Data Mining Techniques Assignment 1 - Group 51

António Évora Quirós Correia[2748054], Dian Liu[2700348], and Xiaoyang Sun[2734554]

Vrije Universiteit Amsterdam

1 Explore a Small Dataset

1.1 TASK 1A - Exploration

Notice all sorts of properties of the dataset The ODI dataset includes 304 rows and 16 features(exclude timestamp attribute), all attributes are recognized as Nominal, but we could convert some features to integer and date types.

Features Individual analysis:

- What programme are you in?
We can see although different names of major are shown such as CS, AI or computational science, there is a great majority of people having a programming-related background, while those doing finance also accounts for a large proportion. There are problems of inconsistent representation in the data set, for example, 'AI', 'Artificial Intelligence', 'artificial', we will deal with it in the follow-up to make the representation uniform. In addition, 5 instances with irrelevant meanings, such as 'cheese', 'chrome', will be replaced by 'unknown'.
- Have you taken a course on machine learning?
This feature is not giving too much room for filling different data types. As is shown in the statistics, about 60% respond "yes". This is to some extent not surprising based on their background. Still, there's a small feature type that people indicate "unknown", 7 people.
- Have you taken a course on information retrieval
A big majority (66%) respond 0 which indicates "No" for this question. "Not known(9%)", being the least shown category.
- Have you taken a course on statistics
This feature shows stronger polarization. Most people have taken a course in statistics. While the category of "unknown" relatively increases a lot.
- Have you taken a course on databases
This statistic shows those responding "yes" nearly equals to "no". Since the course "database" is more specific than "statistics", "unknown" still being the least common category, with the number of 7 people.
- What is your gender?
The majority (93.8%) of our population is male and female, 60.2% of them are male and 33.6% are female. 3.9% are "not willing to answer", and 2% are "gender fluid", being the least present category: "intersex".
- Chocolate makes you. . .
We can see most people say that chocolate has no effect on them. Being "neither" the Mode. Of the ones that notice an impact, the majority believe chocolate makes them "fat"(30%). The third most common category is people that don't understand the question. The least feature type is people who don't know what impact it has on them, 6 people.

- What is your birthday(date)?

This feature shows a big variety of values. Many people have inserted wrong/different date types. Cleaning this feature, or converting it to age, would be more useful to get better insights. We can see that the majority of our population was born between the years 1997 and 2001.

- Number of neighbors

This feature should've also been cleaned, since not all values are integers, and many integers are not valid/normal for the situation (negative and too high values).Nevertheless, the big majority of our population was alone. People who were not alone were mostly surrounded by 1, 2, and 5 neighbors.

- What is your stress level? (1-100)

This feature also needed to be pre-processed. Since we can see a great number of values that are not integers and that are not in the defined interval. Overall we can say that the majority of our population has a stress level above 50•You can get 100 euros...

This feature seems to be interesting. We can see answers for this question mainly focus between 0 and 100, so it seems reasonable to de-noise the other parts, such as numbers out of range and invalid value.

Data pre-processing and visualization

First of all, for the convenience of expression below, we simplified the name of each feature: 'programme', 'ML', 'IR', 'ST', 'DB', 'gender', 'chocolate', 'birthday', 'neighbour', 'standup', 'stress', 'deserve', 'random', 'bedtime','goodday1', 'goodday2'.

Due to different expressions of values in feature *programme*, we need to classify them. Take 'AI' as an example, 'artificial', 'Artificial' ,'ai', 'Ai' are all replaced by unified 'AI'. Then, we carried out similar unified processing for 'CS', 'Finance', 'Econometrics', 'CLS', 'BA' and other programmes. For some less common programmes, we used 'others' instead, for values with irrelevant meanings, such as 'cheese', 'chrome', was replaced by 'unknown'. After simplification, we used a pie chart to visualize the feature *programme*, as shown in Fig 1. Students from AI are the most, with a percentage of 37.5%. AI and CS account for more than 50%, while CLS and BA also account for about 25%. Other programmes, such as Econometrics, Finance, etc. are relatively few.

Then we visualized the gender ratio for each programme (taken in the data mining techniques course), as can be seen in Fig 2, there are significantly more male students than female students in most programmes except BA. Next step, we explored the four courses, features *ML, IR, ST, DB.*

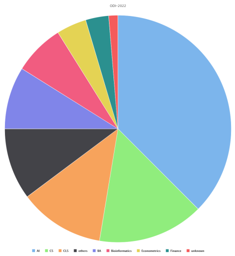

Fig. 1. Percentage of students from different programmes

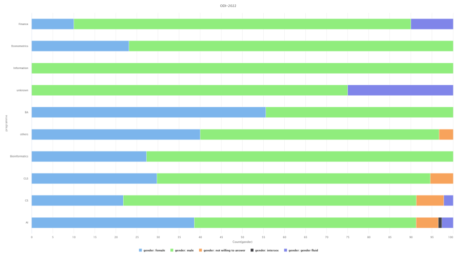

Fig. 2. Gender ratio by programme

Because the expression of each feature is different, such as '0' and '1' of *IR* and Dutch 'ja' and 'nee' of *DB*. We standardized the values of these four features, that is, 'yes', 'no', 'unknown'. As can be seen in Fig 3, Statistics are taken by vast majority(80%) of people, machine learning and database are taken by most people, while information retrieval(24.7%) is only taken by a few people.

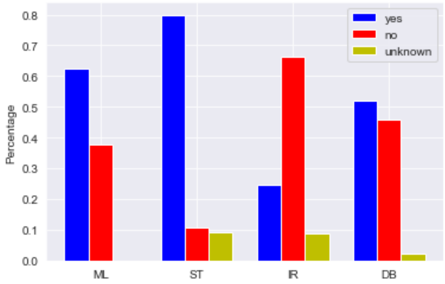

Fig. 3. Percentage of four courses

We then explored a series of gender-related issues. First is the relationship between *stress* and *gender*. We filtered the values of these two features *stress* and *gender* in pre-processing, keeping only 'male' and 'female' for gender and only numbers in range of [0,100] for *stress*. As shown in Fig 4, stress was higher overall for female students than male students who took the survey, with peaks around 15 for male and 60 for female. It is worth noting that the very high stress stage at [95,100] and the very low stage at [0,5] are all male.

We continue working on the relationship between feature *deserve* and *gender*,to some extent, the feature *deserve* reflects a person's self-confidence. We filtered the value of *deserve* as a number in range of [0,100] and removed invalid values. The results are shown in Fig 5, where the peaks for both male and female are concentrated in the lower band, while the middle band is more concentrated for women and the extremely low and extremely high band are all male.

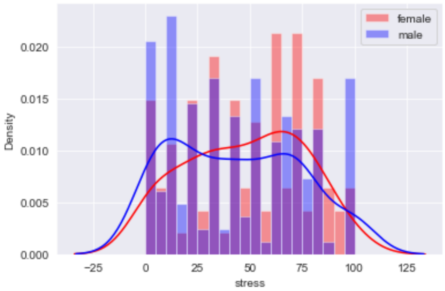

Fig. 4. Stress and gender

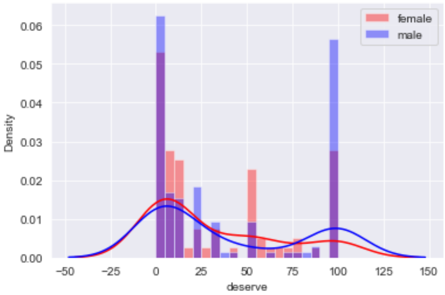

Fig. 5. Deserve(self-confidence) and gender

An interesting finding is that in both of the above experiments, the extremely low and extremely high values are all chosen by male, and not even one female chose these two range of values. In one study it was shown that male tend to be more extreme in their thinking, while women tend to be more moderate[1]. The results of these two experiments confirm the findings of this study.

We combined the features *goodday1* and *goodday2* into new feature *goodday*, divided each value into individual words and analysed them by using the python package word cloud. As shown in Fig 6, 'good', 'sun', 'food', 'friend', and 'sleep' were the five most frequent words. Of these, 'good' appears most often, but 'good' in the dataset appears mainly as a word in a phrase, e.g. good sleep, good food, which suggests that our approach does not take into account phrases.

In Task 1A, our last study is the relationship between *deserve* and *stress*, which can be explained as the relationship between a person's self-confidence level and stress level. We divide feature *stress* into five levels. As shown in Fig 7, people with medium stress have the highest levels of self-confidence, and those with high stress are slightly lower than those with medium stress, but much higher than those with low stress.

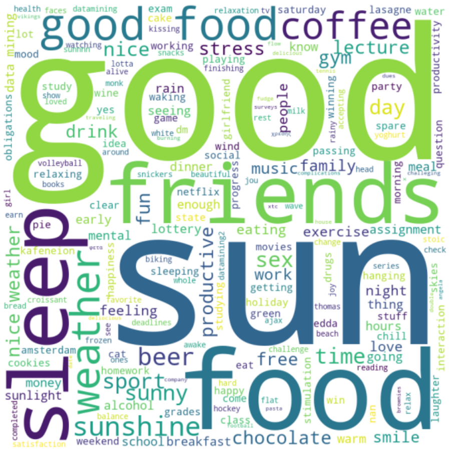

Fig. 6. Word cloud

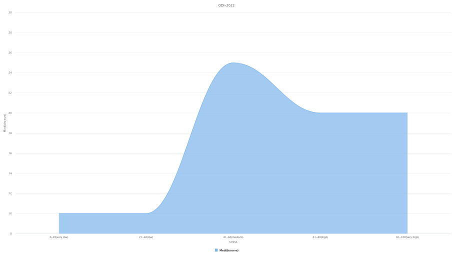

Fig. 7. Deserve(self-confidence) and stress

1.2 TASK 1B - Basic Classification/Regression

Since the ODI-2022 dataset had a large number of missing data and invalid values, we used a new dataset in this task. As each member of our group is a football fan, we downloaded the full dataset of FIFA21 players from the Skypool(Tianchi) platform and selected the goalkeepers for rating prediction[4]. After pre-processing, the data was reduced to a very concise level, consisting of 632 records and 15 object types attributes (including name, nationality, club, age). We used the ability values associated with the goalkeeper, such as *handing*, *reaction*, *reflexes*, to make predictions about the player's rating. We divided 70% of the dataset as training set and 30% as test set. As this is a multi-class classification, we focused on two algorithms, Naive Bayes and Support Vector Machines(SVM) Logistic Regression.

For Naive Bayes, we set the number of folds equals 10, which means each subset has equal number of subsets and the number of iterations that will take place is the same as the number of folds.For

SVM Logistic Regression, because all training data are normalized, we employ polynomial kernels. A polynomial kernel is defined by $k(x, y) = (x * y + 1)^d$, where d is the degree of the polynomial, which is specified by the kernel degree parameter[2]. The C value, which we set to 1, is the complexity constant of the SVM, which sets the tolerance for miss-classification, higher C values allow ”softer” boundaries, and lower values create ”harder” boundaries. A complexity constant that is too large can lead to over-fitting, while a value that is too small can lead to over-generalization[3].

For both of these algorithms, we predicted the average accuracy and calculated the scores by cross-validation. The results are shown in table 1, where the Naive Bayes was more accurate in its predictions, exceeding 80%, while the SVM performed poorly, with an accuracy of just over 50%.

Algorithm	Accuracy	Cross Validation
Naive Bayes	80.4%	88.2%
Logistics Regression-SVM	51.9%	52.4%

Table 1.

A reasonable analysis of the results is that Naive Bayes can effectively make multi-class predictions and has good performance with high-dimensional data. However, SVM only gives algorithms for binary classification, but we are solving a multi-classification problem, so SVM did not perform that well.

2 Compete in a Kaggle Competition to Predict Titanic Survival

2.1 TASK 2A - Preparation

Data Visualization and Analytics

The Titanic training dataset concludes 891 examples and 12 columns, the attributes are:

’PassengerID’, ’Survived’, ’Pclass’, ’Name’, ’Sex’, ’Age’, ’SibSp’, ’Parch’, ’Ticket’, ’Fare’, ’Cabin’, ’Embarked’

Firstly, we checked missing values. Feature *Age* has 177 missing values, feature *Cabin* has 687 missing values and *Embarked* has 2 missing values. Secondly, we did data visualization and analysed each feature.

Survived is the target classification feature, 1 represents survival, 0 represents unfortunate death. In the training set, the overall survival rate of passengers on the Titanic is 38.4%.

Pclass represents which class of cabin the passenger comes from, divided into first class, second class and third class. As can be seen from Fig 8, the survival rate of passengers from high class cabins is higher than that from lower class cabins. Therefore, this feature has a greater impact on survival.

Sex 64.8% of the passengers are male, the rest 35.2% are female. While, in Figure 9, it shows that female’s survival rate is 0.74, but male’s survival rate is only 0.19, female’s survival rate is nearly four times higher than male’s, even though men make up the majority of passengers.

Ticket There are 681 individual values out of 891 data, this does not give effective information for predicting survival, so this feature will be excluded from the model.

SibSp, Parch Which represents passenger’s siblings/spouses aboard the Titanic and parents/children aboard the Titanic. Both of these features can be interpreted as ”the presence or absence of family members on board”, so we have considered them together. 68% of the passengers boarded without siblings/spouses and 76% of them boarded without parents/children, the majority of the passengers boarded alone.

Cabin 687 of the 891 data are missing, so this feature will be excluded from the model.

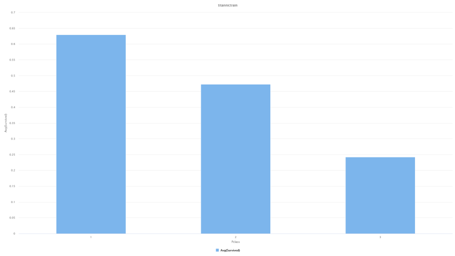

Fig. 8. Percentage of classes

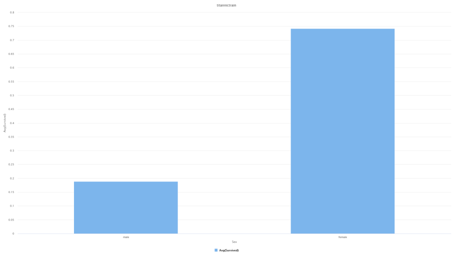

Fig. 9. Survival rate and sex

Fare This feature shows similar information to *Pclass*, the more fare passengers pay, the higher class they will get. Due to its similarity to the feature *Pclass*, this feature will be excluded from the model.

Embarked represents port of embarkation, 72% of the passengers boarded in Southampton, 28% of the passengers boarded in the rest two ports. Survival rate in Southampton is relatively lower than that in the other two ports, this can be explained by the fact that the majority of third class passengers embarked from Southampton. This feature can be more accurately explained in terms of the *Pclass*, so this feature was excluded from the model.

*Name*Names of the passengers themselves are not directly related to the survival rate, but in the context of the time, some titles can directly indicate the social status of the people at that time, which is closely related to the survival rate, so we will extract the title of each name in the next step to generate a new feature.

Age 177 out of 891 ages are missing, without regard to the missing ages, most passengers are between 20 and 35 years old. In Fig 10, we can notice that babies under 1 years old has 100% survival rate, with several survival peaks in other age groups.

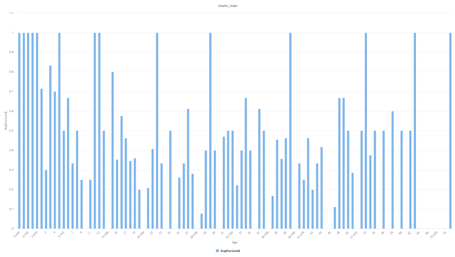

Fig. 10. Survival rate and age

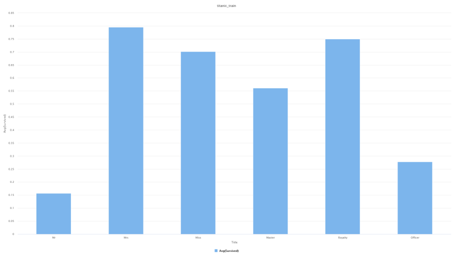

Fig. 11. Survival rate and title

Feature Engineering and Data Transformation *Baby* As mentioned above, people younger than 1 year old has 100% percent survival rate, and those aged less than five years also showed a significantly higher survival rate than other age stages. We have therefore created the new feature *Baby* to represent babies and toddlers up to 5 years old.

Title As mentioned above, the title of a name can reflect a person's social status, which is closely related to the survival rate, we extracted the titles from all passenger names and divided them into

6 groups: 'Mr', 'Mrs', 'Master', 'Miss', 'Officer', 'Royalty'. [5]Some of these titles are non-English (German, French, Dutch) and have also been grouped into the six groups above after a review. As can be seen in Fig 11, 'Mrs', 'Miss' for women and 'Royalty' for high class has higher survival rates, while 'Mr' and 'Officer' has lower survival rates.

FamilyMembers Sum the value of *SibSp* and *Parch* together, to calculate the family members passengers has. As can been seen in Fig 12, passengers who have 2-4 family members has relatively higher survival rates.

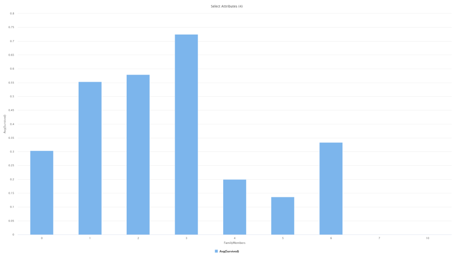

Fig. 12. Survival rate and family members

Since there are so many missing values for Age, it is not reasonable to use the median or average to fill in the missing values. Moreover, there are many features associated with age, such as **Title, Sex, Name**, etc., so we used another filling method: random forest regression trees.

After examining all the features in the dataset and their relevance, the following features were selected as predictors for the classifier: *Title, FamilyMembers, Baby, Age, Pclass, Sex.*

2.2 TASK 2B - Classification and Evaluation

Evaluation of Classifiers

We further split the training dataset of Titanic, with 70% divided into the training set and 30% into the test set. We used three classifiers from Rapidminer, Random Forest classifier, Decision Tree classifier and Neural Network classifier. The experimental results are shown in the following table 2. By comparing the training accuracy and testing accuracy of each classifier, Decision Tree performed the best and therefore we used the Decision Tree classifier. The data processing pipeline is shown in Fig 13. Our score in Kaggle is 0.76076, which is slightly less accurate than in our experiments.

Classifier	Training Accuracy	Testing Accuracy
Random Forest	93.33%	66.02%
Decision Tree	90.91%	78.57%
Neural Network	84.45%	76.07%

Table 2. Training and testing accuracy

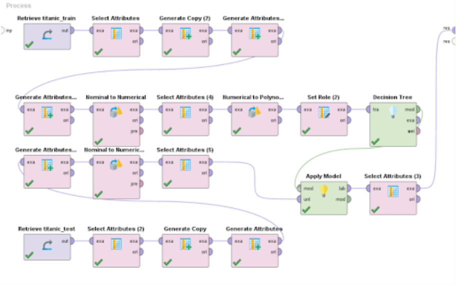

Fig. 13. Data processing pipeline

3 Research and Theory

3.1 Task 3A - STATE OF THE ART SOLUTIONS

For this assignment we choosed “Sberbank Housing Market” competiton, which was held 5 years ago. For this competition, competitors are required to develop algorithms to predict real estate prices using multiple features.

The main purpose of this competition is to predict the sales price of each property. The target variable is price in training set. The training data is from August 2011 to June 2015 and the test set is from July 2015 to May 2016. The dataset also includes information on the general state of the national economy and financial sector, so it is possible to focus on generating accurate price forecasts for each property without guessing how the business cycle will change.

From the winner team Alijs, I have found something interesting. First, instead of predicting directly for the target variable, the price per square meter was predicted and later transformed. Second, they tried a lot of independent models, because they found two variables in one piece that led to a big difference in the model (Investment and OwnerOccupier), and then put the two variables in two different sets of features input to the model. And they remove the outliers and train the model separately[6].

3.2 Task 3B - THEORY - MSE VERSUS MAE

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{1}$$

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

Why would someone use one and not the other?

If they have a data set with many outliers MSE will be much greater then MAE, since it squares the residuals, which will make the optimization algorithm fit more this data points and reduce the model accuracy, so in this case MAE would be a better option. Making sure that no single point overpowers the objective function. MAE gets us more robustness on this case.

On the other hand, MSE is really useful when you want to detect if the outliers are messing with your predictions, so you can remove then, or even better, explore them and discover some features that make them special, if possible.

When do MSE and MAE give identical results?

In a dataset where all the residuals are exactly ±1. MSE and MAE will return identical values of 1. In other words, if a dataset if "free of outliers" (the distance between predicted and real value is never too great), both loss functions give the same results. Also, but less probable, MSE and MAE would give same results if all the prediction errors are equal to zero.

Experiment

To the experiment the Insurance dataset from Kaggle was selected [7]. Age, sex, region, BMI, smoker and children are the features. Charges is the the contribute to predict. Using Linear regression we obtained a MSE equal to 4836.85 and a MAE equal to 3053.65. We then used Polynomial Regression and got MSE equal to 4265098.14 and a MAE equal to 23353973.06. We were exepcting the MSE to be bigger since this data contained many outliers, and those residuals were squared. We also got better results with linear regression, perhaps because Polynomial regression created a models with more unexplained variance, due to the increasing amount of regressors. Also, a more dedicated preprocessing section would decrease both MSE and MAE for both regressions.

3.3 Task 3C - THEORY - ANALYZE A LESS OBVIOUS DATASET

The dataset contains a total of 5574 text messages in the form of strings, including alphanumeric and non-alphanumeric characters. And the test set has to be retrieved in the form of Ham or Spam to predict whether the message under discussion is what we need.

When text messages are involved, the text is converted to numeric messages.The higher the amount of data that is obtained by the wide diversity of transformations applicable, the more accurate the prediction will be. For this reason, we seek to find a model that can manage a dense feature set. The dataset is divided into a training set containing 4000 data entries and a test set containing 1576 data entries. We could obtain the following feature from the text in the test set: length, number of words, average word length, number of numeric characters, number of words containing capital letters, and a matrix consisting of the count of every single word. These methods combined yielded 7330 features.

We then used these obtained features, along with the Ham/Spam labels that came with the dataset, to train via the Model Random Forest classifier. In the next test set, we applied the same transformations to the text (considering that the matrix had to be created to correspond to the words that appeared in the training texts), and then fed these features and labels from the test set back into the previously trained model. In view of the Random Forest classifier obtained a score of 97.7%, this training result is acceptable. At the same time, we can take additional steps to improve the results, such as computing non-alphanumerical characters to create more features, or pre-processing the text, including removing punctuation, capitalization, common words, tokenization, or finding the root of words through lexicology or morphological reduction.

References

1. Maccoby, E. E., Jacklin, C. N. (1980). Sex differences in aggression: A rejoinder and reprise. Child development, 964-980.
2. Jiang Liangxiao. (2009). Naive Bayes Classifier and Its Improved Algorithm Research. Doctoral Thesis, China University of Geosciences, Wuhan.
3. Smits, G. F., Jordaan, E. M. (2002, May). Improved SVM regression using mixtures of kernels. In Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No. 02CH37290) (Vol. 3, pp. 2785-2790). IEEE.
4. Tianchi Aliyun Dataset, FIFA21, https://tianchi.aliyun.com/dataset/dataDetail?dataId=93451

5. Kaggle Competition Titanic Tutorial, csdn.net, https://blog.csdn.net/wydyttxs/article/details/
6. Kaggle Competition sberbank-russian-housing-market, discussion section: https://www.kaggle.com/c/sberbank-russian-housing-market
7. Kaggle, Medical Cost Personal Datasets, Miri Choi, https://www.kaggle.com/datasets/mirichoi0218/insurance