In [1]:

```
import pandas as pd
import numpy as np
import scipy.stats as stats

df=pd.read_csv('./Metadata.csv')
df
```

Out[1]:

	Mean	SD	Chart-Category	Chart-type	Domain	Source	Topic	Sentiment	Purpose
0	-1.65	2.272163	Grid	Heatmap	Infographics	Information is Beautiful	Health	2	2
1	5.74	2.086820	Point	Scatter	Blog	Multiple Views	Demographics	3	3
2	3.08	2.018346	Map + Area	Choropleth + Stacked	Government	Washington Census	Demographics	3	3
3	6.21	2.668193	Bar	Bar	Blog	Flowing Data	Demographics	3	3
4	3.06	2.045362	Area	Proportional	News	Wall Street Journal	Sports	3	3
...	...	...	...	...	...	...	...	...	...
495	-6.35	2.378954	Map	Plotted	Blog	Nightingale	History	2	2
496	-3.98	2.533647	Area + Diagram	Area + Isotype	Blog	Nightingale	Resources	2	2
497	-5.92	2.056095	Diagram	Isotype	Infographics	Information is Beautiful	Environment	2	2
498	-2.23	2.229485	Bar	Bar	Infographics	Information is Beautiful	Environment	3	2
499	-4.21	2.357637	Map	Plotted	News	New York Times	Weather	3	2

500 rows × 41 columns

In [2]:

```
# Regression Results for Ratings Scores in Expt 1

import scipy.stats as stats

import csv
from matplotlib import pyplot as plt
import numpy as np
import statistics
import pandas as pd

import sklearn
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression

import statsmodels.api as sm
from statsmodels.sandbox.regression.predstd import wls_prediction_std

from sklearn.decomposition import FactorAnalysis, PCA
from sklearn.preprocessing import StandardScaler

from sklearn.model_selection import KFold
from sklearn.linear_model import BayesianRidge
from numpy import mean
from numpy import var
from math import sqrt

def cohend(d1, d2):
    n1, n2 = len(d1), len(d2)
    s1, s2 = var(d1, ddof=1), var(d2, ddof=1)
    s = sqrt(((n1 - 1) * s1 + (n2 - 1) * s2) / (n1 + n2 - 2))
    u1, u2 = mean(d1), mean(d2)
    return (u1 - u2) / s
```

In [3]:

```
# Chart-type

# Isotype, Chernoff, Illustrations
y_val=[]
for i in range(len(df)):
    if('Isotype' in df.iloc[i]['Chart-type'] or 'Chernoff' in df.iloc[i]['Chart-type'] o
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Isotype, Chernoff, Illustration : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Area
y_val=[]
for i in range(len(df)):
    if('Area' in df.iloc[i]['Chart-Category']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Area : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Map
y_val=[]
for i in range(len(df)):
    if('Map' in df.iloc[i]['Chart-Category']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Map : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Map with low annotation
y_val=[]
for i in range(len(df)):
    if('Map' in df.iloc[i]['Chart-Category'] and df.iloc[i]['Text-Vol']==1):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Map, Low Annotation : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Radial Charts (chord diagrams, donut charts, timelines, radial bar charts, etc.)
y_val=[]
for i in range(len(df)):
    if('Circle' in df.iloc[i]['Chart-Category'] or 'Radial' in df.iloc[i]['Chart-type'])
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Radial Charts : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Bar
y_val=[]
for i in range(len(df)):
    if('Bar' in df.iloc[i]['Chart-Category']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Bar : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Point
y_val=[]
for i in range(len(df)):
    if('Point' in df.iloc[i]['Chart-Category']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Point : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Line
y_val=[]
for i in range(len(df)):
    if('Line' in df.iloc[i]['Chart-Category']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Line : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Distribution
y_val=[]
for i in range(len(df)):
    if('Distribution' in df.iloc[i]['Chart-Category']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Distribution : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Grid
y_val=[]
for i in range(len(df)):
    if('Grid' in df.iloc[i]['Chart-Category']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
```

```
print('Grid : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Network
y_val=[]
for i in range(len(df)):
    if('Network' in df.iloc[i]['Chart-Category']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Network : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Table
y_val=[]
for i in range(len(df)):
    if('Table' in df.iloc[i]['Chart-Category']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Table : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Tree
y_val=[]
for i in range(len(df)):
    if('Tree' in df.iloc[i]['Chart-Category']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Tree : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************
```

```
Isotype, Chernoff, Illustration :
p-value : 0.00
R2 : 0.78
```

```
Cohen's d : 0.86
*****************************************************************************************
*********
Area
p-value : 0.00
R2 : 0.75
Cohen's d : 0.72
*****************************************************************************************
*********
Map :
p-value : 0.00
R2 : 0.71
Cohen's d : 0.76
*****************************************************************************************
*********
Map, Low Annotation :
p-value : 0.00
R2 : 0.83
Cohen's d : 0.74
*****************************************************************************************
*********
Radial Charts :
p-value : 0.00
R2 : 0.79
Cohen's d : 0.80
*****************************************************************************************
*********
Bar :
p-value : 0.00
R2 : 0.62
Cohen's d : 0.68
*****************************************************************************************
*********
Point :
p-value : 0.00
R2 : 0.67
Cohen's d : 0.64
*****************************************************************************************
*********
Line :
p-value : 0.00
R2 : 0.62
Cohen's d : 0.55
*****************************************************************************************
*********
Distribution :
p-value : 0.00
R2 : 0.69
Cohen's d : 0.51
*****************************************************************************************
*********
Grid :
p-value : 0.00
R2 : 0.66
Cohen's d : 0.67
*****************************************************************************************
*********
Network :
p-value : 0.00
R2 : 0.71
Cohen's d : 0.69
*****************************************************************************************
*********
Table :
p-value : 0.00
R2 : 0.53
```

```
Cohen's d : 0.41
****************************************************************************************
*********
Tree :
p-value : 0.00
R2 : 0.51
Cohen's d : 0.38
****************************************************************************************
*********
```

In [4]:

```
# Photorealistic

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Photorealistic']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Photorealistic : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Human Recognizable Objects

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['HRO']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Human Recognizable Objects : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Human Depiction

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['HD']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Human Depiction : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Skeuomorphism

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Skeumorphic']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Skeuomorphism : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Title

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Title']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Title : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Axis

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Axes']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Axis : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Key
```

```
y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Key']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Key : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("**********************************************************************************

# Title and Key

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Title'] and 'y' in df.iloc[i]['Key']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Title and Key : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("**********************************************************************************

# Text Volume

X=df['Mean']
Y=df['Text-Vol']
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Text Volume : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("**********************************************************************************

# Data Labels
y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Labels']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Data Labels : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Data Ink Ratio

X=df['Mean']
Y=df['DIR']
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Data Ink Ratio : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************


# Visual Density

X=df['Mean']
Y=df['Density']
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Visual Density : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Data Scale

X=df['Mean']
Y=df['Data-Vol']
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Data Scale : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Message Redundancy

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Msg-Red']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Message Redundancy : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Data Redundancy

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Data-Red']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Data Redundancy : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Highlights

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Highlights']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Highlights : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Arrows

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Arrows']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Arrows : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Color

y_val=[]
for i in range(len(df)):
    if('low' in df.iloc[i]['Color']):
        y_val.append(1)
    elif('medium' in df.iloc[i]['Color']):
        y_val.append(2)
    else:
        y_val.append(3)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Color : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Background Color
X=df['Mean']
Y=df['BGC']
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Background Color : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# 3D

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['3D']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('3D : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Gridlines

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Gridlines']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Gridlines : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Caption

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Caption']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Gridlines : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Data Source

y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Source.1']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Data Source : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************

# Dimensionality
y_val=[]
for i in range(len(df)):
    if('low' in df.iloc[i]['Dim']):
        y_val.append(1)
    elif('medium' in df.iloc[i]['Dim']):
        y_val.append(2)
    else:
        y_val.append(3)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Dimensionality : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("********************************************************************************
# Pictorial Units
y_val=[]
for i in range(len(df)):
    if('y' in df.iloc[i]['Pictorial-Units']):
        y_val.append(1)
    else:
        y_val.append(0)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Pictorial Units : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
print("*********************************************************************************

# Domain
y_val=[]
for i in range(len(df)):
    if('Infographics' in df.iloc[i]['Domain']):
        y_val.append(1)
    elif('Blog' in df.iloc[i]['Domain']):
        y_val.append(2)
    elif('Government' in df.iloc[i]['Domain']):
        y_val.append(3)
    elif('News' in df.iloc[i]['Domain']):
        y_val.append(4)
    elif('Media' in df.iloc[i]['Domain']):
        y_val.append(5)
    elif('Scientific' in df.iloc[i]['Domain']):
        y_val.append(6)
    else:
        y_val.append(7)
X=df['Mean']
Y=y_val
d=cohend(Y,X)
X=sm.add_constant(X)
model = sm.OLS(Y,X)
results = model.fit()
print('Domain : ')
print("p-value : %.2f"%results.pvalues[0])
print("R2 : %.2f"%results.rsquared)
print("Cohen's d : %.2f"%d)
```

```
Photorealistic :
p-value : 0.00
R2 : 0.84
Cohen's d : 0.86
****************************************************************************************
*********
Human Recognizable Objects :
p-value : 0.00
R2 : 0.85
Cohen's d : 0.83
****************************************************************************************
*********
Human Depiction :
p-value : 0.00
R2 : 0.83
Cohen's d : 0.83
****************************************************************************************
*********
Skeuomorphism :
```

```
p-value : 0.00
R2 : 0.81
Cohen's d : 0.79
****************************************************************************************
*********
Title :
p-value : 0.00
R2 : 0.85
Cohen's d : 0.89
****************************************************************************************
*********
Axis :
p-value : 0.00
R2 : 0.89
Cohen's d : 0.90
****************************************************************************************
*********
Key :
p-value : 0.00
R2 : 0.75
Cohen's d : 0.81
****************************************************************************************
*********
Title and Key :
p-value : 0.00
R2 : 0.79
Cohen's d : 0.82
****************************************************************************************
*********
Text Volume :
p-value : 0.00
R2 : 0.89
Cohen's d : 0.92
****************************************************************************************
*********
Data Labels :
p-value : 0.00
R2 : 0.78
Cohen's d : 0.79
****************************************************************************************
*********
Data Ink Ratio :
p-value : 0.00
R2 : 0.59
Cohen's d : 0.43
****************************************************************************************
*********
Visual Density :
p-value : 0.00
R2 : 0.66
Cohen's d : 0.56
****************************************************************************************
*********
Data Scale :
p-value : 0.00
R2 : 0.07
Cohen's d : 0.12
****************************************************************************************
*********
Message Redundancy :
p-value : 0.00
R2 : 0.81
Cohen's d : 0.77
****************************************************************************************
*********
Data Redundancy :
```

```
p-value : 0.00
R2 : 0.09
Cohen's d : 0.28
*****************************************************************************************
*********
Highlights :
p-value : 0.00
R2 : 0.75
Cohen's d : 0.71
*****************************************************************************************
*********
Arrows :
p-value : 0.00
R2 : 0.68
Cohen's d : 0.70
*****************************************************************************************
*********
Color :
p-value : 0.00
R2 : 0.71
Cohen's d : 0.73
*****************************************************************************************
*********
Background Color :
p-value : 0.00
R2 : 0.81
Cohen's d : 0.77
*****************************************************************************************
*********
3D :
p-value : 0.00
R2 : 0.53
Cohen's d : 0.33
*****************************************************************************************
*********
Gridlines :
p-value : 0.00
R2 : 0.48
Cohen's d : 0.29
*****************************************************************************************
*********
Caption :
p-value : 0.00
R2 : 0.72
Cohen's d : 0.78
*****************************************************************************************
*********
Data Source :
p-value : 0.00
R2 : 0.05
Cohen's d : 0.17
*****************************************************************************************
*********
Dimensionality :
p-value : 0.00
R2 : 0.05
Cohen's d : 0.11
*****************************************************************************************
*********
Pictorial Units :
p-value : 0.00
R2 : 0.85
Cohen's d : 0.86
*****************************************************************************************
*********
Domain :
```

```
p-value : 0.00
R2 : 0.01
Cohen's d : 0.08
```

In [ ]:

```

```