Patient-Specific Modeling with Personalized Decision Paths

Adriana Johnson, MS[1], Gregory F. Cooper, MD, PhD[1, 2], Shyam Visweswaran, MD, PhD[1, 2]
[1]Department of Biomedical Informatics, University of Pittsburgh, Pittsburgh, PA
[2]Intelligent Systems Program, University of Pittsburgh, Pittsburgh, PA

Abstract

Predictive models can be useful in predicting patient outcomes under uncertainty. Many algorithms employ "population" methods, which optimize a single model to perform well on average over an entire population, but the model may perform poorly on some patients. Personalized methods optimize predictive performance for each patient by tailoring the model to the individual. We present a new personalized method based on decision trees: the Personalized Decision Path using a Bayesian score (PDP-Bay). Performance on eight synthetic, genomic, and clinical datasets was compared to that of decision trees and a previously described personalized decision path method in terms of area under the ROC curve (AUC) and expected calibration error (ECE). Model complexity was measured by average path length. The PDP-Bay model outperformed the decision tree in terms of both AUC and ECE. The results support the conclusion that personalization may achieve better predictive performance and produce simpler models than population approaches.

Introduction

A central challenge in clinical medicine is accurate prediction of risk of developing disease and of outcomes in those with existing disease. For example, clinicians seek to assess the risk of developing diseases such as chronic pancreatitis and to estimate the chance of serious morbidity and mortality in patients with conditions like heart failure and community acquired pneumonia (1). Predictive models have the potential to provide high-quality predictions that guide clinical decision making to improve patient care.

Machine learning methods produce predictive models for specific patient outcomes by identifying patterns in large clinical and biomedical datasets. In the standard paradigm, a single model is produced from a collection of patient cases, and then the model is applied to all future patients to predict the outcome of interest. Such a *population model* is optimized to perform well on average on an entire population of future patients. This approach does not ensure optimal performance for every member of the population, however.

An alternate approach when seeking to make a prediction for a specific patient of interest (or *test case*) is to construct a model tailored to the specific features of that patient. This approach allows a method to use information about the test case to guide optimization of the model, resulting in a *personalized model*. In contrast to the population model, the personalized model is optimized to perform well for the test case of interest. This optimization is achieved by utilizing information about the test case to guide the search for the best model.

Personalized methods may therefore result in improved predictive performance because every test case has a tailored predictive model. Additionally, they may produce more concise, simplified models, as each model must only predict well for a single test case, not for all possible cases. These concise models may be easier for clinician-users to interpret, leading to better understanding of both the strengths and limitations of the models. Also, a personalized modeling approach may better identify rarely occurring but highly predictive features that might otherwise be overlooked in a population model, allowing personalized models to not only produce better predictions but also provide insight into personalized risk factors.

An example of a classical personalization method is k-Nearest-Neighbor (*k*NN) (2). In this approach, the *k* most similar samples to the test case are selected from the training dataset, and an average or majority vote is used to make a prediction for the test case. *k*NN does not construct a model, per se, but uses the matched cases directly to predict an outcome. It is possible, however, to apply machine learning to matched cases to construct a personalized model. For example, decision trees (a popular type of population model in biomedicine) can be personalized. One approach to personalizing the decision tree is to use the cases in the training dataset that are most similar to the test case to construct just one path in the tree. Such personalized-decision-path approaches have been shown to improve predictive performance and generate simpler models than the standard decision tree (3–5).

In this paper, we present a new method called the Personalized Decision Path that uses a Bayesian score (PDP-Bay). We hypothesized that the new method will perform better than population decision tree approaches and a previously described personalized decision path method called the Personalized Decision Path that uses an Entropy score (PDP-Ent), and we evaluated this hypothesis on a range of synthetic and real datasets.

Background

In this section, we provide brief descriptions of decision trees that are population models and decision paths that are examples of personalized models.

Decision Trees. The decision tree model consists of interior nodes that represent predictor variables and (terminal) leaf nodes that represent parameters of the predictive probability distribution of the target variable, such as a clinical outcome (6). We differentiate between a variable and a feature; a feature is a variable – value pair. For example, if variable *V* denotes a history of angina and takes values absent and present, then *V* = absent and *V* = present are features. Note that there are two distinct features that correspond to a single binary variable *V,* and a patient will have only one of the two features for *V*. A path in the tree from the root node to a leaf node represents a conjunction of features, and the parameters in the leaf node are estimated from the known outcomes of cases in the training set whose features match the features in the corresponding path.

To perform inference for a test case using a decision tree, a path in the tree is identified such that all of the features in the path match the features of the test case, and the parameters in the leaf node specify the probability distribution of the target variable. Each test patient will have only one applicable path in the tree.

As it is computationally intractable to exhaustively search the space of all possible entire decision trees, decision tree methods rely on heuristic search to derive a locally optimal model from a dataset. Many decision tree methods use greedy search (called recursive splitting) to optimize a specified scoring criterion. At each step, every candidate variable is scored using the criterion, and the best-scoring variable is added to the tree, splitting the variable space into distinct branches in the process. The next best-scoring variable is then identified; this process is repeated until a stopping condition is met. Decision tree methods differ in the scoring criterion used: the Classification And Regression Tree (CART) uses the Gini index (7), while the Interactive Dichotomizer 3 (ID3) and the C4.5 use entropy (6).

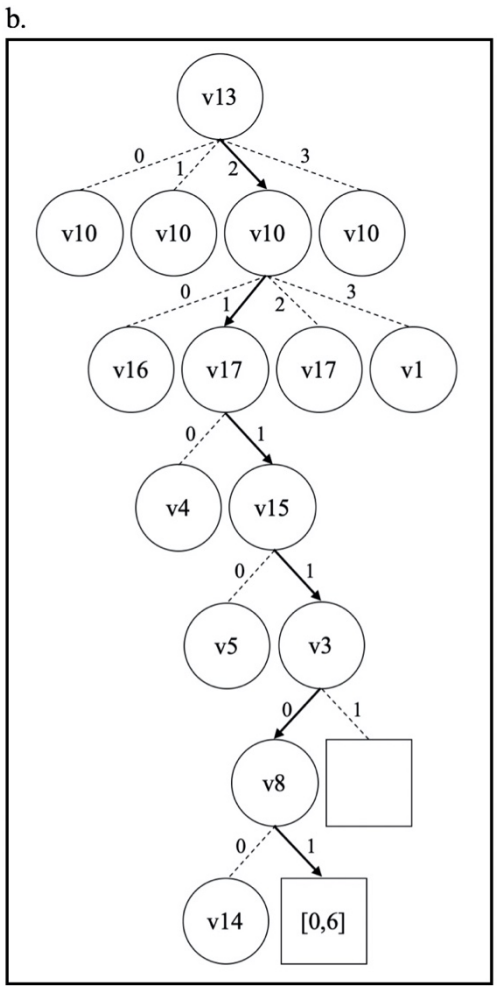
Decision Paths. A decision path model is also derived from a dataset using greedy search and optimizes a scoring criterion. Unlike a decision tree, the decision path model consists of a single path (rather than a collection of paths) whose features are shared by the test case. The search proceeds by extending the path by appending one feature (rather than a variable as in the tree) at a time from the test case that optimizes the scoring criterion.

Figure 1 illustrates the difference between a decision tree model and a decision path model for predicting in-hospital mortality for a patient admitted with heart failure. For the current patient, the decision tree model contains a path with features that are present in that patient, and the decision path model contains just one path with features from the patient. The probabilities of in-hospital mortality estimated by the decision tree and the decision path are 1.0 and 0.03 respectively. The patient in this case did not die in the hospital. With a probability threshold of 0.5, the decision tree would misclassify the patient, but the decision path would classify them correctly. Furthermore, the features in the two paths differ, and in this example the path in the tree has more features than the personalized path.

Several methods that construct decision path models have been described in the literature. One of the earliest methods is the Lazy Decision Tree (Lazy DT) (3), which uses a greedy search strategy and entropy as the scoring criterion to build a single path. Ferreira et al. developed two patient-specific decision path methods (4) that use entropy and balanced accuracy as the scoring criteria and apply pruning to the models. Visweswaran et al. described three personalized decision path methods (5) that use a Bayesian score, entropy, and area under the receiver operating characteristic curve (AUC) as scoring criteria. All personalized path methods demonstrated improvements in performance over decision trees in terms of accuracy or AUC.

a.

Variables		Test Case
v1	sex	0
v2	history of angina	0
v3	history of coronary artery disease	0
v4	history of lung disease	1
v5	history of diabetes	0
v6	history of percutaneous transluminal coronary angiography	0
v7	pleural effusion	1
v8	pulmonary congestion	1
v9	positive finding on EKG	0
v10	systolic blood pressure	1
v11	serum potassium	0
v12	serum sodium	1
v13	serum blood urea nitrogen	2
v14	serum creatinine	1
v15	white blood cell count	1
v16	arterial pH	1
v17	respiratory rate	1



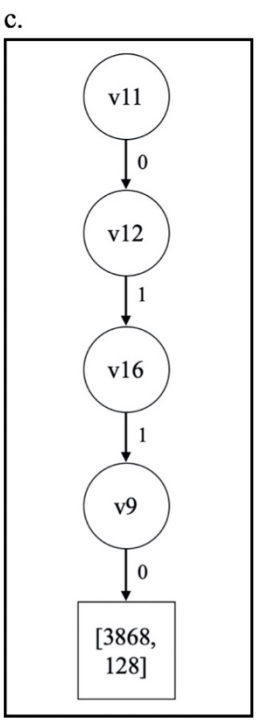


Figure 1. An example decision tree and a personalized decision path for predicting in-hospital mortality for a patient admitted with heart failure. Panel (a) lists the variables and corresponding values for a patient (test case) whose outcome we want to predict. Panel (b) shows a decision tree and the path (arrows in bold) used for inference for the patient, and panel (c) shows a personalized decision path (derived by the PDP-Bay method that is described in the section on Algorithmic Methods) for the patient. Terminal leaf nodes contain counts of corresponding training samples who survived or died during hospitalization. The patient in this case survived.

Algorithmic Methods

In this section, we provide details of the novel personalized decision path method. We first describe the PDP-Bay method in depth and then provide brief descriptions of PDP-Ent and decision tree methods used for comparison.

The PDP-Bay Method

We first describe the model structure, then the search strategy, and finally the scoring criterion of PDP-Bay.

Model structure. A decision-path model M is represented as $M = (S, \boldsymbol{\theta})$, where S is a path and $\boldsymbol{\theta}$ are the parameters of the probability distributions over the target T, which can take r possible values denoted by $(t_1, t_2, \ldots t_k, \ldots, t_r)$. Let $\boldsymbol{V} = (X_1, X_2, \ldots, X_i, \ldots, X_n)$ be a list of the n variables in the training dataset D. A path S consists of a conjunction of q features, such that $S = (X_1 = x_1 \wedge X_2 = x_2 \wedge \ldots X_j = x_j \wedge \ldots \wedge X_q = x_q)$. The variable list $\boldsymbol{V_S} = (X_1, X_2, \ldots, X_j, \ldots, X_q)$ is a subset of $\boldsymbol{V}$. The value list $\boldsymbol{v_S} = (x_1, x_2, \ldots, x_j, \ldots, x_q)$ consists of values for the variables in $\boldsymbol{V_S}$ in the test case. Finally, the parameter list $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_k, \ldots, \theta_r)$ denotes the r probabilities for the distribution $P(T \mid \boldsymbol{V_S} = \boldsymbol{v_S})$ over the target variable T.

The values of those probabilities are estimated from the cases in the training set for which $\boldsymbol{V_S} = \boldsymbol{v_S}$. To control for overfitting, we use a Bayesian estimator called the K2 score for calculating these probabilities. The estimate for probability θ_k is given as follows:

$$\theta_k \equiv P(T = t_k | \boldsymbol{V}_S = \boldsymbol{v}_S) = \frac{1+N_k}{r+N}, \quad (1)$$

where N is the number of cases in the training set that satisfy $\boldsymbol{V_S} = \boldsymbol{v_S}$ and N_k is the number of those cases that satisfy $\boldsymbol{V_S} = \boldsymbol{v_S}$ and for which $T = t_k$.

Search strategy. The pseudocode for the PDP-Bay method is shown in Figure 2. The method uses a forward-stepping, greedy hill-climbing search and a sample-normalized Bayesian score (explained below) as the criterion for evaluating features for inclusion in the path. Beginning with an empty path *S*, the algorithm successively adds features to *S* that locally maximize the score, until the score can no longer be improved. Currently, the method is designed to work with discrete data only.

PDP-Bay uses a training dataset *D* and a test case ***Test***. For each possible feature $X = x$, the algorithm temporarily appends *X* to *S* to produce candidate path *S'*. This path is scored using training data *DTemp*, which is the data for all cases that share values with ***Test*** for the features in *S'*. If the score of *S'* is greater than that of *S,* the score and candidate feature are stored. When all the features in ***V*** have been scored, the highest scoring feature is appended to *S* to create a new version of *S*, and the training dataset is reduced to cases that share values with ***Test*** for the variables in that new *S*. The growth of the path is terminated when no candidate feature can improve the score of path *S*, no remaining training cases share values with ***Test*** for the variables in *S'*, or all remaining cases have the same value for *T*.

```
PDP-Bay (V, D, Test)
    INPUT:    V = (X1, X2,..., Xi,..., Xn) ,
              D is a training dataset of cases described using V and target variable T,
              Test is data for a test case that is not in D and whose T is to be predicted
    OUTPUT:   S, where S is a personalized path for the test case and an estimate of P(T | Test)

1   S ← path with no predictor variables with a single leaf node
2   DPath ← D
3   BayScore ← Bayesian score of S computed from DPath using Equation 6
4   LOOP until DPath is empty or all cases in DPath have the same value for T
5       BayScoreBest ←BayScore
6       FOR each variable X in V whose value is not missing and takes the value x in Test DO
7           S' ← add X = x to S
8           DTemp ← cases in DPath with X = x
9           BayScoreTemp ← Bayesian score of S' computed from DTemp using Equation 6
10          IF BayScoreTemp > BayScoreBest THEN
11              BayScoreBest ← BayScoreTemp
12              XBest ← X
13          END IF
14      END FOR
15      IF a XBest is found THEN
16          S ← add XBest to S
17          DPath ← cases in DPath with X = x
18          V ← remove XBest from V
19      ELSE
20          EXIT from LOOP
21      END IF
22  END LOOP
23  RETURN S with P(T | Test) that is estimated using Equation 1 from training data that satisfy S
```

Figure 2. Pseudocode for the PDP-Bay method.

Bayesian scoring criterion. The PDP-Bay method uses a Bayesian score, the derivation of which we outline in brief. Given a candidate path *S'* that is derived from path *S* by temporarily appending a candidate feature $X = x$, we compute the posterior probability of the path *S'* (i.e., the model) given the data, *DTemp*, that contains the cases that satisfy the path *S'* as

$$P(S'|DTemp) \propto P(DTemp|S')P(S'). \quad (2)$$

We assume all paths to be equally likely a priori; therefore, $P(S'|DTemp) \propto P(DTemp|S')$. Using the Bayesian approach, we compute the marginal likelihood of the data given the path, $P(DTemp|S')$, by integrating over the parameter values:

$$P(DTemp|S') = \int_{\boldsymbol{\theta}} P(DTemp|S', \boldsymbol{\theta})P(\boldsymbol{\theta}|S')d\boldsymbol{\theta}, \quad (3)$$

where $P(DTemp|S', \boldsymbol{\theta})$ is the likelihood of the data given the path-model $(S', \boldsymbol{\theta})$ and $P(\boldsymbol{\theta}|S')$ is the prior distribution over different parameter values. Assuming parameter modularity and independence and that the variables are discrete, no data is missing, cases occur independently, and parameter priors follow a Dirichlet distribution, we compute the integral in Equation 3 in closed form using the K2 score (8), which is given by

$$P(DTemp|S') = \frac{(r-1)!}{(N+r-1)!} \prod_{k=1}^{r} N_k!\,, \quad (4)$$

where r is the number of values of the target T, N_k is the number of cases in *DTemp* for which $T = t_k$, and $N = \sum_{k=1}^{r} N_k$, which is equal to $|DTemp|$.

We sample-normalize the posterior probability by taking the geometric mean to enable score comparison between paths with varying sample sizes. The sample-normalized posterior probability represents how well on average $P(T \mid \boldsymbol{Test})$ predicts the cases in *DTemp*. For computational efficiency and precision, the final score is calculated in logarithmic form. The score of a path *S'* that includes the addition of variable *X* is therefore defined as

$$BayScore(S') = \log\left[P(DTemp|S')^{1/N}\right]. \quad (5)$$

Substituting the equation for $P(DTemp|S')$ from Equation 4 into Equation 5, we obtain

$$BayScore(S') = \log\left[\left(\frac{(r-1)!}{(N+r-1)!} \prod_{k=1}^{r} N_k!\right)^{1/N}\right] = \frac{1}{N}\left(\log\left[\frac{(r-1)!}{(N+r-1)!}\right] + \sum_{k=1}^{r} \log N_k!\right). \quad (6)$$

The BayScore is simpler than the Bayesian score that is used in the DP-Bay method (5). In the DP-Bay method, for a candidate path *S'* the data is partitioned into *DTemp* and *D – DTemp* that contain cases that satisfy the path *S'* and cases that do not satisfy the path *S',* respectively. The score for the path *S'* is computed as the K2 score of *DTemp* and *D – DTemp.* Thus, the DP-Bay score uses all the cases in *D* while the BayScore uses only the cases in *DTemp*.

The PDP-Ent Method

The PDP-Ent method is our implementation of a previously described decision path method that uses an entropy score (like Lazy DT (3)) and utilizes a search strategy that is similar to that used in the PDP-Bay method. For a candidate path *S'* and data *DTemp* that contains the cases that satisfy the path *S'*, the entropy is given by

$$H(S) = -\sum_{k=1}^{r} P(T = t_k|\boldsymbol{V}_S = \boldsymbol{v}_S) \log_2 P(T = t_k|\boldsymbol{V}_S = \boldsymbol{v}_S), \quad (7)$$

where $P(T = t_k)$ is the proportion of the dataset *DTemp* that includes samples that have the value t_k for T. The search selects variables that maximize the model score, and so we maximize the negative entropy to minimize overall entropy. Thus, the score of a variable X under consideration for inclusion in the path *S'* is defined as

$$EntScore(S') = \sum_{k=1}^{r} P(T = t_k|\boldsymbol{V}_S = \boldsymbol{v}_S) \log_2 P(T = t_k|\boldsymbol{V}_S = \boldsymbol{v}_S). \quad (8)$$

After some algebraic manipulation, Equation 8 can be rewritten as

$$EntScore(S') = \frac{1}{N}\left(\sum_{k=1}^{r} N_k \log_2 \frac{N_k}{N}\right), \quad (9)$$

where N is the number of cases in the training set that satisfy $\boldsymbol{Vs} = \boldsymbol{vs}$, N_k is the number of those cases that satisfy $\boldsymbol{Vs} = \boldsymbol{vs}$ and for which $T = t_k$, and the term inside the parentheses is the log-likelihood. Thus, the PDP-Ent method uses Equation 9 instead of Equation 6 in Figure 2.

In the PDP-Ent method, we use the maximum likelihood estimator to calculate the probabilities $\boldsymbol{\theta}$ associated with a path S. The estimate for probability θ_k is given by

$$\theta_k \equiv P(T = t_k | \boldsymbol{V}_S = \boldsymbol{v}_S) = \frac{N_k}{N}, \quad (10)$$

where N is the number of cases in the training set that satisfy $\boldsymbol{Vs} = \boldsymbol{vs}$ and N_k is the number of those cases that satisfy $\boldsymbol{Vs} = \boldsymbol{vs}$ and for which $T = t_k$. This is a common estimator that is used in many decision tree methods. Thus, the PDP-Ent method uses Equation 10 instead of Equation 1 in Figure 2.

The Decision Tree Method

The decision-tree model M is represented as $M = (\boldsymbol{T}, \boldsymbol{\theta})$, where $\boldsymbol{T} = (S_1, S_2, \ldots, S_h, \ldots, S_m)$ is a collection of m paths. Given a training dataset D, the method performs forward stepping, greedy hill-climbing to add one variable at a time that locally minimizes the entropy (or maximizes information gain), which is given by

$$TreeScore(T) = \sum_{h=1}^{m} \left[\frac{|DS_h|}{|D|} \sum_h H(S_h)\right], \quad (11)$$

where DS_h is the set of cases in D that satisfy the path S_h and $H(S_h)$ is given by Equation 7. Like in PDT-Ent, we use the maximum likelihood estimator as given by Equation 9 for calculating the probabilities $\boldsymbol{\theta}$ associated with a path S_h. This is a standard implementation of information gain in decision tree algorithms (6). The decision tree method does not perform pruning and terminates when the addition of a variable does not decrease the TreeScore.

Experimental Methods

In this section, we describe the datasets used for the study, the experimental design and evaluation, and the methods used to implement the experiments.

Datasets

We used five main datasets that included synthetic, genomic, and clinical datasets. We provide brief descriptions of the datasets in Table 1.

Table 1. Description of the datasets used in the experiments described in this paper.

Dataset	# Variables	# Values per variable	# Target values	Training set # samples (# cases)	Test set # samples (# cases)
synthetic-large	1000	3	2	9000 (1124)	1000 (146)
synthetic-small	35	3	2	9000 (1124)	1000 (146)
chronic-pancreatitis	142	3	2	1761 (784)	440 (196)
pneumonia	156	2-8	2	1601 (182)	686 (79)
sepsis-d	19	2-5	2	1115 (130)	558 (59)
sepsis-s	18	2-5	2	1115 (332)	558 (146)
heart-failure-d	17	2-7	2	7453 (333)	3725 (167)
heart-failure-c	20	2-7	2	7453 (837)	3725 (418)

Synthetic dataset. The large dataset consists of 1,000 single nucleotide variants (SNVs) as predictor variables and a binary disease variable that was modeled as a function of 35 "signal" SNVs. The first 25 signal SNVs were modeled as rare variants with minor allele frequencies (MAFs) sampled uniformly from (0.0001, 0.01) and odds ratios within (2, 10). The remaining 10 signal SNVs were modeled as common variants with MAFs sampled uniformly from (0.05, 0.50) and odds ratios within (1.05, 1.50). The remaining 965 SNVs serve as "noise" variants, ranging from common to rare with no effect on the disease status. The small dataset consists only of the 35 "signal" SNVs and the binary disease variable. Both datasets consist of 10,000 "patients," of which 12.7% have a disease target value.

Chronic pancreatitis dataset. This dataset was collected as part of the multicenter North American Pancreatitis Study 2 (9). It consists of the predictor variables, which are 142 SNVs, and a binary target variable (developed chronic pancreatitis or not). The data were previously de-identified and consist of 2,201 patients, 980 of whom were diagnosed with chronic pancreatitis, and 1,221 of whom were not.

Pneumonia dataset. This dataset was collected by the Pneumonia Patient Outcomes Research Team in a multisite study (10). The dataset consists of 2,287 adult patients admitted with community acquired pneumonia. From data collected at the time of presentation, we have 156 predictor variables, which include clinical, laboratory, and radiographic findings. The target variable is a binary variable called dire outcome. A patient was considered to have experienced a dire outcome if any of the following occurred: 1) death within 30 days of presentation, 2) an initial intensive care unit admission for respiratory failure, respiratory or cardiac arrest, or shock, or 3) the presence of one or more specific, severe complications. According to these criteria, 261 patients experienced a dire outcome and 2,026 did not.

Sepsis dataset. The sepsis dataset was collected in the multisite Genetic and Inflammatory Markers of Sepsis (GenIMS) project (11). Data were collected on 1,673 patients who were admitted from an emergency department with a diagnosis of community acquired pneumonia. From the data collected at the time of presentation, we have 19 predictor variables that consist of demographic, clinical, and genetic findings as well as inflammatory markers. Two binary outcome variables were used: 1) death within 90 days of enrollment in the study, which was true for 189 patients (sepsis-d dataset) and 2) development of severe sepsis during hospitalization, which was true for 478 patients (sepsis-s dataset).

Heart failure dataset. The heart failure dataset was collected by 192 hospitals in Pennsylvania and consists of 11,178 patients who presented in emergency departments and were admitted with a diagnosis of heart failure (12). There are 20 predictor variables that consist of demographic, clinical, laboratory, electrocardiographic, and radiographic findings. Two binary outcome variables were used: 1) death from any cause during hospitalization, which was true for 500 patients (heart failure-d dataset) and 2) development of one or more severe complications (including death) during hospitalization, which was true for 1,255 patients (heart failure-c dataset).

Experimental Protocols

We compared the predictive performance of the novel personalized path method, PDP-Bay, to that of the previously described decision tree (DT) method and the PDP-Ent method on the datasets listed in Table 1. Each dataset was randomly split approximately 80%/20% into a training and a test set such that the proportion of positive cases was the same across each pair of training and test sets. We trained models using the training sets and performed the evaluations on the test sets. We evaluated the methods on discrimination,

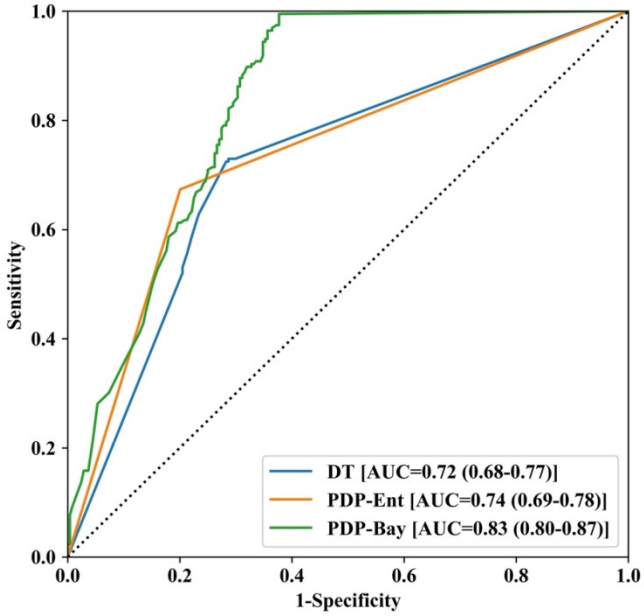


Figure 3. ROC plots for DT, PDP-Ent, and PDP-Bay methods applied to the chronic pancreatitis dataset. The respective AUCs and estimated 95% confidence intervals are also provided. Note that the confidence intervals of the PDP-Bay and DT do not overlap, indicating statistically significantly better performance of the personalized method.

calibration, and model complexity. For discrimination, we computed the AUC with the R package "pROC" (13). For calibration, we computed the expected calibration errors (ECEs), which are a measure of the difference between expected and predicted probabilities of outcomes (14). For model complexity, we measured the average path length and defined path length as the number of variables in the path that was used for inference. We statistically compared the path methods for each of the evaluation measures with each other and with the tree method across the datasets with the Wilcoxon signed-rank test statistic using the R function "wilcox.test". We implemented the three methods in Python (version 3.6). We performed all experiments on a MacBook Pro with a 3.3 GHz Dual-Core Intel Core i5 processor and 16GB of RAM, running the 64-bit macOS Sierra operating system.

Results

AUC. Over all datasets, the mean AUC was 0.63 for the DT method, 0.64 for the PDP-Ent method, and 0.74 for the PDP-Bay method. Individual AUCs for each method and dataset are shown in Table 2. As an example, the ROC plots for the DT, PDP-Ent, and PDP-Bay models on the chronic pancreatitis dataset are shown in Figure 3. On a paired two-tailed Wilcoxon signed-rank test, PDP-Bay when compared to DT had statistically significantly better performance at the 0.05 level ($p = 0.015$), and when compared to PDP-Ent had statistically significantly better performance at the 0.05 level ($p = 0.039$). PDP-Ent when compared to DT did not have statistically significantly better performance at the 0.05 level ($p = 0.742$).

Table 2. AUC values with estimated 95% C.I. of DT, PDP-Ent, PDP-Bay methods for eight datasets. The bottom row gives the mean AUCs.

Dataset	**DT**	**PDP-Ent**	**PDP-Bay**
synthetic-large	0.59 (0.55-0.63)	0.81 (0.77-0.84)	0.77 (0.72-0.82)
synthetic-small	0.66 (0.61-0.72)	0.83 (0.79-0.86)	0.82 (0.78-0.86)
chronic-pancreatitis	0.72 (0.68-0.77)	0.74 (0.69-0.78)	0.83 (0.80-0.87)
pneumonia	0.67 (0.61-0.72)	0.51 (0.49-0.53)	0.66 (0.59-0.73)
sepsis-d	0.66 (0.60-0.73)	0.55 (0.51-0.59)	0.81 (0.75-0.86)
sepsis-s	0.63 (0.58-0.68)	0.64 (0.60-0.68)	0.74 (0.69-0.79)
heart-failure-d	0.54 (0.51-0.57)	0.54 (0.52-0.57)	0.69 (0.65-0.73)
heart-failure-c	0.58 (0.56-0.60)	0.52 (0.50-0.53)	0.64 (0.61-0.67)
Mean	0.63	0.64	0.74

Calibration. The mean expected calibration error (ECE) was 0.16 for DT, 0.12 for PDP-Ent, and 0.10 for PDP-Bay. On a paired two-tailed Wilcoxon signed-rank test, PDP-Bay when compared to DT had statistically significantly better performance at the 0.05 level ($p = 0.041$), and when compared to PDP-Ent did not have statistically significantly better performance at the 0.05 level ($p = 0.104$). PDP-Ent when compared to DT had statistically significantly better performance at the 0.05 level ($p = 0.049$). ECEs for each method and dataset are shown in Table 3.

Path length. The mean path length was 7.15 for DT, 4.90 for PDP-Ent, and 5.50 for PDP-Bay. On a paired two-tailed Wilcoxon signed-rank test, PDP-Bay when compared to DT did not have statistically significantly shorter paths at the 0.05 level ($p = 0.148$), and when compared to PDP-Ent had statistically significantly longer paths at the 0.05 level ($p = 0.039$). PDP-Ent when compared to DT had statistically significantly shorter paths at the 0.05 level ($p = 0.008$). The average path length was shortest for the PDP-Ent method. Average path lengths for each method and dataset can be found in Table 4.

Table 3. Mean expected calibration errors (ECEs) of DT, PDP-Ent, PDP-Bay methods for eight datasets. The bottom row gives the average ECEs.

Dataset	DT	PDP-Ent	PDP-Bay
synthetic-large	0.19	0.07	0.05
synthetic-small	0.10	0.06	0.07
chronic-pancreatitis	0.25	0.26	0.24
pneumonia	0.14	0.11	0.11
sepsis-d	0.12	0.09	0.09
sepsis-s	0.26	0.21	0.13
heart-failure-d	0.01	0.04	0.04
heart-failure-c	0.17	0.11	0.09
Mean	0.16	0.12	0.10

Table 4. Average ± standard deviation and range of path lengths of the DT, PDP-Ent, and PDP-Bay methods for eight datasets. The bottom row gives the mean path lengths.

Dataset	DT	PDP-Ent	PDP-Bay
synthetic-large	8.72 ± 1.89 (2, 13)	2.09 ± 0.396 (1, 4)	2.41 ± 0.602 (1, 4)
synthetic-small	9.79 ± 1.91 (2, 12)	11.8 ± 4.60 (1, 25)	13.5 ± 4.85 (1, 25)
chronic-pancreatitis	5.24 ± 2.81 (1, 11)	2.44 ± 0.720 (1, 5)	2.76 ± 0.878 (1, 6)
pneumonia	4.78 ± 1.26 (3, 9)	2.36 ± 0.695 (1, 5)	2.96 ± 0.981 (1, 6)
sepsis-d	5.79 ± 2.00 (2, 12)	3.15 ± 0.850 (1, 6)	3.52 ± 0.858 (1, 7)
sepsis-s	6.25 ± 1.67 (2, 11)	3.95 ± 1.59 (0, 10)	4.26 ± 1.62 (0, 10)
heart-failure-d	7.80 ± 2.62 (4, 17)	5.96 ± 1.60 (3, 12)	6.37 ± 1.53 (2, 12)
heart-failure-c	8.87 ± 2.93 (2, 19)	7.42 ± 1.93 (2, 15)	8.19 ± 1.82 (2, 14)
Mean	7.15	4.90	5.50

Discussion

The novel PDP-Bay method demonstrated better discriminative performance by AUC than the DT method and the PDP-Ent method. The PDP-Bay method also demonstrated better calibration than the DT method. Although the path lengths of PDP-Bay were not statistically significantly shorter than those of DT, PDP-Bay path lengths were shorter for seven out of eight datasets, and its mean path length overall was shorter than the DT paths. This allows for the possibility that the single longer average path might be an outlier. If this is the case, the search and score used in the PDP-Bay method may have a regularization effect and typically produce simpler models than a standard DT.

As it uses a Bayesian score, the PDP-Bay method includes a parameter prior that is not present in the DT and PDP-Ent methods, as they use an entropy score that is equivalent to the sample-normalized log likelihood. Additionally, the parameters for PDP-Bay are estimated using Laplace smoothing, whereas parameters for DT and PDP-Ent are estimated using maximum likelihoods. Both the parameter penalty and Laplace smoothing may help the PDP-Bay method better handle uncertainty in the data and prevent it from overfitting. We conjecture that, combined with personalization, these features may be responsible for its superior performance as measured by AUC.

The previously described decision path method, PDP-Ent, demonstrated variable performance. Its AUC values were not statistically significantly better than those of DT or PDP-Bay. However, its calibration was better than that of DT, although it was not statistically significantly different from the calibration of PDP-Bay. PDP-Ent also produced the shortest path lengths. The results using PDP-Ent indicate that choice of the score highly impacts the discriminative performance of personalized modeling methods.

There are several limitations to our approach. One limitation of the PDP-Bay approach is that it can currently only handle discrete data. Any continuous data must be discretized before use, and the target variable must also be discrete. We also used data that were collected for research purposes, which may limit the generalizability of the results. Further testing on a wider variety of data, such as electronic health record data, is needed. Finally, although the predictive performance of the PDP-Bay method was better than that of the decision tree, the average AUC was 0.74. This leaves room for improvement, especially prior to use in clinical decision support. Future directions include comparing the performance of PDP-Bay to traditional personalized methods such as kNN (2), as well as other personalized path methods like DP-BAY (5). We will also strive to improve predictive performance by performing model averaging over multiple paths in an ensemble approach rather than constructing a single path model for prediction.

Conclusion

Overall, a new Bayesian machine learning method that uses personalized decision paths to predict outcomes for 8 synthetic and real clinical datasets achieved better predictive performance than a population decision tree approach.

Acknowledgements

Research reported in this publication was supported by the National Institutes of Health under award number T32GM008208 from the National Institute of General Medical Sciences, under award number U54HG008540 from the National Human Genome Research Institute, and under award numbers T15LM007059 and R01LM012095 from the National Library of Medicine. It was also supported by the Pennsylvania Department of Health (DOH) under award number 4100070287, by the National Science Foundation under award number IIS-1636786, and by the Defense Advanced Research Projects Agency under award number PA-18-02-01 (ASKE).

References

1. Visweswaran S, Angus DC, Hsieh M, Weissfeld L, Yealy D, Cooper GF. Learning patient-specific predictive models from clinical data. *J Biomed Inform.* 2010;43(5):669–85.
2. Cover TM, Hart PE. Nearest neighbor pattern classification. IEEE Trans Inf Theory. 1967;13(1):21–7.
3. Friedman JH, Kohavi R, Yun Y. Lazy decision trees. *AAAI-96 Proc.* 1996;1.
4. Ferreira A, Cooper GF, Visweswaran S. Decision path models for patient-specific modeling of patient outcomes. *AMIA Annu Symp Proc.* 2013;(2013):413–21.
5. Visweswaran S, Ferreira A, Ribeiro G, Oliveira AC, Cooper GF. Personalized modeling for prediction with decision-path models. *PLoS ONE.* 2015;10(6):e0131022.
6. Kotsiantis SB. Decision trees: a recent overview. Artif Intell Rev. 2013;39:261–83.
7. Murthy SK. Automatic construction of decision trees from data: a multi-disciplinary survey. *Data Min Knowl Discov.* 1998;2:345–89.
8. Cooper GF, Herskovits E. A Bayesian method for the induction of probabilistic networks from data. *Mach Learn.* 1992;9(4):309–47.
9. Whitcomb DC, Yadav D, Adam S, Hawes RH, Brand RE, Anderson MA, et al. Multicenter approach to recurrent acute and chronic pancreatitis in the United States: The North American Pancreatitis Study 2 (NAPS2). *Pancreatology.* 2008;8:520–31.
10. Fine MJ, Stone RA, Singer DE, Coley CM, Marrie TJ, Lave JR, et al. Processes and outcomes of care for patients with community-acquired pneumonia: results from the Pneumonia Patient Outcomes Research Team (PORT) cohort study. *Arch Intern Med.* 1999;159(9):970–80.
11. Kellum JA, Kong L, Fink MP, Weissfeld LA, Yealy DM, Pinsky MR, et al. Understanding the inflammatory cytokine response in pneumonia and sepsis: results of the Genetic and Inflammatory Markers of Sepsis (GenIMS) study. *Arch Intern Med.* 2007;167(15):1655–63.
12. Auble TE, Hsieh M, Gardner W, Cooper GF, Stone RA, McCausland JB, et al. A prediction rule to identify low-risk patients with heart failure. *Acad Emerg Med.* 2005;12(6):514–21.
13. Robin X, Turck N, Hainard A, Tiberti N, Lisacek F, Sanchez J-C, et al. pROC: an open-source package for R and S+ to analyze and compare ROC curves. *BMC Bioinformatics.* 2011;12(77).
14. Naeini P, Cooper GF, Hauskrecht M. Obtaining well calibrated probabilities using Bayesian binning. *Twenty-Ninth AAAI Conf Artif Intell.* 2015;2901–7.