Identity Resolution

Importing Relevant Modules

In [1]:
```
import pandas as pd
import textdistance as td
import plotly.express as px
```

Reading Dataset

In [2]:
```
df = pd.read_csv('IdentityResolution.csv',names=['Name','Twitter_Username','Facebook_Username','Instagram_Username'])
df.drop_duplicates(inplace=True, ignore_index=True)
df.dropna(inplace=True)
```

Preprocessing

In [3]:
```
Twitter = []
Instagram = []
Facebook = []

for i in df.index:
    Twitter.append(df['Twitter_Username'][i].split("/")[-1].lower())
    Instagram.append(df['Instagram_Username'][i].split("/")[-1].lower())
    if (str(df['Facebook_Username'][i]).split("/")[-1] != 0 ):
        Facebook.append(df['Facebook_Username'][i].split("/")[-1].lower())
    else:
        Facebook.append(df['Facebook_Username'][i].split("/")[-1].lower())

df['Twitter_Username']=Twitter
df['Instagram_Username']=Instagram
df['Facebook_Username']=Facebook
df
```

Out[3]:

	Name	Twitter_Username	Facebook_Username	Instagram_Username
0	Alain Stephan Dominguez Lucero	alainstephan	alainpdc	alainstephan
1	Alex Sultan	alexsultancom	alexsultancom	a_sultan
2	Xavier Gasa	xavigasaco	xgasaco	xavigasaco
3	Nicolás Leyton	nicoleyton	nicoleyton	nicoleyton
4	Mattias Griffel	mattiasgriffel	mattiasgriffel	mattiasgriffel
...	...	...	...	...
318	Vasu Chawla	vasuchawla	vasuchawla26	vasuchawla
319	Dayn Wilberding	dayn	dayne	dayn
320	Guillermo Navarro	bibliotim	dirguillermonavarro	bibliotim
321	Antonio J. Cuevas	pennurunnus	antonio.j.cuevas	pennurunnus
322	Ghibril Anadna	arghibril	ghibril	ghibril

323 rows × 4 columns

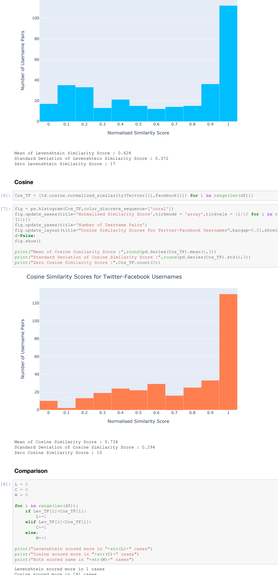
Twitter-Facebook Identity Resolution

Levenshtein

In [4]:
```
Lev_TF = [td.levenshtein.normalized_similarity(Twitter[i],Facebook[i]) for i in range(len(df))]
```

In [5]:
```
fig = px.histogram(Lev_TF,color_discrete_sequence=['deepskyblue'])
fig.update_xaxes(title="Normalised Similarity Score",tickmode = 'array',tickvals = [i/10 for i in range(11)])
fig.update_yaxes(title="Number of Username Pairs")
fig.update_layout(title="Levenshtein Similarity Scores for Twitter-Facebook Usernames",bargap=0.01,showlegend=False)
fig.show()

print("Mean of Levenshtein Similarity Score :",round(pd.Series(Lev_TF).mean(),3))
print("Standard Deviation of Levenshtein Similarity Score :",round(pd.Series(Lev_TF).std(),3))
print("Zero Levenshtein Similarity Score :",Lev_TF.count(0))
```



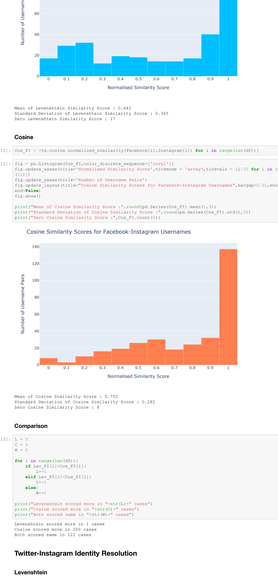

```
Mean of Levenshtein Similarity Score : 0.428
Standard Deviation of Levenshtein Similarity Score : 0.372
Zero Levenshtein Similarity Score : 27
```

Cosine

In [6]:
```
Cos_TF = [td.cosine.normalized_similarity(Twitter[i],Facebook[i]) for i in range(len(df))]
```

In [7]:
```
fig = px.histogram(Cos_TF,color_discrete_sequence=['coral'])
fig.update_xaxes(title="Normalised Similarity Score",tickmode = 'array',tickvals = [i/10 for i in range(11)])
fig.update_yaxes(title="Number of Username Pairs")
fig.update_layout(title="Cosine Similarity Scores for Twitter-Facebook Usernames",bargap=0.01,showlegend=False)
fig.show()

print("Mean of Cosine Similarity Score :",round(pd.Series(Cos_TF).mean(),3))
print("Standard Deviation of Cosine Similarity Score :",round(pd.Series(Cos_TF).std(),3))
print("Zero Cosine Similarity Score :",Cos_TF.count(0))
```




```
Mean of Cosine Similarity Score : 0.734
Standard Deviation of Cosine Similarity Score : 0.294
Zero Cosine Similarity Score : 10
```

Comparison

In [8]:
```
L = 0
C = 0
W = 0

for i in range(len(df)):
    if Lev_TF[i]>Cos_TF[i]:
        L+=1
    elif Lev_TF[i]<Cos_TF[i]:
        C+=1
    else:
        W+=1

print("Levenshtein scored more in "+str(L)+" cases")
print("Cosine scored more in "+str(C)+" cases")
print("Both scored same in "+str(W)+" cases")
```

```
Levenshtein scored more in 1 cases
Cosine scored more in 191 cases
Both scored same in 131 cases
```

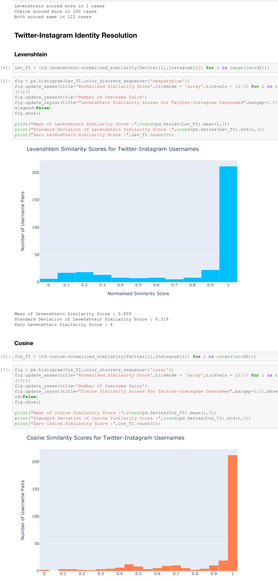
Facebook-Instagram Identity Resolution

Levenshtein

In [9]:
```
Lev_FI = [td.levenshtein.normalized_similarity(Facebook[i],Instagram[i]) for i in range(len(df))]
```

In [10]:
```
fig = px.histogram(Lev_FI,color_discrete_sequence=['deepskyblue'])
fig.update_xaxes(title="Normalised Similarity Score",tickmode = 'array',tickvals = [i/10 for i in range(11)])
fig.update_yaxes(title="Number of Username Pairs")
fig.update_layout(title="Levenshtein Similarity Scores for Facebook-Instagram Usernames",bargap=0.01,showlegend=False)
fig.show()

print("Mean of Levenshtein Similarity Score :",round(pd.Series(Lev_FI).mean(),3))
print("Standard Deviation of Levenshtein Similarity Score :",round(pd.Series(Lev_FI).std(),3))
print("Zero Levenshtein Similarity Score :",Lev_FI.count(0))
```



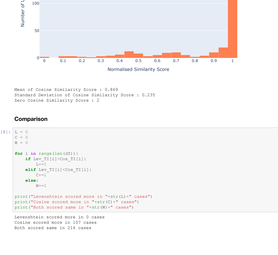

```
Mean of Levenshtein Similarity Score : 0.642
Standard Deviation of Levenshtein Similarity Score : 0.365
Zero Levenshtein Similarity Score : 17
```

Cosine

In [11]:
```
Cos_FI = [td.cosine.normalized_similarity(Facebook[i],Instagram[i]) for i in range(len(df))]
```

In [12]:
```
fig = px.histogram(Cos_FI,color_discrete_sequence=['coral'])
fig.update_xaxes(title="Normalised Similarity Score",tickmode = 'array',tickvals = [i/10 for i in range(11)])
fig.update_yaxes(title="Number of Username Pairs")
fig.update_layout(title="Cosine Similarity Scores for Facebook-Instagram Usernames",bargap=0.01,showlegend=False)
fig.show()

print("Mean of Cosine Similarity Score :",round(pd.Series(Cos_FI).mean(),3))
print("Standard Deviation of Cosine Similarity Score :",round(pd.Series(Cos_FI).std(),3))
print("Zero Cosine Similarity Score :",Cos_FI.count(0))
```




```
Mean of Cosine Similarity Score : 0.752
Standard Deviation of Cosine Similarity Score : 0.282
Zero Cosine Similarity Score : 8
```

Comparison

In [13]:
```
L = 0
C = 0
W = 0

for i in range(len(df)):
    if Lev_FI[i]>Cos_FI[i]:
        L+=1
    elif Lev_FI[i]<Cos_FI[i]:
        C+=1
    else:
        W+=1

print("Levenshtein scored more in "+str(L)+" cases")
print("Cosine scored more in "+str(C)+" cases")
print("Both scored same in "+str(W)+" cases")
```

```
Levenshtein scored more in 1 cases
Cosine scored more in 185 cases
Both scored same in 122 cases
```

Twitter-Instagram Identity Resolution

Levenshtein

In [14]:
```
Lev_TI = [td.levenshtein.normalized_similarity(Twitter[i],Instagram[i]) for i in range(len(df))]
```

In [15]:
```
fig = px.histogram(Lev_TI,color_discrete_sequence=['deepskyblue'])
fig.update_xaxes(title="Normalised Similarity Score",tickmode = 'array',tickvals = [i/10 for i in range(11)])
fig.update_yaxes(title="Number of Username Pairs")
fig.update_layout(title="Levenshtein Similarity Scores for Twitter-Instagram Usernames",bargap=0.01,showlegend=False)
fig.show()

print("Mean of Levenshtein Similarity Score :",round(pd.Series(Lev_TI).mean(),3))
print("Standard Deviation of Levenshtein Similarity Score :",round(pd.Series(Lev_TI).std(),3))
print("Zero Levenshtein Similarity Score :",Lev_TI.count(0))
```



```
Mean of Levenshtein Similarity Score : 0.809
Standard Deviation of Levenshtein Similarity Score : 0.319
Zero Levenshtein Similarity Score : 6
```

Cosine

In [16]:
```
Cos_TI = [td.cosine.normalized_similarity(Twitter[i],Instagram[i]) for i in range(len(df))]
```

In [17]:
```
fig = px.histogram(Cos_TI,color_discrete_sequence=['coral'])
fig.update_xaxes(title="Normalised Similarity Score",tickmode = 'array',tickvals = [i/10 for i in range(11)])
fig.update_yaxes(title="Number of Username Pairs")
fig.update_layout(title="Cosine Similarity Scores for Twitter-Instagram Usernames",bargap=0.01,showlegend=False)
fig.show()

print("Mean of Cosine Similarity Score :",round(pd.Series(Cos_TI).mean(),3))
print("Standard Deviation of Cosine Similarity Score :",round(pd.Series(Cos_TI).std(),3))
print("Zero Cosine Similarity Score :",Cos_TI.count(0))
```



```
Mean of Cosine Similarity Score : 0.869
Standard Deviation of Cosine Similarity Score : 0.235
Zero Cosine Similarity Score : 2
```

Comparison

In [18]:
```
L = 0
C = 0
W = 0

for i in range(len(df)):
    if Lev_TI[i]>Cos_TI[i]:
        L+=1
    elif Lev_TI[i]<Cos_TI[i]:
        C+=1
    else:
        W+=1

print("Levenshtein scored more in "+str(L)+" cases")
print("Cosine scored more in "+str(C)+" cases")
print("Both scored same in "+str(W)+" cases")
```

```
Levenshtein scored more in 6 cases
Cosine scored more in 187 cases
Both scored same in 216 cases
```