An Examination of an Automated Inmate Classification System Using RIPPER

Ronnie Johnson

Abstract

This project is based on a data set provided by the California Department of Corrections (CDC) on the effectiveness of prisoner placement, and the likelihood of prisoner misconduct while incarcerated. The problem of prisoner placement is important because of the high cost of operating high-security prison facilities and the limited space in these facilities. Additionally, proper placement of a prisoner within the prison population decreases potential misconduct within a facility thereby creating a safer environment for the staff and the inmates.

This project will use the CDC data set and the inductive rule system called RIPPER to examine inmate misconduct using classification score and security level as predictors.

Introduction

The nation's inmate population as of June 30, 2003 was an astonishing 2,078,570 including Federal, State, or local incarceration.[6] This enormous incarcerated population represents 1 in every 140 people in the United States or roughly 727 prisoners per 100,000 residents. No other country on the planet is known to have an incarceration rate equal to the United States. For example, most European countries incarcerate fewer than 100 prisoners per 100,000 residents.[10] It can be argued that the cause of this astronomically high incarceration rate is the result of changes in the sentencing rules imposed on Federal and State courts, 3-strikes laws, more stringent parole policies, and longer prison terms.

Inmate management in such an overcrowded environment becomes tantamount. Optimization of facility capacity, safety of staff and inmates, and cost per day are all affected by inmate housing decisions. Most prison systems and some jails have inmate classification systems which divide inmates into groups based on security risk. Security at the institutional level is divided into minimum, medium, and maximum security facilities. Inmates are assigned to these facilities based on many factors such as the severity of their offense, length of sentence, incarceration history, and personal history. Within the assigned facilities inmates are further subdivided based on housing placement and program needs. The goal of the initial classification of an inmate is to ensure proper placement for programming and security. Reclassification of inmates is conducted based on the inmate's behavior to determine subsequent classification decisions. Ultimately, the goal of the classification system in any prison system is to reduce violence, limit security risks, and facilitate rehabilitation of the inmate.

Whatever the cause of the increases, the cost of dealing with the population explosion within the prison system is also increasing. The average cost per day for incarceration of a prisoner is $68. National expenditures for incarceration exceed $57 billion per year in 2001.[6] In California it costs $30,929 per year to incarcerate each of their 163,500 inmates. California operates 32 prison facilities ranging from minimum to maximum security which accounted for $5.7 billion in the state's 2003-2004 budget.

In California, as in other large prison systems, inmates are introduced to the system at a "reception center". The reception center screens inmates for program needs and security

risks. The reception center uses an objective and uniform inmate classification system based on a "score sheet" (CDC 839 for new inmates, see appendix). Using the score sheet inmates are awarded "points" for various factors associated with their offense and personal histories. Calculation of an inmate's "classification score" is simply a linear combination of the items on the score sheet with sentence length representing approximately 70% of the variance.[1] Security levels are represented in the table below:

Points	Security Level	Facility
0 to 18	Level I	Minimum
19 to 27	Level II	Medium
28 to 51	Level III	Medium
52 and Above	Level IV	Maximum

Proper determination of inmate security levels directly impacts facility placement and, therefore, cost of incarceration.

The goal of this study is to examine an automated approach for inmate population security classification using the inductive rule-based system known as RIPPER. RIPPER is applied against data collected from the California Department of Corrections representing approximately 4,000 inmate classification questionnaire results. The data include incidents of inmate misconduct which provides a basis for testing the effectiveness of the security classification in meeting its goal.

Related Past Work

Significant work in this area has been and is being conducted albeit using statistical approaches. Some of this work has been conducted by Alexander and Austin, Cowles and Gransky, and Proctor to name but a few. It is believed that a computational data mining approach is uncommon.

Method

Previous studies have shown that it is difficult to identify a classification algorithm that would perform well on all tasks.[11] Some classification algorithms may perform quite well in general but, they may be easily surpassed by other algorithms in specific circumstances. It is then desirable to take the characteristics of the task into account when attempting to identify a suitable algorithm.[4] As described in [12] the goal is to characterize the domain within which the individual classification algorithms achieve "positive generalization performance". The RIPPER classification system was chosen for its utility as well as its documented positive generalized performance characteristics.

Rule-based classification systems order data based on collections of "if…then" rules. The general form of the rules is given below:

Rule: (*Condition*) $\rightarrow y$

- where *Condition* is a conjunction(s) of attributes and *y* is the class label
- *LHS*: rule antecedent or condition
- *RHS*: rule consequent
- Examples of classification rules:
  - (Assault on Inmate=Yes) $\wedge$ (Weapon=Yes) $\rightarrow$ Level IV = Yes
  - (Assault on Staff = Yes) $\rightarrow$ Level IV = Yes

A rule *r* is said to cover an instance **x** if the attributes of the instance satisfy the condition of the rule.

RIPPER is a system for inducing classification rules from a set of preclassified examples; as such it is broadly similar to learning methods such as neural networks, nearest neighbor, and decision trees.[13] To simplify the description of the process, the user provides RIPPER with a set of examples labeled with the appropriate class. RIPPER will examine the examples to construct a set of rules that will predict the class of later examples. As described in [13] RIPPER has several advantages over other learning systems. Of primary importance to this study was the utility of use provided by RIPPER in its representation of rules in an if…then form. The speed of the algorithm was not a concern as the data set to which the system was applied is small nor was the use of constraints. However, the ability to represent attributes as nominal, continuous, or "set-valued" within RIPPER was important because the CDC score sheet value is continuous.

Summary data was used to represent the CDC classification data rather than individual attributes represented in the CDC score sheet. Factors including availability of raw data and resource constraints led to the decision to use summary data. A description of the methodology used to summarize the CDC data is beyond the scope of this paper (the reader is directed to [1] for a description of this process). The summary data contains five attributes represented below:

RESPONSE.....Misconduct Violation (1) or not (0)
SCORE.....Classification Score
STRIKE 2…....Two Striker Inmate (1) or not (0)
STRIKE 3…....Three Striker Inmate (1) or not (0)
TREAT.....Classified to Level 4 (1) or not (0)

Security levels I to III were combined in the data set to yield two classes; Level IV or Not Level IV. As described in [1] this combination of the lower security levels was based on the determination made in previous research of the lack of impact on results in addressing the two main questions posed in their study, namely:

1) How effective is the current classification system at sorting inmates by their potential for misconduct?
2) How effective is the current classification system at controlling inmate misconduct.

The data was provided in a format that required very little preprocessing. The original study data from [1] used a space delimited, column representation for each of the attributes listed above. The data was transformed using the Preprocessor program included in the appendix. The Preprocessor transformed the Treat attribute from 0 (false) and 1 (true) to N and Y, respectively as required by RIPPER. Limited other data manipulation included adding line terminators and comma separators, in general, the data, as obtained from Dr. Berk and Dr. de LEEUW, was "ready-to-use".

Discussion

Rule induction systems are a tool in the formation of high-level cognition of information. High-level cognition determines patterns in information, finding categories and casual relations between the data elements; however, for any finite data set there is an infinite possible combination of data elements.[15] In addressing this issue one should remember the principle of parsimony eloquently espoused in Ockham's razor: "one should not increase, beyond what is necessary, the number of entities required to explain anything" (William of Ockham, 1285 – 1349). The power of any induction system is derived from its ability to develop a representation of the patterns that is easily understood and accurate from the infinite set of possible patterns.

Measures of cognitive simplicity include proposals from Ernst Mach in [16] "simplicity of a description is measured by its length", Isaac Newton in the *Principia* "admit no more causes of natural things than are both true and sufficient to explain the appearances", the Gestalt laws, and Shepard's Universal Law of generalization. Cognitive simplicity is a key goal of any learning system.

The RIPPER algorithm produces a set of rules derived from the training data that can be easily decoded by a human reader. Further, the RIPPER algorithm ensures during its rule postprocessing phase that the induced rules are of minimal length and redundancy. An example of an induced rule developed by the RIPPER algorithm using the CDC training data is given below:

```
Y :- Score>=48, Score>=52, Score>=61, Strike3='0', Response='1',
Strike2='0', Score<=63
```

This rule classifies an inmate to Level IV (maximum security) if the inmate's Classification Score Sheet (CDC 839) score is greater than 48 but less than 63, and the inmate has had an infraction (Response attribute), and the inmate is neither a 2^{nd} or 3^{rd} strike offender (Strike2 and Strike3 attributes, respectively).

Experiments

The primary experiment was the parameterization of the training set size and measurement of the effectiveness of the system as applied to the remaining data. Metrics for evaluation of system performance include execution time, complexity, and test data

error rate. Another goal is to determine the optimal training set size to meet performance goals.

The training data set was generated from the complete CDC data set using the Preprocessor tool. The Preprocessor tool (pseudo-) randomly selected data elements from the 3,919 elements in the CDC data set to create a training data set. The balance of the elements was placed in the testing data set.

Experimental results are summarized in Table 1 below. As the training set size was parameterized for the RIPPER system the learning time was not appreciably affected. With 6 elements in the training set RIPPER required an indistinguishable amount of time (reported by the system as 0.0) for learning. In comparison, when the entire training set of 3,919 elements was past to RIPPER the system required only 0.4 seconds to develop its hypotheses. Learning time is represented in Chart 1 below. As noted above learning time was not considered a factor in this study. The results achieved are consistent with the claims of the RIPPER developers in [13].

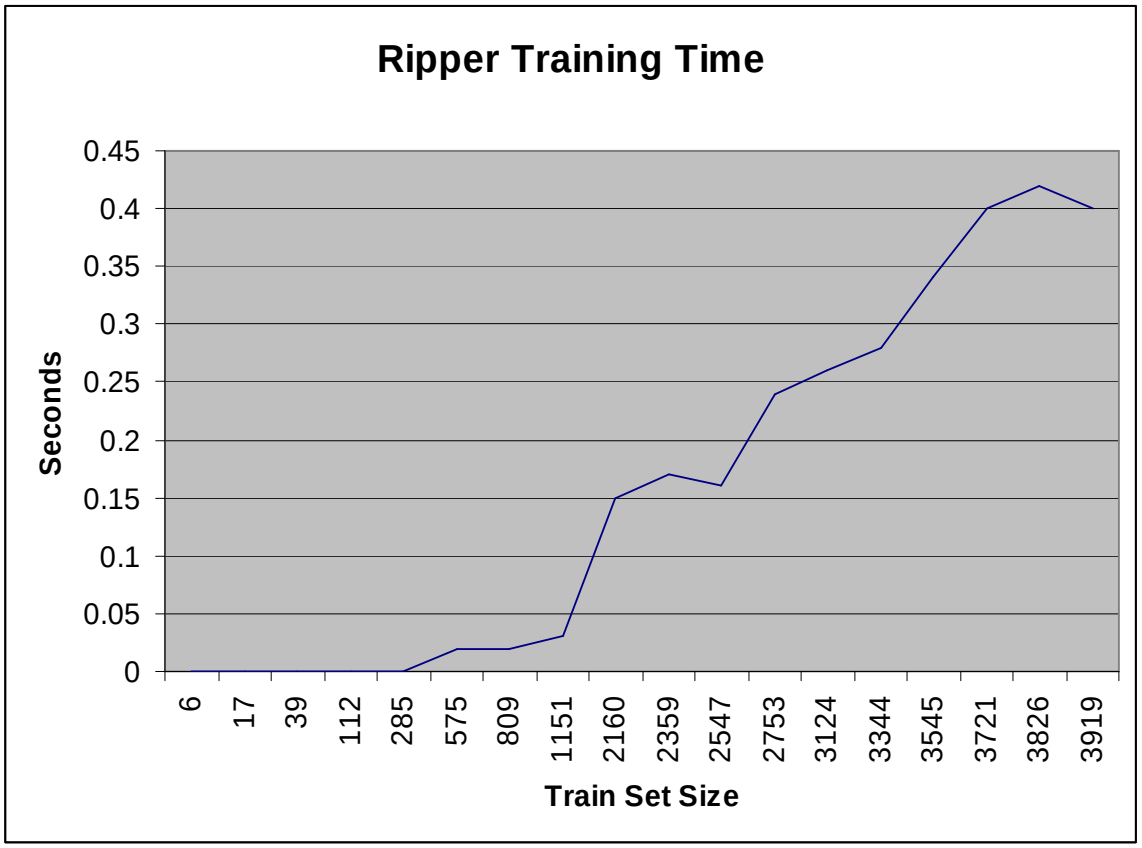

Chart 1 RIPPER Training Time.

CDC Experimental Results																		
Total Number of Elements	3919																	
Number of Training Elements	6	17	39	112	285	575	809	1151	2160	2359	2547	2753	3124	3344	3545	3721	3826	3919
Number of Testing Elements	3913	3902	3880	3807	3634	3344	3110	2768	1759	1560	1372	1166	795	575	374	198	93	0
Number of Rules	1	1	2	3	10	15	15	14	32	34	29	38	37	36	41	45	47	43
Number of Conditions	2	2	6	8	50	79	66	77	179	188	162	223	217	221	249	278	288	265
Testing Error Rate	15.41	2.82	3.14	4.2	5.26	5.47	4.02	4.3	4.15	3.85	5.47	3.34	4.78	4	3.48	1.01	4.3	0
Testing Error Variance	0.58	0.27	0.28	0.33	0.37	0.39	0.35	0.39	0.48	0.49	0.61	0.53	0.76	0.82	0.95	0.71	2.12	0
Training Error Rate	0	0	0	0	2.46	2.96	2.35	3.3	2.27	2.92	3.42	3.38	3.04	3.26	2.99	3.01	3.03	2.91
Training Error Variance	0	0	0	0	0.92	0.71	0.53	0.53	0.32	0.35	0.36	0.34	0.31	0.31	0.29	0.28	0.28	0.27
Learning Time	0	0	0	0	0	0.02	0.02	0.03	0.15	0.17	0.16	0.24	0.26	0.28	0.34	0.4	0.42	0.4

Table 1 Summary of Experiment Results.

Average Number of Rules			25
Average Number of Conditions			142
Average Testing Error Rate			4.39
Average Learning Time			0.16

Table 5 Average Experimental Results.

A key metric in the study was the effectiveness of RIPPER in classifying the inmate data. One measure of this effectiveness is the complexity of the hypothesis developed by the system. The primary measure of complexity is the number of rules and conditions within those rules generated by the system. The minimum number of rules / conditions generated by RIPPER was 1 / 2 and the maximum was 47 / 288. The complexity results are represented in Chart 2.

The accuracy of the rule system developed by RIPPER based on the parameterization of the training set size is represented in Chart 3. These results demonstrate that increases in training set size did not appreciably affect the accuracy of the system. In this case there is no significant increase in the effectiveness of the system against the testing data as the training set size and the complexity of the system is increased.

An issue of interest was detected when the training set was increased from 6 data elements to 17 elements. The complexity is unchanged in both scenarios: 1 rule and 2 conditions. The learning time is negligible in both cases, reported as 0.0 by the system. The accuracy of the system trained with 6 data elements is 15.41%; however, the accuracy for the system trained with 17 data elements is a mere 2.82%. In this case complexity did not play a role in the accuracy. The rules for these scenarios are given below:

Scenario 1 - 6 data elements in the training set.
Rule:
If Classification score => 77 then
Level IV incarceration = Yes
Else
Level IV incarceration = No

Scenario 2 - 17 data elements in the training set.
Rule:
If Classification score => 52 then
Level IV incarceration = Yes
Else
Level IV incarceration = No

Data representing further investigation of this issue is summarized in Table 2 below:

CDC Experimental Results					
Total Number of Elements	3919				
Number of Training Elements	18	16	16	6	6
Number of Testing Elements	3901	3903	3903	3913	3913
Number of Rules	1	1	1	1	1
Number of Conditions	2	2	2	2	2
Testing Error Rate	3.61	12.02	3.61	4.14	6.21
Testing Error Variance	0.30	0.52	0.30	0.32	0.39
Training Error Rate	0.00	0.00	0.00	0.00	0.00
Training Error Variance	0.00	0.00	0.00	0.00	0.00
Learning Time	0.00	0.00	0.00	0.00	0.00
Classification Score	54.00	70.00	54.00	48.00	60.00

Table 2.

As would be expected with small training sets the results can easily be skewed by selecting training elements that are not representative of the entire data set. This is clearly the case in the results shown above where the key to the accuracy of the system is directly related to the Classification score (last row of the table) selected as the antecedent value in the rule.

The system appears to have reached an acceptable stasis point when using approximately ¼ to 1/3 of the available data for training. The average error rate for the system is 4.39%. The error rate for experimental results in the ¼ to 1/3 is 4.3%. The results of further experimentation in this range are represented in table 3. The averages of the results in table 3 are represented in table 4. These results are consistent with the above expectation of a local optimal solution.

CDC Experimental Results										
Total Number of Elements	3919									
Number of Training Elements	1177	1156	1097	1231	1267	1006	1085	1169	1241	1293
Number of Testing Elements	2742	2763	2822	2688	2652	2913	2834	2750	2678	2626
Number of Rules	15	20	21	17	18	19	17	15	22	26
Number of Conditions	83	106	109	84	104	109	95	75	119	134
Testing Error Rate	3.68	3.15	3.72	4.13	6.26	4.63	3.88	4.28	4.07	4.07
Testing Error Variance	0.36	0.33	0.36	0.38	0.47	0.39	0.36	0.39	0.38	0.39
Training Error Rate	3.14	3.03	2.92	2.11	4.10	3.28	2.67	1.97	2.58	2.09
Training Error Variance	0.51	0.50	0.51	0.41	0.56	0.56	0.49	0.45	0.45	0.40
Learning Time	0.03	0.04	0.05	0.04	0.05	0.04	0.04	0.03	0.05	0.07

Table 3.

Average Number of Rules	19
Average Number of Conditions	102
Average Testing Error Rate	4.19
Average Learning Time	0.04

Table 4.

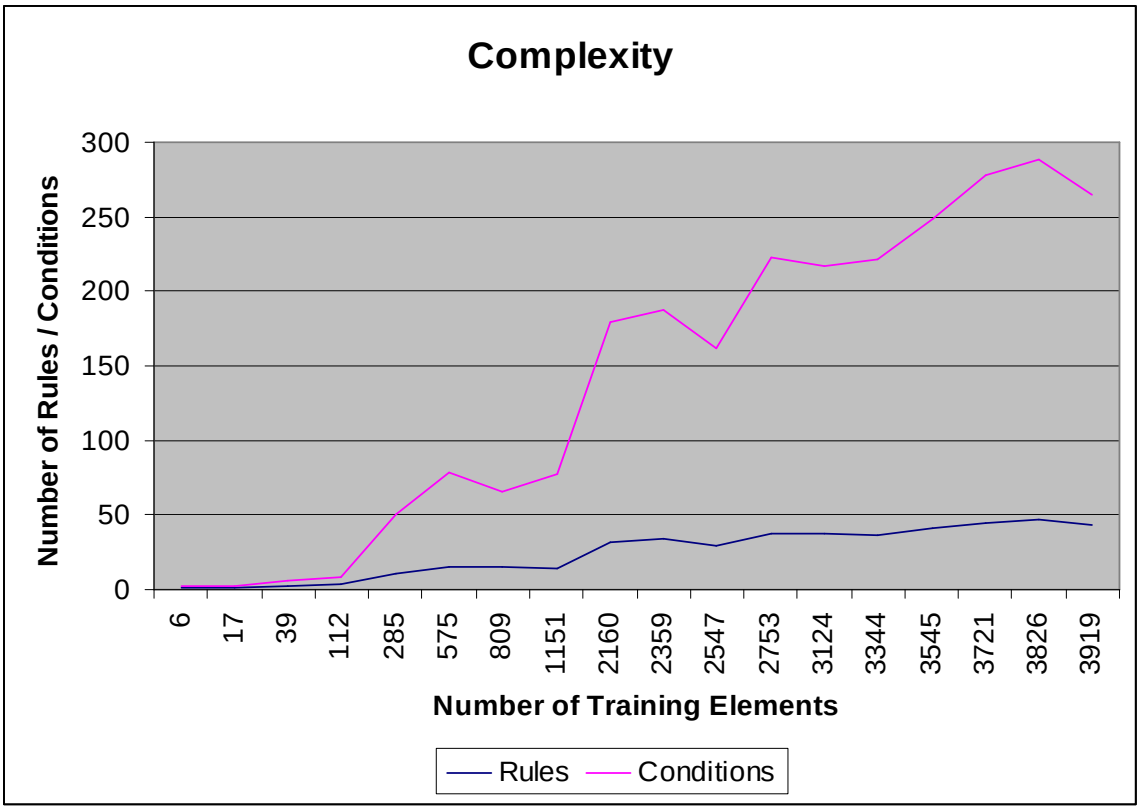

Chart 2 Complexity.

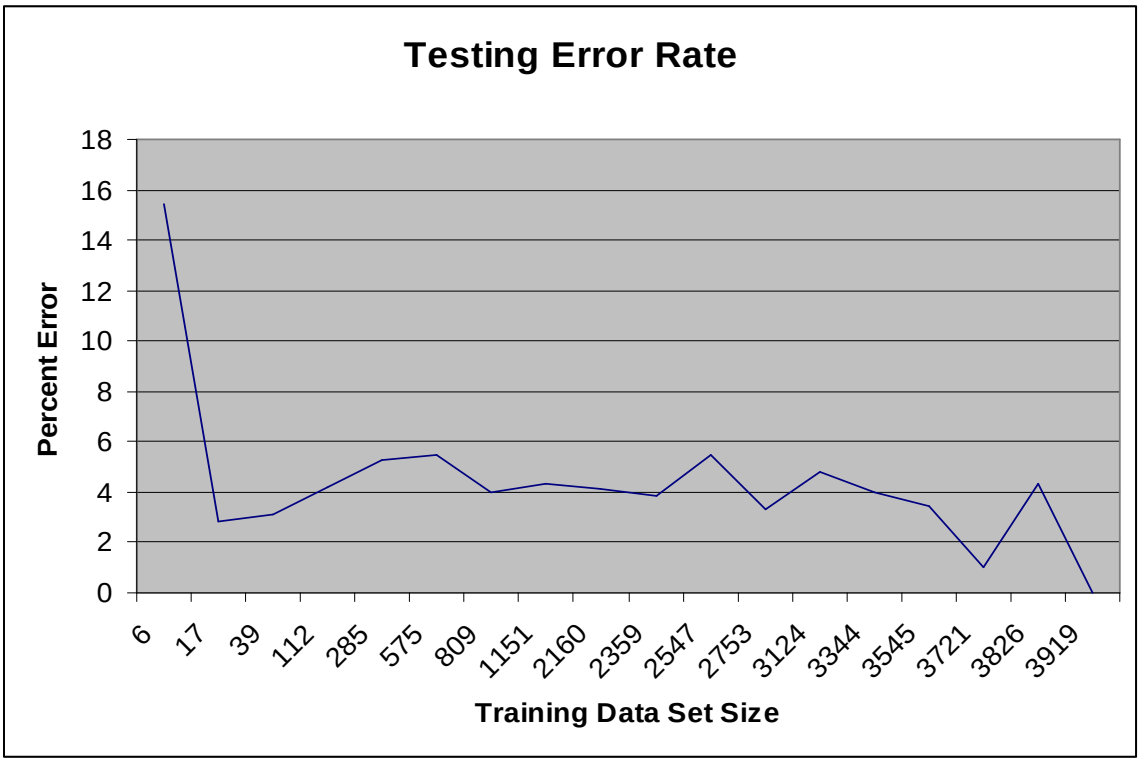

Chart 3 Testing Error Rate.

Conclusion

Objective prison classification systems that decide which facility an inmate should be housed in are well-established in virtually every state correctional system. Despite considerable progress in the area of classification, many prison systems are under significant pressure to review and update their institutional classification systems in response to changes and pressures associated with truth-in-sentencing and three-strikes-and-you're-out laws, tremendous growth and diversity of the correctional populations, overcrowding, and public sentiment against programs and services.[8]

Using RIPPER this study was able to classify approximately 3,000 inmates with 96% accuracy. In addition to the accuracy of the model, the rules generated are simple and easy to comprehend making the accuracy measure meaningful to the user.

Future studies of interest in this area might include comparisons using the entire attribute set from the CDC 839 encoding with other classification algorithms such as C4.5 and AQ21.

Acknowledgements

Mr. Jim Short, California Department of Corrections, for his invaluable assistance in acquiring data and information for this study.

Dr. Berk and Dr. de LEEUW of the Department of Statistics at the University of California, Los Angeles for making their data set available for use.

References

[1] Berk, Richard and Leeuw, Jan de; "An Evaluation of California's Inmate Classification System Using a Generalized Regression Discontinuity Design"; Journal of the American Statistical Association, Dec. 1999, Vol. 94, No.448, Applications and Case Studies.

[2] Austin, J.; "How Well is your Classification System Operating: A Practical Approach"; Crime and Delinquency, 1986, Vol. 32.

[3] Buchanan, R.A., Whitlow, K.L., and Austin, J.; "National Evaluation of Objective Prison Classification Systems: The Current State of the Art"; Crime and Delinquency, 1986, Vol. 32.

[4] Gama, J. and Brazdil, P.; "Characterization of Classification Algorithms"; LIACC, University of Porto, Portugal.

[5] Kumar, V.; "Data Mining"; Course Notes from Army High Performance Computing Research Center Department of Computer Science University of Minnesota; 2002.

[6] Bureau of Justice Statistics, United States Department of Justice; "Bureau of Justice Statistics Correctional Surveys"

[7] Weinstein, C. and Cummins, E.; "The Crime of Punishment: Pelican Bay Maximum Security Prison"; from the book Criminal Justice; South End Press; 1996.

[8] Hardyman, P. and Austin, J. and Tulloch, O.; "Revalidating External Prison Classification Systems: The Experience of Ten States and Model for Classification Reform"; The Institute on Crime, Justice and Corrections at The George Washington University; January 2002.

[9] Hardyman, P. and Austin, J.; "Objective Prison Classification: A Guide for Correctional Agencies"; The Institute on Crime, Justice and Corrections at The George Washington University; July 2004.

[10] Kuhn, A.; "Incarceration Rates Across the World"; *Overcrowded Times*; Vol. 10; April 1999.

[11] Michie, D. and Spiegelhalter, D.J. and Taylor, C.; Machine Learning, Neural and Statistical Classification, Ellis Horwood, 1994.

[12] Shaffer, C.; "Selecting Classification Method by Cross- Validation", *Machine Learning,* Vol. 13; Kluwer Academic Publishers; 1993.

[13] RIPPER User Commands Manual.

[14] Cohen, W.; "Fast Effective Rule Induction"; *Machine Learning: Proceedings of the Twelfth International Conference*; 1995.

[15] *Chater N., Vitányi P.;* "Simplicity: A unifying principle in cognitive science?" http://homepages.cwi.nl/~paulv/papers/tcs02.pdf.

[16] Mach, E.; *The analysis of sensations and the relation of the physical to the psychical*. NewYork: Dover Publications. (Original work published 1914).

Appendix

Preprocessor User's Guide

The Preprocessor tool was developed specifically for preparing the data sets for the RIPPER program. A single input file is provided and the tool prepares the training and testing data files as well as reports to standard output the statistics of the processing.

1) Required input to the Preprocessor tool is a file named ***CSVCDCdatafile.csv*** which contains a comma separated list of data elements

2) Data attributes chould appear in the following order:

    - Response
    - Score
    - Strike 3
    - Strike 2
    - Treat

3) Run the ***Preprocessor*** tool which will prompt the user for a cut-off value as inpout for separation of the data into test and training sets. A value between 1-1,000 is expected. A higher value will result in a larger testing data set.

4) Output Files produced are:

    - ***CDCTrainData.data***
    - ***CDCTestData.data***

5) The Preprocessor also reports, to standard output, the total number of data elements processed, the number processed to the test data set, and the number processed to the training data set.

```
//Preprocessor program for transforming
//CDC data input file.
//Removes '.' from input file.
//The program will also provide a count
//of input and output elements processed.
//The program uses a psuedo-random number
//generator to create a training set for the
//RIPPER system.


#include <algorithm>
#include <cstdlib>
#include <iostream>
#include <fstream>
#include <ctime>

using namespace std;

int main()
{
  ifstream input("CSVCDCdatafile.csv");
  ofstream output("CDCTrainData.data");
  ofstream testout("CDCTestData.test");


  //get input from user to adjust test/train data dispersion
  cout << "Enter an integer value between 0 and 1000 to be used" << endl;
  cout << "to distribute data over training and testing sets.  The" << endl;
  cout << "lower the value entered then more data will be directed to" << endl;
  cout << "the testing data set" << endl;
  int disperse(0);
  cin >> disperse;

  //initialize the random number generatot
  srand(static_cast<unsigned>(time(0)));
  string transform;
  string test;
  int inputCount(0);
  int dumpCount(0);
  int testCount(0);
  int trainCount(0);
  int testOrNot(0);

  while (input) {
    //increment the counter
    inputCount++;
```

```
    input >> transform;
    testOrNot =  rand()%1000;
    //dump the input lines with no data for a field
    // missing data is marked with a '.' character
    if (transform.find_first_of('.',0) < 9) {
      cout << "Dumping the input line: " << transform << endl;
      dumpCount++;
      continue;
    }
    //transform the last character from a 0 or 1 to a class value
    //if 0 then NOT Level IV and replace '0' with 'N'
    //if 1 then Level IV is true replace '1' with 'Y'
    if (transform.at(transform.size()-1) == '0') {
      transform.at(transform.size()-1) = 'N';
      // take some of the data for a test file
    }
    else {
          transform.at(transform.size()-1) = 'Y';
    }
    if (testOrNot > disperse) {
      //terminate the line with a '.' and output to testing file
      testout << transform << '.' << endl;
      testCount++;
    }
    else {
      //terminate the line with a '.' and output to training file
      output << transform << '.' << endl;
      trainCount++;
    }
  }
  cout << "This is the total number of lines read: " << inputCount << endl;
  cout << "This is the total number of lines dumped: " << dumpCount << endl;
  cout << "This is the total number of test data: " << testCount << endl;
  cout << "This is the total number of training data: " << trainCount << endl;
}
```

STATE OF CALIFORNIA
CDC 839 (Rev 07/02)

CDC CLASSIFICATION SCORE SHEET

DEPARTMENT OF CORRECTIONS
Original - Central File
Canary - OIS
Green - Inmate

4. COUNTY OF LAST LEGAL RESIDENCE [ ] 18

5. FORM IDENTIFICATION (ENTER X in a, b or c)
a) NEW [ ] 21 b) CORRECTION [ ] 22 DATE CORRECTED [ ][ ] – [ ][ ] – [ ][ ] 23 (MO – DAY – YR) c) DELETE [ ] 29

B. BACKGROUND FACTORS

AGE AT FIRST ARREST MATRIX

POINTS		POINTS	
0 TO 17 =	12	30 TO 35 =	4
18 TO 21 =	10	36 + =	0
22 TO 29 =	8		

FIRST ARREST DATE OR AGE _____
DATE OF BIRTH _____

1. AGE AT FIRST ARREST _____ POINTS (See Matrix) = [ ][ ] 30

AGE AT RECEPTION MATRIX

POINTS		POINTS	
16 TO 20 =	8	27 TO 35 =	4
21 TO 26 =	6	36 + =	0

2. AGE AT RECEPTION _____ POINTS (See Matrix) = [ ] 32

3. TERM IN YEARS ( ) x 2 (MAX 50) = [ ][ ] 33

4. STREET GANG/DISRUPTIVE GROUP + 6 = [ ] 35
a) Type Code [ ][ ] 36 b) Verification Code [ ] 38

5. MENTAL ILLNESS + 4 = [ ] 43

CCCMS	EOP	MHCB	DMH
[ ] 39	[ ] 40	[ ] 41	[ ] 42

Level of Care Code (Enter X)

6. PRIOR JAIL OR COUNTY JUVENILE SENTENCE OF 31+ DAYS (MAX 1) + 1 = [ ] 44

7. PRIOR INCARCERATION(S)
a) CYA, Juvenile State/Federal Level (MAX 1) + 1 = [ ] 45
b) CDC, CRC, Adult State/Federal Level (MAX 1) + 1 = [ ] 46

8. TOTAL BACKGROUND FACTORS SCORE = + _____

C. SCORE ADJUSTMENT

1. Total Score Adjustment to CDC 839 for correction(s) subsequent to endorsement (+ or -) [ ][ ] 47

D. PRIOR INCARCERATION BEHAVIOR

1. LAST 12 MONTHS OF INCARCERATION
a) FAVORABLE - (No Serious Disciplinary) (MINUS) – 4 = [ ] 50
b) UNFAVORABLE (Serious Disciplinaries) Dates: NUMBER OF _____ x 4 = [ ][ ] 51

2. SERIOUS DISCIPLINARY HISTORY
a) Battery or Attempted Battery on a Non-Prisoner Dates: _____ x 8 = [ ][ ] 53
b) Battery or Attempted Battery on an Inmate Dates: _____ x 4 = [ ][ ] 55
c) Distribution of Drugs Dates: _____ x 4 = [ ][ ] 57
d) Possession of a Deadly Weapon (Double-weight if within last 5 years) Dates: _____ x 4 = [ ][ ] 59
e) Inciting a Disturbance Dates: _____ x 4 = [ ][ ] 61
f) Battery Causing Serious Injury Dates: _____ x 16= [ ][ ] 63

3. TOTAL PRIOR INCARCERATION BEHAVIOR SCORE (Combine D.1. and sum of D.2.) = + / - _____

E. PLACEMENT

MANDATORY MINIMUM SCORE FACTOR CODES AND SCORES

CODE		SCORE
[A]	Condemned	52
[B]	Life Without Possibility of Parole	52
[C]	CCR 3375.2(a)(7) Life Inmate	28
[D]	History of Escape	19
[E]	Warrants "R" Suffix	19
[F]	Violence Exclusion	19
[G]	Public Interest Case	19
[H]	Other Life Sentence	19

1. **PRELIMINARY SCORE (Not less than 0)** (Combine Item B.8., Total Backgound Factors Score, and Item D.3., Total Prior Incarceration Behavior Score) [ ][ ][ ] 65
2. **MANDATORY MINIMUM SCORE FACTOR CODE** (Assess Highest Factor) [ ] 68
3. **MANDATORY MINIMUM SCORE** [ ][ ] 69

ENTER PRELIMINARY SCORE OR MANDATORY MINIMUM SCORE WHICHEVER IS GREATER

4. PLACEMENT SCORE [ ][ ][ ] 71

F. SPECIAL CASE FACTORS

1. **HOLDS, WANTS and DETAINERS** (Enter A or P)
Felony [ ] 74 USINS [ ] 75
2. **RESTRICTED CUSTODY SUFFIX** (Enter R) [ ] 76
3. **CURRENT INSTITUTION AND FACILITY** [ ][ ][ ][ ] – [ ][ ][ ] 77
4. **ELIGIBLE FOR RESTITUTION CENTER** (Enter Y or N) [ ] 84
5. **LEVEL IV DESIGN**
a) 180 Status (Y/N) _____
b) Reason Code _____
6. **US ARMED FORCES** (Enter Y or N) [ ] 85
7. **CASEWORKER'S NAME** [ ][ ][ ][ ][ ][ ][ ][ ] FI [ ] 86

G. CLASSIFICATION STAFF REPRESENTATIVE

1. **LAST NAME** [ ][ ][ ][ ][ ][ ][ ][ ] 95
2. **DATE OF ACTION** [ ][ ] – [ ][ ] – [ ][ ] 103 (MO – DAY – YR)
3. **LEVEL IV DESIGN**
a) 180 Status (Enter Y to Apply) [ ] 109 b) Reason Code [ ][ ] 110
4. **MINIMUM CUSTODY**
a) Eligibility (Enter E, L or P) [ ] 112 b) Reason Code [ ][ ][ ] 113
5. **CCRC ELIGIBILITY** (ENTER REN or REX) [ ][ ][ ] 116
6. **DEVELOPMENTAL DISABILITY PROGRAM (DDP) CODE** [ ][ ][ ] 119
7. **DISABILITY PLACEMENT PROGRAM (DPP) CODE(S)**
a) Primary (affects placement) [ ][ ][ ] 122 b) [ ][ ][ ] 125 c) [ ][ ][ ] 128 d) [ ][ ][ ] 131
8. **ADMINISTRATIVE DETERMINANT CODE(S)**
a) [ ][ ][ ] b) [ ][ ][ ] c) [ ][ ][ ] d) [ ][ ][ ] e) [ ][ ][ ]

STATE OF CALIFORNIA
CDC 840 (Rev 07/02)

DEPARTMENT OF CORRECTIONS
Original - Central File
Canary - OIS
Green - Inmate

CDC RECLASSIFICATION SCORE SHEET

4. DATE OF LAST SCORE SHEET MO / DAY / YR [ ][ ] [ ][ ] [ ][ ] 18

5. FORM IDENTIFICATION (ENTER X in a, b or c) a) NEW [ ] 24 b) CORRECTION [ ] 25 DATE CORRECTED MO [ ][ ] DAY [ ][ ] YR [ ][ ] 26 c) DELETE [ ] 32

B. ANNUAL/ 6 MONTH REVIEW PERIOD DATES

1. REVIEW PERIOD BEGINNING DATE MO [ ][ ] DAY [ ][ ] YR [ ][ ] 33
3. (Enter X) Annual [ ] 39
2. REVIEW PERIOD ENDING DATE MO [ ][ ] DAY [ ][ ] YR [ ][ ] 40
4. Number of Full Review Periods _____

C. FAVORABLE BEHAVIOR SINCE LAST REVIEW

1. Continuous Minimum Custody _____ x 4 = [ ][ ] 46
2. No Serious Disciplinary _____ x 2 = [ ][ ] 48
3. Average or Above Performance in Work, School or Vocational Program _____ x 2 = [ ][ ] 50
4. **TOTAL FAVORABLE POINTS** = - _____

D. UNFAVORABLE BEHAVIOR SINCE LAST REVIEW

SERIOUS DISCIPLINARIES Number of

1. Div. A-1/A-2 Dates: _____ x 8 = [ ][ ] 52
   Div. B, C & D Dates: _____ x 6 = [ ][ ] 54
   Div. E & F Dates: _____ x 4 = [ ][ ] 56
2. Battery or Attempted Battery on a Non-Prisoner Dates: _____ x 8 = [ ][ ] 58
3. Battery or Attempted Battery on an Inmate Dates: _____ x 4 = [ ][ ] 60
4. Distribution of Drugs Dates: _____ x 4 = [ ][ ] 62
5. Possession of a Deadly Weapon Dates: _____ x 16 = [ ][ ] 64
6. Inciting a Disturbance Dates: _____ x 4 = [ ][ ] 66
7. Battery Causing Serious Injury Dates: _____ x 16 = [ ][ ] 68
8. **TOTAL UNFAVORABLE POINTS** = + _____

E. SCORE ADJUSTMENT

1. **TOTAL CORRECTION/ADJUSTMENT (+ OR -)** [ ] [ ][ ] 70

F. COMPUTATION OF SCORE

1. **PRIOR PRELIMINARY SCORE** (Preliminary Score from 839/New Preliminary Score from 840 or 841) = [ ][ ][ ] 73
2. Net Change in Score (D. 8 minus C. 4) = (+ or -) [ ] [ ][ ] 76
3. **PRELIMINARY SCORE SUBTOTAL (Not less than 0)** = _____
4. Change in Term Points (T/P) (x 2) - Old T/P _____ + New T/P _____ = (+ or -) [ ] [ ][ ] 79
5. **NEW PRELIMINARY SCORE (Not less than 0)** = [ ][ ][ ] 82

G. PLACEMENT

MANDATORY MINIMUM SCORE FACTOR CODES AND SCORES

CODE	SCORE	CODE	SCORE
[A] Condemned	52	[E] Warrants "R" Suffix	19
[B] Life Without Possibility of Parole	52	[F] Violence Exclusion	19
[C] CCR 3375.2(a)(7) Life Inmate	28	[G] Public Interest Case	19
[D] History of Escape	19	[H] Other Life Sentence	19

1. **SCORE FACTOR CODE (Assess Only Highest Factor)** [ ] 85
2. **MANDATORY MINIMUM SCORE** [ ][ ] 86
3. **PLACEMENT SCORE** ENTER NEW PRELIMINARY SCORE OR MANDATORY MINIMUM SCORE WHICHEVER IS GREATER [ ][ ][ ] 88

H. SPECIAL CASE FACTORS

1. **HOLDS, WANTS and DETAINERS (Enter A, P or *)** Felony [ ] 91 USINS [ ] 92
2. **RESTRICTED CUSTODY SUFFIX** (Enter R or *) [ ] 93
3. **ELIGIBLE FOR RESTITUTION CENTER** (Enter Y or N) [ ] 94
4. **LEVEL IV DESIGN** a) 180 Status (Y/N) _____ b) Reason Code _____
5. **US ARMED FORCES** (Enter Y or N) [ ] 95
6. **CURRENT INSTITUTION AND FACILITY** [ ][ ][ ][ ] [ ][ ][ ] 96
7. **COUNTY OF LAST LEGAL RESIDENCE** [ ][ ][ ] 103
8. **CASEWORKER'S NAME** [ ][ ][ ][ ][ ][ ][ ][ ] FI [ ] 106

I. CLASSIFICATION STAFF REPRESENTATIVE

1. **LAST NAME** [ ][ ][ ][ ][ ][ ][ ][ ] 115
2. **DATE OF ACTION** MO [ ][ ] DAY [ ][ ] YR [ ][ ] 123
3. **LEVEL IV DESIGN** a) 180 Status (Enter Y or *) [ ] 129 b) Reason Code [ ][ ] 130
4. **MINIMUM CUSTODY** a) Eligibility (Enter E, L or P) [ ] 132 b) Reason Code [ ][ ][ ] 133
5. **CCRC ELIGIBILITY** (Enter REN, REX or *) [ ][ ][ ] 136
6. **DEVELOPMENTAL DISABILITY PROGRAM (DDP) CODE** [ ][ ][ ] 139
7. **DISABILITY PLACEMENT PROGRAM (DPP) CODE(S)** a) (*) **Primary** (affects placement) [ ][ ][ ][ ] 142 b) (*) [ ][ ][ ][ ] 146 c) (*) [ ][ ][ ][ ] 150 d) (*) [ ][ ][ ][ ] 154
8. **ADMINISTRATIVE DETERMINANT CODE(S)** a) (*) [ ][ ][ ][ ] 158 b) (*) [ ][ ][ ][ ] 162 c) (*) [ ][ ][ ][ ] 166 d) (*) [ ][ ][ ][ ] 170 e) (*) [ ][ ][ ][ ] 174

STATE OF CALIFORNIA
CDC 841 (07/02)

DEPARTMENT OF CORRECTIONS

CDC READMISSION SCORE SHEET

4. DATE OF LAST SCORE SHEET — MO / DAY / YR — 18
5. COUNTY OF LAST LEGAL RESIDENCE — 24
6. FORM IDENTIFICATION (ENTER X in a, b or c) — a) NEW 27 — b) CORRECTION 28 — DATE CORRECTED MO / DAY / YR 29 — c) DELETE 35

B. READMISSION REVIEW PERIOD CALCULATION

1. DATE PAROLED — MO / DAY / YR — 36
2. REVIEW PERIOD BEGINNING DATE — MO / DAY / YR — 42
3. Total Review Periods Not Previously Addressed — MONTHS / DAYS / YEARS
4. Number of 6 Month Review Periods
5. Partial Review Period (Less Than 6 Months) — MONTHS / DAYS

C. FAVORABLE BEHAVIOR SINCE LAST REVIEW

Number Of Review Periods
Apply points for each Review Period of 6 Months at full value
Apply points for Review Period less than 6 Months at half value

1. Continuous Minimum Custody — x 4 = — 48
2. No Serious Disciplinary — x 2 = — 50
3. Average or Above Performance in Work, School or Vocational Program — x 2 = — 52
4. TOTAL FAVORABLE POINTS = -

D. UNFAVORABLE BEHAVIOR SINCE LAST REVIEW

SERIOUS DISCIPLINARIES — Number of

1. Div. A-1/A-2 — Dates: — x 8 = — 54
   Div. B, C & D — Dates: — x 6 = — 56
   Div. E & F — Dates: — x 4 = — 58
2. Battery or Attempted Battery on a Non-Prisoner — Dates: — x 8 = — 60
3. Battery or Attempted Battery on an Inmate — Dates: — x 4 = — 62
4. Distribution of Drugs — Dates: — x 4 = — 64
5. Possession of a Deadly Weapon — Dates: — x 16 = — 66
6. Inciting a Disturbance — Dates: — x 4 = — 68
7. Battery Causing Serious Injury — Dates: — x 16 = — 70
8. TOTAL UNFAVORABLE POINTS = +

E. SCORE ADJUSTMENT

1. TOTAL CORRECTION/ADJUSTMENT (+ OR -) — 72

F. COMPUTATION OF SCORE

1. PRIOR PRELIMINARY SCORE (Preliminary Score from 839/New Preliminary Score from 840 or 841) = — 75
2. Net Change in Score (D. 8 minus C. 4) = (+ or -) — 78
3. PRELIMINARY SCORE SUBTOTAL (Not less than 0) =
4. Change in Term Points (T/P) (x 2) - Old T/P _____ + New T/P _____ = (+ or -) — 81
5. NEW PRELIMINARY SCORE (Not less than 0) = — 84

G. PLACEMENT

MANDATORY MINIMUM SCORE FACTOR CODES AND SCORES

CODE	SCORE	CODE	SCORE
[A] Condemned	52	[E] Warrants "R" Suffix	19
[B] Life Without Possibility of Parole	52	[F] Violence Exclusion	19
[C] CCR 3375.2(a)(7) Life Inmate	28	[G] Public Interest Case	19
[D] History of Escape	19	[H] Other Life Sentence	19

1. SCORE FACTOR CODE (Assess Only Highest Factor) — 87
2. MANDATORY MINIMUM SCORE — 88
3. PLACEMENT SCORE — ENTER NEW PRELIMINARY SCORE OR MANDATORY MINIMUM SCORE WHICHEVER IS GREATER — 90

H. SPECIAL CASE FACTORS

1. HOLDS, WANTS and DETAINERS (Enter A or P) — Felony 93 — USINS 94
2. RESTRICTED CUSTODY SUFFIX (Enter R) — 95
3. ELIGIBLE FOR RESTITUTION CENTER (Enter Y or N) — 96
4. LEVEL IV DESIGN — a) 180 Status (Y/N) _____ — b) Reason Code _____
5. US ARMED FORCES (Enter Y or N) — 97
6. CURRENT INSTITUTION AND FACILITY — 98
7. RETURN STATUS (Enter RTC or WNT) — 105
8. CASEWORKER'S NAME — FI — 108

I. CLASSIFICATION STAFF REPRESENTATIVE

1. LAST NAME — 117
2. DATE OF ACTION — MO / DAY / YR — 125
3. LEVEL IV DESIGN — a) 180 Status (Enter Y to apply) 131 — b) Reason Code 132
4. MINIMUM CUSTODY — a) Eligibility (Enter E, L or P) 134 — b) Reason Code 135
5. CCRC ELIGIBILITY (Enter REN or REX) — 138
6. DEVELOPMENTAL DISABILITY PROGRAM (DDP) CODE — 141
7. DISABILITY PLACEMENT PROGRAM (DPP) CODE(S) — a) Primary (affects placement) 144 — b) 147 — c) 150 — d) 153
8. ADMINISTRATIVE DETERMINANT CODE(S) — a) 156 — b) 159 — c) 162 — d) 165 — e) 168