. 2) many false cases might actually be true cases, as a large proportion of the false and true cases were close to together. In reality, if an individual committed a crime, but he/she was not caught, he/she would be recorded as a false case. If there are too many such cases, the classification or prediction from the data might be less convincing.

3.11. What should NIJ have considered changing (other than metrics) to improve this Challenge?
Answer: I did not know if the methods I used and my predictions were better or worse than the other teams until the end of July. It would be great if NIJ could release the results (including ranking of the teams) of Phase 1 before all teams work on the next phase. The ranking could be anonymized by assigning IDs to the teams.

3.12. For future Challenges, what should NIJ consider changing to improve Challenges? For example, more/less time, different topic, or data issues (missing data)?
Answer: I will be interested in different topics. I am mostly working in the field of DNA forensics. Some hot topics in DNA forensics would attract lots of attention in the field and would also encourage technological advancement and applications. For example, DNA mixture is one of the most challenging problems in DNA forensics. NIJ may provide mimic DNA mixture samples to the contestants to use any wet-lab technologies and computational methods to recover the true DNA genotypes of the contributors. Investigative genetics genealogy is another hot topic. NIJ may also provide DNA samples from two individuals and ask contestants to determine the relationship between these two individuals by any available technologies, both wet-lab and computational tools.

4. Conclusion and future considerations

This NIJ challenge attracted a large number of teams. It was a fairly close contest. I was fortunate to be one of the winners in Year 1. I should have separated the males and females in predictions, as the recidivism patterns for males and females could be different and this challenge did determine the winners for the males and female Parolees.

As a professor, I would like to encourage more students to join challenges in the future. It would be a great opportunity for the next generation of talents to understand and learn computational skills through solving real-world problems. I hope NIJ will have more similar challenges in the future.

References:

1. Witten IH, Frank E, Hall MA. Data Mining: Practical machine learning tools and techniques, third edition. Morgan Kaufmann. 2011.

2. Eibe F, Hall MA, Witten IH. The WEKA workbench. Online appendix for data mining: practical machine learning tools and techniques. Morgan Kaufmann2016.
3. Fan R-E, Chang K-W, Hsieh C-J, Wang X-R, Lin C-J. LIBLINEAR: A library for large linear classification. the Journal of machine Learning research. 2008;9:1871-4.
4. Chang C-C, Lin C-J. LIBSVM: a library for support vector machines. ACM transactions on intelligent systems and technology (TIST). 2011;2(3):1-27.
5. Chen T, Guestrin C, editors. Xgboost: A scalable tree boosting system. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining; 2016.
6. XGBoost JVM Package, https://xgboost.readthedocs.io/en/latest/jvm/index.html.
7. Frank E, Wang Y, Inglis S, Holmes G, Witten IH. Using model trees for classification. Machine learning. 1998;32(1):63-76.