

03-Pandas-Data-Visualization-Exercise-Solutions

October 19, 2022

Copyright Pierian Data

For more information, visit us at www.pieriandata.com

1 Pandas Data Visualization Exercises - Solution

This is just a quick exercise to review the various plots we showed earlier. Use df3.csv to replicate the following plots.

IMPORTANT NOTE! Make sure you don't run the cells directly above the example output shown, otherwise you will end up writing over the example output!

```
[1]: # RUN THIS CELL
import pandas as pd
%matplotlib inline

df3 = pd.read_csv('df3.csv')
print(len(df3))
print(df3.head())
```

500

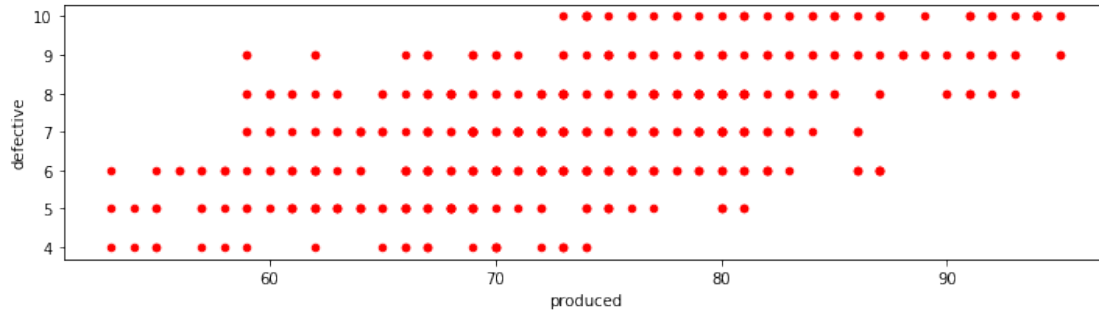
| | weekday | produced | defective |
|---|-------------|----------|-----------|
| 0 | 1.Monday | 73 | 7 |
| 1 | 2.Tuesday | 75 | 10 |
| 2 | 3.Wednesday | 86 | 7 |
| 3 | 4.Thursday | 64 | 7 |
| 4 | 5.Friday | 70 | 6 |

So df3 has 500 records and 3 columns. The data represents factory production numbers and reported numbers of defects on certain days of the week.

1.0.1 1. Recreate this scatter plot of 'produced' vs 'defective'. Note the color and size of the points. Also note the figure size. See if you can figure out how to stretch it in a similar fashion.

```
[ ]: # CODE HERE
```

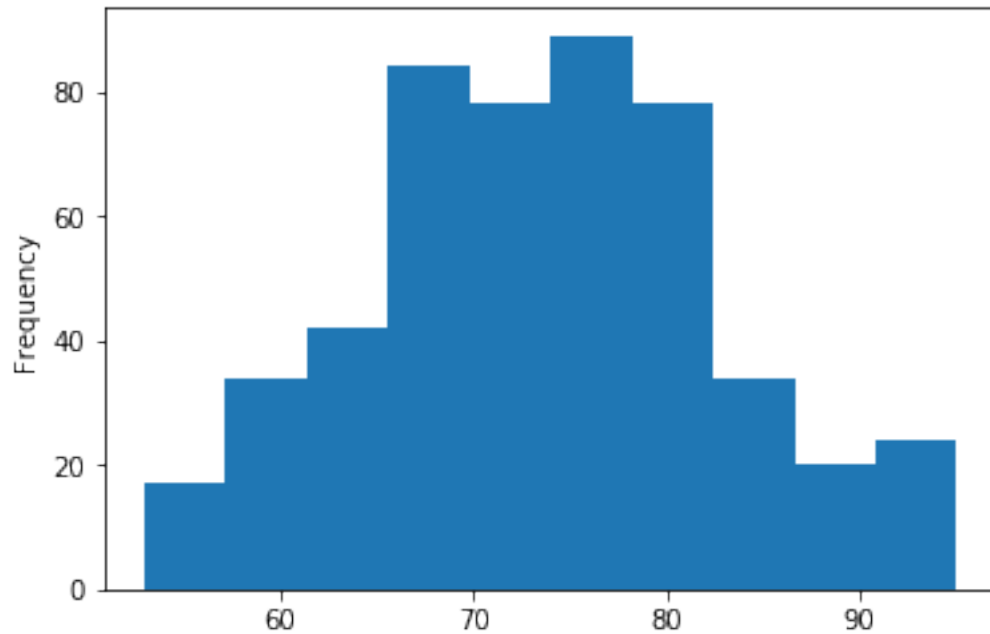
```
[2]: # DON'T WRITE HERE
df3.plot.scatter(x='produced',y='defective',c='red',s=20,figsize=(12,3));
```



1.0.2 2. Create a histogram of the 'produced' column.

```
[ ]:
```

```
[3]: # DON'T WRITE HERE
df3['produced'].plot.hist();
```

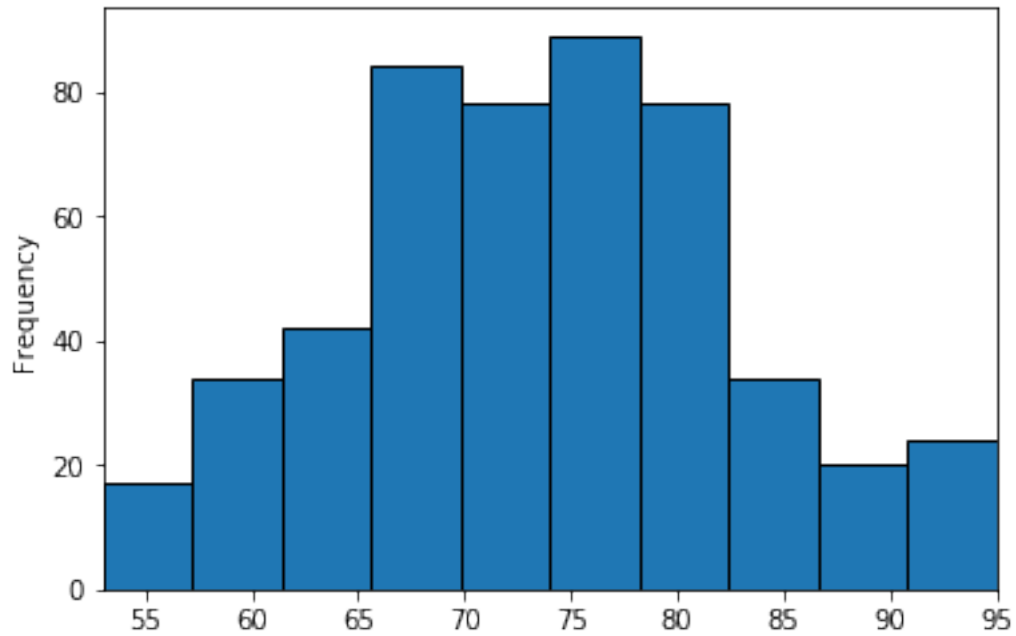


1.0.3 3. Recreate the following histogram of 'produced', tightening the x-axis and adding lines between bars.

[]:

[4]:

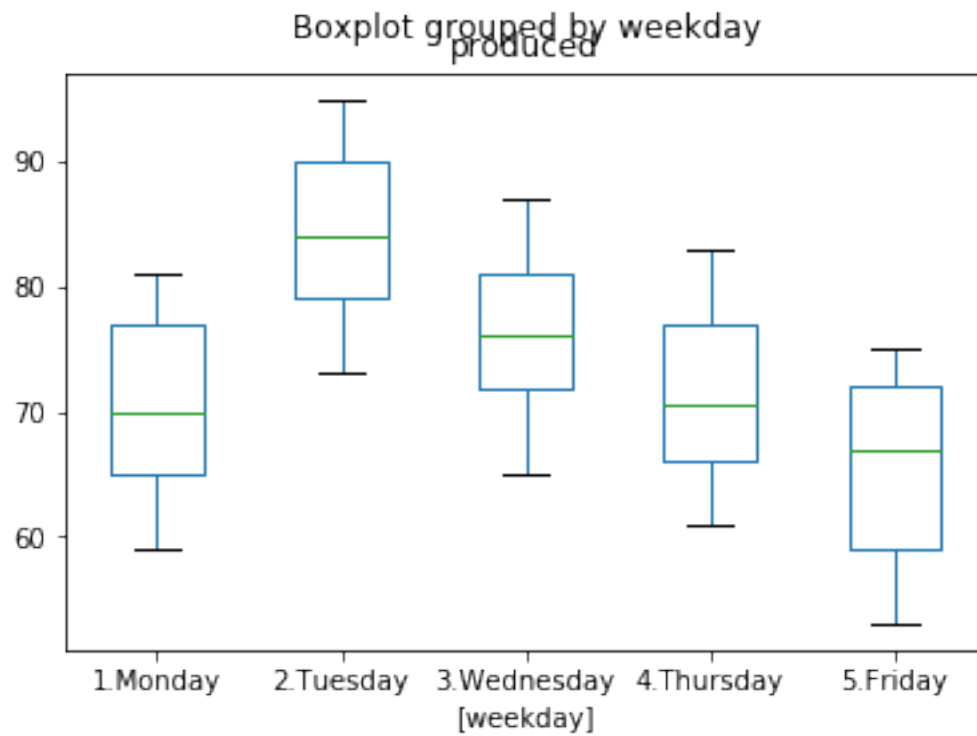
```
# DON'T WRITE HERE
df3['produced'].plot.hist(edgecolor='k').autoscale(axis='x',tight=True);
```



1.0.4 4. Create a boxplot that shows 'produced' for each 'weekday' (hint: this is a groupby operation)

[]:

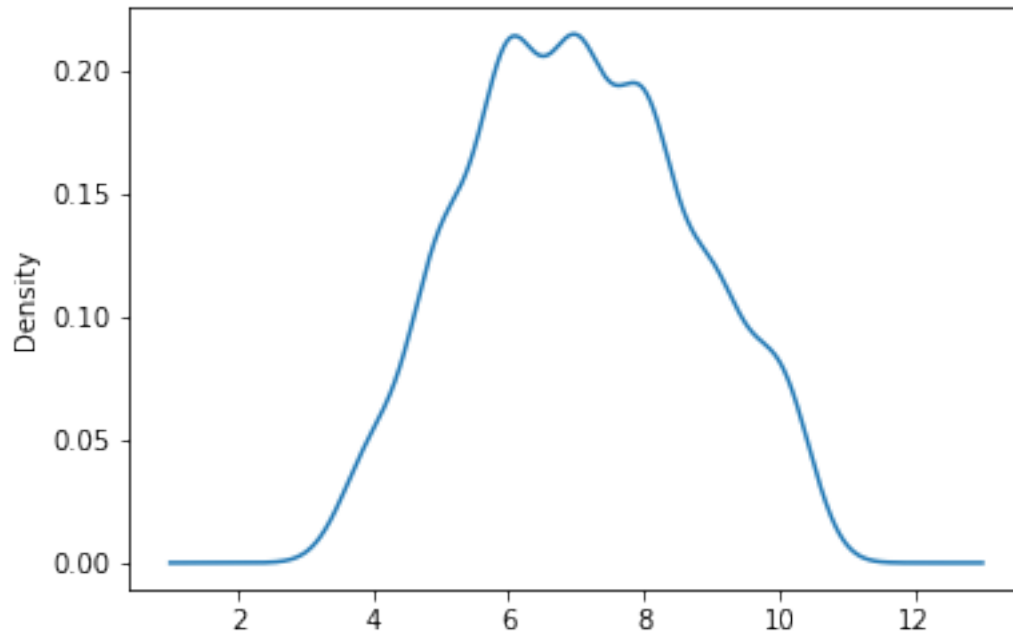
```
[5]: # DON'T WRITE HERE
df3[['produced', 'weekday']].boxplot(by='weekday', grid=False);
```



1.0.5 5. Create a KDE plot of the 'defective' column

[]:

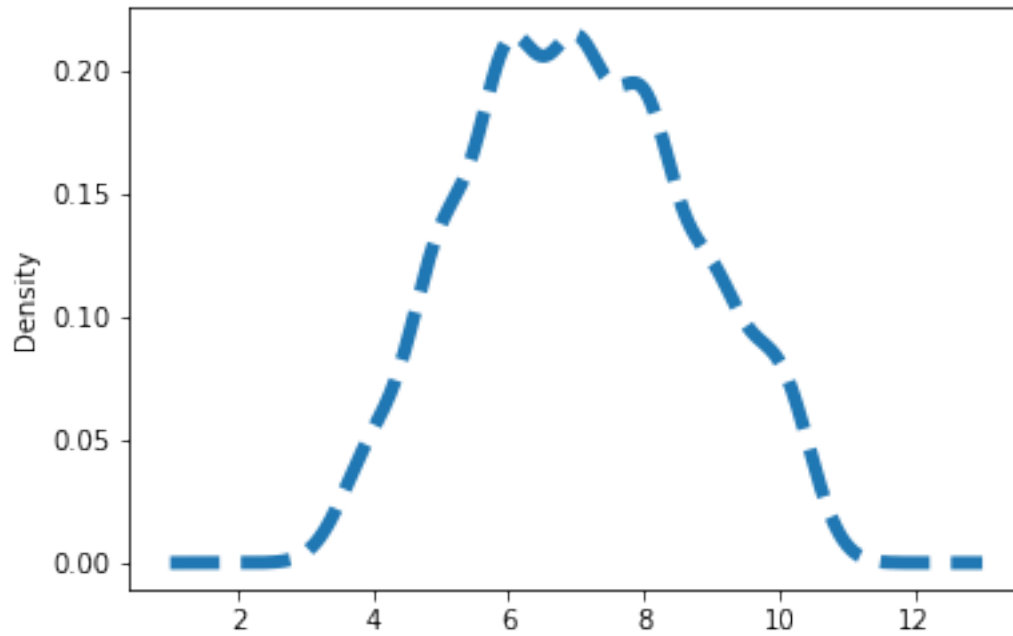
```
[6]: # DON'T WRITE HERE
df3['defective'].plot.kde();
```



1.0.6 6. For the above KDE plot, figure out how to increase the linewidth and make the linestyle dashed.(Note: You would usually not dash a KDE plot line)

[]:

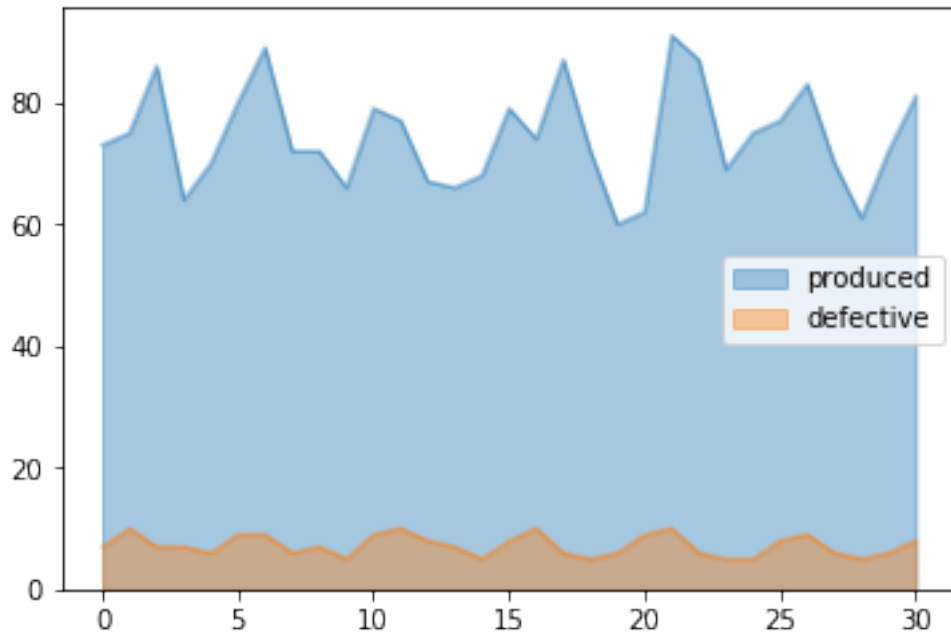
```
[7]: # DON'T WRITE HERE
df3['defective'].plot.kde(lw=5,ls='--');
```



1.0.7 7. Create a blended area plot of all the columns for just the rows up to 30.
(hint: use `.loc`)

[]:

```
[8]: # DON'T WRITE HERE
df3.loc[0:30].plot.area(stacked=False,alpha=0.4);
```



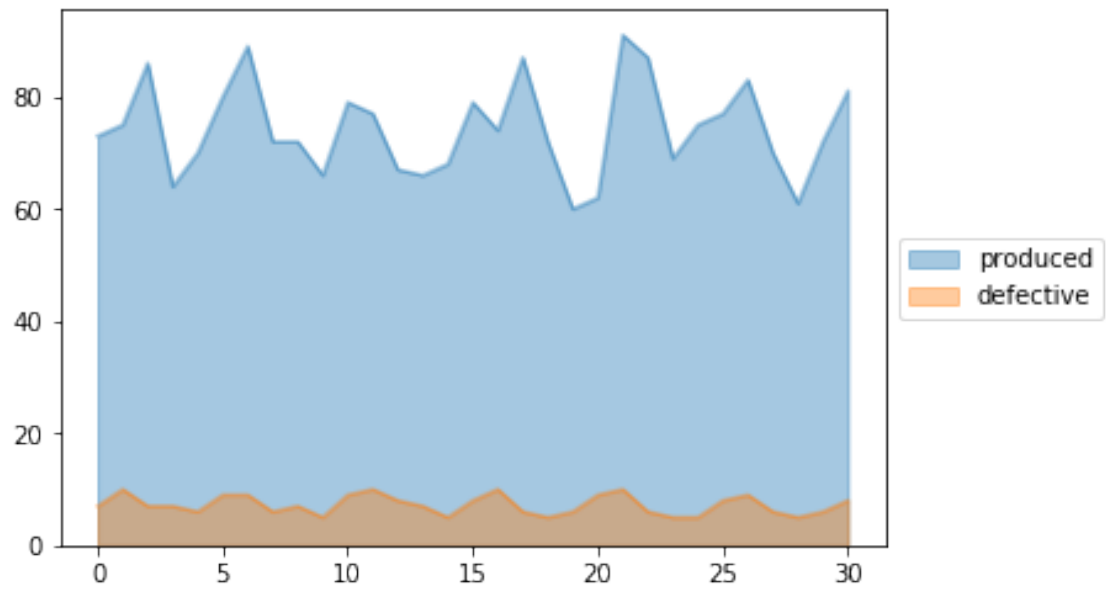
1.1 Bonus Challenge!

Notice how the legend in our previous figure overlapped some of actual diagram. Can you figure out how to display the legend outside of the plot as shown below?

[]:

```
[9]: # DON'T WRITE HERE
ax = df3.loc[0:30].plot.area(stacked=False,alpha=0.4)
ax.legend(loc='center left',bbox_to_anchor=(1.0, 0.5));

# This also works:
# df3.loc[0:30].plot.area(stacked=False,alpha=0.4).legend(loc='center_
↳ left',bbox_to_anchor=(1.0, 0.5));
```

2 Great Job!