In [1]:

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import statsmodels.api as sm
import seaborn as sns
import itertools
from ipywidgets import interact, interactive
import re
import hashlib
sns.set(style="dark")
plt.style.use("ggplot")
%matplotlib inline
```

In [2]:

```
copd = pd.read_csv("chronic_obstructive_pulmonary_disease.csv")
state_populations = pd.read_csv("pop.csv")
```

In [3]:

```
copd_race = copd[copd['StratificationCategoryID1'] == 'RACE']
```

In [4]:

```
copd_race = copd_race.rename(columns={"LocationAbbr": "STATE", "Topic": "NUM
```

In [5]:

```
copd_race = copd_race[['STATE','NUM OF COPD CASES', 'StratificationID1']].gr
```

In [6]:

```
state_populations.columns = ["Geographic Area", "April 1 2020", "2020", "202
state_populations = state_populations.iloc[8:-8].reset_index()
state_populations = state_populations[["Geographic Area", "2023"]]
```

In [7]:

```
state_populations['Geographic Area'] = state_populations['Geographic Area'].
state_populations['2023'] = state_populations['2023'].replace(',','',regex=T
state_populations['2023'] = pd.to_numeric(state_populations['2023'])
state_populations = state_populations.rename(columns={"Geographic Area": "St
```

In [8]:

```
states = pd.read_csv("states.csv")
```

In [9]:

```
states = states.rename(columns={"Abbreviation": "STATE"})
combined_gendata = pd.merge(states, copd_race, on="STATE", how="inner")
combined_df = pd.merge(combined_gendata, state_populations, on='State', how=
combined_df.head()
```

Out[9]:

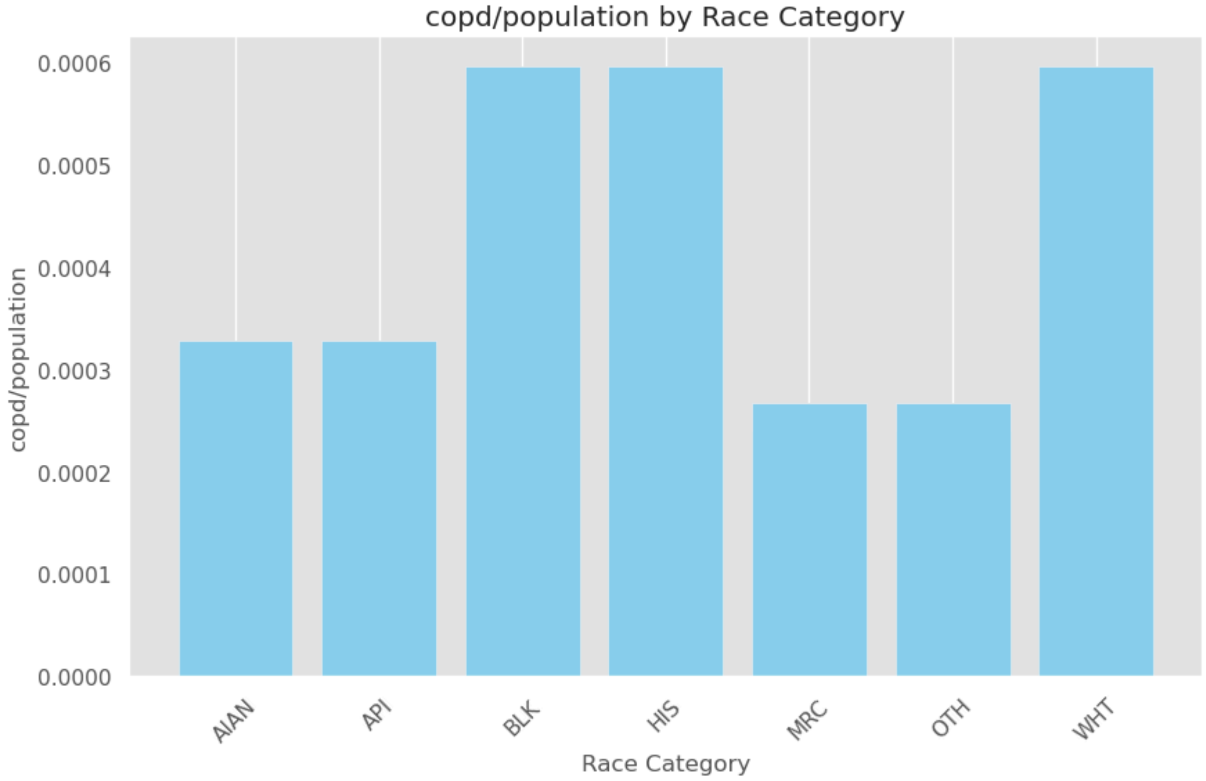
	State	STATE	StratificationID1	NUM OF COPD CASES	2023
0	Alabama	AL	AIAN	192	5108468
1	Alabama	AL	API	192	5108468
2	Alabama	AL	BLK	348	5108468
3	Alabama	AL	HIS	348	5108468
4	Alabama	AL	MRC	156	5108468

In [10]:

```
combined_df['copd/population'] = combined_df['NUM OF COPD CASES'] / combinec
```

In [77]:

```
plt.figure(figsize=(10, 6))
plt.bar(combined_df['StratificationID1'], combined_df['copd/population'], co
plt.xlabel('Race Category')
plt.ylabel('copd/population')
plt.title('copd/population by Race Category')
plt.xticks(rotation=45)  # Rotate x-axis labels for better readability
plt.grid(axis='y')  # Add gridlines on the y-axis
plt.show()
```



Describe trends.

Black and Hispanic show the highest copd/population percentage, which means that some of the variances between COPD cases by states could be explained by external factors like racial backgrounds, and thus, race should be considered as one of the variables used to construct our bayesian model.

How this visualization support our RQ.

There is a spike in the percentage for BLK, HIS, and WHT, which again would motivate us to include racial background as one of the RVs to estimate future COPD instances by state. This would also motivate us to include more exploration of the underlying distribution of COPDs by Race Category, which would then inform our decision to choose a particular distribution for our Bayesian model.