```
df = pd.read_csv('../input/train.csv')
df.head()
```

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Comment this if the data visualisations doesn't work on your side
%matplotlib inline

plt.style.use('bmh')
```

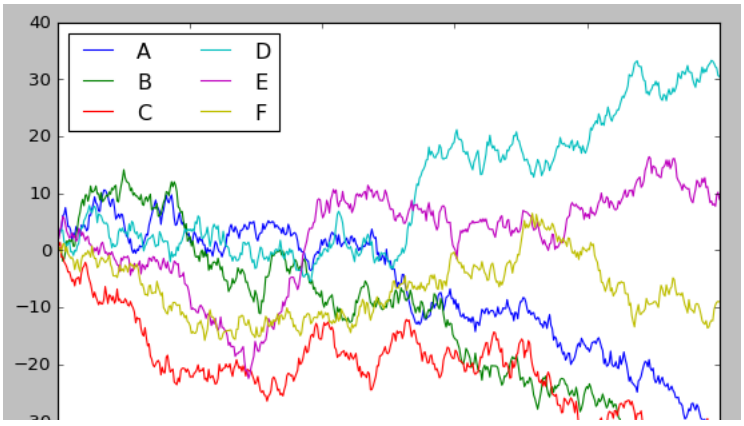
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

data = pd.read_csv("sample_data/california_housing_test.csv")
data.tail(10)
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
2990	-118.23	34.09	49.0	1638.0	456.0	1500.0	430.0	2.6923	150000.0
2991	-117.17	34.28	13.0	4867.0	718.0	780.0	250.0	7.1997	253800.0
2992	-122.33	37.39	52.0	573.0	102.0	232.0	92.0	6.2263	500001.0
2993	-117.91	33.60	37.0	2088.0	510.0	673.0	390.0	5.1048	500001.0
2994	-117.93	33.86	35.0	931.0	181.0	516.0	174.0	5.5867	182500.0
2995	-119.86	34.42	23.0	1450.0	642.0	1258.0	607.0	1.1790	225000.0
2996	-118.14	34.06	27.0	5257.0	1082.0	3496.0	1036.0	3.3906	237200.0
2997	-119.70	36.30	10.0	956.0	201.0	693.0	220.0	2.2895	62000.0
2998	-117.12	34.10	40.0	96.0	14.0	46.0	14.0	3.2708	162500.0
2999	-119.63	34.42	42.0	1765.0	263.0	753.0	260.0	8.5608	500001.0

```
import matplotlib.pyplot as plt
plt.style.use('classic')
%matplotlib inline
import numpy as np
import pandas as pd
```

```
# Create some data
rng = np.random.RandomState(0)
x = np.linspace(0, 10, 500)
y = np.cumsum(rng.randn(500, 6), 0)
```

```
# Plot the data with Matplotlib defaults
plt.plot(x, y)
plt.legend('ABCDEF', ncol=2, loc='upper left');
```

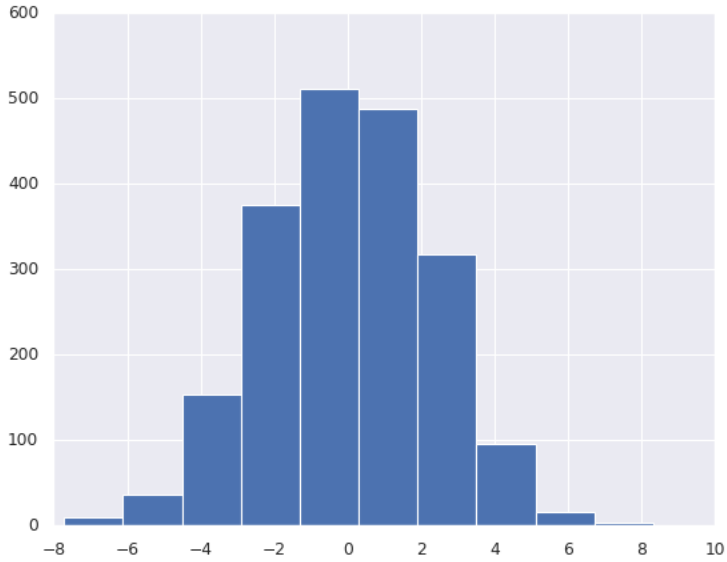


```
import seaborn as sns
sns.set()
```

```
# same plotting code as above!
plt.plot(x, y)
plt.legend('ABCDEF', ncol=2, loc='upper left');
```


```
data = np.random.multivariate_normal([0, 0], [[5, 2], [2, 2]], size=2000)
data = pd.DataFrame(data, columns=['x', 'y'])

for col in 'xy':
    plt.hist(data[col], normed=True, alpha=0.5)
```

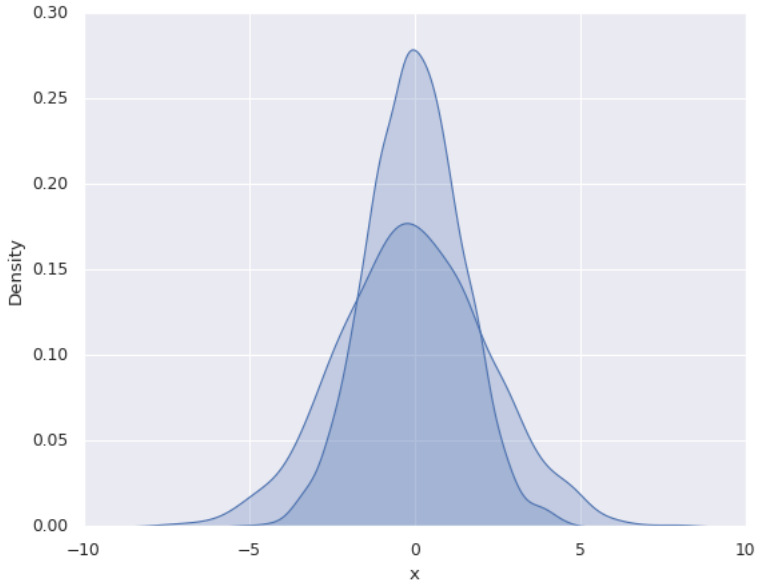
```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-43-7c25cafad82f> in <cell line: 4>()
      3
      4 for col in 'xy':
----> 5     plt.hist(data[col], normed=True, alpha=0.5)

 4 frames
/usr/local/lib/python3.10/dist-packages/matplotlib/artist.py in _update_props(self, props, errfmt)
   1195                     func = getattr(self, f"set_{k}", None)
   1196                     if not callable(func):
-> 1197                         raise AttributeError(
   1198                             errfmt.format(cls=type(self), prop_name=k))
   1199                     ret.append(func(v))

AttributeError: Rectangle.set() got an unexpected keyword argument 'normed'
```



Next steps: Explain error

```
for col in 'xy':
    sns.kdeplot(data[col], shade=True)
```

```
<ipython-input-24-929878336419>:2: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(data[col], shade=True)
<ipython-input-24-929878336419>:2: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(data[col], shade=True)
```




```
sns.distplot(data['x'])
sns.distplot(data['y']);
```

```
<ipython-input-25-f74e07db4076>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).
```

```
sns.kdeplot(data);
```