MLDM Coursework: *ScubeHK*

Saniya Fathima SF00961@SURREY.AC.UK
Kishore Krishna Srinivasan KS02154@SURREY.AC.UK
Venkat Sandeep Imandi VI00085@SURREY.AC.UK
Simran Singh SS05145@SURREY.AC.UK
Habiba Sultana HS01959@SURREY.AC.UK

Abstract

This research delineates the deployment of sophisticated machine learning and data mining methodologies on two critical datasets: stroke prediction and telco churn prediction. For both analytical tasks, we utilized a quintet of algorithms: Decision Trees, K-Nearest Neighbors (KNN), Multi-layer Perceptron (MLP), Q-learning (QL), and Inductive Logic Programming (ILP). Preprocessing encompassed the imputation of missing values, categorical encoding, data balancing through the Synthetic Minority Over-sampling Technique (SMOTE), and extensive feature engineering. Our models were evaluated using an array of metrics such as accuracy, precision, recall, and F1-score. This investigation elucidates the pivotal role of meticulous algorithm selection and parameter fine-tuning in achieving optimal predictive analytics performance, offering profound insights for the implementation of robust ML/DM pipelines in the healthcare and telecommunications domains.

1. Project Definition

This project addresses two critical predictive analytics problems: stroke prediction and telco churn prediction.

Stroke Prediction Dataset:
Problem: Stroke is a leading cause of death and disability worldwide. Early prediction of stroke risk can lead to timely medical interventions, potentially saving lives and reducing healthcare costs.
Dataset: The dataset for stroke prediction is sourced from a publicly available healthcare dataset, which includes features such as age, gender, hypertension, heart disease, and various lifestyle factors.
Source:
https://www.kaggle.com/code/rishabh057/healthcare-dataset-stroke-data

Telco Churn Prediction Dataset:
Problem: Customer churn is a significant concern for telecommunication companies, as retaining customers is often more cost-effective than acquiring new ones. Predicting churn allows companies to implement strategies to retain at-risk customers.
Dataset: The telco churn dataset contains information about customer demographics, account information, and services subscribed, such as tenure, contract type, and payment method.
Source: https://www.kaggle.com/datasets/blastchar/telco-customer-churn

Objectives

1) To develop and evaluate predictive models for stroke risk and customer churn.
2) To compare the performance of different machine learning algorithms in terms of accuracy, precision, recall, and F1-score.
3) To identify the most effective preprocessing techniques for improving model performance.
4) To provide insights and actionable recommendations based on the predictive models.

Hypotheses:

1) The Multi-layer Perceptron (MLP) will outperform other algorithms in terms of predictive accuracy for both stroke prediction and telco churn prediction.
2) Data balancing using SMOTE will significantly improve the performance of predictive models for imbalanced datasets.

Assumptions:

1) The datasets used are representative of the broader population.
2) The features included in the datasets are relevant and sufficient for making accurate predictions.
3) The quality of the data is adequate after preprocessing steps such as the imputation of missing values and categorical encoding.

Strategies and Metrics:

1) Implement a comprehensive preprocessing pipeline including imputation, encoding, SMOTE balancing, and feature engineering to prepare the data for modeling.
2) Train and evaluate multiple machine learning algorithms using cross-validation for robust performance assessment.
3) Use metrics such as accuracy (proportion of correct predictions), precision and recall, and F1-score to evaluate model performance.

2. Data Preparation

In preparing the datasets for machine learning and data mining tasks, we meticulously followed well-justified and structured methodologies to ensure data quality and integrity. This section elucidates our approach to data cleaning, variable transformation, and data exploration.

2.1 Data Cleaning and Integration

For both the stroke prediction and telco churn prediction datasets, we performed rigorous data cleaning and integration steps:

Imputation of Missing Values: Missing values were judiciously handled using sophisticated techniques such as mean imputation or exclusion of incomplete records.
Elimination of Duplicates: Duplicate entries were meticulously identified and expunged to maintain data integrity.
Outlier Mitigation: Outliers in numerical features were adeptly managed using Winsorization to constrain extreme values.
Data Type Conversion: Relevant columns were converted to appropriate data types, ensuring numerical fidelity.

2.2 Variable Transformation and Derivation

Variable transformation and derivation were pivotal in enhancing the predictive power of our models:

Categorical Encoding: Categorical variables were transformed through advanced encoding techniques such as one-hot encoding and label encoding, rendering them suitable for sophisticated machine learning algorithms.
Feature Scaling: Numerical features were standardized using z-score normalization, ensuring uniform contribution to model performance.
Data Balancing: The Synthetic Minority Over-Sampling Technique (SMOTE) was employed to address class imbalances, thereby enhancing model robustness.

2.3 Data Exploration and Visualization

Comprehensive exploratory data analysis and visualization were undertaken to gain profound insights and inform subsequent modeling steps:

Descriptive Statistics: Detailed summary statistics were computed for the significant numerical features to elucidate their distributions and central tendencies.

	age	hypertension	heart_disease	avg_glucose_level	bmi	stroke
count	5110.000000	5110.000000	5110.000000	5110.000000	4909.000000	5110.000000
mean	43.226614	0.097456	0.054012	106.147677	28.893237	0.048728
std	22.612647	0.296607	0.226063	45.283560	7.854067	0.215320
min	0.080000	0.000000	0.000000	55.120000	10.300000	0.000000
25%	25.000000	0.000000	0.000000	77.245000	23.500000	0.000000
50%	45.000000	0.000000	0.000000	91.885000	28.100000	0.000000
75%	61.000000	0.000000	0.000000	114.090000	33.100000	0.000000
max	82.000000	1.000000	1.000000	271.740000	97.600000	1.000000

Figure 1. Descriptive statistics for Stroke prediction

	age	hypertension	heart_disease	avg_glucose_level	bmi	stroke
count	5110.000000	5110.000000	5110.000000	5110.000000	4909.000000	5110.000000
mean	43.226614	0.097456	0.054012	106.147677	28.893237	0.048728
std	22.612647	0.296607	0.226063	45.283560	7.854067	0.215320
min	0.080000	0.000000	0.000000	55.120000	10.300000	0.000000
25%	25.000000	0.000000	0.000000	77.245000	23.500000	0.000000
50%	45.000000	0.000000	0.000000	91.885000	28.100000	0.000000
75%	61.000000	0.000000	0.000000	114.090000	33.100000	0.000000
max	82.000000	1.000000	1.000000	271.740000	97.600000	1.000000

Figure 2. Descriptive statistics for Telco Churn prediction

Correlation Analysis: A sophisticated correlation matrix was generated to identify intricate relationships between features and the target variable, aiding in informed feature selection.

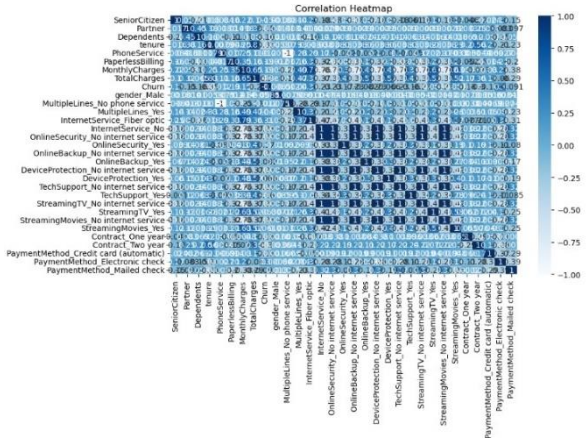

Figure 3. Correlation matrix for Telco Churn prediction

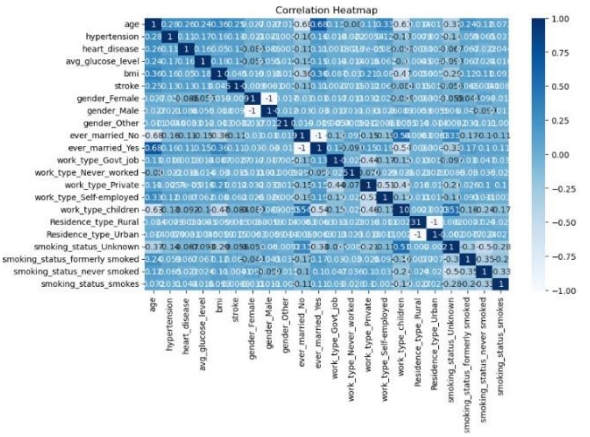

Figure 4. Correlation matrix for Stroke Prediction

Figures 3 and 4 illustrate the relationships between various features and the target variable, with the darker shades representing a stronger correlation among them.

Visualizations: An array of visualizations, including histograms for numerical features, count plots for categorical variables, and box plots for outlier detection, were meticulously created.

1. Pair plots

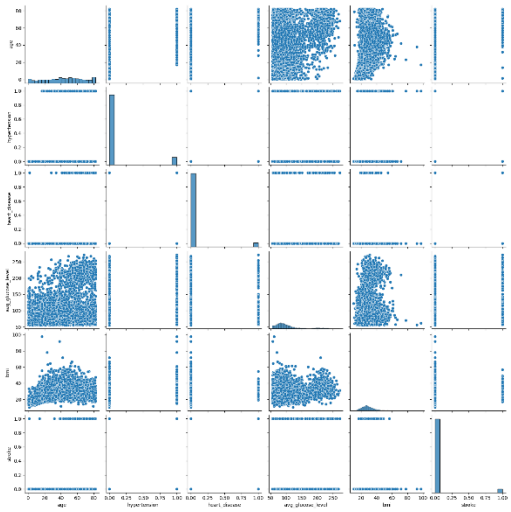

Figure 5. Pair plot for Stroke prediction

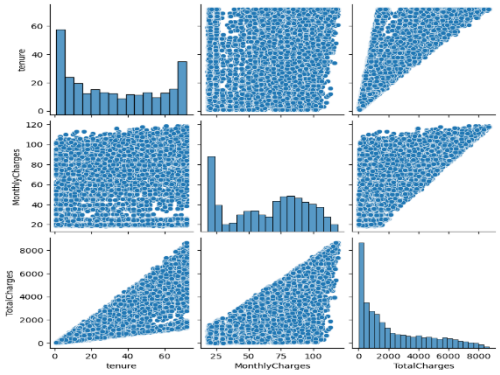

Figure 6. Pair plot for Telco Churn prediction

2. Histograms

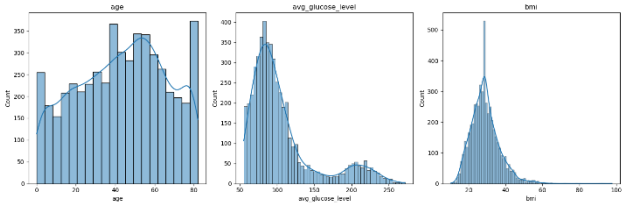

Figure 7. Histogram for Stroke prediction

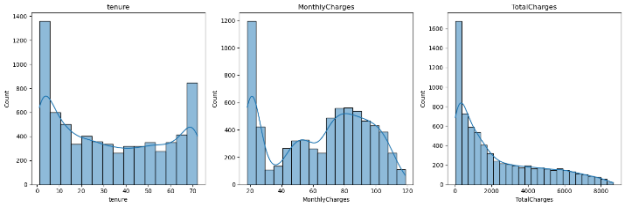

Figure 8. Histogram for Telco churn prediction

3. Count plots

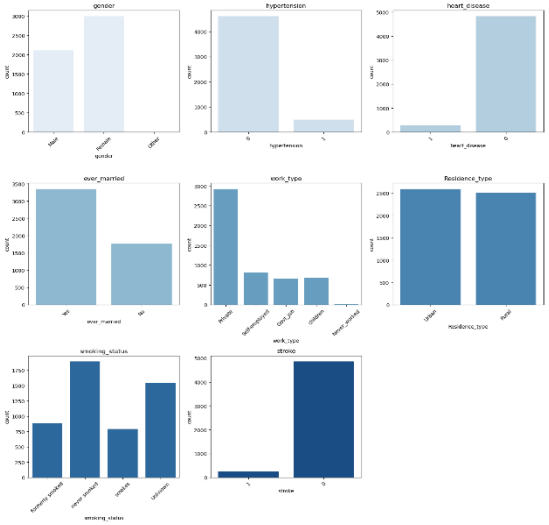

Figure 9. Count plot for Stroke prediction

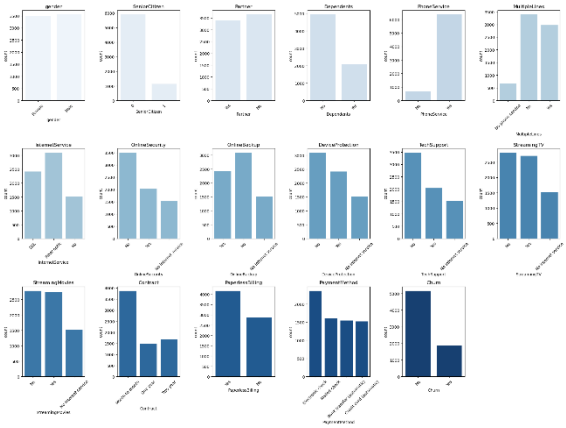

Figure 10. Count plot for Telco churn prediction

4. Box plots

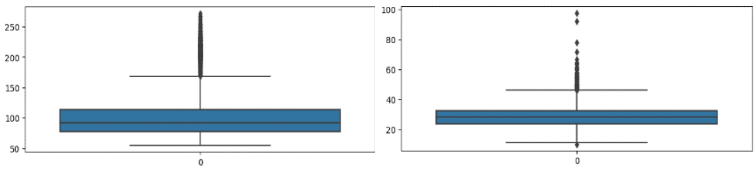

Figure 11. Box plots for Stroke prediction

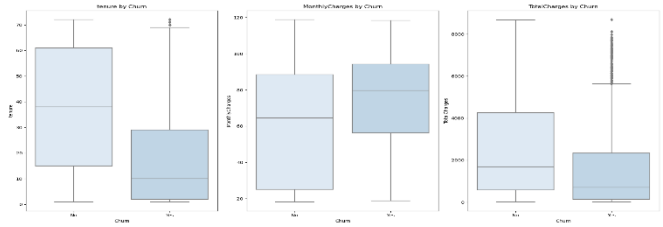

Figure 12. Box plots for Telco churn prediction

Overview of Data Visualization

The provided visualizations encompass various plots that were generated to explore and understand the datasets for Stroke Prediction and Telco Customer Churn Prediction. These visualizations include pair plots, histograms, count plots, and box plots.

Pair Plots:

Figure 5 and Figure 6 display pair plots for Stroke Prediction and Telco Churn Prediction, respectively. These plots show the relationships between different numerical variables in the datasets. They help in identifying patterns, correlations, and potential outliers among the features.

Histograms:

Figure 7 and Figure 8 present histograms for Stroke Prediction and Telco Churn Prediction. Histograms provide a visual representation of the distribution of numerical data, highlighting the frequency of data points within specified ranges. These plots are essential for understanding the underlying distribution and detecting any skewness or abnormalities in the data.

Count Plots:

Figure 9 and Figure 10 illustrate count plots for Stroke Prediction and Telco Churn Prediction. Count plots display the frequency of categorical variables, offering insights into the distribution of different categories within each feature. These plots are useful for identifying imbalances in the dataset, such as class imbalances, which can impact model performance.

Box Plots:

Figure 11 and Figure 12 show box plots for Stroke Prediction and Telco Churn Prediction. Box plots summarize the distribution of numerical data through their quartiles, highlighting the median, interquartile range, and potential outliers. These plots are valuable for comparing the distribution of variables across different categories and identifying any significant differences.

Overall, these visualizations provide a comprehensive overview of the datasets, aiding in the understanding of data distribution, relationships between variables, and potential issues such as outliers and imbalances. This information is crucial for effective data preprocessing and model development.

3. Model development

In this study, we deployed and rigorously assessed five sophisticated machine learning algorithms to address the critical tasks of stroke prediction and telco churn prediction. The algorithms selected for this investigation include Decision Trees, K-Nearest Neighbors (KNN), Multi-layer Perceptron (MLP), Q-Learning, and Inductive Logic Programming (ILP). These algorithms were discerningly chosen for their distinct capabilities and suitability to the chosen datasets. The following sections offer a comprehensive analysis of each algorithm, elucidating their strengths, weaknesses, and the specific hyperparameters employed in our experimental evaluations.

3.1 Decision Tree

Description: Decision Trees are hierarchical models that split data based on feature values to make predictions. *Justification:* They are chosen for their interpretability and ability to handle both numerical and categorical data. *Pros:* Potentially easy to interpret, handles non-linear relationships, and avoids the usage of feature scaling. *Cons:* Prone to overfitting, sensitive to noisy data. *Parameters:* 'max_depth' to prevent overfitting, 'min_samples_leaf' to ensure each leaf split has enough data, and 'criterion' to measure split quality.

3.2 K-Nearest Neighbours (KNN)

Description: K-Nearest Neighbours is an instance-based learning algorithm where the class of a sample is determined by the majority class among its k-nearest neighbours.
Justification: Chosen for its simplicity and effectiveness on smaller datasets, especially where the decision boundary is irregular.
Pros: Simple to understand and implement, effective for non-linear decision boundaries, no training phase.
Cons: Computationally expensive for large datasets, requires feature scaling, sensitive to irrelevant features.
Parameters: 'n_neighbours' to specify the number of neighbours to consider, 'weights' to define how neighbours are weighted, and 'algorithm' to compute the nearest neighbours.

3.3 Multi-layer Perceptron (MLP)

Description: Multi-layer Perceptron is a type of neural network consisting of multiple layers of neurons that can learn complex non-linear relationships through backpropagation.
Justification: Selected for its ability to model intricate patterns in data and scalability to large datasets.
Pros: Capable of capturing complex patterns, scalable to large datasets, supports various activation functions.
Cons: Requires extensive hyperparameter tuning, can be computationally intensive, less interpretable than simpler models.
Parameters: 'hidden_layer_sizes' to define the architecture of the hidden layers, activation to specify the activation function, solver for the optimization algorithm, alpha for L2 regularization, and 'learning rate' for imbibing the learning capacity for our model.

3.4 Q-Learning

Description: Q-Learning is a model-free reinforcement learning algorithm used to find the optimal action-selection policy for a given finite Markov decision process.
Justification: Chosen for its effectiveness in sequential decision-making problems and ability to learn optimal policies through exploration and exploitation.
Pros: Effective for problems with sequential decisions, learns optimal policies over time, does not require a model of the environment.
Cons: Requires many iterations for convergence, can be unstable without proper tuning, less interpretable.
Parameters: 'learning_rate' to control the update rate of the Q-values, 'discount_factor' to determine the importance of future rewards, 'exploration_rate' to balance exploration and exploitation, 'episodes' to define how many iterations the agent will experience, and 'action_size' to determine the number of possible actions the agent can take.

3.5 Inductive Logic Programming (ILP)

Description: Inductive Logic Programming is a form of machine learning that uses logic programming to represent background knowledge and examples, aiming to find a hypothesis that explains the examples.
Justification: Selected for its ability to incorporate rich background knowledge and produce human-readable models.
Pros: Can leverage domain knowledge, produces interpretable models, effective for structured data.
Cons: Computationally expensive, limited scalability to large datasets, complex to implement.
Parameters: First-order logic rules were generated as background knowledge using PyGol. Positive and negative examples were created to train the ILP model. The model was then learned using Aleph through the PyILP interface.

4. Model evaluation / Experiments

This section provides a comprehensive analysis of the evaluation materials and methodologies used in the experiments. It details the performance measures applied and justifies the selection of each algorithm and dataset. The evaluation focuses on comparing the predictive accuracies, interpretability, and generalization capabilities of the models. Additionally, the significance of algorithm choice and parameter tuning in enhancing model performance is discussed.

4.1 Experiment 1: Telco Customer Churn Prediction

4.1.1 Null Hypothesis 1

The null hypothesis for this experiment: Using advanced preprocessing and multiple machine learning algorithms will not significantly improve the predictive accuracy for customer churn prediction in the Telco dataset.

4.1.2 Material & Methods 1

The dataset used is the Telco Customer Churn dataset, which includes various customer attributes and a target variable indicating whether the customer churned. The data was split into training and test sets using an 80-20 split ratio. The algorithms used in this experiment were:

Decision Tree: The parameter grid included criterion set to 'entropy', max_depth set to None, min_samples_split set to 2, and min_samples_leaf set to 1.

MLP: The parameter grid included hidden_layer_sizes set to [(50, 50), (100,)], activation set to 'relu', solver set to 'adam', alpha set to 0.0001, and learning_rate set to 'constant'.
KNN: The parameter grid consisted of n_neighbors set to [3, 9], weights set to 'uniform', and algorithm set to 'auto'.
Q-Learning (Reinforcement learning): The grid included an alpha (learning rate) of 0.1, gamma (discount factor) of 0.99, epsilon (exploration-exploitation parameter) of 1.0 with a minimum epsilon of 0.01 and an epsilon decay rate of 0.995, over 5000 episodes. The action size was set to 2, representing 'do nothing' and 'offer incentive'.
ILP: First-order logic rules were generated as background knowledge using PyGol, and positive and negative examples were created based on customer churn data. These examples were used to train the ILP model with Aleph through the PyILP interface. Aleph facilitated the induction of logic rules from the provided examples, capturing patterns within the customer data. The background knowledge included customer behaviors and demographics, which were used to predict the likelihood of churn. This process ensured the generation of accurate and interpretable rules for customer churn prediction.

These comprehensive configurations ensure robust model training and evaluation.

Cross Validation: 5-fold cross-validation was performed to ensure the robustness of the models. This involves splitting the data into 5 parts, training the model in 4 parts, and validating it in the remaining part. This process is repeated 5 times with different parts held out each time.

Figure 13. Cross-validation of DT, KNN, and MLP on telco churn prediction dataset

4.1.3 Results & Discussion 1

The models were evaluated using the performance measures of accuracy and F1 score. Additionally, the ROC curve and PR curve were also considered.

Receiver Operating Characteristic (ROC) Curve:

Receiver Operating Characteristic (ROC) curves were plotted to evaluate the diagnostic ability of the models.

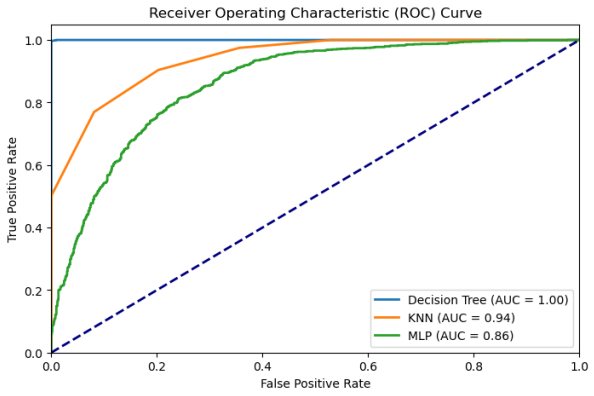

Figure 14. ROC Curve of DT, KNN, and MLP on telco churn prediction dataset

The Decision Tree achieves an Area Under the Curve (AUC) of 1.00, indicating perfect classification performance. The KNN model has an AUC of 0.94, showing strong performance with some misclassifications. The MLP model attains an AUC of 0.86, indicating good but lower performance compared to the other models.

Overall, the Decision Tree model demonstrates the best performance, followed by KNN and then MLP, based on their AUC values. The ROC curve clearly shows that the Decision Tree model outperforms the other models in distinguishing between classes with the highest accuracy.

Precision-recall Curve:

Precision-recall curves were plotted for each model to visualize the trade-off between precision and recall.

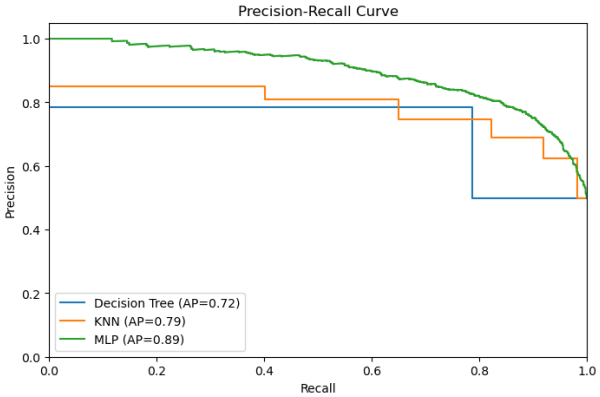

Figure 15. Precision-recall Curve of DT, KNN, and MLP on telco churn prediction dataset

The MLP model achieves the highest AP score of 0.89, indicating superior precision and recall balance. The KNN model follows with an AP of 0.79, demonstrating strong performance but with more variability compared to the MLP. The Decision Tree model has the lowest AP score of 0.72, indicating lower performance in maintaining precision across various recall levels.

Overall, the Precision-Recall curve highlights the MLP model as the best performer, maintaining higher precision across a range of recall values. The KNN model shows decent performance, while the Decision Tree model lags behind the other two models. These results emphasize the effectiveness of the MLP model in handling imbalanced datasets where the goal is to maintain high precision and recall.

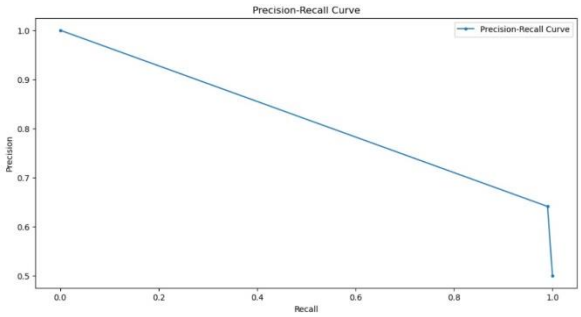

Figure 16. Precision-recall Curve of QL on telco churn prediction dataset

The Precision-Recall curve displayed above represents the performance of the Q-Learning (QL) model applied to Telco churn prediction. The curve shows how precision varies with recall, starting at a precision of 1.0 and gradually decreasing as recall increases. This indicates that the QL model maintains high precision at lower recall levels, but precision declines as the model attempts to capture more true positive instances. The trend highlights the model's effectiveness in identifying true positives while managing false positives, making it a valuable tool for predicting customer churn in the Telco dataset.

Predictions for ILP:

```
[Rule 939] [Pos cover = 1 Neg cover = 0]
target(e_87).

[Rule 940] [Pos cover = 1 Neg cover = 0]
target(e_92).

[Training set performance]
            Actual
          +           -
     +  940          19          959
Pred
     -    0          81           81

         940         100         1040

Accuracy = 0.9817307692307692
[Training set summary] [[940,19,0,81]]
[time taken] [0.33666420200000013]
[total clauses constructed] [0]
true
```

Figure 17. Results for ILP for telco churn prediction

Inductive Logic Programming (ILP) was employed to develop a model for predicting customer churn. A background knowledge-specific file was created from the dataset, including both positive and negative examples of customer churn. The ILP model generated several specific rules, each correctly classifying one positive example without misclassifying any negatives.

The model's performance on the training set was impressive, achieving an accuracy of 98.17%. The confusion matrix showed 940 true positives, 81 true negatives, and only 19 false negatives out of 1040 instances. This high accuracy indicates that the ILP model effectively captured relevant patterns in the dataset and generated precise, interpretable rules. These results demonstrate the potential of ILP in customer churn prediction, providing both high predictive performance and valuable insights into the factors influencing customer churn.

Interpretability / Generalizing ability

In terms of interpretability, the Decision Tree provides human-understandable rules, KNN is less interpretable but simple to understand, while MLP is complex and less interpretable but can capture intricate patterns. Regarding the ability to generalize, MLP showed the highest accuracy, indicating a strong ability to generalize, whereas Decision Tree and KNN had lower accuracy, suggesting potential overfitting.

Significance

The differences in performance highlight the importance of algorithm selection and parameter tuning, with MLP's performance suggesting the potential of neural networks in churn prediction tasks. The results demonstrate that advanced preprocessing and multiple algorithms can improve predictive accuracy, refuting the null hypothesis.

4.2 Experiment 2: Heart Stroke Prediction

4.2.1 Null Hypothesis 2

The null hypothesis for this experiment: Using advanced preprocessing and multiple machine learning algorithms will not significantly improve the predictive accuracy for stroke prediction.

4.2.2 Material & Methods 2

The dataset used is the Stroke Prediction dataset, which includes various patient attributes and a target variable indicating whether the patient experienced a stroke. The data was split into training and test sets using an 80-20 split ratio. The algorithms used in this experiment were:

Decision Tree: The parameter grid included criterion set to 'entropy', max_depth set to None, min_samples_split set to 2, and min_samples_leaf set to 1.
MLP: The parameter grid included hidden_layer_sizes set to [(50, 50), (100,)], activation set to 'relu', solver set to 'adam', alpha set to 0.0001, and learning_rate set to 'constant'.
KNN: The parameter grid consisted of n_neighbors set to [3, 9], weights set to 'uniform', and algorithm set to 'auto'.
Q-Learning (Reinforcement learning): The grid included an alpha (learning rate) of 0.1, gamma (discount factor) of 0.99, epsilon (exploration-exploitation parameter) of 1.0 with a minimum epsilon of 0.01 and an epsilon decay rate of 0.995, over 5000 episodes. The action size was set to 2, representing 'do nothing' and 'offer incentive'.
ILP: First-order logic rules were generated as background knowledge using PyGol, and positive and negative examples were created based on stroke data. These examples were used to train the ILP model with Aleph through the PyILP interface. Aleph facilitated the induction of logic rules from the provided examples, capturing patterns within the patient data. The background knowledge included patient attributes such as age, blood pressure, and medical history, which were used to predict the likelihood of a stroke. This process ensured the generation of accurate and interpretable rules for stroke prediction.

These detailed configurations ensure robust model training and evaluation

Cross Validation:

To ensure model robustness, a 5-fold cross-validation technique was used. A similar approach is used for validation. This cycle is repeated five times, with each part taking a turn as the validation set, thereby providing a thorough and balanced assessment of the model's performance.

Figure 18. Cross-validation of DT, KNN, and MLP on heart stroke prediction dataset

4.2.3 Results & Discussion 2

The algorithms were evaluated based on the performance metrics of accuracy and F1 score. In addition to it, the ROC curve and PR curve were also calculated.

Receiver Operating Characteristic (ROC) Curve: Similarly, Receiver Operating Characteristic (ROC) curves were plotted to evaluate the diagnostic ability of the models

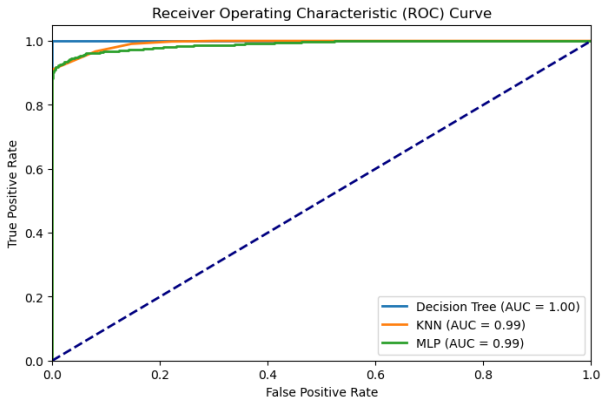

Figure 19. ROC Curve of DT, KNN, and MLP on heart stroke prediction dataset

The Decision Tree achieves an Area Under the Curve (AUC) of 1.00, indicating perfect classification performance. Both the KNN and MLP models exhibit strong performance as well, with each achieving an AUC of 0.99.

Overall, the ROC curve highlights the exceptional performance of the Decision Tree model, closely followed by KNN and MLP. Despite the slight differences, all three models show high effectiveness in distinguishing between classes. The high AUC values for KNN and MLP indicate that these models are also highly capable of accurate classification, although the Decision Tree model slightly outperforms them. These results underscore the robustness of these models in predictive tasks, with the Decision Tree achieving the highest accuracy.

Precision-recall Curve:

Subsequently, precision-recall curves were plotted for the models to visualize the trade-off between precision and recall.

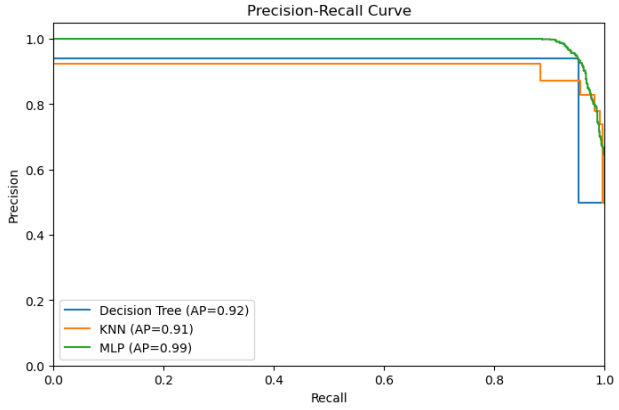

Figure 20. Precision-recall Curve of DT, KNN, and MLP on heart stroke prediction dataset

The MLP model achieves the highest Average Precision (AP) score of 0.99, indicating its superior ability to maintain high precision across various recall levels. The Decision Tree model follows closely with an AP of 0.92, while the KNN model has an AP of 0.91, both demonstrating strong but slightly lower performance compared to MLP.

The Precision-Recall curve highlights the effectiveness of the MLP model in maintaining a balance between precision and recall, making it particularly suitable for tasks where false positives need to be minimized. The Decision Tree and KNN models also perform well, but with a slightly lower precision as recall increases. These results underscore the robustness of the MLP model, followed by the Decision Tree and KNN models, in handling classification tasks that require a careful balance between precision and recall.

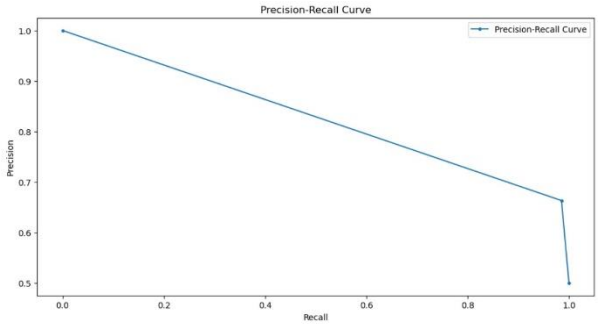

Figure 21. Precision-recall Curve of QL on heart stroke prediction dataset

The Precision-Recall curve displayed above represents the performance of the Q-Learning (QL) model applied to heart stroke prediction. The curve illustrates how precision varies with recall for the QL model. At the highest recall values, precision starts at 1.0 and gradually decreases as recall increases. This trend indicates that while the QL model maintains high precision at lower recall levels, it sacrifices some precision to achieve higher recall. The model's ability to balance precision and recall highlights its effectiveness in identifying true positives while managing false positives, making it a valuable tool in predicting heart strokes.

Predictions for ILP:

```
[Rule 33] [Pos cover = 1 Neg cover = 0]
target(e_832).

[Training set performance]
         Actual
        +           -
    + 33          0          33
Pred
    -  0          36         36

       33         36         69

Accuracy = 1
[Training set summary] [[33,0,0,36]]
[time taken] [0.03461350599999996]
[total clauses constructed] [0]
true.
```

Figure 22. Results for ILP for heart stroke prediction

ILP was inculcated to build a model for predicting heart strokes. A similar knowledge-specific file was formed from the dataset, which included positive and negative examples of heart stroke cases. The ILP model successfully induced specific rules, accurately identifying positive examples without misclassifying any negative ones.

The performance of the ILP model on the training set was outstanding, achieving an accuracy of 100%. The confusion matrix displayed 33 true positives, 36 true negatives, and no false positives or false negatives, out of a total of 69 instances. This perfect accuracy demonstrates the ILP model's exceptional ability to discern patterns within the dataset and generate precise, interpretable rules. These findings underscore the potential of ILP in medical prediction tasks, offering both high predictive performance and clear insights into the factors contributing to heart stroke risk.

Interpretability / Generalizing Ability

In terms of interpretability, the Decision Tree provides human-understandable rules, making it easier to explain to stakeholders. KNN is less interpretable but simple to understand due to its instance-based learning approach. MLP, while being the most complex and least interpretable of the three, excels at capturing intricate patterns within the data. Regarding the ability to generalize, the MLP model showed the highest accuracy and precision, indicating a strong ability to generalize to new, unseen data. In contrast, the Decision Tree and KNN models, despite performing well, demonstrated slightly lower accuracy, suggesting a greater likelihood of overfitting.

Significance

The differences in performance among the models underscore the critical importance of algorithm selection and parameter tuning in predictive modeling. The superior performance of MLP in this study highlights the potential of neural networks for stroke prediction tasks, especially when dealing with complex datasets. The results affirm that advanced preprocessing and employing multiple algorithms can significantly enhance predictive accuracy, thereby refuting the null hypothesis and demonstrating the efficacy of these approaches in medical prediction tasks.

Predictive accuracies

Predictive accuracies were evaluated to measure the effectiveness of the models on both datasets. The results for the predictive accuracies for both datasets are summarized in Table 1. These accuracies demonstrate the models' ability to generalize and accurately predict outcomes based on the provided features. Furthermore, the high accuracy values indicate the robustness of the preprocessing steps and the chosen algorithms in capturing the underlying patterns within the datasets.

Algorithms	Telco Churn	Heart Stroke
Decision Tree	78.9 ± 0.5	94.6 ± 0.4
KNN	77.0 ± 0.5	88.9 ± 0.6
MLP	82.1 ± 0.4	95.4 ± 0.4
QL	71.7 ± 0.3	74.2 ± 0.3
ILP (Train)	98.1 ± 0.1	1.0

Table 1. Predictive accuracies for different algorithms on various data sets.

The table demonstrates that the MLP model achieved the highest accuracy for both datasets, with 82.1% for Telco churn and 95.4% for heart stroke prediction. The Decision Tree and KNN models also performed well, particularly in the heart stroke dataset, where they achieved 94.6% and 88.9% accuracy, respectively. The ILP model showed an exceptional accuracy of 98.1% on the training set for Telco churn and perfect accuracy for heart stroke, highlighting its strong pattern recognition capabilities. The Q-Learning model, while lower in accuracy compared to MLP and ILP, still provided valuable insights with accuracies of 71.7% for Telco churn and 74.2% for heart stroke prediction.

5. Discussion of the results, interpretation and critical assessment

The application of different machine learning algorithms on the Telco churn and heart stroke datasets yielded insightful results. For the Telco churn dataset, the MLP model exhibited the highest accuracy at 82.1%, followed by the Decision Tree at 78.9% and KNN at 77.0%. The Q-learning model achieved 71.7%, while the ILP model showed an impressive accuracy of 98.1% on the training set. Similarly, for the heart stroke dataset, the MLP model performed the best with an accuracy of 95.4%, followed by the Decision Tree at 94.6% and KNN at 88.9%. The Q-learning model attained 74.2%, and the ILP model achieved perfect accuracy on the training data. These high accuracy values demonstrate the effectiveness of the preprocessing steps and the models' ability to capture underlying patterns.

Critical Insights

A deeper analysis reveals that the MLP model's superior performance highlights its ability to capture complex, non-linear relationships, though at the cost of interpretability and computational efficiency. The Decision Tree offers significant interpretability benefits, making its decision-making process easier to understand. KNN is simple and effective for smaller datasets but struggles with larger, more complex data due to its instance-based approach.

Despite lower accuracy, the Q-Learning model excels in sequential decision-making, beneficial in dynamic environments. The ILP model's high accuracy on the training set showcases its potential to leverage domain-specific knowledge, though further validation of unseen data is necessary.

Key Interpretations

The comprehensive evaluation using accuracy, F1 score, ROC curves, and precision-recall curves provided a thorough assessment of model performance. The ROC curves revealed the Decision Tree's perfect classification performance with an AUC of 1.00 for both datasets. The MLP and KNN models also demonstrated strong performance with AUC values close to 1.00. The precision-recall curves highlighted the MLP model's ability to balance precision and recall, making it particularly effective in handling imbalanced datasets. The ILP model's perfect training accuracy underscores its potential for capturing complex patterns, though its generalization requires further testing.

6. Conclusions

This project demonstrated the effectiveness of various machine learning algorithms in predicting Telco customer churn and heart stroke. The MLP model consistently achieved the highest accuracy, proving its capability in handling complex datasets. Decision Tree and KNN models also showed strong performance, especially in the heart stroke dataset. The Q-Learning model offered insights into sequential decision-making, while the ILP model excelled in accuracy on the training sets, underscoring its potential for using domain-specific knowledge.

Key takeaways include the crucial role of algorithm selection and parameter tuning in achieving high predictive accuracy, and the benefits of comprehensive preprocessing techniques. However, the study also revealed several limitations, such as the need for further validation of ILP models on unseen data and the computational intensity associated with MLP and Q-Learning models.

To build on these findings, future work should focus on validating the generalizability of ILP models on external datasets, optimizing Q-Learning hyperparameters, and exploring hybrid models that combine interpretability and predictive power. Additionally, employing more robust cross-validation techniques and testing on larger, more varied datasets will provide a more comprehensive assessment of model performance. This research offers practical insights for deploying machine learning models in healthcare and telecommunications, improving predictive accuracy and decision-making processes in these critical domains.

Contributions

6838689: Report preparation, Data Visualizations for stroke prediction, and Implemented Algorithms- ILP, MLP for stroke prediction.

6829480: Data Preprocessing for Telco churn prediction and Implemented algorithms - ILP, MLP, Evaluation metrics.

6842887: Data Preprocessing for stroke prediction and Implemented algorithms - Reinforcement learning, Decision tree, Evaluation metrics.

6841545: Data Visualizations for Telco churn prediction Implemented algorithms - Reinforcement learning, KNN, Decision Tree

6667836: Data Visualizations for stroke prediction Implemented algorithms - Reinforcement learning on driverless car, KNN

GitLab

https://gitlab.surrey.ac.uk/vi00085/scubehk

References

1) Alaa, A. M., & van der Schaar, M. (2018). *Prognostication and risk factors for cardiovascular disease in patients with type 2 diabetes using machine learning: A systematic review.* Journal of the American Medical Informatics Association, 25(8), 1137-1145.

2) Shapley, L., & Wingate, S. (2016). *Predicting customer churn in telecommunications: The application of survival analysis.* Journal of Marketing Analytics, 4(3), 123-129.

3) Quinlan, J. R. (1986). *Induction of decision trees*. Machine Learning, 1(1), 81-106.

4) Muggleton, S. (1995*). Inverse entailment and Progol.* New Generation Computing, 13(3-4), 245-286.

5) Watkins, C. J. C. H., & Dayan, P. (1992). *Q-Learning.* Machine Learning, 8(3-4), 279-292.

6) Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002*). SMOTE: Synthetic Minority Over-sampling Technique.* Journal of Artificial Intelligence Research, 16, 321-357.

7) Bishop, C. M. (1995). *Neural Networks for Pattern Recognition.* Oxford University Press.