Data mining assignment 1 - Basic

Qinghe Gao 2659024, Wenlin Chen 2664925, and Xiaoqing Han 2686611

Group 126

1 Introduction

Data mining is about extracting the inner information and discovering the pattern of data by analysing data, to solve present problems. In data mining, the data is stored and processed by computer, along with different algorithms[9]. In this report, we explore some basic data to introduce the basic steps and information about data mining. **For the part 1**, first we explore a small dataset from class to do the data cleaning and visualization. And we propose some suggestion for the process of collecting data. Then we move on a sentiment analysis for NLP. Using two machine learning algorithms to predict the sentiment of movie review. **For the part 2**, we analyze a big dataset from kaggle competition to predict Titanic survival. We do the data visualization, feature engineering, cross validation, tuning parameters and use random forest and decision tree to get the result. And final accuracy is 0.79425. **For the part 3**, First, the differecne between MSE and MAE and their suitableness under different condition are discussed . Second, the Lung Cancer Detection project of the winner in 2017's Data Science bowl is taken as a excellent DM example for learning. At last, sentence segmentation and topic modeling are useful model techniques to regular text for spam message detection after data transformation.

2 Explore a small dataset

2.1 Exploration

We used python to explore the whole dataset. First, In this dataset, there are 280 instances (i.e. students), and 16 attributes in total such as "what program are you in?". The types of each attribute are object and only 1 number (what is your stress level?). From results, the ranges of category values are variously divided by the instances (such as yes, no, unknown and other descriptions), the range of the only numeric attribute is (-100,100). Then we found there are only two missing values at attribute 'Time you went to be Yesterday '. Besides, we found the columns' names are quite long and difficult to deal with. We extract the feature of columns name and rename with **major, machine, infor, statistics, database, gender, choco, birthday, neighbors, stand, stress, money, random, bed, good1, good2**.

First, we deal with the **major**, which contain the information of program of students. The data of major is quite a mess and we need to normalize the name of program. Thus, we change all entries involved **'Artificial intelligence'** into **AI**. Then we did the same thing for **Computational, CS, BA, Finance, Econometrics, Bioinformatics, Information and others**. Thus, **major** attribute has been cleaned. Next, for **machine, infor, statistics, database** these four attributes basically check whether students have background about machine learning, information retrieval, statistics, and database. And every column has different entries but has same meanings : *Yes, No, Unknown.* Thus, we map *Yes* with *1*, *No* with *0* and *Unknown* with *2*. Thus, all the entries has been normalized. And then for **gender**, it is interesting to find that there are four people choose *unknown*. Furthermore, we are interested in the last two columns. These two columns contains the information about *What makes a good day for you?*. These two columns are quite suitable to the wordcloud. Thus, we split each entry with space and count the frequency for each single words to make a wordcould.

2.2 Data visualization and discussion

As top left of figure 1 shows the distribution of program of students. It is clear to see the AI students take the biggest percentage, which is 25%. The next is Business Analytics. We can see the programs are quite diverse, which indicates data mining has various application in many fields. And for top right of figure 1 it contains how many students have relevant background(percentage). Most students have background about statistics and machine learning. But for information retrieval only 23% students have background. Thus, from this it is possible to increase the difficulty of course since many students have background about machine learning and statistics. Bottom left plot of figure 1 is stress level of two genders. It is interesting to see that the distribution of male is quite wide and even three males' stress level are below 0, which are -1 and -100. Distribution of female is quite narrow and mainly centered around 50. This may indicates nowadays everyone are anxious to some certain tings, but some few males are live in the world without trouble. Bottom right plot of figure 1 is wordcloud of *What makes a good day for you?*. It is clear to see that most peopple are happy with nice food, friends, sun, good weather(especially for Netherlands). And there are some interesting entries such as *full agenda, sex, bitches, Orgasm.*

Based on the previous descriptions, we find this raw data file is not good to do data mining. We think when collecting the data it is better to do multiple choices for each question. In this way the data can be collected in a formal way. And it can save time to process the data. And for question involves time it is better to have a formal way to collect such as *Year-Month-Day.* For further improvement, because there is no data that needs to be predicted later we do not choose any column as the target column and then fill the missing value of attribute *Bed time.* We can set the missing value with median of the entries. In summary, the dataset takes two measures: remove low-quality columns and replace missing values.

2.3 Basic classification and regression

Natural language processing(NLP) currently is a heated topic in machine learning field. And sentiment analysis is an important application.[8] Thus, in this

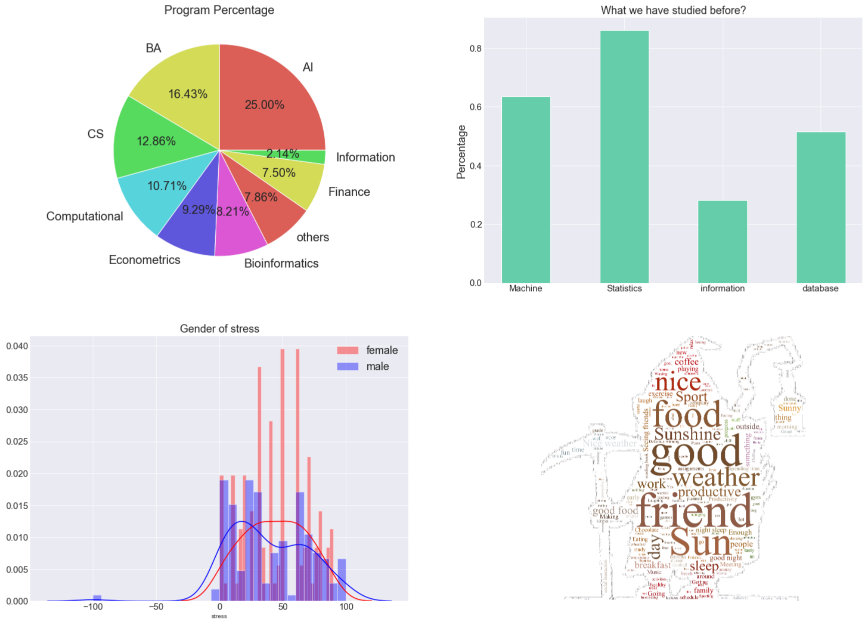


Fig. 1. Plots about the features and resulting models. Top left is plot for distribution of program of students. Top right is plot for relevant background investigation of student. Bottom left plot is stress level for different gender. And bottom right is wordcloud for *What makes a good day for you?*

part small sentiment analysis of 2000 movie reviews is discussed and we used **Logistic regression and linear support vector Machine(SVM)** algorithms to predict the sentiment. 2000 movie reviews were download from [5]. Two columns are included. One is attitude :*0* means *negative*. *1* means *positive*, which are extremely suitable for classification and regression model because there are only two attributes for outcome. Another column is content, which contains review of movie.

step 1: whole data was split into 1600 train dataset and 400 test dataset. **step 2**: content of review can not directly put into the classifiers and need to transform into words frequency vectors. We choose **three parameters for this process: tokenization, removing stop-words,C**. Tokenization is the process that splitting text into every single words. And removing stop-words is deleting useless words. And C is inverse of regularization strength. The *TfidfVectorize* is used to transform text into words vectors which contain the frequency of every words and are also features for prediction. *GridSearchCV* was used to do 10-fold stratified cross-validation and find optimal parameters(from creating words vectors and also classifier itself) for two classifiers respectively. **step 3**: using optimal parameters to predict test dataset and calculate the accuracy with true sentiment for Logistic regression and linear support vector Machine(SVM).

The optimal parameter for Logistic regression are **Without Tokenization, with stop wording,C=10**. And the optimal parameter for linear SVM are **Without Tokenization, with stop wording, C=1**. Form the steps listed before, we know for sentiment analysis task the input are words frequency vectors which are also the main features in the model. And output of classifier for test dataset is prediction of negative or positive labels. The result of two classifiers shows both two classifiers have high performance on prediction. The accuracy are 0.8675 and 0.87 respectively. And F1 score are 0.8685 and 0.8713 respectively. Linear SVM is slightly better than logistic regression. One possible explanation is just a coincidence and we need to perform a significant test to test whether Linear SVM is indeed better than Logistic regression. Another possible explanation is that Linear SVM's loss function only considers support vectors, which can reduce over-fitting than logistic regression.

3 Titanic survival

3.1 Basic information and Data visualization

In this part, titanic competition was discussed and the main idea of this competition is building model based on train dataset to predict survival of test dataset.

First, we explore the train and test dataset. The size of train dataset is $891 * 12$. And test dataset is $418 * 11$. And each attribute of column shows as table 1. It is clear to see that there are lots of missing value in **Age**, **Cabin**, **Embarked** and **Fare**.

Table 1. Information of train and test dataset

Attributes	Meaning	Size
PassengerId(int64)	Number of passenger	Train:891, Test:418
Survived (int64)	0:dead, 1:survived	Train:891, Test:0
Pclass(int64)	Three different classes of ship. 1,2,3 and 1 means best class.	Train:891, Test:418
Name(object)	Name of each passenger, include title.	Train:891, Test:418
Sex(object)	Sex of passenger. Male and female	Train:891, Test:418
Age(float64)	Age of passenger.	Train:714, Test:332
SibSp(int64)	Number of siblings for each passenger	Train:891, Test:418
Parch(int64)	Number of children and parents for each passenger	Train:891, Test:418
Ticket(object)	Ticket number	Train:891, Test:418
Fare(float64)	Price of ticket	Train:891, Test:417
Embarked(object)	Port of Embarkation. C,Q,S.	Train:889, Test:418
Cabin(object)	Cabin of ship	Train:204, Test:91

For visualization, firstly we look at whether there is difference of survival on **sex** attribute. Figure 2 shows different sex has significantly difference on survival. 81% male were died in disaster and 74% male has been saved. And age distribution of two sex shows same trend, which is that around $20 - 25$ yeas old survived and dead rate are both high. Thus, sex is a significantly important attribute and we do need to include in the model. **For class**, we explore whether the class in the ship influences the survival. Figure 3 shows the high class has high survived percentage. While for third class the survived percentage is only 24%. For age distribution we can see most saved and died passengers are around $20 - 30$ years old but for first class died passengers' ages are around 45 years old. Thus, class is also a important attribute for prediction. Furthermore, **siblings**: we explore whether the number of siblings, parents and children of passenger influence the survival. At first, we explore the number of siblings and it turns out large number of siblings has low survival percentage. Additionally, we combine **SibSp** and **Parch** columns into new attribute **FamSize**. And it also shows when family size is relatively small(< 4) the survival percentage is quite high and when family size is bigger than 5 the survival percentage is quite low. Thus, family size can be used into model.

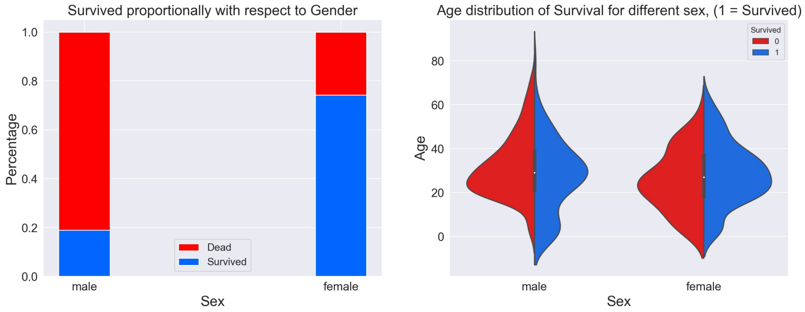


Fig. 2. Survival of different sex. Left picture shows percentage of death or survival for two gender respectively.Red color means dead percentage and blue color means survived percentage. Right picture shows age distribution of two different genders.

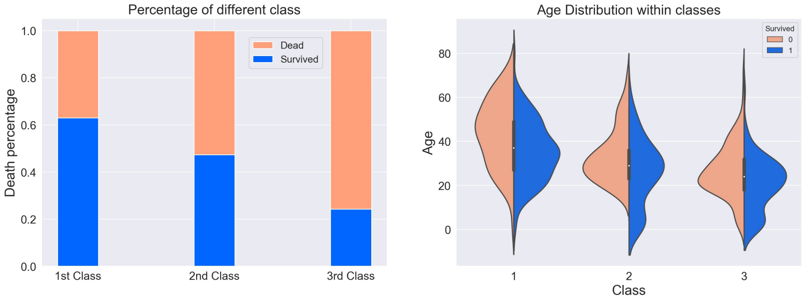


Fig. 3. Survival of different class. Left picture shows percentage of death or survival for three different respectively. Right picture shows age distribution of two different genders.

3.2 Feature engineering, training and result

After data visualization, before fitting model we need to do feature engineering. First we deal with the missing value of **Cabin**. 77.46% value of cabin has missed, which is quite large and it is hard to predict the category by model. Thus, we set all missing value of cabin into **U** category. Thus cabin has nine categories: **A,B,C,D,E,F,G,T,U.**

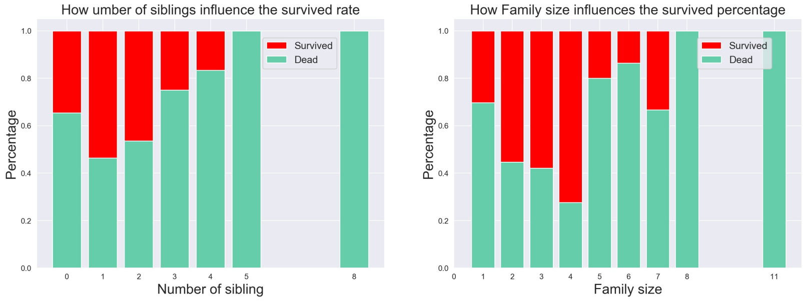


Fig. 4. Survival of different number of siblings and family size. Left picture shows percentage of death or survival for different number of siblings respectively. Right picture shows percentage of death or survival for different number of family size respectively.

Then, we deal with two missing value of **Embarked**. We used mode of Pclass to fill these two missing values.

Then, as explained before we combined attributes **SibSp** and **Parch** into a new attributes **FamSiz**. And figure 4 shows, we combine similar dead percentage into same category. When family size is $1, 5, 6, 7$ we set them as category 1. When family size is $2, 3, 4$ we set them as category 2. When family size is 8 or more we set them as category 0. For **Fare**,there is only one missing value and this missing value is belong to S embarked and 1 class. Thus, we used median number of S embarked and 1 class to fill the missing value.[2]

Furthermore, we found there are some useful information in attribute **Name**. **Name** contain title of each passenger. We merge some titles into same category. **Don, Sir, the Countess, Dona, Lady, Jonkheer** are into **Roy** category. **Mme, Ms** are into **Ms** category. **Mlle, Miss** are into **Miss** category. And **Rev, Dr, Col, Major, Capt** are into **Stuff** category. Then we have six categories **Master, Mr, Ms, Miss, Roy, Stuff**. In this way, information of Name attribute can be used. Besides, for **ticket** attribute, we found the data are quite random. But when counting the number for each ticket number we found a method to organize ticket data. If a ticket number appears more than 8 times, we set it as category 0. And if it appears between 2 and 4 times, we set it as category 2. Otherwise it is in category 1. In this way after transformation ticket attribute decidedly has correlation with survival percentage. Thus, in this way ticket information can be used into model.

Finally, we deal with though missing value of **Age**. Age has so many missing values and it is not suitable to just fill up with median value. Thus, in order to fill the missing values we used random forest to predict the age with attributes **Class, Sex, Title**. We split data into known age dataset and unknown age dataset. And using known age dataset as train dataset to train the random forest model. And then we used unknown age dataset to predict the missing values of age.

Then, we fixed all the missing values and we select **Class, Sex, Age, Fare, Embarked, Cabin, FamSiz, Title, Ticket** as features in the model. And before we actually fit model some features need to be transformed into dummy variables since some features have random order. We used Random Forest and Decision Tree to predict the results. For both classifiers firstly we used *Grid-*

SearchCV to do 10-fold cross validation and to find optimal parameters. And optimal parameter for random forest are **6 max depth, 48 estimator**. For decision tree are **3 max depth**. And using corresponding optimal parameters to predict the result of dataset. Random Forest's accuracy is 0.7942 and decision tree is 0.7799. This outcome matches what we expected. Because decision tree method depends on whole dataset and using all features, which can easily lead to overfit. While random forest randomly chooses the features to build the decision tree and repeat many times to vote for the 'Winner'. This process will significantly reduce the chance of overfitting. [6] Thus the result of random forest is better for our result. And the accuracy from kaggle is 0.7942.

4 Research and theory

4.1 Research the Lung Cancer Prediction

The 2017 year's Data Science Bowl competition [1] was discussed in this paper. The data mining topic is **Can you improve lung cancer detection?**. Specially, its dataset contains the 3-Dimension Compute Tomography data by scanning the suspected patients' lungs. The performance of their model was determined by applying validation datas, whose instances were a fixed set sampled from the Data, to their predicting model. The model with highest accuarcy and meet other standards in the evaluation metric is the winner.

A team called "grt123" won the first place among 1972 teams. Here is their publication - "Evaluate the Malignancy of Pulmonary Nodules Using the 3D Deep Leaky Noisy-or Network". [7]. First, They use **3D convolutional netural network (CNN)** to detect the nodule in the lungs from the CT data; second, They take the top five most suspected nodules to evaluate cancer probabilities with a **leakey noisy-or gate**.

For more details in DM approach, **Step 1**: Select and dispart the precise part of lungs from other organ issues by image processing. Use CNN to detect the nodules from the CT scanning data. Since 3D DM require high GPU, it is not possible to input the whole lungs once. The authors take a size of 128x128x128x1 voxels cube as the input each time and each cube are located with central point to avoid any omission. For these vertex cubes, data extension iis used to fitting process. **Step 2**: Since the most of nodules in this dataset are large in size and usually the shape of malignant tumor varies a lot, includes small size. This autor trained another dataset from A competition called "LUng Nodule Analysis 2016" [4], which includes many mini size nodules. **Step 3**: Not all nodules are malignant tumor, from the result of CNN detection, this team picked the top 5 most suspected nodules to detect the probability of cancer. And 'leakey noisy-or gate' is an advanced method, which has highest accuracy based on common 'MaxP' and 'noise-or' method.

Compared with other teams, the highlight of their work is that the twp steps share the same backbone network, which is a modified U-net. And the over-fitting caused by the lack of training data is alleviated by training the two modules(steps) alternately. And the addition of LUna 2016's dataset increased the prediction accuracy. And the 'leakey noisy-or gate' has lower probability to ascribe existed cancer to benign nodules, when some malignant nodules are missed by detection work.

4.2 MSE versus MAE

Mean squared error, abbreviated as MSE, has the formula:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y_i})^2 \tag{1}$$

Mean abusolute error, abbreviated as MAE, has the formula:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \tilde{y_i}| \tag{2}$$

The prediction value could have negative or positive difference compared with the real value. In order to get rid of negative error, MSE method squares the error and MAE method take an absolute value. From the numerical perspective, MSE amplify the error by squaring the difference if there are outliers in the dataset, which could ruin the entire model's predicting abilities. In the contrast, if the difference between the prediction and data is very small, the error converges to zero by MSE method. From the punishment perspective, the MAE method doesn't punish huge errors and give linear errors. In summary, **(1)** Someone take **MSE** method **when** there are obvious outliers in the dataset and want to punish huge errors if huge errors have more effect on prediction than small errors. **(2)** Someone take **MAE** method **when** the dataset contains a lot of noise, where outliers have limited effect on the accuracy. And it is usually used when the performance is measured on continuous variable data. **(3)** Both **MAE and MSE** are suitable when the outliers and influential points are deleted or not a lot noise in the dataset. And the magnitude of data is not very large.

Take data with **White Noise Process** in seasonal profit of a fashion company as an example. There are similar trends in each season by years. If taking linear regression or polynomial regression to fit data. The MSE can be rather larger than MAE.

Although the mean of white noise is zero, and noises are uncorrelated. But the sum of lots of squared white noise is huge, which leads to huge MSE result. However, in this seasonal profit situation, the white noise has limited effect on the prediction model because of its small values and the strong seasonal trend. Here, MAE are robust to white noises without ruining the prediction.

To discuss the difference of MSE and MAE deeply, download the **dataset** from one competition - ”House Prices: Advanced Regression Techniques” on kaggle platfrom [3] to do regressive prediction on ”SalePrice” and calculate the MSE and MAE. **Reason** for select this data: It contains 1459 recordings and has 63 variables, which provides effective factors thoroughly. And 1167 (80%) recordings are used to training model, the rest 292 (20%) recordings are used to test model.

First step, processing the dataset. Talking about the missing data, When more than 0.005% of the data is missing for each variable, we delete this variable and pretend it never existed: PoolQC(0.995), MiscFeature(0.963), Alley(0.938), Fence(0.807), FireplaceQu(0.473), LotFrontage(0.177), GarageCond(0.055), GarageType(0.055), GarageYrBlt(0.055), GarageFinish(0.055), GarageQual(0.055), BsmtExposure(0.026), BsmtFinType2(0.026), BsmtFinType1(0.025), BsmtCond(0.025), BsmtQual(0.025), MasVnrArea(0.005), MasVnrType(0.005) are also deleted. This suits to nearly all the variables except 'Electrical', Which has only one missing-recording count . In this case, we just delete this only one missing recording not the variable.

Then there 44 variables are catergorised to different data type 'numerical' and 'string'. From the figure 5 lists the top 10 most correlated variables to 'SalePrice', these 10 variables all are 'string' data type. To start a simplified predicting version, these variables with 'string' type are not quantified here and not being considered.

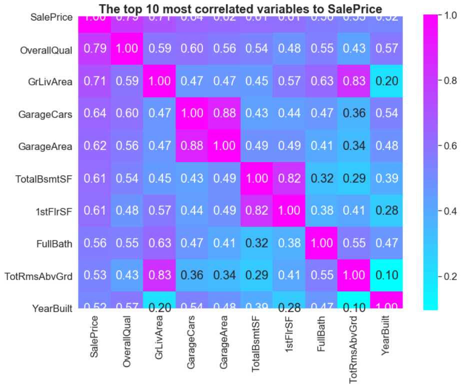


Fig. 5. The most 10 correlated and contributed variables to 'SalePrice'.

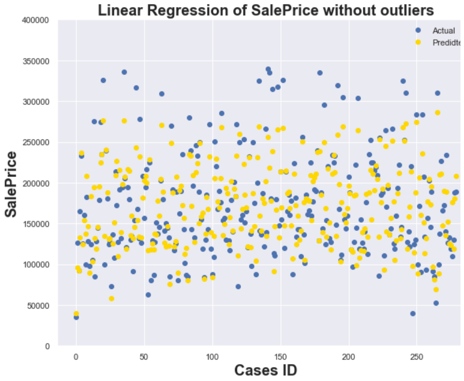
From the figure 5, it is obvious that 'GarageCars' and 'GarageArea'; 'TotalBsmtSF' and '1stFlrSF'; 'GrLivArea' and 'TotRmsAbvGrd' are highly correlated. We delete these less contributing factors and only 6 factors are left in the table 2.

Table 2. Information of train and test dataset

Attributes	Meaning	Min	Mean	Max
FullBath(int64)	Number of passenge	0	1.57	3
GarageCars(int64)	Size car capacity	0	1.77	4
GrLivArea(int64)	above ground living area	334	1515.5	5642
OverallQual(int64)	Overall quality	1	6.1	10
TotalBsmtSF(float64)	Area of basement	0	1057.4	6110
YearBuilt(int64)	construction year	1872	1971.26	2010
SalePrice(int64)	Price of property	34900	180921.2	755000

Second step, detect and delete outliers. The value of 'SalePrice' fall outside the 75 quantile and 25 quantile are considered outliers. In statistics, there are 61 outliers are dropped.

Third step, establish the Linear Regression and Random Forest regression model to predict the 'SalePrice'. From the Blue and Green points in the figures 6 to 7, we can see that the dependent variable in this housing price dataset [3] has a lot noise. In this cases, MSE can be very large.



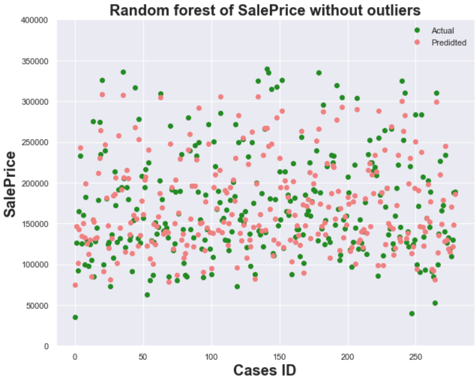


Fig. 6. MAE=22203, MSE=847379857

Fig. 7. MA20460, MSE=825145266

As the order of magnitude of housing 'SalePrice' is very large, both the MAE and MSE have a large order. In conclusion, the MSE is quite bigger than MAE when there are a lot of noise in the dataset.

4.3 Spam message detection

sentence segmentation and Topic modeling are suitable techniques for regular text message. First, we explore the **sentence segmentation** in details firstly.

There are several steps in the data transformation.(1) Read the .csv file and separating the labels and content into different table cells. And split the dataset into training and test section. (2)The most important transformation is how to extract the features from the text and put the features into classifiers. There are three parameters has been chosen **three parameters for this process: tokenization, removing stop-words,C**. The main process was introduced at *Part 1.2 step 2*. And We also use *GridSearchCV* to do cross validation and find optimal parameters. (3) After building *Logic regression* and *Naive Bayes* classifiers, we use test dataset to predict the results and calculate the accuracy. The accuracy of Logistic regression is 0.9892 and for Naive Bayes is 0.9883.

We also do another transformation, for **topic modeling**, with the help of 'gensim.model' (topic modelling in humans) in python, there are more than 100 classified topics in this dataset. And we contract the features from the email and math the topics in the topic modeling. And the most prevalent Sentence fragment in most prevalent topic are : 'bus',' boy', 'weeks', 'stop', 'online', 'min', 'love', 'grins', 'voucher', 'pain'. And using these topic to predict the results of 'LinearSVC' and 'Naive Bayes' model, The linear SVM has a test accuracy score of 0.907 after training 10^4 itertions. The Gaussian naive Bayes classifer has a test accuracy score of 0.676. And we can see the best classifier is Logic regression and accuracy is 0.9892. Scores of other three classifiers are lower than it. To improve it, deleting 'stopwords' first and increasing iteration times in Linear SVM would be helpful.

References

1. Data science bowl 2017. https://www.kaggle.com/c/data-science-bowl-2017

2. Data vis. https://github.com/Biedlin/Titanic84/blob/master/Titanic$_{8}$4.*ipynb*
3. House prices: Advanced regression techniques. https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data
4. Lung nodule analysis 2016. https://luna16.grand-challenge.org/download/
5. Movie review data. http://www.cs.cornell.edu/people/pabo/movie-review-data/
6. Random. https://stats.stackexchange.com/questions/285834/difference-between-random-forests-and-decision-tree
7. Fangzhou Liao, Ming Liang, Z.L.X.H.S.S.: Evaluate the Malignancy of Pulmonary Nodules Using the 3D Deep Leaky Noisy-or Network, vol. 10.1109. Computer Vision and Pattern Recognition (2019)
8. Kolchyna, O., Souza, T.T., Treleaven, P., Aste, T.: Twitter sentiment analysis: Lexicon method, machine learning method and their combination. arXiv preprint arXiv:1507.00955 (2015)
9. Witten, I.H., Frank, E., Hall, M.A.: Practical machine learning tools and techniques. Morgan Kaufmann p. 578 (2005)