Technical Details

Predict Pursuit:

We created Predict Pursuit to discover patient subgroups based upon different response to medication, and optimize the corresponding predictive model for each subgroup in order to predict the asthma controllability with specific medication. Predict Pursuit adopts a novel framework for crafting complex predictive models out of simpler baseline models by creating a phenotypic characterization of the clinical feature space in which separate predictive models are assigned to disjoint partitions of the feature space. That is, Predict Pursuit outputs a set of partitions that cover the entire feature space, together with a set of predictive models, each tailored to a given partition, thereby leading to an overall complex, granular predictive model only with simpler baseline models.

To determine phenotypes, the general approach of Predict Pursuit is to iteratively perform the following two steps:

1. From the entire patient population, the algorithm splits the patients into all possible combinations of two subgroups (pairs) based on features. Next, the subgroups are analyzed to determine whether the medication responses are statistically different between the two groups – we compute p-values for this using student t-test. Among all the possible paired subgroups, the algorithm selects the ones for which the difference between their p-values is highest (maximized).

2. The maximization in step 1 above is performed under the constraints: the difference between two sub-patient groups regarding clinical outcomes must be statistically significance. The first constraint ensures that the discovered subgroups

achieve statistically different (significant versus not-significant) response by the medication regarding clinical outcomes.

Fig. 1 portrays the block diagram of Predictor Pursuit, which sequentially discovers patient subgroups until there is no further personalization feasible (based on the optimization performed in the method described above). The final leaves of the tree discovered by our personalization method represent will be patient subgroups. A hypothetical example is: The algorithm first splits into two subgroups of young and old patients because it determined these features (of all features) have the most significantly difference of treatment response to Medication A versus B. Among the old and young patients (the old patients responded best to Medication A), a second split is among the young patients into male and females (the female responded best to Medication B) and the young males are further split into smokers and non-smokers (and only to the young male non-smoker patients Medication A. This continues until no further significant differences between paired treatment responses are determined.

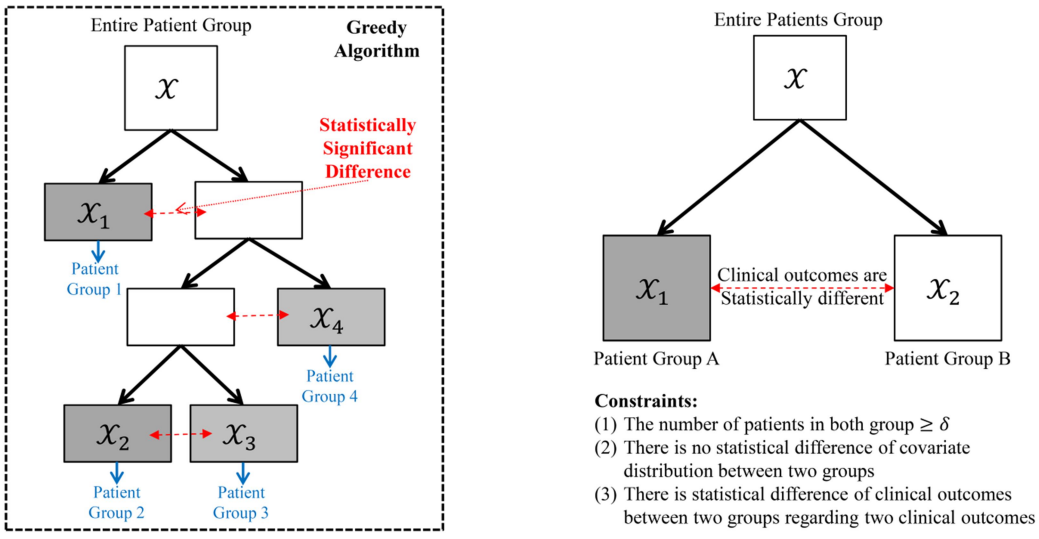

Fig 1. Block diagram of Predict Pursuit (left: greedy algorithm, right: one step of the algorithm)

In detail, Predictor Pursuit divides the clinical feature space χ into k disjoint subsets, where k is to be determined based on the given dataset, in such a way that for each subset, we optimize a separate predictive model that minimizes the overall expected risk. We write $\{\chi_1, \dots, \chi_k\}$ as a partition of the feature space χ, where all such partitions are ensured to be disjoint and cover the entire χ. Given the above construct, the learning problem becomes a problem of (jointly) finding the optimal partitioning $\{\chi_1, \dots, \chi_k\}$ of the clinical feature space, together with the optimal predictive model $h_1 \in \mathcal{H}$ associated with every partition i. This optimal partitioning and corresponding optimal predictive model-constructing problem can be formalized as follow.

$$\min_{\{\chi_1,\dots,\chi_k\}} \left[\min_{h_1,\dots,h_k \in \mathcal{H}} \sum_{i=1}^{k} \mathcal{F}(X \in \chi_i) \times \mathbb{E}_{\mathcal{F}_i}[l(h_i(\boldsymbol{x}), y)] \right]$$

$$\text{subject to } \chi = \bigcup_{i=1}^{k} \chi_i \text{, and } \chi_i \cap \chi_l = \emptyset \text{ for } \forall i \neq l$$

However, the computational complexity of solving the above optimization problem is exponentially increasing. Furthermore, the objective cannot be computed with finite number of samples. Therefore, Predict Pursuit adopts an efficient greedy algorithm for approximating the solution to the optimal partitioning problem with alternative objective (the upper bound of the expected loss based on empirical loss). As a first step to construct such an algorithm, we reformulate the optimal partitioning problem by incorporating two more constraints. First, we restrict the partitions of the recipient-donor feature space to be hypercubes. A hypercubic partition of the feature space χ is defined as $\{\chi_1, \dots, \chi_k\}$ where $\chi_i = \prod_{i=1}^{k} [a_{il}, b_{il}]$, $a_{il} \leq b_{il}$, a_{il} & $b_{il} \in \mathbb{R}$. Second, we restrict the number of partitions to be $\gamma \in \mathbb{Z}$ to solve this problem sequentially. The original optimization problem with these additional constraints can be stated as follows.

$$\min_{\{\chi_1,\dots,\chi_k\}} \left[\sum_{i=1}^{k} \min_{h_i \in \widehat{\mathcal{H}}(S_i)} \frac{1}{n} \sum_{(x_j^v, y_j^v)} l\left(h_i\left(\boldsymbol{x_j^v}\right), y_j^v\right) \right] + \alpha \sqrt{\frac{k^2 \log M}{n}}$$

$$\text{subject to } \begin{cases} \chi_i = \prod_{i=1}^{k} [a_{il}, b_{il}], \;\; a_{il} \le b_{il}, \;\; a_{il}, b_{il} \in \mathbb{R} \\ k \le \gamma, \text{where } k \in \mathbb{Z} \\ \chi = \cup_{i=1}^{k} \chi_i \text{ , and } \chi_i \cap \chi_l = \emptyset \text{ for } \forall i \neq l \end{cases}$$

Let $\mathrm{Opt}(\chi)$ be the optimal partition of the above optimization problem; we construct a greedy algorithm which iteratively solve the above optimization problem to achieve the approximate solution for the original optimization problem. More specifically, we solve the above optimization problem recursively on each χ_i separately up to the point where we do not expect to improve the objective function (i.e. the optimal k is 1). The final partition is the union of the partitions that are generated by applying this procedure recursively to each χ_i. We write the final partition achieved by the greedy algorithm as $\{\hat{\chi}_1^*, \dots, \hat{\chi}_k^*\}$ and the corresponding predictive models as $\{\hat{h}_1^*, \dots, \hat{h}_k^*\}$.

State-Transition Model:

To understand the short-term response for different treatments, we use Markov Chain framework (2-3) to construct the state transition model for asthma controllability. Markov Chain consists of three components: states (S), Action (A), and transition probability $(P_a(s, s'))$, and we need to define those components to construct the state-transition model.

First, the states of asthma (S) are defined using the 2007 NAEPP Asthma Guideline(4) criteria (State 1: Well-Controlled, State 2: Not-Well-Controlled, and State 3: Very-Poorly-Controlled). Next, actions for the asthma management (A) are either

Budesonide (Bud) or Nedocromil (Ned). Finally, the transition probability ($P_a(s, s')$) is defined as the probability that the future state will be s' if current state is s and the current action is a. Mathematically, it can be defined as

$$P_a(s, s') = \mathrm{P}(s_{t+1} = s' | s_t = s, a_t = a)$$

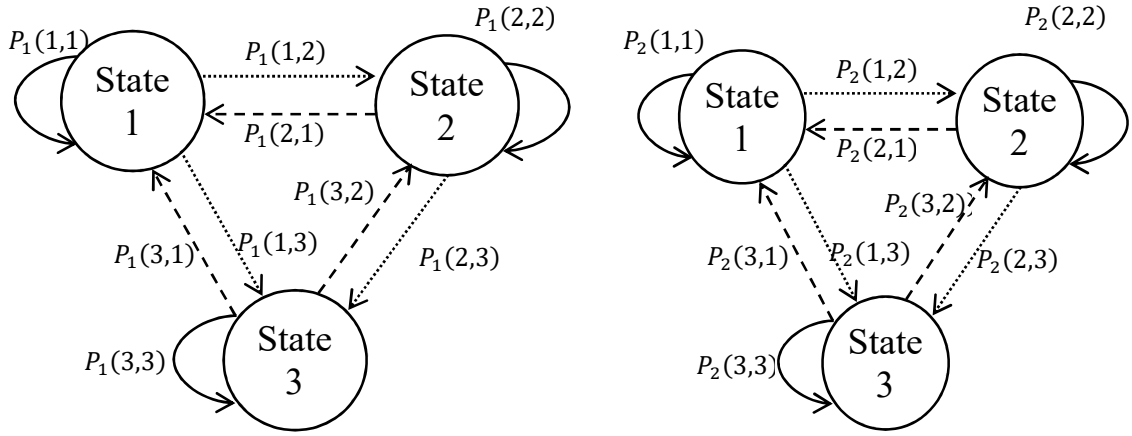

Figure 2 illustrates the state-transition model that we defined.

Budesonide (Action 1) **Nedocromil (Action 2)**

Fig 2. State-transition models (left: for budesonide, right: for nedocromil)

To complete the above state-transition models, we need to estimate the entire transition probabilities, $P_a(i, j)$ for all $a \in \{1,2\}$ and $i, j \in \{1,2,3\}$. We used Monte Carlo method to estimate the entire transition probabilities. Based on the mathematical definition of the transition probability, $P_a(i, j) = P(s_{t+1} = j | s_t = i, a_t = a)$, the estimator is,

$$\hat{P}_a(i, j) = \hat{P}(s_{t+1} = j | s_t = i, a_t = a), = \frac{N(s_{t+1} = j, s_t = i, a_t = a)}{N(s_t = i, a_t = a)}$$

for all $t \in T$, the entire time horizon. $N(s_t = i, a_t = a)$ is defined as the number of cases that the current state is i with the medication a. $N(s_{t+1} = j, s_t = i, a_t = a)$ is also similarly defined.

Feature Prediction:

We analyzed phenotype groups according to assigned treatment to determine the most predictive features to determine controllability throughout time. The predictive power of the features can be evaluated by the prediction accuracy of the Adaptive Boosting algorithm (5). We used single feature among the entire features and compute the prediction accuracy (area under the curve) for each feature. Then, we sorted the features based on the value of the area under the curve (AUC). We defined the top five predictive features as the features whose AUC values are highest. This methodology can be easily extended to multiple feature analysis if we treat multiple features as a subset of features.

To understand the effect of the feature on the asthma controllability, we used density estimation and computed the mean and standard deviation of the feature for well-controlled patients and not-well-controlled patients. For instance, if the average value of the well-controlled patients for this feature is higher than not-well-controlled patients, we can say that higher value of this feature implies that this patient has a higher chance to be well-controlled.

References

1. Yoon J, Alaa AM, Cadeiras M, van der Schaar M. Personalized Donor-Recipient Matching for Organ Transplantation. arXiv preprint arXiv:161103934. 2016.
2. Asmussen S. Applied Probability and Queues. Springer Science & Business Media. 2003:7
3. Parzen E. Stochastic Processes. Courier Dover Publications. 2015:188
4. National Asthma E, Prevention P. Expert Panel Report 3 (EPR-3): Guidelines for the Diagnosis and Management of Asthma-Summary Report 2007. J Allergy Clin Immunol. 2007;120(5 Suppl):S94-138.
5. Freund Y, Robert ES. A decision-theoretic generalization of on-line learning and an application to boosting. European conference on computational learning theory. Springer Berlin Heidelberg. 1995