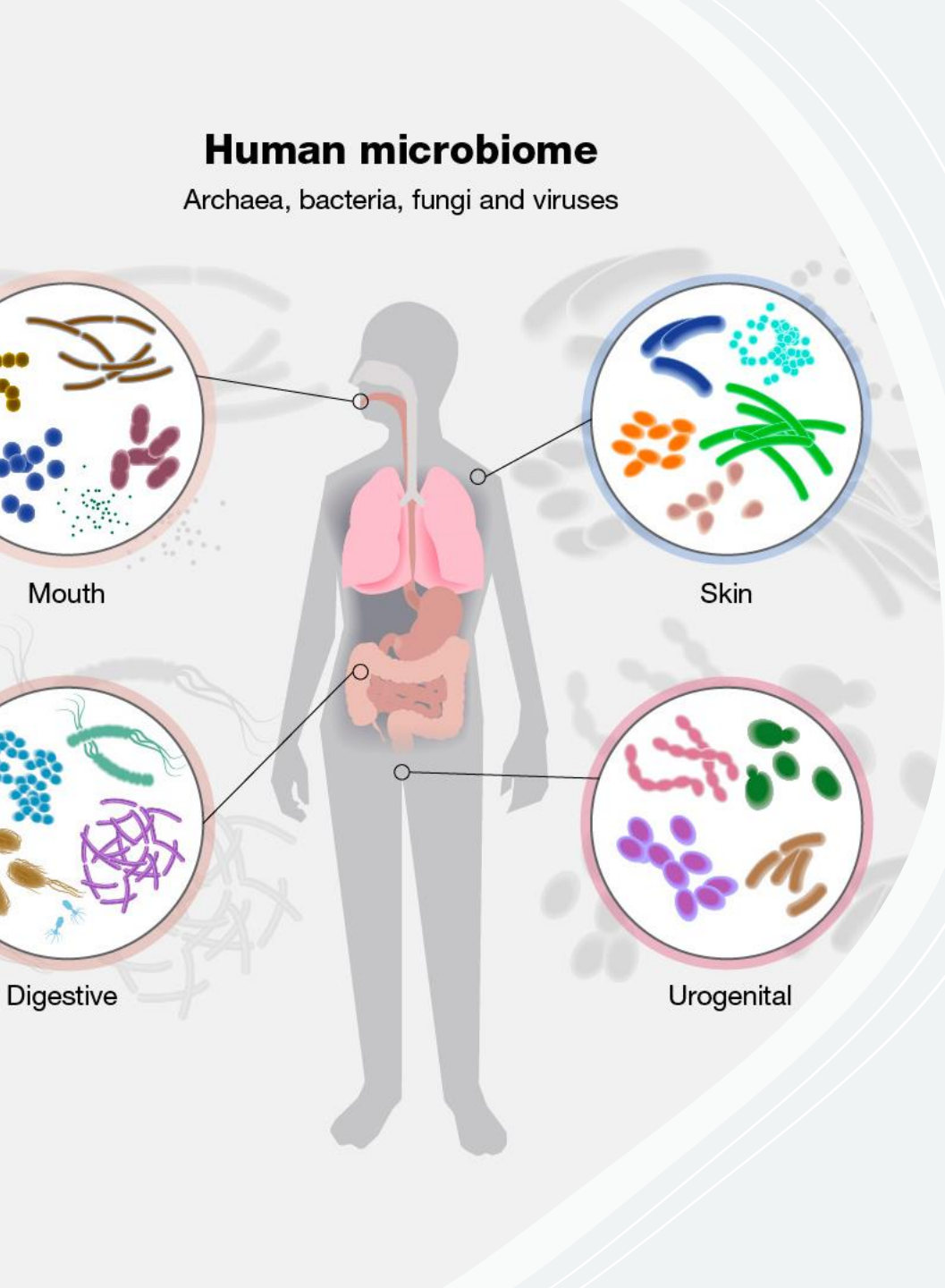


Metagenomic Binning

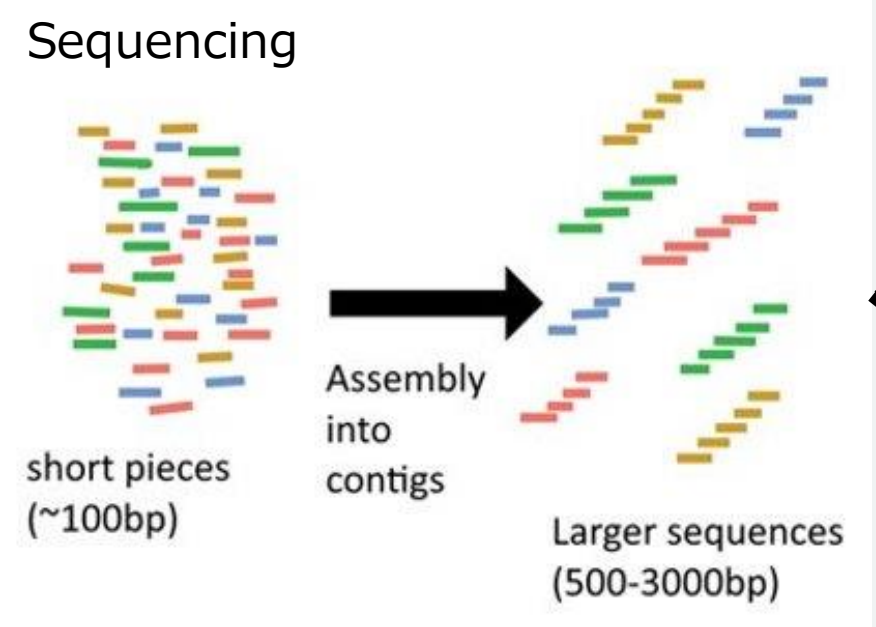
Mads: Preprocessing and clustering
David: Variational Autoencoder and clustering
Panagiotis: NN classifier
Jie: Treebased classifier

Metagenomics data

Dataset from Critical Assessment of Metagenome Interpretation (CAMI 2) challenge

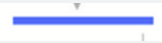
Millions of reads

700k contigs



Composition

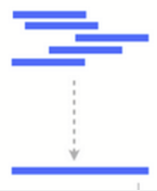
What does each contig look like?



Tetramer-frequencies

Abundance

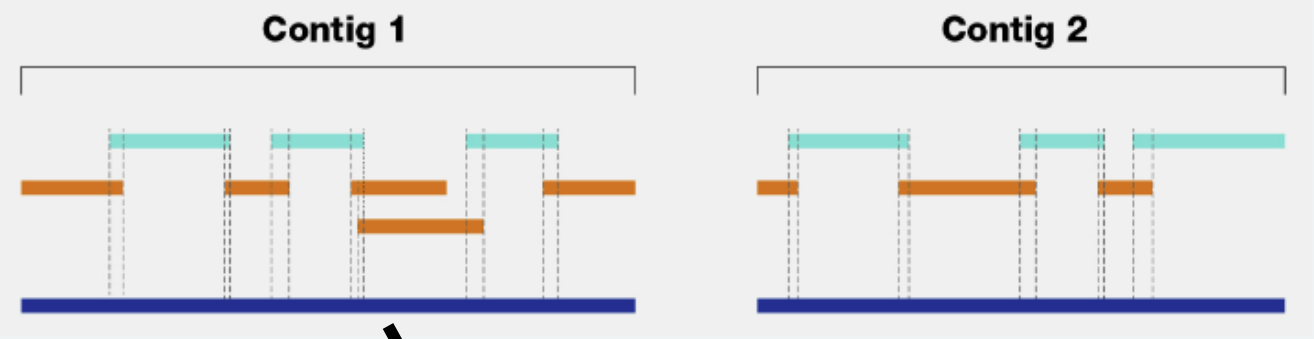
how much DNA is mapping to each contig, in 10 technical replicates



The big question

Which contigs comes from the same organism, and what organism is that?

Composition



- Binning genomics data based on entire sequences requires immense computational power
  - Can be simplified by using tetramer-composition
  - 103 combinations of A, T, C, and G

- Ratios of nucleotides and tetramer-composition varies between species, and can be used as a "fingerprint".

K-mer	Frequencies
ATG	0.4
GCA = TGC	0.4
CAA	0.2

Metagenomics data

Input data

Composition data

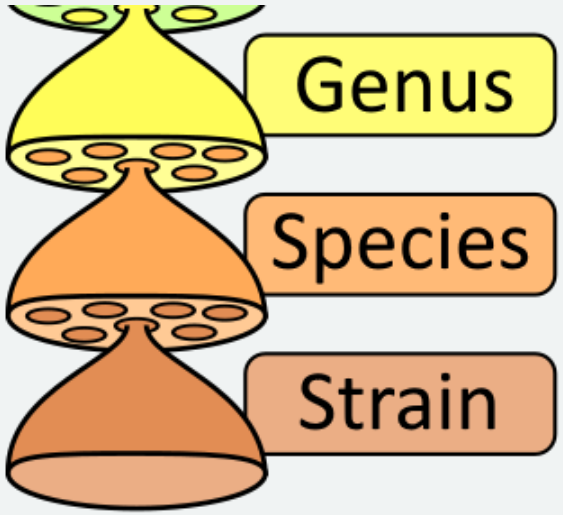
	4mer 1	4mer 2	…	4mer 102	4mer 103
contig1					
…					
Contig 700k					

Abundance data

	1	2	…	10
contig1				
contig2				
…				

Ground truth

We chose Genus, as species and strain is very specific in biological context



224 different genera

596 different species

2006 different strains

Metagenomics - Scientific application

Why is it important?

Objectives - outline

1. Unsupervised – Use dimensionality reduction and clustering to pool organisms together based on composition and abundance (Metagenomic Binning)

Methods:

- Variational Autoencooder, PCA, UMAP Kmeans, DBSCAN

2. SupervisedAnnotate taxonomy of contig, can we predict the genus or species of a contig based on composition alone? (Annotation)

Methods:

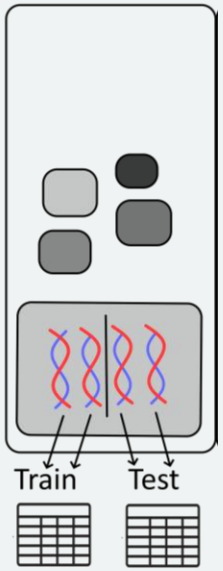
- LightGBM and Neural Network classifiers

The perfect start - 99% accuracy

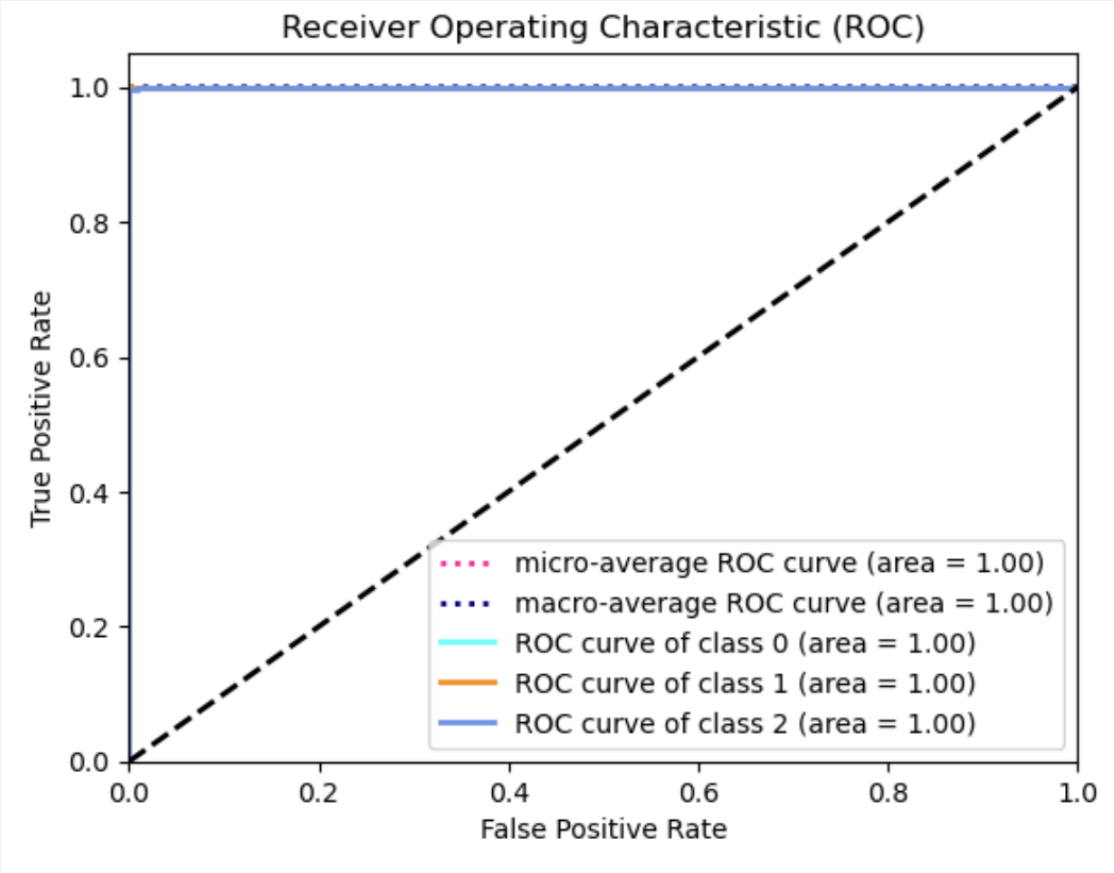
Classification on small dataset to test the feasibility and model performance using different methods.

Data preprocessing:

- subset samples: n = 5,560
- Limited dimensions: #label = 5
- Train/test split: completely random



The perfect start - 99% accuracy



Real journey

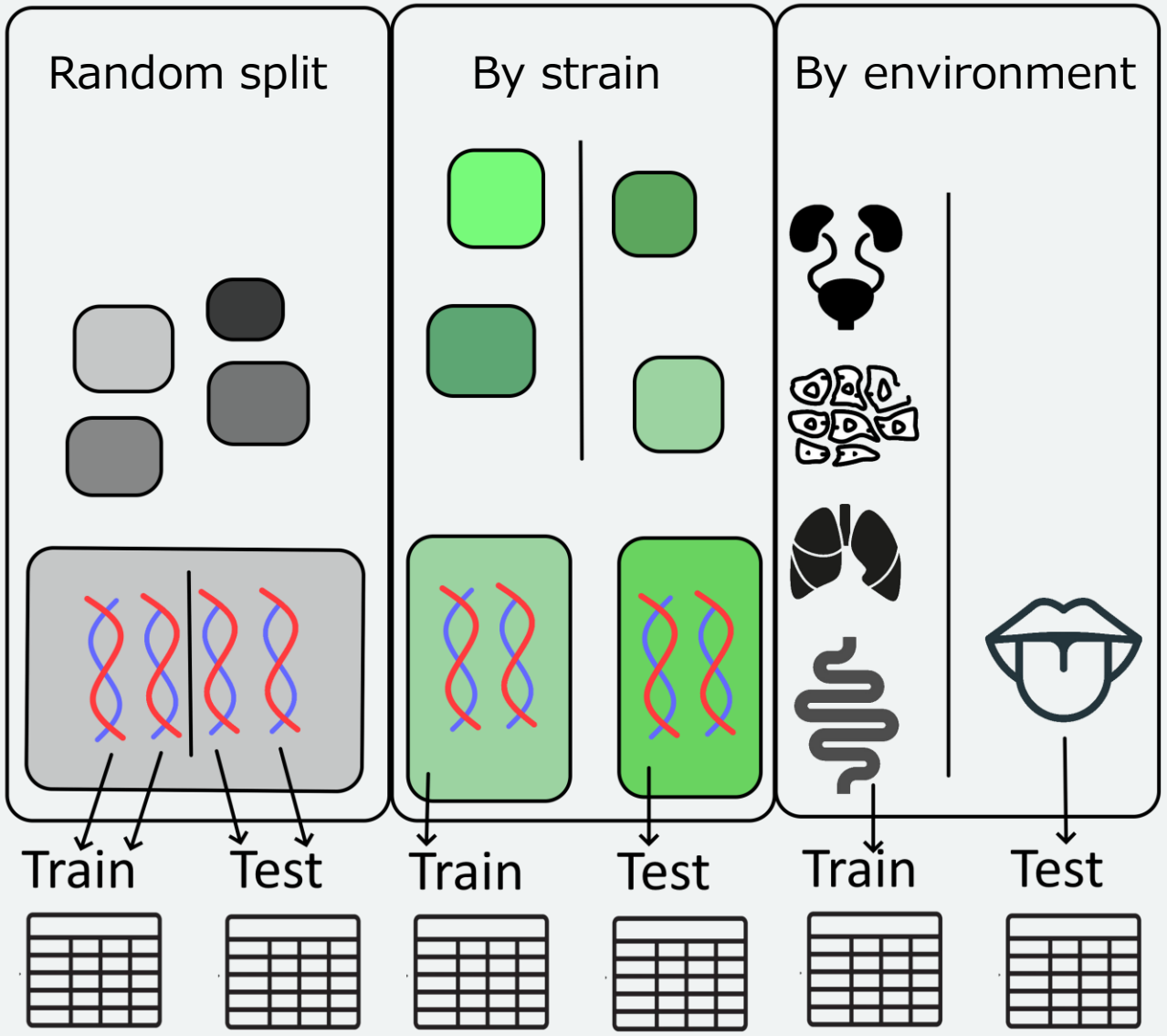
with data.2.0

Splitting in the correct way

Random split:
Splitting completely random
Problem: DNA from the same strain wind up in both training and test sets, very similar (perfect score)

By strain:
Splitting strain for each species into training and test datasets.
Problem: in some cases testing on unseen data (for dissimilar strains)

By environment:
Training on known species, but unseen strains.



Unsupervised clustering

Problem: Which pieces of DNA comes from the same organism/species/genus.

Included **composition data** and **abundance data**:

Urogenitalia: 112 dimensions (Composition: 103 & Abundance: 9).

Methods:

PCA, UMAP and clustering

Creating a Variational Autoencoder (VAE) for dimensionality reduction using pytorch and to allow for flexibility and variations in contigs.

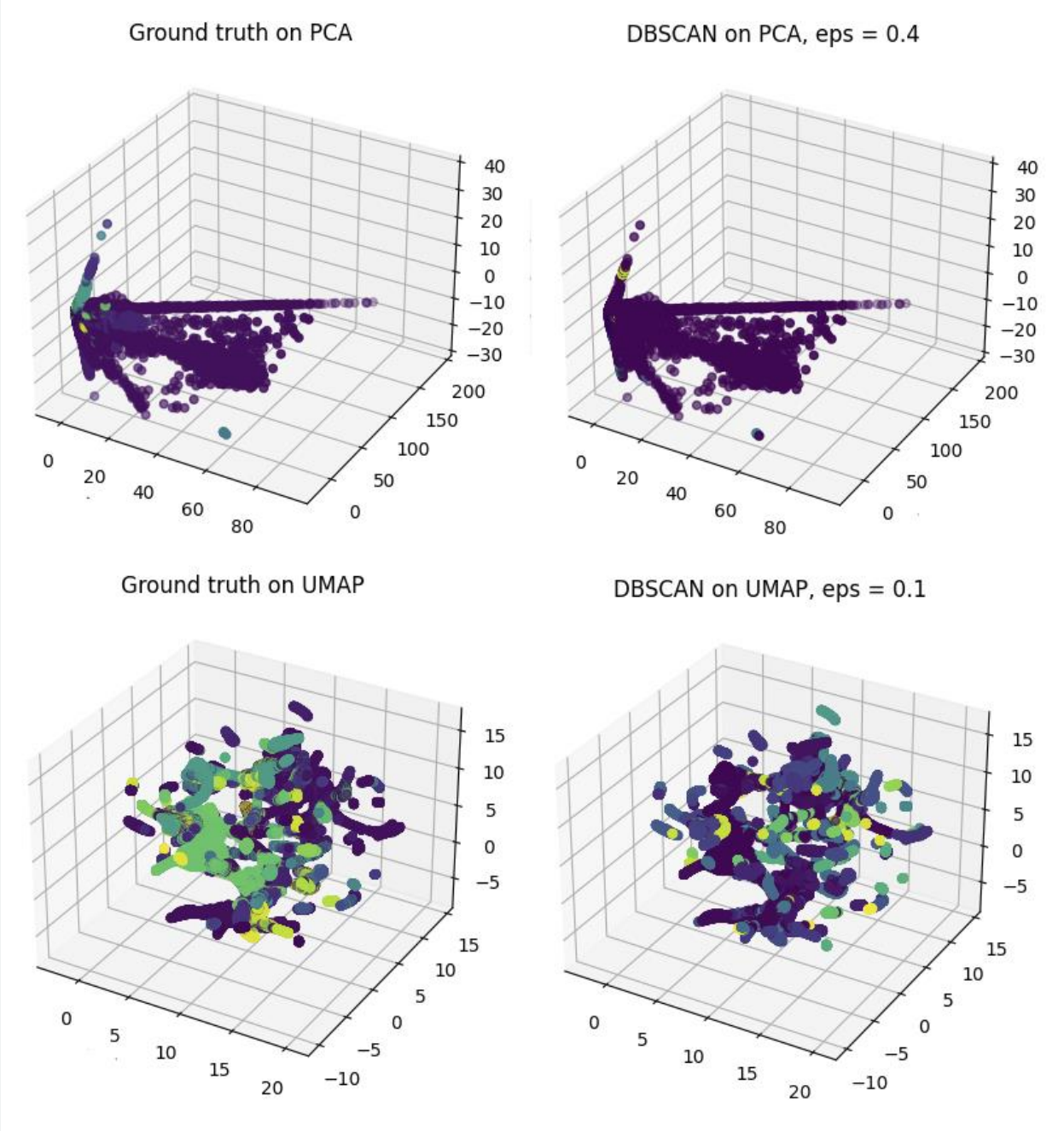
PCA & UMAP

PCA

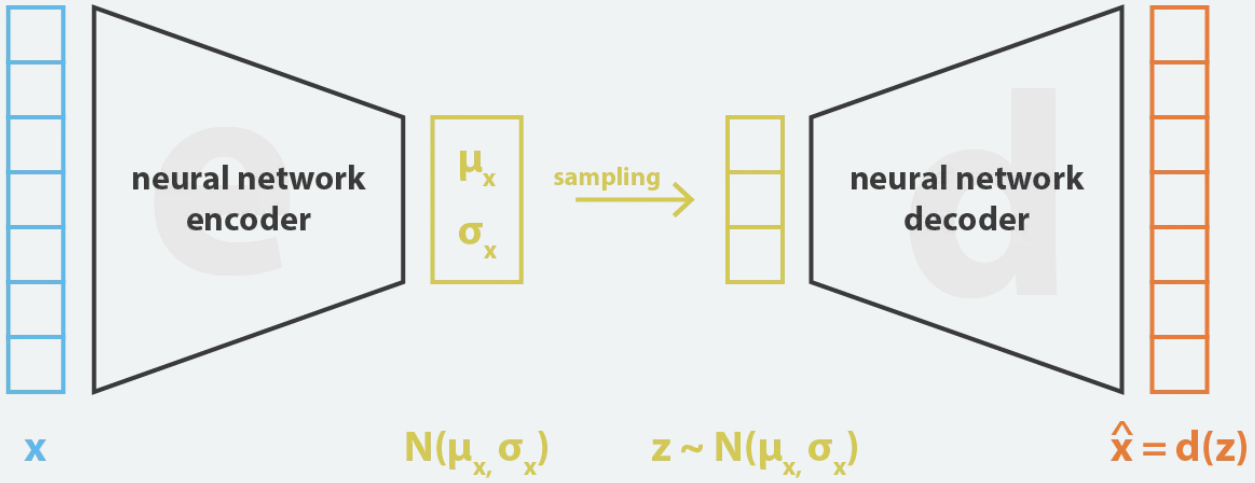
- Reduced to 10 dimensions, which were used for DBSCAN, but clustering did not capture the genera with high accuracy.
- Data is not linear

UMAP

- Reduced to 3 dimensions. Produces plots with clear "blobs".
- DBSCAN produces quite good clusters, with an accuracy of 80% of DBSCAN clusters labeled by the most common true value in that cluster.



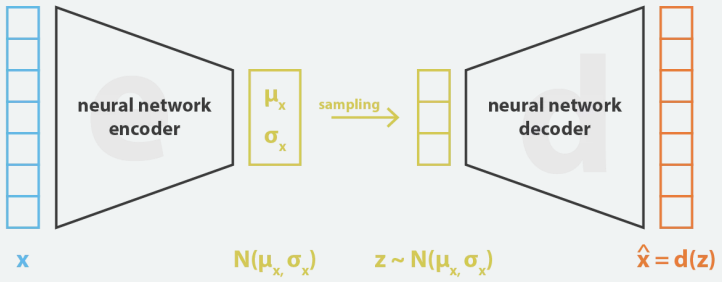
Creating the VAE

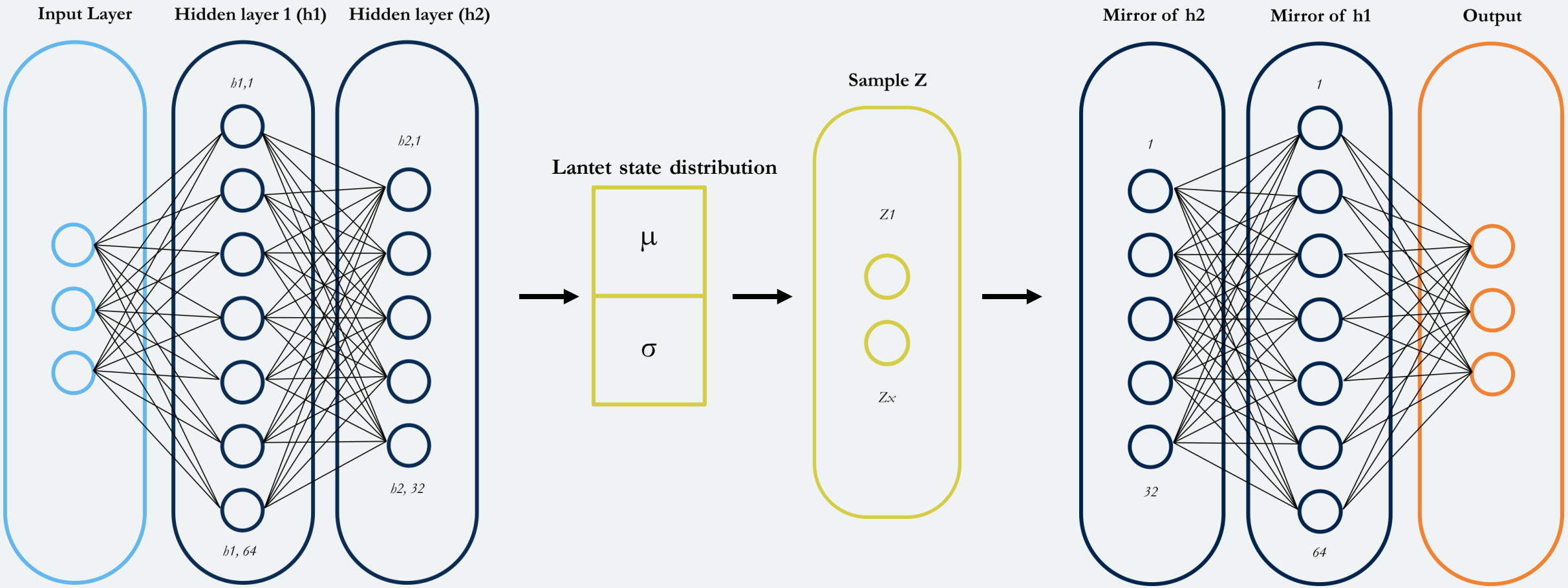


Loss = mean squared error (squared L2norm) + Kullback–Leibler divergence

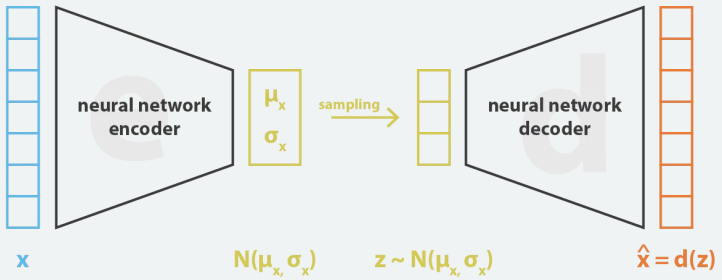
Activation function: rectified linear unit (ReLU)

Architecture of VAE





Training of VAE



Optimizer: Adam with a learning rate of 0.001.

Epochs = 500

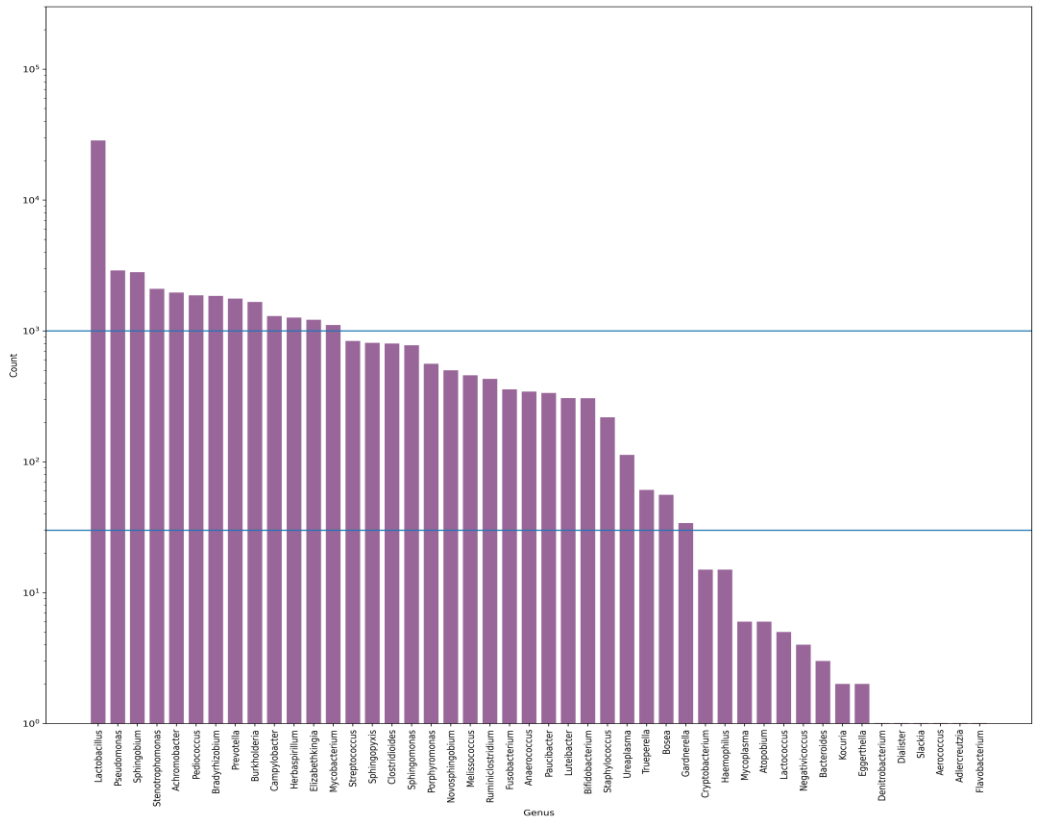
Multiple issues

Problem: The loss went to inf.
Solution: Batch Normalization (hidden layer two and latent space). Consider the hardware available and how the data is usually run (GPU cluster).

Problem: No converge
Solution: learning rate of 1e-3 to 1e-5 and increased epochs to 5000. Patience due to the two parts of the loss-function.

Problem: Model did not perform well on large dataset.
Solution: Data curation was not done correctly. Test your model toy data or small datasets to test if it's a data problem or a model problem.

Data balanced and unbalanced



Creating two models

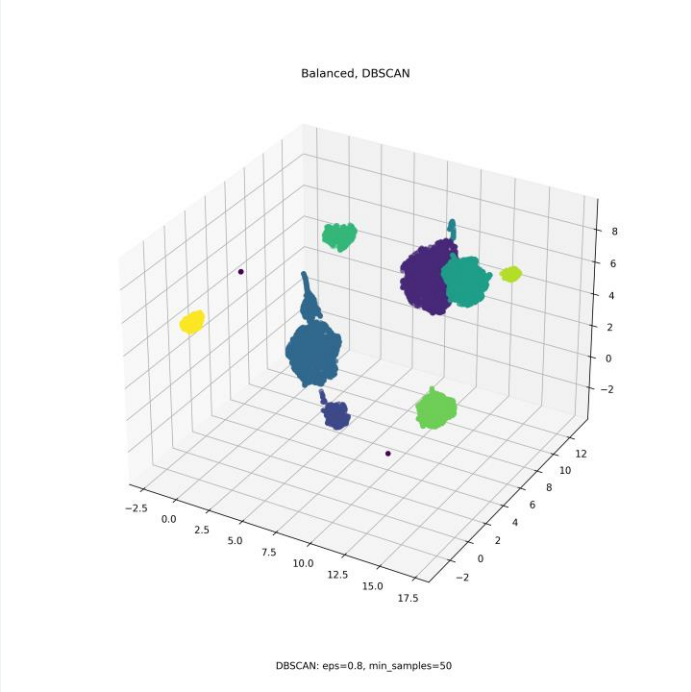
Balanced:

- ≥1000 contigs (13 genera)
- Sampled 1000 contigs per genera

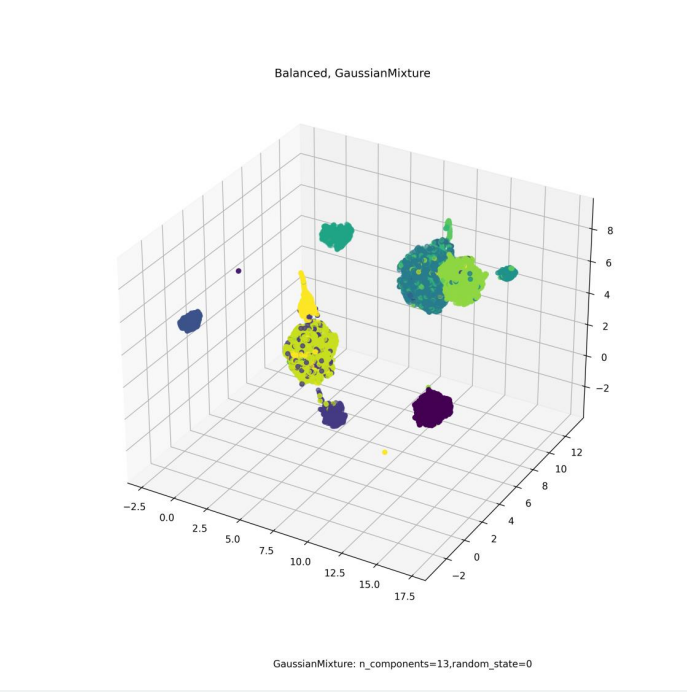
Unbalanced

- ≥30 contigs (31 genera)
- Included everything

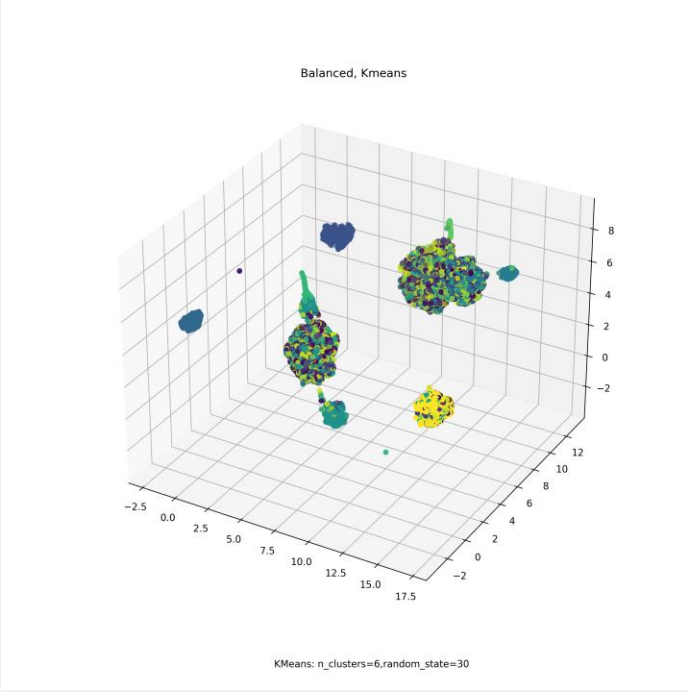
Balanced clustering & UMAP (z=8)



Homegeneity score: 16%

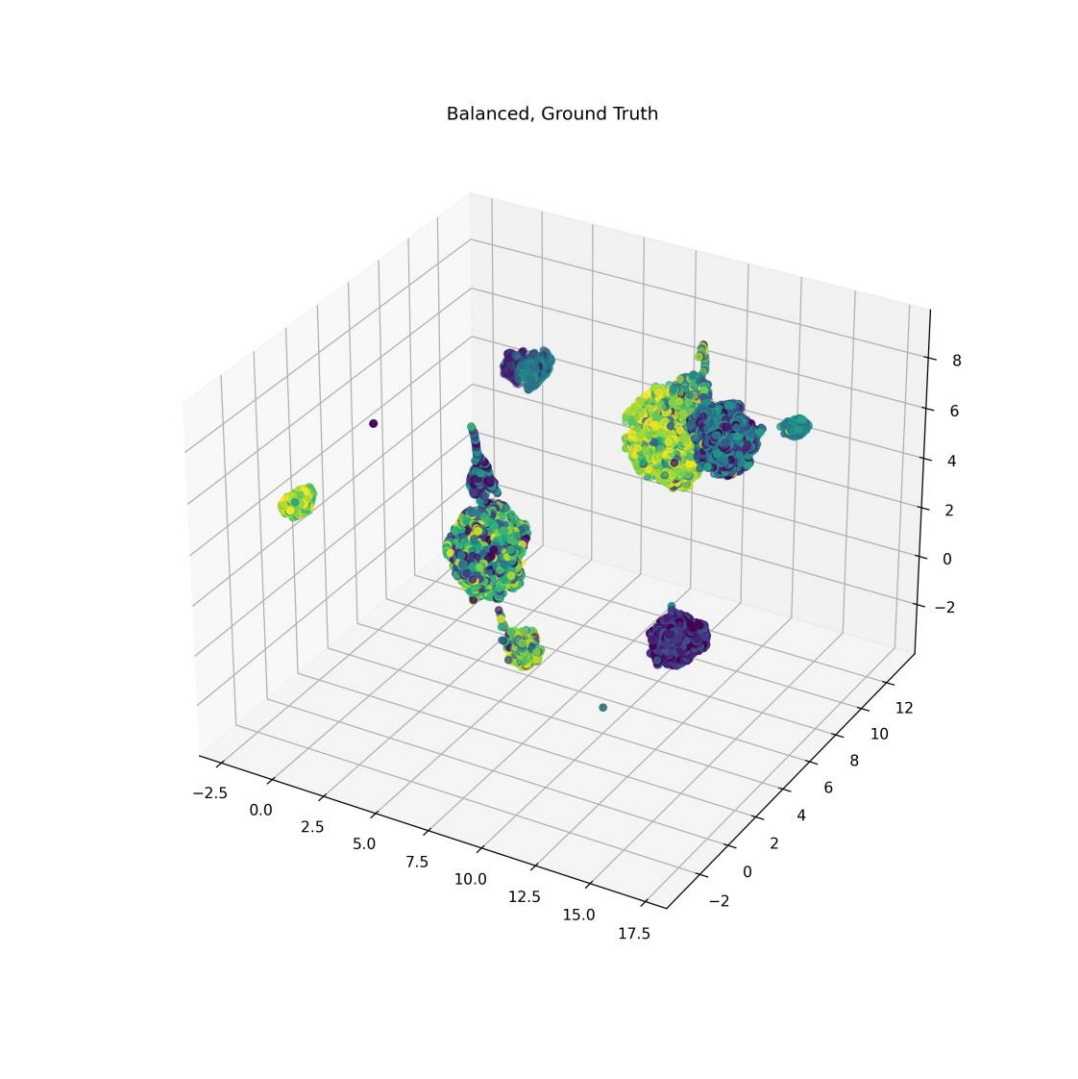


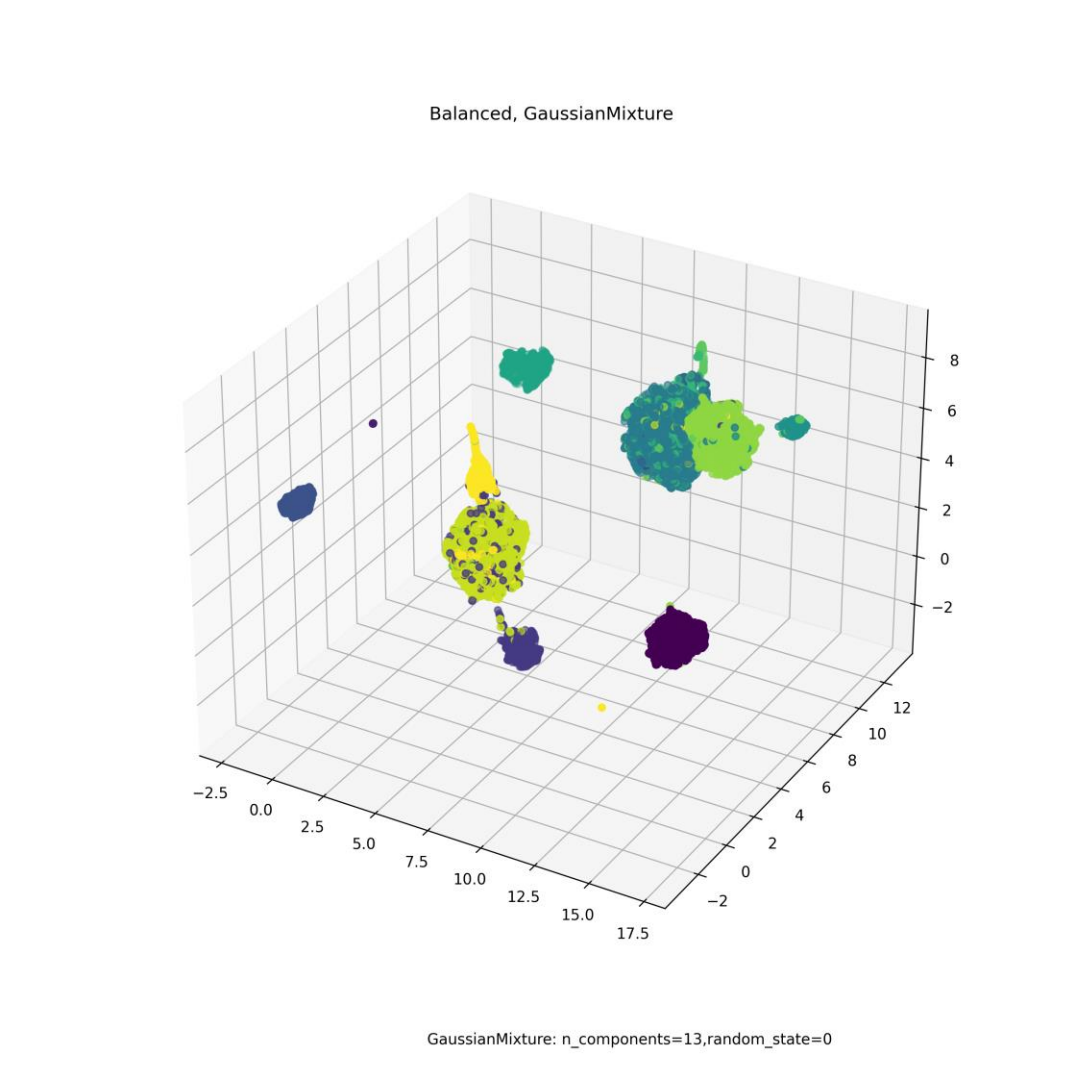
Homegeneity score: 23%



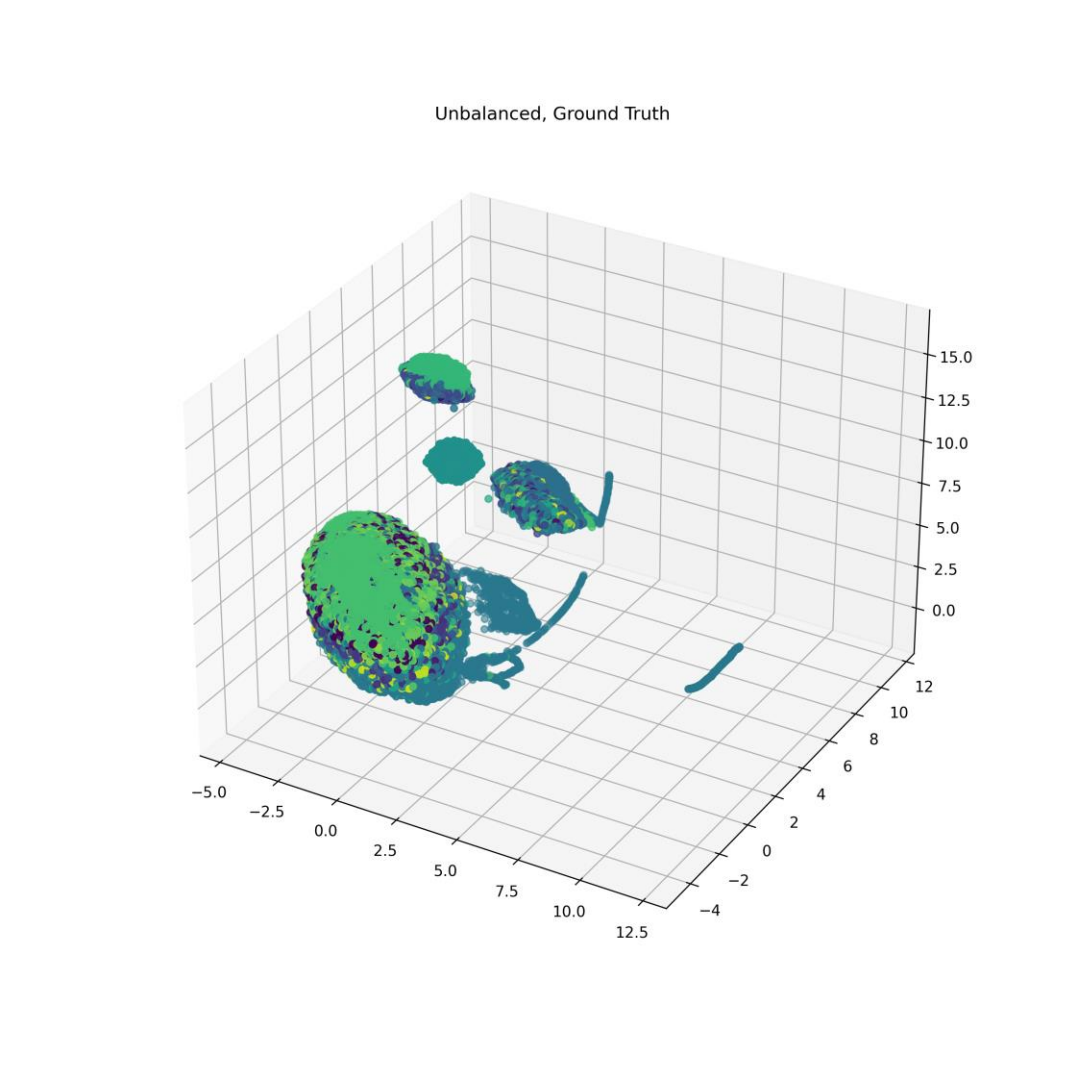
Homegeneity score: 3%

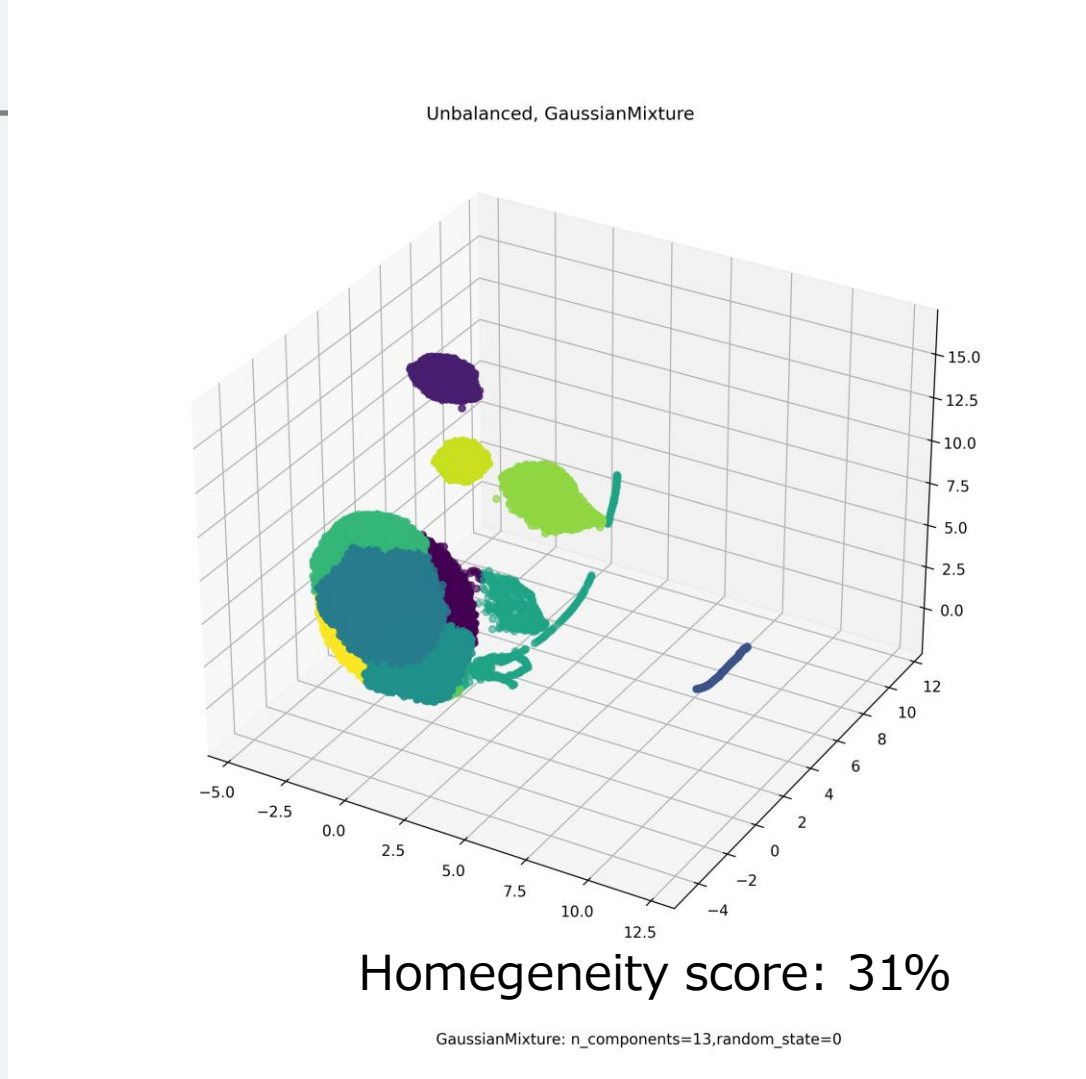
Balanced clustering & UMAP (z=8)





UNBalanced clustering & UMAP (z=8)





Homegeneity score: 31%

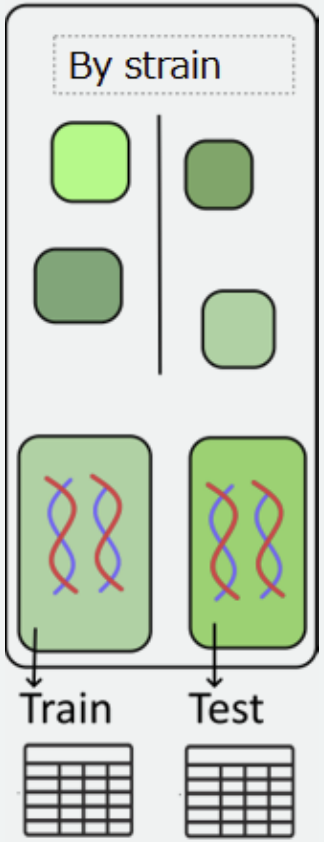
GaussianMixture: n_components=13,random_state=0

How well does it cluster?

Method dimensionality reduction	Method for clustering	Number of clusters	Number of correctly placed genera/species in clusters
PCA	DBSCAN	495	42% (Most of which was the same, huge cluster)
UMAP		619	82% (unbalanced data)
VAE (z=8) Balanced	DBSCAN	11 (Total genera: 13)	16%
VAE (z=8) Balanced	Gaussian Mixture	13 (predefined)	23%
VAE (z=8) Balanced	KMeans	13 (predefined)	3%
VAE (z=8) Unbalanced	DBSCAN	x	

Supervised: Decision trees

- Problem: Imbalanced classes
- LightGBM
- Bayesian hyperparameter optimization



Play around with parameters

Method	model	Learning_rate	Num_leaves	Depth	estimators	AUC
XGBoost	1	0.01	60	30	200	0.046
	2	0.01	60	40	200	0.046
	3	0.001	60	30	200	0.042
	4	0.004	56	30	100	0.043
LGBM	1	0.01	30	30	100	0.002
	2	0.004	56	30	100	0.22

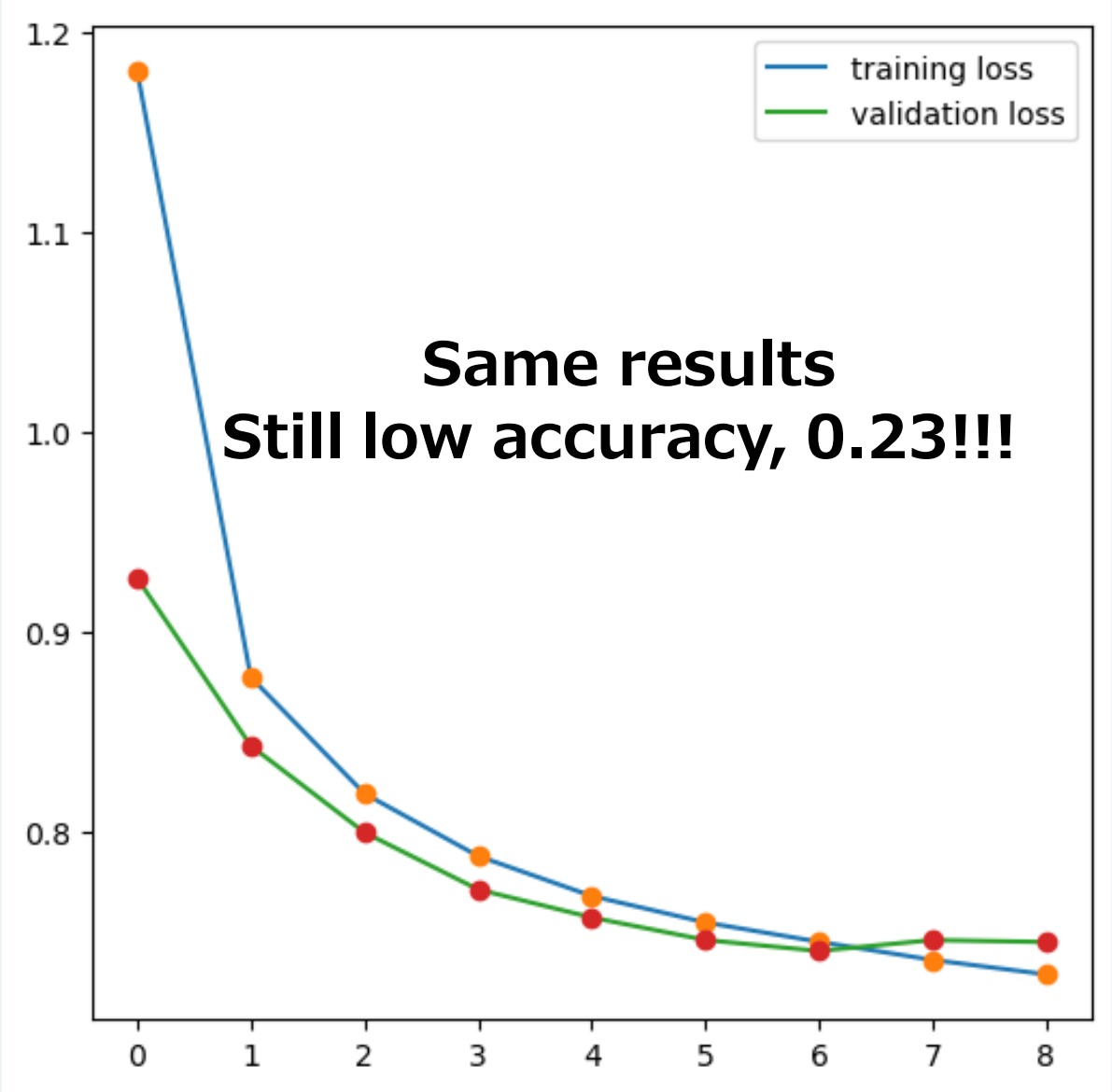
Problem: low accuracy

Top 3 AUC Scores	
Genus	AUC Score
Citrobacter	1.0
Campylobacter	0.97
Arcanobacterium	0.96

Worst 3 AUC Scores	
Genus	AUC Score
Croceibacter	0.46
Treponema	0.40
Tannerella	0.38

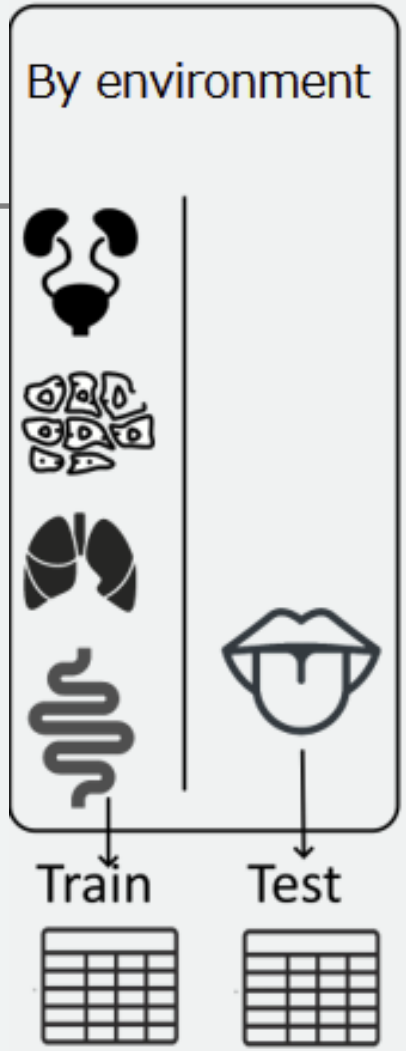
- Number of genus that we do not predict at all: 15

FFNN


1.2
1.1
1.0
0.9
0.8
0
1
2
3
4
5
6
7
8
training loss
validation loss
Same results
Still low accuracy, 0.23!!!

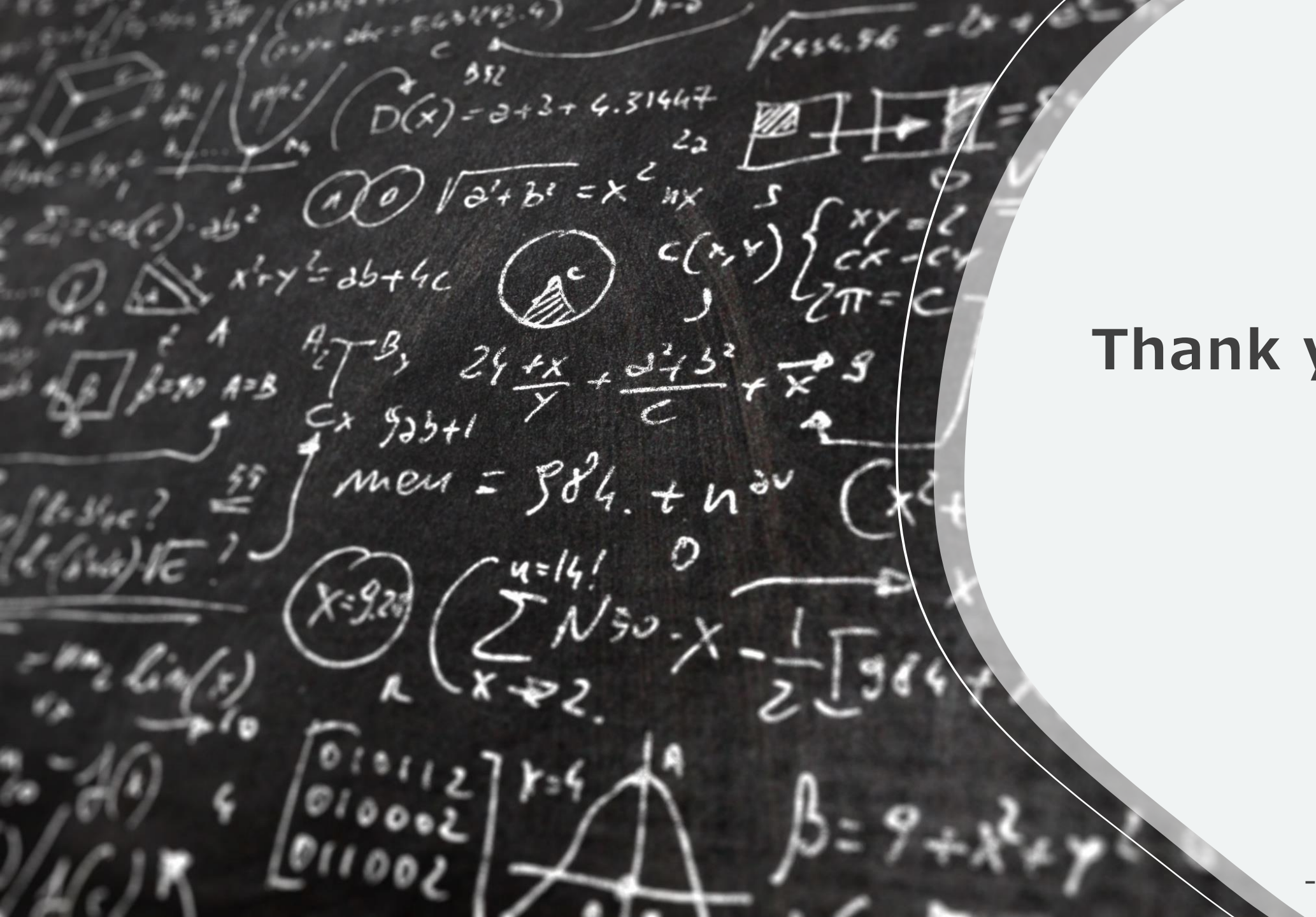
Can we predict taxonomy in a new dataset?

- Predictor: LightGBM multiclassification
- Hyperparameters (Optuna):
  - Num_leaves: 25
  - learning_rate: 0.01
- Results:
  - Species level: 35% accuracy
  - Genus level: 68% accuracy
- Problem: Is this even biologically relevant?



Conclusion

1. Unsupervised – Our current model is not sufficient in determining the genera of contigs and is therefore not applicable for metagenomic binning in its current form.
2. Supervised – Our model performs better than just a random guess and has been shown to predict species in an "unseen" environment. However, this model is not robust to generalize to new data and species and can only predict genera which it trained on.



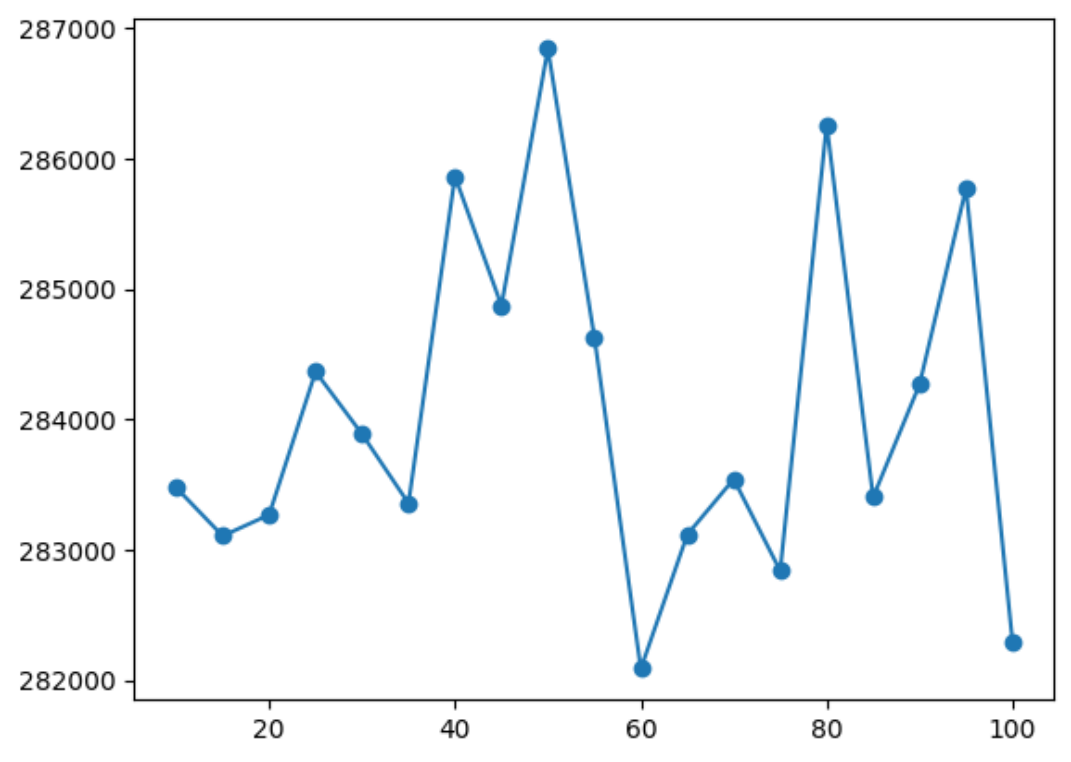
Thank you

--- Break slide ---

Supplementary

Searching for the optimal size of Z

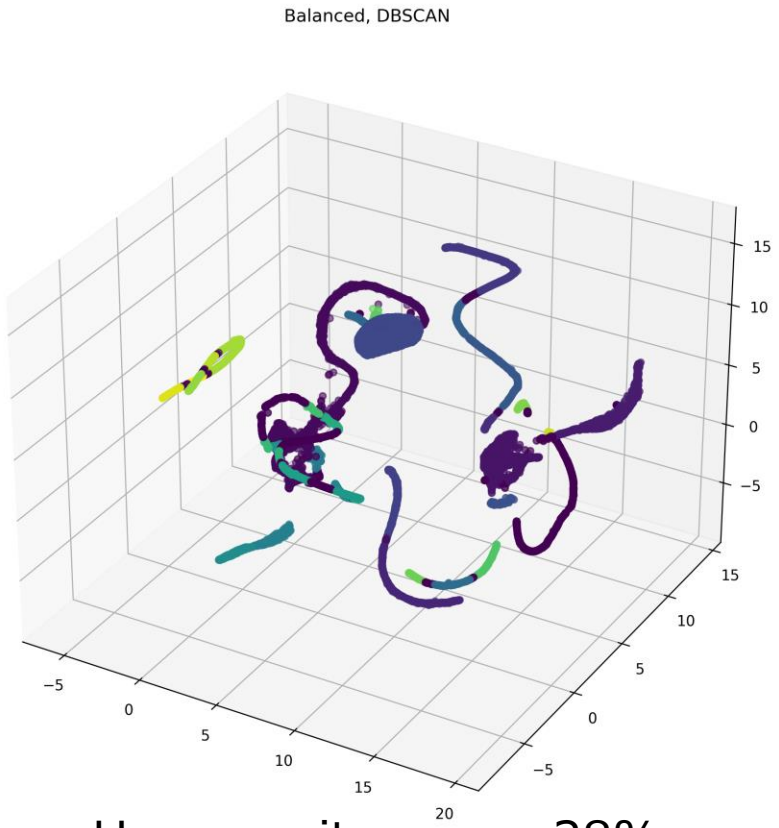
Using the “elbow method” finding the minimum loss (normalized)



Direct UMAP and clustering

Only finds 4 clusters mist of which are snake structures

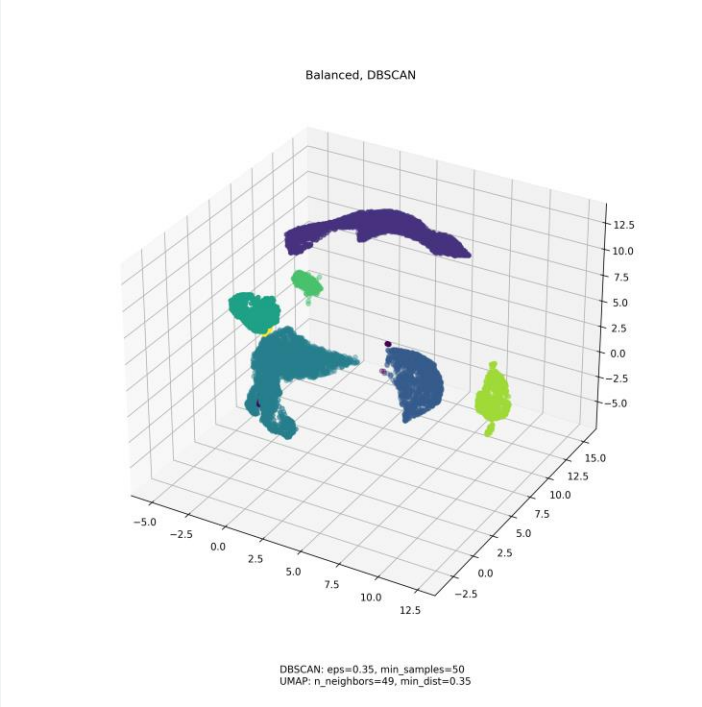
"Snakes" - It has been raised in the literature that it could be due to correlation

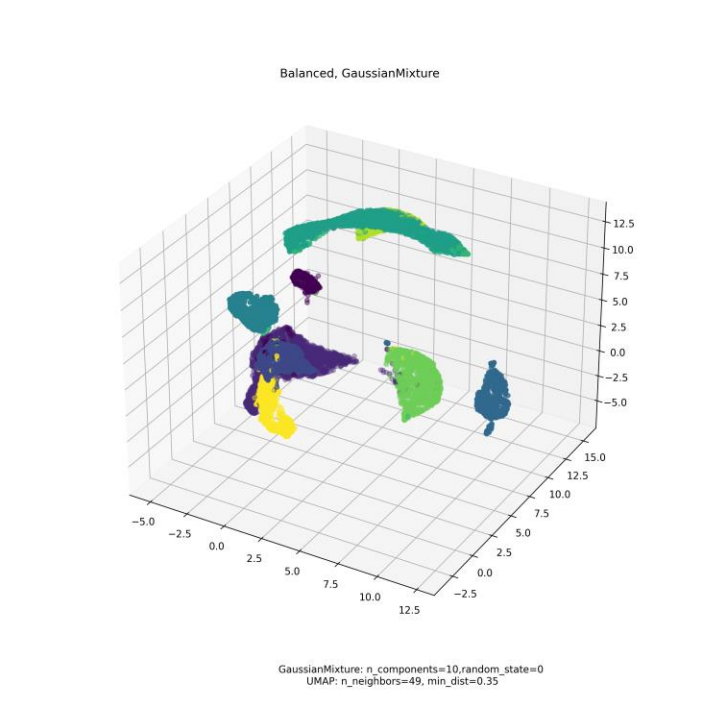


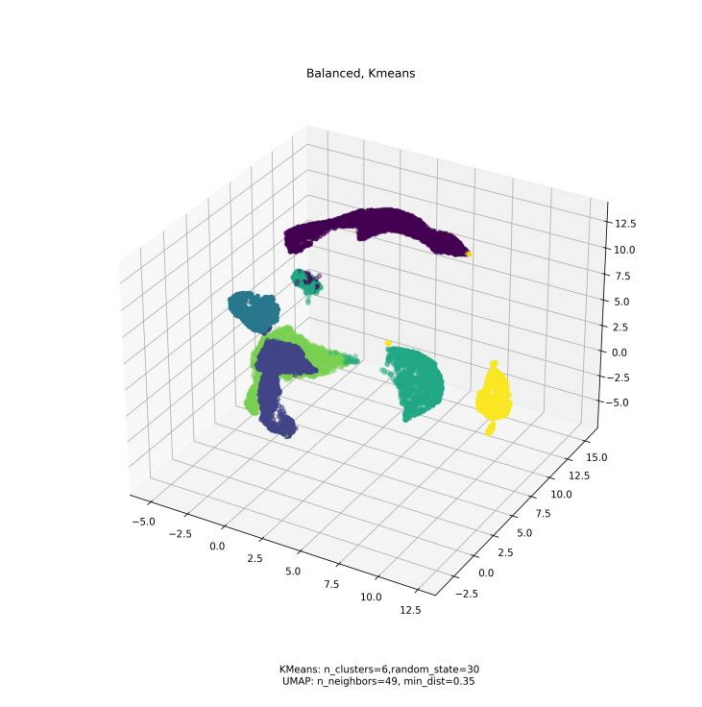
Homegneity score: 28%

DBSCAN: eps=0.8, min_samples=50

Balanced clustering (data) (z=8) Pytorch

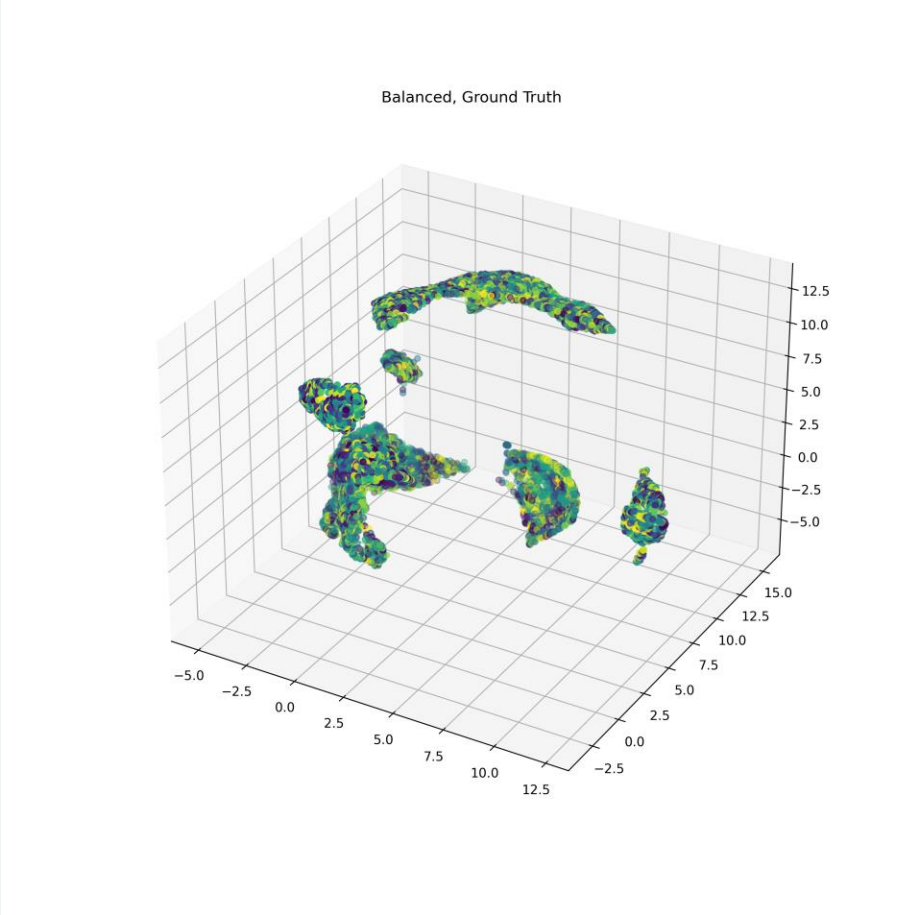






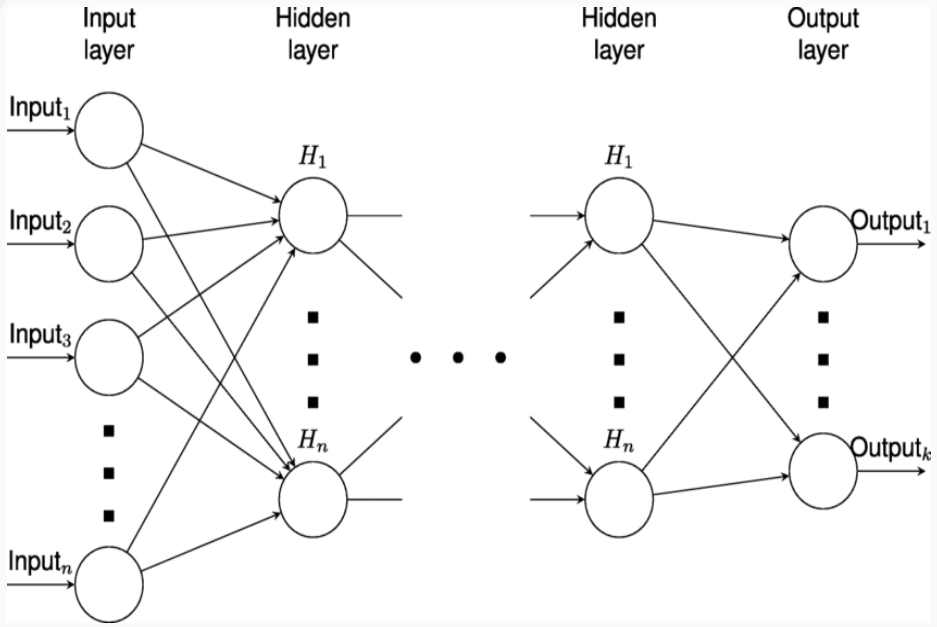
Homegeneity score: <1%

Balanced clustering (Pytorch)



FFNN

- TensorFlow and Keras
- Bayesian hyperparameter opt
  - learning_rate: 0.001
- Results: Baseline performance
- Problem: Imbalance classes



	Input_layer	Hidden_layer1	Hidden_layer2	Output_layer
Units	288	64	32	214
Activation	relu	relu	relu	sofmax