Proceedings of the Twenty-Fifth Innovative Applications of Artificial Intelligence Conference

Early Prediction of Coronary Artery Calcification Levels Using Machine Learning

Sriraam Natarajan[#], Kristian Kersting[*#], Edward Ip[#], David R. Jacobs, Jr[**], Jeffrey Carr[#]
[#]Wake Forest University School of Medicine, USA
[*]Fraunhofer IAIS, Germany
[**]University of Minnesota, USA

Abstract

Coronary heart disease (CHD) is a major cause of death worldwide. In the U.S. CHD is responsible for approximated 1 in every 6 deaths with a coronary event occurring every 25 seconds and about 1 death every minute based on data current to 2007. Although a multitude of cardiovascular risks factors have been identified, CHD actually reflects complex interactions of these factors over time. Today's datasets from longitudinal studies offer great promise to uncover these interactions but also pose enormous analytical problems due to typically large amount of both discrete and continuous measurements and risk factors with potential long-range interactions over time. Our investigation demonstrates that a statistical relational analysis of longitudinal data can easily uncover complex interactions of risks factors and actually predict future coronary artery calcification (CAC) levels — an indicator of the risk of CHD present subclinically in an individual — significantly better than traditional non-relational approaches. The uncovered long-range interactions between risk factors conform to existing clinical knowledge and are successful in identifying risk factors at the early adult stage. This may contribute to monitoring young adults via smartphones and to designing patient-specific treatments in young adults to mitigate their risk later.

Introduction

Heart disease and stroke – cardiovascular diseases, generally – encumber society with enormous costs. According to the World Heart Federation [1], cardiovascular disease costs the European Union € 169 billion in 2003 and the USA about $400 billion in direct and indirect annual costs.

One major cardiovascular disease is coronary heart disease (CHD). It is reported to be a major cause of morbidity and death in adults through heart attacks or acute myocardial infarctions (AMI). CHD is a condition which includes plaque build up inside the coronary arteries, i.e., atherosclerosis. Atherosclerosis is a disease process that begins in childhood, eventually resulting in clinical events later in life. The factors that determine development and progression of CHD are in large part established; however, the causes are very closely related with risk factors present in youth. Early detection of risks will help in designing effective treatments targeted at youth in order to prevent cardiovascular events in adulthood and to dramatically reduce the costs associated with cardiovascaular dieases.

Our major contribution is to demonstrate the impact of AI and machine learning on CHD research and the potential for developing treatment plans. We show that relationships between the measured risk factors, the development of CHD, and overall plaque burden can be automatically extracted and understood. As the cohort ages and sufficient clinical events occur, this work will allow us to apply these methods to clinical events such as AMI and heart failure. Specifically, we propose to use the longitudinal data collected from the Coronary Artery Risk Developments in Young Adults (CARDIA)[2] study over different years to automatically estimate models using machine learning techniques for predicting the Coronary Artery Calcification (CAC) amounts, a measure of subclinical CHD, at year 20 given the measurements from the previous years. This longitudinal study began in $1985-86$ and measured risk factors in different years (2,5,7,10,15,20) respectively. Several vital factors such as *body mass index (bmi)*, *cholesterol*-levels, *blood pressure* and *exercise* level are measured along with *family history, medical history, nutrient intake, obesity questions, pyschosocial, pulmonary function* etc. Using the predictions of the CAC levels we can predict cardiovascular events such as heart attacks. This in turn allows us to enable pro-active treatment planning for the high-risk patients i.e., identify young adult patients who are potentially at high-risk to cardiovascular events and design patient-specific treatments that would mitigate the risks. And this could even be ported to one people's laptops and/or smartphones, illustrating that AI could indeed empower people to take control of their CV health.

Specifically, we use Statistical Relational Learning (SRL) (Getoor and Taskar 2007) algorithms for predicting CAC-levels in year 20 (corresponding to year 2005 when the patients were between 38 and 50 years old) given the measurements from all the previous years. SRL approaches, unlike what is traditionally done in statistical learning, seek to avoid explicit state enumeration as, through a symbolic

[1]See http://www.world-heart-federation.org/ cardiovascular-health/global-facts-map/ economic-impact/ and references in there.

[2]http://www.cardia.dopm.uab.edu/

representation of states. The advantage of these models is that they can succinctly represent probabilistic dependencies among the attributes of different related objects leading to a compact representation of learned models that allow for sharing of parameters between similar objects. Given that the CARDIA data is highly relational (multiple measurements are performed over examinations for each participant) and temporal, we use SRL methods to learn to predict CAC-levels. We use a de-identified version of the data set for methodological development.

More precisely, we use two kinds of SRL algorithms for this task – *Relational Probability Trees* (RPT) (Neville et al. 2003) and a more recently popular *Relational Functional Gradient Boosting* (RFGB) (Natarajan et al. 2012) approach. RPTs upgrade the attribute-value representation used within classical classification trees. The RFGB approach, on the other hand, involves learning a set of regression trees, each of which represents a potential function. The functional gradient approach has been found to give state of the art results in many relational problems and we employ the same for CAC prediction. We use a sub-set of measurements from the CARDIA data set and predict the CAC-levels. We compared the SRL algorithms against propositional machine learning algorithms and demonstrated the superiority of the SRL algorithms. The learned models were also verified by our medical expert and the results conform to known medical risks. The results also provided a few insights about the relationships between risk factors and age of the individual. Identifying risk factors such as cholesterol level in young adulthood has potential to enable both the physician and the subject to devise a personalized plan to optimize it. Keeping track of these risk factors in young adulthood will prevent serious cardio-vascular events in their late adulthood.

In summary, we first present a very important real-world problem: that of predicting cardio-vascular risks in adults given their risk factors early in the adult stage. This problem has a significant potential impact in the design of preventive personalized treatments for adults. Second, this problem is addressed using ML techniques. These techniques were developed recently within the SRL community and we adapt these algorithms to our particular task and present the algorithms from the perspective of this task. Third, the introduction of the application to the AI community is important. Fourth, long-range interactions between the risk factors are mined effectively in our approach. For example, being a smoker in young adulthood and having a low hdl-cholestrol level in mid-adulthood could have a negative impact in the older adults. Such dependencies are extracted by using time as an explicit parameter in our models. Finally, the proposed approaches are compared against state-of-the-art machine learning techniques on the task of predicting CAC-levels in patients in their adulthood.

Methodology

Before explaining how to adapt the CARDIA data to the relational setting, we will justify and detail our relational methodology.

The Need for Relational Models

Are relational models really beneficial? Could we also use propositional models? As we show, relational approaches are able to comprehensively outperform standard machine learning and data mining approaches. Beyond this, there are several justifications for adopting statistical relational analyses. First, the data consists of several diverse features (e.g., demographics, psychosocial, family history, dietary habits) that interact with each other in many complex ways making it *relational*. Without extensive feature engineering, it is difficult — if not impossible — to apply propositional approaches to such structured domains. Second, the data was collected as part of a longitudinal study, i.e., over many different time periods such as 0, 5, 10, years etc., making it *temporal*. Third, like most data sets from biomedical applications, it contains missing values i.e., all data are not collected for all individuals. Fourth, the nature of SRL algorithms allow for more complex interactions between features. Finally, the learned models can be generalized across different sub-groups of participants and across different studies themselves. And, the relational models are very easily interpretable and hence enable the physician and policy-maker to identify treatable risk factors and plan preventative treatments

While SRL methods seem appropriate, this data poses a few challenges for SRL, to necessitate some preprocessing.

(1) Since the data is longitudinal, there are multiple measurements of many of the risk factors over different years of study. Hence time has to be incorporated into the model. To do so, the features are treated as fluents with time being the last argument. For instance, $weight(X, W, T)$ would refer to person X having weight W at time T. **(2)** CAC-levels of the participants are negligible (and often actually unobserved) in early years. This prevents us from using standard Dynamic Bayesian Network or HMMs; the values are nearly always zero in the initial years, being non-zero only in 10% at year 15 and 18% at year 20. **(3)** The input data consists of mainly continuous values. SRL methods use predicate logic where attributes are binary. In the case of features such as *cholesterol* level, *ldl*, *bmi*, we discretized them into bins based on domain knowledge. This is one of the key lessons learned: *using the domain expert's knowledge (for discretization in our case) makes it possible to learn very highly predictive models in real problems*. **(4)** The cohort decreased over the years. There were a number of participants who did not appear for a certain number of years and returned for others. We did not try to normalize the data set by removing all the missing participants or replacing them with the most commonly observed value. Instead, we allowed the values to be missing. The only case where we dropped the participants from the data base was when they were not present in year 20 where we predict the CAC-levels. This is to say that we are not considering the problem to be a semi-supervised learning problem but treat it as a strictly supervised learning one. **(5)** Recall the goal of the study is to identify the effect of the factors in early adulthood on cardiovascular risks in middle-aged adults. The algorithm should be allowed to search through all the risk factors in all the years for predicting CAC-levels. This implies that the data must not be altered or tailored in any form. In this work, we did not make any modifications

to the data except for the discretizations mentioned earlier. As we show, our methods are very successful in identifying long-range correlations. One of the biggest lessons learned from this work is that risk factors between the ages of 18 through 30 are very significant for CAC-level prediction at age 38 to 50.

However, which SRL approach should we use?

Relational Gradient Boosting

One of the most important challenges in SRL is learning the structure of the models, i.e., the weighted relational rules. This problem has received much attention lately. Most approaches follow a traditional greedy hill-climbing search: first obtain the candidate rules/clauses, score them, i.e., learn the weights, and select the best candidate. The temporal nature of our task at hand makes it difficult to use these approaches. Therefore, we use a boosting approach based on functional gradients recently proposed that learns the structure and parameters simultaneously (Natarajan et al. 2012). It was proven successful in several classical SRL domains and achieves state-of-the art performances. Also, it easily — as we will show — accounts for the temporal aspects of CAC-level prediction.

Functional gradient methods have been used previously to train conditional random fields (CRF) Dietterich et al. (2004) and their relational extensions (TILDE-CRF) (Gutmann and Kersting 2006). Assume that the training examples are of the form $(\mathbf{x}_i, y_i)$ for $i = 1, ..., N$ and $y_i \in \{1, ..., K\}$. We use $\mathbf{x}$ to denote the vector of features. The goal is to fit a model $P(y|\mathbf{x}) \propto e^{\psi(y,\mathbf{x})}$. The potentials are trained using Friedman's (2001) gradient-tree boosting algorithm where the potential functions are represented by sums of regression trees that are grown stage-wise. More formally, functional gradient ascent starts with an initial potential ψ_0 and iteratively adds gradients Δ_i.After m iterations, the potential is given by $\psi_m = \psi_0 + \Delta_1 + ... + \Delta_m$. Here, Δ_m is the functional gradient at episode m and is

$$\Delta_m = \eta_m \times E_{x,y}[\partial / \partial \psi_{m-1} log\, P(y|x; \psi_{m-1})] \quad (1)$$

where η_m is the learning rate. Dietterich *et al.* suggested evaluating the gradient at every position in every training example and fitting a regression tree to these derived examples i.e., fit a regression tree h_m on the training examples $[(x_i, y_i), \Delta_m(y_i; x_i)]$.

Let us denote the CAC-level as y and for ease of explanation assume that it is binary valued (i.e., present vs absent). Let us denote all the other variables measured over the different years as $\mathbf{x}$. Our aim is to learn $P(y|\mathbf{x})$ where, $P(y|\mathbf{x}) = e^{\psi(y;\mathbf{x})} / \sum_y e^{\psi(y;\mathbf{x})}$ Note that in the functional gradient presented in Equation 1, the expectation $E_{x,y}[..]$ cannot be computed as the joint distribution $P(\mathbf{x}, \mathbf{y})$ is unknown. Instead of computing the functional gradients over the potential function, they are instead computed for each training example i given as $\langle \mathbf{x_i}, y_i \rangle$. Now this set of local gradients form a set of training examples for the gradient at stage m. The main idea in the gradient-tree boosting is to fit a regression-tree on the training examples at each gradient step. We replace the propositional regression trees with relational regression trees.

The functional gradient with respect to $\psi(y_i = 1; \mathbf{x_i})$ of the likelihood for each example $\langle y_i, \mathbf{x}_i \rangle$ can be shown to be: $\frac{\partial \log P(y_i;\mathbf{x_i})}{\partial \psi(y_i=1;\mathbf{x_i})} = I(y_i = 1; \mathbf{x_i})$ - $P(y_i = 1; \mathbf{x_i})$, where I is the indicator function that is 1 if $y_i = 1$ and 0 otherwise. The expression is simply the adjustment required to match the predicted probability with the true label of the example. If the example is positive (i.e., if the participant has significant CAC-level in year 20) and the predicted probability is less than 1, this gradient is positive indicating that the predicted probability should move towards 1. Conversely, if the example is negative and the predicted probability is greater than 0, the gradient is negative driving the value the other way.

Now, to fit the gradient function for every training example, we use *Relational Regression Trees* (RRTs) (Blockeel and Raedt 1998).At a fairly high level, the learning of RRT proceeds as follows: The learning algorithm starts with an empty tree and repeatedly searches for the best test for a node according to some splitting criterion such as weighted variance. Next, the examples in the node are split into *success* and *failure* according to the test. For each split, the procedure is recursively applied further obtaining subtrees for the splits. We use weighted variance on the examples as the test criterion. In our method, we use a small depth limit (of at most 3) to terminate the search. In the leaves, the average regression values are computed.

The *key* idea underlying the present work is to represent the distribution over CAC-levels as a set of RRTs on the features. When learning to predict the CAC-levels in year 20, we use the data collected from all the previous years. We ignore the CAC-levels that are present for some individuals at year 15 since we are interested in planning preventive treatments in early adulthood based on other risk factors. We bear in mind that CAC rarely regresses from present to absent or from a higher level to a lower level. These trees are learned such that at each iteration the new set of RRTs aim to maximize the likelihood of the distributions w.r.t ψ. When computing $P(cac(X)|\mathbf{f}(X))$ for a particular patient X, given the feature set $\mathbf{f}$, each branch in each tree is considered to determine the branches that are satisfied for that particular grounding (x) and their corresponding regression values are added to the potential ψ.

To investigate the usefulness of other relational learners, we also considered Relational Probability Trees (Neville et al. 2003). We modified the RPT learning algorithm to learn a regression tree similar to TILDE to predict positive examples and turn the regression values in the leaves into probabilities by exponentiating the regression value and normalizing them. We modified TILDE to automatically include aggregate functions such as count, mode, max, mean etc. while searching for the next node to add to the tree. Also, the regression tree learner can use conjunctions of predicates in the inner nodes as against a single test by the traditional RPT learner. This modification has been shown to have better performance than RPTs (Natarajan et al. 2012) and hence we employ this modified RPT learner in our experiments.

Adapting the CARDIA Data

The CARDIA Study examines the development and determinants of clinical and subclinical cardiovascular disease and its risk factors. It began in $1985-6$ (Year 0) with a group of 5115 men and women whose age were between 18-30 years from 4 centers : Birmingham, Chicago, Minneapolis, and Oakland. The same participants were asked to participate in follow-up examinations during $87-88$ (Year 2), $90-91$ (Year 5), $92-93$ (Year 7), $95-96$ (Year 10), $2000-2001$ (Year 15), and $05-06$ (Year 20). Data have been collected on a variety of factors believed to be related to heart disease. This rich data set provides a valuable opportunity to identify risk factors in young adults that could cause serious cardiovascular issues in their adulthood. This in turn will allow physicians and policy makers to develop patient-specific preventive treatments.

We used known risk factors such as *age, sex, cholesterol, bmi, glucose, hdl level and ldl level of cholestrol, exercise, trig level, systolic bp and diastolic bp* that are measured between years 0 and 20 over the patients. Our goal is to predict if the CAC-levels of the patients are above 0 for year 20 given the above mentioned factors. Any CAC-level over 0 indicates the presence of advanced coronary atheroma and elevated risk for future CHD and needs to be monitored. So, we are in a binary classification setting of predicting 0 vs non-0 CAC levels. Also, most of the population had CAC-level of 0 (less than 20% of subjects had significant CAC-levels) in year 20. Hence there is a huge skew in the data.

We converted the data set into predicate logic, see e.g. (De Raedt 2008) for an introduction. The first argument of every predicate is the ID of the person and the last argument is the year of measurement. It is possible for our algorithm to search at the level of the variables or ground the variable to a constant while searching for the next predicate to add to the tree. For example, we could use some values such as "never smoked", "quit smoking" etc. directly in the learned model and in other cases, use variables in the node. We are able to learn at different levels of variablizations in the model.

The risk factors, however, are continuous variables. For instance, *ldl, hdl, glucose, bmi, dbp, bp* etc. all take real numbered values with different ranges. Many methods exist that can discretize the data and/or directly operate on the continuous data. While automatically discretizing the features based on the data is preferred, some of these risk factors have been analyzed by the medical community for decades and the thresholds have been identified. For example, a *bmi* of less than 16 is *severely underweight*, greater than 40 is *extremely obese* etc. Hence, we used inputs from a domain expert to identify the thresholds for discretizing the numeric features and these are presented in Table 1. A particular value of the measurements can fall into only one of these bins.

We also included the difference between the two successive measurements as input features. This represents the "change" in risk factor for the subject. For the boosting algorithm (RFGB), we used the preset parameter of number of trees, namely 20. The tree-depth of the trees was set to 3 and hence we preferred a reasonably large set of small trees. As mentioned above, we allowed the algorithm to construct the aggregators on the fly. We compare against learning a single tree(RPT) of depth 10. This is due to the fact that every path from root to leaf indicates an interaction between the risk factors and our domain expert indicated that 10 should be the upper limit of the interactions. We also compared our algorithms against various algorithms and parameter settings using the weka package and report the results. Hence, we propositionalized our features by creating one feature for every measurement at every year. We included the change (difference between measurements in successive years) features for the propositional data set as well.

Feature	Thresholds
cholestrol	70, 100, 150, 200, 250, 300, 400
dbp	0, 30, 50, 70, 90, 100, 150
glucose	0, 50, 100, 200, 300, 400
hdl	10, 30, 50, 70, 100,120,200
ldl	0,50,100,150,200,400
trig	0,25,50,100,300,1000,3000
bmi	0,16,18.5,25,30,35,40,100

Classifier	Parameters
J48	C 0.25 M 2
SVM	C 1.0, L 0.01, P 1E12, N 0, V 1, W 1 Poly Kernel
AdaBoost	P 100, S 1, I 10
Bagging	P 100, S 1, I 10, M 2, V 0.001, N 3, S 1, L -1
Logistic	R 1.0E-8, M -1

Table 1: **(Top)**Domain expert's discretization of some of the input features. **(Bottom)** Parameters of the propositional classifiers

The best parameters determined by cross-validation and used for the propositional classifiers are presented in Table 1(top). For J48, C was set as 0.25 while the minimum number of examples is 2. For support vector machines, C was set to be 1.0, the rescale parameter was set to 0.01 and the poly kernel was used. For Logistic regression, R was set to 1.0E-8 and allowed to converge (instead of maximum number of iterations). In bagging, we used 10 iterations with a bag size of 100 and no depth limit. We performed 5-fold cross validation for evaluation.

Predicting CAC Levels

Our intention here is to investigate the benefits and the quality of relational models for CAC level prediction.

Comparison with Propositional Learners: We now present the results of learning to predict CAC-levels using our algorithms and the standard ML techniques. A full test set has a very large skew towards one class and hence the accuracies can be very inflated. Hence in the test set, we sampled twice the number of negatives as positives. Recall that the positive class would mean that the CAC-level of the subject in year 20 is significant (i.e., greater than 0). Table 2 compares the results of boosting (*RFGB*) and *RPT* — against decision-trees (*J48*), *SVM*, *AdaBoost*, *Bagging*, *Logistic Regression* (LR) and Naive Bayes (*NB*).

A key property of many real-world data sets is a significantly increased number of negative examples relative to positive examples. This is also seen in our data set since most

Algorithm	AUC-ROC
J48	0.609 ± 0.04
SVM	0.5 ± 0.0
AdaBoost	0.528 ± 0.02
Bagging	0.563 ± 0.02
LR	0.605 ± 0.02
NB	0.603 ± 0.03
RPT	0.778 ± 0.02
RFGB	0.819 ± 0.01

Table 2: Results on CARDIA data set. Area under ROC curves have been presented.

CAC-levels are zero and hence the number of negatives can be order of magnitude more than the number of positives. In these cases, simply measuring accuracy or conditional log-likelihood (CLL) over the entire data set can be misleading. It can be shown easily that predicting all the examples as the majority class (when the number of examples in one class are far greater than the other) can have a very good CLL value, but a very low AUC-ROC or AUC-PR value (nearly 0). For more details of PR and ROC curves, we refer the reader to (Davis and Goadrich 2006).

The AUC-ROC results presented clearly show that the SRL approaches dominate the propositional ones. Most of the standard algorithms classify nearly all the examples as negative and as mentioned earlier, presenting accuracies can be misleading. We chose to present AUC-ROC instead. SVM and AdaBoost classify all examples as negative while Bagging, LR, Naive Bayes and J48 classify a very small number of examples (nearly 5% of positive examples correctly). In contrast, the SRL approaches have a smoother classification performance and hence have a higher AUC-ROC with RFGB having the best ROC.

We present the Precision Recall curves for the SRL algorithms in Figure $1.d$. We did not include the other algorithms since their PR values were very low. The boosting approach has a better performance particularly in the medically-relevant high recall region. Evaluating precision at high recalls assesses an algorithm's ability to predict while disallowing many false negatives, which is the critical component to a good screening tool. In the case of predicting CAC levels, a false negative means categorizing a patient as "low-risk" who might potentially go on to have a heart attack, a costly outcome we wish to avoid.

The effect of the number of trees on the performance of RFGB is presented in Figure $1.a$. We have presented both the AUC-ROC and AUC-PR values as a function of the number of trees. As with the previous case, the results are averaged over 5 runs. As the number of trees increase, there is an increase in the performance of RFGB. Also, it can be noted that beyond a certain number of trees (in our case 10), there is not a significant increase in the performance. When the number of trees is close to 30, there is a slight dip in the performance of the algorithm. This could be due to overfitting. When we look at the trees closer, it appears that with larger number of trees (say 30), the last few trees are picking up random correlations in the data (though the regression values in the leaves are quite low). Figure $1.b$ presents the effect of the depth of the tree when learning a single tree (i.e., RPT). It appears that the performance of the algorithm stablizes around a depth of 5. Increasing beyond 5 does not have a statistically significant impact on the performance showing that interactions between 5 risk factors is sufficient to predict the CAC-levels.

Assessment of the Results: The results were verified by our *radiologist*, and are very interesting from a medical perspective for several reasons: First, as our last set of experiments show, the risk of CAC levels in later years is mostly indicated by risk factors in early years (ages 25 through 40). This is very significant from the point of view of the CARDIA study since the goal is to identify risk factors in early adult stage so as to prevent cardio-vascular issues in late adulthood. Second, the learned tree conforms to some known or hypothesized facts. For instance, it is believed that females are less prone to cardiovascular issues than males. The tree identifies sex as the top splitting criterion. Similarly, in men, it is believed that the ldl and hdl levels are very predictive and the tree confirms this. Third, the tree also identifies complex interaction between risk factors at different years. For instance – (i) smoking in year 5 interacts with cholesterol level in later years in the case of females, and (ii) the triglyceride level in year 5 interacts with the cholesterol level in year 7 for males. Finally, the structure of the tree could enable the physician and policy-maker to identify treatable risk factors.

Prediction Based on Early Adulthood Data only: We performed three additional experiments.

1 In the first setting, we repeated the earlier experiment with one major change. Instead of considering all the risk factors at all years, we considered the measurements only till year 10 i.e., only the risk factors from young adulthood. We aim to predict the CAC-level in year 20. The average AUC-ROC are 0.779 ± 0.01 and are not significantly different from the ones learned using the entire data set. This confirms our hypothesis that the risk factors in younger age are responsible for the cardiovascular risks in older age.

2 To further understand the impact of the different years, we ran the RFGB algorithm using data from individual years only i.e., the first set of trees were learned with only year 0, the second with only year 5 and so on. The goal is again to predict CAC-level in year 20 given these individual years. The results are presented in Figure $1.c$ (solid line) and again they were averaged over 5 runs for each year. As can be seen, year 0 has the highest AUC-ROC compared to the other years. This is a very significant result. This further shows that the factors of a subject between his/her ages $18 - 30$ determine the risks in later years. This validates the observations made by Loria et al. (Loria, Liu, and et al 2007) where individual correlations between risk factors at different years and CAC-level at year 15 are measured to show that year 0 risk factors are as informative as later years. Of course, in the current experiment, we did not include things such as changes in the behavior (for example, how much one reduced the cholesterol level) and it is interesting to understand how lifestyle changes of a person in early adulthood can affect the cardio-vascular

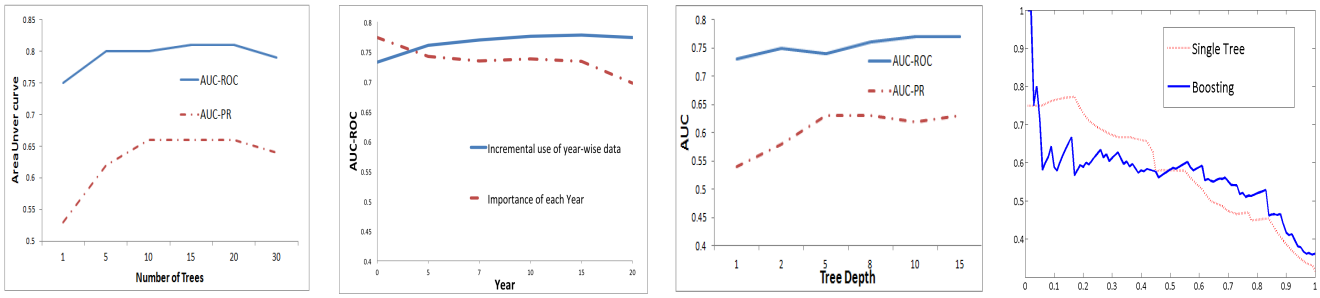

Figure 1: **(a)** Effect of number of trees in performance of RFGB. **(b)** Effect of the depth of the tree in the performance of a single RRT. **(c)** The impact of the measurements in different years in the CAC-level at year 20. **(d)** PR curves comparing the SRL algorithms.

risks in later years.

3 The final experiment aims to compute the change in predictive performance of the algorithm with increase in data. We first used only year 0 data and learned a single tree. Now using this tree, we learned the second tree using year 5 data and so on. So the goal is to see how AUC-PR changes with adding more observations in future yeas and can be seen as the progress of the risk factor over time. The results are presented in Figure 1.*c* (dashed). As expected from the previous experiment, year 0 has a big leap and then adding individual years increases performance till year 7 and then plateaus beyond that. This is again a significant result. Our initial results show that beyond ages $25 - 37$ of a person, there is not much significant information from the risk factors.

Prediction using only socio-economic data: In addition, we were also interested in finding how non-standard risk factors such as family history, daily habits and drug use can affect the CAC-levels i.e., can we determine if these factors are as predictive as the ones considered above? These diverse set of features included the age of the children, whether the participant owns or rents a home, their employment status, salary range, their food habits, their smoking and alcohol history etc. There were about 200 such questions that were considered. Initial experiments showed that we were able to predict the CAC-levels reasonably well and in fact with comparable statistical performance to that of the clinical risk factors. While the results are preliminary, they reveal striking socio-economic impacts on the health state of the population, factors that have long been speculated on, but which can be conclusively quantified.

Conclusion

Coronary heart disease (CHD) kills millions of people each year. The broadening availability of longitudinal studies and electronic medical records presents both opportunities and challenges to apply AI techniques to improve CHD treatment. We discussed the important problem of identifying risk factors in young adults that can lead to cardiovascular issues in their late adulthood. We addressed the specific problem of uncovering interactions among risk factors and of using them for predicting CAC levels in adults given the risk factor measurements of their youth. Our experimental results indicate that the risk factors from the early adulthood of the subjects seem to be the most important ones in predicting risks at later years.

Motivated by the initial success of our work, we plan to pursue research in several different directions. First, we plan to include all the collected features for training the models. This will allow one to identify complex relationships between different types of features such as demographics and psychosocial etc. Second, while the boosted set of trees have high predictive accuracy, they may not necessarily be easy to interpret by physicians. Hence our goal is to convert the set of trees into a single tree. Third, the current data set does not have the notion of "events" i.e., there are no records of cardiovascular issues such as heart attacks in the current data set (year 20). Recently, these events have started to appear. It will be extremely important to directly predict the events instead of the surrogates such as CAC levels. The ultimate goal is to make a tool for personalized prevention of heart disease using ideas from machine learning that could potentially be used for all young adults to get a precise estimate of their future risk of disease.

Acknowledgements The authors gratefully acknowledge Santiago Saldana and Jose Picado for their implementation support.

References

Blockeel, H., and Raedt, L. D. 1998. Top-down induction of first-order logical decision trees. *Artificial Intelligence* 101:285–297.

Davis, J., and Goadrich, M. 2006. The relationship between precision-recall and roc curves. In *ICML*.

De Raedt, L. 2008. *Logical and Relational Learning*. Springer.

Dietterich, T.; Ashenfelter, A.; and Bulatov, Y. 2004. Training conditional random fields via gradient tree boosting. In *ICML*.

Friedman, J. H. 2001. Greedy function approximation: A gradient boosting machine. *Annals of Statistics* 1189–1232.

Getoor, L., and Taskar, B. 2007. *Introduction to Statistical Relational Learning*. MIT Press.

Gutmann, B., and Kersting, K. 2006. TildeCRF: Conditional Random Fields for Logical sequences. In *ECML*.

Loria, C.; Liu, K.; and et al, C. E. L. 2007. Early adult risk factor levels and subsequent coronary artery calcification - the cardia study. *Journal of the American College of Cardiology* 49.

Natarajan, S.; Khot, T.; Kersting, K.; Guttmann, B.; and Shavlik, J. 2012. Gradient-based boosting for statistical relational learning: The relational dependency network case. *Machine Learning*.

Neville, J.; Jensen, D.; Friedland, L.; and Hay, M. 2003. Learning Relational Probability trees. In *KDD*.