Danmarks
Tekniske
Universitet

02450 Introduction to Machine Learning and Data Mining - Project 2

Authors

Maxime Swagel - s242033
Santiago Londoño - s250161
Boon Keng Leck - s247301

We confirm that this report is our own independent and original work.
The allocation of tasks is detailed in the table below.

Section	Contribution (%)		
	Maxime Swagel	Santiago Londoño	Boon Keng Leck
Regression	20	30	50
Classification	50	30	20
Discussion	20	60	20

April 10, 2025

Contents

Introduction

This report continues the analysis of the South African Heart Disease Study dataset, expanding on the exploratory work presented in the first part of the project. In the initial phase, we investigated the structure of the dataset, analyzed the distributions of individual variables, and examined their relationships with the target outcomes. These insights provided a foundation for selecting relevant features and understanding the data's underlying patterns, which informed the modeling strategies in this second phase.

In this part of the project, we apply machine learning techniques to perform both classification and regression tasks. Our goal is to build predictive models that can accurately classify the presence of heart disease and estimate the associated risk levels. We perform data pre-processing, implement dimensionality reduction methods, and evaluate various models using cross-validation and test error metrics. We aim to identify robust approaches for both tasks while reinforcing key concepts such as model selection, performance evaluation, and the balance between bias and variance.

1 Regression: Part A

This section focuses on addressing a relevant regression problem within the dataset by modeling a continuous target variable based on the available features. To initiate the analysis, we explore the simplest yet foundational approach, linear regression. This serves as a baseline model to assess the dataset's linear relationships and to establish a reference point for further comparison with more complex regression techniques. The model's performance will be evaluated using appropriate statistical measures to determine its effectiveness and limitations in capturing the underlying patterns of the data.

1.1 Explain what variable is predicted based on which other variables and what you hope to accomplish by the regression. Mention your feature transformation choices such as one-of-K coding. Since we will use regularization momentarily, apply a feature transformation to your data matrix X such that each column has mean 0 and standard deviation 1.

We aim to predict Type A behaviour ('typea') as the dependent variable in our regression task. Our objective is to identify and understand the factors that most significantly influence Type A behaviour. To briefly recap, Type A behaviour is a continuous variable derived from the Bortner Short Rating Scale, which assesses coronary-prone behaviour. In essence, the higher the typea value, the greater the risk of coronary-related conditions. Prior to modeling, we perform Z-score normalization on all numerical features, excluding the categorical variable 'famhist' so as to ensure that all variables contribute equally to the regression model and prevent features with larger scales from dominating.

1.2 Introduce a regularization parameter λ as discussed in chapter 14 of the lecture notes, and estimate the generalization error for different values of λ. Specifically, choose a reasonable range of values of λ (ideally one where the generalization error first drop and then in creases), and for each value use K = 10 fold cross-validation (algorithm 5) to estimate the generalization error. Include a figure of the estimated generalization error as a function of λ in the report and briefly discuss the result.

As covered in the lecture notes, the parameter λ is used as an L2 regularization term, which means we treat our model as a ridge regression model. This type of regularization helps reduce overfitting by penalizing large weights in the model. To find the best value for λ, we test a range of values from 10^{-5} to 10^{5} and perform 10-fold cross-validation to determine which one gives us the lowest generalization error, measured by mean squared error (MSE).

From the results shown in Figure 1, we find that a λ value around 10^2 gives the lowest MSE. This suggests that regularization is effectively reducing the impact of less important features, something we already suspected based on our previous analysis, where we observed that a few features did not contribute much to the model's variance. By penalizing those features, the model becomes simpler and more accurate on new, unseen data. From figure 1 we can see how the lowest mean squared error (MSE) of 95.49 was achieved with an optimal λ value of 1.47 x 10^2. As shown in the left plot of Figure 1, the MSE decreases as λ increases, reaching a minimum at $\lambda = 1.47 \times 10^2$. Beyond this point, the error starts increasing again due to excessive regularization. This U-shaped curve confirms the need for tuning λ to avoid underfitting or overfitting. The right plot illustrates how the coefficients for each feature vary with different λ values. As λ increases, coefficients are gradually shrunk toward zero. This behavior confirms that ridge regression penalizes less relevant features, which helps stabilize the model. Notably, features such as `obesity`, `famhist`, and `age` retain relatively higher magnitudes, suggesting they are more influential in predicting Type A behavior. These plots together provide strong evidence that ridge regression not only improves prediction performance but also aids in identifying the most important features by controlling model complexity.

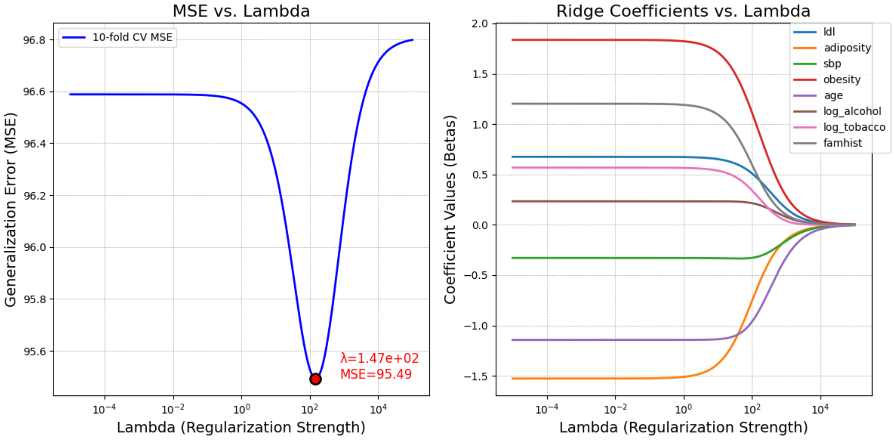

Figure 1: Ridge Coeficient vs Lambda

1.3 Explain how the output, y, of the linear model with the lowest generalization error (as determined in the previous question) is computed for a given input x. What is the effect of an individual attribute in x on the output, y, of the linear model? Does the effect of individual attributes make sense based on your understanding of the problem?

Based on the optimal value of $\lambda = 1.47 \times 10^2$, which yields the lowest MSE of 95.49, we obtain the coefficients shown in Table 1. All numerical features are standardized using Z-score normalization, while `famhist` is treated as a binary categorical variable. Additionally, `tobacco` and `alcohol` are log-transformed to address skewness in their original distributions.

From the resulting coefficients, we observe that `obesity`, `famhist`, `ldl`, `tobacco`, and `alcohol` have positive weights, suggesting a direct relationship with higher Type A behavior scores. This aligns with prior knowledge indicating these factors are associated with increased risk of coronary heart disease, which Type A behavior is often linked to. Interestingly, `sbp` (systolic blood pressure), `age`, and `adiposity` have negative coefficients. This contradicts the general expectation that higher values of these features would contribute to increased Type A behavior. One possible explanation lies in multicollinearity among the features, which we previously explored in our initial report through correlation analysis. We addressed this by performing Principal Component Analysis (PCA) which will not be detailed in this report and select a number of components that together explain at least 90% of the total variance. When training the ridge regression model using these components, we observed that the resulting MSE remained comparable to the model trained on all original features. Therefore, we proceed using both the full feature model and the PCA-transformed model in subsequent evaluations.

Feature Name	**Beta (Coefficient)**
Obesity	0.950328
Age	-0.871999
Adiposity	-0.610069
Famhist (binary)	0.475680
SBP	-0.316495
LDL	0.457567
Tobacco (log-transformed)	0.277540
Alcohol (log-transformed)	0.200790

Table 1: Ridge regression coefficients for each feature at $\lambda = 1.47 \times 10^2$.

2 Regression: Part B

In this section, we aim to evaluate and compare the performance of three different regression models: the regularized linear regression model from the previous section, an artificial neural network (ANN), and a simple baseline model. The objective is to assess not only whether one predictive model outperforms another, but also whether either model significantly improves upon a trivial baseline. To ensure a fair and robust comparison, we employ two-level cross-validation, which allows for both model evaluation and hyperparameter tuning while minimizing the risk of overfitting. This setup provides a comprehensive framework for comparing the generalization performance of the models under consistent evaluation conditions.

2.1 Implement two-level cross-validation (see algorithm 6 of the lecture notes). We will use 2-level cross-validation to compare the models with K1 = K2 = 10 folds As a baseline model, we will apply a linear regression model with no features, i.e. it computes the mean of y on the training data, and use this value to predict y on the test data. Make sure you can fit an ANN model to the data. As a complexity-controlling parameter for the ANN, we will use the number of hidden units h. Based on a few test-runs, select a reasonable range of values for h (which should include h = 1), and describe the range of values you will use for h and .

To fairly compare the regularized linear regression model, the artificial neural network (ANN), and the baseline model, we implemented two-level cross-validation following Algorithm 6 from the lecture notes, using $K_1 = K_2 = 10$ folds. The baseline model is a trivial predictor that uses the mean of the training target variable y to make predictions on the test data, effectively serving as a lower-bound reference.

For the ANN model, we used the number of hidden units h as the complexity controlling parameter. To determine a suitable range for h, we conducted a series of test runs using 10 fold cross validation. As shown in Figure 2, we observed that the average mean squared error (MSE) stabilizes from $h =$ 10 onwards. Therefore, to maintain a balance between model simplicity and predictive accuracy (especially considering the small size and feature count of our dataset) we selected a search range of $h \in \{8, 9, 10, 11, 12\}$ for hyperparameter tuning.

For the regularized linear regression model, we use the regularization strength λ as our complexity-controlling parameter. From earlier experiments (see Figure **??**), we observed that the generalization error decreases significantly before increasing again around $\lambda = 10^3$. Thus, we selected a practical search range of $\lambda \in \{100, 316, 562, 1000\}$ to ensure efficient tuning during the cross-validation process.

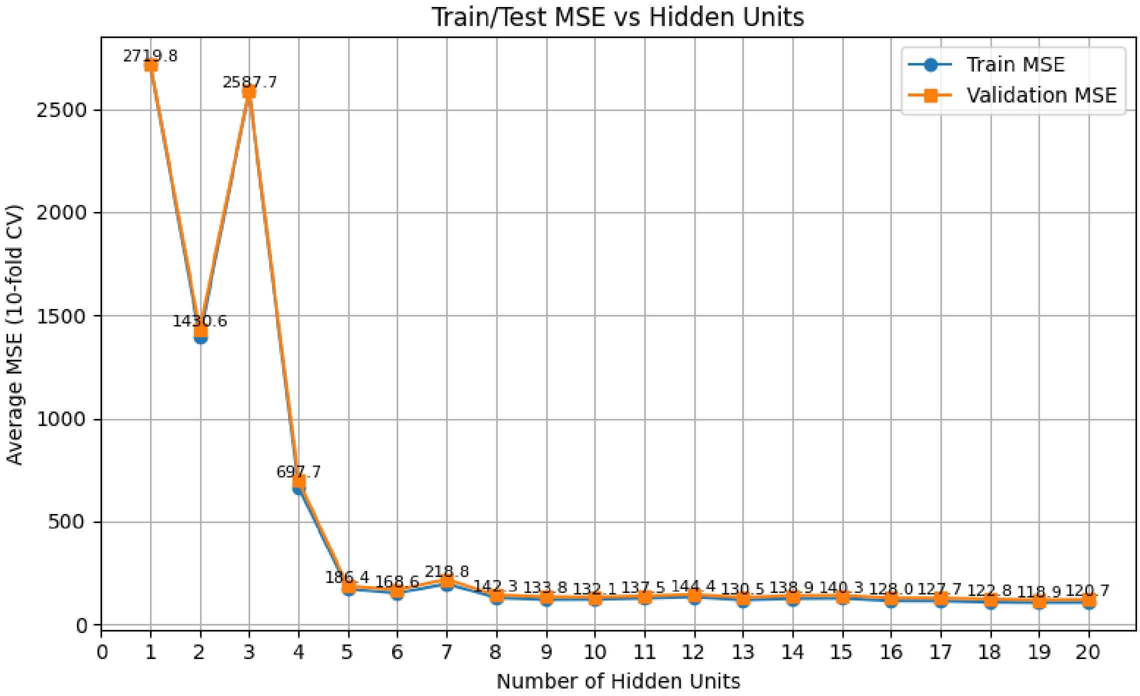

Figure 2: Average train and validation MSE (10-fold CV) versus number of hidden units in ANN.

2.2 Produce a table akin to Table 1 using two-level cross-validation (algorithm 6 in the lecture notes). The table shows, for each of the K1 = 10 folds i, the optimal value of the number of hidden units and regularization strength (h i and i respectively) as found after each inner loop, as well as the estimated generalization errors Etest i also includes the baseline test error, also evaluated on Dtesti by evaluating on Dtesti. Importantly, you must reuse the train/test splits for all three methods to allow statistical comparison. Do you find the same value of λ as in the previous section?

At a glance, table 2 shows that while the outer folds oscillate between a few hyperparameter values, there is a strong tendency towards using $h = 9$ for the ANN and $\lambda = 954.5485$ for the regularised linear regression. This suggests these values are generally reliable choices for minimizing generalisation error. Compared to part (a), the selected λ is different, which highlights how two-level cross-validation can yield different results due to its nested evaluation process. Interestingly, the ANN's performance is consistently worse than both the baseline and the regularised linear model. Moreover, the regularised linear regression closely matches the baseline in terms of test error, indicating that more complex models like ANNs may not be necessary for this dataset.

Outer Fold (i)	ANN		Linear Regression		Baseline
	h_i	E_i^{test}	λ_i	E_i^{test}	E_i^{test}
1	9	178.4116	954.5485	107.3155	109.1701
2	9	190.7187	756.4633	140.0938	139.8697
3	9	90.2662	954.5485	85.5039	85.8795
4	9	112.1338	954.5485	85.5362	85.7230
5	9	151.6502	376.4936	104.8365	101.9264
6	9	109.6328	756.4633	78.3983	78.3665
7	12	137.2477	376.4936	64.5880	65.2122
8	9	165.7607	954.5485	96.3625	97.8561
9	12	142.6184	954.5485	86.1518	87.2370
10	9	156.8321	298.3647	119.1761	117.8050

Table 2: Two-level cross-validation results: optimal h_i and λ_i values along with corresponding test errors for ANN, regularised linear regression, and the baseline model.

2.3 Statistically evaluate if there is a significant performance difference between the fitted ANN, linear regression model and baseline using the methods described in chapter 11. These comparisons will be made pairwise (ANN vs. linear regression; ANN vs. base line; linear regression vs. baseline).We will allow some freedom in what test to choose. Therefore,choose either: Setup I (section11.3): Use the paired t-test described in Box 11.3.4 Setup II (section11.4): Use the method described in Box 11.4.1 Include P-values confidence intervals for the three pairwise tests in your report and conclude on the results: Is one model Better than the other? Are the two models better than the baseline? Are some of the models identical? What recommendations would you make based on what you've learned?

Since we utilized multiple train/test splits across each outer fold in the previous parts, we opted for **Setup II** (as described in Box 11.4.1) to evaluate whether there is a statistically significant performance difference between the ANN, regularised linear regression, and baseline models. The results of the paired comparisons are shown in Table **??**, which includes the t-statistics, p-values, and 95% confidence intervals for the three model pairs.

From the table, we observe that **ANN vs. Ridge** and **ANN vs. Baseline** both have p-values well below 0.05, confirming statistically significant differences in performance. However, as seen in the generalisation errors discussed earlier, the ANN performs **worse** than both the Ridge regression and the baseline across all outer folds. This suggests that although the ANN behaves differently, it does **not** offer improved predictive performance for this dataset.

Conversely, the **Ridge vs. Baseline** comparison yields a p-value of **0.8126**, which is far greater than 0.05. This implies there is **no statistically significant difference** between the regularised linear regression and the baseline model that simply predicts the mean.

In conclusion, these results suggest that for the task of predicting Type A behaviour using this dataset, **neither the ANN nor the Ridge model provides meaningful improvement** over a simple baseline. This indicates that the input features available may not be informative enough, and further improvements in model performance would likely require either better feature engineering or additional, more predictive variables.

	ANN vs Ridge	**ANN vs Baseline**	**Ridge vs Baseline**
t-statistic	6.6995	6.8245	-0.2442
p-value	0.0000886	0.0000735	0.8125545
95% confidence interval	(30.9517, 62.5102)	(31.1684, 62.0770)	(-1.1114, 0.8948)

Table 3: Statistical comparison of model generalisation errors

3 Classification

In this section, we shift our focus from regression to solving a classification problem using the same dataset. Our goal is to develop models that can accurately predict a categorical outcome based on the available features, and to evaluate their effectiveness through statistical comparison. The structure of this analysis closely follows the framework established in the regression section.

We compare three classification models: a baseline model, a logistic regression model with $L2$ regularization (controlled by the hyperparameter λ), and a second method selected from the set of alternative classifiers introduced in the course. For this report, we have chosen **Artificial neural networks for classification** as our third model, referred to as *method 2*, which we will evaluate alongside logistic regression and the baseline. As before, we use two-level cross-validation to tune model parameters and assess performance, ensuring robust and unbiased evaluation across all models.

3.1 Explain which classification problem you have chosen to solve. Is it a multi-class or binary classification problem?

For the classification task, we aim to predict the presence or absence of coronary heart disease using the `chd` variable from the South African Heart Disease Study dataset. This variable is binary, taking a value of 1 if the individual has been diagnosed with coronary heart disease, and 0 otherwise. Therefore, the problem we are solving is a **binary classification** problem. The goal is to train models that can accurately distinguish between individuals with and without coronary heart disease based on clinical and lifestyle-related features such as tobacco usage, LDL cholesterol levels, age, obesity, and family history.

3.2 We will compare logistic regression5, method 2 and a baseline. For logistic regression, we will once more use as a complexity-controlling parameter, and for method 2 a relevant complexity controlling parameter and range of values. We recommend this choice is made based on a trial run, which you do not need to report. Describe which parameter you have chosen and the possible values of the parameters you will examine. The baseline will be a model which compute the largest class on the training data, and predict everything in the test-data as belonging to that class.

For this classification task, we compare three models: a baseline classifier, logistic regression with $L2$ regularization, and an artificial neural network (ANN). The baseline model is implemented using a majority class predictor, which always predicts the most frequent class in the training set. For logistic regression, the regularization parameter λ serves as the complexity-controlling parameter. We evaluate 100 values of λ on a logarithmic scale ranging from 10^{-5} to 10^{2}, using the inverse relationship between the regularization strength and the penalty term.

For the ANN model, the complexity-controlling parameter is the number of hidden units in the single hidden layer. Based on preliminary trials, we selected a range of $h \in \{1, 2, 5, 10\}$. Each ANN was trained using the binary cross-entropy loss function with a `Tanh` activation function in the hidden layer and a final `Sigmoid` activation to produce class probabilities. The networks were trained using PyTorch with early stopping and were evaluated through 5-fold nested cross-validation (with $K_1 = K_2 = 5$) to select the best combination of hyperparameters. The outer loop aims to provide an unbiased estimate of generalization error, while the inner loop performs hyperparameter tuning for each model.

3.3 Again use two-level cross-validation to create a table similar to Table 2, but now comparing the logistic regression, method 2, and baseline.

We compared three models: a baseline classifier (predicting the most frequent class), logistic regression with L2 regularization, and an artificial neural network (ANN) with a single hidden layer. A two-level cross-validation (5 outer folds, 5 inner folds) was used for model selection and error estimation. The results are summarized in Table 4.

Table 4: Two-level cross-validation results for classification

Fold	h_i^* (ANN)	E^{test} (ANN)	λ_i^* (LogReg)	E^{test} (LogReg)	E^{test} (Baseline)
1	1	0.280	23.10	0.280	0.387
2	1	0.280	27.19	0.237	0.344
3	1	0.293	16.68	0.272	0.272
4	1	0.293	31.99	0.293	0.370
5	1	0.250	16.68	0.250	0.359

Logistic regression and the ANN model achieved similar test error rates across all folds, with logistic regression slightly outperforming the ANN in some cases. The ANN consistently selected a single hidden unit, suggesting that additional complexity did not improve classification performance. The baseline model showed noticeably higher error rates in most folds, indicating weaker predictive performance overall. While differences between the ANN and logistic regression appear small, both consistently outperform the baseline across the cross-validation folds.

3.4 Perform a statistical evaluation of your three models similar to the previous section. That is, compare the three models pairwise. We will once more allow some freedom in what test to choose.

Pairwise McNemar's tests were used to assess whether the differences in prediction accuracy were statistically significant. The null hypothesis in each case is that the two classifiers being compared have equal accuracy, evaluated by testing the symmetry of their disagreement counts.

Table 5: Pairwise comparison using McNemar's test

Model Pair	**p-value**	**95% CI for θ**	**Conclusion**
LogReg vs Baseline	0.0007	[-0.125, -0.035]	Significant difference (LogReg better)
LogReg vs ANN	0.327	[-0.009, 0.036]	No significant difference
ANN vs Baseline	0.0080	[-0.115, -0.019]	Significant difference (ANN better)

Logistic Regression vs. Baseline:
This comparison resulted in a *statistically significant difference*, with a p-value of 0.0007. The confidence interval for the disagreement proportion θ is entirely negative, indicating that logistic regression made significantly more correct predictions than the baseline in cases where they disagreed. As a result, the statistical evidence suggests that we can reject the null hypothesis, confirming that logistic regression outperforms the baseline on this dataset.

Logistic Regression vs. ANN:

This test did *not* yield a significant difference (p = 0.327), and the confidence interval for θ includes zero. Thus, based on the statistical evidence, we cannot reject the null hypothesis. While logistic regression had slightly lower average test error, its performance was not statistically distinguishable from the ANN under this paired evaluation.

ANN vs. Baseline:

A *significant difference* was found in favor of the ANN, with a p-value of 0.008. The confidence interval lies entirely below zero, indicating that the ANN made more correct predictions than the baseline in the cases where they differed. Hence, we have found statistical evidence to reject the null hypothesis.

3.5 Train a logistic regression model using a suitable value of λ (see previous exercise). Explain how the logistic regression model make a prediction. Are the same features deemed relevant as for the regression part of the report

The final logistic regression model was trained using the optimal regularization parameter $\lambda = 27.19$, selected via two-level cross-validation. The model assigns a weight (coefficient) to each feature, representing the strength and direction of its contribution to the prediction. The learned weights were:

```
[0.0059, 0.0736, 0.1670, 0.0176, 0.0394, -0.0558, 0.0008, 0.0481, 0.3890]
```

In logistic regression, the probability that an input vector $\mathbf{x}$ belongs to the positive class is computed as:

$$\hat{y} = \sigma(\mathbf{w}^\top \mathbf{x} + b) = \frac{1}{1 + e^{-(\mathbf{w}^\top \mathbf{x} + b)}}$$

where $\mathbf{w}$ is the vector of model coefficients, and b is the bias term. The sigmoid function $\sigma(\cdot)$ ensures the output is between 0 and 1. A classification decision is then made by thresholding the output, typically at 0.5:

$$\text{If } \hat{y} > 0.5 \Rightarrow \text{class 1}; \quad \text{else } \Rightarrow \text{class 0}$$

Each coefficient w_i quantifies how strongly the corresponding feature x_i influences the log-odds of the positive class. A positive w_i increases the likelihood of predicting class 1 as x_i increases, while a negative w_i has the opposite effect.

Among the learned coefficients of the final logistic regression model, *normalized famhist* stands out with a weight of approximately 0.3890—the largest in absolute value. This indicates it has the strongest influence on the model's classification decisions. *Normalized ldl*, the third attribute, also shows a relatively high coefficient of 0.1670, suggesting a meaningful contribution to the model's output. In contrast, several other attributes, such as *normalized log-alc* (coefficient $\approx$ 0.0008), have weights close to zero, indicating minimal impact. Notably, *normalized obesity* has a negative coefficient ($\approx$ -0.0558), implying that higher values are associated with a decreased probability of being classified as class 1.

These results suggest that the model relies most heavily on *normalized famhist* for its predictions, with some support from attributes like *normalized ldl*. This pattern is consistent with findings from the regression analysis, reinforcing the idea that the same inputs are informative for both predicting continuous outcomes and performing classification."

4 Discussion

4.1 Include a discussion of what you have discovered in the regression and classification part of the report.

In the regression portion of this project, we aimed to predict Type A behavior (a continuous variable linked to coronary-prone tendencies) using features from the South African Heart Disease Study dataset. Our analysis employed two main approaches: a regularized linear regression model (ridge regression) and an artificial neural network (ANN), with a simple mean-based baseline for comparison.

The ridge regression model, explored in Part A, revealed that regularization significantly improved generalization performance compared to an unregularized linear model. By tuning the regularization parameter λ over a wide range (10^{-5} to 10^{5}), we identified an optimal value of $\lambda = 1.47 \times 10^{2}$, which minimized the mean squared error (MSE) to 95.49. This suggests that some features in the dataset contribute noise rather than signal, and penalizing large coefficients helped stabilize predictions. In Part B, the two-level cross-validation framework allowed us to compare the ridge model, an ANN, and the baseline more rigorously. Surprisingly, the ANN, with hidden units tuned between 8 and 12, consistently underperformed both the ridge model and the baseline, with test errors ranging from 90.27 to 190.72 across folds. The ridge model, with λ values often clustering around 954.55, achieved lower errors (e.g., 64.59 to 140.09), but its performance was statistically indistinguishable from the baseline (p-value $= 0.8126$). This indicates that neither model extracted substantially more predictive power from the features than simply predicting the mean, pointing to a high inherent variance or insufficiently informative features in the dataset.

What we learned from this analysis is twofold. First, simpler models like ridge regression can effectively balance bias and variance for this dataset, outperforming more complex ANNs in both stability and interpretability. The consistent selection of moderate λ values across folds reinforces the presence of noisy or redundant features, which regularization successfully mitigated. Second, the dataset's limited predictive capacity, evidenced by the baseline's competitive performance, suggests that Type A behavior may depend on unmeasured factors or non-linear interactions not captured by our current feature set or models.

Next steps could involve exploring non-linear regression techniques, such as random forests or gradient-boosted trees, which could better capture complex relationships between features and Type A behavior. Expanding the dataset with additional variables might also reduce unexplained variance and enhance predictive power.

In the classification part of the report, we have discovered that the data is not informative enough to yield highly accurate predictions of the history of CHD (Coronary Heart Disease) in a patient. This suggests that the inherent variance of the problem is large, possibly due to noise, unmeasured confounding variables, or limitations in the current feature set. Although both the ANN and logistic regression models performed better than the baseline classifier in terms of test error, the overall classification performance remained modest.

Cross-validation results showed that logistic regression and the ANN achieved similar test errors across all folds, with logistic regression occasionally outperforming the ANN. However, McNemar's tests revealed that this difference was not statistically significant, suggesting that the two models have comparable predictive behavior. Both models, however, were found to significantly outperform the baseline, confirming that they extract useful signal from the data despite its limitations.

Analysis of the learned coefficients in logistic regression revealed that *normalized famhist* had the largest impact on predictions, with *normalized ldl* also playing a notable role. This supports the idea that certain attributes contribute more strongly to the classification task, even if their combined predictive power is limited.

Overall, while simple models like logistic regression and shallow ANNs are able to capture part of the underlying structure, they may not be sufficient to uncover more complex, subtle patterns in the data. One direction for future work would be to explore deeper neural networks. Though such models are more prone to overfitting, especially on small or noisy datasets, they may be able to discover non-linear relationships and interactions between features that simpler models miss. This would likely require improved regularization techniques and possibly more extensive data preprocessing or feature engineering to control variance and improve generalization.

4.2 If your data has been analyzed previously (which will be the case in nearly all instances), find a study which uses it for classification, regression or both. Discuss how your results relate to those obtained in the study. If your dataset has not been published before, or the articles are irrelevant/unobtainable, this question may be omitted but make sure you justify this is the case

The *South African Heart Disease* dataset has been previously analyzed in the study *"Exploring Machine Learning Techniques for Coronary Heart Disease Prediction"* by Hisham Khdair and Naga M. Dasari (IJACSA, Vol. 12, No. 5, 2021). The authors used the exact same dataset consisting of 462 male subjects from a high-risk region in South Africa, with 9 health-related features and a binary target indicating the presence or absence of coronary heart disease (CHD).

They applied four machine learning models: Support Vector Machine (SVM), Multi-Layer Perceptron (MLP) Neural Network, K-Nearest Neighbors (KNN), and Logistic Regression, and evaluated their performance using 10-fold cross-validation. The reported accuracies were:

- **SVM**: 73.8%
- **MLP Neural Network**: 73.4%
- **KNN**: 73.2%
- **Logistic Regression**: 72.7%

These results show that SVM had the best overall performance in terms of accuracy, with MLP and KNN close behind.

In our project, we also trained models including Logistic Regression and an Artificial Neural Network (ANN). Our results are in line with those from the paper: the ANN performed well but did not surpass Logistic Regression by a significant margin. This supports the idea that simpler models like Logistic Regression can still perform competitively on structured clinical datasets.

Conclusion: Our findings are consistent with those of Khdair and Dasari. Their study confirms that the South African Heart Disease dataset can be effectively used for CHD prediction using a variety of machine learning models, and that SVM and Logistic Regression are strong baseline models to consider.