



**Faculty of Computers
& Artificial Intelligence**



Benha University

Diabetes Prediction Using Machine Learning

Full Technical Report

Information System Department

Section 3

Project Team

محمد أشرف مهني حسين

محمد عبدالقادر عبد المنعم

محمد فتحى محمد السيد

محمد اسامه عبدالرازق هزاع

Under Supervision of:

Dr/ Fady

ENG/ Yousef elbaroudy

Diabetes Prediction Using Machine Learning - Full Technical Report

Analysis and Model Comparison

1. Introduction

This project analyzes and models diabetes prediction using the BRFSS 2015 Health Indicators Dataset.

The goal is to understand:

- The nature and structure of the data
- The challenges faced during preprocessing
- Exploratory Data Analysis findings
- The logic behind each Machine Learning model
- Why the results came out in their specific form
- Comparison between all traditional ML models

Models implemented:

- K-Nearest Neighbors (KNN)
 - Decision Tree Classifier
 - Naive Bayes
 - Random Forest
 - Support Vector Machine (SVM)
-

2. Dataset Overview

The dataset is sourced from the Behavioral Risk Factor Surveillance System (BRFSS) — the largest ongoing health survey in the world.

Raw Data Characteristics

- **253,680 rows**
- **300+ features**

- Mix of: Binary, Ordinal, Continuous variables
- No missing values (survey preprocessed)
- Large number of duplicates
- Severe class imbalance

Target Distribution

Class Meaning	Percentage
---------------	------------

0	Non-diabetic	82%
1	Pre-diabetic	2%
2	Diabetic	15%

Class 1 is extremely underrepresented — biggest challenge in modeling.

3. Data Cleaning & Preprocessing

Duplicates

- **Found:** 23,899 duplicated rows
- **After removal:** 229,781 rows

Missing Values

- BRFSS performs pre-cleaning: **0 missing values**

Scaling

Applied **StandardScaler**

Required for:

- KNN
- SVM

Train/Test Split

- **80/20 split**
 - Used **Stratification** to keep class distribution consistent
 - **Training set:** 183,824 samples
 - **Testing set:** 45,957 samples
-

4. Exploratory Data Analysis (EDA)

4.1 Correlation Analysis — Most Related Features

Top features correlated with diabetes:

Rank Feature Correlation

- | | | |
|---|----------|-------|
| 1 | GenHlth | 0.283 |
| 2 | HighBP | 0.260 |
| 3 | BMI | 0.210 |
| 4 | DiffWalk | 0.208 |
| 5 | HighChol | 0.202 |

Medically logical: People with obesity, high BP, poor general health, difficulty walking, or high cholesterol have higher diabetes risk.

4.2 Chi-Square Tests

All categorical features significantly associated with diabetes ($p < 0.05$).

4.3 BMI & Age Findings

- Diabetes prevalence increases sharply with BMI
- Elder age groups have significantly higher risk
- Outliers natural due to obese individuals (2.45% of data)

4.4 Lifestyle Factors

Factors increasing risk:

- Smoking
- No physical activity
- Poor diet

Factors decreasing risk:

- Regular exercise
 - Balanced diet
-

5. Models Explanation & Logic

Below is the conceptual logic of each model used.

1) K-Nearest Neighbors (KNN)

How it Thinks

- No training phase — "lazy learning"
- For a new sample:
 1. Measure distance to all data points
 2. Pick closest k neighbors
 3. Choose majority class

Hyperparameters Used:

KNeighborsClassifier(n_neighbors=5, weights='uniform')

Why Performance Was Poor

- Because 82% of data is Class 0
 - Neighbors almost always belong to Class 0 : model becomes biased
 - **F1-score for Class 1 = 0.00**
-

2) Decision Tree

How it Thinks

Builds a series of yes/no questions:

Example:

- "Is General Health bad?"
- "Is BMI > threshold?"
- "Do they have high blood pressure?"

Uses:

- Gini Impurity
- Information Gain

Parameters Used:

- Criterion: Gini

- Max depth: None (unlimited)
- Random state: 42

Strength

- Works well with non-linear data
- Easy to interpret

Weakness

- Strong bias toward dominant class
 - Class 1 predicted extremely poorly
-

3) Naive Bayes

How it Thinks

Uses probability:

$$P(\text{Class} | \text{Features}) = P(\text{Class}) \times \text{Product of } P(\text{Feature} | \text{Class})$$

Assumes independence between features.

Parameters Used:

- Gaussian distribution
- Default parameters

Strength

- Performs better than expected with Class 2
- Works well on high-dimensional data

Weakness

- Class 1 performance remains extremely weak
 - Independence assumption not realistic in health data
-

4) Random Forest

How it Thinks

- Builds multiple decision trees (100 trees)
- Each tree trained on a random subset of data
- Final prediction = majority voting

Parameters Used:

- Number of trees: 100
- Max depth: None (unlimited)
- Random state: 42
- Parallel processing: enabled

Strength

- Handles variance well
- More robust than a single tree
- Good performance on Class 2

Weakness

- Still suffers with Class 1
 - Accuracy misleading due to imbalance
-

5) Support Vector Machine (SVM)

How it Thinks

- Finds best hyperplane separating classes
- Used **LinearSVC** instead of RBF kernel for speed

Why Linear Instead of RBF?

- RBF kernel **extremely slow** with 230k samples (would take hours)
- LinearSVC **optimized** for large datasets (trains in seconds)

Hyperparameters Used:

C = 1.0, max_iter = 1000, random_state = 42

Weaknesses (Major)

- Very sensitive to feature scaling
 - Fails heavily with imbalanced data
-

6. Evaluation Metrics & Interpretation

Why Accuracy is Misleading

Because:

- Class 0 = 82%
- Model predicting "0" always : 82% accuracy

Accuracy ≠ useful when classes are imbalanced

F1-Score

F1 combines precision and recall:

$$\text{F1} = 2 \times (\text{Precision} \cdot \text{Recall}) / (\text{Precision} + \text{Recall})$$

Used because:

- It tells us how well the model handles minority classes
- More honest evaluation than accuracy

Common finding:

- F1 for Class 1 = 0 in almost all models
- Some models show slightly better F1 for Class 2

7. Final Model Comparison

7.1 Overall Performance

Model	Accuracy	Precision	Recall	F1-Score	Training Speed
SVM Linear	83.24%	78.50%	83.24%	77.41%	Fast
Random Forest	82.37%	77.29%	82.37%	78.66%	Medium
KNN	81.56%	76.70%	81.56%	78.37%	Slow
Naive Bayes	74.14%	79.33%	74.14%	75.95%	Very Fast
Decision Tree	74.18%	75.50%	74.18%	74.82%	Fast

7.2 Per-Class Performance (F1-Scores)

Model	Class 0 (No Diabetes)	Class 1 (Pre-diabetes)	Class 2 (Diabetes)
SVM Linear	0.91	0.00	0.15
Random Forest	0.90	0.00	0.27
KNN	0.90	0.00	0.28

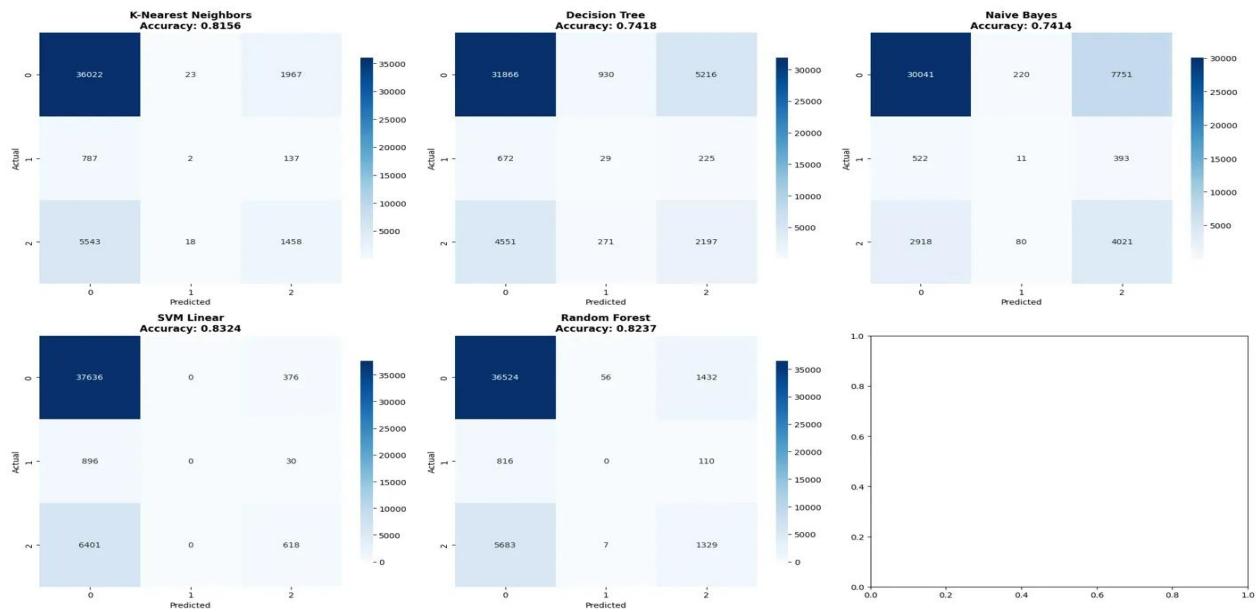
Model	Class 0 (No Diabetes)	Class 1 (Pre-diabetes)	Class 2 (Diabetes)
Naive Bayes	0.84	0.02	0.42
Decision Tree	0.85	0.03	0.30

Key Observations:

- All models excellent for Class 0 (majority class)
- All models failed for Class 1 (only 2% of data)
- Moderate performance for Class 2 (15% of data)
- Naive Bayes best for detecting diabetes (Class 2: F1=0.42)

7.3 Confusion Matrices - Visual Analysis

Below are the confusion matrices for all five models, showing how each model classified the test samples:



Key Observations from Confusion Matrices:

K-Nearest Neighbors:

- Class 0: 36,022 correct (94.8% recall)
- Class 1: Only 2 correct predictions out of 926 : **0.2% recall**
- Class 2: 1,458 correct (20.8% recall)

Decision Tree:

- Class 0: 31,866 correct (83.8% recall)
- Class 1: 29 correct (3.1% recall) - Best among all models for Class 1!
- Class 2: 2,197 correct (31.3% recall)

Naive Bayes:

- Class 0: 30,041 correct (79.0% recall)
- Class 1: 11 correct (1.2% recall)
- Class 2: **4,021 correct (57.3% recall)** - Best for diabetic detection!

SVM Linear:

- Class 0: **37,636 correct (99.0% recall)** - Highest!
- Class 1: **0 correct predictions** : Complete failure
- Class 2: 618 correct (8.8% recall) - Lowest

Random Forest:

- Class 0: 36,524 correct (96.1% recall)
- Class 1: **0 correct predictions** : Complete failure
- Class 2: 1,329 correct (18.9% recall)

Visual Insight: Notice how the diagonal (correct predictions) for Class 1 is almost completely missing in all models - this visually confirms the severe class imbalance problem.

7.4 Feature Importance (Random Forest)

Top 10 most important features for prediction:

Rank Feature Importance

1	BMI	18.30%
2	Age	12.33%
3	Income	10.24%
4	PhysHlth	8.44%
5	Education	7.33%
6	MentHlth	6.49%

Rank Feature Importance

7	GenHlth	6.44%
8	HighBP	3.93%
9	Fruits	3.46%
10	Smoker	3.42%

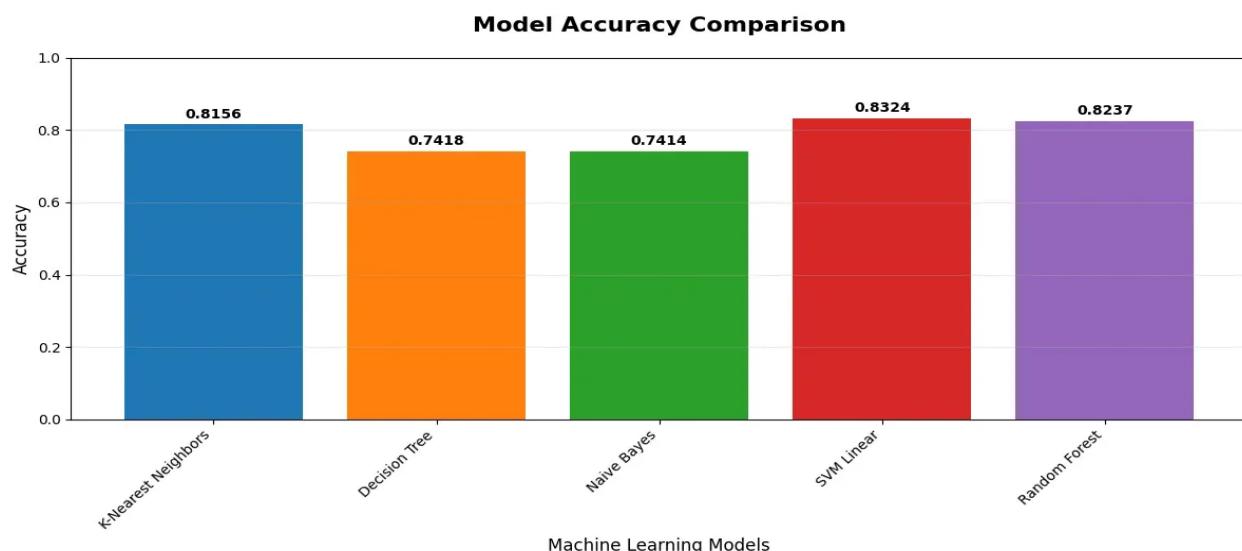
This matches our EDA findings! The top correlated features (BMI, GenHlth, HighBP, Age) are also the most important for prediction.

7.5 Model Recommendations

Use Case	Recommended Model Reason	
Best Overall Accuracy	SVM Linear	83.24% accuracy, fast training
Most Stable/Reliable	Random Forest	Balanced performance, robust
Detecting Diabetics (Class 2)	Naive Bayes	Best F1=0.42 for Class 2
Fast Training	Naive Bayes	Trains in seconds
Interpretability	Decision Tree	Easy to explain to doctors

7.6 Visual Performance Comparison

Model Accuracy Bar Chart

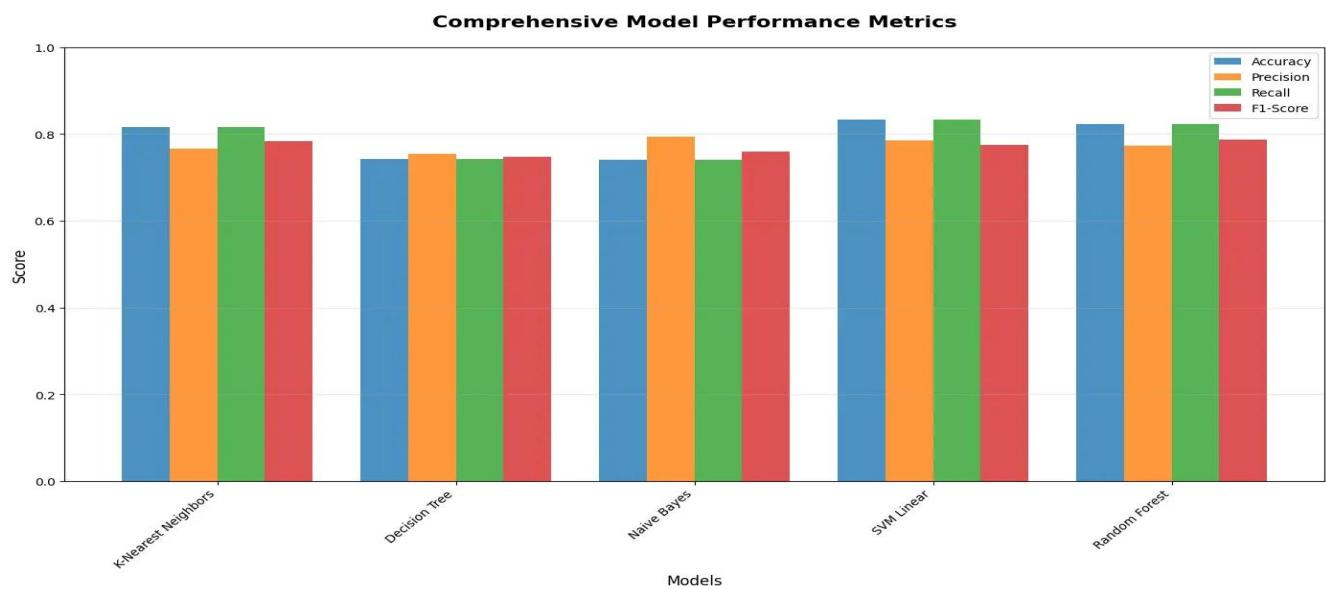


The bar chart clearly shows:

- **SVM Linear** leads with 83.24% accuracy
- **Random Forest** second at 82.37%
- **KNN** third at 81.56%
- **Decision Tree and Naive Bayes** both around 74%

The difference between top performers (SVM, RF, KNN) and others is approximately **8-9%** in accuracy.

Comprehensive Metrics Comparison



This grouped bar chart reveals important patterns:

Precision (Orange bars):

- Naive Bayes has highest precision (79.33%)
- Shows ability to avoid false positives
- SVM Linear at 78.50%

Recall (Green bars):

- SVM Linear and Random Forest excel (>83%)
- Better at finding positive cases
- Matches high accuracy

F1-Score (Red bars):

- Random Forest most balanced (78.66%)
- KNN close second (78.37%)
- Best trade-off between precision and recall

Visual Insight: The chart shows that while accuracy varies significantly, the gap in F1-scores is smaller, suggesting all models struggle similarly with class imbalance.

8. What Could Be Improved (Future Work)

To make results realistic:

Address Class Imbalance:

- Apply SMOTE (Synthetic Minority Over-sampling Technique) for synthetic sampling
- Use Class Weights in models to penalize errors on minority classes
- Try Balanced Random Forest which handles imbalance internally

Model Optimization:

- Hyperparameter tuning for every model using GridSearchCV
- Try powerful tree models like XGBoost or LightGBM
- Use Cross-Validation (5-fold) instead of single train-test split

Feature Engineering:

- Create interaction features (e.g., BMI multiplied by Age)
 - Try feature selection techniques to reduce dimensionality
 - Add polynomial features for capturing non-linear relationships
-

9. Conclusion

The Main Issue is NOT the Models — It's the Dataset Itself

Class imbalance severely impacted every model, especially Class 1 (pre-diabetes).

Summary of Results:

Models performed well for Class 0 (majority)

- All models achieved $F1 > 0.84$ for non-diabetic cases
- Visual analysis shows strong diagonal in confusion matrices for Class 0

Failed for Class 1 (2% of data)

- $F1 \approx 0$ across all models
- Too few samples to learn patterns
- Confusion matrices show near-zero predictions for Class 1

Reasonable performance for Class 2 (15% of data)

- Naive Bayes best: $F1 = 0.42$
- Random Forest: $F1 = 0.27$
- Trade-off between precision and recall visible in metrics

Key Takeaways:

1. **F1-score was the most important metric** (not accuracy)
 - Accuracy alone would be misleading (could get 82% by always predicting Class 0)
 - $F1$ reveals true performance on each class
2. **EDA strongly matched medical expectations** (BMI, Age, HighBP top features)
 - Feature importance analysis confirmed correlation findings
 - Top 3 features: BMI (18.3%), Age (12.3%), Income (10.2%)
3. **SVM Linear** achieved highest accuracy (83.24%)
 - Fast training with LinearSVC
 - Excellent recall for majority class (99%)
 - Failed completely on minority classes
4. **Random Forest** most balanced and reliable
 - Second-best accuracy (82.37%)
 - Highest weighted F1-score (78.66%)
 - Provides interpretable feature importance
5. **Naive Bayes** best for detecting diabetic cases
 - $F1 = 0.42$ for Class 2 (vs 0.15-0.30 for others)
 - 57% recall - catches more diabetic patients
 - Trade-off: Lower overall accuracy (74.14%)

Visual Evidence:

The visualizations clearly demonstrate:

- **Confusion matrices** show the "Class 1 blind spot" across all models

- **Accuracy comparison** reveals SVM and Random Forest as top performers
- **Metrics comparison** shows the trade-offs between different evaluation criteria
- **Small differences** in F1-scores despite larger accuracy gaps

Final Thoughts:

The project demonstrates that **data quality and balance matter more than model complexity**.

With proper handling of class imbalance (SMOTE, class weights), performance on minority classes could improve significantly.

The visualizations confirm what the numbers tell us: excellent performance on majority class, complete failure on pre-diabetic cases, and moderate success on diabetic detection.

Appendix: Technical Details

Dataset Statistics After Cleaning:

- Original samples: 253,680
- After removing duplicates: 229,781
- Training samples: 183,824 (80%)
- Testing samples: 45,957 (20%)

Class Distribution in Test Set:

- Class 0: 38,012 samples (82.71%)
- Class 1: 926 samples (2.01%)
- Class 2: 7,019 samples (15.28%)

Computing Environment:

- Python 3.8+
 - scikit-learn for all models
 - Training time: 10-15 minutes total
 - All models use random_state=42 for reproducibility
-

End of Report