# Report #14310919

CHAPTER 1 INTRODUCTION Background Passenger satisfaction is one of the important factors for the improvement of an airline. The airline can find out what things need to be improved. With the hope that more and more airplane passengers use the airline, of course this increase must be done so that income also increases. To improve service, of course, you must know what things make passengers satisfied. This can be done from the data of passengers who have traveled by plane. In this digital era, data is very easy to store and obtain. Not like in the past, which used paper to record data, but used the help of computers. One of the advantages is that it is easy to store large amounts of data, including passenger satisfaction data. If there are about 130,000 airline passenger satisfaction data, of course it is very difficult to process manually. This will make it difficult for airlines to improve services. Because data storage uses a computer, we can also use a computer to process it. However, to process the existing data in order

to get the results we want, an algorithm is needed. With the algorithm implemented on passenger satisfaction data, we can classify things that can make passengers satisfied with airline flight services. Of course, this is better than processing thousands of data manually. Therefore, this time I implemented the Learning Vector Quantization (LVQ) and Naïve Bayes algorithms on the airline passenger satisfaction data that I got through Kaggle. It is hoped that this algorithm can process thousands of existing data and classify them. I am using 2 different algorithms so that I can compare the results of each implemented algorithm. And also, to find out which algorithm is better for classifying airline passenger satisfaction data by comparing the accuracy of the two algorithms. The results of this classification algorithm are expected to help airlines know what to do in the future.
Problem Formulation From the background above, we can formulate the existing problems. Can the Naïve Bayes algorithm classify airline passenger satisfaction data? Can the Learning

Vector Quantization algorithm classify airline passenger satisfaction data? Based on the level of accuracy, which algorithm is better in classifying passenger satisfaction data? Scope In this project, I applied Learning vector quantization and Naive Bayes algorithm only for the data I used from https://www.kaggle.com/binaryjoker/airline-passenger-satisfaction with 129,880 data. The data consists of 23 measuring columns and 1 response column. To find out a better algorithm, I use the accuracy parameters of each algorithm. There will be 5 tests for each algorithm with a percentage of training data of 90%, 75%, 50%, 25%, and finally 10%. Objective The purpose of this project is to find out whether the Learning Vector Quantization and Naive Bayes algorithms can classify aircraft passenger satisfaction from existing data. In addition, to find out from the two implemented algorithms, which algorithm is better based on the level of accuracy. CHAPTER 2 LITERATURE STUDY Gorzalczany et al. [1] explain that a lot of data mining does not provide deeper

explanations and justifications than decisions. Therefore, they apply their knowledge discovery technique based on fuzzy rules to the problem of airline passenger satisfaction. They used a dataset from Kaggle of 259,760 records. With 23 variable columns, the dataset is almost the same as the dataset that I will use. The results obtained are that the most significant attribute is Inflight Entertainment with an accuracy of 75.2%. Followed by the attributes of Seat comfort and Inflight Wi-Fi Service. They do not classify, but can determine which classification variables affect airline passenger satisfaction more. With the US Airlines dataset which is almost the same as before, Hayadi et al. [2] uses several classification algorithms. The algorithms used are KNN, Logistic regression, Gaussian NB, Decision Trees and Random Forest. The author runs using the GridSearchCV algorithm from Scikit-Learn. Of all the algorithms that have been run, Random forest has the best performance with 99% accuracy, 97% precision and 94% recall. From the many simulations carried out, the

authors suggest optimizing the in-flight wi-fi service. After that also simplicity about online booking. Unlike before, this time with around 130,000 data that becomes 70,000 after deleting the NaN (Not a Number) value, it doesn't include inflight entertainment as an attribute that needs to be improved. Different from the previous ones, but still about airline customer satisfaction. Hanif et al. [3] uses a dataset of 152 respondents who have used one of the Indonesian airlines, namely Lion Air. The data is taken and grouped by occupation so that it becomes 100 data and 5 classes of work. The author uses the SPSS tool to get the conclusions. By looking for multiple regression, validity, reliability, T test, F value test and the coefficient of determination and correlation, it is found that there is a positive and significant influence between service quality, passenger satisfaction and passenger behavioral intentions. The disadvantage of this research is that the data used is too little so that it can get different results if there are

more datasets. In the journal written by Wijayanto et al. [4], the Naive Bayes algorithm is also used for the passenger satisfaction dataset taken from Kaggle. The dataset used is most likely the same as that which will be used from this journal. With 129,880 data, the author uses the help of the KNime application for classification with Naive Bayes. The distribution of training data and data testing consists of 4 experiments. The first is training data: testing data is 90:10, the second is 85:15, the third is 80:20 and the last is 75:25. The results obtained that 90% of training data and 10% of testing data have an accuracy of 81.466%. Religia and Amali [5] also uses Naive Bayes to classify airline passenger satisfaction. The dataset used is also from Kaggle but is different, as many as 25,976 data. In their research, they used Naive Bayes, Naive Bayes optimized particle swarm Optimization (PSO) and finally Naive Bayes optimized Genetic Algorithm (GA). 1 To measure the performance used accuracy, precision and recall. The results obtained are that Naive Bayes optimized by PSO has the best results, namely the accuracy value is 86.13%, the precision value is 87.9% and the recall value is 87.29%. Similar to this journal, Nugraha et al. [6] compare Naive Bayes with Learning Vector Quantization (LVQ) to classify. But here it is used to

classify uterine diseases. In using Naive Bayes, the author uses 2 methods, Naive Bayes by using Laplacian Smoothing and without using it. The data used are 125 data from the medical records of patients at RSUD Dr. Moewardi Solo. The data here is divided into 4 experiments/simulations with the first experiment being training: the data is 20:80, the second is 40:60, the third is 60:40 and the last is 80:20. The results of 4 trials with training 20%, 40%, 60%, 80% got Naive Bayes without Laplacian Smoothing had 32%, 67.8%, 79%, 88.8% accuracy. These results are less good than if Naive Bayes using Laplician smoothing has an accuracy of 88%, 92.4%, 92.8%, 92.4%. The accuracy is said to be stable even though the training data is changed. Compared to LVQ the accuracy is 82.4%, 88.8%, 89.4%, 95.2%. However, the highest accuracy is obtained from LVQ with 80% training. In another journal, for LVQ signature pattern recognition compared by Prabowo et al. [7] and combined by Ginting et al. [8]. Prabowo et al. compared with the Kohonen Neural Network (KNN), while Ginting et al. combined with Self Organizing Kohonen (SOK). In the journal Prabowo et al. did 3 tests. Each test with a different number of classes, resolutions and patterns. In the first test with 25 patterns and a resolution of 30x20 Kohonen had 96% success for 1 second

while LVQ was 100% for 2 seconds. Second with 40 patterns and 30x20 resolution with 95% Kohonen less than 1 second while LVQ 92.5% less than 1 second. The last test was 9 patterns with 100x100 resolution with 77.78% Kohonen for 2 seconds and LVQ 88.89% for 7 seconds. LVQ does have better accuracy than Kohonen, but it takes longer. While in the journal Ginting et al. can speed up the computational process. The combination of LVQ with SOK increases the processing speed of computing during training or during signature pattern recognition. Unlike previous comparisons or combinations, Meliawati et al. [9] implement LVQ to predict majors at SMA PGRI 1 Banjarbaru. The data used is obtained from the value of report cards in 2010, 2011 and 2013. The data is used as training data, while the value of report cards in 2014 is used as testing data. It is not known how much of the exact amount of data was used. Researchers get 79.31% accuracy for iterations 60 and 90. Samsir [10] also implements LVQ. LVQ is used to classify Throat Nose and Ear (ENT) disease at Rantauprapat Hospital Labuhanbatu. The input variable consists of 10 disease symptoms. The dataset used is small, which is only 57 data. Of the 57 data divided into 4 training. With the comparison of training data: Testing data is 60:40, second

70:30, third 80:20 and 90:10. In the results of testing accuracy, it is not found that the more testing data, the accuracy will improve. Maybe it's because there are too few datasets, so you might get different results if you get more datasets. From the journal Gorzalczany et al. [1] and Hayadi et al. [2], the dataset used is almost the same. But both use different algorithms in classifying them. While Hanif et al. [3] using very different datasets and different algorithms, but it's still about passenger satisfaction. However, Wijayanto et al. [4] using the same dataset and algorithm, namely Naive Bayes only, but not compared to LVQ. Likewise, Religia and Amali [5] use only Naive Bayes to classify airline passenger satisfaction, but the datasets used are different. In the journal Nugraha et al. [6] The algorithms both compare LVQ and Naive Bayes, but they use it to classify obstetrical diseases. Prabowo et al. [7] also compared LVQ but with KNN for the case of signature pattern recognition. While Ginting et al. [8] combines LVQ with SOK for signature pattern recognition cases as well. For Meliawati et al. [9] and Samsir [10], they only implement LVQ with different datasets without comparing them or combining them. CHAPTER 3 RESEARCH METHODOLOGY Data Collection In collecting datasets, I use websites that provide various

kinds of datasets. For this research I used data from https://www.kaggle.com/binaryjoker/airline-passenger-satisfaction. Data with the file name airline_passengeer_satisfaction.csv has a file size of 14.34MB. I downloaded this data on September 20, 2021. To download it you are required to Sign In first (Register if you don't have an account). The downloaded file will be a zip file, so it must be extracted to get the csv file. The total data obtained were 129,880 with 23 measuring columns and 1 response column. Algorithm In choosing the algorithm, I consulted my supervisor. During the consultation, my lecturer informed and suggested the Learning Vector Quantization (LVQ) algorithm. This algorithm has not been used very often. Therefore, I use this LVQ algorithm. After using LVQ I looked for another algorithm to use as a comparison. Then I chose Naive Bayes because this algorithm is an algorithm that is often used, easy and has good accuracy. I use these two algorithms to classify supervised learning data about airline passenger satisfaction that has been obtained previously. In addition to knowing which algorithm is better in accuracy. Coding and Design In this step, the MySql tools will be used. MySql is used because the existing dataset is in the form of 2-dimensional data (columns and rows) the same as the MySql database

table. In addition, the installation of Mysql is very easy.
By downloading xampp through the website https:
//www.apachefriends.org/download.html. Xampp already provides severa
l versions for Windows, Linux and OS X operating systems.
Here I use Linux. After MySql is installed, the data will
be preprocessed. Continuous data such as age and distance
will be changed first to make it easier to classify.
Analysis In analyzing, I will do 5 tests as follows:
Analysis In this analysis, it is divided into 5 stages to
determine whether the amount of training has an effect.
Influence on Naive Bayes accuracy and on LVQ accuracy. Make
a Report In making the report, I wrote chapters 1-4 first.
After chapter 4 finished, I started the coding stage for
program development. Then the results that have been carried
out during the coding stage will be recorded in the
chapter 5 report. And finally, conclusions will be drawn
from the results of the coding stage which will be written
in chapter 6. ANALYSIS AND DESIGN In this research, there
are several steps in outline. The first to get the data.
The second is data preprocessing. Continued implementation of
Naive Bayes and Learning Vector (LVQ) and the last is
calculating accuracy. The flow is as in the following
workflow: Workflow The first workflow is getting data. The

data I use is data taken through Kaggle on September 20, 2021. The file can be downloaded with the file name airline_passenger_satisfaction.csv via the link https:/ /www.kaggle.com/binaryjoker/airline-passenger-satisfaction. The data has 129,880 records in all. Has 24 attributes consisting of id, 22 input attributes and 1 label attribute. These attributes are as in table 4.1. Data Table The id attribute is only used as the line numbering of each record. Meanwhile, the gender attribute to the delay in minutes attribute will be used as input variables for both algorithms. The input variable is the value of the attribute. 2 For example, the variables of gender are female and male. And lastly, the satisfaction attribute is a label attribute. The label attribute is an attribute that already contains the class of each record because the algorithm that will be used is supervised learning, which is an algorithm where the class has been determined. The class consists of 2 classes, namely the "satisfied" and "neutral or dissatisfied" classes. After the data is obtained, the next step is to enter the data into the database.
The data is entered into the database so that it can be processed by the program. From the existing data as shown above, there is data that cannot be processed by the program. Therefore, according to the workflow, the next step

after getting the data is "data preprocessing" . In this step the data that has a null value will be deleted first. This is so that the data processed is quality data. In preprocessing there are also attribute records that will be changed. Notes will be converted to numbers at small intervals. For example, there are too many age categories, which will then be changed to "0" where the age is <28, then "1" where the age is between 28 and 52, and finally "2" which is over 52. The value of 28 and 52 is based on quantile values that can be seen on the data link i s downloaded. There is also data that is not in the form of numbers will be converted to numbers. This is because the LVQ algorithm will be calculated based on the value of the attribute. So that it is converted into a number so that it can be calculated. For example, the original gender "Female" and "Male" will be changed to "0" and "1". The attributes that are changed in the preprocessing stage are as follows: Modified Attribute Data Table In addition to changing the data, in the preprocessing, deletion of data will be carried out. Deleted data are records that have attributes with null or empty values. This is done so that the data can be processed by the program. I did not change the blank data with 0 or 1 to maintain the

quality of the existing data. After deleting the data, the preprocessing step has been completed. The next step is to implement an algorithm for airline passenger satisfaction data. In implementing the two algorithms, 5 tests will be carried out on each algorithm. In each test, the amount of training data and testing data will be different. The difference in the amount of data is later to see whether the amount of different data will affect the final result. Comparison of the amount of data as shown in the following table. Distribution of Training and Testing Data As in the workflow, after preprocessing it will implement Naive Bayes. Naive Bayes will be tested up to 5 times. Each test will use a different number of datasets as shown in table 4.3. And at the end of the Naive Bayes implementation, the accuracy value will be calculated. Likewise with LVQ, which will test 5 times and look for accuracy. In finding the value of accuracy will use the formula. The formula used is like the following function. True Positive (TP) = Tota l class 1 (satisfied) and classified as class 1 True Negative (TN) = Total class 0 (neutral or dissatisfied) an d classified as class 0 False Negative (FN) = Total class 1 (satisfied) and classified as class 0 False Positive (FP) = Total class 0 (neutral or dissatisfied) and classified a

s class 1 Accuracy = The result of dividing the number o f correct classifications with the total data and multiplied by 100% The formula above will be used to find the accuracy value of each test from the two algorithms. Therefore in each test will be calculated the number of TP, TN, FP and FN. After all the tests are complete, the accuracy value of all the tests will be obtained. For the first, testing will be carried out using the Naive Bayes algorithm. This algorithm is a supervised learning classification algorithm. Which means the class of data has been defined or labeled. In this study, there is the attribute 'satisfaction'. Naive Bayes itself is a good algorithm. Because the formula used is easy and also has a high accuracy value. Broadly speaking, the Bayes theorem formula used is like the following function. x = attribute class/label y = attribute input In the Naive Bayes algorithm there ar e steps in implementing it. As an example of implementation, in this report I use 20 sample data from data that has been preprocessed. This data will also be used as an example implementation in this report for the LVQ algorithm. The data is as shown in the table below. Data Sample Naïve Bayes (20 data) The table data above will be used as an example of implementation in this chapter 4 report.

Column 1 is gender, 2 is customer type and so on as in table 4.1. The Naive Bayes steps used in this study are: Divide the dataset into training datasets and testing datasets. The distribution of the dataset is as shown in table 4.3. For example, I will take 20 sample data that has been preprocessed. Because this is the first test, 18 data are used as training and 2 data as testing. I separate manually by id. Training Dataset Naïve Bayes Testing Dataset Naïve Bayes Calculate P(x) for each class/label attribute variable ('satisfied' and 'neutral or dissatisfied') . The formula used to find it is as follows. x = Tota l data class/label from training data Total data = Total dat a from training data P(x) = Probability of variable class/labe l The class/label consists of data, there are 2 labels, namely 0 instead of "neutral or dissatisfied" and 1 substitute for "satisfied". Therefore, P (0) and P (1) will be calculated from the training data. Because the number of data that has the label 0 is 12 then: After that we also look for the value of P (1). Because the number of data that has the label 1 is 6 then: Calculate the probability of the input variable from each class/label or P(a|x). Calculate P(a|x) for all input attributes(gender, customer type, age, etc). The formula used is as follows for each attribute. a = clas

s input from testing data x = class/label Total data a x = Total data where class/label is x and input is a from training data Total data x = Total data where clas s/label is x from training data P(a|x) = Probability of a against x In the first testing data (id=1), the value of a is 0 for the gender class. Since the number of data with gender 0 and also label 0 is 6 and the number of data with label 0 is 12, then p(gender=0|label= 0) is: Next we also look for the probability value for gender 0 as well but with the label 1, the probability is: Because the customer type value in the first testing data is 1, then look for P (customer type = 1 | labe l=0). After that also calculate P (customer type =1 | label=1). Do the same for the age, type of travel and other attributes. Then we will get P(a|x) a number of input attributes, which is 22 P(a|x). Calculate the result of multiplying P(a|x) all attributes and P(x). Because the number of attributes is too many, the result of the multiplication of P(a|x) that I show as an example is only the gender and customer type attributes. Then the result of each class/label is: The biggest results are prediction results From the results of label 0 and label 1, it can be seen that label 0 has a greater distance value with a

value of 0.22445. Therefore, the prediction result is 0. To find the accuracy results later then if: Label class “1” a nd prediction results is “1”, TP added 1 Label class “ 0” and prediction result is “0”, TN added 1 Label clas s “1” and prediction results is “0”, FN added 1 Lab el class “0” and prediction results is “1”, FP added 1 In the testing data table, it can be seen that the class/ label from the first test (id=1) is 1. However, the prediction result from the calculation that has been done is 0. Therefore, we add 1 number of FN for this first Naive Bayes test. Repeat steps 3-6 for the second test(id=2) and so on until the last id of the testing data. After step 7 is complete then we calculate the accuracy. To calculate accuracy like Function 1 with input in step 6. Then the first test is done. Repeat steps 1-7 for the second to fifth Naïve Bayes test with the number of training data and datasets as specified. If it has been tested 5 times, then Naive Bayes has been completed in this study. After 5 times of testing Naive Bayes, next is the Learning Vector Quantization (LVQ) algorithm. LVQ is a classification algorithm like Naive Bayes which is supervised learning. The architecture of LVQ in this study looks like the following design. LVQ Architecture In the LVQ architecture there are layers, namely

input, process or competitive and finally output. In the input layer, there are 22 inputs, namely X1 to X22. Xn is the value of the input attribute, namely gender as the first attribute, customer type as the second attribute to the 22nd attribute. From 22 inputs it will be 2 in the competitive layer. This is because there are 2 class/labels, namely 'satisfied' and 'neutral or dissatisfied'. To make these two results, calculations are carried out using the Euclidean distance. The calculation is to find the input distance to each class/label. Euclidean distance formula like the following function. ||X-W|| = Euclidean distance Xn = Va lue from attribute n Wcn = Weight of class/label c an d attribute n After calculating the input to the weight of each class, we can get the prediction results. Prediction results on the output layer can be obtained by looking for a smaller value. However, if the values are the same, it can be determined which class will be entered. Here I specify enter the class "1" which is satisfied. As an example of implementation of LVQ, I use sample for dataset like Naïve Bayes. I use 20 sample datasets. The data is as shown in the table below. Data Sample LVQ (20 data) In doing this LVQ, the steps taken are as follows: Divide the dataset into training datasets and testing datasets. This

step is similar to step 1 of Naïve Bayes. Then the training and testing data will look like below. Training Dataset LVQ Testing Dataset LVQ Initialization The initial weight (W) is randomly or manual selected 1 input data training from each class. Because in this dataset there are 2 class/labels, namely "satisfied" and "neutral or dissatisfied" , then there are 2 initial weights. Weight for satisfied (Ws) and Weight for neutral or dissatisfied (Wns). For example, I manually select data from the training data with id 1 for Wns, because data where id 1 has class/label 0 or “neutral or dissatisfied” . Wns1 is the value of the first attribute, namely the gender attribute, so Wns1 is 1. Wns2 is from the second attribute, so Wns2 is 1 and then on to the 22nd attribute of the training data with id 1. In addition to Wns initialization, it also needs Ws initialization. For example, I manually select data from the training data with id 3 for Ws, because data with id 3 has class/label 1 or “satisfied”. Same as Wns initialization, Ws1 is 0. Ws2 is 1 and so on from training data with id 3. So that the initialization value of W is like the table below. Initial Weight The data used as weight initialization is not reused during the training process later. Therefore, training data with id 1 and 3 are not reused. To facilitate

programming, the data will be removed from the training data so that it is no longer possible to use it. Maximum Iterations (MaxEpoch). The maximum iteration that I set is 16. I set it that way because the amount of data from the training data is 18 data minus 2 for the initialization weight. Epoch. Epoch initialization is 1. Parameters learning rate/alpha (α). Alpha initialization is 0,9 Minimum error (Eps). Eps initialization is 0,0000001 Input Input XnX = input value n = attribute input to n The i nput value is taken from the input attribute values, namely gender, customer type, age and so on. So for the first iteration, the value of X is taken from the first training data. The first training data is data with id 2 because data with id 1 and 3 have been used as initial weights. So X1 is 1, X2 is 0, X3 is 0 and so on. Target = Class/label of data from testing data. From Input Xn above , the data used is training data with id 2. Therefore, Target is the class/label value of training data with id 2. So the value of target is 0. If Epoch < MaxEpoch or α > eps: Because the epoch with a value of 1 is less than the max epoch with a value of 16 and an alpha value of 0.9 more than the eps value of 0.0000001 then this provision is true. So it will run the steps below.

Find the input distance to each weight using ||X-W||. Then determine the minimum value as the prediction class (J). However, if the distance between the two weights is the same, the prediction class can be determined, whether it is "satisfied" or "neutral or dissatisfied" . I specify here as class satisfied. In this step we will look for weight satisfied (Ws) and weight neutral or not satisfied (Wns). For example, I only use the initial 3 attributes. Here it can be seen that the results ||X-Ws|| is 1,4142 and ||X-Wns|| is 1. Because the minimum value is 1 that is the result ||X-Wns|| then the prediction result (J) is 0 (neutral or dissatisfied). Update Wj for each Wn. If J= T then Wj` = Wj + α (X - Wj) If J ≠ T then Wj` = Wj – α (X - Wj) T=T arget Wj = Weight class j α = Learning ratio j = predi ction class X = data value Wn = Weigh index n The target (T) of the data with id 2 is 0 and J is also 0. So we will change W from prediction class to Wj` = Wj + ( X - Wj). So Wns1` = Wns1 + (X1-Wns1). Then Wns1 will cha nge to 1, Wns2 to 0.1 and so on until Wns22. Update the value of α. In updating the alpha value, I use the formula as in the function below α` = new learning ratio α = learning ratio MaxEpoch = Maximum Iteration α` = new lea rning ratio Then the value of the new alpha is 0.9 -

(0.9 * 0.0000001) which is 0.89999991. This new alpha value will be used as the alpha value for the next iteration.
If all training data has been processed, then epoch = epoc h +1. Then the epoch changes to 2. Repeat step 3 and 4 until condition 4 is false This step will repeat the steps until the condition Epoch < MaxEpoch or alpha > eps is false. The data used is training data. In this loop, the Ws and Wns values will continue to be updated until t he condition is false. If the condition is false (stopped) then the last Ws and Wns values will be used for th e weights on the testing data. After step 5 is complete, do step 3 but from testing data. After that looking for J like 4b. To find the accuracy results then if: T class "1" and J class results is "1", TP added 1 T class "0" and J class results is "0", TN added 1 T clas s "1" and J class results is "0", FN added 1 T cla ss "0" and J class results is "1", FP added 1 In th is step 6, we repeat step 3 which is to determine the value of X and the target. This value is obtained from the first testing data (id = 1). So X1=0, X2=1, X3= 1 and so on and T=1. After that we calculate ||X-Ws|| and ||X-Wns|| where Ws and Wns are the final results of step 5. Then we will get the predicted value of class(J)

by finding the minimum value between ||X-Ws|| and ||X-Wns||. After that we add the value of TP, TN, FN or FP according to the conditions. The addition of this value is the same as when the Naive Bayes algorithm. Repeat step 6 for all testing datasets In this step we repeat where the X and T data are testing data also for id = 2, id = 3 and so on until the last data from the testing data . After step 8 is complete, do steps 1-8 with the amount of training data and testing data as shown in table 4.3. Then find the accuracy value of all LVQ tests that have been carried out using function 1. By getting the accuracy value of each LVQ test, the LVQ algorithm is complete. Then the whole workflow process has also been completed. The accuracy results of the five Naive Bayes tests and the five LVQ tests were then compared. The accuracy of the Naive Bayes 1 test is compared to the accuracy of the LVQ 1 test, the accuracy of the 2 Naive Bayes test is compared to the 2 LVQ test and so on. The result of a better comparison is the sum of the better accuracy of each comparison. The results of each test will also be seen. Are the 1,2,3,4 and 5 Naive Bayes tests the accuracy results much different or almost the same. Similarly, the results of the 1,2,3,4 and 5 LVQ tests are the accuracy

results much different or almost the same. CHAPTER 5 IMPLEMENTATION AND RESULTS Implementation Lines 1-2 to load the downloaded file into the 'tbldata' table with the following conditions. The provisions are as in lines 3-5 based on the csv file format. Row 6 to ignore the first row because the first row is the column heading. Lines 7-17 so that the data in the empty csv (null), when entered into the database remains empty (null). Lines 18-19 are used to copy tbldata into tbldataprocess. This is so that tbldata has the exact same data as csv data. The program that will run later is taken from the tbldataprocess data. Lines 20-89 is a procedure that contains commands to perform preprocessing, namely removing null data and changing data. On lines 20-22 and 88-89 is the program code to create a procedure with the name of the procedure is preprocessing. Lines 23 and 24 to declare the variables that will be used in the procedure. Lines 26-31 are used to delete data that has a null value. Lines 31-33 so that the id attribute returns to order because there is an id that jumps after data is deleted. Lines 35-87 to change all data from tbldataprocess with the conditions as in table 4.2. Lines 90-92 and 201-202 are creating a procedure with a bayesian name that will perform the Naive Bayes algorithm.

This procedure has parameters to determine how many tests. Lines 93-103 to declare the variable used to store the result P(gender=1|label=a). Line 104 to store the values P(gender=1) and P(gender=0). While lines 106-116 for th e results P(gender=0|label=a). On lines 122-131 to determine the amount of distribution of training and testing data. Then it will be entered into tbldatatraining and tbldatatesting as much as the previous amount. The data entered comes from tbldataprocess. In lines 156 and 200-201 are repetitions for tbldatatesting which include steps 3-6 Naive Bayes. This iteration is the 7th step of Naive Bayes. In lines 158-169 calculate the probability of P(a|x) for both class/label x that is "satisfied" and "neutral or dissatisfied" from all input attributes. Lines 171-172 to get the result of multiplying P(a|x) all attributes and P(x) as in step 4 of Naive Bayes. Then 174-201 to get the prediction result and also add tp, tn, fn or fp to tblaccuracy. Here I also add tnull and fnull to see if there are any unpredictable tests. On lines 205-215 this is a procedure used to call a Bayesian procedure with parameters 1-5. These parameters indicate how many tests to determine the amount of training and testing data. Then after running the Naive Bayes algorithm 5 times, the accuracy value will be searched on

line 213. This line 216-235 is the code for creating the ed function. This function returns a Euclidian distance value from the given parameter. This function will be used during the LVQ algorithm. Lines 236-398 are the procedures in which the LVQ algorithm is executed. This procedure has 3 parameters, namely testing to how much, alpha and eps. The first parameter is used to determine the distribution of the amount of training and testing data. While alpha and eps are used to simplify the analysis by replacing the two values. Line 240 is a variable declaration for label weight 1 while 242 is for label weight 2. In line 244 is a variable for the value of the input weight. Next 245-255 is the variable declaration used in the lvq procedure. Lines 257-265 are for dividing the amount of training data and testing data. Furthermore, on lines 266-279 will enter the data from tbldataprocess into tbldatatraining and tbldatatesting the amount that has been obtained earlier. After dividing the data, then entering the data into tblaccuracy for testing that is being carried out on line 281. Then do the initialization step as in line 283-302. Initial weights for label 0 and label 1 are chosen randomly as in lines 287-288. Then enter into the variables for the 22 input attributes of the two classes as in lines 290-294. After

being stored in the data variable that has been used as the initial weight, it is deleted on lines 296-297. the id from tbldatatraining is updated again so that no id jumps because it has been deleted as in lines 299-301. In lines 305-351 do repetitions for the training process. At each repetition of the training process, the Euclidian distance value for labels 0 and 1 is searched using the ed function as in lines 306-311. After that, the prediction result (J) is determined by looking for the minimum value in lines 313-319. After that the weight value will be updated on lines 321-339. After updating the weights, the alpha value is also updated on line 341. In lines 343-349 it is used to make the loop condition false and exit the loop. After completing the training repetition, the final weight of the training is obtained. These weights will be used in the iteration of lines 352-377 which is the iteration for all testing data from tbldatatesting. In each of these iterations get the input values as in line s 353-356 for all input attributes. After that, look for the value of the Euclidian distance for the two labels as in lines 358-360. The minimum value of the two Euclidian distances is the prediction result. The value of tn, fp, fn or tp will be added by 1 if it is in accordance

with the provisions. After adding this value, it will then repeat for the next testing data until all the data is tested. The final results of tn, fp, fn and tp will be updated to tblaccuracy. In lines 382-392 this is a procedure to run the lvq procedure 5 times. After 5 tests with different amounts of training and testing data, the accuracy value will be calculated as in line 390. On lines 393-408 these are Naive Bayes and LVQ procedures. On line 396 run the procedure processb which will perform all 5 Naive Bayes tests. Next it will run the 5 LVQ tests by calling the processl procedure. In running the process, it is done several times with different alpha and eps values. Results From the results of the trials that have been carried out, Naive Bayes and LVQ can be used to determine airline passenger satisfaction. The results of the trial run for almost 4 days. The longest test run when running Naive Bayes is around 3 days and 3 hours. The results of the program that runs the Naive Bayes algorithm are as follows: Naive Bayes Results From the table above, Naive Bayes can determine quite a lot of predictions that match the original class. This can be seen from the accuracy obtained between the range of 88-89%. The average of this test is 89.076%. However, in tests 4 and 5 there are some testing data

whose prediction results are unknown. This is probably due to the large number of input attributes, namely 22 attributes, when multiplied all the results cannot be saved by the computer. Because the program can only store a maximum of 30 digits behind the comma. So it is possible that the result is 0.0 for both predictions and no conclusions can be drawn from the prediction results. Furthermore, the LVQ algorithm can also be implemented for this airline's passenger satisfaction data. Because there are alpha and eps parameters that do not have an exact value, this study tries to use several kinds of values. For the alpha value, try 4 variations of the value, namely 0.9; 0.1; 0.01; and 0.05. While for eps there are only 3 variations, namely 0.0000001; 0.0001; and 0.01. The results of the LVQ implementation of these variations are shown in the tables below. Results LVQ Alpha 0.9 and Eps 0.0000001, 0.0001, 0.01 Results LVQ Alpha 0.1 and Eps 0.0000001 Results LVQ Alpha 0.1 and Eps 0.0001 Results LVQ Alpha 0.1 and Eps 0.01 Results LVQ Alpha 0.01 and Eps 0.0000001 Results LVQ Alpha 0.01 and Eps 0.0001 Results LVQ Alpha 0.05 and Eps 0.0000001 Results LVQ Alpha 0.05 and Eps 0.0001 Of the 10 variations and each variation remains 5 times the results of the accuracy test also vary. In table 5.2 where alpha

0.9 and different eps the prediction results and accuracy have the same value even though the eps is changed. The average accuracy is 56.366%. But if you look at the Tp values, all are 0. This means that testing with an alpha of 0.9 will produce all "neutral or dissatisfied" predictions. Of course it cannot be said to be predicting the outcome because 100% of the predicted results are labeled "neutral or dissatisfied" . However, when alpha 0.1 results are quite good, this result can be said to be predicting the label. From the experiments that have been done, the best results are when alpha 0.1 and eps 0.01. The average accuracy is 79.39%. Because the average accuracy is the best, it will be used when compared to the Naive Bayes algorithm. So, if based on the accuracy value, Naive Bayes is better than LVQ where Naive Bayes has an average accuracy of 89.076% while LVQ is 79.39%. However, from the experimental results, the drawback of Naive Bayes is the processing time for the 5 tests, which is more than 3 days. Meanwhile, the LVQ for 5 tests with training and testing data differs by an average of 30 minutes. However, it is necessary to find the optimal value for LVQ through alpha and eps values. If the results of the accuracy of the LVQ algorithm in graphical form will be as below. Graph of accuracy LVQ alpha=0.9

Graph of accuracy LVQ alpha=0.1 and eps=0.0000001Graph of accuracy LVQ alpha=0.1 and eps=0.0001 Graph of accuracy alpha=0.1 and eps=0.01 Graph of accuracy alpha=0.01 and eps=0.0000001 Graph of accuracy alpha=0.01 and eps=0.0001 Graph of accuracy alpha=0.05 and eps=0.0000001 Graph of accuracy alpha=0.05 and eps=0.0001 When viewed from each graph of the accuracy of the LVQ algorithm, it has a different curve. This is of course because of the influence of the alpha and eps values. But it is possible because of the quality of the data itself. CHAPTER 6 CONCLUSION Based on the results of the tests that have been carried out, the following conclusions can be drawn: 1. Naive Bayes can be used to classify airline passenger satisfaction. It is proven that the Naive Bayes algorithm can be implemented for customer satisfaction data obtained through Kaggle.2. Learning Vector Quantization can also classify airline passenger satisfaction. This is because this algorithm can be implemented on the same data to implement Naive Bayes. 3. Of the two algorithms, Naive Bayes is better at classifying airline passenger satisfaction than Learning Vector Quantization. This is based on the average accuracy of the two algorithms. Naive Bayes has an average accuracy of 89.076% while the LVQ is 79.39%. Suggestions for further research is

to focus on one algorithm, namely Learning Vector Quantization (LVQ). With the same data, LVQ is implemented but with the aim of finding the optimal value. What is the most optimal combination of alpha and eps so that it can get the best acc

PLAGIARISM CHECK.ORG

# Sources

| | | |
|---|---|---|
| PLAGIARISM | 0.29% | |
| 1 deepai.org | 0.15% | |
| 2 eprints.covenantunive.. | 0.14% | |