Making Predictions with Data

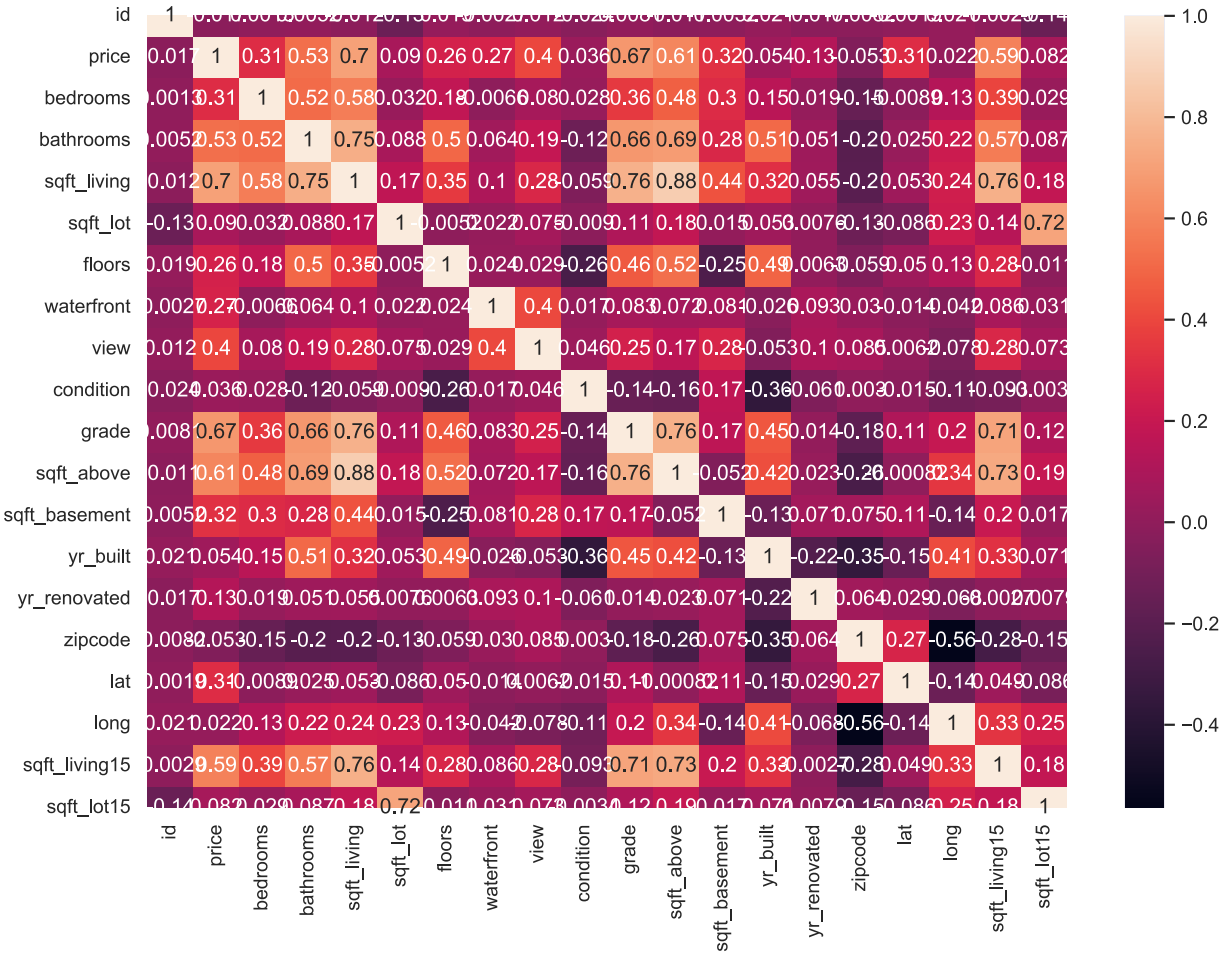
In [ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

data = pd.read_csv("../housing_data.csv")
```

In this lesson, we're going to put together all of the data science skills we've picked up so far to see how we can use machine learning to make predictions about the future or about hypothetical scenarios using our data!

Specifically, we're going to be learning how to create an algorithm that predicts what a house in Kansas City would cost based on one trait of the house. And to figure out that trait, we're going to start by taking a look at the heatmap plot from a few lessons ago:

In [ ]:
```
sns.set(rc={'figure.figsize':(11.7,8.27)})
correlations = data.corr()
sns.heatmap(correlations, annot=True)
```

Out[ ]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fc8c9e6cfd0>
```

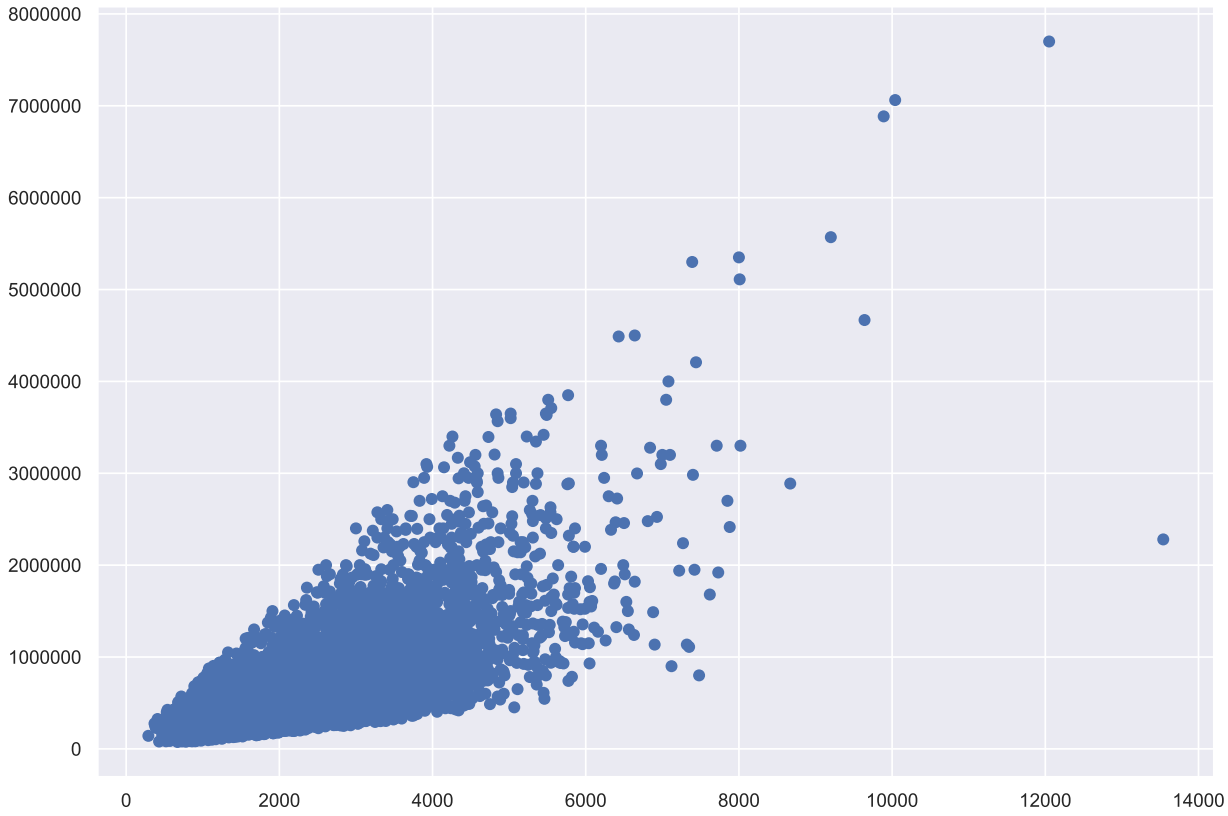


First, think back to the heatmap we covered a few lessons ago to show us the correlations and relationships between different variables in the dataset. As you might recall, the variable that had the strongest correlation with the house's price was `sqft_living`, so we're going to train our machine learning model to make the prediction off of that.

To explore this correlation a bit further, we can use a scatter plot to visualize the exact relationship between `sqft_living` and `price`:

In [ ]:
```
fig, ax = plt.subplots()

ax.scatter(data["sqft_living"], data["price"])
```

Out[ ]:
```
<matplotlib.collections.PathCollection at 0x7fc8c5ab0da0>
```



As you might be able to tell, it looks like `price` generally increases as `sqft_living` does. And, it looks like we may be able to draw a straight line on this graph that represents the average relationship between the two. if we can train our machine learning model to figure out what the line might be, we can use that to make predictions.

So, we will use a linear regression model. To do so, we'll import the model from a library called scikit learn, or `sklearn` for short. Then, we'll create a `LinearRegression()` object. Finally, we can train the model by giving it the data we want it to find the relationship between -- `data[['sqft_living']]` and `data['price']`.

In [ ]:
```
from sklearn.linear_model import LinearRegression

regression_model = LinearRegression()
regression_model.fit(data[['sqft_living']], data['price'])
```

Out[ ]:
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)
```

In [ ]:
```
regression_model.predict([[4600]])
```

Out[ ]:
```
array([1247287.66923379])
```