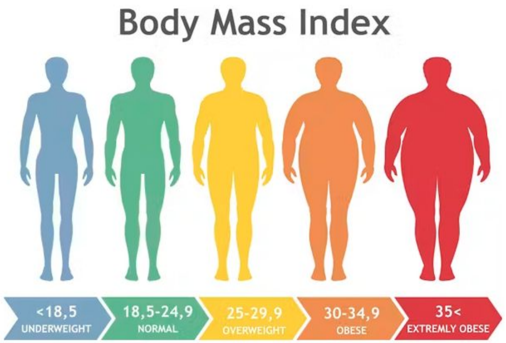
Machine Learning: Obesity Prediction

The Scale Doesn’t Lie — But Does Our Model?

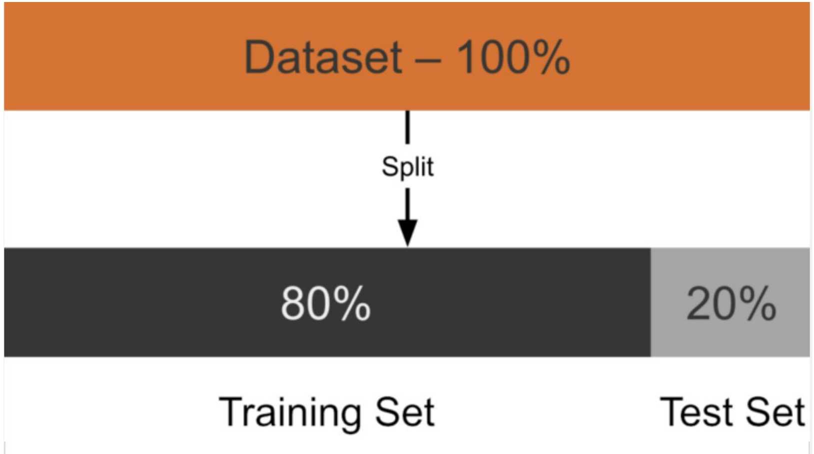


Team:
Nadine Daum, Jasmin Mehnert,
Ashley Razo, Nicolas Reichardt

Intro

- **Predict 7 Obesity Levels**
    - Insufficient Weight
    - Normal Weight
    - Overweight Level I
    - Overweight Level II
    - Obesity Type I
    - Obesity Type II
    - Obesity Type III
- Cross-sectional Data (Kaggle)

Shared Preprocessing



Logistic Regression

Without Ridge

Test Set Performance:
Accuracy: 0.9220

Classification Report:

	precision	recall	f1-score	support
Insufficient_Weight	0.89	0.98	0.93	56
Normal_Weight	0.91	0.81	0.85	62
Obesity_Type_I	0.96	0.94	0.95	78
Obesity_Type_II	0.95	0.97	0.96	58
Obesity_Type_III	1.00	1.00	1.00	63
Overweight_Level_I	0.84	0.86	0.85	56
Overweight_Level_II	0.88	0.90	0.89	50
accuracy			0.92	423
macro avg	0.92	0.92	0.92	423
weighted avg	0.92	0.92	0.92	423

With Ridge

Test Accuracy: 0.9362

Classification Report:

	precision	recall	f1-score	support
Insufficient_Weight	0.90	1.00	0.95	56
Normal_Weight	0.96	0.84	0.90	62
Obesity_Type_I	0.96	0.97	0.97	78
Obesity_Type_II	0.95	0.93	0.94	58
Obesity_Type_III	0.95	0.98	0.97	63
Overweight_Level_I	0.88	0.91	0.89	56
Overweight_Level_II	0.94	0.90	0.92	50
accuracy			0.94	423
macro avg	0.94	0.93	0.93	423
weighted avg	0.94	0.94	0.94	423

KNN - Strategy & Encoding

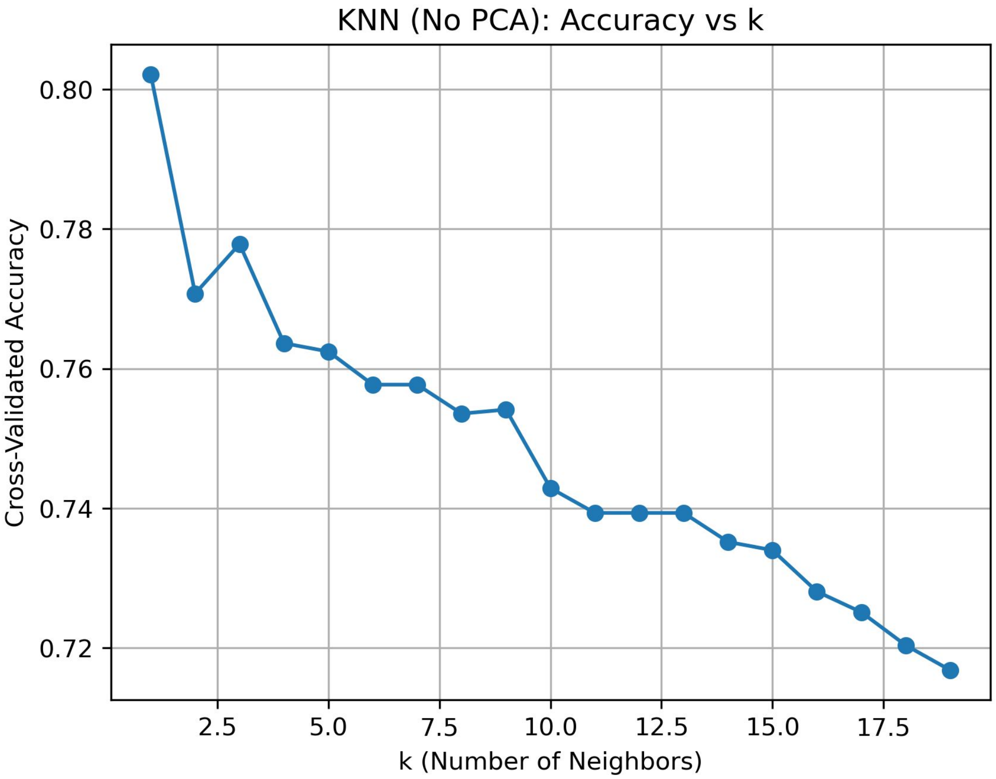
Baseline KNN first, constantly making it better

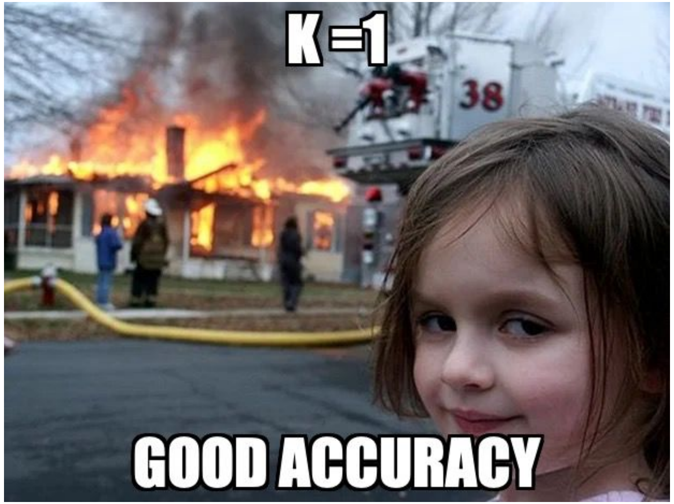
```
ordinal_mappings = {
    "vegetables_freq": ["Never", "Sometimes", "Always"],
    "main_meal_count": ["Between 1 y 2", "Three", "More than three"],
    "snacking_freq": ["no", "Sometimes", "Frequently", "Always"],
    "water_intake": ["Less than a liter", "Between 1 and 2 L", "More than 2 L"],
    "physical_activity_freq": ["I do not have", "1 or 2 days", "2 or 4 days", "4 or 5 days"],
    "screen_time_hours": ["0-2 hours", "3-5 hours", "More than 5 hours"],
    "alcohol_consumption_freq": ["no", "Sometimes", "Frequently", "Always"]
}
```

Never < Sometimes < Always

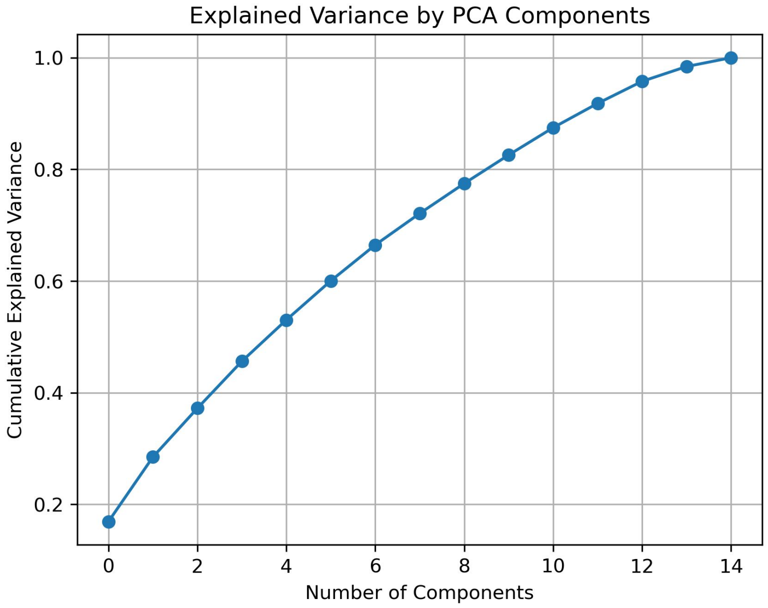
0 < 1 < 2

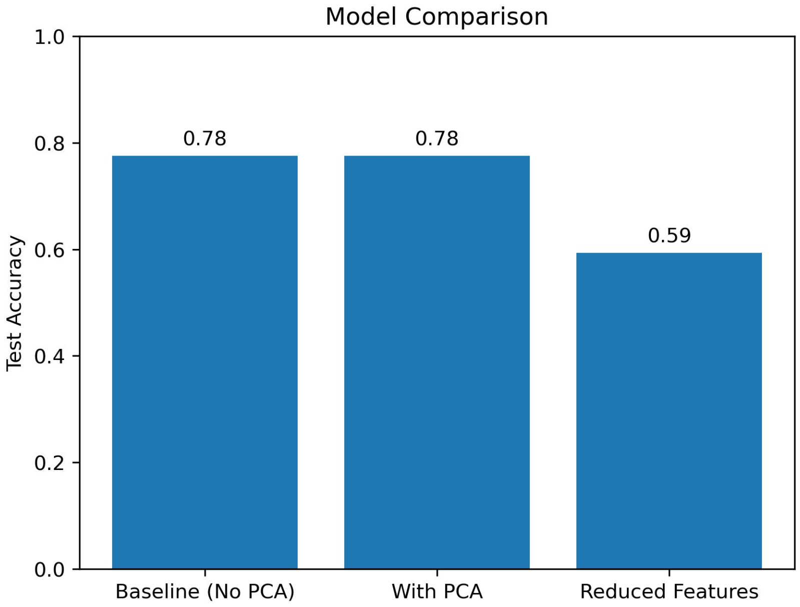
KNN - The Baseline



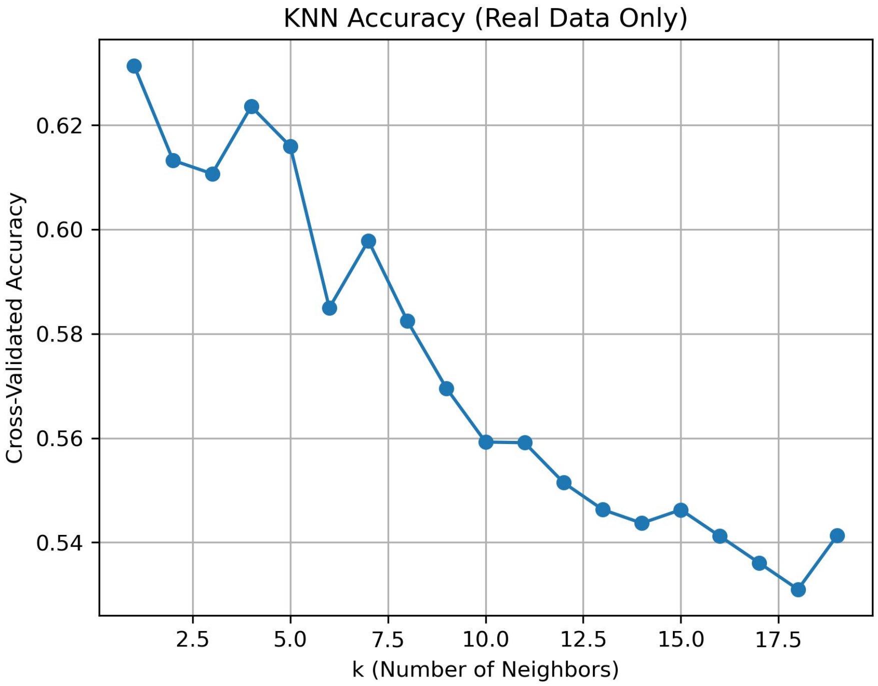


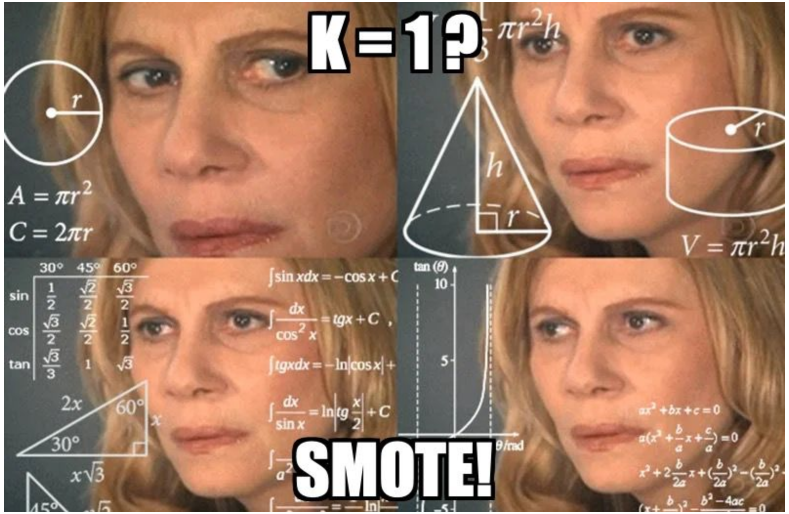
KNN … but with PCA





KNN … but why k=1?





KNN didn’t give the best results, but …

- Ordered encoding is better than one-hot for distance-based models

- PCA only helps if there’s actual redundancy

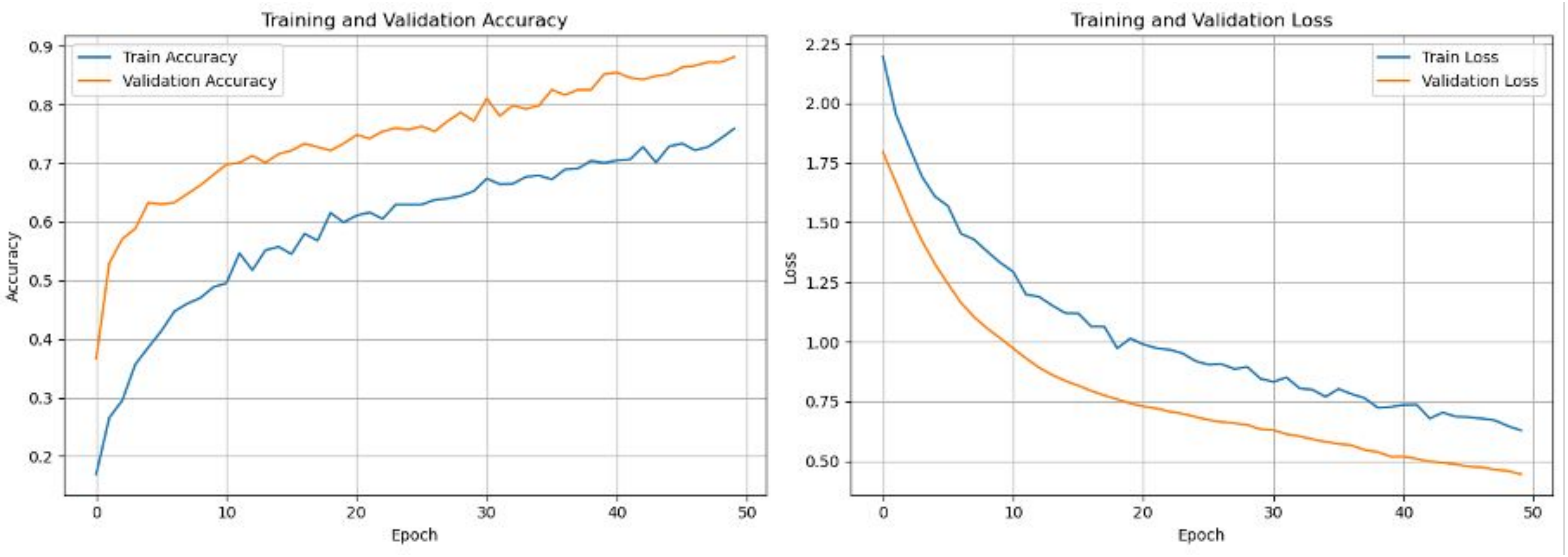
- SMOTE can make small k look better than it is

Neural Network in a Nutshell 🌰

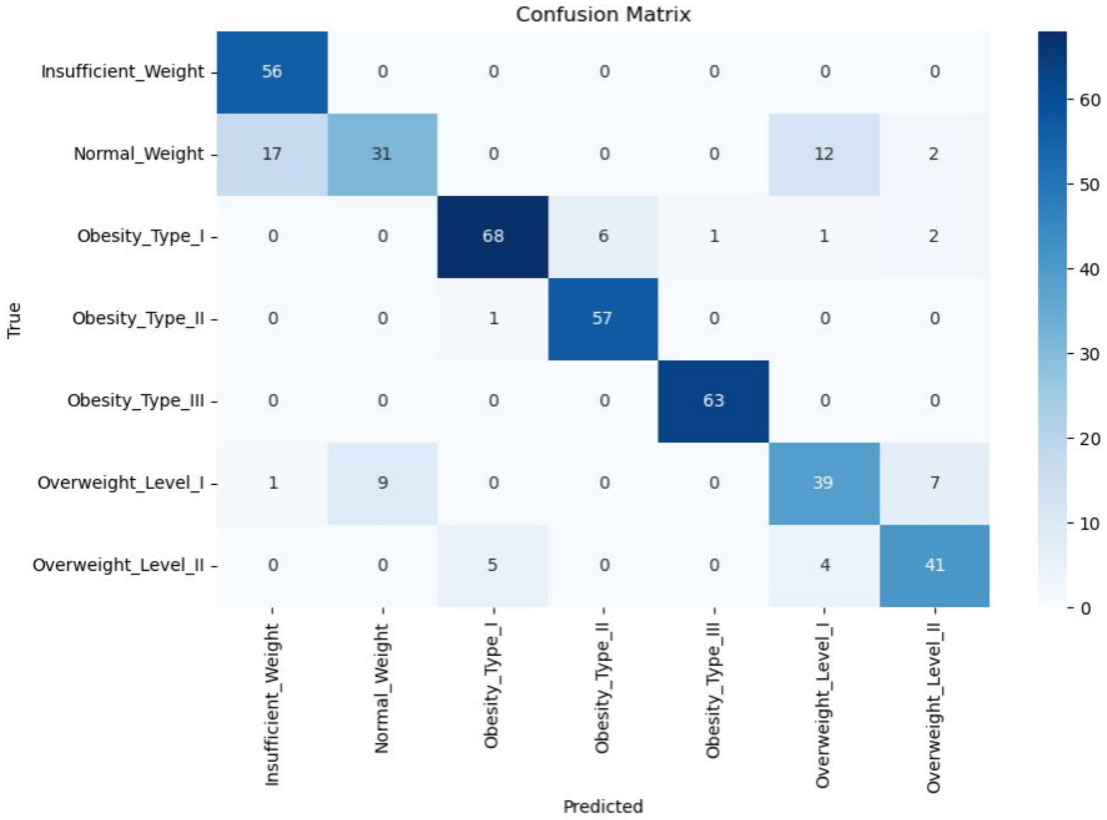
- Multi-class classifier (7 classes) using ReLU & dropout layers
- Trained on shared preprocessed data (scaled & one-hot encoded)
- Achieved **83.9% test accuracy** with smooth learning curve
- Balanced predictions across all obesity categories
- Training setup: categorical crossentropy, Adam optimizer, 50 epochs

→ after 50 rounds, the model stopped guessing & started generalizing

Neural Network: Training Curve



Neural Network: Confusion Matrix



Tree-based models

Leaf Us Alone, We're Too Busy Making Accurate Predictions

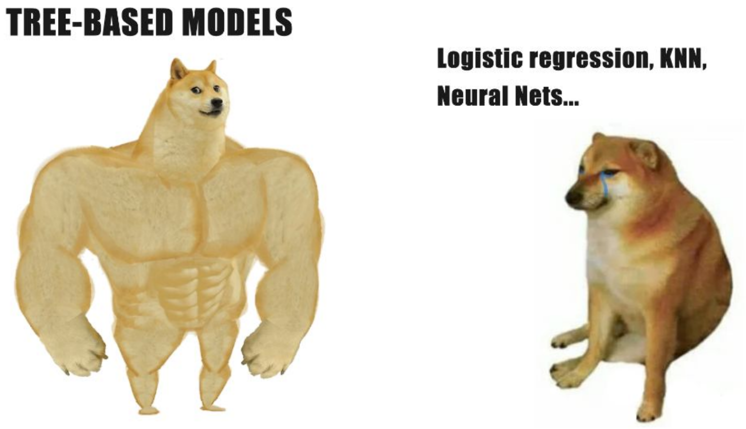
- Tree-based models achieved **exceptional accuracy**:
  - Decision Tree: **96.11%**
  - XGBoost: **95.74%**
  - Random Forest: **93.6%**

- All of them show **excellent generalizability**, extremely high cross-validation accuracies (>95%)

- Decision Tree hyperparameters:
  - Criterion: Entropy (measures information gain at each split)
  - Max Depth: 15 (maximum levels in tree hierarchy)
  - Min Samples Split: 2 (minimum samples needed to split a node)
  - Min Samples Leaf: 1 (minimum samples required in leaf nodes)

Leaf Us Alone, We're Too Busy Making Accurate Predictions

- Decision Tree surprisingly outperformed more complex models, why?
  - Relationship between features and obesity appears relatively simple (too simple maybe?)
  - Decision Trees directly select features with highest immediate predictive power
  - Complex models may have lost efficiency trying to model interactions that weren't necessary



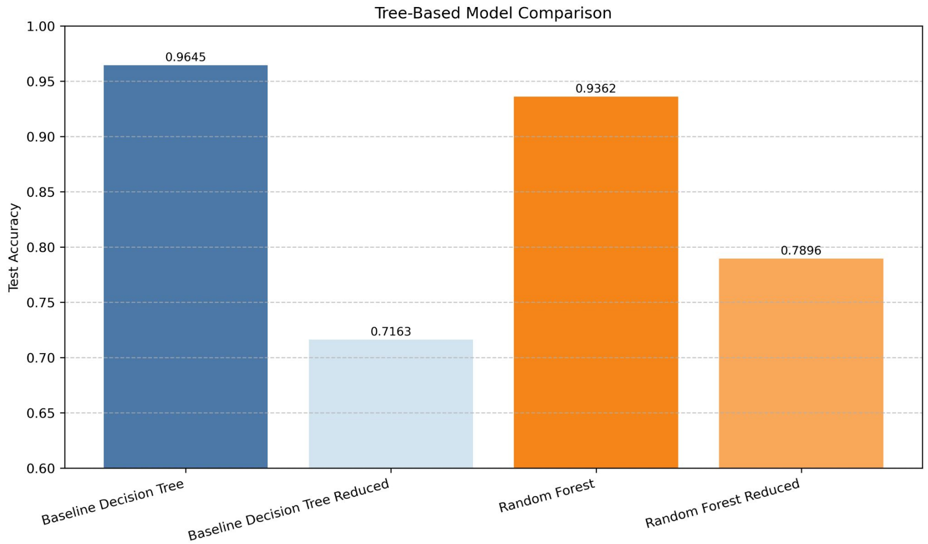
Plot Twist: Our Forest Isn't As Dense As We Thought

Feature_DT	Importance_DT	Feature_RF	Importance_RF
weight_kg	0.6249	weight_kg	0.3335
height_m	0.1965	age	0.114
gender_Male	0.1208	height_m	0.1105
age	0.0237	gender_Male	0.0489
high_caloric_food_freq	0.01	vegetables_freq	0.0424

$$BMI = \frac{weight\,(kg)}{height^2\,(m^2)}$$

Plot Twist: Our Forest Isn't As Dense As We Thought

- Features “weight” and “height” dominated feature importance!
- Models likely reverse-engineered BMI rather than discovering behavioral patterns
- We re-run the three tree-based models without the two features… and accuracy dropped dramatically
  (to ~79% for Random Forest) 🥲



Model Type	Accuracy	Strengths	Challenges
Logistic Regression	92.2%	Easily interpretable, Computationally inexpensive	Sensitive to multicollinearity, linearity assumption
Linear Regression w/ Ridge	93.62%	Controls overfitting, stable solution	Hard to interpret, hyperparameter tuning needed
KNN + PCA	~77%	Helped to understand the data better	With PCA hard to interpret, compared to other models low accuracy
Neural Network	~84%	Stable learning, strong generalization	Needs tuning, hard to interpret
Baseline Decision Tree	**96.45%**	Highest accuracy, generalization, interpretability	Potential target leakage, Limited behavioral insights
Random Forest	93.62%	Strong performance, consistent results	Same feature dominance issue
XGBoost	95.74%	Excellent performance	Computational intensity using GridSearch, API compatibility issue

Model Type	Accuracy	Strengths	Challenges
Logistic Regression	9		Sensitive to multicollinearity, linearity assumption
Linear Regression w/ Ridge	9		Hard to interpret, hyperparameter tuning needed
KNN + PCA	~		With PCA hard to interpret, compared to other models low accuracy
Neural Network	~		Needs tuning, hard to interpret
Baseline Decision Tree	**9**		Potential target leakage, Limited behavioral insights
Random Forest	9		Same feature dominance issue
XGBoost	9		Computational intensity using GridSearch, API compatibility issue