[2021.11.09]

Bulky mediastinal lymphoma classification with ML-techniques

Activity status n. 2

Matteo Barbetti

University of Florence
INFN - Firenze

From previous meetings: *objective*

Classification of *bulky mediastinal lymphoma* using **Machine Learning** techniques applied to PET and CT images.

Lymphoma types:

1. *Hodgkin* (HL)
2. *Gray Zone* (GZ)
3. *Primary Mediastinal Lymphoma* (PML)

PLANNED

From low-level data

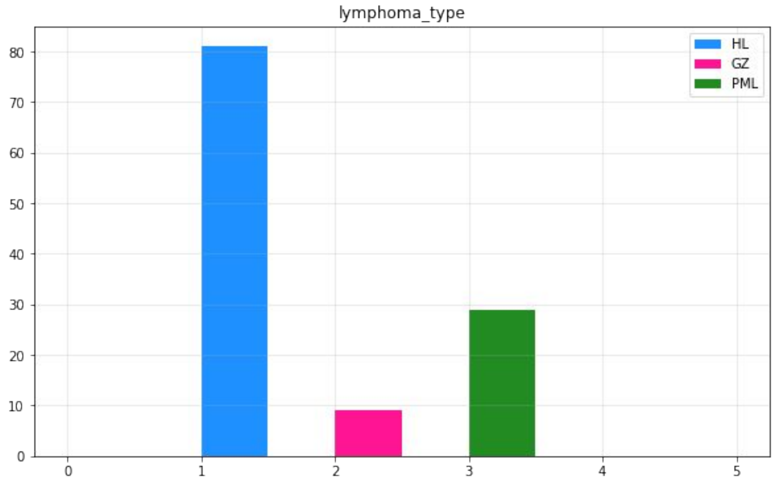
- Data: PET and CT images
- Technique: Image classification
- Algorithm: *Convolutional Neural Nets*

IN PROGRESS

From high-level data

- Data: features drawn by LIFEx from images
- Technique: Binary and multiclass classification
- Algorithm: *Logistic regression* + *Random Forest*

From previous meetings: *learned lessons*



- Dataset heavily **unbalanced**
  - HL-class is over-represented w.r.t. the other two ones
  - classification suffers from unbalancing
- **Binary classification** looks promising
  - HL and non-HL classification
  - dataset a bit more balanced
- **Multiclass classification** fails
  - HL, GZ and PML classification
  - only HL-class is well-identified by models
- GZ-class is a *middle-class*
  - **classification uncertainty** btw HL and PML
  - only HL and PML are true disjoint classes

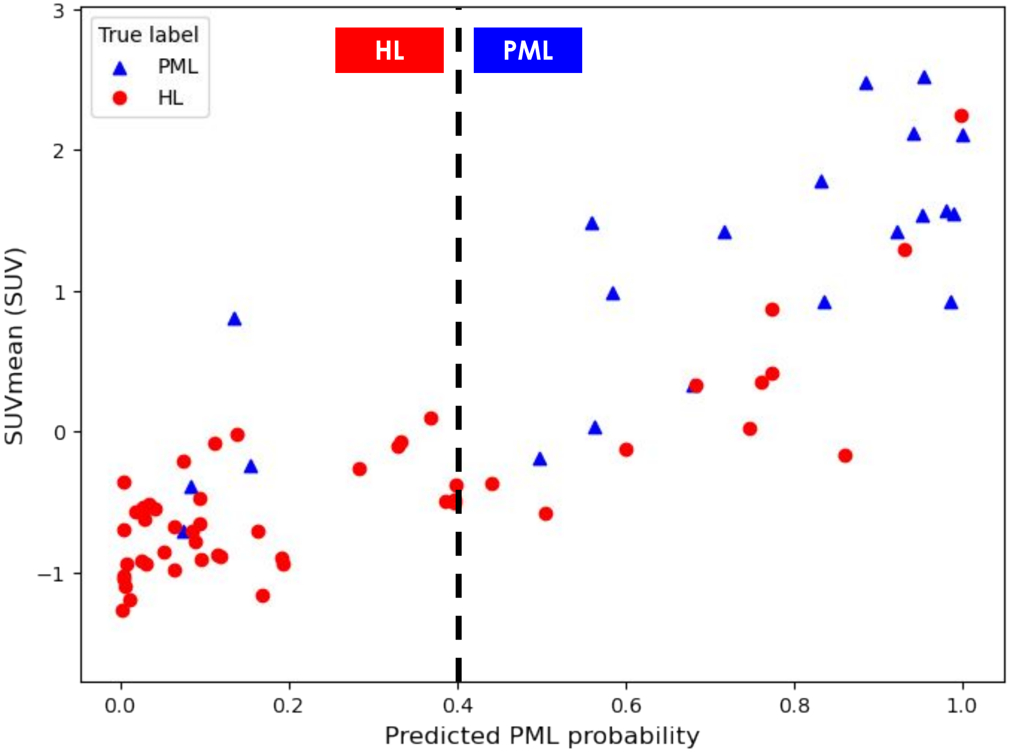
From previous meetings: *implemented suggestions*

- Classifiers **performance improvement**
  - custom probability threshold to make decisions
  - more robustness against unbalanced datasets
  - ROC AUC score for optimization
- New **multiclass classification** implementation
  - GZ-class treated as a classification uncertainty
  - binary classifiers ***promoted*** to multiclass classifiers

mbarbetti/lymphoma-classification

Binary classification

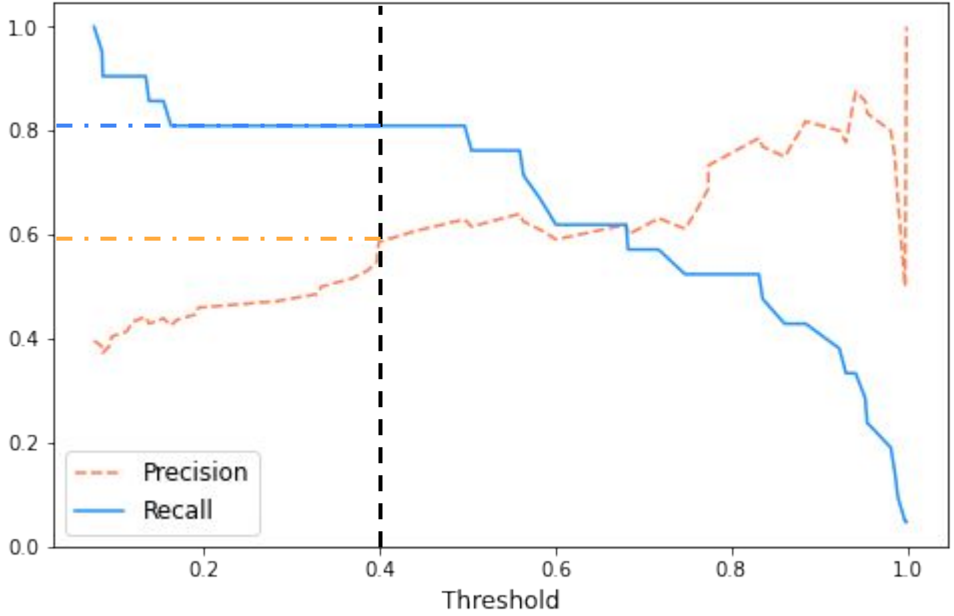
Decision rules: *custom probability threshold*



A trained classifier *induces* a 1-D space (represented by the predicted probability) where items belonging to different classes are separated as much as possible: **more powerful classifiers result in greater separations**.

Classifiers typically adopt the standard threshold of 0.5 in predicted probability to make a decision. Customizing this threshold allows to obtain models **more robust** against unbalanced dataset, and to have **more control** on models performance.

Decision rules: *precision/recall tradeoff*



Moving the threshold at will, one can obtain <u>any target score desired</u> (recall or precision).

- high recall → low precision
  - **high TPR → low TNR**
- high precision → low recall
  - **high TNR → low TPR**

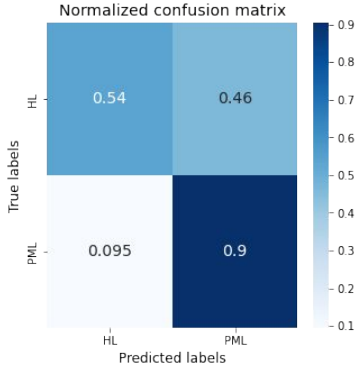
$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

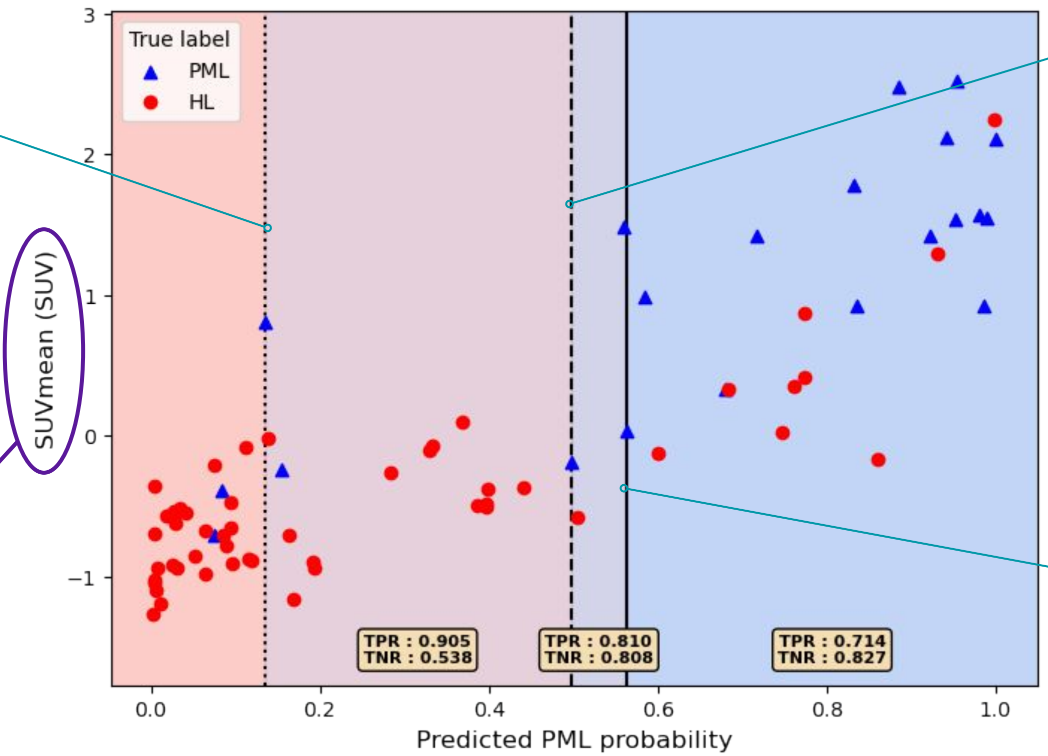
$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

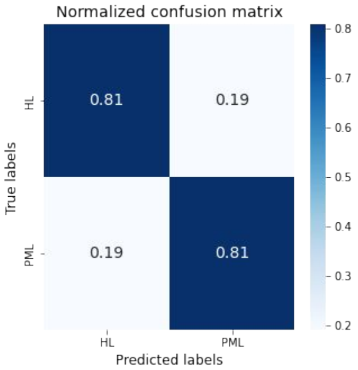
Predicted labels

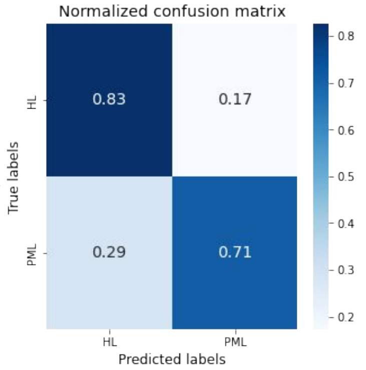
Actual labels	Negative	Positive
Negative	**TN**	**FP**
Positive	**FN**	**TP**

Logistic regression: *decision regions*



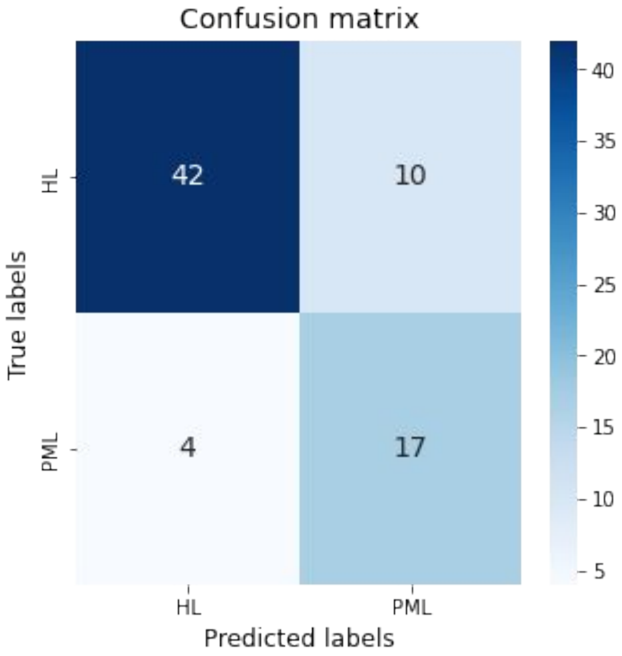


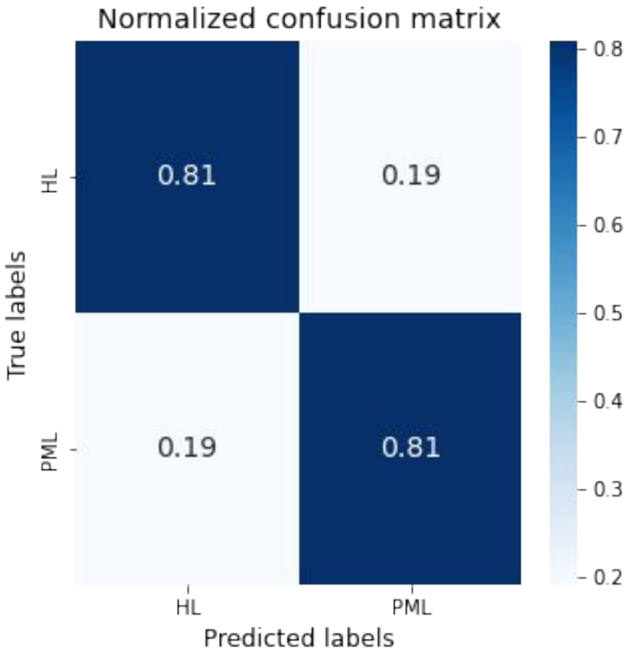




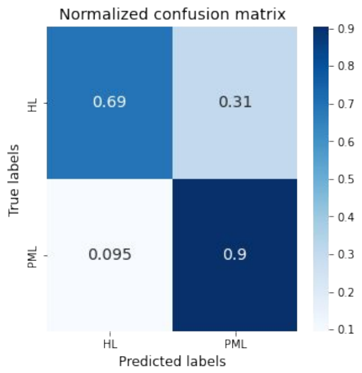
Most important feature in classification

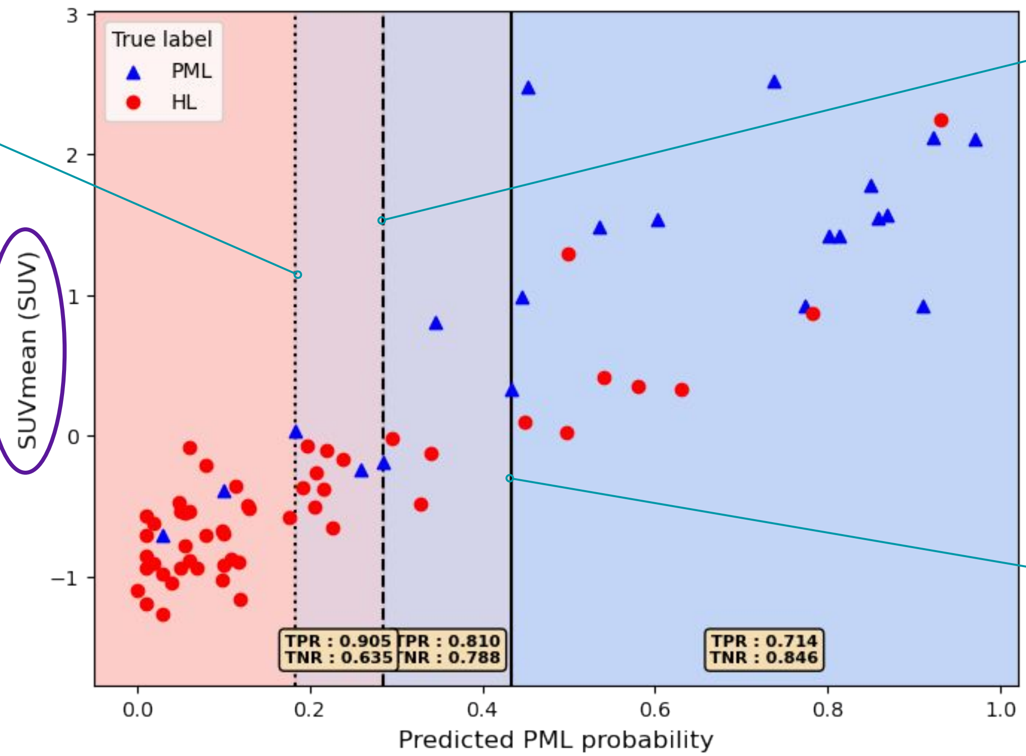
Logistic regression: *performance*

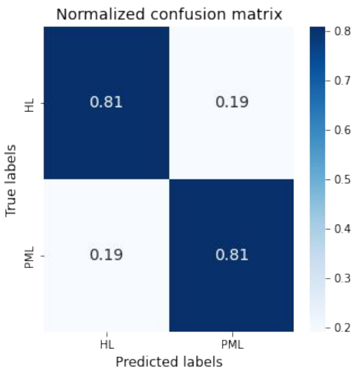


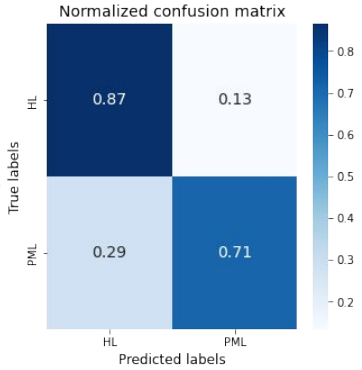


Random Forest: *decision regions*



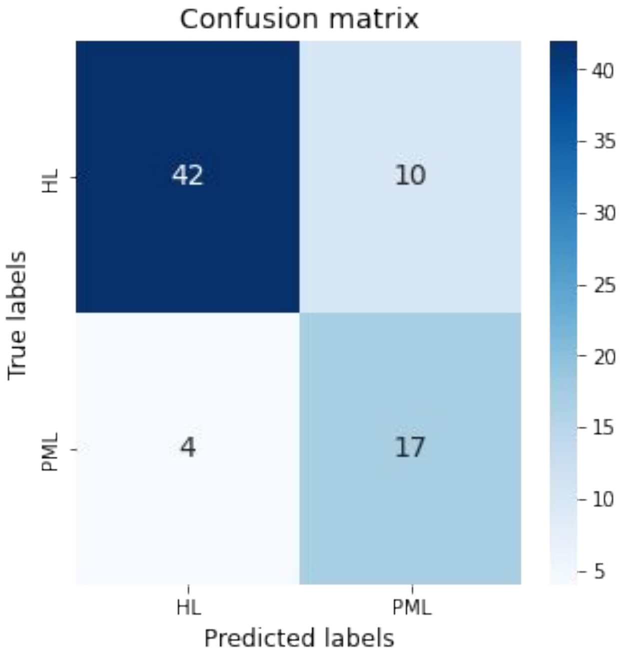


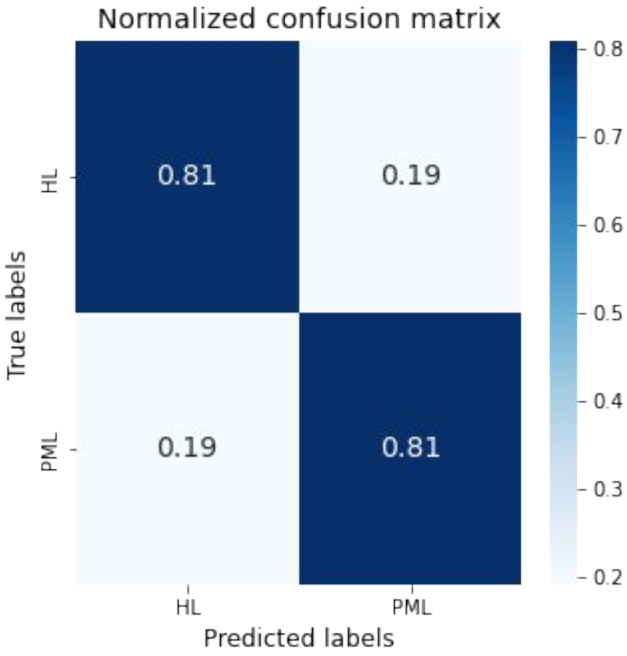




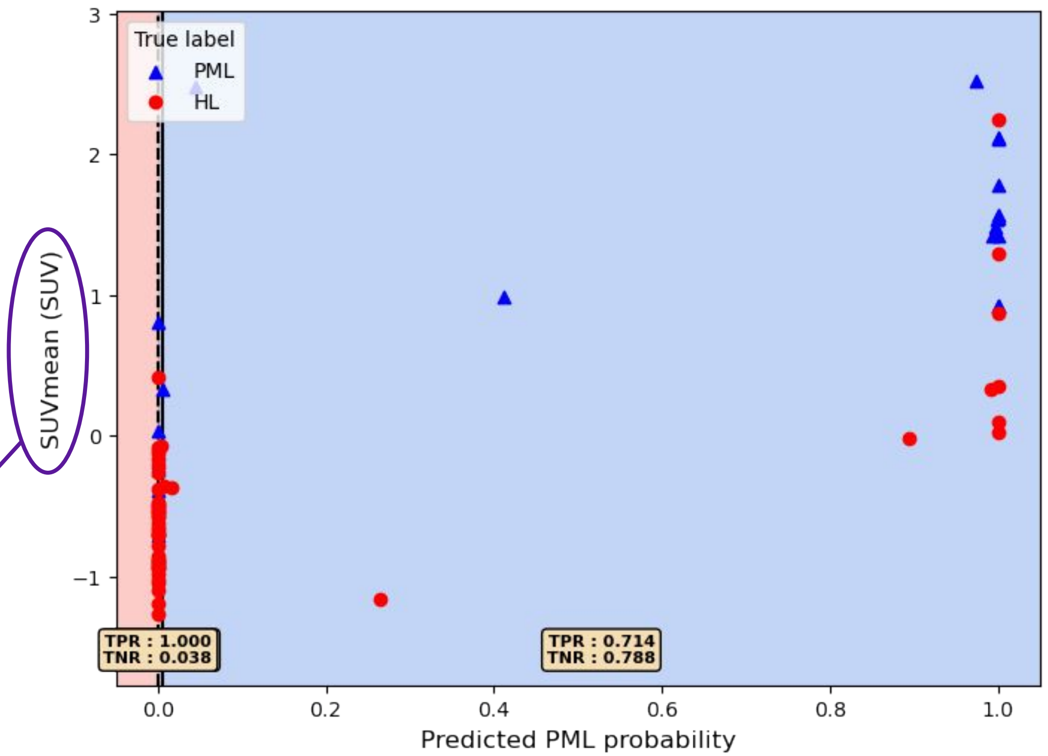
Most important feature in classification

Random Forest: *performance*



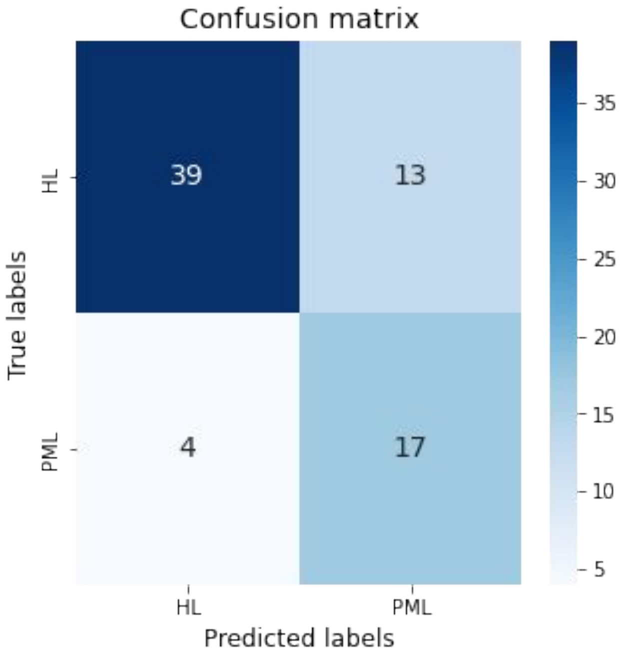


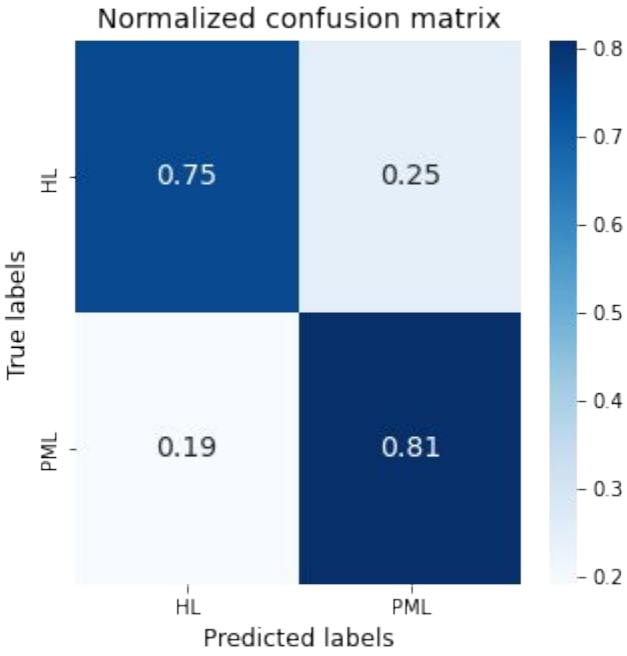
Gradient BDT: *decision regions*



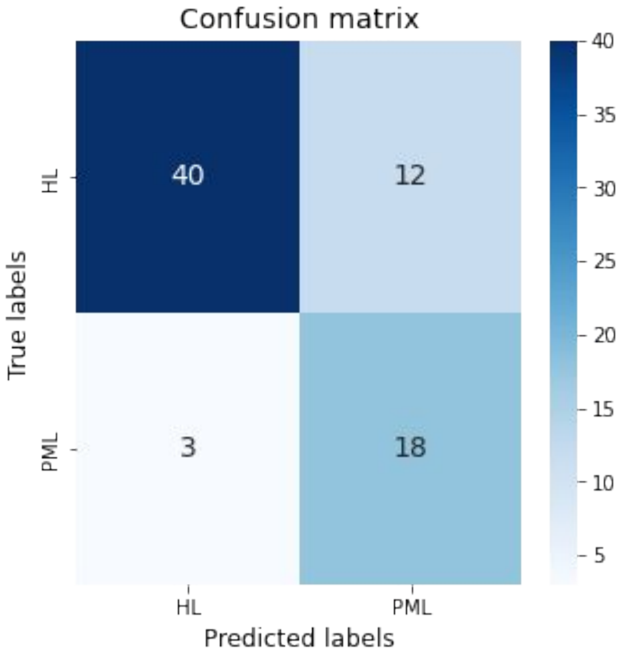
Most important feature in classification

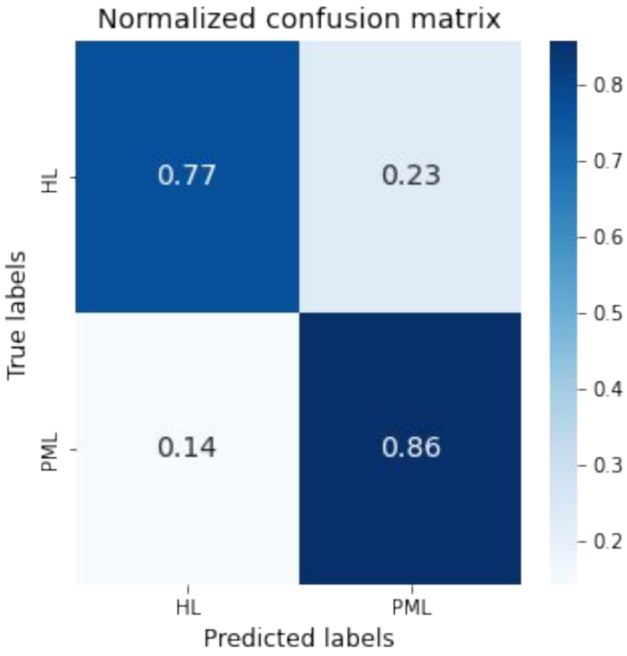
Gradient BDT: *performance*



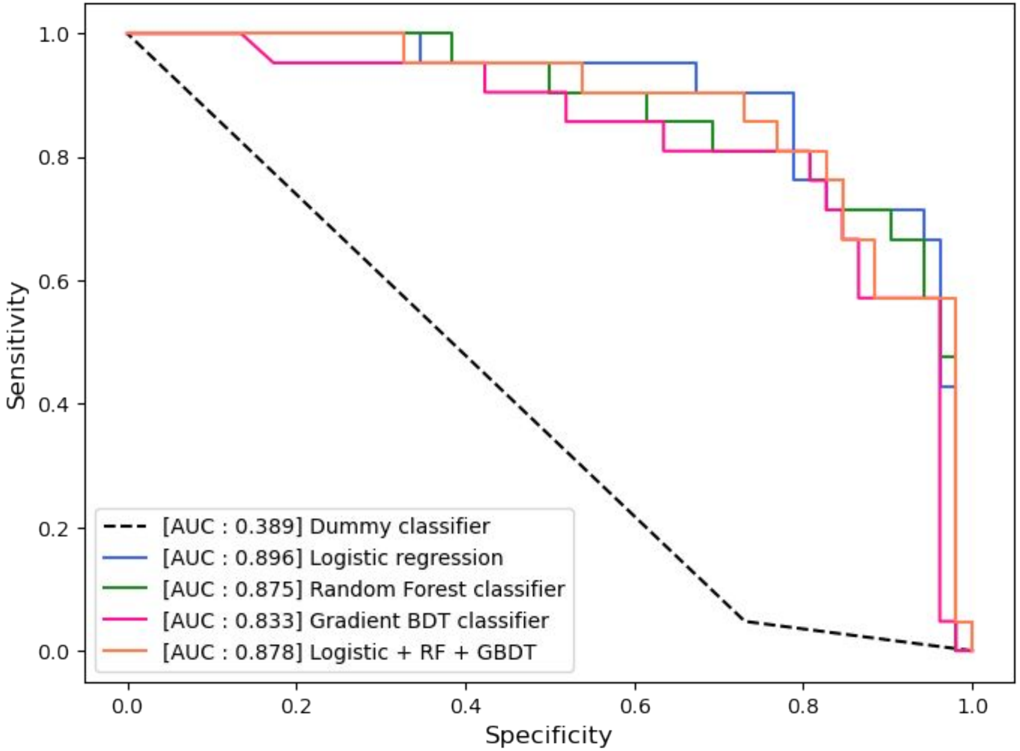


Model combination: *performance*

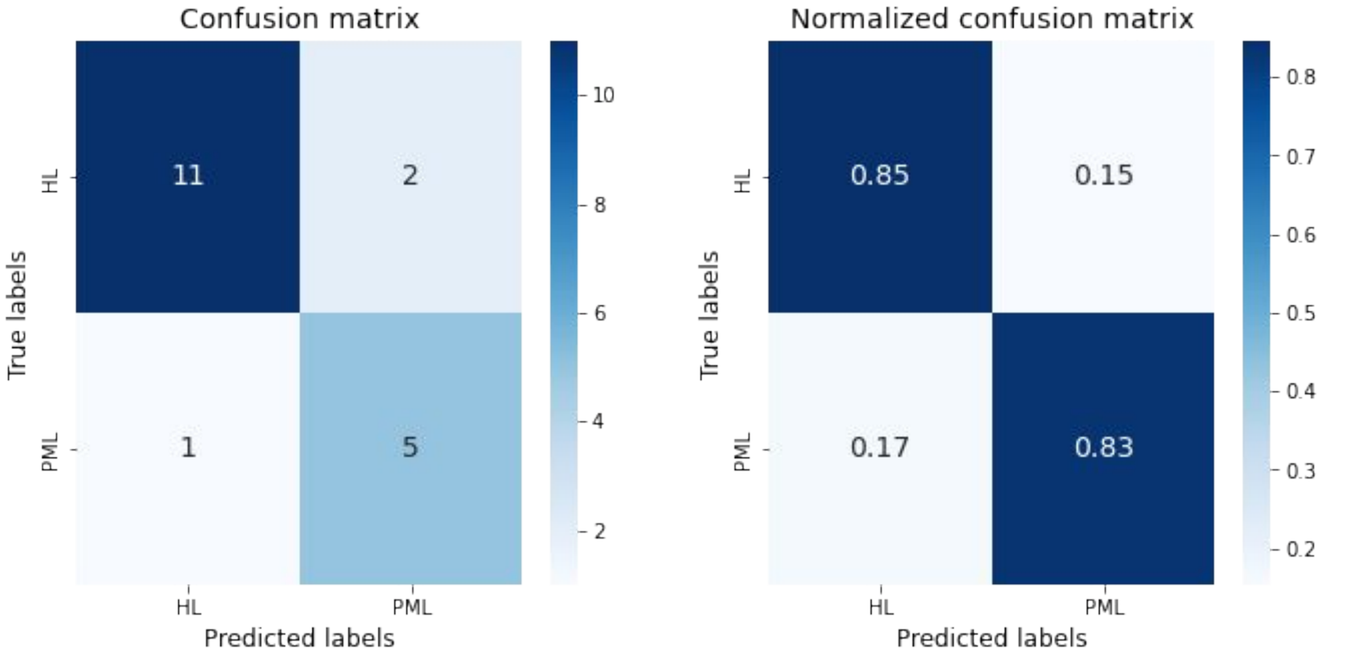




Models comparison

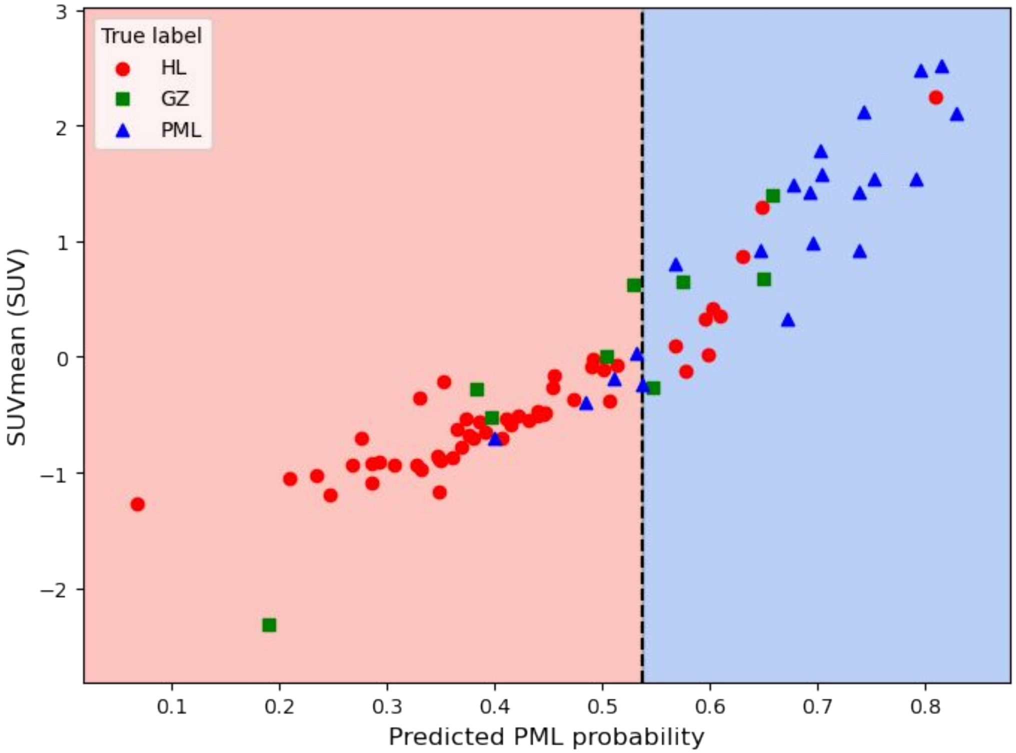


Results on the test-set



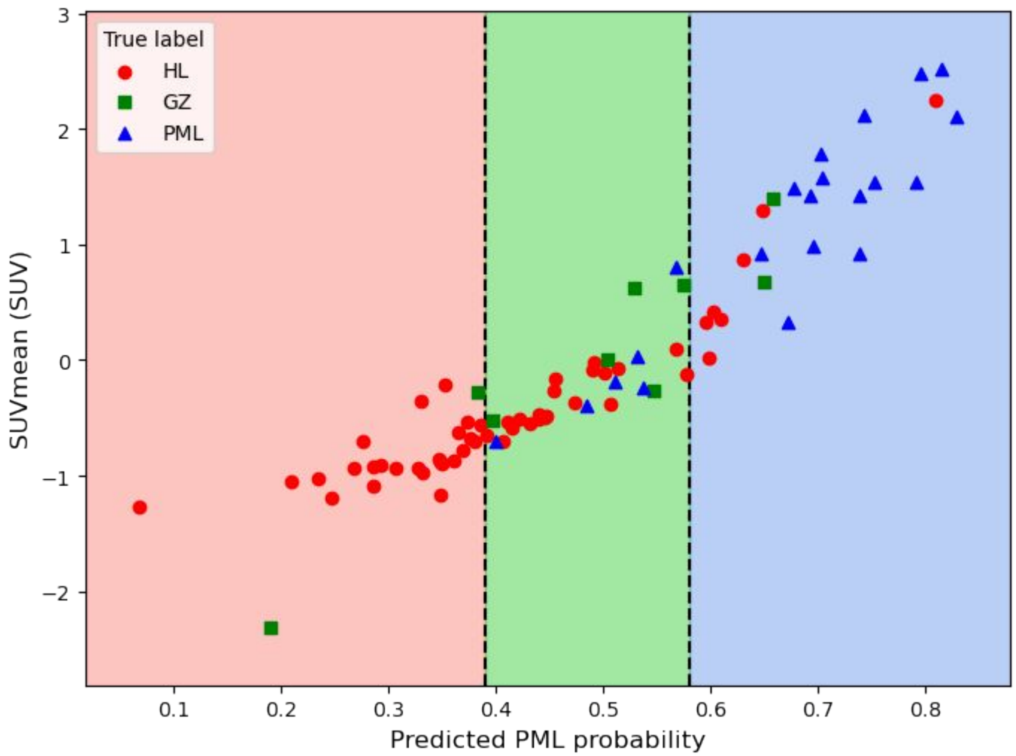
Multiclass classification

The Gray Zone class



- Binary classifier
  - trained to **separate HL-PML classes**
  - good performance also on the test-set
- Behaviour with the third class
  - half of GZ-items are predicted as HL
  - GZ-items have **mixed characteristics** btw HL and PML
  - GZ is a sort of *middle-class*

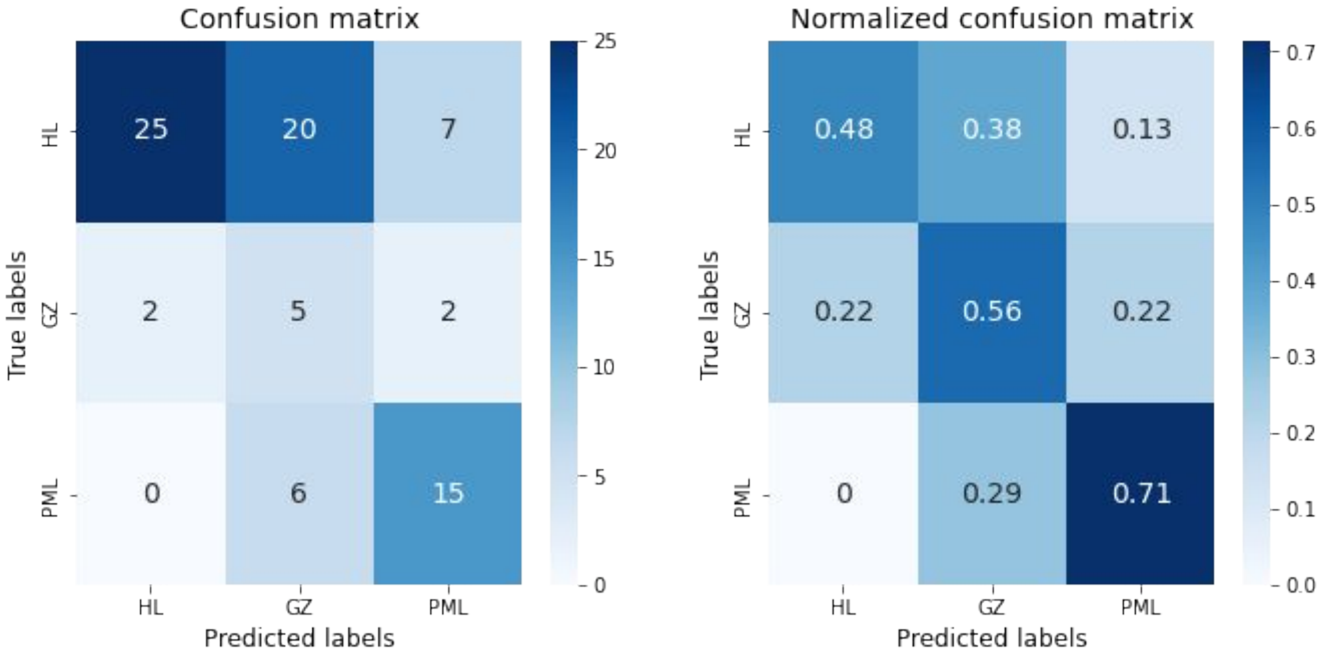
Promotion to multiclass classification



- 2 thresholds → **3 decision regions**
  - outer regions for HL and PML predictions
  - middle region for GZ predictions
- **lower performance** w.r.t. binary classifier
  - promising performance w.r.t. the multiclass classifier trained with HL, GZ and PML as disjoint classes

Custom thresholds can be used to promote trained binary classifiers into multiclass classifiers!

Multiclass classification: *performance*



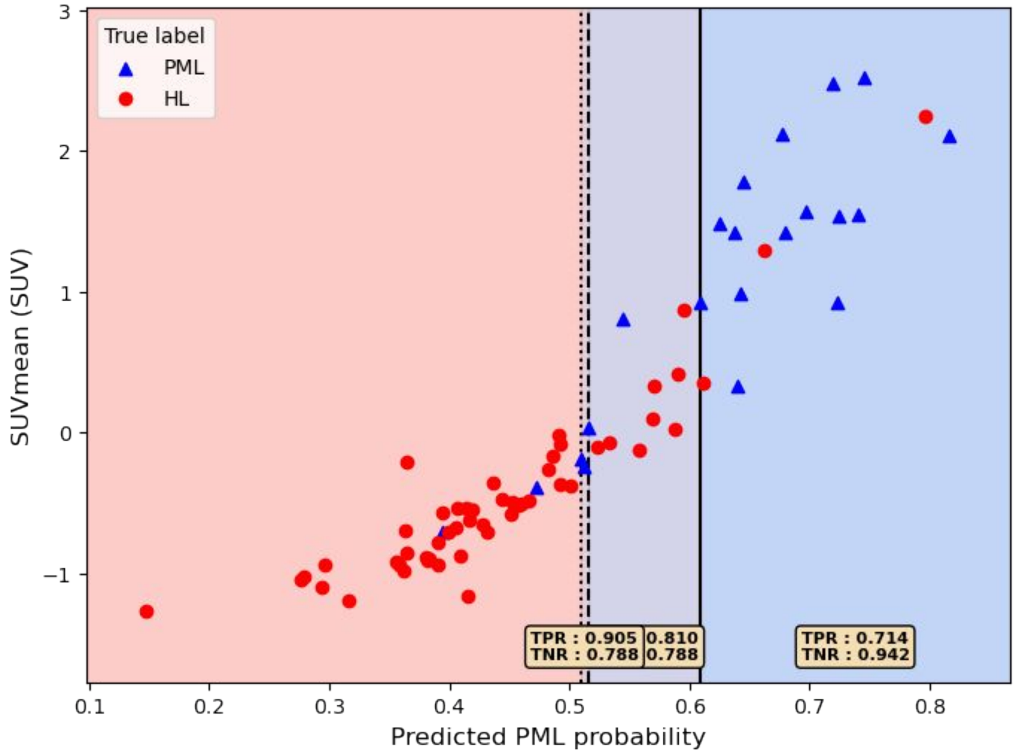
Conclusion

- Preliminary studies
- Still need for taking a look at literature
- There is room for improvements
  - **increasing** the dataset → more performant classifiers
  - **balancing** the dataset → more sensitive classifiers
  - working with **low-level data** → *image classification* techniques
- Necessity to define a final pipeline for models optimization
  - Accuracy? Precision? Recall? **ROC AUC**?
- Necessity to add **statistical errors** for performance evaluation

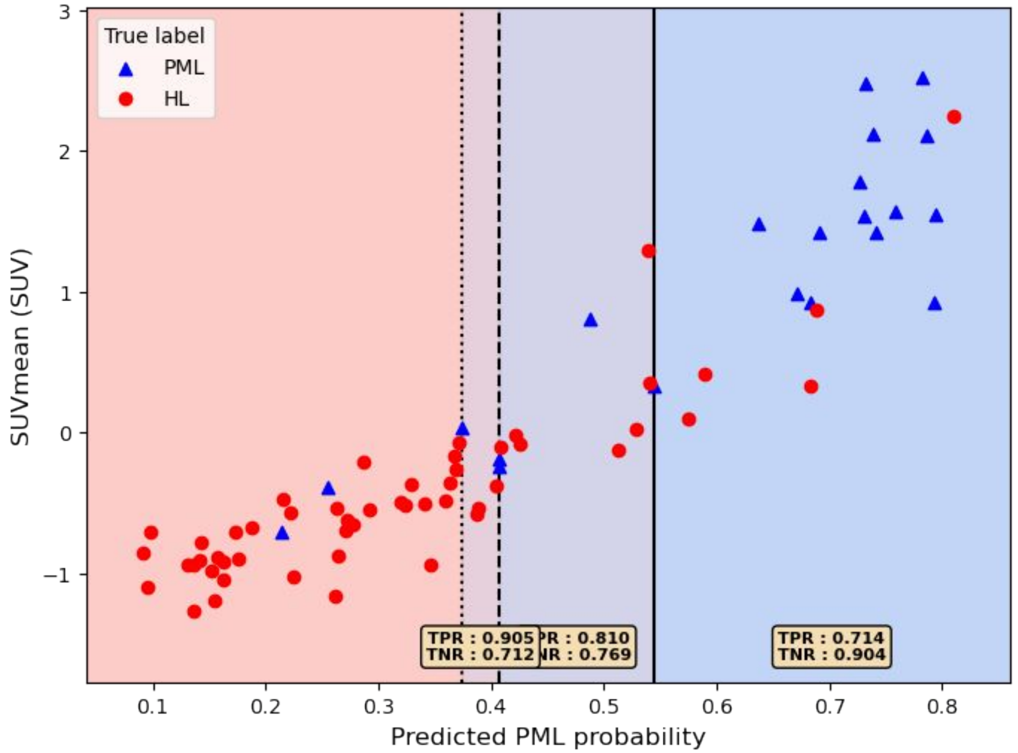
Open to any kind of suggestions!

Backup

Logistic regression: *AUC optimized version*



Random Forest: *AUC optimized version*



Gradient BDT: *AUC optimized version*