Pandas_plot_exercise

May 22, 2019

```
In [ ]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
```

```
In [ ]: plt.rcParams['figure.figsize'] = (15,5)
        plt.rcParams['font.size'] = 14
```

```
In [ ]: drink_cols = ['country', 'beer', 'spirit', 'wine', 'liters', 'continent']
        url = 'https://raw.githubusercontent.com/justmarkham/DAT8/master/data/drinks.csv'
        drinks = pd.read_csv(url, header=0, names=drink_cols, na_filter=False)
```

```
In [ ]: drinks.describe()
```

```
In [ ]: drinks.info()
```

```
In [ ]: drinks.head()
```

```
In [ ]: drinks1 = drinks.copy()
        drinks1.set_index(drinks['country'], inplace =True)
        del drinks1['country']
        del drinks1.index.name
        drinks1.head()
```

0.0.1 Line Plot

```
In [ ]: drinks['beer'].plot()

        # default lineploit
```

```
In [ ]: drinks1['beer'][:10].plot(kind='bar')
```

```
In [ ]: drinks[['beer','wine']].plot()
```

```
In [ ]: drinks.plot()
```

```
In [ ]: url = 'https://raw.githubusercontent.com/justmarkham/DAT8/master/data/ufo.csv'
        ufo = pd.read_csv(url)
        ufo['Time'] = pd.to_datetime(ufo.Time)
        ufo['Year'] = ufo.Time.dt.year
```

```
In [ ]: ufo.head()

In [ ]: # count the number of ufo reports each year (and sort by year)
        ufo.Year.value_counts().sort_index(ascending=False).head()

In [ ]: # compare with line plot
        ufo.Year.value_counts().sort_index().plot()

In [ ]: drinks.continent.value_counts().plot()
```

0.0.2 Histogram

```
In [ ]: drinks.beer.plot(kind='hist', bins = 100)

In [ ]: drinks.beer.plot(kind='hist', bins=20)
        plt.xlabel('Beer Servings')
        plt.ylabel('Frequency')
        plt.title('Histogram of Beer Servings')
        plt.savefig('histo.png')

In [ ]: # compare with density plot (smooth version of a histogram)
        drinks.beer.plot(kind='density', xlim=(0, 500))

In [ ]: # histogram of beer servings grouped by continent
        drinks.hist(column='beer', by='continent')

In [ ]: # share the x axes
        drinks.hist(column='beer', by='continent', sharex=True)
```

0.0.3 Scatter Plot

```
In [ ]: drinks[['beer', 'wine']].sort_values('beer').head()

In [ ]: drinks.plot(kind='scatter', x='beer', y='wine')

In [ ]: drinks.plot(kind='scatter', x='beer', y='wine', alpha=0.2)

In [ ]: # vary point color by spirit servings
        drinks.plot(kind='scatter', x='beer', y='wine', c='spirit', colormap='Blues')
```

0.0.4 Scatter-Matrix

```
In [ ]: pd.plotting.scatter_matrix(drinks)

In [ ]: pd.plotting.scatter_matrix(drinks[['beer', 'spirit', 'wine']])

In [ ]: pd.plotting.scatter_matrix(drinks[['beer', 'spirit', 'wine']], figsize=(5, 4))
```

0.0.5 Bar Plot

```
In [ ]: drinks.head(3)

In [ ]: drinks.continent.value_counts()

In [ ]: drinks.continent.value_counts().plot(kind='barh')

In [ ]: drinks.groupby('continent').mean()

In [ ]: # side-by-side bar plots
        drinks.groupby('continent').mean().plot(kind='bar')

In [ ]: # side-by-side bar plots
        drinks.groupby('continent').mean().plot(kind='bar', stacked=True)

In [ ]: # drop the liters column
        drinks.groupby('continent').mean().drop('liters', axis=1).plot(kind='bar')

In [ ]: drinks.groupby('continent').mean().drop('liters', axis=1).plot(kind='bar', stacked=Tru

In [ ]: drinks.spirit.sort_values().head()

In [ ]: drinks.head(3)

In [ ]: drink_country = drinks[['beer', 'wine']].sum(axis=1)
        drink_country.index=drinks.country
        drink_country10 = drink_country.sort_values(ascending=False)[:10]
        drink_country10.plot(kind='bar')

In [ ]: drinks.sum(axis=1).plot(kind='bar')

In [ ]: drinks[['beer','spirit','wine','liters']].sum(axis=0).plot(kind='bar')
```

0.0.6 Box-Plot

```
In [ ]: drinks.head()

In [ ]: drinks.plot(kind='box')

In [ ]: drinks.spirit.plot(kind='box')

In [ ]: # box plot of beer servings grouped by continent
        drinks.boxplot(column='beer', by='continent')

In [ ]: # box plot of beer servings grouped by continent
        drinks.groupby('continent').mean().T

In [ ]: # box plot of beer servings grouped by continent
        drinks.groupby('continent').mean().T.plot(kind='box')

In [ ]: # box plot of all numeric columns grouped by continent
        drinks.boxplot(by='continent')
```

0.0.7 Saving Plot

```
In [ ]: drinks.beer.plot(kind='hist', bins=20, title='Histogram of Beer Servings')
        plt.xlabel('Beer Servings')
        plt.ylabel('Frequency')
        plt.savefig('beer_histogram_original.png')
```