Sampling

The analysis of data on air quality with respect to carbon monoxide—a major air pollutant. The data utilized in this activity includes information from over 200 sites, identified by their state name, county name, city name, and local site name. You will use effective sampling within this dataset.

In [2]:

```
# Import libraries and packages

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.api as sm
from scipy import stats
```

In [3]:

```
# Load data

### YOUR CODE HERE ###

df = pd.read_csv("startdata1.csv", index_col = 0)
```

Data exploration

In [4]:
```
df.head(10)
```

Out[4]:

	date_local	state_name	county_name	city_name	local_site_name	parameter_name	units_of
0	2018-01-01	Arizona	Maricopa	Buckeye	BUCKEYE	Carbon monoxide	Parts
1	2018-01-01	Ohio	Belmont	Shadyside	Shadyside	Carbon monoxide	Parts
2	2018-01-01	Wyoming	Teton	Not in a city	Yellowstone National Park - Old Faithful Snow ...	Carbon monoxide	Parts
3	2018-01-01	Pennsylvania	Philadelphia	Philadelphia	North East Waste (NEW)	Carbon monoxide	Parts
4	2018-01-01	Iowa	Polk	Des Moines	CARPENTER	Carbon monoxide	Parts
5	2018-01-01	Hawaii	Honolulu	Not in a city	Kapolei	Carbon monoxide	Parts
6	2018-01-01	Hawaii	Honolulu	Not in a city	Kapolei	Carbon monoxide	Parts
7	2018-01-01	Pennsylvania	Erie	Erie	NaN	Carbon monoxide	Parts
8	2018-01-01	Hawaii	Honolulu	Honolulu	Honolulu	Carbon monoxide	Parts
9	2018-01-01	Colorado	Larimer	Fort Collins	Fort Collins - CSU - S. Mason	Carbon monoxide	Parts

In [5]:
```
df.describe(include = "all")  #descriptive statistics
```

Out[5]:

	date_local	state_name	county_name	city_name	local_site_name	parameter_name	units
count	260	260	260	260	257	260	
unique	1	52	149	190	253	1	
top	2018-01-01	California	Los Angeles	Not in a city	Kapolei	Carbon monoxide	P
freq	260	66	14	21	2	260	
mean	NaN	NaN	NaN	NaN	NaN	NaN	
std	NaN	NaN	NaN	NaN	NaN	NaN	
min	NaN	NaN	NaN	NaN	NaN	NaN	
25%	NaN	NaN	NaN	NaN	NaN	NaN	
50%	NaN	NaN	NaN	NaN	NaN	NaN	
75%	NaN	NaN	NaN	NaN	NaN	NaN	
max	NaN	NaN	NaN	NaN	NaN	NaN	

In [6]:
```
# The aqi(air quality index) column has mean of 6.757692 and count of 260
```

In [7]:
```
population_mean = df['aqi'].mean()
```

In [8]:
```
population_mean
```

Out[8]: 6.757692307692308

Sample with replacement

First, name a new variable sampled_data. Then, set the arguments for the sample function N, sample size, equal to 50. Set replace equal to “True” to specify sampling with replacement. For random_state, choose an arbitrary number for random seed. Make that arbitrary number 42.

In [9]:
```
sampled_data = df.sample(n=50, replace=True, random_state=42)
```

In [10]:
```
sampled_data.head(8) #    few first rows
```

Out[10]:

	date_local	state_name	county_name	city_name	local_site_name	parameter_name	units_c
102	2018-01-01	Texas	Harris	Houston	Clinton	Carbon monoxide	Parts
106	2018-01-01	California	Imperial	Calexico	Calexico-Ethel Street	Carbon monoxide	Parts
71	2018-01-01	Alabama	Jefferson	Birmingham	Arkadelphia/Near Road	Carbon monoxide	Parts
188	2018-01-01	Arizona	Maricopa	Tempe	Diablo	Carbon monoxide	Parts
20	2018-01-01	Virginia	Roanoke	Vinton	East Vinton Elementary School	Carbon monoxide	Parts
102	2018-01-01	Texas	Harris	Houston	Clinton	Carbon monoxide	Parts
121	2018-01-01	North Carolina	Mecklenburg	Charlotte	Garinger High School	Carbon monoxide	Parts
214	2018-01-01	Florida	Broward	Davie	Daniela Banu NCORE	Carbon monoxide	Parts

In [11]:
```
# observe repeatition
```

In [12]:
```
sample_mean = sampled_data['aqi'].mean() #   mean from the sample data
```

In [13]:
```
sample_mean
```

Out[13]: 5.54

In [14]:
```
# 	of course the sample mean is not the same as the population mean
```

Application of central limit theorem

The central limit theorem states that the mean of sampling distribution should be roughly equall to the population mean

In [15]:
```
estimate_list = []
for i in range(10000):
    ''' sampling distribution of 10,000 random samples with replacement'''
    estimate_list.append(df['aqi'].sample(n=50,replace=True).mean())
```

In [16]:
```
estimate_df = pd.DataFrame(data={'estimate': estimate_list}) # estimate_li
estimate_df
```

Out[16]:

	estimate
0	5.82
1	6.54
2	5.44
3	7.64
4	5.14
...	...
9995	6.50
9996	6.04
9997	9.68
9998	6.24
9999	6.54

10000 rows × 1 columns

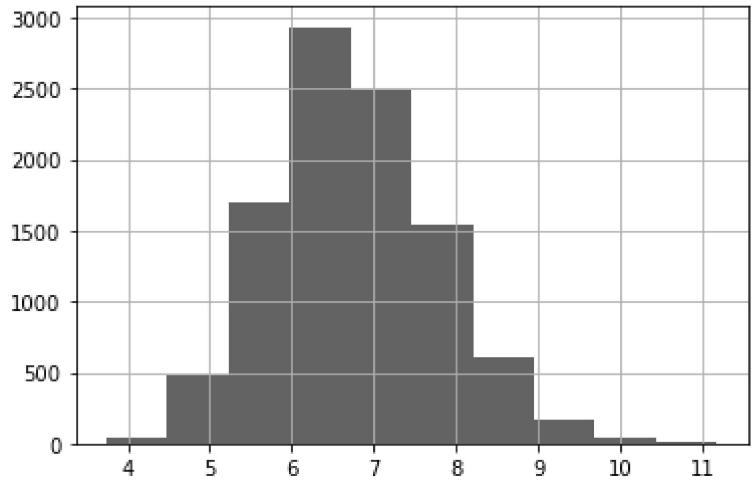
In [17]:
```
mean_sample_means = estimate_df['estimate'].mean() # 	mean of the sampli

mean_sample_means
```

Out[17]:
```
6.753288000000018
```

sampling distribution mean roughly = population mean

In [18]:
```
estimate_df['estimate'].hist();
```



Calculate the standard error

Standard error is the sampling distribution standard deviation

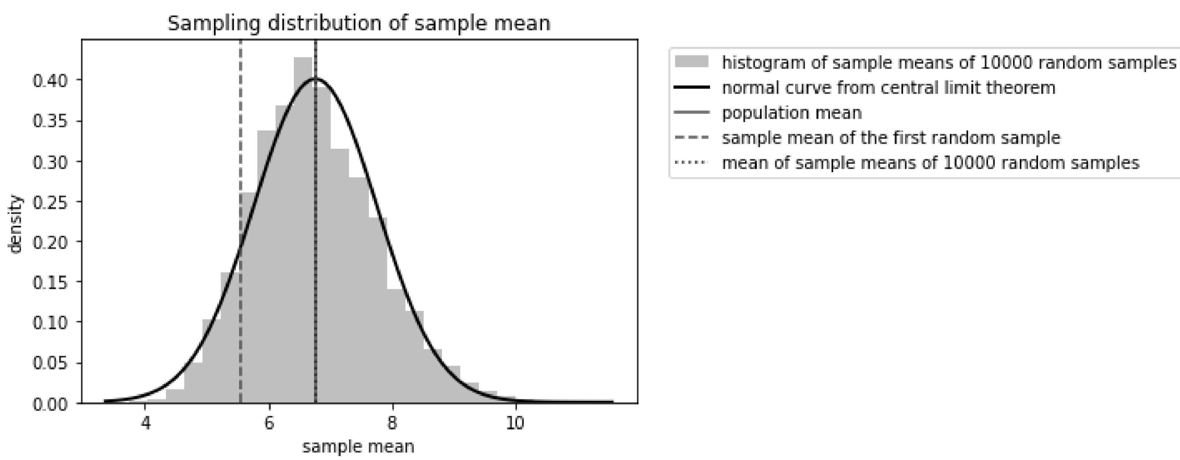
In [20]:
```
standard_error = estimate_df['estimate'].std()
standard_error
```

Out[20]: 0.9953641062577979

Relationship between ing and normal distibution

In [23]:
```
# Generate a grid of 100 values from xmin to xmax.


plt.hist(estimate_df['estimate'], bins=25, density=True, alpha=0.4, label
xmin, xmax = plt.xlim()
x = np.linspace(xmin, xmax, 100) # generate a grid of 100 values from xmin
p = stats.norm.pdf(x, population_mean, standard_error)
plt.plot(x, p, 'k', linewidth=2, label = 'normal curve from central limit
plt.axvline(x=population_mean, color='g', linestyle = 'solid', label = 'po
plt.axvline(x=sample_mean, color='r', linestyle = '--', label = 'sample me
plt.axvline(x=mean_sample_means, color='b', linestyle = ':', label = 'mean
plt.title("Sampling distribution of sample mean")
plt.xlabel('sample mean')
plt.ylabel('density')
plt.legend(bbox_to_anchor=(1.04,1));
```



Summary and conclusion

Sampling with replacement on a dataset leads to duplicate rows. Sample means are different from population means due to sampling variability. The central limit theorem helps describe the sampling distribution of the sample mean for many different types of datasets.

The mean AQI in a sample of 50 observations was below 100 in a statistically significant sense (at least 2–3 standard errors away). For reference, AQI values at or below 100 are generally thought of as satisfactory. This notebook didn't examine values outside the "satisfactory" range so analysis should be done to investigate unhealthy AQI values.

Carbon monoxide levels are satisfactory in general. Funding should be allocated to further investigate regions with unhealthy levels of carbon monoxide and improve the conditions in those regions.

In [ ]:
```

```