CCC-GPU: A graphics processing unit (GPU)-accelerated nonlinear correlation coefficient for large-scale transcriptomic analyses

Authors

- **Haoyu Zhang**
  0009-0005-6025-0217 · haoyu-zc
  Department of Biomedical Informatics, University of Colorado Anschutz, Aurora, CO 80045, USA

- **Kevin Fotso**
  0009-0002-2196-6157 · kf-cuanschutz
  Office of Information Technology, University of Colorado Anschutz, Aurora, CO 80045, USA

- **Marc Subirana-Granés**
  0000-0003-3934-839X · msubirana · msubirana20
  Department of Biomedical Informatics, University of Colorado Anschutz, Aurora, CO 80045, USA

- **Milton Pividori** ✉
  0000-0002-3035-4403 · miltondp · miltondp · @miltondp@genomic.social
  Department of Biomedical Informatics, University of Colorado School of Medicine, Aurora, CO 80045, USA; Colorado Center for Personalized Medicine, University of Colorado Anschutz, Aurora, CO 80045, USA · Funded by The National Human Genome Research Institute (K99/R00 HG011898); The Eunice Kennedy Shriver National Institute of Child Health and Human Development (R01 HD109765)

✉ — Correspondence possible via GitHub Issues or email to Milton Pividori <milton.pividori@cuanschutz.edu>.

Abstract

Motivation: Identifying meaningful patterns in complex biological data necessitates correlation coefficients capable of capturing diverse relationship types beyond simple linearity. Furthermore, efficient computational tools are crucial for handling the ever-increasing scale of biological datasets.

Results: We introduce CCC-GPU, a high-performance, GPU-accelerated implementation of the Clustermatch Correlation Coefficient (CCC). CCC-GPU computes correlation coefficients for mixed data types, effectively detects nonlinear relationships, and offers significant speed improvements over its predecessor.

Availability: The source code of CCC-GPU is openly available on GitHub (https://github.com/pivlab/ccc-gpu) and archived on Zenodo (https://doi.org/10.5281/zenodo.18310318), distributed under the BSD-2-Clause Plus Patent License.

Introduction

Correlation coefficients are fundamental tools for uncovering meaningful patterns within data. While traditional measures like Pearson and Spearman are adept at capturing linear and monotonic

relationships, newer methodologies, such as the Clustermatch Correlation Coefficient (CCC) [1] and the Maximal Information Coefficient (MIC) [2,3] have emerged to detect a broader spectrum of associations. These coefficients represent three distinct statistical assumptions: Pearson captures linear relationships, Spearman measures monotonic patterns using a nonparametric approach, and CCC detects general nonlinear associations. We centered our analysis on these three to effectively cover the most common correlation scenarios without redundancy; notably, we excluded Kendall's τ due to its high correlation with Spearman's ρ and ~540× slower computation.

The CCC is a clustering-based statistic designed to identify both linear and nonlinear patterns. Previously, we demonstrated CCC's utility in analyzing gene expression data from GTEx, showcasing its robustness to outliers and its ability to detect both strong linear relationships and biologically significant nonlinear patterns often missed by conventional coefficients [1]. Unlike MIC, CCC accommodates both numerical and categorical data types and is up to two orders of magnitude faster. However, despite leveraging CPU multi-threading for acceleration, the original CCC implementation can still be computationally expensive for large datasets. For instance, in the original study [1], we used only the top 5,000 most variable genes in a single tissue (whole blood) to ease computation.

Here, we introduce a new implementation, CCC-GPU, that harnesses the power of NVIDIA CUDA for GPU programming. We have computed CCC-GPU values for all ~50,000 genes in the Genotype-Tissue Expression (GTEx) v8 dataset [4] across all 54 tissues in a fraction of the time that would have been needed using the original CCC implementation. This advancement achieves a substantial speedup over the original CPU-based implementation, making comprehensive correlation analysis of large biological datasets more practical and efficient.

Methods

CCC was originally developed in Python, as detailed in the original publication [1]. We provide the algorithm equation and pseudocode in Supplementary Information E1 and S1, respectively. Profiling CCC revealed that Adjusted Rand Index (ARI) computation [5] was the primary performance bottleneck in our experiments, accounting for 74–91% of total runtime depending on workload size (Figures S2–S4). This is expected as CCC heavily relies on ARI computation for internal data clustering. For example, when computing the pairwise correlations for one GTEx v8 tissue with ~50,000 genes using default CCC parameters, approximately 1.5 billion ARI calculations are needed. Since individual ARI calculations have no data dependencies and require minimal synchronization, they are ideal candidates for utilizing graphics processing units (GPUs), which contain thousands of compute units designed for massively parallel operations.

We developed a GPU-accelerated implementation by rewriting the core ARI computation module in CUDA C++ (requiring an NVIDIA GPU with CUDA Compute Capability 8.6 or higher, i.e., RTX 30-series or newer). While existing CUDA-based Python libraries were evaluated, native C++ code provides greater flexibility and complete access to CUDA's feature set [6]. In this implementation, we replaced the Python-based ARI computation logic with CUDA C++ kernel functions. Each feature (gene) pair's CCC correlation involves multiple ARI calculations, which we assigned unique global indices. These calculations are carefully mapped to kernel functions by their global indices, then executed concurrently across multiple GPU cores. This required rebuilding internal algorithms that Python libraries such as NumPy conveniently provide. An orchestration layer processes inputs in batches, allocating up to 4.5 GB of GPU memory per batch—well within the capacity of modern GPUs, enabling memory-efficient streaming of large-scale matrices. Within each batch, kernel functions use shared memory for caching and efficient inter-thread communication, device memory buffers are reused across batches to avoid repeated allocation overhead, and results are transferred to host memory upon batch completion. To minimize overhead, device synchronization events occur only at critical

points: after each ARI batch computation completes and during p-value calculations to ensure kernel completion before data transfer. In summary, our custom solution incorporates GPU architecture optimizations including explicit memory management, tailored caching strategies, and coordinated CPU-GPU communication to maximize performance. Future software versions will expose more parameters for hardware-specific fine-tuning.

We then used pybind11 [7] to integrate the CUDA C++ backend with the existing Python framework, ensuring full compatibility with the original CCC interface. This hybrid approach preserves the original system's comprehensive feature set while enabling GPU acceleration, facilitating thorough testing and validation.

Results

We conducted a comprehensive comparative evaluation of Pearson, Spearman, and CCC-GPU using simulated data and all genes and tissues in the GTEx v8 dataset. The evaluation was performed on an AMD Ryzen Threadripper 7960X CPU with an NVIDIA RTX 4090 GPU, representing a typical high-performance workstation configuration. The original analysis and benchmarking code is available in the GitHub repository (https://github.com/pivlab/ccc-gpu/tree/main/analysis).

We compared the computational complexity of CCC-GPU (using the GPU) and the original CCC (using 24 CPU cores) when analyzing each tissue in the GTEx dataset. For whole blood (56,200 genes, 755 samples), CCC-GPU demonstrated a 37x speedup. On average, the original CCC required approximately 6 hours to process one GTEx tissue. This acceleration enabled the complete analysis of all 54 GTEx tissues in just 8 hours on a single machine, compared to ~312 hours (~13 days) that would be needed with the original implementation. Consistent performance gains were observed across all tissue types, confirming CCC-GPU's broad applicability.

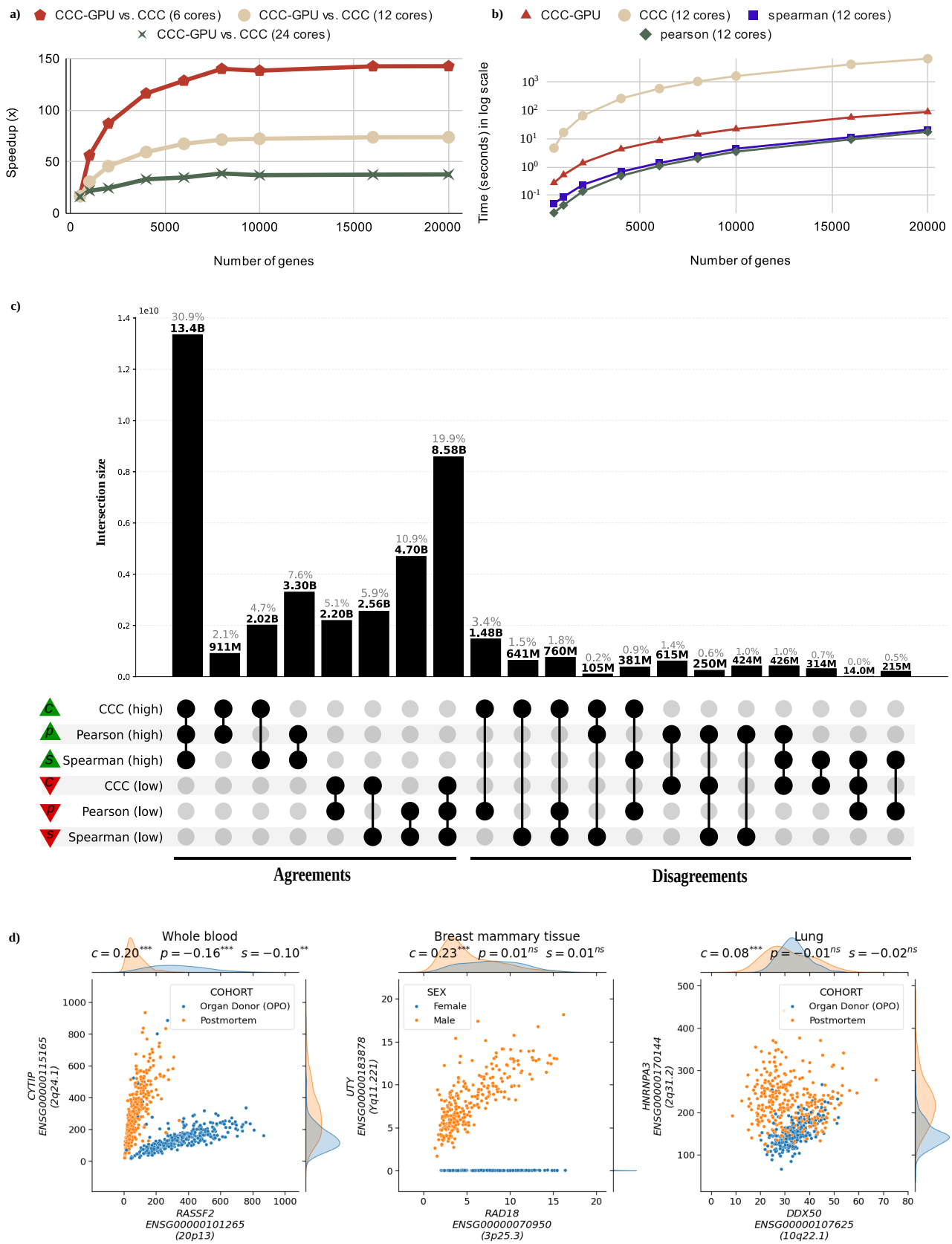


Figure 1: **a)** CCC-GPU scalability analysis showing speedup relative to CPU-based CCC across varying gene counts (1,000 samples fixed) and CPU core configurations. See Supplementary Table S1 for detailed metrics. **b)** Runtime comparison of correlation methods (Pearson, Spearman, original CCC, CCC-GPU) demonstrating CCC-GPU's competitive performance despite its complexity to capture nonlinear patterns. Analysis used 12 CPU cores with varying gene counts (1,000 samples fixed). See Supplementary Table S2. **c)** UpSet plot showing correlation agreement (left) and disagreement (right) patterns of all gene pairs in all 54 GTEx v8 tissues among methods. Green triangles indicate high correlations (top 30%

gene pairs per tissue), and red triangles show low correlations (bottom 30%). We also created UpSet plots using permutation-based thresholds (see Supplementary Information and Supplementary Figure S5). UpSet plots for individual tissues are provided in Supplementary Figures S6 to S59 in panels c and d. **d)** Expression levels of selected gene pairs with associations across GTEx tissues: *RASSF2-CYTIP*, in whole blood, shows two clear and strong linear patterns across different sample subsets (postmortem versus organ donor). Although all coefficients are statistically significant, Pearson and Spearman detect the wrong pattern (i.e., their slopes are negative); *UTY-RAD18* shows a linear relationship only in male samples, whereas female samples exhibit a constant value of zero (since *UTY* is located in chromosome Y). *HNRNPA3-DDX50* demonstrates a case where one sample subset (organ donor) shows a linear relationship that is masked by another subset (postmortem) with no or weak relationship.

To further understand CCC-GPU's scalability with increasing data size, we performed benchmarks using synthesized input, comparing its speedup against the original CPU implementation. The results, illustrated in Figure 1a, demonstrate that CCC-GPU maintains a stable speedup trend as input size increases, confirming its robust scaling capabilities when handling large datasets. The different curves in Figure 1a simulate CCC-GPU's prospective speedup over CPU hardware with varying numbers of cores. We also compared the runtime of the original CCC, CCC-GPU, Pearson, and Spearman methods, as shown in Figure 1b. While Pearson and Spearman are inherently faster due to their reliance on simple statistics, CCC-GPU's runtime is remarkably close to these methods, showcasing its efficiency while maintaining its advanced capabilities for capturing complex relationships.

The significant performance enhancement provided by CCC-GPU expands the scope of transcriptomic analyses and allowed us to uncover more nonlinear patterns. In the UpSet analysis [8] shown in Figure 1c, we compared how Pearson, Spearman, and CCC-GPU agreed or disagreed in prioritizing gene pairs across all genes in the 54 GTEx tissues. We found that ~2.9 billion gene pairs found only by CCC-GPU (Disagreements in Figure 1c, where CCC-GPU is "high" and any of the others "low") likely have biologically meaningful nonlinear patterns, as it was found in the original CCC study [1]. Likewise, gene pairs where Pearson is "high" and the rest are "low" are likely driven by outliers [1].

Leveraging CCC-GPU's mixed data type capability, we correlated gene expression with GTEx metadata to interpret nonlinear patterns. For this, we included categorical variables such as sex, mortality status, and numerical variables such as BMI, age, among others. From the five intersection groups where CCC values were high but Pearson or Spearman remained low, we selected the top 100 gene pairs per tissue with the largest CCC value. Figure 1d highlights gene pairs with biologically interpretable nonlinear patterns explained by a metadata variable, such as *UTY-RAD18* (sex) and *RASSF2-CYTIP* (pre/post-mortem status; these two genes were previously found to be differentially expressed in these conditions [9]). We provide the CCC values computed for all gene pairs, and the top gene-metadata correlation results across all GTEx v8 tissues (see Supplementary Note 1).

Conclusions

We present CCC-GPU, a GPU-accelerated implementation of the original Clustermatch Correlation Coefficient (CCC). Our work demonstrates that CCC-GPU delivers a remarkable acceleration over its CPU-based predecessor, enabling the rapid and efficient computation of correlation coefficients in large transcriptomic datasets. This performance leap transforms analyses that previously required weeks into tasks achievable within hours on standard research hardware.

Challenges often reside not only in detecting but also in interpreting complex, nonlinear patterns between genes. CCC-GPU's ability to correlate different data types allows researchers to easily incorporate available metadata into their analyses. For example, a previously highlighted nonlinear relationship for the gene pair *RASSF2-CYTIP*, detected by CCC, was explained by GTEx metadata field "COHORT" (pre- vs. post-mortem status). These two genes were previously reported to be differentially expressed under these conditions [9]. CCC also detected a strong linear pattern between these genes regardless of the organism's mortality status.

Our new CCC-GPU delivers a next-generation correlation coefficient at a fraction of the computational cost without sacrificing its accessibility, accuracy, or reliability. This significant performance enhancement makes comprehensive correlation analysis of large genomic data practical on standard research hardware. Beyond standard correlation analyses, CCC-GPU empowers biologists to perform sophisticated tasks such as advanced feature selection prior to machine learning model training with large datasets. Moreover, by accelerating the discovery of novel nonlinear relationships in expression data, CCC-GPU helps researchers quickly identify patterns beyond conventional, linear-only relationships. This can potentially uncover new biological insights that would otherwise remain hidden and drive forward our understanding of complex biological systems.

Acknowledgements

This work is supported by the National Human Genome Research Institute (R00 HG011898 to M.P.), and The Eunice Kennedy Shriver National Institute of Child Health and Human Development (R01 HD109765 to M.P.).

Conflict of Interest

K.F. has recently held less than $3,000 worth of NVIDIA stock. All other authors declare no competing interests.

References

1. **An efficient, not-only-linear correlation coefficient based on clustering**
Milton Pividori, Marylyn D Ritchie, Diego H Milone, Casey S Greene
Cell Systems (2024-09) https://doi.org/nhs4
DOI: 10.1016/j.cels.2024.08.005 · PMID: 39243756 · PMCID: PMC11951854

2. **Detecting Novel Associations in Large Data Sets**
David N Reshef, Yakir A Reshef, Hilary K Finucane, Sharon R Grossman, Gilean McVean, Peter J Turnbaugh, Eric S Lander, Michael Mitzenmacher, Pardis C Sabeti
Science (2011-12-16) https://doi.org/bzn5c3
DOI: 10.1126/science.1205438 · PMID: 22174245 · PMCID: PMC3325791

3. **Measuring Dependence Powerfully and Equitably**
Yakir Reshef, David Reshef, Hilary Finucane, Pardis Sabeti, Michael Mitzenmacher
Journal of Machine Learning Research (2016) https://jmlr.org/papers/v17/15-308.html

4. **The GTEx Consortium atlas of genetic regulatory effects across human tissues**
, François Aguet, Shankara Anand, Kristin G Ardlie, Stacey Gabriel, Gad A Getz, Aaron Graubert, Kane Hadley, Robert E Handsaker, Katherine H Huang, ... Simona Volpi
Science (2020-09-11) https://doi.org/ghbnhr
DOI: 10.1126/science.aaz1776 · PMID: 32913098 · PMCID: PMC7737656

5. **Comparing partitions**
Lawrence Hubert, Phipps Arabie
Journal of Classification (1985-12) https://doi.org/bphmzh
DOI: 10.1007/bf01908075

6. **Lessons learned from comparing C-CUDA and Python-Numba for GPU-Computing**
Lena Oden

2020 28th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP) (2020-03) https://doi.org/g9m4km
DOI: 10.1109/pdp50117.2020.00041

7. **pybind11 — Seamless operability between C++11 and Python**
Wenzel Jakob, Jason Rhinelander, Dean Moldovan
GitHub (2017) https://github.com/pybind/pybind11

8. **UpSet: Visualization of Intersecting Sets**
Alexander Lex, Nils Gehlenborg, Hendrik Strobelt, Romain Vuillemot, Hanspeter Pfister
IEEE Transactions on Visualization and Computer Graphics (2014-12-31) https://doi.org/f3ssr5
DOI: 10.1109/tvcg.2014.2346248 · PMID: 26356912

9. **The effects of death and post-mortem cold ischemia on human tissue transcriptomes**
Pedro G Ferreira, Manuel Muñoz-Aguirre, Ferran Reverter, Caio P Sá Godinho, Abel Sousa, Alicia Amadoz, Reza Sodaei, Marta R Hidalgo, Dmitri Pervouchine, Jose Carbonell-Caballero, … Roderic Guigó
Nature Communications (2018-02-13) https://doi.org/gc2xd8
DOI: 10.1038/s41467-017-02772-x · PMID: 29440659 · PMCID: PMC5811508

Supplementary Information

Supplementary Note 1: Datasets

In our Zenodo archive (https://doi.org/10.5281/zenodo.17156519), we provide:

1. the CCC values for all gene pairs of GTEx v8 whole blood (other tissues are omitted since the total file size is too large, but they can be easily computed using the code in the GitHub repository),
2. the threshold tables for top and bottom genes using both the 30% threshold and the permutation-based thresholds (see section below for an explanation of how these thresholds were computed), and
3. the top gene-metadata correlation results for all tissues.

Supplementary Note 2: High and Low Correlation Thresholds

We used two approaches to define correlation values that are "high" or "low" for each coefficient per tissue. The first approach captures, for each tissue and correlation coefficient, the top 30% of gene pairs by using the 70th percentile of correlation values, and the bottom 30% of gene pairs by using the 30th percentile (Figure 1c, in the main text, for all tissues combined; and panel c for individual tissues in the supplementary figures section below). The second approach computes the null distribution of coefficient values per tissue by taking a random subset of 10,000 genes and shuffling samples. Then, we define genes with "high" correlation as those with coefficient values larger than the 95th percentile ($P < 0.05$), and "low" correlation as those with coefficient values smaller than the 80th percentile ($P > 0.20$) (Figure S5 for all tissues, and panel d for individual tissues, in the supplementary figures section below).

Definition

The *Clustermatch Correlation Coefficient (CCC)* between a feature pair (i.e., gene pair in the context of transcriptomics), represented by data vectors $\mathbf{x}$ and $\mathbf{y}$, is defined as the maximum ARI between all possible object partitions (i.e., grouping of objects, which are groups of RNA-seq samples in the context of transcriptomics) derived from the data vectors:

$$\mathrm{CCC}(\mathbf{x}, \mathbf{y}) = \max\{0, \max_{\substack{\pi_j \in \Pi^{\mathbf{x}} \\ \pi_l \in \Pi^{\mathbf{y}}}} \{\mathrm{ARI}(\pi_j, \pi_l)\}\}, \forall\, |\pi| \in [2, k_{\max}] \quad \textbf{(E1)}$$

where $\Pi^{\mathbf{x}}$ is a set of partitions derived from $\mathbf{x}$, $\Pi^{\mathbf{y}}$ is a set of partitions derived from $\mathbf{y}$, and $k_{\max}$ specifies the maximum number of clusters allowed for partitions. The ARI has an upper bound of 1 (achieved when both partitions are identical), and although it does not have a well-defined lower bound, values equal or less than zero are achieved when partitions are independent. Therefore, $\mathrm{CCC}(\mathbf{x}, \mathbf{y}) \in [0, 1]$. In the special case where all n objects in either $\mathbf{x}$ or $\mathbf{y}$ have the same value, the CCC is undefined. Refer to the original CCC article [1] for extended definitions, explanation of $k_{\max}$, properties, statistical significance, among other details.

Pseudocode

Algorithm 1: CCC algorithm

1 **Function** `get_partitions`($\mathbf{v}$, $k_{\max}$):
  Input:
    $\mathbf{v}$: feature values on n objects
    $k_{\max}$: maximum number of clusters
  Output:
    Π: a set of partitions over n objects
2 **if** $\mathbf{v} \in \mathbb{R}^n$ **then**
3 **for** $k \leftarrow 2$ **to** $\min\{k_{\max}, n-1\}$ **do**
4 $\boldsymbol{\rho} \leftarrow (\rho_\ell \mid \Pr(v_i < \rho_\ell) \leq (\ell-1)/k), \forall \ell \in [1, k+1]$
5 $\pi_\ell \leftarrow \{i \mid \rho_\ell < v_i \leq \rho_{\ell+1}\}, \forall \ell \in [1, k]$
6 $\Pi_k \leftarrow \pi$
7 **else**
8 $\mathcal{C} \leftarrow \{c_1, c_2, \ldots, c_m\}$ (set of m unique categorical values in $\mathbf{v}$)
9 $\pi_\ell \leftarrow \{i \mid v_i = \mathcal{C}_\ell\}, \forall \ell \in [1, m]$
10 $\Pi_m \leftarrow \pi$
11 $\Pi \leftarrow \{\Pi_k \mid |\Pi_k| > 1\}, \forall k$
12 **return** Π
13
14 **Function** `ccc`($\mathbf{x}$, $\mathbf{y}$, $k_{\max}$):
  Input:
    $\mathbf{x}$: feature values on n objects
    $\mathbf{y}$: feature values on n objects
    $k_{\max}$: maximum number of clusters
  Output:
    c: correlation value for $\mathbf{x}$ and $\mathbf{y}$ ($c \in [0, 1]$)
15 $\Pi^{\mathbf{x}} = \text{get\_partitions}(\mathbf{x}, k_{\max})$
16 $\Pi^{\mathbf{y}} = \text{get\_partitions}(\mathbf{y}, k_{\max})$
17 $c \leftarrow \max\{\text{ARI}(\pi_j, \pi_l)\}, \forall \pi_j \in \Pi^{\mathbf{x}}, \pi_l \in \Pi^{\mathbf{y}}$
18 **return** $\max(c, 0)$

Figure S1: The CCC algorithm.

The main function of the algorithm, `ccc`, generates a set of partitions $\Pi^{\mathbf{x}}$ for variable $\mathbf{x}$ (line 15), and another set of partitions $\Pi^{\mathbf{y}}$ for variable $\mathbf{y}$ (line 16). Then, it computes the ARI between each partition $\pi_j \in \Pi^{\mathbf{x}}$ and $\pi_l \in \Pi^{\mathbf{y}}$ and gets the maximum (line 17), returning either this value or zero if this is negative (line 18). Refer to the original CCC article [1] for more details.

Supplementary Note 3: CCC Runtime Profiling

To quantify the computational bottleneck in the CCC algorithm, we performed comprehensive profiling using Python's cProfile on representative workloads. We categorized all function calls into: ARI (Adjusted Rand Index computation), Partitioning (quantile-based clustering), Coordination (algorithm orchestration), and NumPy/Numba (numerical operations).

Figure S2 shows the runtime breakdown by individual function for a medium workload (2,500 features × 500 samples). The three ARI-related functions (`adjusted_rand_index`, `get_pair_confusion_matrix`, and `get_contingency_matrix`) collectively consume approximately 80% of total runtime, confirming that ARI computation is the dominant bottleneck.

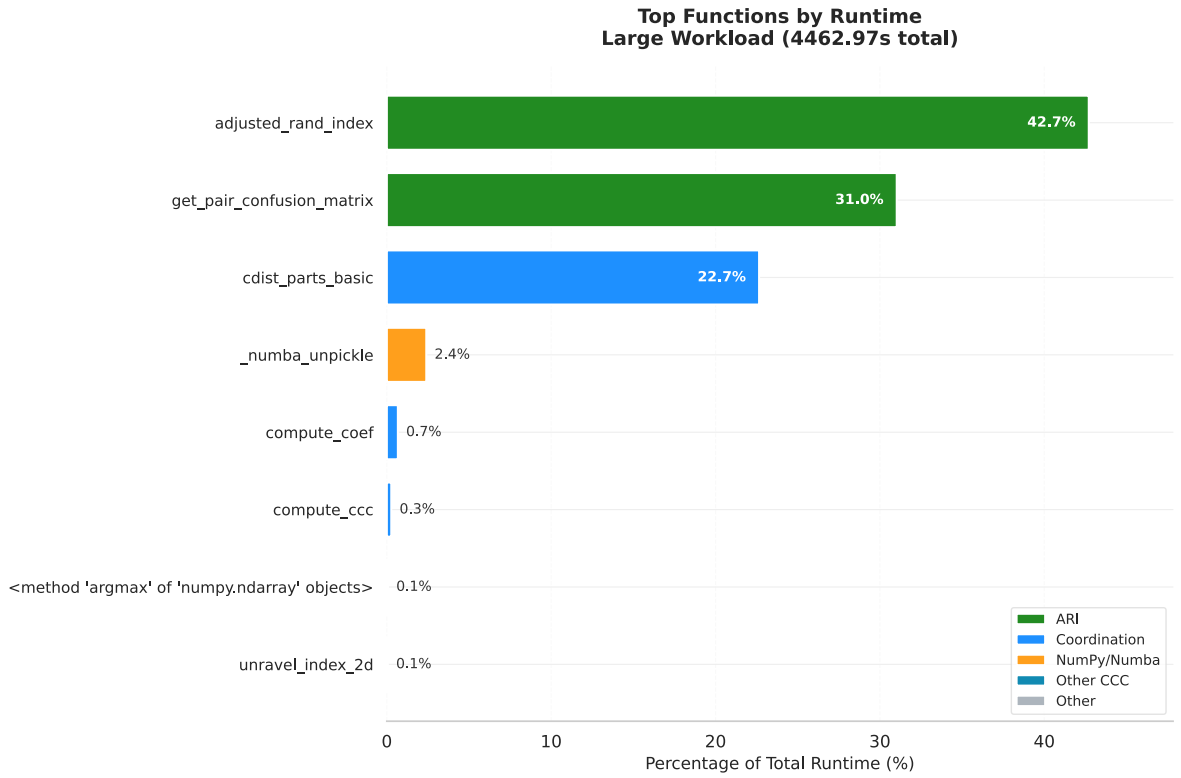


Figure S2: Runtime breakdown by function for CCC computation. Top functions sorted by percentage of total runtime for a 2,500 features × 500 samples workload.

Figure S3 compares runtime distribution across three feature sizes: Small (500 features), Medium (2,500 features), and Large (5,000 features), all with 500 samples. The ARI category consistently dominates across all workload sizes, accounting for ~74% of runtime.

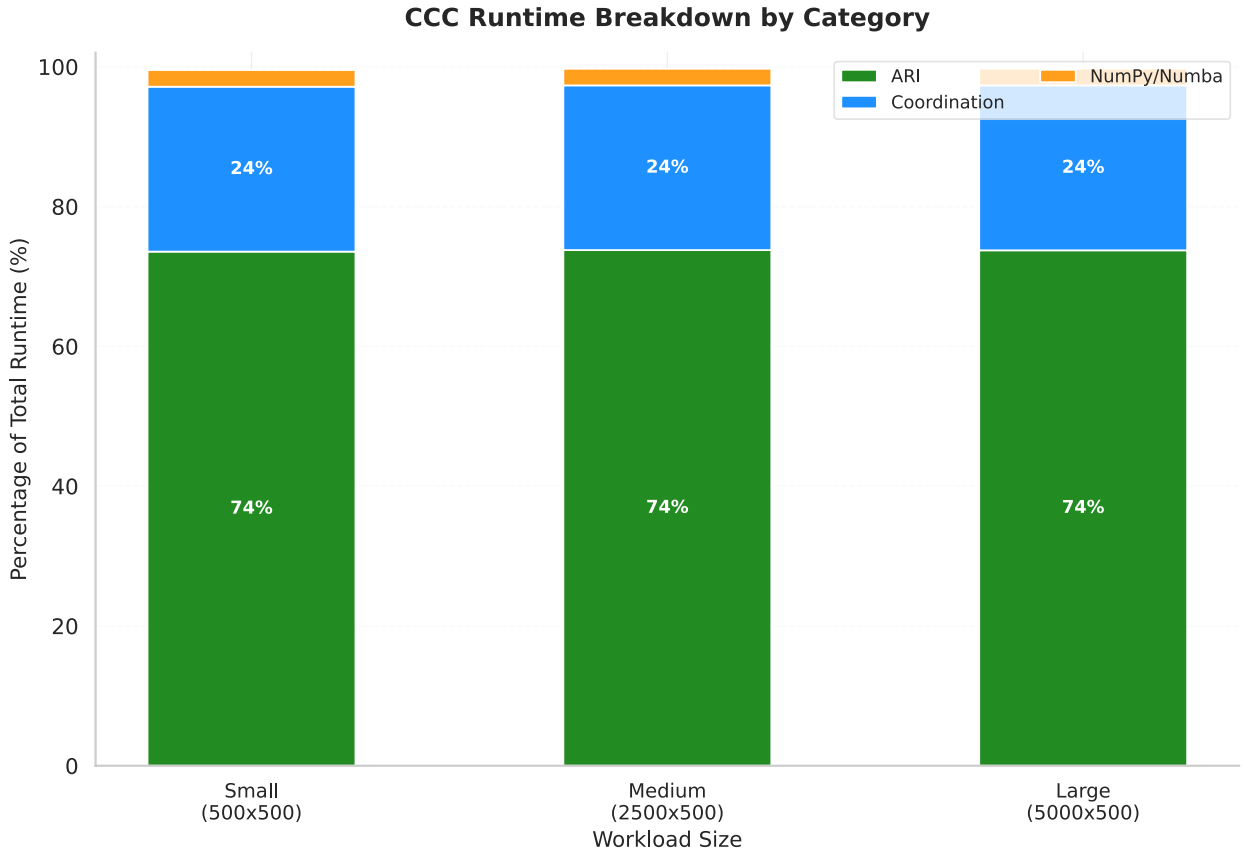


Figure S3: Category breakdown comparison across different feature sizes. Runtime distribution for Small (500 features), Medium (2,500 features), and Large (5,000 features) workloads, each with 500 samples.

Figure S4 shows how ARI's share of total runtime changes as sample count increases (500 to 4,000 samples, fixed 500 features). The ARI percentage increases from approximately 74% at 500 samples to over 91% at 4,000 samples, demonstrating that for larger datasets typical in transcriptomics, ARI computation becomes even more dominant. This analysis confirms that GPU acceleration of ARI computation addresses the correct computational bottleneck.

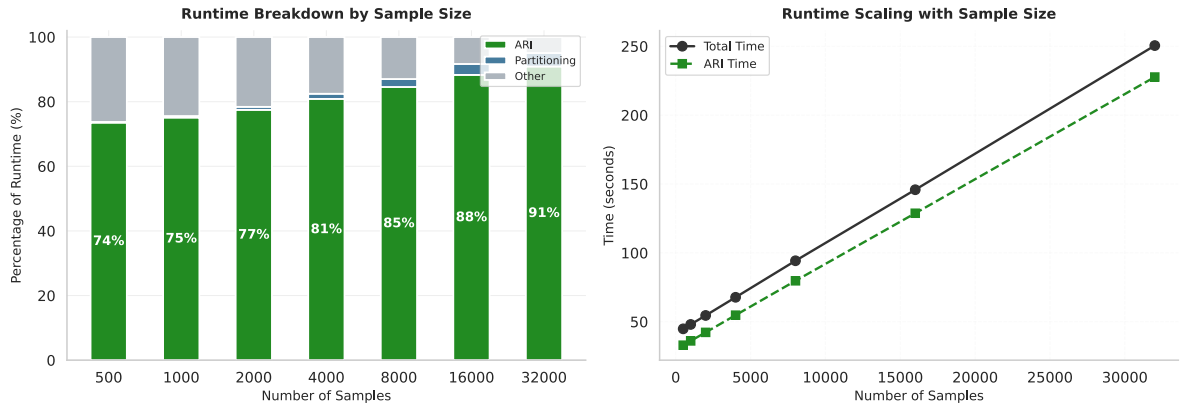


Figure S4: ARI runtime percentage scaling with sample size. Percentage of total CCC runtime spent on ARI computation as sample count increases from 500 to 4,000 (fixed 500 features).

Tables

Table S1: Speedup using different CPU configurations (1,000 fixed samples).

Number of genes	CCC-GPU vs. CCC (6 cores)	CCC-GPU vs. CCC (12 cores)	CCC-GPU vs. CCC (24 cores)
500	17.6x	16.52x	16.1x
1,000	56.17x	30.65x	21.82x
2,000	87.06x	45.72x	24.45x
4,000	116.39x	59.46x	33.03x
6,000	128.74x	67.46x	34.77x
8,000	140.24x	71.48x	38.67x
10,000	138.51x	72.38x	37.02x
16,000	142.7x	73.83x	37.53x
20,000	142.83x	73.88x	37.73x

Table S2: Execution time of different methods (1,000 fixed samples).

Number of genes	CCC-GPU	CCC (12 cores)	Spearman (12 cores)	Pearson (12 cores)
500	0.279s	4.603s	0.051s	0.024s
1,000	0.529s	16.198s	0.089s	0.045s
2,000	1.384s	63.263s	0.235s	0.138s
4,000	4.290s	255.071s	0.679s	0.489s
6,000	8.441s	569.409s	1.379s	1.093s
8,000	14.116s	1009.069s	2.443s	1.973s
10,000	21.694s	1570.269s	4.379s	3.448s

Number of genes	CCC-GPU	CCC (12 cores)	Spearman (12 cores)	Pearson (12 cores)
16,000	55.469s	4051.114s	11.169s	9.310s
20,000	86.286s	6374.781s	20.282s	17.238s

Figures

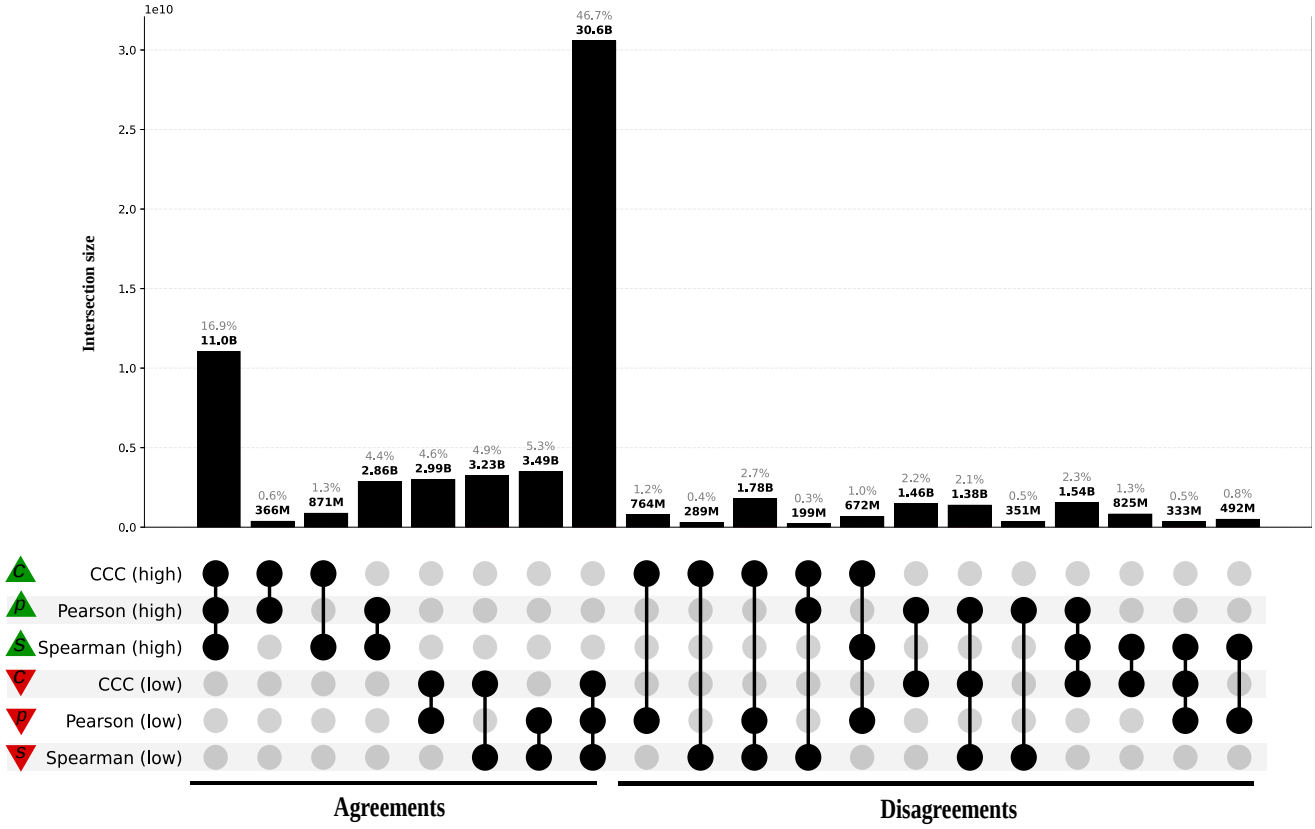
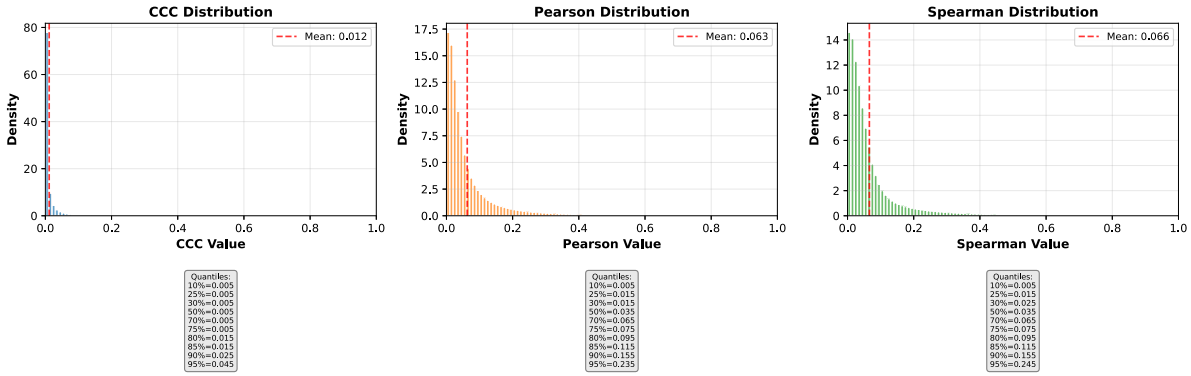


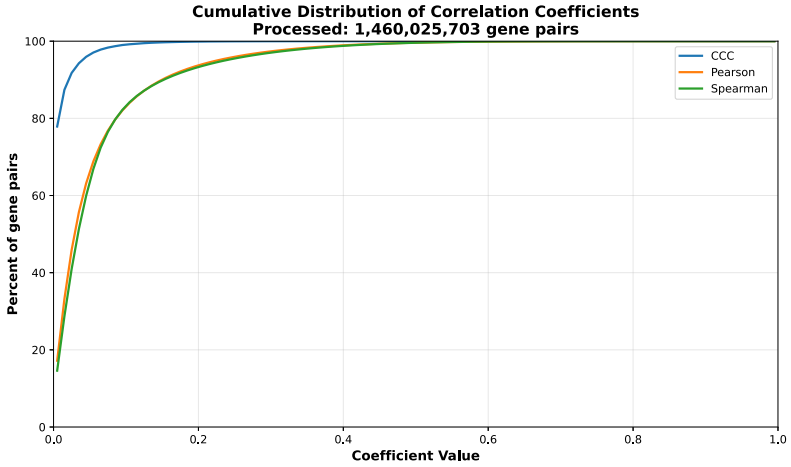
Figure S5: UpSet plot for gene pairs in all 54 tissues in GTEx using permutation-based thresholds for each coefficient for grouping.

Adipose Subcutaneous

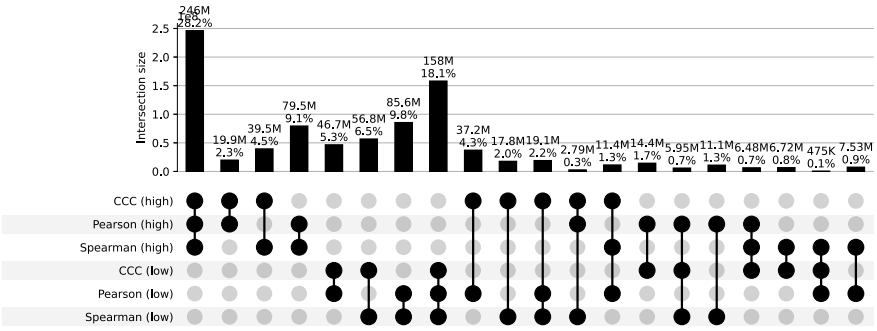
a) Correlation coefficient distributions between gene pairs within GTEx v8 Adipose Subcutaneous



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

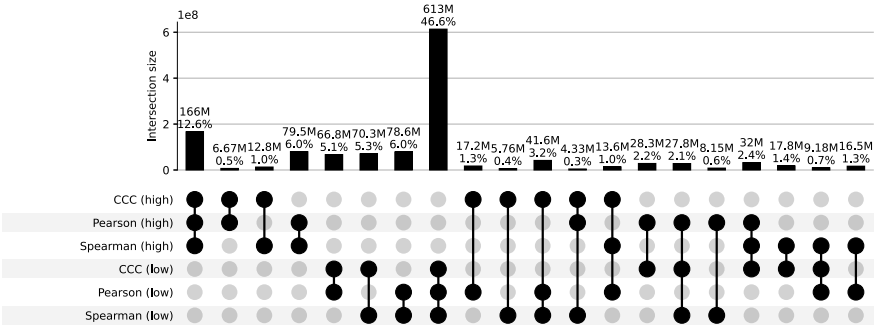
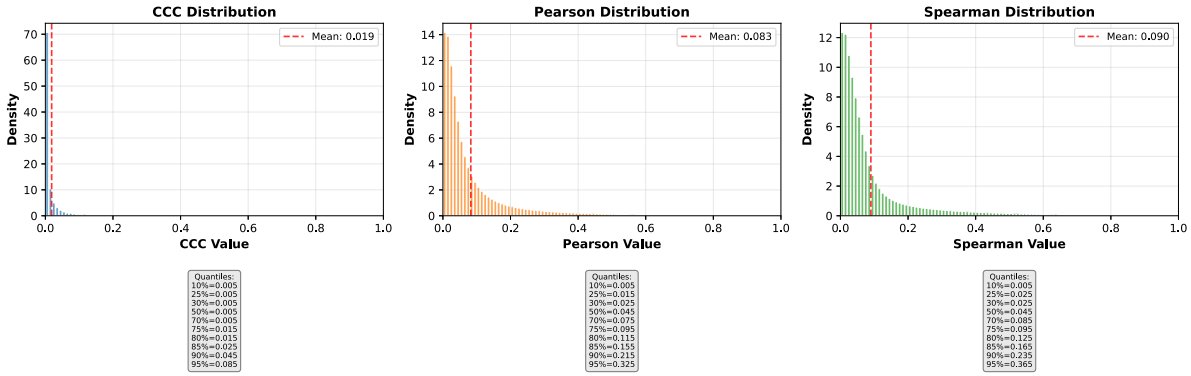


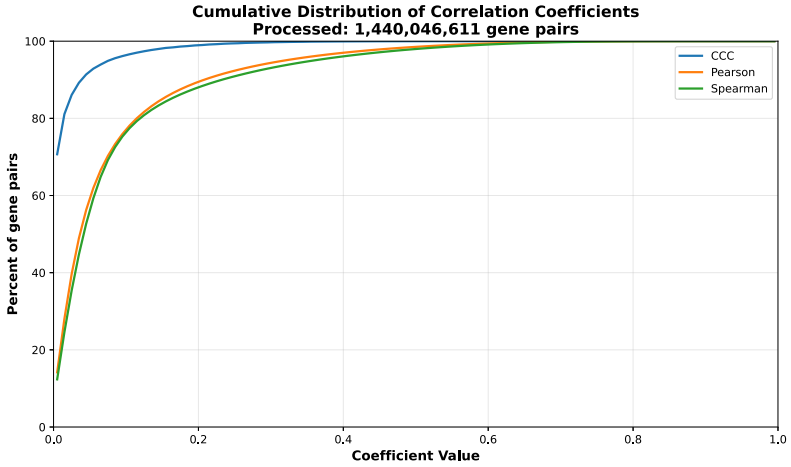
Figure S6: Distribution and UpSet plots for GTEx v8 adipose subcutaneous.

Adipose Visceral Omentum

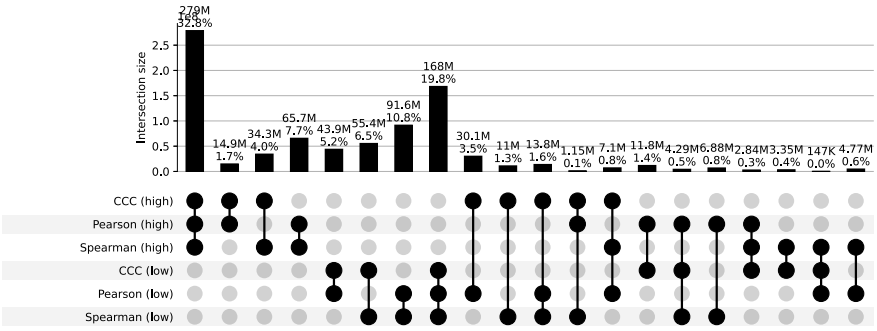
a) Correlation coefficient distributions between gene pairs within GTEx v8 Adipose Visceral Omentum



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

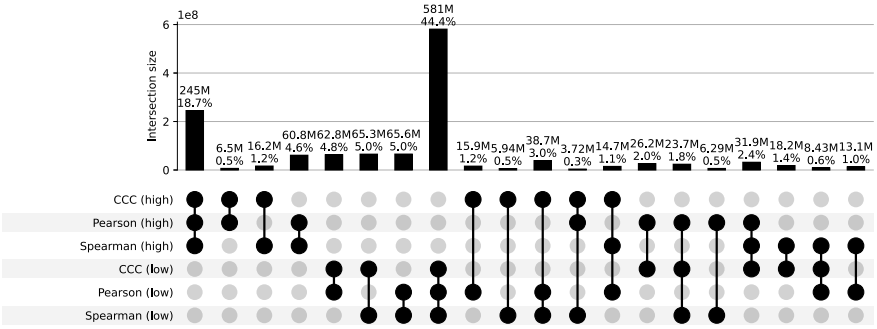
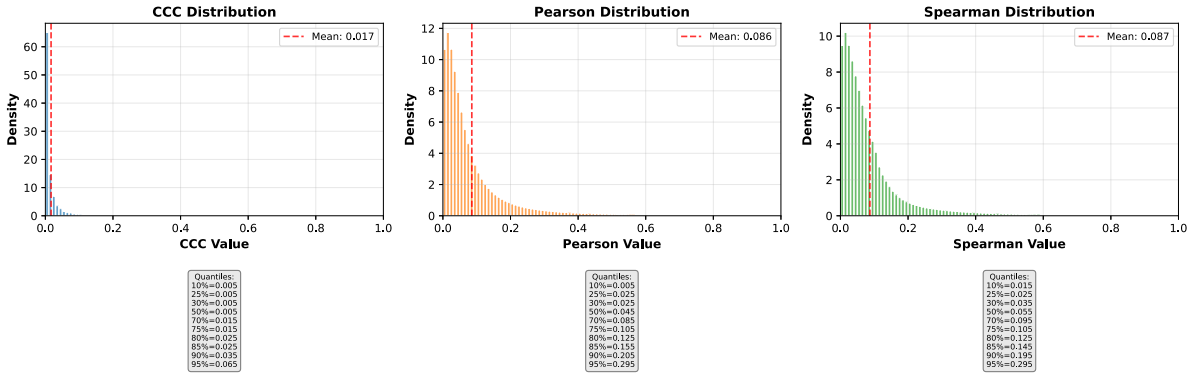


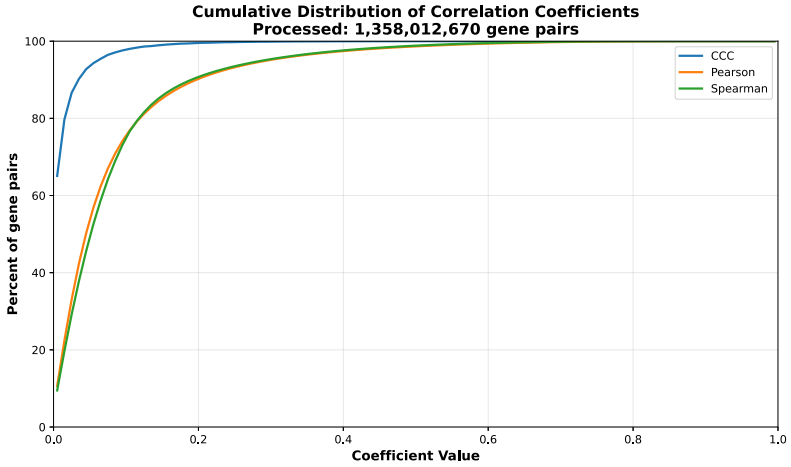
Figure S7: Distribution and UpSet plots for GTEx v8 adipose visceral omentum.

Adrenal Gland

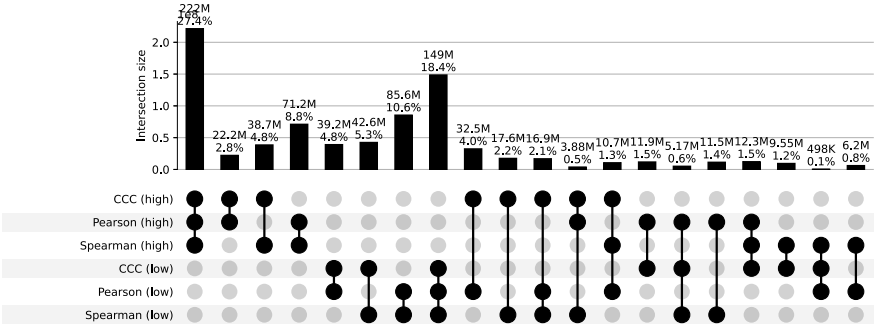
a) Correlation coefficient distributions between gene pairs within GTEx v8 Adrenal Gland



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

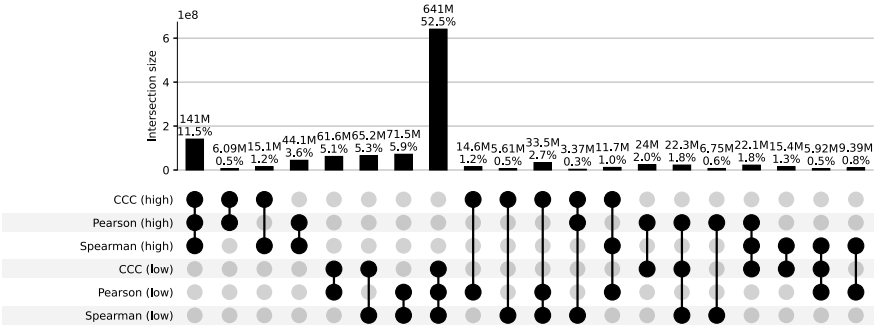
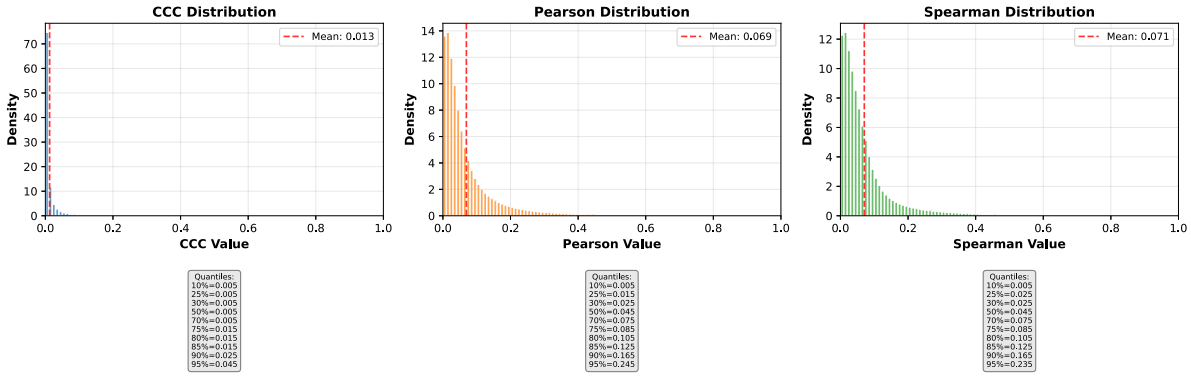


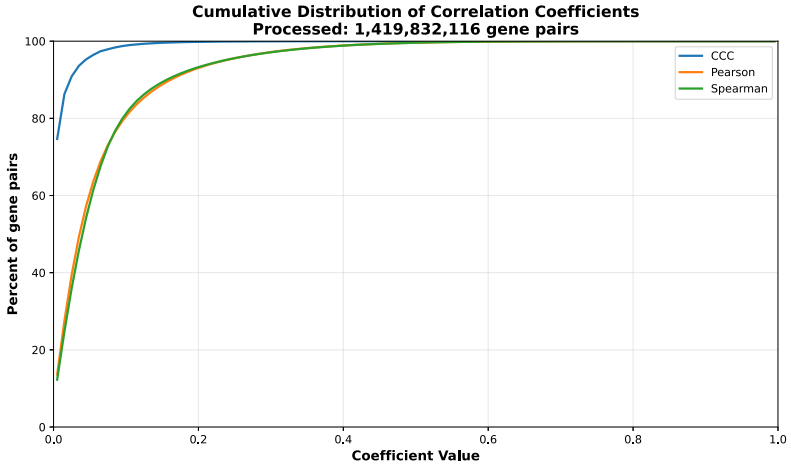
Figure S8: Distribution and UpSet plots for GTEx v8 adrenal gland.

Artery Aorta

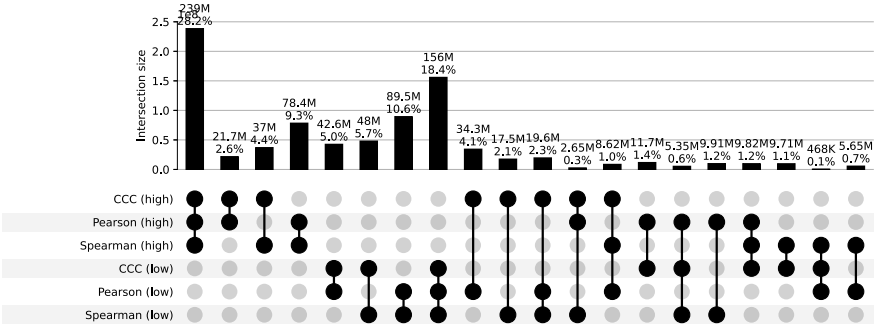
a) Correlation coefficient distributions between gene pairs within GTEx v8 Artery Aorta



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

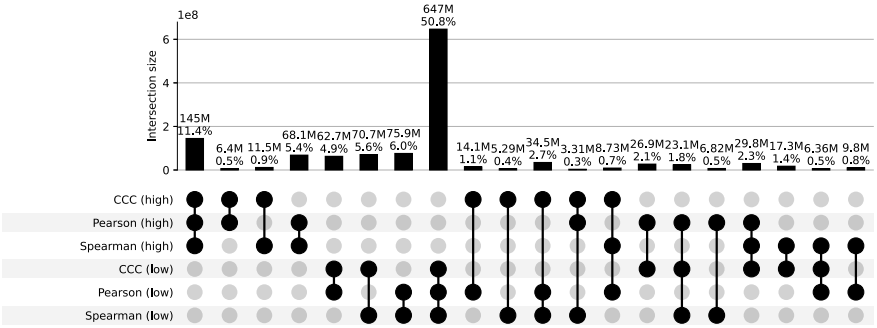
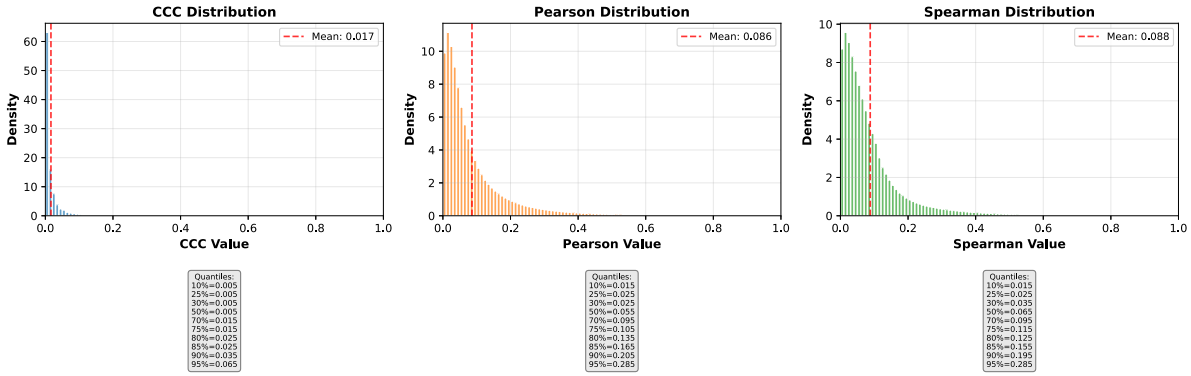


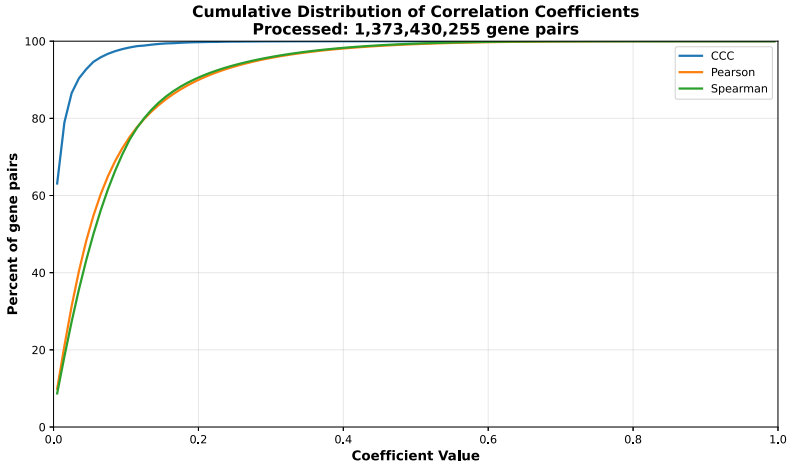
Figure S9: Distribution and UpSet plots for GTEx v8 artery aorta.

Artery Coronary

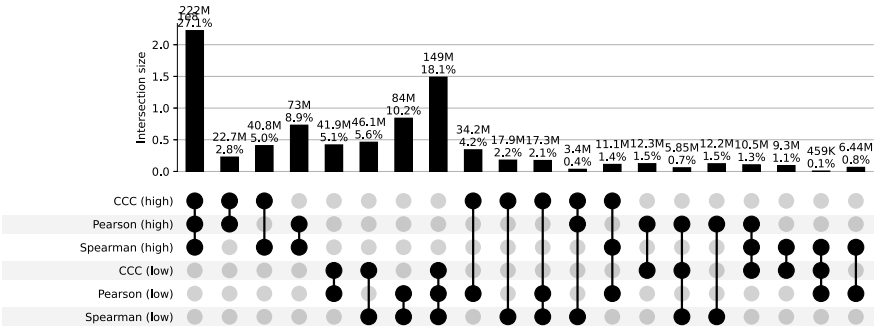
a) Correlation coefficient distributions between gene pairs within GTEx v8 Artery Coronary



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

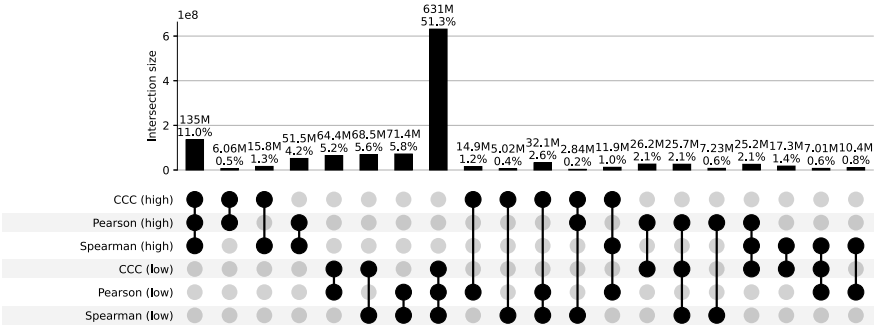
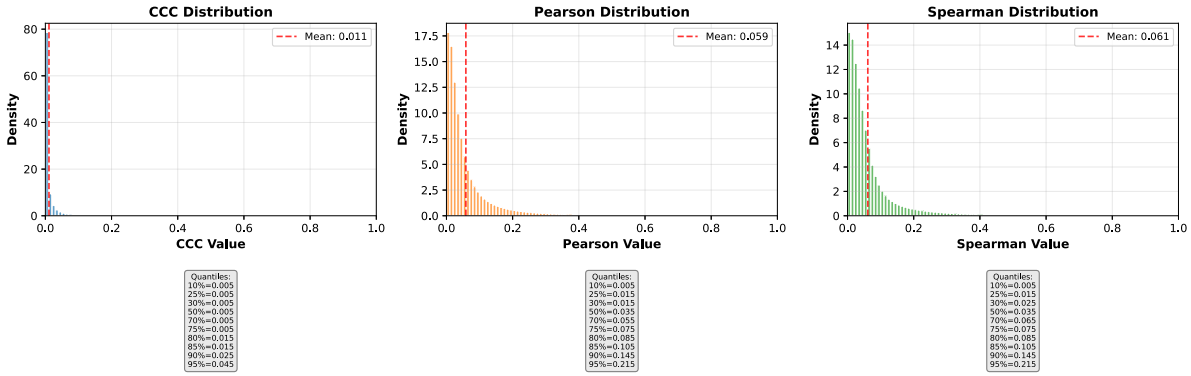


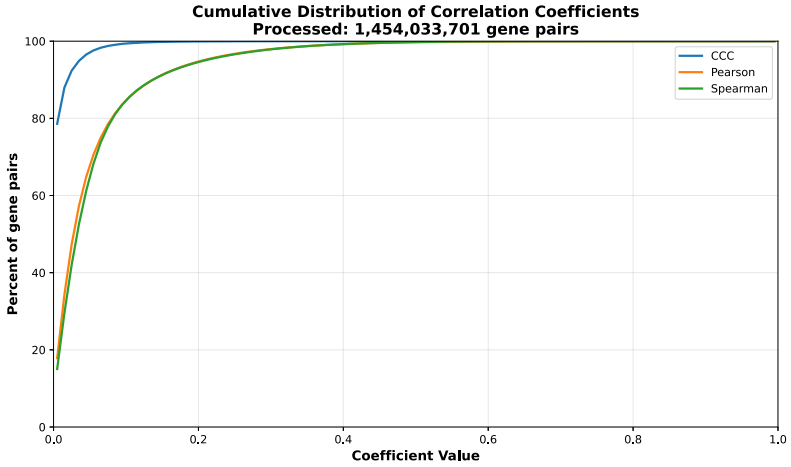
Figure S10: Distribution and UpSet plots for GTEx v8 artery coronary.

Artery Tibial

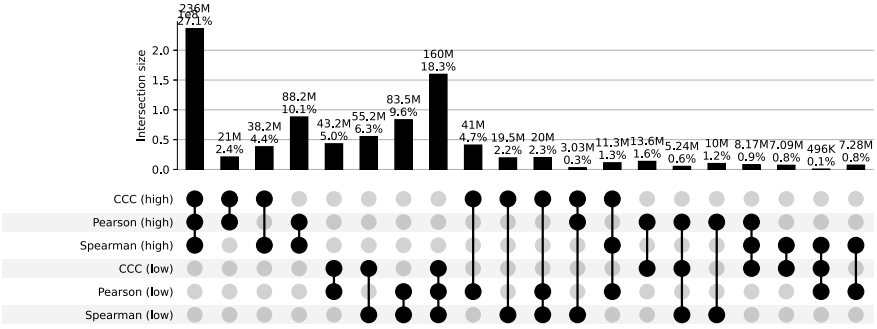
a) Correlation coefficient distributions between gene pairs within GTEx v8 Artery Tibial



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

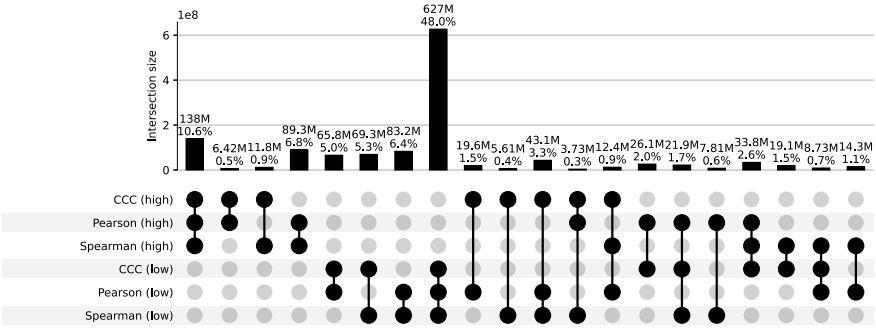
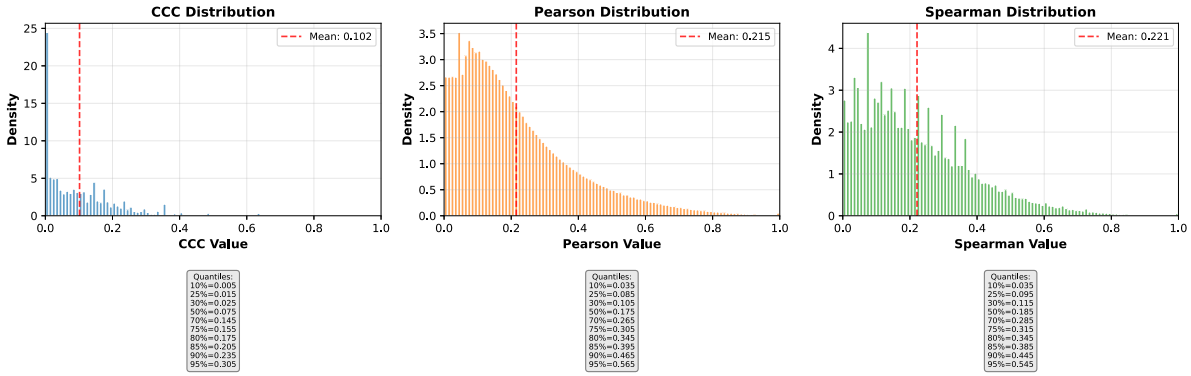


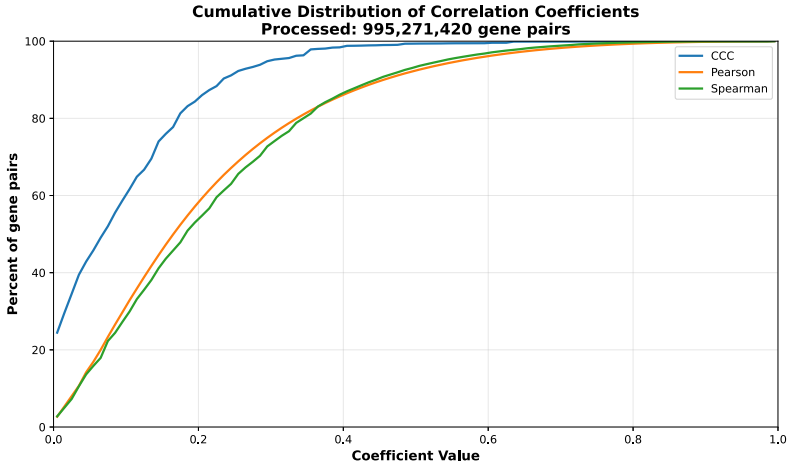
Figure S11: Distribution and UpSet plots for GTEx v8 artery tibial.

Bladder

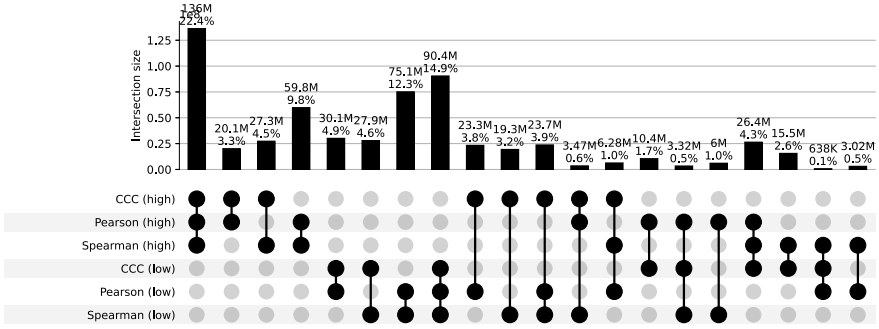
a) Correlation coefficient distributions between gene pairs within GTEx v8 Bladder



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

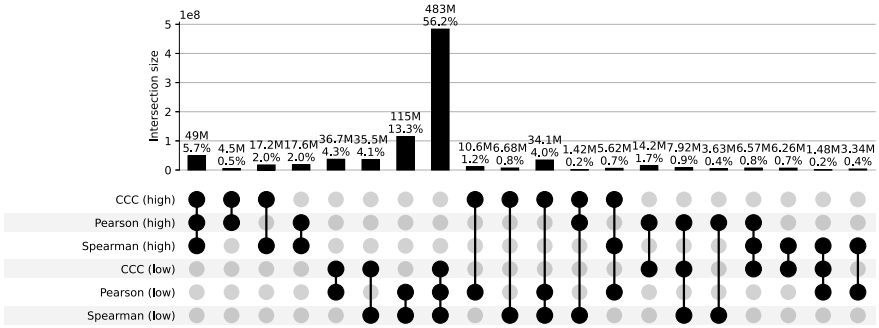
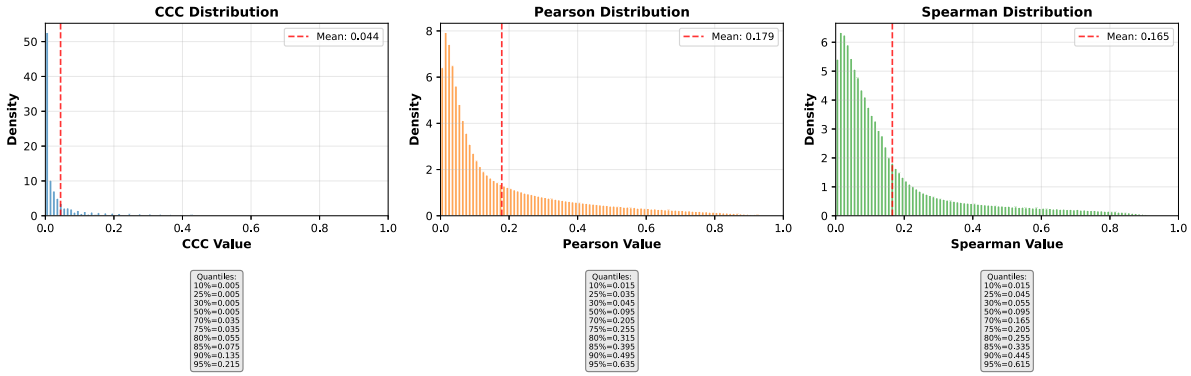


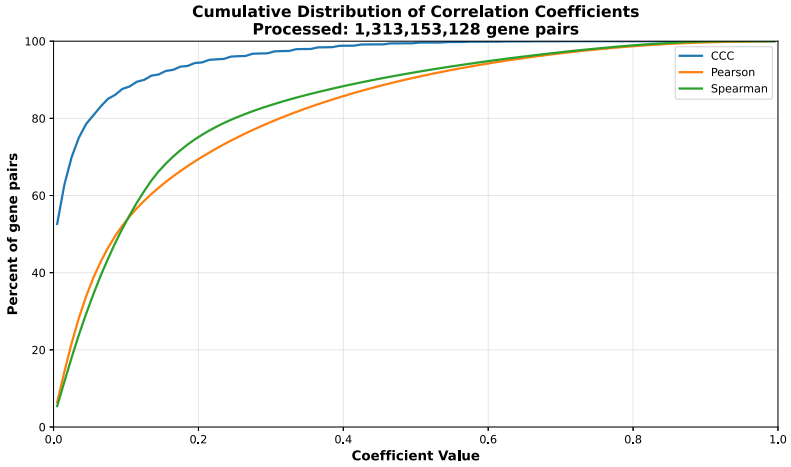
Figure S12: Distribution and UpSet plots for GTEx v8 bladder.

Brain Amygdala

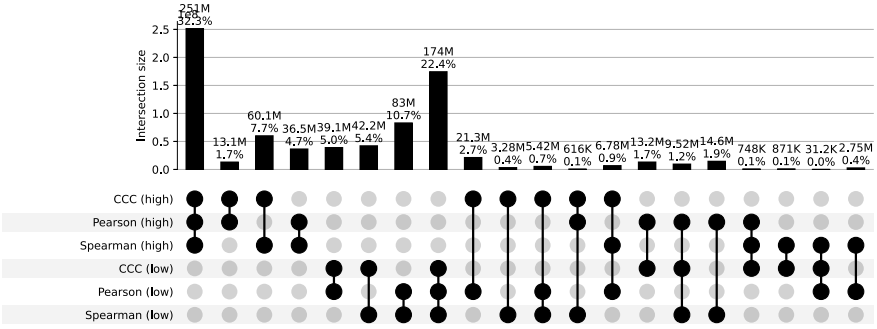
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Amygdala



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

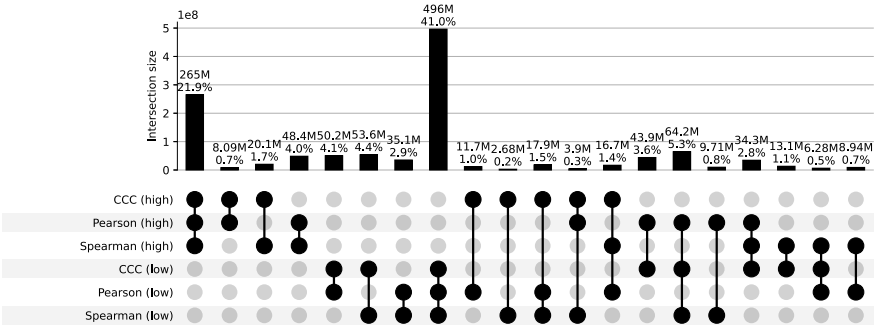
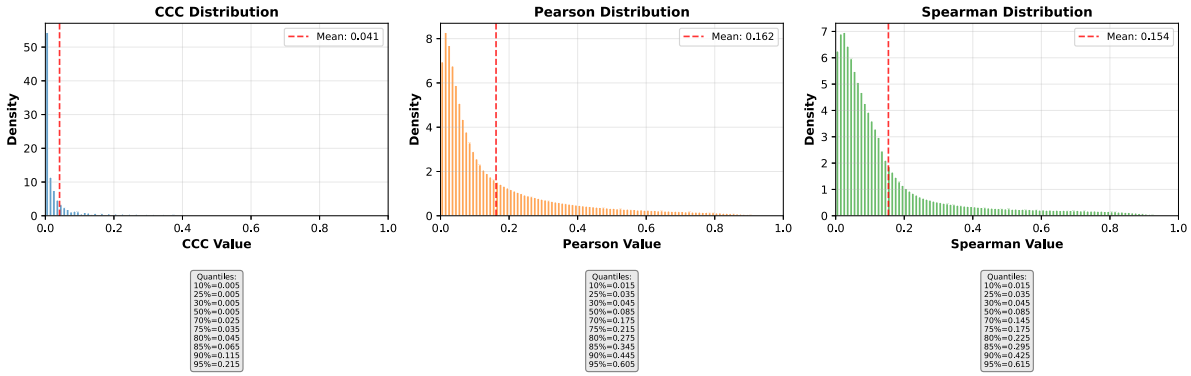


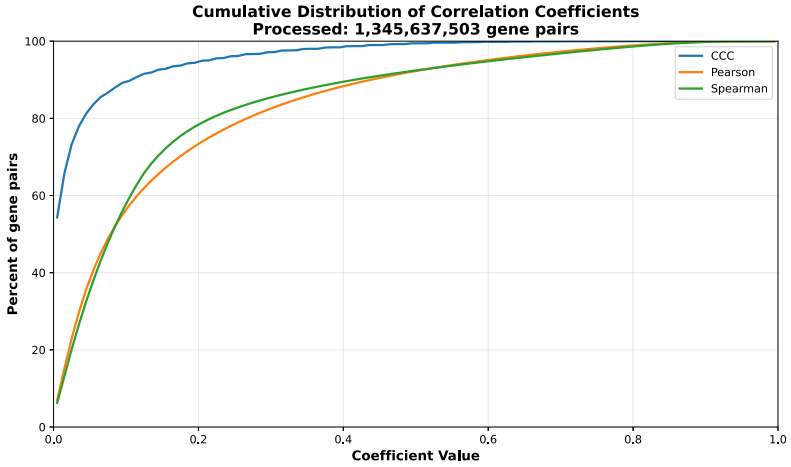
Figure S13: Distribution and UpSet plots for GTEx v8 brain amygdala.

Brain Anterior Cingulate Cortex Ba24

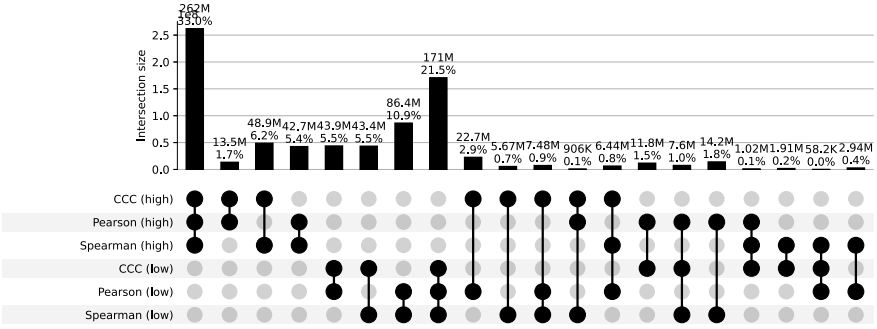
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Anterior Cingulate Cortex Ba24



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

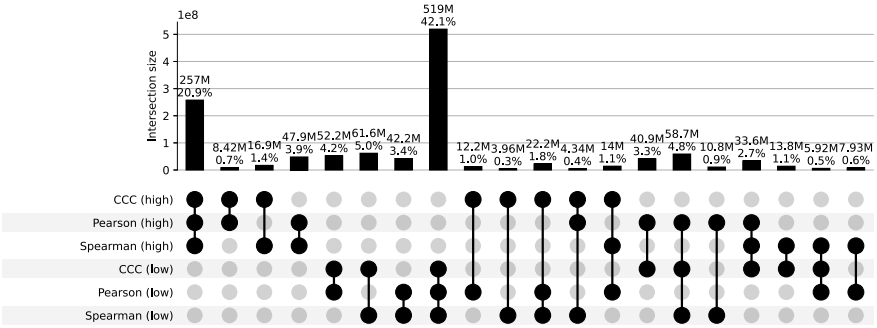
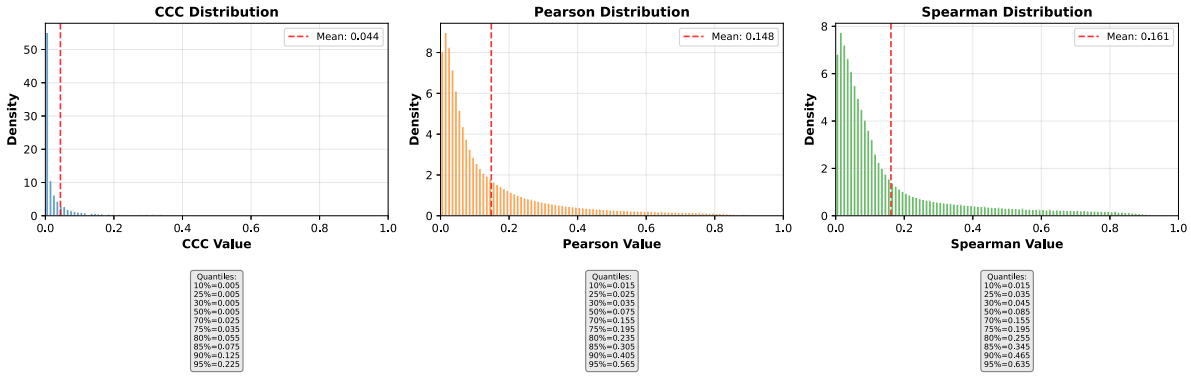


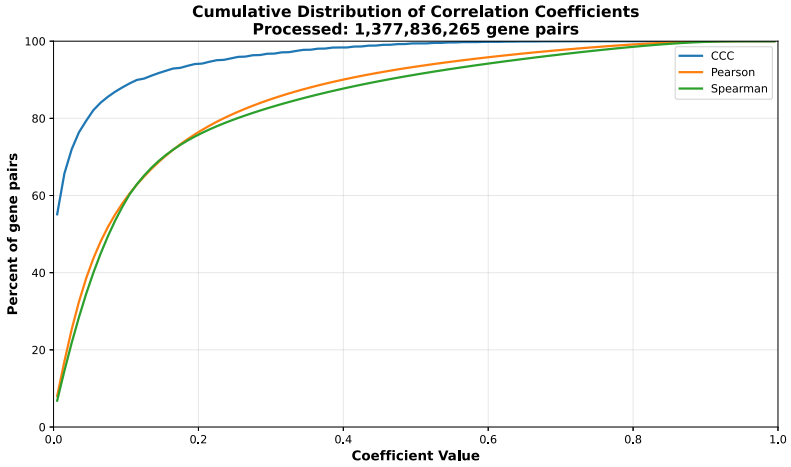
Figure S14: Distribution and UpSet plots for GTEx v8 brain anterior cingulate cortex BA24.

Brain Caudate Basal Ganglia

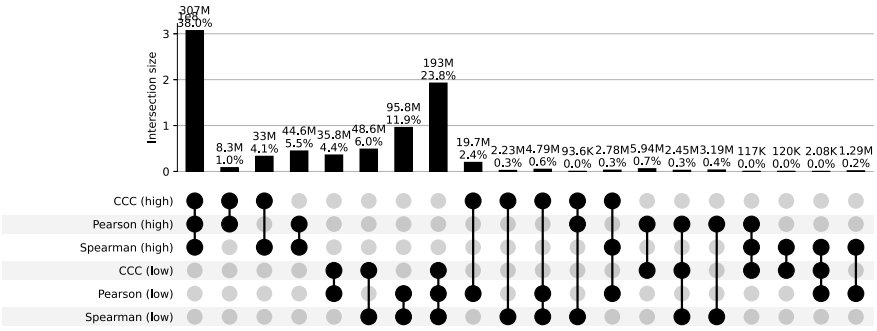
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Caudate Basal Ganglia



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

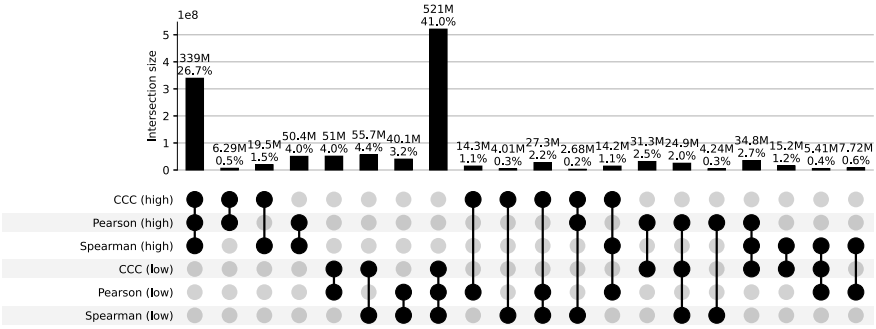
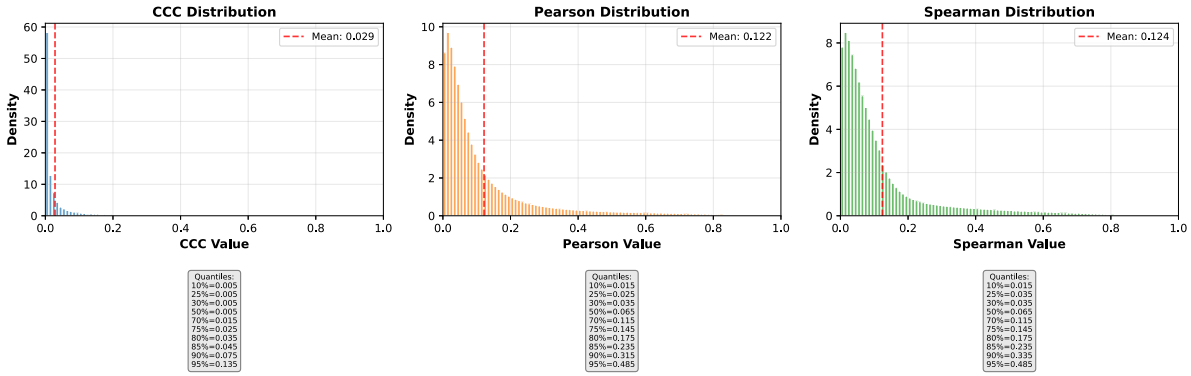


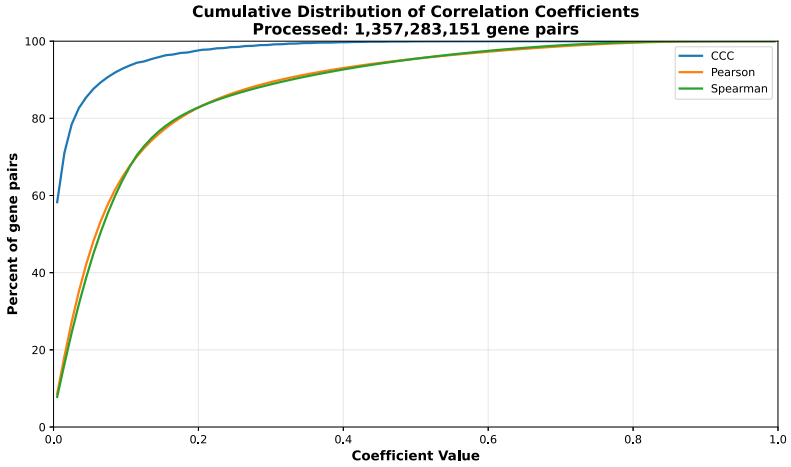
Figure S15: Distribution and UpSet plots for GTEx v8 brain caudate basal ganglia.

Brain Cerebellar Hemisphere

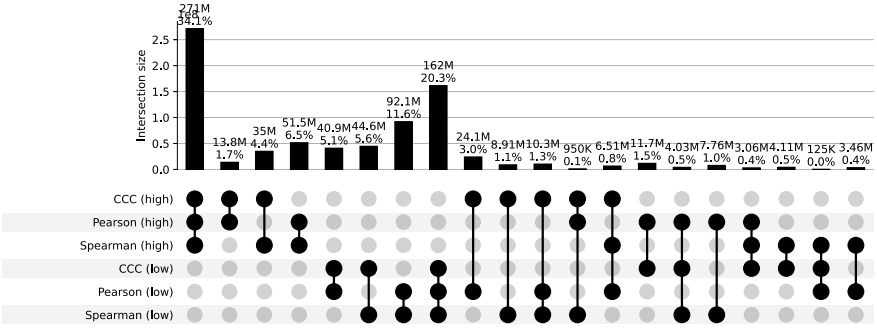
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Cerebellar Hemisphere



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

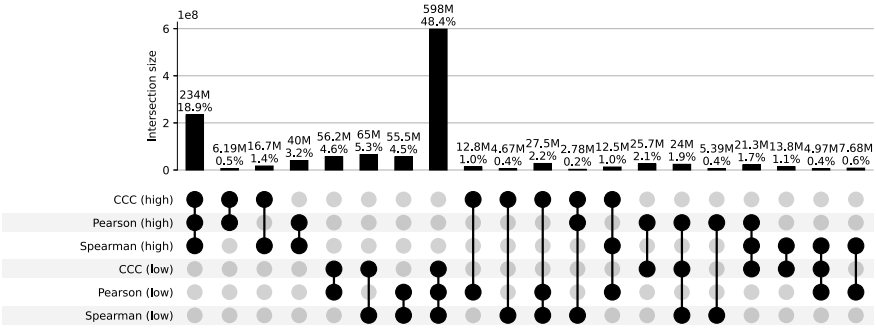
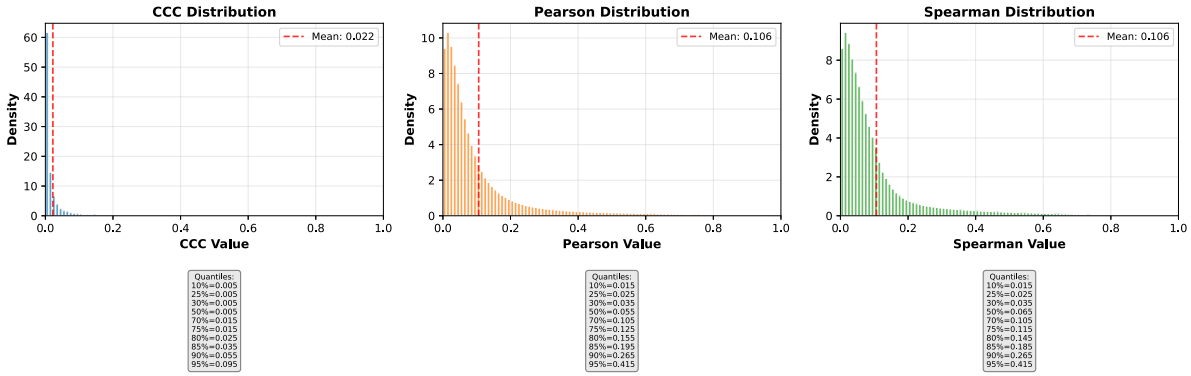


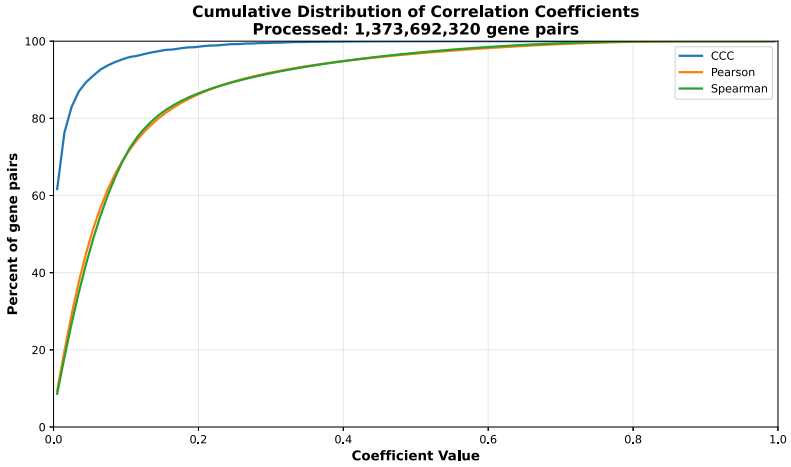
Figure S16: Distribution and UpSet plots for GTEx v8 brain cerebellar hemisphere.

Brain Cerebellum

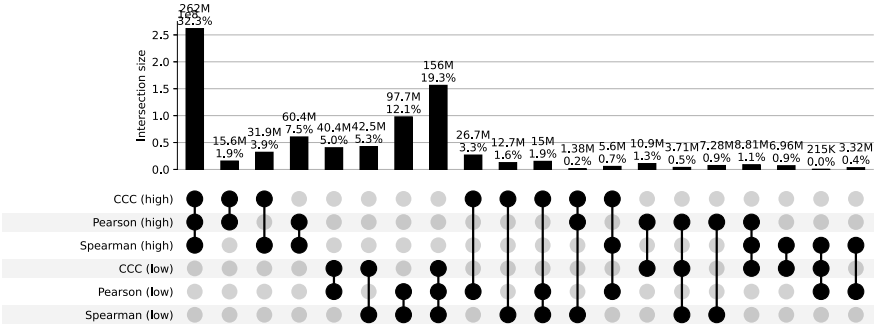
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Cerebellum



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

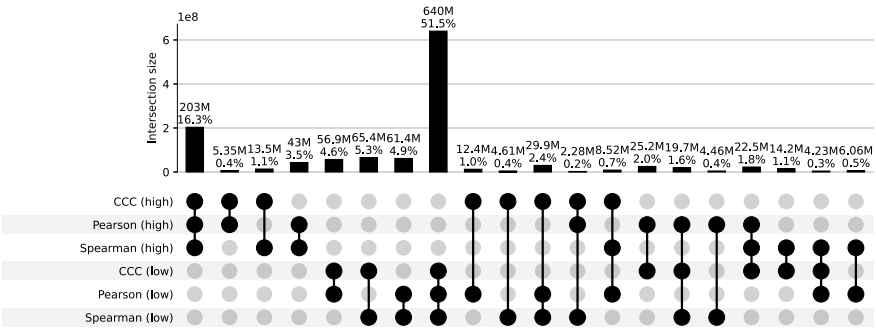
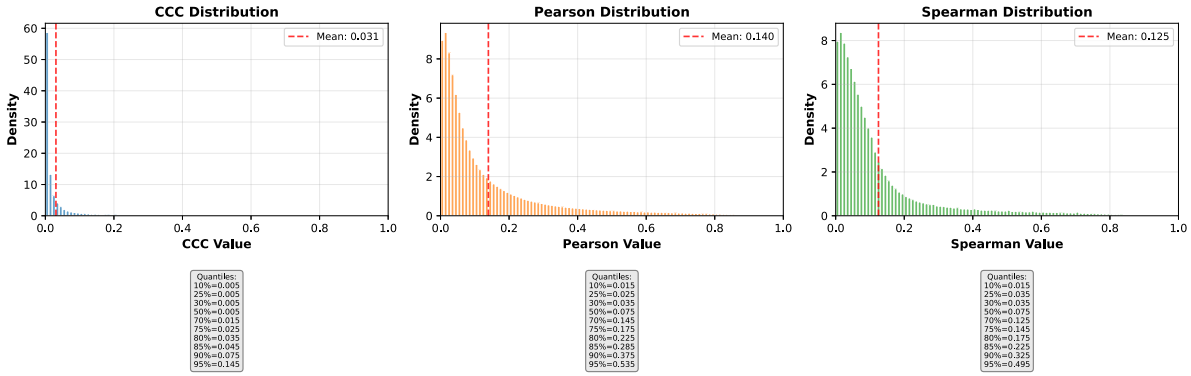


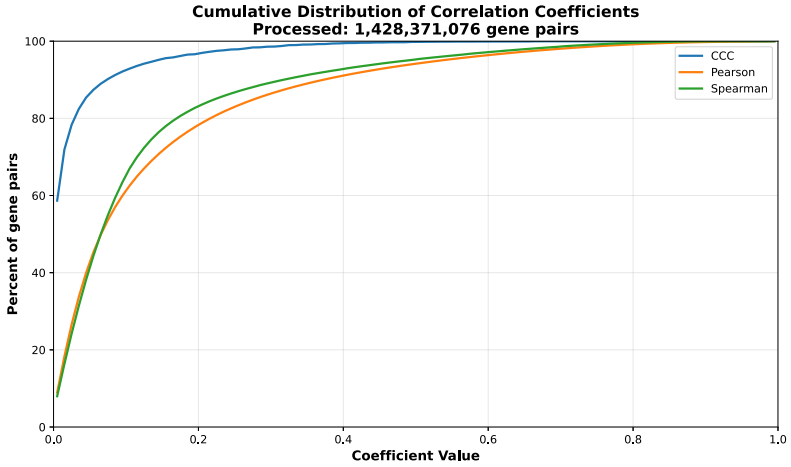
Figure S17: Distribution and UpSet plots for GTEx v8 brain cerebellum.

Brain Cortex

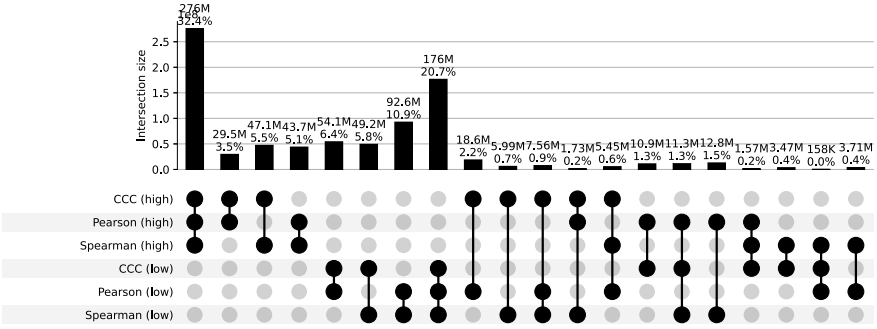
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Cortex



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

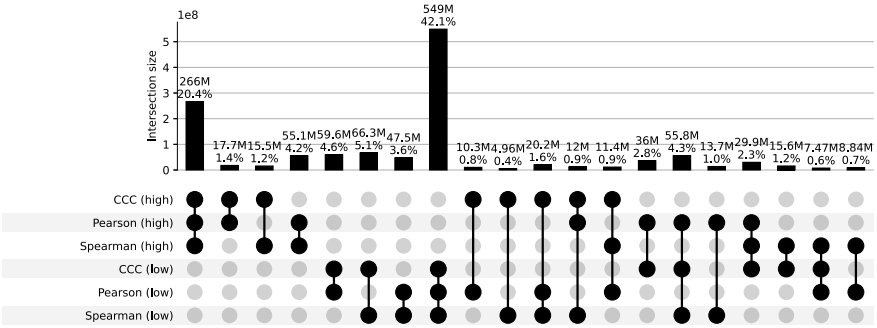
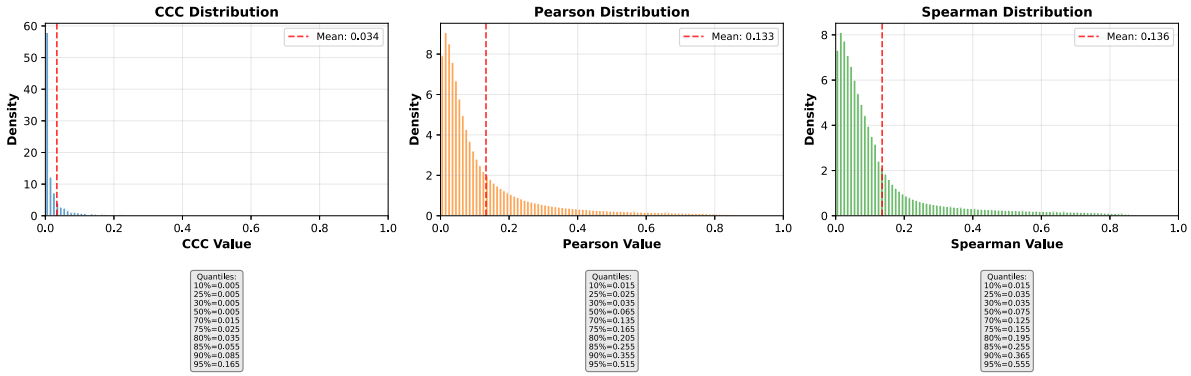


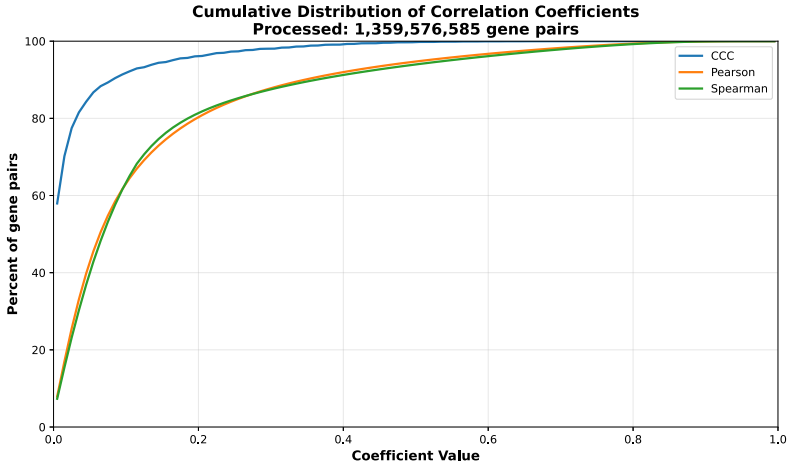
Figure S18: Distribution and UpSet plots for GTEx v8 brain cortex.

Brain Frontal Cortex Ba9

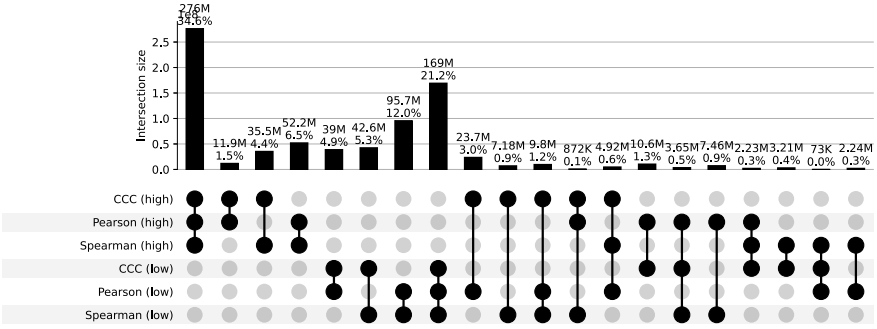
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Frontal Cortex Ba9



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

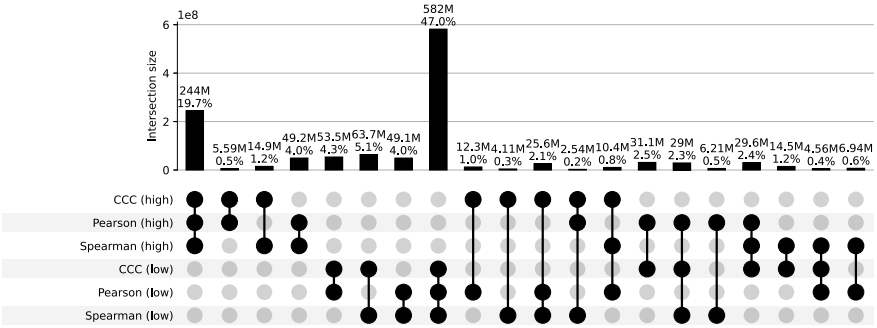
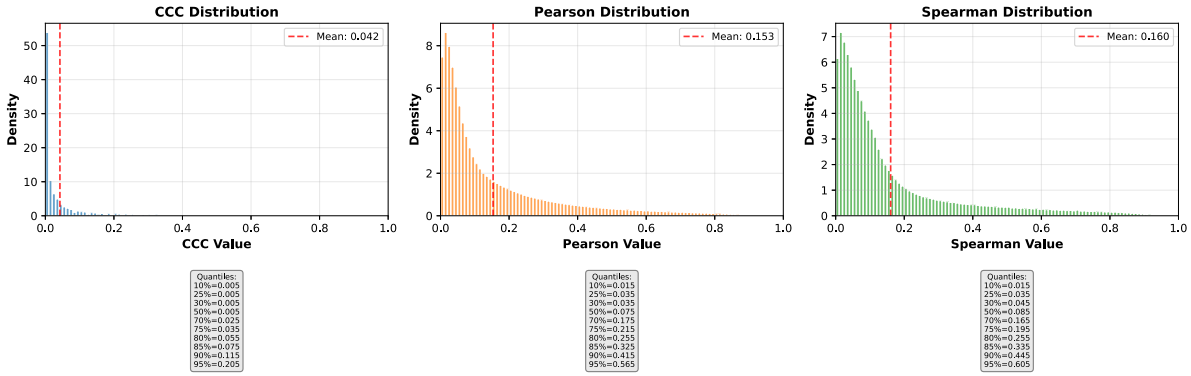


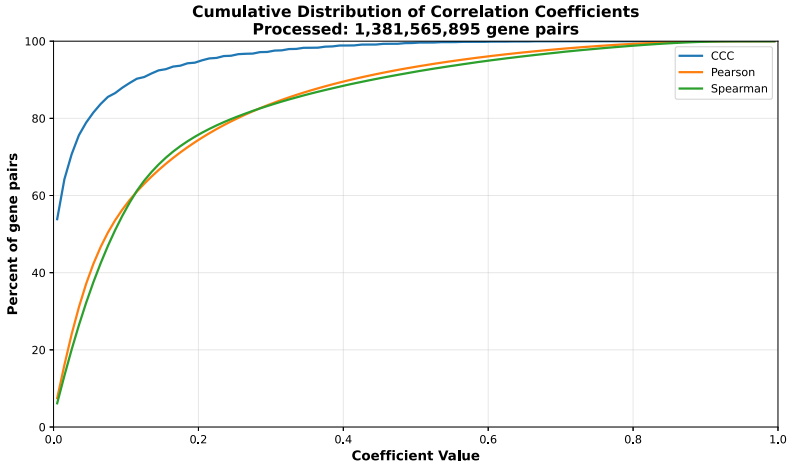
Figure S19: Distribution and UpSet plots for GTEx v8 brain frontal cortex BA9.

Brain Hippocampus

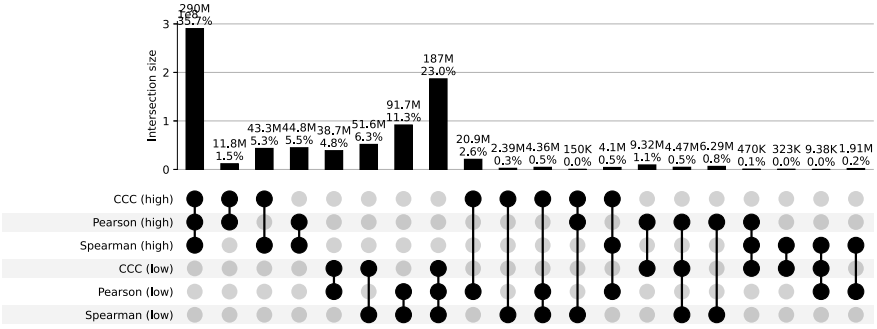
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Hippocampus



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

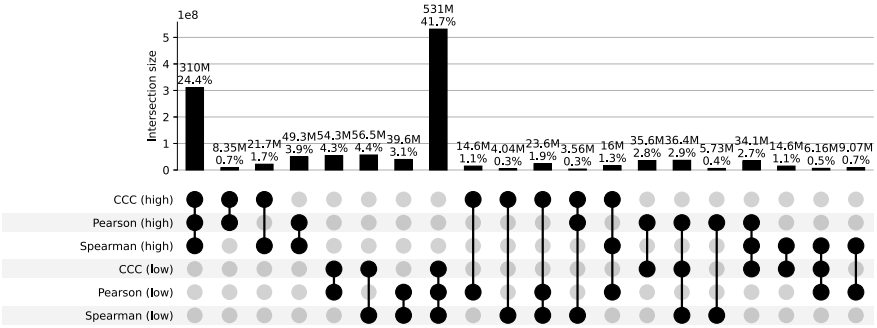
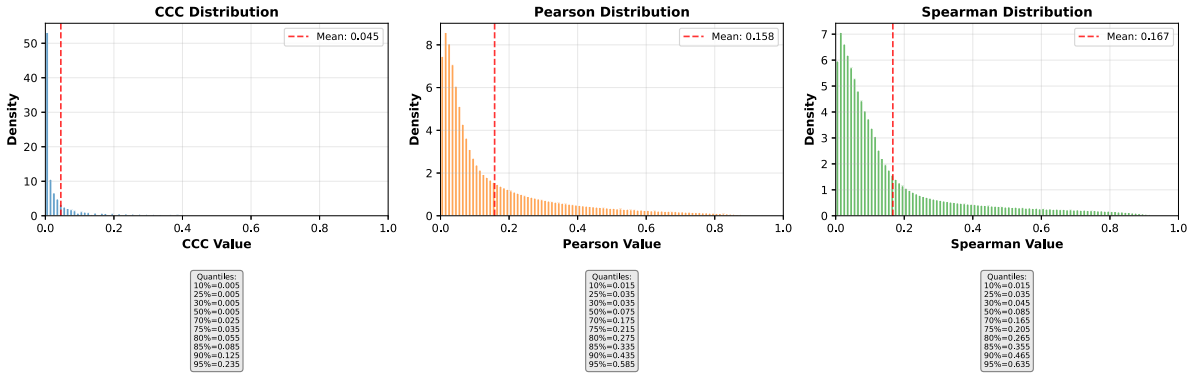


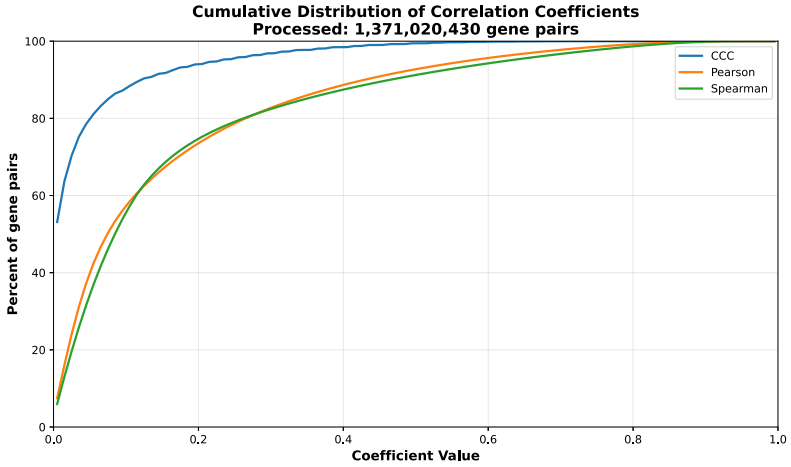
Figure S20: Distribution and UpSet plots for GTEx v8 brain hippocampus.

Brain Hypothalamus

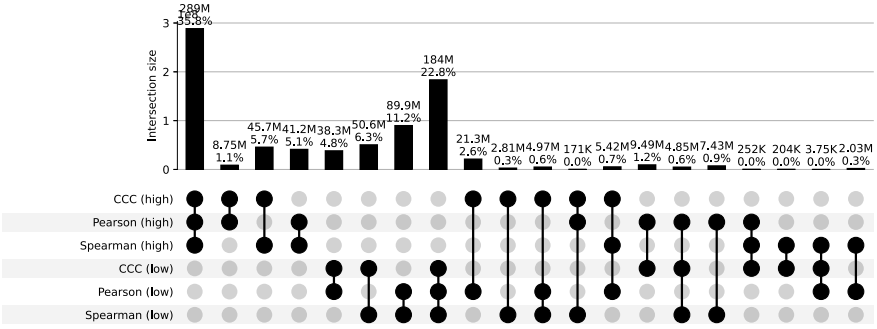
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Hypothalamus



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

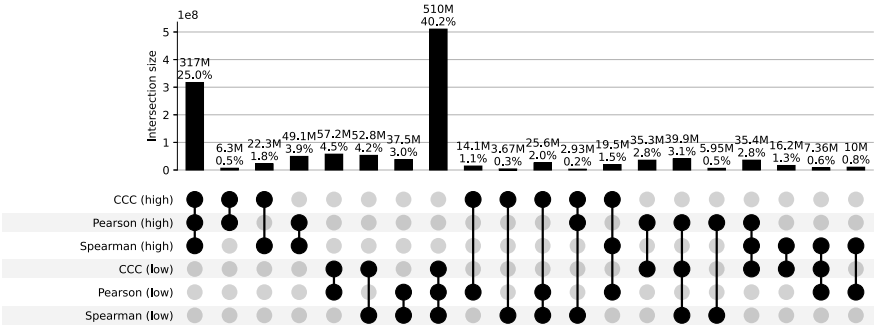
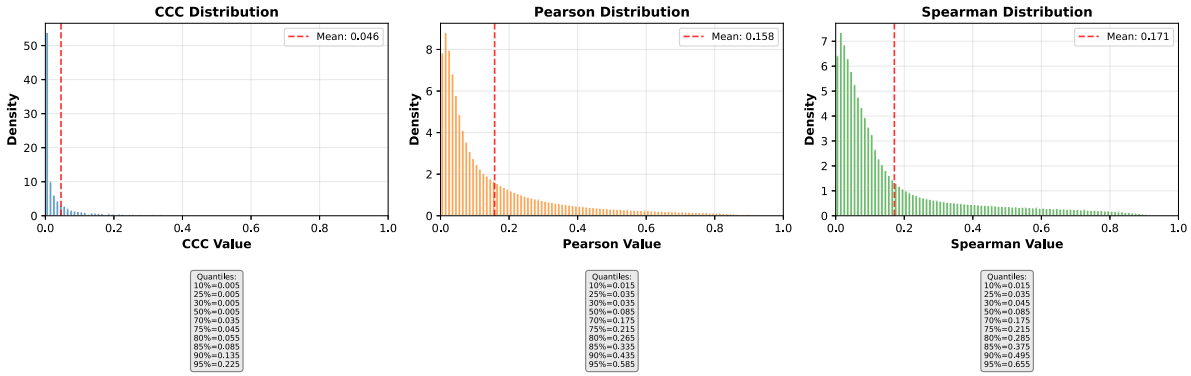


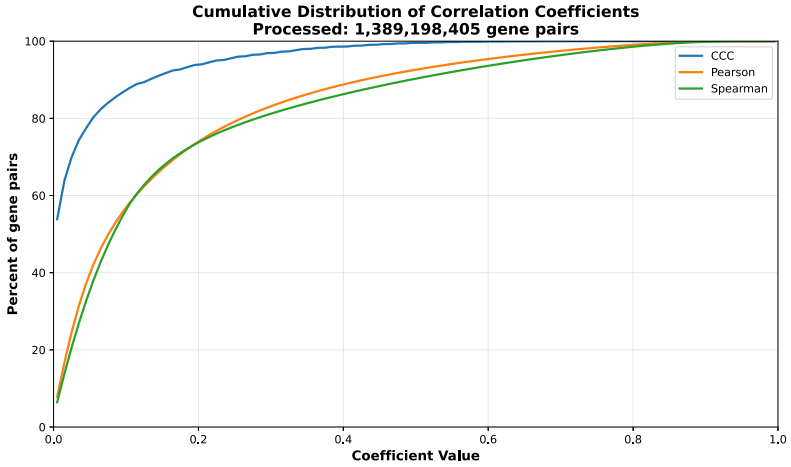
Figure S21: Distribution and UpSet plots for GTEx v8 brain hypothalamus.

Brain Nucleus Accumbens Basal Ganglia

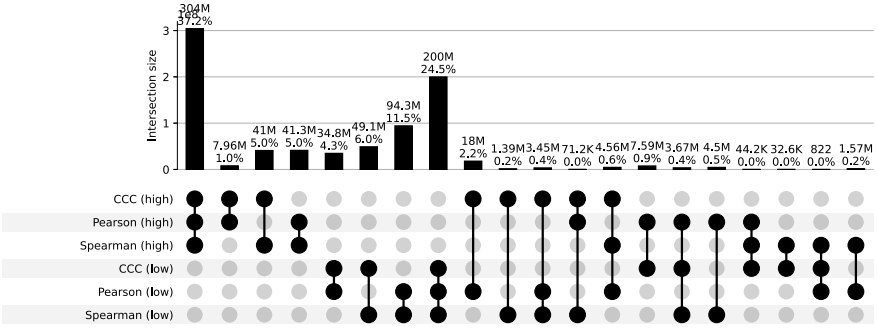
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Nucleus Accumbens Basal Ganglia



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

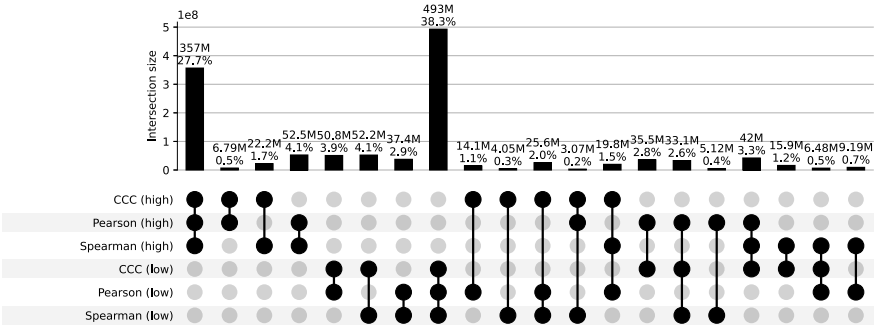
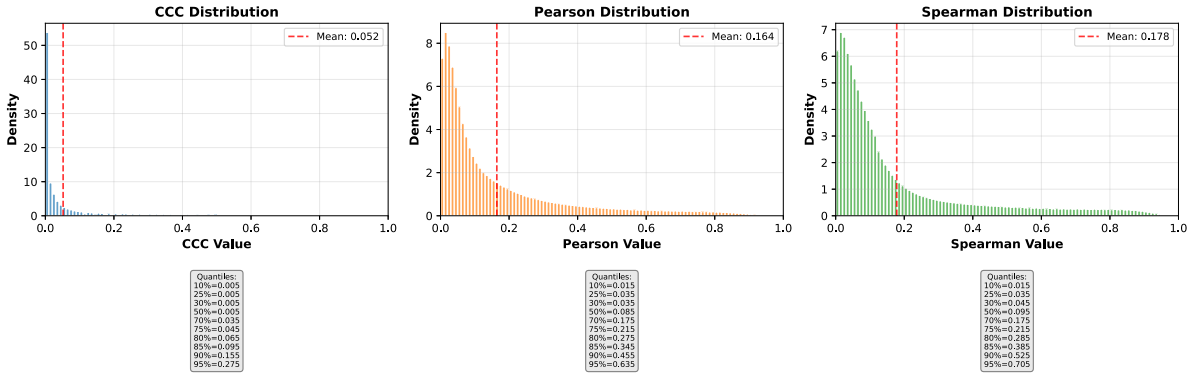


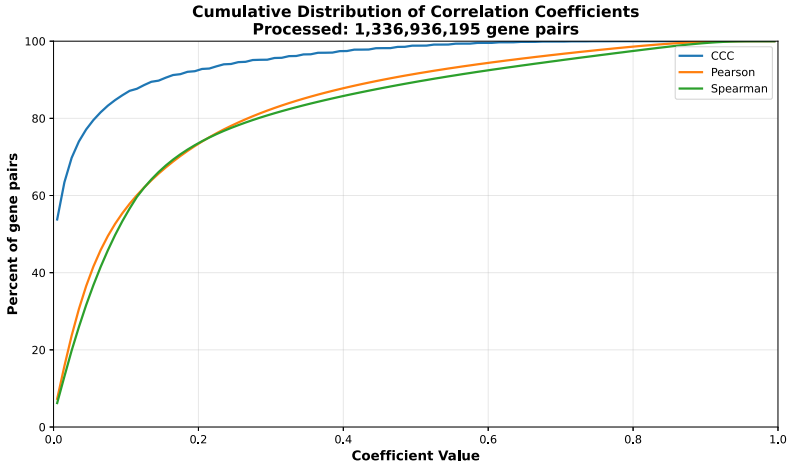
Figure S22: Distribution and UpSet plots for GTEx v8 brain nucleus accumbens basal ganglia.

Brain Putamen Basal Ganglia

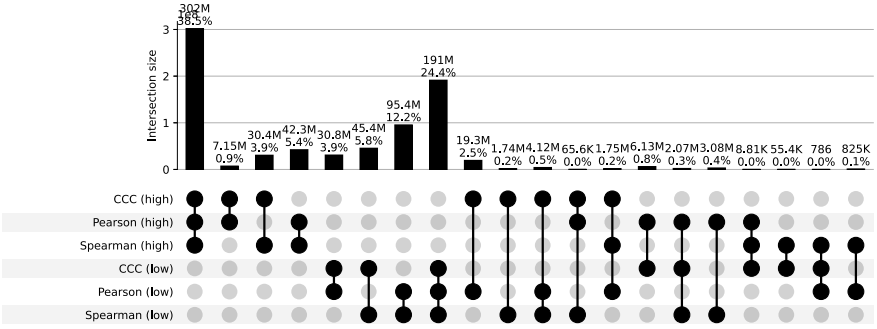
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Putamen Basal Ganglia



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

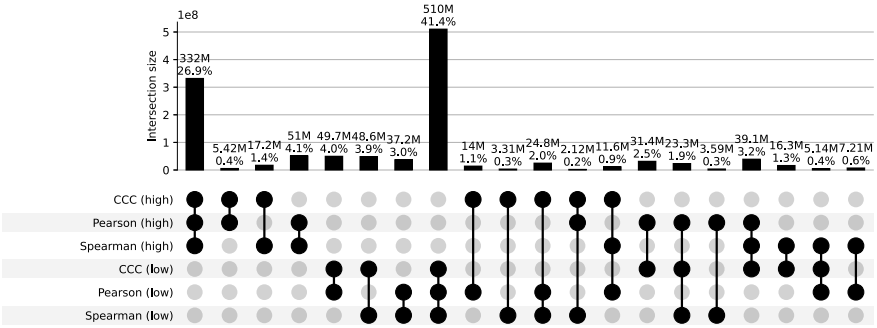
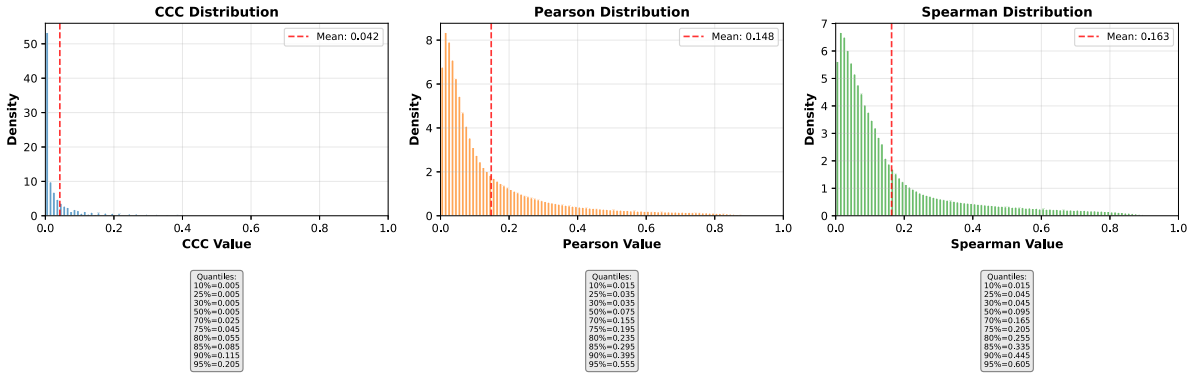


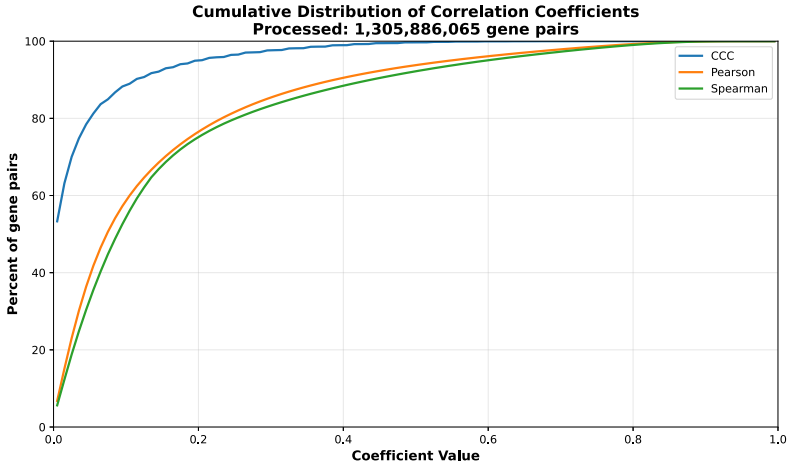
Figure S23: Distribution and UpSet plots for GTEx v8 brain putamen basal ganglia.

Brain Spinal Cord Cervical C1

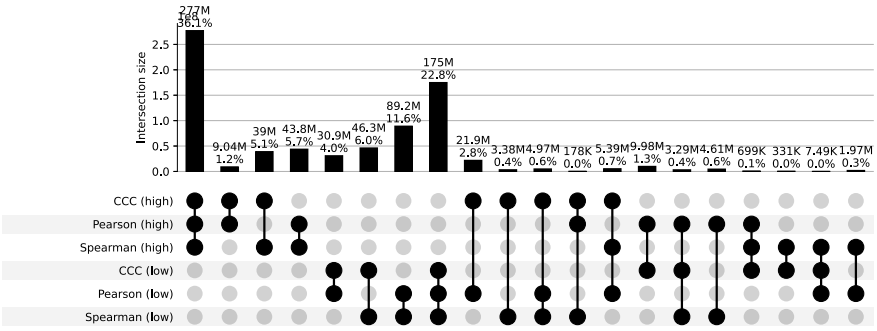
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Spinal Cord Cervical C1



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

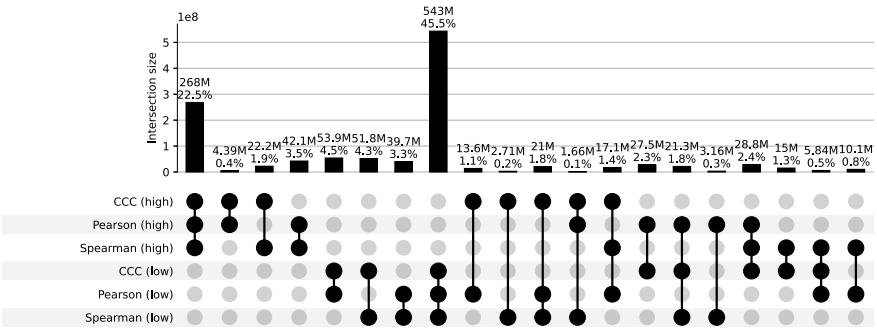
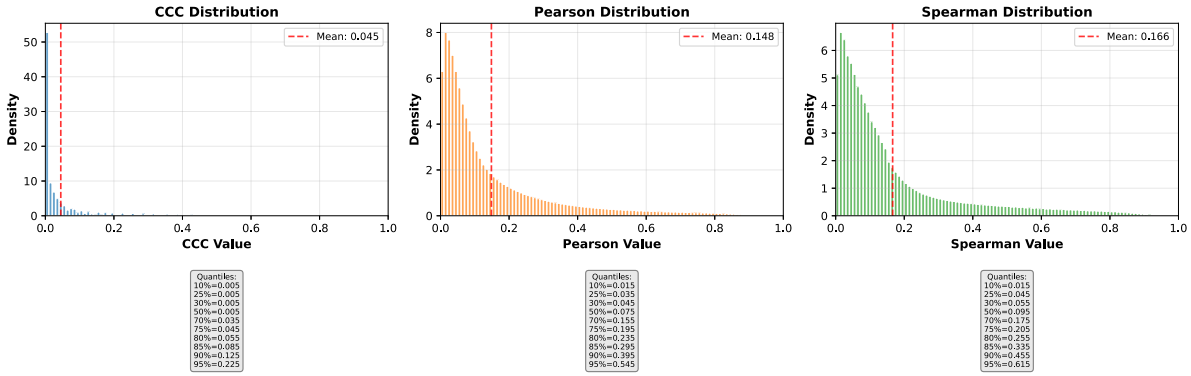


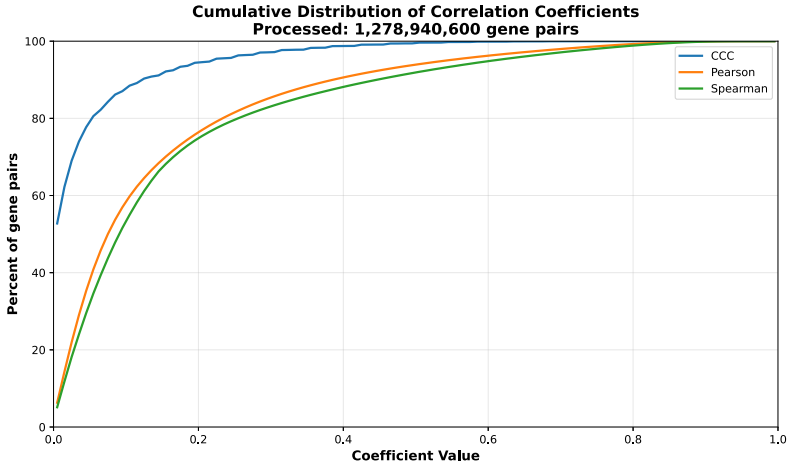
Figure S24: Distribution and UpSet plots for GTEx v8 brain spinal cord cervical C1.

Brain Substantia Nigra

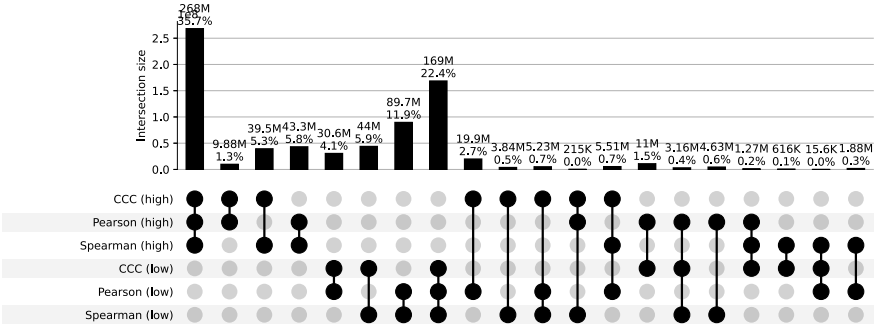
a) Correlation coefficient distributions between gene pairs within GTEx v8 Brain Substantia Nigra



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

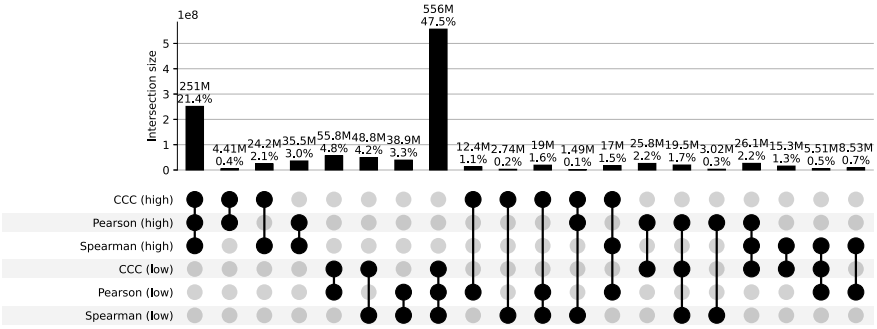
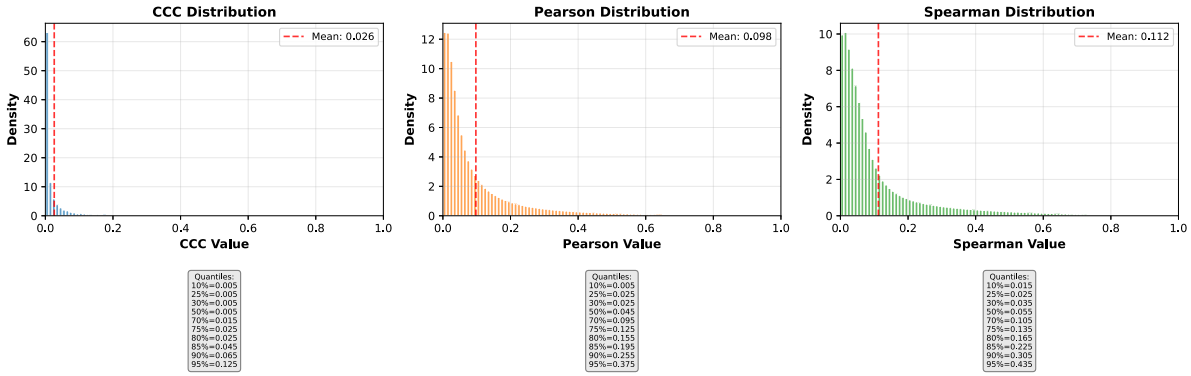


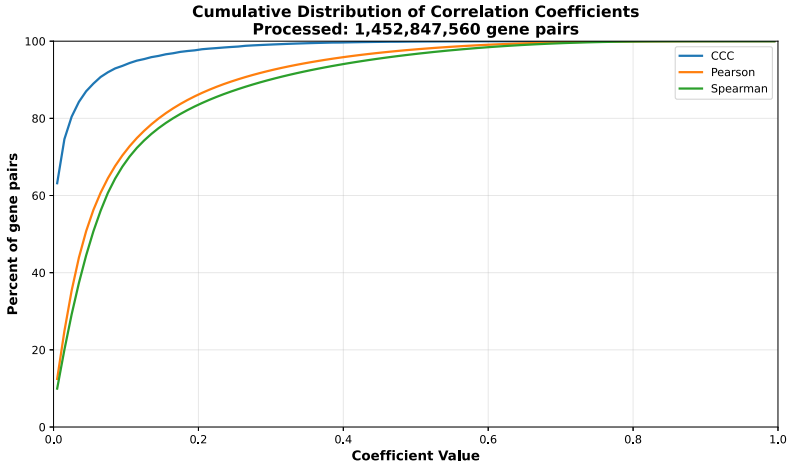
Figure S25: Distribution and UpSet plots for GTEx v8 brain substantia nigra.

Breast Mammary Tissue

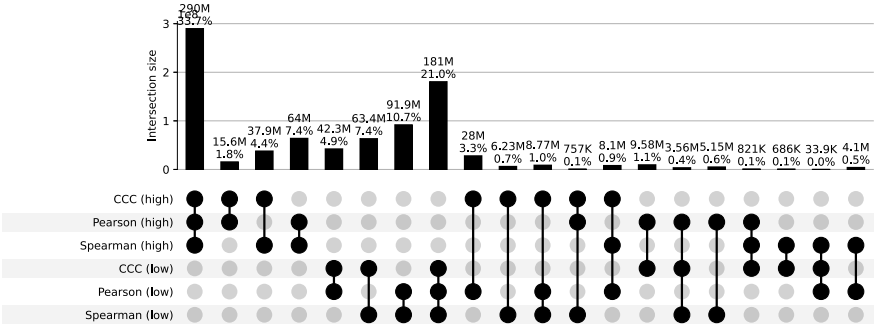
a) Correlation coefficient distributions between gene pairs within GTEx v8 Breast Mammary Tissue



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

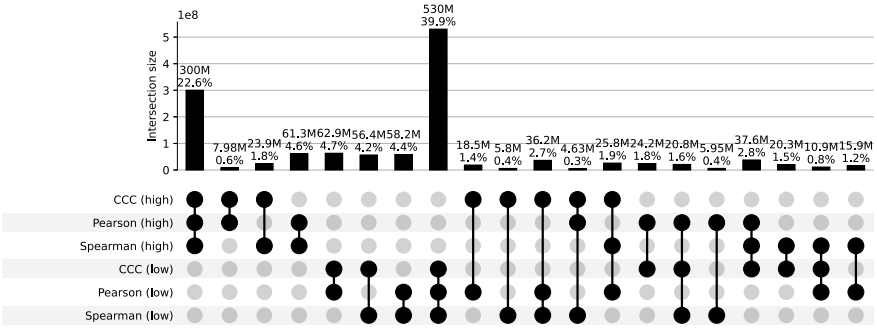
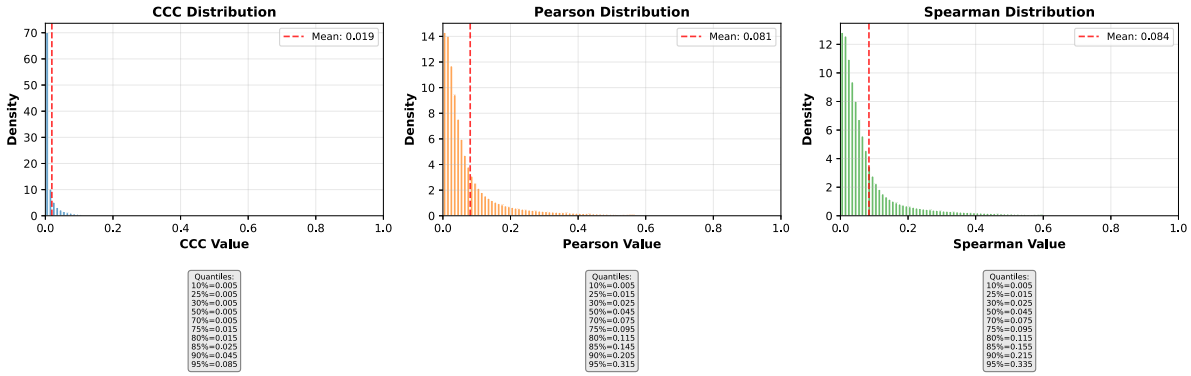


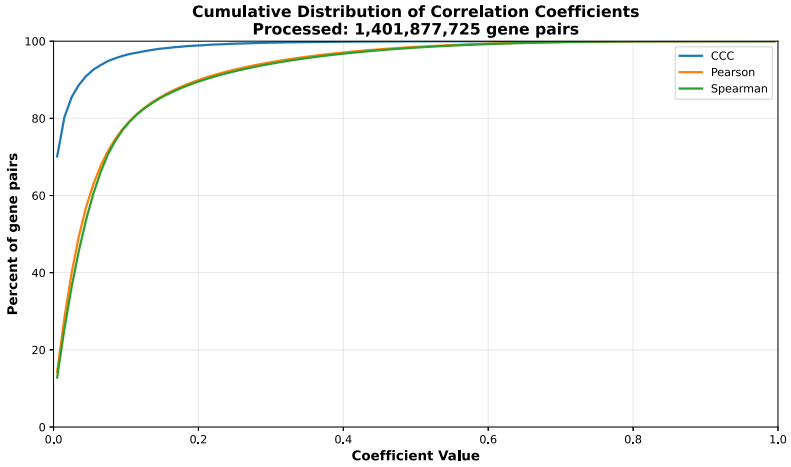
Figure S26: Distribution and UpSet plots for GTEx v8 breast mammary tissue.

Cells Cultured Fibroblasts

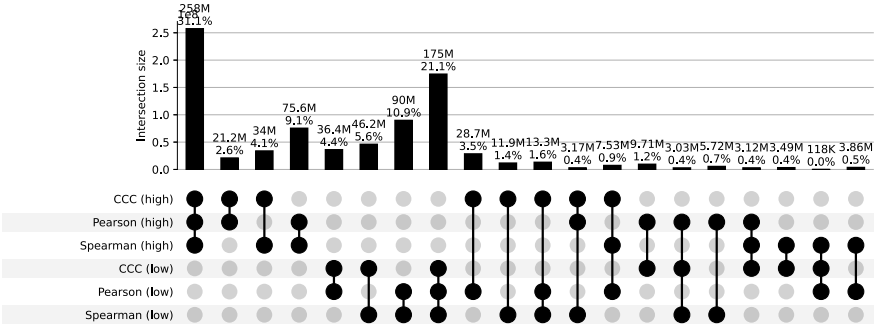
a) Correlation coefficient distributions between gene pairs within GTEx v8 Cells Cultured Fibroblasts



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

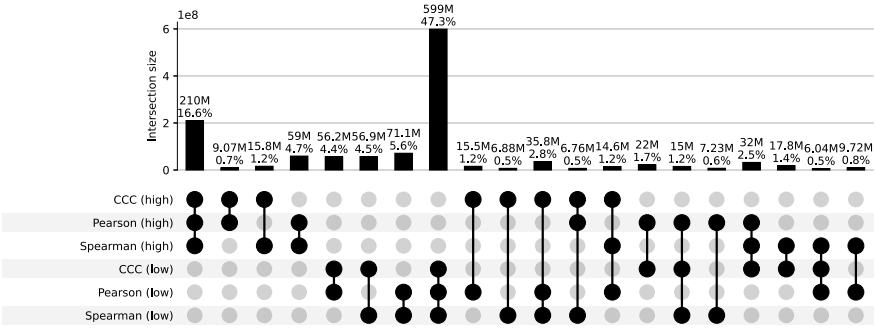
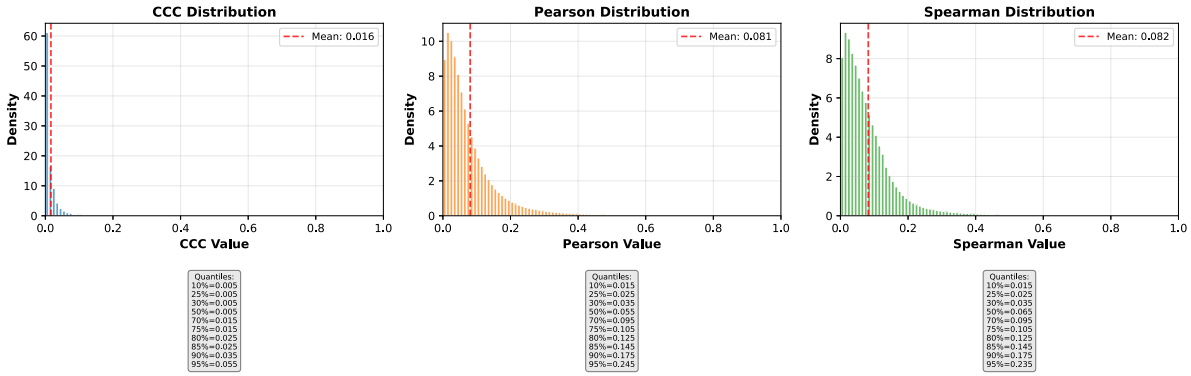


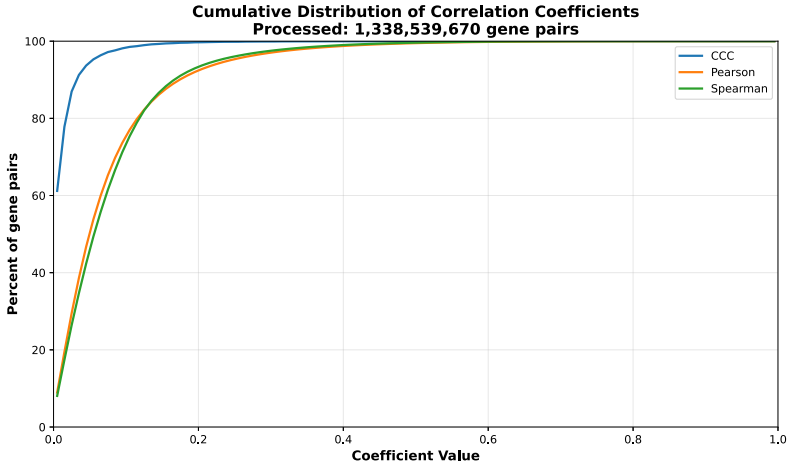
Figure S27: Distribution and UpSet plots for GTEx v8 cells cultured fibroblasts.

Cells Ebvtransformed Lymphocytes

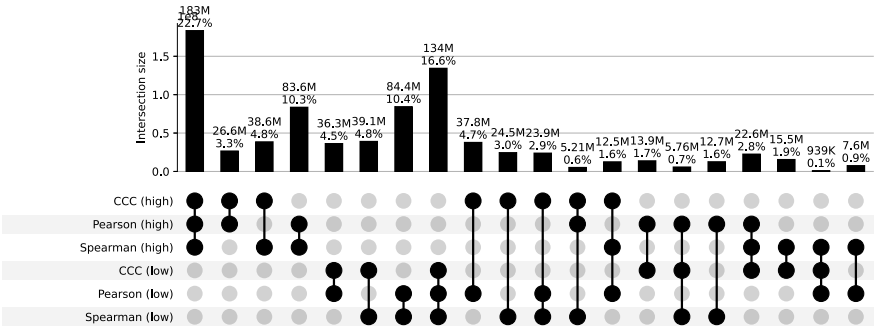
a) Correlation coefficient distributions between gene pairs within GTEx v8 Cells Ebvtransformed Lymphocytes



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

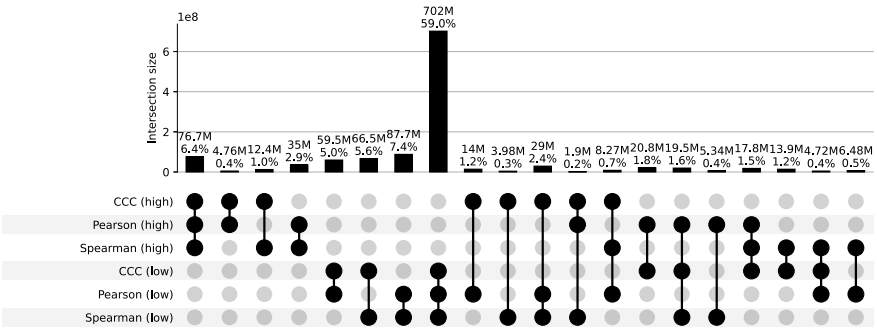
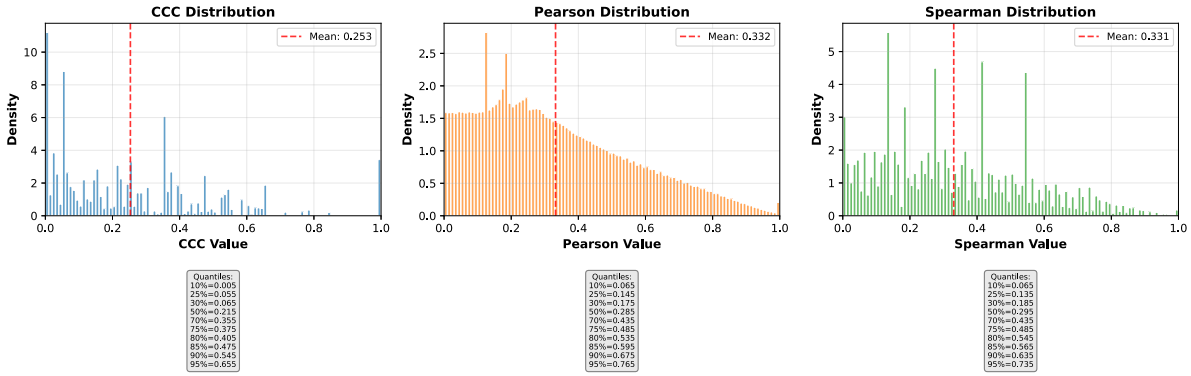


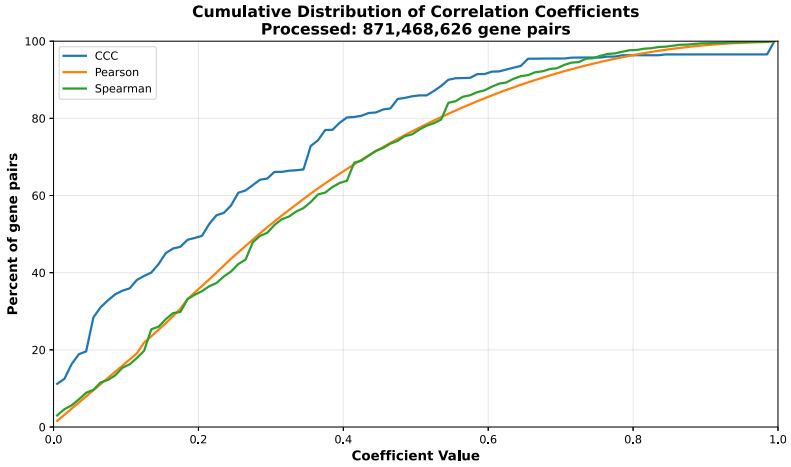
Figure S28: Distribution and UpSet plots for GTEx v8 cells EBV-transformed lymphocytes.

Cervix Ectocervix

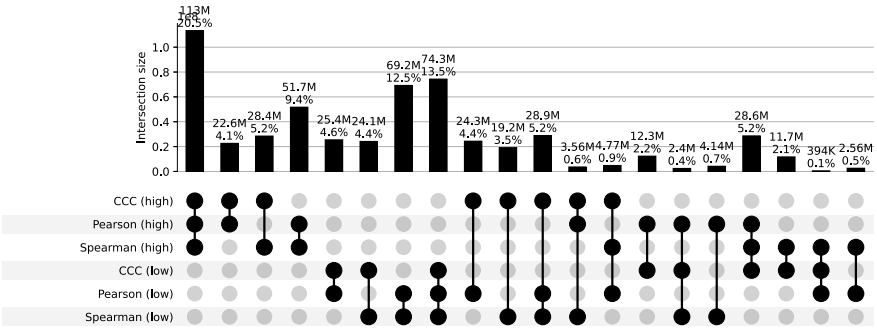
a) Correlation coefficient distributions between gene pairs within GTEx v8 Cervix Ectocervix



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

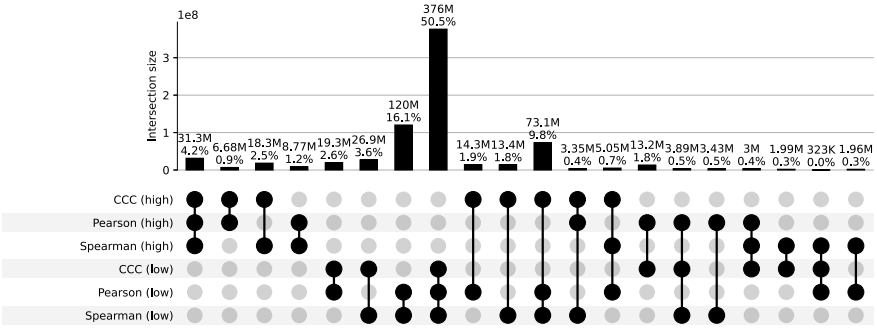
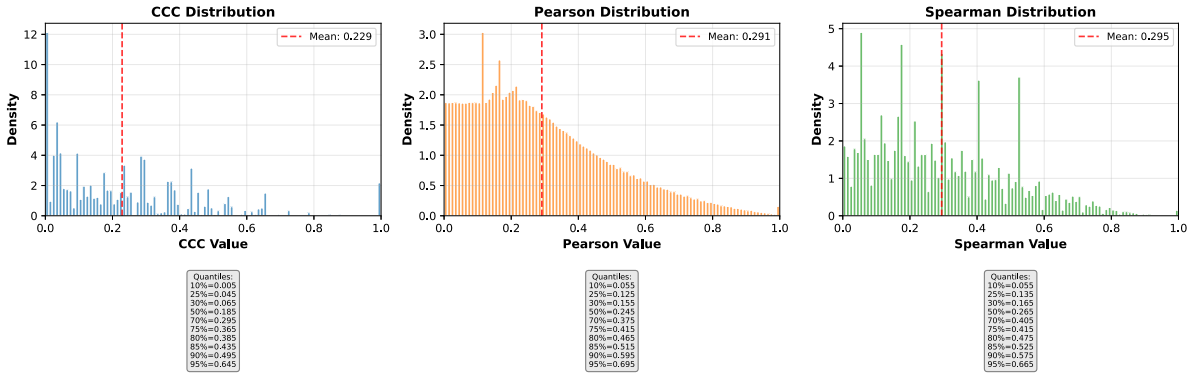


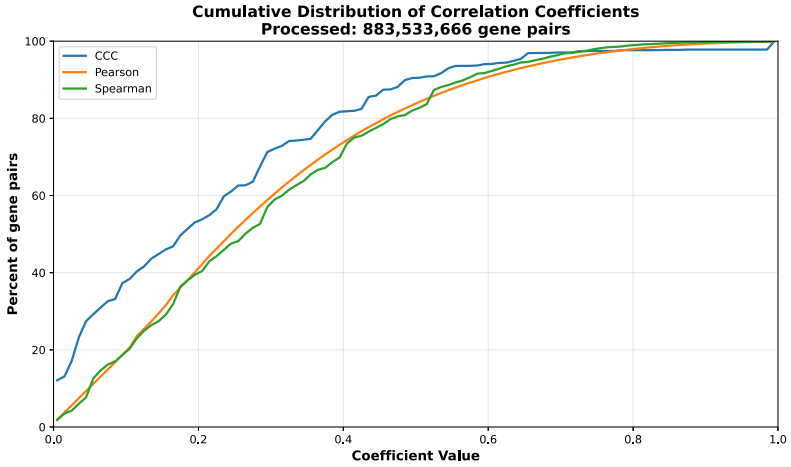
Figure S29: Distribution and UpSet plots for GTEx v8 cervix ectocervix.

Cervix Endocervix

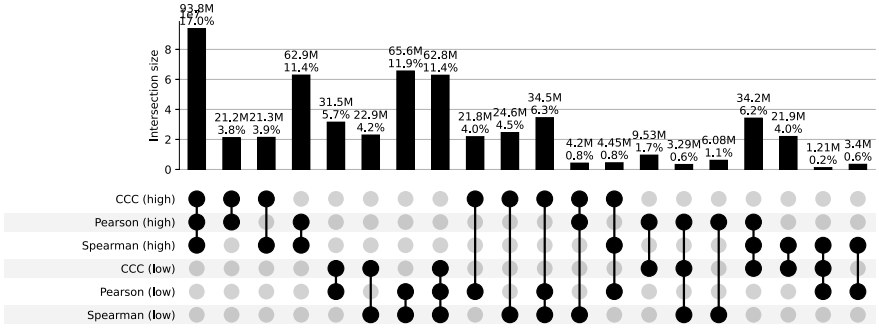
a) Correlation coefficient distributions between gene pairs within GTEx v8 Cervix Endocervix



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

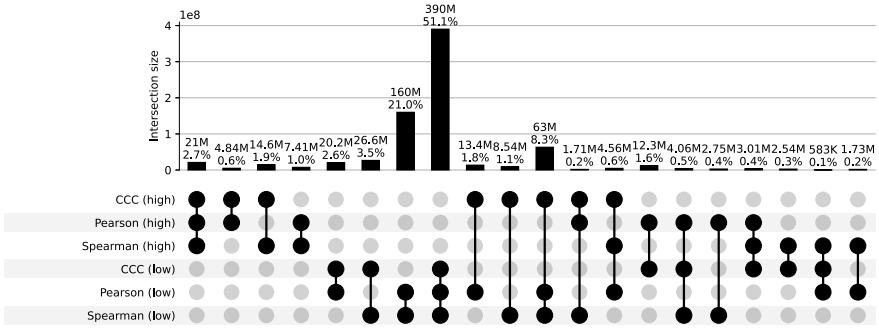
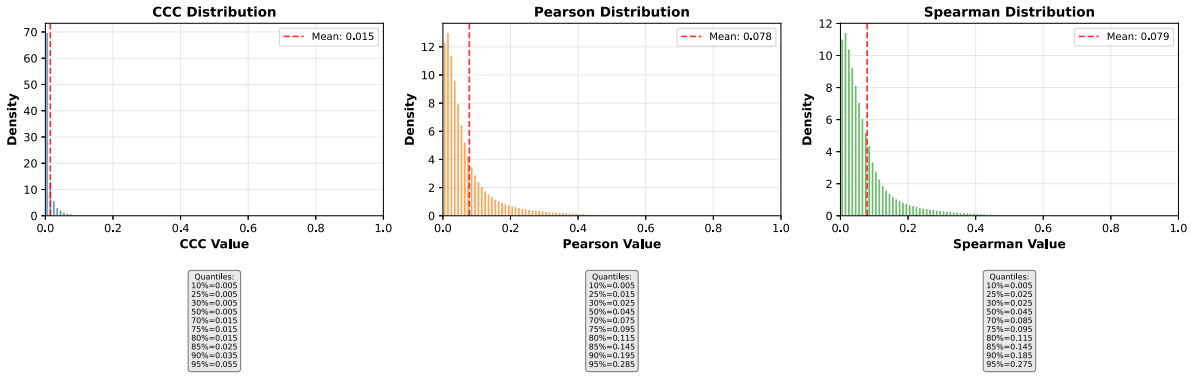


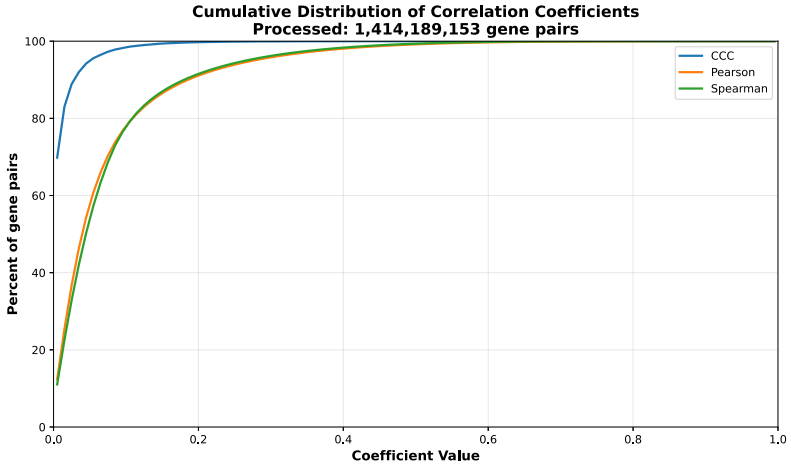
Figure S30: Distribution and UpSet plots for GTEx v8 cervix endocervix.

Colon Sigmoid

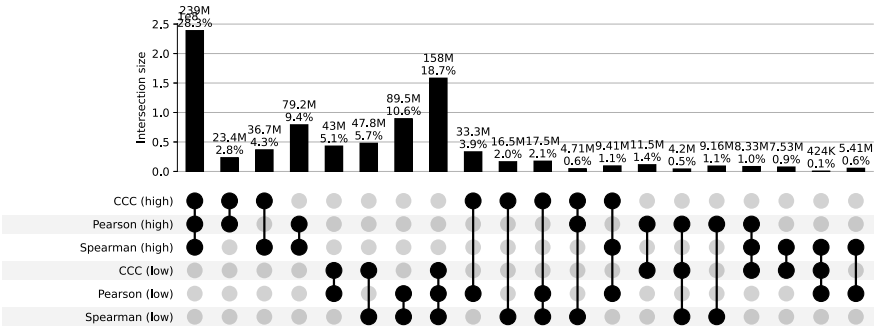
a) Correlation coefficient distributions between gene pairs within GTEx v8 Colon Sigmoid



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

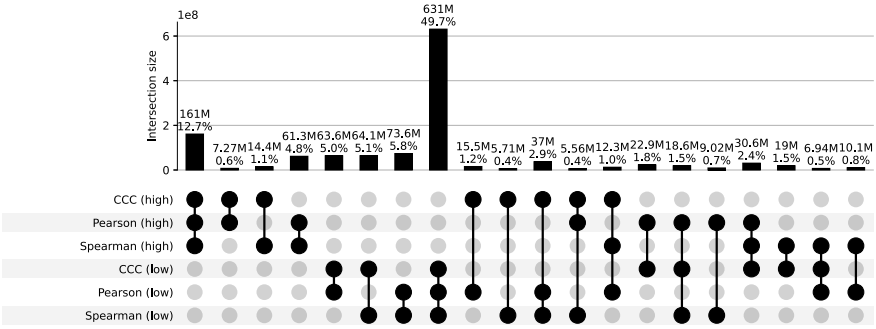
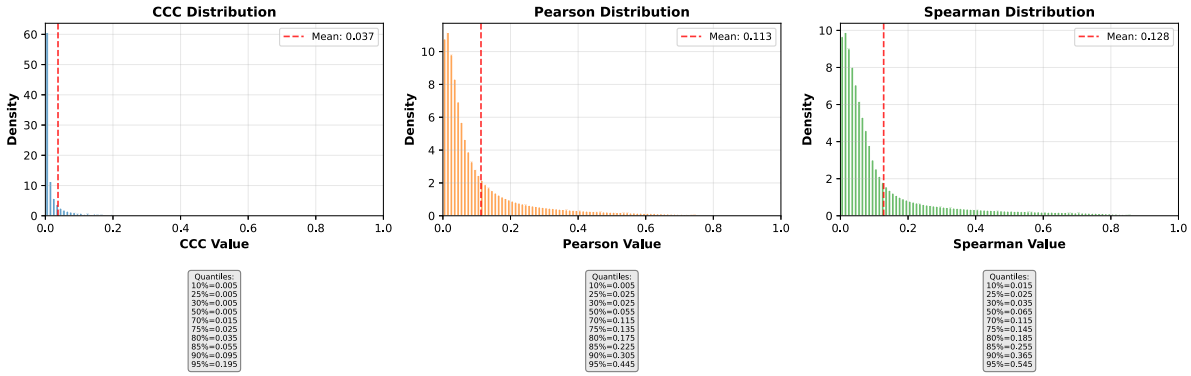


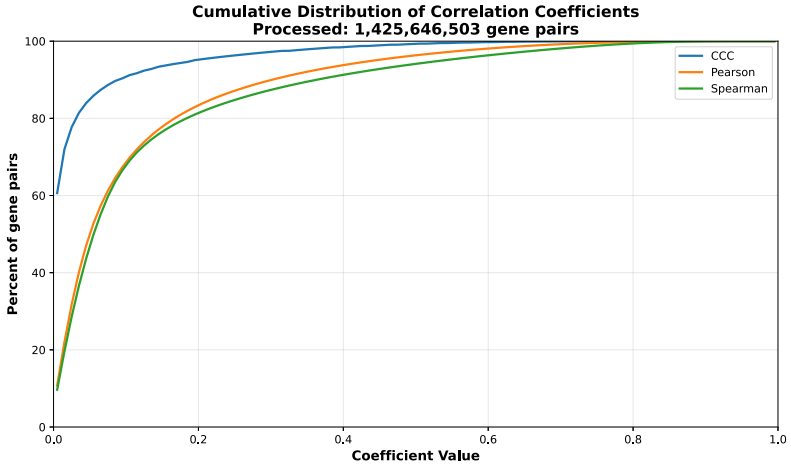
Figure S31: Distribution and UpSet plots for GTEx v8 colon sigmoid.

Colon Transverse

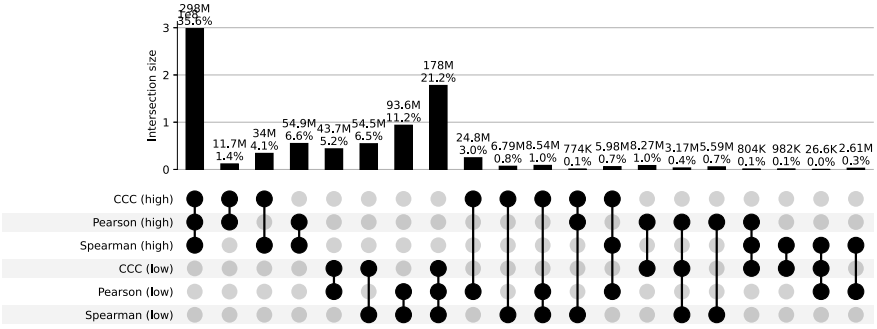
a) Correlation coefficient distributions between gene pairs within GTEx v8 Colon Transverse



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

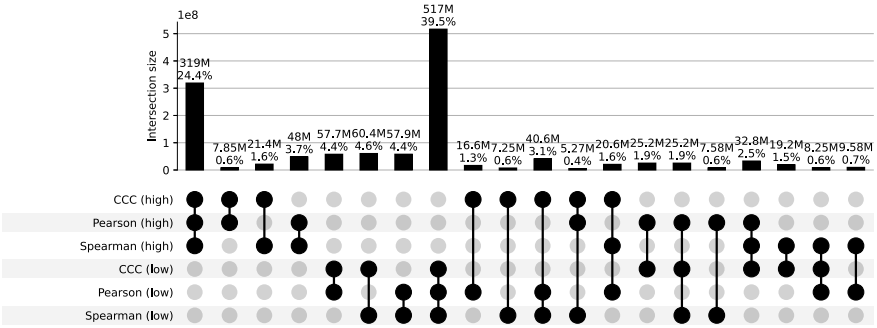
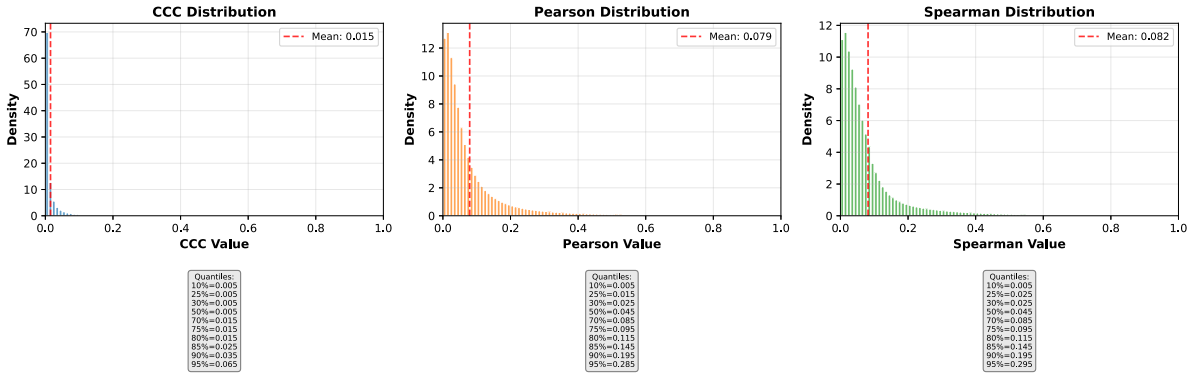


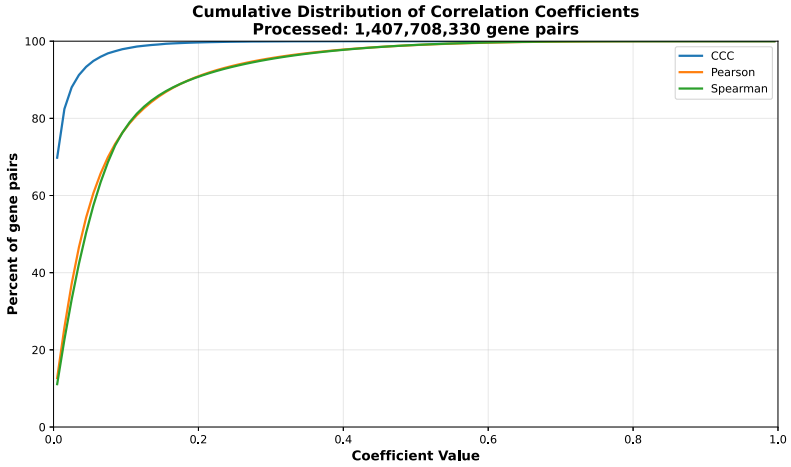
Figure S32: Distribution and UpSet plots for GTEx v8 colon transverse.

Esophagus Gastroesophageal Junction

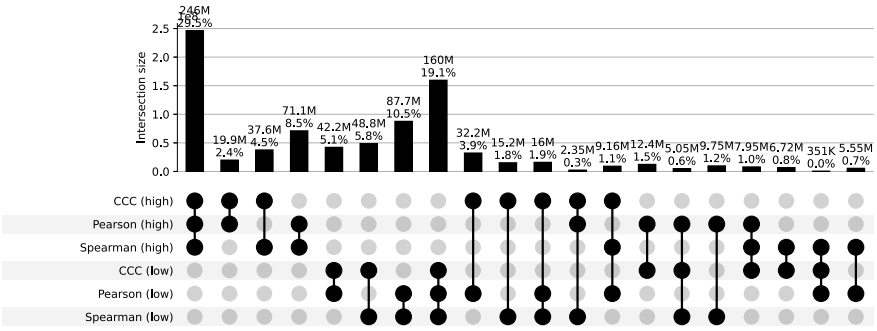
a) Correlation coefficient distributions between gene pairs within GTEx v8 Esophagus Gastroesophageal Junction



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

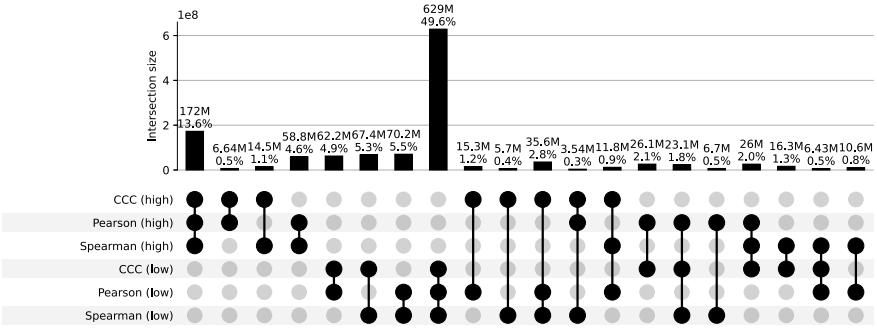
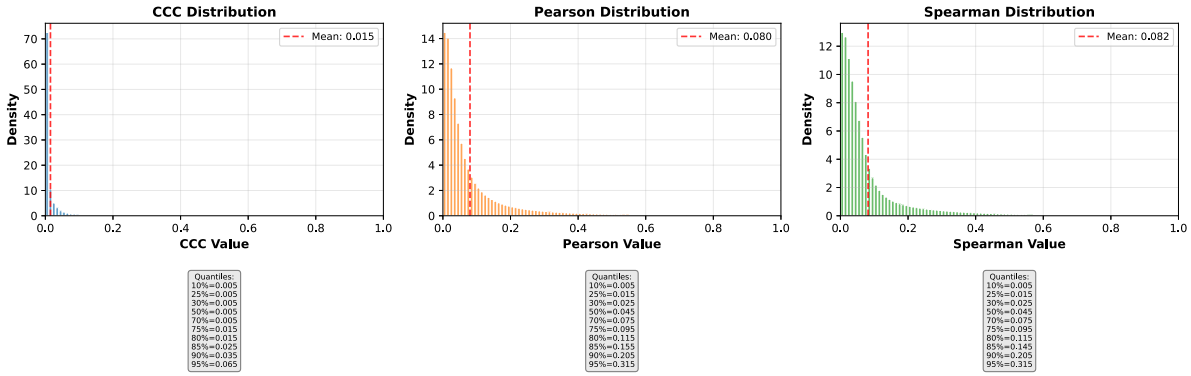


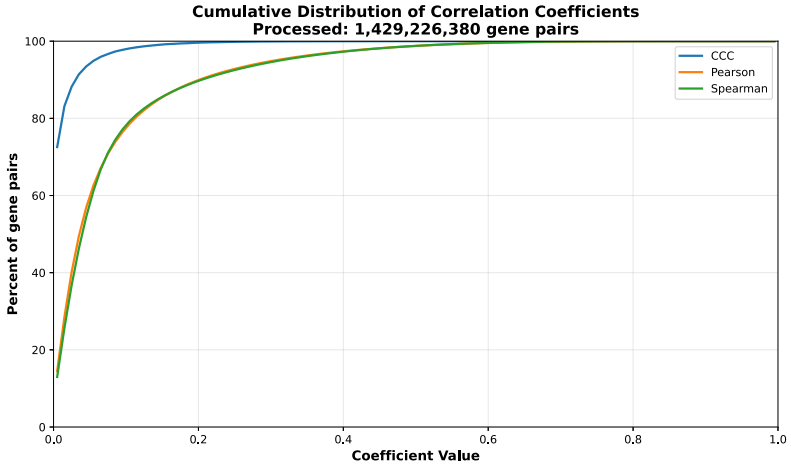
Figure S33: Distribution and UpSet plots for GTEx v8 esophagus gastroesophageal junction.

Esophagus Mucosa

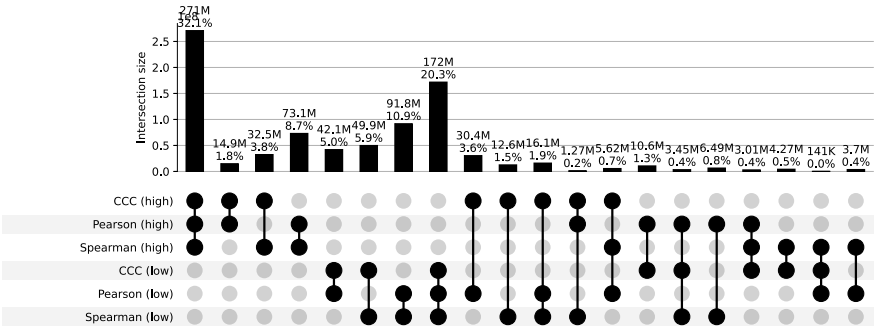
a) Correlation coefficient distributions between gene pairs within GTEx v8 Esophagus Mucosa



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

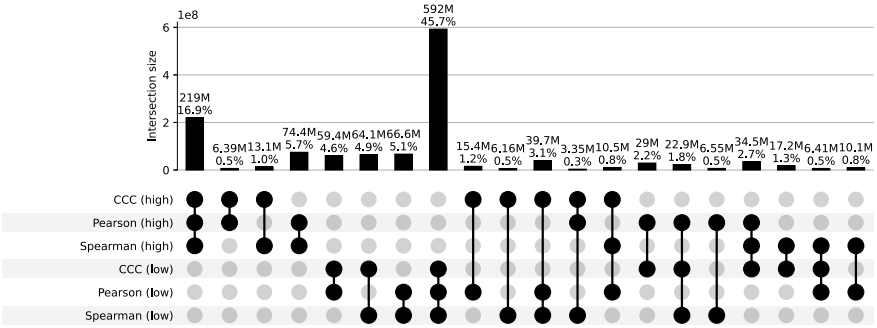
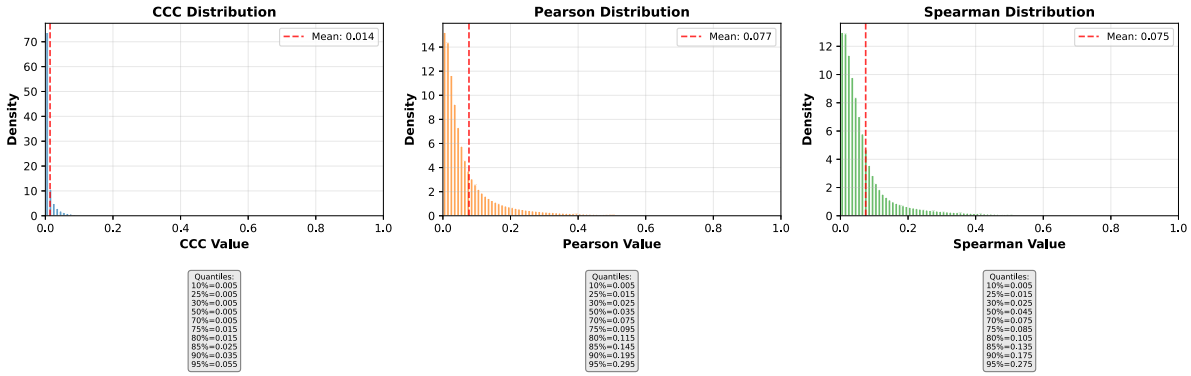


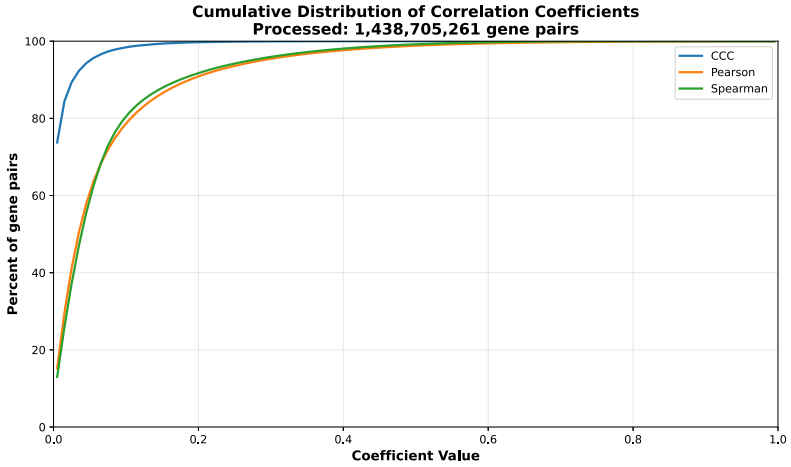
Figure S34: Distribution and UpSet plots for GTEx v8 esophagus mucosa.

Esophagus Muscularis

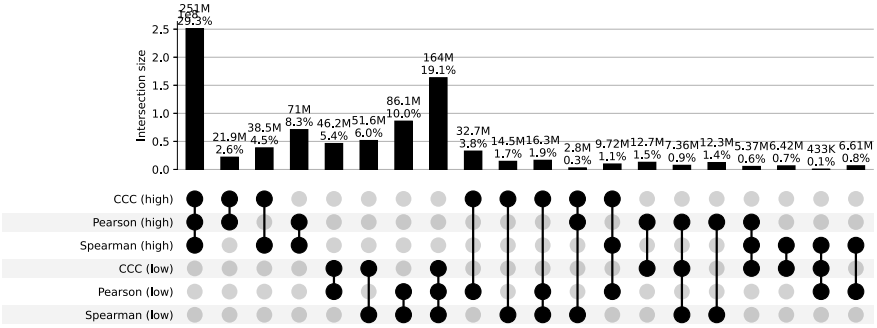
a) Correlation coefficient distributions between gene pairs within GTEx v8 Esophagus Muscularis



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

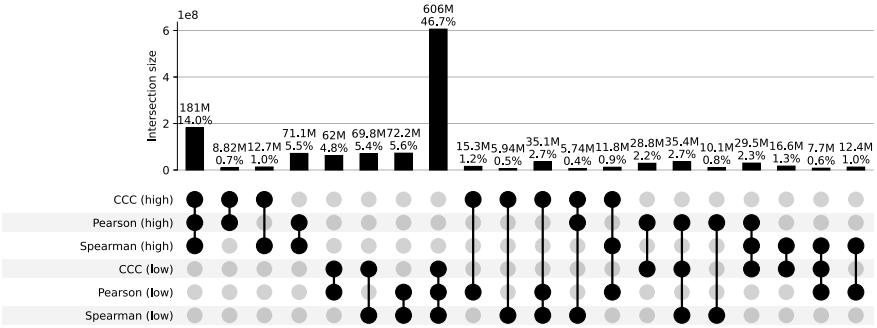
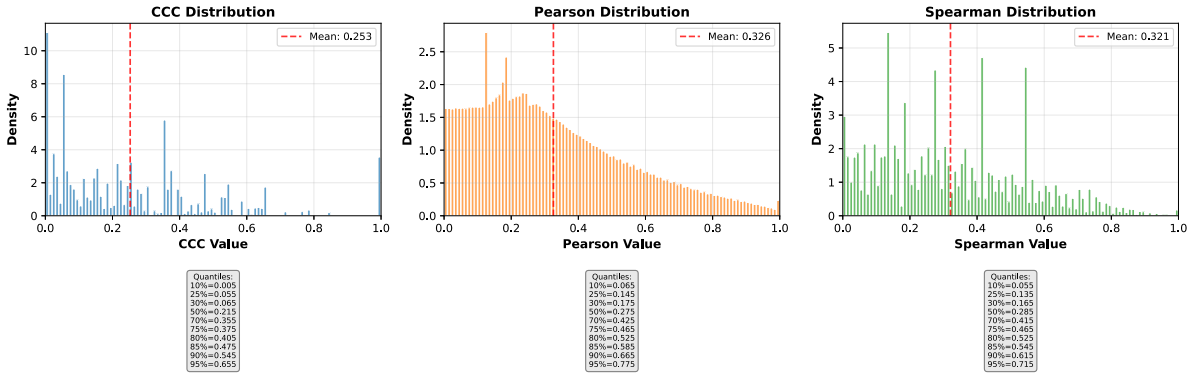


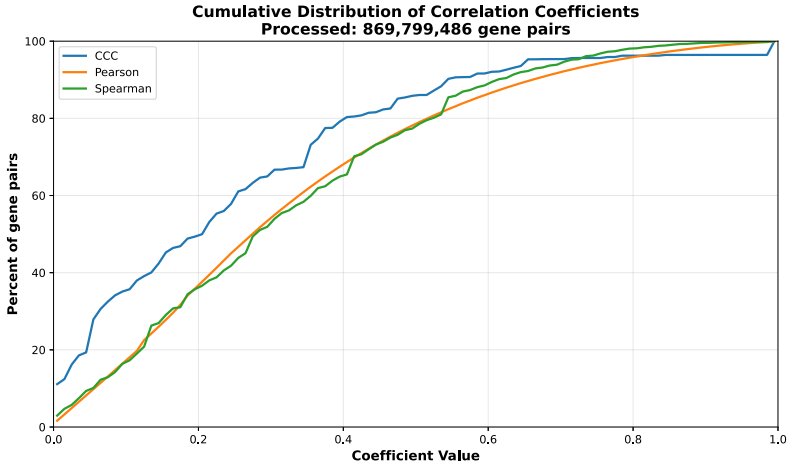
Figure S35: Distribution and UpSet plots for GTEx v8 esophagus muscularis.

Fallopian Tube

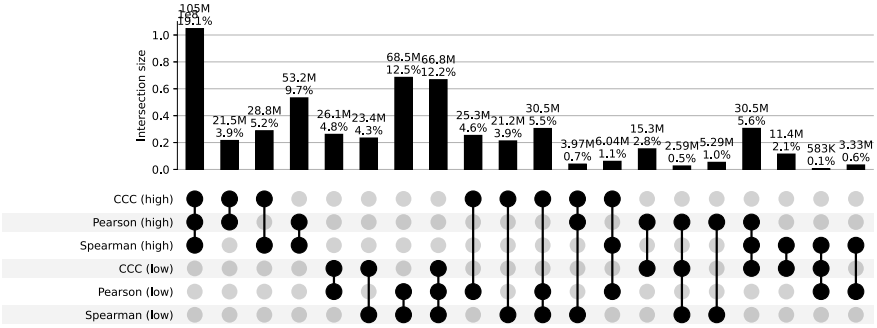
a) Correlation coefficient distributions between gene pairs within GTEx v8 Fallopian Tube



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

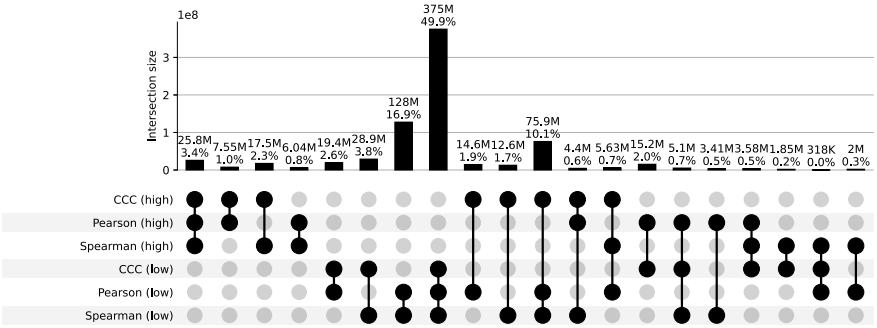
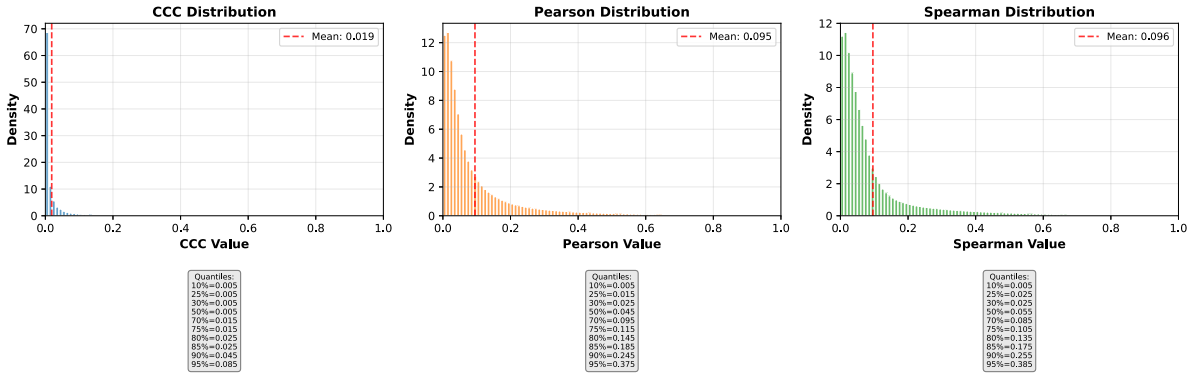


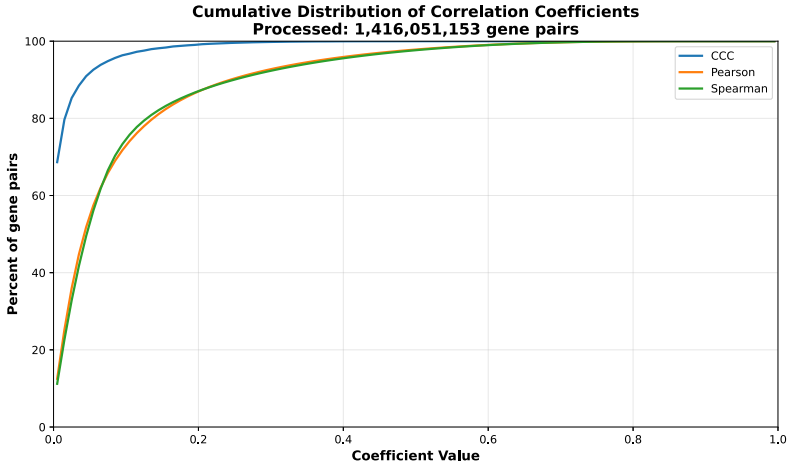
Figure S36: Distribution and UpSet plots for GTEx v8 fallopian tube.

Heart Atrial Appendage

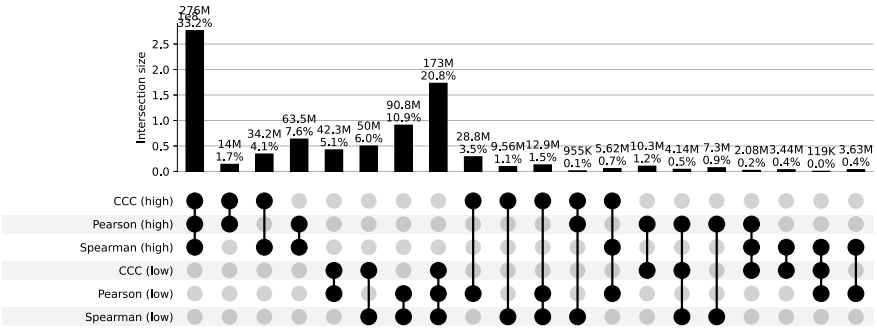
a) Correlation coefficient distributions between gene pairs within GTEx v8 Heart Atrial Appendage



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

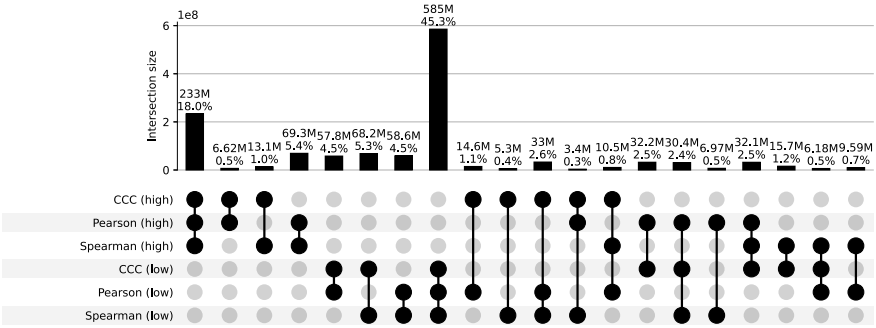
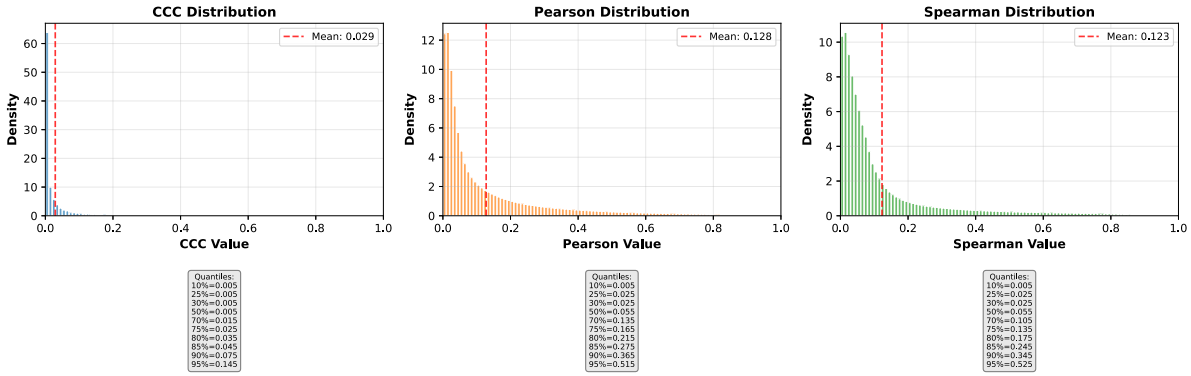


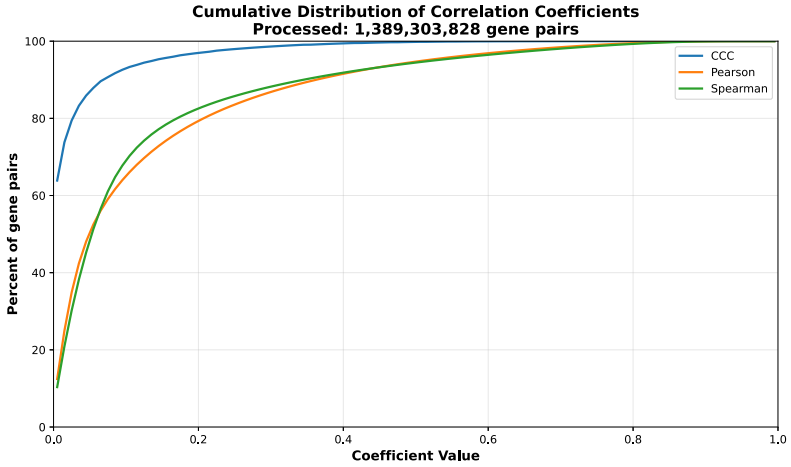
Figure S37: Distribution and UpSet plots for GTEx v8 heart atrial appendage.

Heart Left Ventricle

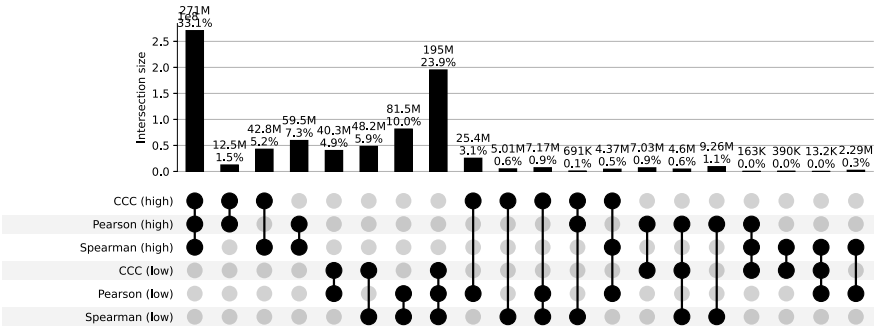
a) Correlation coefficient distributions between gene pairs within GTEx v8 Heart Left Ventricle



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

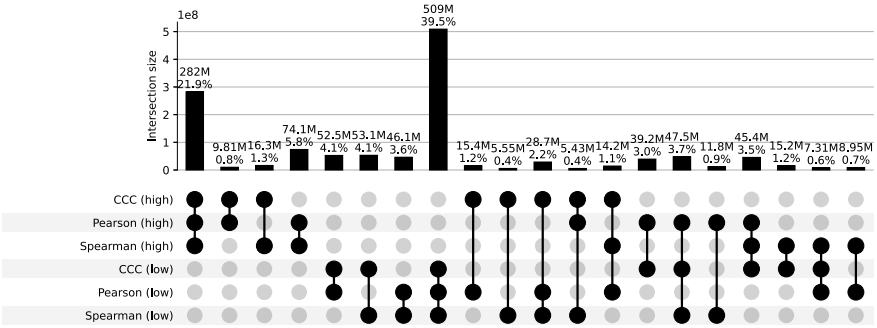
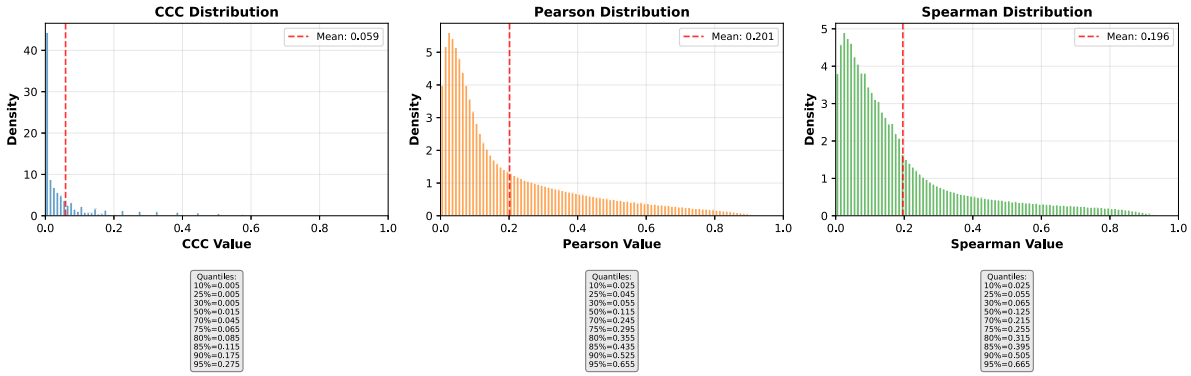


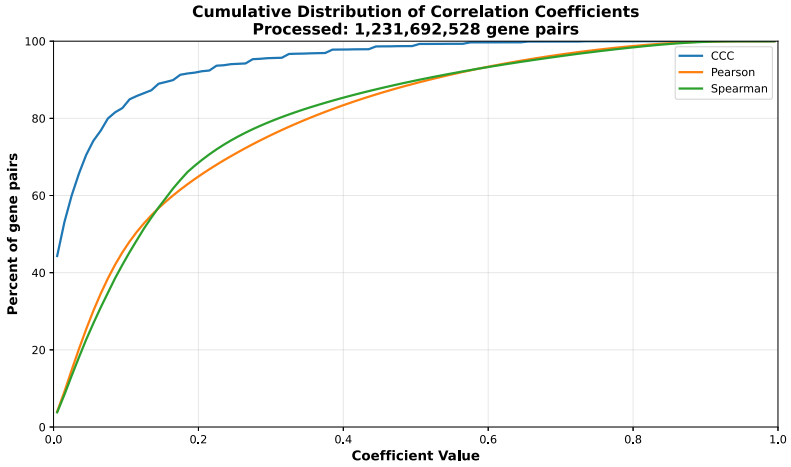
Figure S38: Distribution and UpSet plots for GTEx v8 heart left ventricle.

Kidney Cortex

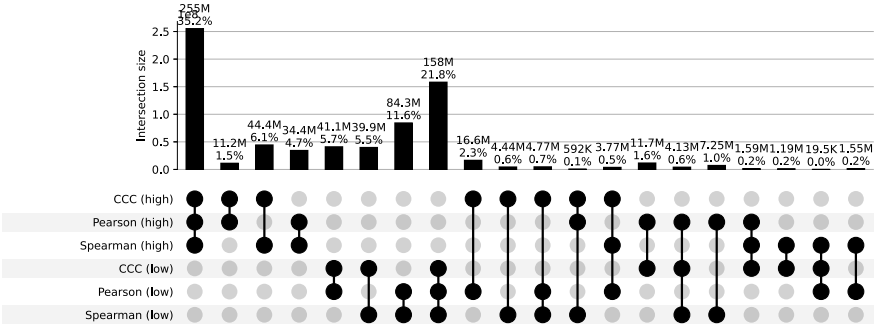
a) Correlation coefficient distributions between gene pairs within GTEx v8 Kidney Cortex



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

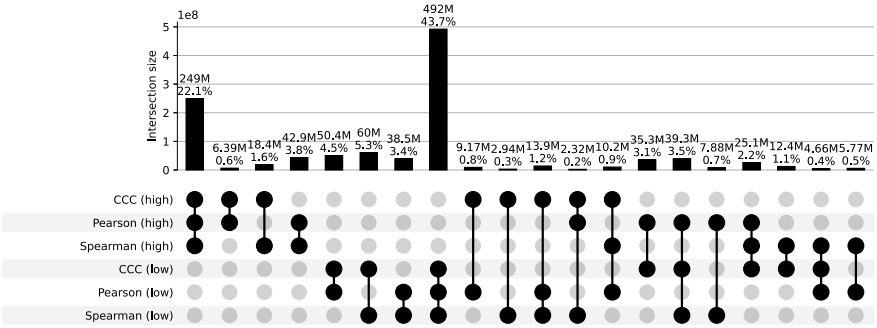
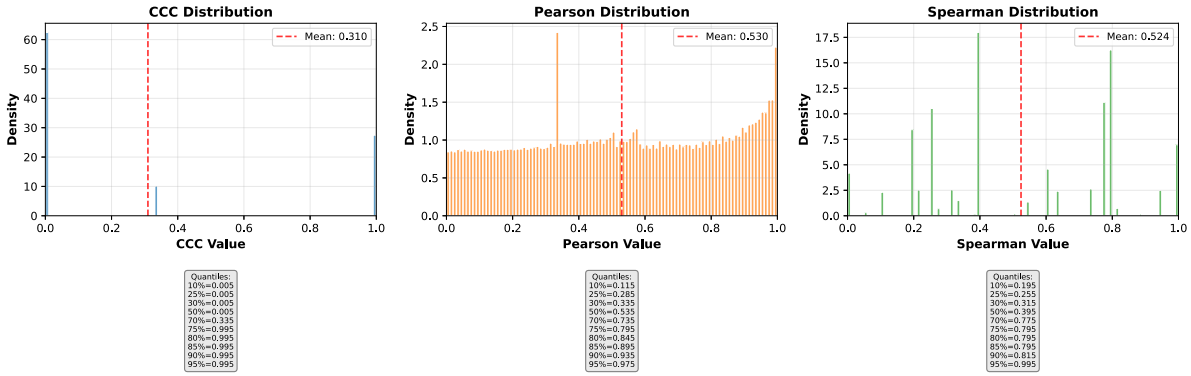


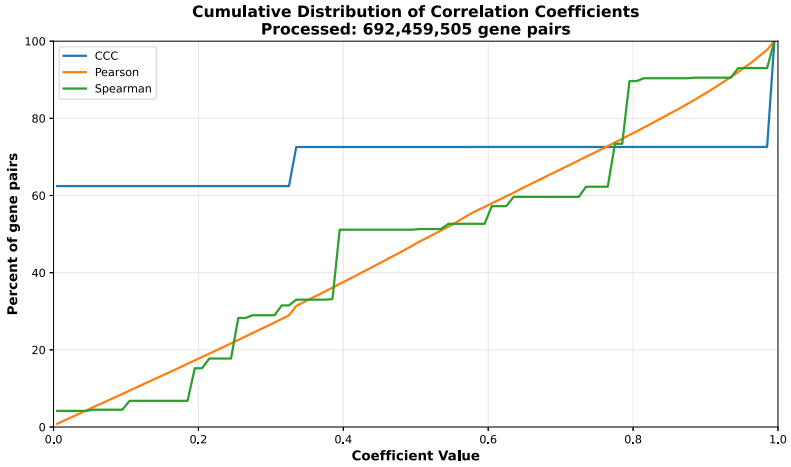
Figure S39: Distribution and UpSet plots for GTEx v8 kidney cortex.

Kidney Medulla

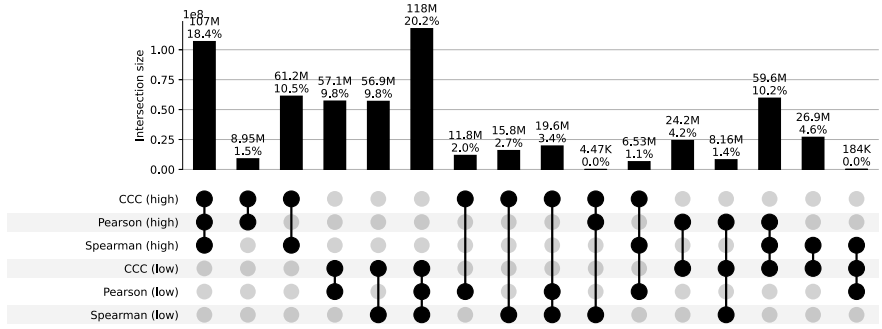
a) Correlation coefficient distributions between gene pairs within GTEx v8 Kidney Medulla



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

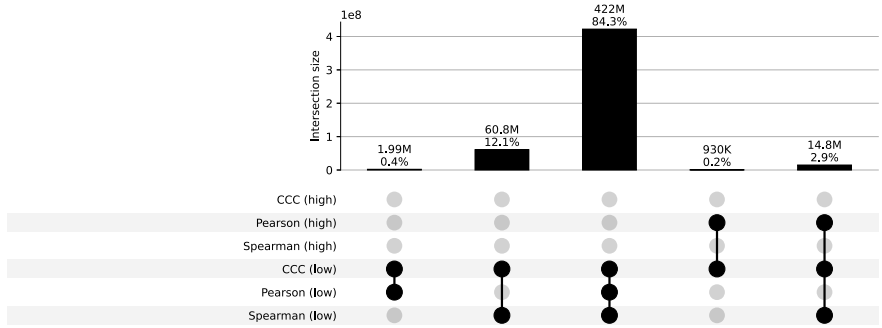
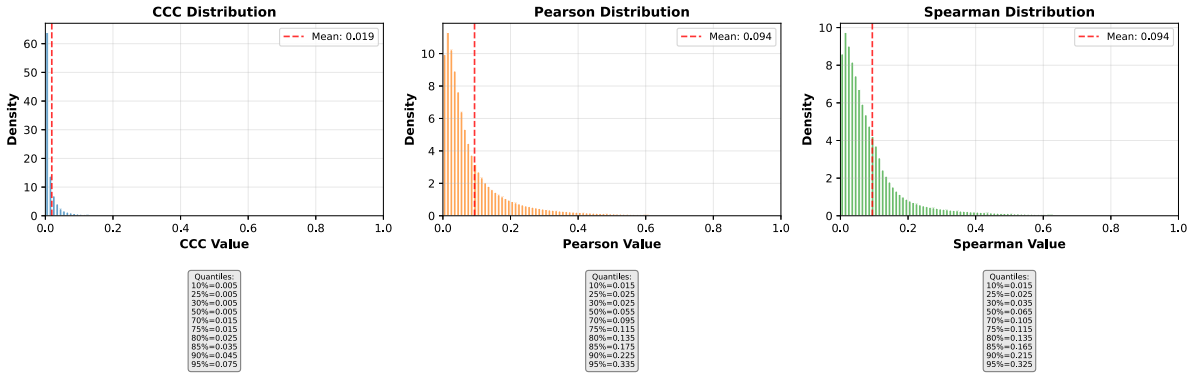


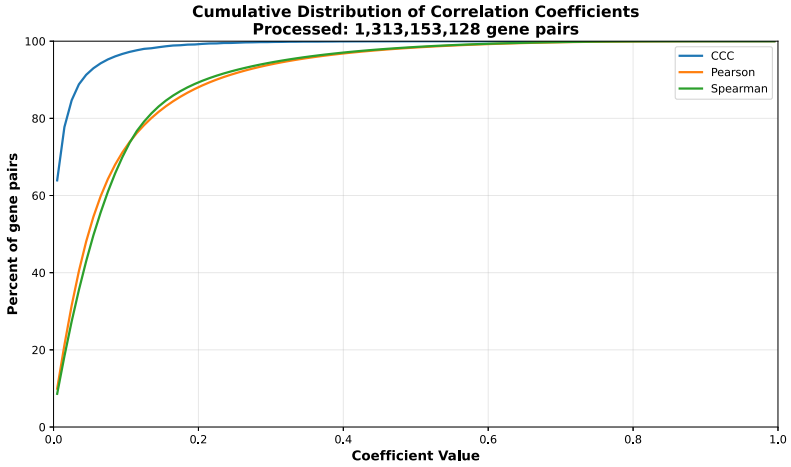
Figure S40: Distribution and UpSet plots for GTEx v8 kidney medulla.

Liver

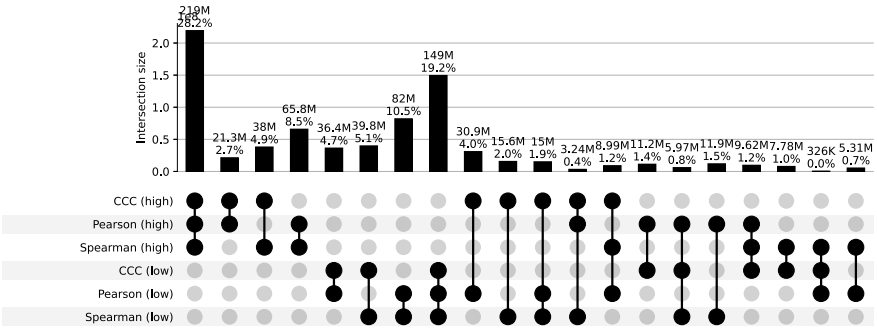
a) Correlation coefficient distributions between gene pairs within GTEx v8 Liver



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

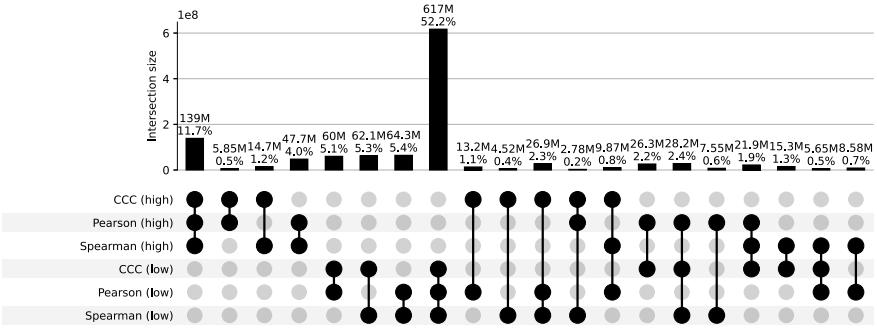
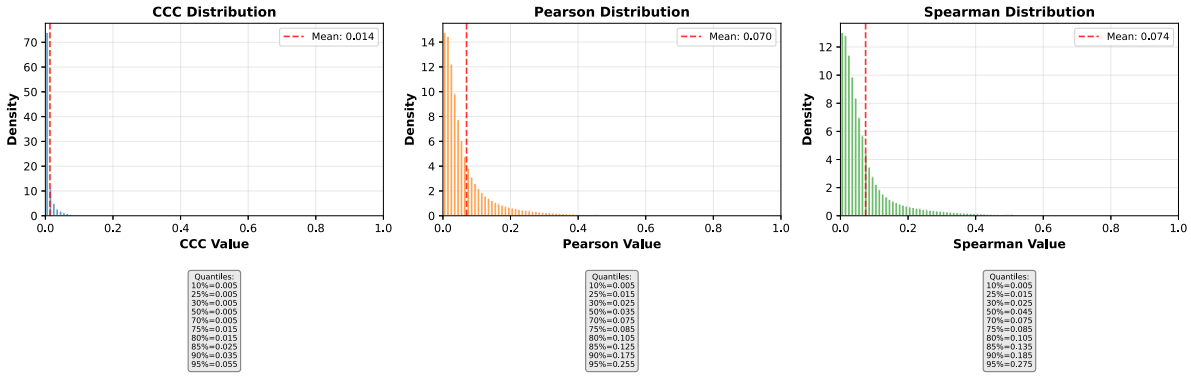


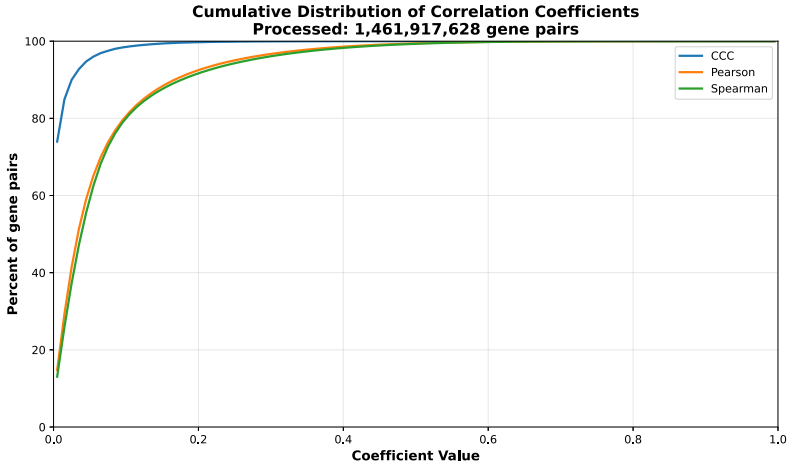
Figure S41: Distribution and UpSet plots for GTEx v8 liver.

Lung

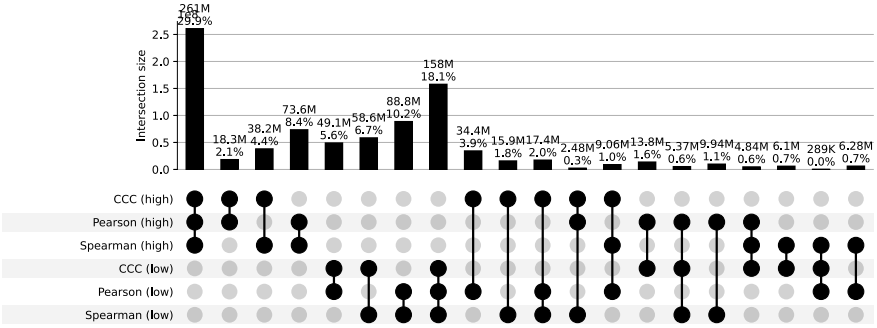
a) Correlation coefficient distributions between gene pairs within GTEx v8 Lung



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

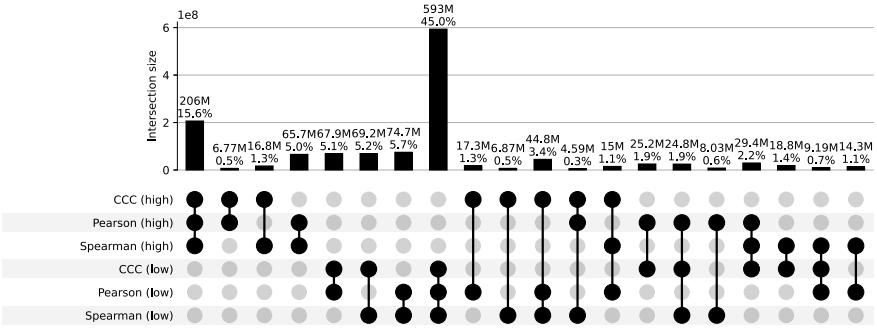
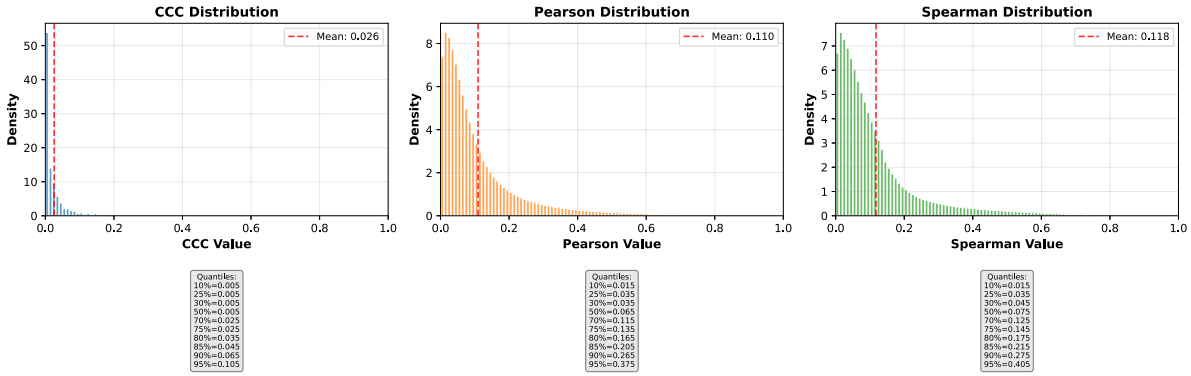


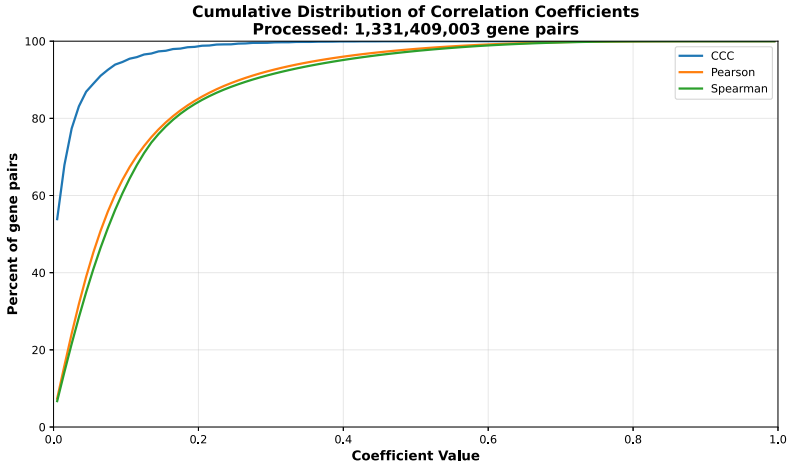
Figure S42: Distribution and UpSet plots for GTEx v8 lung.

Minor Salivary Gland

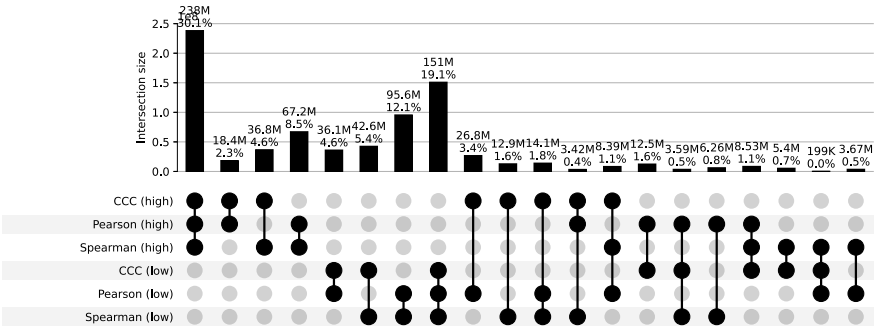
a) Correlation coefficient distributions between gene pairs within GTEx v8 Minor Salivary Gland



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

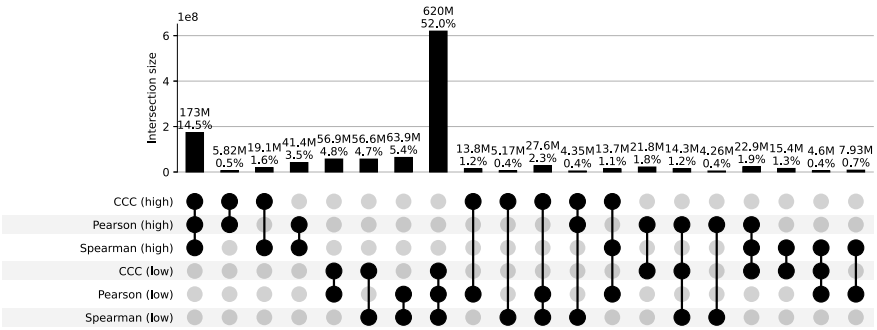
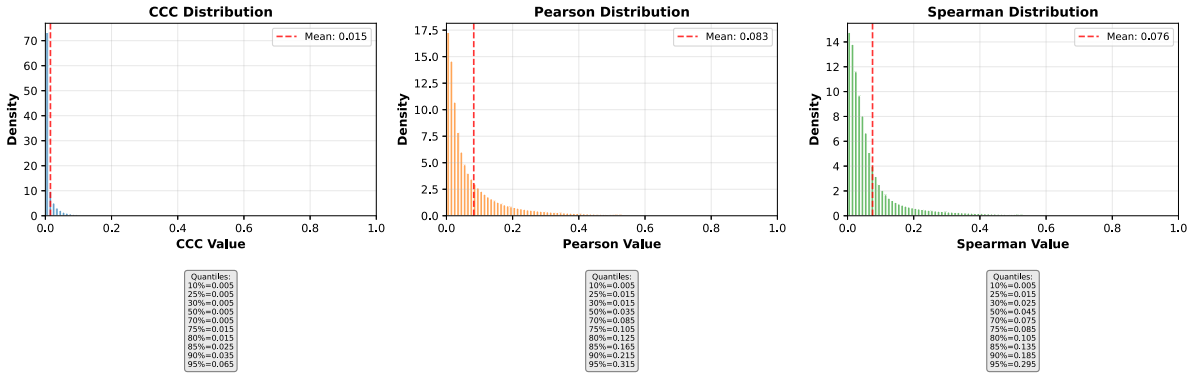


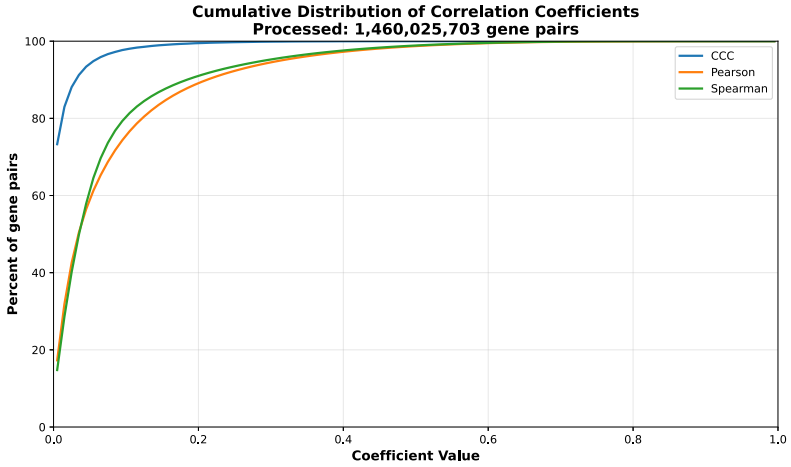
Figure S43: Distribution and UpSet plots for GTEx v8 minor salivary gland.

Muscle Skeletal

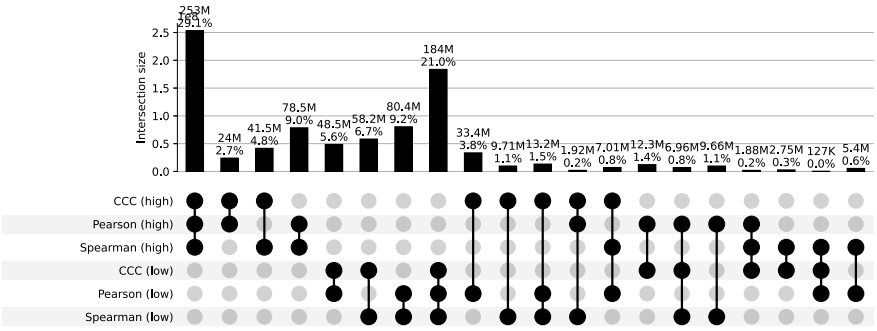
a) Correlation coefficient distributions between gene pairs within GTEx v8 Muscle Skeletal



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

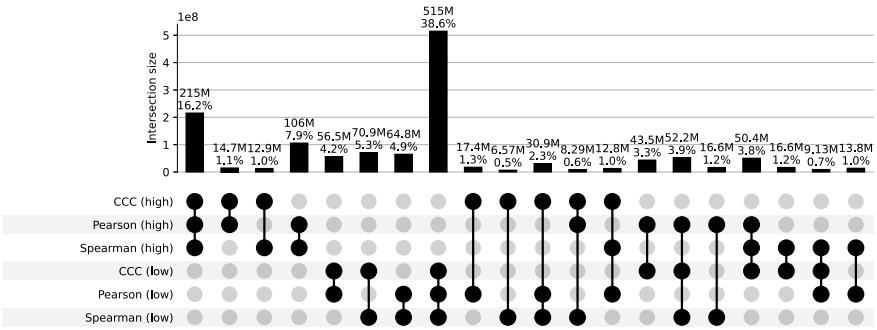
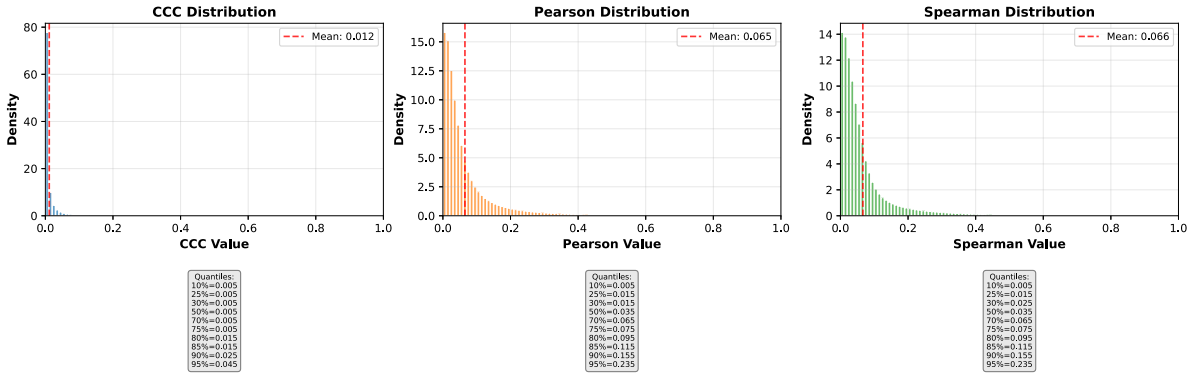


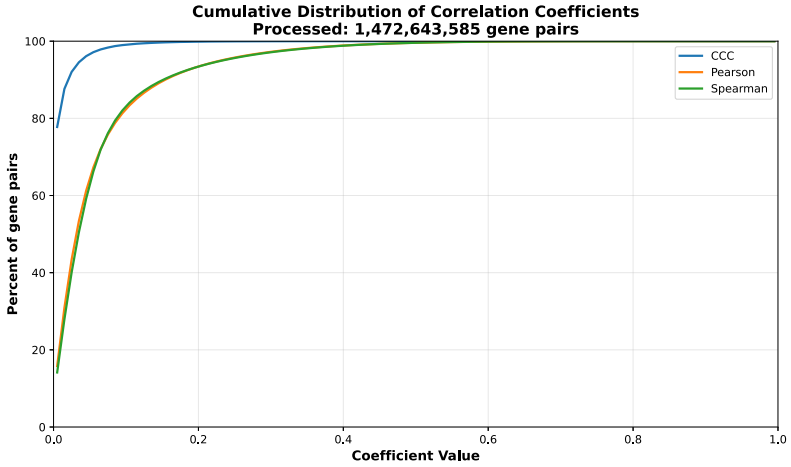
Figure S44: Distribution and UpSet plots for GTEx v8 muscle skeletal.

Nerve Tibial

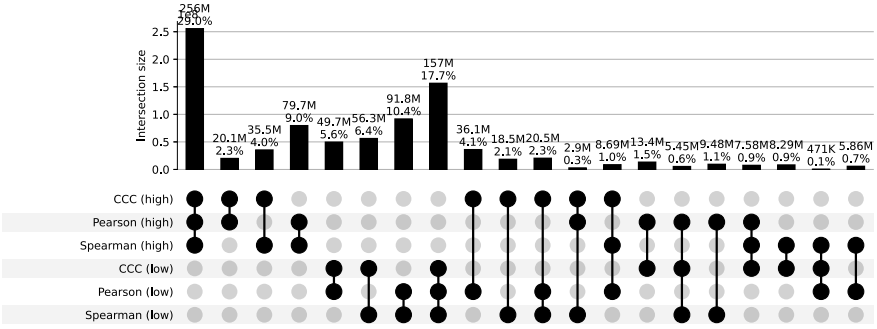
a) Correlation coefficient distributions between gene pairs within GTEx v8 Nerve Tibial



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

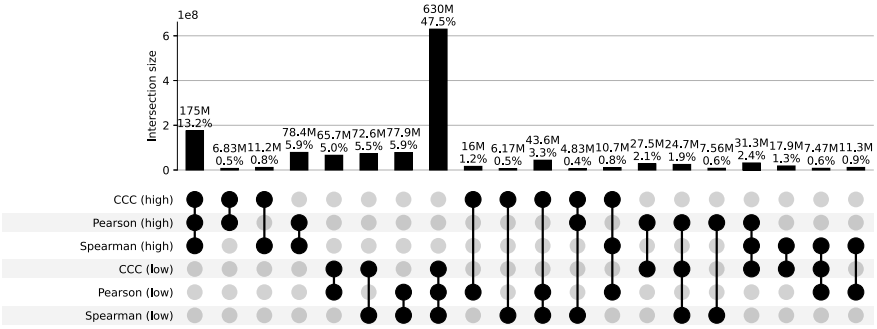
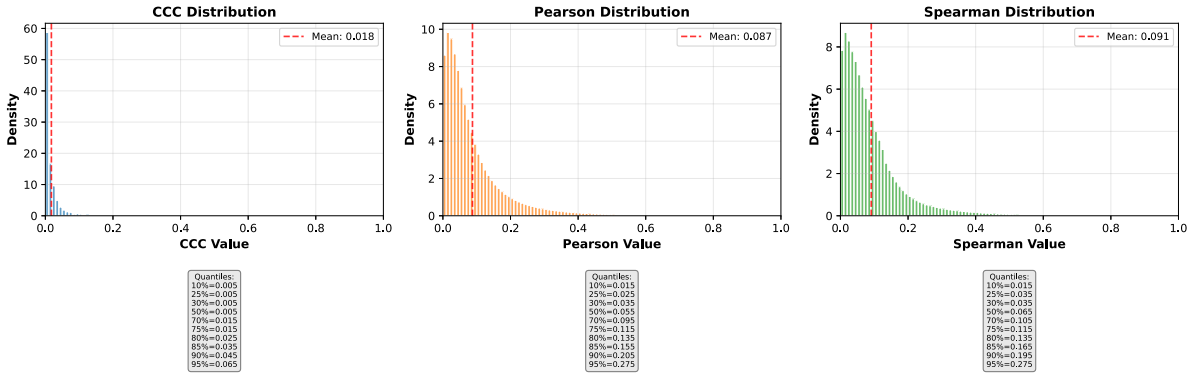


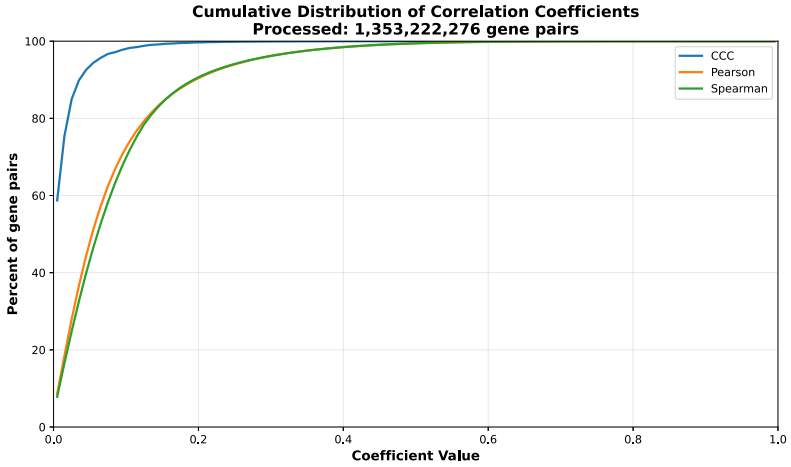
Figure S45: Distribution and UpSet plots for GTEx v8 nerve tibial.

Ovary

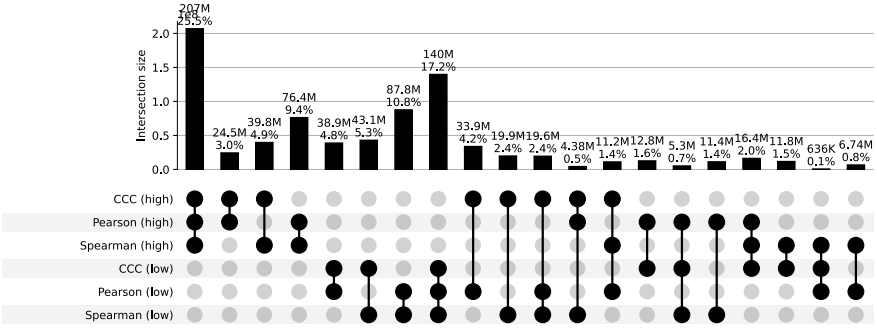
a) Correlation coefficient distributions between gene pairs within GTEx v8 Ovary



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

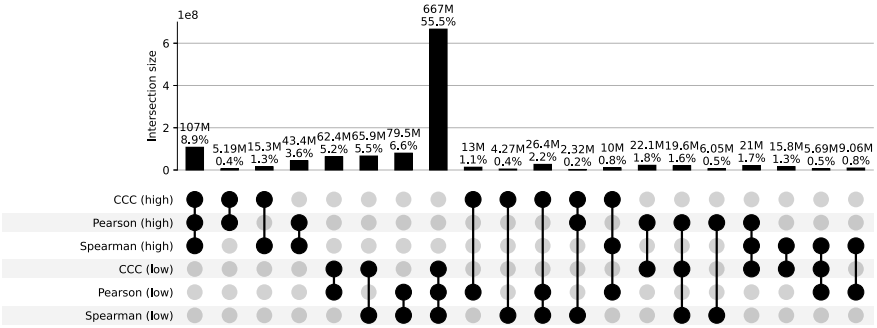
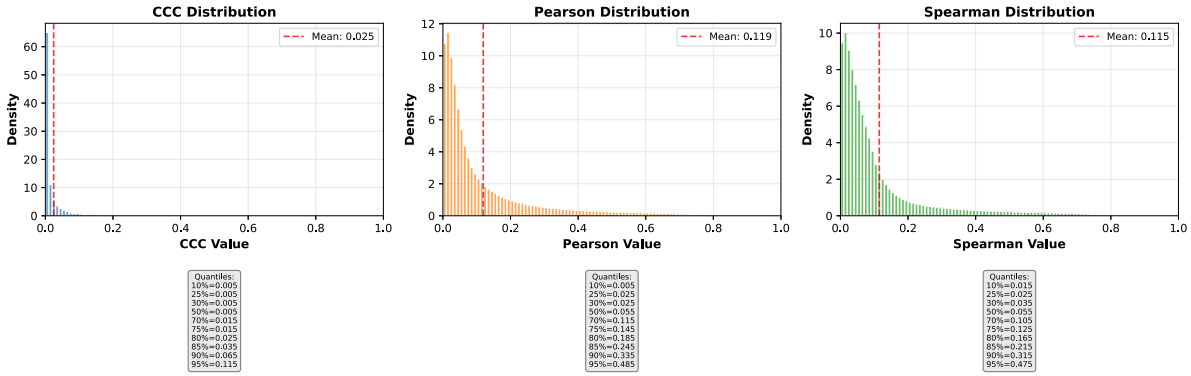


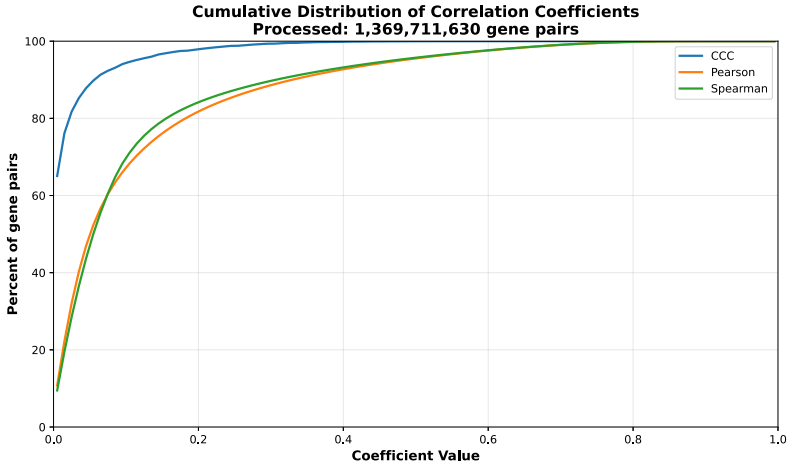
Figure S46: Distribution and UpSet plots for GTEx v8 ovary.

Pancreas

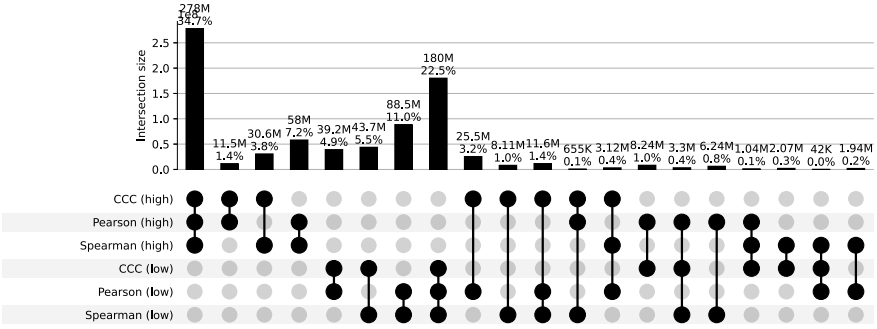
a) Correlation coefficient distributions between gene pairs within GTEx v8 Pancreas



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

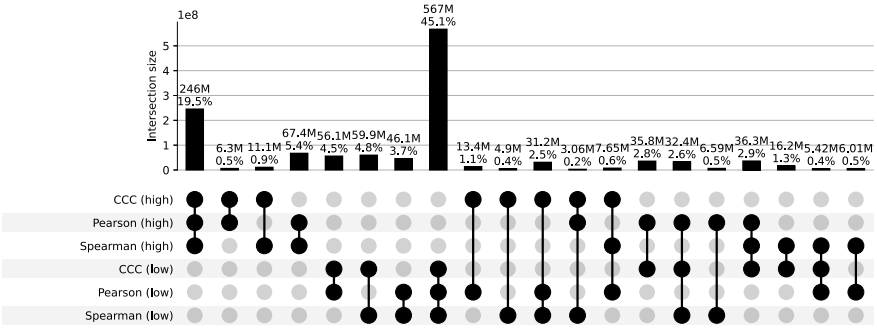
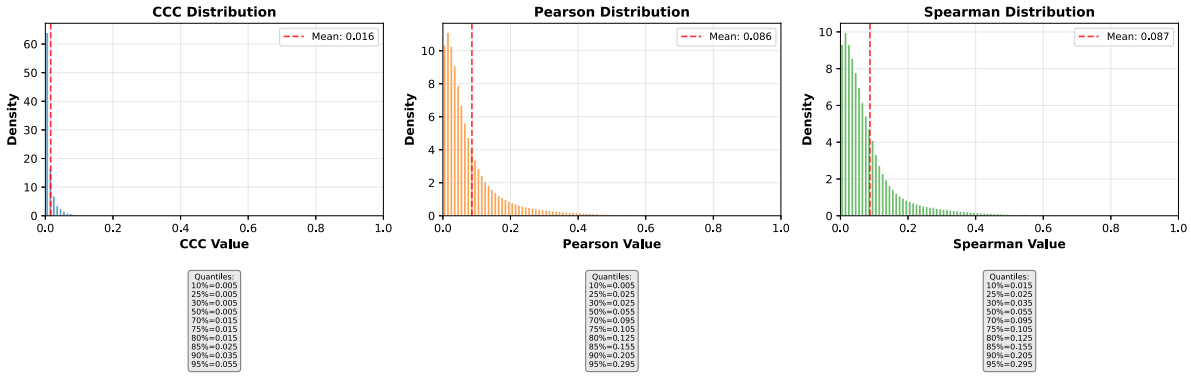


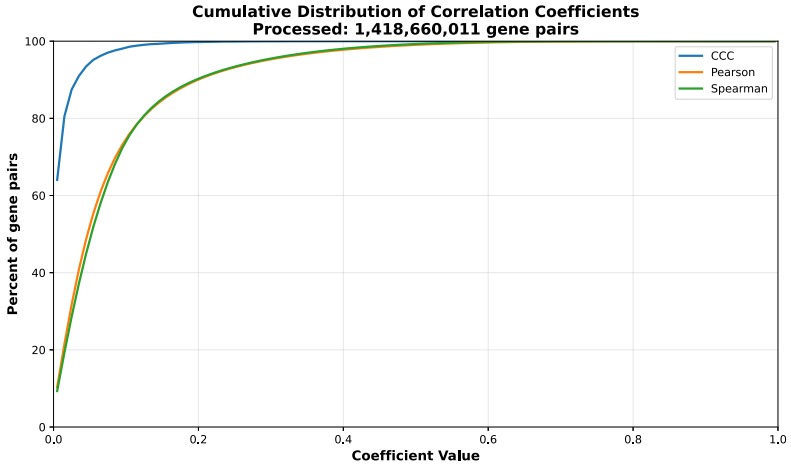
Figure S47: Distribution and UpSet plots for GTEx v8 pancreas.

Pituitary

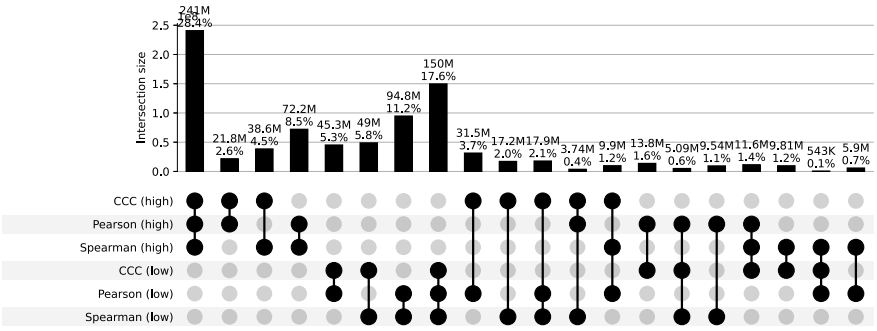
a) Correlation coefficient distributions between gene pairs within GTEx v8 Pituitary



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

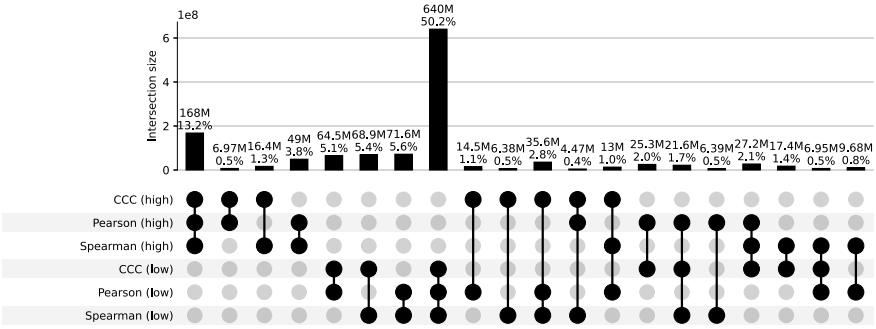
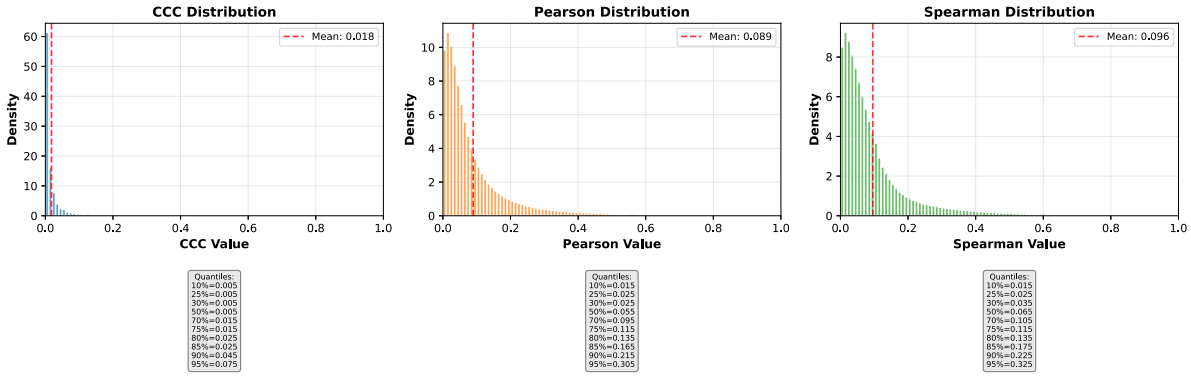


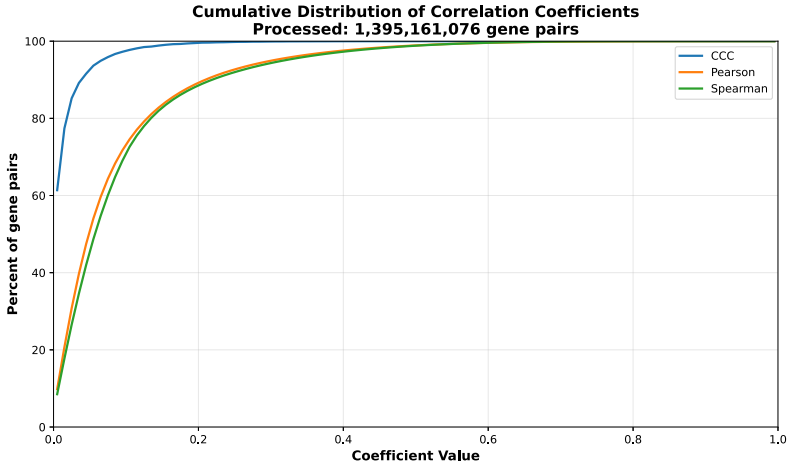
Figure S48: Distribution and UpSet plots for GTEx v8 pituitary.

Prostate

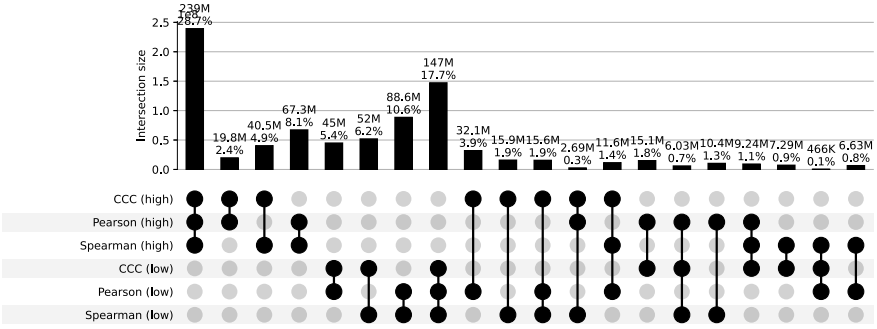
a) Correlation coefficient distributions between gene pairs within GTEx v8 Prostate



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

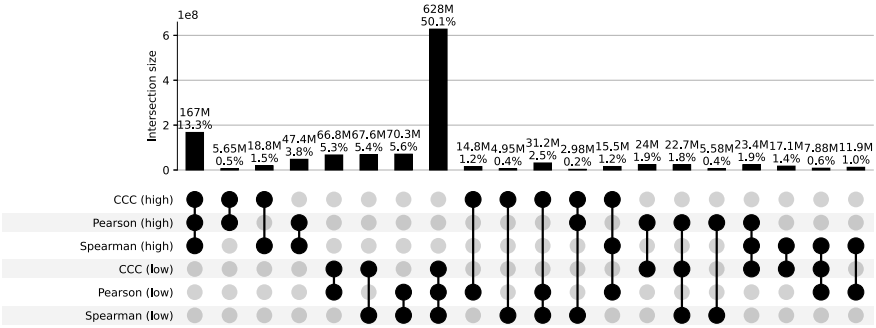
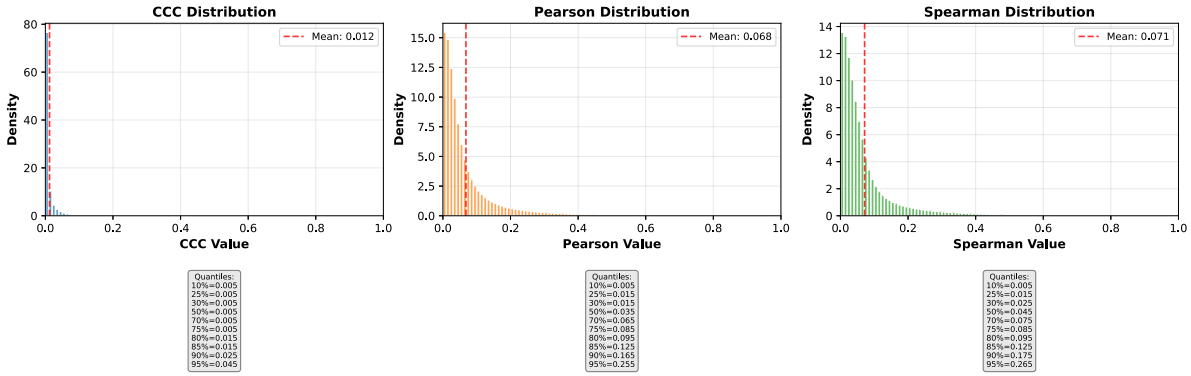


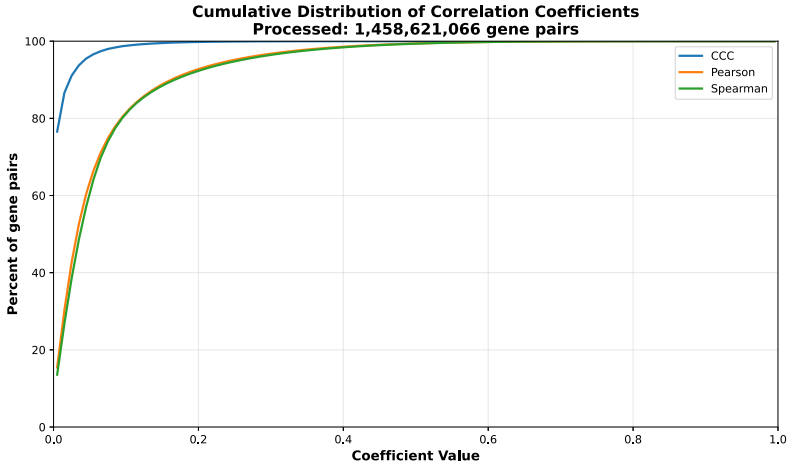
Figure S49: Distribution and UpSet plots for GTEx v8 prostate.

Skin Not Sun Exposed Suprapubic

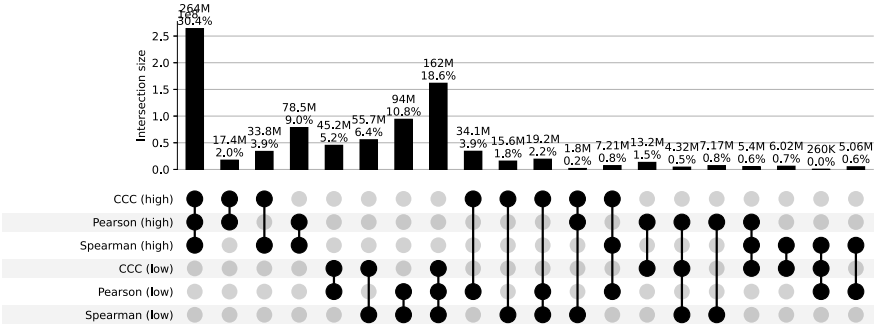
a) Correlation coefficient distributions between gene pairs within GTEx v8 Skin Not Sun Exposed Suprapubic



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

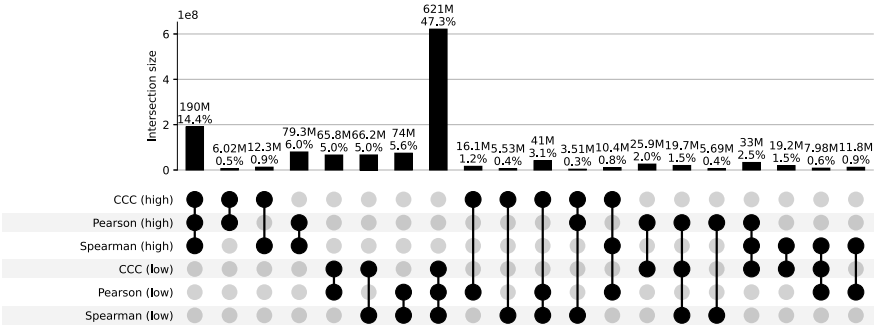
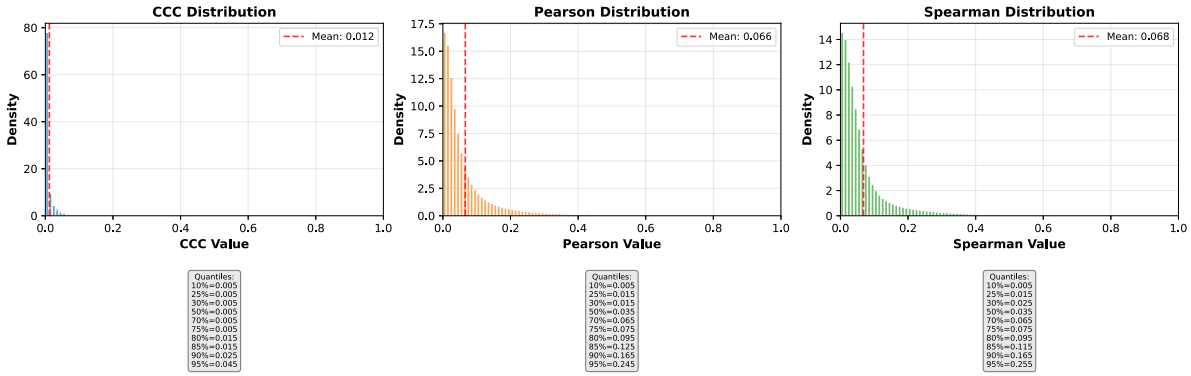


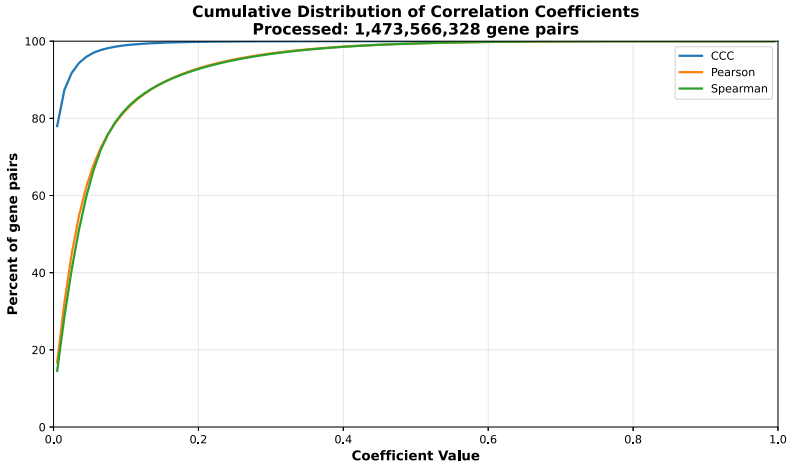
Figure S50: Distribution and UpSet plots for GTEx v8 skin not sun exposed suprapubic.

Skin Sun Exposed Lower Leg

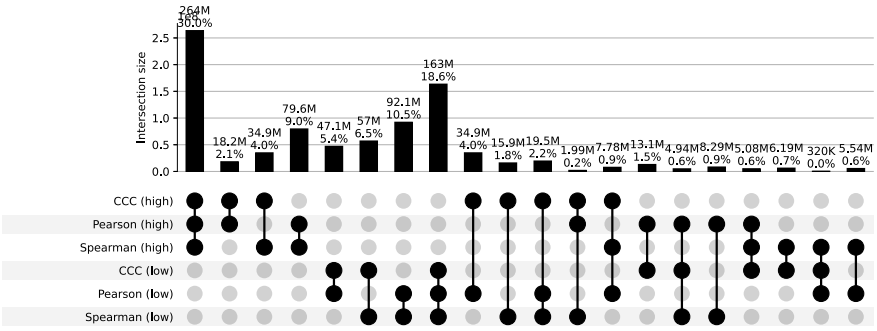
a) Correlation coefficient distributions between gene pairs within GTEx v8 Skin Sun Exposed Lower Leg



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

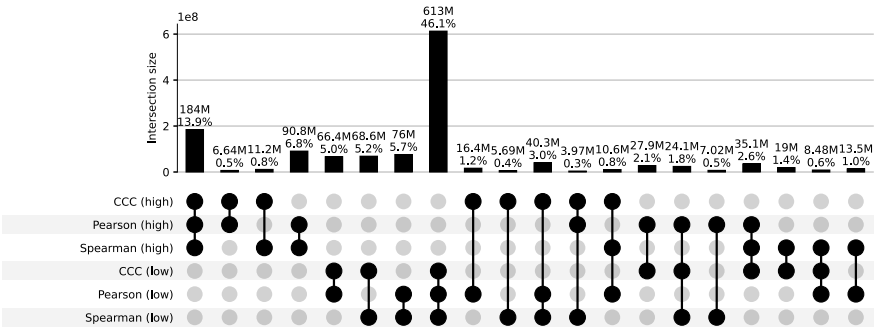
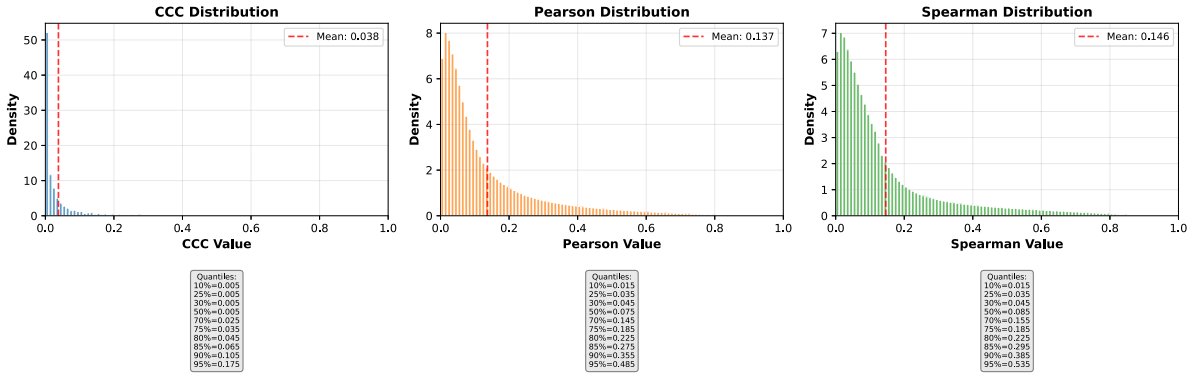


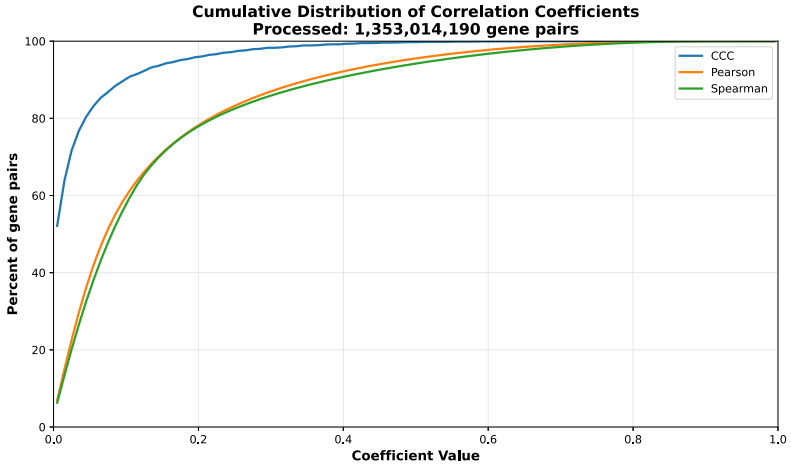
Figure S51: Distribution and UpSet plots for GTEx v8 skin sun exposed lower leg.

Small Intestine Terminal Ileum

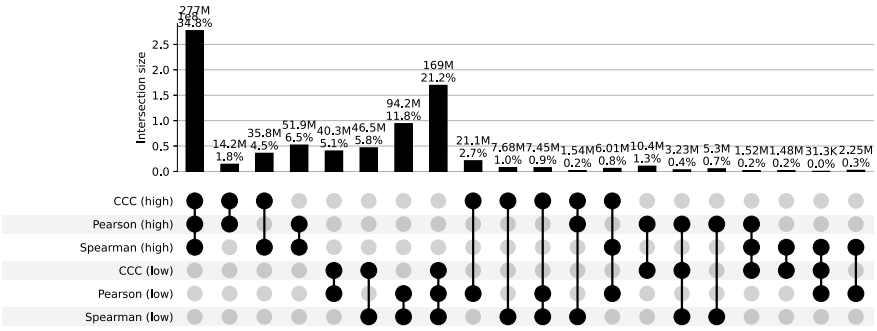
a) Correlation coefficient distributions between gene pairs within GTEx v8 Small Intestine Terminal Ileum



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

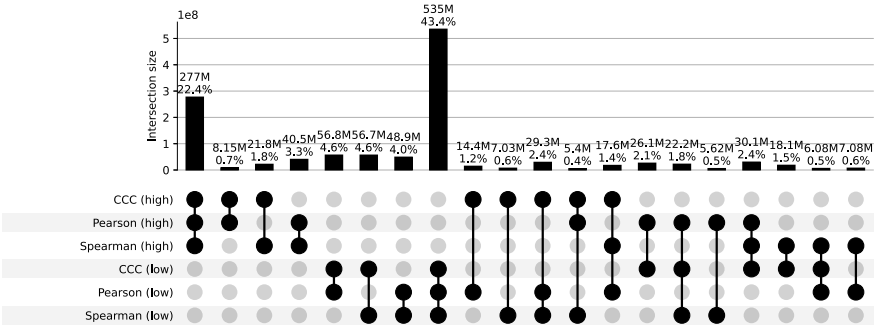
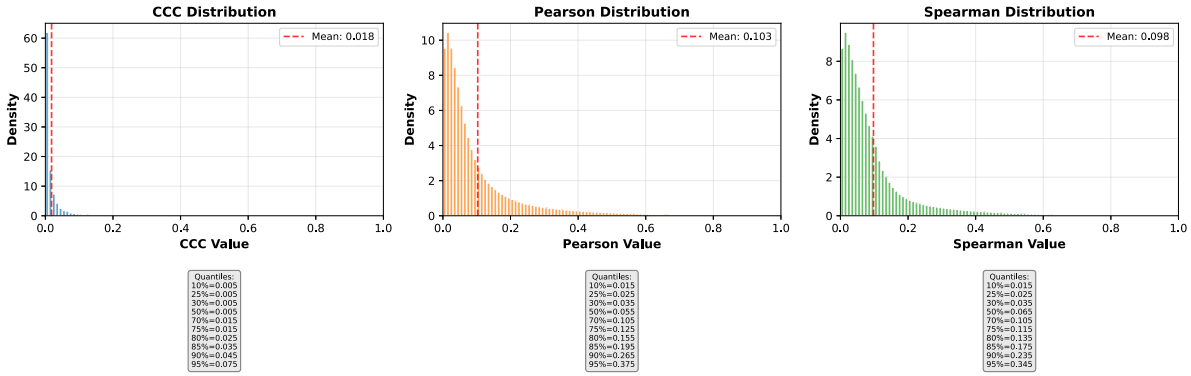


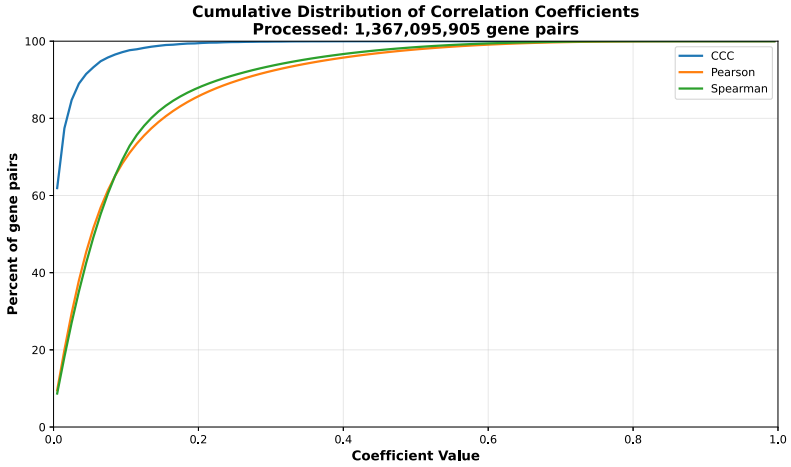
Figure S52: Distribution and UpSet plots for GTEx v8 small intestine terminal ileum.

Spleen

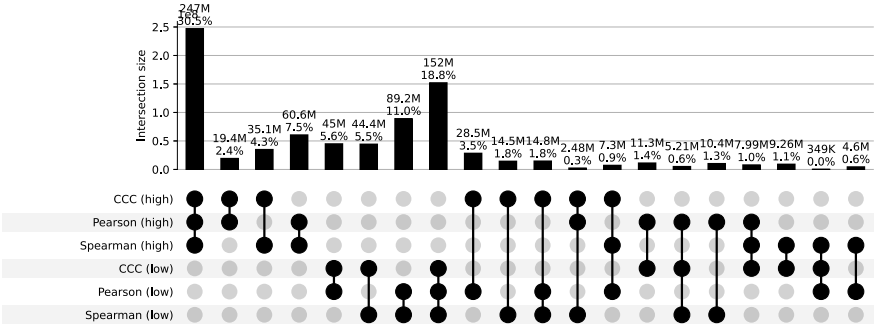
a) Correlation coefficient distributions between gene pairs within GTEx v8 Spleen



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

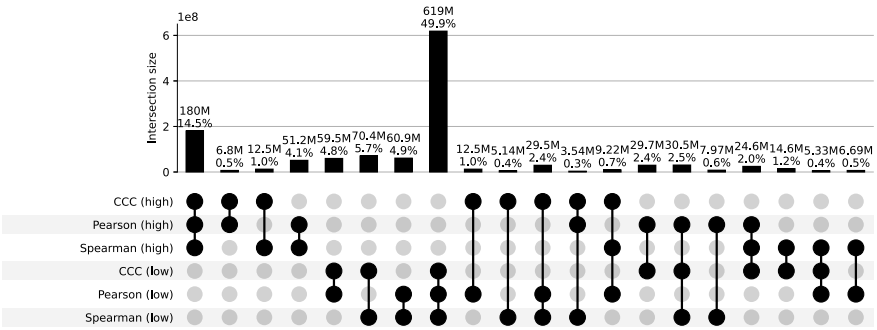
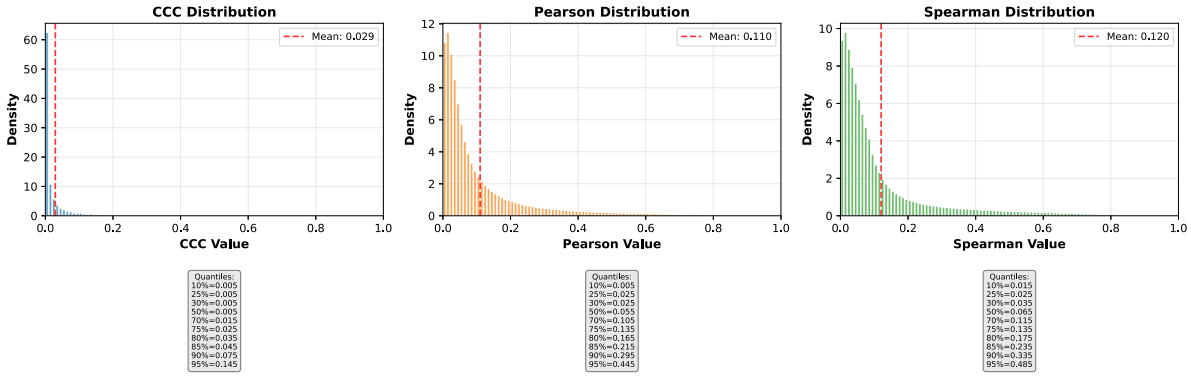


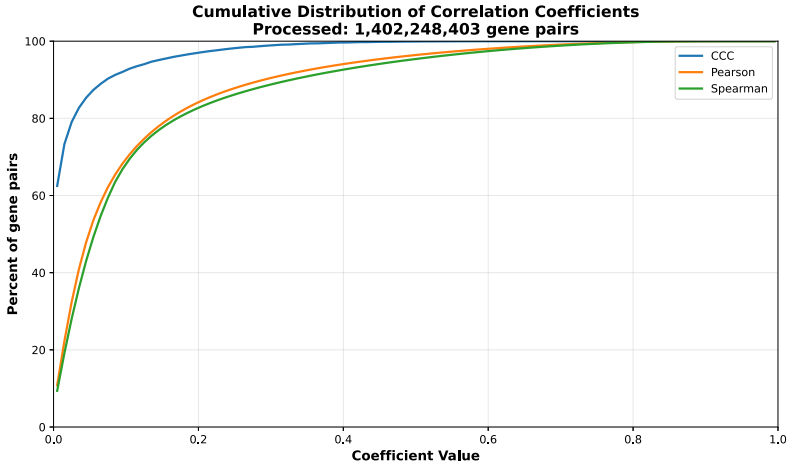
Figure S53: Distribution and UpSet plots for GTEx v8 spleen.

Stomach

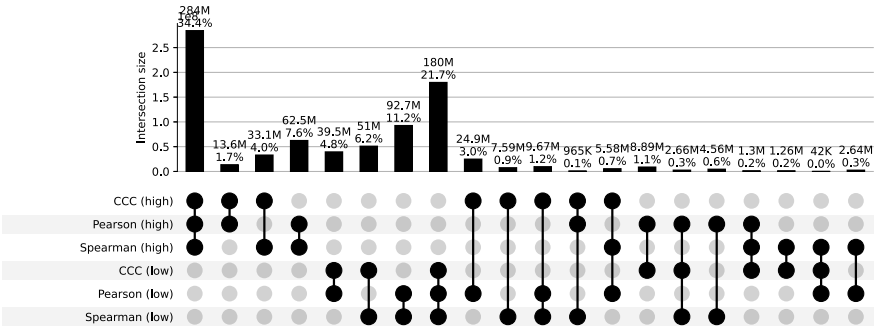
a) Correlation coefficient distributions between gene pairs within GTEx v8 Stomach



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

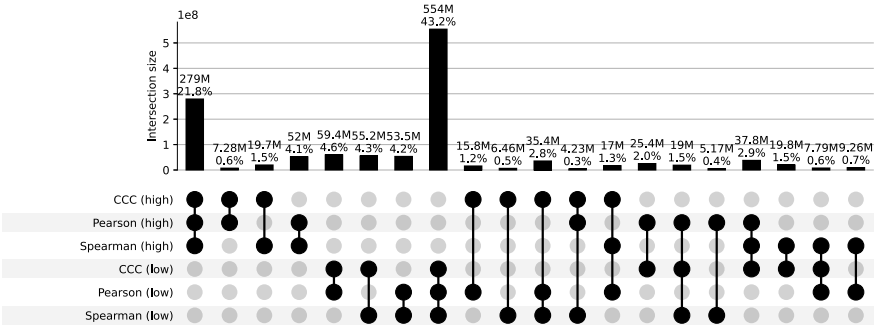
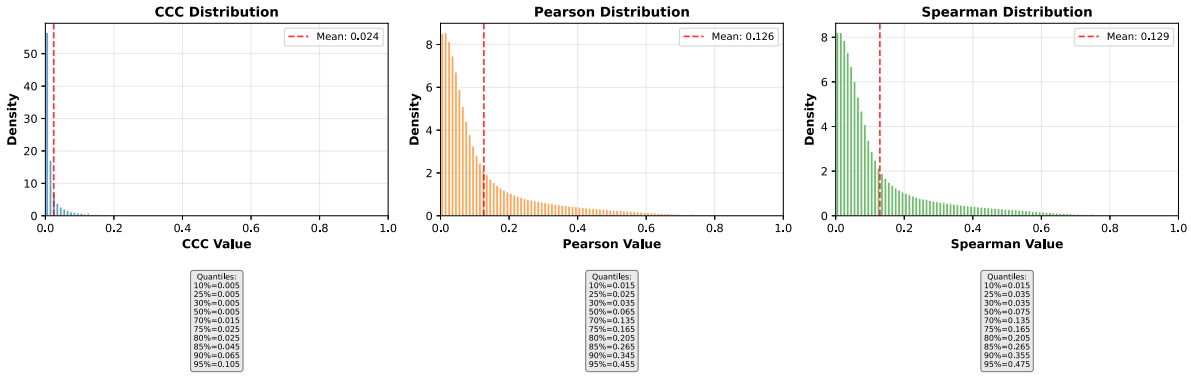


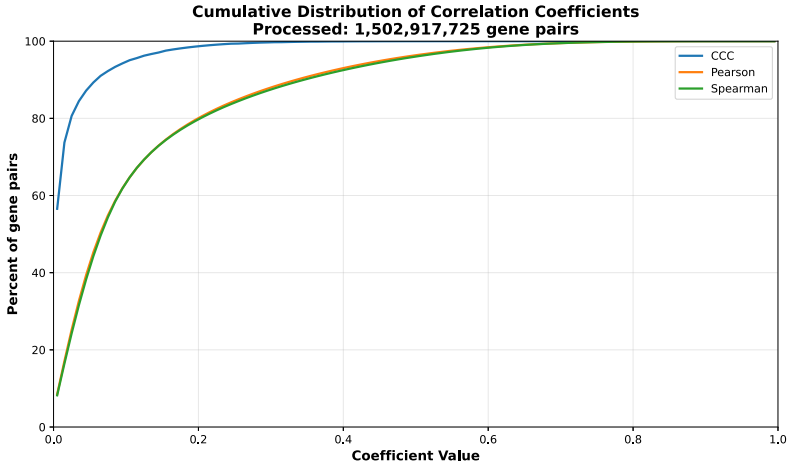
Figure S54: Distribution and UpSet plots for GTEx v8 stomach.

Testis

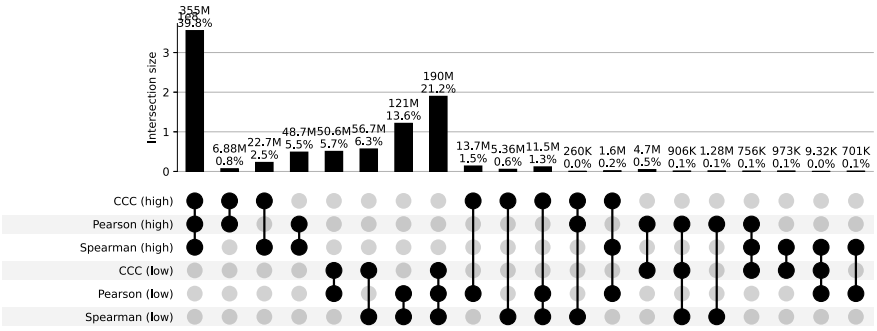
a) Correlation coefficient distributions between gene pairs within GTEx v8 Testis



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

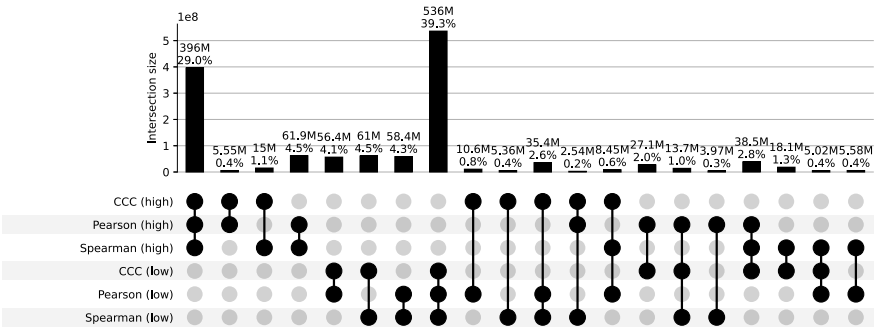
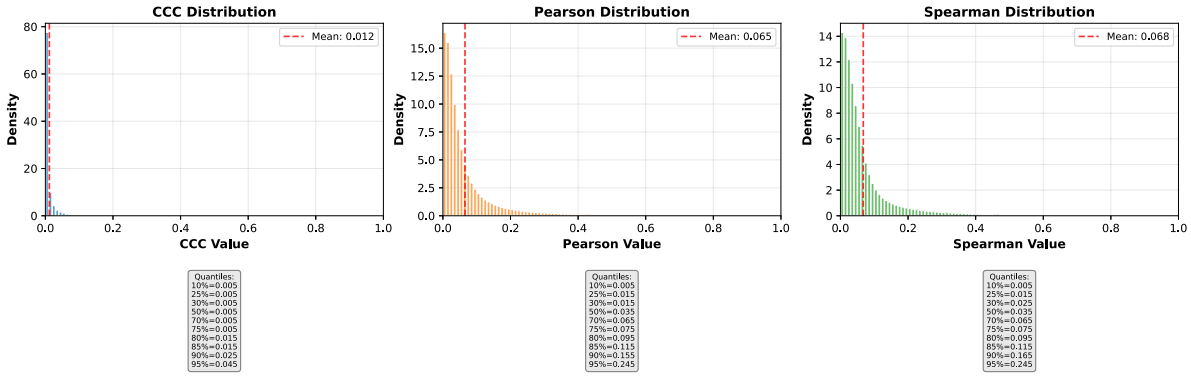


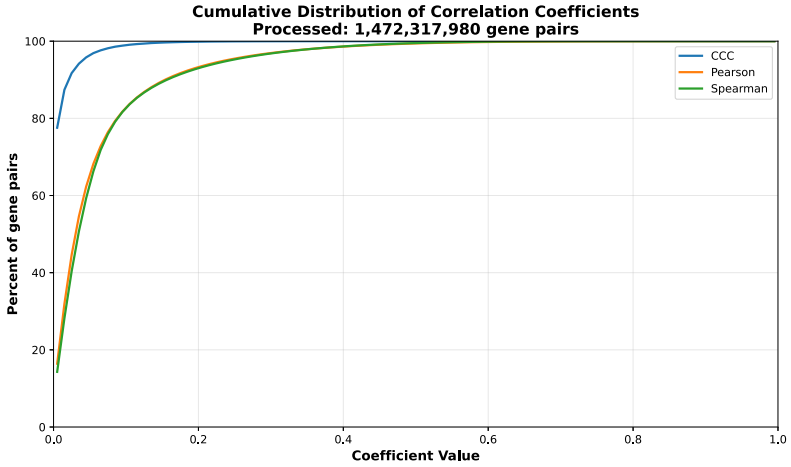
Figure S55: Distribution and UpSet plots for GTEx v8 testis.

Thyroid

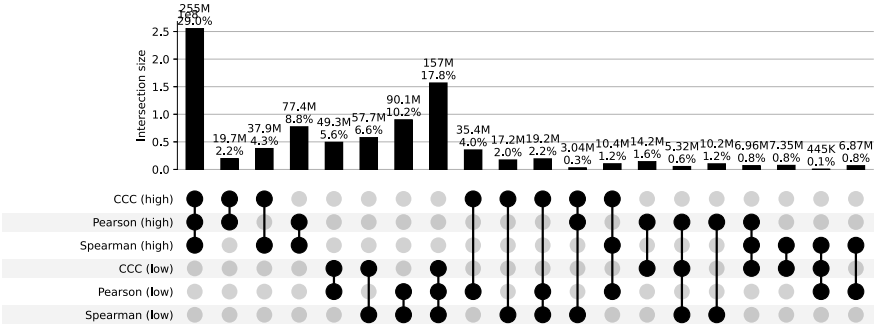
a) Correlation coefficient distributions between gene pairs within GTEx v8 Thyroid



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

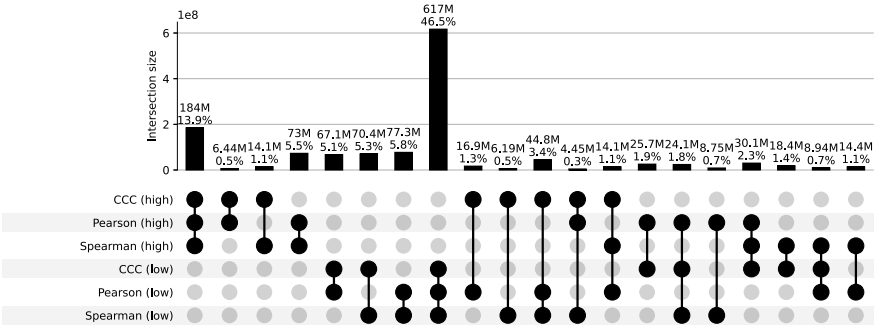
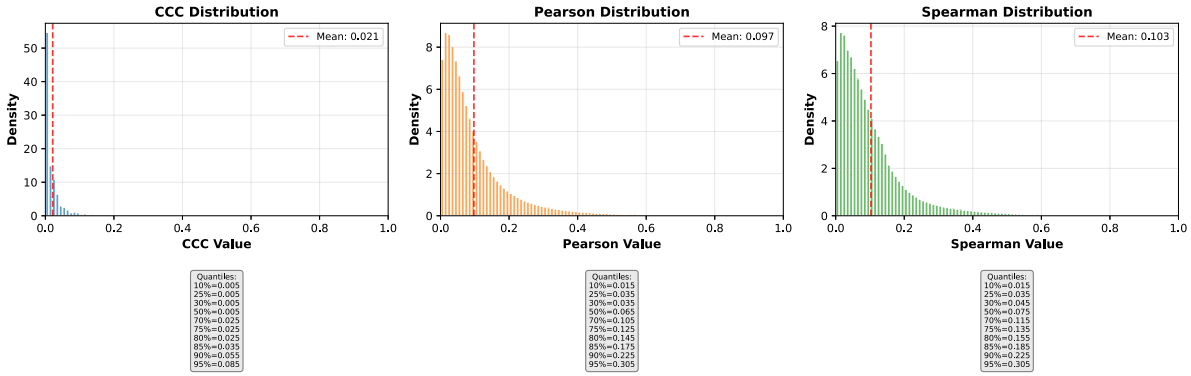


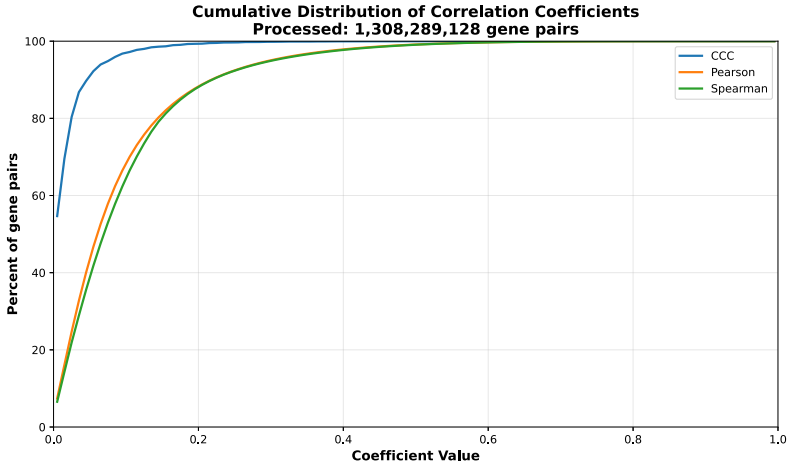
Figure S56: Distribution and UpSet plots for GTEx v8 thyroid.

Uterus

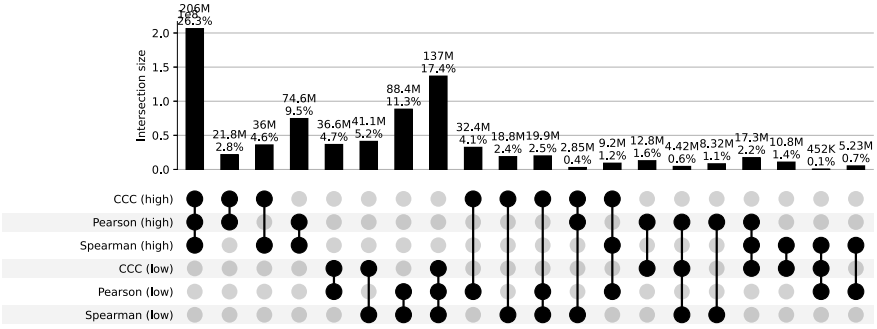
a) Correlation coefficient distributions between gene pairs within GTEx v8 Uterus



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

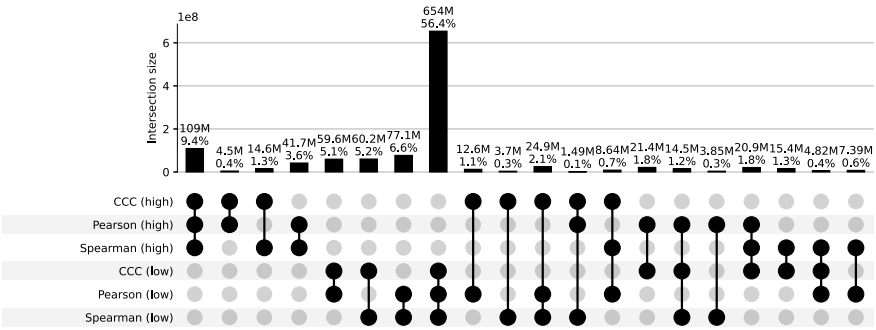
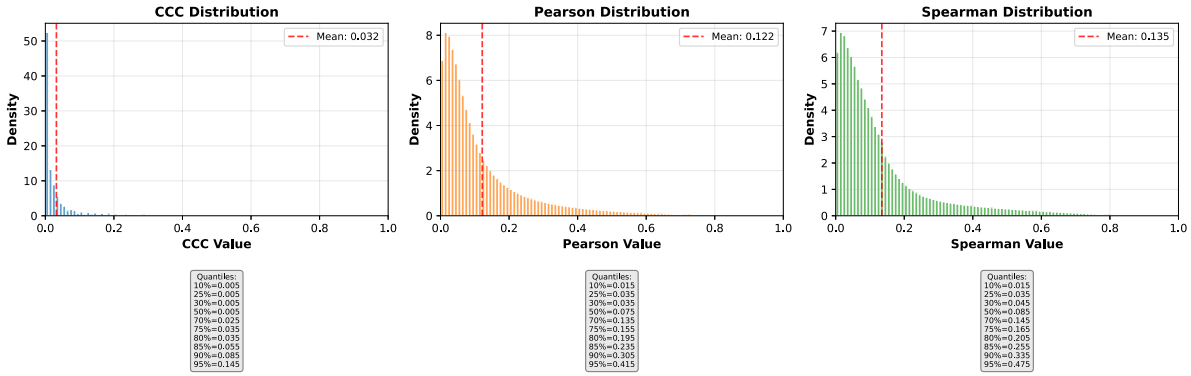


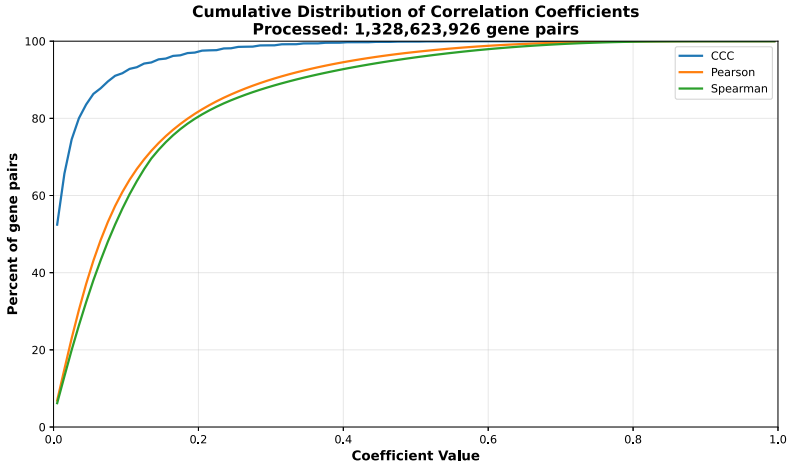
Figure S57: Distribution and UpSet plots for GTEx v8 uterus.

Vagina

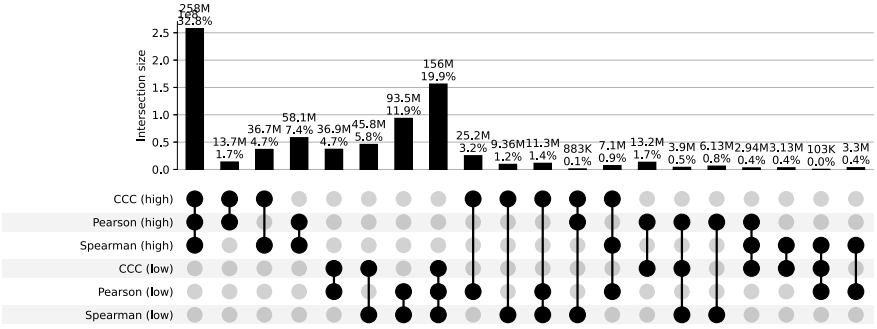
a) Correlation coefficient distributions between gene pairs within GTEx v8 Vagina



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

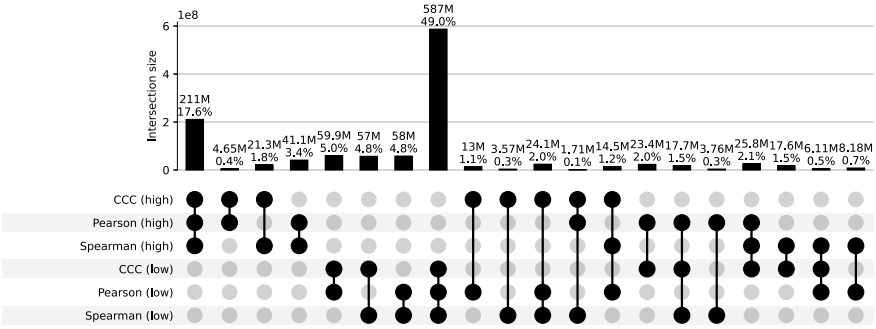
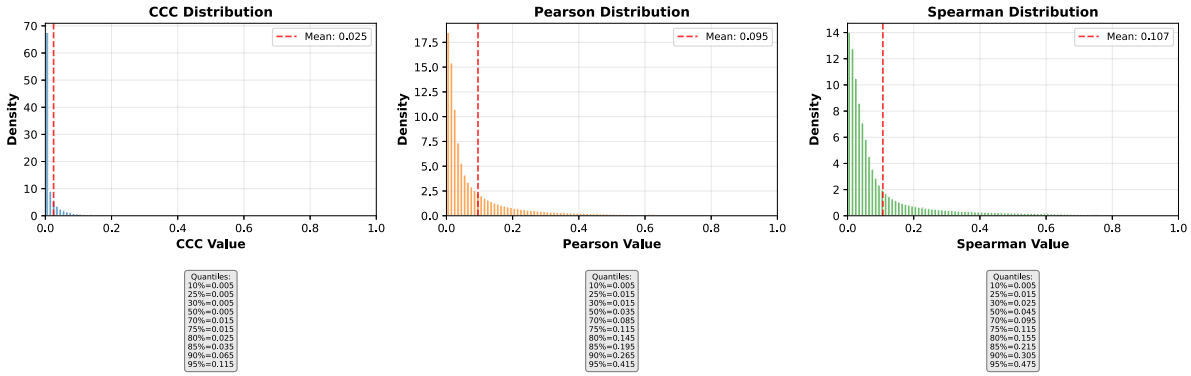


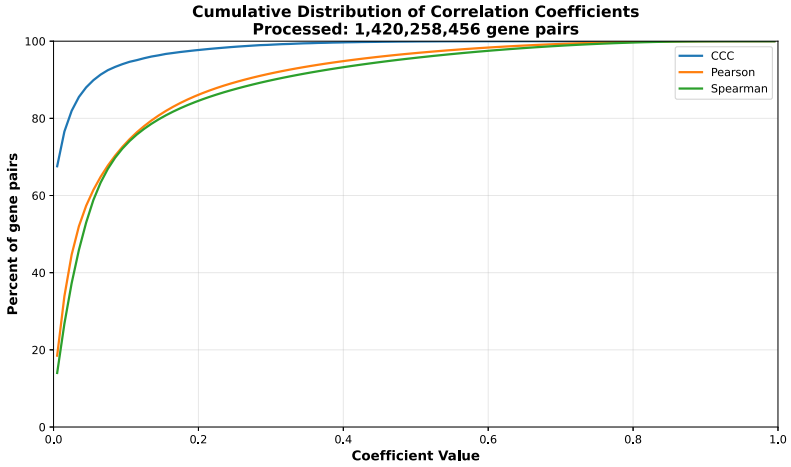
Figure S58: Distribution and UpSet plots for GTEx v8 vagina.

Whole Blood

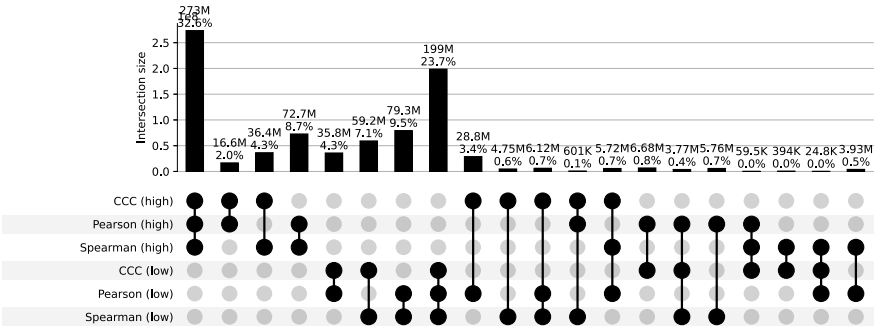
a) Correlation coefficient distributions between gene pairs within GTEx v8 Whole Blood



b) Corresponding cumulative histogram



c) UpSet plot using top and bottom 30% correlations



d) UpSet plot using permutation-based statistical thresholds

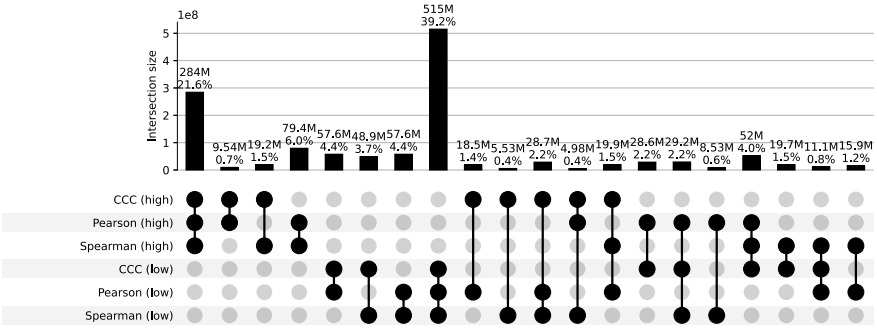


Figure S59: Distribution and UpSet plots for GTEx v8 whole blood.