

---

# iSeaborn

*Release 0.0.1*

Sep 13, 2020



---

# Contents

---

<b>1</b>	<b>Installation</b>	<b>1</b>
1.1	From PyPI . . . . .	1
1.2	From Source . . . . .	1
<b>2</b>	<b>Tutorial</b>	<b>3</b>
2.1	Bar Plots . . . . .	3
2.1.1	Vertical Bar Plot . . . . .	3
2.1.2	Choose from Interactive Tools . . . . .	3
2.1.3	Control Plot Orders . . . . .	4
2.1.4	Draw Horizontal Bar . . . . .	4
2.1.5	Set Desired Estimator . . . . .	4
2.1.6	Choose From Color Palettes . . . . .	4
2.1.7	Set Specific Color . . . . .	5
2.1.8	Use Hue for Visualization . . . . .	5
2.1.9	Change Plot Properties . . . . .	5
2.1.10	And More . . . . .	5
<b>3</b>	<b>Jupyter Notebook Playground</b>	<b>7</b>
<b>4</b>	<b>Community</b>	<b>9</b>
4.1	Code of Conduct . . . . .	9
4.1.1	Our Pledge . . . . .	9
4.1.2	Our Standards . . . . .	9
4.1.3	Our Responsibilities . . . . .	10
4.1.4	Scope . . . . .	10
4.1.5	Enforcement . . . . .	10
4.1.6	Attribution . . . . .	10
4.2	Contributing . . . . .	10
4.2.1	Environment Setup and Dev Installation . . . . .	10
4.2.2	Developing Docs . . . . .	11



# CHAPTER 1

---

## Installation

---

### 1.1 From PyPI

Installing the package from PyPI is the easiest method of installation. To install:

```
$ pip install iSeaborn
```

### 1.2 From Source

The package can also be installed direct from source. This a method of installation is preferred for development or contribution to the package. To install from git source:

```
$ git clone https://github.com/pseudoPixels/iSeaborn.git  
$ cd iSeaborn  
$ pip install -e .
```



## 2.1 Bar Plots

### 2.1.1 Vertical Bar Plot

Draw a set of vertical bar plots grouped by a categorical variable:

```
1 import iSeaborn as isn
2 from bokeh.plotting import output_file, save
3
4 tips = isn.load_dataset("tips")
5 fig = isn.barplot(x="day", y="total_bill", data=tips)
6
7 output_file("verticalBarPlot.html")
8 save(fig)
```

### 2.1.2 Choose from Interactive Tools

Choose the required interactive tools for visualization by passing value to tools options. Default tools are “pan,box\_select,wheel\_zoom,box\_zoom,reset,save”

```
1 import iSeaborn as isn
2 from bokeh.plotting import output_file, save
3
4 tips = isn.load_dataset("tips")
5 fig = isn.barplot(x="day", y="total_bill", data=tips, tools="pan, save")
6
7 output_file("chooseInteractiveTools.html")
8 save(fig)
```

### 2.1.3 Control Plot Orders

Control bar order by passing an explicit order:

```
1 import iSeaborn as isn
2 from bokeh.plotting import output_file, save
3
4 tips = isn.load_dataset("tips")
5 fig = isn.barplot(x="time", y="tip", data=tips, order=["Lunch", "Dinner"])
6
7 output_file("chooseInteractiveTools.html")
8 save(fig)
```

### 2.1.4 Draw Horizontal Bar

Draw a set of horizontal bars automatically with change of axis.

```
1 import iSeaborn as isn
2 from bokeh.plotting import output_file, save
3
4 tips = isn.load_dataset("tips")
5 fig = isn.barplot(x="tip", y="day", data=tips)
6
7 output_file("drawHorizontalBar.html")
8 save(fig)
```

### 2.1.5 Set Desired Estimator

For example, use median as the estimate of central tendency

```
1 import iSeaborn as isn
2 from bokeh.plotting import output_file, save
3 from numpy import median
4
5 tips = isn.load_dataset("tips")
6 fig = isn.barplot(x="day", y="tip", data=tips, estimator=median)
7
8 output_file("setDesiredEstimator.html")
9 save(fig)
```

### 2.1.6 Choose From Color Palettes

Use a different color palette for the bars:

```
1 import iSeaborn as isn
2 from bokeh.plotting import output_file, save
3
4 tips = isn.load_dataset("tips")
5 fig = isn.barplot(x="day", y="total_bill", data=tips, palette="Blues_d")
6
7 output_file("chooseFromColorPalettes.html")
8 save(fig)
```



## 2.1.7 Set Specific Color

Plot all bars in a single color:

```

1 import iSeaborn as isn
2 from bokeh.plotting import output_file, save
3
4 tips = isn.load_dataset("tips")
5 fig = isn.barplot(x="day", y="total_bill", data=tips, color="salmon")
6
7 output_file("setPreferredColor.html")
8 save(fig)

```

## 2.1.8 Use Hue for Visualization

Draw a set of vertical bars with nested grouping by a two variables:

---

**Note:** Click the legend text to view only a selected category.

---

```

1 import iSeaborn as isn
2 from bokeh.plotting import output_file, save
3
4 tips = isn.load_dataset("tips")
5 fig = isn.barplot(x="day", y="total_bill", hue="sex", data=tips)
6
7 output_file("useHue.html")
8 save(fig)

```

## 2.1.9 Change Plot Properties

Change different plot properties:

```

1 import iSeaborn as isn
2 from bokeh.plotting import output_file, save
3
4 tips = isn.load_dataset("tips")
5 fig = isn.barplot(x="day", y="total_bill", data=tips,
6                  plot_width=600, plot_height=200,
7                  plot_title="Awesome Plot Title")
8
9 output_file("setPlotProps.html")
10 save(fig)

```

## 2.1.10 And More

Change the other aesthetics of the plot as key word arguments as available in *bokeh.plotting.figure.vbar* , such as changing alpha of the plot.

---

**Note:** List of all the aesthetics properties :: <https://docs.bokeh.org/en/latest/docs/reference/plotting.html#bokeh.plotting.figure.Figure.vbar>

---

```
1 import iSeaborn as isn
2 from bokeh.plotting import output_file, save
3
4 tips = isn.load_dataset("tips")
5 fig = isn.barplot(x="day", y="total_bill", data=tips, alpha=0.3)
6
7 output_file("vbarProps.html")
8 save(fig)
```

## CHAPTER 3

---

### Jupyter Notebook Playground

---

You can run examples and tryout iSeaborn on the cloud from Jupyter Notebook without any requirements of installation. Click the following to launch the project.

---

**Note:** You can find example notebooks of iSeaborn at: *examples/notebooks/*

---



Here, we've including instructions for how to contribute to the iSeaborn project. We welcome both code contributions and issue reports.

## 4.1 Code of Conduct

### 4.1.1 Our Pledge

In the interest of fostering an open and welcoming environment, we as contributors and maintainers pledge to making participation in our project and our community a harassment-free experience for everyone, regardless of age, body size, disability, ethnicity, sex characteristics, gender identity and expression, level of experience, education, socio-economic status, nationality, personal appearance, race, religion, or sexual identity and orientation.

### 4.1.2 Our Standards

Examples of behavior that contributes to creating a positive environment include:

- Using welcoming and inclusive language
- Being respectful of differing viewpoints and experiences
- Gracefully accepting constructive criticism
- Focusing on what is best for the community
- Showing empathy towards other community members

Examples of unacceptable behavior by participants include:

- The use of sexualized language or imagery and unwelcome sexual attention or advances
- Trolling, insulting/derogatory comments, and personal or political attacks
- Public or private harassment

- Publishing others' private information, such as a physical or electronic address, without explicit permission
- Other conduct which could reasonably be considered inappropriate in a professional setting

### 4.1.3 Our Responsibilities

Project maintainers are responsible for clarifying the standards of acceptable behavior and are expected to take appropriate and fair corrective action in response to any instances of unacceptable behavior.

Project maintainers have the right and responsibility to remove, edit, or reject comments, commits, code, wiki edits, issues, and other contributions that are not aligned to this Code of Conduct, or to ban temporarily or permanently any contributor for other behaviors that they deem inappropriate, threatening, offensive, or harmful.

### 4.1.4 Scope

This Code of Conduct applies both within project spaces and in public spaces when an individual is representing the project or its community. Examples of representing a project or community include using an official project e-mail address, posting via an official social media account, or acting as an appointed representative at an online or offline event. Representation of a project may be further defined and clarified by project maintainers.

### 4.1.5 Enforcement

Instances of abusive, harassing, or otherwise unacceptable behavior may be reported by contacting the project team on Slack. All complaints will be reviewed and investigated and will result in a response that is deemed necessary and appropriate to the circumstances. The project team is obligated to maintain confidentiality with regard to the reporter of an incident. Further details of specific enforcement policies may be posted separately.

Project maintainers who do not follow or enforce the Code of Conduct in good faith may face temporary or permanent repercussions as determined by other members of the project's leadership.

### 4.1.6 Attribution

This Code of Conduct is adapted from the Contributor Covenant, version 1.4, available at <https://www.contributor-covenant.org/version/1/4/code-of-conduct.html>

For answers to common questions about this code of conduct, see <https://www.contributor-covenant.org/faq>

## 4.2 Contributing

If you are planning to contribute to iSeaborn, you will first need to set up a local development environment.

### 4.2.1 Environment Setup and Dev Installation

1. Install Python. Python 3.6 or above recommended, but our CI/CD pipeline currently tests against up-to-date patch versions of Python 2.7, 3.5, 3.6, and 3.7.
2. Create and activate a virtualenv, using the tool of your choice:

```
$ conda create -n iSeabornEnv python=3.6
$ conda activate iSeabornEnv
```

3. Clone the dagster repository to the destination of your choice:

```
$ git clone https://github.com/pseudoPixels/iSeaborn.git
```

4. Install from the setup.py in development mode:

```
$ cd iSeaborn
$ pip install -e .
```

## 4.2.2 Developing Docs

The documentation of the project is created using sphinx. Our documentation employs a combination of Markdown and reStructuredText. To build your updated docs:

```
$ cd docs
$ make html
```