Bachelor Kunstmatige Intelligentie

Thesis

7.5 ECTS

A Closer Look at Your Age: Determining Age Based on Eye Tracking Data

Author:
Merlijn van Uden
(6491081)

Supervisor:
Christoph Strauch

Second assessor:
Jakub Dotlacil

July 2, 2021

Universiteit Utrecht

Abstract

Classifying the age of a person based on eye tracking data is something that has been successfully done for younger participants with data collected in a supervised setup in a laboratory. This thesis tries to find out if it is also possible to classify peoples age using gaze data acquired in a public environment without supervision. To do this, multiple supervised machine learning algorithms are trained and tested on this data. Their performances are evaluated using the accuracy, precision, recall F1 score performance measures. Unfortunately, the performances of the classifiers are not great, mostly scoring an accuracy around the chance levels. However, the Support Vector Machine and the K-NN algorithm were able to achieve an accuracy of 20% above chance on multiple classes, which shows that they are able to find patterns in the data. These are promising results and with extra features extracted from the data or with more advanced classifiers it might be possible to achieve high accuracy age classification based on unsupervised, publicly acquired data.

Keywords: Machine learning, Eye-tracking, Gaze patterns, Age, Classification.

Contents

Chapter 1

Introduction

In 1901 Raymond Dodge developed the first non-invasive and precise eye tracker [1]. He found a way of using photographic plates to register the horizontal movements of the eye, consisting of fixations and saccades. During a fixation, the eyes are focused on a single location. A saccade is quick movement of the gaze going from one fixation point to another [2]. Since then, the field of eye tracking has evolved a lot. Currently, eye tracking data can be used for a wide range of tasks including classification. It is used to better train image classification algorithms [3], to classify diseases based on eye movement features [4] and even to determine the age of toddlers [5] or whether a person is an adult or a child [6]. A problem with these classification tasks is that the accuracy of the classification can be influenced by the quality of the data.

Where [5] and [6] made use of high-end eye trackers in a controlled testing environment, the data used to predict age in this thesis is acquired in a public room in a museum with an entry level tracker. It will be interesting to learn if it is possible to provide a fair estimation of age of the tracked beholders of an image given this data. Multiple machine learning algorithms will be considered for this task because they are able to handle high dimensional problems, discover formerly unknown knowledge and identify implicit patterns in data sets [7]. If it is indeed possible to predict the age of a person based on this easily acquired data, it would show that the way we look at images changes while our brains develop. But besides that, it would also tell us that the differences are easy to find, do not need an advanced laboratory to be noticed and that short periods of tracking might be sufficient which allows for applications in research and in industry.

The question that will be addressed in this thesis is: "How well can age be classified based on a data set obtained without supervision, entry level equipment and only 10s of data per participant?". To answer this question, the following sub questions will be addressed first: "What classifiers will result in the highest accuracy give the data that I have" and "what age bins will result in the best results?". The found algorithms will be implemented using the chosen age groups. We will then be able to use the results from these algorithms to answer the research question.

The thesis will continue as follows. Chapter 2 will highlight the data and the data processing. The 3rd chapter highlights different machine learning algorithms together with their applicability to the current data. In the 4th chapter the best fitting algorithms from chapter 3 are implemented and ran on the data. Chapter 5 shows their results. Findings and possible limitations are discussed in chapter 6 with chapter 7 concluding the here presented efforts.

Chapter 2

Data

2.1 The setup

The data used in this thesis was collected using a Tobii 4c eye-tracker, located at the NEMO Science Museum in Amsterdam. It was a brightness-controlled setup in a metal box, where a screen and eye tracker were located 80 centimetres from eye position. The image shown to the participants can be seen in Figure 2.1. Participants could have their eyes tracked in the setup in order for their gaze path to be visualized afterwards. All data was collected during the year 2020, with the intention of it to be used for different types of research.

Figure 2.1: The image that was presented to the participants.

2.2 Interesting properties of data

Most eye tracking datasets are small. The best eye tracking equipment is expensive and recording the movement of the eyes of a participant is usually done in a laboratory under the supervision of an experimenter. This means they can only process one participant at a time, resulting in the small data sets. These data sets are however of good quality because the high-end tracker is well calibrated and there is a lot of data available per participant. The data set in this thesis is large (N = 1242). This is because the installation was in a public place and everybody who was at the museum could have their gaze tracked. Most data sets also have a limited range in demographics. This is also caused by the time it takes to measure a participants gaze movements, but also because the research for which the data is collected is focused on a certain group of people. The eye tracker at the

NEMO museum was accessible to everybody so it collected the eye tracking data of people of all ages between 11 and 59 (see Figure 2.2). But given the relatively uncontrolled setup, the use of lower end equipment and short periods of data assessments per participants it is yet unclear how useful the data can be for classification.

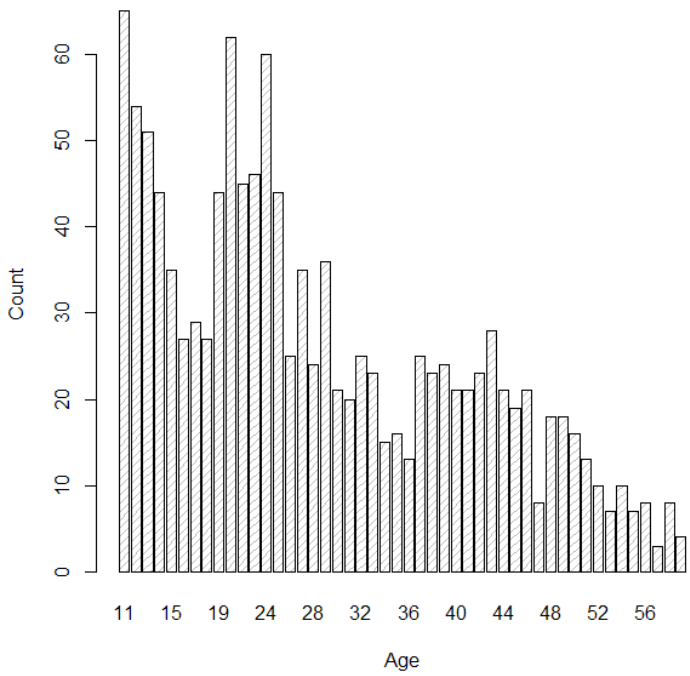


Figure 2.2: The age distribution of the participants.

2.3 The data

There are 2 distinct types of features in the data. There are 600 coordinates per person, consisting of an X and Y coordinate. These make up the path which the eye followed during the 10 seconds it was measured. This coordinate data has been analysed by the I2MC algorithm, which extracts the number of fixations, the mean fixation length, the mean fixation duration, the standard deviation of the fixation duration and the total fixation length from the coordinate data and has been purposefully built for noisy data [8].

2.4 Filtering

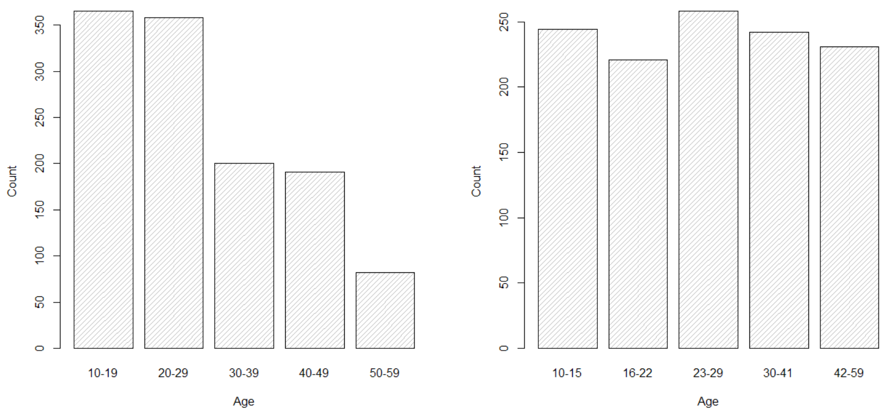
Because of the entry-level eye tracker, some filtering had to be applied to make the data more usable. One of the problems was that the eye tracker registered people looking outside of the picture. Where the resolution of the picture is 1920 x 1080, the x coordinates ranged from 2239.5 to -240.2 and the y coordinates from 1297 to -290.6. All participants for whom a coordinate was recorded which deviated more than 200 points from the edge of the image were removed from the data set, as it might be indicative of gaze estimation errors. The I2MC algorithm had an issue where it returned 0 on all fixture features. Because those values would influence the algorithms sensitive to outliers too much, they were also removed.

2.5 Age bins

Predicting the exact age of a person is not feasible, as there would be too many labels and the difference between labels would be too small. Therefore, the ages were put into age bins. Because of the large range of ages, two different binning approaches will be used. The data entries will be put into 5 bins as done in [6] to see if the classifiers can find distinct patters for multiple classes.

The first grouping is in equal age bins, where the different ages are divided into groups of 10 years each, so 10-19, 20-29, 30-39,40-49 and 50-59 (see Figure 2.3a). This however results in bins with an unequal number of cases, which can be an issue with some of the classification algorithms.

To overcome this issue, equal size bins will be implemented. It makes sense to split the younger age bins up into smaller ones because, as depicted in Figure 2.2, there are more younger participants and young kids make shorter saccades and longer fixations [9] which is something the classification algorithms might use. As the brain stops developing around the age of 25 [10] it also makes sense to group the higher age classes together. This resulted in the following bins: 10-15, 16-22, 23-29, 30-41, 42-59 (see Figure 2.3b). It is not possible to achieve completely equal bins due to the participants already being put into a certain age.



(a) Equal age bins (b) Equal size bins

Figure 2.3: The two different binning approaches used to group the data entries.

Chapter 3

Algorithms

3.1 Machine Learning Algorithms

The problem dealt with is the classification of the age of participants based on their recorded gaze patterns. As said before, machine learning algorithms will be used for this classification task since they work well with high dimensional data and because of their ability to find hidden patterns [7]. The first problem which has to be overcome is choosing the type of machine learning algorithm. There are supervised learning algorithms, unsupervised learning algorithms and reinforcement learning algorithms.

A supervised learning algorithm is given a data set with which consists of independent variables together with a dependent variable or a label. From this data it has to learn a pattern, so that when it is handed a data sample without a label, it can predict its label based on what it has learned from the labelled data [11]. An unsupervised learning algorithm is not given labelled data. It has to find the groups in the data by itself or find patterns which were previously undetected. When it is given new data, it may recognize patterns or features from data it has previously seen and is able to classify it. It can also be used to label unlabelled data which will then be used to train a supervised algorithm [11]. Reinforcement learning algorithms make guesses or try certain actions and gets rewarded based on how accurate or good that guess or action is. This way it gradually learns what the right guesses or actions are [12].

Although Reinforcement learning algorithms can be used for classification [13], it is mainly used to train agents into making decisions in a certain environment. Its application to our data would require unnecessary extra steps such as specifying the actions which the algorithm can perform and defining a reward function [13]. Unsupervised learning is more fit for the classification task, but because the labels of all the data points are known, it would perform an extra unnecessary step which would only take up time. Supervised learning however does use labelled data in the training step, so it is the most applicable type to the data used in this thesis.

The next problem will be choosing the specific reinforcement learning algorithms which might produce good results. This will be done by looking at five of the most common ones and looking at their applicability to the data: SVM, Logistic Regresion, K-NN, Decision Tree and Naïve Bayes. These specific ones are looked at because of their relatively easy implementation in comparison to algorithms like neural networks. Implementing them is not feasible for the time span of this thesis.

3.1.1 SVM

A support-vector machine, or SVM in short, tries to separate the data into the right classes by trying to find a hyperplane or multiple hyperplanes which separate the data based on their class label. It is a robust tool which does not overfit, but choosing the right kernel and parameters is critical [14].

3.1.2 Logistic Regression

Logistic regression tries to model a logistic predictor function on the training data. It will return a probability between 0 and 1 for each class. It is quick to implement and makes no assumptions on the class distribution, but the independent variables should not be highly correlated because this can cause problems with the estimation [15]. It also cannot find complex nonlinear relationships [16] which might pose a problem as it is unlikely that the data used in this thesis is linearly separable.

3.1.3 K-NN

K-nearest neighbours is a classification algorithm which does not really learn. It is given a training data set with labels, and when it is presented a data point without label it assigns the most common label from the k nearest points from the training set. It is easy to implement because there are just two main parameters: the value of k and the type of distance function. However, classifying a point can take a long time when working with a large data set and data sets with high dimensionality which makes choosing the parameter k difficult [17].

3.1.4 Decision Tree

A decision tree does learn from the training data is it given. It finds the property which differs the most between the classes and selects it as the first selection criteria. It then chooses the next property based on the same criteria [18]. It keeps going until it has reached the set maximum depth or used all features. It is able to break a complex decision-making problem down into simpler decisions resulting in easy to interpret solution [18]. Unfortunately, it tends to overfit on the training data and can take a long time to train [19].

3.1.5 Naïve Bayes

A Naïve bayes classifier calculates the possibility of a datapoint belonging to a certain class based on applying Bayes' theorem. It is easy and fast to predict the class of a data point and also works on multi class data [20]. However, a strong independence between the features is assumed which is almost never the case [21].

3.2 Chosen Algorithms

Naïve bayes is not suited for our data because the features of the data consist of coordinates and fixation information about those coordinates. They influence each other so an independence between them cannot be assumed. Because the other classifiers look like they might be applicable to the data and each have their own strengths and weaknesses, it is useful to implement them and look at their results.

Chapter 4

Methods

4.1 Implementation

After the literature research the found algorithms were implemented. They were all implemented in the programming language Python, using the scikit-learn library [22]. Python was chosen in combination with this library because it offers an easy way to use a broad spectrum of machine learning algorithms. The specific implementations of the algorithms can be found below. Undiscussed parameters were not specified so their default values were used.

For the SVM, C-Support Vector Classification was chosen because it works on multiclass data and it allows for the regularization parameter (C) to be set. The Radial Basis Function kernel was used because it is able to find nonlinear hyperplanes to separate the data [22]. The C value was set to 10 which made sure the decision surface was not too smooth where it misclassified a lot of data points in the test set, but it also makes sure it did not overfit on the test data by aiming to classify all points correctly. The class weight was set to balanced, to make sure the classifier did not label everything as the biggest class.

Logistic regression was run with the Limited-memory Broyden-Fletcher-Goldfarb-Shanno solver because it works on multiclass problems and is relatively quick compared to the other multiclass solvers available for logistic regression. It was set to classify multi class data and also here the class weight was set to balanced to make the classifier also work correctly on the equal age bins.

For the K-nearest neighbour algorithm the value for k was set to 10. This made sure the algorithm looked at sufficient neighbours to determine the class label. It was not too big which would make the classifier biased against the bigger classes, because those would have a greater chance in being part of the neighbourhood.

The decision tree algorithm was implemented with a max depth of 10, to make sure the tree would not be overfit on the training data.

After the algorithms were implemented they were run 4 times. Once on the coordinate data in combination with the equal age bins, once on the coordinate data with the equal size, once on the fixation data in combination with the equal age bins and once on the coordinate data in combination with the equal size bins.

4.2 Evaluation

After implementing the algorithms, they had to be evaluated. This was done with the following performance measures: accuracy, precision, recall and the f1 score.

Accuracy is the most straightforward. It is the number of correct predictions divided by the total number of predictions and shows what percentage of the data points is classified correctly.

$$accuracy = \frac{TP + FP}{TP + TN + FP + FN}$$

Precision is calculated per class label, and represents the ability of the classifier to not label a data point this label when it should be another one. It is defined by the number of true positives divided by the number of true and false positives:

$$precision = \frac{TP}{TP + FP}$$

Recall is a slight variation on precision. It stands for how many of the data points with a certain label it classified as this label and is calculated by dividing the number of true positives by the number of true positives and false negatives:

$$recall = \frac{TP}{TP + FN}$$

The F1 score is the harmonic mean of the precision and recall, which is defined as:

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

Chapter 5

Results

	SVM	LR	KNN	DT
Accuracy	0.342	0.220	0.304	0.288
Precision	0.324	0.243	0.262	0.269
Recall	0.342	0.220	0.304	0.288
F1	0.329	0.225	0.270	0.273

(a) Coordinate data in combination with equal age bins.

	SVM	LR	KNN	DT
Accuracy	0.244	0.217	0.289	0.234
Precision	0.294	0.223	0.247	0.248
Recall	0.244	0.217	0.289	0.234
F1	0.189	0.196	0.241	0.230

(b) Fixation data in combination with equal age bins.

	SVM	LR	KNN	DT
Accuracy	0.248	0.170	0.213	0.204
Precision	0.246	0.169	0.217	0.201
Recall	0.248	0.170	0.213	0.204
F1	0.244	0.169	0.213	0.194

(c) Coordinate data in combination with equal size bins.

	SVM	LR	KNN	DT
Accuracy	0.201	0.195	0.229	0.209
Precision	0.230	0.209	0.239	0.189
Recall	0.201	0.195	0.229	0.209
F1	0.140	0.188	0.221	0.174

(d) Fixation data in combination with equal size bins.

Table 5.1: Accuracy, Precision, Recall, and F1 Score per classifier. Results are split into different tabels according to the used data and bins.

The best scores were achieved by the SVM on the coordinate data with the equal age bins, where the K-NN algorithm also performed above average (Table 5.1a). On all other combinations between the data and the bins, the scores were around 0.2 which is the score expected when the classifier labels the data by chance.

	10-19	20-29	30-39	40-49	50-59
Precision	0.379	0.449	0.250	0.212	0.000
Recall	0.500	0.443	0.200	0.167	0.000
F1	0.431	0.446	0.222	0.187	0.000

Table 5.2: In-depth performance results for the SVM per age class on the coordinate data in combination with equal age bins.

	10-19	20-29	30-39	40-49	50-59
Precision	0.316	0.365	0.0.152	0.186	0.000
Recall	0.469	0.461	0.135	0.060	0.000
F1	0.377	0.407	0.143	0.091	0.000

Table 5.3: In-depth performance results for the K-NN algorithm per age class on the coordinate data in combination with equal age bins.

The in-depth results for the SVM and K-NN algorithm on the coordinate data together with the equal age bins show how well the classifiers worked on each class. The best scores were achieved on the 10-19 and 20-29 classes, and they both did not label any data entry as 50-59.

Chapter 6

Discussion

The results of this study indicate that classification of age based on easily acquired data is not easy, and that the used algorithms are not able to complete this task with a high accuracy. The support vector machine and the k-nearest neighbour were however able to achieve an accuracy above 30% which is 10% above chance. The SVM achieved this accuracy on the coordinate dataset in combination with the equal age bins. It has an F1-score of around 45% for the classes 10-19 and 20-29. As depicted in Table 5.2 this F1-score is achieved because roughly 45% of the data points which were given said label were really that label (precision) and roughly 45% of the data points which should have been assigned that label were actually given that label (recall). This means it is able to separate a significant part of the data points belonging to the 10-19 and 20-29 class. For the K-Nearest Neighbours algorithm the same two classes had the best results, but with a lower F1-score of around 40%.

It is worth mentioning that the best results were found for the two biggest classes of the equal age bins, and the fact that there were more training samples available for those classes must have helped achieve the higher accuracies [23]. That these algorithms did not achieve the same above average results on any classes when using the equal size bins on the coordinate date reinforces this. However, the class weights were balanced for the classifiers so the impact should not be that big.

The Logistic regression classifier achieved low scores on all the different data configurations, only scoring slightly above or below chance. This is a good indicator that the data is not linearly separable. As logistic regression is used in statistics and considered a statistical algorithm [24], it also shows that proper machine learning algorithms are needed to find the complex patterns to classify the data.

Where this thesis tried to implement multiple different classification algorithms, papers with a similar goal ([5], [6]) just went with the one classifier, sometimes combined with a different preprocessing algorithm. This made sure they got the maximum accuracy possible for the chosen classifier. The limited time and attention available for this thesis had to be divided over multiple classifying algorithms which might have badly influenced the accuracy of the algorithms.

[5] and [6] achieved the best results in classifying people of younger ages. [5] classified toddlers of 2 different age groups, and the classes in [6] were 2 y.o., 4-6 y.o., 6-8 y.o., 8-10 y.o. and adults. What they have in common is that they both classify participants who are young and whose brains are still in development [10], where [6] also has an additional class for adults. The best accuracy achieved in this thesis is for the classes made up of the younger age groups, so the low over-

all accuracy might have something to do with the fact that it is hard to classify adults. It is possible that as their brains are fully developed [10] their gaze patterns do not differ in a significant way.

Another difference with [6] is the choice of algorithms. [6] used a neural network with four convolutional layers, two pooling layers and two fully connected layers to extract features form the gaze path and classify the data accordingly. As creating such a complicated neural network is a time intensive task, implementing one for the classification task of this thesis was not achievable within the given time frame. Doing so might have resulted in better accuracy on this data, which is an interesting topic for further research.

What might have improved the accuracy was the access to more information about the gazing patterns of the participants. Metrics like AOI, where fixations are linked to specific parts of an image [2], could give us extra information about what the participants were looking at. Because the beheld image contains a lot of objects, information about what objects where mainly viewed per participant could give us the insight needed to also classify the older participants with higher accuracy. This is also interesting for future research

Chapter 7

Conclusion

This thesis aimed to answer the question "How well can age be classified based on a dataset obtained in a public space without supervision using entry level equipment?" This was done by implementing 4 different supervised learning algorithms who were run twice on the coordinate data and twice on fixation data, with a different one of the two age bins each time.
It was found that the used algorithms were not able to classify the participants well, which other papers were able to do ([5], [6]). The low accuracy could be caused by multiple factors, like the quality of the data, the used algorithms, or the chosen age bins. Although the accuracies achieved in this thesis were not on par with other research, two machine learning algorithms did show promising results. The SVM and the K-NN algorithm were able to find 20% above chance accuracy for multiple classes, indicating that there were patterns in the data which the algorithms recognized. If more features about the data were available, like the AOI metric, or if a neural network were implemented, the accuracy might have been higher so it is not yet certain if this classification task cannot be achieved with good accuracy.

Machine learning algorithms which are relatively easy to implement were used in this thesis. The fact that some of them are able to extract usable information from the data used in this thesis is good news. This means that in the field of AI, machine learning algorithms have the potential to be applied to data acquired in a public place without supervision. It shows that these algorithms are applicable to data which is not acquired in a lab under perfect conditions, which makes machine learning relevant for a broader amount of applications and people.

References

[1] Monika Płużyczka. "The first hundred years: A history of eye tracking as a research method". In: *Applied Linguistics Papers* 25/4 (2018), pp. 101–116.

[2] Dario D Salvucci and Joseph H Goldberg. "Identifying fixations and saccades in eye-tracking protocols". In: *Proceedings of the 2000 symposium on Eye tracking research & applications*. 2000, pp. 71–78.

[3] Xuan Zhou et al. "Eye tracking data guided feature selection for image classification". In: *Pattern Recognition* 63 (2017), pp. 56–70.

[4] Yuxing Mao et al. "Disease classification based on eye movement features with decision tree and random forest". In: *Frontiers in Neuroscience* 14 (2020), p. 798.

[5] Kirsten A Dalrymple et al. "Machine learning accurately classifies age of toddlers based on eye tracking". In: *Scientific reports* 9.1 (2019), pp. 1–10.

[6] A Tianyi Zhang and B Olivier Le Meur. "How old do you look? Inferring your age from your gaze". In: *2018 25th IEEE International Conference on Image Processing (ICIP)*. IEEE. 2018, pp. 2660–2664.

[7] Thorsten Wuest et al. "Machine learning in manufacturing: advantages, challenges, and applications". In: *Production & Manufacturing Research* 4.1 (2016), pp. 23–45.

[8] Roy S Hessels et al. "Noise-robust fixation detection in eye movement data: Identification by two-means clustering (I2MC)". In: *Behavior research methods* 49.5 (2017), pp. 1802–1823.

[9] Andrea Helo et al. "The maturation of eye movement behavior: Scene viewing characteristics in children and adults". In: *Vision research* 103 (2014), pp. 83–91.

[10] Andrey P Anokhin et al. "Age increases brain complexity". In: *Electroencephalography and clinical Neurophysiology* 99.1 (1996), pp. 63–68.

[11] Sotiris B Kotsiantis, I Zaharakis, P Pintelas, et al. "Supervised machine learning: A review of classification techniques". In: *Emerging artificial intelligence applications in computer engineering* 160.1 (2007), pp. 3–24.

[12] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. "Reinforcement learning: A survey". In: *Journal of artificial intelligence research* 4 (1996), pp. 237–285.

[13] Marco A Wiering et al. "Reinforcement learning algorithms for solving classification problems". In: *2011 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*. IEEE. 2011, pp. 91–96.

[14] Himani Bhavsar and Mahesh H Panchal. "A review on support vector machine for data classification". In: *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)* 1.10 (2012), pp. 185–189.

[15] Viv Bewick, Liz Cheek, and Jonathan Ball. “Statistics review 14: Logistic regression”. In: *Critical care* 9.1 (2005), pp. 1–7.

[16] Jack V Tu. “Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes”. In: *Journal of clinical epidemiology* 49.11 (1996), pp. 1225–1231.

[17] Kashvi Taunk et al. “A brief review of nearest neighbor algorithm for learning and classification”. In: *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*. IEEE. 2019, pp. 1255–1260.

[18] S Rasoul Safavian and David Landgrebe. “A survey of decision tree classifier methodology”. In: *IEEE transactions on systems, man, and cybernetics* 21.3 (1991), pp. 660–674.

[19] Oleksandr Sosnovshchenko and Oleksandr Baiev. *Machine Learning with Swift: Artificial Intelligence for IOS*. Packt Publishing Ltd, 2018.

[20] Jason DM Rennie. “Improving multi-class text classification with naive Bayes”. In: (2001).

[21] Zhongheng Zhang. “Naıve Bayes classification in R”. In: *Annals of translational medicine* 4.12 (2016).

[22] Fabian Pedregosa et al. “Scikit-learn: Machine learning in Python. the Journal of machine Learning research 12”. In: (2011).

[23] Sarah J Dixon et al. “Application of classification methods when group sizes are unequal by incorporation of prior probabilities to three common approaches: Application to simulations and mouse urinary chemosignals”. In: *Chemometrics and Intelligent Laboratory Systems* 99.2 (2009), pp. 111–120.

[24] Chao-Ying Joanne Peng and Tak-Shing Harry So. “Logistic regression analysis and reporting: A primer”. In: *Understanding Statistics: Statistical Issues in Psychology, Education, and the Social Sciences* 1.1 (2002), pp. 31–70.