Boolean Analysis of Gene Expression Project (Code Explanation)

Step 1: Load Dataset

We loaded GSE2034_series_matrix.txt (expression matrix) from GEO database.

df = pd.read_csv(...)

-> Reads the expression matrix file into a DataFrame.

Step 2: Data Cleaning

---

df = df.apply(pd.to_numeric, errors='coerce')

-> Converts all values to numeric, replaces non-numeric with NaN.

df = df.dropna(how='all')

-> Drops any gene rows that are completely empty.

Step 3: Boolean Conversion

---

gene_means = df.mean(axis=1)

-> Calculates mean expression of each gene across all samples.

boolean_df = df.gt(gene_means, axis=0).astype(int)

-> For each gene/sample:

If expression > mean: mark as ON (1)

Else: mark as OFF (0)

Step 4: Extract Metadata (Bone Relapse Status)

Extracted bone relapse labels from annotation line:

- 1 = Relapse YES
- 0 = Relapse NO

Mapped samples to relapse status using boolean_df column names.

Step 5: Calculate ON Percentage

---

on_in_yes = boolean_df[relapse_yes_samples].sum(axis=1) / len(relapse_yes_samples)

on_in_no = boolean_df[relapse_no_samples].sum(axis=1) / len(relapse_no_samples)

-> Calculates % of patients where gene is ON in each group.

Step 6: Select Boolean Marker Genes

---

marker_mask = (on_in_yes >= 0.6) & (on_in_no <= 0.4)
-> Selects genes ON in >=60% relapse samples AND OFF in <=40% non-relapse samples.

marker_genes = boolean_df.loc[marker_mask]
-> Creates DataFrame of selected marker genes.

Step 7: Map Probe IDs to Gene Symbols
---
mygene used to map Affymetrix probe IDs (e.g., 1007_s_at) to official gene names (e.g., DDR1).

Step 8: Pathway Enrichment
---
enr = enrichr(...)
-> Runs pathway enrichment (KEGG, GO) on gene list using gseapy.
-> Outputs biological pathways linked to your marker genes.

Conclusion

This pipeline converts raw gene expression into a simple binary ON/OFF matrix, compares groups, finds relapse-associated genes, and links them to known pathways.