

### Import the Seaborn Library

```
from matplotlib import pyplot as plt
import seaborn as sns
import numpy as np
```

Seaborn is an extension to Matplotlib with more visually appealing syntax and additional Chart Types. That's why Matplotlib should also be imported.

If we want to calculate **aggregates** we need to **import numpy** as well.

### Bar Plot

```
sns.barplot(
    data=df,
    x="x value column",
    y="y value column",
    # everything specified below is optional
    ci="sd"
    estimator=np.median | len
    hue="column to compare"
)
plt.show()
# ci="sd" changes the error bar to standard deviation
# estimator is used to specify the aggregation and takes any argument that works on a list. (examples provided in code)
# hue adds a nested categorical variable to compare to the "y value column"
```

If the specified columns need to be **aggregated** first, Seaborn will perform that aggregation automatically. (*mean by default*)

Seaborn will, by default, provide an **error bar** displaying the **bootstrapped confidence interval**(95%).

### Aggregates (with numpy)

Median `np.median(df.column_name)`

### KDE (Key Density Estimator) Plots

### Boxplots

```
sns.boxplot(
    data=df,
    x='label',
    y='value',

    # optional
    width=0.45
)
plt.show()
# In Seaborn it's also possible to plot multiple Boxplots in one viz
```

The box represents the **interquartile range**  
The line in the middle of the box is the **median**  
The end lines are the **first and third quartiles**  
The diamonds show **outliers**

### Violin Plots

```
sns.violinplot(
    data=df,
    x="label",
    y="value"
)
```

Two KDE plots that are symmetrical along the center line. (Just for visual effect)

A **white dot** represents the median.

The **thick black line** in the center of each violin represents the interquartile range.

The **lines that extend from the center** are the confidence intervals (95%)

### Seaborn Styling (Figure Style and Scale)

### Seaborn Styling (Figure Style and Scale) (cont)

```
'font.size': 19.2,
'grid.linewidth': 1.6,
'legend.fontsize': 16.0,
'lines.linewidth': 2.8,
'lines.markeredgewidth': 0.0,
'lines.markersize': 11.2,
'patch.linewidth': 0.48,
'xtick.labelsize': 16.0,
'xtick.major.pad': 11.2,
'xtick.major.width': 1.6,
'xtick.minor.width': 0.8,
'ytick.labelsize': 16.0,
'ytick.major.pad': 11.2,
'ytick.major.width': 1.6,
'ytick.minor.width': 0.8
}
```

### Seaborn Styling (Color)

```
# If you want to quickly see what a palette looks like
# Save a palette to a variable:
palette = sns.color_palette("bright")
# Use palplot and pass in the variable:
sns.palplot(palette)
# Select a palette in Seaborn:
sns.set_palette("Paired")
# Default Palettes
-> deep, muted, pastel, bright, dark, colorblind
# More Palettes using Color Brewer:
http://colorbrewer2.org
```

```
sns.kdeplot(dataset1, shade=True)
```

```
sns.kdeplot(dataset2, shade=True)
```

...

KDE Plots show the **distribution** of an **univariate** dataset.

**univariate** datasets have only **one** variable. (e.g.: Temperature)

**shade** defines if the area under the line is shaded

```
# Themes: (called prior to plot)
sns.set_style("")
->darkgrid, whitegrid, dark,
white, ticks
#Removes Plot Borders (called
after plot)
sns.despine() (default:
top=True, right=True)
-> bottom, left
#Adjust font- and label size
sns.set_context(context="paper",
font_scale=1.4, rc={"grid.li-
newidth": 0.6} )
-> paper, notebook (default),
talk, poster
    In order of relative size
-> Arguments for rc parameter:
{
'axes.labelsize': 17.6,
'axes.titlesize': 19.0,
```

C

By **Justin1209** (Justin1209)  
[cheatography.com/justin1209/](http://cheatography.com/justin1209/)

Published 26th November, 2019.  
Last updated 26th November, 2019.  
Page 1 of 2.

Sponsored by **CrosswordCheats.com**  
Learn to solve cryptic crosswords!  
<http://crosswordcheats.com>