On the Power of Oblivious Poisoning Attacks

Samuel Deng [1] **Sanjam Garg** [2] **Somesh Jha** [3] **Saeed Mahloujifar** [4]
Mohammad Mahmoody [4] **Abhradeep Thakurta** [5]

Abstract

Poisoning attacks have emerged as a significant security threat to machine learning algorithms. It has been demonstrated that adversaries who make small changes to the training set, such as adding specially crafted data points, can hurt the performance of the output model. Most of these attacks require the full knowledge of training data or the underlying data distribution. In this paper we study the power of *oblivious* adversaries who do not have any information about the training set. We first formalize the definitions related to various aspects of such attacks. We then demonstrate that such attacks, in general, are *provably* weaker by showing a separation between full-information and oblivious poisoning attacks in various settings. Specifically, we show the existence of natural learning problems that are rather robust against oblivious adversaries whose goal is to add a non-relevant features to the model with certain poisoning budget. On the other hand, we prove that for same problem setting, non-oblivious (full information) adversaries with the same budget can craft poisoning examples based on the rest of the training data and successfully hurt the performance. Finally, we design and run experiments and observe that they validate our theoretical findings.

1. Introduction

Traditional approach to supervised machine learning focuses on a benign setting; honestly sampled training data, perhaps with random noise, is given to a learner that outputs a model that will later get tested on the same data distribution used during the learning phase. Due to the broad deployment of learning algorithms in safety-critical applications, however, recently machine learning has gone through a revolution of studying the same problem under so-called poisoning attacks where there is an adversary who can interfere with data sampling by changing a honestly sampled dataset $\mathcal{S}$ into a close dataset $\mathcal{S}'$.

The central focus of this work is on understanding the power of poisoning adversaries, when it comes to adversary's *information* about the data set.

What does the adversary know about the data? Many previous work on theoretical analysis of poisoning attacks implicitly, or explicitly, assume that the adversary has full knowledge of the training data $\mathcal{S}$ before choosing what examples to add or delete from $\mathcal{S}$. For example, the adversary in universal targeted data poisoning attacks of (Mahloujifar et al., 2019a; Mahloujifar & Mahmoody, 2019) which is based on the (computational) concentration of measure in product spaces, needs to start off by knowing the full data set $\mathcal{S}$ and then select, one by one, whether or not to change each particular example in $\mathcal{S}$. Similarly, the attacks described in work of (Koh et al., 2018), construct poisoning datasets based on the knowledge of the "clean" training data. This assumption about the "full information" about $\mathcal{S}$ given to the poisoning adversary, however, is not realistic in all scenarios, as adversary might not have access to all of the data before deciding on what part of it to tamper with.

As a particular practical example where a poisoning adversary might naturally have limited information about (most of) training data, consider a federated (or any form of distributed) learning system (McMahan & Ramage, 2017; McMahan et al., 2016; Bonawitz et al., 2017; Konečný et al., 2016) between multiple hospitals who share their data with a trusted server with the goal of training a shared model over their aggregate data. Now one can imagine an adversary who wants to participate in this system and inject malicious data with the hope of degrading the quality of the trained model. In such scenario, the adversary might only know the examples that it would submit itself, and not the examples submitted by other hospitals, given that they only share their data with the trusted server. In this case, we are dealing with an oblivious poisoning adversary.

Main question: ***how much stronger are full-information attackers?*** Motivated by understanding the role of the knowledge about the data set by a poisoning adversary, in

[1] Columbia University [2] University of California, Berkeley [3] University of Wisconsin, Madison [4] University of Virginia [5] Google Research-Brain. Correspondence to: Saeed Mahloujifar <saeed@virginia.edu>.

Presented at the ICML 2020 Workshop on Uncertainty and Robustness in Deep Learning. Copyright 2020 by the author(s).

this work, we directly study whether having full information can help a poisoning attacker. Namely, we study whether there is a learning task in which oblivious poisoning adversaries who might only know the trained model θ (but not the entire training data $\mathcal{S}$ that has led to θ) are *provably* weaker than full-information adversaries who know the entire data set $\mathcal{S}$ (in addition to perhaps knowing the model θ).

A new motivation for data privacy. Privacy is often viewed as a utility for data owners in the machine learning pipeline. Due to the trade-offs between privacy and the utility of the users, data users sometimes ignore the privacy of data owners while doing their analysis, specially when they do not have any incentive to enforce the privacy. A positive answer to the main question posed above could create a new motivation for keeping training dataset private. Specifically, the users of data would try to keep training dataset private, with the goal of securing their models against poisoning and increasing their utility in scenarios where part of data is coming from potentially malicious sources.

1.1. Our Contribution

In this work, we initiate a formal study of the role of adversary's knowledge in poisoning attacks by comparing the two attack models: traditional full information attacks vs. oblivious attacks. In particular, we study the provable difference that it makes when the adversary knows *all* of the training data before launching the poisoning attack, called the full-information threat model, in contrast to when the adversary adds malicious data to the training set in an oblivious way.

Formalizing oblivious poisoning attacks. We start by formalizing what it means mathematically to be an oblivious poisoning adversary. We present a comprehensive treatment of the subject by separately studying the issues of *how* the poisoning attack is done vs. *what goals* the attacker pursues.

After formalizing the concept of oblivious poisoning attacks, we do a comparative study of the power of oblivious attack vs. their full-information counterparts in both contexts of feature selection as well as risk minimization.

Separations for feature selection tasks. We first prove our separation between full-information and oblivious poisoning adversarial models in the context of *feature selection*, and specifically for a sparse linear regression problem. In a feature selection problem, the learning algorithm wants to discover the relevant features that determine the ground truth function. For example, imagine a dataset of patients with many features, who suffer from an specific disease with different levels of severity. One can try to find the most important features contributing to the severity of disease in the context of feature selection. Specifically, the learners' goal here is to recover a vector $\theta^* \in \mathbb{R}^p$ whose non-zero coordinates determine the relevant features contributing to the disease. In this scenario, the goal of the adversary is to deceit the learning process and make it output a model $\hat{\theta}' \in \mathbb{R}^p$ with a different set of non-zero coordinates.

Separations for risk minimization tasks. Most poisoning attacks in the literature deal with increasing the population (or sometimes "targeted") risk of a produced model $\hat{\theta}$. In this work, we additionally prove some preliminary separation results in this context, proving further evidence that oblivious attacks are provably less powerful compared to their full-information counterparts for the task of learning half spaces. See Section 3 for the formal statements and the supplemental material for proofs.

Experiments. To empirically investigate the power of oblivious and full-information attacks, we experiment on synthetic and real world datasets. Our experiments confirm our theoretical findings as they also show that the power of oblivious and poisoning attacks differs significantly.

Related work. Due to space limitations, more related works could be found in Appendix A.

Open question. In this work, we separate the power of two extreme cases of adversaries. Oblivious adversaries that do not know anything about the training set and adversaries who know all the training set. An interesting open question is to study the power of adversaries with partial knowledge over the training set.

2. Oblivious vs. Full-information Poisoning: Formally Defining Threat Models

In this section, we formally define the security games of various learning systems under *oblivious* poisoning attacks. It is common in cryptography to define security model based on a game between an adversary and a challenger (Katz & Lindell, 2007). Here, we use the same approach and introduce game based definitions for oblivious and full-information adversaries. We use some (of the standard) notations that are reviewed in Appendix B.

In what follows, we give full security games for a feature selection task. Later, in Section 3 we will see how to construct problem instances (by defining their data distributions) that provably separate the power of oblivious attacks from full-information counterparts.

Definition 2.1 (Oblivious and full-information data injection poisoning for feature selection)**.** *We first describe the* data oblivious *security game between a challenger C and an adversary A, and the game is parameterized by adversary's budget k and the training data $\mathcal{S} = [X \,|\, Y]$ which is a matrix X and a set of labels Y, and the feature selection algorithm* $\mathsf{FtrSelector}$. *The implicit assumption here is that $[X \,|\, Y]$ defines a true set of features* $\mathrm{Supp}(\theta)$ *that can be recovered using* $\mathsf{FtrSelector}$.

$\mathbf{OblFtrSel}(k, \mathcal{S} = [X \,|\, Y], \mathsf{FtrSelector})$.

1. *Adversary A generates a poisoning dataset $[X' \,|\, Y'] \in [-1,1]^{k \times (p+1)}$ of size k.*
2. *A sends $[X' \,|\, Y']$ to C.*
3. *C recovers model $\hat{\theta} = \mathsf{FtrSelector}([X \,|\, Y])$.*
4. *C also recovers $\hat{\theta}' = \mathsf{FtrSelector}\left(\left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]\right)$.*
5. *Adversary wins if $\mathrm{Supp}(\hat{\theta}) \neq \mathrm{Supp}(\hat{\theta}')$.*

In the full information *security game, all the steps are the same as above, except that in the beginning (as "zeroth" step) the following is done.*

$\mathbf{FullFtrSel}(k, \mathcal{S} = [X \,|\, Y], \mathsf{FtrSelector})$.

- *Step 0: C sends $[X \,|\, Y]$ to A.*
- *The rest of the steps are the same as those of the game $\mathbf{OblFtrSel}(k, [X \,|\, Y], \mathsf{FtrSelector})$.*

We now give a formal security game to define the security of learning tasks under (oblivious) poisoning, where adversary's goal is to increase the (population) risk of the produced model. More formally, the goal of the adversary A is simply to modify $\mathcal{S}$ into some $\mathcal{S}'$ that leads to producing θ' (instead of θ) with as large a population risk $\mathsf{Risk}(\theta')$ as possible. Therefore, the game below will not have a zero-one output, but rather it will have a real number as how much the adversary wins.

Definition 2.2 (Oblivious and full-information data injection poisoning for population risk)**.** *We first describe the* data oblivious *security game between a challenger C and an adversary A, and then will describe how to modify it into a full-information variant. Such game is parameterized by adversary's budget k, a data set $\mathcal{S}$ a learning algorithm L, and a distribution D over $\mathcal{X} \times \mathcal{Y}$ (where $\mathcal{X}$ is the space of inputs and $\mathcal{Y}$ is the space of outputs).*[1]

$\mathbf{OblRisk}(k, \mathcal{S}, L, D)$.

1. *Adversary A generates k new examples $(e'_1, \ldots, e'_k)$ and them to C.*
2. *C obtains $\mathcal{S}'$ by adding the injected examples to $\mathcal{S}$.*
3. *C runs L over $\mathcal{S}'$ to obtain (poisoned) $\theta' \leftarrow L(\mathcal{S}')$.*
4. *A's advantage (in winning the game) will be $\mathsf{Risk}(\theta', D) = \Pr_{(x,y) \leftarrow D}[\theta'(x) \neq y]$.*[2]

In the full information *security game, all the steps are the same as above, except the first step:*

$\mathbf{FullRisk}(k, \mathcal{S}, L, D)$.

- *Step 0: C sends $\mathcal{S}$ to A.*
- *The rest of the steps are the same as those of the game $\mathbf{OblRisk}(k, \mathcal{S}, L, D)$.*

Explained above for Definition 2.1, one can also envision variations of Definition 2.2 in which the goal of the attacker is to increase the error on a particular instance (i.e., a *targeted* poisoning (Barreno et al., 2006; Shen et al., 2016)) or use other poisoning methods that eliminate or substitute poison data rather than just adding some.

3. Separating the Power of Oblivious and Full-information Attacks

In this section, we will provably demonstrate that the power of oblivious and full-information adversaries could significantly differ. In particular, we study the power of poisoning attacks in two contexts of feature selection and classification. We focus on two important algorithms, Lasso Estimator for Feature selection and Empirical Risk minimization for classification.

Separation for feature selection. We first prove the existence of a feature selection problem such that, with high probability, it stays secure in the oblivious attack model of Definition 2.1, while the same problem's setting is highly vulnerable to poisoning adversaries as defined in the full-information threat model of Definition 2.1. We use Lasso estimator for proving our separation result.

Recalling feature selection through Lasso estimator. We work in the feature selection setting, and the exact format of our problem is as follows. There is a target parameter vector $\theta^* \in (0,1)^p$. We have a $n \times p$ matrix X (n vectors, each of p features) and we have $Y = X \times \theta^* + W$ where W itself is a small noise, and Y is the vector of noisy observations about θ^*. Number of non-zero elements (denoting the actual relevant features) in θ^* are bounded by s namely, $\mathrm{Supp}(\theta^*) \leq s$. For the setting of the problem mentioned above, the Lasso Estimator tries to learn θ^* by optimizing the a penalized loss and obtain the solution $\hat{\theta}$ as $\hat{\theta} = \mathrm{argmin}_{\theta \in (0,1)^p} \frac{1}{n} \cdot \|Y - X \times \theta\|_2^2 + \frac{2\lambda}{n} \cdot \|\theta\|_1$. We use $\mathsf{Lasso}([X \,|\, Y])$ to denote $\hat{\theta}$, as learned by the Lasso optimization described above. We also use $\mathrm{Risk}(\hat{\theta}, [X \,|\, Y])$ to denote the "scaled up" value of the Lasso's objective function $\mathrm{Risk}(\hat{\theta}, [X \,|\, Y]) = \left\|Y - X \times \hat{\theta}\right\|_2^2 + 2 \cdot \lambda \cdot \left\|\hat{\theta}\right\|_1$.

Theorem 3.1. *For any $k, p \in \mathbb{N}$, there exist $n \in \mathbb{N}$ and a dataset $[X \,|\, Y] \in \mathbb{R}^{n \times (p+1)}$ such that if we randomly shuffle the coordinates of X according to a permutation π to get $\pi(X)$ and uses the dataset $[\pi(X) \,|\, Y]$ to run the Lasso estimator. Then, the probability of winning for an oblivious adversary in the security game of Definition 2.1 is at most $2/p$, namely, for all adversaries A we have*

$$\mathop{\mathbb{E}}_{\pi \leftarrow G}\left[\mathsf{Adv}\big(A, \mathbf{OblFtrSel}(k, [\pi(X) \,|\, Y], \mathsf{Lasso})\big)\right] \leq \frac{2}{p}.$$

[1] Since we deal with risk, we need to add D as a new parameter compared to the games of Definition 2.1.

[2] Note that this is a real number. More generally we can use any loss function, which covers the case of regression as well.

while a full-information adversary can win the security game of Definition 2.1 with probability 1. *Namely, there exist adversary A such that*

$$\mathop{\mathbb{E}}_{\pi\leftarrow G}\left[\mathsf{Adv}\big(A, \mathbf{FullFtrSel}(k, [\pi(X) \,\big|\, Y], \mathsf{Lasso})\big)\right] = 1.$$

Separation for classification Now we also show a separation on the power of oblivious and full-information poisoning attacks on classification. In particular we show that empirical risk minimization (ERM) algorithm could be much more susceptible to full-information poisoning adversaries, compared to oblivious adversaries.

Theorem 3.2. *There is a distribution of distributions $\mathfrak{D}$ such that there is a data injecting adversary with budget $\varepsilon \cdot n$ that wins the full-information security game for classification by advantage ε, namely there exists A such that*

$$\mathop{\mathbb{E}}_{\substack{D\leftarrow\mathfrak{D}\\ \mathcal{S}\leftarrow D^n}}\left[\mathsf{Adv}\big(A, \mathbf{FullRisk}(\varepsilon\cdot n, \mathcal{S}, \mathsf{ERM}, D)\big)\right] \geq \Omega(\varepsilon).$$

On the other hand, any adversary A has much smaller advantage in the oblivious game. Namely,

$$\mathop{\mathbb{E}}_{\substack{D\leftarrow\mathfrak{D}\\ \mathcal{S}\leftarrow D^n}}\left[\mathsf{Adv}\big(A, \mathbf{OblRisk}(\varepsilon\cdot n, \mathcal{S}, \mathsf{ERM}, D)\big)\right] \leq O(\varepsilon^2).$$

4. Experiments

In this section [3], we present experiment to show the power of oblivious and full-information poisoning attacks. Below are are some details about our experiments. For more details we refer readers to Appendix 1.

Feature Selection. In Appendix E.1 we show an oblivious attack with provable success for LASSO. Here we use that attack to explore different features, to see how many poison points we need to add each feature to the final model.

- *Synthetic dataset.* We empirically validate the claims about Construction E.6 by instantiating it with a synthetic dataset of $n = 1000$ rows and $p = 100000$ features.
- *Real-world datasets.* We use four datasets frequently used in feature selection to explore the separation in real world data: Boston, TOX, Protate_GE, and SMK.

For all the datasets, we observe that there is a separation between the average value of r across all the features and the minimum value of r, exhibiting the existence of a few unstable features. For instance, in the Boston dataset, the `1 (ZN)` feature requires less than 40 poison rows to add it to the dataset, while the average is around 120, and the `9 (TAX)` feature requires about 170. Because an oblivious adversary would not have information on what these unstable features are, while a non-oblivious adversary would, these experiments show an empirical gap between the power of a non-oblivious adversary and an oblivious one.

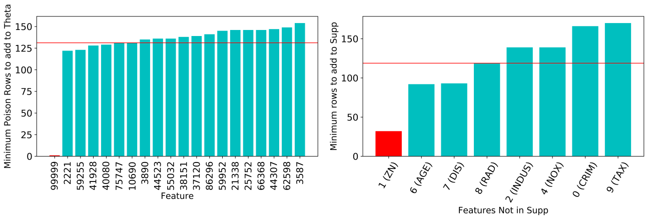

Figure 1. Synthetic experiment and Boston experiment. We attacked 20 randomly sampled features of our synthetic dataset and all the features of the Boston dataset, represented by cyan bars. The y-axis is r, the number of features needed to add the feature to $\hat{\theta}$. The red horizontal line is the average of all r. Both datasets have a significantly unstable feature, shown as the leftmost red bar.

Classification. We design an experiment to empirically validate the claim made in Theorem 3.2, that there is a separation between oblivious and full-information poisoning adversaries for classification. We setup the experiment just as in the proof of Theorem 3.2, as follows.

First, we sample training points $X = x_1, x_2, \ldots x_m$ for $m = 1,000$ from the Gaussian space $\mathcal{N}(0,1)^2$, and pick a random ground-truth halfspace w^* from $\mathcal{N}(0,1)^2$. Using w^*, we find our labels $y_1, y_2, \ldots y_m$ by taking $(w^*)^T x_k$ for $k \in [m]$. This ensures the data is linearly separable by the homogeneous halfspace produced by w^*. Since we are working in 2D settings, it is possible to implement the ERM for this dataset. To attack this, we implement one full-information poisoning attack and three different attack strategies for oblivious adversarys. We observe in Figure 2 that the full-information adversary can increase the error linearly with ϵ using this strategy, while the oblivious adversaries fail to have any consistent impact on the resulting classifier's error with their strategies.

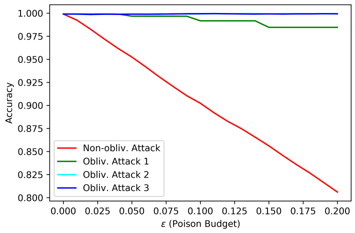

Figure 2. Oblivious and full-information poisoning separation in classification. We plot the effect of each adversary's attack on the accuracy of our resulting poisoned ERM halfspace. See Appendix H for the discription of the attacks.

[3] The code for our experiments can be fond at https://github.com/essdeee/oblivious_poisoning.

References

Barreno, M., Nelson, B., Sears, R., Joseph, A. D., and Tygar, J. D. Can machine learning be secure? In *Proceedings of the 2006 ACM Symposium on Information, computer and communications security*, pp. 16–25. ACM, 2006.

Bhagoji, A. N., Chakraborty, S., Mittal, P., and Calo, S. Analyzing federated learning through an adversarial lens. In *International Conference on Machine Learning*, pp. 634–643, 2019.

Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., Ramage, D., Segal, A., and Seth, K. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1175–1191. ACM, 2017.

Bubeck, S., Lee, Y. T., Price, E., and Razenshteyn, I. Adversarial examples from cryptographic pseudo-random generators. *arXiv preprint arXiv:1811.06418*, 2018a.

Bubeck, S., Price, E., and Razenshteyn, I. Adversarial examples from computational constraints. *arXiv preprint arXiv:1805.10204*, 2018b.

Charikar, M., Steinhardt, J., and Valiant, G. Learning from untrusted data. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 47–60. ACM, 2017.

Degwekar, A. and Vaikuntanathan, V. Computational limitations in robust classification and win-win results. *arXiv preprint arXiv:1902.01086*, 2019.

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Moitra, A., and Stewart, A. Robust estimators in high dimensions without the computational intractability. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pp. 655–664. IEEE, 2016.

Diakonikolas, I., Kane, D. M., and Stewart, A. Statistical query lower bounds for robust estimation of high-dimensional Gaussians and Gaussian mixtures. In *Foundations of Computer Science (FOCS), 2017 IEEE 58th Annual Symposium on*, pp. 73–84. IEEE, 2017.

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Steinhardt, J., and Stewart, A. Sever: A robust meta-algorithm for stochastic optimization. *arXiv preprint arXiv:1803.02815*, 2018a.

Diakonikolas, I., Kane, D. M., and Stewart, A. List-decodable robust mean estimation and learning mixtures of spherical Gaussians. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1047–1060. ACM, 2018b.

Diakonikolas, I., Kong, W., and Stewart, A. Efficient algorithms and lower bounds for robust linear regression. *arXiv preprint arXiv:1806.00040*, 2018c.

Diochnos, D. I., Mahloujifar, S., and Mahmoody, M. Lower bounds for adversarially robust pac learning. *arXiv preprint arXiv:1906.05815*, 2019.

Garg, S., Jha, S., Mahloujifar, S., and Mahmoody, M. Adversarially robust learning could leverage computational hardness. *arXiv preprint arXiv:1905.11564*, 2019.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., and et al. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999a. doi: 10.1126/science.286.5439.531.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., and et al. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999b. doi: 10.1126/science.286.5439.531.

Harrison, D. and Rubinfeld, D. Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5:81–102, 03 1978. doi: 10.1016/0095-0696(78)90006-2.

Katz, J. and Lindell, Y. *Introduction to Modern Cryptography*. Chapman & Hall/CRC, 2007.

Koh, P. W., Steinhardt, J., and Liang, P. Stronger data poisoning attacks break data sanitization defenses. *arXiv preprint arXiv:1811.00741*, 2018.

Konečnỳ, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., and Bacon, D. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

Lai, K. A., Rao, A. B., and Vempala, S. Agnostic estimation of mean and covariance. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pp. 665–674. IEEE, 2016.

Mahloujifar, S. and Mahmoody, M. Blockwise p-Tampering Attacks on Cryptographic Primitives, Extractors, and Learners. In *Theory of Cryptography Conference*, pp. 245–279. Springer, 2017.

Mahloujifar, S. and Mahmoody, M. Can adversarially robust learning leverage computational hardness? *arXiv preprint arXiv:1810.01407*, 2018.

Mahloujifar, S. and Mahmoody, M. Can adversarially robust learning leverage computational hardness? In *Algorithmic Learning Theory*, pp. 581–609, 2019.

Mahloujifar, S., Diochnos, D. I., and Mahmoody, M. The curse of concentration in robust learning: Evasion and poisoning attacks from concentration of measure. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 4536–4543, 2019a.

Mahloujifar, S., Mahmoody, M., and Mohammed, A. Universal multi-party poisoning attacks. In *International Conference on Machine Learning*, pp. 4274–4283, 2019b.

McMahan, B. and Ramage, D. Federated learning: Collaborative machine learning without centralized training data. *Google Research Blog*, 2017.

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., et al. Communication-efficient learning of deep networks from decentralized data. *arXiv preprint arXiv:1602.05629*, 2016.

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pp. 506–519. ACM, 2017.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Prasad, A., Suggala, A. S., Balakrishnan, S., and Ravikumar, P. Robust estimation via robust gradient estimation. *arXiv preprint arXiv:1802.06485*, 2018.

Shafahi, A., Huang, W. R., Najibi, M., Suciu, O., Studer, C., Dumitras, T., and Goldstein, T. Poison frogs! targeted clean-label poisoning attacks on neural networks. In *Advances in Neural Information Processing Systems*, pp. 6103–6113, 2018.

Shen, S., Tople, S., and Saxena, P. A uror: defending against poisoning attacks in collaborative deep learning systems. In *Proceedings of the 32nd Annual Conference on Computer Security Applications*, pp. 508–519. ACM, 2016.

Suciu, O., Marginean, R., Kaya, Y., Daume III, H., and Dumitras, T. When does machine learning {FAIL}? generalized transferability for evasion and poisoning attacks. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pp. 1299–1316, 2018.

Thakurta, A. G. and Smith, A. Differentially private feature selection via stability arguments, and the robustness of the lasso. In *Conference on Learning Theory*, pp. 819–850, 2013.

Turner, A., Tsipras, D., and Madry, A. Label-consistent backdoor attacks. *arXiv preprint arXiv:1912.02771*, 2019.

Wainwright, M. J. Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1-constrained quadratic programming (lasso). *IEEE transactions on information theory*, 55(5):2183–2202, 2009.

Wang, Y. and Chaudhuri, K. Data poisoning attacks against online learning. *arXiv preprint arXiv:1808.08994*, 2018.

Xiao, H., Biggio, B., Brown, G., Fumera, G., Eckert, C., and Roli, F. Is feature selection secure against training data poisoning? In *ICML*, pp. 1689–1698, 2015.

Zhu, C., Huang, W. R., Shafahi, A., Li, H., Taylor, G., Studer, C., and Goldstein, T. Transferable clean-label poisoning attacks on deep neural nets. *arXiv preprint arXiv:1905.05897*, 2019.

Supplementary Material

A. Related Work

As opposed to data poisoning setting, the question of adversary's knowledge is previously studied in the line of work on adversarial examples. In a test time evasion attack the adversary's goal is to find an adversarial example, the adversary knows the input x fully before trying to find a close input x' (such that $x' \sim x$) that is misclassified. In that setting, the question of adversary's knowledge can be formed around whether or not it knows the model θ completely or it only has a black-box access to it (Papernot et al., 2017). Note that, in our work, the model θ is known to the adversary, and the information complexity of the attack focuses on whether or not the adversary is aware of the full training data.

Some previous work have studied poisoning attacks in the setting of federated/distributed learning (Bhagoji et al., 2019; Mahloujifar et al., 2019b). Their attacks, however, either (implicitly) assume a full (or partial) information attacker, or aim to increase the population risk (as opposed to injecting features in a feature selection task). Thus, our work is novel in both formally studying the differences between full-information vs. oblivious attacks, and *provably* separating the power of these two attack models in the contexts of feature selection as well as risk-minimization tasks. We note that Xiao et al. (2015) also empirically examine the robustness of feature selection in the context of poisoning attacks, but their measure of stability is across sets of features. We are distinct in a sense that our paper studies the effect of oblivious attacks on *individual* features and with provable guarantees.

We also distinguish our work with another line of work that studies the computational complexity of the attacker (Mahloujifar & Mahmoody, 2018; Garg et al., 2019). Here, we study the "information complexity" of the attack; namely, what information the attacker needs to succeed in a poisoning attack, while those works study the *computational resources* that a poisoning attacker needs to successfully degrade the quality of the learned model. Another recent exciting line of work that studies the computational aspect of robust learning in poisoning contexts, focuses on the computational complexity of the *learning* process itself (Diakonikolas et al., 2016; Lai et al., 2016; Charikar et al., 2017; Diakonikolas et al., 2017; 2018b;a; Prasad et al., 2018; Diakonikolas et al., 2018c), and other works have studied the same question about the complexity of the learning process for evasion attacks (Bubeck et al., 2018b;a; Degwekar & Vaikuntanathan, 2019). Furthermore, since we deal with information complexity, our work is distinct from previous work that studies the impact of the training set (e.g., using clean labels) on the success of poisoning (Shafahi et al., 2018; Zhu et al., 2019; Suciu et al., 2018; Turner et al., 2019).

Finally, we remark that *online* poisoning adversaries studied in (Mahloujifar & Mahmoody, 2017; Wang & Chaudhuri, 2018; Mahloujifar & Mahmoody, 2019), roughly speaking, is a form of attack that lies somewhere between oblivious and full-information attacks. In their model, an online adversary needs to choose its decision about the i^{th} example (i.e., to tamper or not tamper it) based only on the history of the first $i-1$ examples, and without the knowledge of the future examples. So, their knowledge about the training data is limited, in a partial way. Since we separate the power of full information vs. oblivious attacks, a corollary of our results is that at least one of these models is different from the online variant for recovering sparse linear regression. In other words, we are in one of the following worlds: (i) online adversaries are provably stronger than oblivious adversaries or (ii) full-information adversaries are provably stronger than online adversaries.

B. Notation.

We first define some useful notation. For an arbitrary vector $\theta \in \mathbb{R}^p$ we use $\mathrm{Supp}(\theta) = \{i \colon \theta_i \neq 0\}$, we denote the set of (indices of) its non-zero coordinates of $\theta \in \mathbb{R}^p$. We also use $\|\theta\|_2$ and $\|\theta\|$ to denote the ℓ_2 and ℓ_1 norms of θ respectively. For two matrices $X \in R^{n\times p}$ and $Y \in R^{n\times 1}$, we use $\left[X \,\middle|\, Y\right] \in R^{n\times(p+1)}$ to denote a set of n regression observations on feature vectors $X_{i\in[n]}$ such that Y_i is the real-valued observation for X_i. For two matrices $X_1 \in \mathbb{R}^{n_1\times p}$ and $X_2 \in \mathbb{R}^{n_2\times p}$, we use $\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \in \mathbb{R}^{(n_1+n_2)\times p}$ to denote the concatenation of X_1 and X_2. Similarly, for two set of observations $\left[X_1 \,\middle|\, Y_1\right] \in \mathbb{R}^{n_1\times(p+1)}$ and $\left[X_2 \,\middle|\, Y_2\right] \in \mathbb{R}^{n_2\times(p+1)}$, we use $\left[\begin{array}{c|c} X_1 & Y_1 \\ X_2 & Y_2 \end{array}\right] \in \mathbb{R}^{(n_1+n_2)\times(p+1)}$ to denote the concatenation of $\left[X_1 \,\middle|\, Y_1\right]$ and $\left[X_2 \,\middle|\, Y_2\right]$. For a security game G and an adversary A we use $\mathsf{Adv}(A, G)$ to refer to the advantage (the amount of win) of adversary A in G.

C. Further Details on Defining Oblivious Attacks

In this section, we discuss other definitional aspecstw of oblivious and full-information poisoning attacks.

C.1. Oblivious Variants of (Full-Information) Data Poisoning Attacks

In this section, we explain how to formalize oblivious poisoning attackers in general, and in the next subsection we will describe how to instantiate this general approach for the case of feature selection.

A poisoning adversary of "budget" k, can tamper with a training sequence $\mathcal{S} = \{e_1, \ldots, e_n\}$, by "modifying" $\mathcal{S}$ by at most k changes. Such changes can be in three forms

- **Injection.** Adversary can inject k new examples $e'_1, \ldots, e'_k$ to $\mathcal{S}$. This is without loss of generality when the learner is symmetric and is not sensitive to the order in the training examples. More generally, when the training set is treated like a sequence $\mathcal{S} = (e_1, \ldots, e_n)$, the adversary can even choose the *location* of these planted examples $e'_1, \ldots, e'_k$. More formally, the adversary picks k numbers $1 \leq i_1 < \cdots < i_k \leq n + k$, and constructs the new data sequence $\mathcal{S}' = (e''_1, \ldots, e''_{n+k})$ by letting $e''_j = e'_{i_j}$ and letting $\mathcal{S}$ fill the remaining coordinates of $\mathcal{S}'$ in their original order from $\mathcal{S}$.
  Oblivious injection. In the full-information setting, the adversary can choose the poison examples and their locations based on $\mathcal{S}$. In the oblivious variant, the adversary chooses the poison examples $e'_1, \ldots, e'_k$ and their locations $1 \leq i_1 < \cdots < i_k \leq n + k$ without knowing the original set $\mathcal{S}$.
- **Elimination.** Adversary can eliminate k of the examples in $\mathcal{S}$. When $\mathcal{S}$ is a sequence, the adversary only needs to state the indexes $1 \leq i_1 < \ldots, i_k \leq n$ of the removed examples.
  Oblivious elimination. In the full-information setting, the adversary can choose the locations of the deleted examples based on $\mathcal{S}$. In the oblivious variant, the adversary chooses the locations without knowing the original set $\mathcal{S}$.
- **Substitution and it oblivious variant.** These two settings are similar to data elimination, with the difference that the adversary, in addition to the sequence of locations, chooses k poison examples $e'_1, \ldots, e'_k$ to substitute e_{i_j} by e'_j for all $j \in [k]$.

More general attack strategies. One can think of more fine-grained variants of the substitution attacks above by having different ”budgets” for injection and elimination processes (and even allowing different locations for eliminations and injections), but we keep the setting simple by default.

C.2. Taxonomy for Attacks on Feature Selection

Sometimes the goal of a learning process is to recover a model $\hat{\theta}$, perhaps from noisy data, that has the same set of features $\mathrm{Supp}(\hat{\theta})$ as the true model θ. For example, those features could be the relevant factors determining a decease. Such process is called feature selection (or model recovery). A poisoning attacker attacking a feature selection task would directly try to counter this goal. Now, regardless of *how* an attacker is transforming a data set $\mathcal{S}$ into $\mathcal{S}'$, let $\hat{\theta}'$ be the model that is learned from $\mathcal{S}'$. Below we give a taxonomy of various attack scenarios.

- **Feature adding.** In this case, the adversary's goal is to achieve $\mathrm{Supp}(\hat{\theta}') \not\subseteq \mathrm{Supp}(\theta)$. Namely, adding a feature that is not present in the true model θ.
- **Feature removal.** In this case, the adversary's goal is to achieve $\mathrm{Supp}(\theta) \not\subseteq \mathrm{Supp}(\hat{\theta}')$. Namely, removing a feature that is present in the true model θ.
- **Feature flipping.** In this case, the adversary's goal is to do either of the above. Namely, $\mathrm{Supp}(\theta) \neq \mathrm{Supp}(\hat{\theta}')$, which means that at least one of the features' existence is flipped.

Targeted variants of the attacks above. For each of the three attack goals above (in the context of feature selection), one can envision a *targeted* variant in which the adversary aims to add/remove or flip a specific feature $i \in [p]$ where p is the data dimension.

D. Borrowed Results

D.1. Sufficient Conditions for Model Recovery Using Lasso

In this section, we specify the sufficient conditions for a dataset that makes it a good dataset for robust recover using Lasso estimator. We borrow these specifications from the work of (Thakurta & Smith, 2013).

Definition D.1 (Typical systems)**.** *Suppose $\theta^* \in [0,1]^p$ be a model such that $|\operatorname{Supp}(\theta^*)| = s$. Let $X \in \mathbb{R}^{n\times p}$ and $Y \in \mathbb{R}^{n\times 1}$ and $W = Y - X \times \theta^*$. Also let $X_I \in \mathbb{R}^{n\times s}$ be a matrix formed by columns of X whose indices are in $\operatorname{Supp}(\theta^*)$ and $X_O \in \mathbb{R}^{n\times(p-s)}$ be a matrix formed by columns of X whose indices are not in $\operatorname{Supp}(\theta^*)$. The pair $(\theta^*, [X \,|\, Y])$ is called an (n, p, s, ψ, σ)-*typical *system, if the following hold:*

- ***Column normalization:*** *Each column of X has ℓ_2 norm bounded by $\sqrt{n}$.*
- ***Incoherence:*** $\left\|((X_O^T X_I)(X_I^T X_I)^{-1}\mathsf{sign}(\theta^*))\right\|_\infty \le 1/4.$
- ***Restricted strong restricted:*** *The minimum eigenvalue of $X_I X_I^T$ is at least ψ.*
- ***Bounded noise*** $\left\|X_O^T(I_{n\times n} - X_I(X_I^T X_I)^{-1}X_I^T)W\right\|_\infty \le 2\sigma\sqrt{n\log(p)}.$

The following theorem is a modified version of result of (Wainwright, 2009) borrowed from (Thakurta & Smith, 2013).

Theorem D.2 (Model recovery with Lasso (Wainwright, 2009))**.** *Let $(\theta^*, [X \,|\, Y])$ be a (n, p, s, σ, ψ)-typical system. Let $\alpha = \operatorname{argmin}_{i\in p}\max(\theta^*_i, 1-\theta^*_i)$. If $n \ge 16\cdot\frac{\sigma}{\psi\cdot\alpha}\sqrt{s\cdot\log(p)}$ and then $\hat\theta = \mathsf{Lasso}([X \,|\, Y])$ would have the same support as θ^* when $\lambda = 4\sigma\sqrt{n\cdot\log(p)}$.*

The following theorem is about robust model recovery with Lasso in (Thakurta & Smith, 2013).

Theorem D.3 (Robust model recovery with Lasso (Thakurta & Smith, 2013))**.** *Let $(\theta^*, [X \,|\, Y])$ be a (n, p, s, σ, ψ)-typical system. Let $\alpha = \operatorname{argmin}_{i\in p}\max(\theta^*_i, 1-\theta^*_i)$. If*

$$n \ge \max\left(\frac{16\sigma}{\psi\cdot\alpha}\sqrt{s\cdot\log(p)}, \frac{4s^4k^2(1/\psi+1)^2}{\log(p)\sigma^2}\right)$$

then $\hat\theta = \mathsf{Lasso}([X \,|\, Y])$ would have the same support as θ^* when $\lambda = 4\sigma\sqrt{n\cdot\log(p)}$.

In addition, adding any *set of k labeled vectors $[X' \,|\, Y']$ with ℓ_∞ norm at most 1 to $[X \,|\, Y]$ would not change the support set of the model recovered by Lasso estimator. Namely,*

$$\begin{aligned}\operatorname{Supp}\left(\mathsf{Lasso}\left(\left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]\right)\right) &= \operatorname{Supp}(\mathsf{Lasso}([X \,|\, Y])) \\ &= \operatorname{Supp}(\theta^*).\end{aligned}$$

Two theorems above are sufficient conditions for (robust) model recovery using lasso estimator. Bellow, we show two simple instantiating of the theorems on Normal distribution. Theorem bellow from Wainwright (2009) shows that the Lasso estimator with proper parameters provably finds the correct set of features, if the dataset and noise vectors are sampled from normal distributions.

Theorem D.4 (Wainwright (2009))**.** *Let X be a dataset sampled from $\mathcal{N}(0, 1/4)^{n\times p}$ and W be a noise vector sampled from $\mathcal{N}(0, \sigma^2)^n$. For any $\theta^* \in (0,1)^p$ with at most s number of non-zero coordinates, for $\lambda = 4\sigma\sqrt{n \times \log(p)}$ and $n = \omega(s\cdot\log(p))$, with probability at least $3/4$ over the choice of X and W (that determine Y as well) we have $\operatorname{Supp}(\hat\theta) = \operatorname{Supp}(\theta^*)$ where $\hat\theta = \mathsf{Lasso}([X \,|\, Y])$. Moreover, $\hat\theta$ is a unique minimizer for $\operatorname{Risk}(\cdot, [X \,|\, Y])$.*

The above theorem requires the dataset to be sampled from a certain distribution and does not take into account the possibilities of outliers in the data. The robust version of this theorem, where part of the training data is chosen by an adversary, can be instantiatet using Theorem D.2 as follows:

Theorem D.5 (Thakurta & Smith (2013))**.** *Let X be a dataset sampled from $\mathcal{N}(0, 1/4)^{n\times p}$ and W be a noise vector sampled from $\mathcal{N}(0, \sigma^2)^n$. For any $\theta^* \in (0,1)^p$, if $\lambda = 4\sigma\sqrt{n \times \log(p)}$ and $n = \omega(s\log(p) + s^4\cdot k^2)$, with probability at least $3/4$ over the choice of X, W (determining Y), and $Y = X \times \theta^* + W$ it holds that, adding* any *set of k labeled vectors $[X' \,|\, Y']$, such that rows of X' has ℓ_∞ norm at most 1 and Y has ℓ_∞ norm at most s, to $[X \,|\, Y]$ would not change the support set of the model recovered by Lasso estimator. Namely,*

$$\begin{aligned}\operatorname{Supp}\left(\mathsf{Lasso}\left(\left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]\right)\right) &= \operatorname{Supp}(\mathsf{Lasso}([X \,|\, Y])) \\ &= \operatorname{Supp}(\theta^*).\end{aligned}$$

Note that Theorems D.4 and D.5 are instantiating of the generalized theorems D.2 for normal distribution and D.3 and are proved by showing that the sufficient conditions of those theorems will happen with high probability over the choice of dataset.

E. Proof of Theorem 3.1

In this section, we prove Theorem 3.1. We first restate the Theorem for convenience.

Theorem 3.1 (Restated). *For any $k, p \in \mathbb{N}$, there exist some $n \in \mathbb{N}$ and a dataset $[X \,|\, Y] \in \mathbb{R}^{n\times(p+1)}$ such that if we randomly shuffle the coordinates of X according to a permutation π to get $\pi(X)$ and uses the dataset $[\pi(X) \,|\, Y]$ to run the Lasso estimator. Then, the probability of winning for an oblivious adversary in the security game of Definition 2.1 is at most $2/p$, namely,*

$$\forall A : \underset{\pi\leftarrow G}{\mathbb{E}} \left[\text{Advantage of } A \text{ in } \mathbf{OblFtrSel}(k, [\pi(X) \,|\, Y], \mathsf{Lasso})\right] \leq \frac{2}{p}.$$

while a full-information adversary can win the security game of Definition 2.1 with probability 1.

$$\exists A : \underset{\pi\leftarrow G}{\mathbb{E}} \left[\text{Advantage of } A \text{ in } \mathbf{FullFtrSel}(k, [\pi(X) \,|\, Y], \mathsf{Lasso})\right] = 1.$$

we first show two properties of a dataset $[X \,|\, Y]$ that if hold, we can prove separation. Then we will show how to instantiate a dataset with those two properties by changing a dataset that is sampled from a Gaussian distribution. The first notion divides the columns of data to stable and unstable features based on the number of poisoning points required to remove or add those features from or to the support set of the resulting model.

Now we state and prove the following theorem that separates the notion of oblivious and full-information poisoning attacks. This theorem assumes the existence of a (k, ϵ)-resilient dataset that is k-stable on all but one feature.

Definition E.1 (Stable and unstable coordinates). *Consider a dataset $[X \,|\, Y] \in \mathbb{R}^{n\times(p+1)}$ with a unique solution for the Lasso minimization. $[X \,|\, Y]$ is k-unstable on coordinate $i \in [p]$ if its i^{th} coordinate of model learn on it is 0, namely $\mathsf{Lasso}\left([X \,|\, Y]\right)_i = 0$ and there exist a data set $[X' \,|\, Y']$ with size k and ℓ_∞ norm at most 1 on each row such that $i \in \mathrm{Supp}\left(\mathsf{Lasso}\left(\begin{bmatrix} X & Y \\ X' & Y' \end{bmatrix}\right)\right)$. On the other hand, $[X \,|\, Y]$ is k-stable on a coordinate i, if for all datasets $[X' \,|\, Y']$ with k rows and ℓ_∞ norm at most 1 on each row we either have*

$$\mathsf{Lasso}\left([X \,|\, Y]\right)_i = \mathsf{Lasso}\left(\begin{bmatrix} X & Y \\ X' & Y' \end{bmatrix}\right)_i = 0 \quad \textit{or} \quad \mathsf{Lasso}([X \,|\, Y])_i \cdot \mathsf{Lasso}\left(\begin{bmatrix} X & Y \\ X' & Y' \end{bmatrix}\right)_i > 0.$$

Now we show how to use a dataset with only one k-unstable feature and prove the separation. The core idea is to shuffle the columns and prevent the adversary from finding the unstable coordinate. The adversary who does not know which of the coordinates is unstable cannot perform the attack but an adversary with the knowledge of the unstable coordinate can add poisoning points and cause the unstable coordinate to be added to the support set. The following definition captures the property of a dataset that adding the unstable feature is hard unless the adversary knows which feature is unstable.

Definition E.2 ((k, ϵ)-resilience). *Consider a dataset $[X \,|\, Y] \in \mathbb{R}^{n\times(p+1)}$ with a unique solution for the Lasso minimization and let $T = \mathrm{Supp}(\mathsf{Lasso}([X \,|\, Y]))$. Also, let G be the set of all permutations that are fixed on T namely, $G = \{\pi : [p] \to [p] \mid \forall i \in T; \pi(i) = i\}$. We say $[X \,|\, Y]$ is (k, ϵ)*-resilient *if for any dataset $[X' \,|\, Y']$ with k rows with ℓ_∞ norm at most 1, we have*

$$\Pr_{\pi\leftarrow G}\left[\mathrm{Supp}\left(\mathsf{Lasso}\left(\begin{bmatrix} X & Y \\ \pi(X') & Y' \end{bmatrix}\right)\right) \neq T\right] \leq \epsilon,$$

where $\pi(X')$ is the matrix produced by permuting the columns of X' according to π.

Theorem E.3 (Separating oblivious and full-information adversaries). *Consider a dataset $[X \,|\, Y]$ that is (k, ϵ)-resilient and k-stable on all the coordinates except on 1 coordinate that is k-unstable. Suppose the challenger takes this dataset and randomly shuffles the coordinates according to a permutation π to get $\pi(X)$ and uses the dataset $[\pi(X) \,|\, Y]$ to run the Lasso estimator. Then, the probability of winning for an oblivious adversary in the security game of Definition 2.1 is at most ϵ, while a full-information adversary can win the security game of Definition 2.1 with probability* 1.

Proof of Theorem E.3. We first show that winning the full-information game of Definition 2.1 is always possible. After getting the dataset $[X \,|\, Y]$ the adversary inspects the dataset to find out which coordinate is unstable and find a poisoning dataset that would add that unstable coordinate to the support set of the model.

Now we show that no adversary can win the oblivious security game of Definition 2.1 with probability more than ϵ. The intuition behind this claim is the symmetric nature of the Lasso estimator—by permuting the rows of a dataset $[X \,|\, Y]$ to $[\pi(X) \,|\, Y]$ the Lasso would output the same output with permuted coordinates. Namely, $\mathsf{Lasso}([\pi(X) \,|\, Y]) = \pi(\mathsf{Lasso}([X \,|\, Y]))$.

Now, let π be the permutation chosen by the challenger and let $\hat{\theta} = \mathsf{Lasso}([X \,|\, Y])$ and let $T = \mathrm{Supp}(\hat{\theta})$. Adversary receives $\pi(\theta)$ and generates a poisoning dataset $[X' \,|\, Y']$. Let $A(\pi(\hat{\theta}))$ denote the potentially randomized algorithm that the adversary uses to generate the poison data, and let $G = \{\pi : [p] \to [p] \mid \forall i \in T; \pi(i) = i\}$. Now we use (k, ϵ)-resiliency of $[X \,|\, Y]$ to argue about the probability of an oblivious adversary winning the game. The high level idea is that because the oblivious adversary cannot discriminate between zero coordinates, he cannot find the right ordering of coordinates with a high probability.

$$\begin{aligned}
&\Pr_{\substack{\pi \leftarrow S_p \\ [X'|Y'] \leftarrow A(\pi(\hat{\theta}))}} \left[\mathsf{Lasso}\left(\begin{bmatrix} \pi(X) & Y \\ X' & Y' \end{bmatrix}\right) \neq T\right] \\
&= \Pr_{\substack{\pi \leftarrow S_p, \pi' \leftarrow G \\ [X'|Y'] \leftarrow A(\pi(\pi'(\hat{\theta})))}} \left[\mathsf{Lasso}\left(\begin{bmatrix} \pi(\pi'(X)) & Y \\ X' & Y' \end{bmatrix}\right) \neq T\right] \\
&= \Pr_{\substack{\pi \leftarrow S_p, \pi' \leftarrow G \\ [X'|Y'] \leftarrow A(\pi(\hat{\theta}))}} \left[\mathsf{Lasso}\left(\begin{bmatrix} \pi(\pi'(X)) & Y \\ X' & Y' \end{bmatrix}\right) \neq T\right] \\
&= \Pr_{\substack{\pi \leftarrow S_p, \pi' \leftarrow G \\ [X'|Y'] \leftarrow A(\pi(\hat{\theta}))}} \left[\mathsf{Lasso}\left(\begin{bmatrix} X & Y \\ \pi'^{-1}(\pi^{-1}(X')) & Y' \end{bmatrix}\right) \neq T\right] \\
&= \Pr_{\substack{\pi \leftarrow S_p, \pi' \leftarrow G \\ [X'|Y'] \leftarrow A(\pi(\hat{\theta}))}} \left[\mathsf{Lasso}\left(\begin{bmatrix} X & Y \\ \pi'(\pi(X')) & Y' \end{bmatrix}\right) \neq T\right] \\
&\leq \epsilon.
\end{aligned}$$

Therefore, the proof of Theorem E.3 is complete. □

Remark E.4. *Note that in Theorem E.3 the full information adversary is an "information-theoretic" adversary. In particular, the the theorem states that the full-information adversary has all the information that is required to find the unstable coordinate and also the poisoning dataset that would result in adding the unstable coordinate. Finding the right set of poisoning examples might be computationally hard but that is not an issue since we are dealing with information theoretic adversaries. Specifically, since we do not put any restriction on time complexity, all the adversary has to do is to try all possible combinations of poisoning datasets and find the one that will add the unstable coordinate to the model. However, for Construction E.6 in next section, we not require the full-information adversary to be computationally unbounded and its running time is $O(p)$.*

E.1. Constructing the Dataset

Now we move on to constructing a dataset that is k-stable on all but one coordinate and is (k, ϵ)-resilient.

What values of k can we use? Before constructing the dataset, lets first see what values of k we can use to prove separation. The following theorem states that if $k > \omega(\lambda)$ even an oblivious adversary can add any non-relevant feature to the support set of resulting model. Therefore, in our separation, we are interested in values of $k = o(\lambda)$ as we know for $k = \omega(\lambda)$ both the oblivious and full-information adversaries have almost full advantage.

Theorem E.5. *Let $X \in \mathbb{R}^{n \times p}$ be an arbitrary matrix, $\theta^* \in [0,1]^p$ be an arbitrary vector, W be a noise vector sampled from $\mathcal{N}(0, \sigma^2)^{n \times 1}$, and let $Y = X \times \theta^* + W$. Also let λ be the penalty parameter that is used for Lasso. For any $i \in [p]$,* *there is an oblivious adversary that adds $k = 2\lambda$ labeled examples $[X' \,|\, Y']$ with ℓ_1 norm at most 1 such that*

$$i \in \mathrm{Supp}\left(\mathsf{Lasso}\left(\begin{bmatrix} X & Y \\ X' & Y' \end{bmatrix}\right)\right).$$

Theorem E.5 shows that there are oblivious adversaries that use budget 2λ and add non-relevant features to the model independent of what distribution the dataset is sampled from. On the other hand, based on Theorem D.5 we know that if the data is sampled from Gaussian distribution, for $\lambda = O(\sqrt{n})$, the Lasso estimator is robust against full-information adversaries with budget $O(\sqrt{n})$. Theorem E.5 which shows the almost tightness of the robustness bounds of Theorem D.5 makes Gaussian distribution not suitable for separating full-information and oblivious adversaries. Following, we show that by tweaking the Gaussian distribution we can achieve the separation.

Construction E.6. *Consider a vector $\theta^* \in \mathbb{R}^p$ with first s coordinates having non-zero values and last $p - s$ coordinates are equal to 0. Let $\lambda = 4\sigma\sqrt{n \times p}$ and $k < \lambda$ such that $n \geq s \cdot \log(p) + s^4 \cdot k^2$. We construct a dataset*

$$[X \,|\, Y] = \begin{bmatrix} X_0 & Y_0 \\ X_1 & Y_1 \end{bmatrix} \in \mathbb{R}^{(n+\lambda-k)\times p}$$

where X_0 is generated by first sampling from $\mathcal{N}(0, 1/4)^{n\times p}$ and setting the last coordinate to 0. Namely

$$X_0 = \begin{bmatrix} \mathcal{N}(0,1) & \dots & \mathcal{N}(0,1) & 0 \\ \vdots & \ddots & \vdots & \vdots \\ \mathcal{N}(0,1) & \dots & \mathcal{N}(0,1) & 0 \end{bmatrix}.$$

And

$$Y_0 = X_0 \times \theta^* + W$$

for W is the noise vector sampled from $\mathcal{N}(0, \sigma^2)^n$.

$X_1 \in \mathbb{R}^{(\lambda-k)\times p}$ whose all elements are equal to 0 except the last coordinate that 1 and Y_1 is a vector that is equal to 1 everywhere. Namely,

$$X_1 = \begin{bmatrix} 0 & \dots & 0 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & 1 \end{bmatrix} \text{ and } Y_1 = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}.$$

Now we prove that the dataset of Construction E.6 has all the required properties of Theorem E.5. First show that Lasso estimator can recover θ^* from the dataset $[X \,|\, Y]$ of Construction E.6.

Claim E.7. *Let $[X \,|\, Y] = \begin{bmatrix} X_0 & Y_0 \\ X_1 & Y_1 \end{bmatrix}$ be the dataset of Construction E.6. With probability at least 3/4 over the choice of X_0 and W in Construction E.6, $\mathrm{Risk}(\cdot, [X \,|\, Y])$ has a unique minimizer and we have*

$$\mathrm{Supp}(\mathsf{Lasso}([X \,|\, Y])) = \mathrm{Supp}(\theta^*).$$

In addition, we have

$$\mathsf{Lasso}([X \,|\, Y])) = \mathsf{Lasso}([X_0 \,|\, Y_0]).$$

Now we show that the dataset of Construction E.6 is k-stable on all features except the last coordinate.

Claim E.8. *With probability at least 3/4 over the choice of X_0 and W_0. The dataset $[X, Y]$ of construction E.6 is k-stable on all but the first $p - 1$ coordinates and is k-unstable on the last coordinate.*

And finally we show the the dataset of Construction E.6 is (k, ϵ)-resilient.

Claim E.9. *Let $[X \,|\, Y] = \begin{bmatrix} X_0 & Y_0 \\ X_1 & Y_1 \end{bmatrix}$ be the dataset of Construction E.6. With probability at least 3/4 over the choice of X_0 and W in Construction E.6, $[X \,|\, Y]$ is $(k, \frac{(1+q)^2}{p-s})$-resilient, where $q = \max_{i\in[p]}(|\mathsf{Lasso}\left([X_0 \,|\, Y_0]\right)_i|)$.*

Note that in Claim E.9, the value of q would be at most 1. This is because Lasso estimator would always output a model $\hat{\theta}$ in $(0,1)^p$. Moreover, since (X_0, Y_0) is sampled from a Gaussian setting of Theorem distribution, q would be very close to $\max_{i\in[p]} \theta^*_i$ as well. This is because that is very close to θ^* in ℓ_2 norm. Since $\theta^* \in (0,1)^p$ and q cannot be much larger than 1, for large enough n.

Remark E.10 (Generalization of Construction E.6)**.** *In Construction E.6, the feature matrix X_0 is sampled from Normal distribution. The reason behind this is because we need the sampled dataset to be suitable for Lasso estimator. In particular, based on Theorem D.4 and D.5, we know that sampling from Normal distribution would generate a "good" dataset for Lasso estimator to robustly recover the correct feature set, with high probability. The specifications of a "good" dataset for Lasso are explained in Theorems D.2 and D.2 in appendix. We can build the dataset of Construction E.6 based on any feature matrix X_0 that satisfies these conditions. This means that sampling from Gaussian is not necessary and as long as X_0 is "good", one can prove the separation.*

E.2. Proofs of Theorem E.5, Claim E.7, Claim E.8 and Claim E.9

We first state the following useful lemma. See supplementary material for proof of the Lemma.

Lemma E.11. *Let $X \in \mathbb{R}^{n\times p}$ and $Y \in \mathbb{R}^n$. Let $\hat{\theta}$ be a vector that minimizes $\mathrm{Risk}(\cdot, [X \,|\, Y])$. Then, for all non-zero coordinates $j \in [p]$, where $\hat{\theta}_j \neq 0$ we have*

$$\sum_{i=1}^{n} X_{(i,j)} \cdot (Y_i - \langle \hat{\theta}, X_i \rangle) = -\lambda \cdot \mathrm{Sign}(\hat{\theta}_j),$$

and for all 0 coordinates $j \in [p]$, where $\theta_j = 0$, we have

$$\left| \sum_{i=1}^{n} X_{(i,j)} \cdot (Y_i - \langle \hat{\theta}, X_i \rangle) \right| < \lambda.$$

Now, we prove Theorem E.5 and show how to construct the poisoning dataset by using the lemma above.

Proof of Theorem E.5. Consider X' which is a $k \times p$ matrix that is 0 everywhere except on the i^{th} column that is 1 and Y' is a $k \times 1$ vector that is equal to 1 everywhere. We show that by adding this matrix the adversary is able to add i^{th} coordinate to the support set of the $\hat{\theta}' = \mathsf{Lasso}\left(\left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]\right)$. To prove this, suppose the i^{th} coordinate of $\hat{\theta}'$ is 0. Thus, we have

$$\begin{aligned} &\left(\begin{bmatrix} X \\ X' \end{bmatrix}^T \times \left(\begin{bmatrix} Y \\ Y' \end{bmatrix} - \begin{bmatrix} X \\ X' \end{bmatrix} \times \hat{\theta}' \right) \right)_i \\ &= k + \left(X^T \times (Y - X \times \hat{\theta}') \right)_i . \end{aligned} \tag{1}$$

Now we prove that $\hat{\theta}'$ also minimizes the Lasso loss over $[X \,|\, Y]$. This is because for any vector θ with i^{th} coordinate 0, we have

$$\mathrm{Risk}\left(\theta, \left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]\right) = k + \mathrm{Risk}(\theta, [X \,|\, Y]).$$

Now, let $\hat{\theta}$ be the minimizer of $\mathrm{Risk}(\cdot, [X \,|\, Y])$. We know that $\hat{\theta}$ is 0 on the i^{th} coordinate. Therefore we have,

$$\begin{aligned} \mathrm{Risk}\left(\hat{\theta}, \left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]\right) &= k + \mathrm{Risk}\left(\hat{\theta}, [X \,|\, Y]\right) \\ &\geq \mathrm{Risk}\left(\hat{\theta}', \left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]\right) = k + \mathrm{Risk}(\hat{\theta}', [X \,|\, Y]). \end{aligned} \tag{2}$$

where the last inequality comes from the fact that $\hat{\theta}'$ minimizes the loss over $\left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]$. On the other hand, we know that

$$\mathrm{Risk}(\hat{\theta}', [X \,|\, Y]) \geq \mathrm{Risk}(\hat{\theta}, [X \,|\, Y]) \tag{3}$$

because $\hat{\theta}$ minimizes $\mathrm{Risk}(\cdot, [X \mid Y])$. Inequalities 2 and 3 imply that

$$\mathrm{Risk}(\hat{\theta}, [X \mid Y]) = \mathrm{Risk}(\hat{\theta}', [X \mid Y])$$

and that $\hat{\theta}'$ minimizes $\mathrm{Risk}(\cdot, [X \mid Y])$. Therefore, based on Lemma E.11, since the i^{th} coordinate of $\hat{\theta}'$ is zero we have

$$\left|(X^T \times (Y - X \times \hat{\theta}))_i\right| \leq \lambda. \tag{4}$$

Combining Equations 1 and 4 we have

$$\left| \begin{bmatrix} X \\ X' \end{bmatrix}^T \left(\begin{bmatrix} Y \\ Y' \end{bmatrix} - \begin{bmatrix} X \\ X' \end{bmatrix} \times \hat{\theta} \right)_i \right| > \lambda.$$

This, however, is a contradiction because of Lemma E.11 and the fact that the i^{th} coordinate is zero. Hence, the i^{th} coordinate could not be 0 and the proof is complete. □

Proof of Claim E.7. We first prove the uniqueness property. Let X_0' be the first $p-1$ columns of X_0. Suppose there are two solutions $\hat{\theta}_1$ and $\hat{\theta}_2$ for Lasso on $[X \mid Y]$. We show that $[X_0' \mid Y_0]$ has two solutions as well. We first observe that the last coordinates of $\hat{\theta}_1$ and $\hat{\theta}_2$ should both be 0. This is because of the fact that for any θ, we have

$$\left| \sum_{i=1}^{n} X_{(i,p)} \cdot (Y_i - \langle \hat{\theta}, X_i \rangle) \right| = |\lambda - k| < \lambda$$

which by Lemma E.11 implies that the last coordinate for any Lasso solution should be 0. Now let $\hat{\theta}_1'$ and $\hat{\theta}_2'$ be the first $p-1$ coordinates of $\hat{\theta}_1$ and $\hat{\theta}_2$ respectively. We show that $\hat{\theta}_1'$ and $\hat{\theta}_2'$ both minimize $\mathrm{Risk}(\cdot, [X_0' \mid Y_0])$. We have

$$\begin{aligned}
\mathrm{Risk}(\hat{\theta}_0, [X \mid Y]) &= \mathrm{Risk}(\hat{\theta}_0, [X_0 \mid Y_0] \\
&\quad + \mathrm{Risk}(\hat{\theta}_0, [X_1 \mid Y_1]) - 2\lambda \cdot \left\|\hat{\theta}_0\right\|_1 \\
(\text{since } (\hat{\theta}_0)_p \text{ is } 0) &= \mathrm{Risk}(\hat{\theta}_0', [X_0' \mid Y_0] \\
&\quad + \mathrm{Risk}(\hat{\theta}_0, [X_1 \mid Y_1] - 2\lambda \cdot \left\|\hat{\theta}_0\right\|_1 \\
&= \mathrm{Risk}(\hat{\theta}_0', [X_0' \mid Y_0] \\
&\quad + \lambda - k + 2\lambda \cdot \left\|\hat{\theta}_0\right\|_1 - 2\lambda \cdot \left\|\hat{\theta}_0\right\|_1
\end{aligned}$$

Similarly, we have

$$\mathrm{Risk}(\hat{\theta}_1', [X, Y \mid)] = \mathrm{Risk}(\hat{\theta}_1', [X_0' \mid Y_0] + \lambda - k$$

which implies

$$\mathrm{Risk}(\hat{\theta}_0', [X_0' \mid Y]) = \mathrm{Risk}(\hat{\theta}_1', [X_0' \mid Y])$$

and they both minimize $\mathrm{Risk}(\cdot, [X_0' \mid Y_0])$.

Now note that $[X_0' \mid Y_0]$ have all the properties of Theorem D.4 and we have $[X_0' \mid Y_0]$ with probability at least $3/4$ has a unique Lasso solution. Therefore, the Lasso solution for $[X \mid Y]$ is also unique with probability at least $3/4$. Also note that this unique solution would have the correct support set as $[X_0' \mid Y_0]$ would have the correct support set based on Theorem D.4. □

Proof of Cliam E.8. We first show the k-unstability of the last coordinate. Consider a poisoning dataset $[X' \mid Y'] \in \mathbb{R}^p$, where

$$X' = \begin{bmatrix} 0 & \dots & 0 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & 1 \end{bmatrix} \text{ and } Y' = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}.$$

We prove that $p \in \mathsf{Lasso}\left(\left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]\right)$. Suppose this is not the case and we have zero p^{th} corodinate namely, $\mathsf{Lasso}\left(\left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]\right)_p = 0$. We have

$$\left|\sum_{i=1}^{n+\lambda} X_{(i,p)} \cdot (Y_i - \langle\hat{\theta}, X_i\rangle)\right| = \lambda.$$

This is contradictory with Lemma E.11 that states this value should be less than λ because the p^{th} coordinate is 0. Therefore, the p^{th} coordinate is in the support set.

Now, we focus on proving the k-stability of all other coordinates. Suppose by adding a subset $[X' \,|\, Y']$ to $[X \,|\, Y]$ and we get a model $\hat{\theta}' = \mathsf{Lasso}\left(\left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]\right)$ that is non-zero on the i^{th} coordinate for some $p > i > s$. The idea is to build a poisoning dataset $[X'' \,|\, Y''] \in \mathbb{R}^{k\times p}$ that when added to $[X_0 \,|\, Y_0]$ causes the i^{th} coordinate to be added to the support set. Let X'' be the same matrix as X' except the last column that is set to 0. Namely,

$$X'' = \begin{bmatrix} X'_{(1,1)} & \dots & X'_{(1,p-1)} & 0 \\ \vdots & \ddots & \vdots & \vdots \\ X'_{(k,1)} & \dots & X'_{(k,p-1)} & 0 \end{bmatrix}.$$

And let

$$Y'' = Y' - (X' - X'') \times \hat{\theta}'.$$

We prove that $i \in \mathrm{Supp}\left(\mathsf{Lasso}\left(\left[\begin{array}{c|c} X_0 & Y_0 \\ X'' & Y'' \end{array}\right]\right)\right)$. Note that if we prove this, the proof would be complete as we know that $[X_0, Y_0]$ is k-stable for all coordinates with probability at least $3/4$ based on Theorem D.5 (Note that similar to the proof of Claim E.7, the fact that the last coordinate of X_0 is not sampled from Gaussian would not cause any issue). However, there is one subtle issue that might happen here, if the ℓ_∞ norm of Y'' might be larger than s, then the guarantee of theorem D.5 does not hold anymore. But that will not happen because we restrict the ℓ_∞ norm of Y' to be at most 1 and the fact that ℓ_∞ norm of $\hat{\theta}'$ is at most 1 based on the way lasso estimator is defined. This means that the ℓ_∞ norm of Y'' is at most 2. Let $\hat{\theta}'' \in \mathbb{R}^p$ be equal to $\hat{\theta}'$ everywhere except on the last coordinate that is equal to 0. We have

$$\begin{aligned}
\mathrm{Risk}(\hat{\theta}'', \left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]]) &= \left\|\begin{bmatrix} Y_0 \\ Y_1 \\ Y' \end{bmatrix} - \begin{bmatrix} X_0 \\ X_1 \\ X' \end{bmatrix} \times \hat{\theta}'\right\|_2^2 \\
&\quad + 2\cdot\lambda\cdot\left\|\hat{\theta}'\right\|_1 \\
&= \left\|Y_0 - X_0 \times \hat{\theta}''\right\|_2^2 \\
&\quad + |(\lambda - k)(1 - \hat{\theta}'_p)^2| \\
&\quad + \left\|Y' - X' \times \hat{\theta}'\right\|_2^2 \\
&\quad + 2\cdot\lambda\cdot\left\|\hat{\theta}''\right\|_1 + 2\lambda\cdot|\hat{\theta}'_p|
\end{aligned}$$

Now based on the way X'', Y'' and θ'' are defined we have $Y'' - X'' \times \theta'' = Y' - X' \times \theta'$. Therefore,

$$\begin{aligned}
\mathrm{Risk}\left(\hat{\theta}', \left[\begin{array}{c|c} X & Y \\ X' & Y' \end{array}\right]\right) &= \left\|\begin{bmatrix} Y_0 \\ Y'' \end{bmatrix} - \begin{bmatrix} X_0 \\ X'' \end{bmatrix} \times \hat{\theta}''\right\|_2^2 \\
&\quad + |(\lambda - k)(1 - \hat{\theta}'_p)^2| \\
&\quad + 2\cdot\lambda\cdot\left\|\hat{\theta}''\right\|_1 + 2\lambda\cdot|\hat{\theta}'_p| \\
&= \mathrm{Risk}(\hat{\theta}'', \left[\begin{array}{c|c} X_0 & Y_0 \\ X'' & Y'' \end{array}\right]) \\
&\quad + |(\lambda - k)(1 - \hat{\theta}'_p)^2| + 2\lambda\cdot|\hat{\theta}'_p|
\end{aligned}$$

This means in order for $\hat{\theta}'$ to minimize Risk $\left(\cdot, \begin{bmatrix} X & Y \\ X' & Y' \end{bmatrix}\right)$, the parameter $\hat{\theta}''$ should also minimize Risk $\left(\cdot, \begin{bmatrix} X_0 & Y_0 \\ X'' & Y'' \end{bmatrix}\right)$. This means that $\theta'' = \mathsf{Lasso}\left(\begin{bmatrix} X_0 & Y_0 \\ X'' & Y'' \end{bmatrix}\right)$ has the i^{th} column in its support. Hence the proof is complete. □

Proof of Claim E.9. Suppose by adding a poisoning dataset $[X' \,|\, Y']$ to $[X \,|\, Y]$ the p^{th} coordinate would be added to the support of the solution. Namely, $p \in \mathrm{Supp}(\hat{\theta}')$ for $\hat{\theta}' = \mathsf{Lasso}\left(\begin{bmatrix} X & Y \\ X' & Y' \end{bmatrix})\right)$. Now let

$$[X'' \,|\, Y''] = \begin{bmatrix} X & Y \\ X' & Y' \end{bmatrix} \in \mathbb{R}^{(n+\lambda)\times(p+1)}.$$

Based on Lemma E.11, we have

$$\sum_{i=1}^{n+k} X''_{(i,p)} \cdot (Y''_i - \langle \hat{\theta}', X''_i \rangle) = \lambda \cdot \mathrm{Sign}(\hat{\theta}'_p). \tag{5}$$

Based on the way the dataset is constructed, we have

$$\begin{aligned}
&\sum_{i=1}^{n+\lambda} X''_{(i,p)} \cdot (Y''_i - \langle \hat{\theta}', X''_i \rangle) \\
&\quad = \sum_{i=1}^{n} X''_{(i,p)} \cdot (Y''_i - \langle \hat{\theta}', X''_i \rangle) + \sum_{i=n+1}^{n+\lambda-k} X''_{(i,p)} \cdot (Y''_i - \langle \hat{\theta}', X''_i \rangle) + \sum_{i=1}^{k} X'_{(i,p)} \cdot (Y'_i - \langle \hat{\theta}', X'_i \rangle) \\
&\quad = 0 + (\lambda - k) \cdot (1 - \hat{\theta}'_p) + \sum_{i=1}^{k} X'_{(i,p)} \cdot (Y'_i - \langle \hat{\theta}', X'_i \rangle).
\end{aligned} \tag{6}$$

Therefore by combining 5 and 6 we have

$$|\sum_{i=1}^{k} X'_{(i,p)} \cdot (Y'_i - \langle \hat{\theta}', X'_i \rangle)| = |\lambda \,\mathrm{Sign}(\hat{\theta}'_p) - (\lambda - k)(1 - \hat{\theta}'_p)| > k. \tag{7}$$

Now consider the quantity $\sum_{i=1}^{k} |Y'_i - \langle \hat{\theta}', X'_i \rangle|^2$. First note that we have

$$\sum_{i=1}^{k} |Y'_i - \langle \hat{\theta}', X'_i \rangle|^2 \leq \sum_{i=1}^{k} |Y'_i - \langle \hat{\theta}'', X'_i \rangle|^2$$

where $\hat{\theta} = \mathsf{Lasso}([X \,|\, Y])$. This is correct because otherwise $\hat{\theta}''$ would have smaller loss than $\hat{\theta}'$. By Claim E.7 we know that $\hat{\theta} = \hat{\theta}''$ which implies

$$\sum_{i=1}^{k} |Y'_i - \langle \hat{\theta}', X'_i \rangle|^2 \leq \sum_{i=1}^{k} |Y'_i - \langle \hat{\theta}, X'_i \rangle|^2.$$

On the other hand we have

$$\sum_{i=1}^{k} |Y'_i - \langle \hat{\theta}, X'_i \rangle|^2 \leq \sum_{i=1}^{k} (1 + q|X_i|)^2 \leq k(1+q)^2 \tag{8}$$

which in turn implies

$$\sum_{i=1}^{k} |Y'_i - \langle \hat{\theta}', X'_i \rangle|^2 \leq k(1+q)^2. \tag{9}$$

Now by Cauchy–Schwarz inequality we have

$$\left(\sum_{i=1}^{k} |Y'_i - \langle \hat{\theta}', X'_i \rangle|^2\right) \cdot \left(\sum_{i=1}^{k} X'^2_{(i,p)}\right) > \left(\sum_{i=1}^{k} |Y'_i - \langle \hat{\theta}', X'_i \rangle| \cdot X'_{(i,p)}\right)^2. \tag{10}$$

Combining inequalities 10, 7 and 9 we get

$$\sum_{i=1}^{k} X'^2_{(i,p)} > k/(1+q)^2. \tag{11}$$

This means that the average of last coordinate of X' should have most of the weight of the whole matrix. In particular, since $X'_{(i,j)} < 1$ for all (i,j) we have

$$\sum_{i=1}^{k} |X'_{(i,p)}| > k/(1+q)^2.$$

Also since each row in X' have ℓ_1 norm bounded by 1, we have

$$\sum_{i=1}^{k} \sum_{j=1}^{p} |X'_{(i,p)}| \le k.$$

This implies that number of columns j for which

$$\sum_{i=1}^{k} |X'_{(i,j)}| > k/(1+q)^2$$

holds is at most $(1+q)^2$. Therefore, for any $[X' \mid Y']$ the probability of $\pi([X' \mid Y'])$ having sum at least k over the last column is at most $(1+q)^2/(p-s)$. This means the probability of $\pi(X')$ adding the p^{th} column to the support set would be at most $(1+q)^2/(p-s)$. □

F. Separating Oblivious and Full-information Poisoning on Classification

In this section, we show a separation on the power of oblivious and full-information poisoning attacks on classification. In particular we show that empirical risk minimization (ERM) algorithm could be much more susceptible to full-information poisoning adversaries, compared to oblivious adversaries. We first restate Theorem 3.2 for convenience.

Theorem 3.2[Restated]. *There is a distribution of distributions $\mathfrak{D}$ such that there is a data injecting adversary with budget $\varepsilon \cdot n$ that wins the full-information security game for classification by advantage ε, namely*

$$\exists A: \mathop{\mathbb{E}}_{\substack{D \leftarrow \mathfrak{D} \\ \mathcal{S} \leftarrow D^n}} \Big[\text{Advantage of } A \text{ in } \mathbf{FullRisk}(\varepsilon \cdot n, \mathcal{S}, \mathsf{ERM}, D)\Big] \ge \Omega(\varepsilon).$$

On the other hand, any adversary will have much smaller advantage in the oblivious game. Namely, the following holds.

$$\forall A: \mathop{\mathbb{E}}_{\substack{D \leftarrow \mathfrak{D} \\ \mathcal{S} \leftarrow D^n}} \Big[\text{Advantage of } A \text{ in } \mathbf{OblRisk}(\varepsilon \cdot n, \mathcal{S}, \mathsf{ERM}, D)\Big] \le O(\varepsilon^2).$$

Proof. Here we only sketch the proof. To prove this we use the problem of learning concentric halfspaces in Gaussian space $\mathcal{N}(0,1)^2$. We assume that the prior distribution is uniform over all concentric halfspaces. We first show that there is a full-information attack with success (ε). The way this attack works is as follows, attacker first uses ERM to learn a halfspace w_1 on the clean data. Assume this halfspace has risk δ. Then the attacker selects another halfspace w_2 that disagrees with w_1 on $\varepsilon \cdot n - 1$ number of points in the training data. Note that this is possible because the attacker can keep rotating the half-space until it has exactly $n \cdot \varepsilon - 1$ points disagreeing with w_1. Now if the adversary puts all the poison points on the separating line for w_1 and with the opposite label of what w_1 predicts, then ERM would prefer w_2 over w_1. Therefore the empirical error of w_2 on clean dataset would be at least equal to $\varepsilon - \delta$. Now if we increase n, the generalization error would go to zero which means the population error of w_2 would be close to $\varepsilon - \delta$. Also, since we are assuming the problem is realizable by half-spaces, it means δ would also converge to 0. Therefore, the final population risk could be bounded to be at least $\varepsilon/2$ for n larger than some reasonable values. Which means our proof for the full-information attack is complete.

Now, we show that no oblivious adversary cannot increase the error by more than ε^2, on average. The reason behind this boils down to the fact that each poison point added can affect at most ϵ-fraction of the choices of ground truth. To be more specific, we can fix the poison data to a fixed set D_p with size $\epsilon \cdot n$, as we can assume that the oblivious adversary is deterministic. Now if we fix the ground truth to some w^g, and define the epsilon neighborhood of a model w to be all the points that have angle at most $\epsilon \cdot \pi$ with w and denote it by w_ϵ. Then we have

$$\begin{aligned}\mathop{\mathbb{E}}_{\substack{X_c \leftarrow \mathcal{N}(0,1)^n\\ y_c = w^g(X_c)\\ D_c=(X_c,y_c)\\ w^p=\mathsf{ERM}(D_c\cup D_p), w^c=\mathsf{ERM}(D_c)}}[\mathsf{Risk}(w^p) - \mathsf{Risk}(w^c)] &\leq \mathop{\mathbb{E}}_{\substack{X_c \leftarrow \mathcal{N}(0,1)^n\\ y_c = w^g(X_c)\\ D_c=(X_c,y_c)\\ w^p=\mathsf{ERM}(D_c\cup D_p)}}[\mathsf{Risk}(\mathsf{ERM}(w^p))]\\ &\leq \mathop{\mathbb{E}}_{\substack{X_c \leftarrow \mathcal{N}(0,1)^n\\ y_c = w^g(X_c)\\ D_c=(X_c,y_c)\\ w^p=\mathsf{ERM}(D_c\cup D_p)}}[\mathsf{Risk}_{D_c}(w^p)] + \delta \qquad (12)\end{aligned}$$

Where δ is the generalization parameter that relates to n and goes to 0 with rate $1/n$. Now consider an event E where the angle between w^c and w^g is at most $\epsilon \cdot \pi$ and $w^g_{2\epsilon} \cap X_c$ has at least ϵ points on each side of w^g. We denote the probability of this event by $1 - \delta'$ and we know that δ' goes down to 0 as n grows, by rate $1/\sqrt{n}$ (Using Chernoff Bound). Now we can observe that conditioned on E, we have $\mathsf{Risk}_{D_c}(w_p) \leq |w^g_{2\epsilon} \cap X_c|$. This is because the poison points cannot increase the errorn by more than ϵ so w^p would disagree with w^c on at most $\epsilon \cdot n$ points in D_c. On the other hand, we know that in 2ϵ neighborhood of w_g there are at least $\epsilon \cdot n$ points on each side of w_g, which means there are at least $\epsilon \cdot n$ points on each side of w^c (because w^c and w^g would fall between the same two points in D_c). Therefore, the poisoned model, would definitely be in the $2 \cdot \epsilon$ neighborhood of the w_g. At the same time, we know that the maximum number of points in D_c that w^g and w^p disagree on are at most equal to the number of poison points that fall in their disagreement region. And since the disagreement region is a subset of $w^g_{2\epsilon}$, we have the maximum number of points in D_c that w^g and w^p disagree on are at most equal to $|w^g_{2\epsilon} \cap X_c|$. Now having this, using Equation (12) we can write

$$\mathop{\mathbb{E}}_{\substack{X_c \leftarrow \mathcal{N}(0,1)^n\\ y_c = w^g(X_c)\\ D_c=(X_c,y_c)\\ w^p=\mathsf{ERM}(D_c\cup D_p), w^c=\mathsf{ERM}(D_c)}}[\mathsf{Risk}(w^p) - \mathsf{Risk}(w^c)] \leq \frac{|D_p \cap w^g_{2\epsilon}|}{n} + \delta + \delta'$$

Now by also taking the average over w^g we get

$$\mathop{\mathbb{E}}_{\substack{w^g \leftarrow \mathfrak{D}\\ X_c \leftarrow \mathcal{N}(0,1)^n\\ y_c = w^g(X_c)\\ D_c=(X_c,y_c)\\ w^p=\mathsf{ERM}(D_c\cup D_p), w^c=\mathsf{ERM}(D_c)}}[\mathsf{Risk}(w^p) - \mathsf{Risk}(w^c)] \leq \mathop{\mathbb{E}}_{w^g \leftarrow \mathfrak{D}}[\frac{|D_p \cap [w^g_{2\epsilon}|}{n}] + \delta + \delta' = 2\epsilon^2 + \delta + \delta'$$

As δ and δ' converge to 0 with rate $1/\sqrt{n}$, for $n \geq \omega(1/\epsilon^2)$ we have

$$\mathop{\mathbb{E}}_{\substack{w^g \leftarrow \mathfrak{D}\\ X_c \leftarrow \mathcal{N}(0,1)^n\\ y_c = w^g(X_c)\\ D_c=(X_c,y_c)\\ w^p=\mathsf{ERM}(D_c\cup D_p), w^c=\mathsf{ERM}(D_c)}}[\mathsf{Risk}(w^p) - \mathsf{Risk}(w^c)] \leq O(\epsilon^2).$$

□

We also state the theorem about separation of oblivious and full-information adversaries in the data elimination setting. This theorem has shows that the gap between oblivious and full-information adversaries could be wider in the data elimination settings. We use $\mathbf{FullRisk}^{\mathsf{elim}}$ and $\mathbf{OblRisk}_{\mathsf{elim}}$ to denote the information risk in presence of oblivious and full-information data elimination attacks.

Theorem F.1. *There is a distribution of distributions $\mathfrak{D}$ such that there is a data elimination adversary with budget $\varepsilon \cdot n$ that wins the full-information security game for classification by advantage ε, namely*

$$\exists A : \mathop{\mathbb{E}}_{\substack{D \leftarrow \mathfrak{D} \\ \mathcal{S} \leftarrow D^n}} \left[\text{Advantage of } A \text{ in } \mathbf{FullRisk}^{\mathsf{elim}}(\varepsilon \cdot n, \mathcal{S}, \mathsf{ERM}, D)) \right] \geq \Omega(\varepsilon).$$

On the other hand, any adversary will have much smaller advantage in the oblivious game. Namely, the following holds.

$$\forall A : \mathop{\mathbb{E}}_{\substack{D \leftarrow \mathfrak{D} \\ \mathcal{S} \leftarrow D^n}} \left[\text{Advantage of } A \text{ in } \mathbf{OblRisk}_{\mathsf{elim}}(\varepsilon \cdot n, \mathcal{S}, \mathsf{ERM}, D)) \right] \leq e^{-\omega((1-\varepsilon)n)}.$$

Proof. For the negative part on the power of full-information attacks, we observe that for a fixed w_g the attacker can find a half-space w_c that has angle $\pi\epsilon \cdot /2$ with the ground-truth w_g, and remove all the points where w_c and w_g disagree. Note that the number of points in the disagreement region would be at most ϵ with some large probablity $1 - \delta$ where δ goes to 0 with rate $1/\sqrt{n}$. After the adversary removes all the points in disagreement region, the learner cannot distinguish them and will incur an error $\epsilon/2$ on average. We note that this attack is similar to the hybrid attack described in Diochnos et al. (2019). For the positive result, we make a simple observation that oblivious poisoning adversary can only reduce the sample complexity for the learner. In other words, non-removed examples would remain i.i.d examples. This means that after removal, we can still use uniform convergence theorem to bound the error of resulting classifier. Since the error of learning realizable half-spcaces will go to zero with rate $\Omega(1/n)$, therefore the average error after the attack would be $\Omega(1/(1-\epsilon)n))$. □

G. Feature Selection Experiments; More Details

In this section, we give further details about the particular construction for our model recover experiments in Section 4 and the real-world experiments for larger datasets.

G.1. Synthetic Experiments

We construct our dataset exactly as in Construction E.6, so we have:

$$[X \,|\, Y] = \begin{bmatrix} X_0 & Y_0 \\ X_1 & Y_1 \end{bmatrix} \in \mathbb{R}^{(n+\lambda-k)\times p},$$

In particular, we set $n = 1000$, $p = 100,000$, $k = 1$, and $s = 5$ in our construction, so our dataset has $999 + \lambda$ rows and $100,000$ features. We set $\sigma = 1/4$ for our noise vector W, which determines λ. We use scikit-learn's implementation of Lasso (Pedregosa et al., 2011), setting the regularization parameter to $2\lambda/n$, as defined in Section 1. Due to numerical instability, we use $\lambda = 121$ (which is slightly higher than the $4\sigma\sqrt{n\log(p)}$ in the construction) to instantiate the dataset, giving us:

$$[X \,|\, Y] = \begin{bmatrix} X_0 & Y_0 \\ X_1 & Y_1 \end{bmatrix} \in \mathbb{R}^{(n+\lambda-k)\times p} \in \mathbb{R}^{1,120\times 100,000}$$

where X_0, Y_0, X_1, and Y_1 are given in Construction E.6.

G.2. Real-world Experiment

For our real-world datasets, we use four datasets typically used in the model recovery setting:

- **Boston.** (Harrison & Rubinfeld, 1978) (506 examples, 13 features) The task in this dataset is to predict the median value of a house in the Boston, Mass. area, given attributes that describe its location, features, and surrounding area. The outcome variable is continuous in the range $[0, 50]$.
- **TOX.** (Golub et al., 1999b) (171 examples, 5,748 features) The task in this dataset is to predict whether a patient is a myocarditis and dilated cardiomyopathy (DCM) infected male, a DCM infected female, an uninfected male, or an uninfected female. Each feature is a gene, and each example is a patient. The outcome variable is discrete in $\{1, 2, 3, 4\}$, for each of the four possibilities.

- **Prostate_GE.** (Golub et al., 1999a) (102 examples, 5,966 features) The task in this dataset is to predict whether a patient has prostate cancer. Each feature is a gene, and each example is a patient. The outcome variable is binary in $\{0, 1\}$, for cancer or no cancer.
- **SMK.** (Golub et al., 1999b) (187 examples, 19,993 features) The task in this dataset is to predict whether a smoker has lung cancer or not. Each example is a smoker, and each faeture is a gene. The outcome variable is binary in $\{0, 1\}$ for cancer or no cancer.

For the larger datasets, we see the same phenomenon as the Boston dataset in Section 4, where there certainly exist unstable features that require a small number of rows to poison, and stable features that require a large number of rows, as shown in Figure 3. Even in subsets of 50 features from the full dataset, we see the same phenomenon, seen in 4. For instance, in TOX, feature 106 requires less than 5 poison rows to add it to the dataset, while the average is 22 and the maximum number of rows is above 60. An non-oblivious adversary could exploit feature 106, while an oblivious adversary would pick a feature that takes 22 poison rows, in expectation. In Figure 3, we show the same phenomenon on each dataset for a subset of 50 features for visual clarity.

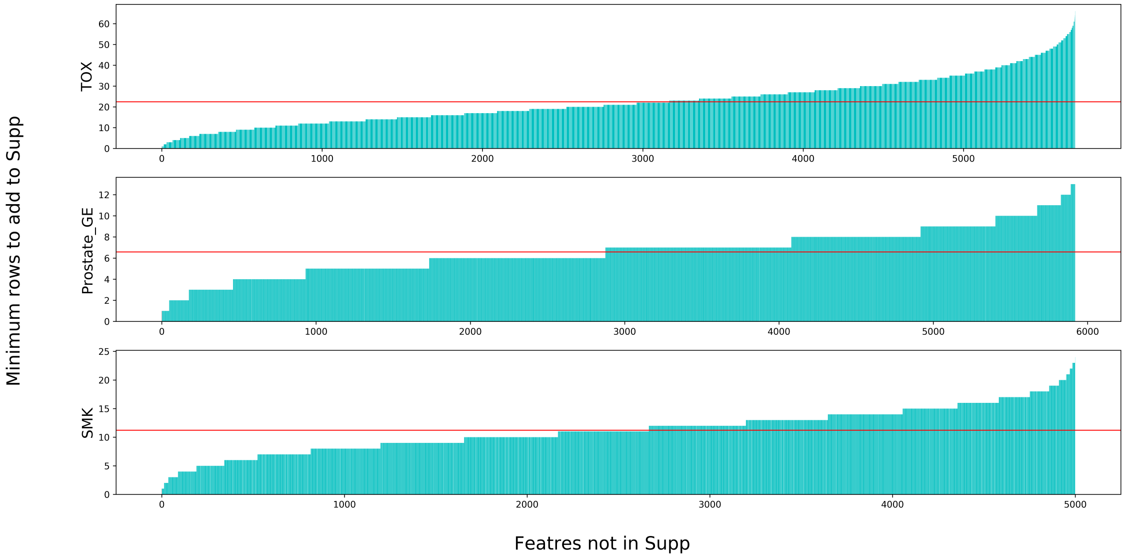

Figure 3. TOX, Prostate_GE, SMK (full dataset). We attacked all the features not in $\mathrm{Supp}(\hat{\theta})$, and plotted the number of rows needed to add feature i to the dataset. The red horizontal line is the mean number of rows. The top figure shows Prostate_GE, the middle TOX, and the bottom SMK.

H. Separating Oblivious and Full-information Classification: Experiments

In this section, we design an experiment to empirically validate the claim made in Theorem 3.2, that there is a separation between oblivious and full-information poisoning adversaries for classification. We setup the experiment just as in the proof of Theorem 3.2, as follows.

First, we sample training points $X = x_1, x_2, \dots x_m$ for $m = 1,000$ from the Gaussian space $\mathcal{N}(0, 1)^2$, and pick a random ground-truth halfspace w^* from $\mathcal{N}(0, 1)^2$. Using w^*, we find our labels $y_1, y_2, \dots y_m$ by taking $(w^*)^T x_k$ for $k \in [m]$. This ensures the data is linearly separable by the homogeneous halfspace produced by w^*.

To attack this dataset simulating our full-information adversary with budget ϵ, we construct $\epsilon \cdot m$ poison points p as follows:

$$p = \cos(\epsilon\pi) \cdot \frac{v}{\|v\|} + \sin(\epsilon\pi) \cdot \frac{w}{\|w\|}, \quad \text{where } v = \begin{bmatrix} 1 & -\frac{w_1}{w_2} \end{bmatrix}$$

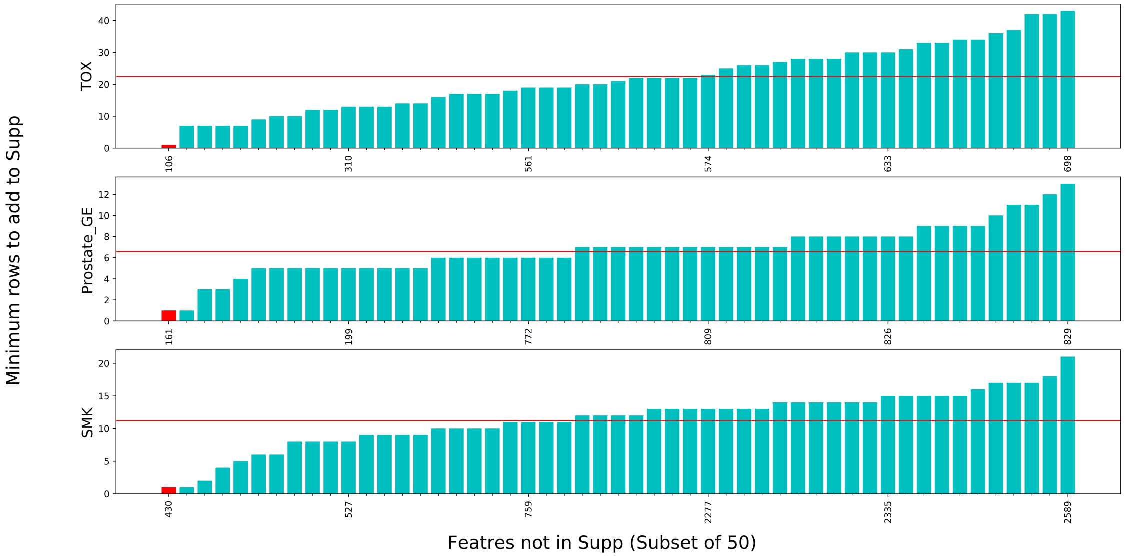

Figure 4. TOX, Prostate_GE, SMK (50 features). We choose a random subset of 50 features not in $\mathrm{Supp}(\hat{\theta})$, and plotted the number of rows needed to add feature i to the dataset. The red horizontal line is the mean number of rows. The top figure shows Prostate_GE, the middle TOX, and the bottom SMK. The red colored bar shows the k-unstable feature in the dataset that a non-oblivious adversary could exploit.

and we add $\epsilon \cdot m$ of these p rows to our dataset. Note that this specific p corresponds to halfspace w_2 in our Proof of Theorem 3.2, the halfspace obtained by rotating the original halfspace until it has exactly $\epsilon \cdot m$ points disagreeing with w^*. We label each of these p rows to be $y_p = -(w^*)^T p$, the opposite label from ground-truth. Then, we train our halfspace via ERM on this poisoned dataset of $m \cdot (1 + \epsilon)$ points (from appending $\epsilon \cdot m$ rows of p). We evaluate our poisoned halfspace on another $X' = x'_1, x'_2, \ldots x'_m$ test points from the same Gaussian $\mathcal{N}(0, 1)^n$ distribution.

To attack this dataset simulating the oblivious adversary, we try three oblivious strategies of attack that an adversary with no knowledge of the dataset might wage, each with ϵ budget:

1. Sample a single random point p from $\mathcal{N}(0, 1)^n$ and repeat it $\epsilon \cdot m$ times. Choose the label p_y uniformly at random from $\{-1, 1\}$. Poison by adding these $\epsilon \cdot m$ rows to the dataset.

2. Sample $\epsilon \cdot m$ points IID from $\mathcal{N}(0, 1)^n$ and choose the label p_y uniformly at random from $\{-1, 1\}$. Label all of the $\epsilon \cdot m$ points with p_y. Poison by adding these $\epsilon \cdot m$ rows to the dataset.

3. Sample $\epsilon \cdot m$ points IID from $\mathcal{N}(0, 1)^n$ and choose the label p_y uniformly at random from $\{-1, 1\}$ for *each point.* That is, we flip a coin to label each poison example, rather than just choosing one label, as in 2. Poison by adding these $\epsilon \cdot m$ rows to the dataset.

We also use the same ERM algorithm, as in the full-information case, to train the poisoned classifiers on these three oblivious poisoning strategies.

We repeat this experiment 20 times for poison budget $\epsilon \in \{0, 0.01, 0.02, \ldots 0.19, 0.2\}$. We observe in Figure 5 that there indeed exists a separation between the power of our full-information adversary and the oblivious adversaries. The full-information adversary can increase the error linearly with ϵ using this strategy, while the oblivious adversaries fail to have any consistent impact on the resulting classifier's error with their strategies.

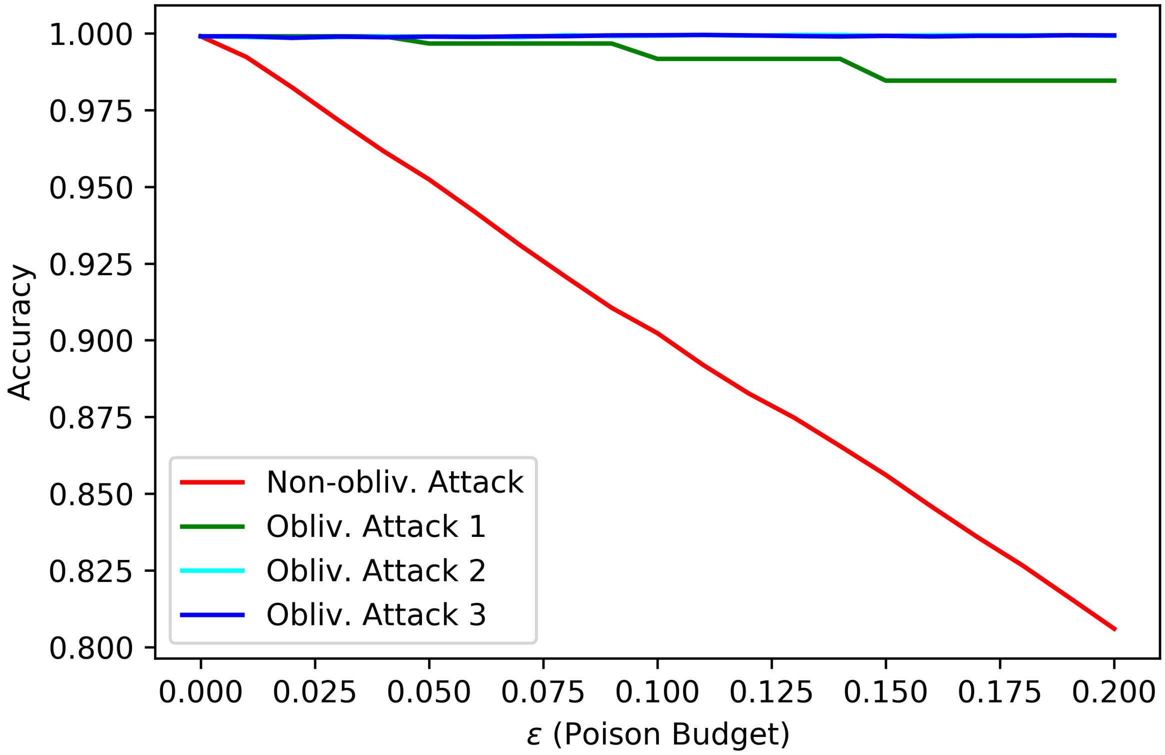

Figure 5. Oblivious and full-information poisoning separation in classification. Over 20 trials, we vary the poisoning budget ϵ and construct poisoned datasets as discussed above for each adversary. We plot the effect of each adversary's attack on the accuracy of our resulting poisoned ERM halfspace.