HAL
open science

Automatic phenotyping of electronical health record: PheVis algorithm

Thomas Ferté, Sébastien Cossin, Thierry Schaeverbeke, Thomas Barnetche, Vianney Jouhet, Boris Hejblum

To cite this version:

Thomas Ferté, Sébastien Cossin, Thierry Schaeverbeke, Thomas Barnetche, Vianney Jouhet, et al.. Automatic phenotyping of electronical health record: PheVis algorithm. Journal of Biomedical Informatics, 2021, 117, pp.103746. 10.1016/j.jbi.2021.103746 . hal-03100435

HAL Id: hal-03100435

https://inria.hal.science/hal-03100435v1

Submitted on 7 Jan 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Automatic phenotyping of electronical health record: PheVis algorithm

Title: Automatic phenotyping of electronical health record: PheVis algorithm

Corresponding author: Thomas Ferté, Université de Bordeaux, 146 Rue Léo Saignat 33076 Bordeaux, thomas.ferte@u-bordeaux.fr

Authors: Thomas Ferté[1,2], Sébastien Cossin[1,3], Thierry Schaeverbeke[4], Thomas Barnetche[4], Vianney Jouhet[1,3*] and Boris P Hejblum[2*].

1 : Bordeaux Hospital University Center, Pôle de santé publique, Service d'information médicale, Unité Informatique et Archivistique Médicales, F-33000 Bordeaux, France

2: Univ. Bordeaux ISPED, Inserm Bordeaux Population Health Research Center UMR 1219, Inria BSO, team SISTM, F-33000 Bordeaux, France

3: Univ. Bordeaux, Inserm, Bordeaux Population Health Research Center, team ERIAS, UMR 1219, F-33000 Bordeaux, France

4: Rheumatology department, FHU ACRONIM, Bordeaux University Hospital, F-33076 Bordeaux, France

*: These authors contributed equally

CRediT: Thomas Ferté: Methodology, Formal analysis, writing – Original draft, Sébastien Cossin: Supervision, Methodology, Resources, Writing – review, Thierry Schaeverbeke: Conceptualisation, writing – review, Thomas Barnetche: Conceptualisation, writing – review, Vianney Jouhet and Boris P Hejblum: Conceptualization, Methodology, Supervision, writing - review

Keywords (5): electronic health records; high-throughput phenotyping; phenotypic big data; precision medicine

Word count: 3718

Color should be used for figures 2 and 3.

Abstract

Electronic Health Records (EHRs) often lack reliable annotation of patient medical conditions. *Phenorm*, an automated unsupervised algorithm to identify patient medical conditions from EHR data, has been developed. *PheVis* extends *PheNorm* at the visit resolution. *PheVis* combines diagnosis codes together with medical concepts extracted from medical notes, incorporating past history in a machine learning approach to provide an interpretable "white box" predictor of the occurrence probability for a given medical condition at each visit. *PheVis* is applied to two real-world use-cases using the datawarehouse of the University Hospital of Bordeaux: i) rheumatoid arthritis, a chronic condition; ii) tuberculosis, an acute condition (cross-validated AUROC were respectively 0.943 [0.940 ; 0.945] and 0.987 [0.983 ; 0.990]). *PheVis* performs well for chronic conditions, though absence of exclusion of past medical history by natural language processing tools limits its performance in French for acute conditions. It achieves significantly better performance than state-of-the-art methods especially for chronic diseases.

1. Introduction

As the amount of data collected on a daily basis from hospital health care system keeps increasing,[1] the appeal for leveraging the full potential of these data for research purposes and to investigate clinical questions is also becoming stronger than ever.[2–5] Yet, EHR data are quite different from research oriented data (e.g. cohort or trial data): i) they are less structured, more heterogeneous, ii) they present finer granularity, iii) data collection is done for health care purpose.[1,6–8] Currently, one of the main barriers to use such data for studying disease risk factors is the necessity to first identify patients having diseases of interest, a task that we will denote as phenotyping.

Several approaches have been recently proposed to phenotype patients.[9–13] They often rely on either rule-based algorithms specifically designed with clinicians, or on supervised models trained on annotated patient datasets. Such algorithms are limited because their development is disease specific, must be (re-)started from scratch for every new disease and demand a lot of clinician expertise time. In addition, portability and generalization to new databases (e.g. different hospitals) can often fail, requiring once again the process to be reiterated in the new institution. Hripcsak and Albers defined high-throughput phenotyping as an approach that "should generate thousands of phenotypes with minimal human intervention".[8] In this perspective, multiple methods have been developed for automatic phenotyping. Agarwal *et al.* proposed *XPRESS* which learns on noisy labels.[10] Halpern *et al* proposed *Anchor* which learns on so-called "anchor patients", i.e. patients with highly disease-specific features.[11] Wagholikar *et al* developed *Polar,* which learns on so called "polar patients", i.e extreme patients which are almost certain to either have or not have the disease.[12] Finally Yu *et al.* developed *PheNorm* which learns the phenotype as a continuous score.[13] And they also developed *SAFE* which selects relevant features for phenotyping in an automated manner.[14] All these frameworks are unsupervised, in the sense that they require neither manual chart review nor complex rule definitions to classify phenotypes, and thus allow automated high-throughput phenotyping.

While those frameworks are appealing, they only consider phenotyping at the patient level and neglect the timing of illness onset and cure. Yet, we need increased resolution for phenotyping, especially for studying acute diseases (that can occur repeatedly) or for answering epidemiological questions (where temporal sequence is important): phenotyping at the visit level would allow to precisely take into account the dynamic evolution of patient's conditions. Besides, those frameworks were developed using English databases, leveraging advanced NLP tools and relying on rich terminologies not necessarily available in other languages.[15,16] Portability to other languages is not straightforward, as they still often lack resources of matching quality.

We propose a new, portable, approach for unsupervised algorithm extending *PheNorm* at the visit level: *PheVis*. This new *PheVis* method accumulates past information to provide an up-to-date estimation of a phenotype probability at any given visit. This accumulation of previous information from EHR can be tuned to

match the disease length, making *PheVis* a versatile tool suitable for both chronic and acute conditions. Section 2 presents the *PheVis* analysis and modeling strategy. Section 3 demonstrates *PheVis* performance for rheumatoid arthritis (RA) and tuberculosis (TB), a chronic and an acute condition respectively, using French EHRs from Bordeaux University Hospital. The method is compared to other state-of-the-art methods. Finally Section 4 discusses these findings, the limits of the approach, and offers a conclusion.

2. Materials and Methods

PheVis combines ICD10 (international classification of diseases 10th revision) billing codes together with medical concepts extracted from clinical notes, incorporating past information through a user-tunable exponential decay. This creates a silver-standard surrogate of the medical condition of interest. Then variable selection (through elastic-net logistic regression) and pseudo-labelling (using random-forest) are performed, leveraging extreme values of this silver-standard. Finally, a logistic regression model is estimated on those noisy labels to provide an interpretable "white box" predictor of the occurrence probability for a given medical condition at each visit. The different steps of *PheVis* are outlined in Figure 1 and are described below.

1. Input data

The input data of the *PheVis* approach are the clinical notes and the ICD10 codes from an EHR datawarehouse. All the notes and ICD10 codes are collapsed by visit, and IAMsystem, a dictionary-based named entity recognition tool, is used to extract relevant Unified Medical Language System (UMLS) concept unique identifiers (CUIs, i.e. CUIs associated with disease to be phenotyped – see Section 2.3 for details).[17,18] CUIs features are the number of occurrence of the terms in the UMLS dictionary related to the CUIs. No semantic analysis is performed avoiding the distinction between current disease and past disease or negation. ICD10 codes are aggregated at the category level (i.e. the first three characters, M05.1 and M05.2 codes are both counted under the same category code M05). This results in a matrix $X_{k_{ij}p}$ of dimension $\varphi \times P$, where φ is the total number of visits and P the total number of ICD and CUI concepts. Rows of $X_{k_{ij}p}$ are indexed by the vector $k_{ij} \in \{1, \dots, \varphi\}$ with $i \in \{1, \dots, n\}$ the patient index and $j \in \{1, \dots, v_i\}$ the visit index. The v_i notation takes into account the variation of number of visits between patients so that $\varphi = \sum_{i=1}^{n} v_i$. To facilitate reading, k_{ij} will be denoted as k when the patient information is not needed. Columns are indexed by $p \in \{1, \dots, P\}$ the covariate index.

2. Build a surrogate of the disease status

As there are no disease labels for the visits (hence requiring a phenotyping algorithm), a supervised model cannot be trained right away. To be able to train our phenotyping algorithm, we first build a surrogate variable expected to be close to the true disease status. This surrogate is based on the main ICD and UMLS codes that represent a disease.

We define mC_k the standardized sum of main disease concepts as:

$$mC_k = Z(mainICD_k) + Z(mainCUI_k) + min\big(Z(mainICD_k) + Z(mainCUI_k)\big) \text{ and } Z(x) = \frac{x-\mu}{\sigma} \quad (1)$$

$mainICD$ and $mainCUI$ are main concepts related to the disease. For example, for RA we used:

- $mainICD_k = M05_k + M06_k$ with $M05_k$ the number of times the code *M05 Rheumatoid arthritis with rheumatoid factor* was recorded for observation k, and similarly for $M06_k$ and *M06 Other rheumatoid arthritis*
- $mainCUI_k = C0003873_k$ with *C0003873 Rheumatoid arthritis*

At this stage, standardization (centering and scaling) is critical because CUIs occurrences often largely outnumber ICD code occurrences. Without such standardization, the weight of ICD codes in the prediction would be negligible.

To phenotype a given visit, it is necessary to take into account information available from previous visits as well. For example, a patient can be diagnosed with RA at the age of 50 and have a visit at 52 for an infectious event containing no information about RA. RA being a chronic disease, we want to be able to predict RA in both visits. To do so we propose to accumulate past history information with an exponential decay as follow:

$$mCumul_{k_{ij}} = mC_{k_{ij}} + mCumul_{k_{ij-1}} exp\,(-\lambda D_{k_{ij}}) \text{ with } mCumul_{k_{i1}} = mC_{k_{i1}} \text{ and } D_{k_{ij}} = t_{k_{ij}} - t_{k_{ij-1}} \quad (2)$$

λ is a constant parameter tuned by the user that controls the "memory loss" of the algorithm. For easier interpretation one can prefer to set the value of the half-life equal to $ln(2)/\lambda$. The natural half-life chosen is the usual duration of the disease (e.g. 180 days for TB and +∞ for RA — being a chronic disease currently without a cure). Setting the half-life to +∞ for RA is equivalent to simply accumulating the information of all previous visits.

The same exponential decay accumulation is applied to each ICD and UMLS codes. We also define five other features accumulating the information on the last day, last 5 days, last month and last year:

- $lastvis_{k_{ij}} = mC_{k_{ij-1}}$
- $last5vis_{k_{ij}} = \sum_{h=j-5}^{j-1} mC_{k_{ih}}$
- $lastmonth_{k_{ij}} = \sum_{h=1}^{j-1} mC_{k_{ih}} \times \mathbf{1}_m$ with $\mathbf{1}_m = 1$ if $D_{k_{ij}} - D_{k_{ih}} \leq 30days, 0$ otherwise
- $lastyear_{k_{ij}} = \sum_{h=1}^{j-1} mC_{k_{ih}} \times \mathbf{1}_y$ with $\mathbf{1}_y = 1$ if $D_{k_{ij}} - D_{k_{ih}} \leq 365days, 0$ otherwise
- $Cum_{k_{ij}} = \sum_{1}^{j} mC_{k_{ij}}$

This yields an augmented matrix X^a of $\varphi \times (2P+5)$ dimensions: CUIs and ICDs (*P*), their accumulate counts (*P*), and 5 new variables.

3. Variable selection and pseudo-labelling

We use the *SAFE* algorithm to select predictive variables of interest and reduce the dimensionality of the optimization problem. First we use IAM system to extract ICD10 and UMLS concepts in external resources: medical text books and Wikipedia disease specific chapter or page.[19–22] A concept and its accumulate count are kept in the model only if it is found in the two resources. As the true phenotype is not available, we categorize $mCumul_k$ into $S_k = \{0, 0.5, 1\}$ to provide a surrogate to train models. Those three categories distinguish visits for which phenotype is really unlikely (i.e $S_k = 0$), really likely (i.e $S_k = 1$) or uncertain (i.e $S_k = 0.5$) based on $mCumul_k$. To define the two thresholds separating the three categories of S_k, we used $mainICD_k$ which takes into account prevalence variability depending on the disease and the cohort. We first count the proportion of visits with at least one occurrence of *mainICD* code (i.e the *mainICD* ≥ 1 prevalence) that we denote $quant_{mainICD}$ as :

$$quant_{mainICD} = \frac{1}{\varphi}\sum_{k} mainICD_k \geq 1 \quad \text{with } quant_{mainICD} \in [0; 1] \tag{3}$$

We divide this quantity by a constant ω that we set to 5 to define $quant_{extreme}$ as :

$$quant_{extreme} = \frac{quant_{mainICD}}{\omega} \text{ with } \omega \text{ a constant} \tag{4}$$

This $quant_{extreme}$ proportion allows to define three categories: $[0;\ quant_{extreme}]$, $(quant_{extreme};\ 1 - quant_{extreme})$, $[1 - quant_{extreme}; 1]$ and we define the surrogate S_k as:

$$S_k = \begin{cases} 0, \text{ if visit belongs to the } quant_{extreme} \text{ percentile of visits with the lowest } mCumul_k \\ 1, \text{ if visit belongs to the } quant_{extreme} \text{ percentile of visits with the highest } mCumul_k \\ 0.5 \text{ otherwise} \end{cases} \tag{5}$$

For instance, if 20% of visits have at least one occurrence of main ICD codes, then, given $\omega = 5$, S_k is set to 1 for visits belonging to the 4% percentile with the highest $mCumul_k$, S_k is set to 0 for visits belonging to the 4% percentile with the lowest $mCumul_k$ and set to 0.5 otherwise. One can note that the higher the ω constant (i.e the extreme patients are more extreme), the more confident we are in the specificity of S_k in $\{0,1\}$ toward the true phenotype but the smaller training size is for next steps. We found $\omega = 5$ to work well in our setting.

Then we train a logistic regression with elastic-net penalization to select a subset X' of relevant predictors from the X^a matrix using only the visits for which S_k is either 0 or 1. X' is a $\varphi \times P'$ matrix with the subset of predictors with non-zero estimated coefficients. Of note, $mainICD$ and $mainCUI$ are always forced into the set of selected variables in X', while Cum_k is systematically removed for acute conditions.

We then assign a pseudo-label $\{0,1\}$ to all visits. This increases the number of visits available to train the final logistic regression, and also adds visits with more uncertain phenotype status which overall results in smoother

predicted probabilities and better performance. To perform this pseudo-labelling, we train a random-forest with majority vote for trees aggregation for which S_k is either 0 or 1. The trained model is then used to predict the pseudo-label $PL_k \in \{0,1\}$ status for each visit. Disease probability can be estimated at this step. However, random forest variable importance is hard to interpret in a clinical perspective which is particularly annoying for unsupervised learning without gold standard label to evaluate the accuracy of the model. To ease the understanding of the disease representation, probabilities are estimated through a logistic regression.

4. Probability estimation

To estimate the disease occurrence probability, we used a noising-denoising logistic regression with random intercept similarly to *PheNorm*. First, $max\ (10^5, \varphi)$ visits are randomly sampled with replacement with inverse probability weighting defined as $\frac{1}{P(PL_k=1)PL_k+(1-P(PL_k=1))(1-PL_k)}$ in order to balance the training set. This new $max(10^5, \varphi) \times P'$ matrix is denoted X^b. Then we perform a noising-denoising step to force the algorithm to use other variables than the main ICD and UMLS concepts (and thus avoid overfitting with respect to the surrogate construction). Every value of explanatory variables has a probability of $p_{bern} = 0.3$ to be replaced by the mean of the explanatory variable as in *PheNorm*.[13] For instance if M05 ICD10 code mean occurrence is 0.2 then each visit has a probability of 0.3 to have its true M05 value replaced by 0.2. This noisy matrix of dimension $max(10^5, \varphi) \times P'$ is denoted $\mathrm{X}^{\mathrm{n}}_{kp}$:

$$\mathrm{X}^{\mathrm{n}}_{kp} = \begin{cases} mean\left({X^b}_p\right), \text{if } r_{bern_{kp}} = 1 \\ \mathrm{X}^{\mathrm{b}}_{kp}\ , \text{if } r_{bern_{kp}} = 0 \end{cases} \quad \text{with } r_{bern} \sim Bern(p_{bern}) \tag{6}$$

For the denoising step a logistic regression with random intercept is used:

$$logit\left(P\left(PL_{k_{ij}} = 1\right)\right) = {X^n_{k_{ij}p}}^T \beta_p + b_{0i} \text{ with } b_{0i} \sim N(0, \sigma_0^2) \tag{7}$$

And finally the probability of having the disease is estimated on the noise free matrix as:

$$P(Disease_k = 1) = \frac{e^{\left(X^b_{kp}\right)^T \beta_p}}{1 + e^{\left(X^b_{kp}\right)^T \beta_p}} \tag{8}$$

This final probability illustrates the level of confidence of the estimated phenotype based on the used variables.

INPUT

matICD: ICD codes matrix, one column per ICD10 code and one row by visit. ICD10 codes are aggregated at three characters (e.g M05.1 -> M05)

matText: medical text matrix, one column and one row per visit

mainICD: sum of the disease of interest ICD10 codes by visit

mainCUI: sum of the disease of interest CUI codes by visit

matextRessources: the external resources text matrix, one column, each row is a text related to the disease of interest from a different resource

BEGIN

(1) /* DATA STRUCTURATION */

matCUIs := **extract** CUIs from *matText* with IAM system, one column per CUIs code and one row per visit

matStructExt := **extract** CUIs and ICD10 codes from *matextRessources* with IAM system (or other name entity recognition algorithm), one column per CUIs code and one row per resource

X^a := *matCUIs* + *matICD*

(2) /* BUILD SURROGATE */

mC := Standardised **sum** of *mainICD* and *mainCUI*

mCumul := **accumulate** *mC* with exponential decay

S := **categorise** *mCumul* in three categories (0: do not have the disease, 1: has the disease, 0.5: uncertain disease status). Thresholds depend on *mainICD* prevalence. We denote v_{xtr} visits where $S \in \{0,1\}$.

(3) /* VARIABLE SELECTION AND PSEUDO-LABEL */

Xfilt: = **filter** X^a where CUI and ICD is in majority of *matStructExt* rows

ENmodel := **train** (<u>model</u> = elastic-net, <u>predictors</u> = *Xfilt*[v_{xtr},], <u>outcome</u> = *S*[v_{xtr}])

X' := **select** concepts of interest as non-zero beta in *ENmodel*

RFmodel := **train** (<u>model</u> = random-forest, <u>predictors</u> = $X'[v_{xtr},]$, <u>outcome</u> = *S*[v_{xtr}])

PL := **predict** (<u>model</u> = *RFmodel*, <u>new.data</u> = *X'*) for all observation (*S* = {0, 0.5, 1})

(4) /* PROBABILITY ESTIMATION */

X^{boot} := **weighted bootstrap** of *X'*, weight is inverse probability from *PL*

X^{noise} := **replace** 30% of matrix cells by the corresponding mean of the column variable (noising step)

LRmodel := **train** (<u>model</u> = logistic regression with random intercept, <u>predictors</u> = X^{noise}, <u>outcome</u> = *PL*)

Return FinalProba := **predict** (<u>model</u> = *LRmodel* with fixed coefficient only, <u>new.data</u> = *X'*)

END

Figure 1: Pseudo-code of *PheVis*.

3. Results

3.1. Application design

We illustrate the *PheVis* method on RA, a chronic disease which cannot be cured, and active TB, an acute disease which usually last between 6 to 12 months.[19–22] The model performance was evaluated on an imperfect gold standard for both diseases: for RA we used the presence of at least one rheumatoid arthritis form, a form specifically used by rheumatologists at the University Hospital of Bordeaux in usual RA care, for tuberculosis we manually reviewed patients with at least one mention of tuberculosis treatment while other patients were considered not having the disease. Latent tuberculosis was labelled as tuberculosis negative because, even if the bacterium is the same, symptoms, diagnosis and treatment are different. Patients were included in the study cohort if i) they had been hospitalized at the University Hospital of Bordeaux at least once since 2010 and ii) if they had either one primary or secondary ICD code of RA (M05 or M06), or one biological measurement of Anti-Citrullinated Peptide Antibody. The cohort was split into training and test datasets at the patient level with a 70% to 30% ratio. The cohort is described in Table 1, highlighting the discrepancy between ICD, CUI and gold-standard justifying the need for automated phenotyping algorithms.

Table 1 Description of phenotyping cohort. *University Hospital of Bordeaux*.

	Train set		**Test set**	
	Patients	**Visits**	**Patients**	**Visits**
n	9,102	237,875	2,359	62,004
Gold standard RA (%)	953 (10.5)	27,077 (11.4)	274 (11.6)	7,883 (12.7)
ICD RA[1] ≥ 1 (%)	3,682 (40.5)	21,448 (9.0)	901 (38.2)	5,823 (9.4)
CUI RA ≥ 1 (%)	3,703 (40.7)	32,775 (13.8)	952 (40.4)	8,632 (13.9)
Gold standard TB (%)	49 (0.5)	618 (0.3)	5 (0.2)	90 (0.1)
ICD TB[2] ≥ 1 (%)	88 (1.0)	277 (0.1)	15 (0.6)	50 (0.1)
CUI TB ≥ 1 (%)	647 (7.1)	2,393 (1.0)	147 (6.2)	439 (0.7)

1: ICD RA: M05, M06
2: ICD TB: A15, A16, A17, A18, A19

Ten different prediction models were evaluated for each disease: (i) our proposed *PheVis* approach, for which we set $\lambda_{RA} = \frac{ln\,(2)}{Inf} = 0$ and $\lambda_{TB} = \frac{ln\,(2)}{180}$ (tuberculosis typically lasting around 6 months), (ii) *XPRESS* for which we defined the silver standard as the main ICD code presence, (iii) *ANCHOR* for which we defined the anchor visits as the one with at least one main ICD and main CUI code, (iv) the elastic-net and (v) the random forest *POLAR* algorithms for which we adapted the polar patient definition to our setting (negative polar visits neither main ICD codes and neither main CUI codes, whereas the positive polar visits had both), (vi) *PheNorm*, (vii) a supervised elastic-net with accumulate variables (denoted as *ENET_CUM*) and also viii) without accumulate

variables (*ENET_NOCUM*), (ix) a supervised random forest with accumulate variables (*RF_CUM*) and (x) without cumulated variables (*RF_NOCUM*). Of note, supervised models (*ENET_NOCUM*, *RF_NOCUM*, *ENET_CUM* and *RF_CUM*) used the gold-standard labels for training. We used *SAFE* to select variables used for prediction by each algorithm in order to ease their comparison.

3.2. Application results

Figure 2 shows individual *PheVis* and other state-of-the-art methods predictions for four patients. For RA, as the information is accumulated over time in *PheVis*, the model is able to maintain relatively stable predictions over time even if there is not much information about RA in a visit. Without this feature, other approaches display highly variables predictions oscillating over time. For TB, the advantage of accumulating information confronts the problem of accumulating too much. This problem is increased in French because the majority of natural language processing tools and terminological systems were developed for the English language. Tools to detect negation and past history are not yet implemented in Bordeaux University Hospital datawarehouse.[23] Prediction of *PheNorm* is really close to 0.5 for all visits. This can be explained by the final step of the *PheNorm* algorithm which is a mixture model on the predicted sum of main ICD and main CUI. As our dataset is largely imbalanced, with many more negative patients than positive ones, both normal distributions of the mixture model are concentrated on the negative class. Because they are close to each other, the probability of belonging to each class (positive or negative phenotype) is really close. However, as shown on figure 3, those probabilities are not constant and still give good prediction performance according to AUROC and AUPRC, in spite of being poorly interpretable. Other methods do not have this problem, as they mainly learn on binary silver standard.

Figure 3 shows the performance of *PheVis* and the other methods on the test set for both TB and RA. For RA, *PheVis* significantly outperforms any other unsupervised method, both in term of AUROC (RA: 0.943 [0.940 ; 0.945], TB: 0.987 [0.983 ; 0.990]) and AUPRC (RA: 0.754 [0.744 ; 0.763], TB: 0.299 [0.198 ; 0.403]). Table 3 details the performances of the algorithm in the train and test sets compared to other methods and with different hyperparameters (half-life, ω and pseudo-labels). More elements justifying pseudo-labels are available in the supplementary material. As expected, for TB the advantage of *PheVis* is less important because the information is accumulated over a shorter time period, still it performs well especially in term of AUROC compared to other unsupervised methods. Supervised methods worked significantly better with accumulated variables supporting the importance of taking into account past history to predict actual phenotype. Among the other unsupervised methods, we can denote that *PheNorm* and the random forest *POLAR* seem to be the best methods, however, as shown on figure 2 *PheNorm* has serious calibration problem on unbalanced datasets. *XPRESS* failed to reach convergence in our setting possibly because it uses lasso penalization instead of elastic-net as in *SAFE* or *PheVis*, or ridge as in *ANCHOR* or *POLAR*. Both for *POLAR* and the supervised methods, random forest was able to significantly improve the performance of the model, probably because it is able to learn more complex structure than penalized linear regression.

Table 3 shows point performance for two arbitrary phenotyping decision rules: i) a predicted probability above 0.5 ii) a probability above the threshold maximizing the sum of the precision and the recall. Specificity and negative predictive values are good partly because the diseases are rare at the visit level. Matching the results from figure 3, the sensitivity/positive predictive values trade-off is better for RA than for TB.

We investigated the lack of performance for TB phenotyping by comparing the number of visits with at least one occurrence of main TB CUI among patients with at least one visit labelled TB positive by chart-review. Visits labelled TB negative by chart-review with TB positive past history were significantly more likely to have at least one occurrence of main TB CUI compared to visits labelled TB negative without TB past history (31.0% vs 4.4%, chi-2 p-value = 2.13×10^{-62}). Also there was no signifant difference compared to visits with current TB (31.0% vs 29.1%, chi-2 p-value = 0.43).

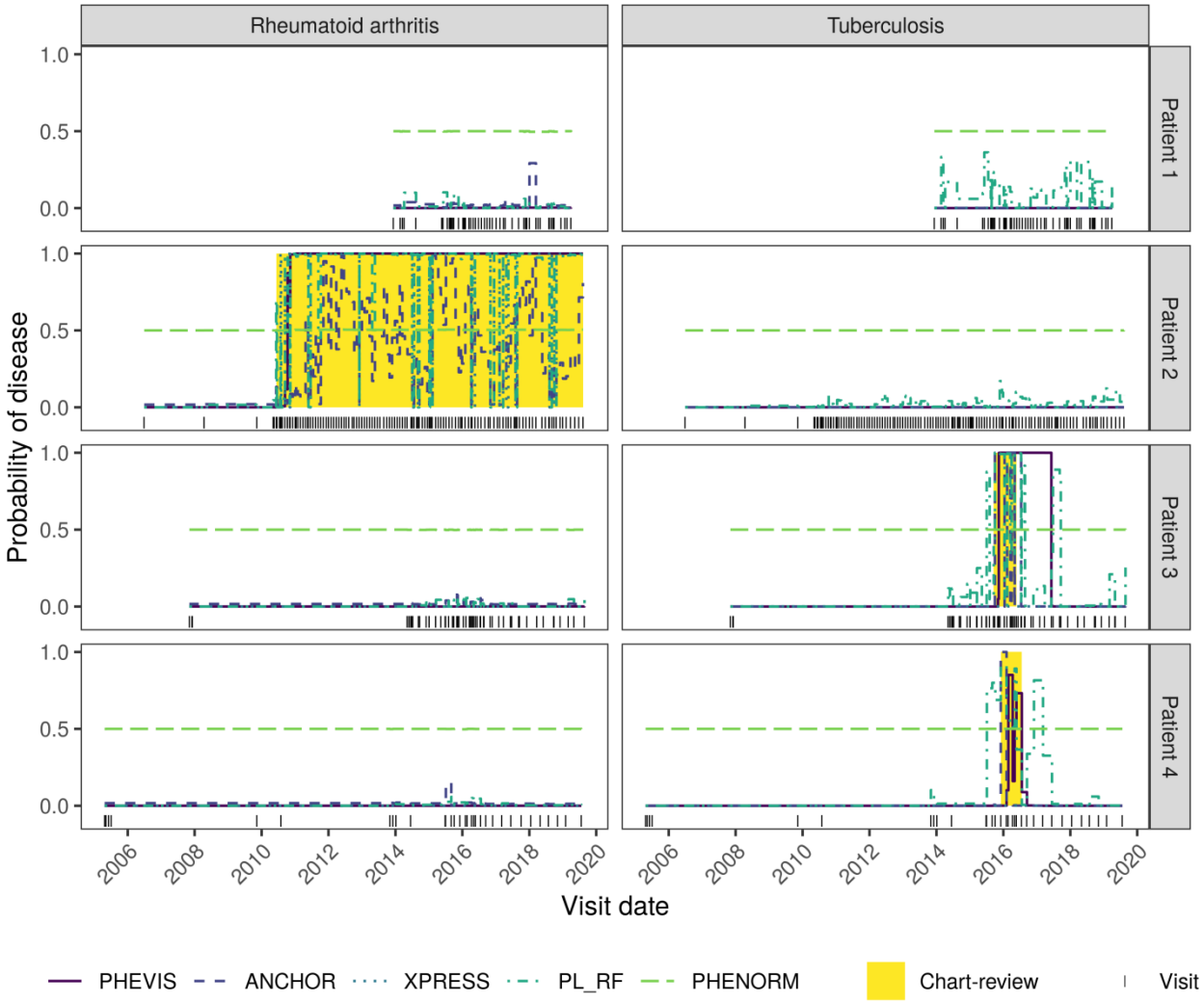


Figure 2 : Individual prediction of rheumatoid arthritis (RA) and tuberculosis (TB). Each column corresponds to a disease, each row to a patient. Yellow areas corresponds to visits having the disease, white areas corresponds to visits without disease. Each vertical bar corresponds to a visit. Patient 1 has no disease. Patient 2 has RA which is well estimated by *PheVis*, other algorithms have high variability in their prediction. Patient 3 and 4 have tuberculosis. For patient 3, that information is still trailing behind after the patient is cured for *PheVis,* partly because of the lack of advanced natural language processing tool hindering the distinction between past and actual disease history. *PheNorm* predictions are close to 0.5 because the mixture model fails to learn meaningful probabilities in this extremely imbalanced setting. *XPRESS* fails to estimate any probability higher than 0 for TB because of convergence failure and RA probabilities are either almost 0 or 1 providing a binary interpretation of the disease status rather than a continuous one. Both *PL_RF* and *Anchor* are too volatile to provide a trusted interpretation of their output probability.

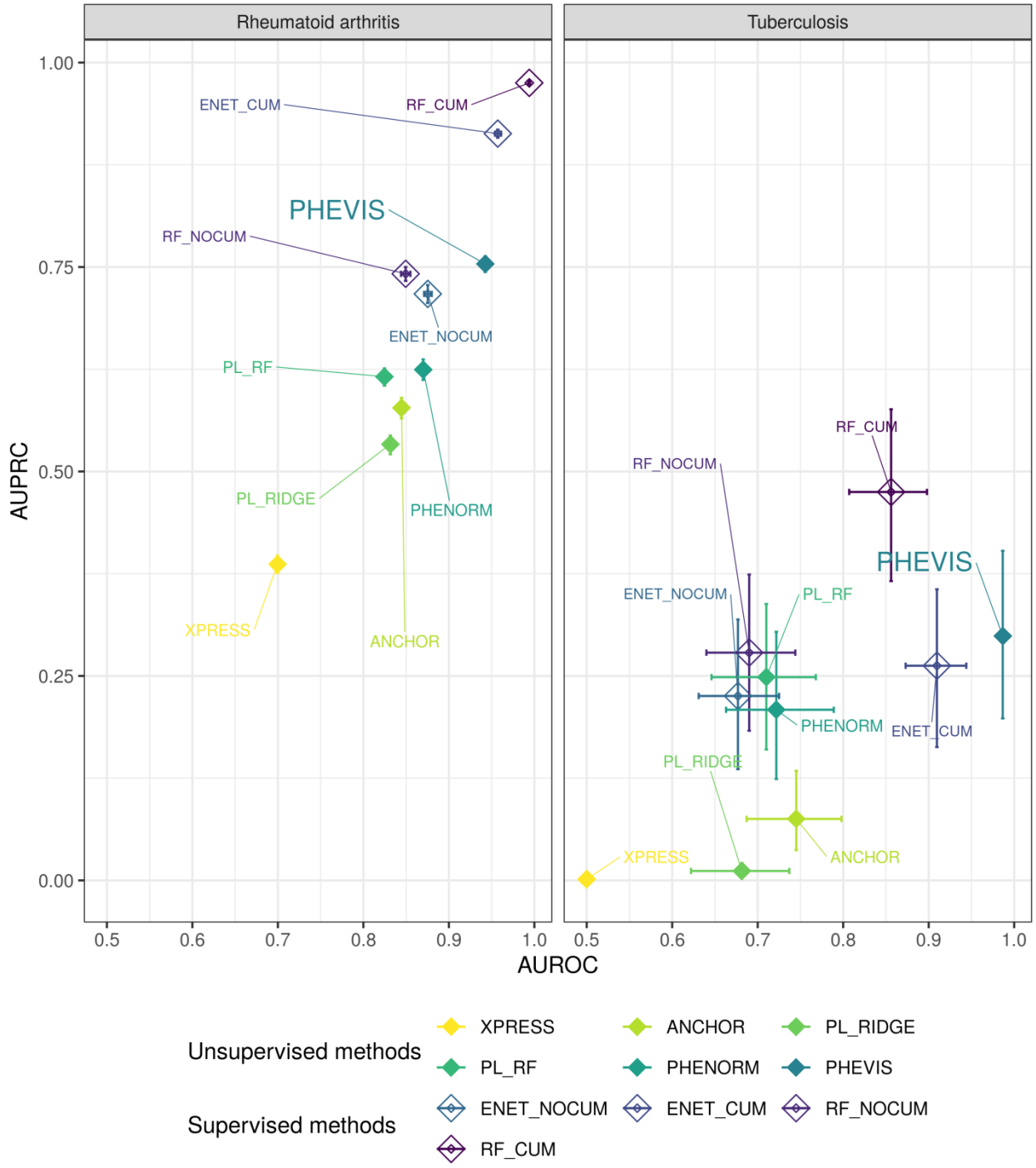


Figure 3: Phenotyping unsupervised and supervised methods performances comparison on the test set. University Hospital of Bordeaux. Confidence Intervals for AUROC and AUPRC are represented by horizontal and vertical segments respectively. PL_RIDGE: *POLAR* method with ridge logistic regression. PL_RF: *POLAR* method with random forest. ENET: Supervised elastic-net logistic regression with (CUM) or without (NOCUM) accumulate variables. RF: Supervised random forest with or without accumulate variables.

Table 2: Phenotyping unsupervised and supervised methods comparison. University Hospital of Bordeaux.

	Test set		Train set	
Algorithm	**AUROC**	**AUPRC**	**AUROC**	**AUPRC**
Rheumatoid Arthritis				
ANCHOR	0.845 [0.839 ; 0.850]	0.578 [0.565 ; 0.590]	0.827 [0.824 ; 0.830]	0.481 [0.476 ; 0.487]
ENET_CUM	**0.957 [0.954 ; 0.961]**	**0.913 [0.908 ; 0.918]**	**0.986 [0.985 ; 0.987]**	**0.953 [0.951 ; 0.955]**
ENET_NOCUM	0.875 [0.871 ; 0.880]	0.717 [0.706 ; 0.728]	0.865 [0.862 ; 0.867]	0.687 [0.682 ; 0.692]
PHENORM	0.870 [0.865 ; 0.875]	0.625 [0.612 ; 0.637]	0.855 [0.852 ; 0.857]	0.577 [0.571 ; 0.584]
PHEVIS F/5/D	0.942 [0.939 ; 0.944]	**0.759 [0.750 ; 0.768]**	0.659 [0.656 ; 0.662]	0.424 [0.418 ; 0.429]
PHEVIS T/1/D	0.929 [0.926 ; 0.932]	0.708 [0.698 ; 0.718]	0.933 [0.932 ; 0.934]	0.676 [0.671 ; 0.681]
PHEVIS T/10/D	**0.951 [0.948 ; 0.953]**	**0.772 [0.762 ; 0.782]**	**0.951 [0.950 ; 0.952]**	**0.745 [0.739 ; 0.750]**
PHEVIS T/2/D	**0.946 [0.944 ; 0.948]**	0.736 [0.727 ; 0.746]	**0.944 [0.943 ; 0.945]**	0.689 [0.684 ; 0.694]
PHEVIS T/20/D	**0.957 [0.955 ; 0.959]**	**0.798 [0.789 ; 0.806]**	**0.951 [0.950 ; 0.952]**	**0.765 [0.760 ; 0.770]**
PHEVIS T/5/0	0.900 [0.896 ; 0.904]	0.571 [0.559 ; 0.582]	0.898 [0.896 ; 0.900]	0.524 [0.518 ; 0.529]
PHEVIS T/5/D	**0.943 [0.940 ; 0.945]**	**0.754 [0.744 ; 0.763]**	**0.943 [0.942 ; 0.944]**	**0.717 [0.712 ; 0.722]**
PL_RF	0.825 [0.819 ; 0.831]	0.616 [0.605 ; 0.626]	0.813 [0.810 ; 0.816]	0.546 [0.540 ; 0.552]
PL_RIDGE	0.832 [0.826 ; 0.837]	0.533 [0.521 ; 0.544]	0.821 [0.818 ; 0.824]	0.496 [0.490 ; 0.502]
RF_CUM	**0.994 [0.993 ; 0.995]**	**0.975 [0.973 ; 0.977]**	**0.999 [0.999 ; 1.000]**	**0.999 [0.998 ; 0.999]**
RF_NOCUM	0.849 [0.844 ; 0.855]	0.742 [0.733 ; 0.750]	0.845 [0.843 ; 0.848]	**0.729 [0.724 ; 0.734]**
XPRESS	0.700 [0.695 ; 0.705]	0.387 [0.376 ; 0.397]	0.693 [0.690 ; 0.695]	0.345 [0.339 ; 0.350]
Tuberculosis				
ANCHOR	0.745 [0.687 ; 0.798]	0.075 [0.037 ; 0.134]	0.635 [0.608 ; 0.660]	0.128 [0.100 ; 0.155]
ENET_CUM	0.910 [0.873 ; 0.944]	0.262 [0.163 ; 0.356]	**0.928 [0.914 ; 0.941]**	0.451 [0.405 ; 0.488]
ENET_NOCUM	0.677 [0.631 ; 0.725]	0.225 [0.136 ; 0.319]	0.667 [0.650 ; 0.685]	0.188 [0.152 ; 0.222]
PHENORM	0.722 [0.663 ; 0.789]	0.209 [0.124 ; 0.304]	0.682 [0.658 ; 0.707]	0.097 [0.075 ; 0.122]
PHEVIS F/5/D	0.987 [0.983 ; 0.991]	**0.309 [0.204 ; 0.411]**	0.729 [0.711 ; 0.747]	0.157 [0.131 ; 0.185]
PHEVIS T/1/D	0.986 [0.982 ; 0.990]	0.248 [0.167 ; 0.334]	**0.951 [0.940 ; 0.960]**	0.191 [0.164 ; 0.220]
PHEVIS T/10/D	0.987 [0.984 ; 0.991]	0.282 [0.185 ; 0.382]	0.844 [0.823 ; 0.862]	0.153 [0.128 ; 0.176]
PHEVIS T/2/D	0.987 [0.983 ; 0.991]	0.249 [0.168 ; 0.335]	0.847 [0.827 ; 0.865]	0.177 [0.149 ; 0.205]
PHEVIS T/20/D	0.697 [0.651 ; 0.751]	0.158 [0.084 ; 0.254]	0.646 [0.628 ; 0.662]	0.069 [0.052 ; 0.089]
PHEVIS T/5/0	0.757 [0.704 ; 0.806]	0.139 [0.080 ; 0.218]	0.506 [0.485 ; 0.526]	0.003 [0.002 ; 0.003]
PHEVIS T/5/D	**0.987 [0.983 ; 0.990]**	**0.299 [0.198 ; 0.403]**	**0.853 [0.834 ; 0.870]**	**0.191 [0.164 ; 0.216]**
PL_RF	0.710 [0.646 ; 0.768]	0.249 [0.160 ; 0.338]	0.634 [0.609 ; 0.661]	0.149 [0.120 ; 0.179]
PL_RIDGE	0.681 [0.622 ; 0.737]	0.011 [0.006 ; 0.021]	0.598 [0.573 ; 0.627]	0.013 [0.010 ; 0.018]
RF_CUM	0.856 [0.807 ; 0.898]	**0.475 [0.366 ; 0.576]**	**0.985 [0.977 ; 0.991]**	**0.942 [0.923 ; 0.958]**
RF_NOCUM	0.690 [0.640 ; 0.744]	0.278 [0.183 ; 0.374]	0.672 [0.656 ; 0.690]	**0.232 [0.198 ; 0.269]**
XPRESS	0.500 [0.500 ; 0.500]	0.001 [0.001 ; 0.002]	0.500 [0.500 ; 0.500]	0.003 [0.002 ; 0.003]

Bold: median AUROC or AUPRC superior to PHEVIS T/5/D

PHEVIS Pseudo-labels/ ω /Half-life. Pseudo-labels is True (T) or False (F). ω is the constant defining $quant_{extreme}$. Half-life is 0 or the disease duration (D), 180 for TB and +∞ for RA

PL_RIDGE: POLAR method with ridge logistic regression. PL_RF: POLAR method with random forest.

ENET: Supervised elastic-net logistic regression with (CUM) or without (NOCUM) cumulated accumulate variables.

RF: Supervised random forest with or without cumulated accumulate variables.

Table 3 *PheVis* performance on the test set for different thresholds of the output probabilities.

Disease	Threshold	SE	SP	PPV	NPV
Rheumatoid arthritis	0.5	0.740	0.942	0.651	0.961
	Optimal P-R* (0.322)	0.761	0.936	0.633	0.964
Tuberculosis	0.5	0.300	1.000	0.519	0.999
	Optimal P-R* ($1.30\ 10^{-9}$)	0.989	0.945	0.025	1.000

* Threshold maximizing the precision recall sum.
Se: sensitivity – Sp: specificity – PPV: positive predictive value – NPV: negative predictive value

4. Conclusions

We developed *PheVis* as an unsupervised automatic phenotyping algorithm at the visit level. Our innovative approach resembles the human medical probabilistic approach of diagnosis as the output is a probability taking into account the uncertainty of the information inside EHR.[24] It is able to achieve interesting performances for RA, which is promising for other chronic conditions. While *PheVis* represent a significant improvement over the current state-of-the-art thanks to its versatile and tunable information accumulating feature, it also suffers from limitations when it comes to acute conditions such as TB. The algorithm is fully automated, not requiring any (time-consuming and expensive) manual chart review, and can in theory be used for different kinds of medical conditions (either acute or chronic). However, the optimal values of hyperparameters might vary depending on the disease of interest and the EHR. In our setting, even if $\omega = 5$ worked well for both diseases, setting it to 10 or 20 would have increase RA phenotyping performances.

PheVis adds many innovations to the previous *PheNorm* algorithm it builds upon: the needs for standardizing the information from medical notes and ICD codes, the accumulation of past history with exponential decay, the definition of silver standard using ICD codes to take into account prevalence of the disease, and pseudo-labelling to improve performance and increase stability of predicted probabilities. Also we demonstrated the portability (and limitations) of those methods in French and in a different datawarehouse than the one used to develop *PheNorm*, with consistent performances for phenotyping RA compared to Yu et al.[13]

Our application setting is different from the other methods original paper, mainly because there is intra-patient correlation between visits phenotype. This is not accounted for in *Phenorm*, *XPRESS* or *POLAR* where the learning is at patient level, nor in *ANCHOR* because it learns on acute diseases.[10–13] As in *POLAR* or *ANCHOR*, our dataset is largely unbalanced towards the negative class (e.g 1.1% epilepsy prevalence in *POLAR*, 2.0% septic shock prevalence in *ANCHOR*) contrary to *PheNorm* (lowest disease prevalence was RA with 22.5%). This unbalanced setting seems to favor unsupervised learning with silver standard (*PheVis*, *POLAR*, *ANCHOR*, *XPRESS*) in terms of calibration. In terms of performance, *PheVis* performed better for both diseases

on both AUROC and AUPRC. Random forest *POLAR* and *PheNorm* have close performances. As those three methods rely on different approaches, future developments might be able to leverage and combine each of their strengths.

These phenotyping algorithms are highly sensitive to the input features, which emphasizes the need for finer natural language processing tools able to perform semantic analysis. The use of other features such as biological test results or treatment should also be considered, as they should be highly predictive of the phenotype, but further works is needed to define how they could be integrated into the silver-standard surrogate strategy used in *PheVis.* Also, instead of providing raw variables as input of the algorithm, *PheVis* could benefit from embedding or tensor factorization approaches that might provide more informative variables but would also increase the complexity for non-expert users.[25–27]

Our performance evaluation is made against an imperfect gold standard, mainly due to the lack of large annotated patient reference sets. For TB, the gold standard was manually curated, while for RA, we used a highly specific form but which might lack sensitivity: interestingly, upon manual inspection it appeared that *PheVis* was able to accurately recover RA patients visits of 5 patients who were not treated in the Rheumatology department of the University Hospital of Bordeaux and thus had no record of this specific form, resulting in a failure of the gold standard. Such phenomena might underestimate the algorithm performance.

PheVis can provide a probability for a large set of diseases and medical conditions with little effort. The performances might vary depending on the disease of interest, the database quality and the EHR language but were better than state-of-the-art method in our study. The use of those estimated probabilities opens new horizon for the use of EHR for medical and epidemiological research purposes.

Supplementary material

ICD10 codes

Table S1: Main ICD codes of rheumatoid arthritis and tuberculosis used by PheVis.

Tuberculosis	A15, A16, A17, A18, A19
Rheumatoid Arthritis	M05, M06

Algorithm performance

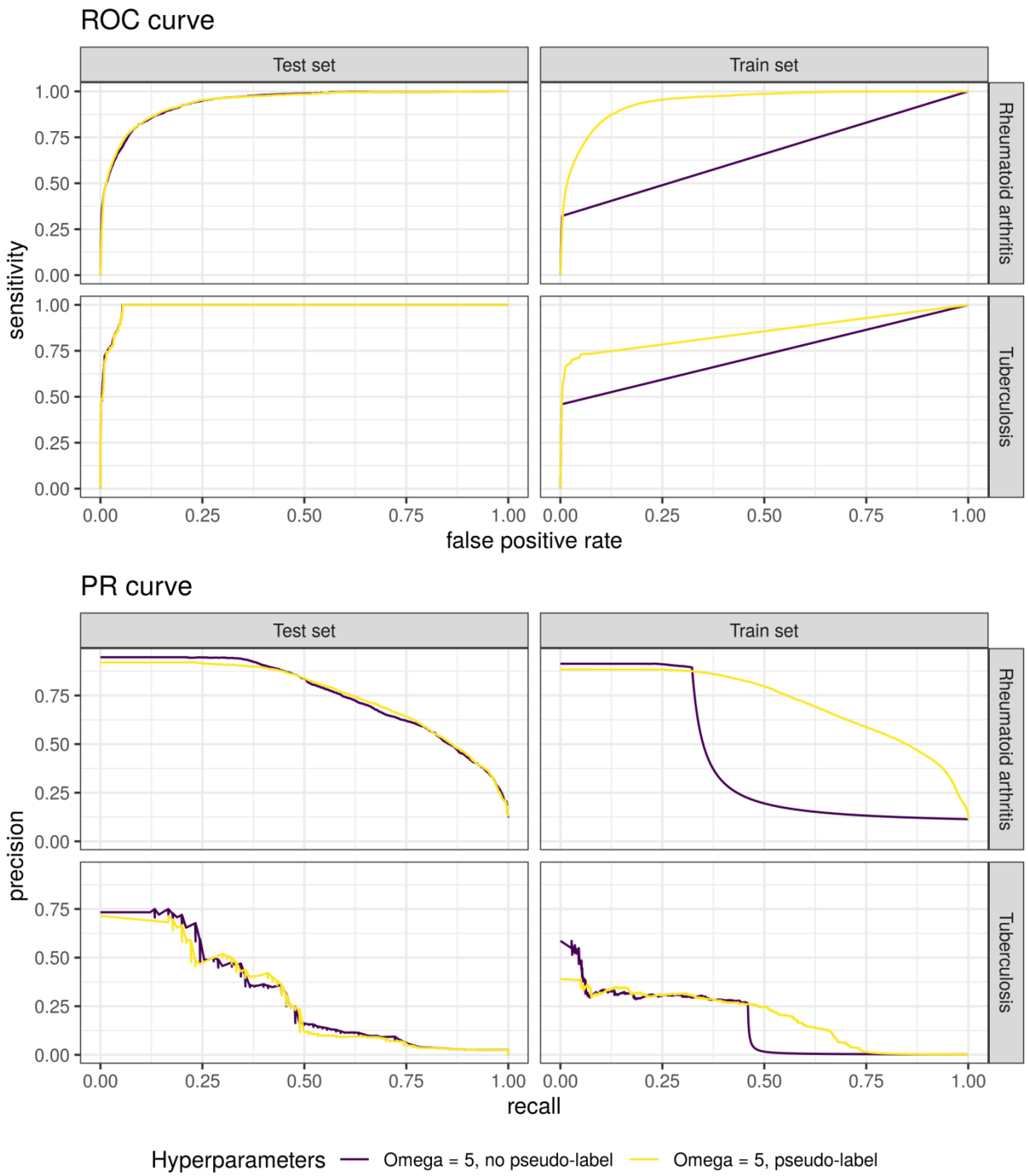


Figure S1: ROC and PR curve of *PheVis* predicted probabilities with and without pseudo-labels. University Hospital of Bordeaux.

As shown in figure S1, pseudo-labels have little effect on the performances on the test set. However, on the train set, it provides more intermediate probabilities leading to better AUROC and AUPRC. As *PheVis* is

unsupervised, users might want to use the predicted probabilities on the training set. For this reason, pseudo-labels are kept in the algorithm.

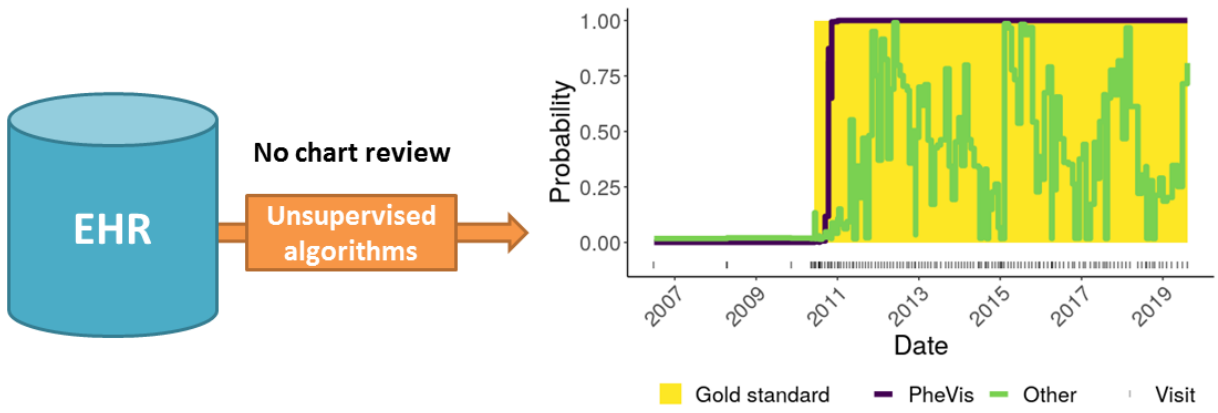
Competing Interests

None.

Highlights

- Electronic Health Record phenotyping is challenging especially at the visit level.
- *PheVis* is a new unsupervised approach extending *PheNorm* to visit level.
- Incorporating accumulated features to take into account disease dynamic increase model performances.
- *PheVis* outperforms other phenotyping algorithms at the visit level.

Graphical abstract



Bibliography

1 Kim E, Rubinstein SM, Nead KT, *et al.* The Evolving Use of Electronic Health Records (EHR) for Research. *Semin Radiat Oncol* 2019;**29**:354–61. doi:10.1016/j.semradonc.2019.05.010

2 Beesley LJ, Salvatore M, Fritsche LG, *et al.* The emerging landscape of health research based on biobanks linked to electronic health records: Existing resources, statistical challenges, and potential opportunities. *Stat Med* 2019;:1–28. doi:10.1002/sim.8445

3 Coorevits P, Sundgren M, Klein GO, *et al.* Electronic health records: new opportunities for clinical research. *J Intern Med* 2013;**274**:547–60. doi:10.1111/joim.12119

4 Danciu I, Cowan JD, Basford M, *et al.* Secondary use of clinical data: The Vanderbilt approach. *J Biomed Inform* 2014;**52**:28–35. doi:10.1016/j.jbi.2014.02.003

5 Sandhu E, Weinstein S, McKethan A, *et al.* Secondary Uses of Electronic Health Record Data: Benefits and Barriers. *Jt Comm J Qual Patient Saf* 2012;**38**:34–40. doi:10.1016/S1553-7250(12)38005-7

6 Wilcox AB. Leveraging Electronic Health Records for Phenotyping. In: Payne PRO, Embi PJ, eds. *Translational Informatics: Realizing the Promise of Knowledge-Driven Healthcare*. London: : Springer 2015. 61–74. doi:10.1007/978-1-4471-4646-9_4

7 Weiskopf NG, Weng C. Methods and dimensions of electronic health record data quality assessment: enabling reuse for clinical research. *J Am Med Inform Assoc JAMIA* 2013;**20**:144–51. doi:10.1136/amiajnl-2011-000681

8 Hripcsak G, Albers DJ. Next-generation phenotyping of electronic health records. *J Am Med Inform Assoc JAMIA* 2013;**20**:117–21. doi:10.1136/amiajnl-2012-001145

9 Banda JM, Seneviratne M, Hernandez-Boussard T, *et al.* Advances in Electronic Phenotyping: From Rule-Based Definitions to Machine Learning Models. *Annu Rev Biomed Data Sci* 2018;**1**:53–68. doi:10.1146/annurev-biodatasci-080917-013315

10 Agarwal V, Podchiyska T, Banda JM, *et al.* Learning statistical models of phenotypes using noisy labeled training data. *J Am Med Inform Assoc JAMIA* 2016;**23**:1166–73. doi:10.1093/jamia/ocw028

11 Halpern Y, Horng S, Choi Y, *et al.* Electronic medical record phenotyping using the anchor and learn framework. *J Am Med Inform Assoc JAMIA* 2016;**23**:731–40. doi:10.1093/jamia/ocw011

12 Wagholikar KB, Estiri H, Murphy M, *et al.* Polar labeling: silver standard algorithm for training disease classifiers. *Bioinforma Oxf Engl* 2020;**36**:3200–6. doi:10.1093/bioinformatics/btaa088

13 Yu S, Ma Y, Gronsbell J, *et al.* Enabling phenotypic big data with PheNorm. *J Am Med Inform Assoc JAMIA* 2017;**25**:54–60. doi:10.1093/jamia/ocx111

14 Yu S, Chakrabortty A, Liao KP, *et al.* Surrogate-assisted feature extraction for high-throughput phenotyping. *J Am Med Inform Assoc JAMIA* 2017;**24**:e143–9. doi:10.1093/jamia/ocw135

15 Yu S, Cai T, Cai T. NILE: Fast Natural Language Processing for Electronic Health Records. *ArXiv13116063 Cs* Published Online First: 23 November 2013.http://arxiv.org/abs/1311.6063 (accessed 5 Sep 2019).

16 Bodenreider O, McCray AT. From French vocabulary to the Unified Medical Language System: A preliminary study. *Stud Health Technol Inform* 1998;**52**:670–4.

17 Cossin S, Jouhet V, Mougin F, *et al.* IAM at CLEF eHealth 2018 : Concept Annotation and Coding in French Death Certificates. ;:7.

18 Bodenreider O. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Res* 2004;**32**:D267–70. doi:10.1093/nar/gkh061

19 Polyarthrite rhumatoïde. http://www.lecofer.org/item-objectifs-0-19.php (accessed 12 Dec 2019).

20 Polyarthrite rhumatoïde. Wikipédia. 2019.https://fr.wikipedia.org/w/index.php?title=Polyarthrite_rhumato%C3%AFde&oldid=164700221 (accessed 12 Dec 2019).

21 Tuberculose. Wikipédia. 2019.https://fr.wikipedia.org/w/index.php?title=Tuberculose&oldid=165260234 (accessed 12 Dec 2019).

22 Référentiel National de Pneumologie – CEP. http://cep.splf.fr/enseignement-du-deuxieme-cycle-dcem/referentiel-national-de-pneumologie/ (accessed 12 Dec 2019).

23 Garcelon N, Neuraz A, Benoit V, *et al.* Improving a full-text search engine: the importance of negation detection and family history context to identify cases in a biomedical data warehouse. *J Am Med Inform Assoc* 2016;:ocw144. doi:10.1093/jamia/ocw144

24 Owens DK, Sox HC. Medical Decision-Making: Probabilistic Medical Reasoning. In: Shortliffe EH, Perreault LE, eds. *Medical Informatics*. New York, NY: : Springer New York 2001. 76–131. doi:10.1007/978-0-387-21721-5_3

25 Henderson J, He H, Malin BA, *et al.* Phenotyping through Semi-Supervised Tensor Factorization (PSST). *AMIA Annu Symp Proc AMIA Symp* 2018;**2018**:564–73.

26 Glicksberg BS, Miotto R, Johnson KW, *et al.* Automated disease cohort selection using word embeddings from Electronic Health Records. *Pac Symp Biocomput Pac Symp Biocomput* 2018;**23**:145–56.

27 Ho JC, Ghosh J, Steinhubl SR, *et al.* Limestone: High-throughput candidate phenotype generation via tensor factorization. *J Biomed Inform* 2014;**52**:199–211. doi:10.1016/j.jbi.2014.07.001