Visualization_II_Class5-Completed

March 2, 2021

1. import the necessary packages

[2]:
```
import warnings
warnings.filterwarnings('ignore')

from plotnine import *
import pandas as pd
import numpy as np
```

2. Use heart disease data set to build a BAD graph ("https://raw.githubusercontent.com/cmparlettpelleriti/CPSC392ParlettPelleriti/master/Data
   - have fun with it! What could you make worse? Add visual clutter? Reduce Contrast? Make it inaccessible? Make the message difficult to understand?
   - Talk in your Breakout groups about WHY these things make the graph bad.

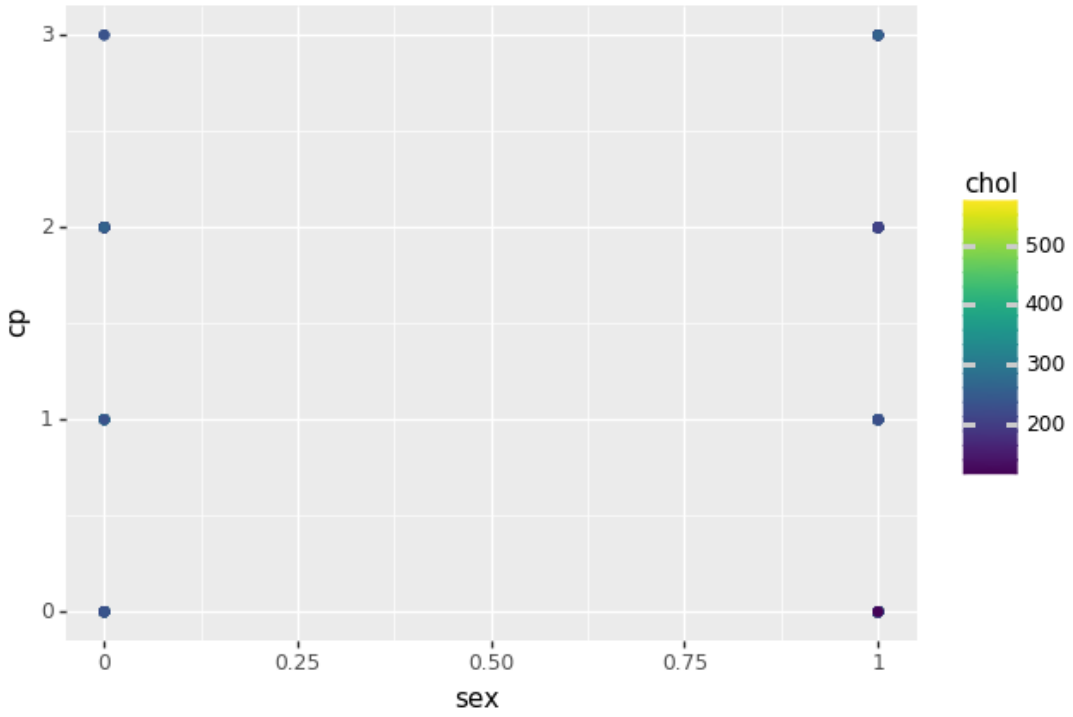
[3]:
```
heart = pd.read_csv("https://raw.githubusercontent.com/cmparlettpelleriti/
↪CPSC392ParlettPelleriti/master/Data/heart.csv")

heart.head()
```

[3]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

[4]:
```
ggplot(heart, aes(x = "sex", y = "cp", color = "chol")) + geom_point()
```

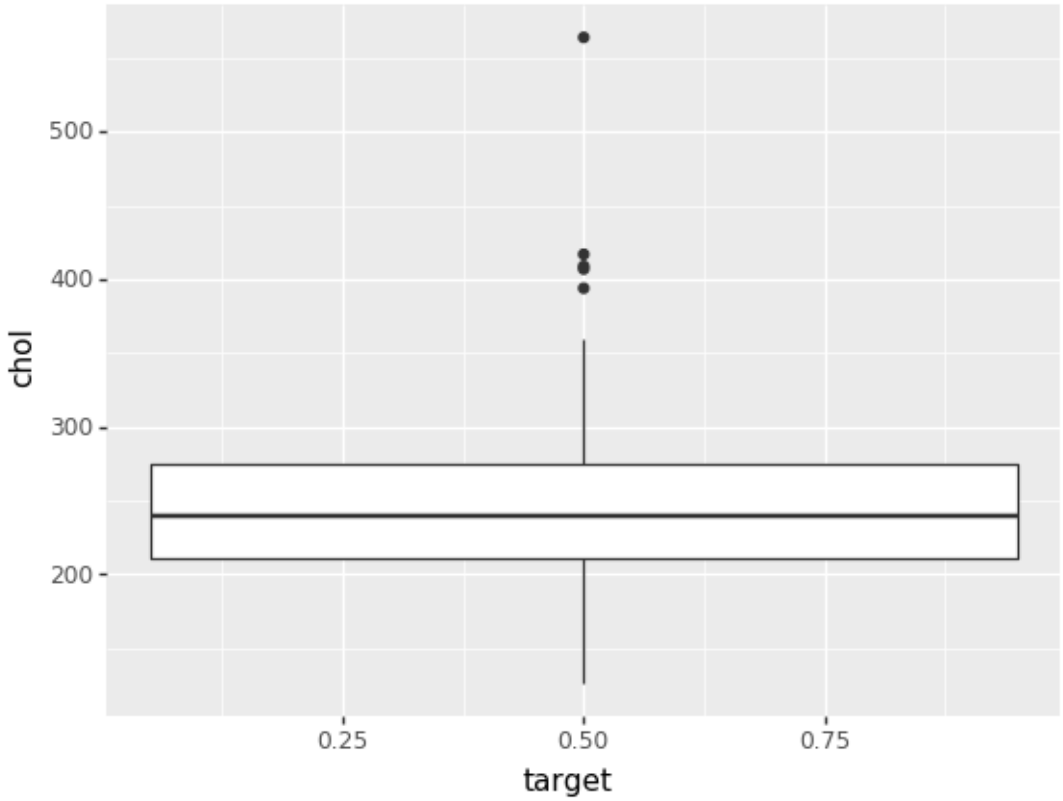


[4]: <ggplot: (8764929572512)>

This is bad because there's no title, we're using points to represent categorical data, and there's a ton of visual clutter

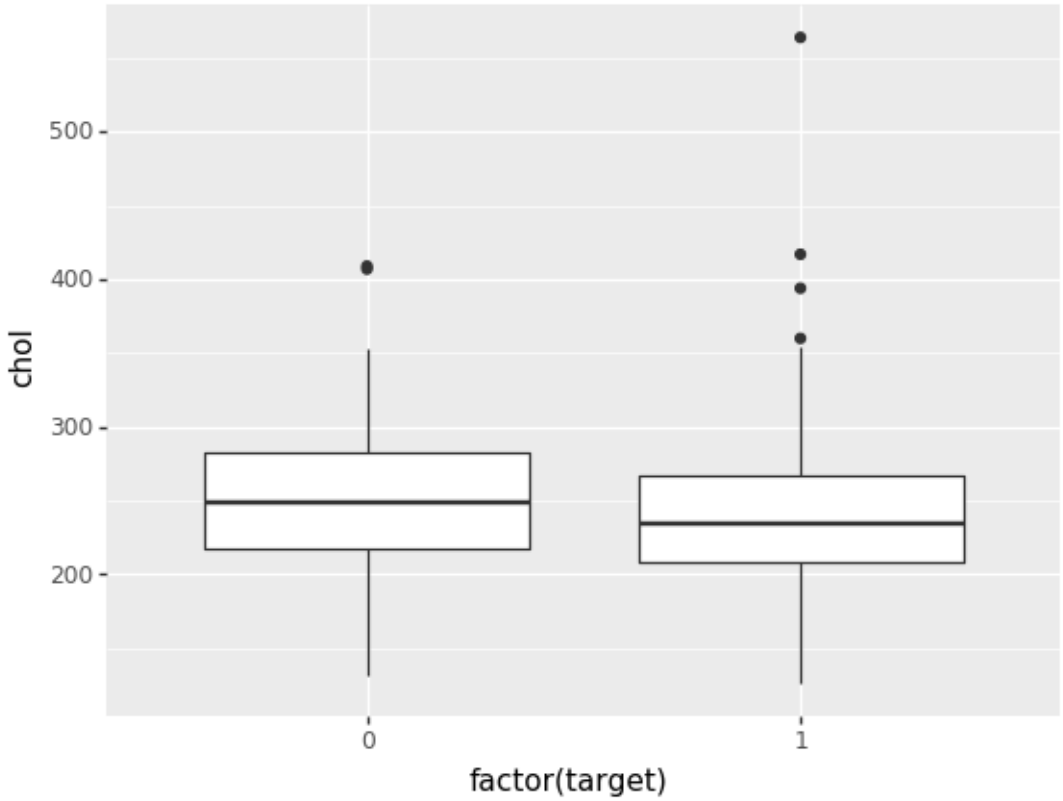
3. Use heart disease data set to answer the question of whether people with heart disease (people with heart disease have a value of `1` for the variable `target`) have higher cholesterol than people without heart disease. Create your graph one step at a time, starting with a default `ggplot() + geom_XXX()` type of graph and build from there, adding markdown cells to explain your reasoning for making changes. Think about the principles we talked about:
    - which elements do NOT support your message? Take them out (you can google ggplot/plotnine syntax to figure out how to remove or change things like text, grids, axes, legends…etc)
    - which elements DO support your message? How can you make these more noticeable/salient?
    - Who is your audience? How can you make your graph more inclusive and accessible?

[5]:
```
ggplot(heart, aes(x = "target", y = "chol")) + geom_boxplot()
```



[5]: <ggplot: (8764929748649)>

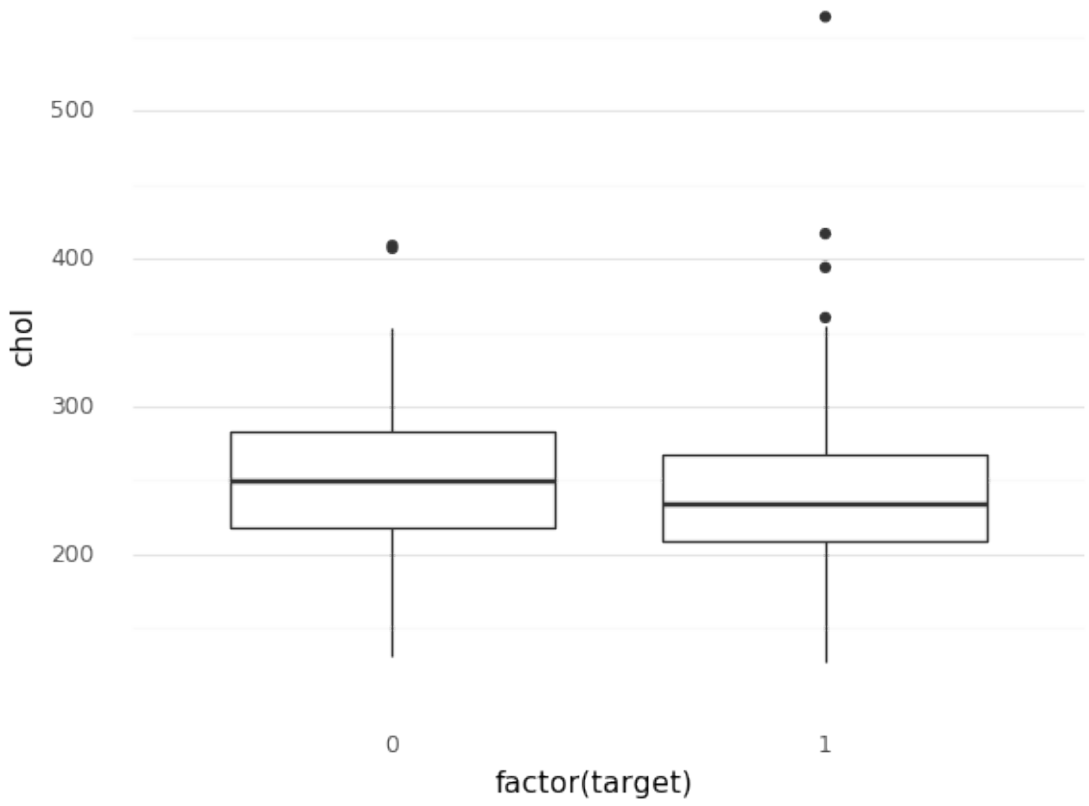
[6]:
```
ggplot(heart, aes(x = "factor(target)", y = "chol")) + geom_boxplot()
```



[6]: <ggplot: (8764929590640)>

listed target as a factor to get two boxplots

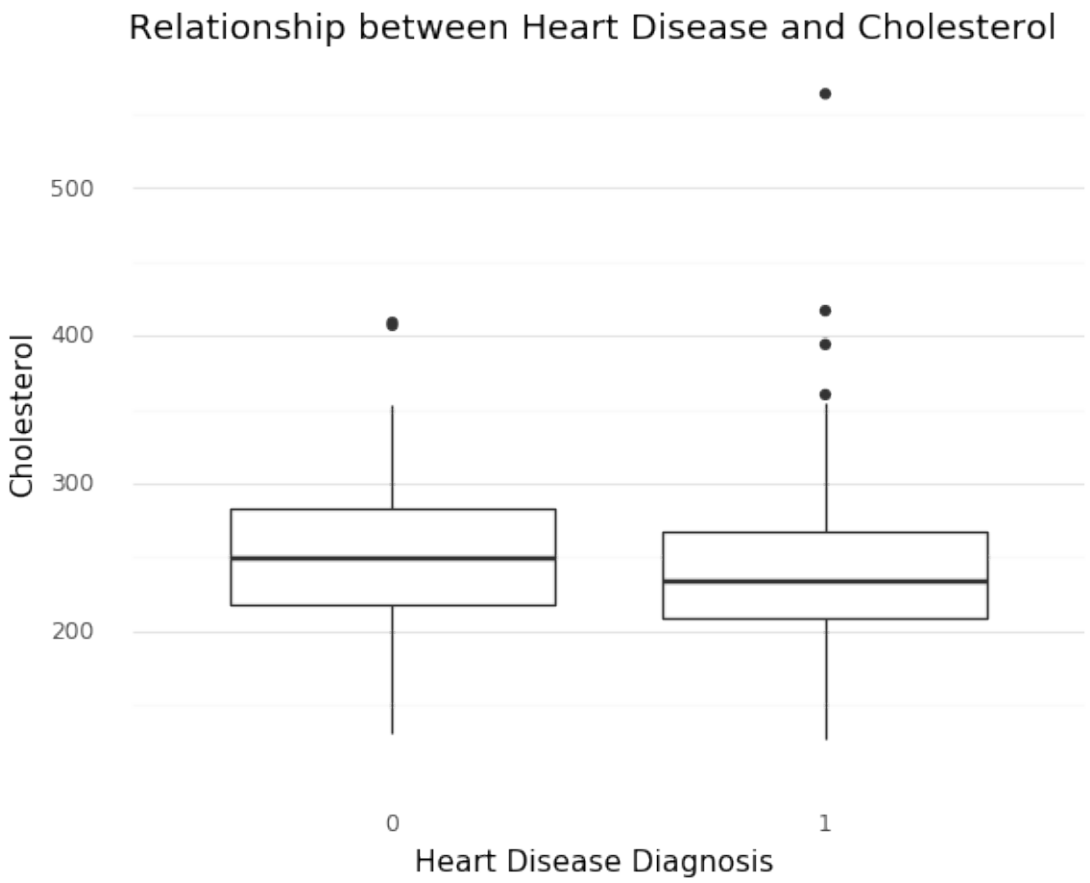
[7]:
```
(ggplot(heart, aes(x = "factor(target)", y = "chol")) + geom_boxplot() +
theme_minimal() + theme(panel_grid_major_x = element_blank(),
                        panel_grid_minor_x = element_blank()))
```



[7]: <ggplot: (8764929845038)>

got rid of background and ink, and got rid of gridlines for x axis since target is a category, we don't need gridlines to see values inbetween.

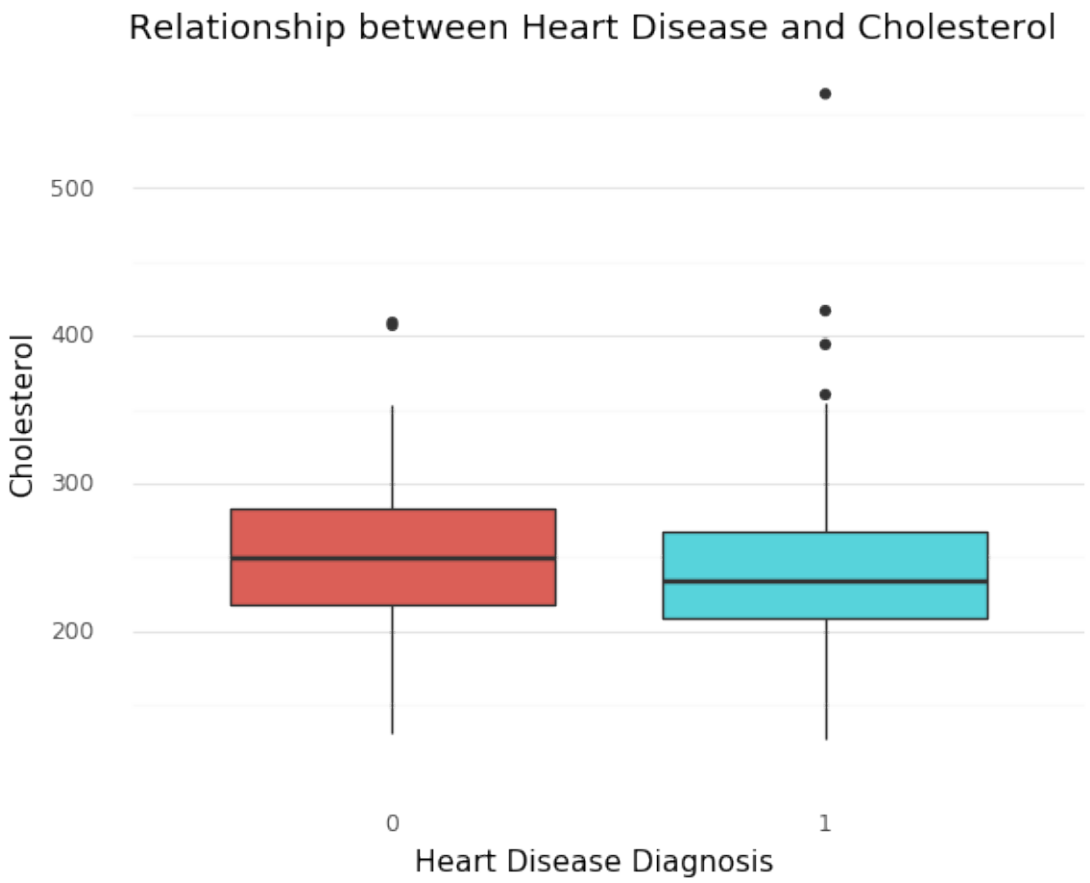
[8]:
```
(ggplot(heart, aes(x = "factor(target)", y = "chol")) + geom_boxplot() +
theme_minimal() + theme(panel_grid_major_x = element_blank(),
                        panel_grid_minor_x = element_blank()) +
labs(x = "Heart Disease Diagnosis", y = "Cholesterol", title = "Relationship␣
 ↪between Heart Disease and Cholesterol"))
```



[8]: <ggplot: (8764929572452)>

changed X and y labels, and added title so that people oculd more clearly understand the graph.

[11]:
```
(ggplot(heart, aes(x = "factor(target)", y = "chol", fill = "factor(target)"))␣
 ↪+ geom_boxplot() +
theme_minimal() + theme(panel_grid_major_x = element_blank(),
                        panel_grid_minor_x = element_blank(),
                        legend_position = "none") +
labs(x = "Heart Disease Diagnosis", y = "Cholesterol", title = "Relationship␣
 ↪between Heart Disease and Cholesterol"))
```



[11]: <ggplot: (8764925569323)>

Added color to differentiate the two boxplots since the message is that the two groups are different, but got rid of legend as it is unneeded.

4. Use the KC house data (`"https://raw.githubusercontent.com/cmparlettpelleriti/CPSC392ParlettPe` to build a clear graph that answers the question "Is the relationship between price and square footage the same for houses with different numbers of floors?". Create your graph one step at a time, starting with a default `ggplot() + geom_XXX()` type of graph and build from there, adding markdown cells to explain your reasoning for making changes. Again, think about the principles we talked about:

   - which elements do NOT support your message? Take them out (you can google ggplot/plotnine syntax to figure out how to remove or change things like text, grids, axes, legends...etc)
   - which elements DO support your message? How can you make these more noticeable/salient?
   - Who is your audience? How can you make your graph more inclusive and accessible?

[12]:
```
kc = pd.read_csv("https://raw.githubusercontent.com/cmparlettpelleriti/
↪CPSC392ParlettPelleriti/master/Data/kc_house_data.csv")

kc.head()
```

[12]:

	id	date	price	bedrooms	bathrooms	sqft_living	\
0	7129300520	20141013T000000	221900.0	3	1.00	1180	
1	6414100192	20141209T000000	538000.0	3	2.25	2570	
2	5631500400	20150225T000000	180000.0	2	1.00	770	
3	2487200875	20141209T000000	604000.0	4	3.00	1960	
4	1954400510	20150218T000000	510000.0	3	2.00	1680	

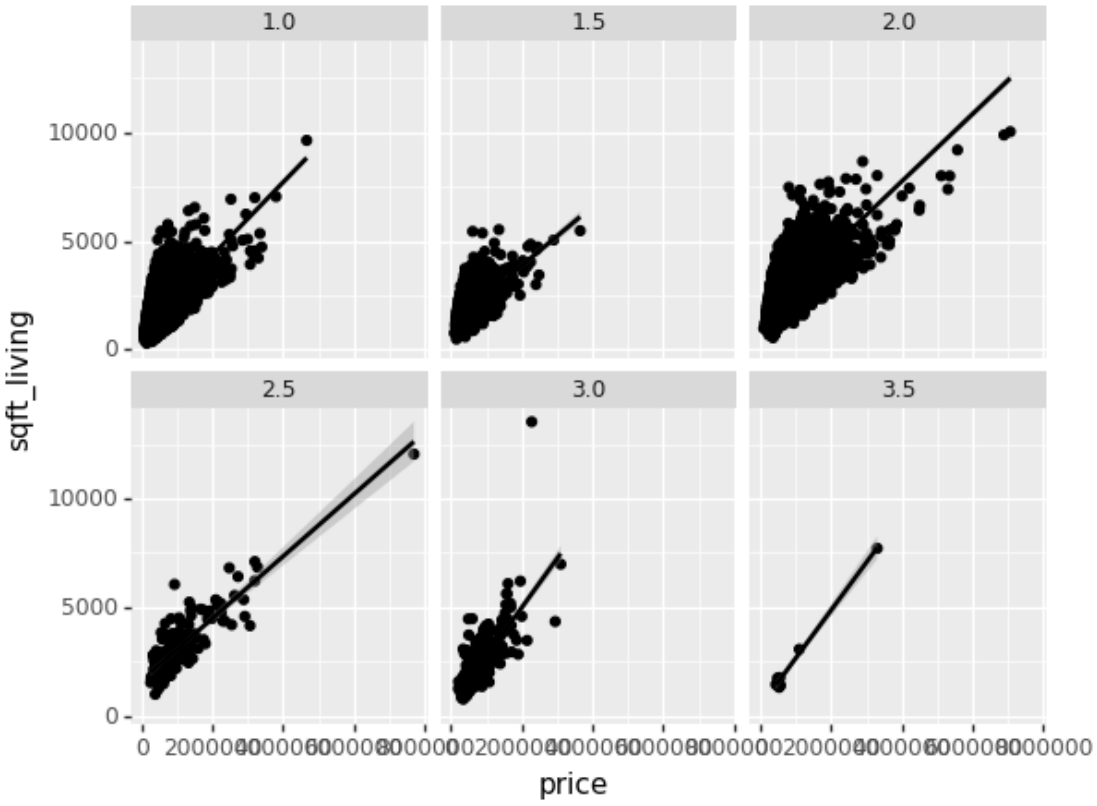
	sqft_lot	floors	waterfront	view	…	grade	sqft_above	sqft_basement	\
0	5650	1.0	0	0	…	7	1180	0	
1	7242	2.0	0	0	…	7	2170	400	
2	10000	1.0	0	0	…	6	770	0	
3	5000	1.0	0	0	…	7	1050	910	
4	8080	1.0	0	0	…	8	1680	0	

	yr_built	yr_renovated	zipcode	lat	long	sqft_living15	\
0	1955	0	98178	47.5112	-122.257	1340	
1	1951	1991	98125	47.7210	-122.319	1690	
2	1933	0	98028	47.7379	-122.233	2720	
3	1965	0	98136	47.5208	-122.393	1360	
4	1987	0	98074	47.6168	-122.045	1800	

	sqft_lot15
0	5650
1	7639
2	8062
3	5000
4	7503

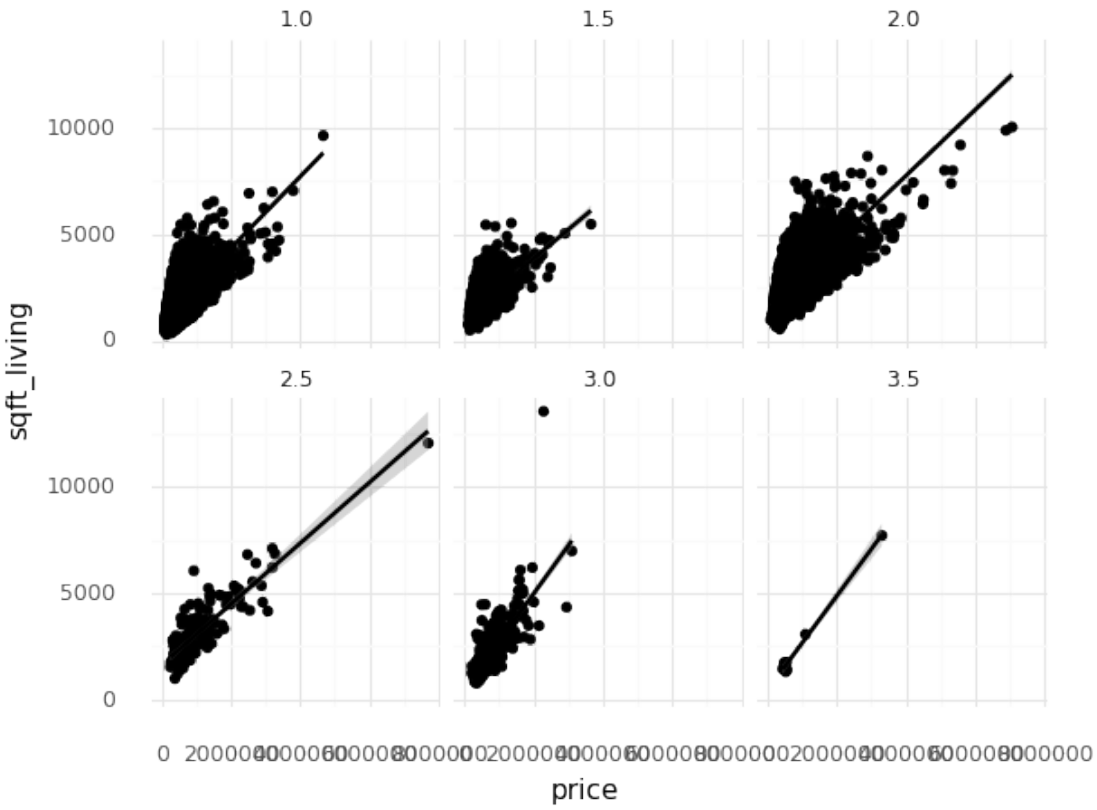
[5 rows x 21 columns]

[13]:
```
(ggplot(kc, aes("price", "sqft_living")) + geom_point() +
 geom_smooth(method = "lm") + facet_wrap("~floors"))
```



[13]: <ggplot: (8764930688528)>

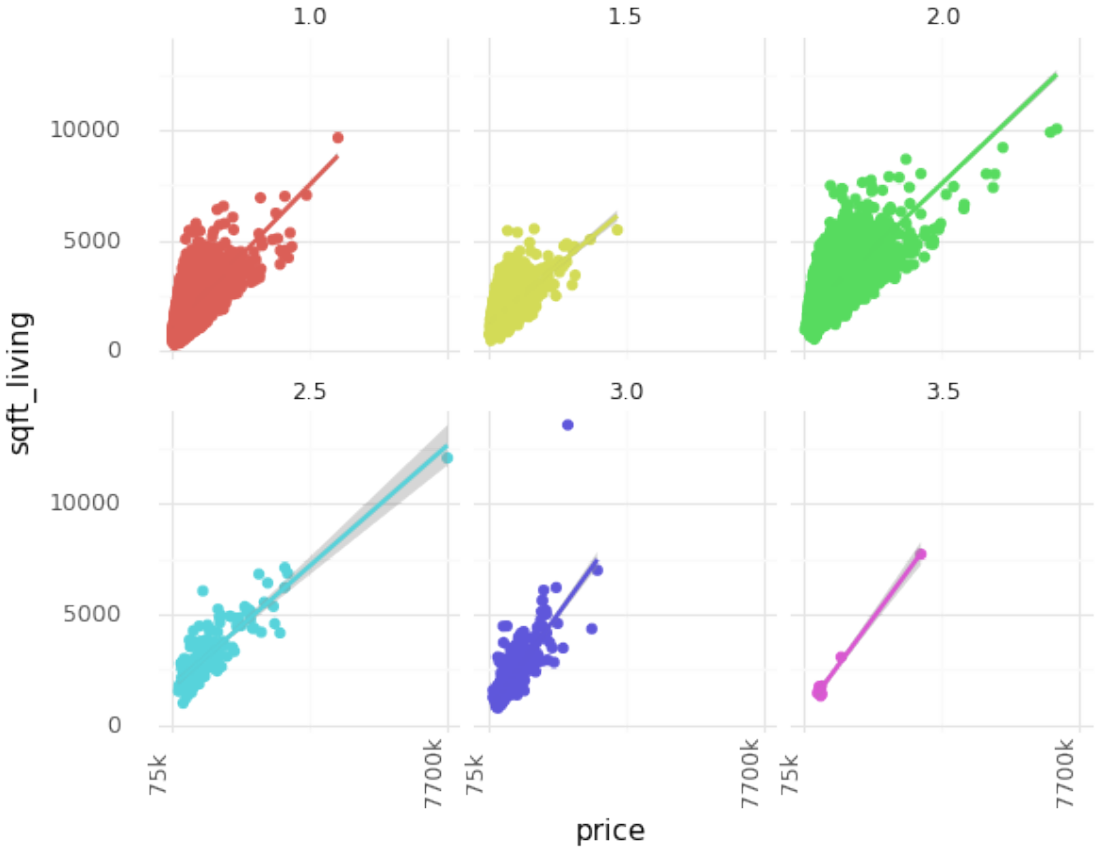
[14]:
```
(ggplot(kc, aes("price", "sqft_living")) + geom_point() +
 geom_smooth(method = "lm") + facet_wrap("~floors") +
theme_minimal())
```



[14]: <ggplot: (8764931425796)>

Got rid of extra ink to remove distractions.

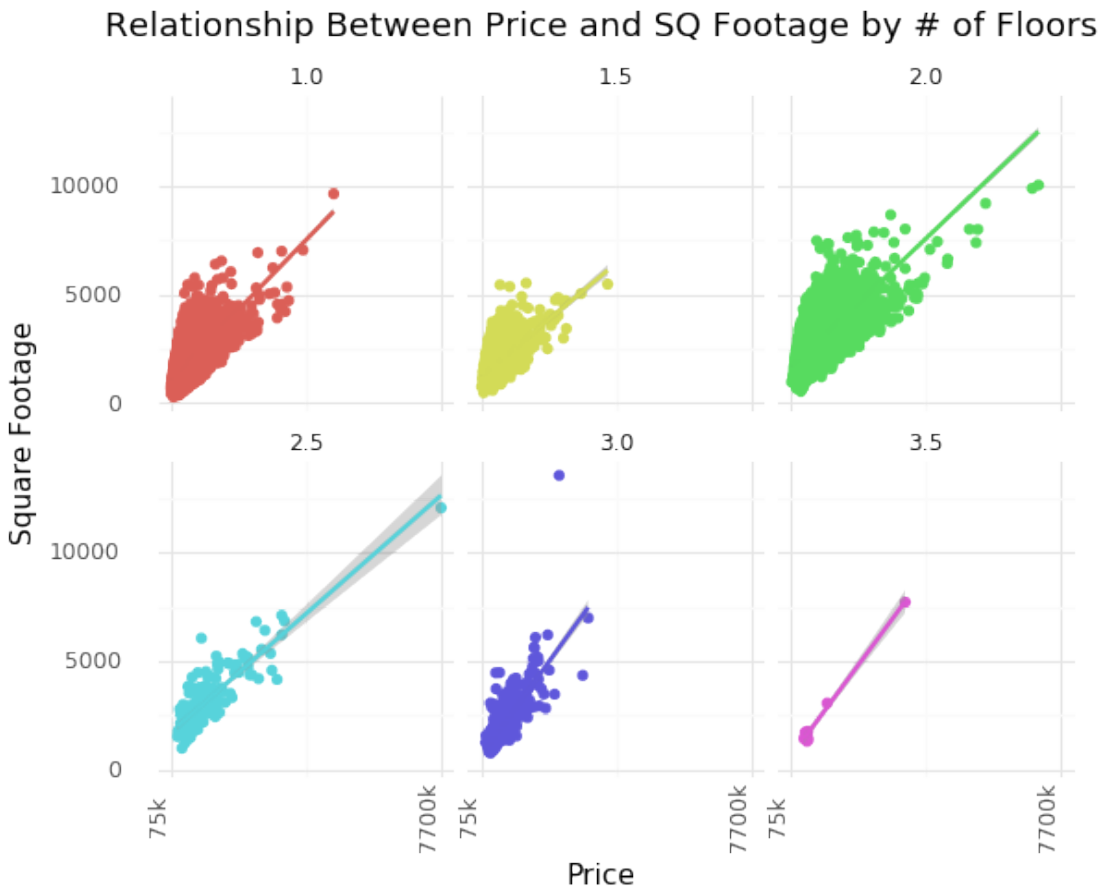
[21]:
```
(ggplot(kc, aes("price", "sqft_living", color = "factor(floors)")) +␣
 ↪geom_point() +
 geom_smooth(method = "lm") + facet_wrap("~floors") +
theme_minimal() +
theme(legend_position = "none",
     axis_text_x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
scale_x_continuous(breaks = [min(kc.price), max(kc.price)], labels = ["75k",␣
 ↪"7700k"]))
```



[21]: <ggplot: (8764932976050)>

Filled by color to help differentiate between floors. Simplified X axis text and rotated it so it stopped overlapping.

[22]:
```
(ggplot(kc, aes("price", "sqft_living", color = "factor(floors)")) +␣
 ↪geom_point() +
 geom_smooth(method = "lm") + facet_wrap("~floors") +
theme_minimal() +
theme(legend_position = "none",
     axis_text_x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
scale_x_continuous(breaks = [min(kc.price), max(kc.price)], labels = ["75k",␣
 ↪"7700k"]) +
labs(title = "Relationship Between Price and SQ Footage by # of Floors",
    x = "Price", y = "Square Footage"))
```



[22]: <ggplot: (8764924365493)>

Added Title and clear axis labels.

5. Come up with **one specific question** with your group that you want to *always* ask yourselves when making a graph to make sure it's accessible to a certain group. Draw on your own experiences (and your friends/family's experiences). For example, if you have a rare type of colorblindness, you might come up with the question "Is my palette readable by someone with tritanomaly colorblindness?". Think about visual impairments, cultural/group contex that not everyone shares, language barriers…etc. We'll share these with the class.

Answers discussed in class. Answers may vary.