

Python 画图

1、t-sne 图

```
import matplotlib.pyplot as plt
fig, axes = plt.subplots(nrows=2,ncols=2, figsize=(12,10))
axes[0][0].set_xlabel('(a) E-GraphSAGE', fontsize=16)
axes[0][1].set_xlabel('(b) E-GraphSAGE M', fontsize=16)
axes[1][0].set_xlabel('(c) GAT', fontsize=16)
axes[1][1].set_xlabel('(d) E-ResGAT', fontsize=16)
sns.scatterplot(x=X_embed[0][:,0], y=X_embed[0][:,1], hue=y[test], legend='brief', palette = palette, ax=axes[0][0])
sns.scatterplot(x=X_embed[1][:,0], y=X_embed[1][:,1], hue=y[test], legend=False, palette = palette, ax=axes[0][1])
sns.scatterplot(x=X_embed[2][:,0], y=X_embed[2][:,1], hue=y[test], legend=False, palette = palette, ax=axes[1][0])
sns.scatterplot(x=X_embed[3][:,0], y=X_embed[3][:,1], hue=y[test], legend=False, palette = palette, ax=axes[1][1])
```

图 1 t-sne 代码

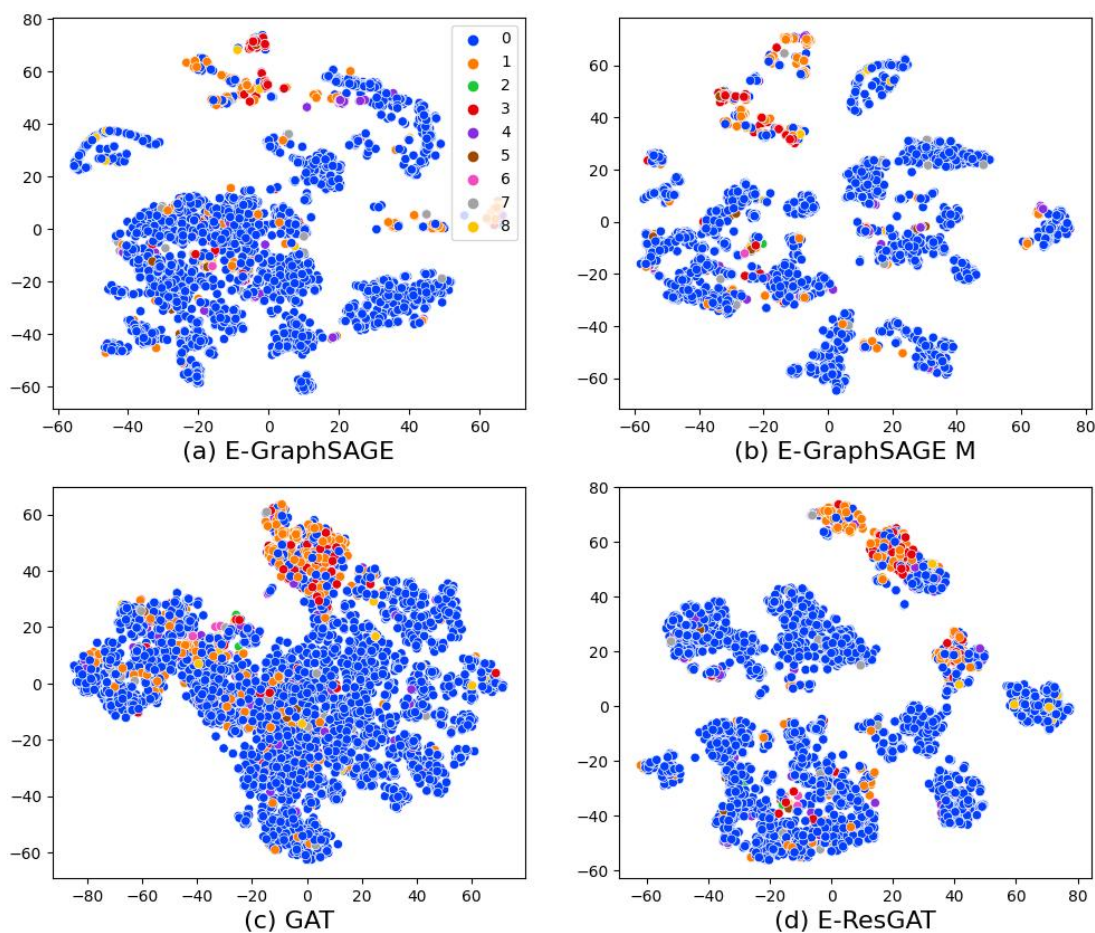


图 2 t-sne 图

2、热力图

```
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt

data = np.random.rand(8,8)

plt.figure(figsize=(10,8))
sns.set(font_scale=1.5)
sns.set_style("whitegrid")

sns.heatmap(data, cmap='coolwarm', annot=True, fmt='.1f', linewidths=1, cbar_kws={'shrink':.5})

plt.title('Heatmap')

plt.show()
```

✓ 0.3s

图 3 热力图源代码

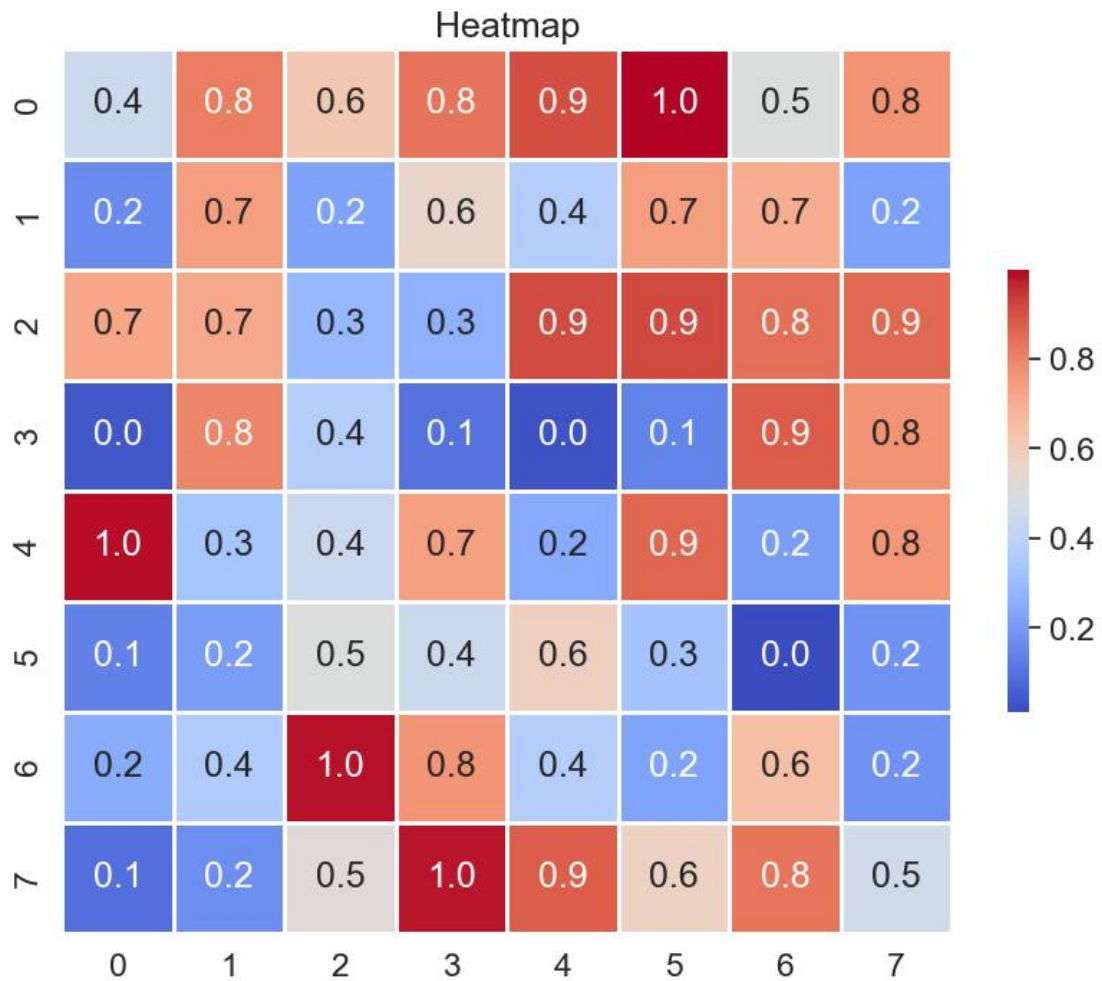


图 4 热力图

3、全连接神经网络

```
import itertools
import matplotlib.pyplot as plt
import networkx as nx

subset_sizes = [5, 5, 4, 3, 2, 4, 4, 3]
subset_colors = [
    "gold",
    "violet",
    "violet",
    "violet",
    "violet",
    "blue",
    "blue",
    "darkorange",
]

def multilayered_graph(*subset_sizes):
    extents = nx.utils.pairwise(itertools.accumulate((0,)+subset_sizes))
    layers = [range(start, end) for start, end in extents]
    G = nx.Graph()
    for i, layer in enumerate(layers):
        G.add_nodes_from(layer, layer=i)
    for layer1, layer2 in nx.utils.pairwise(layers):
        G.add_edges_from(itertools.product(layer1, layer2))
    return G

G = multilayered_graph(*subset_sizes)
color = [subset_colors[data["layer"]] for v, data in G.nodes(data=True)]
pos = nx.multipartite_layout(G, subset_key="layer")
plt.figure(figsize=(8,8))
nx.draw(G, pos, node_color=color, with_labels=False)
plt.show()
```

图 5 全连接神经网络代码

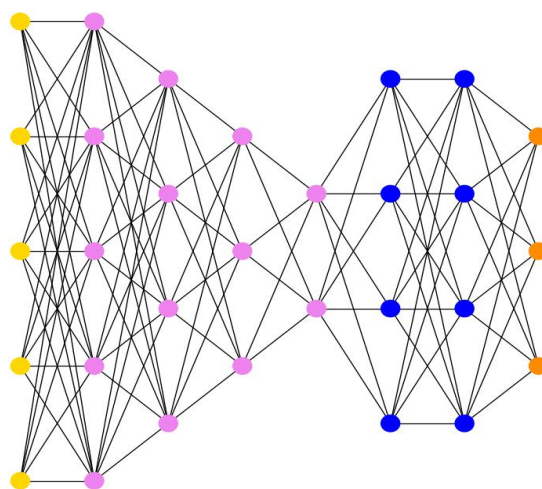


图 6 全连接神经图

4、散点图及回归线

```
import seaborn as sns
import pandas as pd

np.random.seed(42)
data = pd.DataFrame({
    'X': np.random.normal(50, 15, 100),
    'Y': np.random.normal(50, 15, 100) * 0.5 + np.random.normal(0, 5, 100)
})

plt.figure(figsize=(10, 6))
sns.regplot(x='X', y='Y', data=data,
            scatter_kws={'alpha':0.6, 'color':'blue'},
            line_kws={'color':'red', 'linestyle':'--'})

plt.title('X与Y的散点图及回归线', fontsize=14)
plt.xlabel('X', fontsize=12)
plt.ylabel('Y', fontsize=12)

plt.grid(True, linestyle=':', alpha=0.5)
plt.show()
```

图 7 散点图源代码

