bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

REM: An Integrative Rule Extraction Methodology for Explainable Data Analysis in Healthcare

Zohreh Shams[1,*], **Botty Dimanov**[1], **Sumaiyah Kola**[1], **Nikola Simidjievski**[1], **Helena Andres Terre**[1], **Paul Scherer**[1], **Urška Matjašec**[1], **Jean Abraham**[2,3,4], **Pietro Liò**[1], **Mateja Jamnik**[1]

1 Department of Computer Science and Technology, University of Cambridge, UK
2 Department of Oncology, University of Cambridge, Addenbrooke's Hospital, Cambridge, UK
3 Cambridge Breast Cancer Research Unit, Cambridge University Hospitals NHS Foundation Trust, Cambridge, UK
4 NIHR Cambridge Biomedical Research Centre, Cambridge, UK
*zohreh.shams@cst.cam.ac.uk

Abstract

Deep learning models are receiving increasing attention in clinical decision-making, however the lack of interpretability and explainability impedes their deployment in day-to-day clinical practice. We propose REM, an interpretable and explainable methodology for extracting rules from deep neural networks and combining them with other data-driven and knowledge-driven rules. This allows integrating machine learning and reasoning for investigating applied and basic biological research questions. We evaluate the utility of REM on the predictive tasks of classifying histological and immunohistochemical breast cancer subtypes from genotype and phenotype data. We demonstrate that REM efficiently extracts accurate, comprehensible and, biologically relevant rulesets from deep neural networks that can be readily integrated with rulesets obtained from tree-based approaches. REM provides explanation facilities for predictions and enables the clinicians to validate and calibrate the extracted rulesets with their domain knowledge. With these functionalities, REM caters for a novel and direct human-in-the-loop approach in clinical decision making.

1 Introduction

Diagnosis, prognosis and treatment planning in healthcare are nowadays informed by a variety of data types ranging from imaging to genomic biomarkers and electronic health records. The differences among these data types challenge common statistical methods. This has turned the attention to Machine learning (ML), and in particular deep neural networks (DNNs) that are capable of handling high volume of heterogeneous data. Unfortunately, while accurate, the opacity of many ML models such as DNNs poses a great challenge for their deployment in safety critical domains, such as oncology. This is evidenced by the scepticism of clinical community about ML systems [4, 21].

Addressing the ML opacity issues has gained a lot of attention in recent years to the extent that explainability of decisions made by machine is now a legal requirement in some cases [14]. This has given rise to the emerging fields of interpretable and explainable ML (See Supplementary materials for essential definitions). While typically these terms are used interchangeably, we make a distinction: interpretable ML focuses on the model itself and "how" well its underlying processes for making a prediction can be understood. Explainable ML, on the other hand, solely focuses on understanding "why" an ML model makes a prediction.

Preprint. Under review.

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Models that are interpretable (e.g., decision tree, ruleset) are explainable by default. For non-interpretable models (e.g., DNN, SVM), the explanation is provided by post-hoc means, most common of which is feature importance [3], where for each prediction the importance of individual features is approximated by removal or perturbation [24, 27, 25]. Recent user studies, however, show that feature importance does not necessarily increase human understanding of the model and its predictions [17, 22]. This causes serious challenges in domains such as healthcare, where verifiability and simulatability are crucial in supporting clinical decision-making. The former allows clinicians to inspect the predictions of an ML model and contrast them with their expert knowledge to verify the biological relevance, while the latter allows checking the impact of perturbation in input on the output and adjusting the model accordingly.

To support clinical decision making, we propose an integrative Rule Extraction Methodology (REM)[1] (Figure 1) that addresses the issue of interpretability and explainability simultaneously, while catering for verifiability and simulatability. The REM-D component of REM approximates a DNN with an interpretable ruleset model and uses that ruleset to explain the predictions of DNN. The REM-T component, on the other hand, extracts rulesets from tree-based approaches (e.g., decision trees, random forests). The REM-D is a decompositional method that decomposes a DNN into adjacent layers from which rules are extracted. Compared to the methods that extract rules from the network predictions directly without considering the inner working of the network, decompositional approaches benefit from the noise removal property for neural networks and take into account the role of hidden features as well as input features [26].

In addition to gaining explainability and interpretability, the advantage of approximating DNNs with ruleset models is in easy integration of such rulesets with other data and knowledge driven rules to allow reasoning with more than one modality. REM pipeline (Figure 1) shows an instantiation of this process, where rulesets are extracted by REM-D and REM-T from data modalities modelled with DNNs and tree-based approaches, respectively. Together with rulesets coming from knowledge modalities, the final ruleset allows domain experts to conduct reasoning, get explanation for the predictions made by the reasoning process, inspect the biological relevance of the predictions and, calibrate them with their expertise. With these functionalities, REM essentially caters for a novel and direct human-in-the-loop approach in clinical decision making.

REM pipeline, by using ruleset as surrogate for DNNs, makes the following contributions: (i) facilitates highlighting the similarities and differences between the rules of decision making by a machine and human experts; (ii) provides the clinicians/physicians with an explanation of predictions; (iii) allows clinicians/physicians to adjust the model and therefore its decision-making based on their expertise; and (iv) enables flexible multi-modality reasoning with rules coming from various modalities and models.

2 Results

We employ REM in two real world case studies on breast cancer for predicting imaging histological subtypes based on mRNA expression data and predicting immunohistochemical (IHC) subtypes based on a combination of clinical and mRNA expression data. The results of rule extraction are evaluated from three quantitative perspectives: (i) predictive performance (i.e., accuracy and fidelity of the ruleset), (ii) comprehensibility (i.e., size of ruleset and rules average length), and (iii) efficiency of rule extraction (i.e., time and memory usage). We demonstrate that in a time and memory efficient manner, REM extracts ruleset models that are accurate and comprehensible. In addition to these quantitative measures, we show that the rulesets extracted are biologically relevant, calibratable based on experts' domain knowledge and easy to integrate with rules coming from other modalities and models. In what follows, we first outline the two case studies.

2.1 Case Studies

Predicting histological subtypes of breast cancer from mRNA expressions Histological subtypes of breast cancer aim to capture the heterogeneity of breast cancer based on the morphology evident in the pathology images of tumours. Invasive Ductal Carcinoma (IDC) and Invasive Lobular Carcinoma

[1]The repository will be available after review process. In the meantime, please contact the corresponding author for related questions.

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

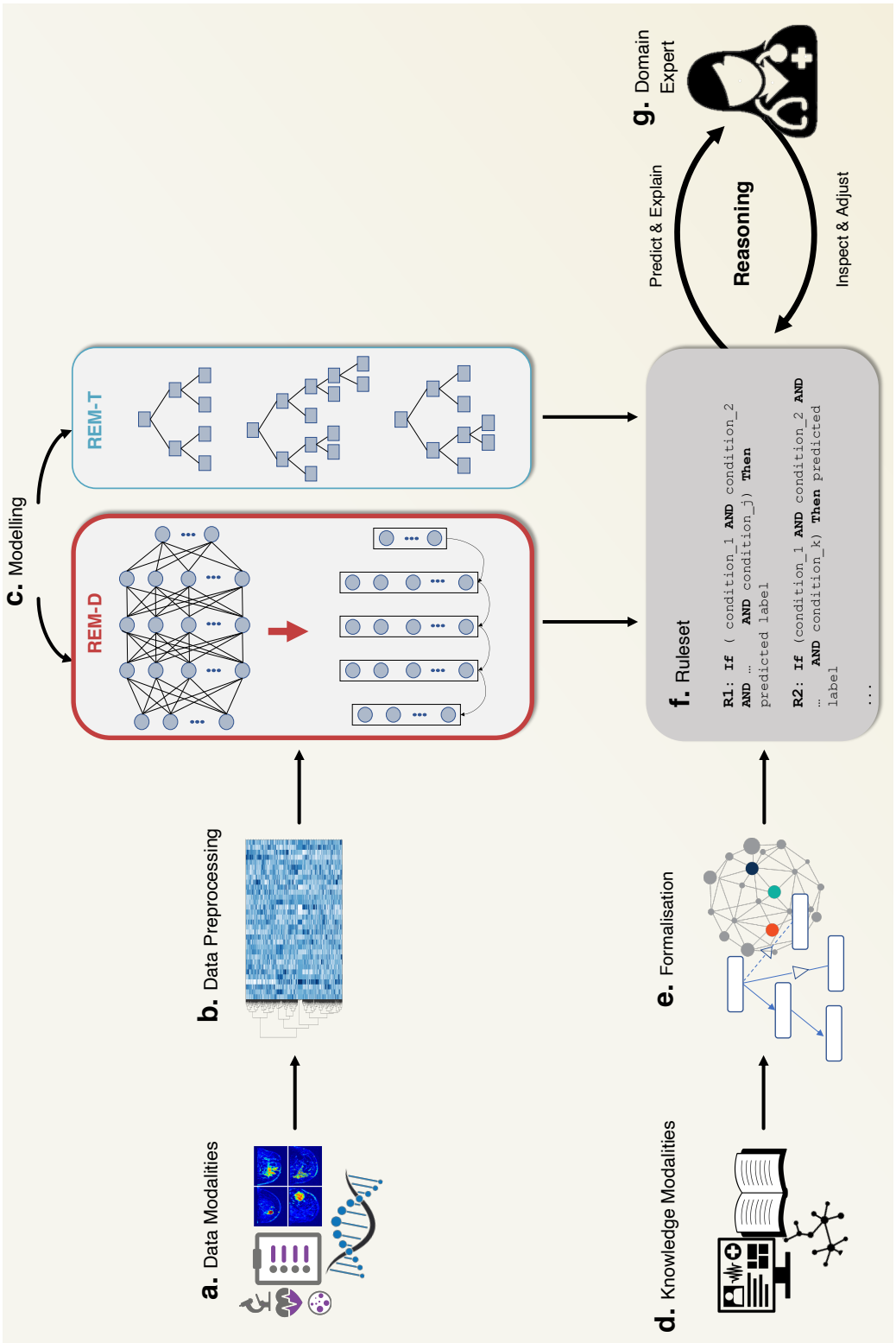

Figure 1: Workflow of REM employed to support clinical decision-making. **a.** REM gets tabular input from various data modalities (e.g., genomic, imaging, clinical). **b.** These modalities are subjected to preprocessing including feature selection. **c.** REM-D and REM-T extract rules from trained DNNs and tree-based approaches used for modelling certain modalities. **d., e.** Rules extracted from Knowledge modalities (e.g., medical guidelines, biomedical knowledge, electronic health record), and formalised (e.g., using knowledge graphs and ontologies), can be integrated into the rules extracted from data modalities. **f.** The ruleset extracted from various modalities and models allows reasoning involved in predicting a target using multiple data, multiple knowledge, or a combination of data and knowledge modalities simultaneously. **g.** Reasoning processes can be carried out for prediction and explaining the predictions to clinical experts, while allowing them to inspect the biological relevance of the reasoning and adjusting it with their expertise.

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

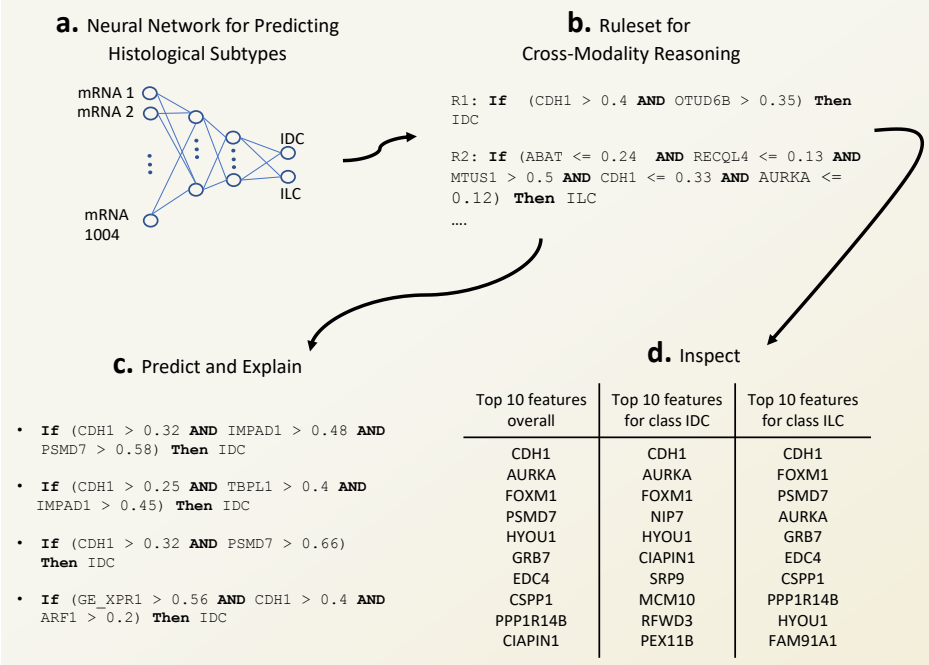

Figure 2: Cross-modality reasoning. **a.** A neural network is trained to predict histological subtypes of breast cancer based on 1004 mRNA expressions. **b.** Rules are extracted from the trained network using REM-D. **c.** The explanation for predicting the histological type of a hypothetical patient (generated by randomly sampling numbers between 0 and 1 to represent the input features) in the form of specific rules used for this prediction. **d.** Top features appearing in the rules extracted from the entire dataset to allow inspection of biological relevance of extracted ruleset using existing bioinformatics tool.

(ILC) are the two most common histological subtypes of breast cancer identified from pathology images, respectively. There is little known about the connection between genetic factors and these histological subtypes [10, 6]. To reveal the connection between the two, we use what we refer to as cross-modality reasoning: reasoning involved in predicting a target that is based on different data modalities rather than those inputted to the model. Whilst cost consideration makes predicting genetic targets with imaging more practical than the other way around, research wise, it is equally informative to investigate the prediction of histological subtypes with genetic factors. To this end, we predict the histological subtypes of IDC and ILC from mRNA expression profiles of patients in the METABRIC dataset [7], using DNNs (Figure 2a). 1694 patients out of the total 1980 in METABRIC, belong to one of these two subtypes. Using 80% of these 1694 records for training (1355 patients), the predictive performance of a neural network is measured across five folds on the remaining 20% (339 patients), when identifying IDC vs ILC subtypes from 1004 mRNA expressions.

Predicting IHC subtypes of breast cancer from mRNA expressions Apart from histological subtypes, patients in the METABRIC dataset are assigned to various other groups, one of which is coming from the two IHC subtypes (ER+ and ER-) that are very important in deciding treatment options. Similar to previous scenario, using 80% of the METABRIC dataset for training (1584 patients), the predictive performance of neural network is measured across five folds on the remaining 20% (396 patients), when identifying patients IHC subtypes (1506 ER+ cases vs 474 ER- cases) from 1000 mRNA expressions (Figure 3a).

2.2 REM-D efficiently extracts accurate and comprehensible rules

Using REM-D, we extract rules from neural networks for both case studies above (Figure 2b and Figure 3c). The results in terms of predictive performance, efficiency and comprehensibility are reported in Table 1.

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

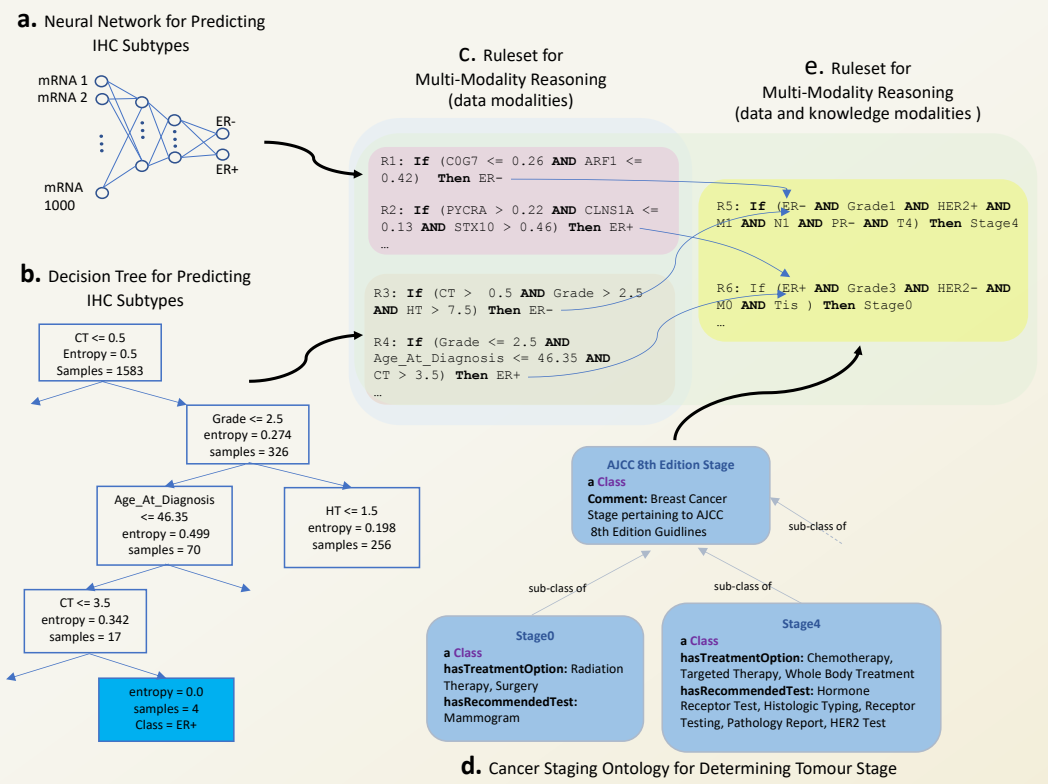

Figure 3: Multi-modality reasoning. **a.** A neural network is trained to predict IHC subtypes of breast cancer based on 1000 mRNA expressions. **b.** A decision tree is trained to predict IHC subtypes of breast cancer based on 13 clinical variables. **c.** A combination of rules extracted from mRNA expressions using REM-D and from clinical variables using REM-T gives rise to a ruleset for multi-modality reasoning. **d.** An existing ontology formalises the medical guidelines in the latest AJCC Cancer Staging Manual. **e.** Rules from data modalities can be integrated with rules from knowledge modalities.

For the first case study, from predictive performance perspective, the extracted rules closely mimic the decision of the neural networks (average fidelity of 88.4%). The average accuracy of the rules when used for prediction is 88.6%, which is almost identical to the original neural network. Comprehensibility remains high too, making the ruleset easy to audit and comprehensible for domain experts: the median of the size of the ruleset is 137, while the median of the average length of rules is 4.3 (i.e., rules have between 4 to 5 conditions). With respect to efficiency, rules are extracted efficiently in below 3 minutes (median) with memory consumption of 573.231 megabytes (median), using the hardware stated in Section 4.2. Note that the standard deviation across folds for comprehensibility and efficiency metrics can be high. Therefore, for these metrics we report the median instead of average, which gives a clearer overview of the REM-D behaviour. We outline our hypothesis about this observation in the Discussion section.

Regarding the second case study, the average accuracy and fidelity of the rules when used for prediction are 92.1% and 92.6%, respectively. This means that replacing the neural network with the ruleset compromises just over 3% of the accuracy. The median of the size of the ruleset is 103, while the median length of rules is 5.1, ensuring a high degree of comprehensibility. The efficiency remains high. It takes below 350 megabytes and 2 minutes to extract the rules, using the same hardware.

The extracted rulesets for each case study can be simulated easily to get a prediction for a new patient for either of the targets. In addition, it can be explained why a prediction was made, where the explanation is in the form of rules that were satisfied (i.e., the conditions in them were met) and thus counted towards the prediction (e.g., Figure 2c). Having assessed the extracted ruleset from

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

target	DNN accuracy	REM-D accuracy	REM-D fidelity	REM-D duration (sec)	REM-D memory (MB)	size of ruleset	rules average length
ILC/IDC	0.882	0.886	0.884	143.792	573.231	137	4.3
ER+/ER-	0.957	0.921	0.926	100.714	349.951	103	5.1

Table 1: Rule extraction from neural network when predicting ILC/IDC (row 1) and ER+/ER- (row 2) using mRNA expressions. Results reported are average across five fold cross validation for accuracy and fidelity, and median for the rest of metrics. Results for each fold for ILC/IDC and ER+/ER- predictions are in Supplementary Tables 5 and 6, respectively.

quantitative perspective, in the next three sections, we focus on the qualitative aspects. Each aspect is elaborated on for one of the case studies and can be applied to the other one in a similar fashion.

2.3 Extracted rules are biologically relevant

To investigate the biological relevance of the rules extracted for the first case study, we look at the features (i.e., mRNA expressions) that appear most frequently in the rules extracted across all folds. As a proof of principle, the top ten most occurring genes are listed in Figure 2d. The top genes code for a variety of proteins that do not cluster in a unique functional group and have key function in most of cell compartments, therefore promoting cancer as transcription factors, cell cycle regulators, cell-cell contacts and cell-cell signalling disruption. They seem remarkably in synergy with promoting the progression and sustainability of the cancer along all key steps [15]. CDH1 is the most common feature. This gene codes for a cadherin involved in the key Wnt-mediated beta catenin signaling. Close to 80% of the rules associated with class ILC that include this feature prescribe an upper bound restriction for it, which is aligned with the reported association of ILC tumour and the under-expression of CDH1 [10, 6]. In contrast, all the rules associated with class IDC that include this feature restrict its lower bound. This means that these rules are more likely to get triggered and therefore predict IDC for patients with over-expression of CDH1. In addition to CDH1, the three features that are commonly used in rules for both classes are AURKA, FOXM1, HYOU1. They are all protein coding genes. AURKA is a kinase and one of the most powerful regulators of the cell cycle. All rules for class IDC restrict the lower bound for AURKA, FOXM1 and HYOU1. This is likely to point towards the over-expression of these genes in IDC tumours. The opposite is true for rules associated with class ILC: close to 99% and 95% of the rules limit the upper bound of AURKA and FOXM1, respectively; as well as all the rules limiting the upper bound of HYOU1, making a case for potential under-expression of FOXM1 and HYOU1 in ILC tumours. Apart from CDH1, FOXM1 HYOU1, and AURKA, PSMD7 appears in the top half of the overall features. The variety of functional roles and the different localisations of the proteins coded by these genes provide an overview of different aspects of molecular and cellular functions involved, helping clinicians and physicians to form an overall vision about the task in hand.

2.4 Extracted rules can be adjusted based on domain knowledge

Clinicians and physicians can use their expert knowledge to adjust the model by expressing which features to focus on. This information can be used to impose a hierarchy on the ruleset by assigning scores to rules, where the score takes into account the inclusion of the preferred expressed features. We use an augmented hill climbing algorithm for scoring rules. In the base version [19] the algorithm assigns a score to each rule based on its coverage, accuracy and length, where high accuracy and coverage are rewarded, while high score for length is penalised. In the augmented algorithm, referred to as personalised ranking [20], the rules that include features proposed by domain experts get additional scores and are thus likely to rank higher in the hierarchy of rules (Equation 2, Methods). To show the impact of scoring, we pick four genes at random (e.g., KCTD3, RARA, STARD3 and ERLIN2) and assume they are expressed as highly relevant by an expert when predicting IHC subtypes using 1000 mRNA expressions (Figure 3a). The frequency of favourite features in the top 70% of the ruleset increases when using personalised ranking (purple bars, Figure 4) as opposed to the ranking merely based on coverage, accuracy and length (blue bars, Figure 4), indicating that the model is indeed adjusted in the direction of expert knowledge. ERLIN2 is the exception, where the ranking does not make a difference. Potential reasons for this are the absence of more rules that

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

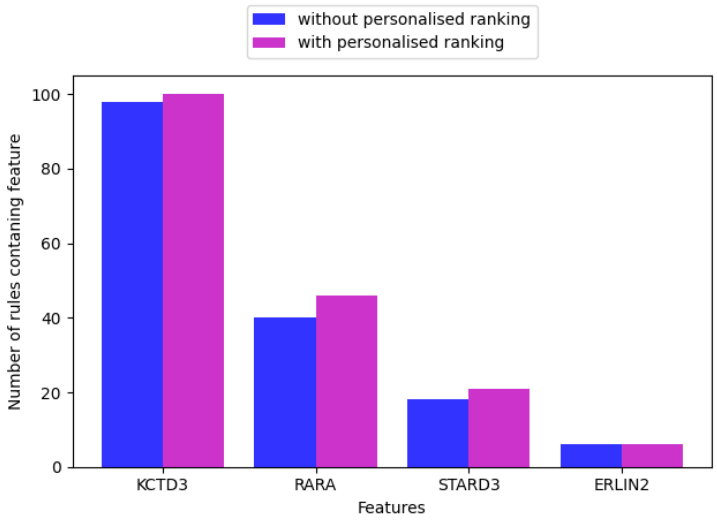

Figure 4: The impact of personalised ranking on the increase in the number of higher rank rules (top 70% of the rules) including the preferred features expressed by domain experts.

data modality	rule extraction mode	rule extraction accuracy	size of ruleset	rules average length
mRNA	REM-D	0.921	103	5.1
Clinical	REM-T	0.81	29	4.6
mRNA + Clinical	REM-D + REM-T	0.933	133	5.0

Table 2: Integration of rules extracted from mRNA data modelled using a neural network (mode REM-D) with rules from clinical data modelled using a decision tree (mode REM-T), when predicting IHC subtypes. Results reported are averaged across five fold cross validation, apart from size of ruleset, where median is reported. Results for each fold are in Supplementary Tables 6, 7 and 8 for mRNA, Clinical, mRNA + Clinical, respectively.

include ERLIN2. High specificity of the rules that include ERLIN2 could be another reason: although the ranking favours the inclusion of this feature, it does not favour long rules that are highly specific.

2.5 Extracted rules can be integrated with rules from other modalities and models

Beside mRNA expressions, IHC subtypes can also be predicted from clinical data in METABRIC. Unlike the genomic part of the data that tends to be high dimensional, the clinical part often consists of a handful of variables. We predict the IHC subtypes from clinical data using a decision tree (Figure 3b). The average accuracy of decision trees across the folds is 81%, while the median of the ruleset size and rule length are 29 and 4.6, respectively, as presented in Table 2, second row.

To allow multi-modality reasoning, we combine the rules extracted from the decision tree, using REM-T, with the rules from the network, using REM-D (Figure 3c). As displayed in Table 2, third row, this increases the accuracy of the ruleset slightly over the five folds, in comparison with the ruleset that relies on rules extracted from the neural network only. The size of the ruleset increases as expected, while the median of the average length of the rules drops below that of the rules extracted by REM-D. More important than the slight increase in the accuracy is the fact that rules from different modalities and modelled with various models can indeed be integrated and contribute to one another to increase the accuracy of predictions.

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

3 Discussion

The majority of interpretability and explainability methods used in healthcare are local, pointing to important input features that were most influential in the prediction [31, 30, 1, 3]. Recent user studies, however, show that feature-based explanations do not necessarily increase human understanding of the model and its prediction [17, 22]. In addition, minor adversarial perturbations to the input, or model parameters, can conceal important aspects of the model's decision making process, making feature importance explanations unreliable [11, 9, 8]. Finally, feature importance methods, as most local methods, are hard to scale since interpreting a single prediction requires the execution, or sometimes even the retraining [24], of a local method. REM provides a global view of interpretability and explainability simultaneously. The ruleset models extracted by REM give an overview of the original underlying model (e.g., DNNs), while they remain usable for local explanation of the predictions. They also highlight the similarities and differences between the rules of decision making in machine and human experts, while easily integratable with rules coming from other sources. In addition, since rulesets make the decision-making process more transparent, they enable complex human-in-the-loop scenarios. For example, domain experts could manipulate the rules, impose hierarchy on rules, and emphasise the subset of rules that may be more useful for a subgroup of patients (e.g., of certain ethnicity).

The Majority of rule extraction methods from neural networks are limited to rule extraction from networks with one hidden layer [26]. Zilke et al. [33] extract rules from networks with more than one hidden layer (i.e., DNNs), however, the approach has scalability issues due to high memory and time consumption [12, 33]. REM-D has few key differences with [33]: it uses a more efficient rule extraction algorithm between layers. It merges the rules extracted between layers incrementally (e.g., rules extracted between layer 4 and 5 will be merged with rules extracted between layers 3 and 4 as soon as the latter become available) as opposed to first extracting all layer-wise rules followed by a final merge. This is crucial because rules can be subject to quality check after each step of merge (see Section 4.3 for satisfiability and redundancy checking). This eliminates the need for considering unsatisfiable rules as well as terms that are redundant, which in turns improve the time and memory consumption. Kazhdan et al. [16] propose a decompositional approach to extract information from Convolutional Neural Network (CNN)'s layers in the form of "concepts". Concepts, in the setting of images, are high-level semantic units (e.g., lung opacity, foreign object) rather than individual input features (e.g., pixels, or characters). These concepts may be used in combination with REM to form rulesets containing high-level explanations for image-based data modalities.

REM embodies a set of functionalities that can be used for revealing the connections between various data modalities (cross-modality reasoning) and integrating the modalities for multi-modality reasoning, despite being modelled using a combination of methodological approaches (DNNs and tree-based). The latter is particularly important when working with different data modalities. Instead of using a single model for all modalities, each modality is modelled using the prediction model that suits the data best, while the integration of rules extracted from each model allows multi-modality reasoning. REM functionalities also allow direct incorporation of knowledge into data-driven reasoning by catering for rule ranking based on the expertise of clinicians/physicians. Alternatively, rules can be extracted from knowledge modalities (e.g., electronic health records and biomedical ontologies) and integrated with data-driven rules. Figure 3 showcases such a scenario. Rules extracted from data provide input to rules extracted from the Breast Cancer Staging Ontology [28], which is based on the latest AJCC Cancer Staging Manual [2]. In a hypothetical scenario, where the IHC subtypes and tumour stage of the patients are unknown, the former can be predicted from data modalities (Figure 3d) and used in the prediction of the latter, assuming other variables required for this latter prediction (i.e., Grade, HER2 status, PR status, T: severity of tumour size, N: severity of the spread to the lymph nodes, M: metastasize status) are known and provided as facts.

We demonstrated the use of REM and its functionalities in two breast cancer case studies, making this work the first, to the best of our knowledge, that integrates machine learning and reasoning in oncology domain. In cross-modality case study we used genetic data to predict image-based targets. Cost consideration, however, often makes accessing genetic data challenging. Applied in reversed order, genetic components involved in targeted gene panel testing may be predicted based on widely available imaging data. REM paves the way to investigate the rules of going directly from imaging to panel testing and thus bringing target therapy to more patients in absence of genetic data. In multi-modality reasoning we look at the complementariness of modalities (e.g., mRNA expression

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

and clinical) for predicting certain targets (e.g., IHC subtypes), which can be exploited for examining the importance of modalities for various prediction tasks.

We envision several exciting avenues for future work. Deep learning models often have multiple optima of similar predictive accuracy, while their interpretability can vary considerably [13]. We postulate that neural networks that give rise to smaller number of rules in a shorter span of time and with less memory consumption, when subjected to rule extraction are those that are more interpretable and hence easier to explain with a smaller number of symbolic rules. Our results (Supplementary Table 5 and Table 6) also suggest that the larger rulesets tend to have longer rules that are highly specific due to inclusion of several conditions and are therefore less suited to generalise to unseen samples. Using optimisation techniques that focus on optimising deep models for human-simulatability [32] is expected to encourage finding models whose decision boundaries are well-approximated by small decision trees that in turn give rise to a small ruleset, which are less likely to overfit. Further to quantity, the quality of the rules in terms of biological relevance can extend to investigating the connection between proteins coded by genes in rules and sets of proteins participating in known cancer pathways. While user studies need to verify this, we are positive that accommodating such functionalities in REM has the potential to decrease the scepticism of clinical community about ML systems [4, 21] even further.

4 Methods

4.1 Data, pre-processing and feature selection

For all experiments we use a public breast cancer dataset of 1980 patients, METABRIC [7], that aims to characterise breast cancer subtypes based on genomic and imaging data as well as clinical.

Feature selection in cross-modality case study. The aim of this case study is to predict the two main histological subtypes of breast cancer, IDC and ILC (1547 IDC vs 147 ILC cases), using mRNA expressions. Feature selection from METABRIC for this task is done based on existing bioinformatics findings: we use putative breast cancer genes identified in [7], the significance of which is validated by revealing novel subgroups that have distinct clinical outcomes. These genes are identified based on a landscape created by integration of copy number aberrations (CNA) and expressions that highlights genomic regions which are likely to contain driver genes. The top 1,000 cis-associated genes across the integrated CNA-expression landscape are identified as putative genes. Further bioinformatics finding shows that in addition to distinct morphology, mRNA expression profiling of the two main histological subtypes demonstrates distinct molecular differences [10]. The main differences include the variant in expression of four genes: CDH1, MKI67, FOXA1 and PTEN. We add the mRNA expression of these four genes to the 1,000 pre-selected ones (Figure 2(a)).

Feature selection in multi-modality case study. The aim of this case study is to predict the two main IHC subtypes of breast cancer, ER+ and ER- (1506 ER+ vs 474 ER- cases), using mRNA expressions and clinical data. Similar to the above case study, we use the mRNA expressions of the 1,000 putative genes (Figure 3(a)). The clinical data (Figure 3(b)) used in combination with the genomic data consists of 13 variables (age at diagnosis, tumour laterality, Nottingham Prognostic Index, menopause status, number of positive lymph nodes, chemotherapy agent, hormone therapy agent, radiotherapy agent, grade, tumour size, histological type, stage, cellularity) that are either continuous (e.g., age) or categorical (e.g., menopause status).

There are no missing values in mRNA expressions. Expressions are normalised and scaled to [0,1] prior to use. Occasional missing values in clinical part are replaced by mean value for continuous variables. They are treated as a new category for categorical variables. After pre-processing, the data is sampled into five-fold cross-validation splits based on the class distribution of the target.

4.2 Model selection

A fully connected neural network with two hidden layers is set up for the classification tasks of predicting histological subtypes and IHC subtypes outlined above. We used Keras library [5] for this. Tanh is used as the activation function for the hidden layers, and softmax is used for the output layer. Adam optimiser [18] is used for training without any regularisation. The weight of classes are considered too when fitting the model. Batch size, number of epochs as well as the number of neurons in the two hidden layers are determined based on grid search as follows:

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

batch-size = $\{16, 32, 64, 128\}$, epochs = $\{50, 100, 150, 200\}$, layer-1 = $\{16, 32, 64, 128\}$ and layer-2 = $\{8, 16, 32, 64\}$. Best performance (i.e., average AUC of five-fold cross validation) is obtained with: batch-size=16, epochs=50, layer-1=128, and layer-2=16 when predicting histological subtypes using 1004 mRNA expressions. Incidentally, he same applies to predicting IHC subtypes. When modelling clinical data using decision tree (Keras library [5]), rules were extracted without limiting the depth of the tree, as well as limiting trees to depths 5, 10 and 15 (max_depth=$\{5, 10, 15\}$). The accuracy of predicting IHC subtypes was highest when maximum depth of the decision tree was limited to 5. All the experiments are done on a server with 66 GB RAM and 2 AMD 6367 processor.

4.3 REM-D rule extraction methodology

REM-D extracts rules (Algorithm 1) for a classification problem with n data points, $\vec{x_1}, \vec{x_2}, ..., \vec{x_n}$, each with an associated class, $y_i \in \{y_1, y_2, ..., y_u\}$. Rules are extracted from a trained network with k hidden layers, $\{h_0, h_1, ..., h_k, h_{k+1}\}$, where h_0 and h_{k+1} are in fact the input and output layers. Each layer h_i has H_i neurons. The input layer, thus has as many neurons as the features available for each data point: $H_0 = |\vec{x_i}|$. The number of neurons in the output layer equals to the number of classes: $H_{k+1} = u$. The activation values sampled at layer h_i for data point $\vec{x_j}$ are denoted as $h_i(\vec{x_j})$.

Ruleset structure. Prior to outlining the methodology for rule extraction, we present the structure of the ruleset extracted by REM-D:

- Total ruleset: $R_{total} = \bigcup_{i=1}^{u} R^i_{total}$
- Ruleset for each class: $R^i_{total} = \bigvee_{j=1}^{m} R^i_j$
- Individual rules: $R^i_j =$ If $antecedent$ then i
- Antecedent: $\bigwedge_{k=1}^{n} t_k$
- Terms: $t_k : h <= threshold$ or $t_k : h > threshold$

The total ruleset is the union of ruleset for each class. The ruleset for each class is a disjunction of individual rules, each of which is a conditional statement with an antecedent and the class they belong to as conclusion. The antecedent itself is a conjunction of conditional statements referred to as terms that are conditions on the activation value of neurons.

Rule extraction algorithm. In order to extract rules of the form described above from the network, REM-D first decomposes the trained network into adjacent layers and then uses a tree induction algorithm to extract rules from pairs of layers in the network. In contrast to existing approaches [33], instead of C4.5 [23], REM-D uses a more efficient tree induction and rule extraction algorithm, called C5 [23]. C5 is faster than C4.5, while consuming less memory. Furthermore, C5 induces trees and extracts rules from them that are more accurate.[2]

When generating the decision tree, in order to identify the features that lead to a split, C5 uses entropy [29] for measuring purity. The information gain of a feature is calculated based on the difference of entropy in the segment before the split and the partitions resulting from the split. The features with higher information gain are then used for splitting the data in the decision tree. Once the tree is generated, nodes and branches that have little effect on the classification errors are pruned. Rules are then extracted from the pruned tree. The rules extracted by C5, from each two adjacent layers, k and $k + 1$, use the features from layer k. Thus, except when k is the input layer, the rules use hidden features in a hidden layer. To make sure that the final ruleset maps the input to the output, rules are merged in a backward fashion, such that the hidden features in each rule are replaced by features in the previous layer until the input layer is reached and all rules are expressed only in terms of input features. Algorithmic details are as follows.

The algorithm starts by iterating over each class (line 1). For each class, a rule is defined that maps the last hidden layer to the output classes (line 2). For example, if there are two classes, represented by neuron 1, and 2 in the output layer, which we assume is the 5th layer, for the first one we have: $R^v_{initial} =$ IF $h_{5,1} > 0.5$ Then y_v, where $h_{5,1}$ refers to neuron 1 of layer 5. A total rule is introduced for each class that only contains the initial rule to begin with (line 3). For the class in hand, each layer preceding the output layer is considered in a descending order (line 4). For each layer and

[2] See `https://rulequest.com/see5-comparison.html` for comparison between C4.5 and C5 by the author of both algorithms.

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Algorithm 1 REM-D

Input: Artificial neural network $\{h_0, h_1, ..., h_{k+1}\}$, training data $\{(\vec{x_1}, y_1), ..., (\vec{x_n}, y_n)\}$
Output: Ruleset approximating the neural network R_{total}

1: **for** *class* $y_v \in y_1, y_2, ..., y_u$ **do**
2: $R^v_{initial} \leftarrow \{IF\ h_{k+1,v} > 1/u\ THEN\ y_v\}$
3: $R^v_{total} = R^v_{initial}$
4: **for** layer $j = k, ..., 0$ **do**
5: $I_{h_j \to h_{j+1}} \leftarrow \emptyset$
6: $T = Set(getTermsFromRuleset(R^v_{total}))$
7: $\vec{x_1}', \vec{x_2}', ..., \vec{x_n}' \leftarrow h_j(\vec{x_1}), h_j(\vec{x_2}), ..., h_j(\vec{x_n})$
8: **for** $t \in T$ **do**
9: $y_1', y_2', ..., y_n' \leftarrow t(h_{j+1}(\vec{x_1})), t(h_{j+1}(\vec{x_2})), ..., t(h_{j+1}(\vec{x_n}))$
10: $I_{h_j \to h_{j+1}} \leftarrow I_{h_j \to h_{j+1}} \cup \texttt{C5.0}((\vec{x_1}', y_1'), (\vec{x_2}', y_2'), ..., (\vec{x_n}', y_n'))$
11: **end for**
12: $R^v_{total} \leftarrow substitute(I_{h_j \to h_{j+1}}, R^v_{total})$
13: $R^v_{total} \leftarrow deleteUnsatisfiableRules(R^v_{total})$
14: $R^v_{total} \leftarrow deleteRedundantTerms(R^v_{total})$
15: **end for**
16: **end for**
17: $R_{total} \leftarrow R^1_{total} \cup R^2_{total} \cup ... \cup R^u_{total}$
18: returns: R_{total}

its proceeding layer an intermediate empty ruleset is formed that is going to be populated with rules extracted from this layer and the next one (line 5). Unique conditions (referred to as Terms) that appear in the individual rules within the total ruleset are collected in a set (note that the set implementation guarantees the uniqueness of these conditions, and avoids the need for looking for redundancies and deleting them after the terms are collected) (line 6). These conditions are based on hidden neurons in the proceeding layer and need to be replaced with conditions based on the neurons in the current layer till we reach the input layer. For this first the activation values sampled from the current layer are noted (line 7). Next, each condition collected (line 6), is applied to the activations sampled from the proceeding layer to give a target value (line 9). Then in line 10, the values noted in line 7 and targets from line 9 are passed on to C5.0 algorithm for rule extraction. In the same line, the extracted rules will be added to the intermediate ruleset initiated in line 5. By substituting (Algorithm 2) the updated intermediate rules into the total class ruleset, the rules for the class are now described based on neurons in a layer one step closer to input layer (line 12). Merging rules extracted from different layers may give rise to unsatisfiable rules that have contradictory conditions (e.g., Age_At_Diagnosis > 65, Age_At_Diagnosis <= 46) or rules with redundant conditions (e.g., Age_At_Diagnosis > 65, Age_At_Diagnosis > 67). Unsatisfiable rules are deleted in line 13, followed by deleting redundant conditions in line 14. For optimisation purposes, substitution, satisfiability and redundancy checking is done after each step of merging. This results in a lower number of shorter rules for the subsequent merge step, thereby improving time and memory usage. The procedure is repeated until the input layer is reached and the rules for each class are based on input features instead of neurons in the hidden layers. Finally, the rules that describe the behaviour of the network for each class in terms of input features are combined to give the overall ruleset (line 17) that is returned as output (line 18).

In order to make a prediction for a data point using the final ruleset, the majority vote is used: the ruleset for each class has a vote for the prediction which is essentially the number of rules within the class ruleset that are satisfied by the data point. The prediction for the data point is the class that has the majority vote (highest number of rules satisfied).

C5 parameters. The default parameter values are used in C5 (Algorithm 1, line 10). "winnowing" attribute is set to "True" and the number of "minCases" per leaf is determined by grid search. Winnowing in C5 works by calculating a feature importance for each feature based on error rate increase in the training set if the feature was excluded. When set to "True", winnowing allows C5 to use only the important features for tree induction and rule extraction. The minCases parameter stops the decision tree from splitting further if the number of samples in each node drops below the set minimum number of cases after a split. For each experiment we extracted rules by setting the minCases values to: minCases= $\{5, 10, 15, 20\}$ and chose the value that gave the most accurate final

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Algorithm 2 Substitution Procedure

1: **procedure** SUBSTITUTE($I_{h_j \to h_{j+1}}, R^v_{total}$)
2: $R^v_{temp} \leftarrow \emptyset$
3: **for** *rule* $\in R^v_{total}$ **do**
4: $a \leftarrow []$
5: **for** $t \in$ *getTermsFromRule*$(rule)$ **do**
6: $a \leftarrow a + getAntecedent(I_{h_j \to h_{j+1}}, t)$
7: **end for**
8: **for** $C \in CartesianProduct(a)$ **do**
9: $R^v_{temp} \leftarrow R^v_{temp} \cup formRule(C)$
10: **end for**
11: **end for**
12: $R^v_{total} \leftarrow R^v_{temp}$
13: returns: R^v_{total}
14: **end procedure**

u	number of classes
k	number of hidden layers
n	number of data points
m	number of features (i.e., neurons)
x	maximum number of rules in each class ruleset
y	maximum number of terms in each rule
z	maximum number of substitutions for each term

Table 3: Notation used in complexity analysis.

ruleset. When predicting histological subtypes, this value was 10, while 5 was the best value when predicting IHC subtypes.

Substitution algorithm. The substitution procedure replaces the terms in individual rules within a class ruleset with the antecedent of the individual rules within the intermediate ruleset that have these terms as conclusion. For example, if we have t AND $t' \to y$ and within intermediate ruleset we have the following two rules $a \to t$ and $a' \to t'$ the substitution gives rule a AND $a' \to y$. This substitution essentially replaces the terms appearing in the rules of total ruleset with terms from a previous layer. As a result, the rules for the class are now described based on neurons in a layer one step closer to input layer.

The procedure starts with initiating a temporary ruleset for the class in hand (line 2). It then iterates over individual rules in the class total ruleset (line 3). For each of them an empty list is initiated (line 4). For each term in the rule (line 5), the intermediate ruleset is searched for rules with this specific term as conclusion. Those found are added to the list initiated earlier (line 6). If a rule has two terms t_1 and t_2, there maybe several intermediate rules that have each term as conclusion, thus the list may look like $[\{a^1_1, a^2_1\}\{a^1_2, a^2_2, a^3_2\}]$, where a^1_1, a^2_1 are antecedents for t_1 and a^1_2, a^2_2, a^3_2 are antecedents for t_2. Cartesian product of antecedent sets gives all the combination possible when substituting terms in rules within total ruleset (line 8). Each combination forms a new individual rule that is added to the temporary ruleset for the class (line 9). Once this procedure is repeated for each rule, the temporary ruleset replaces the total ruleset (line 12). The updated total ruleset is returned (line 13).

Complexity analysis. The theoretical complexity of REM-D (Algorithm 1) equals: number of classes $\times$ (number of hidden layers + 1) $\times$ [(number of C5 calls $\times$ complexity of C5) + complexity of substitution + complexity of satisfiability checking + complexity of redundancy checking]. The notation used throughout this analysis is presented in Table 3.

Number of C5 calls (Algorithm 1): This number equals to the cardinality of set T defined in line 6. Set T consists of terms in the rules extracted between two adjacent layers proceeding the current layer j (line 4). In absence of pruning, the number of terms from the mentioned ruleset is the same as the number of non-leaf nodes in the tree from which the rules were extracted. In order to calculate the maximum number of non-leaf nodes, we first look at the maximum number of leaf nodes. In the worst case scenario each tree induced between two layers has n leaves, where n is the number of data

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Algorithm 3 REM-T

Input: random forest rf, training data $\{(\vec{x_1}, y_1), ..., (\vec{x_n}, y_n)\}$
Output: Ruleset R_{total}

1: **for** *class* $y_v \in y_1, y_2, ..., y_u$ **do**
2: $R^v_{total} \leftarrow \emptyset$
3: **for** $tree \in rf$ **do**
4: **for** $branch \in tree$ **do**
5: **if** $getLeaf(branch) == class$ **then**
6: $R^v_{total} \leftarrow R^v_{total} \cup createRule(branch)$
7: **end if**
8: **end for**
9: **end for**
10: $R^v_{total} \leftarrow$ *deleteRedundantTerms*(R^v_{total})
11: **end for**
12: $R_{total} \leftarrow R^1_{total} \cup R^2_{total} \cup ... \cup R^u_{total}$
13: returns: R_{total}

points. The total number of non-leaf nodes in a complete binary tree with n leaves is $n - 1$. Thus the number of C5 calls made for each layer j is $n - 1$ in the worst case scenario.

Complexity of C5 (Algorithm 1): C5 is a binary tree induction algorithm the complexity for which is known as $m \times n^2$ in the worst case scenario, where n is the number of data points and m is the number of features for each data point. In REM-D m refers to the number of neurons in layer j (line 4).

Complexity of substitution (Algorithm 2): complexity of this procedure depends on the total number of rules in a class (line 3) and the Cartesian product calculated in (line 8). Assuming that the maximum number of rules in each class ruleset, terms in each rule and substitutes for each term are x, y and z, respectively, the overall complexity of substitution procedure is $x \times z^y$.

Complexity of satisfiability and redundancy checking (Algorithm 2): We established x and y as maximum number of rules in a ruleset and maximum number of terms within each rule, respectively. Satisfiability and redundancy checking both require iterating through every term in a rule for every rule, thus for each of this operations we have the complexity of $x \times y$.

The overall complexity is thus: $u \times (k + 1) \times [((n - 1) \times m \times n^2) + (x \times z^y) + (2 \times x \times y)]$. Determined by the dominant factor, the substitution procedure, the complexity is exponential. While interventions that restrict the number of terms that need substitution, such as limiting the depth of trees between layers, helps with the empirical complexity, the high computational complexity remains a limitation.

4.4 REM-T rule extraction methodology

REM-T extracts rules (Algorithm 3) for a classification problem with n data points, $\vec{x_1}, \vec{x_2}, ..., \vec{x_n}$, each with an associated class, $y_i \in \{y_1, y_2, ..., y_u\}$. Rules are extracted from a trained random forest or decision tree. Here we explain the rule extraction from random forest, the procedure for decision tree is identical to that of random forest with a single tree.

The algorithm starts by iterating over each class (line 1). A total rule is introduced for each class (line 2). For each tree in the random forest (line 3), the branches of the tree (line 4) are traversed. If the branch ends at a leaf node that has the same label as the class (line 5), the algorithm creates a rule from it by the conjunction of conditions in each node in the branch and adds the created rule to the total ruleset for the class (line 6). Similar to Algorithm 1 (line 14) redundant terms are deleted (line 10). The total ruleset for each class are combined to give the overall ruleset (line 12) that is returned as output (line 13).

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

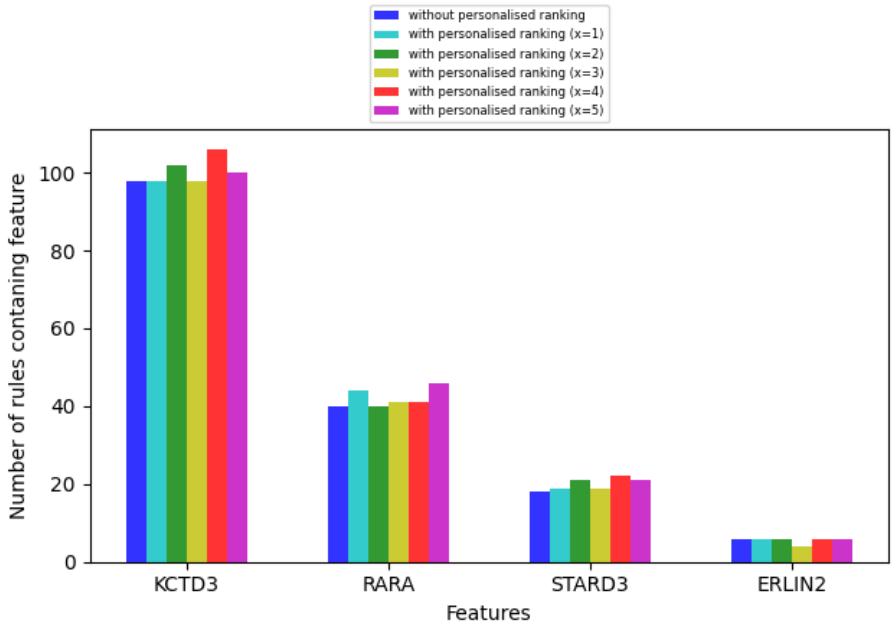

Figure 5: The impact of personalised ranking on increase in the number of higher rank rules (top 70% of the rules) that include the preferred features picked by domain experts, where x ranges from 1 to 5.

4.5 Rule scoring and ranking

The basis of personalised rule scoring and ranking is the score proposed by [19],

$$Rule - Score = \frac{cc - ic}{cc + ic} + \frac{cc}{ic + k} + \frac{cc}{rl} \tag{1}$$

where cc and ic are the number of training samples covered by the rule and classified correctly and incorrectly, respectively, by the rule. rl denotes the rule length. k is set to 4 as per original work in [19]. However, other positive values can be used for k as its role is mostly to avoid the denominator becoming zero, and no significant change in the results is observed by modifying k.

In the personalised rule scoring and ranking proposed by [20], where the assumption is that a list of preferred features is in hand, the formula is extended as follows:

$$Personalised - Rule - Score = \frac{cc - ic}{cc + ic} + \frac{cc}{ic + k} + \frac{cc}{rl} + \frac{x}{i + 2} \tag{2}$$

where i refers to the feature's index in the list of preferred features: the lower the index, the more important is the feature. For a fixed x, the first feature (index 0) adds $x/2$ to the score, while the second feature (index 1) adds less ($x/3$) and so on. The positive constant x can be tuned to give the desired impact.

In the ranking imposed in multi-modality scenario (Figure 4) we assumed all features are equally preferred (i.e., $i = 0$) and experimented with x ranging from 1 to 5 (Figure 5). Some values of x such as 4 and 5 give the most consistent results across all features in terms of increasing the number of rules that include the preferred features, while others are less consistent and may give more boost to only a certain feature. The results presented in Figure 4 use $x = 5$ in personalised ranking.

Acknowledgements

This work was supported by The Mark Foundation Institute for Integrated Cancer Medicine (MFICM). MFICM is hosted at the University of Cambridge, with funding from The Mark Foundation for Cancer Research (NY, U. S. A.) and the Cancer Research UK Cambridge Centre [C9685/A25177] (UK).

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

References

[1] Muhammad Aurangzeb Ahmad, Carly Eckert, and Ankur Teredesai. Interpretable machine learning in healthcare. In *ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, BCB*, pages 559–560. ACM, 2018.

[2] Mahul B. Amin, Frederick L. Greene, Stephen B. Edge, Carolyn C. Compton, Jeffrey E. Gershenwald, Robert K. Brookland, Laura Meyer, Donna M. Gress, David R. Byrd, and David P. Winchester. The eighth edition ajcc cancer staging manual: Continuing to build a bridge from a population-based to a more "personalized" approach to cancer staging. *Ca-A Cancer Journal for Clinicians*, 2017.

[3] Umang Bhatt, Alice Xiang, Shubham Sharma, Adrian Weller, Ankur Taly, Yunhan Jia, Joydeep Ghosh, Ruchir Puri, José M. F. Moura, and Peter Eckersley. Explainable machine learning in deployment. pages 648–657, 2020.

[4] Jonathan H Chen and Steven M Asch. Machine learning and prediction in medicine - beyond the peak of inflated expectations. *The New England journal of medicine*, 376(26):2507–2509, 2017.

[5] François Chollet et al. Keras, `https://keras.io`. Technical report, 2015.

[6] G. Ciriello, M. L. Gatza, A. H. Beck, M. D. Wilkerson, S. K. Rhie, A. Pastore, H. Zhang, M. McLellan, C. Yau, C. Kandoth, et al. Comprehensive molecular portraits of invasive lobular breast cancer. *Cell*, 163(2):506–519, 2015.

[7] Christina Curtis, Sohrab P. Shah, Suet-Feung Chin, Gulisa Turashvili, Oscar M. Rueda, Mark J. Dunning, Doug Speed, Andy G. Lynch, Shamith Samarajiwa, Yinyin Yuan, et al. The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature*, 486:346–352, 2012.

[8] Botty Dimanov, Umang Bhatt, Mateja Jamnik, and Adrian Weller. You shouldn't trust me: Learning models which conceal unfairness from multiple explanation methods. In *European Conference on Artificial Intelligence*, 2020.

[9] Ann-Kathrin Dombrowski, Maximilian Alber, Christopher J. Anders, Marcel Ackermann, Klaus-Robert Müller, and Pan Kessel. Explanations can be manipulated and geometry is to blame, 2019.

[10] Tian Du, Li Zhu, Kevin M. Levine, Nilgun Tasdemir, Adrian V. Lee, Dario A. A. Vignali, Bennett Van Houten, George C. Tseng, and Steffi Oesterreich. Invasive lobular and ductal breast carcinoma differ in immune response, protein translation efficiency and metabolism. *Scientific Reports*, 8(7205), 2018.

[11] Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. *AAAI*, 2019.

[12] Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. Explaining explanations: An overview of interpretability of machine learning. In *IEEE International Conference on Data Science and Advanced Analytics,DSAA*, pages 80–89. IEEE, 2018.

[13] Ian J. Goodfellow, Yoshua Bengio, and Aaron C. Courville. *Deep Learning*. Adaptive computation and machine learning. MIT Press, 2016.

[14] Bryce Goodman and Seth Flaxman. European union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine*, 38(3):50–57, 2017.

[15] Douglas Hanahan and Robert A. Weinberg. Hallmarks of cancer: The next generation. *Cell*, 144(5):646–674, March 2011.

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

[16] Dmitry Kazhdan, Botty Dimanov, Mateja Jamnik, Pietro Liò, and Adrian Weller. Now you see me (CME): concept-based model extraction. In Stefan Conrad and Ilaria Tiddi, editors, *Proceedings of the CIKM 2020 Workshops co-located with 29th ACM International Conference on Information and Knowledge Management (CIKM 2020), Galway, Ireland, October 19-23, 2020*, volume 2699 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2020.

[17] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie J. Cai, James Wexler, Fernanda B. Viégas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *International Conference on Machine Learning, ICML 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 2673–2682. PMLR, 2018.

[18] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR 2015*, 2015.

[19] Morteza Mashayekhi and Robin Gras. Rule extraction from random forest: the RF+HC methods. In *Advances in Artificial Intelligence*, volume 9091 of *LNCS*, pages 223–237. Springer, 2015.

[20] Tamara T. Müller. Personalisable clinical decision support system for neurological diseases. Master's thesis, University of Cambridge, 2018.

[21] Myura Nagendran, Yang Chen, Christopher A Lovejoy, Anthony C Gordon, Matthieu Komorowski, Hugh Harvey, Eric J Topol, John P A Ioannidis, Gary S Collins, and Mahiben Maruthappu. Artificial intelligence versus clinicians: systematic review of design, reporting standards, and claims of deep learning studies. *BMJ*, 368, 2020.

[22] Forough Poursabzi-Sangdeh, Daniel G. Goldstein, Jake M. Hofman, Jennifer Wortman Vaughan, and Hanna M. Wallach. Manipulating and measuring model interpretability. *CoRR*, abs/1802.07810, 2018.

[23] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., 1993.

[24] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In *International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM, 2016.

[25] Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations. In *International Joint Conference on Artificial Intelligence, IJCAI 2017*, pages 2662–2670. ijcai.org, 2017.

[26] M. Sato and H. Tsukimoto. Rule extraction from neural networks via decision tree induction. In *International Joint Conference on Neural Networks (IJCNN)*, volume 3, pages 1870–1875. IEEE, 2001.

[27] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *IEEE International Conference on Computer Vision, ICCV*, pages 618–626. IEEE Computer Society, 2017.

[28] Oshani Seneviratne, Sabbir M. Rashid, Shruthi Chari, James P. McCusker, Kristin P. Bennett, James A. Hendler, and Deborah L. McGuinness. Knowledge integration for disease characterization: A breast cancer example. In *The Semantic Web - ISWC 2018 - 17th International Semantic Web Conference, Monterey, CA, USA, October 8-12, 2018, Proceedings, Part II*, volume 11137 of *LNCS*, pages 223–238. Springer, 2018.

[29] Claude E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27(3):379–423, 1948.

[30] Radwa El Shawi, Youssef Sherif, Mouaz H. Al-Mallah, and Sherif Sakr. Interpretability in healthcare A comparative study of local machine learning interpretability techniques. In *IEEE International Symposium on Computer-Based Medical Systems,CBMS*, pages 275–280. IEEE, 2019.

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

[31] Gregor Stiglic, Primoz Kocbek, Nino Fijacko, Marinka Zitnik, Katrien Verbert, and Leona Cilar. Interpretability of machine learning-based prediction models in healthcare. *WIREs Data Mining and Knowledge Discovery*, 10(5):e1379, 2020.

[32] Mike Wu, Michael C. Hughes, Sonali Parbhoo, Maurizio Zazzi, Volker Roth, and Finale Doshi-Velez. Beyond sparsity: Tree regularization of deep models for interpretability. In *AAAI Conference on Artificial Intelligence, (AAAI-18)*, pages 1670–1678. AAAI Press, 2018.

[33] Jan Ruben Zilke, Eneldo Loza Mencía, and Frederik Janssen. Deepred - rule extraction from deep neural networks. In *International Conference on Discovery Science, DS*, volume 9956 of *LNCS*, pages 457–473, 2016.

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Supplementary Materials

Abbreviations and Essential Definitions

Abbreviations

ML: Machine Learning, **DNN:** Deep Neural Network, **REM:** Rule Extraction Methodology, **REM-D:** Rule Extraction Methodology from Deep Neural Networks, **REM-T:** Rule Extraction Methodology from Trees

Essential Definitions

Interpretability: The degree to which "how" a model makes a prediction is understandable
Explainability: The degree to which "why" a model makes a prediction is understandable
Feature importance: Set of domain features (i.e. piece of data or evidence) that contributed the most to a prediction
Verifiability: The degree to which the biological relevance of a model can be determined
Simulatability: The ease of generating a prediction for an input by following the operation of a model
Reasoning: Applying rules towards making predictions
Accuracy: the degree of alignment between the predictions of an extracted ruleset model and ground truth
Fidelity: the degree of alignment between the predictions of an extracted ruleset model and the predictions of the original model they are extracted from
Efficiency: time and memory consumed to extract ruleset models
Comprehensibility: the number of rules in a ruleset and their average length

Rule Extraction Details for Cross-Modality Reasoning Case Study

Table 4 shows that the neural network model outperforms the decision tree model and the random forest model in terms of AUC, despite their high accuracy. The parameters used for the neural network model were outlined in the Method section. The random forest model used 50 trees each with maximum depth of 10 and half of the features considered at every split. The decision tree model also used a maximum depth of 10. Class weights were considered in both, random forest and decision tree models.

	decision tree		random forest		neural network	
fold	accuracy	AUC	accuracy	AUC	accuracy	AUC
0	0.897	0.718	0.918	0.639	0.903	0.811
1	0.879	0.723	0.918	0.609	0.906	0.813
2	0.846	0.666	0.929	0.617	0.870	0.804
3	0.876	0.698	0.905	0.589	0.908	0.731
4	0.885	0.624	0.914	0.578	0.825	0.78
average	**0.877**	**0.686**	**0.917**	**0.606**	**0.882**	**0.788**

Table 4: A performance comparison between Decision Tree, Random Forest and Neural Network with 1004 mRNA expressions as input and ILC/IDC as output.

Table 5 displays the details of rule extraction using REM-D for cross-modality scenario reported in the Results and Method sections.

fold	nn accuracy	REM-D accuracy	REM-D fidelity	REM-D duration (sec)	REM-D memory (MB)	size of ruleset	rules average length
0	0.903	0.888	0.903	143.792	573.231	155	4.3
1	0.906	0.885	0.891	32.06	146.538	6	1.8
2	0.870	0.867	0.861	143.881	642.576	137	7.7
3	0.908	0.902	0.929	59.239	227.721	18	2.4
4	0.825	0.888	0.837	157.059	630.239	240	6.6
average	**0.882**	**0.886**	**0.884**	**107.206**	**444.061**	**111.2**	**4.6**

Table 5: Rule extraction from neural network with 1004 mRNA expressions as input and ILC/IDC as output.

bioRxiv preprint doi: https://doi.org/10.1101/2021.01.22.427799; this version posted January 24, 2021. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Rule Extraction Details for Multi-Modality Reasoning Case Study

Tables 6 and 7 show the details of rule extraction using REM-D and REM-T from mRNA and clinical data as outlined in the multi-modality case study, while Table 8 displays the details of their integration.

fold	nn accuracy	REM-D accuracy	REM-D fidelity	REM-D duration (sec)	REM-D memory (MB)	size of ruleset	rules average length
0	0.950	0.927	0.932	103.31	699.958	178	5.1
1	0.955	0.942	0.952	101.202	349.951	103	5.1
2	0.955	0.949	0.944	100.714	466.281	182	5.4
3	0.957	0.896	0.919	37.186	301.164	14	2.9
4	0.97	0.889	0.884	37.762	117.204	17	3.0
average	**0.957**	**0.921**	**0.926**	**76.035**	**386.912**	**99**	**4.3**

Table 6: Rule extraction from neural network with 1000 mRNA expressions as input and ER+/ER- as output.

fold	REM-T accuracy	size of ruleset	rules average length
0	0.768	28	4.6
1	0.788	30	4.7
2	0.803	29	4.6
3	0.848	29	4.6
4	0.841	30	4.4
average	**0.81**	**29.2**	**4.6**

Table 7: Rule extraction from decision tree with 13 clinical variables as input and ER+/ER- as output.

fold	REM-D + REM-T accuracy	size of ruleset	rules average length
0	0.927	206	5.0
1	0.952	133	5.0
2	0.952	211	5.3
3	0.922	43	4.0
4	0.911	47	3.9
average	**0.933**	**128**	**4.7**

Table 8: Rule integration from neural network and decision tree.