final_assign

April 18, 2020

[1]:
```
import numpy as np
import pandas as pd
```

[15]:
```
#read data
df_survey = pd.read_csv('https://cocl.us/datascience_survey_data', index_col =␣
↪0)
df_survey.head()
```

[15]:

	Very interested	Somewhat interested	Not interested
Big Data (Spark / Hadoop)	1332	729	127
Data Analysis / Statistics	1688	444	60
Data Journalism	429	1081	610
Data Visualization	1340	734	102
Deep Learning	1263	770	136

[18]:
```
%matplotlib inline

import matplotlib as mpl
import matplotlib.pyplot as plt
```

[17]:
```
#create bar chart
df_survey.sort_values(['Very interested'], ascending = False, inplace = True)
df_survey.head()
```

[17]:

	Very interested	Somewhat interested	\
Data Analysis / Statistics	1688	444	
Machine Learning	1629	477	
Data Visualization	1340	734	
Big Data (Spark / Hadoop)	1332	729	
Deep Learning	1263	770	

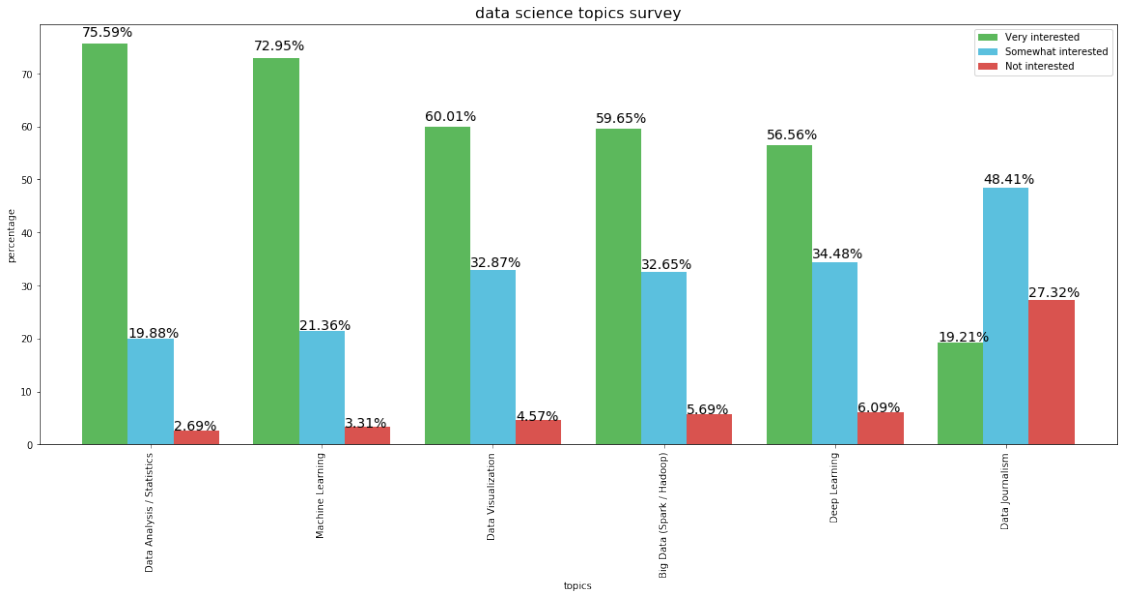
	Not interested
Data Analysis / Statistics	60
Machine Learning	74
Data Visualization	102
Big Data (Spark / Hadoop)	127
Deep Learning	136

[58]:
```
#find the percentage
df_survey_percentage = ((df_survey / 2233) * 100).round(2)

#plot
ax = df_survey_percentage.plot(kind='bar', width = 0.8, figsize=(20, 8), color␣
 ↪= ['#5cb85c', '#5bc0de', '#d9534f'])

ax.set_title('data science topics survey', fontsize = 16)
ax.set_ylabel('percentage')
ax.set_xlabel('topics')

#Annotate Text
for p in ax.patches:
    ax.annotate(str(p.get_height()) + '%', xy = (p.get_x(), p.get_height()*1.
 ↪02),fontsize=14)
```



[137]:
```
#Final: Choropleth map
df_sf_= pd.read_csv('https://cocl.us/sanfran_crime_dataset')   #read data
```

```
df_sf = df_sf_.groupby(['PdDistrict']).count().reset_index()
df_sf
```

[137]:

	PdDistrict	IncidntNum	Category	Descript	DayOfWeek	Date	Time	Resolution	Address	X	Y	Location	PdId
0	BAYVIEW	14303	14303	14303	14303	14303	14303	14303	14303	14303	14303	14303	14303
1	CENTRAL	17666	17666	17666	17666	17666	17666	17666	17666	17666	17666	17666	17666
2	INGLESIDE	11594	11594	11594	11594	11594	11594	11594	11594	11594	11594	11594	11594
3	MISSION	19503	19503	19503	19503	19503	19503	19503	19503	19503	19503	19503	19503
4	NORTHERN	20100	20100	20100	20100	20100	20100	20100	20100	20100	20100	20100	20100
5	PARK	8699	8699	8699	8699	8699	8699	8699	8699	8699	8699	8699	8699
6	RICHMOND	8922	8922	8922	8922	8922	8922	8922	8922	8922	8922	8922	8922
7	SOUTHERN	28445	28445	28445	28445	28445	28445	28445	28445	28445	28445	28445	28445
8	TARAVAL	11325	11325	11325	11325	11325	11325	11325	11325	11325	11325	11325	11325
9	TENDERLOIN	9942	9942	9942	9942	9942	9942	9942	9942	9942	9942	9942	9942

[138]:
```
#drop unnecessary data
df_sf.drop(['Category', 'Descript', 'DayOfWeek','Date', 'Time', 'Resolution',␣
↪'Address', 'X','Y','Location','PdId'], axis = 1,inplace=True)
df_sf.rename(columns = {'PdDistrict':'Neighborhood', 'IncidntNum':'Count'},␣
↪inplace = True)
df_sf
```

[138]:

	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

[139]:
```
#get json
!wget --quiet https://cocl.us/sanfran_geojson
```

[140]:
```
import folium
```

[141]:
```
sf_geo = r'sanfran_geojson' # geojson file

# create a plain map
sf_map = folium.Map(location=[37.4639, -122.2459], zoom_start=12)
sf_map.choropleth(
    geo_data=sf_geo,
    data=df_sf,
    columns=['Neighborhood', 'Count'],
    key_on='feature.properties.DISTRICT',
    fill_color='YlOrRd',
    fill_opacity=0.7,
    line_opacity=0.2,
    legend_name='San Francisco'
)

# display map
sf_map
```

[141]:
```
<folium.folium.Map at 0x7f1b949b8b38>
```

[ ]:
```

```