Group 2 Final Documentation: Heart Disease Risk Analysis Using Lifestyle and Behavioral Data

Team Members: Mary Benjamin, Almin Karowadia, Sarita Sapkota Subedi

Submitted to: Soma Datta

Department of Software Engineering, University of Houston – Clear Lake

Date: May 07, 2025

1. Introduction

Heart disease remains one of the most critical health challenges worldwide, particularly in the form of heart attacks. Our objective was to explore and analyze behavioral, demographic, and medical data to detect patterns that could be associated with increased risk of heart attacks. By leveraging national survey data and applying advanced data science techniques, we aimed to identify and validate lifestyle-related risk factors that could guide public health strategies and early intervention programs.

This study follows a structured data science process, starting from data cleaning and preparation, followed by exploratory data analysis (EDA), dimensionality reduction using Principal Component Analysis (PCA), unsupervised learning through clustering, supervised learning via decision trees, and rule mining using association algorithms. Each step contributed either to discovering new insights or verifying the predictive accuracy and reliability of the identified patterns.

2. Methodology and Workflow

2.1 Data Cleaning and Preparation

We used the dataset heart_2022_no_nans.csv, which included over 240,000 observations without any missing values. In R, categorical variables were converted into factors to ensure appropriate statistical treatment. Non-numeric columns were then encoded as numeric for PCA compatibility. Scaling and normalization were applied to standardize the dataset, enabling fair comparison among features. Some unique data, and redundant and correlated columns are removed. This cleaned and transformed dataset was saved as heart_cleaned.csv for use in Weka and JMP for further analysis.

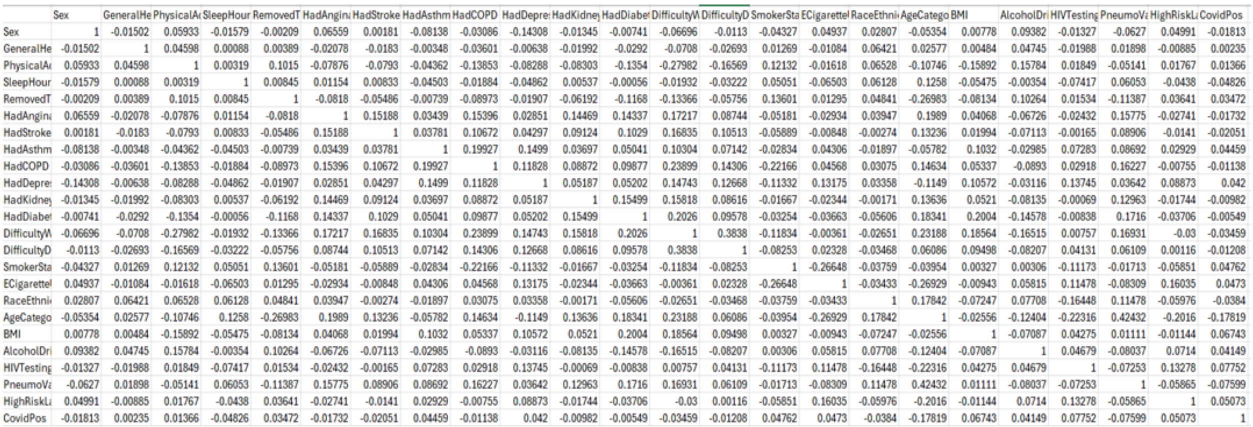
2.2 Exploratory Data Analysis (EDA)

EDA was performed using R and JMP to understand the distribution of each feature and identify potential patterns. **Univariate** and **bivariate analyses** were used to examine both individual attributes and their relationships with the target variable (`HadHeartAttack`). These early explorations showed that no single variable had a strong predictive influence on its own.

To further explore inter-feature relationships, we created a **correlation matrix** and a **heatmap** in R. As shown in the figures below, most variables displayed **weak pairwise correlations** (correlation coefficients near 0), suggesting little to no strong linear association between any two features. This lack of multicollinearity supports the idea that important predictive patterns may lie in **complex multivariate interactions**.

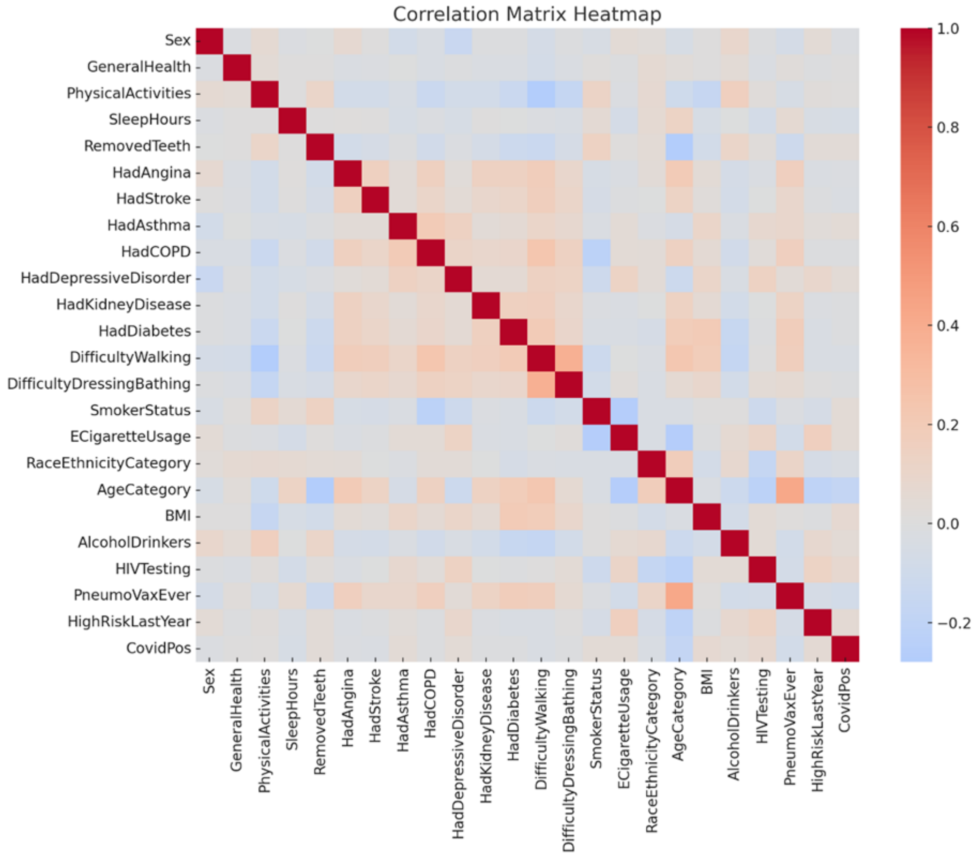
These findings justified the use of **Principal Component Analysis (PCA)**, which can reveal hidden structure in the data by combining multiple weakly correlated variables into a set of more informative components.

Figure 1: Correlation Matrix of Numerically Encoded Heart Health Dataset



	Sex	GeneralHe	PhysicalAc	SleepHour	RemovedT	HadAngina	HadStroke	HadAsthm	HadCOPD	HadDepre	HadKidney	HadDiabet	DifficultyW	DifficultyD	SmokerSta	ECigarette	RaceEthni	AgeCatego	BMI	AlcoholDri	HIVTesting	PneumoVa	HighRiskLa	CovidPos
Sex	1	-0.01502	0.05933	-0.01579	-0.00209	0.06559	0.00181	-0.08138	-0.03086	-0.14308	-0.01345	-0.00741	-0.06696	-0.0113	-0.04327	0.04937	0.02807	-0.05354	0.00778	0.09382	-0.01327	-0.0627	0.04991	-0.01813
GeneralHe	-0.01502	1	0.04598	0.00088	0.00389	-0.02078	-0.0183	-0.00348	-0.03601	-0.00638	-0.01992	-0.0292	-0.0708	-0.02693	0.01269	-0.01084	0.06421	0.02577	0.00484	0.04745	-0.01988	0.01898	-0.00885	0.00235
PhysicalAc	0.05933	0.04598	1	0.00319	0.1015	-0.07876	-0.0793	-0.04362	-0.13853	-0.08288	-0.08303	-0.1354	-0.27982	-0.16569	0.12132	-0.01618	0.06528	-0.10746	-0.15892	0.15784	0.01849	-0.05141	0.01767	0.01366
SleepHour	-0.01579	0.00088	0.00319	1	0.00845	0.01154	0.00833	-0.04503	-0.01884	-0.04862	0.00537	-0.00056	-0.01932	-0.03222	0.05051	-0.06503	0.06128	0.1258	-0.05475	-0.00354	-0.07417	0.06053	-0.0438	-0.04826
RemovedT	-0.00209	0.00389	0.1015	0.00845	1	-0.0818	-0.05486	-0.00739	-0.08973	-0.01907	-0.06192	-0.1168	-0.13366	-0.05756	0.13601	0.01295	0.04841	-0.26983	-0.08134	0.10264	0.01534	-0.11387	0.03641	0.03472
HadAngina	0.06559	-0.02078	-0.07876	0.01154	-0.0818	1	0.15188	0.03439	0.15396	0.02851	0.14469	0.14337	0.17217	0.08744	-0.05181	-0.02934	0.03947	0.1989	0.04068	-0.06726	-0.02432	0.15775	-0.02741	-0.01732
HadStroke	0.00181	-0.0183	-0.0793	0.00833	-0.05486	0.15188	1	0.03781	0.10672	0.04297	0.09124	0.1029	0.16835	0.10513	-0.05889	-0.00848	-0.00274	0.13236	0.01994	-0.07113	-0.00165	0.08906	-0.0141	-0.02051
HadAsthm	-0.08138	-0.00348	-0.04362	-0.04503	-0.00739	0.03439	0.03781	1	0.19927	0.1499	0.03697	0.05041	0.10304	0.07142	-0.02834	0.04306	-0.01897	-0.05782	0.1032	-0.02985	0.07283	0.08692	0.02929	0.04459
HadCOPD	-0.03086	-0.03601	-0.13853	-0.01884	-0.08973	0.15396	0.10672	0.19927	1	0.11828	0.08872	0.09877	0.23899	0.14306	-0.22166	0.04568	0.03075	0.14634	0.05337	-0.0893	0.02918	0.16227	-0.00755	-0.01138
HadDepre	-0.14308	-0.00638	-0.08288	-0.04862	-0.01907	0.02851	0.04297	0.1499	0.11828	1	0.05187	0.05202	0.14743	0.12668	-0.11332	0.13175	0.03358	-0.1149	0.10572	-0.03116	0.13745	0.03642	0.08873	0.042
HadKidney	-0.01345	-0.01992	-0.08303	0.00537	-0.06192	0.14469	0.09124	0.03697	0.08872	0.05187	1	0.15499	0.15818	0.08616	-0.01667	-0.02344	-0.00171	0.13636	0.0521	-0.08135	-0.00069	0.12963	-0.01744	-0.00982
HadDiabet	-0.00741	-0.0292	-0.1354	-0.00056	-0.1168	0.14337	0.1029	0.05041	0.09877	0.05202	0.15499	1	0.2026	0.09578	-0.03254	-0.03663	-0.05606	0.18341	0.2004	-0.14578	-0.00838	0.1716	-0.03706	-0.00549
DifficultyW	-0.06696	-0.0708	-0.27982	-0.01932	-0.13366	0.17217	0.16835	0.10304	0.23899	0.14743	0.15818	0.2026	1	0.3838	-0.11834	-0.00361	-0.02651	0.23188	0.18564	-0.16515	0.00757	0.16931	-0.03	-0.03459
DifficultyD	-0.0113	-0.02693	-0.16569	-0.03222	-0.05756	0.08744	0.10513	0.07142	0.14306	0.12668	0.08616	0.09578	0.3838	1	-0.08253	0.02328	-0.03468	0.06086	0.09498	-0.08207	0.04131	0.06109	0.00116	-0.01208
SmokerSta	-0.04327	0.01269	0.12132	0.05051	0.13601	-0.05181	-0.05889	-0.02834	-0.22166	-0.11332	-0.01667	-0.03254	-0.11834	-0.08253	1	-0.26648	-0.03759	-0.03954	0.00327	0.00306	-0.11173	-0.01713	-0.05851	0.04762
ECigarette	0.04937	-0.01084	-0.01618	-0.06503	0.01295	-0.02934	-0.00848	0.04306	0.04568	0.13175	-0.02344	-0.03663	-0.00361	0.02328	-0.26648	1	-0.03433	-0.26929	-0.00943	0.05815	0.11478	-0.08309	0.16035	0.0473
RaceEthni	0.02807	0.06421	0.06528	0.06128	0.04841	0.03947	-0.00274	-0.01897	0.03075	0.03358	-0.00171	-0.05606	-0.02651	-0.03468	-0.03759	-0.03433	1	0.17842	-0.07247	0.07708	-0.16448	0.11478	-0.05976	-0.0384
AgeCatego	-0.05354	0.02577	-0.10746	0.1258	-0.26983	0.1989	0.13236	-0.05782	0.14634	-0.1149	0.13636	0.18341	0.23188	0.06086	-0.03954	-0.26929	0.17842	1	-0.02556	-0.12404	-0.22316	0.42432	-0.2016	-0.17819
BMI	0.00778	0.00484	-0.15892	-0.05475	-0.08134	0.04068	0.01994	0.1032	0.05337	0.10572	0.0521	0.2004	0.18564	0.09498	0.00327	-0.00943	-0.07247	-0.02556	1	-0.07087	0.04275	0.01111	-0.01144	0.06743
AlcoholDri	0.09382	0.04745	0.15784	-0.00354	0.10264	-0.06726	-0.07113	-0.02985	-0.0893	-0.03116	-0.08135	-0.14578	-0.16515	-0.08207	0.00306	0.05815	0.07708	-0.12404	-0.07087	1	0.04679	-0.08037	0.0714	0.04149
HIVTesting	-0.01327	-0.01988	0.01849	-0.07417	0.01534	-0.02432	-0.00165	0.07283	0.02918	0.13745	-0.00069	-0.00838	0.00757	0.04131	-0.11173	0.11478	-0.16448	-0.22316	0.04275	0.04679	1	-0.07253	0.13278	0.07752
PneumoVa	-0.0627	0.01898	-0.05141	0.06053	-0.11387	0.15775	0.08906	0.08692	0.16227	0.03642	0.12963	0.1716	0.16931	0.06109	-0.01713	-0.08309	0.11478	0.42432	0.01111	-0.08037	-0.07253	1	-0.05865	-0.07599
HighRiskLa	0.04991	-0.00885	0.01767	-0.0438	0.03641	-0.02741	-0.0141	0.02929	-0.00755	0.08873	-0.01744	-0.03706	-0.03	0.00116	-0.05851	0.16035	-0.05976	-0.2016	-0.01144	0.0714	0.13278	-0.05865	1	0.05073
CovidPos	-0.01813	0.00235	0.01366	-0.04826	0.03472	-0.01732	-0.02051	0.04459	-0.01138	0.042	-0.00982	-0.00549	-0.03459	-0.01208	0.04762	0.0473	-0.0384	-0.17819	0.06743	0.04149	0.07752	-0.07599	0.05073	1

Figure 2: Heatmap Visualization of Feature Correlations

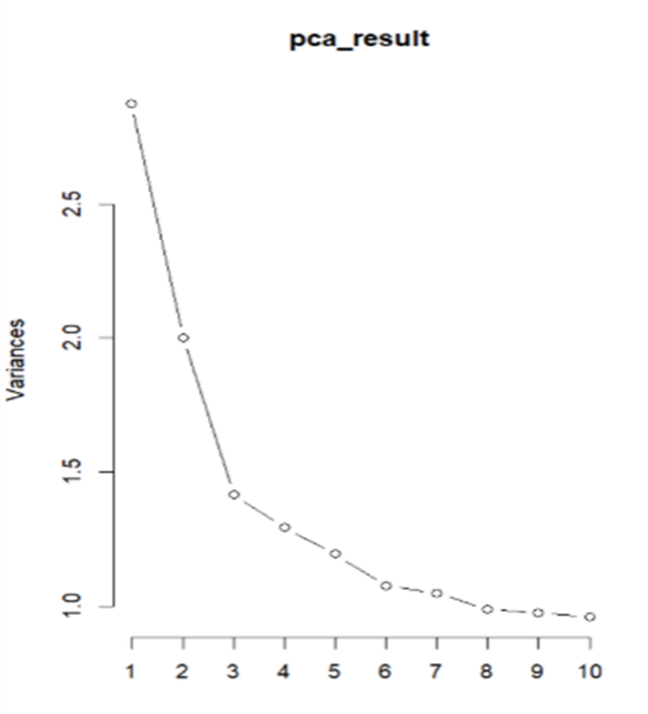


The correlation matrix as shown in figure above, revealed weak pairwise correlations between attributes, indicating the presence of more complex multivariate relationships. These observations justified the need for dimensionality reduction and clustering to uncover latent structure in the data.

2.3 Principal Component Analysis (PCA)

PCA was performed using the prcomp() function in R. This reduced the dimensionality of the dataset while retaining key variance-contributing components. The scree plot shown in Figure below guided the selection of the top six principal components (PCs) using elbow method, which explained a significant proportion of the total variance.

Figure 3: Scree Plot of Principal Component Variance



The PCA loadings are used to relate those PCs back to original attributes. PCA loading has the exact weight (portion) each attribute has on principle components. The PCA loading document is attached in the appendix section. These components were interpreted using PCA loadings as follows:

- **PC1**: Chronic health issues (e.g., diabetes, difficulty walking, angina)
- **PC2**: Age and lifestyle indicators
- **PC3**: Smoking behavior, BMI, and heart attack history
- **PC4**: Mental health and demographic influences
- **PC5**: Respiratory and dental health

- **PC6**: General health perception and metabolic conditions

The transformed dataset containing the top six PCs along with the target variable (HadHeartAttack) was saved as heart_pca6_with_target.csv. We have used both the clean data with original attributes and new data after PCA to find out if we get any better results with uncorrelated PCA reduced data.

2.4 Clustering

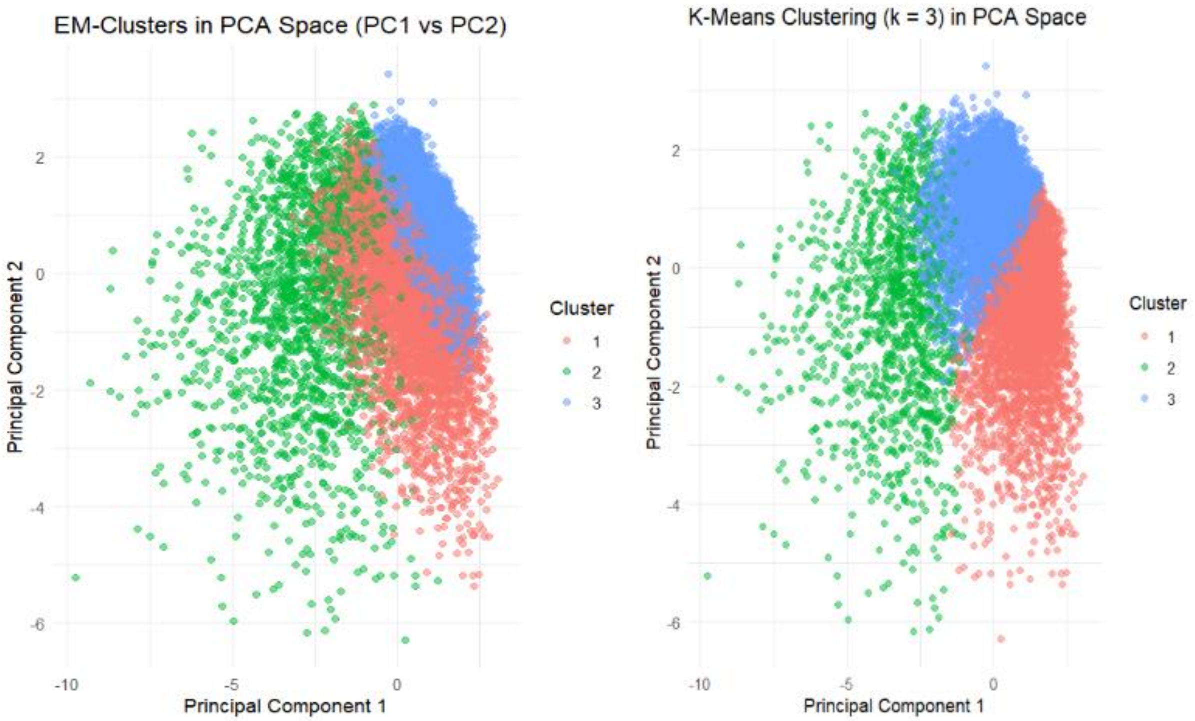
Clustering was performed to uncover hidden patterns and group individuals based on shared health, behavioral, and demographic characteristics that might not be apparent through individual variables alone. This unsupervised learning approach allowed us to segment the population into meaningful risk groups for targeted analysis.

We applied clustering to both the **PCA-reduced dataset (6 components)** and the **original full dataset** to compare how dimensionality reduction impacts the clarity and interpretability of groupings.

i. Clustering for PCA reduced data using JMP

Using JMP, k-means and EM clustering were performed on the PCA-reduced data. The EM clustering was done only to get the optimal number of clusters. The figure below shows the scatter plot of clusters in PCA space, which clearly shows 3 clusters were assigned in this case, and that number is used for K-means clustering. We have not analyzed the clustering assignment and the possible risk factors associated with those EM generated clusters. It was done only to get the optimal number of clusters. Then K-means clustering is performed, and each cluster is profiled based on the mean scores of the six principal components:

Figure 4: Scatter plot of EM and K-means Clustering in PCA space. EM Clustering is used to find the optimal number of clusters.

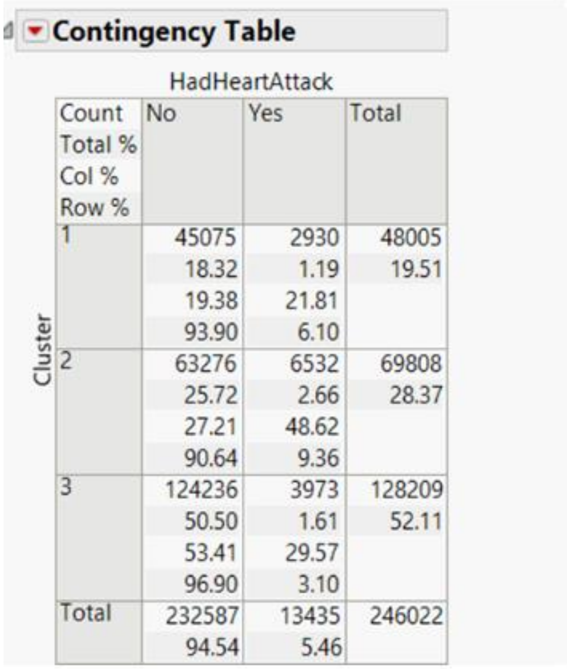


- **Cluster 1**: Moderate risk; characterized by lower PC2 and PC3 values, suggesting younger, less smoking-prone individuals with moderate chronic health conditions.

- **Cluster 2**: High risk; marked by low PC1 and high PC3/PC6 values, indicating chronic illness, smoking behavior, and poor general health perception.

- **Cluster 3**: Low risk; higher PC1 and PC2 scores were associated with fewer health risks and better overall health status.

Figure 5: Cluster distribution of PCA-reduced data.



Contingency Table

Cluster	Count Total % Col % Row %	HadHeartAttack No	 Yes	 Total
1		45075 18.32 19.38 93.90	2930 1.19 21.81 6.10	48005 19.51
2		63276 25.72 27.21 90.64	6532 2.66 48.62 9.36	69808 28.37
3		124236 50.50 53.41 96.90	3973 1.61 29.57 3.10	128209 52.11
Total		232587 94.54	13435 5.46	246022

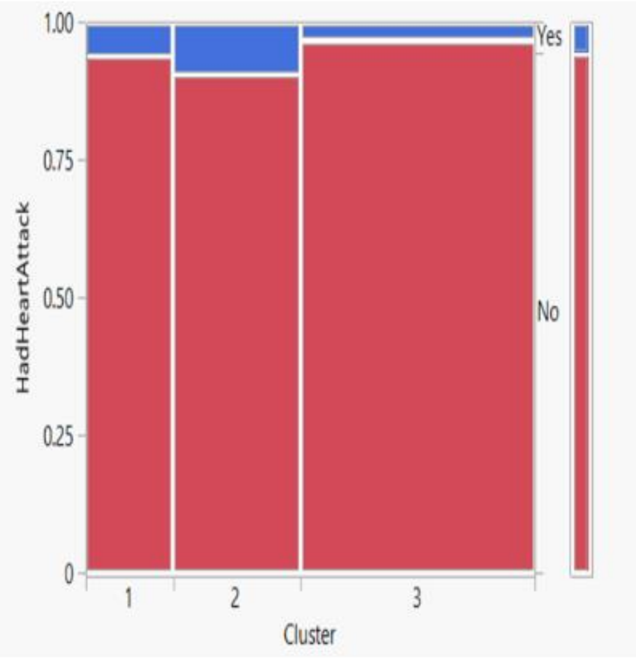


Table 1: Interpretation summary explaining cluster distribution and associated risk

Cluster	Description	Heart Attack Rate	Notes
Cluster 1	48,005 people	**6.10%** (2930/48005)	❗ moderate risk
Cluster 2	69,808 people	**9.36%** (6532/69808)	‼️ high risk— nearly 1 in 10 (high risk group)
Cluster 3	128,209 people	**3.10%** (3973/128209)	✅ Low heart attack risk

Visualization of cluster assignments and heart attack rates confirmed the interpretability of these clusters, with **Cluster 2 showing the highest heart attack rate (~9.36%)**.

ii. Clustering for PCA reduced data using R

Using R, the elbow method was used to find the optimal number of clusters. The number of clusters was decided to be k = 3, and this was also confirmed using EM Means, which used 3 clusters when running using R and JMP.

a. Methods

The clustering algorithm used is k-means, and the number of clusters is k = 3. For processing the k-means tools, such as kmean(), ggplot2, and dplyr in R have been used. As part of preprocessing, the numeric features have been normalized, categorical features have been converted to factors, near-zero variance and highly correlated variables have been removed.

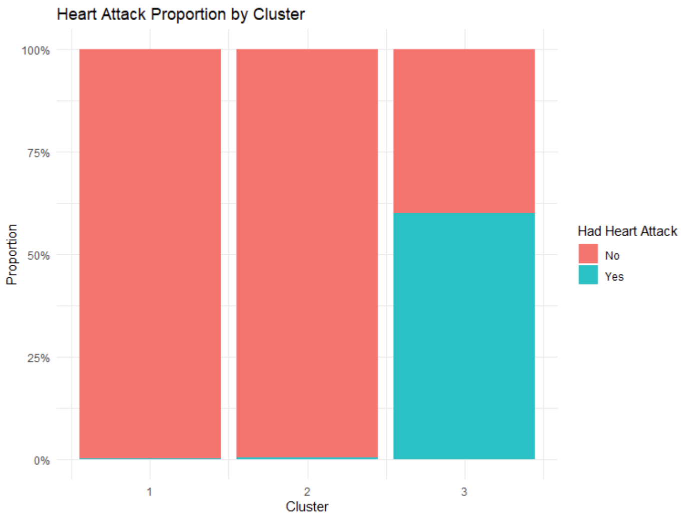
b. Overview

Table 2: Interpretation summary explaining cluster distribution using k-means (k = 3) in R

Cluster	% No Heart Attack	% Yes Heart Attack
1	~99.94%	~0.06%
2	~99.54%	~0.46%
3	~40.00%	~60.00%

- Cluster 3 has a much higher heart attack rate (60%), indicating this group is at significantly higher risk.
- Clusters 1 and 2 have a very low incidence of heart attacks and likely represent low-risk populations.

Figure 6: Interpretation summary showing the distribution of heart attack cases across PCA-based clusters



c. Visual Insights

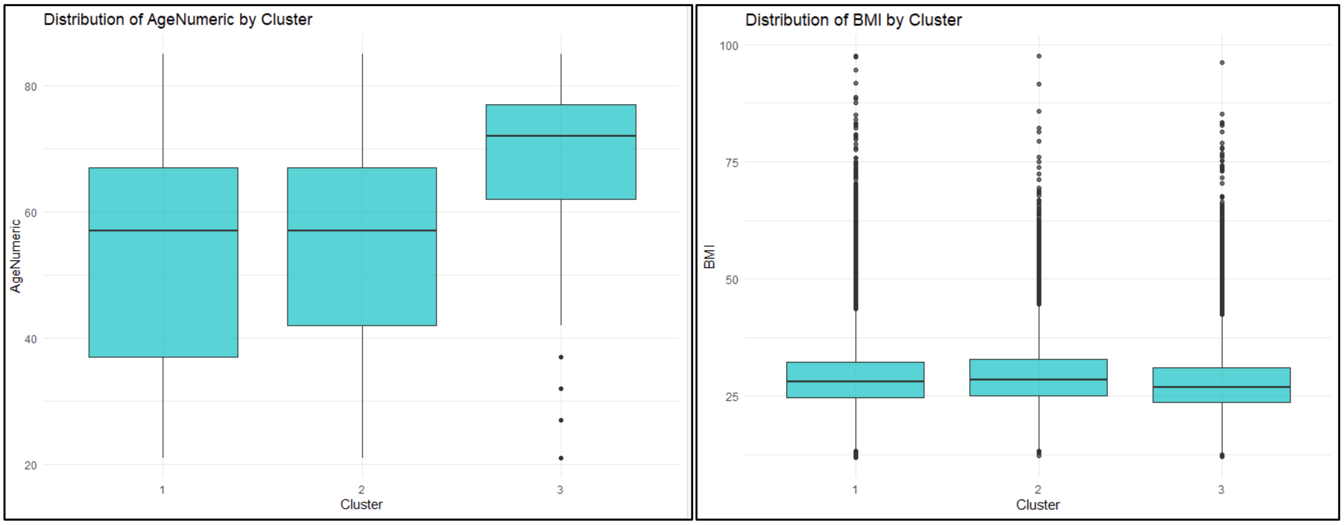
Age and BMI

Table 3: Interpretation summary explaining the cluster distribution of Heart Attack Rate based on Mean Age and BMI

Cluster	Mean Age	Mean BMI	Heart Attack Rate
1	54.0	26.5	0.06%
2	55.3	32.8	0.46%
3	69.6	29.3	60.0%

- Cluster 3 shows the highest heart attack rate (60%) and includes older individuals with moderately high BMI.
- Clusters 1 and 2 are predominantly lower risk, with younger average ages and fewer comorbid conditions

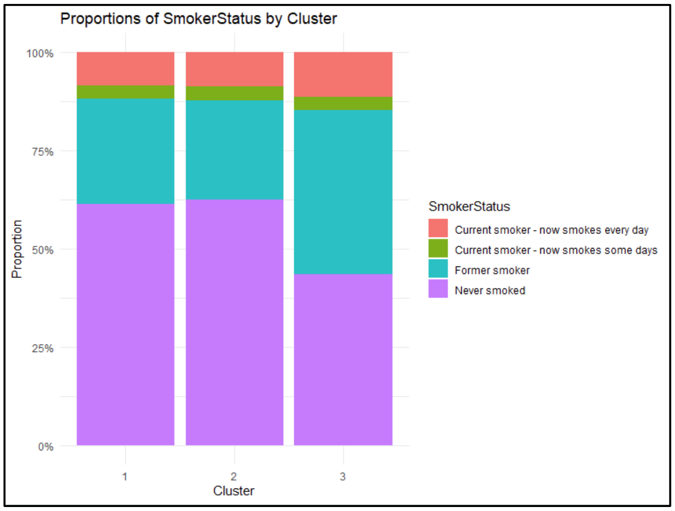
Figure 7: Interpretation summary of age (left) and BMI (right) distribution across PCA-based clusters



- Boxplots show that Cluster 3 has the highest median age.
- BMI is relatively similar across clusters, though Cluster 2 has the highest variability.

Smoker Status

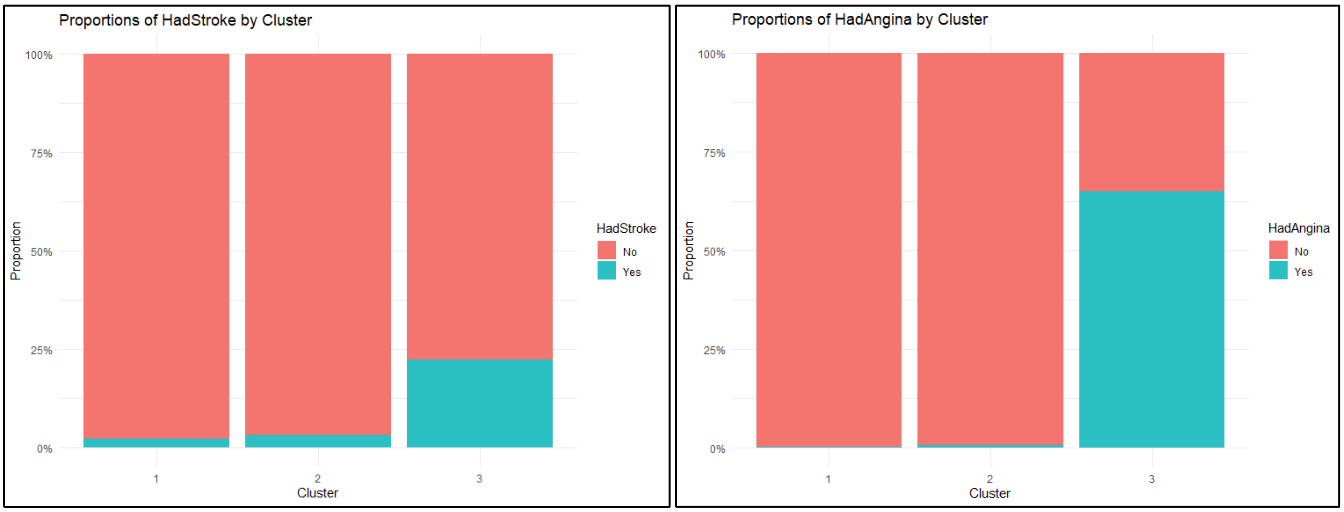
Figure 8: Interpretation summary of smoking behavior distribution across PCA-based clusters

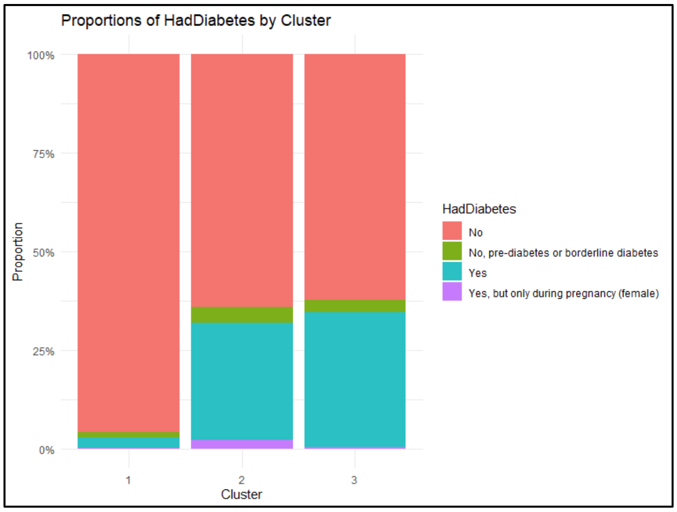


- Cluster 3 has the highest proportion of former smokers (teal bar).
- It also has the lowest proportion of never smokers (purple bar) compared to Clusters 1 and 2.
- It also has a larger share of current smokers (every day) (red bar) than the others.

Comorbidities

Figure 9: Interpretation summary of comorbidity distribution (stroke (left), angina (right), diabetes(bottom)) across PCA-based clusters





- Cluster 3 is significantly associated with higher proportions of stroke, angina, and diabetes.
- Clear clustering of high-risk individuals based on these factors.

iii. Clustering Conclusion

Three separate population groupings with different cardiovascular risk profiles were identified by clustering. Because of their advanced age, comorbidities, and history of smoking, Cluster 3 is considered a high-risk group. Low-risk profiles are represented by Clusters 1 and 2, while BMI may help to further distinguish them. These groups can guide focused screening techniques, function as extra characteristics to enhance supervised models (like Random Forest), and support the process of risk assessment for public health initiatives.

2.5 Classification

To evaluate the predictability of heart attack risk, we implemented and compared four supervised learning models using the PCA-reduced dataset and four supervised learning models on the cleaned dataset with PCA-reduced clusters added back as an attribute:

- **Logistic Regression**: A fast, interpretable baseline model
- **Random Forest**: A robust ensemble model that handles noise and imbalance well
- **Naive Bayes**: Efficient for categorical-heavy datasets
- **XGBoost**: A gradient boosting model known for high accuracy
- **J48:** Mostly a better model than CART and can handle categorical data efficiently.
- **AdaBoost:** Combines weak learners and is good with noise or imbalanced classes.

i. PCA-reduced Model

Each PCA-reduced model was trained on 80% of the data and evaluated on the remaining 20%. The following metrics were computed: **Accuracy**, **Sensitivity**, **Precision**, **F1 Score**, and **AUC (Area Under ROC Curve)**. The results are summarized in the table below:

Table 4: Comparison of Classification Models Based on Performance Metrics on PCA-reduced Model

Model	Accuracy	Sensitivity	Precision	F1 Score	AUC
Logistic Regression	0.9751	0.6870	0.8285	0.7512	0.9923
Random Forest	0.9773	0.7343	0.8311	0.7797	0.9935
Naive Bayes	0.9709	0.6911	0.7546	0.7214	0.9897
XGBoost	0.9758	0.7406	0.8018	0.7700	0.9928

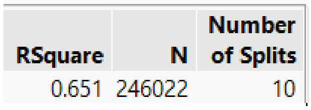
Interpretation: All models performed well, but Random Forest had the highest overall scores across metrics, making it the best-performing model for this dataset. XGBoost was a close second with slightly lower accuracy but comparable sensitivity and AUC.

Decision Tree: In addition to comparing classification models, we also created a **decision tree using the full PCA-reduced dataset** in JMP. This tree allowed us to visualize how combinations of principal components (PCs) contribute to heart attack predictions.

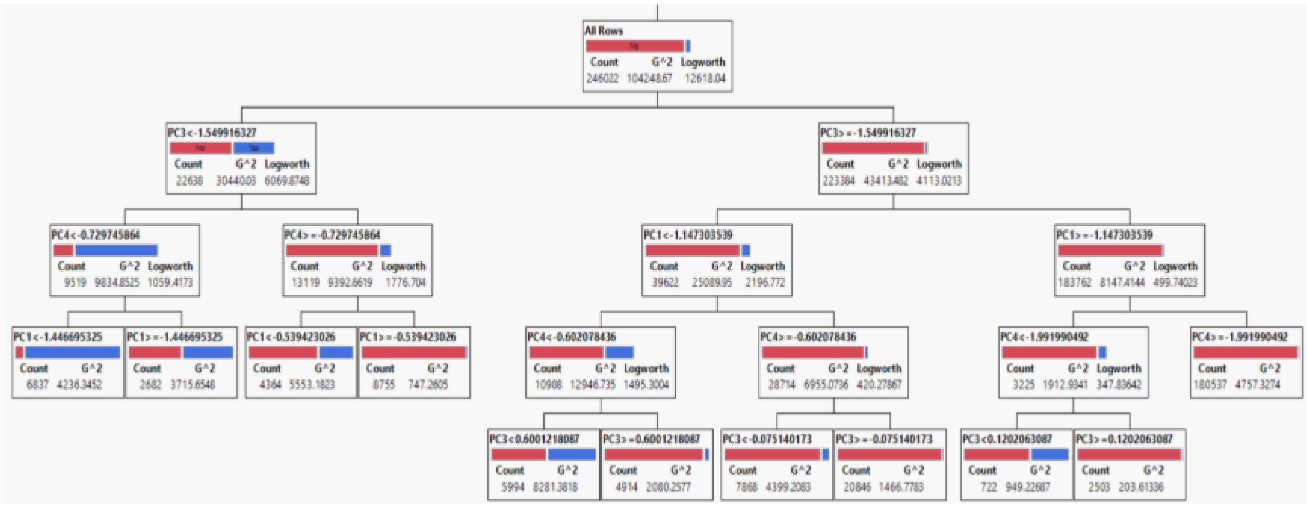
The root node of the tree splits on **PC3**, which was previously associated with **smoking behavior, BMI, and prior heart attack history**. This reinforces its importance in predicting cardiovascular risk. Further splits involve PC1 (chronic health issues) and PC4 (mental health and demographics), showing how multiple health dimensions interact.

This model aids interpretability by offering a **clear, visual rule-based structure** for risk stratification. It complements the more complex black-box models (e.g., XGBoost and Random Forest) by showing how the PCA features can logically segment the population. The decision tree's structure confirms that a combination of **chronic illness (PC1)**, **smoking/weight (PC3)**, and **psychosocial or demographic factors (PC4)** meaningfully predicts heart attack risk in this dataset.

Figure 10: Decision tree on PCA reduced whole dataset.



RSquare	N	Number of Splits
0.651	246022	10



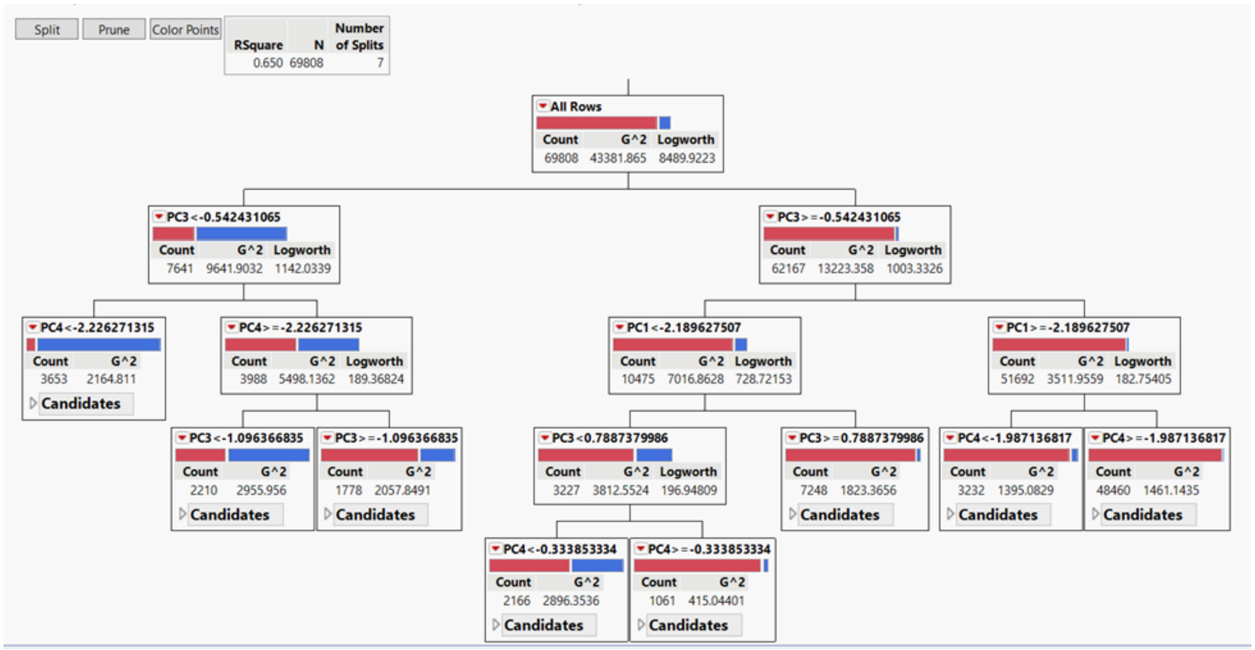
To further understand the high-risk population, we built a separate **decision tree exclusively for Cluster 2** (which had the highest heart attack rate at 9.36%). This model helped us identify what **distinct patterns** exist within that high-risk group.

As shown in the tree below, the root split occurs on **PC3**, which reflects **smoking behavior, BMI, and heart attack history**. The model then splits on **PC4** (associated with mental health and demographics) and **PC1** (chronic conditions like diabetes and difficulty walking), showing layered patterns of vulnerability.

For example:

- Individuals with very low PC3 and very low PC4 values form a clearly defined high-risk subgroup (left branch).
- Those with higher PC3 but low PC1 still show elevated risk, revealing that even without chronic disease, **smoking and poor general health** can drive risk.

Figure 11: Decision Tree on high-risk group (cluster 2)



This tree enables actionable segmentation of the high-risk group and supports targeted intervention planning. It also validates the clustering structure by exposing logical, data-driven splits based on principal component profiles.

ii. Cleaned dataset with PCA-based clusters added back as a feature

The k-means clustering was performed on PCA-transformed data, and the clusters were added back into the cleaned dataset as features to enrich the cleaned dataset.

Each cleaned dataset with PCA-based clusters added back was trained on 70% data and tested on 30% data. The data was trained using four models in total. The following metrics were evaluated using models: Accuracy, Kappa, F1, Precision, Recall, ROC AUC, PRC AUC.

Table 5: Comparison of Classification Models Based on Performance Metrics on cleaned dataset with PCA-based clusters added back

Model	Accuracy	Kappa	F1 (Yes)	Precision (Yes)	Recall (Yes)	ROC AUC	PRC AUC	Notes
Random Forest	96.96%	0.7023	0.718	0.723	0.714	0.981	0.824	Best overall
J48 (C4.5 Tree)	96.96%	0.6899	0.706	0.743	0.672	0.924	0.734	Most interpretable
Naive Bayes	93.94%	0.5621	0.592	0.467	0.810	0.956	0.493	Best recall, but low precision
AdaBoost	96.72%	0.6702	0.688	0.712	0.664	0.980	0.698	Strong performance, second to RF

a. Interpretation

In every metric, Random Forest continuously produced the best overall performance. With a Kappa of 0.70 and an accuracy of 96.96%, it demonstrated strong agreement with actual labels that went above chance. Additionally, it earned the greatest ROC AUC (0.981) and PRC AUC (0.824), indicating high confidence and reliability in its positive class predictions (heart attack = "Yes") and the best balance between precision (72.3%) and recall (71.4%).

The J48 Decision Tree had a slightly lower Kappa and recall, but it fared similarly in terms of accuracy (96.96%). But compared to Random Forest, it demonstrated more precision (74.3%), which means fewer false positives. Because of this, J48 can be helpful in situations when it's important to avoid overestimating risk, and its clear structure makes it perfect for interpretation in clinical or public health settings.

Of all the models, Naive Bayes had the highest recall (81.0%), making it very effective in detecting heart attack cases. It did, however, have the lowest precision (46.7%), which indicates a significant rate of false positives. In pre-screening or early warning systems, where catching all potential positive instances is more crucial than accuracy, this model might be helpful.

AdaBoost maintained strong precision (71.2%) and F1 score (0.688) while achieving somewhat lower accuracy (96.72%) than Random Forest. Its performance was quite competitive and can be regarded as a strong substitute, particularly where model complexity or training time are issues.

b. Insights

A potent method for estimating the risk of a heart attack was the combination of supervised machine learning with unsupervised learning (PCA + K-means clustering). In order to successfully fill the feature space with latent structure that conventional features alone might not disclose, we clustered PCA-transformed data and then reintroduced cluster labels into the dataset.

Among the models that were examined were:

- The best overall performer was Random Forest, which has outstanding AUC scores and a good mix between precision and recall, making it very dependable for real-world implementation.
- For stakeholders or clinicians who need interpretability, J48 provides a clear decision-making process.
- Although less accurate, Naive Bayes was able to identify almost all real heart attack patients and could be useful in general screening situations.
- A quick, well-balanced model with great potential for use in real-time or resource-constrained settings, AdaBoost stood out.

In addition to enhancing model performance, clustering revealed information about patient subgroups with varying cardiovascular risk levels. These findings have the potential to inspire targeted health interventions, direct future research, and be incorporated into preventative care decision-support technologies.

c. Random Forest Model Performance: A Comparison with and Without Clusters

Table 6: Comparison of Random Forest Model Performance when including cluster as a feature vs not as a feature

Metric	Random Forest (With Clusters)	Model B – Random Forest (No Clusters)	📌 Observation
Accuracy	96.96%	93.98%	✅ With clusters is clearly more accurate
Kappa	0.7023	0.4427	✅ Much stronger agreement with clusters
Sensitivity ("No")	98.4%	96.53%	👍 Slight improvement
Specificity ("Yes")	71.4%	**49.78%** ❗	❌ Huge drop in detecting heart attacks
NPV	97.0%	97.08%	⚖️ About the same
PPV	72.3%	45.34%	❌ Much worse without clusters
Balanced Accuracy	84.9% (approximate)	73.16%	✅ Big gain with clusters

Findings

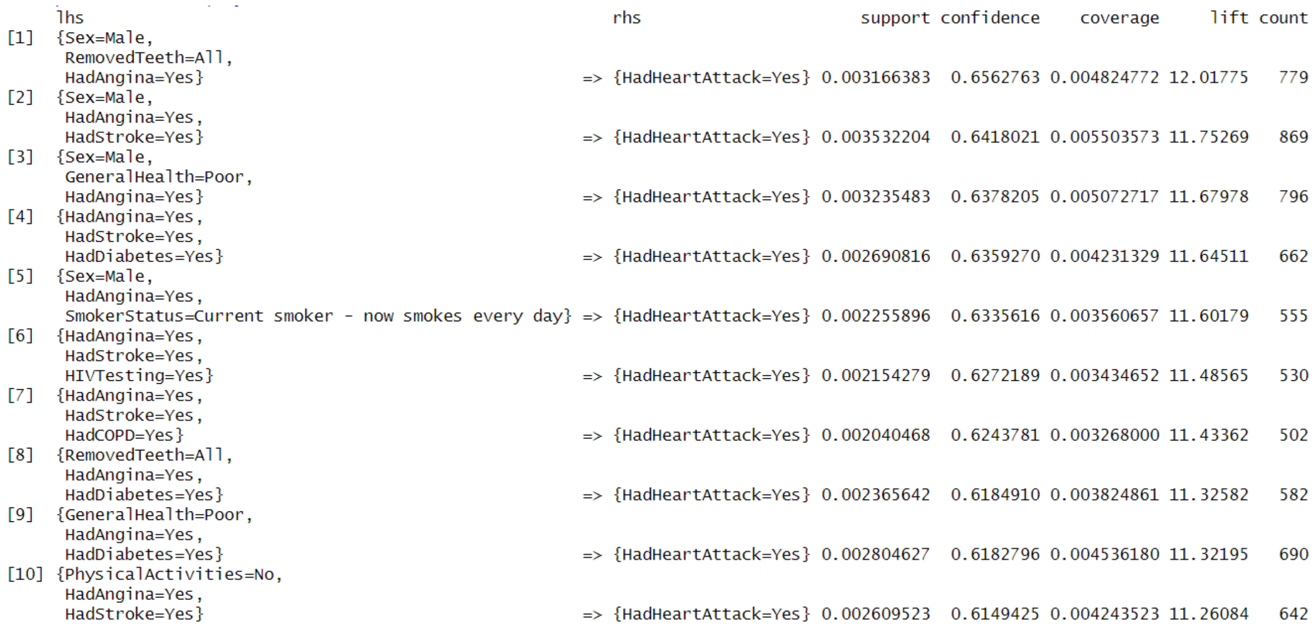
Without compromising accuracy or NPV, the model's capacity to identify heart attacks (specificity, PPV, and balanced accuracy) was greatly increased by the addition of cluster labels obtained from PCA-transformed data.

When it comes to forecasting heart attack cases, Model B's specificity fell to just under 50%, which is hardly better than guesswork.

Random Forest's capacity to identify positive cases (heart attacks) and minimize false positives was significantly enhanced by the PCA-based cluster feature. This demonstrates that using cluster labels as features is not only appropriate but also beneficial.

2.6 Association Rule Mining

To identify interpretable patterns between lifestyle behaviors, medical history, and heart attack occurrence, we applied **association rule mining** using the Apriori algorithm in R. The target outcome was set as `HadHeartAttack=Yes`, and rules were generated using a support threshold of 0.002 and a confidence threshold of 0.6. The top rules uncovered strong and recurring combinations of risk factors—including chronic conditions like angina, stroke, and diabetes, as well as lifestyle factors such as smoking and inactivity. Below are the highest-lift rules, indicating the strongest associations with heart attack risk:



```
     lhs                                                        rhs                      support confidence    coverage     lift count
[1]  {Sex=Male,                                                                                                                        
      RemovedTeeth=All,                                                                                                                
      HadAngina=Yes}                                         => {HadHeartAttack=Yes} 0.003166383   0.6562763 0.004824772 12.01775   779
[2]  {Sex=Male,                                                                                                                        
      HadAngina=Yes,                                                                                                                   
      HadStroke=Yes}                                         => {HadHeartAttack=Yes} 0.003532204   0.6418021 0.005503573 11.75269   869
[3]  {Sex=Male,                                                                                                                        
      GeneralHealth=Poor,                                                                                                              
      HadAngina=Yes}                                         => {HadHeartAttack=Yes} 0.003235483   0.6378205 0.005072717 11.67978   796
[4]  {HadAngina=Yes,                                                                                                                   
      HadStroke=Yes,                                                                                                                   
      HadDiabetes=Yes}                                       => {HadHeartAttack=Yes} 0.002690816   0.6359270 0.004231329 11.64511   662
[5]  {Sex=Male,                                                                                                                        
      HadAngina=Yes,                                                                                                                   
      SmokerStatus=Current smoker - now smokes every day} => {HadHeartAttack=Yes} 0.002255896   0.6335616 0.003560657 11.60179   555
[6]  {HadAngina=Yes,                                                                                                                   
      HadStroke=Yes,                                                                                                                   
      HIVTesting=Yes}                                        => {HadHeartAttack=Yes} 0.002154279   0.6272189 0.003434652 11.48565   530
[7]  {HadAngina=Yes,                                                                                                                   
      HadStroke=Yes,                                                                                                                   
      HadCOPD=Yes}                                           => {HadHeartAttack=Yes} 0.002040468   0.6243781 0.003268000 11.43362   502
[8]  {RemovedTeeth=All,                                                                                                                
      HadAngina=Yes,                                                                                                                   
      HadDiabetes=Yes}                                       => {HadHeartAttack=Yes} 0.002365642   0.6184910 0.003824861 11.32582   582
[9]  {GeneralHealth=Poor,                                                                                                              
      HadAngina=Yes,                                                                                                                   
      HadDiabetes=Yes}                                       => {HadHeartAttack=Yes} 0.002804627   0.6182796 0.004536180 11.32195   690
[10] {PhysicalActivities=No,                                                                                                           
      HadAngina=Yes,                                                                                                                   
      HadStroke=Yes}                                         => {HadHeartAttack=Yes} 0.002609523   0.6149425 0.004243523 11.26084   642
```

Association rule mining revealed consistent patterns of chronic illness and poor health behaviors among individuals with heart attacks. The most frequently appearing predictor was **HadAngina**, followed by **HadStroke**, **HadDiabetes**, and **Male sex**. Rules combining these conditions yielded high **lift values (>11)**, indicating a **strong association** with heart attack occurrence. Additional lifestyle factors like smoking, physical inactivity, and poor general health further amplified risk. These rules provide interpretable insights and reinforce findings from clustering and classification stages.

3. Conclusion

This project successfully applied a comprehensive data science workflow to analyze heart attack risk using large-scale health survey data. Through data cleaning, exploratory data analysis, dimensionality reduction, clustering, classification, and association rule mining, we uncovered important lifestyle and medical patterns associated with heart disease.

Principal Component Analysis (PCA) enabled dimensionality reduction while preserving key relationships among features. Clustering was performed on the PCA-reduced dataset. While PCA-based clustering allowed for a compact and interpretable grouping in principal component space, the data was significantly hard to interpret on its own. **Adding the clusters back to the cleaned datasets** for model training proved effective, as it **significantly improved heart attack predictability and general interpretability of clusters.** Clusters produced **distinct risk segments with a 60% heart attack rate, while the other 2 clusters have less than 1% heart attack rate**, clearly separating low- and high-risk individuals based on interpretable attributes like age and smoker status.

4. Appendix

i. Code for comparison of different classification algorithm.

📦 Install & load required packages

```
install.packages(c("caret", "pROC", "e1071", "naivebayes", "xgboost")) library(caret) library(pROC) library(e1071) library(naivebayes) library(xgboost)
```

📂 Load your PCA-reduced dataset (or original)

```
data <- read.csv("C:/Users/18328/Desktop/Data Science & R/Project/heart_pca6_with_target.csv") data$HadHeartAttack <-
as.factor(data$HadHeartAttack)
```

🎯 Split into training and test sets

```
set.seed(123) index <- createDataPartition(data$HadHeartAttack, p = 0.8, list = FALSE) train <- data[index, ] test <- data[-index, ]
```

📊 Storage for predictions

```
model_metrics <- data.frame(Model = character(), Accuracy = numeric(), Sensitivity = numeric(), Precision = numeric(), F1 = numeric(), AUC =
numeric(), stringsAsFactors = FALSE)
```

🔍 Function to evaluate performance

```
evaluate_model <- function(true, predicted, probs, model_name) { cm <- confusionMatrix(predicted, true, positive = "Yes") roc_obj <- roc(true,
probs) auc_val <- auc(roc_obj) acc <- cm$overall["Accuracy"] sens <- cm$byClass["Sensitivity"] prec <- cm$byClass["Precision"] f1 <-
cm$byClass["F1"]

model_metrics <<- rbind(model_metrics, data.frame( Model = model_name, Accuracy = acc, Sensitivity = sens, Precision = prec, F1 = f1, AUC =
auc_val )) }
```

✅ 1. Logistic Regression

```
model_glm <- glm(HadHeartAttack ~ ., data = train, family = "binomial") probs_glm <- predict(model_glm, test, type = "response") pred_glm <-
ifelse(probs_glm > 0.5, "Yes", "No") evaluate_model(test$HadHeartAttack, as.factor(pred_glm), probs_glm, "Logistic Regression")
```

✅ 2. Random Forest

```
model_rf <- randomForest(HadHeartAttack ~ ., data = train) pred_rf <- predict(model_rf, test) probs_rf <- predict(model_rf, test, type = "prob")[,
"Yes"] evaluate_model(test$HadHeartAttack, pred_rf, probs_rf, "Random Forest")
```

✅ 3. Naive Bayes

```
model_nb <- naive_bayes(HadHeartAttack ~ ., data = train) pred_nb <- predict(model_nb, test) probs_nb <- predict(model_nb, test, type = "prob")[,
"Yes"] evaluate_model(test$HadHeartAttack, pred_nb, probs_nb, "Naive Bayes")
```

✅ 4. XGBoost (requires matrix format)

```
train_matrix <- model.matrix(HadHeartAttack ~ . -1, data = train) test_matrix <- model.matrix(HadHeartAttack ~ . -1, data = test) train_label <-
ifelse(train$HadHeartAttack == "Yes", 1, 0) test_label <- ifelse(test$HadHeartAttack == "Yes", 1, 0)

dtrain <- xgb.DMatrix(data = train_matrix, label = train_label) dtest <- xgb.DMatrix(data = test_matrix, label = test_label)

xgb_model <- xgboost(data = dtrain, nrounds = 100, objective = "binary:logistic", verbose = 0) probs_xgb <- predict(xgb_model, dtest) pred_xgb <-
ifelse(probs_xgb > 0.5, "Yes", "No") evaluate_model(test$HadHeartAttack, as.factor(pred_xgb), probs_xgb, "XGBoost")
```

📋 View comparison

```
print(model_metrics)
```

ii. R code for Association Rule Mining:

```
📦 Install required packages (run only once)

install.packages("arules") library(arules)

📁 Load the cleaned dataset

data <- read.csv("C:/Users/18328/Desktop/Data Science & R/Project/heart_cleaned_for_weka.csv")

✅ Convert all columns to factors (required for arules)

data[] <- lapply(data, as.factor)

🔄 Convert data to "transactions" format

trans_data <- as(data, "transactions").

 Mine association rules using Apriori

rules <- apriori( trans_data, parameter = list(supp = 0.005, conf = 0.6, target = "rules"), appearance = list(rhs =
c("HadHeartAttack=Yes")), control = list(verbose = TRUE) )

Inspect the top 10 rules

inspect(sort(rules, by = "lift")[1:10])
```

i.

Steps took to create tableau dashboard:

1. Loaded the Dataset

2. Created Key Calculated Fields

To prepare the dataset for meaningful analysis:

BMI Category

Grouped BMI into: Underweight, Normal, Overweight, Obese

```
IF [BMI] >= 30 THEN "Obese"
ELSEIF [BMI] >= 25 THEN "Overweight"
ELSEIF [BMI] >= 18.5 THEN "Normal"
ELSE "Underweight"
END
```

HeartAttackRate (%)

Turned Yes/No into numeric 1/0 for average calculation

```
IF [Had Heart Attack] = "Yes" THEN 1 ELSE 0 END
```

Lifestyle Risk Group

Grouped users based on physical activity + smoking

```
IF [Smoker Status] = "Current" AND [Physical Activities] = "No" THEN "High Risk"
ELSEIF [Smoker Status] = "Current" THEN "Smoker Only"
ELSEIF [Physical Activities] = "No" THEN "Inactive Only"
ELSE "Low Risk"
END
```

3. Built 4 Individual Charts

Each one focused on a different risk factor:

Sheet 1: BMI Category × Smoker Status (Bubble Chart)

X-axis: BMI Category

Y-axis: Smoker Status

Size: Number of Records

Color: AVG HeartAttackRate (%)

Sheet 2: Age Category × General Health (Treemap / Heatmap)

Columns: Age Category

Rows: General Health

Size: People in each group

Color: Average HeartAttackRate (%)

Sheet 3: Sleep Hours vs. Heart Attack (Bar Chart)

X-axis: Had Heart Attack (Yes/No)

Y-axis: Average Sleep Hours

Color: Heart Attack Status

Sheet 4: Lifestyle Risk Group vs. Heart Attack (Stacked Bar)

X-axis: Lifestyle Risk Group

Y-axis: Number of Records

Color: Had Heart Attack

4. Customized Tooltips

Used emojis and visual cues:

🧑 Group: Obese × Smoker
❤️ Risk: 21.5%
🛏️ Avg Sleep: 6.3 hours

5. Added Interactive Filters

Added filters for:

Smoker Status

Age Category

Sex

Physical Activities

Lifestyle Risk Group

Had Heart Attack

Set filters to "Apply to All Using This Data Source"
Placed filters neatly on the right side of the dashboard

6. Built the Dashboard

Title: ❤️ Heart Health Risk Explorer

Subtitle: Explore how lifestyle and health factors impact heart attack outcomes

Layout:

Top: Bubble Chart and Heatmap

Bottom: Bar and Stacked Bar

Used containers for clean structure

7. Added Hover-Based Highlight Action

Dashboard → Actions → Add Highlight

Source/Target: All 4 charts

Run on: Hover

Target field: All 4 charts

Enables cross-chart interaction by hovering

8. Final Touches

Unified color scheme (Blue = No, Orange = Yes)

Cleaned axis labels and legends

Previewed layout at 1000 × 800 for desktop sharing

ii. R Code for PCA Clustering and clean data + cluster model predictions (note some models has been done using WEKA on 70/30 data.

```
# -------------------------------
# Libraries and Data Loading
# -------------------------------
library(readxl)
library(factoextra)
library(dplyr)
library(tidyverse)
library(ClusterR)
library(caret)
library(kml)
library(writexl)
library(foreign)
library(rpart)
library(rpart.plot)
library(pROC)

# Load PCA and cleaned datasets
pca_data <- read_excel("path/to/heart_pca6_with_target.xlsx")
```

```r
cleaned_data <- read_excel("path/to/heart_cleaned.xlsx")

# ------------------------------
# Preprocessing
# ------------------------------

# Recode AgeCategory into ordered factor
age_levels <- c(
  "Age 18 to 24", "Age 25 to 29", "Age 30 to 34", "Age 35 to 39", "Age 40 to 44",
  "Age 45 to 49", "Age 50 to 54", "Age 55 to 59", "Age 60 to 64", "Age 65 to 69",
  "Age 70 to 74", "Age 75 to 79", "Age 80 or older"
)
age_map <- c(
  "Age 18 to 24" = 21, "Age 25 to 29" = 27, "Age 30 to 34" = 32, "Age 35 to 39" = 37,
  "Age 40 to 44" = 42, "Age 45 to 49" = 47, "Age 50 to 54" = 52, "Age 55 to 59" = 57,
  "Age 60 to 64" = 62, "Age 65 to 69" = 67, "Age 70 to 74" = 72, "Age 75 to 79" = 77,
  "Age 80 or older" = 85
)

cleaned_data$AgeCategory <- factor(cleaned_data$AgeCategory, levels = age_levels, ordered = TRUE)
cleaned_data$AgeNumeric <- age_map[as.character(cleaned_data$AgeCategory)]

# Remove near-zero variance features
nzv_cols <- nearZeroVar(cleaned_data, saveMetrics = TRUE)
nzv_cols[nzv_cols$nzv, ]

# Remove highly correlated features if needed (cutoff = 0.9)
cor_matrix <- cor(cleaned_data %>% select(where(is.numeric)), use = "complete.obs")
findCorrelation(cor_matrix, cutoff = 0.9)

# ------------------------------
# Clustering (MiniBatch KMeans)
# -------------------------------
set.seed(123)
pca_data_sample <- pca_data[, 1:6]
scaled_data <- scale(pca_data_sample)
sampled_data <- scaled_data[sample(1:nrow(scaled_data), 8000), ]

# Elbow and silhouette methods
fviz_nbclust(sampled_data, kmeans, method = "wss") + ggtitle("Elbow Method")
fviz_nbclust(sampled_data, kmeans, method = "silhouette") + ggtitle("Silhouette Method")

# 70/30 split
train_index <- createDataPartition(pca_data$HadHeartAttack, p = 0.7, list = FALSE)
train_data <- pca_data[train_index, ]
test_data <- pca_data[-train_index, ]

# Fit MiniBatchKmeans on training set only
train_scaled <- scale(train_data %>% select(PC1:PC6))
kmeans_mb <- MiniBatchKmeans(train_scaled, clusters = 3, batch_size = 5000, num_init = 5, max_iters = 100)

# Predict cluster for full dataset
train_clusters <- predict_MBatchKMeans(train_scaled, kmeans_mb$centroids)
test_scaled <- scale(test_data %>% select(PC1:PC6))
test_clusters <- predict_MBatchKMeans(test_scaled, kmeans_mb$centroids)

# Combine clusters with original cleaned data
all_clusters <- numeric(nrow(pca_data))
all_clusters[train_index] <- train_clusters
all_clusters[-train_index] <- test_clusters
heart_clustered <- cbind(cleaned_data, cluster = all_clusters)
```

```r
# -------------------------------
# Factor Conversion
# -------------------------------
factor_cols <- c(
  "Sex", "GeneralHealth", "PhysicalActivities", "RemovedTeeth", "HadHeartAttack",
  "HadAngina", "HadStroke", "HadAsthma", "HadCOPD", "HadDepressiveDisorder",
  "HadKidneyDisease", "HadDiabetes", "DifficultyWalking", "DifficultyDressingBathing",
  "SmokerStatus", "ECigaretteUsage", "RaceEthnicityCategory", "AlcoholDrinkers",
  "HIVTesting", "PneumoVaxEver", "HighRiskLastYear", "CovidPos", "cluster"
)

heart_clustered <- heart_clustered %>%
  mutate(across(all_of(factor_cols), as.factor))

# Save outputs
write_xlsx(heart_clustered, "heart_clustered.xlsx")
write.arff(heart_clustered, "heart_clustered.arff")

# -------------------------------
# Modeling (Decision Tree with Cluster)
# -------------------------------
train_data <- heart_clustered[train_index, ]
test_data <- heart_clustered[-train_index, ]

# Convert non-numeric columns to factor (safe fallback)
convert_non_numeric_to_factors <- function(df) {
  df[] <- lapply(df, function(x) {
    if (!is.numeric(x)) as.factor(x) else x
  })
  return(df)
}
```

```
train_data <- convert_non_numeric_to_factors(train_data)
test_data <- convert_non_numeric_to_factors(test_data)

# Build full tree model
predictor_vars <- setdiff(names(train_data), "HadHeartAttack")
formula_all <- as.formula(paste("HadHeartAttack ~", paste(predictor_vars, collapse = " + ")))

tree_model <- rpart(formula_all,
        data = train_data,
        method = "class",
        control = rpart.control(cp = 0.002, minsplit = 20))

# Plot and evaluate
rpart.plot(tree_model, type = 2, extra = 106, cex = 0.6)
printcp(tree_model)
tree_model$variable.importance

# Predict and evaluate
probs <- predict(tree_model, test_data, type = "prob")[, "Yes"]
pred_custom <- ifelse(probs > 0.3, "Yes", "No") %>% factor(levels = c("No", "Yes"))
conf_matrix_cluster <- confusionMatrix(pred_custom, test_data$HadHeartAttack)
print(conf_matrix_cluster)

# ROC Curve
roc_obj <- roc(test_data$HadHeartAttack, probs)
plot(roc_obj, main = "ROC Curve - Heart Attack Prediction", col = "#2C3E50", lwd = 2)
abline(a = 0, b = 1, lty = 2, col = "gray")
cat("AUC:", auc(roc_obj), "\n")

# -------------------------------
# Modeling Without Cluster
# -------------------------------

predictor_vars_nocluster <- setdiff(names(train_data), c("HadHeartAttack", "cluster"))

formula_nocluster <- as.formula(paste("HadHeartAttack ~", paste(predictor_vars_nocluster, collapse = " + ")))


tree_model_nocluster <- rpart(formula_nocluster,

                        data = train_data,

                        method = "class",

                        control = rpart.control(cp = 0.002, minsplit = 20))


# Predictions without cluster

probs_nocluster <- predict(tree_model_nocluster, test_data, type = "prob")[, "Yes"]

pred_nocluster <- ifelse(probs_nocluster > 0.3, "Yes", "No") %>% factor(levels = c("No", "Yes"))

conf_matrix_nocluster <- confusionMatrix(pred_nocluster, test_data$HadHeartAttack)

print(conf_matrix_nocluster)
```