Predicting and Evaluating Individual Outcomes and Net Treatment Benefit from Randomized Trial Data: A Case Study on the Diabetes Prevention Program

Yu Liu and Michael C. Hughes
Department of Computer Science
Tufts University
Medford, MA 02155

Abstract

Using retrospective data from a randomized clinical trial (the Diabetes Prevention Program), we develop and evaluate preliminary supervised machine learning pipelines to predict two related outcomes for an individual patient: the probability of diabetes onset (a binary classification task) and the net-benefit of possible treatments. Our prediction models take as input several individual covariates as well as two possible randomized treatment assignments (lifestyle intervention and pharmaceutical intervention). Our evaluation considers discriminative quality and net-benefit. Results indicate that random forests and their causal extension (Causal Forests) have a slight benefit over simpler linear models, more noticeable on some interventions than others. Future work will look at extending our proposed evaluation pipeline to other methods, as well as to calibration and interpretability.

1 Introduction

Many clinical decisions can be framed as a binary choice: Should we treat this patient, or not? Should we prescribe the aggressive therapy, or the less-risky one? Doctors often have abundant personalized information about a patient that could inform decisions, such as demographics, medical history, and lab test results. However, most guidelines about treatment decisions, such as evidence from clinical trials, rely primarily on average effects across many subjects. There is a pressing need to develop methods that can take evidence derived from groups and apply it effectively at the individual level. Specifically, given a large dataset of personalized covariates, treatments, and outcomes, our goal is to build a predictor of the individual treatment effect (ITE), the overall net benefit of treating the patient rather than not treating.

In this preliminary report, we develop both a *modeling pipeline* to predict individual treatment effect using several off-the-shelf models, as well as a *validation procedure* to assess these methods on real datasets.The key challenge for ITE prediction methods is validation: experiments should assess a method for accuracy, calibration, interpretability, and clinical benefit. Thus far, ML predictors for ITE have been checked only on simulated data or via a single experiment on real data. For example, the recently proposed Causal Boosting (Powers et al., 2018) was evaluated on one randomized blood pressure management trial. The Causal VAE (Louizos et al., 2017) evaluated their proposed deep probabilistic model on one study of twin infant mortality. Neither included formal assessments of calibration or net benefit. Rigorous comparison and evaluation is sorely needed. We offer a preliminary first step toward more comprehensive evaluation.

Preprint. Under review.

2 Related Work

Work in this emerging field spans many important axes, including what quantity is predicted (risk vs. effect of treatment), which models are used, and whether data is collected via randomized trials or more difficult "observational" settings where treatments are not random but may depend on patient characteristics.

Previous work by Kent et al. (2016) used generalized linear models (e.g. logistic regression) to estimate outcome risk from 32 clinical trials with at least 1000 subjects. Variable selection used clinical reasoning and the literature, including at most 20 covariates per study. Findings suggest that stratifying patients by predicted risk can lead to clinically significant differences in outcomes between the extreme quartiles (top 25% and lowest 25%). Among the 18 trials with non-null treatment effects, risk-modeling often revealed clinically important treatment effect differences. This work produced several followup clinical papers reporting potential practice-changing insights, including a model to improve diabetes interventions by Sussman et al. (2015) now deployed in a PCORI-sponsored study in two health networks serving over 300,000 patients. However, the use of generalized linear models, chosen for simplicity and reliability, could be potentially underperforming at prediction compared to more recent non-linear ML methods.

Machine learning researchers have developed state-of-the-art non-linear approaches to ITE prediction for both the randomized and observational data settings. One promising approach is the Causal Tree (Athey and Imbens, 2016), an extension of decision trees from non-causal supervised learning to the treatments and outcomes setting. Tree partitioning models can learn more flexible decision boundaries than linear models, leading to better predictions, while still allowing clinicians with little computational expertise to inspect the underlying decision tree visually to understand why a prediction was made for a given input. Powers et al. (2018) then proposed Causal Boosting, a method to successively fit several causal trees one after the other, each predicting the residual error from the previous tree so the ensemble can produce an overall low-error prediction. Causal boosting thus has similar benefits to the well-known gradient boosting technique from standard supervised learning used in the winning solutions of 17 of 29 challenges in recent Kaggle benchmarking competitions (Chen and Guestrin, 2016). Alternatively, Athey et al. (2019) propose the Generalized Random Forest, which trains many independent causal trees on random subsets of data and features and averages their predictions to reach low heldout error estimates. Causal Boosting and the Generalized Random Forest thus represent state-of-the-art ML treatment effect prediction approaches whose components are decision trees understandable by an average physician.

Other machine learning approaches to individual treatment effect prediction develop principled probabilistic models to capture nonlinear relationships between covariates, treatments, and outcomes. (Louizos et al., 2017) propose the Causal Effect Variational Autoencoder, a method designed to account for the influence of hard-to- measure confounders (such as socioeconomic status) by treating them as unobserved random variables that mediate outcomes and which can be inferred from noisy "proxy" covariates (e.g. zip codes and occupation). By using modern deep neural networks to parameterize flexible distributions, the CE-VAE might deliver more accurate predictions than linear models while using its probabilistic framing to remain calibrated. However, possible limitations include the potential for overfitting and the need to counteract this with careful model selection techniques.

3 Dataset

We have data from the Diabetes Prevention Program trial (Dia, 1999), a randomized clinical trial designed to assess various treaatment strategies to prevent or delay the development of type 2 diabetes in individuals deemed at high risk based on physiological measurements prior to enrollment. Total enrollment consisted of over 3000 individuals from 27 clinical centers. Participants were recruited over a 2.5 year period, then followed for additional 3-5 years (the study closed in 2002). The primary outcome of interest is the development of diabetes, diagnosed by physiological measurement of glucose concentrations that meet or exceed official diagnostic criteria of the American Diabetes Association. There are two possible treatments considered: an intense lifestyle intervention ('lifestyle') and a pharmaceutical intervention ('metformin'). Each of these was compared with a "control" intervention population. Treatments (lifestyle, metformin, control) were randomly assigned to each patient.

Numeric variables	min	max	5%	95%	median	missing values
Age	38	66	38	66	52	0
Fasting plasma glucose	99	139	99	122	105	0
Glycosolated hemoglobin concentration	3.2	8.5	5.1	6.7	5.9	8
height	139.00	197.55	152.75	182.90	166.25	0
Waist to hip ratio.	0.62	1.89	0.79	1.06	0.92	4
Waist	69.45	190.00	84.14	131.04	103.90	3
Bmi	24	44	24	44	33	0
Average of systolic BP	80	179	103	150	122	0
Triglycerides	31	920	62	339	141	5
Cholesterol high-density lipoproteins	19	105	30	67	44	5
Metabolic quantity	0.0	683.40	0.00	49.91	9.70	0

Table 1: Summary of the range of observed values available in raw data for the numerical variables used in this study. Clearly there are extreme outliers indicative of measurement error (either sensor failure or data entry error). In future, we will establish clinically-sound plausibility ranges and discard values which fall outside.

Binary variables	most frequent category	most frequency	missing values
Treatment with metformin	0	0.67	0
Treatment with intensive lifestyle	0	0.67	0
Female	1	0.67	0
Black	0	0.79	0
Hispanic	0	0.79	0
Other races	0	0.79	0
Gestational diabetes	0	0.90	1
History of high blood pressure	0	0.80	0
Family history of diabetes	1	0.69	2
Smoking status	0	0.93	0
Hypertension	0	0.73	0

Table 2: Summary of observed values available in raw data for the binary variables used in this study. Clearly there are extreme outliers indicative of measurement error (either sensor failure or data entry error). In future, we will establish clinically-sound plausibility ranges and discard values which fall outside.

Our project team gained permission to use the data through our affiliation with the overall Data Usage Agreement obtained by the PACE center at Tufts Medical Center.

4 Prediction Task

In this section, we describe how we prepared the raw clinical trial data to be used to develop and evaluate prediction models for both binary diagnostic outcomes and net benefit of treatment. We formulate these tasks using the raw data from the DPP trial (Dia, 1999) as available in the official release and previously preprocessed by (Kent et al., 2016) to select covariates and handle missingness.

There are 3081 total cases in the dataset. There were 3 possible treatment conditions (each assigned at random): 1024 cases treated by lifestyle intervention, 1027 cases treated by metformin, and the 1030 cases receiving neither treatment.

4.1 Outcome determination.

Our goal is to predict the risk of the onset of diabetes within the 3-5 year follow-up period after subjects were recruited into the DPP study. We use the onset of diabetes binary outcome recorded in the original DPP trial dataset. Typically, around 20-25% of the subjects eventually were diagnosed with diabetes (across all treatments).

	training				test			
fold	num. patients	y_n=1 freq.	num. treated	num. control	num. patients	y_n=1 freq.	num. treated	num. control
1	1851	0.251	916	935	206	0.194	111	95
2	1851	0.244	922	929	206	0.262	105	101
3	1851	0.248	923	102	928	0.223	104	102
4	1851	0.243	923	928	206	0.267	104	102
5	1851	0.238	930	921	206	0.311	97	109
6	1852	0.245	920	932	205	0.249	107	98
7	1851	0.242	936	915	206	0.277	91	115
8	1852	0.246	937	915	205	0.239	90	115
9	1851	0.252	922	929	206	0.189	105	101
10	1852	0.246	914	938	205	0.244	113	92

Table 3: Summary of outcome and treatment data used for metformin intervention treatment analysis, including hyperparameter selection, modeling and evaluation of generalization. Here, outcome=1 means indicator of diagnosed with diabetes during DPP (Computed based on fasting and/or 2-hour glucose values from the central laboratory).

4.2 Covariate Selection and Preprocessing

We selected the same compact set of covariates as in previous studies (Kent et al., 2016). Selected covariates are summarized in Table 1 (numerical) and Table 2 (categorical). This includes sociodemographic information (age, sex, race), physical characteristics (height, waist-to-hip ratio, bmi), historical disease indicators (family history of diabetes, history of high blood pressure), and behavioral indicators (smoking status, diagnosis of gestational diabetes, hypertension). Several vital signs and laboratory measurements are also included (concentration of fasting plasma glucose, glycosolated hemoglobin, average systolic BP, triglyceride, cholesterol high-density lipoproteins).

Missing values. Missing values of some covariates could either prevent model evaluation (if methods cannot handle missing data) or compromise the learned predictions (if imputation is not done carefully). Across entire dataset, we find missingness is rather rare, as Tables 1 and 2 show that no variable has more than 8 missing cases out of 3000. The most often missing variable was glycosolated hemoglobin concentration, while over half of the covariates had no missingness at all. We examine the missing data in glycosolated hemoglobin concentration with other variable and do not detect any correlation between the them, so we assume the missingness in glycosolated hemoglobin concentration is at random. For each variable with missingness we used predictive mean matching (Rubin, 1986) as imputation method to deal with missing data. Specifically, a linear regression model is built for variable with missingness on variables without missingness; then random sampling posterior predictive distribution from the coefficients of the model and generate new set of coefficients, use the new coefficients to predict on variable with missingness. Identify predictions that are closest to the missing data and randomly choose one's observed value as imputation for the missing value. Repeat this procedure if needed.(Heitjan and Little, 1991; Schenker and Taylor, 1996).

Imputation is done once for the entire dataset, before any other preprocessing. We applied sequential imputation for all missingness using the open-source R software package for Multivariate Imputation by Chained Equations ("MICE") (van Buuren and Groothuis-Oudshoorn, 2011), though we emphasize that we performed just one imputation (rather than multiple) because the missingness rate was so small.

Preprocessing. All covariates (both numerical and categorical) were preprocessed with a centering and scaling transform, such that each covariate in the overall dataset had empirical mean zero and empirical variance equal to one. Treatment indicators are not scaled or centered. This transformation improves the reliability and robustness of later model training by standardizing the numerical scales of all covariates (and thus any corresponding parameters). We expect this is far more important for generalized linear models than tree-based models.

	training				test			
fold	num. patients	y_n=1 freq.	num. treated	num. control	num. patients	y_n=1 freq.	num. treated	num. control
1	1848	0.261	915	933	206	0.165	109	97
2	1849	0.216	926	923	205	0.166	98	107
3	1849	0.209	918	931	205	0.229	106	99
4	1848	0.216	908	940	206	0.165	116	90
5	1849	0.208	935	914	205	0.239	89	116
6	1849	0.207	931	918	205	0.254	93	112
7	1848	0.21	912	936	206	0.223	112	94
8	1849	0.214	925	924	205	0.185	99	106
9	1848	0.211	924	924	206	0.214	100	106
10	1849	0.204	922	927	205	0.273	102	103

Table 4: Summary of outcome and treatment data used for lifestyle intervention treatment analysis. including hyperparameter selection, modeling and evaluation of generalization. Here, outcome=1 means indicator of diagnosed with diabetes during DPP (Computed based on fasting and/or 2-hour glucose values from the central laboratory).

5 Methods for Predicting Outcomes

The binary outcome prediction computational problem is defined as follows for each individual patient (indexed by n). The input is a covariate vector $\tilde{x}_n$ with F entries, $\tilde{x}_n \in \mathcal{X} \subset \mathbb{R}^F$. Given this input, the prediction system should output the predicted *probability* of the binary outcome of interest $y_n \in \{0, 1\}$. We use notation $\tilde{x}_n$ to indicate that the covariates for outcome prediction may include both the "raw" covariates x_n (e.g. those listed in Table 1, Table 2) as well as the known treatment t_n assigned to the patient.

We approach this problem as a supervised binary classification problem with probabilistic predictions. Our model development procedure consumes a labeled "training" dataset of N pairs of covariates and outcome labels: $\{\tilde{x}_n, y_n\}$, and produces a function $\hat{y}_\theta$ that maps any covariate vector in $\mathcal{X}$ to a probability (a numerical value in the unit interval $[0, 1]$). We define this probability as: $\hat{y}_\theta(\tilde{x}_n = \text{feat}(x_n, t_n)) = \Pr(y_n = 1|x_n, t_n)$.

Below, we describe two methods to perform probabilistic binary classification, each of which will produce a learned function $\hat{y}_\theta(\cdot)$ which maps from $\mathcal{X}$ to a predicted probability of the diagnosis outcome (where binary value 1 indicates having diabetes after the study's designated follow-up period, and binary value 0 indicates having no such diagnosis). Training these prediction models requires both fitting parameters (on the training set) and selecting hyperparameters on a heldout validation to avoid overfitting. To do fitting and selection while also providing accurate estimates of generalization performance on never-before-seen data, we use a nested cross-validation evaluation framework (see Sec. 5.3).

5.1 Logistic Regression (LR)

The first predictor we consider is logistic regression (LR). Given a fixed-length vector of covariates, LR makes a prediction by applying a linear transformation (using a weight parameter for every covariate plus a bias parameter) to get a real-valued score, and then feeding that score through a logistic sigmoid function to obtain a predicted probability. LR belongs to the class of generalized linear models, with several advantages including interpretable model structure and reliable parameter learning. Parameters are learned from training data via gradient descent on a penalized maximum likelihood objective. We use the public implementation package in glmnet (Friedman et al., 2010). We control overfitting by searching for the strength of L1 and L2-regularization penalty. No penalty is added to treatment variables. Our grid search considers 50 logarithmically spaced values from 10^{-5} to 10^2. We verified that values at the edges of the grid are never selected (thus indicating our grid spans a reasonable range containing the ideal optima for this data).

5.2 Random Forests (RF)

The second predictor we consider is an ensemble of decision trees known as a random forest (RF) (Breiman, 2001). We use the public implementation in the `RandomForest` R package (Liaw and Wiener, 2002). When building for each tree, a subset of sampling data is split based on a random sampled set of variables. Each tree in the random forest is built on a separate random sub-sampling of the training data. We set the forest's size to 500 trees (we found more would not improve prediction quality much but become computationally expensive). We control underfitting and overfitting by searching for number of features used within each tree, ranging from 2 to 15, as the hyperparameter grid search.

5.3 Training and Validation Protocol.

Given a dataset with only a few thousand total examples, we need to carefully consider how we train and evaluate models so that we can adequately estimate generalization to new data. We consider a *nested cross-validation* paradigm (Cawley and Talbot, 2010; Raschka, 2018), a gold standard when evaluation needs to provide accurate assessments of generalization performance for a full *pipeline* that requires both parameter training as well as hyperparameter selection. The "inner loop" performs both parameter training and hyperparameter selection using grid search where the performance is averaged across 10 folds. Among all grid search candidate models, we select the one that maximizes the area-under-the-ROC-curve metric (averaged across folds). The "outer loop" provides an estimate of this training-and-selection procedure's generalization error, by repeating it over 10 outer folds. All performance metrics report the average across folds, as well as the 20-th percentile and 80-th percentile across 10 folds to communicate uncertainty (e.g. answering the practical question of what range of performance values could be expected for another test set of the same size drawn from the same population).

We train and evaluate two *separate* prediction models, one for each of the two targeted treatments: lifestyle and metformin. For each model, we include only the targeted treatment and the control treatment population (so the total dataset has roughly 2000 examples before the nested cross-validation is applied).

5.4 Evaluation Metrics for Outcome Prediction

To assess overall discriminative quality, we look at area under the receiver operating curve (AUROC), also know as the C-statistic. Higher values indicate better performance (AUROC = 1.0 would be perfect, AUROC = 0.5 would represent guessing the binary outcome with a fair coin).

In future, we plan to also include *calibration* and *interpretability* assessments.

6 Methods for Predicting Net Benefit of Treatment

Our approach to determining net benefit of treatment for an individual is based on the potential outcomes framework (Rubin, 2005). We consider two possible treatments given to the n-th patient indicated by random variable T_n, such that the possible values are $T_n = 1$ or $T_n = 0$). We model the *unobservable* outcomes under each treatment condition: $Y_n(1), Y_n(0)$, as two other random variables each with its own distribution. Given the treatment assignment T_n, we observe only one outcome Y_n, which is constructed deterministically given T and the potential outcomes: $Y_n = T_n Y_n(1) + (1 - T_n) Y_n(0)$. Now, define the *net benefit* of a treatment for the individual patient n as:

$$\tau(x_n) = \mathbb{E}_{p(Y_n(1), Y_n(0)|x_n)} \left[Y_n(1) - Y_n(0) \right] \tag{1}$$

where the expectation is taken with respect to the true underlying distribution over potential outcomes given the covariates used to generate the data.

We can call this quantity the *net benefit*, or it might also be called the *individual treatment effect*. Other authors refer to this quantity as the "conditional average treatment effect", with conditional understood to mean conditioning on specific covariate values x_n.

6.1 Prediction methods for net benefit

Causal Random Forests (Causal-RF) . The Causal Forest is based on an ensemble of decision trees like the random forest, but uses carefully designed modifications to predict the heterogeneous treatment effect (Athey et al., 2019). Similar with random forest, Causal Forest adapts that idea of recursive partitioning, sub-sampling, and random split selection when building each tree, except that Causal Forest specifically is aware of treatment variables and tries to maximize heterogeneity of treatments when splitting data to build each tree. The causal forest predicts the net benefit for each case directly. We used open-source `grf` package (Tibshirani et al., 2020). The model sample a subset of data for each tree. Trees are built with honesty (Wager and Athey, 2018). Namely, within each subset sample, half of the sample is used for splitting and building the tree while the other half is used for evaluating the treatment variance of this tree. We did parameter search on number of variables used when building each tree, ranging from 2 to 15. Since our dataset is small($\sim$ 2000 entries for both lifestyle and metformin), we did not tuning leaves after honest split as suggested in paper (Tibshirani et al., 2020).

Adapting the Causal Forest to Outcome Prediction. Methods exist to produce a useful estimator of the outcome $\hat{y}(x_i)$ given a trained causal forest (Athey et al., 2019). For now, as a (simple but clearly suboptimal) attempt, we simply use the predicted treatment effect $\hat{\tau}(x_i)$ as a plug-in estimate of the prediction score $\hat{y}(x_i)$.

6.2 Evaluation of Net Benefit Prediction

To capture the benefit prediction performance, we use the recent c-for-benefit statistic (van Klaveren et al., 2018). Net benefit is defined as the net gain of predicted risk of treated patients rather than not treated. The predicted benefit is calculate through model, taking the difference of predicted risk from original treatment and the predicted risk alternative treatment.

C-for-benefit. Recall that the C-statistic for binary outcome tasks measures the probability that the classifier will produce the correct ranking given two input features x_i and x_j with known distinct true labels $y_i = 1$ and $y_j = 0$:

$$\text{C-for-discrimination:} \qquad p(\hat{y}_\theta(x_i) > \hat{y}_\theta(x_j) | y_i = 1, y_j = 0) \tag{2}$$

The performance metric of C-for-benefit was introduced by van Klaveren et al. (2018) to extend this notion of discriminative performance ("how well does this tool correctly suggest separating patients that should be separated?") to treatment benefit prediction. First, we form candidate pairs of patients such that the pair (i, j) have similar predicted benefit ($\hat{\tau}(x_i) \approx \hat{\tau}(x_j)$) but distinct treatment assignments (without loss of generality, $T_i = 1; T_j = 0$). This lets us form a local empirical estimate to the treatment effect: $\tau_A = y_i - y_j$, which will be positive if the outcome for patient i (who received the treatment).

Consider two pairs of patients $A = (i_a, j_a)$ and $B = (i_b, j_b)$. The C-for-benefit statistic then measures the probability that the predicted benefit for pair A is better than for pair B, given that the *observed* empirical benefit was better for A than B:

$$\text{C-for-benefit:} \qquad p(\hat{\tau}_\theta(x_A) > \hat{\tau}_\theta(x_B) | \tau_A > \tau_B) \tag{3}$$

Like the classic C-statistic, the C-for-benefit can be interpreted such that a "random" net-benefit prediction would achieve 0.5 chance, while a perfect predictor would achieve 1.0.

7 Results

The prediction performance of models, in terms of both outcome prediction and benefit prediction quality (c-for-benefit), is shown in Tab. 5 and Tab. 6. We show prediction performance for our proposed pipeline. We further provide example ROC plots in Fig. 1.

Several trends are noticeable and discussed below.

Causal forest does better on c-for-benefit, esp. for lifestyle. When predicting net benefit, causal forest outperforms LR models (Ridge and Lasso), for both lifestyle intervention (c-for-benefit of 0.661-0.764 vs. 0.606-0.679) and metformin treatment (c-for-benefit of 0.544-0.646 vs. 0.510-0.590).

metformin	c-statistic for outcome (AUROC)			c statistic for benefit		
	mean	20%	80%	mean	20%	80%
ridge	0.846	0.833	0.863	0.543	0.510	0.590
lasso	0.846	0.834	0.862	0.546	0.498	0.596
random forest	0.859	0.850	0.870	0.567	0.516	0.609
causal forest	0.753	0.720	0.768	0.589	0.544	0.646

Table 5: Result comparison of models, using metformin treatment dataset. We show the mean of 10 heldout test sets under our nested cross-validation framework. We communicate uncertainty via the 20th and 80th percentiles of the 10 folds.

lifestyle	c statistic for outcome (AUROC)			c statistic for benefit		
	mean	20%	80%	mean	20%	80%
ridge	0.841	0.808	0.865	0.640	0.606	0.679
lasso	0.840	0.807	0.865	0.629	0.590	0.653
random forest	0.846	0.823	0.877	0.679	0.652	0.698
causal forest	0.827	0.808	0.846	0.715	0.661	0.764

Table 6: Result comparison of models, using lifestyle intervention treatment dataset. We show the mean of 10 heldout test sets under our nested cross-validation framework. We communicate uncertainty via the 20th and 80th percentiles of the 10 folds.

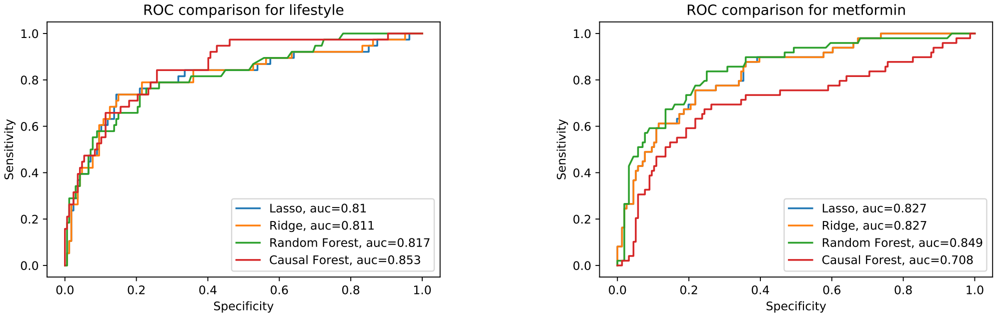

Figure 1: ROC curves showing tradeoff in achievable sensitivity and specificity (true positive rate and true negative rate) for binary outcome predictions evaluated on *one* training-test split with 1849 train examples and 206 test examples for lifestyle intervention treatment, and 1852 train examples and 205 test examples for metformin treatment. *Left:* Performance of models for lifestyle-vs-control treatment, where available covariates were all those in Tables 1 and 2 plus a binary indicator for the DPP trial's lifestyle. *Right:* Performance of models for metformin-vs-control treatment, where available covariates were all those in Tables 1 and 2 plus a binary indicator for the DPP trial's lifestyle. In all panels, higher y-axis values indicate better performance. A random guessing classifier would achieve a diagonal line.

In calculation of predicted benefit, LR models and random forest calculate the prediction difference of treatment and alternative treatment. These models are trained and built to make accurate prediction on outcome, whereas for causal forest, model is built to maximize the variance between treatment groups. As a result, we would expect the net benefit would have a higher concordance with observed benefit.

Lifestyle yields higher c-for-benefit than metformin. The result of c-for-benefit in lifestyle intervention is higher than metformin treatment across all models. Eg. for ridge regression model, the c-for-benefit ranges is 0.510-0.590 for metformin treatment vs. 0.606-0.679 for lifestyle intervention. This result seems to mathc previous results looking at benefit across risk quartiles **?**, Fig. 5, which suggest that the effect size (overall absolute risk reduction) of lifestyle intervention was overall larger than the metformin intervention.

For benefit prediction, RF appears modestly better than or equal to LR. In treatment with lifestyle intervention, random forest model has a higher c-for-benefit (0.652-0.698) comparing with LR models (0.606-0.679, 0.590-0.653). The more flexible non-linear RF predictor is likely advantageous here. For metformin intervention, RF seems to be about the same as LR (c-for-benefit 20-80% percentile range is 0.516 - 0.609 for RF compared to 0.510 - 0.590 for LR). Perhaps there is not a strong non-linear interaction between treatment indicator and covariates, and thus linear predictors are adequate in this case.

For outcome prediction, RF does not appear better than LR . When predicting outcome, LR models (Ridge and Lasso) perform comparably with forest model (Random Forest) for both lifestyle intervention (AUC of 0.808-0.865 vs. 0.823-0.877) and metformin treatment (AUC of 0.833-0.863 vs. 0.850-0.870). Of note, the c statistic of prediction is calculated through predicted benefit vs. true outcome, so the c statistic would expect not as accurate as other models.

References

The Diabetes Prevention Program. *Diabetes care*, 22(4):623–634, 1999. `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1351026/`.

S. Athey and G. Imbens. Recursive partitioning for heterogeneous causal effects. *Proceedings of the National Academy of Sciences*, 113(27):7353–7360, 2016.

S. Athey, J. Tibshirani, and S. Wager. Generalized Random Forests. *The Annals of Statistics*, 47(2): 1148–1178, 2019. `http://arxiv.org/abs/1610.01271`.

L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

S. van Buuren and K. Groothuis-Oudshoorn. {mice}: Multivariate imputation by chained equations in r. *Journal of Statistical Software*, 45(3):1–67, 2011. `https://www.jstatsoft.org/v45/i03/`.

G. C. Cawley and N. L. C. Talbot. On Over-fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation. *Journal of Machine Learning Research*, 11:29, 2010.

T. Chen and C. Guestrin. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, pages 785–794, San Francisco, California, USA, 2016. ACM Press.

J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010. `http://www.jstatsoft.org/v33/i01/`.

D. F. Heitjan and R. J. A. Little. Multiple Imputation for the Fatal Accident Reporting System. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 40(1):13–29, 1991.

D. M. Kent, J. Nelson, I. J. Dahabreh, P. M. Rothwell, D. G. Altman, and R. A. Hayward. Risk and treatment effect heterogeneity: Re-analysis of individual participant data from 32 large clinical trials. *International Journal of Epidemiology*, 45(6):2075–2088, 2016.

D. van Klaveren, E. W. Steyerberg, P. W. Serruys, and D. M. Kent. The proposed 'concordance-statistic for benefit' provided a useful metric when modeling heterogeneous treatment effects. *Journal of Clinical Epidemiology*, 94:59–68, 2018.

A. Liaw and M. Wiener. Classification and regression by randomForest. *R News*, 2(3):18–22, 2002. `https://CRAN.R-project.org/doc/Rnews/`.

C. Louizos, U. Shalit, J. M. Mooij, D. Sontag, R. Zemel, and M. Welling. Causal Effect Inference with Deep Latent-Variable Models. In *Neural Information Processing Systems*, 2017.

S. Powers, J. Qian, K. Jung, A. Schuler, N. H. Shah, T. Hastie, and R. Tibshirani. Some methods for heterogeneous treatment effect estimation in high dimensions. *Statistics in Medicine*, 37(11): 1767–1787, 2018.

S. Raschka. Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning. *arXiv preprint*, 2018. `http://arxiv.org/abs/1811.12808`.

D. B. Rubin. Statistical Matching Using File Concatenation With Adjusted Weights and Multiple Imputations. *Journal of Business & Economic Statistics*, 4(1):87–94, 1986.

D. B. Rubin. Causal Inference Using Potential Outcomes. *Journal of the American Statistical Association*, 100(469):322–331, 2005. `https://amstat.tandfonline.com/doi/abs/10.1198/016214504000001880`.

N. Schenker and J. M. G. Taylor. Partially parametric techniques for multiple imputation. *Computational Statistics & Data Analysis*, 22(4):425–446, 1996. `http://www.sciencedirect.com/science/article/pii/0167947395000577`.

J. B. Sussman, D. M. Kent, J. P. Nelson, and R. A. Hayward. Improving diabetes prevention with benefit based tailored treatment: Risk based reanalysis of Diabetes Prevention Program. *BMJ (Clinical research ed.)*, 350:h454, 2015.

J. Tibshirani, S. Athey, and S. Wager. *Grf: Generalized Random Forests*, 2020. `https://CRAN.R-project.org/package=grf`.

S. Wager and S. Athey. Estimation and Inference of Heterogeneous Treatment Effects using Random Forests. *Journal of the American Statistical Association*, 113(523):1228–1242, 2018. `https://doi.org/10.1080/01621459.2017.1319839`.