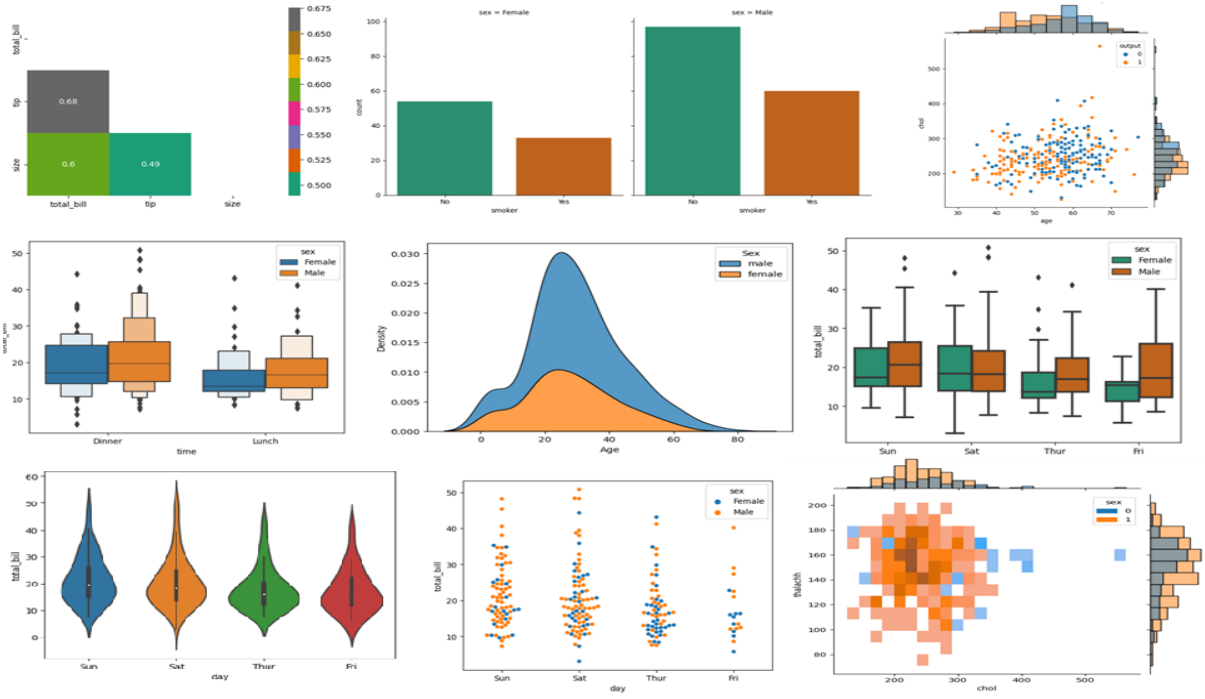
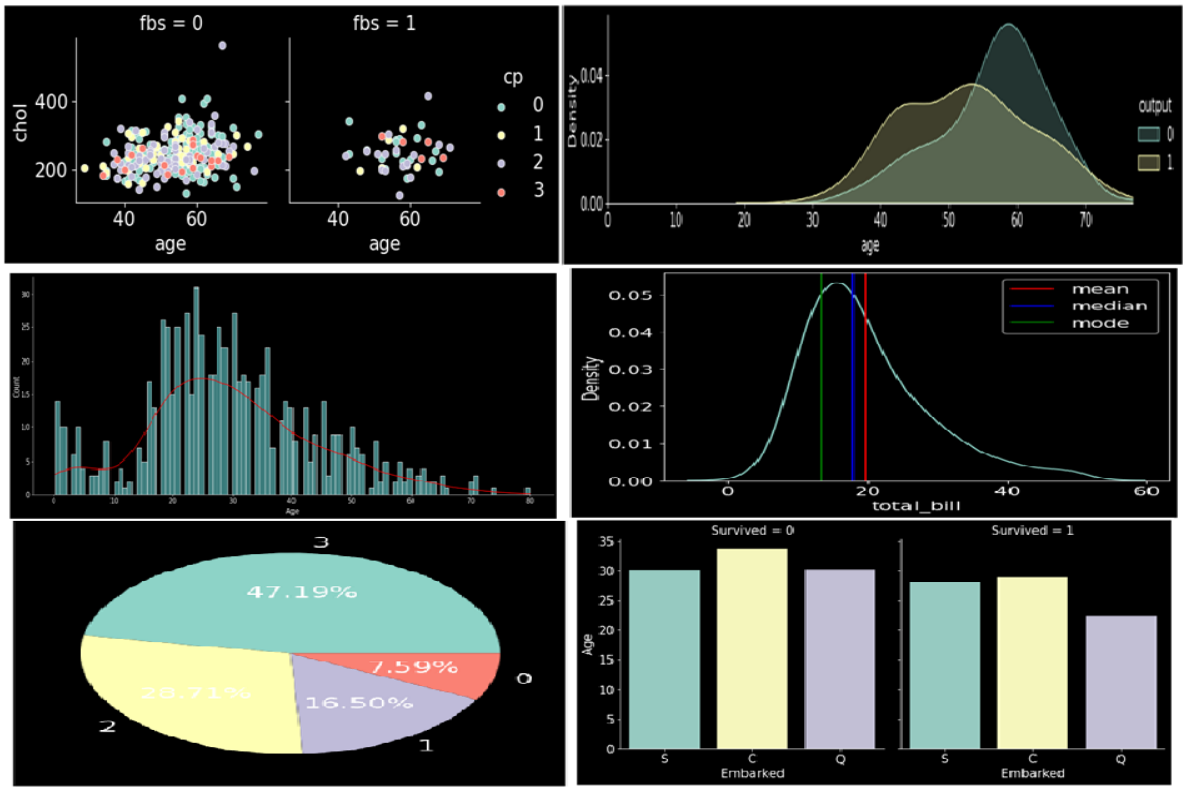
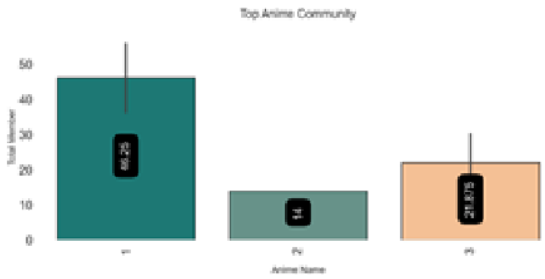
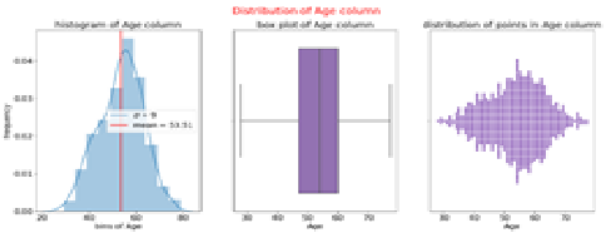
Machine Learning Visualization from Basic to Advance

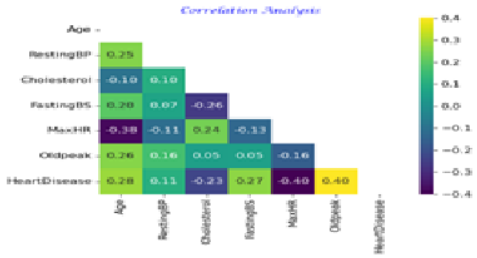


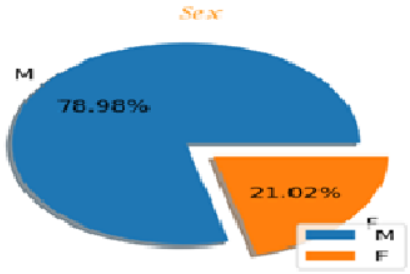


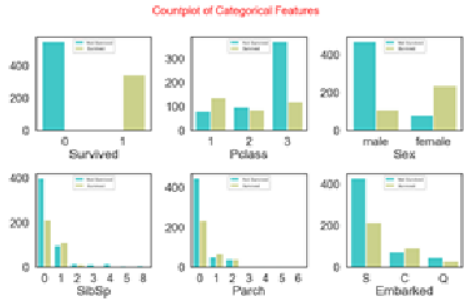
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

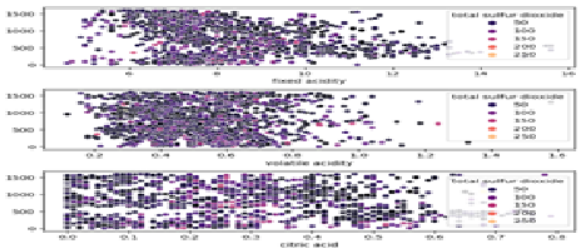


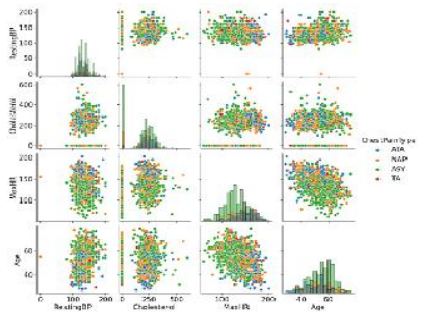


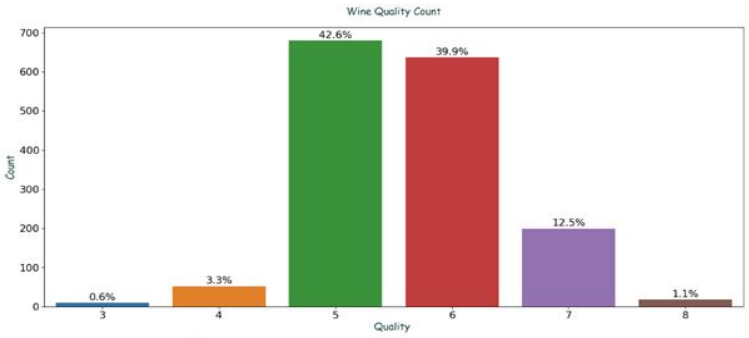


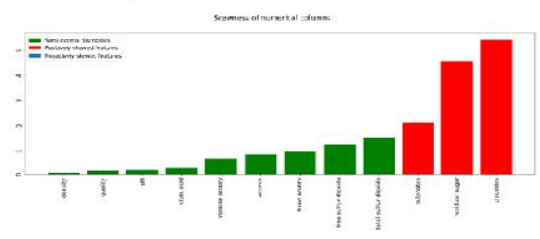


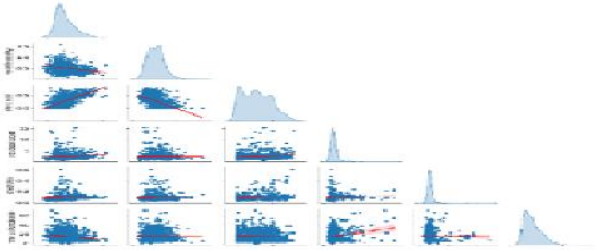


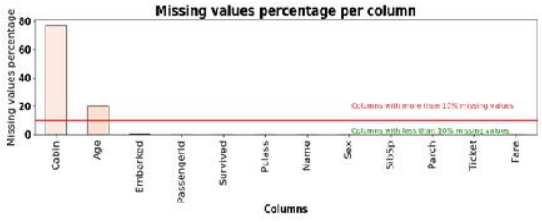




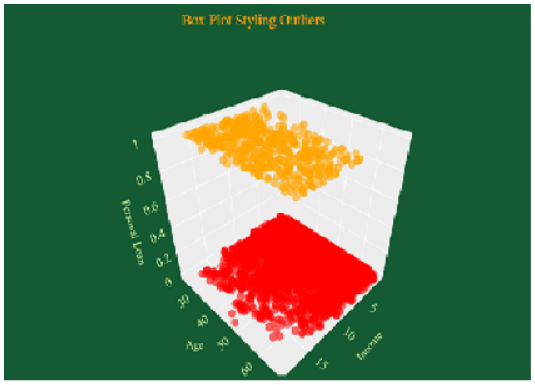


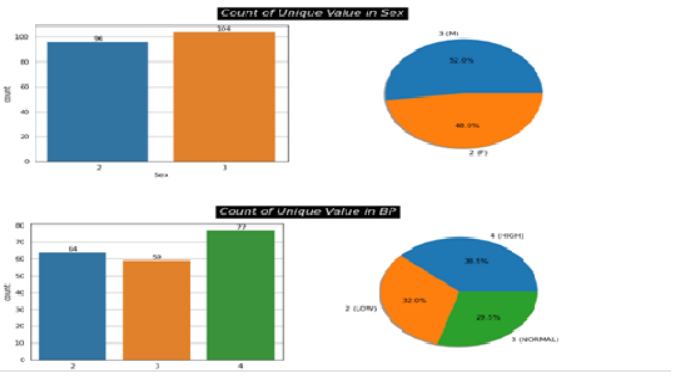


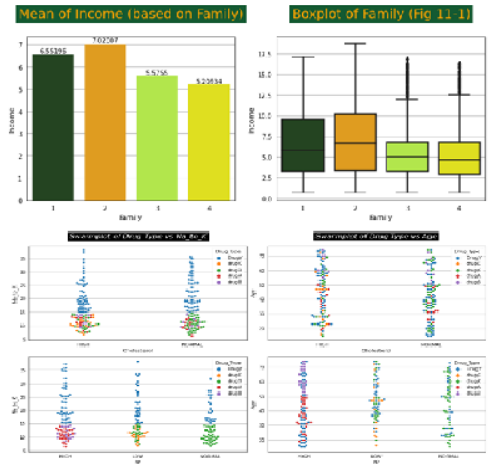


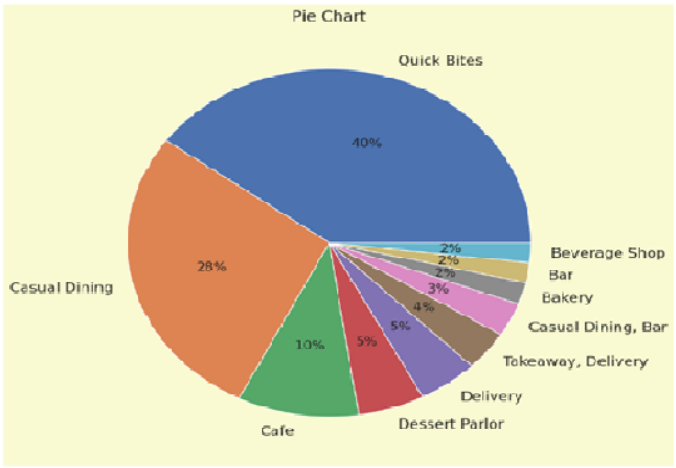


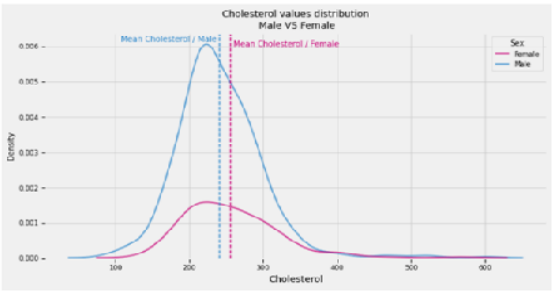
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

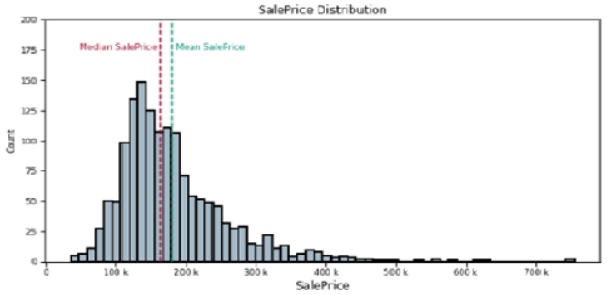


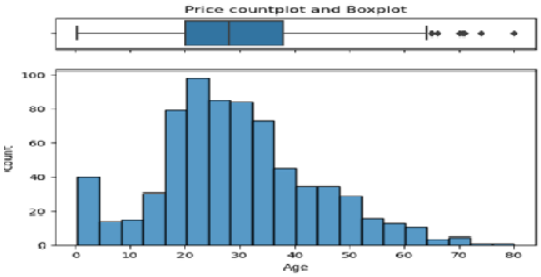


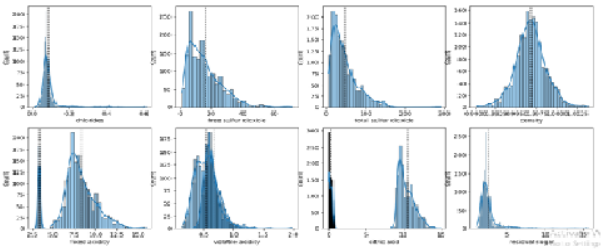


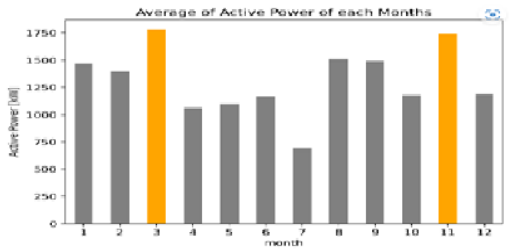


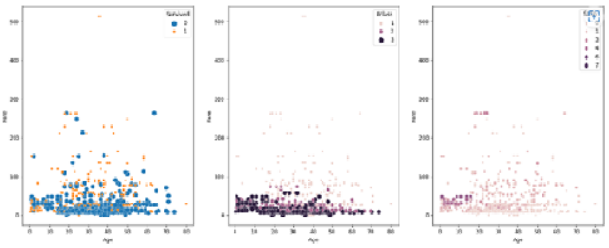




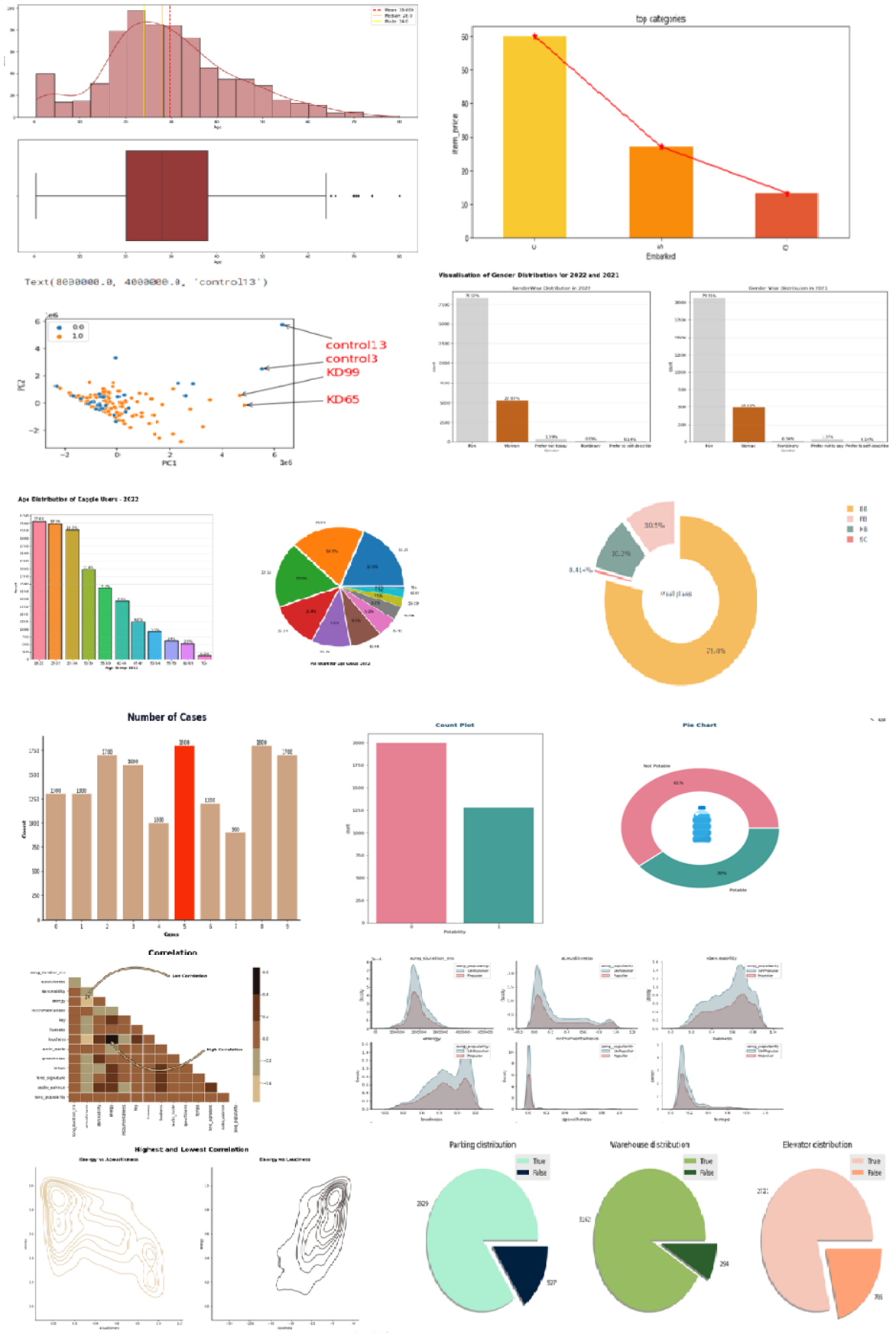




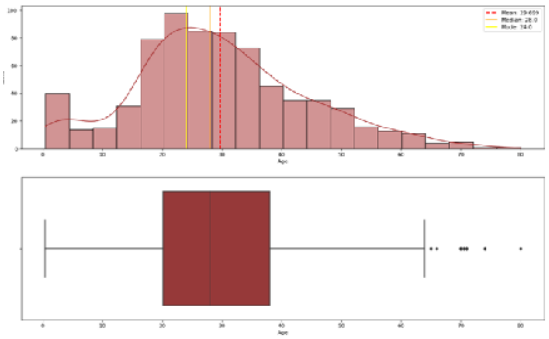


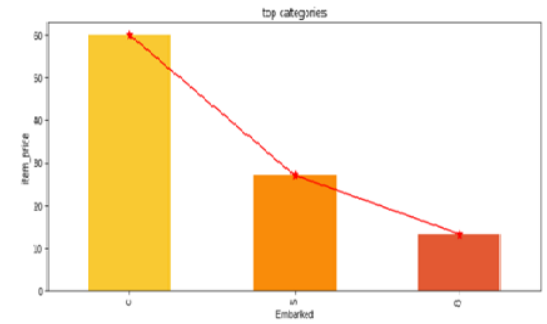


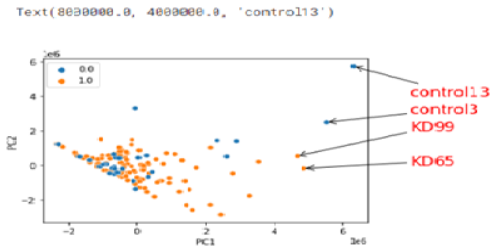
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

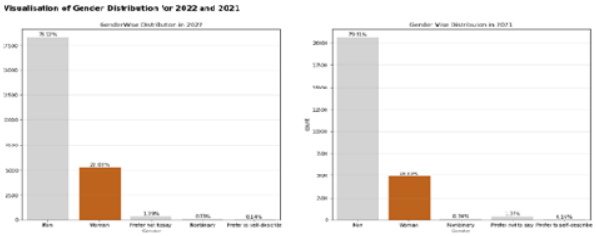


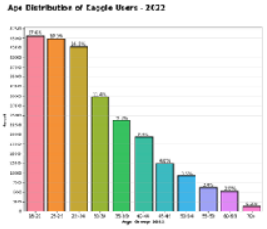
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

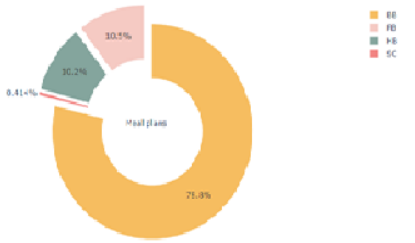


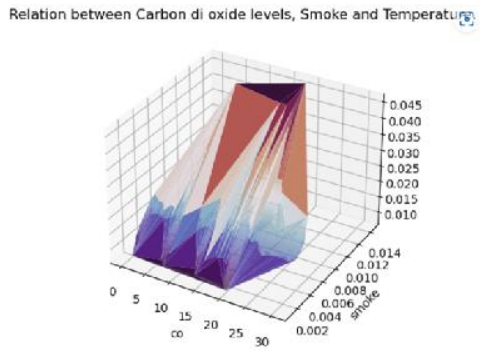


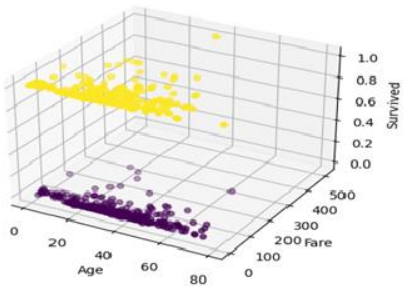


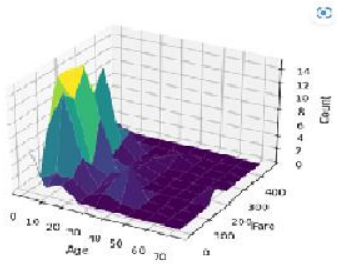


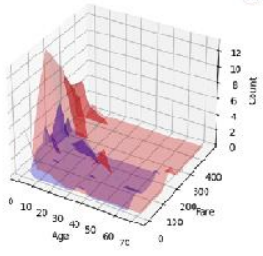


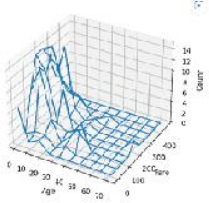


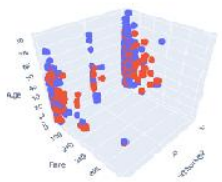








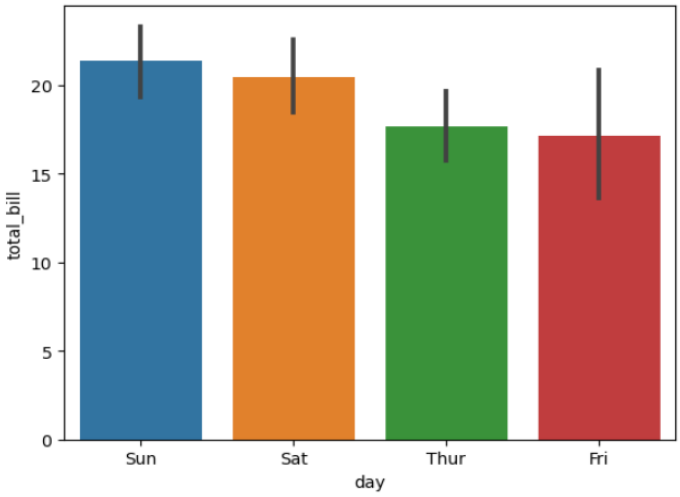




Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

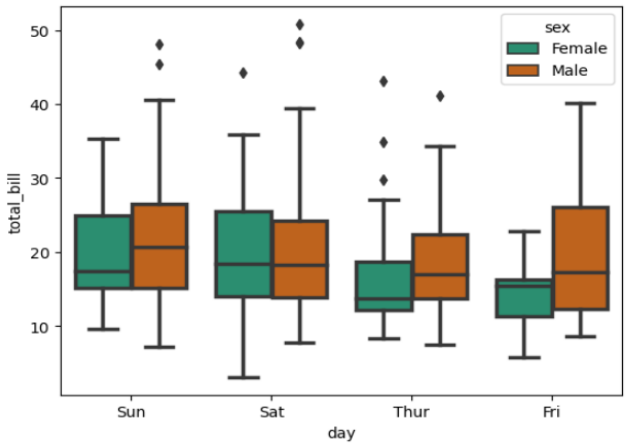
Barplot:

sns.barplot(x='day', y='total_bill', data=tips, palette='tab10');



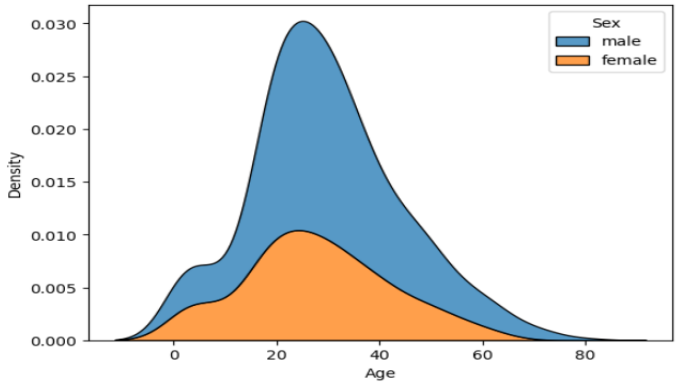
Boxplot

sns.boxplot(x='day', y='total_bill', hue='sex', data=tips, linewidth =2.5, palette='Dark2');



Kdeplot

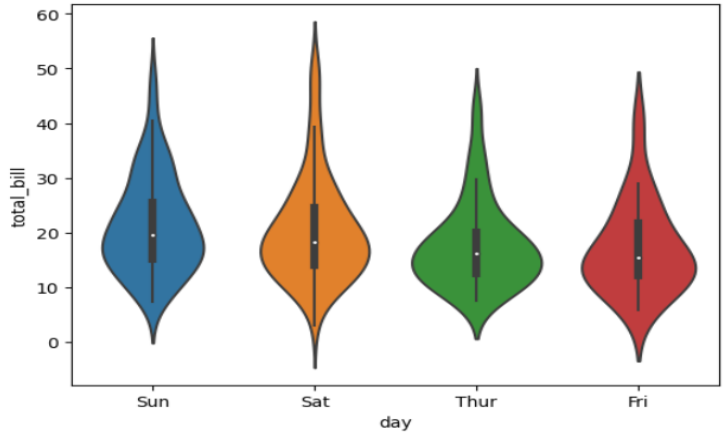
sns.kdeplot(data=df , x='Age', hue='Sex', multiple='stack', palette='tab10');



Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

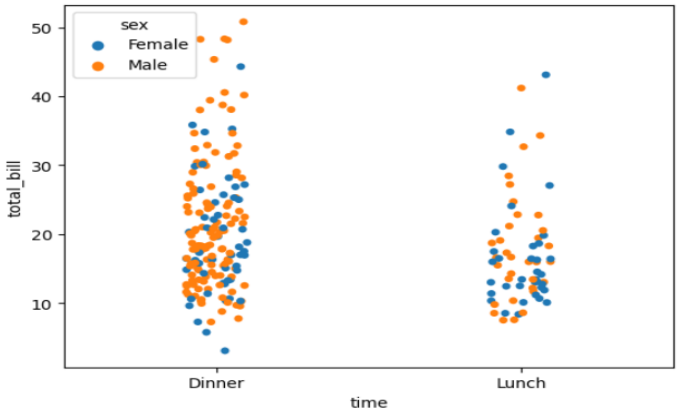
Violinplot

sns.violinplot(x="day", y="total_bill", data=tips);



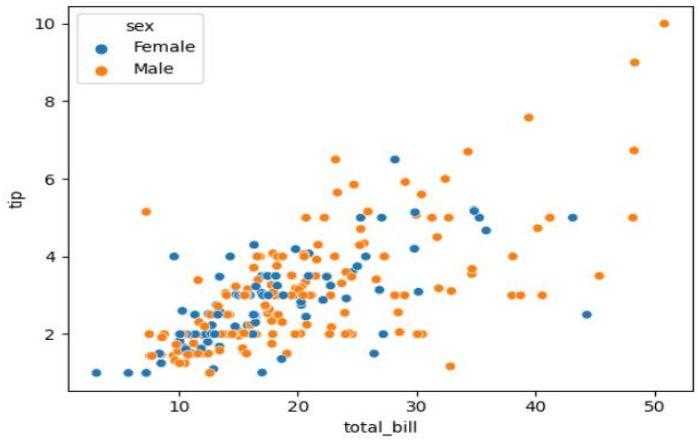
Stripplot

sns.stripplot(x="time", y="total_bill", hue="sex", data=tips);



Scatterplot

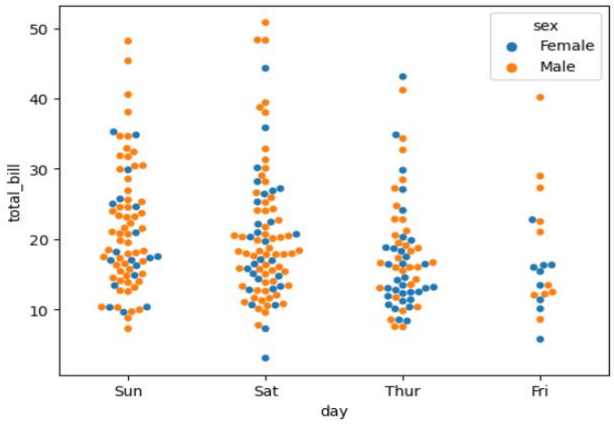
sns.scatterplot(x = 'total_bill', y = 'tip', hue = 'sex', data = tips);



Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

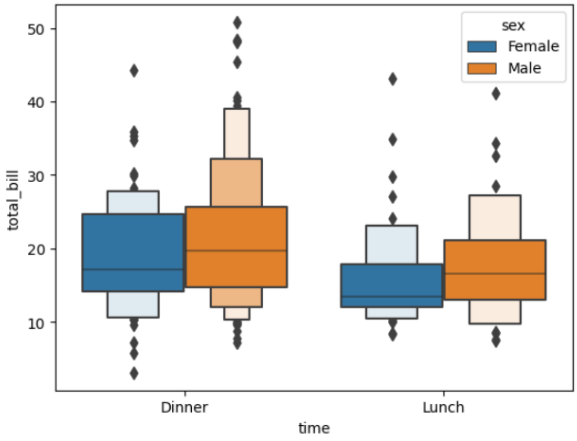
Swarmplot

```
sns.swarmplot(x="day", y="total_bill", hue="sex", data=tips);
```



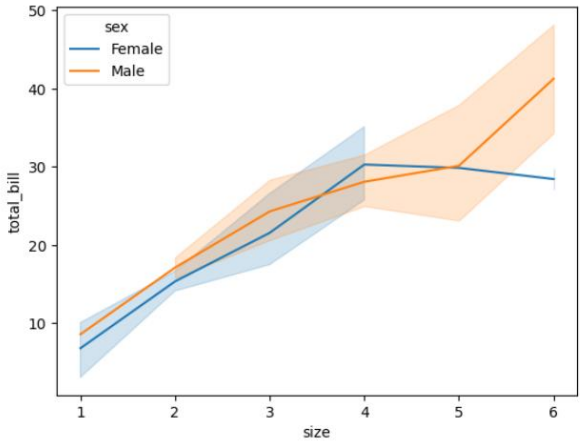
Boxenplot

```
sns.boxenplot( x='time', y="total_bill", hue='sex', data=tips);
```



Lineplot

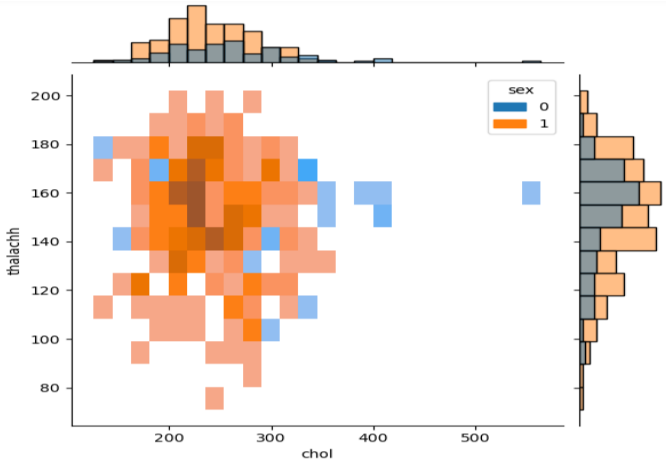
```
sns.lineplot(x="size",y="total_bill",data=tips,hue='sex',markers=True);
```



Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

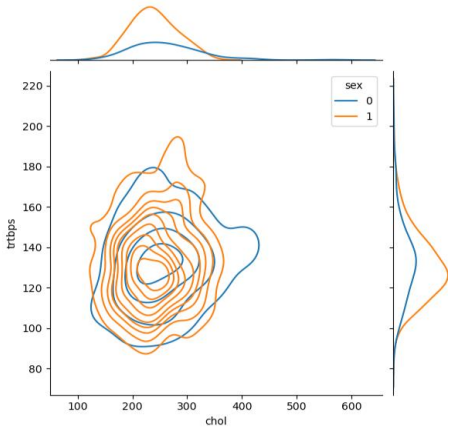
Jointplot

```
sns.jointplot(x="chol", y="thalachh",data=heart,kind="hist",hue='sex');
```



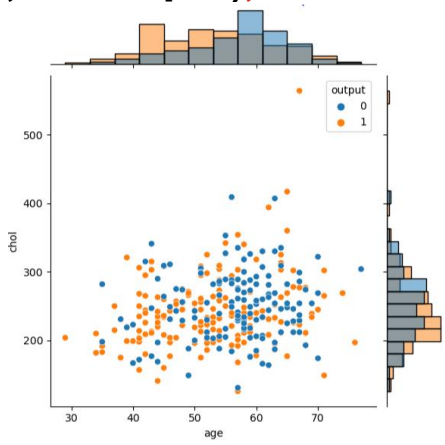
Jointplot

```
sns.jointplot(x="chol",y="trtbps",data=heart,kind="kde",hue='sex');
```



JointGrid

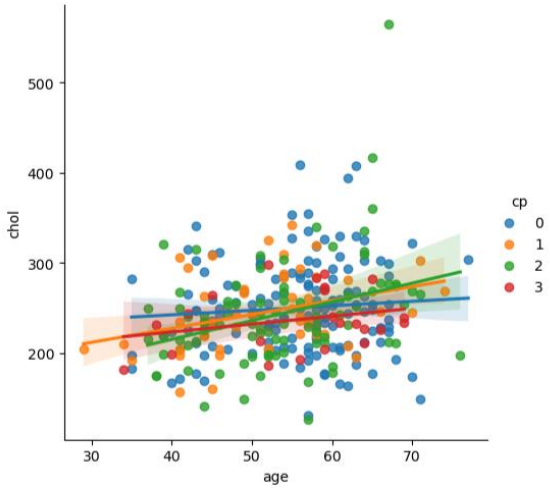
```
g = sns.JointGrid(data=heart, x="age", y="chol", hue="output")
g.plot(sns.scatterplot, sns.histplot);
```



Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

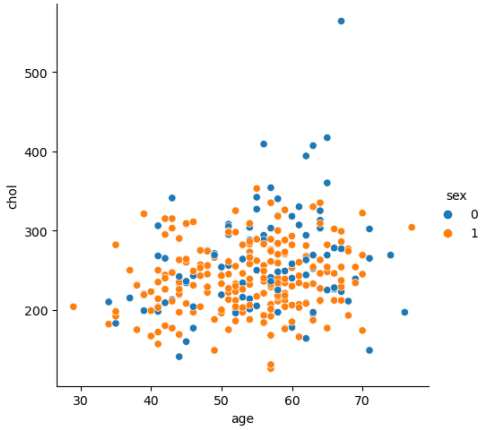
Lmplot

```
g= sns.lmplot(x="age", y="chol", hue="cp", data=heart)
```



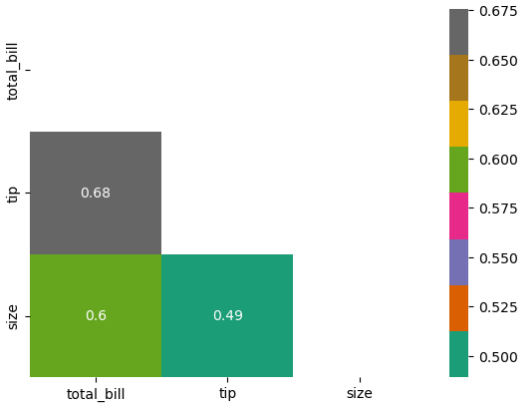
Relplot

```
g = sns.relplot(x="age", y="chol", data=heart,hue='sex')
```



Heatmap

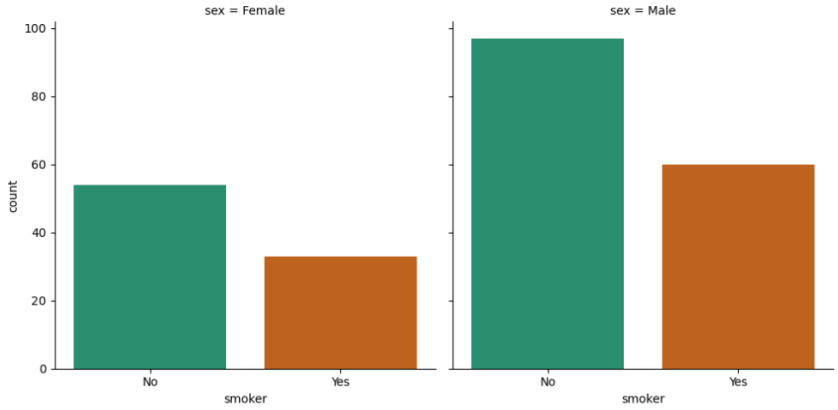
```
mask = np.triu(np.ones_like(tips.corr(), dtype=bool))
sns.heatmap(tips.corr(), mask = mask, annot=True, cmap='Dark2');
```



Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

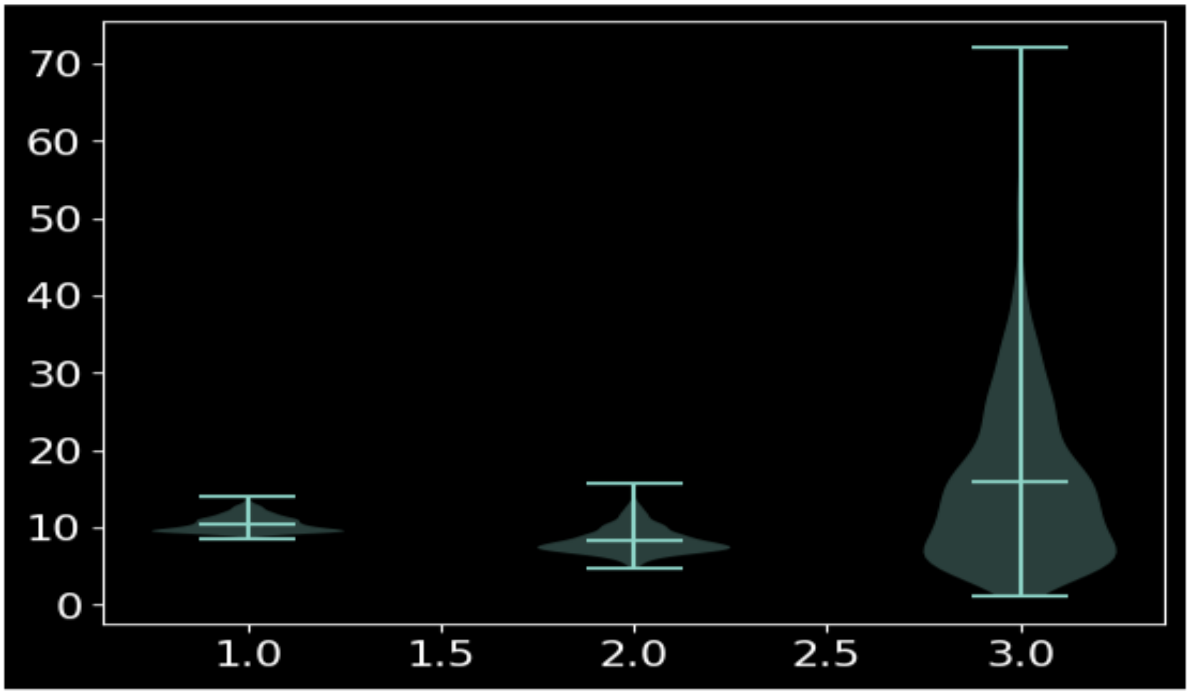
Catplot

```
sns.catplot(x='smoker', col='sex', kind='count', data=tips
,palette="Dark2");
```



Violinplot

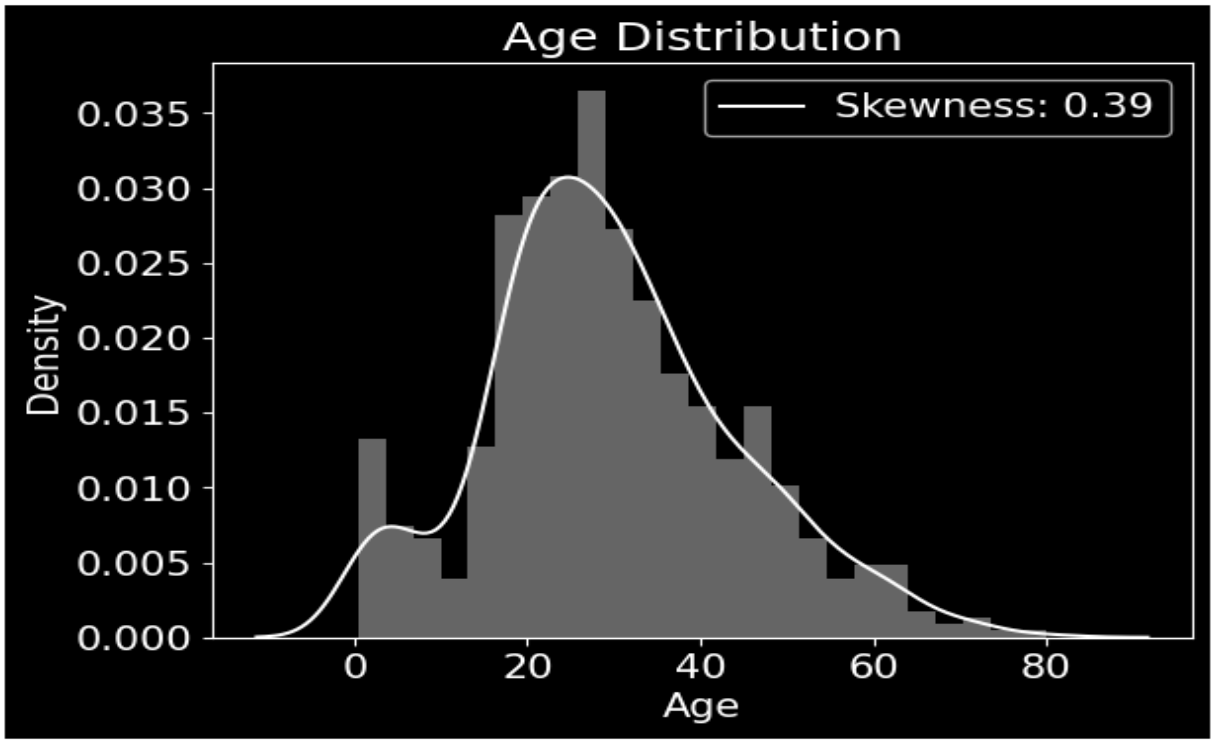
```
plt.violinplot([wine["alcohol"], wine['fixed acidity'],wine['free sulfur
dioxide']], positions=[1,2,3], showmeans=True);
```



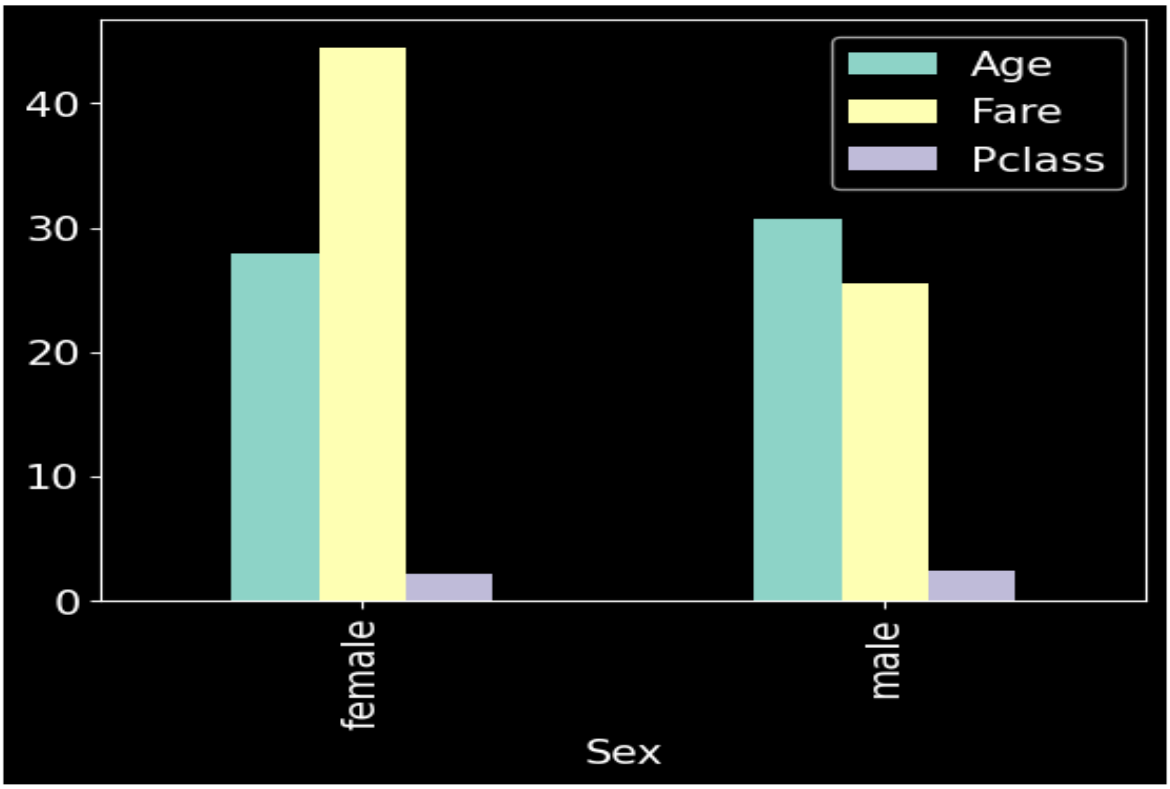
Distplot

```
bar = sns.distplot(titanic["Age"],color='Blue',kde=True,bins=25)
bar.legend(["Skewness: {:.2f}".format(titanic['Age'].skew())])
plt.title("Age Distribution");
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

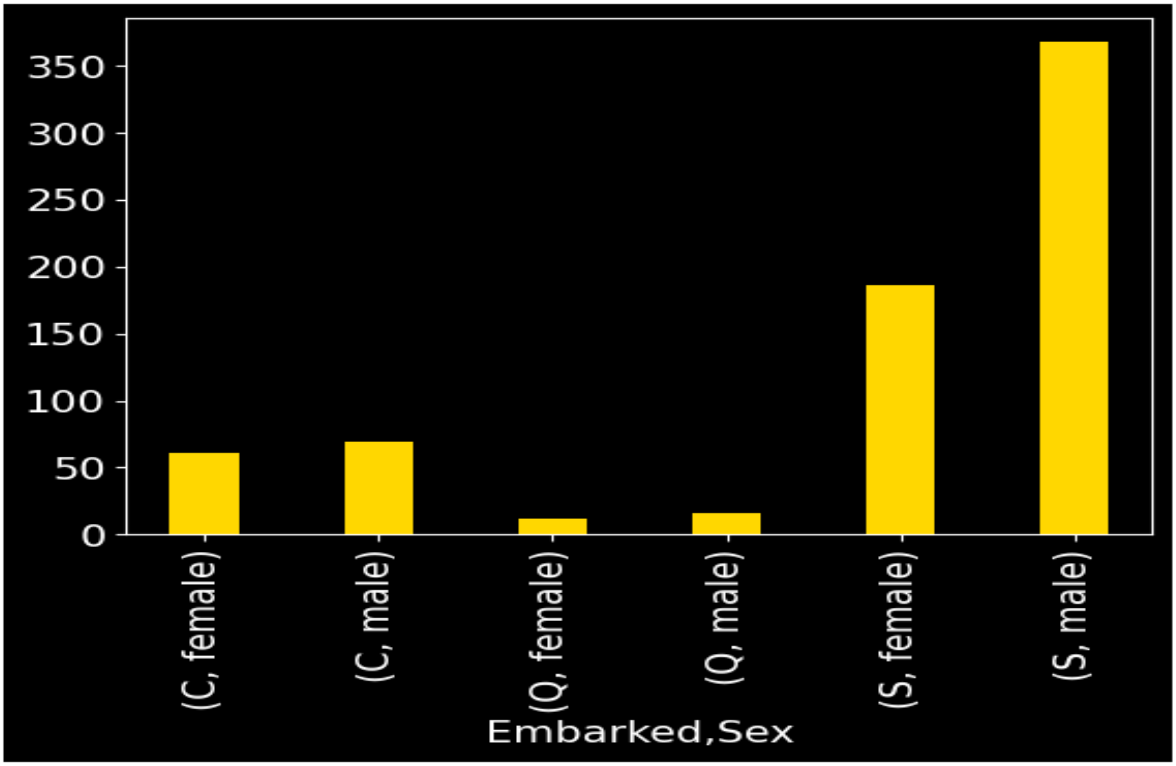



```
titanic.groupby("Sex")["Age","Fare","Pclass"].mean().plot(kind='bar')
```

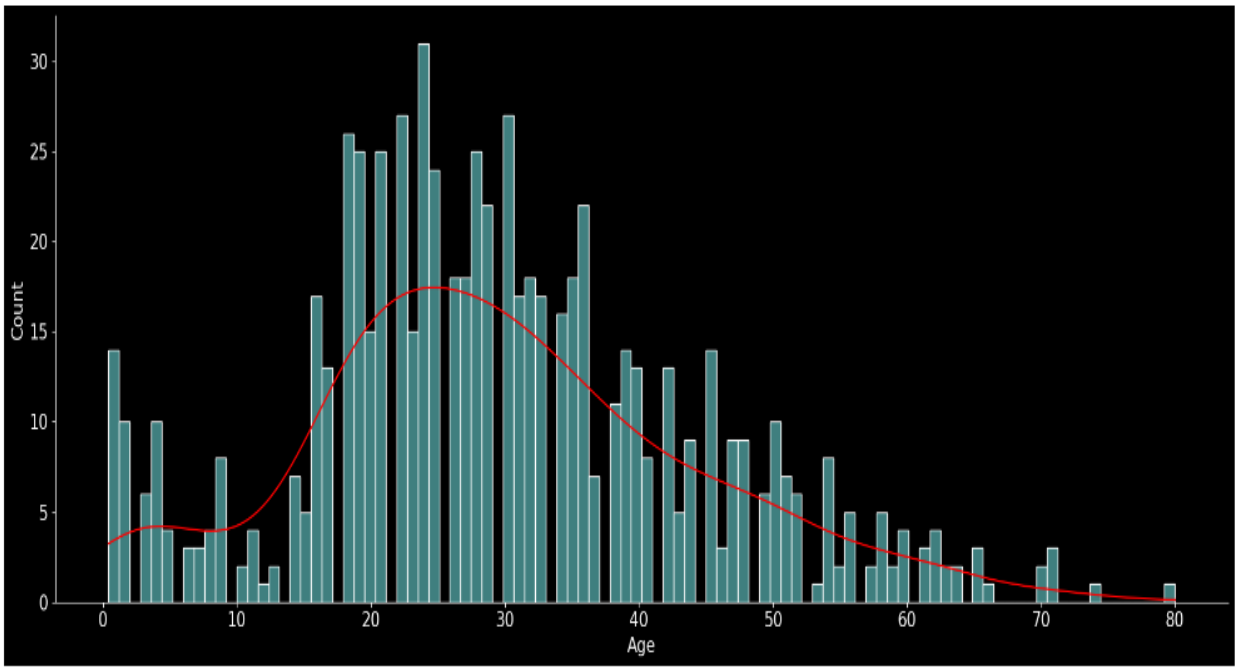



```
color = plt.cm.copper(np.linspace(0, 1, 10))
titanic.groupby(['Embarked','Sex'])['Age'].count().plot(kind='bar',
width=.4,color='gold');
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

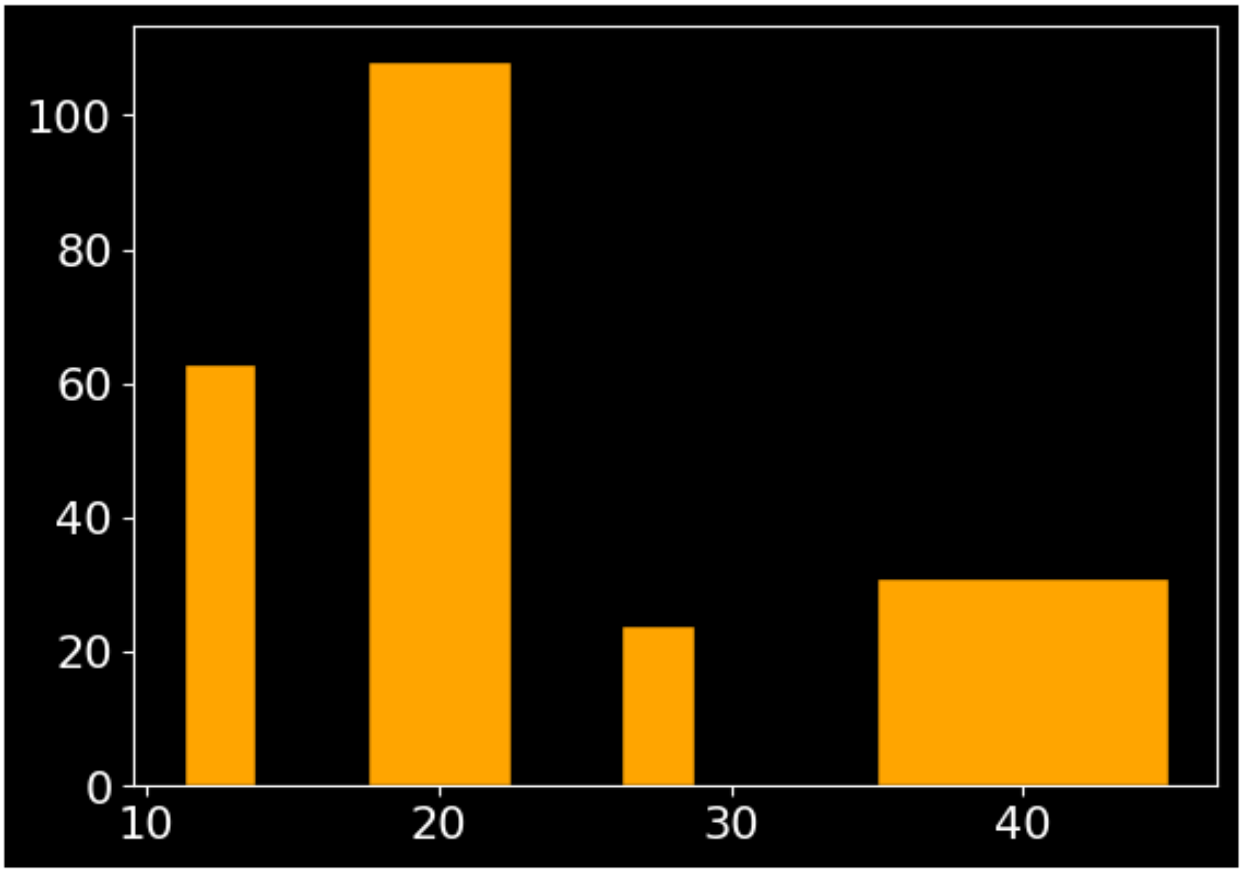



```
sns.displot(data=titanic, x="Age", kde=True, bins = 100,color = "red", facecolor = "#3F7F7F",height = 5, aspect = 3.5);
```

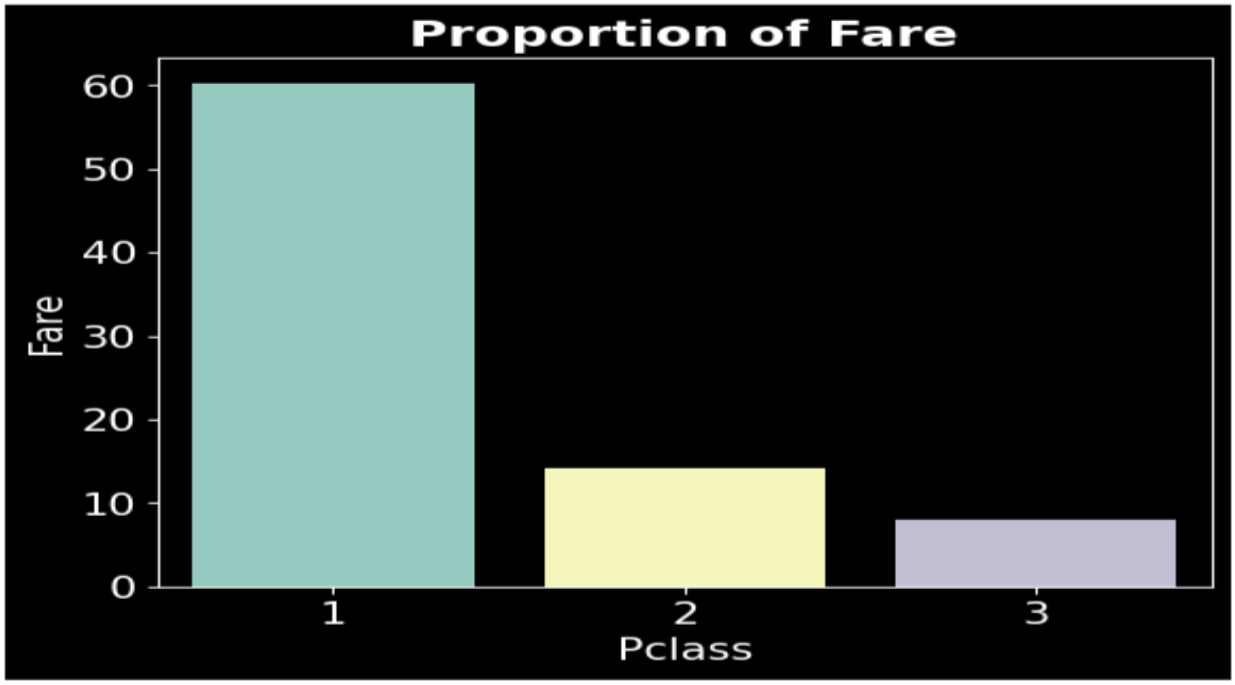



```
plt.hist(tips['total_bill'],color='orange',bins=[10,15,25,30,50],edgecolor='black',rwidth=0.5);
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

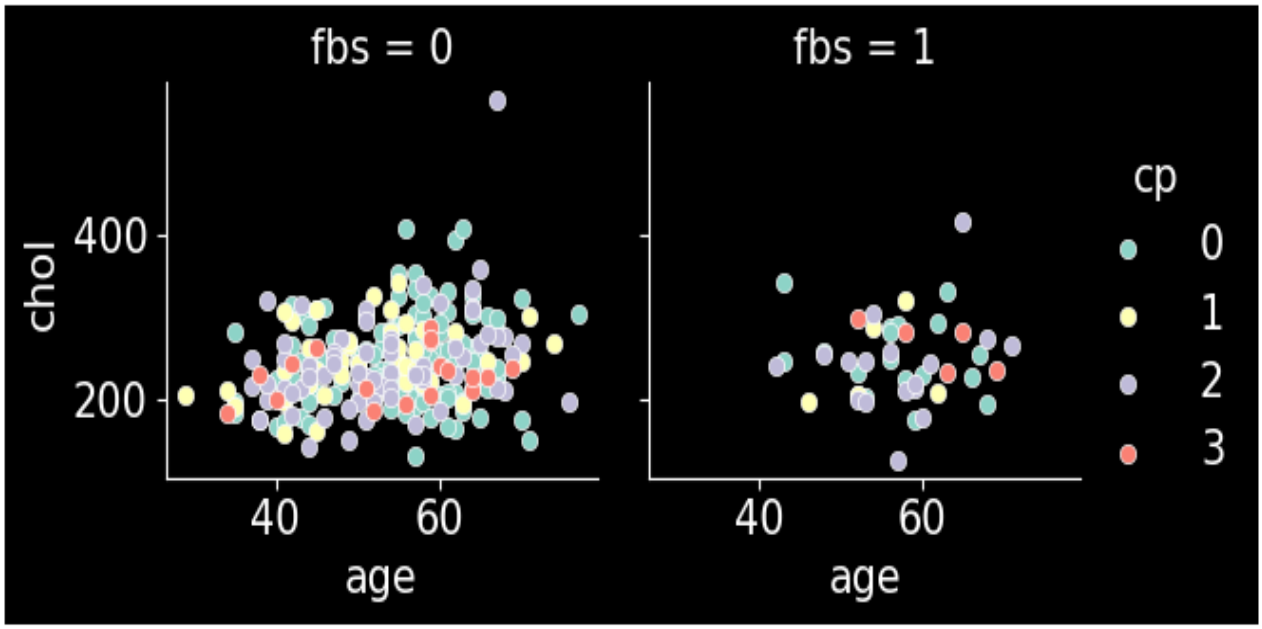



```
ht = pd.pivot_table(data=titanic, index="Pclass", aggfunc="median")
sns.barplot(x=ht.index,y=ht['Fare'])
plt.title("Proportion of Fare", fontweight="bold");
```

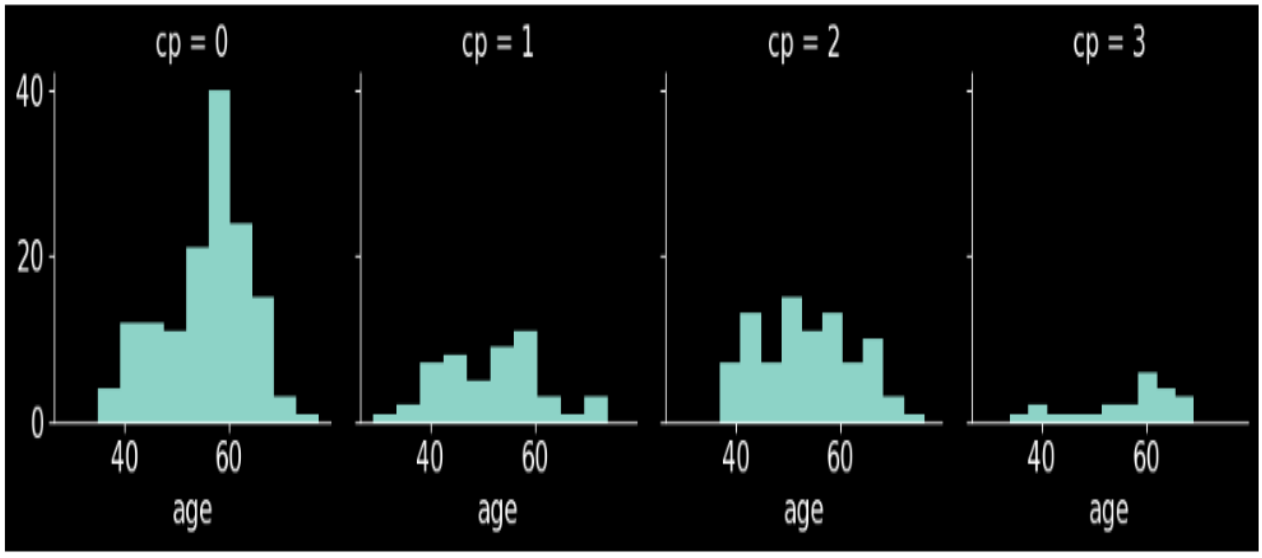


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
g = sns.FacetGrid(heart, col="fbs", hue="cp")
g.map_dataframe(sns.scatterplot, x="age", y="chol")
g.add_legend();
```

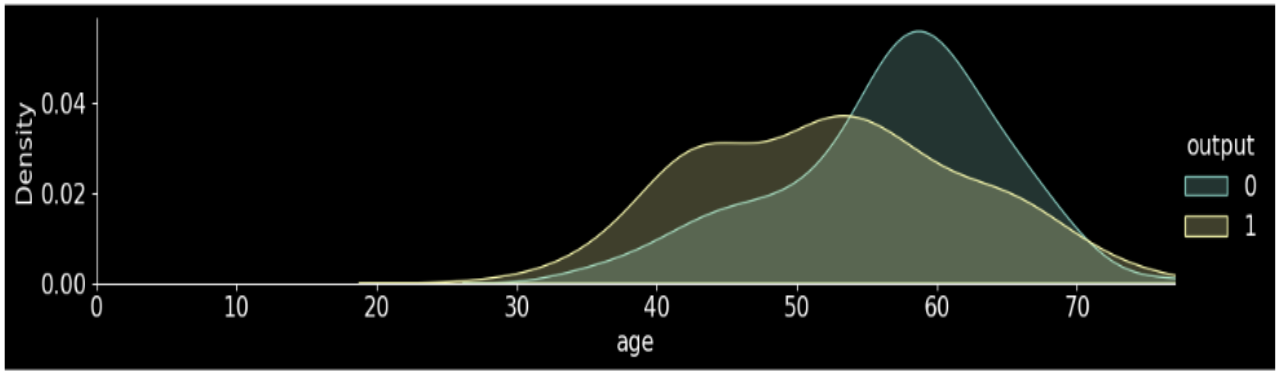



```
g = sns.FacetGrid(heart, col="cp")
g = g.map(plt.hist, "age");
```

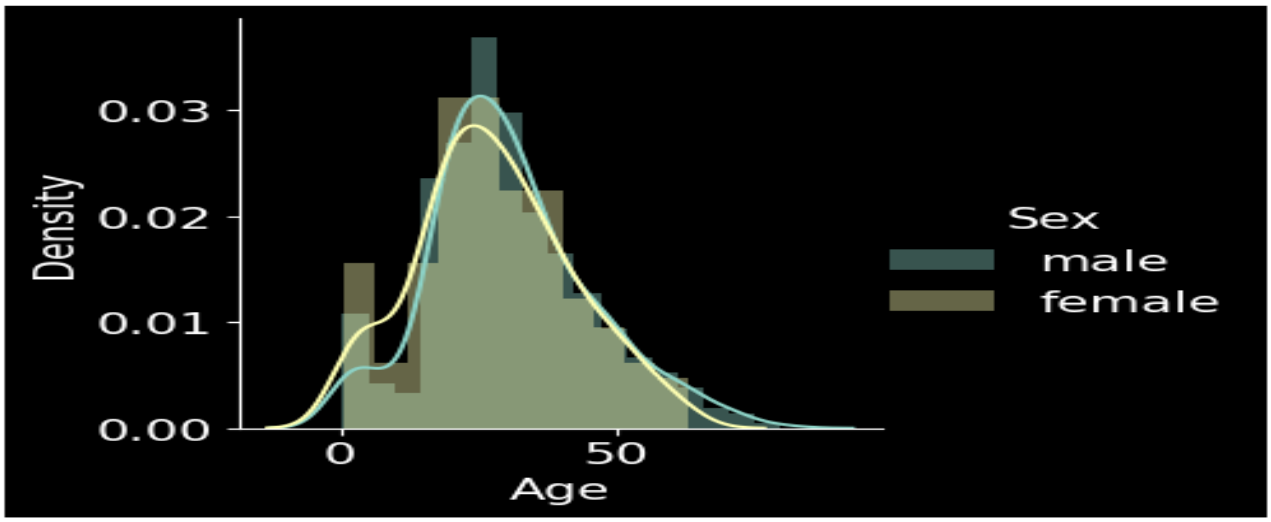



```
fig = sns.FacetGrid(heart, hue="output", aspect=4)
fig.map(sns.kdeplot, 'age', shade=True)
oldest = heart['age'].max()
fig.set(xlim=(0, oldest))
fig.add_legend()
plt.show()
```

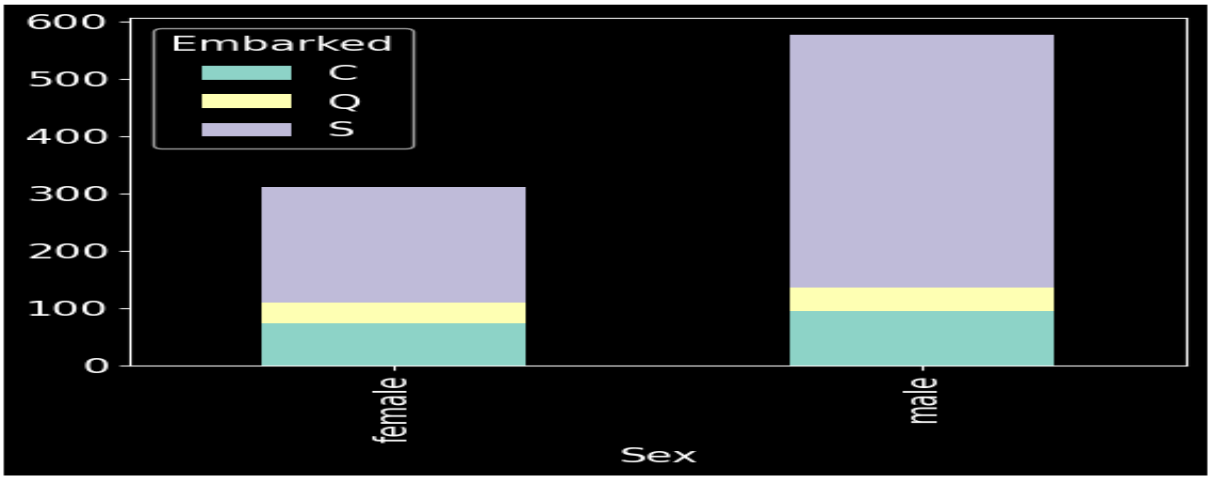
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/




```
sns.FacetGrid(titanic, hue='Sex', height=4).map(sns.distplot,
'Age').add_legend();
```

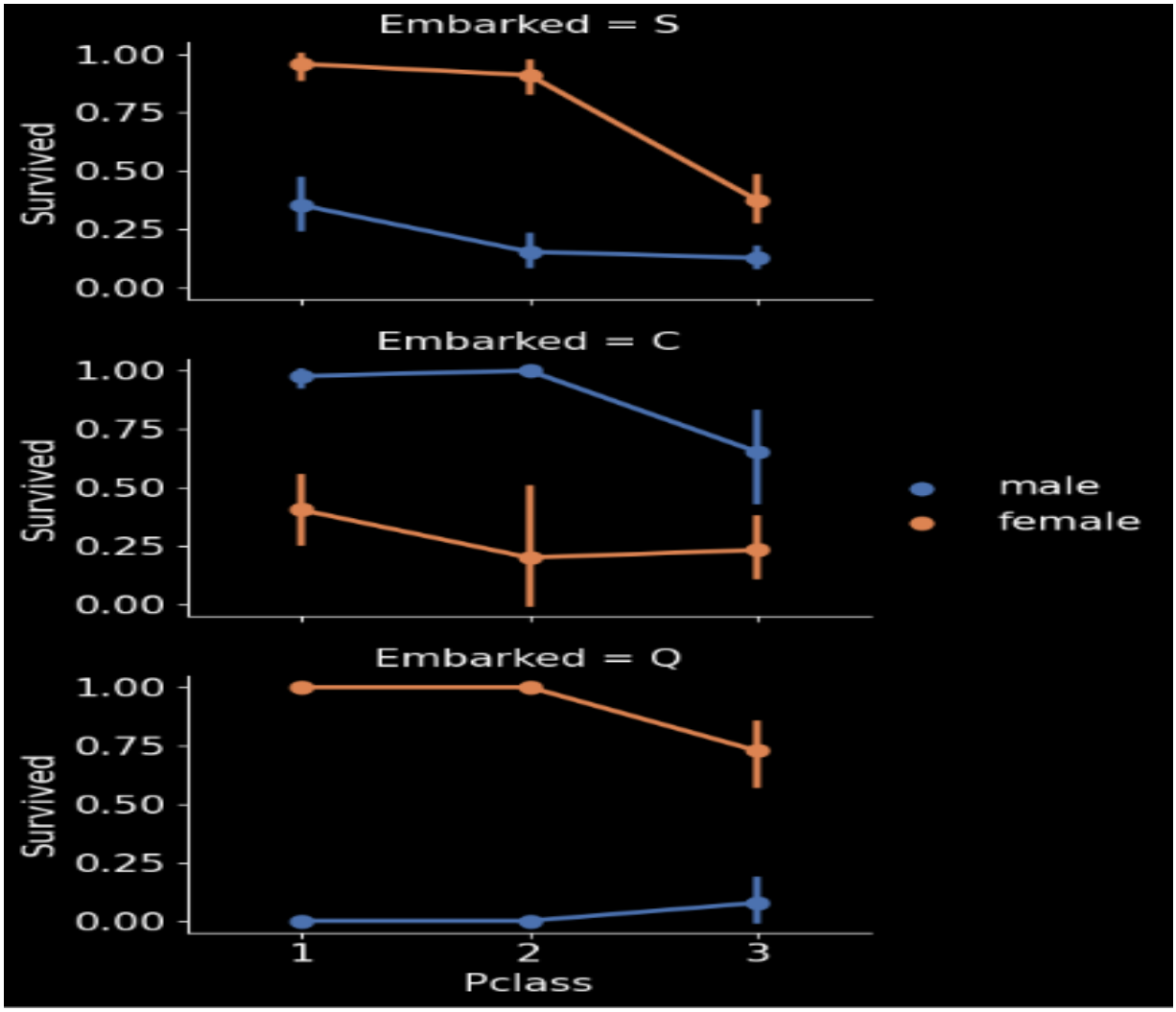



```
pd.crosstab(titanic['Sex'],titanic['Embarked']).plot(kind='bar',stacke
d=True);
```



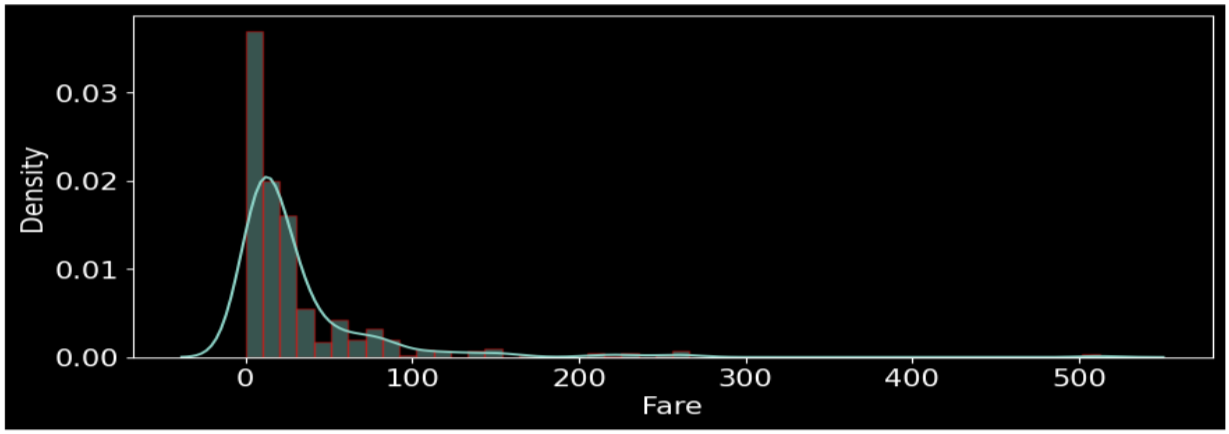

```
grid = sns.FacetGrid(titanic, row='Embarked', aspect=1.6)
grid.map(sns.pointplot, 'Pclass', 'Survived', 'Sex', palette='deep')
grid.add_legend();
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/



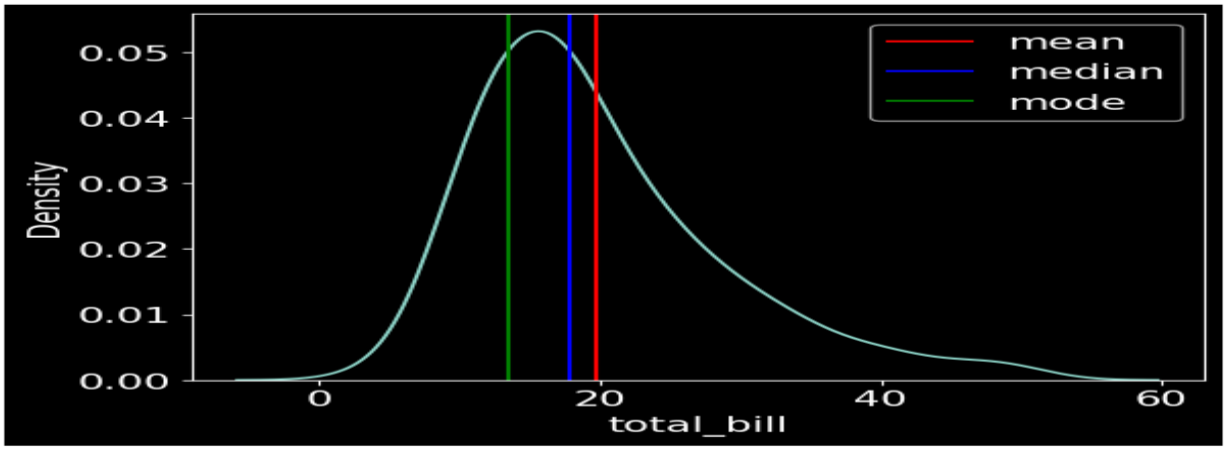

```
plt.figure(figsize=(10,4))
print("Skewness: %f" % titanic['Fare'].skew())
print("Kurtosis: %f" % titanic['Fare'].kurt())
sns.distplot(titanic['Fare'],bins=50,hist_kws={"edgecolor": (1,0,0,1)})
plt.show()
```

```
Skewness: 4.787317
Kurtosis: 33.398141
```

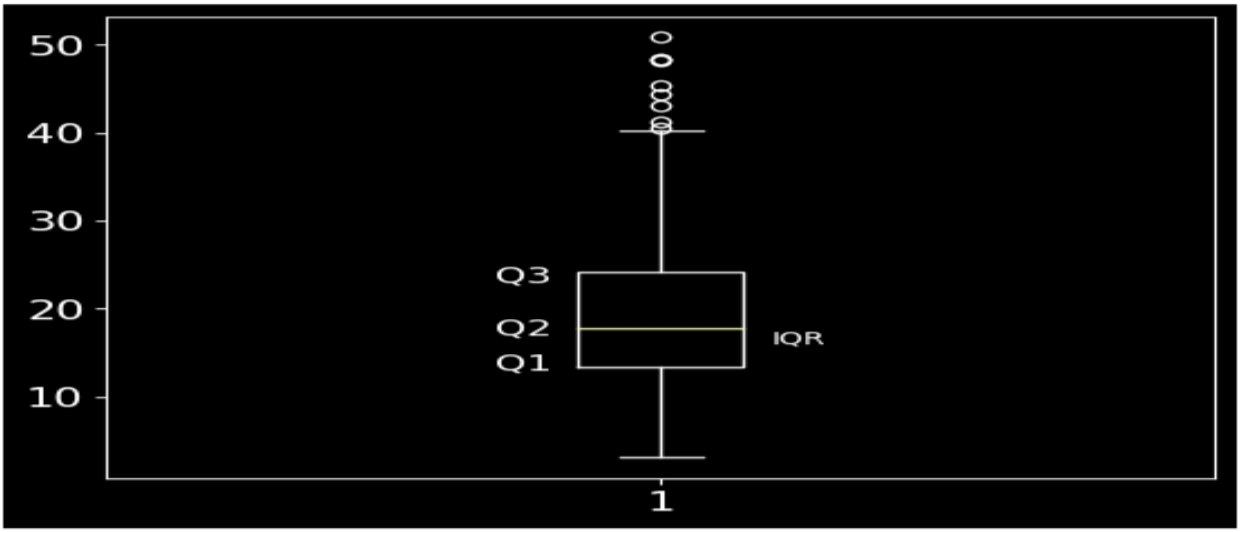


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
a=tips['total_bill']
mean=a.mean()
median=np.median(a)
mode=a.mode()
sns.distplot(a,hist=False)
plt.axvline(mean,color='r',label='mean')
plt.axvline(median,color='b',label='median')
plt.axvline(mode[0],color='g',label='mode')
plt.legend()
plt.show()
```

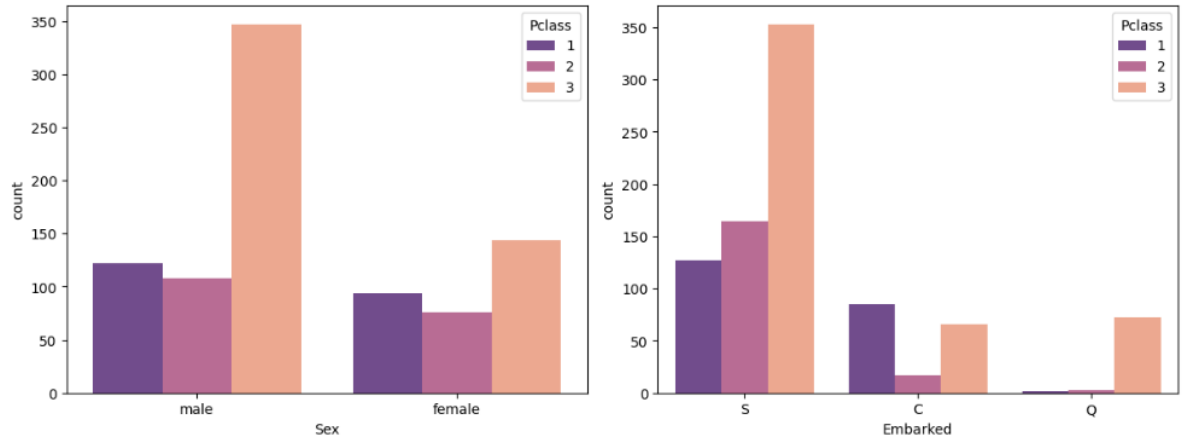



```
plt.boxplot(a)
plt.text(0.85,13,s='Q1',size=13)
plt.text(0.85,17,s='Q2',size=13)
plt.text(0.85,23,s='Q3',size=13)
plt.text(1.1,16,s='IQR',rotation=0,size=10)
plt.show()
```



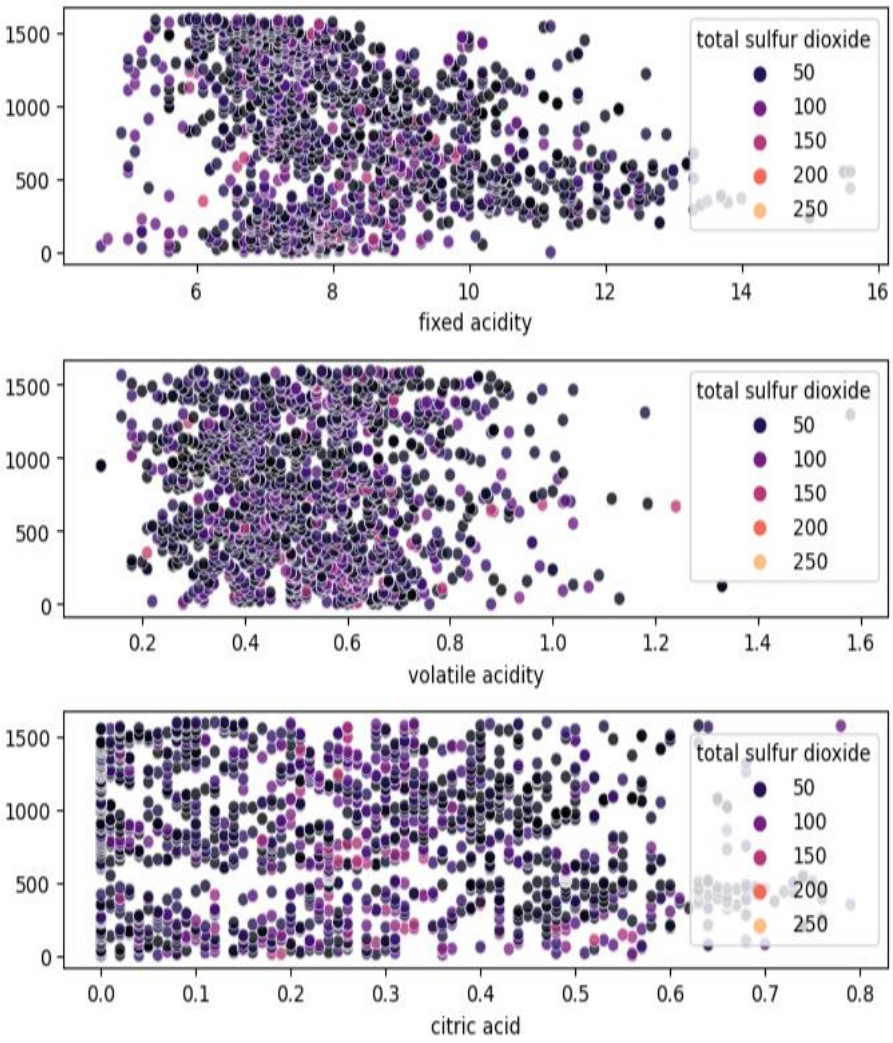
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
cat = ['Sex','Embarked']
sns.set_theme(rc = {'figure.dpi': 100, 'axes.labelsize': 12,
                    'axes.facecolor': '#f0eee9', 'grid.color': '#fffdfa',
                   'figure.facecolor': '#e8e6e1'}, font_scale = 1.2)
fig, ax = plt.subplots(5, 2, figsize = (12, 22))
for indx, (column, axes) in list(enumerate(list(zip(cat,
                                        ax.flatten())))):

    sns.countplot(ax = axes, x = titanic[column], hue = titanic['Pclass'],
                  palette = 'magma', alpha = 0.8)

else:
    [axes.set_visible(False) for axes in ax.flatten()[indx + 1:]]
plt.tight_layout()
plt.show()
```



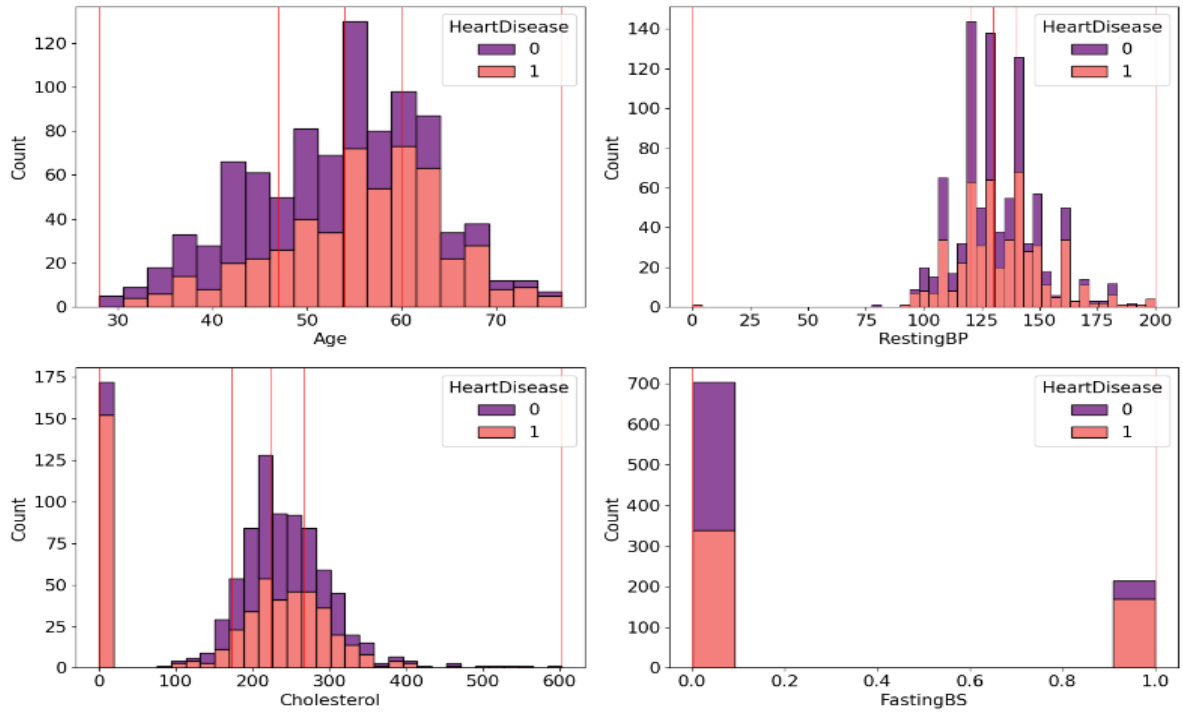

```
num = wine.select_dtypes(include="number")
fig, ax = plt.subplots(14, 1, figsize = (7, 30))
for indx, (column, axes) in list(enumerate(list(zip(num, ax.flatten())))):

    sns.scatterplot(ax = axes, y = wine[column].index, x = wine[column],hue =
wine['total sulfur dioxide'],
                    palette = 'magma', alpha = 0.8)

else:
    [axes.set_visible(False) for axes in ax.flatten()[indx + 1:]]
plt.tight_layout()
plt.show()
```


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

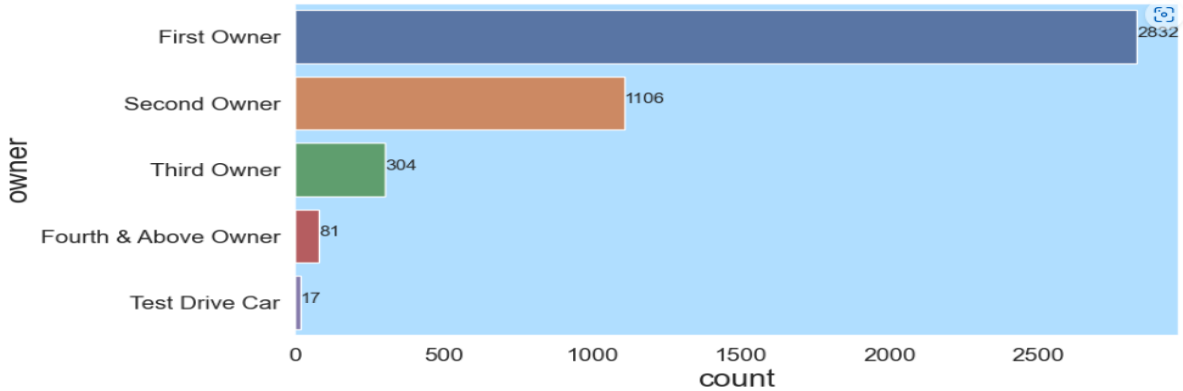



```
num = heart.select_dtypes(include="number")
fig, ax = plt.subplots(3, 2, figsize = (14, 15))
for indx, (column, axes) in list(enumerate(list(zip(num, ax.flatten())))):

    sns.histplot(ax = axes, x = heart[column],hue = heart['HeartDisease'],
                 palette = 'magma', alpha = 0.8, multiple = 'stack')

    legend = axes.get_legend() # sns.hisplot has some issues with legend
    handles = legend.legendHandles
    legend.remove()
    axes.legend(handles, ['0', '1'], title = 'HeartDisease', loc = 'upper right')
    Quantiles = np.quantile(heart[column], [0, 0.25, 0.50, 0.75, 1])

    for q in Quantiles: axes.axvline(x = q, linewidth = 0.5, color = 'r')
plt.tight_layout()
plt.show()
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

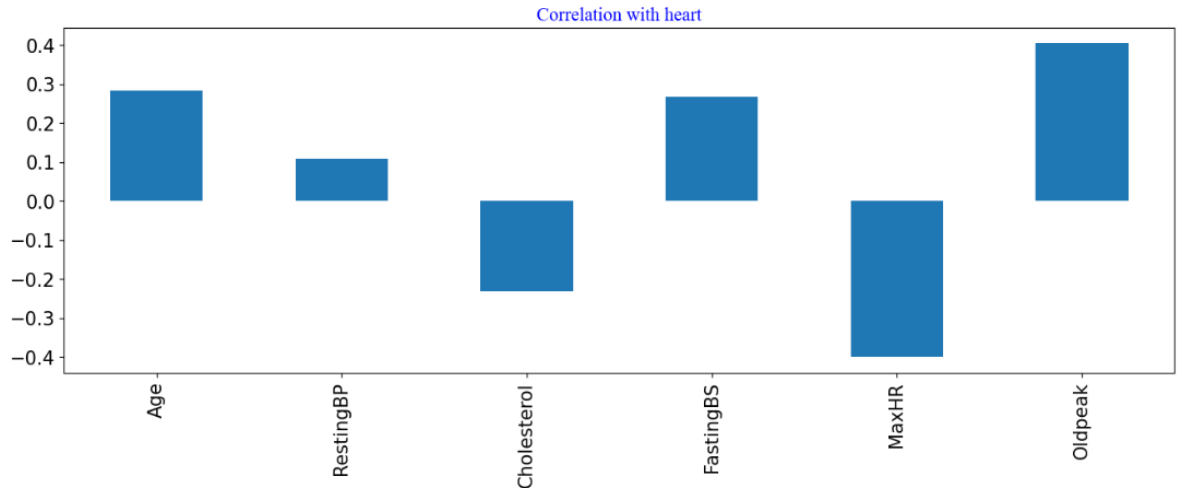



```
raw_df = raw_df [['name', 'year', 'selling_price', 'km_driven', 'fuel',
'seller_type',
       'transmission', 'owner']]
def barw(ax):
    for p in ax.patches:
        val = p.get_width() #height of the bar
        x = p.get_x()+ p.get_width() # x- position
        y = p.get_y() + p.get_height()/2 #y-position
        ax.annotate(round(val,2),(x,y))
plt.figure(figsize=(10,5))
ax0 = sns.countplot(data = raw_df, y ='owner', order =
raw_df['owner'].value_counts().index)
barw(ax0)
plt.show()
```



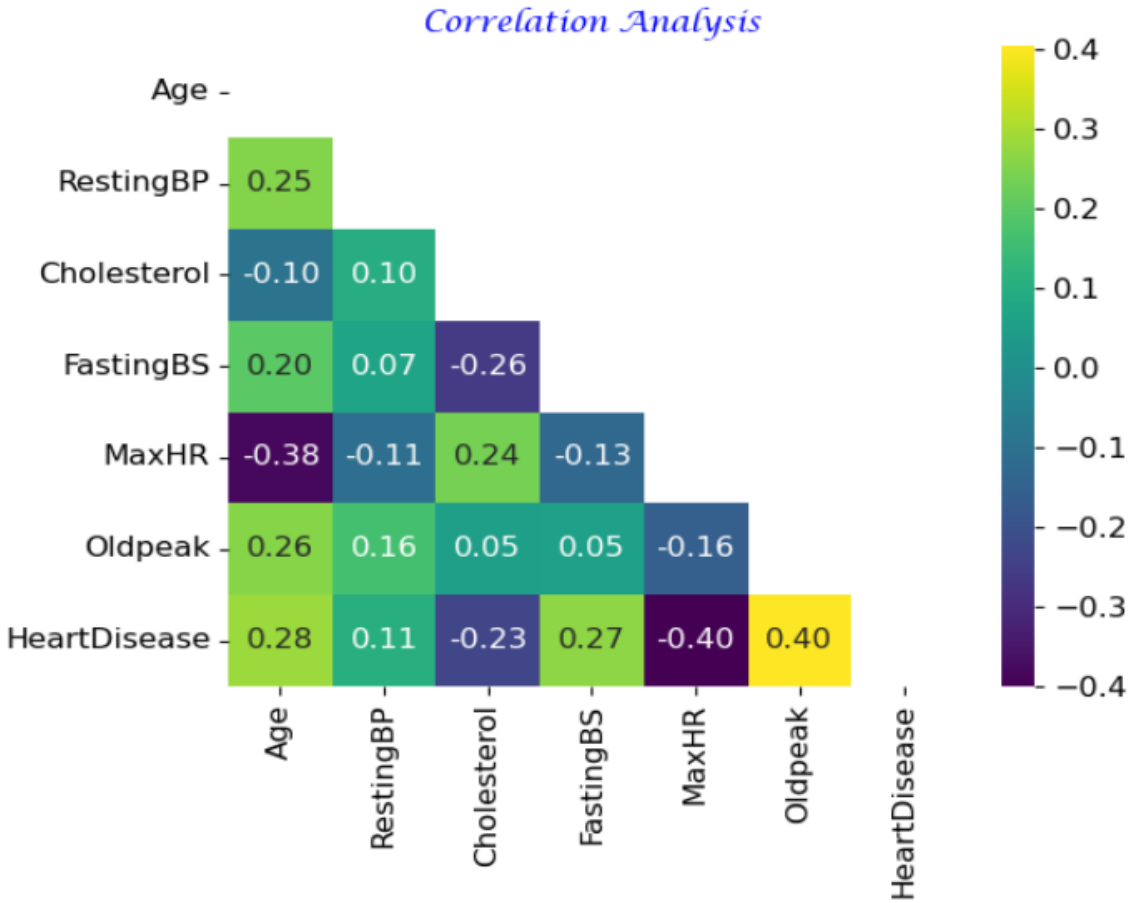
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
#Correlation with Response Variable class
X = heart.drop(['HeartDisease'], axis=1)
y = heart['HeartDisease']

X.corrwith(y).plot.bar(figsize=(16, 4), rot=90, grid=False)
plt.title('Correlation with heart',
          fontsize=25,
          color='Blue',
          font='Times New Roman')
plt.show()
```

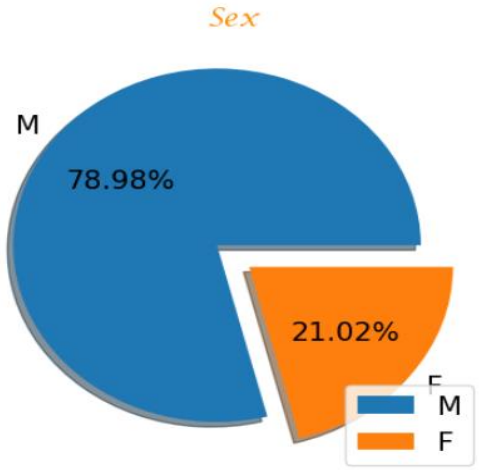



```
import matplotlib
matplotlib.rcParams.update({'font.size': 12})
corr = heart.corr()
mask = np.triu(np.ones_like(corr, dtype=bool))
plt.figure(dpi=100)
plt.title('Correlation Analysis',
          fontsize=15,
          color='Blue',
          font='Lucida Calligraphy')
sns.heatmap(corr,
            mask=mask,
            annot=True,
            lw=0,
            linecolor='white',
            cmap='viridis',
            fmt="0.2f")
plt.xticks(rotation=90)
plt.yticks(rotation=0)
plt.show()
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

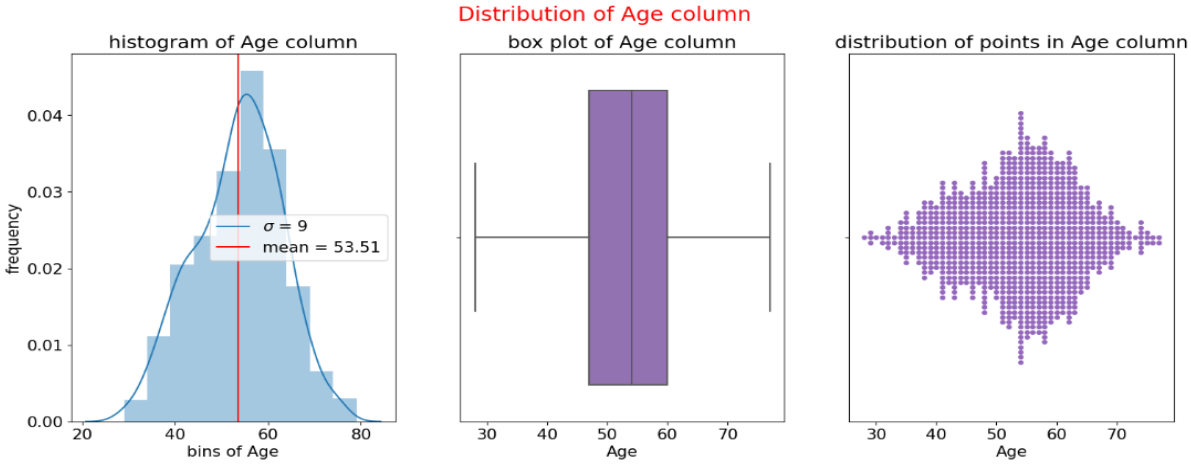



```
matplotlib.rcParams.update({'font.size': 15})
ax=heart['Sex'].value_counts().plot.pie(explode=[0.1,
0.1],autopct='%1.2f%%',shadow=True);
ax.set_title(label = "Sex", fontsize = 40,color='DarkOrange',font='Lucida
Calligraphy');
plt.legend(labels=['M','F'])
plt.axis('off');
```



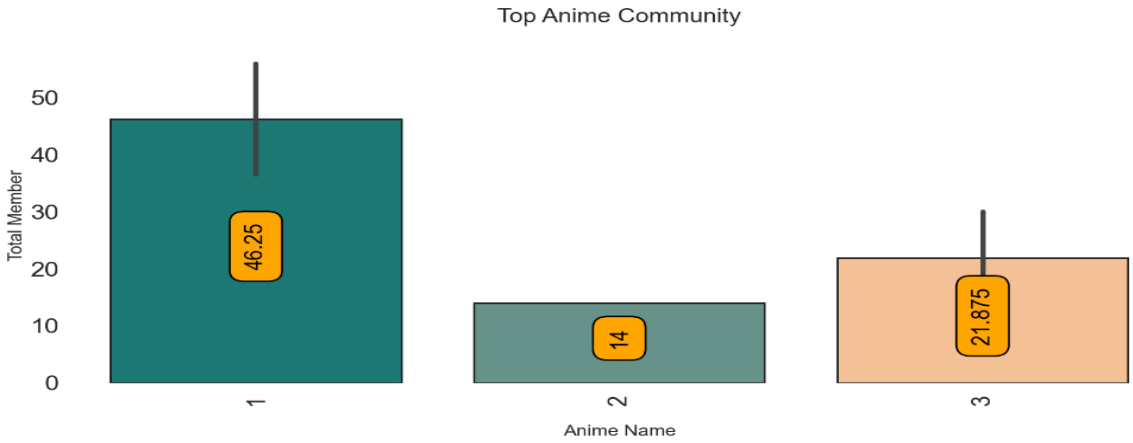
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
#set configuration for charts
plt.rcParams["figure.figsize"]=[18 , 7]
plt.rcParams["font.size"]=15
plt.rcParams["legend.fontsize"]="medium"
plt.rcParams["figure.titlesize"]="medium"

def plot_disribution(data , x ,color,bins ):
    mean = data[x].mean()
    std = data[x].std()
    info=dict(data = data , x = x , color = color)
    plt.subplot(1 , 3 , 1 , title =f"Ditstribution of {x} column")
    sns.distplot(a=data[x] , bins = bins)
    plt.xlabel(f"bins of {x}")
    plt.axvline(mean , label ="mean" , color ="red")
    plt.ylabel("frequency")
    plt.legend(["${\sigma}$ = %d"%std , f"mean = {mean:.2f}"])
    plt.title(f"histogram of {x} column")
    plt.subplot(1 , 3 , 2)
    sns.boxplot(**info)
    plt.xlabel(f"{x}")
    plt.title(f"box plot of {x} column")
    plt.subplot(1 , 3 , 3)
    sns.swarmplot(**info)
    plt.xlabel(f"{x}")
    plt.title(f"distribution of points in {x} column")
    plt.suptitle(f"Distribution of {x} column" , fontsize =20 , color="red")
plt.show()

age_bins = np.arange(29 , 77+5 , 5)
base_color = sns.color_palette()[4]
plot_disribution(data = heart , x ="Age" , color = base_color ,
bins=age_bins)
```



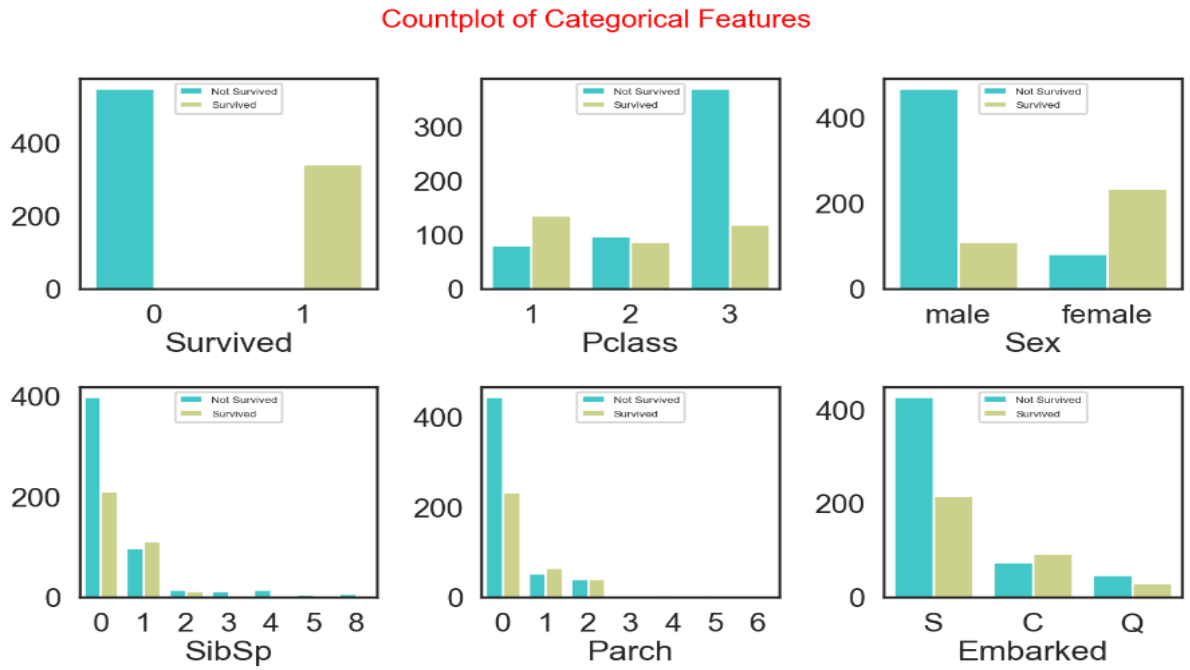
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
sns.set_style("white")
sns.set_context("poster",font_scale = 1.2)
palette =
["#1d7874","#679289","#f4c095","#ee2e31","#ffb563","#918450","#f85e00","#a416
23","#9a031e","#d6d6d6","#ffee32","#ffd100","#333533","#202020"]
plt.subplots(figsize=(20,8))
p = sns.barplot(x=titanic["Pclass"][:14],y=titanic["Age"],palette=palette,
saturation=1, edgecolor = "#1c1c1c", linewidth = 2)
p.axes.set_title("\nTop Anime Community\n", fontsize=25)
plt.ylabel("Total Member" , fontsize = 20)
plt.xlabel("\nAnime Name" , fontsize = 20)
plt.xticks(rotation = 90)
for container in p.containers:
    p.bar_label(container,label_type = "center",padding = 6,size = 25,color =
"black",rotation = 90,
    bbox={"boxstyle": "round", "pad": 0.6, "facecolor": "orange", "edgecolor":
"black", "alpha": 1})

sns.despine(left=True, bottom=True)
plt.show()
```

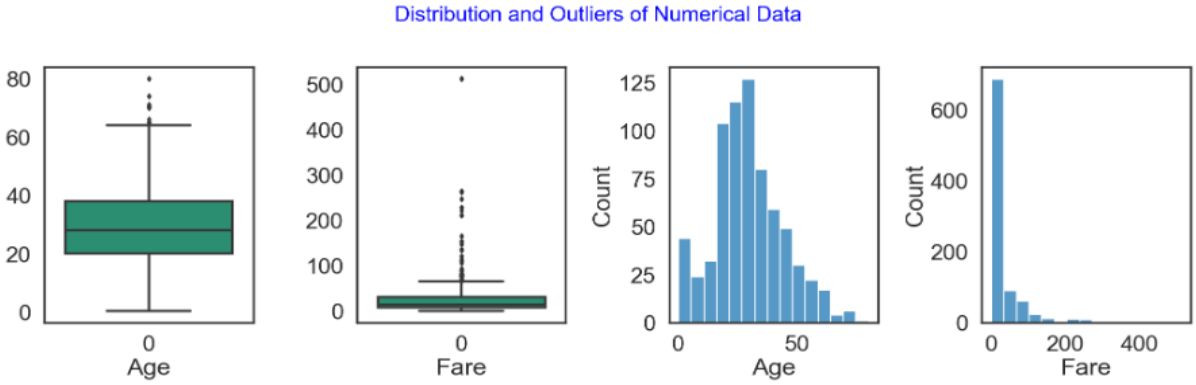



```
countfeature = ["Survived", "Pclass", "Sex", "SibSp", "Parch", "Embarked"]
countlist = list(enumerate(countfeature))
plt.figure(figsize = (15,10))
plt.suptitle("Countplot of Categorical Features", fontsize=25,color='Red')
for i in countlist:
    plt.subplot(2,3,i[0]+1)
    sns.countplot(data = titanic, x = i[1], hue = "Survived",
palette="rainbow")
    plt.ylabel("")
    plt.legend(['Not Survived', 'Survived'], loc='upper center', prop={'size':
10})
plt.tight_layout()
plt.show()
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

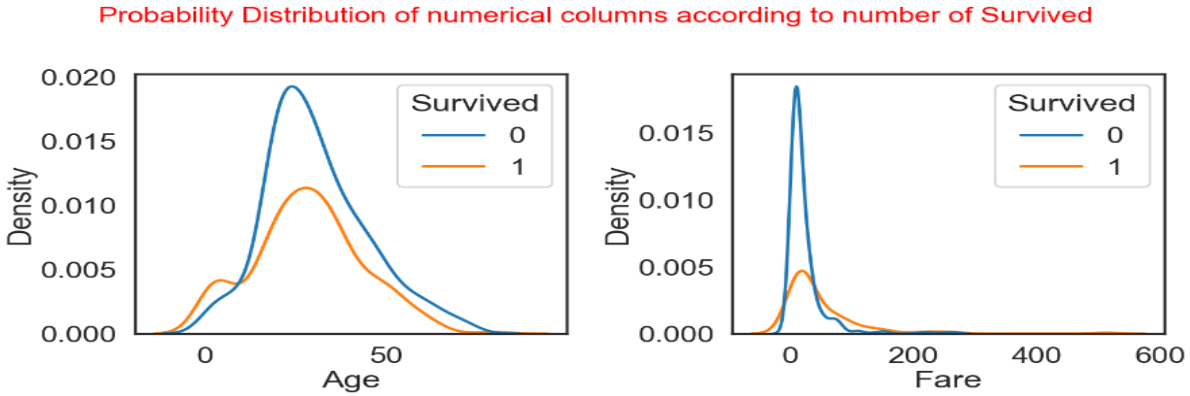



```
numfeature = ["Age", "Fare"]
enumfeat = list(enumerate(numfeature))

plt.figure(figsize=(20,7))
plt.suptitle("Distribution and Outliers of Numerical Data",
fontsize=25,color='Blue')
for i in enumfeat:
    plt.subplot(1,4,i[0]+1)
    sns.boxplot(data = titanic[i[1]], palette="Dark2")
    plt.xlabel(str(i[1]))
for i in enumfeat:
    plt.subplot(1,4,i[0]+3)
    sns.histplot(data = titanic[i[1]], palette="tab10", bins=15)
    plt.xlabel(str(i[1]))
plt.tight_layout()
plt.show()
```

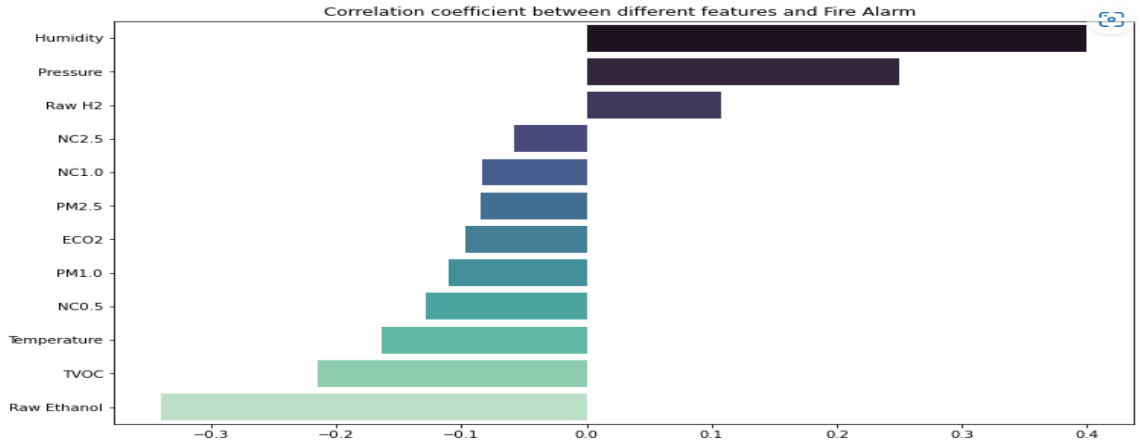


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
plt.figure(figsize=(15,7))
plt.suptitle("Probability Distribution of numerical columns according to number of Survived", fontsize = 25,color="Red")
for i in enumfeat:
    plt.subplot(1,2,i[0]+1)
    sns.kdeplot(data=titanic, x=i[1], hue="Survived")
plt.tight_layout()
plt.show()
```

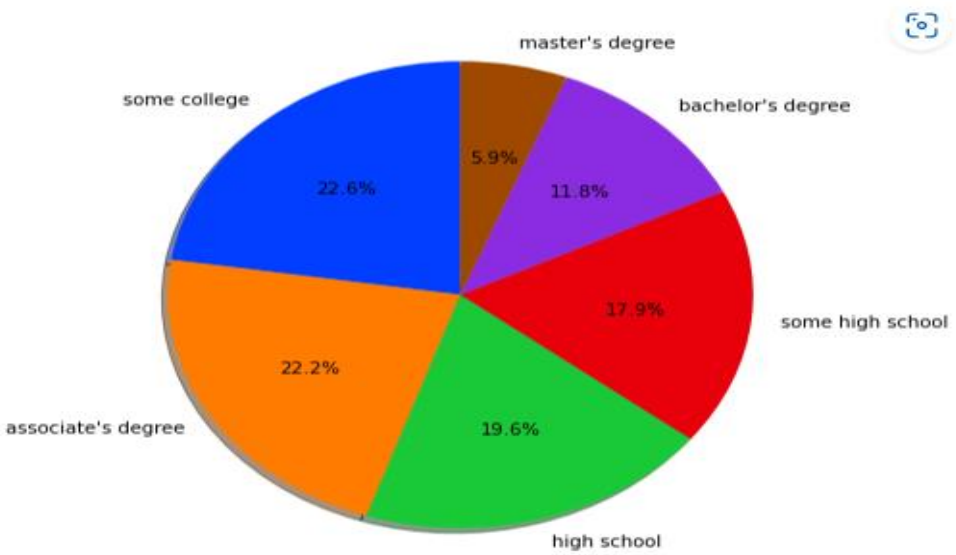



```
plt.figure(figsize=(12,8))
data_4 = data.corr()["Fire Alarm"].sort_values(ascending=False)
indices = data_4.index
labels = []
corr = []
for i in range(1, len(indices)):
    labels.append(indices[i])
    corr.append(data_4[i])
sns.barplot(x=corr, y=labels, palette='mako')
plt.title('Correlation coefficient between different features and Fire Alarm ')
plt.show()
```

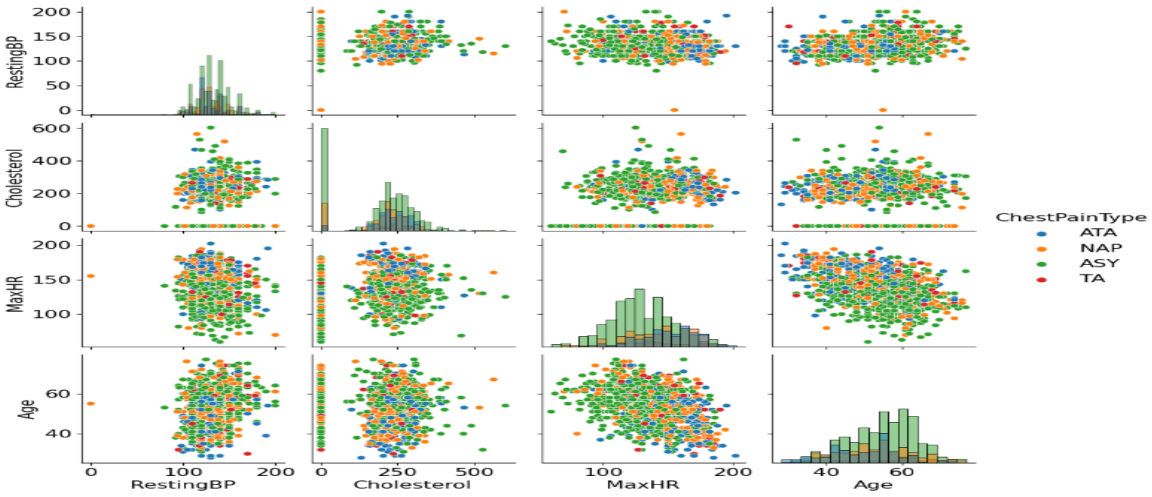


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
education=df['parental level of education'].value_counts()
sns.set_palette('bright')
plt.figure(figsize=(10,7))
labels=education.index
sizes=education.values
plt.pie(sizes,labels=labels,autopct='%1.1f%%',
        shadow=True,startangle=90)
plt.show()
```

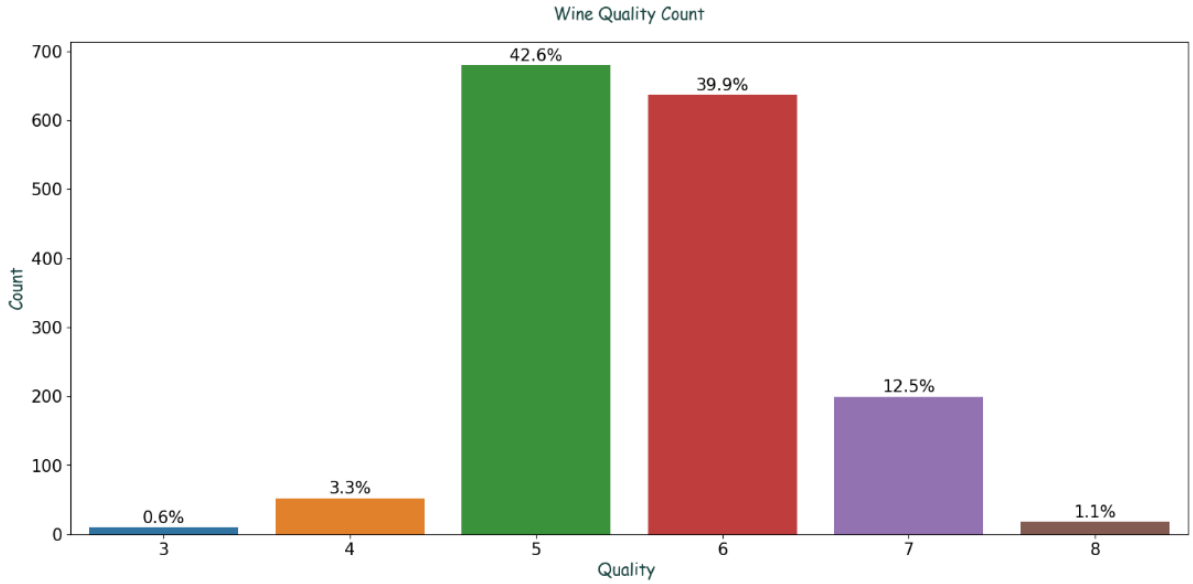



```
import matplotlib
matplotlib.rcParams.update({'font.size': 15})
plt.figure(figsize=(18,9))
cols_out = ["RestingBP", "Cholesterol", "MaxHR", "Age",'ChestPainType']
sns.pairplot(heart[cols_out], hue="ChestPainType", diag_kind="hist",
palette="tab10") # tab10
plt.show();
```



Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

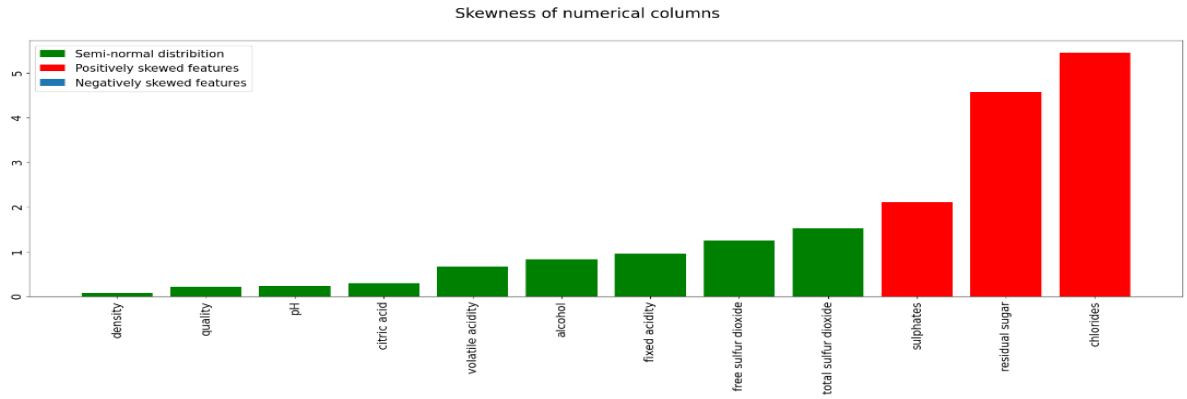
```
fig, ax = plt.subplots(figsize = (18,8))
sns.countplot(x= wine["quality"])
plt.title("Wine Quality Count",fontsize=20,color='#1a4441',font='Comic Sans Ms',pad=20)
plt.xlabel("Quality ",fontsize=15,color='#1a4441',font='Comic Sans Ms')
plt.ylabel("Count",fontsize=15,color='#1a4441',font='Comic Sans Ms');

total = len(wine)
for p in ax.patches:
    percentage = f'{100 * p.get_height() / total:.1f}%\n'
    x = p.get_x() + p.get_width() / 2
    y = p.get_height()
    ax.annotate(percentage, (x, y), ha='center', va='center')
```




```
print("Skewly distributed columns by skewness value:\n")
skew_df = wine.skew().sort_values()

fig,ax = plt.subplots(figsize=(25,7))
ax.bar(x = skew_df[(skew_df<2)& (skew_df>-2)].index, height = skew_df[(skew_df<2)& (skew_df>-2)], color = "g", label= "Semi-normal distribition")
ax.bar(x = skew_df[skew_df>2].index, height = skew_df[skew_df>2], color = "r", label = "Positively skewed features")
ax.bar(x = skew_df[skew_df<-2].index, height = skew_df[skew_df<-2], color = "b", label = "Negatively skewed features")
ax.legend()
fig.suptitle("Skewness of numerical columns",fontsize = 20)
ax.tick_params(labelrotation=90);
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

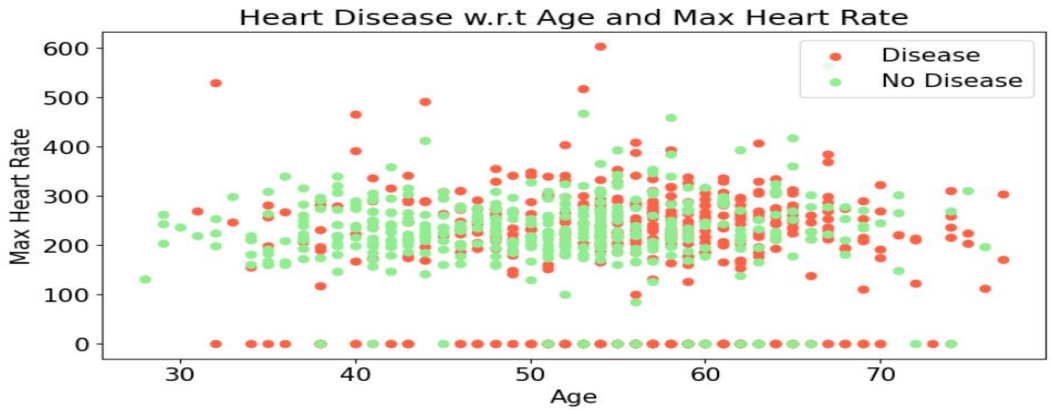
```
Skewly distributed columns by skewness value:
```



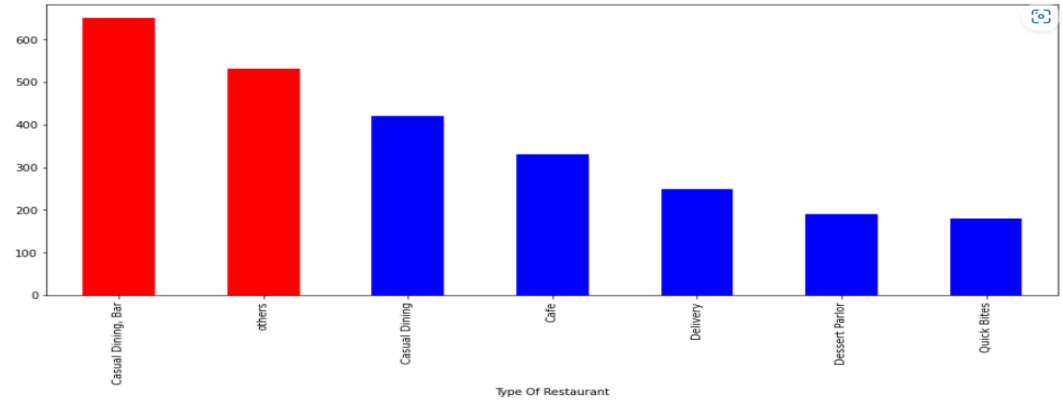

```
from wordcloud import WordCloud, STOPWORDS
text = " ".join(Company for Company in df["Cuisines"])
#font = "Quicksand-Bold.ttf"
word_cloud = WordCloud(width = 2300,
                       height = 800,
                       colormap = 'jet',
                       background_color = "white").generate(text)
plt.figure(figsize = (50, 8))
plt.imshow(word_cloud, interpolation = "gaussian")
plt.axis("off")
plt.show()
```


```
plt.figure(figsize=(10,5))
#plotting the values for people who have heart disease
plt.scatter(heart.Age[heart.HeartDisease==1],
            heart.Cholesterol[heart.HeartDisease==1],
            c="tomato")
#plotting the values for people who doesn't have heart disease
plt.scatter(heart.Age[heart.HeartDisease==0],
            heart.Cholesterol[heart.HeartDisease==0],
            c="lightgreen")
plt.title("Heart Disease w.r.t Age and Max Heart Rate")
plt.xlabel("Age")
plt.legend(["Disease", "No Disease"])
plt.ylabel("Max Heart Rate");
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

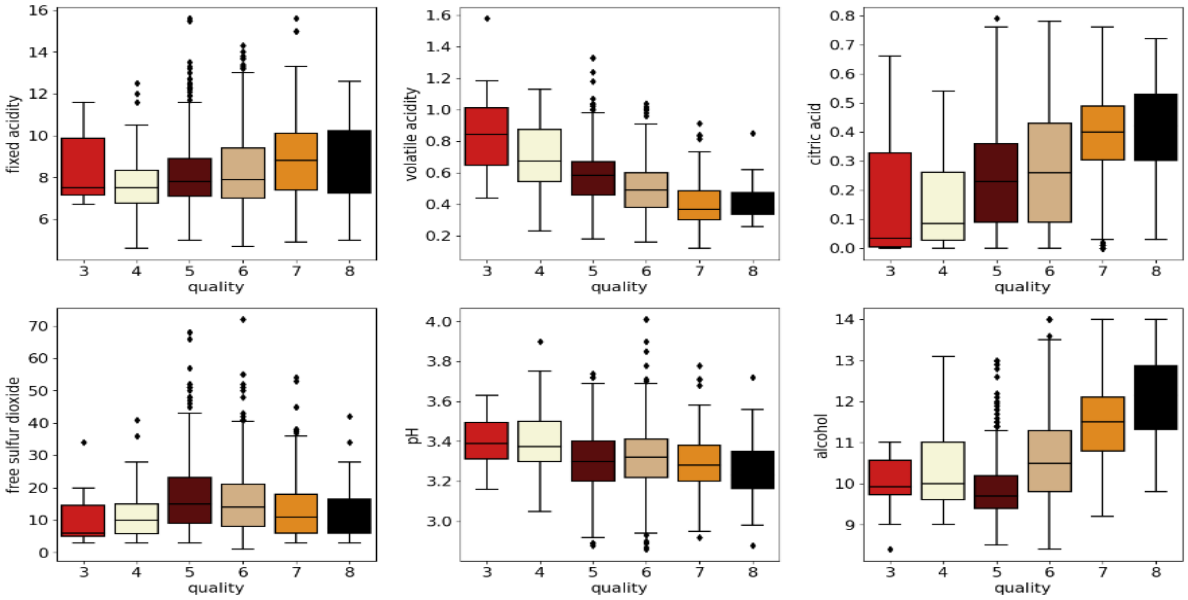



```
df2=df.groupby('Type Of Restaurant')['Cost Per Head'].mean().sort_values(ascending=False)
plt.figure(figsize = (15,6))
color = [('b' if i < 500 else 'r') for i in df2]
df2.plot.bar(color=color);
```

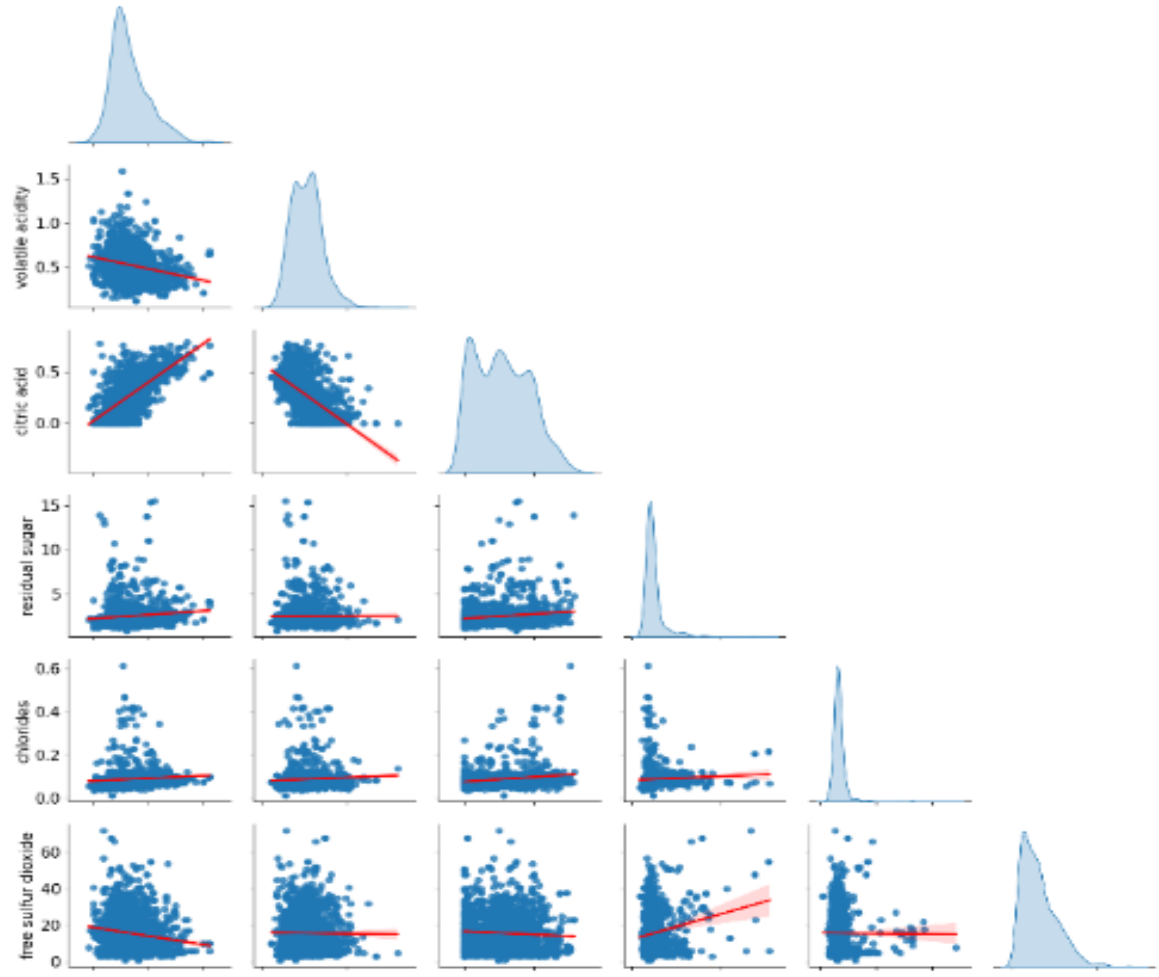



```
import math
cont_features=['fixed acidity', 'volatile acidity', 'citric acid','free sulfur dioxide','pH', 'alcohol']

y=3
x=math.ceil(len(cont_features)/y)

plt.subplots(x,y,figsize=(15,10))
for i in range(1,len(cont_features)+1) :
    plt.subplot(x,y,i)
    sns.boxplot(data=wine,y=cont_features[i-1],x='quality',palette=['#e60000','#FAFAD2','#660000','#DEB078','#FF8C00','black'])
plt.tight_layout()
plt.show()
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

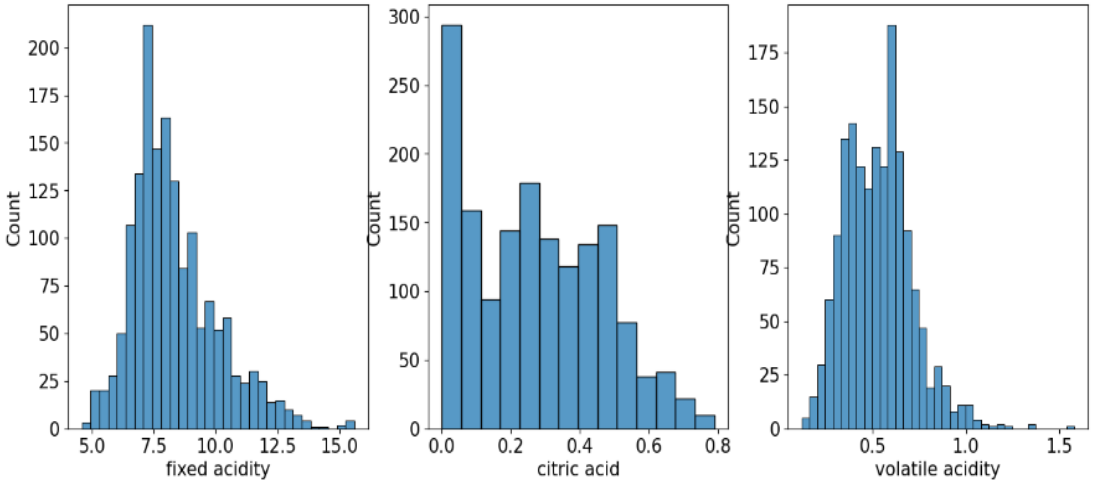



```
sns.pairplot(wine.drop(columns=['quality']),kind="reg",diag_kind='kde',plot_kws={'line_kws':{'color':'red'}},corner=True)
plt.tight_layout()
plt.show()
```

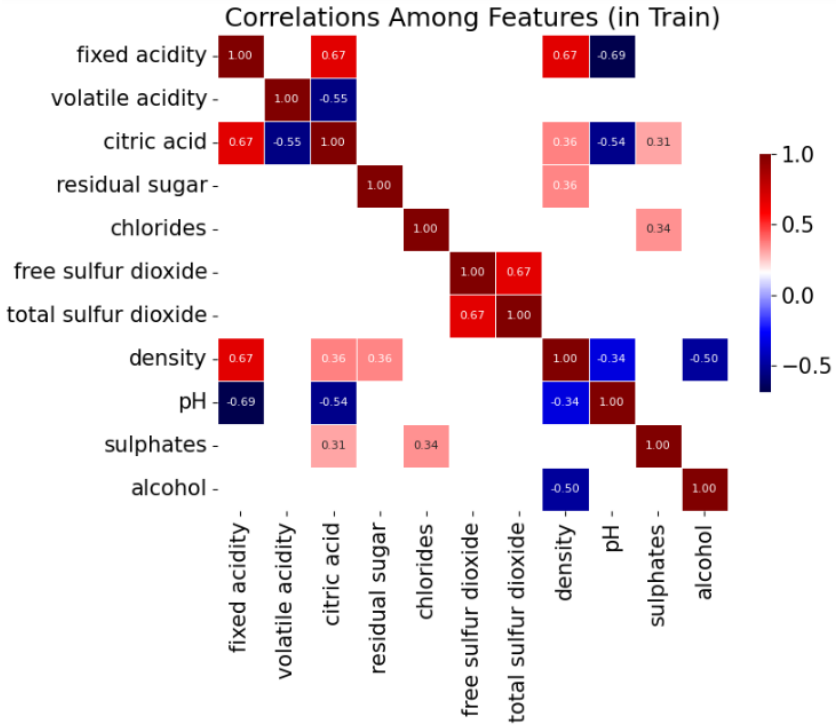


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
features = ['fixed acidity','citric acid','volatile acidity']
fig, axs = plt.subplots(1,3, figsize=(16,6))
for f, ax in zip(features,axs.ravel()):
    sns.histplot(wine, x=f, ax=ax)
plt.show()
```

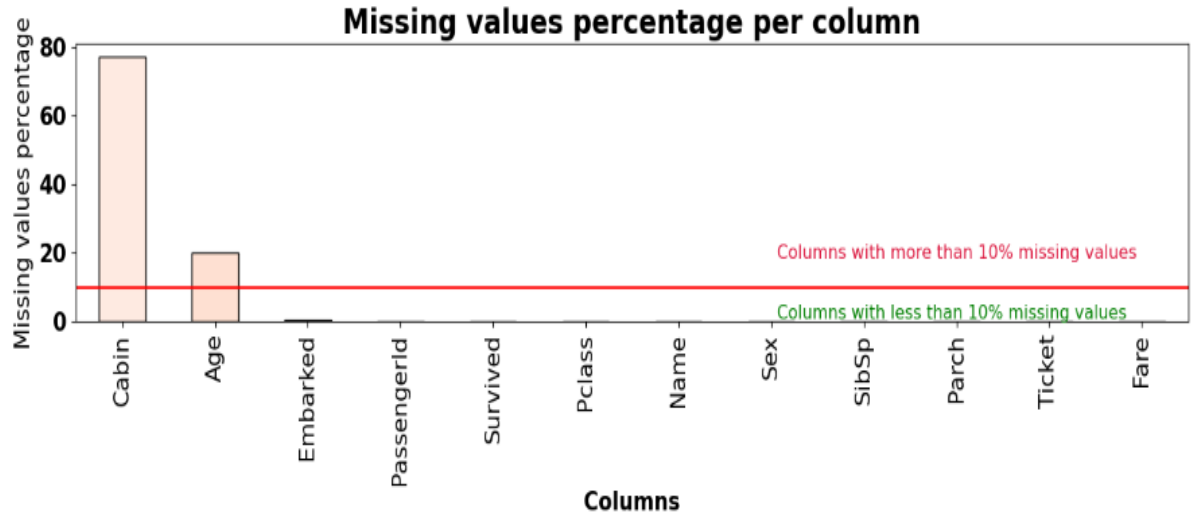



```
corr_mat_train = wine.drop(columns = ['quality'], axis = 1).corr()
threshold = 0.3
corr_threshold_train = corr_mat_train[(corr_mat_train > threshold) | (corr_mat_train
< -threshold)]
plt.figure(figsize = (8, 6))
sns.heatmap(corr_threshold_train, annot = True, cmap = 'seismic', fmt = ".2f",
            linewidths = 0.5, cbar_kws={'shrink': .5},annot_kws={'size':
8}).set_title('Correlations Among Features (in Train)');
```



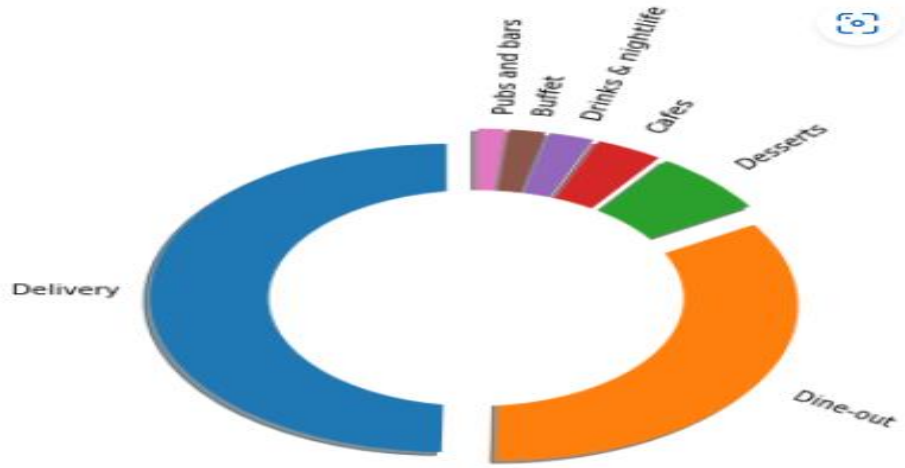
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

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

def missing_values(data, thresh = 20, color = 'black', edgecolor = 'black',
height = 3, width = 15):

    plt.figure(figsize = (width, height))
    percentage = (data.isnull().mean()) * 100
    percentage.sort_values(ascending = False).plot.bar(color = color,
edgecolor = edgecolor)
    plt.axhline(y = thresh, color = 'r', linestyle = '-')

    plt.title('Missing values percentage per column', fontsize = 20, weight =
'bold' )

    plt.text(len(data.isnull().sum()/len(data))/1.7, thresh + 12.5, f'Columns
with more than {thresh}% missing values', fontsize = 12, color = 'crimson',
         ha = 'left' ,va = 'top')
    plt.text(len(data.isnull().sum()/len(data))/1.7, thresh - 5, f'Columns with
less than {thresh}% missing values', fontsize=12, color='green',
         ha = 'left' ,va = 'top')
    plt.xlabel('Columns', size = 15, weight = 'bold')
    plt.ylabel('Missing values percentage')
    plt.yticks(weight = 'bold')

    return plt.show()

missing_values(titanic, thresh = 10, color = sns.color_palette('Reds',15))
```

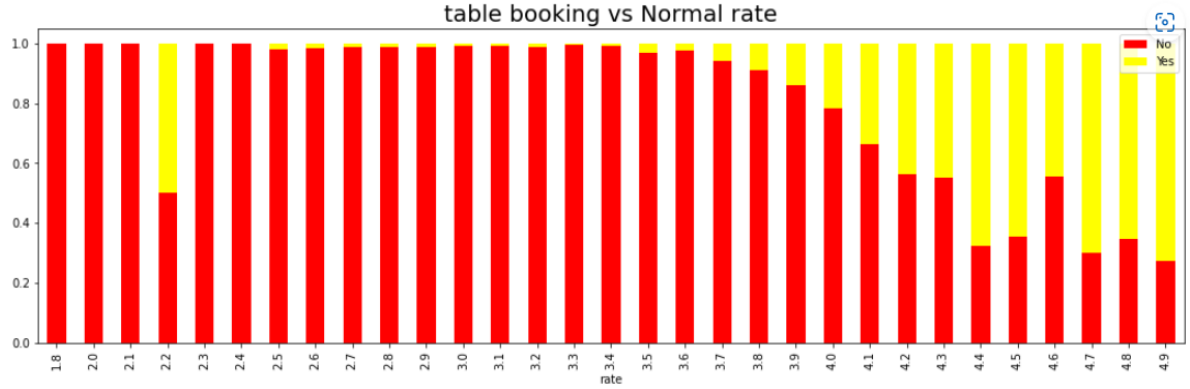


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
# Pie chart
labels = df['listed_in(type)'].value_counts().index
sizes = df['listed_in(type)'].value_counts().values
# only "explode" the 2nd slice (i.e. 'Hogs')
explode = (0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1)
fig1, ax1 = plt.subplots(figsize = (8, 8))
ax1.pie(sizes, labels = labels,
        shadow = True, startangle = 90, explode = explode, rotatelabels = True)
centre_circle = plt.Circle((0, 0), 0.70,fc = 'white')
fig = plt.gcf()
fig.gca().add_artist(centre_circle)
ax1.axis('equal')
plt.tight_layout()
plt.show()
```

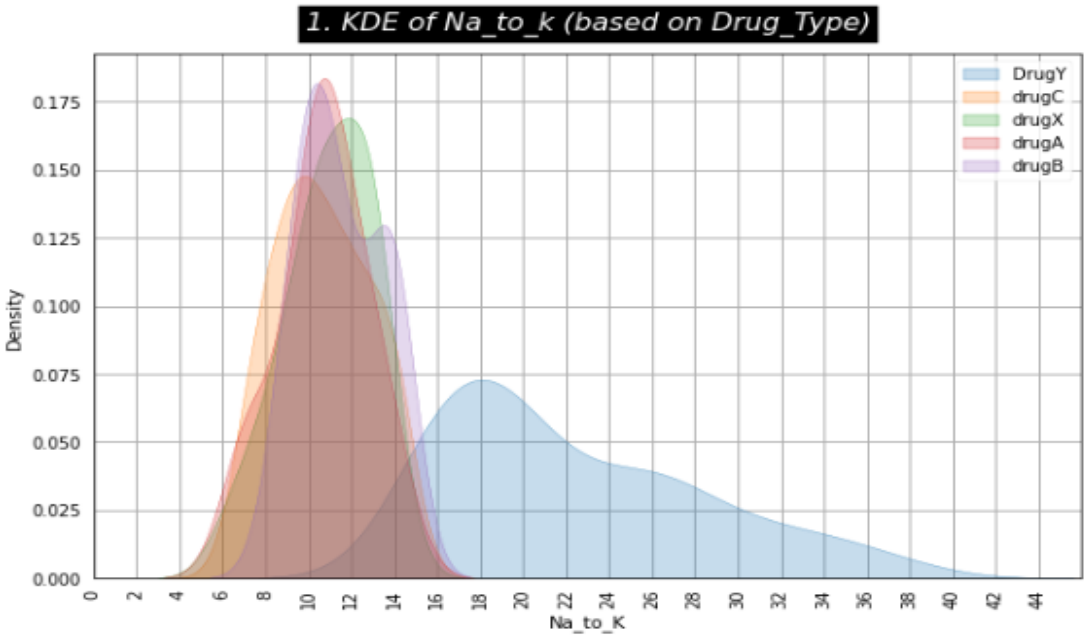



```
plt.rcParams['figure.figsize'] = (18, 5)
Y = pd.crosstab(df['rate'], df['book_table'])
Y.div(Y.sum(1).astype(float), axis = 0).plot(kind = 'bar', stacked =
True,color=['red','yellow'])
plt.title('table booking vs Normal rate', fontweight = 30, fontsize = 20)
plt.legend(loc="upper right")
plt.show()
```



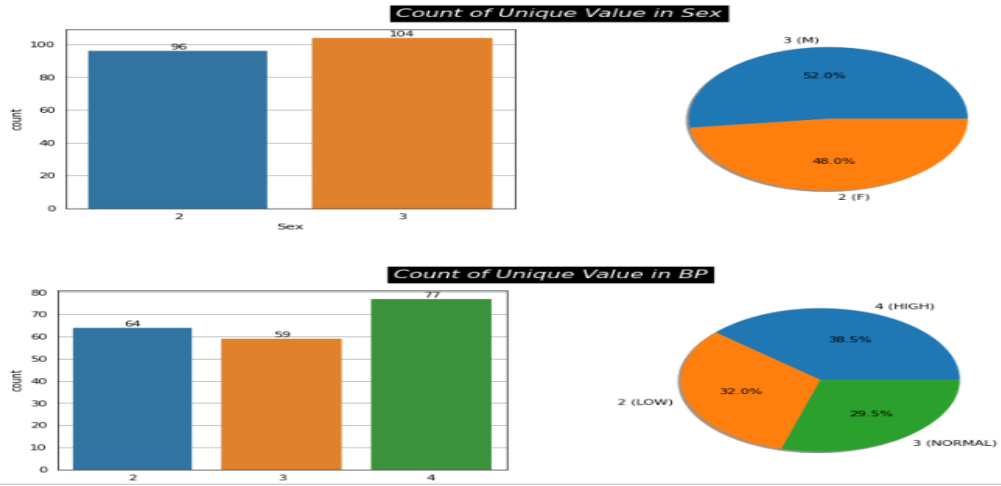
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
# check distribution of Na_to_k (based on Drug_Type)
%matplotlib inline
plt.style.use('seaborn-notebook')
for i, label in enumerate(df.Drug_Type.unique().tolist()):
    sns.kdeplot(df2.loc[df2['Drug_Type'] == i+1, 'Na_to_K'],
label=label, shade=True)
plt.title('1. KDE of Na_to_k (based on Drug_Type)', fontdict=font,
pad=15)
plt.xticks(np.arange(0,46,2), rotation=90)
plt.xlim([0,46])
plt.legend()
plt.show()
```

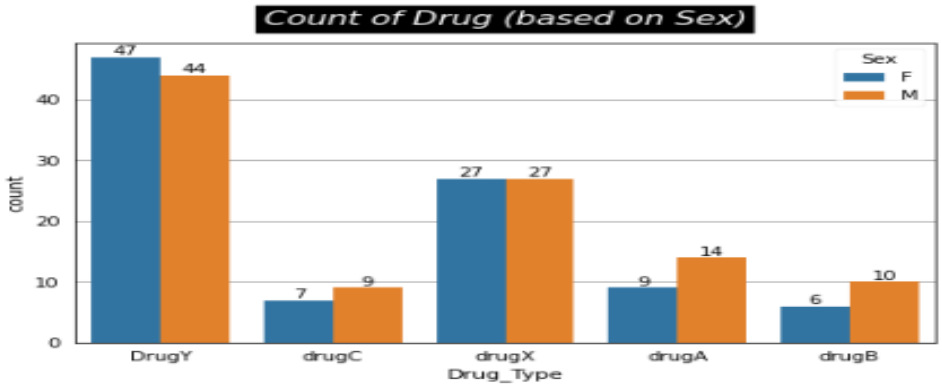



```
# draw countplot and pie plot of categorical data
for col in categorical:
    fig, axes = plt.subplots(1,2,figsize=(10,4))
    # count of col (countplot)
    sns.countplot(data=df2, x=col, ax=axes[0])
    for container in axes[0].containers:
        axes[0].bar_label(container)
    # count of col (pie chart)
    slices = df2[col].value_counts().values
    activities = [f"{i} ({var})" for i, var in zip(df2[col].value_counts().index,
df[col].value_counts().index)]
    axes[1].pie(slices, labels=activities, shadow=True, autopct='%1.1f%%')
    plt.suptitle(f'Count of Unique Value in {col}', y=1.09, **font)
    plt.show()
```


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

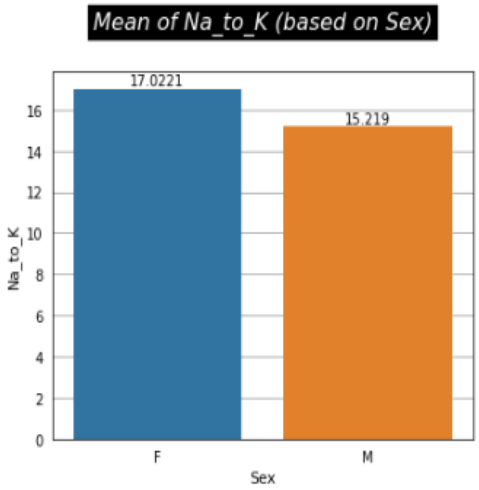


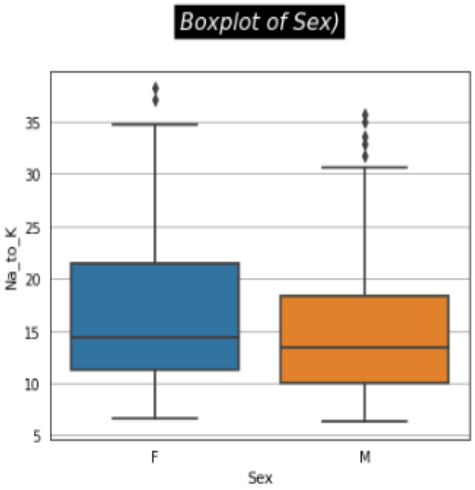

```
# count of purchased based on Gender
%matplotlib inline
for col in ['Sex','BP','Cholesterol']:
    ax = sns.countplot(data=df, x='Drug_Type', hue=col)
    for container in ax.containers:
        ax.bar_label(container)
    plt.title(f'Count of Drug (based on {col})', fontdict=font, pad=15)
    plt.show()
```



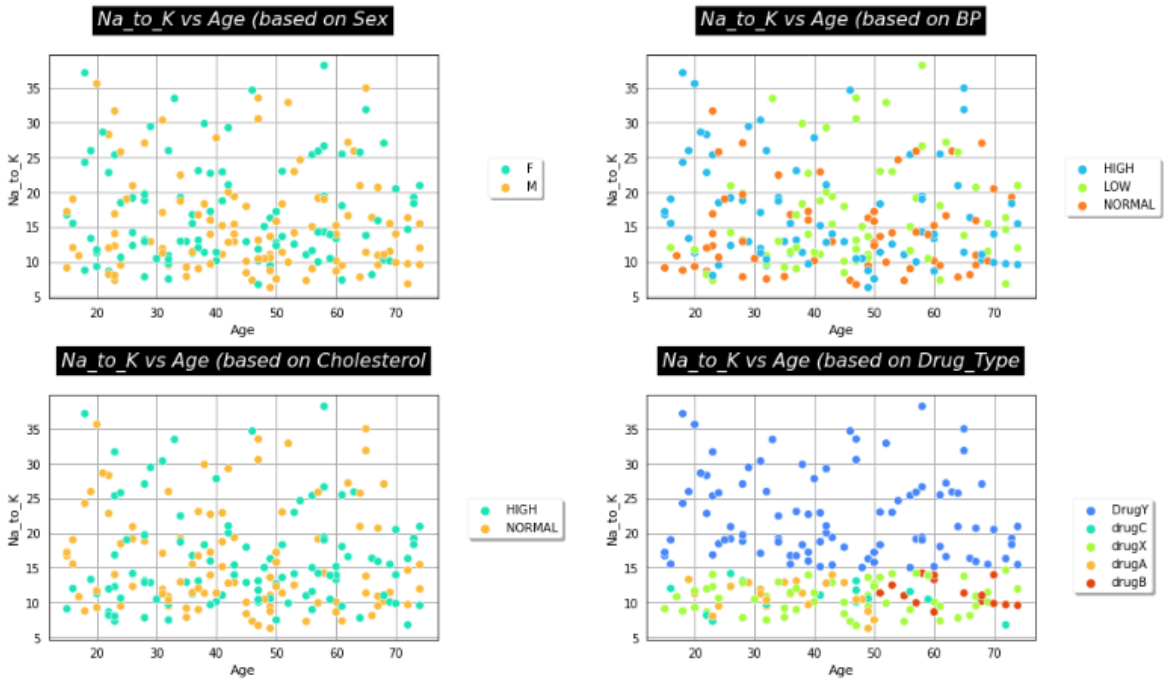

```
# Mean of Age and Na_to_K based on each feature
for col in ['Sex', 'BP', 'Cholesterol']:
    fig , ax= plt.subplots(1,2, figsize=(10,4))
    gp = df.groupby([col])['Na_to_K'].mean().to_frame().reset_index()
    sns.barplot(data=gp, x=col, y='Na_to_K', ax=ax[0])
    for container in ax[0].containers:
        ax[0].bar_label(container)
    ax[0].set_title(f'Mean of Na_to_K (based on {col})', y=1.09, **font)
    sns.boxplot(data=df, x=col, y='Na_to_K', ax=ax[1])
    ax[1].set_title(f'Boxplot of {col})', y=1.09, **font)
    plt.show()
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/



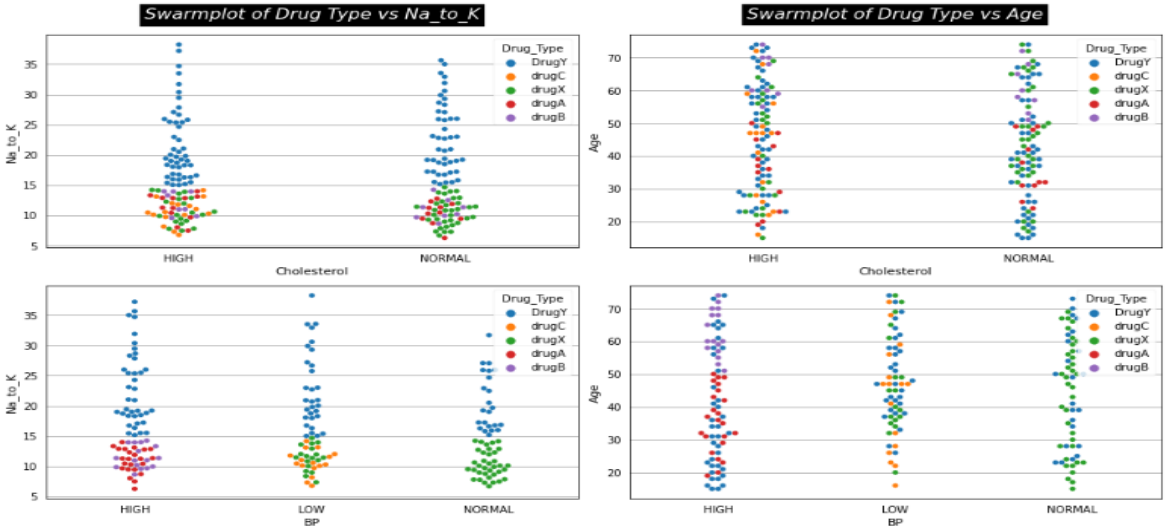



```
# use scatter plot for numerics feature (Age and Na_to_K)
fig, ax = plt.subplots(2,2,figsize=(14,8))
for i, col in enumerate(['Sex', 'BP', 'Cholesterol', 'Drug_Type']):
    sns.scatterplot(data=df, x='Age', y='Na_to_K', hue=col, ax=ax[i//2, i%2],
palette='turbo')
    ax[i//2, i%2].set_title(f'Na_to_K vs Age (based on {col}', y=1.09, **font)
    ax[i//2, i%2].legend(loc='upper center', bbox_to_anchor=(1.2, 0.6),
       fancybox=True, shadow=True)

fig.tight_layout()
plt.show()
```



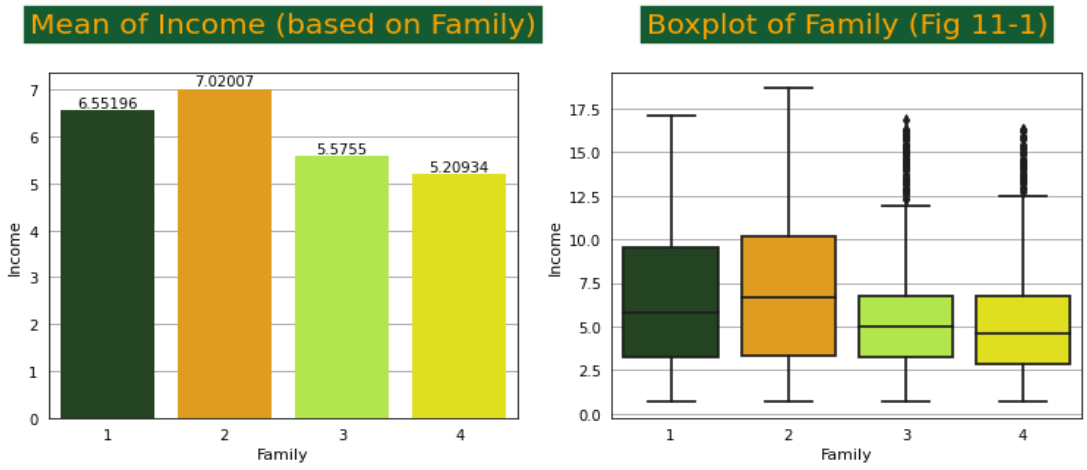
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
fig, ax = plt.subplots(3,2,figsize=(14,12))
sns.swarmplot(data=df, x='Cholesterol', y='Na_to_K', hue='Drug_Type',
ax=ax[0,0])
sns.swarmplot(data=df, x='Cholesterol', y='Age', hue='Drug_Type',
ax=ax[0,1])
sns.swarmplot(data=df, x='BP', y='Na_to_K', hue='Drug_Type', ax=ax[1,0])
sns.swarmplot(data=df, x='BP', y='Age', hue='Drug_Type', ax=ax[1,1])
sns.swarmplot(data=df, x='Sex', y='Na_to_K', hue='Drug_Type', ax=ax[2,0])
sns.swarmplot(data=df, x='Sex', y='Age', hue='Drug_Type', ax=ax[2,1])
ax[0,0].set_title('Swarmplot of Drug Type vs Na_to_K',y=1.05, **font)
ax[0,1].set_title('Swarmplot of Drug Type vs Age',y=1.05, **font)
plt.tight_layout()
plt.show()
```



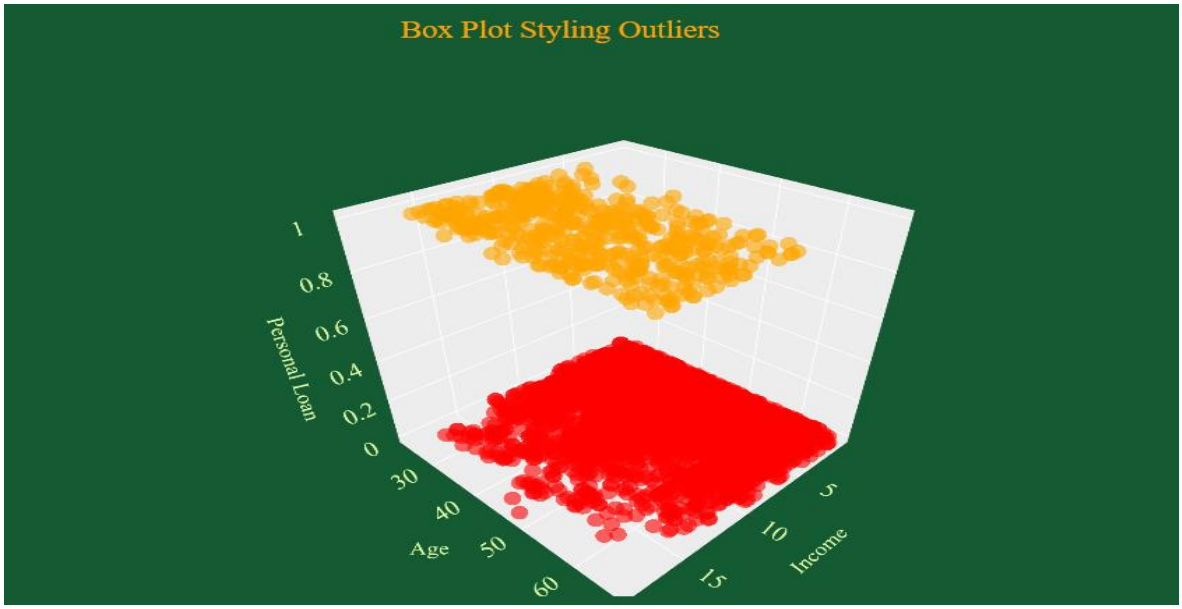

```
# Mean of Income and CCAvg based on each feature
for i, col in enumerate(['Income', 'CCAvg','Mortgage']):
    print('='*30, f"Mean of {col} in each categorical feature", '='*30)
    for j, cat in enumerate(discrete_cols2):
        fig , ax= plt.subplots(1,2, figsize=(10,4))
        gp = df.groupby([cat])[col].mean().to_frame().reset_index()
        sns.barplot(data=gp, x=cat, y=col, ax=ax[0])
        for container in ax[0].containers:
            ax[0].bar_label(container)
        ax[0].set_title(f'Mean of {col} (based on {cat})', y=1.09, **FONT)
        sns.boxplot(data=df, x=cat, y=col, ax=ax[1])
        ax[1].set_title(f'Boxplot of {cat} (Fig {i+11}-{j+1})', y=1.09,
**FONT)
        plt.show()
```


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

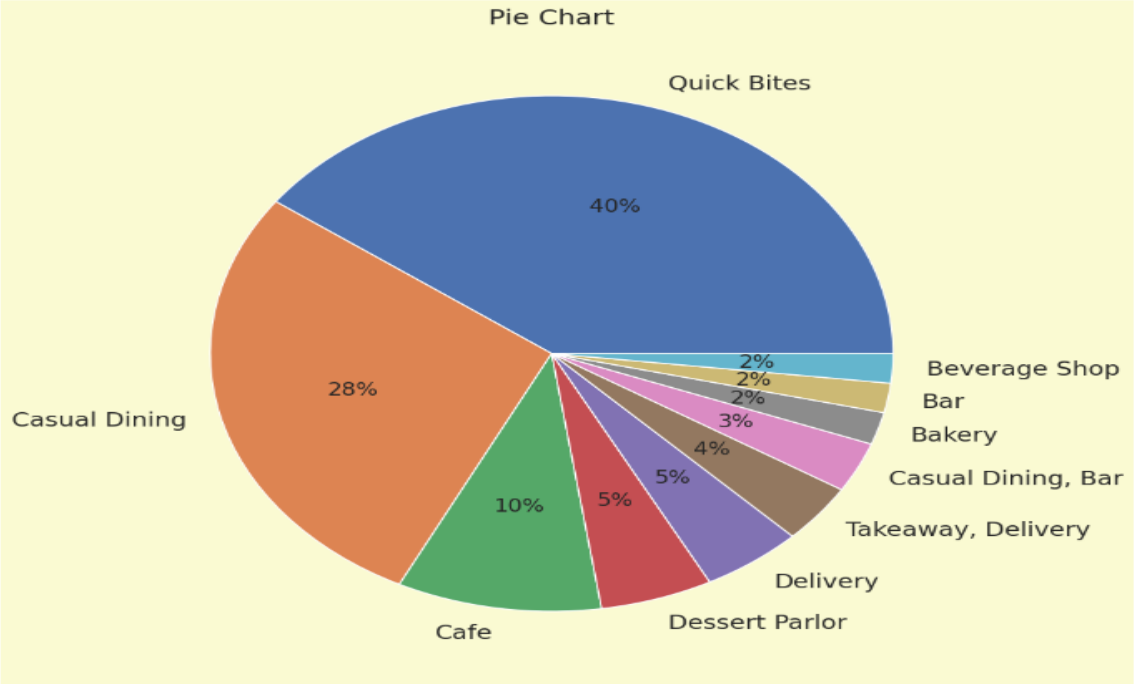



```
continuous_cols = ['Age','Experience','CCAvg','Mortgage']

for i, col in enumerate(continuous_cols):
    fig = px.scatter_3d(
        data_frame= df,
        x=df.Income,
        y=df[col],
        z=df['Personal Loan'],
        color=df['Personal Loan'].astype(str),
        color_discrete_map={'1':'orange', '0':'red'},
        template='ggplot2',
        hover_name='Age',
        # hover_data=
        opacity=0.6,
        # symbol='Transmission',
        # symbol_map=
        # log_x=True,
        # log_z=True,
        height=700,
        title=f'3D scatter of features based on Personal Loan (Fig {i+1})')
    fig.update_layout(
        title_text="Box Plot Styling Outliers",
        title_font=dict(color='orange', family='newtimeroman', size=25),
        title_x=0.45,
        paper_bgcolor='#145A32',
        # plot_bgcolor='#DAF7A6',
        font=dict(color='#DAF7A6', family='newtimeroman', size=16),
    )
    pio.show(fig)
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

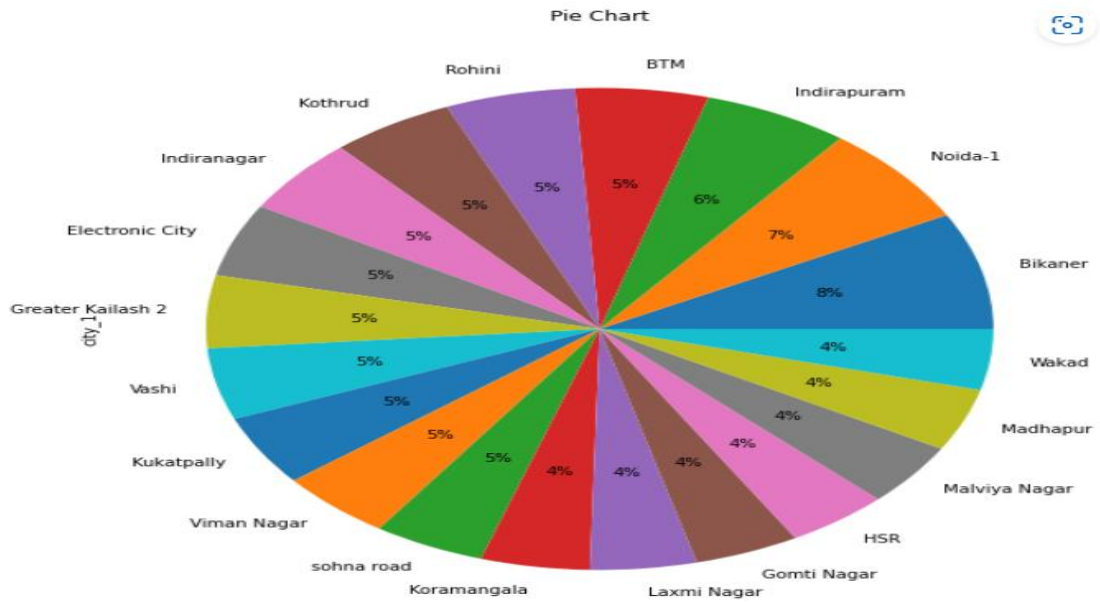



```
df["Type Of Restaurant"].value_counts()[:10].plot.pie(figsize = (10, 10),
                                                      autopct = '%1.0f%%')
plt.title("Pie Chart")
plt.xticks(rotation = 90)
plt.show()
```

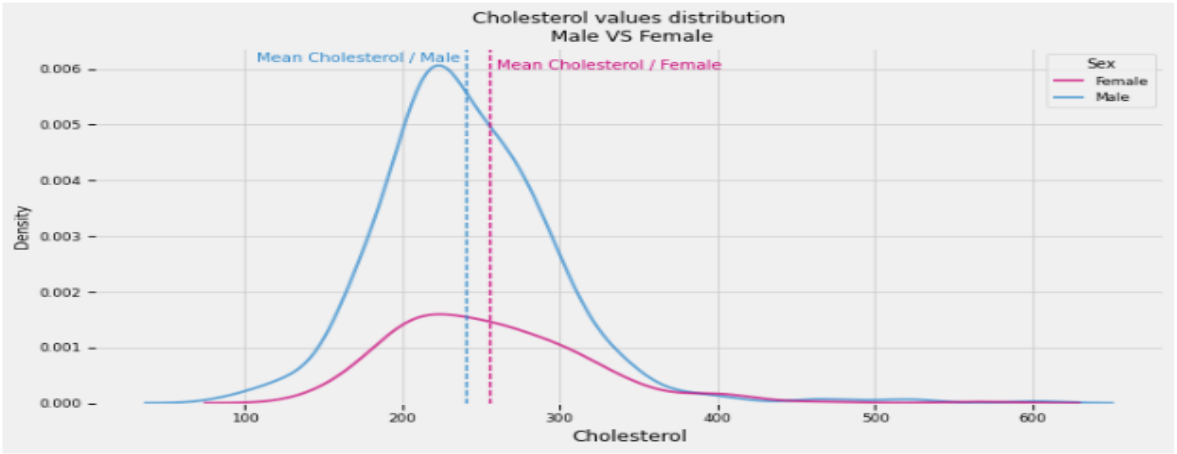



```
df['city_1'].value_counts().nlargest(n=20, keep='first').plot.pie(figsize = (10, 10),
                                                                  autopct = '%1.0f%%')
plt.title("Pie Chart")
plt.xticks(rotation = 90)
plt.show()
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

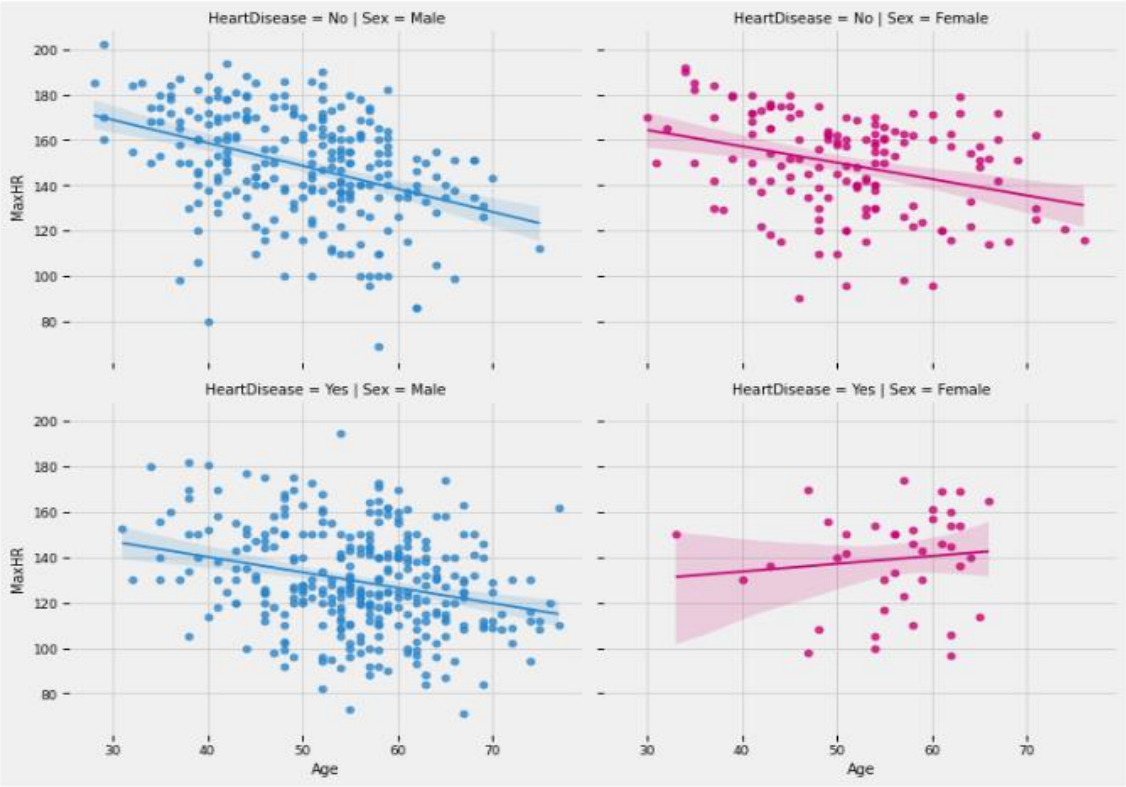



```
plt.figure(figsize=(10, 5))
sns.set_context("paper")

kdeplt = sns.kdeplot(
    data=heart_dft_chol_n0,
    x="Cholesterol",
    hue="Sex",
    palette=sex_color,
    alpha=0.7,
    lw=2,
)
kdeplt.set_title("Cholesterol values distribution\n Male VS Female", fontsize=12)
kdeplt.set_xlabel("Cholesterol", fontsize=12)
plt.axvline(x=Chol_mean_f, color="#c90076", ls="--", lw=1.3)
plt.axvline(x=Chol_mean_m, color="#2986cc", ls="--", lw=1.3)
plt.text(108, 0.00612, "Mean Cholesterol / Male", fontsize=10, color="#2986cc")
plt.text(260, 0.006, "Mean Cholesterol / Female", fontsize=10, color="#c90076")
plt.show()
```



Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

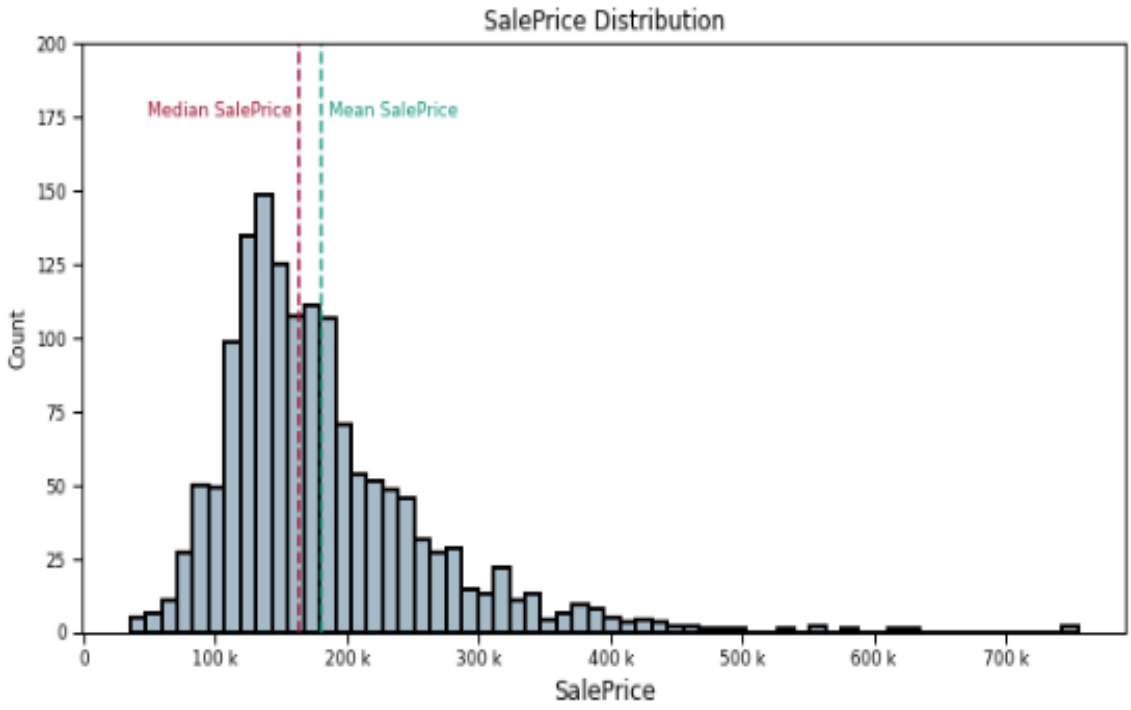
```
heart_df_fg = sns.FacetGrid(
    data=heart_dft_chol_n0,
    col="Sex",
    hue="Sex",
    row="HeartDisease",
    height=4,
    aspect=1.3,
    palette=sex_color,
    col_order=["Male", "Female"],
)
heart_df_fg.map_dataframe(sns.regplot, "Age", "MaxHR")
plt.show()
```




```
mean_SalePrice = usa_housing_df[["SalePrice"]].mean().squeeze()
median_SalePrice = usa_housing_df[["SalePrice"]].median().squeeze()

plt.figure(figsize=(10, 5))
sns.set_context("paper")

histplt = sns.histplot(
    data=usa_housing_df,
    x="SalePrice",
    color="#4f758f",
    bins=60,
```

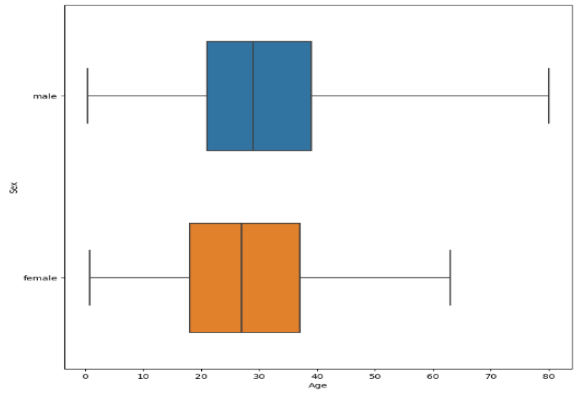
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

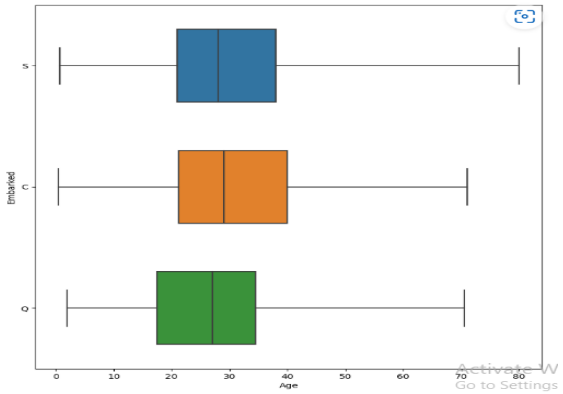
```
    alpha=0.5,
    lw=2,
)
histplt.set_title("SalePrice Distribution", fontsize=12)
histplt.set_xlabel("SalePrice", fontsize=12)

plt.axvline(x=mean_SalePrice, color="#14967f", ls="--", lw=1.5)
plt.axvline(x=median_SalePrice, color="#9b0f33", ls="--", lw=1.5)
plt.text(mean_SalePrice + 5000, 175, "Mean SalePrice", fontsize=9,
color="#14967f")
plt.text(
    median_SalePrice - 115000, 175, "Median SalePrice", fontsize=9,
color="#9b0f33"
)
histplt.xaxis.set_major_formatter(ticker.EngFormatter())
plt.ylim(0, 200)
plt.show()
```



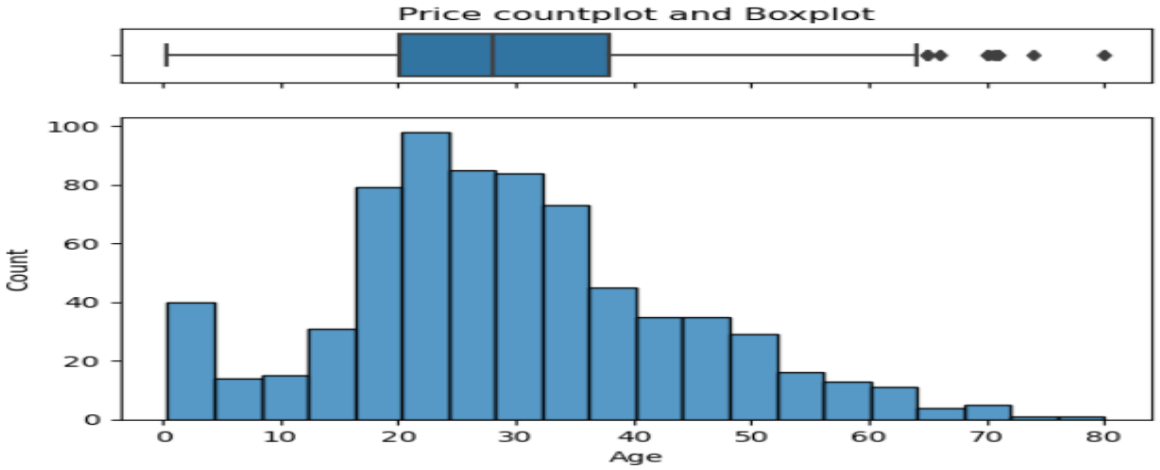

```
df2 = titanic[['Survived','Pclass','Sex','Embarked','SibSp','Parch',"Age"]]

fig, axes = plt.subplots(1, 2)
fig.set_figheight(10)
fig.set_figwidth(20)
for i,col in enumerate(df2.select_dtypes('object')):
    sns.boxplot(x="Age", y=col, data=df2,
                whis=[0, 100], width=.6,ax=axes[i])
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/



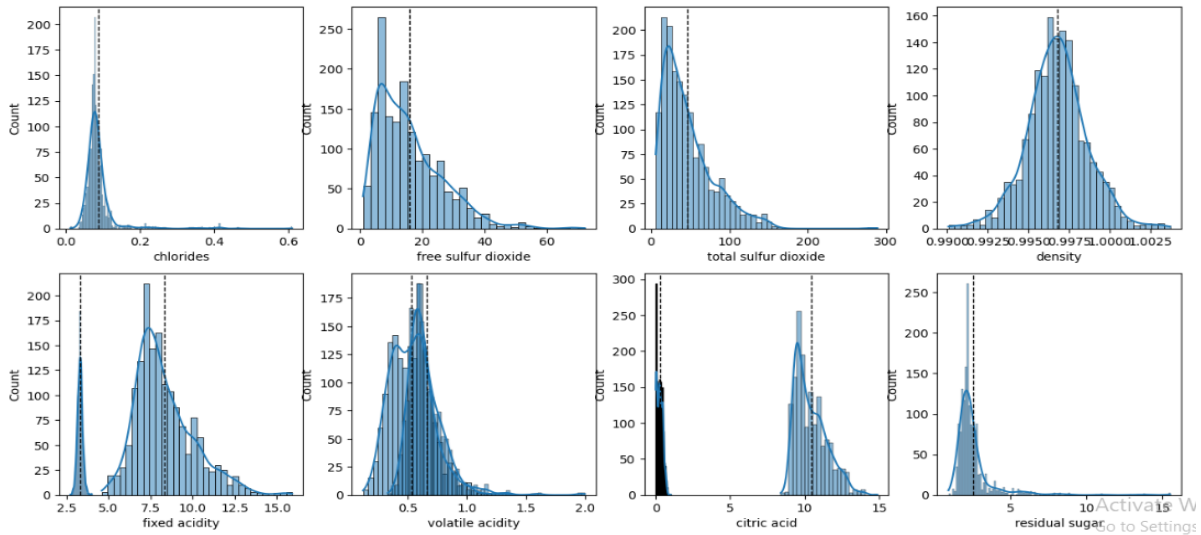



```
df2 = titanic[['Survived','Pclass','Sex','Embarked','SibSp','Parch',"Age"]]
#create the subplots
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True,
gridspec_kw={"height_ratios": (.15, .85)})
#title
ax_box.title.set_text('Price countplot and Boxplot')
# assigning a graph to each ax
sns.boxplot(df2["Age"], orient="h" ,ax=ax_box)
sns.histplot(data=df2, x="Age", ax=ax_hist)
# Remove x axis name for the boxplot
ax_box.set(xlabel='')
plt.show()
```

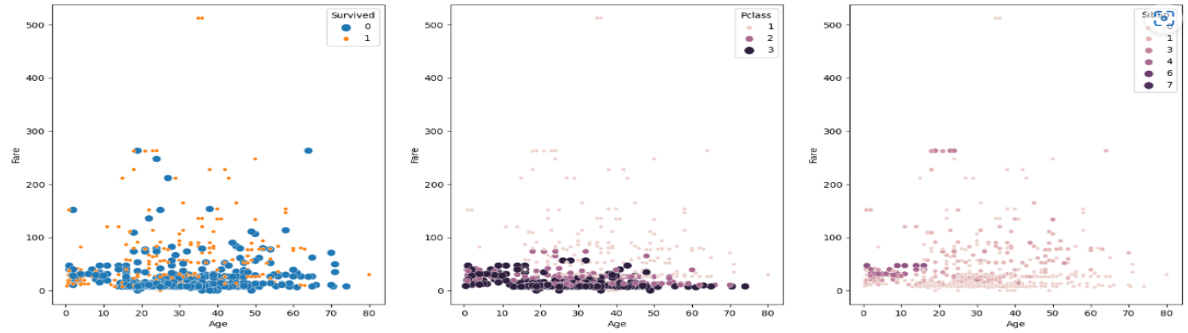



```
NUMERICAL = wine[['fixed acidity', 'volatile acidity', 'citric acid', 'residual sugar',
       'chlorides', 'free sulfur dioxide', 'total sulfur dioxide', 'density',
       'pH', 'sulphates', 'alcohol']]
fig, axes = plt.subplots(2, 4)
fig.set_figheight(12)
fig.set_figwidth(16)
for i,col in enumerate(NUMERICAL):
    sns.histplot(wine[col],ax=axes[(i // 4) -1 ,(i % 4)], kde = True)
    axes[(i // 4) -1 ,(i % 4)].axvline(wine[col].mean(), color='k', linestyle='dashed',
linewidth=1)
```

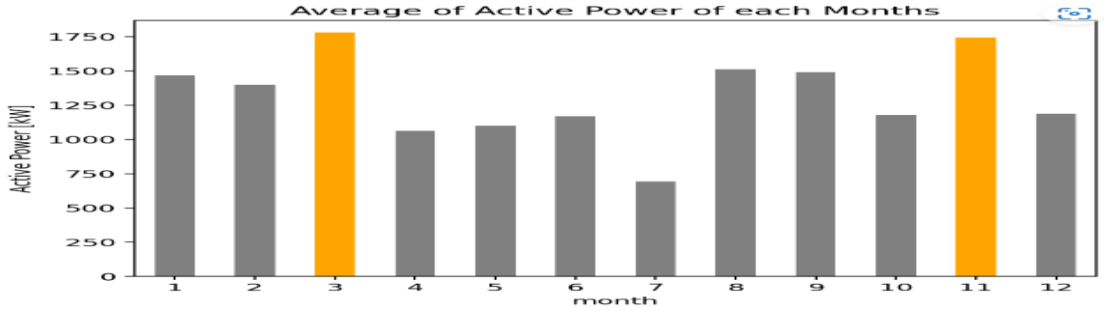
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/




```
fig, axes = plt.subplots(1, 3)
fig.set_figheight(7)
fig.set_figwidth(20)
sns.scatterplot(data=titanic, x="Age", y="Fare", hue="Survived", size="Survived",
ax=axes[0])
sns.scatterplot(data=titanic, x="Age", y="Fare", hue="Pclass", size="Pclass",
ax=axes[1])
sns.scatterplot(data=titanic, x="Age", y="Fare", hue="SibSp", size="SibSp",
ax=axes[2]);
```

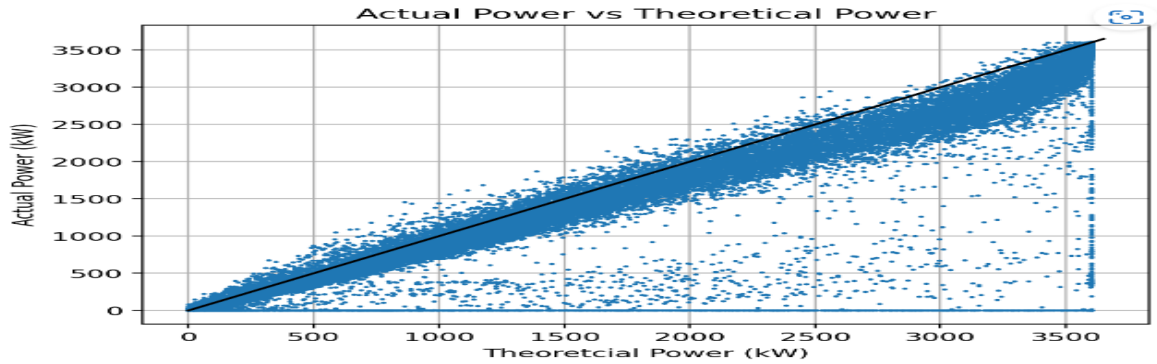



```
color = list(np.full(12, 'grey'))
color[2], color[10] = 'orange', 'orange'
df.groupby('month').mean().active_power.plot(kind='bar', title='Average of Active
Power of each Months', color=color, rot=0)
plt.ylabel('Active Power [kW]');
```

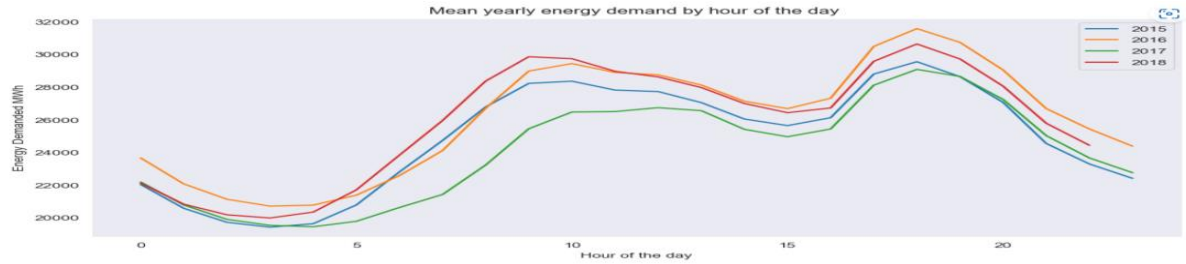


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
plt.title('Actual Power vs Theoretical Power')
plt.plot(df.theor_power, df.active_power, 'o', markersize= 1)
plt.grid('both')
plt.xlabel('Theoretcial Power (kW)')
plt.ylabel('Actual Power (kW)')
plt.plot([0,3650], [0,3650], '-', c= 'k')
plt.show()
```

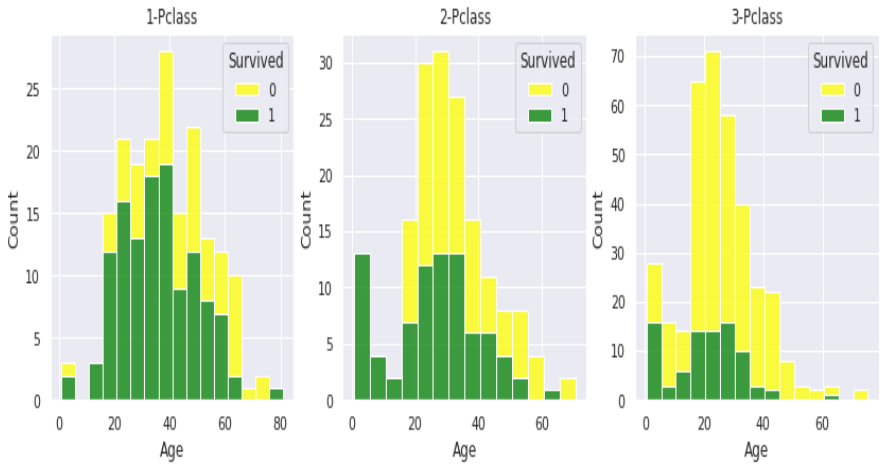



```
group_hours = df_demand['load'].groupby(pd.Grouper(freq='D', how='mean'))
fig, axs = plt.subplots(1,1, figsize=(12,5))
year_demands = pd.DataFrame()
for name, group in group_hours:
    year_demands[name.year] = pd.Series(group.values)
year_demands.plot(ax=axs)
axs.set_xlabel('Hour of the day')
axs.set_ylabel('Energy Demanded MWh')
axs.set_title('Mean yearly energy demand by hour of the day ');
```

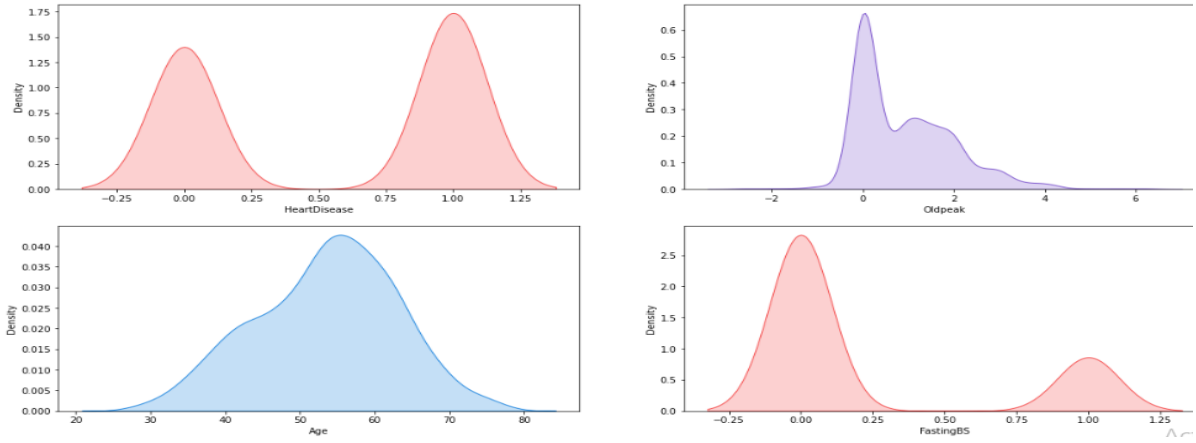



```
plot , ax = plt.subplots(1 , 3 , figsize=(14,4))
sns.histplot(data = train_data.loc[train_data["Pclass"]==1] , x = "Age" , hue = "Surv
ived",binwidth=5,ax = ax[0],palette = sns.color_palette(["yellow" , "green"]),multip
le = "stack").set_title("1-Pclass")
sns.histplot(data = train_data.loc[train_data["Pclass"]==2] , x = "Age" , hue = "Surv
ived",binwidth=5,ax = ax[1],palette = sns.color_palette(["yellow" , "green"]),multip
le = "stack").set_title("2-Pclass")
sns.histplot(data = train_data.loc[train_data["Pclass"]==3] , x = "Age" , hue = "Surv
ived",binwidth=5,ax = ax[2],palette = sns.color_palette(["yellow" , "green"]),multip
le = "stack").set_title("3-Pclass")
plt.show()
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

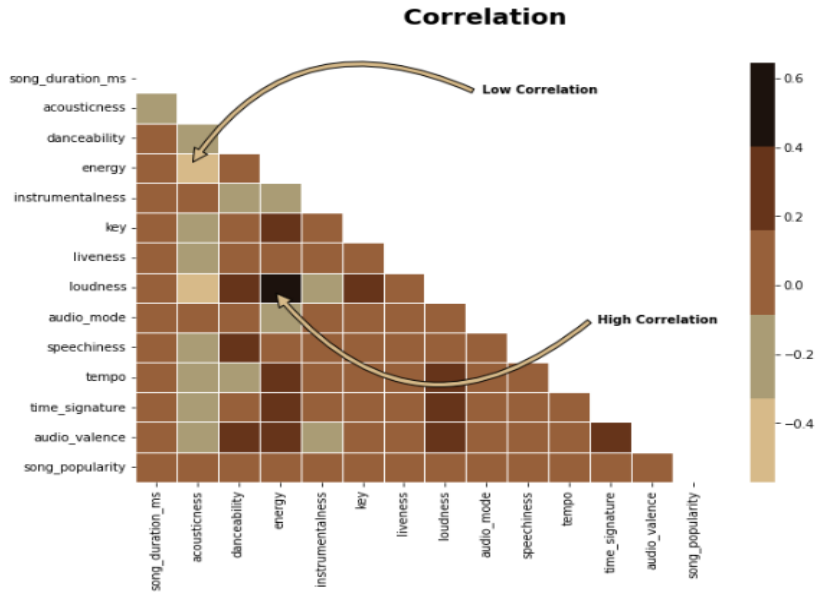



```
#Plotting the distributions of the numerical variables
color_plot =
['#de972c','#74c91e','#1681de','#e069f5','#f54545','#f0ea46','#7950cc']

fig,ax = plt.subplots(4,2,figsize=(20,20))
sns.kdeplot(df['HeartDisease'],color=np.random.choice(color_plot), ax=ax[0][0],
shade=True)
sns.kdeplot(df['Oldpeak'],color=np.random.choice(color_plot), ax=ax[0][1],
shade=True)
sns.kdeplot(df['Age'],color=np.random.choice(color_plot), ax=ax[1][0],
shade=True)
sns.kdeplot(df['FastingBS'],color=np.random.choice(color_plot), ax=ax[1][1],
shade=True)
sns.kdeplot(df['RestingBP'],color=np.random.choice(color_plot),
ax=ax[2][0],shade=True)
sns.kdeplot(df['Cholesterol'],color=np.random.choice(color_plot), ax=ax[2][1],
shade=True)
sns.kdeplot(df['MaxHR'],color=np.random.choice(color_plot),
ax=ax[3][0],shade=True)
fig.delaxes(ax[3][1])
```



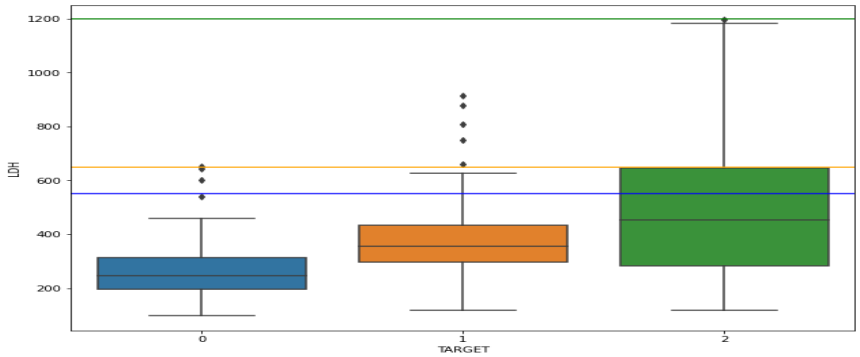
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
hm= df.drop('id', axis =1)
mask = np.zeros_like(hm.corr(), dtype=np.bool)
mask[np.triu_indices_from(mask)]= True

plt.suptitle('Correlation', size = 20, weight='bold')

ax = sns.heatmap(hm.corr(), linewidths = 0.9, linecolor = 'white', cbar = True,mask
=mask, cmap=heatmap)

ax.annotate('Low Correlation',
                fontsize=10,fontweight='bold',
                xy=(1.3, 3.5), xycoords='data',
                xytext=(0.6, 0.95), textcoords='axes fraction',
                arrowprops=dict(
                facecolor=heatmap[0], shrink=0.025,
                connectionstyle='arc3, rad=0.50'),
                horizontalalignment='left', verticalalignment='top'
)

ax.annotate('High Correlation',
                fontsize=10,fontweight='bold',
                xy=(3.3, 7.5), xycoords='data',
                xytext=(0.8, 0.4), textcoords='axes fraction',
                arrowprops=dict(
                facecolor=heatmap[0], shrink=0.025,
                connectionstyle='arc3, rad=-0.6'),
                horizontalalignment='left', verticalalignment='top'
)
plt.show()
```

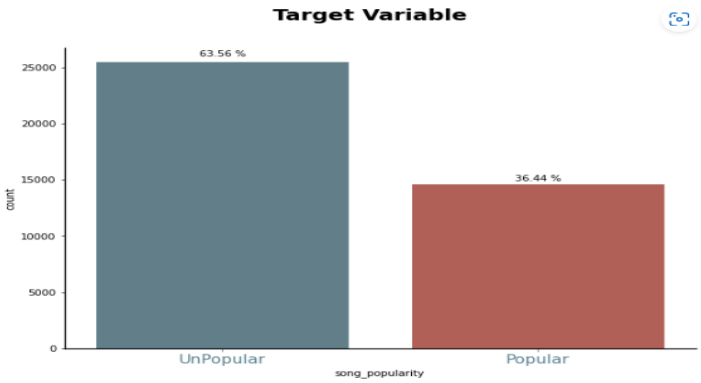


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
fig = plt.figure( figsize=(8, 6))
ax = fig.add_axes([0,0,1,1])
sns.boxplot(ax=ax, data=df, x='TARGET', y='LDH')#,flierprops=dict(marker='o', markersize=6),fliersize=2)

ax.axhline(y=550,color='b')
ax.axhline(y=650,color='orange')
ax.axhline(y=1200,color='g')
```

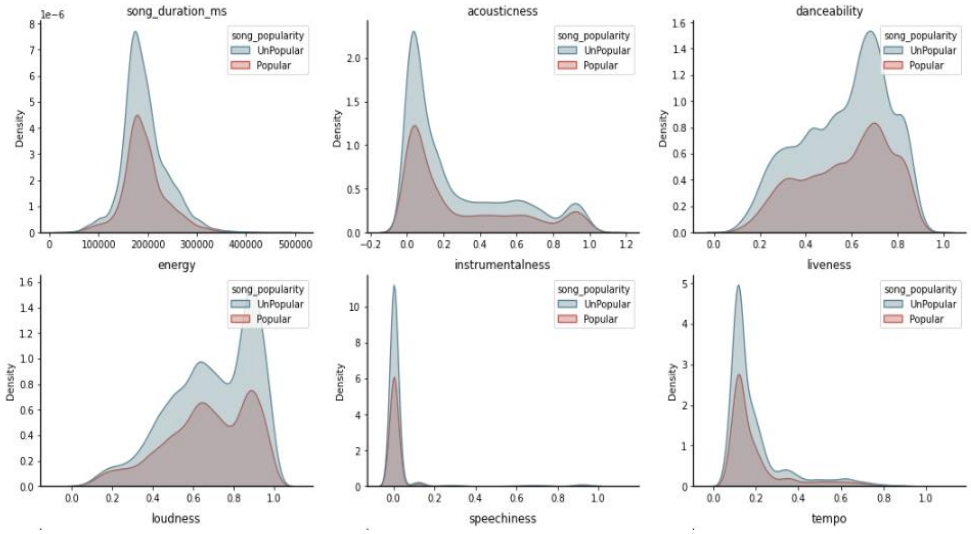



```
plt.suptitle('Target Variable', size = 20, weight='bold')

song_popularity = df['song_popularity'].map({0:'UnPopular', 1:'Popular'})

a = sns.countplot(data = df, x =song_popularity,palette=theme)
plt.tick_params(axis="x", colors=theme[0],labelsize=15)

for p in a.patches:
    width = p.get_width()
    height = p.get_height()
    x, y = p.get_xy()
    a.annotate(f'{height/df.shape[0]*100} %', (x + width/2, y + height*1.02), ha='center')

plt.show()
```



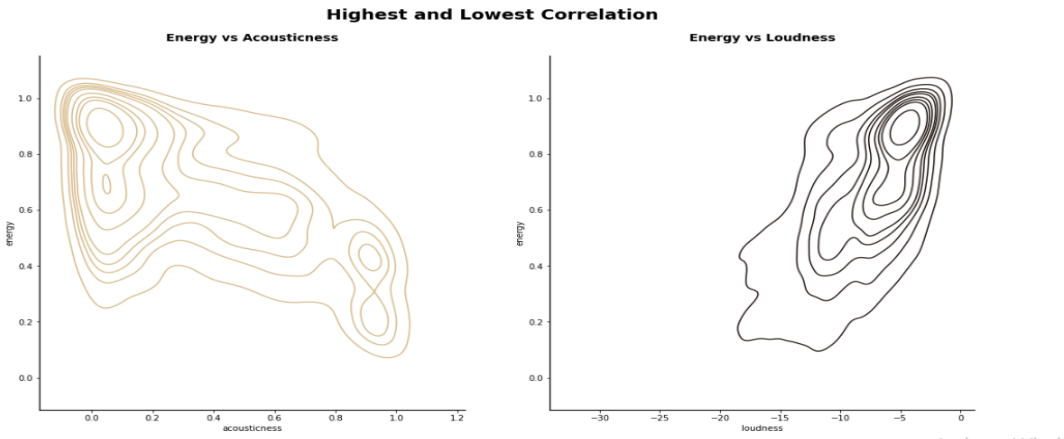
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
cont = ['song_duration_ms', 'acousticness', 'danceability', 'energy',
       'instrumentalness', 'liveness', 'loudness',
       'speechiness', 'tempo', 'audio_valence']
cat = [ 'key', 'audio_mode', 'time_signature']

a = 4  # number of rows
b = 3  # number of columns
c = 1  # initialize plot counter

plt.figure(figsize= (18,18))

for i in cont:
        plt.suptitle('Distribution of Features', size = 20, weight='bold')
        plt.subplot(a, b, c)
        A=sns.kdeplot(data= df, x=i,hue=song_popularity,palette=theme[:-2], linewidt
h = 1.3,shade=True, alpha=0.35)
        plt.title(i)
        plt.xlabel(" ")
        c = c + 1
```



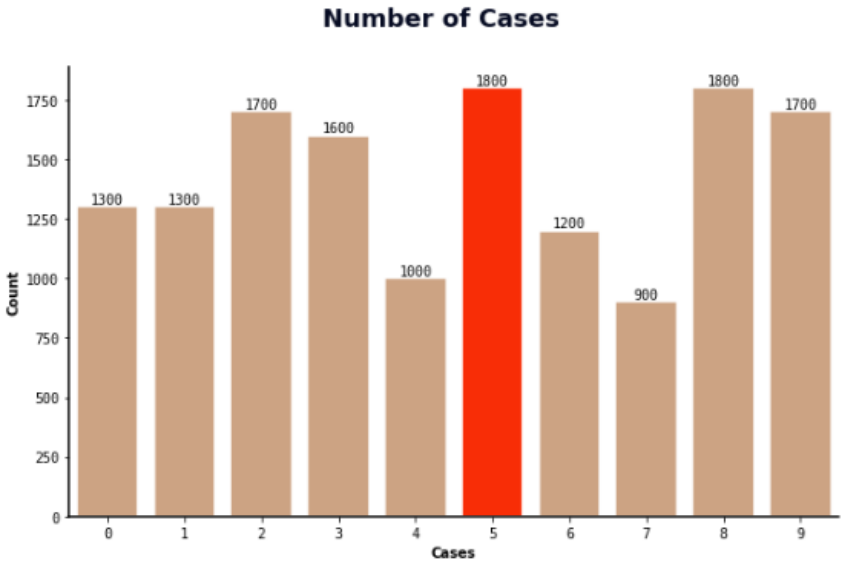

```
#plotting
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(18, 9))
fig.suptitle(' Highest and Lowest Correlation ', size = 20, weight='bold')
axs = [ax1, ax2]

#kdeplot
sns.kdeplot(data=df, y='energy', x='acousticness', ax=ax1, color=heatmap[0])
ax1.set_title('Energy vs Acousticness', size = 14, weight='bold', pad=20)

#kdeplot
sns.kdeplot(data=df, y='energy', x='loudness', ax=ax2, color=heatmap[4])
ax2.set_title('Energy vs Loudness', size = 14, weight='bold', pad=20);
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/




```
#Parameters for Plots
plt.rcParams['figure.figsize'] = (10,6)
plt.rcParams['axes.edgecolor'] = 'black'
plt.rcParams['axes.linewidth'] = 1.5
plt.rcParams['figure.frameon'] = True
plt.rcParams['axes.spines.top'] = False
plt.rcParams['axes.spines.right'] = False
plt.rcParams["font.family"] = "monospace";

#Colors for charts
colors = ["#e9d9c8","#cca383","#070c23","#f82d06","#e8c195","#cd7551","#a499
95","#a3a49c","#6c7470"]
sns.palplot(sns.color_palette(colors))

#plot
A = sns.countplot(train_df['case_num'],
            color=colors[1],
            edgecolor='white',
            linewidth=1.5,
            saturation=1.5)

#Patch
patch_h = []
for patch in A.patches:
    reading = patch.get_height()
    patch_h.append(reading)

idx_tallest = np.argmax(patch_h)
A.patches[idx_tallest].set_facecolor(colors[3])

#Lables
plt.ylabel('Count', weight='semibold', fontname = 'Georgia')
plt.xlabel('Cases', weight='semibold', fontname = 'Georgia')
plt.suptitle('Number of Cases', fontname = 'Georgia', weight='bold', size = 18, color = colors
[2])
A.bar_label(A.containers[0], label_type='edge')
plt.show()
```

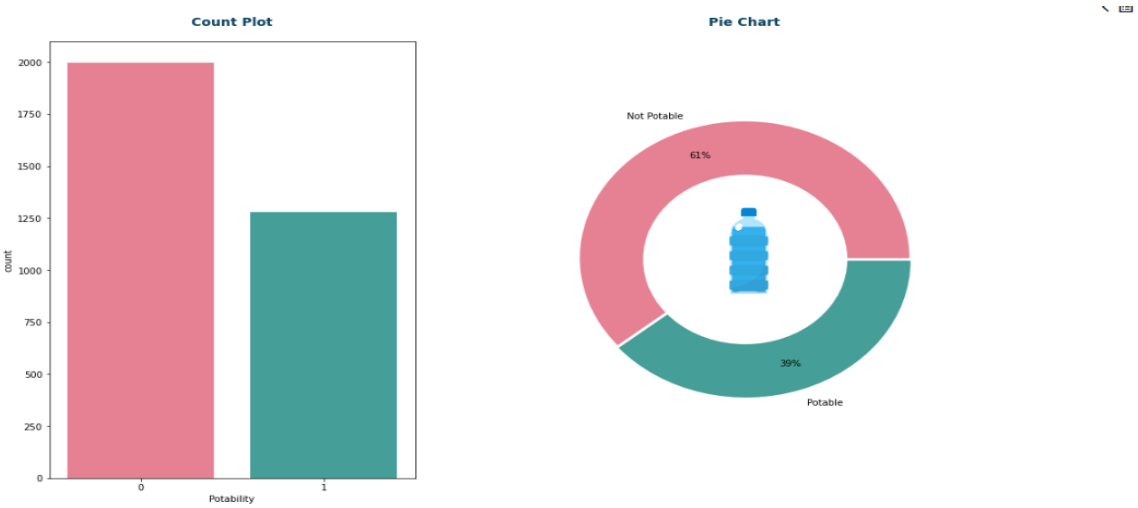
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/



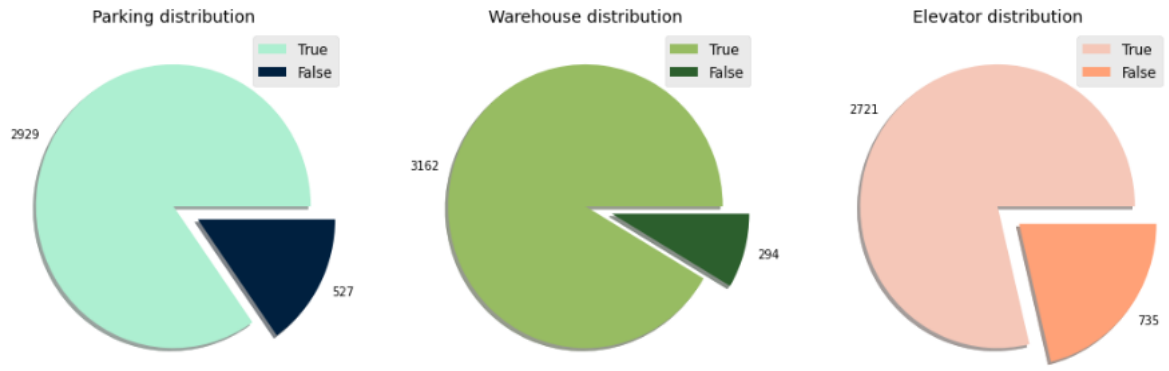

```
import matplotlib as mlb
import matplotlib.image as mpimg
from matplotlib.offsetbox import AnnotationBbox, OffsetImage

#plotting
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(16, 11))
fig.suptitle(' Potablity of Water Quality ', size = 26, color = theme[3], weight='bold')
axs = [ax1, ax2]

#Count-Plot
sns.countplot(water_df['Potability'], ax=ax1, palette='husl')
ax1.set_title('Count Plot', size = 14, color = theme[3], weight='bold', pad=20)

#Data-2
names = ["Not Potable", "Potable"]
values = water_df['Potability'].value_counts()
colors = ["#E68193","#459E97"]
explode = (0.01, 0.01)

#Doughnut-chart
ax2.pie(x= values,labels =names, colors=colors,autopct='%1.0f%%', pctdistance=
0.8,explode=explode)

#draw-circle
centre_circle = plt.Circle((0,0),0.62,fc='white')
ax2.add_artist(centre_circle)
ax2.axis('equal')

ax2.set_title('Pie Chart', size = 14, color = theme[3], weight='bold', pad=20)

#Image
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
path = mpimg.imread('../input/water/water bottle.png')
imagebox = OffsetImage(path , zoom=0.3)
xy = (0.5, 0.7)
ab = AnnotationBbox(imagebox, xy, frameon=False, pad=1, xybox=(0.02, 0.05))
ax2.add_artist(ab)

plt.subplots_adjust(left=None, bottom=None, right=None, top=0.8, wspace=0.4, hs
pace=None);
```

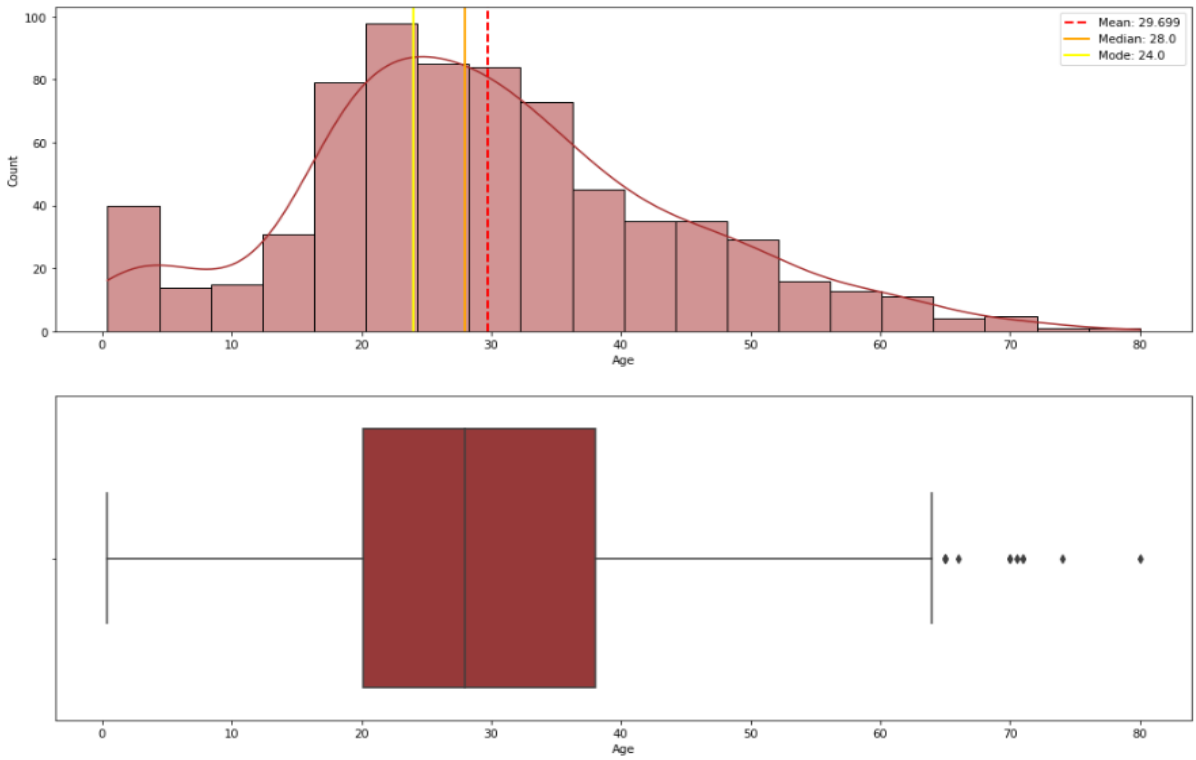



```
fig, ax = plt.subplots(ncols=3, figsize=(18,6))

colors = [['#ADEFD1FF', '#00203FFF'], ['#97BC62FF', '#2C5F2D'], ['#F5C7B8FF', '#F
FA177FF']]
explode = [0, 0.2]
columns = ['Parking', 'Warehouse', 'Elevator']
for i in range(3):
    data = df[columns[i]].value_counts()
    ax[i].pie(data, labels=data.values, explode=explode, colors=colors[i], shadow
=True)
    ax[i].legend(labels=data.index, fontsize='large')
    ax[i].set_title('{} distribution'.format(columns[i]))
```



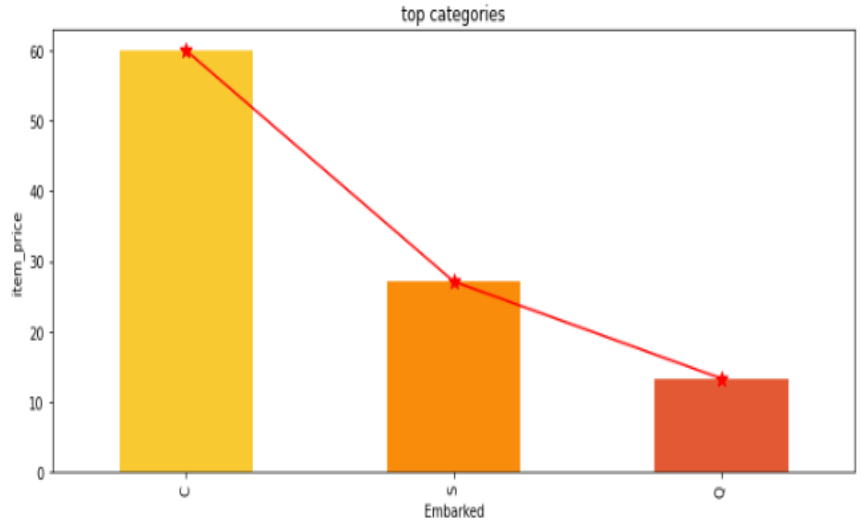
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
def plot_hist(feature):
    fig, ax = plt.subplots(2, 1, figsize=(17, 12))
    sns.histplot(data = titanic[feature], kde = True, ax =
ax[0],color="Brown")
    ax[0].axvline(x = titanic[feature].mean(), color = 'r', linestyle = '--',
linewidth = 2, label = 'Mean: {}'.format(round(titanic[feature].mean(), 3)))
    ax[0].axvline(x = titanic[feature].median(), color = 'orange', linewidth =
2, label = 'Median: {}'.format(round(titanic[feature].median(), 3)))
    ax[0].axvline(x = statistics.mode(titanic[feature]), color = 'yellow',
linewidth = 2, label = 'Mode: {}'.format(statistics.mode(titanic[feature])))
    ax[0].legend()

    sns.boxplot(x = titanic[feature], ax = ax[1],color="Brown")
    plt.show()
plot_hist('Age')
```



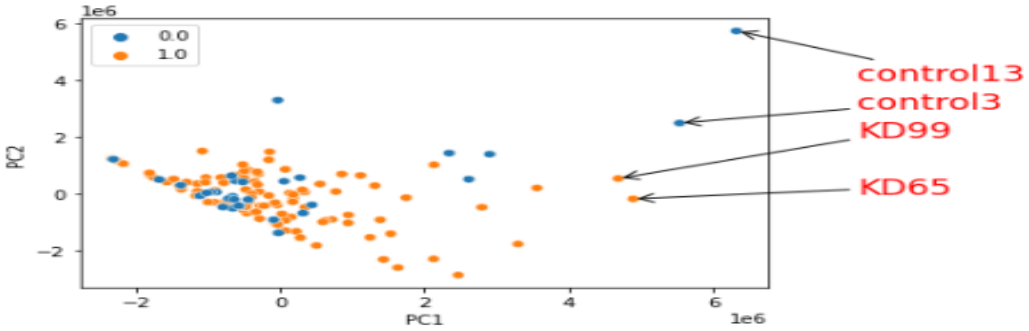

```
plt.figure(figsize=(12,5))
plt.title('top categories')
plt.ylabel('item_price')
titanic.groupby('Embarked')['Fare'].mean().sort_values(ascending=Fa
lse)[0:15].plot(kind='line', marker='*', color='red', ms=10)
titanic.groupby('Embarked')['Fare'].mean().sort_values(ascending=Fa
lse)[0:15].plot(kind='bar',color=sns.color_palette("inferno_r", 7))
plt.show()
```


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/




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

sns.scatterplot(x=df.iloc[:,0], y=df.iloc[:,1], hue=y)
plt.annotate("KD65", (df.iloc[64,0], df.iloc[64,1]), (8*1e6, 1), arrowprops=dict(arrowstyle="->"), fontsize="xx-large",c='red')
plt.annotate("KD99", (df.iloc[98,0], df.iloc[98,1]), (8*1e6, 2*1e6), arrowprops=dict(arrowstyle="->"), fontsize="xx-large",c='red')
plt.annotate("control3", (df.iloc[107,0], df.iloc[107,1]), (8*1e6, 3*1e6), arrowprops=dict(arrowstyle="->"), fontsize="xx-large",c='red')
plt.annotate("control13", (df.iloc[117,0], df.iloc[117,1]), (8*1e6, 4*1e6), arrowprops=dict(arrowstyle="->"), fontsize="xx-large",c='red')
```

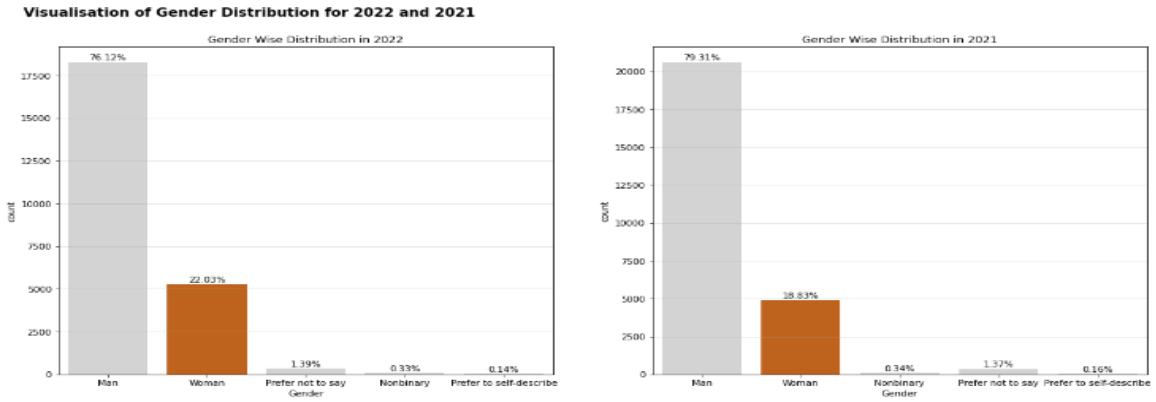
```
Text(8000000.0, 4000000.0, 'control13')
```




```
l = df_current['Q3'].value_counts(normalize=True).mul(100).tolist()[1]-df_old['Q2'].value_counts(normalize=True).mul(100).values.tolist()[1]

print(5*'\n',"\033[1;32m Increase in Woman is only\033[1;32m",round(l, 2),'%\033[1;32m Over Last Year\033[1;32m',5*'\n')

fig, ax = plt.subplots(1, 2, figsize=(20,8))
fig.text(0.1, 0.95, "Visualisation of Gender Distribution for 2022 and 2021", fontsize=15, fontweight='bold')
```

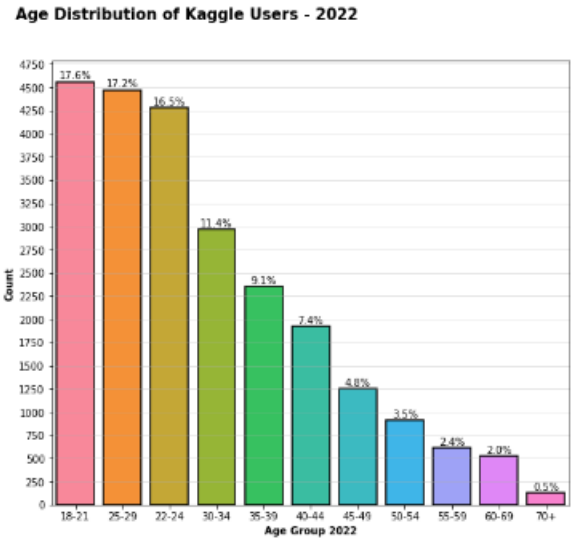
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

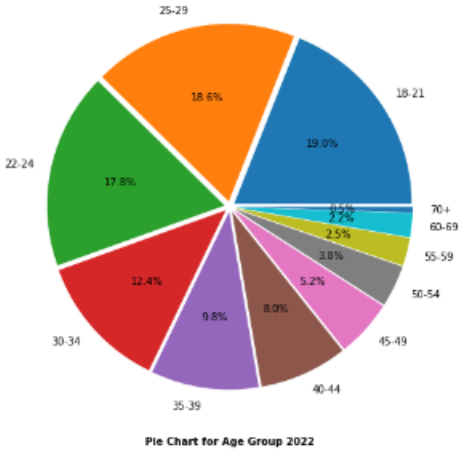
```
sns.countplot(x='Q3', data=df_current,palette="Dark2", ax=ax[0]); #Current Year
sns.countplot(x='Q2', data=df_old,palette="Dark2",ax=ax[1]); #Last Year
for i, ax in enumerate(ax.flatten()):
    ax.grid(axis='y', linestyle='-', alpha=0.4)
    if i==0:t=shape;year = 2022
    else:t=shape_21;year =2021
    for p in ax.patches:
        percentage = f'{100 * p.get_height() / t:.2f}%\n'
        ax.annotate(percentage, (p.get_x() + p.get_width() / 2,p.get_height()), ha='cen
ter', va='center')
        ax.set_xlabel('Gender');ax.set_title("Gender Wise Distribution in "+ str(year))
        if not(0.5 <= p.get_x() < 1.5):
            p.set_facecolor('lightgrey')
plt.show()
```




```
fig, ax = plt.subplots(1,2, figsize=(20,8))
fig.text(0.1, 0.95, "Age Distribution of Kaggle Users - 2022", fontsize=15, font
weight='bold')
sns.barplot(x=df_current['Q2'].value_counts().index, y=df_current['Q2'].value_c
ounts().values, ax=ax[0],
            edgecolor='black', linewidth=1.5, saturation=1.5)
ax[0].yaxis.set_major_locator(MaxNLocator(nbins=20));ax[0].grid(axis='y', line
style='-', alpha=0.4)
ax[0].set_ylabel('Count', weight='semibold')
ax[0].set_xlabel('Age Group 2022', weight='semibold')
ax[1].set_xlabel('Pie Chart for Age Group 2022', weight='semibold')
for p in ax[0].patches:
        percentage = f'{100 * p.get_height() / t:.1f}%\n'
        ax[0].annotate(percentage, (p.get_x() + p.get_width() / 2,p.get_height()), h
a='center', va='center')

ax[1].pie(df_current['Q2'].value_counts(), labels = df_current['Q2'].value_count
s().index,  autopct='%1.1f%%',
        explode=[0.03 for i in df_current['Q2'].value_counts().index])
plt.show()
```

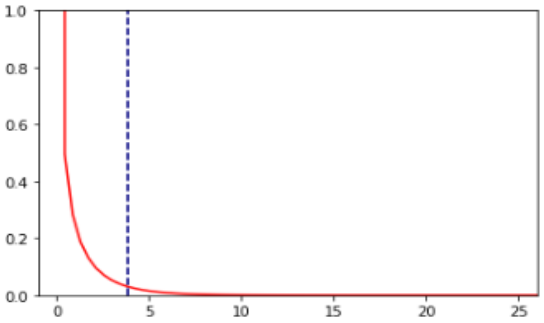

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/






```
fig, ax = plt.subplots(1, 1)

plt.xlim(-1,26)
plt.ylim(0,1)
x = np.linspace(f.ppf(0.0000000001, dfn, dfd),f.ppf(0.9999999999, dfn, dfd), 10
0)
ax.plot(x, f.pdf(x, dfn, dfd), 'r-')
ax.axvline(f.ppf(0.95, dfn, dfd), ls = "--", color = "navy")
print('upper 5%:', f.ppf(0.95, dfn, dfd))
```

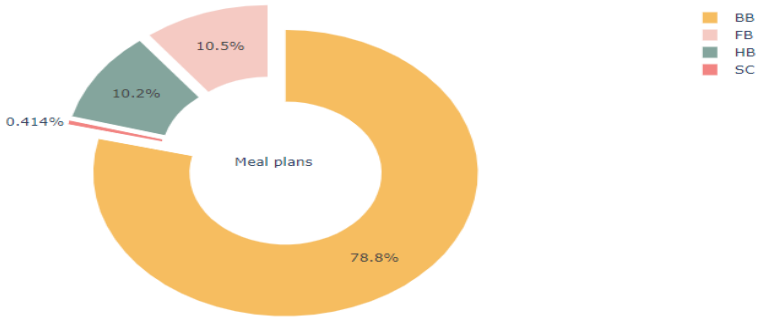
```
upper 5%: 3.8426563592313365
```



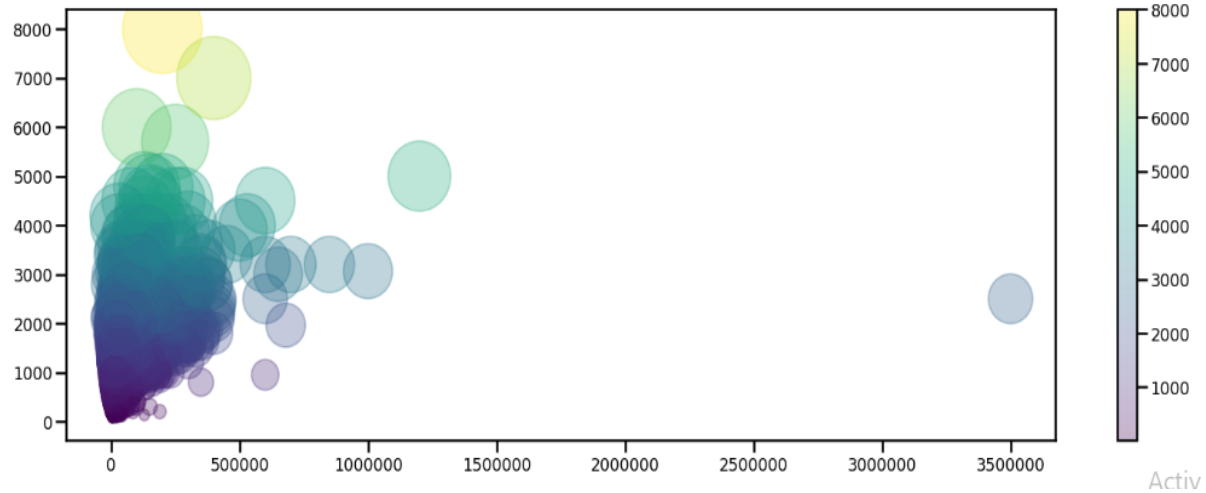

```
import plotly.graph_objects as go

labels = confirmed_bookings['meal'].unique()
values = confirmed_bookings['meal'].value_counts()
palette = ["#f6bd60", "#f5cac3", "#84a59d", "#f28482"]

fig = go.Figure(data=[go.Pie(labels = labels,
                             values = values,
                             hole=.5,
                             title = 'Meal plans',
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
                    legendgroup = True,
                    pull = [0.1, 0.1, 0.1, 0.1]
                   )
            ]
        )

fig.update_traces(marker = dict(colors = palette))
fig.show()
```



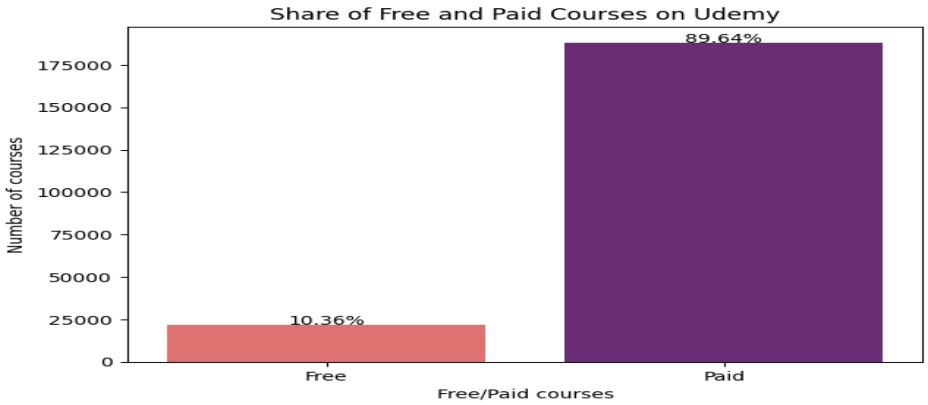

```
x = rent_df["Rent"]
y = rent_df["Size"]
colors = rent_df["Size"]
sizes = rent_df["Size"]

plt.figure(figsize = (25, 8))
plt.ticklabel_format(style = 'plain')
plt.scatter(x, y, c = colors, s = sizes, alpha = 0.3, cmap = 'viridis')
plt.colorbar();
```



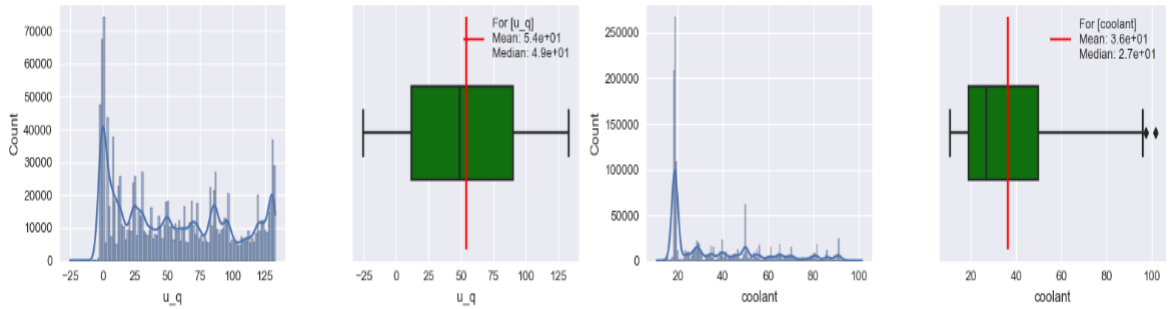

```
# Free or Paid Courses - Countplot
fig, ax = plt.subplots(figsize=(7,5), dpi=100)
ax = sns.countplot(data=courses, x='is_paid', palette='magma_r')
ax.set_xticklabels(labels=['Free', 'Paid'])
ax.set_xlabel("Free/Paid courses")
ax.set_ylabel("Number of courses")
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
ax.set_title("Share of Free and Paid Courses on Udemy")
percentage = round(courses['is_paid'].value_counts() * 100 /len(courses), 2)
patches = ax.patches
for i in range(len(patches)):
   x = patches[i].get_x() + patches[i].get_width()/2
   y = patches[i].get_height()+.05
   ax.annotate('{:.2f}%'.format(percentage[i]), (x, y), ha='center')
```



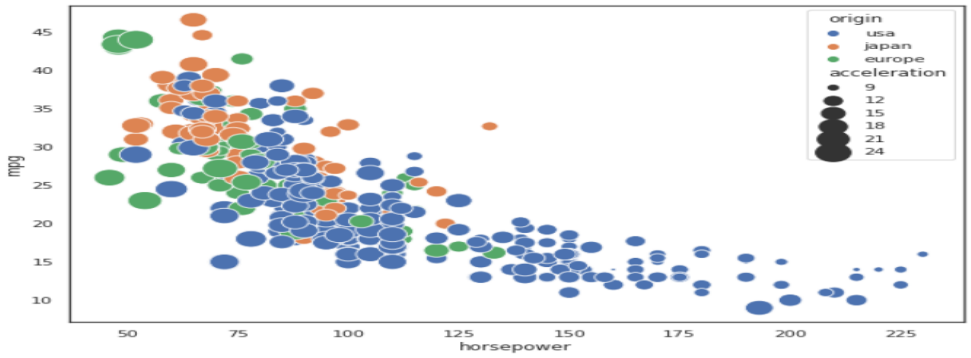

```
df_cpy = df.drop(['profile_id'], axis=1)
flierprops = dict(markerfacecolor='g', color='g', alpha=0.5)
n_cols = 4
n_rows = int(np.ceil(df_cpy.shape[-1]*2 / n_cols))
fig, axes = plt.subplots(n_rows, n_cols, figsize=(4 * n_cols, 3 * n_rows))
for i, (col) in enumerate(list(df_cpy.columns)):
    mean = df_cpy[col].mean()
    median = df_cpy[col].median()
    sns.histplot(df_cpy[col], ax=axes.flatten()[2*i], kde=True)
    sns.boxplot(x=df_cpy[col], orient='h', ax=axes.flatten()[2*i+1], color='g')
    axes.flatten()[2*i+1].vlines(mean, ymin = -1, ymax = 1, color='r',
label=f"For [{col}]\nMean: {mean:.2}\nMedian: {median:.2}")
    axes.flatten()[2*i+1].legend()
    if i % n_cols == 0:
        ax.set_ylabel('Frequency')
    else:
        ax.set_ylabel('')
plt.tight_layout()
```



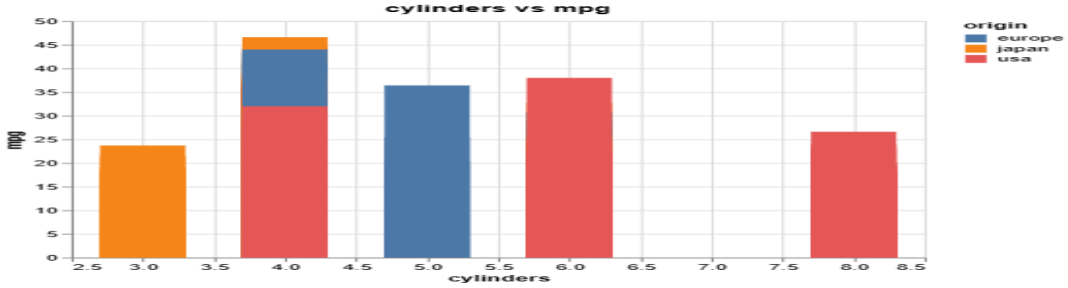

```
sns.set(rc={'figure.figsize':(10,7)})
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
sns.set_style("white")
sns.scatterplot(data=df, x="horsepower", y="mpg", size="acceleration",
hue='origin',legend=True, sizes=(10, 500))
```



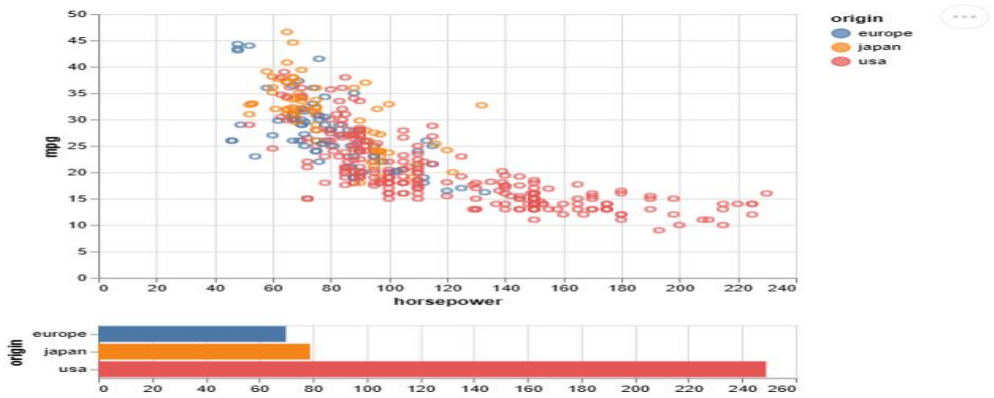

```
import altair as alt
plot=alt.Chart(df).mark_bar(size=40).encode(
    alt.X('cylinders'),
    alt.Y('mpg'),
    alt.Color('origin')
)
plot.properties(title='cylinders vs mpg')
```



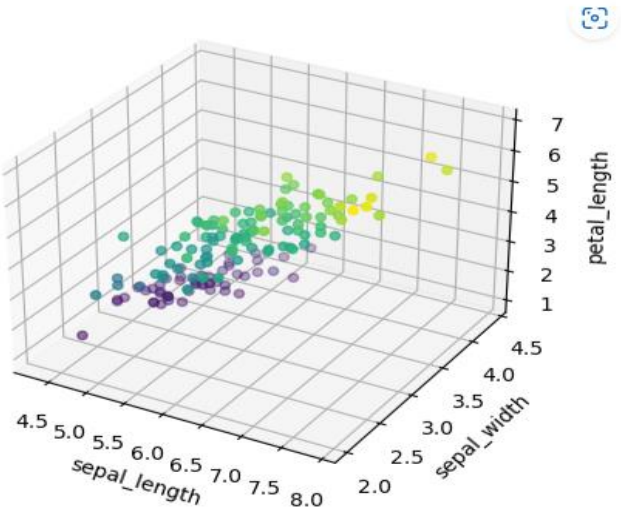

```
import altair as alt
select = alt.selection(type='interval')
values = alt.Chart(df).mark_point().encode(
    x='horsepower:Q',
    y='mpg:Q',
    color=alt.condition(select, 'origin:N', alt.value('lightgray'))
).add_selection(
    select
)
bars = alt.Chart(df).mark_bar().encode(
    y='origin:N',
    color='origin:N',
    x='count(origin):Q'
).transform_filter(
    select
)
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

values & bars

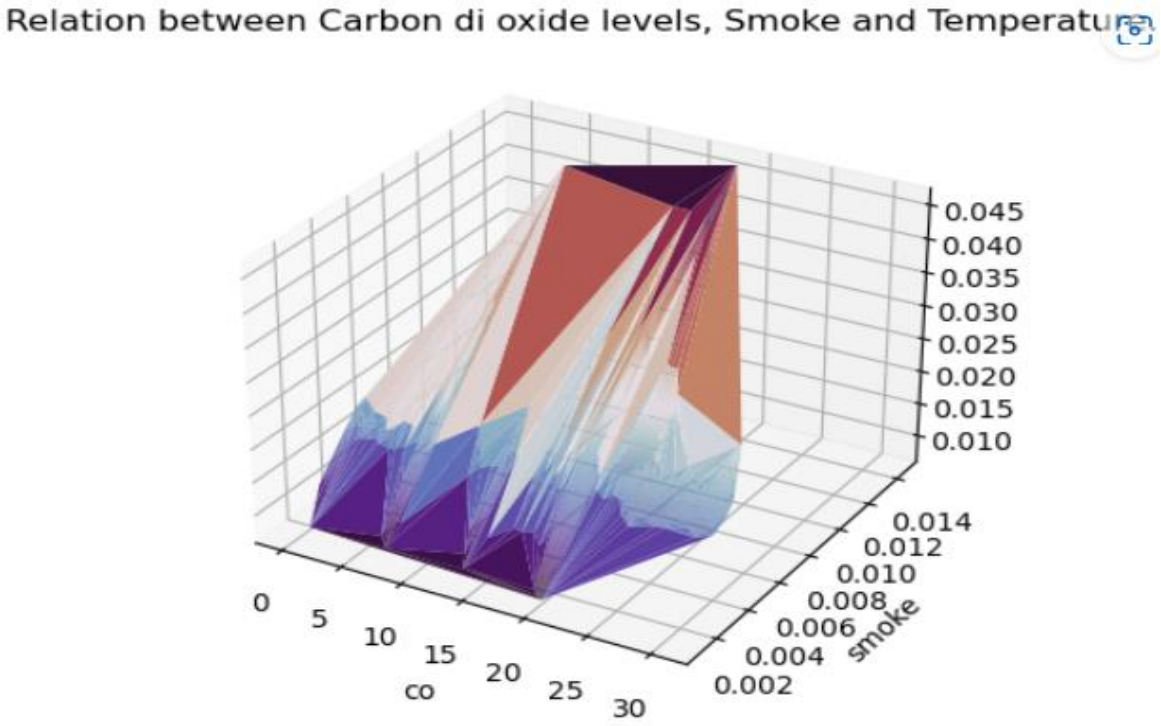



```
import pandas as pd
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
iris = pd.read_csv("Iris.csv")
# Create a figure and axes for the 3D plot
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
# Scatter plot the data
ax.scatter(iris["sepal_length"], iris["sepal_width"], iris["petal_length"],
c=iris["petal_length"], cmap='viridis')
# Add labels to the axes
ax.set_xlabel("sepal_length")
ax.set_ylabel("sepal_width")
ax.set_zlabel("petal_length")
# Show the plot
plt.show()
```



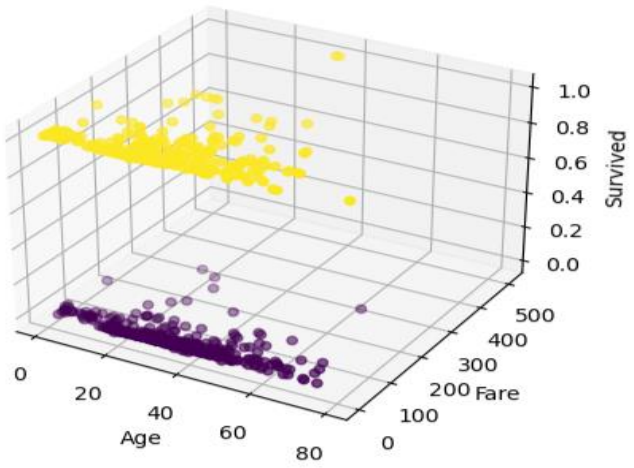
Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
fig = plt.figure()
ax = fig.gca(projection='3d')
ax.plot_trisurf(data['temp'], data['co'], data['smoke'], cmap =
plt.cm.twilight_shifted)
plt.title('Relation between Carbon di oxide levels, Smoke and
Temperature.')
plt.xlabel('co')
plt.ylabel('smoke')
plt.show()
```



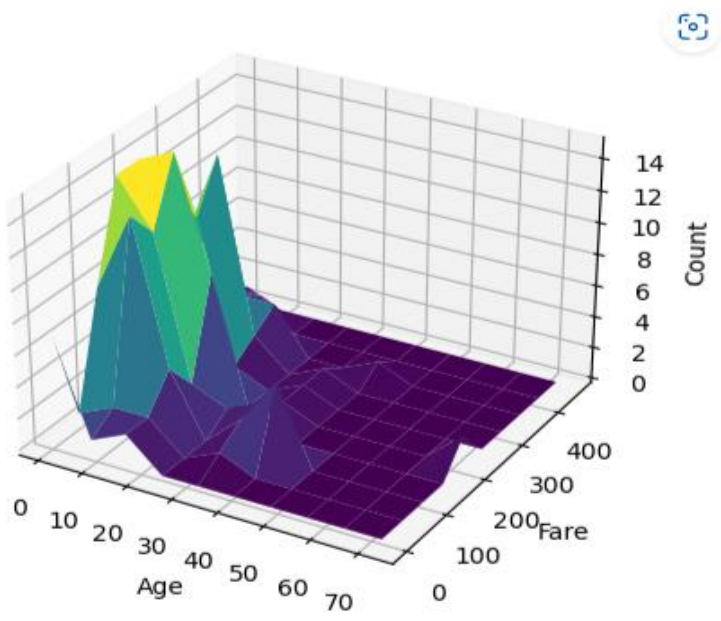

```
import pandas as pd
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
df = pd.read_csv("titanic.csv")
# Create a figure and axes for the 3D plot
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
# Scatter plot the data
ax.scatter(df["Age"], df["Fare"], df["Survived"], c=df["Survived"],
cmap='viridis')
# Add labels to the axes
ax.set_xlabel("Age")
ax.set_ylabel("Fare")
ax.set_zlabel("Survived")
plt.show()
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/



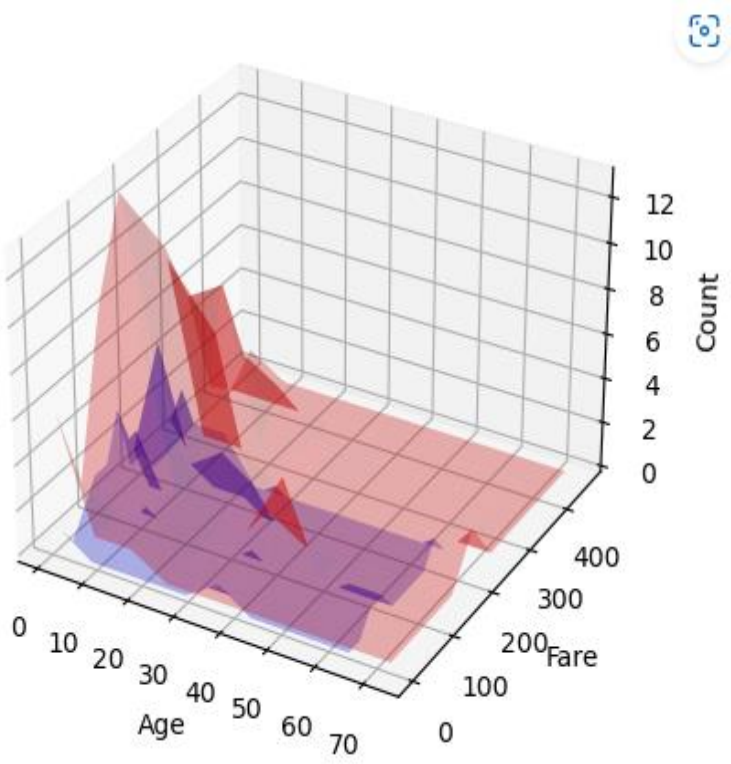

```
from mpl_toolkits.mplot3d import Axes3D
x = np.linspace(-5, 5, 100)
y = np.linspace(-5, 5, 100)
X, Y = np.meshgrid(x, y)
Z = np.sin(np.sqrt(X**2 + Y**2))
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.plot_surface(X, Y, Z, cmap='viridis')
ax.set_xlabel("X")
ax.set_ylabel("Y")
ax.set_zlabel("Z")
plt.show()
# Create a figure and axes for the 3D plot
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
hist, xedges, yedges = np.histogram2d(df["Age"], df["Fare"],
bins=10)
X, Y = np.meshgrid(xedges[:-1], yedges[:-1])
ax.plot_surface(X, Y, hist, cmap='viridis')
# Add labels to the axes
ax.set_xlabel("Age")
ax.set_ylabel("Fare")
ax.set_zlabel("Count")

# Show the plot
plt.show()
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

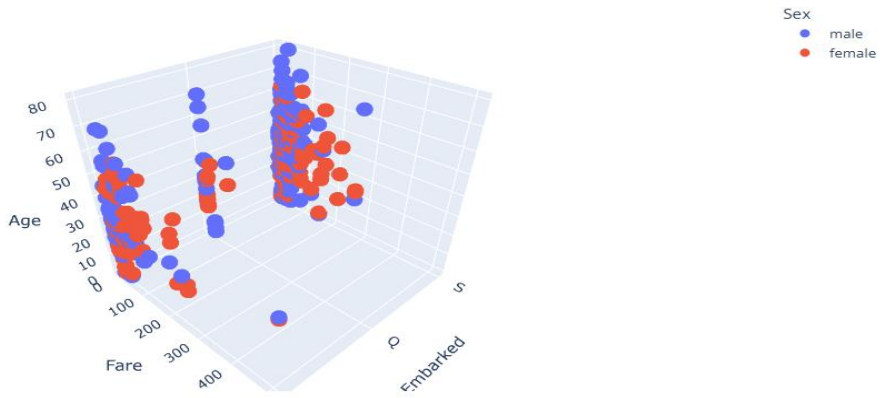



```
# Create a figure and axes for the 3D plot
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

# Create a histogram of the data
hist1, xedges1, yedges1 = np.histogram2d(df[df["Survived"] ==
1]["Age"], df[df["Survived"] == 1]["Fare"], bins=10)
hist2, xedges2, yedges2 = np.histogram2d(df[df["Survived"] ==
0]["Age"], df[df["Survived"] == 0]["Fare"], bins=10)

# Create a mesh grid of the binned data
X1, Y1 = np.meshgrid(xedges1[:-1], yedges1[:-1])
X2, Y2 = np.meshgrid(xedges2[:-1], yedges2[:-1])

# Plot the Tri-Surface plot
ax.plot_surface(X1, Y1, hist1, color='r', alpha=0.3)
ax.plot_surface(X2, Y2, hist2, color='b', alpha=0.3)

# Add labels to the axes
ax.set_xlabel("Age")
ax.set_ylabel("Fare")
ax.set_zlabel("Count")

# Show the plot
plt.show()
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

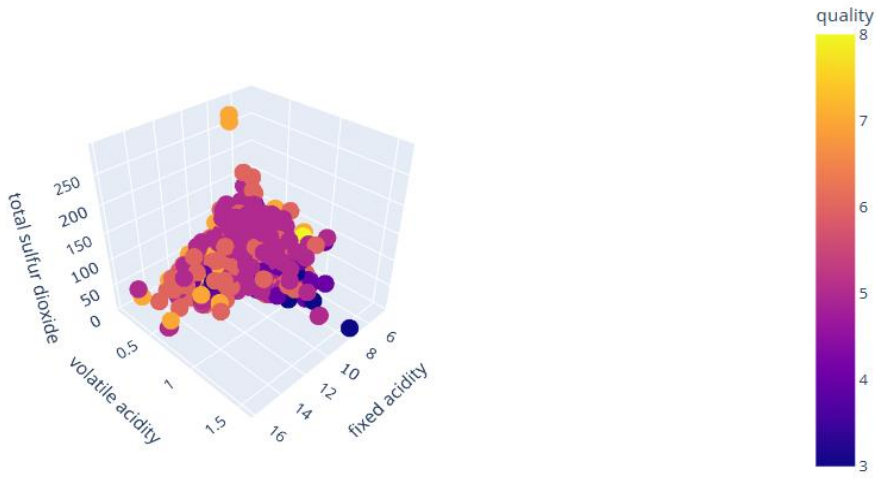



```
# 3D Scatter Plot
import plotly.express as px

fig = px.scatter_3d(titanic, x='Embarked', y='Fare', z='Age',
color='Sex')
fig.show()
```

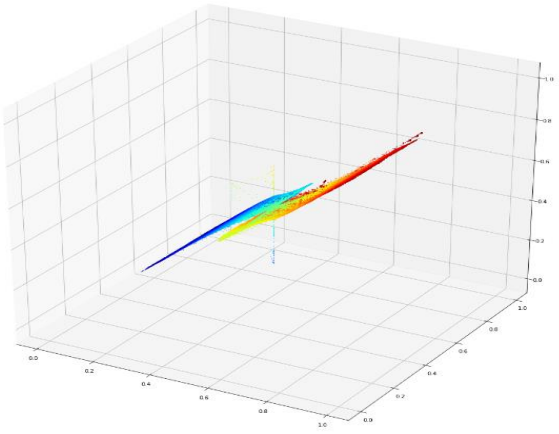


Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
# 3D Scatter Plot
import plotly.express as px
fig = px.scatter_3d(wine, x='fixed acidity', y='volatile acidity',
z='total sulfur dioxide', color='quality')
fig.show()
```



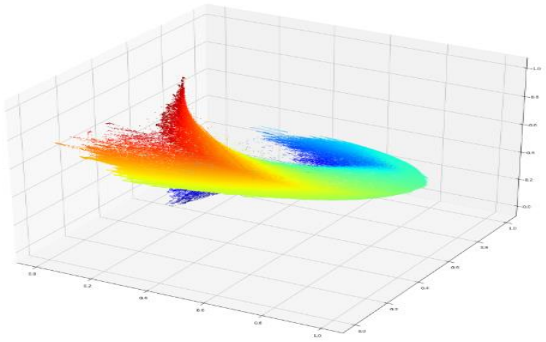

```
fig = plt.figure(figsize=(20,20))
ax = plt.axes(projection="3d")
ax.scatter3D(normalized_i_q, normalized_u_d, normalized_torque, s=0.5,
c=normalized_torque, cmap=plt.get_cmap("jet"))
plt.show()
```



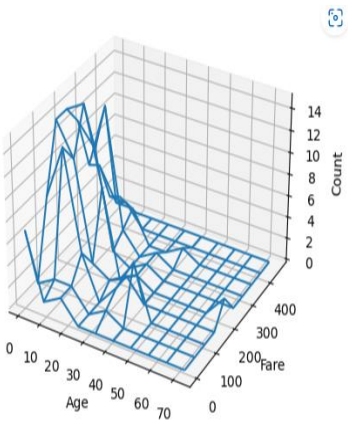
```
u_q = electric_motor_temprature_data['u_q']
u_d = electric_motor_temprature_data['u_d']
i_q = electric_motor_temprature_data['i_q']
i_d = electric_motor_temprature_data['i_d']
```

Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/

```
torque = electric_motor_temprature_data['torque']
normalized_u_q = (u_q - u_q.min())/(u_q.max()-u_q.min())
normalized_u_d = (u_d - u_d.min())/(u_d.max()-u_d.min())
normalized_i_q = (i_q - i_q.min())/(i_q.max()-i_q.min())
normalized_i_d = (i_d - i_d.min())/(i_d.max()-i_d.min())
normalized_torque = (torque - torque.min())/(torque.max()-torque.min())
fig = plt.figure(figsize=(20,20))
ax = plt.axes(projection="3d")
ax.scatter3D(normalized_u_q, normalized_u_d, normalized_torque, s=0.5,
c=normalized_torque, cmap=plt.get_cmap("jet"))
plt.show()
```



```
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
hist, xedges, yedges = np.histogram2d(df["Age"], df["Fare"],
bins=10)
X, Y = np.meshgrid(xedges[:-1], yedges[:-1])
ax.plot_wireframe(X, Y, hist)
ax.set_xlabel("Age")
ax.set_ylabel("Fare")
ax.set_zlabel("Count")
plt.show()
```



Syed Afroz Ali
Data Scientist (Kaggle Grandmaster)
https://www.kaggle.com/pythonafroz
https://www.linkedin.com/in/syed-afroz-70939914/