1 Supplementary Data

AIC versus Kolomogorov-Smirnoff D statistic.

The preference for the NB may be biased due to the fact that different distributions have different numbers of parameters (e.g., $1, 2, 3$ for the Poisson, NB and ZINB respectively). The D statistic provides an alternative approach that is insensitive to model complexity. Here we used a null hypothesis H_0 stating that there is no difference between the cumulative distribution of the observed expression of the gene and the cumulative distribution of the theoretical distribution. For each dataset and each gene (trimmed mean of 1%), we rejected the null hypothesis using an (unadjusted) p-value of 0.01. In 12 out of the 13 datasets, the preference remained the NB distribution (Figure S1A).

NB versus exponential distribution.

For the majority of transcripts, the NB and exponential distributions had the first and second lowest AIC scores with averages of 60.4% and 35.8% respectively (Figure 1A). We asked if the preference for NB was an artifact of how expression was pre-processed. More specifically, we note that the shape of the NB distribution when there is high dispersion θ and a low expected count μ is similar to the shape of the exponential distributions especially when there are extreme transcript counts in the right tail. Therefore, we hypothesize that relatively strong preference for the exponential distribution is due to the presence of outliers. In particular, we asked whether an increase in the trimming of outliers (especially those in the right tail) would decrease the preference for the exponential distribution. This is confirmed in Figure S1B, further strengthening our belief that the fRNA-seq transcript count is distributed according to the NB distribution.

The PREFFECT model

We describe a set of non-linear mappings to compute variational posteriors given the expression matrix and variables (e.g., batch) for adjustment. q_Φ is the resultant estimation of the true posterior distribution p_Θ, where Φ, Θ denote all relevant underlying parameters. The exact parameterization of q_Φ depends on which of the tree models (simple, single-tissue and full) is used (Figure S3). q_Φ is composed of several component conditional VAEs described below.

The input to all models is partitioned into minibatches with row dimension M' where size $M' \leq M$ (Figure S3, top). Count matrices are restricted to only the M' samples of the minibatch with width equal to the number of transcripts N. Optionally, (biological and technical) adjustment variables that describe the samples can also be passed. Such variables can either be categorical (with total width of their encoding denoted by k' or continuous (with total width denoted by k''). We use K to denote all such adjustment variables. Adjacency networks for the single tissue and full models are also partitioned into $M' \times M'$ adjacency matrices. The relevant components of VAEs to handle adjacencies can also include adjustment variables K.

Library sizes are computed and fit using a log-Normal distribution. Across all models, the observed sample library sizes L can be allowed to vary during model fitting. Figure S3A depicts the neural network underlying the encoder $q_\Phi^{L,(\tau)}(Z_L^{(\tau)}|L^{(\tau)}, K^{(\tau)})$ that approximates the true posterior distribution $p_\Theta(Z^{(\tau)}|L^{(\tau)}, K^{(\tau)})$. The encoder for the simple model is a three-layer neural network that outputs a latent space with final dimension r (Figure S3B). The encoder $q_\Phi(Z|X, K)$ is an estimate of the true posterior $p_\theta(Z|x, K)$ where θ is the set of all parameters of the neural network. When multiple tissues are present (full model), we write $q_\Phi^{(\tau)}(Z^{(\tau)}|X^{(\tau)}, K^{(\tau)})$.

The single tissue layer ($\mathcal{T} = 1$) and full models ($\mathcal{T} > 1$) encode the input sample-sample adjacency networks into separate Gaussian VAEs. The encoder $q_{A,\Phi}(Z_A|X, A, K)$ for the networks and associated adjustment variables estimate the true posterior distribution $p_{A,\theta}$. The underlying neural network for each tissue τ begins with a GAT version 2 convolutional layer [1] with a tune-able parameter h denoting the number of heads in the attention mechanism (Figure S3C). For each expression matrix X, each edge e_{ij} in the adjacency matrix S associated with X, and for each attention head $1 \leq \eta \leq h$. we compute:

$$e_{ij}^\eta = ELU(a^\eta([W^\eta X_{i,\cdot}||W^\eta X_{j,\cdot}])).$$

Here $||$ denotes the concatenation operation, W^η is the weight matrix, and a^η is a vector with dimension $\mathbb{R}^{1\times 2r}$. These values are then normalized across all neighbors $\mathcal{N}_S^{(\tau)}(i)$ using a *softmax* function:

$$\alpha_{ij}^\eta = softmax_j(e_{ij}^\eta) = \frac{exp(e_{ij}^\eta)}{\sum_{k\in\mathcal{N}_A(i)} exp(e_{ik}^\eta)}.$$

These normalized attention coefficients are then used to compute a linear combination of their features, and the h heads are concatenated as follows:

$$X_i' = \Big\|_{\eta=1}^{h} \sum_{j \in \mathcal{N}_S} \alpha_{ij}^{\eta} W^{\eta} X_{j,\cdot}.$$

The width of the embedding is therefore rh.

The reparameterization step generates two sets of variates for the latent space of each tissue:

$$\begin{aligned} Z_{i,j}^{(\tau)} &\sim \mu_{i,j}^{(\tau)} + \epsilon_{i,j}^{(\tau)} \cdot (e^{\frac{1}{2} logvar_{i,j}^{(\tau)}}), \\ \epsilon_{i,j}^{(\tau)} &\sim \mathcal{N}(0, \sigma_{i,j}^{(\tau)}), \end{aligned}$$

across samples $1 \leq i \leq M'$ and for each dimension of the latent space $1 \leq j \leq r$. A separate decoder produces an $M' \times M'$ estimation of $\hat{A}^{(\tau)}$ for each adjacency tissue (only single-tissue and full models).

The decoder in the simple model consists of two linear layers that produce estimates of the parameters for the NB distribution including c_m, which is a probability density function where $c_{m,g}$ is the fraction of counts for the transcript g in each a sample m, and the inverse dispersion θ. θ can be estimated for each gene-sample combination (denoted $\theta_{m,g}$, or for each gene within each batch (θ_g). Let l_m be the library size for sample m (optionally, this may be optimized during the learning step). The NB distribution is expressed as a hierarchical Gamma-Poisson model:

$$\begin{aligned} \mu_{m,g} &= l_m \cdot c_{m,g}, \\ \omega_{m,g} &\sim \Gamma(shape = \theta_g, rate = \theta_g / \mu_{m,g}) \\ c_{m,g} &\sim Poisson(\omega_{m,g}), \end{aligned}$$

for each sample m and transcript g. See also Figure 3C. If ZINB is preferred, the dropout probability π is also estimated. The final reconstructed expression is defined as follows:

$$\hat{c}_{mg}^{(\tau)} = \begin{cases} c_{m,g} & \text{if } \mathrm{Bern}(\pi_{m,g}) \\ 0 & \text{otherwise.} \end{cases}$$

The other models have a set of latent spaces $Z_1, \ldots, Z_{\mathcal{T}}$ which are concatenated during the decoding process (Figure S3E):

$$Z' = \Big\|_{\tau=1}^{\mathcal{T}} Z_A^{(\tau)} = Z_A^{(1)} \; Z_A^{(2)} \; \cdots \; Z_A^{(\mathcal{T})}.$$

This extends the neighborhood expression learned via the attention mechanisms from a single tissue to the neighborhoods from all tissues (Figure 7).

Loss

We follow standard practice [2] for computing loss during training via the evidence lower bound, which combines the expected log-likelihoods of the reconstruction terms and the reverse KL divergence to regularize the posterior distributions of the latent variables. We assume that the priors follow a multidimensional normal with the identity covariance matrix. That is, they follow a $N(0, I_r)$ distribution where I_r is the identity matrix over the r dimensions of the latent space.

For the full model, this is formulated as:

$$\begin{aligned}
\mathcal{L} &= \mathbb{E}_{q_\Phi}\left[\log p_\Theta(X|Z, Z_A, Z_L, K) + \log p_\Theta(A|Z, Z_A, Z_L, K) + \log p_\Theta(L|Z_L, K)\right] && (1)\\
&\quad - D_{KL}(q_\Phi(Z|X, A, K) \,||\, p_\Theta(Z_A))\\
&\quad - D_{KL}(q_\Phi(Z_A|X, A, K) \,||\, p_\Theta(Z_A))\\
&\quad - D_{KL}(q_\Phi(Z_L|L, K) \,||\, p_\Theta(Z_L))\\
&= \mathbb{E}_{q_\Phi} \sum_{\tau=1}^{\mathcal{T}} \log p_\Theta(X^{(\mathcal{T})}|Z^{(\tau)}, Z_A^{(\tau)}, Z_L^{(\tau)} K^{(\tau)}) + && (2)\\
&\quad + \mathbb{E}_{q_\Phi} \sum_{\tau=1}^{\mathcal{T}} \log p_\Theta(A^{(\mathcal{T})}|Z^{(\tau)}, Z_A^{(\tau)}, Z_L^{(\tau)} K^{(\tau)}) + && (3)\\
&\quad + \mathbb{E}_{q_\Phi} \sum_{\tau=1}^{\mathcal{T}} \log p_\Theta(L^{(\mathcal{T})}|Z^{(\tau)}, Z_L^{(\tau)} K^{(\tau)}) + && (4)\\
&\quad + \sum_{\tau=1}^{\mathcal{T}} -D_{KL}(q_\Phi(Z^{(\mathcal{T})}|X^{(\mathcal{T})}, A^{(\mathcal{T})}, K^{(\tau)}) \,||\, p_\Theta(Z^{(\mathcal{T})})) && (5)\\
&\quad + \sum_{\tau=1}^{\mathcal{T}} -D_{KL}(q_\Phi(Z_A^{(\mathcal{T})}|X^{(\mathcal{T})}, A^{(\mathcal{T})}, K^{(\tau)}) \,||\, p_\Theta(Z_A^{(\mathcal{T})})) && (6)\\
&\quad + \sum_{\tau=1}^{\mathcal{T}} -D_{KL}(q_\Phi(Z_L^{(\mathcal{T})}|X^{(\mathcal{T})}, bK^{(\tau)}) \,||\, p_\Theta^L(Z_L^{(\mathcal{T})})) && (a)
\end{aligned}$$

Equation 1 is computed as either $L_{ZINB} = -log(ZINB(\hat{c}|\pi, \omega, \theta))$, where $ZINB(X|\pi, \omega, \theta) = \pi\delta_0 + (1 - \pi)NB(X|\omega, \theta)$ and δ_0 is the Dirac function, or $L_{NB} = -log(NB(c|\omega, \theta))$, where NB is the negative binomial function. Equations 2 and 3 are computed using binary cross entropy. D_{KL} refers to the KL divergence. Equation (a) is the KL divergence if the library sizes are allowed to vary. There are also options within the PREFFECT code base to bias the learner to a disentangled latent space with respect to the adjustment variables K following the approach from Chen et al. [3].

Optimizations

PREFFECT incorporates several strategies to more reliably train good, general models. This includes early stopping (halting training when validation performance stops improvement, preventing the model from overfitting), input masking (randomly zeroing a fraction of expression values so the cVAE learns to impute and denoise true counts), loss delays (introducing specific losses after a certain number of epochs), and gradient clipping (to prevent exploding gradients from destabilizing model weights).

Hyper-parameter search

For each of the three PREFFECT model types, we performed a hyper-parameter search across pseudo-synthetic datasets and the fRNA-seq compendium. In all cases, an $8:2:2$ ratio for the train, validation and test datasets were formed. The hyper-parameters are enumerated in Table S3. Since fRNA-seq datasets vary in size and other important dimensions, end-users may need to explore alternative settings primarily for the central parameters including learning rate, epochs, batch size and latent dimension (r). In Table S3, hyper-parameters with names ending in "weights" correspond to the β parameters of the conditional VAE. Parameter values may also depend on user choices (e.g. when a ZINB is preferred over a NB distribution for modeling transcript counts). Other parameters are model dependent and may not be required if the 'simple' PREFFECT model is used (e.g. parameters associated with edge reconstruction weights). PREFFECT default values were chosen to minimize validation loss across all experiments where there was evidence of model convergence.

Computational resources

PREFFECT was developed on a high performance compute cluster with access to NVIDIA A100 GPUs. Note however that although these high-end GPUs are required for the hyper-parameter search conducted over ~ 25 variables

(Table S3), a typical single execution of PREFFECT (for a given set of parameter values) for a dataset with $M = 1000$ samples and $N = 1000$ transcripts with 500 epochs of training required only ~ 7 minutes using a single Intel Xeon Platinum 8480 CPU (without multi-threading). Training of a 'full' model using a 2 tissue dataset with $M = 2000$ samples and $N = 2000$ genes (500 epochs) required ~ 19 with a GPU and ~ 37 minutes with a CPU. Memory usage was always below 2 GB. The GPU did not provide a significant benefit over a CPU when training a simple model.

References

[1] S. Brody et al. "How Attentive are Graph Attention Networks?" *arXiv 2105.14491* (2022).

[2] C. Doersch. "Tutorial on Variational Autoencoders". *arXiv 1606.05908* (2021).

[3] R. T. Q. Chen et al. *Isolating Sources of Disentanglement in Variational Autoencoders*. 2018.

A.

Subsection	Data	Treatment of Data	Analysis Goals	Train / Validation	Findings
Exploration of the statistical properties of fRNA-seq data					
1. (Fig. 1, S1)	fRNA-seq compendium	right-trimming	descriptive statistics	Not applicable	Enriched for zero & extreme large counts
2. (Fig. 1)	fRNA-seq compendium	right-trimming	fit to distributions	Not applicable	NB is best
Development of the generative model					
3. (Fig. 2) (Fig. S2, S3)	Pseudo-synthetic data	PREFFECT simple model	Test quality of fit to NB, ZINB	train / validation / test	Good fit almost everywhere
4. (Fig. 3)	Pseudo-synthetic data	PREFFECT simple model	Batch correction with conditional VAE	train / validation / test	Corrects three types of effects
5. (Fig. 4)	Pseudo-synthetic data	PREFFECT single layer	adjacency matrix	train / validation / test	Improves quality of clusters
6. (Fig. 5)	Pseudo-synthetic data	PREFFECT full model	matched tissues	train / validation / test	Improvement of clusters per tissue
Application to current fRNA-seq datasets					
7. (Fig. 6)	fRNA-seq compendium	PREFFECT full model	vs. traditional bulk RNA-seq methodology	train / validation / test	Improvement in all datasets

B.

Name	Tissues	Adjacency networks	Description
simple	only one target	no	Generative model can use sample metadata.
single	only one target	yes	GAT layer exploits sample–sample correlations.
full	multiple	yes	Borrows information across tissues.

Table S1: Overview of PREFFECT model development. A. Subsections 1 and 2 perform descriptive statistics on a series of fRNA-seq datasets in order to determine which distribution best fits the data. Then in subsections 3 and 4 the *simple* generative model, which uses only transcript count data for a single tissue, is designed and evaluated using pseudo-synthetic data. We show that the PREFFECT VAEs can be conditioned to correct for batch effects. Subsection 5 extends the simple model to also include an adjacency matrix which can encode different, heterogeneous types of information about each sample and the relationships between samples. We show that this sample-sample adjacency information improves the sample clustering in the sense that samples of the same intrinsic breast cancer subtype more tightly co-cluster. Finally, in subsection 6 and 7, the *full* model is presented. This model extends the single layer model to allow multiple matched tissue samples (e.g. tumor and matched normal, or tumor and matched stromal samples). Again, we show that the resultant sample clusters improve by exploiting information contained in different tissues. B. Summary of the main features of each of the three PREFFECT models.

Dataset	**Date profiled**	**Date of samples**	**Sample description**	**RNA Extraction**	**Sequencing**	**Number samples**	**Ave. size of library**	**Frac. zeros**	**Ave. count per transcript**	**Ave. NB θ $\pm std$**	**NB θ 95% C.I.**
GSE120795	2018	2012-18	Various normal tissues	RecoverAll$^{\beta}$	HiSeq 3000$^{\alpha}$	147	8.84 M	0.55	179 ± 2937	0.41 ± 0.44	$0.014 - 1.46$
GSE146889	2020	NA	Colo./Endometrial	TruSeq RNA	HiSeq 2500	91	39.1 M	0.46	799 ± 7623	1.17 ± 1.44	$0.022 - 4.68$
			cancer/normal tissue	Exome$^{\alpha}$		91	41.1 M	0.46	863 ± 9285	1.14 ± 1.35	$0.021 - 4.41$
GSE167977	2015-16	2004-12	IDC	AllPrep$^{\gamma}$	HiSeq 2500	528	14.1 M	0.52	325 ± 4087	1.12 ± 2.04	$0.005 - 3.73$
GSE181466	NA	2010-15	IDC (Triple Negative)	RNeasy$^{\gamma}$	HiSeq 2500	97	20.4 M	0.10	1005 ± 6732	2.02 ± 1.50	$0.044 - 5.11$
GSE209998	2022	NA	IDC & metastatic (FFPE)	AllPrep$^{\gamma}$	HiSeq 2500	36	30.0 M	0.28	575 ± 9486	1.72 ± 1.96	$0.070 - 6.81$
			matched fresh frozen			93	70.0 M	0.31	1323 ± 43713	1.01 ± 1.47	$0.040 - 3.75$
GSE47462	2014	2002-10	IDC, EN, and DCIS	RecoverAll$^{\beta}$	GA IIx$^{\delta}$	48	3.56 M	0.26	172 ± 711	1.63 ± 2.83	$0.057 - 4.07$
			Normal tissue			24	3.22 M	0.22	165 ± 666	2.28 ± 2.92	$0.129 - 5.64$
TMBC	NA	2012-20	Metastatic breast cancer	AllPrep$^{\gamma}$	HiSeq 2500	210	43.2 M	0.52	861 ± 20356	0.63 ± 0.99	$0.010 - 2.47$
Sunnybrook	2022	1994-	DCIS	AllPrep$^{\gamma}$	NovaSeq$^{\alpha}$	220	50.2 M	0.53	874 ± 28434	0.48 ± 0.85	$0.007 - 2.93$
		2003	Stromal tissue			65	58.5 M	0.52	1109 ± 21167	0.56 ± 1.20	$0.013 - 3.38$
			Normal tissue			101	55.0 M	0.52	981 ± 20584	0.62 ± 1.54	$0.017 - 3.81$

Table S2: The fRNA-seq compendium. IDC: Invasive Ductal Carcinoma of the breast; EN: Early Neoplasia of the breast; DCIS: Ductal Carcinoma *in situ*; NA: Not Available; α: Illumina Inc. β: Ambion/Life Technologies Inc. γ: Qiagen Inc. δ: GA IIx: Genome Analyzer IIx, Illumina Inc.

Hyper-parameter	**Search Space**	**Default Value**
Model type	{simple, single-layer, full tissue}	choice
Adjacency matrix	{Yes, No}	Yes
Learning rate	{1e-5, 1e-4, 5e-4, 1e-3, 1e-2}	1×10^{-3}
Epochs	{100, 500, 1000, 2000}	1000[a]
Batch size	{50, 100, 200, 500}	100
Bottleneck dimension (r')	{64, 128, 256}	128
Latent dimension (r)	{10, 16, 20, 32, 64}	20
Number of attention heads	{4, 8, 12}	8
Dropout rate	0.0, 0.1, 0.2, 0.5	0.1
Number of layers	{2, 3, 4}	3
Distribution type	{NB, ZINB, Poisson}[b]	NB
Activation function	{ReLU, ELU, Tanh}	ReLU
Layer initialization	{Xavier, Norm, Const, Orthogonal}	Orthogonal
Weight decay (L2)	0.0, 1e-5, 5e-4, 1e-4, 1e-3	5×10^{-4}
Alpha (LeakyReLU slope)	0.0, 0.1, 0.2, 0.5	0.2
Random count masking	0.0, 0.1, 0.2, 0.5	0.1
Random adj. edge masking	0.0, 0.1, 0.2, 0.5	0.1
Gradient clip max norm	{0.5, 1.0, 5.0, 10.0}	10.0
Learn library size	Off, On	Off
Batch normalization	Off, On	On
Expression KL divergence weight	{0.01, 0.1, 1.0, 10.0}	0.1
Lib. size KL divergence weights	{0.01, 0.1, 1.0, 10.0}	0.1
Expression reconstruction weights	{0.1, 1.0, 10.0, 100.0, 1000.0}	100.0
Edge recon. weights	{0.1, 1.0, 10.0, 100.0, 1000.0}	1000.0
Library size recon. weights	{0.1, 1.0, 10.0, 100.0, 1000.0}	1
Number of input samples	{50, 100, 500, 1000, 10000}	10000
Number of input transcripts	{50, 100, 500, 1000, 10000}	100[c]

Table S3: Hyper-parameter search space and selected values. Measuring the loss in the validation set across the compendium and pseudo-synthetic data, we determined default parameter values which should serve as reasonable starting points for end-users seeking to train a PREFFECT model with a new dataset. a. This is set by default to a very liberal 1000, however PREFFECT has early stopping functionality to avoid over-fitting. b. Since it did not perform well, the Poisson distribution option was not included in the final release of PREFFECT. c. In general, the number of epochs and latent dimensions increase with the number of input transcripts.

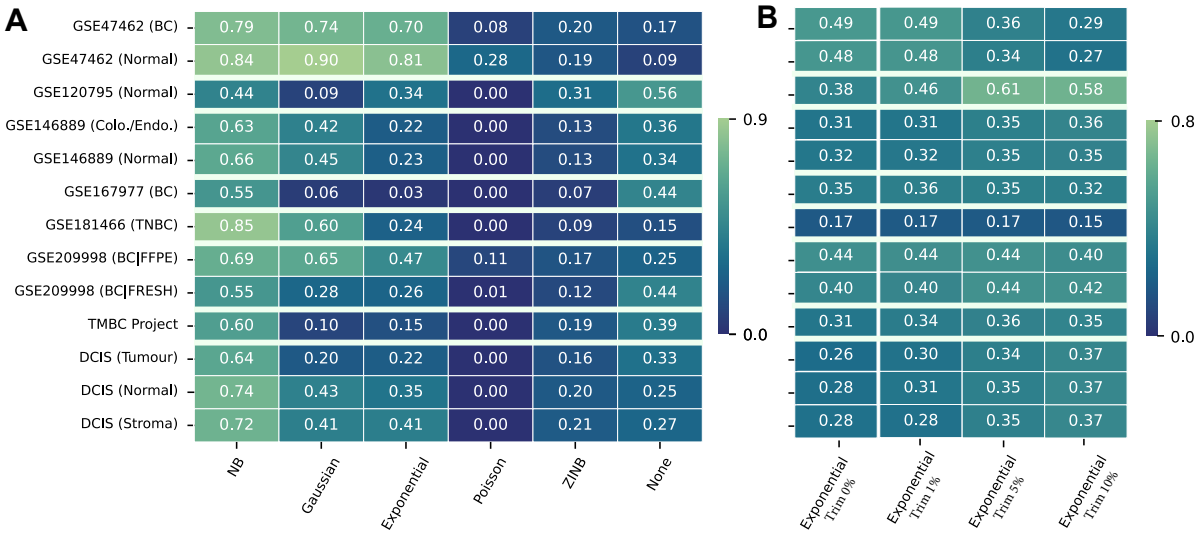

Figure S1: **A**. The fraction of transcripts where the null hypothesis is accepted according to the KS two-samples test. Column labelled None indicates the fraction of transcripts where the null hypothesis was not accepted for any of the distributions. **B.** The fraction of transcripts that preferred an exponential distribution (via the AIC measure) across different thresholds for mean trimming. Increased trimming reduces the number of outliers; the removal of right tail outliers reduces the preference for the exponential distribution.

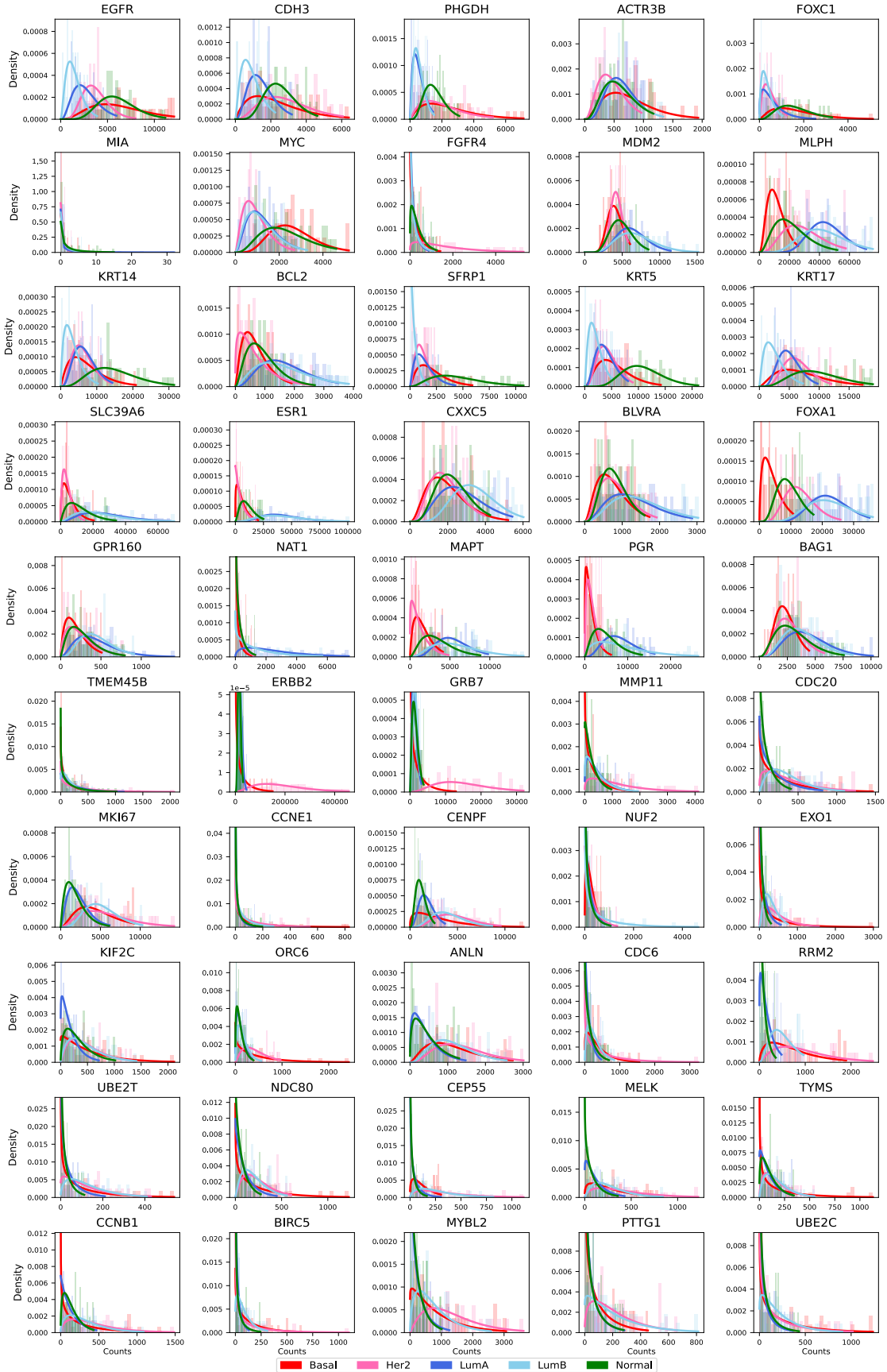

Figure S2: Histograms of counts for each PAM50 transcript in the Sunnybrook DCIS tumor cohort. Samples were first normalized by median library size, and subjected to outlier trimming. Samples were stratified by the PAM50 classifier before an NB distribution was fit to each transcript. The y-axis was truncated to improve readability.

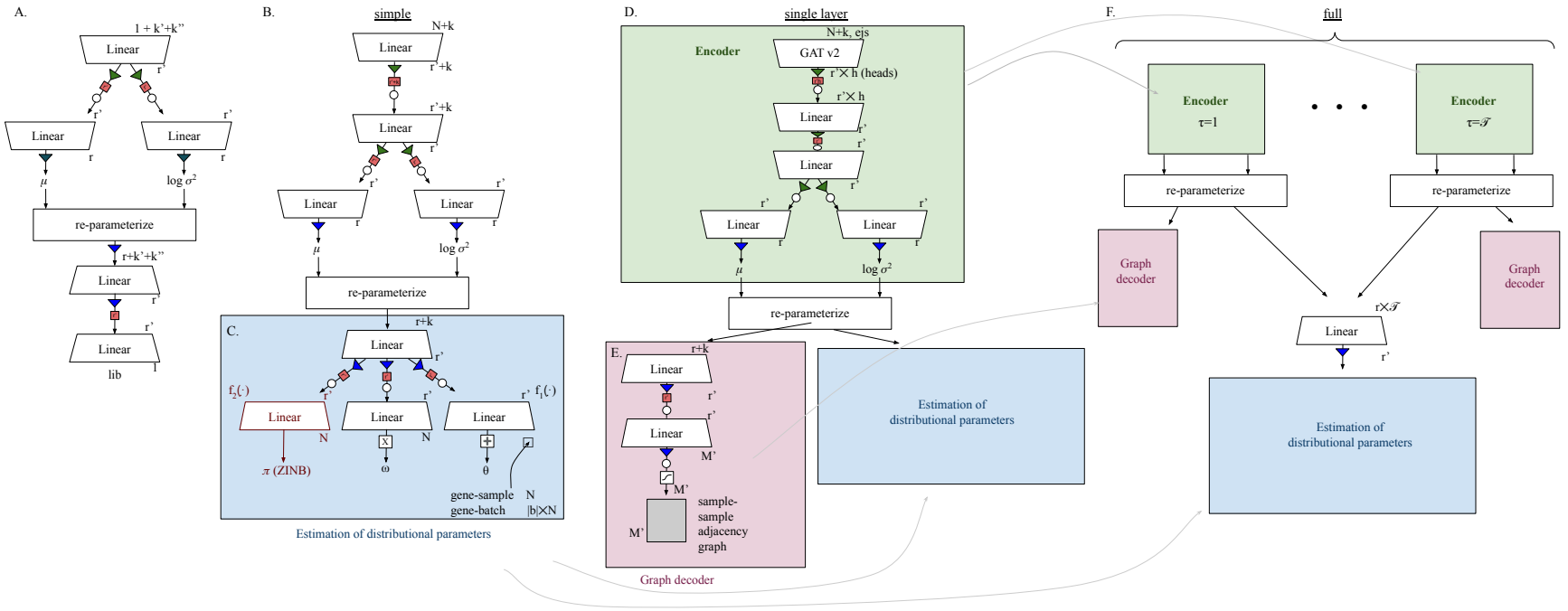

Figure S3: A schematic describing the neural networks. **A.** This VAE encodes sample library size. Here k' and k'' are the number of categorical and continuous correction variables respectively, and r is the dimension of the latent space. **B.** The simple network with a single tissue and without an adjacency network. The empty circle indicates a dropout node, a red box indicates a `FiLM` layer (learns feature-wise gains and offsets from k' and k'' to transform a latent feature), a green triangle indicates an ELU function, and a blue triangle indicates a leaky ReLU function. b represents the width (in a one-hot encoding) of the adjustment variables. **C.** The decoder (blue box) generates estimates of the parameters for the negative binomial distribution with location parameter ω and dispersion θ. If ZINB is used, it also produces π, the probability that a variate is 0 and not generated by the NB distribution. The X and $-$ indicate a *softmax* and *softplus* function, respectively. **D.** The single layer model starts with a GAT version 2 layer with h heads. **E.** (pink box) The decoder consists of a graph decoder used to estimate edges of the sample-sample adjacency network, and the estimators of the distribution parameters for the NB or ZINB. The output is transformed with a sigmoid function. **F.** The full model accepts $\mathcal{T}$ tissues described by both count matrices and adjacency networks.

Figure S4: Example of vignettes describing common PREFFECT workflows. **A.** This describes how to load the PREFFECT package into Python and configure a session (terms a *factory* as it controls the workflow for the user). This step will partition the learning set into train, test and validation components. **B.** The factory can now control the training of the PREFFECT model. **C.** Once a model is trained, we can ask the factory to infer a model for a given dataset. In this case here we apply the learnt model to the training, validation or held out test set. This produces a normalized and transformed count matrix for the given dataset. **D.** Using this PREFFECT-treated dataset, we can generate embeddings and visualize clusters. Note here that the samples cluster by their batch ID (depicted using UMAPs with blue (batch 0) and orange (batch 1) dots. **E.** PREFFECT can use conditioning to adjust for batch effects when inferring counts. Using clustering on the frequency vector ω, we now see intermixing of samples with different batch IDs.

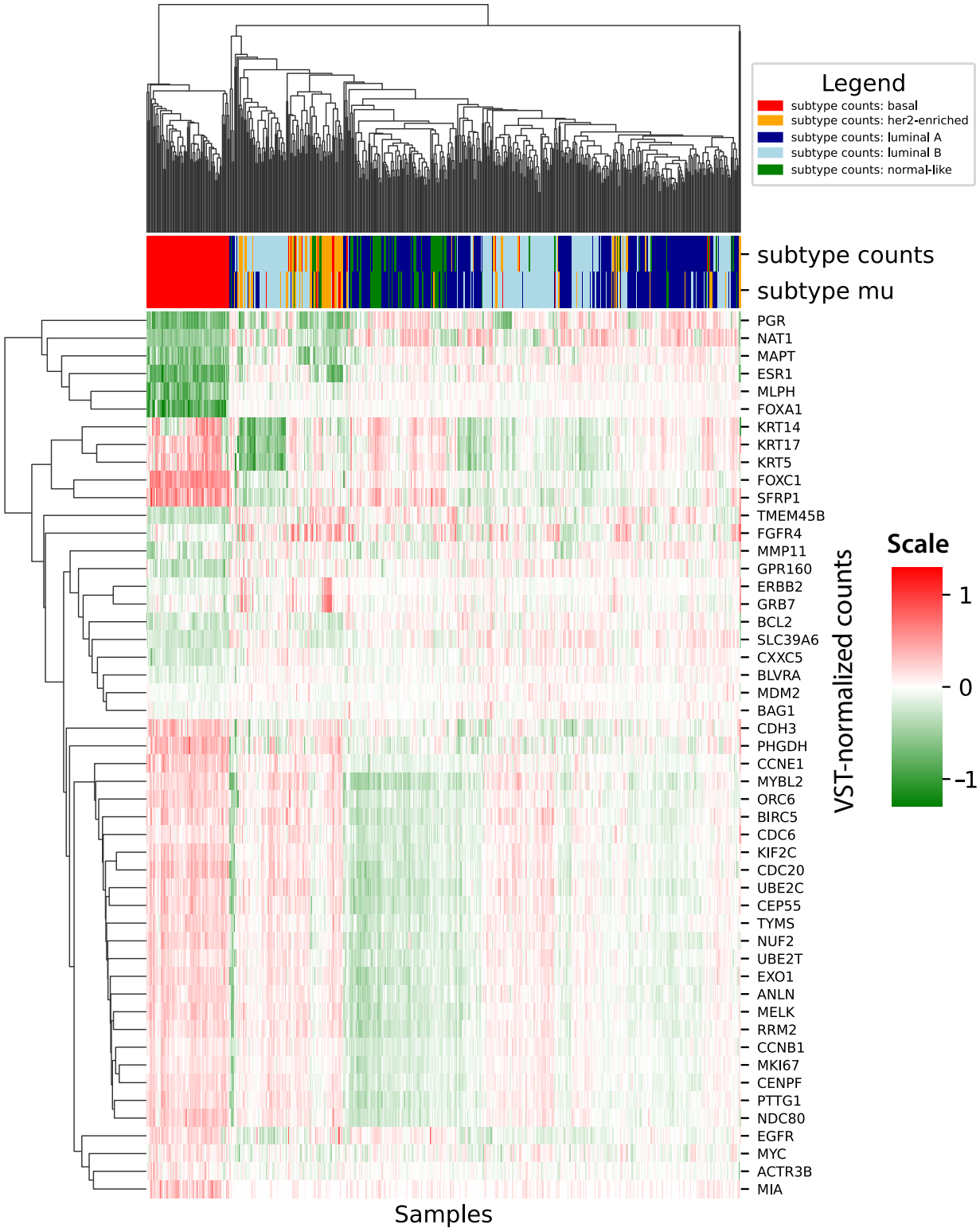

Figure S5: Patient clustering of VST-normalized counts. Counts from the same fRNA-seq dataset as Figure 9 but here the data was normalized using a variance stabilizing transformation (commonly used in bulk RNA-seq studies) and subjected to hierarchical clustering. The adjusted rand index (ARI) of this clustering with respect to breast cancer subtype is lower (0.28) than the raw count data of Figure 9A (0.32) and much lower than the ARI of the PREFFECT-adjusted Figure 9C (0.61).