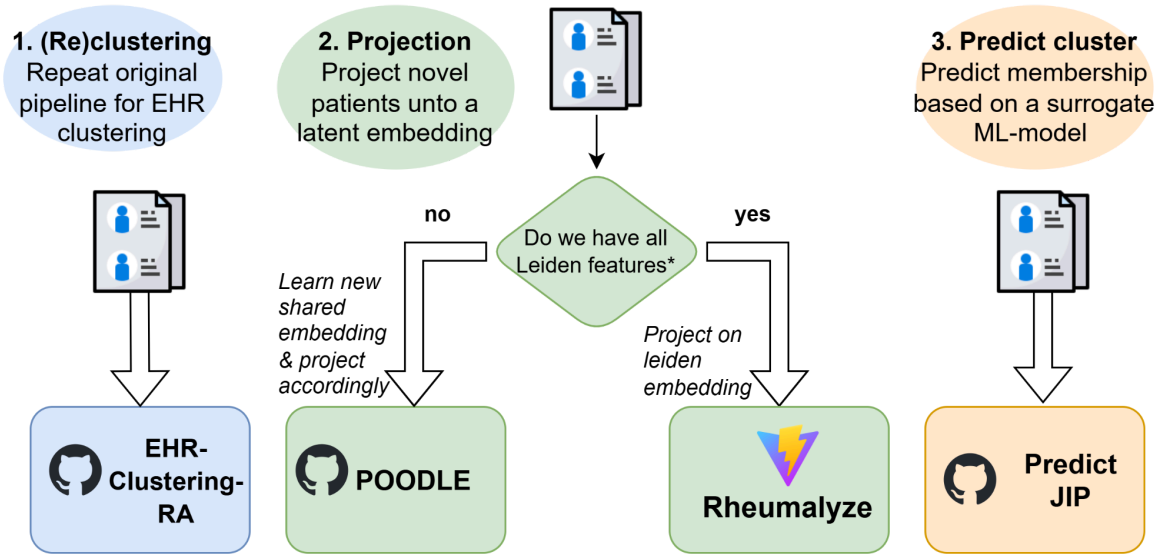
Manual: How to detect JIPs?

Datasets

Original Leiden EHR dataset
The processed Leiden EHR dataset (n=1387) can be found in SHARK at the following location:
exports/reum/tdmaarseveen/tdmaarseveen/github/PredictJIP/data/MMAE_clustering_270.csv

To get insight in every preprocessing step that lead to the creation of this dataset please consult the SOP RA clustering [1].

Methodology



Option 1: (Re)clustering

Run the entire cluster prediction pipeline again (warning this may result in new clusters entirely).

Our clustering pipeline consists of the following steps:

1. **Preprocessing**
   We preprocess both laboratory (numeric) data and clinical (categorical) information to ensure consistent formatting and quality.
2. **Learning the Latent Embedding**
   We identify the major axes of variation in the dataset and encode them using a MMAE.

The resulting latent embedding serves as the input space for clustering.

3. **Clustering With PhenoGraph**
   We apply PhenoGraph to the latent embedding and determine the optimal number of clusters based on cluster stability.
4. **Assessing Stability in Parameter Space**
   We evaluate robustness by varying clustering hyperparameters and assessing how consistent the resulting cluster assignments are.
   This can be visualized using a heatmap of Adjusted Rand Index (ARI) scores across configurations, allowing us to select clusters from a stable region.
5. **Assessing Stability in Subsample Space**
   We perform bootstrapping (typically across 1,000+ iterations) to evaluate how often pairs of patients co-occur in the same cluster across resampled datasets.
   High co-occurrence indicates stable, reliable clusters.
6. **Validate with clinical outcomes**
   We conduct a survival analysis to determine if our clusters correspond to clinically relevant outcomes (Methotrexate failure after 1 year & remission) and histological variation

References

Link to github repository with all code: https://github.com/levrex/EHR-Clustering-RA

For more information on the specific motivation behind certain steps, read the SOP [1]:

Option 2: projection

For option 2, patients are assigned to clusters (their JIP) based on their position within the learned latent embedding. This is a somewhat similar approach as the single cell atlas, but then for mapping the clinical diversity of patients instead.

We currently offer two approaches for this:

1. **POODLE** — a tool that learns a shared embedding (or shared product space) that captures a comparable representation using fewer variables. This allows you to assign patients to clusters even when your dataset contains a reduced set of input features. Link to Poodle: https://github.com/levrex/Poodle
   (On Shark: /exports/reum/tdmaarseveen/Poodle/poodle/)

For NORDSTAR projection in POODLE you need:

- DAS66/68 joints.
- Demographics: Age, Sex
- ESR (or CRP)
- RF, ACPA
- Thrombocytes
- Mean Corpuscular Volume (MCV)
- Hemoglobine (Hb)
- Leukocytes

2. **Rheumalyze** — a client-based web tool that runs the entire pipeline locally on your computer, ensuring that all patient data remains on your device and is therefore fully GDPR-compliant.

For Rheumalyze you need:
- DAS66/68 joints.
- Demographics: Age, Sex
- ESR,
- RF, ACPA
- Thrombocytes
- Mean Corpuscular Volume (MCV)
- Hemoglobine (Hb)
- Leukocytes (White Blood Cells/ WBC)

This tool was developed by Nick Bos (previous bachelor student)

Link to rheumalyze client-based webtool:
https://knevel-lab.github.io/Rheumalyze/

Link to github Rheumalyze repository:
https://github.com/Knevel-Lab/Rheumalyze/

Option 3: Prediction

To make our clusters easier to use across different centers, we developed machine-learning surrogate models that can predict a patient's JIP using a smaller set of variables. Because the primary axes of variation in our data are the **hand–feet differentiation** and the **oligo–poly joint involvement** (few vs. many affected joints), we found that JIP can be predicted reasonably well even with limited information.

We note that the cluster labels used below have not all been named according to their JIP. For clarity, the numeric-notation is as follows:
- cluster 0 = JIP-foot
- cluster 1 = JIP-oligo
- cluster 2 = JIP-hand
- cluster 3 = JIP-poly

We built **three probabilistic (surrogate) models**, each designed around the types of joint assessments that centers most commonly collect. These models output **class probabilities**, enabling users to choose decision thresholds that best fit their desired balance of precision, sensitivity, or overall accuracy. *However, note that applying a very stringent cut-off may result in patients not being assigned to any cluster at all.*

The three models—each validated on a hold-out set—are based on the following joint schemes:

- **DAS44**
- **DAS66/68 (ideal)**
- **DAS28 (suboptimal)**

For DAS44 we only have the pip (hand), mcp, ip (hand), mtp, wrist, elbow, shoulder, knee, ankle, sternoclaviculair, acromioclaviaculair

For DAS66 we have more joints in the feet, like pip (feet), dip (feet) also dip (hand), hip (only tenderness)

For DAS28, we only have the joints in the hand, elbow, shoulder, wrist & the knee. But no joints in the feet. Hence this model is suboptimal.

DAS44

The DAS44 model is based on 44 joint scheme, which is often measured in outpatient clinic. One limitation, is that it only counts MTP joints in the feet, and thus focuses more on the upper extremities/ smaller joints in the hand.

Example script:
To run DAS44 model see an example in Github:
https://github.com/levrex/PredictJIP/blob/main/notebooks/Example_notebook_das44.ipynb

Model Location:
https://github.com/levrex/PredictJIP/models/das44/JIP_xgb_pred_44.pk

Dataset

- Demographics: age, sex
- Serology: ACPA and RF status
- Tenderness per joint (44 joint scheme)
- Swelling per joint (44 joint scheme)
- Timepoints: baseline pre-DMARD

Features

The DAS44 model is based on 16 features al derived from the dataset

Joint regions (based on 44-joint scheme):

1. 'SJC_FOOT' (**only** mtp & ankle)
2. 'SJC_HAND' (pip, mcp, ip, wrist)
3. 'TJC_FOOT' (**only** mtp & ankle)
4. 'TJC_HAND' (pip, mcp, ip, wrist)
5. 'SJC44'
6. 'TJC44'
7. 'Small44'
8. 'Big44'

9. 'bigFrac' (fraction of big joints versus small)
10. 'handFrac', (fraction of joints from hand versus foot)

Demographics :

11. 'Age'
12. 'Sex',

Autoantibody:

13. 'RF'
14. 'aCCP'

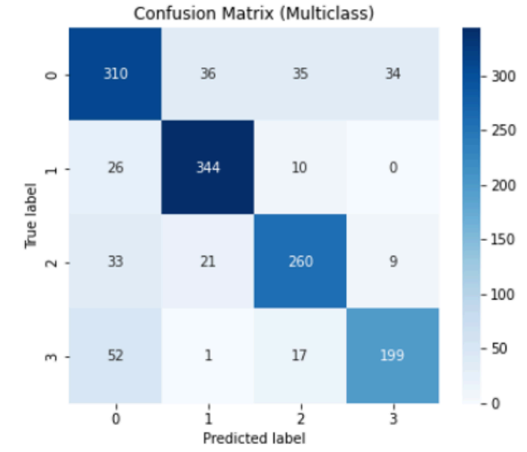
Ensemble features (learned w/ KAN):

15. 'pred_foot'
16. ‘pred_poly’

Location of KAN-encoder (you can simply copy/paste functions into python):
https://github.com/levrex/PredictJIP/models/das44/kan_encoders44.py

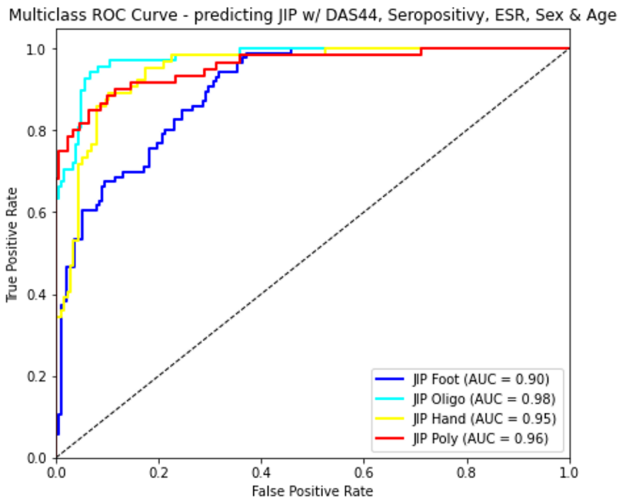
Performance on hold out set

For the DAS44 model, the macro-average accuracy = 80%

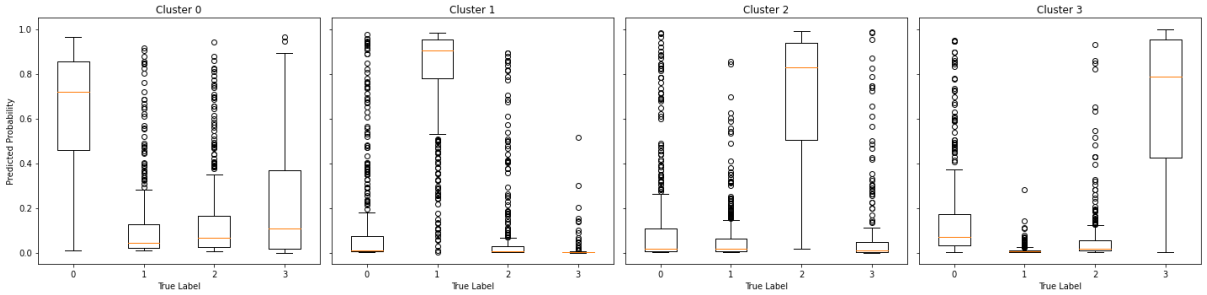


	sens	spec	ppv	npv	accuracy
0	0.747	0.886	0.736	0.891	0.844
1	0.905	0.942	0.856	0.963	0.932
2	0.805	0.942	0.807	0.941	0.910
3	0.740	0.962	0.822	0.939	0.919
macro-avg	0.799	0.933	0.805	0.934	0.802

Accuracy: 0.802451333813987



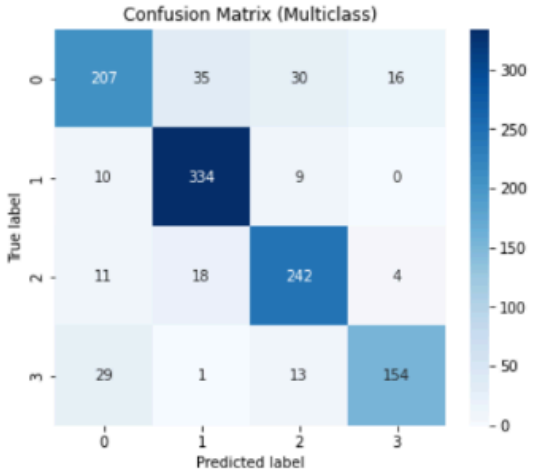
However you can adjust with threshold:



Threshold to reach minimal PPV of 0.80

For example precision, if you desire a PPV of 0.80, you could set a threshold at 0.72

```
Keeping 1113 of 1387 samples (80.2%)
Confusion matrix for confident predictions (threshold=0.72):
```




```
Accuracy: 0.8418688230008985
            sens   spec   ppv   npv  accuracy
0          0.719  0.939  0.805  0.905     0.882
1          0.946  0.929  0.861  0.974     0.934
2          0.880  0.938  0.823  0.960     0.924
3          0.782  0.978  0.885  0.954     0.943
macro-avg  0.832  0.946  0.844  0.948     0.842
```

DAS66

To compute the full DAS66 (in the ideal scenario), we require detailed information on all individual joints, relevant laboratory values, demographic data, autoantibody status, and several engineered features derived from the joint assessment scheme.

Example model:
To run DAS66 model see an example in Github:
https://github.com/levrex/PredictJIP/models/das66/JIP_xgb_pred_66.pk

Example script:
To run DAS66 model see an example in Github:
https://github.com/levrex/PredictJIP/blob/main/notebooks/Example_notebook_das66.ipynb

Dataset

- Demographics: age, sex
- Serology: ACPA and RF status
- Lab values: Hemoglobine, Leukocytes, Trombocytes
- Tenderness per joint (68 joint scheme)
- Swelling per joint (66 joint scheme)
- Individual joint involvement information

Features

For the DAS66 we need all individual joints plus some lab values

Joint regions (based on 44-joint scheme):

1. 'SJC_FOOT' (**only** mtp & ankle)
2. 'SJC_HAND' (pip, mcp, ip, wrist)
3. 'TJC_FOOT' (**only** mtp & ankle)
4. 'TJC_HAND' (pip, mcp, ip, wrist)
5. 'SJC66'
6. 'TJC66'
7. 'SJC44'
8. 'TJC44'
9. 'SJC28'
10. 'TJC28'
11. 'Small66'
12. 'Big66'
13. 'bigFrac' (fraction of big joints versus small)
17. 'handFrac', (fraction of joints from hand versus foot)

Lab values:

18. ‘Hb’ (hemoglobine)
19. ‘Trom’ (trombocytes)
20. ‘Leuko’ (leukocytes)

Demographics :

21. 'Age'
22. 'Sex',

Autoantibody:

23. 'RF'
24. 'aCCP'

Individual joints

25. SJC66, TJC68 (68, because it also includes tender Hip Left & right)

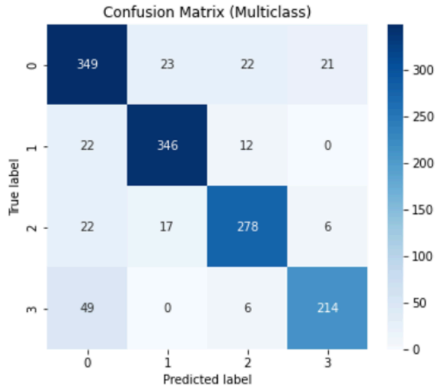
Ensemble features (learned w/ KAN):

26. 'pred_foot'

Location of KAN-encoder (you can simply copy/paste functions into python):
https://github.com/levrex/PredictJIP/models/das66/kan_encoders66.py

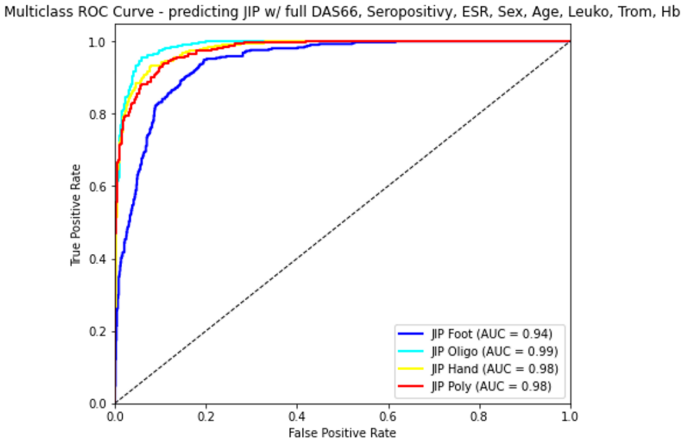
Performance on hold out set

For the DAS66 model we reach an accuracy of 86% with the surrogate model on the hold-out set.

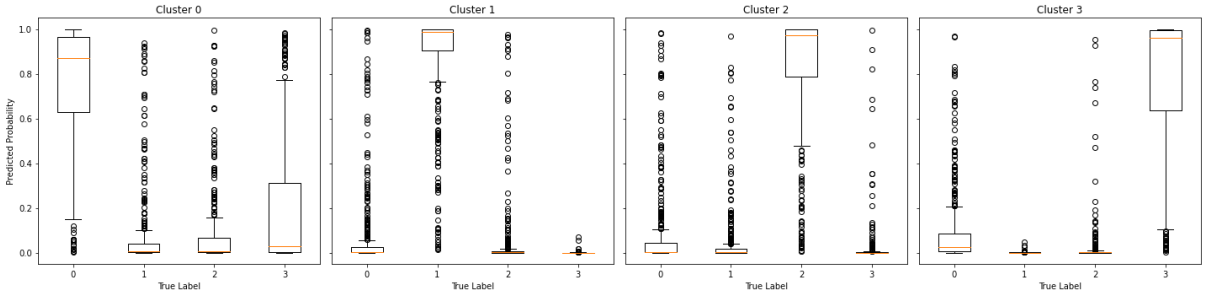


	sens	spec	ppv	npv	accuracy
0	0.841	0.904	0.790	0.930	0.885
1	0.911	0.960	0.896	0.966	0.947
2	0.861	0.962	0.874	0.958	0.939
3	0.796	0.976	0.888	0.952	0.941
macro-avg	0.852	0.951	0.862	0.952	0.856

Accuracy: 0.855803893294881

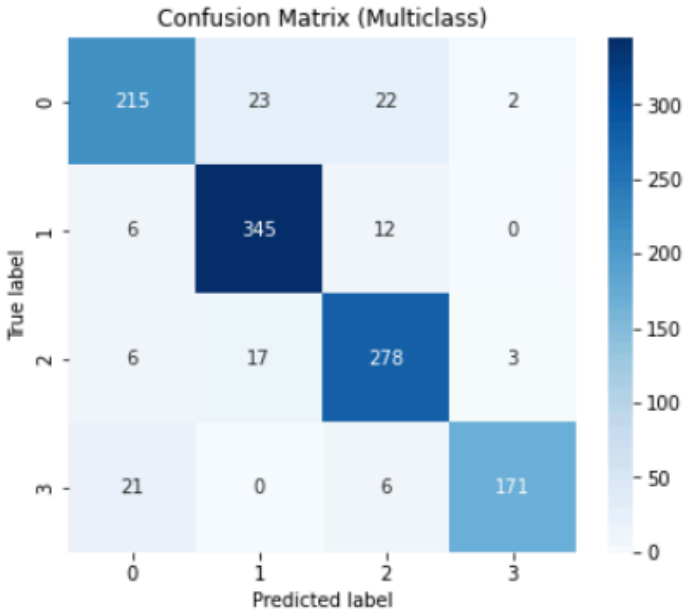


However, you can further adjust with threshold given that it is a probabilistic model. See probabilities per cluster in the boxplot below:



Threshold to reach minimal Accuracy of 0.90 & PPV > 0.85 per cluster

```
Keeping 1127 of 1387 samples (81.3%)
[0.87, 0.45, 0.45, 0.85]
Confusion matrix for confident predictions (threshold:[0.87, 0.45, 0.45, 0.85])
```




```
Accuracy: 0.8952972493345164
```

	sens	spec	ppv	npv	accuracy
0	0.821	0.962	0.867	0.947	0.929
1	0.950	0.948	0.896	0.976	0.949
2	0.914	0.951	0.874	0.968	0.941
3	0.864	0.995	0.972	0.972	0.972
macro-avg	0.887	0.964	0.902	0.965	0.895

DAS28 (not ideal)

The DAS28 model is not ideal for predicting the JIPs, as it does not take the foot joints into account, this makes it difficult to predict JIP-foot phenotype which is (as the name suggests) particularly characterized by involvement in the foot. Nonetheless we may still be able to predict some phenotypes - with a prediction model , using only DAS28 joint scheme.

We found that we can still get accuracy of ~64% (and higher if we focus on other JIPs), if we tinker with thresholds we can increase PPV to 70% for all JIPs. You might be able to classify some, to include additional data from other centers. However one should be careful with JIP classification based on such a low-resolution profile of a patient.

Example script:
To run DAS44 model see an example in Github:
https://github.com/levrex/PredictJIP/blob/main/notebooks/Example_notebook_das28.ipynb

Model Location:
https://github.com/levrex/PredictJIP/models/das28/JIP_xgb_pred_28.pk

Dataset

- Demographics: age, sex
- Serology: ACPA and RF status
- Tenderness per joint (28 joint scheme)
- Swelling per joint (28 joint scheme)
- Timepoints: baseline pre-DMARD

Features

The DAS28 model is based on 12 features derived from the dataset:

Joint regions (only 28-joints):

1. 'JC_HAND' (swelling + tenderness)
2. 'SJC_HAND' (pip, mcp, ip, wrist)
3. 'TJC_HAND' (pip, mcp, ip, wrist)
4. 'SJC28'
5. 'TJC28'
6. 'Small28'
7. 'Big28'
8. 'bigFrac28' (fraction of big joints versus small)

Demographics :

9. 'Age'
10. 'Sex',

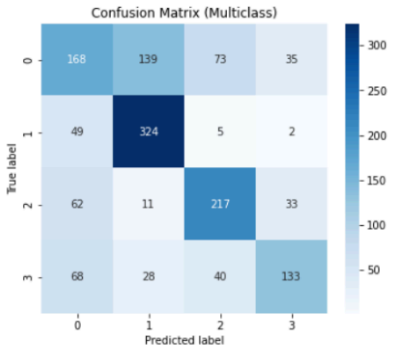
Autoantibody status:

11. 'RF'

12. 'aCCP'

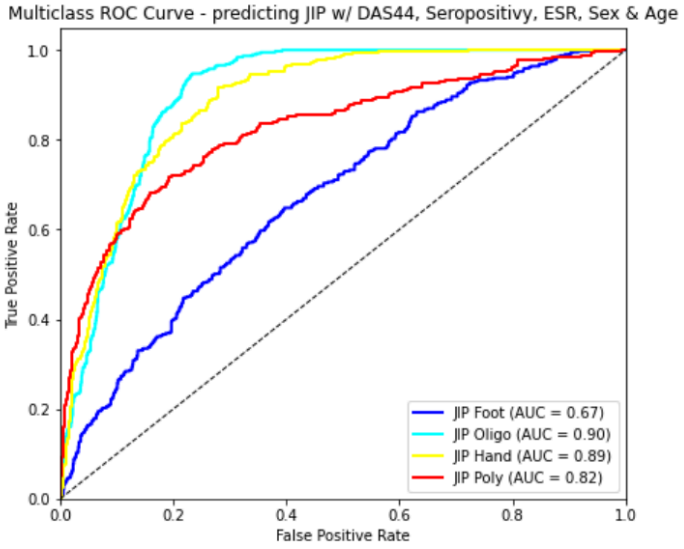
Performance on hold out set

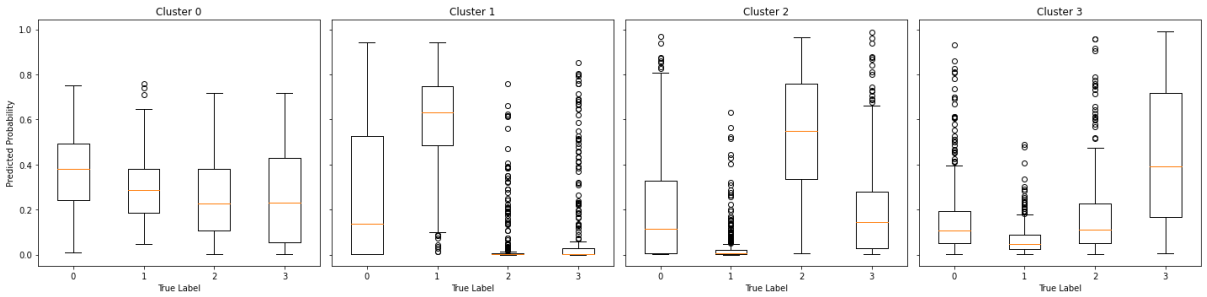
<Figure size 432x288 with 0 Axes>



	sens	spec	ppv	npv	accuracy
0	0.405	0.816	0.484	0.762	0.693
1	0.853	0.823	0.645	0.937	0.831
2	0.672	0.889	0.648	0.899	0.839
3	0.494	0.937	0.655	0.885	0.851
macro-avg	0.606	0.866	0.608	0.871	0.607

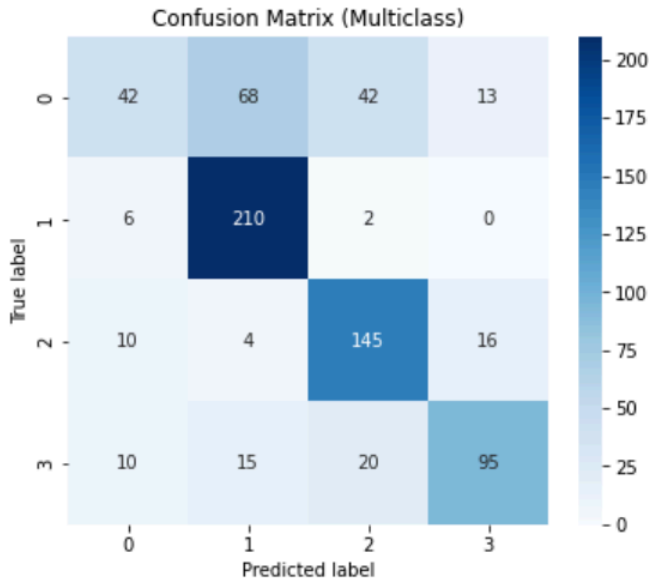
Accuracy: 0.6070656092285508





Threshold to reach macro-average accuracy of 70

```
Keeping 698 of 1387 samples (50.3%)
Confusion matrix for confident predictions (threshold=0.70):
```




```
Accuracy: 0.7048710601719198
             sens   spec    ppv    npv  accuracy
0           0.255  0.951  0.618  0.805     0.787
1           0.963  0.819  0.707  0.980     0.864
2           0.829  0.878  0.694  0.939     0.865
3           0.679  0.948  0.766  0.922     0.894
macro-avg   0.681  0.899  0.696  0.911     0.705
```

References

[1] SOP RA clustering (featuring specific paths to different preprocessed datasets & code) https://lumconline.sharepoint.com/:w:/r/sites/TeamsREUMKnevelGroup/Gedeelde%20documenten/General/SOP/Secondary%20Care%20(Leiden%20EHR)%20SOP/SOP%20RA%20Clustering.docx?d=wb88d3207556a4890a34dda4d33e385ed&csf=1&web=1&e=QcHGvh