lab0-visualization-auto_mpg

September 26, 2022

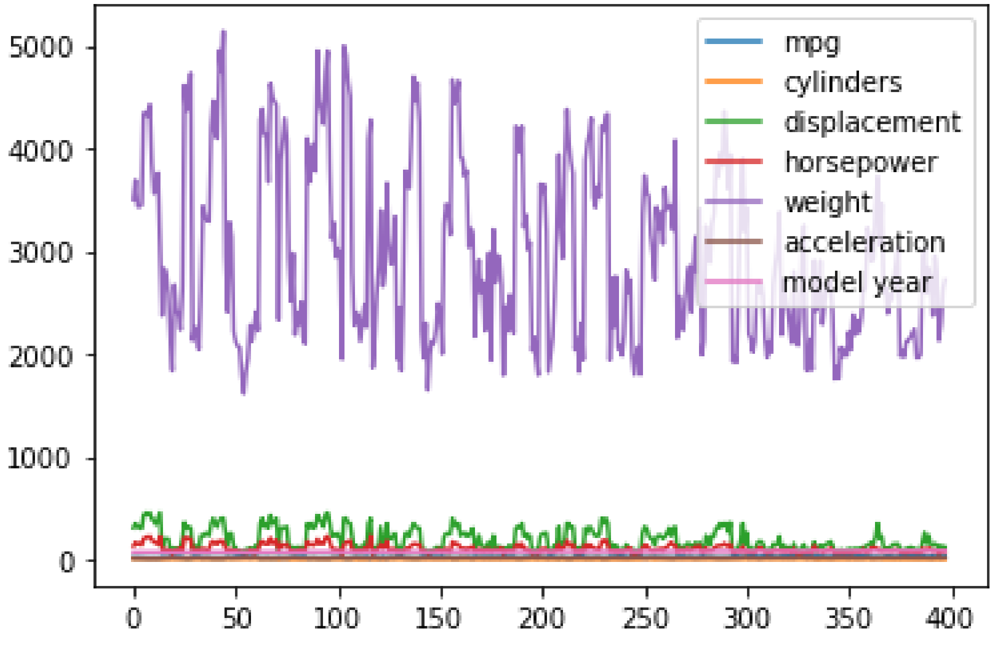
[1]:
```
%matplotlib inline

import numpy as np
import pandas as pd
import matplotlib as mpl
import matplotlib.pyplot as plt
```

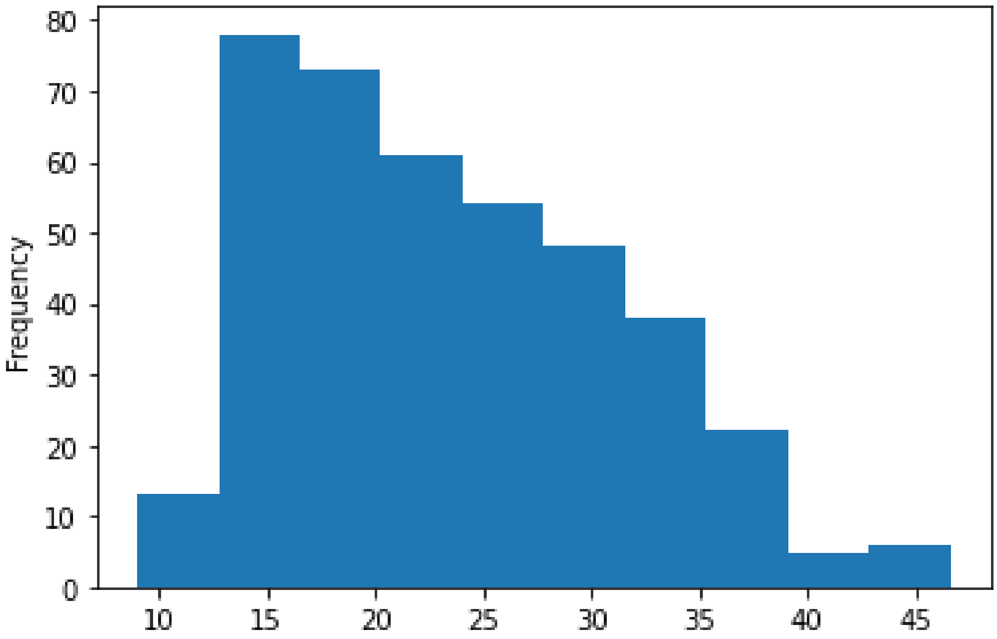
[2]:
```
data = pd.read_fwf('auto-mpg.data', na_values="?",␣
 ↪names=['mpg','cylinders','displacement','horsepower','weight',␣
 ↪'acceleration', 'model year', 'origin', 'car name'])
data['origin'] = data['origin'].astype('category')
```

[3]:
```
data.plot()
```

[3]:
```
<AxesSubplot:>
```



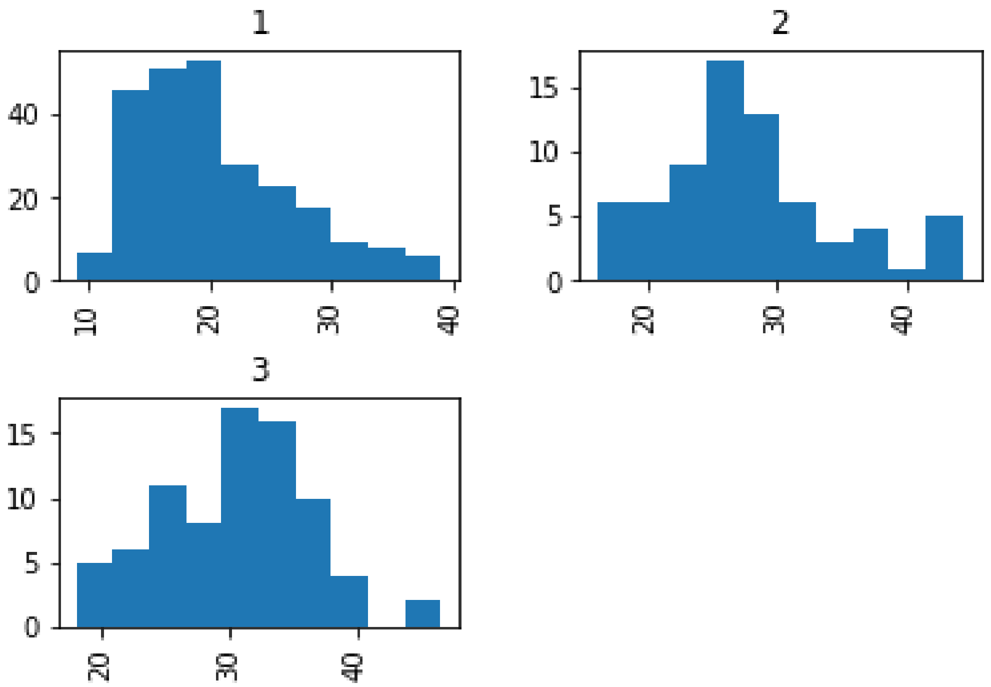
[4]:
```
data['mpg'].plot.hist();
```



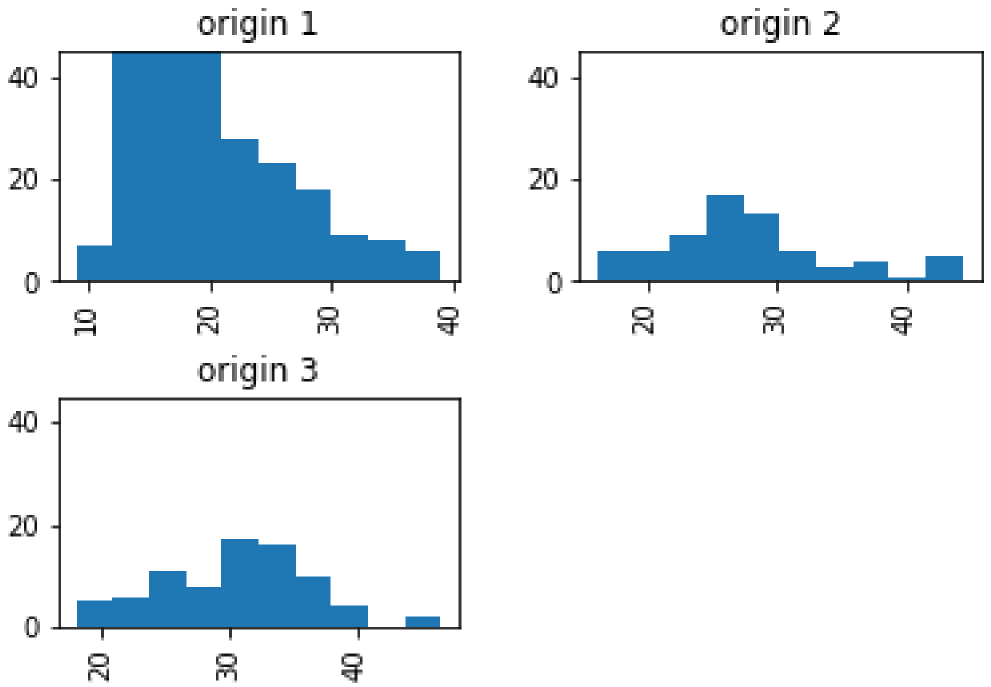
[5]:
```
data.origin.value_counts()
```

[5]:
```
1    249
3     79
2     70
Name: origin, dtype: int64
```

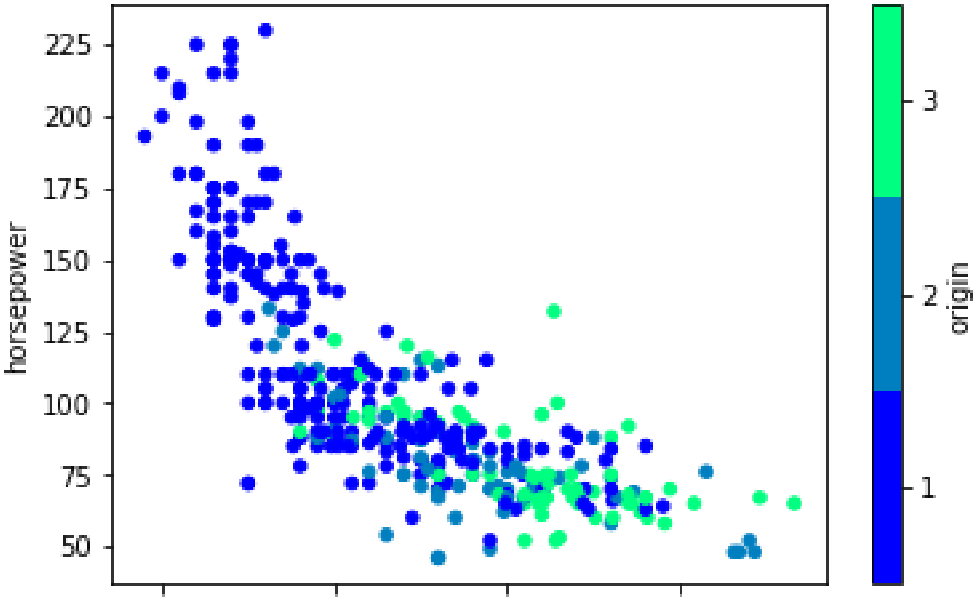
[6]:
```
axs = data.hist(column='mpg', by='origin')
```



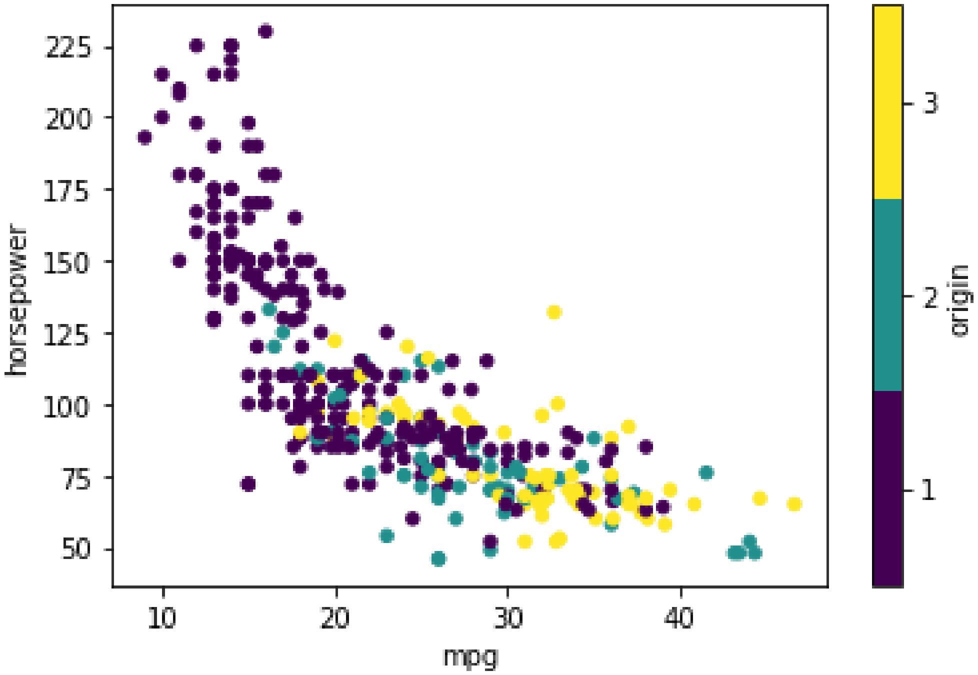
[7]:
```python
axs = data.hist(column='mpg', by='origin')
axs[0][0].set(title='origin 1', ylim=[0, 45])
axs[0][1].set(title='origin 2', ylim=[0, 45])
axs[1][0].set(title='origin 3', ylim=[0, 45]);
```



```
[8]: data.plot.scatter('mpg', 'horsepower', c='origin', colormap='winter');
```

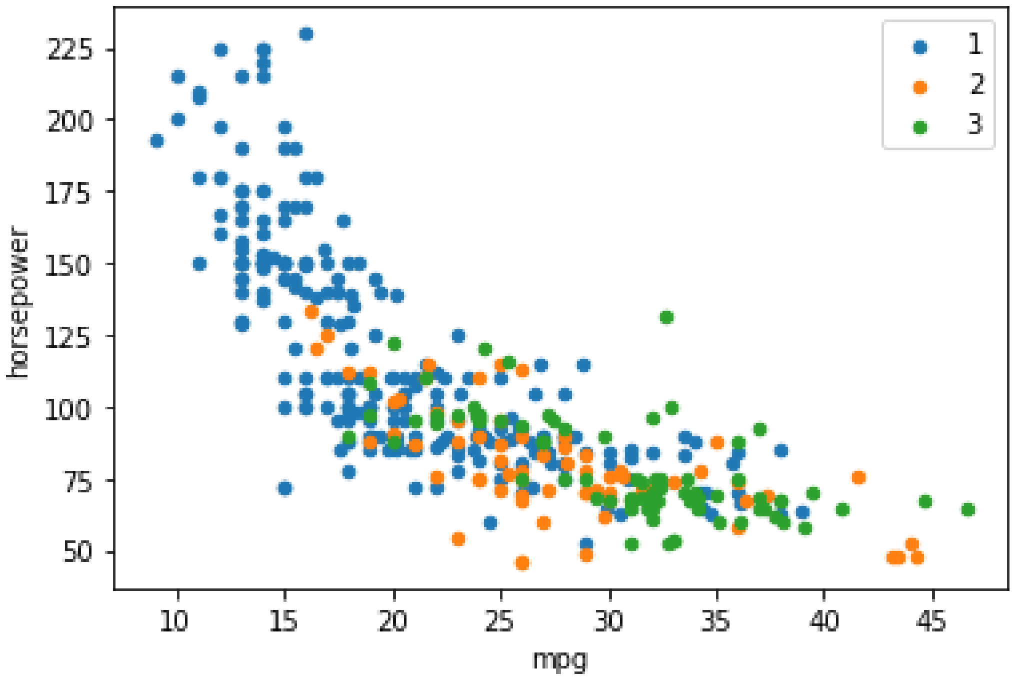


```
[9]: fig, ax = plt.subplots()
     data.plot.scatter('mpg', 'horsepower', c='origin', colormap='viridis', ax=ax);
```



[10]:

```
colors = {1: 'tab:blue', 2: 'tab:orange', 3: 'tab:green'}
fig, ax = plt.subplots()
for key, group in data.groupby(by='origin'):
    group.plot.scatter('mpg', 'horsepower', c=colors[key], label=key, ax=ax);
```

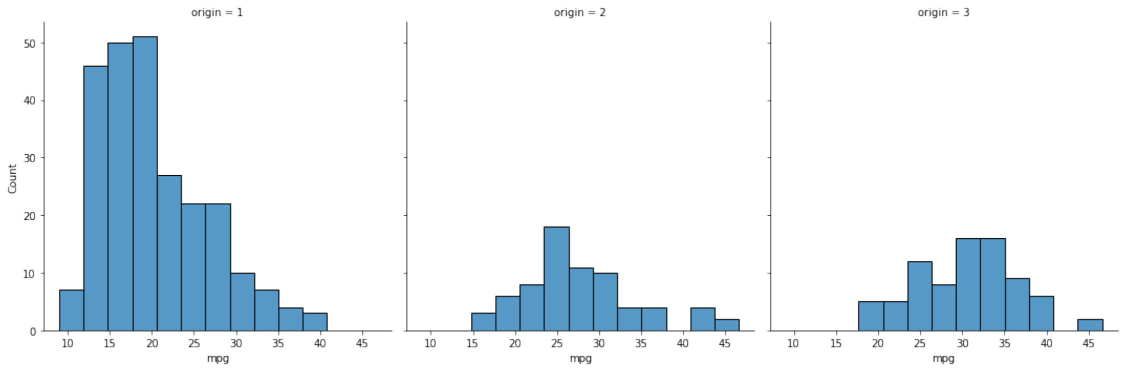


[11]:

```
import seaborn as sns
```

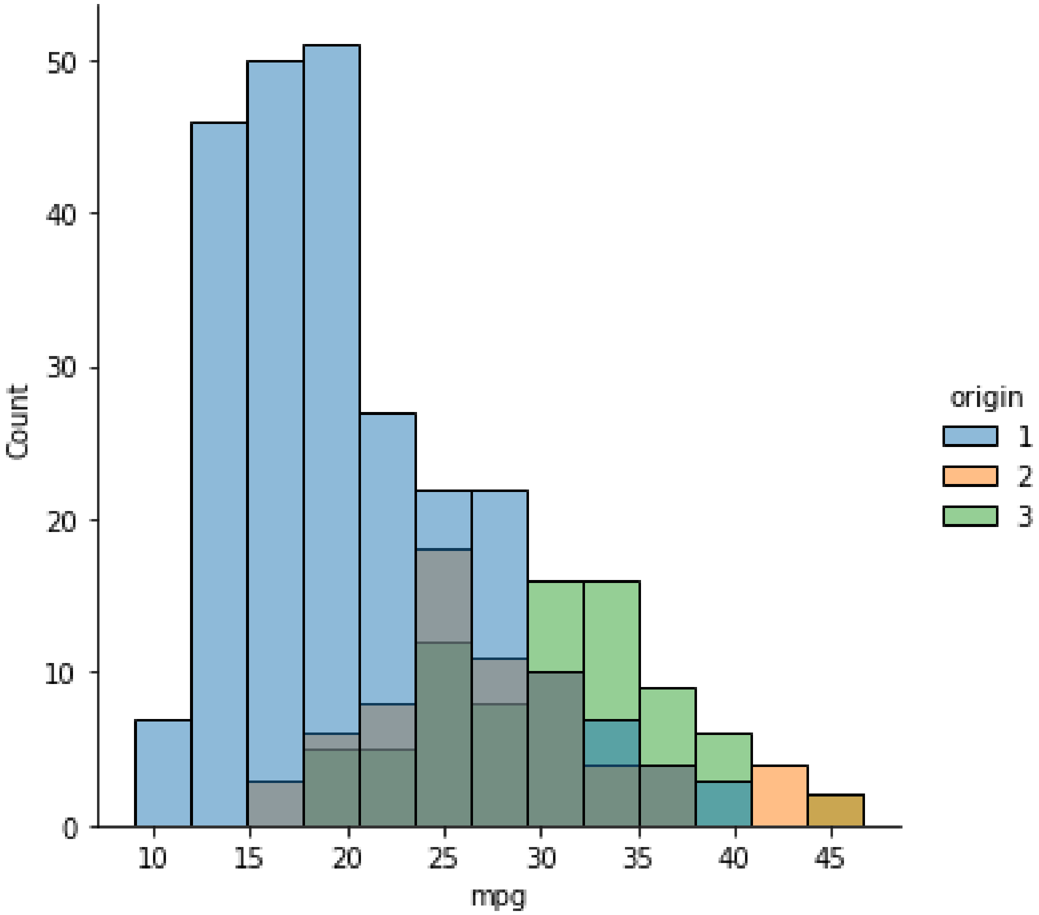
[12]:

```
# Use gender to split age into columns
sns.displot(x='mpg', col='origin', data=data);
```



[13]:

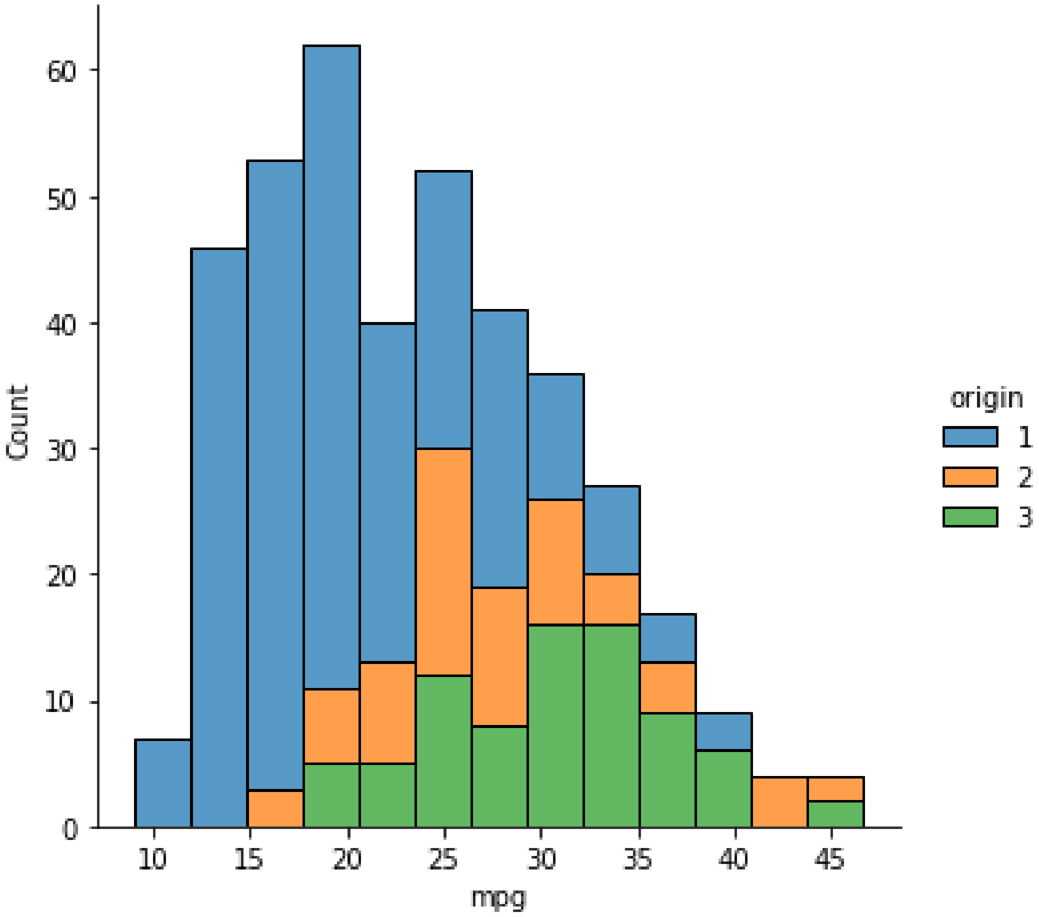
```
# Use origin to color (hue) in the same plot
sns.displot(x='mpg', hue='origin', data=data);
```



To have an idea of the split between male and female, we can stack the counts, adding up to total.

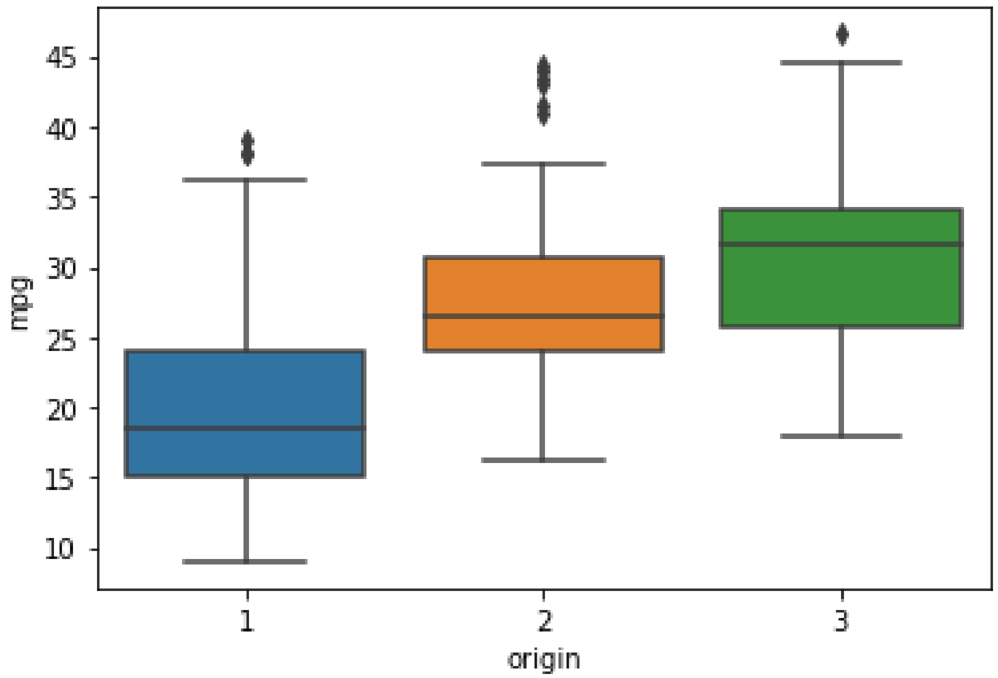
[14]:

```
sns.displot(x='mpg', hue='origin', data=data, multiple='stack');
```



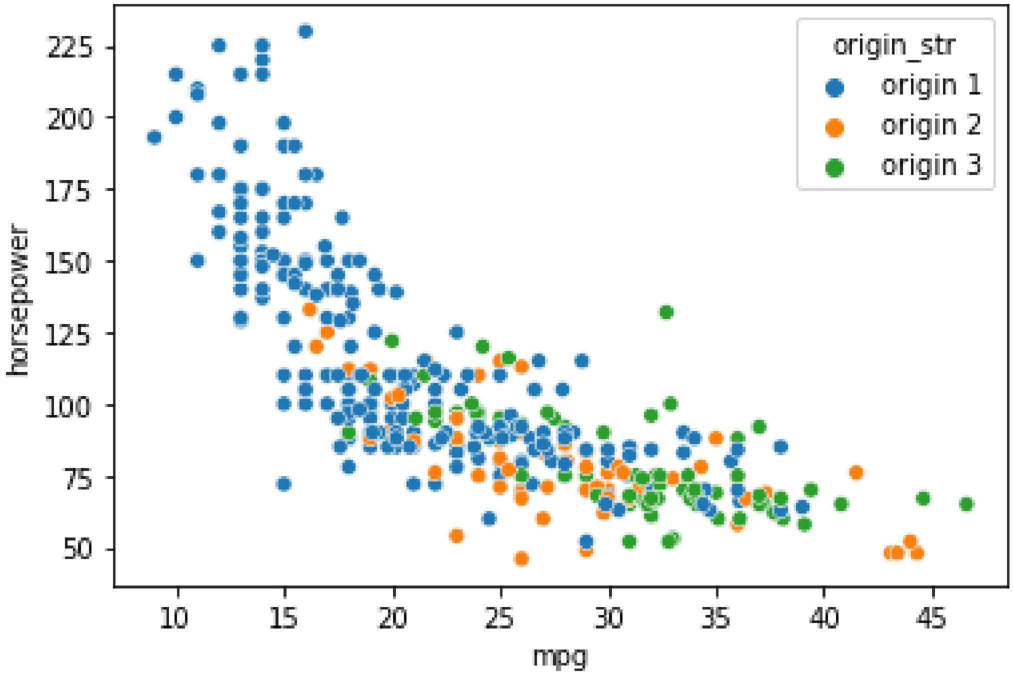
We can look at the differences in ages with a boxplot too

[15]:
```
sns.boxplot(x='origin', y='mpg', data=data);
```

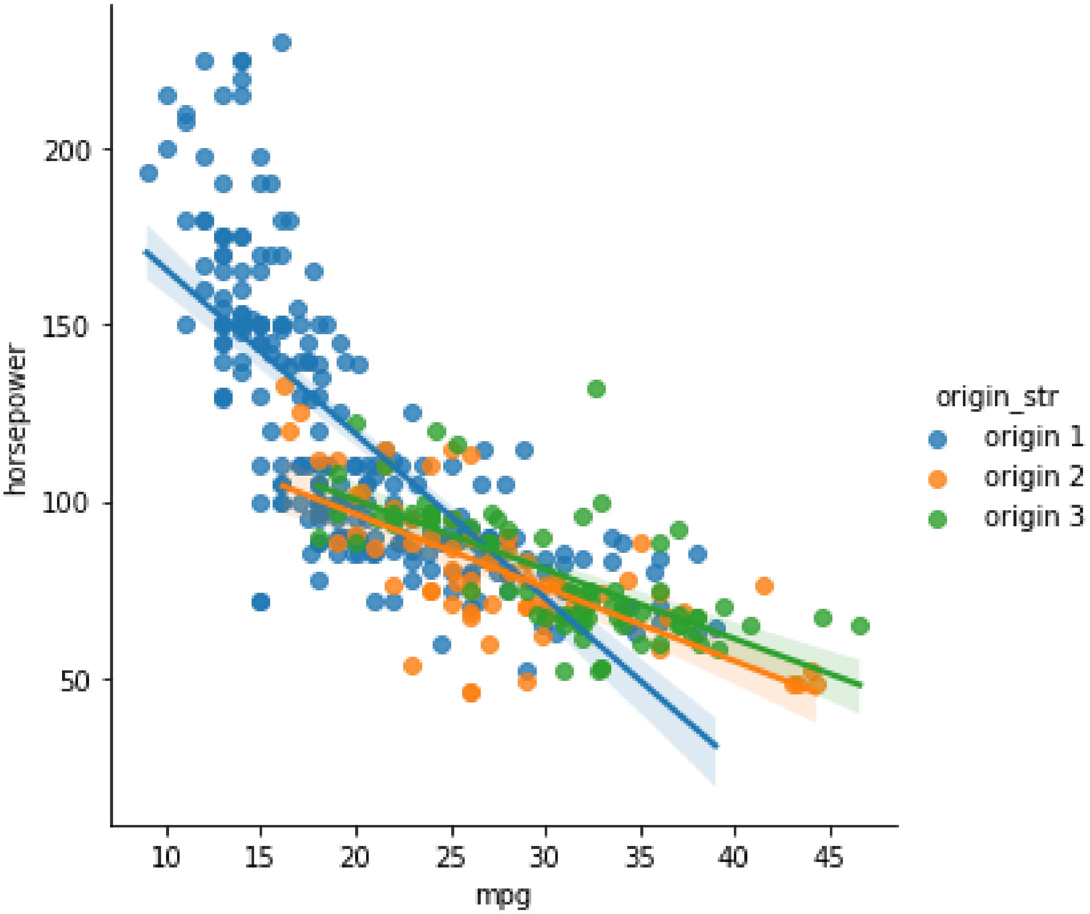


```
[16]: data['origin_str'] = data['origin'].replace([1, 2, 3], ['origin 1', 'origin 2',␣
       ↪'origin 3'])
```

```
[17]: ax = sns.scatterplot(x='mpg', y='horsepower', data=data, hue='origin_str')
```

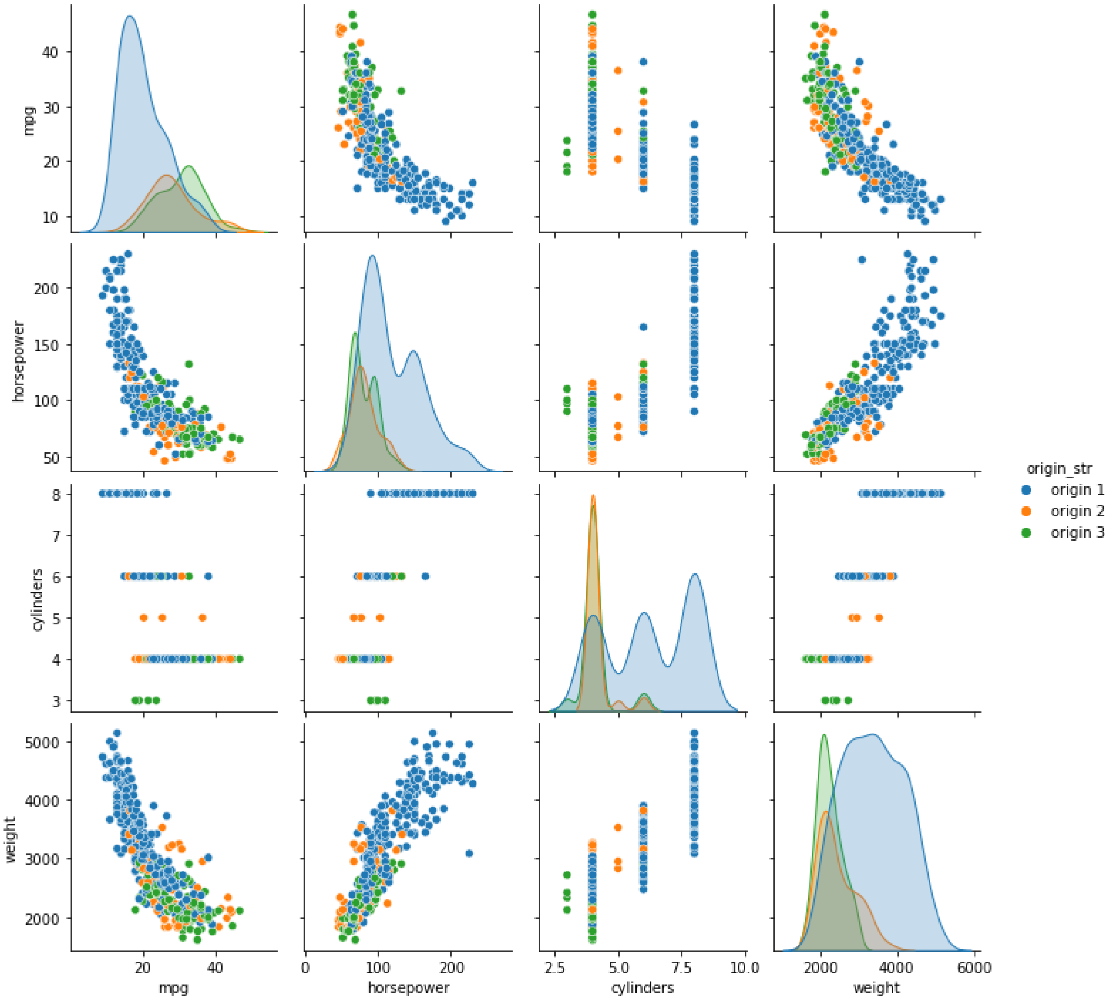


```
[18]: ax = sns.lmplot(x='mpg', y='horsepower', data=data, hue='origin_str')
```

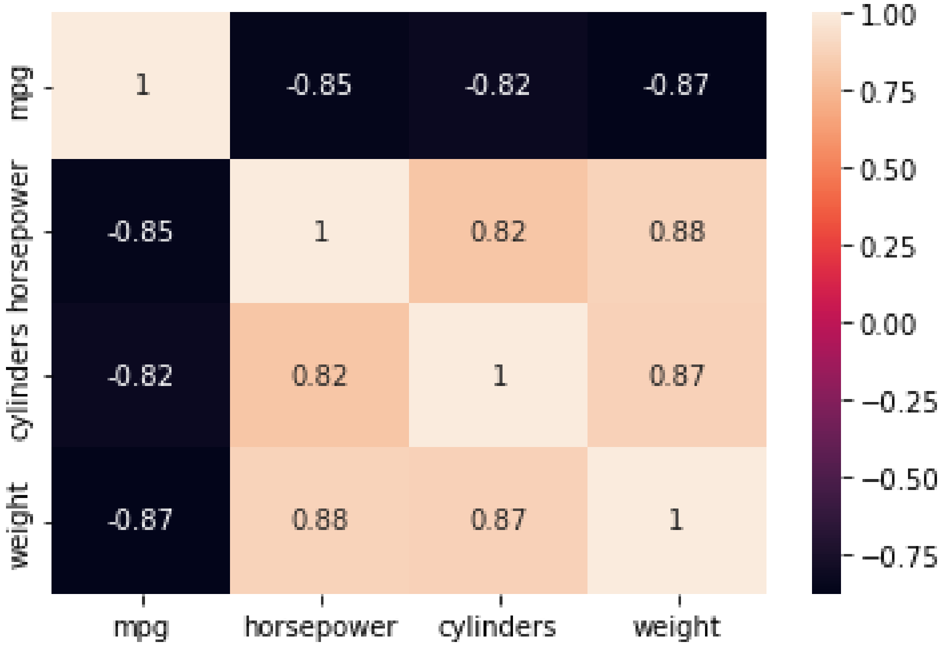


Maybe there are other correlations in the data set. Pairplot is a great way to get an overview

```
[19]: sns.pairplot(data, vars=['mpg', 'horsepower', 'cylinders', 'weight'],␣
       ↪hue='origin_str');
```



```
[20]: g = sns.heatmap(data[['mpg', 'horsepower', 'cylinders', 'weight']].
       ↪corr(method='spearman'),
                   annot=True)
```


mpg
horsepower
cylinders
weight
1
-0.85
-0.82
-0.87
-0.85
1
0.82
0.88
-0.82
0.82
1
0.87
-0.87
0.88
0.87
1
mpg
horsepower
cylinders
weight
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

[ ]: