

Seaborn Cheatsheet from datacamp

```
In [1]: %config Completer.use_jedi = False
```

```
In [2]: # Import Libraries
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings("ignore")
sns.set_style('whitegrid')
sns.set()
```

```
In [3]: # Load dataset
tips = sns.load_dataset('tips')
titanic = sns.load_dataset('titanic')
iris = sns.load_dataset('iris')
data = pd.DataFrame({'x': np.arange(1,101), 'y': np.random.normal(0,4,100)})
```

```
In [4]: # Function for Data info
def data_info (df):
    column, nunique, null, null_p, dtype = [],[],[],[],[]
    for col in df.columns:
        column.append(col)
        nunique.append(df[col].nunique())
        null.append(df[col].isnull().sum())
        null_p.append((df[col].isnull().sum()/df[col].count())*100)
        dtype.append(df[col].dtype)
    return pd.DataFrame({'Column': column, 'N-unique': nunique, 'Null': null, 'Null
```

```
In [5]: data_info(tips)
```

Out[5]:

	Column	N-unique	Null	Null Percent	Dtype
0	total_bill	229	0	0.0	float64
1	tip	123	0	0.0	float64
2	sex	2	0	0.0	category
3	smoker	2	0	0.0	category
4	day	4	0	0.0	category
5	time	2	0	0.0	category
6	size	6	0	0.0	int64

```
In [6]: data_info(iris)
```

Out[6]:

	Column	N-unique	Null	Null Percent	Dtype
0	sepal_length	35	0	0.0	float64

	Column	N-unique	Null	Null Percent	Dtype
1	sepal_width	23	0	0.0	float64
2	petal_length	43	0	0.0	float64
3	petal_width	22	0	0.0	float64
4	species	3	0	0.0	object

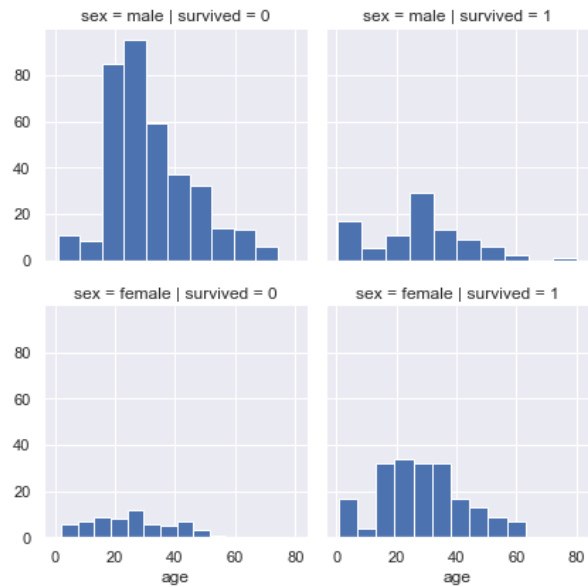
```
In [7]: data_info(titanic)
```

Out[7]:

	Column	N-unique	Null	Null Percent	Dtype
0	survived	2	0	0.000000	int64
1	pclass	3	0	0.000000	int64
2	sex	2	0	0.000000	object
3	age	88	177	24.789916	float64
4	sibsp	7	0	0.000000	int64
5	parch	7	0	0.000000	int64
6	fare	248	0	0.000000	float64
7	embarked	3	2	0.224972	object
8	class	3	0	0.000000	category
9	who	3	0	0.000000	object
10	adult_male	2	0	0.000000	bool
11	deck	7	688	338.916256	category
12	embark_town	3	2	0.224972	object
13	alive	2	0	0.000000	object
14	alone	2	0	0.000000	bool

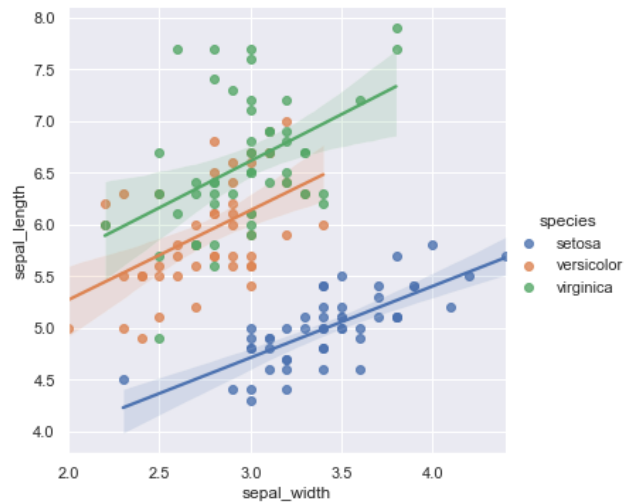
Axis Grids

```
In [8]: a = sns.FacetGrid(data=titanic, col='survived', row='sex')
a = a.map(plt.hist, 'age')
```

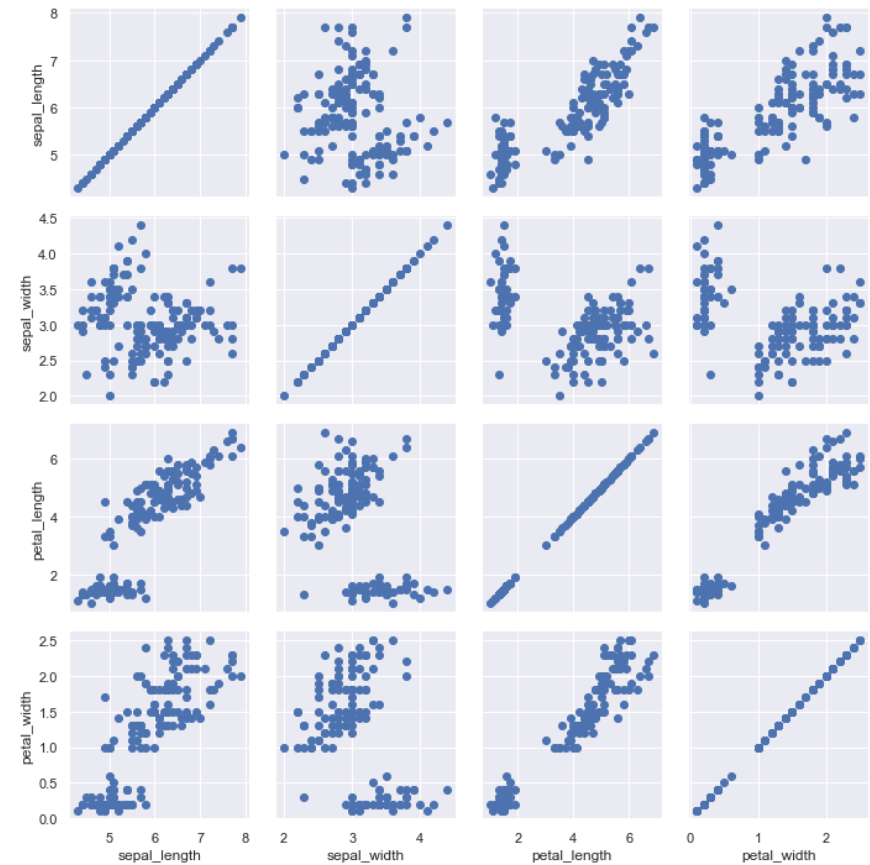


```
In [9]: sns.lmplot(data=iris, x='sepal_width', y='sepal_length', hue='species')
```

```
Out[9]: <seaborn.axisgrid.FacetGrid at 0x1411da0f9d0>
```

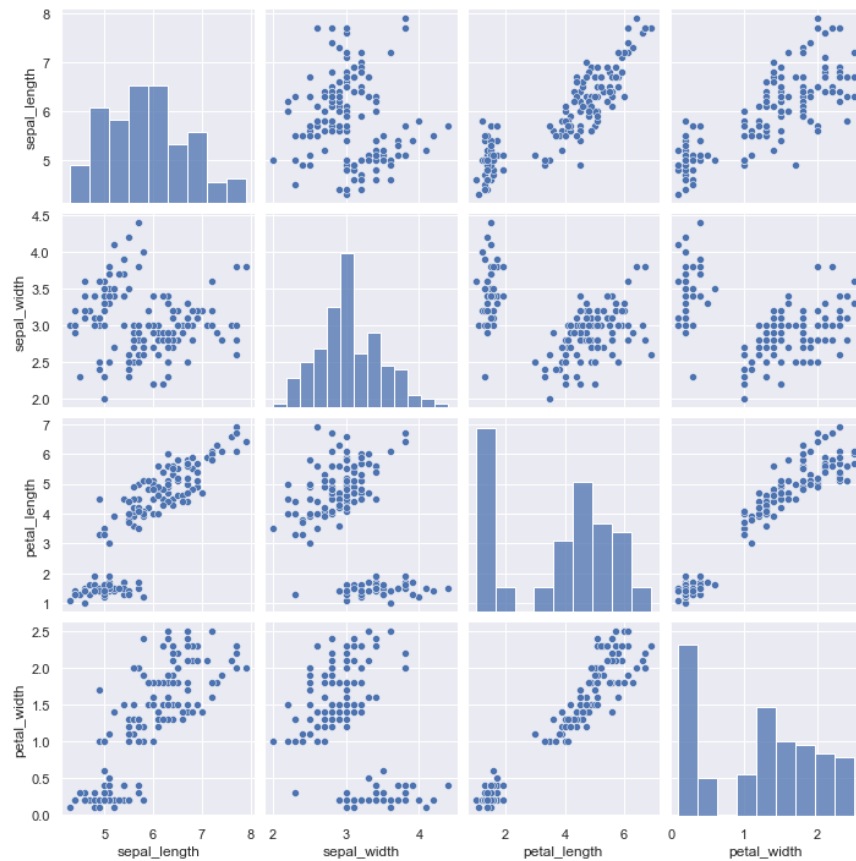


```
In [10]: p = sns.PairGrid(iris)
p = p.map(plt.scatter)
```



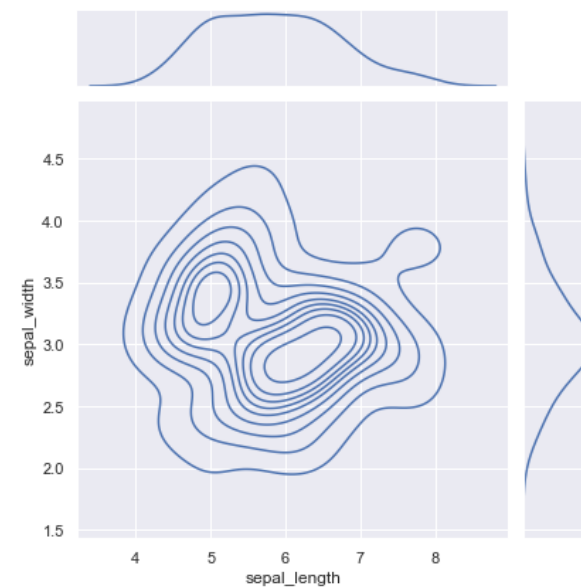
```
In [11]: sns.pairplot(iris)
```

```
Out[11]: <seaborn.axisgrid.PairGrid at 0x14126375ac0>
```



```
In [12]: sns.jointplot(data=iris, x='sepal_length', y='sepal_width', kind='kde')
```

```
Out[12]: <seaborn.axisgrid.JointGrid at 0x14126c6e1f0>
```



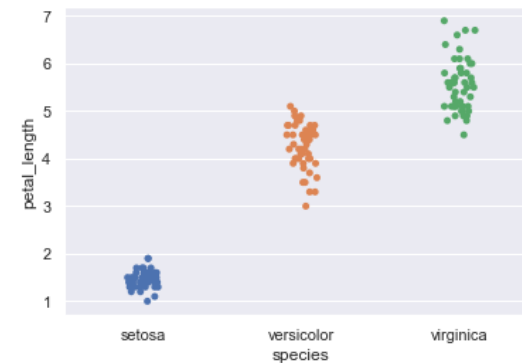
```
In [13]: iris.columns
```

```
Out[13]: Index(['sepal_length', 'sepal_width', 'petal_length', 'petal_width',  
              'species'],  
              dtype='object')
```

Categorical Plot

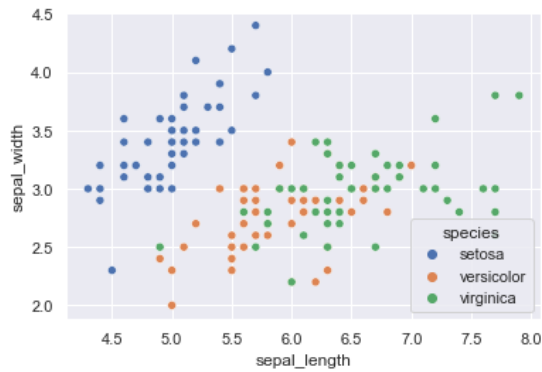
```
In [14]: # Strip Plot  
sns.stripplot(data=iris, x='species', y='petal_length')
```

```
Out[14]: <AxesSubplot:xlabel='species', ylabel='petal_length'>
```



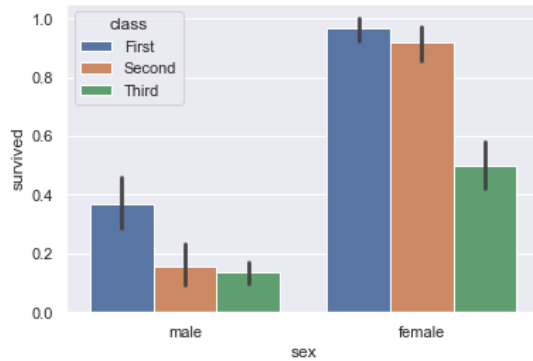
```
In [15]: # Scatter Plot  
sns.scatterplot(data=iris, x='sepal_length', y='sepal_width', hue='species')
```

```
Out[15]: <AxesSubplot:xlabel='sepal_length', ylabel='sepal_width'>
```



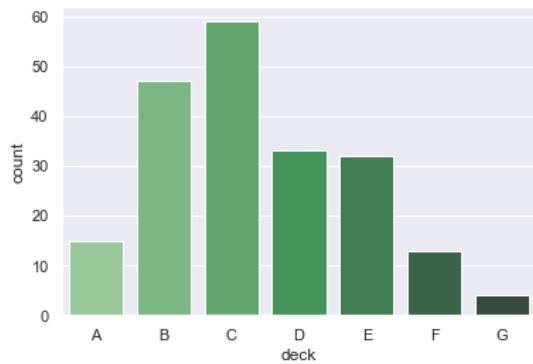
```
In [16]: # Bar Chart
sns.barplot(data=titanic, x='sex', y='survived', hue='class')
```

```
Out[16]: <AxesSubplot:xlabel='sex', ylabel='survived'>
```

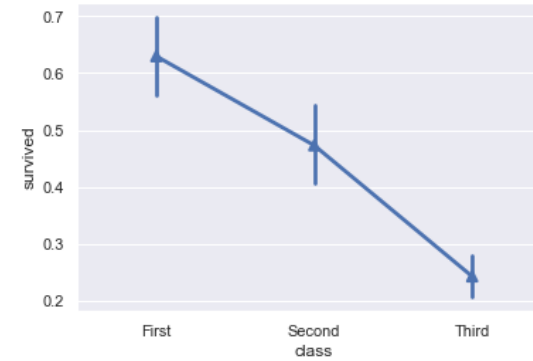


```
In [17]: # Count Plot
sns.countplot(data=titanic, x='deck', palette='Greens_d')
```

```
Out[17]: <AxesSubplot:xlabel='deck', ylabel='count'>
```

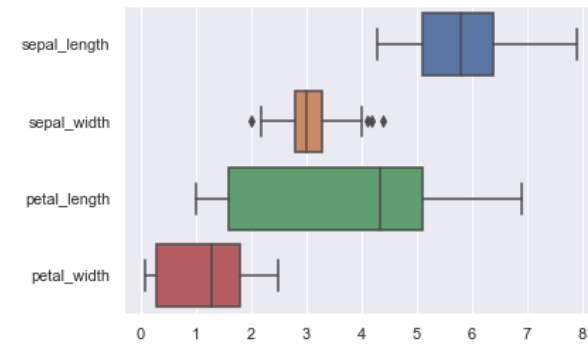


```
In [18]: # Point Plot
sns.pointplot(data=titanic, x='class', y='survived', markers=['^', 'o'], linestyle=[
```



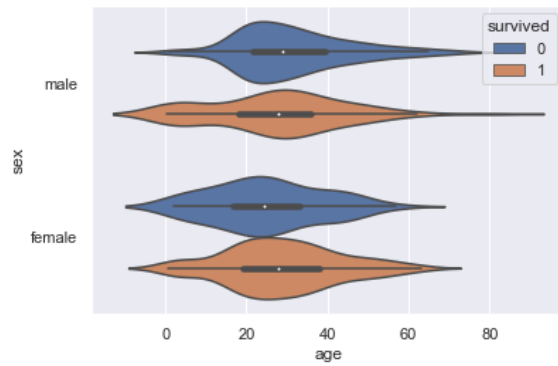
```
In [19]: # Box plot
sns.boxplot(data=iris, orient='h')
```

```
Out[19]: <AxesSubplot:>
```



```
In [20]: # Violin Plot
sns.violinplot(data=titanic, x='age', y='sex', hue='survived')
```

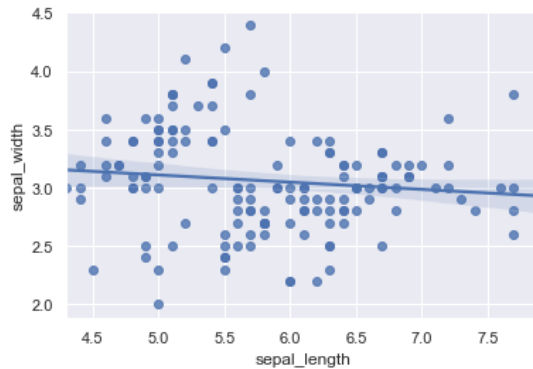
```
Out[20]: <AxesSubplot:xlabel='age', ylabel='sex'>
```



Regression Plot

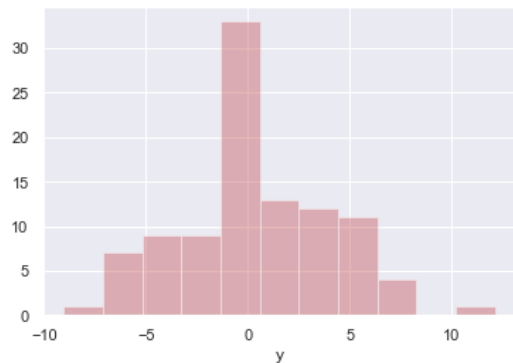
```
In [21]: sns.regplot(data=iris, x='sepal_length', y='sepal_width')
```

```
Out[21]: <AxesSubplot:xlabel='sepal_length', ylabel='sepal_width'>
```



Distribution Plot

```
In [22]: sns.distplot(data['y'], kde=False, color='r');
```



Matrix Plot

```
In [23]: sns.heatmap(iris.corr(), annot=True);
```

