Data Analysis with Bayesian Network Models:

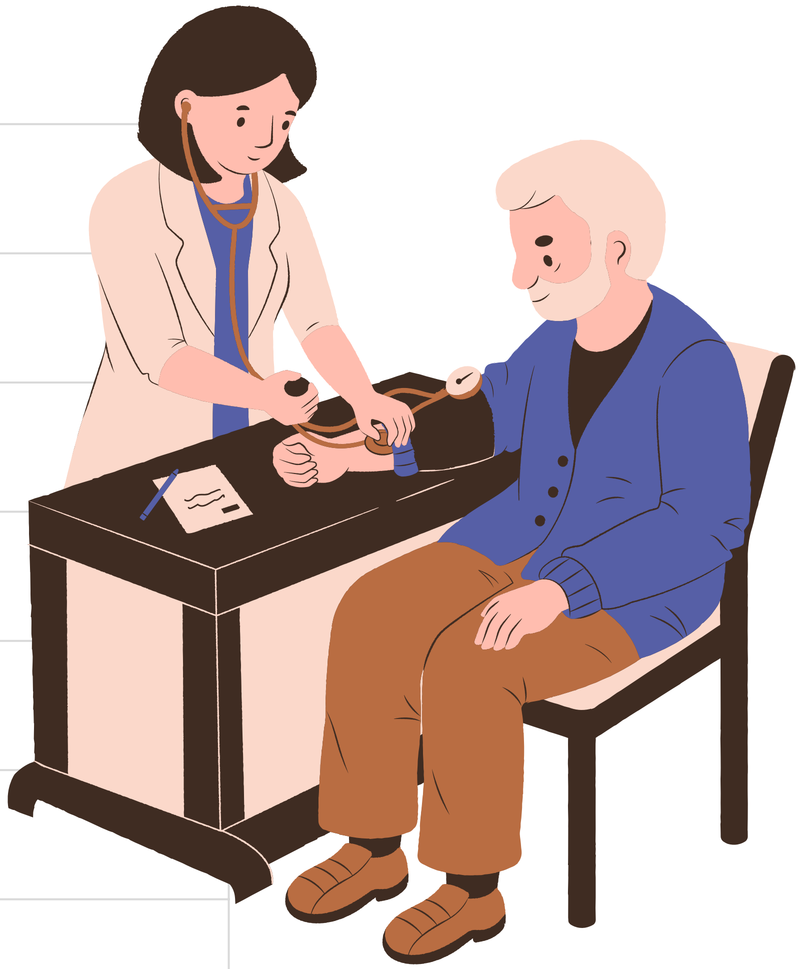
Predicting No-Shows

Sevda Ghasemzadehnaghadehy

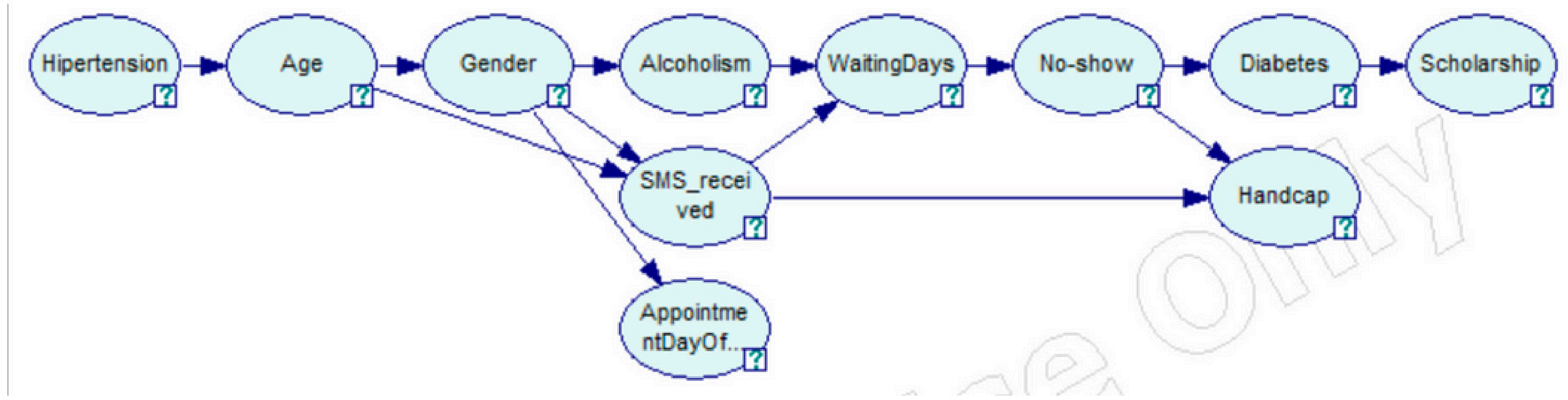
Introduction

Dataset Description: The dataset used in this study is the Medical Appointment No Shows Dataset, collected from medical appointment scheduling records in Vitória, Brazil.
Source: Kaggle, originally provided by a public healthcare institution in Brazil.
Objective: Predict whether a patient will show up for their scheduled medical appointment.
Records: Approximately 110,527 appointment records.
Time Period: Data collected over several months in 2016.
Key Challenge: The dataset is imbalanced, with a much larger number of patients showing up compared to those who miss their appointments.

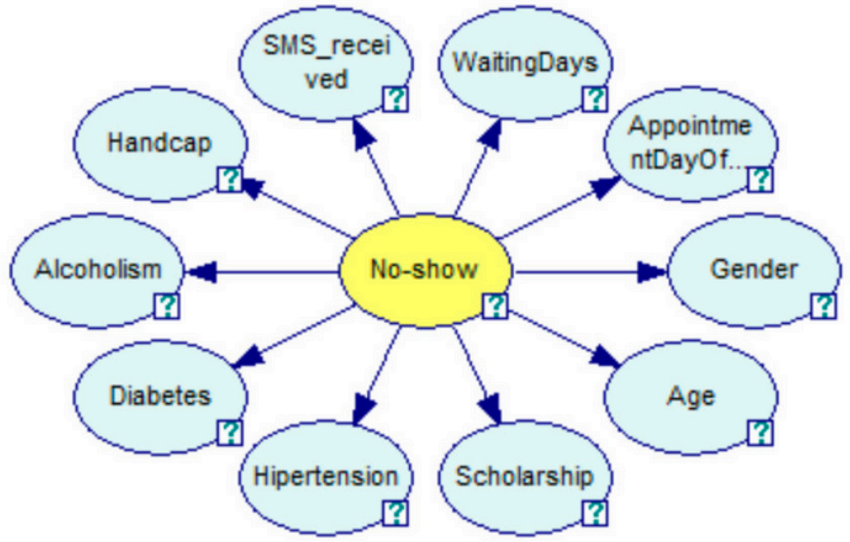
Feature	Description
Gender	Male or Female
Age	Patient’s age
Neighbourhood	Patient’s residential area
Scholarship	Enrollment in the Bolsa Família welfare program (Yes/No)
Hypertension	Presence of hypertension (Yes/No)
Diabetes	Presence of diabetes (Yes/No)
Alcoholism	Presence of alcoholism (Yes/No)
Handicap	Presence of a handicap (binary indicator)
ScheduledDay	Date when the appointment was scheduled
AppointmentDay	Date of the appointment
SMS_received	Whether a reminder SMS was sent to the patient (Yes/No)
No-show	Whether the patient showed up (No) or missed the appointment (Yes)



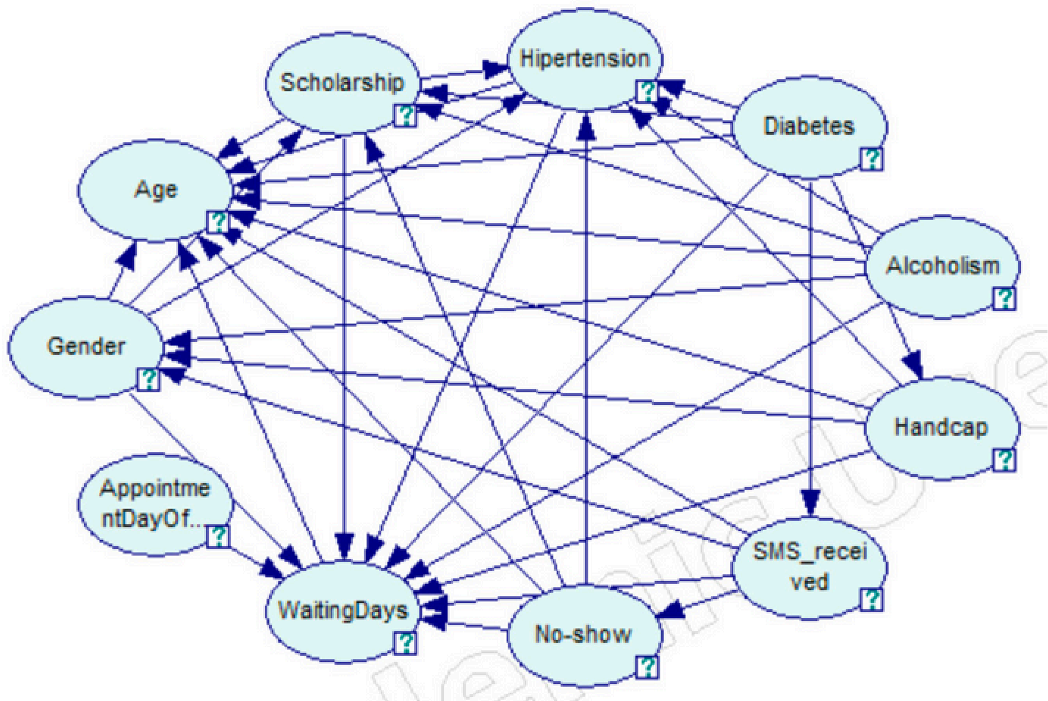
Bayesian network models



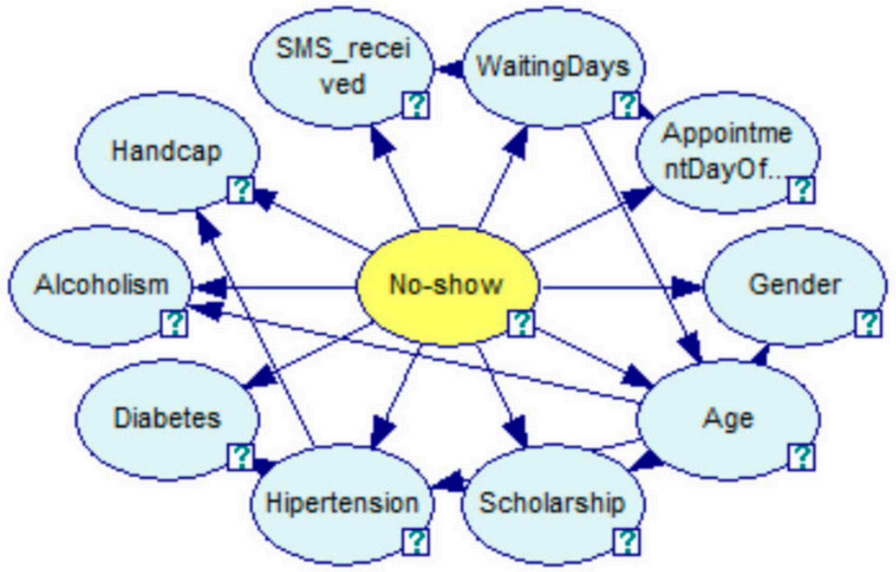
Bayesian Search



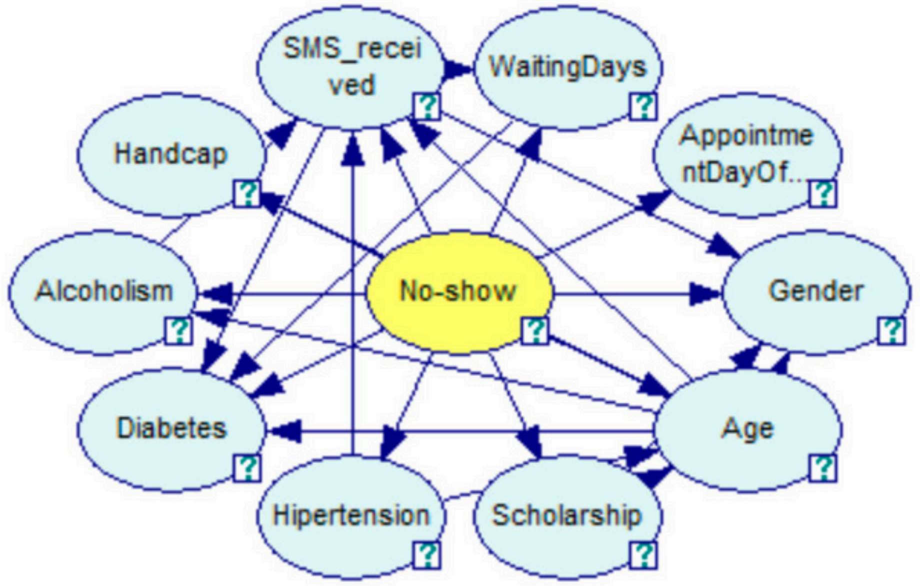
Naive Bayes



PC Algorithm



TAN



ANB

Basic statistics of the created models

Model	Nodes	Arcs	Avg. Parents per Node
Bayesian Search	11	10	0.91
PC Algorithm	11	34	3.091
Augmented Naive Bayes (ANB)	11	25	2.273
Naive Bayes	11	10	0.9091
Tree-Augmented Naive Bayes (TAN)	11	19	1.727

All models have **11** nodes. The **PC Algorithm is the most complex** with 34 arcs and an average of 3.091 parents per node. **Naive Bayes is the simplest**, with only 10 arcs and assumes features are independent. **ANB and TAN are in the middle**, offering a balance between simplicity and connections. **Bayesian Search** also builds a simple network but allows for some extra links. In general, the PC Algorithm finds more complex patterns, while Naive-based models are easier to understand.

Details of discretization of continuous variables

To optimize model performance, continuous variables were transformed into categorical ones:

- **Age:** 4 groups → 0–17, 18–37, 38–54, 55+ years
- **Handicap:** Binary → 0 (No Disability) vs 1+ (Disability)
- **Waiting Days:** 4 groups → 0 days, 1–3 days, 4–7 days, 8+ days
- **Appointment Day Of Week:** 2 groups → Weekday (Mon–Fri) vs Weekend (Sat–Sun)

Strongest Influences in the Model

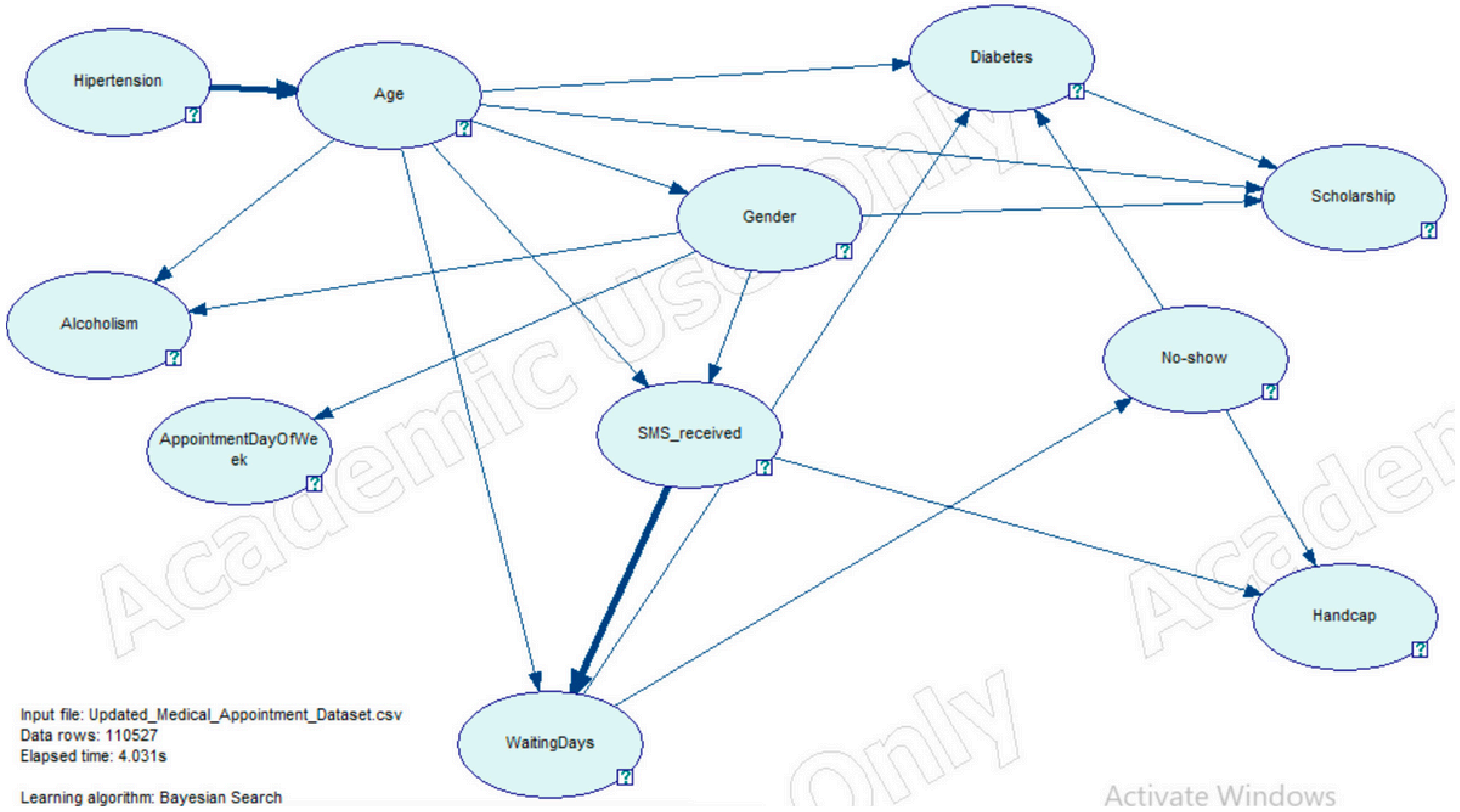
Key variable connections and their average influence:

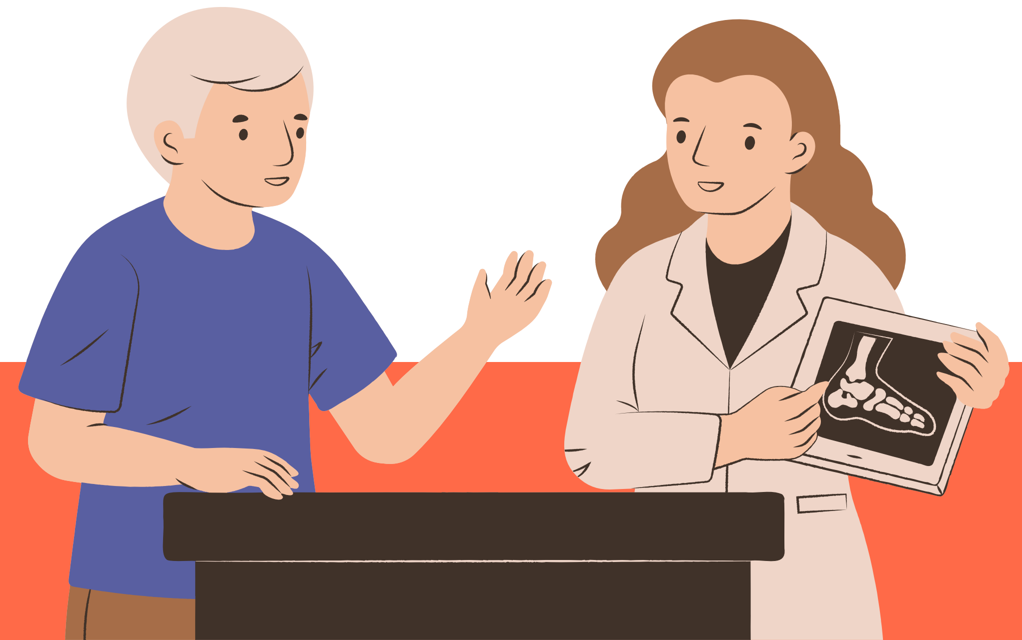
SMS received → Waiting Days (0.5116)

Waiting Days → No-show (0.1178)

Age → Diabetes (0.1131)

No-show → Diabetes / Handicap





Top Diagnostic Variables

Key variables influencing patient no-show probability:

Handicap (+303):
Patients with disabilities are much more likely to miss appointments.

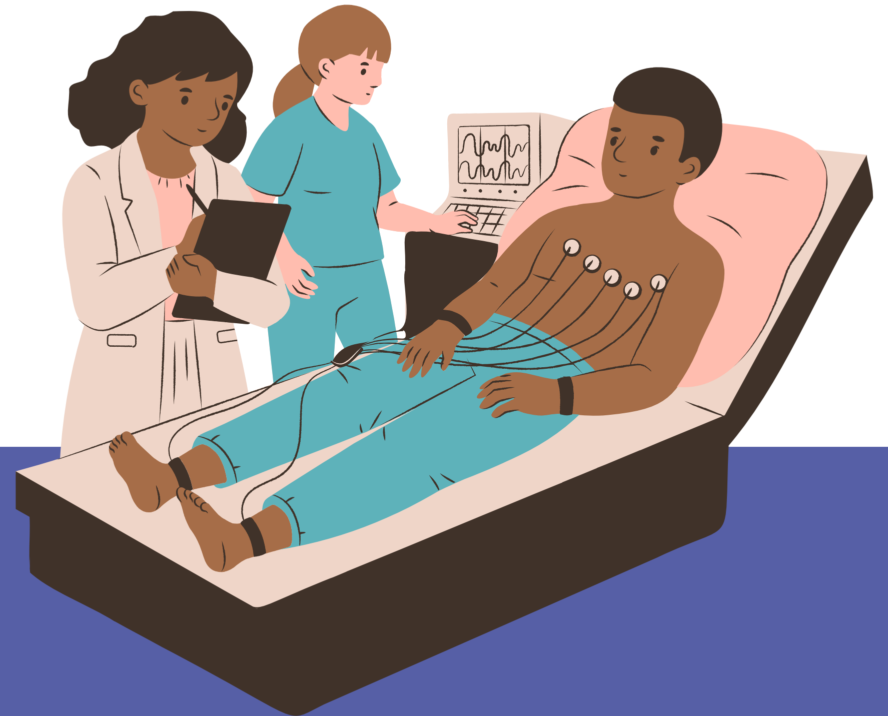
WaitingDays (–135):
Longer waiting times reduce the likelihood of a patient missing their appointment.

SMS received (+96):
Receiving an SMS reminder slightly increases the probability of a no-show, contrary to expectations.

Diabetes (+16):
Patients with diabetes are a bit more prone to miss appointments.

Age (−9):
Older patients are slightly more likely to attend.

➡️ Positive values increase no-show risk;
Negative values decrease it.



Model Performance Summary

Key Insights

Accuracy is ~80%:
Models predict patients who show up very well (majority class).
Sensitivity is very low:
Models fail to predict no-show patients (minority class).
Specificity is very high (>99%):
Models are very good at predicting who will attend.
➡️ **Reason:**

- Class imbalance — most patients show up.
- High accuracy but poor no-show detection.

Conclusion:
Accuracy alone is misleading; sensitivity and specificity must be considered for imbalanced data.

Model	Accuracy (%)	Sensitivity (%)	Specificity (%)
Naive Bayes	79.82	2.02	99.12
ANB	79.2	0.28	99.92
PC	79.66	1.55	99.43
Bayesian Search	79.8	0	100
TAN	79.8	0.04	99.99

CONFUSION MATRIX OVERVIEW

Each matrix shows:

True Positive (TP): Correctly predicted No-show = Yes

True Negative (TN): Correctly predicted No-show = No

False Positive (FP): Incorrectly predicted No-show = Yes

False Negative (FN): Incorrectly predicted No-show = No

Key Points

Accuracy: Overall correct predictions.

Sensitivity: Correctly predicting missed appointments.

Specificity: Correctly predicting attended appointments.

		Predicted	
		No	Yes
Act.	No	**87441**	767
	Yes	21860	**459**

Naive Bayes

		Predicted	
		No	Yes
Act.	No	**87721**	487
	Yes	21995	**324**

PC Algorithm

		Predicted	
		No	Yes
Act.	No	**88207**	1
	Yes	22319	**0**

Bayesian Search

		Predicted	
		No	Yes
Act.	No	**88134**	74
	Yes	22256	**63**

ANB

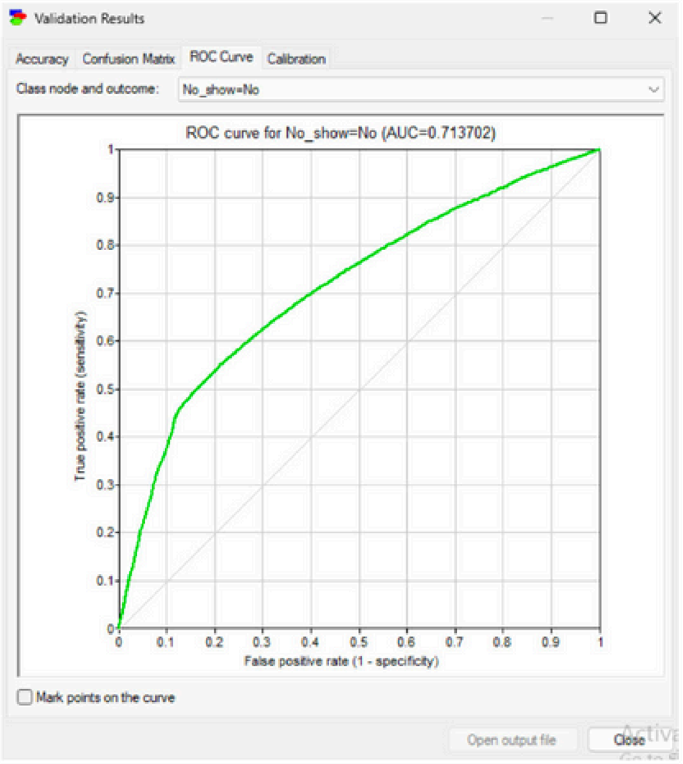
		Predicted	
		No	Yes
Act.	No	**88202**	6
	Yes	22309	**10**

TAN

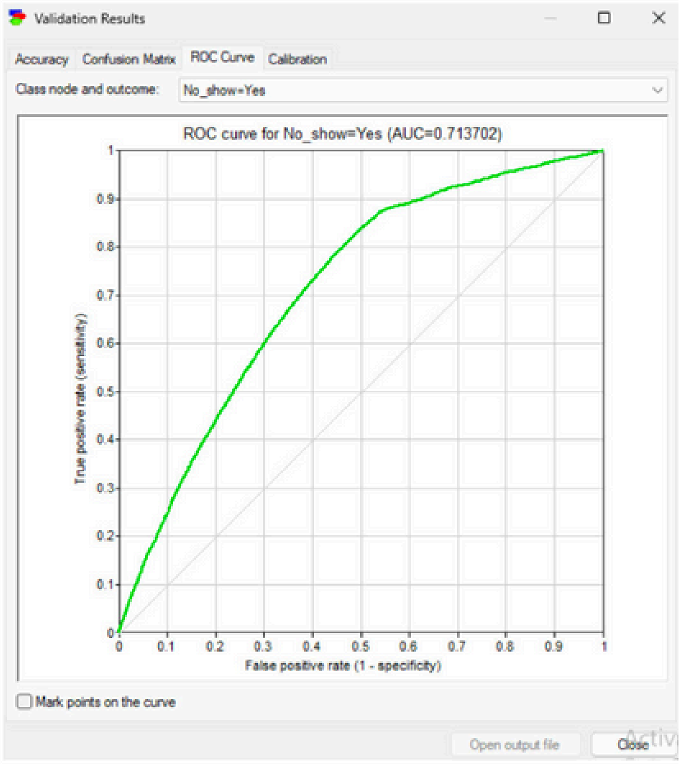
➡️ **Observation:**

Models have high specificity but low sensitivity — they predict attending patients well, but struggle with no-shows due to class imbalance.

ROC Curve Summary



No Show = No



No Show = Yes

The ROC curves generated for the models show moderate AUC values (around 0.68–0.71).

AUC (Area Under Curve) indicates how well the models distinguish between patients who will attend and those who will miss appointments. AUC values close to 0.5 suggest random guessing, while values closer to 1 indicate better discrimination.

In this case, the models demonstrate moderate discrimination ability, meaning they are somewhat effective but not perfect in differentiating between no-shows and shows.

➡️ **Conclusion:**

While the models perform better than random, there is room for improvement, especially due to the class imbalance affecting their ability to detect no-shows.

Handling Class Imbalance with SMOTE

- **The dataset had a class imbalance:**
  - Most patients showed up (majority class).
  - Very few missed their appointments (minority class).
- **Problem:**
  - Models predicted showing up well but failed to predict no-shows, resulting in low sensitivity.
- **Solution: SMOTE (Synthetic Minority Over-sampling Technique):**
  - SMOTE creates new synthetic samples for the minority class by interpolating between existing samples.
  - This improves model learning and reduces bias toward the majority class.
- **Implementation:**
  - Python with imbalanced-learn and scikit-learn libraries.
  - Label Encoding for categorical variables.
  - Balanced dataset generated and used for model training.

➡️ **Benefit:**

SMOTE helped balance the dataset, improving the model's ability to detect no-shows without overfitting.

Impact of SMOTE on Model Performance

Model	Overall Accuracy (Before)	Overall Accuracy (After)	Sensitivity (Before)	Sensitivity (After)	Specificity (Before)	Specificity (After)
Naive Bayes	79.82%	72.07%	2.02%	76.02%	99.12%	68.11%
TAN	79.80%	73.04%	0.04%	75.02%	99.99%	71.05%
Bayesian Search	79.80%	72.73%	0.00%	74.87%	99.10%	70.59%

Impact of SMOTE on Model Performance

I evaluated the effect of SMOTE on three models:

Naive Bayes, TAN, and Bayesian Search.

Before applying SMOTE, all models showed high overall accuracy but extremely poor sensitivity, failing to detect the minority class (no-shows).

After applying SMOTE:

- Sensitivity significantly improved across all models (from nearly 0% to ~75%).
- A slight decrease in overall accuracy and specificity was observed.
- The models became more balanced and capable of identifying no-show patients.

Key Takeaway:

SMOTE helps improve minority class detection, making the models more practical for real-world healthcare scenarios, where predicting no-shows is critical for better resource management.

CONCLUSION

Objective:
Predict whether a patient will attend their scheduled medical appointment using demographic and health-related data (e.g., age, gender, medical conditions, appointment details).

Initial Observations:

- High overall accuracy (~79–80%) across models.
- Extremely low sensitivity (~0%), failing to predict no-shows.
- Very high specificity (>99%), meaning models mostly predicted shows.
- This indicated a class imbalance problem: no-shows were much fewer than shows.

Solution:

- To address this imbalance, SMOTE (Synthetic Minority Over-sampling Technique) was applied:
- Generated synthetic no-show samples to balance the dataset.
- Categorical features were label-encoded to enable SMOTE.
- Models were retrained on the balanced dataset.

Results:
Sensitivity significantly improved (~75%).
AUC scores increased, indicating better class discrimination.
Slight drop in accuracy and specificity, but overall the models became much better at detecting no-shows.

Conclusion:
Balancing the dataset with SMOTE made the models more practical for real-world healthcare settings, helping reduce missed appointments and improving hospital resource planning.

Thank You!