report

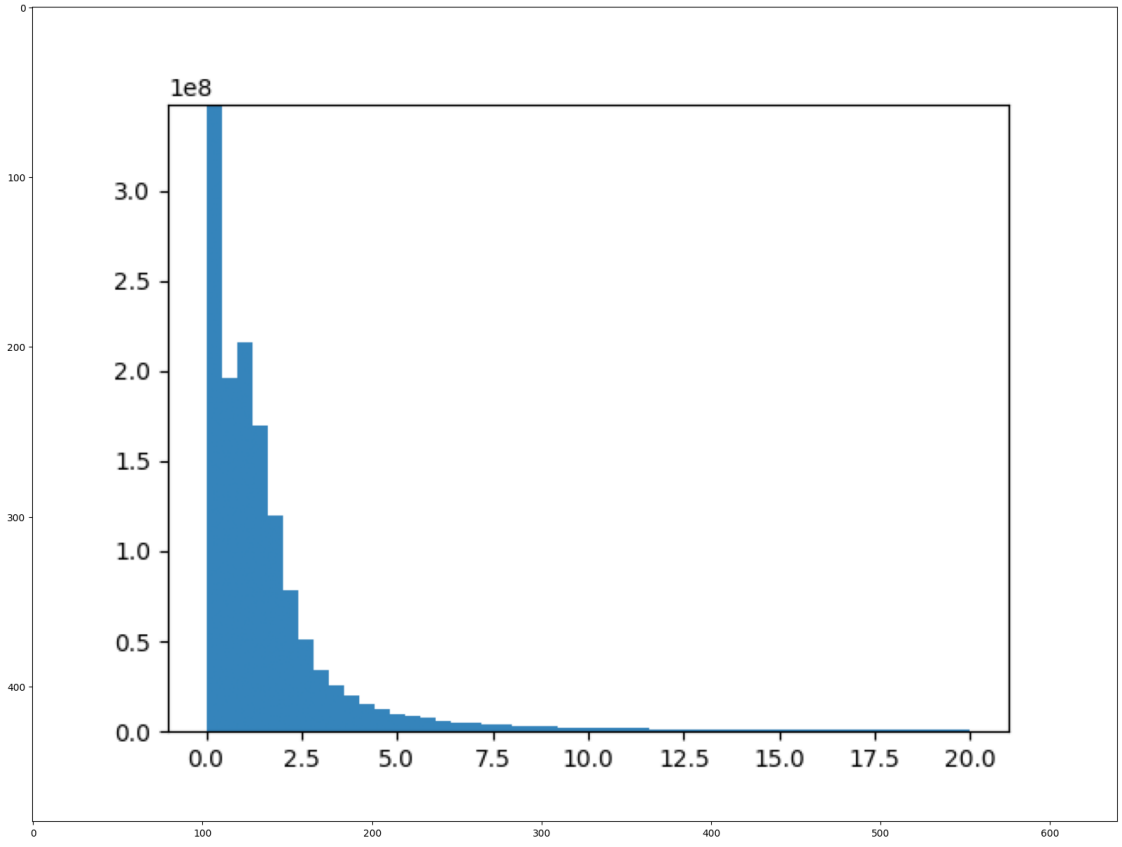
December 15, 2022

0.0.1 Number of observations / variables in the dataset

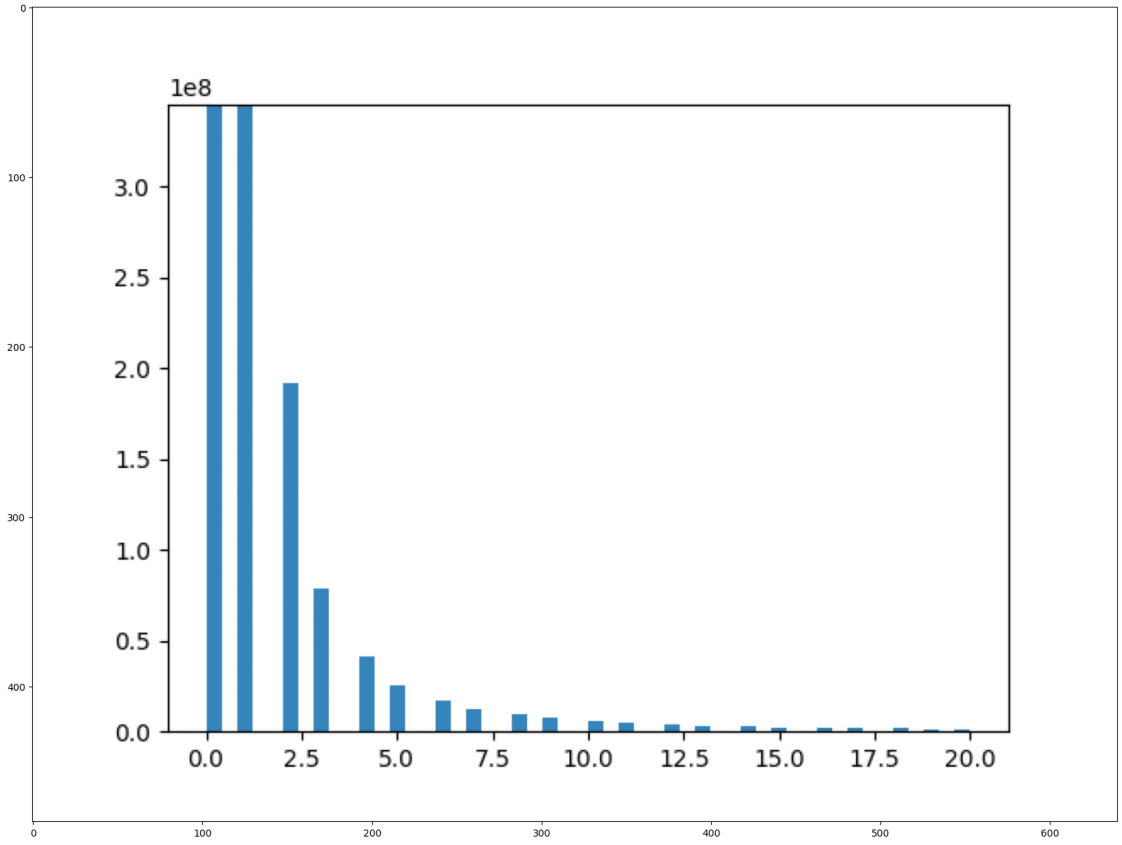
	num obs	num variables
train	72208	5000
test	18052	5000

0.0.2 Histograms of the data (capped)

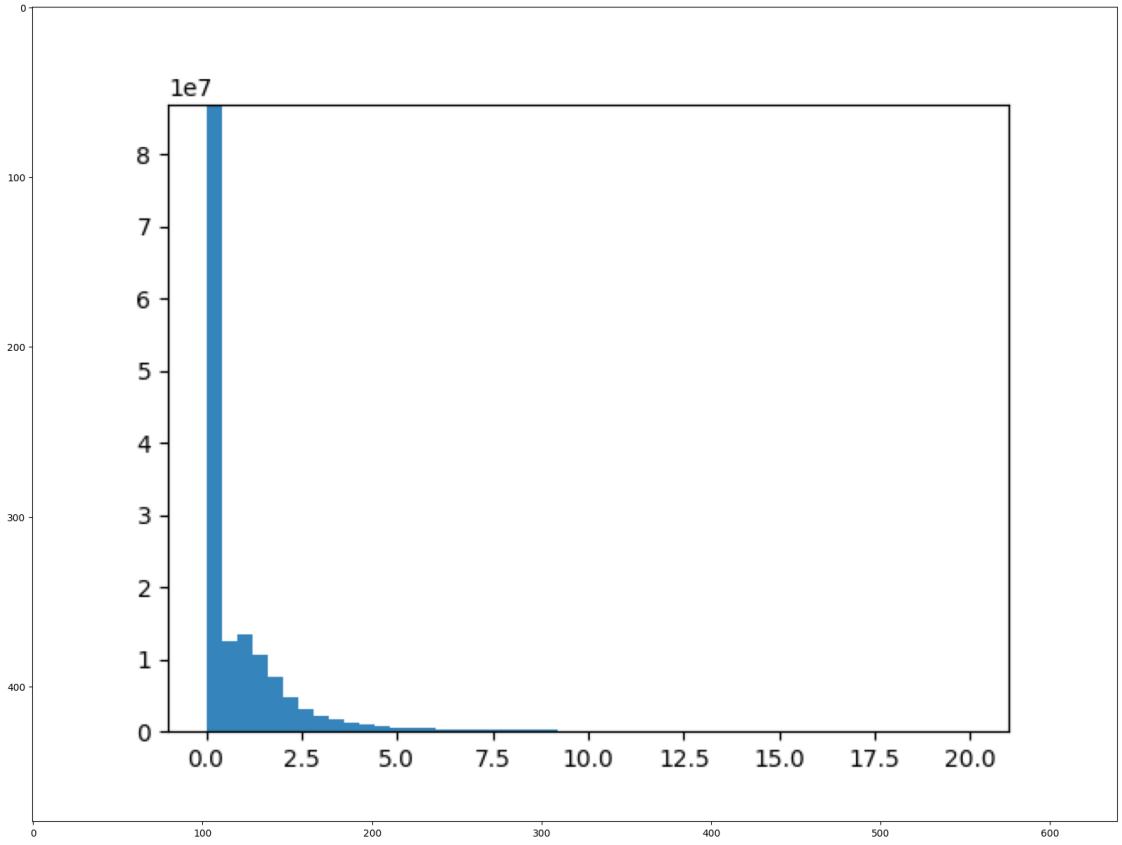
preprocessed train



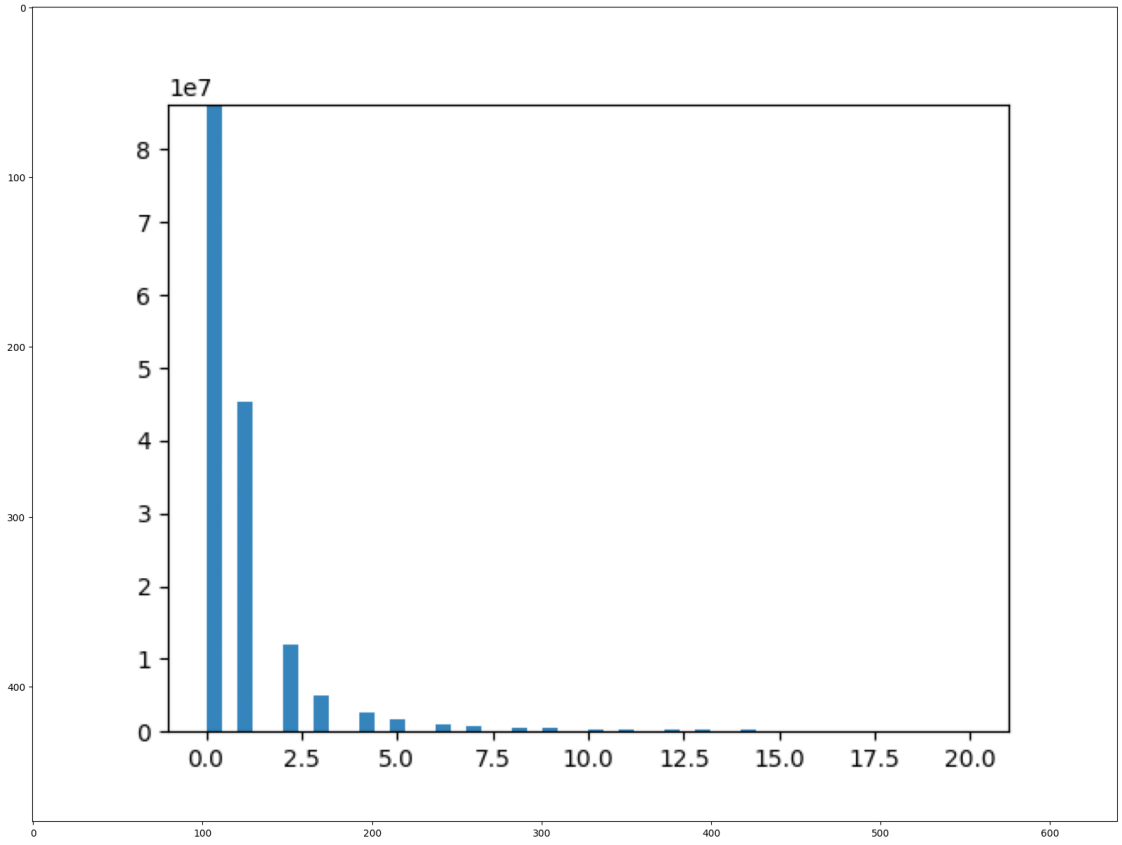
raw train



preprocessed test


1e7
8
7
6
5
4
3
2
1
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
0
100
200
300
400
500
600

raw test

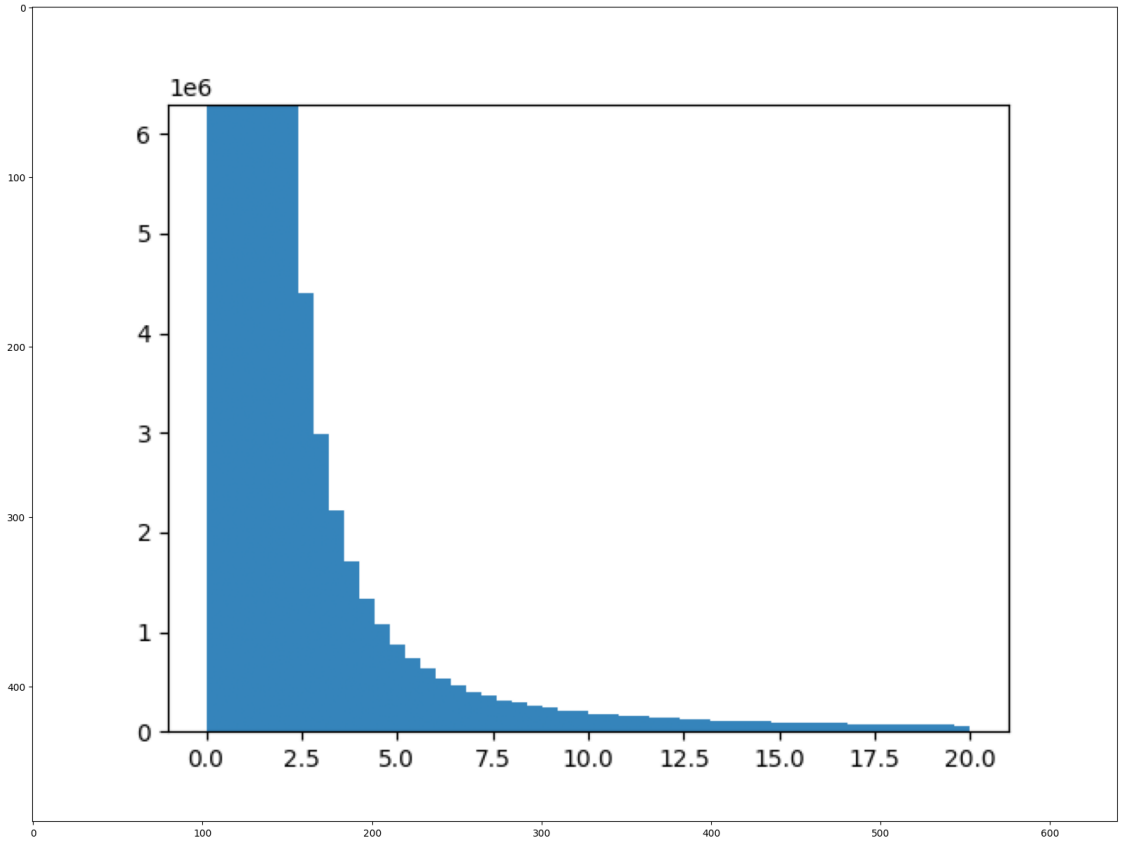


0.0.3 Data preprocessing

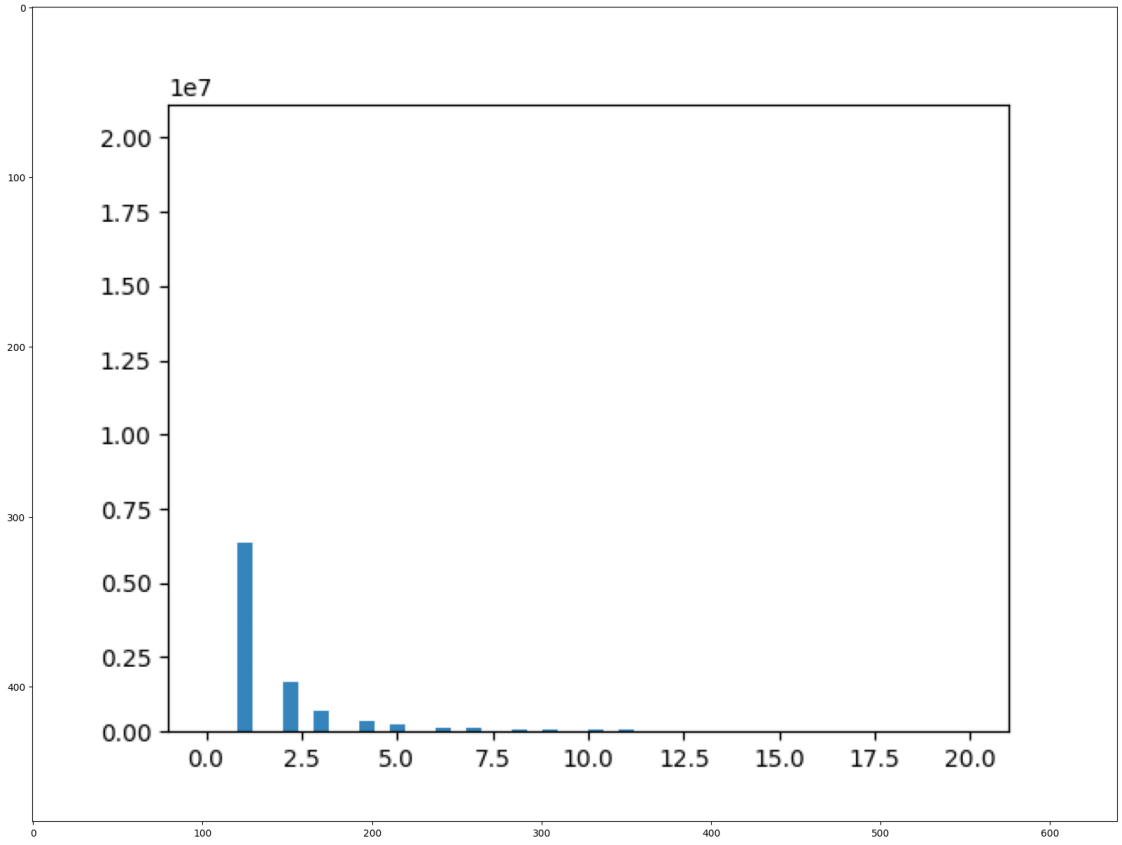
Data was preprocessed by dividing it by the GEX_size_factors variable in the obs

0.0.4 Histograms with zeros left out (capped as well)

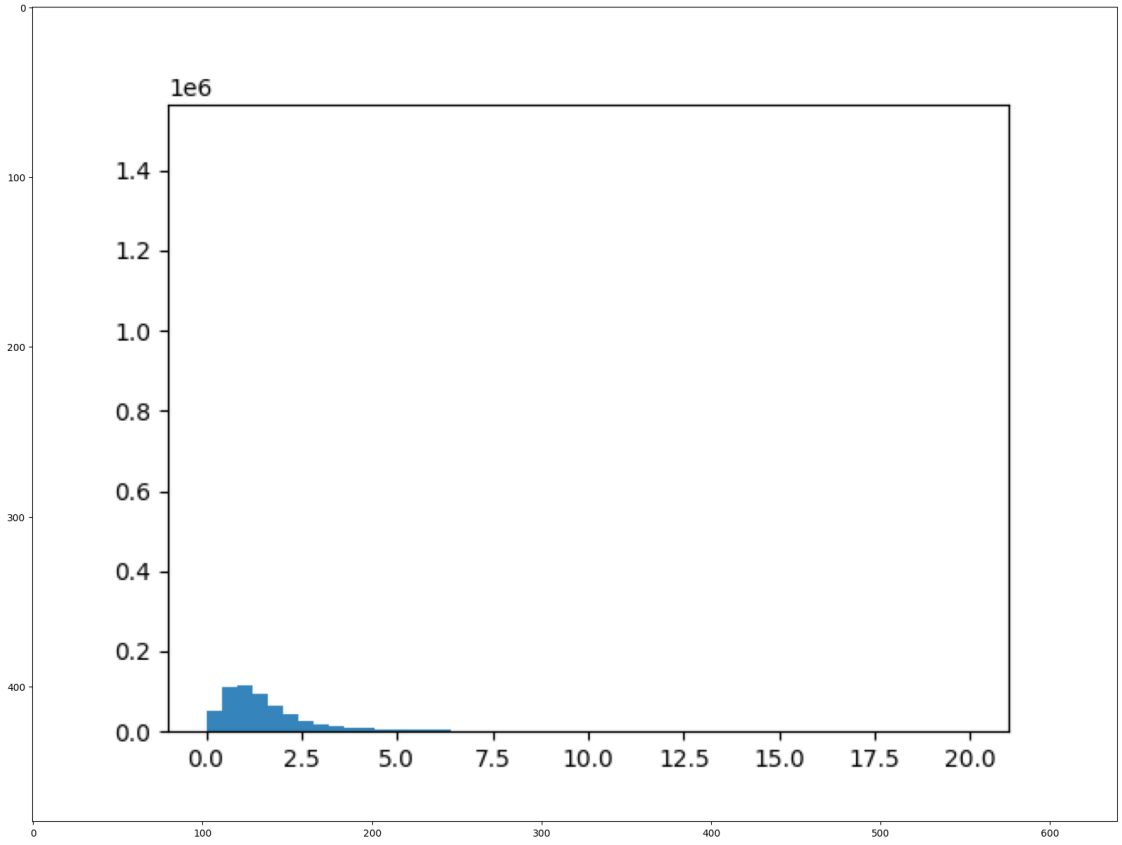
preprocessed train


1e6
6
5
4
3
2
1
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
0
100
200
300
400
500
600

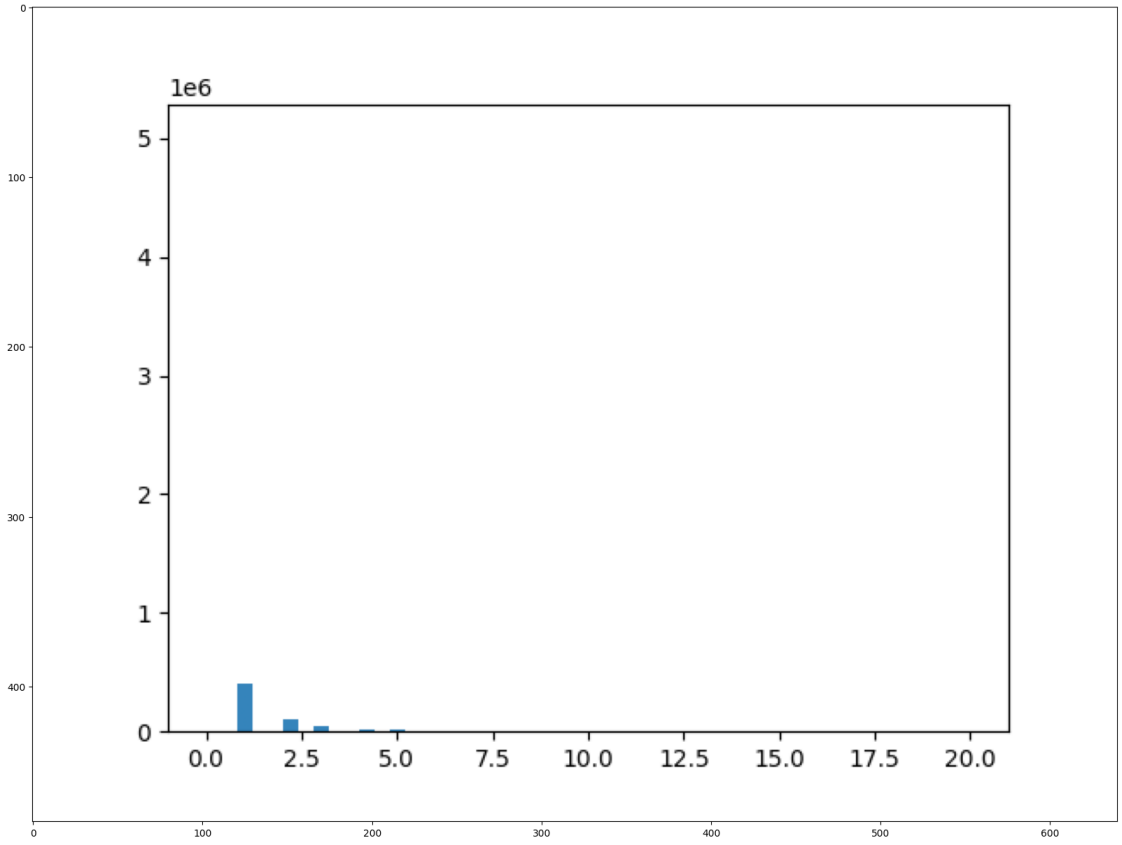
raw train



preprocessed test


1e6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
0
100
200
300
400
500
600

raw test



0.0.5 What is the distribution of the data

At first it seems that a poisson distribution would be sufficient. (which is common in RNA modelling, according to my almost nonexistent biological knowledge).

However due to the variance being larger than the mean (and a lot of zeros present) it is wise to suggest a negative binomial distribution (which is tightly connected to poisson distribution)

0.0.6 What are the contexts of the adata.obs

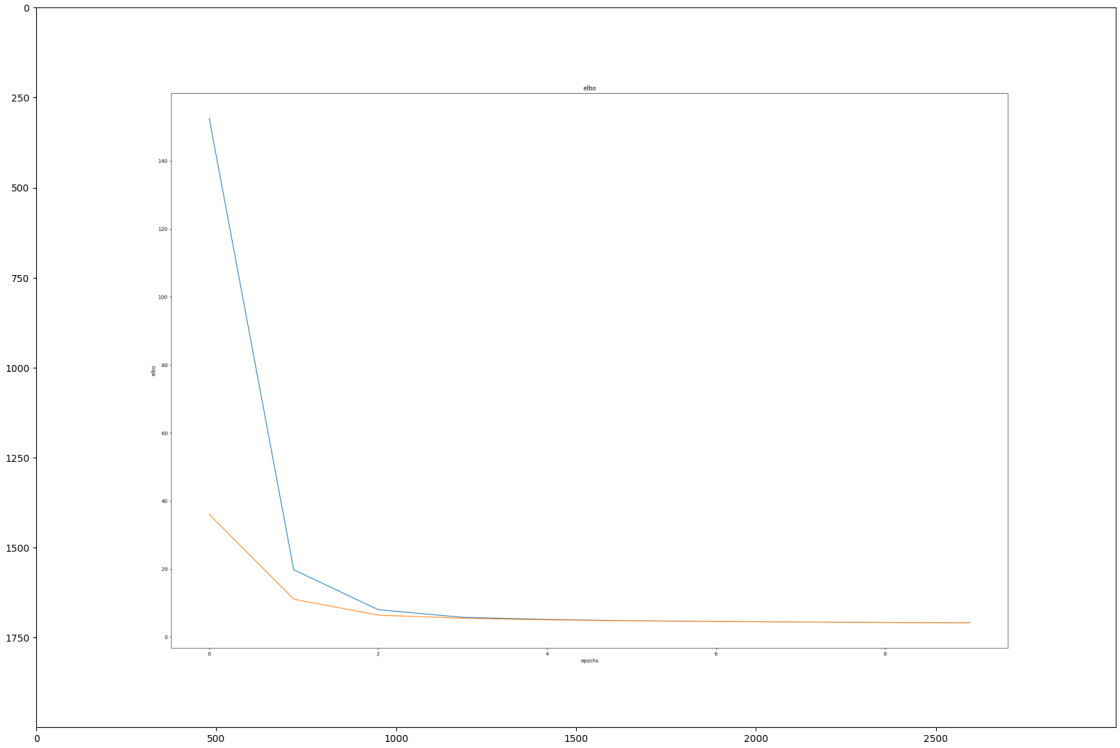
additional, explanatory information for each observation. Biological details about RNA sequences and the patients

0.0.7 number of categories (donors, sites, cell types)

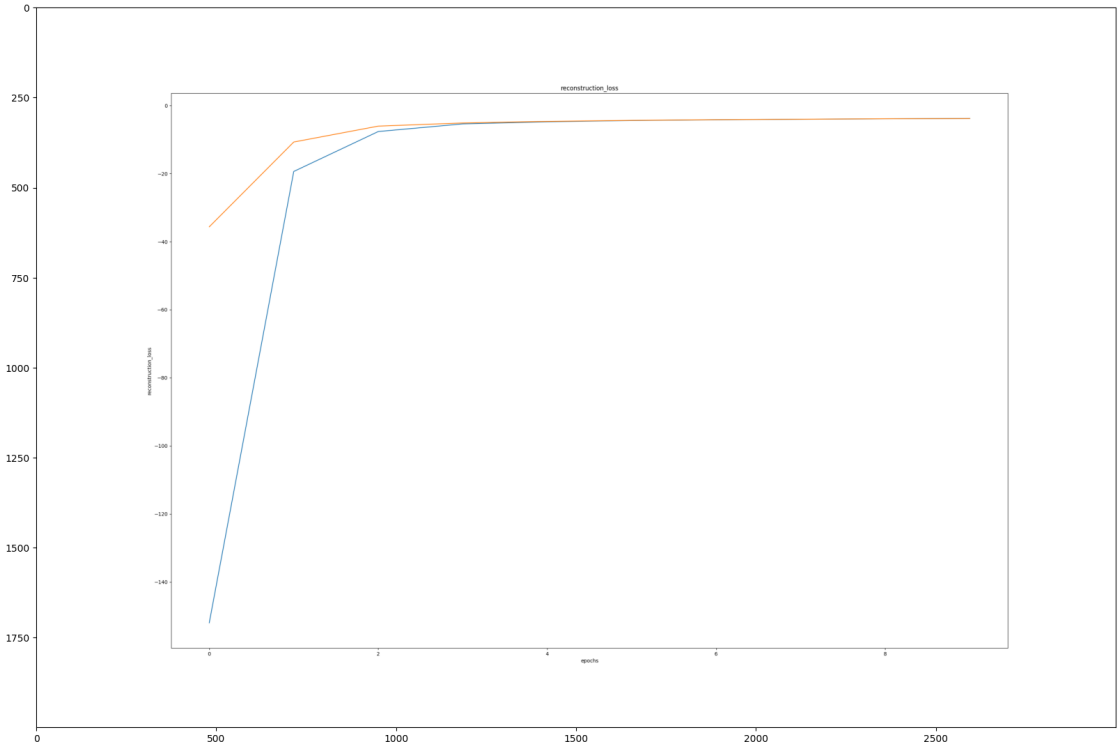
	n_donors	n_sites	n_cell_types
0	9	4	45

0.0.8 Training of the vanilla VAE (latent size of 32, 10 epochs, 128 batch size, 4 hidden layers of width 256 in encoder/decoder)

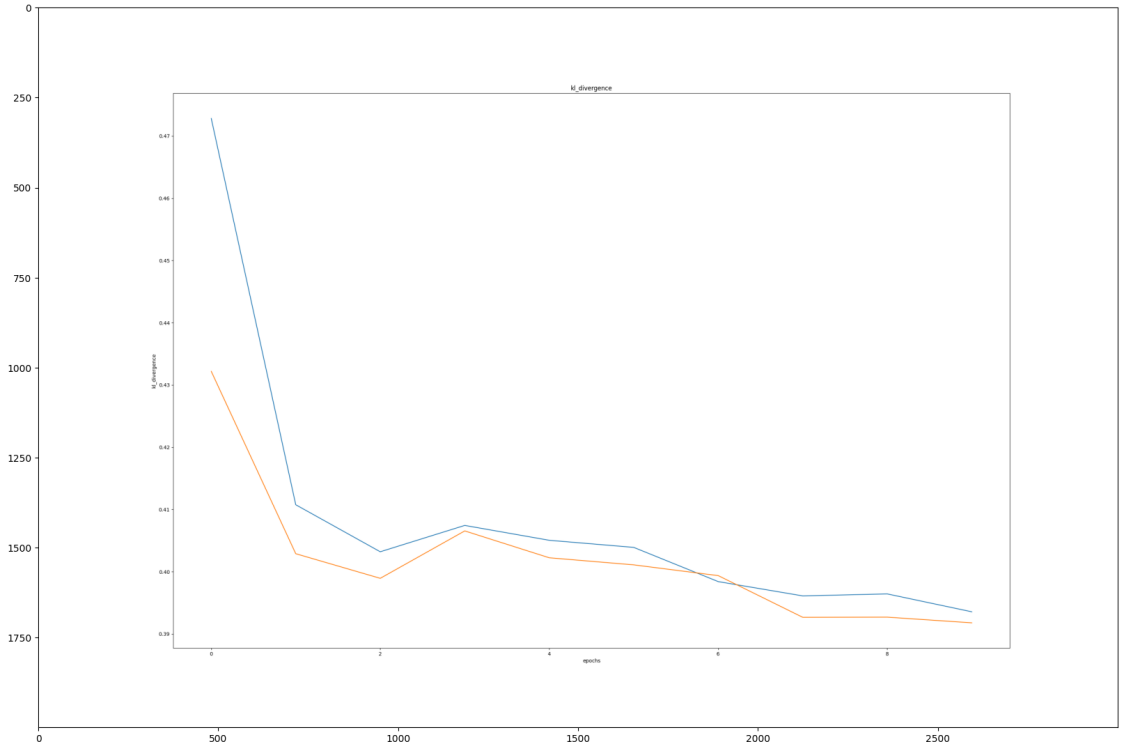
ELBO



reconstruction loss



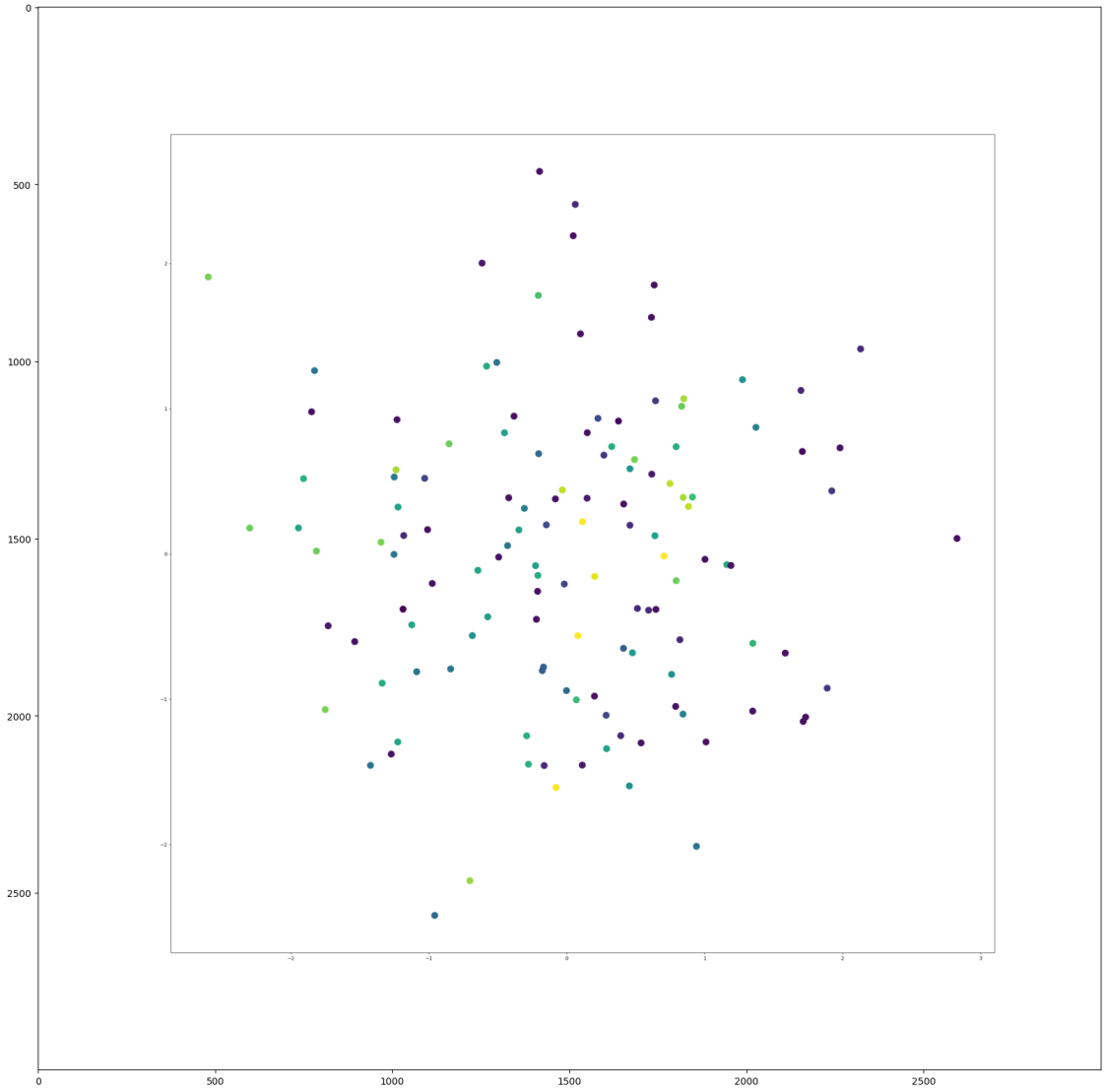
KL divergence



0.0.9 3 runs with different latent sizes (and minimum number of PCA components to reach 95% variance explained)

	latent_size	elbo	kl divergence	recon loss	min components	explained_variance
0	32	4.220623	0.391793	3.828830	31	0.971123
1	16	4.306129	0.421133	3.884996	16	1.000000
2	8	4.416090	0.460025	3.956065	8	1.000000

0.0.10 2-PCA for the 32-dimensional latent_size, coloured cell types



0.0.11 Overall thoughts of the training

I chose to use the raw dataset, since there was no clear empirical benefit of using the preprocessed dataset. I'll be using the raw data (integers) for the custom decoder as well. That way it'll be easier to compare both methods.

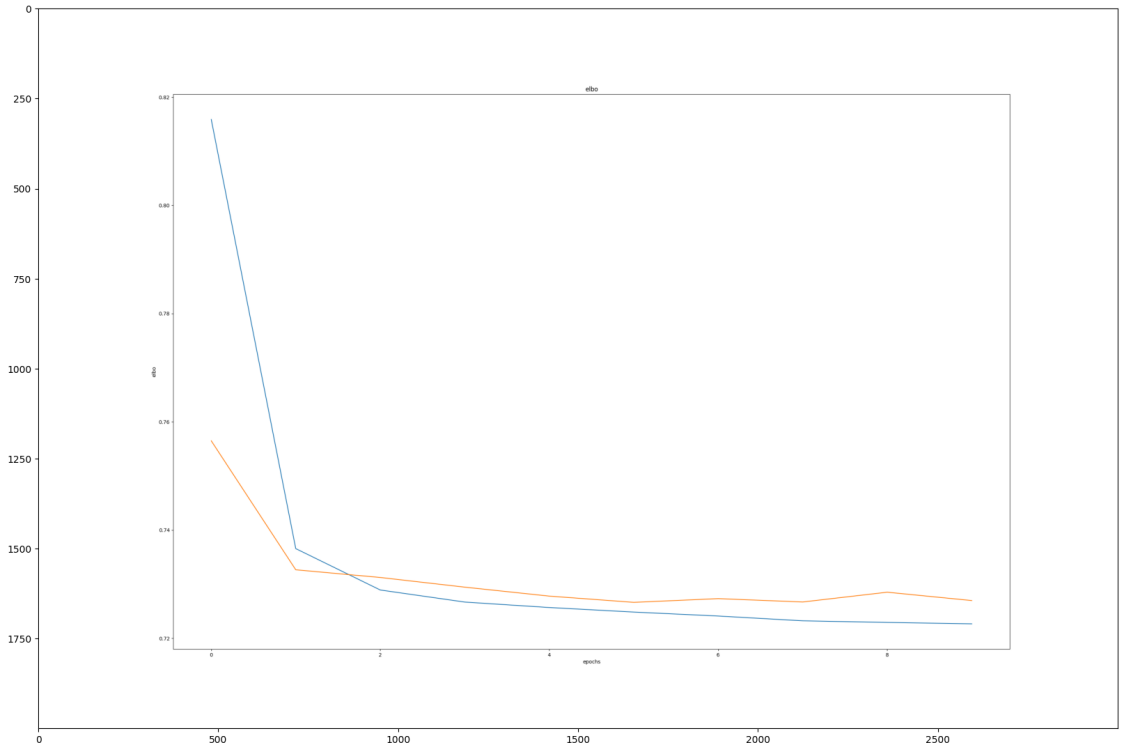
I used latent_size of 32, since it performed better on the test set than 128, 64, 16 or 8. I used 10 epochs and such latent sizes for the training times to not be too long (while training on cpu).

0.0.12 Custom decoder

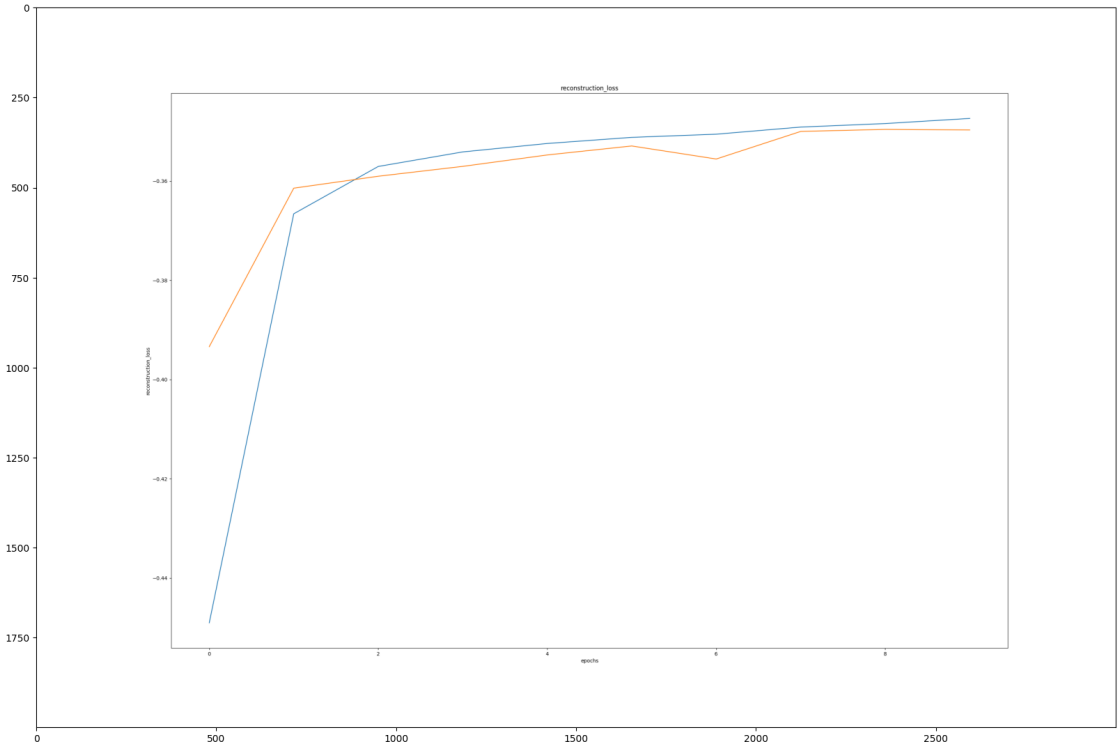
Negative binomial decoder results in few times lower ELBO, reconstruction loss is much lower and KL divergence is slightly lower as well.

I chose that distribution mainly because the data resembles it a lot.

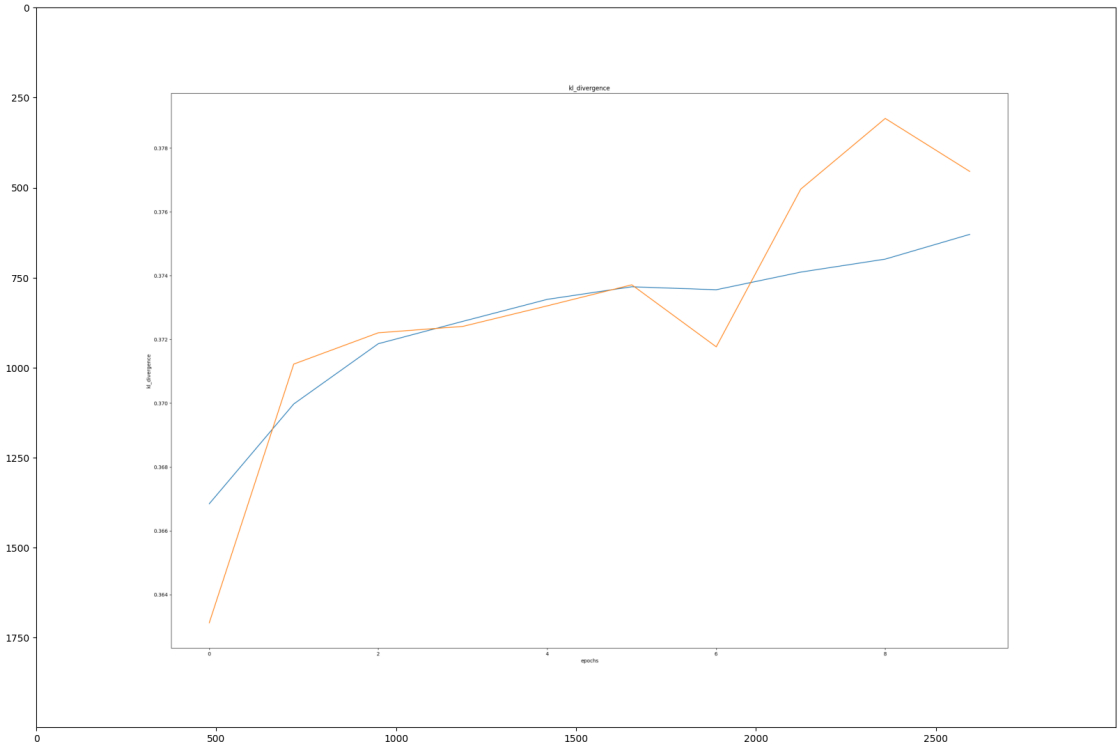
ELBO



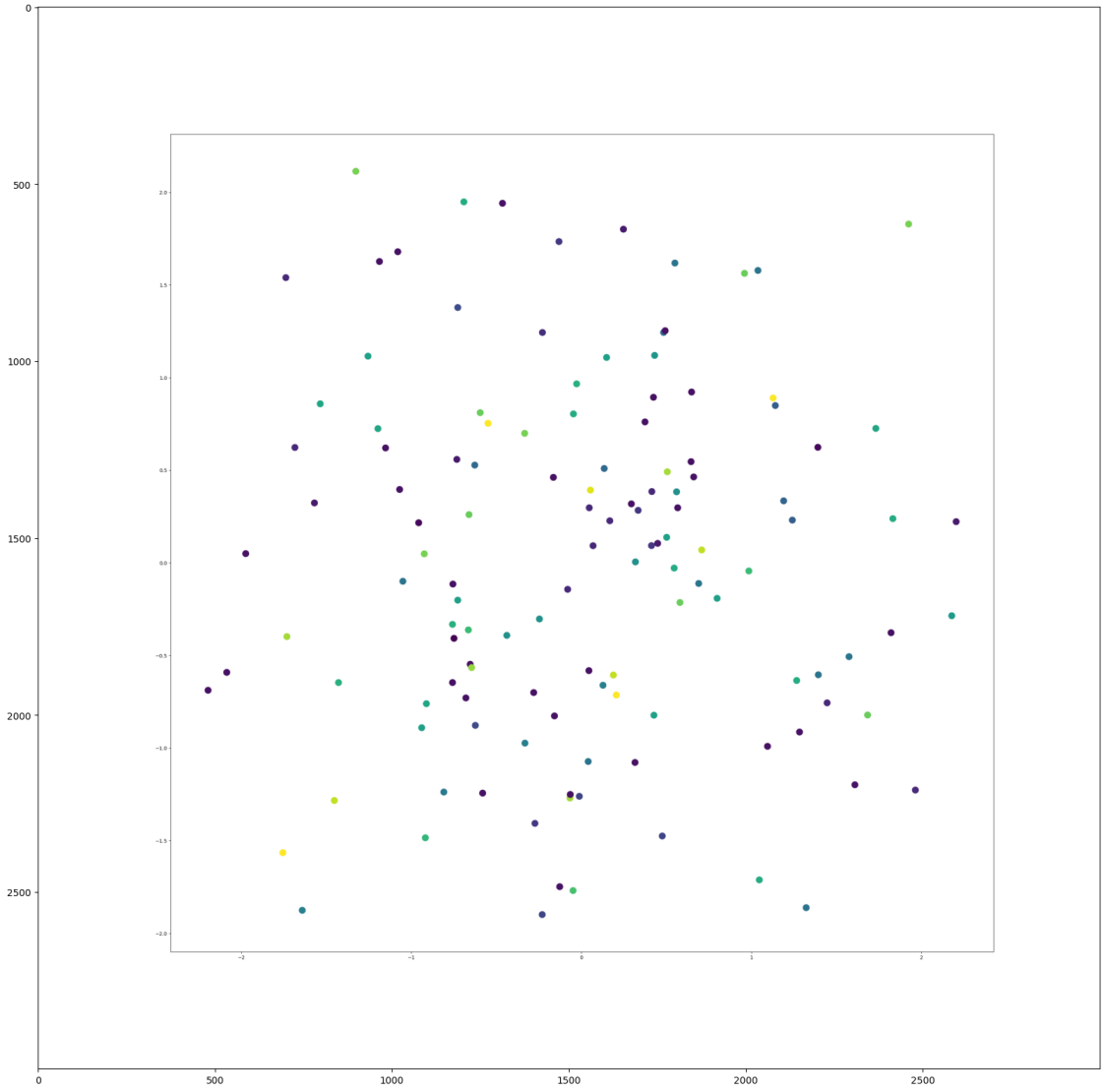
reconstruction loss



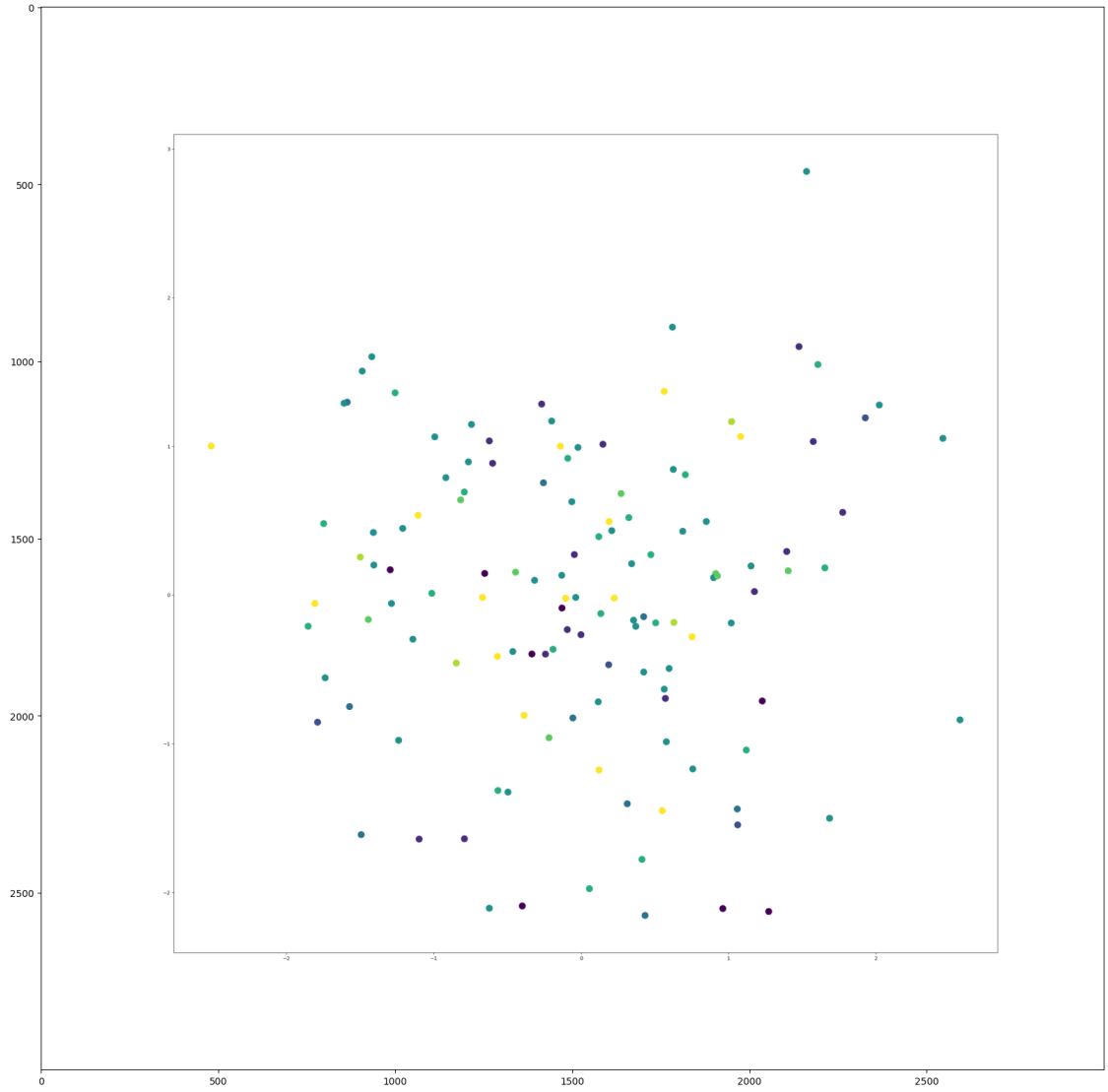
KL divergence



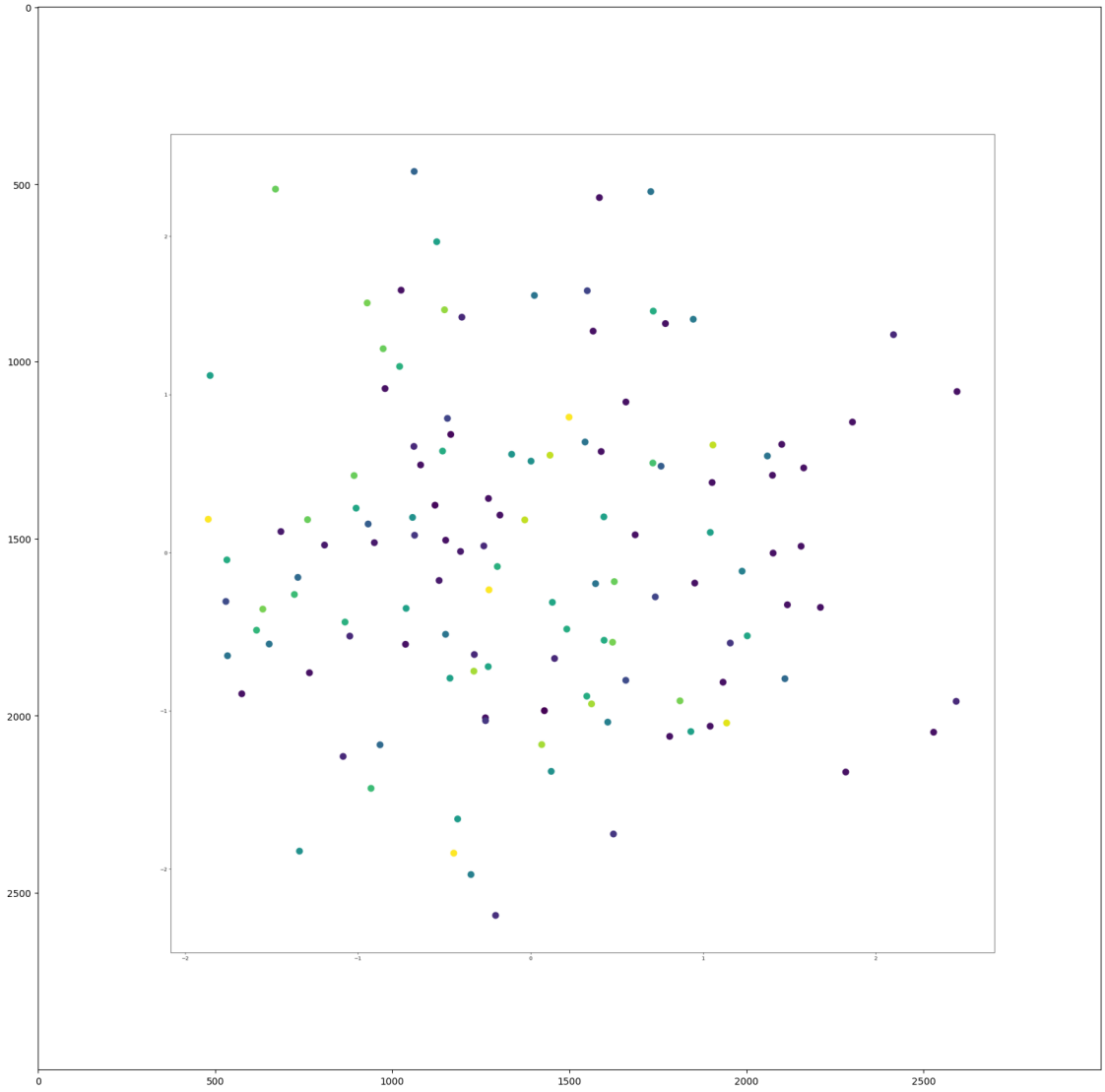
0.0.13 2-PCA for 32-latent_dim custom VAE, coloured cell type



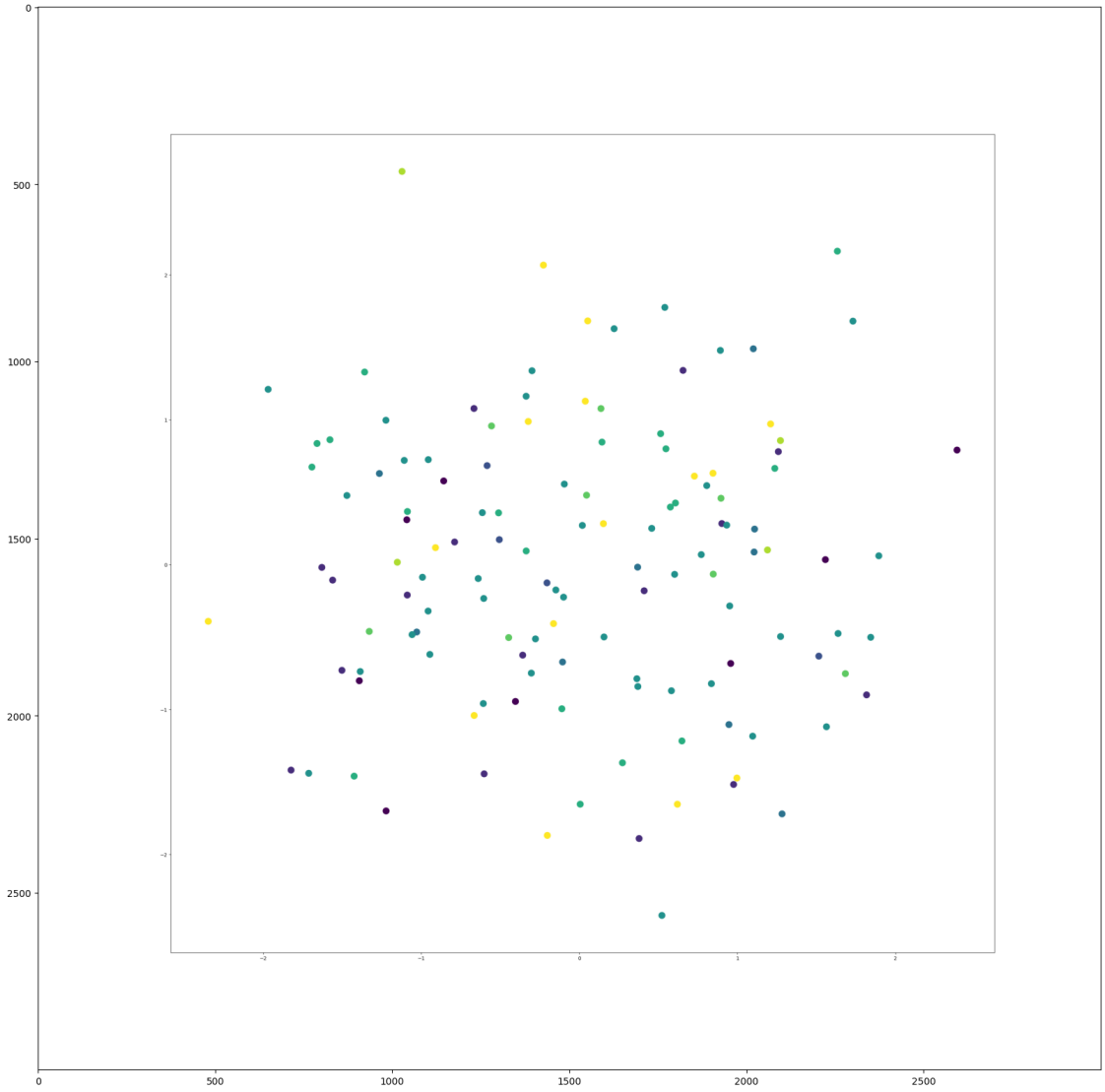
0.0.14 2-PCA for 32-latent_dim vanilla VAE, coloured donor id



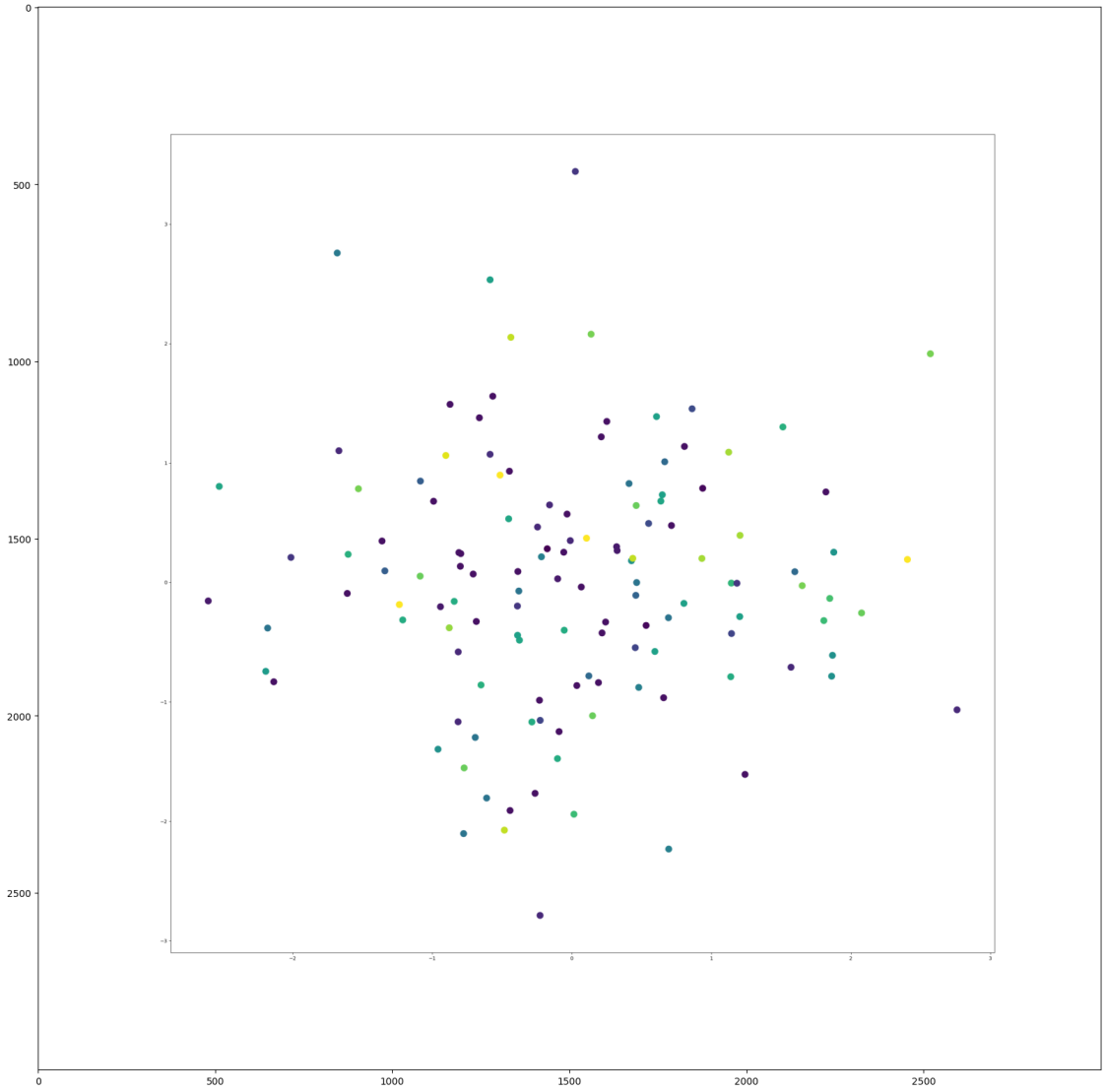
0.0.15 2-PCA for 32-latent_dim vanilla VAE, coloured site



0.0.16 2-PCA for 32-latent_dim custom VAE, coloured donor id

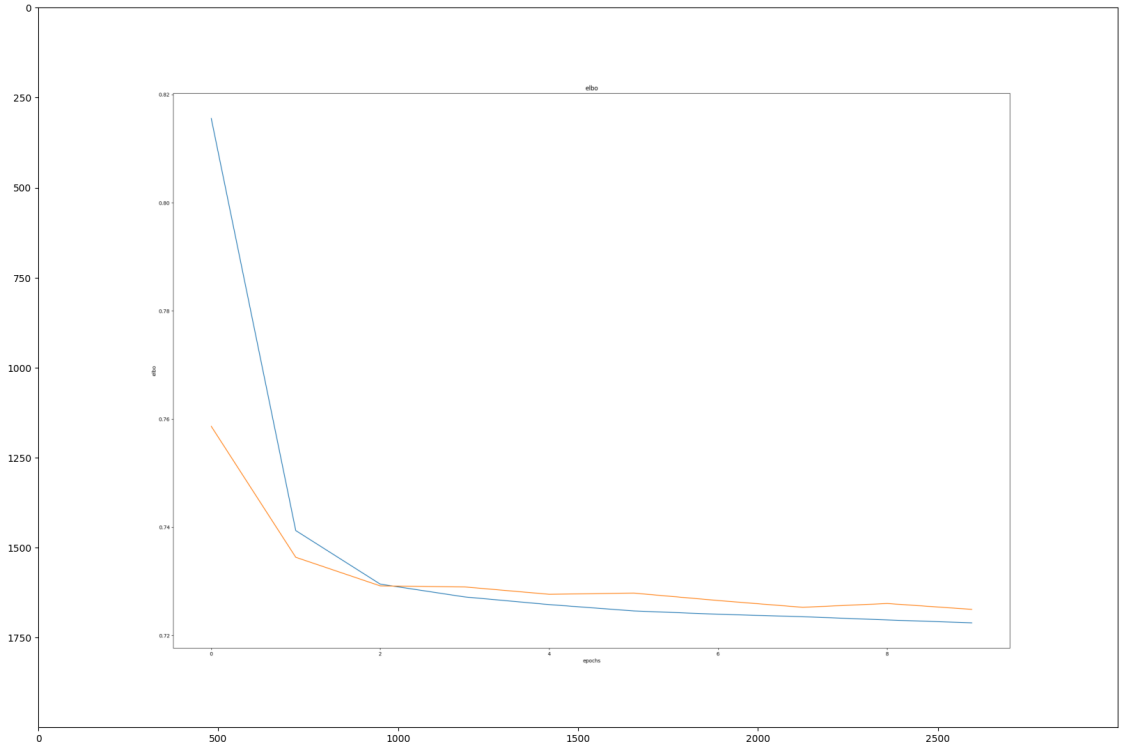


0.0.17 2-PCA for 32-latent_dim custom VAE, coloured site

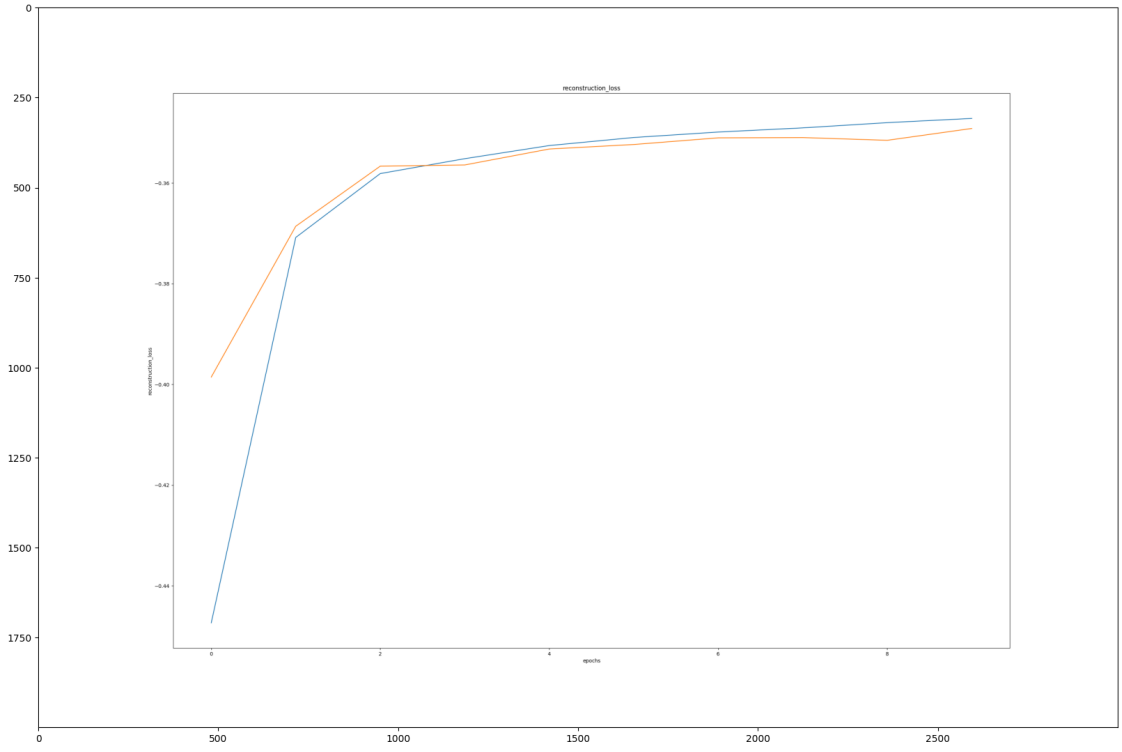


0.0.18 I took one-hot encoded site and concatenated it to the latent variables, fed into the decoder

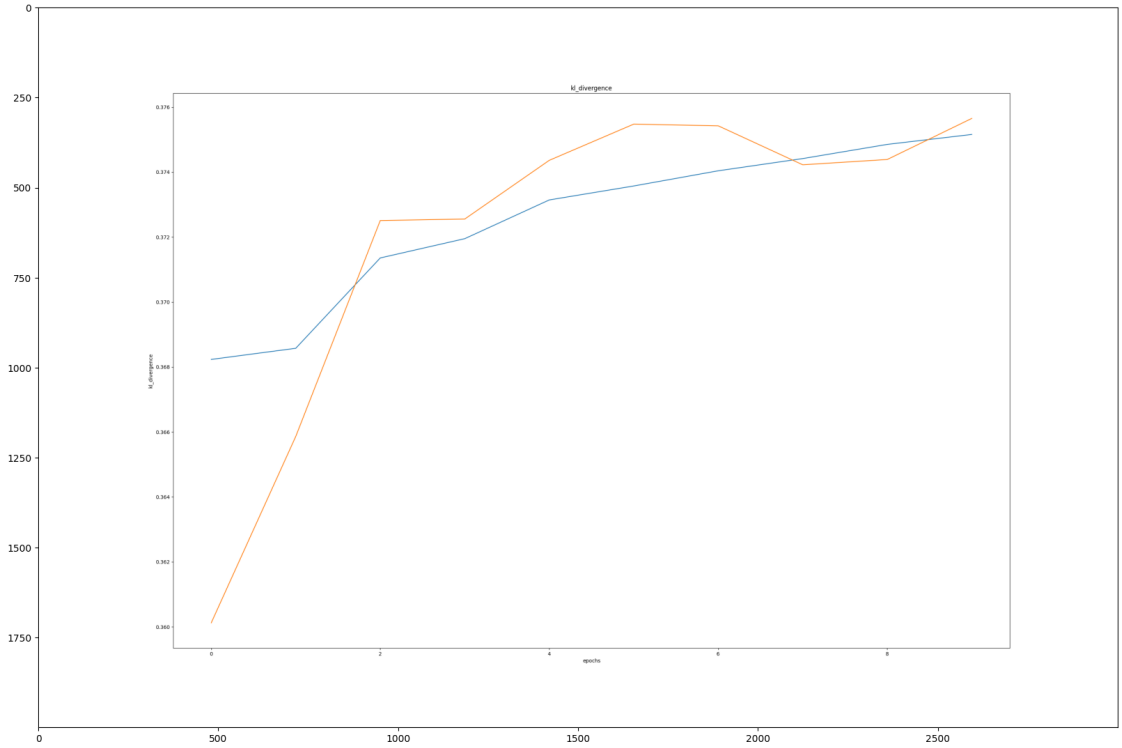
ELBO



reconstruction loss



KL divergence



0.0.19 2-PCA, coloured cell type