In [72]:
```python
# Import Library
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import statsmodels.api as sm
from scipy import stats
```

In [120…
```python
epa_data = pd.read_csv(r'C:\Users\HP\Desktop\Advance Data Analyst\4. The Power of Stats\3. Module 3\3. Work with sampling distribution\Files\c4_epa_air_quality.csv')
epa_data.head(10)
```

Out[120…

	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
5	5	2018-01-01	Hawaii	Honolulu	Not in a city	Kapolei	Carbon monoxide	Parts per million	0.994737	14
6	6	2018-01-01	Hawaii	Honolulu	Not in a city	Kapolei	Carbon monoxide	Parts per million	0.200000	2
7	7	2018-01-01	Pennsylvania	Erie	Erie	NaN	Carbon monoxide	Parts per million	0.200000	2
8	8	2018-01-01	Hawaii	Honolulu	Honolulu	Honolulu	Carbon monoxide	Parts per million	0.400000	5
9	9	2018-01-01	Colorado	Larimer	Fort Collins	Fort Collins - CSU - S. Mason	Carbon monoxide	Parts per million	0.300000	6

In [128…
```python
# Get descriptive stats.

epa_data.describe(include ='all')
```

Out[128…

	Unnamed: 0	date_local	state_name	county_name	city_name	local_site_name	parameter_name	units_of_measure	arithmetic_mean	aqi
count	260.000000	260	260	260	260	257	260	260	260.000000	260.000000
unique	NaN	1	52	149	190	253	1	1	NaN	NaN
top	NaN	2018-01-01	California	Los Angeles	Not in a city	Kapolei	Carbon monoxide	Parts per million	NaN	NaN
freq	NaN	260	66	14	21	2	260	260	NaN	NaN
mean	129.500000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.403169	6.757692
std	75.199734	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.317902	7.061707
min	0.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.000000	0.000000
25%	64.750000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.200000	2.000000
50%	129.500000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.276315	5.000000
75%	194.250000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.516009	9.000000
max	259.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.921053	50.000000

In [130…
```python
population_mean = epa_data['aqi'].mean()
population_mean
```

Out[130… 6.757692307692308

In [136…
```python
# Sample with replacement

sampled_data = epa_data.sample(n=50, replace=True, random_state=42)
sampled_data.head(10)
```

Out[136…

	Unnamed: 0	date_local	state_name	county_name	city_name	local_site_name	parameter_name	units_of_measure	arithmetic_mean	aqi
102	102	2018-01-01	Texas	Harris	Houston	Clinton	Carbon monoxide	Parts per million	0.157895	2
106	106	2018-01-01	California	Imperial	Calexico	Calexico-Ethel Street	Carbon monoxide	Parts per million	1.183333	26
71	71	2018-01-01	Alabama	Jefferson	Birmingham	Arkadelphia/Near Road	Carbon monoxide	Parts per million	0.200000	2
188	188	2018-01-01	Arizona	Maricopa	Tempe	Diablo	Carbon monoxide	Parts per million	0.542105	10
20	20	2018-01-01	Virginia	Roanoke	Vinton	East Vinton Elementary School	Carbon monoxide	Parts per million	0.100000	1
102	102	2018-01-01	Texas	Harris	Houston	Clinton	Carbon monoxide	Parts per million	0.157895	2
121	121	2018-01-01	North Carolina	Mecklenburg	Charlotte	Garinger High School	Carbon monoxide	Parts per million	0.200000	2
214	214	2018-01-01	Florida	Broward	Davie	Daniela Banu NCORE	Carbon monoxide	Parts per million	0.273684	5
87	87	2018-01-01	California	Humboldt	Eureka	Jacobs	Carbon monoxide	Parts per million	0.393750	5
99	99	2018-01-01	California	Santa Barbara	Goleta	Goleta	Carbon monoxide	Parts per million	0.222222	3

In [138…
```python
# Compute the mean value from the aqi column

sample_mean = sampled_data['aqi'].mean()
sample_mean
```

Out[138… 5.54

In [140…
```python
# Apply the central limit theorem

estimate_list = []
for i in range(10000):
    estimate_list.append(epa_data['aqi'].sample(n=50,replace=True).mean())
```

In [142…
```python
# Create a new DataFrame

estimate_df = pd.DataFrame(data={'estimate': estimate_list})
estimate_df
```

Out[142…

	estimate
0	7.38
1	6.74
2	6.98
3	6.96
4	6.78
...	...
9995	6.48
9996	8.04
9997	7.20
9998	7.74
9999	4.72

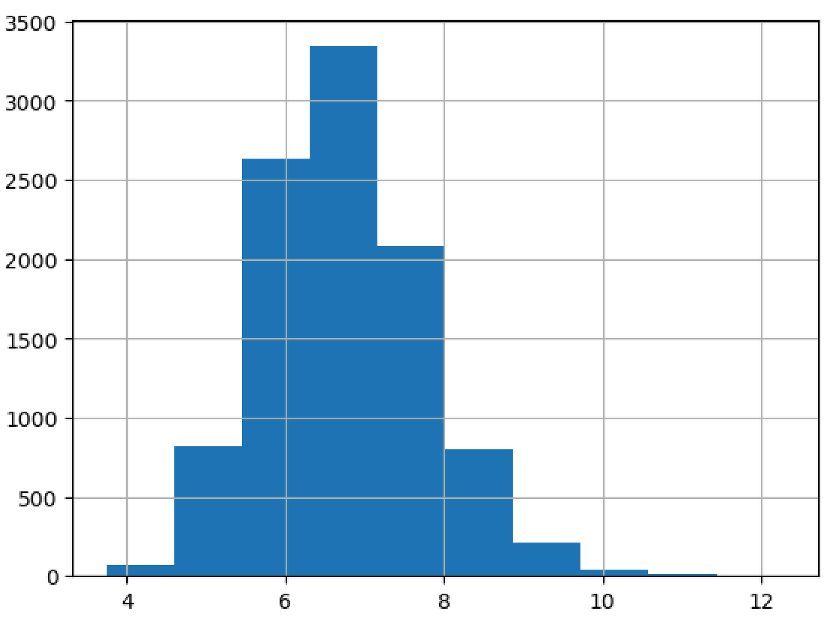
10000 rows × 1 columns

In [144…
```python
# Compute the mean() of the sampling distribution

mean_sample_means = estimate_df['estimate'].mean()
mean_sample_means
```

Out[144… 6.743504

In [146…
```python
# Output the distribution using a histogram

estimate_df['estimate'].hist()
```

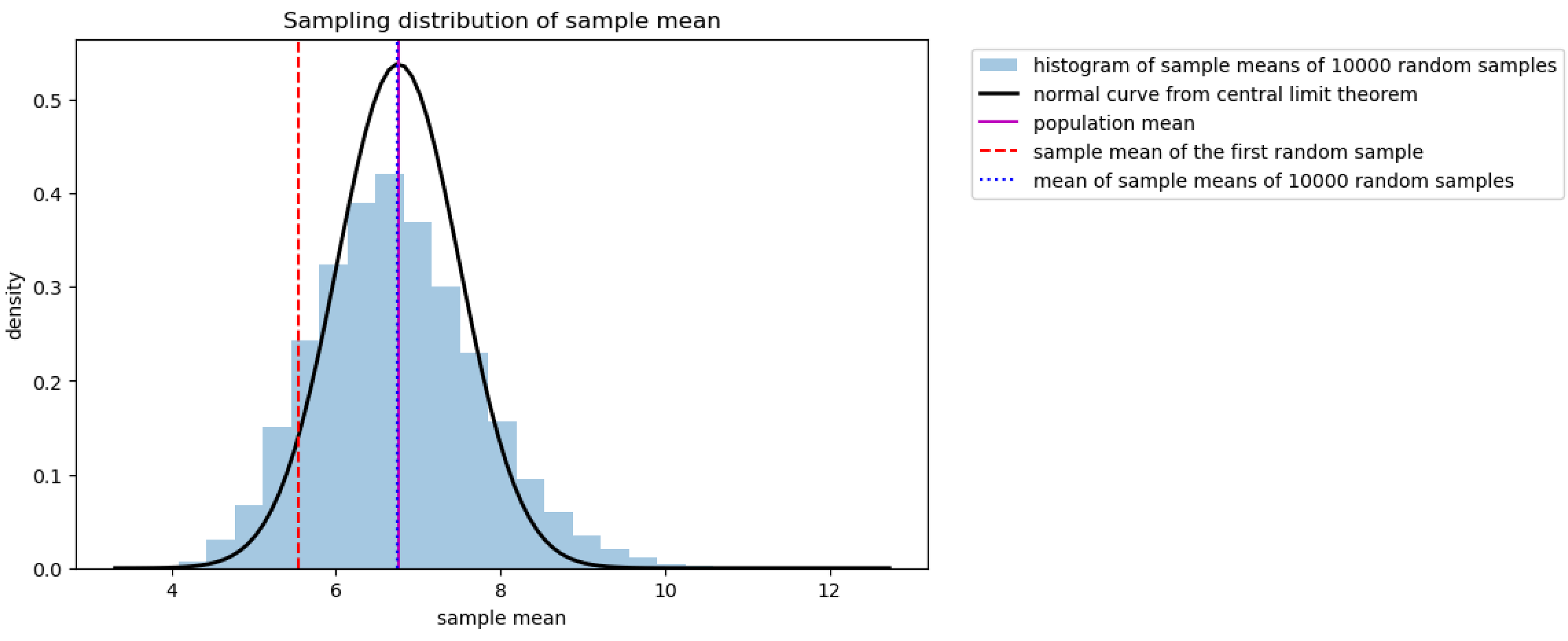
Out[146… <Axes: >



In [148…
```python
# Calculate the standard error

standard_error = sampled_data['aqi'].std() / np.sqrt(len(sampled_data))
standard_error
```

Out[148… 0.7413225908290327

In [150…
```python
# Results and evaluation

plt.figure(figsize=(8,5))
plt.hist(estimate_df['estimate'], bins=25, density=True, alpha=0.4, label = "histogram of sample means of 10000 random samples")
xmin, xmax = plt.xlim()
x = np.linspace(xmin, xmax, 100) # generate a grid of 100 values from xmin to xmax.
p = stats.norm.pdf(x, population_mean, standard_error)
plt.plot(x, p, 'k', linewidth=2, label = 'normal curve from central limit theorem')
plt.axvline(x=population_mean, color='m', linestyle = 'solid', label = 'population mean')
plt.axvline(x=sample_mean, color='r', linestyle = '--', label = 'sample mean of the first random sample')
plt.axvline(x=mean_sample_means, color='b', linestyle = ':', label = 'mean of sample means of 10000 random samples')
plt.title("Sampling distribution of sample mean")
plt.xlabel('sample mean')
plt.ylabel('density')
plt.legend(bbox_to_anchor=(1.04,1));
```



In [ ]:
```python
# some key takeaways that you learned

# 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.

# What findings would you share with others?

# 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.
```

```
# What would you convey to external stakeholders?

# 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.
```