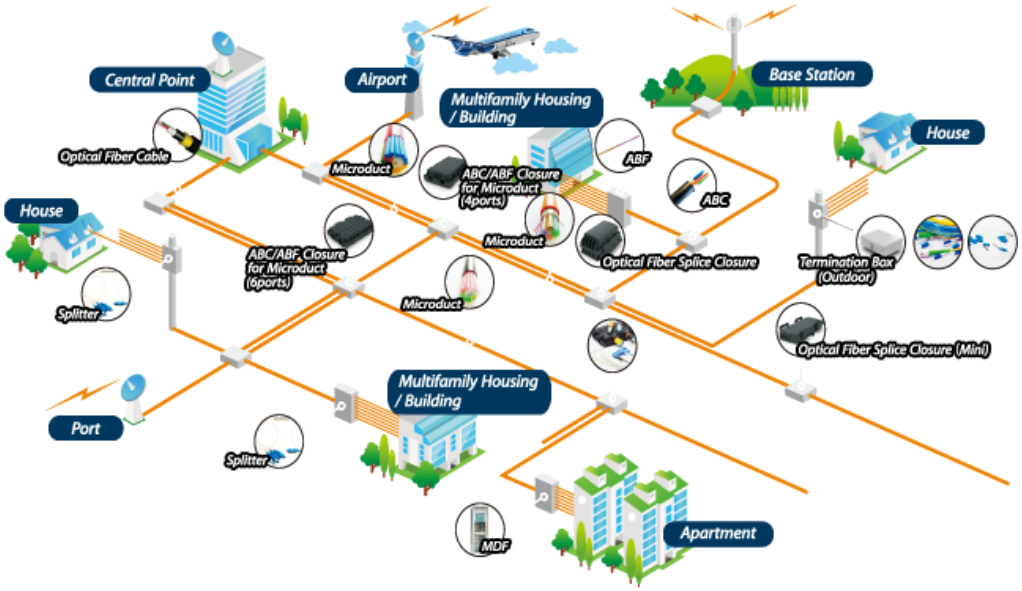
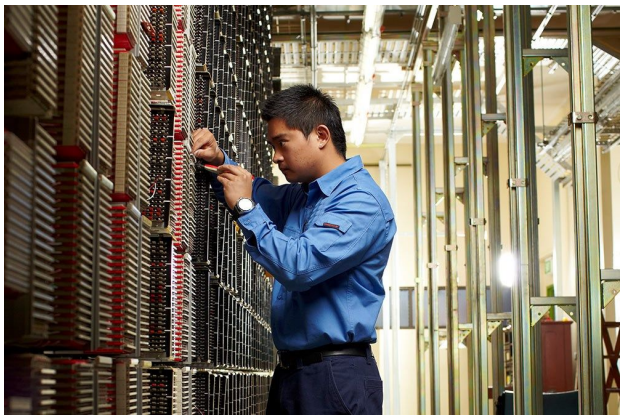
Predicting Network Fault Severity

Final Report

Introduction



In the telephony network, telephone exchanges and nodes are connected by wire, microwaves or via satellite to provide communication service for people to communicate across long distances.



However service disruption can arise from physical damage and network congestion from increased call traffic. When these disruptions occur, the matter could be life and death in emergency situations or delays and losses for businesses. So to ensure network availability to

service customers, Telstra is constantly monitoring the network to ensure the network is available to service customers for their communication needs.



Since these disruptions are done by nature or human activity, their forces can at times can be hard to predict but within Telstra's maintenance system these problems are acknowledged and managed to reduce cost and disruption. When a disruption is present, a log of the error is recorded by software then, if available, signals are redirected to alternate signal paths while repairs are scheduled. Some faults will only disservice customers from a decrease service capability whilst other faults can be a complete shutdown that need immediate attention.


Thus in a business point of view the dilemma is how to allocate resources effectively to reduce repair costs and proactively fix faults before they occur and avoid unnecessary costs.

DataSet

Data has been sourced with Telstra's kaggle competition at address link:
https://www.kaggle.com/c/telstra-recruiting-network
There are 5 out of 6 data tables will be used. The test data has been omitted as there is no data to add to the project.

train.csv - the training set for fault severity with *id , location & faulty_severity** data
test.csv - the test set for fault severity with *id* & *location* data
event_type.csv - event type related to the main dataset with *id* & *event_type** data

log_feature.csv - features extracted from log files with *id , log_feature* & volume* data
resource_type.csv - type of resource related to the main dataset with *id* & *resource_type** data
severity_type.csv - warning message from the log with *id* & *severity_type** data
(* is a variable of discrete nature)

Initial Data Cleaning and Wrangling

The data received were very tidy data where all columns were not data themselves. Also the data itself was relatively clean, where most values are discrete values. The log_feature dataset did require some data extraction as values were joined as one variable. This was separated into 3 columns (id, log_feature & volume) for use. In loading other datasets, all data was extracted as plain strings to cater for manual "factoring" conversion when required. This is so that exploratory analysis can be eased with control of data types. All discrete variables are trimmed to their numeric representation and taken as integers to assist analysis.

Then to gather all data, each dataset is ordered by **id** to join tables together as one. Then the data is summarised to uncover and remove missing data. In this step it is found that the train dataset has less entries than the other tables of log_feature, resource_type, event_type and severity_type. Hence in the later part of the analysis, 2 tables are generated so that all data can be used, *network* and *incident_log*. The difference between the two is that the *incident_log* does not have location nor *fault_severity* columns.

Joined Table contents: ***network***

Data name	Data type	Brief description
id	integer	Record's identifier , time point
location	Integer (factor)	Fault location
faulty_severity	Integer (factor)	3 levels, 0= no fault ; 1 = few ; 2 = many - actual data from reported fault from users
log_feature	Integer (factor)	Assumption: network service's faulting feature
volume	integer	Assumption: unit measure of faulting feature
resource_type	Integer (factor)	10 distinct types
event_type	Integer (factor)	Assumption: Network service feature's fault behaviour
severity_type	Integer (factor)	5 unordered types - warning message received from monitoring machines

Limitations and Assumptions

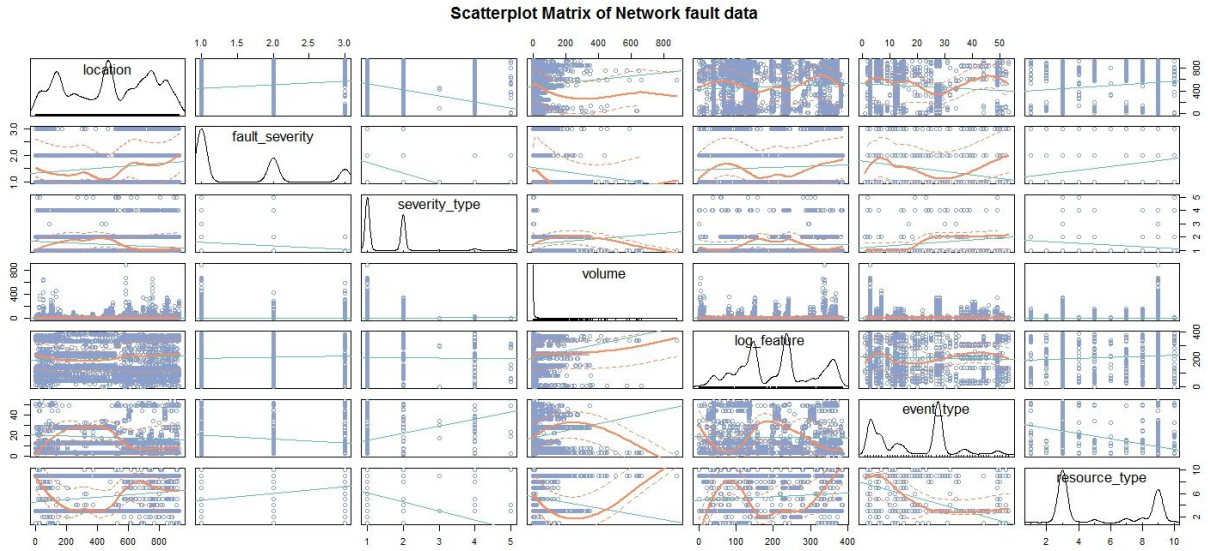
Log_feature data's relevance have not been described by Telstra nor Kaggle forums. It will be assumed to describe the description of the network service's contributing output since each entry is paired with volume data.

All variables are coded, so we will not be able to deduce specific reasons but only assumptions placed by the variables names.

Preliminary Exploration

Scatterplot matrix

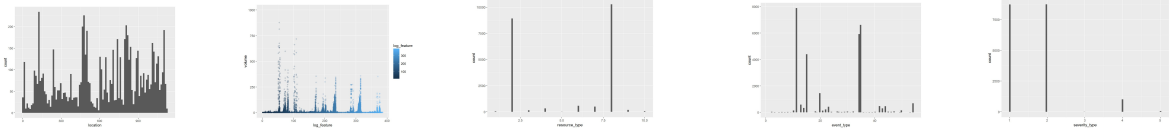
each dataset to understand shape and distribution, bins width set to 100 to reveal if there is distinct cat



Histograms

Each dataset does reveal discrete data and shows each set's range and mode values

(1)Location **(2)Log_feature** **(3)Resource_type** **(4)Event_type** **(5)Severity_type**

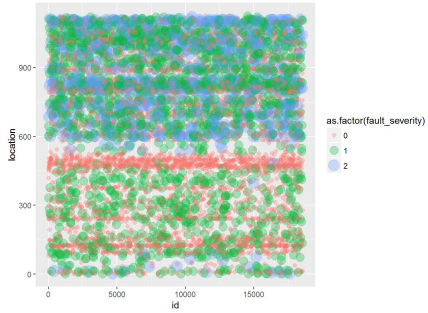


(1) & (2) Some locations and features require more maintenance calls than others
(3) There are 10 distinct types with 2 requiring more maintenance calls
(5) There are 5 severity types with 1 & 2 of frequent types

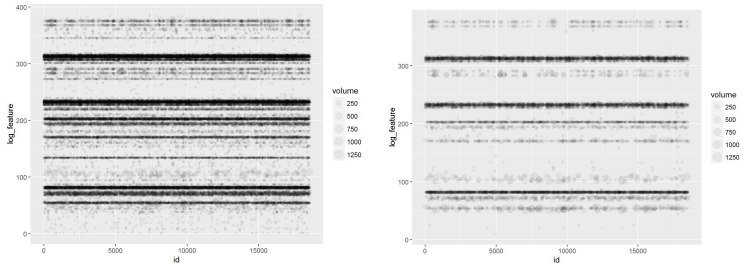
Scatter plot : id(time) scale

Scatter plot of each dataset compared to id (time) to explore time related patterns

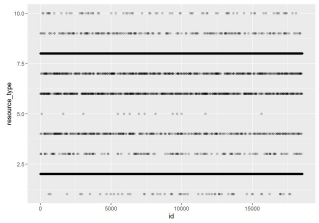
(1) Location



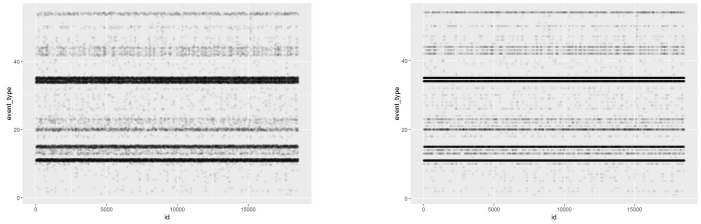
(2) Log_feature



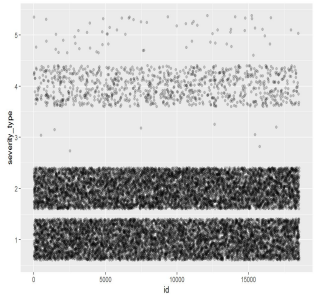
(3) Resource_type



(4) Event_type



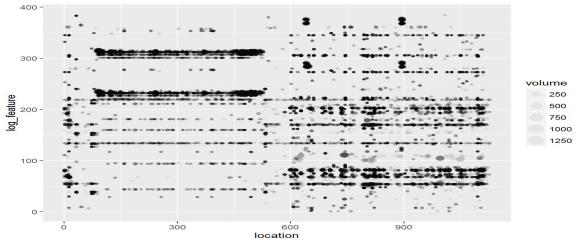
(5) Severity_type



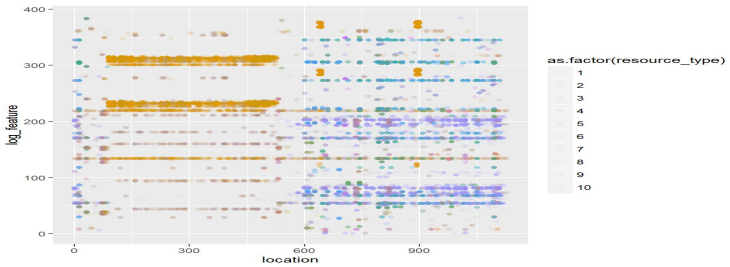
(1) & (2) & (3) & (4) Reported faults are continuous at some locations and resource, it suggests a relationship of certain resource type that requires frequent maintenance stationed at a particular location.
(5) The distinct categories have clear proportion clear distributed in the logs. Severity_type no.1 has the most severity impact on users. Severity_type no.3 only gives a low level of severity on rare occasions.

Scatter plot : location scale

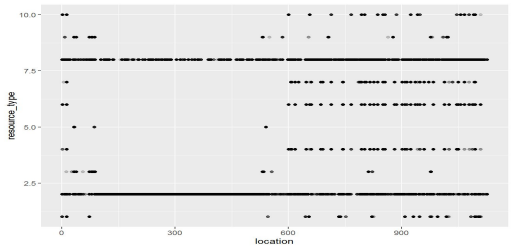
(1) Log_feature



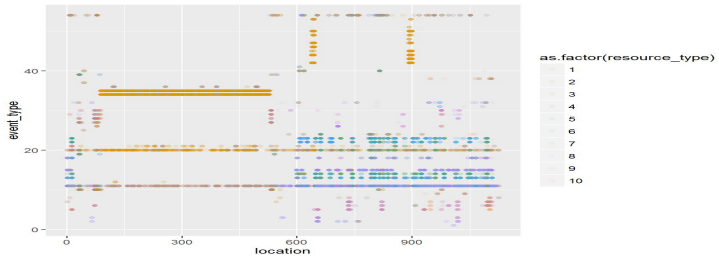
(1a) col = resource



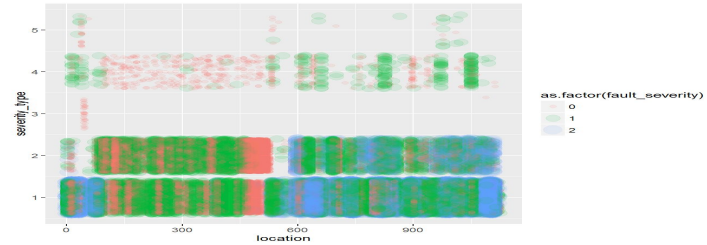
(3) Resource_type



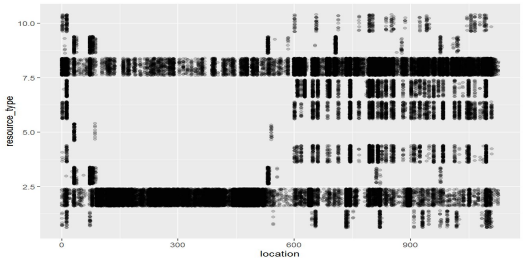
(4) Event_type



(5) Severity_type



(6) Resource: Jitter



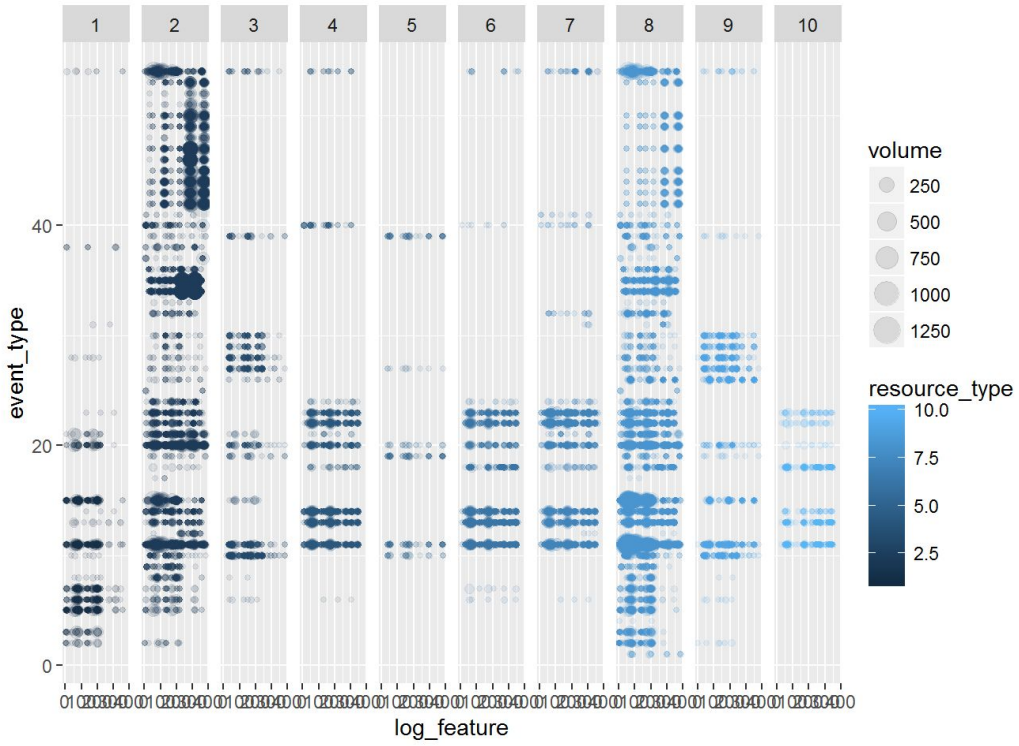
Log features appear distinctly at grouped location numbers, with the mapping of resource and event data following shape. A clustering of resource types at certain locations is shown.

However from this graphs it is unclear whether all locations have all resource types stationed. The other assumption take that all resource types are present but by location some equipment may be more prone to fault then others. The location's position in a high traffic hub can be one reason or that a particular resource is due for replacement.

At (5) & (6), the length of these variables able suggest a particular resource type can correspond to a certain severity type.

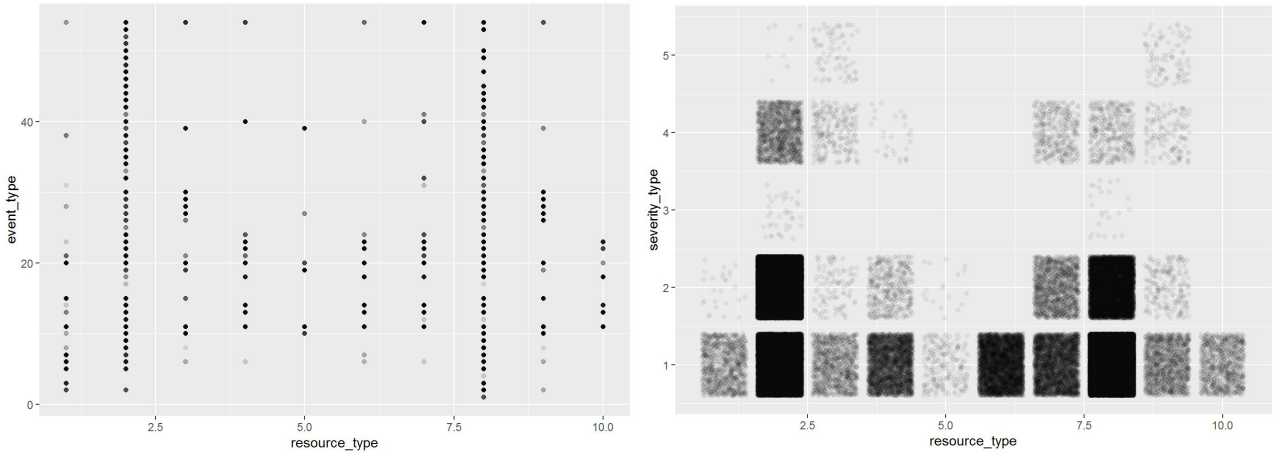
Scatter plot : other significant comparisons

The following plot looks at the relationship between event_type and log_feature with sized by volume and cut-away by resource type:



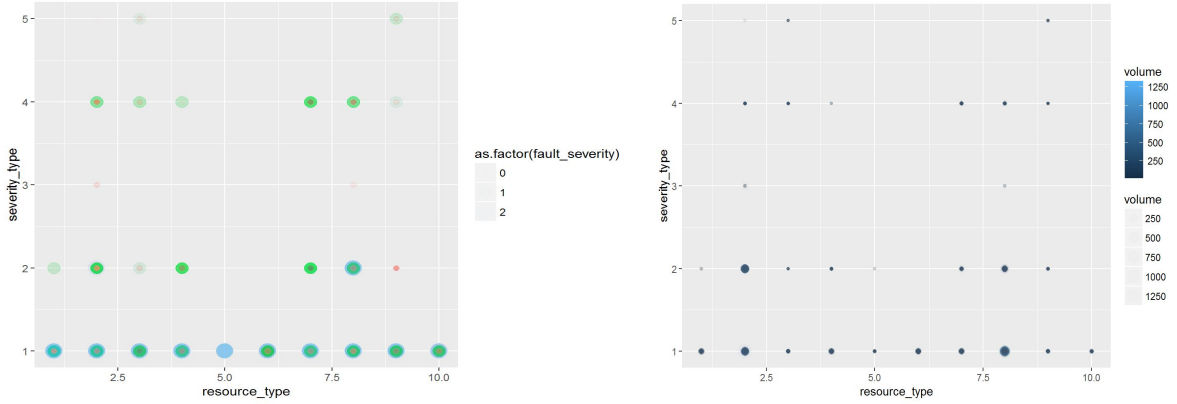
Event_type seem to describe a different between the resource types, while also suggesting each resource type attract particular fault events like its unique behaviour. Resources 2 & 8 have very similar patterns, as with 3 & 9 and 4, 6 & 7 .

The following scatter plots seeks the relationship between event_type, severity_type and resource type to see if there are set behaviour patterns.



Like the previous graph each resource share the same event_type but has set fault events like behaviours in a human. Each resource type also has a distinct pattern on displaying severity type. Again 2 & 8 have similar patterns on the right hand side graph as well as 3 & 9 and 4, 6 & 7. This leads to deduce a relationship is also there between events and severity_types.

Lastly, the further the exploration, the following plots then shows faulty_severity magnitudes are also interrelated with the resource's feature volume. The left graph shows the magnitude by different colour and shape, while the left graph 's points are sized and coloured by its volume.



Both graphs shows a very similar makeup.

Initial Findings

A visual body of data suggests data that is hierarchical and interdependent by the resource that is reporting fault. You can imagine it is a particular machine that does a particular job hence has different faulting events. The system emits a particular severity warning and this warning is highly correlated to faulty severity that can mean high, medium or no disruption impact to users.

The faults can also vary by location and this can be explained by the location's geographical features. Whether it is near a business hub of high signal volume or situated at adverse climate places.

Some resource types require regular attention yet the has little impact on the network. On all high impact faults are found in types 1, 2 severity_type. By location each resource_type has particular features that result in events. From these events and magnitude of volume a severity_type can be deduced and compared to a fault_severity rating.

On predicting fault_severity, which is the fault's impact on its users, the report will propose the following conceptual model:

Fault_severity ~ Location | resource_type | log_feature*volume | severity_type

Where fault_severity = High user impact rating (2) occurs at

1. Location > resource_type >log_feature / volume / event > severity_type (1 or 2)
   or
2. Resource_type = 5

Random Forest modeling

Approach to the problem

As the data is categorical and hierarchical, randomForest method is chosen as the modelling method. This method is chosen over linear, logistic and clustering methods because of the data's non-linear nature, complexity and working time. The degree of complexity of events will benefit by walking through a decision tree for prediction.

Feature Engineering

Here, feature engineering's aim is to reduce input combinations to create simplex, flexible variables.
Each resource type exhibits certain behaviours that then conclude to be deduced as one severity_type. It will be good to quickly eliminate repeated or unused combinations to simplify and quicken the modelling as well as facilitate randomForest features.

This following variables will be engineered for use. Variables that do not need location or fault_severity information will use the incident_log which has more data to describe the case.

1-3 Exploring combination size and creating index

1 - location & resource combinations (network tbl)

There are 1439 combinations in the network. An attempt to reduce combinations to combination that are real in the data, since some resource are not on registered.

2 - log_feature & event_type = service call (incident_log tbl)

There are 1690 combinations in the network. An attempt to reduce combinations to combination that are real in the data, since some features will not have certain events.

3 - service_call & resource (incident_log tbl)

There are 4505 combinations in the network. An attempt to reduce combinations to combination that are real in the data, since some resource may not perform certain features.

4-7 feature generation

4 - Extract count of log_features at each location

This variable is done by extracting the sum of distinct features. The value can improve volume scores that the location outputs and looks after.

5 - Extract count of events present at each location

This variable is done by extracting the sum of distinct events. The value can improve resource_type scores as resource_types leads towards severity ratings.

6 - Extract count of resource at each location

This variable is done by extracting the sum of distinct events. The value can improve resource_type scores as resource_types leads towards severity ratings.

7 - Get probability of fault_severity by resource (proportion)

This variable is done by composing a contingency table. The value can give a weighting on the prediction.

```
> t_res.sev
Source: local data frame [29 x 4]
Groups: severity_type [5]

   severity_type resource_type count_res.sev          rf
          <fctr>        <fctr>         <int>       <dbl>
1             1             1          1056 0.007211982
2             1             2         28944 0.197673863
3             1             3          1133 0.007737855
4             1             4          3308 0.022592079
```

```
> head(merge_res)
  sev.res  sev_res.rf
1     1-1 0.007211982
2     1-2 0.197673863
3     1-3 0.007737855
4     1-4 0.022592079
```

Modelling

There were 7 variables to select. The training set ratio picked is 0.7. A high depth ntree is chosen to cater for more outcomes.

variable	description	reason
resource_type	unit measure of faulting feature	High correlation in scatterplot
severity_type	5 unordered types - warning message received from monitoring machines	High correlation in scatterplot
volume	unit measure of faulting feature	High correlation in scatterplot
t_feature	Tally of unique service that location registers fault on	Influences portion of logs against the location
t_resource	Tally of unique resources that location utilises	Influences portion of logs against the location
t_event	Tally of unique events occurring at that location	High relation to feature and severity_type
sev_res.rf	Probability of severity_type rating with a particular	Gives a weighing

Results and discussion

	Fault_severity ~	**Overall Accuracy**
B	resource_type + severity_type	0.5987278
B + e	resource_type + severity_type + event_type	0.5987278
B + v	resource_type + severity_type + volume	0.603202
1a	resource_type + severity_type + t_feature	0.6332812
1b	resource_type + severity_type + t_resource	0.6152768
1c	resource_type + severity_type + t_event	0.6116651

1d	resource_type + severity_type + sev_res.rf	0.6053582
2a	t_feature + sev_res.rf + t_resource	0.6584551
2b	**t_feature + sev_res.rf + volume**	**0.6665409***
2c	t_feature + t_resource + volume	0.6578082
2d	t_feature + sev_res.rf + volume	0.6665409
3a	**t_feature + sev_res.rf + volume + t_event**	**0.7405531****
3b	t_feature + sev_res.rf + volume + t_resource	0.7259447
4a	t_feature + sev_res.rf + t_event+ t_resource	0.7155409
4b	t_feature + volume + t_event+ t_resource	0.7268072

To predict fault_severity with 74% accuracy, by location, the information on :

- Number of feature services the location performs - How busy
- The knowledge of the resource_type's severity message type?
- The service volume output - How busy
- Number of frequently logged events - fault history range

Hence in summary, the busier and more output the location contributes the more prone to location will cause network disruption.

Some faults are continuous but have low disruption impact. While severe disruptions are few and at specific locations.

The equipment's fault history is also telling as this translates to equipment's end of life. Also the accuracy of the severity_type message is important to keep hold. It is likely possible that severity_type messages become wrong hence lead to unnecessary maintenance calls.

Also, those maintenance calls that have no impact to service maybe from excellent alternate signal routes that can alleviate extra traffic until the fault is fixed. Meanwhile, it can also mean the location is not of high traffic to begin with.

Recommendation

For maintenance management and staff working in the system, fault incidences cannot be avoided as equipment have their life and require repair and renewal.

From the study's findings it is recommended that,

1. Monitoring systems have a special or unique alert from high traffic volume service locations. In specific not only does volume count, but the number feature tasks that the location performs. The segregation of this signal will help managers sort important to urgent tasks. As currently it is clouded by type 1 or 2 band.
2. The upkeep of innovative monitoring systems to ensure the communication of the severity_type is more accurate so that the scheduling of important to urgent jobs is effective and economical in case of an error.
3. Upgrading the network with quality materials and workmanship is also important as it prolongs the equipment's working life. If the assumption is that all service locations houses all resource types, those previously predicted may be signalling aging equipment

Future study directions

To further the case study, better feature engineering, using a hybrid of methods would be able to keep the variable count down while gaining better prediction accuracy. While the use of mosaic graphs in exploration may be able to find better ways to engineer useful variables.

- Feature engineering that incorporates more probability theories to maximise use of data across the available variables.
- The use of mosaic graphs in exploration phase can also help visualise the ratio and proportions of variable relationships to construct better features.
- Incorporate the use of gradient boosting methods to improve predictions.
- Consider using a hybrid method such as clustering and randomForest