<u>QUESTION 1</u>

1. Introduction

Education plays a pivotal role in shaping the social and economic future of any country. In Sweden, mathematics education at the elementary level has been under scrutiny for several years due to concerns over student performance. Although recent results from the Programme for International Student Assessment (PISA) indicate an overall improvement in Sweden's national averages over the past decade, this aggregated progress may conceal substantial regional disparities. One such region is Dalarna County, located in central Sweden, which comprises both urban and rural municipalities. Educational outcomes in Dalarna have drawn attention due to inconsistencies in performance, particularly in core subjects such as mathematics. Poor math outcomes can limit students' opportunities for higher education and future employment, especially in an increasingly data-driven and technical world.

This study therefore seeks to explore grade 9 mathematics passing rates throughout the municipalities in Dalarna County over the span of the last decade from 2015 to 2024. Also, we investigate whether average income per household, higher education, and political preferences, such as voting for the Social Democrats or Moderates, influence these outcomes.

Understanding these associations can help policymakers and education practitioners to identify root-cause factors for children's underperformance and to develop better-tailored support for students, particularly in struggling regions. The analysis utilizes several publicly available datasets to construct simple and multiple linear regression models in order to look for trends and predictors.

1.1 Research Questions:-

1. Has there been a significant trend (increasing or decreasing) in students' mathematics performance across municipalities in Dalarna County over the last 10 years (2015–2024)?
2. Are demographic characteristics (such as average income and level of higher education) and political preferences associated with variations in students' mathematics performance at the municipal level?

2 Methodology

This section details the data sources and the preprocessing steps, including data cleaning, merging, and transformation, and the statistical modeling processes used to explore trends and possible predictors of mathematics performance in Dalarna County's municipalities.

2.1 Data Acquisition

To investigate trends and predictors of math performance among grade 9 students in Dalarna County, the following four datasets were utilized.

- **Grade 9 Mathematics Results (Result_AK9_Dalarna_Kum.xlsx):** Shows yearly passing percentage in mathematics for the various municipalities.
- **Municipal Election Results (Dalarna_MunicipalElectionResults.xlsx):** Total vote share for the major parties.
- **Average Income (Dalarna_AverageIncome.xlsx):** Yearly average and median incomes by municipality.

- **Population with Higher Education (Dalarna_Population_HigherEducation.xlsx):** Provides data on individuals who have completed post-secondary education of less than three years and those with three years or longer.

2.2 Data Preprocessing

To ensure the quality and usability of the final dataset for analysis, the following preprocessing steps were undertaken:

2.2.1 Data Cleaning and Transformation

Data cleaning was performed using Microsoft Excel to ensure the datasets were consistent, accurate, and ready for analysis. Key steps included:

- Removing irrelevant columns that are not required for the analysis to reduce redundancy and simplify the datasets.
- Renaming variables for consistency, ensuring that similar data fields had uniform naming conventions across all datasets.
- Filtering the data to the target period (2015–2024) to align with the study's defined timeframe.
- Standardizing municipality names across all datasets to ensure accurate merging and comparison during analysis.

2.2.2 Data Merging

The cleaned datasets are then merged in R using the dplyr package to handle the data merging efficiently. They are merged using the left_join() function, which is designed to integrate a number of datasets with common identifiers. Here, Municipality, Year and Code are used as "join keys" to combine the various data sources because they provide a unique identifier that guarantees that all records in the various data sources combine accurately and consistently. This is important to keep the combined dataset clean so that duplicate or mismatched data does not exist.

Along with the merge, a new derived variable called HigherEducationPercent is created to streamline the analysis. It captures the proportion of the municipal population that has finished the equivalent of three or more years of higher education. It is obtained by dividing the total number of people with this degree of education over the overall municipality population and then multiplying it by hundred. This transformation allows for a comparable measure of the level of education across municipalities.

3 Results and Discussion

All analyses performed using the software program R, employing descriptive and exploratory inferential analyses aimed at identifying trends and exploring possible associations within the data. These were the methods employed:

3.1 Trends in Mathematics Performance

The analysis starts with the development of a simple linear regression model in which Year is the independent variable used to predict grade 9 students' mathematics passing rate in Dalarna County. The regression results show that Year has a statistically significant negative coefficient of (-0.3646 p=0.0228), indicating that average passing rates decrease over the general time period of 2015-2024.

However, the model explains very little of the overall variation in passing rates, with a Multiple R-squared of 0.0345 and an Adjusted R-squared of 0.0280. This indicates that only about 3% of the variation is accounted for by Year alone.

3.1.1 Average Math Passing Rate Trend In Dalarna County 2015-2024:

A second plot is generated to illustrate the overall trend in mathematics performance in Dalarna County between 2015-2024 in **Figure 1** below. A red dotted regression line is added to the plot to illustrate a downward trend over the decade in which passing rates have consistently decreased. The grey confidence band represents the uncertainty around the trend, but with a negative slope it also supports the Student Performance trend found in Model 1, where overall Student Performance tends to decrease over time.

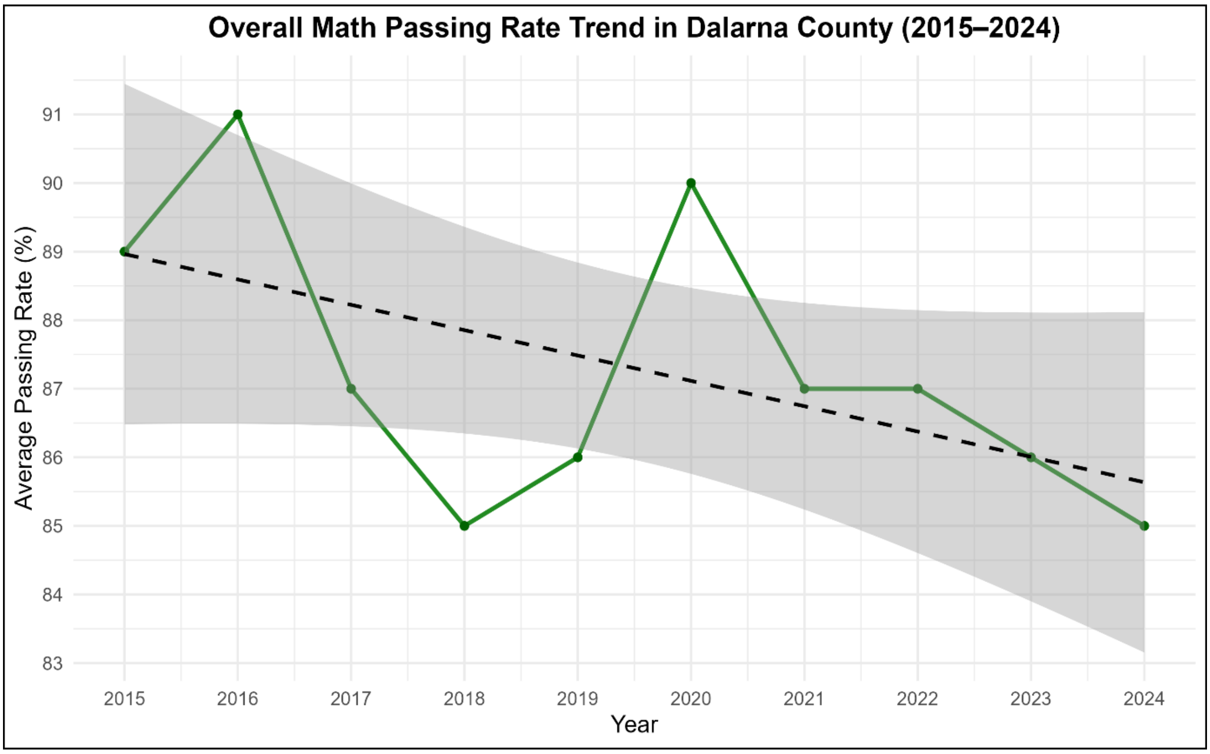

Figure 1

3.1.2 Math Passing Rate Trends by Municipality 2015-2024:

Line plots are created to examine individual municipality trends in mathematics passing rates over time in **Figure 2**. The plot shows the local trends for all municipalities in Dalarna County, and it is clear that there is a lot of variation in students' performance year by year at the local level. Borlänge and Mora are municipalities with more stable trends, while Vansbro and Malung-Sälen municipalities show a sharper decline. This variation is indicative of localized, municipality-specific influences on math performance, rather than regional trends.

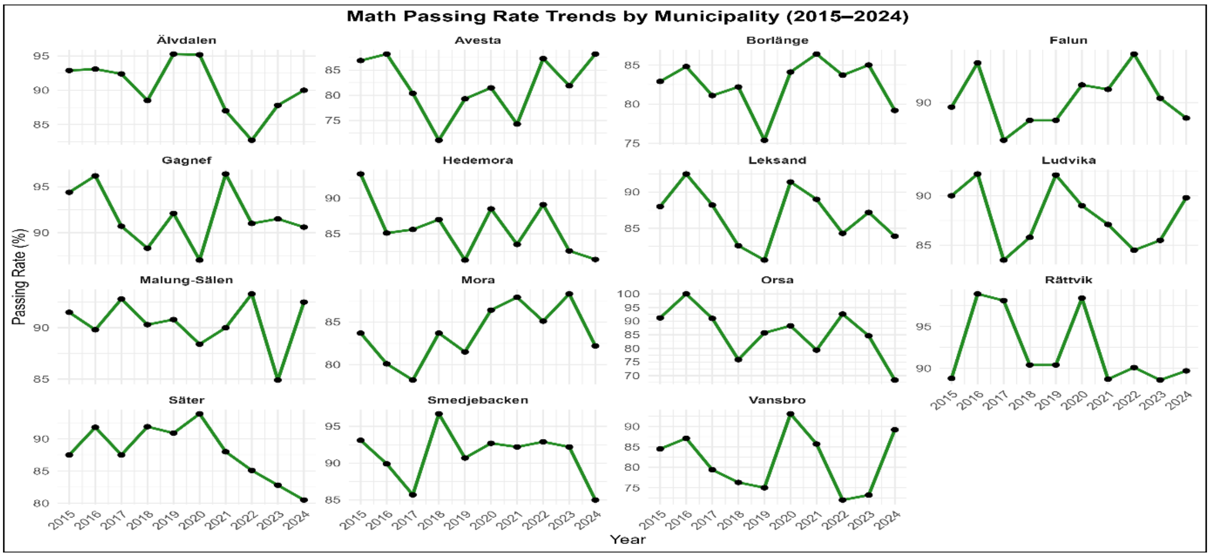

Figure 2

3.2 Regression Models:

Two regression models are developed to assess the impact of various factors on grade 9 mathematics passing rates in Dalarna County:

3.2.1 Simple Linear Modeling:

Model 1: PassingRate ~ Year

- **Purpose:** Examines the overall trend in passing rates over time, using *Year* as the sole predictor.

Model 1 Output (PassingRate ~ Year):

- **Coefficient for Year:** -0.3646 (p = 0.0228)
- **R-squared:** 0.0345
- **Interpretation:** This model suggests that there is a statistically significant negative trend in passing rate of about 0.36 percentage points per year. The R-squared is relatively low, with a 3.45% of the variance being explained by year alone, but the trend is consistent with national concerns over regional differences.

3.2.2 Multiple Linear Regression

Model 2: PassingRate ~ Year + HigherEducationPercent + AverageIncome + SocialDemocrats + Moderates.

- **Purpose:** Expands the analysis by incorporating socio-economic and political variables to explain variation in municipal-level performance.

Model 2 Output:

- Year remains significant (p = 0.0476)
- **Higher Education Percent:** Positive coefficient but not statistically significant (p = 0.1099)
- **Average Income:** Positive, not statistically significant (p = 0.1352)

- **Social Democrats Vote Share:** Not significant (p = 0.5241)
- **Moderates Vote Share:** Marginally approaching significance with a negative effect (p = 0.0855)
- **R-squared:** 0.0852
- **Interpretation:** Model 2 adds important demographic and political variables. Despite showing an improved fit compared to model 1, with R-squared of 8.5%, the majority of the predictors are still non-significant. Year is the only variable that remains significantly effective. Interestingly, but not conclusive of anything, the vote share variable for Moderates hints at a weak negative relationship to performance that should be investigated more in depth.

3.3 Correlation Analysis:

In order to provide a visual sense of possible associations between the mathematics passing rate and each of the predictor variables, a correlation matrix is created **(Figure 3).** The matrix gives a clear picture of variables' relations to one another. HigherEducationPercent is weakly positively correlated with Passing Rate, which would indicate that as the population of highly-educated individuals increases in a municipality, math performance would slightly increase as well. Passing Rate also has a weak positive correlation with AverageIncome.

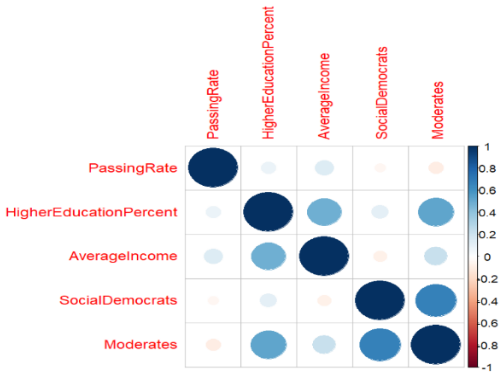

Figure 3

Political variables show the highest correlations, such as the negative correlation between support for the Social Democrats and the Moderates. This is expected due to the opposing nature of their political ideologies and competition for voter support. Overall, the light blue shading and smaller circle sizes throughout most of the matrix confirm that the relationships among the variables are generally weak, indicating the absence of strong linear associations.

3.4 Scatter Plot Analysis of Predictor Relationships

3.4.1 Math Passing Rate vs Higher Education (%):

The scatter plot **(Figure 4)** reveals a slight upward trend, suggesting that municipalities with higher percentages of residents with higher education may see better math passing rates among students. Nonetheless, the wide dispersion of data points indicates that this relationship is relatively weak, and other factors likely influence student performance across

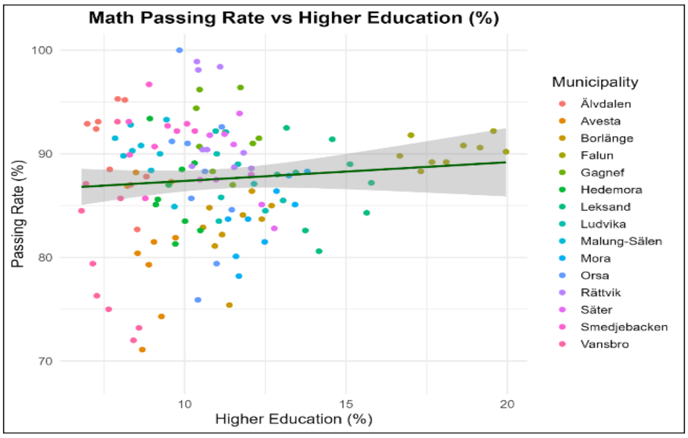

Figure 4

3.4.2 Math Passing Rate versus Average Income:

The scatter plot **(Figure 5)** and trend line indicate a clearer upward trend, suggesting that municipalities with higher average income tend to have higher math passing rates. This relationship appears stronger than with education level alone, reinforcing the broader understanding that socio-economic status significantly influences academic achievement.

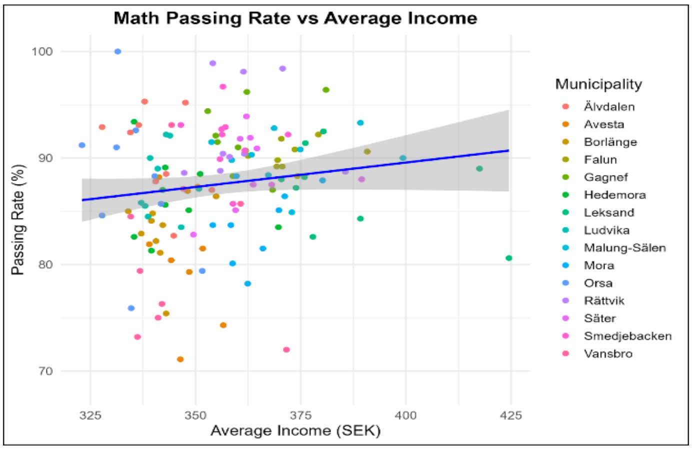

Figure 5

3.4.3 Math Passing Rate vs Social Democrats Vote Share:

The scatter plot **(Figure 6)** shows the trend line is almost horizontal and the data points are widely spread out, which indicates that there is very little or no correlation between students' math passing rates and Social Democrats' vote share. This seems to indicate that there is no apparent correlation between support for the Social Democrats and how students perform in math in these municipalities.

3.4.4 Math Passing Rate vs Moderates Vote Share:

The scatter plot **(Figure 7)** shows a slight downward trend between Moderates' vote share and math passing rates, but the wide spread of data points indicates a weak and inconclusive relationship. Although statistical analysis may suggest marginal significance, the visual evidence does not strongly support a meaningful association between support for the Moderates and student performance.

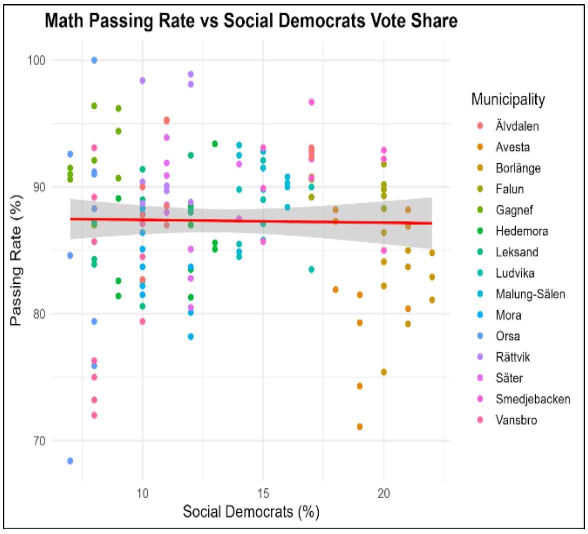

Figure 6

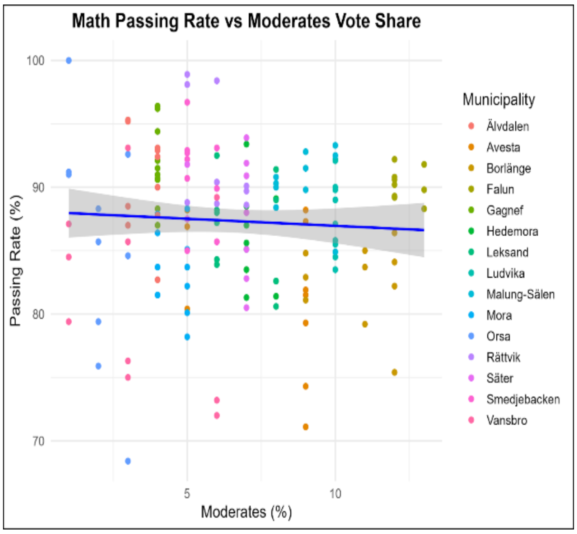

Figure 7

4. Conclusion

The study's findings indicate a statistically significant decline in mathematics passing rates among grade 9 students in Dalarna County from 2015 to 2024. Regression analysis confirms that time (year) is the only consistently significant predictor, pointing to a modest but steady decrease in performance over the years. Even though the study examined other factors such as education levels, average income, and political preferences (measured by vote shares for the Social Democrats and the Moderates), none of them showed a strong or consistent connection to student performance. The scatter plots and correlation matrix supported these findings, showing only weak or scattered relationships.There were slight positive relationships between higher income or education levels and better passing rates, but these associations were not strong enough to draw firm conclusions. Political support, in contrast, appeared to have little or no influence on how students performed.

Overall, these results highlight that student achievement is influenced by a wide range of factors. The factors analyzed in the present study accounted for a small percentage of the variance in math performance.This suggests that other elements, such as school resources, teaching quality, curriculum design, or individual student circumstances, may play a more important role. Future research should explore these aspects more deeply to better understand what truly affects student success in mathematics.

References:

1) **OECD. (2023).** *PISA 2022 results*. Organisation for Economic Co-operation and Development. **https://www.oecd.org/pisa/**
2) **Skolverket. (n.d.).** *Statistics on grade 9 results and educational attainment in Sweden*. Swedish National Agency for Education.**https://www.skolverket.se**

QUESTION 2

1. Introduction

Accurate classification of animal behaviour is essential for monitoring the health, welfare, and productivity of domestic species. Traditional observation methods are often time-consuming and subjective, making automated approaches increasingly valuable. Wearable technologies, particularly accelerometer sensors, offer a non-invasive and continuous means of collecting objective movement data, enabling detailed behavioral analysis in real-world conditions.

The accelerometer can capture motion in the three-dimensional X, Y, and Z axes and thus obtain data indicative of patterns associated with postures and activity. When segmented into time windows and paired with machine learning algorithms, this data can be used to distinguish between behaviors such as walking, standing, or lying. However, successful classification depends on effective preprocessing, feature extraction, and model selection. Furthermore, selecting suitable models and evaluation metrics is crucial when dealing with imbalanced behaviour classes or noisy real-world data.

In this study, a dataset of thousands of labeled 5-second accelerometer recordings is used to evaluate four classification algorithms: Logistic Regression, k-Nearest Neighbours, Naive Bayes, and Neural Networks. After preprocessing and normalisation, each model was assessed using accuracy, precision, recall, F1-score, and confusion matrix analysis. The results demonstrate the viability of using accelerometer data and machine learning for automated animal behaviour classification and highlight key differences in model performance under real-world constraints.

Research Questions:-

1. Can animal behavior be accurately classified using accelerometer-derived features from a 5-second interval window?
2. How do different classification models—Logistic Regression, k-Nearest Neighbors, Naive Bayes, and Neural Networks—compare in predicting animal behavior?

2 Methodology

This section describes the systematic procedure followed for data preparation and model development. Beginning with the raw accelerometer readings paired with corresponding behavioral annotations, the dataset underwent a series of preprocessing steps, including data cleaning, normalization of features, and division into training and testing subsets. To evaluate predictive performance, four commonly adopted classification techniques were employed: Logistic Regression, k-Nearest Neighbors, Naive Bayes, and Neural Networks. Each model was trained to distinguish between multiple behavioral categories. Their effectiveness was measured using standard evaluation metrics such as accuracy, precision, recall, and F1-score, along with confusion matrix visualizations to facilitate comprehensive comparison.

2.1 Data Description

The dataset used in this study contains features extracted from accelerometer data collected from sensors positioned on the necks of domestic animals. Each observation represents a 5-second interval during which motion-related features were computed to capture animal behavior. These features include measurements of pitch, roll, dynamic body acceleration, and various rotational statistics. After removing entries labeled as “Missing data,” the final dataset was used for training and evaluating classification models.

Key Characteristics of the Dataset:

- **Sensor Location** : Neck-mounted accelerometer on domestic animals
- **Sampling Window** : 5-second intervals per observation
- **Original Observations :** 16,920 labeled data points
- **Excluded Data** : 1,464 instances removed due to "Missing data" labels
- **Final Dataset Size** : 15,456 valid observations used in analysis
- **Feature Count** : 52 numerical features per observation
- **Target Variable** : 1 categorical variable indicating observed behavior ("Modifiers")
- **Feature Types** : Includes pitch, roll, dynamic acceleration, and rotational statistics

2.2 Preprocessing

To ensure the quality and consistency of the data used for model training and evaluation, several preprocessing steps were performed:

- **Removal of Missing Labels:** All rows where the target variable (Modifiers) was labeled as "Missing data" were excluded from the dataset. This step eliminated 1,464 non-informative entries, improving the overall reliability of the classification task.
- **Dropping Non-Predictive Columns:** The ID and Timestamp columns were removed, as they do not contribute to behavioral prediction and may introduce noise or bias in the learning process.
- **Feature Normalization:** All numeric features were scaled to a uniform range between 0 and 1 using min-max normalization. This step ensures that no feature dominates others due to differing numerical scales, and it facilitates better convergence during model training.
- **Train-Test Split:** The final dataset was randomly divided into two parts, with 70 percent of the data allocated for training the models and the remaining 30 percent used for testing and evaluation. This split allowed for an unbiased assessment of model performance on previously unseen data.

2.3 Classification Models Used

Four classification algorithms were selected to evaluate their effectiveness in predicting animal behavior, representing a balance of linear, probabilistic, distance-based, and nonlinear approaches.

- **Logistic Regression:** A linear model adapted for multi-class classification using a multinomial formulation. It provides a simple and interpretable baseline.
- **k-Nearest Neighbors (kNN):** A non-parametric model that classifies each instance based on the majority class among its five nearest neighbors in the feature space.
- **Naive Bayes:** A probabilistic classifier that applies Bayes' theorem with the assumption of feature independence. It is computationally efficient and often effective despite its simplicity.
- **Neural Network (nnet):** A feed-forward neural network with one hidden layer, trained using backpropagation. It captures nonlinear relationships and is suitable for complex patterns.

These models were chosen to compare simple linear and probabilistic methods (Logistic Regression, Naive Bayes) against more flexible, nonlinear techniques (kNN, Neural Network), offering a broad perspective on classifier performance.

3. Results

3.1 Model Accuracy Comparison

Classification performance was assessed only in terms of accuracy, a simple and effective measure of predictive performance for the models. A clear view of the relative model performances is provided in **Figure 1**. This study investigates four classification algorithms: Logistic Regression, k-Nearest Neighbors, Naive Bayes, and Neural Network. After all preprocessing procedures, each model trained on 70% of the data and was tested on the remaining 30%. The use of this framework for the analysis generated in this study allowed for models to be consistently and impartially evaluated in terms of their ability to predict animal behaviour from features derived from the accelerometer data.

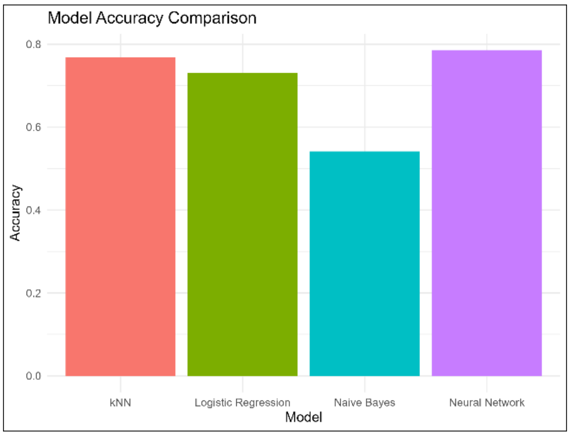

Figure 1: Accuracy Comparison of Classification Models

Interpretation: The overall classification accuracy of each model is presented in the table below.

The Neural Network achieved the highest accuracy at 78.6 percent, reflecting its ability to capture complex, nonlinear patterns. The k Nearest Neighbors model followed with 77.1 percent, demonstrating the effectiveness of proximity-based classification. Logistic Regression showed moderate performance with 73.1 percent, while Naive Bayes recorded the lowest accuracy at 54.2 percent, likely due to its assumption of feature independence.

Model	Accuracy
Logistic Regression	73.1%
k-Nearest Neighbors	77.1%
Naive Bayes	54.2%
Neural Network	78.6%

These results indicate that models capable of capturing nonlinear relationships are better suited for classifying animal behavior from accelerometer data.

3.2 Precision, Recall, and F1-Score

To provide a more comprehensive evaluation of model performance beyond overall accuracy, three additional metrics were considered: precision, recall, and F1-score. These metrics are particularly useful in multi-class classification tasks, where class imbalances or varying misclassification costs may affect the interpretation of accuracy alone.

The following formulas were used:

- **Precision**= True Positives / (True Positives + False Positives)
- **Recall** = True Positives / (True Positives + False Negatives)
- **F1-Score** = 2 × (Precision × Recall) / (Precision + Recall)

These metrics offer deeper insights into the quality of the predictions. Precision reflects the proportion of correct positive predictions out of all predicted positives, indicating the model's ability to avoid false alarms. Recall measures the ability of the model to correctly identify actual positive cases, highlighting its sensitivity. The F1-score, as the harmonic mean of precision and recall, balances both concerns and is especially informative when there is an uneven distribution of classes.

Model	Precision	Recall	F1_Score
Logistic Regression	0.60	0.51	0.57
kNN	0.64	0.50	0.60
Naive Bayes	0.41	0.54	0.45
Neural Network	0.66	0.52	0.62

These metrics offer a more detailed understanding of how each classifier performs, especially in the context of multi-class prediction where accuracy alone may not be sufficient.

Interpretation:

The evaluation using precision, recall, and F1-score shows that the Neural Network performed the best overall. It had the highest precision and F1-score, meaning it made more correct predictions and maintained a good balance between identifying actual behaviors and avoiding incorrect ones. The k Nearest Neighbors model also gave strong results, performing well in all three metrics. Logistic Regression showed moderate performance, missing some true behaviors but still being fairly accurate. Naive Bayes had the lowest precision and F1-score, meaning it often predicted incorrectly, although it did detect more actual behaviors than expected. Overall, models like Neural Networks and k Nearest Neighbors, which can handle complex patterns, were more effective than simpler models for this task.

3.3 Confusion Matrix Analysis:

The confusion matrix heatmap presented in **Figure 2** illustrates the performance of the Neural Network model in classifying different animal behaviors. Each row represents the predicted class, and each column represents the actual observed behavior. The intensity of the green shading indicates the frequency of predictions, with darker tiles corresponding to higher counts. A strong diagonal pattern indicates accurate classification.

Interpretation of Confusion Matrix Heatmap (Neural Network)

The Neural Network model demonstrates strong classification performance for several frequent behaviors. Clear diagonal patterns for classes such as Low, Ground, Resting, and Ruminating indicate that the model correctly identified these behaviours with high consistency. This reflects the model's ability to learn and generalize well from abundant and distinctive training data for these categories.

However, the heatmap also highlights some areas of confusion. For instance, the model occasionally misclassified Standing as Low, and Grooming as Low, which suggests that these behaviours may share similar movement profiles in the accelerometer data. Such overlap in motion features can challenge the model's ability to distinguish subtle differences between behaviors.

Moreover, rare behaviors such as Trotting, Digging, Breaking branches, and Agnostic behavior were frequently misclassified or not predicted accurately. This is likely due to the limited number of samples available for these classes, leading to underrepresentation during training and weaker model generalization for infrequent behaviors.

Overall, the confusion matrix confirms that the Neural Network effectively classifies the most common behaviors but reveals limitations when distinguishing less frequent or behaviorally similar classes. These findings suggest that model performance could be further improved by addressing class imbalance, enriching the dataset with additional samples from underrepresented behaviors, or incorporating more discriminative features.

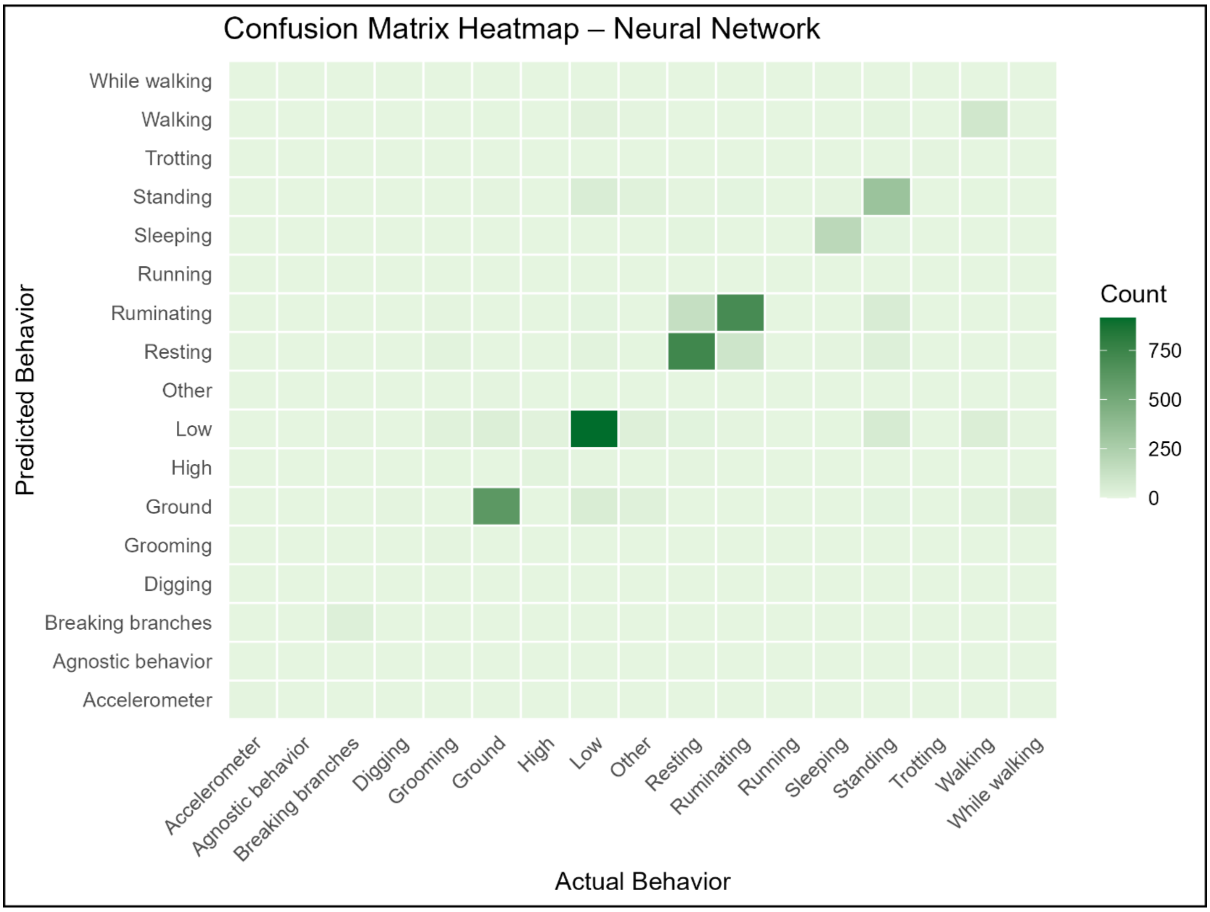

Figure 2 Confusion Matrix Heatmap – Neural Network

4. Discussion

This work shows that animal behaviour can be predicted from five second windows of accelerometer sensor data. The Neural Network outperformed all other models in the majority of scenarios amongst the models that were evaluated, particularly in regard to complicated multiclass classification. The fact that this model performed better is most likely an indication of the capacity of this model to explain inherently nonlinear associations and interactions among several features inherent to the patterns of movement of animals.

The k Nearest Neighbors Algorithm performed well and would have benefited from a feature normalization since kNN performs better in a normalized feature space . Having input variables normalized to the same range allowed for a better distance assessment in the feature space, which is important for models that determine similarity between samples. This was also an advantage for Neural Networks, since having consistent ranges for the inputs makes the learning process more stable and the network to converge to a solution more efficiently.

The Naive Bayes classifier was, relatively speaking, the lowest performing model. This is because of its central hypothesis of conditional independence of features given the class label. This assumption seldom applies to accelerometer-derived data, because measurements across many axes and computed features frequently exhibit significant correlations.

5. Limitations

The class imbalance problem in the dataset is one of the main drawbacks of this study. Behaviors like Low, Ground, and Resting were very well represented and had enough instances for the models to learn their patterns well. On the contrary, Trotting, Digging and Agnostic behavior were greatly missed. The skewness of the distribution also resulted in a limited training experience of the models on the infrequent types of activities, that in turn affected their capacity of classifying those particular behaviors. The impact of this restriction is evident in the confusion matrix, which shows that rare behaviors were misclassified more often.

Another limitation is related to the underlying assumptions of specific algorithms. For example, the Naive Bayes classifier assumes that all features are conditionally independent given the class label. However, this assumption does not align well with accelerometer data, where motion features such as pitch, roll, and dynamic acceleration are often strongly correlated. This violation of the independence assumption likely contributed to the relatively poor performance of the Naive Bayes model in this study.

6. Conclusion

This study demonstrates the effectiveness of using accelerometer data for classifying animal behaviour with a high degree of accuracy. By analyzing motion patterns over five-second intervals, meaningful behavioral information was extracted and used to train multiple classification models. Among the models evaluated, Neural Networks achieved the highest performance, likely due to their capacity to capture complex, nonlinear relationships and interactions between features.

The results highlight the potential of wearable sensor technology combined with machine learning to support automated behavioral monitoring. Such systems could offer valuable insights in various domains, including animal welfare assessment, livestock management, and behavioral science. The successful classification of common behaviors suggests that this approach is both feasible and scalable for real-world applications. Future research can build upon these findings by exploring more advanced or hybrid models that further improve classification of rare or ambiguous behaviors. Additionally, applying this methodology in real-time field environments would help validate its practical use and identify challenges related to sensor deployment, data streaming, and decision support systems in dynamic settings.

References:

1) Nathan, R., Spiegel, O., Fortmann-Roe, S., Harel, R., Wikelski, M., & Getz, W. M. (2012). Using tri-axial acceleration data to identify behavioral modes of free-ranging animals: General concepts and tools illustrated for griffon vultures. *Journal of Experimental Biology, 215*(6), 986–996. https://doi.org/10.1242/jeb.058602
2) Resheff, Y. S., Rotics, S., Harel, R., Spiegel, O., & Nathan, R. (2014). AcceleRater: A web application for supervised learning of behavioral modes from acceleration measurements. *Movement Ecology*, *2*(1), 27. https://doi.org/10.1186/s40462-014-0027-0

Appendix

Results of Question 1:

1.Simple Linear Regression Model

```
Call:
lm(formula = PassingRate ~ Year, data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-17.294  -3.207   1.068   3.870  11.389 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)  
(Intercept) 823.6566   320.0274   2.574   0.0110 *
Year         -0.3646     0.1585  -2.301   0.0228 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '

Residual standard error: 5.575 on 148 degrees of freedom
Multiple R-squared:  0.03453,	Adjusted R-squared:  0.02801 
F-statistic: 5.294 on 1 and 148 DF,  p-value: 0.0228
```

2.Multiple Linear Regression Model

```
Call:
lm(formula = PassingRate ~ Year + HigherEducationPercent + Ave
    SocialDemocrats + Moderates, data = data)

Residuals:
     Min       1Q   Median       3Q      Max 
-14.9967  -2.6070   0.4673   3.5495  10.8601 

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)  
(Intercept)             855.9950   393.9198   2.173   0.0316 *
Year                     -0.3902     0.1951  -2.000   0.0476 *
HigherEducationPercent    0.3806     0.2364   1.610   0.1099  
AverageIncome             0.0472     0.0314   1.503   0.1352  
SocialDemocrats           0.1062     0.1663   0.639   0.5241  
Moderates                -0.4620     0.2666  -1.733   0.0855 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.42 on 129 degrees of freedom
  (15 observations deleted due to missingness)
Multiple R-squared:  0.08517,	Adjusted R-squared:  0.04971 
F-statistic: 2.402 on 5 and 129 DF,  p-value: 0.04047
```

1.Multinomial Logistic Regression

```
> log_model <- multinom(Modifiers ~ ., data = train)
# weights:  884 (816 variable)
initial  value 30655.368383
iter  10 value 16442.099835
iter  20 value 12997.469160
iter  30 value 10399.667882
iter  40 value 9539.971591
iter  50 value 8734.108154
iter  60 value 8441.082585
iter  70 value 8289.269496
iter  80 value 8210.305248
iter  90 value 8142.739072
iter 100 value 8089.480728
final  value 8089.480728
stopped after 100 iterations
```

1.1 Logistic Regression Confusion Matrix

```
> confusionMatrix(log_pred, test_y)
Confusion Matrix and Statistics

                   Reference
Prediction          Accelerometer Agnostic behavior Breaking branches Digging Grooming
  Accelerometer                 3                 0                 0       0        0
  Agnostic behavior             0                 2                 0       0        0
  Breaking branches             0                 1                36       1        0
  Digging                       0                 0                 2       0        0
  Grooming                      0                 0                 0       0        8
  Ground                        0                 0                 0       3        0
  High                          0                 0                 0       0        1
  Low                           0                 0                 1       0       20
  Other                         0                 0                 1       0        1
  Resting                       0                 0                 0       0       12
  Ruminating                    0                 0                 0       0        6
  Running                       1                 0                 0       0        0
  Sleeping                      0                 0                 0       0        3
  Standing                      0                 0                 0       0        0
  Trotting                      0                 0                 0       0        0
  Walking                       0                 2                 3       0        0
  While walking                 0                 0                 0       0        0
```

```
                   Reference
Prediction          Ground High Low Other Resting Ruminating Running Sleeping Standing
  Accelerometer          0    0   0     0       0          0       0        0        0
  Agnostic behavior      1    0   0     0       0          0       0        0        0
  Breaking branches      0    2   3     3       0          0       0        0        1
  Digging                3    0   0     1       0          0       0        0        0
  Grooming               0    0   3     1       0          0       0        1        3
  Ground               611    0  51    23       0          0       0        0        5
  High                   0   10   8     1       2          0       0        1        0
  Low                   42   19 912    30      20          5       0        4       63
  Other                  1    0   1     1       0          0       0        0        4
  Resting                0    0  12     5     612        197       0        9       70
  Ruminating             0    3   7     4     265        611       1        0       29
  Running                0    1   0     0       0          0       1        0        0
  Sleeping               1    0   3     3       7          0       0      176       12
  Standing               3    1  54    18       6         10       0        9      318
  Trotting               0    0   0     0       1          0       0        0        0
  Walking                4    0  14     3       0          0       2        0        3
  While walking          3    0   2     2       0          0       0        0        3
```

```
                   Reference
Prediction          Trotting Walking While walking
  Accelerometer            0       0             0
  Agnostic behavior        0       0             0
  Breaking branches        2       2             0
  Digging                  0       0             0
  Grooming                 0       0             0
  Ground                   0       9            22
  High                     1       0             0
  Low                      0      66            11
  Other                    0       2             0
  Resting                  0       0             0
  Ruminating               0       0             0
  Running                  0       0             0
  Sleeping                 0       0             0
  Standing                 0       1             0
  Trotting                 5       0             0
  Walking                  1      73             1
  While walking            0       3             9

Overall Statistics

               Accuracy : 0.7308
                 95% CI : (0.7178, 0.7435)
    No Information Rate : 0.2308
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6759

 Mcnemar's Test P-Value : NA
```

Statistics by Class:

	Class: Accelerometer	Class: Agnostic behavior
Sensitivity	0.7500000	0.4000000
Specificity	1.0000000	0.9997841
Pos Pred Value	1.0000000	0.6666667
Neg Pred Value	0.9997842	0.9993525
Prevalence	0.0008628	0.0010785
Detection Rate	0.0006471	0.0004314
Detection Prevalence	0.0006471	0.0006471
Balanced Accuracy	0.8750000	0.6998920

	Class: Breaking branches	Class: Digging	Class: Grooming
Sensitivity	0.837209	0.0000000	0.156863
Specificity	0.996734	0.9987047	0.998255
Pos Pred Value	0.705882	0.0000000	0.500000
Neg Pred Value	0.998473	0.9991361	0.990693
Prevalence	0.009275	0.0008628	0.011001
Detection Rate	0.007765	0.0000000	0.001726
Detection Prevalence	0.011001	0.0012942	0.003451
Balanced Accuracy	0.916972	0.4993523	0.577559

	Class: Ground	Class: High	Class: Low	Class: Other	Class: Resting
Sensitivity	0.9133	0.277778	0.8523	0.0105263	0.6703
Specificity	0.9715	0.996957	0.9212	0.9977978	0.9181
Pos Pred Value	0.8439	0.416667	0.7645	0.0909091	0.6674
Neg Pred Value	0.9852	0.994363	0.9541	0.9796757	0.9191
Prevalence	0.1443	0.007765	0.2308	0.0204918	0.1969
Detection Rate	0.1318	0.002157	0.1967	0.0002157	0.1320
Detection Prevalence	0.1562	0.005177	0.2573	0.0023727	0.1978
Balanced Accuracy	0.9424	0.637367	0.8868	0.5041621	0.7942

	Class: Ruminating	Class: Running	Class: Sleeping	Class: Standing
Sensitivity	0.7424	0.2500000	0.88000	0.62231
Specificity	0.9174	0.9995682	0.99346	0.97527
Pos Pred Value	0.6598	0.3333333	0.85854	0.75714
Neg Pred Value	0.9429	0.9993525	0.99458	0.95422
Prevalence	0.1775	0.0008628	0.04314	0.11022
Detection Rate	0.1318	0.0002157	0.03796	0.06859
Detection Prevalence	0.1997	0.0006471	0.04422	0.09060
Balanced Accuracy	0.8299	0.6247841	0.93673	0.79879

	Class: Trotting	Class: Walking	Class: While walking
Sensitivity	0.555556	0.46795	0.209302
Specificity	0.999784	0.99263	0.997170
Pos Pred Value	0.833333	0.68868	0.409091
Neg Pred Value	0.999136	0.98168	0.992631
Prevalence	0.001941	0.03365	0.009275
Detection Rate	0.001079	0.01575	0.001941
Detection Prevalence	0.001294	0.02286	0.004745
Balanced Accuracy	0.777670	0.73029	0.603236

2. Confusion Matrix – k-Nearest Neighbors (k = 5)

Confusion Matrix and Statistics

Prediction \ Reference	Accelerometer	Agnostic behavior	Breaking branches	Digging	Grooming	Ground	High	Low	Other	Resting	Ruminating
Accelerometer	3	0	0	0	0	0	0	0	0	0	0
Agnostic behavior	0	1	0	0	0	0	0	0	0	0	0
Breaking branches	0	1	35	0	0	0	0	1	3	0	0
Digging	0	0	0	0	0	0	0	0	0	0	0
Grooming	0	0	0	0	15	0	0	3	1	6	3
Ground	0	0	4	4	0	623	0	44	26	0	0
High	0	0	0	0	0	1	14	8	0	3	0
Low	0	0	3	0	16	25	15	889	28	14	4
Other	0	0	0	0	0	2	1	5	8	1	1
Resting	0	0	0	0	10	0	3	18	5	718	113
Ruminating	0	0	0	0	6	0	1	5	3	155	693
Running	1	1	0	0	0	0	0	0	0	0	0
Sleeping	0	0	0	0	0	1	0	3	2	9	2
Standing	0	0	0	0	3	2	2	69	15	7	7
Trotting	0	0	0	0	0	0	0	0	0	0	0
Walking	0	2	1	0	1	7	0	20	3	0	0
While walking	0	0	0	0	0	8	0	5	1	0	0

Prediction \ Reference	Running	Sleeping	Standing	Trotting	Walking	While walking
Accelerometer	0	0	0	0	0	0
Agnostic behavior	1	0	0	0	0	0
Breaking branches	0	0	0	4	0	0
Digging	0	0	0	0	0	0
Grooming	0	0	1	0	0	0
Ground	0	0	8	0	12	22
High	0	1	1	1	0	0
Low	0	2	94	0	56	8
Other	0	0	6	0	3	0
Resting	0	11	32	0	1	0
Ruminating	0	0	49	0	0	0
Running	0	0	0	0	0	0
Sleeping	0	184	6	0	0	0
Standing	1	2	305	0	7	0
Trotting	0	0	0	4	0	0
Walking	2	0	5	0	74	6
While walking	0	0	4	0	3	7

Overall Statistics

```
               Accuracy : 0.7707
                 95% CI : (0.7583, 0.7827)
    No Information Rate : 0.2308
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7246

 Mcnemar's Test P-Value : NA
```

Statistics by Class:

	Class: Accelerometer	Class: Agnostic behavior	Class: Breaking branches	Class: Digging	Class: Grooming
Sensitivity	0.7500000	0.2000000	0.813953	0.0000000	0.294118
Specificity	1.0000000	0.9997841	0.998040	1.0000000	0.996947
Pos Pred Value	1.0000000	0.5000000	0.795455	NaN	0.517241
Neg Pred Value	0.9997842	0.9991368	0.998258	0.9991372	0.992186
Prevalence	0.0008628	0.0010785	0.009275	0.0008628	0.011001
Detection Rate	0.0006471	0.0002157	0.007550	0.0000000	0.003236
Detection Prevalence	0.0006471	0.0004314	0.009491	0.0000000	0.006255
Balanced Accuracy	0.8750000	0.5998920	0.905997	0.5000000	0.645532

	Class: Ground	Class: High	Class: Low	Class: Other	Class: Resting	Class: Ruminating	Class: Running
Sensitivity	0.9312	0.388889	0.8308	0.084211	0.7864	0.8420	0.0000000
Specificity	0.9698	0.996739	0.9257	0.995816	0.9482	0.9426	0.9995682
Pos Pred Value	0.8385	0.482759	0.7704	0.296296	0.7881	0.7599	0.0000000
Neg Pred Value	0.9882	0.995225	0.9480	0.981124	0.9477	0.9651	0.9991368
Prevalence	0.1443	0.007765	0.2308	0.020492	0.1969	0.1775	0.0008628
Detection Rate	0.1344	0.003020	0.1918	0.001726	0.1549	0.1495	0.0000000
Detection Prevalence	0.1603	0.006255	0.2489	0.005824	0.1965	0.1967	0.0004314
Balanced Accuracy	0.9505	0.692814	0.8783	0.540013	0.8673	0.8923	0.4997841

	Class: Sleeping	Class: Standing	Class: Trotting	Class: Walking	Class: While walking
Sensitivity	0.92000	0.59687	0.4444444	0.47436	0.162791
Specificity	0.99482	0.97212	1.0000000	0.98951	0.995428
Pos Pred Value	0.88889	0.72619	1.0000000	0.61157	0.250000
Neg Pred Value	0.99639	0.95114	0.9989206	0.98184	0.992188
Prevalence	0.04314	0.11022	0.0019413	0.03365	0.009275
Detection Rate	0.03969	0.06579	0.0008628	0.01596	0.001510
Detection Prevalence	0.04465	0.09060	0.0008628	0.02610	0.006040
Balanced Accuracy	0.95741	0.78450	0.7222222	0.73193	0.579109

3.Confusion Matrix – Naive Bayes Classification

```
> confusionMatrix(nb_pred, test_y)
Confusion Matrix and Statistics

                    Reference
Prediction           Accelerometer Agnostic behavior Breaking branches Digging Grooming Ground High Low Other Resting Ruminating
  Accelerometer                  4                 2                 0       0        0      0    0   0     0       0          0
  Agnostic behavior              0                 1                 0       0        0     13    1   4     4       1          0
  Breaking branches              0                 1                32       1        0      4    3  17     8       2          0
  Digging                        0                 0                 5       0        0      6    0   0     1       0          0
  Grooming                       0                 0                 0       0       23      0    6  79     5      40          9
  Ground                         0                 0                 0       3        0    497    0  27    11       0          0
  High                           0                 0                 0       0        3      0   22  61     0       6          5
  Low                            0                 0                 0       0        6     21    2 548    10       5          2
  Other                          0                 0                 0       0        0      6    0   8     7       0          0
  Resting                        0                 0                 0       0        5      0    0  28     2     196         44
  Ruminating                     0                 0                 0       0        8      0    1  13     3     432        731
  Running                        0                 0                 0       0        0      0    0   0     0       0          0
  Sleeping                       0                 0                 0       0        3      2    0  43    15     215         27
  Standing                       0                 0                 0       0        2      1    0 132    13      16          5
  Trotting                       0                 0                 6       0        0      0    0   0     0       0          0
  Walking                        0                 0                 0       0        1      9    1  59     3       0          0
  While walking                  0                 1                 0       0        0    110    0  51    13       0          0
```

```
                   Reference
Prediction          Running Sleeping Standing Trotting Walking While walking
  Accelerometer           0        0        0        0       0             0
  Agnostic behavior       0        0        5        0       0             0
  Breaking branches       0        0        1        1       6             0
  Digging                 0        0        0        0       1             0
  Grooming                0        0        9        0       0             0
  Ground                  0        0        1        0       7            11
  High                    0        1       10        1       2             0
  Low                     0        1       17        0      45             6
  Other                   0        0        7        0       1             0
  Resting                 0        0       60        0       0             0
  Ruminating              1        3      113        0       0             0
  Running                 3        0        1        0       1             0
  Sleeping                0      193      103        0       0             0
  Standing                0        2      166        0       1             0
  Trotting                0        0        0        7       1             0
  Walking                 0        0       12        0      58             2
  While walking           0        0        6        0      33            24

Overall Statistics

               Accuracy : 0.5418
                 95% CI : (0.5274, 0.5563)
    No Information Rate : 0.2308
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.473

 Mcnemar's Test P-Value : NA
```

```
Statistics by Class:

                     Class: Accelerometer Class: Agnostic behavior Class: Breaking branches Class: Digging Class: Grooming
Sensitivity                     1.0000000                0.2000000                 0.744186      0.0000000        0.450980
Specificity                     0.9995682                0.9939538                 0.990420      0.9971934        0.967721
Pos Pred Value                  0.6666667                0.0344828                 0.421053      0.0000000        0.134503
Neg Pred Value                  1.0000000                0.9991318                 0.997588      0.9991348        0.993729
Prevalence                      0.0008628                0.0010785                 0.009275      0.0008628        0.011001
Detection Rate                  0.0008628                0.0002157                 0.006903      0.0000000        0.004961
Detection Prevalence            0.0012942                0.0062554                 0.016393      0.0028041        0.036885
Balanced Accuracy               0.9997841                0.5969769                 0.867303      0.4985967        0.709351
                     Class: Ground Class: High Class: Low Class: Other Class: Resting Class: Ruminating Class: Running
Sensitivity                 0.7429    0.611111     0.5121     0.073684        0.21468            0.8882      0.7500000
Specificity                 0.9849    0.980652     0.9678     0.995155        0.96266            0.8495      0.9995682
Pos Pred Value              0.8923    0.198198     0.8265     0.241379        0.58507            0.5602      0.6000000
Neg Pred Value              0.9578    0.996906     0.8686     0.980899        0.83329            0.9724      0.9997841
Prevalence                  0.1443    0.007765     0.2308     0.020492        0.19694            0.1775      0.0008628
Detection Rate              0.1072    0.004745     0.1182     0.001510        0.04228            0.1577      0.0006471
Detection Prevalence        0.1201    0.023943     0.1430     0.006255        0.07226            0.2815      0.0010785
Balanced Accuracy           0.8639    0.795882     0.7400     0.534420        0.58867            0.8688      0.8747841
                     Class: Sleeping Class: Standing Class: Trotting Class: Walking Class: While walking
Sensitivity                  0.96500         0.32485        0.777778        0.37179             0.558140
Specificity                  0.90803         0.95830        0.998487        0.98058             0.953407
Pos Pred Value               0.32113         0.49112        0.500000        0.40000             0.100840
Neg Pred Value               0.99827         0.91973        0.999567        0.97818             0.995680
Prevalence                   0.04314         0.11022        0.001941        0.03365             0.009275
Detection Rate               0.04163         0.03581        0.001510        0.01251             0.005177
Detection Prevalence         0.12964         0.07291        0.003020        0.03128             0.051337
Balanced Accuracy            0.93651         0.64158        0.888132        0.67619             0.755773
```

4.Model Convergence Output – Neural Network

```
> nn_model <- nnet(Modifiers ~ ., data = train, size = 10, maxit = 200)
# weights:  697
initial  value 28700.316671
iter  10 value 18070.455573
iter  20 value 13420.954917
iter  30 value 11635.973327
iter  40 value 10235.024507
iter  50 value 9291.404047
iter  60 value 8712.409479
iter  70 value 8303.200972
iter  80 value 7996.667271
iter  90 value 7778.591603
iter 100 value 7531.367683
iter 110 value 7362.345157
iter 120 value 7207.492240
iter 130 value 7057.842119
iter 140 value 6943.977096
iter 150 value 6838.988615
iter 160 value 6765.530840
iter 170 value 6667.446898
iter 180 value 6606.318490
iter 190 value 6546.565963
iter 200 value 6493.234168
final  value 6493.234168
stopped after 200 iterations
```

4.1 Confusion Matrix – Neural Network Predictions

```
> confusionMatrix(nn_pred, test_y)
Confusion Matrix and Statistics
```

Prediction \ Reference	Accelerometer	Agnostic behavior	Breaking branches	Digging	Grooming	Ground	High	Low	Other	Resting	Ruminating	Running	Sleeping	Standing	Trotting	Walking	While walking
Accelerometer	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Agnostic behavior	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Breaking branches	0	0	35	0	0	0	2	1	1	0	0	0	0	0	2	2	0
Digging	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Grooming	0	0	0	0	6	0	1	1	0	2	0	0	0	1	0	0	0
Ground	0	1	0	4	0	616	0	54	26	1	0	0	0	7	0	12	32
High	0	0	0	0	3	0	12	6	1	4	2	0	0	1	0	0	0
Low	0	0	4	0	22	41	18	916	32	16	4	2	3	66	1	46	6
Other	0	0	1	0	0	4	1	3	1	0	0	0	0	0	0	0	0
Resting	0	0	0	0	12	0	0	12	3	735	105	0	5	35	0	0	0
Ruminating	0	0	0	0	6	0	2	7	2	141	705	1	0	57	0	0	0
Running	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Sleeping	0	0	0	0	1	0	0	2	2	9	0	0	187	7	0	0	0
Standing	0	0	0	0	0	4	0	51	22	4	7	0	5	328	0	3	1
Trotting	0	2	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0
Walking	0	1	2	0	1	3	0	17	5	1	0	0	0	9	1	92	4
While walking	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0

```
Overall Statistics

               Accuracy : 0.786
                 95% CI : (0.7739, 0.7978)
    No Information Rate : 0.2308
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7425

 Mcnemar's Test P-Value : NA
```

Statistics by Class:

	Class: Accelerometer	Class: Agnostic behavior	Class: Breaking branches	Class: Digging	Class: Grooming
Sensitivity	1.0000000	0.2000000	0.813953	0.0000000	0.117647
Specificity	1.0000000	0.9997841	0.998258	1.0000000	0.998909
Pos Pred Value	1.0000000	0.5000000	0.813953	NaN	0.545455
Neg Pred Value	1.0000000	0.9991368	0.998258	0.9991372	0.990270
Prevalence	0.0008628	0.0010785	0.009275	0.0008628	0.011001
Detection Rate	0.0008628	0.0002157	0.007550	0.0000000	0.001294
Detection Prevalence	0.0008628	0.0004314	0.009275	0.0000000	0.002373
Balanced Accuracy	1.0000000	0.5998920	0.906106	0.5000000	0.558278

	Class: Ground	Class: High	Class: Low	Class: Other	Class: Resting	Class: Ruminating	Class: Running
Sensitivity	0.9208	0.333333	0.8561	0.0105263	0.8050	0.8566	0.2500000
Specificity	0.9655	0.996304	0.9268	0.9980181	0.9538	0.9434	1.0000000
Pos Pred Value	0.8181	0.413793	0.7782	0.1000000	0.8104	0.7655	1.0000000
Neg Pred Value	0.9864	0.994791	0.9555	0.9796801	0.9523	0.9682	0.9993528
Prevalence	0.1443	0.007765	0.2308	0.0204918	0.1969	0.1775	0.0008628
Detection Rate	0.1329	0.002588	0.1976	0.0002157	0.1585	0.1521	0.0002157
Detection Prevalence	0.1624	0.006255	0.2539	0.0021570	0.1956	0.1987	0.0002157
Balanced Accuracy	0.9431	0.664819	0.8914	0.5042722	0.8794	0.9000	0.6250000

	Class: Sleeping	Class: Standing	Class: Trotting	Class: Walking	Class: While walking
Sensitivity	0.93500	0.64188	0.555556	0.58974	0.0000000
Specificity	0.99527	0.97648	0.999568	0.99018	0.9995646
Pos Pred Value	0.89904	0.77176	0.714286	0.67647	0.0000000
Neg Pred Value	0.99706	0.95654	0.999136	0.98578	0.9907208
Prevalence	0.04314	0.11022	0.001941	0.03365	0.0092752
Detection Rate	0.04034	0.07075	0.001079	0.01984	0.0000000
Detection Prevalence	0.04487	0.09167	0.001510	0.02934	0.0004314
Balanced Accuracy	0.96513	0.80918	0.777562	0.78996	0.4997823