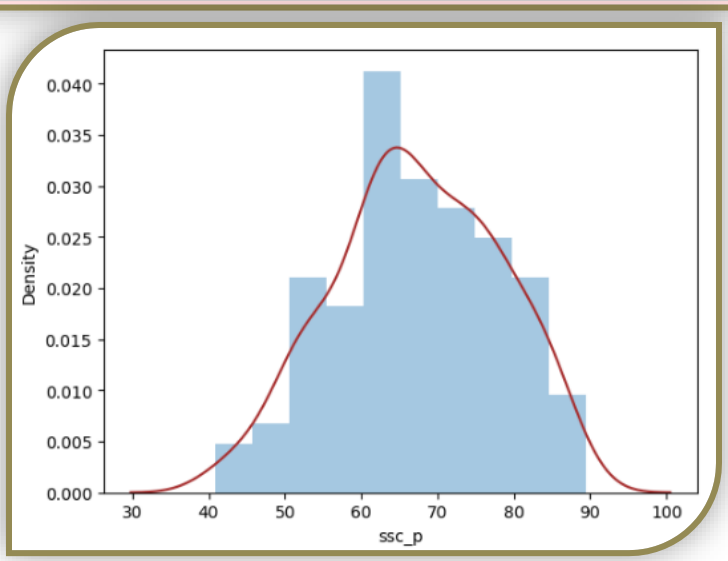
Seaborn – Plots

In Univariate Analysis

- **Dist plot:** *Used for visualizing the distribution of data.*

- *This plot shows how ssc_p varies with density.*

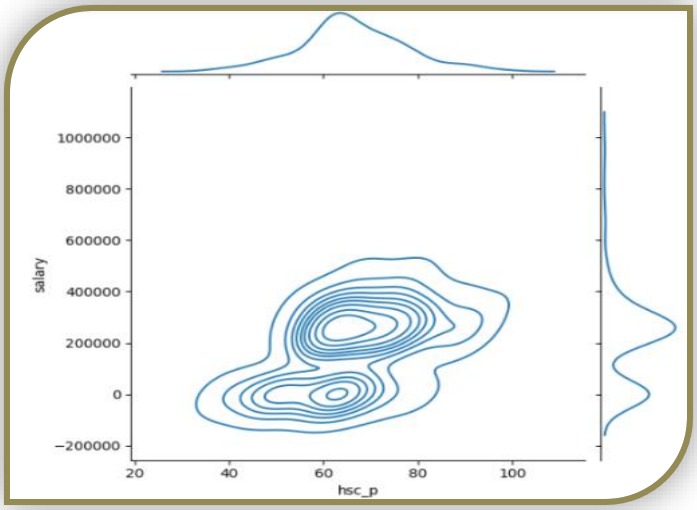
```
import seaborn as sns
import matplotlib.pyplot as plt
sns.distplot(dataset["ssc_p"],hist = True, kde = True, kde_kws = {'color':'brown'} )
```



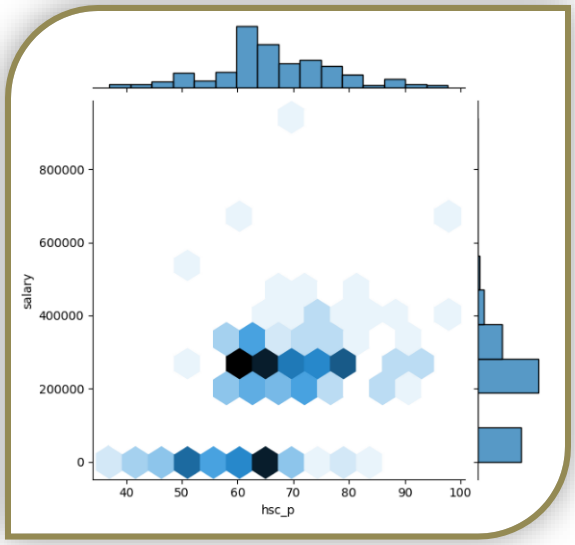
In Bivariate Analysis

- **Joint plot :** *It displays the relationship between two variables along with their individual distributions*
- *kde – refers to kernel density estimation.*
- *This plot shows the relationship between hsc_p and salary distribution as well as the density varies.*

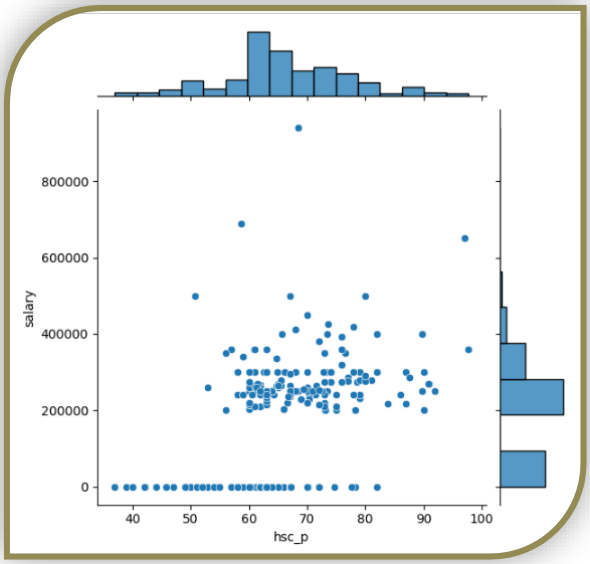
```
sns.jointplot(x='hsc_p', y='salary', data = dataset, kind = 'kde')
plt.show()
```



```
sns.jointplot(x="hsc_p", y="salary",data=dataset,kind = 'hex')
plt.show()
```

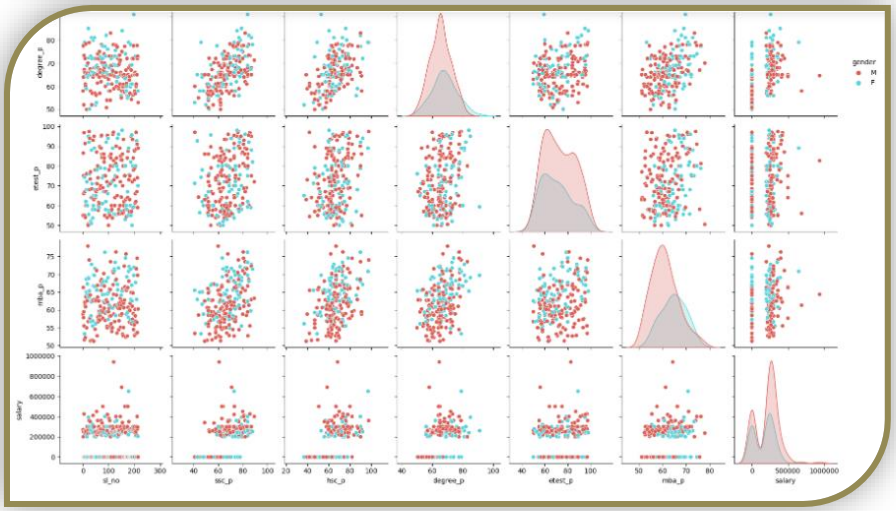



```
sns.jointplot(x = 'hsc_p', y = 'salary', data = dataset)
plt.show()
```



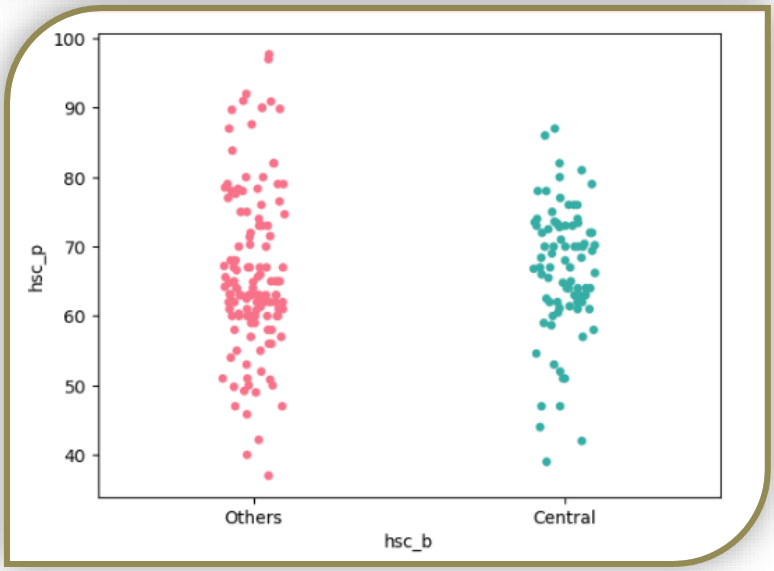
- **Pair plot :**
  Powerful tool for visualizing pairwise relationships in a dataset.
- *This pairplot shows relationship between all the two numerical columns with gender in kde and scatter plot.*

```
sns.pairplot(dataset, hue ="gender", diag_kind = "kde", kind = "scatter", palette = "hls")
plt.show()
```



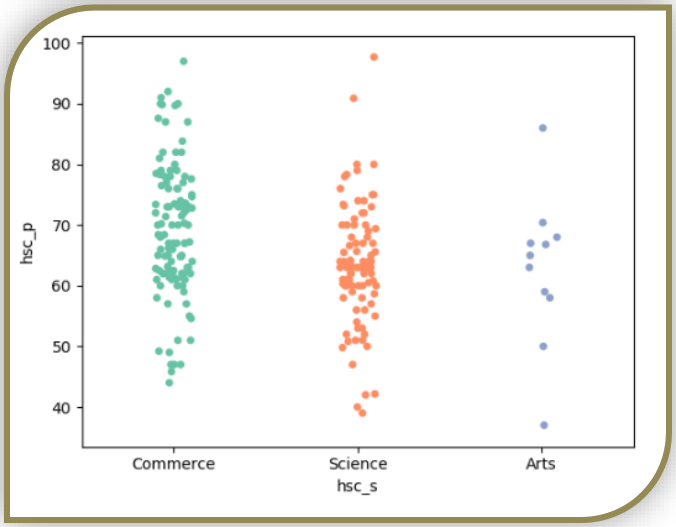
- **<u>Strip Plot :</u>** *It is a type of categorical scatterplot that displays individual data points for a given categorical variable.*
- *Here the plot compares hsc board and hsc marks, state board students are getting higher marks than central board students.*

```
sns.stripplot(x = "hsc_b", y = "hsc_p", data = dataset, palette = "husl")
plt.show()
```

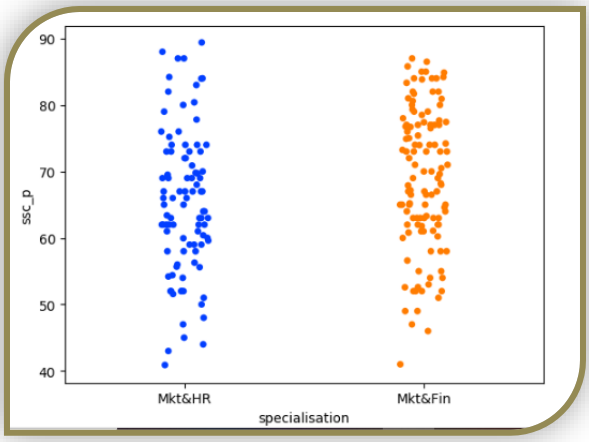


Other Strip plot Comparison:

```
sns.stripplot(x = "hsc_s", y = "hsc_p", data = dataset, palette = "Set2")
```

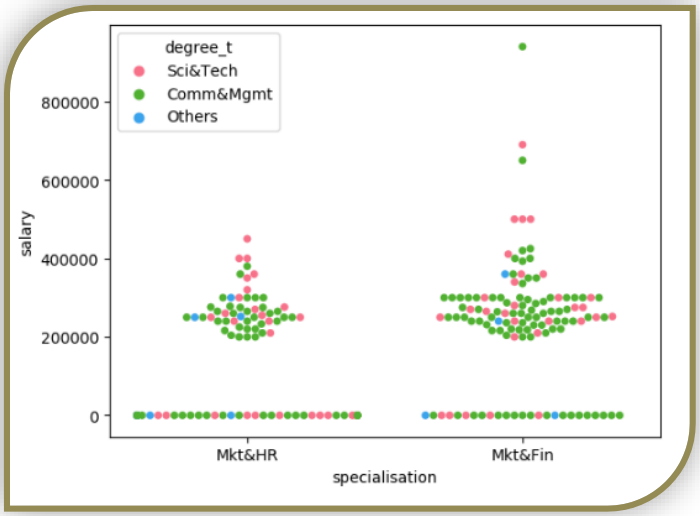



```
sns.stripplot(x = "specialisation", y = "ssc_p", data = dataset, palette = "bright")
plt.show()
```



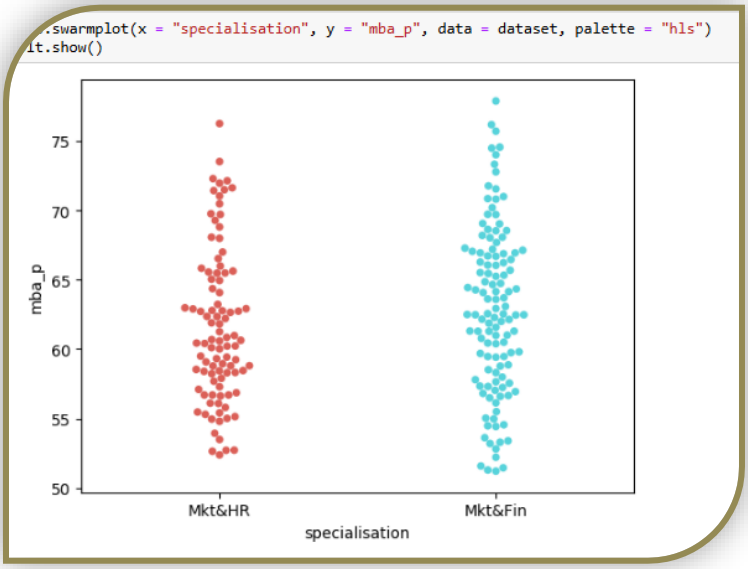
- **<u>Swarm Plot</u>** : *It is a type of categorical scatter plot used to visualize the distribution of data points within different categories without* ***overlapping markers****.*
- *This is an example plot for comparison between specialisation and salary with respect to degree*

```
sns.swarmplot(x = 'specialisation',y = 'salary',data = df, palette = "husl", hue = "degree_t")
plt.show
```

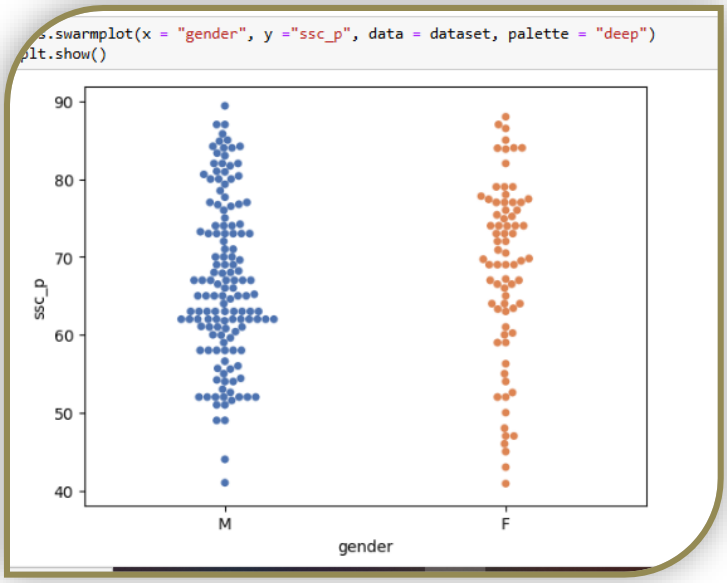


Other Swamplots:

```
.swarmplot(x = "specialisation", y = "mba_p", data = dataset, palette = "hls")
lt.show()
```



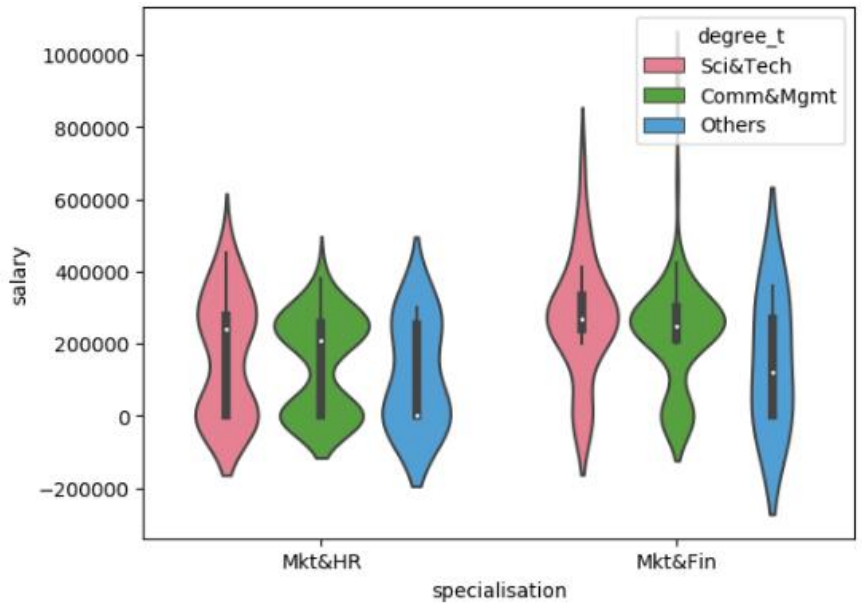

```
s.swarmplot(x = "gender", y ="ssc_p", data = dataset, palette = "deep")
plt.show()
```



- **<u>Distribution Plot(Violin Plot):</u>** *This technique combines the aspects of a box plot and a kernel density plot.*
- *It is used to visualize the distribution of quantitative data across one or more categorical variables.*

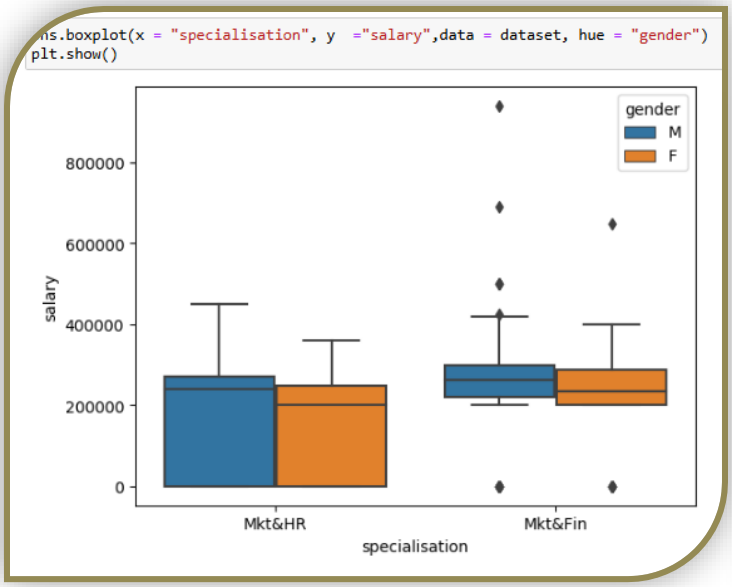
Comparison between specialisation vs salary with respect to degree

```
sns.violinplot(x = 'specialisation',y = 'salary',data = df, palette = "husl", hue = "degree_t")
plt.show()
```

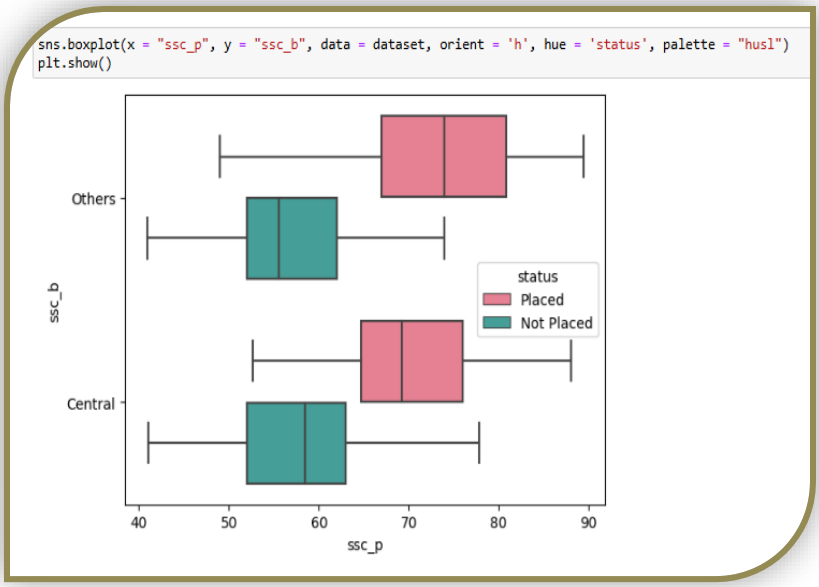


- **<u>Box Plot:</u>***Box plot also known as box-and-whisker plot is a visual representation of the distribution of quantitative data, particularly useful for comparing distributions across different categories.*
- **Box:** *Represents IQR, spanning from I quartile(25th percentile) to the III quartile(75th percentile). The line within the box indicates median(50th percentile).*
- **Whiskers:** *Extend from the box to the minimum and maximum data points within a certain range(typically 1.5 IQR)*

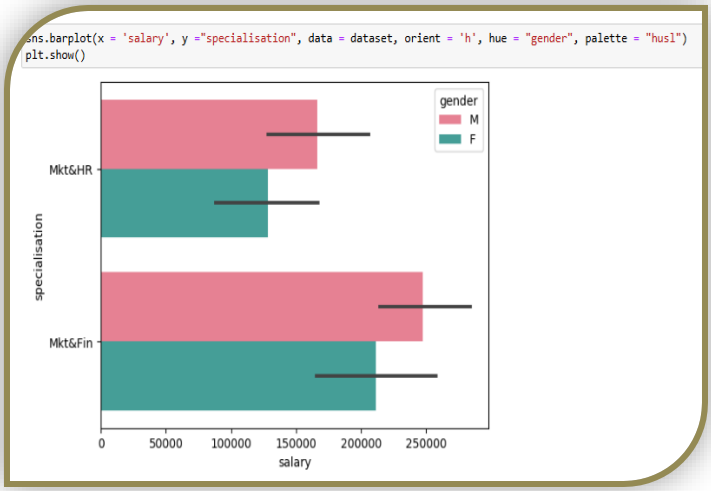
Specialisation vs Salary



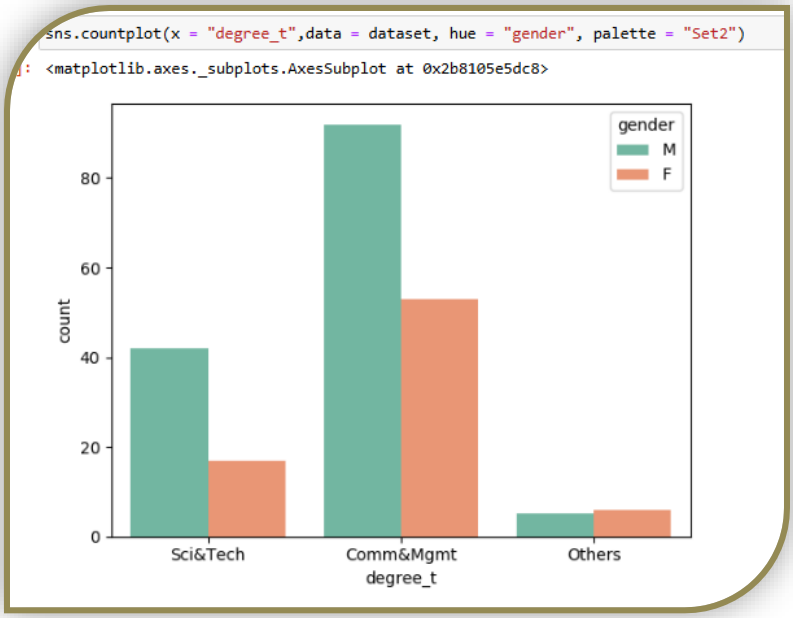
SSCmark vs SSC board w.r.t status



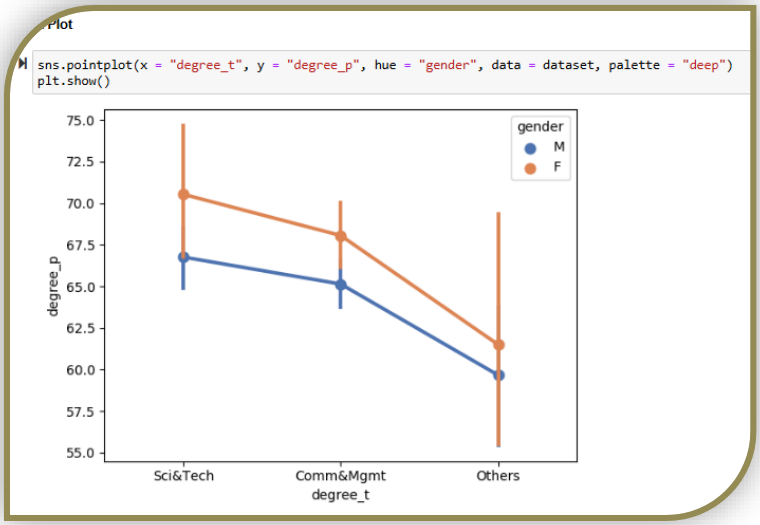
- **<u>Bar Plot :</u>** *This represents an estimate of central tendency (by default, the mean) for a numerical variable across different categories of categorical variable.*



- **Count Plot:** *It displays the count of observations in each category using bars, similar to histogram but for categorical data.*

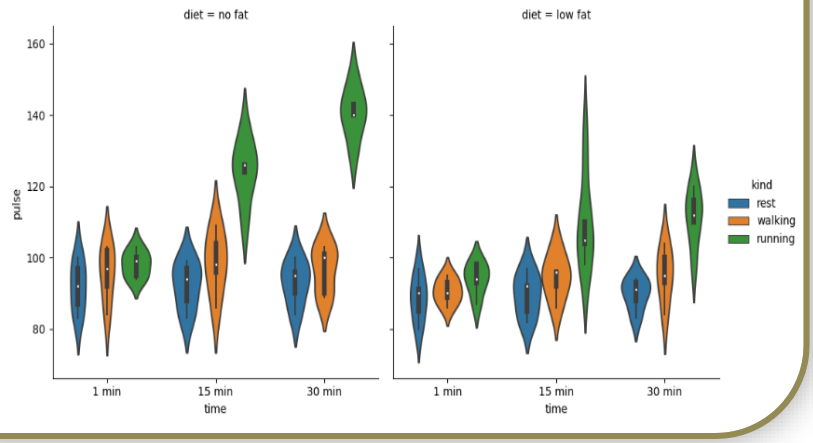


- **Point Plot:***It visualizes point estimates and confident intervals for a numerical variables across different levels of one or more categorical variables.*
- *The error bars provides a visual representation of the variability or uncertainity associated with each point estimate.*



- **<u>Factor Plot:</u>** *Cat plot formerly known as Factor plot used to draw different types of categorical plot.*
- *This plot shows the relationship between time and pulse with respect to diet for an exercise dataset.*

```
df = sns.load_dataset("exercise")
# print(df)
sns.catplot(x = "time", y ="pulse", hue = "kind", kind = "violin",col = "diet", data = df)
plt.show()
```



<u>*Summary of the above plot*</u>

- *When a person is doing exercise, he/she is at no fat diet, if the pulse>110 for 1min, that person is considered to be in walking else in rest state. If the pulse is between 120~160, then the person is at 30 min of running.*
- *When a person is at low fat diet, if the pulse range is between 80~150, then the person is at 15 mins of running. If that person is running for 30 mins, then the pulse ranges between 90~130.*

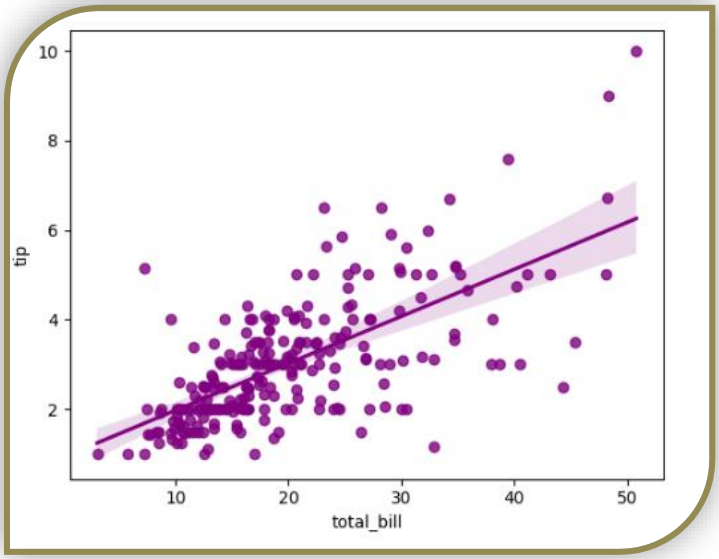
- **<u>Regression Plot:</u>** *Seaborn provides functions for visualizing linear relationships in datasets, primarily through regression plots.*

```
df = sns.load_dataset("tips")
# print(df)
sns.regplot(x = "total_bill", y = "tip", data = df,  color = "purple")   # takes x&y as variety of formats
sns.lmplot(x = "total_bill",y = "tip", data = df, palette = "husl", hue = "sex")      # takes x&y as strings
plt.show()
```

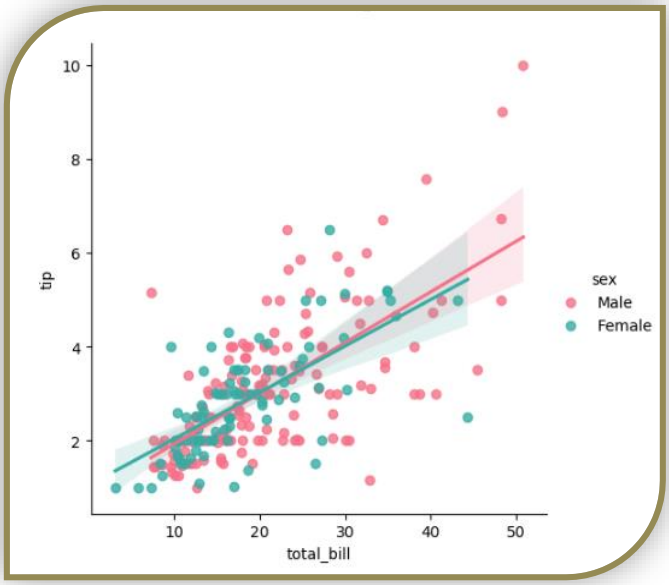
- *These plots help to understand the relationship between two variables and fit a regression model to the data.*

- *It automatically plots a scatter plot of the data points and overlays a linear regression line with a confidence interval around it.*

Regplot



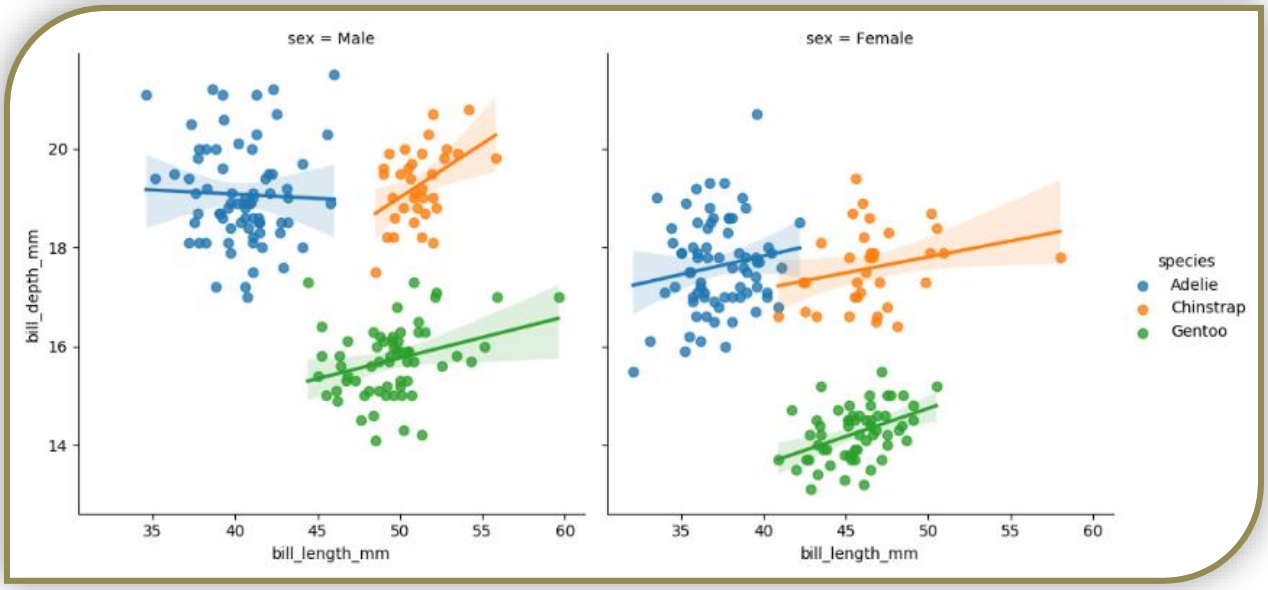
lmplot



- **<u>Lmplot:</u>** *This function is similar to regplot() but offers more flexibility for creating plots across multiple facets of a dataset.*
- *It allows you to specify hue, col, or row variables to create separate plots for different categories or subsets of the data, arranged in a grid.*
- *This is particularly useful for exploring conditional relationships.*
- *The order parameter is used to specify the degree of polynomial regression model to fit to the data.*
- *Order 1:LinearRegression(Default)*
- *Order 2: Polynomial Regression(Quadratic fit)*
- *Order 3: Polynomial Regression(Cubic fit)*

```
import seaborn as sns
import matplotlib.pyplot as plt
df = sns.load_dataset("penguins")
df
sns.lmplot(data=df, x="bill_length_mm", y="bill_depth_mm",col="species", row="sex",order = 2)
plt.show
```

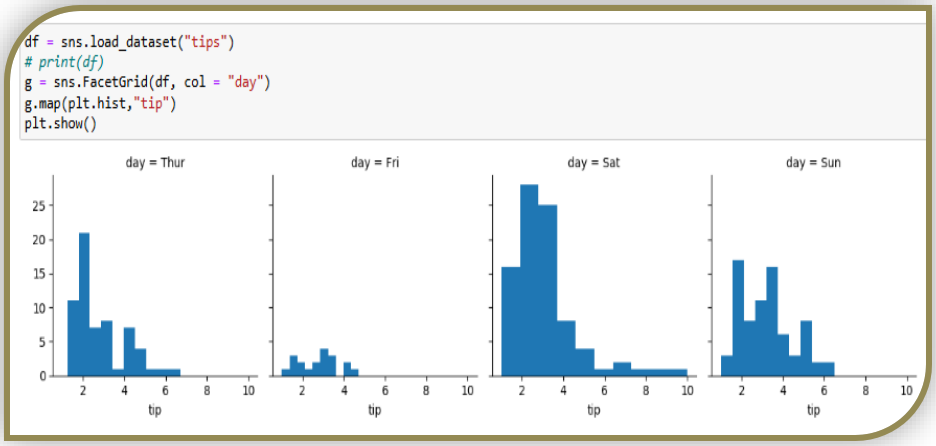
This lmplot shows how male and female penquins differs from bill length and bill depth.



Map plots

- **Facet Grid:** *FacetGrid class helps in visualizing distribution of one variable as well as the relationship between multiple variables separately within subsets of*

```
df = sns.load_dataset("tips")
# print(df)
g = sns.FacetGrid(df, col = "day")
g.map(plt.hist,"tip")
plt.show()
```



your dataset using multiple panels.

- *A FacetGrid can be drawn with up to three dimensions row, col, and hue.*
- *The above plot shows how tips dataset varies with each day, tip in the x axis and total bill in the y-axis.*

- **Pair Grid:** *It constructs a grid of subplots where each variable in the dataset is mapped to a row and a column.*
- *This class maps each variable in a dataset onto a column and row in a grid of multiple axes.*

```
df = sns.load_dataset("tips")
df
g = sns.PairGrid(df)
g.map(plt.scatter)
plt.show()
```

Scatter plot with pair grid