Cognitive Workload Detection and Classification

Submitted as Research Report in SIT723

SUBMISSION DATE
T3-2022

Andrew Hallam
STUDENT ID 220477645
COURSE - Master of Data Science (S777)

Word Count: 7,958

Supervised by: Dr. Atul Sajjanhar, Dr. Glory Lee

Abstract

Cognitive workload is an increasingly studied topic which encapsulates a broad spectrum of fields, including health, psychology [1][2] and defense applications [3][4]. One such method of discerning potential levels of cognitive workload is through the use of eye tracking [5][4]. Previously, much of this has been tracked manually by researchers, but with the rise of computer vision and machine learning has come the potential of automated end to end methods of categorization [3]. Ensemble modeling is considered as one of the state of the art approaches to machine learning problems for improving models' efficiency and accuracy [6]. Until now, the most common ensemble model used in this domain is a random forest or gradient boosted decision tree[7]. Benchmark models including support vector machine (SVM), decision tree and Gaussian Naïve Bayes (GNB) within the domain of eye movement and cognitive load have been identified and analysed through multiple evaluation metrics. Furthermore, I have built an ensemble model using the strongest predicting models based on their strong predictive results[8][9][10]. The best performing hyperparameters were analysed and adopted. The machine learning framework in this study aimed to fill a gap in the literature by combining multiple, high performing machine learning models to create a novel ensemble model. I hypothesised that the proposed machine learning framework would strengthen the capability of the current machine learning models through the use of feature extraction and ensemble learning. The proposed model was evaluated based on a comparison of performance to previous models. I found that the stacked models did not significantly improve upon the single models, the individual models did not perform as well as those presented by the dataset authors, however almost all of the models performed above 0.5 accuracy, indicating that they performed above chance. Suggestions for improvements and future directions are discussed.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Aim & Objectives

This research aims to examine the prediction of cognitive workload through eye movement data. It aims to do this through the use of multiple publicly available datasets and multiple machine learning models. It aims to fill a gap in the literature of limited research on prediction through ensemble modelling by proposing a stacking ensemble model.

1.2 Structure

Section 2 reviews the literature. Section 3 presents the research design and methodology. Section 4 describes the approach and the technical details of artefact development. Section 5 evaluates the artefacts, on the basis of research questions (RQs) in Section 5.1 and discusses the RQs in Section 6. Section 7 discusses threats to validity and Section 8 concludes the report.

1.3 Background

Cognitive workload is an increasingly studied topic which encapsulates a broad spectrum of fields[1][2][3][4]. Previously, much of this has been tracked manually by researchers, but with the rise of computer vision and machine learning has come the potential of automated end to end methods of categorization [3].

Below is a comprehensive analysis of recent literature. Firstly, search methodology will be provided, followed by key features, proposed datasets and machine learning models, and papers that have used these datasets and models to allow for a baseline, and finally the proposed direction for filling a gap in the literature and associated research questions.

2 Literature Review

2.1 Literature search method

This literature review was conducted using Google Scholar, SCOPUS and IEEE Explore. Current literature published between 2019 and 2022 were included. A total of 21 papers were reviewed. 15 were excluded as they were not relevant enough to the research topic. Relevance, citations, publisher, publication date and content were all considered when including or excluding a paper.

The following search terms were initially included: eye movement cognitive workload machine learning, eye tracking data set, eye movement data set, machine learning cognitive workload, eye movement machine learning. Further search terms were amended and updated as reading continued, for example pupil size cognitive workload machine learning, blink frequency cognitive workload machine learning, fixation cognitive workload machine learning.

2.2 Features

The following features have been found to be beneficial in machine learning models used to predict cognitive workload in recent literatures. Each of these will be reviewed.

2.2.1 Pupil Size

Shojaeizadeh, Djamasbi, Paffenroth Trapp [11] sought to predict task demand through eye movement data. One of the key findings from their study was the importance of pupil size data in predicting task demand, with their model built using saccade-to-fixation pupil dilation and pupil dilation variance ratio, with an accuracy of 79%, using a random forest model.

This is not the only study examining this relationship, and further researchers have corroborated this relationship, including Joseph and Murugesh [2], Jerčić, Sennersten and Lindley [12] and Ramakrishan, Balasingam and Biondi [13].

Ramakrishan et al.[13] simulated 4 different experiments to test the relationship between multiple eye movement measures including pupil size and cognitive workload. These tasks were unmanned vehicle operation, memory recall tasks, delayed memory recall tasks, and finally simulated driving. They found that in all but one task, pupil size increased, on average, as the task difficulty increased. The one task where this was not the case was due to a confounding stimulus of light. In summary, they found that while pupil size was a strong indicator of cognitive workload, that pupil size alone is not enough due to the affects of other stimuli, such as light. As this research will be using a publicly available dataset, external stimuli are not something that can be controlled for as it is limited to existing datasets.

2.2.2 Blinks

Another physical indicator that has been linked to cognitive workload is blink frequency. Multiple researchers have provided evidence to show this link, including Shojaeizadeh et al. [11], and Bafna, Hansen and Baekgaard [14].

Bafna, et al. [14] sought to measure the link between cognitive workload and blink frequency during a typing task. They had 18 participants memorize simple and difficult sentences over a period of four days. Using blink frequency, pupil diameter frequency and interval, typing speed and error rate as features in their prediction they found that as the task performance increased, typing performance lowered; and blink frequency, duration and interval increased. This provides support for this concept that blink frequency can be a useful predictor when determining cognitive workload.

2.2.3 Fixation

The final feature that will be discussed in this literature review is fixations. This is another physical indicator that has been identified as integral to predicting cognitive workload in recent literature [15] [16]. Fixation is the point that a gaze is directed to. The duration is the length of time that a person's gaze is fixated at a specific point.

Liu, Li, Yeh and Chien [17] narrowed in on fixation as a predictor for cognitive workload. They obtained fixation data and level of cognitive workload using a modified video game. They changed the number of tiles and words presented in this game to simulate low and high levels of cognitive workload. The measure of cognitive workload was obtained using the NASA TLX score. This is the same measure used by by Ktistakis, Skaramagkas, Manousos, Tachos, Tripoliti, Fotiadis and Tsiknakis [18], which will be examined later.

They used one way ANOVA to analyse the relationship between fixations and cognitive workload. Their findings were that in high cognitive workload tasks, participants had longer fixation durations and less fixations. This study is important to note as the same cognitive measure is used by Ktistakis et al.[18] on the proposed COLET dataset. It is therefore possible that similar results are obtained when using machine learning models on this dataset.

2.3 Data sets

This section will firstly be prefaced by noting that there are very limited, appropriate, open data sources available to use in this study. Of all the studies examined, only 5 contained open data sources. At least one contained a link to repositories that were no longer supported, and one contained the note that it would be supplied upon request, however due to time restrictions this is not feasible to obtain. As a result, 2 open-source data sources were included in this research and are described below. He et al. [8] used a specific dataset available to physiotherapists, however this is not available to use for this research. Bozkir et al. [9] used a private dataset they compiled in their previous work using a virtual reality driving simulation. This was done specifically to examine the effect of a virtual or augmented reality setup.

2.3.1 COLET

The Colet dataset was created by Ktistakis et al. [18]. The authors sought to create a publicly available dataset that combined both objective and subjective measures of cognitive workload. They noted these levels of cognitive workload as low, medium and high.

28 participants were recruited and performed 4 activities. These activities consisted of visual search activities of variable complexity. Performance was measured in 2 different ways. The first was using the NASA-TLX. This is a measure that incorporates multiple measures, including mental demand, physical demand, frustration, and effort [19]. Multiple objective measures were included and are explained in detail in table 5.

Further to this, performance measures were also included as the number of mistakes and time taken to complete. This is reported as both an inverse efficiency score, reaction time and percentage of correct errors.

Ktistakis et al. [18] applied several machine learning methods to this dataset to predict cognitive workload. The models applied were GNB, random forest, SVM, Ensemble Gradient Boosting, k-nearest neighbours, Bernoulli Naïve Bayes, logistic regression and decision trees.

Hyperparameters were tuned through a random grid search iterated 1,000 times and a test-train split of 20-80 was applied. Ktistakis et al. [18] reported both accuracy and F-score and applied a k-fold cross validation of 5. The results obtained are detailed in table 3.

Key Measure	Author Measures
Fixation	Frequency, duration
Saccade	Frequency, duration, velocity
Blink	Frequency, duration
Pupil	Diameter

Table 1: Key cognitive workload measures and measures obtained by COLET authors.

2.3.2 A Machine Learning Approach for Detecting Cognitive Interference Based on Eye-Tracking Data (MLA)

Rizzo, Ermini, Zanca, Bernabini and Rossi [20] provide a publicly available dataset. They sought to determine cognitive workload through the use of cognitive interference in the form of a Stroop test. They tracked eye movement data across four tasks with varying levels of cognitive interference.

The data consists of 64 subjects that were tasked with performing 2 actions, reading and naming. Levels of cognitive workload were varied by either presenting interference, called reading with interference (RWI), naming with interference (NWI) or not presenting interference, named reading (R) or naming (N). This interference presented itself as a more difficult reading task. For example, in the R task participants would read the world ‘blue’ in black text, while the same word would be presented with incorrect spelling and an opposite colour, for example ‘blu’ in red text. Multiple eye movements were recorded, including number of fixations, length of fixation, and saccades. These are summarised in table 6.

Key Measure	Author Measures
Fixation	Count, average, minimum, maximum, regressions
Saccade	Frequency, duration, amplitude, angle, distance

Table 2: Key cognitive workload measures and measures obtained by MLA authors.

2.4 Machine learning models

The previous machine learning models used by previous researchers to predict cognitive workload will be reviewed. Table 3 shows accuracies and F1 measures (where reported) obtained by the authors on the datasets reviewed.

Accuracy is the proportion of correct predictions, and the F1 score is the harmonic mean of the precision and recall, combining precision and recall into a single metric [21]. Deep learning models have been employed by these authors, howeverI will limit the scope of this research to non-deep learning models.

2.4.1 SVM

SVM classifiers seek to develop a hyperplane between values, enabling it to assign data points to classes [22]. It has been extensively used as a classifier across many fields, including cognitive workload eye movement data [9][20][18].

SVM models have several advantages and disadvantages. The first advantage is that they works well for classification, however it is a non-probabilistic model [23], meaning that it is not able to consider random variation. This is a potential downside to the single use of the model.

SVM works well when there are two distinct classes, however it can under-perform when there are more features than data points [22]. This is a non-issue with the proposed datasets as neither of these have more features than data points, and further to this, the goal of the research is to classify binary classes, high and low cognitive workload. Due to its high performance on similar datasets, difference to GNB and decision trees, I will include it in the proposed stacked, ensemble model.

Model	Accuracy	F1	Author	Dataset
Decision Tree	73.4% (high, low)	71.65% (high, low)	Bozkir et al.(2019)	Private
	86.8% (high, low)	Not reported	Joseph et al.(2021)	Private
	74% (low/med, high)	73% (low/med, high)	Ktistakis et al.(2022)	COLET
SVM	93.3% (low, med, high)	Not reported	He et al. (2022)	Private
	80.7% (high, low)	80.98% (high, low)	Bozkir et al. (2019)	Private
	67% (R, RWI)	68% (R, RWI)	Rizzo et al. (2022)	MLA
	72% (N, NWI)	68% (R, RWI)	Rizzo et al. (2022)	MLA
	69% (low, med)	69% (low, med)	Ktistakis et al. (2022)	COLET
GNB	59% (low, med/high)	59% (low, med/high)	Ktistakis et al. (2022)	COLET
	88% (low, high)	86% (low, high)	Ktistakis et al. (2022)	COLET
LR	51% (low, med,high)	50% (low, ned,high)	Ktistakis et al. (2022)	COLET
	68% (N, NWI)	69% (N, NWI)	Rizzo et al. (2022)	MLA
	68% (R, RWI)	68% (R, RWI)	Rizzo et al. (2022)	MLA
Ensemble	58% (low, high)	57% (low, high)	Ktistakis et al. (2022)	COLET
RF	84% (low, high)	84% (low, high)	Ktistakis et al. (2022)	COLET
	62% (R, RWI)	67% (R, RWI)	Rizzo et al. (2022)	MLA
	62% (N, NWI)	59% (N, NWI)	Rizzo et al. (2022)	MLA

Table 3: Summary of model performance found by previous authors.

2.4.2 Decision Trees

Decision trees are models that classify through multiple nodes. Each node will seek to classify a data point based on several input features [24]. From these multiple nodes, the final classification is made.

One of the strengths of decision trees is that they can handle both categorical and continuous data and can classify quite quickly, in a simple and interpretable way [24]. It is a strong model for prediction and, unlike KNN or SVM, it bases its classification on multiple nodes. It does not require previous assumptions of the distribution of the data, and handles collinearity efficiently [25]. It is, however less robust and accurate than a decision tree, which is an ensemble model consisting of multiple decision trees.

As with the SVM model, it has been used previously to great effect, is an appropriate classifier, and differs from the previously reviewed models. Due to this, I will include it in the proposed stacked, ensemble model.

2.4.3 Gaussian naïve Bayes

Gaussian Naïve Bayes (GNB) classifiers work on the probabilistic theory of Bayes, and classifies data points accordingly. It works under the assumption of a gaussian distribution on an attribute given the class label. It has been used in multiple previous classification problems including text classification, document classification [26], and was used by Ktsikatis et al. [18] in classifying high, medium and low levels of cognitive workload given eye movement data.

One of the key strengths of GNB is that it can still classify accurately even when independence assumptions are violated [27]. A further strength is its flexibility and ability to work well with large datasets [28]. The main weakness of this model is the requirement for large datasets to get the best results [28].

Due to its high accuracy on the Colet dataset, ability to work even when independence assumptions are violated, and flexibility, I have selected it to be included in the proposed, stacked ensemble model.

2.4.4 Ensemble models

Two main ensemble models were discovered during the literature search. These are random forests and gradient boosted regression trees. These are usually chosen for their lack of overfitting, low computational expense and speed [29]. There is a lack of literature examining other ensemble methods, including stacked modelling which is a method to address single, weak learning models.

A stacked ensemble model is a subtype of ensemble modelling. The main advantage of using a stacked ensemble model is the ability to strengthen a model's predictive power by combining single, weak learning models [30].

2.5 Motivation

Based on recent literature, there is limited literature examining the effectiveness of ensemble models outside Random Forest and Gradient Boosted models in predicting levels of cognitive workload using eye movement data. My research will seek to further research in the field by investigating this gap in the literature.

I will use several machine learning models and a final, stacked ensemble model. The specific models are SVM, GNB and decision tree. Each of these models has been used in recent research to high levels of accuracy with a summary shown in table 3. Please note that the classes predicted are noted next to the accuracies obtained in table 3. It is important to note this as Ktistakis et al. [18] sought to classify multiple classes.

I hypothesise that by using multiple models together in a stacked ensemble this research will improve upon the existing models on the datasets presented by Ktistakis et al. [18] and Rizzo et al.[18]. By employing 3 different categorization methods; a Gaussian probabilistic approach with GNB; hyperplanes with SVM and a probabilistic approach with CART then the resulting stacked ensemble model will have a broader approach to categorization than a single, weaker model.

This literature review has presented important measures in predicting cognitive workload through eye movement data, including fixations, pupil size and blinks. I have presented multiple effective machine learning models that will serve as baseline models, as well as 2 data sets that will be used for analysis. I have presented the strengths and weaknesses of these models, as well as an argument for using ensemble models. The next part of this research report will introduce the research design.

3 Research Design & Methodology

3.1 Data Collection

As noted in the introduction, 2 appropriate, recent, and open-source datasets were identified. COLET [18] and MLA [20]. Multiple data sets were examined but only these datasets contained pupil, blink, fixation and saccade data, and are freely available. Using a free, open-source dataset means that ethical considerations have already been examined by the dataset authors and consent and other ethical considerations are not required to be assessed. Further to this, the authors have already de-identified the data.

3.2 Data pre-processing

Feature	Velocity threshold	Time threshold	Group
Saccade	> 0.45	N/A	N/A
Fixation	< 0.45	> 0.55 between movements	successive fixations grouped together

Table 4: Summary of thresholds used to identify fixations and saccades in COLET.

Feature	Aggregate applied
Blink	count
Blink duration	average
Pupil size	average
Fixations	count
saccades	count
Cognitive workload	N/A

Table 5: Summary of features for the COLET dataset.

3.2.1 Data Pre-processing: COLET

The COLET dataset was provided as a Matlab file. This was converted to excel files for ease of analysis in python.

Features were extracted based on features identified in the literature review. The count of blinks was obtained by counting the number of data points within the blink data, grouped by participant by task, indicating the frequency of blinks per participant per task. The average pupil diameter was obtained by taking the average value of the diameter grouped by participant by task. The count of fixations and the count of saccades were obtained by using the velocity identification threshold. This method was used by the COLET authors [18]. It works under the assumption that saccades and fixations have different velocities. Consecutive fixations are collapsed into groups and counted once, and the difference in time between observations must be above a specific threshold for it to be considered a fixation. These counts are then grouped together by participant and by task. This is summarised in table 5.

The final COLET dataset consists of the number of blinks per participant per task, average blink length per participant per task, the average score identified through the NASA TLX survey per participant per task, the average pupil diameter per participant per task, the number of fixations per participant per task and the count of saccades per participant per task.

Any rows with missing values were dropped as these cannot be used for prediction.

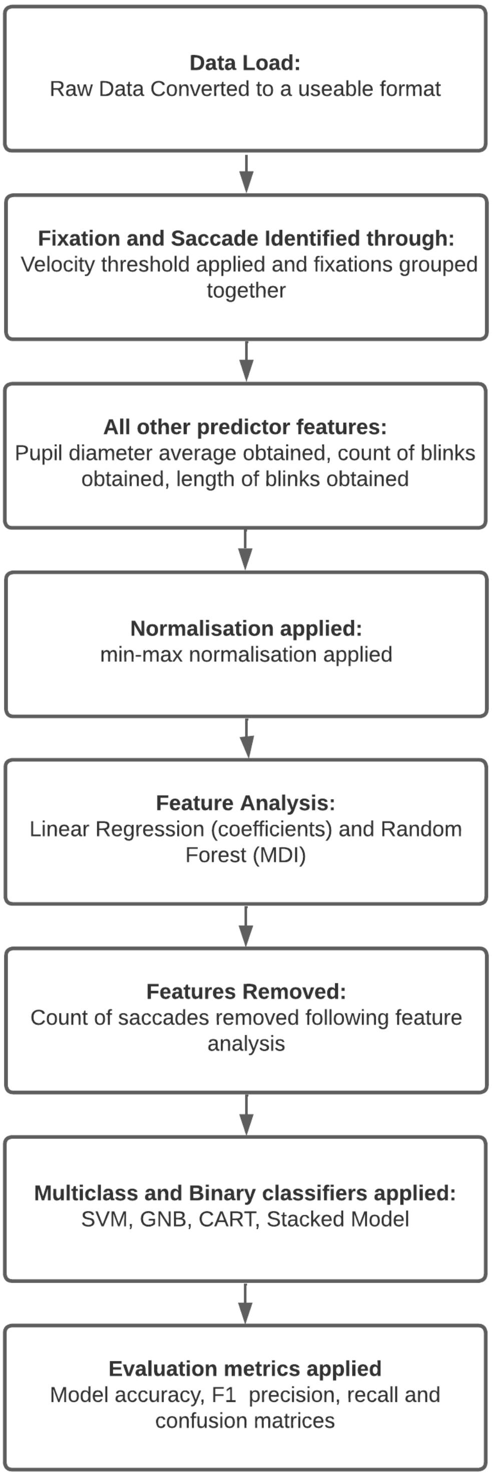

Data Load:
Raw Data Converted to a useable format
Fixation and Saccade Identified through:
Velocity threshold applied and fixations grouped together
All other predictor features:
Pupil diameter average obtained, count of blinks obtained, length of blinks obtained
Normalisation applied:
min-max normalisation applied
Feature Analysis:
Linear Regression (coefficients) and Random Forest (MDI)
Features Removed:
Count of saccades removed following feature analysis
Multiclass and Binary classifiers applied:
SVM, GNB, CART, Stacked Model
Evaluation metrics applied
Model accuracy, F1 precision, recall and confusion matrices

Figure 1: Summary of modelling process for COLET Dataset.

This resulted in a total of 18 records being removed. The values of the NASA TLX recordings (values used to identify cognitive workload) were converted from their raw values to 1, 2, or 3 meaning low, medium and high cognitive workload. These values were identified from the author's paper where they noted 50-100 is high, 30-49 is medium, and 0-29 are considered low values [18]. Binary conditions are noted in 4.3.

Finally, the feature data was scaled to values between 0 and 1 using a min-max scaler. A summary of the entire process is shown in figure 1.

3.2.2 Data Pre-processing: A Machine Learning Approach for Detecting Cognitive Interference Based on Eye-Tracking Data (MLA)

The MLA dataset was provided as 1996 excel files by Rizzo et al. [20]. These were converted to CSV files for ease of analysis.

Multiple files were included, but not all were used in the analysis. Interest area was not used as it was not relevant to prediction. Fixation files were used, gaze files were used and saccades files were used.

Minimal pre-processing was required for this data. Pupil size was recorded for both left and right eyes sequentially. These values were combined into one eye record. Average diameter was taken from these values.

The count of fixations was taken from the 'current fix pupil' field by taking the count of this, and grouping by the participant, resulting in a count of fixations by participant by task. Similarly, the count of saccades was obtained by taking the count of values for 'current sac duration', resulting in the count of saccades by participant by task. The count of blinks were taken by counting the number of values for the 'current sac blink end' field, resulting in the count of blinks by participant and task. These are summarised in 6

As with the COLET dataset, any rows with missing data were dropped as these missing values could not be used for prediction.

In this case, the target feature for cognitive workload are the tasks, as these are what was manipulated by the authors. The target variables were the tasks (1 to 4). Binary conditions are noted in 4.3. The summary of data processing and modelling for the

Feature	Aggregate applied
Blink	count
Blink duration	average
Pupil size	average
Fixations	count
saccades	count

Table 6: Summary of features for the MLA dataset.

MLA dataset is summarised in figure 2.

4 Artefact Development Approach

4.1 Feature analysis

4.1.1 Feature analysis: COLET

I analysed the COLET dataset features through 2 different methods. The first method was a linear regression model. The feature coefficients were used as a way to examine feature importance. The second method for feature analysis was to examine the mean decrease in impurity using a random forest model. Both of these methods have been used by previous researchers as a form of feature analysis [31][32].

The results are summarised in figures 3 and 4. As the count of saccades had the lowest coefficient score and acted as a detractor in the linear regression analysis, and did not contribute significantly in the random forest mean decrease in impurity analysis, I removed the feature from the dataset.

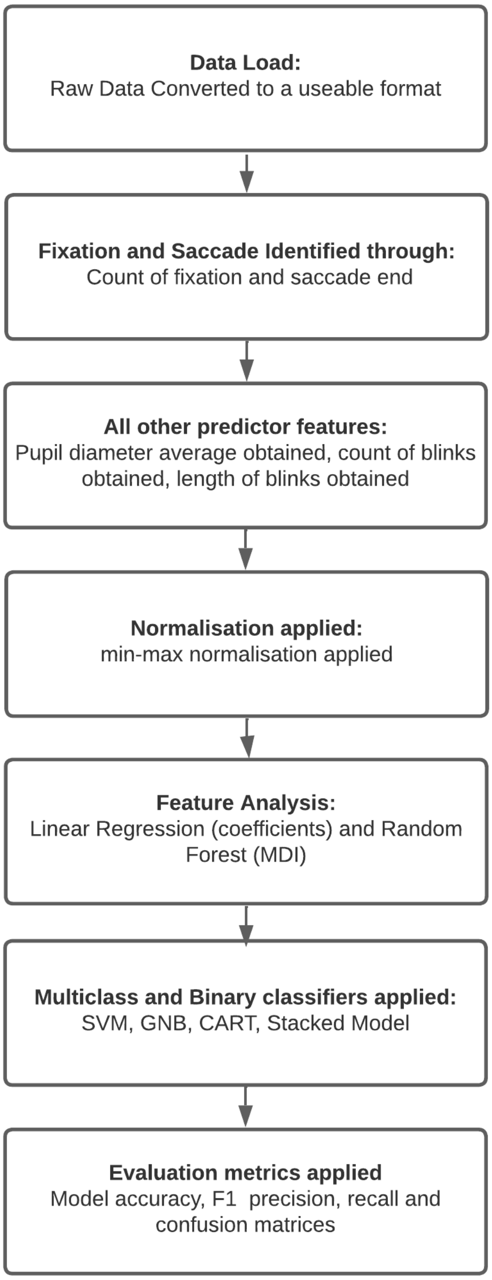

Data Load:
Raw Data Converted to a useable format
Fixation and Saccade Identified through:
Count of fixation and saccade end
All other predictor features:
Pupil diameter average obtained, count of blinks obtained, length of blinks obtained
Normalisation applied:
min-max normalisation applied
Feature Analysis:
Linear Regression (coefficients) and Random Forest (MDI)
Multiclass and Binary classifiers applied:
SVM, GNB, CART, Stacked Model
Evaluation metrics applied
Model accuracy, F1 precision, recall and confusion matrices

Figure 2: Summary of modelling process for the MLA Dataset .

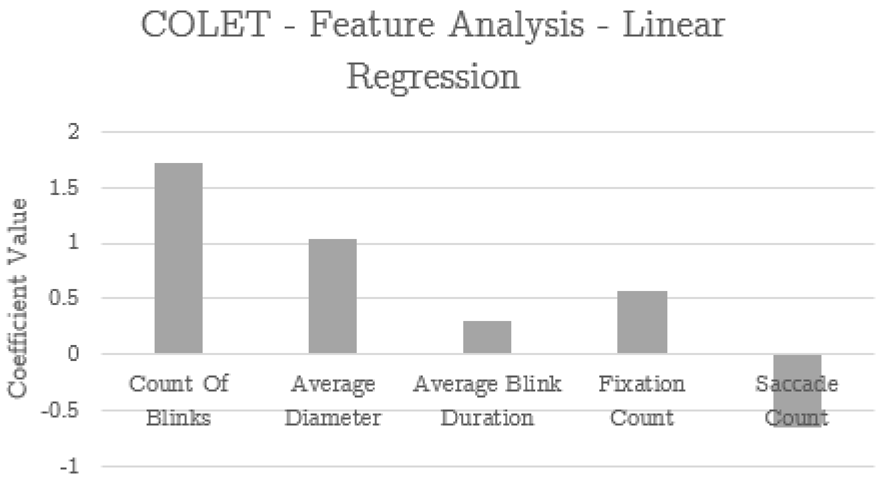


Figure 3: COLET feature analysis linear regression.

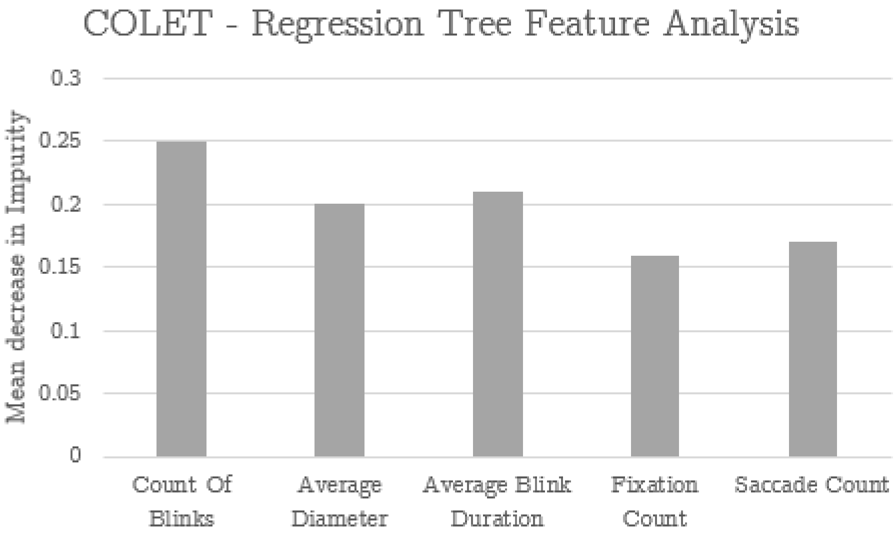


Figure 4: COLET feature analysis regression tree.

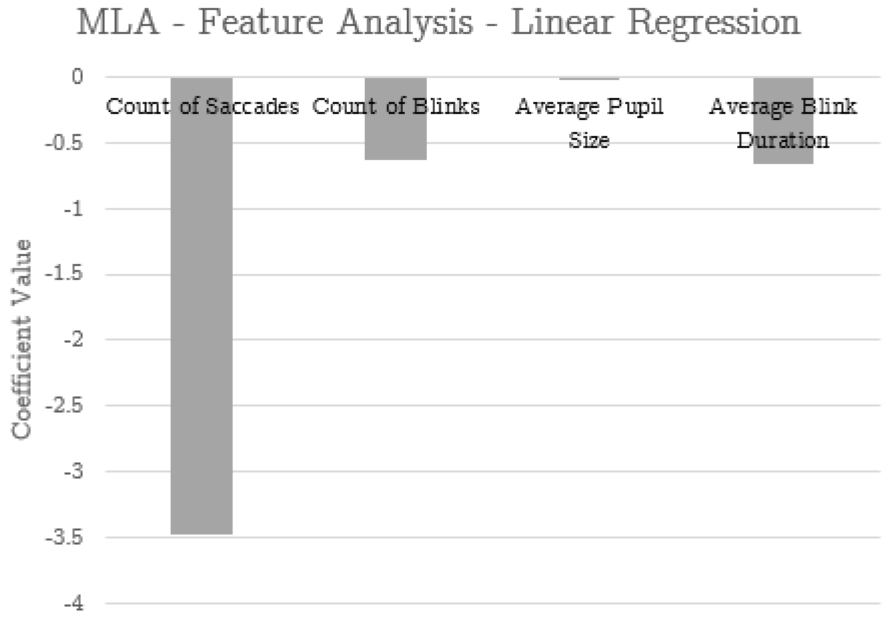


Figure 5: MLA feature analysis linear regression.

4.1.2 Feature analysis: MLA

I conducted the same feature analyses on the MLA dataset. The results of these are summarised in figures 5 and 6. It was decided that no features would be removed following this analysis. The reasoning was that while the count of saccades was the largest detractor in the linear regression analysis, it was also the largest contributor in the regression tree analysis. I could therefore not justifiably remove it and it remained in the analysis.

4.2 Hyperparameter Tuning

Prior to model building I tuned the hyperparameters. A 10 cross fold validation, random grid search method was used. The main drawback of this approach us its inneficiency on large datasets[33]; however, following data cleaning, the datasets are not considered large, and this approach can be used effectively. The optimal hyperparameters are noted table 7.

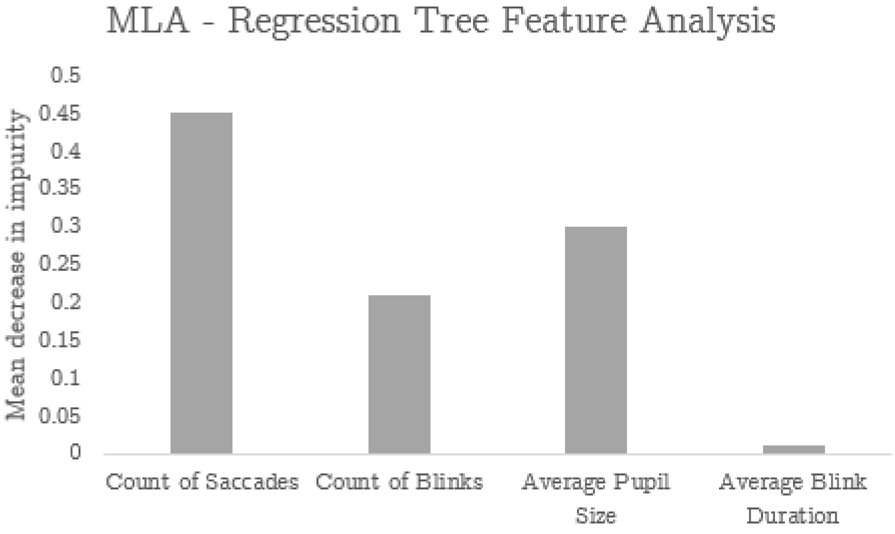


Figure 6: MLA feature analysis regression tree.

Model	Hyperparamter tested	Optimal Multiclass	Optimal Binary
SVM	C:0.1, 1, 10, 100, 1000	COLET: 10	COLET:10
	kernel:linear, poly, rbf,sigmoid	COLET: rbf	COLET: sigmoid
	gamma:1, .1, .01, .001, .0001, .00001	COLET: 1	COLET: 1
	C:0.1, 1, 10, 100, 1000	MLA: 100	MLA:10
	kernel:linear, poly, rbf,sigmoid	MLA: sigmoid	MLA: sigmoid
	gamma:1, .1, .01, .001, .0001, .00001	MLA: 1	MLA: 1
GNB	smoothing:100 values 0:-9 log scaled	COLET: 0.5336699231	COLET: 1
	smoothing:100 values 0:-9 log scaled	MLA: 2.8480358684e-09	MLA: 0.151991108295
CART	max depth: 2, 3,4,5, 6,8,10	COLET: 2	COLET:2
	min samples: .04, .06, .08, .10, .2, .4	COLET: .08	COLET: .2
	max features: .2, .4, .6, .8, .10, .2, .4	COLET: .6	COLET: .6
	max depth: 2, 3,4,5, 6,8,10	MLA: 3	MLA:3
	min samples: .04, .06, .08, .10, .2, .4	MLA: .04	MLA: .04
	max features: .2, .4, .6, .8, .10, .2, .4	MLA: .2	MLA: .2

Table 7: Summary of hyperparameters tested and optimal hyperparamters found for the COLET and MLA datasets on the multiclass and binary classifiers.

4.3 Models

Finally, with the hyperparameters tuned, I created the models used to predict their targets. The first set of models predicted the targets as provided by the authors. For COLET, this was predicting low, medium and high cognitive workload. This was identified through the mean NASA TLX score ranges. These ranges were classified as 0-29 as low, 30 to 49 as medium, and 50 to 100 as high. These 3 levels were used for the multiclass classifiers. For the binary classifiers, high and low scores were merged into 1, and are considered 'abnormal' ranges, while the medium score acted as the second target.

For the MLA dataset, the authors did not measure levels of cognitive workload, but did manipulate the levels of workload through their activities. As noted in section 2.3.2 these activities were split into 4. R, RWI, N, and NWI. For the multiclass classifiers, all 4 were target variables, while for the binary classifiers, the interference conditions were grouped together, while the no interference were grouped together.

The models used were SVM, classification and regression tree (CART) and GNB. Finally, these classifiers were stacked together into a stacked, ensemble model.

The ensemble stacking model used is comprised of the 3 previously mentioned models, CART, GNB and SVM. The model uses both the CART and GNB in the first layer, with SVM in the final layer due to its high performance as a single model.

I conducted additional experiments to validate the model by examining how the model performed in its intermediate stages, by removing either the CART or GNB models from the ensemble. The performance was also tested by using GNB or CART as the final classifier rather than SVM. The results are shown in tables 8,9.

4.4 Ethical consideration

While the final models are not in production, I have taken into account multiple ethical considerations while preparing this research report. These include bias, transparency and explainability [34][35].

Machine learning bias is a widely acknowledged ethical issue in machine learning[35]. One such issue for this research paper would be participants that are blind, or only have one eye. The data obtained is limited to eye tracking, and participants are therefore required to have 2 eyes that can be tracked.

Transparency has been noted early in this research paper by identifying the goal of detecting cognitive workload using eye movement data.

Finally, explainability must be examined. While many machine learning models may follow a 'black box' approach, each of the models within this research paper have been explained thoroughly as noted in 2.4.

5 Empirical Evaluation

5.1 Research Questions (RQs)

I have posed the following research questions. “Which features in the available datasets are relevant for detecting cognitive workload?”, the second is “can eye movement features be used to determine high and low cognitive workload through the use of machine learning?” and the final question is “will a stacked ensemble model outperform the previously presented models on the previously presented datasets?”.

Based on the literature presented I hypothesized that only the most relevant features will be retained through the use of feature analysis, that all models will achieve their task of predicting cognitive workload through the use of eye movement data, and finally that the ensemble model will outperform the individual models.

5.2 Evaluation metrics

I used multiple evaluation metrics to examine the performance of the models, including accuracy, F1 score, recall and precision. All papers examined to date use metrics comparison rather than hypothesis testing for model evaluation, and as such the same approach has been adopted for this report.

Accuracy is defined as the sum of the correct predictions divided by the sum of the total predictions [36] or in other words, the proportion of correct predictions. It is a

Classifier	Dataset	Accuracy	Precision	Recall	F1
SVM	COLET	.482	.412	.383	.344
CART	COLET	.482	.510	.489	.456
GNB	COLET	.482	.172	.311	.222
Stacked (CART + GNB + SVM Final)	COLET	.448	.257	.411	.311
Stacked (CART + SVM)	COLET	.448	.260	.411	.311
Stacked (GNB + SVM)	COLET	.482	.180	.311	.227
Stacked (CART + SVM + GNB Final)	COLET	.483	.414	.383	.345
Stacked (SVM + GNB + CART Final)	COLET	.483	.277	.433	.331
SVM	MLA	.370	.296	.417	.344
CART	MLA	.407	.307	.379	.328
GNB	MLA	.222	.146	.184	.155
Stacked (CART + GNB + SVM Final)	MLA	.222	.213	.257	.217
Stacked (CART + SVM)	MLA	.259	.287	.323	.254
Stacked (GNB + SVM)	MLA	.222	.168	.260	.199
Stacked (CART + SVM + GNB Final)	MLA	.333	.214	.323	.257
Stacked (SVM + GNB + CART Final)	MLA	.296	.245	.416	.259

Table 8: Summary of classifier results for multiclass classifiers.

good general predictor.

F1 score is defined as the harmonic mean of the precision and recall [37]. It is considered the most widely used metric of accuracy in most areas of machine learning [37]. It is particularly useful in situations with an unbalanced dataset.

It is for the above reasons, the generality of the accuracy metric, and the widespread use of the F1 score that these metrics have been chosen as the comparison metrics, further to this they are available as baseline measures in the datasets provided by the previous authors.

Beyond these comparison metrics, additional metrics have been included to allow a greater understanding of the model results. These are recall and precision.

The recall can be defined as the ratio of true positives in comparison to the ground truth. It focuses on type 2 errors. A lower recall value (less than 0.5) indicates a high number of false negatives [38].

Precision is the ratio of true positives and total positives. A low precision score (less than 0.5) indicates a high number of false positives [38].

Classifier	Dataset	Accuracy	Precision	Recall	F1
SVM	COLET	.724	.852	.600	.579
CART	COLET	.621	.533	.521	.505
GNB	COLET	.655	.327	.500	.396
Stacked (CART + GNB + SVM Final)	COLET	.655	.327	.500	.396
Stacked (CART + SVM)	COLET	.655	.185	.500	.396
Stacked (GNB + SVM)	COLET	.655	.494	.494	.396
Stacked (CART + SVM + GNB Final)	COLET	.655	.327	.500	.396
Stacked (SVM + GNB + CART Final)	COLET	.655	.583	.524	.476
SVM	MLA	.629	.617	.624	.617
CART	MLA	.621	.477	.477	.444
GNB	MLA	.512	.493	.494	.493
Stacked (CART + GNB + SVM Final)	MLA	.482	.557	.547	.475
Stacked (CART + SVM)	MLA	.370	.185	.500	.270
Stacked (GNB + SVM)	MLA	.519	.494	.494	.493
Stacked (CART + SVM + GNB Final)	MLA	.593	.587	.594	.583
Stacked (SVM + GNB + CART Final)	MLA	.629	.617	.624	.615

Table 9: Summary of classifier results for binary classifiers.

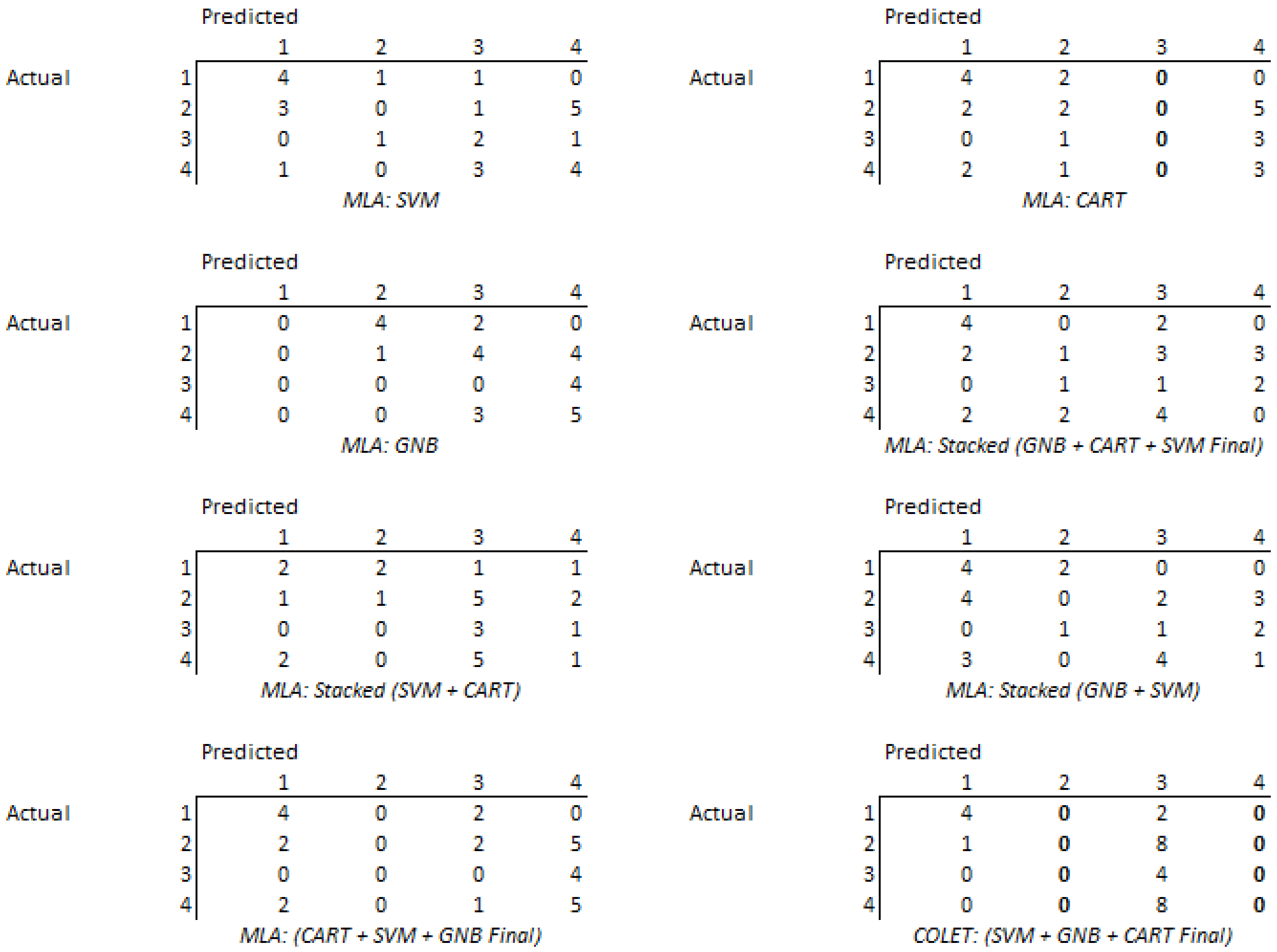
5.3 Confusion matrices

Confusion matrices are presented in figures 7, 8, 9, 10. These figures have been included to aid in the discussion. Classes that do not appear to have been identified are in bold.

6 Results & Discussion

6.1 Results

Through feature analysis I identified that blink frequency, blink durations, fixation frequency, and pupil diameter were all relevant features for COLET. The relevant features for MLA were more difficult to identify. Linear regression revealed that all features were detractors, while the greatest detractor in this analysis was found to be the strongest predictor in the regression tree analysis. Due to this, I could not justify removing any features.



		Predicted			
		1	2	3	4
Actual	1	4	1	1	0
	2	3	0	1	5
	3	0	1	2	1
	4	1	0	3	4

MLA: SVM

		Predicted			
		1	2	3	4
Actual	1	4	2	**0**	0
	2	2	2	**0**	5
	3	0	1	**0**	3
	4	2	1	**0**	3

MLA: CART

		Predicted			
		1	2	3	4
Actual	1	0	4	2	0
	2	0	1	4	4
	3	0	0	0	4
	4	0	0	3	5

MLA: GNB

		Predicted			
		1	2	3	4
Actual	1	4	0	2	0
	2	2	1	3	3
	3	0	1	1	2
	4	2	2	4	0

MLA: Stacked (GNB + CART + SVM Final)

		Predicted			
		1	2	3	4
Actual	1	2	2	1	1
	2	1	1	5	2
	3	0	0	3	1
	4	2	0	5	1

MLA: Stacked (SVM + CART)

		Predicted			
		1	2	3	4
Actual	1	4	2	0	0
	2	4	0	2	3
	3	0	1	1	2
	4	3	0	4	1

MLA: Stacked (GNB + SVM)

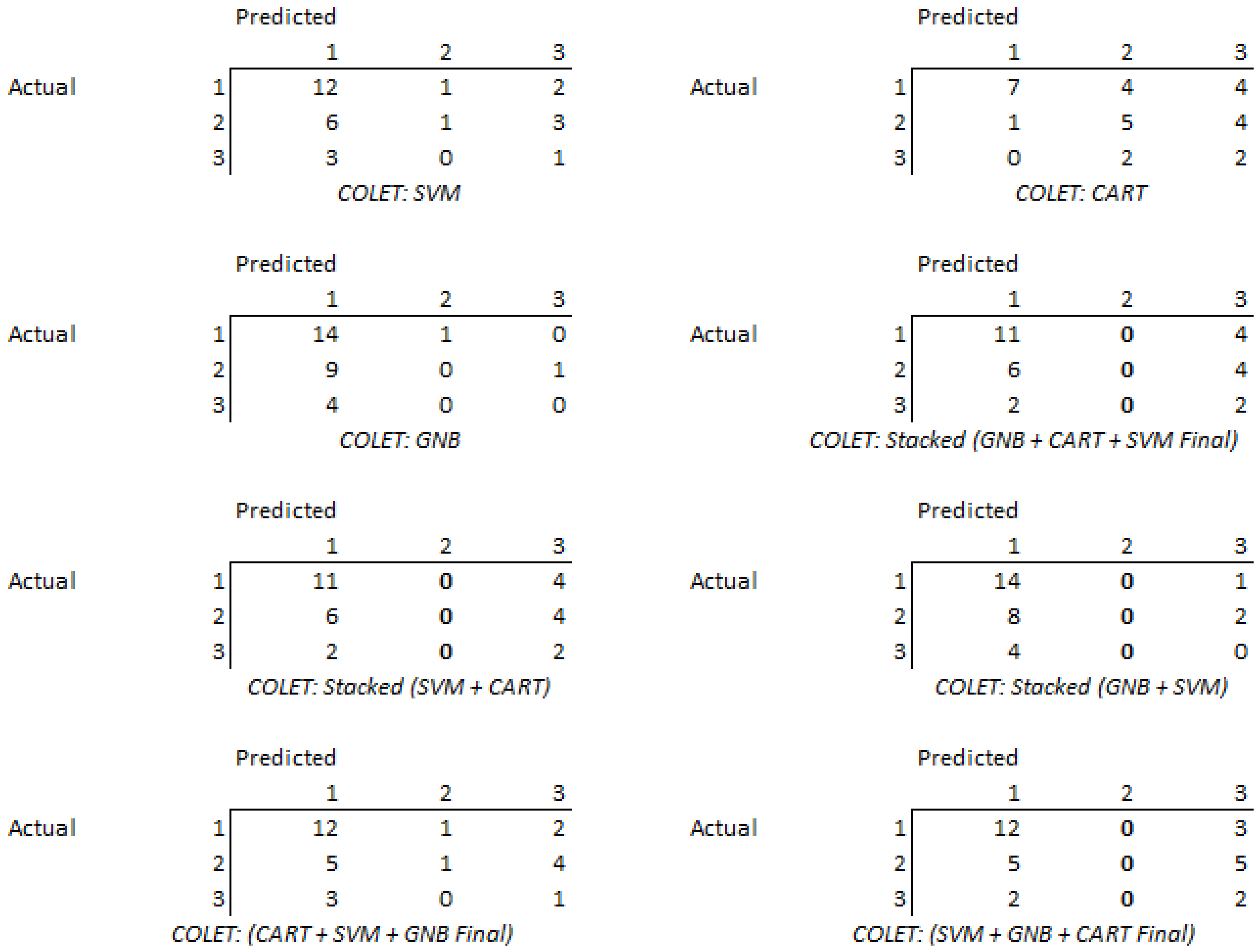
		Predicted			
		1	2	3	4
Actual	1	4	0	2	0
	2	2	0	2	5
	3	0	0	0	4
	4	2	0	1	5

MLA: (CART + SVM + GNB Final)

		Predicted			
		1	2	3	4
Actual	1	4	**0**	2	**0**
	2	1	**0**	8	**0**
	3	0	**0**	4	**0**
	4	0	**0**	8	**0**

COLET: (SVM + GNB + CART Final)

Figure 7: MLA multi-class confusion matrices.



		Predicted		
		1	2	3
Actual	1	12	1	2
	2	6	1	3
	3	3	0	1

COLET: SVM

		Predicted		
		1	2	3
Actual	1	7	4	4
	2	1	5	4
	3	0	2	2

COLET: CART

		Predicted		
		1	2	3
Actual	1	14	1	0
	2	9	0	1
	3	4	0	0

COLET: GNB

		Predicted		
		1	2	3
Actual	1	11	0	4
	2	6	0	4
	3	2	0	2

COLET: Stacked (GNB + CART + SVM Final)

		Predicted		
		1	2	3
Actual	1	11	0	4
	2	6	0	4
	3	2	0	2

COLET: Stacked (SVM + CART)

		Predicted		
		1	2	3
Actual	1	14	0	1
	2	8	0	2
	3	4	0	0

COLET: Stacked (GNB + SVM)

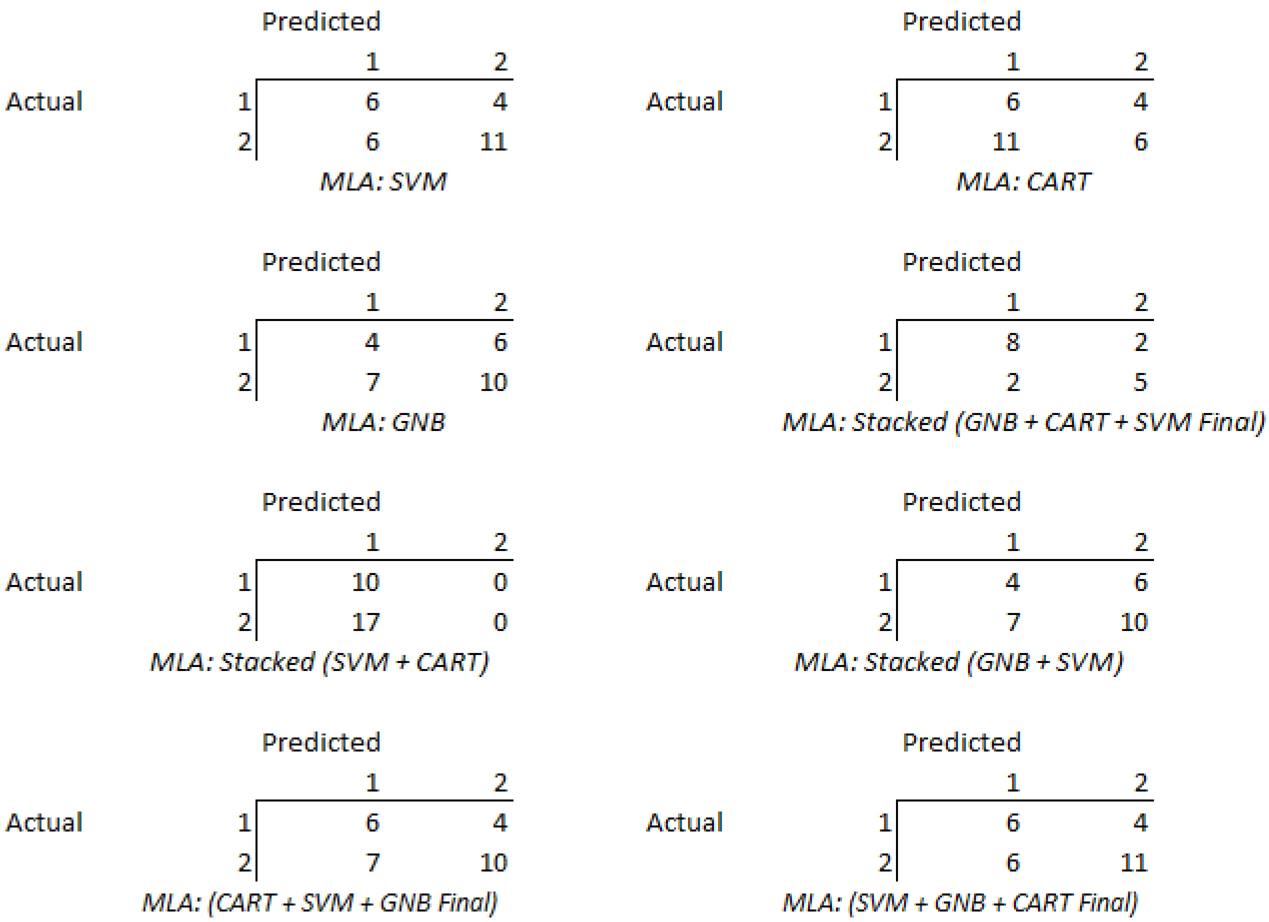
		Predicted		
		1	2	3
Actual	1	12	1	2
	2	5	1	4
	3	3	0	1

COLET: (CART + SVM + GNB Final)

		Predicted		
		1	2	3
Actual	1	12	0	3
	2	5	0	5
	3	2	0	2

COLET: (SVM + GNB + CART Final)

Figure 8: COLET multi-class confusion matrices.



		Predicted	
		1	2
Actual	1	6	4
	2	6	11

MLA: SVM

		Predicted	
		1	2
Actual	1	6	4
	2	11	6

MLA: CART

		Predicted	
		1	2
Actual	1	4	6
	2	7	10

MLA: GNB

		Predicted	
		1	2
Actual	1	8	2
	2	2	5

MLA: Stacked (GNB + CART + SVM Final)

		Predicted	
		1	2
Actual	1	10	0
	2	17	0

MLA: Stacked (SVM + CART)

		Predicted	
		1	2
Actual	1	4	6
	2	7	10

MLA: Stacked (GNB + SVM)

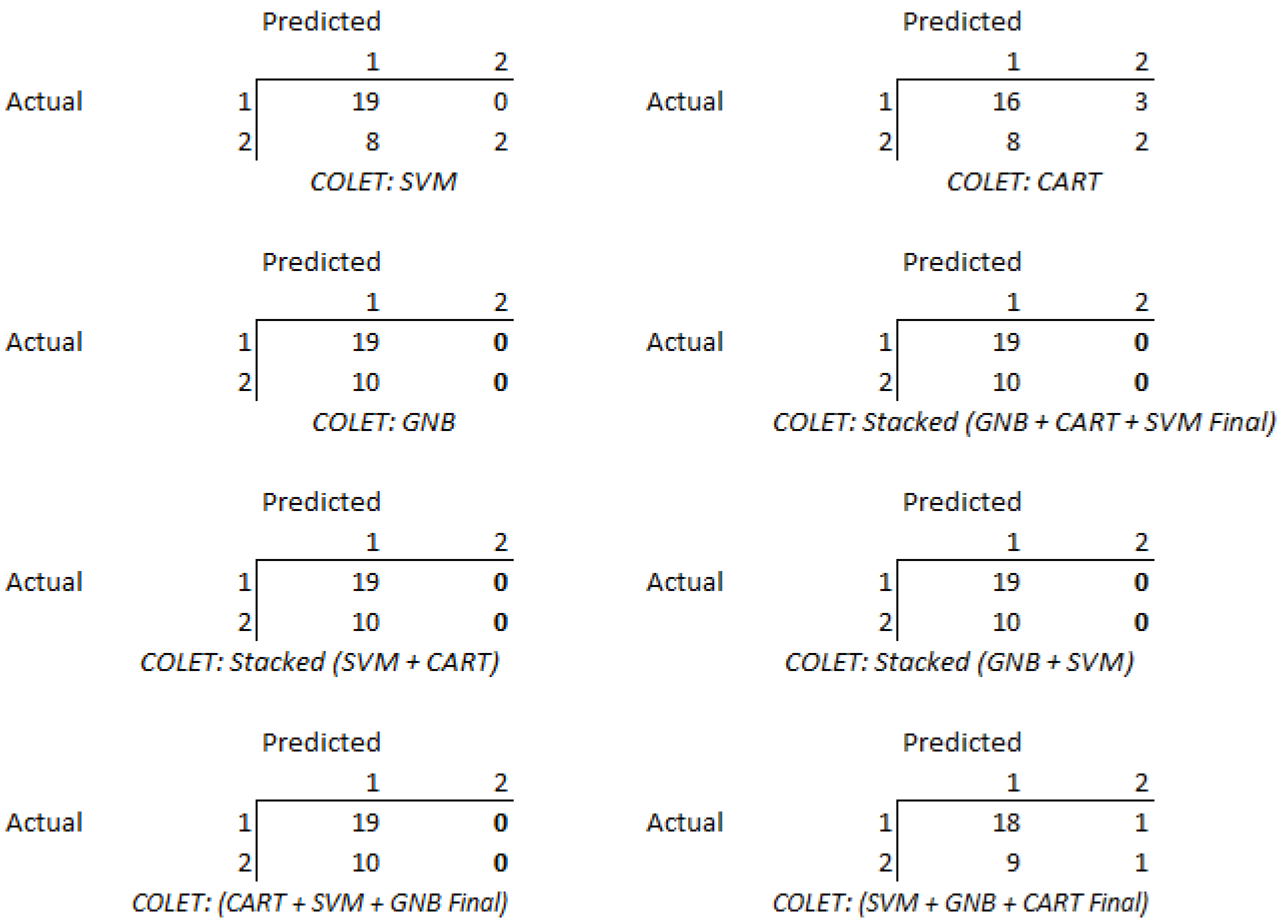
		Predicted	
		1	2
Actual	1	6	4
	2	7	10

MLA: (CART + SVM + GNB Final)

		Predicted	
		1	2
Actual	1	6	4
	2	6	11

MLA: (SVM + GNB + CART Final)

Figure 9: MLA binary confusion matrices.



		Predicted	
		1	2
Actual	1	19	0
	2	8	2

COLET: SVM

		Predicted	
		1	2
Actual	1	16	3
	2	8	2

COLET: CART

		Predicted	
		1	2
Actual	1	19	**0**
	2	10	**0**

COLET: GNB

		Predicted	
		1	2
Actual	1	19	**0**
	2	10	**0**

COLET: Stacked (GNB + CART + SVM Final)

		Predicted	
		1	2
Actual	1	19	**0**
	2	10	**0**

COLET: Stacked (SVM + CART)

		Predicted	
		1	2
Actual	1	19	**0**
	2	10	**0**

COLET: Stacked (GNB + SVM)

		Predicted	
		1	2
Actual	1	19	**0**
	2	10	**0**

COLET: (CART + SVM + GNB Final)

		Predicted	
		1	2
Actual	1	18	1
	2	9	1

COLET: (SVM + GNB + CART Final)

Figure 10: COLET binary confusion matrices.

The results of each classifier across both the binary and multiclass classifiers are noted in table 9,8. SVM performed best when examining the accuracy on both datasets (0.742 COLET binary, 0.629 MLA binary), however both its performance on both datasets exhibited lower F1 scores than accuracy (0.579 COLET binary, 0.617), indicating the model is underperforming overall, and may be indicate a lower recall or precision. The high precision value (0.852, COLET binary, 0.617, MLA binary) indicates that the model is unlikely to make a type 1 error and is able to accurately identify correctly, while the high recall values (0.600 COLET binary, 0.624 MLA binary) indicate the model is less likely to make a type 2 error. Reviewing the confusion matrices provides further insight into the best performing model. We can see that the classes are fairly imbalanced, with only 2 predictions in the second class in total for COLET. This imbalance persists for multiple COLET multiclass classifiers as seen in figure 8. However this effect is less pronounced for the MLA dataset, with more balanced predictions in each class. This imbalance will be part of the reason for these high metrics in the COLET data.

For both datasets, binary classifiers were able to identify cognitive workload at greater than 50% accuracy (except for the stacked binary classifier on the MLA dataset), indicating that eye movement data can be used to predict cognitive workload.

The evaluation metrics are split into multiclass and binary. See section 5.2 for an explanation of these. In 9 and 8 the models noted under classifier are titled based on the individual model, the models contained within and the order of these models. Additional experimentation on the ensemble stacking models identified that removing either the GNB or CART models from the intermediate stages made little difference to the accuracy in both the multiclass and binary approaches.

Overall, the binary classifiers outperformed the multiclass classifiers. The single models underperformed compared to those in the presented papers [20],[18], and the stacked ensemble models did not perform as well as expected. The likely reasons for these results will be examined below in the discussion.

6.2 Discussion

There are several possible reasons for the difference between the results obtained in this report compared to previous literature. While processing COLET it was noted that there were large amounts of missing or incorrect data, particularly in blink data.

As records with missing data were removed, this resulted in a smaller amount of remaining data. This low sample size will impact the model prediction as has been shown in previous literature (for example,[39]).

Additionally, it is possible that the method of identifying saccades and fixations in the COLET dataset could be misunderstood. It was interpreted as noted in 4. If this was interpreted incorrectly this will result in further difference in analysis for the COLET dataset.

Another reason for difference are the hyperparameters. It is unclear exactly which hyperparameters were used by Rizzo et al. [20] and Ktistakis et al. [18]. Addtiionally, due to the random selection in the test-train split, there will be a difference in the data selected for testing and training data in both datasets, which will further result in further differences.

It is difficult to compare the results obtained to those presented by Rizzo et al. [20]. In order to limit the scope of the study, the binary targets identified were created by combining both interference conditions and by combining both non-interference conditions. This differs from how Rizzo et al. [20] performed for their analysis.

While the performance of the ensemble modelling was not as high as expected, the values of the ensemble models do not differ significantly from the individual models. As the ensemble model is comprised of each of these single models, it is not unsurprising that it does not significantly outperform the individual models.

7 Threats to Validity

I identified several threats to validity. Neither datasets indicate where participants were recruited from. This is identified as a threat to external validity as it limits the applicability of the research in a real world setting.

Further to this, it is difficult to generalize the results to real world settings. In the study by Rizzo et al. [20], eye tracking data is obtained while asking participants to do a Stroop test. It is unclear exactly when this may be applicable in real-life situations, while the data obtained by Ktistakis et al. [18] has participants performing a captcha-like task. This is more generalizable to situations such as driving as participants are searching a specific area for hazards.

It is difficult to examine most threats to internal validity such as the situation or Hawthorne effects [40] due to the limited knowledge about participant recruitment, however several are identified. An internal threat to validity is the identification of saccades and fixations in the COLET dataset. As noted previously, this was interpreted as shown in table 4. If interpreted incorrectly it will change the count of saccades and fixations, and thus will change the feature values. Another internal threat to validity is the small sample size which will impact the model performance.

8 Conclusion & Future Work

8.1 Future Work

Following this research several ideas for future directions are presented. One such direction would be employing the use of deep learning. Multiple researchers have used deep learning to predict cognitive workload from eye movement data, including Rizzo et al. [20]. This will expand upon the current models.

As noted previously a possible reason for the underperformance of the models may be due to additional features used by the authors not being included in the presented models. As such, the inclusion of additional features is suggested for future directions. Due to time limitations a limited number of features were included, however it is suggested to use the same features as the authors including minimum, maximum saccade and fixation durations, minimum, maximum and average velocity, amongst many others.

Another suggestion to address the randomness of the test train split would be to average the results of multiple iterations of the models with different test train splits to limit the impact of this on the overall results.

As noted in section 6.1 while the model metrics look quite promising, the predictions on the model appear to be quite imbalanced once split into binary predictors (see figure 8 for examples of this). Addressing this through oversampling is one suggested method. This method has been noted in current literature to be an effective way of handling the issue of class imbalance [41].

Another suggestion would be to use different models in the ensemble stacking model. As noted previously, the models used in this research were chosen due to their high performance noted by previous authors, however it may be useful to add in additional single models to test their performance and add in only the best performing models, and testing different stacked models. Model suggestions include logistic regression and K-Nearest-Neighbors.

8.2 Conclusion

In conclusion, this report has posed and attempted to answer the 3 research questions noted in 5.1 in regards to multiple machine learning models and multiple datasets for cognitive workload detection. The research has revealed that using the datasets COLET and MLA whose features are noted in tables 5.6 were identified as relevant for detecting cognitive workload following the feature analysis noted in figures 3, 4, 5, 6. Further to this, it was found that using SVM, CART, GNB and stacked ensemble models that levels of cognitive workload could be detected. Finally, through the use of a stacked ensemble model comprised of SVM, CART and GNB, it was found that this did not significantly improve cognitive workload detection. This model was tested for intermediate results and different combinations were tested but ultimately, none of these experiments produced significantly better results than the single models, nor did the models outperform those presented by the COLET and MLA authors.

References

[1] Mastaneh Torkamani-Azar, Ahreum Lee, and Roman Bednarik. Methods and measures for mental stress assessment in surgery: A systematic review of 20 years of literature. IEEE Journal of Biomedical and Health Informatics, 2022.

[2] Antony William Joseph and Ramaswamy Murugesh. Potential eye tracking metrics and indicators to measure cognitive load in human-computer interaction research. J. Sci. Res, 64(1):168–175, 2020.

[3] Hadia Tazeem, Atul Sajjanhar, Glory Lee, and Dawei Jia. Random forest for event classification of eye movements: Towards effective cognitive workload estimation. 2021.

[4] LRD Murthy and Pradipta Biswas. Deep learning-based eye gaze estimation for military aviation. In 2022 IEEE Aerospace Conference (AERO), pages 1–8. IEEE, 2022.

[5] Linnéa Larsson. Event detection in eye-tracking data for use in applications with dynamic stimuli. PhD thesis, Lund University, 2016.

[6] Omer Sagi and Lior Rokach. Ensemble learning: A survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8(4):e1249, 2018.

[7] Ahmad F Klaib, Nawaf O Alsrehin, Wasen Y Melhem, Haneen O Bashtawi, and Aws A Magableh. Eye tracking algorithms, techniques, tools, and applications with an emphasis on machine learning and internet of things technologies. Expert Systems with Applications, 166:114037, 2021.

[8] Dengbo He, Ziquan Wang, Elias B Khalil, Birsen Donmez, Guangkai Qiao, and Shekhar Kumar. Classification of driver cognitive load: exploring the benefits of fusing eye-tracking and physiological measures. Transportation research record, 2676(10):670–681, 2022.

[9] Efe Bozkir, David Geisler, and Enkelejda Kasneci. Person independent, privacy preserving, and real time assessment of cognitive load using eye tracking in a virtual reality setup. In 2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), pages 1834–1837. IEEE, 2019.

[10] Antony William Joseph, J Sharmila Vaiz, and Ramaswami Murugesh. Modeling cognitive load in mobile human computer interaction using eye tracking metrics. In Advances in Artificial Intelligence, Software and Systems Engineering: Proceedings of the AHFE 2021 Virtual Conferences on Human Factors in Software and Systems Engineering, Artificial Intelligence and Social Computing, and Energy, July 25-29, 2021, USA, pages 99–106. Springer, 2021.

[11] Mina Shojaeizadeh, Soussan Djamasbi, Randy C Paffenroth, and Andrew C Trapp. Detecting task demand via an eye tracking machine learning system. Decision Support Systems, 116:91–101, 2019.

[12] Petar Jerčić, Charlotte Sennersten, and Craig Lindley. Modeling cognitive load and physiological arousal through pupil diameter and heart rate. Multimedia Tools and Applications, 79(5):3145–3159, 2020.

[13] P Ramakrishnan, B Balasingam, and F Biondi. Cognitive load estimation for adaptive human–machine system automation. In Learning control, pages 35–58. Elsevier, 2021.

[14] Tanya Bafna, John Paulin Paulin Hansen, and Per Baekgaard. Cognitive load during eye-typing. In ACM symposium on eye tracking research and applications, pages 1–8, 2020.

[15] Tobias Appel, Natalia Sevcenko, Franz Wortha, Katerina Tsarava, Korbinian Moeller, Manuel Ninaus, Enkelejda Kasneci, and Peter Gerjets. Predicting cognitive load in an emergency simulation based on behavioral and physiological measures. In 2019 International Conference on Multimodal Interaction, pages 154–163, 2019.

[16] M Dilli Babu, DV JeevithaShree, Gowdham Prabhakar, Kamal Preet Singh Saluja, Abhay Pashilkar, and Pradipta Biswas. Estimating pilots' cognitive load from ocular parameters through simulation and in-flight studies. Journal of Eye Movement Research, 12(3), 2019.

[17] Jung-Chun Liu, Kuei-An Li, Su-Ling Yeh, and Shao-Yi Chien. Assessing perceptual load and cognitive load by fixation-related information of eye movements. Sensors, 22(3):1187, 2022.

[18] Emmanouil Ktistakis, Vasileios Skaramagkas, Dimitris Manousos, Nikolaos S Tachos, Evanthia Tripoliti, Dimitrios I Fotiadis, and Manolis Tsiknakis. Colet: A dataset for cognitive workload estimation based on eye-tracking. Computer Methods and Programs in Biomedicine, 224:106989, 2022.

[19] NASA. Nasa tlx - task load index.

[20] Antonio Rizzo, Sara Ermini, Dario Zanca, Dario Bernabini, and Alessandro Rossi. A machine learning approach for detecting cognitive interference based on eye-tracking data. Frontiers in Human Neuroscience, 16, 2022.

[21] Evaluate the performance of processors nbsp;|nbsp; document ai nbsp;|nbsp; google cloud.

[22] William S Noble. What is a support vector machine? Nature biotechnology, 24(12):1565–1567, 2006.

[23] Xin Sui, Shan He, Søren B Vilsen, Jinhao Meng, Remus Teodorescu, and Daniel-Ioan Stroe. A review of non-probabilistic machine learning-based state of health estimation techniques for lithium-ion battery. Applied Energy, 300:117346, 2021.

[24] Carl Kingsford and Steven L Salzberg. What are decision trees? Nature biotechnology, 26(9):1011–1013, 2008.

[25] Danny Varghese. Comparative study on classic machine learning algorithms. Retrieved July, 28:2021, 2018.

[26] Ali Haghpanah Jahromi and Mohammad Taheri. A non-parametric mixture of gaussian naive bayes classifiers based on local independent features. In 2017 Artificial intelligence and signal processing conference (AISP), pages 209–212. IEEE, 2017.

[27] Khadija Mohammad Al-Aidaroos, Azuraliza Abu Bakar, and Zalinda Othman. Naive bayes variants in classification learning. In 2010 international conference on information retrieval & knowledge management (CAMP), pages 276–281. IEEE, 2010.

[28] Berke Akkaya and Nurdan Çolakoğlu. Comparison of multi-class classification algorithms on early diagnosis of heart diseases. 2019.

[29] Tobias Appel et al. Cross-participant and cross-task classification of cognitive load based on eye tracking. PhD thesis, Eberhard Karls Universität Tübingen, 2021.

[30] J Brownlee. Essence of stacking ensemble for machine learning. https://machinelearningmastery. com/essence-of-stackingensembles-for-machine-learning/. Accessed on January, 24:2022, 2021.

[31] Hong Han, Xiaoling Guo, and Hua Yu. Variable selection using mean decrease accuracy and mean decrease gini based on random forest. In 2016 7th ieee international conference on software engineering and service science (icsess), pages 219–224. IEEE, 2016.

[32] Sean Stijven, Wouter Minnebo, and Katya Vladislavleva. Separating the wheat from the chaff: on feature selection and feature importance in regression random forests and symbolic regression. In Proceedings of the 13th annual conference companion on Genetic and evolutionary computation, pages 623–630, 2011.

[33] Kishan Maladkar. Why is random search better than grid search for machine learning. Analytics India Magazine, 2020.

[34] Danton S Char, Michael D Abràmoff, and Chris Feudtner. Identifying ethical considerations for machine learning healthcare applications. The American Journal of Bioethics, 20(11):7–17, 2020.

[35] Adrienne Yapo and Joseph Weiss. Ethical implications of bias in machine learning. Proceedings of the 51st Hawaii International Conference on System Sciences, Jan 2018.

[36] Classification: Accuracy nbsp;|nbsp; machine learning nbsp;|nbsp; google developers.

[37] Davide Chicco and Giuseppe Jurman. The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. BMC genomics, 21:1–13, 2020.

[38] Aayush Bajaj. What does your classification metric tell about your data?, Mar 2021.

[39] Andrius Vabalas, Emma Gowen, Ellen Poliakoff, and Alexander J Casson. Machine learning algorithm validation with a limited sample size. PloS one, 14(11):e0224365, 2019.

[40] Charis Demetriou, Lisi Hu, Toby O Smith, and Caroline B Hing. Hawthorne effect on surgical studies. ANZ Journal of Surgery, 89(12):1567–1576, 2019.

[41] Anjana Gosain and Saanchi Sardana. Handling class imbalance problem using oversampling techniques: A review. In 2017 international conference on advances in computing, communications and informatics (ICACCI), pages 79–85. IEEE, 2017.