untitled4-2

June 8, 2023

1 DATA VISUALIZATION

STEP-1 IMPORT LABORARY

[5]:
```
import seaborn as sns
import matplotlib.pyplot as plt
```

STEP-2 LOAD DATASET

[6]:
```
diamonds = sns.load_dataset("diamonds")
```

[7]:
```
diamonds.head()
```

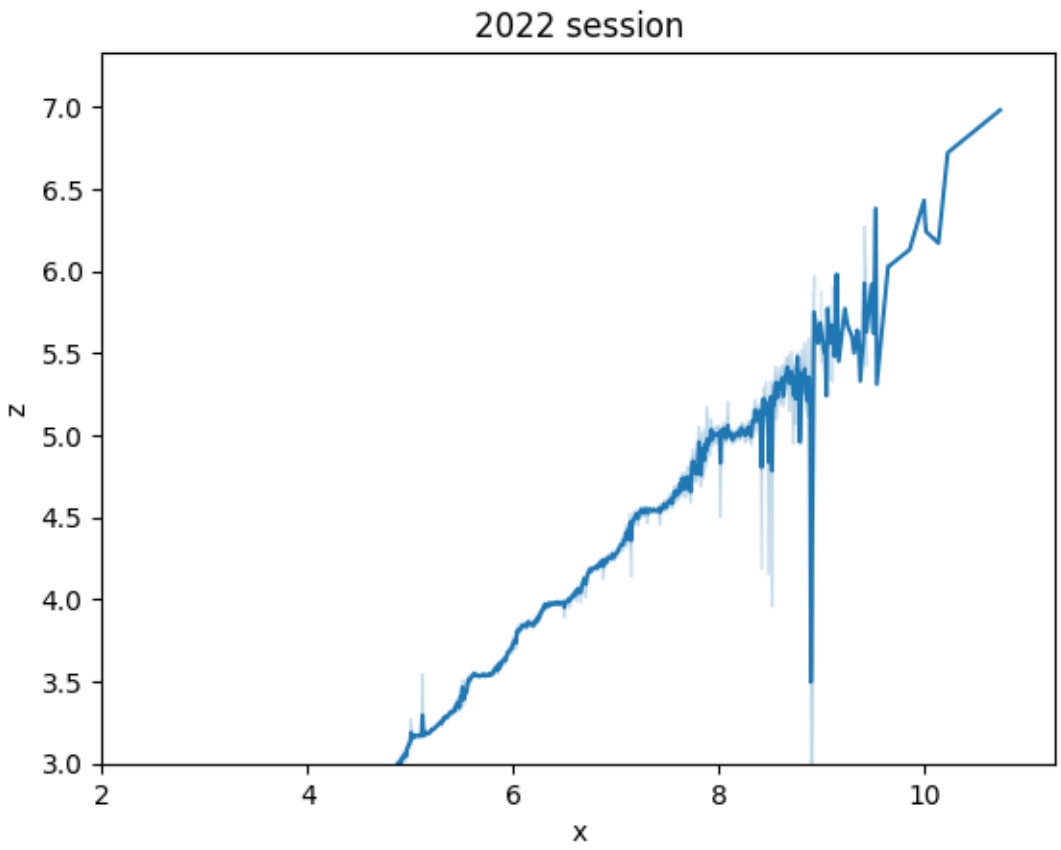
[7]:

	carat	cut	color	clarity	depth	table	price	x	y	z
0	0.23	Ideal	E	SI2	61.5	55.0	326	3.95	3.98	2.43
1	0.21	Premium	E	SI1	59.8	61.0	326	3.89	3.84	2.31
2	0.23	Good	E	VS1	56.9	65.0	327	4.05	4.07	2.31
3	0.29	Premium	I	VS2	62.4	58.0	334	4.20	4.23	2.63
4	0.31	Good	J	SI2	63.3	58.0	335	4.34	4.35	2.75

3STEP-3 PLOT GRAPH

[9]:
```
sns.lineplot(x="x",y="z",data=diamonds)
plt.xlim(2)
plt.ylim(3)
plt.title("2022 session")
```

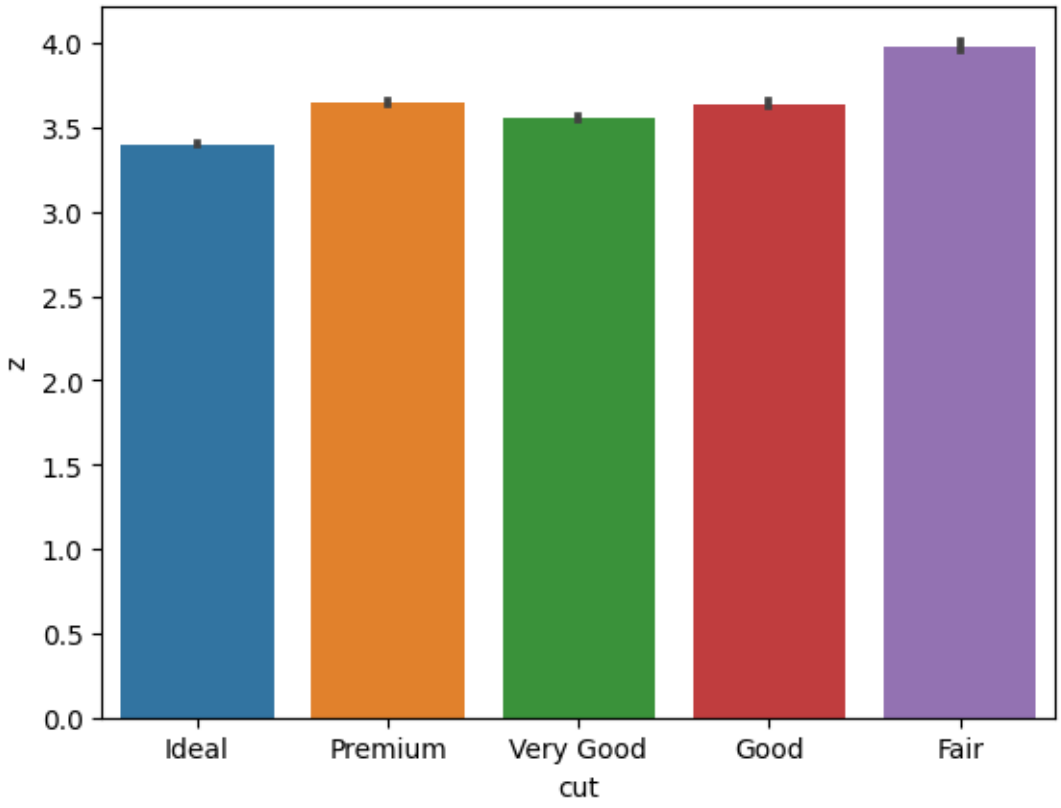
[9]:
```
Text(0.5, 1.0, '2022 session')
```



###BAR PLOT

[11]:
```
sns.barplot(x="cut",y="z",data=diamonds)
```

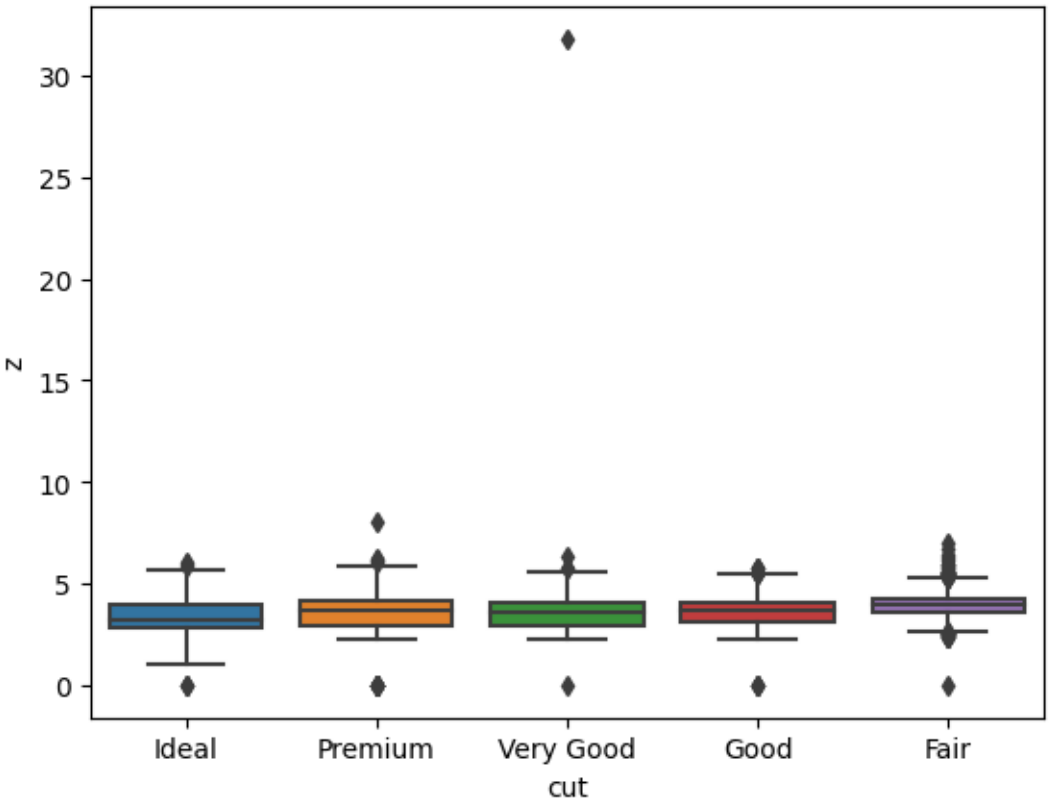
[11]:
```
<Axes: xlabel='cut', ylabel='z'>
```



BOXPLOT

[12]:
```
sns.boxplot(x="cut",y="z",data=diamonds)
```

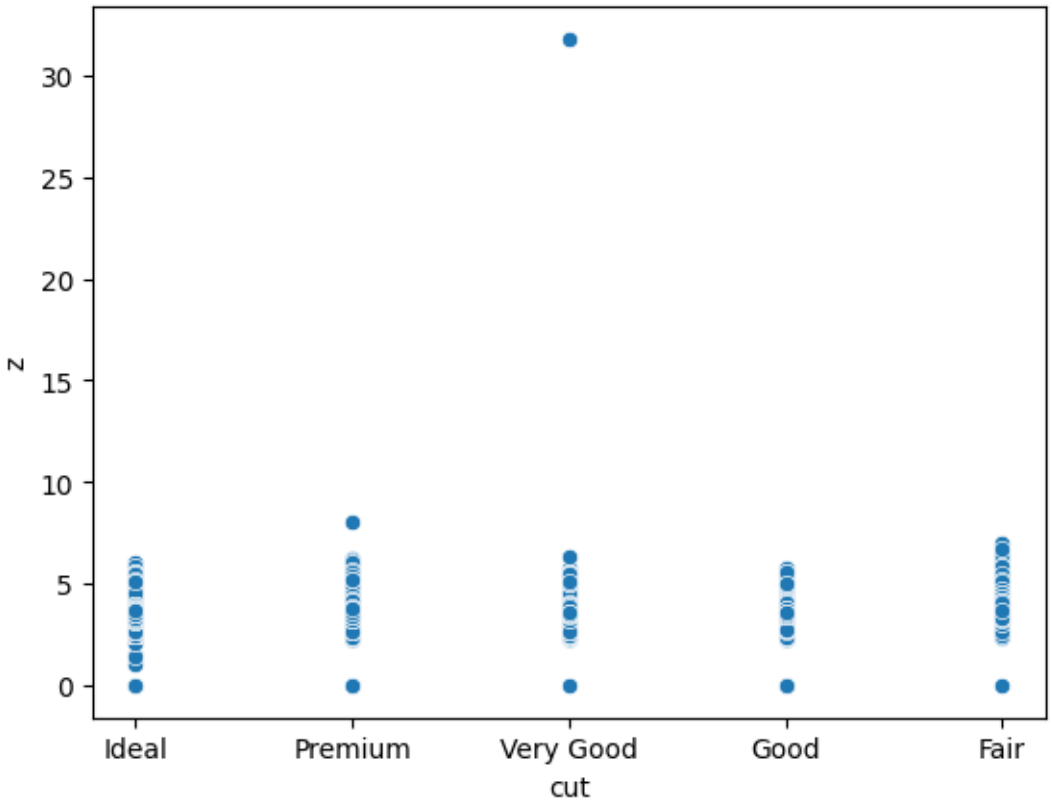
[12]:
```
<Axes: xlabel='cut', ylabel='z'>
```



###SCATTER PLOT

[13]:
```
sns.scatterplot(x="cut",y="z",data=diamonds)
```

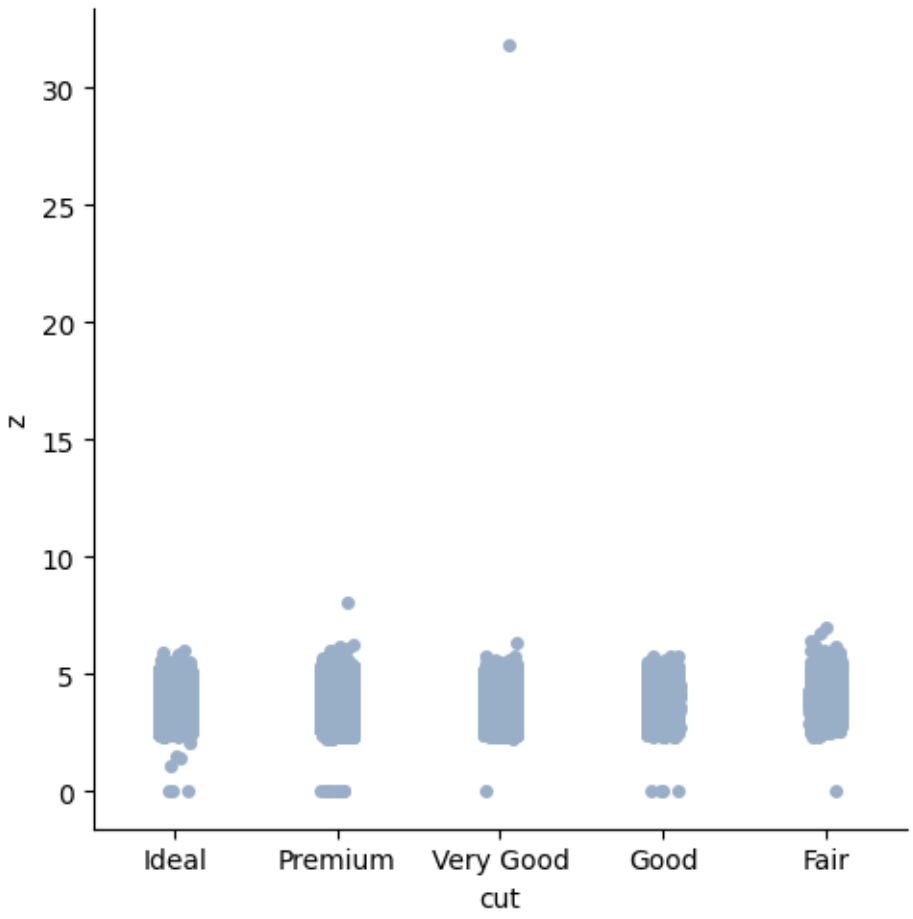
[13]:
```
<Axes: xlabel='cut', ylabel='z'>
```



CATPLOT

[14]:
```
sns.catplot(x="cut",y="z",data=diamonds,color="#98AFC7")
```

[14]:
```
<seaborn.axisgrid.FacetGrid at 0x7f6d7df67b20>
```

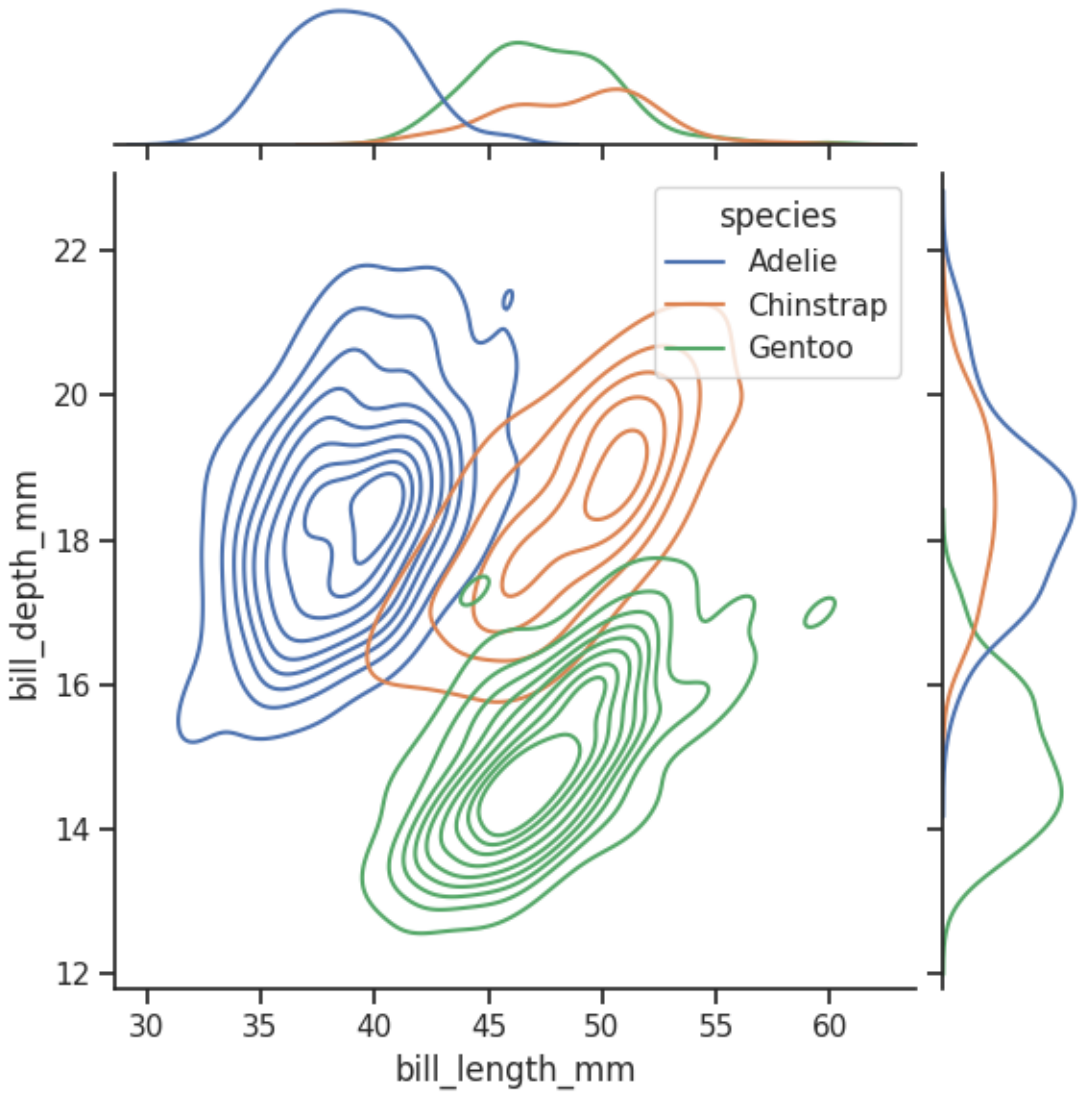


SEABORN GALLERY

[ ]:
```
import seaborn as sns
sns.set_theme(style="ticks")

# Load the penguins dataset
penguins = sns.load_dataset("penguins")

# Show the joint distribution using kernel density estimation
g = sns.jointplot(
    data=penguins,
    x="bill_length_mm", y="bill_depth_mm", hue="species",
    kind="kde",
)
```

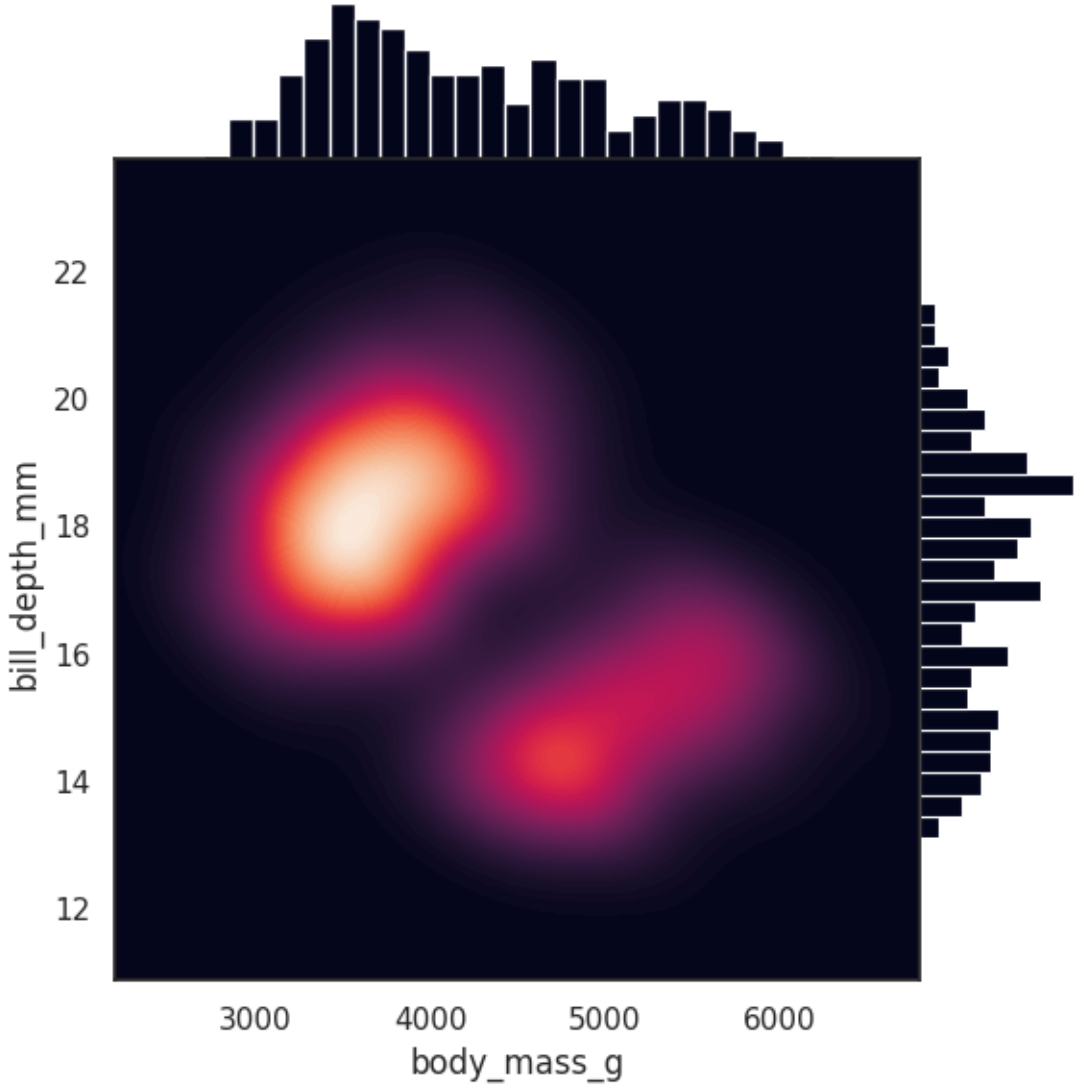


[ ]:
```
import seaborn as sns
sns.set_theme(style="white")

df = sns.load_dataset("penguins")

g = sns.JointGrid(data=df, x="body_mass_g", y="bill_depth_mm", space=0)
g.plot_joint(sns.kdeplot,
             fill=True, clip=((2200, 6800), (10, 25)),
             thresh=0, levels=100, cmap="rocket")
g.plot_marginals(sns.histplot, color="#03051A", alpha=1, bins=25)
```

[ ]:
```
<seaborn.axisgrid.JointGrid at 0x7f856f1cc7c0>
```



[ ]:
```
import plotly.express as px
fig = px.scatter(iris,␣
 ↪x="sepal_width",y="sepal_length",color="species",size="petal_length",␣
 ↪hover_data=["petal_width"])
fig.show()
```

[ ]:
```
import plotly.express as px
gapminder = px.data.gapminder()
fig = px.scatter(gapminder, x="gdpPercap", y="lifeExp", animation_frame="year",␣
 ↪animation_group="country",
                 size="pop", color="continent", hover_name="country")
fig.show()
```