Human and machine learning of prognostic prediction for prelabor rupture of membranes and the time of delivery: a nationwide development, validation, and deployment using medical history

Herdiantri Sufriyana, MD, MSc; [a, b] Yu-Wei Wu, PhD; [a, c] Emily Chia-Yu Su, PhD [a, c, d, *]

[a] Graduate Institute of Biomedical Informatics, College of Medical Science and Technology, Taipei Medical University, 250 Wu-Xing Street, Taipei 11031, Taiwan.
[b] Department of Medical Physiology, Faculty of Medicine, Universitas Nahdlatul Ulama Surabaya, 57 Raya Jemursari Street, Surabaya 60237, Indonesia.
[c] Clinical Big Data Research Center, Taipei Medical University Hospital, 250 Wu-Xing Street, Taipei 11031, Taiwan.
[d] Research Center for Artificial Intelligence in Medicine, Taipei Medical University, 250 Wu-Xing Street, Taipei 11031, Taiwan.
* Corresponding author at: Clinical Big Data Research Centre, Taipei Medical University Hospital, 250 Wu-Xing Street, Taipei 11031, Taiwan. Phone: +886-2-66382736 ext. 1515.

Abstract

Prognostic prediction of prelabor rupture of membrane (PROM) lacks of sample size and external validation. We compared a statistical model, machine learning algorithms, and a deep-insight visible neural network (DI-VNN) for PROM and estimating the time of delivery. We selected visits, including PROM (*n*=23,791/170,730), retrospectively from a nationwide health insurance dataset. DI-VNN achieved the best prediction (area under receiver operating characteristics curve [AUROC] 0.73, 95% CI 0.72 to 0.75). Meanwhile, random forest using principal components achieved the best estimation with root mean squared errors ± 2.2 and 2.6 weeks respectively for the predicted event and nonevent. DI-VNN outperformed previous models by an external validation set, including one using a biomarker (AUROC 0.641; *n*=1,177). We deployed our models as a web application requiring diagnosis/procedure codes and dates. In conclusion, our models may be used solely in low-resource settings or as a preliminary model to reduce a specific test requiring high-resource setting.

Keywords: prelabor rupture of membranes, preterm delivery, medical history, causal diagram, deep learning

Preterm prelabor rupture of membranes (PROM) are widely used as an inclusion criterion for diagnostic and prognostic predictions of other conditions.[1-8] The disease precedes 40–50% of all preterm delivery and arises from multiple disease pathways.[9] Yet, the antecedent remains unclear and the prognostic prediction lacks of sample size and external validation.

Preterm delivery occurs in ~10% births in United States, of which 2–3% contributed by preterm PROM, while term PROM occurs in ~8% pregnancies.[10] The premature baby requires a neonatal intensive care unit,[11,12] that is still scarce in several countries worldwide.[13-17] Meanwhile, the utility spends a majority of healthcare costs in pediatrics and the single largest healthcare spending.[18] Predicting this disease, estimating the time of delivery, and tracing the possible root causes enable development of prevention strategy and improve efficiency for conducting a prospective cohort study of the associated complications.

Prediction of PROM is mostly diagnostic.[19-22] For prognostic prediction, a model of preterm PROM was recently developed using maternal factors during first trimester.[23] Based on a training set (*n*=10,280), the area under receiver operating characteristics curve (AUROC) was 0.667. Prediction of all-cause spontaneous preterm delivery also remains poor (AUROC 0.54 to 0.70; <37 weeks' gestation; *n*=118/2540) and exposed to high risk of bias based on a systematic review.[24] However, there was no development and external validation of a prognostic prediction model for PROM. In addition, because of reasonably challenging, no studies developed a model to estimate the time of delivery before the day.

For both classification and estimation tasks, machine learning demonstrate promising performances for predicting pregnancy outcomes[25] and other conditions.[26-28] Albeit its hype, most machine learning magnificent successes are diagnostic, especially deep learning models that surpass human-level performance.[29-31] Machine learning is not yet able to infer causality; thus, human learning is needed to estimate what would likely happen if the conditionals are different to those existing in a dataset which a machine learns from.[32] Nonetheless, solving PROM problems requires prediction and causal modeling to develop better prevention strategy either at population or individual level. To address this issue, we applied both human and machine learning. In human learning, we made a knowledge-based causal diagram as a central assumption with statistical learning as a tool to verify that assumption. This is expected to mitigate data-driven bias using contextual knowledge, systematically. In machine learning, we applied state-of-the-art algorithms. These included models found outperforming other prediction algorithms for pregnancy outcomes.[25] We also developed a deep-insight visible neural network (DI-VNN) based on recent studies.[33,34] But, unlike any of these studies, the algorithm and network architecture of our model allow deep exploration by human learning on 'subconscious mind' of the machine to catch both insight and bias exploited for prediction. Using only medical history, the model deployment would be accessible worldwide via web application. This study aimed to develop, validate, and deploy a prognostic prediction model for PROM and an estimator for the time of delivery using a nationwide health insurance database.

Results

We selected all visits (*n*=883,376) by 12-to-55-years-old women (*n*=219,272) to healthcare providers using a single-payer, nationwide health insurance, up to the end of pregnancy if the women are pregnant (Figure 1). We applied both simple and stratified random splitting for external validation. Common situations nationwide are reflected by the random-split subset (see Methods and Supplementary Information).

We have developed five models, using medical history and/or causal factors (Figure 2), for either predicting PROM (Figure 3) or estimating the time of delivery (Figure 4). The first model was a statistical model using ridge regression on the causal factors selected by human (causal RR). The second to fourth models were using principal components (PCs) selected by machine learning algorithms of elastic net regression (PC-ENR), random forest (PC-RF), and gradient boosting machine (PC-GBM). The fifth model was DI-VNN. This was developed to achieve moderate predictive performance but interpretable results. DI-VNN allows deep exploration on how this machine learning algorithm predicts the outcome (see Design of deep-insight visible neural network algorithm and watch Supplemental Video in Supplementary Information). We calibrated each model applying a general additive model by locally weighted scatterplot smoothing (GAM-LOESS). The calibration used ~20% split of a training set (~64% of all the selected visits). All models were also internally validated (see Methods).

Causal diagram

By systematic human learning (Algorithm 1), we constructed 12 causal diagrams of PROM based on 56 studies from PubMed, starting from guidelines of PROM by an authoritative institution;[35] in turn, a causal inference was conducted by outcome regression and inverse probability weighting (IPW) for each causal factor (Figure 1 to 12 and Table 11 to 13

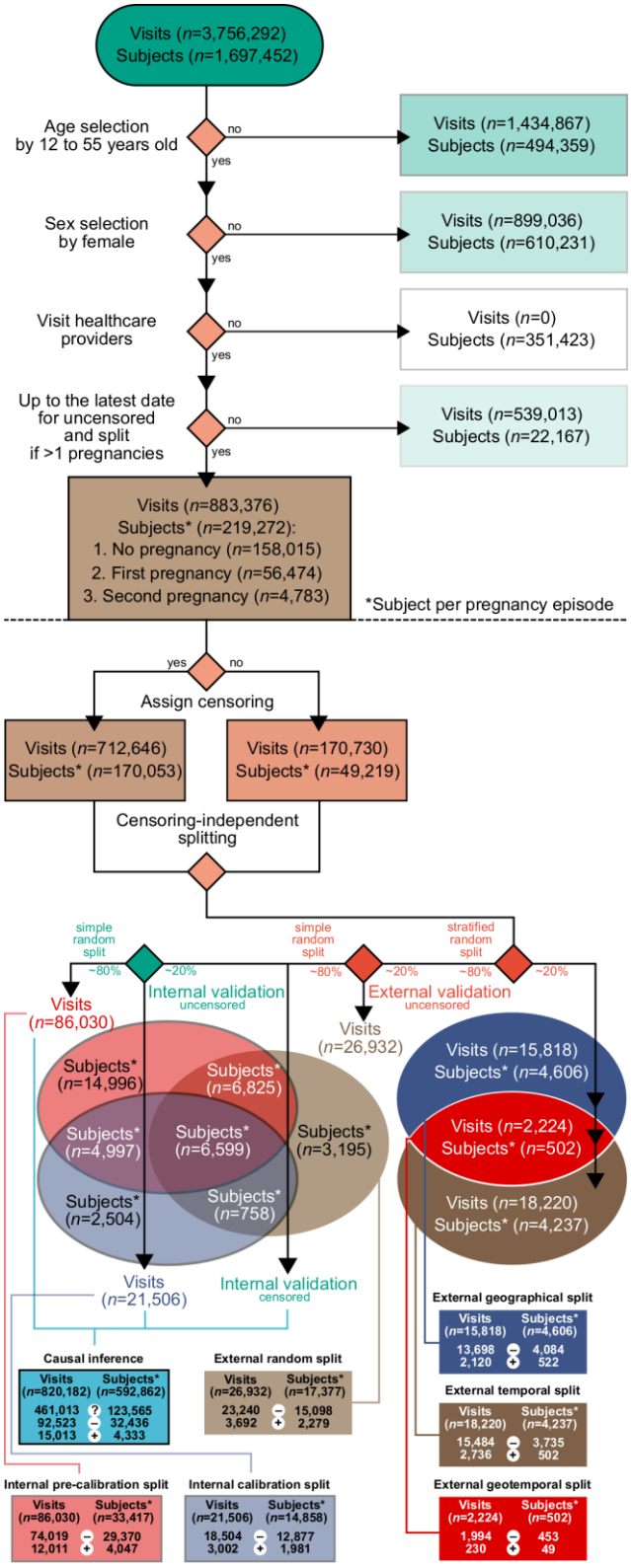

Figure 1. Subject selection applying retrospective design and data partition for internal and external validations. A set for causal inference included censored outcome. *n*, sample size; *, subject per pregnancy episode; (?) censoring; (–) nonevents; and (+) events.

in Supplementary Information). Our training set provided 12 out of 27 causal factors, in which 11 out of 12 factors were the causal factors by IPW. Chorioamnionitis (odds ratio [OR] 1.351, 95% CI 1.33 to 1.372), intra-amniotic infection (OR 1.118, 95% CI 1.083 to 1.153), and genital tract infection (OR 1.116, 95% CI 1.101 to 1.132) were the top three highest causal effects estimated by IPW. Polyhydramnios was not verified as a causal factor of PROM by both IPW (OR 0.998, 95% CI 0.989 to 1.006) and outcome regression (OR 1.238, 95% CI 0.851 to 1.801) using our data. Outcome regression mostly showed larger effects compared to those by IPW (Table 13 in Supplementary Information). The odds ratios were farther from 1 compared to those of IPW.

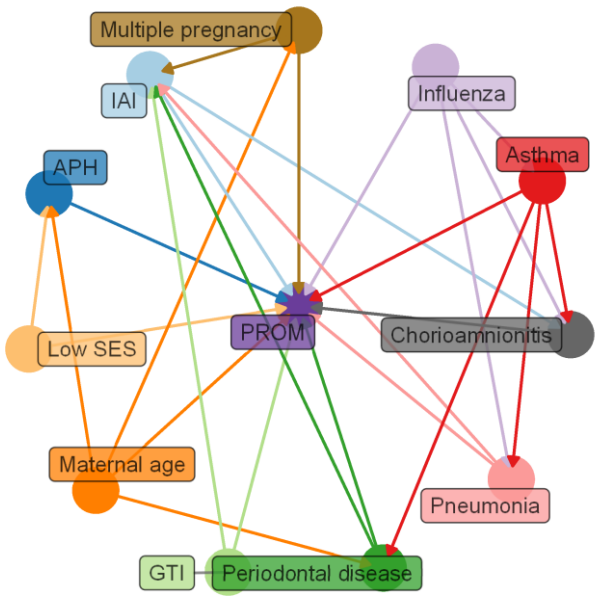

Figure 2. Causal diagram of PROM. All causal factors had been verified by inverse probability weighting (IPW) using our data. Causal inference was only conducted between PROM and each of the causal factors. Inter-causal factor relationship was not verified, but it demonstrates how a causal factor was included in the causal model of another causal factor in this figure. APH, ante-partum hemorrhage; GTI, genital tract infection; IAI, intra-amniotic infection; SES, socio-economic status.

In final causal diagram (Figure 2), infection- and immune-related conditions are seen: (1) influenza; (2) asthma; and (3) pneumonia. Maternal factors are also observed: (1) maternal age; (2) low socio-economic status; (3) multiple pregnancy; and (4) ante-partum hemorrhage. Both groups collide on periodontal disease then collides with genital tract infection on intra-amniotic infection to be continued to either chorioamnionitis or PROM.

Prognostic prediction of premature rupture of membranes

After calibration (Figure 3a), by internal validation of the calibration split, two well-calibrated models were

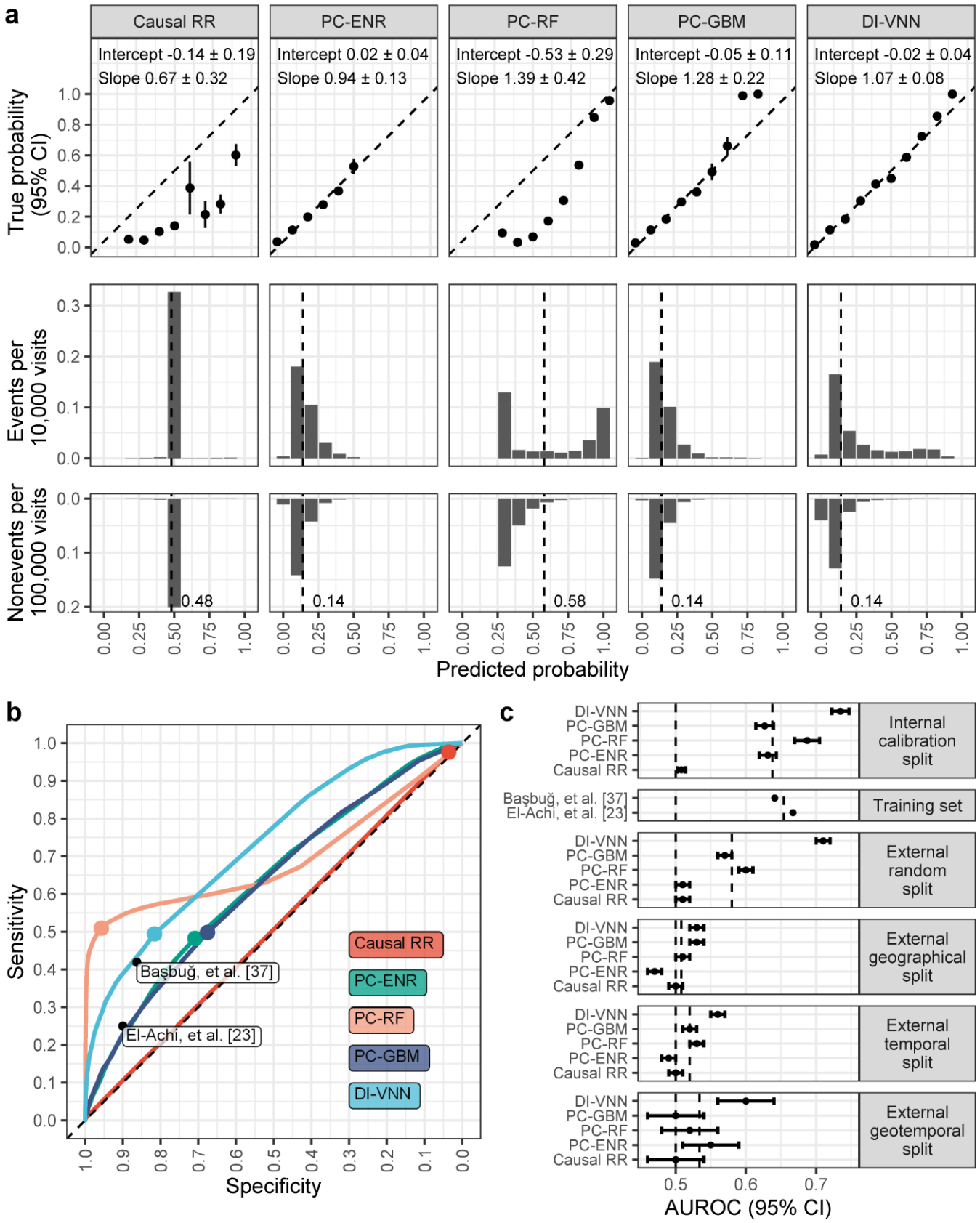

Figure 3. Model evaluation: (a) calibration; (b) receiver operating characteristics (ROC); (c) areas under ROC (AUROCs). Showing thresholds (a, b) and average AUROCs per set (c). DI-VNN, deep-insight visible neural network; ENR, elastic net regression; GBM, gradient boosting machine; PC, principal component; RF, random forest; RR, ridge regression.

PC-ENR and DI-VNN; however, distribution of predicted probabilities from 0 to 1 was mostly covered by the latter model only. For clinical application, this will help adjusting threshold widely, depending on local data distribution. Using internal validation, the optimum threshold for DI-VNN is 0.14. Similar to PC-RF, distribution of the predicted probabilities was

visually differentiated between those of events and nonevents.

Receiver operating characteristics (ROC) curves by internal validation (calibration split) were compared among models (Figure 3b). To screen a disease, sensitivity is an important metric from a population-level standpoint. At 95% specificity, the most sensitive model is PC-RF (0.513, 95% CI 0.509 to 0.517); unfortunately, this model is not well-calibrated. At the same specificity, DI-VNN followed PC-RF by sensitivity of 0.297 (95% CI 0.293 to 0.301; threshold at 0.29). By optimum threshold at 0.14, DI-VNN achieved sensitivity of 0.494 (95% CI 0.489 to 0.5), almost similar to PC-RF, with specificity of 0.816 (95% CI 0.814 to 0.818).

To confirm DI-VNN robustness on this task, we can check if this model consistently outperforms other models and AUROC of 0.5. By external validation (Figure 3c), DI-VNN was the best model by AUROC in most of the sets. The AUROCs was reasonably lower than that by a training set (0.73, 95% CI 0.72 to 0.75). For random split that reflected common situations nationwide, DI-VNN achieved AUROC of 0.71 (95% CI 0.70 to 0.72). For other external validations, the AUROCs of DI-VNN were lower than that for the random split but higher than either the average AUROCs of all models or AUROC of 0.5.

To compare with the models from previous studies, we followed methods in the preferred reporting items for systematic reviews and meta-analyses (PRISMA) 2020 expanded checklist (Table 4 in Supplementary Information).[36] From three literature databases (see Methods), we identified 209 non-duplicated records. These were screened, retrieved, and assessed to find two prediction models[23,37] for preterm PROM (Figure 13 and Table 4 to 5 in Supplementary Information). Prognostic prediction of PROM by DI-VNN and PC-RF achieved AUROCs that were higher than those of the previous models (Figure 3b and 3c): (1) a logistic regression using maternal factors with AUROC of 0.667 (sensitivity 0.25 and specificity 0.90) but without internal validation (144 preterm PROM; 10,136 not preterm PROM);[23] and (2) a prediction rule using serum alpha-fetoprotein with AUROC of 0.641 (sensitivity 0.419 and sensitivity 0.863) but without internal validation (31 preterm PROM; 1,146 not preterm PROM).[37]

Estimation of the time of delivery

Preterm PROM is reasonably an outcome of interest in previous studies because clinical consequences of PROM are related to preterm delivery for the mother or prematurity for the baby; thus, we also developed a prediction model to estimate the time of delivery. Since diagnosis/procedure codes for preterm delivery or premature newborn may not be consistently assigned to every cases, we decided to predict how many days from current visit a mother will deliver, instead of predicting a particular code that refers to preterm delivery. Albeit challenging, we viewed this estimation task giving a benefit if the model precisely estimates the time of delivery. This is reasonably acceptable if the predicted time is included within an interval estimate of the true one and the interval is maximum $\pm x$ weeks when predicting $> x$ weeks.

By these criteria, we determined the best estimation model was PC-RF (Figure 4). Using internal validation set, this model fulfilled the criteria for 78.57% (Figure 4a). This means most of the 42 weeks were precisely predicted. Unfortunately, estimated time of delivery by DI-VNN was within the 6th week from the prediction dates of any samples. This is considerably due to differential analysis (see Methods), which filtered the predictors based on categorical outcome only. The estimation model was developed based on those predictors. Nonetheless, we used DI-VNN, as the best classification model, to stratify the estimated time of delivery, including that estimated by PC-RF. We also confirmed this model consistently outperforming the other models for the time estimation task using the external validation sets. The comparison of estimation performances is described further (Supplementary Information). In addition, the estimation task is reasonably challenging, particularly in external validation sets because of differences in distributions of the times of deliveries. More details on the distributions are described (Supplementary Information).

We also determined the precise estimation window of PC-RF using internal validation set (Figure 4b). In nonevents, the maximum estimated time of delivery was 36 weeks from a visit when the prediction is conducted, while the corresponding true time of delivery was 42 weeks. This coincides with the maximum duration of pregnancy. In events, the maximum predicted time was 6 to 10 weeks earlier. This implies the event, which was PROM modeled by DI-VNN, tends to have a shorter duration of pregnancy, as modeled by PC-RF, compared to the nonevent.

Design of deep-insight visible neural network

This model works like neural network in a brain. Predictors are feed as inputs into multiple neurons and the outputs follow activation function with all-or-none principle. These inputs are arrayed as a minimum of two-dimensional imaging such an object projection onto retina. We call this as ontology array. A predictor that is correlated to another predictor would have

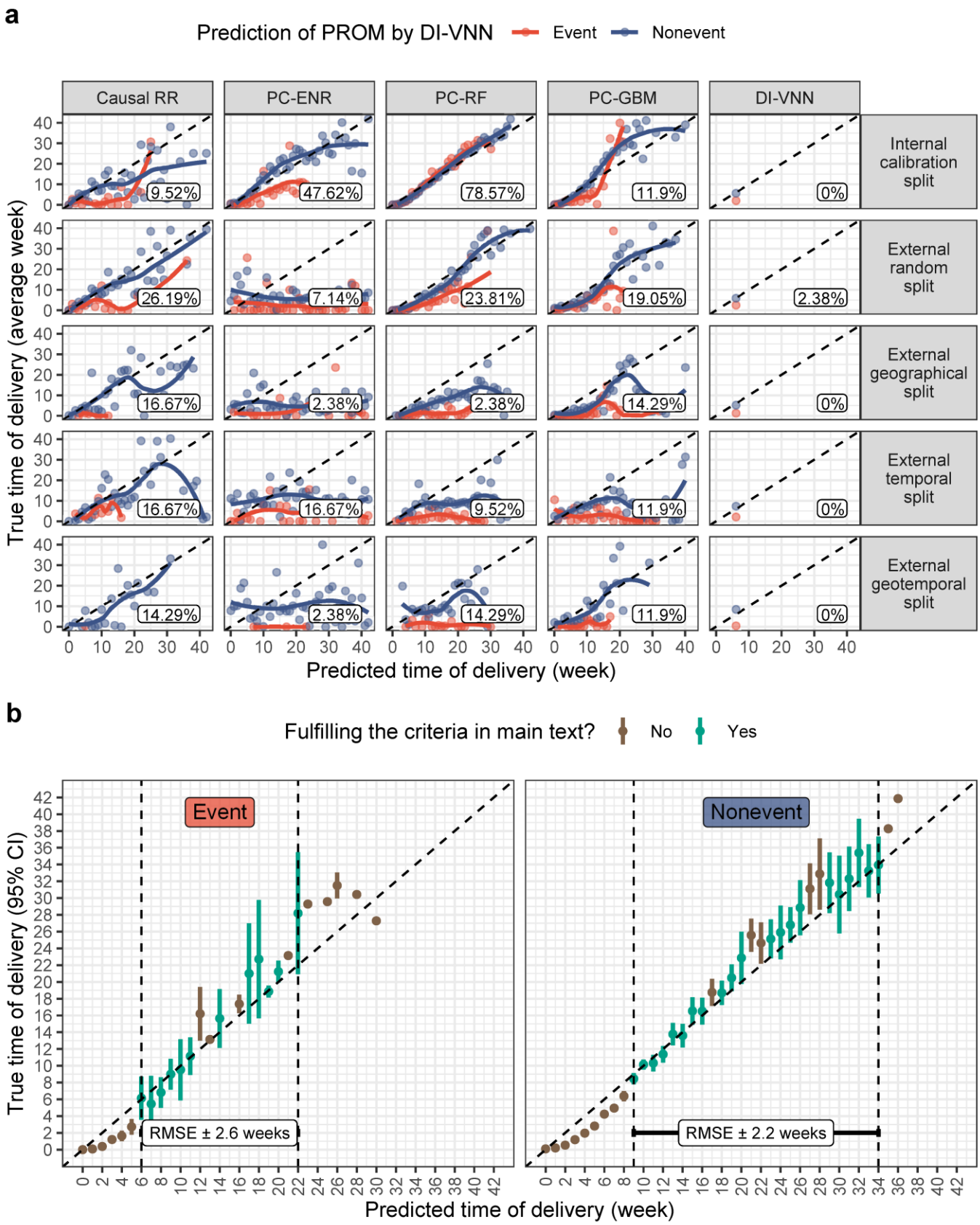

Figure 4. Estimation plots: (a) model comparison; (b) PC-RF estimation window. Showing % criteria fulfilled (a) and precision (95% CI) per week (b). DI-VNN, deep-insight visible neural network; ENR, elastic net regression; GBM, gradient boosting machine; PC, principal component; RF, random forest; RMSE, root mean squared error; RR, ridge regression.

closer position than another less-correlated predictor. From the receptive field in retina, the signals propagate to the primary visual cortex then the visual association cortex. The signal pathway of neural network is dedicated for specific part of the array. We called this ontology network. By seeing similar object with uncertain variation, the neural network is maintained at specific activation thresholds and

weights. This makes human have visual memory in the association cortex to recognize a particular object by segmenting it into several parts.

Exploring deep-insight visible neural network

An interactive figure and table of DI-VNN are provided in our web application in a public repository (https://predme.app/promtime), allowing a user to explore this model at population level in addition to that at individual level (see next section). For exploratory data analysis, we also computed an interval estimate of AUROC for every 4 weeks before the end of pregnancy using internal validation set (Figure 5a). Most of the AUROC intervals were higher than 0.5 from 44 ± 2 weeks before the end of pregnancy. This may cover up to a full period of pregnancy (maximum 42 weeks) and approximately a month before the beginning of pregnancy.

In DI-VNN, the network architecture is data-driven (Figure 5b). By clique-extracted ontology (CliXO)[38] algorithm, we constructed an architecture for the convolutional neural network (CNN) using only training set (see Methods). Both reasoning and technical details are described for this exploration (Supplementary Information). First, we can explore each node of DI-VNN at population level. The diagnosis/procedure codes constructing the feature members were International Classification of Disease version 10 (ICD-10) codes. We can start from the most visually-distinguished arrays, as depicted (Figure 5b), which was ONT:171. There were N760 (acute vaginitis) and B379 (unspecified candidiasis). Positive is not straightforward to an event. Nevertheless, positive and negative (color-coded) outputs tend to contribute on the opposite outcomes, interpreted based on external contextual knowledge. Another distinguished array, which is ONT:144, connected to the same node with that by the previous array. The feature member (9059, other microscopic examination of blood) tends to contribute on the same outcome with that unspecified candidiasis tends to contribute on. Unlike acute vaginitis, which is a local infection, both B379 and 9059 may be related to systemic infection. Up to this point, we have got an insight describing coincidence between systemic vs. local infection and PROM.

Another array ONT:154 was also visually-distinguished. Although the value of the feature member causal_A03 (chorioamnionitis) was zero, it was next to higher values that support the same outcome with that acute vaginitis supports. We can trace the node on the upper layer is ONT:171. This array had similar value distribution on the same channel ($z = 2$) with ONT:154. More details on explanation for this phenomenon are described (Supplementary Information).

Position of a feature within an array of any ontology terms were determined using *t*-moderated stochastic network embedding (*t*-SNE) using Barnes-Hut approximation (see Methods).[39] By understanding how this algorithm works (Supplementary Information), we may assume causal_A03 (ONT:154), which is chorioamnionitis, is closer with acute vaginitis than the unspecified candidiasis since both at the same channel and the positions at the first and second dimensions are adjacent and that *t*-SNE preserves neighborhood identity. Acute vaginitis is semantically a genital tract infection (GTI). By external contextual knowledge inferred by the systematic human learning and confirmed by IPW using our data, GTI and chorioamnionitis were causal factors of PROM.

Exploring this DI-VNN should be with caution since we developed the model using medical histories, which were diagnosis or procedure codes given by medical doctors. We may or may not model a human pathophysiology using this algorithm, but, we definitely model the doctor's behavior on coding a diagnosis or procedure.[40] By providing an interface to the internal properties of this model, a human user may assess each prediction case-by-case. More details related to this phenomenon are described in this exploration (Supplementary Information).

Web application

A web application is provided using DI-VNN for the PROM prognostic prediction and PC-RF for the time of delivery estimation. We hosted this application on a public repository for clinical prediction models (https://predme.app/promtime). This allows a user, e.g. a doctor, to make a decision after critically appraising an individual prediction by several approaches. We described a use case example (Supplemental Information), reflecting a real-world situation.

Briefly, we uploaded a record of 20 visits from December 2nd 2015 to July 30th 2016, consisting 28 code entries, by a 19-years-old female subject. After determining the date of prediction, which was set on July 30th 2016, we ran the application in 5.14 minutes (95% CI 5.11 to 5.18 minutes when we repeated for 10 times). We could download the report after the application was done (Figure 6). The predicted outcome was PROM, and the estimated time of delivery was 11 weeks after the time of prediction. The predicted probability was 0.867.

Eventually, a user may want to know if the PROM prediction and the estimated time of delivery are similar to the true values. One can save this model online and come back later to the web application to enter the true outcome and time of delivery. By this way, a user may collect data for external validation purpose, specifically describing the model

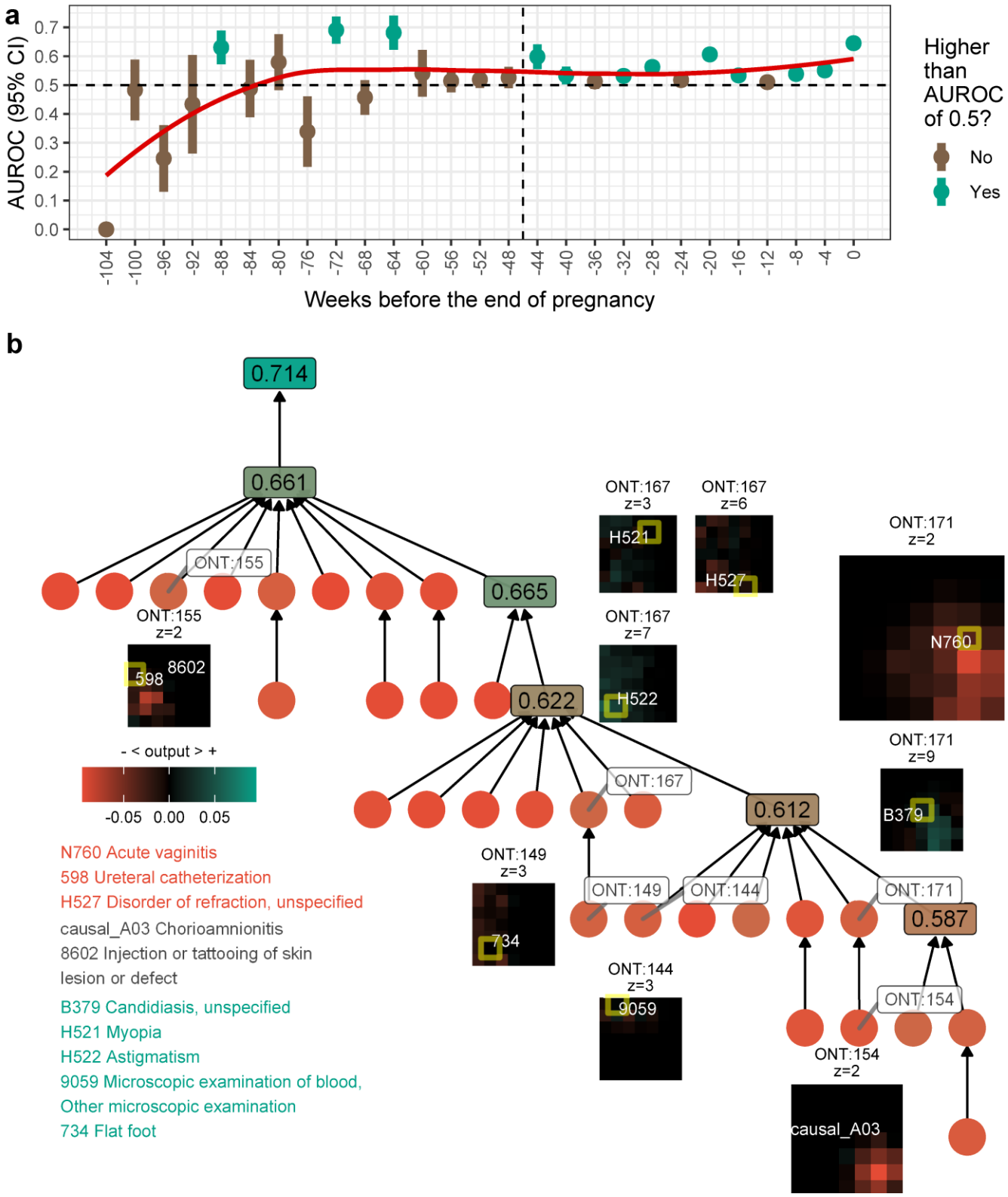

Figure 5. Exploratory data analysis of DI-VNN: (a) areas under receiver operating characteristics curve (AUROC) every 4 weeks; (b) ontology network and arrays of DI-VNN. Showing the best time window for the prediction by DI-VNN (a) and AUROCs >0.55 for prediction using parts of the network architecture up to each layer which a node resides (b). Each node is a CliXO term, prefixed by ONT. Only showing those with distinguished output arrays for a particular channel denoted by 'z'. A feature in the array may tend to positive or negative output, color-coded based on the gradient as shown, including the feature description. Yellow square in an array refer to a feature if only its output is non-zero. A feature may not have this square, e.g., causal_A03 and 8602. ONT:154 is an example of a backpropagation effect from ONT:171. CliXO, clique-extracted ontology; DI-VNN, deep-insight visible neural network.

performances based on local data distribution. In our case, the true outcome was also PROM and the time of delivery is 12 weeks after the time of prediction, a week later than the predicted time of delivery.

Discussion

To ensure a rigorous conduct of research, we applied three standard guidelines to conduct and report this study, specifically designed for a multivariable prediction model applying a machine learning algorithm such that is suitable in healthcare.[41-43] For a fair comparison to the previous models, we also followed methods in PRISMA 2020 expanded checklist.[36] We also applied four approaches based on the previous methods[33,34,39,44-48] to develop the pipeline with several modifications. R packages of some approaches were developed to help future investigators for easily applying these approaches on their studies (see Methods). Methodological reasoning on novelty of these approaches is also discussed (Supplementary Information).

The DI-VNN model in this study outperformed the previous models[23,37], using larger training set and external validation sets (8,778 visits and 3,352 subjects for events only), and no need for a biomarker testing. External validation by stratified random splitting was also applied for prediction of gestational diabetes using a nationwide electronic health records from a health insurance company.[49] The PC-RF model also estimated the time of delivery in days or weeks to predict preterm delivery, while previous models only predict whether a preterm delivery happens in the future without estimating the date interval.[24] Our models also used cohort paradigm preventing temporal bias in delivery prediction by either a statistical or deep learning models.[50]

To get an insight of the PROM antecedent, we have shown how this was achieved by exploring the model at both population (Figure 5) and individual levels (Figure 6). Based on population-level exploration of DI-VNN, a role of a systemic vs. local infection is implied in PROM. As inferred from the PC-RF model, both explorations obviously showed preterm PROM. This disease may be considered as an infection-related preterm delivery. Clustering analysis of placental gene signatures assigned this type of preterm delivery as a subclass of preeclampsia.[51] The other subclasses were early-onset preeclampsia, immunological preeclampsia (with fetal growth restriction), preeclampsia with chromosomal abnormalities, and another subclass consisting several conditions. The last subclass included women with healthy placenta but the pregnancy is ended as non-infection preterm delivery, normotensive term delivery, and preeclampsia with maternal cardiovascular risk factors. The DI-VNN model in this study also included medical histories of all subtypes of preeclampsia (Table 20 in Supplementary Information). Several features in DI-VNN were counterintuitive, e.g. H527 (unspecified disorder of refraction), 734 (flat foot), H521 (myopia), and H522 (astigmatism). Yet, these describe blurry vision and swelling in the feet, which are also symptoms of preeclampsia. The DI-VNN model likely used preeclampsia-related codes as competing risks to predict PROM. Similarly, competing risk models were also developed for preeclampsia.[52] Similar eye-related codes, i.e. myopia and astigmatism, were also found important to predict preeclampsia by a random forest model.[25]

If we assume PROM as a subclass of preeclampsia, this may also explain why population-level exploration implying an insight of systemic vs. local infection by competing risk. A hematogenous infection are associated with reproductive-tract microbial dysbiosis and affects several pregnancy outcomes, including PROM, preeclampsia, and fetal growth restriction.[53] Both hematogenous and ascending infection from reproductive tract were found in PROM.[54,55] The hematogenous infection included those from digestive and respiratory organs. Similar to population-level exploration (Figure 5), we also found a possibility of hematogenous infection at individual level (Figure 6), which were from infectious gastroenteritis and unspecified acute upper respiratory infection (Supplementary Information). Both population- and individual-level explorations also implied a period surrounding the beginning of pregnancy as the onset of optimum prediction time of PROM. This is also similar to a finding based on predictive modeling for preeclampsia.[25]

Our DI-VNN model has higher sensitivity and specificity than the previous models.[23,37] Improvement on sensitivity given the same specificity may be interpreted as a potential improvement in safety of a patient with PROM, including that in low-resource setting. Meanwhile, improvement on specificity given the same sensitivity is indicated as a potential cost reduction if we apply DI-VNN as a preliminary model to decide a prognostic test in high-resource setting. Cost reduction is also potential if we use DI-VNN to help in subject selection in a long-term prospective cohort, e.g. one for a study on an etiology, prognostic factor, or intervention in order to develop a prevention strategy for a complication of PROM. However, an impact study is needed to verify these assumptions. For clinical implication, our models allow collaboration between human and machine. Instead of developing automation in clinical decision making, a human-machine framework is more sensible for precision medicine with a subpopulation approach. Our models provide such framework. A clinician can use population-level data case-by-case, assisted by machine. We did not only rely on the DI-VNN and PC-RF models but also datasets to estimate the model performances at individual level based on subpopulation similar to that individual,[56] collaborating machine learning and evidence-based medicine.[57]

Identity and results

Name: _____ Age: _____

Input: 20 visits consisting 28 code entries

Time of prediction: 2016-07-30

Estimated time of delivery: 2016-10-18

Predicted outcome: Prelabor rupture of membranes (PROM)

Predicted probability: 0.867 (threshold=0.67)

Population-level performances

Metric	Value
Sensitivity	0.107 (95% CI 0.104 to 0.11)
Specificity	0.996 (95% CI 0.996 to 0.996)
Positive Predictive Value	0.809 (95% CI 0.798 to 0.82)
Negative Predictive Value	0.873 (95% CI 0.872 to 0.874)
Timing	11 (95% CI 9 to 13)

Timeline

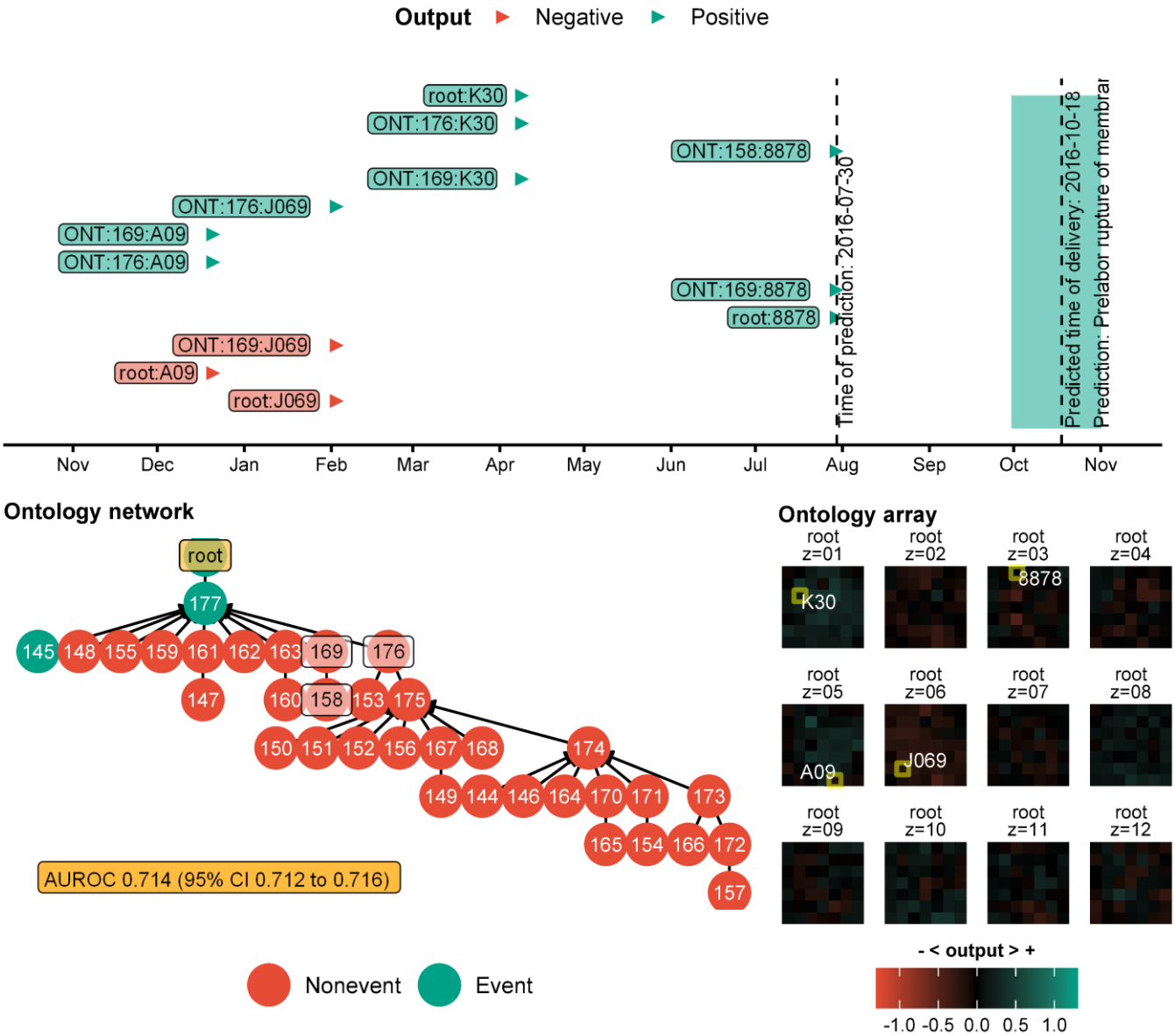

Figure 6. An report example of PROM prediction and estimation of the time of delivery. An ontology term in the timeline is prefixed by ONT, followed by the number and one of the feature members. 8878, diagnostic ultrasound of gravid uterus; A09, diarrhea and gastroenteritis of presumed infectious origin; AUROC, area under receiver operating characteristics curve; J069, unspecified acute upper respiratory infection; K30, dyspepsia; PROM, prelabor rupture of membranes.

This framework also enables exploration of a medical history of a patient, answering key challenges to evaluate model weakness,[58] verify if the predicted outcome is reasonable,[59] and make sense clinical prediction by utilizing electronic medical records.[60] Clinical implications by our novel approaches on the methodology are discussed further (Supplemental Information).

However, we noticed several limitations from our study. For systematic human learning (see Methods), the snowball sampling may not consider a potentially novel cause of PROM since this only goes through the existing evidences exhaustively. We also

cannot provide data for several causal factors of PROM; thus, several backdoors or confounding factors were not blocked. One may also find that the causal effects were considerably small as inferred by IPW. A factor that mainly causes PROM may not be included yet. Although the DI-VNN in this study was the best prediction model among others and outperformed those from previous studies,[23,37] we can only apply it as a preliminary model of PROM because a clinically-acceptable AUROC is 0.8 or higher.[42] More sensitive models are still needed to use as a second-line model. Yet, we can apply this model to selectively offer tests that need high-resource setting in order to apply the second-line model. This strategy leads to an efficient prediction. In addition, DI-VNN is obviously for exploration only beyond its predictive ability. Any findings should be with caution and need more explanatory studies. We also realized DI-VNN is not easy to interpret. But, this model already expanded a larger yet comprehensible space for hypothesis exploration. A human user with specific competences in medicine, statistics, and machine learning is needed to explore DI-VNN with a good confidence. A clinician should also consider the definitions used to diagnose conditions or determine procedures whose codes are used in DI-VNN as the predictors, which are the definitions used in the country of the dataset. Another consideration is population in the dataset which covered Asian and Austronesian. As we recommend in a clinician checklist[42] for assessing suitability of machine learning applications in healthcare (Table 3 in Supplementary Information), external validation and determining optimum using local data are needed.

In conclusion, our DI-VNN model was found robust for PROM prognostic prediction. The PC-RF model was reasonably precise within a specific time window based on predicted outcome by the DI-VNN. A clinical trial and an impact study are warranted for these models. Exploration utilizing internal properties of DI-VNN also support PROM as a subclass of preeclampsia. Both conditions may share a common causes and pathophysiological derangement involving microbial communities. Future investigations are needed at molecular level to confirm these findings. In addition, DI-VNN also opens intensive collaboration between human and machine for more sensible clinical prediction. Yet, this requires a user with multi-disciplinary competences.

Acknowledgments

The social security administrator for health or *badan penyelenggara jaminan sosial (BPJS) kesehatan* in Indonesia gave permission to access the sample dataset in this study (dataset request approval no.: 5064/I.2/0421). This work was supported by the Ministry of Science and Technology (MOST) in Taiwan (grant number MOST109-2221-E-038-018) and the Higher Education Sprout Project from the Ministry of Education (MOE) in Taiwan (grant number DP2-110-21121-01-A-13) to Emily Chia-Yu Su. The sponsor had no role in the research design or contents of the manuscript for publication.

Author contributions

HS, YWW, and ECYS developed the concept and design of this study. Dataset access was requested by HS. This author extracted and processed the data, performed training and validation of machine learning algorithms, conducted the literature search and wrote the draft of the manuscript. This author and YWW independently assessed the eligibility criteria of ambiguous, reviewed studies which were previously determined by HS. YWW and ECYS critically revised the drafted manuscript. All authors approved the submitted manuscript and agreed to be personally accountable for the author's own contributions and to ensure that questions related to the accuracy or integrity of any part of the work, including ones in which the author was not personally involved.

Competing interests

HS, YWW, and ECYS declare no competing interests.

Data availability

The R Markdown, R Script, and others are available in https://github.com/herdiantrisufriyana/prom. To get raw data, one need to request an access from the BPJS Kesehatan for their sample dataset published in August 2019. Up to this date, there are three sample datasets they published in February 2019, August 2019, and December 2020. For the first and second versions, a request is applied via https://e-ppid.bpjs-kesehatan.go.id/, while the third is applied via https://data.bpjs-kesehatan.go.id. To preprocess the raw data into the input dataset of this study, follow the codes of the R Markdown in https://github.com/herdiantrisufriyana/medhist/preprocessing.

References

1. Salman, L., *et al.* Predictors for cesarean delivery in preterm premature rupture of membranes. *J Matern Fetal Neonatal Med* **33**, 3761-3766 (2020).
2. Mikołajczyk, M., Wirstlein, P., Adamczyk, M., Skrzypczak, J. & Wender-Ożegowska, E. Value of cervicovaginal fluid cytokines in prediction of fetal inflammatory response syndrome in pregnancies complicated with

preterm premature rupture of membranes (pPROM). *J Perinat Med* **48**, 249-255 (2020).
3. Cobo, T.*, et al.* Contribution of Amniotic Fluid along Gestation to the Prediction of Perinatal Mortality in Women with Early Preterm Premature Rupture of Membranes. *Fetal Diagn Ther* **43**, 105-112 (2018).
4. Toukam, M.E.*, et al.* A predictive neonatal mortality score for women with premature rupture of membranes after 22-27 weeks of gestation. *J Matern Fetal Neonatal Med* **32**, 258-264 (2019).
5. Esteves, J.S., de Sá, R.A., de Carvalho, P.R. & Coca Velarde, L.G. Neonatal outcome in women with preterm premature rupture of membranes (PPROM) between 18 and 26 weeks. *J Matern Fetal Neonatal Med* **29**, 1108-1112 (2016).
6. Duncan, J.R., Dorsett, K.M., Vilchez, G., Schenone, M.H. & Mari, G. Uterine artery pulsatility index for the prediction of obstetrical complications in preterm prelabor rupture of membranes. *J Matern Fetal Neonatal Med*, 1-4 (2019).
7. Musilova, I.*, et al.* Gastric fluid used to assess changes during the latency period in preterm prelabor rupture of membranes. *Pediatr Res* **84**, 240-247 (2018).
8. Sim, W.H., Ng, H. & Sheehan, P. Maternal and neonatal outcomes following expectant management of preterm prelabor rupture of membranes before viability. *J Matern Fetal Neonatal Med* **33**, 533-541 (2020).
9. Menon, R. & Richardson, L.S. Preterm prelabor rupture of the membranes: A disease of the fetal membranes. *Semin Perinatol* **41**, 409-419 (2017).
10. Prelabor Rupture of Membranes: ACOG Practice Bulletin, Number 217. *Obstet Gynecol* **135**, e80-e97 (2020).
11. Braun, D.*, et al.* Trends in Neonatal Intensive Care Unit Utilization in a Large Integrated Health Care System. *JAMA Netw Open* **3**, e205239 (2020).
12. Speer, R.R., Schaefer, E.W., Aholoukpe, M., Leslie, D.L. & Gandhi, C.K. Trends in Costs of Birth Hospitalization and Readmissions for Late Preterm Infants. *Children (Basel)* **8**(2021).
13. Chellani, H., Mittal, P. & Arya, S. Mother-Neonatal Intensive Care Unit (M-NICU): A Novel Concept in Newborn Care. *Indian Pediatr* **55**, 1035-1036 (2018).
14. Umran, R.M. & Al-Jammali, A. Neonatal outcomes in a level II regional neonatal intensive care unit. *Pediatr Int* **59**, 557-563 (2017).
15. Shrestha, D.*, et al.* Predicting deaths in a resource-limited neonatal intensive care unit in Nepal. *Trans R Soc Trop Med Hyg* **111**, 287-293 (2017).
16. Horbar, J.D.*, et al.* Racial Segregation and Inequality in the Neonatal Intensive Care Unit for Very Low-Birth-Weight and Very Preterm Infants. *JAMA Pediatr* **173**, 455-461 (2019).
17. Murthy, S., Leligdowicz, A. & Adhikari, N.K. Intensive care unit capacity in low-income countries: a systematic review. *PLoS One* **10**, e0116949 (2015).
18. Ho, T., Zupancic, J.A.F., Pursley, D.M. & Dukhovny, D. Improving Value in Neonatal Intensive Care. *Clin Perinatol* **44**, 617-625 (2017).
19. Julien, M.*, et al.* Premature rupture of the membranes: analytical evaluation of diagnostic tests. *Ann Biol Clin (Paris)* **76**, 300-306 (2018).
20. Eldaly, A.*, et al.* Use of beta subunit of human chorionic gonadotropin assay as a diagnostic tool for prelabor rupture of membranes. *J Matern Fetal Neonatal Med* **32**, 1965-1970 (2019).
21. Ireland, K.E., Rodriguez, E.I., Acosta, O.M. & Ramsey, P.S. Intra-amniotic Dye Alternatives for the Diagnosis of Preterm Prelabor Rupture of Membranes. *Obstet Gynecol* **129**, 1040-1045 (2017).
22. Musilova, I.*, et al.* Vaginal fluid interleukin-6 concentrations as a point-of-care test is of value in women with preterm prelabor rupture of membranes. *Am J Obstet Gynecol* **215**, 619.e611-619.e612 (2016).
23. El-Achi, V.*, et al.* First-Trimester Prediction of Preterm Prelabour Rupture of Membranes. *Fetal Diagn Ther* **47**, 624-629 (2020).
24. Meertens, L.J.E.*, et al.* Prediction models for the risk of spontaneous preterm birth based on maternal characteristics: a systematic review and independent external validation. *Acta Obstet Gynecol Scand* **97**, 907-920 (2018).
25. Sufriyana, H.*, et al.* Comparison of Multivariable Logistic Regression and Other Machine Learning Algorithms for Prognostic Prediction Studies in Pregnancy Care: Systematic Review and Meta-Analysis. *JMIR Med Inform* **8**, e16503 (2020).
26. Fleuren, L.M.*, et al.* Machine learning for the prediction of sepsis: a systematic review and meta-analysis of diagnostic test accuracy. *Intensive Care Med* **46**, 383-400 (2020).
27. Lee, Y.*, et al.* Applications of machine learning algorithms to predict therapeutic outcomes in depression: A meta-analysis and systematic review. *J Affect Disord* **241**, 519-532 (2018).
28. Gonem, S., Janssens, W., Das, N. & Topalovic, M. Applications of artificial intelligence and machine learning in respiratory medicine. *Thorax* **75**, 695-701 (2020).
29. Bien, N.*, et al.* Deep-learning-assisted diagnosis for knee magnetic resonance imaging: Development and retrospective validation of MRNet. *PLoS Med* **15**, e1002699 (2018).
30. Hannun, A.Y.*, et al.* Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nat Med* **25**, 65-69 (2019).

31. Rajpurkar, P., *et al.* Deep learning for chest radiograph diagnosis: A retrospective comparison of the CheXNeXt algorithm to practicing radiologists. *PLoS Med* **15**, e1002686 (2018).
32. Wilkinson, J., *et al.* Time to reality check the promises of machine learning-powered precision medicine. *Lancet Digit Health* **2**, e677-e680 (2020).
33. Sharma, A., Vans, E., Shigemizu, D., Boroevich, K.A. & Tsunoda, T. DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture. *Sci Rep* **9**, 11399 (2019).
34. Ma, J., *et al.* Using deep learning to model the hierarchical structure and function of a cell. *Nat Methods* **15**, 290-298 (2018).
35. ACOG. Practice Bulletin No. 172: Premature Rupture of Membranes. *Obstet Gynecol* **128**, e165-177 (2016).
36. Page, M.J., *et al.* The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *Bmj* **372**, n71 (2021).
37. Başbuğ, D., Başbuğ, A. & Gülerman, C. Is unexplained elevated maternal serum alpha-fetoprotein still important predictor for adverse pregnancy outcome? *Ginekol Pol* **88**, 325-330 (2017).
38. Kramer, M., Dutkowski, J., Yu, M., Bafna, V. & Ideker, T. Inferring gene ontologies from pairwise similarity data. *Bioinformatics* **30**, i34-42 (2014).
39. Maaten, L.V.D. Accelerating t-SNE using tree-based algorithms. *J. Mach. Learn. Res.* **15**, 3221–3245 (2014).
40. Beaulieu-Jones, B.K., *et al.* Machine learning for patient risk stratification: standing on, or looking over, the shoulders of clinicians? *NPJ Digit Med* **4**, 62 (2021).
41. Luo, W., *et al.* Guidelines for Developing and Reporting Machine Learning Predictive Models in Biomedical Research: A Multidisciplinary View. *J Med Internet Res* **18**, e323 (2016).
42. Scott, I., Carter, S. & Coiera, E. Clinician checklist for assessing suitability of machine learning applications in healthcare. *BMJ Health Care Inform* **28**(2021).
43. Moons, K.G.M., *et al.* PROBAST: A Tool to Assess Risk of Bias and Applicability of Prediction Model Studies: Explanation and Elaboration. *Ann Intern Med* **170**, W1-w33 (2019).
44. Ritchie, M.E., *et al.* limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res* **43**, e47 (2015).
45. Stalpers, L.J.A. & Kaplan, E.L. Edward L. Kaplan and the Kaplan-Meier Survival Curve. *BSHM Bulletin: Journal of the British Society for the History of Mathematics* **33**, 109-135 (2018).
46. Goodman, L.A. Snowball Sampling. *The Annals of Mathematical Statistics* **32**, 148-170, 123 (1961).
47. Hernán, M.A. & Robins, J.M. *Causal Inference: What If.* , (Chapman & Hall/CRC, Boca Raton, 2020).
48. Lee, J. & Spratling, R. Recruiting Mothers of Children With Developmental Disabilities: Adaptations of the Snowball Sampling Technique Using Social Media. *J Pediatr Health Care* **33**, 107-110 (2019).
49. Artzi, N.S., *et al.* Prediction of gestational diabetes based on nationwide electronic health records. *Nat Med* **26**, 71-76 (2020).
50. Yuan, W., *et al.* Temporal bias in case-control design: preventing reliable predictions of the future. *Nat Commun* **12**, 1107 (2021).
51. Leavey, K., *et al.* Unsupervised Placental Gene Expression Profiling Identifies Clinically Relevant Subclasses of Human Preeclampsia. *Hypertension* **68**, 137-147 (2016).
52. Wright, D., Wright, A. & Nicolaides, K.H. The competing risk approach for prediction of preeclampsia. *Am J Obstet Gynecol* **223**, 12-23.e17 (2020).
53. Amir, M., *et al.* Maternal Microbiome and Infections in Pregnancy. *Microorganisms* **8**(2020).
54. Romero, R., *et al.* Evidence that intra-amniotic infections are often the result of an ascending invasion - a molecular microbiological study. *J Perinat Med* **47**, 915-931 (2019).
55. Solt, I. The human microbiome and the great obstetrical syndromes: a new frontier in maternal-fetal medicine. *Best Pract Res Clin Obstet Gynaecol* **29**, 165-175 (2015).
56. Ng, K., Kartoun, U., Stavropoulos, H., Zambrano, J.A. & Tang, P.C. Personalized treatment options for chronic diseases using precision cohort analytics. *Sci Rep* **11**, 1139 (2021).
57. Scott, I., Cook, D. & Coiera, E. Evidence-based medicine and machine learning: a partnership with a common purpose. *BMJ Evid Based Med* (2020).
58. Yu, K.H., Beam, A.L. & Kohane, I.S. Artificial intelligence in healthcare. *Nat Biomed Eng* **2**, 719-731 (2018).
59. Rajkomar, A., Dean, J. & Kohane, I. Machine Learning in Medicine. *N Engl J Med* **380**, 1347-1358 (2019).
60. Beam, A.L. & Kohane, I.S. Big Data and Machine Learning in Health Care. *Jama* **319**, 1317-1318 (2018).

Methods

This study is a part of DI-VNN project that applies our algorithm to variety outcomes of prediction. We followed guidelines for developing and reporting machine learning predictive models in biomedical research.[41] We also conducted a self-assessment following PROBAST guidelines.[43] These guidelines is specific for multivariable prediction models for making individualized, prognostic or diagnostic prediction, instead of those applying multivariable modeling to identify risk or prognostic factors.[43] To ensure the clinical suitability of our models, we also fulfilled clinician checklist for assessing suitability of machine learning applications in healthcare, but only the items that are applicable to this stage of model development.[42] We also followed other guidelines to find comparable models to evaluate success criteria. The checklists for this study based on all of the guidelines and the comparable models are available (Table 1 to 5 in Supplementary Information).

We used R 4.0.2 programming language (R Foundation, VIE, Austria) to conduct most steps of the data analysis. For any steps related to DI-VNN, we also used Python 3.6.3 programming language (Anaconda Inc., TX, United States). The integrated development environment software was RStudio 1.3.959 (RStudio PBC, MA, United States). To ensure reproducibility, we used Bioconductor 3.11;[61] thus, versions of the included R packages were all in sync according to the versions in this Bioconductor version. For all models except DI-VNN, we used an R package of caret 6.0.86 that wraps R packages for the modeling algorithms, which were glmnet 4.1, Rborist 0.2.3, and gbm 2.1.8. For DI-VNN, we used keras 2.3.0 and tensorflow 2.0.0 Python libraries via R packages of reticulate 1.16, keras 2.3.0.0, and tensorflow 2.0.0. We also create R packages for many steps in the data analysis, including DI-VNN, which are medhist 0.1.0, gmethods 0.1.0, rsdr 0.1.0, clixo 0.1.1, and divnn 0.1.3 (both an R package and Python library). All of these packages/libraries are available to download from this repository https://github.com/herdiantrisufriyana. For model deployment, we used Shiny Server 1.4.16.958 and Node.js 12.20.0. Details on other R package versions and all of the source codes (vignette) for the data analysis are available (Table 6 in Supplementary Information).

To reproduce our work, a set of hardware requirements may be needed. We used a single machine for all models, except DI-VNN, with 16 logical processors for central processing unit (CPU) 2.10 GHz (Intel® Xeon® E5-2620 v4, CA, United States), 128 GB RAM, and 11 GB graphics processing unit (GPU) memory (NVIDIA GeForce GTX 1080 Ti, CA, United States). Parallelism was applied for the CPU computing. Meanwhile, DI-VNN required a higher GPU capability than that provided by the previous specification. For hyperparameter tuning and training, we used multiple machines on cloud with 90 GB RAM and 32 GB GPU memory (NVIDIA Tesla V100-SXM2, CA, United States). For prediction, DI-VNN only needs CPU in a local machine, or that in a cloud machine for the web application.

Study design

We applied a retrospective design for selecting subjects from a nationwide health insurance dataset provided by the *badan penyelenggara jaminan sosial (BPJS) kesehatan* which is a government-owned health insurance company in Indonesia. This dataset was the second version published on August 2019 and originally a cross-sectional sampling of ~1% (*n*=1,697,452) of insurance holders from all the affiliated healthcare providers (*n*=22,024; primary, secondary, and tertiary care) nationwide for any visits admitted on 2015 and 2016. The dataset was opened publicly by request. The BPJS Kesehatan had already approved our request (dataset request approval no.: 5064/I.2/0421). The dataset had been already deidentified before going public; thus, the ethical clearance to Institutional Review Board of Taipei Medical University was waived. Details of sampling procedures of the data source are described (Supplementary Information). Population of this country covered races of Asian and Austronesian. The health insurance covered 200,259,147 (75.8%) individuals in that country. Therefore, our retrospective design may reflect a prevalence in population of the nation, including that of an outcome in pregnant women, predicted by our model.

We included the health insurance holders of 12-to-55-years-old females who had ever visited the affiliated healthcare providers within the dataset period. Pregnancy might be the first or the second within the dataset period. We excluded visits after delivery, and of course, we did not have visits beyond the dataset period. However, for a person who was pregnant twice within the period, we labeled the same person as a different subject with each pregnancy period while retaining the medical history before or during the first pregnancy for the secondly-labeled person. A complete list of the codes for determining delivery or immediately after delivery care is available (Table 7 in Supplementary Information).

There were two tasks that we developed prediction models for. The first was to classify if a visit is held by a subject that will end the pregnancy period with PROM. The second was to estimate the time of delivery. Outcome for classification task was defined as an event for a subject that had been encountered O42 code within a pregnancy period, which is a code for PROM based on ICD-10. Otherwise, a subject was assigned as a nonevent if the pregnancy ends within the dataset period using the same codes for determining the end of pregnancy period. The selected females might or might not be pregnant while those who were pregnant might or might not deliver yet up to the end of the dataset period. For those who were negatives for either pregnancy or delivery, we assigned censoring labels. We took into account of the subjects with censored outcome for causal inference and weighting the uncensored outcome (events vs. nonevents) over both censored and uncensored outcomes when training models. But, we did not include visits with censored outcome in any sets for predictive modeling. This was because we need to preserve similar distribution of any outcomes with those in the target population when training our model and solving the class imbalance problem by inverse probability weighting.

Meanwhile, outcome for estimation task was a number of days from a code encounter within a pregnancy period to current visit when the prediction model application is used. That code was O42 for events, while nonevents were assigned with the same codes for determining the end of pregnancy period. As classification task, the visits with censored outcome were not included into any sets, but as causal inference task, those were included to preserve outcome distribution similar with that in the target population. While 'regression' term is more common to use for this kind of task, we used 'estimation' term to avoid potential confusion with regression algorithm used for fitting the model parameters. In addition, unlike other time-varying outcome, e.g. cancer, we did not predict survival rate for the estimation task because a time interval for a pregnancy period is definitely known. It is also more intuitive for clinicians if the given information is the estimated days from current visit to the day of a pregnant woman will deliver a baby, such normal delivery estimations based on last menstrual period and ultrasound examination.

Practical costs of prediction errors are considered when evaluating models. Under-prognosis may cause pregnancy monitoring off-guard. A pregnant woman with preterm delivery may not reside in an area with readily available neonatal intensive care unit, particularly in low-resource setting. Over-prognosis may lead to unnecessary enrollment of patients into a cohort study on early intervention for preventing PROM or the complications, e.g. antibiotic administration. This may also lead to unnecessary test for more specific prognostication. Under-prognosis is likely more frequent since a model that requires simpler features tends to be used in low-resource setting. Therefore, related to under-prognosis, a well-calibrated model with lower false negatives or higher sensitivity should be given priority. In addition, estimation of the time of delivery should also need more precision to ensure patient safety. The estimation error causes the same situation with the false negative. The error is considerably around 2 to 4 weeks since this is a common interval between antenatal visits closer to the time of delivery.

Candidate predictors only consisted of medical histories. We avoided demographics as candidate predictors to reduce usage on private data and prevent social and economic discrimination by our models. We included demographics to provide data for causal inference only, which were age at admission (<20, 20–35, or >35

years old) and insurance class (first, second, or third class). We also avoided maternal age albeit this variable is often a strong predictor. But, this is also the reason why we avoid the variable, which is, machine learning models often memorize age non-linearly. In turn, a large weight is often assigned on maternal age followed by weight shrinking on other predictors. All of these demographic variables were binarized into 0 or 1 for respectively no or yes in each category of each variable. Meanwhile, we extracted medical histories from all of the ICD-10 codes for either diagnoses or procedures.

We have identified candidate predictors that took single value or zero variance. We removed all candidate predictors which are positive (value of 1) in only one of the outcome in training set. This is a perfect separation problem.[41] A predictor may be exclusively for one of the outcome by chance due to sampling error. To prevent such bias, we removed the perfect-separation candidate predictors. Details on the candidate predictors and the selection are described (Table 8 in Supplementary Information). Eventually, we removed mother or baby diagnosis/procedure codes that demonstrate delivery or immediately after delivery care (typically up to 6 weeks following childbirth), including a code for the events and other conditions that typically happen only during the delivery or post-delivery period. Otherwise, these codes would unexpectedly leak the outcome information. A complete list of the excluded codes is available (Table 9 in Supplementary Information). In addition, we also identified redundant candidate predictors by pair-wise Pearson correlation coefficients (Table 10 in Supplementary Information). None of the coefficients showed perfect correlation (r=1). High correlation (more than ~0.70) were reasonably identified between causal factors and the code components. We did not remove those pairs of predictors.

We have identified candidate predictors with non-zero variances, irredundant candidate predictors, and ones that did not show perfect separation problems, which were 372 candidate predictors. These included 9 candidate predictors for causal RR model. We did not include low social-economic status and maternal age for reducing private data usage and preventing social-economic discrimination. The final model also used the same number of predictors because of the nature of RR algorithm. We transformed 372 candidate predictors into the same number of PCs as candidate predictors for the PC-ENR model. The PCs were ordered by the weights in PC-ENR. Then, we picked 60 PCs with the highest absolute, non-zero weights as candidate predictors for PC-RF and PC-GBM models in order to achieve 200 events per variable (EPV), as recommended by prediction model risk of bias assessment tools (PROBAST) guidelines.[43] For DI-VNN, there were 144 candidate predictors after differential analyses of 372 candidate predictors with multiple testing corrections. Weights, variable importance, and intermediate outputs, that indicated to what extent a predictor contributes on the prediction, are shown respectively for: (1) causal RR and PC-ENR (Table 14 to 16 in Supplementary Information); (2) PC-RF and PC-GBM (Table 15, 17, and 18 in Supplementary Information); and (3) DI-VNN (Table 19 and 20 in Supplementary Information).

Quantifying medical history by Kaplan-Meier estimator

Recorded as a diagnosis/procedure code, a condition affects a health state in the future in relative to the time interval between that condition and the health state at the time of prediction. The effect may be improving, devastating, or undifferentiated, but there should be a quantity that is differential through time; thus, a computer is able to associate such trend of the condition with the health state among individuals. That quantity should also differ among conditions given the same time interval to the time of prediction, and be inferred from population as a generalized quantity for each individual given the time interval. This is the intuition behind application of the Kaplan-Meier (KM) estimator in this study.

$$\widehat{\boldsymbol{H}}(\boldsymbol{d}) = \prod_{\boldsymbol{i}=\boldsymbol{1}}^{\boldsymbol{d}} \left(\boldsymbol{1} - \frac{\boldsymbol{e_i}}{\boldsymbol{v_i}}\right) \qquad (\textbf{Equation } 1)$$

For medical history, we computed a number of days for a code in the latest encounter before current visit (the time of prediction or d_0). A KM estimate was calculated for each code.[45] The KM estimate (Equation 1), as we denote it as an estimator of the historical function $H(d)$, is a probability or fraction of visits that a condition is longer than d days before d_0, with d_i as a day when at least an encounter happening for a code that refers to that condition, e_i as the number of encounters for the code at day d_i, and v_i as the visit recorded not to have the code encountered (been censored) up to day d_i. A KM estimate of a code for an individual is then determined given the day number from a code encounter to current visit. Because there might be a day on which no encounter for a code in the population, we applied linear interpolation between time points on which a KM estimate is able to calculate. We used only training set to infer these estimates at population level; thus, our models were blind to the distribution of the KM estimates in any external validation sets. We made medhist 0.1.0, an R package, that allows future investigators to implement this historical rate.

Human learning: Knowledge-based causal diagram and statistical learning

We applied a systematic human learning by literature mining from September 30th to October 2nd, 2020. This would draw our assumption on PROM causality. For simplicity and avoiding redundant records, we only used PubMed because it is the most frequently updated (daily), the longest period coverage (1950–present), and life science-focus literature database.[62] This database also allows usage of a specific term in Medical Subject Headings (MeSH) vocabulary thesaurus from National Library of Medicine, National Institutes of Health, United States. We adapted snowball sampling method by starting from convenient sampling,[46,48] to get similar sense with human intuition when learning through literatures.

Algorithm 1. Snowball sampling modified by starting from convenient sampling to get initial document (d_0)

Require: d_0
$A = \emptyset$
$L = \emptyset$
$k_0 = read(d_0)$
while $k_s \Delta A \neq \emptyset$ **do**
    $a_s \leftarrow k_s$
    $d_s = search(k_s)$
    if $d_s \neq \emptyset$ **then**
        $k_{s+1} = read(d_s)$
        if $causal(k_{s+1})$ **then**
            pass
        else
            $l_s \leftarrow k_{s+1}$
            end while
    else
        end while

First, we looked for a document, denoted as d_0 (Algorithm 1), using a keyword of '"Fetal Membranes, Premature Rupture"[Mesh]' from an authoritative institution as the convenient sampling step. This led to a Practice Bulletin No. 172 from American College of Obstetrics and Gynecology (ACOG).[35] Then, we denoted causal factors of PROM as A, while the confounders are denoted as L. Confounders are causal factors of a causal factor of PROM. This means L are the same factors that cause both A and PROM. Initially, there is no A or L yet. By reading an article/document d_0, identify $a \in A$ to determine k_0 keyword that refers to a at $s = 0$ stage. The next steps are iterative until no k_s keyword refers to any $a \in A$. Assign k_s to a_s and search for d_s document using k_s for causal factors of a_s. If a document is found, then continue; otherwise, end iteration. Continue by reading d_s to determine k_{s+1} keyword. It refers to a causal factor of $a \in A$ that is referred by the previous k_s keyword. Then, search and read documents to check if k_{s+1} keyword also refers to causal factors of PROM. If yes, then pass k_{s+1} keyword to $s + 1$ stage; otherwise, assign k_{s+1} to l_s then end iteration. In addition, we only considered pregnant women as population those studies investigated on.

Factors of *A* and *L* are called as first- or second-level factors of PROM while only the first-level are the causal factors. This would determine the position of factors within a circular network depicting a causal diagram we used for causal inference. Since the first-level factor may come from the second-level factors in the process, we may also find inter-causal factor relationships. We also included these relationships as edges in the network because these are needed to construct the causal inference formulas. However, only first-level factors were included to the formula. For example, both asthma and influenza were the first-level factors of PROM, while varicella was the second-level factor of PROM via asthma. To determine formula for causal inference of asthma, we included only asthma and influenza. We took only asthma significance to determine if asthma is a causal factor of PROM. Therefore, only the causal factor of interest and the confounding factors or common causes were included in the causal formula. We avoided common effects to be included to prevent collider-stratification bias, or unnecessary inclusion of second-level factors[47].

To verify our assumption of PROM causality, we applied one of generalized (G) methods, which is IPW, for each causal factor.[63,64] This method is design for time-varying exposure.[65,66] However, we also conducted outcome regression for the causal inference since this method is one of the methods that are more common in use although do not work in general.[47,67] Another common method is propensity-score matching with various versions, but, we did not apply this method for simplicity. While adjusting all confounding factors are difficult, if not impossible, we would disclose opened backdoors (confounding factors that were not blocked) because of limitation on providing data for each causal factor (Figure 1 to 12 and Table 11 in Supplementary Information). This would help interpreting the results of this study with caution.[68,69] We also used only training set for causal inference. We represented demographics and medical histories into candidate causal factors if applicable. These were binarized into 0 and 1 respectively for negative and positive. Details on either ICD-10 codes or demographical variables we assigned for each causal factors are available (Table 12 in Supplementary Information).

After verifying the causal factors, we only included these into a prediction model applying logistic regression with a shrinkage method, as recommended by PROBAST, instead of that using a stepwise selection method.[43] We chose ridge regression, which apply L_2-norm or beta regularization, because this method retains all causal factors within the model after weights updated by training.[70] We understood that this model would not necessarily be the best model because of the use of causal factors. Predictive modeling may normally exploit confounders to achieve better performance, while causal model may not explain all variations among individuals, that are contributed by confounding factors.[47] However, by comparing a predictive model to one that uses only causal factors, we can imagine how much confounder effects are exploited for improving predictive performance by machine learning algorithms. This, in turn, warns a human user of machine learning algorithm when conducting a critical appraisal to internal properties of a machine learning model. We followed the same procedures for hyperparameter tuning and parameter fitting (training) with those for machine learning, as described in following section. Nonetheless, we viewed tuning and training of this prediction model is already a part of machine learning since less intervention is given by human user.

Machine learning: State-of-the-art algorithms and deep-insight visible neural network

To do iterative tasks with enormous amount of data, human needs machine. We used medical histories and causal factors as the candidate features/predictors. The medical histories were provider-wise by estimation. This means our prediction models would only use medical histories recorded in a healthcare provider, blind to those recorded in the others. This reflects most real-world situations in which a healthcare provider does not have an access to medical records in the other providers. However, we already have a nationwide historical (KM) rates for each code, derived from training set only. All medical history in days were transformed into the historical rates. Therefore, this technique allows generalization of individual data based on a nationwide, population-level data, without a need to access data from the other providers.

Five modeling approaches were applied for supervised machine learning, which were a set of procedures covering feature representation, feature selection, hyperparameter tuning, and training strategy. For all models, we applied a grid search method for hyperparameter tuning of minimum 10 alternatives in each modeling approach. The best hyperparameters or tuning parameters were used for training the model or fitting the parameters. For classification task, outcome (Y) was weighted by half of inverse probability/prevalence (w_i), including the censored outcome ($\emptyset$). For example, if the prevalence of $Y = 1$ is 0.2, then w_i is $1 \div 0.2 \times 0.5$ for outcome of $Y = 1$. The sum of the three probabilities is equal to 1. The weight formula is shown (Equation 2). These weights were plugged in a general equation for loss function in this study (Equation 3), in which training was generally conducted to estimate parameter θ_j in a model $f(x_{ij}, \theta_j)$ that minimize L, while n is a number of visits, p is a number of candidate predictors, x_{ij} is a value for each ith instance and j^{th} candidate predictor, and regularization factors of α and λ. Meanwhile, no weighting would be applied for the estimation task ($w = 1$ for any i). In addition, a specific training strategy was applied for the last modeling approach, which was DI-VNN, a model using the pipeline that we developed in this study.

$$w_i = \frac{1}{Pr\left(Y \begin{cases} 0 \\ 1 \\ \emptyset \end{cases}\right)} \times 0.5 \qquad (\textbf{Equation } 2)$$

$$L = \sum_{i=1}^{n} w_i \left(y_i - f(x_{ij}, \theta_j)\right)^2 + \alpha \sum_{j=0}^{p} \|\theta_j\| + \lambda \sum_{j=0}^{p} \|\theta_j\|^2 \quad (\textbf{Equation } 3)$$

The first model was causal RR. We used all causal factors except ones that included demographics. This model applied a filter method for feature selection by verifying assumption based on domain knowledge with a statistical test for causal inference, as described in the previous section. Ridge regression was applied as the parameter fitting algorithm. Tuning parameter was $\lambda = \{10^{-9}, 10^{-8}, \cdots, 10^{0}\}$ while keeping $\alpha = 0$. These values were plugged into the loss function (Equation 3).

From 2nd to 4th models, we represented medical histories and causal factors into principal components by k-fold cross validation. First, an X matrix was constructed with $n \times p$ dimensions for $i = [1,2,\cdots,n]$ instances and $j = [1,2,\cdots,p]$ candidate predictors. Each vector was standardized with column-wise μ_j mean and σ_j standard deviation of all instances for each candidate predictor. Then, we mapped each vector $x_{(i)} \in X$ onto a new vector of PC scores $t_{l(i)} = x_{(i)} \cdot \beta_l$ for $l = [1,2,\cdots,q]$ by a matrix β of weight vectors where q up to p. The mapping was finding the estimates of weight vectors that maximize variance (Equation 4). The lth PC was calculated by subtracting the $l^{th} - 1$ PC from X, then finding the estimates l^{th} PC as such $l^{th} - 1$ PC. We made the rsdr 0.1.0 (an R package) that allows future investigators to conduct principal component analysis or singular value decomposition by resampling methods, as described in this study. Instead of singular values by bootstrapping, we computed PCs by k-fold cross validation for simplicity reason considering on a simpler theoretical framework and an achievable computational capacity. To compute PCs by k-fold cross validation, each of β_l, μ_j, and σ_j were inferred from training set only of which a $(K - k_m)/K$ part of n instances for $m = [1,2,\cdots,K]$ is used to compute a variance each time. An estimate of a weight vector β_l is calculated by averaging β_l from all $K = 10$ of $(K - k_m)/K$ parts. The eigenvalue of the matrix is commonly known for $X^T X$ that achieves maximum variance by β as the eigenvector.

$$\begin{aligned} \hat{\beta}_l &= \frac{1}{K} \sum\nolimits_{i \in k^{th}part} \arg\max_{\|\beta\|=1} \left\{ \sum_{i=1}^{n} \left(x_{(i)} \cdot \beta_l\right) \right\}^{-k} \\ &= \frac{1}{K} \sum\nolimits_{i \in k^{th}part} \arg\max_{\|\beta\|=1} \left\{ \sum_{i=1}^{n} \left(\beta^T X^T X \beta\right) \right\}^{-k} \qquad (\textbf{Equation } 4) \end{aligned}$$

The second model was PC-ENR (elastic net regression). We used all PCs for the second model since the EPV of training set was >20, of which a variable is a PC. This number, instead of 10, is recommended when applying a regression algorithm.[43] This model also applied a shrinkage method for feature selection. Tuning parameters were combinations of $\alpha = [0, 0.25, \cdots, 1]$ and $\lambda = [10^{-9} + g\,(10^{0} - 10^{-9})/G, \cdots, 10^{0}]$ for $g = [1, 2, \cdots, G]$ and $G = 5$; thus, the best tuning parameters were search over 5×5 alternatives. These values were plugged into the loss function (Equation 3), where $\alpha = 0$ means this model becoming a ridge regression but using PCs instead of the original candidate predictors. But if $\alpha > 0$, some candidate predictors $x_{(j)}$ might be zero after being multiplied with θ_j that minimize the training loss. Each time, this applied different α and λ to find a couple of values that minimize the validation loss.

The 3rd and 4th models respectively applied PC-RF (random forest) and PC-GBM (gradient boosting machine). These models also applied wrapper method for feature selection. This means the candidate predictors were pre-selected by a model before being used for these models. The wrapper model was the PC-ENR. We expected the smaller number of PCs were pre-selected; thus, the EPV of training set is 200 or more. This is allowed by either using wrapper method only or additionally applying filter method afterward. The filter method was conducted by ranking the PCs based on the corresponding θ_j in descending order. We selected l of PCs where l is a number such that the EPV is 200. Unlike regression algorithm, modern machine algorithms are data hungry, of which the tested algorithms are classification and regression tree (CART), random forest, support vector machine, and shallow neural network.[71]

RF and GBM are ensemble algorithms. This means both used prediction results of multiple models that apply the same algorithm, which is CART. However, RF ensembles the models in parallel while GBM sequentially ensembles the models.[72,73] Parallel ensemble means the prediction is respectively the majority and the average of CART predictions for classification and estimation tasks. Meanwhile, sequential ensemble means a simple CART is made to predict classification or estimation error of an earlier CART model. Tuning parameters for random forest were combinations from a number of random candidate predictors to build a CART each time $[5 + g(45 - 5)/G, \cdots, 45]$ and a number of minimum unique instances per node $[20 + g(100 - 20)/G, \cdots, 100]$ for $G = 5$, while 500 CARTs were built for each of the candidate models. For GBM, we set tuning parameters as combinations from a number of CARTs $[100, 200, \cdots, 2500]$ and a shrinkage factor or L_2-norm regularizer $[0.0005 + g\,(0.05 - 0.0005)/G, \cdots, 0.05]$ for $G = 25$, while minimum samples per node (tree) is 20 and only 1 random predictor was used to build a CART each time. Since exhaustively comparing all possible combinations are time consuming, if not impossible, all the numbers for both ensemble models were determined based on common practices for simplicity. However, this considered a sample size, a number of candidate predictors, and diversity of approaches between two algorithms to heuristically optimize the hypothesis search. The prediction results of these models were plugged in into the loss function (Equation 3) with $\alpha = 0$ and $\lambda = 0$ to compute the errors that are minimized by the training.

The last model was DI-VNN. We left the technical details for this model in Supplementary Information, since it is relatively more complex than the other models. We developed this model as a human-machine interface for a prediction task. The machine does what it is currently good at, which is handling large and complex data, and we organized multiple methods in an analysis pipeline such that the complexity can be understood by the human user chunk-by-chunk, although it does not mean easy to understand. Meanwhile, the human does thing that is better than the machine up to now, which is reasoning by extra data knowledge. The following description focuses on methodological reasoning that is critical for interpreting the results of DI-VNN by a human user. A 5-minute video is also available for explaining DI-VNN pipeline that focuses on the technical details (watch Supplemental Video in Supplementary Information).

Instead of PCs, we used all of the original candidate predictors. All of the subsequent procedures were conducted using training set only. First, we conducted feature selection using filter method by differential analysis. This analysis is commonly used in genomics, including microarray, RNAseq, and microbiome analysis.[44] In such field, the analysis deals with high dimensional data with extremely lower sample size in relative to those of other biomedical fields. For any candidate predictors, the values from a sample were made to be similar to other samples by quantile-to-quantile. Moderated-*t* statistics was applied, that is, fitting a univariable logistic regression for each pair of a candidate predictor and an outcome. Only a candidate predictor with a significant differential effect were filtered into the downstream analysis. Differential effect means a selected predictor does not necessarily have the expected effect; however, the effect is more than the unselected one. Since the selection involving multiple regressions that are exposed to multiple testing effects, a procedure was applied for each of the significances by Benjamini-Hochberg correction. This penalizes *p*-values along with the increasing number of testing. A false discovery rate (FDR) or an adjusted *p*-value of 0.05 is commonly chosen as the lowest value of a significance for excluding a candidate predictor. Using the average value of each predictor, we transformed the normalized values into 0 if exactly the same with the average, 1 if greater than the average, and -1 otherwise. This is a 1-bit stochastic gradient descent transformation method which simplifies computation thus allowing to train deeper neural network.[74] The biological relevance for this transformation is the same with undifferentiated, up- and down-regulation. Under high dimensional data perspective, there is a chance that an increasing/decreasing factor is followed by the change of other factors to maintain values of particular ones, such that physiological homeostasis is maintained.

Using the unnormalized values, we standardized those values by feature-wise average and standard deviation using only training set. A feature-to-feature Pearson correlation matrix was computed. Using this matrix, we applied respectively dimensional reduction and hierarchical clustering algorithms to create a feature map and a network architecture. The algorithms were *t*-SNE using Barnes-Hut approximation[39] and CliXO.[38] Motivation behind application of these algorithms is to achieve a moderate but acceptable predictive performance while allowing a human user to explore interactions among predictors in the context of predicting the outcome. Therefore, this model is expected to enable exploratory analysis along with external contextual knowledge developed by human learning. To apply Barnes-Hut *t*-SNE, an R package of Rtsne 0.15 is available to download from Bioconductor. Since CliXO is implemented by C++ programming language, we developed an R package of clixo 0.1.1 for simpler implementation of CliXO by R users.

Application of a dimensional reduction algorithm to create a feature map is a method called as DeepInsight by the inventor.[33] The feature map is a multidimensional array (typically two dimensions) which is the same input for a CNN. It is a deep learning algorithm to extract important features of an image, e.g. x-ray imaging. These features are feed-forward to a fully-connected neural network model to predict an outcome. CNN is the most successful deep learning algorithm within recent years.[75] Since it is developed for image data, its application on non-image data is limited, conventionally. DeepInsight allows non-image data being fed to the CNN model; therefore, we may expect a state-of-the-art predictive performance from the constructed model. Although other methods were also available for the same purpose, a feature map of DeepInsight is data-driven. Dimensional reduction principally works to map features on higher dimensional to lower dimensional space but minimizing information loss. This means a two dimensional output is optimized to represent many features/predictors and to infer inter-predictor distances. In real-world data, bivariate interaction may or may not happen only between a predictor and an outcome, but inter-predictor interactions are also other dynamics that may indirectly affect an outcome. Furthermore, DI-VNN applied *t*-SNE in DeepInsight based on the rank or order instead of the values of *t*-SNE dimensions. This approach greatly reduces the feature map size; thus, a lighter size of array can be achieved for training a DI-VNN model. In addition to the original values of predictors that reside on the feature map, the distances between all possible pairs of predictors enlarge hypothesis space for a CNN algorithm to fit representative weights to as many as variations available to predict an outcome.

Unlike DeepInsight, application of a hierarchical clustering algorithm to create network architecture is motivated to

improve interpretability by introducing transparency of internal properties of a neural network. Originally called as visible neural network by the inventor,[34] this method constructs a data-driven neural network architecture. Specifically, this method applied CliXO algorithm which can be considered as agglomerative clustering similar to complete- or single-linkage hierarchical clustering. Unlike many hierarchical clustering, CliXO algorithm respects biological pleiotropy, which is, an entity may belong to one or more entities. For example, in genomics, a child term of gene ontology (GO) for biological process is a part of >1 parent terms. CliXO precisely mimics the human-curated GO.[38] By ontology alignment algorithm, the CliXO may or may not aligned to existing GO, but, one that is not aligned may be a novel ontology term. While no ontology database is known for medical histories for any diseases, we may expect semantical grouping among medical histories. This means we expect those within the same group are also ones that has similar conditions in a particular circumstance.

The original visible neural network only uses vanilla neural network instead of CNN. This neural network model may not represent all variations within the data as we may expect from CNN. Meanwhile, DeepInsight allows application of CNN. Therefore, we developed this pipeline that organizes DeepInsight and visible neural network into a DI-VNN model. Unlike sequential CNN, DI-VNN isolates backpropagation following the hierarchical structure by CliXO. Backpropagation is a sequential computation of loss function from the surface layer, which is closer to the outcome (depicted on top), to the deeper layer, which closer to the predictors (depicted on the bottom or leaf nodes). This means an error on prediction is corrected iteratively to the lowest possible given the maximum iterations and the smallest improvement to pursue, chosen by a human user. In DI-VNN, the correction is made by considering similarity (represented as a distance) among predictors layer-by-layer from the surface (more common ontology term) to the deeper layer (more specific ontology term). In origin, a visible neural network maps genotype in the deeper layer to phenotype in the surface layer by ontotypes which are ontology terms convoluted from deep to surface. Thus, we expect to find a specific insight in a deeper layer in the context of predicting an outcome.

We applied several best practices on CNN model development for DI-VNN. First, we applied Inception v4-Resnet that enables faster and lighter computation;[76] thus, we do not need to simplify the network architecture by CliXO to get acceptable speed for training. Unlike regression algorithms, a neural network starts with random weights. If the randomization is not appropriate, the updated weights or parameters can be too high or too low preventing the optimal combination of weights to achieve within the iterations. We apply He and Glorot methods for initial randomization respectively in the hidden layer (between the deepest and the most surficial layers) and the output (the most surficial) layer.[77,78] We also applied warm-up strategy to train DI-VNN.[79] This allows a larger batch size which a number of samples is used for updating parameters on each step of iteration until all training samples are used in each epoch. It is a term for a complete cycle to update parameters using all the samples. Commonly, a smaller batch size maximizes predictive performance with longer training time as its price; thus, training the model using the warm-up strategy requires shorter time without sacrificing the predictive performance. Larger batch size also demands for lower learning rate. Moreover, learning rate is reduced across epochs if no improvement is made for the predictive performance after an epoch compared to that of the previous one.[76] Lower learning rate makes parameter updating subtler; thus, the parameters do not jump too much pursuing the optimal values allowing the predictive performance is improved further compared to that without decreasing learning rate. All of these empirical approaches were chosen to allow realistic computation in term of time and capacity for most architecture constructed by CliXO and to achieve acceptable predictive performance.

The prediction result of this model was also plugged in into the loss function (Equation 3) with $\alpha = 0$ and $\lambda = \{10^{-9}, 10^{-8}, \cdots, 10^{0}\}$ to compute the errors that are minimized by the training. But, we computed errors for each ontology term or node, not only the root node that we used for prediction. The errors of non-root nodes t are weighted as much as $\gamma = 0.3$ of that of root node r (Equation 5), as previously used in GoogLeNet[80] and the initial model of visible neural network[34]. All losses were normalized within 0 to 1.

$$L = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{L_i^{(r)} + \gamma \sum_{t \neq r} L_i^{(t)}}{r + \gamma \times t}\right) \qquad (\textbf{Equation } 5)$$

Evaluation metrics

We needed to evaluate our models for both classification and estimation tasks. All evaluation metrics were expressed as interval estimates by 95% confidence. An interval estimate was calculated from the metrics of multiple resampling subsets for a model validation.

Calibration and discrimination measures were calculated for classification task. Calibration measures were evaluated by fitting predicted probabilities to true probabilities using a univariable linear regression; thus, the measures were calibration intercept and slope. We also demonstrated a calibration plot. A model is well-calibrated if the interval of calibration intercept and slope respectively fall closer onto 0 and 1, and the calibration plot approximately hugs the reference line. Meanwhile, a discriminative ability of a model was quantified by an AUROC interval estimate. A non-overlapping, higher interval of AUROC determined the best model among the most calibrated ones.

For estimation, we applied RMSE to train the models. However, since longer time interval is reasonably more difficult to predict, a single evaluation by RMSE may not be sufficient to evaluate the estimation performance. We also evaluate the estimation plot between the predicted and the true time of delivery, binned per week. Different lines in the estimation plot were provided for positive and negative prediction by the best classification model. This would be reasonably applicable for clinical application by giving an interval estimate of days for the true time of delivery conditional on the predicted one. In addition, we also limited this evaluation for the estimated time of 42 weeks or less, which is a maximum duration of pregnancy.

To determine the best estimation model, we computed a proportion of weeks in which each predicted time, as a week, is included within an interval estimate of the true one. The interval should be maximum $\pm\, x$ weeks when predicting $> x$ weeks. For example, if a model predict a woman will deliver in 6 weeks, then the interval estimate of the true time of delivery should be maximum ± 6 weeks. For any women predicted to deliver in 6 weeks, this number should fall onto that interval. For the best estimation model, we would determine the minimum and maximum predicted time of delivery with acceptable precision for each predicted outcome of PROM based on visual assessment using internal validation. We also computed the RMSE within this acceptable range.

Success criteria of the modeling were AUROC greater than those of recent models (last 5 years) of PROM prognostic prediction using predictors in low-resource setting (e.g. maternal factors), or greater or equal to those using predictors in high-resource settings (e.g. biophysical or biochemical markers). To prevent a common cause of overfitting, we applied PROBAST criterion on EPV for all models. This means the number of events in training set should be 20 or more after divided by the number of candidate predictors.

Previous prognostic models were developed for preterm PROM. Meanwhile, we also did not have data for the gestational age. These made us to develop both PROM prognostic prediction and estimation of the time of delivery by days from the time of prediction. For simplicity, we did not compare our estimation model with those for estimating the time of delivery. Instead, we compared our PROM prediction model with the previous models which were developed for preterm PROM.

We applied PRISMA 2020 guidelines to find comparable models to evaluate the success criteria. Since systematic review and meta-analysis are not the main purposes of this study, we only applied all items in methods section of the guidelines, except item number 11 and 14 regarding risk of bias assessment and item number 13 and 15 regarding synthesis method and certainty. This is because we applied the guidelines only for facilitating a fair comparison to previous studies, not taking a conclusion of how valid

and accurate PROM is able to predict. To achieve that purpose, following PICOTS framework, the eligibility criteria were: (1) Population, either pregnant or non-pregnant women without specializing the medical conditions; (2) Index, the best prediction model in this study; (3) Comparator, prediction models or rules; (4) Outcome, either preterm or term PROM as a binary outcome (event or nonevent) with 20 EPV or more; (5) Time, prognostic prediction from days to weeks; and (6) Setting, either primary care or hospital patient. We excluded any article types beyond original article, including conference abstract but not the full paper. The studies were not grouped since no synthesis was conducted.

We searched the studies up to April 3rd 2021. Unlike the human learning that only used PubMed, we also included Scopus and Web of Science as the literature/bibliographic databases in which we searched for the previous models, because the prediction studies were also extensively reported beyond life-science journals (i.e. computer-science journals). The keyword was 'prelabor rupture of membranes prediction'. We limited the records within the last 5 years and using English. We applied similar, but not exactly the same, search strategy for the three databases considering different interfaces (Table 4 in Supplementary Information). HS searched for the literatures and loosely filtered these by title and abstract. Then, HS manually assessed the eligibility by the full texts and identified the ambiguous ones. Independently, HS and YWW assessed the ambiguous studies. If no consensus made between HS and YWW, final decision was made by ECYS. HS collected data from each full text. We only extracted a sensitivity and a specificity of the best model with the most similar outcome definition from each study to that of our study. The AUROCs were also included. If not reported, the AUROC was inferred by trapezoidal rule utilizing the sensitivity (Sn.) and specificity (Sp.) (Equation 6). We also extracted the study design, population, setting, outcome definition, sample size, including details on events and nonevents, number of candidate predictors, EPV, predictors in the best final model, and the most recommended validation techniques (external over internal validation; bootstrapping over cross validation, or cross validation over test split). The plots of sensitivities and specificities would overlay those of our models for PROM prediction. We also plotted a point at a sensitivity and a specificity for each of our models at the optimum threshold, on each ROC curve. AUROCs were compared with the models in this study (with or without 95% CI).

$$AUROC = \frac{Sn \times (1 - Sp)}{2} + Sn \times Sp + \frac{(1 - Sn) \times Sp}{2} \qquad (\textbf{Equation } 6)$$

We also evaluated the best time window using the best model for PROM prognostic prediction. This will give an additional insight when interpreting the best prediction model. Using internal validation set, we grouped samples by binning the days to the end of pregnancy for every 4 weeks. An interval estimate of AUROCs was computed for each bin from all resampling subsets. By observing the plot, the best time window should mostly cover the interval estimates that are higher than an AUROC of 0.5 representing prediction by simply guessing.

For population-level exploration using DI-VNN, we extracted arrays of intermediate outputs for all ontology terms, including the root one. The intermediate output was an array after being filtered by each block of Inception v4-Resnet but before being convoluted into a single number as a predicted probability inferred from a neural network up to each ontology term. The intermediate array had the same dimensions with the input array. An array was extracted for each instance and ontology term. We computed element-wise average values for each ontology array across all instances with the same outcome, which might be event or nonevent. For estimation task, we computed those values per outcome, as if it is event or nonevent, respectively as: (1) less than or equal to the average time of delivery; or (2) greater than the average time of delivery. Then we computed element-wise subtraction of the average array of event with that of nonevent for each ontology term. Since a feature member of an ontology term resided on a particular position in an array, the subtracted array showed values that might surround that at the feature member. This value might be positive or negative. However, positive and negative outputs are not straightforward respectively to event and nonevent. Yet, these tend to contribute on the opposite outcomes. Interpretation should apply extra data knowledge using the results of the causal inferences.

Model calibration

A model can be calibrated by training an additional model to estimate true probability of PROM given the predicted probability by the model. We applied GAM-LOESS. Instead of using the pre-calibrated models, the final comparison was made for the models calibrated using the predicted probabilities from each of the developed models. This calibration was intended to get more linear predicted probability. No calibration was conducted for the estimation models.

Model validation

We split the dataset into two subsets which were intended for internal and external validation. For all external validation sets, evaluation metrics were resampled by bootstrapping for 30 times to get interval estimates. As recommended, we split out a dataset for external validation by geographical, temporal, and geotemporal splitting. Geographical splitting was conducted by stratified random sampling of cities as many as p proportion of cities in each state. Temporal splitting was also conducted by stratified random sampling of days as many as p proportion of days in each subtropical season. Visits of subjects from either the sampled cities or days were respectively excluded as either a geographical or temporal subset. Then, we furthermore excluded as many as p proportion of cities in the geographical subset or that of days in the temporal subset. Visits of subjects from both of the excluded subsets were overlapped as a new geotemporal subset. Either geographical or geotemporal subset was subtracted by that geotemporal subset. We randomly tried different p for geographical, temporal, and geotemporal splitting such that approximately ~20% visits belonging to any of three subsets. These external validation sets would be stress tests for our models since the distributions of predictors and outcome are conceivably uncertain among the excluded cities, days, or the overlaps. This reflects situations in some real-world settings but this may not reflect common situations nationwide.

To estimate the predictive performance nationwide, the remaining set was randomly split out ~20% thus leaving ~64% of the original sample size for training set. This random subset was more representative for external validation to estimate predictive performances of our models nationwide. After splitting out this random subset, the remaining samples was intended for internal validation.

We split out ~20% of the internal validation for calibrating each model. The final predictive performance for internal validation came from this calibration subset. We resampled it by bootstrapping for 30 times to compute the interval estimates. To compute EPV, we only used the ~80%, which is, the pre-calibrated subset. This was used for training the models. For hyperparameter tuning, we applied 5-fold cross validation, while the final training was conducted using the best tuning parameters by bootstrapping for 30 times. Both tuning and training used the same pre-calibrated set applying cross validation for all models except DI-VNN. Since this model is computationally expensive, we trained the model using ~80% of the pre-calibrated set and validated the predictive performance using the ~20% for each epoch. After all epochs, we used the ~20% to evaluate the predictive performance by bootstrapping for 30 times. The best tuning parameters were determined based on this evaluation. Like other models, DI-VNN also used the calibration set to evaluate the predictive performance.

Model deployment

We would deploy the best model for either classification or estimation task. Almost all models were quite complex to apply

without a software/application; thus, we provided a web application to use the best models using R Shiny. A user needs to upload a deidentified, two-column comma-separated value (CSV) file of admission dates and ICD-10 codes of medical history from a subject. A use case is described with an example dataset for immediate demonstration of how to use the web application.

A user can set a threshold to get classification performance with expected population-level performances of sensitivity, specificity, positive predictive value, and negative predictive value. A true time of delivery was also estimated based on subpopulation that had the same estimation of the time of delivery conditional on the same classification result. All population- or subpopulation-level metrics were expressed as interval estimates. The population data for classification task were those of the calibration split, while those for estimation task were the data of internal validation set which consisted of both pre-calibrated and calibrated subsets. In addition, a timeline was also shown to visualize the predicted time of delivery and the true time interval estimate at the subpopulation level. The timeline showed the time of prediction and those of code entries that were used as the features; thus, a medical history of a subject is visualized by this timeline.

For DI-VNN, we also visualized the ontology network and array as such for population-level exploration. But, obviously, we did not compute the average value for the array. The intermediate outputs were shown for an ontology array that is chosen by the user. Either positive or negative value at this individual level had the same meaning with that at population level. The predicted outcomes were also shown based on parts of the architecture up to each node (ontology). An AUROC was also shown for an ontology chosen by the user for the array. But, this metric was computed using the pre-calibrated DI-VNN only. This is because the calibrated DI-VNN used all parts of the architecture up to the root ontology array.

All results can be saved online and/or downloaded as a report. We showed an example of the report in this paper. In addition, we also measured inference durations for 10 times for a use case and reported the interval estimates.

References

33. Sharma, A., Vans, E., Shigemizu, D., Boroevich, K.A. & Tsunoda, T. DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture. *Sci Rep* **9**, 11399 (2019).
34. Ma, J., *et al.* Using deep learning to model the hierarchical structure and function of a cell. *Nat Methods* **15**, 290-298 (2018).
35. ACOG. Practice Bulletin No. 172: Premature Rupture of Membranes. *Obstet Gynecol* **128**, e165-177 (2016).
38. Kramer, M., Dutkowski, J., Yu, M., Bafna, V. & Ideker, T. Inferring gene ontologies from pairwise similarity data. *Bioinformatics* **30**, i34-42 (2014).
39. Maaten, L.V.D. Accelerating t-SNE using tree-based algorithms. *J. Mach. Learn. Res.* **15**, 3221–3245 (2014).
41. Luo, W., *et al.* Guidelines for Developing and Reporting Machine Learning Predictive Models in Biomedical Research: A Multidisciplinary View. *J Med Internet Res* **18**, e323 (2016).
42. Scott, I., Carter, S. & Coiera, E. Clinician checklist for assessing suitability of machine learning applications in healthcare. *BMJ Health Care Inform* **28**(2021).
43. Moons, K.G.M., *et al.* PROBAST: A Tool to Assess Risk of Bias and Applicability of Prediction Model Studies: Explanation and Elaboration. *Ann Intern Med* **170**, W1-w33 (2019).
44. Ritchie, M.E., *et al.* limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res* **43**, e47 (2015).
45. Stalpers, L.J.A. & Kaplan, E.L. Edward L. Kaplan and the Kaplan-Meier Survival Curve. *BSHM Bulletin: Journal of the British Society for the History of Mathematics* **33**, 109-135 (2018).
46. Goodman, L.A. Snowball Sampling. *The Annals of Mathematical Statistics* **32**, 148-170, 123 (1961).
47. Hernán, M.A. & Robins, J.M. *Causal Inference: What If.* , (Chapman & Hall/CRC, Boca Raton, 2020).
48. Lee, J. & Spratling, R. Recruiting Mothers of Children With Developmental Disabilities: Adaptations of the Snowball Sampling Technique Using Social Media. *J Pediatr Health Care* **33**, 107-110 (2019).
61. Huber, W., *et al.* Orchestrating high-throughput genomic analysis with Bioconductor. *Nat Methods* **12**, 115-121 (2015).
62. Falagas, M.E., Pitsouni, E.I., Malietzis, G.A. & Pappas, G. Comparison of PubMed, Scopus, Web of Science, and Google Scholar: strengths and weaknesses. *Faseb j* **22**, 338-342 (2008).
63. Hernán, M.A. How to estimate the effect of treatment duration on survival outcomes using observational data. *Bmj* **360**, k182 (2018).
64. Chatton, A., *et al.* G-computation, propensity score-based methods, and targeted maximum likelihood estimator for causal inference with different covariates sets: a comparative simulation study. *Sci Rep* **10**, 9219 (2020).
65. Naimi, A.I., Cole, S.R. & Kennedy, E.H. An introduction to g methods. *Int J Epidemiol* **46**, 756-762 (2017).
66. Doosti-Irani, A., Mansournia, M.A. & Collins, G. Use of G-methods for handling time-varying confounding in observational research. *Lancet Glob Health* **7**, e35 (2019).
67. Dukes, O. & Vansteelandt, S. A Note on G-Estimation of Causal Risk Ratios. *Am J Epidemiol* **187**, 1079-1084 (2018).
68. Hernán, M.A. The C-Word: Scientific Euphemisms Do Not Improve Causal Inference From Observational Data. *Am J Public Health* **108**, 616-619 (2018).
69. Hernán, M. The C-Word: The More We Discuss It, the Less Dirty It Sounds. *Am J Public Health* **108**, 625-626 (2018).
70. Van Calster, B., van Smeden, M., De Cock, B. & Steyerberg, E.W. Regression shrinkage methods for clinical prediction models do not guarantee improved performance: Simulation study. *Stat Methods Med Res* **29**, 3166-3178 (2020).
71. van der Ploeg, T., Austin, P.C. & Steyerberg, E.W. Modern modelling techniques are data hungry: a simulation study for predicting dichotomous endpoints. *BMC Med Res Methodol* **14**, 137 (2014).
72. Friedman, J. Greedy function approximation: A gradient boosting machine. *Annals of Statistics* **29**, 1189-1232 (2001).
73. Breiman, L. Random Forests. *Machine Learning* **45**, 5-32 (2001).
74. Seide, F., Fu, H., Droppo, J., Li, G. & Yu, D. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs. in *INTERSPEECH* (2014).
75. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436-444 (2015).
76. Filonenko, A., Kurnianggoro, L. & Jo, K. Comparative study of modern convolutional neural networks for smoke detection on image data. in *2017 10th International Conference on Human System Interactions (HSI)* 64-68 (2017).
77. He, K., Zhang, X., Ren, S. & Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. in *2015 IEEE International Conference on Computer Vision (ICCV)* 1026-1034 (2015).
78. Glorot, X. & Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, Vol. 9 (eds. Yee Whye, T. & Mike, T.) 249--256 (PMLR, Proceedings of Machine Learning Research, 2010).
79. Goyal, P., *et al.* Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. Preprint at https://arxiv.org/abs/1706.02677 (2017).
80. Szegedy, C., *et al.* Going deeper with convolutions. in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 1-9 (2015).