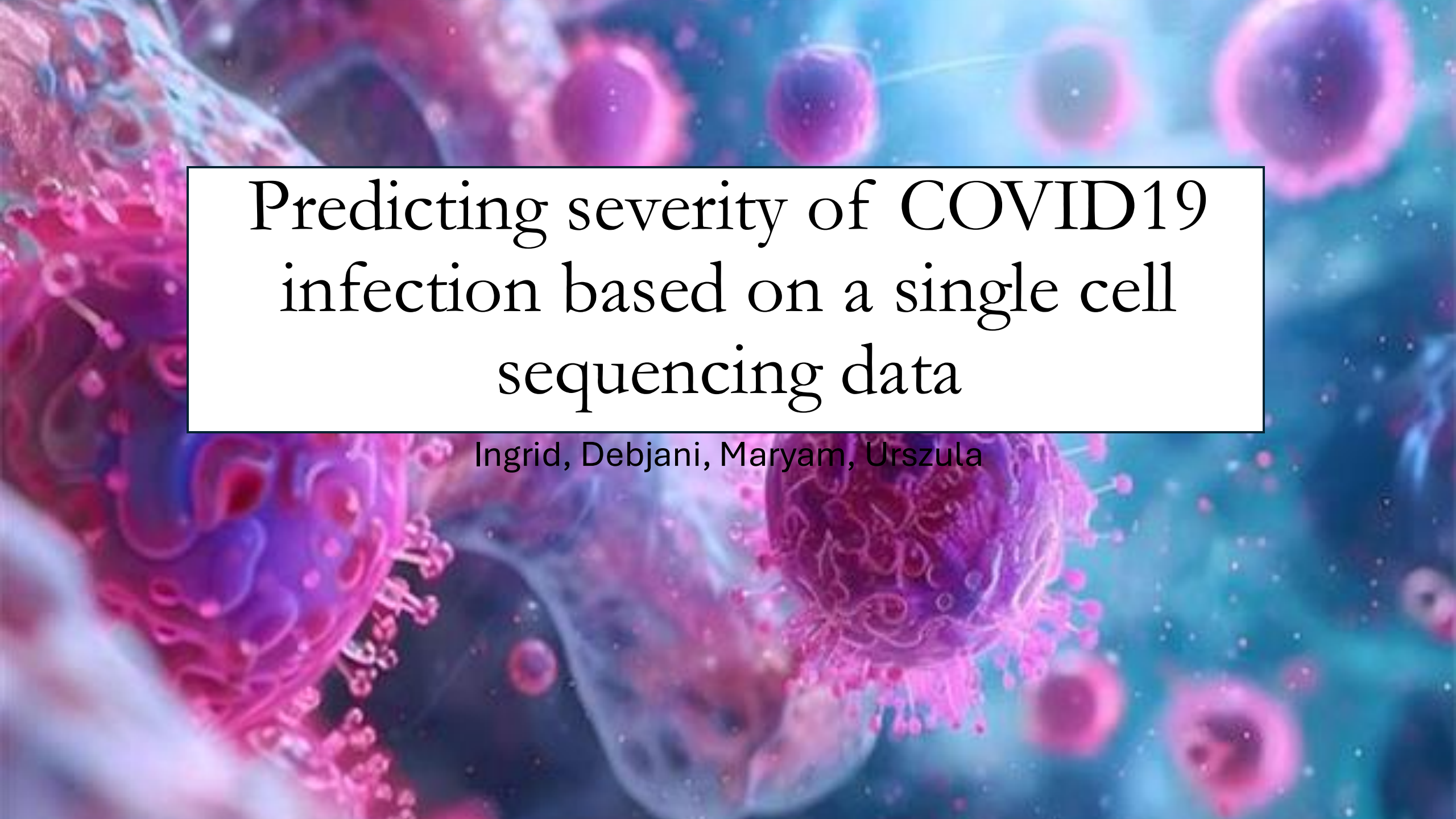


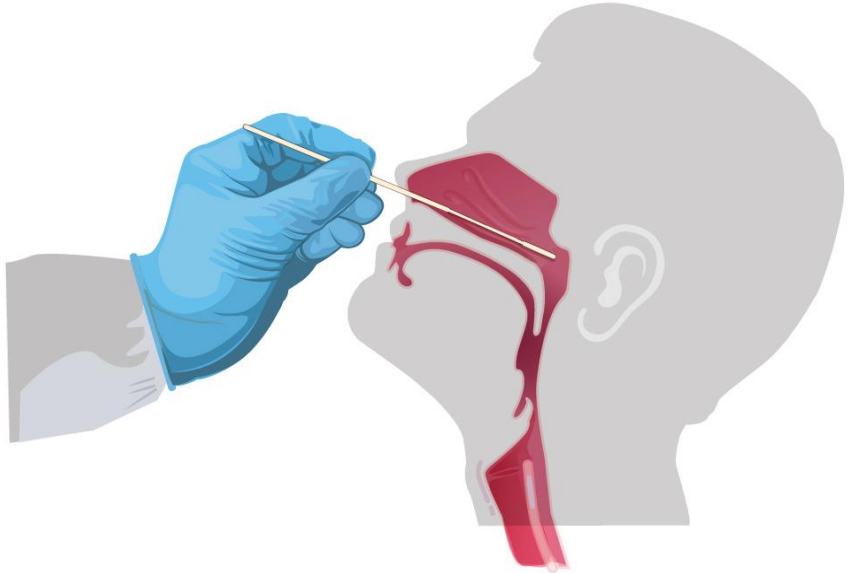
Predicting severity of COVID19 infection based on a single cell sequencing data

Ingrid, Debjani, Maryam, Urszula

WHO Covid Severity Score

0:	uninfected / healthy
1–3:	symptomatic, not hospitalized (mild)
4:	hospitalized, no oxygen required
5:	hospitalized, oxygen required
6:	high-flow oxygen or non-invasive ventilation
7:	invasive mechanical ventilation
8:	death

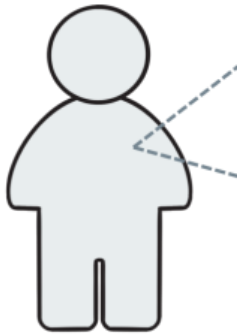
What makes COVID a severe disease for some and not others?



Metadata/clinical data vs. Genetic data

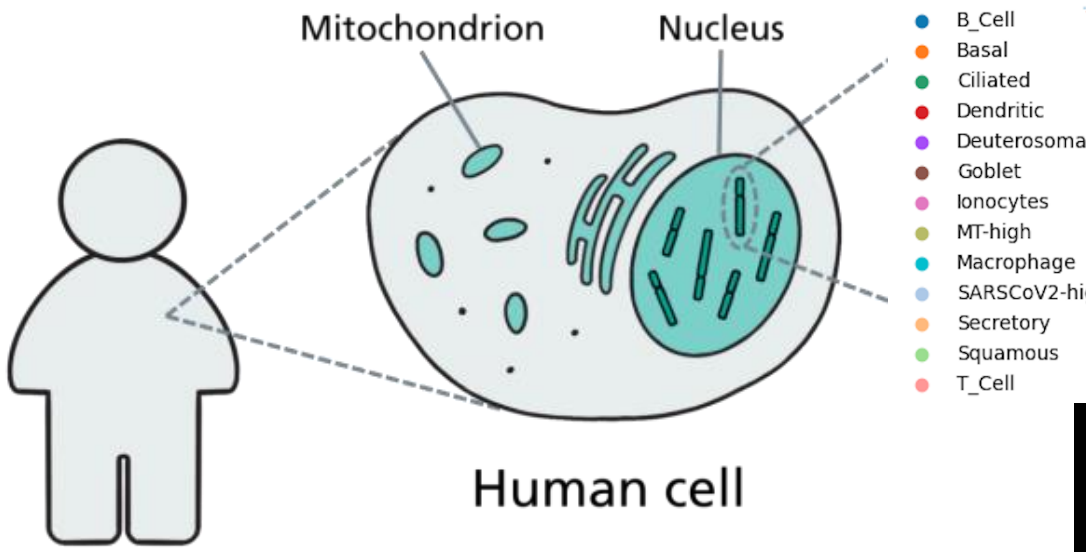
Metadata - patientdata

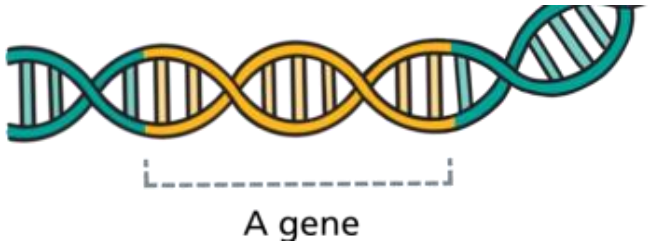
Participants



Gender	COVID status	COVID variation	Vaccination status	WHO serverity score	Diabetes	...
Male	Positive	Delta	no	2	no	
Female	Negative	Omicron	yes	8	no	
...	...					

Data





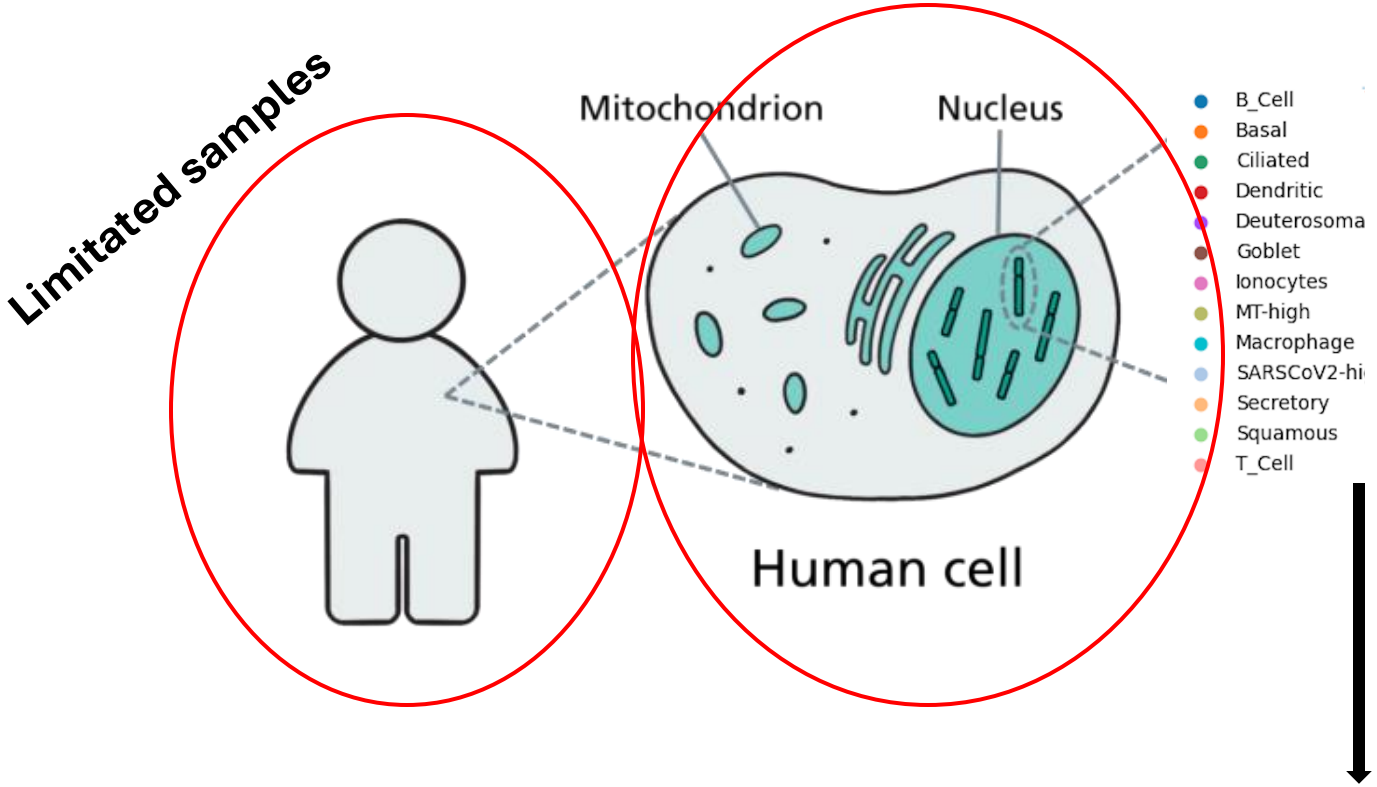
112 → 100 - 1750 → 13 kinds → 30.000

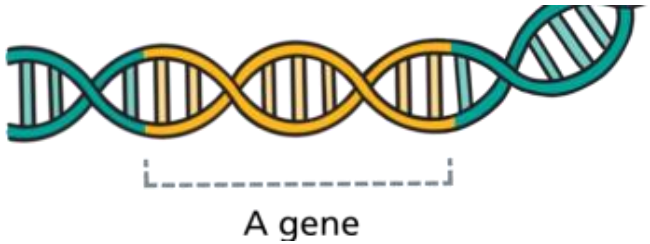
Health Education England Genomics Education Programme (n.d.) *Visual communication aid: genes, chromosomes and DNA*. Available at: https://www.genomicseducation.hee.nhs.uk/genotes/knowledge-hub/visual-communication-aid-genes-chromosomes-and-dna/

Data

Inherently Noisy!

Cells not independent

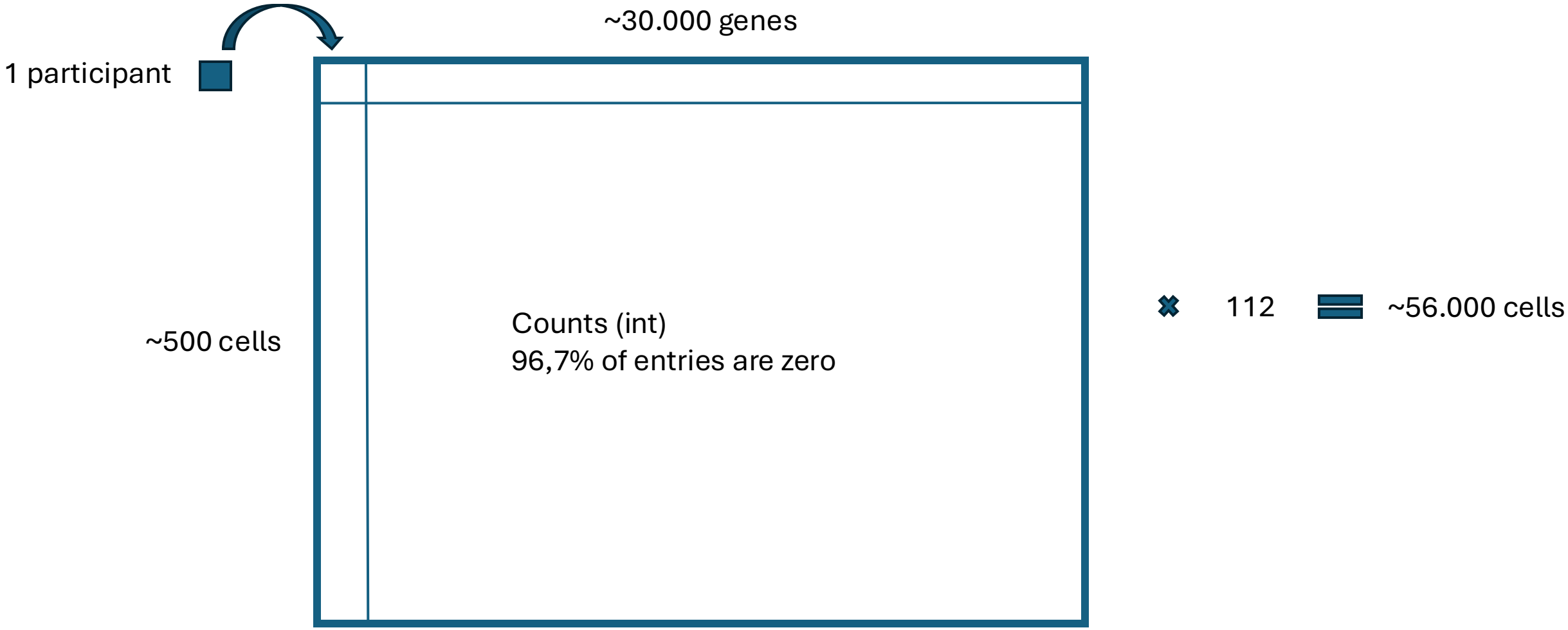




112 → 100 - 1750 → 13 kinds → 30.000

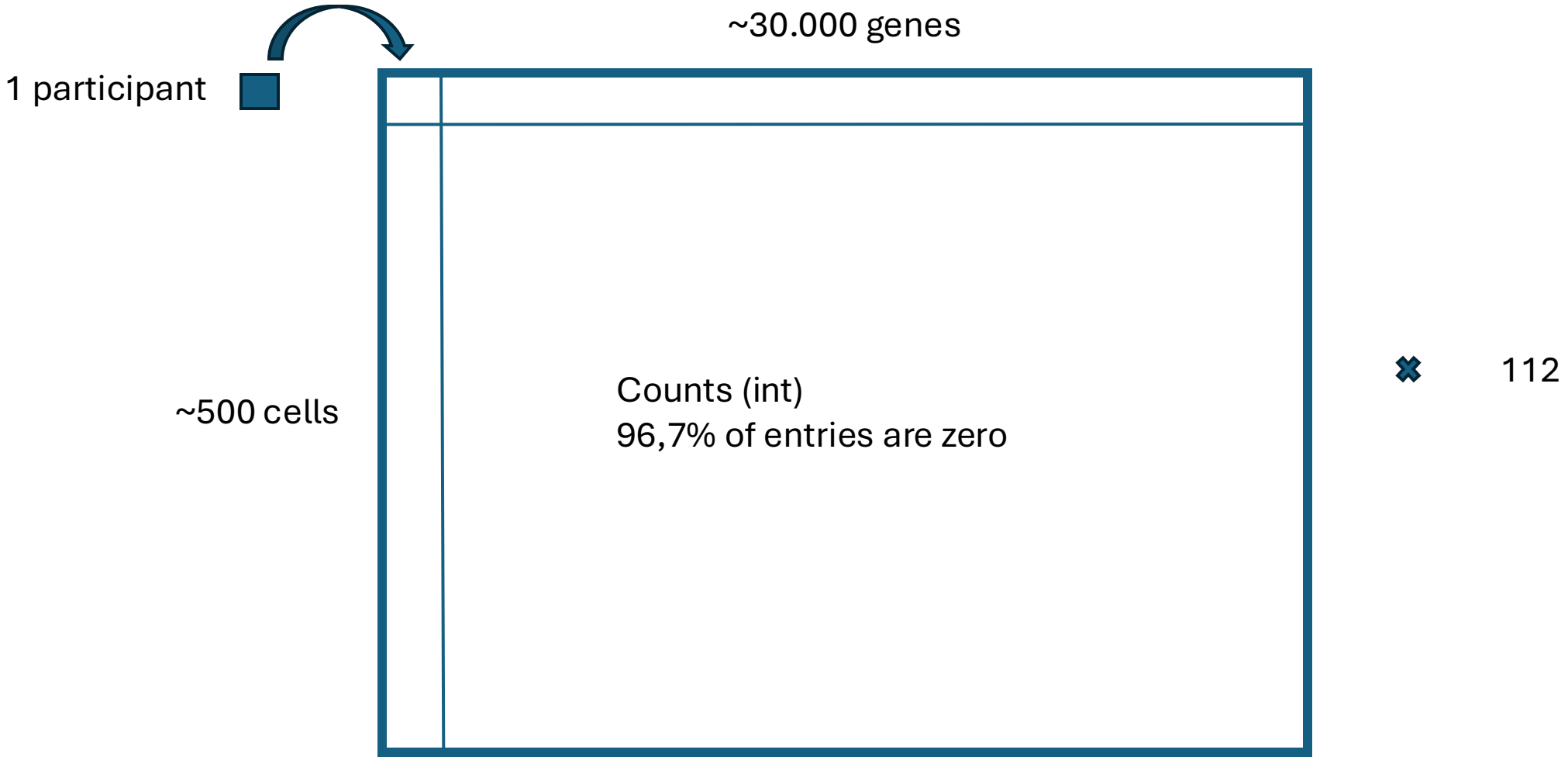
Health Education England Genomics Education Programme (n.d.) *Visual communication aid: genes, chromosomes and DNA*. Available at: https://www.genomicseducation.hee.nhs.uk/genotes/knowledge-hub/visual-communication-aid-genes-chromosomes-and-dna/

Data structure



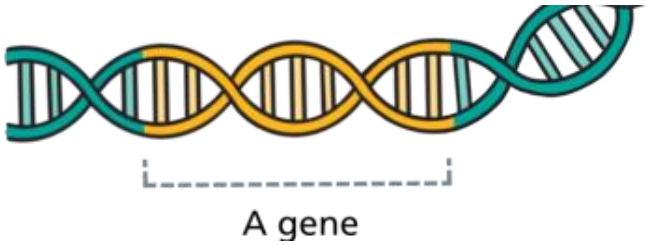
Data structure

Challenge:
Dimentionality Reduction / Feature Selection
Non-Independence & pseudobulking



Aim

- *<u>Detecting real genetic signal</u>* for further research and drug discovery
  - Cellular or genetic level
- *Not(!)* building a high-accuracy predition model



Exploration!

Feature Representation	Task	Model	Model Family
Tabular clinical features	Binary classification	Logistic regression	Linear
Tabular clinical features	Binary classification	Random forest	Tree-based
Tabular clinical features	Regression	Ridge regression	Linear
Tabular clinical features	Regression	Random forest regressor	Tree-based
Tabular clinical features	Clustering	K-means	Clustering
Tabular clinical features	Classification sensitivity check	Logistic regression+ SMOTE	Linear

Feature Representation	Task	Models	Model Family
Cell-type fractions (13 features)	Binary classification (severity)	RF / XGBoost / LightGBM (3x)	Tree-based
Secretory pseudobulk gene expression	Dimensionality reduction	PCA (50 PCs)	Linear
Secretory pseudobulk PCs (50 features)	Binary classification (severity)	RF / XGBoost / LightGBM (3x)	Tree-based
Secretory pseudobulk PCs (50 features)	3-class classification (variant)	RF / XGBoost / LightGBM (3x)	Tree-based
Composition + Secretory PCs (63 features)	Binary classification (severity)	Random Forest	Tree-based

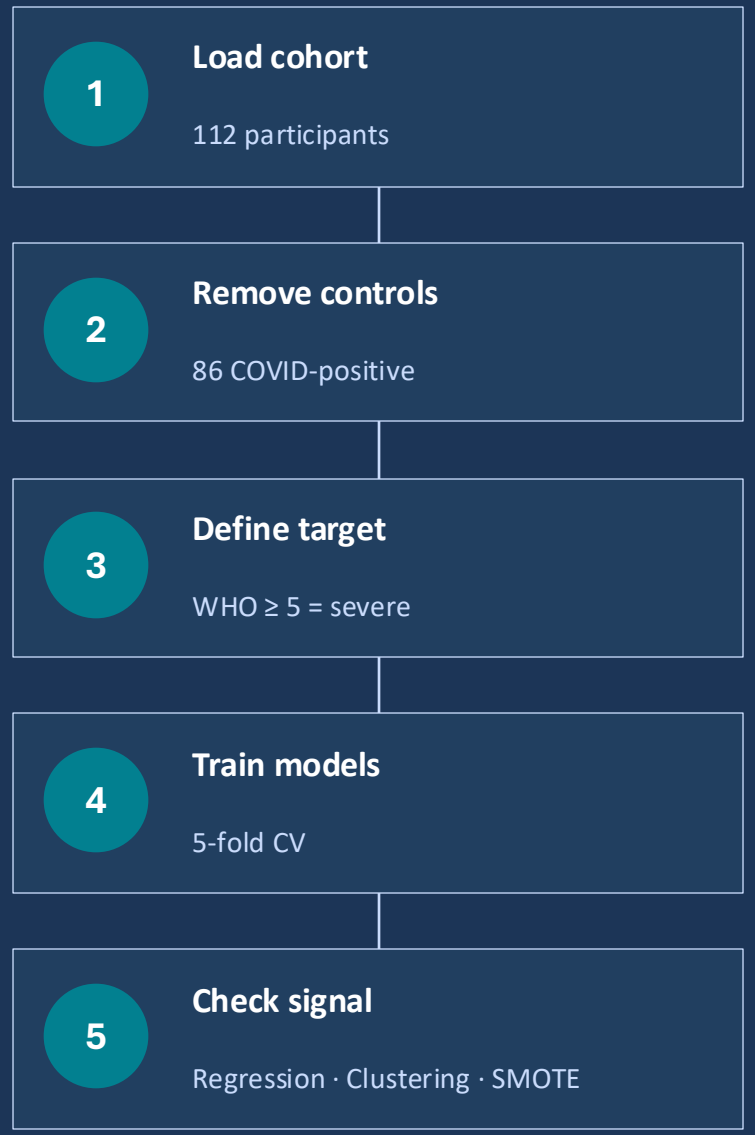
Feature Representation	Task	Model	Model Family
ScVI latent states	Clustering	Leiden + UMAP	Graph-based
ScVI latent states, (4x)	Classification	XGBoost	Tree-based
ScVI latent states (4x)	Regression	XGBoost	Tree-based
ScVI-VAE latent states (4x)	Classification	Tensorflow	MLP
ScVI-VAE latent states (4x)	Regression	Tensorflow	MLP
PyTorch vanilla autoencoder for GO terms (different fltering)	Classification	PyTorch	MLP

Patient Features & COVID-19 Severity

Small ML sub-test alongside the gene-level analysis

Purpose

Do basic clinical features carry any severity signal?



Data and Target

Controls removed — only COVID-positive patients used for training

112

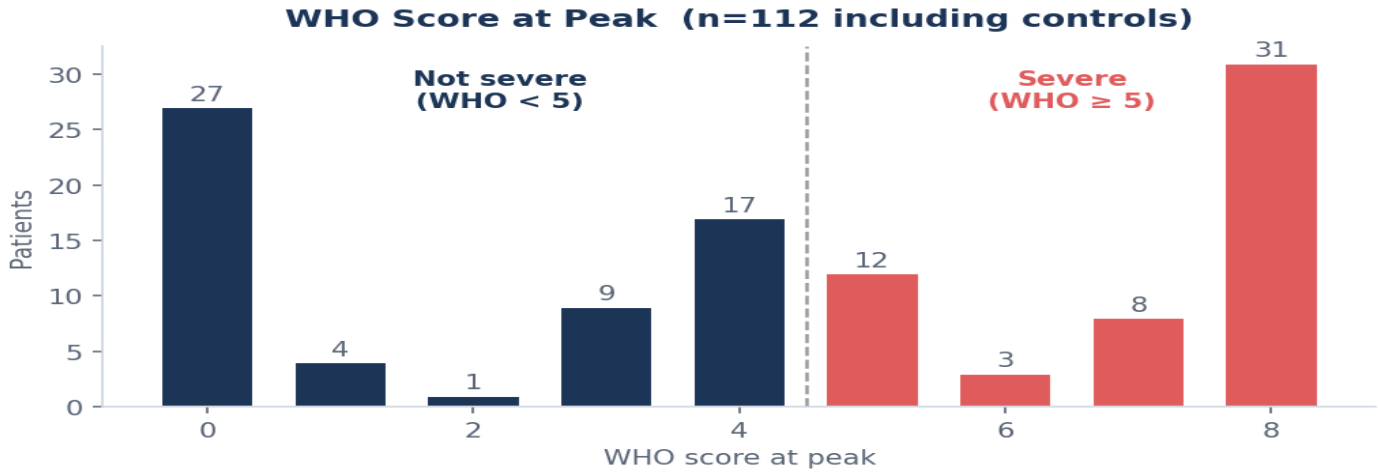
raw participants
(full cohort)

86

COVID-positive
(controls removed)

54 / 32

severe vs not severe
(WHO ≥ 5 vs < 5)



Features used

Numeric (12)

Age, BMI, WBC, Platelets,
D-Dimer, Ferritin,
Diabetes, HTN, CKD,
Heart/Lung disease,
Vaccine doses

Categorical (3)

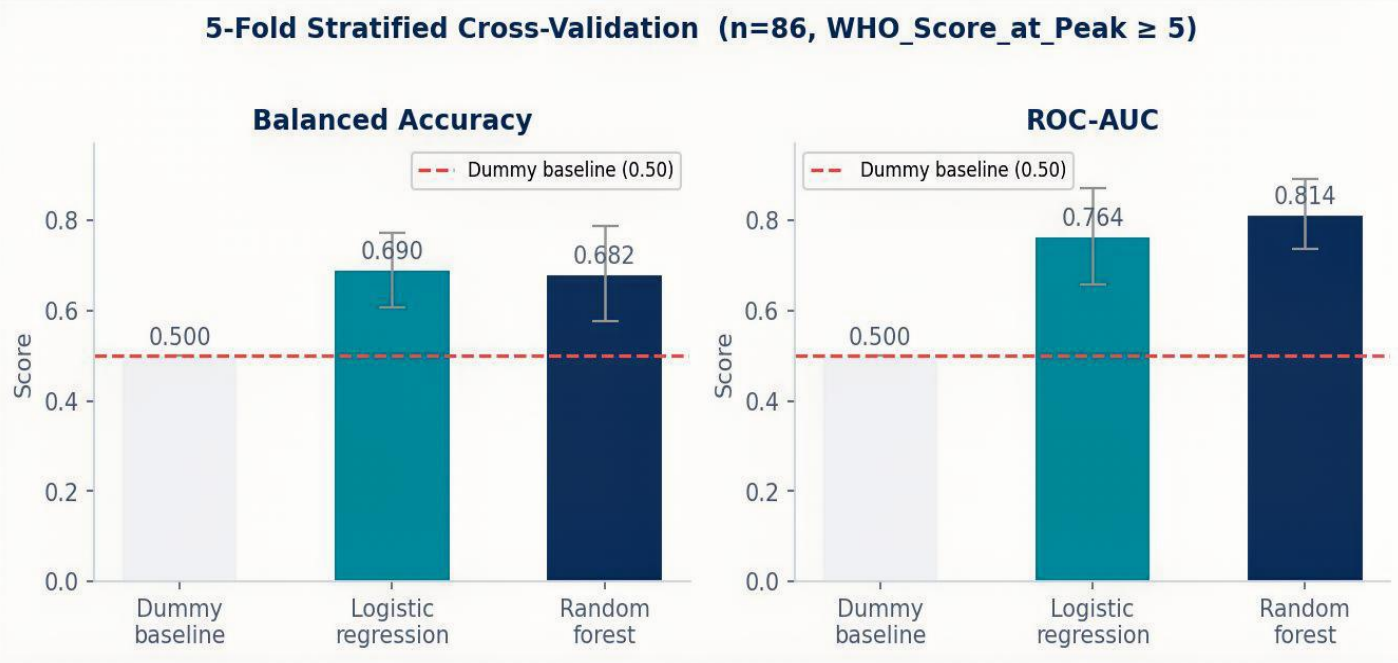
Sex
Race/Ethnicity
Vaccine Status

Excluded

Variant Group
insignificant at primary level
missing data features

Classification Results

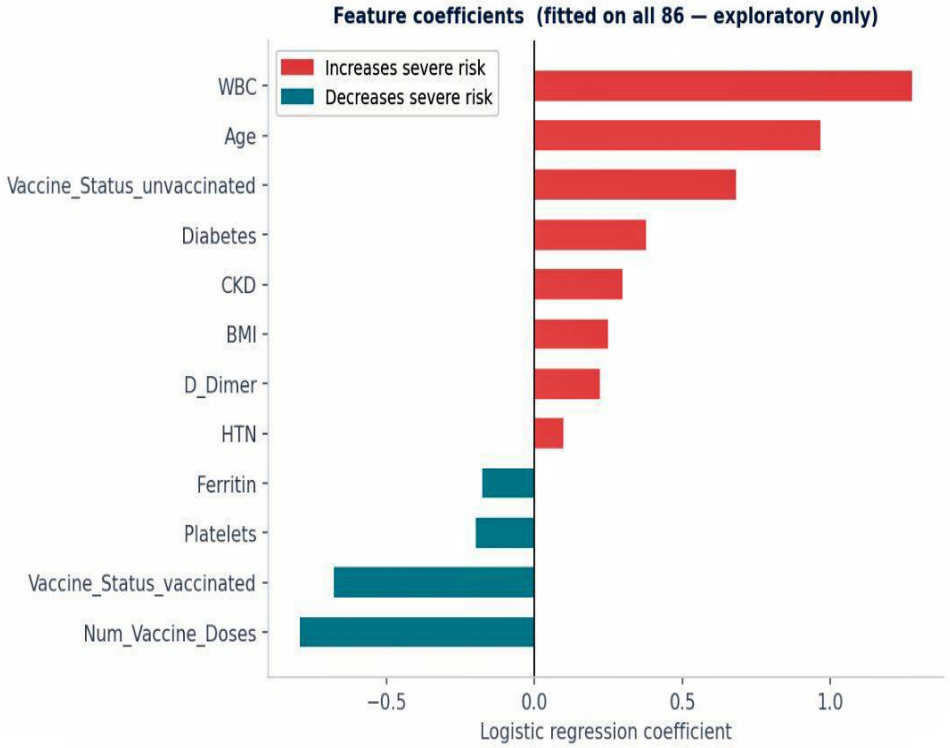
5-fold stratified cross-validation — WHO_Score_at_Peak ≥ 5 = severe (n=86)



Signal found: Both models beat the dummy. Random forest AUC 0.814 vs 0.500. Large ± values (±0.08–0.10) confirm results are unstable at n=86

What the model used

Logistic regression coefficients (fitted on all 86 - exploratory)



Secondary Checks

Regression · Clustering · SMOTE — all using the same 86-patient cohort

Check	Result (notebook)	What it means
Regression (WHO score)	R^2=0.330 MAE≈1.51 WHO pts	Some signal — better than mean baseline
Clustering (K-Means)	Best ARI=0.087 (k=2)	No clear severity groups in features
SMOTE augmentation	AUC 0.764 → 0.746	No gain — inside CV folds only

Clustering result: ARI ≈ 0.09 means patient features do not form groups that match severity. Severity is not a simple patient-type difference

SMOTE: applied inside training folds only. Test set always = real patients. No meaningful gain - synthetic rows are not real patients.

Limitations and Conclusion

Honest result — exploratory signal, not a deployable model

Small sample (n=86)

Results shift with a few patients - large ±std confirms this

No external validation

CV tests inside the same small cohort only

Prediction ≠ causation

Coefficients do not prove biological mechanism

SMOTE ≠ real patients

Synthetic rows used only as sensitivity check

Classes are imbalanced

54 severe vs 32 not severe - class_weight used throughout

Missing clinical checks

No calibration curve or subgroup fairness testing

Conclusion: Prediction possible with massive amount of data

Cell Composition

- **Can the proportions of nasal cell types predict COVID-19 severity?**

SEVERITY LABEL (WHO SCORE AT PEAK)

Not severe: 32 (WHO < 5)

Severe: 54 (WHO ≥ 5)

Imbalanced → class_weight="balanced" in RF

	B_Cell	Basal	Ciliated	...					
A_COVID19_01	0	0.006	0.417						
A_COVID19_02	0	0	0.460						
...									

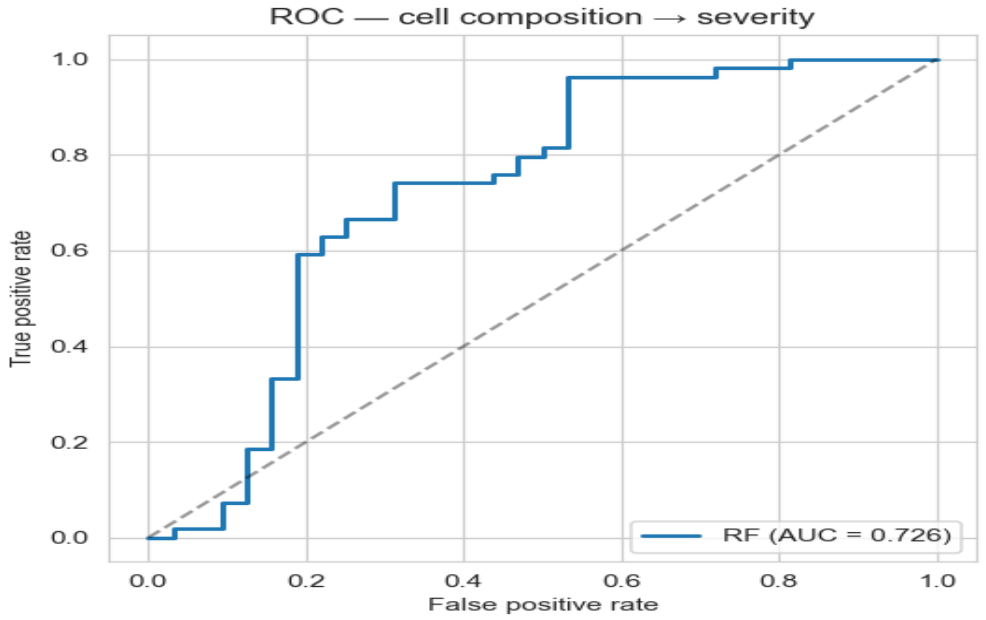
Train Random Forest with 5-fold cross-validation

- Random Forest: 500 trees
- max_depth = 5
- class_weight = balanced
- 5-fold StratifiedKFold
- Pooled AUC + bootstrap 95% CI (n=1000) : `single fold AUCs are noisy. A bootstrap CI on the pooled cross-validated predictions is more honest to report.`

  20-seed stability check, also tested LightGBM & XGBoost

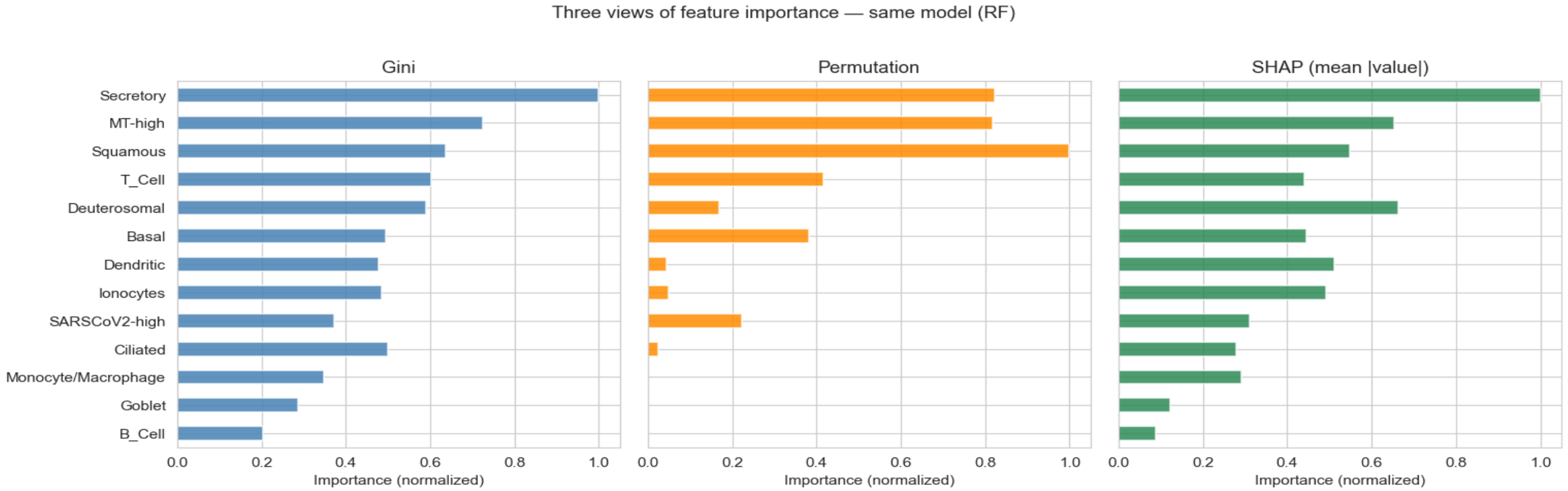
Pooled CV AUC	Bootstrap 95% CI	20-seed mean AUC	Chance baseline
0.726	0.586 – 0.847	0.730	0.500
5-fold, pooled	n=1000 resamples	± 0.024 (stable)	random classifier

```
5-fold CV AUC: 0.717 ± 0.221
Per-fold: [0.87  0.803 0.318 0.652 0.943]
```



Fold AUCs ranged 0.318 – 0.943 (high variance). The 20-seed mean (0.730 ± 0.024) confirms the signal is real, but n=86 makes individual fold estimates noisy.

Secretory fraction is the strongest predictor



Key finding: Secretory fraction is top predictor (Secretory is the top predictor by consensus across all three methods) This motivates Step 2: what are the gene expression patterns within Secretory cells that drive this signal?

Pseudobulk of Secratory Cells→ Normalize → 2,000 HVGs → 50 PCA components

6,565

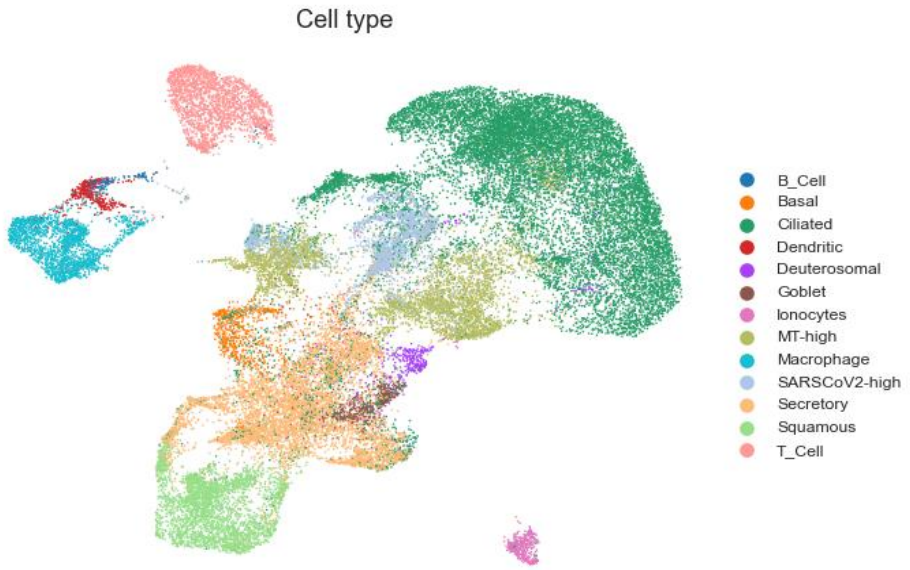
Secretory cells
from 86 patients

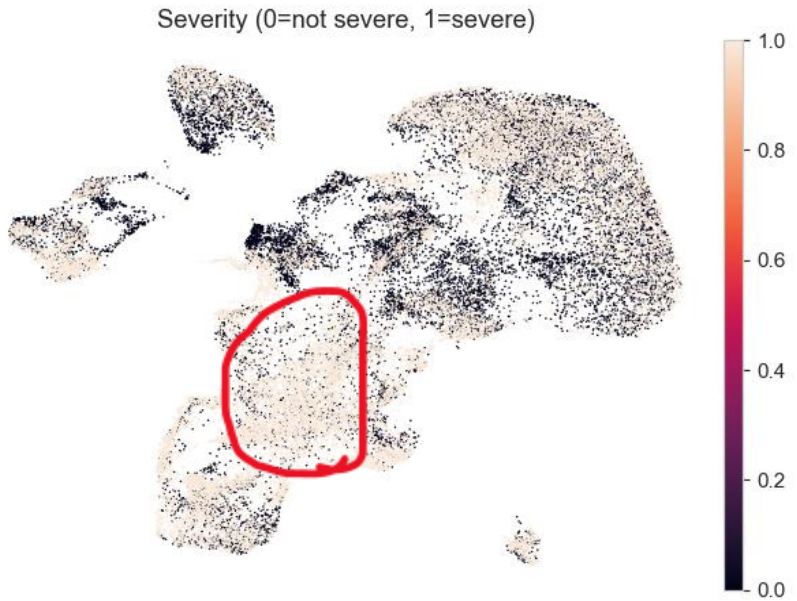
25/43

not-severe / severe

68

After ≥10 cell threshold
68
patients kept



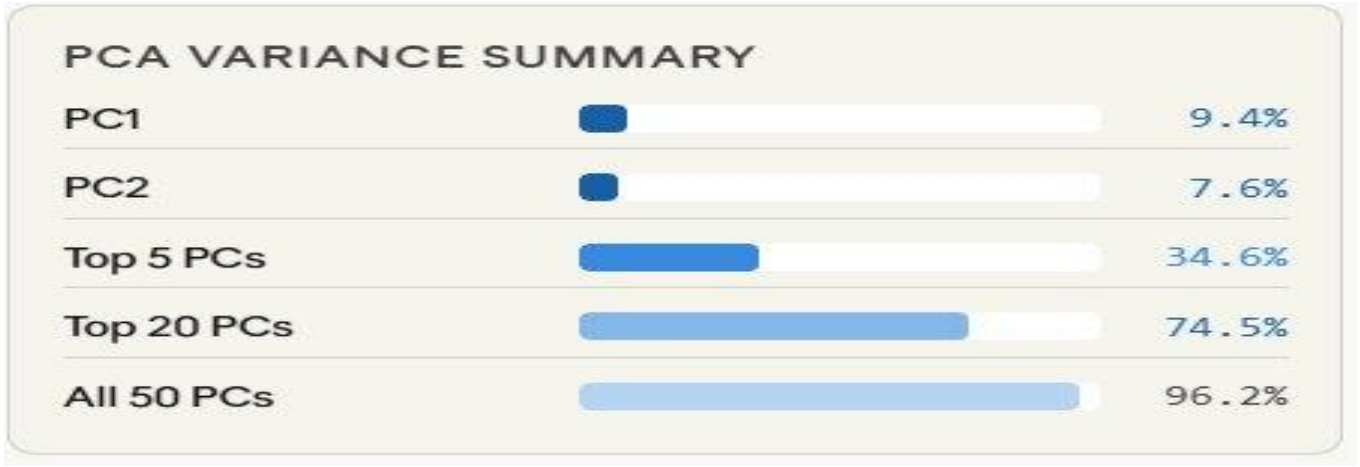


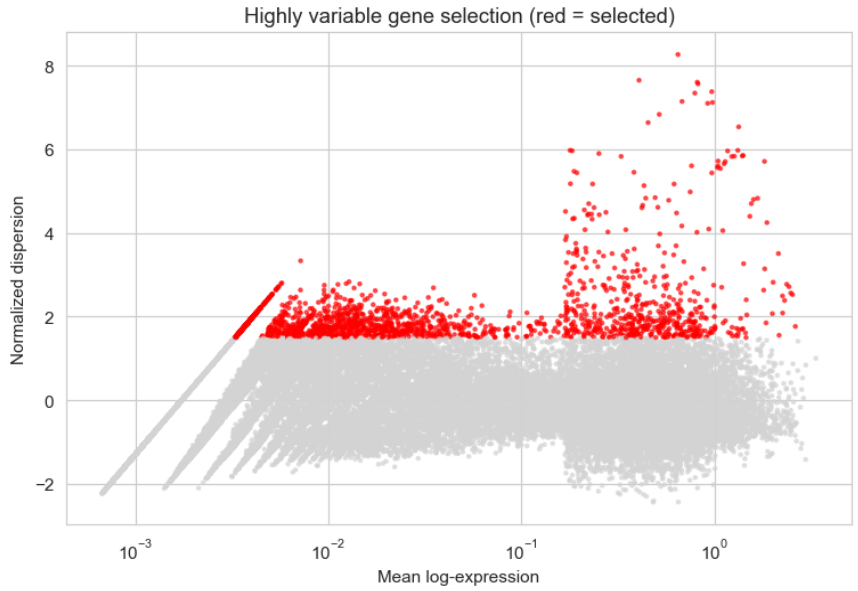
Key Challenge: ~500 cells per patient, all sharing one patient. Treating them as independent would be data leakage. Solution: pseudobulk (average across cells per patient → one profile per patient)

From cell proportion to gene expression: zooming into Secretory cells

Does the gene expression pattern within Secretory cells carry severity signal, beyond just their proportion?

1- **Pseudobulk:** average log-normalized expression across all Secretory cells per patient → 68 × 29,961 matrix
2- **HVG selection:** 2,000 most variable genes from 29,961 → reduces noise from uninformative genes
3- **Standardize:** z-score per gene (zero mean, unit variance) before PCA
4- **PCA:** 50 components → 68 patients × 50 features for ML

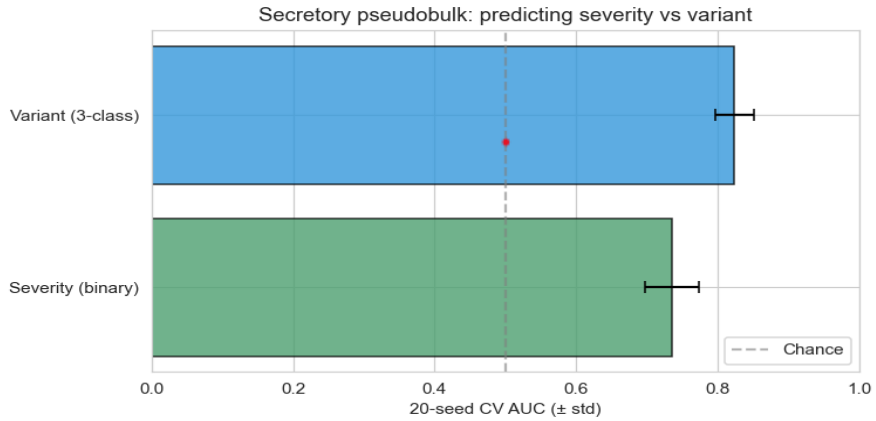




Key Note: The top HVGs are SARSCoV-2 viral RNA genes (SARSCoV2-E, SARSCoV2-ORF8, SARSCoV2-M...). PC1 likely encodes viral load, which differs strongly between variants → inflates variant-prediction AUC. Details in appendix

Gene expression adds signal; but variant is more predictable than severity!

```
=== Severity prediction: feature set comparison ===
                  Feature set      AUC      Std
Composition only (matched n) 0.704306 0.032535
          Secretory PCs only 0.734389 0.037774
 Composition + Secretory PCs 0.752722 0.031152
```



Key Note: Secretory gene expression contains real severity signal (AUC 0.734), adding value over cell composition alone. Variant is more strongly encoded (AUC 0.823), consistent with the paper's finding that severity signatures are variant-conditional , stronger in Ancestral/Delta, absent in Omicron.

TOP SEVERITY-RELEVANT PCS (FOR FURTHER INVESTIGATION)

PC14: IL2RA+

PC27: TOX2+

PC2 : GSTM5+, TLR8–

Specific gene sets require deeper biological interpretation

ScVI: VAE autoencoder

Build for this type of biological data!

Negative binomial distribution (as opposed to gaussian)

- Counts with many zeros

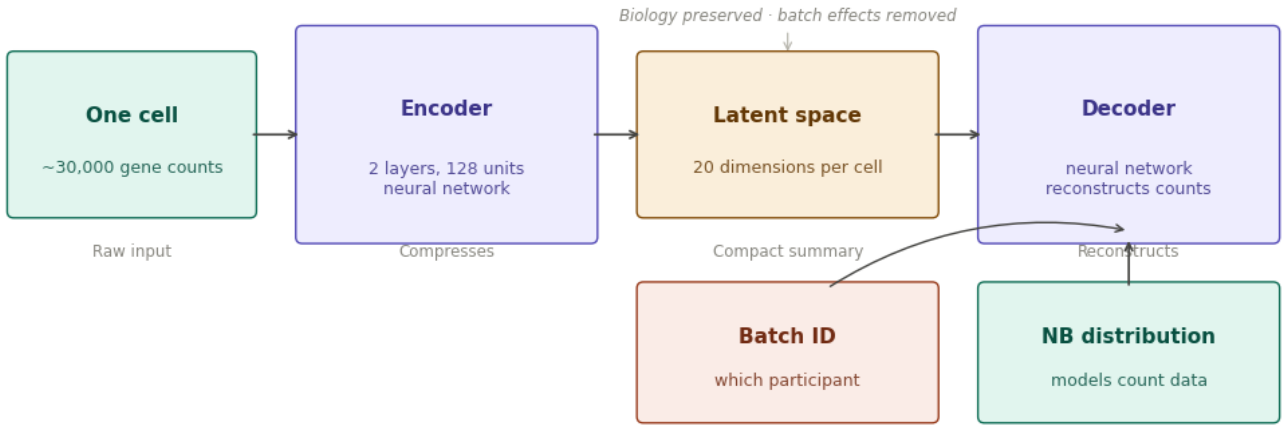
Integrated Batch Correction:

- Patient level

ELBO evaluation metric

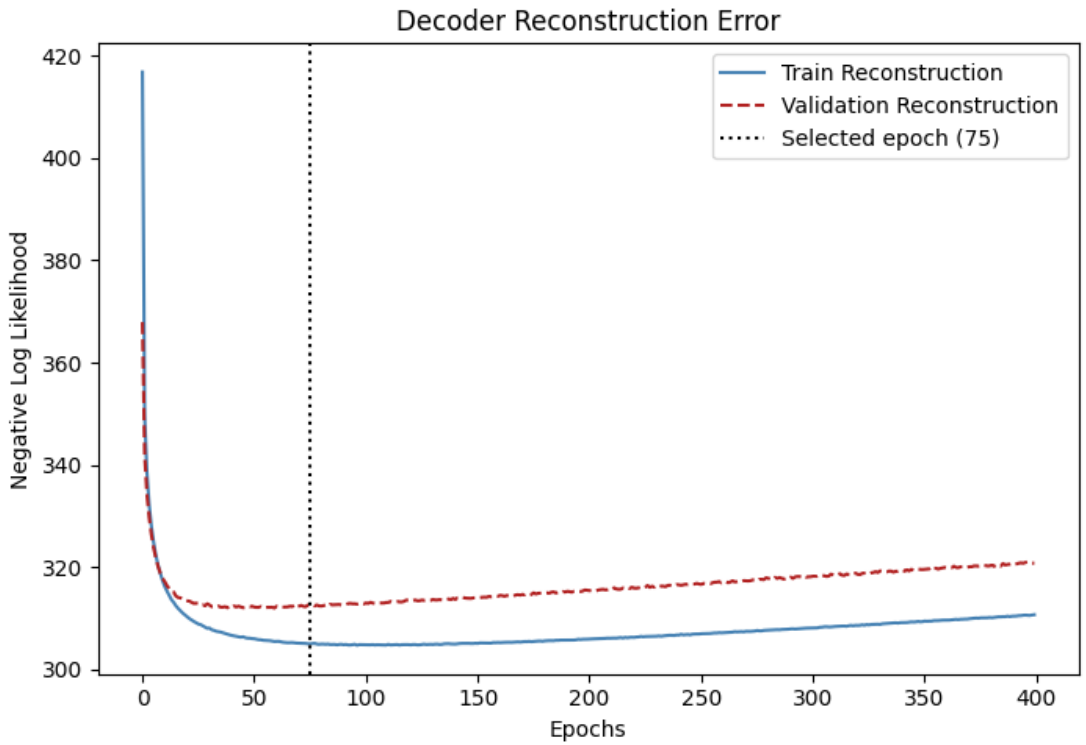
Runtime: 45 min on GPU

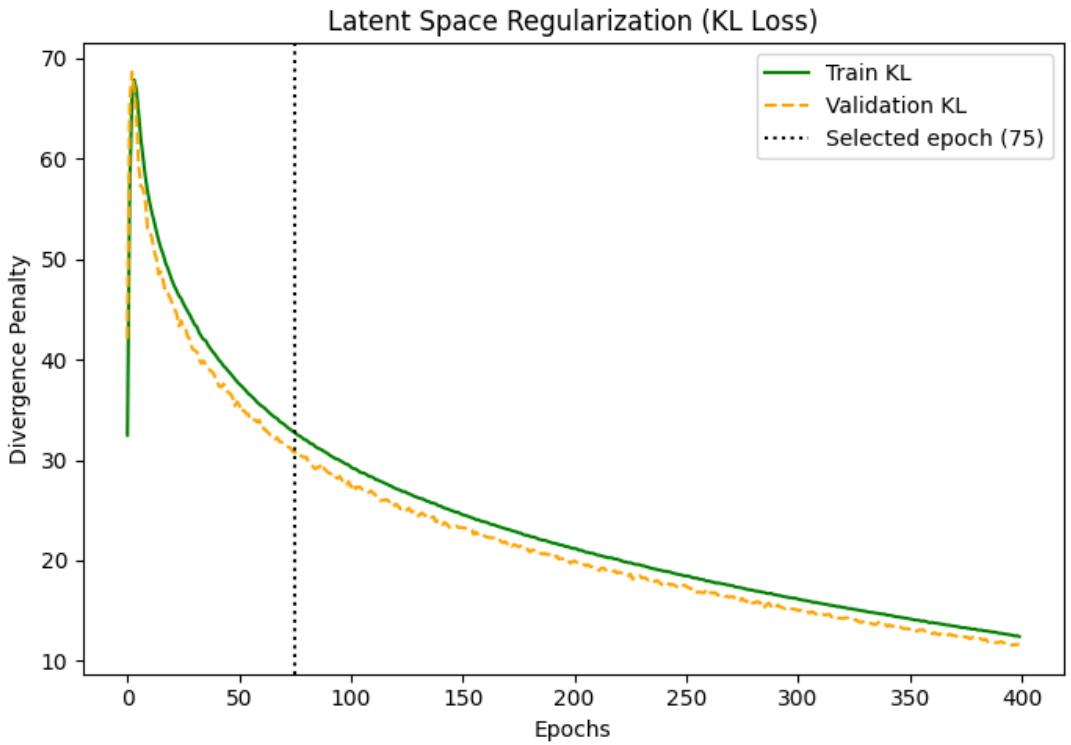
Parameter	Value
n_latent	20
n_layers	2
n_hidden	128
gene_likelihood	nb
batch_key	Participant



scVI model

Latent state extraction at 75 epochs



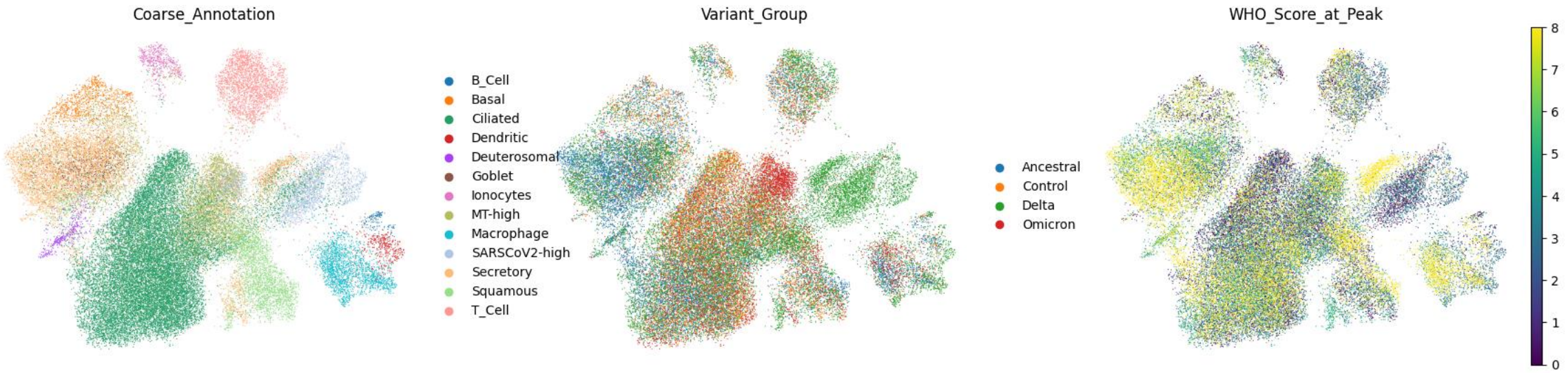


Leiden Clustering & UMAP (Ingrid)

Graph-clustering algorithm

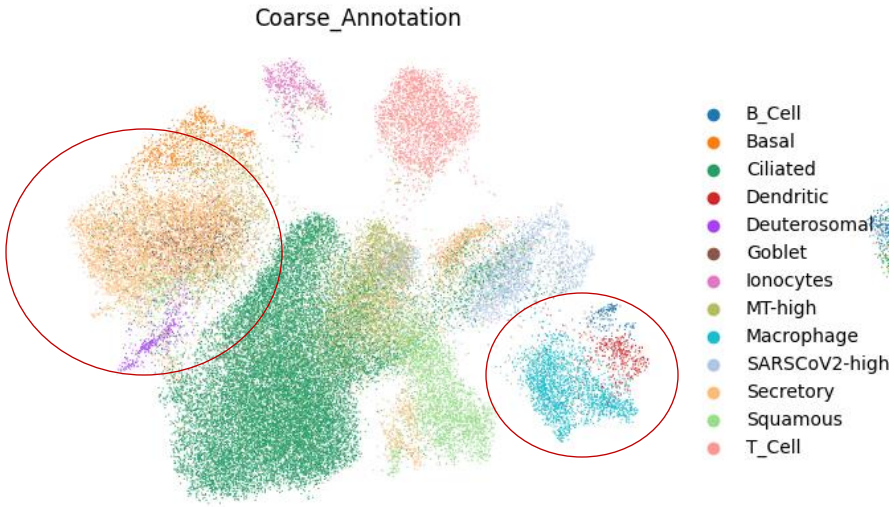
Optimizes **modularity -->** a measure of how well a graph is divided into communities, while ensuring that each cluster is internally well-connected.

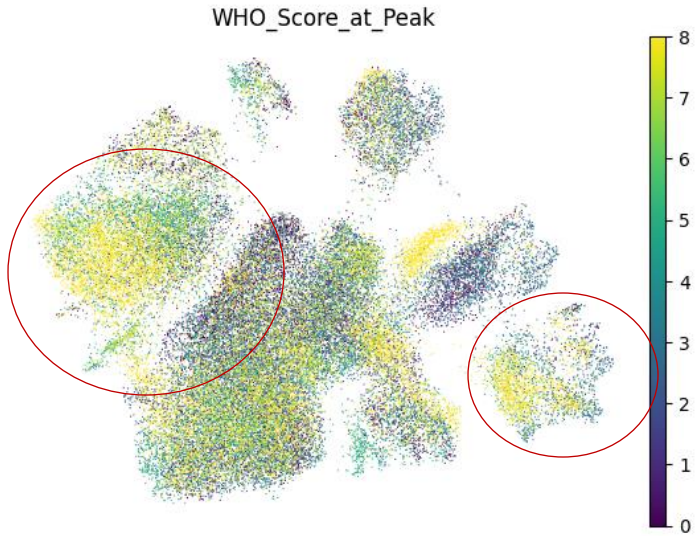
Leiden found 11 clusters. Our dataset has 13 cell annotations



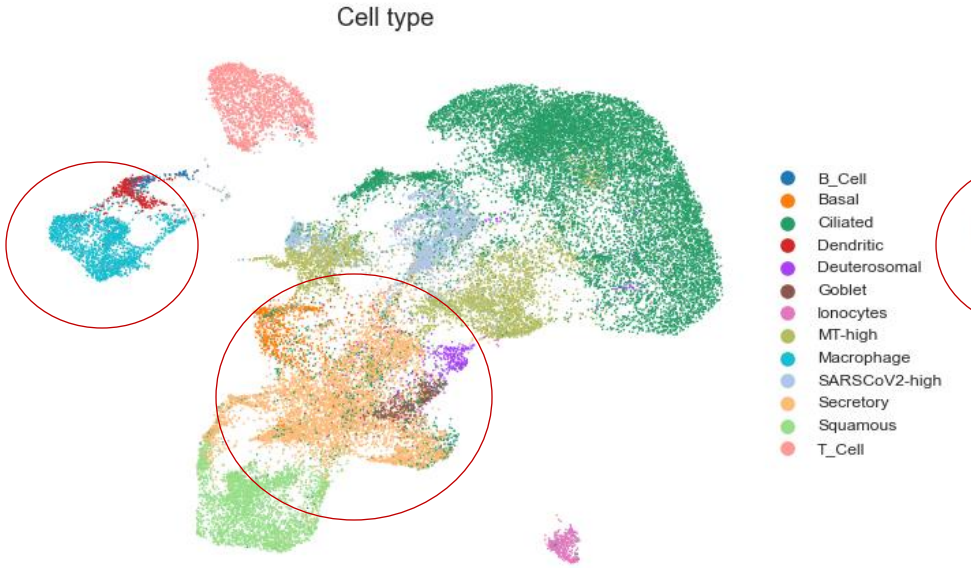
UMAP clustering: PC and scVI latent states

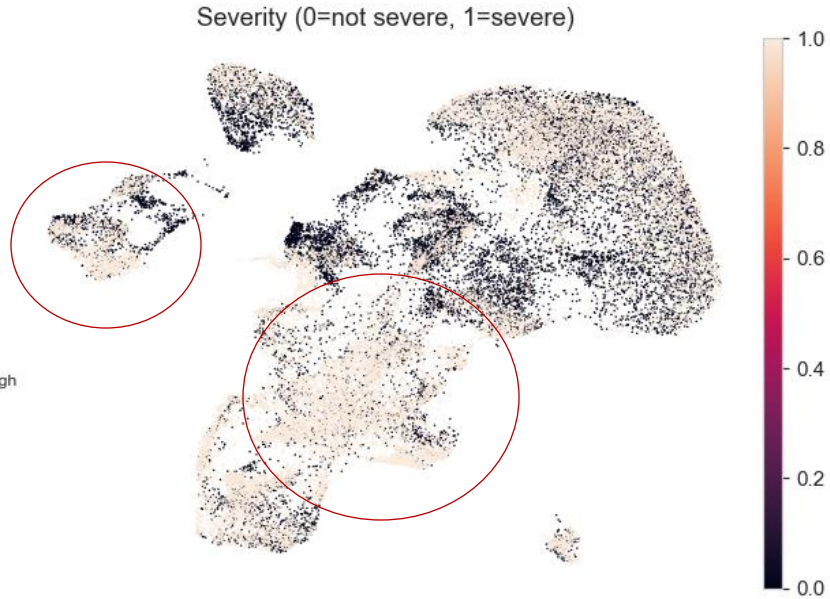
ScVI





PCA





scVI latent states

Regression (XGBoost & MLP) & Classification (XGBoost)

Model	Level	Features	Data grouping	Sample size	CV strategy
1. Cell latent only	Cell	20 scVI latent dimensions	Rows = cells. No aggregation	~38,000 cells	StratifiedGroupKFold (5-fold) grouped by participant
2. Cell latent + cell type	Cell	20 scVI latent dimensions + cell type label = 21 features	Rows = cells each	~38,000 cells	StratifiedGroupKFold (5-fold) grouped by participant
3. Participant mean latent	Participant	Mean of 20 scVI dimensions across all cells per participant = 20 features	Rows = paticipant Participant cells are averaged into one vector.	71 participants	StratifiedKFold (5-fold)
4. Participant per-celltype + composition	Participant	Mean of 20 scVI dimensions per cell type (×13) + proportion of cell type = 273 features	Rows = participant Averaged separately within each cell type and concatenated. Preserves cell-type-specific signals.	71 participants	StratifiedKFold (5-fold)

XGBoost for both Regression & Classification

Regression:
- How severe?
- A little wrong is better than very wrong

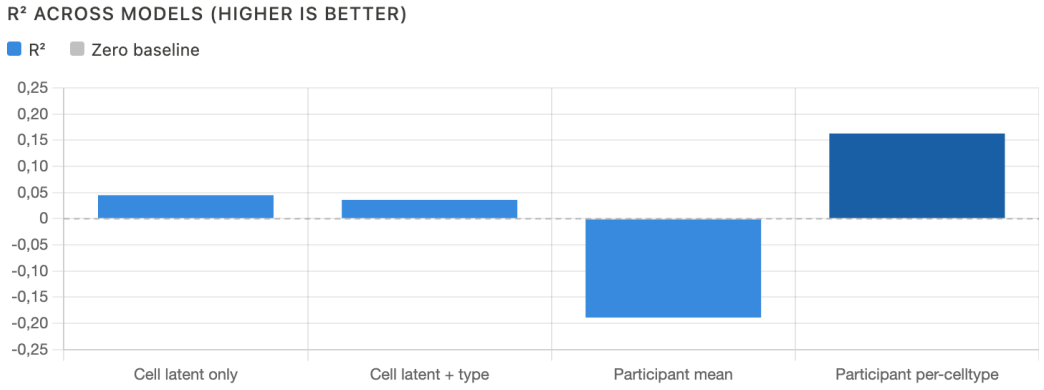
Classification:
- Severe or not?
- Might be better with small sample size

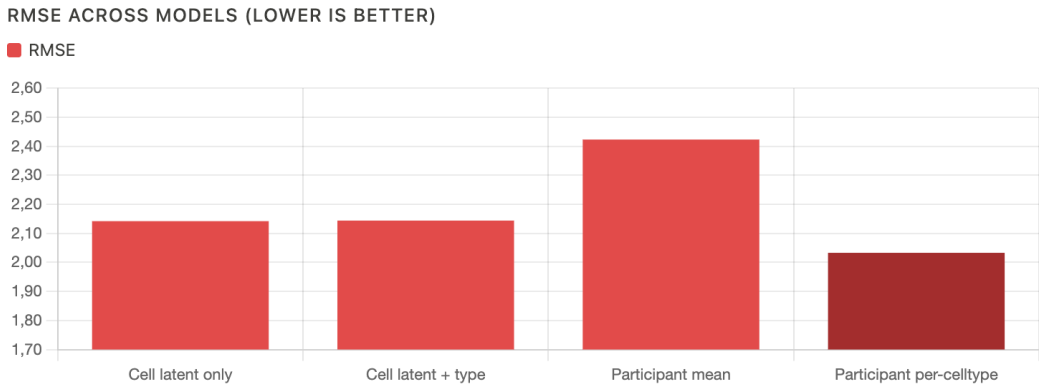
Best model:
Patient level model

Features:
- Mean scVI latent states per cell
- Cell composition
    - Total: 273 features

Architechture for best model:
- `n_estimators`: 200
- `max_depth`: 2
- `learning_rate`: 0.05
- `min_child_weight`: 5
- `subsample`: 0.8
- `colsample_bytree`: 0.3 (only 30% of features per tree)
- `reg_alpha`: 0.1 (L1)
- `reg_lambda`: 1.0 (L2)
- `tree_method`: hist

ScVI latent states XGBoost: Regression





Best regression R^2

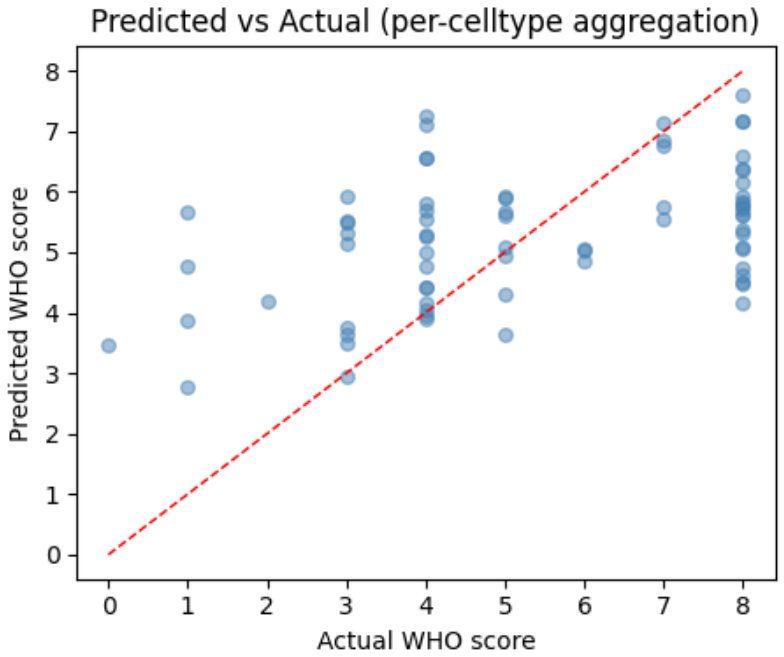
0.163

Participant per-celltype

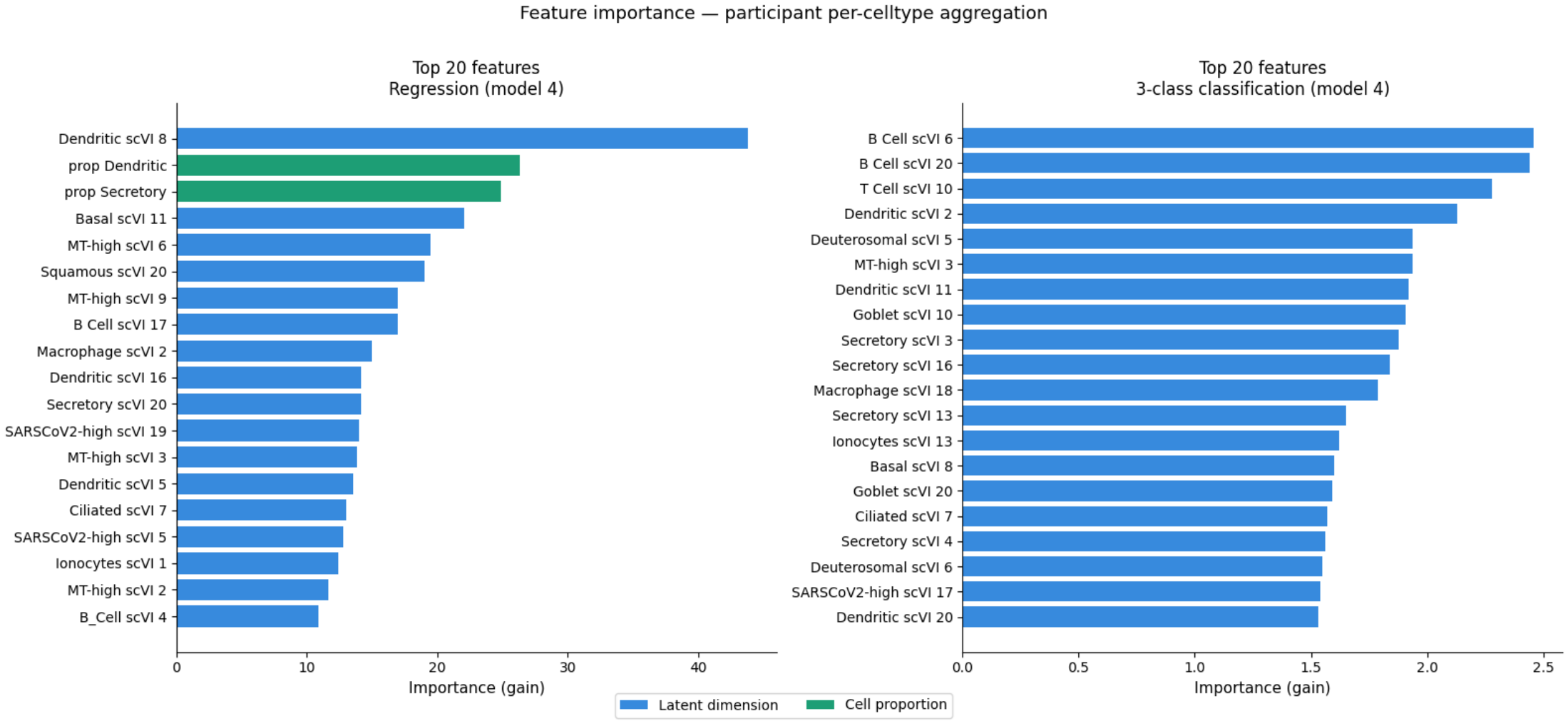
Regression p-value

<0.002

Permutation test, n=500



Feature importance



Findings using scVI

scVI latent representations *does* contain statistically significant information about COVID-19 severity

Cell type Findings

- *Dendritic cells* are important cell type
- *Secretory cell proportion* is a top feature in regression (+UMAP suggestion)
- *B cells and T cells* emerge as important in classification
- *Macrophages* appear consistently across both tasks

Methodological Findings

- Per-celltype aggregation outperforms simple mean aggregation

Predicting COVID-19 Disease Severity from Gene Ontology Pathway Vectors Using a Biologically-Constrained Autoencoder

List of differentially expressed genes

Test for enriched categories/pathways

Infer networks and integrate with other data

Biological insight

This biological insight can be numerically encoded by incorporation of GO terms

What are GO-terms and why to incorporate them in ML?

Increase the statistical/predictive power in our analysis

Ease the interpretation

Predict new roles for genes

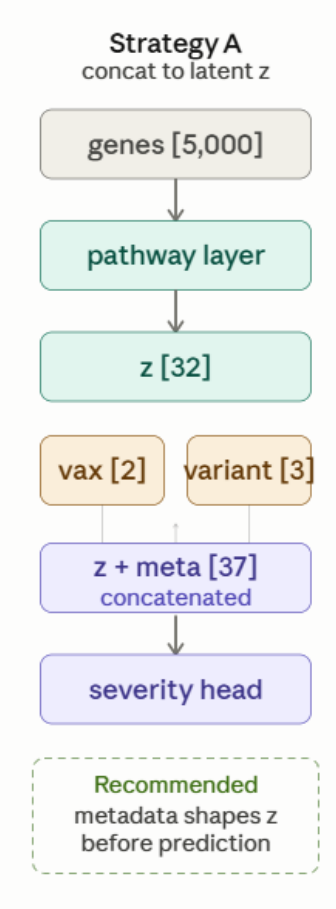
Better integrate data from different methods

N of cells= 48,730

29,961 genes Transformed to a vector with a length of 3000 storing numerical values

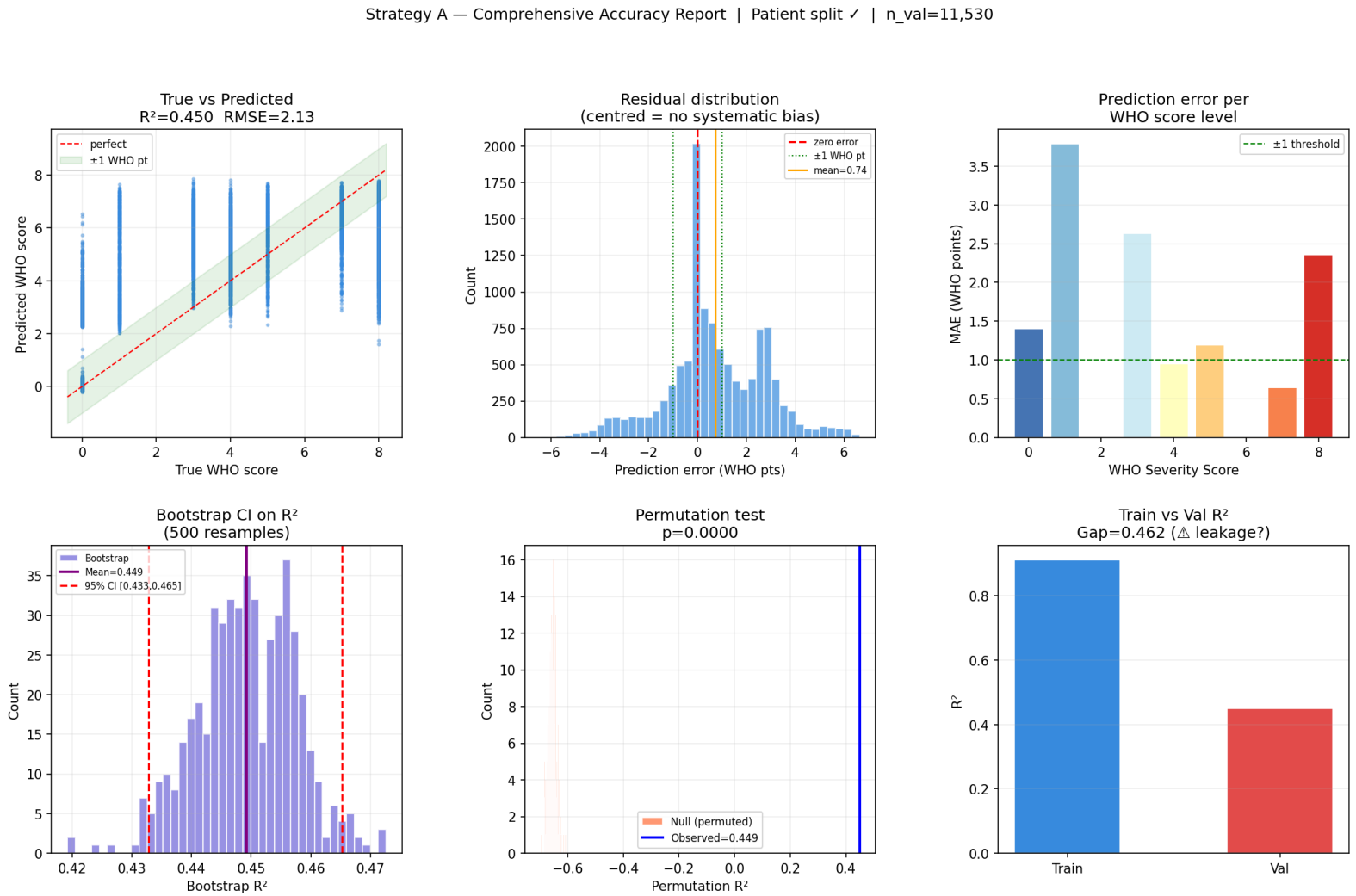
	Cell 1	Cell 2	Cell …						
GO term score 1									
GO term score 2									
…									

Strategy A- Autoencoders informed by GO term vector + metadata information about vaccination

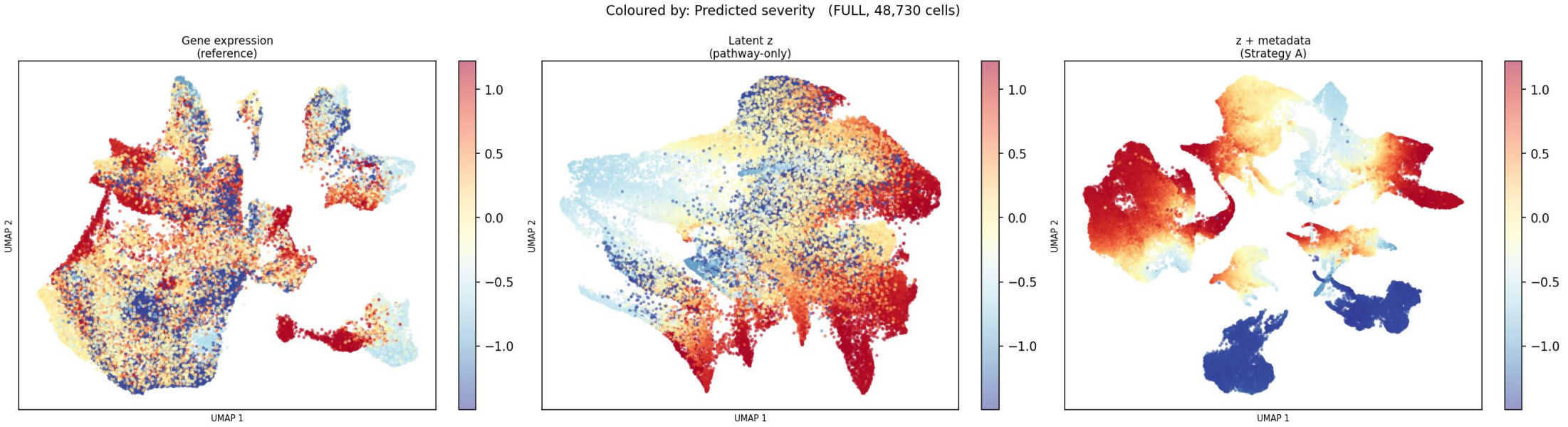



```
# Strategy A architecture
# ──────────────────────
#  Encoder:
#    genes [5000] → MaskedLinear → pathway scores [n_pw]
#                 → hidden [256] → latent z [32]
#
#  Metadata fusion- latent contatenation:
#    z [32] || one_hot(vaccination) [n_vax] || one_hot(variant) [n_var]
#    → z_meta [32 + n_vax + n_var]
#
#  Severity head:
#    z_meta → hidden [64] → severity score / class
#
#  Decoder (reconstruction — keeps AE honest):
#    z [32] → hidden [256] → pathway scores → genes [5000]
```

Results


Strategy A — Comprehensive Accuracy Report | Patient split ✓ | n_val=11,530
True vs Predicted
R²=0.450 RMSE=2.13
perfect
±1 WHO pt
Predicted WHO score
True WHO score
Residual distribution
(centred = no systematic bias)
zero error
±1 WHO pt
mean=0.74
Count
Prediction error (WHO pts)
Prediction error per
WHO score level
±1 threshold
MAE (WHO points)
WHO Severity Score
Bootstrap CI on R²
(500 resamples)
Bootstrap
Mean=0.449
95% CI [0.433,0.465]
Count
Bootstrap R²
Permutation test
p=0.0000
Null (permuted)
Observed=0.449
Count
Permutation R²
Train vs Val R²
Gap=0.462 (⚠ leakage?)
R²
Train
Val

Results


Coloured by: Predicted severity (FULL, 48,730 cells)
Gene expression
(reference)
Latent z
(pathway-only)
z + metadata
(Strategy A)
UMAP 1
UMAP 2
1.0
0.5
0.0
−0.5
−1.0

Why these results could be over-optimistic?

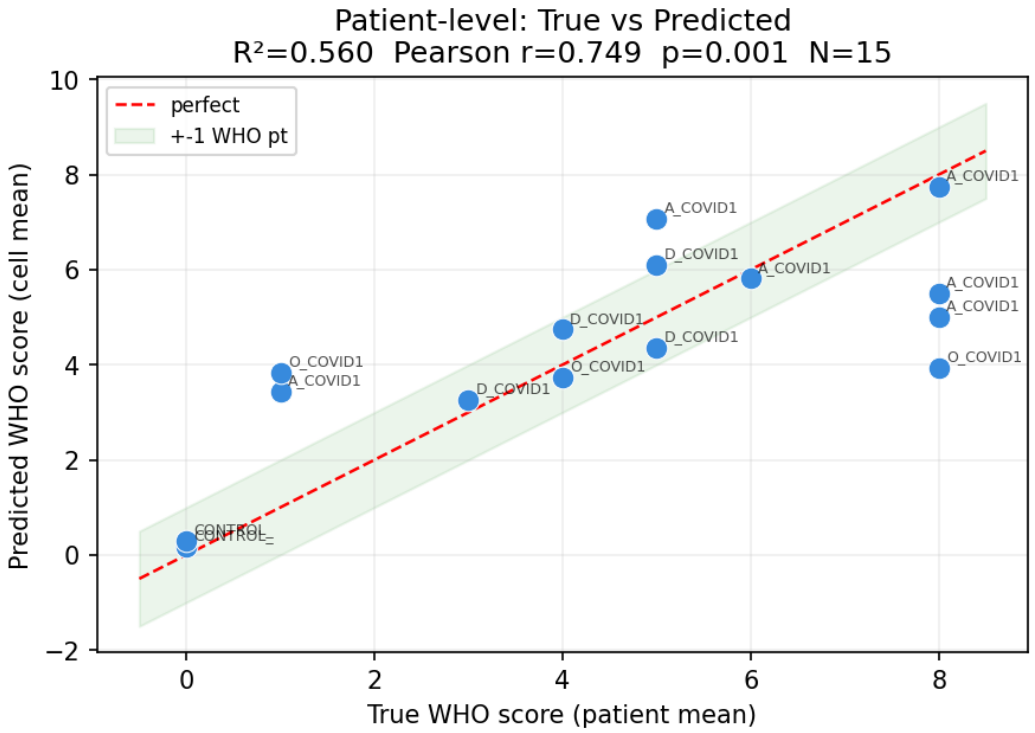
- **Severity signal is weak at cell level.** Individual cells don't reliably encode patient-level severity - many cell types are not directly responsive to COVID severity, so they add noise
- Pseudo/replication of cells. Many cells can convey the same biological information and are not independent data points.
- I used test/train split, cross validation could be better

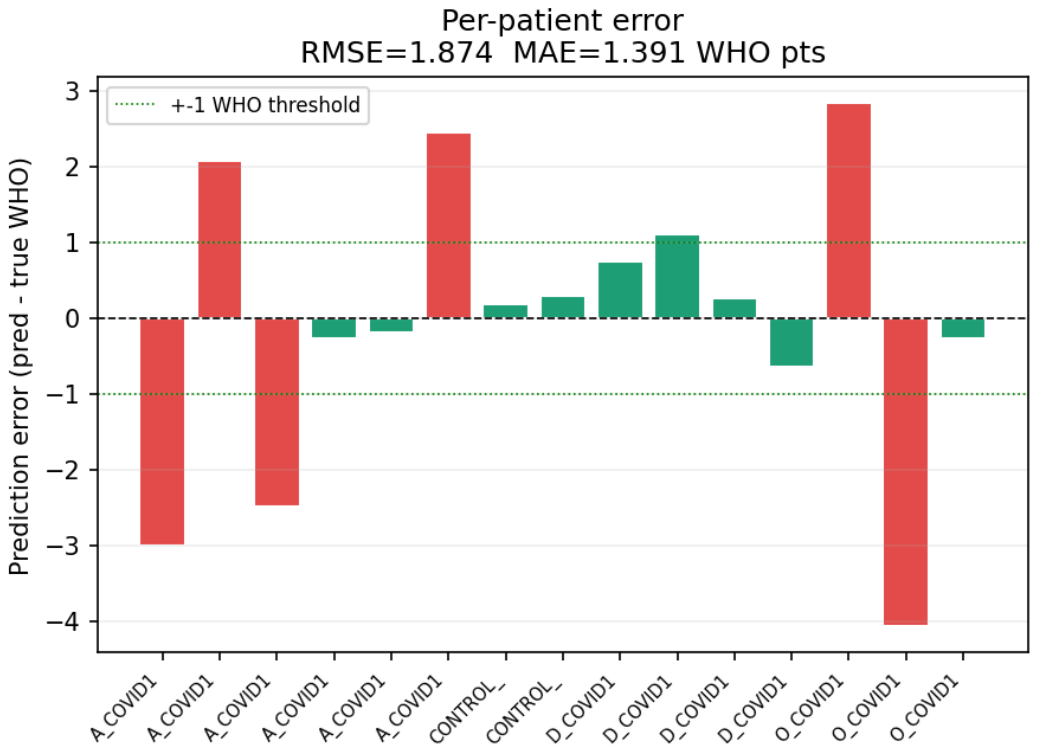
How I tried to make this model more reliable?

- **Patient-level R^2** - the primary metric. For each val patient it averages all their cell predictions into one number, then runs R^2, RMSE, MAE, Pearson r, and Spearman ρ against true WHO.
- Filter to **immune-relevant cell types only** (monocytes, T cells, NK cells) before training.

Strategy B- Results

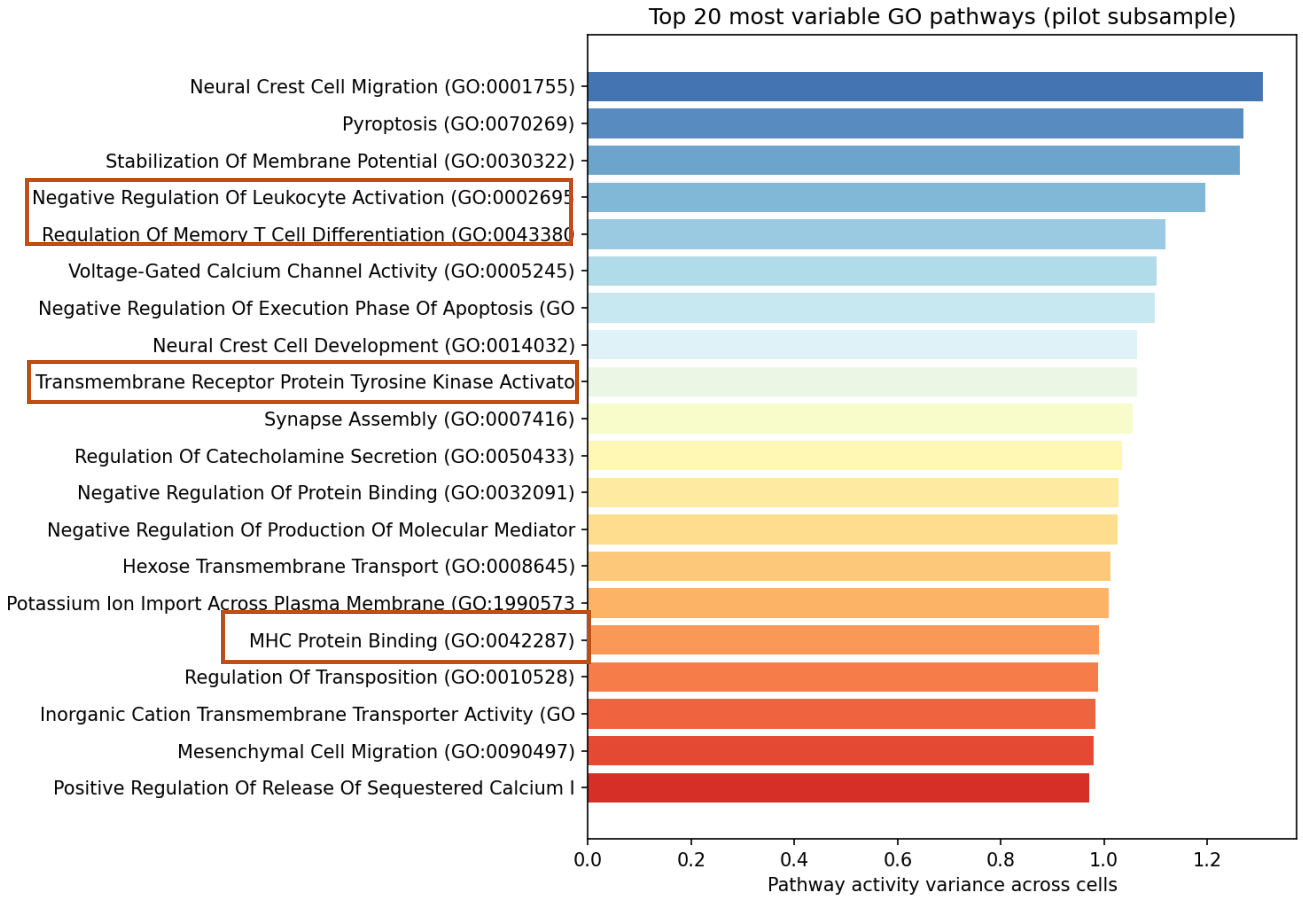
Patient-level severity prediction [PRIMARY METRIC]
N=15 val patients | Immune cells only: True | 95% CI=[0.021,0.802]





Results

Highly associated with Macrophages and Dendritic Cells - which proves earlier prediction of their importance



Conclusions Biological

- Our models are not directly comparable, but we find similar biological signal
- PC-based and latent state based models provide similar signal

Cell & Gene discovery:

- Secretory cell cluster drives severity score in multiple
- Dendritic cells, Macrophages, B- & T-cells

GO pathway vectors+ AE can predict COVID severity (R2=0.56)

Final conclusions

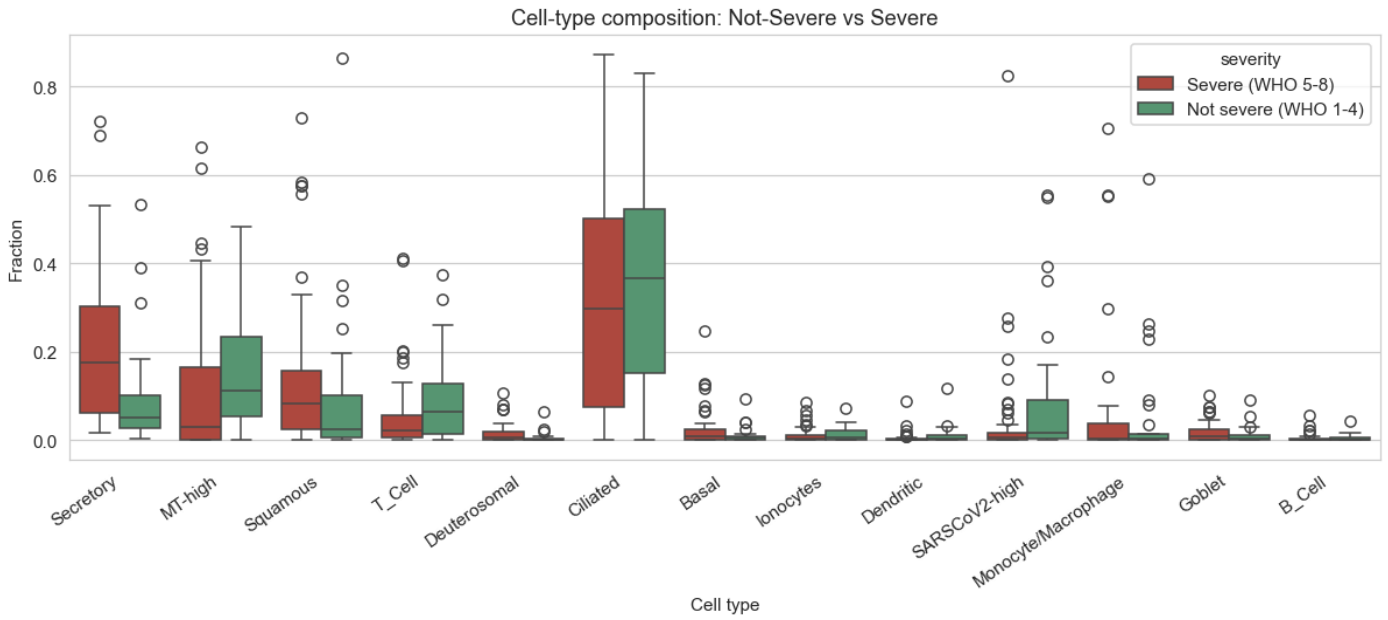
- More training data could improve our models ability to generalize and clinical relevance of the results
- ML models underperform on genetic data alone.
- Different dimensionality reduction lead to similar results

APPENDIX-slides

Cell Composition (step1-notebook1)

- "Given a random COVID+ patient from this dataset (some vaccinated, some not, all variants mixed), can cell composition predict severe vs not=severe"

- Feature importance tells you which cell types *matter*; this plot tells you *how* they matter:

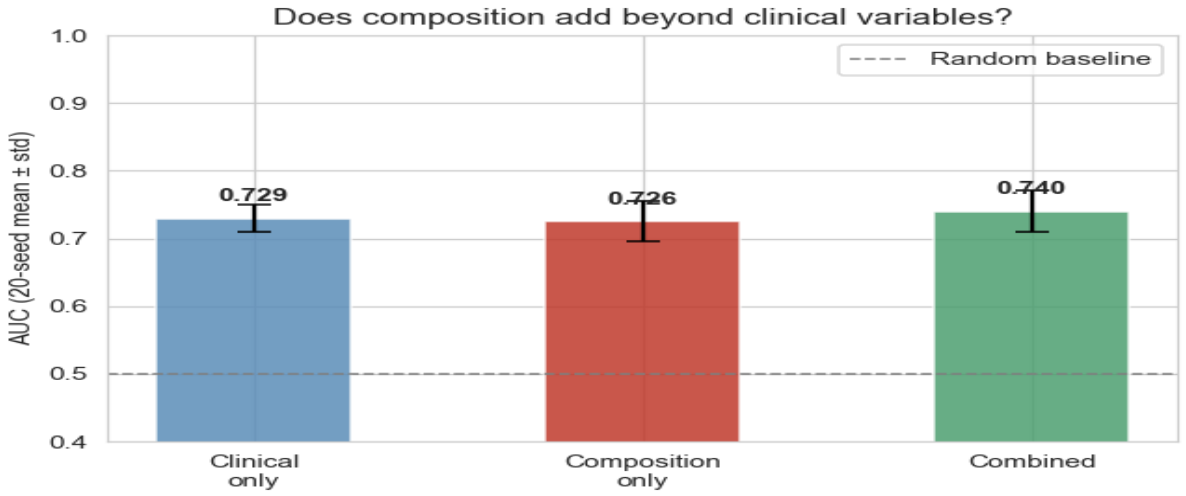


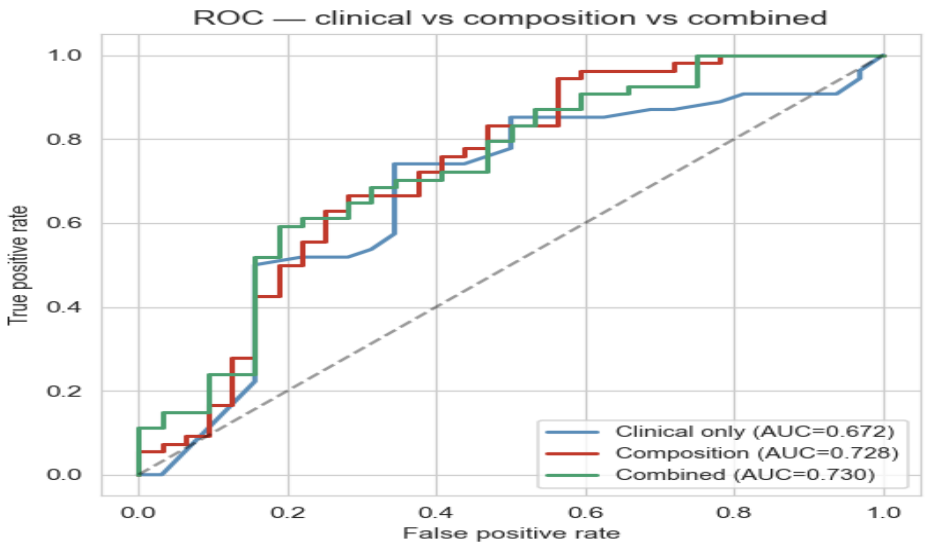
Does composition add anything beyond clinical variables alone?

We compare three models (RF):

- Clinical only — Variant_Vax_Group encoded {Encoding: {'Ancestral': np.int64(0), 'Delta Unvax': np.int64(1), 'Delta Vax': np.int64(2), 'Omicron Unvax': np.int64(3), 'Omicron Vax': np.int64(4)}
- Composition only — cell-type fractions
- Combined — both together

— Model comparison —
[Clinical only] 20-seed mean AUC: 0.729 ± 0.020 | pooled AUC: 0.672
[Composition] 20-seed mean AUC: 0.726 ± 0.030 | pooled AUC: 0.728
[Combined] 20-seed mean AUC: 0.740 ± 0.030 | pooled AUC: 0.730





Trying a lightgbm / xgboost models and compare them to RF

Model	Mean AUC	Std	Min fold	Max fold
Random Forest	0.717143	0.221452	0.318182	0.942857
LightGBM	0.699307	0.172190	0.393939	0.922078
XGBoost	0.684329	0.168724	0.409091	0.922078

All three classifiers fall within ~0.03 AUC of one another (RF 0.717, LightGBM 0.699, XGBoost 0.684), well inside the bootstrap CI of ±0.13. Random Forest has the highest mean but also the widest spread (std 0.22, worst fold 0.32), while the boosting methods are noticeably more stable (std ≈ 0.17, worst folds ≈ 0.40). The fact that a linear-ish ensemble and two gradient-boosting methods give statistically indistinguishable results means the severity signal in cell composition is robust to model choice, it is not an artefact of any single algorithm's inductive bias. We retain Random Forest as the primary model for downstream interpretability, but note that the boosting models would be equally defensible.

Moving to the secretory cells (step2-notebook1)

- The structure of the data:

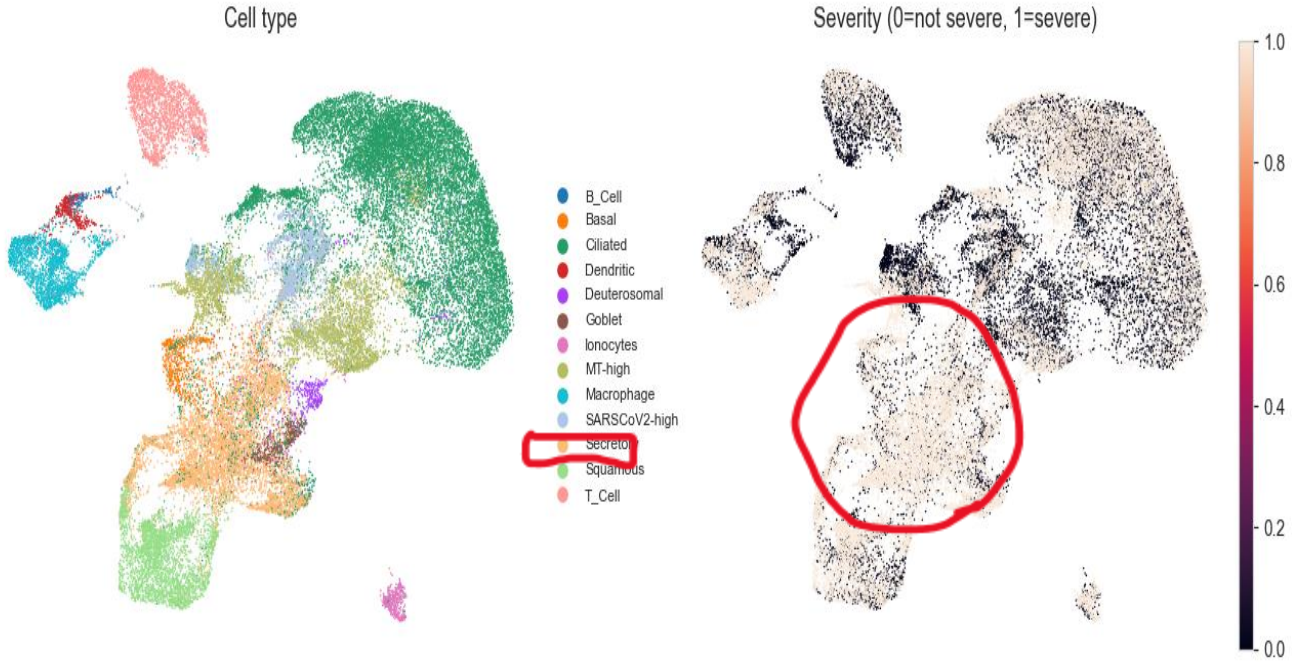
Cells × Genes: 48,730 × 29,961

Matrix type: csr_matrix

Matrix dtype: int64

Nonzero count: 48,459,870

Sparsity: 96.68% zeros (scRNA-seq matrices are typically 90–95% zeros (most genes aren't expressed in most cells).



COVID+ cells: 38,989 , COVID+ patients: 86

Severity label distribution per cell:
1 22275
0 16714

Patients per severity class:
0 32
1 54

Pseudobulk + Normalize + PCA for Secretory Cells (Step2-Notebook 2&3)

For each patient, turn their Secretory cells into a single gene-expression vector

(a "pseudobulk" profile), then reduce dimensionality with PCA so we can:

1. Visualize whether severe vs not-severe patients separate in PC space

2. Use those PCs as features for ML

Why pseudobulk? Each patient contributes hundreds of Secretory cells. We can't treat

them as independent samples (they share a patient), that would be data leakage. Instead,

we average across each patient's Secretory cells to get one number per gene per patient.

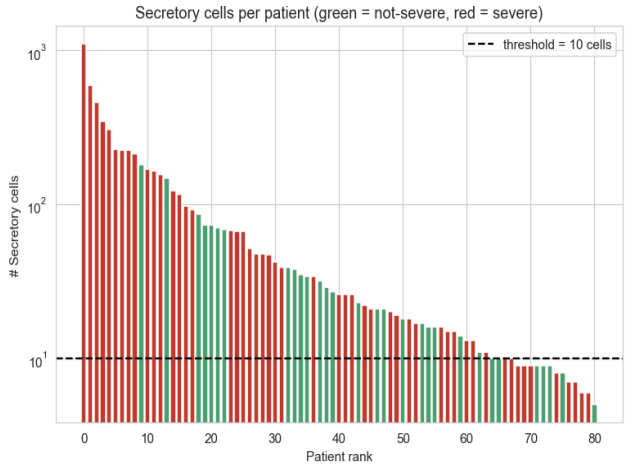
That gives us a clean **86 patients × ~30,000 genes** matrix, which we then PCA down to **86 patients × 50 PCs.**

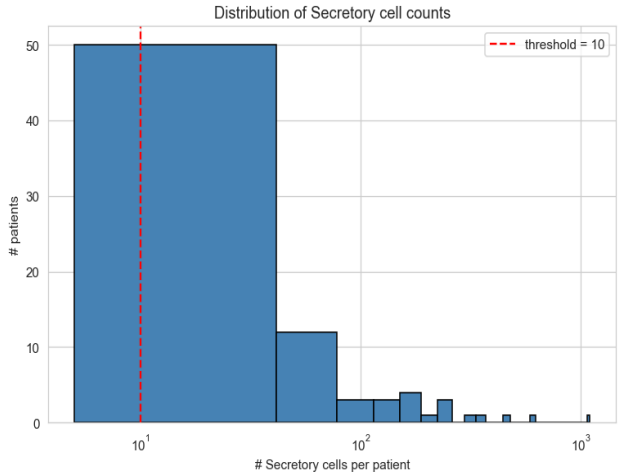
- **The total secretory cells: 6.656 from 86 patients:** Matrix shape: (6656, 29961) , some patients may have *zero* Secretory cells AND THEY will drop out.

Patient filtering decision

- Below 10 cells, the pseudobulk average is dominated by 1-2 cells and is unreliable. We chose 10 as the minimum — losing 18 patients but keeping a stable signal.

Threshold	n_total	n_not_severe	n_severe	ratio
3	81	30	51	30/51
5	81	30	51	30/51
10	68	25	43	25/43
15	59	21	38	21/38
20	49	17	32	17/32





Preprocessing: normalize → log1p → pseudobulk → HVG selection"

Raw counts per cell

→ normalize_total (each cell sums to 10,000)

→ log1p transformation

→ mean across patient's Secretory cells (pseudobulk)

→ select top 2000 highly variable genes (Seurat method)

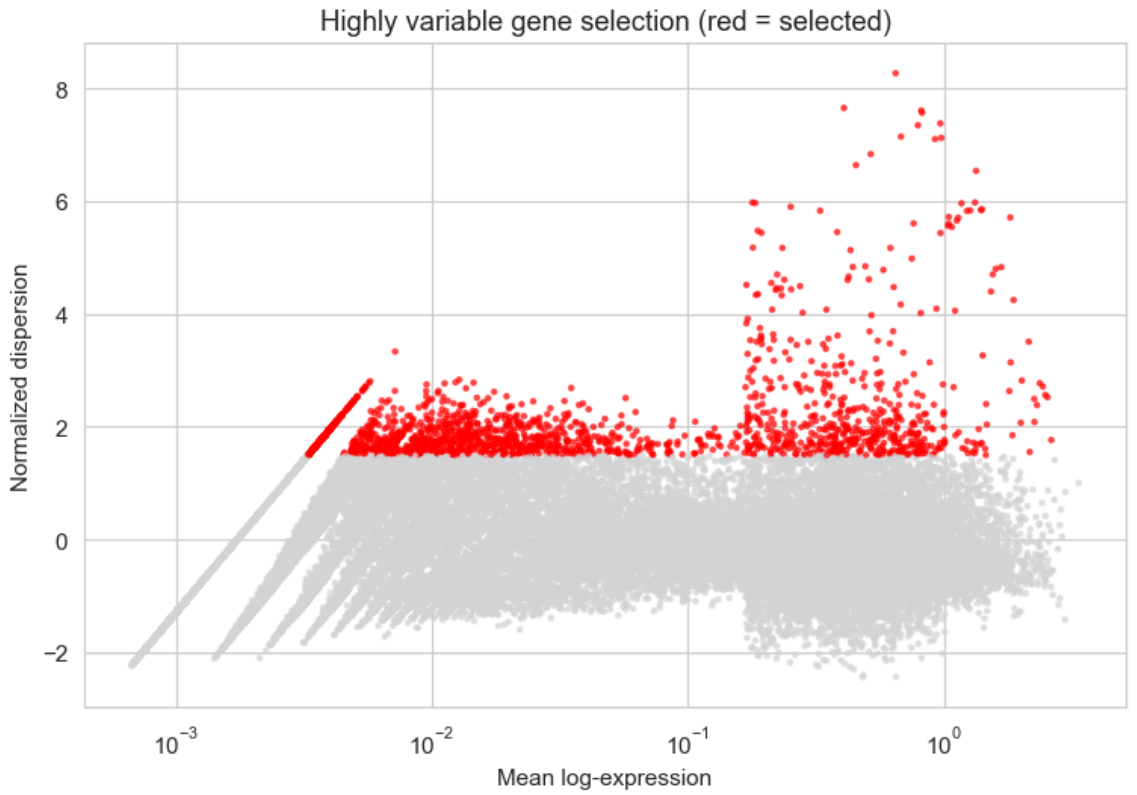
→ StandardScaler (zero mean, unit variance per gene)

→ PCA (50 components)

" Normalization removes sequencing depth differences between cells. Log transform handles the heavy-tailed distribution of gene counts. HVG selection removes uninformative genes before PCA."

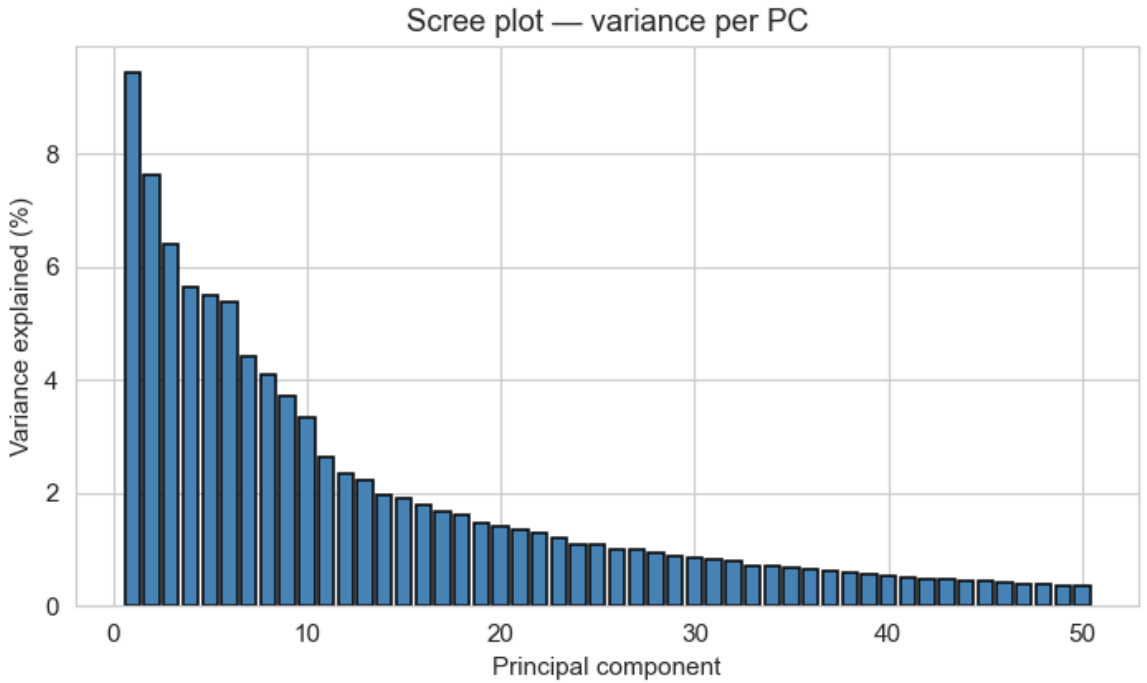
2000 most variable genes selected (red) out of 30,000

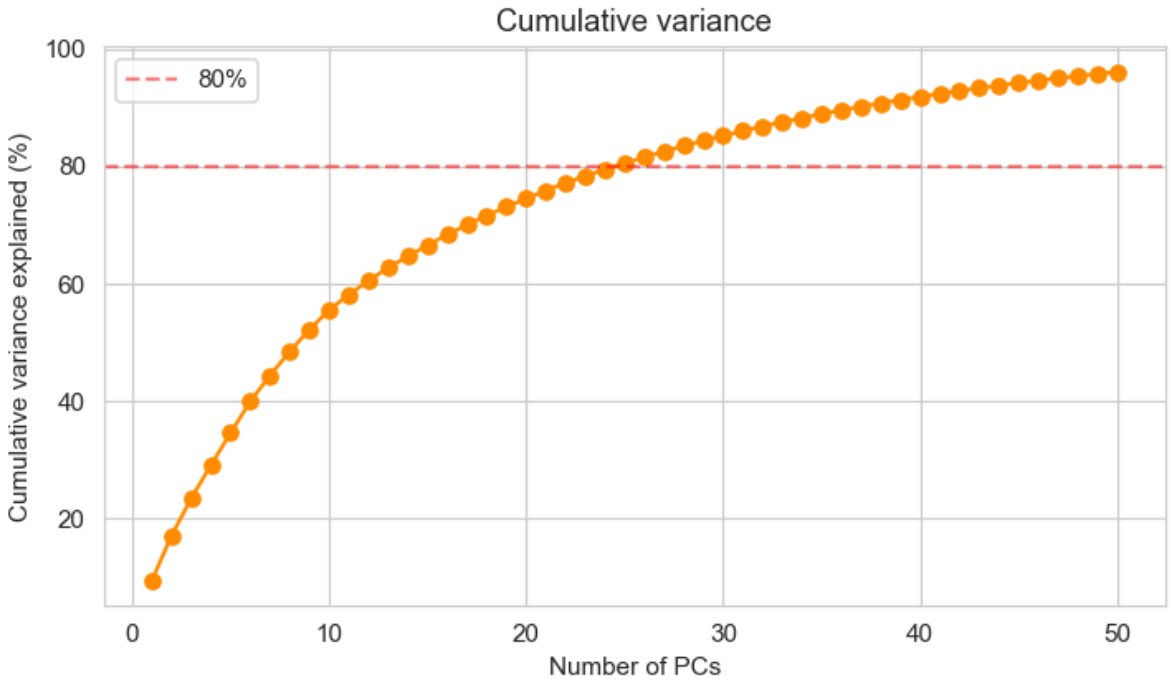
- We select genes that vary most across cells , these are most likely to carry biological signal. Housekeeping genes (low dispersion) and unexpressed genes are excluded. Note: SARS-CoV-2 viral genes appear in the HVG list; relevant for variant prediction but a potential confounder for severity



50 PCs capture ~96% of variance in Secretory expression

- PC1 alone captures the most variance but doesn't separate severity groups cleanly. Severity signal is distributed across many PCs , this is why we use all 50 as features for ML rather than just the top 2-3.

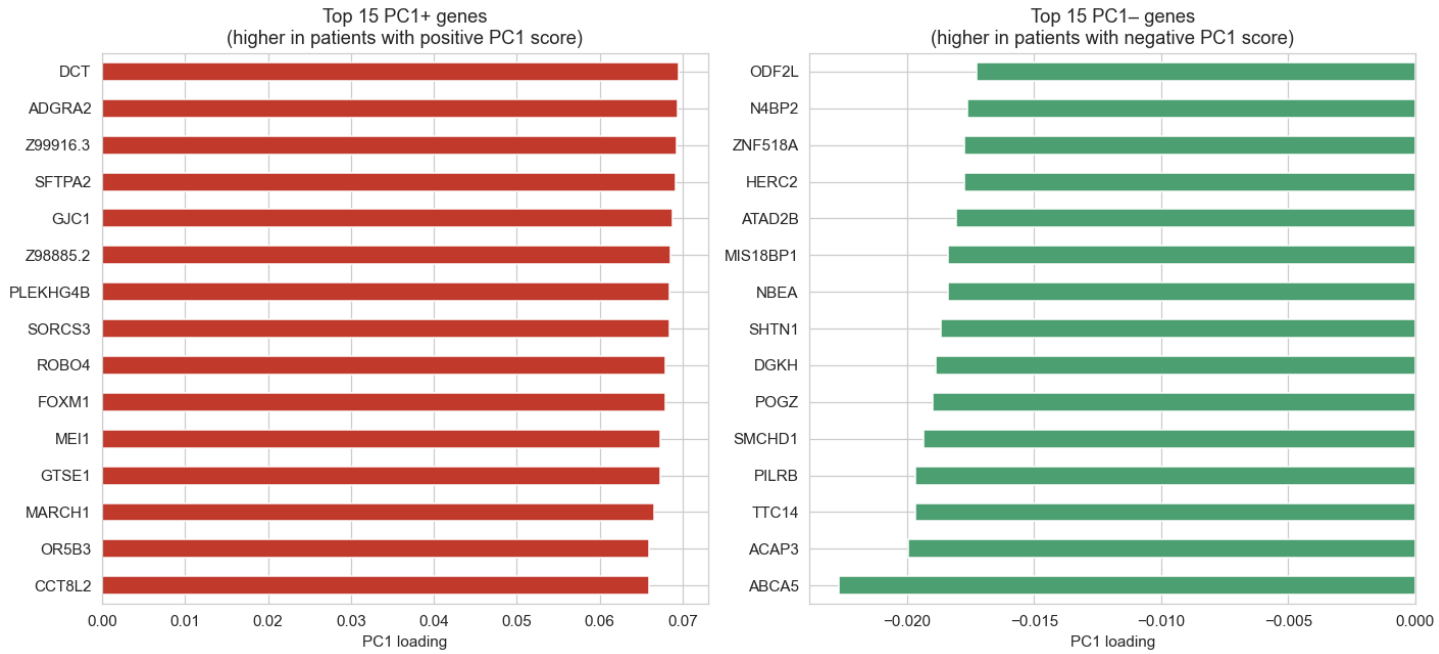




Which genes drive PC1? (loadings)

Top 10 PC1+ genes: ['DCT', 'ADGRA2', 'Z99916.3', 'SFTPA2', 'GJC1', 'Z98885.2', 'PLEKHG4B', 'SORCS3', 'ROBO4', 'FOXM1']

Top 10 PC1– genes: ['ABCA5', 'ACAP3', 'TTC14', 'PILRB', 'SMCHD1', 'POGZ', 'DGKH', 'SHTN1', 'NBEA', 'MIS18BP1']



ML setup:
3 models,
2 targets, 5-fold CV

- "Dummy baseline is always 0.5. All results reported as 20-seed mean ± std to account for fold assignment randomness."

	Severity	Variant
Target	WHO ≥ 5 (binary)	Ancestral / Delta / Omicron
Patients	68 (25 mild, 43 severe)	68 (29 / 23 / 16)
Features	50 PCs from Secretory pseudobulk	same
Metric	ROC-AUC	AUC one-vs-rest weighted
CV	5-fold stratified, 20-seed stability	same

Severity prediction:
Random Forest AUC = 0.742 (single seed) / 0.734 ± 0.038 (20-seed)

- Random Forest is the clear winner for severity. XGBoost and LightGBM show high variance across folds, suggesting they overfit on this small dataset of 68 patients.

Model	20-seed AUC	Std
Random Forest	0.743	0.038
XGBoost	0.658	0.125
LightGBM	0.658	0.087
Dummy	0.5	0.0

Variant prediction: XGBoost AUC = 0.823 ± 0.027 — much stronger than severity

- Variant is far more predictable than severity from Secretory gene expression (0.823 vs 0.734). This is consistent with the original paper, the variant shapes the nasal immune landscape strongly, while severity signal is weaker and more variabl

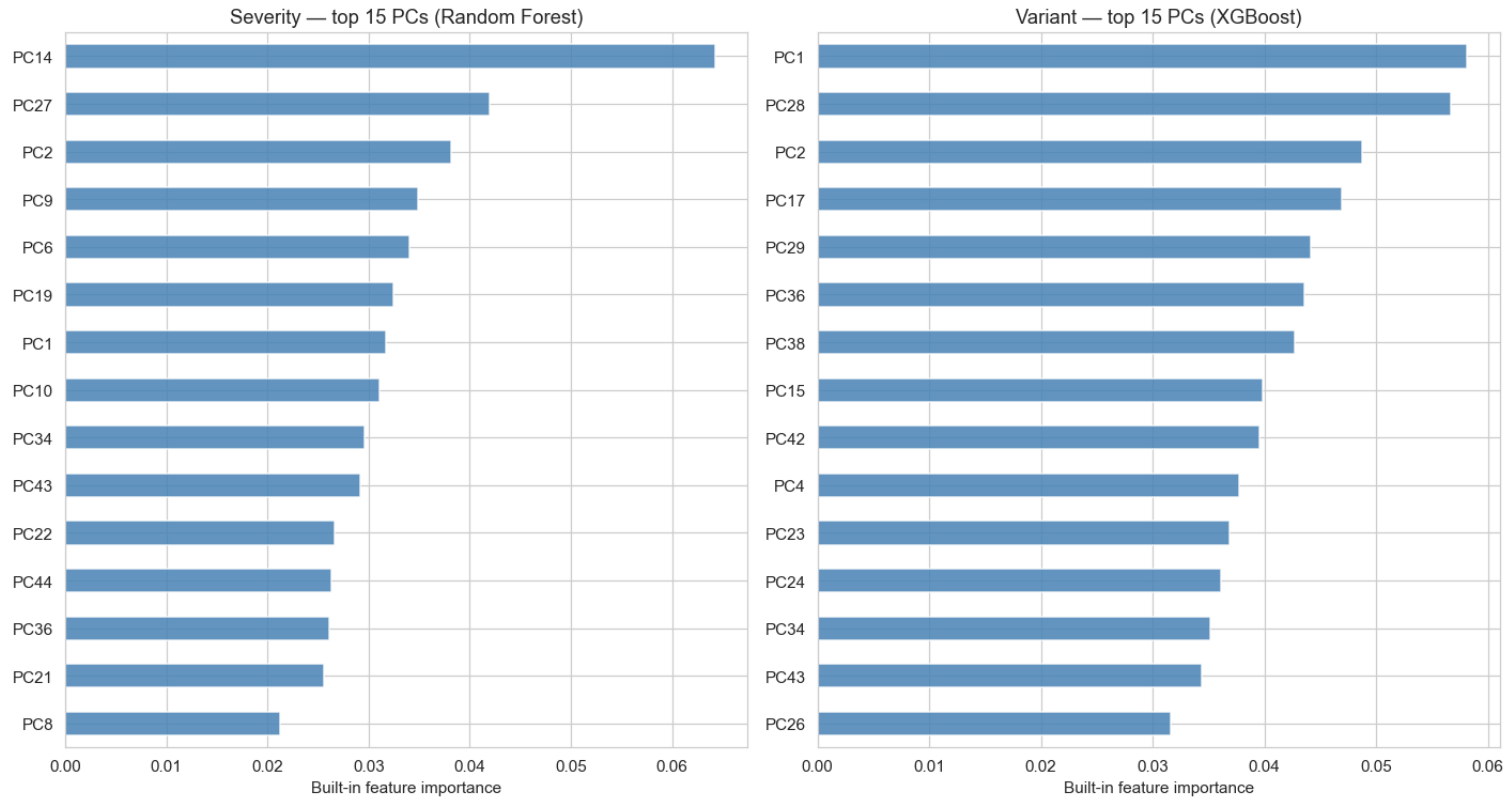
Model	20-seed AUC	Std
Random Forest	0.823	0.027
XGBoost	0.824	0.076
LightGBM	0.768	0.106
Dummy	0.5	0.0

Adding gene expression improves on composition alone

- Each layer adds information. Composition alone: 0.704. Gene expression alone: 0.734. Combined: 0.753. The improvement from combining (+0.05) suggests composition and gene expression capture complementary aspects of severity.

Feautre Set	20-seed AUC	Std
Composition only	0.704	0.033
Secretory PCs only	0.734	0.038
Composition + PCs	0.753	0.031

Severity and variant use different PCs -> different biological signals



Limitations

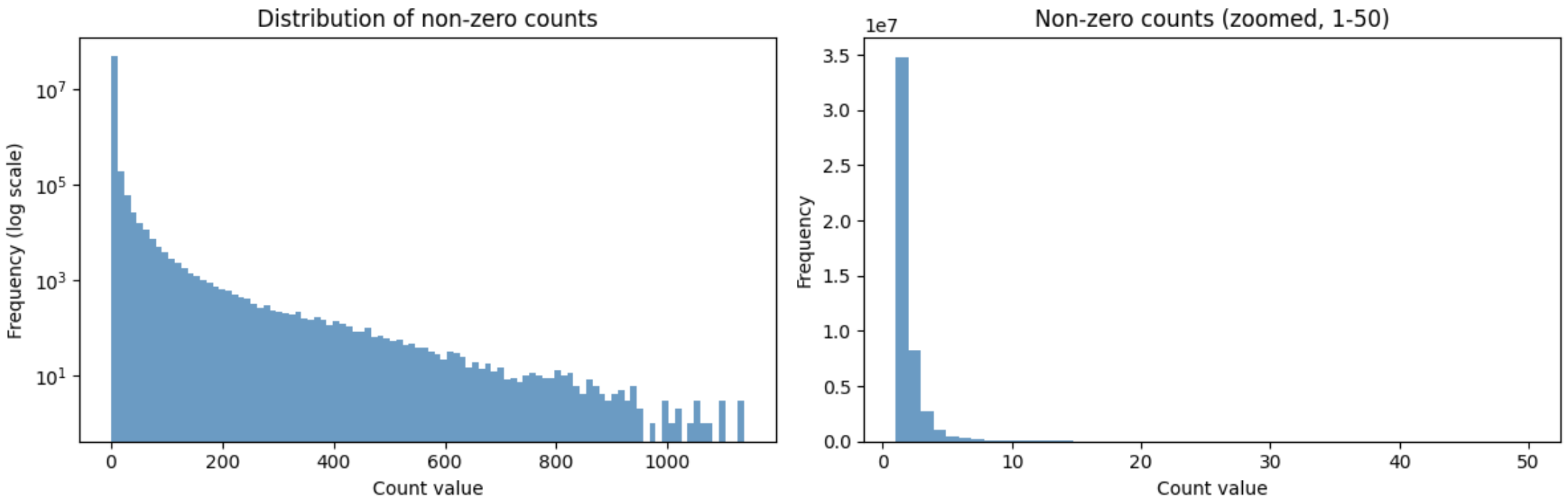
68 patients is small ; fold AUC variance is high (min 0.675, max 0.844 for severity RF)

Secretory cells only; other cell types may carry stronger severity signal"

Variant and severity are correlated; Omicron patients tend to be less severe, which may inflate variant AUC

No external validation; results are cross-validated within the same cohort

Distribution of non-zero values



ScVI appendix – Model Architechture

XGBoost (regression and classification)

All XGBoost models used the same base architecture, with slight differences:

Cell-level models (1 and 2):

- n_estimators: 200
- max_depth: 4
- learning_rate: 0.05
- subsample: 0.8
- colsample_bytree: 0.8
- tree_method: hist

- **Participant-level models (3 and 4)**, more regularized due to small sample size:
- n_estimators: 200
- max_depth: 2 (shallower)
- learning_rate: 0.05
- min_child_weight: 5
- subsample: 0.8
- colsample_bytree: 0.3 (only 30% of features per tree)
- reg_alpha: 0.1 (L1)
- reg_lambda: 1.0 (L2)
- tree_method: his

MLP (regression only)

Cell-level models (1 and 2):

- 2 hidden layers: 64 → 32 units
- BatchNorm + ReLU + Dropout (0.3) per layer
- Output: 1 unit (regression)
- Optimizer: Adam, lr=0.001, weight decay=1e-4
- Batch size: 256
- Early stopping: patience=15 on internal 15% validation split
- Max epochs: 200

Participant-level models (3 and 4):

- Model 3: 2 hidden layers: 32 → 16 units
- Model 4: 2 hidden layers: 128 → 64 units
- ReLU + Dropout per layer (no BatchNorm)
- Optimizer: Adam, lr=0.001, weight decay=1e-4
- Batch size: 16
- No early stopping — trained for fixed 200 epochs
- Max epochs: 200

scVI

- Architecture: Variational Autoencoder
- Encoder/decoder: 2 layers, 128 hidden units each
- Latent dimensions: 20
- Gene likelihood: Negative Binomial
- Batch correction: per participant
- Training: 75 epochs (chosen based on validation reconstruction loss)
- Optimizer: Adam (default scVI settings)

ScVI: VAE autoencoder

Build for this type of biological data!

Negative binomial distribution (as opposed to gaussian)

- Counts with many zeros

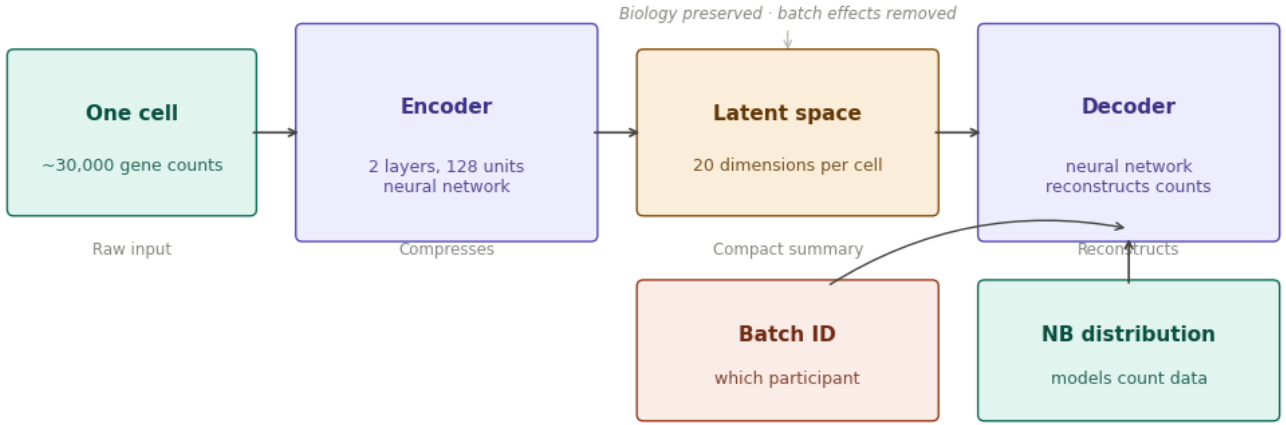
Integrated Batch Correction: Patient level

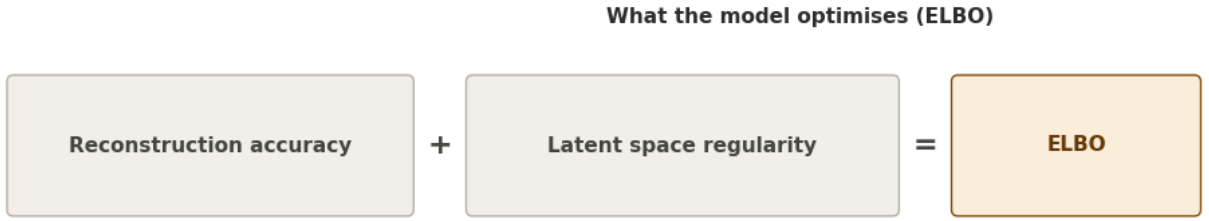
ELBO evaluation metric

Runtime: 45 min on GPU

Parameter	Value
n_latent	20
n_layers	2
n_hidden	128
gene_likelihood	nb
batch_key	Participant

We ran the autoencoder using both all genes and the 3000 most highly variable genes in the dataset. The 3000 HVG version was cleaner and used for prediction models.





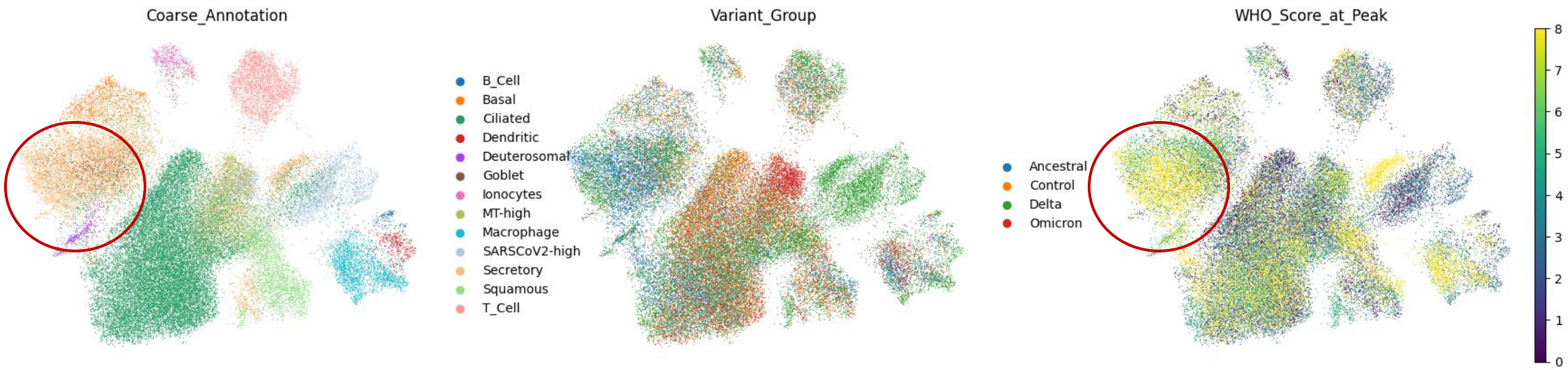
ScVI - Leiden Clustering & UMAP

Graph-clustering algorithm (Runtime: 15 min on GPU)

Optimizes **modularity -->** a measure of how well a graph is divided into communities, while ensuring that each cluster is internally well-connected.

- **Local moving:** Each node (cell) is moved to the neighboring community that increases modularity the most.
- **Refinement:** Leiden splits communities that are not internally well-connected.
- **Aggregation** Communities become super-nodes, and the process repeats on the compressed graph.

Leiden found 11 clusters. Our dataset has 13 cell annotations



Best Regression Model (scVI) - model 4

XGBoost model 4 performed the best, outperforming the other aggregations as well as MLP.

XGBoost

Fold 1: RMSE=1.757 | MAE=1.438 | R^2=0.149 | train=56 | test=15
Fold 2: RMSE=2.218 | MAE=1.758 | R^2=0.292 | train=57 | test=14
Fold 3: RMSE=2.310 | MAE=2.032 | R^2=-0.145 | train=57 | test=14
Fold 4: RMSE=1.924 | MAE=1.516 | R^2=0.153 | train=57 | test=14
Fold 5: RMSE=1.957 | MAE=1.692 | R^2=0.368 | train=57 | test=14

CV Summary (aggregated participant level): RMSE: 2.033 ± 0.226 MAE: 1.687 ± 0.232 R^2: 0.163 ± 0.196

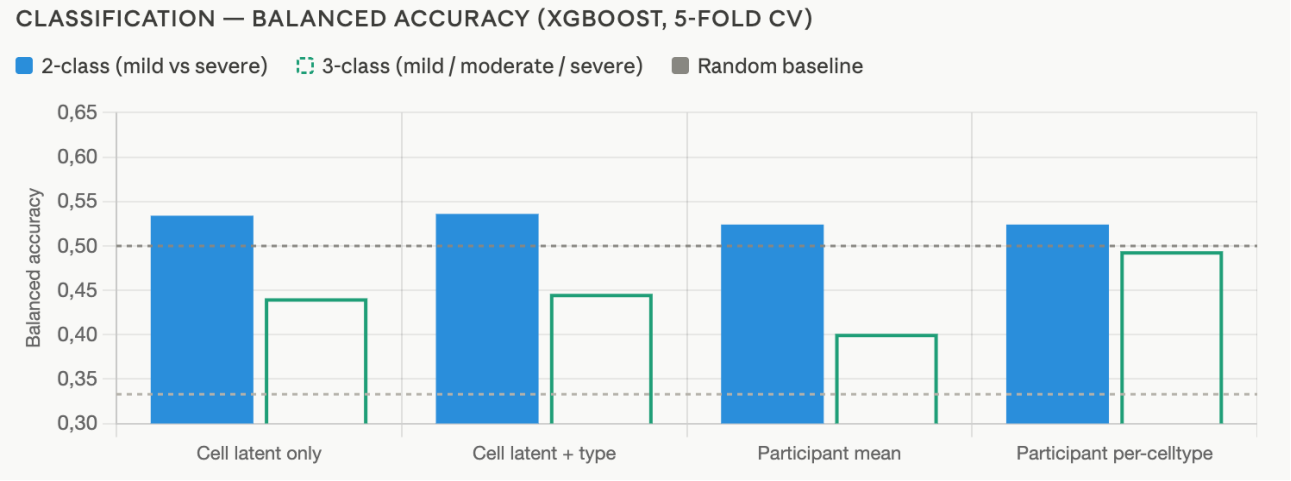
Full Comparison of all 4 XGBoost Reg models:

Model	RMSE	MAE	R^2
Cell-level latent only	2.142	1.823	0.045
Cell-level + cell type	2.144	1.827	0.036
Participant mean latent	2.423	2.067	-0.189
Participant per-celltype	2.033	1.687	0.163

Classification

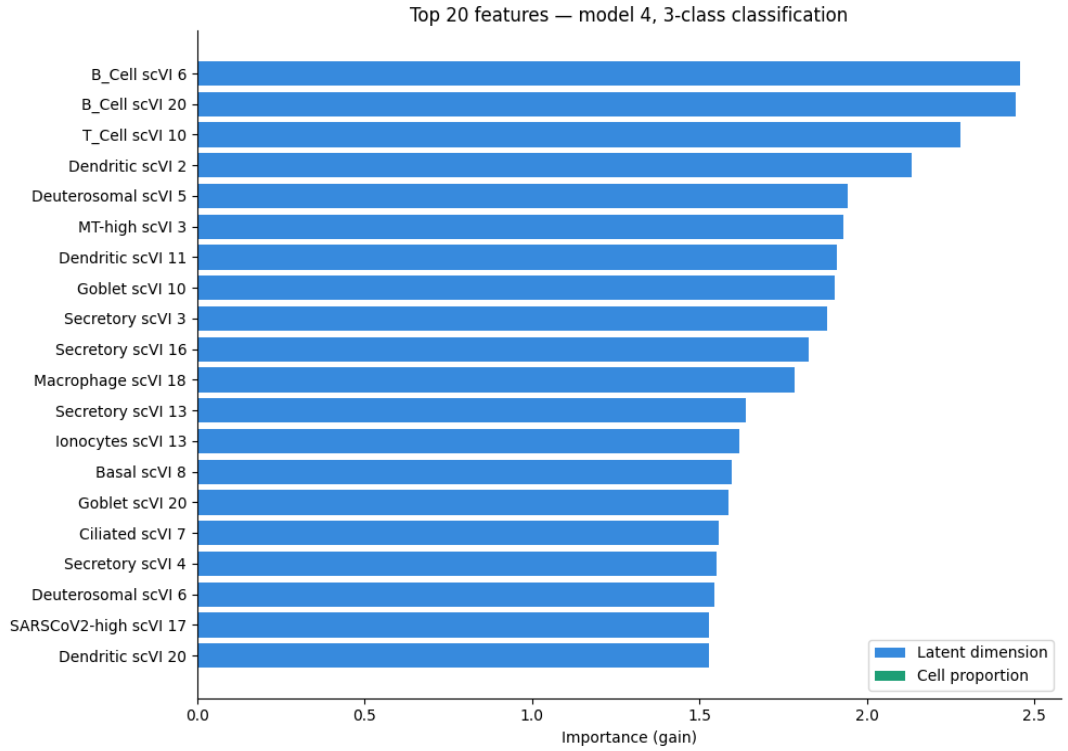
Two Classes:
- Mild (0-4)
- Severe (5-8)

Three Classes:
- Mild (0-3)
- Moderate (4-6)
- Severe (7-8)



Best 3-class bal. acc.
0.494
Participant per-celltype

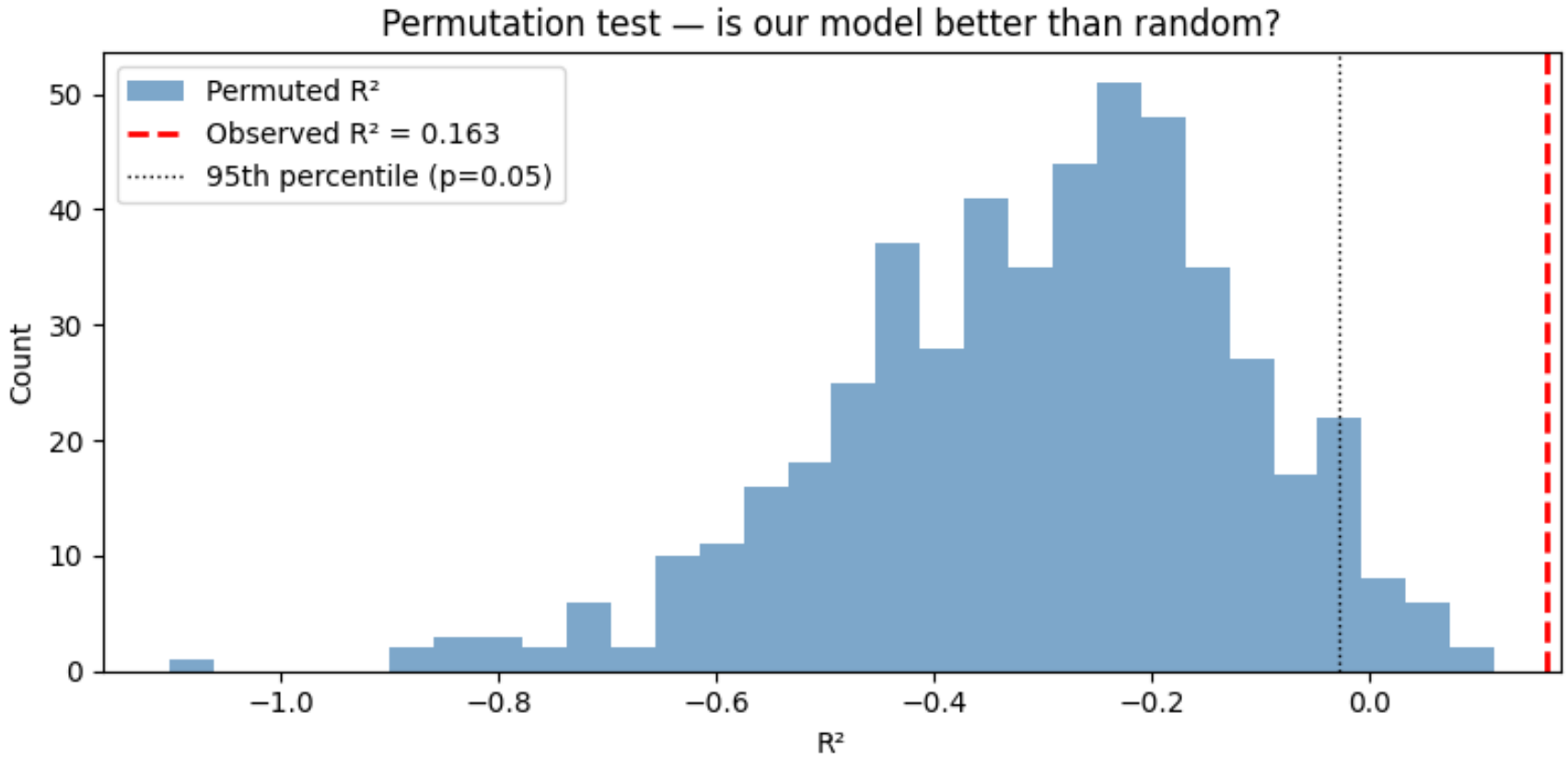
3-class p-value
0.016
Permutation test, n=500



Permutation Test scVI

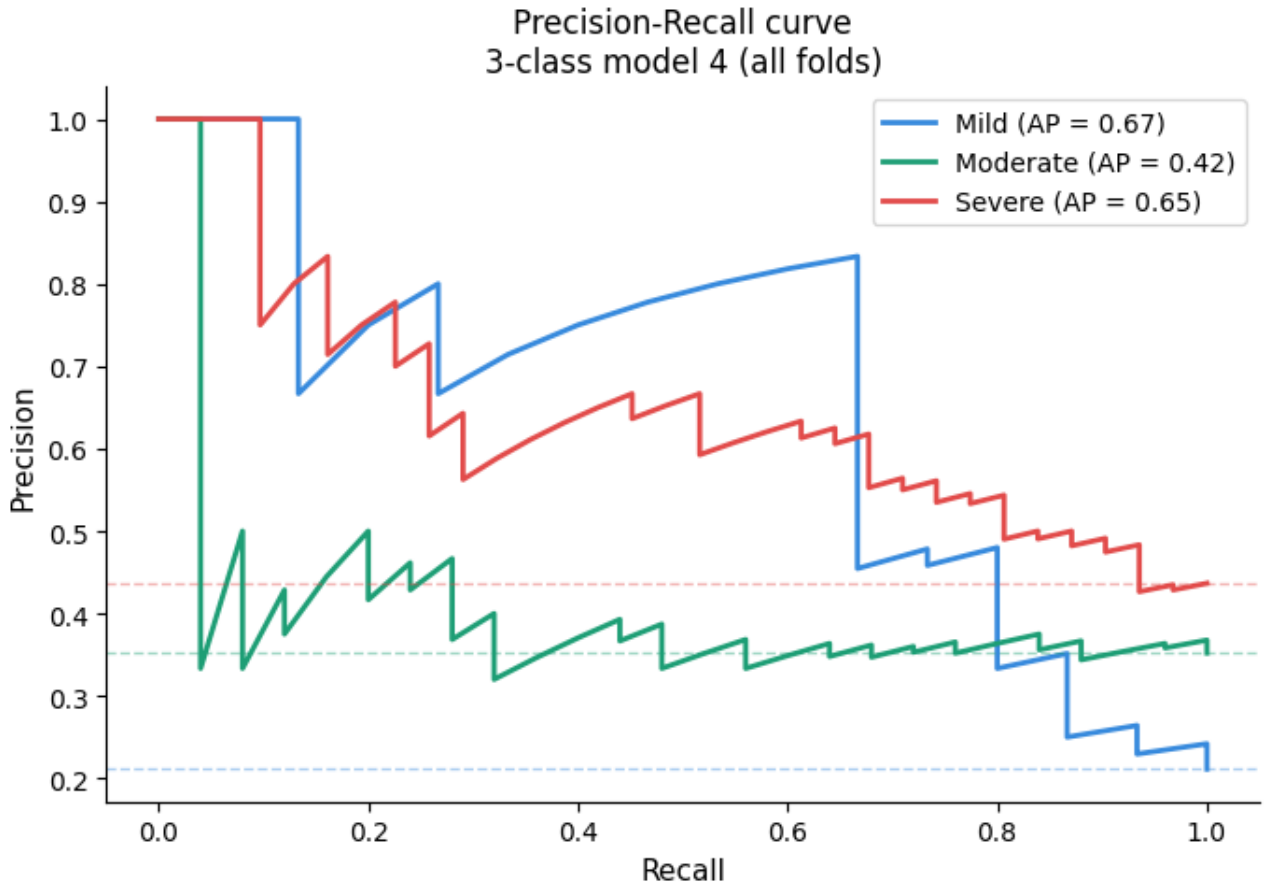
We performed Permutation Test (n_permutations = 500) for all models that showed a reasonable R2.

ScVI Regression model 4:
p-value:
< 2.00e-03



Precision-Recall Curve for best Classification model (scVI)

- The model distinguishes mild and severe cases well above chance (AP = 0.67 and 0.65 respectively), both substantially above their random baselines of 0.21 and 0.44
- Moderate cases are hardest to classify (AP = 0.42), expected since moderate sits between mild and severe and has the least distinct gene expression profile
- The stepwise appearance reflects the small dataset size (71 participants) each step represents one patient crossing the classification threshold



Limitations of each type of model (scVI)

(cell) Model 1: Cell latent only

- Predicts a patient-level label from individual cells — inherently noisy
- Cells from the same patient are not independent observations

→ StratifiedGroupKFold ensures no participant appears in both train and test

(cell) Model 2: Cell latent + cell type

- Model may learn cell composition differences rather than gene expression

→ Observed no improvement over model 1, concluded latent-only is the cleaner approach

(cell) Model 3: Participant mean latent

- Averaging blends biologically distinct cell types into one vector
- Only 71 samples

→ Poor results motivated building model 4 with per-celltype aggregation

Shallower trees and regularization to limit overfitting

(cell) Model 4: Participant per-celltype + composition

- 273 features, 71 samples — high overfitting risk
- Participants missing a cell type receive zeros

→ Regularization via max_depth=2, L1/L2 penalties, colsample_bytree=0.3

Participants missing a cell type receive zeros

Findings using ScVI autoencoder

- scVI latent representations *does* contain statistically significant information about COVID-19 severity
- Regression (R^2=0.163) and 3-class classification (balanced accuracy=0.494) both exceed chance:
  - Cell type proportion drive Regression and latent states drive Classification

Cell type Findings

- Dendritic cells are the most important cell type across both regression and classification
- Secretory cell proportion is a top feature in regression (As UMAP also suggests)
- B cells and T cells emerge as important in classification
- Macrophages appear consistently across both tasks

Methodological Findings

- Per-celltype aggregation consistently outperforms simple mean aggregation — preserving cell-type-resolved structure is important

What are GO-terms and why to incorporate them in ML?

GO terms are standardized, controlled categories used to describe the function of genes and gene products.

GO terms can be divided into 3 categories:

- Molecular Function (MF)- exact biochemical activity performed by the gene product
- Biological Process (BP)- The broader cellular or physiological goals accomplished by one or more molecular functions
- Cellular Component (CC): the specific location within a cell

GO solution: Each pathway score = mean z-scored expression of member genes