**Histogramme obtenu avec un autre algorithme**

****

**import** matplotlib.pyplot **as** plt  
  
**def** histogramme\_tweet(a,b,c,f,g,h):  
 **import** seaborn **as** sns  
 **import** pandas **as** pd  
 **import** csv  
 colonne1=[**u'Lepen'**,**u'Positif'**,str(a)]  
 colonne2=[**u'Macron'**,**u'Positif'**,str(f)]  
 colonne3=[**u'Lepen'**,**u'Neutre'**,str(b)]  
 colonne4=[**u'Macron'**,**u'Neutre'**,str(g)]  
 colonne5=[**u'Lepen'**,**u'Negatif'**,str(c)]  
 colonne6=[**u'Macron'**,**u'Negatif'**,str(h)]  
 entetes=[**u'nomCandidat'**,**u'qualite'**,**u'Pourcentage'**]  
 valeurs=[colonne1,colonne2,colonne3,colonne4,colonne5,colonne6]  
 tableau\_histogramme = open(**'Tableau\_histogramme.csv'**, **'w'**)  
 ligneEntete = **","**.join(entetes) + **"\n"** tableau\_histogramme.write(ligneEntete)  
 **for** valeur **in** valeurs:  
 ligne = **","**.join(valeur) + **"\n"** tableau\_histogramme.write(ligne)  
 tableau\_histogramme.close()  
 sns.set(style=**"whitegrid"**)  
  
*# Load the example tableau\_histogramme dataset* tableau\_histogramme = pd.read\_csv(**"Tableau\_histogramme.csv"**)  
 **print**(tableau\_histogramme.head())  
  
*# Draw a nested barplot to show pourcentage for candidate and quality of tweet* g = sns.catplot(x=**"nomCandidat"**, y=**"Pourcentage"**, hue=**"qualite"**, data=tableau\_histogramme,  
 height=6, kind=**"bar"**, palette=**"muted"**)  
 g.despine(left=True)  
 g.set\_ylabels(**"Pourcentage"**)  
 plt.show()  
  
  
histogramme\_tweet(0.7,0.2,0.1,0.6,0.3,0.1)