Introduction to relational plots and subplots

INTRODUCTION TO DATA VISUALIZATION WITH SEABORN

Erin Case
Data Scientist

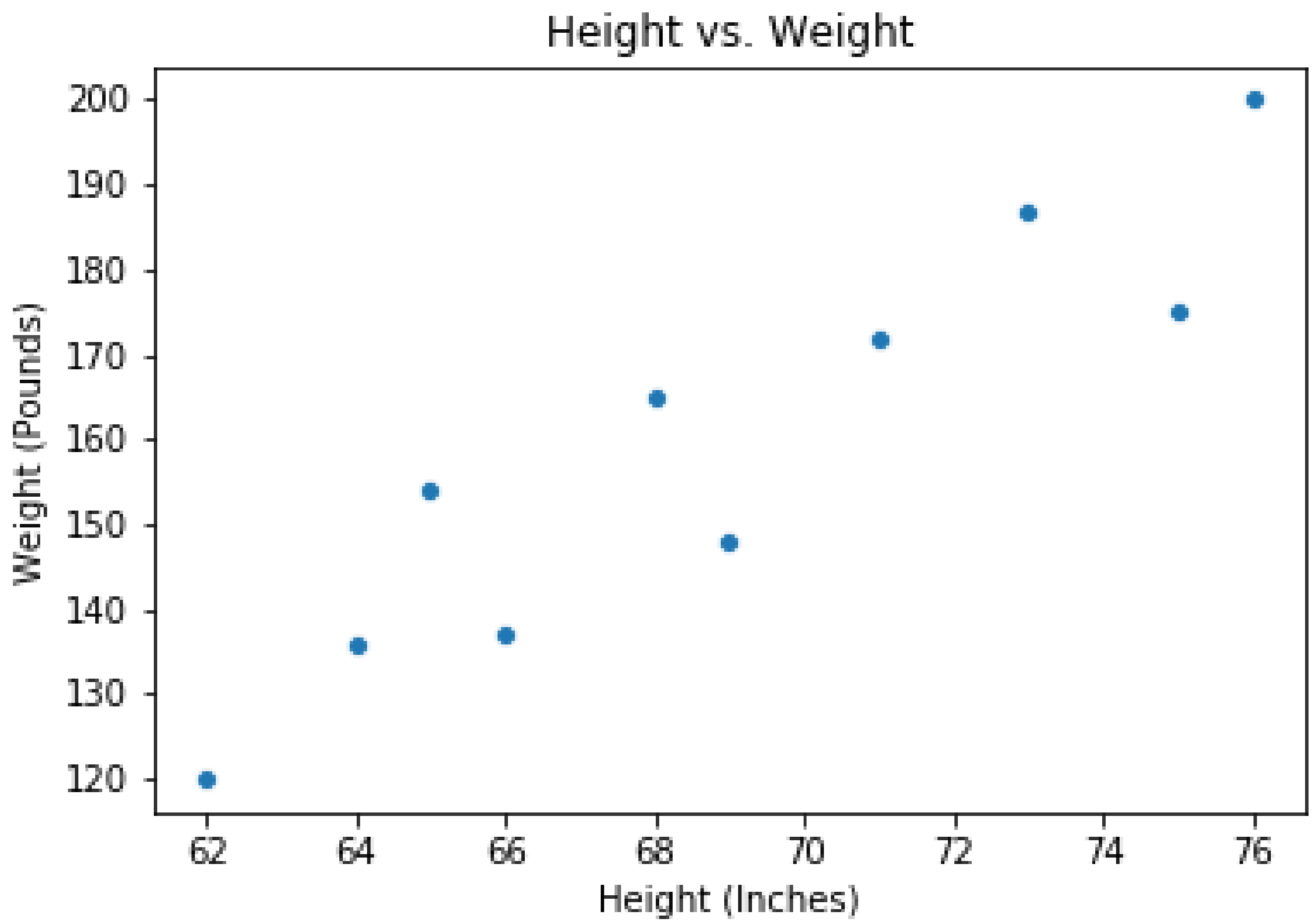
Questions about quantitative variables

Relational plots

x

- Height vs. weight

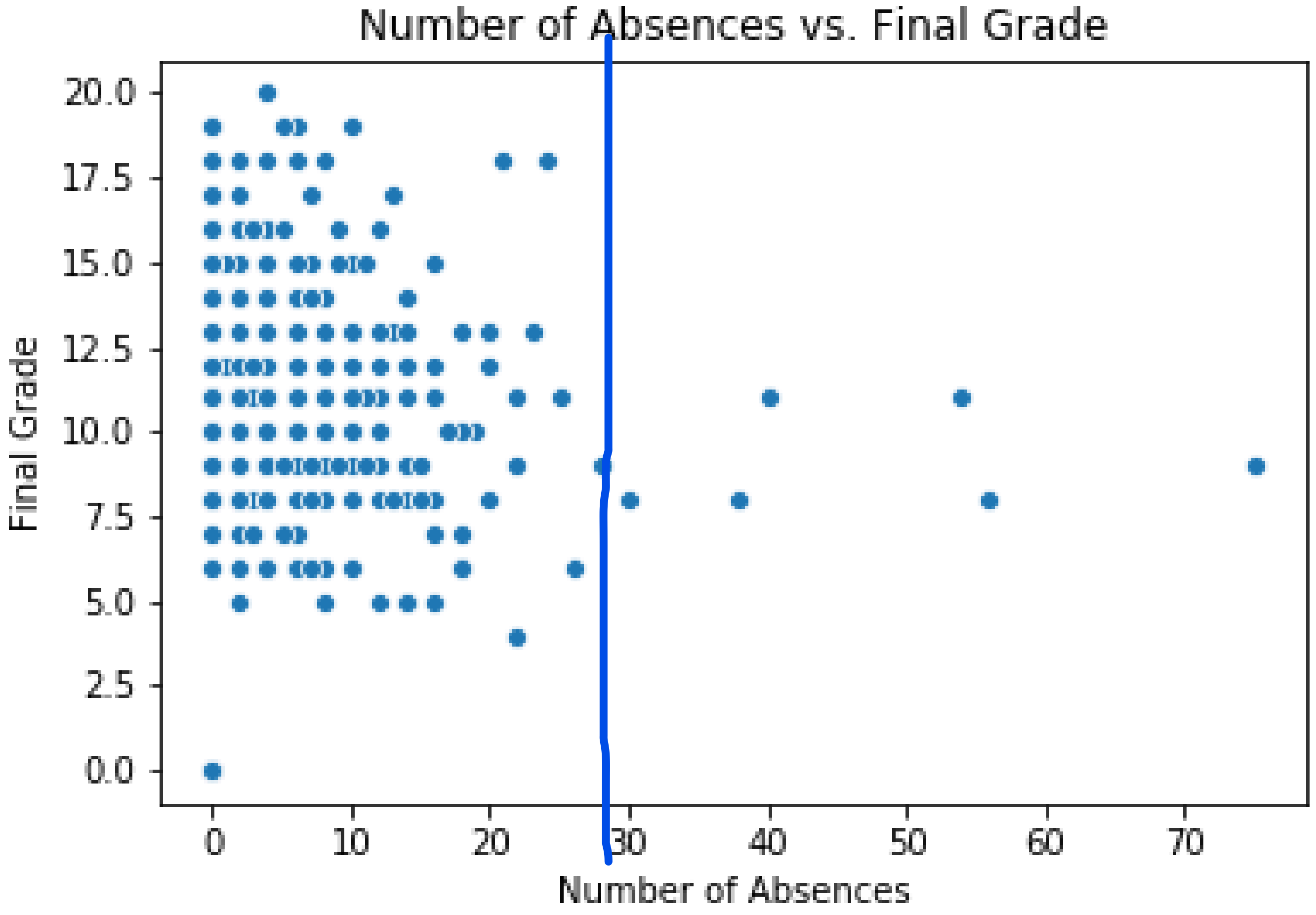
x y



Questions about quantitative variables

Relational plots

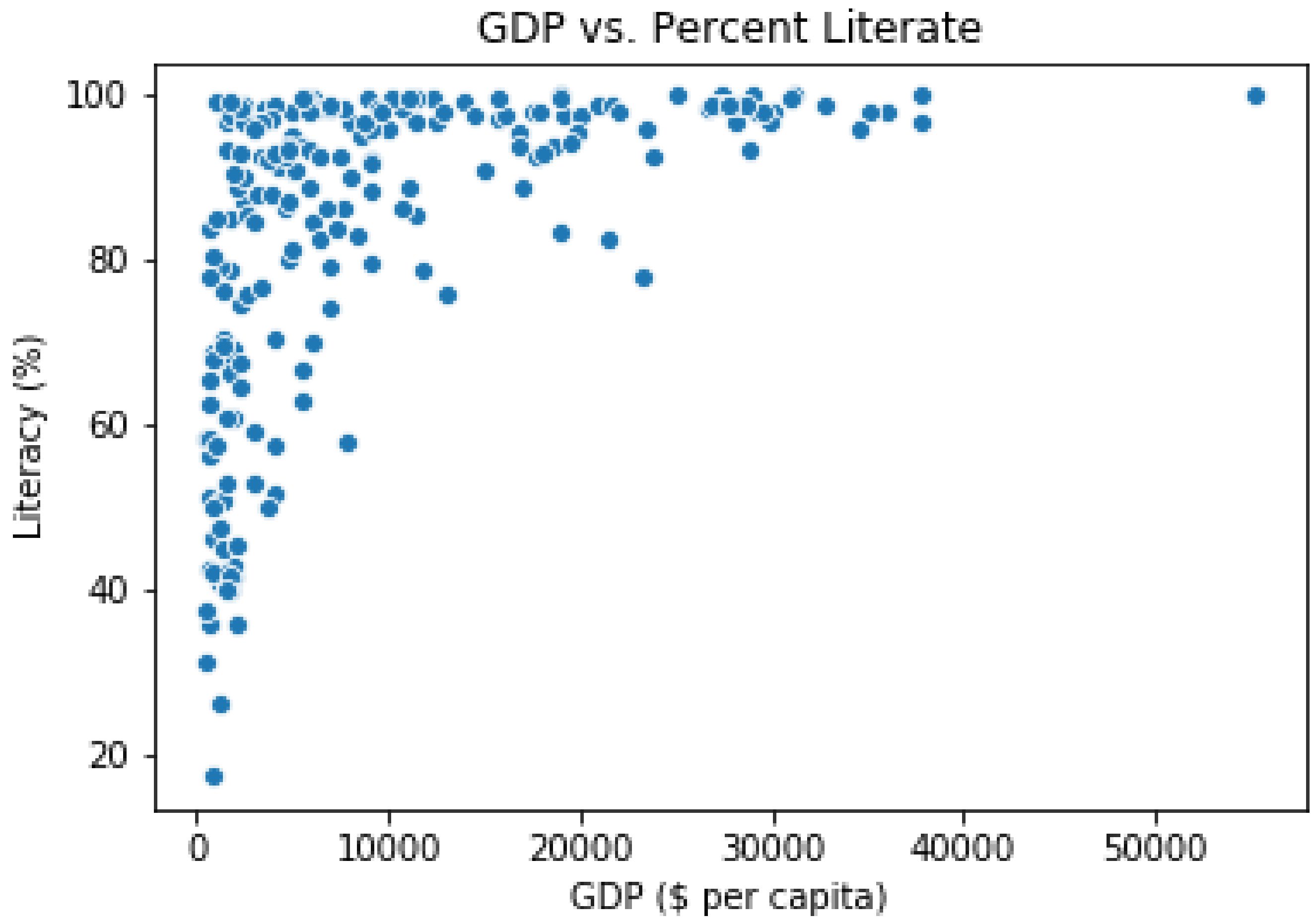
- Height vs. weight
- Number of school absences vs. final grade

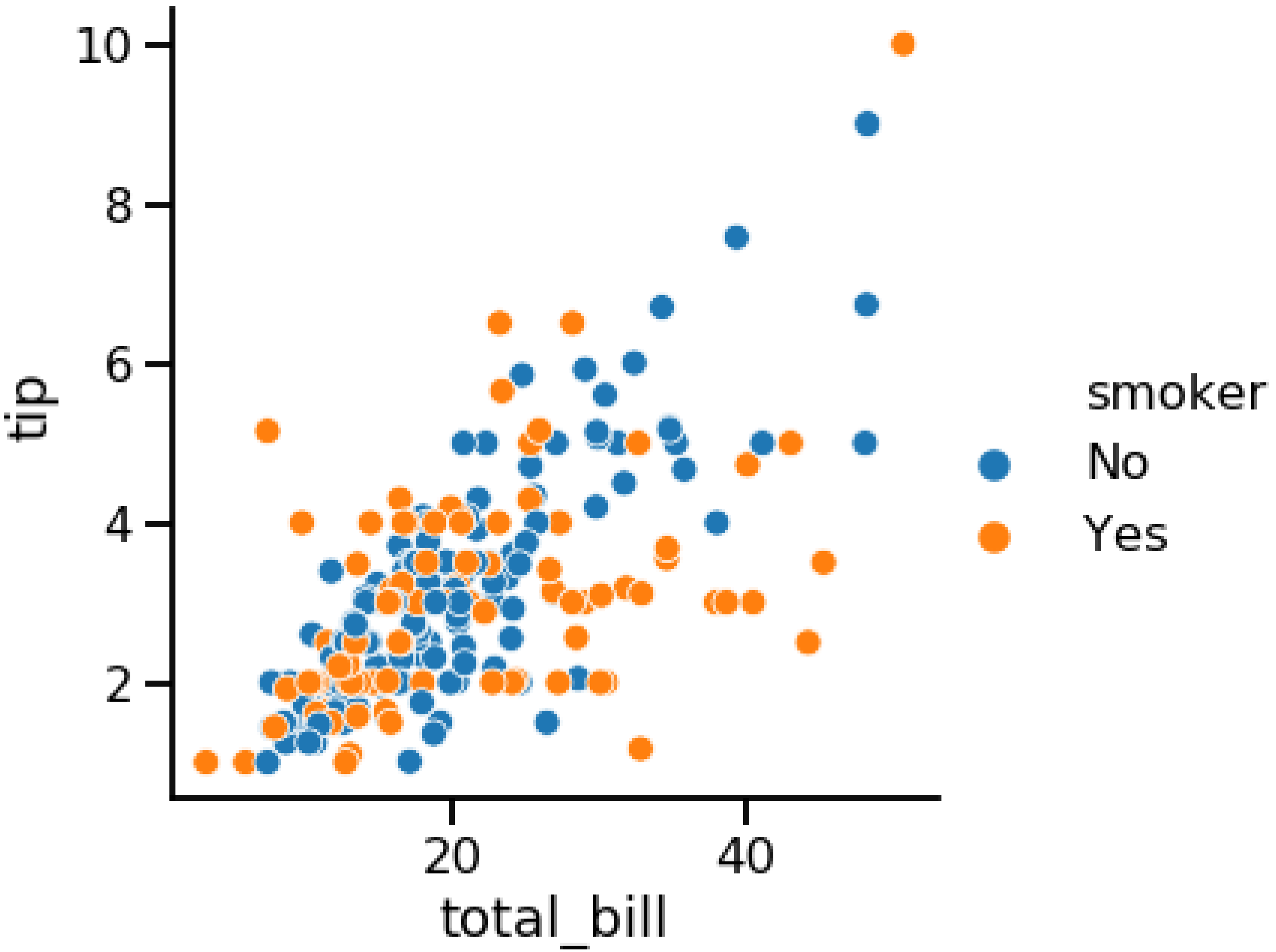


Questions about quantitative variables

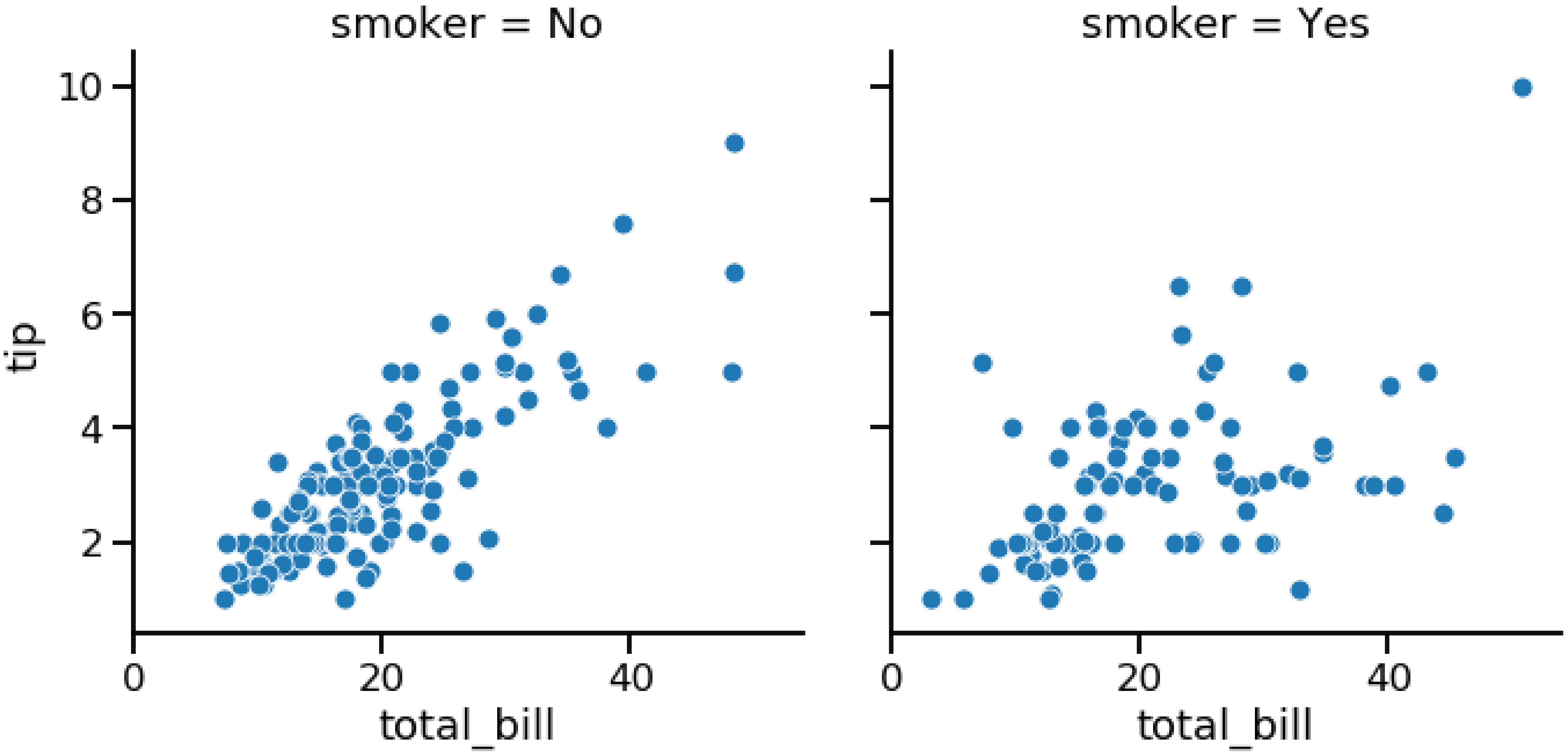
Relational plots

- Height vs. weight
- Number of school absences vs. final grade
- GDP vs. percent literate





[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/



[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Introducing relplot()

- Create "relational plots": scatter plots or line plots

Why use `relplot()` instead of `scatterplot()` ?

- `relplot()` lets you create subplots in a single figure

scatterplot() vs. relplot()

Using `scatterplot()`

```python
import seaborn as sns
import matplotlib.pyplot as plt

sns.scatterplot(x="total_bill",
                y="tip",
                data=tips)

plt.show()
```

Using `relplot()`

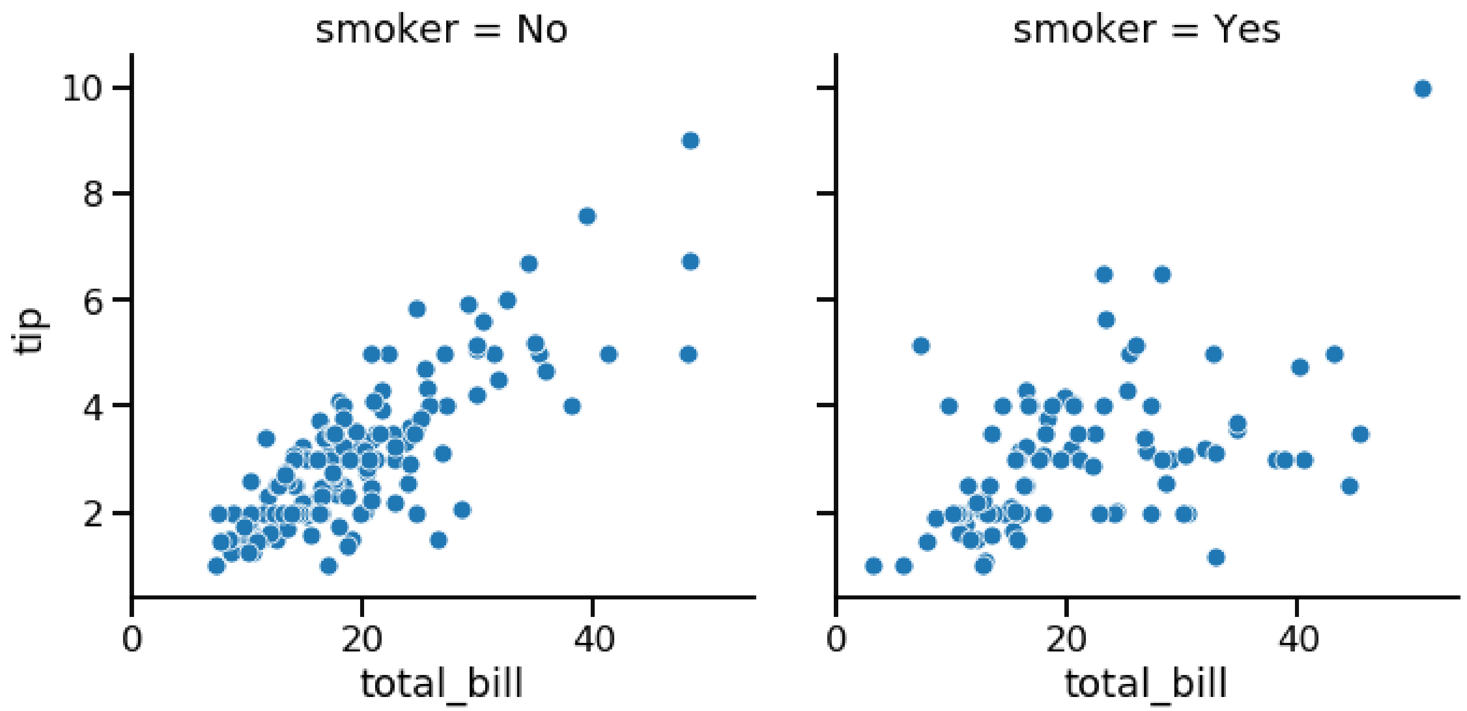
```python
import seaborn as sns
import matplotlib.pyplot as plt

sns.relplot(x="total_bill",
            y="tip",
            data=tips,
            kind="scatter")

plt.show()
```

[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Subplots in columns

wihtout wrap :[Thur] [Fri] [Sat] [Sun]

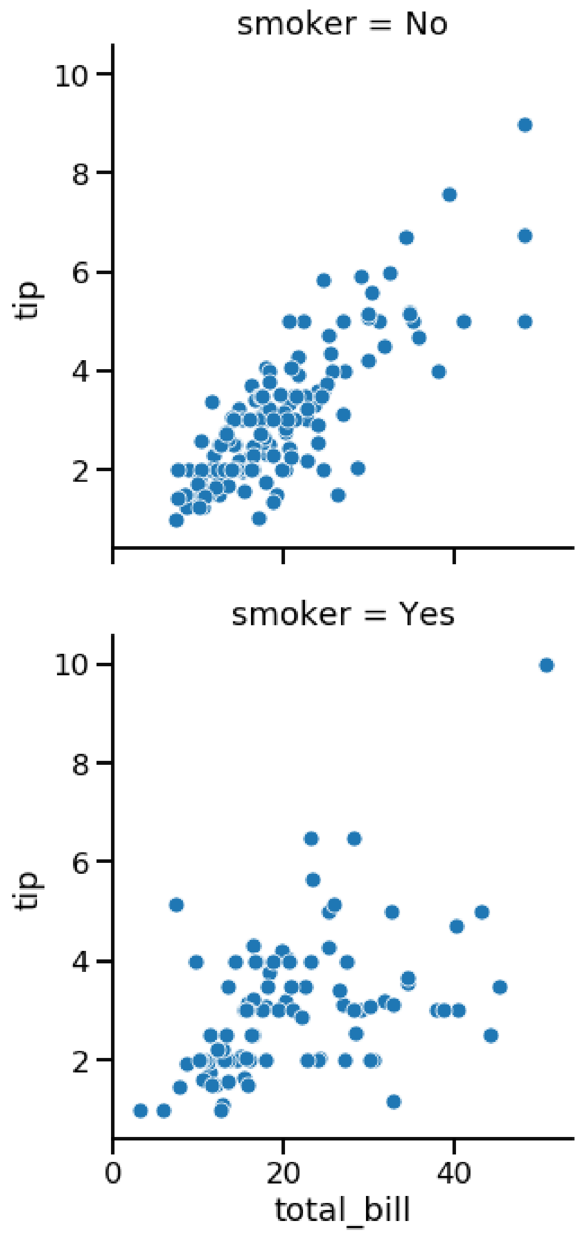
```
import seaborn as sns
import matplotlib.pyplot as plt

sns.relplot(x="total_bill",
            y="tip",
            data=tips,
            kind="scatter",
            col="smoker")
plt.show()
```



[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Subplots in rows

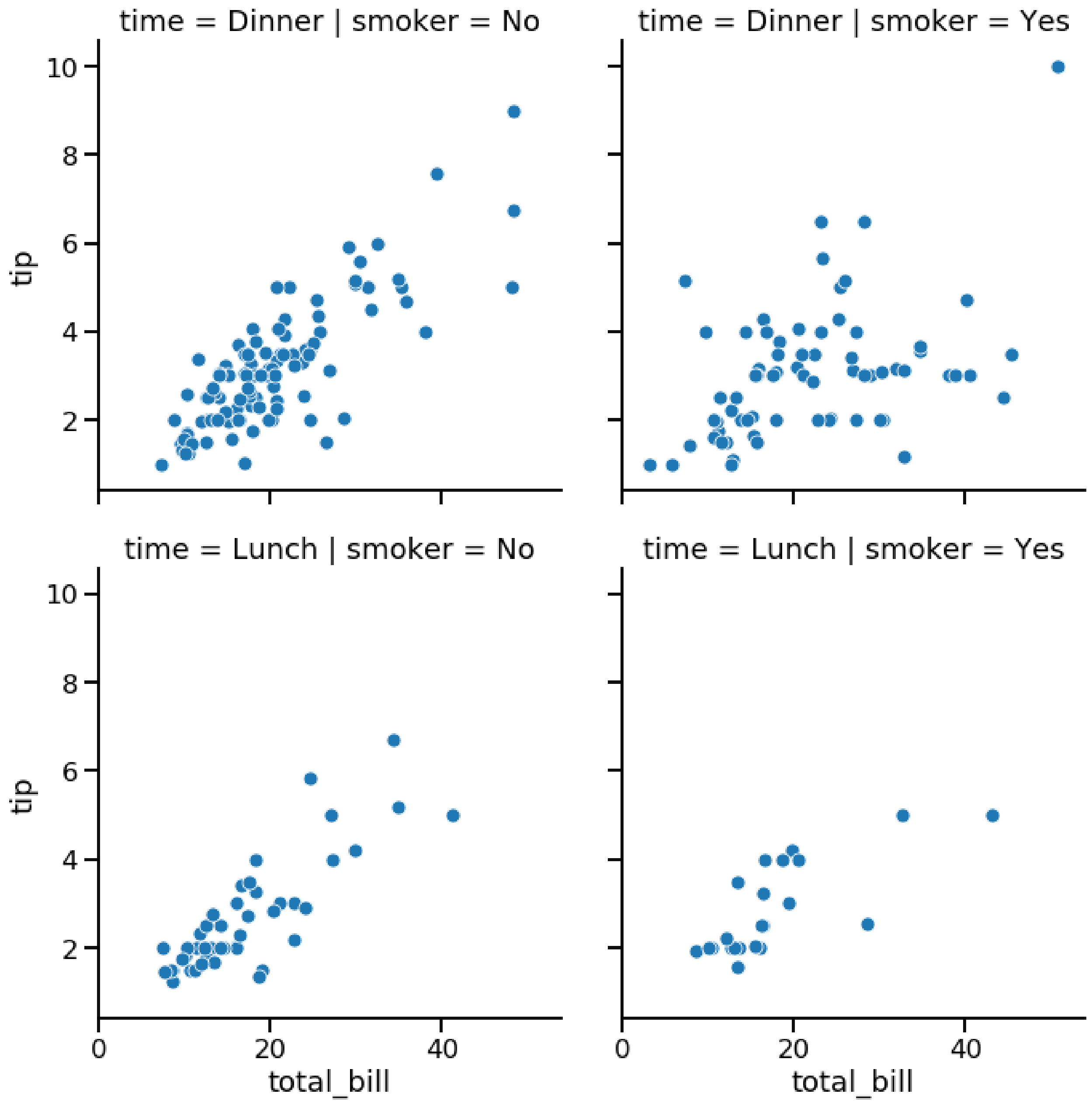
```
import seaborn as sns
import matplotlib.pyplot as plt

sns.relplot(x="total_bill",
            y="tip",
            data=tips,
            kind="scatter",
            row="smoker")
plt.show()
```



[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Subplots in rows and columns

```
import seaborn as sns
import matplotlib.pyplot as plt

sns.relplot(x="total_bill",
            y="tip",
            data=tips,
            kind="scatter",
            col="smoker",
            row="time")
plt.show()
```

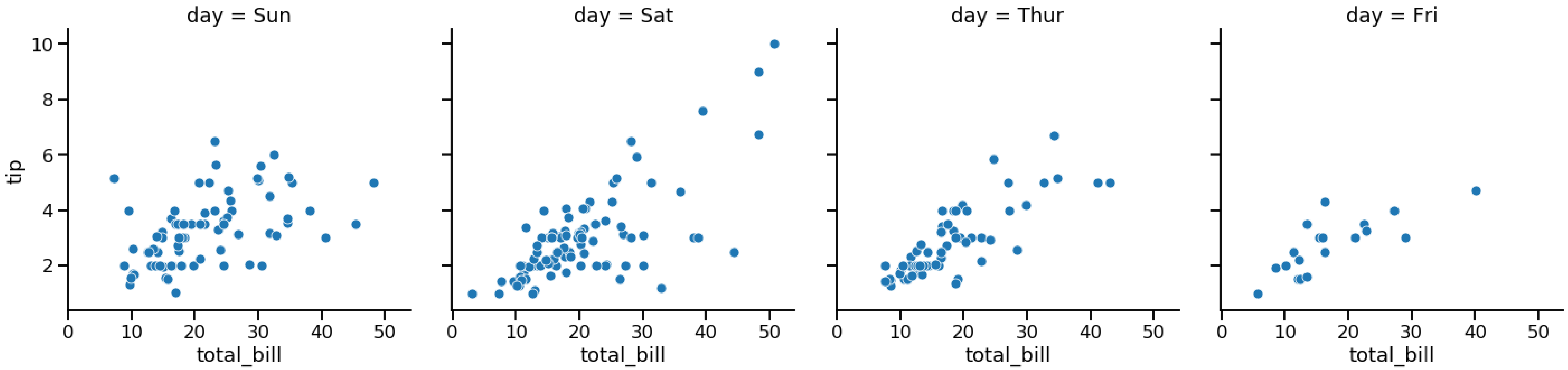


swapping the row and col wont make much difference

	smoker=No	smoker=Yes
time=Lunch	[Plot A]	[Plot B]
time=Dinner	[Plot C]	[Plot D]

[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Subgroups for days of the week



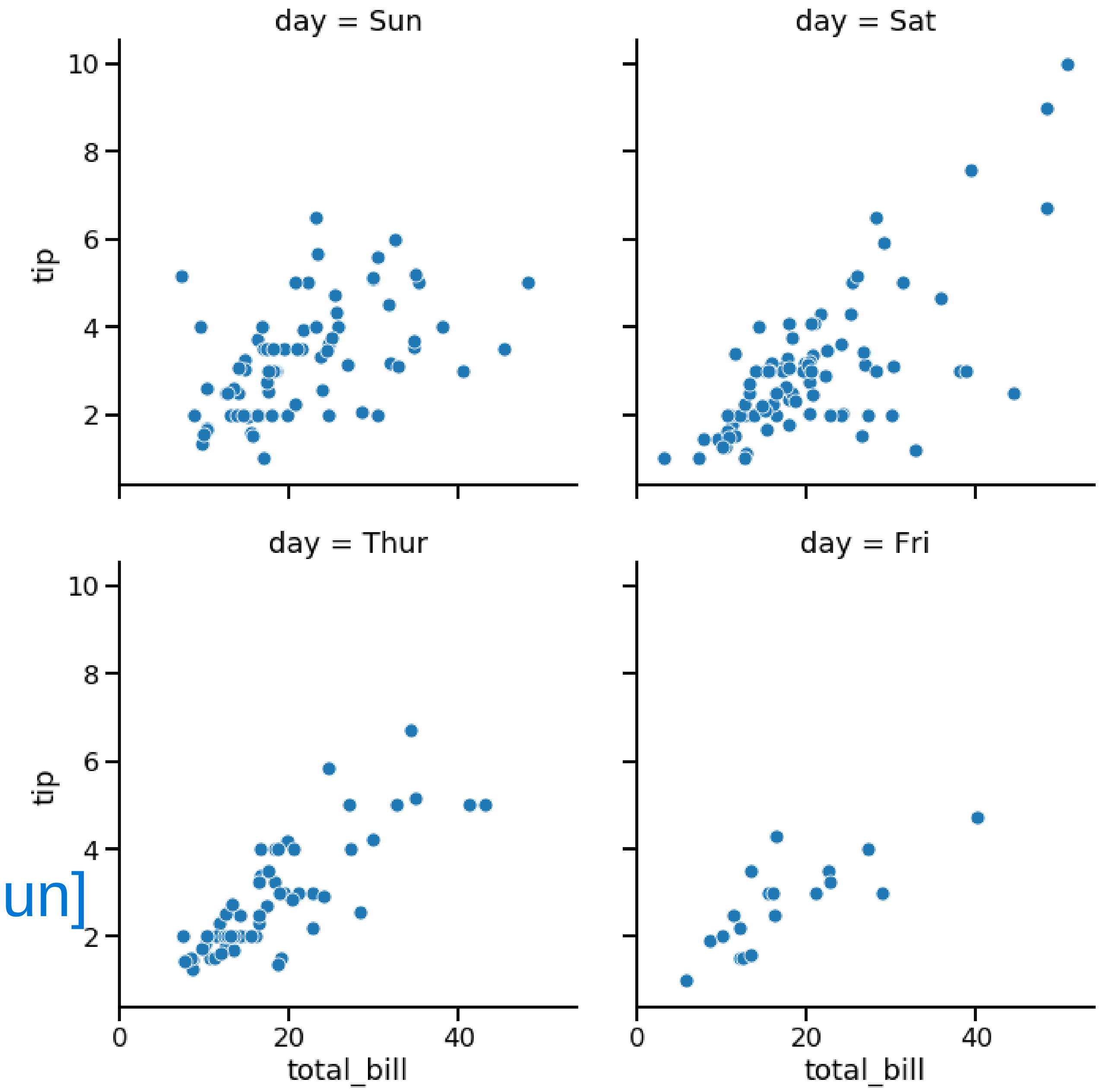
[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Wrapping columns

```
import seaborn as sns
import matplotlib.pyplot as plt

sns.relplot(x="total_bill",
            y="tip",
            data=tips,
            kind="scatter",
            col="day",
            col_wrap=2)
plt.show()
```

wihtout wrap :[Thur] [Fri] [Sat] [Sun]

with wrap :[Thur] [Fri]
[Sat] [Sun]

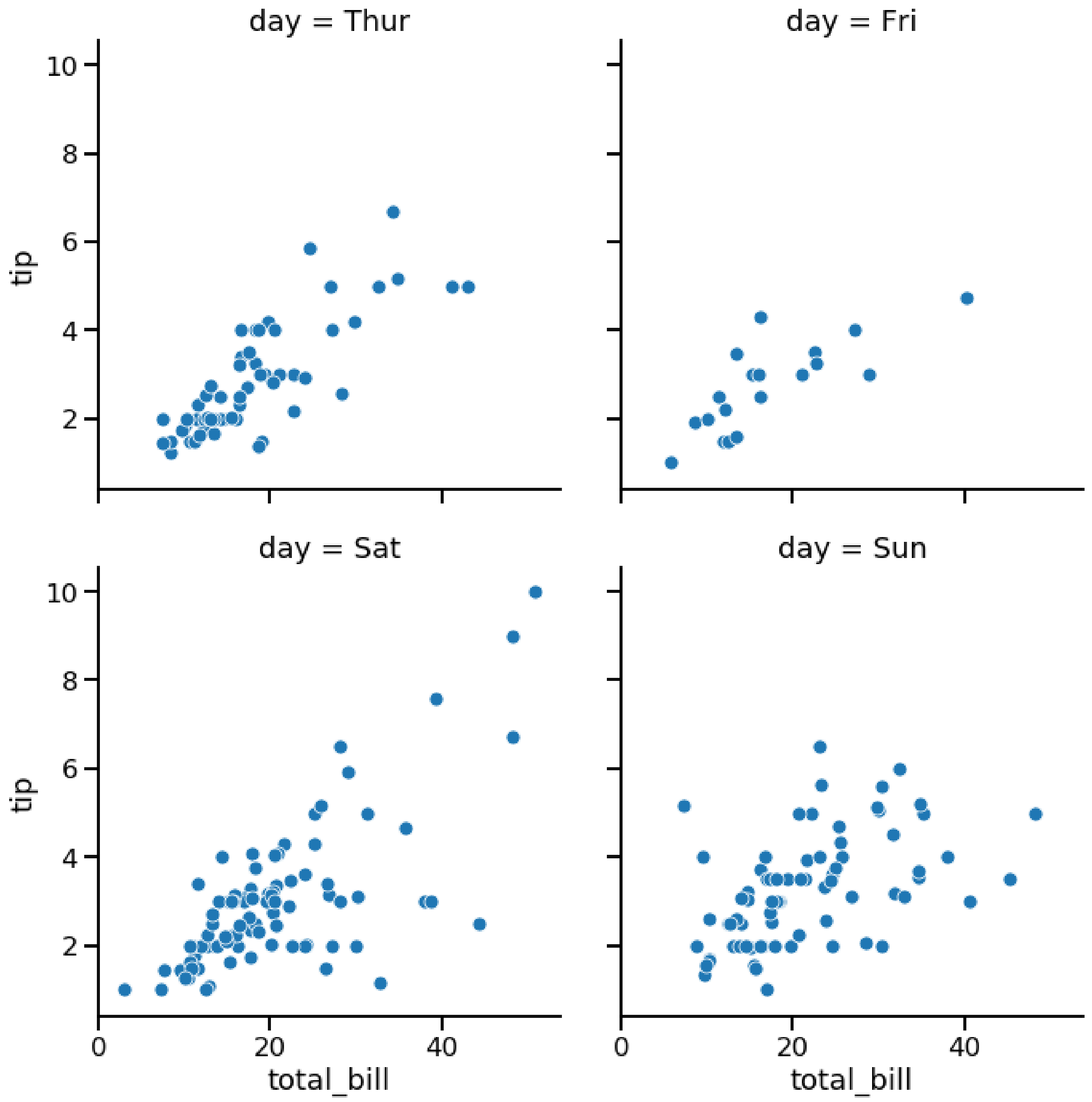


[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Ordering columns

row_order=["yes","no"] -> also exists

```python
import seaborn as sns
import matplotlib.pyplot as plt

sns.relplot(x="total_bill",
            y="tip",
            data=tips,
            kind="scatter",
            col="day",
            col_wrap=2,
            col_order=["Thur",
                       "Fri",
                       "Sat",
                       "Sun"])
plt.show()
```



[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Let's practice!

Part ·02 :

Customizing scatter plots

INTRODUCTION TO DATA VISUALIZATION WITH SEABORN

Erin Case
Data Scientist

Scatter plot overview

Show relationship between two quantitative variables

We've seen:

- Subplots (`col` and `row`)
- Subgroups with color (`hue`)

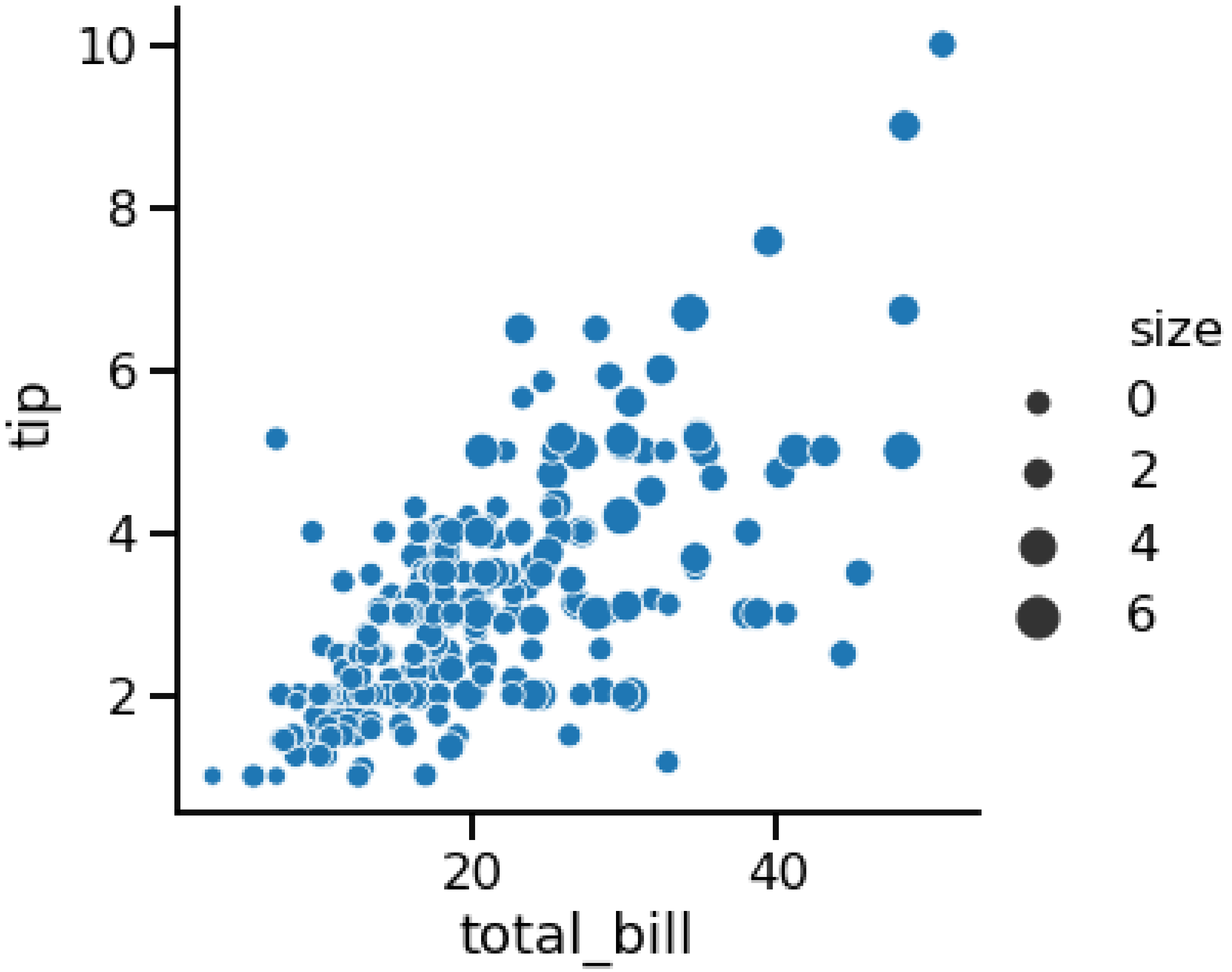
New Customizations:

- Subgroups with point size and style
- Changing point transparency

Use with both `scatterplot()` and `relplot()`

Subgroups with point size

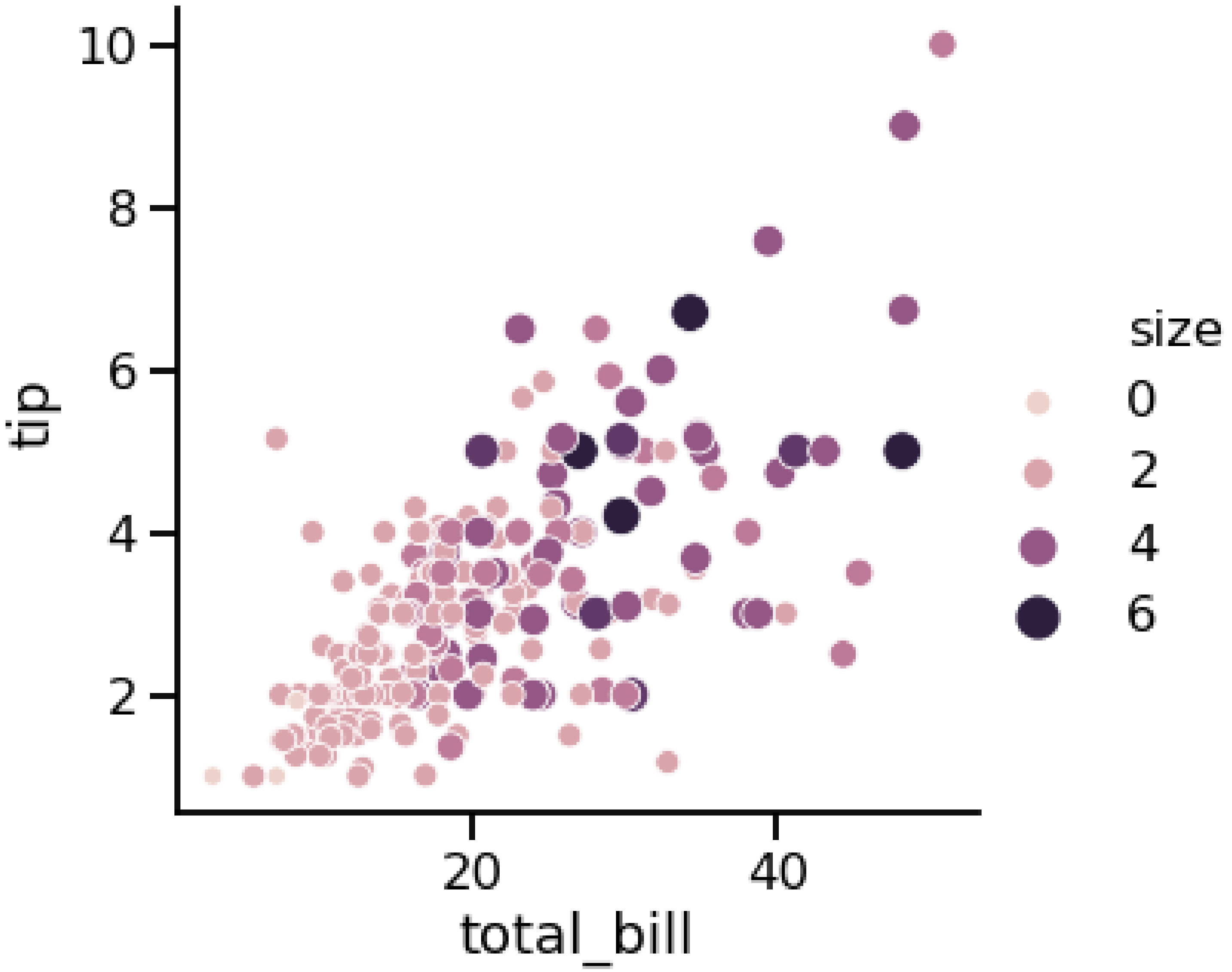
```python
import seaborn as sns
import matplotlib.pyplot as plt

sns.relplot(x="total_bill",
            y="tip",
            data=tips,
            kind="scatter",
            size="size")
plt.show()
```



[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Point size and hue

```
import seaborn as sns
import matplotlib.pyplot as plt

sns.relplot(x="total_bill",
            y="tip",
            data=tips,
            kind="scatter",
            size="size",
            hue="size")
plt.show()
```

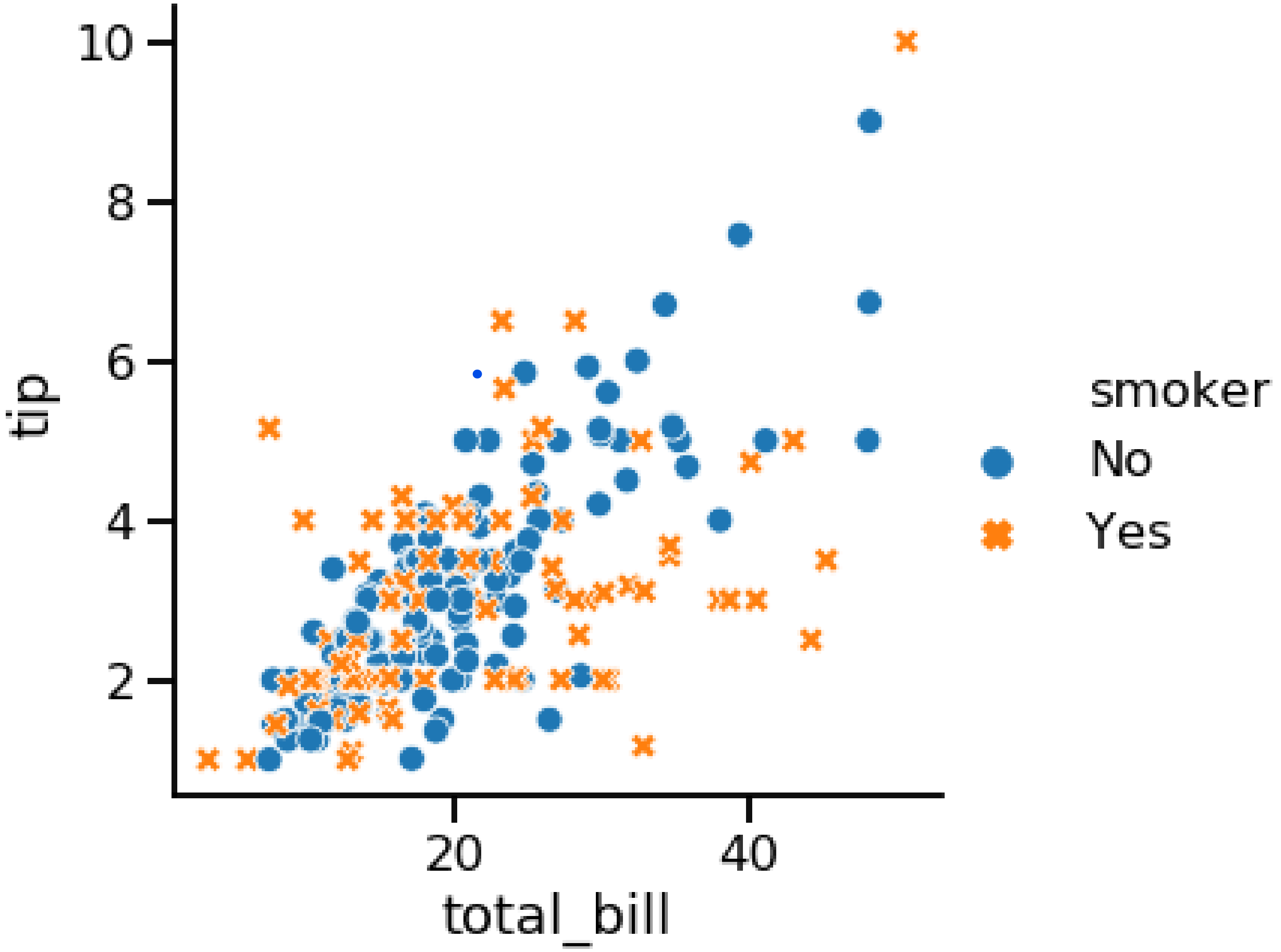


[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Subgroups with point style

```python
import seaborn as sns
import matplotlib.pyplot as plt

sns.relplot(x="total_bill",
            y="tip",
            data=tips,
            kind="scatter",
            hue="smoker",
            style="smoker")
plt.show()
```

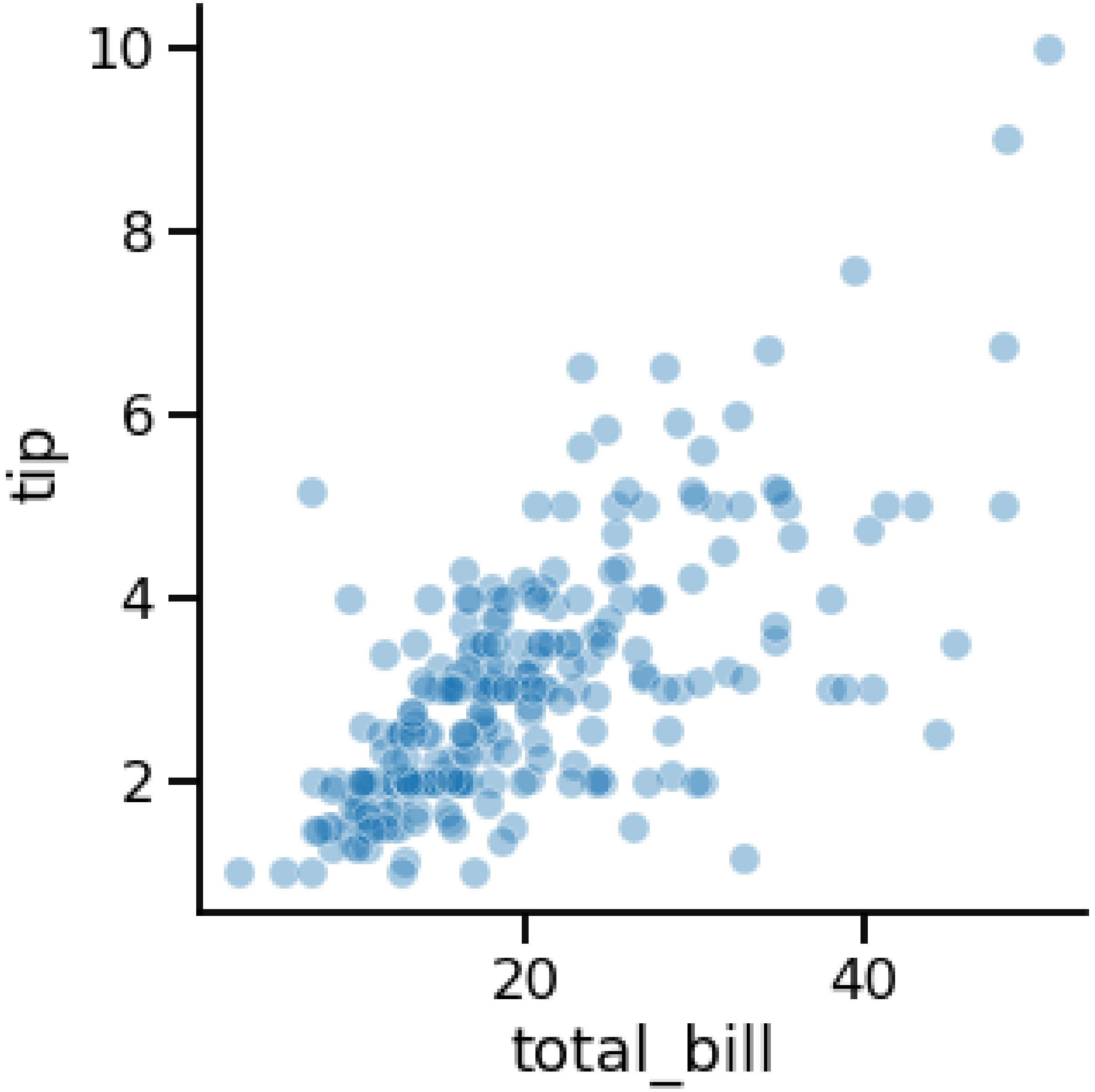
point marker shape :
smoker=Yes -> circles
smoker=No -> X marks



[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Changing point transparency

```
import seaborn as sns
import matplotlib.pyplot as plt

# Set alpha to be between 0 and 1
sns.relplot(x="total_bill",
            y="tip",
            data=tips,
            kind="scatter",
            alpha=0.4)
plt.show()
```



[1] Waskom, M. L. (2021). seaborn: statistical data visualization. https://seaborn.pydata.org/

Let's practice!

INTRODUCTION TO DATA VISUALIZATION WITH SEABORN

Part -03

Introduction to line plots

INTRODUCTION TO DATA VISUALIZATION WITH SEABORN

Erin Case
Data Scientist

datacamp

What are line plots?

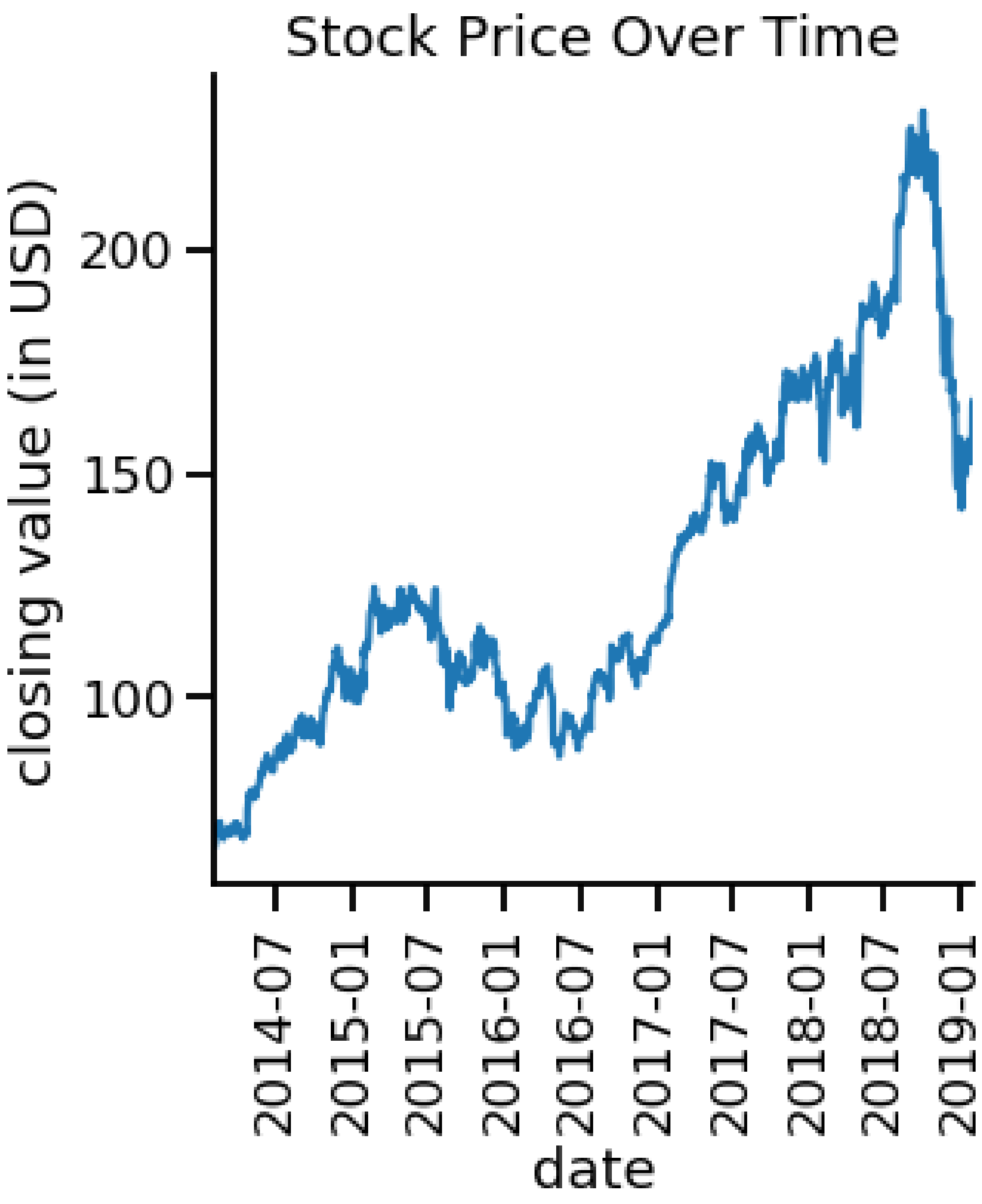
Two types of relational plots: scatter plots and line plots

Scatter plots

- Each plot point is an independent observation

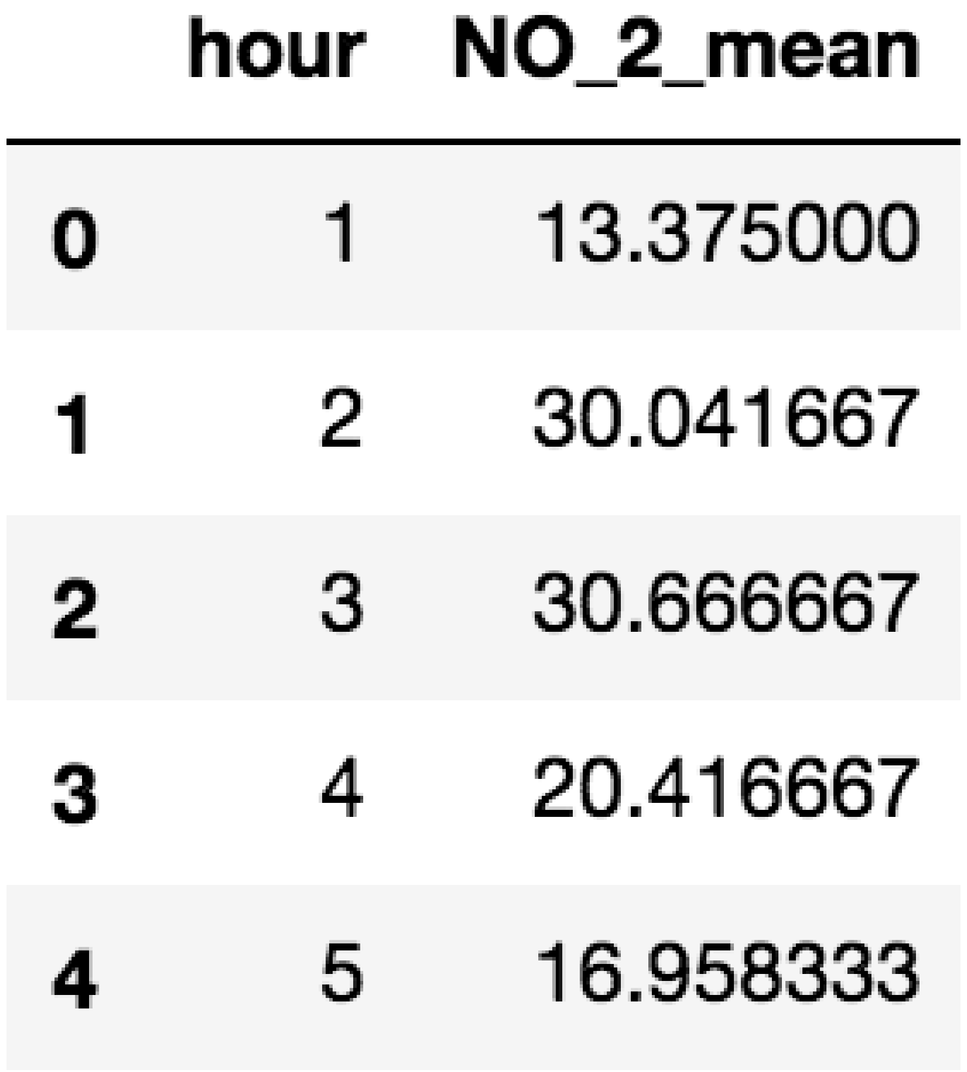
Line plots

- Each plot point represents the same "thing", typically tracked over time



Air pollution data

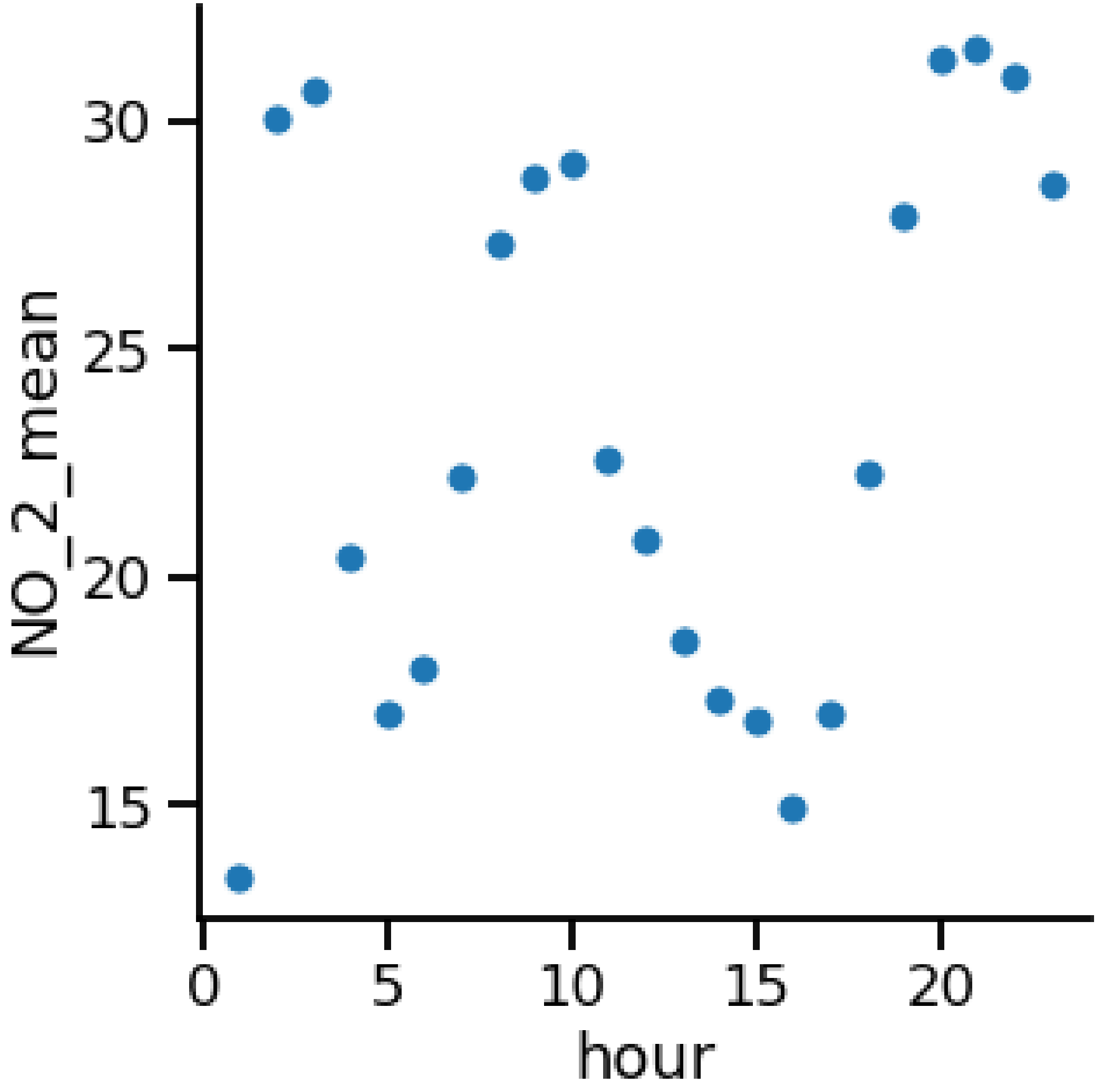
- Collection stations throughout city
- Air samples of nitrogen dioxide levels



	hour	NO_2_mean
0	1	13.375000
1	2	30.041667
2	3	30.666667
3	4	20.416667
4	5	16.958333

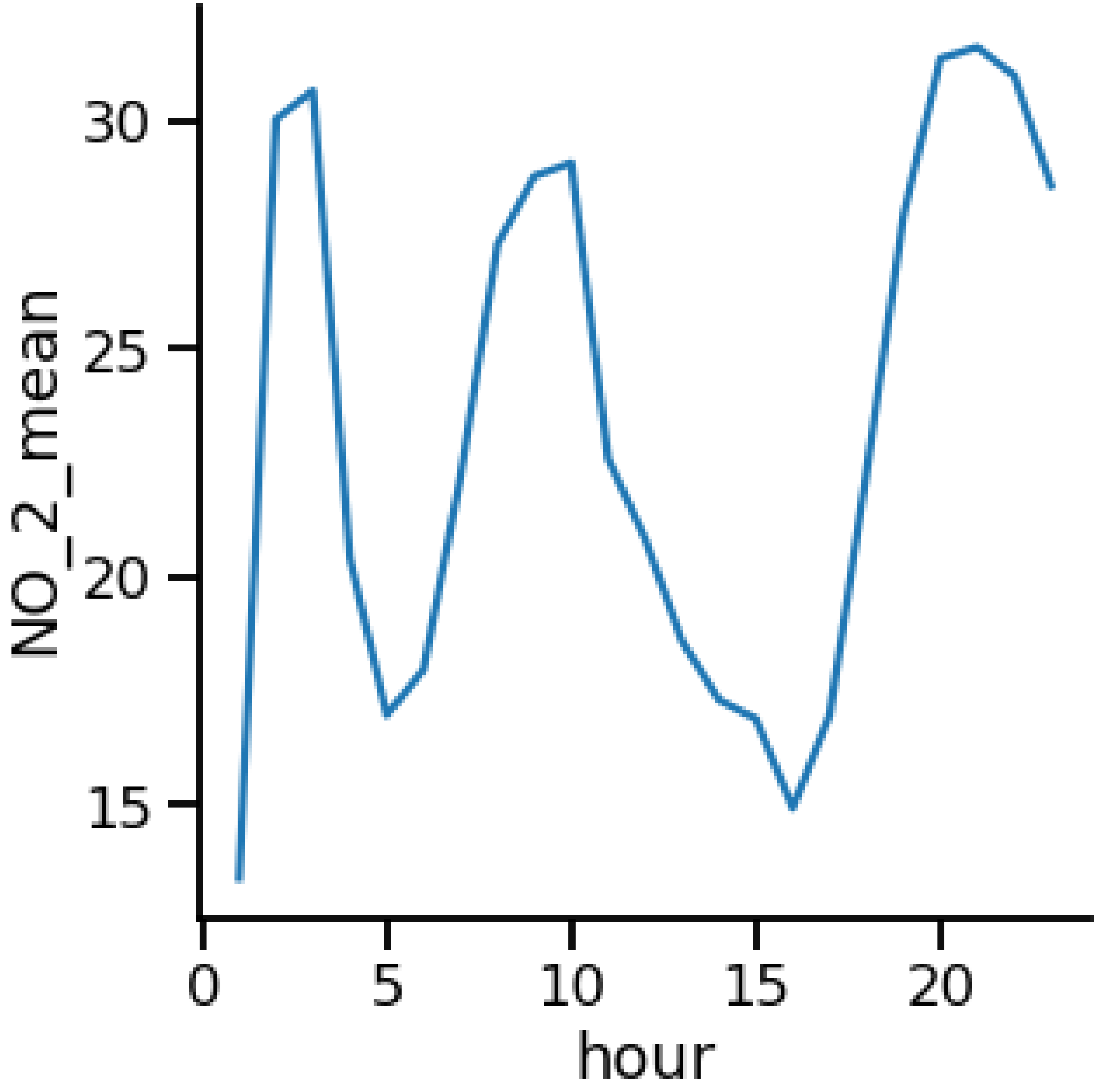
Scatter plot

```python
import matplotlib.pyplot as plt
import seaborn as sns

sns.relplot(x="hour", y="NO_2_mean",
            data=air_df_mean,
            kind="scatter")

plt.show()
```

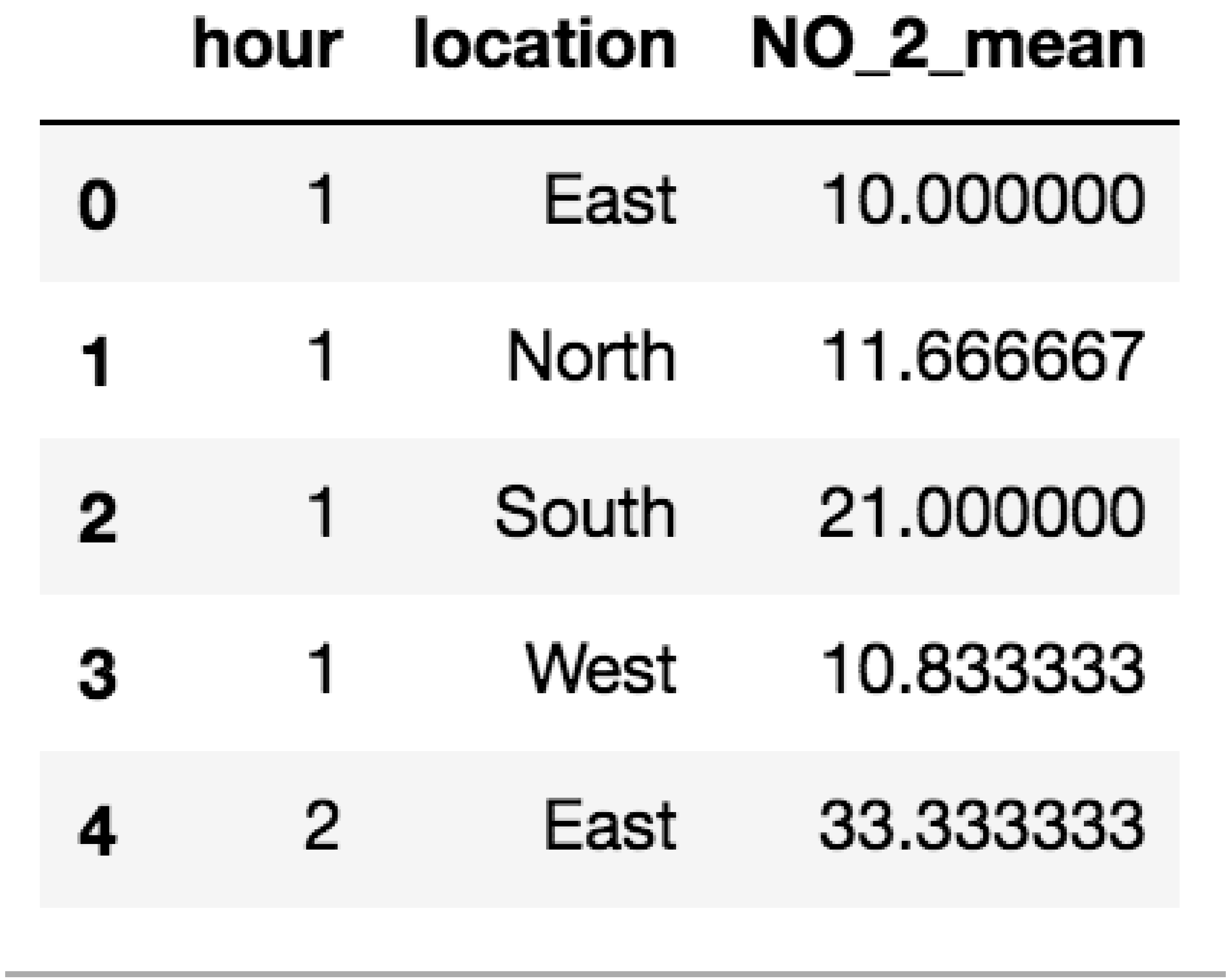


Line plot

```python
import matplotlib.pyplot as plt
import seaborn as sns

sns.relplot(x="hour", y="NO_2_mean",
            data=air_df_mean,
            kind="line")

plt.show()
```



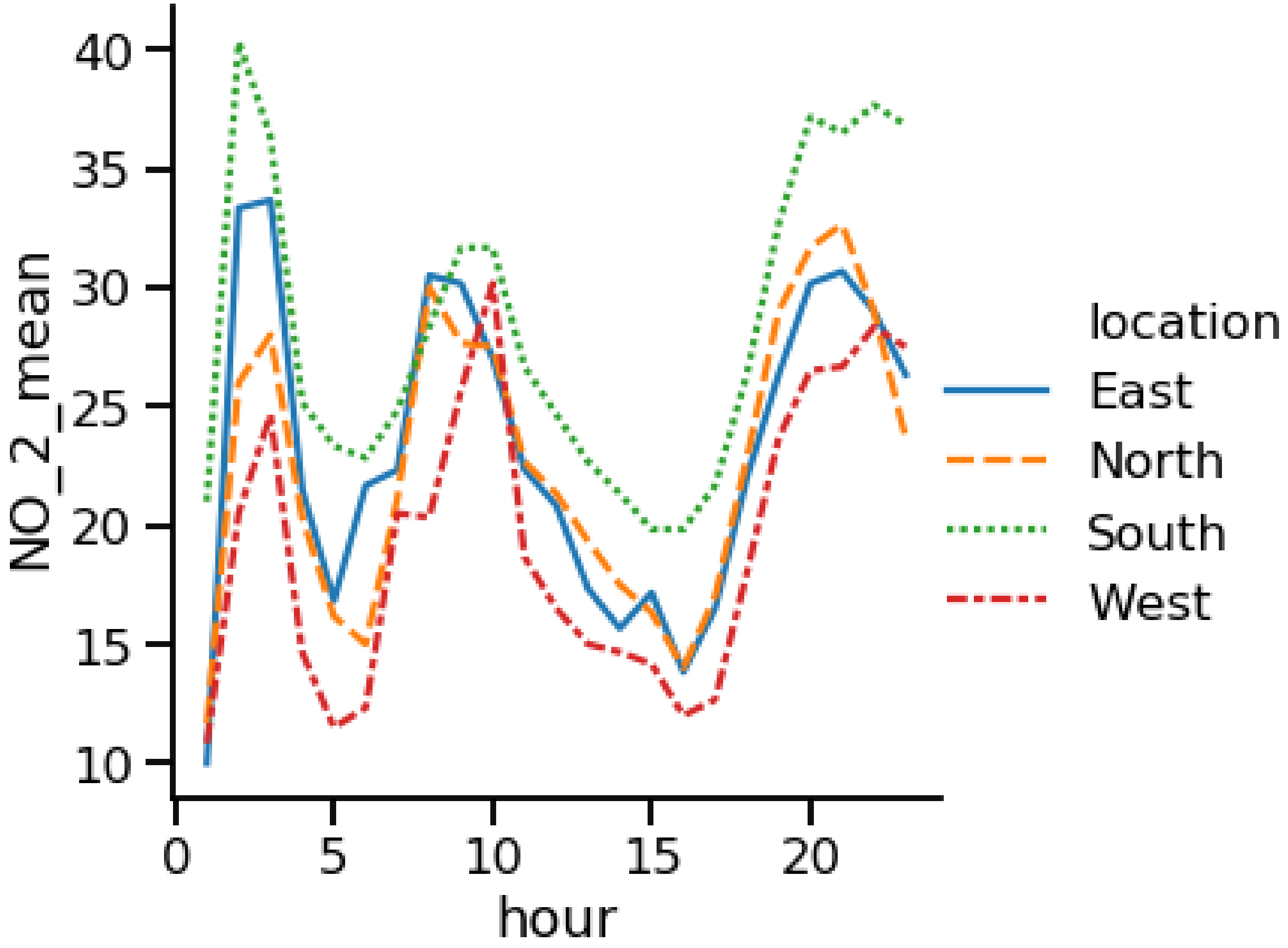
Subgroups by location



	hour	location	NO_2_mean
0	1	East	10.000000
1	1	North	11.666667
2	1	South	21.000000
3	1	West	10.833333
4	2	East	33.333333

Subgroups by location

```
import matplotlib.pyplot as plt
import seaborn as sns

sns.relplot(x="hour", y="NO_2_mean",
            data=air_df_loc_mean,
            kind="line",
            style="location",
            hue="location")
plt.show()
```

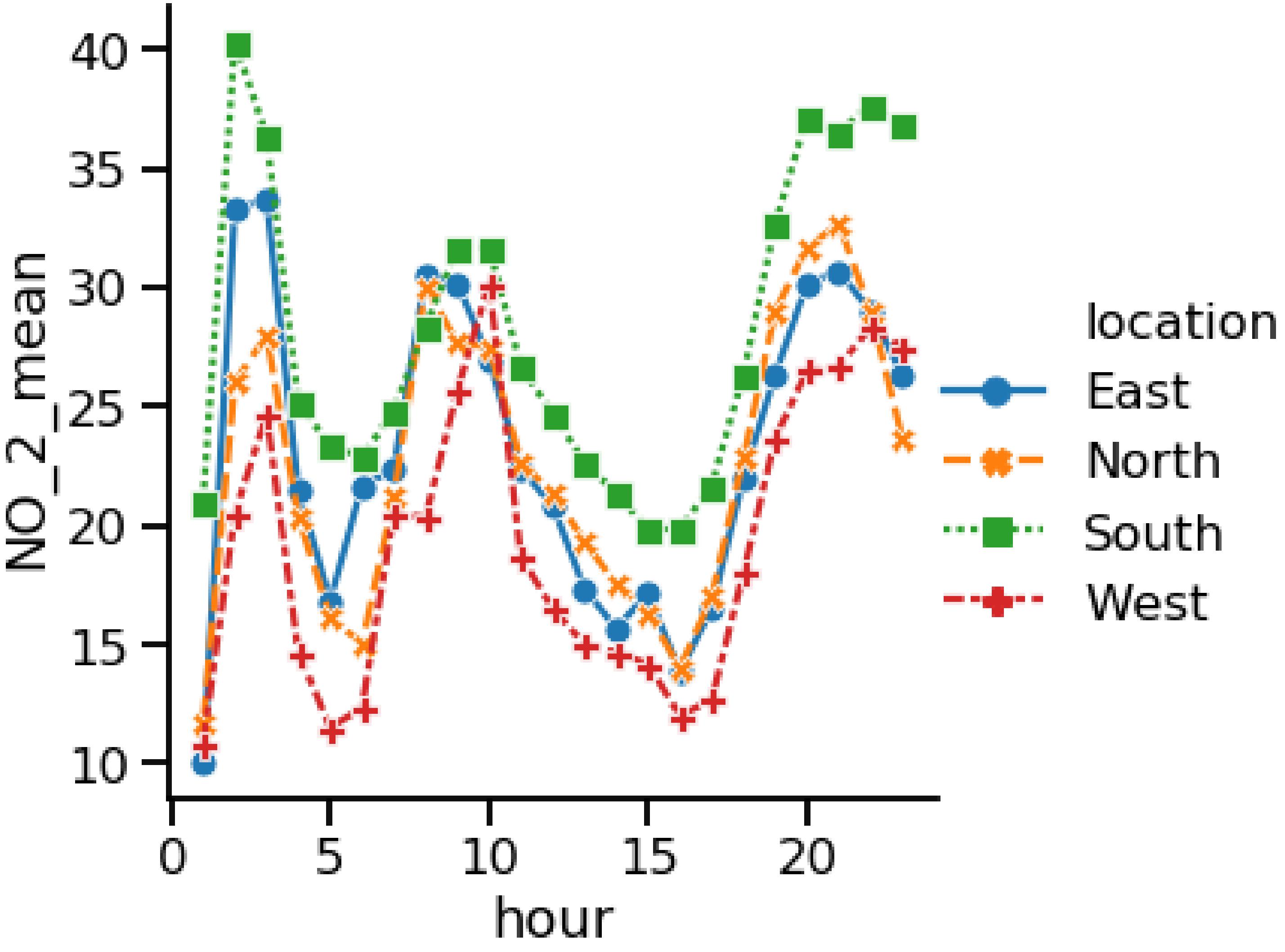


Seaborn draws one line per group of data — and that grouping is controlled by: hue, style, and/or size

Even if you removed style, you'd still get one line per location, because hue alone is enough to group the data , vice versa

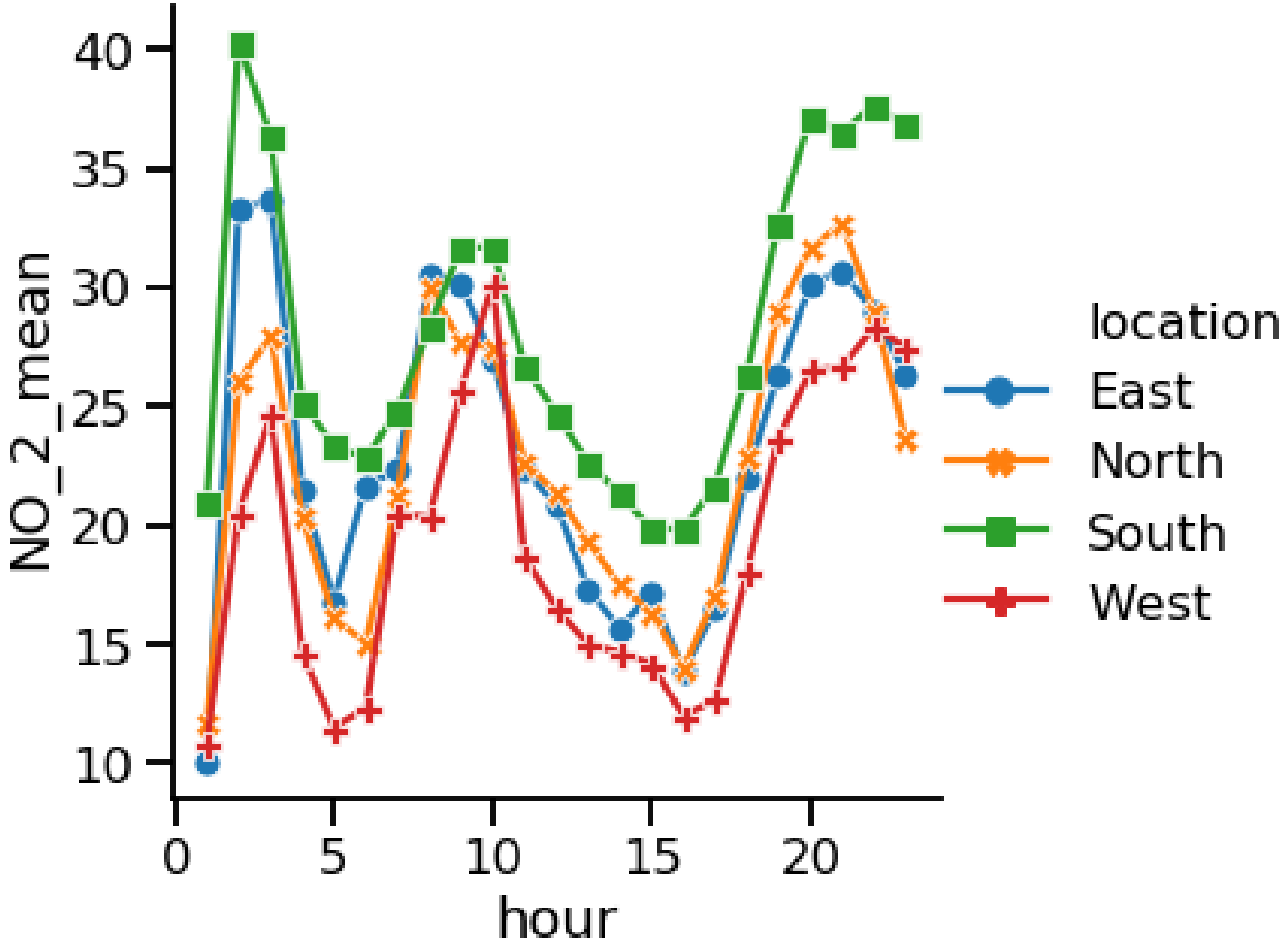
Adding markers

```python
import matplotlib.pyplot as plt
import seaborn as sns

sns.relplot(x="hour", y="NO_2_mean",
            data=air_df_loc_mean,
            kind="line",
            style="location",
            hue="location",
            markers=True)
plt.show()
```



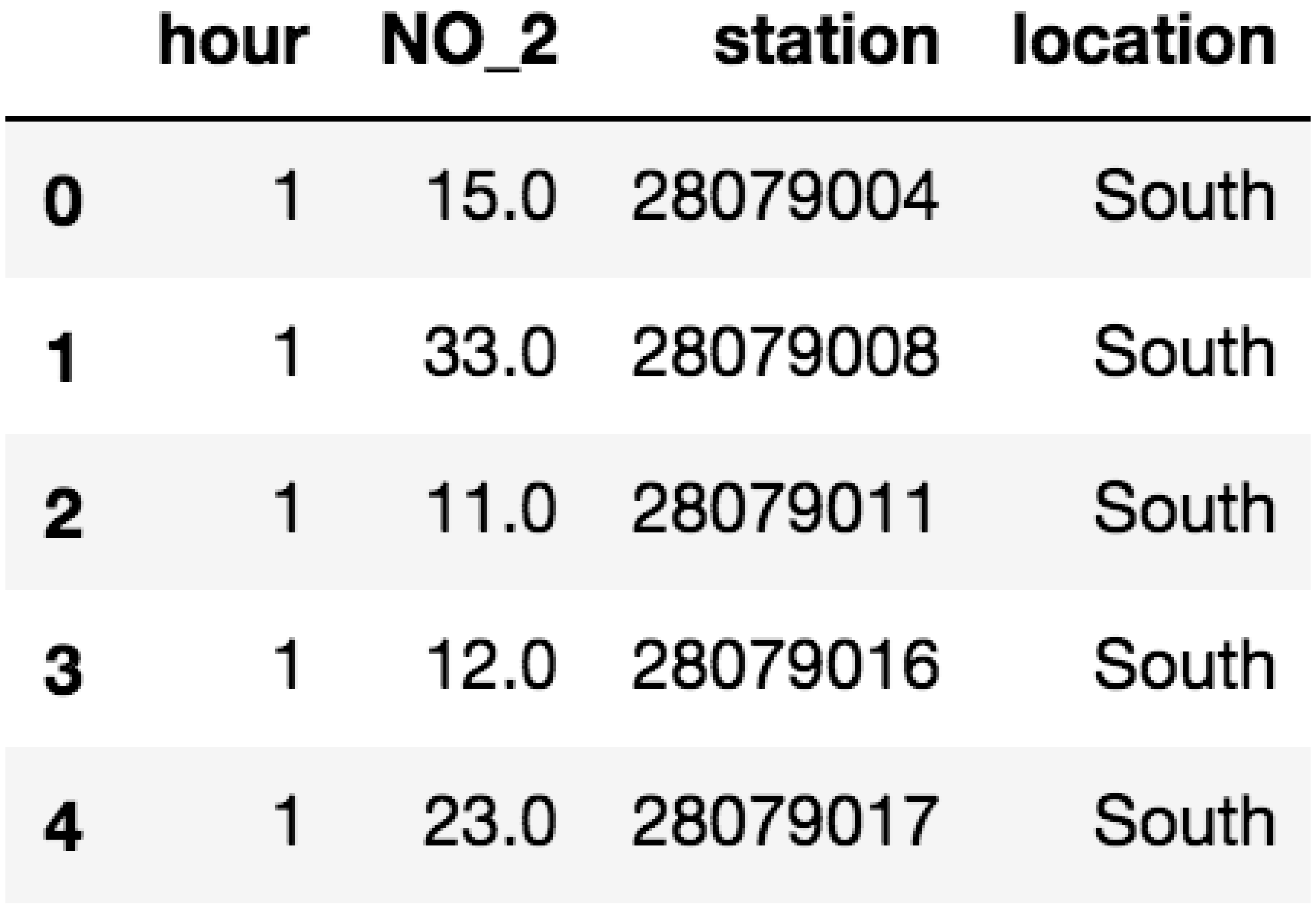
Turning off line style

```python
import matplotlib.pyplot as plt
import seaborn as sns

sns.relplot(x="hour", y="NO_2_mean",
            data=air_df_loc_mean,
            kind="line",
            style="location",
            hue="location",
            markers=True,
            dashes=False)
plt.show()
```



lines are not dashed

Multiple observations per x-value

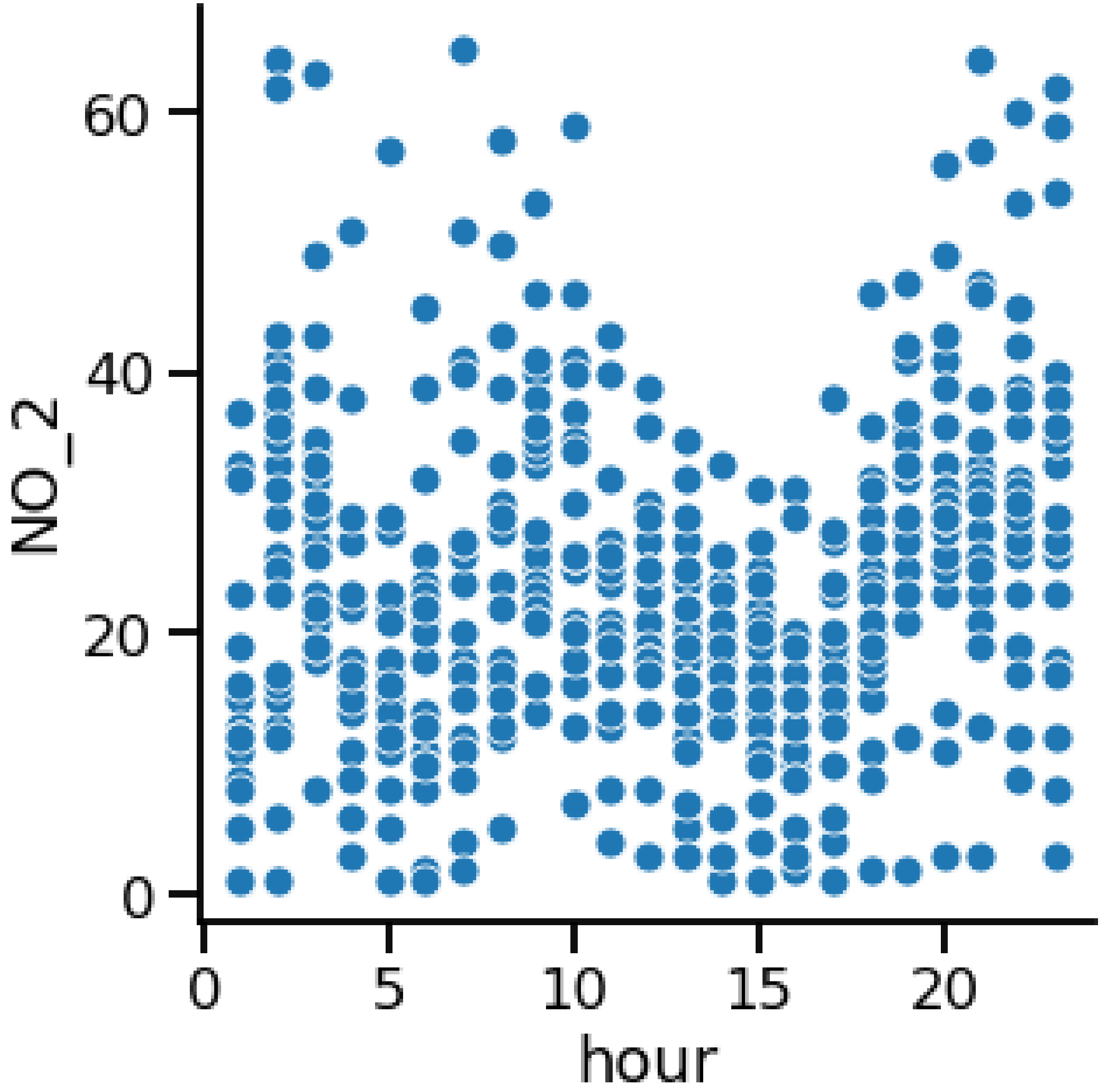


	hour	NO_2	station	location
0	1	15.0	28079004	South
1	1	33.0	28079008	South
2	1	11.0	28079011	South
3	1	12.0	28079016	South
4	1	23.0	28079017	South

Multiple observations per x-value

Scatter plot

```
import matplotlib.pyplot as plt
import seaborn as sns

sns.relplot(x="hour", y="NO_2",
            data=air_df,
            kind="scatter")

plt.show()
```

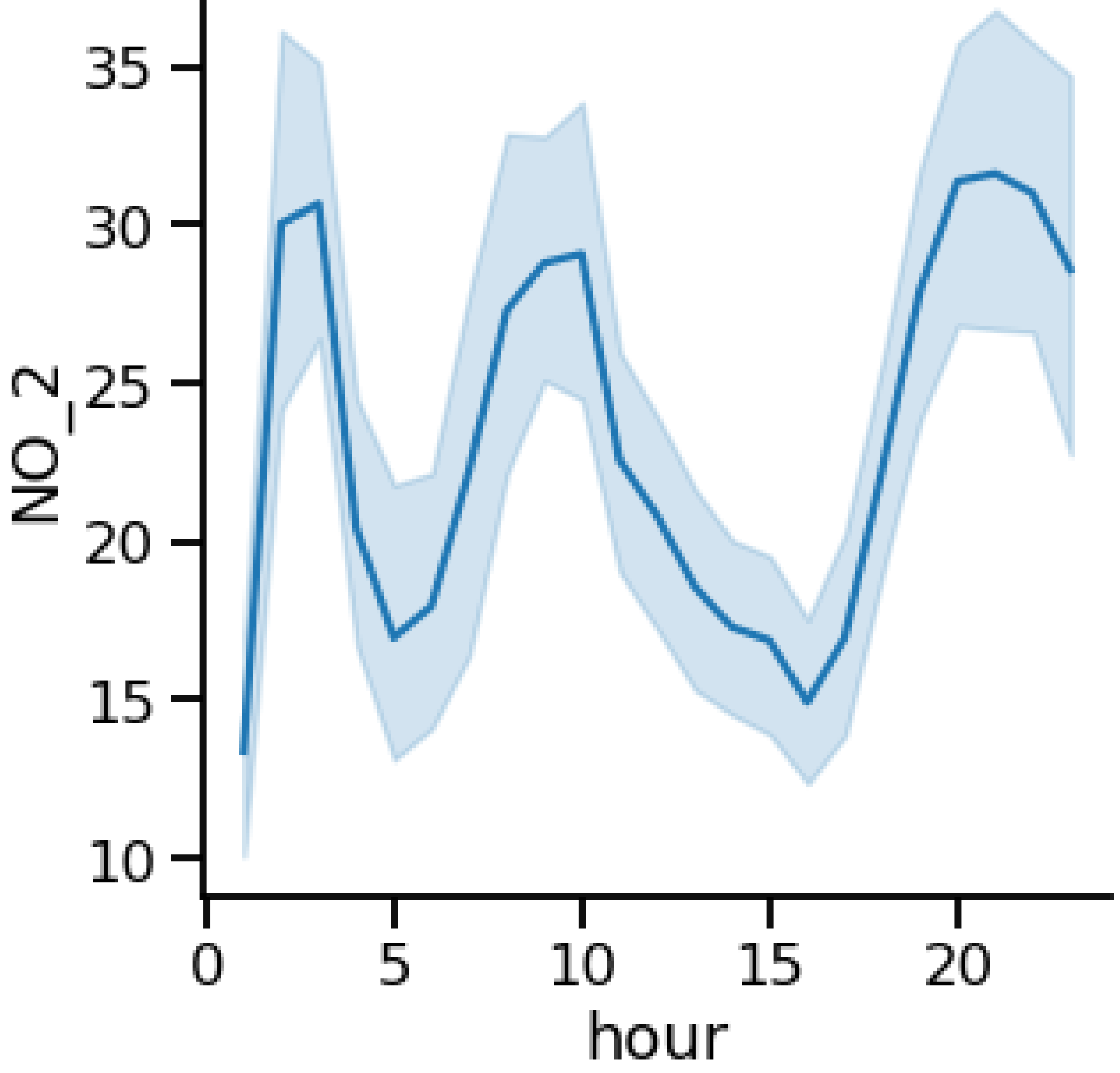


Multiple observations per x-value

Line plot

if there are multiple y for a single x , line plot uses the mean by default

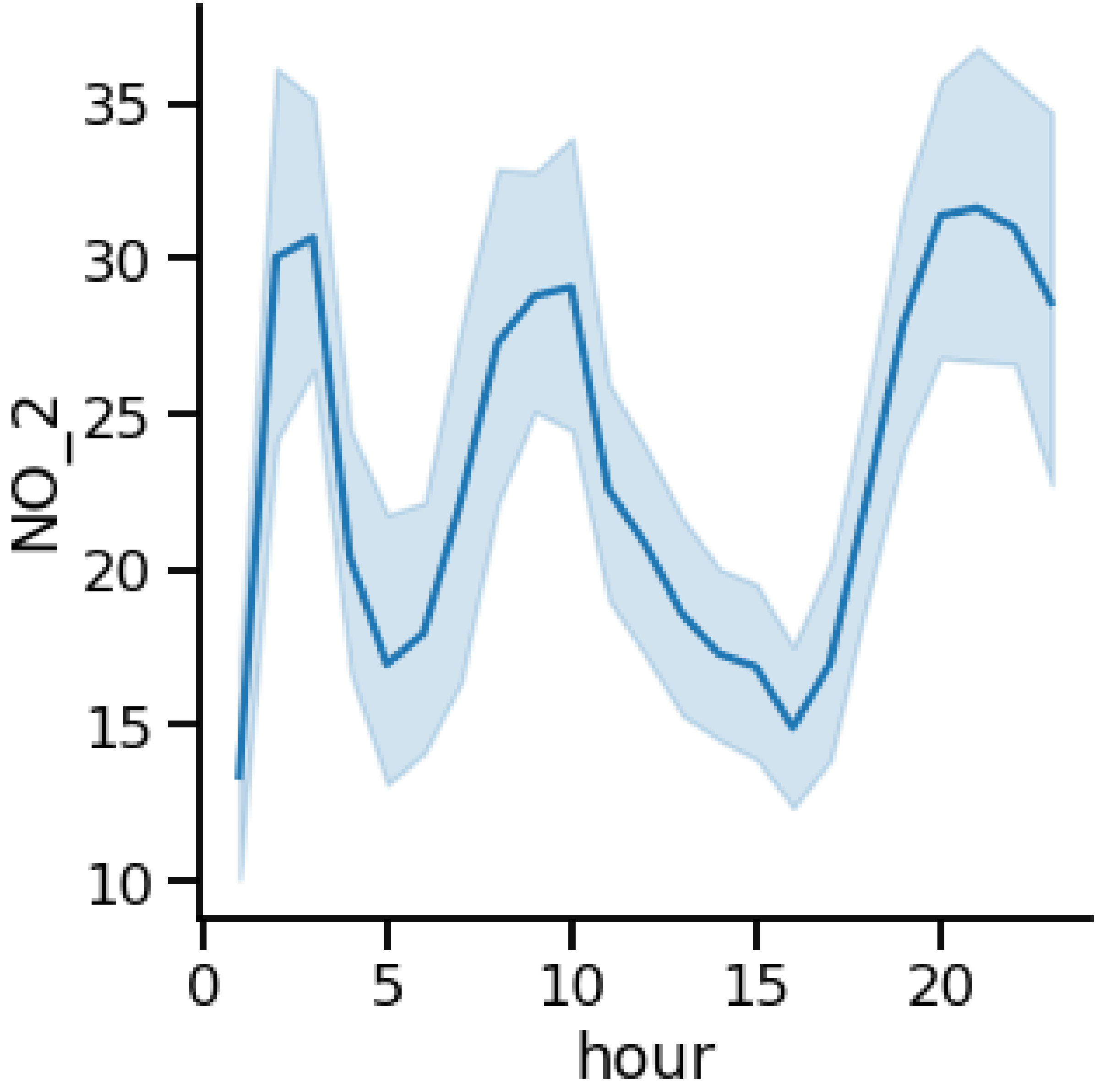
```
import matplotlib.pyplot as plt
import seaborn as sns

sns.relplot(x="hour", y="NO_2",
            data=air_df,
            kind="line")

plt.show()
```



Multiple observations per x-value

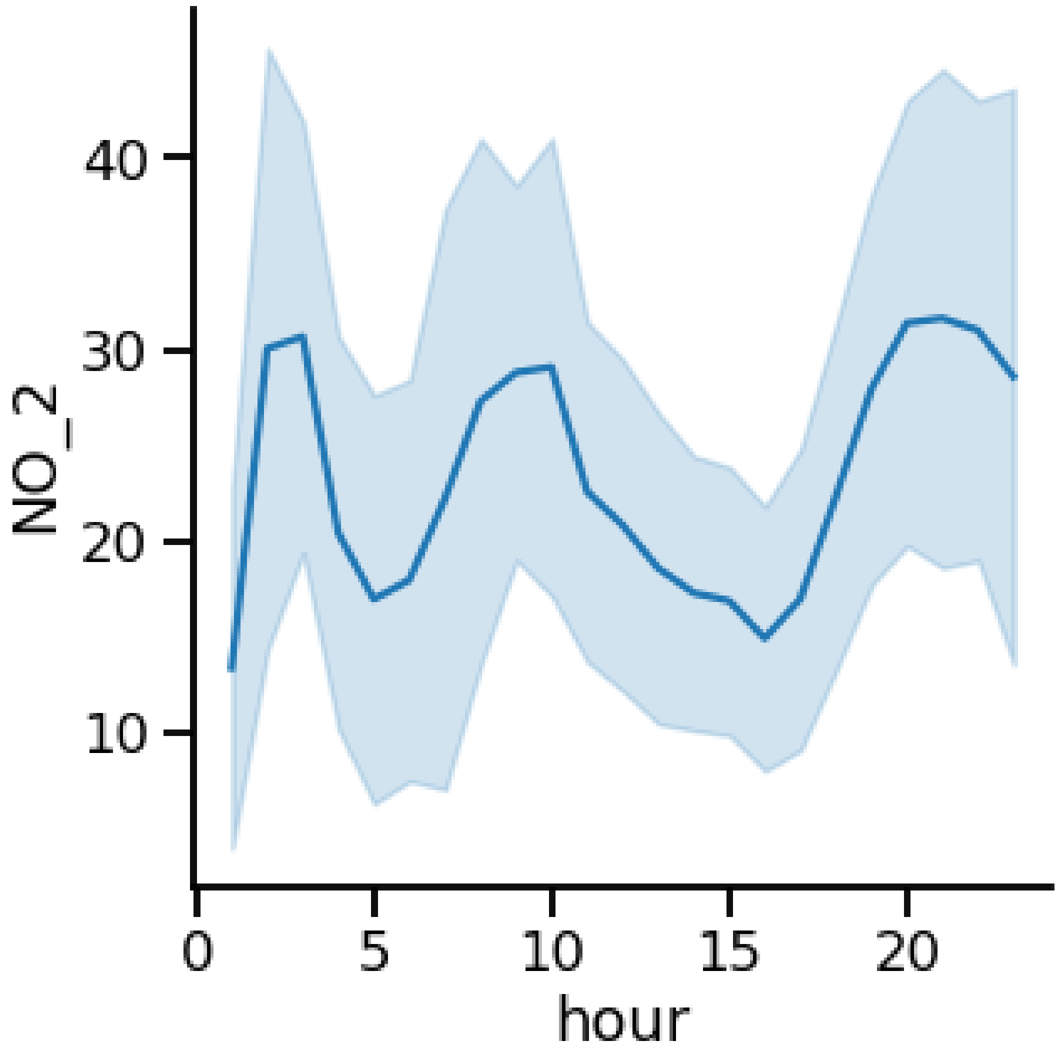
Shaded region is the confidence interval

- Assumes dataset is a random sample
- 95% confident that the mean is within this interval default ci = 95
- Indicates uncertainty in our estimate



Replacing confidence interval with standard deviation

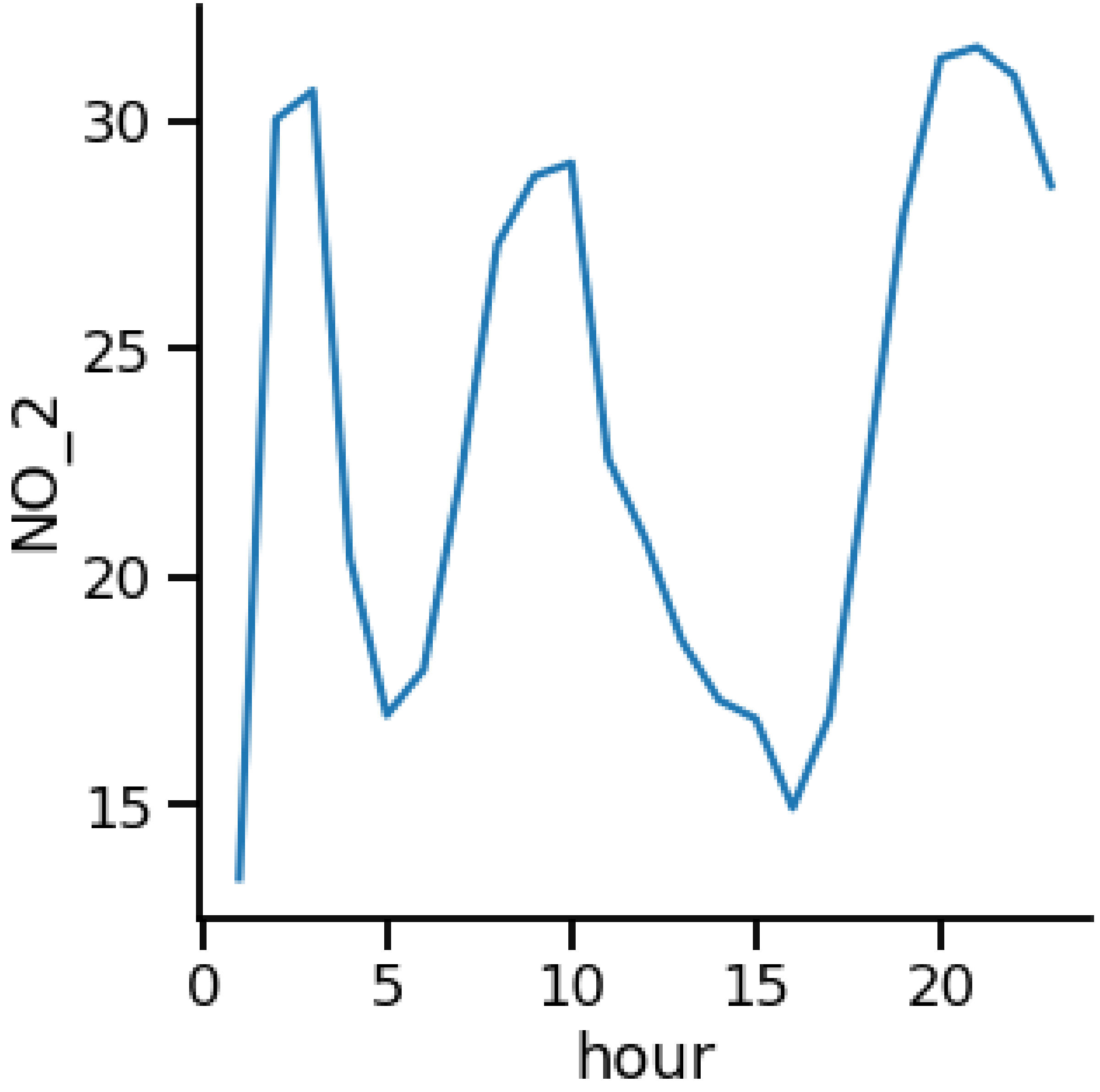
```
import matplotlib.pyplot as plt
import seaborn as sns

sns.relplot(x="hour", y="NO_2",
            data=air_df,
            kind="line",
            ci="sd")
plt.show()
```



default ci=95 ,iit means ->“If we repeated this measurement many times, 95% of the time the true mean would fall within this shaded region.”

Turning off confidence interval

```
import matplotlib.pyplot as plt
import seaborn as sns

sns.relplot(x="hour", y="NO_2",
            data=air_df,
            kind="line",
            ci=None)
plt.show()
```



Let's practice!

INTRODUCTION TO DATA VISUALIZATION WITH SEABORN