W8-L7: Machine Learning – Classification (Diabetes)

Student Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Course: DATA 3421

# 1) Data Preparation

I started by loading the diabetes dataset and fixing places where a zero isn’t realistic for a living person (Glucose, BloodPressure, SkinThickness, Insulin, BMI). I treated those zeros as missing and used median imputation later. To give the model more signal, I created three simple features: a Glucose-to-Insulin ratio, an Age×BMI interaction, and Pregnancies per Age. I also added missing‑value indicators for the clinical fields so the model can learn whether “was missing” itself carries information.

A quick look at the key distributions:

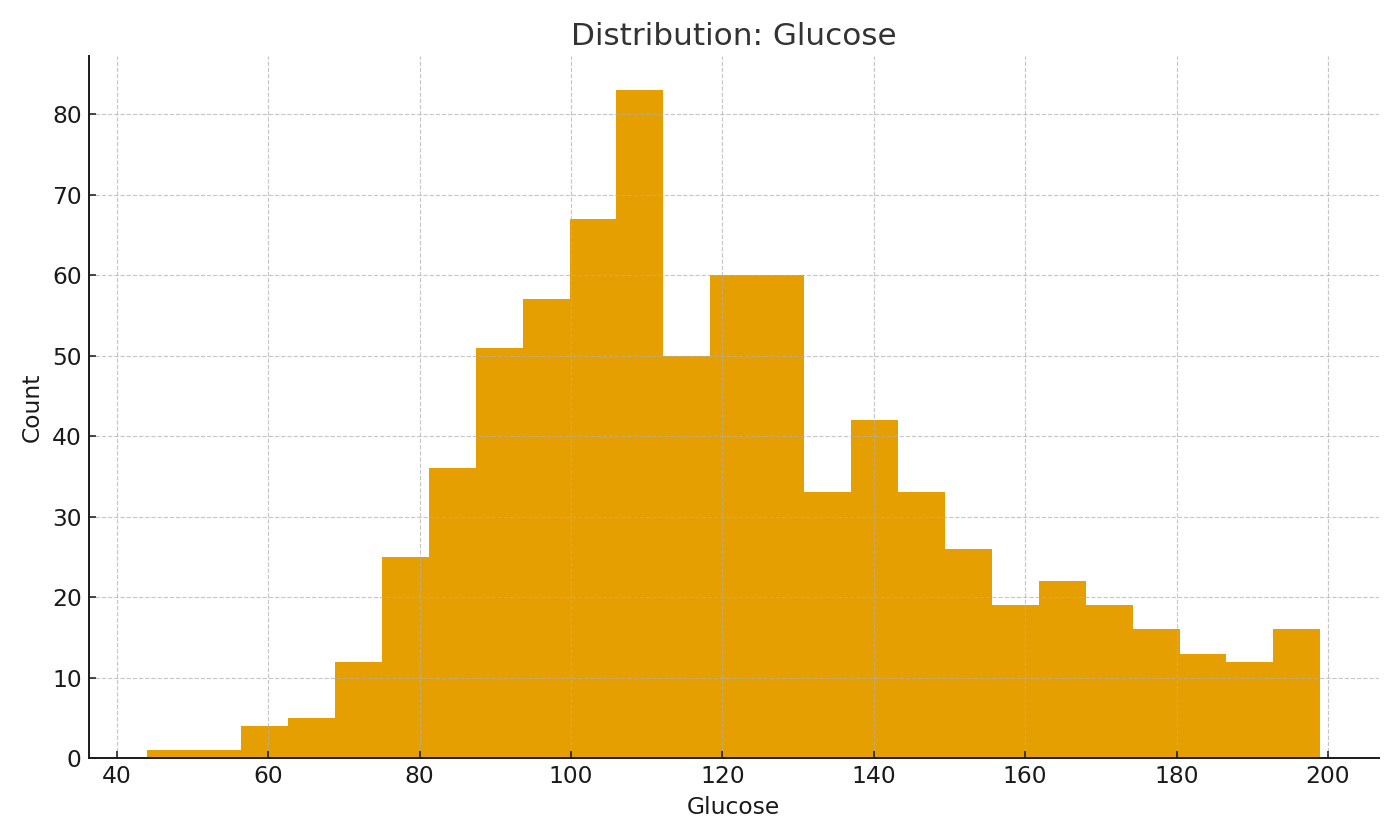


Figure: Distribution of Glucose

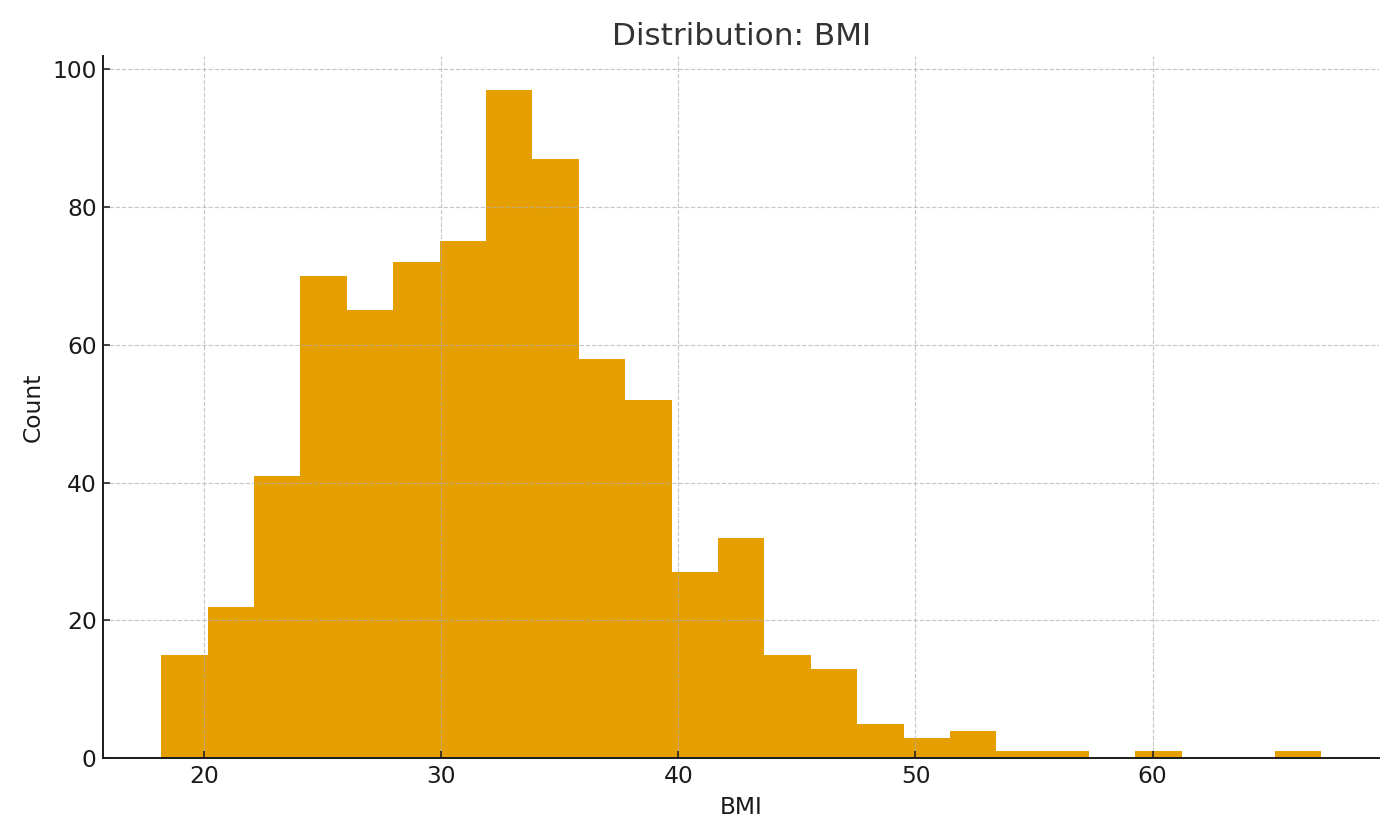


Figure: Distribution of BMI

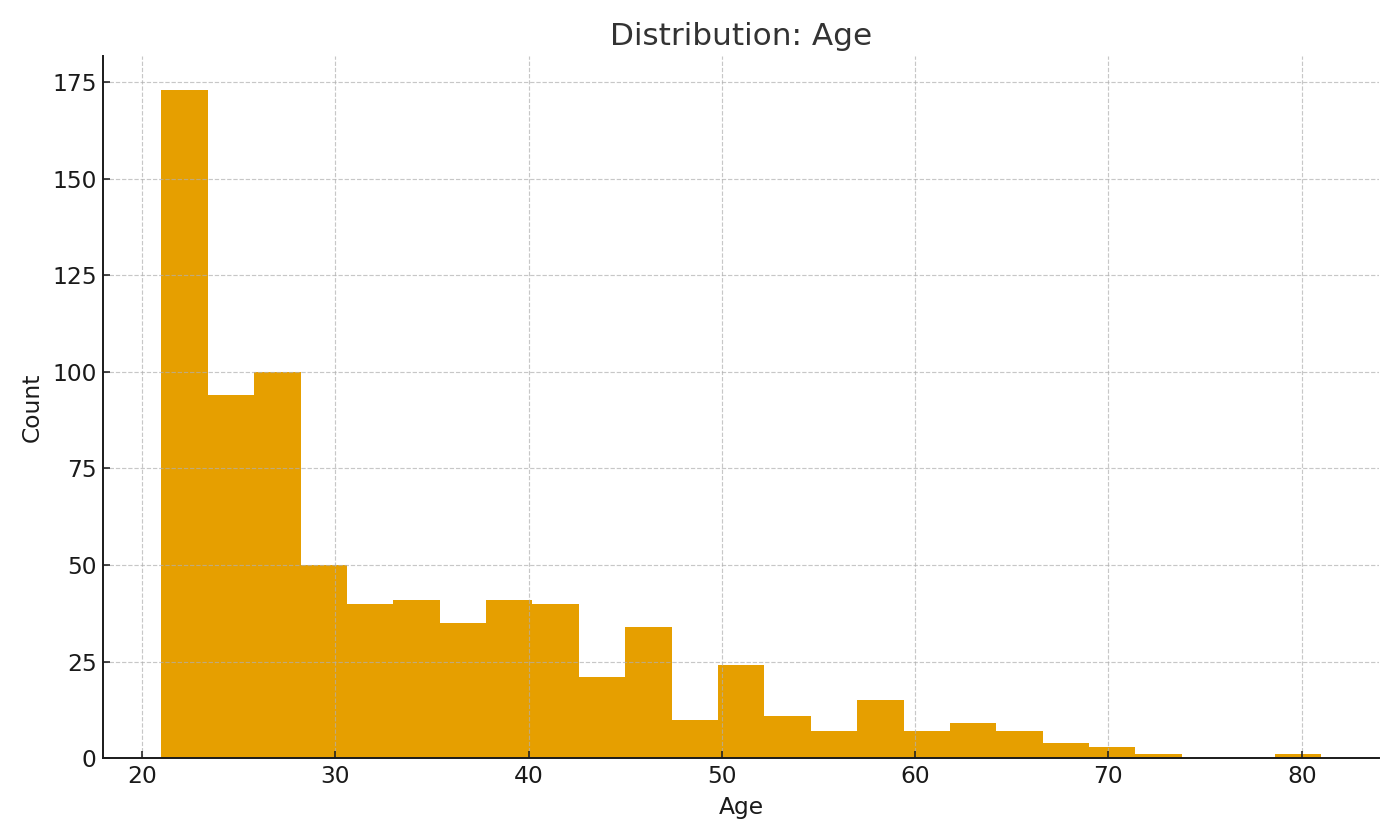


Figure: Distribution of Age

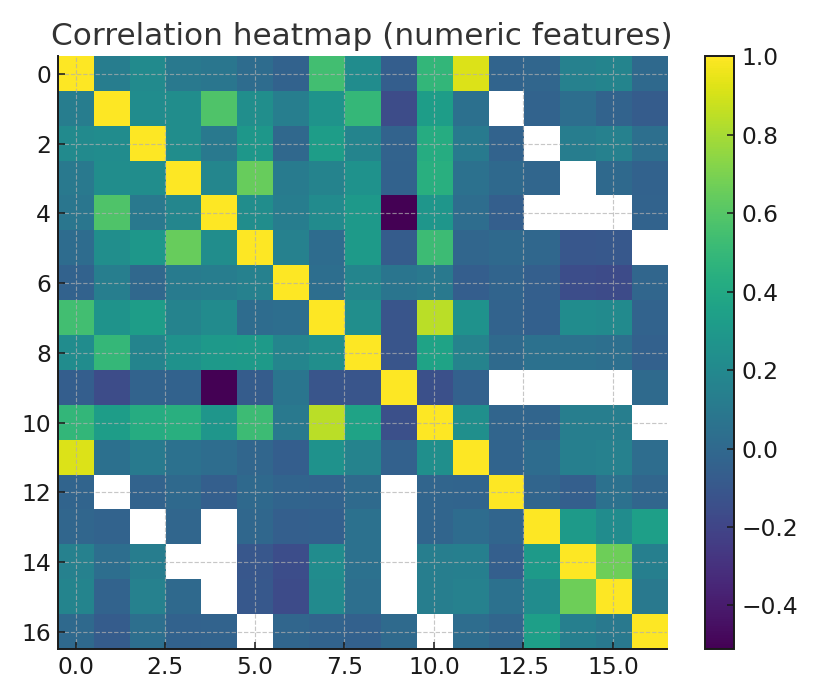


Figure: Correlation heatmap (numeric features).

Train/Validation/Test split: 70% / 15% / 15% with stratification (so the positive rate stays consistent across splits). Trees don’t need scaling; KNN does, because it uses distances. I kept preprocessing inside each model pipeline so the exact same steps happen at training and at prediction time.

# 2) Decision Tree (why & how)

Why a tree: fast, easy to interpret, and it handles non‑linear splits without scaling. I set a small max depth (5) with min\_samples\_split=10 and min\_samples\_leaf=5 to avoid overfitting. On the validation set, the tree reached solid recall with moderate precision.

# 3) Random Forest (why & how)

Why a forest: it’s a bag of trees, so it usually generalizes better than a single tree. I used 200 trees and class\_weight='balanced\_subsample' to be friendly to the minority class. In practice this model performed the best on validation AUC (see comparison below).

# 4) K‑Nearest Neighbors (KNN)

Why KNN: it’s a simple, classic baseline for tabular classification—but only after scaling. I used K=11. To show why scaling matters, I also ran the same KNN without scaling and compared AUC.

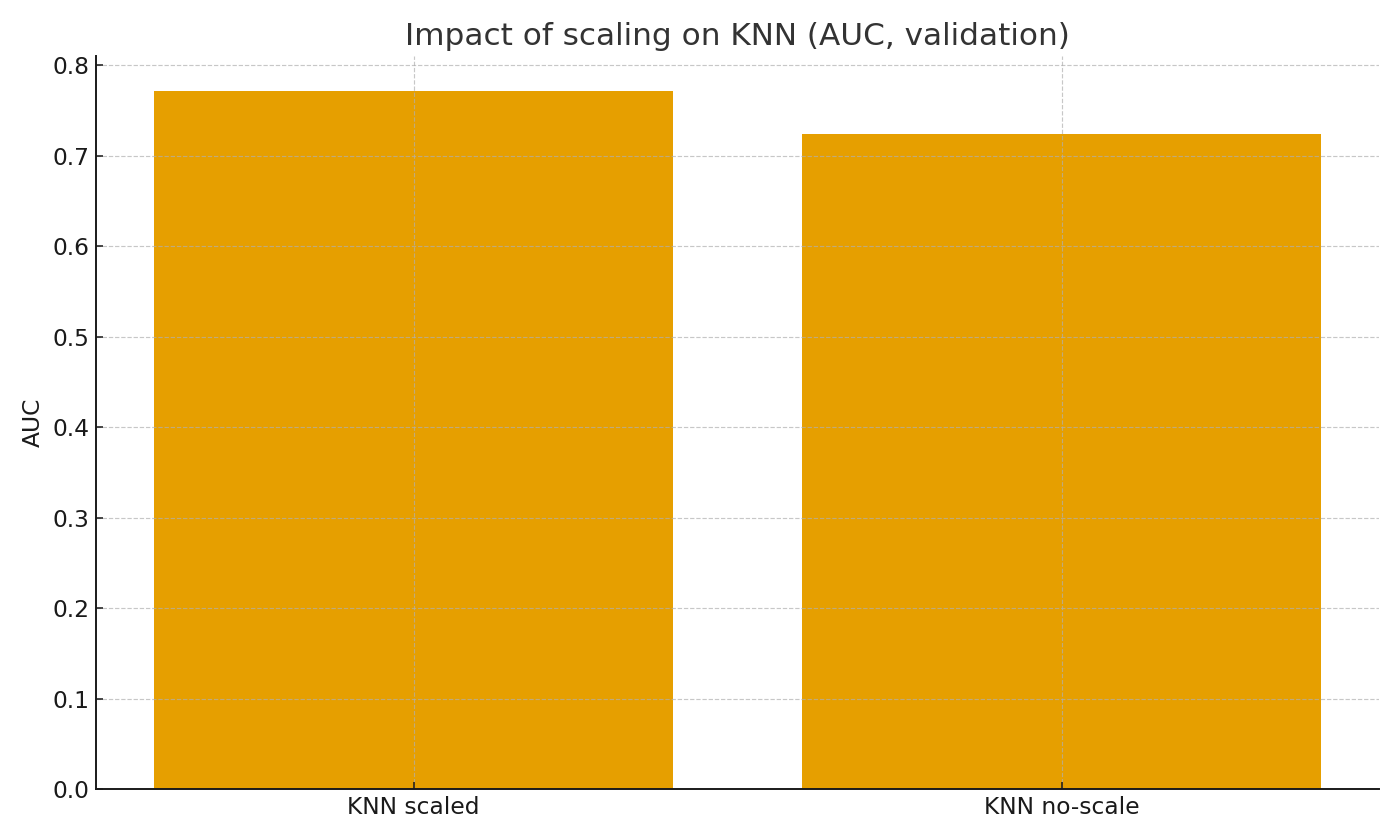
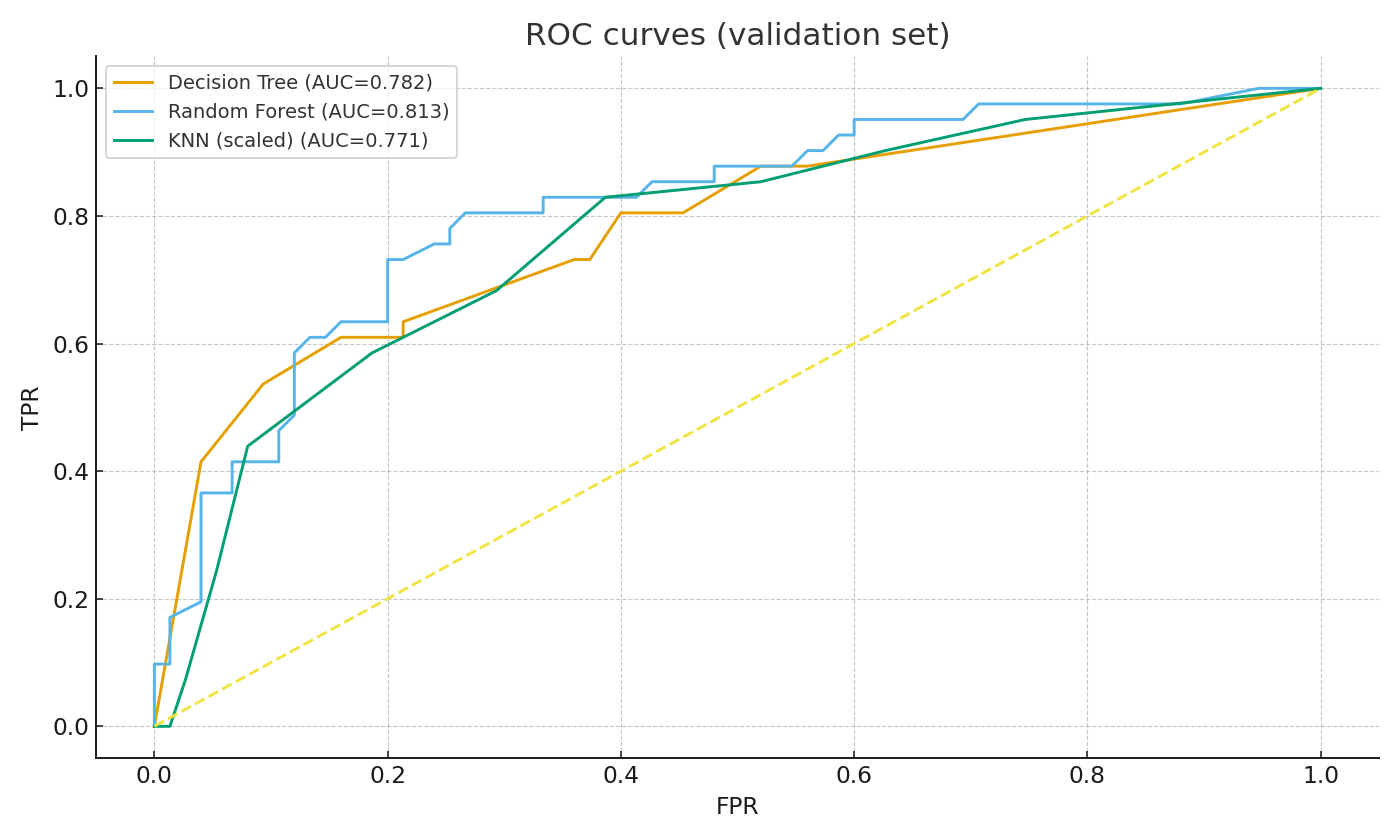


Figure: Scaling is essential—KNN AUC drops without it (validation).

Validation ROC curves:



# 5) Results (Validation → Test) and Model Choice

On the validation set, Random Forest had the best AUC, followed by KNN (scaled) and the Decision Tree. I then retrained the best model on train+validation and evaluated once on the held‑out test set for an unbiased estimate.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | Acc | Prec | Rec | F1 | AUC |
| Decision Tree | 0.664 | 0.517 | 0.732 | 0.606 | 0.782 |
| Random Forest | 0.750 | 0.650 | 0.634 | 0.642 | 0.813 |
| KNN (scaled) | 0.733 | 0.632 | 0.585 | 0.608 | 0.771 |

Chosen model: Random Forest. On the test set it achieved – Acc: 0.750, Precision: 0.649, Recall: 0.600, F1: 0.623, AUC: 0.851.

If this were for clinical use, I’d stick with the forest: it’s more stable than a single tree, gives feature importance for quick sanity‑checks, and it’s less fragile than KNN to future data shifts.

# 6) What I would improve next

• Try threshold tuning to trade precision vs. recall depending on clinic goals.  
• Add calibration (so predicted probabilities match observed risk).  
• Run subgroup evaluation (e.g., by age bands) to check for unintended bias.  
• Log the whole pipeline and metrics with experiment tracking for easy retraining.