US Communities & Crime

TEAM 6

The Scenario

The Goal

Help US government **reduce crime rates** to create new policies and allocate resources effectively.

Why do it?

Multiple benefits to society
"A 10% decrease in homicides can increase housing values by 0.83%"

Requirements

- Crime data is well-organized, **easily accessible** for analysis
- New data is stored and **retrieved effectively.**

Audiences

- Analytical: Can access & query the data
- Managers / C-Suite: High-level overview through interactive visualizations

The Dataset

Communities and Crime Unnormalized Data Set from *UCI Machine Learning Repository*.*

01 147 Attributes

125 predictive & 4 non-predictive

02 Predictive Attributes

- **Demographic and socio-economic data** (age, education, income, employment, race, etc)
- **Crime-related data** (number of homicides, burglaries, and drug-related offense)

03 Non-predictive Attributes

- City and state identifiers
- Community name
- Other geographic descriptors.

Ècommunityname	state	countyCode	communityCode	fold	population	householdsize	ra
BerkeleyHeightstownship	NJ	39	5320	1	11980	3.1	
Marpletownship	PA	45	47616	1	23123	2.82	
Tigardcity	OR	?	?	1	29344	2.43	
Gloversvillecity	NY	35	29443	1	16656	2.4	
Bemidjicity	MN	7	5068	1	11245	2.76	
Springfieldcity	MO	?	?	1	140494	2.45	
Norwoodtown	MA	21	50250	1	28700	2.6	
Andersoncity	IN	?	?	1	59459	2.45	
Fargocity	ND	17	25700	1	74111	2.46	
Wacocity	TX	?	?	1	103590	2.62	
Shermancity	TX	?	?	1	31601	2.54	
SanPablocity	CA	?	?	1	25158	[illegible]	
[illegible]owlingGreencity	KY	?	?	1	40641		

**Source: US Census of 1990, the US FBI Uniform Crime Report of 1995, and the US Law Enforcement Management and Administrative Statistics Survey of 1990.*

Normalization Plan

01 Community (1 table)

- Contains **information about each community,** such as the community_ID (primary key), name, state, population, and codes
- **One to many relationships** between community table and the community_factor tables

02 Value_Type (1 table)

- Store the value type of a value with ***value_type_id*** **(primary key) and** ***value_type***
- **One to many relationships** with the factor tables.

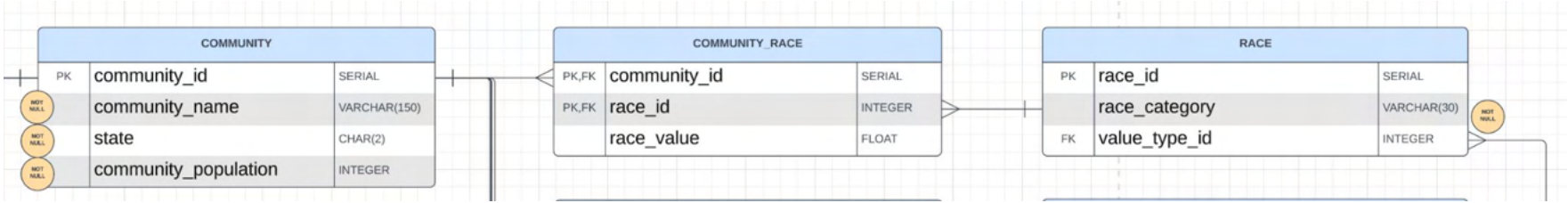
03 Demographic/Factor (21 tables)

- Contain a **unique** ***[factor]_id*** **(primary key), the corresponding** ***[factor]_category*****, and the** ***value_type_id***

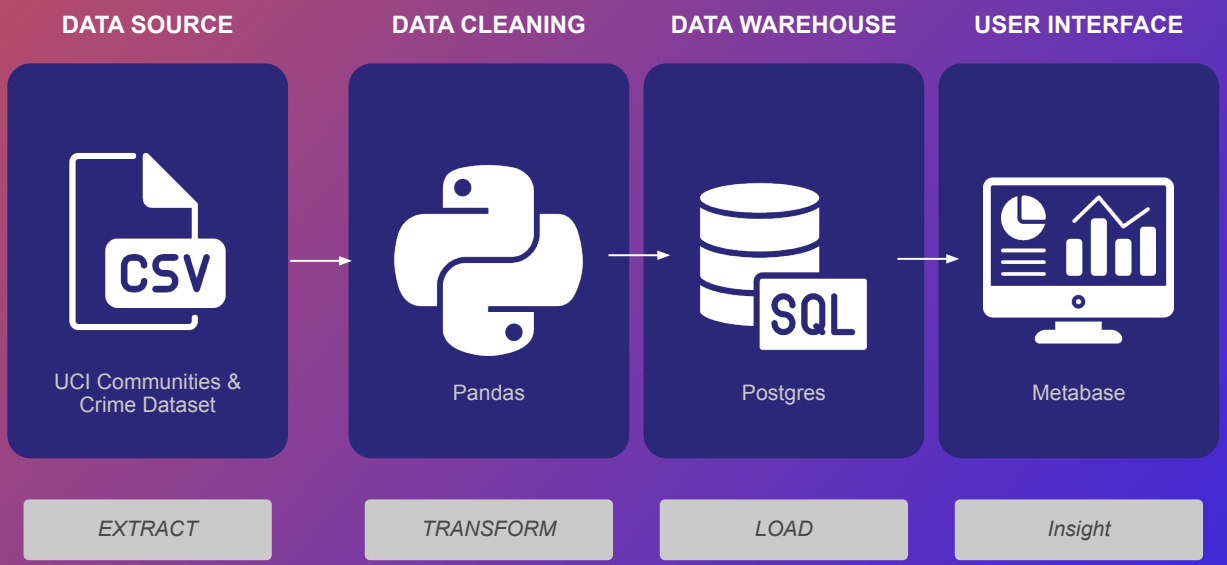
04 Community_factor (21 tables)

- **Combination primary key of the** ***community_id*** **and the** ***[factor]_id*****.** The final column will be ***[factor]_value*****.**
- **One to many relationships** with the factor tables and the community tables.

Schema snapshot (Race factor)



ETL Process


DATA SOURCE
DATA CLEANING
DATA WAREHOUSE
USER INTERFACE
CSV
UCI Communities & Crime Dataset
Pandas
SQL
Postgres
Metabase
EXTRACT
TRANSFORM
LOAD
Insight

01. **Create database and tables** in postgreSQL database

02. **Read csv** into pandas df

03. **Clean df**
Rename communityname column and replace “?” with “-1” as default to represent null values

04. Create **primary key** for community_id

```
# Create commands
stmt = (
"""
CREATE TABLE community (
  community_id serial PRIMARY KEY,
  community_name varchar(150) NOT NULL,
  state char(2) NOT NULL,
  community_population int NOT NULL
);

""",
"""
CREATE TABLE value_type (
  value_type_id serial PRIMARY KEY,
  value_type varchar(30) NOT NULL
);

""",
"""

CREATE TABLE race (
  race_id serial PRIMARY KEY,
  race_category varchar(30) NOT NULL,
  value_type_id int,
  FOREIGN KEY (value_type_id) REFERENCES value_type
);

""",
"""
CREATE TABLE community_race (
  community_id serial,
  race_id int,
  race_value float,
  PRIMARY KEY (community_id, race_id),
  FOREIGN KEY (race_id) REFERENCES race,
    FOREIGN KEY (community_id) REFERENCES community
);

"""
```

Clean Dataset

```
[ ] df1 = df1.rename(columns = {"Êcommunityname":"communityname"})
```

```
[ ] df1.head()
```

	communityname	state	countyCode	communityCode	fold	population	householdsize	racepctblack
0	BerkeleyHeightstownship	NJ	39	5320	1	11980	3.10	1.37
1	Marpletownship	PA	45	47616	1	23123	2.82	0.80
2	Tigardcity	OR	?	?	1	29344	2.43	0.74
3	Gloversvillecity	NY	35	29443	1	16656	2.40	1.70
4	Bemidjicity	MN	7	5068	1	11245	2.76	0.53

5 rows × 147 columns

Convert object type to float and replace ? with -1 as default value for null

```
[ ] for col in df1.columns:
        if df1[col].dtype == object:
            if '?' in df1[col].unique():
                df1[col] = df1[col].replace('?', -1)
            try:
                df1[col] = df1[col].astype(float)
            except ValueError:
                pass
        elif df1[col].dtype == int:
            df1[col] = df1[col].astype(float)
```

```
[ ] df1.info()

    <class 'pandas.core.frame.DataFrame'>
    RangeIndex: 2215 entries, 0 to 2214
    Columns: 147 entries, communityname to nonViolPerPop
    dtypes: float64(145), object(2)
    memory usage: 2.5+ MB
```

05. Follow **E-R diagram** for each table

- Community table: slice columns, rename columns, insert into pgSQL database.
- Value_type table: Create PK IDs for percentages and integers.
- Race table: Slice columns, convert to list as string for race_category, rename variables
- Community_race table: Slice columns, pivot longer, joined df, reorder columns

Slice necessary columns

```
[ ] df_community = df1[['community_id','communityname', 'state', 'population']]
```

Rename columns

```
[ ] df_community = df_community.rename(columns ={"communityname": "community_name",
                                                 "population": "community_population"})
```

Insert into df_community table in database

```
[ ] df_community.to_sql(name='community', con=engine, if_exists='append', index=False)
    215
```

Create df_race

```
[ ] df_race = pd.DataFrame()
```

Slice necessary columns and convert to list as string for category column

```
[ ] df1.iloc[0:5, 8:12]
    race_category = df1.iloc[0:5, 8:12].columns.tolist()
    print(race_category)

    ['racepctblack', 'racePctWhite', 'racePctAsian', 'racePctHisp']
```

```
[ ] df_race['race_category'] = race_category
```

Create df_community_race

Slice necessary columns

```
[ ] df_community_race = df1.iloc[:, [0,8,9,10,11]]
```

pd.melt() to pivot longer

```
[ ] df_community_race = pd.melt(df_community_race, id_vars = ['community_id'], value_vars=race_category)
```

```
[ ] df_community_race.head()
```

	community_id	variable	value
0	1	racepctblack	1.37
1	2	racepctblack	0.80
2	3	racepctblack	0.74
3	4	racepctblack	1.70
4	5	racepctblack	0.53

Customer Interaction Plan

TECHNICAL AUDIENCE

Purpose: Identifying trends and patterns

- **Visualize the violent crime rates** by county or state
- **Analyze correlations** between demographic factors and violent crime numbers across the US

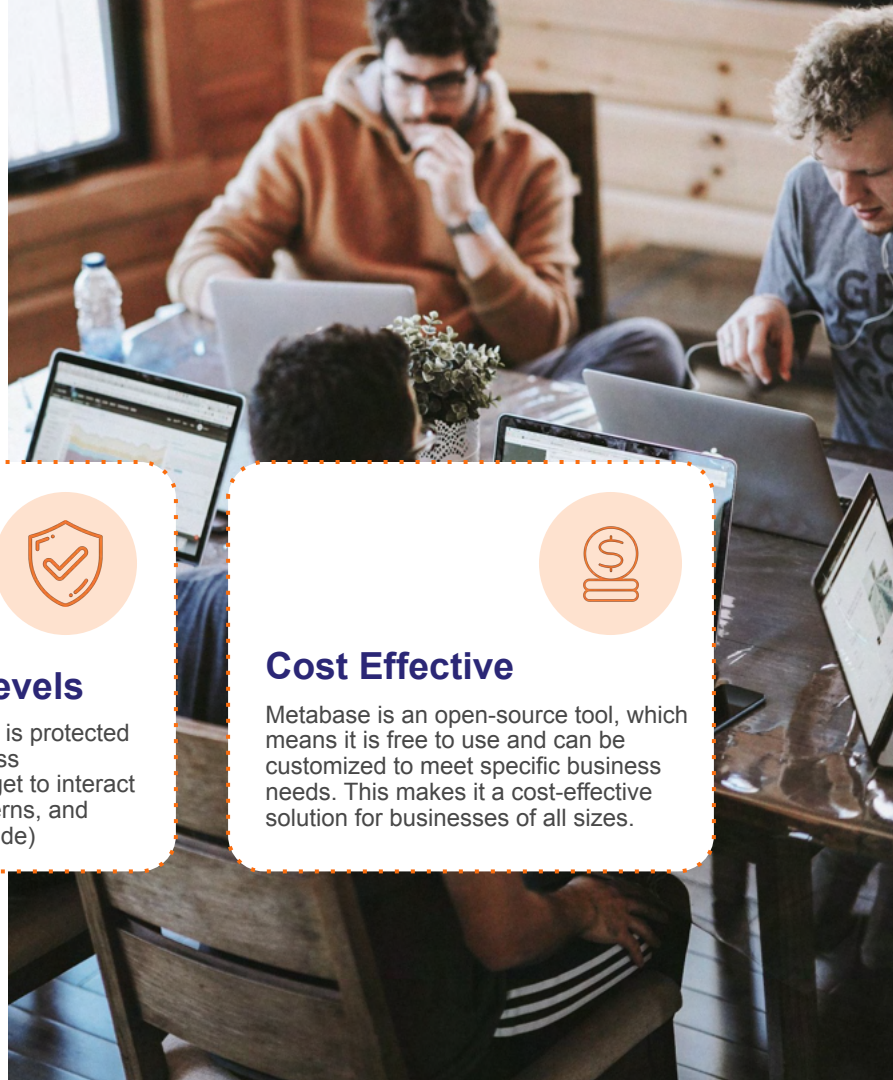
C-SUITE AUDIENCE

Purpose: Viewing patterns & correlations

- Identify better ways to allocate resources
- **Highlight** the most correlated demographic factors with violent crimes for future research
- **Recognize potential societal reasons** and solutions **for committing violent crimes**

Customer Interaction Process

We have used Metabase, a data visualization tool that is customized and interactive



Fast Decision Making

Provides high-level summary of key metrics to support C-levels in decision making circumstances

Enhanced data security for C-levels

- Secure web link so data is protected from unauthorized access
- Non technical C-levels get to interact with the data, view patterns, and generate insights (no code)

Cost Effective

Metabase is an open-source tool, which means it is free to use and can be customized to meet specific business needs. This makes it a cost-effective solution for businesses of all sizes.

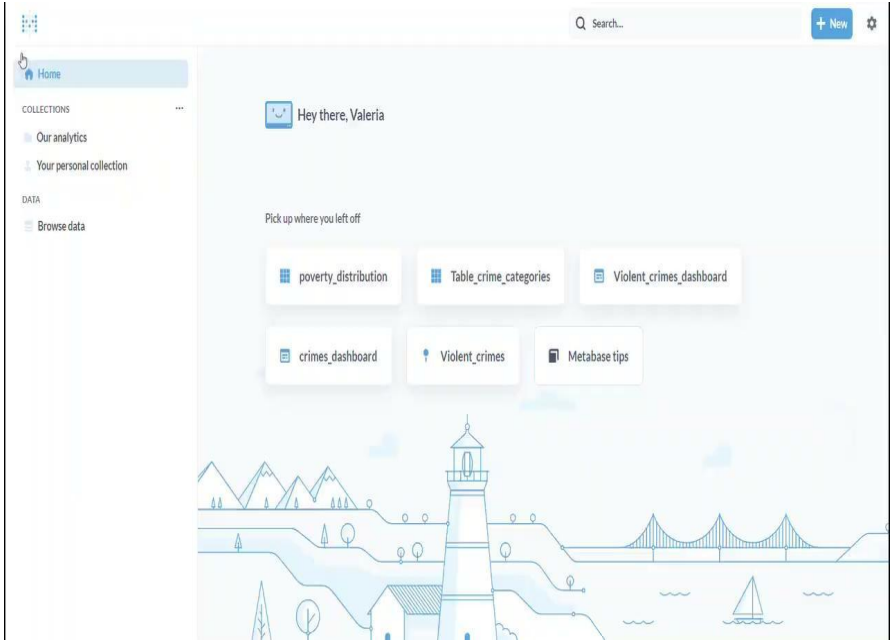
Demo:
Analyst Level

Role Permissions:

- **Data access:** Unrestricted
- **Native query editing:** Yes
- **Download results:** 1 Million rows
- **Manage data model:** Yes
- **Manage database:** No

crime category by US State

```
SELECT community.state,
crimes.crimes_category,
community_crimes.crimes_value
FROM community
JOIN community_crimes ON
community.community_id =
community_crimes.community_id
JOIN crimes ON community_crimes.crimes_id
= crimes.crimes_id
```



Demo:
C-Suite Level

Role Permissions:

- **Data access:** No self-service access
- **Native query editing:** No
- **Download results:** No
- **Manage data model:** No
- **Manage database:** No

Analytical Procedures:

- Which region/states of the US has higher levels of crime for both violent and non-violent?
- Is there a correlation between demographics and violent crimes?
- Does police play a role in violent crimes?