

## **Data Visualization with Python**

## Cheat Sheet: Maps, Waffles, WordCloud and Seaborn

The state of the s				
Function	Description	Syntax	Example	Visual
Folium				
Map	Create a map object with specified center coordinates and zoom level.	<pre>folium.Map(location=[lat, lon], zoom_start=n)</pre>	<pre>world_map = folium.Map() canada =folium.Map(location=[56.130,</pre>	
Marker	Add a marker to the map with custom icon, popup, and tiles Tiles as Stamen Toner	<pre>folium.Marker(location=[lat , lon ], popup='Marker Popup', tiles='Stamen Toner').add_to(map)</pre>	<pre>folium.Marker(location=[556.130,     -106.35], tooltip='Marker', tiles='Stamen Toner').add_to(world_map)</pre>	
	Tiles as Stamen Terrain	<pre>folium.Marker(location=[lat , lon ], popup='Marker Popup', tiles='Stamen Terrain').add_to(map)</pre>	folium.Marker(location=[556.130, -106.35], tooltip='Marker', tiles='Stamen Terrain').add_to(world_map)	
Circle	Add a circle to the map with specified radius, color, and fill opacity.	<pre>folium.features.CircleMarker(location=[lat, lon], radius=n, color='red', fill_opacity=n).add_to(map)</pre>	<pre>folium.features.CircleMarker(location= [56.130, -106.35], radius=1000, color='red', fill_opacity=0.5).add_to(world_map)</pre>	
Chorpleth	Create a choropleth map based on a GeoJSON file and a specified data column.	<pre>folium.Choropleth(geo_data='path/to/geojson_file', data=df, columns=['region', 'value_column'], key_on='feature.properties.id', fill_color='Y1GnBu', fill_opacity=0.7, line_opacity=0.2, legend_name='Legend').add_to(map)</pre>	<pre>world_map.choropleth(geo_data=world_geo, data=df_can, columns=['Country',    'Total'], key_on='feature.properties.name', fill_color='YlOrRd', fill_opacity=0.7,line_opacity=0.2, legend_name='Immigration to Canada')</pre>	
PyWaffle				
Waffle	Create a waffle chart based on values and categories.	<pre>plt.figure(FigureClass = Waffle,rows = 20, columns = 30, values = values) waffle_chart = waffle.Waffle(values=[value1, value2,], rows=n, columns=n)</pre>	<pre>plt.figure(FigureClass = Waffle,rows = 20, columns = 30, values = df_dsn['Total'], cmap_name = 'tab20', legend = {'labels': label,'loc': 'lower left', 'bbox_to_anchor':(0,-0.1),'ncol': 3})</pre>	

Denmark (3901) Norway (2327) Swed

Add a legend to the waffle chart. legend(loc='upper left', bbox\_to\_anchor=(1, 1))

Add a title to the waffle chart.

Title the waffle chart.

Add labels to the waffle chart.

Labels waffle\_chart.set\_labels(['Label 1', 'Label 2', ...])

WordCloud

Create a word

WordCloud cloud object based on text

wordcloud = WordCloud().generate(text\_data)

alice wc = WordCloud(background\_color='white', max\_words=2000, mask=alice\_mask, stopwords=stopwords) alice\_wc.generate(alice\_novel) plt.imshow(alice\_wc, interpolation='bilinear')



Generate the

word cloud Generate wordcloud.generate(text data) based on the

> text data. Display the word cloud

using **Display** plt.imshow(wordcloud, interpolation='bilinear') matplotlib or

other plotting libraries. Set various

options for the wordcloud = wordcloud (font\_path='path/to/font\_file', word cloud, **Options** 

such as font,

background\_color='white', colormap='Blues', mask=mask\_image, stopwords=stopwords).generate(text\_data) colors, mask.

and stopwords.

Seaborn

barplot

Create a bar plot to visualize the relationship

 $sns.barplot(x='x\_variable', y='y\_variable',$ between a data=dataframe)

categorical variable and a numeric

variable. Create a count plot to display

the frequency

countplot of each  $\verb|sns.countplot(x='category', data=dataframe)|\\$ 

category in a categorical variable. Create a scatter plot with a

linear regression line

regression line to visualize the v='y\_variable', data=dataframe) regplot

relationship between two numeric

variables.

sns.barplot(x='Continent', y='Total', data=df\_can1)

sns.countplot(x='Continent', data=df\_can)

sns.regplot(x='year', y='total',

data=df\_tot)

## Author(s)

Dr. Pooja

## Changelog

Version Changed by Change Description

2023-06-18 0.1 Initial version created Dr. Pooja