Journal of Actuarial, Finance and Risk Management (JAFRM)
Vol. 1(2), November, 2022.

Identifying Fraud in Automobile Insurance Using Naïve Bayes Classifier

Dadang Amir Hamzah[1*], Annisa Sentya Hermawan[2], Shintya Jasmine Pertiwi[3], Syarifah Intan Nabilah[4]

[1,2,3,4] Study Program of Actuarial Science, School of Business, President University, 17550, Indonesia
**Corresponding Author: dadang.hamzah@president.ac.id*

Abstract— In this article, the Naïve Bayes Classifier is employed to detect fraud in automobile insurance. The Naïve Bayes classier is a simple probabilistic method based on the Bayes theorem. The data used in this article is determined from databricks.com which consists of 40 attributes and 1000 entries. The target attribute that will be predicted consists of two categories," yes" or "no", which inform whether there is a fraud or not. The Data is split into training and testing with suitable proportions. Based on training data, the Naïve Bayes Classifier is applied to the testing data and returns the predictions data. Then, the prediction data is compared with the actual data to see the performance of the method. The result shows that the Naïve Bayes Classifier gives a good result to predict the insurance fraud with 78% accuracy, 67% precision, 3% of recall, and 6% of F1 score for "Yes".

Keywords— Automobile Insurance; Fraud Detection; Naïve Bayes Classifier.

I. INTRODUCTION

Automobile insurance is a contract between the Insured and the Insurance company which protects the Insured from financial loss in the event of accident or theft. The Insured agrees to pay some amount of money which is called a premium and in exchange, the Insurer company agrees to pay the Insured losses as written in an insurance policy. The Auto insurance provides coverage for several components. Those components are property which includes damage or theft, liability which includes legal responsibility to others for bodily injury or property damage, and medical which includes the cost for treating injuries, rehabilitation, lost wages, and even funeral expenses. According to Bodaghi et. al [1] insurance fraud is an illegal act that is done by either the seller of an Insurer or the Insured party of an insurance contract. Insurance fraud from the seller includes selling policies from non-existence companies, failing to submit premiums, and churning policies to create more commission. On the other hand, the fraud from the Insured party includes excessive claims, falsified medical history, post-dated policies, faked death or kidnapping, and murder. In this paper, we will focus on the fraud case which is done by the Insured party. Several studies about automobile insurance fraud are as follows. Bodaghi et.al in [1] studies fraud in automobile insurance using social network analysis, Caruana and Grech in [2] compared artificial Neural network technique and naïve Bayes classifier. They show that both classification techniques gave comparable results. Ghorbani et. al in [3] studies fraud in automobile insurance using a data mining based approach, Liu et. al in [4] applies the evidential reasoning approach and data-driven inferential modelling to detect automobile insurance fraud, Majhi et. al in [5] use fuzzy clustering using salp swarm algorithm to detect fraud, Wang in [6] use leveraging deep learning with LDA-based text analytics to detect the fraud, and Nugroho in [7] use Naïve Bayes Classifier to predict the film rating. In this paper, we use Naïve Bayes Classifier since it gives a good result in detecting fraud in automobile insurance according to [2]. The data that we use is taken from Kaggle.com which is a public domain data where the source is unknown.

The organization of this paper is given as follows. In section 2 the Naïve Bayes Classifier is explained. This section discusses how the method can be applied to perform the classification along with the assumptions therein. The application and result are discussed in section 3. Finally the conclusion and the possible future works in given in section 4.

II. METHOD

The presentation in this section is summarized from [8] and [9]. The Naïve Bayes Classifier is a probabilistic classifier based on the Bayes theorem. Let X and Y be random variables of an event. The conditional probability $P(Y|X)$ calculates the probability of occurrence of event Y prior to the occurrence of event X. This probability is defined as

$$P(Y|X) = \frac{P(X \cap Y)}{P(X)} \quad (1)$$

and the conditional probability $P(X|Y)$ is defined as

$$P(X|Y) = \frac{P(X \cap Y)}{P(Y)} \quad (2)$$

By (1) and (2) we can express probability $P(X \cap Y)$ in two ways that are

$$P(X \cap Y) = P(X)P(Y|X) \quad (3)$$

and

$$P(X \cap Y) = P(Y)P(X|Y) \quad (4)$$

Bayes theorem is used to revise previously calculated probabilities based on new information. Developed by Thomas Bayes in the eighteenth century. The Bayes theorem is an extension of previously learned about conditional probability. Suppose $Y = \{Y_1, Y_2, \dots, Y_n\}$ where Y_k's for $k = 1,2, \dots, n$ are mutually exclusive and collectively exhaustive events which denote the classification classes and $X = \{X_1, X_2, \dots, X_n\}$ be the attributes or variables that will be used for classifications. Now the equation (1) for $Y = Y_k$ with fixed k, can be written as

$$\begin{aligned} P(Y_k|X_1, X_2, X_3 \dots X_n) &= \frac{P(Y_k \cap \{X_1, X_2, X_3, \dots, X_n\})}{P(X)} \\ &= \frac{P(Y_k)\, P(\{X_1, X_2, X_3, \dots, X_n\}|Y_k)}{P(X)} \\ &= \frac{P(Y_k)P(X_1|Y_k)P(X_2|Y_k)P(X_3|Y_k)\dots P(X_n|Y_k)}{P(X_1)P(X_2)P(X_3)\dots P(X_n)} \end{aligned}$$

where from line one to line two the eq. (4) is applied and from line two to line three the same approach is repeated. Because for each Yk the denominator $P(X_1)P(X_2)P(X_3) \dots P(X_n)$ is fixed then this value can be omitted, therefore $P(Y_k|X_1, X_2, \dots, X_n)$ is proportional to $P(Y_k)P(X_1|Y_k)P(X_2|Y_k)P(X_3|Y_k) \dots P(X_n|Y_k)$. An entry with attributes $X_1, X_2, \dots, X_n$ can be classified into class Y_k if

$$P(Y_k)P(X_1|Y_k)P(X_2|Y_k) \dots P(X_n|Y_k) = \max_{i=1,2,\dots,n} \{P(Y_i)P(X_1|Y_i)P(X_2|Y_i) \dots P(X_n|Y_i)\} \quad (5)$$

The Naïve Bayes Classifier relies on two important assumptions. First, there are no hidden or latent attributes. In other words, the set of features in X is complete; Second, all attributes are independent of each other given the class, so that:

$$P(X_1, X_2, \dots, X_n|Y_k) \approx \prod_{i=1}^{n} P(X_i|Y_k) \quad (6)$$

This assumption reduces the number of parameters to be estimated.

III. Application And Results

In this section, the Naïve Bayes Classifier will be constructed. The data used in this paper are taken from databricks.com named insurance_claim.csv. The construction of the Naïve Bayes Classifier will involve three processes that are data observation, data pre-processing, and Classifier building and evaluation. Data observation is aimed to determine the independent variables $\boldsymbol{X}$ and target variable $\boldsymbol{Y}$ and to observe the contains of the data which is known as data type. Data pre-processing is aimed to examine whether there is missing data, inconsistency, non-numerical observation, and outlier data. This process will make sure that the data are ready to analyze that is by removing all the mentioned situations. The data are then divided into training and testing. The final step is to construct the Naïve Bayes Classifier and perform an evaluation. The classifier will be applied to training data and return the estimated claim data. The resulted data are then compared with the actual data to examine the Naïve Bayes Classifier performance.

A. *Data Observation*

The data is explored using python data analysis library or Pandas [10]. The data consist of 40 columns with 1000 entries. The target column that will be predicted is labelled as fraud_reported which consists of "Yes" or "No" responses. In this column 24.7% reported "Yes" and 75.3% reported "No" and there is no missing data found in the whole entries of the columns. The list of attributes in this data is depicted in Figure 1. To avoid overfitting several attributes are removed from the analysis. The reason of removing these attributes is due to the high variance of responses. Too many attributes will confuse the classifications because of too many conditions that need to be satisfied. Other than that, incompatibility with the target attribute becomes another reason why these attributes are removed from the analysis. Now, there are 10 attributes left that will be used to build the Classifier. Those attributes are depicted in Figure 2.

B. Data Pre-processing

Among all attributes in Figure 2, the following conversion is performed. The entries in 'Total_Claim_Amount' attribute are converted into categorical data which consists of three categories. The data are divided into three equal parts and labeled as "1" if the data is greater than 66% of the data, labelled as "2" if the data is between 33% and 66% of the data, and labelled "3" if the data is lower than 33% of the data. The same treatment is also applied for "Policy_Annual_Premium", and "Age". The other attributes are also converted into categorical data with a various number of categories that depend on the number of unique answers in each attribute.

C. Classifier Building and Evaluation

The Naïve Bayes Classifier is constructed based on categories in each attribute. The package used to build the Naïve Bayes Classifier is the Scikit-Learn Package with Gaussian Probabilities. The data are divided into training and testing with 70% for training and 30% for testing. The parameters that are used to test the classification result are Precision, recall, f1 score, and accuracy. To determine all the parameter values we need to compare the classification result with the actual result. The observations are recorded as True positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). The True positive (TP) records the number of predicted "Yes" that agree with the actual "Yes" responses, the True Negative (TN) records the number of predicted "No" that agree with the actual "No" responses, the False Positive (FP) records the number of predicted "Yes" but actual responses is "No", and False Negative (FN) records the number of predicted "No" but actual responses is "Yes". In this observation we have TP = 2, TN = 233, FP = 64, and FN = 1. Accuracy is the ratio between the correctly classified data with the total number of data. In our case, the accuracy is 78%. Precision is the ratio between correctly predicted data with the total of correctly and falsely predicted observations. In our observation, the precision for "No" is 78% while the precision for "Yes" is 67%. Recall is a measure of positivity or sensitivity defined as the ratio between correctly predicted answers with the total of correctly predicted for all answers. In our observation, the recall for "Yes" is 3% and the recall for "No" is 100%. F1 is the metric that includes precision and recall defined as 2 times precision times recall divide by the precision plus recall. In our observation, the f1 score for the "No" answer is 88% and the f1 score for the "Yes" answer is 6%. The result of the Naïve Bayes Classifier is depicted in Figure 3.

IV. CONCLUSION

The fraud claim in automobile insurance has been detected using Naïve Bayes Classifier based on "Age" of the insured, is there any "Authorities_Contacted", number of "Incident_Hour_of_the_Day", "Total_Claim_Amount", "Month_as_Customer", "Police_Report_Available", "Policy_Annual_Premium", and "Insured_Sex". The precision of this classifier is 78% for reported as "No" fraud and 67% for reported "Yes" there is a fraud. Compared to the previous results such as [1], [2], [3], and [4], this method offers an alternative approach to detect fraud with a compatible result. This result can be improved by considering attributes that are highly influencing the target attribute. Another factors that can influence the results is the pre-processing steps where the classification label can be performed by considering the relation of one attribute with another. Finding and processing those attributes will give interesting challenges and left as future works.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 40 columns):
 #   Column                       Non-Null Count  Dtype  
---  ------                       --------------  -----  
 0   months_as_customer           1000 non-null   int64  
 1   age                          1000 non-null   int64  
 2   policy_number                1000 non-null   int64  
 3   policy_bind_date             1000 non-null   object 
 4   policy_state                 1000 non-null   object 
 5   policy_csl                   1000 non-null   object 
 6   policy_deductable            1000 non-null   int64  
 7   policy_annual_premium        1000 non-null   float64
 8   umbrella_limit               1000 non-null   int64  
 9   insured_zip                  1000 non-null   int64  
 10  insured_sex                  1000 non-null   object 
 11  insured_education_level      1000 non-null   object 
 12  insured_occupation           1000 non-null   object 
 13  insured_hobbies              1000 non-null   object 
 14  insured_relationship         1000 non-null   object 
 15  capital-gains                1000 non-null   int64  
 16  capital-loss                 1000 non-null   int64  
 17  incident_date                1000 non-null   object 
 18  incident_type                1000 non-null   object 
 19  collision_type               1000 non-null   object 
 20  incident_severity            1000 non-null   object 
 21  authorities_contacted        1000 non-null   object 
 22  incident_state               1000 non-null   object 
 23  incident_city                1000 non-null   object 
 24  incident_location            1000 non-null   object 
 25  incident_hour_of_the_day     1000 non-null   int64  
 26  number_of_vehicles_involved  1000 non-null   int64  
 27  property_damage              1000 non-null   object 
 28  bodily_injuries              1000 non-null   int64  
 29  witnesses                    1000 non-null   int64  
 30  police_report_available      1000 non-null   object 
 31  total_claim_amount           1000 non-null   int64  
 32  injury_claim                 1000 non-null   int64  
 33  property_claim               1000 non-null   int64  
 34  vehicle_claim                1000 non-null   int64  
 35  auto_make                    1000 non-null   object 
 36  auto_model                   1000 non-null   object 
 37  auto_year                    1000 non-null   int64  
 38  fraud_reported               1000 non-null   object 
 39  _c39                         0 non-null      float64
dtypes: float64(2), int64(17), object(21)
memory usage: 312.6+ KB
```

Figure 1. List of attributes 1.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 9 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Age                       1000 non-null   int64
 1   Authorities_Contacted     1000 non-null   int64
 2   Incident_Hour_of_the_Day  1000 non-null   int64
 3   Total_Claim_Amount        1000 non-null   int64
 4   Month_as_Customer         1000 non-null   int64
 5   Fraud_Reported            1000 non-null   int8 
 6   Police_Report_Available   1000 non-null   int64
 7   Policy_Annual_Premium     1000 non-null   int64
 8   Insured_Sex               1000 non-null   int8 
dtypes: int64(7), int8(2)
memory usage: 56.8 KB
```

Figure 2. List of attributes 2.

	precision	recall	f1-score	support
0	0.78	1.00	0.88	234
1	0.67	0.03	0.06	66
accuracy			0.78	300
macro avg	0.73	0.51	0.47	300
weighted avg	0.76	0.78	0.70	300

Figure 3. Classification Report

REFERENCES

[1] A. Bodaghi and B. Teimourpour, “Automobile Insurance Fraud Detection Using Social Network Analysis,” 2018. doi: 10.1007/978-3-319-95810-1_2.

[2] M. A. Caruana and L. Grech, “Automobile insurance fraud detection,” *Commun Stat Case Stud Data Anal Appl*, vol. 7, no. 4, 2021, doi: 10.1080/23737484.2021.1986169.

[3] A. Ghorbani and S. Farzai, “Fraud Detection in Automobile Insurance using a Data Mining Based Approach,” *International Journal of Mechatronics, Electrical and Computer Technology (IJMEC)*, vol. 8, no. 27, 2018.

[4] X. Liu, J. B. Yang, D. L. Xu, K. Derrick, C. Stubbs, and M. Stockdale, “Automobile insurance fraud detection using the evidential reasoning approach and data-driven inferential modelling,” in *IEEE International Conference on Fuzzy Systems*, 2020, vol. 2020-July. doi: 10.1109/FUZZ48607.2020.9177589.

[5] S. K. Majhi, S. Bhatachharya, R. Pradhan, and S. Biswal, “Fuzzy clustering using salp swarm algorithm for automobile insurance fraud detection,” in *Journal of Intelligent and Fuzzy Systems*, 2019, vol. 36, no. 3. doi: 10.3233/JIFS-169944.

[6] Y. Wang and W. Xu, “Leveraging deep learning with LDA-based text analytics to detect automobile insurance fraud,” *Decis Support Syst*, vol. 105, 2018, doi: 10.1016/j.dss.2017.11.001.

[7] R. W. Pratiwi, D. Yusuf, and S. Nugroho, “Prediksi Rating Film Menggunakan Metode Naïve Bayes,” *Jurnal Teknik Elektro*, vol. 8, no. 2, 2016.

[8] R. James, G., Witten, D., Hastie, T., Tibshirani, *An Introduction to Statistical Learning - with Applications in R | Gareth James | Springer*. 2013.

[9] D. Berrar, “Bayes’ theorem and naive bayes classifier,” in *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, vol. 1–3, 2018. doi: 10.1016/B978-0-12-809633-8.20473-1.

[10] W. McKinney and P. D. Team, “Pandas - Powerful Python Data Analysis Toolkit,” *Pandas - Powerful Python Data Analysis Toolkit*, 2015.