Vivek Reddy Devi (vdevi2@uic.edu)

Prathyusha Dasari (pdasar2@uic.edu)

Avisha Mishra (amishr25@uic.edu)

Santosh Kumar Ponnada (sponna6@uic.edu)

**IDS 476**

**Business forecasting using Time Series Methods**

**Earning Statement Fraudsters prediction using**

**Logistic Regression**

1. **Introduction**

In this study, the methods of machine learning are used to analyze a dataset and develop a model that would predict earnings statement fraudsters in organizations.

* 1. **Dataset Description**

This datasetwascollectedfrom Harvard Business Publishing website.

Please refer the below table for data dictionary:

|  |  |
| --- | --- |
| **Column Name** | **Description** |
| DSRI | Days Sales to Receivables Index |
| GMI | Gross Margin Index |
| AQI | Asset Quality Index |
| SGI | Sales Growth Index |
| DEPI | Depreciation Index |
| SGAI | Sales General and Administrative Index |
| ACCR | Accruals to Total Assets |
| LEVI | Leverage Index |

* 1. **Purpose of analysis**

Earning manipulations by corporates has been a major concern for banks. There have been numerous occasions in which some established firms have confessed defaulting of their account by billions of dollars. Hence, our aim is to develop a model to correctly categorize someone as a fraudster with relatively less error.

1. **Statistical Analysis**
   1. **Pre-processing**

The given dataset is unbalanced i.e. between the two classes of DEFAULTERS and NON- DEFAULTERS, the data related to DEFAULTERS is very less. There are 39 DEFAULTERS and 1200 NON-DEFAULTERS. We have large data related to “NON-DEFAULTERS” compared to “DEFAULTERS”. If the model is built on the same dataset then our model will predict the not required class with high accuracy neglecting the required class. Hence, we need to balance the given dataset.

Imbalanced data leads to modeling problem in binary classification. There are some classifier algorithms like Decision trees and Logistic Regression which have a bias towards classes with more number of instances. So, when such models are constructed on unbalanced data sets accuracy or any other conventional evaluation measures cannot determine the model performance. If we get an accuracy of 98% in such cases, it is not correct as one of the classes (one with less instances) has been eliminated or ignored. We can handle these problems by either balancing the data or by using any other evaluation measure for the performance of the model.

**Random over sampling method:** Over-Sampling increases the number of instances in the minority class by randomly replicating them to present a higher representation of the minority class in the sample.

* 1. **Software used**

We have used R to predict Fraudsters using Logistic Regression

* 1. **Procedures used**

Step 1: Balancing the given dataset

Step2: Dividing into Training and Testing data (70%- Training, 30%- Testing)

Step3: Constructing a Logistic Regression Model on Training Data

Step 4: Testing our logistic model on Test data and finding the Accuracy

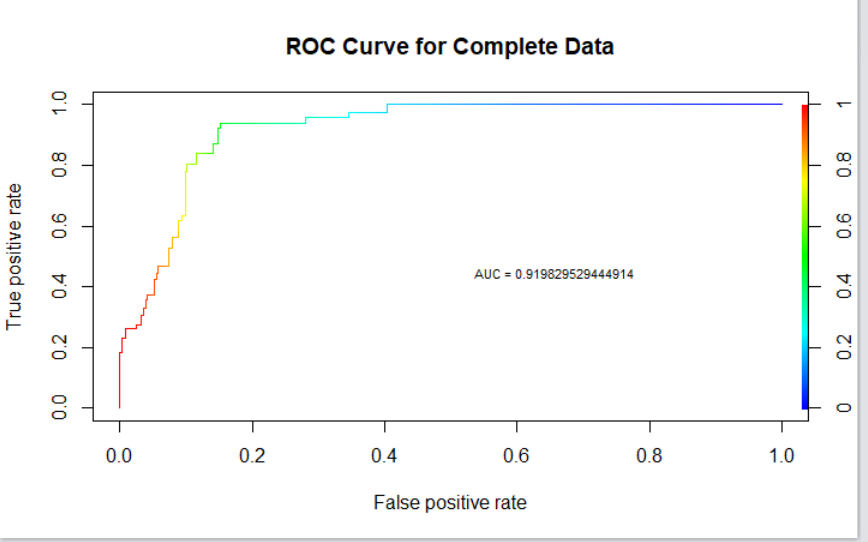
1. **Conclusion**

The model developed makes prediction about the probability of a company being DEFAULTER. This has been developed using 7 significant variables. The target variable has a positive relationship with all the variables used in the model except “LEVI” with which model has a negative relationship evident from the coefficients.

As we have a balanced dataset, we can use accuracy as the measure for model performance with confusion matrix.

We have achieved an accuracy of 83.13% from the model.

Below is the ROCR Curve:



Dataset – Accuracy – 83.13%, AUC – 91.9

* We have also found out the DEFAULTER Score as below:

. D-score can be calculated as:

D = -13.6620 + 2.1532\*DSRI + 1.6164\*GMI +0.7272\*AQI + 4.1932\*SGI + 3.4004\*DEPI +9.5747\* ACCR – 0.8664\*LEVI

Applying the above model to some sample values, we get D-score = 0. If D-score is 0, the company is a potential defaulter.

**Technical Appendix (Results and Output)**

* **Initial Model Building**

Call:

glm(formula = MANIPULATOR ~ ., family = "binomial", data = sample\_train)

Deviance Residuals:

Min 1Q Median 3Q Max

-3.2002 -0.5125 -0.0399 0.5801 1.9190

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -14.0390 2.2125 -6.345 2.22e-10 \*\*\*

DSRI 2.1128 0.4261 4.958 7.11e-07 \*\*\*

GMI 1.6843 0.4363 3.860 0.000113 \*\*\*

AQI 0.7046 0.1393 5.060 4.20e-07 \*\*\*

SGI 4.2290 0.7511 5.631 1.79e-08 \*\*\*

DEPI 3.3472 0.9179 3.647 0.000266 \*\*\*

SGAI 0.3340 0.4014 0.832 0.405412

ACCR 9.3367 1.4808 6.305 2

.88e-10 \*\*\*

LEVI -0.8372 0.2806 -2.984 0.002845 \*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 321.55 on 231 degrees of freedom

Residual deviance: 171.78 on 223 degrees of freedom

AIC: 189.78

Number of Fisher Scoring iterations: 8

Variable “SGAI” is not significant

* **Logistic Model**

**sample\_logit <- glm(MANIPULATOR~.-SGAI,data=sample\_train,family = "binomial")**

**summary(sample\_logit)**

Call:

glm(formula = MANIPULATOR ~ . - SGAI, family = "binomial", data = sample\_train)

Deviance Residuals:

Min 1Q Median 3Q Max

-3.2341 -0.5161 -0.0397 0.5623 1.8789

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -13.6620 2.1616 -6.320 2.61e-10 \*\*\*

DSRI 2.1532 0.4187 5.143 2.70e-07 \*\*\*

GMI 1.6164 0.4071 3.970 7.18e-05 \*\*\*

AQI 0.7272 0.1380 5.268 1.38e-07 \*\*\*

SGI 4.1932 0.7548 5.555 2.77e-08 \*\*\*

DEPI 3.4004 0.9300 3.656 0.000256 \*\*\*

ACCR 9.5747 1.4895 6.428 1.29e-10 \*\*\*

LEVI -0.8644 0.2734 -3.162 0.001568 \*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 321.55 on 231 degrees of freedom

Residual deviance: 172.50 on 224 degrees of freedom

AIC: 188.5

Number of Fisher Scoring iterations: 8

* **Predictions on Test data using the model**

res1 <- predict(sample\_logit,sample\_test,type="response")

* **Probabilities of earnings manipulations**

7 8 11 16 18 20 22

0.0632988446 0.0003264352 0.1376538482 0.0602070225 0.1443225802 0.0473438841 0.1055562224

23 27 28 37 39 40 42

0.0466562568 0.0412966078 0.0967295681 0.9474940098 0.0511452088 0.1466450443 0.0013758509

43 49 53 57 59 67 68

0.1311820573 0.1549208311 0.4590093128 0.0330789541 0.0333950770 0.0879390722 0.1791287068

69 71 72 76 79 83 85

0.0612461774 0.9999873220 0.0071959674 0.2070522460 0.7536578228 0.9997490790 0.4555558691

94 95 96 99 100 102 105

0.0084488669 0.0306374695 0.0436107794 0.0008846658 0.0160746163 0.1072977004 0.0885127351

108 109 112 113 115 125 126

0.3123554825 0.4552148967 0.1888506127 0.0508368916 0.0020532464 0.1699516161 0.9966344825

127 130 138 141 142 147 148

0.0042895671 0.0040712640 0.0011693225 0.2166328335 0.3075451475 0.0642740689 0.0720366792

150 152 153 156 159 160 163

0.0222036750 0.0229616432 0.2537713008 0.0039948928 0.5823391987 0.0405359133 0.0198328373

168 170 171 173 174 176 180

0.3414415829 0.0339923751 0.0781054646 0.0029911973 0.1380576929 0.0288745999 0.1559648423

182 183 187 188 190 194 195

0.9315034027 0.8593850572 1.0000000000 1.0000000000 0.5723582153 0.9472322949 0.5351289345

198 200 203 205 207 208 210

0.9006073365 0.6267673422 0.9999999996 0.9999946725 0.9999946725 0.8014187935 0.9899260588

212 217 218 220 221 223 224

0.8593850572 0.1711719261 0.3151430150 0.9899260588 0.9999999996 0.6267673422 0.4529846655

226 230 232 235 237 243 246

1.0000000000 0.5351289345 0.4529846655 0.2506878236 0.1711719261 0.9006073365 0.2506878236

249 254 256 258 260 266 281

0.8537917704 0.2617881834 0.2506878236 0.9237088159 1.0000000000 0.5230090488 0.1711719261

284 286 288 293 295 299 300

0.9999999996 0.5230090488 0.6119841642 0.6119841642 0.9472322949 0.7788596531 0.8593850572

304 305 306 307 316 323 324

1.0000000000 1.0000000000 1.0000000000 0.5317617048 0.7429419083 1.0000000000 0.2617881834

327 328 339 341 347 348

1.0000000000 0.9586978166 1.0000000000 0.9940945581 0.9315034027 0.8921853631

**References and Sources**

* An ***Introduction to Statistical Learning with Applications in R*** for implementing logistic regression in R
* For developing logistic model: <https://www.r-bloggers.com/how-to-perform-a-logistic-regression-in-r/>
* For balancing the data: <https://dzone.com/articles/handle-class-imbalance-data-with-r>