
run_length

Unknown Author

December 03, 2013

```
In [1]: %matplotlib inline
import pandas as pd
import os
import numpy as np
import matplotlib.pyplot as plt
try:
    os.mkdir('../figures')
except WindowsError:
    print "figures exists"
figures exists
```

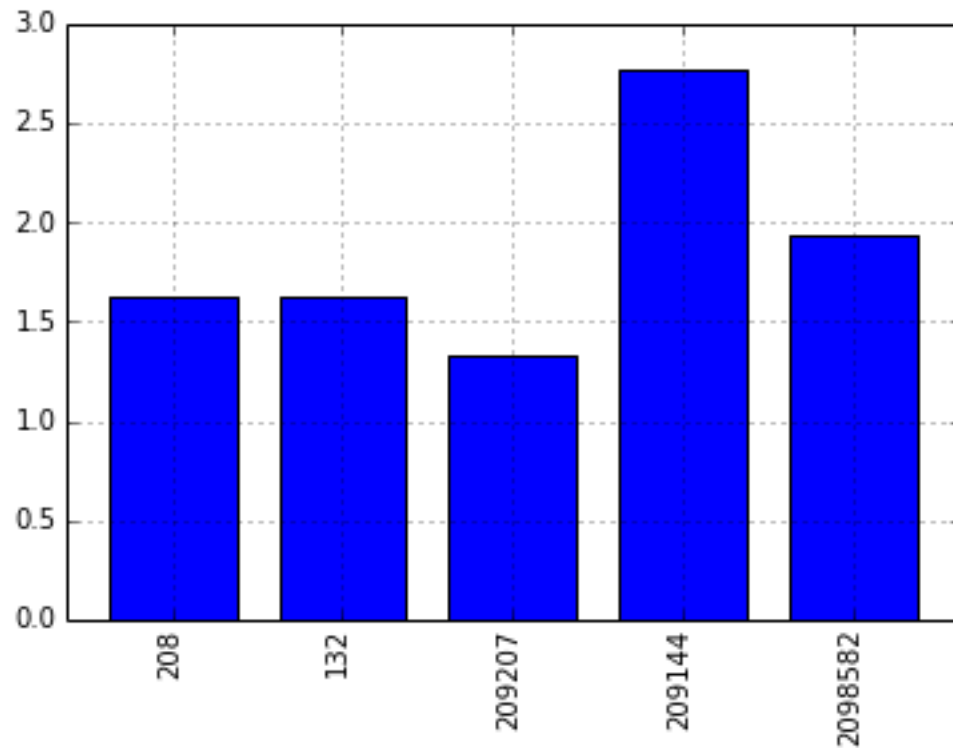
```
In [2]: data = pd.read_csv("../input/run_length_data.csv")
data.head()
```

```
Out [2]:
```

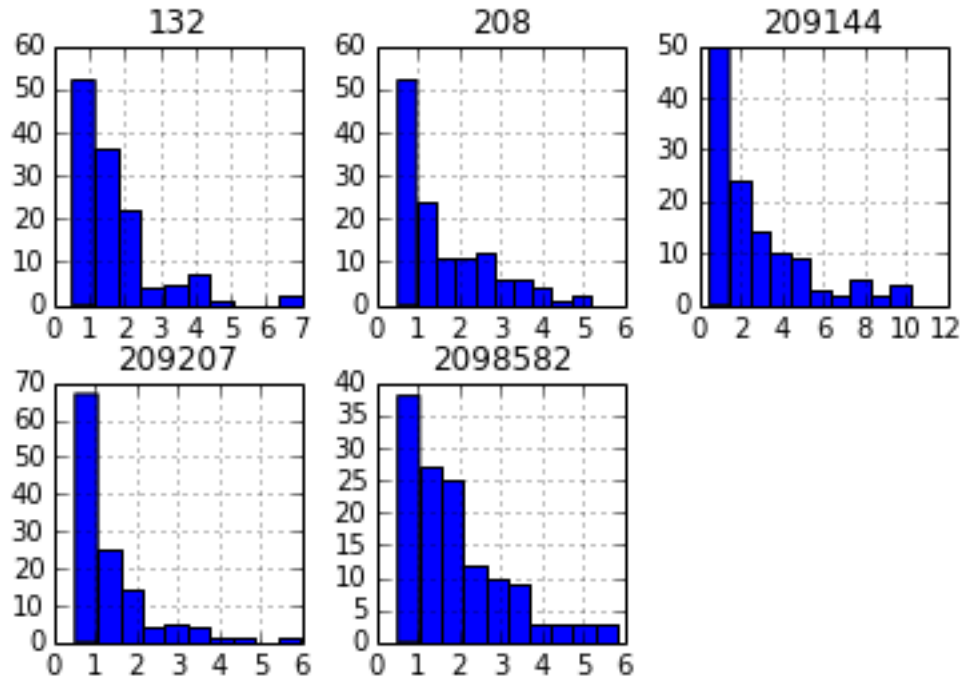
	208	132	209207	209144	2098582
0	3.8790	1.0344	0.6465	5.9478	2.1981
1	1.2930	1.0344	0.6465	1.8102	2.5860
2	2.4567	1.0344	0.6465	0.7758	2.3274
3	1.9395	0.5172	0.7758	3.4911	1.1637
4	1.4223	2.3274	0.9051	7.1115	4.9134

```
data.mean()
In [3]: 208          1.629781
Out [3]: 132          1.627777
209207      1.325855
209144      2.771015
2098582     1.930750
dtype: float64
```

```
data.mean().plot(kind='bar')
In [4]: <matplotlib.axes.AxesSubplot at 0x8c29a90>
Out [4]:
```

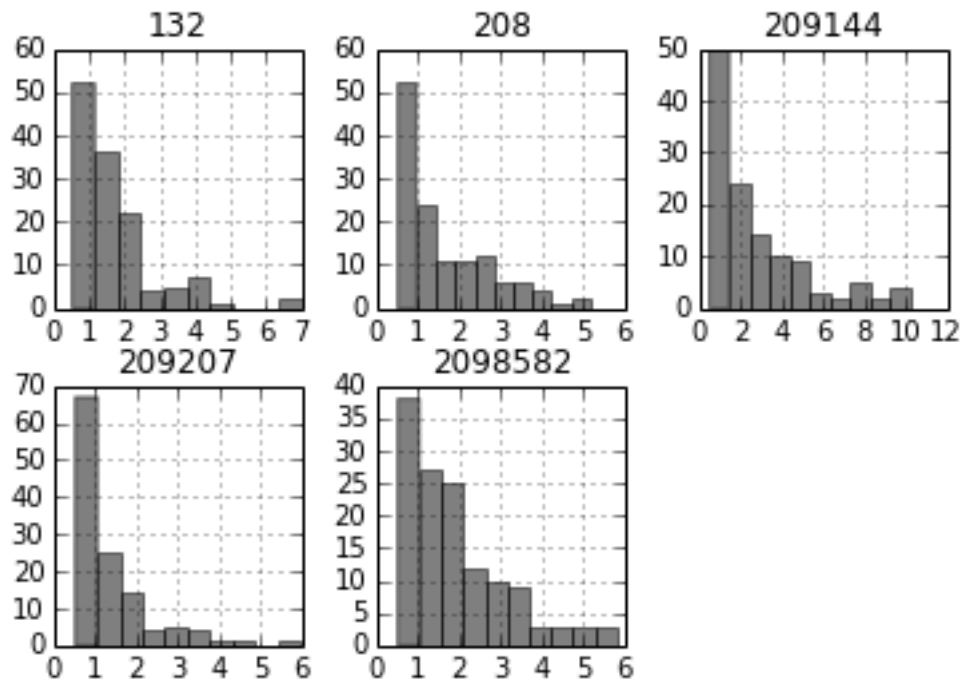


```
data.hist()  
In [5]: array([[<matplotlib.axes.AxesSubplot object at 0x0000000008C41710>,  
Out [5]: <matplotlib.axes.AxesSubplot object at 0x000000000906D860>,  
          <matplotlib.axes.AxesSubplot object at 0x0000000009094630>],  
          [<matplotlib.axes.AxesSubplot object at 0x000000000A7E76A0>,  
          <matplotlib.axes.AxesSubplot object at 0x000000000A805F60>,  
          <matplotlib.axes.AxesSubplot object at 0x000000000A9AECC0>]],  
          dtype=object)
```



```
data.hist(color='k', alpha=0.5, bins=10)
```

```
In [6]: array([[<matplotlib.axes.AxesSubplot object at 0x000000000A818E48>,
Out [6]: <matplotlib.axes.AxesSubplot object at 0x000000000AF3F8D0>,
          <matplotlib.axes.AxesSubplot object at 0x000000000B101F60>],
          [<matplotlib.axes.AxesSubplot object at 0x000000000B122320>,
          <matplotlib.axes.AxesSubplot object at 0x000000000B2902B0>,
          <matplotlib.axes.AxesSubplot object at 0x000000000B2B1940>]],
          dtype=object)
```



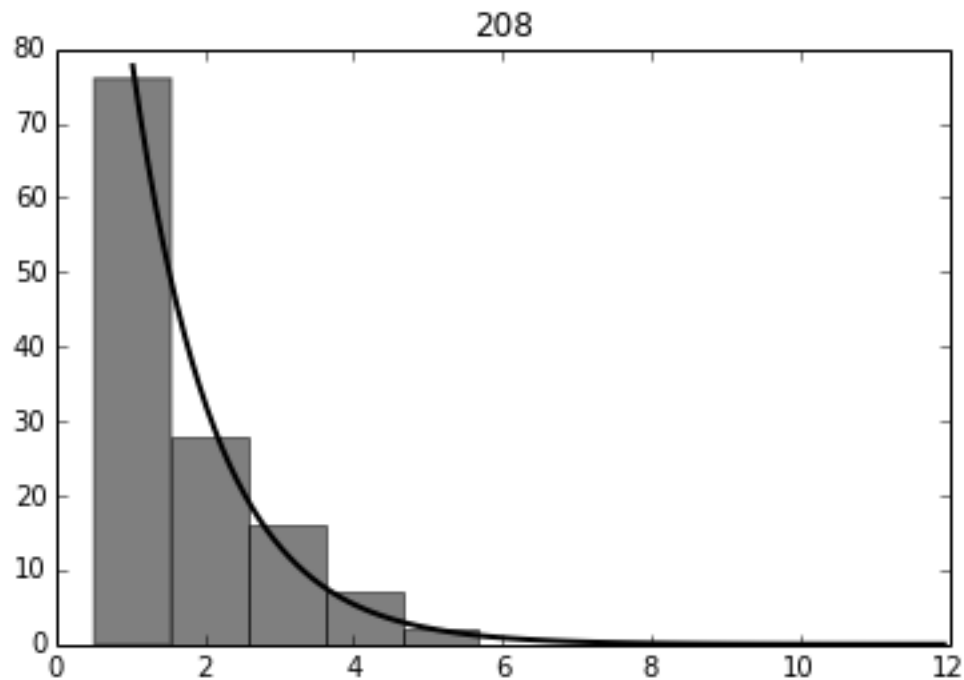
```

In [8]: def fitline(x, mean, binsize, n, start):
        y = 1/mean*np.exp(-(x-start)/mean)*n*binsize
        return y

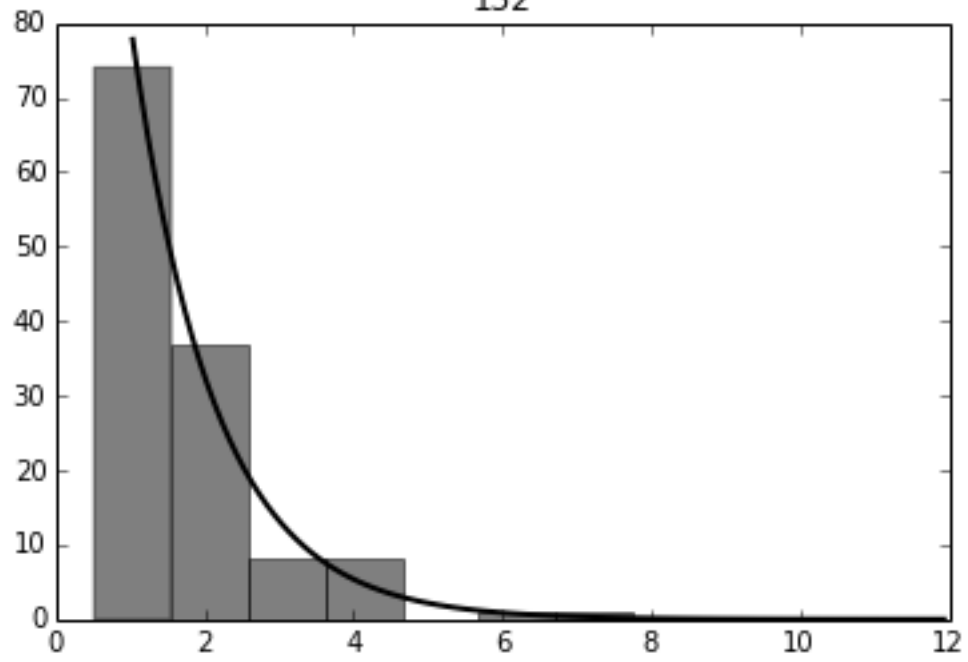
        def rlplot(col, start, stop, binsize):
            runs = col[~np.isnan(col)] # remove nans
            x = np.arange(start + binsize/2, stop, .1) # xvals for fit line
            decayconst = np.mean(col) - start
            y = fitline(x, decayconst, binsize, len(col), start) # compute fit line
            plt.figure() # make a new figure
            plt.hist(col.values, bins=np.arange(start, stop, binsize), color='k', alpha=0.5) #
            plt.plot(x, y, color='k', linewidth=2) # add the fit lines
            plt.title(col.name)
            plt.savefig("../figures/" + col.name + "_runlength.eps") # save as .eps
            return decayconst

        start = .5172
        binsize = .5172*2
        stop = 12
        data.apply(rlplot, args=(start, stop, binsize)) # apply the plotting function to each
208      1.112581
Out [8]: 132      1.110577
        209207      0.808655
        209144      2.253815
        2098582     1.413550
dtype: float64

```



132



209207

