Targeting Prospective Customers: Robustness of Machine Learning Methods to Typical Data Challenges

Duncan Simester[*], Artem Timoshenko[*], and Spyros I. Zoumpoulis[†]

[*]Marketing, MIT Sloan School of Management, Massachusetts Institute of Technology
[†]Decision Sciences, INSEAD

November 4, 2018

Appendices

Contents

A Notation

Table 1: Summary of notation

Term	Description
N	The number of carrier routes in Stage 1, i.e., 5,976
p	The number of targeting variables, i.e., 13
i	The index of the unit of observation (a carrier route)
t	The experimental treatment
$\mathbf{X}$	A $N \times p$ matrix capturing the targeting variables for all observations
$\mathbf{x}_i$	The ith row of matrix $\mathbf{X}$, i.e., a $1 \times p$ vector that holds the values of all the targeting variables for observation i
$\mathbf{y}^t$	A $N \times 1$ response vector capturing the *realized* outcome measure (expected 12-month profit) for all observations under treatment t
y_i^t	The *realized* outcome for observation i under treatment t
$\hat{y}_i^t$	The *predicted* outcome for observation i under treatment t

B Customer Segmentation Methods

B.1 Distance-driven Methods

Distance-driven methods make the best possible profit predictions for each new (i.e., Stage 2) observation, for each treatment, by using the Stage 1 observations that are the closest to the new observation, where "closest" is meant in terms of some distance metric computed from the observations' targeting variables. Each observation is then assigned the treatment that results in the highest predicted profit.

B.1.1 Kernel Regression

Overview. Kernel regression is a non-parametric technique that finds a non-linear relation between the targeting variables and the response variable, i.e., profit. This non-linear function is estimated using a kernel as a weighting function. We estimate a different function for each treatment, based on the Stage 1 observations, and we assign to each new observation the treatment that results in the highest predicted profit.

Implementation and cross-validation. In the kernel regression approach, we estimate the following function for each treatment t for a new (i.e., Stage 2) observation with targeting variables $\mathbf{x}_{new}$:

$$\hat{y}^t_{new} = \frac{\sum_{i=1}^{N} K_\gamma(\mathbf{x}_{new}, \mathbf{x}_i) w_i^t y_i^t}{\sum_{i=1}^{N} K_\gamma(\mathbf{x}_{new}, \mathbf{x}_i) w_i^t}, \tag{1}$$

where N is the number of observations in Stage 1, $K_\gamma(\mathbf{x}, \mathbf{x}_i) = e^{-\gamma||\mathbf{x}-\mathbf{x}_i||^2}$ is a Gaussian kernel, w_i^t is a weight reflecting the number of households in carrier route i that were treated with treatment t in Stage 1, and y_i^t is the average effect of treatment t in Stage 1 over the households in carrier route i that were treated with treatment t.

There are four key elements to be calibrated in the kernel regression. First, we decide on the form of the conditional expectation function. In our study, we use the Nadaraya-Watson kernel estimator. Second, we decide on the kernel, i.e., the weighting function. The radial basis function, also called Gaussian, kernel is most often used and is our choice.

Third, we select a distance metric. Our goal is to demonstrate the taxonomy of segmentation techniques and their comparison, so we favor simplicity across all methods. In particular, we use the Euclidean distance for all distance-driven methods:

$$||\mathbf{x}_i - \mathbf{x}_j|| = \sqrt{(x_{i1} - x_{j1})^2 + \ldots + (x_{ip} - x_{jp})^2}, \tag{2}$$

where p is the number of targeting variables of each carrier route i.

The last parameter to be specified for the kernel regression estimator is the bandwidth γ of the kernel. We use cross-validation to find the best bandwidth. At every iteration of the cross-validation, we randomly split the Stage 1 observations into a training set (80%) and a validation set (20%). For each treatment, and for each observation in the validation set, we derive a prediction using the kernel regression estimator with the training set observations, and we compute a prediction mean squared error (MSE). The optimal bandwidth is fine-tuned separately for each treatment to minimize the average MSE over 200 cross-validation iterations.

Assigning a treatment to new observations in the Stage 2 experiment. Having cross-validated the kernel regression estimator, for each new (i.e., Stage 2) observation we derive a predicted profit based on the kernel regression, for each of the three treatments. We then assign to the new observation the treatment that results in the highest predicted profit.

B.1.2 k-Nearest Neighbors (k-NN)

Overview. The k-nearest neighbors approach approximates the effectiveness of each treatment for each new observation by averaging the profit under each treatment across the k Stage 1 observations that are the closest to the new observation. It then selects for each new observation the treatment with the highest predicted profit.

Implementation and cross-validation. In the k-nearest neighbors approach, there are two key elements to be calibrated: the distance measure and the number of neighbors k to be considered. The choice of the distance measure for the k-nearest neighbors method is discussed in Stone (1977). To be consistent across the proposed distance-based methods, and for simplicity, we use the Euclidean distance.

The optimal number of neighbors k to use generally depends upon the dimensionality of the space of explanatory variables and the distribution of explanatory variables and observations. To fine-tune the number of neighbors, we conduct cross-validation similar to the cross-validation for the kernel regression. For each cross-validation iteration and for each observation in the validation set, we identify the observation's k nearest neighbors among the training set. The observation's predicted profit under each treatment is then computed as a weighted average of the profit (under the respective treatment) of the observation's k nearest neighbors in the training set. We repeat this process for a range of different k's. The number of neighbors k is fine-tuned separately for each treatment to minimize the average MSE of the predictions over all cross-validation iterations.

Assigning a treatment to new observations in the Stage 2 experiment. For each new (i.e., Stage 2) observation we derive a predicted profit by weight-averaging profit across its k nearest neighbors from the Stage 1 observations, for each of the three treatments. We then assign to the new observation the treatment that results in the highest predicted profit.

B.1.3 Hierarchical Clustering (HC)

Overview. Hierarchical clustering is a classic greedy clustering technique that links pairs of Stage 1 observations that are in close proximity. These binary clusters are grouped into larger clusters, until a hierarchical tree is formed. The hierarchical tree is then cut to create a partition of the observations into the desired number of clusters. For each new observation, a predicted profit is derived as a weighted average of the profit of the Stage 1 observations in the cluster that are the closest to the new observation, and the new observation is assigned the treatment that achieves the highest predicted profit.

Implementation and cross-validation. Three key elements need to be calibrated: the distance measure, the linkage criterion, and the desired number of clusters. For the distance measure between pairs of observations, we use Euclidean distance to be consistent with the other distance-based methods that we employ. The linkage criterion determines how clusters will be grouped with other clusters and observations to form higher-level clusters. We use a minimum distance linkage criterion,

which sets the distance between two clusters to be the minimum distance between observations in the two clusters.

To fine-tune the number of clusters, we cross-validate using a cross-validation procedure similar to the one described previously. For each cross-validation iteration, we perform hierarchical clustering on the training set. We do so for a range of values for the number of clusters. We make predictions as follows: for each observation in the validation set, its closest cluster from the training set is identified. Then the predicted profit under each treatment is calculated as the weighted average of the profit (under the respective treatment) of the training set observations that lie in the closest cluster. The closest cluster is selected based on the minimum distance criterion. The number of clusters is fine-tuned separately for each treatment to minimize the average MSE of the predictions over all cross-validation iterations.

Assigning a treatment to new observations in the Stage 2 experiment. For each new (i.e., Stage 2) observation we derive a predicted profit by averaging profit across its closest cluster of Stage 1 observations, for each of the three treatments. We then assign to the new observation the treatment that results in the highest predicted profit.

B.2 Model-driven Methods

B.2.1 Lasso Regression

Overview. The Lasso regression, a regularized regression method proposed by Tibshirani (1996), minimizes the sum of square errors subject to a constraint on the l_1-norm. For each new observation, we predict a profit using Lasso for each treatment and assign the treatment that results in the highest predicted profit.

Implementation and cross-validation. The Lasso regression estimates for treatment t are given by

$$\hat{\beta}^t = \arg\min_{\beta} \Big((\mathbf{y}^t - \boldsymbol{X}\beta)^T W^t (\mathbf{y}^t - \boldsymbol{X}\beta) + \lambda ||\beta||_1 \Big), \tag{3}$$

where $\mathbf{y}^t$ is the effect of treatment t in Stage 1 on the households in each carrier route that were treated with treatment t, W^t is a diagonal matrix whose ith diagonal entry is w_i^t, i.e., the number of households in carrier route i that were treated with treatment t in Stage 1, and $\lambda \geq 0$ is a regularization parameter. The predicted profit for some observation i can then be calculated as

$$\hat{y}_i^t = \hat{\beta}^t \mathbf{x}_i. \tag{4}$$

We use the Glmnet implementation of the elastic net (Qian et al., 2013) to train a Lasso model. Glmnet uses cyclical coordinate descent for the optimization[1], and performs ten-fold cross-validation[2] to fine-tune hyper-parameter λ. As every observation in the dataset pertains to a different carrier route, we weight observations by the number of households in the carrier route. The hyper-parameter λ is configured separately for each treatment.

[1] Cyclical coordinate descent successively optimizes the objective function over each parameter with others fixed, and cycles repeatedly until convergence.

[2] Split the data set into ten buckets. Estimate $\boldsymbol{\beta}$ on data from nine buckets and cross-validate on the tenth. Rotate and do this for all ten buckets and calculate the average error.

Assigning a treatment to new observations in the Stage 2 experiment. Having cross-validated the Lasso estimator from Stage 1 observations, for each new (i.e., Stage 2) observation, we derive a predicted profit based on the Lasso regression, for each of the three treatments. We then assign to the new observation the treatment that results in the highest predicted profit.

B.2.2 Finite Mixture Models (FMM)

Overview. Finite mixture models express the response as a finite mixture of regression models, where the regression models can have different specifications. Maximum likelihood estimates of the segment proportions, the regression coefficients, and the distribution parameters for each segment are obtained using the expectation-maximization (EM) algorithm on Stage 1 observations. For each new observation, the finite mixture model makes a profit prediction for each treatment; we assign the treatment that results in the highest predicted profit.

Implementation and cross-validation. We assume the response variable y_i^t of carrier route i under treatment t is distributed according to the finite mixture model

$$y_i^t \sim f(y_i^t|\mathbf{x}_i;\boldsymbol{\theta}^t;\boldsymbol{\pi}^t) = \sum_{\ell=1}^{K} \pi_\ell^t f_\ell(y_i^t|\mathbf{x}_i;\boldsymbol{\theta}_\ell^t), \tag{5}$$

where $\boldsymbol{\pi}^t \geq \mathbf{0}$, $\sum_{\ell=1}^{K} \pi_\ell^t = 1$.

We use the Flexmix package in R (Grün and Leisch, 2008) to estimate the model.[3] The maximum likelihood estimation of the regression coefficients, the distribution parameters, and the weight π_ℓ^t for each segment is carried out using the EM algorithm, which iterates between evaluating the expectation of the log-likelihood using current estimates (E step), and updating the estimates to maximize the expectation of the log-likelihood (M step).

The Stage 1 promotion campaign had a low response rate, and therefore the revenue is zero for a significant number of carrier routes. To deal with zero inflation, we use revenue (and not profit) as the response variable y_i^t and consider a zero-inflated Poisson model, which is approximated by setting an intercept at the first component fixed to $-\infty$ and other coefficients to zero, while the rest of the model is a usual mixture of Poisson distributions. The estimated model takes the following form:

$$\begin{aligned} f_\ell(y_i^t|\mathbf{x}_i;\boldsymbol{\theta}_\ell^t) &= \frac{e^{-\mu_i}\mu_i^{y_i^t}}{y_i^t!}, \text{ where} \\ \log(\mu_i) &= \mathbf{x}_i'\boldsymbol{\theta}_\ell^t. \end{aligned}$$

Zero-inflated Poisson models require the response variable to be discrete. For this reason, we bucket the observed revenue. The size of the buckets is a parameter we fine-tune in cross-validation, along with the number of segments K. At each iteration of the cross-validation, the mixture model is estimated from observations in the training set, and then a prediction is calculated for observations in the validation set. The optimal parameters are selected separately for different treatments to minimize the average MSE of the predictions over all cross-validation iterations.

[3] `https://cran.r-project.org/web/packages/flexmix/index.html`

Assigning a treatment to new observations in the Stage 2 experiment. Having learnt and cross-validated the finite mixture estimator from Stage 1 observations, for each new (i.e., Stage 2) observation, we derive a predicted revenue based on the estimated model for each of the three treatments, and then subtract the mailing costs to retrieve predicted profit. We then assign to the new observation the treatment that results in the highest predicted profit.

B.3 Classification Methods

B.3.1 CHi-square Automatic Interaction Detection (CHAID)

Overview. The CHi-square Automatic Interaction Detection (CHAID), a multiway classification tree technique introduced by Kass (1976), became popular in the marketing practice because of its interpretability and convenience for segmentation analysis. CHAID recursively partitions the training observations into subsegments, maximizing at each round the significance of a chi-squared statistic for cross-tabulations between the dependent variable, which is the optimal treatment decision, and the targeting variables at each partition. By the end of the process, the Stage 1 observations are partitioned into mutually exclusive and collectively exhaustive segments that best describe the optimal treatment decision. New observations are assigned the optimal treatment of the segment in which they are classified.

Implementation and cross-validation. To obtain the decision tree, at each split CHAID looks for the targeting variable that best explains the response variable if split. In order to decide whether to create a particular split based on this variable, the algorithm performs a chi-squared test for independence between the split variable and the categorical response. If the test decides that the split variable and the response are independent, the tree stops growing; otherwise, the split is created, and the next best split is searched. The process terminates when none of the leaves can be split.

We used the CHAID package in R.[4] This requires all variables to be categorical, so we categorized continuous variables into five quantiles.

Computationally, CHAID is the most expensive method that we implemented. Cross-validation is used to select seven model parameters: the levels of significance used for merging of predictor categories and splitting of previously merged categories, the level of significance used for splitting of a node in the most significant predictor, the number of observations in split response at which no further split is desired, the minimum number and frequency of observations in terminal nodes, and the maximum height of the tree. As the number of parameters is high, we consider a small grid of three values for each parameter in the process of cross-validation. The optimal parameters were selected to maximize the average classification accuracy. We weight observations by the number of households in the corresponding carrier route (similar to other methods).

Assigning a treatment to new observations in the Stage 2 experiment. CHAID assigns new observations to classes, where each class identifies the optimal treatment.

B.3.2 Support Vector Machines (SVM)

Overview. The method first labels each Stage 1 observation according to the treatment that has the highest profit for that observation. It then divides the space of targeting variables with

[4] `https://r-forge.r-project.org/R/?group_id=343`

separating hyperplanes that separate the Stage 1 observations, so that the separation between observations of different labels is maximized. New observations are then assigned the treatment that corresponds to their spatial representation in the high-dimensional space of targeting variables.

Implementation and cross-validation. Support vector machines (SVM) is, inherently, a two-class classification technique. Given the training set of labeled pairs $(\mathbf{x}_i, z_i), i = 1, \ldots, N$, where labels $\mathbf{z} \in \{+1, -1\}^N$ indicate the class, the SVM technique finds a separating hyperplane between the two classes that is a solution to the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{\theta},\theta_0,\boldsymbol{\xi}} \quad & \frac{1}{2}\boldsymbol{\theta}^T\boldsymbol{\theta} + C\sum_{i=1}^{N}\xi_i \\ \text{subject to} \quad & z_i(\boldsymbol{\theta}^T\boldsymbol{\phi}(\mathbf{x}_i) + \theta_0) \geq 1 - \xi_i, \\ & \xi_i \geq 0, \end{aligned} \tag{6}$$

where $\boldsymbol{\phi}(\mathbf{x}_i)$ are feature vectors. We refer to $K(\mathbf{x}_i, \mathbf{x}_i) = \boldsymbol{\phi}(\mathbf{x}_i)^T\boldsymbol{\phi}(\mathbf{x}_i)$ as the kernel function. We use the Gaussian (radial basis function) kernel $K_\gamma(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma||\mathbf{x}_i - \mathbf{x}_j||^2}$, which is known to be a reasonable modeling choice for a broad range of applications, as it can handle a nonlinear relation between class labels and attributes, while having a moderate number of hyperparameters.

We use the LibSVM multiclass SVM library for MATLAB (Chang and Lin, 2011) to do a one-versus-one multi-class classification.[5] We have three classes, one for each of the three treatments; we find a two-class SVM for all $\binom{3}{2} = 3$ pairs of classes, and assign new observations to the class which is selected by the most classifiers. In the cross-validation stage, we fine-tune the misclassification penalty parameter C and the bandwidth parameter for the Gaussian kernel γ to achieve the highest prediction accuracy on the validation set. The same cost and bandwidth parameters are used for all three two-class SVMs.

Assigning a treatment to new observations in the Stage 2 experiment. Like CHAID, SVM assigns each new observation to a class, where the class identifies the optimal treatment.

B.4 Uniform Policies

We assign each new observation the same treatment. We evaluate three uniform policies: the policy assigning the $25 paid membership uniformly, the policy assigning the 120-day free trial uniformly, and the policy assigning the no-mail treatment uniformly.

[5] `https://www.csie.ntu.edu.tw/~cjlin/libsvm/`

C Sources of Variation in the Twelve-month Estimated Profit Measure

We estimated four nested equations using OLS (the equations are summarized in the table below). In all four models the unit of analysis is a household and the dependent variable is the twelve-month estimated *Profit* measure. The sample size includes all of the households in the Stage 2 experiment.

In the first equation we include a binary variable (*Mailing Cost*) indicating whether the household was in one of the two mailing conditions (120-day trial or $25 paid). The second model adds a binary variable identifying whether the household responded by signing up for a trial or regular membership. We distinguish between households that signed up for trial versus regular memberships in the third model. Finally, in the fourth model we use fixed effects identifying the amount that the households spent in the stores (if any) during the first 77 days.

For each model we calculate the percentage of variance explained (using the R^2) and in the last column of the table we report the incremental variance explained by each additional feature of the model.

Incremental Feature	**Equation**	**Variance Explained**	
		Total	**Incremental**
Mailing Cost	$Profit_i = \alpha + \beta_1 Mailing\ Cost_i + \epsilon_i$	0.01%	0.01%
Any Membership	$Profit_i = \alpha + \beta_1 Mailing\ Cost_i + \beta_2 Any\ Membership_i + \epsilon_i$	35.1%	35.1%
Membership Type	$Profit_i = \alpha + \beta_1 Mailing\ Cost_i + \beta_2 Trial\ Membership_i + \beta_3 Regular\ Membership_i + \epsilon_i$	47.0%	11.9%
Store Purchases	$Profit_i = \alpha + \beta_1 Mailing\ Cost_i + \beta_2 Trial\ Membership_i + \beta_3 Regular\ Membership_i + \beta_4 Store\ Purchases_i + \epsilon_i$	96.3%	49.3%

Notice that the total variation explained is slightly less than 100%. This is because the initial store revenue and the type of membership interact to project future spending, and so the relationship between membership type and store purchases is non-linear.

D Incorporating Budget Constraints into the Mailing Decision

Due to budget constraints, some retailers may impose a ceiling on the total number of mailings. The retailer that participated in this study did not impose any restrictions on the total number of promotional mailings sent in our experiments. However, a budget constraint could easily be accommodated by the model-driven and distance-driven methods using the following greedy algorithm:

i Produce estimates of $\hat{y}_c^{(25)}, \hat{y}_c^{(120)}, \hat{y}_c^{(no-mail)}$ for each carrier route c.

ii Calculate the estimated lift in the profit for each type of promotion: $\widehat{lift}_c^{(25)} = \hat{y}_c^{(25)} - \hat{y}_c^{(no-mail)}$ and $\widehat{lift}_c^{(120)} = \hat{y}_c^{(120)} - \hat{y}_c^{(no-mail)}$.

iii Across all of the carrier routes and the two types of promotions, rank the (carrier route, promotion) combinations according to the largest lift.

iv Select the (carrier route, promotion) combination with the largest lift, and use the promotion associated with this combination. If the remaining mailing budget is insufficient for the entire carrier route, send to as many households as possible (selecting households at random). Keep track of the total number of mailings, and omit both (carrier route, promotion) combinations for this carrier route from the ranking used to select the next combination.

v Repeat step (iv) with the next best (carrier route, promotion) combination until the budget constraint is met (or there are no carrier routes left).

E Average Profit in Each Experimental Condition

Table 2: Average Profit in Each Experimental Condition

	Average	Standard Error	Sample Size
CHAID	100.00	3.47	436 832
SVM	107.38	3.16	424 875
HC	108.00	3.90	402 804
Kernel	110.57	3.47	407 838
FMM	114.09	4.12	417 677
k-NN	114.92	4.01	390 328
Lasso	118.16	3.94	424 989
Uniform No Mail	83.36	3.44	412 795
Uniform 120-day free	90.76	3.65	404 182
Uniform $25 paid	106.51	4.43	396 924

The table reports the average profit and standard error averaged across the households in each experimental condition. To preserve confidentiality, the profits are indexed to 100 for the CHAID data point.

Table 3: Average Profit in Each Experimental Condition — Taxonomy of Methods

	Average	Standard Error	Sample Size
Classification	100.00	2.26	861 707
Distance-Driven	107.22	2.12	1 200 970
Model-Driven	112.06	2.75	842 666

The table reports the average profit and standard error when pooling the households using the taxonomy of methods. To preserve confidentiality, the profits are indexed to 100 for the Classification data point.

F Comparison Between Lasso and the Uniform $25 Policy: Who Are the Customers that Lasso Does Not Mail to?

If we focus just on customers in the Lasso and $25 experimental conditions, there are 150,956 households for which Lasso would have recommended not mailing. Of these, 75,190 were randomly assigned to the Lasso condition (and so were not mailed) and 75,766 were randomly assigned to the Uniform $25 condition (and so were mailed the $25 promotion). We compare these 150,956 customers with the other customers who were randomly assigned to the Lasso and $25 experimental conditions. The findings are summarized in the table below.

Table 4: Comparison Between Lasso and the Uniform $25 Policy and Targeting Variables

	Lasso: $25 Paid	Lasso: 120-day Trial	Lasso: No Mail	Lasso: $25 Paid or 120-day Trial	Difference: No Mail vs. $25 Paid or 120-day Trial
Age	56.87	56.62	57.69	56.85	0.85*
					(0.36)
Home Value (in 1000s)	246.75	264.35	275.23	248.30	26.93**
					(10.09)
Income (in 1000s)	84.17	92.93	86.91	84.95	1.97
					(3.60)
Single Family	0.7825	0.7694	0.7923	0.7814	0.0109
					(0.0174)
Multi-Family	0.2140	0.2243	0.2026	0.2149	−0.0124
					(0.0174)
Distance	7.8908	12.8340	16.9085	8.3264	8.5821**
					(0.4602)
Comp. Distance	10.8807	10.8542	10.6214	10.8784	−0.2569
					(0.6906)
Penetration Rate	0.5791	0.1914	0.4283	0.5449	−0.1166
					(0.1368)
3yr Response	18.0053	4.7263	1.8928	16.8350	−14.9422**
					(0.4877)
F Flag	0.4452	0.8533	0.8223	0.4812	0.3412**
					(0.0283)
M Flag	0.3589	0.1200	0.1576	0.3378	−0.1802**
					(0.0268)
Past Paids	0.0505	0.0195	0.0110	0.0478	−0.0367**
					(0.0017)
Trialists	0.0051	0.0003	0.0002	0.0046	−0.0044**
					(0.0004)
Sample Size	1,552	150	379	1,702	

The table reports the average of each of the thirteen targeting variables for carrier routes that were randomly assigned to the Lasso and Uniform $25 experimental conditions. The unit of analysis is a carrier route. Carrier routes are grouped according to the action recommended by Lasso. Significance: † for $p < 0.1$; * for $p < 0.05$; ** for $p < 0.01$.

The findings reveal that the customers to whom Lasso would have recommended not mailing

are in carrier routes located a long way from the retailer's stores and that have had historically very low response and membership rates. Lasso chooses not to mail to these customers because they are unlikely to respond even if they receive a promotional mailing.

G Comparison Between Optimized Policies and the Uniform $25 Policy

We first pool the households in the optimized condition with the households in the uniform $25 paid condition, and then ask what treatment the optimized policy would have recommended. This yields sub-groups of households, within which we can compare the uniform $25 paid and optimized policy outcomes.

Table 5: Comparison with the Uniform $25 Policy

	Optimized Policy: $25 paid offer	Optimized Policy: 120-day free trial	Optimized Policy: no mail	Optimized Policy: 120-day free trial or no mail
CHAID	3.182	15.066†	−11.042**	−5.147
	(8.984)	(8.984)	(4.211)	(3.837)
SVM	9.920	−415.497**	−2.433	−2.995
	(13.195)	(154.127)	(3.088)	(3.088)
HC	−1.029	100.505	17.032	56.335*
	(3.743)	(66.161)	(12.259)	(26.670)
Kernel	2.807	39.304**	4.305	12.259**
	(5.241)	(13.382)	($3.743)	($3.930)
k-NN	3.743	9.732	10.200†	7.674†
	(5.708)	(7.861)	(5.428)	(4.492)
FMM	1.965	36.122	18.622**	18.716**
	(5.428)	(52.124)	(3.088)	(3.088)
Lasso	5.521	7.580	20.868**	17.406**
	(4.960)	(5.802)	(2.246)	(2.340)

The table focuses on households in the uniform $25 paid and each optimized policy condition. Households are grouped according to the treatment recommended by the optimized policy. The table reports the difference in profit for the (randomly assigned) sub-group that received the optimized policy and the randomized sub-group that received the uniform $25 treatment. A positive value indicates that profits were higher in the optimized policy sub-group. To preserve confidentiality, the profits in each condition use the same indexing as Figure 4.3 in the paper. Significance: † for $p < 0.1$; * for $p < 0.05$; ** for $p < 0.01$. Standard errors are in parentheses and sample sizes are reported in Table 6.

For the households for which an optimized policy recommends sending the $25 paid offer, we do not observe any significant differences between the optimized policy and the uniform $25 paid condition. As we discussed in the paper, this is reassuring, as this comparison serves as a randomization check. Any differences in these outcomes could only be attributed to differences in the households (there is no difference in the experimental treatment). For the households for which the optimized policy did not recommend sending the $25 paid offer, there is a sharp difference in outcomes for the best-performing and worst-performing methods. For HC, Kernel, FMM, k-NN, and Lasso, the optimal methods consistently outperformed the uniform policy. However, for CHAID and SVM the reverse occurred: when these methods chose not to mail the $25 paid offer, the uniform policy tended to perform better than these optimized methods. Notably these were also the two methods that chose to mail the $25 paid offer least frequently; they mailed the $25 paid offer to less than 20% of the households, while all of the other methods mailed this promotion to over 55% of the households. In Section 5.4 we provide an explanation for these differences.

Table 6: Sample Sizes of Comparison with the Uniform $25 Policy

	Optimized Policy: $25 paid offer	Optimized Policy: 120-day free trial	Optimized Policy: no mail	Optimized Policy: 120-day free or no mail
Optimized Policy				
CHAID	82,764	64,288	289,780	354,068
SVM	74,819	615	349,441	350,056
HC	383,023	9,411	10,370	19,781
Kernel	236,849	34,723	136,266	170,989
FMM	278,680	886	138,111	138,997
k-NN	225,956	43,032	121,340	164,372
Lasso	310,280	39,519	75,190	114,709
Uniform $25 Paid				
CHAID	75,451	40,957	280,516	321,473
SVM	66,454	437	330,033	330,470
HC	383,830	4,145	8,949	13,094
Kernel	259,008	24,204	113,712	137,916
FMM	277,081	767	119,076	119,843
k-NN	230,300	57,043	109,581	166,624
Lasso	294,919	26,239	75,766	102,005

The table reports the sample sizes for the analysis in which we group households in the uniform $25 paid and each optimized policy condition according to the treatment recommended by the optimized policy.

H Relationship between Profit and Targeting Variables

In the table below we report estimated coefficients when regressing the *Profit* outcome measure on the thirteen targeting variables. The unit of analysis in the models is a carrier route and the model is estimated separately for each treatment condition using the 5,976 carrier routes in the Stage 1 experiment. We normalize the targeting variables to zero mean and unit variance to demonstrate the relative predictive power of the variables, and scale the profits in the three conditions to preserve confidentiality. The coefficients can be interpreted as the expected change in *Profit* associated with an increase by one standard deviation in each targeting variable (holding the other variables constant).

Table 7: Stage 1 Profits and the Thirteen Targeting Variables

	Profit: No Mail	Profit: $25 Paid	Profit: 120-day Trial
Intercept	100.000**	210.996**	131.142**
	(6.318)	(10.165)	(8.499)
M Flag	−12.720	−52.688**	−18.951
	(9.764)	(15.710)	(13.135)
F Flag	−11.032	−75.463**	−44.127**
	(12.171)	(19.584)	(16.373)
Distance	−47.171**	−12.078	1.130
	(15.568)	(25.048)	(20.941)
Comp. Distance	51.022**	43.126*	13.083
	(13.388)	(21.541)	(18.009)
Single Family	2.781	46.280**	56.682**
	(8.628)	(13.882)	(11.606)
Multi-Family	−0.763	−33.251*	−10.421
	(9.243)	(14.872)	(12.434)
Past Paids	0.174	62.455**	25.939*
	(8.579)	(13.803)	(11.540)
Trialists	2.320	3.266	−1.202
	(7.266)	(11.691)	(9.775)
Income	−4.343	−68.539**	−27.849
	(13.763)	(22.145)	(18.514)
Home Value	1.280	6.041	9.813
	(12.907)	(20.767)	(17.362)
Age	−11.983†	−43.454**	−45.133**
	(6.470)	(10.410)	(8.703)
Penetration Rate	−7.307	−20.594†	−11.108
	(6.554)	(10.545)	(8.816)
3yr Response	44.717**	213.516**	135.959**
	(10.515)	(16.918)	(14.144)

The table reports coefficients from an OLS model with *Profit* as the dependent variable. The unit of analysis is a carrier route and the model is estimated separately for each treatment condition using the carrier routes in the Stage 1 experiment. The sample size in all models is 5,976 (carrier routes). The targeting variables are all scaled to zero mean and unit variance. To preserve confidentiality, the profits are indexed to 100 for the average profit in the no-mail condition. Standard errors are in parentheses. Significance: † for $p < 0.1$; * for $p < 0.05$; ** for $p < 0.01$.

I Covariate Shift: Performance Inside vs. Outside the Range of the Training Data

Table 8: Average Profit Inside vs. Outside the Range of the Training Data

		Average	Standard Error	Sample Size
Inside the Range	CHAID	94.77	5.27	175 405
	SVM	97.81	5.41	159 955
	HC	101.67	6.53	158 423
	Kernel	114.83	6.56	161 461
	k-NN	120.37	7.19	145 971
	FMM	137.00	8.38	182 027
	Lasso	140.48	8.16	164 005
Outside the Range	CHAID	159.53	6.88	261 427
	SVM	170.90	5.91	264 920
	HC	171.79	7.48	244 381
	Kernel	169.14	6.36	246 377
	k-NN	173.21	7.40	244 357
	FMM	164.20	7.30	235 650
	Lasso	168.65	6.85	260 984

The table reports the average Stage 2 *Profit*, standard error and sample size when grouping the households according to whether they are inside or outside the range of the training data. To preserve confidentiality, the profits are indexed to 100 in the No Mail control for the Inside the Range data point. We categorize the households in the Stage 2 validation data as falling inside the range of the Stage 1 training data if they have values on all thirteen targeting variables within two standard deviations of the average value in the training data.

Table 9: Average Profit Inside vs. Outside the Range of the Training Data — Taxonomy of Methods

		Average	Standard Error	Sample Size
Inside the Range	Classification	89.82	3.53	335 360
	Distance-Driven	104.63	3.64	465 855
	Model-Driven	129.42	5.47	346 032
Outside the Range	Classification	90.99	2.49	526 347
	Distance-Driven	94.35	2.26	735 115
	Model-Driven	91.69	2.75	496 634

The table reports the average Stage 2 *Profit*, standard error and sample size when grouping the households according to whether they are inside or outside the range of the training data, and when pooling the households using the taxonomy of methods. To preserve confidentiality, the profitsare indexed to 100 for the most profitable uniform condition (in that group of carrier routes). We categorize the households in the Stage 2 validation data as falling inside the range of the Stage 1 training data if they have values on all thirteen targeting variables within two standard deviations of the average value in the training data.

J Additional Covariate Shift Results

The following analysis complements the analysis of the covariate shift problem presented in Section 5.1. We investigate which targeting variables most frequently shifted between the two stages of the study, and ask whether deviations on some variables had a larger impact on the performance of the targeting methods than deviations on other variables.

In the table below we summarize the frequency of deviations for each targeting variable. We distinguish positive and negative deviations, and report the frequency of times in the Stage 2 validation data that each variable deviated more than two standard deviations from the average in the Stage 1 training data.

Table 10: Frequency of Deviations from the Training Data Means

		Positive Deviations	**No Deviation**	**Negative Deviations**
Demographics	Age	5.6%	92.2%	2.2%
	Home Value	22.2%	77.8%	0.0%
	Income	13.0%	87.0%	0.0%
	Single Family	5.9%	94.1%	0.0%
	Multi-Family	11.6%	88.4%	0.0%
Distance	Distance	16.3%	83.7%	0.0%
	F Flag	0.0%	90.1%	9.9%
	M Flag	16.7%	83.3%	0.0%
	Comp. Distance	10.9%	89.1%	0.0%
Past Response	Penetration Rate	0.0%	100.0%	0.0%
	3 yr Response	0.0%	100.0%	0.0%
	Past Paids	4.6%	95.4%	0.0%
	Trialists	10.0%	90.0%	0.0%

The table reports the percentage of times that each of the targeting variables in the Stage 2 validation data deviated from the Stage 1 training data average by more than two standard deviations. "No Deviation" indicates the variable was within two standard deviations of the training data mean. The table distinguishes positive and negative deviations from the training data mean. The unit of analysis is a carrier route and the sample size is 10,419 carrier routes.

The most common deviations are *Income* and *Home Value* (higher in the validation data), distance to the competitors' and retailer's own stores (higher in the validation data), and the proportion of past trialists in the carrier route (higher in the validation data).

In the next table we report the average performance of the targeting methods when the validation data deviates along each of these dimensions. We group the highly correlated variables together, and identify carrier routes in which deviations occurred for just those variables (i.e., the other targeting variables did not deviate more than two standard deviations from the training data mean).[6] The deviations are all positive, except for *Age*, where we report separate results for both positive and negative deviations.

We report the difference between the naïve benchmark and the profit from the targeting methods (grouped by taxonomy). The naïve benchmark reports the average profits earned per household

[6]Because they are highly correlated, we group the *Income* and *Home Value* together. We also group the *Penetration Rate* and *3yr Response* for the same reason.

under the most profitable uniform policy. To preserve confidentiality the profits are multiplied by a (common) random number. For example, a value of -19% for the model-driven methods in the *More Trialists* row indicates that when the proportion of *Trialists* in a carrier route was more than two standard deviations higher than the training data mean, then the average profit from the model-driven methods was 19% less than the most profitable uniform policy.

Table 11: Profitability of Targeting Methods Relative to the Naïve Benchmark When the Validation Data Deviates from the Training Data Means

	Naïve Benchmark	**Classification**	**Distance-Driven**	**Model-Driven**
Greater Distance to a Store	\$0.17	−42%	−143%	−81%
More Multi-Family Dwellings	\$0.58	−42%	12%	−31%
Higher Income and Home Value	\$0.81	16%	10%	30%
Older Age	\$0.87	−26%	−21%	−33%
Younger Age	\$0.95	34%	41%	16%
Greater Distance to Competitors	\$1.21	25%	7%	27%
More Trialists	\$2.51	−25%	−30%	−19%
Higher Penetration Rate and 3yr Response	\$3.98	−16%	−13%	−13%
More Past Paids	\$4.38	−42%	−42%	−25%

The table summarizes the Stage 2 average *Profit* when restricting attention to carrier routes that deviate on the targeting variables by more than two standard deviations from the training data means, and when pooling using the taxonomy of methods. The *Naïve Benchmark* describes the average profitability of the most profitable uniform policy (multiplied by a common random number). All deviations are positive, except for *Age*, where we report separate results for both positive and negative deviations. The unit of analysis is a household.

Across all Stage 2 carrier routes, the average profit in the 120-day trial promotion and \$25 paid promotions were \$1.05 and \$1.24 respectively (recall that these numbers are scaled by a random number). This suggests that we can classify deviations in the targeting variables (from the training data mean) into three types according to the profitability of the deviations, as in Table 12.

Table 12: Grouping of Deviations according to Profitability

Profitability of parameter space region	**Deviations on Targeting Variables**
Low	*Greater Distance to a Store*, *More Multi-Family Dwellings*
Moderate	*Higher Income and Home Value*, *Older Age*, *Younger Age*, *Greater Distance to Competitors*
High	*More Trialists*, *Higher Penetration Rate and 3yr Response*, *More Past Paids*

This grouping of the deviations in the Stage 2 carrier routes from the Stage 1 training data is related to our earlier findings describing how the Stage 1 profits in each condition varied with the thirteen targeting variables (reported earlier in the Appendix).[7]

[7]For example, increases in the *3yr Response* variable are associated with the largest change in Stage 1 profits in

We might expect that the targeting methods will produce less optimal policies the more extreme the deviations. This is consistent with the pattern in Table 11; deviations to either high or low value regions result in the targeting methods underperforming the naïve benchmarks. In contrast, deviations to regions of average value do not lead to the same deterioration in performance. In these deviations the targeting methods are generally able to continue to outperform the naïve benchmarks (with the exception of *Older Age* deviations).

Table 7. This is consistent with positive deviations yielding higher benchmark profits in Table 11. However, while many of the changes in the benchmark profits are consistent with the parameters reported in Table 7, there are exceptions. For example, higher *Age* is associated with significantly lower profits in Table 7, yet deviations to *Older Age* and *Younger Age* do not exhibit a large difference in the profitability of the naïve benchmark in Table 11. This may indicate a nonlinear relationship between the average age and the *profit*.

K Concept Shift: Did the Performance of the Model-Driven Methods Deteriorate Faster than the Other Methods?

These findings complement the concept shift analysis in Section 5.2. We estimate the following OLS models:

$$\begin{aligned}
\textit{Indexed Stage 2 Profit}_i &= \alpha + \beta_1 \textit{Positive Growth}_i + \epsilon_i && (7)\\
\textit{Indexed Stage 2 Profit}_i &= \alpha + \beta_1 \textit{Positive Growth}_i + \beta_2 \textit{Classification}_i + \beta_3 \textit{Distance}_i \\
&\quad + \beta_4 \textit{Classification}_i * \textit{Positive Growth}_i \\
&\quad + \beta_5 \textit{Distance}_i * \textit{Positive Growth}_i + \epsilon_i && (8)\\
\textit{Indexed Stage 2 Profit}_i &= \alpha + \beta_1 \textit{Negative Growth}_i + \epsilon_i && (9)\\
\textit{Indexed Stage 2 Profit}_i &= \alpha + \beta_1 \textit{Negative Growth}_i + \beta_2 \textit{Classification}_i + \beta_3 \textit{Distance}_i \\
&\quad + \beta_4 \textit{Classification}_i * \textit{Negative Growth}_i \\
&\quad + \beta_5 \textit{Distance}_i * \textit{Negative Growth}_i + \epsilon_i && (10)
\end{aligned}$$

We estimate Equations (7) and (8) using the *Flat Growth* and *Positive Growth* carrier routes, and Equations (9) and (10) using the *Flat Growth* and *Negative Growth* carrier routes. The *Positive Growth* and *Negative Growth* variables are binary indicators identifying carrier routes with positive and negative growth (respectively). Similarly, the *Distance* and *Classification* variables are binary indicators identifying the distance-driven and classification methods. In all four models the dependent variable is the indexed Stage 2 *Profit* (indexed at 100 in the no mail treatment with *Flat Growth*).

Equations (7) and (9) measure whether profits are significantly lower in the *Positive Growth* and *Negative Growth* carrier routes, compared to the *Flat Growth* carrier routes. In Equations (8) and (10) the coefficients of interest are β_4 and β_5. Positive coefficients on these variables indicate that the deterioration in performance when moving from a *Flat Growth* carrier route to a *Positive* or *Negative Growth* carrier route is larger among the model-driven methods than the classification or distance-driven methods. The findings are reported in the table below.

In Equations (7) and (9) the *Positive Growth* and *Negative Growth* coefficients confirm that profits are lower in carrier routes for which revenue among existing customers changed between the two stages. This is true for either positive or negative revenue changes. In Equation (8) the positive and significant *Distance* * *Positive Growth* coefficient confirms that the deterioration in profit in the *Positive Growth* carrier routes is larger for the model-driven methods than the distance-driven methods. In model (10) we also see that the model-driven methods deteriorate more than the classification methods when growth is negative.

Table 13: Concept Shift: Did the Performance of the Model-Driven Methods Deteriorate Faster than the Other Methods?

	Equation (7)	Equation (8)	Equation (9)	Equation (10)
Intercept	0.713**	0.840**	0.713**	0.593**
	(0.049)	(0.083)	(0.052)	(0.023)
Positive Growth	−0.308**	−0.495**		
	(0.056)	(0.097)		
Negative Growth			−0.413**	−0.646**
			(0.073)	(0.127)
Classification		0.191		−0.191
		(0.127)		(0.135)
Distance		0.197†		−0.197
		(0.115)		(0.122)
Classification * Positive Growth		0.194		
		(0.145)		
Distance * Positive Growth		0.343**		
		(0.132)		
Classification * Negative Growth				0.597**
				(0.195)
Distance * Negative Growth				0.228
				(0.170)

The table reports the coefficients from estimating Equations (7), (8), (9), and (10). The unit of analysis is a household and the dependent variable is the Stage 2 *Profit* indexed at 100 in the no mail (control) condition with *Flat Growth*. We restrict attention to households in carrier routes that participated in both stages of the experiment. Standard errors are in parentheses. The sample sizes are 240,037 (Equations (7) and (8)) and 112,342 (Equations (9) and (10)). Significance: † for $p < 0.1$; * for $p < 0.05$; ** for $p < 0.01$.

L Concept Shift: Results at the Method Level

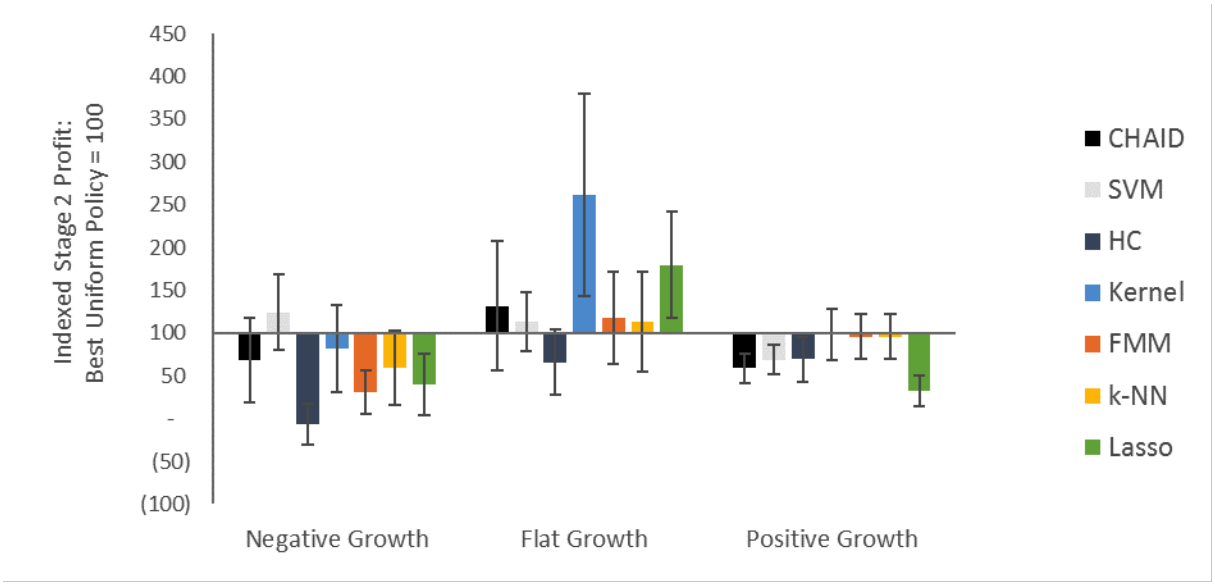

Figure 1: This figure illustrates the average Stage 2 *Profit* earned from carrier routes that participated in both stages of the study, for each of the seven optimized methods. *Profit* is indexed at 100 in the optimal uniform policy (for that *Revenue Change* group). The error bars indicate 95% confidence intervals.

M Aggregation of the Targeting Variables

M.1 Complete Findings When Using a Median Split

Table 14: Average Profit for Small and Large Carrier Routes

		Average	Standard Error	Sample Size
Below Median Size (More Info)	CHAID	114.13	4.96	212 627
	SVM	118.67	4.92	212 679
	HC	117.24	5.59	212 426
	Kernel	130.21	5.65	199 000
	k-NN	131.04	5.36	202 248
	FMM	134.82	6.78	199 099
	Lasso	147.61	6.66	216 888
Above Median Size (Less Info)	CHAID	108.99	5.88	224 205
	SVM	120.78	5.03	212 196
	HC	123.94	6.78	190 378
	Kernel	116.67	5.31	208 838
	k-NN	125.10	7.22	191 229
	FMM	120.06	6.24	215 429
	Lasso	115.21	5.66	208 101

The table reports the average Stage 2 *Profit*, standard error and sample size when grouping the carrier routes according to whether they contain more or less than the median number of households. To preserve confidentiality, the profits are indexed to 100 in the No Mail - Below Median Size data point.

Table 15: Average Profit for Small and Large Carrier Routes — Taxonomy of Methods

		Average	Standard Error	Sample Size
Below Median Size (More Info)	Classification	116.40	3.49	425 306
	Distance-Driven	125.97	3.20	610 525
	Model-Driven	141.44	4.75	419 136
Above Median Size (Less Info)	Classification	114.72	3.89	436 401
	Distance-Driven	121.74	3.71	590 445
	Model-Driven	117.67	4.22	423 530

The table reports the average Stage 2 *Profit*, standard error and sample size when grouping the carrier routes according to whether they contain more or less than the median number of households, and when pooling the households using the taxonomy of methods. To preserve confidentiality, the profits are indexed to 100 in the No Mail - Below Median Size data point.

M.2 Robustness Checks

We repeat the analysis using the top and bottom 25% of carrier routes, and using the top and bottom 10% of carrier routes.

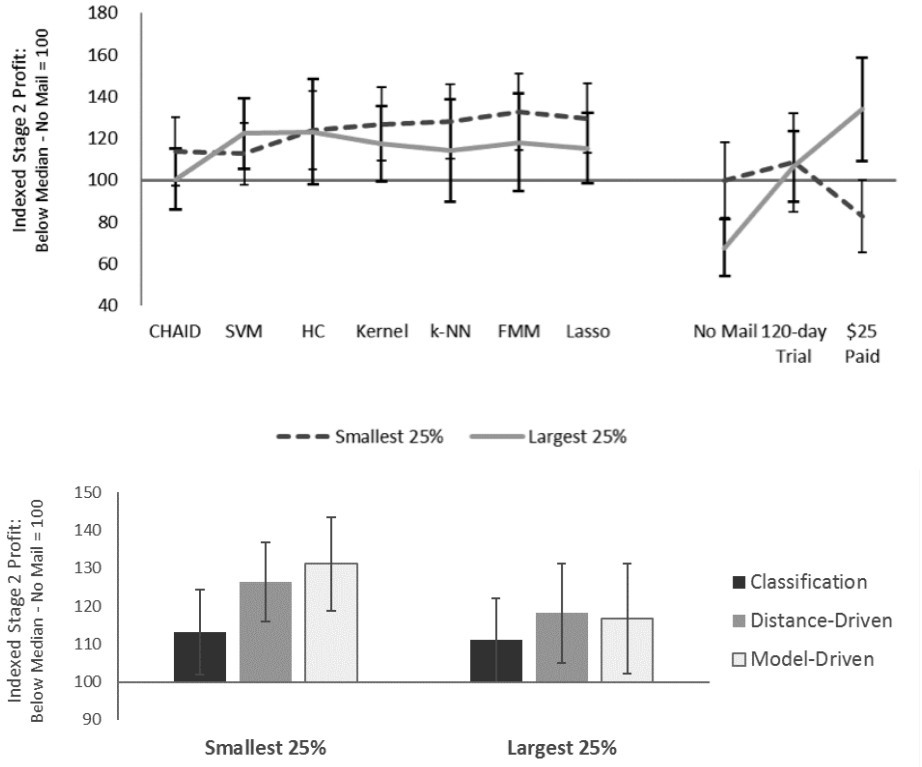

Figure 2: The average Stage 2 *Profit* for the top and bottom 25% of carrier routes (in terms of the number of households in each carrier route). The profits are indexed to 100 in the No Mail - Smallest 25% data point. The error bars indicate 95% confidence intervals.

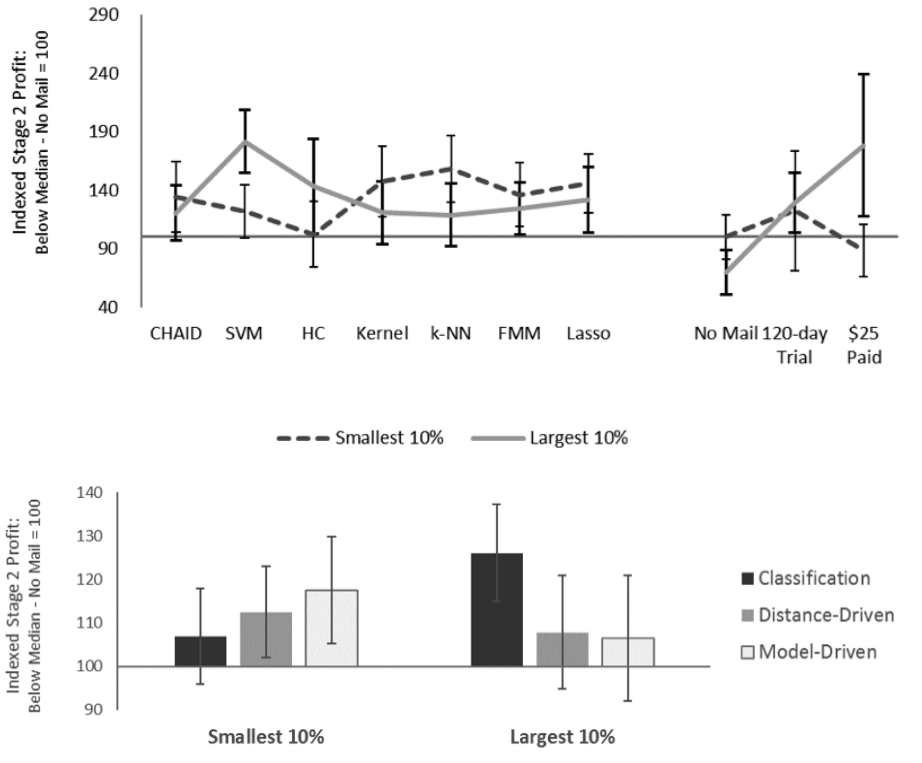

Figure 3: The average Stage 2 *Profit* for the top and bottom 10% of carrier routes (in terms of the number of households in each carrier route). The profits are indexed to 100 in the No Mail - Smallest 10% data point. The error bars indicate 95% confidence intervals.

N Additional Methods and Covariate Shift, Concept Shift, and Aggregation Analyses

N.1 Additional Methods

The five additional targeting methods we evaluate include: Lasso with interactions, adjusted SVM, random forests, XGBoost, and neural networks.

Lasso with interactions incorporates all pairwise interaction terms to the model specification of the Lasso regression. The adjusted SVM is an SVM that asymmetrically penalizes false positives and false negatives while training the classification model.

Random forests are an ensemble learning method for classification (as well as regression) that operate by constructing multiple decision trees, and choosing the majority class (or mean prediction) of the respective decisions of the individual trees. We use the randomForest package in R.[8]

XGBoost is a gradient boosting method that incrementally builds an ensemble of weak trees by training each new tree to emphasize the training instances that previous trees mis-classified. We use the xgboost package in R.[9]

Neural networks are networks of connected nodes, with hidden layers between the input and output layer, where the output of each node is computed by some non-linear function of its inputs. Neural networks can model complex non-linear relationships. We use the H2O package in R.[10]

N.2 Covariate Shift

We returned to the covariate shift analysis in which we identified the carrier routes in the Stage 2 validation data for which one or more of the variables was at least two standard deviations away from the (training data) mean. Using just these Stage 2 carrier routes, we repeated our comparison of the five additional methods with standard Lasso. We then also repeated the comparisons using the Stage 2 carrier routes that were inside the range of the training data. The findings are summarized in Table 16. To preserve confidentiality, all of the numbers are indexed by setting the average profit for standard Lasso in Stage 2 at 100.

Out of the five additional methods, only neural networks performs significantly better than Lasso inside the range of the training data, and neural networks and adjusted SVM perform significantly worse than Lasso outside the range of the training data. This offers additional support for the conclusion that the performance of Lasso is very good and is hard to improve upon.

We also observe that the methods that offer the largest performance improvement over standard Lasso inside the range, have the largest reduction in performance outside the range. Figure 4 compares the performance improvement over standard Lasso inside the range with the performance improvement over standard Lasso outside the range.

The pairwise correlation between improvement over standard Lasso inside the range and improvement over standard Lasso outside the range is -0.925. This is consistent with the argument that methods that make the best use of the information in the training data have the greatest deterioration in performance when the information in the training data deteriorates.

[8] `https://cran.r-project.org/web/packages/randomForest/index.html`

[9] `https://cran.r-project.org/web/packages/xgboost/index.html`

[10] `https://cran.r-project.org/web/packages/h2o/index.html`

Table 16: Performance Improvement of Additional Methods over Standard Lasso, Overall and for the Covariate Shift, Concept Shift, and Aggregation Analyses

	Overall	Covariate Shift		Concept Shift			Aggregation	
		Inside the Range	Outside the Range	Negative Growth	Flat Growth	Positive Growth	Below Median Size	Above Median Size
Adjusted SVM	−1.166	3.209	−5.610***	−9.508	−9.588	−45.747***	2.500	-4.894**
	(2.314)	(4.809)	(1.824)	(9.309)	(6.794)	(14.637)	(3.847)	(2.420)
XGBoost	−1.785	1.527	−4.420	−15.797**	7.026	−8.708	−3.022	0.172
	(2.368)	(2.540)	(3.729)	(6.544)	(7.116)	(16.977)	(2.269)	(4.452)
Neural Network	−2.956	4.486**	−8.077**	5.923	14.538*	2.639	−0.475	−4.861
	(2.526)	(2.287)	(4.119)	(3.871)	(7.815)	(7.946)	(2.128)	(4.588)
Random Forest	−0.589	3.758	−4.630	−8.496	1.990	−35.063**	3.802	−5.247
	(2.717)	(3.286)	(4.132)	(13.166)	(7.344)	(14.111)	(2.964)	(4.629)
Lasso with Interactions	0.271	−0.454	0.657	-	−1.630	-	0.774	−0.851
	(0.560)	(0.932)	(0.686)		(4.052)		(0.764)	(0.820)

The table compares the increase (or decrease) in average *Profit* for each of the five additional policies compared to standard Lasso, overall as well as for the covariate shift, concept shift, and aggregation analyses. The standard errors of these *Profit* differences are also reported. To preserve confidentiality, *Profits* are indexed to 100 for the standard Lasso average *Profit* in the Stage 2 experiment. Negative values indicate that standard Lasso is more profitable than the other policies. Significance: * for $p < 0.05$; ** for $p < 0.01$; *** for $p < 0.001$.

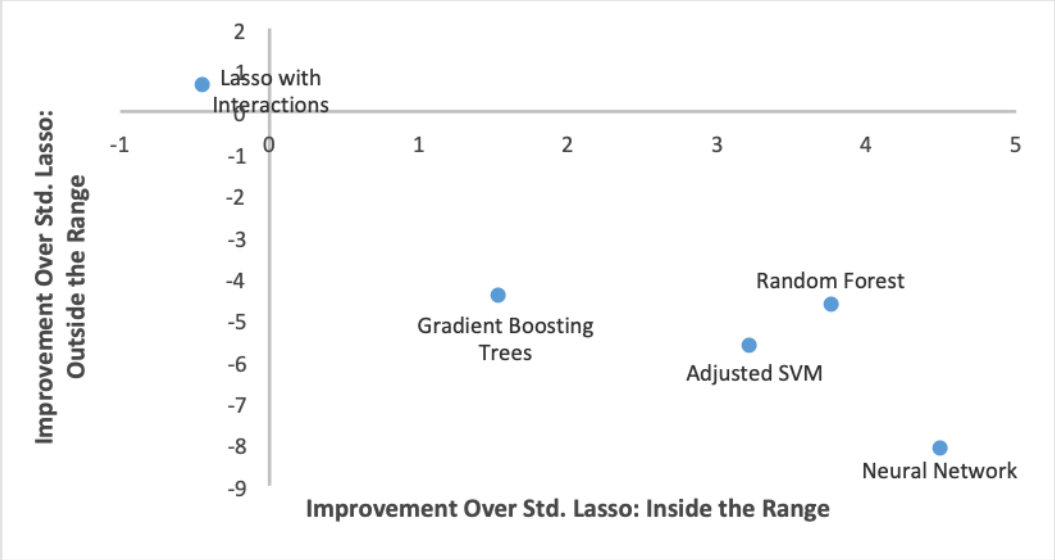

Figure 4: The x-axis measures the performance of the additional methods compared to standard Lasso on the carrier routes inside the range. The y-axis measures the performance of the additional methods compared to standard Lasso outside the range.

N.3 Concept Shift

Figure 5 reports the increase in average profits of the additional methods compared to standard Lasso for the three *Revenue Change* groups.[11] The findings replicate the evidence in the paper

[11] We exclude Lasso with interactions. In the concept shift sample, the assignments of Lasso and Lasso with interactions differ only in a single carrier route. We thus cannot estimate their difference in performance in the

that the methods perform best when *Revenue Change* is flat, but performance deteriorates when *Revenue Change* is negative or positive.

Figure 5: Performance improvement of the additional methods over standard Lasso for the three *Revenue Change* groups.

We can also evaluate whether the methods that performed best when revenue growth is flat suffer the largest deterioration in performance when revenue growth is positive or negative. The evidence here is mixed. XGBoost and the neural network offer the largest profit improvement over Lasso when revenue growth is flat. These two methods also experience a larger profit decrease than adjusted SVM and random forest when revenue growth is negative. However, this is not true when revenue growth is positive.

N.4 Aggregation of the Targeting Variables

Figure 6 compares the performance of the five additional methods on below-median-size carrier routes against above-median-size carrier routes.

We observe that the methods that offer the largest performance improvement over standard Lasso in the small carrier routes, have the largest reduction in performance in the large carrier routes. The pairwise correlation between improvement over standard Lasso in the small carrier routes and improvement over standard Lasso in the large carrier routes is -0.724. This is again consistent with the argument that methods that make the best use of the information in the training data have the greatest deterioration in performance when the information in the training data deteriorates.

segment where two policies disagree.

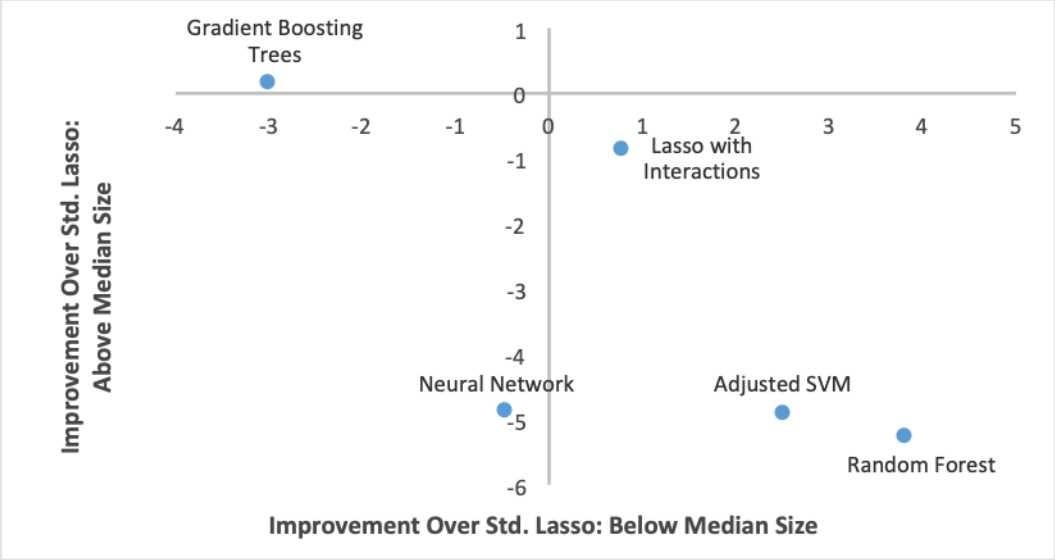

Figure 6: The x-axis measures the performance of the additional methods compared to standard Lasso on the small carrier routes (below median size). The y-axis measures the performance of the additional methods compared to standard Lasso on the large carrier routes (above median size).

References

Chang, Chih-Chung and Chih-Jen Lin (2011), "LIBSVM: A library for support vector machines." *ACM Transactions on Intelligent Systems and Technology*, 2, 1–27.

Grün, Betina and Friedrich Leisch (2008), "Flexmix version 2: Finite mixtures with concomitant variables and varying and constant parameters." *Journal of Statistical Software*, 28, 1–35.

Kass, Gordon V. (1976), *Significance Testing in, and Some Extensions of, Automatic Interaction Detection*. Doctoral dissertation, University of Witwatersrand, Johannesburg, South Africa.

Qian, J., T. Hastie, J. Friedman, R. Tibshirani, and N. Simon (2013), "Glmnet for MATLAB." `http://www.stanford.edu/~hastie/glmnet_matlab/`.

Stone, C.J. (1977), "Consistent nonparametric regression." *The Annals of Statistics*, 5, 595–645.

Tibshirani, Robert (1996), "Regression shrinkage and selection via the lasso." *Journal of the Royal Statistical Society. Series B*, 58, 267–288.