M2M TECH
DataTalent Program

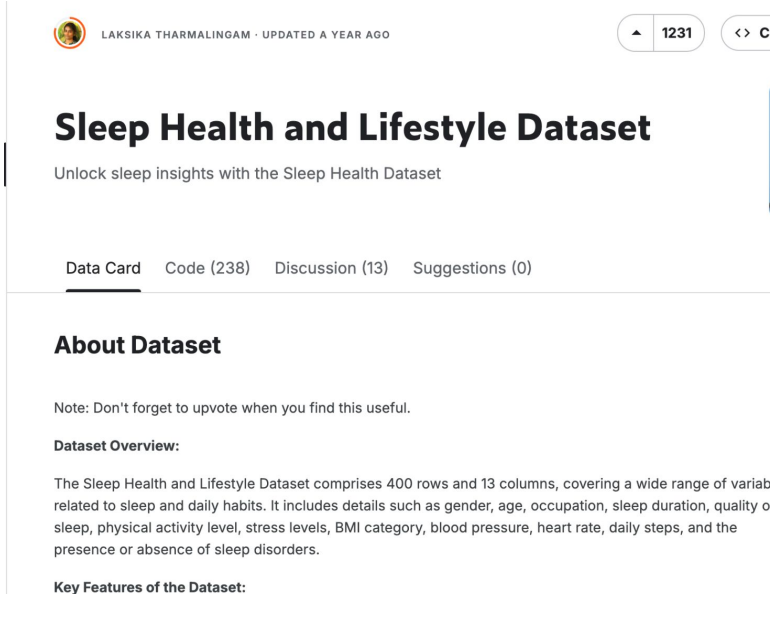
Sleep Health and Lifestyle

CP 1 and 2

Veronika Ulianova, March 17, 2025


SLEEP HEALTH
"Sleep is one of the most important factors for a healthy body and mind. Get enough restful nights, and you will be able to make the most out of every day."
https://www.sleephealth.org/

Data



Importing and Displaying Data

```
[ ]  import pandas as pd
     data = pd.read_csv('Sleep_health_and_lifestyle_dataset.csv')
```

Check information about the DataFrame including the index dtype and columns, non-null values and memory usage.

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 374 entries, 0 to 373
Data columns (total 13 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Person ID                374 non-null    int64  
 1   Gender                   374 non-null    object 
 2   Age                      374 non-null    int64  
 3   Occupation               374 non-null    object 
 4   Sleep Duration           374 non-null    float64
 5   Quality of Sleep         374 non-null    int64  
 6   Physical Activity Level  374 non-null    int64  
 7   Stress Level             374 non-null    int64  
 8   BMI Category             374 non-null    object 
 9   Blood Pressure           374 non-null    object 
 10  Heart Rate               374 non-null    int64  
 11  Daily Steps              374 non-null    int64  
 12  Sleep Disorder           155 non-null    object 
dtypes: float64(1), int64(7), object(5)
memory usage: 38.1+ KB
```

Understanding the gender, age and occupation data

```
[ ] genders = data['Gender'].value_counts()
    print(genders)
```

```
Gender
Male      189
Female    185
Name: count, dtype: int64
```

The dataset represents a nearly balanced gender distribution, with 189 males and 185 females

```
data['Age'].describe()
```

	Age
count	374.000000
mean	42.184492
std	8.673133
min	27.000000
25%	35.250000
50%	43.000000
75%	50.000000
max	59.000000

dtype: float64

```
data['Age'].value_counts(bins=5, sort=False)
```

Age	count
(25, 30]	32
(30, 35]	62
(35, 40]	71
(40, 45]	99
(45, 50]	34
(50, 55]	43
(55, 60]	33

dtype: int64

```
data.groupby(['Occupation','Gender']).size()
```

Occupation	Gender	0
Accountant	Female	36
	Male	1
Doctor	Female	2
	Male	69
Engineer	Female	32
	Male	31
Lawyer	Female	2
	Male	45
Manager	Female	1
Nurse	Female	73
Sales Representative	Male	2
Salesperson	Male	32
Scientist	Female	4
Software Engineer	Male	4
Teacher	Female	35
	Male	5

- nearly balanced gender distribution
- a majority of participants in the 30-50 years age range.
- nurses, teachers, and accountants are predominantly female, while doctors, lawyers, and salespersons are largely male.
- engineers show more gender balance, scientists, software engineers, sales representatives and managers are underrepresented.

Assumption to exclude underrepresented groups

```
[ ] data = data[~data['Occupation'].isin(['Scientist', 'Software Engineer', 'Sales Representative', 'Manager'])]

    print(data.info())
    print(data['Occupation'].value_counts())
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 363 entries, 1 to 373
Data columns (total 13 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Person ID                363 non-null    int64  
 1   Gender                   363 non-null    object 
 2   Age                      363 non-null    int64  
 3   Occupation               363 non-null    object 
 4   Sleep Duration           363 non-null    float64
 5   Quality of Sleep         363 non-null    int64  
 6   Physical Activity Level  363 non-null    int64  
 7   Stress Level             363 non-null    int64  
 8   BMI Category             363 non-null    object 
 9   Blood Pressure           363 non-null    object 
 10  Heart Rate               363 non-null    int64  
 11  Daily Steps              363 non-null    int64  
 12  Sleep Disorder           363 non-null    object 
dtypes: float64(1), int64(7), object(5)
memory usage: 39.7+ KB
None
Occupation
Nurse          73
Doctor         71
Engineer       63
Lawyer         47
Teacher        40
Accountant     37
Salesperson    32
Name: count, dtype: int64
```

Analysis of the sleep parameters and lifestyle for each occupation

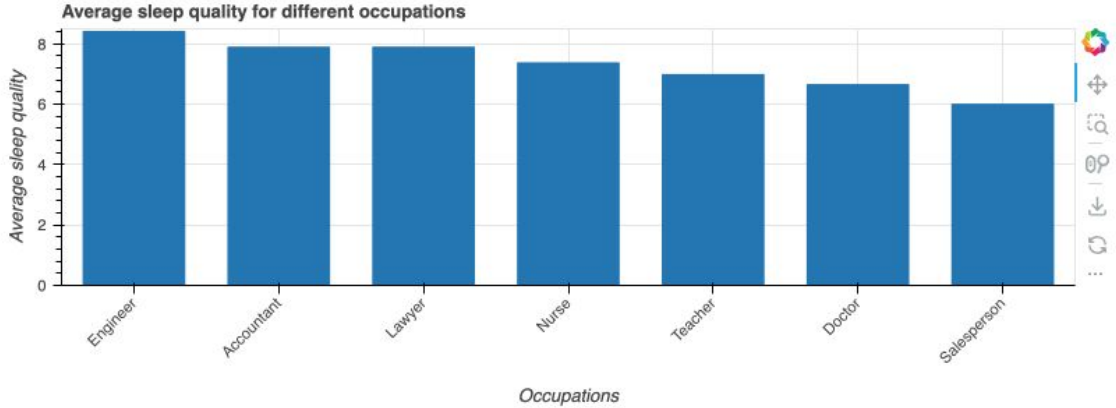
```
groups_occupation = data.groupby(['Occupation'])


avg_sleep_duration = round(groups_occupation['Sleep Duration'].mean() ,2)
avg_sleep_quality = round(groups_occupation['Quality of Sleep'].mean(), 2)

avg_activity = round(groups_occupation['Physical Activity Level'].mean(), 0)
avg_stress = round(groups_occupation['Stress Level'].mean(), 0)

avg_HR = round(groups_occupation['Heart Rate'].mean(), 0)
avg_steps = round(groups_occupation['Daily Steps'].mean(), 0)


df_avg = pd.DataFrame({
    'Average Sleep Duration': avg_sleep_duration,
    'Average Sleep Quality': avg_sleep_quality,
    'Average Physical Activity Level': avg_activity,
    'Average Stress Level': avg_stress,
    'Average Heart Rate': avg_HR,
    'Average Daily Steps': avg_steps
})

df_avg = df_avg.sort_values(by = 'Average Sleep Quality', ascending=False)
df_avg
```

Occupation	Average Sleep Duration	Average Sleep Quality	Average Physical Activity Level	Average Stress Level	Average Heart Rate	Average Daily Steps
Engineer	7.99	8.41	52.0	4.0	67.0	5981.0
Accountant	7.11	7.89	58.0	5.0	69.0	6881.0
Lawyer	7.41	7.89	70.0	5.0	70.0	7662.0
Nurse	7.06	7.37	79.0	6.0	72.0	8058.0
Teacher	6.69	6.98	46.0	5.0	67.0	5958.0
Doctor	6.97	6.65	55.0	7.0	72.0	6808.0
Salesperson	6.40	6.00	45.0	7.0	72.0	6000.0

Analysis of the sleep parameters and lifestyle for each occupation

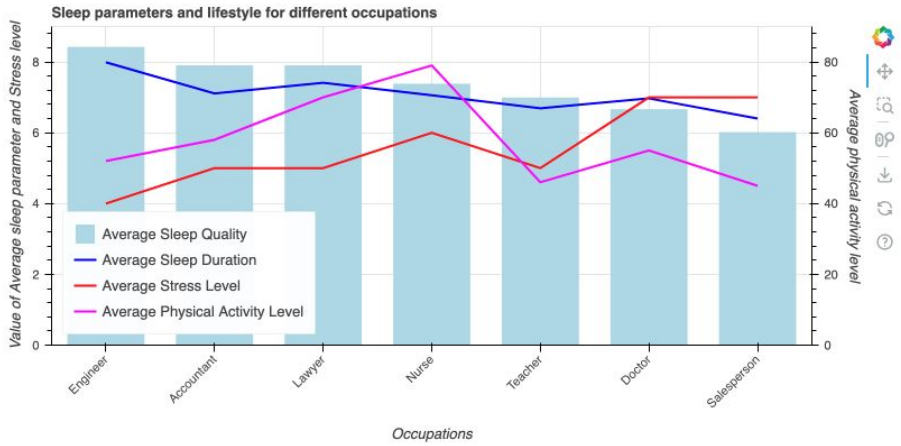
```
[ ] import math
    from bokeh.plotting import figure, show
    from bokeh.io import output_notebook

    output_notebook()
```

```
occupation = list(df_avg.index)
source = {'Occupations': occupation, 'Average sleep quality': df_avg['Average Sleep Quality']}

Figure1 = figure(title = "Average sleep quality for different occupations", x_range = occupation, y_range = (0, 8.5),
                 x_axis_label = "Occupations", y_axis_label = "Average sleep quality",
                 height = 300, width = 800)

Figure1.vbar(x = 'Occupations', top = 'Average sleep quality', source = source, width = 0.7)
Figure1.xaxis.major_label_orientation = math.pi/4

show(Figure1)
```

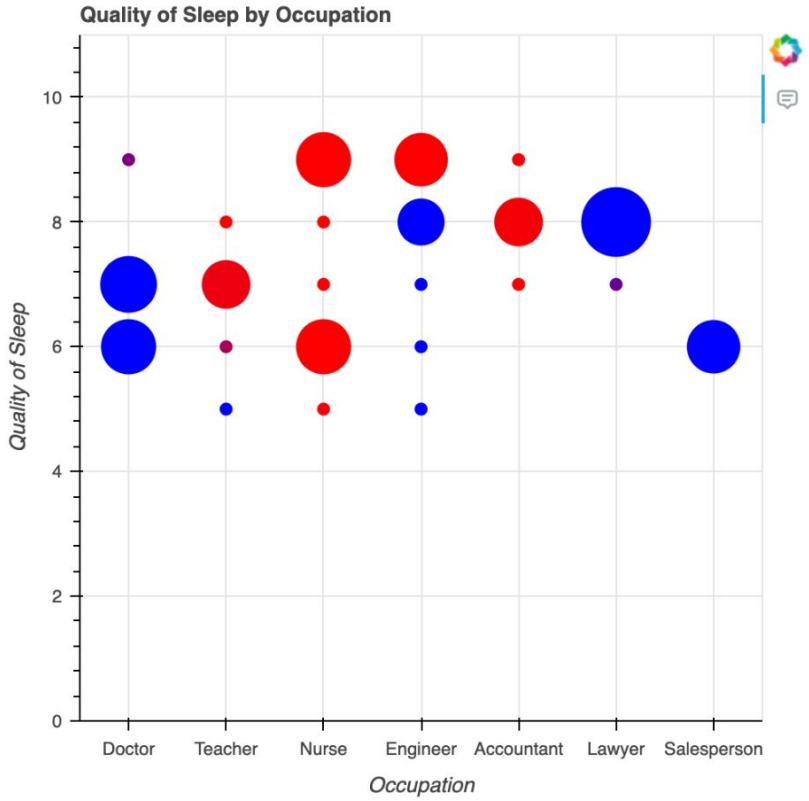


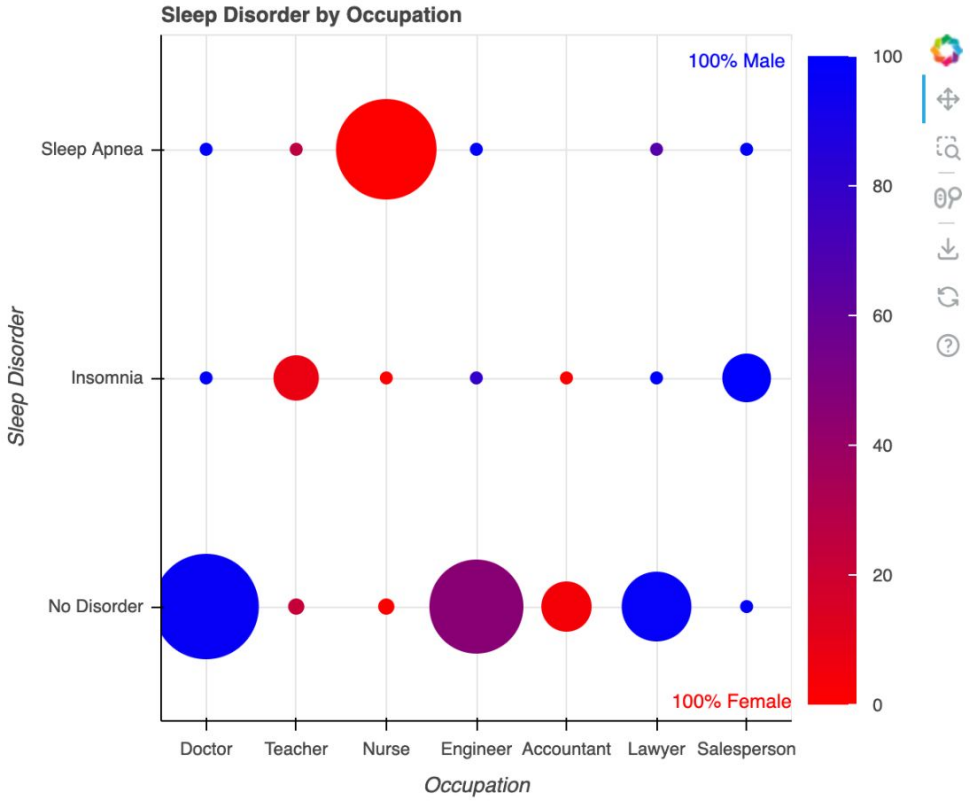
Analysis of the sleep parameters and lifestyle for each occupation

```
from bokeh.models import ColumnDataSource
from bokeh.models import LinearAxis, Range1d

source = ColumnDataSource(df_avg)


Figure2 = figure(title = "Sleep parameters and lifestyle for different occupations", x_range = occupation, y_range = (0, 9),
                 x_axis_label = "Occupations", y_axis_label = "Value of Average sleep parameter and Stress level",
                 height = 400, width = 800)

Figure2.vbar(x = 'Occupation', top = 'Average Sleep Quality', source = source, color="lightblue", width =0.7,  legend_label="Average Sleep Quality" )
Figure2.line(x = 'Occupation', y = 'Average Sleep Duration', source = source, color="blue", line_width =2, legend_label="Average Sleep Duration")
Figure2.line(x = 'Occupation', y = 'Average Stress Level', source = source, color="red", line_width =2, legend_label="Average Stress Level")
Figure2.line(x = 'Occupation', y = 'Average Physical Activity Level', source = source, color="magenta", line_width =2,
             legend_label="Average Physical Activity Level", y_range_name="y2")

Figure2.add_layout(LinearAxis(y_range_name='y2', axis_label='Average physical activity level'), 'right')
Figure2.extra_y_ranges = {"y2": Range1d(start=0, end=90)}

Figure2.xaxis.major_label_orientation = math.pi/4
Figure2.legend.location = 'bottom_left'
show(Figure2)
```

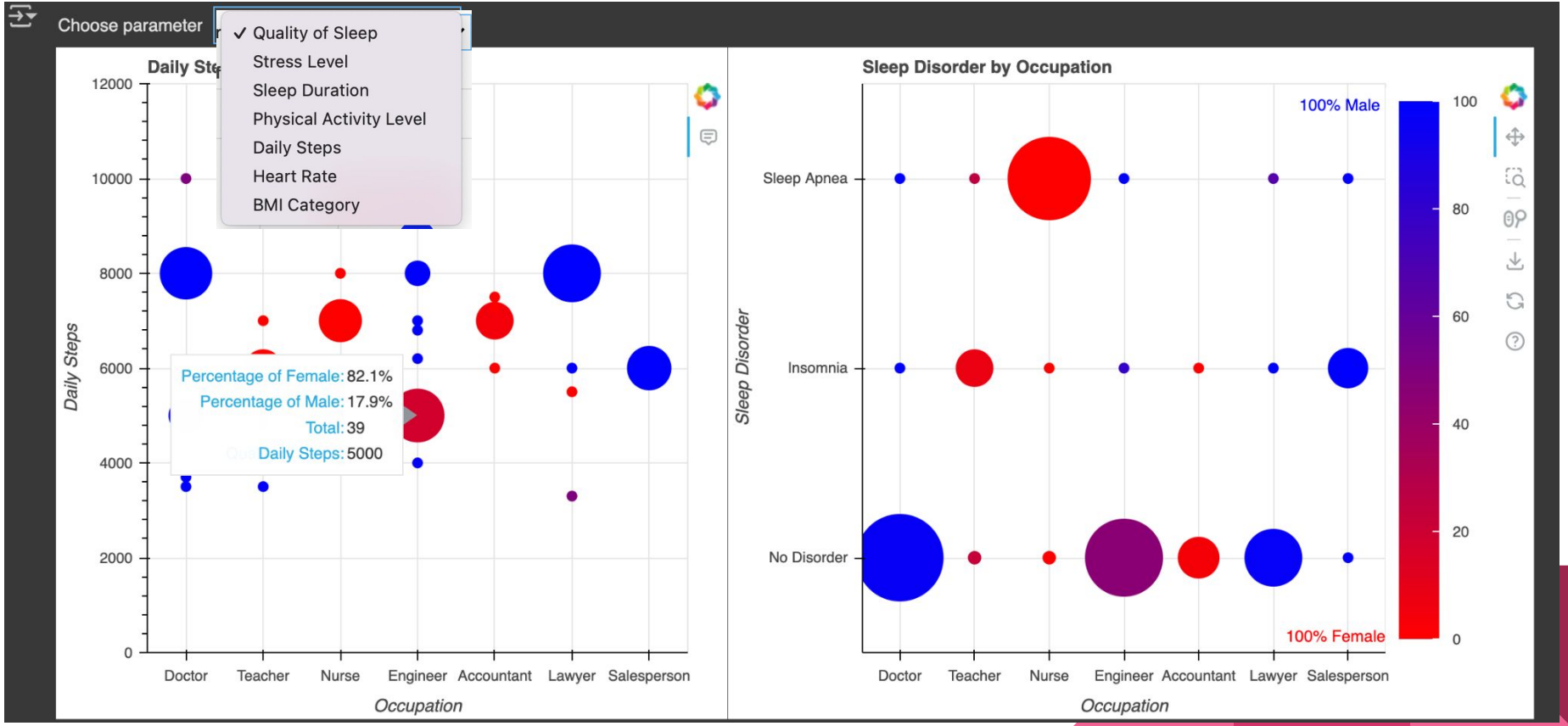


Visualization of Lifestyle and Sleep Disorders for Male and Female Participants





Visualization of Lifestyle and Sleep Disorders for Male and Female Participants



Visualization of Lifestyle and Sleep Disorders for Male and Female Participants

```
import pandas as pd
import ipywidgets as widgets
from ipywidgets import interact
from bokeh.plotting import figure, show, output_notebook
from bokeh.models import ColumnDataSource, ColorBar, LinearColorMapper, HoverTool, Label, BasicTicker, Range1d, FactorRange
from bokeh.io import push_notebook
from bokeh.layouts import row
from bokeh.colors import RGB

# Transform BMI Category to numeric values:
# 0 for 'Normal' and 'Normal Weight', 1 for 'Overweight', 2 for 'Obese'
data["BMI Category"] = data["BMI Category"].replace({
    "Normal": 0,
    "Normal Weight": 0,
    "Overweight": 1,
    "Obese": 2
})

# Aggregation
def process_data(metric):
    grouped = data.groupby(["Occupation", metric, "Gender"]).size().unstack(fill_value=0)
    grouped["Total"] = grouped.sum(axis=1)
    grouped["Male_Ratio"] = grouped["Male"] / grouped["Total"]
    grouped["Female_Ratio"] = grouped["Female"] / grouped["Total"]
    grouped["Size"] = grouped["Total"].clip(lower=7)
    grouped = grouped.reset_index()
    grouped.rename(columns={metric: "Metric"}, inplace=True)

    def compute_rgb(male_ratio, female_ratio):
        r = int(255 * female_ratio)
        g = 0
        b = int(255 * male_ratio)
        return RGB(r, g, b)

    grouped["Color"] = grouped.apply(lambda row: compute_rgb(row["Male_Ratio"], row["Female_Ratio"]).to_hex(), axis=1)
    return grouped
```

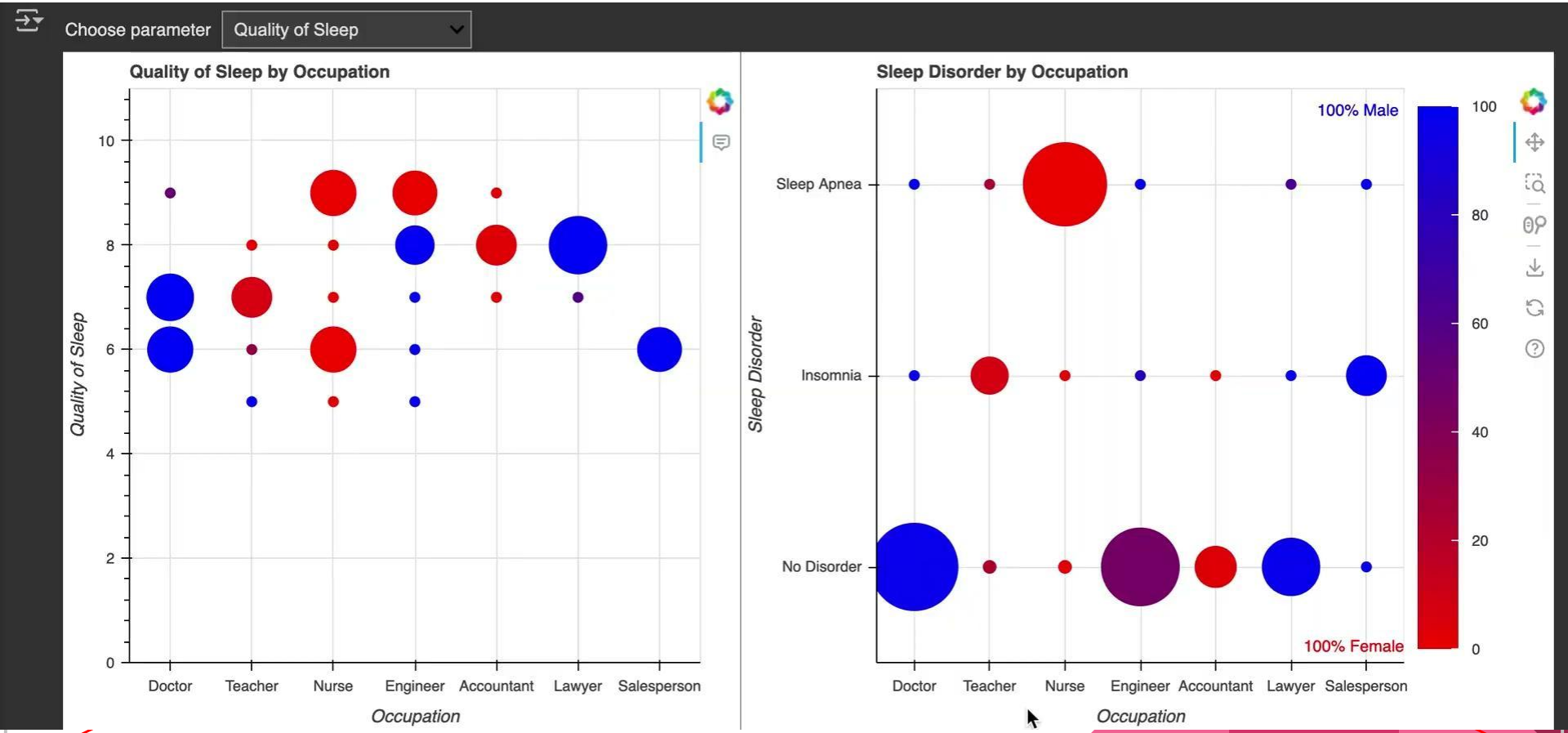
Visualization of Lifestyle and Sleep Disorders for Male and Female Participants

```
# Create figures
source = ColumnDataSource(process_data("Quality of Sleep"))
p1 = figure(x_range=data["Occupation"].unique(), x_axis_label="Occupation", y_axis_label="Quality of Sleep", width=500, height=500)

p2 = figure(x_range=data["Occupation"].unique(), y_range=['No Disorder', 'Insomnia', 'Sleep Apnea'], title = "Sleep Disorder by Occupation", x_axis_la

# Define color mapper for color bar
palette = [RGB(int(255 * (1 - i / 100)), 0, int(255 * (i / 100))).to_hex() for i in range(101)]
mapper = LinearColorMapper(palette=palette, low=0, high=100)
color_bar = ColorBar(color_mapper=mapper, location=(0, 0), ticker=BasicTicker(), width=30, height=400, label_standoff=10)
p2.add_layout(color_bar, 'right')

# Add labels
p2.add_layout(Label(x=325, y=400 , x_units='screen', y_units='screen', text='100% Male', text_color='blue', text_font_size='9pt'))
p2.add_layout(Label(x=315, y=5, x_units='screen', y_units='screen', text='100% Female', text_color='red', text_font_size='9pt'))

# Add scatter plot
p1.scatter(x="Occupation", y="Metric", size="Size", color="Color", source=source, alpha=1)

# Create data source for Sleep Disorder plot
source_p2 = process_sleep_disorder()
# Add scatter plot to p2
p2.scatter(x="Occupation", y="Sleep Disorder Category", size="Size", color="Color", source=source_p2, alpha=1)

# Define y-axis ranges
y_ranges = {
    "Quality of Sleep": Range1d(0, 11),
    "Stress Level": Range1d(0, 11),
    "Sleep Duration": Range1d(4, 10),
    "Physical Activity Level": Range1d(0, 100),
    "Daily Steps": Range1d(0, 12000),
    "Heart Rate": Range1d(40, 100),
    "BMI Category": Range1d(-1, 3)  # Range for BMI Category (0 for 'Normal' and 'Normal Weight', 1 for 'Overweight', 2 for 'Obese')
}
```

Visualization of Lifestyle and Sleep Disorders for Male and Female Participants

```
def update_plot(metric):

    # Process data based on the selected metric
    new_data = process_data(metric)  # Process for the selected metric
    source.data = new_data.to_dict(orient='list')  # Update data source

    # Set the y_range based on predefined ranges for continuous metrics
    p1.y_range = y_ranges.get(metric, Range1d(0, 11))  # Continuous range based on metric
    p1.yaxis.axis_label = metric  # Update y-axis label for continuous axis

    # Clear previous scatter plot and add new one
    p1.renderers = []  # Clear existing renderers
    p1.scatter(x="Occupation", y="Metric", size="Size", color="Color", source=source, alpha=1)  # Add new scatter plot

    # Update the plot title dynamically based on the metric
    p1.title.text = f"{metric} by Occupation"

    # Tooltips for the hover tool
    tooltips = [
        ('Percentage of Female', '@Female_Ratio{0.0%}'),
        ('Percentage of Male', '@Male_Ratio{0.0%}'),
        ('Total', '@Total'),
        (f'{metric}', f'@{{Metric}}')
    ]

    hover = HoverTool(tooltips=tooltips)
    p1.tools = [hover]

    push_notebook()  # Ensure the plot updates in the notebook

# Interactive widget
interact(update_plot, metric=widgets.Dropdown(options=["Quality of Sleep", "Stress Level", "Sleep Duration", "Physical Activity Level", "Daily Steps",

# Show initial plots
handle = show(row(p1, p2), notebook_handle=True)
```

Visualization of Lifestyle and Sleep Disorders for Male and Female Participants

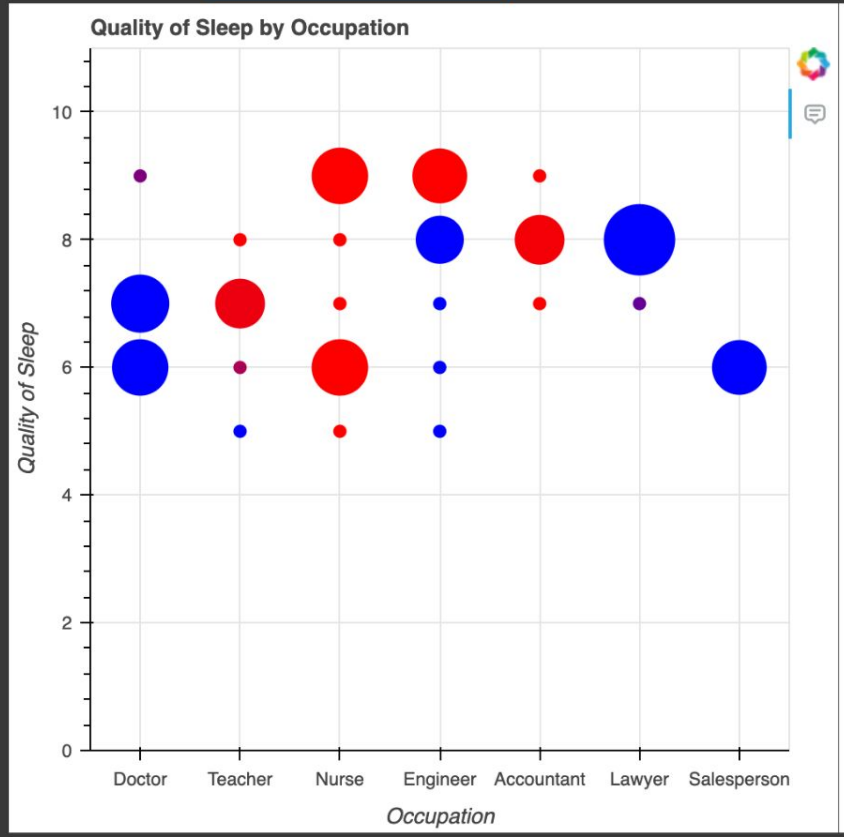


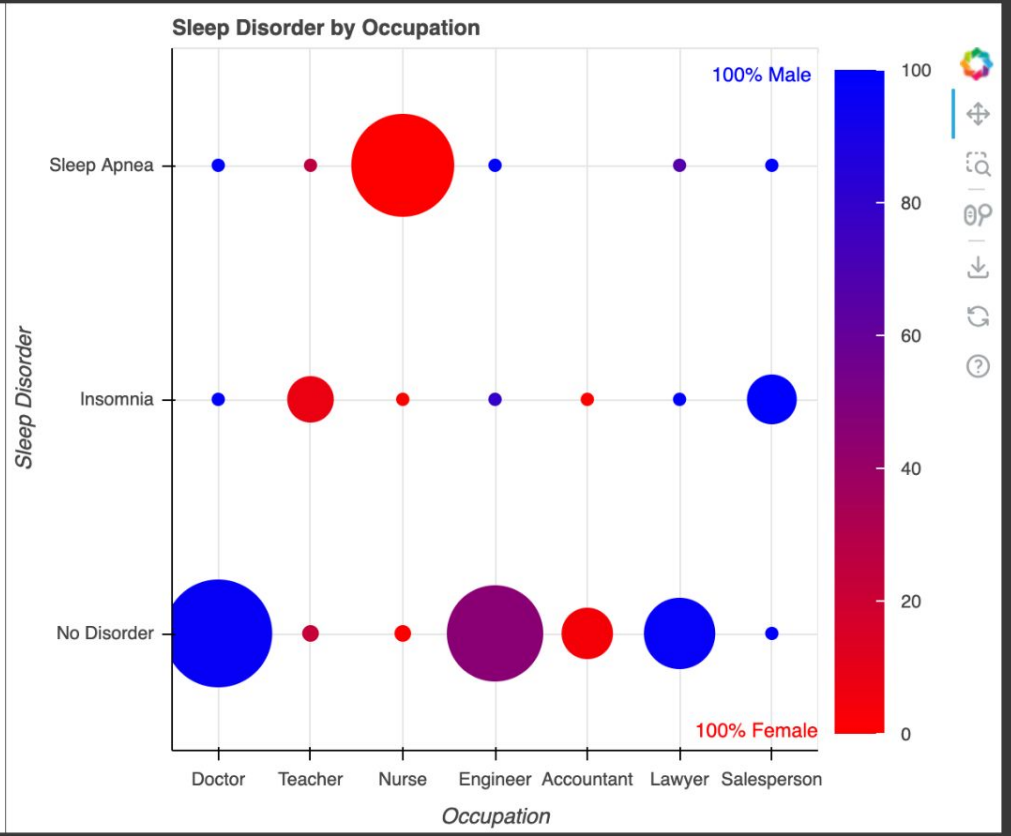
Conclusion

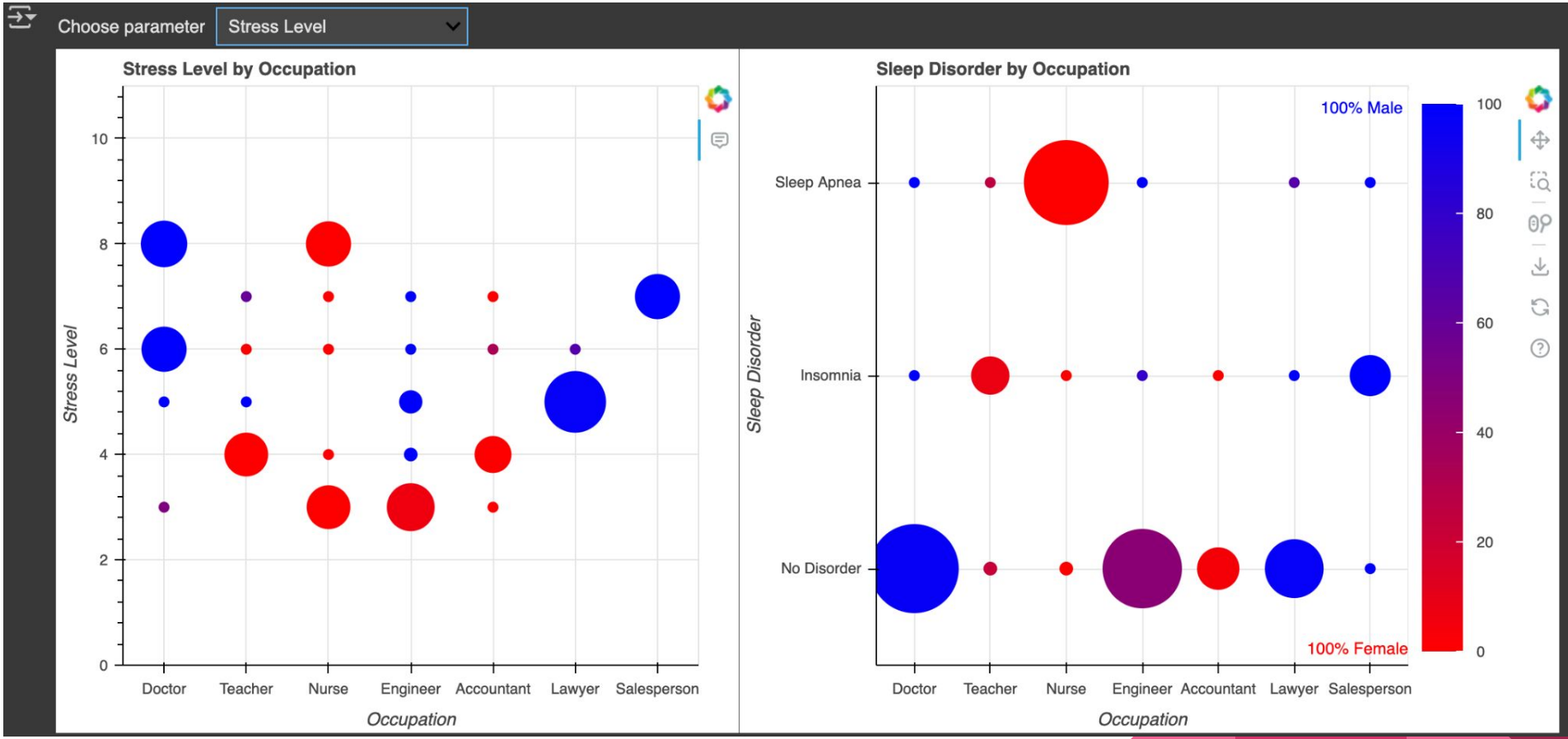
- Female participants have better Quality of Sleep and lower Stress Level in most Occupation groups.
- Physical activity level, Daily steps and Heart Rate are higher for Male participants.
- BMI Category is the most important parameter for Sleep Disorder.

Thank you!

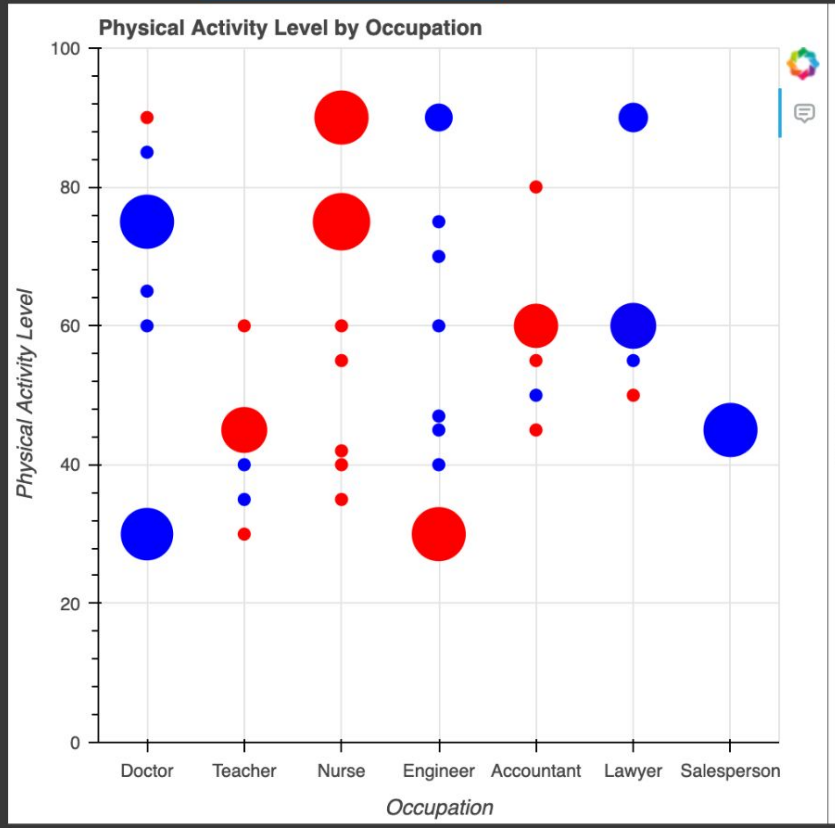
Choose parameter: Quality of Sleep

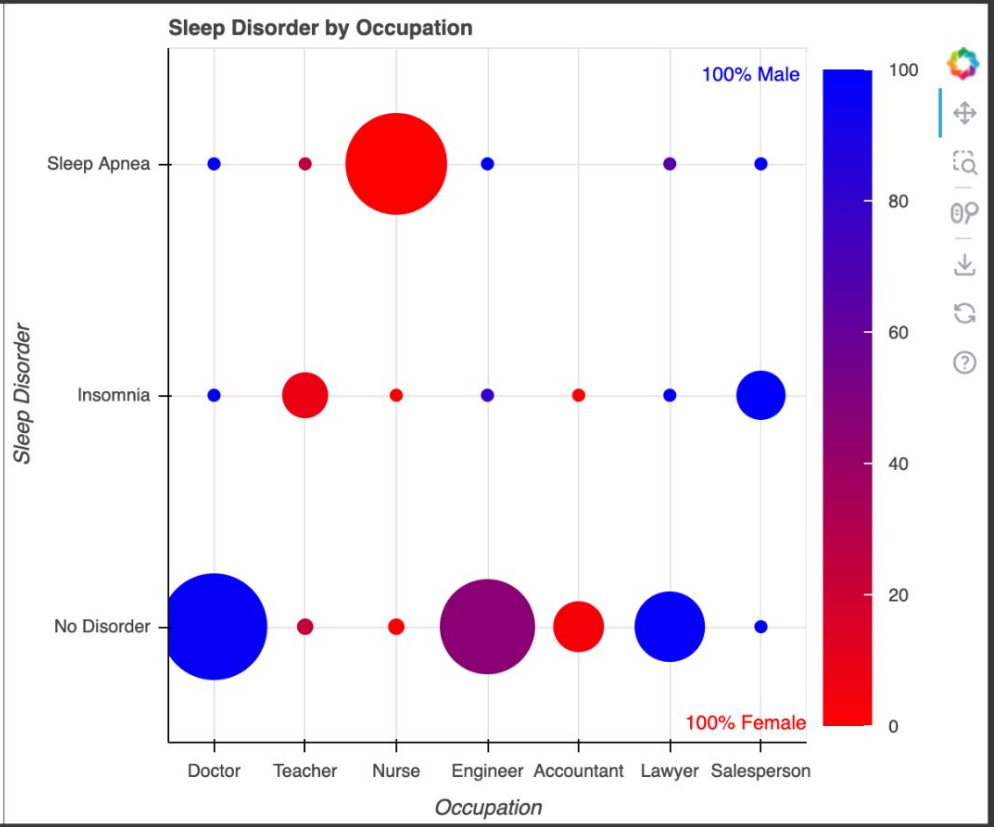


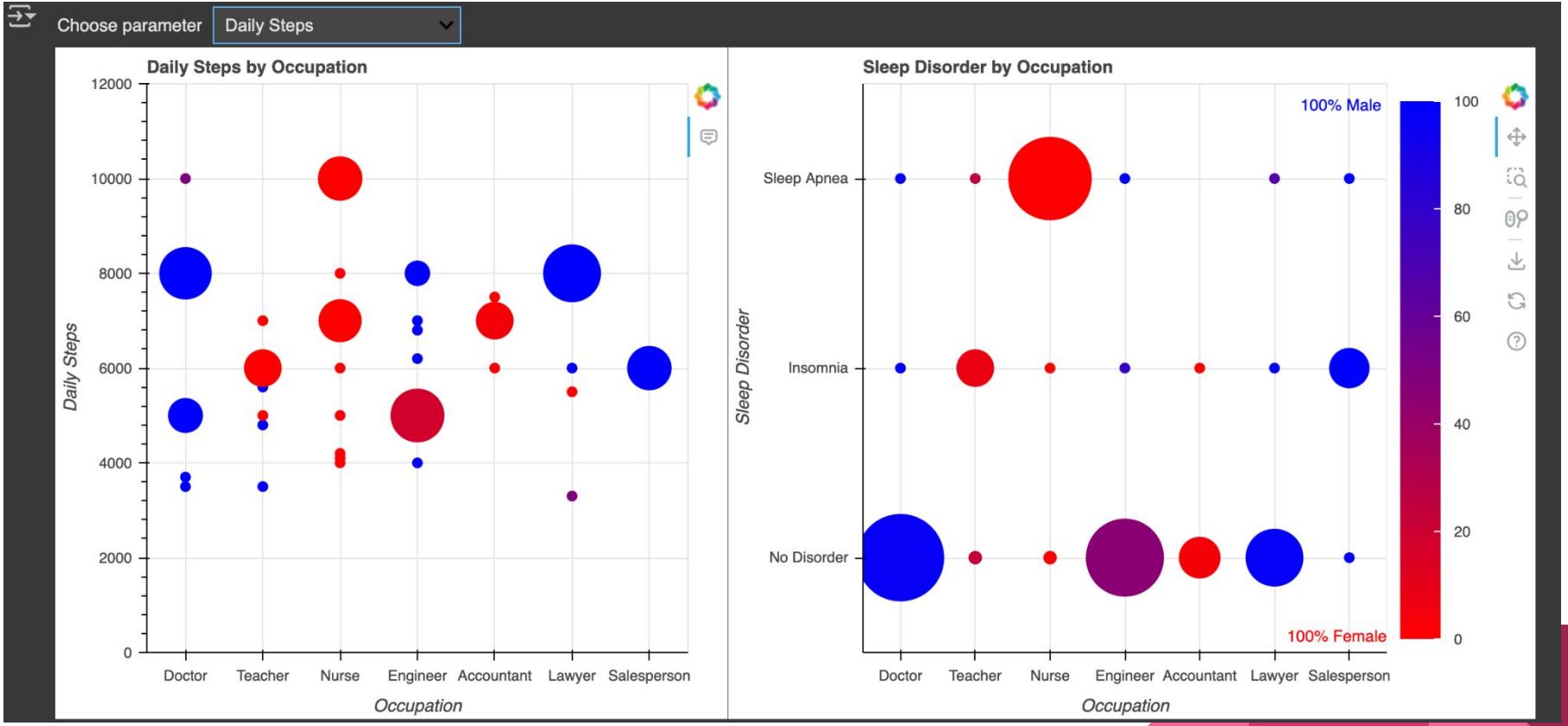


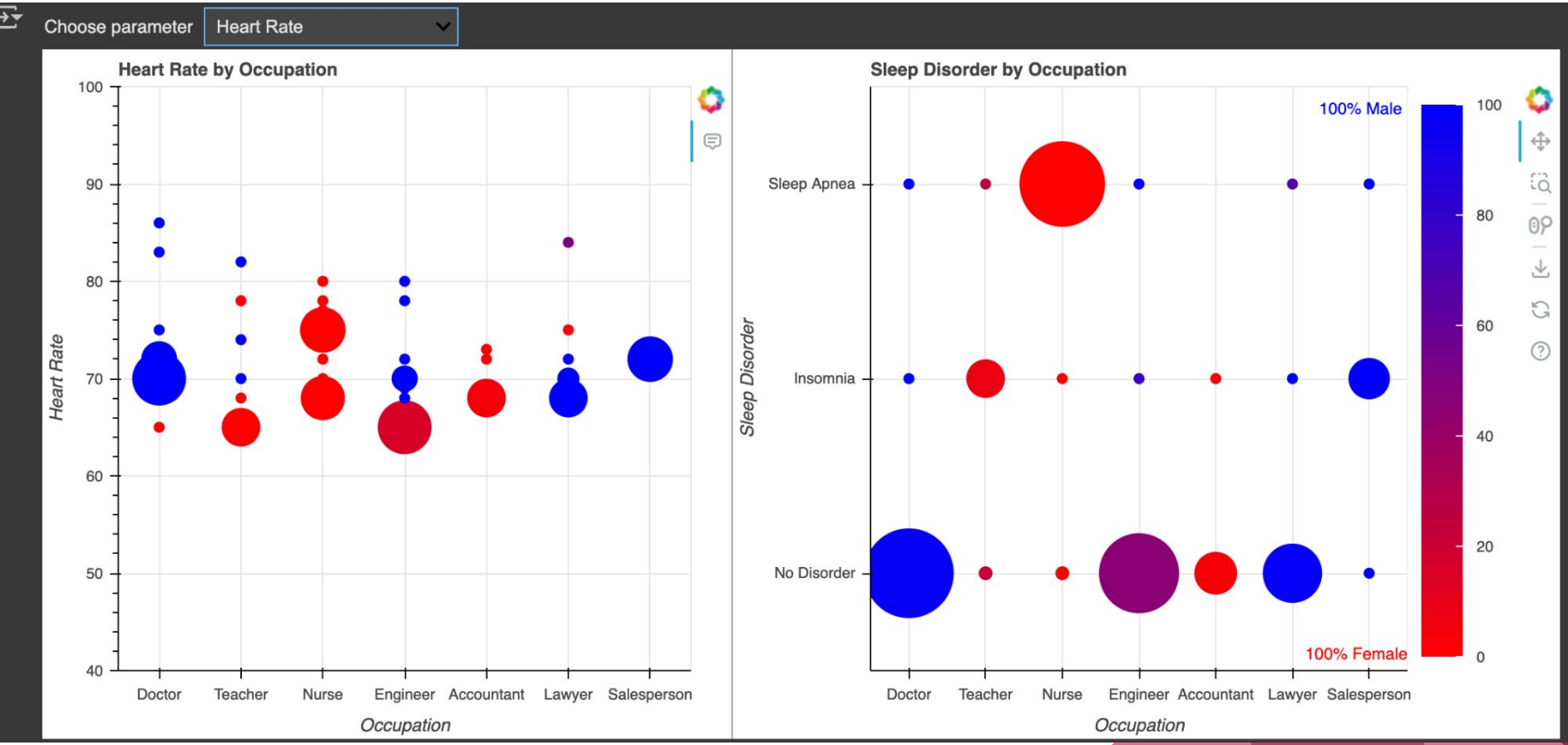

Choose parameter
Stress Level
Stress Level by Occupation
Stress Level
Occupation
Doctor
Teacher
Nurse
Engineer
Accountant
Lawyer
Salesperson
0
2
4
6
8
10
Sleep Disorder by Occupation
Sleep Disorder
Occupation
Sleep Apnea
Insomnia
No Disorder
100% Male
100% Female
0
20
40
60
80
100

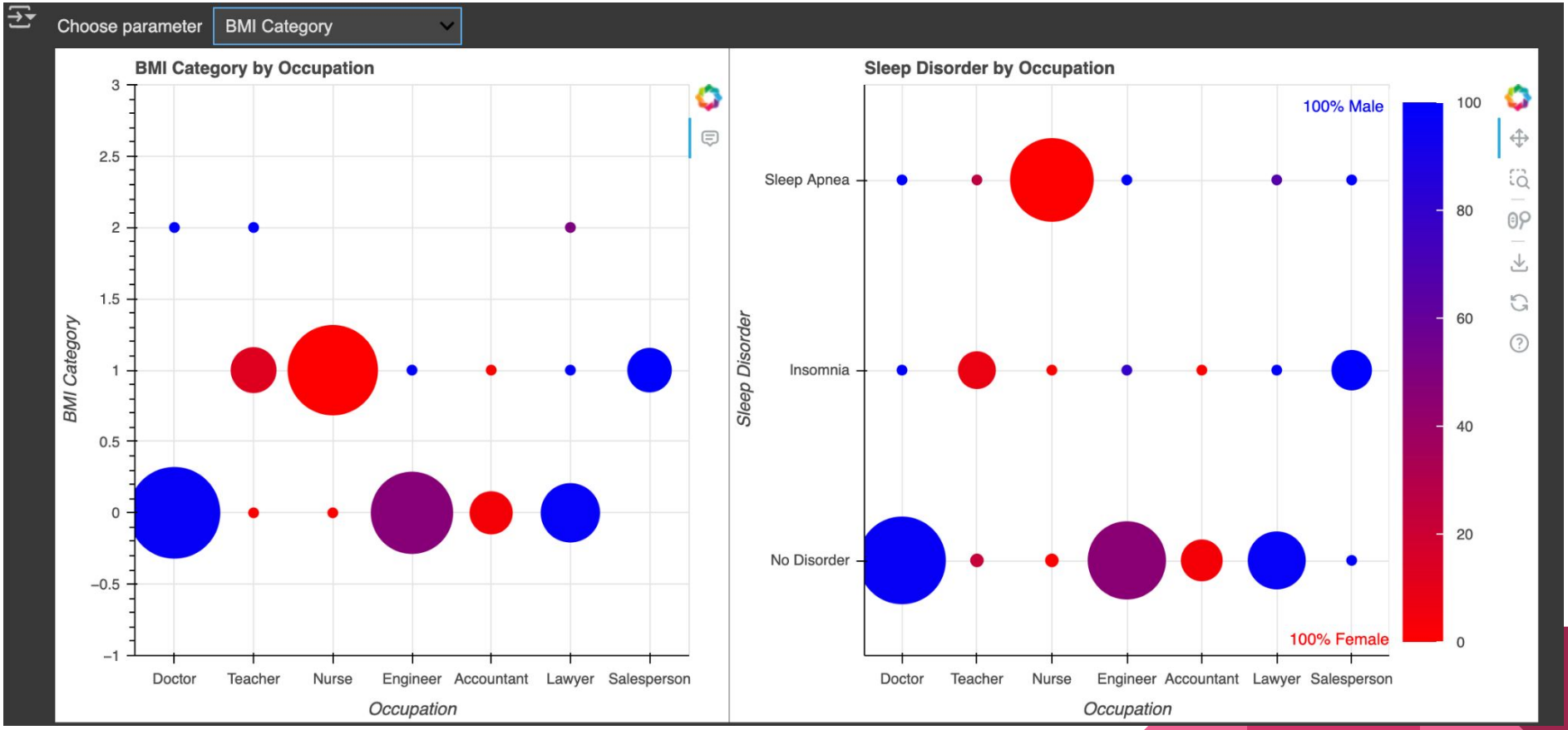
Choose parameter | Physical Activity Level






Choose parameter
Daily Steps
Daily Steps by Occupation
Daily Steps
Occupation
Doctor
Teacher
Nurse
Engineer
Accountant
Lawyer
Salesperson
Sleep Disorder by Occupation
Sleep Disorder
Sleep Apnea
Insomnia
No Disorder
100% Male
100% Female


Choose parameter
Heart Rate
Heart Rate by Occupation
Heart Rate
100
90
80
70
60
50
40
Doctor
Teacher
Nurse
Engineer
Accountant
Lawyer
Salesperson
Occupation
Sleep Disorder by Occupation
100% Male
100% Female
Sleep Disorder
Sleep Apnea
Insomnia
No Disorder
100
80
60
40
20
0
Doctor
Teacher
Nurse
Engineer
Accountant
Lawyer
Salesperson
Occupation


Choose parameter
BMI Category
BMI Category by Occupation
BMI Category
3
2.5
2
1.5
1
0.5
0
−0.5
−1
Doctor
Teacher
Nurse
Engineer
Accountant
Lawyer
Salesperson
Occupation
Sleep Disorder by Occupation
100% Male
100% Female
Sleep Disorder
Sleep Apnea
Insomnia
No Disorder
100
80
60
40
20
0
Doctor
Teacher
Nurse
Engineer
Accountant
Lawyer
Salesperson
Occupation