CUCHE Claire
DARDE Inès
DE WLIDE Chloé
GINDRE Louis

GROUP 9
EXPLAINABILITY AI

LIFESURE INSURANCE

DASHBOARD PRESENTATION

INTRODUCTION

INTERACTIVE ANALYSIS OF MEDICAL COSTS

Data visualization project using Dash & Plotly.
Effects of multiple demographic, behavioral, and environmental factors

Identify key factors influencing individual medical expenses

WHY EXPLORE THESE QUESTIONS ?

Economic, Public health, Policies, ...

Notebook

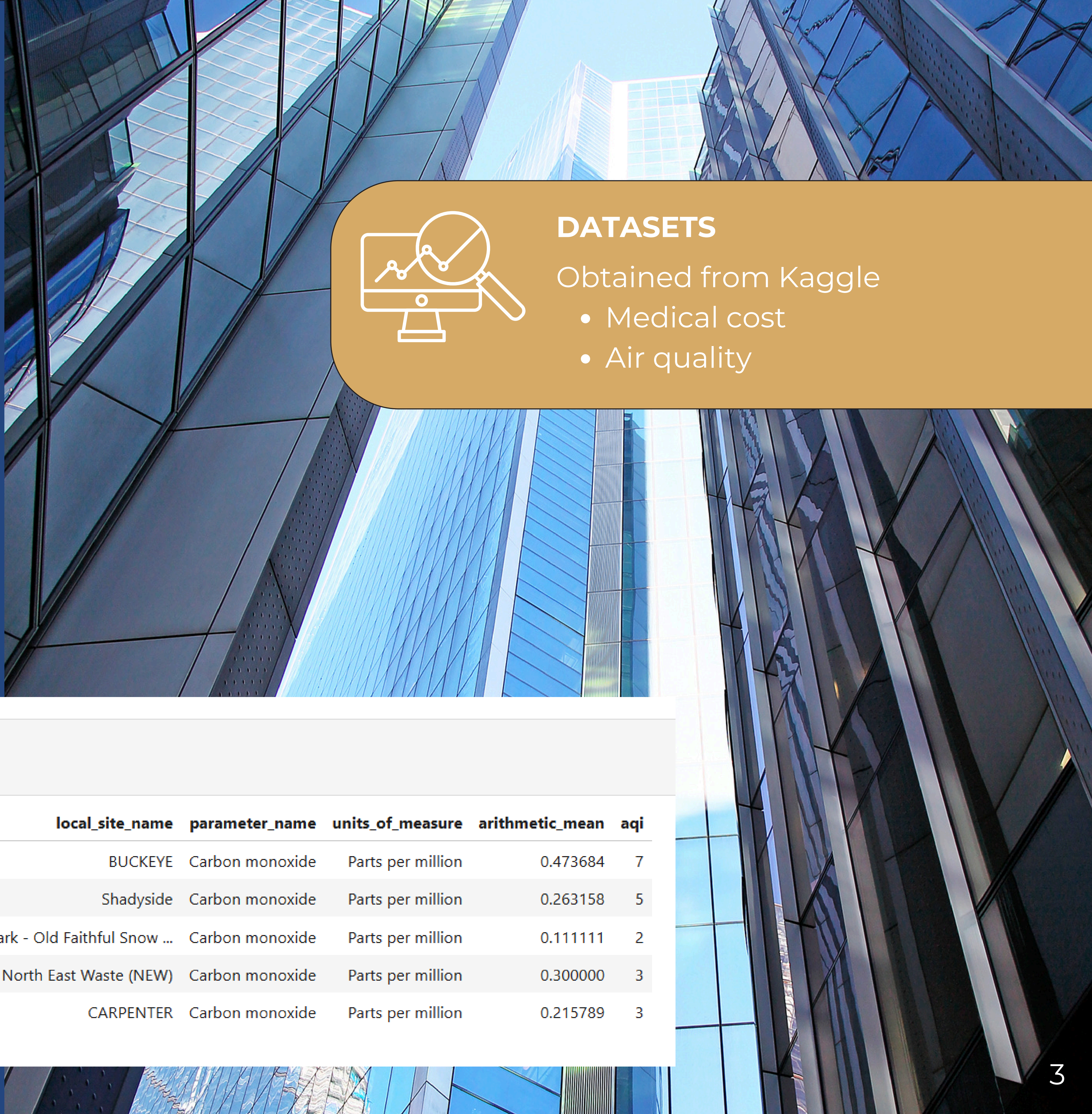
NOTEBOOK

```
df_cost = pd.read_csv('insurance.csv', sep=',', encoding='latin-1')
df_cost = pd.DataFrame(df_cost)
df_cost.head()
```

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32						

```
df_air = pd.read_csv('c4_epa_air_quality.csv', sep=',', encoding='latin-1')
df_air = pd.DataFrame(df_air)
df_air.head()
```

	Unnamed: 0	date_local	state_name	county_name	city_name	local_site_name	parameter_name	units_of_measure	arithmetic_mean	aqi
0	0	2018-01-01	Arizona	Maricopa	Buckeye	BUCKEYE	Carbon monoxide	Parts per million	0.473684	7
1	1	2018-01-01	Ohio	Belmont	Shadyside	Shadyside	Carbon monoxide	Parts per million	0.263158	5
2	2	2018-01-01	Wyoming	Teton	Not in a city	Yellowstone National Park - Old Faithful Snow ...	Carbon monoxide	Parts per million	0.111111	2
3	3	2018-01-01	Pennsylvania	Philadelphia	Philadelphia	North East Waste (NEW)	Carbon monoxide	Parts per million	0.300000	3
4	4	2018-01-01	Iowa	Polk	Des Moines	CARPENTER	Carbon monoxide	Parts per million	0.215789	3



DATASETS

Obtained from Kaggle

- Medical cost
- Air quality

NOTEBOOK

```python
bins_age = [18, 25, 35, 45, 55, 65, 100]
labels_age = ['18-25', '26-35', '36-45', '46-55', '56-65', '65+']
df_cost['age_group'] = pd.cut(df_cost['age'], bins=bins_age, labels=labels_age, right=False)
```

```python
bins_bmi = [0, 18.5, 24.9, 29.9, 35, 100]
labels_bmi = ['Underweight', 'Normal', 'Overweight', 'Obese', 'Extremely Obese']
df_cost['bmi_category'] = pd.cut(df_cost['bmi'], bins=bins_bmi, labels=labels_bmi, right=False)
```

```python
region_to_state = {
    "northeast": [
        "Maine", "New Hampshire", "Vermont", "Massachusetts", "Rhode Island", "Connecticut",
        "New York", "New Jersey", "Pennsylvania"
    ],
    "southeast": [
        "Delaware", "Maryland", "West Virginia", "Virginia", "Kentucky", "Tennessee",
        "North Carolina", "South Carolina", "Georgia", "Florida", "Alabama",
        "Mississippi", "Arkansas", "Louisiana", "District Of Columbia", "Puerto Rico"
    ],
    "southwest": [
        "Texas", "Oklahoma", "New Mexico", "Arizona", "Nevada", "Utah", "California", "Colorado"
    ],
    "northwest": [
        "Washington", "Oregon", "Idaho", "Montana", "Wyoming", "North Dakota", "South Dakota",
        "Nebraska", "Kansas", "Missouri", "Iowa", "Minnesota", "Wisconsin", "Illinois",
        "Indiana", "Ohio", "Michigan", "Alaska", "Hawaii"
    ]
}

def assign_region(state):
    for region, states in region_to_state.items():
        if state in states:
            return region
    return None

df_air["region"] = df_air["state_name"].apply(assign_region)
```

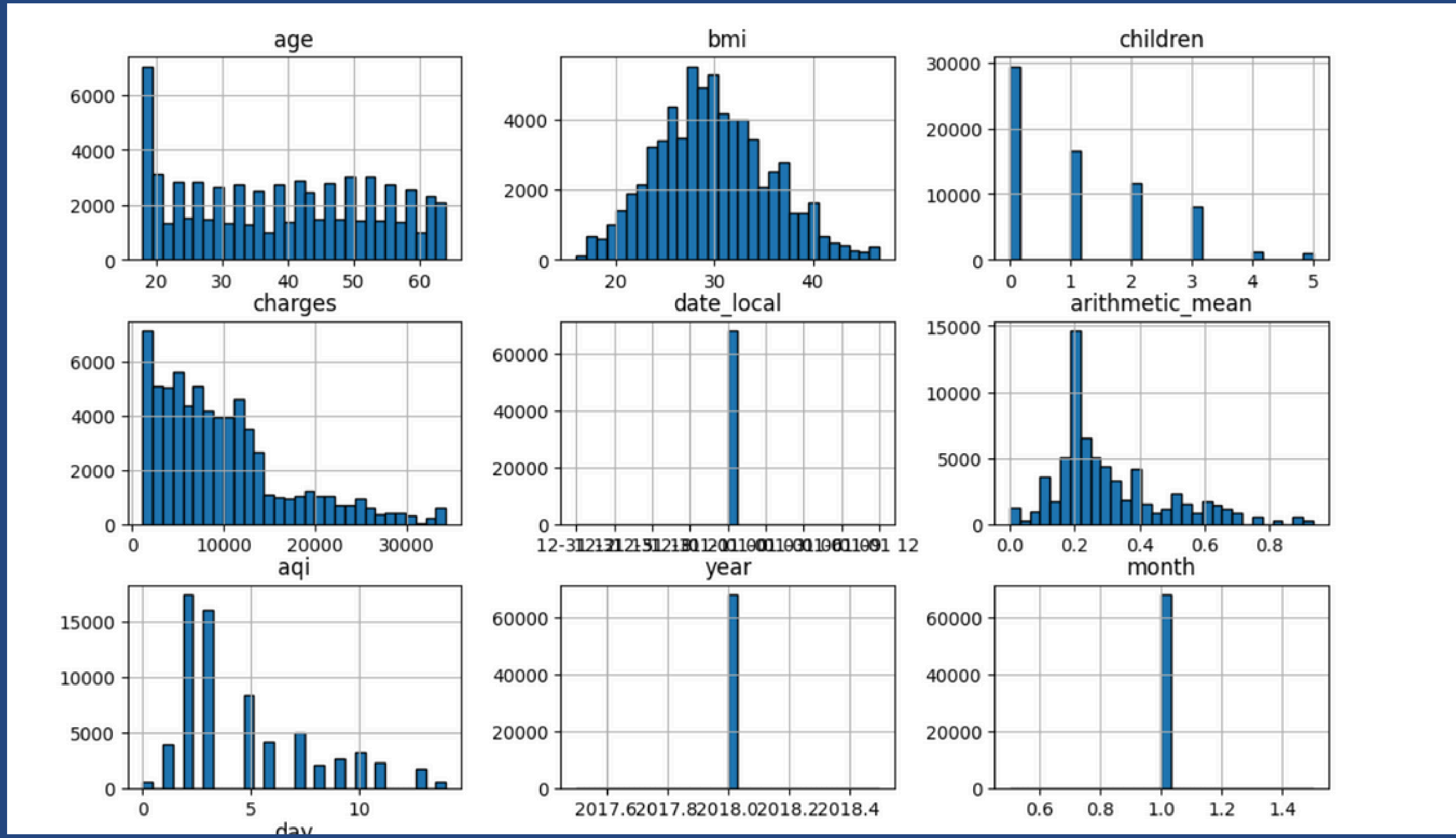
DATA PREPARATION AND CLEANING

- Data Cleaning
- Binning : bmi and sex
- Linking & Merging

```python
df_air.head()
```

	date_local	state_name	county_name	city_name	local_site_name	parameter_name	units_of_measure	arithmetic_mean	aqi	year	month	day	region
0	2018-01-01	Arizona	Maricopa	Buckeye	BUCKEYE	Carbon monoxide	Parts per million	0.473684	7	2018	1	1	southwest
1	2018-01-01	Ohio	Belmont	Shadyside	Shadyside	Carbon monoxide	Parts per million	0.263158	5	2018	1	1	northwest
2	2018-01-01	Wyoming	Teton	Not in a city	Yellowstone National Park - Old Faithful Snow ...	Carbon monoxide	Parts per million	0.111111	2	2018	1	1	northwest
3	2018-01-01	Pennsylvania	Philadelphia	Philadelphia	North East Waste (NEW)	Carbon monoxide	Parts per million	0.300000	3	2018	1	1	northeast
4	2018-01-01	Iowa	Polk	Des Moines	CARPENTER	Carbon monoxide	Parts per million	0.215789	3	2018	1	1	northwest

NOTEBOOK



DATA STATISTICS & DISTRIBUTIONS

Key observations

DASHBOARD

Content:

- Built with Dash (Python framework) for interactivity
- Plotly for dynamic data visualizations
- Integrated HTML/CSS for design and layout
- Data from two merged datasets:
  df_merged.csv & **df_merged1.csv**

Raw data: medical costs, demographics, AQI

Engineered data: encoded variables, interactions (e.g. bmi_aqi)

```
oard.py > ...
import dash
from dash import dcc, html
from dash.dependencies import Input, Output
import pandas as pd
import plotly.express as px
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from scipy.stats import chi2_contingency
import plotly.figure_factory as ff

# Charger les données
df_merged = pd.read_csv("df_merged.csv")
df_merged1 = pd.read_csv("df_merged1.csv")

# Définition de la fonction Cramér's V pour les variables c
def cramers_v(x, y):
    confusion_matrix = pd.crosstab(x, y)
    chi2 = chi2_contingency(confusion_matrix)[0]
    n = confusion_matrix.sum().sum()
    phi2 = chi2 / n
    r, k = confusion_matrix.shape
    phi2corr = max(0, phi2 - ((k-1)*(r-1)) / (n-1))

    # Corriger r_corr et k_corr
    r_corr = r - ((r-1)**2) / (n-1)
    k_corr = k - ((k-1)**2) / (n-1)

    # Assurer que r_corr-1 et k_corr-1 ne soient pas égaux
```

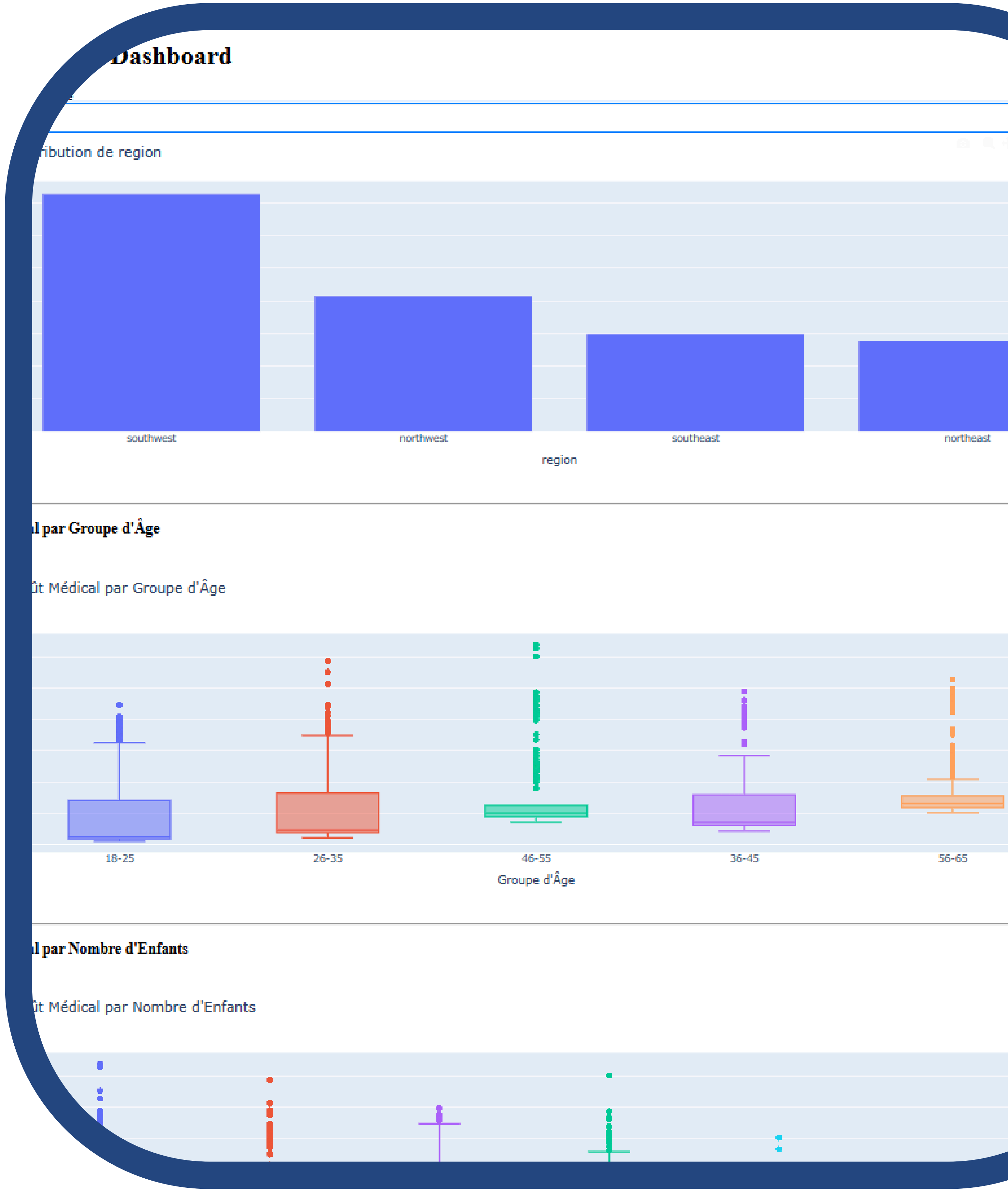
DASHBOARD

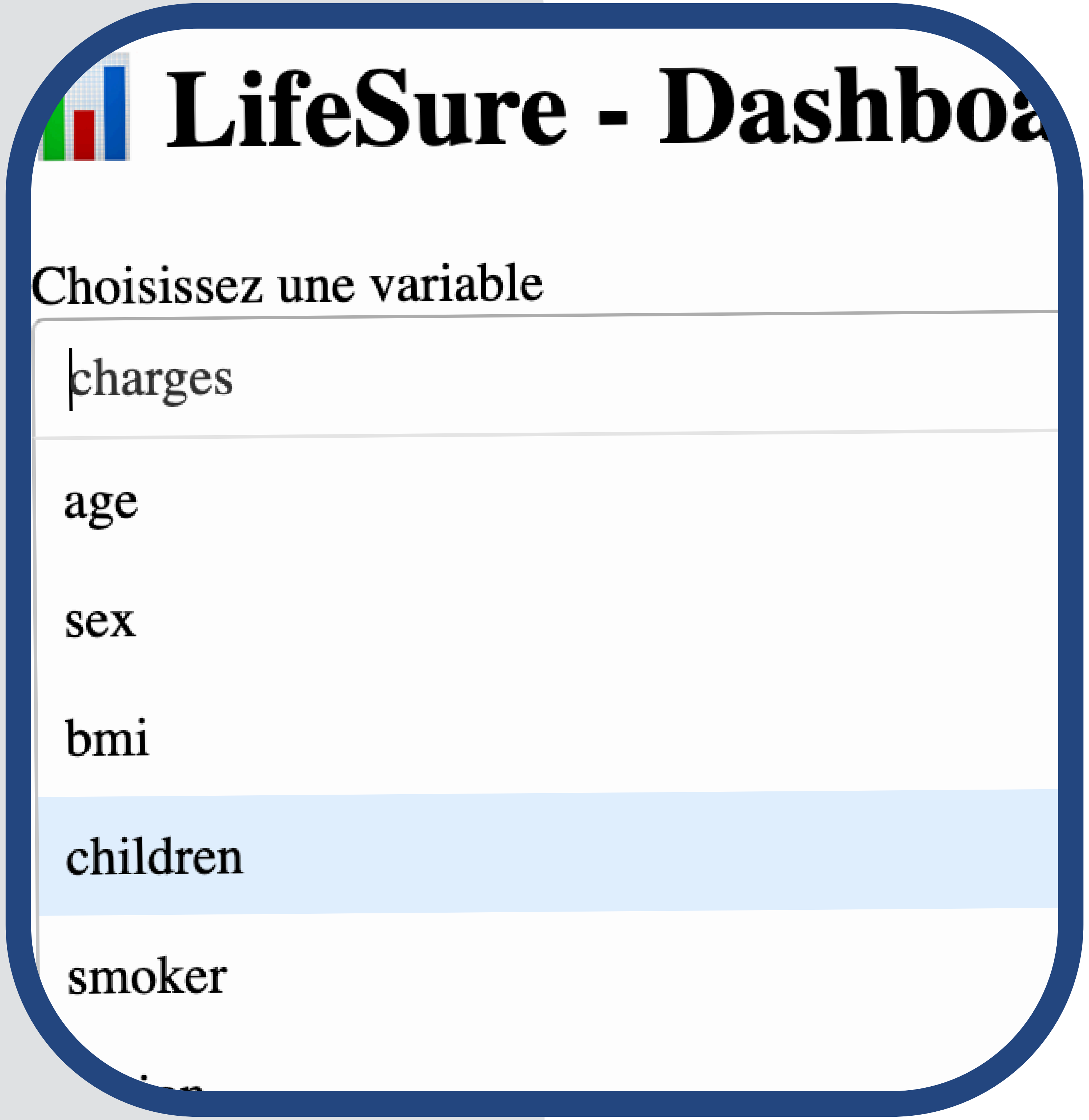
Content :

- Calculated a mixed correlation matrix (Cramér's V & Pearson)
- Interactive components: dropdowns, callbacks, real-time graph updates
- Visualizations: boxplots, barplots, heatmaps

GENERAL ARCHITECTURE

Pipeline, Frontend, Backend & Interactivity
Callbacks & User Interactions
Data Loading & Processing in Code

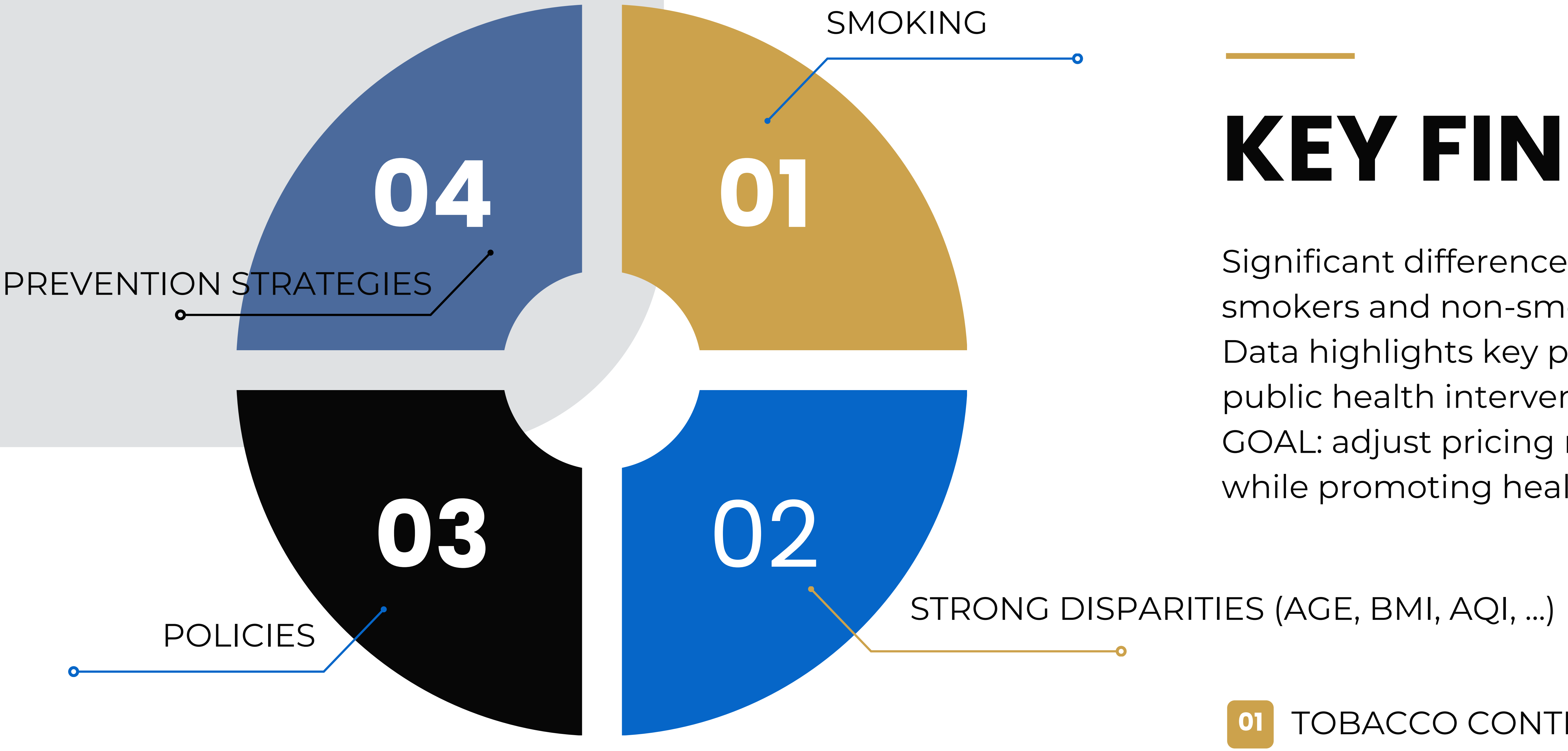




INTERACTIVE EXPLORATION OF KEY RESULTS

LET'S EXPLORE THE DASHBOARD HTTP://127.0.0.1:8050/

Distribution graphs & explaination of the variables:
2.1. Distribution of medical costs
2.2 Geographical influence
2.3 Correlations and advanced analysis
2.4. Combined factors



KEY FINDINGS

Significant difference in medical expenses between smokers and non-smokers.
Data highlights key populations that require focused public health interventions.
GOAL: adjust pricing models based on risk factors while promoting healthier lifestyles.

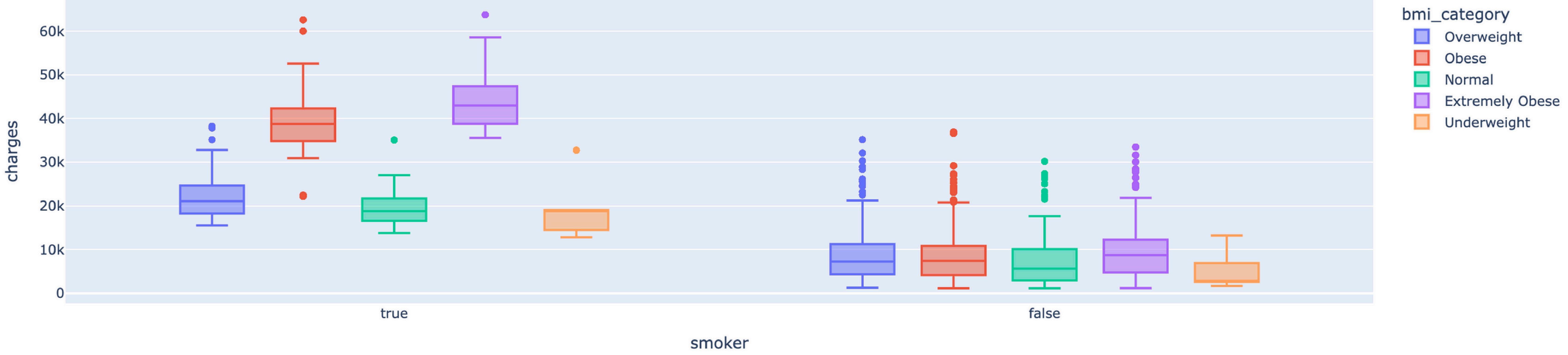
- 01 TOBACCO CONTROL
- 02 OBESITY & NUTRITION
- 03 ENVIRONMENTAL IMPACT
- 04 POLICY IMPLICATIONS

IN-DEPTH ANALYSIS & KEY INSIGHTS

IMPACT OF SMOKING, BMI & POLLUTION ON COSTS

Combined Effect of Smoking & BMI

- Higher costs for smokers with obesity.
- Justifies a dual prevention approach.



IN-DEPTH ANALYSIS & KEY INSIGHTS

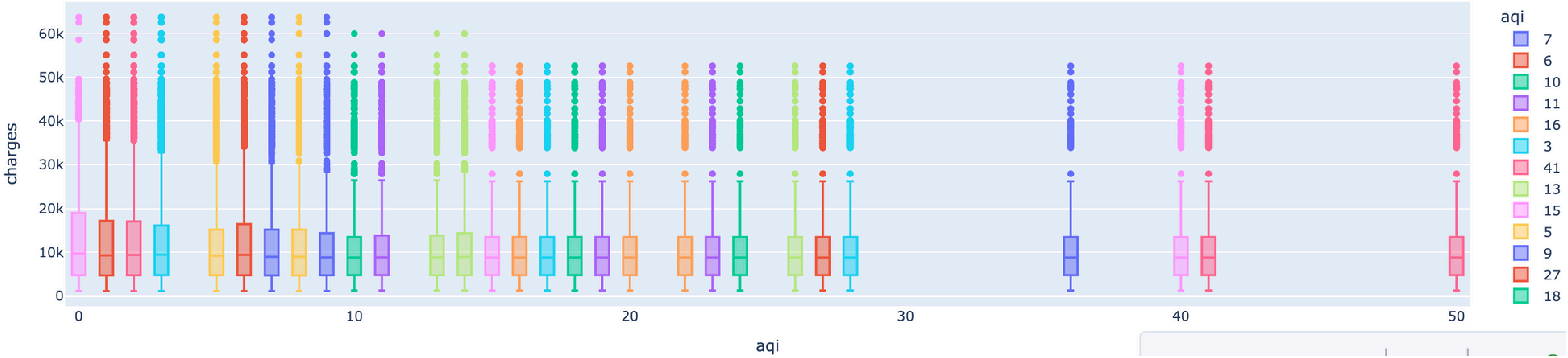
IMPACT OF SMOKING, BMI & POLLUTION ON COSTS

Charges & Pollution

- No direct correlation between AQI & medical expenses.

Vulnerable Populations & Healthcare Access

- Elderly & overweight individuals require greater medical attention.
- Healthcare policies should be tailored for high-risk groups.



POLICY RECOMMENDATIONS & CONCLUSION

POLICY RECOMMENDATIONS & NEXT STEPS

Strengthen anti-smoking policies (higher taxes, prevention programs).

Implement regional health policies (AQI-focused interventions).

Promote healthy lifestyle choices (combat obesity & BMI risks).

PROJECT LIMITATIONS

DATA

Physical activity levels, dietary habits, ...

snapshot in time

GEOGRAPHIC

vary significantly within states

city-level dataset

VARIABLES

Mental health data
Income levels

Healthcare accessibility

<u>Future Extensions & Enhancements</u>

Predictive Analytics with Machine Learning
Dynamic Predictive Dashboard

CUCHE Claire
DARDE Inès
DE WLIDE Chloé
GINDRE Louis

GROUP 9
EXPLAINABILITY AI

THANKS FOR YOUR ATTENTION

DASHBOARD PRESENTATION