Report on my own edx project

FUNG CHE HEI

8/19/2020

1. Introduction

1.1 Background

There are many valuable minerals and geological materials in the Earth that are stored in the form of some kinds of ores, lode, vein and reef. Mining is the human activities to extract those of them. Although mining activity seems interesting for human to discover different kinds of minerals in the Earth, it is a dangerous activities because a hazard of seismic bumps would occurs in many underground mines. Inaccurate prediction and detection would cause great damage to human life.

Therefore a good seismic hazard assessment is important and required for mining activities. With the aid of machine learning technologies, some research including clustering [1] and artificial neural networks [2] are used for prediction of seismic tremors in the past years.

1.2 Aim

Our main aim is:

- To forecast whether the high energy seismic bumps (higher than 10^4 J) would occur in coal mine in next shift, in order to predict whether the coal mine is under hazardous state or non-hazardous state.

With predicting the possibility of the occurrence of hazardous situation, appropriate risk assement and supervision service can be made. For example, reducing the risk of rockburst by the use of distressing shooting method and withdrawing workers from the threatened area.

1.3 Data set Information

The data set used is called "seismic-bumps Data Set" which is downloaded from UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/seismic-bumps#

Here we read the downloaded data set and call it 'seismic'.

```
seismic <- as.data.frame(read_csv("data/seismic_bumps.csv"))
```

An overview on the seismic-bumps Data Set

```r
nrow(seismic)
```

```
## [1] 2584
```

```r
ncol(seismic)
```

```
## [1] 20
```

```r
head(seismic)
```

```
##   id seismic seismoacoustic shift genergy gpuls gdenergy gdpuls ghazard nbumps
## 1  1       a              a     N   15180    48      -72    -72       a      0
## 2  2       a              a     N   14720    33      -70    -79       a      1
## 3  3       a              a     N    8050    30      -81    -78       a      0
## 4  4       a              a     N   28820   171      -23     40       a      1
## 5  5       a              a     N   12640    57      -63    -52       a      0
## 6  6       a              a     W   63760   195      -73    -65       a      0
##   nbumps2 nbumps3 nbumps4 nbumps5 nbumps6 nbumps7 nbumps89 energy maxenergy
## 1       0       0       0       0       0       0        0      0         0
## 2       0       1       0       0       0       0        0   2000      2000
## 3       0       0       0       0       0       0        0      0         0
## 4       0       1       0       0       0       0        0   3000      3000
## 5       0       0       0       0       0       0        0      0         0
## 6       0       0       0       0       0       0        0      0         0
##   class
## 1     0
## 2     0
## 3     0
## 4     0
## 5     0
## 6     0
```

After having a quick look on the data set, there are 2584 rows (observations) and 20 columns (attributes). Each observation contains a summary statement about seismic activity in the rock mass within one shift (8 hours) which will be described in section 1.4, to predict 'hazardous' (positive class with value = 1) and 'non-hazardous' (negative class with value = 0) states. If 'hazardous' is predicted, it is possibly that seismic bump with an energy higher than 10^4 J would occur in the next shift.

Here note the there is unbalanced distribution of positive and negative class. Among 2584 observations, only 170 of them are positive class.

```r
sum(seismic$class == 1)
```

```
## [1] 170
```

1.4 Arributes Information

- 1. seismic: result of shift seismic hazard assessment in the mine working obtained by the seismic method (a - lack of hazard, b - low hazard, c - high hazard, d - danger state);

- 2. seismoacoustic: result of shift seismic hazard assessment in the mine working obtained by the seismoacoustic method;
- 3. shift: information about type of a shift (W - coal-getting, N -preparation shift);
- 4. genergy: seismic energy recorded within previous shift by the most active geophone (GMax) out of geophones monitoring the longwall;
- 5. gpuls: a number of pulses recorded within previous shift by GMax;
- 6. gdenergy: a deviation of energy recorded within previous shift by GMax from average energy recorded during eight previous shifts;
- 7. gdpuls: a deviation of a number of pulses recorded within previous shift by GMax from average number of pulses recorded during eight previous shifts;
- 8. ghazard: result of shift seismic hazard assessment in the mine working obtained by the seismoacoustic method based on registration coming from GMax only;
- 9. nbumps: the number of seismic bumps recorded within previous shift;
- 10. nbumps2: the number of seismic bumps (in energy range [10ˆ2, 10ˆ3)) registered within previous shift;
- 11. nbumps3: the number of seismic bumps (in energy range [10ˆ3, 10ˆ4)) registered within previous shift;
- 12. nbumps4: the number of seismic bumps (in energy range [10ˆ4, 10ˆ5)) registered within previous shift;
- 13. nbumps5: the number of seismic bumps (in energy range [10ˆ5, 10ˆ6)) registered within the last shift;
- 14. nbumps6: the number of seismic bumps (in energy range [10ˆ6, 10ˆ7)) registered within previous shift;
- 15. nbumps7: the number of seismic bumps (in energy range [10ˆ7, 10ˆ8)) registered within previous shift;
- 16. nbumps89: the number of seismic bumps (in energy range [10ˆ8, 10ˆ10)) registered within previous shift;
- 17. energy: total energy of seismic bumps registered within previous shift;
- 18. maxenergy: the maximum energy of the seismic bumps registered within previous shift;
- 19. class: the decision attribute - '1' means that high energy seismic bump occurred in the next shift ('hazardous state'), '0' means that no high energy seismic bumps occurred in the next shift ('non-hazardous state').

1.5 Variables Information

There are totally 18 input variables (attributes) and 1 binary output variable (class) in the data set. The below table summarize some information of the variables.

Table 1: Variable Summary Table

variable	Cardinality	Filled	Nulls	Total	Uniqueness
class	2	2584	0	2584	0.0
energy	242	2584	0	2584	0.1
gdenergy	334	2584	0	2584	0.1
gdpuls	292	2584	0	2584	0.1
genergy	2212	2584	0	2584	0.9
ghazard	3	2584	0	2584	0.0
gpuls	1128	2584	0	2584	0.4
id	2584	2584	0	2584	1.0
maxenergy	33	2584	0	2584	0.0
nbumps	10	2584	0	2584	0.0
nbumps2	7	2584	0	2584	0.0
nbumps3	7	2584	0	2584	0.0
nbumps4	4	2584	0	2584	0.0
nbumps5	2	2584	0	2584	0.0
nbumps6	1	2584	0	2584	0.0
nbumps7	1	2584	0	2584	0.0
nbumps89	1	2584	0	2584	0.0
seismic	2	2584	0	2584	0.0
seismoacoustic	3	2584	0	2584	0.0
shift	2	2584	0	2584	0.0

Although 'maxenergy' and 'nbumps' are numeric data representing the magnitude of energy and number of bumps respectively, they have a relatively small cardinality which result in zero Uniqueness (defined by the ratio of Cardinality to Total). Therefore they are classified into categorical variables. For those variables with uniqueness greater than zero are then classified as numeric variables. Below is the table that summarizing the variable type of class and each attributes.

Table 2: Variable Type

Variable	Type
class	binary
energy	numeric
gdenergy	numeric
gdpuls	numeric
genergy	numeric
ghazard	catagorical
gpuls	numeric
maxenergy	catagorical
nbumps	catagorical
nbumps2	catagorical
nbumps3	catagorical
nbumps4	catagorical
nbumps5	catagorical
nbumps6	catagorical
nbumps7	catagorical
nbumps89	catagorical
seismic	catagorical
seismoacoustic	catagorical
shift	catagorical

1.6 Key Steps

---

2. Data Analysis

2.1 Data Cleaning

2.1.1 Present of Nulls

Refer to Table 1 in section 1.5, there is no Null value in the data set therefore removing of those null values is not required.

2.1.2 Statistic

Statistic of attributes is presented as follow:

```r
summary(seismic)
```

```
##        id           seismic          seismoacoustic        shift          
##  Min.   :   1.0   Length:2584        Length:2584        Length:2584       
##  1st Qu.: 646.8   Class :character   Class :character   Class :character  
##  Median :1292.5   Mode  :character   Mode  :character   Mode  :character  
##  Mean   :1292.5                                                           
##  3rd Qu.:1938.2                                                           
##  Max.   :2584.0                                                           
##     genergy            gpuls           gdenergy          gdpuls        
##  Min.   :    100   Min.   :   2.0   Min.   : -96.00   Min.   :-96.000  
##  1st Qu.:  11660   1st Qu.: 190.0   1st Qu.: -37.00   1st Qu.:-36.000  
##  Median :  25485   Median : 379.0   Median :  -6.00   Median : -6.000  
##  Mean   :  90242   Mean   : 538.6   Mean   :  12.38   Mean   :  4.509  
##  3rd Qu.:  52832   3rd Qu.: 669.0   3rd Qu.:  38.00   3rd Qu.: 30.250  
##  Max.   :2595650   Max.   :4518.0   Max.   :1245.00   Max.   :838.000  
##    ghazard              nbumps          nbumps2          nbumps3      
##  Length:2584        Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
##  Class :character   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000  
##  Mode  :character   Median :0.0000   Median :0.0000   Median :0.0000  
##                     Mean   :0.8595   Mean   :0.3936   Mean   :0.3928  
##                     3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000  
##                     Max.   :9.0000   Max.   :8.0000   Max.   :7.0000  
##     nbumps4           nbumps5             nbumps6     nbumps7     nbumps89
##  Min.   :0.00000   Min.   :0.000000   Min.   :0   Min.   :0   Min.   :0  
##  1st Qu.:0.00000   1st Qu.:0.000000   1st Qu.:0   1st Qu.:0   1st Qu.:0  
##  Median :0.00000   Median :0.000000   Median :0   Median :0   Median :0  
##  Mean   :0.06772   Mean   :0.004644   Mean   :0   Mean   :0   Mean   :0  
##  3rd Qu.:0.00000   3rd Qu.:0.000000   3rd Qu.:0   3rd Qu.:0   3rd Qu.:0  
##  Max.   :3.00000   Max.   :1.000000   Max.   :0   Max.   :0   Max.   :0  
##      energy         maxenergy           class        
##  Min.   :     0   Min.   :     0   Min.   :0.00000  
##  1st Qu.:     0   1st Qu.:     0   1st Qu.:0.00000  
##  Median :     0   Median :     0   Median :0.00000  
```

```
## Mean   :  4975   Mean   :  4279   Mean   :0.06579
## 3rd Qu.:  2600   3rd Qu.:  2000   3rd Qu.:0.00000
## Max.   :402000   Max.   :400000   Max.   :1.00000
```

Refer to the above summary and looking at attributes 'nbumps6', 'nbumps7' and 'nbumps89', it is observed all the values are zero which means those of them do not provide any information for classifying positive and negative class. Therefore, we remove 'nbumps6', 'nbumps7' and 'nbumps89' from the entire data set. For the attribute 'id', it can be regarded as primary key of the data set and do not use for binary classification.

2.1.3 Correctness

It is obvious that the total number of seismic bumps (nbumps) equals to the sum of seismic bumps with different energy levels (nbumps2 + nbumps3 + … + nbumps7 + nbumps89) and they should have no difference. The below code test this fact to ensure the correctness of the data set.

```
#test the correctness of the data set
seismic %>%
  mutate(total = nbumps2+nbumps3+nbumps4+nbumps5+nbumps6+nbumps7+nbumps89) %>%
  mutate(diff = total - nbumps) %>%
  filter(diff!=0) %>%
  dplyr::summarize(n=n()) %>%
  pull(n)
```

```
## [1] 2
```

From the above result we can see that two observations suffer from the problem of inconsistency of the number of seismic bumps. Therefore these two observations will be removed from the entire data set and we call the corrected data set 'corrected_seismic'.

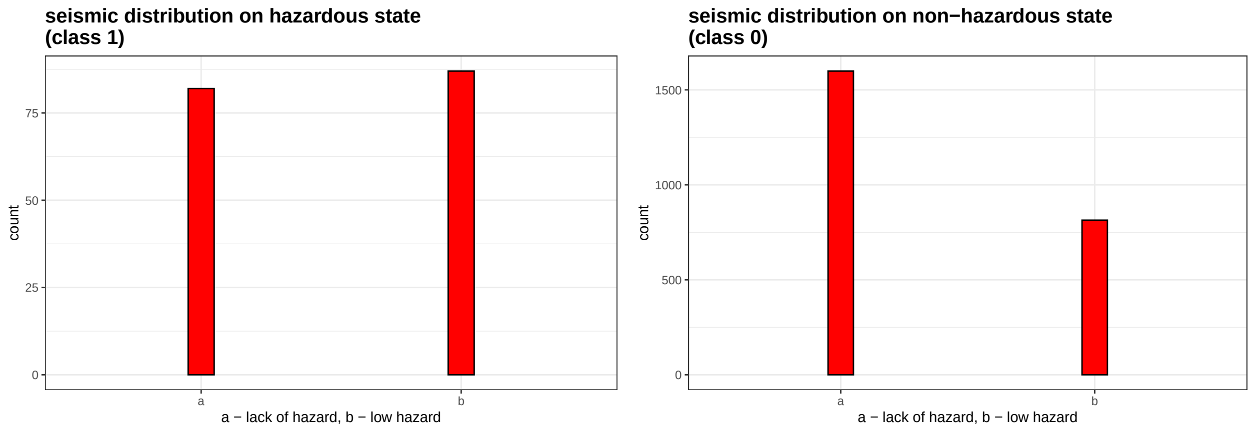
```
#extract the index of incorrect data set
incorrect_index<-
  seismic %>%
  mutate(total = nbumps2+nbumps3+nbumps4+nbumps5+nbumps6+nbumps7+nbumps89) %>%
  mutate(diff = total - nbumps) %>%
  filter(diff!=0) %>%
  pull(id)

#filter out the incorrect observations
#correct the data set
corrected_seismic<-
  seismic %>%
  filter(id!=incorrect_index) %>%
  select(-nbumps6,-nbumps7,-nbumps89,-id)
```

2.2 Data Exploration and Visualization

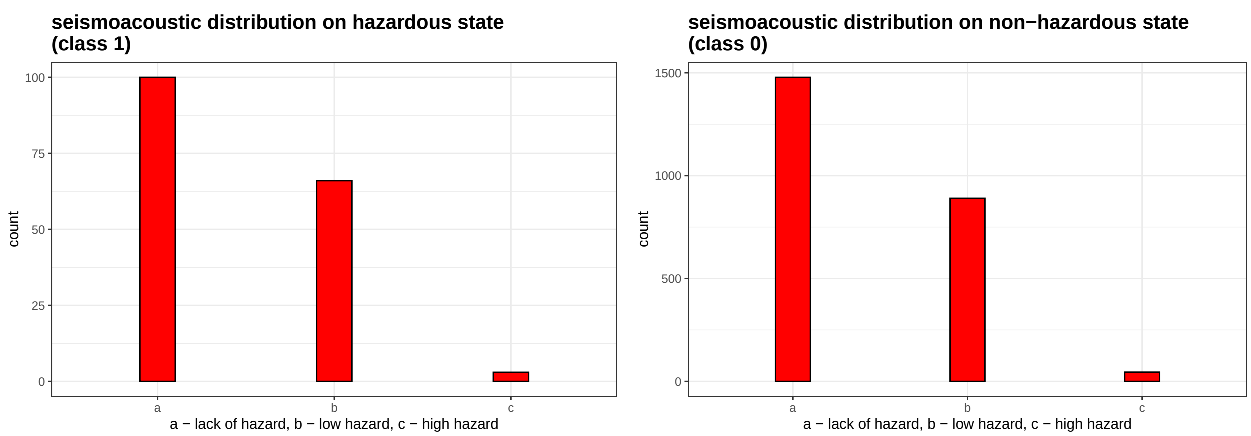
2.2.1 Distribution of seismic (result of shift seismic hazard assessment) on mine with hazardous and non-hazardous state

In this section, the distribution of the result of shift seismic hazard assessment obtained by seismic method on mine with hazardous state and non-hazardous state is visualized. We can observe that the distribution of assessment result a (lack of hazard) and b (low hazard) is almost the same in mine with hazardous state. While the distribution of assessment result a (lack of hazard) is much higher than b (low hazard) in mine with non-hazardous state. The result is quite make sense since the mine with non-hazardous state should be more safe than that with hazard state which is supported by the result of hazard assessment.



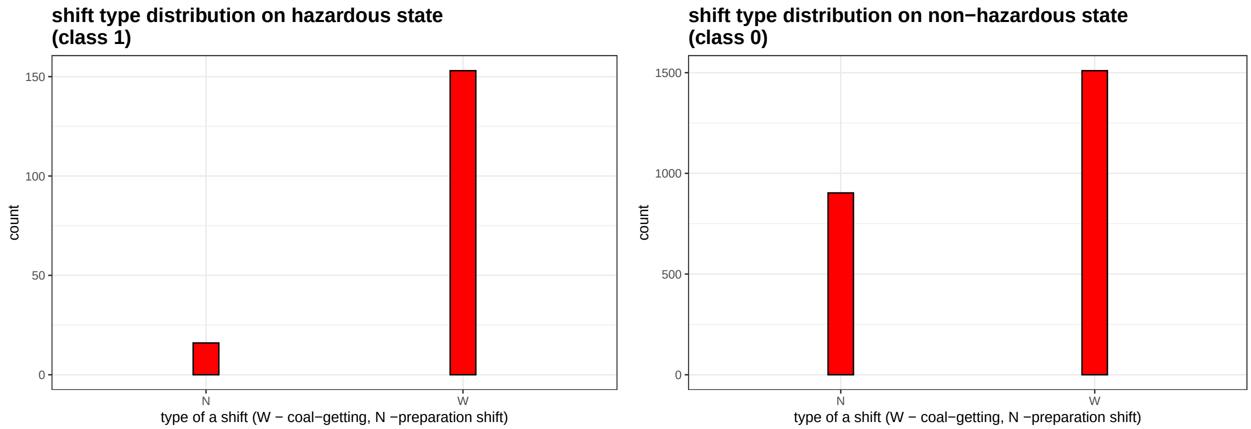
2.2.2 Distribution of seismoacoustic (result of shift seismic hazard assessment) on mine with hazardous and non-hazardous state

Similar to the previous section, the distribution of the result of shift seismic hazard assessment on mine with hazardous state and non-hazardous state is visualized, but the assessment result is obtained by seismoacoustic method. This time we can observe that the distribution of assessment result a (lack of hazard), b (low hazard) and c (high hazard) are almost the same in mine with hazardous state and non-hazardous state. This is pretty much surprise because we can deduce from the result that possibly the seismic hazard assessment result do not affect whether the mine is to be classified as hazardous or non-hazardous. The main reason may cause by the method used for hazard assessment this time is different from the previous one.



2.2.3 Distribution of shift type on mine with hazardous and non-hazardous state

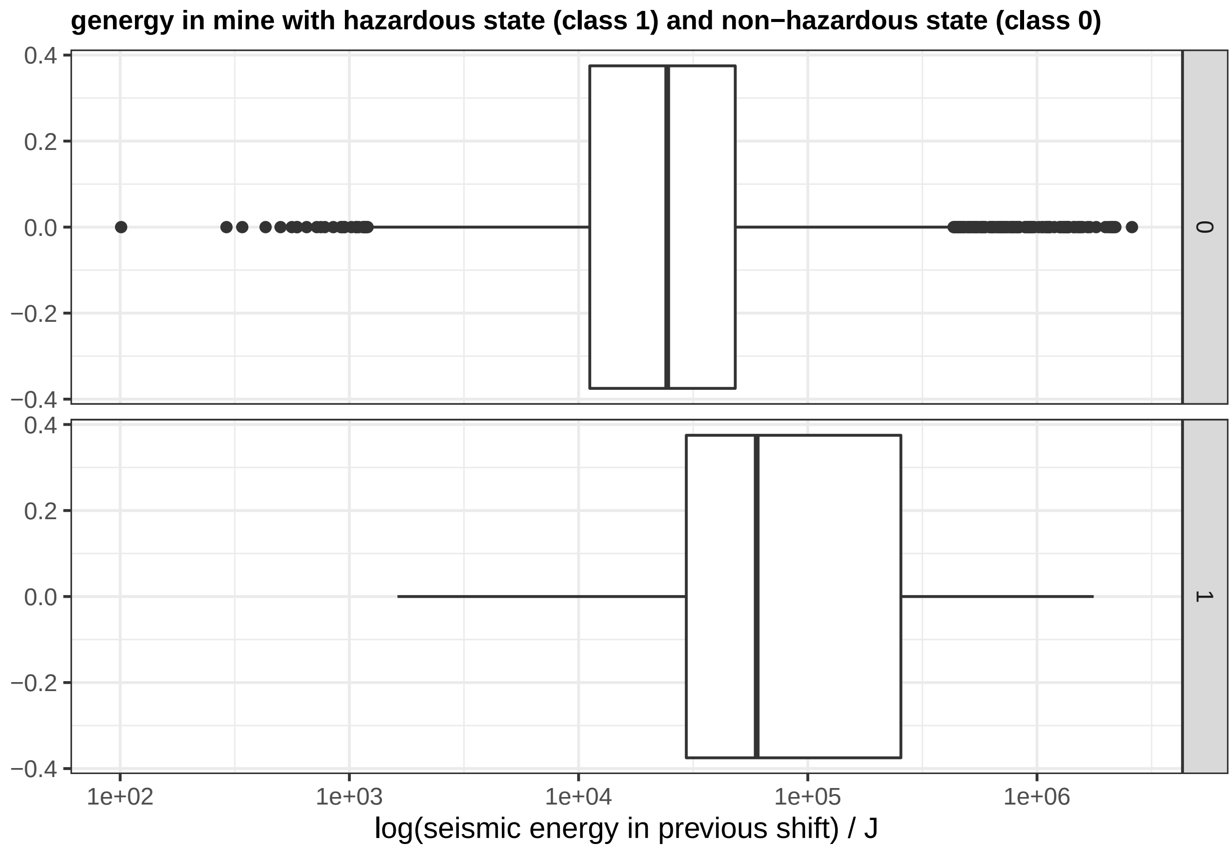
The effects of shift type (coal-getting or preparation) on seismic hazard of mine can be observed in the below graphs. Comparing to mine with non-hazardous state, it clearly shows that the ratio of W to N (i.e. the ratio of time period of coal-getting to preparation in mine) is much higher in mine with hazardous state. This comes to a reasonable observation because there is a possibility that coal-getting activity would trigger a high energy seismic bumps.



2.2.4 Seismic energy recorded in previous shift (genergy) in mine with hazardous and non-hazardous state

This presents the records of seismic energy registered by the most active geophone (GMax) in the previous shift in both classes (i.e. mine with hazardous and non-hazardous state). By observing the below graph, the quartiles of seismic energy are having a higher value in mine with hazardous state than mine with non-hazardous state. It shows that the magnitude of previous seismic energy gives a significant effect on energy of seismic bumps in the next shift. That is, the higher the previous seismic energy is, the higher the chance of high energy seismic bump happens in the next shift.

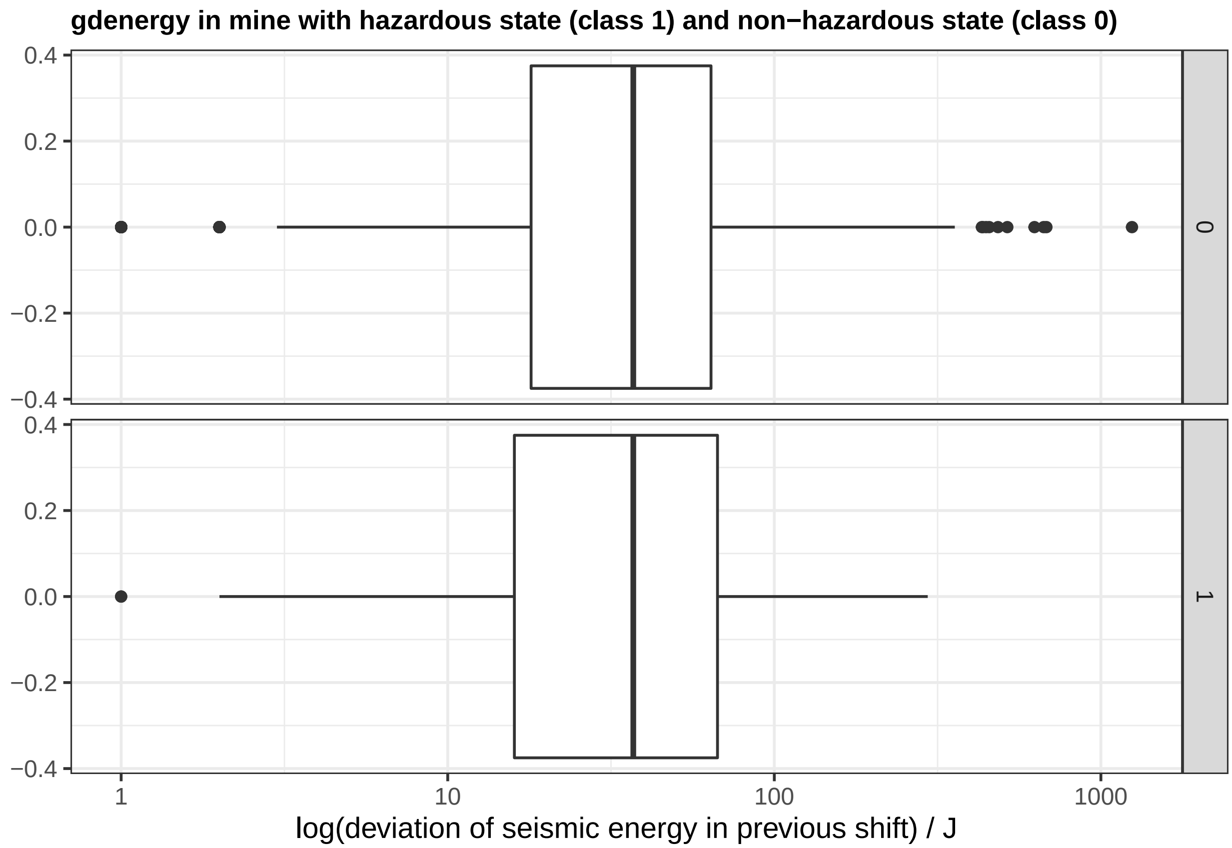
Another thing we observed from the findings is that the range is larger, and the quartile range is smaller in the negative class data set. The reason of this fluctuation may be due to the imbalance number of observations in positive class and negative class.



2.2.5 Deviation of Seismic energy recorded in previous shift (gdenergy) in mine with hazardous and non-hazardous state

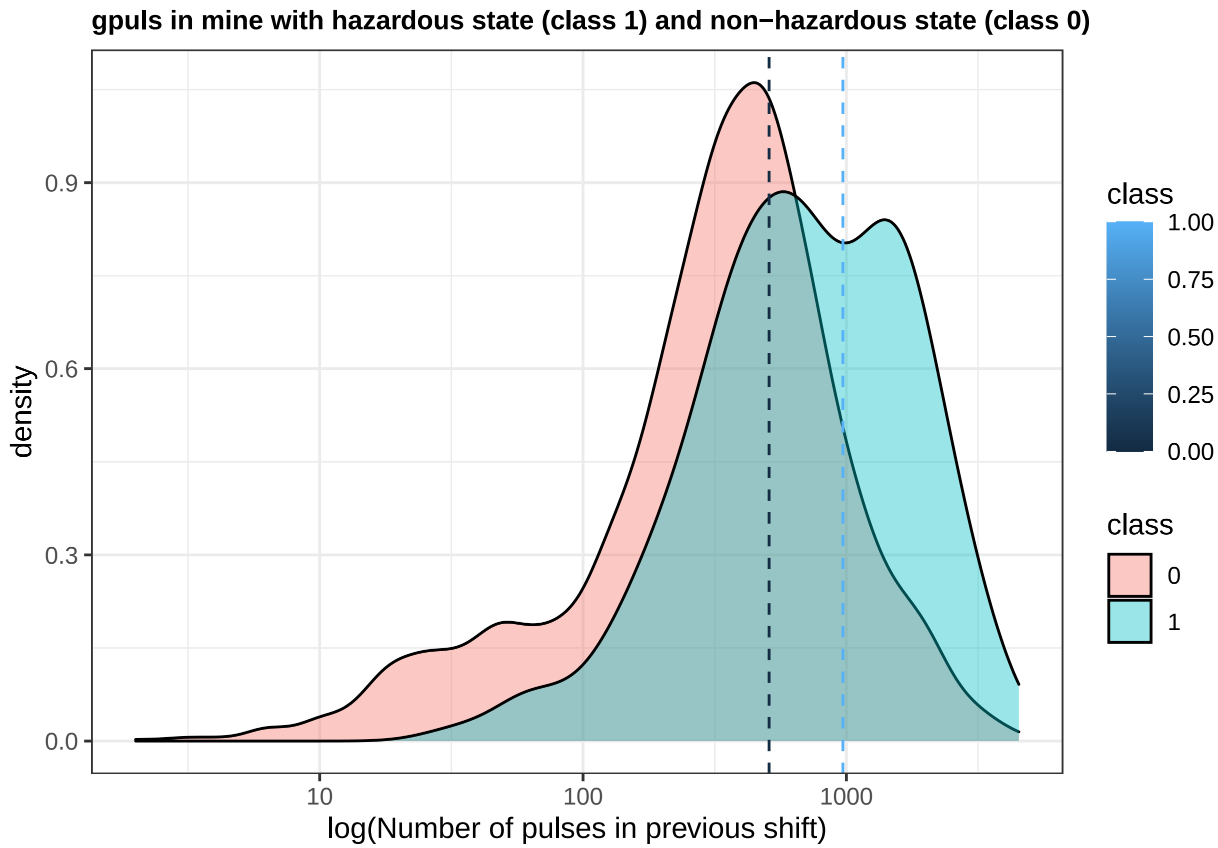
In this section, the deviation of records of seismic energy registered by the most active geophone (GMax) in previous shift in both classes (i.e. mine with hazardous and non-hazardous state) is presented in below graph. It is observed that other than the upper range, the distribution of deviation of seismic energy in the previous shift is almost the same in both classes (i.e. mine with hazardous state and non-hazardous state). Therefore the information given by the deviation of seismic energy in the previous shift may have a less effect on the prediction of occurrence of high energy seismic bumps in the next shift.

The difference in range of data set may cause by the imbalance number of observations in positive class and negative class.



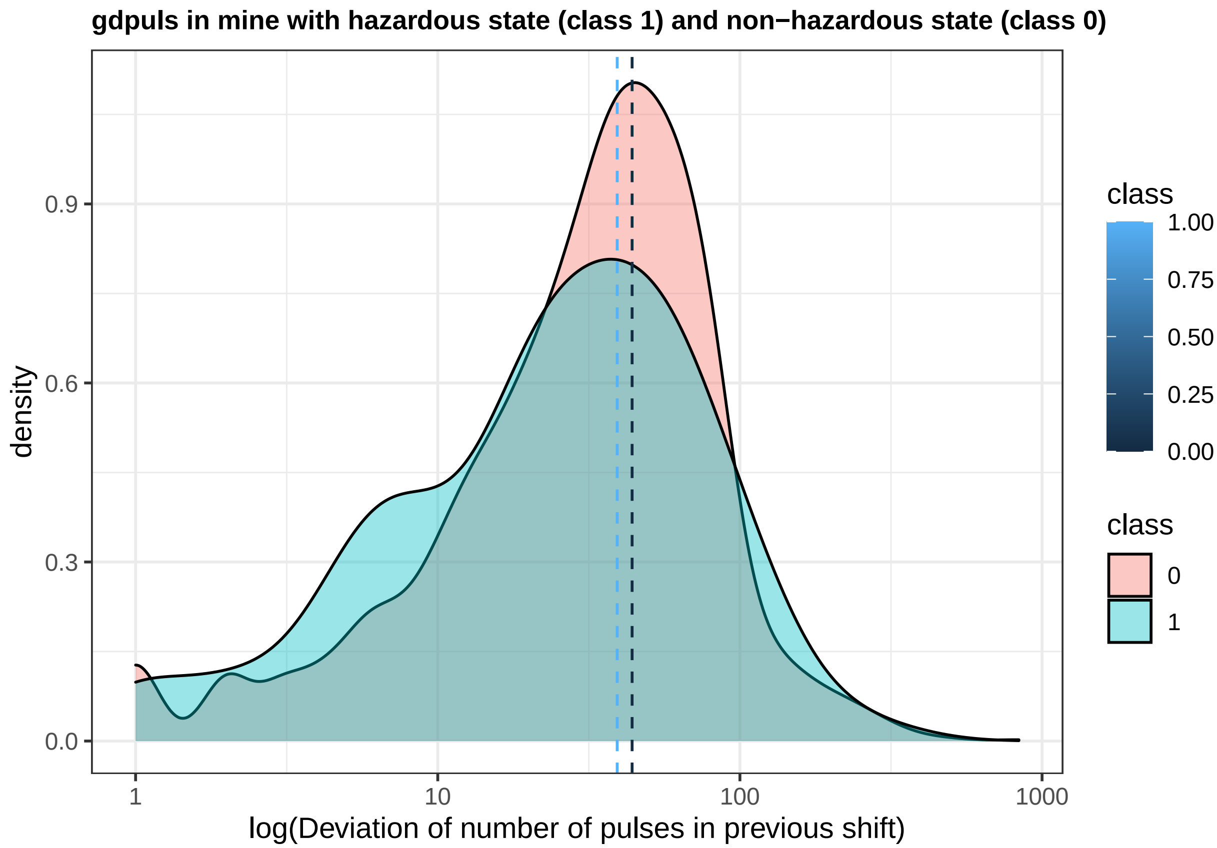
2.2.6 Number of pulses recorded in previous shift (gpuls) in mine with hazardous and non-hazardous state

The below graph shows the distribution of number of seismic pulses recorded in previous shift on the two classes (i.e. mine with hazardous and non-hazardous state). It is observed that the more the seismic pulses recorded in the previous shift, the higher chance the high energy seismic bump occurs in the next shift.From this observation, 'gpuls' seems a good attribute for classifying whether the mine is in hazardous state or non-hazardous state.



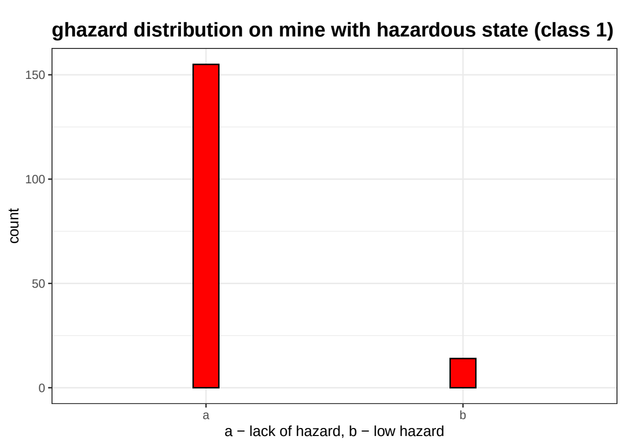
2.2.7 Deviation of number of pulses recorded in previous shift (gdpuls) in mine with hazardous and non-hazardous state

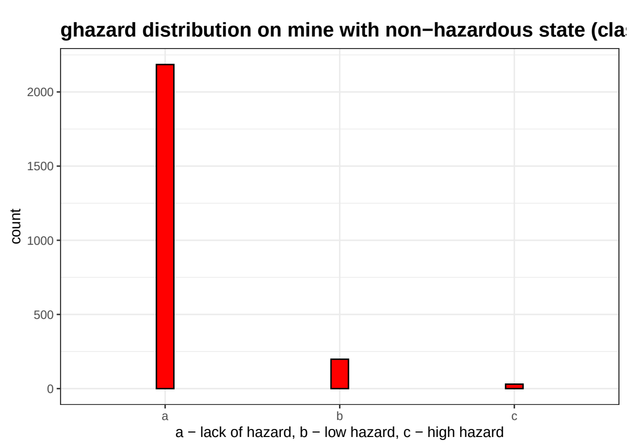
Here we try to observe how deviation of number of seismic pulses recorded in the previous shift would affect the occurrence of high energy bumps in next shift. In the below graph, the distributions of deviation of number of pulses in the previous shift are seem to be similar in two classes. An increase or decrease in number of seismic pulses in the previous shift seems not providing enough information for predicting the occurrence of high energy seismic bump in the next shift.



2.2.8 Distribution of ghazard (result of shift seismic hazard assessment) on mine with hazardous and non-hazardous state

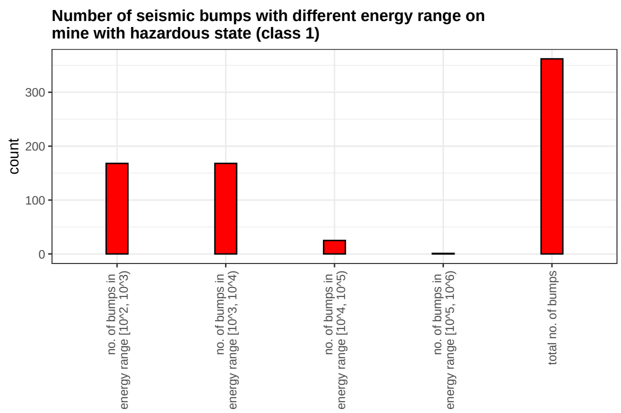
Similar to section 2.2.2, in each of the class (i.e. mine with hazardous and non-hazardous state), distribution of results of hazard assessment is summarized in the below graphs. With keeping the use of seismoacoustic method, only the observations coming from the most active geophone (GMax) are registered. The result is a bit unexpected because in negative class (mine with non-hazardous state), some hazard assessment with result c (high hazard) is recorded while there is no such assessment result recorded in positive class (mine with hazard state). This may due to again the imbalance number of opbservations in positive class and negative class.

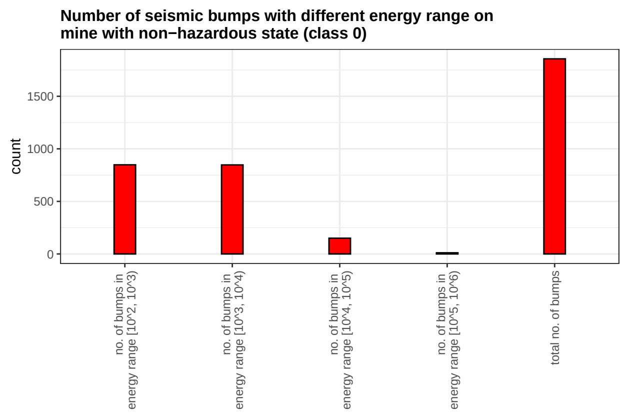




2.2.9 Number of seismic bumps with different energy levels in previous shift on mine with hazardous and non-hazardous state

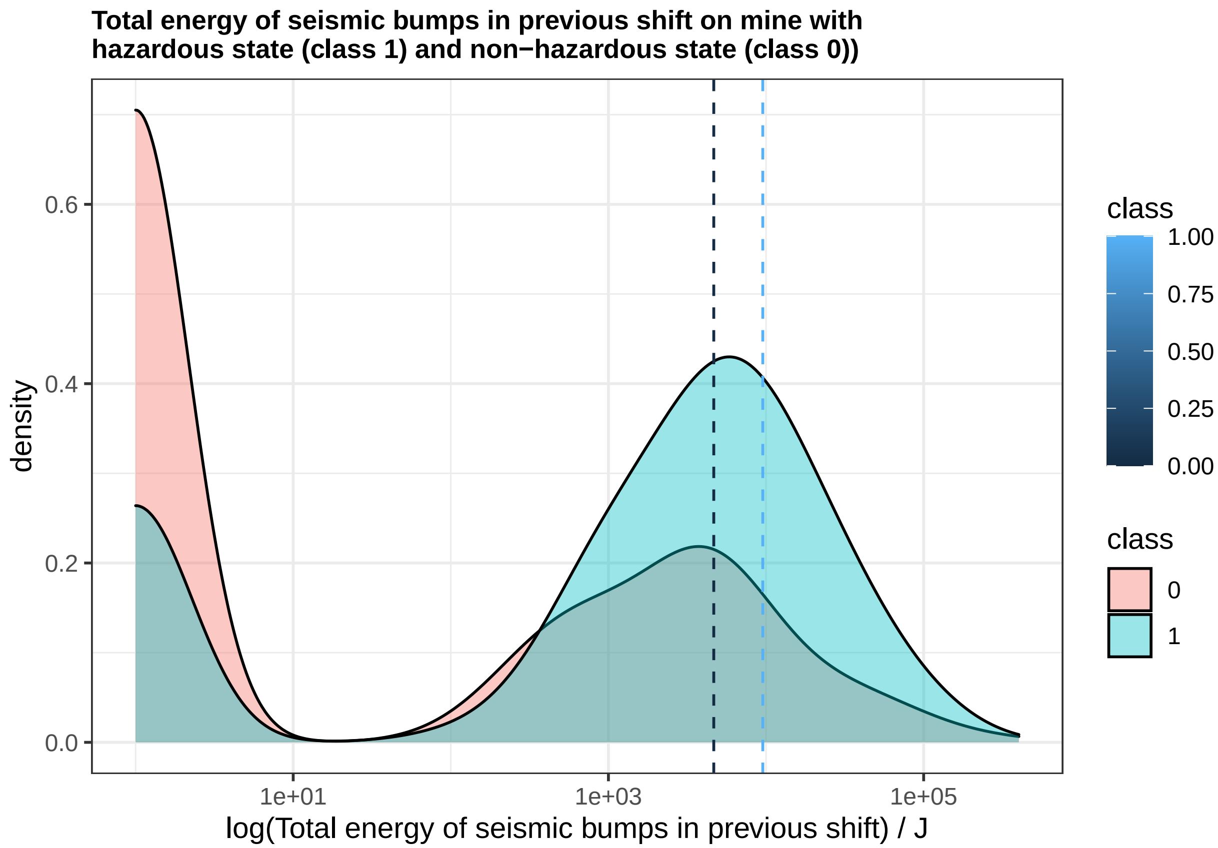
From the below graphs, although the total numbers of seismic bumps recorded in the previous shift are different in two classes, it is interesting to observed that the distributions of seismic bumps in different energy ranges are almost the same. This result gives us a valuable information that the distribution of seismic bumps with different energy ranges in previous shift may has no main effect on causing a high energy seismic bump in the next shift.





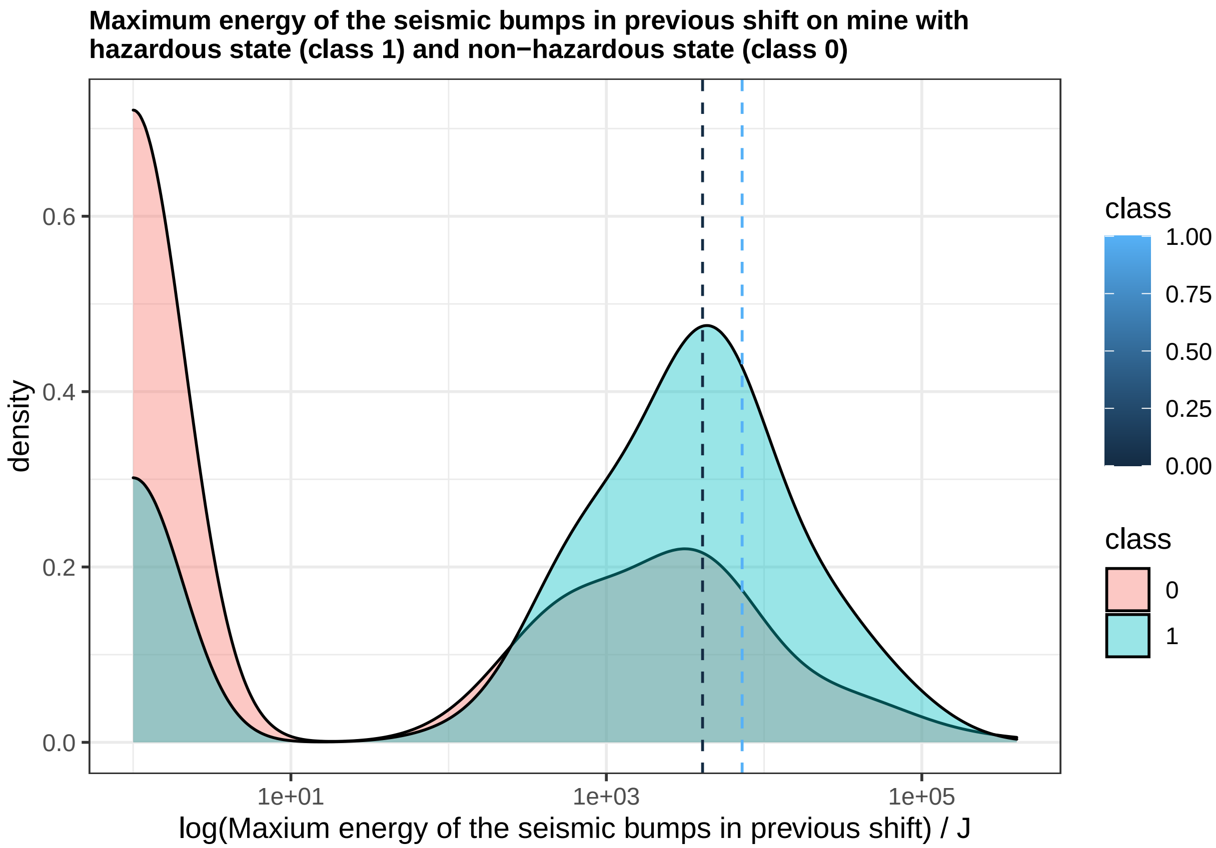
2.2.10 Total energy of seismic bumps in previous shift on mine with hazardous and non-hazardous state

The below graph presents the total energy of seismic bumps registered in previous shift in both classes. Although the mean values of total energy are close to each other, the mean in negative class is only caused and calculated by 30 percent of it's observations. At most time (about 70 percent of the observations), the total energy of seismic bumps recorded in previous shift is zero in non-hazardous mine. While in the mine with hazardous state, about 75 percent of time there is a seismic bump with high energy level (10ˆ3 to 10ˆ5 J) occur in the previous shift. It then comes to a very useful information for predicting the occurrence of high energy seismic bumps in next shift.



2.2.11 Maximum energy of seismic bumps in previous shift on mine with hazardous and non-hazardous state

A very similar result is obtained when we observe the relationship between the Maximum energy of seismic bumps recorded in previous shift and the occurrence of high energy seismic bumps in next shift. It may due to the reason that the Maximum seismic energy recorded in previous shift takes almost the while part of the total seismic energy recorded in previous shift (i.e. maxenergy / total energy approximately equal to 1).



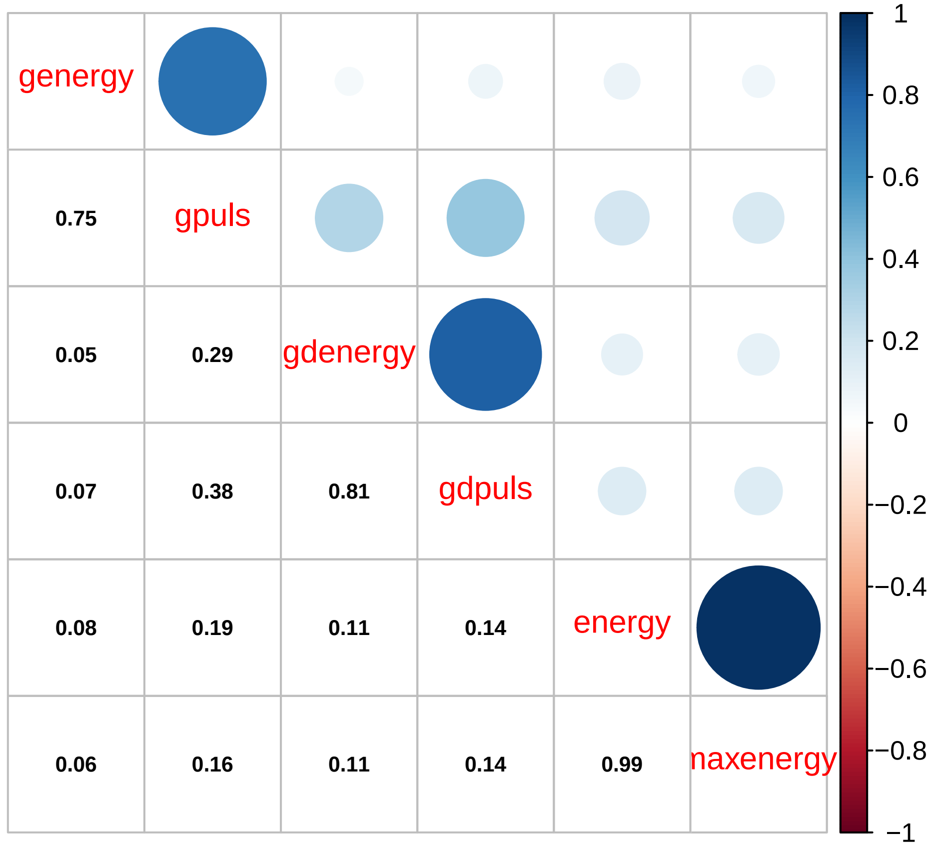
Here we can calculate the ratio of 'maximum energy' to 'total energy' and see how it matches our prediction.

```
corrected_seismic %>%
  mutate(maxenergy=maxenergy+1) %>%
  mutate(energy=energy+1) %>%
  mutate(max_to_total = maxenergy/energy) %>%
  summarize(mean(max_to_total))
```

```
##   mean(max_to_total)
## 1          0.9377593
```

2.2.12 Correlation between numeric variables

Up to now we have visualized how different attributes affect the state of mine (hazardous when it is likely that a high energy seismic bump would occur in the next shift; non-hazardous when it is likely that a high energy seismic bump would not occur in the next shift). However we haven't seen the correlation between attributes. Here a correlation matrix is generated to see how attributes are correlate to each other.



From the graph above we can see that there are 3 pairs of attributes that are closely correlate to each other.

pairs	correlation
energy~maxenergy	0.99
gdenergy~gdpuls	0.81
genergy~gpuls	0.75

2.3 Modeling Approach

2.3.1 Potential problem in data set

As mentioned in section 1.3, imbalance of class variables is a great problem in the data set. Among 2584 observations, only 170 of them belongs to class 1. Therefore if every time we just keep guessing the zero (the negative class), the result would be quite accurate or even perform better than other machine learning methods. Section 2.3.2 and 2.3.3 will demonstrate that problem.

2.3.2 Model - Logistic Regression (with imbalance class)

Here we would try to use logistic regression for this classification problem. First a logistic regression model is trained with the whole data set. Then the model is used to predict the probability of positive class and negative class that an observation belongs to. After that, the predicted probability is compared with the actual probability to decide whether an observation should be classified as positive class or negative class. Finally by comparing the predicted class with actual class, accuracy of the model is calculated. We can obtain an accuracy of 0.735 for logistic regression model.

```r
model_glm <- glm(class ~ ., data = corrected_seismic, family = "binomial")
predict_glm <- predict(model_glm, type = "response")
class_glm <- ifelse(predict_glm >= mean(corrected_seismic$class), 1, 0)
mean(class_glm == corrected_seismic$class)
```

```
## [1] 0.7350891
```

2.3.3 Model - All Zero (with imbalance class)

Second we would try a simple method be just guessing output class as zero every time. The accuracy of 0.935 is obtained with this model.

```r
model_zero <- 0
mean(model_zero == corrected_seismic$class)
```

```
## [1] 0.9345469
```

2.3.4 A solution to imbalance number of class variables

There are many ways to deal with class imbalance problem such as collecting more data, changing the performance metric, resampling data set, generating synthetic samples, using different algorithms and penalizing models [3]. In this report, resampling would be used for creating new examples in the minority class and randomly selecting a number of cases from the majority class.

SMOTE (Synthetic minority over-sampling technique) Algorithm from DMwR package is used for resampling.

```r
re_seismic <- SMOTE(class ~ ., corrected_seismic, perc.over = 200, k = 5, perc.under=150)
```

After resampling we now have 507 positive class and 507 negative class. The data set now becomes balance.

```r
sum(re_seismic$class==1)
```

```
## [1] 507
```

```r
sum(re_seismic$class==0)
```

```
## [1] 507
```

2.3.5 Model - All Zero

After resampling our data set with same number of positive and negative class, an accuracy of 0.5 would be resulted by using the model that always predict zero.

```r
mean(model_zero == re_seismic$class)
```

```
## [1] 0.5
```

2.3.6 Data Partition

In order to evaluate our model performance, it is required to split our data set into training set and test set. A partition of 80 percent of training data to 20 percent of test data would be chosen.

```
y <- re_seismic$class
set.seed(1)
test_index <- createDataPartition(y, times = 1, p = 0.2, list = FALSE)
re_seismic_train <- re_seismic %>% slice(-test_index)
re_seismic_test <- re_seismic %>% slice(test_index)
```

We have now 810 training data and 204 test data. In the following sections, we would try different training methods and see evaluate their performance.

During the training process, Repeated k-fold Cross Validation will be used in order to tune the hyper-parameters to best values. This is the repeated process of splitting the data into k-folds, and the final model accuracy is calculated as the mean from the numbers of repeats which should be more objective. For the following training, 10-fold cross validation with 3 repeats will be used for our data set.

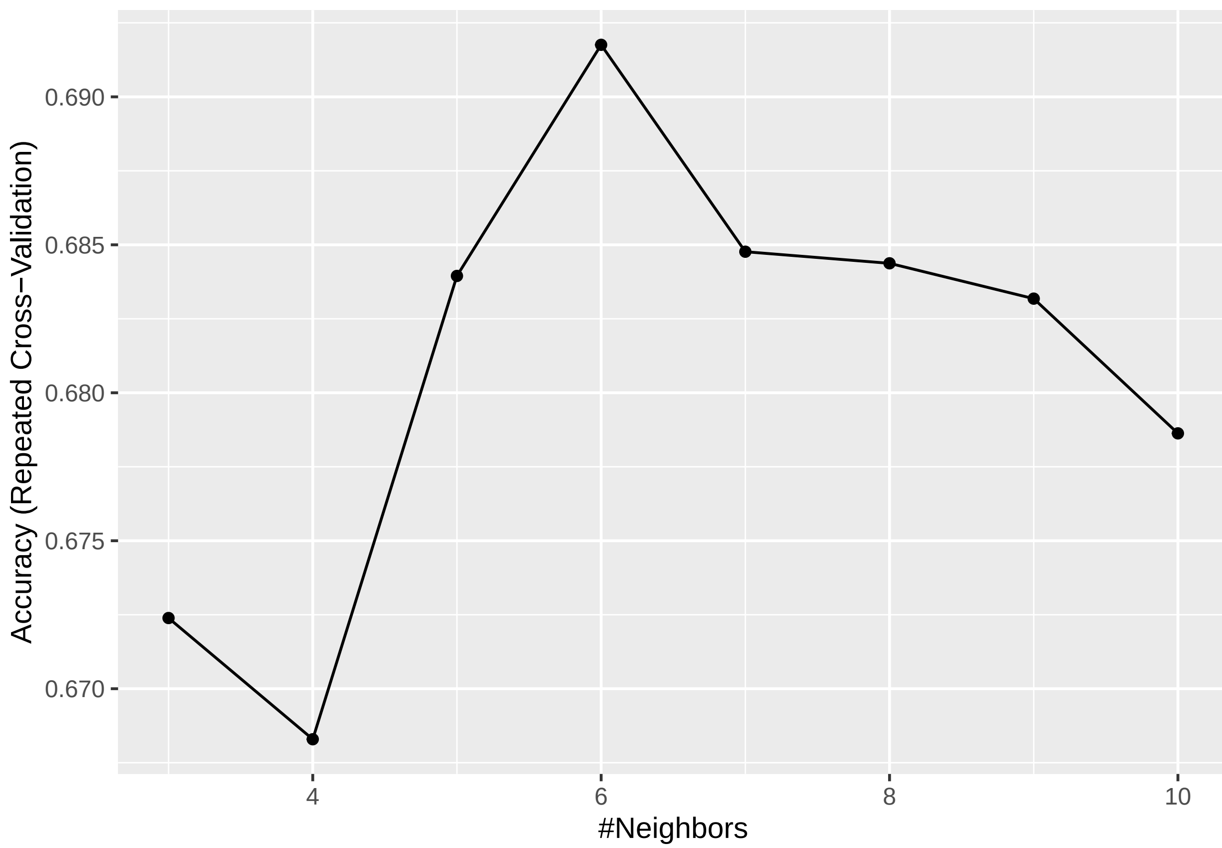
2.3.7 Model - KNN

Here, K nearest neighbors model is used for training. The numbers of neighbors are set to be 3 to 10.

```
set.seed(1)
train_control <- trainControl(method="repeatedcv", number=10, repeats=3)
fit_knn <- train(class~.,
                 data = re_seismic_train,
                 trControl = train_control,
                 method = "knn",
                 tuneGrid = data.frame(k = seq(3, 10, 1)))
```

Below is the graph showing the accuracy in repeated cross-validation test with different numbers of neighbors.

```
ggplot(fit_knn)
```



And the best number of neighbors is 6 for this model.

```
fit_knn$bestTune
```

```
##   k
## 4 6
```

Now in test set, we use the trained model to predict the output and compare with the actual output. The accuracy is then evaluated.

```
predict_knn <-
  re_seismic_test %>%
  mutate(y_hat = predict(fit_knn, newdata = re_seismic_test)) %>%
  pull(y_hat) %>%
  factor(levels = levels(re_seismic_test$class))
cm_knn <- confusionMatrix(predict_knn, re_seismic_test$class)
cm_knn$overall["Accuracy"]
```

```
##  Accuracy
## 0.6911765
```

The accuracy of knn model is 0.6911765.

2.3.8 Model - Logistic Regression

This time we use logistic regression for training our binary classification model.

```
set.seed(1)
train_control <- trainControl(method="repeatedcv", number=10, repeats=3)
fit_glm_2 <- train(class ~ .,
                 method = "glm",
                 data = re_seismic_train,
                 trControl = train_control,
                 family = "binomial")
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

The trained model is then used to predict the output in test set. After comparing the actual result in test set, accuracy is calculated.

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
## prediction from a rank-deficient fit may be misleading
```

```
##  Accuracy
## 0.7107843
```

The accuracy of logistic regression model is 0.7107843.

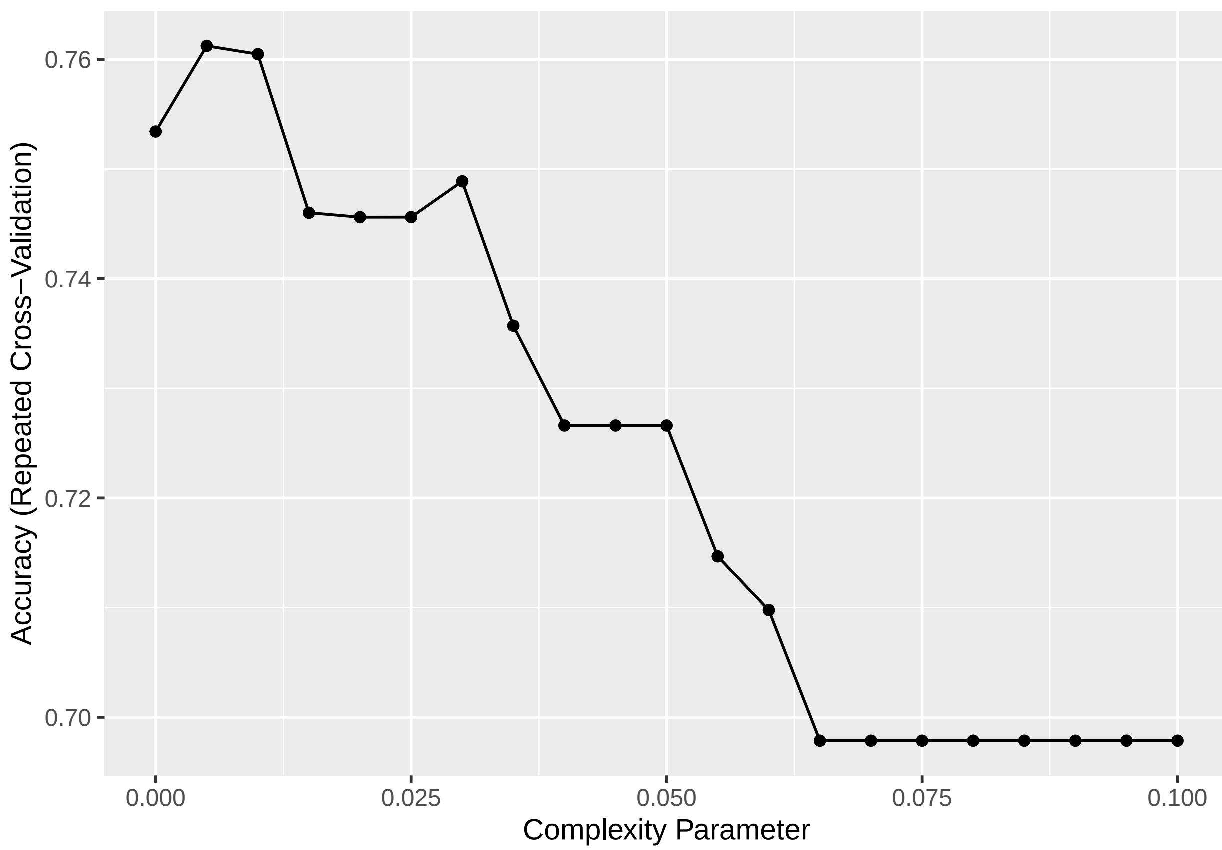
2.3.9 Model - Decision Tree

A well known machine learning method Decision tree is used in this section for training. A complexity parameter 'cp' from range 0 to 0.1 is used for fine tune in repeated cross-validation test.

```
set.seed(1)
train_control <- trainControl(method="repeatedcv", number=10, repeats=3)
fit_rpart <- train(class~.,
                   data = re_seismic_train,
                   trControl = train_control,
                   tuneGrid = data.frame(cp = seq(0, 0.1, 0.005)),
                   method = "rpart")
```

The below graph shows the relationship between complexity parameter and accuracy in repeated cross-validation test.

```
ggplot(fit_rpart)
```



The best value for complexity parameter is 0.005.

```
fit_rpart$bestTune
```

```
##      cp
## 2 0.005
```

The trained model is then used to predict the output in test set. Accuracy is then calculated by comparing the result with actual output.

```
predict_rpart <-
  re_seismic_test %>%
  mutate(y_hat = predict(fit_rpart, newdata = re_seismic_test)) %>%
  pull(y_hat) %>%
  factor(levels = levels(re_seismic_test$class))
cm_rpart <- confusionMatrix(predict_rpart, re_seismic_test$class)
cm_rpart$overall["Accuracy"]
```

```
##  Accuracy
## 0.7303922
```

The accuracy of decision tree model is 0.7303922.

2.3.10 Model - Random Forest

The final machine training method used for training is Random Forest. The mtry parameter, which is the number of variables available for splitting at each tree node, from range 1 ro 7 is used in repeated cross-validation for fine tuning the best model. Also, the number of trees in forest would be set differently (100, 250 and 500) to examine the effect on accuracy in test set.

```
set.seed(1)
train_control <- trainControl(method="repeatedcv", number=10, repeats=3)
cm_rf <-
  sapply( c(100,250,500), function(x) {

    fit_rf <-
      train(class~.,
      data = re_seismic_train,
      method = "rf",
      tuneGrid = data.frame(mtry = seq(1:7)),
      trControl = train_control,
      ntree = x)

    predict_rf <-
      re_seismic_test %>%
      mutate(y_hat = predict(fit_rf, newdata = re_seismic_test)) %>%
      pull(y_hat) %>%
      factor(levels = levels(re_seismic_test$class))

    confusionMatrix(predict_rf, re_seismic_test$class)

  }
  )
```

After our model is trained and tested with test set, the below table summarize the accuracy of random forest model with different number of trees.

```
acc_rf_100 <- cm_rf[,1]$overall["Accuracy"]
acc_rf_250 <- cm_rf[,2]$overall["Accuracy"]
acc_rf_500 <- cm_rf[,3]$overall["Accuracy"]

df<- data.frame("number of trees" = c(100, 250, 500),
           Accuracy = c(acc_rf_100, acc_rf_250, acc_rf_500))

kable(df, caption = "Accuracy of Random Forest Model v.s. number of trees")
```

Table 4: Accuracy of Random Forest Model v.s. number of trees

number.of.trees	Accuracy
100	0.8382353
250	0.8235294
500	0.8284314

---

3. Results and Discussion

knn - not consistant

---

4. Conclusion and Future Work

4.1 Limitation

More and more advanced seismic and seismoacoustic monitoring systems allow a better understanding rock mass processes and definition of seismic hazard prediction methods. Accuracy of so far created methods is however far from perfect. Complexity of seismic processes and big disproportion between the number of low-energy seismic events and the number of high-energy phenomena (e.g. > 10^4J) causes the statistical techniques to be insufficient to predict seismic hazard Therefore, it is essential to search for new opportunities of better hazard prediction, also using machine learning methods. Unbalanced distribution of positive ('hazardous state') and negative ('non-hazardous state') examples is a serious problem in seismic hazard prediction. Currently used methods are still insufficient to achieve good sensitivity and specificity of predictions.

Aknowledgement

Marek Sikora^{1,2} (marek.sikora '@' polsl.pl), Lukasz Wrobel^{1} (lukasz.wrobel '@' polsl.pl) (1) Institute of Computer Science, Silesian University of Technology, 44-100 Gliwice, Poland (2) Institute of Innovative Technologies EMAG, 40-189 Katowice, Poland

---

Reference

- 1. Lesniak A., Isakow Z.: Space-time clustering of seismic events and hazard assessment in the Zabrze-Bielszowice coal mine, Poland. Int. Journal of Rock Mechanics and Mining Sciences, 46(5), 2009, 918-928
- 2. Kabiesz, J.: Effect of the form of data on the quality of mine tremors hazard forecasting using neural networks. Geotechnical and Geological Engineering, 24(5), 2005, 1131-1147
- 3. https://machinelearningmastery.com/tactics-to-combat-imbalanced-classes-in-your-machine-learning-dataset/