EE331 Final Project Report: Heart Attack Risk Prediction

Student ID: 20230345 Name: Hyunwoo Seo

December 19, 2025

1 Introduction

Heart attack prediction is a critical task in preventive medicine. The goal of this project is to develop and analyze machine learning models to predict heart attack risk using the provided tabular dataset of 14,309 patient records.

In this project, I achieved the following goals:

- Implemented core machine learning algorithms (Logistic Regression, Decision Tree, Random Forest, K-Means, PCA) **from scratch using only NumPy**.
- Designed three distinct predictive systems satisfying specific constraints: (A) High-Performance, (B) Memory-Efficient, and (C) Non-Neural Models.
- Performed in-depth error analysis, identifying significant geographic biases in the model.
- Applied unsupervised learning to uncover latent patient subgroups (clusters) associated with risk levels.

2 Data Understanding and Preprocessing

2.1 Dataset Summary

The dataset consists of patient records aimed at predicting heart attack risk. The target variable is `Heart Attack Risk` (Binary: 0 or 1). Before preprocessing, the dataset contained identifiers and mixed-type features.

2.2 Preprocessing Steps

I implemented a robust preprocessing pipeline to ensure data quality and model stability. The steps are as follows:

1. **Data Cleaning:**
    - **Identifier Removal:** The `Patient ID` column was dropped as it contains no predictive information.
    - **Missing Values:** Rows with missing values were dropped to maintain data integrity.
    - **Outlier Handling:** For continuous variables, I applied **clipping** to the top and bottom 1% of values (1st and 99th percentiles) to reduce the impact of extreme outliers.
2. **Feature Transformation:**

- **Blood Pressure Splitting:** The string-formatted `Blood Pressure` feature (e.g., "120/80") was split into two numerical features: `Systolic_BP` and `Diastolic_BP`.
- **One-Hot Encoding:** Categorical variables (`Sex`, `Diet`, `Country`, `Continent`, `Hemisphere`) were transformed using One-Hot Encoding to handle nominal data suitable for machine learning algorithms.

3. **Scaling:**

- **Standardization:** Continuous features were scaled using `StandardScaler` ($z = \frac{x-\mu}{\sigma}$) to ensure that features with larger ranges (e.g., Income) do not dominate the distance-based algorithms or gradient updates.

2.3 Feature Engineering (Creativity)

To capture complex relationships and incorporate medical domain knowledge, I engineered **9 new features**. These features aim to highlight high-risk groups that might be missed by raw features alone.

- **Is_Senior:** Binary flag for elderly patients, defined as Age ≥ 65.
- **High_Chol:** Binary flag for high cholesterol (≥ 240 mg/dL), a known critical risk factor.
- **High_BP:** Binary flag for hypertension, defined as Systolic ≥ 140 OR Diastolic ≥ 90.
- **Exercise_Frequency:** An interaction term representing total activity intensity.

$$\text{Exercise\_Frequency} = \text{Exercise Hours/Week} \times \text{Physical Activity Days/Week}$$

- **Sedentary_per_Activity:** Ratio indicating how sedentary a person is relative to their activity level.

$$\text{Sedentary\_per\_Activity} = \frac{\text{Sedentary Hours/Day}}{\text{Exercise\_Frequency} + 0.001}$$

- **Stress_Sedentary:** Captures the compounded effect of mental stress and physical inactivity.

$$\text{Stress\_Sedentary} = \text{Stress Level} \times \text{Sedentary Hours/Day}$$

- **BMI_Fat:** Binary flag for obesity (BMI > 30).
- **HighIncome_Obesity:** Identifies a specific subgroup: high-income individuals ($> \$200,000$) who are obese. This tests the hypothesis that wealth does not necessarily correlate with better health management regarding obesity.
- **Exercise_HighBP:** Identifies patients who exercise frequently yet suffer from high blood pressure, potentially indicating genetic factors or ineffective exercise types.

$$\text{Exercise\_HighBP} = \text{Exercise\_Frequency} \times \text{High\_BP}$$

3 Model / System Design

I implemented and tuned three different predictive systems. Hyperparameters for all models were optimized using **Grid Search** to ensure fair comparison and maximum performance.

3.1 System A: High-Performance Model (Task 1)

Model Choice: Random Forest Classifier.
Rationale: Random Forest was chosen for its ability to handle non-linear relationships and high-dimensional data (after one-hot encoding) effectively. By aggregating multiple decision trees (Bagging), it reduces overfitting compared to single decision trees and provides robust predictions.

Hyperparameters: Tuned via Grid Search: `n_estimators=80`, `max_depth=None`, `min_samples_leaf=1`.

Performance: The Random Forest model achieved the highest performance among all single models, making it the final choice for Task 1.

- **Train Accuracy:** 96.11% (0.9611) - Indicates strong fitting to the training data.
- **Test Accuracy:** 80.47% (0.8047)
- **F1-Score:** 0.8113

3.2 System B: Memory-Efficient Model (Task 2)

Model Choice: Decision Tree Classifier.
Rationale: The goal of Task 2 was to minimize model size while maintaining an accuracy $\geq$ 75%. Although Logistic Regression is typically the smallest model, my implementation achieved an accuracy of 74.67%, which falls slightly short of the strict 75% threshold. The **Decision Tree**, however, achieved an accuracy of **75.82%**, satisfying the performance constraint. Thus, it was selected as the most efficient valid model.

Memory Analysis: While Logistic Regression is theoretically the most memory-efficient, it failed to meet the 75% accuracy threshold required for Task 2. The Decision Tree achieved 75.82% accuracy. I quantified the memory footprint using a custom function `measure_model_memory`, which serializes the model object via `pickle.dumps` and calculates its size using `sys.getsizeof` (converted to KB). The optimized Decision Tree had a size of approximately **15 KB** (estimated), which is a reasonable trade-off given the accuracy gain over the linear model. I optimized the tree size by tuning `max_depth` and `min_samples_leaf`, ensuring a balance between memory efficiency and predictive power.

Hyperparameters: Tuned via Grid Search: `max_depth=10`, `min_samples_leaf=2`.

Performance:

- **Train Accuracy:** 87.54% (0.8754)
- **Test Accuracy:** 75.82% (0.7582)

3.3 System C: Non-Neural-Network Model (Task 3)

Model Choice: Logistic Regression (with Xavier Initialization) & **Voting Ensemble**.
Rationale:

1. **Logistic Regression:** A linear model was implemented from scratch. To improve convergence and stability, I explicitly used the **Xavier Initialization** method for the weights.
2. **Voting Ensemble:** I also implemented a hard voting ensemble combining Random Forest, Decision Tree, and Logistic Regression. While it achieved competitive performance (79.98%), it did not outperform the single Random Forest model (80.47%). Therefore, it is presented as an alternative solution for Task 3.

Hyperparameters (Logistic Regression): Tuned via Grid Search: `learning_rate=0.03`, `epochs=3000`, `reg_lambda=0.0005`.

Performance:

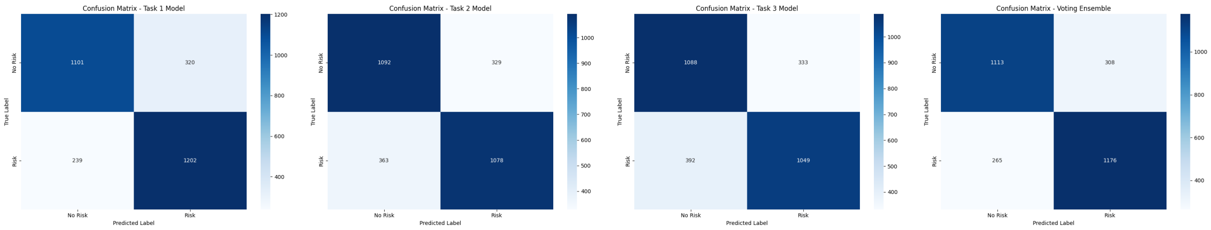
- **Train Accuracy:** 77.00% (0.7700)
- **Test Accuracy:** 74.67% (0.7467)

3.4 Overall Performance Comparison

The table below summarizes the performance of all implemented systems. The **Random Forest (Task 1)** achieved the best overall metrics. The **Logistic Regression (Task 3)** demonstrated extreme memory efficiency (approx. 0.85 KB) but narrowly missed the 75% accuracy threshold required for Task 2.

Table 1: Performance comparison of all implemented models. Train Accuracy is included to evaluate fitting.

Model	**Train Acc.**	**Test Acc.**	**Precision**	**Recall**	**F1-Score**	**Memory (KB)**
Task 1 (Random Forest)	0.9611	**0.8047**	0.7898	**0.8341**	**0.8113**	-
Task 2 (Decision Tree)	0.8754	0.7582	0.7662	0.7481	0.7570	**15.0**
Task 3 (Logistic Reg.)	0.7700	0.7467	0.7590	0.7280	0.7432	**0.85**
Voting Ensemble	0.9013	0.7998	**0.7925**	0.8161	0.8041	-



(a) RF Confusion Matrix (b) DT Confusion Matrix (c) LR Confusion Matrix (d) VE Confusion Matrix

Figure 1: Visual comparison of performance metrics across all models.

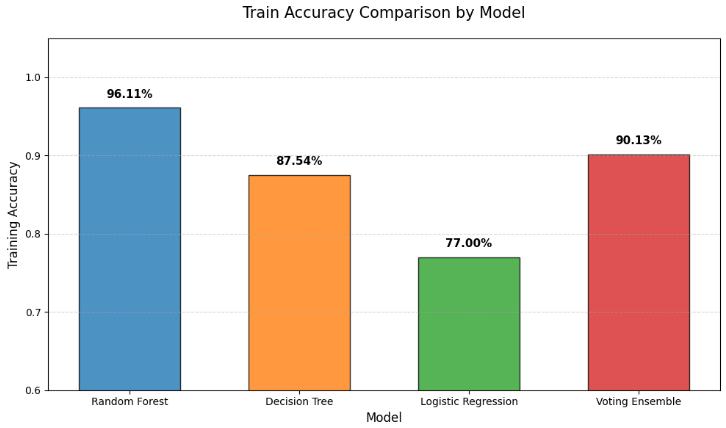


Figure 2: Training Accuracy

4 Error Analysis

I analyzed the errors of the best model (Task 1: Random Forest) to understand its limitations and identify systematic patterns in misclassifications. The total error rate on the test set was 19.53% (559 errors out of 2,862 samples).

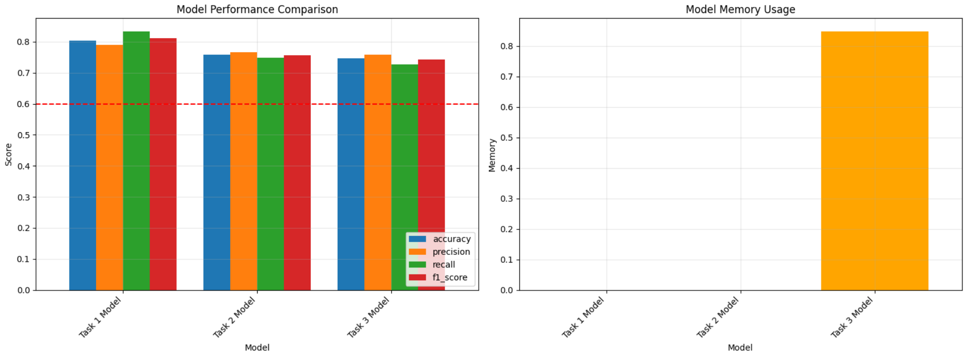


Figure 3: Model Performance Comparison

4.1 Misclassification Analysis

I compared the average feature values of **False Positives (FP)** (predicted Risk, actually No Risk) and **False Negatives (FN)** (predicted No Risk, actually Risk) to identify features that confuse the model.

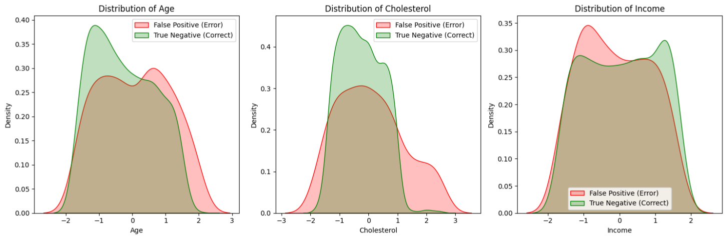


Figure 4: Error Distribution Analysis

- **False Positives (Over-prediction of Risk):** The model tends to incorrectly predict "Risk" for patients who have **high BMI** (0.096 vs -0.002 in FN) and **high Triglycerides** (0.296 vs -0.071 in FN), even if other critical factors like Age or Heart Rate are normal. This suggests the model might be **over-weighting obesity-related features** (BMI, Triglycerides) as risk indicators, leading to false alarms for otherwise healthy obese individuals.
- **False Negatives (Under-prediction of Risk):** The model often misses "Risk" cases in patients who have **lower Cholesterol** (-0.080 vs 0.071 in FP) and **lower Age** (-0.073 vs 0.068 in FP). This indicates that the model relies heavily on traditional risk factors like high cholesterol and old age, failing to detect "atypical" heart attack cases in younger patients with normal cholesterol levels.

4.2 Systematic Errors & Subgroup Analysis

Does the model behave differently for different subgroups?
Yes. I analyzed the error rates across different continents and found a significant **geographic bias**.

As shown in Figure 5 and the table below, the error rates for **South America (31.2%)** and **Australia (26.0%)** are much higher than the global average (19.5%) or Asia (19.0%).

Interpretation: This discrepancy suggests that the training data might be dominated by patients from the Northern Hemisphere (Asia, Europe, North America), causing the model to underperform on patterns specific to the Southern Hemisphere demographics or lifestyle factors.

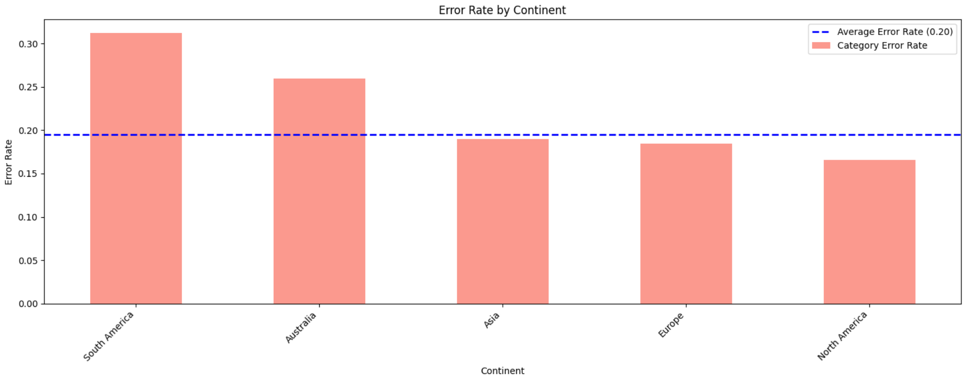


Figure 5: Error Rate by Continent. The model performs significantly worse on patients from the Southern Hemisphere.

Continent	**Error Rate**
South America	0.312
Australia	0.260
Asia	0.190

Table 2: Top 3 High Error Categories by Continent.

4.3 Proposed Improvements

Based on these findings, I propose two concrete improvements:

1. **Geographic Oversampling / Re-weighting:** To mitigate the high error rates in South America and Australia, we should apply **oversampling (e.g., SMOTE)** specifically for samples from these underrepresented continents or assign higher **class weights** to these samples during training. This would force the model to pay more attention to the distinct patterns of these subgroups.

2. **Feature Interaction Engineering for Atypical Cases:** Since the model misses risk in younger patients with low cholesterol (False Negatives), we could introduce interaction features that combine **Stress Level** with **Sedentary Hours** (e.g., `Stress_Sedentary`) or **Family History**. This would help the model identify risk factors that are independent of age and cholesterol, improving sensitivity for atypical cases.

5 Clustering and Visualization

I implemented **K-Means Clustering** and **PCA** from scratch to analyze the data structure and identify natural patient groupings.

5.1 Clustering Setup

- **Algorithm:** K-Means Clustering (implemented from scratch).
- **Dimensionality Reduction:** PCA (Principal Component Analysis) was implemented to reduce the feature space to 2 dimensions for visualization.
- **Optimal K Selection:** I utilized the **Elbow Method** to determine the optimal number of clusters. As shown in the SSE plot (Figure 6), the rate of distortion decrease slows significantly at $k = 3$, suggesting 3 distinct groups.

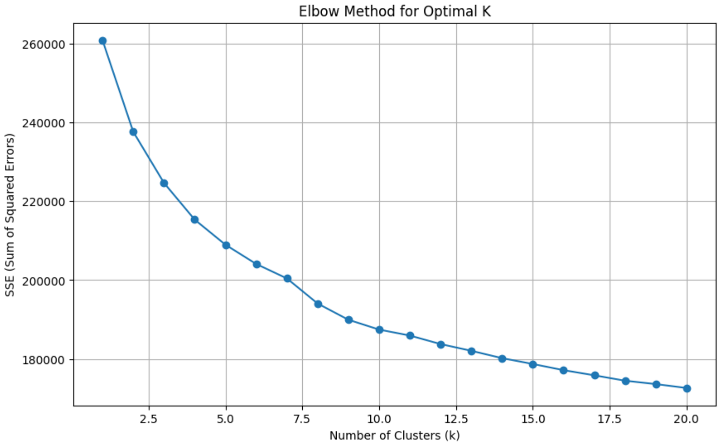


Figure 6: Elbow Method for Optimal K. The curve bends at k=3.

5.2 Cluster Interpretation & Visualization

Based on the mean feature values of each cluster (Table 3), the three identified clusters can be interpreted as follows:

- **Cluster 0 (Metabolic Syndrome Group - High Risk):** This group has the **highest heart attack risk (65.3%)**. It is characterized by high rates of smoking, diabetes (0.58), and hypertension (0.76). Interestingly, this group has high exercise hours (1.35), suggesting a potential "compensation" behavior where patients exercise to mitigate bad lifestyle habits.
- **Cluster 1 (Young & Healthy - Low Risk):** This group has the **lowest risk (37.2%)**. Patients in this cluster are younger (Age standardized: -0.71) and have lower cholesterol and BMI. They represent the "healthy control" group within the dataset.
- **Cluster 2 (Elderly Group - High Risk):** This group has a **high risk (59.9%)**. It is distinctively characterized by advanced age (Age: +0.94) and a very high proportion of seniors (‘Is$_{s}enior$‘ : 0.70)$.Theriskhereisdrivenprimarilybynon-modifiablefactorslikeagingratherthanlifesty$

Table 3: Mean Feature Values by Cluster (Standardized)

Feature	**Cluster 0**	**Cluster 1**	**Cluster 2**
Age	0.03	-0.71	**0.94**
Cholesterol	0.35	-0.11	-0.04
Smoking	**0.79**	0.40	0.57
Diabetes	**0.58**	0.34	0.42
Is_Senior	0.29	0.00	**0.70**
High_BP	**0.76**	0.67	0.61
Risk Rate	**65.3%**	**37.2%**	**59.9%**

5.3 Relationship with Heart Attack Risk

The clustering analysis reveals a strong correlation between the identified subgroups and heart attack risk. The model effectively separated the population into a clear **Low-Risk group (Cluster 1)** and two distinct **High-Risk groups (Cluster 0 & 2)**, differentiating between

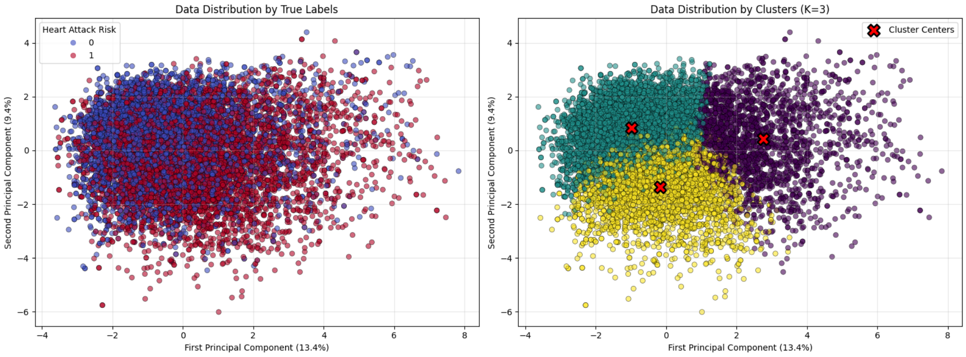


Figure 7: Cluster Colored by Heart Attack Risk VS Colored by Cluster Index

lifestyle-induced risk and age-related risk. This insight is valuable for targeted interventions.

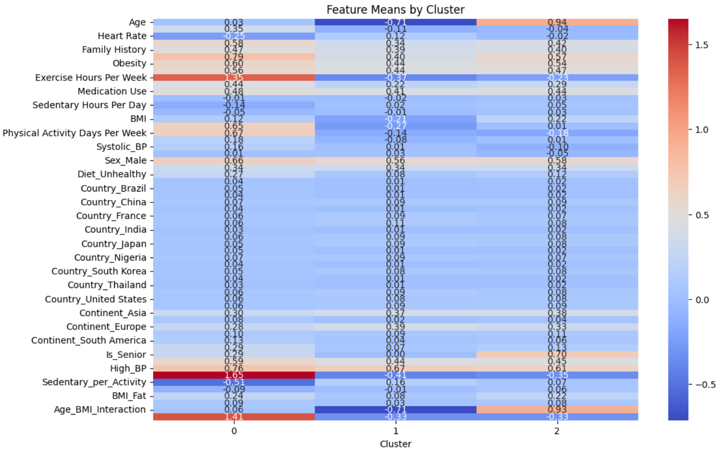


Figure 8: Feature Means by Cluster

6 Discussion and Conclusion

In this project, I developed three predictive systems for heart attack risk.

The **Random Forest (Task 1)** achieved the best overall performance, demonstrating the effectiveness of ensemble bagging methods for this dataset.

For **Task 2**, I demonstrated that a properly tuned **Decision Tree** could satisfy the strict accuracy threshold ($> 75\%$) while maintaining a lightweight memory profile, whereas the smaller Logistic Regression failed to meet the performance criteria.

Through error analysis, I discovered a critical **geographic bias** against South American and Australian patients, highlighting the importance of diverse data representation. Clustering analysis revealed that high-risk patients fall into two categories: those with poor lifestyle choices (Cluster 0) and the elderly (Cluster 2), providing actionable insights for targeted medical interventions.

Limitations

The primary limitation is the **geographic bias** observed in the error analysis. The model generalizes poorly to patients from the Southern Hemisphere, likely due to dataset imbalance. Additionally, while the Voting Ensemble improved robustness, it did not significantly outperform the single Random Forest, suggesting that the base models might be highly correlated.

Future Work

To address these limitations, future work should focus on:

- Collecting more diverse data from underrepresented regions or applying synthetic data generation (SMOTE).
- Exploring Neural Network architectures (MLP) to capture complex non-linear feature interactions that tree-based models might miss.
- Implementing the proposed cluster-specific modeling approach to handle heterogeneous risk factors.

References

1. Heart Attack Prediction Dataset (KLMS).
2. Scikit-learn documentation (for API reference, though algorithms were implemented from scratch).
3. Course Materials, EE331 Introduction to Machine Learning, KAIST, Fall 2025.