Gene Expression Analysis with GTEx (STT 303 Final Project)

Overview

This project was designed to explore patterns in human gene expression using RNA-seq data from the [GTEx (Genotype-Tissue Expression) Project](https://gtexportal.org/home/). The analysis aimed to investigate how gene expression varies across tissues, sexes, and age groups using both **supervised and unsupervised learning methods** in R.

Goals

- Identify highly variable genes across tissue types.
- Explore sex-linked differences in expression.
- Predict **age** from expression levels using regression.
- Classify **tissue types** and **biological sex** using gene profiles.
- Perform PCA and clustering to uncover structure in gene expression.

Tools & Technologies

- **Language**: R
- **Libraries**: tidyverse, caret, cluster, pROC, factoextra, heatmaply, matrixStats

---

What Went Wrong

Despite having a well-structured pipeline, the project encountered several key issues that blocked full

execution:

- **Memory Limits**: Attempting to load and transform high-dimensional gene expression data (10,000+ genes) strained system resources and caused R to crash or freeze during matrix operations.
- **Data Conversion Errors**: Some variables could not be coerced to numeric, leading to warnings and disrupted PCA/regression steps.
- **Missing or Misaligned Data**: Certain genes or metadata variables (e.g., AGE) were missing or inconsistently formatted, causing modeling steps to fail.
- **Regression Instability**: In some tissue subsets, the response variable (AGE) lacked enough variability to support model training.
- **Plotting Limits**: Heatmap and PCA plots were skipped when not enough valid data points remained after filtering.

What Worked

- Successfully read in and subset GTEx gene expression and metadata.
- Performed log transformation and variance-based filtering on genes.
- Merged expression and metadata into a tidy format.
- Implemented safe PCA and heatmap fallback checks to avoid crashes.
- Laid groundwork for machine learning models using `caret`.

How I Would Fix It

For a more successful and reproducible analysis in the future, I would:

- **Subset by Tissue Type Early**: Working with one tissue at a time would drastically reduce dimensionality and memory use.
- **Use Sparse Matrices or DelayedArray**: These allow memory-efficient processing of large bioinformatics data.
- **Add Robust Error Handling**: Check for missing or non-numeric entries before modeling or plotting.
- **Incremental Debugging**: Run and verify each step independently using small gene/sample subsets before scaling.
- **Test on HPC or Cloud Resources**: Use RStudio Cloud or a university HPC server with more memory for full dataset analysis.
- **Modularize Code**: Break down steps into reusable functions or scripts for clarity and debugging ease.

Files

- `FINAL 222.R` Full R script for analysis pipeline
- GTEx data and metadata (downloaded separately from the GTEx portal)

Author

Amina Aboulhana
Senior, STT 303: Data Science with R
Spring 2025