Strategy Free Machine Learning

-

March 23, 2020

1 Literature Review

Previous work on mechanism design for machine learning with strategic data sources focus on designing robust algorithms to incentivize the data providers to report their private data truthfully. Their models mainly differ in the objective and the possible actions of the data providers (agents) and the machine learner (principal).

- The first group of papers focuses on principal-agent problems in which each agent's private data point is the agent's type that the agent cannot change. The only action the agents can take is whether to report their private information truthfully.

1. Some models assume the agents' feature vectors are public, but their labels are private. Perote and Perote-Pena (2004), Chen, Podimata, Procaccia, and Shah (2018), and Gast, Ioannidis, Loiseau, and Roussillon (2013) focus on strategy-proof linear regression algorithms and introduced clockwise repeated median estimators, generalized resistant hyperplane estimators, and modified generalized linear squares estimators. Dekel, Fischer, and Procaccia (2010) investigates the general regression problem with empirical risk minimization and absolute value loss. All the previously mentioned papers assume the labels are continuous variables (regression problems), and Meir, Procaccia, and Rosenschein (2012) assumes the labels are discrete variables (classification problems) and proposes a class of random dictator mechanisms.

2. Some models assume the agents' feature vectors are also private. Chen, Liu, and Podimata (2019) investigates such problems for linear regressions.

3. Other models do not distinguish between feature vectors and labels. Each agent has a private valuation. These problems are usually modeled as facility locations problems and the solution involves some variant of the Vickrey-Clarke-Groves or Meyerson auction. These include Dütting, Feng, Narasimhan, Parkes, and Ravindranath (2017), Golowich, Narasimhan, and Parkes (2018), Epasto, Mahdian, Mirrokni, and Zuo (2018), and Procaccia and Tennenholtz (2009).

- The second group papers focus on moral-hazard problems in which each agent does not have a type but they can choose an action (with a cost) that affects the probability of obtaining the correct label. Richardson, Rokvic, Filos-Ratsikas, and Faltings (2019) focuses on the linear regression problem in this scenario, and Cai, Daskalakis, and Papadimitriou (2015) and Shah and Zhou (2016) investigates the problem for more general machine learning problems. Mihailescu and Teo (2010) also discusses a similar problem for general machine learning algorithms.

- The last group of papers uses machine learning or robust statistics techniques without game-theoretic models. This group of papers include Dekel and Shamir (2009b), Dekel and Shamir (2009a).

2 Model

2.1 Maximum Likelihood Estimation

There are n strategic agents each providing the label of one data point to the principal. The principal is the learner and builds a machine learning model based on the data points provided by the agents, in particular, we assume the principal is training a softmax regression for multi-class classification. An agent, i, has publicly known feature vector, $x^{(i)}$, and a private discrete label, $y^{(i)}$. The objective of the agent is to maximize the probability that her data point is labeled correctly by the principal's model, and the agent can choose to report $\hat{y}^{(i)}$ to achieve the objective, with possibly $\hat{y}^{(i)} \neq y^{(i)}$. The objective of the principal is to minimize the loss from the data set with the correct labels.

2.2 Partial Loss Function Derivations

A dataset is incentive incompatible if,

$$\mathbb{P}\left\{Y = y^{(i)} | w^{\star}, x^{(i)}\right\} \leqslant \mathbb{P}\left\{Y = y^{(i)} | \hat{w}, x^{(i)}\right\},$$

where,

$$w^{\star} = \arg\max_{w} \log\left(\mathbb{P}\left\{Y = y^{(i)} | w, x^{(i)}\right\}\right) + C_{-i}(w),$$
$$\hat{w} = \arg\max_{w} \log\left(\mathbb{P}\left\{Y = \hat{y} | w, x^{(i)}\right\}\right) + C_{-i}(w).$$

The function $C(w)$ summarizes the loss due to other agents, assuming they are reporting labels truthfully,

$$C_{-i}(w) = \sum_{i' \neq i} \log\left(\mathbb{P}\left\{Y = y^{(i')} | w, x^{(i')}\right\}\right).$$

Suppose the function L is globally convex, differentiable, etc, then the problem translates to the first derivative condition,

$$\frac{\nabla_w \mathbb{P}\left\{Y = y^{(i)} | w^{\star}, x^{(i)}\right\}}{\mathbb{P}\left\{Y = y^{(i)} | w^{\star}, x^{(i)}\right\}} + \nabla_w\left(C_{-i}(w^{\star})\right) = 0,$$
$$\frac{\nabla_w \mathbb{P}\left\{Y = \hat{y} | \hat{w}, x^{(i)}\right\}}{\mathbb{P}\left\{Y = \hat{y} | \hat{w}, x^{(i)}\right\}} + \nabla_w\left(C_{-i}(\hat{w})\right) = 0.$$

For logistic regression with weights $w_c, c = 1, 2, ..., K$,

$$\mathbb{P}\{Y = c | w, x\} = \frac{e^{w_c^T x + b_c}}{\sum_{c'} e^{w_{c'}^T x + b_{c'}}},$$

$$\nabla_{w_c}\mathbb{P}\left\{Y=c|w,x\right\}=\frac{e^{w_c^Tx+b_c}\sum\limits_{c'\neq c}e^{w_c^Tx+b_c}}{\left(\sum\limits_{c'}e^{w_{c'}^Tx+b_{c'}}\right)^2}x.$$

$$\nabla_{w_c}\mathbb{P}\left\{Y=\hat{c}\neq c|w,x\right\}=\frac{e^{w_c^Tx+b_c}e^{w_{\hat{c}}^Tx+b_{\hat{c}}}}{\left(\sum\limits_{c'}e^{w_{c'}^Tx+b_{c'}}\right)^2}x.$$

The derivative conditions implies,

$$\left(1-\mathbb{P}\left\{Y=c|w^\star,x^{(i)}\right\}\right)x^{(i)}+\nabla_{w_c}\left(C_{-i}\left(w^\star\right)\right)=0,c=y^{(i)},$$
$$\left(\mathbb{P}\left\{Y=c|w^\star,x^{(i)}\right\}\right)x^{(i)}+\nabla_{w_c}\left(C_{-i}\left(w^\star\right)\right)=0,c\neq y^{(i)},$$

same for the expression with $\hat{w}$.
Substitute into the incentive incompatibility condition,

$$\nabla_{w_{y^{(i)}j}}\left(C_{-i}\left(w^\star\right)\right))x_j^{(i)}\leqslant\nabla_{w_{y^{(i)}j}}\left(C_{-i}\left(\hat{w}\right)\right)x_j^{(i)},j=1,2,...,m.$$

2.3 Old: Continuous Y Derivations

Envolope theorem version of the derivation.

Specifically, for the softmax regression, the objective of agent i with $x^{(i)}\in\mathbb{R}^m$ and $y^{(i)}\in\{1,2,...,k\}$ is,

$$\max_{\hat{y}\in\{1,2,...,k\}}\mathbb{P}\left\{Y=y^{(i)}|w,x^{(i)}\right\},$$

where,

$$\mathbb{P}\left\{Y=c|w,x^{(i)}\right\}=\frac{e^{z_c^{(i)}}}{\sum\limits_{c'=1}^{k}e^{z_{c'}^{(i)}}},$$
$$z_c^{(i)}=\sum_{j=1}^{m}w_{j,c}x_j^{(i)}+b_c,\text{ for }c\in\{1,2,...,k\}.$$

The learner maximizes the likelihood of the data.

$$\max_w\sum_{i=1}^{n}\log\left(\mathbb{P}\left\{Y=\hat{y}^{(i)}|w,x^{(i)}\right\}\right),$$

where w is the $m\times(k+1)$ weight matrix that includes the bias terms for the softmax regression.

This formulation does not permit $\hat{y}^{(i)}$ to be a continous variable, but if we rewrite the optimization problem

as the maximization of the cross entropy, then we could treat $\hat{y}^{(i)}$ as a continous multinomial distribution,

$$\min_{w} \sum_{i=1}^{n} \sum_{c=1}^{k} -\hat{y}_c^{(i)} \log \left(\mathbb{P} \left\{ Y = c | w, x^{(i)} \right\} \right)$$

Use envolope theorem and assume $w^\star \left(\hat{y}^{(i)} \right)$ is the optimal weights. The objective function is,

$$\mathcal{L} \left(w, \hat{y}^{(i)} \right) = \sum_{i=1}^{n} \sum_{c=1}^{k} -\hat{y}_c^{(i)} \log \left(\mathbb{P} \left\{ Y = c | w, x^{(i)} \right\} \right),$$

and the value function is,

$$\mathcal{L}^\star \left(\hat{y}^{(i)} \right) = \sum_{i=1}^{n} \sum_{c=1}^{k} -\hat{y}_c^{(i)} \log \left(\mathbb{P} \left\{ Y = c | w^\star, x^{(i)} \right\} \right),$$

apply envolope theorem,

$$\begin{aligned} \frac{\partial \mathcal{L}^\star \left(\hat{y}^{(i)} \right)}{\partial \hat{y}^{(i)}} &= -\log \left(\mathbb{P} \left\{ Y = \hat{y}^{(i)} | w^\star, x^{(i)} \right\} \right) \\ &> 0. \end{aligned}$$

Gradient descent version of the derivation.

One iteration of the gradient descent with learning rate η is given by,

$$w'_{j,c} = w_{j,c} - \eta x_j^{(i)} \left(\mathbb{P} \left\{ Y = c | x^{(i)} \right\} - \mathbb{1}_{\hat{y}=c} \right).$$

We start by treating $y^{(i)}$ as a continuous distribution over $\{1, 2, ..., k\}$ and investigate if reporting $\hat{y} \neq y^{(i)}$ increases the probability that the model classifies $x^{(i)}$ as $y^{(i)}$. Denote the probability $\mathbb{P} \left\{ Y^{(i)} = c \right\}$ by $y_c^{(i)}$ (slight abuse of notation). Then $\begin{bmatrix} y_1^{(i)} & y_2^{(i)} & ... & y_k^{(i)} \end{bmatrix}^T \in \Delta^{k-1}$ is the one-hot encoding of $y^{(i)}$, for example, for a generic data point (x, y), when $k = 3$,

$$\begin{aligned} y = 1 \text{ iff } \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} &= \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \\ y = 2 \text{ iff } \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} &= \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \\ y = 3 \text{ iff } \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \end{aligned}$$

Now fix instance i and define $o_c = \mathbb{P}\left\{Y = c|x^{(i)}\right\}$, then,

$$\begin{aligned}
\frac{\partial o_c}{\partial y_c} &= \frac{\partial o_c}{\partial z_c} \sum_{j=1}^{m} \frac{\partial z_c}{\partial w_{j,c}} \frac{\partial w_{j,c}}{\partial y_c} \\
&= o_c \left(1 - o_c\right) \sum_{j=1}^{m} x_j^{(i)} x_j^{(i)} \eta \\
&= \eta o_c \left(1 - o_c\right) \sum_{j=1}^{m} \left(x_j^{(i)}\right)^2 \\
&\geqslant 0.
\end{aligned}$$

Similarly,

$$\begin{aligned}
\frac{\partial o_c}{\partial y_{c'}} &= \frac{\partial o_c}{\partial z_{c'}} \sum_{j=1}^{m} \frac{\partial z_{c'}}{\partial w_{j,c'}} \frac{\partial w_{j,c'}}{\partial y_{c'}} \\
&= -o_c o_{c'} \sum_{j=1}^{m} x_j^{(i)} x_j^{(i)} \eta \\
&= -\eta o_c o_{c'} \sum_{j=1}^{m} \left(x_j^{(i)}\right)^2 \\
&\leqslant 0.
\end{aligned}$$

This implies (why?) decreasing y_c and thus increasing $y_{c'}$ for some c' will always increase o_c. Therefore, there should be no incentive to misreport y.

2.4 Naive Bayes Model

No example in which agents have incentive to misreport is found for the Gaussian Naive Bayes estimator.

The prediction is given by:

$$\arg\max_c \mathbb{P}\left\{Y = c\right\} \mathbb{P}\left\{x^{(i)}|Y = c\right\}$$

Consider agent i with $\left(x^{(i)}, y^{(i)}\right)$, if she reports truthfully,

$$\begin{aligned}
\mathbb{P}\left\{Y = y^{(i)}\right\} &= \frac{n_c}{n}, \\
\mathbb{P}\left\{x^{(i)}|Y = y^{(i)}\right\} &\geqslant \mathbb{P}\left\{x^{(i)}|Y = \hat{y}\right\},
\end{aligned}$$

and if she reports $\hat{y} \neq y^{(i)}$,

$$\begin{aligned}
\mathbb{P}\left\{Y = y^{(i)}\right\} &= \frac{n_c - 1}{n}, \\
\mathbb{P}\left\{x^{(i)}|Y = c\right\} &\leqslant \mathbb{P}\left\{x^{(i)}|Y = \hat{y}\right\},
\end{aligned}$$

and both terms are smaller.

Given the dataset, the parameters $w = (\mu, \Sigma, \pi)$ (mean, variance, prior) are given by,

$$\begin{aligned}
\mu_c &= \frac{\sum_{i'=1}^{n} x^{(i')} \mathbb{1}_{y^{(i')}=c}}{n_c}, \\
\Sigma_c &= \frac{\sum_{i'=1}^{n} \left(x^{(i')} - \mu_c\right)\left(x^{(i')} - \mu_c\right)^T \mathbb{1}_{y^{(i')}=c}}{n_c}, \\
\pi_c &= \frac{n_c}{n}.
\end{aligned}$$

Then, the classification probability is,

$$\mathbb{P}_c\left\{Y = c|x^{(i)}\right\} \propto \pi_c \frac{1}{|\Sigma_c|} \exp\left(-\frac{1}{2}\left(x^{(i)} - \mu_c\right)^T (\Sigma_c)^{-1}\left(x^{(i)} - \mu_c\right)\right).$$

The condition for incentive compatibility is,

$$\pi_c' \frac{1}{|\Sigma_c'|} \exp\left(-\frac{1}{2}\left(x^{(i)} - \mu_c'\right)^T \left(\Sigma_c'\right)^{-1}\left(x^{(i)} - \mu_c'\right)\right) < \pi_c \frac{1}{|\Sigma_c|} \exp\left(-\frac{1}{2}\left(x^{(i)} - \mu_c\right)^T (\Sigma_c)^{-1}\left(x^{(i)} - \mu_c\right)\right),$$

where,

$$\begin{aligned}
\mu_c' &= \frac{\sum_{i' \neq i} x^{(i')} \mathbb{1}_{y^{(i')}=c}}{n_c - 1}, \\
\Sigma_c' &= \frac{\sum_{i' \neq i} \left(x^{(i')} - \mu_c\right)\left(x^{(i')} - \mu_c\right)^T \mathbb{1}_{y^{(i')}=c}}{n_c - 1}, \\
\pi_c' &= \frac{n_c - 1}{n}.
\end{aligned}$$

3 Numerical Results

3.1 Structured Examples

A dataset is incentive incompatible with respect to the model parameterized by w if at least one agent has the incentive to report $\hat{y}^{(i)} \neq y^{(i)}$. Formally, let the model estimated with the true label be,

$$w^\star = \arg\max_w \sum_{i'=1^n} \log\left(\mathbb{P}\left\{Y = y^{(i')}|w, x^{(i')}\right\}\right) + \lambda \|w\|,$$

and the model estimated with the misreported label be,

$$\hat{w} = \arg\max_w \left(\sum_{i' \neq i} \log\left(\mathbb{P}\left\{Y = y^{(i')}|w, x^{(i')}\right\}\right)\right) + \log\left(\mathbb{P}\left\{Y = \hat{y}^{(i)}|w, x^{(i)}\right\}\right) + \lambda \|w\|.$$

The parameter λ is the regulerization parameter.
The dataset is incentive incompatible if there is i such that,

$$\mathbb{P}\left\{Y = y^{(i)} | \hat{w}, x^{(i)}\right\} > \mathbb{P}\left\{Y = y^{(i)} | w^{\star}, x^{(i)}\right\}.$$

The following example with a simple structure illustrates the incentive incompatibility problem with three classes. The numerical estimation is done using the "multinom" function in R with the "nnet" library. The parameters are estimated using maximum likelihood without regularization. The estimated parameters are replicated with BFGS using the "optim" function in R with a likelihood value within 0.01. Incentive incompatibility does not disappear when regularization with $\lambda = 0.001, 0.01$, and 0.1 are used. The parameter estimation are done using BFGS.

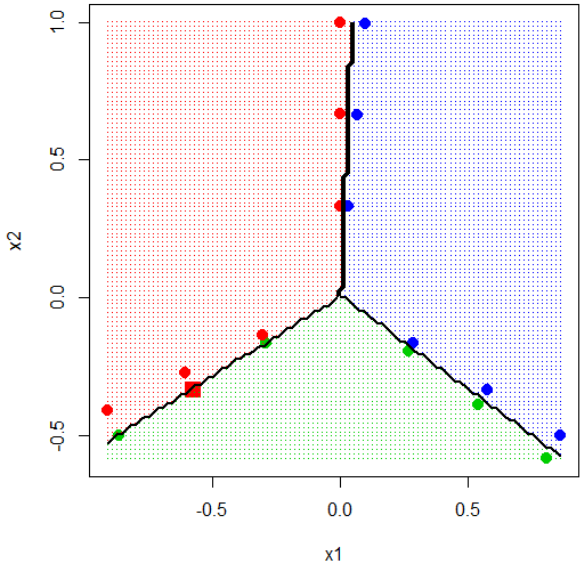

Figure 1: The 2D boundary example

-

The figure is a plot of the feature vectors of 18 points, with their true class labels shown by colors. The classification boundary of the original model $w^{\star}$ with the true labels are drawn. In particular, if the agent represented by the red square report her label as blue, the probability that the point is classified as red will increase from 0.3325 to 0.4625.
Intuitively, the agent with feature vector x and label y close to the classification boundary between two classes, say y and y', might have incentive to report the third class's label $\hat{y} \notin \{y, y'\}$ to shift the decision boundary and increase the probability that x is classified as y.

The data points are:

$$\left(y^{(1)}, x_1^{(1)}, x_2^{(1)}\right) = (1, 0.00000000, 0.3333333)$$

$$\left(y^{(2)}, x_1^{(2)}, x_2^{(2)}\right) = (1, 0.00000000, 0.6666667)$$
$$\left(y^{(3)}, x_1^{(3)}, x_2^{(3)}\right) = (1, 0.00000000, 1.0000000)$$
$$\left(y^{(4)}, x_1^{(4)}, x_2^{(4)}\right) = (3, 0.28867513, -0.1666667)$$
$$\left(y^{(5)}, x_1^{(5)}, x_2^{(5)}\right) = (3, 0.57735027, -0.3333333)$$
$$\left(y^{(6)}, x_1^{(6)}, x_2^{(6)}\right) = (3, 0.86602540, -0.5000000)$$
$$\left(y^{(7)}, x_1^{(7)}, x_2^{(7)}\right) = (2, -0.28867513, -0.1666667)$$
$$\left(y^{(8)}, x_1^{(8)}, x_2^{(8)}\right) = (1, -0.57735027, -0.3333333) \leftarrow \text{ This point}$$
$$\left(y^{(9)}, x_1^{(9)}, x_2^{(9)}\right) = (2, -0.86602540, -0.5000000)$$
$$\left(y^{(10)}, x_1^{(10)}, x_2^{(10)}\right) = (3, 0.03327781, 0.3316681)$$
$$\left(y^{(11)}, x_1^{(11)}, x_2^{(11)}\right) = (3, 0.06655561, 0.6633361)$$
$$\left(y^{(12)}, x_1^{(12)}, x_2^{(12)}\right) = (3, 0.09983342, 0.9950042)$$
$$\left(y^{(13)}, x_1^{(13)}, x_2^{(13)}\right) = (2, 0.27059406, -0.1946535)$$
$$\left(y^{(14)}, x_1^{(14)}, x_2^{(14)}\right) = (2, 0.54118812, -0.3893069)$$
$$\left(y^{(15)}, x_1^{(15)}, x_2^{(15)}\right) = (2, 0.81178218, -0.5839604)$$
$$\left(y^{(16)}, x_1^{(16)}, x_2^{(16)}\right) = (1, -0.30387186, -0.1370146)$$
$$\left(y^{(17)}, x_1^{(17)}, x_2^{(17)}\right) = (1, -0.60774373, -0.2740292)$$
$$\left(y^{(18)}, x_1^{(18)}, x_2^{(18)}\right) = (1, -0.91161559, -0.4110438)$$

A similar example with 33 points in which the points are away from the classification boundary. If the agent represented by the red square report her label as blue, the probability that the point is classified as red will increase from 0.8940 to 0.8970.

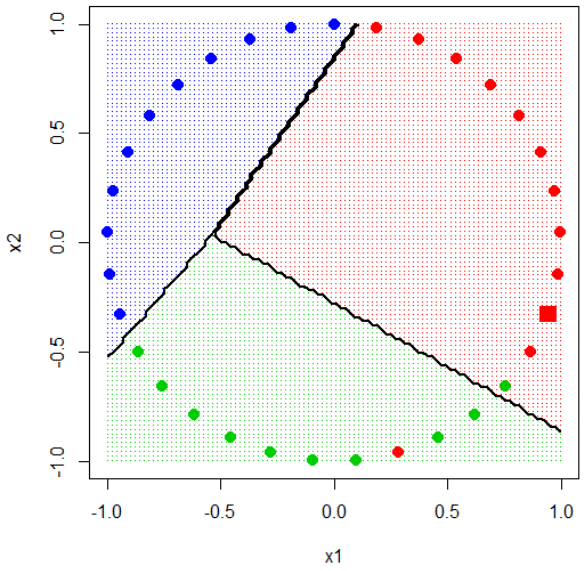

Figure 2: The 2D edge example

-

The data points are:

$$\left(y^{(1)}, x_1^{(1)}, x_2^{(1)}\right) = (1, 0.1892512, 0.9819287)$$
$$\left(y^{(2)}, x_1^{(2)}, x_2^{(2)}\right) = (1, 0.3716625, 0.92836793)$$
$$\left(y^{(3)}, x_1^{(3)}, x_2^{(3)}\right) = (1, 0.5406408, 0.84125353)$$
$$\left(y^{(4)}, x_1^{(4)}, x_2^{(4)}\right) = (1, 0.690079, 0.72373404)$$
$$\left(y^{(5)}, x_1^{(5)}, x_2^{(5)}\right) = (1, 0.814576, 0.58005691)$$
$$\left(y^{(6)}, x_1^{(6)}, x_2^{(6)}\right) = (1, 0.909632, 0.41541501)$$
$$\left(y^{(7)}, x_1^{(7)}, x_2^{(7)}\right) = (1, 0.9718116, 0.23575894)$$
$$\left(y^{(8)}, x_1^{(8)}, x_2^{(8)}\right) = (1, 0.9988673, 0.04758192)$$
$$\left(y^{(9)}, x_1^{(9)}, x_2^{(9)}\right) = (1, 0.9898214, -0.14231484)$$
$$\left(y^{(10)}, x_1^{(10)}, x_2^{(10)}\right) = (1, 0.9450008, -0.32706796) \leftarrow \text{ This point}$$
$$\left(y^{(11)}, x_1^{(11)}, x_2^{(11)}\right) = (1, 0.8660254, -0.5)$$
$$\left(y^{(12)}, x_1^{(12)}, x_2^{(12)}\right) = (2, 0.7557496, -0.65486073)$$
$$\left(y^{(13)}, x_1^{(13)}, x_2^{(13)}\right) = (2, 0.618159, -0.78605309)$$
$$\left(y^{(14)}, x_1^{(14)}, x_2^{(14)}\right) = (2, 0.4582265, -0.88883545)$$

$$\left(y^{(15)}, x_1^{(15)}, x_2^{(15)}\right) = (1, 0.2817326, -0.95949297)$$
$$\left(y^{(16)}, x_1^{(16)}, x_2^{(16)}\right) = (2, 0.09505604, -0.99547192)$$
$$\left(y^{(17)}, x_1^{(17)}, x_2^{(17)}\right) = (2, -0.09505604, -0.99547192)$$
$$\left(y^{(18)}, x_1^{(18)}, x_2^{(18)}\right) = (2, -0.2817326, -0.95949297)$$
$$\left(y^{(19)}, x_1^{(19)}, x_2^{(19)}\right) = (2, -0.4582265, -0.88883545)$$
$$\left(y^{(20)}, x_1^{(20)}, x_2^{(20)}\right) = (2, -0.618159, -0.78605309)$$
$$\left(y^{(21)}, x_1^{(21)}, x_2^{(21)}\right) = (2, -0.7557496, -0.65486073)$$
$$\left(y^{(22)}, x_1^{(22)}, x_2^{(22)}\right) = (2, -0.8660254, -0.5)$$
$$\left(y^{(23)}, x_1^{(23)}, x_2^{(23)}\right) = (3, -0.9450008, -0.32706796)$$
$$\left(y^{(24)}, x_1^{(24)}, x_2^{(24)}\right) = (3, -0.9898214, -0.14231484)$$
$$\left(y^{(25)}, x_1^{(25)}, x_2^{(25)}\right) = (3, -0.9988673, 0.04758192)$$
$$\left(y^{(26)}, x_1^{(26)}, x_2^{(26)}\right) = (3, -0.9718116, 0.23575894)$$
$$\left(y^{(27)}, x_1^{(27)}, x_2^{(27)}\right) = (3, -0.909632, 0.41541501)$$
$$\left(y^{(28)}, x_1^{(28)}, x_2^{(28)}\right) = (3, -0.814576, 0.58005691)$$
$$\left(y^{(29)}, x_1^{(29)}, x_2^{(29)}\right) = (3, -0.690079, 0.72373404)$$
$$\left(y^{(30)}, x_1^{(30)}, x_2^{(30)}\right) = (3, -0.5406408, 0.84125353)$$
$$\left(y^{(31)}, x_1^{(31)}, x_2^{(31)}\right) = (3, -0.3716625, 0.92836793)$$
$$\left(y^{(32)}, x_1^{(32)}, x_2^{(32)}\right) = (3, -0.1892512, 0.9819287)$$
$$\left(y^{(33)}, x_1^{(33)}, x_2^{(33)}\right) = (3, 0, 1)$$

Another geometrically more symmetric example with 18 points in which the points are away from the classification boundary. If the agent represented by the red square report her label as blue, the probability that the point is classified as red will increase from 0.3239 to 0.4966, which changes from an incorrect classification to the correct classification if the label with the largest prediction probability is chosen.

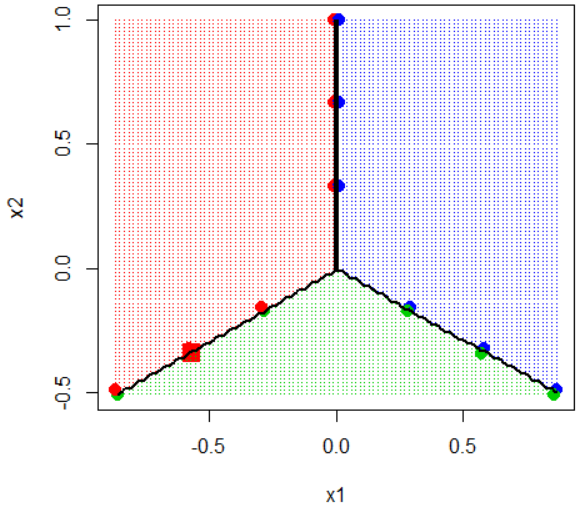

Figure 3: The 2D edge example

-

The data points are:

$$
\begin{aligned}
\left(y^{(1)}, x_1^{(1)}, x_2^{(1)}\right) &= (1, -0.0100000, 0.3333333) \\
\left(y^{(2)}, x_1^{(2)}, x_2^{(2)}\right) &= (1, -0.0100000, 0.6666667) \\
\left(y^{(3)}, x_1^{(3)}, x_2^{(3)}\right) &= (1, -0.0100000, 1.0000000) \\
\left(y^{(4)}, x_1^{(4)}, x_2^{(4)}\right) &= (2, -0.2836751, -0.1753269) \\
\left(y^{(5)}, x_1^{(5)}, x_2^{(5)}\right) &= (1, -0.5723503, -0.3419936) \leftarrow \text{ This point} \\
\left(y^{(6)}, x_1^{(6)}, x_2^{(6)}\right) &= (2, -0.8610254, -0.5086603) \\
\left(y^{(7)}, x_1^{(7)}, x_2^{(7)}\right) &= (3, 0.2936751, -0.1580064) \\
\left(y^{(8)}, x_1^{(8)}, x_2^{(8)}\right) &= (3, 0.5823503, -0.3246731) \\
\left(y^{(9)}, x_1^{(9)}, x_2^{(9)}\right) &= (3, 0.8710254, -0.4913397) \\
\left(y^{(10)}, x_1^{(10)}, x_2^{(10)}\right) &= (3, 0.0100000, 0.3333333) \\
\left(y^{(11)}, x_1^{(11)}, x_2^{(11)}\right) &= (3, 0.0100000, 0.6666667) \\
\left(y^{(12)}, x_1^{(12)}, x_2^{(12)}\right) &= (3, 0.0100000, 1.0000000) \\
\left(y^{(13)}, x_1^{(13)}, x_2^{(13)}\right) &= (1, -0.2936751, -0.1580064) \\
\left(y^{(14)}, x_1^{(14)}, x_2^{(14)}\right) &= (1, -0.5823503, -0.3246731) \\
\left(y^{(15)}, x_1^{(15)}, x_2^{(15)}\right) &= (1, -0.8710254, -0.4913397)
\end{aligned}
$$

$$\left(y^{(16)}, x_1^{(16)}, x_2^{(16)}\right) = (2, 0.2836751, -0.1753269)$$
$$\left(y^{(17)}, x_1^{(17)}, x_2^{(17)}\right) = (2, 0.5723503, -0.3419936)$$
$$\left(y^{(18)}, x_1^{(18)}, x_2^{(18)}\right) = (2, 0.8610254, -0.5086603)$$

3.2 Randomly Generated Examples

This section is no longer necessary.

Three types of examples are randomly generated and tested for incentive incompatibility, for $m = 1, 2$ and $k = 2, 3, 4$

1. Random labels: $x_i \in \mathbb{R}^m, x_i \sim N(0, I), y_i \sim$ Unif $[0, k-1]$.
2. Linearly separable labels: $x_i \in \mathbb{R}^m, x_i \sim N(0, I), y_i = \arg\max_{j \in \{0,1,\ldots,k-1\}} w_j \begin{bmatrix} x_i \\ 1 \end{bmatrix}, w_j \in \mathbb{R}^k, w_j \sim N(0, I)$.
3. Linearly separable with variance σ : $x_i \in \mathbb{R}^m, x_i \sim N(0, I), y_i = \arg\max_{j \in \{0,1,\ldots,k-1\}} w_j \begin{bmatrix} x_i \\ 1 \end{bmatrix} + \varepsilon_j, w_j \in \mathbb{R}^k, w_j \sim N(0, I), \varepsilon \in \mathbb{R}^k, \varepsilon \sim N(0, I)$.

Currently, all the following examples use the second and third methods of generating points.

3.3 One Dimensional Case

In the case x is 1-dimensional, one randomly generated dataset with 20 points illustrates the problem with softmax regression being not incentive compatible.

$$(x_1, y_1) = (0.8048, 2)$$
$$(x_2, y_2) = (0.5694, 2)$$
$$(x_3, y_3) = (1.016, 2)$$
$$(x_4, y_4) = (1.2838, 2)$$
$$(x_5, y_5) = (0.0747, 2)$$
$$(x_6, y_6) = (-0.4731, 2)$$
$$(x_7, y_7) = (-0.5567, 2)$$
$$(x_8, y_8) = (0.8745, 2)$$
$$(x_9, y_9) = (-0.4735, 2)$$
$$(x_{10}, y_{10}) = (-1.1037, 3)$$
$$(x_{11}, y_{11}) = (1.3496, 1)$$
$$(x_{12}, y_{12}) = (-0.4792, 3)$$
$$(x_{13}, y_{13}) = (-0.0358, 3)$$

$$(x_{14}, y_{14}) = (1.2837, 1)$$
$$(x_{15}, y_{15}) = (-0.0038, 2)$$
$$(x_{16}, y_{16}) = (1.4523, 1)$$
$$(x_{17}, y_{17}) = (-2.3012, 3)$$
$$(x_{18}, y_{18}) = (-0.3552, 2)$$
$$(x_{19}, y_{19}) = (0.6926, 2)$$
$$(x_{20}, y_{20}) = (0.488, 2)$$

If agent 4 reports $(1.2838, 3)$, resulting in model Y', instead of $(1.2838, 2)$ truthfully, resulting in model Y,

$$\mathbb{P}\left\{Y' = 2\right\} = 0.4254 > 0.4198 = \mathbb{P}\left\{Y = 2\right\}.$$

In the following plot, class $1, 2, 3$ has colors red, green, blue respectively. If the square green point pretends to be blue, the probability of it getting classified as green will increase.

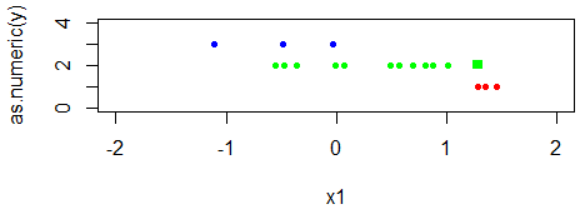

Figure 4: 1D Example 1

-

A similar but linearly separable case is,

$$(x_1, y_1) = (0.6699, 3)$$
$$(x_2, y_2) = (1.3181, 3)$$
$$(x_3, y_3) = (-0.724, 2)$$
$$(x_4, y_4) = (0.3886, 1)$$
$$(x_5, y_5) = (-1.0845, 2)$$
$$(x_6, y_6) = (0.4201, 3)$$
$$(x_7, y_7) = (-1.5717, 2)$$
$$(x_8, y_8) = (1.8893, 3)$$
$$(x_9, y_9) = (-1.3195, 2)$$
$$(x_{10}, y_{10}) = (1.2162, 3)$$
$$(x_{11}, y_{11}) = (-0.3737, 1)$$
$$(x_{12}, y_{12}) = (-0.406, 1)$$

$$
\begin{aligned}
(x_{13}, y_{13}) &= (1.0343, 3) \\
(x_{14}, y_{14}) &= (-0.0174, 1) \\
(x_{15}, y_{15}) &= (1.4013, 3) \\
(x_{16}, y_{16}) &= (0.6972, 3) \\
(x_{17}, y_{17}) &= (1.2113, 3) \\
(x_{18}, y_{18}) &= (-1.0789, 2) \\
(x_{19}, y_{19}) &= (0.5583, 3) \\
(x_{20}, y_{20}) &= (-0.1254, 1)
\end{aligned}
$$

If agent 4 reports $(0.3886, 1)$, resulting in model Y', instead of $(0.3886, 2)$ truthfully, resulting in model Y,

$$
\mathbb{P}\left\{Y' = 2\right\} = 0.7736 > 0.6353 = \mathbb{P}\left\{Y = 2\right\}.
$$

In the following plot, class $1, 2, 3$ has colors red, green, blue respectively. If the square red point pretends to be green, the probability of it getting classified as red will increase.

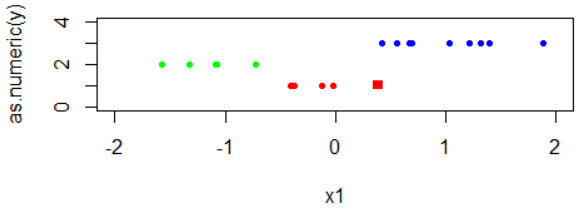

Figure 5: 1D Example 2

-

3.4 Two Dimensional Case

One example in which the data is not linearly separable:

$$
\begin{aligned}
(x_{1,1}, x_{2,1}, y) &= (-0.2078, 1.7184, 1) \\
(x_{1,2}, x_{2,2}, y) &= (-0.648, -0.1972, 1) \\
(x_{1,3}, x_{2,3}, y) &= (-1.551, 0.2337, 1) \\
(x_{1,4}, x_{2,4}, y) &= (0.5005, 0.0292, 2) \\
(x_{1,5}, x_{2,5}, y) &= (0.6556, 0.2169, 2) \\
(x_{1,6}, x_{2,6}, y) &= (-0.7498, -0.0237, 1) \\
(x_{1,7}, x_{2,7}, y) &= (0.3018, 0.2766, 2) \\
(x_{1,8}, x_{2,8}, y) &= (-0.6075, 1.8255, 1)
\end{aligned}
$$

$$
\begin{aligned}
(x_{1,9}, x_{2,9}, y) &= (0.093, 0.0902, 3) \\
(x_{1,10}, x_{2,10}, y) &= (-0.341, 0.3715, 2) \\
(x_{1,11}, x_{2,11}, y) &= (-0.1007, 0.4598, 2) \\
(x_{1,12}, x_{2,12}, y) &= (0.6475, 1.8673, 2) \\
(x_{1,13}, x_{2,13}, y) &= (0.0962, 1.5949, 2) \\
(x_{1,14}, x_{2,14}, y) &= (-1.1301, 0.9306, 1) \\
(x_{1,15}, x_{2,15}, y) &= (-1.6144, 0.5291, 1) \\
(x_{1,16}, x_{2,16}, y) &= (1.3819, 0.5892, 2) \\
(x_{1,17}, x_{2,17}, y) &= (0.3037, -1.255, 3) \\
(x_{1,18}, x_{2,18}, y) &= (-0.275, -1.1986, 3) \\
(x_{1,19}, x_{2,19}, y) &= (0.9622, -0.103, 3) \\
(x_{1,20}, x_{2,20}, y) &= (-0.2279, -0.5819, 3)
\end{aligned}
$$

If agent 4 reports $(0.5005, 0.0292, 1)$, resulting in model Y', instead of $(0.5005, 0.0292, 2)$ truthfully, resulting in model Y,

$$\mathbb{P}\left\{Y' = 2\right\} = 0.4681 > 0.4548 = \mathbb{P}\left\{Y = 2\right\}.$$

In the following plot, class $1, 2, 3$ has colors red, green, blue respectively. If the square red point pretends to be a green point, the probability of it getting classified as red will increase.

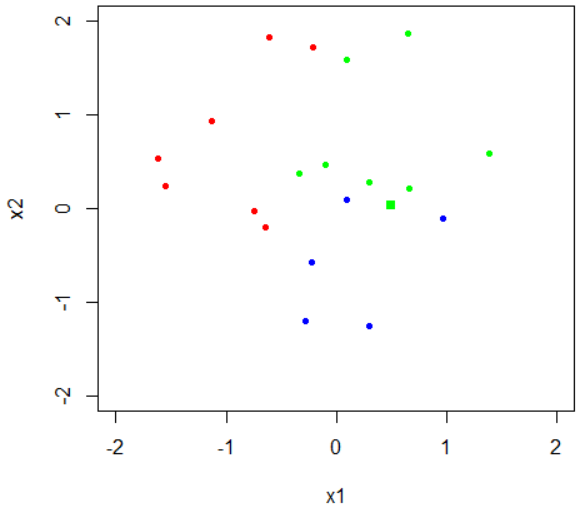

Figure 6: 2D Example 1

-

One example in which the data is linearly separable:

$$
\begin{aligned}
(x_{1,1}, x_{2,1}, y) &= (0.0684, 1.2528, 1) \\
(x_{1,2}, x_{2,2}, y) &= (-0.8979, 1.7309, 1) \\
(x_{1,3}, x_{2,3}, y) &= (-0.0656, -1.6169, 2) \\
(x_{1,4}, x_{2,4}, y) &= (1.1107, -0.6537, 3) \\
(x_{1,5}, x_{2,5}, y) &= (0.2553, 1.4098, 1) \\
(x_{1,6}, x_{2,6}, y) &= (1.3888, 0.8675, 2) \\
(x_{1,7}, x_{2,7}, y) &= (0.4308, -0.8872, 2) \\
(x_{1,8}, x_{2,8}, y) &= (1.5898, 1.468, 2) \\
(x_{1,9}, x_{2,9}, y) &= (0.6773, 2.2383, 1) \\
(x_{1,10}, x_{2,10}, y) &= (-0.5839, -0.071, 1) \\
(x_{1,11}, x_{2,11}, y) &= (0.2665, 0.4818, 2) \\
(x_{1,12}, x_{2,12}, y) &= (-0.8833, -0.6374, 1) \\
(x_{1,13}, x_{2,13}, y) &= (0.0716, -1.082, 2) \\
(x_{1,14}, x_{2,14}, y) &= (-0.512, -0.1667, 1) \\
(x_{1,15}, x_{2,15}, y) &= (1.353, -0.0793, 3) \\
(x_{1,16}, x_{2,16}, y) &= (-0.6957, -0.1058, 1) \\
(x_{1,17}, x_{2,17}, y) &= (0.2837, -0.8186, 2) \\
(x_{1,18}, x_{2,18}, y) &= (0.8643, -1.1354, 2) \\
(x_{1,19}, x_{2,19}, y) &= (1.2658, 0.9953, 2) \\
(x_{1,20}, x_{2,20}, y) &= (-1.6236, 0.0485, 1)
\end{aligned}
$$

If agent 18 reports $(0.8643, -1.1354, 1)$, resulting in model Y', instead of $(0.8643, -1.1354, 2)$ truthfully, resulting in model Y,

$$\mathbb{P}\left\{Y' = 2\right\} = 0.9213 > 0.9098 = \mathbb{P}\left\{Y = 2\right\}.$$

In the following plot, class $1, 2, 3$ has colors red, green, blue respectively. If the square red point pretends to be a green point, the probability of it getting classified as red will increase.

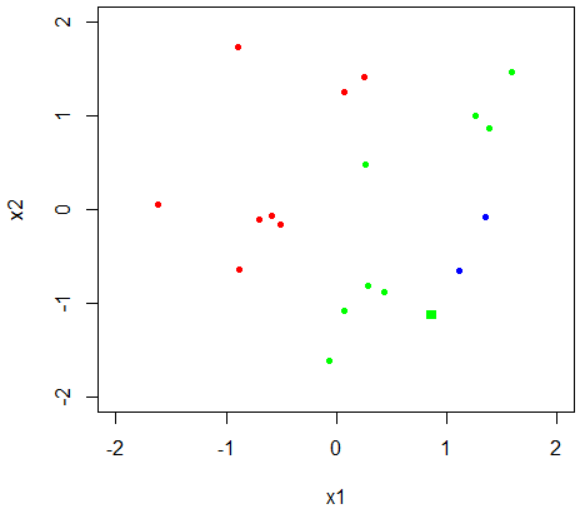

Figure 7: 2D Example 2

-

References

Cai, Yang, Constantinos Daskalakis, and Christos Papadimitriou (2015), "Optimum statistical estimation with strategic data sources." In *Conference on Learning Theory*, 280–296.

Chen, Yiling, Yang Liu, and Chara Podimata (2019), "Grinding the space: Learning to classify against strategic agents." *arXiv preprint arXiv:1911.04004*.

Chen, Yiling, Chara Podimata, Ariel D Procaccia, and Nisarg Shah (2018), "Strategyproof linear regression in high dimensions." In *Proceedings of the 2018 ACM Conference on Economics and Computation*, 9–26.

Dekel, Ofer, Felix Fischer, and Ariel D Procaccia (2010), "Incentive compatible regression learning." *Journal of Computer and System Sciences*, 76, 759–777.

Dekel, Ofer and Ohad Shamir (2009a), "Good learners for evil teachers." In *Proceedings of the 26th annual international conference on machine learning*, 233–240.

Dekel, Ofer and Ohad Shamir (2009b), "Vox populi: Collecting high-quality labels from a crowd." In *COLT*.

Dütting, Paul, Zhe Feng, Harikrishna Narasimhan, David C Parkes, and Sai Srivatsa Ravindranath (2017), "Optimal auctions through deep learning." *arXiv preprint arXiv:1706.03459*.

Epasto, Alessandro, Mohammad Mahdian, Vahab Mirrokni, and Song Zuo (2018), "Incentive-aware learning for large markets." In *Proceedings of the 2018 World Wide Web Conference*, 1369–1378.

Gast, Nicolas, Stratis Ioannidis, Patrick Loiseau, and Benjamin Roussillon (2013), "Linear regression from strategic data sources." *arXiv preprint arXiv:1309.7824*.

Golowich, Noah, Harikrishna Narasimhan, and David C Parkes (2018), "Deep learning for multi-facility location mechanism design." In *IJCAI*, 261–267.

Meir, Reshef, Ariel D Procaccia, and Jeffrey S Rosenschein (2012), "Algorithms for strategyproof classification." *Artificial Intelligence*, 186, 123–156.

Mihailescu, Marian and Yong Meng Teo (2010), "Strategy-proof dynamic resource pricing of multiple resource types on federated clouds." In *International Conference on Algorithms and Architectures for Parallel Processing*, 337–350, Springer.

Perote, Javier and Juan Perote-Pena (2004), "Strategy-proof estimators for simple regression." *Mathematical Social Sciences*, 47, 153–176.

Procaccia, Ariel D and Moshe Tennenholtz (2009), "Approximate mechanism design without money." In *Proceedings of the 10th ACM conference on Electronic commerce*, 177–186.

Richardson, Adam, Ljubomir Rokvic, Aris Filos-Ratsikas, and Boi Faltings (2019), "Privately computing influence in regression models."

Shah, Nihar B and Dengyong Zhou (2016), "Double or nothing: Multiplicative incentive mechanisms for crowdsourcing." *The Journal of Machine Learning Research*, 17, 5725–5776.