

Imports

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import matplotlib
```

Load Dataset

The dataset is collected from Centers of Disease control and prevention data.cdc.gov under the category of Vaccination Coverage among Pregnant Women

```
# Read the data
data = pd.read_csv('Vaccination_Coverage_among_Pregnant_Women.csv')
```

data

```
Index(['Vaccine', 'Geography Type', 'Geography',
       'Survey Year/Influenza Season', 'Dimension Type', 'Dimension',
       'Estimate (%)', '95% CI (%)', 'Sample Size'],
      dtype='object')
```

```
# Display Values
data
```

Vaccine	Geography Type	Geography	Survey Year/Influenza Season	Dimension Type	Dimension	Estimate (%)	95% CI (%)	Sample Size
0 Influenza	States	Alaska	2012	Age	≥18 Years	49.2	45.3 to 53.1	852.0
1 Influenza	States	Arkansas	2012	Age	≥18 Years	46.6	40.7 to 52.5	756.0
2 Influenza	States	Colorado	2012	Age	≥18 Years	56.1	52.1 to 60.0	1170.0
3 Influenza	States	Delaware	2012	Age	≥18 Years	41.6	38.4 to 44.8	981.0
4 Influenza	States	Georgia	2012	Age	≥18 Years	33.6	29.6 to 37.7	1007.0
...
4132 Tdap	States	Utah	2020	Race/Ethnicity	White, Non-Hispanic	80.1	77.0 to 83.0	979.0
4133 Tdap	States	Vermont	2020	Race/Ethnicity	White, Non-Hispanic	86.4	83.6 to 88.9	696.0
4134 Tdap	States	Virginia	2020	Race/Ethnicity	White, Non-Hispanic	83.1	76.9 to 88.2	503.0
4135 Tdap	States	Washington	2020	Race/Ethnicity	White, Non-Hispanic	80.9	76.2 to 85.0	352.0
4136 Tdap	States	Wisconsin	2020	Race/Ethnicity	White, Non-Hispanic	82.8	78.4 to 86.7	364.0

4137 rows x 9 columns

```
# Display Dimension
data.shape
```

```
(4137, 9)
```

```
# Display Information
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4137 entries, 0 to 4136
Data columns (total 9 columns):
# Column Non-Null Count Dtype
---
0 Vaccine 4137 non-null object
1 Geography Type 4137 non-null object
2 Geography 4137 non-null object
3 Survey Year/Influenza Season 4137 non-null int64
4 Dimension Type 4137 non-null object
5 Dimension 4137 non-null object
6 Estimate (%) 4137 non-null object
7 95% CI (%) 4137 non-null object
8 Sample Size 3933 non-null float64
dtypes: float64(1), int64(1), object(7)
memory usage: 291.0+ KB
```

```
# Display the unique values
data['Vaccine'].value_counts()
```

```
Influenza 2891
Tdap 1246
Name: Vaccine, dtype: int64
```

```
# Display top 3 values
data.head(3)
```

Vaccine	Geography Type	Geography	Survey Year/Influenza Season	Dimension Type	Dimension	Estimate (%)	95% CI (%)	Sample Size
0 Influenza	States	Alaska	2012	Age	≥18 Years	49.2	45.3 to 53.1	852.0
1 Influenza	States	Arkansas	2012	Age	≥18 Years	46.6	40.7 to 52.5	756.0
2 Influenza	States	Colorado	2012	Age	≥18 Years	56.1	52.1 to 60.0	1170.0

```
data['Survey Year/Influenza Season'].unique()
```

```
array([2012, 2020, 2013, 2014, 2015, 2016, 2017, 2018, 2019])
```

```
# Make a new copy
new_data = data.copy()
```

```
# Display the presence of missing values
new_data.isnull().sum()
```

```
Vaccine 0
Geography Type 0
Geography 0
Survey Year/Influenza Season 0
Dimension Type 0
Dimension 0
Estimate (%) 0
95% CI (%) 0
Sample Size 204
dtype: int64
```

```
new_data.describe()
```

```
count 4137.000000 3933.000000
mean 2016.236602 805.390796
std 2.663058 2688.305261
min 2012.000000 30.000000
25% 2014.000000 169.000000
50% 2017.000000 325.000000
75% 2019.000000 659.000000
max 2020.000000 43737.000000
```

```
type(new_data['Estimate (%)'][0])
```

```
str
```

```
new_data['Estimate (%)'].unique()
```

```
array(['49.2%', '46.6%', '56.1%', '41.6%', '33.6%', '42.0%', '49.1%', '53.0%',
       '47.9%', '66.1%', '42.8%', '66.0%', '45.5%', '58.8%', '38.6%', '73.8%',
       '39.5%', '43.6%', '54.3%', '47.3%', '47.4%', '60.4%', '44.1%', '48.9%',
       '57.5%', '60.9%', '43.4%', '58.6%', '39.1%', '49.6%', '64.0%', '57.3%',
       '58.2%', '78.7%', '73.5%', '62.9%', '73.1%', '46.3%', '60.8%', '68.7%',
       '65.1%', '77.1%', '71.2%', '53.7%', '72.4%', '70.5%', '79.8%', '58.4%',
       '68.2%', '48.1%', '64.8%', '68.5%', '77.6%', '75.0%', '60.8%', '67.6%',
       '65.0%', '78.1%', '67.1%', '64.5%', '54.8%', '61.1%', '76.1%', '54.9%',
       '67.2%', '64.4%', '74.0%', '64.7%', '70.7%', '63.3%', '39.0%', '51.3%',
       '33.4%', '27.6%', '38.5%', '42.5%', '49.0%', '41.0%', '51.0%', '37.0%',
       '59.4%', '32.6%', '51.6%', '30.7%', '50.6%', '42.5%', '50.0%', '38.1%',
       '39.8%', '56.7%', '38.9%', '41.9%', '51.7%', '53.2%', '39.7%', '52.4%',
       '32.0%', '40.5%', '66.8%', '56.4%', '67.8%', '63.5%', '51.1%', '48.7%',
       '25.9%', '29.9%', '72.5%', '50.3%', '73.3%', '63.9%', '40.4%', '69.5%',
       '58.3%', '67.0%', '45.1%', 'NAN', '37.3%', '60.1%', '65.3%', '55.2%',
       '65.6%', '58.0%', '69.9%', '62.8%', '38.3%', '50.2%', '65.2%', '51.5%',
       '70.2%', '59.1%', '58.7%', '61.7%', '48.3%', '46.9%', '35.7%', '42.6%',
       '48.5%', '53.1%', '49.9%', '45.4%', '66.4%', '51.8%', '60.7%', '40.1%',
       '48.7%', '37.7%', '44.4%', '46.7%', '54.7%', '49.8%', '61.4%', '45.2%',
       '64.3%', '64.3%', '72.5%', '53.4%', '62.7%', '46.0%', '76.8%', '40.8%',
       '64.2%', '67.4%', '42.9%', '63.7%', '57.4%', '57.1%', '72.2%', '79.1%',
       '61.6%', '68.6%', '65.9%', '68.1%', '77.9%', '76.8%', '68.4', '67.5',
       '31.2%', '75.6%', '65.4', '73.2%', '73.8%', '65.5', '39.4', '40.6',
       '44.7%', '60.0%', '50.5', '68.1', '72.9%', '54.1', '40.7', '46.2',
       '45.6', '52.1', '62.5', '64.1', '57.2', '61.0', '82.0', '76.4',
       '71.1', '90.8', '50.9', '50.8', '71.6', '72.1', '63.7', '77.6',
       '59.8%', '75.1', '69.6', '85.8', '63.8', '57.6', '66.7', '75.9',
       '77.7', '68.8', '77.2', '65.4', '80.5', '66.9', '76.9', '78.5',
       '70.0', '28.3', '20.3', '37.5', '39.6', '63.9', '60.2', '40.4', '45.6',
       '50.1', '33.1', '26.5', '41.7', '48.8', '56.0', '36.1', '56.5',
       '28.2', '63.4', '64.3', '54.4', '45.3', '24.8', '23.3', '36.2',
       '62.6', '36.0', '52.9', '38.7', '53.8', '46.6', '55.8', '48.6',
       '77.0', '35.6', '45.9', '53.3', '30.6', '47.7', '35.0', '46.8',
       '41.4', '36.3', '51.9', '67.7', '59.9', '61.5', '57.0', '68.3',
       '90.9', '36.0', '51.9', '62.1', '53.8', '46.6', '55.8', '48.6',
       '77.0', '45.3', '40.2', '63.7', '37.6', '43.5', '47.0', '69.3',
       '61.3', '78.0', '83.9', '82.6', '81.3', '73.5', '70.8', '56.2',
       '43.3', '44.2', '29.8', '63.9', '52.2', '61.3', '71.3', '80.4', '54.5',
       '46.5', '55.3', '46.4', '49.5', '43.8', '59.3', '38.7', '74.9',
       '78.4', '35.1', '69.7', '79.6', '54.0', '80.8', '70.3', '53.9',
       '70.9', '76.6', '63.9', '77.8', '56.6', '72.1', '61.7', '60.2',
       '48.2', '43.2', '54.6', '72.7', '58.1', '62.4', '54.0', '65.7',
       '42.4', '61.8', '52.3', '65.8', '60.6', '55.9', '63.2', '55.5',
       '52.8', '54.8', '51.9', '59.6', '77.8', '64.6', '58.3', '55.4',
       '45.8', '52.9', '52.0', '56.3', '58.9', '47.5', '62.3', '73.7',
       '51.2', '75.5', '55.0', '59.0', '70.4', '70.1', '25.0', '75.3',
       '72.0', '40.3', '61.2', '55.7', '48.4', '70.6', '31.0', '74.3',
       '72.3', '48.0', '69.2', '56.8', '66.2', '73.9', '82.2', '14.7',
       '44.3', '60.3', '40.6', '43.0', '44.3', '42.4', '60.5', '39.2',
       '56.9', '36.5', '54.9', '47.6', '38.2', '75.1', '28.0', '32.8',
       '54.5', '36.6', '63.0', '33.5', '55.1', '30.5', '41.1', '35.8',
       '71.4', '29.3', '36.8', '36.9', '69.0', '44.8', '76.7', '55.8',
       '41.5', '25.3', '71.8', '62.1', '79.2', '60.1', '76.5', '73.6',
       '73.0', '67.9', '75.4', '62.2', '52.6', '52.7', '55.6', '75.2',
       '67.3', '63.1', '46.1', '63.6', '21.7', '68.9', '41.8', '78.1',
       '56.9', '71.7', '32.1', '61.3', '80.9', '67.1', '61.7', '77.6',
       '77.4', '74.5', '20.7', '44.9', '82.1', '49.3', '38.1', '78.7',
       '31.1', '83.3', '81.9', '78.2', '82.9', '62.0', '87.1', '78.8',
       '72.8', '16.6', '39.9', '40.0', '30.9', '75.7', '54.0', '48.6',
       '35.9', '83.5', '79.4', '38.0', '43.1', '24.7', '42.1', '27.8',
       '26.6', '38.4', '44.5', '66.6', '75.8', '74.2', '25.2', '86.8',
       '71.9', '76.2', '64.4', '42.3', '84.6', '80.9', '67.3', '81.5',
       '80.3', '81.8', '84.2', '78.6', '80.2', '29.6', '87.0', '91.5',
       '42.7', '79.5', '82.7', '20.5', '8.2', '15.5', '14.0', '35.8',
       '9.1', '18.8', '25.6', '31.3', '25.4', '47.1', '28.0', '32.8',
       '47.8', '84.1', '86.5', '86.0', '74.4', '81.2', '83.6', '88.6',
       '85.6', '83.2', '90.2', '86.9', '85.9', '86.3', '89.5', '85.1',
       '82.8', '78.9', '88.4', '86.1', '87.7', '80.9', '87.3', '80.4',
       '81.0', '86.2', '83.7', '21.6', '7.7', '12.1', '18.3', '14.5',
       '15.1', '25.5', '20.4', '25.4', '29.7', '39.3', '84.7', '86.7',
       '77.5', '91.6', '67.9', '77.8', '80.9', '80.9', '81.7', '80.4',
       '89.3', '87.4', '87.6', '89.7', '90.1', '18.7', '8.4', '15.9',
       '10.8', '18.0', '8.1', '14.3', '23.1', '26.8', '31.6', '27.0',
       '46.0', '84.0', '80.7', '83.4', '84.8', '85.1', '61.1', '90.0',
       '80.0', '88.0', '84.8', '84.3', '82.3', '89.1', '85.5', '89.0',
       '88.2', '92.1', '83.1', '24.9', '8.5', '19.0', '17.7', '15.7',
       '21.4', '13.8', '25.7', '31.3', '26.4', '71.1', '28.0', '32.8',
       '90.6', '87.5', '91.8', '82.4', '84.9', '87.8', '89.6', '88.6',
       '11.2', '21.8', '31.1', '29.0', '19.9', '32.7', '79.0', '18.5',
       '22.6', '22.9', '30.3', '32.2', '85.7', '88.7', '86.4', '91.3',
       '79.9', '85.4', '14.9', '23.6', '80.1', '93.7', '82.5', '93.0',
       '98.2', '88.8', '18.6', '5.2', '13.7', '13.6', '13.4', '26.2',
       '27.9', '31.4', '47.1', '91.0', '91.4', '78.3', '89.8',
       ],
      dtype=object)
```

```
# There is some unwanted values so replacing it with NaN for easy cleaning of data
new_data['Estimate (%)'] = new_data['Estimate (%)'].replace('NAN', 'NaN')
```

```
new_data['Estimate (%)'].unique()
```

```
array(['49.2%', '46.6%', '56.1%', '41.6%', '33.6%', '42.0%', '49.1%', '53.0%',
       '47.9%', '66.1%', '42.8%', '66.0%', '45.5%', '58.8%', '38.6%', '73.8%',
       '39.5%', '43.6%', '54.3%', '47.3%', '47.4%', '60.4%', '44.1%', '48.9%',
       '57.5%', '60.9%', '43.4%', '58.6%', '39.1%', '49.6%', '64.0%', '57.3%',
       '58.2%', '78.7%', '73.5%', '62.9%', '73.1%', '46.3%', '60.8%', '68.7%',
       '65.1%', '77.1%', '71.2%', '53.7%', '72.4%', '70.5%', '79.8%', '58.4%',
       '68.2%', '48.1%', '64.8%', '68.5%', '77.6%', '75.0%', '60.8%', '67.6%',
       '65.0%', '78.1%', '67.1%', '64.5%', '54.8%', '61.1%', '76.1%', '54.9%',
       '67.2%', '64.4%', '74.0%', '64.7%', '70.7%', '63.3%', '39.0%', '51.3%',
       '33.4%', '27.6%', '38.5%', '42.5%', '49.0%', '41.0%', '51.0%', '37.0%',
       '59.4%', '32.6%', '51.6%', '30.7%', '50.6%', '42.5%', '50.0%', '38.1%',
       '39.8%', '56.7%', '38.9%', '41.9%', '51.7%', '53.2%', '39.7%', '52.4%',
       '32.0%', '40.5%', '66.8%', '56.4%', '67.8%', '63.5%', '51.1%', '48.7%',
       '25.9%', '29.9%', '72.5%', '50.3%', '73.3%', '63.9%', '40.4%', '69.5%',
       '58.3%', '67.0%', '45.1%', 'NAN', '37.3%', '60.1%', '65.3%', '55.2%',
       '65.6%', '58.0%', '69.9%', '62.8%', '38.3%', '50.2%', '65.2%', '51.5%',
       '70.2%', '59.1%', '58.7%', '61.7%', '48.3%', '46.9%', '35.7%', '42.6%',
       '48.5%', '53.1%', '49.9%', '45.4', '66.4', '51.8', '60.7', '40.1',
       '48.7%', '37.7', '44.4', '46.7', '54.7', '49.8', '61.4', '45.2',
       '64.3', '64.3', '72.5%', '53.4%', '62.7%', '46.0%', '76.8%', '40.8%',
       '64.2%', '67.4', '42.9%', '63.7', '57.4', '57.1', '72.2', '79.1',
       '61.6', '68.6', '65.9', '68.1', '77.9', '76.8', '68.4', '67.5',
       '31.2%', '75.6%', '65.4', '73.2%', '73.8%', '65.5', '39.4', '40.6',
       '44.7%', '60.0', '50.5', '68.1', '72.9%', '54.1', '40.7', '46.2',
       '45.6', '52.1', '62.5', '64.1', '57.2', '61.0', '82.0', '76.4',
       '71.1', '90.8', '50.9', '50.8', '71.6', '72.1', '63.7', '77.6',
       '59.8%', '75.1', '69.6', '85.8', '63.8', '57.6', '66.7', '75.9',
       '77.7', '68.8', '77.2', '65.4', '80.5', '66.9', '76.9', '78.5',
       '70.0', '28.3', '20.3', '37.5', '39.6', '63.9', '60.2', '40.4', '45.6',
       '50.1', '33.1', '26.5', '41.7', '48.8', '56.0', '36.1', '56.5',
       '28.2', '63.4', '64.3', '54.4', '45.3', '24.8', '23.3', '36.2',
       '62.6', '36.0', '52.9', '38.7', '53.8', '46.6', '55.8', '48.6',
       '77.0', '35.6', '45.9', '53.3', '30.6', '47.7', '35.0', '46.8',
       '41.4', '36.3', '51.9', '67.7', '59.9', '61.5', '57.0', '68.3',
       '90.9', '36.0', '51.9', '62.1', '53.8', '46.6', '55.8', '48.6',
       '77.0', '45.3', '40.2', '63.7', '37.6', '43.5', '47.0', '69.3',
       '61.3', '78.0', '83.9', '82.6', '81.3', '73.5', '70.8', '56.2',
       '43.3', '44.2', '29.8', '63.9', '52.2', '61.3', '71.3', '80.4', '54.5',
       '46.5', '55.3', '46.4', '49.5', '43.8', '59.3', '38.7', '74.9',
       '78.4', '35.1', '69.7', '79.6', '54.0', '80.8', '70.3', '53.9',
       '70.9', '76.6', '63.9', '77.8', '56.6', '72.1', '61.7', '60.2',
       '48.2', '43.2', '54.6', '72.7', '58.1', '62.4', '54.0', '65.7',
       '42.4', '61.8', '52.3', '65.8', '60.6', '55.9', '63.2', '55.5',
       '52.8', '54.8', '51.9', '59.6', '77.8', '64.6', '58.3', '55.4',
       '45.8', '52.9', '52.0', '56.3', '58.9', '47.5', '62.3', '73.7',
       '51.2', '75.5', '55.0', '59.0', '70.4', '70.1', '25.0', '75.3',
       '72.0', '40.3', '61.2', '55.7', '48.4', '70.6', '31.0', '74.3',
       '72.3', '48.0', '69.2', '56.8', '66.2', '73.9', '82.2', '14.7',
       '44.3', '60.3', '40.6', '43.0', '44.3', '42.4', '60.5', '39.2',
       '56.9', '36.5', '54.9', '47.6', '38.2', '75.1', '28.0', '32.8',
       '54.5', '36.6', '63.0', '33.5', '55.1', '30.5', '41.1', '35.8',
       '71.4', '29.3', '36.8', '36.9', '69.0', '44.8', '76.7', '55.8',
       '41.5', '25.3', '71.8', '62.1', '79.2', '60.1', '76.5', '73.6',
       '73.0', '67.9', '75.4', '62.2', '52.6', '52.7', '55.6', '75.2',
       '67.3', '63.1', '46.1', '63.6', '21.7', '68.9', '41.8', '78.1',
       '56.9', '71.7', '32.1', '61.3', '80.9', '67.1', '61.7', '77.6',
       '77.4', '74.5', '20.7', '44.9', '82.1', '49.3', '38.1', '78.7',
       '31.1', '83.3', '81.9', '78.2', '82.9', '62.0', '87.1', '78.8',
       '72.8', '16.6', '39.9', '40.0', '30.9', '75.7', '54.0', '48.6',
       '35.9', '83.5', '79.4', '38.0', '43.1', '24.7', '42.1', '27.8',
       '26.6', '38.4', '44.5', '66.6', '75.8', '74.2', '25.2', '86.8',
       '71.9', '76.2', '64.4', '42.3', '84.6', '80.9', '67.3', '81.5',
       '80.3', '81.8', '84.2', '78.6', '80.2', '29.6', '87.0', '91.5',
       '42.7', '79.5', '82.7', '20.5', '8.2', '15.5', '14.0', '35.8',
       '9.1', '18.8', '25.6', '31.3', '25.4', '47.1', '28.0', '32.8',
       '47.8', '84.1', '86.5', '86.0', '74.4', '81.2', '83.6', '88.6',
       '85.6', '83.2', '90.2', '86.9', '85.9', '86.3', '89.5', '85.1',
       '82.8', '78.9', '88.4', '86.1', '87.7', '80.9', '87.3', '80.4',
       '81.0', '86.2', '83.7', '21.6', '7.7', '12.1', '18.3', '14.5',
       '15.1', '25.5', '20.4', '25.4', '29.7', '39.3', '84.7', '86.7',
       '77.5', '91.6', '67.9', '77.8', '80.9', '80.9', '81.7', '80.4',
       '89.3', '87.4', '87.6', '89.7', '90.1', '18.7', '8.4', '15.9',
       '10.8', '18.0', '8.1', '14.3', '23.1', '26.8', '31.6', '27.0',
       '46.0', '84.0', '80.7', '83.4', '84.8', '85.1', '61.1', '90.0',
       '80.0', '88.0', '84.8', '84.3', '82.3', '89.1', '85.5', '89.0',
       '88.2', '92.1', '83.1', '24.9', '8.5', '19.0', '17.7', '15.7',
       '21.4', '13.8', '25.7', '31.3', '26.4', '71.1', '28.0', '32.8',
       '90.6', '87.5', '91.8', '82.4', '84.9', '87.8', '89.6', '88.6',
       '11.2', '21.8', '31.1', '29.0', '19.9', '32.7', '79.0', '18.5',
       '22.6', '22.9', '30.3', '32.2', '85.7', '88.7', '86.4', '91.3',
       '79.9', '85.4', '14.9', '23.6', '80.1', '93.7', '82.5', '93.0',
       '98.2', '88.8', '18.6', '5.2', '13.7', '13.6', '13.4', '26.2',
       '27.9', '31.4', '47.1', '91.0', '91.4', '78.3', '89.8',
       ],
      dtype=object)
```

```
new_data['Estimate (%)'] = new_data['Estimate (%)'].dropna()
```

```
# The column is in str so typecasting it to float for numerical computations
new_data['Estimate (%)'] = new_data['Estimate (%)'].astype(float)
```

```
new_data['Estimate (%)'][0]
```

```
49.2
```

```
new_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4137 entries, 0 to 4136
Data columns (total 9 columns):
# Column Non-Null Count Dtype
---
0 Vaccine 4137 non-null object
1 Geography Type 4137 non-null object
2 Geography 4137 non-null object
3 Survey Year/Influenza Season 4137 non-null int64
4 Dimension Type 4137 non-null object
5 Dimension 4137 non-null object
6 Estimate (%) 3715 non-null float64
7 95% CI (%) 4137 non-null object
8 Sample Size 3933 non-null float64
dtypes: float64(2), int64(1), object(6)
memory usage: 291.0+ KB
```

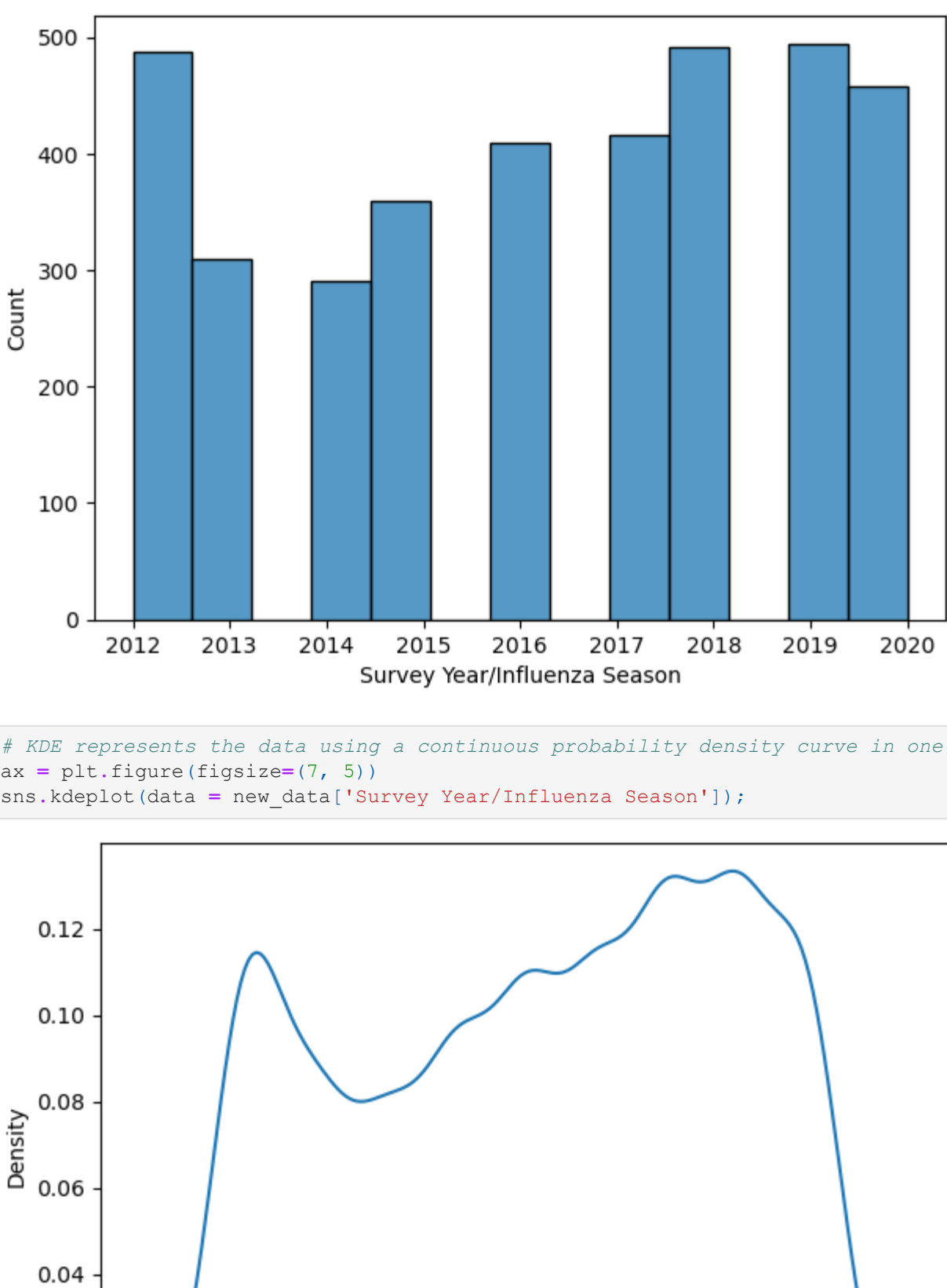
```
# Dropping all the null values
new_data = new_data.dropna()
```

```
new_data.info()
```

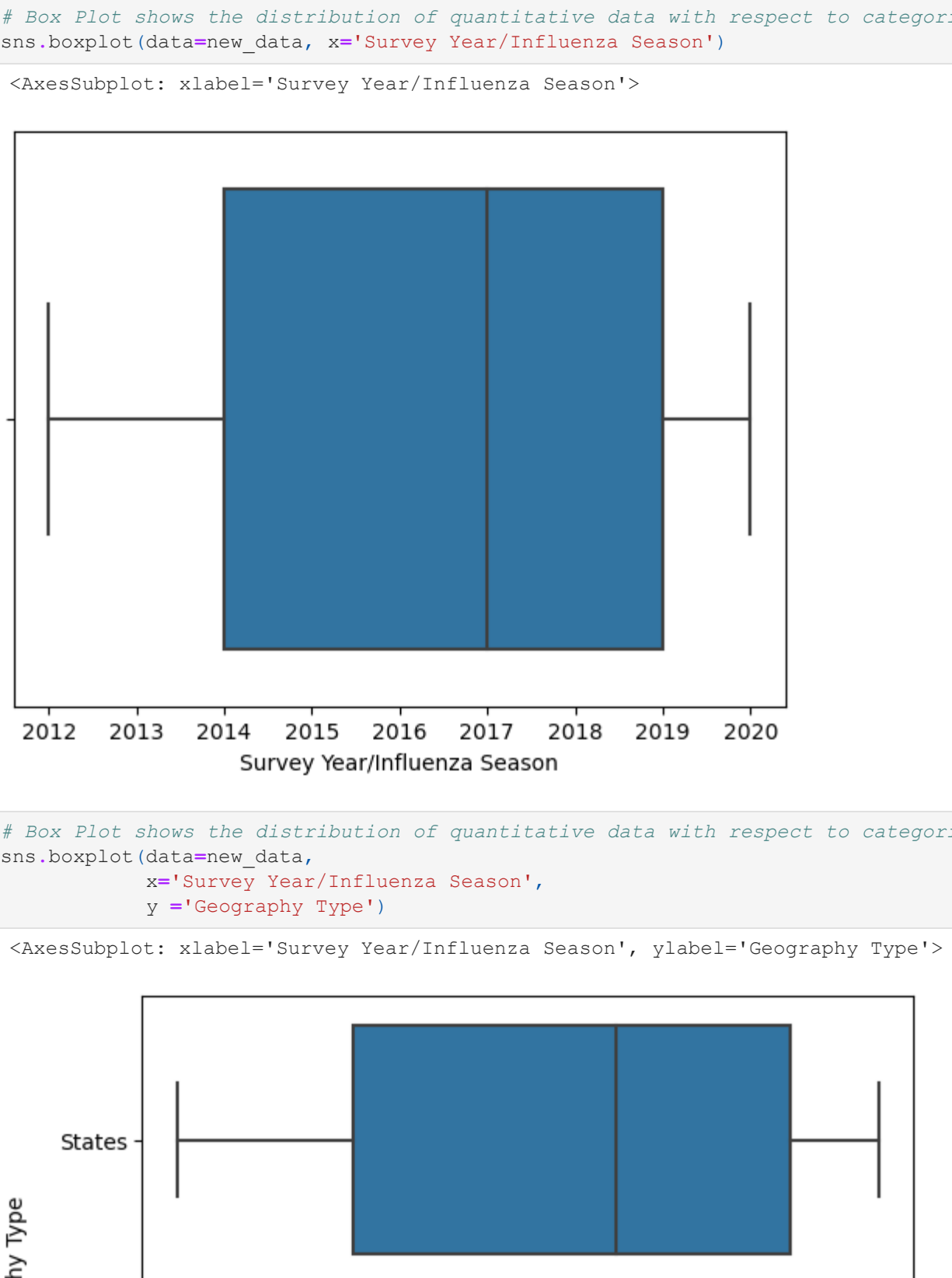
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3715 entries, 0 to 4136
Data columns (total 9 columns):
# Column Non-Null Count Dtype
---
0 Vaccine 3715 non-null object
1 Geography Type 3715 non-null object
2 Geography 3715 non-null object
3 Survey Year/Influenza Season 3715 non-null int64
4 Dimension Type 3715 non-null object
5 Dimension 3715 non-null object
6 Estimate (%) 3715 non-null float64
7 95% CI (%) 3715 non-null object
8 Sample Size 3715 non-null float64
dtypes: float64(2), int64(1), object(6)
memory usage: 281.2+ KB
```

```
new_data.describe()
```

```
count 3715.000000 3715.000000 3715.000000
mean 2016.2
```

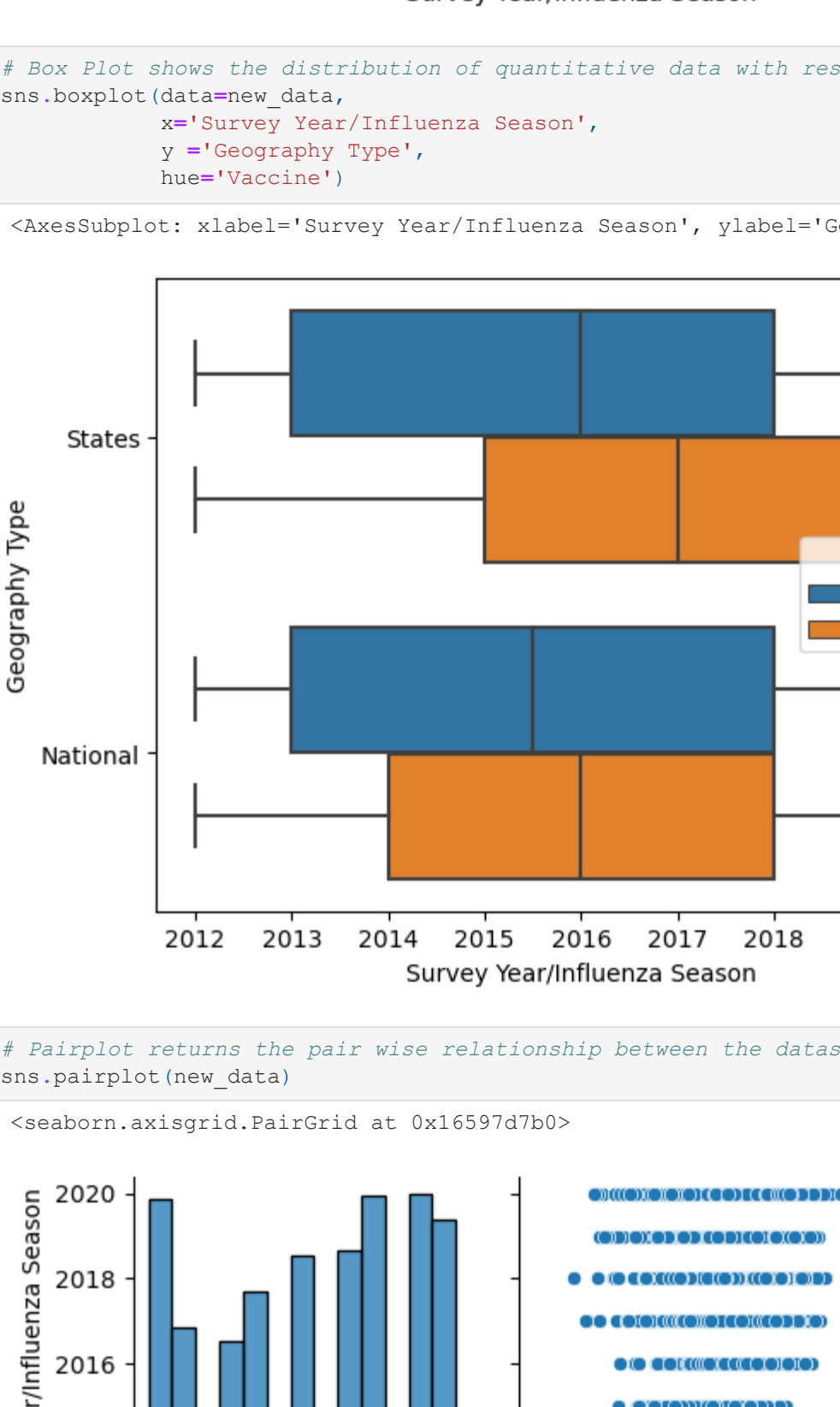



```
In [174]: # KDE represents the data using a continuous probability density curve in one or more dimensions.
ax = plt.figure(figsize=(7, 5))
sns.kdeplot(data = new_data['Survey Year/Influenza Season']);
```



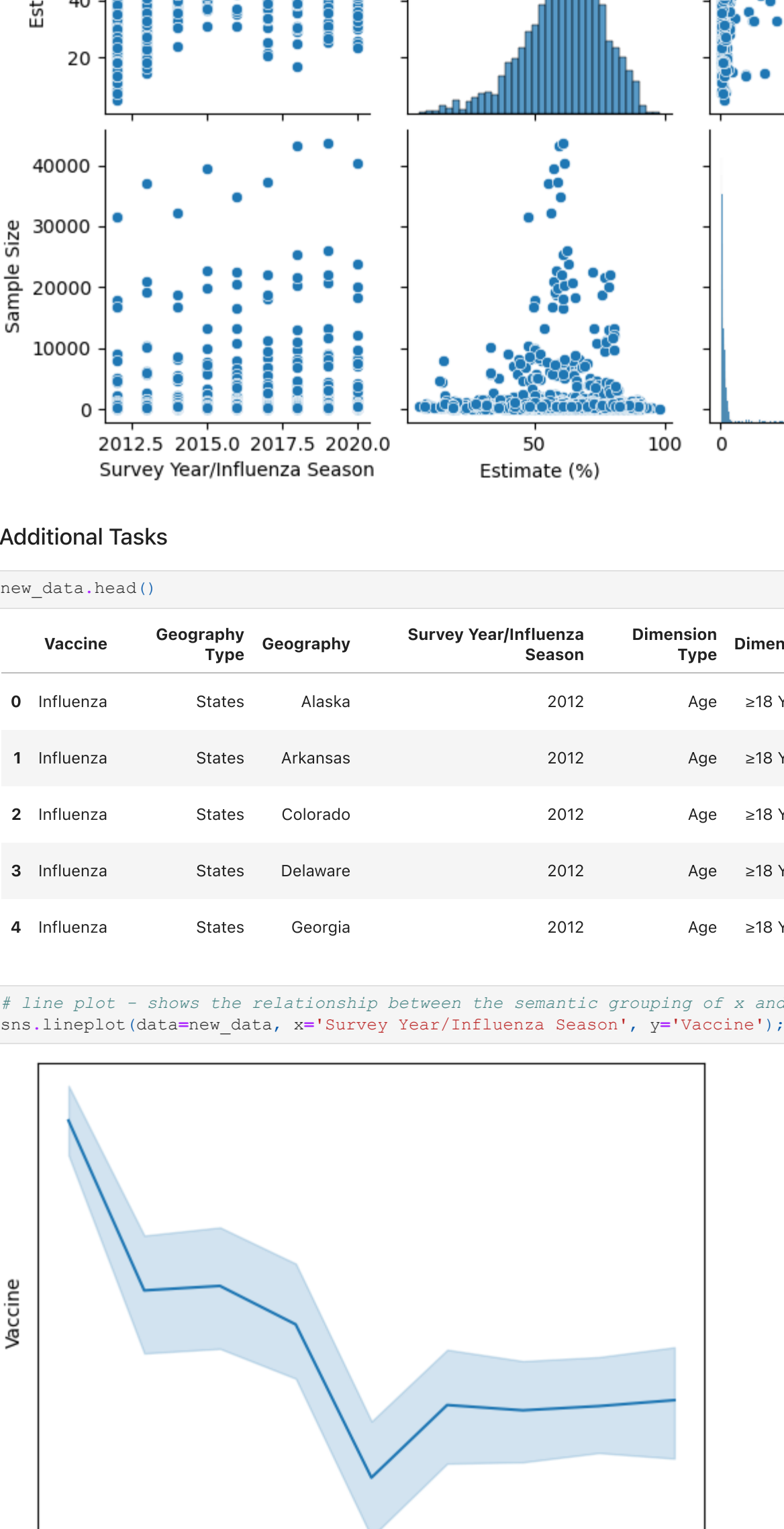
```
In [175]: # Box Plot shows the distribution of quantitative data with respect to categories
sns.boxplot(data=new_data, x='Survey Year/Influenza Season',
            y='Geography Type');
```

```
Out[175]: <AxesSubplot: xlabel='Survey Year/Influenza Season', ylabel='Geography Type'>
```



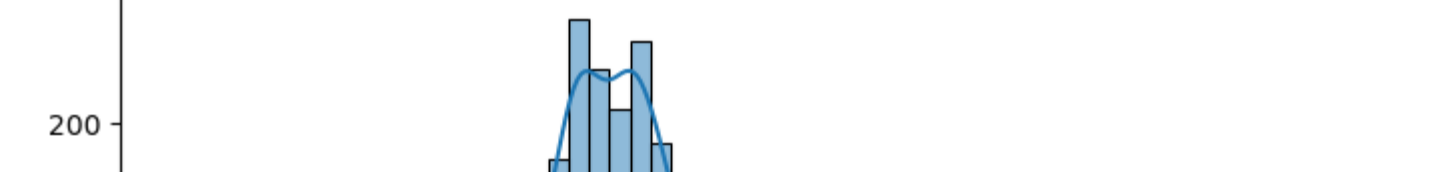
```
In [176]: # Box Plot shows the distribution of quantitative data with respect to categories
sns.boxplot(data=new_data,
            x='Survey Year/Influenza Season',
            y='Geography Type',
            hue='Vaccine');
```

```
Out[176]: <AxesSubplot: xlabel='Survey Year/Influenza Season', ylabel='Geography Type'>
```



```
In [178]: # Pairplot returns the pair wise relationship between the dataset
sns.pairplot(new_data)
```

```
Out[178]: <seaborn.axisgrid.PairGrid at 0x16597d7b0>
```

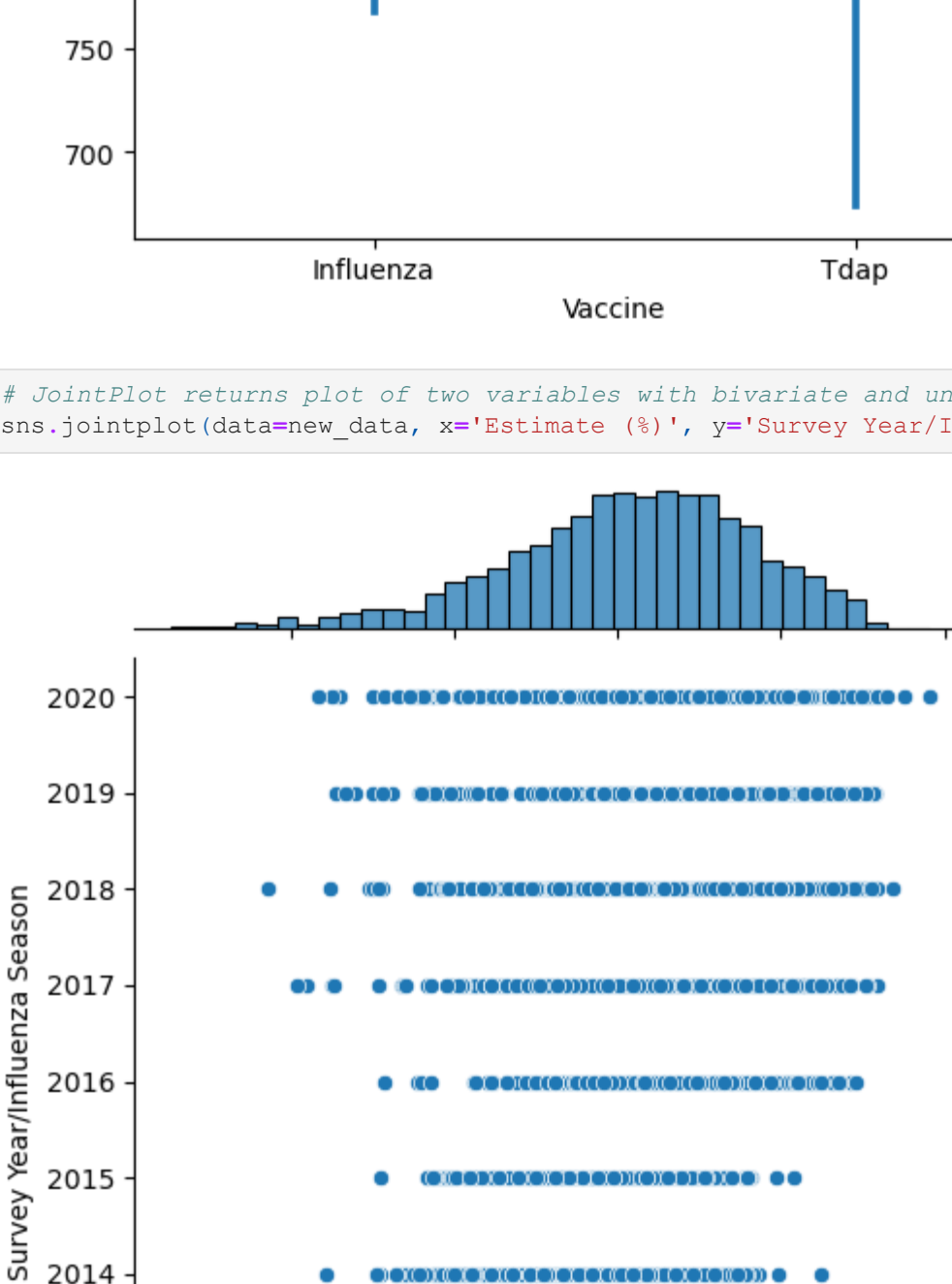


Additional Tasks

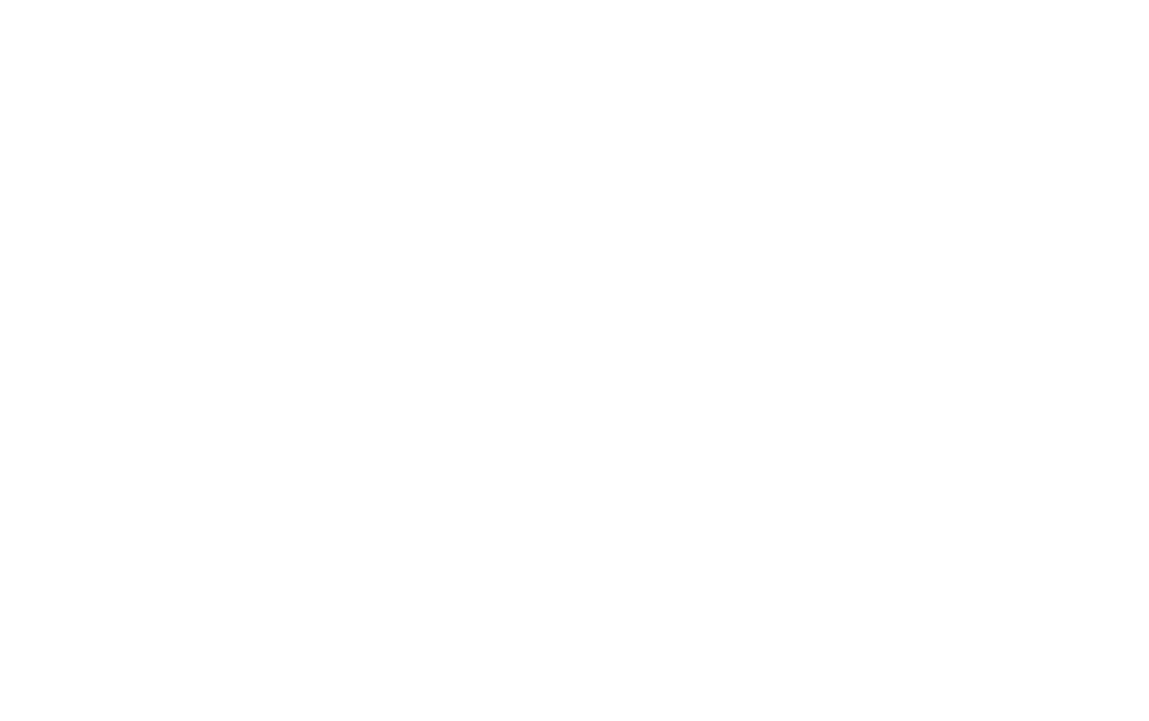
```
In [179]: new_data.head()
```

	Vaccine	Geography Type	Geography	Survey Year/Influenza Season	Dimension Type	Dimension	Estimate (%)	95% CI (%)	Sample Size
0	Influenza	States	Alaska	2012	Age	≥18 Years	49.2	45.3 to 53.1	852.0
1	Influenza	States	Arkansas	2012	Age	≥18 Years	46.6	40.7 to 52.5	766.0
2	Influenza	States	Colorado	2012	Age	≥18 Years	56.1	52.1 to 60.0	1170.0
3	Influenza	States	Delaware	2012	Age	≥18 Years	41.6	38.4 to 44.8	981.0
4	Influenza	States	Georgia	2012	Age	≥18 Years	33.6	29.6 to 37.7	1007.0

```
In [186]: # Line plot shows the relationship between the semantic grouping of x and y
sns.lineplot(data=new_data, x='Survey Year/Influenza Season', y='Vaccine');
```



```
In [181]: # Returns the distribution plot over the Grid of the list
sns.displot(data=new_data, x='Estimate (%)', kde=True, hue='Vaccine');
```



```
In [186]: # A point plot represents an estimate of central tendency for a numeric variable by the position of the dot and
sns.pointplot(data=new_data, y='Sample Size', x='Vaccine');
```



```
In [195]: # JointPlot returns plot of two variables with bivariate and univariate graphs.
sns.jointplot(data=new_data, x='Estimate (%)', y='Survey Year/Influenza Season');
```

