Predicting Diabetes Risk from Behavioral CDC Data: A Comparative Analysis of Logistic Regression, Decision Trees, and K-Nearest Neighbors

Alexandra Lostetter, Rachel Xing, Rohan Gogate, Yukai Li, Mairui Li

THE UNIVERSITY *of* NORTH CAROLINA *at* CHAPEL HILL

Table of Contents

Introduction

Background and Importance

- Leading Cause of Death Worldwide: 830 M people.
- 2015 Global Cost: $1.31 T.
- Early prediction (esp. Type II) enables prevention

Health Impact

- Insulin deficiency/resistance → high blood glucose
- Major complications: heart disease, vision loss, kidney failure
- Manageable with lifestyle changes + medication

U.S. Context

- 2018: 34.2 M diabetics, 88 M pre-diabetics
- Awareness gap: only 20% of each group know their status
- Type II most common; disproportionately affects low-income groups
- Annual economic burden > $327 B

Data Structure Overview

Size: 253,680 rows × 22 columns (original: 330 features, reduced based on chronic disease research)

Target Variable: `Diabetes_binary` (0 = No diabetes, 1 = Pre-diabetes/Diabetes)

Feature Types:

- *Continuous:* `BMI`, `MentHlth`, `PhysHlth`
- *Ordinal:* `GenHlth`, `Age`, `Education`, `Income`
- *Binary Predictor Features:* `HighBP`, `HighChol`, `CholCheck,Smoker,Stroke,HeartDiseaseorAttack,PhysActivity, Fruits,Veggies,HvyAlcoholConsump,AnyHealthcare, NoDocbcCost, DiffWalk, Sex`

Class Imbalance: 84.7% No diabetes, 15.3% Diabetes

Duplicates: 24,206 rows

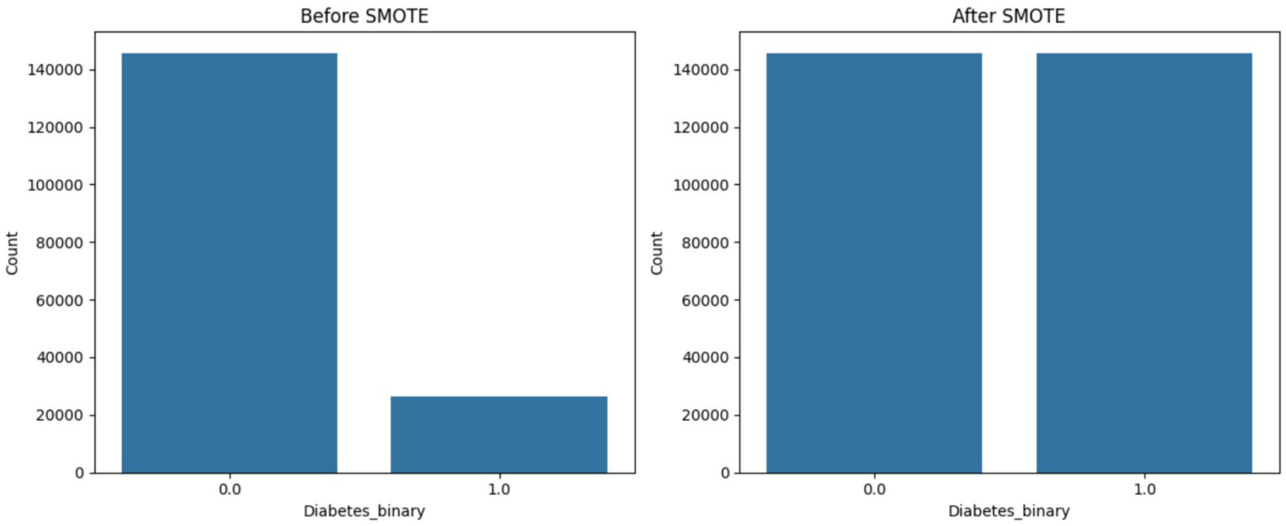
Source: Cleaned BRFSS dataset, preprocessed for ML readiness

Methodology

Data Preprocessing: Resampling / Handling Imbalanced Data

SMOTE (Synthetic Minority Over-sampling Technique)

- Generates **synthetic** examples for the minority class using interpolation.
- Only apply SMOTE on **training set** to avoid data leakage



Methodology

Transform Data and Feature Selection

No need for one-hot encoding: **no categorical data with no natural order/ranking).**

Dimensionality Reduction (manual)

Check multicollinearity, variables were grouped by type:

- **Binary × Cont./Ord. → Point-Biserial**
- **Cont. × Cont. → Pearson**
- **Ord. × Cont./Ord. → Spearman (or Kendall)**

Ranked Variable Pairs by Absolute Correlation:

	Variable 1	Variable 2	Correlation	P-Value	Method
0	GenHlth	PhysHlth	0.524364	0.000000e+00	Pearson
1	Education	Income	0.449106	0.000000e+00	Pearson
2	GenHlth	Income	-0.370014	0.000000e+00	Pearson
3	MentHlth	PhysHlth	0.353619	0.000000e+00	Pearson
4	HighBP	Age	0.344452	0.000000e+00	Point Biserial
..	...	...	...	...	...
97	CholCheck	Income	0.014259	6.870724e-13	Point Biserial
98	Veggies	Age	-0.009771	8.587520e-07	Point Biserial
99	CholCheck	MentHlth	-0.008366	2.514156e-05	Point Biserial
100	AnyHealthcare	PhysHlth	-0.008276	3.066384e-05	Point Biserial
101	CholCheck	Education	0.001510	4.467881e-01	Point Biserial

	Variable	VIF
0	const	7.291940
1	PhysHlth	1.379229
2	GenHlth	1.379229

GenHlth ↔ PhysHlth = 0.524

Education ↔ Income = 0.449

GenHlth ↔ Income = –0.370

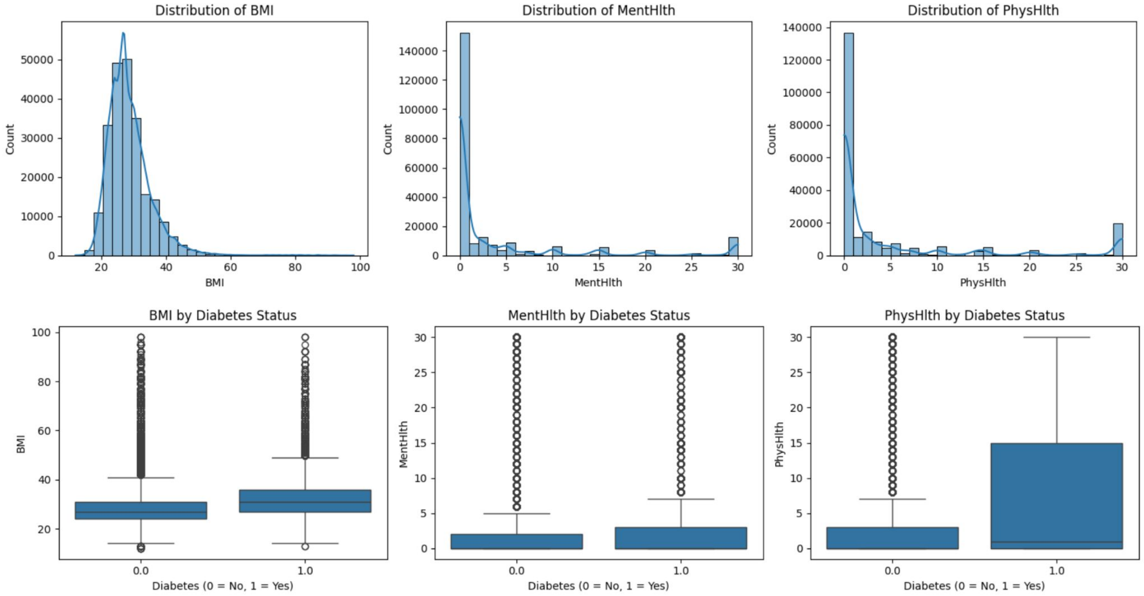
low VIF/ model simplicity — ended up keeping the features

Methodology

Exploratory Data Analysis (EDA)

Continuous Numerical Variables — Histogram and Boxplots

- Right-skewed Distribution
- Diabetes ⬆️ **higher median BMI**
- Diabetes ⬆️ **more poor-health days** (especially PhysHlth)

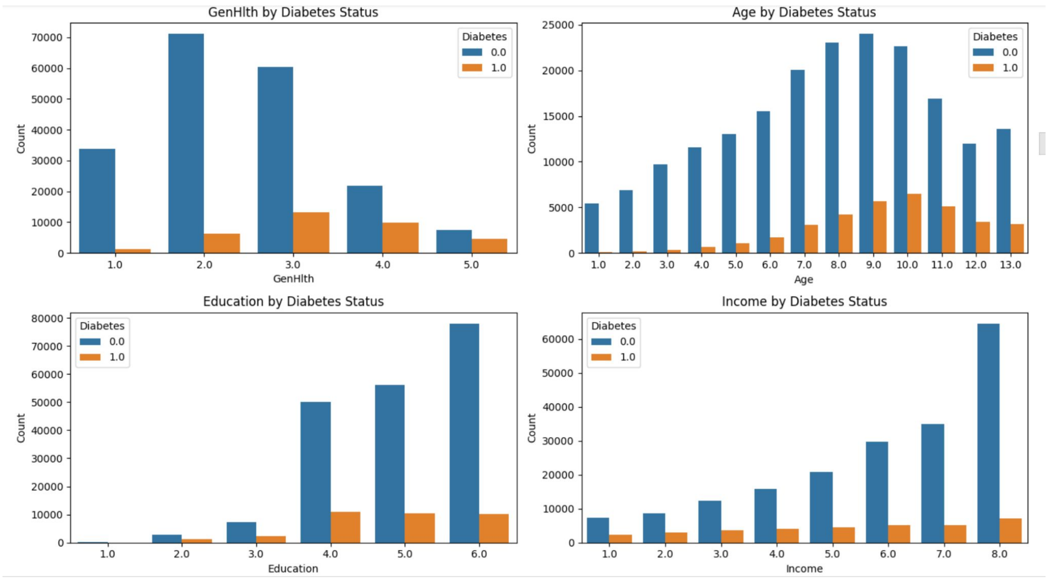


Methodology

Exploratory Data Analysis (EDA)

Ordinal Categorical Variables — Countplots

- Diabetes risk rises with **age**, **worse self-rated health**, **lower education / income**



Binary Categorical Variables —- Stacked Bar Plot

- Strong signals: high BP, high cholesterol, chol-check, stroke, heart disease, mobility limits, inactivity

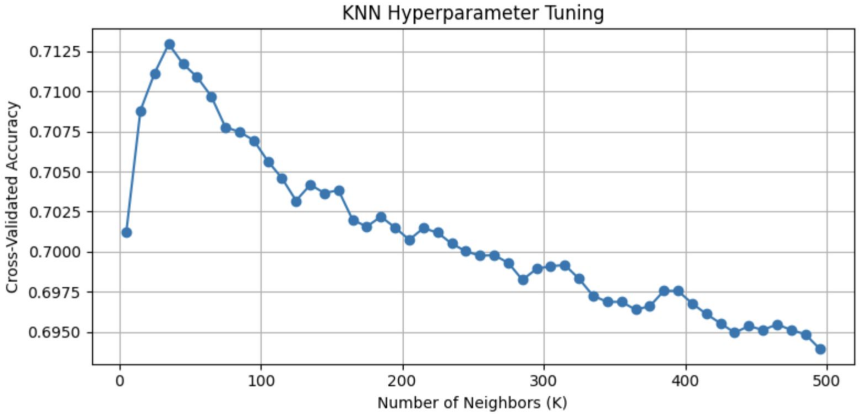
Methodology II: K Nearest Neighbors (KNN)

Selected optimal **K** by balancing the **bias-variance tradeoff**, using the heuristic **K ≈ √N** (N = # of rows) ≈ 500.

Tuned K via **GridSearchCV** with **2-fold cross-validation** and `algorithm='auto'`.

Overall Trend:

```
Best K: {'n_neighbors': 35}
Training Accuracy: 0.7298669227603237
Testing/Validation Accuracy: 0.6276734821942164
```

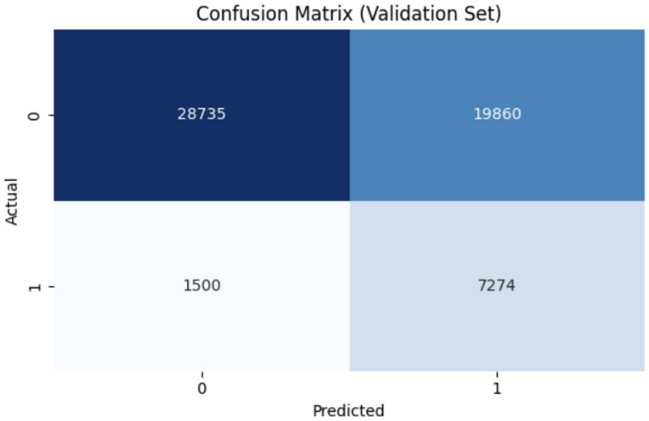


Found **K = 35** yielded the best performance:

- **Training Accuracy:** `0.730`
- **Validation Accuracy:** `0.628`

Used a **subset of `X_resampled`** to reduce computation time, approximating full KNN behavior.

Future improvements: **stratified subsampling (to preserve class distributions and reduce model bias towards majority class)**, **dimensionality reduction**, or **approximate nearest neighbors**.



KNN Findings

The model **excels at identifying diabetics** (high recall) but **falsely labels many healthy individuals** (low precision and specificity), affecting accuracy and trust.

Metric	Value
Recall / Sensitivity	82.9%
Precision	26.81%
Specificity	59.14%
F1 Score	40.51%

Recall / Sensitivity → Of all the people who actually ***do*** have diabetes, our model correctly identifies 82.9% of them.

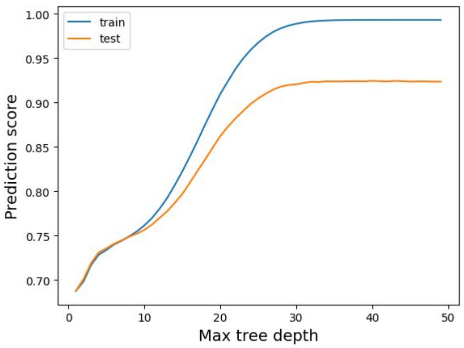
Precision → Of all the people our model says ***have*** diabetes, only 26.81% actually do.

Specificity → Of all the people who ***don't*** have diabetes, our model correctly identifies 59.14% of them.

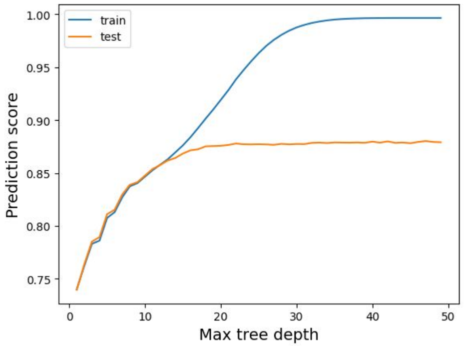
F1 Score → This F1 score conveys that while the model is catching a good portion of cases, the **low precision is dragging performance down overall.**

*This is using the resampled / balanced data after our group performed SMOTE resampling technique.

Oversampling



SMOTE



Methodology II: Decision Tree

```
[23] dt_best = DecisionTreeClassifier(max_depth=41)
     dt_best.fit(X_train,y_train)
     dt_best.score(X_test,y_test)
```

```
0.9236952389676415
```

```
dt_best = DecisionTreeClassifier(max_depth=23)
dt_best.fit(X_train,y_train)
dt_best.score(X_test,y_test)
```

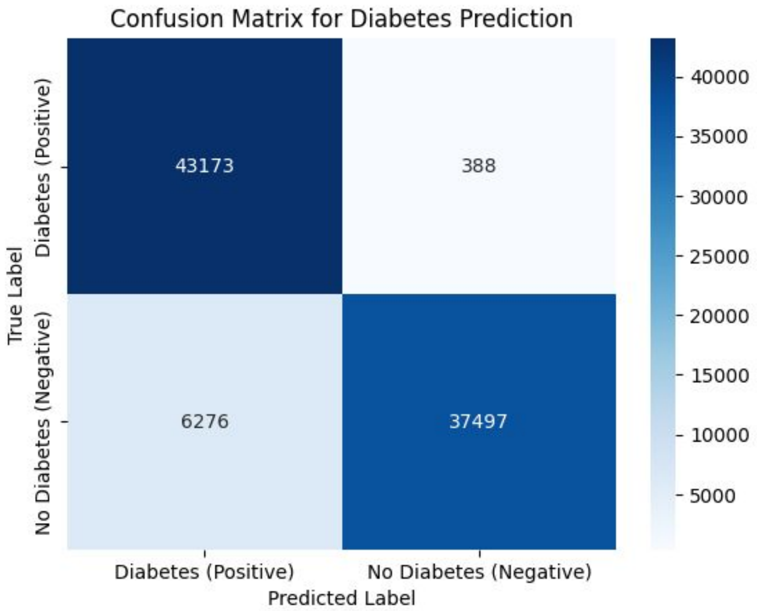
```
0.8768749856871322
```

Compared **Oversampling** vs **SMOTE**

Tuned **max_depth** via train-vs-test “elbow” plot

- **Oversampling:** best depth 41 → test score ≈ 0.924
- **SMOTE:** best depth 23 → test score ≈ 0.877

Oversampling outperforms → used for follow-up work.




```
True Positive Rate (TPR): 0.9910929501159293
False Positive Rate (FPR): 0.14337605373175244
False Negative Rate (FNR): 0.008907049884070614
Precision: 0.8730813565491718
```

Decision Tree Findings

We found that the Oversampling model performed better, but *why*?

Oversampling > SMOTE

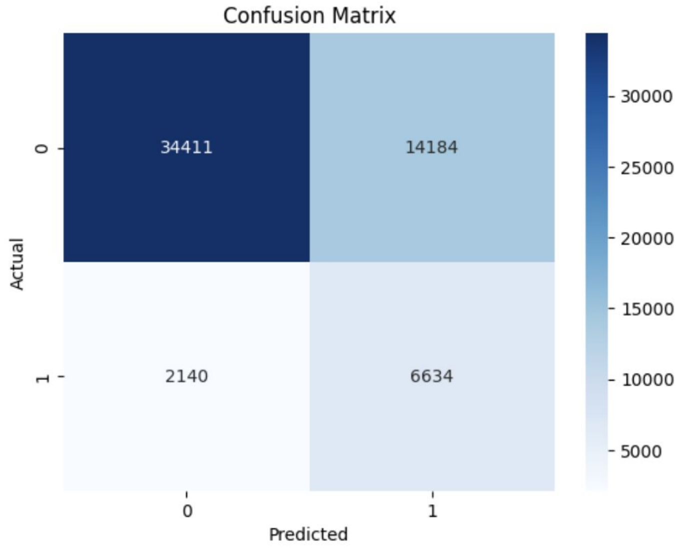
- CART favors real records; SMOTE's synthetic points hurt generalization.

Test metrics (Oversampling)

- **TPR 0.99** / **FNR 0.009** → almost no diabetics missed
- **Precision 0.87**, **FPR 0.14**

Low miss-rate is critical in a medical screening context.

Methodology II: Logistic Regression



Recall: 0.7560975609756098

Metric	Value
Training Score	**0.743**
Testing Score	**0.716**
Accuracy	**0.715**
Precision	**0.319**
Recall	**0.756**
Specificity	**0.708**
F1 Score	**0.448**

Good generalization

Train score 0.743 vs. test 0.716 ($\Delta \approx 0.03$) → minimal over-fit; model should transfer well to unseen patients.

Imbalanced performance

Precision 0.319 → 68 % of predicted diabetics are false alarms.

Recall 0.756 → model detects ~3 out of 4 true diabetics, limiting missed cases.

Moderate overall

F1 0.448 reflects the recall–precision gap; boosting precision (e.g., class weights, threshold tuning) would raise this score.

Findings and Conclusions

Best Model: Decision Tree with Random Oversampling — > 99% recall, 0.9% false negative rate; ideal for medical use.

KNN: High recall (82.9%) but very low precision (26.8%); prone to false positives.

Logistic Regression: Model performs reasonably well, identifying most positive cases (76% recall), but struggles with precision (only 32%), leading to a moderate F1 score (0.448) and indicating room for improvement in balancing accuracy and error.

Future Improvements:

- Combine with other diabetes datasets to enhance feature diversity.
- Introduce multiclass target (e.g., no diabetes, pre-diabetes, diabetes).
- Handle skewed variables (e.g., BMI) by filtering outliers.
- Apply better scaling techniques to improve KNN precision.
- Explore XGBoost for Decision Tree or Random Forest Classifiers

References

Al Jarullah, A. A. (2011). Decision tree discovery for the diagnosis of type II diabetes. *2011 International Conference on Innovations in Information Technology*, Abu Dhabi, United Arab Emirates, 303–307. https://doi.org/10.1109/INNOVATIONS.2011.5893838

Ali, A., Alrubei, M. A., Hassan, L. F., Al-Ja'afari, M. A., & Abdulwahed, S. H. (2020). Diabetes diagnosis based on KNN. *IIUM Engineering Journal, 21*(1), 175–181. https://doi.org/10.31436/iiumej.v21i1.1206

D, X. Z. O. J. (n.d.). Building risk prediction models for type 2 diabetes using Machine Learning Techniques. *Preventing Chronic Disease*. https://pubmed.ncbi.nlm.nih.gov/31538566/

Kumar, N. M. S., Eswari, T., Sampath, P., & Lavanya, S. (2015). Predictive methodology for diabetic data analysis in Big Data. *Procedia Computer Science, 50*, 203–208. https://doi.org/10.1016/j.procs.2015.04.069

Philip, N. Y., Razaak, M., Chang, J., M, S. M., O'Kane, M., & Pierscionek, B. K. (2022). A data analytics suite for exploratory, predictive, and visual analysis of type 2 diabetes. *IEEE Access, 10*, 13460–13471. https://doi.org/10.1109/ACCESS.2022.3146884

Rajendra, P., & Latifi, S. (2021). Prediction of diabetes using logistic regression and ensemble techniques. *Computer Methods and Programs in Biomedicine Update, 1*, 100032. https://doi.org/10.1016/j.cmpbup.2021.100032

Suriya, S., & Joanish Muthu, J. (2023). Type 2 diabetes prediction using K-nearest neighbor algorithm. *Journal of Trends in Computer Science and Smart Technology, 5*(2), 190–205. https://doi.org/10.36548/jtcsst.2023.2.007

Teboul, A. (2021, November 8). Diabetes health indicators dataset. *Kaggle*. https://www.kaggle.com/datasets/alexteboul/diabetes-health-indicators-dataset

Q&A

THE UNIVERSITY
of NORTH CAROLINA
at CHAPEL HILL