
# *Διαχείριση Μεγάλων Δεδομένων*

November 26, 2023


- ΚΟΥΤΣΑΝΤΩΝΗ ΧΡΥΣΟΥΛΑ
AM:1088839
- ΜΠΙΝΙΕΡΗΣ ΑΝΔΡΕΑΣ
AM:1054711
- ΞΑΝΘΟΥΛΑ ΚΑΛΥΒΑ
AM:1069153

# Contents

# 1 Theme 1

In this assignment we answered some multiple choice questions and posted them in the lesson's site.

# 2 Theme 2

## 2.1 1st Paper

In this paper we are going to summarize the article titled "A naive Bayes strategy for classifying customer satisfaction: A study based on online reviews of hospitality services" by Manuel J. Sánchez-Franco, Antonio Navarro-García, Francisco Javier Rondán-Cataluña.

This paper discusses the significance of online reviews in the hospitality industry, particularly focusing on hotels. It highlights that most travelers rely on reviews, with 53% unwilling to book a hotel without them. The impact of review ratings on booking rates is explored, emphasizing the influence of electronic word of mouth (eWOM) in decision-making. The study analyzes a dataset of 47,172 reviews from 33 hotels, employing a machine learning approach to classify reviews as positive or negative. The role of infomediation systems in shaping consumer attitudes and the importance of user-generated content (UGC) are emphasized. The text underscores the important role of customer satisfaction in the success of hotels. The research aims to identify differences in the perceived importance of guest experience features and their impact on traveler satisfaction, addressing what makes a hotel good or bad and understanding travelers' perception to enhance their services.

Then on, methodology and results of a study based on data are presented, focusing on guests' preferences in the hotel industry, particularly in Las Vegas. The dataset, underwent thorough preprocessing, including text cleansing, tokenization, and feature selection based on tf-idf values. The study employed a two-step filter approach, utilizing a maximum relevance minimum redundancy (MRMR) method to identify significant features and discard the redundant.

The classification model, based on the naive Bayes algorithm, achieved a Matthews correlation coefficient (MCC) and metrics such as precision, recall, F1 score, Kappa, and area under the curve to assess the model's performance along with a network graph analysis that provides insights into the underlying semantic structure of hotel reviews. Overall, it is offered a comprehensive analysis of hotel reviews, feature selection, and community detection, showing on the factors influencing guest's perceptions and satisfaction.

The paper talks about the difficulty of interpeting the importance of reviews , emphasizing the effectiveness of naive Bayes in review analysis . Then the contribution of the study is mentioned in helping hotels allocate resources effectively based on what guest desire such as staff experience or better facilities and managers should consider those desires when making decisions. However there are some limitations , such as not examining infrequent terms and potential self-selection bias in guest reviews. Future research suggestions include exploring affective trust, assessing psychological sentiments in relationship continuation, and analyzing gender based differences in hotel experiences.

Concluding is presents the importance of the classification method it is used, based on the ability to identify effectively the polarity of guest's opinion and to offer insights for the hospitality industry.

## 2.2 2nd Paper

In this assignment, we are going to summarize the article titled "A hybrid approach for generating reputation based on opinions fusion and sentiment analysis" by Abdessamad Benlahbib & El Habib Nfaoui.

In their published work, Abdessamad Benlahbib and El Habib Nfaoui aim to introduce a hybrid approach to building reputations, leveraging opinion fusion and sentiment analysis. In the rapidly evolving landscape of the 21st century, the internet has become a platform where people can freely express their opinions about desired entities without constraints. These opinions serve as a valuable information source contributing to the establishment of a target entity's reputation, reflecting the preferences and behaviors of individuals. The majority of opinions expressed in natural languages carry

either a positive or negative sentiment towards a target, enabling their categorization for reputation calculation through a weighted numerical average.

The innovation presented by Abdessamad Benlahbib and El Habib Nfaoui is that they:

1. They initiate a classification process using two classifiers, Naïve Bayes and Linear Support Vector Machine (LSVM), to distinguish positive and negative opinions.

2. Positive and negative opinions are segregated into distinct sets based on semantic similarity.

3. An adapted reputation value is calculated separately for positive and negative groups, taking into consideration specific statistical elements of the foundational opinion sets.

4. The final reputation value towards the target entity is computed through a weighted numerical average.

To validate their approach, the researchers utilized 10 datasets, each comprising 100 reviews (comments and ratings) for 10 movies. Additionally, they incorporated the weighted average user rating on IMDb to enhance the representativeness of the datasets. Upon analyzing the data, it was determined that their proposed classification method outperforms Naïve Bayes and Linear Support Vector Machine (LSVM) in terms of accuracy, recall, and f-score for both positive and negative reviews.

While their method represents an improvement over previous approaches, certain challenges are acknowledged. These include the dependency of the classifier on the usage domain and language, and an increase in processing time resulting in decreased accuracy of reputation due to the presence of numerous irrelevant reviews. However, these challenges can be mitigated through the implementation of techniques such as machine translation, transfer learning, and the incorporation of a filtering phase.

### 2.3 3rd Paper

Now, we are going to summarize the article titled ""The Determinants of Bitcoin's Price: Utilization of GARCH and Machine Learning Approaches" by Ting-Hsuan Chen , Mu-Yen Chen2 , Guan-Ting Du.

In their research paper, Ting-Hsuan Chen, Mu-Yen Chen, and Guan-Ting Du investigate the key determinants influencing Bitcoin's price by employing both the Generalized Autoregressive Conditional Heteroskedastic (GARCH) model and a machine learning approach. Cryptocurrencies, such as Bitcoin, are digital forms of currency capable of conducting transactions online independently of third-party entities, governments, and traditional banks. Currently, Bitcoin holds the highest market value and transaction volume among all circulating virtual currencies.

The motivation behind their exploration of factors affecting Bitcoin's price arises from its notable fluctuations in exchange rates compared to conventional currencies, prompting questions about its viability as a store of value or investment. The GARCH model, a time series model predicting future value variance, and the Support Vector Machine (SVM), a machine learning method for prediction, are employed. SVM establishes an optimal decision boundary to maximize margins, ensuring a uniform data distribution.

A third methodology utilized by the researchers is the decision tree, a data mining technique applicable to classification prediction using patterns. Data for the study were sourced from bitcoinity.org, investing.com, Federal Reserve Economic Data (FRED), and the World Gold Council's Bitcoinity website. This dataset encompasses Bitcoin prices, stock indices, oil prices, exchange rates, interest rates, and gold prices.

Upon analyzing the data, the researchers discovered that USD-GBP, USD-CHF, euro-GBP, USD-euro, the Nikkei 225 index, and gold prices positively impact Bitcoin prices, whereas the Fed's capital interest and the FTSE 100 index exert negative influences. Notably, the decision tree analysis revealed that the Fed's capital interest has the most significant impact on Bitcoin price decisions, underscoring the influence of US monetary policy politicization.

### 2.4 4th Paper

At this point we are going to summarize the article titled "Recession Forecasting Using Bayesian Classification " by Troy Davig and Aaron Smalter Hall.

In their paper, Troy Davig and Aaron Smalter Hall apply Naïve Bayes classification as a tool for predicting recessions. What sets their approach apart is the utilization of a novel measurement method where the penalty for false signals varies based on their timing within an economic cycle. For instance, a false signal during the middle of an expansion incurs a greater penalty than one near a turning point. The onset of a recession is a highly significant macroeconomic event, and while predicting it poses a challenge, researchers are actively working to enhance existing methodologies.

In particular, the researchers leverage Bayes' theorem, incorporating a more comprehensive dataset, lag structure, and capturing the preservation of economic cycle phases by utilizing transition with Markov-switching probabilities.

### 2.5 5th Paper

In the article titled "Fake Review Detection using Naive Bayesian Classifier" by Ms. P. Kalaivani, V. Dinesh Raj, R. Madhavan, A. P. Naveen Kumar, the primary objective is to identify and eliminate fake reviews within a substantial collection of reviews utilizing machine learning models. The focus is on discerning inauthentic reviews written with the intention of manipulating the overall rating or reputation of a product, service, or business, a concept referred to as "Fake Review Detection."

The methodology adopted incorporates the application of the Support Vector Machine (SVM) algorithm for Classification and Regression Analysis, along with the utilization of the "Naive Bayes" algorithm for Classification. The novelty lies in employing these algorithms due to their precision, efficiency, and their capacity to handle both linear and non-linear data, as well as high-dimensional data.

For the classification and detection of fake reviews, the Support Vector Machine (SVM) and Naive Bayes algorithms are employed. The choice of these algorithms is grounded in considerations of accuracy, efficiency, and their ability to manage high-dimensional data and noise.

The dataset, sourced from the Kaggle website, comprises 20,000 reviews categorized by rating. Data cleansing is executed to eliminate unnecessary words, stop words, and punctuation. Moreover, the data undergoes transformation into vectors through vectorization for simplified use with algorithms.

The evaluation process employs a confusion matrix to visually represent the accuracy of the algorithm. Additionally, the "classification report" is utilized to furnish information on accuracy and the roster of test data.

Key findings highlight the superiority of the Naive Bayes algorithm over other classifiers, underscore the necessity of incorporating features like keyword frequency and review length, and emphasize the significance of evaluating accuracy in relation to user reviews for enhancing the model.

# 3 Theme 3

## 3.1 Part I

In this assignment, your goal is to conduct sentiment analysis on movie reviews using the IMDB-Dataset.csv file, employing the Naïve Bayes classification algorithm in the R programming language.

To achieve this, we start by analyzing sentiments in movie reviews using R and the Naive Bayes classification algorithm on the IMDBDataset.csv dataset.

Initially, we follow the data preprocessing procedure. Using libraries such as stringr, tm, and SnowballC, we remove special characters, convert text to lowercase, eliminate stopwords, and apply stemming to each word in the reviews.

Next, we create the Document-Term Matrix (DTM) using the tm library. The DTM is a matrix that associates each review with the words it contains and their respective frequencies. The data is then split into training and testing sets, We have chosen the first 80% of the text as the train set, while we have reserved the last 20% for the test set. This leads us to have 50.00175% negative reviews in the train set and the remaining 49.99825% positive reviews. For the test set, the corresponding percentages are 49.93% and 50.007%.

Finally, we implement the Naive Bayes algorithm. We make a prediction to see the accuracy. From the confusion matrix, we observe that it has not made any incorrect predictions in our test set. It correctly identified 4993 negative reviews as negative and 5007 positive reviews as positive.

Additionally, the accuracy is 84.84% with a 95% confidence interval from 0.8412 to 0.8554. With a statistical significance of 1%.

The statistical significance at 1% confirms that the outstanding accuracy of the algorithm is not merely by chance. This means that the performance of the algorithm is statistically significant at a confidence level of 99%.

| | **Prediction** | |
|---|---|---|
| **Reference** | **Negative** | **Positive** |
| **Negative** | 4289 | 812 |
| **Positive** | 704 | 4195 |

Table 1: Confusion Matrix

In summary, the accuracy of the classifier is a crucial metric for evaluating its performance in the real world.

## 3.2 Part II

Using the Naive Bayes classifier from Part I, we can address the following question: "How do user comments on social media affect the price of a specific product in the supermarket?" Sentiments are separated into positive and negative so we need to understand how each of them impacts pricing in supermarket's products.

First we need to determine which social media platforms are most relevant to our target audience and where discussion about the product and supermarket's pricing are possible to occur. It is wise here to pick between the most popular platforms since they will provide us with a greater representative number of data even though the preprocessing of the data will be a hard task(many irrelevant or random comments).

Second we need to gather our data. To achieve this, data collection is usually done through machines and, less frequently manually, through web scraping tools or Application Programming Interfaces (APIs). Afterwards the data preprocessing is carried out, involving the removal of erroneous or irrelevant data (comments) , handling missing data, standardizing the text (lowercasing, removing special characters, etc.)and a more general filtering process.

Next we can employ sentiment analysis to determine whether user comments as positive, negative, or neutral. We can (use pre-trained models or) train our own depending on the complexity and specificity of your analysis. In this case we can develop a scoring system to quantify sentiment. We

assign scores from 1 to 5(it could be stars in reviews) for extremely negative to extremely positive sentiments and then, after collecting historical pricing data for the product in the supermarket we correlate sentiment scores with pricing changes over time to identify potential patterns. Moreover we can conduct a regression analysis to model the relationship between sentiment and price changes more comprehensively. Also we can create visualizations, such as line charts or scatter plots, to illustrate the relationship between sentiment and price changes. This can help in communicating findings effectively.

To Conclude the classification model ,based on the naive Bayes algorithm ,along with the effectiveness and simplicity , provides valuable insights into the sentiment dynamics that may influence the pricing of a specific product in a supermarket.

# 4 Theme 4

## 4.1

In the content of this assignment, we download a Mushroom data set that contains characteristics of different mushrooms species, indicating whether they are edible or poisonous. It contains of 8123 observations and 23 variables , each one indicating a specific characteristic of a mushroom. Based on that data set we were asked to create a decision tree in the R environment that will be able to predict either a mushrrom is eatable or poisonous based on its characteristics. In order to build the tree we used the rpart package that we installed in our computer. Then we used a random sample of 80% of the data to create the training set after using the command set.seed(2), in order to take completelly random percantage of the observations that we need. The number of observations in training set is 6498 and the remaining 20% composes our test set with 1625 observations.

Afterwards we create the decision tree(Figure 1) based on the attributes of our dataset which is composed of nodes. Each node represents a decision based on a specific feature. Through the algorithm it is selected the best feature to split the data at each node after it has measures the entropy to determine the best split. The process of node splitting is recursive and continues up to the terminal nodes of the tree, the nodes leafs, where the final predictions are made. Each leaf node corresponds to a specific class, if the mushroom is eatable or poisonous.

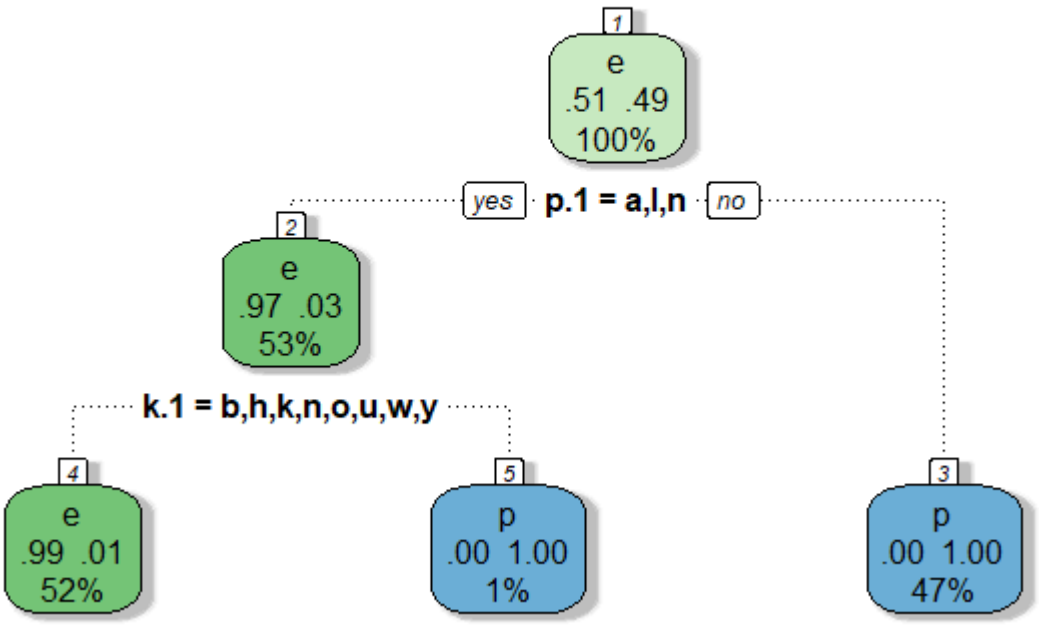


Figure 1: Decision tree for the dataset in R

Then on ,the tree is tested on the test set and after, through the confusion matrix, we can see the prediction accuracy of our decision tree. The prediction accuracy ,which is $accuracy = \frac{TP+TN}{TP+TN+FP+FN}$, was exceptional, reaching 99.26%, with an error rate of only 0.73%. Additionally, we printed the decision tree for visualization, displaying labels on the nodes.

| *tree_predict* | e | p |
|---|---|---|
| e | 866 | 12 |
| p | 0 | 747 |

Table 2: Confusion Matrix

### 4.2

Next, for the first 30 entries of the "habitat" attribute, we were asked to manually calculate the Entropy Gain. Firsty we calculate the entropy of the node(parent) using the foloowing formula:

$entropy = -p1log_2(p1) - p2log_2(p2)$ and its equal to 0.83 . Secondly we calculated the entropy of attribute(in our case "habitat") by finding the entropy of all the partitions(children). For "g" meaning grass, for "m" meaning meadows, for "u" meaning urban and "d" meaning woods. The entropy of attribute is equal to 0.53 so finally in order to find the entropy gain we subtract the entropy of node with entropy of split and the result it 0.30 .

### 4.3

Corresponding, in Python, we utilized the scikit-learn library to implement a decision tree . Through the one-hot encoding we converted the categorical variables into a pandas categorical type and then we created dummy variables. Continuing we created a new column 'newclass' by mapping the values in the 'class' column to numerical values (0 for 'p' and 1 for 'e'). Our data once again were splited into training and test data, the 80% composes the training and the 20% composes the test.

Proceeding , we built the decision tree (Figure2) using the entropy cretirion and fitted into our training set. After, we used the trained decision tree model to predict the class attribute on the testing set and calculated the confusion matrix (Table 3) using the predicted values and the actual values on the testing set. The accuracy was 100% something that indicates that both R's and Python's implementations provided reliable models for categorizing mushrooms based on their characteristics.

| *tree_predict* | e | p |
|---|---|---|
| e | 421 | 0 |
| p | 0 | 708 |

Table 3: confusion matrix

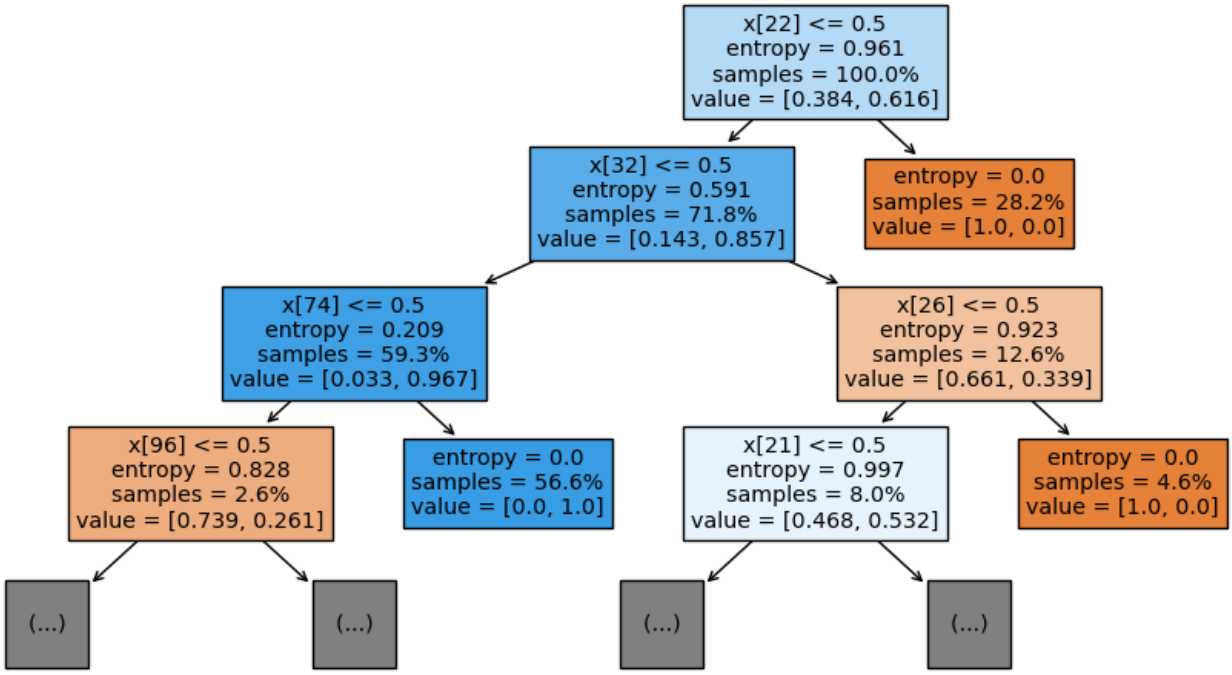


Figure 2: Decision tree for the dataset in PYTHON

# 5 Bibliography


- Mushroom. (1987). UCI Machine Learning Repository. https://doi.org/10.24432/C5959T.
- Feinerer. An introduction to text mining in R. R News, 8(2):19–22, Oct. 2008. URL http://CRAN.R-project. org/doc/Rnews/
- Feinerer, K. Hornik, and D. Meyer. Text mining infrastructure in R. Journal of Statistical Software, 25(5): 1–54, March 2008. ISSN 1548-7660. URL http://www.jstatsoft.org/v25/i05
- Gallo, Amy. "A refresher on regression analysis." Harvard Business Review 4 (2015).