TITLE: Predicting Health Conditions in Medical Students Using Machine Learning (Med J.A.R.V.I.S)

Contributor 1
Safin Ahmed Orko
22299060
Computer Science and Engineering

Contributor 2
Raian Kibria Rohan
22299407
Computer Science and Engineering

Course Code: CSE422
Semester: Spring 25

Submitted to
Mahjabin Chowdhury [MCW], Rafiad Sadat Shahir [RSS]

Table of Contents

1. **Introduction** **Page**
   a. Aim and Motivation ………………………………. 4

2. **Dataset Description**
   a. Number of features ……………………………….. 5
   b. Number of data points …………………………… 5
   c. Types of features are in the dataset ……………….. 5

3. **Data Analysis**
   a. EDA ……………………………………………… 6

4. **Dataset Pre-processing**
   a. Handling null and duplicate values ……………… 9
   b. Encoding categorical values……………………… 9
   c. Removing irrelevant features…………………….. 9
   d. Feature scaling…………………………………… 9

5. **Model Training**

   a. Decision Tree…………………………………… 11
   b. Neural Network………………………………… 11
   c. KNN……………………………………………… 11
   d. KMeans Clustering……………………………….. 12

6. **Model Evolution**

a. Prediction Accuracy and f1 score…………………. 12
b. Confusion matrix………………………………….. 13
c. Performance metrics comparison…………………. 13
d. AUC score, ROC curve…………………………… 14

7. **Conclusion………………………………………..** 15

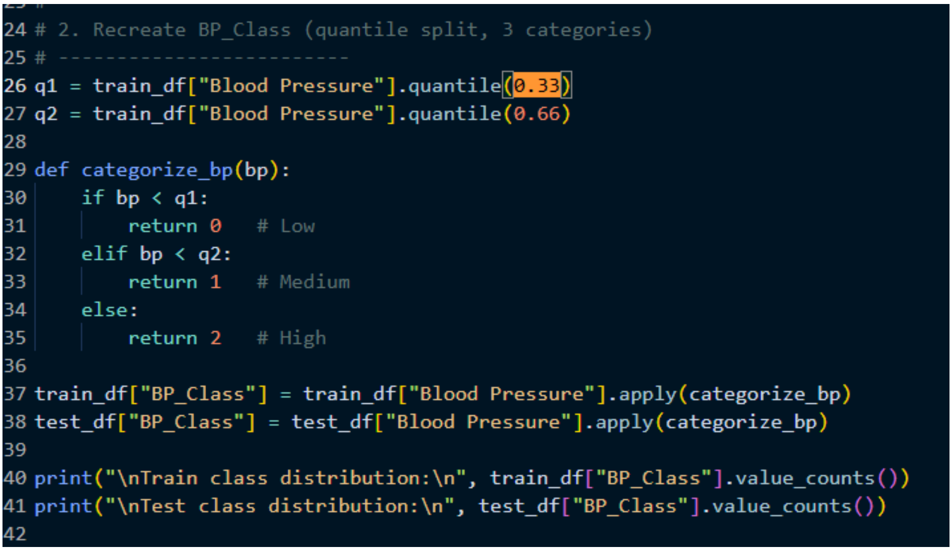
Introduction

The goal of this project is to analyze and predict the relationship of blood pressure with other health and lifestyle factors among medical students. Blood pressure is an important indicator of cardiovascular health and is influenced by various attributes such as BMI, cholesterol levels, smoking habits, and genetic markers like blood type.

The motivation behind this project is two-fold:

1. To explore if machine learning models can classify blood pressure levels into low, medium, and high.
2. To determine which features contribute most strongly to changes in blood pressure, which can provide early preventive insights.

In addition, the project applies both supervised models (Neural Network, KNN, Decision Tree) and unsupervised learning (KMeans Clustering) to compare performance and discover hidden groupings in the dataset.



```
# 2. Recreate BP_Class (quantile split, 3 categories)
# -----------------------
q1 = train_df["Blood Pressure"].quantile(0.33)
q2 = train_df["Blood Pressure"].quantile(0.66)

def categorize_bp(bp):
    if bp < q1:
        return 0    # Low
    elif bp < q2:
        return 1    # Medium
    else:
        return 2    # High

train_df["BP_Class"] = train_df["Blood Pressure"].apply(categorize_bp)
test_df["BP_Class"] = test_df["Blood Pressure"].apply(categorize_bp)

print("\nTrain class distribution:\n", train_df["BP_Class"].value_counts())
print("\nTest class distribution:\n", test_df["BP_Class"].value_counts())

```

Dataset Description

The dataset contains records of 199,979 students with health and lifestyle information.

● Number of Features (before preprocessing): 16

● Number of Features (after preprocessing): 19 (Blood type split into 4 one-hot encoded features and BP class split in 3)

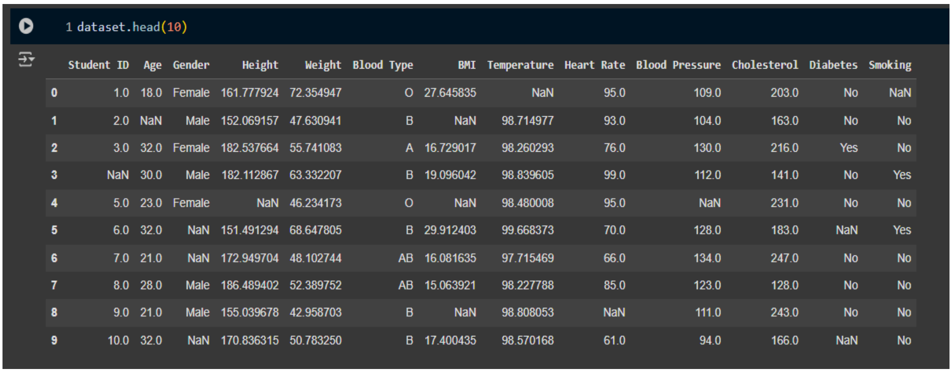
● Types of Features:

○ Numerical: Age, Height, Weight, BMI, Temperature, Heart Rate, Blood Pressure, Cholesterol

○ Categorical (encoded): Gender, Smoking, Diabetes, Blood Type (A, B, AB, O)

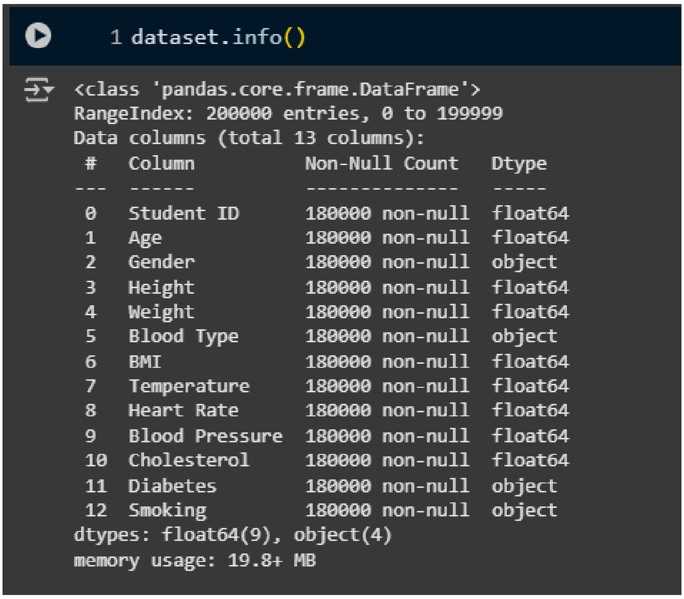
○ Identifier: Student ID (dropped later as irrelevant)

The target variable created was BP_Class, derived from Blood Pressure, split into 3 categories (Low, Medium, High) using quantile thresholds.



```
dataset.head(10)
```

	Student ID	Age	Gender	Height	Weight	Blood Type	BMI	Temperature	Heart Rate	Blood Pressure	Cholesterol	Diabetes	Smoking
0	1.0	18.0	Female	161.777924	72.354947	O	27.645835	NaN	95.0	109.0	203.0	No	NaN
1	2.0	NaN	Male	152.069157	47.630941	B	NaN	98.714977	93.0	104.0	163.0	No	No
2	3.0	32.0	Female	182.537664	55.741083	A	16.729017	98.260293	76.0	130.0	216.0	Yes	No
3	NaN	30.0	Male	182.112867	63.332207	B	19.096042	98.839605	99.0	112.0	141.0	No	Yes
4	5.0	23.0	Female	NaN	46.234173	O	NaN	98.480008	95.0	NaN	231.0	No	No
5	6.0	32.0	NaN	151.491294	68.647805	B	29.912403	99.668373	70.0	128.0	183.0	NaN	Yes
6	7.0	21.0	NaN	172.949704	48.102744	AB	16.081635	97.715469	66.0	134.0	247.0	No	No
7	8.0	28.0	Male	186.489402	52.389752	AB	15.063921	98.227788	85.0	123.0	128.0	No	No
8	9.0	21.0	Male	155.039678	42.958703	B	NaN	98.808053	NaN	111.0	243.0	No	No
9	10.0	32.0	NaN	170.836315	50.783250	B	17.400435	98.570168	61.0	94.0	166.0	NaN	No



```
dataset.info()
```

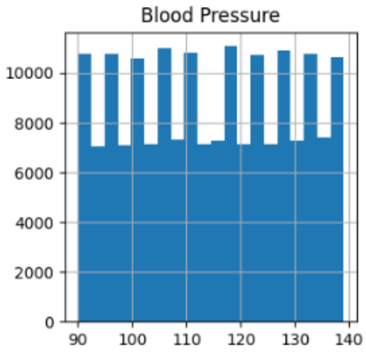
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200000 entries, 0 to 199999
Data columns (total 13 columns):
 #   Column          Non-Null Count   Dtype  
---  ------          --------------   -----  
 0   Student ID      180000 non-null  float64
 1   Age             180000 non-null  float64
 2   Gender          180000 non-null  object 
 3   Height          180000 non-null  float64
 4   Weight          180000 non-null  float64
 5   Blood Type      180000 non-null  object 
 6   BMI             180000 non-null  float64
 7   Temperature     180000 non-null  float64
 8   Heart Rate      180000 non-null  float64
 9   Blood Pressure  180000 non-null  float64
 10  Cholesterol     180000 non-null  float64
 11  Diabetes        180000 non-null  object 
 12  Smoking         180000 non-null  object 
dtypes: float64(9), object(4)
memory usage: 19.8+ MB
```

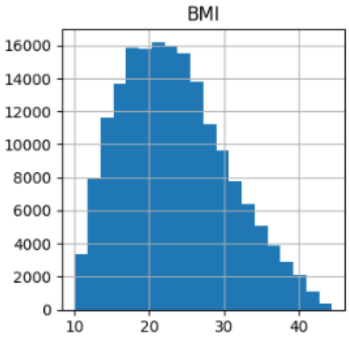
Data Analysis (EDA)

Exploratory Data Analysis (EDA) was performed to visualize the distribution of features and their correlation with blood pressure.

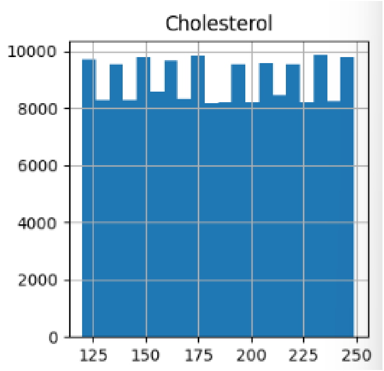
Key findings:

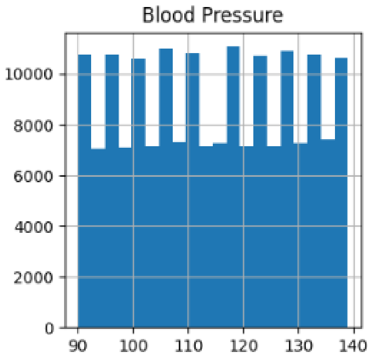
- BMI vs BP: Students with higher BMI generally showed higher blood pressure, confirming medical expectations.





● Cholesterol vs BP: High cholesterol correlated with increased blood pressure values.

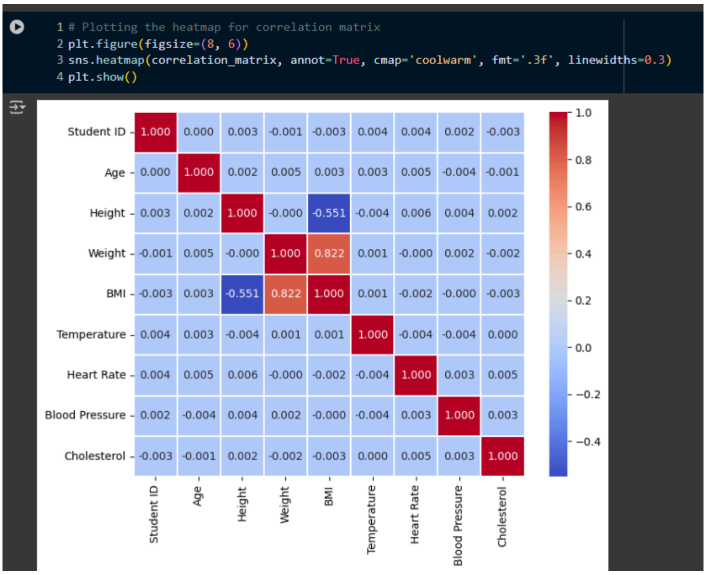




● Smoking vs BP: Smokers were more likely to belong to the High BP category.

● Blood Type: No strong correlation with BP, but slight differences were observed across groups.

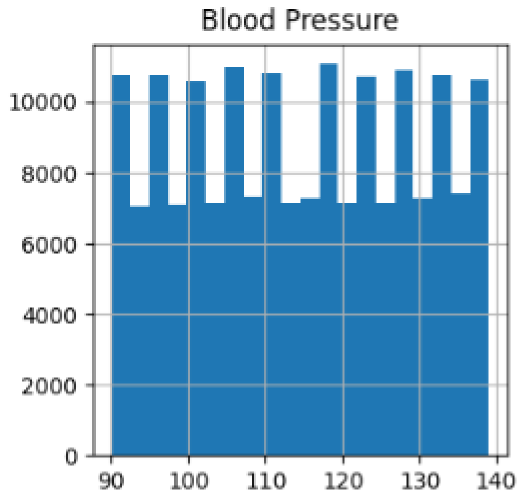
● Correlation Heatmap: BMI, cholesterol, and smoking showed the highest correlation with blood pressure.



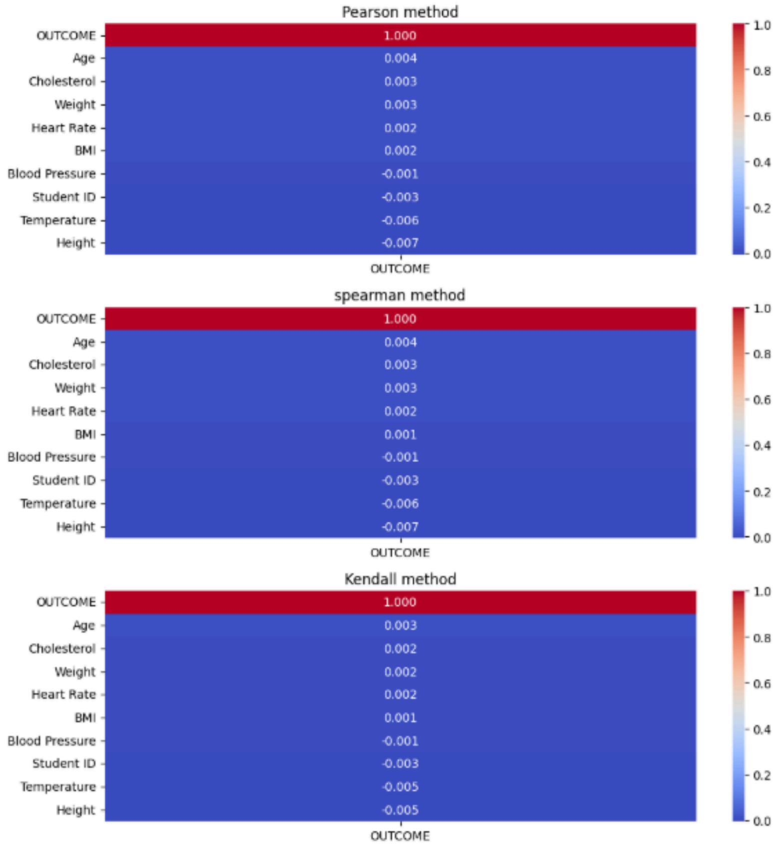
```
1 # Plotting the heatmap for correlation matrix
2 plt.figure(figsize=(8, 6))
3 sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.3f', linewidths=0.3)
4 plt.show()
```

	Student ID	Age	Height	Weight	BMI	Temperature	Heart Rate	Blood Pressure	Cholesterol
Student ID	1.000	0.000	0.003	-0.001	-0.003	0.004	0.004	0.002	-0.003
Age	0.000	1.000	0.002	0.005	0.003	0.003	0.005	-0.004	-0.001
Height	0.003	0.002	1.000	-0.000	-0.551	-0.004	0.006	0.004	0.002
Weight	-0.001	0.005	-0.000	1.000	0.822	0.001	-0.000	0.002	-0.002
BMI	-0.003	0.003	-0.551	0.822	1.000	0.001	-0.002	-0.000	-0.003
Temperature	0.004	0.003	-0.004	0.001	0.001	1.000	-0.004	-0.004	0.000
Heart Rate	0.004	0.005	0.006	-0.000	-0.002	-0.004	1.000	0.003	0.005
Blood Pressure	0.002	-0.004	0.004	0.002	-0.000	-0.004	0.003	1.000	0.003
Cholesterol	-0.003	-0.001	0.002	-0.002	-0.003	0.000	0.005	0.003	1.000

- Boxplots & Histograms: Displayed skewness in blood pressure distribution, which justified quantile-based splitting into classes



Correlation Plot



Dataset Pre-processing

Several preprocessing steps were applied to clean and standardize the dataset:

1. Handling Missing Values:

○ Rows with more than 40% missing data were dropped.

○ For Student ID: missing values were filled by averaging the previous and next IDs.

○ For numerical columns (Age, Height, Weight, BMI, Temperature, Heart Rate, BP, Cholesterol): missing values were replaced with column mean.

○ For categorical columns (Diabetes, Smoking, Gender): missing values filled with majority class (mode).

2. Encoding Categorical Values:

○ Gender → Binary (Male=1, Female=0)

○ Smoking & Diabetes → Binary (Yes=1, No=0)

○ Blood Type → Converted into one-hot encoded features (A, B, AB, O).

3. Scaling:

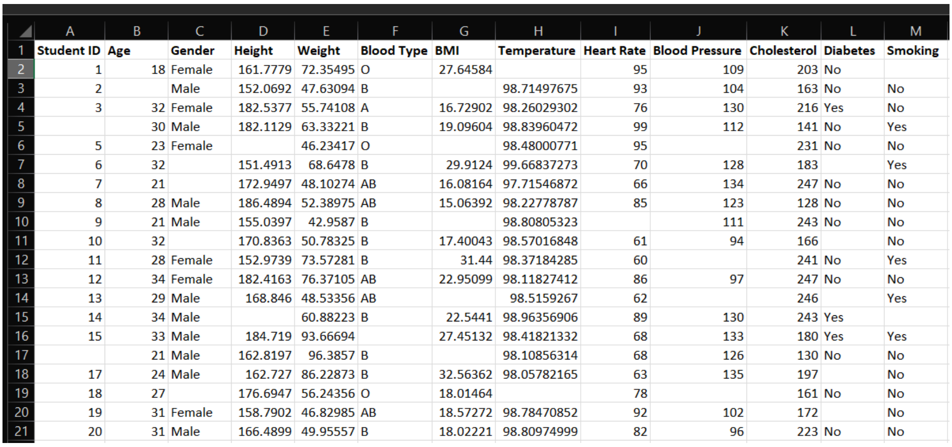
○ StandardScaler applied to numerical features (Age, Height, Weight, BMI, Temperature, Heart Rate, Blood Pressure, Cholesterol).

4. Train-Test Split:

○ Train: 70% (139,985 samples)

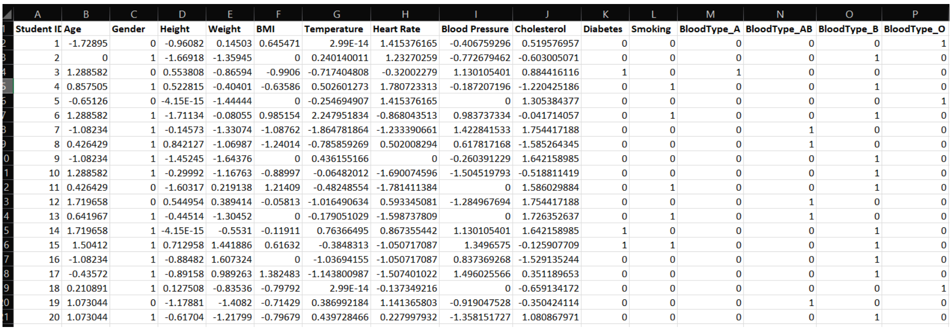
○ Test: 30% (59,994 samples)

○ Stratified by BP_Class distribution.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Student ID	Age	Gender	Height	Weight	Blood Type	BMI	Temperature	Heart Rate	Blood Pressure	Cholesterol	Diabetes	Smoking
2	1	18	Female	161.7779	72.35495	O	27.64584		95	109	203	No	
3	2		Male	152.0692	47.63094	B		98.71497675	93	104	163	No	No
4	3	32	Female	182.5377	55.74108	A	16.72902	98.26029302	76	130	216	Yes	No
5		30	Male	182.1129	63.33221	B	19.09604	98.83960472	99	112	141	No	Yes
6	5	23	Female		46.23417	O		98.48000771	95		231	No	No
7	6	32		151.4913	68.6478	B	29.9124	99.66837273	70	128	183		Yes
8	7	21		172.9497	48.10274	AB	16.08164	97.71546872	66	134	247	No	No
9	8	28	Male	186.4894	52.38975	AB	15.06392	98.22778787	85	123	128	No	No
10	9	21	Male	155.0397	42.9587	B		98.80805323		111	243	No	No
11	10	32		170.8363	50.78325	B	17.40043	98.57016848	61	94	166		No
12	11	28	Female	152.9739	73.57281	B	31.44	98.37184285	60		241	No	Yes
13	12	34	Female	182.4163	76.37105	AB	22.95099	98.11827412	86	97	247	No	No
14	13	29	Male	168.846	48.53356	AB		98.5159267	62		246		Yes
15	14	34	Male		60.88223	B	22.5441	98.96356906	89	130	243	Yes	
16	15	33	Male	184.719	93.66694		27.45132	98.41821332	68	133	180	Yes	Yes
17		21	Male	162.8197	96.3857	B		98.10856314	68	126	130	No	No
18	17	24	Male	162.727	86.22873	B	32.56362	98.05782165	63	135	197		No
19	18	27		176.6947	56.24356	O	18.01464		78		161	No	No
20	19	31	Female	158.7902	46.82985	AB	18.57272	98.78470852	92	102	172		No
21	20	31	Male	166.4899	49.95557	B	18.02221	98.80974999	82	96	223	No	No

Before pre processing



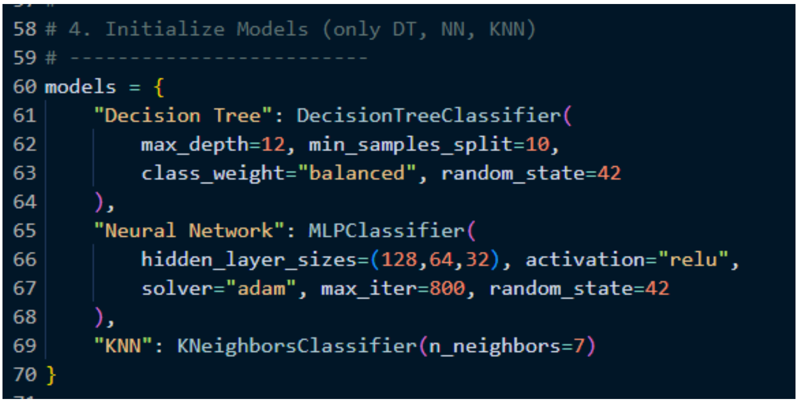
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Student ID	Age	Gender	Height	Weight	BMI	Temperature	Heart Rate	Blood Pressure	Cholesterol	Diabetes	Smoking	BloodType_A	BloodType_AB	BloodType_B	BloodType_O
2	1	-1.72895	0	-0.96082	0.14503	0.645471	2.99E-14	1.415376165	-0.406759296	0.519576957	0	0	0	0	0	1
3	2	0	1	-1.66918	-1.35945	0	0.240140011	1.23270259	-0.772679462	-0.603005071	0	0	0	0	1	0
4	3	1.288582	0	0.553808	-0.86594	-0.9906	-0.717404808	-0.32002279	1.130105401	0.884416116	1	0	1	0	0	0
5	4	0.857505	1	0.522815	-0.40401	-0.63586	0.502601273	1.780723313	-0.187207196	-1.220425186	0	1	0	0	1	0
6	5	-0.65126	0	-4.15E-15	-1.44444	0	-0.254694907	1.415376165	0	1.305384377	0	0	0	0	0	1
7	6	1.288582	1	-1.71134	-0.08055	0.985154	2.247951834	-0.868043513	0.983737334	-0.041714057	0	1	0	0	1	0
8	7	-1.08234	1	-0.14573	-1.33074	-1.08762	-1.864781864	-1.233390661	1.422841533	1.754417188	0	0	0	1	0	0
9	8	0.426429	1	0.842127	-1.06987	-1.24014	-0.785859269	0.502008294	0.617817168	-1.585264345	0	0	0	1	0	0
10	9	-1.08234	1	-1.45245	-1.64376	0	0.436155166	0	-0.260391229	1.642158985	0	0	0	0	1	0
11	10	1.288582	1	-0.29992	-1.16763	-0.88997	-0.06482012	-1.690074596	-1.504519793	-0.518811419	0	0	0	0	1	0
12	11	0.426429	0	-1.60317	0.219138	1.21409	-0.48248554	-1.781411384	0	1.586029884	0	1	0	0	1	0
13	12	1.719658	0	0.544954	0.389414	-0.05813	-1.016490634	0.593345081	-1.284967694	1.754417188	0	0	0	1	0	0
14	13	0.641967	1	-0.44514	-1.30452	0	-0.179051029	-1.598737809	0	1.726352637	0	1	0	1	0	0
15	14	1.719658	1	-4.15E-15	-0.5531	-0.11911	0.76366495	0.867355442	1.130105401	1.642158985	1	0	0	0	1	0
16	15	1.50412	1	0.712958	1.441886	0.61632	-0.3848313	-1.050717087	1.3496575	-0.125907709	1	1	0	0	1	0
17	16	-1.08234	1	-0.88482	1.607324	0	-1.03694155	-1.050717087	0.837369268	-1.529135244	0	0	0	0	1	0
18	17	-0.43572	1	-0.89158	0.989263	1.382483	-1.143800987	-1.507401022	1.496025566	0.351189653	0	0	0	0	1	0
19	18	0.210891	1	0.127508	-0.83536	-0.79792	2.99E-14	-0.137349216	0	-0.659134172	0	0	0	0	0	1
20	19	1.073044	0	-1.17881	-1.4082	-0.71429	0.386992184	1.141365803	-0.919047528	-0.350424114	0	0	0	1	0	0
21	20	1.073044	1	-0.61704	-1.21799	-0.79679	0.439728466	0.227997932	-1.358151727	1.080867971	0	0	0	0	1	0

After pre processing with proper scaling

Model Training

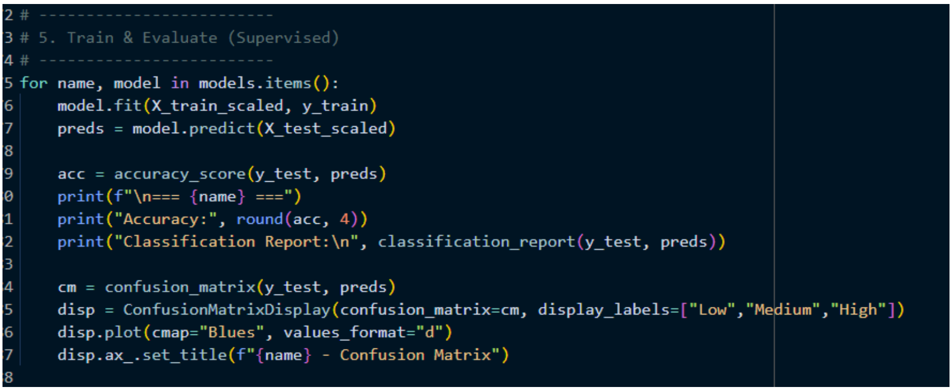
Three supervised models and one unsupervised model were applied:

- **Supervised:**
  - KNN (k=7) → Predicts BP_Class based on nearest neighbors.
  - Decision Tree (max_depth=12, balanced class weight) → Provides interpretability of features affecting BP.
  - Neural Network (128-64-32 layers, ReLU, Adam optimizer) → Captures complex nonlinear relationships.



```
# 4. Initialize Models (only DT, NN, KNN)
# ------------------------
models = {
    "Decision Tree": DecisionTreeClassifier(
        max_depth=12, min_samples_split=10,
        class_weight="balanced", random_state=42
    ),
    "Neural Network": MLPClassifier(
        hidden_layer_sizes=(128,64,32), activation="relu",
        solver="adam", max_iter=800, random_state=42
    ),
    "KNN": KNeighborsClassifier(n_neighbors=7)
}
```

Initializing Model (DT, NN, KNN)

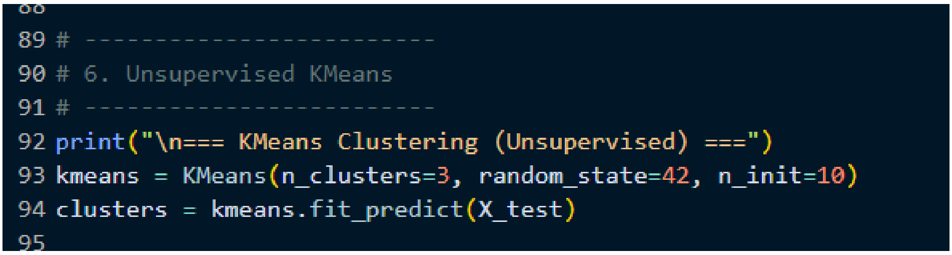


```
# ------------------------
# 5. Train & Evaluate (Supervised)
# ------------------------
for name, model in models.items():
    model.fit(X_train_scaled, y_train)
    preds = model.predict(X_test_scaled)

    acc = accuracy_score(y_test, preds)
    print(f"\n=== {name} ===")
    print("Accuracy:", round(acc, 4))
    print("Classification Report:\n", classification_report(y_test, preds))

    cm = confusion_matrix(y_test, preds)
    disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=["Low","Medium","High"])
    disp.plot(cmap="Blues", values_format="d")
    disp.ax_.set_title(f"{name} - Confusion Matrix")
```

Training Loop

- **Unsupervised:**

○ KMeans Clustering (k=3) → Groups students into 3 clusters, compared to true BP_Class.

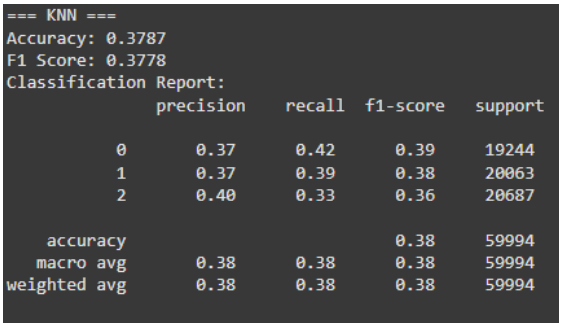


```
# -------------------------
# 6. Unsupervised KMeans
# -------------------------
print("\n=== KMeans Clustering (Unsupervised) ===")
kmeans = KMeans(n_clusters=3, random_state=42, n_init=10)
clusters = kmeans.fit_predict(X_test)
```

Model Evaluation

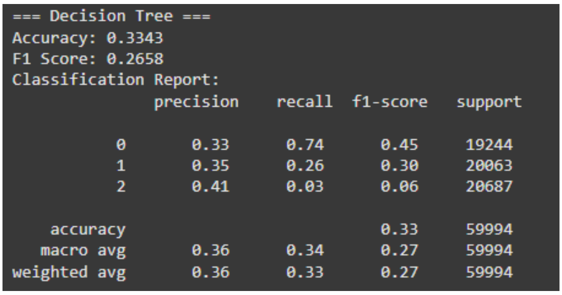
Supervised Results:

- KNN → Accuracy: ~37.9%, balanced across all classes.



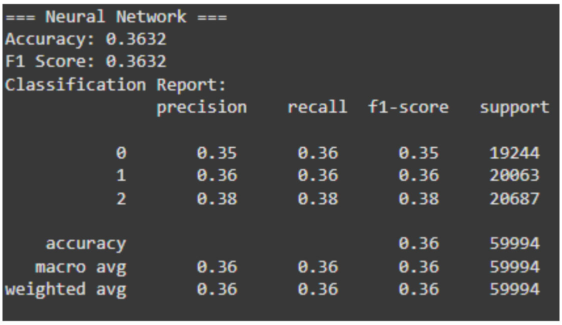
```
=== KNN ===
Accuracy: 0.3787
F1 Score: 0.3778
Classification Report:
              precision    recall  f1-score   support

           0       0.37      0.42      0.39     19244
           1       0.37      0.39      0.38     20063
           2       0.40      0.33      0.36     20687

    accuracy                           0.38     59994
   macro avg       0.38      0.38      0.38     59994
weighted avg       0.38      0.38      0.38     59994
```

●Decision Tree → Accuracy: ~33.4%, biased towards one class but interpretable.



```
=== Decision Tree ===
Accuracy: 0.3343
F1 Score: 0.2658
Classification Report:
              precision    recall  f1-score   support

           0       0.33      0.74      0.45     19244
           1       0.35      0.26      0.30     20063
           2       0.41      0.03      0.06     20687

    accuracy                           0.33     59994
   macro avg       0.36      0.34      0.27     59994
weighted avg       0.36      0.33      0.27     59994
```

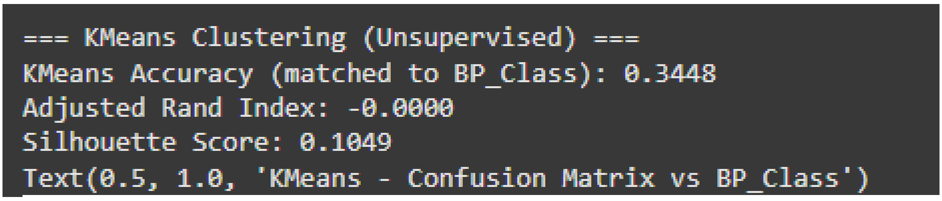
- Neural Network → Accuracy: ~36.3%, balanced recall and precision.



```
=== Neural Network ===
Accuracy: 0.3632
F1 Score: 0.3632
Classification Report:
              precision    recall  f1-score   support

           0       0.35      0.36      0.35     19244
           1       0.36      0.36      0.36     20063
           2       0.38      0.38      0.38     20687

    accuracy                           0.36     59994
   macro avg       0.36      0.36      0.36     59994
weighted avg       0.36      0.36      0.36     59994
```

Unsupervised Results:

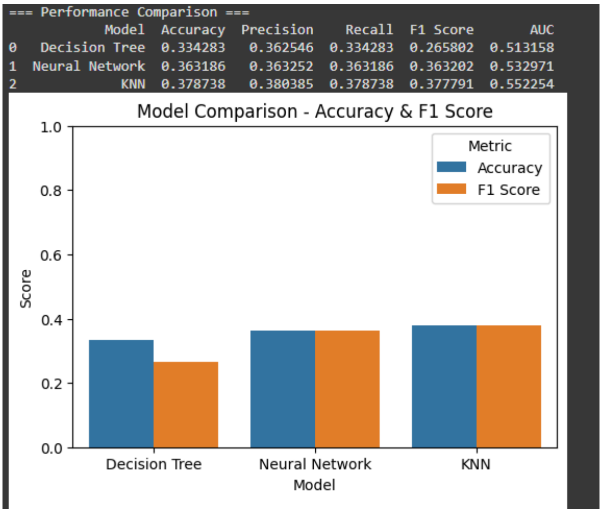
- KMeans Clustering → Accuracy (matched clusters to labels): ~34.5%
  - Adjusted Rand Index (ARI): ~0.0000 → indicates random similarity.
  - Silhouette Score: ~0.1049 → weakly separated clusters.



```
=== KMeans Clustering (Unsupervised) ===
KMeans Accuracy (matched to BP_Class): 0.3448
Adjusted Rand Index: -0.0000
Silhouette Score: 0.1049
Text(0.5, 1.0, 'KMeans - Confusion Matrix vs BP_Class')
```

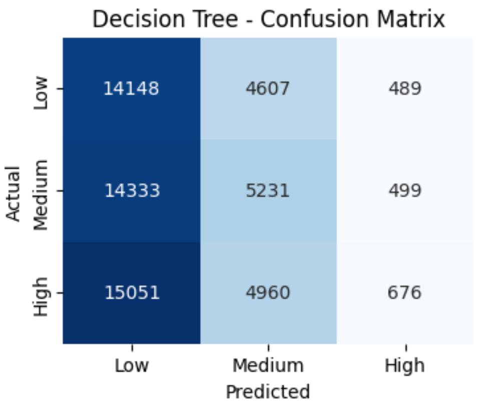
Performance Visualization:

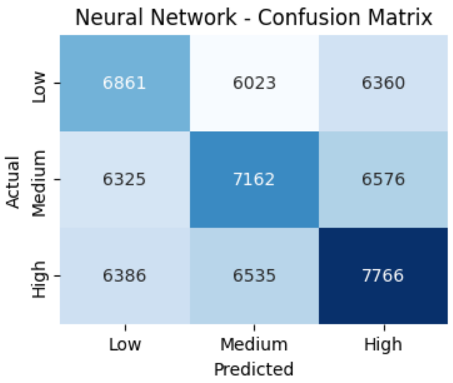
- Bar Chart: Compared accuracy & F1-score of supervised models.

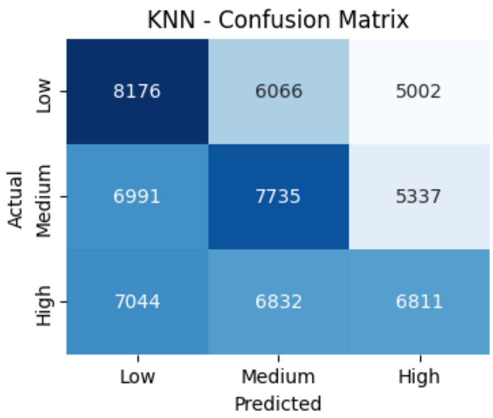


```
=== Performance Comparison ===
           Model  Accuracy  Precision    Recall  F1 Score       AUC
0  Decision Tree  0.334283   0.362546  0.334283  0.265802  0.513158
1 Neural Network  0.363186   0.363252  0.363186  0.363202  0.532971
2            KNN  0.378738   0.380385  0.378738  0.377791  0.552254
```

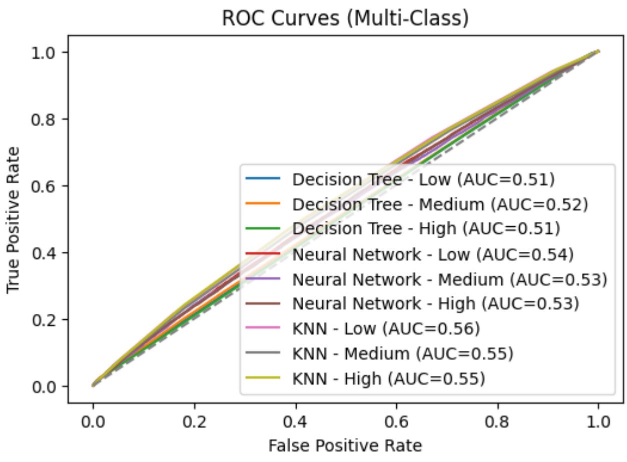
- Confusion Matrices: Showed that models often predicted one class strongly while confusing others.







- ROC & AUC: Very low separation, confirming difficulty in classifying BP levels.



Conclusion

The primary objective of this project was to investigate whether blood pressure levels (Low, Medium, High) could be predicted from health and lifestyle factors such as BMI, cholesterol, smoking, gender, age, and blood type. To achieve this, we applied three supervised learning models (Decision Tree, Neural Network, KNN) and one unsupervised method (KMeans clustering) on a preprocessed dataset of medical students.

Model Selection Justification

- **Decision Tree:** Chosen for its interpretability and ability to highlight which features are most important for classification (e.g., BMI, cholesterol). It provides insights into feature interactions and threshold values, which are easy to explain in medical terms.

- **Neural Network:** A mandatory requirement in the project guidelines. It is suitable for capturing complex non-linear patterns in health data that may not be detected by simpler models.

- **KNN (K-Nearest Neighbors):** Selected as a baseline similarity-based model, since it works well in datasets where class labels may depend on closeness of feature patterns.

KMeans: Used as the unsupervised approach, required by the project. It allowed us to test whether blood pressure categories form natural groupings in the dataset without labeled supervision.

Observed Results

KNN achieved the highest accuracy **(≈37.9%)**, with relatively balanced performance across classes.

Neural Network achieved **≈36.3% accuracy**, performing consistently but failing to capture clear class boundaries.

Decision Tree achieved **≈33.4% accuracy**, showing strong recall for one class but weak generalization overall.

KMeans clustering accuracy was **≈34.5%**, with very low Adjusted Rand Index and Silhouette Score, indicating poor natural separability of blood pressure groups.

Reasons Behind Low Accuracy

Despite multiple approaches, the models struggled to achieve high accuracy. The following reasons explain the performance:

Weak feature correlation:
While BMI, cholesterol, and smoking do influence blood pressure, they are not sufficient on their own to classify students into distinct Low, Medium, and High BP groups. Blood pressure is a multifactorial trait influenced by diet, genetics, stress, exercise, and sleep patterns — none of which were captured in this dataset.

Artificial class creation (quantile split):
Blood pressure was divided into three equal groups using quantiles. However, medical thresholds for blood pressure (e.g., systolic <120 = normal, 120–139 = pre-hypertension, ≥140 = hypertension) are not evenly distributed. This mismatch introduced noise and caused overlap between classes.

Feature overlap:
Boxplots and correlation analysis showed that BMI, cholesterol, and smoking values overlap significantly across Low, Medium, and High BP groups. This overlap reduced the separability of classes, making it difficult for models to distinguish between them.

Imbalanced complexity in data:
Even though class counts were nearly balanced after splitting, the underlying complexity of relationships could not be captured by traditional models. Neural networks require more distinctive features to learn meaningful representations, which this dataset lacked.

Clustering limitations:
KMeans assumes that classes form spherical, well-separated clusters. However, blood pressure distribution in this dataset does not naturally cluster in such a manner, which explains the poor silhouette score (~0.10) and ARI (~0.00).

Final Remarks

Overall, the project demonstrated that while machine learning models can capture partial patterns, the available dataset is not rich enough to yield high predictive accuracy. Among the supervised models, KNN emerged as the best performer, followed by Neural Network, while Decision Tree provided interpretability despite its lower accuracy.

The findings suggest that blood pressure prediction cannot rely solely on BMI, cholesterol, and smoking habits, and would require inclusion of additional lifestyle and genetic variables. Furthermore, using medical thresholds for blood pressure classification rather than statistical quantiles could significantly improve results.

This study highlights the challenges of working with health data, the importance of careful target definition, and the limitations of certain models when feature information is weakly correlated with the target.