Milestone 1 – Final Project
COMPSCI 4AL3

Team

1. **Member 1 (Avyya Singh | singa274):**
   - Implement machine learning models (Decision Tree, SVM).
   - Data Preprocessing & Feature Engineering
   - Visualize, Report and document the model.
2. **Member 2 (Jasraj Singh Johal | johalj11):**
   - Implement machine learning models (Decision Tree, K-Nearest Neighbor).
   - Data Preprocessing & Feature Engineering
   - Visualize, Report and document the model.
3. **Member 3 (Javier Afonso | afonsj2):**
   - Implement machine learning models (Logistic Regression, Decision Tree).
   - Data Preprocessing & Feature Engineering
   - Visualize, Report and document the model.

Context

Problem: Predict the likelihood of cardiovascular disease in patients based on clinical and demographic factors.

Why is it a challenging and important problem?

- Cardiovascular diseases (CVD) are a leading cause of mortality worldwide, and early detection is critical.
- The complexity arises from multiple factors influencing CVD, including medical, genetic, and lifestyle parameters.
- Imbalanced data, outliers, and overlapping feature distributions add difficulty to accurate prediction.

How can it be solved through machine learning?

- Machine learning enables predictive analytics by identifying patterns in large datasets.
- Techniques like Logistic Regression, Decision Trees, KNN, and SVM can classify individuals as at-risk or not based on input features.

What aspects of the problem are you going to solve?

- Develop a predictive model for diagnosing CVD.
- Identify key predictors contributing to cardiovascular risk.

Relevance to the world:

- Accurate prediction reduces medical emergencies by facilitating timely interventions.
- Enhances preventive healthcare, thereby decreasing healthcare costs.

Dataset

Source: Kaggle - Cardiovascular Disease Dataset

Description:

- **Input Features:** Age, Height, Weight, Gender, Blood Pressure, Cholesterol, Glucose levels, Smoking, Alcohol consumption, Physical activity.
- **Target Variable:** Cardio (0 = no disease, 1 = disease).
- **Size:** 70,000 samples, 12 features.

Metadata:

- Objective features: Age, Height, Weight, Gender.
- Examination features: Blood Pressure, Cholesterol, Glucose.
- Subjective features: Smoking, Alcohol intake, Physical activity.

Subset or Entire Dataset: Entire dataset will be used after careful preprocessing to retain maximum information.

Proposed Solution

Is the problem predictive or inferential? The problem is predictive as it uses historical medical and lifestyle data to classify future patients as having heart disease (1) or not (0).

Variables:

- **Features:** Age, Height, Weight, Gender, Blood Pressure, Cholesterol, Glucose, Smoking, Alcohol, Physical Activity, BMI.
- **Target:** Presence of cardiovascular disease (Cardio: binary).

ML Techniques:

- **Logistic Regression:** A baseline model for binary classification.
- **Decision Tree:** Handles non-linear relationships and interprets feature importance.
- **K-Nearest Neighbor (KNN):** Non-parametric method to assess proximity-based prediction.
- **Support Vector Machine (SVM):** Effective in handling high-dimensional spaces.

Preprocessing Strategy:

- Handle missing values, remove duplicates, convert age to years, handle outliers (e.g., height, weight, BMI, blood pressure) using IQR and percentile methods.
- Standardize features (e.g., blood pressure, BMI).

Evaluation Strategy:

- Metrics: Accuracy, Precision, Recall, F1-score, ROC-AUC. These metrics are well-suited for binary classification with imbalanced datasets.
- Cross-validation: 5-fold cross-validation for performance evaluation.

Existing Solutions:

- Research papers and Kaggle projects have explored similar datasets using various classifiers.
- Decision Tree, SVM: sciencedirect.com/science/article/pii/S2772963X24004113
- KNN: pmc.ncbi.nlm.nih.gov/articles/PMC9206502
- KNN, DT: pmc.ncbi.nlm.nih.gov/articles/PMC9855428
- 20 Different Models: kaggle.com/code/vbmokin/20-models-for-cdv-prediction
- Our approach extends these by comparing four distinct techniques under consistent preprocessing steps.

Libraries: Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn, etc.

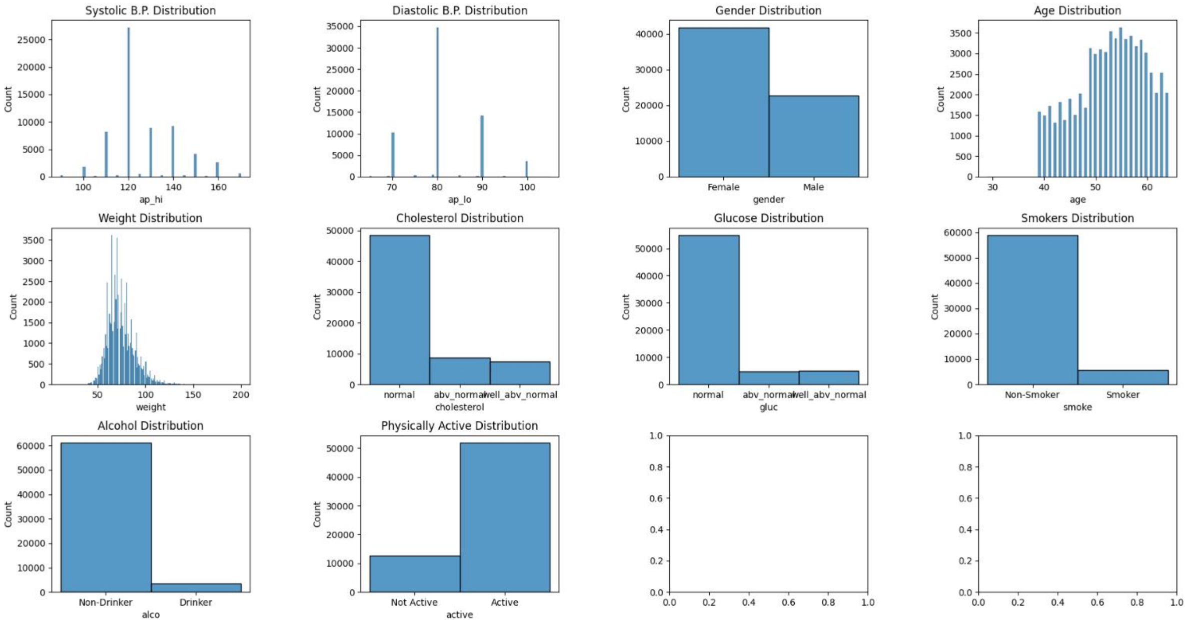
Milestone 2 – Final Project
COMPSCI 4AL3

Team: Refer to the above content in Milestone 1.
Context: Refer to the above content in Milestone 1.
Dataset: Refer to the above content in Milestone 1.

Preprocessing

Techniques Used: IQR, Converted units
Sampling Ratio: 80/20 Split
Remaining Samples: 64502

The main preprocessing performed on the dataset was handling outliers in the blood pressure features. Blood pressure is separated into systolic and diastolic blood pressure. Using IQR the dataset was trimmed of any blood pressure values which were outliers. Additionally, age is represented as day in the dataset and that was changed to years.



One-hot encoding did not need to be applied to features such as cholesterol or glucose since there is an inherent ranking in how it's represented (on a range from 1 to 3).

Model Specifications

Model / Type	Hyperparameters Summary	Regularization	Progress Tracking
Logistic Regression / Supervised	C: 1 max_iter: 100, penalty: 'l2' solver: 'liblinear'	L2 Regularization	Accuracy, Confusion Matrix, Classification Report

Decision Tree / Supervised	criterion: 'gini' max_depth: 5 min_samples_leaf: 20 min_samples_split: 2	Max Depth: 5	Accuracy, Confusion Matrix, Classification Report
KNN / Supervised	CV: 3 fold algorithm: 'auto', leaf_size: 40, metric: minkowski, n_neighbors: 200, weights: 'uniform'	N/A	Accuracy, Confusion Matrix, Classification Report
SVM / Supervised	Kernel = 'rbf', gamma = 0.01, C = 1.0, degree = 3	N/A	Accuracy, Confusion Matrix, Classification Report

All models had hyperparameter tuning performed on them. This process was done using GridSearchCV. For most models, even with tuned hyperparameters, there was a negligible difference in accuracy. SVM was hypertuned and the best model was reported in our submission, due to long training times (1 hour+).

Evaluation

Same as in Milestone 1, changed to 3-Fold Cross Validation for hyperparameter tuning.

Preliminary Results

Metric	**Logistic Regression**	**Decision Tree**	**KNN**	**SVM**
Accuracy	71.59%	72.14%	71.03%	71.79%
Precision (0)	0.79	0.69	0.69	0.70
Precision (1)	0.74	0.78	0.74	0.74
Recall (0)	0.78	0.82	0.77	0.77
Recall (1)	0.65	0.62	0.65	0.67
F1-Score (0)	0.73	0.75	0.73	0.73
F1-Score (1)	0.70	0.69	0.69	0.70
ROC-AUC	0.7155	0.7207	0.7099	0.7176

Hyperparameter Tuning:

Model	**Hyperparameter tuning**
Logistic Regression	{'penalty': ['l2','l1','elasticnet'], 'C': [0.001, 0.01, 0.1, 1], 'solver': ['lbfgs','liblinear','newton-cg','newton-cholesky','sag','saga'], 'max_iter' [100,200,300,400,500, 1000, 2000,5000]}
DT	{'criterion': ['gini', 'entropy', 'log_loss'], 'max_depth': [2,3,5,10,20], 'min_samples_leaf': [1,5,10,20], 'min_samples_split': [2,5,10,20] }
KNN	{'n_neighbors': [1, 10, 25, 50, 100, 150, 200, 250, 300, 400, 500], 'algorithm': ['auto'],' leaf_size': [20, 40], 'weights': ['uniform'], 'metric': ('minkowski', 'chebyshev') }
SVM	{ 'C': [0.1, 1, 10, 100], 'gamma': [1, 0.1, 0.01, 0.001], 'kernel': ['rbf', 'linear']}

Milestone 3 – Final Project
COMPSCI 4AL3

Team: Refer to the above content in Milestone 1.
Context: Refer to the above content in Milestone 1.
Dataset: Refer to the above content in Milestone 1.
Preprocessing: Refer to the above content in Milestone 2.
Preprocessing Improvements:

We ran both feature selection using Chi-Squared scoring for the categorial features and ANOVA f-value scoring for the numerical features. These scoring methods show which features hold importance in predicting the target variable. After removing some of the features that held a lower score, our model actually was less accurate so we decided not to use feature selection. One major improvement to accuracy was using filtering further by creating a BMI feature and filtering out values which were further outliers to our existing IQR method.

Model Specifications: Refer to above content in Milestone 2.
Model Specification Improvements:
On our last milestone we performed hyperparameter tuning already. There were no changes made to the existing model specifications. One thing that was mentioned but not fully explained was the hyperparameter tuning process and its results. For the tuning we used a grid search method, which tries every provided combination of hyperparameters to find the best set. After running this method for each model, we compared the accuracy of the model before and after, and although there was improvement sometimes in accuracy it was tiny (not even a percent).

Evaluation:

We have used sensitivity because it measures the true positive rate, which is critical in minimizing false negatives, especially in cases like predicting diseases where missing a positive case can have severe consequences. The objective value combines accuracy and sensitivity with a weight of 0.25 for sensitivity to balance the trade-off between overall correctness and the importance of identifying positives. This ensures the model prioritizes safety while avoiding overly biased predictions, like classifying everything as positive.

Comparison Table:

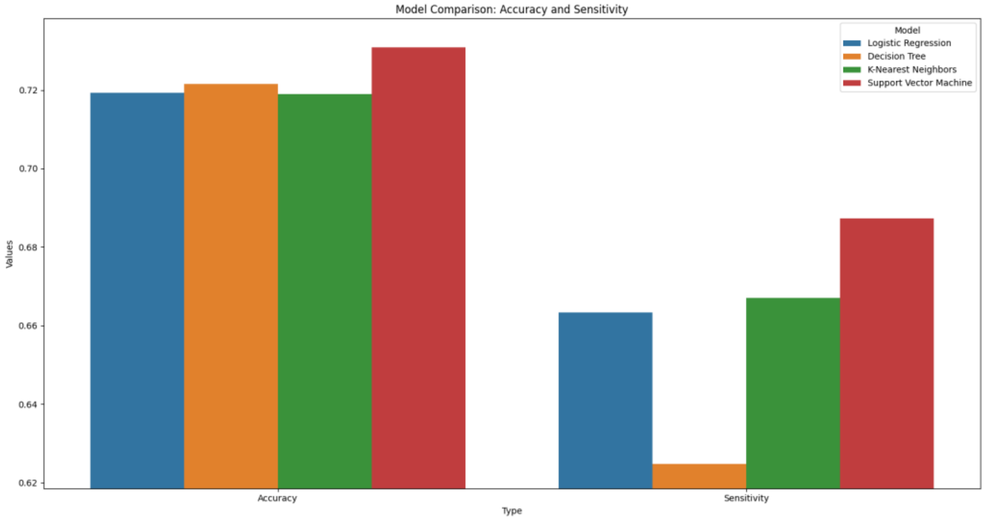
Model	Accuracy	Sensitivity	Objective Value
Logistic Regression	0.719218	0.663399	0.885068
Decision Tree	0.721546	0.624805	0.877747
K-Nearest Neighbors	0.718985	0.666978	0.885730
Support Vector Machine	0.730933	0.687364	0.902774

To refine the models, features with low importance scores and high p-values (height, gender, alco) were removed. However, this negatively impacted the model performance as shown:

	Logistic Regression	DT	KNN	SVM
Accuracy	71.71%	64.56%	68.56%	72.01

Observations:

- The SVM model consistently outperformed others in both accuracy and sensitivity, indicating a balanced trade-off between precision and recall.
- Feature reduction negatively impacted performance.
- Stochasticity in KNN and Decision Tree models resulted in higher variability in performance compared to Logistic Regression and SVM.



<u>Limitations</u>: After performing hyperparameter tuning, most models had little to no changes in their accuracy. Hence, what is left to improve is only feature engineering. One limitation or shortcoming was the lack of accuracy improvement after performing hyperparameter tuning. Utilizing two different methods for scoring the features and removing the lowest scoring features yielded a worst accuracy. Another limitation might be the quality of the dataset. There seems to be many outliers that were filtered out additionally and quite a few strange combinations of height, weight, etc. These were further filtered out through finding BMI and removing rows that didn't fall within the range but brings into question how good of a dataset we are working with.

<u>References:</u> See Footnotes.[1] Evaluation metrics: analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/ , www.nature.com/articles/s41598-024-56706-x

Feature Selection:

https://scikit-learn.org/1.5/modules/feature_selection.html

https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/

Kaggle Dataset:

https://www.kaggle.com/datasets/sulianova/cardiovascular-disease-dataset/1