Age-Dependent Network Rewiring in Mouse Kidney After Spaceflight: A node2vec–Driven Transcriptomic Framework with Cell-Type Deconvolution and Multi-Omics Integration

Your Name
Institution

December 9th, 2025

Abstract

Astronauts have an unusually high rate of kidney stone formation, with 1-year post-flight astronauts experiencing incidence rates 2–7 times that of preflight estimates, as reported by numerous studies on the physiological and morphological effects of spaceflight-induced renal dysfunction. Although it is known that microgravity deactivates the NCC/WNK ion transport hub and remodels the distal convoluted tubule (DCT), the transcriptomic mechanisms behind these changes remain unclear. This study aims to analyze NASA's OSD-771 kidney dataset of 80 bulk RNA-seq samples from young/old cohorts of female mice under flight versus ground control conditions. First, cell-type deconvolution will be performed using single-cell murine kidney atlases to estimate DCT proportions per sample. These estimates and batch covariates will be included as covariates or regressed out before correlation steps. Within each condition (young flight, young control, old flight, old control), weighted co-expression networks will be embedded into 128-dimensional vector spaces using node2vec. Biased random walks will capture topological relationships between genes, and embedding spaces will be aligned across conditions through orthogonal Procrustes analysis using housekeeping and ribosomal gene anchors. For each gene, transcriptomic rewiring is quantified as cosine distance shift Δ between aligned embeddings, with significance assessed using permutation and bootstrap null models controlling sample size and network density. Genes exhibiting high Δ but minimal $\log_2$-fold change are defined as "silent shifters": genes whose network connectivity changes substantially without large expression differences. To interpret these shifts, centrality changes are computed, and gene set enrichment over Δ ranks for curated DCT/NCC–WNK pathway members are also performed, alongside cluster embeddings, all to identify rewired modules subject to Gene Ontology and KEGG enrichment analysis. Finally, cross-condition classifiers trained on embeddings validate biological signal over noise. This framework prioritizes age- and microgravity-responsive modules, identifying silent shifters as molecular drivers of DCT remodeling and therapeutic targets for spaceflight-related renal degeneration.

1 Background & Rationale

Astronauts experience kidney stone formation rates 2–7 times higher than pre-flight estimates within one year post-mission. While microgravity is known to deactivate the NCC/WNK ion transport hub and remodel the distal convoluted tubule (DCT), the transcriptomic mechanisms underlying these changes remain unclear.

The Rodent Research Reference Mission 2 (RRRM-2, NASA OSD-771) provides bulk RNA-seq data from 80 kidney samples of female C57BL/6NTac mice across young/old cohorts under flight versus ground control conditions. This dataset presents a unique opportunity to:

- Characterize baseline ageing signals in kidney transcriptional networks;
- Quantify how spaceflight perturbs age-related network topology;
- Identify "silent shifters"—genes whose network context shifts substantially despite minimal expression changes;
- Prioritize molecular drivers of DCT remodeling as therapeutic targets for spaceflight-related renal dysfunction.

Graph embedding via `node2vec` transforms co-expression networks into dense numeric vectors, enabling systematic comparison of network neighborhoods across experimental conditions while preserving higher-order topological relationships.

2 Specific Aims

Aim 1: Quantify age-dependent transcriptional rewiring in ground controls with cell-type deconvolution and robust statistical frameworks.

Aim 2: Determine whether spaceflight amplifies, dampens, or redirects age-related network rewiring patterns.

Aim 3: Prioritize genes and pathways exhibiting high network rewiring but modest differential expression ("silent shifters"), with particular focus on DCT/NCC–WNK pathway components.

Aim 4: Validate biological relevance through cross-condition predictive modeling, pathway enrichment, and multi-omics triangulation with phosphoproteomic data.

3 Data Overview

Age	Flight Status	Samples	Label
Young	ISS Flight (FLT)	10	YF
Young	Ground Control (GC + VIV)	10	YC
Old	ISS Flight (FLT)	10	OF
Old	Ground Control (GC + VIV)	10	OC

Table 1: Sample distribution across experimental conditions. Each sample measures ∼15,000 expressed genes after filtering.

4 Enhanced Methodological Pipeline

4.1 Phase 0 — Advanced Pre-processing & QC

Expression normalization:

- Variance-stabilizing transformation (DESeq2 `vst`) applied to raw counts.

- Filter genes with TPM < 1 in $> 80\%$ of samples.

Batch effect characterization:

- Explicitly encode batch covariates: library preparation batch, sequencing lane, and shipment cohort.
- Apply Surrogate Variable Analysis (SVA) to detect hidden confounders.
- Quantify variance explained by known technical factors via variance partitioning.

Cell-type deconvolution:

- Deconvolve nephron segment proportions using multiple murine single-cell kidney atlases (e.g., Tabula Muris Senis, Park et al. 2018).
- Estimate DCT, proximal tubule, collecting duct, and glomerular proportions per sample.
- Include segment proportions as covariates in downstream correlation analyses or regress them out before network construction.

Outlier detection:

- Perform PCA and UMAP dimensionality reduction on normalized expression.
- Define exclusion criteria: samples beyond 3 SD from centroid or with extreme first-PC loadings.
- Report variance explained by top 10 PCs and document all exclusions.

4.2 Phase 1 — Robust Graph Construction

For each condition bin (YF, YC, OF, OC):

1. Compute Spearman rank correlation matrix across samples, controlling for deconvolved cell-type proportions.
2. Convert correlations to Fisher-Z transformed values.
3. Construct *signed, weighted* networks retaining both strong positive and negative edges.
4. Test multiple sparsity thresholds (top 0.5%, 1%, 2% of edges) to assess robustness.
5. Store networks as weighted, undirected `networkx` graphs with edge attributes preserved.

Sensitivity analysis: Repeat graph construction across varying correlation thresholds and quantify stability of network density and clustering coefficients.

4.3 Phase 2 — Multi-Seed node2vec Embedding

Hyperparameters:

- Dimensions: 128
- Walk length: 80
- Number of walks: 200
- Return parameter p: 0.25 (local exploration bias)
- In-out parameter q: 4 (breadth-first bias)
- Implementation: PecanPy for computational efficiency

Robustness protocol:

- Run embeddings with 10 different random seeds.
- Average aligned embeddings across seeds to reduce stochastic variation.
- Perform sensitivity analysis over $p \in \{0.25, 0.5, 1, 2\}$ and $q \in \{2, 4, 8\}$.
- Compare `node2vec` embeddings with at least one alternative method (e.g., graph autoencoders, DeepWalk) to validate topological signal stability.

Output: Four sets of 15,000 $\times$ 128 embedding matrices $\mathbf{E}_{\mathrm{YC}}, \mathbf{E}_{\mathrm{YF}}, \mathbf{E}_{\mathrm{OC}}, \mathbf{E}_{\mathrm{OF}}$ for each seed.

4.4 Phase 3 — Enhanced Embedding Alignment

Random walk embeddings exist in arbitrary rotation/reflection spaces. Alignment ensures geometric comparability.

Anchor gene selection:

- Expand beyond 500 low-variance housekeeping genes to include ribosomal proteins, ubiquitin-conjugating enzymes, and core metabolic genes.
- Verify anchor stability: genes must exhibit low coefficient of variation (< 0.2) across all conditions.

Orthogonal Procrustes alignment:

- Align $\mathbf{E}_{\mathrm{YF}}, \mathbf{E}_{\mathrm{OF}}, \mathbf{E}_{\mathrm{OC}}$ to reference space $\mathbf{E}_{\mathrm{YC}}$ using anchor genes.
- Quantify alignment quality: compute residual Frobenius norm and correlation of anchor gene embeddings post-alignment.
- Report alignment error distributions across all genes.

Cross-seed consensus: Average aligned embeddings across 10 seeds to obtain stable consensus embeddings $\mathbf{E}^*_{\mathrm{cond}}$.

4.5 Phase 4 — Rewiring Metrics with Confidence Intervals

For gene g, let $\mathbf{v}_{\text{bin}}(g)$ denote its 128-dimensional consensus embedding vector. Define rewiring scores as cosine distance shifts:

$$\Delta_{\text{age,ctrl}}(g) = 1 - \cos\left(\mathbf{v}_{\text{OC}}(g), \mathbf{v}_{\text{YC}}(g)\right), \tag{1}$$

$$\Delta_{\text{age,flt}}(g) = 1 - \cos\left(\mathbf{v}_{\text{OF}}(g), \mathbf{v}_{\text{YF}}(g)\right), \tag{2}$$

$$\Delta_{\text{flt,young}}(g) = 1 - \cos\left(\mathbf{v}_{\text{YF}}(g), \mathbf{v}_{\text{YC}}(g)\right), \tag{3}$$

$$\Delta_{\text{flt,old}}(g) = 1 - \cos\left(\mathbf{v}_{\text{OF}}(g), \mathbf{v}_{\text{OC}}(g)\right). \tag{4}$$

Interaction term: Quantify age-by-flight interaction as:

$$\Delta_{\text{interaction}}(g) = |\Delta_{\text{age,flt}}(g) - \Delta_{\text{age,ctrl}}(g)|$$

Confidence intervals: Bootstrap sample-level expression matrices 1000 times, recompute networks and embeddings, and derive 95% confidence intervals for each Δ metric.

4.6 Phase 5 — Rigorous Statistical Testing

Permutation framework:

- Perform 2000 permutations (increased from 1000) of age or flight labels within appropriate experimental arms.
- Stratify permutations to preserve sample size balance and network density.
- Rebuild graphs, rerun embeddings with alignment, and recompute Δ for each permutation.
- Adjust null distributions for network density differences across conditions.

Multiple testing correction:

- Apply Benjamini–Hochberg FDR correction for genome-wide hypothesis testing.
- Complement FDR with Westfall–Young family-wise error rate (FWER) control for top candidate genes and modules.
- Report empirical null distributions and statistical power calculations.

Silent shifter definition (pre-registered):

- High rewiring: Δ FDR < 0.1 and top decile of Δ distribution
- Low differential expression: $|\log_2 \text{FC}| < 0.3$ and DE FDR > 0.2

4.7 Phase 6 — Biological Grounding & Pathway Analysis

Pre-registered gene sets:

- Curate DCT/NCC–WNK pathway members: WNK1, WNK4, SPAK (STK39), NCC (SLC12A3), Kir4.1/5.1 (KCNJ10/KCNJ16), ENaC subunits.

- Compile positive control gene sets from spaceflight pan-omics literature: ECM remodeling, lipid metabolism, oxidative stress, calcium signaling.

Module-level analysis:

- Perform k-means clustering on consensus embeddings to identify co-regulated modules.
- Compute module-level Δ scores as median within-module rewiring.
- Test WNK–SPAK, Kir4.1/5.1, SLC12A3, ECM, and lipid pathway modules for enrichment in high-Δ rankings.

Enrichment testing:

- Gene Ontology (GO) and KEGG pathway enrichment on top 5% Δ genes using `gProfiler` and `goatools`.
- Gene Set Enrichment Analysis (GSEA) over ranked Δ distributions for curated DCT/NCC–WNK gene lists.
- Cross-reference findings with published kidney ageing and spaceflight signatures.

Centrality dynamics:

- Compute betweenness and eigenvector centrality changes for high-Δ genes across conditions.
- Identify genes transitioning from peripheral to hub status (or vice versa).

4.8 Phase 7 — Expression vs. Rewiring Integration

Quadrant analysis:

- Plot $|\log_2 \mathrm{FC}|$ versus Δ for all genes across all pairwise comparisons.
- Partition genes into quadrants:
  - Upper-left: High Δ, low DE (silent shifters) — prime mechanistic candidates
  - Upper-right: High Δ, high DE (canonical responders)
  - Lower-left: Low Δ, low DE (stable genes)
  - Lower-right: Low Δ, high DE (expression-driven without network rewiring)
- Prioritize upper-left quadrant genes for mechanistic follow-up.

4.9 Phase 8 — Predictive Validation with Cross-Condition Transfer

Classifier training:

- Train Random Forest classifiers to predict age (young vs. old) using consensus embeddings.
- Training scenarios:
  1. Train on control embeddings ($\mathbf{E}_{\mathrm{YC}}, \mathbf{E}_{\mathrm{OC}}$), test on flight ($\mathbf{E}_{\mathrm{YF}}, \mathbf{E}_{\mathrm{OF}}$).
  2. Train on flight embeddings, test on controls.

- Use nested 5-fold cross-validation within training arms.
- Assess calibration curves to detect overfitting.

Robustness tests:

- Downsample training sets (50%, 75% of samples) to test stability.
- Compare classifier performance with and without deconvolution covariates included.
- Report accuracy, AUROC, and confusion matrices for all scenarios.

Interpretation: High cross-arm classification accuracy indicates embeddings capture biological age signal rather than batch effects or technical artifacts.

5 Novel Extensions for Competitive Advantage

5.1 Causal Priors from Phosphoproteomics

- Integrate kinase–substrate activity scores from flight kidney phosphoproteomics datasets (if available from related RRRM missions).
- Bias embedding interpretation or edge weights around the WNK–SPAK–NCC axis based on observed phosphorylation dynamics.
- Test whether genes with high phospho-activity changes also exhibit elevated Δ scores.

5.2 Multi-Omics Triangulation

- Compare transcriptomic rewiring patterns (Δ scores) with proteomic and phosphoproteomic changes from related spaceflight missions.
- Assess multi-layer consistency: do genes with high transcriptomic Δ also show protein abundance or phosphorylation shifts?
- Prioritize genes exhibiting concordant rewiring across omics layers as high-confidence DCT remodeling drivers.

5.3 Radiation vs. Microgravity Deconvolution

- Where feasible, contrast Δ patterns from RRRM-2 with simulated galactic cosmic radiation (GCR) ground-based datasets.
- Identify microgravity-specific vs. radiation-specific rewiring signatures.
- Partition spaceflight effects into stressor-specific components to refine mechanistic understanding.

5.4 Countermeasure Hypothesis Generation

- Map high-Δ modules to known ion-transport pharmacology: thiazides (NCC inhibitors), amiloride (ENaC blockers), CaSR modulators, bisphosphonates.
- Integrate spaceflight countermeasure literature: hydration protocols, citrate supplementation, potassium-sparing strategies.
- Generate in-silico predictions for countermeasure efficacy constrained by observed network remodeling patterns.
- Propose testable hypotheses for pharmacological interventions to mitigate DCT dysfunction.

6 Expected Outcomes

1. **Ranked atlas:** Comprehensive catalog of genes and modules exhibiting significant age- or flight-specific network rewiring, with statistical confidence bounds.
2. **Spaceflight-age interaction map:** Quantitative evidence whether spaceflight exacerbates, mitigates, or orthogonally modulates age-driven network shifts.
3. **Silent shifter shortlist:** High-priority regulatory genes with strong Δ but weak differential expression, enriched for DCT/NCC–WNK pathway components.
4. **Multi-omics integration:** Cross-validated rewiring signatures triangulated with phosphoproteomics and other spaceflight datasets.
5. **Countermeasure targets:** Pharmacologically tractable modules and pathways for therapeutic intervention.
6. **Publication-ready visualizations:** UMAP of aligned gene embeddings colored by Δ; heatmaps of module-condition Δ; Sankey/hive plots of gained/lost edges; Δ vs. $|\log_2 \mathrm{FC}|$ quadrant plots highlighting candidates.

7 Computational Resources & Timeline

Computational requirements:

- 16-core workstation, 64 GB RAM
- Estimated runtime: 4–6 hours including multi-seed embeddings and 2000 permutations
- Storage: $\sim$50 GB for intermediate embeddings and permutation results

Timeline:

- **Month 1:** Data QC, cell-type deconvolution, batch effect modeling, graph construction across seeds and thresholds.
- **Month 2:** Multi-seed embeddings, Procrustes alignment, Δ computation, permutation testing, bootstrap confidence intervals.

- **Month 3:** Enrichment analyses, module detection, integration with DE results, centrality dynamics, silent shifter prioritization.
- **Month 4:** Predictive validation, multi-omics triangulation, countermeasure mapping, figure generation, manuscript drafting, external expert feedback.

8 Data & Code Availability

- **Raw data:** NASA GeneLab accession OSD-771 (RRRM-2 kidney transcriptome).
- **GitHub repository:** `https://github.com/yourname/rrrm2-kidney-node2vec-enhanced`
- **Reproducibility:** All code, hyperparameters, random seeds, and pre-registered gene sets documented in repository with environment specifications.

9 Starter Code Snippet

```
import pandas as pd, numpy as np, networkx as nx
from pecanpy import pecanpy as pp
from scipy.stats import spearmanr
from sklearn.preprocessing import StandardScaler
from scipy.linalg import orthogonal_procrustes

def build_weighted_graph(mat, covariates=None, top_pct=0.01, signed=True):
    """
␣␣␣␣Build␣weighted␣co-expression␣graph␣with␣optional␣covariate␣adjustment.

␣␣␣␣Parameters:
␣␣␣␣-----------
␣␣␣␣mat␣:␣pd.DataFrame
␣␣␣␣␣␣␣␣Expression␣matrix␣(genes␣x␣samples)
␣␣␣␣covariates␣:␣pd.DataFrame,␣optional
␣␣␣␣␣␣␣␣Covariates␣to␣regress␣out␣(samples␣x␣covariates)
␣␣␣␣top_pct␣:␣float
␣␣␣␣␣␣␣␣Percentage␣of␣top␣edges␣to␣retain
␣␣␣␣signed␣:␣bool
␣␣␣␣␣␣␣␣If␣True,␣preserve␣negative␣correlations␣as␣separate␣edge␣weights
␣␣␣␣"""
    # Residualize expression if covariates provided
    if covariates is not None:
        from sklearn.linear_model import LinearRegression
        lr = LinearRegression()
        residuals = []
        for gene in mat.index:
            lr.fit(covariates, mat.loc[gene])
            residuals.append(mat.loc[gene] - lr.predict(covariates))
        mat = pd.DataFrame(residuals, index=mat.index, columns=mat.columns
           )

    # Compute Spearman correlation
    rho = spearmanr(mat.T).correlation
    np.fill_diagonal(rho, 0)
```

```
    # Fisher-Z transformation
    rho_z = np.arctanh(np.clip(rho, -0.999, 0.999))

    # Threshold edges
    if signed:
        thresh_pos = np.quantile(rho_z[rho_z > 0], 1 - top_pct)
        thresh_neg = np.quantile(rho_z[rho_z < 0], top_pct)
        adj = np.where((rho_z > thresh_pos) | (rho_z < thresh_neg),
                       np.abs(rho_z), 0)
    else:
        thresh = np.quantile(rho_z[rho_z > 0], 1 - top_pct)
        adj = np.where(rho_z > thresh, rho_z, 0)

    # Build weighted graph
    G = nx.from_numpy_array(adj)
    nx.relabel_nodes(G, dict(enumerate(mat.index)), copy=False)
    return G

# Main pipeline
expr = pd.read_csv("vst_expression.csv", index_col=0)  # genes x samples
meta = pd.read_csv("metadata.csv", index_col=0)
deconv = pd.read_csv("cell_type_proportions.csv", index_col=0)

bins = {
    "YC": meta.query("Age == 'Young' & Flight == 'Ctrl'").index,
    "YF": meta.query("Age == 'Young' & Flight == 'Flt'").index,
    "OC": meta.query("Age == 'Old' & Flight == 'Ctrl'").index,
    "OF": meta.query("Age == 'Old' & Flight == 'Flt'").index
}

# Multi-seed embedding
n_seeds = 10
embeddings = {name: [] for name in bins}

for seed in range(n_seeds):
    for name, idx in bins.items():
        covs = deconv.loc[idx]
        G = build_weighted_graph(expr[idx], covariates=covs,
                                 top_pct=0.01, signed=True)

        model = pp.Node2Vec(G, d=128, wl=80, num_walks=200,
                            p=0.25, q=4, workers=8, seed=seed)
        emb = pd.DataFrame(model.fit_transform(), index=G.nodes)
        embeddings[name].append(emb)

# Procrustes alignment (average across seeds)
anchor_genes = pd.read_csv("housekeeping_genes.txt", header=None)[0].
   tolist()

for seed in range(n_seeds):
    ref = embeddings["YC"][seed].loc[anchor_genes]
    for name in ["YF", "OC", "OF"]:
        target = embeddings[name][seed].loc[anchor_genes]
```

```
        R, _ = orthogonal_procrustes(target, ref)
        embeddings[name][seed] = embeddings[name][seed] @ R

# Consensus embeddings
consensus = {name: pd.concat(embs).groupby(level=0).mean()
             for name, embs in embeddings.items()}

# Compute Delta
def compute_delta(emb1, emb2):
    from scipy.spatial.distance import cosine
    genes = emb1.index.intersection(emb2.index)
    return pd.Series({g: cosine(emb1.loc[g], emb2.loc[g])
                      for g in genes})

delta_age_ctrl = compute_delta(consensus["OC"], consensus["YC"])
delta_age_flt = compute_delta(consensus["OF"], consensus["YF"])
delta_flt_young = compute_delta(consensus["YF"], consensus["YC"])
delta_flt_old = compute_delta(consensus["OF"], consensus["OC"])
```

References

1. Grover, A. & Leskovec, J. (2016) *node2vec: Scalable Feature Learning for Networks.* KDD.

2. Hamilton, W. et al. (2017) *Representation Learning on Graphs: Methods and Applications.* IEEE Data Engineering Bulletin.

3. NASA GeneLab Consortium (2024) *RRRM-2 Kidney Transcriptome Dataset.* OSD-771.

4. Park, J. et al. (2018) *Single-cell transcriptomics of the mouse kidney reveals potential cellular targets of kidney disease.* Science.

5. da Silveira, W.A. et al. (2020) *Comprehensive multi-omics analysis reveals mitochondrial stress as a central biological hub for spaceflight impact.* Cell.

6. Storey, J.D. & Tibshirani, R. (2003) *Statistical significance for genomewide studies.* PNAS.

7. Westfall, P.H. & Young, S.S. (1993) *Resampling-Based Multiple Testing.* Wiley.