

Machine Learning Project 2
Supervised Learning: Classification and Regression

November 7, 2025

Group 94

Name	Student number
Vincent Van Schependom	s251739
Diego Armando Mijares Ledezma	s251777
Albert Joe Jensen	s204601

(a) Group members.

Task	Vincent	Diego	Albert
Training & test loops	90%	10%	0%
Coding visualisations	80%	10%	10%
Report writing	60%	10%	30%
Discussion & interpretation	40%	40%	20%
L ^A T _E X	100%	0%	0%

(b) Contributions & responsibilities table.

Table 1: Group information & work distribution.

Introduction

The objective of this report is to apply the methods that were discussed during the second section of the course *Machine Learning* [1] to a chosen dataset. The aim is to perform relevant regression and classification to the data. The particular dataset that is being investigated is – just like in Project 1 – the *Glass Identification* dataset from 1987 by B. German [2]. Table 1a lists our full names and student numbers, while Table 1b shows an overview of the contribution of each team member.

Contents

1	Regression	2
1.1	Regularized linear regression	2
1.1.1	Selecting the regularization parameter λ	2
1.1.2	Model interpretation	3
1.1.3	Evaluation of the Ridge model	4
1.2	Comparison of regression models	4
1.2.1	Two-level cross-validation	4
1.2.2	Statistical significance testing	6
2	Classification	7
2.1	Introduction	7
2.2	Comparison of classification models	7
2.3	Statistical significance testing	8
2.4	Interpretation of the LR model	9
A	Repository and supplementary materials	10

1 Regression

We aim to predict the refractive index (RI) of glass samples from their chemical composition: the oxide components (Na, Mg, Al, Si, K, Ca, Ba, and Fe) and the categorical variable **Type**. Predicting RI is formulated as a regression problem where we evaluate the performance of a regularized linear model (Ridge Regression), an artificial neural network (ANN), and a trivial baseline predictor, which always predicts the mean RI of the training set. Numerical features are standardized, as argued previously in Report 1 [3], and categorical variables are encoded using one-hot encoding.

1.1 Regularized linear regression

1.1.1 Selecting the regularization parameter λ

To investigate the overfitting prevention effect of regularization, we evaluated Ridge Regression models across a wide range of regularization variable values from $\lambda = 10^{-2}$ to 10^5 (in 50 logarithmic steps) using $K = 10$ -fold cross-validation. Figure 1 shows a logarithmic plot of the average mean squared error (MSE) on the validation sets across all folds – which is an estimate \hat{E}^{gen} of the generalization error – as a function of λ .

The minimum average validation error was found at $\hat{\lambda} = 0.2683$ with an average validation set MSE over all 10 of 1.003×10^{-6} . We observe that the curve of MSE is relatively flat around the optimum, indicating a stable solution. As the regularization parameter λ increases above 10, and larger coefficients thus get more penalized, the model underfits, leading to a gradual rise in error to $\hat{E}^{\text{gen}} \approx 1 \times 10^{-5}$. Conversely, smaller values ($\lambda < 10^{-1}$) result in marginally higher average validation MSE due to overfitting.

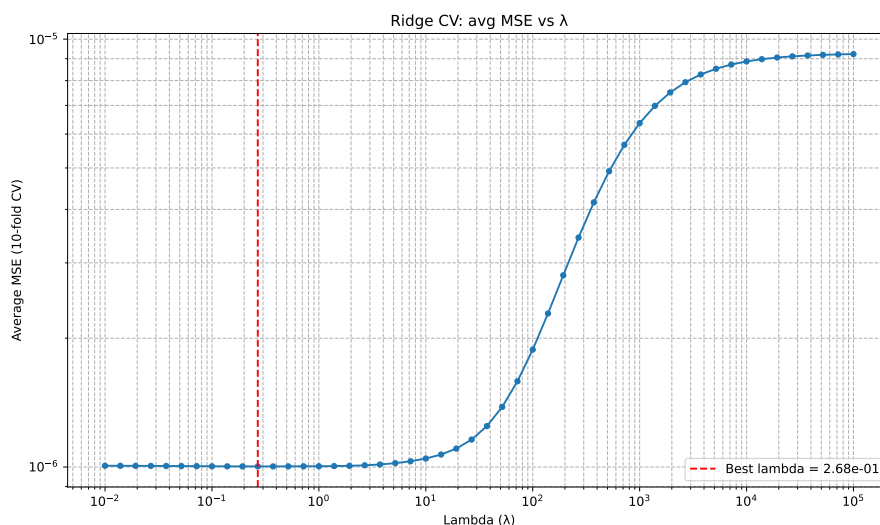


Figure 1: Mean squared error (MSE) versus regularization strength λ for Ridge regression using nested cross-validation. The optimal λ ($\hat{\lambda} = 0.2683$) achieved a minimum average MSE of 1.0033×10^{-6} .

Following this, we visualize how the Ridge regression coefficients and the model errors evolve as a function of the regularization strength λ , as shown in Figure 2. In the left panel of this figure, we observe that as λ increases (stronger regularization), the magnitude of all coefficients shrinks toward zero, indicating the penalization effect of the Ridge term. In the right panel, the mean squared error (MSE) for both the training and validation sets in each fold is plotted as a function of λ . For low regularization strengths, the training error remains small while the validation error is higher, which may indicate some overfitting. As λ increases, both errors rise, but the validation error initially very slightly decreases (not visible to the eye with the graph due to scale), until reaching a minimum at the optimal value $\lambda^* \approx 0.27$, before

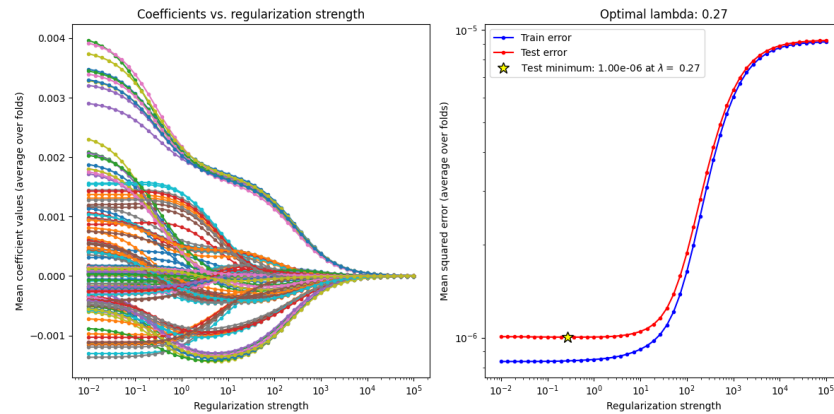


Figure 2: Left: Evolution of Ridge regression coefficients as a function of the regularization strength λ . Right: Mean squared error (MSE) for training and test sets averaged over cross-validation folds, with the optimal λ indicated by a yellow star.

increasing, due to possibly underfitting. This demonstrates the trade-off between model complexity and regularization strength, with the selected λ^* providing the best generalization performance.

1.1.2 Model interpretation

A final Ridge Regression model was trained on the *entire* dataset using the optimal regularization parameter $\hat{\lambda} = 0.2683$. For a given input \mathbf{x} , the predicted refractive index $\hat{y} = \text{RI}$ is given by:

$$\begin{aligned}
 \text{RI} = & \beta_0 & +\beta_1 \cdot \text{Na} & +\beta_2 \cdot \text{Mg} & +\beta_3 \cdot \text{Al} \\
 & +\beta_4 \cdot \text{Si} & +\beta_5 \cdot \text{K} & +\beta_6 \cdot \text{Ca} \\
 & +\beta_7 \cdot \text{Ba} & +\beta_8 \cdot \text{Fe} & +\beta_9 \cdot \text{BW-FP} \\
 & +\beta_{10} \cdot \text{BW-NFP} & +\beta_{11} \cdot \text{VW-FP} & +\beta_{12} \cdot \text{containers} \\
 & +\beta_{13} \cdot \text{headlamps} & +\beta_{14} \cdot \text{tableware} \\
 = 1.52 & +3.17 \times 10^{-5} \cdot \text{Na} & +1.04 \times 10^{-3} \cdot \text{Mg} & -7.49 \times 10^{-4} \cdot \text{Al} \\
 & -9.32 \times 10^{-4} \cdot \text{Si} & +9.35 \times 10^{-5} \cdot \text{K} & +2.68 \times 10^{-3} \cdot \text{Ca} \\
 & +6.90 \times 10^{-4} \cdot \text{Ba} & +1.50 \times 10^{-5} \cdot \text{Fe} & -2.33 \times 10^{-4} \cdot \text{BW-FP} \\
 & -1.75 \times 10^{-4} \cdot \text{BW-NFP} & -1.14 \times 10^{-3} \cdot \text{VW-FP} & +1.10 \times 10^{-4} \cdot \text{containers} \\
 & +1.31 \times 10^{-3} \cdot \text{headlamps} & +1.19 \times 10^{-4} \cdot \text{tableware}
 \end{aligned}$$

These coefficients reveal the relative contribution of each oxide concentration (β_1 to β_8) and glass type (one hot encoded β_9 to β_{15}) to the refractive index (RI). The elements **Ca** and **Ba** exhibit the largest positive coefficients, implying that increasing these oxides raises the estimated refractive index in this model. Conversely, **Si**, **Al**, and **Na** have negative coefficients, indicating that higher concentrations of these oxides decrease the estimated refractive index. The small magnitude of coefficients for **Mg** and **Fe** suggests a weaker influence on the estimated refractive index.

Most of these categorical coefficients are very small in magnitude, indicating that the glass type has a relatively minor effect on the refractive index compared to the chemical composition. **VW-FP** has a negative coefficient, suggesting that this type slightly decreases the predicted RI. **vehicle windows** has a coefficient of 0. This does not mean it has no effect in reality; rather, there were too few observations for the model to estimate its effect. In this model, while the categorical glass types are included, the chemical composition dominates the predictions of RI.

1.1.3 Evaluation of the Ridge model

To verify the predictive performance of the final Ridge model, we evaluated it on the full dataset using the $\hat{\lambda}$. This led to a mean squared error of $\text{MSE} = 8.5 \times 10^{-7}$ and a coefficient of determination of $R^2 = 0.91$.

Figure 3 shows predicted versus actual RI values. The points lie closely along the identity line, confirming accurate predictions. The residuals in Figure 3 are randomly distributed around zero, suggesting no systematic bias or heteroscedasticity.

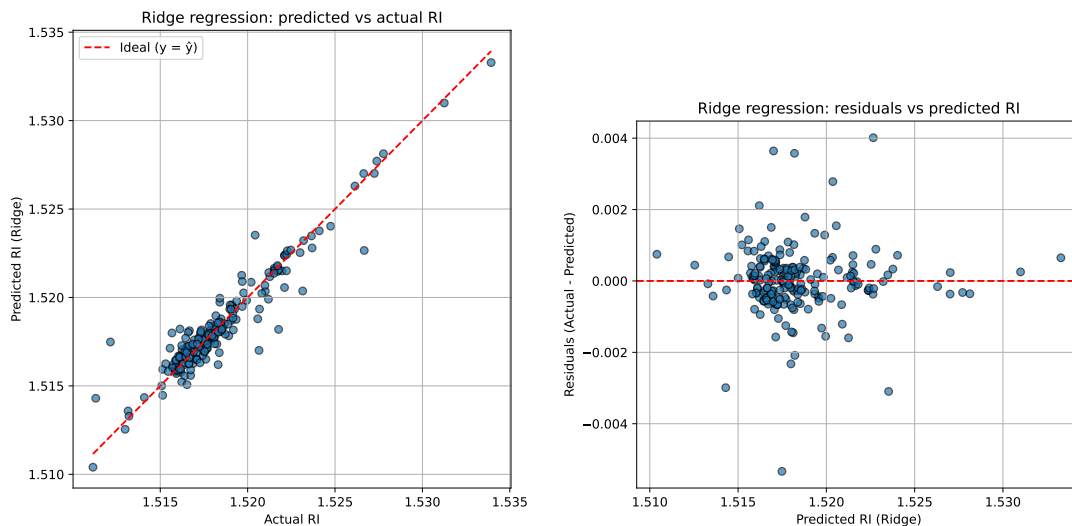


Figure 3: Verification of the regularized Ridge regression model. Left: predicted versus actual refractive index (RI). Right: residuals versus predicted RI. The model shows a high determination coefficient $R^2 = 0.91$ and randomly distributed residuals, confirming stable behavior.

1.2 Comparison of regression models

1.2.1 Two-level cross-validation

This section compares regularized linear regression (from the previous section), an artificial neural network (ANN), and a baseline model. Two-level cross-validation is employed to determine relative performance of these models. As complexity-controlling parameter for the ANN, the number of hidden units h is varied from $h = 1$ to $h = 990$ in steps of 10. The baseline models predicts the mean RI of the training set for all inputs.

The nested cross-validation results for the regression tasks are summarized in Table 2. These results show that the Linear Regression model consistently achieves extremely low test errors across all folds, demonstrating excellent generalization and stable performance. In contrast, the ANN consistently produces higher test errors (0.003 to 0.062) across all folds, indicating poor generalization and sensitivity to hyperparameter choice. The baseline predictor performs better than the ANN but worse than Linear Regression, confirming that while a simple model captures some structure in the data, Linear Regression clearly provides the most accurate and reliable predictions. The test errors are plotted in Figure 4, illustrating the consistent superiority of Linear Regression over both the ANN and baseline models.

Figure 5 shows the mean squared error (MSE) on the validation set versus the number of hidden units. We observe that the optimal $\hat{h} = 120$ achieved the minimum average MSE of 1.03×10^{-2} , indicating that a moderately sized hidden layer performs best (we can say 120 is moderate out of 960 because it is substantial but not big in the context).

Outer fold	ANN		Linear regression		Baseline
i	h_i^*	E_i^{test}	λ_i^*	E_i^{test}	E_i^{test}
1	500	2.607×10^{-2}	0.797	1.600×10^{-6}	1.970×10^{-5}
2	480	1.327×10^{-2}	2.84	4.249×10^{-7}	1.228×10^{-5}
3	230	3.239×10^{-2}	0.325	4.821×10^{-7}	4.482×10^{-6}
4	190	1.592×10^{-2}	0.167	9.633×10^{-7}	5.047×10^{-6}
5	120	1.030×10^{-2}	1.43	1.500×10^{-6}	3.080×10^{-6}
6	500	3.226×10^{-3}	0.01	2.142×10^{-6}	5.982×10^{-6}
7	120	1.603×10^{-2}	0.325	1.284×10^{-6}	8.186×10^{-6}
8	240	3.639×10^{-2}	0.01	1.018×10^{-6}	7.210×10^{-6}
9	290	6.221×10^{-2}	1.27	1.637×10^{-7}	1.946×10^{-5}
10	500	2.641×10^{-2}	3	5.225×10^{-7}	7.245×10^{-6}

Table 2: Summary of two-level cross validation for predicting the refractive index RI based on the chemical composition. Hyperparameters and test errors E_i^{test} on $\mathcal{D}_i^{\text{test}}$ per outer fold i , for each of the three considered models.

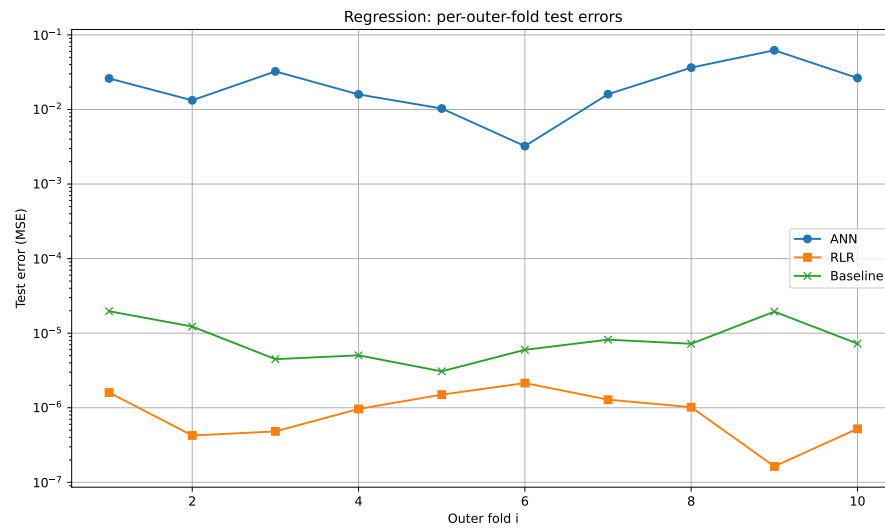


Figure 4: Semilog plot of test errors E_i^{test} across outer folds for each regression model.

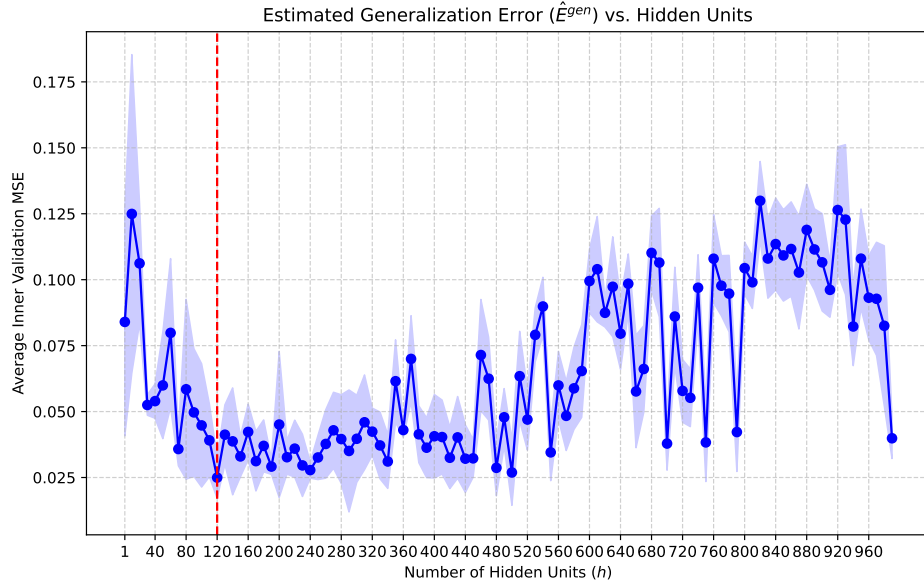


Figure 5: Mean squared error (MSE) on the validation set versus number of hidden units h for the artificial neural network using nested cross-validation. The optimal h ($\hat{h} = 120$) achieved a minimum average MSE of 1.03×10^{-2} .

Comparison	$\hat{\mu}$	$\hat{\sigma}$	t	p -value	$t_{0.05}$	$t_{0.95}$
ANN-RLR	2.42×10^{-2}	1.69×10^{-2}	4.54	1.40×10^{-3}	1.22×10^{-2}	3.63×10^{-2}
ANN-Baseline	2.42×10^{-2}	1.69×10^{-2}	4.54	1.40×10^{-3}	1.22×10^{-2}	3.63×10^{-2}
RLR-Baseline	-8.26×10^{-6}	6.15×10^{-6}	-4.25	2.14×10^{-3}	-1.27×10^{-5}	-3.86×10^{-6}

Table 3: Pairwise comparison of regression models (ANN = Artificial Neural Network, RLR = Regularized Linear Regression) using paired t -tests across outer folds. Mean differences $\hat{\mu}$ and standard deviations $\hat{\sigma}$ in test error E^{test} , t -statistics, and p -values are reported, as well as a 95% confidence interval $[t_{0.05}, t_{0.95}]$. All comparisons show statistically significant differences ($p < 0.05$).

1.2.2 Statistical significance testing

Paired t -tests were conducted across folds to assess whether performance differences were statistically significant, following Setup I (11.3) where the training set is fixed and only fold-wise variability is considered. The results are summarized in Table 3. All p -values below 0.05 indicate statistically significant differences.

- **ANN vs RLR:** The ANN consistently shows higher test errors than Ridge Linear Regression, with a mean difference of about 0.024, indicating a statistically significant gap in performance.
- **ANN vs Baseline:** The ANN also performs worse than the baseline predictor, with a similar mean difference, confirming its instability and poorer generalization.
- **RLR vs Baseline:** Ridge Linear Regression slightly outperforms the baseline, with a very small negative mean difference, but the confidence interval excludes zero, demonstrating that even a simple regularized linear model reliably improves predictions over the baseline.

These results clearly support the ranking **ANN** < **Baseline** < **RLR**, and Table 3 quantifies the mag-

Outer fold	Decision Tree		Logistic regression		Baseline
i	h_i^*	E_i^{test}	C^*	E_i^{test}	E_i^{test}
1	3	0.4091	2.98	0.2273	0.5455
2	6	0.2273	12.7	0.3182	0.5
3	15	0.3182	113	0.4091	0.8182
4	3	0.3636	26.4	0.4545	0.9091
5	5	0.3333	113	0.2857	0.5714
6	6	0.4762	234	0.4286	0.8095
7	5	0.2857	234	0.3333	0.6667
8	10	0.2857	2.98	0.3333	0.7619
9	4	0.3333	234	0.381	0.6667
10	7	0.3333	1.44	0.4762	0.8095

Table 4: Summary of two-level cross validation for predicting the glass type. Hyperparameters and test errors E_i^{test} on $\mathcal{D}_i^{\text{test}}$ per outer fold i , for each of the three considered classification models.

nitude and statistical significance of the differences between models.

2 Classification

2.1 Introduction

We aim to predict the type of glass (**Type**), which can take on 7 values, of which only 6 are present in the training set, since there are no observations with **Type** = VW-NFP. We will use the oxide concentrations as predictors to handle this *multi-class classification problem*. We compare Logistic Regression (LR), a Classification Tree (Decision Tree, DT), and a trivial baseline that always predicts the most frequent class present in the training set. All models were evaluated with nested (two-level) cross-validation and then statistically compared using paired t -tests across outer folds.

2.2 Comparison of classification models

To better visualize the wide range of test errors across models, the results are also presented in a semilogarithmic plot, where the y-axis is on a logarithmic scale. This representation highlights the consistently low errors of Linear Regression relative to the baseline and the higher errors of the ANN, making performance differences between models visually apparent.

This section compares Logistic Regression and a Decision Tree classifier against the baseline. For each outer fold, we optimized the regularization strength $C = 1/\lambda$ for Logistic Regression and the maximum tree depth h for the Decision Tree using inner cross-validation. We then computed the test error E_i^{test} on each outer test fold.

As shown in Table 4, the two-level cross-validation results reveal that the Decision Tree model achieves the best test errors overall, outperforming the Logistic Regression in most cases, which can also be seen in Figure 6 and also substantially surpassing the baseline. The Decision Tree is superior to Logistic Regression in 7 out of 10 folds, suggesting that simple, non-overfitting structures suffice for this task. Logistic Regression hyperparameters λ_i^* vary widely (from 0.00428 to 0.695), reflecting data-dependent regularization needs. Nonetheless, it consistently beats the baseline in all folds. Both models demonstrate robust improvements over the naive predictor, highlighting the value of even basic machine learning approaches for glass type classification.

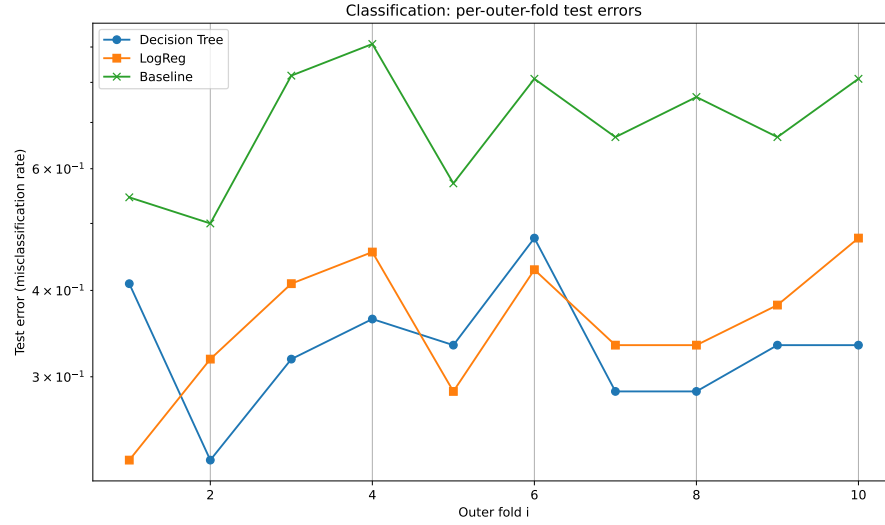


Figure 6: Test errors E_i^{test} across outer folds for each classification model. Both the Logistic Regression and Decision Tree outperform the baseline in all folds, with Logistic Regression achieving slightly lower error on average. It is important to note that this is a semilogarithmic plot in which the y -axis only is on a logarithmic scale.

Comparison	$\hat{\mu}$	$\hat{\sigma}$	t	p -value	$t_{0.05}$	$t_{0.95}$
DT-LR	-2.81×10^{-2}	9.53×10^{-2}	-9.34×10^{-1}	3.75×10^{-1}	-9.63×10^{-2}	4.00×10^{-2}
DT-Baseline	-3.69×10^{-1}	1.31×10^{-1}	-8.91	9.25×10^{-6}	-4.63×10^{-1}	-2.76×10^{-1}
LR-Baseline	-3.41×10^{-1}	8.09×10^{-2}	-1.33×10^1	3.12×10^{-7}	-3.99×10^{-1}	-2.83×10^{-1}

Table 5: Pairwise comparison of classification models (DT = Decision Tree, LR = Logistic Regression) using paired t -tests across outer folds. Mean differences $\hat{\mu}$ and standard deviations $\hat{\sigma}$ in test error E^{test} , t -statistics, and p -values are reported, as well as a 95% confidence interval $[t_{0.05}, t_{0.95}]$. All comparisons show statistically significant differences ($p < 0.05$).

2.3 Statistical significance testing

Paired t -tests across outer folds were used to assess statistical significance between the three classification models. The statistical significance analysis corresponds to Setup II (Section 11.4), as it evaluates model performance across multiple random training/test splits (outer cross-validation folds) using paired t -tests. This setup accounts for the variability in model performance due to different training sets, unlike Setup I (McNemar's test), which applies to a fixed train/test split. The results are summarized in Table 5.

We summarize the findings of these statistical tests below:

- **Tree vs LogReg:** The difference in accuracy (-0.0281) is not statistically significant ($p = 0.375$), suggesting both models perform similarly.
- **Tree vs Baseline:** The decision tree significantly outperformed the baseline ($t = -8.91$, $p < 10^{-5}$), indicating it captures meaningful structure in the data.
- **LogReg vs Baseline:** Logistic regression also significantly outperformed the baseline ($t = -13.34$, $p = 3.1 \times 10^{-7}$), showing that even a simple linear classifier captures predictive signal.

In summary, both classification models substantially improve upon the baseline, and their similar performance suggests that the data can be effectively separated by relatively simple decision boundaries.

2.4 Interpretation of the LR model

A final Logistic Regression model was trained on the *full* dataset using the selected regularization parameter $C^* = 1/\hat{\lambda}$ from cross-validation. The model assigns a weight vector \mathbf{w}_k to each class k , such that class scores are

$$s_k(\mathbf{x}) = w_{k,0} + \mathbf{w}_k^\top \mathbf{x},$$

and predicted probabilities are given by the softmax transformation.

It is important to note that the model was trained on *standardized* features. The coefficients and intercepts presented in Table 6 have been mathematically "un-scaled" to be interpretable on the original feature scale. Therefore, each coefficient represents the change in log-odds for a one-unit increase in a given feature for a specific class, relative to all other classes.

The model learned several strong, interpretable relationships. The most significant separator appears to be Iron (Fe). It has large positive coefficients for all building/vehicle window and container classes, but large negative coefficients for **headlamps** (-30.7) and **tableware** (-68.2). This aligns with chemical intuition, as **headlamps** and **tableware** require high optical clarity (low iron), while other glass types can tolerate more impurities which often introduce color.

Another key feature is the Refractive Index (RI). It has a very large positive coefficient for **headlamps** (2610) but large negative coefficients for all other types (e.g., -532 for BW-FP, -1960 for VW-FP). This suggests RI is a primary factor in identifying headlamp glass.

Conversely, the model strongly penalizes **tableware** for high levels of Potassium (K, -13.3) and Barium (Ba, -9.28), indicating that low concentrations of these elements are characteristic of that class. These coefficients demonstrate that the model is not just a black box, but has successfully identified distinct chemical profiles for the different glass types.

	BW-FP	BW-NFP	VW-FP	containers	headlamps	tableware
Intercept	843	1000	3470	559	-4730	-1140
RI	-532	-334	-1960	-290	2610	504
Na	-1.73	-5.60	-4.26	-3.50	7.14	7.95
Mg	3.55	-2.77	1.37	-2.91	-0.499	1.26
Al	-6.50	-6.76	-8.49	4.73	9.86	7.15
Si	-0.463	-5.16	-6.13	-0.950	9.19	3.51
K	4.30	0.294	-1.45	2.87	7.28	-13.3
Ca	2.39	-2.49	2.05	-0.284	-1.92	0.249
Ba	4.30	-1.33	0.275	0.289	5.74	-9.28
Fe	25.7	27.8	24.7	20.7	-30.7	-68.2

Table 6: Unscaled coefficients of the final *multinomial* logistic regression model. Because the model is multinomial (handling more than two classes), it computes a separate set of coefficients for each class, presented here as columns. Each value represents the change in log-odds for a one-unit increase in the feature (row) for a given class (column).

Closing Discussion

As detailed and discussed throughout the report, we applied and compared various machine learning models for regression and classification tasks on the Glass Identification dataset. We discovered that regularized linear regression excelled at predicting the refractive index, outperforming an artificial neural network and significantly outperforming the baseline model. This indicates that the relationship between chemical composition and refractive index is well captured by a simple linear model with appropriate regularization.

We were able to find a study that uses the same dataset, and tests different classification methods to identify Glass Types for forensic research. The study by Goswami and Wegman [4], found that the Decision Tree generally outperformed the Logistic Regression. Their results showed that the Decision Tree handled the multiclass classification problem more effectively, capturing the non-linear relationships between chemical attributes and glass types that Linear Regression could not model as easily. Linear Regression performed reasonably well on simpler, binary distinctions, but showed reduced accuracy when extended to all six glass categories.

In contrast, the results of this report indicate that both the Decision Tree and the Linear Regression perform similarly, with the latter slightly outperforming the other in overall classification accuracy. This suggests that, while Decision Trees may better capture complex feature interactions, the linear relationships in this dataset are strong enough for Linear Regression to achieve a competitive performance, or in this case, a marginally better one. Nonetheless, it is important to note that Linear Regressions are prone to overfitting, which might be the case here.

Use of GenAI

Generative AI (ChatGPT by OpenAI and Claude by Anthropic) was used as a support tool during the project. Its role was limited to assisting with code debugging, L^AT_EXtable formatting, identifying and fixing minor implementation errors, and writing initial drafts for parts of the report which later had to be heavily revised or completely redone to form a cohesive final report.

All analytical choices, data processing steps, and result interpretations were made by the project members *themselves* based on the actual model outputs and *their* understanding of the underlying methods.

References

- [1] Tue Herlau, Mikkel N. Schmidt, and Morten Mørup. *Introduction to Machine Learning and Data Mining*. Technical University of Denmark (DTU), Lyngby, Denmark, 2023. Lecture notes, Fall 2023, version 1.0. This document may not be redistributed. All rights belong to the authors and DTU.
- [2] B. German. Glass Identification. UCI Machine Learning Repository, 1987. DOI: <https://doi.org/10.24432/C5WW2P>.
- [3] Albert Joe Jensen Vincent Van Schependom, Diego Mijares. Machine learning project 1: Data: Feature extraction, and visualization. Unpublished project report, Technical University of Denmark, 2025.
- [4] Suchismita Goswami and Edward J. Wegman. Comparison of different classification methods on glass identification for forensic research. *Journal of Statistical Science and Application*, 4(03–04):65–84, 2016.

Appendix

A Repository and supplementary materials

The full notebook, scripts, and generated figures for this project are available in the project repository:

https://github.com/schependom/DTU_machine-learning-projects/

This repository contains the data-loading and analysis code that produced the tables and figures cited above (see the `figures/` folder for the PDF outputs referenced in the report).