

# HEART DISEASE ANALYSIS

A Data-Driven Analysis of the Patient Journey

Team: ALX3\_DAT2\_S3\_T296

Tools: Power BI & Python

# PROJECT PURPOSE & THE "PATIENT JOURNEY"

## Objective:

To visualize a comprehensive machine learning study on cardiovascular disease.

## We analyze the "Patient Journey" across three distinct stages:

1. **Prevention:** Analyzing lifestyle risks in the general population.
2. **Diagnosis:** Comparing general screening vs. detailed clinical diagnosis.
3. **Prognosis:** Predicting survival outcomes for heart failure patients.

# DATA INCLUDED OVERVIEW

**Lifestyle dataset:** 309K patients with information on BMI, weight, diet, smoking, and self-reported general health.

**Clinical diagnosis dataset:** 70K patients with detailed cardiac diagnostics such as chest pain type, cholesterol, blood pressure, and thalassemia test results.

**Prognosis dataset:** 5K heart failure patients with follow-up, ejection fraction, creatinine, sodium levels, and mortality outcomes.

**Risk factor dataset:** 1.03K patients with combined clinical and demographic variables, including age, max heart rate, and angina type.



“Transforming heart health through data and insight.”

# HOW IT WAS BUILT

## Machine Learning Integration

Integrated 4 distinct datasets (over 308k+ records) including raw clinical records and screening data.

## Data Sources

We incorporated 'Feature Importance' rankings generated by a Random Forest Classifier in Python.

## Data Model Strategy

Utilized a 'Siloed Model' (Disassociated Tables) to handle distinct patient populations without data integrity issues.

# PREVENTION: LIFESTYLE FACTORS

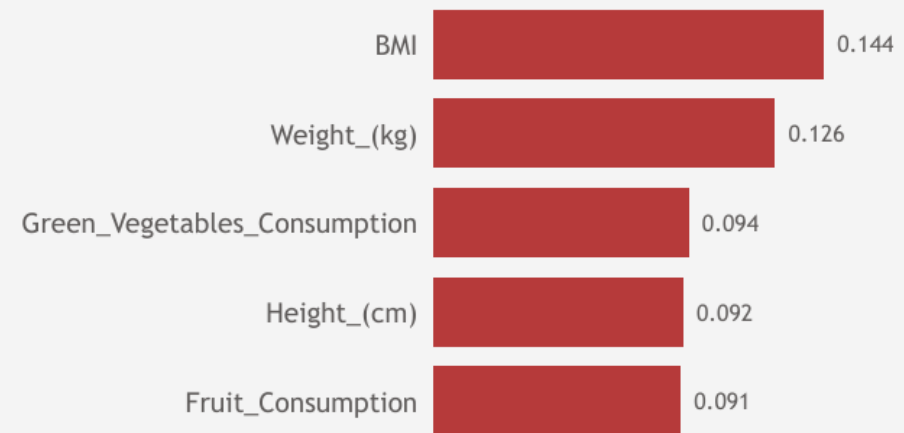
**Core Question:** What lifestyle choices predict heart disease?

**Key Insight:** BMI and Weight are the top risk drivers, outweighing other factors.

## Supporting Data:

- "Poor" general health correlates with a 31.79% risk of disease.
- Smoking history creates a noticeable divide in disease distribution.

Top 5 Lifestyle Predictors of Heart Disease

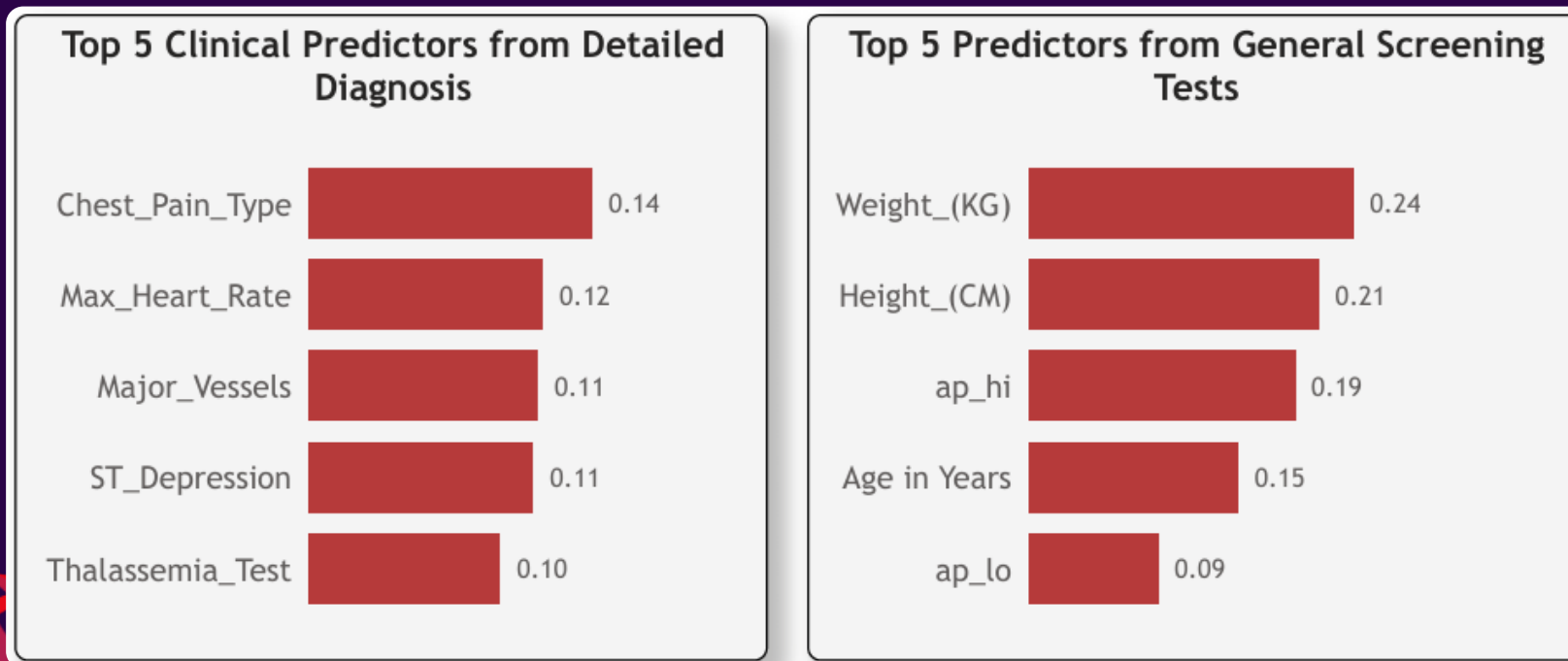


# DIAGNOSIS: SCREENING VS. DETAILED WORKUP

**Comparison:** We compared General Screenings against Detailed Clinical Workups.

**General Screening Drivers:** Driven by Age (0.15) and Blood Pressure (ap\_hi/ap\_lo).

**Clinical Diagnosis Drivers:** Driven by specific symptoms like Chest Pain Type (0.14) and Max Heart Rate (0.12).



# PROGNOSIS: PREDICTING SURVIVAL

**Core Question:** What determines if a heart failure patient survives?

**Key Insight:** Time (Follow-up days) is the #1 factor.

**Biological Markers:** Serum Creatinine (Kidney function) and Ejection Fraction (Heart function) are the dominant biological predictors.

**Risk Curve:** Mortality risk spikes significantly as Serum Creatinine levels rise above 1.5.

# DEEP DIVE: ANALYZING CLINICAL SYMPTOMS

**Core Question:** How do specific symptoms like Chest Pain and Heart Rate correlate with disease?

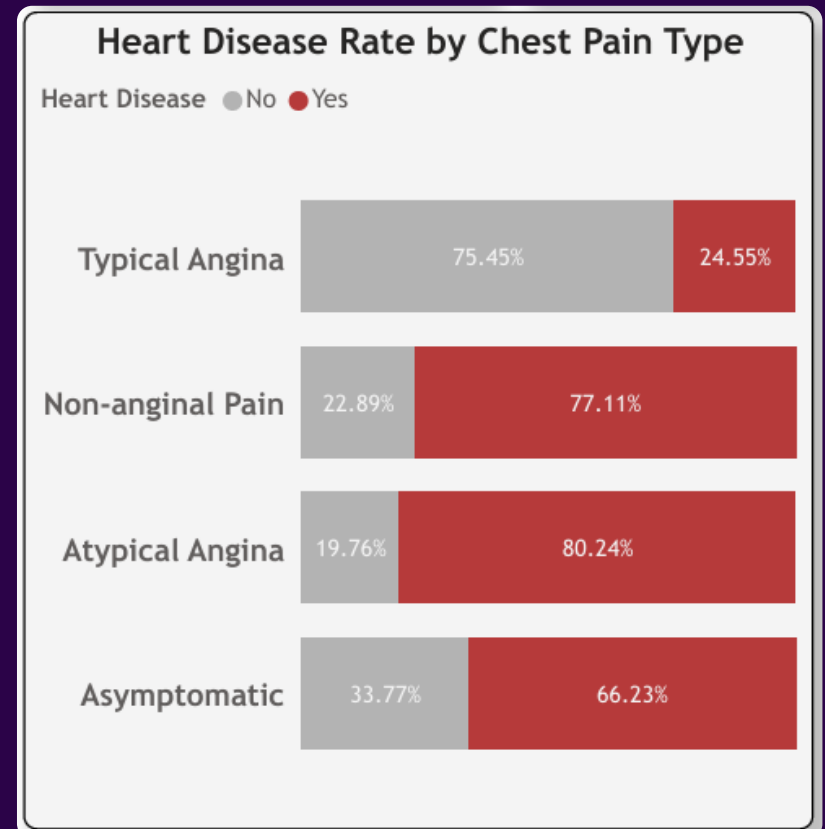
## Chest Pain Insight:

"Typical Angina" is the highest indicator: 75.45% of patients with this pain type tested positive for heart disease.

"Asymptomatic" Paradox: Surprisingly, 66.23% of patients reporting no symptoms (Asymptomatic) still had heart disease, highlighting the danger of "silent" cases.

**Thalassemia:** Patients with "Fixed Defect" Thalassemia showed a 78.33% disease rate.

**Max Heart Rate:** Analysis shows a clear cluster where lower maximum heart rates in older patients correlate with higher disease presence.



# ENGINEERING THE SOLUTION

**TL (Power Query):** Converted binary data (0/1) into meaningful text labels (e.g., "Yes/No") to allow for better chart segmentation.

**DAX Measures:** Created dynamic KPIs using DIVIDE and CALCULATE to normalize data across different sample sizes.

- **Example:** % Disease Rate and Mortality Rate.

**Formatting:** Used "Don't Summarize" settings for Feature Importance to ensure the ML scores were rendered accurately.

```
def login(email, password):  
    if ("@" not in email) or (len(password) < 8):  
        raise Exception("Invalid email or password")  
  
    user = find_user_by_email(email)  
  
    password_is_valid = compare_encrypted_password(password, user.password)  
  
    if (password_is_valid):  
        create_session(user.id)  
    else:  
        raise Exception("Invalid credentials!")  
  
def signup(email, password):  
    if (("@" not in email) or (len(password) < 8)):  
        raise Exception('Invalid input!')  
  
    add_user(email, password)  
    return get_user(email, password)  
def database():
```

# FUTURE ROADMAP

How we'll scale in the future

**Predictive Input Simulator:** Create a "What-If" parameter section where doctors can input patient stats to get a real-time risk score.

**Real-Time Data Connection:** Move from static CSV files to a live API connection with hospital databases (EMR).

**Mobile App Integration:** Optimize the layout for mobile view to allow doctors to check risk factors on rounds.

**Expanded Demographics:** Incorporate geographic data to analyze heart disease "hotspots" by region.

# SUMMARY OF FINDINGS

Moving from reactive treatment to proactive data-driven prevention.

**Operationalized AI:** Successfully turned complex Python Random Forest models into an accessible business intelligence tool.

**The Risk Shift:** Visualized the critical shift from lifestyle-driven risks in the general population to organ-function risks in diagnosed patients.

**Impact:** Enables healthcare providers to target the right risk factors at the right stage of the patient journey.

# THANK YOU