Nottingham Trent University
Department of Computer Science

Applied Artificial Intelligence COMP40511 Coursework

A COMPREHENSIVE STUDY OF AI MODELS FOR MULTICLASS WINE QUALITY CLASSIFICATION

By

Lawrence Addae N1260744

03/06/24

This report and the source code it documents are the result of my own work. Any contributions to the work by third parties, other than tutors, are stated clearly below this declaration. Should this statement prove to be untrue I recognise the right and duty of the Board of Examiners to take appropriate action in line with the university's regulations on assessment.

Name: Lawrence Addae
Student Number: N1260744

Abstract

This study examines the effectiveness of many AI models for multiclass classification using a dataset regarding wine quality. There are 6,497 cases in the dataset, along with 11 features and an output attribute for wine quality. Support Vector Machines (SVM), Multilayer Perceptrons (MLP), Convolutional Neural Networks (CNN), K-means clustering, and Random Forest classifiers (RFC) were among the models whose performance were evaluated. According to the results, RFC had the highest accuracy (67%), closely followed by CNN (66%), MLP (58%), and SVM (52%). For three clusters, K-means clustering produced an accuracy of 32%.

Table of Contents

Introduction

Artificial Intelligence (AI), the discipline devoted to building robots that can execute activities requiring human intelligence, has transformed many sectors and is still pushing the frontiers of technological growth (Kolasani, 2024). Many people credit the Dartmouth Conference in 1956 as the catalyst for the development of artificial intelligence (AI), with inventors like Claude Shannon, John McCarthy, Marvin Minsky, and Nathaniel Rochester proposing the creation of robots that could replicate every aspect of human learning (Klondike, 2021).

Significant progress was made in spite of the "AI winters" setbacks, and as a result, a number of AI algorithms were developed (Glover, 2023). Rule-based and expert systems were among the pioneers in AI, and they dominated the field until the 1980s. The 21st-century AI boom may be attributed to developments in machine learning (ML), a branch of AI focused on creating algorithms that let computers analyse, learn from, and make choices based on data.

SVM is an effective supervised learning model that may be applied to regression and classification problems. The way it operates is by identifying the hyperplane that divides data into several groups the best. Artificial neural networks, such as MLPs, are frequently employed to solve pattern recognition and classification issues. CNNs are particularly good at image identification tasks because of their convolutional layers, which allow them to record spatial hierarchies in data. CNNs are inspired by the visual brain of animals (Dulari Bhatt, 2021).

K-means clustering is used to partition a dataset into k distinct clusters based on similarity. It is used in anomaly detection, picture compression, and consumer segmentation. To reduce overfitting and increase prediction accuracy, RFC combines many decision trees. By using concepts from computer science, statistics, mathematics, and cognitive science, these varied approaches highlight the multidisciplinary character of the discipline and highlight the range of applications of AI.

Project proposal

The goal of this study is to solve the challenge of forecasting wine quality using multiple physicochemical parameters. Accurate wine quality prediction is critical for both winemakers attempting to improve their production methods and consumers looking to make educated purchase decisions. The major goal is to create a strong AI model that properly assesses wine quality, providing significant information to both producers and consumers in the wine business.

The data for this study was obtained from the UCI repository (Paulo Cortez, 2009). This dataset contains comprehensive physicochemical parameters for both red and white wine, such as fixed and volatile acidity. The prediction model used a quality scale of 0 to 10 as its target variable. Combining the two datasets offers a larger sample size, which improves the robustness of the model.

To solve the prediction challenge, a clear and systematic technique was suggested. First, the data were pre-processed to verify they were suitable for modelling. Then, multiple machine learning techniques were used to estimate wine quality. Hence, finding the appropriate technique.

Pre-processing

To prepare the data, the datasets were combined to increase dataset size and improve model robustness. Next, missing values were handled because they might considerably hinder the performance of machine learning systems. Following that, characteristics with low variance were deleted since they do not fluctuate significantly between samples and might add noise into the models. As a result, their elimination increased the overall data quality. The data was then scaled to decrease discrepancies in feature magnitudes and guarantee that each feature contributes equally to the model's learning process.

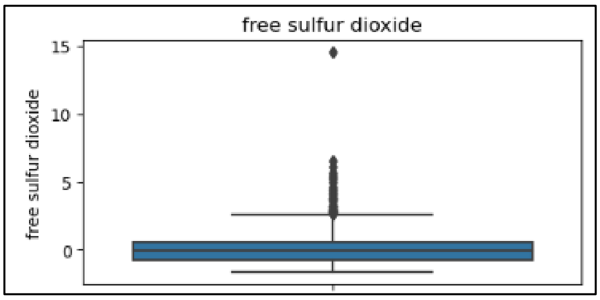

Figure 1 - Feature with outliers

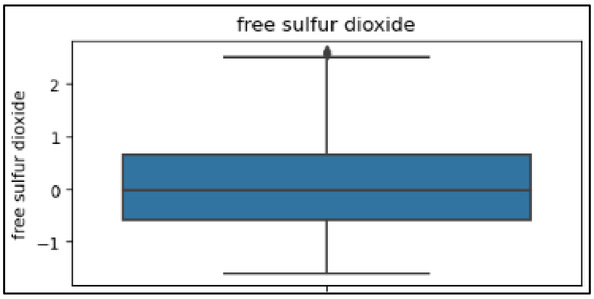

Figure 2 - Feature without outliers

Finally, the features were visualised to detect outliers, as shown in Figure 1. Outliers can bias findings and cause false forecasts; thus they were carefully deleted. As shown in Figure 2, this is an important step in improving the correctness and dependability of the data, and hence optimising the performance of the applied models.

SVM

To implement a multiclass classification model, a Support Vector Machine classifier from the scikit-learn library was chosen (scikit learn, 2024). The SVM model uses a linear kernel, which is appropriate for linearly separable data and frequently performs well in high-dimensional domains. However, given the dataset's complexity, investigating nonlinear kernels such as the radial basis function (RBF) might boost classification performance, particularly for underperforming classes. The penalty parameter 'C' was set to one, balancing the trade-off between low training and testing errors to prevent overfitting. A higher 'C' value may result in better management of class imbalance by prioritising misclassified points, whereas a lower 'C' value may avoid overfitting by allowing for more misclassifications.

The dataset was initially divided into training and testing sets using an 80-20 split to guarantee that the model could be assessed using previously unknown data. To guarantee that the findings would be reproducible, the random state was set to 42. The quality column was chosen as the goal variable, with the remaining attributes employed for training. The model obtains an overall accuracy of 52% for the test set, as shown in Figure 3.

	precision	recall	f1-score	support
5	0.00	0.00	0.00	37
7	0.55	0.53	0.54	295
4	0.51	0.79	0.62	441
6	0.00	0.00	0.00	195
accuracy			0.52	968
macro avg	0.26	0.33	0.29	968
weighted avg	0.40	0.52	0.44	968

Figure 1 - SVM performance

Despite the model's decent overall accuracy (52%), the gap in class-specific indicators indicates space for improvement. For example, classes 7 and 6 had reasonable F1-scores of 0.57 and 0.51, respectively, while the model struggled greatly with class 5. The overall average F1-score of 0.33 reflects this inconsistency and highlights the need for model changes or alternate techniques. Future work might include experimenting with other kernels, optimising hyperparameters, and resolving class imbalance in order to improve the model's overall efficacy and resilience.

MLP

To address the classification task using a neural network, the SVM model was modified to employ a Multilayer Perceptron classifier (scikit learn, 2024). The MLP model was built using the scikit-learn toolkit and has the following architecture: four hidden layers with 25, 18, 10, and 5 neurons each, allowing the model to gradually distil and abstract features from the input data. The activation function utilised was a rectified linear unit (ReLU), a common nonlinear activation function that introduces nonlinearity into the model, allowing it to learn more complicated patterns. Furthermore, it helps to avoid the vanishing gradient problem and speeds up the training process. The weight optimisation solver was set to 'adam,' a stochastic gradient-based optimizer that excels in dealing with the stochastic nature of gradient descent and achieving efficient convergence.

When tested on the test dataset, the MLP classifier attained an accuracy of 58%. When compared to the SVM model used in Task 1, which attained an accuracy of 52%, it is clear that the MLP model performs better. The SVM model, while successful, failed to grasp the data's complexity or MLP. This can be due to MLP's capacity to simulate complicated, non-linear interactions via its numerous layers and neurons, resulting in a more complete representation of data.

The higher accuracy and F1-scores of the MLP model suggest that it has a better generalization capability than the SVM model. However, because to the MLP model's complexity, careful hyperparameter adjustment is required, as is lengthier training durations. Furthermore, the performance of the MLP may be sensitive to the starting weights and specific design of the hidden layers, necessitating substantial testing for optimisation.

	precision	recall	f1-score	support
5	0.00	0.00	0.00	37
7	0.60	0.61	0.61	295
4	0.57	0.70	0.63	441
6	0.54	0.35	0.43	195
accuracy			0.58	968
macro avg	0.43	0.42	0.42	968
weighted avg	0.55	0.58	0.56	968

Figure 2 - MLP performance

The MLP model's strength resides in its adaptability and ability to learn complicated patterns, as demonstrated by its higher accuracy and F1-scores than the SVM model. However, MLP's performance is not consistent throughout all classes, as seen by low accuracy and recall in some courses, notably Class 5. This identifies a possible flaw in which the model suffers with class imbalance or inadequate data for specific classes.

Cross-Validation

```
Cross-validation scores: [0.46177686 0.45661157 0.52272727 0.51549587 0.51652893]
Mean cross-validation score: 0.4946280991735537
```

Figure 3 - Cross-validation

The average cross-validation score for the MLP classifier was 49.5% (Figure 5), which was slightly lower than the original test accuracy of 58%. This mismatch demonstrates that, while the model performed well on the original test set, its ability to generalise over other folds of data was restricted. In comparison, the SVM model from Task 1 had an accuracy of 52% without cross validation. While the SVM model exhibited slightly greater single-evaluation accuracy than the MLP cross-validation mean, it is worth noting that the MLP still provides comparable performance while also modelling complicated data interactions through its deep architecture.

The variation in cross-validation scores indicates that the model may be sensitive to the data split used for training and testing. This sensitivity can be addressed by fine-tuning the hyperparameters, utilising regularisation techniques to avoid overfitting, or using a bigger and more balanced dataset.

CNN

To improve the classification accuracy, the previous models were replaced with a deep CNN. The CNN was implemented using the Keras library, which is a high-level API for TensorFlow (TensorFlow, 2024). The CNN's design consists of three convolutional layers: the first comprises 32 filters, the second 64 filters, and the third 128 filters. All convolutional layers employed a kernel size of (2, 2) and the ReLU activation function, which produced nonlinearity and aided the network in learning difficult patterns. To downsample the feature maps and minimise spatial dimensions, MaxPooling layers with a pool size of (1, 1) are added after each convolutional layer. The dense layers consisted of 64, 128, and 256 neurons, with dropout layers in between to prevent overfitting. The last layer was a softmax layer with ten neurons, representing ten classes in the target variable.

The model was built with the 'adam' optimizer, which is notable for its ability to handle big datasets and sparse gradients. The loss function was set to 'categorical_crossentropy,' which is appropriate for multiple class classification. The model was trained for 1000 epochs, giving it plenty of time to learn, albeit this raised the computational cost with a validation split to assess performance on unseen data. The high dropout rates (0.5) utilised in the thick layers assist to avoid overfitting, but they may potentially impair the model's learning ability, indicating the need for fine-tuning.

The CNN model achieved an overall accuracy of 66% on the test set with 1000 epochs (51% for 300 epochs and 61% for 500 epochs), outperforming earlier models. The CNN performed significantly better in categorising classes 5, 6, and 7, with F1 values of 0.67, 0.70, and 0.57, respectively. However, the model struggled with class 4, yielding a poor F1-score of 0.21 (Figure 6).

	precision	recall	f1-score	support
4	0.50	0.14	0.21	37
5	0.67	0.68	0.67	295
6	0.63	0.78	0.70	441
7	0.78	0.45	0.57	195
accuracy			0.66	968
macro avg	0.64	0.51	0.54	968
weighted avg	0.67	0.66	0.65	968

Figure 4 - CNN performance

Table 1 - Models comparison

Model	**Accuracy**
SVM	52%
MLP	58%
CNN	66%

This table illustrates the incremental improvement in accuracy from SVM to MLP and then to CNN, illustrating the usefulness of deeper and more complicated models in classification tasks. Experimenting with different designs, adjusting hyperparameters, and addressing class imbalances may enhance model performance.

RFC

To further investigate the methods for improving the accuracy of our classification task, an RFC using the scikit-learn library was implemented (scikit learn, 2024). This is due to its capacity to prevent overfitting by averaging numerous decision trees, which improves the model's generalisability. The model was designed using 100 estimators, or 100 decision trees, to achieve a compromise between computational economy and model performance. To ensure consistency, the same training and testing sets as in prior models were employed.

RFC scored a total accuracy of 67% on the test set. The full classification report includes precision, recall, and F1-scores for each class. The model worked well for classes 6 and 5, but struggled with class 4, possibly due to class imbalance, as seen in Figure 7.

	precision	recall	f1-score	support
Class 4	0.75	0.16	0.27	37
Class 5	0.69	0.64	0.67	295
Class 6	0.63	0.80	0.71	441
Class 7	0.78	0.51	0.62	195
accuracy			0.67	968
macro avg	0.71	0.53	0.56	968
weighted avg	0.68	0.67	0.66	968

Figure 5 - RFC performance

Hyperparameter optimisation, which use techniques such as grid search or random search to find the optimal combination of parameters, can improve accuracy. Furthermore, resolving class imbalance using strategies such as oversampling, undersampling, or class weights in the model may enhance performance for underrepresented classes. Experimenting with different ensemble approaches, such as Gradient Boosting or XGBoost, might yield more insights and perhaps improve model performance.

K-means

To investigate clustering as a technique of enhancing classification accuracy, the K-means clustering algorithm was used with the scikit-learn toolkit. The dataset comprised ten unique classes, thus the initial experiment employed ten clusters. To initialise the centroids, the 'init' option was set to 'random', and the 'random_state' was set to 42, ensuring repeatability. The method was ran with a maximum of 300 iterations, and 'n_init' was set to 10 to run the process 10 times with various centroid seeds, selecting the best result in terms of inertia.

The initial results with ten clusters were disappointing, with an overall accuracy of only 10%. Precision, recall, and F1 ratings were extremely low across all courses, suggesting poor performance. This result implies that 10 clusters were inadequate for correctly arranging the data in a way that related to the actual classes.

	precision	recall	f1-score	support
Class 0	0.00	0.00	0.00	0
Class 1	0.00	0.00	0.00	0
Class 2	0.00	0.00	0.00	0
Class 3	0.00	0.00	0.00	0
Class 4	0.03	0.08	0.04	37
Class 5	0.31	0.11	0.16	295
Class 6	0.42	0.11	0.18	441
Class 7	0.08	0.04	0.05	195
Class 8	0.00	0.00	0.00	0
Class 9	0.00	0.00	0.00	0
accuracy			0.10	968
macro avg	0.08	0.03	0.04	968
weighted avg	0.30	0.10	0.14	968

Figure 6 - K-means 10 clusters

The Elbow Method (geeksforgeeks, 2023) and the Silhouette Score (scikit learn, 2024) were used to identify the best number of clusters. The Elbow Method is graphing the within-cluster sum of squares (WCSS) versus the number of clusters and determining the "elbow" point at which the rate of reduction abruptly decreases. According to this strategy, the best number of clusters might be three or four (Figure 8).

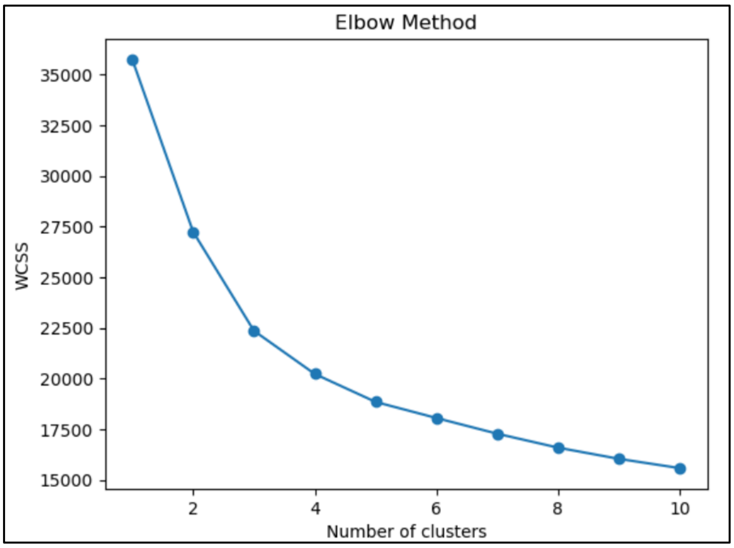

Figure 7 - Elbow method

This is further validated by calculating the Silhouette Scores for three and four clusters. The Silhouette Score measures how similar an object is to its own cluster compared with other clusters, with higher scores indicating better-defined clusters (Figure 9).

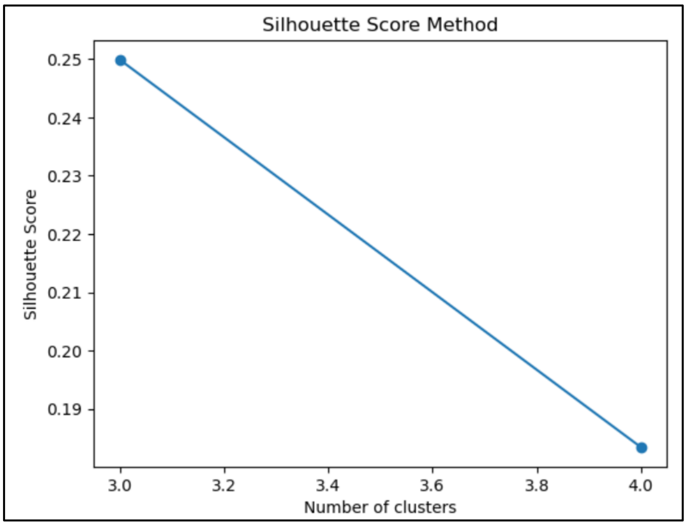

Figure 8 – Silhouette score

The findings showed that three clusters had the greatest Silhouette Score, implying that this was the ideal number of clusters. Thus, the number of clusters decreased to three, and the K-means cluster names were converted to the actual class labels. The accuracy increased to 32%, which, although still low, is a substantial improvement over the original 10-cluster model.

	precision	recall	f1-score	support
Class 4	0.04	0.38	0.08	37
Class 5	0.43	0.25	0.31	295
Class 6	0.46	0.50	0.48	441
Class 7	0.00	0.00	0.00	195
accuracy			0.32	968
macro avg	0.23	0.28	0.22	968
weighted avg	0.34	0.32	0.32	968

Figure 9 - K-means 3 clusters

This indicates that while clustering can capture some structures in the data, it is not as effective as supervised learning methods for this task.

The K-means model's main features are its simplicity and ability to give insight into the data structure without requiring labelled data. However, its shortcomings are clear in its low precision and recall ratings, especially for classes with limited assistance. The reliance on predetermined cluster numbers and the sensitivity to initiation are other significant drawbacks.

Future research might look into different clustering techniques, such as Gaussian Mixture Models or hierarchical clustering. Furthermore, integrating clustering and supervised techniques in a semi-supervised learning framework may capitalise on the characteristics of both approaches, perhaps leading to improved performance. Further adjustment of the clustering parameters, as well as the use of more advanced initialization approaches, may result in improvements.

Ethics

Transparency and accountability are critical ethical considerations in AI. AI systems, particularly those based on deep learning, frequently operate as "black boxes," making judgements that lack apparent explanations (Eschenbach, 2021). This opacity may be an issue in vital industries like healthcare, banking, and criminal justice, where knowing the reasoning behind a decision is critical. A lack of transparency can weaken confidence and responsibility, leading to scepticism and resistance from users. To address this, AI developers must prioritise explainability in their models, ensuring that stakeholders understand and scrutinise decision-making processes.

AI models frequently require a big quantity of data to work properly. Data collecting can raise serious privacy problems, especially when it contains sensitive personal information (Xusen Cheng, 2022). The risk of data breaches and unauthorised access to personal information can jeopardise privacy rights. Furthermore, data collection and analysis might have unexpected repercussions such as profiling and monitoring, which can be intrusive and discriminating. As a result, effective data protection measures and adherence to privacy legislation, such as the General Data Protection Regulation, are critical for protecting individuals' privacy.

AI systems might unintentionally perpetuate and exacerbate existing biases in training data (Natalia Norori, 2021). This prejudice can lead to unequal treatment and discrimination, especially among marginalised populations. Biased AI algorithms in recruiting procedures, for example, might penalise specific groups and prolong job inequity. AI developers must use bias detection and mitigation mechanisms to guarantee that their systems are fair and egalitarian. This involves diversifying training datasets and doing frequent audits to detect and correct biases.

The application of artificial intelligence in surveillance technology presents a severe danger to civil rights. AI-powered surveillance technologies, such as face recognition, have the potential to misuse authority and violate individuals' right to privacy and freedom (Denise Almeida, 2022). This is especially troubling under authoritarian countries, as such technology might be used to silence criticism and control people. To defend human rights and avoid exploitation of AI in surveillance, strict legislation and ethical principles must be established.

The growing dependence on AI for decision-making may reduce human agency and critical thinking. As AI systems do more activities, there is a concern that people may become unduly reliant on automated judgements, thereby reducing human control and judgement. This can be especially problematic in high-stakes situations when sophisticated human knowledge is required. To solve this, it is critical to create a balance between automation and human engagement, ensuring that AI complements rather than replaces human decision making.

References

Denise Almeida, K. S. E. L., 2022. The ethics of facial recognition technologies, surveillance, and accountability in an age of artificial intelligence: a comparative analysis of US, EU, and UK regulatory frameworks. *AI and Ethics,* Volume 2, p. 377–387.
Dulari Bhatt, C. P. H. T. J. P. R. V. S. P. K. M. a. H. G., 2021. *CNN Variants for Computer Vision: History, Architecture, Application, Challenges and Future Scope.* [Online]
Available at: https://www.mdpi.com/2079-9292/10/20/2470
[Accessed 10 May 2024].
Eschenbach, W. J. v., 2021. Transparency and the Black Box Problem: Why We Do Not Trust AI. *Philosophy & Technology,* Volume 34, pp. 1608-1621.
geeksforgeeeks, 2023. *Elbow Method for optimal value of k in KMeans.* [Online]
Available at: https://www.geeksforgeeks.org/elbow-method-for-optimal-value-of-k-in-kmeans/
[Accessed 22 May 2024].
Glover, E., 2023. *What Is AI Winter?.* [Online]
Available at: https://builtin.com/artificial-intelligence/ai-winter
[Accessed 7 May 2024].
Klondike, 2021. *AI history: the Dartmouth Conference.* [Online]
Available at: https://www.klondike.ai/en/ai-history-the-dartmouth-conference/
[Accessed 7 May 2024].
Kolasani, S., 2024. Unleashing Exponential Intelligence: Transforming Businesses through Advanced Technologies. *International journal of sustainable development through AI, ML and IoT,* 3(1), pp. 3-5.
Natalia Norori, Q. H. F. M. A. F. D. F. a. A. T., 2021. *Addressing bias in big data and AI for health care: A call for open science.* [Online]
Available at: https://www.cell.com/patterns/pdf/S2666-3899(21)00202-6.pdf
[Accessed 26 May 2024].
Paulo Cortez, A. C. F. A. T. M. J. R., 2009. *Wine Quality.* [Online]
Available at: https://archive.ics.uci.edu/dataset/186/wine+quality
[Accessed 15 May 2024].
scikit learn, 2024. *KMeans.* [Online]
Available at: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html
[Accessed 22 May 2024].
scikit learn, 2024. *MLPClassifier.* [Online]
Available at: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html
[Accessed 16 May 2024].
scikit learn, 2024. *RandomForestClassifier.* [Online]
Available at: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
[Accessed 20 May 2024].
scikit learn, 2024. *silhouette_score.* [Online]
Available at: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html
[Accessed 22 May 2024].

scikit learn, 2024. *Support Vector Machines.* [Online]
Available at: https://scikit-learn.org/stable/modules/svm.html
[Accessed 15 May 2024].
TensorFlow, 2024. *Convolutional Neural Network (CNN).* [Online]
Available at: https://www.tensorflow.org/tutorials/images/cnn
[Accessed 18 May 2024].
Xusen Cheng, L. S. X. (. L. J. B. a. S. C., 2022. The good, the bad, and the ugly: impact of analytics and artificialintelligence-enabled personal information collection on privacy andparticipation in ridesharing. *EUROPEAN JOURNAL OF INFORMATION SYSTEMS,* 31(3), p. 339–363.