PANDAS

AND

GRAPHICS

A Numerical Dataset

object x_i	Height (H)	Weight (W)	Foot (F)	Label (L)
x_1	5.00	100	6	green
x_2	5.50	150	8	green
x_3	5.33	130	7	green
x_4	5.75	150	9	green
x_5	6.00	180	13	red
x_6	5.92	190	11	red
x_7	5.58	170	12	red
x_8	5.92	165	10	red

- $N = 8$ items
- $M = 3$ (unscaled) attributes

Code for the Dataset

```
import pandas as pd
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                          "Foot","Label"])
```

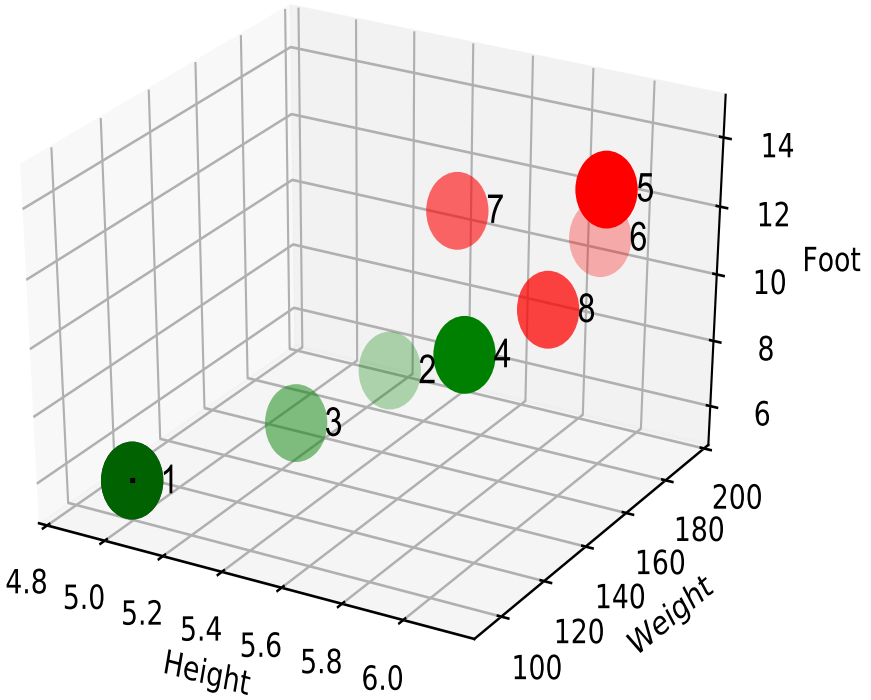
```
ipdb> data
  id Height Weight Foot Label
0  1   5.00    100    6  green
1  2   5.50    150    8  green
2  3   5.33    130    7  green
3  4   5.75    150    9  green
4  5   6.00    180   13    red
5  6   5.92    190   11    red
6  7   5.58    170   12    red
7  8   5.92    165   10    red
```

Desribing the Dataset

```
import pandas as pd
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                          "Foot","Label"])
```

```
ipdb> data.describe()
             id    Height      Weight      Foot
count  8.00000  8.000000    8.000000   8.00000
mean   4.50000  5.625000  154.375000   9.50000
std    2.44949  0.343428   28.962722   2.44949
min    1.00000  5.000000  100.000000   6.00000
25%    2.75000  5.457500  145.000000   7.75000
50%    4.50000  5.665000  157.500000   9.50000
75%    6.25000  5.920000  172.500000  11.25000
max    8.00000  6.000000  190.000000  13.00000
```

A Dataset Illustration

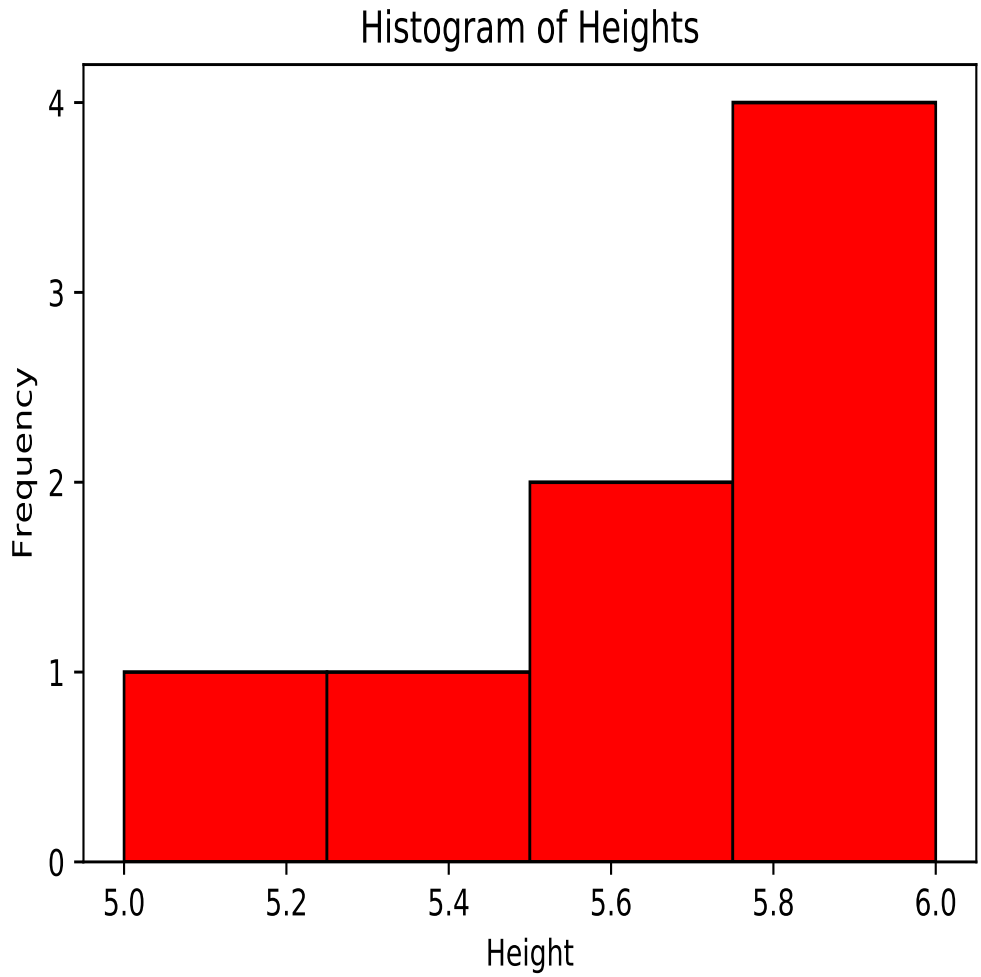


A Simple Histogram

```
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib.ticker import MaxNLocator

data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                          "Foot","Label"])
fig = plt.figure()
axes1 = fig.add_subplot(1,1,1)
axes1.hist(data["Height"], bins = 4,
  histtype='bar',ec="black", color="red")
axes1.set_title("Histogram of Heights")
axes1.set_xlabel("Height")
axes1.set_ylabel("Frequency")
axes1.yaxis.\
 set_major_locator(MaxNLocator(integer=True))
fig.show()
```

Histogram Illustration



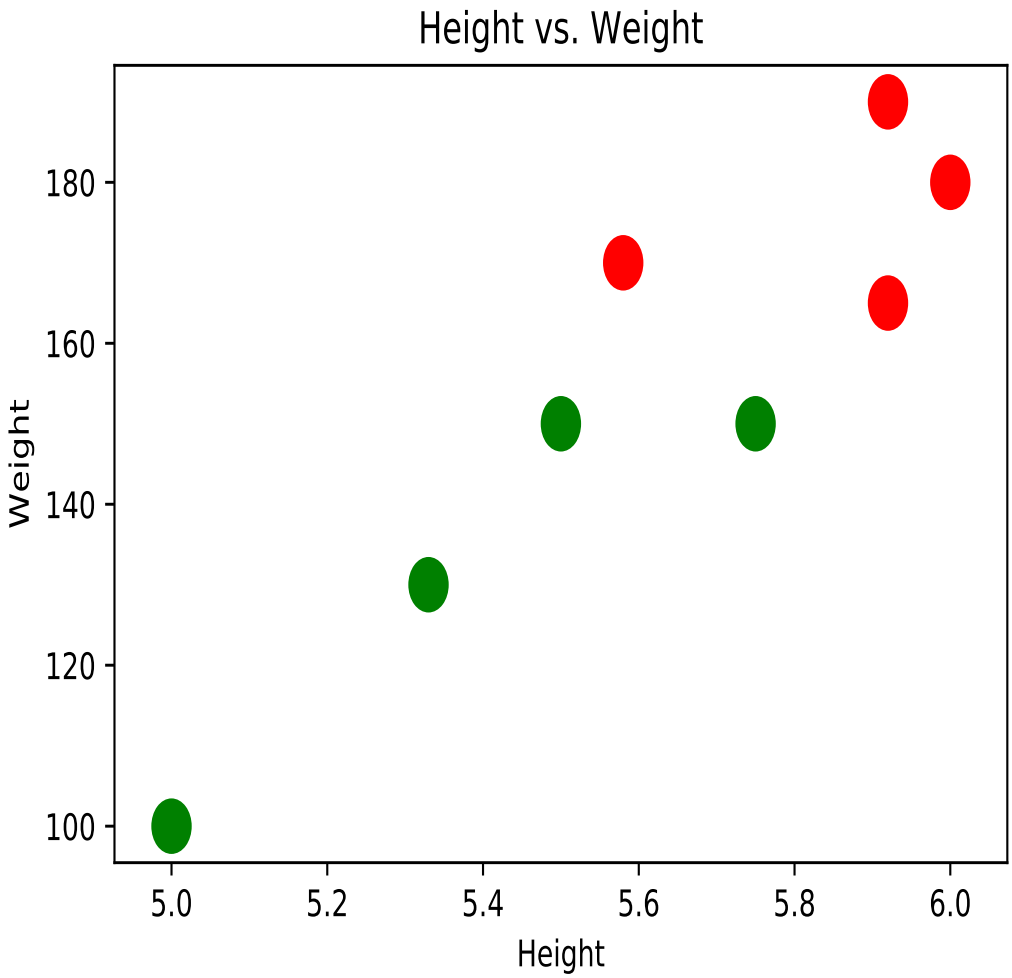
A Simple Scatter Plot

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


data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                           "Foot","Label"])

scatter_plot = plt.figure()
axes1 = scatter_plot.add_subplot(1,1,1)
axes1.scatter(data["Height"], data["Weight"],
               color=data["Label"], s=200)
axes1.set_title("Height vs. Weight")
axes1.set_xlabel("Height")
axes1.set_ylabel("Weight")
scatter_plot.show()
```

A Scatterplot Illustration



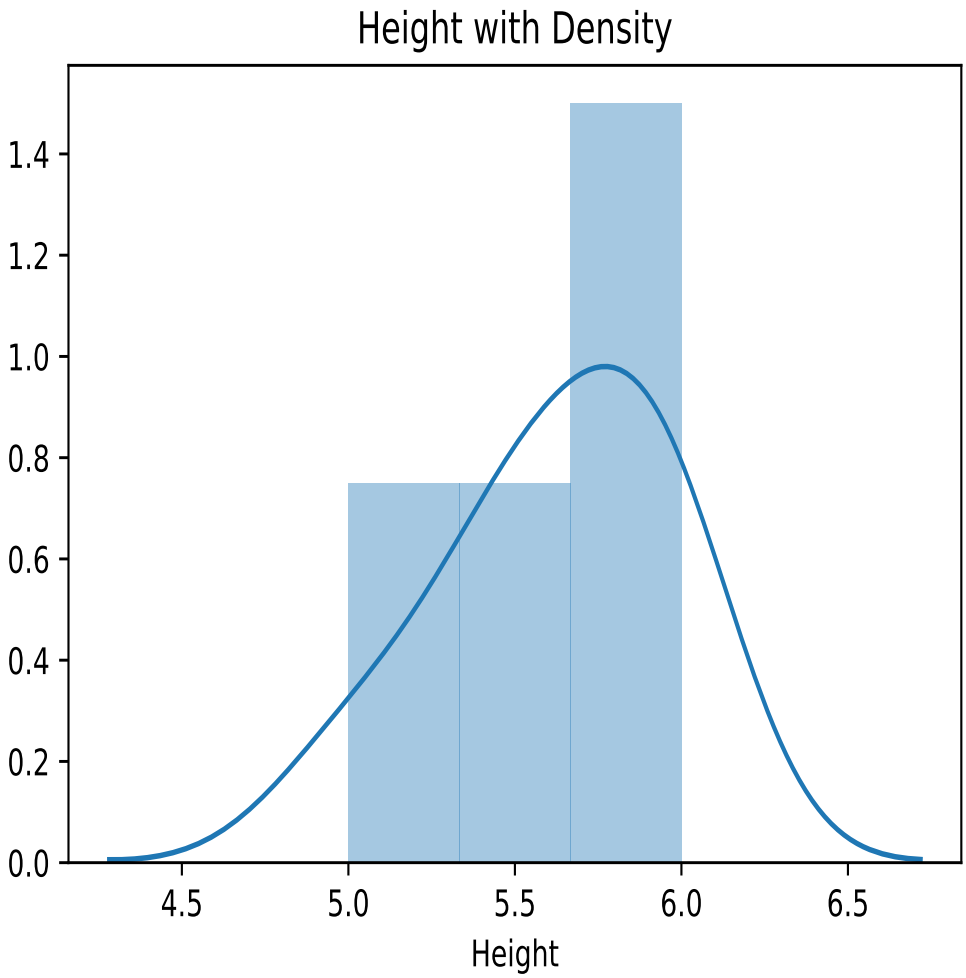
Histogram With Density

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

data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                          "Foot","Label"])

hist, ax = plt.subplots()
ax = sns.distplot(data["Height"])
ax.set_title("Height with Density")
plt.show()
```

Histogram with Density Illustration

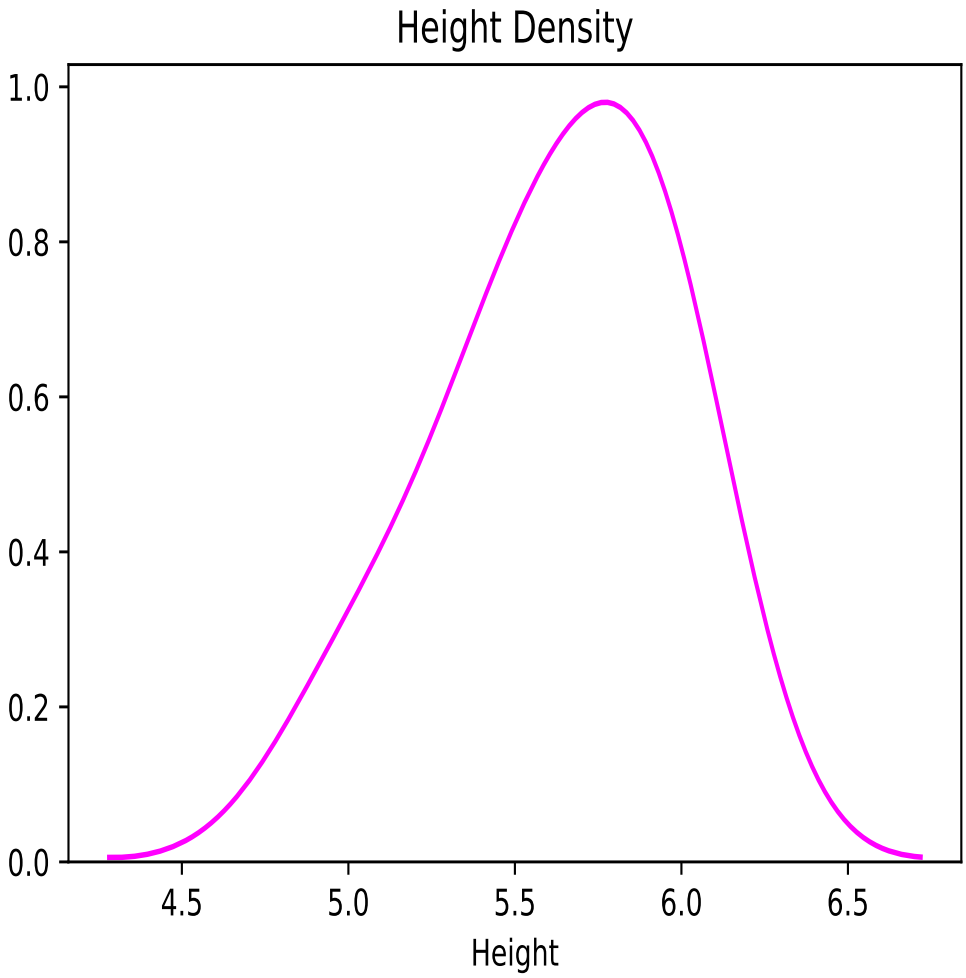


Density

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

data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                           "Foot","Label"])
hist, ax = plt.subplots()
ax=sns.distplot(data["Height"],
     hist=False, color="magenta")
ax.set_title("Height Density")
plt.show()
```

Density Illustration

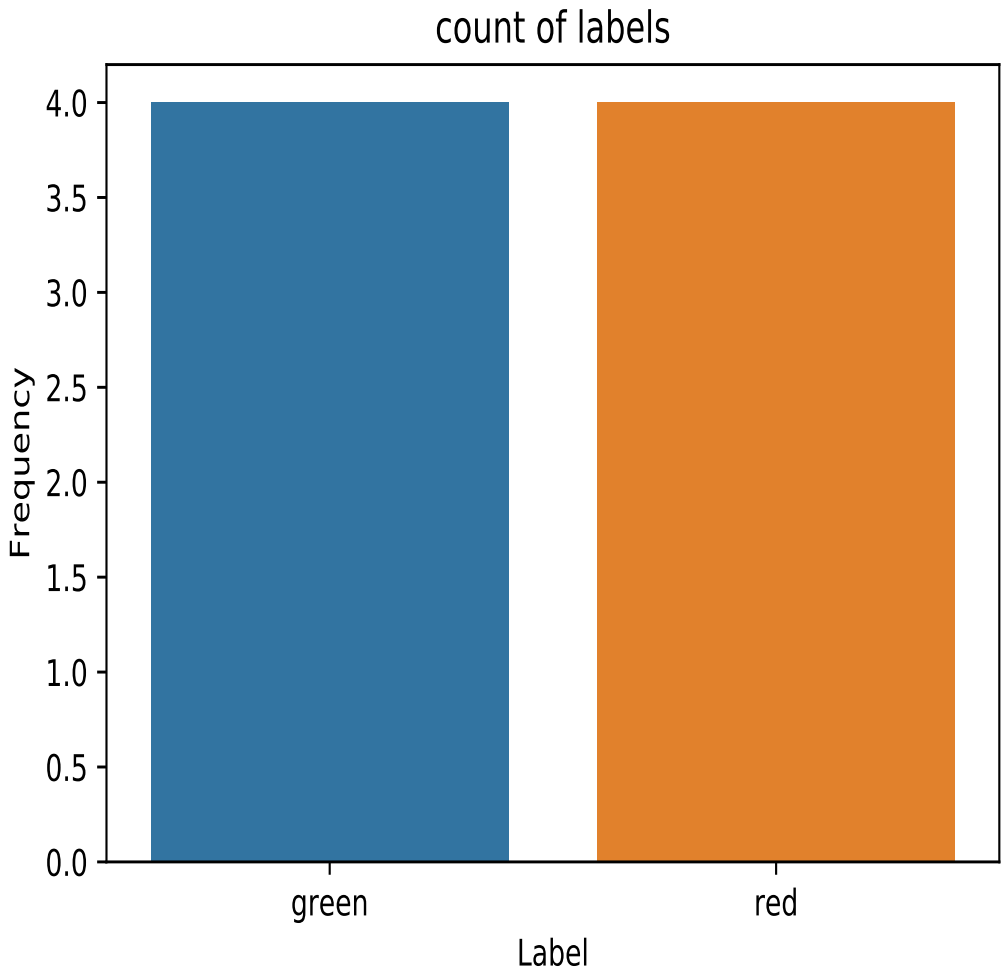


Counting

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

data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                           "Foot","Label"])
count, ax = plt.subplots()
ax = sns.countplot("Label", data=data)
ax.set_title("count of labels")
ax.set_xlabel("Label")
ax.set_ylabel("Frequency")
axes1.yaxis.set_major_locator
      (MaxNLocator(integer=True))
plt.show()
```

Counting Illustration



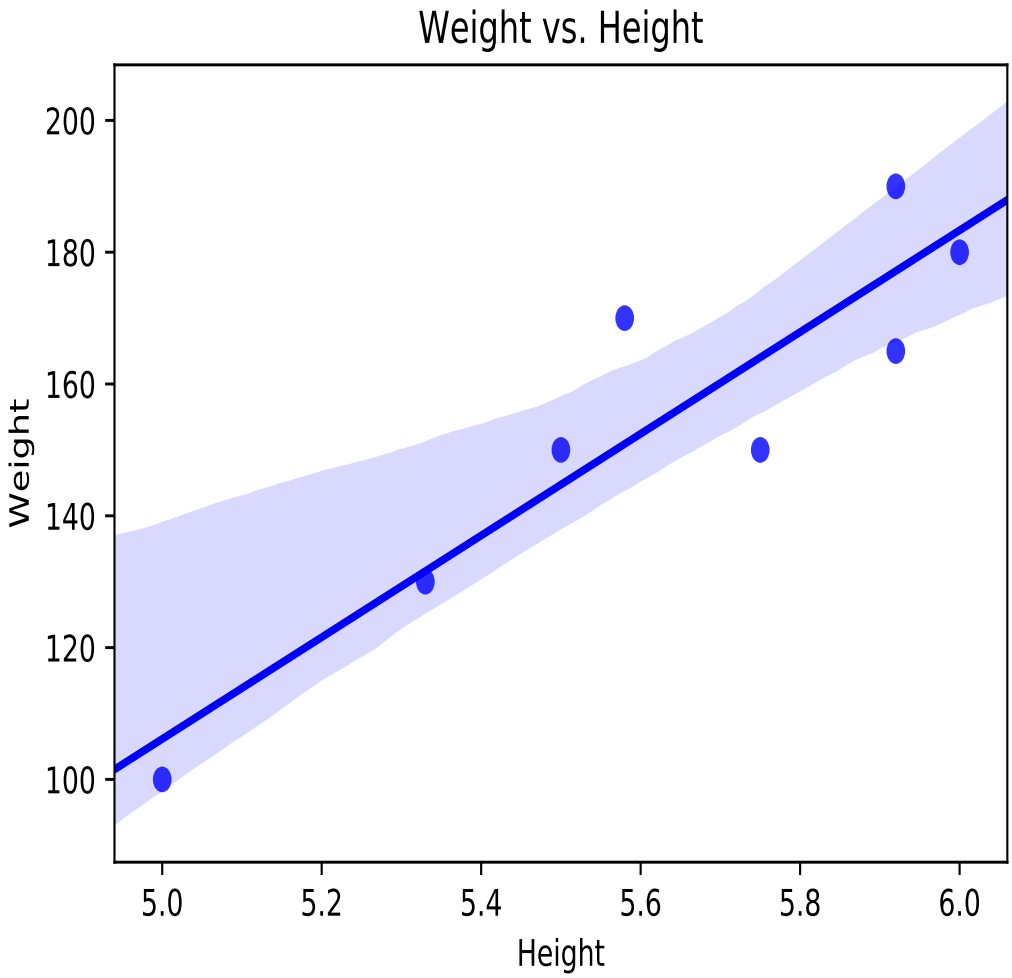
Scatterplot With Regression

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

data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                          "Foot","Label"])

scatter, ax = plt.subplots()
ax = sns.regplot(x="Height", y="Weight",
                 data=data, color="blue")
ax.set_title("Weight vs. Height")
ax.set_xlabel("Height")
ax.set_ylabel("Weight")
plt.show()
```

Scatterplot with Regression Illustration

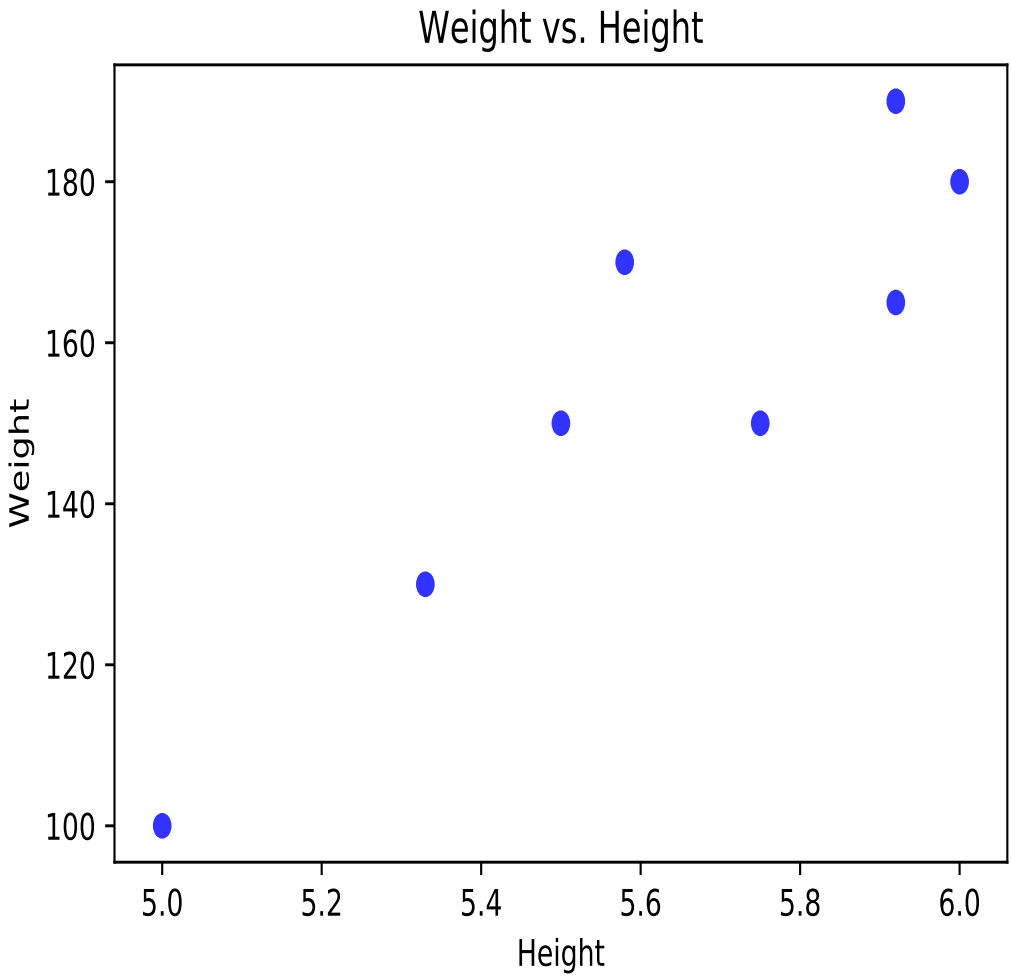


Scatterplot Without Regression

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                          "Foot","Label"])

scatter, ax = plt.subplots()
ax = sns.regplot(x="Height", y="Weight",
      data=data,color="blue", fit_reg=False)
ax.set_title("Weight vs. Height")
ax.set_xlabel("Height")
ax.set_ylabel("Weight")
plt.show()
```

Scatterplot Without Regression

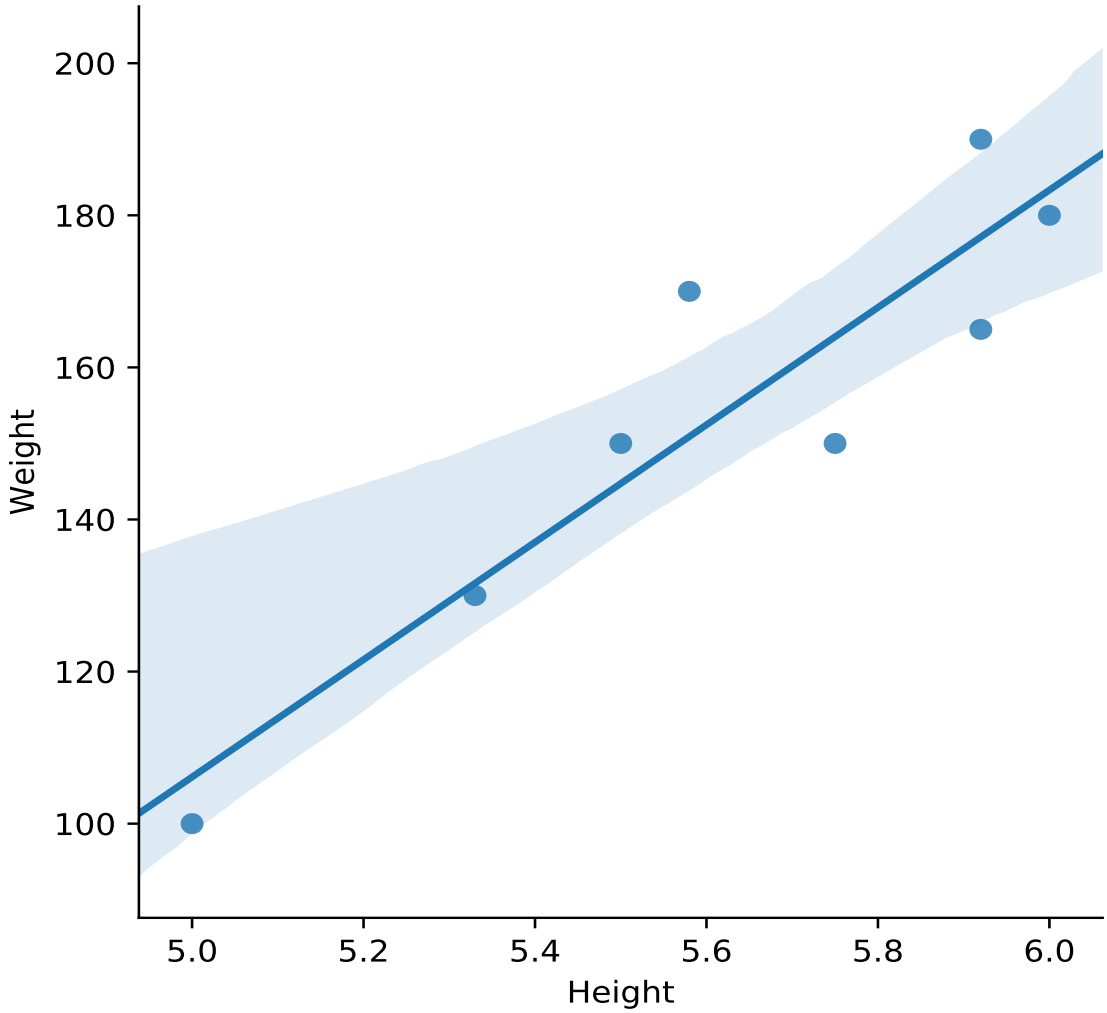


Scatterplot with Regression

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                           "Foot","Label"])

fig = sns.lmplot(x="Height", y="Weight",
                  data=data)
plt.show()
```

Illustration

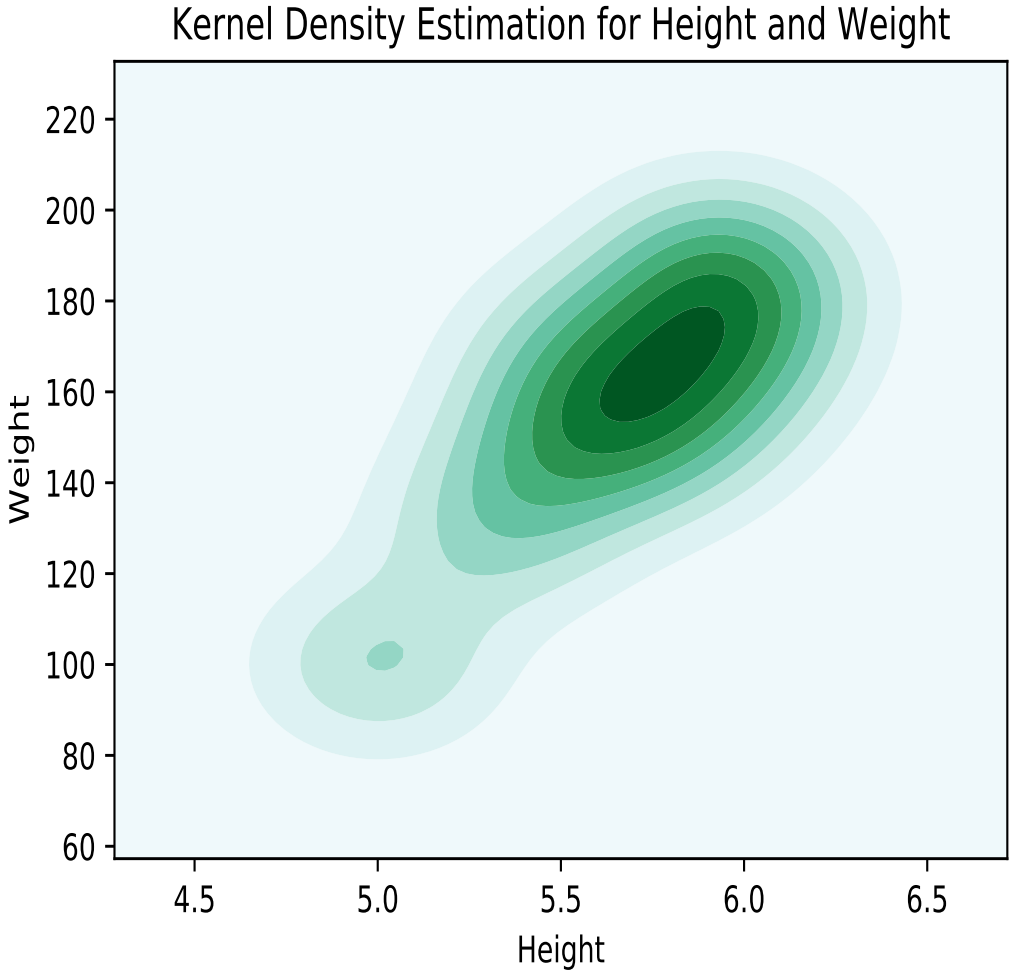


Density for Two Variables

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                   "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                   "Foot","Label"])

kde, ax = plt.subplots()
ax = sns.kdeplot(data=data["Height"],
      data2=data["Weight"], shade=True)
ax.set_title("Kernel Density Estimation \
                for Height and Weight")
ax.set_xlabel("Height")
ax.set_ylabel("Weight")
plt.show()
```

Density for Two Variables

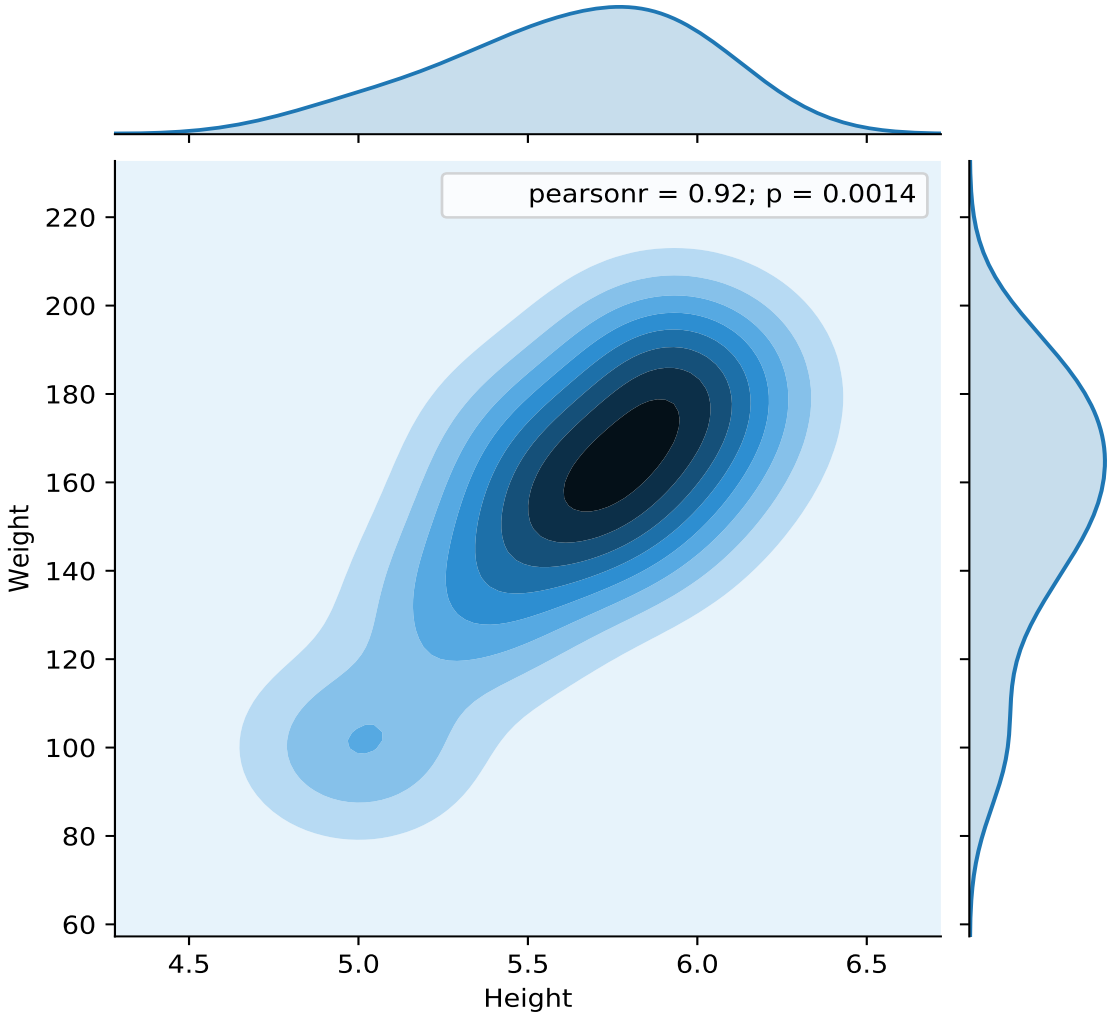


Joint Density

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                           "Foot","Label"])

kde_joint = sns.jointplot(x="Height",
    y="Weight", data=data, kind="kde")
```

Joint Density

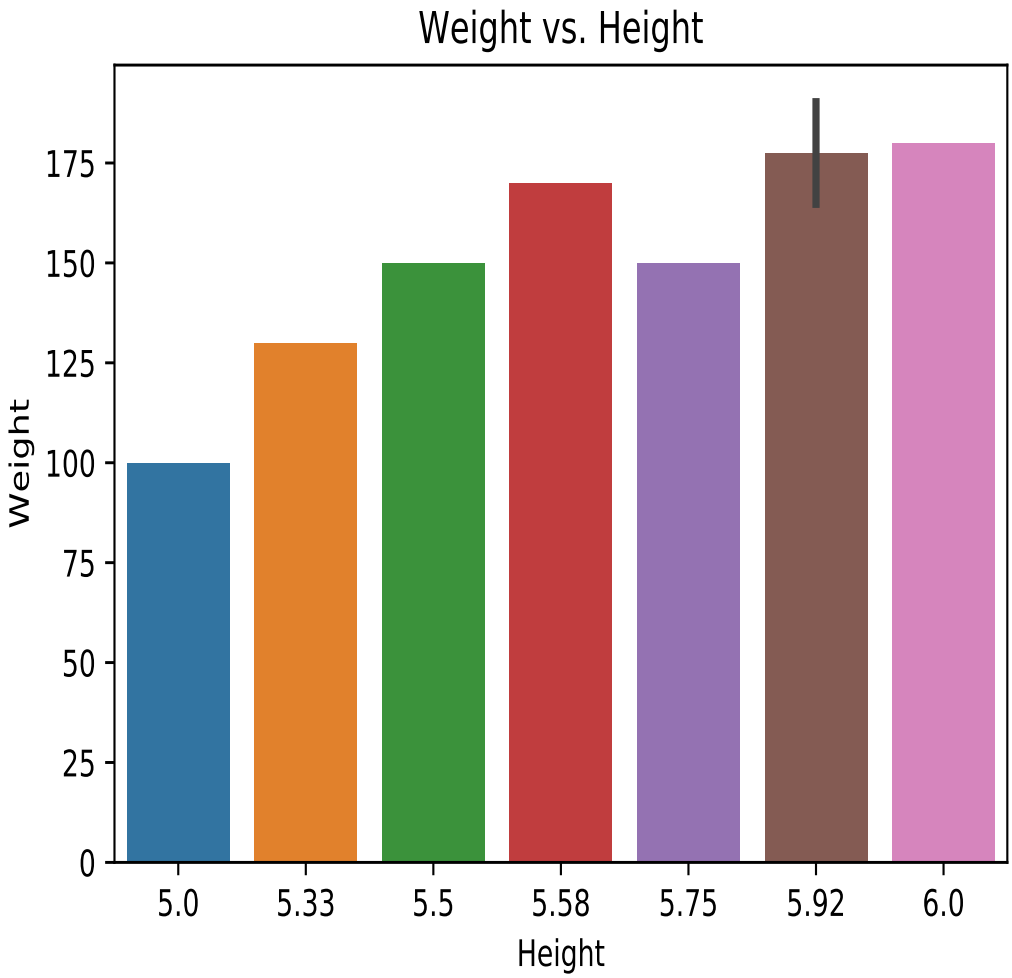


Bar Plots

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",

bar, ax = plt.subplots()
ax = sns.barplot(x="Height",y="Weight",data=data)
ax.set_title("Weight vs. Height")
ax.set_xlabel("Height")
ax.set_ylabel("Weight")
plt.show()
```

Bar Plots

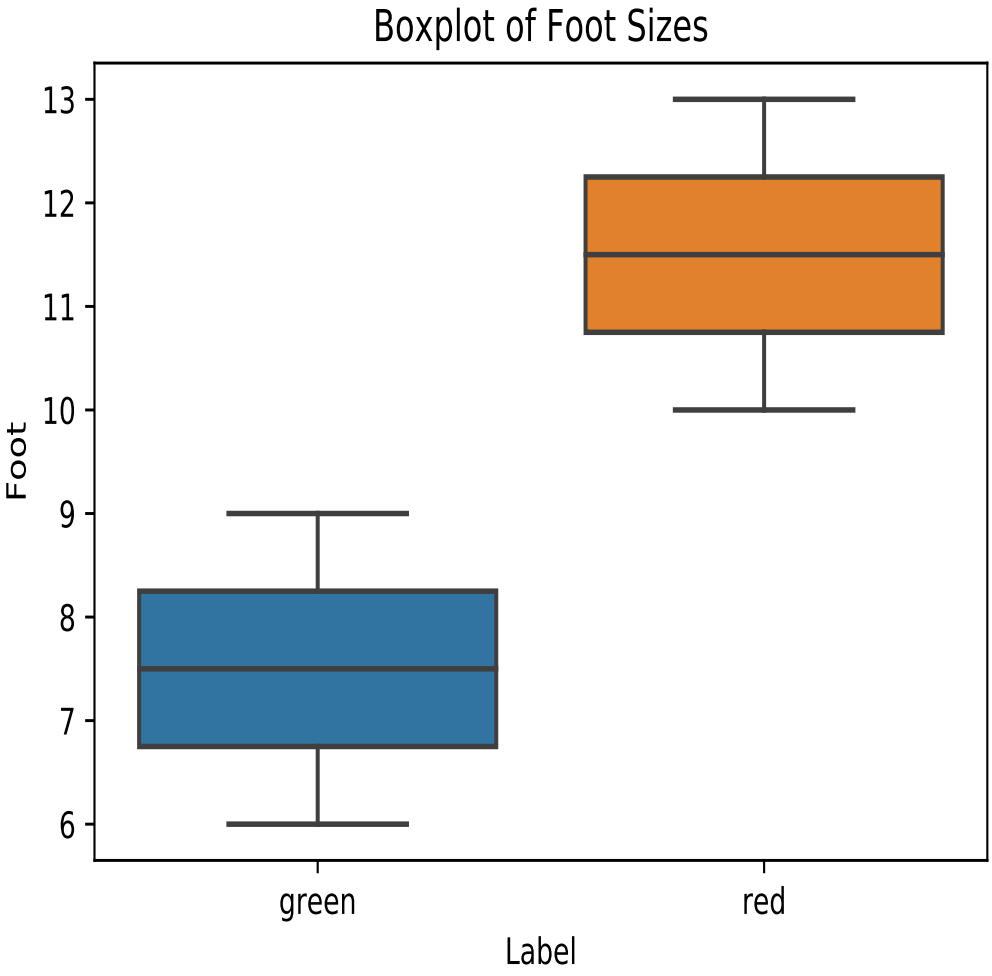

Weight vs. Height
Weight
0
25
50
75
100
125
150
175
5.0
5.33
5.5
5.58
5.75
5.92
6.0
Height

Box Plots

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                   "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                          "Foot","Label"])


box, ax = plt.subplots()
ax = sns.boxplot(x="Label",y="Foot",data=data)
ax.set_title("Boxplot of Foot Sizes")
ax.set_xlabel("Label")
ax.set_ylabel("Foot")
plt.show()
```

Box Plots

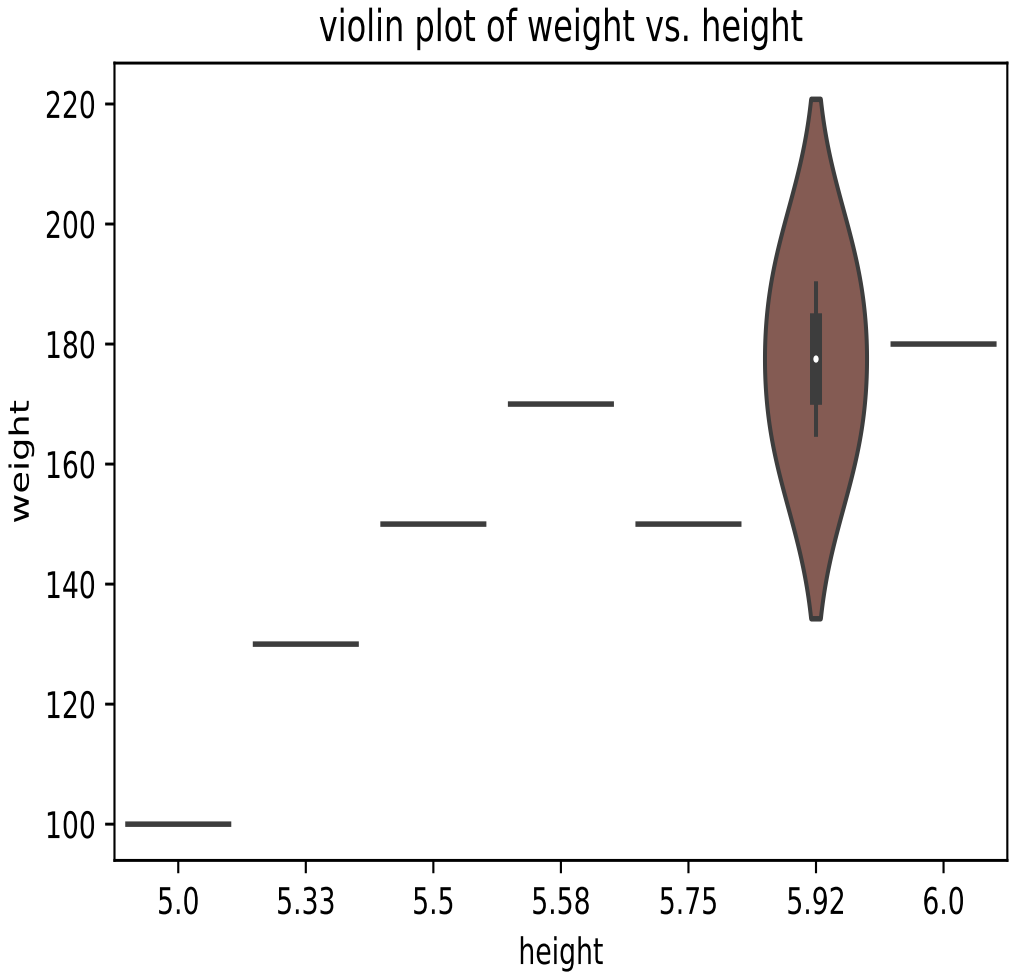


Violin Plots

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                          "Foot","Label"])


violin, ax = plt.subplots()
ax = sns.violinplot(x="Height",
                       y="Weight",data=data)
ax.set_title("violin plot of weight vs. height")
ax.set_xlabel("height")
ax.set_ylabel("weight")
plt.show()
```

Violin Plots

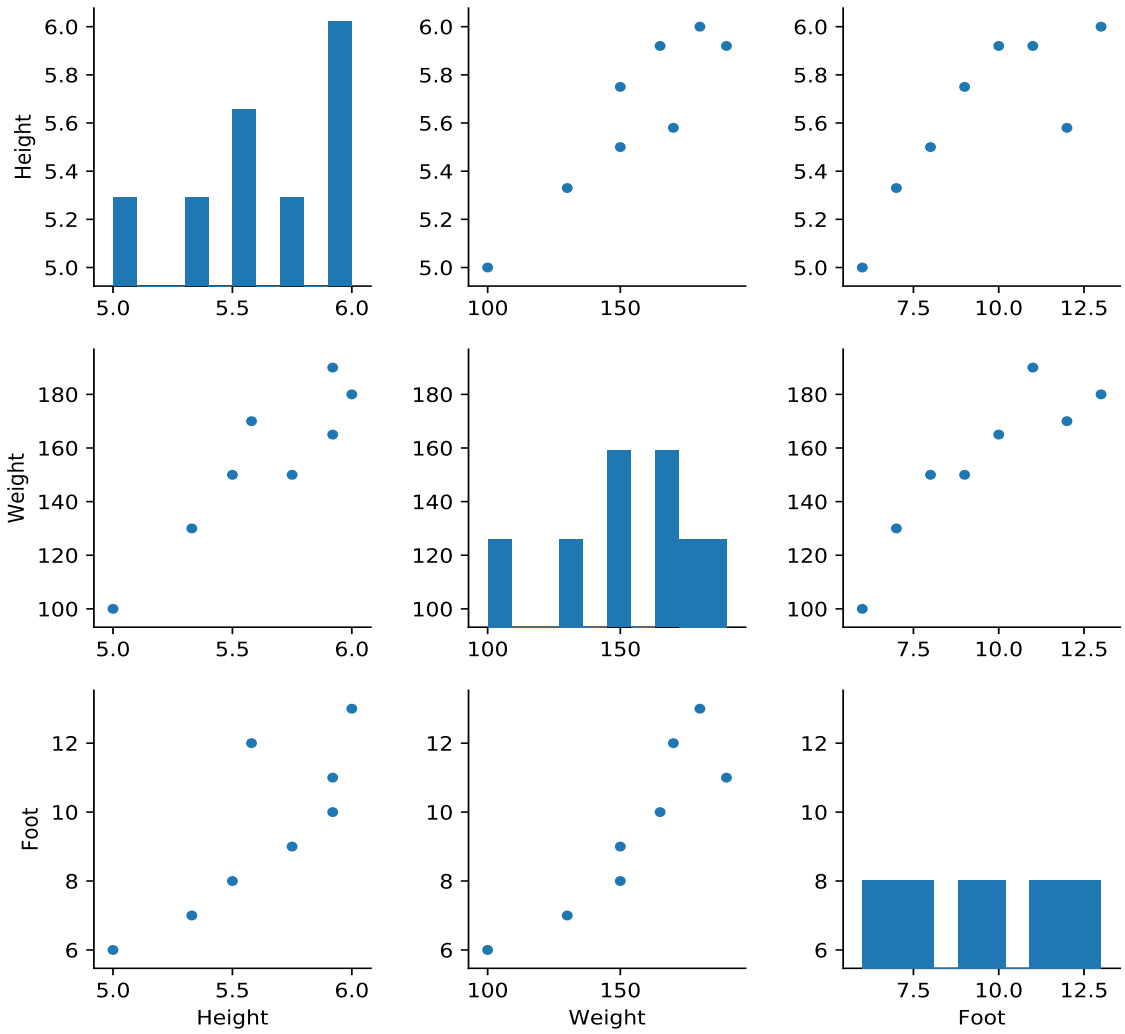


Pairwise Relationships

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                   "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                          "Foot","Label"])


pair_plot = sns.pairplot(data[["Height",
                       "Weight","Foot"]])
plt.show()
```

Pairwise Relationships

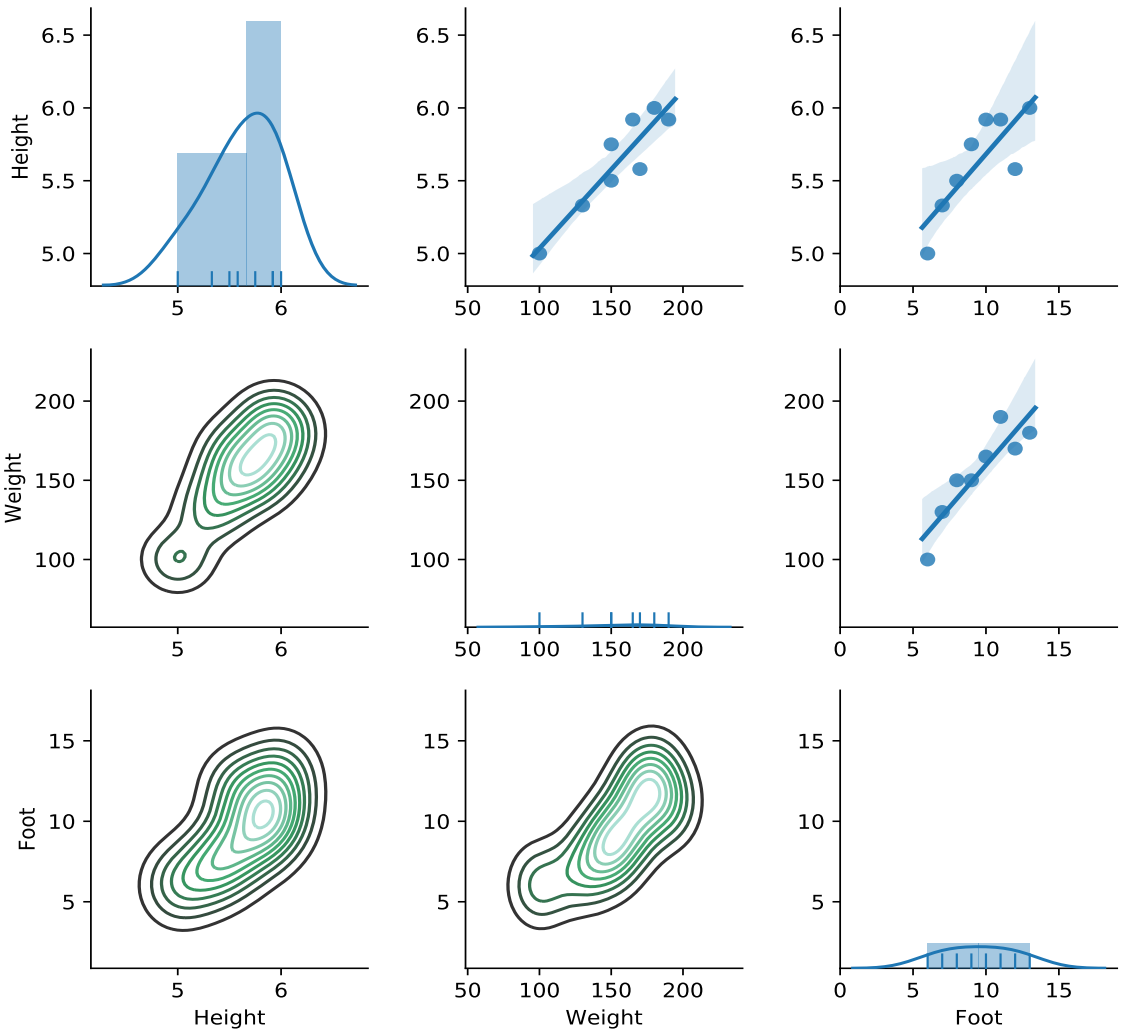


Specific Pairwise relationships

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                   "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                          "Foot","Label"])

pair_grid = sns.PairGrid(data[["Height",
                  "Weight","Foot"]])
pair_grid = pair_grid.map_upper(sns.regplot)
pair_grid = pair_grid.map_lower(sns.kdeplot)
pair_grid = pair_grid.map_diag(sns.distplot,
                        rug=True)
plt.show()
```

Specific Relationships

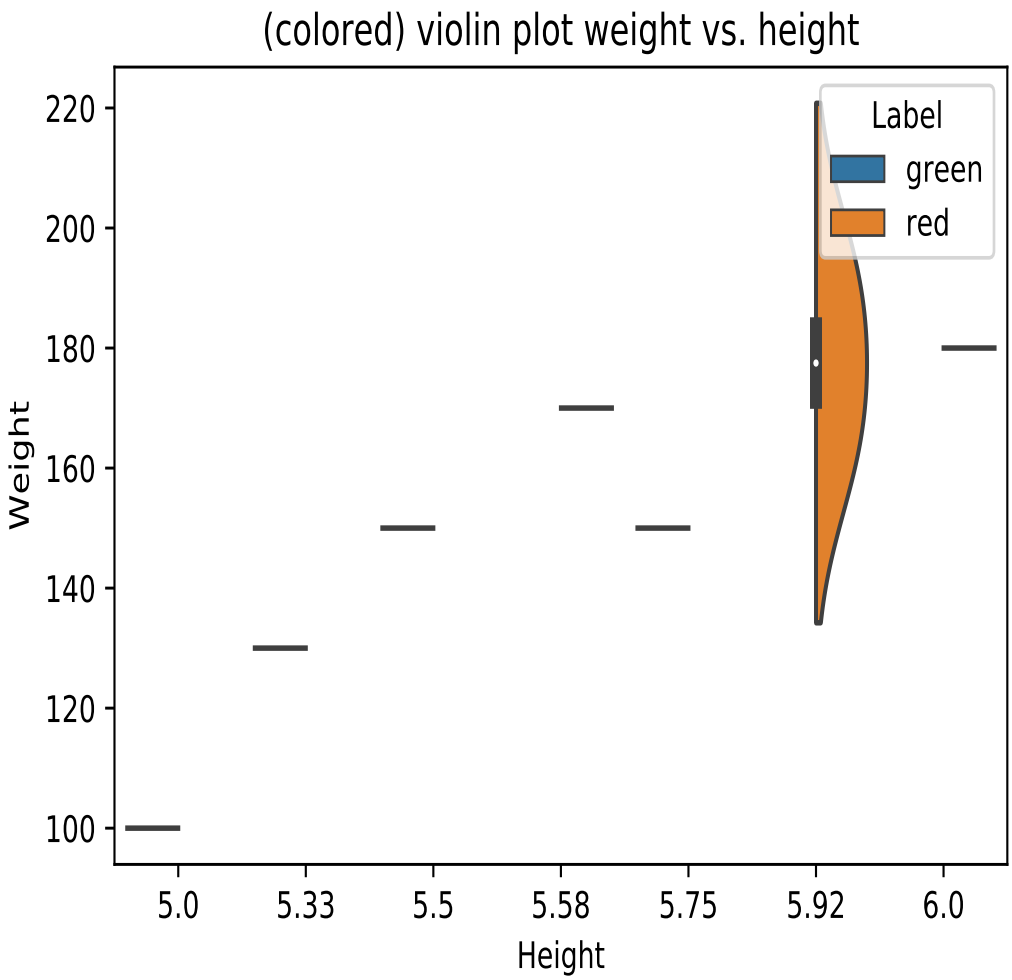


Colored Violin Plot

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                           "Foot","Label"])

colored_violin, ax = plt.subplots()
ax = sns.violinplot(x="Height",y="Weight",
        hue="Label", data=data, split=True)
ax.set_title("(colored) violin plot \
                    weight vs. height")
ax.set_xlabel("Height")
ax.set_ylabel("Weight")
plt.show()
```

Colored Violin Plot

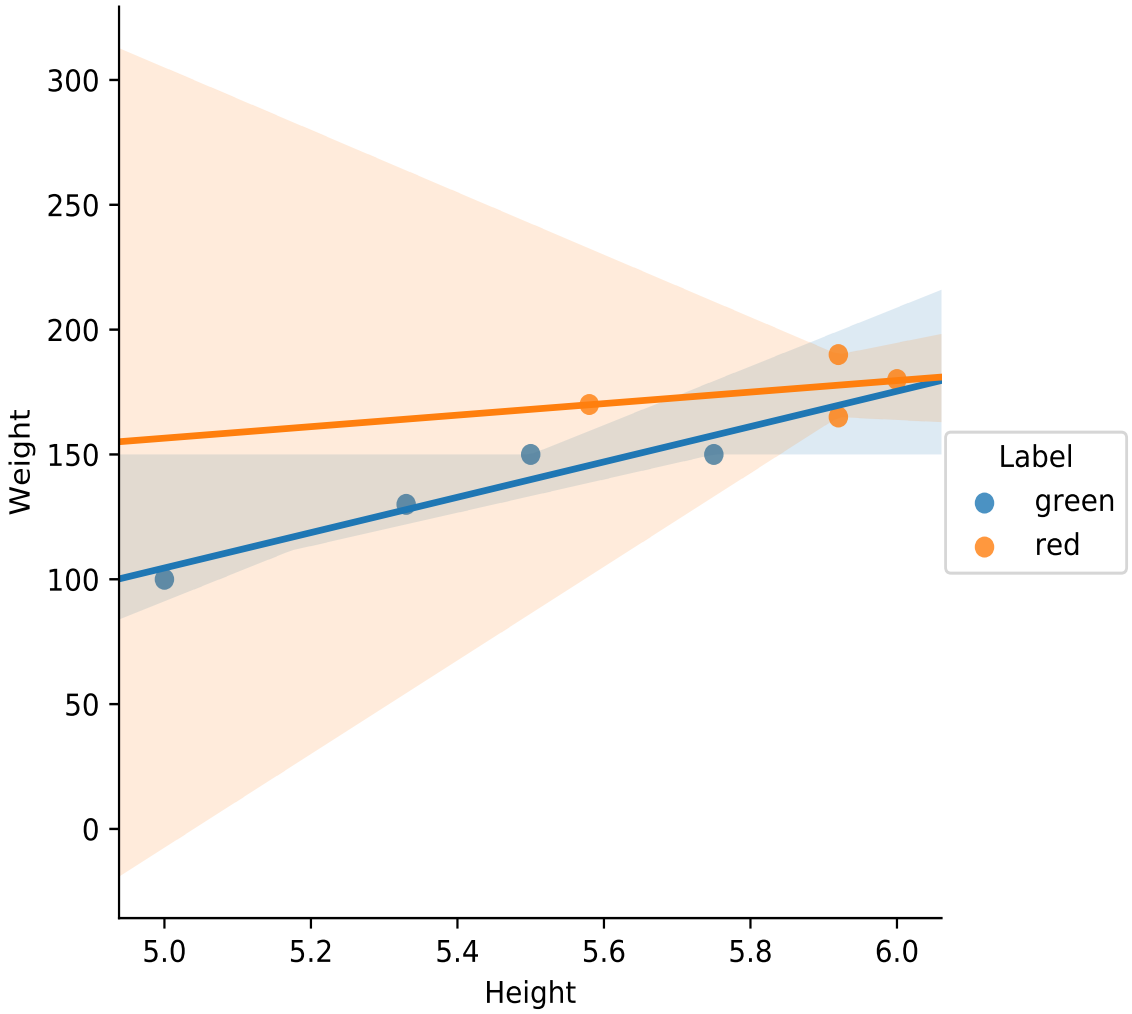


Regression Plot by Label

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                    "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                           "Foot","Label"])

fig = sns.lmplot(x="Height",y="Weight",
        hue="Label",data=data,fit_reg=True)
plt.show()
```

Regression Plot By Label

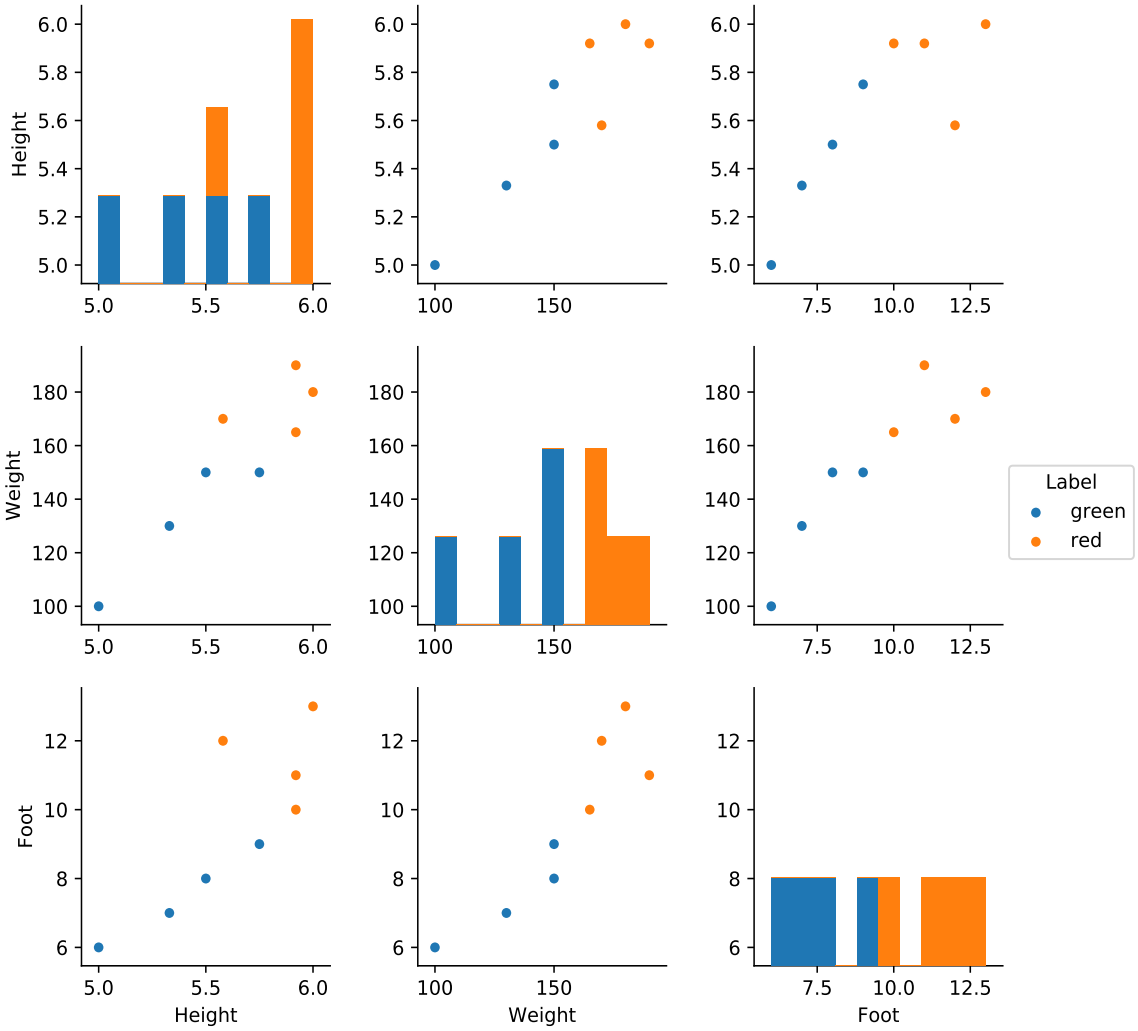


Colored Pair Plots

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
data = pd.DataFrame(
 {"id":[ 1,2,3,4,5,6,7,8],
  "Label":["green","green","green","green",
                   "red","red","red","red"],
  "Height":[5,5.5,5.33,5.75,6.00,5.92,5.58,5.92],
  "Weight":[100,150,130,150,180,190,170,165],
  "Foot":[6, 8, 7, 9, 13, 11, 12, 10]},
  columns=["id","Height","Weight",
                          "Foot","Label"])

fig=sns.pairplot(data[["Height","Weight",
           "Foot","Label"]], hue="Label")
plt.show()
```

Colored Pair Plot



Concepts Check:

(a) histogram

(b) scatter plot

(c) density

(d) multi-variate density

(e) counting

(f) bar and violin plots

(g) pair plots