## Introduction

The Agency is charged with protecting the most vulnerable among us. Allegations are received, and investigations are conducted to validate the allegation, ensure alleged victim safety, and put processes in place to prevent reoccurrence. Post case closure, a sample of cases is reviewed to ensure the investigation was conducted correctly, ensuring the alleged victim is safe and receiving any needed services.

The Agency is concerned that there are connections between errors, that certain errors produce other related errors, and challenge alleged victim safety and wellbeing. Additionally, these connections may be causing cases to be scored lower than they should and provide false indications as to the quality of care being provided. This affects metrics on the strategic plan that needs to be submitted to the state legislature. It also affects staff morale, some of whom do not see any way to improve scoring.

Using this information, the Agency would like to determine if there are relationships between categories and if there are gaps in services provided and/or redundancies in scoring that if eliminated will improve staff morale and provide for a more accurate “picture” to the state legislature.

Using this information, the agency wants to ensure a high standard of care and if indicated, target certain categories for quality-of-care improvements. An additional benefit may be an indication as to whether the current pandemic affects the scoring.

## Analysis and Models

### About the Data

Unfortunately, the data cannot be extracted in one file. Two files were needed, one containing individual errors associated with cases and one with comments associated with those cases.

The datasets being used contain data provided by the agency over six months (June – Nov 2020). The dataset contained almost 24,000 lines of data covering 3.200+ individuals and encompasses state-wide cases. A six-month period was used as the Agency has relayed that case intakes are seasonal. It contains data for all case scorings, over errors given to each case.

There were over 60,000 cases from June – November 2021. The case readings sampled 4.8% of the cases. Very few cases did not receive at least one error.

Cases being reviewed are scored on 11 categories for alleged victim safety and quality containing subcategories which are scored. The scoring uses a Met; Did Not Meet; Exceeded; Unknown/NA ratings. Comments are required for any category that receives a Did Not Meet, Exceeds and Unknown. After discussing the dataset with the data owners, it was determined that categories 6-11 would not be part of this analysis.

The **categories** and subcategories

**1.0 Standard 1: Alleged Victim/Client Safety**

1.1 Adequate Initiation contact completed.

1.2 Safety Assessment from SHIELD

1.3 Locating alleged victim.

1.4 Required contacts for alleged victims

1.5 Refusal or Withdrawal of the Alleged Victim

1.6 Legal Actions - As needed

1.7 Notifications, Written Materials and Investigations Involving other Entities

**2.0 Standard 2: Investigation**

2.1 Interviews: AV, AP Medical Professionals, Collaterals and Reporter

2.2 Contacts for Interviews

2.3 Evidence Collection: Photographic, Demonstrative and Documentary

2.4 Recognition and Investigation of Allegations

2.5 Allegation Conclusion

2.6 Risk of Recidivism Assessment from SHIELD

**3.0 Standard 3: Case Documentation**

3.1 Narrative Documentation

**4.0 Standard 4: Service Provision and Outcomes**

4.1 Strength and Needs Assessment from SHIELD

4.2 Appropriate Action in the Case

4.3 Outcomes for the Case

**5.0 Standard 5: APS Specialist Productivity**

5.1 Appropriate Timeliness of Investigation Elements

Note that 1.0, 2.0, etc. are headings and not used in the clustering or other analyses.

### Data Structure

The data containing comments is text-based with each case reading on its own line and contained multiple errors documented in unstructured text with the appropriate comments. The file is downloaded as a .csv. The initial dataset contained the following categories

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Case ID | Standard | Question | Reading ID | Caseworker | Region | Unit | Comment |
| 12345 | Standard 1: Alleged Victim and Client Safety | 1.0.2 Opportunities for Improvement: | 98762 | Ane Doe | 011 | 86 | 1.4 No - The case narrative does not indicate the contact … |

In the error dataset file provided, each error is on its own line but without comments. This dataset contained over 30,000 lines. The dataset snippet follows. The comments contained are generic and inadequate to make an evaluation.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Date | Reading ID | Region | Unit | Standard | Question | Answer | Comments |
| 6/1/2020 | 137862 | 1 | 31 | Standard 1: Alleged Victim and Client Safety | 1.1 | No | Adequate Initiation contact completed. |
| 6/1/2020 | 137862 | 1 | 31 | Standard 1: Alleged Victim and Client Safety | 1.2 | No | Safety Assessment from SHIELD |
| 6/1/2020 | 137862 | 1 | 31 | Standard 11: Substantial Impairment (For State Office Research Purposes Only) | 11.1 | No | If this AV/CL is 65+, does it appear they would meet the definition of substantially impaired if age were not considered? |
| 6/1/2020 | 137863 | 1 | 31 | Standard 1: Alleged Victim and Client Safety | 1.1 | No | Adequate Initiation contact completed. |
| 6/1/2020 | 137863 | 1 | 31 | Standard 1: Alleged Victim and Client Safety | 1.2 | No | Safety Assessment from SHIELD |
| 6/1/2020 | 137863 | 1 | 31 | Standard 1: Alleged Victim and Client Safety | 1.4 | No | Required contacts for alleged victims |

The dataset required extensive cleaning, removal of case identification information, and other personally identifying information. During the cleaning process, it was discovered that there were a significant number of non-standard responses - other possible words being used, such including Unknown and other caseworker-provided text. These needed to be standardized. Additional categories were added for Month\_Yr, Standard (1.1, 1.2, 2.1, etc.) and as to whether the case met standards or not (Yes\_No). Excel was used for most of the cleaning.

A version of the error dataset was wrangled to a transaction list using Power Pivot which allowed for the association of all errors per case reading ID. Categories and column names were removed creating a dataset similar to the example below, with each row being a single transaction. This allowed for the creation of a text file with the associated error to perform clustering analyses.

1.4,2.2,5.1,6.1

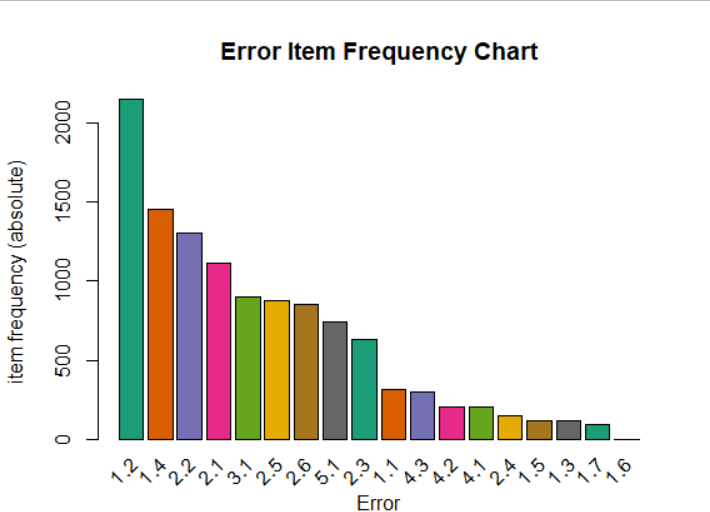
1.2,2.1,2.3

1.2,1.4,2.2,2.5

After the initial clustering and consultations with the data owners, Standards 6 through 11.1 were removed from the analysis as they were determined to be derivative of Standards 1.1 through 5.1. and did not add any value to the analysis. Most were only collecting specific data relating to COVID. The removed standards may be included at a later date.

## Exploratory Data Analysis and Data Visualization

Initial examination of the data produced the following error frequency chart. After showing the frequency chart to the data owners, it was determined that 1.2 errors were being inflated by the current pandemic and a policy change that affected workers but not the way cases were scored, thereby inflating the number of errors. (66.8% of the cases read had 1.2 errors.) We all agreed that using another subcategory, such as 3.1 (28.1% of cases read), could provide more insight.



## Descriptive Statistics

### Brief Summary

### Models

#### Association Rule Mining

In this case, errors were treated as items in a transaction to perform associations. *Caveat emptor*.

Looking at the relationships between errors. If on the left-hand side is if there is this error, then one would find the accompanying right-hand side error. Only confidence levels of .5 or greater were used. For this analysis, only the top 14 for each error was used. Then sorted according to frequency and confidence.

For example, if there is a 2.1 error then there is a high likelihood that there will be a 1.2 error also. This occurred 335 times in the dataset used or 33% of the time.

#### Rules

Multiple combinations of the categories were explored and produced using code similar to below.

## Mulriple rule gereration using a loop

# Values to be used

x <- c("1.1","1.2", "1.3", "1.4", "1.5", "1.6", "1.7", "2.1","2.2", "2.3", "2.4", "2.5", "2.6","3.1", "4.1", "4.2", "4.3", "5.1", "6.1")

#Writing to a file for use later

sink('score\_2\_lhsRev.txt')

for (val in x) {

print(val)

rulesVallhs <- apriori( Book1, parameter = list(supp = .001, conf = .08, minlen = 2), appearance = list(default = "rhs", lhs = val), control = list(verbose = F))

rulesVallhs <- sort(rulesVallhs, decreasing = TRUE, by = 'confidence')

inspect(rulesVallhs)

}

sink()

#append

#Writing to a file for use later

sink('score\_data\_rhsRev.txt')

for (val in x) {

print(val)

rulesValrhs <- apriori( Book1, parameter = list(supp = .001, conf = .08, minlen = 2), appearance = list(default = "lhs", rhs = val), control = list(verbose = F))

rulesValrhs <- sort(rulesValrhs, decreasing = TRUE, by = 'confidence')

inspect(rulesValrhs)

}

sink()

The initial list of rules generated was over 2700 rules long. It was decided that the results would be sorted by frequency taking the rules with 100+ occurrences. Taking the top 12 rules in this list produced the following list of rules

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **lhs** |  | **rhs** | **confidence** | **count** | **% of cases** |
| {2.1} | => | {1.2} | 0.779 | 335 | 33% |
| {5.1} | => | {1.2} | 0.726 | 312 | 31% |
| {2.2} | => | {1.2} | 0.810 | 310 | 31% |
| {2.1} | => | {3.1} | 0.600 | 258 | 26% |
| {3.1} | => | {2.1} | 0.524 | 258 | 26% |
| {5.1} | => | {3.1} | 0.565 | 243 | 24% |
| {2.1} | => | {5.1} | 0.502 | 216 | 21% |
| {5.1} | => | {2.1} | 0.502 | 216 | 21% |
| {6.1} | => | {3.1} | 0.522 | 187 | 19% |
| {4.3} | => | {1.2} | 0.815 | 119 | 12% |
| {4.2} | => | {1.2} | 0.800 | 113 | 11% |
| {2.3} | => | {1.2} | 0.750 | 102 | 10% |

Analyzing the data using all the errors in a case and would it indicate another error. Taking the top rules with 20+ occurrences in this list produced the following list of rules

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **lhs** |  | **rhs** | **confidence** | **count** | **% of cases** |
| {1.2,3.1,5.1,6.1} | => | {2.1} | 0.890 | 47 | 4.7% |
| {4.3,5.1} | => | {2.2} | 0.540 | 43 | 4.3% |
| {1.2,4.1} | => | {2.2} | 0.510 | 41 | 4.1% |
| {1.2,4.3,5.1} | => | {2.2} | 0.550 | 36 | 3.6% |
| {4.1,5.1} | => | {2.2} | 0.510 | 34 | 3.4% |
| {3.1,4.3,5.1} | => | {2.2} | 0.550 | 26 | 2.6% |
| {1.2,3.1,4.1} | => | {2.2} | 0.510 | 23 | 2.3% |
| {1.2,3.1,4.2,6.1} | => | {2.1} | 0.850 | 22 | 2.2% |
| {1.2,3.1,4.3,5.1} | => | {2.2} | 0.580 | 21 | 2.1% |

There were some interesting combinations with high confidence but as the percentages indicate, did not happen very often. 1.2 errors were in 6 of 9 cases listed above. There was no easy way to pull out 1.2 errors in this analysis.

As stated above, the agency thought that errors on 3.1 would indicate errors in other areas, so the analysis concentrated on using 3.1 on both the left-hand and right-hand sides (individually)

An analysis was conducted using the Apriori algorithm with 3.1 on the LHS, then another run with 3.1 resulting on the RHS. Removing 1.2 errors from the analysis produces the following rules sorted by lift. (There were not 25 rules generated)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **lhs** |  | **rhs** | **support** | **confidence** | **coverage** | **lift** | **count** | **% of cases** |
| {2.1} | => | {3.1} | 0.264 | 0.600 | 0.44 | 1.19 | 258 | 8.9% |
| {2.4} | => | {3.1} | 0.042 | 0.590 | 0.07 | 1.16 | 41 | 1.4% |
| {4.1} | => | {3.1} | 0.066 | 0.580 | 0.11 | 1.15 | 64 | 2.2% |
| {4.2} | => | {3.1} | 0.083 | 0.570 | 0.15 | 1.13 | 81 | 2.8% |
| {5.1} | => | {3.1} | 0.249 | 0.565 | 0.44 | 1.10 | 243 | 8.4% |
| {4.3} | => | {3.1} | 0.081 | 0.541 | 0.15 | 1.07 | 79 | 2.7% |
| {2.5} | => | {3.1} | 0.046 | 0.530 | 0.09 | 1.05 | 45 | 1.6% |
| {1.1} | => | {3.1} | 0.089 | 0.521 | 0.17 | 1.03 | 87 | 3.0% |
| {2.6} | => | {3.1} | 0.055 | 0.480 | 0.12 | 0.95 | 54 | 1.9% |
| {2.3} | => | {3.1} | 0.066 | 0.470 | 0.14 | 0.93 | 64 | 2.2% |
| {1.4} | => | {3.1} | 0.059 | 0.468 | 0.13 | 0.93 | 58 | 2.0% |
| {2.2} | => | {3.1} | 0.172 | 0.440 | 0.39 | 0.87 | 168 | 5.8% |

Forcing 3.1 to be on the RHS

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **lhs** |  | **rhs** | **support** | **confidence** | **coverage** | **lift** | **count** | **% of cases** |
| {2.1,5.1} | => | {3.1} | 0.136 | 0.62 | 0.221 | 1.22 | 133 | 4.6% |
| {2.2,5.1} | => | {3.1} | 0.102 | 0.54 | 0.189 | 1.08 | 100 | 3.5% |
| {2.1,4.3} | => | {3.1} | 0.055 | 0.64 | 0.086 | 1.28 | 54 | 1.9% |
| {2.1,4.2} | => | {3.1} | 0.054 | 0.66 | 0.082 | 1.31 | 53 | 1.8% |
| {2.1,2.2} | => | {3.1} | 0.054 | 0.54 | 0.100 | 1.07 | 53 | 1.8% |
| {1.1,2.1} | => | {3.1} | 0.052 | 0.62 | 0.084 | 1.23 | 51 | 1.8% |
| {4.3,5.1} | => | {3.1} | 0.048 | 0.59 | 0.082 | 1.17 | 47 | 1.6% |
| {4.1,5.1} | => | {3.1} | 0.045 | 0.66 | 0.069 | 1.30 | 44 | 1.5% |
| {4.2,5.1} | => | {3.1} | 0.042 | 0.59 | 0.071 | 1.18 | 41 | 1.4% |
| {1.1,5.1} | => | {3.1} | 0.041 | 0.53 | 0.078 | 1.04 | 40 | 1.4% |
| {2.1,4.1} | => | {3.1} | 0.039 | 0.63 | 0.061 | 1.26 | 38 | 1.3% |
| {2.3,5.1} | => | {3.1} | 0.036 | 0.51 | 0.070 | 1.02 | 35 | 1.2% |
| {2.1,2.2,5.1} | => | {3.1} | 0.035 | 0.58 | 0.060 | 1.14 | 34 | 1.2% |
| {1.4,5.1} | => | {3.1} | 0.035 | 0.51 | 0.069 | 1.01 | 34 | 1.2% |
| {1.4,2.1} | => | {3.1} | 0.032 | 0.58 | 0.054 | 1.16 | 31 | 1.1% |
| {2.6,5.1} | => | {3.1} | 0.032 | 0.56 | 0.056 | 1.12 | 31 | 1.1% |
| {2.2,4.3} | => | {3.1} | 0.032 | 0.51 | 0.062 | 1.01 | 31 | 1.1% |
| {2.2,4.2} | => | {3.1} | 0.031 | 0.54 | 0.057 | 1.06 | 30 | 1.0% |
| {2.1,4.2,5.1} | => | {3.1} | 0.03 | 0.67 | 0.044 | 1.34 | 29 | 1.0% |
| {2.1,4.3,5.1} | => | {3.1} | 0.029 | 0.61 | 0.047 | 1.21 | 28 | 1.0% |
| {2.2,4.1} | => | {3.1} | 0.029 | 0.52 | 0.055 | 1.03 | 28 | 1.0% |
| {2.1,4.1,5.1} | => | {3.1} | 0.027 | 0.68 | 0.039 | 1.36 | 26 | 0.9% |
| {1.1,2.1,5.1} | => | {3.1} | 0.027 | 0.62 | 0.043 | 1.23 | 26 | 0.9% |
| {2.2,4.3,5.1} | => | {3.1} | 0.027 | 0.6 | 0.044 | 1.20 | 26 | 0.9% |
| {1.4,2.2} | => | {3.1} | 0.027 | 0.48 | 0.055 | 0.96 | 26 | 0.9% |

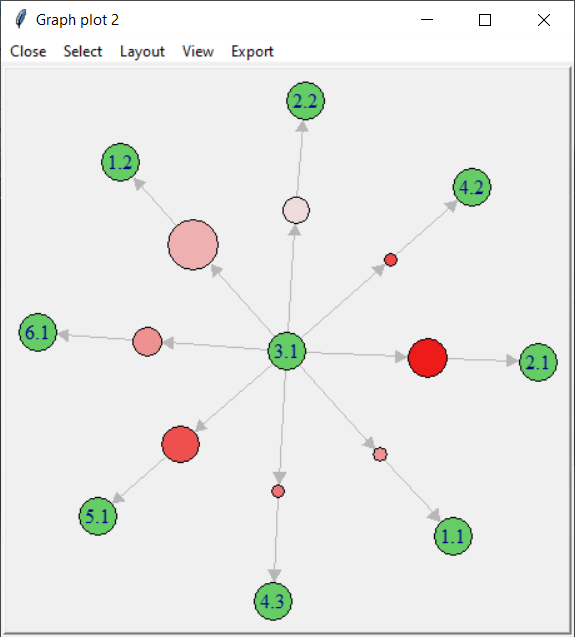
Note: The following counts and percentages and are not mutually exclusive, so there is overlap. *1.2 Safety Assessment from SHIELD* was omitted from the below counts

Combining the two sets of rules with lift >1 and sorted by count produces the following top 25 rules. Rules that indicated (=>) *3.1 Narrative Documentation* errors left-hand side

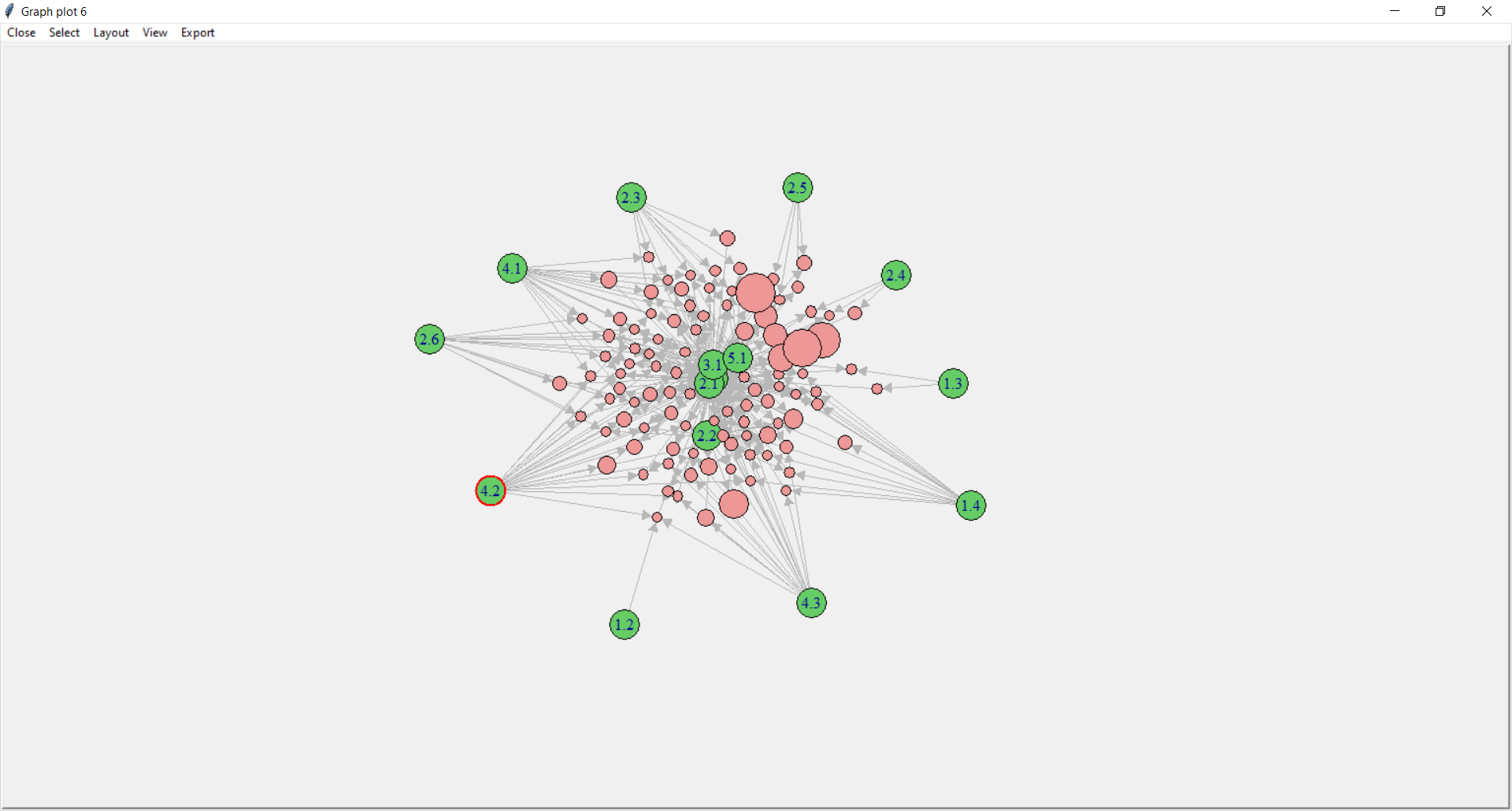
* *2.0 Investigation* errors occurred in 56% of the top 25 rules and 31% of cases read (988/3215 cases), and the specific error,
* *2.1, Interviews: AV, AP Medical Professionals, Collaterals and Reporter*, occurred in 36% of the top 25 rules and 21.9% of cases read (705/3215 cases).
* *4.0 Service Provisions and Outcomes* errors occurred in 40% of the top 25 rules and 17.8% of cases read (573/3215 cases).
* *5.1 Appropriate Timeliness of Investigation Elements* errors occurred in 40% of the top 25 rules and 23.1% of cases read (742/3215 cases).

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **lhs** |  | **rhs** | **support** | **confidence** | **coverage** | **lift** | **count** | **% of cases** |
| {2.1} | => | {3.1} | 0.264 | 0.60 | 0.440 | 1.19 | 258 | 8.9% |
| {5.1} | => | {3.1} | 0.249 | 0.57 | 0.440 | 1.10 | 243 | 8.4% |
| {2.1,5.1} | => | {3.1} | 0.136 | 0.62 | 0.221 | 1.22 | 133 | 4.6% |
| {2.2,5.1} | => | {3.1} | 0.102 | 0.54 | 0.189 | 1.08 | 100 | 3.5% |
| {1.1} | => | {3.1} | 0.089 | 0.52 | 0.170 | 1.03 | 87 | 3.0% |
| {4.2} | => | {3.1} | 0.083 | 0.57 | 0.150 | 1.13 | 81 | 2.8% |
| {4.3} | => | {3.1} | 0.081 | 0.54 | 0.150 | 1.07 | 79 | 2.7% |
| {4.1} | => | {3.1} | 0.066 | 0.58 | 0.110 | 1.15 | 64 | 2.2% |
| {2.1,4.3} | => | {3.1} | 0.055 | 0.64 | 0.086 | 1.28 | 54 | 1.9% |
| {2.1,4.2} | => | {3.1} | 0.054 | 0.66 | 0.082 | 1.31 | 53 | 1.8% |
| {2.1,2.2} | => | {3.1} | 0.054 | 0.54 | 0.100 | 1.07 | 53 | 1.8% |
| {1.1,2.1} | => | {3.1} | 0.052 | 0.62 | 0.084 | 1.23 | 51 | 1.8% |
| {4.3,5.1} | => | {3.1} | 0.048 | 0.59 | 0.082 | 1.17 | 47 | 1.6% |
| {2.5} | => | {3.1} | 0.046 | 0.53 | 0.087 | 1.05 | 45 | 1.6% |
| {4.1,5.1} | => | {3.1} | 0.045 | 0.66 | 0.069 | 1.30 | 44 | 1.5% |
| {4.2,5.1} | => | {3.1} | 0.042 | 0.59 | 0.071 | 1.18 | 41 | 1.4% |
| {2.4} | => | {3.1} | 0.042 | 0.59 | 0.072 | 1.16 | 41 | 1.4% |
| {1.1,5.1} | => | {3.1} | 0.041 | 0.53 | 0.078 | 1.04 | 40 | 1.4% |
| {2.1,4.1} | => | {3.1} | 0.039 | 0.63 | 0.061 | 1.26 | 38 | 1.3% |
| {2.3,5.1} | => | {3.1} | 0.036 | 0.51 | 0.070 | 1.02 | 35 | 1.2% |
| {2.1,2.2,5.1} | => | {3.1} | 0.035 | 0.58 | 0.060 | 1.14 | 34 | 1.2% |
| {1.4,5.1} | => | {3.1} | 0.035 | 0.51 | 0.069 | 1.01 | 34 | 1.2% |
| {1.4,2.1} | => | {3.1} | 0.032 | 0.58 | 0.054 | 1.16 | 31 | 1.1% |
| {2.6,5.1} | => | {3.1} | 0.032 | 0.56 | 0.056 | 1.12 | 31 | 1.1% |
| {2.2,4.3} | => | {3.1} | 0.032 | 0.51 | 0.062 | 1.01 | 31 | 1.1% |

Some of the associations were plotted. An example follows.



And then there is this plot which shows a complex relationship between errors.



### Conclusions for ARM

#### Interesting rules

*1.2 Safety Assessments from SHIELD* errors were the “big winners.” This is very likely due to the pandemic and the subsequent policy changes made to adapt casework to the pandemic with no corresponding changes to the scoring rubrics. This error reflects the ability of a caseworker to meet the requirements and make a face-to-face visit to a client.

The combination of the sets of rules with lift >1 and sorted by count produced rules that pointed to (=>) *3.1 Narrative Documentation* errors had on the left-hand side

* *2.0 Investigation* errors occurred in 56% of the top 25 rules and 31% of cases read (988/3215 cases), and the specific error,
* *2.1, Interviews: AV, AP Medical Professionals, Collaterals and Reporter*, occurred in 36% of the top 25 rules and 21.9% of cases read (705/3215 cases).
* *4.0 Service Provisions and Outcomes* errors occurred in 40% of the top 25 rules and 17.8% of cases read (573/3215 cases).
* *5.1 Appropriate Timeliness of Investigation Elements* errors occurred in 40% of the top 25 rules and 23.1% of cases read (742/3215 cases).

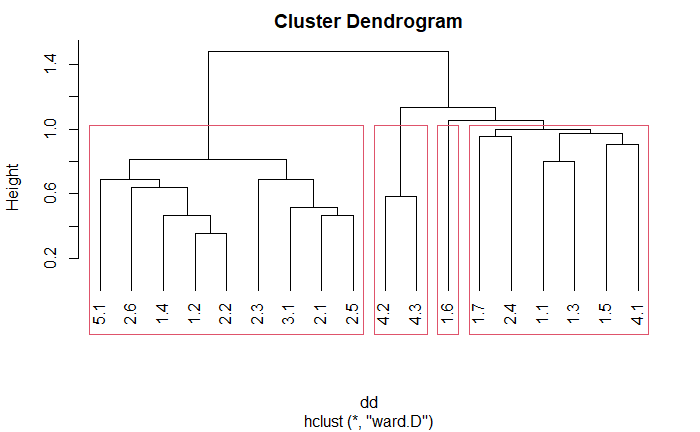
This is interesting as one might expect *3.1 Narrative Documentation* errors to be unaffected by the pandemic.

As stated previously, the associations of 1.2 and 3.1 are interesting but not explored in this analysis. For many of the other associated errors, it does follow that if the documentation is poor, then the case reader may not be able to determine if the requirements were met.

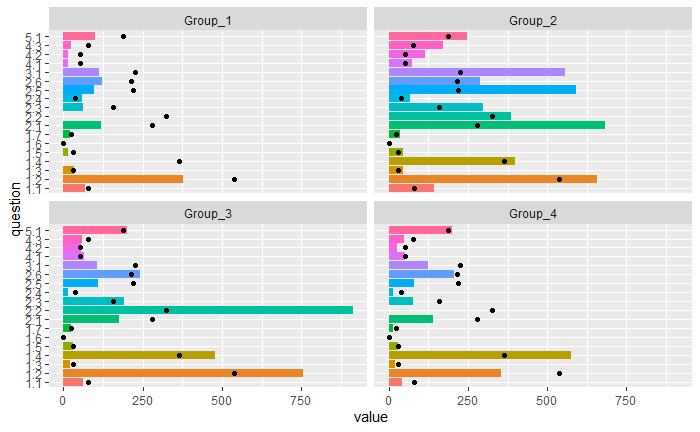
### Clustering

Clustering is used to discover if groups/categories exist and, if so, what they are. If categories can be identified, this information can then be used to classify or predict other vectors. In a very simple way, clustering should group data vectors that are “similar” and at the same time, maintain a dissimilarity between clusters.

Hierarchical clustering produces a set of nested clusters organized as a hierarchical tree-like the one shown below. In this dendrogram, the distance was measured with cosine similarity.

****

Once clusters are generated, we can begin to analyze them in a business context. The graph below is a representation of four clusters generated by k-means clustering. The horizontal bars indicate the sum of errors in its respective cluster. The black points represent the average of the errors across the entire data set. Group one has only one error (2.4) which occurs more than the average. Group two, in contrast, has above-average rates for all errors. Group 3 differs in that it consists of greater than the average of 1.2, 1.4, 2.2, 2.3, 2.6, 4.1, and 5.1 errors. Finally, group 4 consists of above-average rates for 1.4 and 5.1 errors.

****

## Predictive Models

First the data is split into two sections: a training and a testing set of data called train and test.

df <- data[,-1:-2]

df <- df[,-19:-24]

df <- as.data.frame(df)

colnames(df) <- paste("error",colnames(df),sep="\_")

df$error\_1.6 <- as.numeric(df$error\_1.6)

df[is.na(df)] <- 0

df[df>0] <- 1

df <- df %>% mutate\_all(as.factor)

set.seed(10)

smp\_size <- floor(0.75\*nrow(df))

train\_idx <- sample(seq\_len(nrow(df)),size=smp\_size)

train <- df[train\_idx,]

test <- df[-train\_idx,]

Now the team is ready to build the models.

### Naive Bayes

Creating the Naive Bayes Model to test error 3.1

nb\_model <- naiveBayes(error\_X3.1~., data=train)

Using the Naive Bayes model to predict the error in the test set.

pred\_nb <- predict(object = nb\_model, select(test,-error\_X3.1), type = "class")

Creating the confusion matrix for the model

conf\_mat\_nb <- confusion\_matrix(targets = test$error\_X3.1, predictions = pred\_nb)

plot\_confusion\_matrix(

conf\_mat\_nb$`Confusion Matrix`[[1]],

font\_counts = font(

size = 10,

angle = 45,

color = "red"

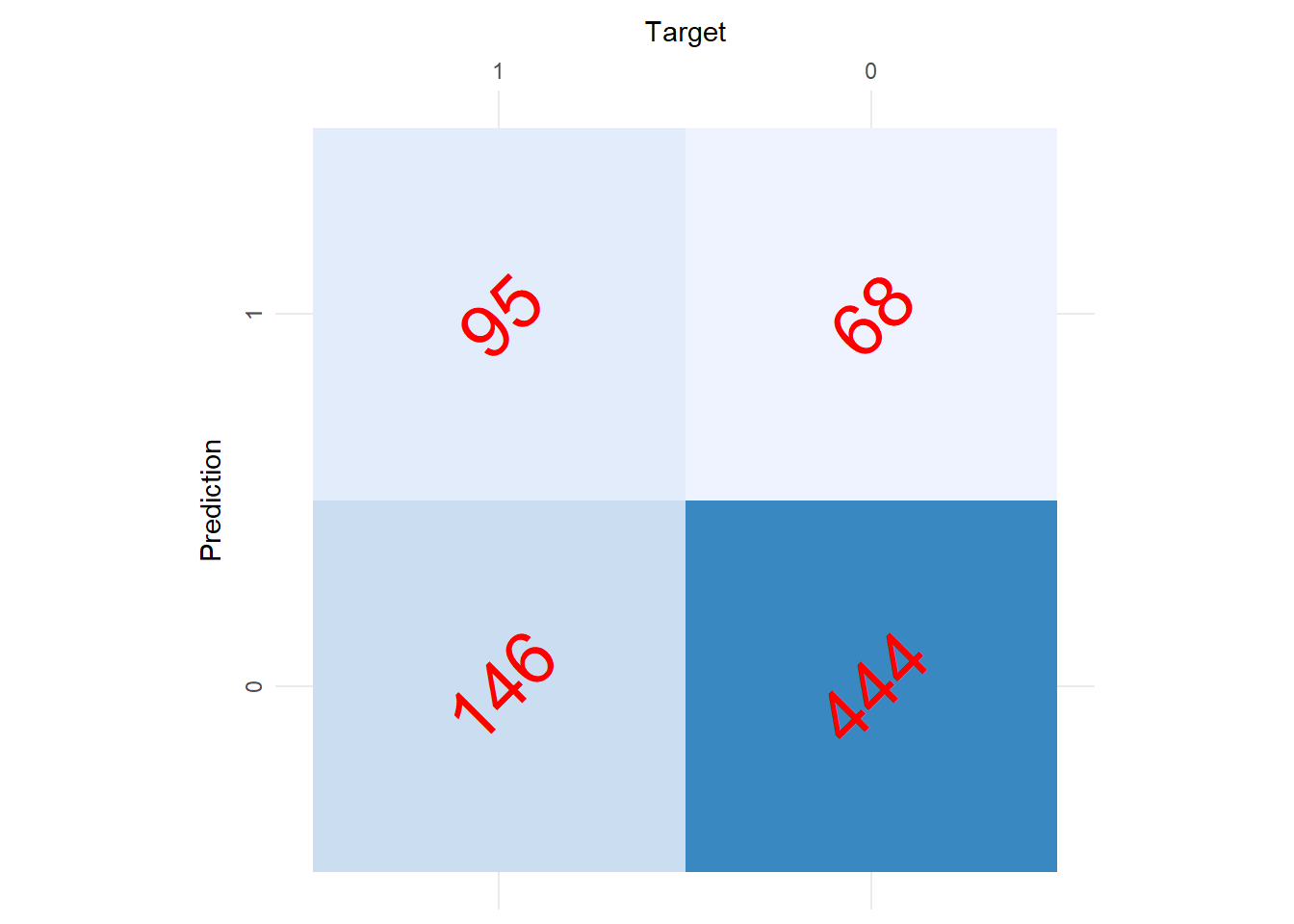
),

add\_normalized = FALSE,

add\_col\_percentages = FALSE,

add\_row\_percentages = FALSE,

)



# credit for plots and confusion matrix (<https://cran.r-project.org/web/packages/cvms/vignettes/Creating_a_confusion_matrix.html>)

paste("Balanced Accuracy",conf\_mat\_nb$`Balanced Accuracy`)

## [1] "Balanced Accuracy 0.630689185684647"

Here balanced accuracy is looked at since overall accuracy is the absolute accuracy of the model predicting the results, but the balanced accuracy has taken into account that the test and train data sets were not the same size and therefore gives a better result as our test and train data sets were not the same size. see from the confusion matrix that the accuracy is very good at predicting when there isn’t an error, but this is not bad since the error 3.1 happens only 30% of the time in our data set.

### SVM Model

The SVM models for 3.1 were checked with linear, polynomial, and radial kernels but the results were all worse than the kernel that was chosen, a sigmoid kernel.

Now checking the SVM model with a sigmoid Kernel and a cost of 15. The cost was chosen by just sampling a few different cost values and picking the best one.

First, create the model.

model\_SVM\_sigmoid <-svm(error\_X3.1~.,data=train, kernel = "sigmoid", cost = 15)

Predicting the test data with the SVM model

pred\_svm\_sigmoid <- predict(model\_SVM\_sigmoid,select(test,-error\_X3.1),type = "class")

Creating the confusion matrix

conf\_mat\_sigmoid <- confusion\_matrix(targets = test$error\_X3.1, predictions = pred\_svm\_sigmoid)

plot\_confusion\_matrix(

conf\_mat\_sigmoid$`Confusion Matrix`[[1]],

font\_counts = font(

size = 10,

angle = 45,

color = "red"

),

add\_normalized = FALSE,

add\_col\_percentages = FALSE,

add\_row\_percentages = FALSE,

******

paste("Balanced Accuracy",conf\_mat\_sigmoid$`Balanced Accuracy`)

## [1] "Balanced Accuracy 0.611040424014523"

The SVM model resulted in a 61% accuracy.

### Knn Model

The team selected k to be the square root of the number of training data points. (This is a common K value to use.)

k <- round(sqrt(nrow(train)))

Creating the Knn model

train\_model\_k <- knn(train = select(train,-error\_X3.1), test = select(test,-error\_X3.1), cl = train$error\_X3.1, k = k, prob= TRUE)

Creating the confusion matrix for the Knn model and the test data set.

conf\_mat\_k <- confusion\_matrix(targets = test$error\_X3.1, predictions = train\_model\_k)

plot\_confusion\_matrix(

conf\_mat\_k$`Confusion Matrix`[[1]],

font\_counts = font(

size = 10,

angle = 45,

color = "red"

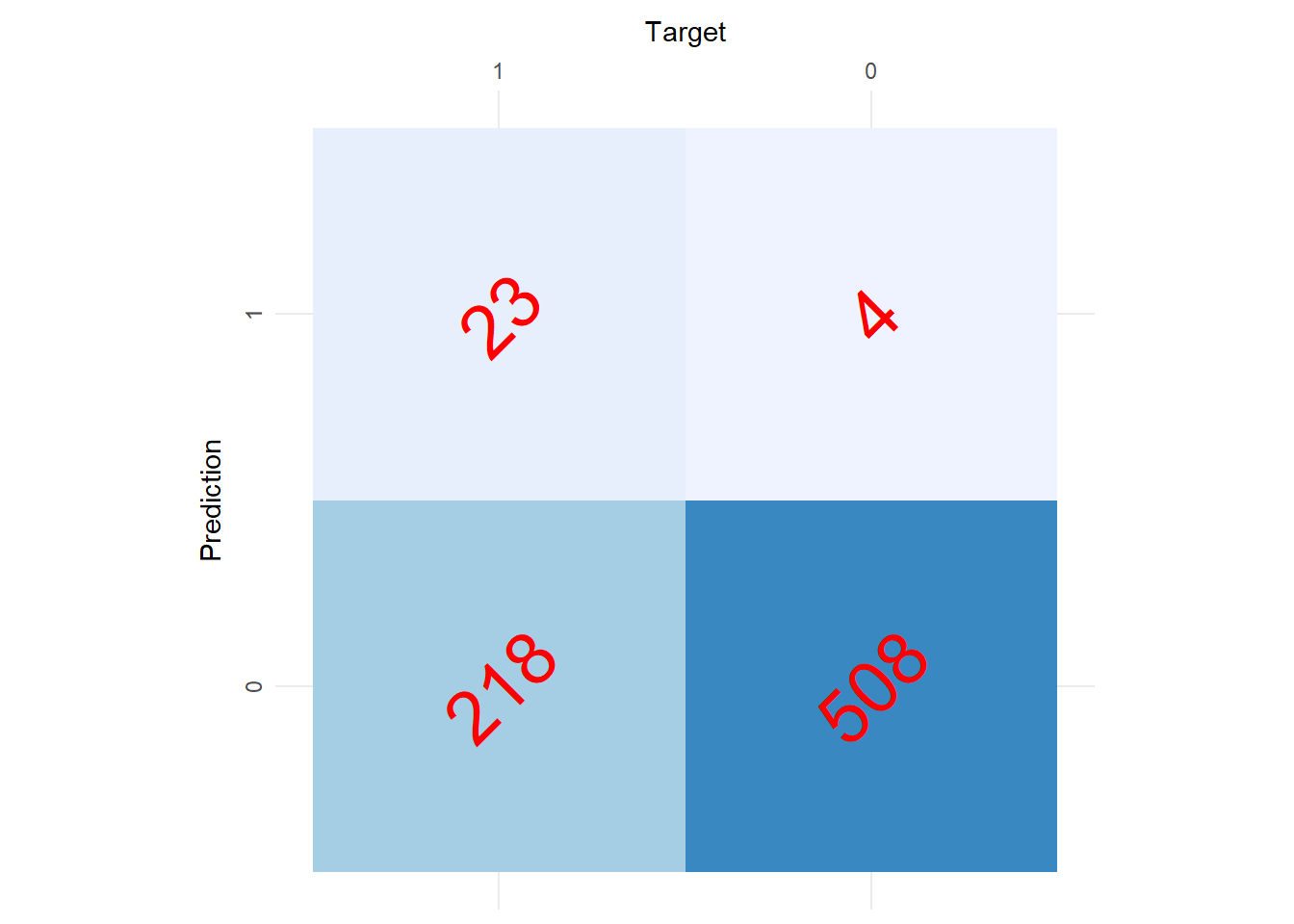
),

add\_normalized = FALSE,

add\_col\_percentages = FALSE,

add\_row\_percentages = FALSE,

)

paste("Balanced Accuracy",conf\_mat\_k$`******Balanced Accuracy`)

## [1] "Balanced Accuracy 0.543811592323652"

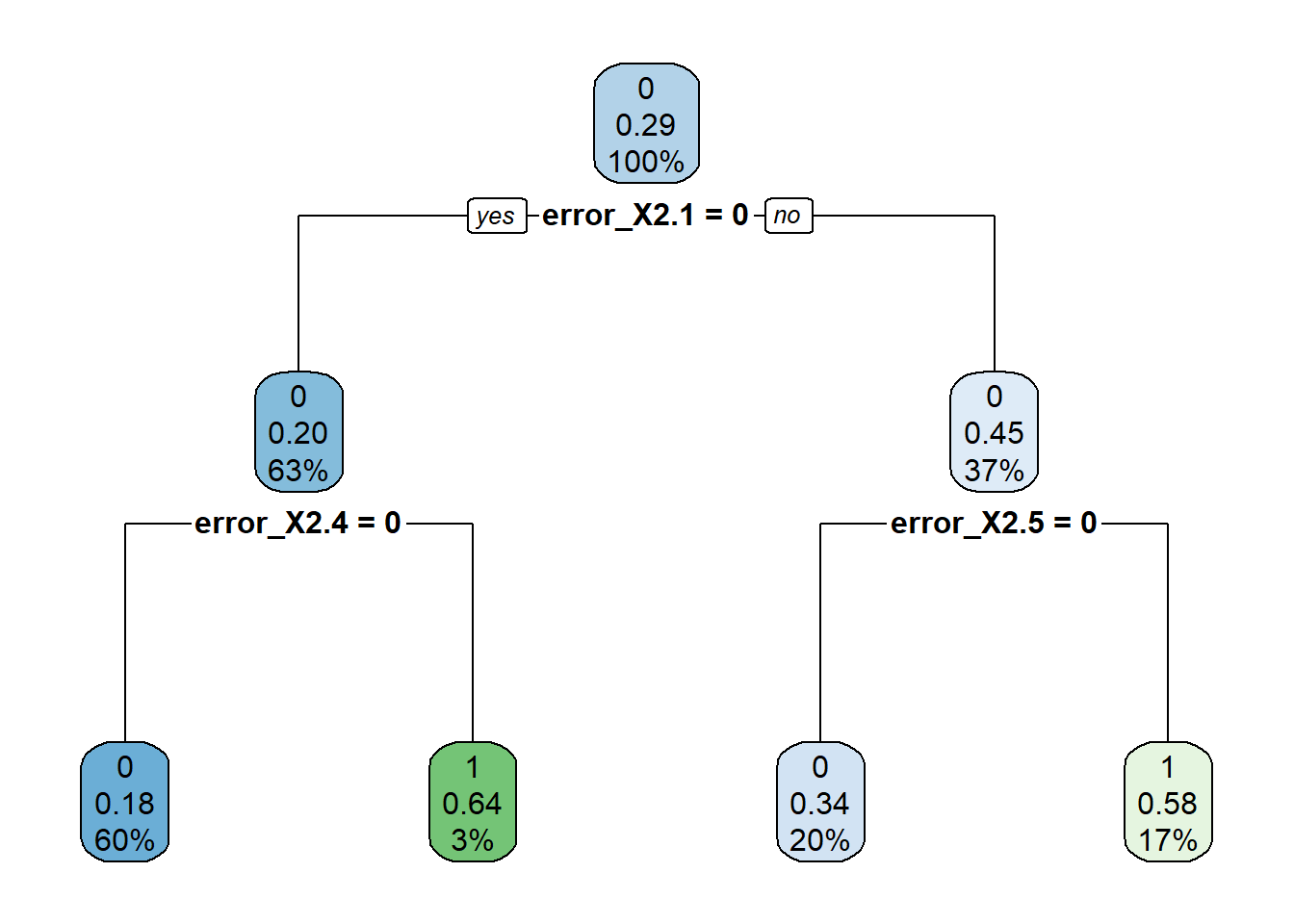
The Knn model resulted in 54% accuracy. A poor result as the worst models result in a 50% accuracy where they predict that error 3.1 never happens. Since this accuracy was so low no further investigation was done to try to improve it.

### Decision Tree Model

Now using a decision tree. The team created the tree and evaluated the variable importance.

train\_model <- rpart(error\_X3.1 ~., data = train, method = "class")

rpart.plot(train\_model)

******

df <- data.frame(imp = train\_model$variable.importance)

df2 <- df %>%

tibble::rownames\_to\_column() %>%

dplyr::rename("variable" = rowname) %>%

dplyr::arrange(imp) %>%

dplyr::mutate(variable = forcats::fct\_inorder(variable))

ggplot2::ggplot(df2[c((nrow(df2)-8):nrow(df2)),]) +

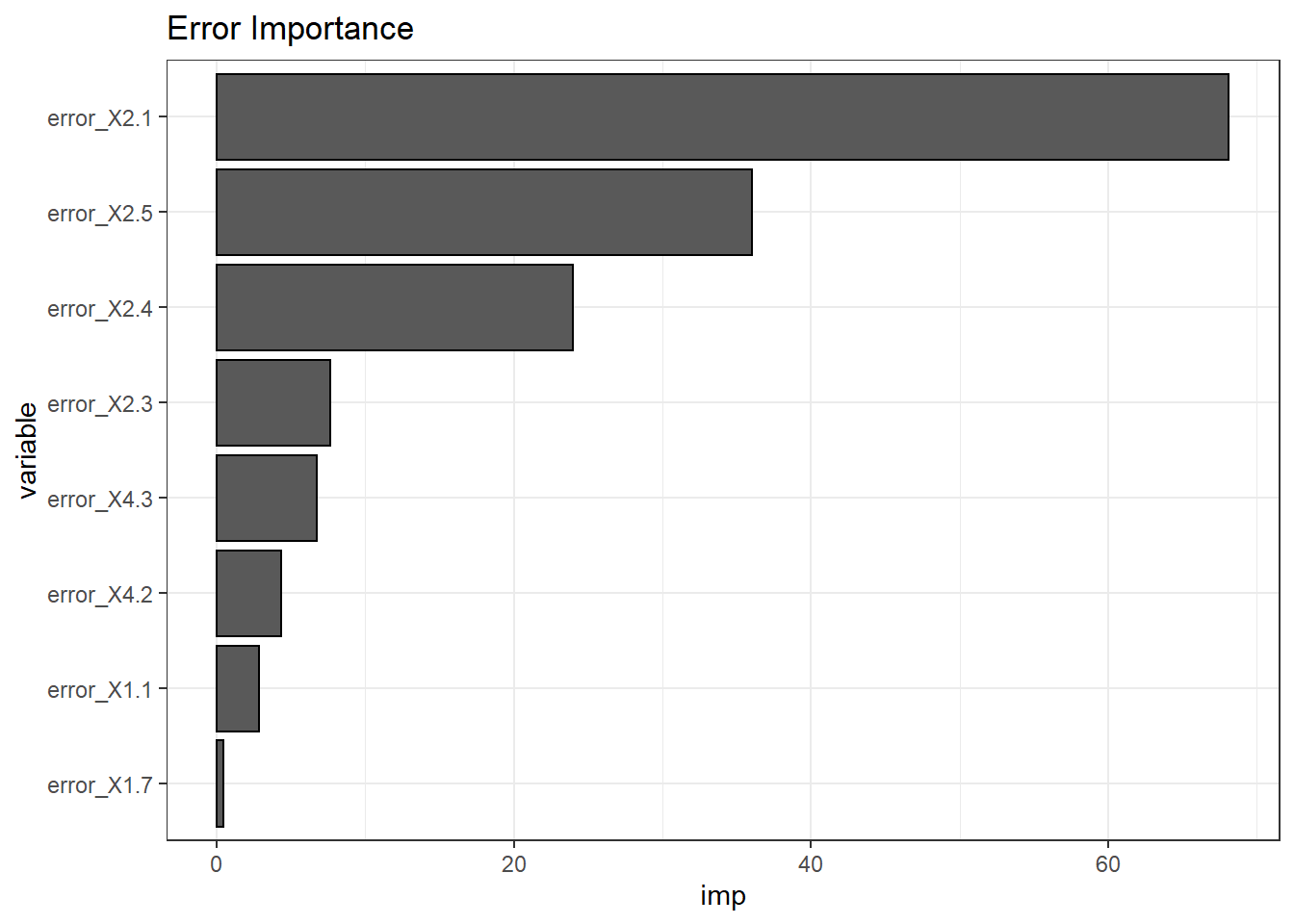
geom\_col(aes(x = variable, y = imp),

col = "black", show.legend = F) +

coord\_flip() +

scale\_fill\_grey() +

theme\_bw() +ggtitle("Error Importance")

******

Error 2.1 is very important when predicting error 3.1, and that error 1.7 does not drive error 3.1. Now to predict the results against the test data and determine the accuracy of the model.

pred <- predict (object = train\_model,select(test,-error\_X3.1),type = "class")

conf\_mat\_Dtree <- confusion\_matrix(targets = test$error\_X3.1, predictions = pred)

plot\_confusion\_matrix(

conf\_mat\_Dtree$`Confusion Matrix`[[1]],

font\_counts = font(

size = 10,

angle = 45,

color = "red"

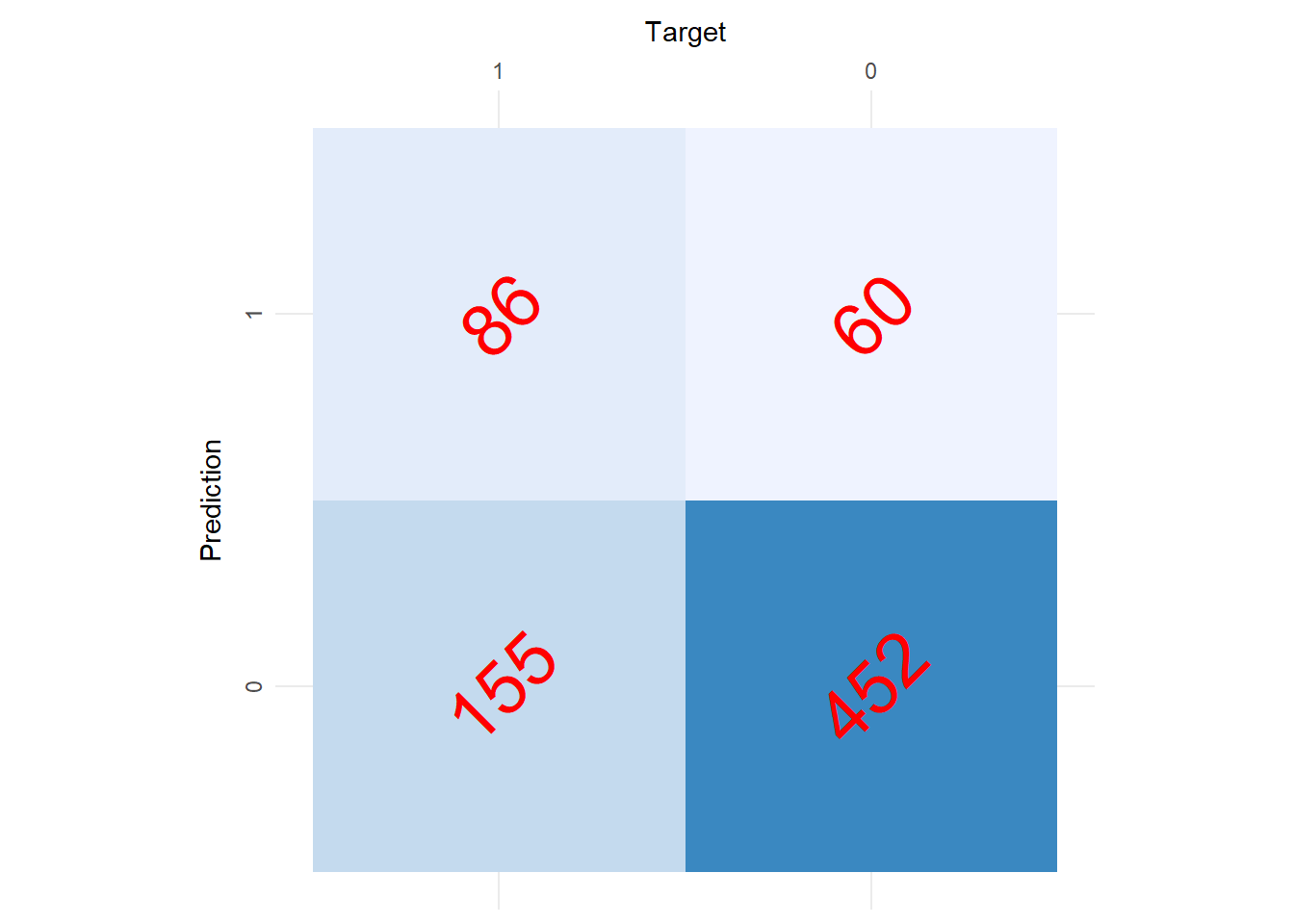
),

add\_normalized = FALSE,

add\_col\_percentages = FALSE,

add\_row\_percentages = FALSE,

)

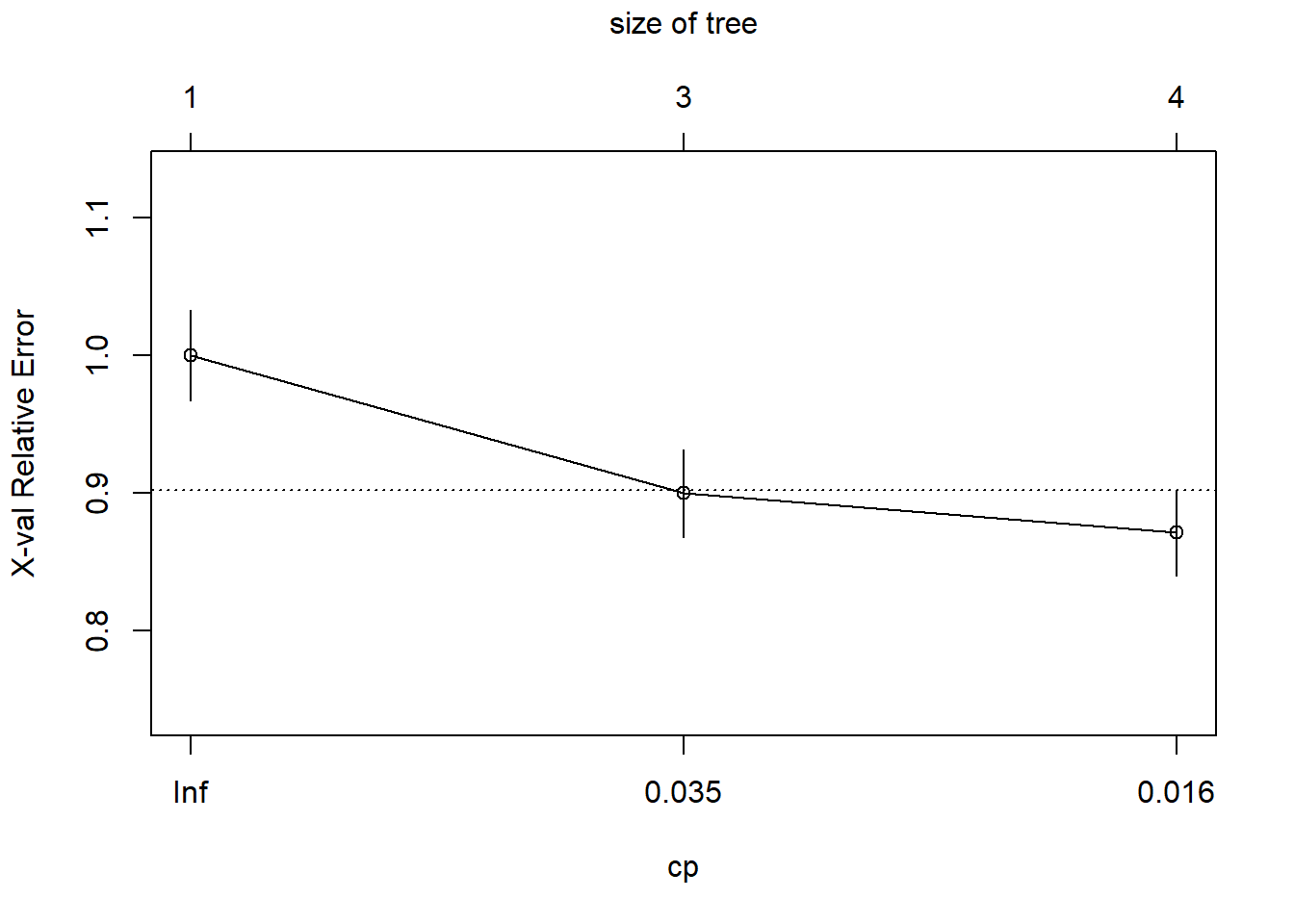
******

paste("Balanced Accuracy",conf\_mat\_Dtree$`Balanced Accuracy`)

## [1] "Balanced Accuracy 0.619829486514523"

62% accuracy for the decision tree. Notice the tree only uses three different nodes for predicting error 2.1, these nodes are errors 2.1, 2.5, and 2.4. This is a relatively low complexity tree, but looking at the plot below models that generate a higher accuracy produce higher error rates. This could be indicative of the complexity of the error interrelationships that were shown during ARM (refer to ARM graphic).

plotcp(train\_model)

******

### Random Forest

Using the randomForest library, a random forest algorithm was run on the training data set. See below for results.

model <- randomForest(error\_3.1~.,data=train,ntree=100,mtry=5,importance=TRUE)

model

##

## Call:

## randomForest(formula = error\_3.1 ~ ., data = train, ntree = 100, mtry = 5, importance = TRUE)

## Type of random forest: classification

## Number of trees: 100

## No. of variables tried at each split: 5

##

## OOB estimate of error rate: 28%

## Confusion matrix:

## 0 1 class.error

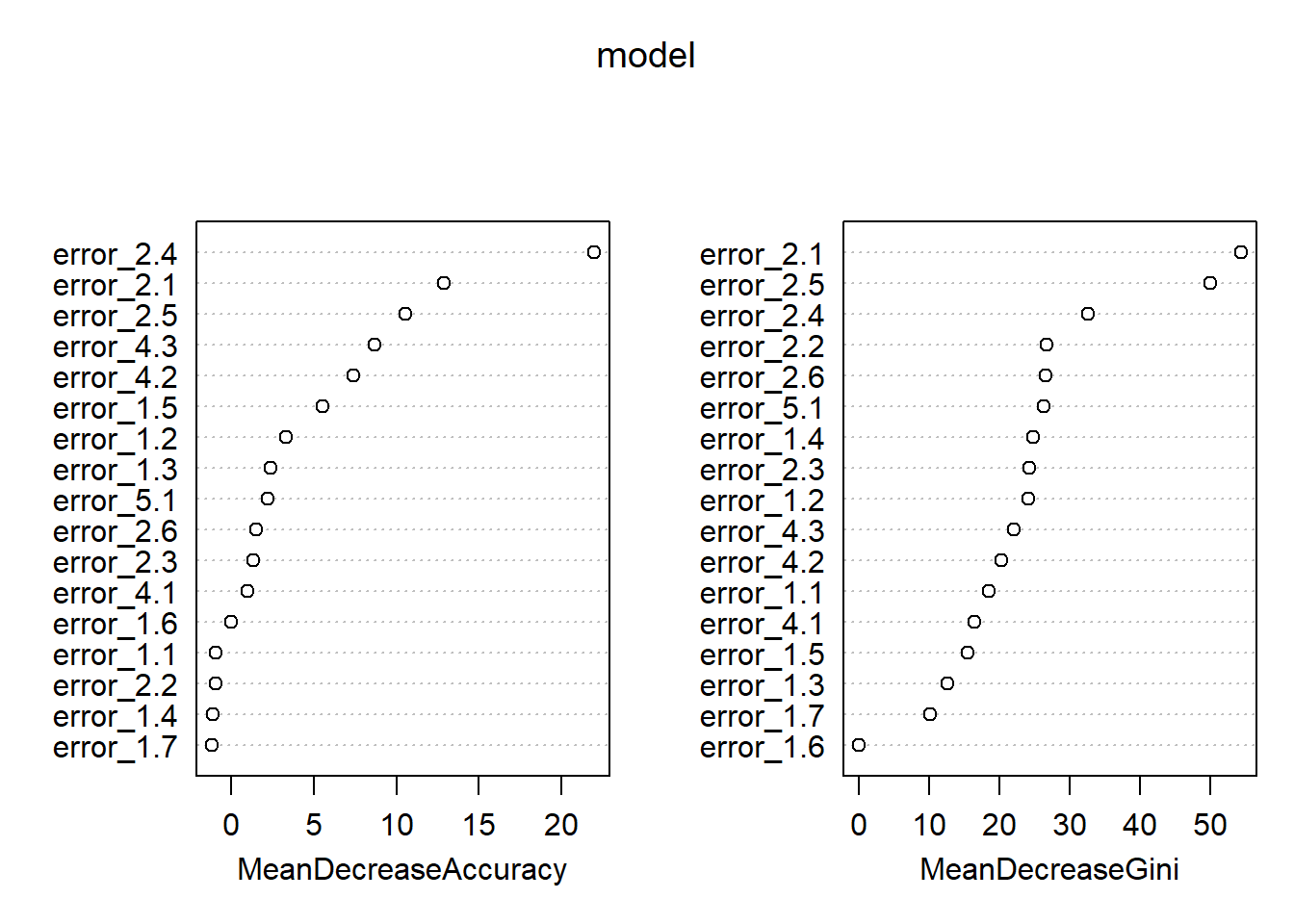
## 0 1423 174 0.1089543

## 1 458 202 0.6939394

Like in previous models, error 3.1 Narrative Documentation is the error the model was trying to predict. One hundred trees were generated to minimize the OOB estimate of the error rate. Also, five random features were used in the construction of each tree. As shown above, by the OOB estimate of error rate, this model misclassified 28% of 3.1 errors in the training data set.

The “importance” parameter enables the algorithm to calculate variable importance. Each variable’s importance metric is plotted below.

varImpPlot(model)



MeanDecreaseAccuracy gives a rough estimate of the loss in prediction performance when that particular variable is omitted from the training set. Gini is a measure of node impurity. Higher purity means that each node contains only elements of a single class. Assessing the decrease in Gini when that feature is omitted leads to an understanding of how important that feature is to split the data correctly. These graphs agree with the decision tree variable importance, 2.1, 2.4, and 2.5 are the most important.

Next, the team used cross-validation to determine the accuracy of the model. The confusion matrix is shown below which used the model’s predictions and the observed data in the testing data set. Concluding that the random forest model correctly predicted 3.1 errors with 71.18% accuracy.

pred <- predict(model,test[-14])

table(observed=test$error\_3.1,predicted=pred)

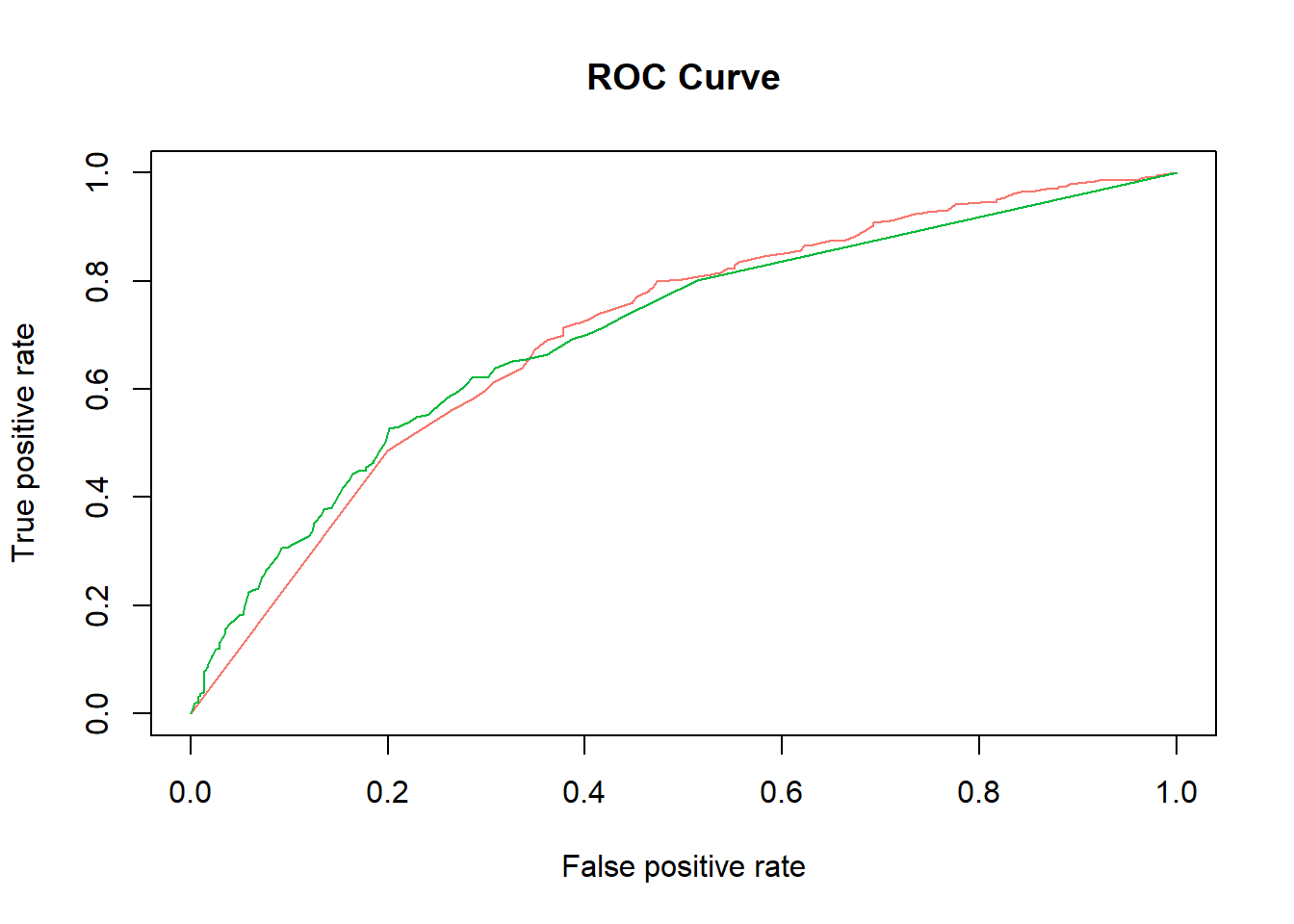
## predicted

## observed 0 1

## 0 466 46

## 1 171 70

Another way of assessing the performance of the model is to generate a ROC curve. A ROC curve (receiver operating characteristic curve) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters: true positive rate and false positive rate of each “class” (0 and 1 in this case). The area under the curve (also referred to as AUC) provides an aggregate measure of performance across all possible classification thresholds. The AUC of our model was calculated at 70.53%.

******

[**https://www.blopig.com/blog/2017/04/a-very-basic-introduction-to-random-forests-using-r/**](https://www.blopig.com/blog/2017/04/a-very-basic-introduction-to-random-forests-using-r/)

[**https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc**](https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc)

### Text Mining

Basic text mining was conducted, and the following word clouds were produced.

In analyzing the data, the team created a word cloud of the error codes to see if there was a pattern that coincided with other analyses, including the Apriori analyses and the frequency chart above (as it should).

As one can see, the most errors occurred in category *1.2* *Safety Assessment from SHIELD*, which aligns with the other analyses. 

Analyzing the words used in the comments, the first cloud included errors 6-11 to determine if there were any visual differences. The second cloud omits 6-11. The third cloud used additional stop words that were related to all the error comments. Word clouds were generated with and without stemming

STOPS <-stopwords('english')

x <- c("1.1","1.2", "1.3", "1.4", "1.5", "1.6", "1.7", "2.1","2.2", "2.3", "2.4", "2.5", "2.6","3.1", "4.1", "4.2", "4.3", "5.1")

myStopwords <- c("and", "the", "that", "she", "client", "case", "apss",  
"yes", "aps","error", "No", "documentation", "document")

## Wordcloud for All

group\_filter <- filter(QAll, Question == "1.0.2")

qQuest <- VCorpus(VectorSource(group\_filter$Comment))

qDTM <- DocumentTermMatrix(qQuest, control = list(removePunctuation = F, tolower = T, stemming = F, remove\_seperators = T, stopwords = c(STOPS, myStopwords, x) ))

#### Without stemming

DTM <- as.matrix(qDTM)

qDTM <- as.matrix((qDTM))

totDTM <- colSums(DTM)

qtotDTM <- colSums(qDTM)

wordcloud(colnames(DTM),totDTM,rot.per=0.35,colors=brewer.pal(8, "Dark2"),

random.order=FALSE, scale = c(3,1), max.words = 50)

## With stemming

qDTM <- DocumentTermMatrix(qQuest, control = list(removePunctuation = F, tolower = T, stemming = T, remove\_seperators = T,stopwords = c(STOPS, myStopwords)))

#### Without 1.2

DTM <- as.matrix(qDTM)

qDTM <- as.matrix((qDTM))

totDTM <- colSums(DTM)

qtotDTM <- colSums(qDTM)

wordcloud(colnames(DTM),totDTM,rot.per=0.35,colors=brewer.pal(8, "Dark2"),

 random.order=FALSE, scale = c(3,1), max.words = 50)

### Word Cloud by Standard 1, 2, 3, 4, 5

xx <- c("1.0.2", "2.0.2", "3.0.2", "4.0.2", "5.0.2")

for (val in xx) {

group\_filter <- filter(QAll, Question == val)

qQuest <- VCorpus(VectorSource(group\_filter$Comment))

qDTM <- DocumentTermMatrix(qQuest, control = list(removePunctuation = F, tolower = T, stemming = F, remove\_seperators = T,stopwords = c(STOPS, myStopwords, x)))

DTM <- as.matrix(qDTM)

qDTM <- as.matrix((qDTM))

totDTM <- colSums(DTM)

qtotDTM <- colSums(qDTM)

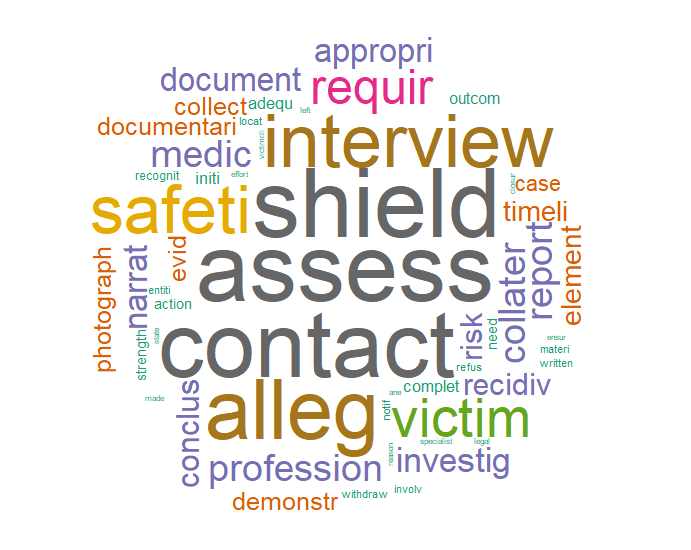
wordcloud(colnames(DTM),totDTM,rot.per=0.35,colors=brewer.pal(8, "Dark2"),

random.order=FALSE, scale = c(3,1), max.words = 50)

print(val)

}

Note: SHIELD is an assessment tool for caseworkers. The data shows that that tool may not be completed correctly. This is in addition to what Apriori indicated. Coupled together, this provides a starting point for process improvement.



Looking at the third cloud, policy errors jump out. Additional “pruning” with added stop words should be looked at further, if only for a graphic that will start the process improvement conversation.

## Results

### ARM

*1.2 Safety Assessments from SHIELD* errors were the “big winners.” This is very likely due to the pandemic and the subsequent policy changes made to adapt casework to the pandemic with no corresponding changes to the scoring rubrics. This error reflects the ability of a caseworker to meet the requirements and make a face-to-face visit to a client.

What is interesting are the *3.1 Narrative Documentation* errors. These are errors that one would expect would not be influenced by the pandemic.

The associations with *1.2 Safety Assessments from SHIELD* and *3.1 Narrative Documentation* are interesting. It was unexpected that *3.1* errors would be associated with *1.2* which is a separate form. For many of the other associated errors, it does follow that if the documentation is poor, then the case reader may not be able to determine if the requirements were met.

### Decision Tree

The decision tree resulted in a 62% accuracy. This also showed that the errors *2.5 Allegation Conclusion*, *2.1 Interviews: AV, AP Medical Professionals, Collaterals and Reporter*, and *2.4 Recognition and Investigation of Allegations* have the most importance in predicting error *3.1 Narrative Documentation*. If 2.1 exists then *2.5 Allegation Conclusion* is important, but if 2.1 does not exist then 2.4 is important for the prediction. While the decision tree only has 3 nodes, looking at the CP graph we can see that more nodes do not make the model better, and we see 62% hard to improve with a decision tree model.

### Naive Bayes

The Naive Bayes model was able to predict error *3.1 Narrative Documentation* with an accuracy of 63%. With the data being 1’s and 0’s Naive Bayes typically is not a good model which is a factor that influences the accuracy. This is because Naive Bayes uses smoothing since 0 valued entries are a problem for the model.

### Knn

The Knn model achieved an accuracy of 54%. This was using the square root of the number of data points as the K value, a typical K value used. This ended up with a very poor result. We view Knn as simply not a good model for the data. This is possibly because the data is very spread out i.e., there are sometimes few errors per row of the data set.

### Random Forest

The Random Forest model achieved an accuracy of 71.18%. Like in previous models, error *3.1 Narrative Documentation* is the error the model was trying to predict. One hundred trees were generated to minimize the OOB estimate of the error rate. Also, five random features were used in the construction of each tree. The team used cross-validation to determine the accuracy of the model.

### SVM

The SVM model resulted in a 61% accuracy. This was with a sigmoid kernel and a cost value of 15. This cost was derived by sampling many cost functions individually, and multiple kernels were run to find the best one. I believe this accuracy value could be increased with a custom kernel, but an attempt to make or find one was not done during this analysis.

**Text Mining**

The text mining proved inconclusive as the same words were used throughout the comments. The word Policy did pop out, but every standard has an associated policy.

### Conclusions

Random Forest produced the highest accuracy rate at 71.18% when predicting *3.1 Narrative Documentation* errors. What we could not determine where it was causation or only correlation.

As the data owners suspected, the results produced positive correlations between the errors. In plain English, errors in documenting the cases are related to, and can possibly produce, errors in the investigation or vice-a-versa. These results appear to support their initial hypothesis that improving the quality of the investigation should directly affect documentation errors.

When dealing with people and their evaluations, there is the propensity for a wide range of results as the data deals with humans with varying levels of experience and abilities, at timers not generating high accuracy rates. Therefore, it was encouraging that the Random Forest produced a 71% accuracy rate.

To completely understand the balance between causation and correlation in this dataset will require a more in-depth analysis of the comments.

Recommend another dataset over a similar period (i.e., Jun 2019 - Nov 2019) be evaluated to determine the consistency of results. An added value may be the ability to determine if any effects can be attributed to COVID-19. This should provide additional information as to associations and I may provide standards to target for improvement. These associations may provide a start as to where to look for case improvements.