Q1

January 8, 2020

[25]:
```
import pandas as pd

file = 'https://cocl.us/datascience_survey_data'

df_data = pd.read_csv(file, index_col=0)
df_data= df_data.sort_values(by=['Very interested'],ascending=False)
print('Data downloaded and read into a dataframe')
df_data.head()
```

```
Data downloaded and read into a dataframe
```

[25]:

	Very interested	Somewhat interested	Not interested
Data Analysis / Statistics	1688	444	60
Machine Learning	1629	477	74
Data Visualization	1340	734	102
Big Data (Spark / Hadoop)	1332	729	127
Deep Learning	1263	770	136

[19]:
```
df_data = df_data.div(df_data.sum(1), axis=0)
df_data.head()
```

[19]:

	Very interested	Somewhat interested	Not interested
Data Analysis / Statistics	0.770073	0.202555	0.027372
Machine Learning	0.747248	0.218807	0.033945
Data Visualization	0.615809	0.337316	
Big Data (Spark / Hadoop)	0.608775	0.333181	
Deep Learning	0.582296	0.355002	

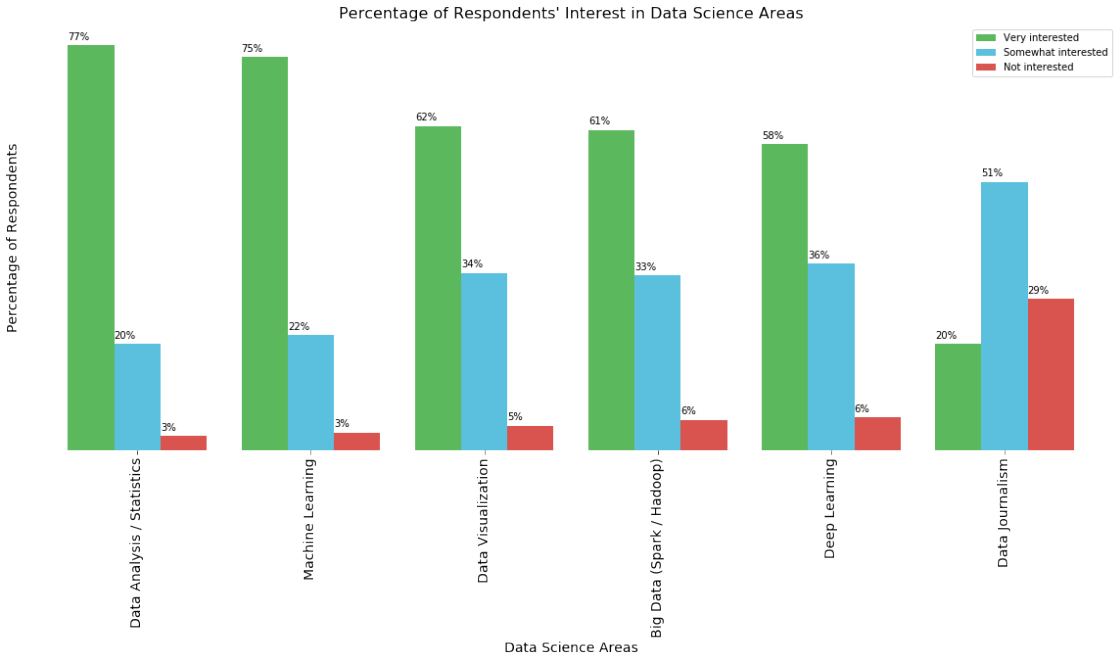
```
Data Visualization                0.046875
Big Data (Spark / Hadoop)         0.058044
Deep Learning                     0.062702
```

[20]:
```
import matplotlib as mpl
import matplotlib.pyplot as plt
#very =(0.3607843137254902,0.7215686274509804,0.3607843137254902)
#some = (0.3568627450980392,0.7529411764705882,0.87058823529411777)
#notint=(0.8509803921568627,0.3254901960784314,0.30980392156862746)
colors_list = ['#5cb85c','#5bc0de','#d9534f']
ax= df_data.plot(kind='bar',
             figsize=(20, 8), width = 0.8, color =colors_list, edgecolor = None␣
 ↪# pass a tuple (x, y) size
             )

plt.title("Percentage of Respondents' Interest in Data Science Areas",␣
 ↪fontsize=16)
plt.ylabel('Percentage of Respondents',fontsize=14)
plt.xlabel('Data Science Areas',fontsize=14)
for p in ax.patches:
    width, height = p.get_width(), p.get_height()
    x, y = p.get_xy()
    ax.annotate('{:.0%}'.format(height), (x, y + height+0.01))

#to take out the bars on left, right, and up
plt.xticks(fontsize=14)
for spine in plt.gca().spines.values():
    spine.set_visible(False)
plt.yticks([])

plt.show()
```



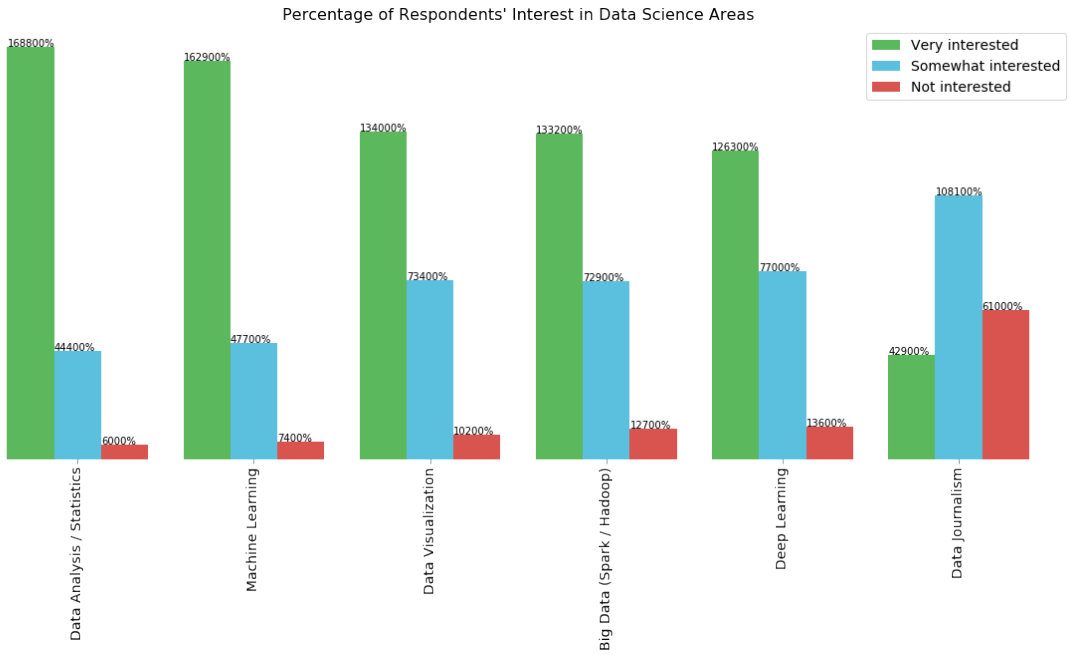
[27]:
```
colors_list = ['#5cb85c','#5bc0de','#d9534f']

# Change this line to plot percentages instead of absolute values

result = df_data
ax = result.plot(kind='bar',figsize=(20,8),width = 0.8,color =␣
↪colors_list,edgecolor=None)
plt.legend(labels=result.columns,fontsize= 14)
plt.title("Percentage of Respondents' Interest in Data Science Areas",fontsize=␣
↪16)

#to take out the bars on left, right, and up
plt.xticks(fontsize=14)
for spine in plt.gca().spines.values():
    spine.set_visible(False)
plt.yticks([])

# Add this loop to add the annotations
for p in ax.patches:
    width, height = p.get_width(), p.get_height()
    x, y = p.get_xy()
    ax.annotate('{:.0%}'.format(height), (x, y + height + 0.01))
```



[30]:
```
import folium
import numpy as np

#print('Folium installed and imported!')

df_incidents = pd.read_csv('https://cocl.us/sanfran_crime_dataset')

df= df_incidents.groupby('PdDistrict', as_index= False).count()
new_df = df[['PdDistrict','Category']]
sf_data = new_df.rename(columns={"Category": "Count","PdDistrict":
 ↪"Neighborhood"})
sf_data.head(20)
```

[30]:

	Neighborhood	Count
0	BAYVIEW	14303
1	CENTRAL	17666
2	INGLESIDE	11594
3	MISSION	19503
4	NORTHERN	20100
5	PARK	8699
6	RICHMOND	8922
7	SOUTHERN	28445
8	TARAVAL	11325
9	TENDERLOIN	9942

[31]:
```
# San Francisco latitude and longitude values
latitude = 37.77
longitude = -122.42

# create map and display it
sf_map = folium.Map(location=[latitude, longitude], zoom_start=12)

# display the map of San Francisco
sf_map
```

[31]:
```
<folium.folium.Map at 0x7fd3fb78d978>
```