Miscellaneous

Statistical or mathematical method for identifying structure in data

Sample dataset

MedQueryID	TermID
MQ1	T1
MQ1	T2
MQ2	T2
MQ2	T3
MQ3	T1
MQ3	T3
MQ4	T1
MQ4	T2
MQ5	T4
MQ5	T5
MQ6	T4
MQ6	T6
MQ7	T5
MQ7	T6
MQ8	T1
MQ8	T3
MQ9	T2
MQ9	T3
MQ10	T6
MQ10	T7

1️⃣ Step 1: Create binary incidence matrix

	T1	T2	T3	T4	T5	T6	T7
MQ1	1	1	0	0	0	0	0
MQ2	0	1	1	0	0	0	0
MQ3	1	0	1	0	0	0	0
MQ4	1	1	0	0	0	0	0
MQ5	0	0	0	1	1	0	0
MQ6	0	0	0	1	0	1	0
MQ7	0	0	0	0	1	1	0
MQ8	1	0	1	0	0	0	0
MQ9	0	1	1	0	0	0	0
MQ10	0	0	0	0	0	1	1

2️⃣ Step 2: Cosine Distance Calculation

Cosine distance formula:

$$d(v_i, v_j) = 1 - (v_i \cdot v_j) / (||v_i|| * ||v_j||)$$

Norms of vectors (all sqrt(2) ≈ 1.414)

Vector norms:

MQ1=1.414, MQ2=1.414, MQ3=1.414, MQ4=1.414, MQ5=1.414, MQ6=1.414, MQ7=1.414, MQ8=1.414, MQ9=1.414, MQ10=1.414

Dot products and distances:

MQ1-MQ2: dot=1 → d=1-1/2=0.5
MQ1-MQ3: dot=1 → d=0.5
MQ1-MQ4: dot=2 → d=1-2/2=0
MQ1-MQ5: dot=0 → d=1
MQ1-MQ6: dot=0 → d=1
MQ1-MQ7: dot=0 → d=1
MQ1-MQ8: dot=1 → d=0.5
MQ1-MQ9: dot=1 → d=0.5
MQ1-MQ10: dot=0 → d=1

MQ2-MQ3: dot=1 → d=0.5
MQ2-MQ4: dot=1 → d=0.5
MQ2-MQ5: dot=0 → d=1
MQ2-MQ6: dot=0 → d=1
MQ2-MQ7: dot=0 → d=1
MQ2-MQ8: dot=1 → d=0.5
MQ2-MQ9: dot=2 → d=0
MQ2-MQ10: dot=0 → d=1

MQ3-MQ4: dot=1 → d=0.5
MQ3-MQ5: dot=0 → d=1
MQ3-MQ6: dot=0 → d=1
MQ3-MQ7: dot=0 → d=1
MQ3-MQ8: dot=2 → d=0
MQ3-MQ9: dot=1 → d=0.5
MQ3-MQ10: dot=0 → d=1

MQ4-MQ5: dot=0 → d=1
MQ4-MQ6: dot=0 → d=1
MQ4-MQ7: dot=0 → d=1
MQ4-MQ8: dot=1 → d=0.5
MQ4-MQ9: dot=1 → d=0.5
MQ4-MQ10: dot=0 → d=1

MQ5-MQ6: dot=1 → d=0.5
MQ5-MQ7: dot=1 → d=0.5
MQ5-MQ8: dot=0 → d=1
MQ5-MQ9: dot=0 → d=1
MQ5-MQ10: dot=0 → d=1

MQ6-MQ7: dot=1 → d=0.5
MQ6-MQ8: dot=0 → d=1
MQ6-MQ9: dot=0 → d=1
MQ6-MQ10: dot=1 → d=0.5

MQ7-MQ8: dot=0 → d=1
MQ7-MQ9: dot=0 → d=1
MQ7-MQ10: dot=1 → d=0.5

MQ8-MQ9: dot=1 → d=0.5
MQ8-MQ10: dot=0 → d=1

MQ9-MQ10: dot=0 → d=1

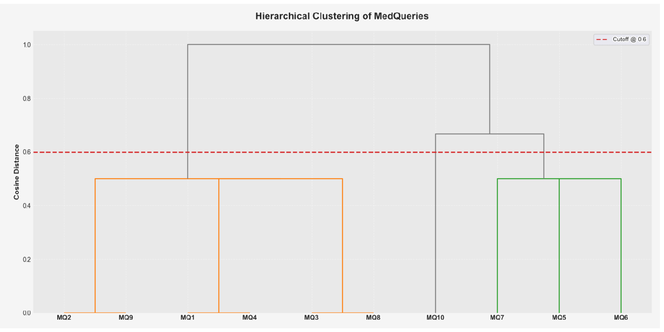
3️⃣ Step 3: Hierarchical Clustering (Average Linkage)

```
# Step 3a: Merge closest pairs (distance=0)
MQ1 + MQ4 → C1={MQ1,MQ4} height=0
MQ2 + MQ9 → C2={MQ2,MQ9} height=0
MQ3 + MQ8 → C3={MQ3,MQ8} height=0

# Step 3b:  Merge MQ5-MQ6 → C4={MQ5,MQ6} height=0.5
# Step 3c:  Merge MQ7-MQ10 → C5={MQ7,MQ10} height=0.5
# Step 3d:  Merge C4-C5 → C6={MQ5,MQ6,MQ7,MQ10} height=0.625

# Step 3e:  Merge C1-C2-C3 → C7={MQ1,MQ2,MQ3,MQ4,MQ8,MQ9} height=0.75
# Step 3f:  Merge C7-C6 → Final cluster {all MQs} height=1.0
```

4️⃣ Step 4: Dendrogram



Code:

```
from sklearn.metrics import pairwise_distances
from scipy.cluster.hierarchy import linkage, dendrogram
from scipy.spatial.distance import squareform
import matplotlib.pyplot as plt

# ----------------------------
# 1) Build the incidence matrix
# ----------------------------
cols = ["T1","T2","T3","T4","T5","T6","T7"]
data = [
[1,1,0,0,0,0,0],  # MQ1
[0,1,1,0,0,0,0],  # MQ2
[1,0,1,0,0,0,0],  # MQ3
[1,1,0,0,0,0,0],  # MQ4
[0,0,0,1,1,0,0],  # MQ5
[0,0,0,1,0,1,0],  # MQ6
[0,0,0,0,1,1,0],  # MQ7
[1,0,1,0,0,0,0],  # MQ8
[0,1,1,0,0,0,0],  # MQ9
[0,0,0,0,0,1,1],  # MQ10
]
mq_labels = [f"MQ{i}" for i in range(1, 11)]

incidence = pd.DataFrame(data, index=mq_labels, columns=cols)
print("Incidence matrix:\n", incidence, "\n")

# ----------------------------
# 2) Compute distance matrix
# ----------------------------
# Use 'cosine' distance (1 - cosine similarity). For binary data Jaccard is also common.
dist_matrix = pairwise_distances(incidence.values, metric="cosine")
print("Pairwise distance matrix (cosine):\n", np.round(dist_matrix, 3), "\n")

# Convert to condensed distance vector (required by linkage)
condensed_dist = squareform(dist_matrix, checks=False)

# ----------------------------
# 3) Hierarchical clustering (average linkage)
# ----------------------------
Z = linkage(condensed_dist, method="average")  # 'single','complete','average','ward',
...
print("Linkage matrix (first 6 merges):\n", np.round(Z[:6], 3), "\n")
# Linkage columns: [idx1, idx2, distance, sample_count]

# ----------------------------
# 4) Plot dendrogram
# ----------------------------
plt.figure(figsize=(10, 6))
dn = dendrogram(
Z,
labels=mq_labels,
leaf_rotation=0,
leaf_font_size=10,
color_threshold=0.6,   # adjust to color clusters at certain height
)

plt.title("Hierarchical Clustering Dendrogram (average linkage, cosine dist)")
plt.ylabel("Distance")
plt.tight_layout()
plt.show()

# ----------------------------
# Optional: save figure
# ----------------------------
# plt.savefig("mq_dendrogram.png", dpi=300)
```

Made with Napkin