Technical University of Denmark

Project 2 - Machine Learning on the Abalone Dataset: Two-Level Cross-Validation with Regression and Classification Models

Authors

Mathias Munck Nikolajsen - s204271

Jakub Bouzan - s251155

May 5, 2025

Author/Section	Regression A	Regression B	Classification	Discussion	Abstract
s204271	100%	100%	0%	50%	0%
s251155	0%	0%	100%	50%	100%

Table 1: Table of contributions

Contents

1 Abstract **2**

2 Regression **2**

2.1 Part A 2

2.2 Part B 5

3 Classification **7**

3.1 Introduction 7

3.2 Compare the three models in the classification problem 8

3.3 Statistically evaluation: setupI 11

4 Discussion and Conclusion **12**

5 Appendix **15**

1 Abstract

This report continues the analysis of the Abalone dataset, building on Project Report 1, which focused on data preprocessing, feature extraction, and visualization. The objective of this second report is to apply supervised learning methods—specifically regression and classification—to develop predictive models based on the physical characteristics of abalones.

In the regression task, we aimed to estimate the age of abalones by predicting the number of shell rings using features such as weight, length, and diameter. We compared a baseline model, regularized linear regression (RLR), and an artificial neural network (ANN).

For the classification task, we formulated a binary problem by excluding the "Infant" class and predicting sex (Male vs. Female). We compared a baseline majority-class predictor, logistic regression, and an ANN.

All models were evaluated using two-level cross-validation [2] to ensure robust comparison. Preprocessing included standardization of all features and encoding the target variable for classification.

2 Regression

2.1 Part A

The Abalone dataset aims to predict the age of abalones based on their physical measurements. The most appropriate variable to predict is "Rings" since it represents the age, which is typically determined by cutting the shell and counting the rings under a microscope. Predicting it using measurable attributes provides a non-destructive way to estimate age.

The regression model uses several predictor variables, including the categorical variable "Sex," which has three categories: Male, Female, and Infant. Other predictor variables include Length, Diameter, Height, Whole weight, Shucked weight, Viscera weight, and Shell weight, all of which are continuous numerical features.

To ensure effective model training, feature transformations are necessary. One-hot encoding was applied to the "Sex" variable in report 1, creating three separate binary features. Additionally, standardization was performed on all numerical features by subtracting the mean and dividing by the standard deviation to ensure each has a mean of zero and a standard deviation of one. This transformation helps stabilize training and improves regularization performance in linear regression.

By predicting the number of rings using these features, the model provides an estimation of the abalone's age without the need for invasive procedures.

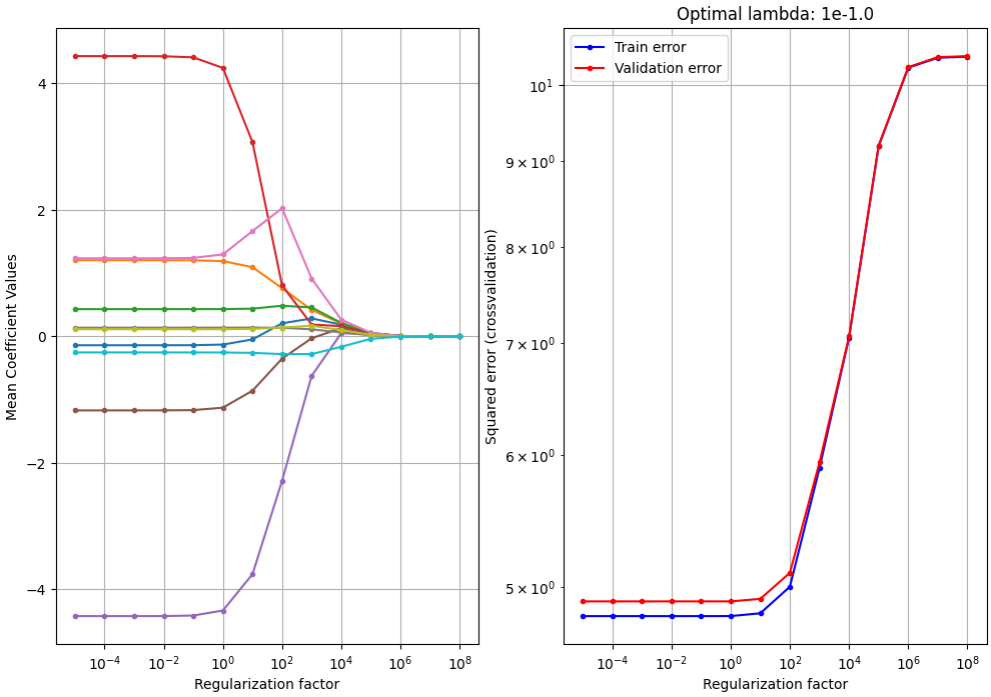

Figure 1: Training and test errors given regularization parameter values

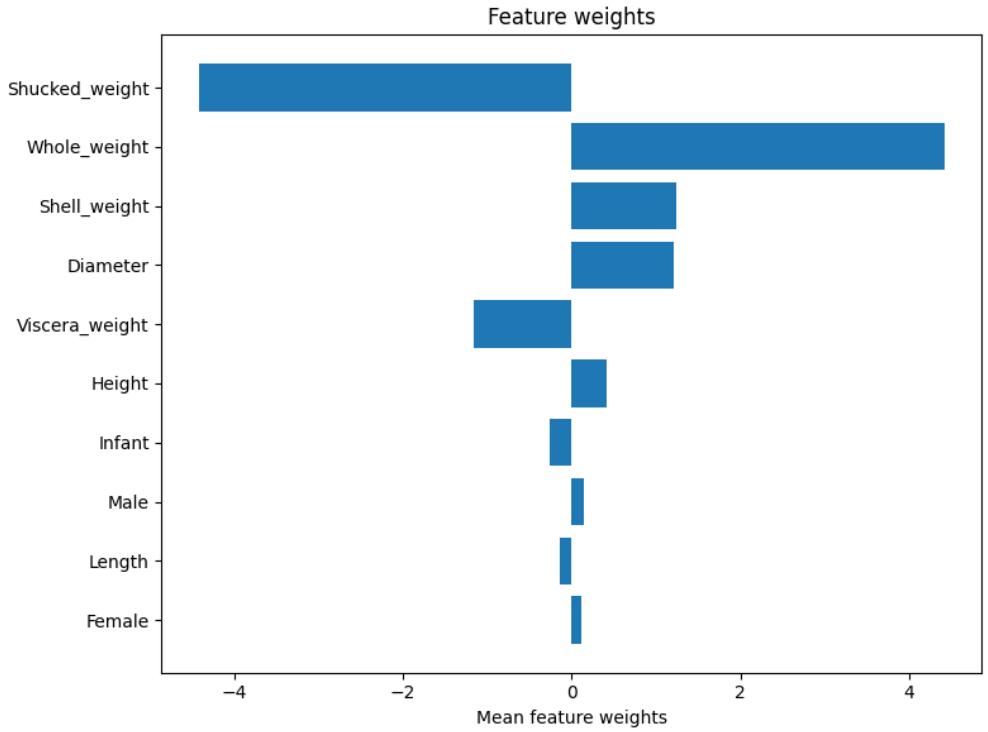

Figure 2: Mean feature weights for each attribute for optimal λ

The linear model with the lowest generalization error uses a regularization factor (λ) of $1e-1$. This value was selected because it resulted in the lowest validation (test) error during cross-validation, meaning it performs best on unseen data.

In this model, the output y is calculated as a weighted sum of the input features. Each input value is multiplied by a corresponding weight, and then a bias term is added. So, for a given input vector x, the predicted output is: $y = w0 + w1 \cdot x1 + w2 \cdot x2 + ... + wM \cdot xM$

Each weight w determines how much its associated feature influences the prediction. A positive weight increases the prediction when that feature increases, and a negative weight decreases the prediction.

From figure 2 showing feature weights:

- Whole_weight has a strong positive effect on the output — increasing this feature leads to a higher prediction.
- Shucked_weight has a large negative effect — increasing this feature lowers the prediction.
- Features like "Shell_weight", "Diameter", and "Viscera_weight" also have positive effects,

but not as strong.

- Other features like "Infant", "Male", "Female", and "Length" have very small weights, so they have little influence on the output.

These results make sense given the problem to predict age in abalones. Physical weight-related attributes are likely to be more informative than gender or size alone. So the model's behavior is reasonable and interpretable.

2.2 Part B

In this part, we implemented two-level cross-validation (Algorithm 6) [2], and also the loop structure illustrated in Figure [6] to compare different regression models on the Abalone dataset. This method helps us evaluate how well each model performs while also allowing us to tune their parameters in a structured and unbiased way. The goal was to compare three models: a baseline model (being a simple linear regression model), a linear regression model with a regularization factor λ, and an artificial neural network (ANN). The baseline model just predicts the mean value of the training data for all test examples. Even though it's basic, it's useful for comparison. If more advanced models can't perform better than this, they may not be worth the extra complexity.
Two-level cross-validation [2] means that the data is first split into outer folds (for testing), and then each of those is further split into inner folds (for training and validation). Both levels uses 10 folds (K1 = K2 = 10)
For the artificial neural network, we we used number of hidden units as the complexity parameter: We chose to test $h \in \{1, 2, 3, 4, 5\}$ and $\lambda \in \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$. These values were picked based on earlier experiments. Originally, we tested a wider range of values, but we narrowed it down to save time and focus on the most promising options. The results are shown in table 2 and 3.
The neural network was set up with two hidden layers, each using the ReLU activation function. For each outer fold, we trained several models using different combinations of h and λ on the inner training data. We then chose the combination that gave the lowest validation error and trained a final model using that setup on the full outer training set.

Outer fold	ANN		Linear regression		Baseline
i	h_i^*	E_i^{test}	λ_i^*	E_i^{test}	E_i^{test}
1	5	4.138	0.1	4.453	9.812
2	2	4.484	0.1	5.096	9.781
3	3	4.644	1.0	4.905	10.593
4	4	6.171	1.0	5.756	10.986
5	4	4.346	1.0	4.588	11.656
6	6	4.759	1.0	5.182	10.336
7	7	4.364	0.1	4.679	10.164
8	5	5.462	1.0	5.671	11.096
9	9	4.074	1.0	4.311	10.771
10	10	4.243	1.0	4.372	8.776

Table 2: Two-level cross-validation results comparing artificial neural network, linear regression, and baseline

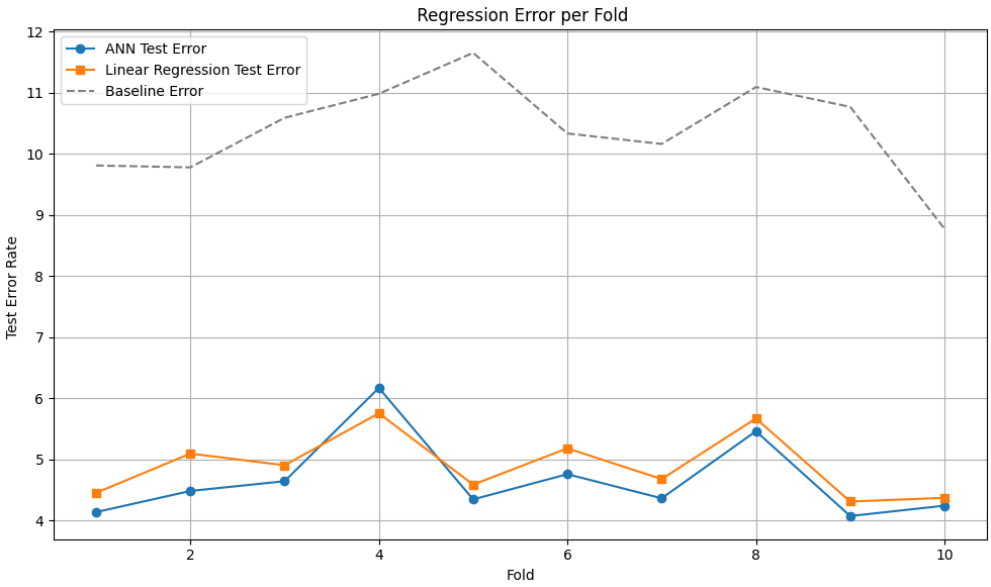

Figure 3: Comparison of test error between models

To statistically evaluate model performance, we conducted three pairwise comparisons between the artificial neural network (ANN), regularized linear regression (RLR), and the baseline model using paired t-tests [3]. Test errors were collected across 10 outer cross-validation folds. The null

hypotheses tested were: $H0 : \{ANN = LineaRegression; ANN = Baseline; LinearRegression = Baseline\}$

Comparison	**Mean Difference in Error**	**p-value**	**95% CI**
ANN vs. Linear Regression	-0.23	0.0210	[-0.42, -0.04]
ANN vs. Baseline	-5.73	3.57×10^{-9}	[-6.31, -5.15]
Linear Regression vs. Baseline	-5.50	3.56×10^{-9}	[-6.05, -4.94]

Table 3: Paired t-test results for model comparisons.

The ANN model performs significantly better than both the linear regression and the baseline models. Specifically, the mean difference in test error between ANN and RLR is -0.23 with a p-value of 0.021, indicating statistical significance at the 5% level. Similarly, the difference between ANN and the baseline is substantial (-5.73) and highly significant ($p < 10^{-8}$). The comparison between RLR and the baseline also yields a significant difference (-5.50, $p < 10^{-8}$).
The confidence intervals for all comparisons do not include zero, confirming statistically significant differences in model performance.

3 Classification

3.1 Introduction

In the classification part of this project, we aimed to predict the sex of abalones based on their physical characteristics. Although the "Rings" attribute—representing age—is included in the dataset and was appended to the feature matrix X for reference, it was not used as the target for classification. Instead, we focused on the "Sex" attribute, which originally includes three categories: Male, Female, and Infant. To formulate a binary classification task, the "Infant" category was removed due to its ambiguous biological meaning. The remaining two classes were then encoded as Male = 0 and Female = 1, forming the binary target variable.
The prediction was based on a set of continuous numerical features including Length, Diameter, Height, Whole weight, Shucked weight, Viscera weight, Shell weight, and Ring, which describe measurable physical characteristics of each abalone. After extracting the binary class label, the original categorical "Sex" feature was removed from the input feature set. To standardize the

data and improve model performance, especially when using regularization—we applied z-score normalization to all features in the input matrix X: each column was transformed to have zero mean and unit variance by subtracting the mean and dividing by the standard deviation. The goal of the classification task was to build predictive models that generalize well to unseen data in identifying the abalone's sex. To compare models and tune hyperparameters, we implemented two-level cross-validation (Algorithm 6) [2] in Figure [5]. Three models:

- A baseline model predicting the majority class from the training set.
- A logistic regression model with complexity controlled by the regularization parameter λ.
- An artificial neural network (ANN), with complexity controlled by the number of hidden units (h).

3.2 Compare the three models in the classification problem

To compare model performance, we implemented the **two-level cross-validation** procedure outlined in Algorithm 6 [2], and also the loop structure illustrated in Figure [6].

- **Outer:** For each of the 10 outer folds, the dataset was split into an outer training set D_i^{par} and an outer test set D_i^{test}. The outer fold was used to estimate the generalization error of the final selected model.
- **Inner:** Inside each outer fold, the outer training set D_i^{par} was further split into 5 inner folds. These inner splits were used solely to tune hyperparameters for each model.
- **Parameter:**
  - For **logistic regression**, the complexity was controlled by the regularization strength $\lambda \in \{10^{-5}, 10^{-4}, \ldots, 10^{0}\}$.
  - For the **artificial neural network (ANN)**, the complexity was controlled by the number of hidden units $h \in \{1, 2, 3, 4, 5\}$.

    For each parameter setting, models were trained on inner training sets and evaluated on inner validation sets. The average inner validation error was computed for each value of λ and h.

- **Retrain:** After identifying the best-performing parameter value (either λ_i^* or h_i^*) within each outer fold, we retrained the corresponding model on the **full outer training set** D_i^{par}.
- **Evaluate:** Finally, the retrained model was tested on the **outer test set** D_i^{test}, and the resulting test errors were recorded. The same outer split was reused across all three models—logistic regression, ANN, and a baseline.

The results of the two-level cross-validation are summarized in Figure 4, Table 4, and Table 5 . The Figure 4 visually displays the test errors across all outer folds for the three models—logistic regression, ANN, and a baseline. Table 4 presents their average test error rates, while Table 5 breaks down individual fold performance and the selected hyperparameters.

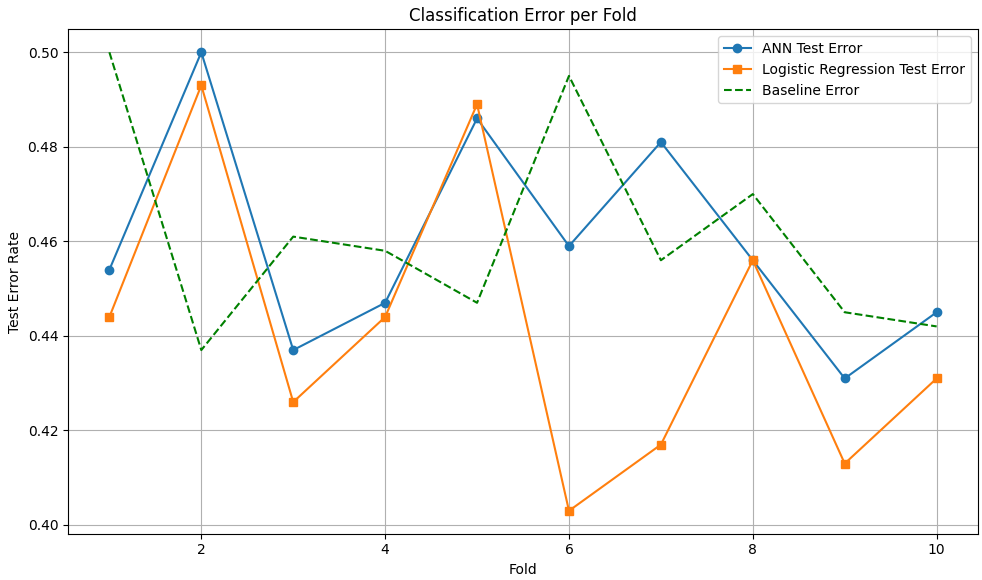

Figure 4: Test error across all folds for each model.

Model	**Average** E^{test}
Logistic Regression (LR)	0.4416
Artificial Neural Network (ANN)	0.4596
Baseline	0.4611

Table 4: Average test errors (E^{test}) across all folds for each model.

From Table 4, we observe that logistic regression (LR) achieves the lowest average test error at

0.4416, slightly outperforming both the artificial neural network (ANN) at **0.4596** and the baseline model at **0.4611**. This trend is also reflected in Figure 4 , where LR consistently performs slightly better across folds.

Outer fold	ANN		Logistic regression		Baseline
i	h_i^*	E_i^{test}	λ_i^*	E_i^{test}	E_i^{test}
1	2	0.454	1.0	0.444	0.500
2	2	0.500	1.0	0.493	0.437
3	1	0.437	1.0	0.426	0.461
4	3	0.447	1.0	0.444	0.458
5	1	0.486	1.0	0.489	0.447
6	3	0.459	1.0	0.403	0.495
7	4	0.481	1.0	0.417	0.456
8	2	0.456	1.0	0.456	0.470
9	4	0.431	0.001	0.413	0.445
10	3	0.445	1.0	0.431	0.442

Table 5: Two-level cross-validation table used to compare the three models in the classification problem (artificial neural network, Logistic regression, and baseline

Looking at Table 5, we see that:

- For **ANN**, the number of hidden units selected varies across folds (e.g., $h^* = 1$ to $h^* = 4$), indicating no universally optimal architecture.
- For **logistic regression**, the optimal λ^* is 1.0 in most folds, with one exception (fold 9) where $\lambda^* = 0.001$ was selected. This suggests a relatively stable preference for stronger regularization across folds.
- The **baseline model**, which predicts the majority class in each fold, yields the highest error in most cases, confirming its limited utility for this task.

Overall, the results indicate that logistic regression is the most reliable model in terms of both consistency and generalization, with the baseline model performing worst, and the ANN offering variable but occasionally strong performance depending on the chosen architecture.

3.3 Statistically evaluation: setupI

To evaluate whether the observed differences in model performance are statistically significant, we conducted pairwise comparisons using **McNemar's test [4]**. This test examines whether two classifiers differ in how often they make errors on the same test instances. The results of the statistical tests are presented in Table 6, which includes the estimated difference in error rates ($\hat{\theta}$), the corresponding 95% confidence intervals, and p-values for each pair of models.

Comparison	**Estimate $\hat{\theta}$**	**p-value**	**95% CI**
ANN vs Logistic Regression	-0.0180	0.0099	$[-0.0314, -0.0046]$
ANN vs Baseline	$+0.0014$	0.9227	$[-0.0200, +0.0228]$
Logistic Regression vs Baseline	$+0.0194$	0.0707	$[-0.0012, +0.0400]$

Table 6: McNemar's test results for pairwise classifier comparisons.

For the comparison between **ANN and logistic regression**, we observe an estimated difference of $\hat{\theta} = -0.0180$, with a confidence interval of $[-0.0314, -0.0046]$ and a p-value of **0.0099**. Since the confidence interval does not include zero and the p-value is below the 0.05 significance threshold, we conclude that logistic regression significantly outperforms the ANN in this classification task.
In contrast, the comparison between **ANN and the baseline** $\hat{\theta} = +0.0014$, with a wide confidence interval $[-0.0200, +0.0228]$ and a very high p-value of **0.9227**. This indicates that there is no statistically significant difference between the ANN and the baseline model, suggesting that the ANN did not consistently perform better than the baseline.
Finally, the comparison between **logistic regression and the baseline** shows $\hat{\theta} = +0.0194$, with a confidence interval $[-0.0012, +0.0400]$ and a p-value of **0.0707**. Although the mean difference favors logistic regression, the confidence interval slightly overlaps zero, and the p-value does not meet the 0.05 threshold. Therefore, we cannot conclude with statistical confidence that logistic regression outperforms the baseline, even though the trend is in that direction.
So, logistic regression is the only model that statistically outperforms model (ANN). The ANN, despite its flexibility, does not show significant improvement over the baseline, while logistic regression demonstrates stronger and more consistent performance. These results support the conclusion that logistic regression is the most effective and reliable model for the task of predicting abalone sex based on physical attributes.

4 Discussion and Conclusion

This project explored two supervised learning problems using the Abalone dataset: a regression task to estimate the age of abalones (measured by the number of shell rings), and a classification task to predict the sex of an abalone based on its physical attributes.

In the **regression part**, we compared three models: a baseline model that predicts the mean target value, a regularized linear regression model (RLR), and an artificial neural network (ANN). All models were evaluated using two-level cross-validation [2]. The ANN consistently achieved the lowest test errors, indicating strong generalization performance. The ANN outperformed the baseline and RLR model in nearly all folds, demonstrating that its additional complexity can be beneficial. However the RLR had great performance and clearly outperformed the baseline .

In the **classification part**, we formulated a binary classification task by removing the "Infant" category from the original three-class "Sex" attribute and encoding the remaining classes as Female = 1 and Male = 0. We compared logistic regression (LR), ANN, and a baseline majority-class predictor using the same two-level cross-validation [2] approach. Logistic regression achieved the lowest average test error (0.4416), followed closely by the ANN (0.4596) and the baseline (0.4611). Statistical comparison using McNemar's test confirmed that logistic regression significantly outperforms the ANN, while neither model showed statistically significant improvement over the baseline.

Overall, the classification results indicate that logistic regression is the most reliable model, combining consistent generalization performance with statistical significance. The ANN, while occasionally competitive, did not offer a clear performance advantage and was more sensitive to hyperparameter settings.

The abalone dataset is a common machine learning dataset that have previously been used to test diffrent kind of models. To contextualize our results, we compared them with a related study by Hassanzadeh et al. (2019) [1], which also used the UCI Abalone dataset. That study focused on predicting abalone age using a regression-based artificial neural network (ANN) model and reported a root mean squared error (RMSE) of 2.0691.

While our ANN model did not consistently reach that level of accuracy across all folds, it approached it in some (e.g., folds 7,10), suggesting that ANNs can be effective when carefully tuned. However, their performance may be less stable due to sensitivity to hyperparameters and data variance. Importantly, our results showed that simpler models like regularized linear regression (RLR) can match

or even outperform ANNs—especially when training data is limited or when model interpretability and consistency are prioritized.

In conclusion, this project showed that simpler models can outperform more complex ones when applied thoughtfully. In regression, ANN delivered the lowest test errors, outperforming both RLR and baseline models. For classification, logistic regression was the most reliable model and significantly outperformed the ANN. Although ANNs showed potential, their performance varied and required careful tuning. Compared to prior work, our results confirm that simpler models like RLR can achieve comparable or better performance. These findings underscore the value of regularization, model selection, and robust cross-validation in building effective predictive models.

References

[1] A. Hassanzadeh et al. Prediction of abalone age using regression-based neural network. *Prediction of Abalone Age Using Regression-Based Neural Network*, 2019.

[2] Tue Herlau, Mikkel N Schmidt, and Morten Mørup. *Introduction to Machine Learning and Data Mining: Lecture Notes, Fall 2023*. Technical University of Denmark., 2023. 177 pages (Algorithm 6).

[3] Tue Herlau, Mikkel N Schmidt, and Morten Mørup. *Introduction to Machine Learning and Data Mining: Lecture Notes, Fall 2023*. Technical University of Denmark., 2023. 215 pages (Comparing two regression models).

[4] Tue Herlau, Mikkel N Schmidt, and Morten Mørup. *Introduction to Machine Learning and Data Mining: Lecture Notes, Fall 2023*. Technical University of Denmark., 2023. 209 pages- (McNemartest).

5 Appendix

Algorithm 6: Two-level cross-validation

Require: K_1, K_2, folds in outer, and inner cross-validation loop respectively
Require: $\mathcal{M}_1, \ldots, \mathcal{M}_S$: The S different models to cross-validate
Ensure: $\hat{E}^{\text{gen}}$, the estimate of the generalization error
for $i = 1, \ldots, K_1$ **do**
Outer cross-validation loop. First make the outer split into K_1 folds
Let $\mathcal{D}_i^{\text{par}}$, $\mathcal{D}_i^{\text{test}}$ be the i'th split of $\mathcal{D}$
for $j = 1, \ldots, K_2$ **do**
Inner cross-validation loop. Use cross-validation to select optimal model
Let $\mathcal{D}_j^{\text{train}}$, $\mathcal{D}_j^{\text{val}}$ be the j'th split of $\mathcal{D}_i^{\text{par}}$
for $s = 1, \ldots, S$ **do**
Train $\mathcal{M}_s$ on $\mathcal{D}_j^{\text{train}}$
Let $E_{\mathcal{M}_s,j}^{\text{val}}$ be the *validation error* of the model $\mathcal{M}_s$ when it is *tested* on $\mathcal{D}_j^{\text{val}}$
end for
end for
For each s compute: $\hat{E}_s^{\text{gen}} = \sum_{j=1}^{K_2} \frac{|\mathcal{D}_j^{\text{val}}|}{|\mathcal{D}_i^{\text{par}}|} E_{\mathcal{M}_s,j}^{\text{val}}$
Select the optimal model $\mathcal{M}^* = \mathcal{M}_{s^*}$ where $s^* = \arg\min_s \hat{E}_s^{\text{gen}}$
Train $\mathcal{M}^*$ on $\mathcal{D}_i^{\text{par}}$
Let E_i^{test} be the *test error* of the model $\mathcal{M}^*$ when it is *tested* on $\mathcal{D}_i^{\text{test}}$
end for
Compute the estimate of the generalization error: $\hat{E}^{\text{gen}} = \sum_{i=1}^{K_1} \frac{|\mathcal{D}_i^{\text{test}}|}{N} E_i^{\text{test}}$

Figure 5: Algorithm 6:Two-level cross-validation

```
For each outer fold:
    Split data into outer training set (D^par_i) and test set (D^test_i)

    For each inner fold:
        Split D^par_i into inner training and validation sets

        For each model parameter (e.g., lambda for RLR or h for ANN):
            Train model on inner training set
            Evaluate on inner validation set

    Select the best parameter (lambda* or h*) based on average inner validation error
    Retrain the model on full outer training set (D^par_i) using the best parameter
    Evaluate on outer test set (D^test_i)
```

Figure 6: Two-Level Cross-Validation Loop Structure for Model Selection and Evaluation