STATIC PREDICTION GAMES FOR ADVERSARIAL LEARNING PROBLEMS

Michael Brückner[1], Christian Kanzow[2], and Tobias Scheffer[1]

Preprint 304 August 2011

[1] University of Potsdam
Department of Computer Science
August-Bebel-Str. 89, 14482 Potsdam, Germany

e-mail: mibrueck@cs.uni-potsdam.de
scheffer@cs.uni-potsdam.de

[2] University of Würzburg
Institute of Mathematics
Emil-Fischer-Str. 30, 97074 Würzburg, Germany
e-mail: kanzow@mathematik.uni-wuerzburg.de

August 11, 2011

Abstract. The standard assumption of identically distributed training and test data is violated when the test data are generated in response to the presence of a predictive model. This becomes apparent, for example, in the context of email spam filtering. Here, email service providers employ spam filters and spam senders engineer campaign templates such as to achieve a high rate of successful deliveries despite any filters. We model the interaction between learner and data generator as a static game in which the cost functions of learner and data generator are not necessarily antagonistic. We identify conditions under which this prediction game has a unique Nash equilibrium and derive algorithms that find the equilibrial prediction model. We derive two instances, the Nash logistic regression and the Nash support vector machine, and empirically explore their properties in a case study on email spam filtering.

Key Words: Static Prediction Games, Adversarial Classification, Nash Equilibrium

1. Introduction

A common assumption on which most learning algorithms are based is that training and test data are governed by identical distributions. However, in a variety of applications, the distribution that governs data at application time may be influenced by an adversary whose interests are in conflict with those of the learner. Consider, for instance, the following three scenarios. In computer and network security, scripts that control attacks are engineered with botnet and intrusion detection systems in mind. Credit card fraudsters adapt their unauthorized use of credit cards—in particular, amounts charged per transactions and per day and the type of businesses that amounts are charged from—such as not to trigger alerting mechanisms employed by credit card companies. Email spam senders design message templates that are instantiated by nodes of botnets. These templates are specifically designed to produce a low spam score with popular spam filters. The domain of email spam filtering will serve as a running example throughout the paper. In all of these applications, the party that creates the predictive model and the adversarial party that generates future data are aware of each other, and factor the possible actions of their opponent into their decisions.

The interaction between learner and data generators can be modeled as a game in which one player controls the predictive model whereas another exercises some control over the process of data generation. The adversary's influence on the generation of the data can be formally modeled as a transformation that is imposed on the distribution that governs the data at training time. The transformed distribution then governs the data at application time. The optimization criterion of either player takes as arguments both, the predictive model chosen by the learner and the transformation carried out by the adversary.

Typically, this problem is modeled under the worst-case assumption that the adversary desires to impose the highest possible costs on the learner. This amounts to a zero-sum game in which the loss of one player is the gain of the other. In this setting, both players can maximize their expected outcome by following a minimax strategy. Lanckriet et al. (2002) study the minimax probability machine (MPM). This classifier minimizes the maximal probability of misclassifying new instances for a given mean and covariance matrix of each class. Geometrically, these class means and covariances define two hyper-ellipsoids which are equally scaled such that they intersect; their common tangent is the minimax probabilistic decision hyperplane. Ghaoui et al. (2003) derive a minimax model for input data that are known to lie within some hyper-rectangles around the training instances. Their solution minimizes the worst-case loss over all possible choices of the data in these intervals. Similarly, worst-case solutions to classification games in which the adversary deletes input features (Globerson and Roweis 2006; Globerson et al. 2009) or performs an arbitrary feature transformation (Teo et al. 2007; Dekel and Shamir 2008; Dekel et al. 2010) have been studied.

Several applications motivate problem settings in which the goals of the learner and the data generator, while still conflicting, are not necessarily entirely antagonistic. For instance, a fraudster's goal of maximizing the profit made from exploiting phished account information is not the inverse of an email service provider's goal of achieving a high spam recognition rate at close-to-zero false positives. When playing a minimax strategy, one often makes overly pessimistic assumptions about the adversary's behavior and may not necessarily obtain an optimal outcome.

Games in which a leader—typically, the learner—commits to an action first whereas the adversary can react after the leader's action has been disclosed are naturally modeled as a *Stackelberg competition*. This model is appropriate when the follower—the data generator—has full information about the predictive model. This assumption is usually a pessimistic approximation of reality because, for instance, neither email service providers nor credit card companies disclose a comprehensive documentation of their current security measures. Stackelberg equilibria of adversarial classification problems can be identified by solving a bilevel optimization problem (Brückner and Scheffer, 2011).

This paper studies *static* prediction games in which both players act simultaneously; that is, without prior information on their adversary's move. When the optimization criterion of either player depends not only on the player's own action but also on the adversary's move, then the concept of a player's optimal action is no longer well-defined. Therefore, we establish the concept of a *Nash equilibrium* of static prediction games. A Nash equilibrium is a pair of actions chosen such that no player gains a benefit by unilaterally selecting a different action. If a game has a unique Nash equilibrium and is played by rational players that aim at maximizing their optimization criteria, it is reasonable for each player to assume that the opponent will follow the Nash equilibrium. If one player follows the Nash equilibrium, the optimal move for the other player is to follow this equilibrium as well. If, however, multiple equilibria exist and the players choose their action according to distinct ones, then the resulting combination may be arbitrarily disadvantageous for either player. It is therefore interesting to study whether adversarial prediction games have a unique Nash equilibrium.

Our work builds on a prior publication (Brückner and Scheffer, 2009) that has identified conditions under which a unique Nash equilibrium of a static prediction game exists and developed an algorithm which identifies this equilibrial model. We will discuss a flaw in Theorem 1 of Brückner and Scheffer (2009) and develop a revised version of the theorem that identifies conditions under which a unique Nash equilibrium of a prediction game exists. In addition to the inexact linesearch approach to finding the equilibrium that Brückner and Scheffer (2009) develop, we will follow a modified extragradient approach and develop Nash logistic regression and the Nash support vector machine. This paper also develops a kernelized version of these methods. An extended empirical evaluation explores the applicability of the Nash instances in the context of email spam filtering. We empirically verify the assumptions made in the modeling process and compare the performance of Nash instances with baseline methods on several email corpora, including a corpus from an email service provider.

The rest of this paper is organized as follows. Section 2 introduces the problem setting. We formalize the Nash prediction game and study conditions under which a unique Nash equilibrium exists in Section 3. Section 4 develops strategies for identifying equilibrial prediction models, and in Section 5 we detail on two instances of the Nash prediction game. In Section 6, we report on experiments on email spam filtering; Section 7 concludes.

2. Problem Setting

We study static prediction games between two players: The *learner* ($v = -1$) and an adversary, the *data generator* ($v = +1$). In our running example of email spam filtering, we study the competition between recipient and senders, not competition among senders. Therefore, $v = -1$ refers to the recipient whereas $v = +1$ models the entirety of all legitimate and abusive email senders as a single, amalgamated player.

In the *past*, the data generator $v = +1$ produced a sample $D = \{(x_i, y_i)\}_{i=1}^n$ of n training instances $x_i \in \mathcal{X}$ with corresponding class labels $y_i \in \mathcal{Y} = \{-1, +1\}$. These object-class pairs are drawn according to a training distribution with density function $p(x, y)$. By contrast, *future* object-class pairs, produced by the data generator at application time, are drawn from some test distribution with density $\dot{p}(x, y)$ which may significantly differ from $p(x, y)$.

The task of the learner $v = -1$ is to select the parameters $\mathbf{w} \in \mathcal{W} \subset \mathbb{R}^m$ of a predictive model $h(x) = \operatorname{sign} f_{\mathbf{w}}(x)$ implemented in terms of a generalized linear decision function $f_{\mathbf{w}} : \mathcal{X} \to \mathbb{R}$ with $f_{\mathbf{w}}(x) = \mathbf{w}^{\mathsf{T}}\phi(x)$ and feature mapping $\phi : \mathcal{X} \to \mathbb{R}^m$. The learner's theoretical *costs* at application time are given by

$$\theta_{-1}(\mathbf{w}, \dot{p}) = \sum_{\mathcal{Y}} \int_{\mathcal{X}} c_{-1}(x, y)\ell_{-1}(f_{\mathbf{w}}(x), y)\dot{p}(x, y)\mathrm{d}x,$$

where weighting function $c_{-1} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ and loss function $\ell_{-1} : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}$ detail the *weighted loss* $c_{-1}(x, y)\ell_{-1}(f_{\mathbf{w}}(x), y)$ that the learner incurs when the predictive model classifies instance x as $h(x) = \operatorname{sign} f_{\mathbf{w}}(x)$ while the true label is y. The positive class- and instance-specific weighting factors $c_{-1}(x, y)$ with $\mathbb{E}_{\mathbf{X},\mathbf{Y}}[c_{-1}(x, y)] = 1$ specify the importance of minimizing the loss $\ell_{-1}(f_{\mathbf{w}}(x), y)$ for the corresponding object-class pair (x, y). For instance, in spam filtering, the correct classification of non-spam messages can be business-critical for email service providers while failing to detect spam messages runs up processing and storage costs, depending on the size of the message.

The data generator $v = +1$ can modify the data generation process for future instances. In practice, spam senders update their campaign templates which are disseminated to the nodes of botnets. Formally, the data generator transforms the training distribution with density p to the test distribution with density $\dot{p}$. The data generator incurs *transformation costs* by modifying the data generation process which is quantified by $\Omega_{+1}(p, \dot{p})$. This term acts as a regularizer on the transformation and may implicitly constrain the space of possible distribution shifts, depending on the nature of the application that is to be modeled. For instance, the email sender may not be allowed to alter the training distribution for non-spam messages, or to modify the nature of the messages by changing the label from *spam* to *non-spam* or vice versa. Additionally, changing the training distribution for spam messages may run up costs depending on the extent of distortion inflicted on the informational payload. The theoretical *costs* of the data generator at application time are the sum of the expected prediction costs and the transformation costs,

$$\theta_{+1}(\mathbf{w}, \dot{p}) = \sum_{\mathcal{Y}} \int_{\mathcal{X}} c_{+1}(x, y)\ell_{+1}(f_{\mathbf{w}}(x), y)\dot{p}(x, y)\mathrm{d}x + \Omega_{+1}(p, \dot{p}),$$

where, in analogy to the learner's costs, $c_{+1}(x, y)\ell_{+1}(f_{\mathbf{w}}(x), y)$ quantifies the weighted loss that the data generator incurs when instance x is labeled as $h(x) = \operatorname{sign} f_{\mathbf{w}}(x)$ while the true label is y. The weighting factors $c_{+1}(x, y)$ with $\mathbb{E}_{\mathbf{X},\mathbf{Y}}[c_{+1}(x, y)] = 1$ express the significance of (x, y) from the perspective of the data generator. In our example scenario, this allows to reflect that costs of correctly or incorrectly classified instances may vary greatly across different physical senders that are aggregated into the amalgamated player.

Since the theoretical costs of both players depend on the test distribution, they can, for all practical purposes, not be calculated. Hence, we focus on a regularized, empirical counterpart of the theoretical costs based on the training sample D. The empirical counterpart $\hat{\Omega}_{+1}(D, \dot{D})$ of the data generator's regularizer $\Omega_{+1}(p, \dot{p})$ penalizes the divergence between training sample $D = \{(x_i, y_i)\}_{i=1}^n$ and a perturbated training sample $\dot{D} = \{(\dot{x}_i, y_i)\}_{i=1}^n$ that would be the outcome of applying the transformation that translates p into $\dot{p}$ to sample D. The learner's cost function, instead of integrating over $\dot{p}$, sums over the elements of the perturbated training sample $\dot{D}$. The players' empirical cost functions can still only be evaluated after the learner has committed to parameters $\mathbf{w}$ and the data generator to a transformation from training to test density function. However this transformation needs only be represented in terms of the effects that it will have on the training sample D. The transformed training sample $\dot{D}$ must not be mistaken for test data; test data will be generated under $\dot{p}$ at application time after the players have committed to their actions.

The empirical costs incurred by the predictive model $h(x) = \operatorname{sign} f_{\mathbf{w}}(x)$ with parameters $\mathbf{w}$ and the shift from p to $\dot{p}$ amount to

$$\hat{\theta}_{-1}(\mathbf{w}, \dot{D}) = \sum_{i=1}^{n} c_{-1,i}\ell_{-1}(f_{\mathbf{w}}(\dot{x}_i), y_i) + \rho_{-1}\hat{\Omega}_{-1}(\mathbf{w}), \tag{1}$$

$$\hat{\theta}_{+1}(\mathbf{w}, \dot{D}) = \sum_{i=1}^{n} c_{+1,i}\ell_{+1}(f_{\mathbf{w}}(\dot{x}_i), y_i) + \rho_{+1}\hat{\Omega}_{+1}(D, \dot{D}), \tag{2}$$

where we have replaced the weighting terms $\frac{1}{n}c_v(\dot{x}_i, y_i)$ by constant cost factors $c_{v,i} > 0$ with $\sum_i c_{v,i} = 1$. The learner's regularizer $\hat{\Omega}_{-1}(\mathbf{w})$ in (1) accounts for the fact that $\dot{D}$ does not constitute the test data itself, but is merely a training sample transformed to reflect the test distribution and then used to learn the model parameters $\mathbf{w}$. The trade-off between the empirical loss and the regularizer is controlled by each player's regularization parameter $\rho_v > 0$ for $v \in \{-1, +1\}$.

Note that either player's empirical costs $\hat{\theta}_v$ depend on both players' actions $\mathbf{w} \in \mathcal{W}$ and $\dot{D} \subseteq \mathcal{X} \times \mathcal{Y}$, respectively. Because of the potentially conflicting players' interests, the decision process for $\mathbf{w}$ and $\dot{D}$ becomes a noncooperative two-player game which we call a *prediction game*. In the following section, we will refer to the *Nash prediction game* (NPG) which identifies the concept of an optimal move of the learner and the data generator under the assumption of simultaneously acting players.

3. The Nash Prediction Game

The outcome of a prediction game is one particular combination of actions $(\mathbf{w}^*, \dot{D}^*)$ that runs up costs $\hat{\theta}_v(\mathbf{w}^*, \dot{D}^*)$ for the players. Each player is aware that this outcome is affected by *both players' action* and that, consequently, their potential to choose an action can have an impact on the other player's decision. In general, there is no action that minimizes one player's cost function independent of the other player's action. In a noncooperative game, the players are not allowed to communicate while making their decisions and therefore they have no information about the other player's strategy. In this setting, any concept of an optimal move requires additional assumptions on how the adversary will act.

We model the decision process for $\mathbf{w}^*$ and $\dot{D}^*$, as a *static* two-player game with *complete information.* In a *static* game, both players commit to an action simultaneously, without information about their opponent's action. In a game with *complete information*, both players know their opponent's cost function and action space.

When $\hat{\theta}_{-1}$ and $\hat{\theta}_{+1}$ are known and *antagonistic*, the assumption that the adversary will seek the greatest advantage by inflicting the greatest damage on $\hat{\theta}_{-1}$ justifies the *minimax strategy* $\operatorname{argmin}_{\mathbf{w}} \max_{\dot{D}} \hat{\theta}_{-1}(\mathbf{w}, \dot{D})$. However, when the players' cost functions are not antagonistic, assuming that the adversary will inflict the greatest possible damage is overly pessimistic. Instead assuming that the adversary acts rationally in the sense of seeking the greatest possible personal advantage leads to the concept of a *Nash equilibrium.* An equilibrium strategy is a steady state of the game in which neither player has an incentive to unilaterally change their plan of actions.

In static games, equilibrium strategies are called Nash equilibria, which is why we refer to the resulting predictive model as *Nash prediction game* (NPG). In a two-player game, a Nash equilibrium is defined as a pair of actions such that no player can benefit from changing their action solely; that is,

$$\begin{aligned}\hat{\theta}_{-1}(\mathbf{w}^*, \dot{D}^*) &= \min_{\mathbf{w}\in\mathcal{W}} \hat{\theta}_{-1}(\mathbf{w}, \dot{D}^*),\\ \hat{\theta}_{+1}(\mathbf{w}^*, \dot{D}^*) &= \min_{\dot{D}\subseteq\mathcal{X}\times\mathcal{Y}} \hat{\theta}_{+1}(\mathbf{w}^*, \dot{D}),\end{aligned}$$

where $\mathcal{W}$ and $\mathcal{X} \times \mathcal{Y}$ denote the players' action spaces.

However, a static prediction game may not have a Nash equilibrium, or it may possess multiple equilibria. If $(\mathbf{w}^*, \dot{D}^*)$ and $(\mathbf{w}', \dot{D}')$ are distinct Nash equilibria and each player decides to act according to a different one of them, then combinations $(\mathbf{w}^*, \dot{D}')$ and $(\mathbf{w}', \dot{D}^*)$ may incur arbitrarily high costs for both players. Hence, one can argue that it is rational for an adversary to play a Nash equilibrium only when the following assumption is satisfied.

Assumption 1 *The following statements hold:*

1. *both players act simultaneously;*

2. *both players have full knowledge about both (empirical) cost functions $\hat{\theta}_v(\mathbf{w}, \dot{D})$ defined in* (1) *and* (2)*, and both action spaces $\mathcal{W}$ and $\mathcal{X} \times \mathcal{Y}$;*

3. *both players act rational with respect to their cost function in the sense of securing their lowest possible costs;*

4. *a unique Nash equilibrium exists.*

Whether Assumptions 1.1-1.3 are adequate—especially the assumption of simultaneous actions—strongly depends on the application. For example, in some applications, the data generator may unilaterally be able to acquire information about the model $f_{\mathbf{w}}$ before committing to $\dot{D}$. Such situations are better modeled as a *Stackelberg competition* (Brückner and Scheffer, 2011). On the other hand, when the learner is able to treat any executed action as part of the training data D and update the model $\mathbf{w}$, the setting is better modeled as repeated executions of a static game with simultaneous actions. The adequateness of Assumption 1.4, which we discuss in the following sections, depends on the chosen loss functions, the cost factors, and the regularizers.

3.1 Existence of a Nash Equilibrium

Theorem 1 of Brückner and Scheffer (2009) identifies conditions under which a unique Nash equilibrium exists. In 2010, Christian Kanzow has located a flaw in the proof of this theorem: The proof argues that the pseudo-Jacobian can be decomposed into two (strictly) positive stable matrices by showing that the real part of every eigenvalue of those two matrices is positive. However, this does not generally imply that the sum of these matrices is positive stable as well since this would require a common Lyapunov solution (*cf.* Problem 2.2.6 of Horn and Johnson 1991). But even if such a solution exists, the positive definiteness cannot be concluded from the positiveness of all eigenvalues as the pseudo-Jacobian is generally non-symmetric.

Having "unproven" prior claims, we will now derive sufficient conditions for the existence of a Nash equilibrium. To this end, we first define

$$\begin{aligned}\mathbf{x} &:= \left[\phi(x_1)^\mathsf{T}, \phi(x_2)^\mathsf{T}, \ldots, \phi(x_n)^\mathsf{T}\right]^\mathsf{T} \in \phi(\mathcal{X})^n \subset \mathbb{R}^{m\cdot n},\\ \dot{\mathbf{x}} &:= \left[\phi(\dot{x}_1)^\mathsf{T}, \phi(\dot{x}_2)^\mathsf{T}, \ldots, \phi(\dot{x}_n)^\mathsf{T}\right]^\mathsf{T} \in \phi(\mathcal{X})^n \subset \mathbb{R}^{m\cdot n},\end{aligned}$$

as long, concatenated, column vectors induced by feature mapping ϕ, training sample $D = \{(x_i, y_i)\}_{i=1}^n$, and transformed training sample $\dot{D} = \{(\dot{x}_i, y_i)\}_{i=1}^n$, respectively. For terminological harmony, we refer to vector $\dot{\mathbf{x}}$ as the data generator's action with corresponding action space $\phi(\mathcal{X})^n$.

We make the following assumptions on the action spaces and the cost functions which enables us to state the main result on the existence of at least one Nash equilibrium in Lemma 1.

Assumption 2 *The players' cost functions defined in Equations 1 and 2, and their action sets $\mathcal{W}$ and $\phi(\mathcal{X})^n$ satisfy the properties:*

1. *loss functions $\ell_v(z, y)$ with $v \in \{-1, +1\}$ are convex and twice continuously differentiable with respect to $z \in \mathbb{R}$ for all fixed $y \in \mathcal{Y}$;*
2. *regularizers $\hat{\Omega}_v$ are uniformly strongly convex and twice continuously differentiable with respect to $\mathbf{w} \in \mathcal{W}$ and $\dot{\mathbf{x}} \in \phi(\mathcal{X})^n$, respectively;*
3. *action spaces $\mathcal{W}$ and $\phi(\mathcal{X})^n$ are nonempty, compact, and convex subsets of finite-dimensional Euclidean spaces $\mathbb{R}^m$ and $\mathbb{R}^{m\cdot n}$, respectively.*

Lemma 1 *Under Assumption 2, at least one equilibrium point $(\mathbf{w}^*, \dot{\mathbf{x}}^*) \in \mathcal{W} \times \phi(\mathcal{X})^n$ of the Nash prediction game defined by*

$$\begin{array}{ll|ll} \min\limits_{\mathbf{w}} & \hat{\theta}_{-1}(\mathbf{w}, \dot{\mathbf{x}}^*) & \min\limits_{\dot{\mathbf{x}}} & \hat{\theta}_{+1}(\mathbf{w}^*, \dot{\mathbf{x}}) \\ \text{s.t.} & \mathbf{w} \in \mathcal{W} & \text{s.t.} & \dot{\mathbf{x}} \in \phi(\mathcal{X})^n \end{array} \quad (3)$$

exists.

Proof. Each player v's cost function is a sum over n loss terms resulting from loss function ℓ_v and regularizer $\hat{\Omega}_v$. By Assumption 2, these loss functions are convex and continuous, and the regularizers are uniformly strongly convex and continuous. Hence, both cost functions $\hat{\theta}_{-1}(\mathbf{w}, \dot{\mathbf{x}})$ and $\hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}})$ are continuous in all arguments and uniformly strongly convex in $\mathbf{w} \in \mathcal{W}$ and $\dot{\mathbf{x}} \in \phi(\mathcal{X})^n$, respectively. As both action spaces $\mathcal{W}$ and $\phi(\mathcal{X})^n$ are nonempty, compact, and convex subsets of finite-dimensional Euclidean spaces, a Nash equilibrium exists—see Theorem 4.3 of Basar and Olsder (1999). ■

3.2 Uniqueness of the Nash Equilibrium

We will now derive conditions for the uniqueness of an equilibrium of the Nash prediction game defined in (3). We first reformulate the two-player game into an $(n+1)$-player game. In Lemma 2 we then present a sufficient condition for the uniqueness of the Nash equilibrium in this game, and by applying Proposition 4 and Lemma 5-7 we verify whether this condition is met. Finally, we state the main result in Theorem 8: The Nash equilibrium is unique under certain properties of the loss functions, the regularizers, and the cost factors which all can be verified easily.

Taking into account the Cartesian product structure of the data generator's action space $\phi(\mathcal{X})^n$, it is not difficult to see that $(\mathbf{w}^*, \dot{\mathbf{x}}^*)$ with $\dot{\mathbf{x}}^* = \left[\dot{\mathbf{x}}_1^{*\mathsf{T}}, \ldots, \dot{\mathbf{x}}_n^{*\mathsf{T}}\right]^\mathsf{T}$ and $\dot{\mathbf{x}}_i^* := \phi(\dot{x}_i^*)$ is a solution of the two-player game if, and only if, $(\mathbf{w}^*, \dot{\mathbf{x}}_1^*, \ldots, \dot{\mathbf{x}}_n^*)$ is a Nash equilibrium of the $(n+1)$-player game defined by

$$\begin{array}{ll} \min\limits_{\mathbf{w}} & \hat{\theta}_{-1}(\mathbf{w}, \dot{\mathbf{x}}) \\ \text{s.t.} & \mathbf{w} \in \mathcal{W} \end{array} \;\Bigg|\; \begin{array}{ll} \min\limits_{\dot{\mathbf{x}}_1} & \hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) \\ \text{s.t.} & \dot{\mathbf{x}}_1 \in \phi(\mathcal{X}) \end{array} \;\Bigg|\; \begin{array}{c} \cdots \\ \cdots \end{array} \;\Bigg|\; \begin{array}{ll} \min\limits_{\dot{\mathbf{x}}_n} & \hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) \\ \text{s.t.} & \dot{\mathbf{x}}_n \in \phi(\mathcal{X}) \end{array}, \qquad (4)$$

which results from (3) by repeating n times the cost function $\hat{\theta}_{+1}$ and minimizing this function with respect to $\dot{\mathbf{x}}_i \in \phi(\mathcal{X})$ for $i = 1, \ldots, n$. Then the *pseudo-gradient* (in the sense of Rosen 1965) of the game in (4) is defined by

$$\mathbf{g_r}(\mathbf{w}, \dot{\mathbf{x}}) := \begin{bmatrix} r_0 \nabla_{\mathbf{w}} \hat{\theta}_{-1}(\mathbf{w}, \dot{\mathbf{x}}) \\ r_1 \nabla_{\dot{\mathbf{x}}_1} \hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) \\ r_2 \nabla_{\dot{\mathbf{x}}_2} \hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) \\ \vdots \\ r_n \nabla_{\dot{\mathbf{x}}_n} \hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) \end{bmatrix} \in \mathbb{R}^{m + m \cdot n}, \qquad (5)$$

with any fixed vector $\mathbf{r} = [r_0, r_1, \ldots, r_n]^\mathsf{T}$ where $r_i > 0$ for $i = 0, \ldots, n$. The derivative of $\mathbf{g_r}$, that is, the *pseudo-Jacobian* of (4), is given by

$$\mathbf{J_r}(\mathbf{w}, \dot{\mathbf{x}}) = \mathbf{\Lambda_r} \begin{bmatrix} \nabla^2_{\mathbf{w},\mathbf{w}} \hat{\theta}_{-1}(\mathbf{w}, \dot{\mathbf{x}}) & \nabla^2_{\mathbf{w},\dot{\mathbf{x}}} \hat{\theta}_{-1}(\mathbf{w}, \dot{\mathbf{x}}) \\ \nabla^2_{\dot{\mathbf{x}},\mathbf{w}} \hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) & \nabla^2_{\dot{\mathbf{x}},\dot{\mathbf{x}}} \hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) \end{bmatrix},$$

where

$$\mathbf{\Lambda}_{\mathbf{r}} := \begin{bmatrix} r_0\mathbf{I}_m & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & r_1\mathbf{I}_m & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & r_n\mathbf{I}_m \end{bmatrix} \in \mathbb{R}^{(m+m\cdot n)\times(m+m\cdot n)}.$$

Notice, that these derivatives, *i.e.,* pseudo-gradient $\mathbf{g}_{\mathbf{r}}$ and pseudo-Jacobian $\mathbf{J}_{\mathbf{r}}$, exist in view of Assumption 2. The above definition of the pseudo-Jacobian enables us to state the following result about the uniqueness of a Nash equilibrium.

Lemma 2 *Let Assumption 2 hold and suppose there exists a fixed vector* $\mathbf{r} = [r_0, r_1, \dots, r_n]^{\mathsf{T}}$ *with* $r_i > 0$ *for all* $i = 0, 1, \dots, n$ *such that the corresponding pseudo-Jacobian* $\mathbf{J}_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}})$ *is positive definite for all* $(\mathbf{w}, \dot{\mathbf{x}}) \in \mathcal{W} \times \phi(\mathcal{X})^n$. *Then the Nash prediction game in* (3) *has a unique equilibrium.*

Proof. The existence of a Nash equilibrium follows from Lemma 1. To prove the uniqueness, recall from our previous discussion, that the original Nash game in (3) has a unique solution if, and only if, the game from (4) with one learner and n data generators admits a unique solution. In view of Theorem 2 of Rosen (1965), the latter attains a unique solution if the pseudo-gradient $\mathbf{g}_{\mathbf{r}}$ is strictly monotone, *i.e.,* for all actions $\mathbf{w}, \mathbf{w}' \in \mathcal{W}$ and $\dot{\mathbf{x}}, \dot{\mathbf{x}}' \in \phi(\mathcal{X})^n$ inequality

$$\left(\mathbf{g}_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}}) - \mathbf{g}_{\mathbf{r}}(\mathbf{w}', \dot{\mathbf{x}}')\right)^{\mathsf{T}} \left(\begin{bmatrix} \mathbf{w} \\ \dot{\mathbf{x}} \end{bmatrix} - \begin{bmatrix} \mathbf{w}' \\ \dot{\mathbf{x}}' \end{bmatrix} \right) > 0$$

holds. A sufficient condition for this pseudo-gradient being strictly monotone is the positive definiteness of the pseudo-Jacobian $\mathbf{J}_{\mathbf{r}}$ (see *e.g.,* Theorem 7.11 in Geiger and Kanzow 1999, Theorem 6 in Rosen 1965). ■

To verify if the condition of Lemma 2 is satisfied, we analyze the pseudo-Jacobian $\mathbf{J}_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}})$. Throughout this section, we denote by $\ell_v'(z, y)$ and $\ell_v''(z, y)$ the first and second derivative of the mapping $\ell_v(z, y)$ with respect to $z \in \mathbb{R}$. A direct calculation shows that the first-order partial derivatives are given by

$$\begin{aligned} \nabla_{\mathbf{w}} \hat{\theta}_{-1}(\mathbf{w}, \dot{\mathbf{x}}) &= \sum_{i=1}^{n} c_{-1,i} \ell_{-1}'(\dot{\mathbf{x}}_i^{\mathsf{T}} \mathbf{w}, y_i) \dot{\mathbf{x}}_i + \rho_{-1} \nabla_{\mathbf{w}} \hat{\Omega}_{-1}(\mathbf{w}), & (6) \\ \nabla_{\dot{\mathbf{x}}_i} \hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) &= c_{+1,i} \ell_{+1}'(\dot{\mathbf{x}}_i^{\mathsf{T}} \mathbf{w}, y_i) \mathbf{w} + \rho_{+1} \nabla_{\dot{\mathbf{x}}_i} \hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}}). & (7) \end{aligned}$$

This allows us to calculate the entries of the pseudo-Jacobian:

$$\begin{aligned} \nabla^2_{\mathbf{w},\mathbf{w}} \hat{\theta}_{-1}(\mathbf{w}, \dot{\mathbf{x}}) &= \sum_{i=1}^{n} c_{-1,i} \ell_{-1}''(\dot{\mathbf{x}}_i^{\mathsf{T}} \mathbf{w}, y_i) \dot{\mathbf{x}}_i \dot{\mathbf{x}}_i^{\mathsf{T}} + \rho_{-1} \nabla^2_{\mathbf{w},\mathbf{w}} \hat{\Omega}_{-1}(\mathbf{w}), \\ \nabla^2_{\mathbf{w},\dot{\mathbf{x}}_i} \hat{\theta}_{-1}(\mathbf{w}, \dot{\mathbf{x}}) &= c_{-1,i} \ell_{-1}''(\dot{\mathbf{x}}_i^{\mathsf{T}} \mathbf{w}, y_i) \dot{\mathbf{x}}_i \mathbf{w}^{\mathsf{T}} + c_{-1,i} \ell_{-1}'(\dot{\mathbf{x}}_i^{\mathsf{T}} \mathbf{w}, y_i) \mathbf{I}_m, \\ \nabla^2_{\dot{\mathbf{x}}_i,\mathbf{w}} \hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) &= c_{+1,i} \ell_{+1}''(\dot{\mathbf{x}}_i^{\mathsf{T}} \mathbf{w}, y_i) \mathbf{w} \dot{\mathbf{x}}_i^{\mathsf{T}} + c_{+1,i} \ell_{+1}'(\dot{\mathbf{x}}_i^{\mathsf{T}} \mathbf{w}, y_i) \mathbf{I}_m, \\ \nabla^2_{\dot{\mathbf{x}}_i,\dot{\mathbf{x}}_j} \hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) &= \begin{cases} c_{+1,i} \ell_{+1}''(\dot{\mathbf{x}}_i^{\mathsf{T}} \mathbf{w}, y_i) \mathbf{w} \mathbf{w}^{\mathsf{T}} + \rho_{+1} \nabla^2_{\dot{\mathbf{x}}_i} \hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}}) & \text{, if } i = j, \\ \rho_{+1} \nabla^2_{\dot{\mathbf{x}}_i,\dot{\mathbf{x}}_j} \hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}}) & \text{, if } i \neq j. \end{cases} \end{aligned}$$

Let us define the matrix

$$\boldsymbol{\Upsilon}(\mathbf{w}, \dot{\mathbf{x}}) := \begin{bmatrix} -\dot{\mathbf{x}}_1^\mathsf{T}- & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & & \vdots \\ -\dot{\mathbf{x}}_n^\mathsf{T}- & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & -\mathbf{w}^\mathsf{T}- & & \mathbf{0} \\ \vdots & & \ddots & \\ \mathbf{0} & \mathbf{0} & & -\mathbf{w}^\mathsf{T}- \end{bmatrix} \in \mathbb{R}^{2n \times (m + m \cdot n)},$$

and let us denote the smallest eigenvalues of the Hessians of the regularizers on the corresponding action spaces $\mathcal{W}$ and $\phi(\mathcal{X})^n$ by

$$\lambda_{-1} := \inf_{\mathbf{w} \in \mathcal{W}} \lambda_{\min}\left(\nabla^2_{\mathbf{w},\mathbf{w}} \hat{\Omega}_{-1}(\mathbf{w})\right), \tag{8}$$

$$\lambda_{+1} := \inf_{\dot{\mathbf{x}} \in \phi(\mathcal{X})^n} \lambda_{\min}\left(\nabla^2_{\dot{\mathbf{x}},\dot{\mathbf{x}}} \hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}})\right), \tag{9}$$

where $\lambda_{\min}(\mathbf{A})$ denotes the smallest eigenvalue of the symmetric matrix $\mathbf{A}$.

Remark 3 Note that the minimum in (8) and (9) is attained and is strictly positive: The mapping $\lambda_{\min} : \mathcal{M}^{k \times k} \to \mathbb{R}$ is concave on the set of symmetric matrices $\mathcal{M}^{k \times k}$ of dimension $k \times k$ (*cf.* Example 3.10 in Boyd and Vandenberghe 2004), and in particular, it therefore follows that this mapping is continuous. Furthermore, the mappings $u_{-1} : \mathcal{W} \to \mathcal{M}^{m \times m}$ with $u_{-1}(\mathbf{w}) := \nabla^2_{\mathbf{w},\mathbf{w}} \hat{\Omega}_{-1}(\mathbf{w})$ and $u_{+1} : \phi(\mathcal{X})^n \to \mathcal{M}^{m \cdot n \times m \cdot n}$ with $u_{+1}(\dot{\mathbf{x}}) := \nabla^2_{\dot{\mathbf{x}},\dot{\mathbf{x}}} \hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}})$ are continuous (for any fixed $\mathbf{x}$) by Assumption 2. Hence, the mappings $\mathbf{w} \mapsto \lambda_{\min}(u_{-1}(\mathbf{w}))$ and $\dot{\mathbf{x}} \mapsto \lambda_{\min}(u_{+1}(\dot{\mathbf{x}}))$ are also continuous since each is precisely the composition $\lambda_{\min} \circ u_v$ of the continuous functions $\lambda_{\min}$ and u_v for $v \in \{-1, +1\}$. Taking into account that a continuous mapping on a nonempty compact set attains its minimum, it follows that there exist elements $\mathbf{w} \in \mathcal{W}$ and $\dot{\mathbf{x}} \in \phi(\mathcal{X})^n$ such that

$$\lambda_{-1} = \lambda_{\min}\left(\nabla^2_{\mathbf{w},\mathbf{w}} \hat{\Omega}_{-1}(\mathbf{w})\right),$$

$$\lambda_{+1} = \lambda_{\min}\left(\nabla^2_{\dot{\mathbf{x}},\dot{\mathbf{x}}} \hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}})\right).$$

Moreover, since the Hessians of the regularizers are positive definite by Assumption 2, we see that $\lambda_v > 0$ holds for $v \in \{-1, +1\}$. ◇

Using the definition of $\boldsymbol{\Upsilon}(\mathbf{w}, \dot{\mathbf{x}})$, λ_v, and the abbreviations

$$\begin{aligned} \ell'_{v,i} &:= \ell'_v(\dot{\mathbf{x}}_i^\mathsf{T}\mathbf{w}, y_i) \quad i = 1, \ldots, n, \\ \ell''_{v,i} &:= \ell''_v(\dot{\mathbf{x}}_i^\mathsf{T}\mathbf{w}, y_i) \quad i = 1, \ldots, n, \\ \boldsymbol{\Gamma}_v &:= \operatorname{diag}(c_{v,1}\ell''_{v,1}, \ldots, c_{v,n}\ell''_{v,n}) \in \mathbb{R}^{n \times n} \end{aligned}$$

for both players $v \in \{-1, +1\}$, we can summarize the previous discussion.

Proposition 4 *The pseudo-Jacobian has the representation*

$$\mathbf{J}_\mathbf{r}(\mathbf{w}, \dot{\mathbf{x}}) = \mathbf{J}_\mathbf{r}^{(1)}(\mathbf{w}, \dot{\mathbf{x}}) + \mathbf{J}_\mathbf{r}^{(2)}(\mathbf{w}, \dot{\mathbf{x}}) + \mathbf{J}_\mathbf{r}^{(3)}(\mathbf{w}, \dot{\mathbf{x}}) \tag{10}$$

where

$$\begin{aligned}
\mathbf{J}_{\mathbf{r}}^{(1)}(\mathbf{w},\dot{\mathbf{x}}) &= \boldsymbol{\Lambda}_{\mathbf{r}}\boldsymbol{\Upsilon}(\mathbf{w},\dot{\mathbf{x}})^{\mathsf{T}}\begin{bmatrix} \boldsymbol{\Gamma}_{-1} & \boldsymbol{\Gamma}_{-1} \\ \boldsymbol{\Gamma}_{+1} & \boldsymbol{\Gamma}_{+1} \end{bmatrix}\boldsymbol{\Upsilon}(\mathbf{w},\dot{\mathbf{x}}), \\
\mathbf{J}_{\mathbf{r}}^{(2)}(\mathbf{w},\dot{\mathbf{x}}) &= \boldsymbol{\Lambda}_{\mathbf{r}}\begin{bmatrix} \rho_{-1}\lambda_{-1}\mathbf{I}_m & c_{-1,1}\ell'_{-1,1}\mathbf{I}_m & \cdots & c_{-1,n}\ell'_{-1,n}\mathbf{I}_m \\ c_{+1,1}\ell'_{+1,1}\mathbf{I}_m & \rho_{+1}\lambda_{+1}\mathbf{I}_m & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ c_{+1,n}\ell'_{+1,n}\mathbf{I}_m & \mathbf{0} & \cdots & \rho_{+1}\lambda_{+1}\mathbf{I}_m \end{bmatrix}, \\
\mathbf{J}_{\mathbf{r}}^{(3)}(\mathbf{w},\dot{\mathbf{x}}) &= \boldsymbol{\Lambda}_{\mathbf{r}}\begin{bmatrix} \rho_{-1}\nabla^2_{\mathbf{w},\mathbf{w}}\hat{\Omega}_{-1}(\mathbf{w}) - \rho_{-1}\lambda_{-1}\mathbf{I}_m & \mathbf{0} \\ \mathbf{0} & \rho_{+1}\nabla^2_{\dot{\mathbf{x}},\dot{\mathbf{x}}}\hat{\Omega}_{+1}(\mathbf{x},\dot{\mathbf{x}}) - \rho_{+1}\lambda_{+1}\mathbf{I}_{m\cdot n} \end{bmatrix}.
\end{aligned}$$

Recall, that we want to investigate whether there is some fixed positive vector $\mathbf{r}$ such that $\mathbf{J}_{\mathbf{r}}(\mathbf{w},\dot{\mathbf{x}})$ is positive definite for each pair of actions $(\mathbf{w},\dot{\mathbf{x}}) \in \mathcal{W} \times \phi(\mathcal{X})^n$. A sufficient condition for the pseudo-Jacobian $\mathbf{J}_{\mathbf{r}}(\mathbf{w},\dot{\mathbf{x}})$ to be positive definite is that $\mathbf{J}_{\mathbf{r}}^{(1)}(\mathbf{w},\dot{\mathbf{x}})$, $\mathbf{J}_{\mathbf{r}}^{(2)}(\mathbf{w},\dot{\mathbf{x}})$, and $\mathbf{J}_{\mathbf{r}}^{(3)}(\mathbf{w},\dot{\mathbf{x}})$ are positive semi-definite and at least one of these matrices is positive definite. Before discussing these matrices separately, let us define $r_0 := 1$, $r_i := \frac{c_{-1,i}}{c_{+1,i}} > 0$ for all $i = 1,\ldots,n$, with corresponding matrix

$$\boldsymbol{\Lambda}_{\mathbf{r}} := \begin{bmatrix} \mathbf{I}_m & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \frac{c_{-1,1}}{c_{+1,1}}\mathbf{I}_m & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \frac{c_{-1,n}}{c_{+1,n}}\mathbf{I}_m \end{bmatrix}, \tag{11}$$

and let us make the following assumption on the loss functions ℓ_v and the regularizers $\hat{\Omega}_v$ for $v \in \{-1,+1\}$. Instances of these functions satisfying Assumptions 2 and 3 will be given in Section 5.

Assumption 3 *For all $\mathbf{w} \in \mathcal{W}$ and $\dot{\mathbf{x}} \in \phi(\mathcal{X})^n$ with $\dot{\mathbf{x}} = \left[\dot{\mathbf{x}}_1^{\mathsf{T}},\ldots,\dot{\mathbf{x}}_n^{\mathsf{T}}\right]^{\mathsf{T}}$ the following conditions are satisfied:*

1. *the second derivatives of the loss functions are equal for all $y \in \mathcal{Y}$ and $i = 1,\ldots,n$,*

$$\ell''_{-1}(f_{\mathbf{w}}(\dot{\mathbf{x}}_i), y) = \ell''_{+1}(f_{\mathbf{w}}(\dot{\mathbf{x}}_i), y),$$

2. *the players' regularization parameters satisfy*

$$\rho_{-1}\rho_{+1} > \tau^2 \frac{1}{\lambda_{-1}\lambda_{+1}}\mathbf{c}_{-1}^{\mathsf{T}}\mathbf{c}_{+1},$$

where λ_{-1}, λ_{+1} are the smallest eigenvalues of the Hessians of the regularizers specified in (8) *and* (9)*, $\mathbf{c}_v = [c_{v,1}, c_{v,2}, \ldots, c_{v,n}]^{\mathsf{T}}$, and*

$$\tau = \sup_{(\mathbf{x},y)\in\phi(\mathcal{X})\times\mathcal{Y}} \frac{1}{2}\left|\ell'_{-1}(f_{\mathbf{w}}(\mathbf{x}), y) + \ell'_{+1}(f_{\mathbf{w}}(\mathbf{x}), y)\right|, \tag{12}$$

3. *for all $i = 1, \ldots, n$ either both players have equal instance-specific cost factors, $c_{-1,i} = c_{+1,i}$, or the partial derivative $\nabla_{\dot{\mathbf{x}}_i} \Omega_{+1}(\mathbf{x}, \dot{\mathbf{x}})$ of the* data generator's *regularizer is independent of $\dot{\mathbf{x}}_j$ for all $j \neq i$.*

Notice, that τ in Equation 12 can be chosen finite as the set $\phi(\mathcal{X}) \times \mathcal{Y}$ is assumed to be compact, and consequently, the values of both continuous mappings $\ell'_{-1}(f_{\mathbf{w}}(\mathbf{x}), y)$ and $\ell'_{+1}(f_{\mathbf{w}}(\mathbf{x}), y)$ are finite for all $(\mathbf{x}, y) \in \phi(\mathcal{X}) \times \mathcal{Y}$.

Lemma 5 *Let $(\mathbf{w}, \dot{\mathbf{x}}) \in \mathcal{W} \times \phi(\mathcal{X})^n$ be arbitrarily given. Under Assumptions 2 and 3, the matrix $\mathbf{J}_{\mathbf{r}}^{(1)}(\mathbf{w}, \dot{\mathbf{x}})$ is symmetric positive semi-definite (but not positive definite) for $\mathbf{\Lambda}_{\mathbf{r}}$ defined as in Equation 11.*

Proof. The special structure of $\mathbf{\Lambda}_{\mathbf{r}}$ and $\mathbf{\Upsilon}(\mathbf{w}, \dot{\mathbf{x}})$ gives

$$\mathbf{J}_{\mathbf{r}}^{(1)}(\mathbf{w}, \dot{\mathbf{x}}) = \mathbf{\Upsilon}(\mathbf{w}, \dot{\mathbf{x}})^{\mathsf{T}} \begin{bmatrix} r_0 \mathbf{\Gamma}_{-1} & r_0 \mathbf{\Gamma}_{-1} \\ \operatorname{diag}(r_1, \ldots, r_n) \mathbf{\Gamma}_{+1} & \operatorname{diag}(r_1, \ldots, r_n) \mathbf{\Gamma}_{+1} \end{bmatrix} \mathbf{\Upsilon}(\mathbf{w}, \dot{\mathbf{x}}).$$

From the assumption $\ell''_{-1,i} = \ell''_{+1,i}$ and the definition $r_0 = 1$, $r_i = \frac{c_{-1,i}}{c_{+1,i}} > 0$ for all $i = 1, \ldots, n$ it follows that $\mathbf{\Gamma}_{-1} = \operatorname{diag}(r_1, \ldots, r_n) \mathbf{\Gamma}_{+1}$, such that

$$\mathbf{J}_{\mathbf{r}}^{(1)}(\mathbf{w}, \dot{\mathbf{x}}) = \mathbf{\Upsilon}(\mathbf{w}, \dot{\mathbf{x}})^{\mathsf{T}} \begin{bmatrix} \mathbf{\Gamma}_{-1} & \mathbf{\Gamma}_{-1} \\ \mathbf{\Gamma}_{-1} & \mathbf{\Gamma}_{-1} \end{bmatrix} \mathbf{\Upsilon}(\mathbf{w}, \dot{\mathbf{x}}),$$

which is obviously a symmetric matrix. Furthermore, we show that $\mathbf{z}^{\mathsf{T}} \mathbf{J}_{\mathbf{r}}^{(1)}(\mathbf{w}, \dot{\mathbf{x}}) \mathbf{z} \geq 0$ holds for all vectors $\mathbf{z} \in \mathbb{R}^{m + m \cdot n}$. To this end, let $\mathbf{z}$ be arbitrarily given, and partition this vector in $\mathbf{z} = \left[\mathbf{z}_0^{\mathsf{T}}, \mathbf{z}_1^{\mathsf{T}}, \ldots, \mathbf{z}_n^{\mathsf{T}}\right]^{\mathsf{T}}$ with $\mathbf{z}_i \in \mathbb{R}^m$ for all $i = 0, 1, \ldots, n$. Then a simple calculation shows that

$$\mathbf{z}^{\mathsf{T}} \mathbf{J}_{\mathbf{r}}^{(1)}(\mathbf{w}, \dot{\mathbf{x}}) \mathbf{z} = \sum_{i=1}^{n} \left(\mathbf{z}_0^{\mathsf{T}} \dot{\mathbf{x}}_i + \mathbf{z}_i^{\mathsf{T}} \mathbf{w} \right)^2 c_{-1,i} \ell''_{-1,i} \geq 0$$

since $\ell''_{-1,i} \geq 0$ for all $i = 1, \ldots, n$ in view of the assumed convexity of mapping $\ell_{-1}(z, y)$. Hence, $\mathbf{J}_{\mathbf{r}}^{(1)}(\mathbf{w}, \dot{\mathbf{x}})$ is positive semi-definite. This matrix cannot be positive definite since we have $\mathbf{z}^{\mathsf{T}} \mathbf{J}_{\mathbf{r}}^{(1)}(\mathbf{w}, \dot{\mathbf{x}}) \mathbf{z} = 0$ for the particular vector $\mathbf{z}$ defined by $\mathbf{z}_0 := -\mathbf{w}$ and $\mathbf{z}_i := \dot{\mathbf{x}}_i$ for all $i = 1, \ldots, n$. ■

Lemma 6 *Let $(\mathbf{w}, \dot{\mathbf{x}}) \in \mathcal{W} \times \phi(\mathcal{X})^n$ be arbitrarily given. Under Assumptions 2 and 3, the matrix $\mathbf{J}_{\mathbf{r}}^{(2)}(\mathbf{w}, \dot{\mathbf{x}})$ is positive definite for $\mathbf{\Lambda}_{\mathbf{r}}$ defined as in Equation 11.*

Proof. A sufficient and necessary condition for the (possibly asymmetric) matrix $\mathbf{J}_{\mathbf{r}}^{(2)}(\mathbf{w}, \dot{\mathbf{x}})$ to be positive definite is that the Hermitian matrix

$$\mathbf{H}(\mathbf{w}, \dot{\mathbf{x}}) := \mathbf{J}_{\mathbf{r}}^{(2)}(\mathbf{w}, \dot{\mathbf{x}}) + \mathbf{J}_{\mathbf{r}}^{(2)}(\mathbf{w}, \dot{\mathbf{x}})^{\mathsf{T}}$$

is positive definite, that is, all eigenvalues of $\mathbf{H}(\mathbf{w}, \dot{\mathbf{x}})$ are positive. Let $\mathbf{\Lambda}_{\mathbf{r}}^{\frac{1}{2}}$ denote the square root of $\mathbf{\Lambda}_{\mathbf{r}}$ which is defined in such a way that the diagonal elements of $\mathbf{\Lambda}_{\mathbf{r}}^{\frac{1}{2}}$ are the square roots of the corresponding diagonal elements of $\mathbf{\Lambda_r}$. Furthermore, we denote by $\mathbf{\Lambda_r}^{-\frac{1}{2}}$ the inverse of $\mathbf{\Lambda_r^{\frac{1}{2}}}$. Then, by Sylvester's law of inertia, the matrix

$$\bar{\mathbf{H}}(\mathbf{w}, \dot{\mathbf{x}}) := \mathbf{\Lambda_r}^{-\frac{1}{2}} \mathbf{H}(\mathbf{w}, \dot{\mathbf{x}}) \mathbf{\Lambda_r}^{-\frac{1}{2}}$$

has the same number of positive, zero, and negative eigenvalues as matrix $\mathbf{H}(\mathbf{w}, \dot{\mathbf{x}})$ itself. Hence, $\mathbf{J_r^{(2)}}(\mathbf{w}, \dot{\mathbf{x}})$ is positive definite if, and only if, all eigenvalues of

$$\begin{aligned}
\bar{\mathbf{H}}(\mathbf{w}, \dot{\mathbf{x}}) &= \mathbf{\Lambda_r}^{-\frac{1}{2}} \left(\mathbf{J_r^{(2)}}(\mathbf{w}, \dot{\mathbf{x}}) + \mathbf{J_r^{(2)}}(\mathbf{w}, \dot{\mathbf{x}})^\mathsf{T} \right) \mathbf{\Lambda_r}^{-\frac{1}{2}} \\
&= \mathbf{\Lambda_r}^{-\frac{1}{2}} \mathbf{\Lambda_r} \begin{bmatrix} \rho_{-1}\lambda_{-1}\mathbf{I}_m & c_{-1,1}\ell'_{-1,1}\mathbf{I}_m & \cdots & c_{-1,n}\ell'_{-1,n}\mathbf{I}_m \\ c_{+1,1}\ell'_{+1,1}\mathbf{I}_m & \rho_{+1}\lambda_{+1}\mathbf{I}_m & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ c_{+1,n}\ell'_{+1,n}\mathbf{I}_m & \mathbf{0} & \cdots & \rho_{+1}\lambda_{+1}\mathbf{I}_m \end{bmatrix} \mathbf{\Lambda_r}^{-\frac{1}{2}} + \\
&\quad \mathbf{\Lambda_r}^{-\frac{1}{2}} \begin{bmatrix} \rho_{-1}\lambda_{-1}\mathbf{I}_m & c_{+1,1}\ell'_{+1,1}\mathbf{I}_m & \cdots & c_{+1,n}\ell'_{+1,n}\mathbf{I}_m \\ c_{-1,1}\ell'_{-1,1}\mathbf{I}_m & \rho_{+1}\lambda_{+1}\mathbf{I}_m & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ c_{-1,n}\ell'_{-1,n}\mathbf{I}_m & \mathbf{0} & \cdots & \rho_{+1}\lambda_{+1}\mathbf{I}_m \end{bmatrix} \mathbf{\Lambda_r} \mathbf{\Lambda_r}^{-\frac{1}{2}} \\
&= \begin{bmatrix} 2\rho_{-1}\lambda_{-1}\mathbf{I}_m & \tilde{c}_1\mathbf{I}_m & \cdots & \tilde{c}_n\mathbf{I}_m \\ \tilde{c}_1\mathbf{I}_m & 2\rho_{+1}\lambda_{+1}\mathbf{I}_m & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{c}_n\mathbf{I}_m & \mathbf{0} & \cdots & 2\rho_{+1}\lambda_{+1}\mathbf{I}_m \end{bmatrix}
\end{aligned}$$

are positive, where $\tilde{c}_i := \sqrt{c_{-1,i}c_{+1,i}}(\ell'_{-1,i} + \ell'_{+1,i})$. Each eigenvalue λ of this matrix satisfies

$$\left(\bar{\mathbf{H}}(\mathbf{w}, \dot{\mathbf{x}}) - \lambda \mathbf{I}_{m+m\cdot n} \right) \mathbf{v} = \mathbf{0}$$

for the corresponding eigenvector $\mathbf{v}^\mathsf{T} = \left[\mathbf{v}_0^\mathsf{T}, \mathbf{v}_1^\mathsf{T}, \dots, \mathbf{v}_n^\mathsf{T} \right]$ with $\mathbf{v}_i \in \mathbb{R}^m$ for $i = 0, 1, \dots, n$. This eigenvalue equation can be rewritten block-wise as

$$\begin{aligned}
(2\rho_{-1}\lambda_{-1} - \lambda)\mathbf{v}_0 + \sum_{i=1}^{n} \tilde{c}_i \mathbf{v}_i &= \mathbf{0}, \qquad (13) \\
(2\rho_{+1}\lambda_{+1} - \lambda)\mathbf{v}_i + \tilde{c}_i \mathbf{v}_0 &= \mathbf{0} \quad \forall\, i = 1, \dots, n. \qquad (14)
\end{aligned}$$

To compute all possible eigenvalues, we consider two cases. First, assume that $\mathbf{v}_0 = \mathbf{0}$. Then (13) and (14) reduce to

$$\sum_{i=1}^{n} \tilde{c}_i \mathbf{v}_i = \mathbf{0} \quad \text{and} \quad (2\rho_{+1}\lambda_{+1} - \lambda)\mathbf{v}_i = \mathbf{0} \quad \forall\, i = 1, \dots, n.$$

Since $\mathbf{v}_0 = \mathbf{0}$ and eigenvector $\mathbf{v} \neq \mathbf{0}$, at least one $\mathbf{v}_i$ is nonzero. This implies that $\lambda = 2\rho_{+1}\lambda_{+1}$ is an eigenvalue. Using the fact that the null space of the linear mapping $\mathbf{v} \mapsto \sum_{i=1}^{n} \tilde{c}_i \mathbf{v}_i$ has dimension $(n-1)\cdot m$ (we have $n\cdot m$ degrees of freedom counting all components of $\mathbf{v}_1, \dots, \mathbf{v}_n$ and m equations in $\sum_{i=1}^{n} \tilde{c}_i \mathbf{v}_i = \mathbf{0}$), it follows that $\lambda = 2\rho_{+1}\lambda_{+1}$ is an eigenvalue of multiplicity $(n-1)\cdot m$.

Now we consider the second case where $\mathbf{v}_0 \neq \mathbf{0}$. We may further assume that $\lambda \neq 2\rho_{+1}\lambda_{+1}$ (since otherwise we get the same eigenvalue as before, just with a different multiplicity).

We then get from (14) that

$$\mathbf{v}_i = -\frac{\tilde{c}_i}{2\rho_{+1}\lambda_{+1} - \lambda}\mathbf{v}_0 \quad \forall\, i = 1, \ldots, n, \tag{15}$$

and when substituting this expression into (13), we obtain

$$\left((2\rho_{-1}\lambda_{-1} - \lambda) - \sum_{i=1}^{n} \frac{\tilde{c}_i^2}{2\rho_{+1}\lambda_{+1} - \lambda} \right) \mathbf{v}_0 = \mathbf{0}.$$

Taking into account that $\mathbf{v}_0 \neq \mathbf{0}$, this implies

$$0 = 2\rho_{-1}\lambda_{-1} - \lambda - \frac{1}{2\rho_{+1}\lambda_{+1} - \lambda} \sum_{i=1}^{n} \tilde{c}_i^2$$

and, therefore,

$$0 = \lambda^2 - 2(\rho_{-1}\lambda_{-1} + \rho_{+1}\lambda_{+1})\lambda + 4\rho_{-1}\rho_{+1}\lambda_{-1}\lambda_{+1} - \sum_{i=1}^{n} \tilde{c}_i^2.$$

The roots of this quadratic equation are

$$\lambda = \rho_{-1}\lambda_{-1} + \rho_{+1}\lambda_{+1} \pm \sqrt{(\rho_{-1}\lambda_{-1} - \rho_{+1}\lambda_{+1})^2 + \sum_{i=1}^{n} \tilde{c}_i^2}, \tag{16}$$

and these are the remaining eigenvalues of $\bar{\mathbf{H}}(\mathbf{w}, \dot{\mathbf{x}})$, each of multiplicity m since there are precisely m linearly independent vectors $\mathbf{v}_0 \neq \mathbf{0}$ whereas the other vectors $\mathbf{v}_i$ $(i = 1, \ldots, n)$ are uniquely defined by (15) in this case. In particular, this implies that the dimensions of all three eigenspaces together is $(n-1)m + m + m = (n+1)m$, hence other eigenvalues cannot exist. Since the eigenvalue $\lambda = 2\rho_{+1}\lambda_{+1}$ is positive by Remark 3, it remains to show that the roots in (16) are positive as well. By Assumption 3, we have

$$\sum_{i=1}^{n} \tilde{c}_i^2 = \sum_{i=1}^{n} c_{-1,i}c_{+1,i}(\ell'_{-1,i} + \ell'_{+1,i})^2 \leq 4\tau^2 \mathbf{c}_{-1}^{\mathsf{T}}\mathbf{c}_{+1} < 4\rho_{-1}\rho_{+1}\lambda_{-1}\lambda_{+1},$$

where $\mathbf{c}_v = [c_{v,1}, c_{v,2}, \cdots, c_{v,n}]^{\mathsf{T}}$. This inequality and Equation 16 give

$$\begin{aligned} \lambda &= \rho_{-1}\lambda_{-1} + \rho_{+1}\lambda_{+1} \pm \sqrt{(\rho_{-1}\lambda_{-1} - \rho_{+1}\lambda_{+1})^2 + \sum_{i=1}^{n} \tilde{c}_i^2} \\ &> \rho_{-1}\lambda_{-1} + \rho_{+1}\lambda_{+1} - \sqrt{(\rho_{-1}\lambda_{-1} - \rho_{+1}\lambda_{+1})^2 + 4\rho_{-1}\rho_{+1}\lambda_{-1}\lambda_{+1}} \;=\; 0. \end{aligned}$$

As all eigenvalues of $\bar{\mathbf{H}}(\mathbf{w}, \dot{\mathbf{x}})$ are positive, matrix $\mathbf{H}(\mathbf{w}, \dot{\mathbf{x}})$ and, consequently, also the matrix $\mathbf{J}_{\mathbf{r}}^{(2)}(\mathbf{w}, \dot{\mathbf{x}})$ are positive definite. ■

Lemma 7 *Let* $(\mathbf{w}, \dot{\mathbf{x}}) \in \mathcal{W} \times \phi(\mathcal{X})^n$ *be arbitrarily given. Under Assumptions 2 and 3, the matrix* $\mathbf{J}_{\mathbf{r}}^{(3)}(\mathbf{w}, \dot{\mathbf{x}})$ *is positive semi-definite for* $\mathbf{\Lambda}_{\mathbf{r}}$ *defined as in Equation 11.*

Proof. By Assumption 3, either both players have equal instance-specific costs, or the partial gradient $\nabla_{\dot{\mathbf{x}}_i}\hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}})$ of the *sender's* regularizer is independent of $\dot{\mathbf{x}}_j$ for all $j \neq i$ and $i = 1, \ldots, n$. Let us consider the first case where $c_{-1,i} = c_{+1,i}$, and consequently $r_i = 1$, for all $i = 1, \ldots, n$, such that

$$\mathbf{J}_{\mathbf{r}}^{(3)}(\mathbf{w}, \dot{\mathbf{x}}) = \begin{bmatrix} \rho_{-1}\nabla^2_{\mathbf{w},\mathbf{w}}\hat{\Omega}_{-1}(\mathbf{w}) - \rho_{-1}\lambda_{-1}\mathbf{I}_m & \mathbf{0} \\ \mathbf{0} & \rho_{+1}\nabla^2_{\dot{\mathbf{x}},\dot{\mathbf{x}}}\hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}}) - \rho_{+1}\lambda_{+1}\mathbf{I}_{m \cdot n} \end{bmatrix}.$$

The eigenvalues of this block diagonal matrix are the eigenvalues of the matrix $\rho_{-1}(\nabla^2_{\mathbf{w},\mathbf{w}}\hat{\Omega}_{-1}(\mathbf{w}) - \lambda_{-1}\mathbf{I}_m)$ together with those of $\rho_{+1}(\nabla^2_{\dot{\mathbf{x}},\dot{\mathbf{x}}}\hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}}) - \lambda_{+1}\mathbf{I}_{m \cdot n})$. From the definition of λ_v in (8) and (9) follows that these matrices are positive semi-definite for $v \in \{-1, +1\}$. Hence, $\mathbf{J}_{\mathbf{r}}^{(3)}(\mathbf{w}, \dot{\mathbf{x}})$ is positive semi-definite as well.

Now, let us consider the second case where we assume that $\nabla_{\dot{\mathbf{x}}_i}\hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}})$ is independent of $\dot{\mathbf{x}}_j$ for all $j \neq i$. Hence, $\nabla^2_{\dot{\mathbf{x}}_i,\dot{\mathbf{x}}_j}\hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}}) = \mathbf{0}$ for all $j \neq i$ such that

$$\mathbf{J}_{\mathbf{r}}^{(3)}(\mathbf{w}, \dot{\mathbf{x}}) = \begin{bmatrix} \rho_{-1}\tilde{\Omega}_{-1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \rho_{+1}\frac{c_{-1,1}}{c_{+1,1}}\tilde{\Omega}_{+1,1} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \rho_{+1}\frac{c_{-1,n}}{c_{+1,n}}\tilde{\Omega}_{+1,n} \end{bmatrix},$$

where $\tilde{\Omega}_{-1} := \nabla^2_{\mathbf{w},\mathbf{w}}\hat{\Omega}_{-1}(\mathbf{w}) - \lambda_{-1}\mathbf{I}_m$ and $\tilde{\Omega}_{+1,i} = \nabla^2_{\dot{\mathbf{x}}_i,\dot{\mathbf{x}}_i}\hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}}) - \lambda_{+1}\mathbf{I}_m$. The eigenvalues of this block diagonal matrix are again the union of the eigenvalues of the single blocks $\rho_{-1}\tilde{\Omega}_{-1}$ and $\rho_{+1}\frac{c_{-1,i}}{c_{+1,i}}\tilde{\Omega}_{+1,i}$ for $i = 1, \ldots, n$. As in the first part of the proof, $\tilde{\Omega}_{-1}$ is positive semi-definite. The eigenvalues of $\nabla^2_{\dot{\mathbf{x}},\dot{\mathbf{x}}}\hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}})$ are the union of all eigenvalues of $\nabla^2_{\dot{\mathbf{x}}_i,\dot{\mathbf{x}}_i}\hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}})$. Hence, each of these eigenvalues is larger or equal to λ_{+1} and thus, each block $\tilde{\Omega}_{+1,i}$ is positive semi-definite. The factors $\rho_{-1} > 0$ and $\rho_{+1}\frac{c_{-1,i}}{c_{+1,i}} > 0$ are multipliers that do not affect the definiteness of the blocks, and consequently, $\mathbf{J}_{\mathbf{r}}^{(3)}(\mathbf{w}, \dot{\mathbf{x}})$ is positive semi-definite as well. ■

The previous results guarantee the existence and uniqueness of a Nash equilibrium under the stated assumptions.

Theorem 8 *Let Assumptions 2 and 3 hold. Then the Nash prediction game in* (3) *has a unique equilibrium.*

Proof. The existence of an equilibrium of the Nash prediction game in (3) follows from Lemma 1. Proposition 4 and Lemma 5 to 7 imply that there is a positive diagonal matrix $\mathbf{\Lambda}_{\mathbf{r}}$ such that $\mathbf{J}_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}})$ is positive definite for all $(\mathbf{w}, \dot{\mathbf{x}}) \in \mathcal{W} \times \phi(\mathcal{X})^n$. Hence, the uniqueness follows from Lemma 2. ■

4. Finding the Unique Nash Equilibrium

According to Theorem 8, a unique equilibrium of the Nash prediction game in (3) exists for suitable loss functions and regularizers. To find this equilibrium, we derive and study two distinct methods: The first is based on the Nikaido-Isoda function that is constructed such that a minimax solution of this function is an equilibrium of the Nash prediction game and vice versa. This problem is then solved by inexact linesearch. In the second approach, we reformulate the Nash prediction game into a variational inequality problem which is solved by a modified extragradient method.

The data generator's action of transforming the input distribution manifests in a concatenation of transformed training instances $\dot{\mathbf{x}} \in \phi(\mathcal{X})^n$ mapped into the feature space $\dot{\mathbf{x}}_i := \phi(\dot{x}_i)$ for $i = 1, \ldots, n$, and the learner's action is to choose weight vector $\mathbf{w} \in \mathcal{W}$ of classifier $h(x) = \operatorname{sign} f_{\mathbf{w}}(x)$ with linear decision function $f_{\mathbf{w}}(x) = \mathbf{w}^{\mathsf{T}}\phi(x)$.

4.1 An Inexact Linesearch Approach

To solve for a Nash equilibrium, we again consider the game from (4) with one learner and n data generators. A solution of this game can be identified with the help of the weighted *Nikaido-Isoda* function (Equation 17). For any two combinations of actions $(\mathbf{w}, \dot{\mathbf{x}}) \in \mathcal{W} \times \phi(\mathcal{X})^n$ and $(\mathbf{w}', \dot{\mathbf{x}}') \in \mathcal{W} \times \phi(\mathcal{X})^n$ with $\dot{\mathbf{x}} = \left[\dot{\mathbf{x}}_1^{\mathsf{T}}, \ldots, \dot{\mathbf{x}}_n^{\mathsf{T}}\right]^{\mathsf{T}}$ and $\dot{\mathbf{x}}' = \left[\dot{\mathbf{x}}_1'^{\mathsf{T}}, \ldots, \dot{\mathbf{x}}_n'^{\mathsf{T}}\right]^{\mathsf{T}}$, this function is the weighted sum of relative cost savings that the $n+1$ players can enjoy by changing from strategy $\mathbf{w}$ to $\mathbf{w}'$ and $\dot{\mathbf{x}}_i$ to $\dot{\mathbf{x}}_i'$, respectively, while the other players continue to play according to $(\mathbf{w}, \dot{\mathbf{x}})$, that is,

$$\vartheta_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}}, \mathbf{w}', \dot{\mathbf{x}}') := r_0 \left(\hat{\theta}_{-1}(\mathbf{w}, \dot{\mathbf{x}}) - \hat{\theta}_{-1}(\mathbf{w}', \dot{\mathbf{x}})\right) + \sum_{i=1}^{n} r_i \left(\hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) - \hat{\theta}_{+1}(\mathbf{w}, \dot{\mathbf{x}}^{(i)})\right), \quad (17)$$

where $\dot{\mathbf{x}}^{(i)} := \left[\dot{\mathbf{x}}_1^{\mathsf{T}}, \ldots, \dot{\mathbf{x}}_i'^{\mathsf{T}}, \ldots, \dot{\mathbf{x}}_n^{\mathsf{T}}\right]^{\mathsf{T}}$. Let us denote the weighted sum of greatest possible cost savings with respect to any given combination of actions $(\mathbf{w}, \dot{\mathbf{x}}) \in \mathcal{W} \times \phi(\mathcal{X})^n$ by

$$\bar{\vartheta}_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}}) := \max_{(\mathbf{w}', \dot{\mathbf{x}}') \in \mathcal{W} \times \phi(\mathcal{X})^n} \vartheta_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}}, \mathbf{w}', \dot{\mathbf{x}}'), \quad (18)$$

where $\bar{\mathbf{w}}(\mathbf{w}, \dot{\mathbf{x}})$, $\bar{\mathbf{x}}(\mathbf{w}, \dot{\mathbf{x}})$ denotes the corresponding pair of maximizers. Notice, that the maximum in (18) is attained for any $(\mathbf{w}, \dot{\mathbf{x}})$, since $\mathcal{W} \times \phi(\mathcal{X})^n$ is assumed to be compact and $\vartheta_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}}, \mathbf{w}', \dot{\mathbf{x}}')$ is continuous in $(\mathbf{w}', \dot{\mathbf{x}}')$.

By these definitions, a combination $(\mathbf{w}^*, \dot{\mathbf{x}}^*)$ is an equilibrium of the Nash prediction game if, and only if, $\bar{\vartheta}_{\mathbf{r}}(\mathbf{w}^*, \dot{\mathbf{x}}^*)$ is a global minimum of mapping $\bar{\vartheta}_{\mathbf{r}}$ with $\bar{\vartheta}_{\mathbf{r}}(\mathbf{w}^*, \dot{\mathbf{x}}^*) = 0$ for any fixed weights $r_i > 0$ and $i = 0, \ldots, n$, *cf.* Proposition 2.1(b) of von Heusinger and Kanzow (2009). Equivalently, a Nash equilibrium simultaneously satisfies both equations $\bar{\mathbf{w}}(\mathbf{w}^*, \dot{\mathbf{x}}^*) = \mathbf{w}^*$ and $\bar{\mathbf{x}}(\mathbf{w}^*, \dot{\mathbf{x}}^*) = \dot{\mathbf{x}}^*$.

The significance of this observation is that the equilibrium problem in (3) can be reformulated into a minimization problem of the continuous mapping $\bar{\vartheta}_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}})$. To solve this minimization problem, we make use of Corollary 3.4 of von Heusinger and Kanzow (2009). We set the weights $r_0 := 1$ and $r_i := \frac{c_{-1,i}}{c_{+1,i}}$ for all $i = 1, \ldots, n$ as in (11), which ensures the main condition of Corollary 3.4, that is, the positive definiteness of the Jacobian $\mathbf{J}_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}})$

in (10) (*cf.* proof of Theorem 8). According to this corollary, vectors

$$\mathbf{d}_{-1}(\mathbf{w}, \dot{\mathbf{x}}) := \bar{\mathbf{w}}(\mathbf{w}, \dot{\mathbf{x}}) - \mathbf{w} \quad \text{and} \quad \mathbf{d}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) := \bar{\mathbf{x}}(\mathbf{w}, \dot{\mathbf{x}}) - \dot{\mathbf{x}}$$

form a descent direction $\mathbf{d}(\mathbf{w}, \dot{\mathbf{x}}) := [\mathbf{d}_{-1}(\mathbf{w}, \dot{\mathbf{x}})^{\mathsf{T}}, \mathbf{d}_{+1}(\mathbf{w}, \dot{\mathbf{x}})^{\mathsf{T}}]^{\mathsf{T}}$ of $\bar{\vartheta}_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}})$ at any position $(\mathbf{w}, \dot{\mathbf{x}}) \in \mathcal{W} \times \phi(\mathcal{X})^n$ (except for the Nash equilibrium where $\mathbf{d}(\mathbf{w}^*, \dot{\mathbf{x}}^*) = \mathbf{0}$), and consequently, there exists $t \in [0, 1]$ such that

$$\bar{\vartheta}_{\mathbf{r}}(\mathbf{w} + t\mathbf{d}_{-1}(\mathbf{w}, \dot{\mathbf{x}}), \dot{\mathbf{x}} + t\mathbf{d}_{+1}(\mathbf{w}, \dot{\mathbf{x}})) < \bar{\vartheta}_{\mathbf{r}}(\mathbf{w}, \dot{\mathbf{x}}).$$

Since, $(\mathbf{w}, \dot{\mathbf{x}})$ and $(\bar{\mathbf{w}}(\mathbf{w}, \dot{\mathbf{x}}), \bar{\mathbf{w}}(\mathbf{w}, \dot{\mathbf{x}}))$ are feasible combinations of actions, the convexity of the action spaces ensures that $(\mathbf{w} + t\mathbf{d}_{-1}(\mathbf{w}, \dot{\mathbf{x}}), \dot{\mathbf{x}} + t\mathbf{d}_{+1}(\mathbf{w}, \dot{\mathbf{x}}))$ is a feasible combination for any $t \in [0, 1]$ as well. The following algorithm exploits these properties.

Algorithm 1 ILS: Inexact Linesearch Solver for Nash Prediction Games

```
Require: Cost functions $\hat{\theta}_v$ as defined in (1) and (2), and action spaces $\mathcal{W}$ and $\phi(\mathcal{X})^n$.
 1: Select initial $\mathbf{w}^{(0)} \in \mathcal{W}$, set $\dot{\mathbf{x}}^{(0)} := \mathbf{x}$, set $k := 0$, and select $\sigma \in (0,1)$ and $\beta \in (0,1)$.
 2: Set $r_0 := 1$ and $r_i := \frac{c_{-1,i}}{c_{+1,i}}$ for all $i = 1, \ldots, n$.
 3: repeat
 4:    Set $\mathbf{d}_{-1}^{(k)} := \bar{\mathbf{w}}^{(k)} - \mathbf{w}^{(k)}$ where $\bar{\mathbf{w}}^{(k)} := \operatorname{argmax}_{\mathbf{w}' \in \mathcal{W}} \vartheta_{\mathbf{r}}\left(\mathbf{w}^{(k)}, \dot{\mathbf{x}}^{(k)}, \mathbf{w}', \dot{\mathbf{x}}^{(k)}\right)$.
 5:    Set $\mathbf{d}_{+1}^{(k)} := \bar{\mathbf{x}}^{(k)} - \dot{\mathbf{x}}^{(k)}$ where $\bar{\mathbf{x}}^{(k)} := \operatorname{argmax}_{\dot{\mathbf{x}}' \in \phi(\mathcal{X})^n} \vartheta_{\mathbf{r}}\left(\mathbf{w}^{(k)}, \dot{\mathbf{x}}^{(k)}, \mathbf{w}^{(k)}, \dot{\mathbf{x}}'\right)$.
 6:    Find maximal step size $t^{(k)} \in \left\{\beta^l \mid l \in \mathbb{N}\right\}$ with
       $\bar{\vartheta}_{\mathbf{r}}\left(\mathbf{w}^{(k)}, \dot{\mathbf{x}}^{(k)}\right) - \bar{\vartheta}_{\mathbf{r}}\left(\mathbf{w}^{(k)} + t^{(k)}\mathbf{d}_{-1}^{(k)}, \dot{\mathbf{x}}^{(k)} + t^{(k)}\mathbf{d}_{+1}^{(k)}\right) \geq \sigma\, t^{(k)} \left(\left\|\mathbf{d}_{-1}^{(k)}\right\|_2^2 + \left\|\mathbf{d}_{+1}^{(k)}\right\|_2^2\right).$
 7:    Set $\mathbf{w}^{(k+1)} := \mathbf{w}^{(k)} + t^{(k)}\mathbf{d}_{-1}^{(k)}$.
 8:    Set $\dot{\mathbf{x}}^{(k+1)} := \dot{\mathbf{x}}^{(k)} + t^{(k)}\mathbf{d}_{+1}^{(k)}$.
 9:    Set $k := k + 1$.
10: until $\left\|\mathbf{w}^{(k)} - \mathbf{w}^{(k-1)}\right\|_2^2 + \left\|\dot{\mathbf{x}}^{(k)} - \dot{\mathbf{x}}^{(k-1)}\right\|_2^2 \leq \epsilon$.
```

The convergence properties of Algorithm 1 are discussed in von Heusinger and Kanzow (2009), so we skip the details here.

4.2 A Modified Extragradient Approach

In Algorithm 1, line 4 and 5, as well as the linesearch in line 6, require to solve a concave maximization problem within each iteration. As this may become computationally demanding, we derive a second approach based on extragradient descent. Therefore, instead of reformulating the equilibrium problem into a minimax problem, we directly address the first-order optimality conditions of each players' minimization problem in (4): Under Assumption 2, a combination of actions $(\mathbf{w}^*, \dot{\mathbf{x}}^*)$ with $\dot{\mathbf{x}}^* = \left[\dot{\mathbf{x}}_1^{*\mathsf{T}}, \ldots, \dot{\mathbf{x}}_n^{*\mathsf{T}}\right]^{\mathsf{T}}$ satisfies each player's first-order optimality conditions if, and only if, for all $(\mathbf{w}, \dot{\mathbf{x}}) \in \mathcal{W} \times \phi(\mathcal{X})^n$ the following inequalities hold:

$$\begin{array}{rcl}\nabla_{\mathbf{w}}\hat{\theta}_{-1}(\mathbf{w}^*,\dot{\mathbf{x}}^*)^{\mathsf{T}}(\mathbf{w}-\mathbf{w}^*) & \geq & 0,\\ \nabla_{\dot{\mathbf{x}}_i}\hat{\theta}_{+1}(\mathbf{w}^*,\dot{\mathbf{x}}^*)^{\mathsf{T}}(\dot{\mathbf{x}}_i-\dot{\mathbf{x}}_i^*) & \geq & 0 \quad \forall\, i=1,\ldots,n.\end{array}$$

As the joint action space of all players $\mathcal{W}\times\phi(\mathcal{X})^n$ is precisely the full Cartesian product of the learner's action set $\mathcal{W}$ and the n data generators' action sets $\phi(\mathcal{X})$, the (weighted) sum of those individual optimality conditions is also a sufficient and necessary optimality condition for the equilibrium problem. Hence, a Nash equilibrium $(\mathbf{w}^*,\dot{\mathbf{x}}^*)\in\mathcal{W}\times\phi(\mathcal{X})^n$ is a solution of the *variational inequality problem*,

$$\mathbf{g}_{\mathbf{r}}(\mathbf{w}^*,\dot{\mathbf{x}}^*)^{\mathsf{T}}\left(\begin{bmatrix}\mathbf{w}\\ \dot{\mathbf{x}}\end{bmatrix}-\begin{bmatrix}\mathbf{w}^*\\ \dot{\mathbf{x}}^*\end{bmatrix}\right)\geq 0 \quad \forall\,(\mathbf{w},\dot{\mathbf{x}})\in\mathcal{W}\times\phi(\mathcal{X})^n \tag{19}$$

and vice versa (*cf.* Proposition 7.1 of Harker and Pang 1990). The pseudo-gradient $\mathbf{g}_{\mathbf{r}}$ in (19) is defined as in (5) with fixed vector $\mathbf{r}=[r_0,r_1,\ldots,r_n]^{\mathsf{T}}$ where $r_0:=1$ and $r_i:=\frac{c_{-1,i}}{c_{+1,i}}$ for all $i=1,\ldots,n$ (*cf.* Equation 11). Under Assumption 3, this choice of $\mathbf{r}$ ensures that the mapping $\mathbf{g}_{\mathbf{r}}(\mathbf{w},\dot{\mathbf{x}})$ is continuous and strictly monotone (*cf.* proof of Lemma 2 and Theorem 8). Hence, the variational inequality problem in (19) can be solved by *modified extragradient descent* (see, for instance, Chapter 7.2.3 of Geiger and Kanzow 1999). Before presenting Algorithm 2 which is an extragradient-based algorithm for the Nash prediction game, let us denote the L^2-projection of $\mathbf{a}$ into the nonempty, compact, and convex set $\mathcal{A}$ by

$$\Pi_{\mathcal{A}}(\mathbf{a}):=\arg\min_{\mathbf{a}'\in\mathcal{A}}\|\mathbf{a}-\mathbf{a}'\|_2^2.$$

Notice, that if $\mathcal{A}:=\{\mathbf{a}\in\mathbb{R}^m\mid\|\mathbf{a}\|_2\leq\kappa\}$ is the closed l^2-ball of radius $\kappa>0$ and $\mathbf{a}\notin\mathcal{A}$, this projection simply reduces to a rescaling of vector $\mathbf{a}$ to length κ.

Based on this definition of $\Pi_{\mathcal{A}}$, we can now state an iterative method (Algorithm 2) which—apart from back projection steps—does not require to solve an optimization problem in each iteration. The proposed algorithm converges to a solution of the variational inequality problem in 19, *i.e.,* the unique equilibrium of the Nash prediction game, if Assumptions 2 and 3 hold—*cf.* Theorem 7.40 of Geiger and Kanzow (1999).

5. Instances of the Nash Prediction Game

In this section we present two instances of the Nash prediction game and investigate under which conditions those games possess unique Nash equilibria. We start by specifying both players' loss functions and regularizers. An obvious choice for the loss function of the learner $\ell_{-1}(z,y)$ is the *zero-one loss* defined by

$$\ell_{0/1}(z,y):=\begin{cases}1 & \text{, if } yz<0\\ 0 & \text{, if } yz\geq 0\end{cases}.$$

A possible choice for the data generator's loss is $\ell_{0/1}(z,-1)$ which penalizes positive decision values z, independently of the class label. The rationale behind this choice is that the data generator experiences costs when the learner blocks an event, that is, assigns an instance to the positive class. For instance, a legitimate email sender experiences costs when a

Algorithm 2 EDS: Extragradient Descent Solver for Nash Prediction Games

```
Require: Cost functions $\hat{\theta}_v$ as defined in (1) and (2), and action spaces $\mathcal{W}$ and $\phi(\mathcal{X})^n$.
Select initial $\mathbf{w}^{(0)} \in \mathcal{W}$, set $\dot{\mathbf{x}}^{(0)} := \mathbf{x}$, set $k := 0$, and select $\sigma \in (0,1)$ and $\beta \in (0,1)$.
Set $r_0 := 1$ and $r_i := \frac{c_{-1,i}}{c_{+1,i}}$ for all $i = 1, \ldots, n$.
repeat
    Set $\begin{bmatrix} \mathbf{d}_{-1}^{(k)} \\ \mathbf{d}_{+1}^{(k)} \end{bmatrix} := \Pi_{\mathcal{W} \times \phi(\mathcal{X})^n} \left( \begin{bmatrix} \mathbf{w}^{(k)} \\ \dot{\mathbf{x}}^{(k)} \end{bmatrix} - \mathbf{g_r}\left(\mathbf{w}^{(k)}, \dot{\mathbf{x}}^{(k)}\right) \right) - \begin{bmatrix} \mathbf{w}^{(k)} \\ \dot{\mathbf{x}}^{(k)} \end{bmatrix}$.
    Find maximal step size $t^{(k)} \in \left\{ \beta^l \mid l \in \mathbb{N} \right\}$ with
        $$-\mathbf{g_r}\left(\mathbf{w}^{(k)} + t^{(k)}\mathbf{d}_{-1}^{(k)}, \dot{\mathbf{x}}^{(k)} + t^{(k)}\mathbf{d}_{+1}^{(k)}\right)^\mathsf{T} \begin{bmatrix} \mathbf{d}_{-1}^{(k)} \\ \mathbf{d}_{+1}^{(k)} \end{bmatrix} \geq \sigma \left( \left\| \mathbf{d}_{-1}^{(k)} \right\|_2^2 + \left\| \mathbf{d}_{+1}^{(k)} \right\|_2^2 \right).$$
    Set $\begin{bmatrix} \bar{\mathbf{w}}^{(k)} \\ \bar{\mathbf{x}}^{(k)} \end{bmatrix} := \begin{bmatrix} \mathbf{w}^{(k)} \\ \dot{\mathbf{x}}^{(k)} \end{bmatrix} + t^{(k)} \begin{bmatrix} \mathbf{d}_{-1}^{(k)} \\ \mathbf{d}_{+1}^{(k)} \end{bmatrix}$.
    Set step size of extragradient
        $$\gamma^{(k)} := -\frac{t^{(k)}}{\left\| \mathbf{g_r}\left(\bar{\mathbf{w}}^{(k)}, \bar{\mathbf{x}}^{(k)}\right) \right\|_2^2} \mathbf{g_r}\left(\bar{\mathbf{w}}^{(k)}, \bar{\mathbf{x}}^{(k)}\right)^\mathsf{T} \begin{bmatrix} \mathbf{d}_{-1}^{(k)} \\ \mathbf{d}_{+1}^{(k)} \end{bmatrix}.$$
    Set $\begin{bmatrix} \mathbf{w}^{(k+1)} \\ \dot{\mathbf{x}}^{(k+1)} \end{bmatrix} := \Pi_{\mathcal{W} \times \phi(\mathcal{X})^n} \left( \begin{bmatrix} \mathbf{w}^{(k)} \\ \dot{\mathbf{x}}^{(k)} \end{bmatrix} - \gamma^{(k)} \mathbf{g_r}\left(\bar{\mathbf{w}}^{(k)}, \bar{\mathbf{x}}^{(k)}\right) \right)$.
    Set $k := k + 1$.
until $\left\| \mathbf{w}^{(k)} - \mathbf{w}^{(k-1)} \right\|_2^2 + \left\| \dot{\mathbf{x}}^{(k)} - \dot{\mathbf{x}}^{(k-1)} \right\|_2^2 \leq \epsilon$.
```

legitimate email is erroneously blocked just like an abusive sender, also amalgamated into the *data generator*, experiences costs when spam messages are blocked.

However, the zero-one loss violates Assumption 2 as it is neither convex nor twice continuously differentiable. In the following sections, we therefore approximate the zero-one loss by the *logistic loss* and a newly derived *trigonometric loss* which both satisfy Assumption 2.

To regularize the players' actions, recall that $\hat{\Omega}_{+1}(D, \dot{D})$ is an estimate of the transformation costs that the data generator incurs when shifting the training distribution—where the training instances x_i are drawn from—to the test distribution which is empirically represented by the transformed training instances $\dot{x}_i$. In our analysis, we approximate these costs by the average squared l^2-distance between x_i and $\dot{x}_i$ in the feature space induced by mapping ϕ, that is,

$$\hat{\Omega}_{+1}(D, \dot{D}) := \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2} \left\| \phi(\dot{x}_i) - \phi(x_i) \right\|_2^2. \tag{20}$$

The learner's regularizer $\hat{\Omega}_{-1}(\mathbf{w})$ penalizes the complexity of the predictive model $h(x) = \operatorname{sign} f_\mathbf{w}(x)$. We consider Tikhonov regularization which, for linear decision functions $f_\mathbf{w}$, reduces to the squared ℓ^2-norm of $\mathbf{w}$,

$$\hat{\Omega}_{-1}(\mathbf{w}) := \frac{1}{2}\|\mathbf{w}\|_2^2. \tag{21}$$

Before presenting the *Nash logistic regression* (NLR) and the *Nash support vector machine* (NSVM), we turn to a discussion on the applicability of general kernel functions.

5.1 Applying Kernels

So far, we assumed the knowledge of feature mapping $\phi : \mathcal{X} \rightarrow \phi(\mathcal{X})$ such that we can compute an explicit feature representation $\phi(x_i)$ of the training instances x_i for all $i = 1, \ldots, n$. However, in some applications, such a feature mapping is unwieldy or hard to identify. Instead, one is often equipped with a kernel function $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ which measures the similarity between two instances. Generally, kernel function k is assumed to be a positive-semidefinite kernel such that it can be stated in terms of a scalar product in the corresponding reproducing kernel Hilbert space, that is, $\exists \phi$ with $k(x, x') = \phi(x)^{\mathsf{T}}\phi(x')$.

To apply the representer theorem (see, *e.g.,* Schölkopf et al. 2001) we assume that the transformed instances lie in the span of the mapped training instances, that is, we restrict the data generator's action space such that the transformed instances $\dot{x}_i$ are mapped into the same subspace of the reproducing kernel Hilbert space as the unmodified training instances x_i. By this assumption, the weight vector $\mathbf{w} \in \mathcal{W}$ and the transformed instances $\phi(\dot{x}_i) \in \phi(\mathcal{X})$ for $i = 1, \ldots, n$ can be expressed as linear combinations of the mapped training instances, *i.e.,* $\exists \alpha_i, \Xi_{ij}$ such that

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i \phi(x_i) \quad \text{and} \quad \phi(\dot{x}_j) = \sum_{i=1}^{n} \Xi_{ij}\phi(x_i) \quad \forall\, j = 1, \ldots, n.$$

Further, let us assume that the action spaces $\mathcal{W}$ and $\phi(\mathcal{X})^n$ can be adequately translated into dual action spaces $\mathcal{A} \subset \mathbb{R}^n$ and $\mathcal{Z} \subset \mathbb{R}^{n \times n}$, which is possible, for instance, if $\mathcal{W}$ and $\phi(\mathcal{X})^n$ are closed ℓ^2-balls. Then, a kernelized variant of the Nash prediction game is obtained by inserting the above equations into the players' cost functions in (1) and (2) with regularizers in (20) and (21),

$$\hat{\theta}_{-1}(\boldsymbol{\alpha}, \boldsymbol{\Xi}) = \sum_{i=1}^{n} c_{-1,i}\ell_{-1}(\boldsymbol{\alpha}^{\mathsf{T}}\mathbf{K}\boldsymbol{\Xi}\mathbf{e}_i, y_i) + \rho_{-1}\frac{1}{2}\boldsymbol{\alpha}^{\mathsf{T}}\mathbf{K}\boldsymbol{\alpha}, \tag{22}$$

$$\hat{\theta}_{+1}(\boldsymbol{\alpha}, \boldsymbol{\Xi}) = \sum_{i=1}^{n} c_{+1,i}\ell_{+1}(\boldsymbol{\alpha}^{\mathsf{T}}\mathbf{K}\boldsymbol{\Xi}\mathbf{e}_i, y_i) + \rho_{+1}\frac{1}{2n}\mathrm{tr}\left((\boldsymbol{\Xi} - \mathbf{I}_n)^{\mathsf{T}}\mathbf{K}(\boldsymbol{\Xi} - \mathbf{I}_n)\right), \tag{23}$$

where $\mathbf{e}_i \in \{0,1\}^n$ is the i-th unit vector, $\boldsymbol{\alpha} \in \mathcal{A}$ is the dual weight vector, $\boldsymbol{\Xi} \in \mathcal{Z}$ is the dual transformed data matrix, and $\mathbf{K} \in \mathbb{R}^{n \times n}$ is the kernel matrix with $K_{ij} := k(x_i, x_j)$. In the dual Nash prediction game with cost functions (22) and (23), the learner chooses the dual weight vector $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]^{\mathsf{T}}$ and classifies a new instance x by $h(x) = \operatorname{sign} f_{\boldsymbol{\alpha}}(x)$ with $f_{\boldsymbol{\alpha}}(x) = \sum_{i=1}^{n} \alpha_i k(x_i, x)$. In contrast, the data generator chooses the dual transformed data matrix $\boldsymbol{\Xi}$ which implicitly reflects the change of the training distribution. Their transformation costs are in proportion to the deviation of $\boldsymbol{\Xi}$ from the identity matrix $\mathbf{I}_n$, where if

$\boldsymbol{\Xi}$ equals $\mathbf{I}_n$, the learner's task reduces to standard kernelized empirical risk minimization. The proposed Algorithms 1 and 2 can be readily applied when replacing $\mathbf{w}$ by $\boldsymbol{\alpha}$ and $\dot{\mathbf{x}}_i$ by $\boldsymbol{\Xi}\mathbf{e}_i$ for all $i = 1, \ldots, n$.

An alternative approach to a kernelization of the Nash prediction game is to first construct an explicit feature representation with respect to the given kernel function k and the training instances, and then to train the Nash model by applying this feature mapping. Here, we again assume that the transformed instances $\phi(\dot{x}_i)$ as well as the weight vector $\mathbf{w}$ lie in the span of the explicitly mapped training instances $\phi(x)$. Let us consider the kernel PCA map (see, *e.g.,* Schölkopf and Smola 2002) which is defined by

$$\phi_{\text{PCA}} : x \mapsto \boldsymbol{\Lambda}^{\frac{1}{2}^{+}} \mathbf{V}^{\mathsf{T}} \left[k(x_1, x), \ldots, k(x_n, x)\right]^{\mathsf{T}}, \tag{24}$$

where $\mathbf{V}$ is the column matrix of eigenvectors of kernel matrix $\mathbf{K}$, $\boldsymbol{\Lambda}$ is the diagonal matrix with the corresponding eigenvalues such that $\mathbf{K} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{\mathsf{T}}$, and $\boldsymbol{\Lambda}^{\frac{1}{2}^{+}}$ denotes the pseudo-inverse of the square root of $\boldsymbol{\Lambda}$ with $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}^{\frac{1}{2}}\boldsymbol{\Lambda}^{\frac{1}{2}}$.

Remark 9 Notice that for any positive-semidefinite kernel function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and fixed training instances $x_1, \ldots, x_n \in \mathcal{X}$, the PCA map is a uniquely defined real function with $\phi_{\text{PCA}} : \mathcal{X} \to \mathbb{R}^n$ such that $k(x_i, x_j) = \phi_{\text{PCA}}(x_i)^{\mathsf{T}}\phi_{\text{PCA}}(x_j)$ for any $i, j \in \{1, \ldots, n\}$: We first show that ϕ_{PCA} is a real mapping from the input space $\mathcal{X}$ to the Euclidean space $\mathbb{R}^n$. As $x \mapsto [k(x_1, x), \ldots, k(x_n, x)]^{\mathsf{T}}$ is a real vector-valued function and $\mathbf{V}$ is a real $n \times n$ matrix, it remains to show that the pseudo-inverse of $\boldsymbol{\Lambda}^{\frac{1}{2}}$ is real as well. Since the kernel function is positive-semidefinite, all eigenvalues λ_i of $\mathbf{K}$ are non-negative, and hence, $\boldsymbol{\Lambda}^{\frac{1}{2}}$ is a diagonal matrix with real diagonal entries $\sqrt{\lambda_i}$ for $i = 1, \ldots, n$. The pseudo-inverse of this matrix is the uniquely defined diagonal matrix $\boldsymbol{\Lambda}^{\frac{1}{2}^{+}}$ with real non-negative diagonal entries $\frac{1}{\sqrt{\lambda_i}}$ if $\lambda_i > 0$ and zero otherwise. This proves the first claim. The PCA map also satisfies $k(x_i, x_j) = \phi_{\text{PCA}}(x_i)^{\mathsf{T}}\phi_{\text{PCA}}(x_j)$ for any pair of training instances x_i and x_j as

$$\begin{aligned}
\phi_{PCA}(x_i) &= \boldsymbol{\Lambda}^{\frac{1}{2}^{+}} \mathbf{V}^{\mathsf{T}} \left[k(x_1, x_i), \ldots, k(x_n, x_i)\right]^{\mathsf{T}} \\
&= \boldsymbol{\Lambda}^{\frac{1}{2}^{+}} \mathbf{V}^{\mathsf{T}}\mathbf{K}\mathbf{e}_i \\
&= \boldsymbol{\Lambda}^{\frac{1}{2}^{+}} \mathbf{V}^{\mathsf{T}}\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{\mathsf{T}}\mathbf{e}_i \\
&= \boldsymbol{\Lambda}^{\frac{1}{2}^{+}} \boldsymbol{\Lambda}\mathbf{V}^{\mathsf{T}}\mathbf{e}_i
\end{aligned}$$

for all $i = 1, \ldots, n$ and consequently

$$\begin{aligned}
\phi_{PCA}(x_i)^{\mathsf{T}}\phi_{PCA}(x_j) &= \mathbf{e}_i^{\mathsf{T}}\mathbf{V}\boldsymbol{\Lambda}\boldsymbol{\Lambda}^{\frac{1}{2}^{+}}\boldsymbol{\Lambda}^{\frac{1}{2}^{+}}\boldsymbol{\Lambda}\mathbf{V}^{\mathsf{T}}\mathbf{e}_j \\
&= \mathbf{e}_i^{\mathsf{T}}\mathbf{V}\boldsymbol{\Lambda}\boldsymbol{\Lambda}^{+}\boldsymbol{\Lambda}\mathbf{V}^{\mathsf{T}}\mathbf{e}_j \\
&= \mathbf{e}_i^{\mathsf{T}}\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{\mathsf{T}}\mathbf{e}_j \\
&= \mathbf{e}_i^{\mathsf{T}}\mathbf{K}\mathbf{e}_j = \mathbf{K}_{ij} = k(x_i, x_j)
\end{aligned}$$

which proves the second claim. ◇

An equilibrium strategy pair $\mathbf{w}^* \in \mathcal{W}$ and $[\phi_{\text{PCA}}(\dot{x}_1^*)^\mathsf{T}, \ldots, \phi_{\text{PCA}}(\dot{x}_n^*)^\mathsf{T}]^\mathsf{T} \in \phi(\mathcal{X})^n$ can be identified by applying the PCA map together with Algorithms 1 or 2. To classify a new instance $x \in \mathcal{X}$ we may first map x into the PCA map-induced feature space and apply the linear classifier $h(x) = \operatorname{sign} f_{\mathbf{w}^*}(x)$ with $f_{\mathbf{w}^*}(x) = {\mathbf{w}^*}^\mathsf{T}\phi_{\text{PCA}}(x)$. Alternatively, we can derive a dual representation of $\mathbf{w}^*$ such that $\mathbf{w}^* = \sum_{i=1}^n \alpha_i^* \phi_{\text{PCA}}(x_i)$, and consequently $f_{\mathbf{w}^*}(x) = f_{\boldsymbol{\alpha}^*}(x) = \sum_{i=1}^n \alpha_i^* k(x_i, x)$, where $\boldsymbol{\alpha}^* = [\alpha_1^*, \ldots, \alpha_n^*]^\mathsf{T}$ is a not necessarily uniquely defined dual weight vector of $\mathbf{w}^*$. Therefore, we have to identify a solution $\boldsymbol{\alpha}^*$ of the linear system

$$\mathbf{w}^* = {\boldsymbol{\Lambda}^{\frac{1}{2}}}^+ \mathbf{V}^\mathsf{T}\mathbf{K}\boldsymbol{\alpha}^*. \tag{25}$$

A direct calculation shows that

$$\boldsymbol{\alpha}^* := \mathbf{V}{\boldsymbol{\Lambda}^{\frac{1}{2}}}^+ \mathbf{w}^* \tag{26}$$

is a solution of (25) provided that either all elements λ_i of the diagonal matrix $\boldsymbol{\Lambda}$ are positive or that $\lambda_i = 0$ implies that the same component of the vector $\mathbf{w}^*$ is also equal to zero (in which case the solution is nonunique). In fact, inserting (26) in (25) then gives

$${\boldsymbol{\Lambda}^{\frac{1}{2}}}^+ \mathbf{V}^\mathsf{T}\mathbf{K}\boldsymbol{\alpha}^* = {\boldsymbol{\Lambda}^{\frac{1}{2}}}^+ \mathbf{V}^\mathsf{T}\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^\mathsf{T}\mathbf{V}{\boldsymbol{\Lambda}^{\frac{1}{2}}}^+ \mathbf{w}^* = {\boldsymbol{\Lambda}^{\frac{1}{2}}}^+ \boldsymbol{\Lambda}^{\frac{1}{2}}\boldsymbol{\Lambda}^{\frac{1}{2}}{\boldsymbol{\Lambda}^{\frac{1}{2}}}^+ \mathbf{w}^* = \mathbf{w}^*$$

whereas in the other cases the linear system (25) is obviously inconsistent. The advantage of the latter approach is that classifying a new instances $x \in \mathcal{X}$ requires the computation of the scalar product $\sum_{i=1}^n \alpha_i^* k(x_i, x)$ rather than a matrix multiplication when mapping x into the PCA map-induced feature space (*cf.* Equation 24).

5.2 Nash Logistic Regression

In this section we study the particular instance of the Nash prediction game where each players' loss function rests on the negative logarithm of the logistic function $\sigma(a) := \frac{1}{1+e^{-a}}$, that is, the *logistic loss*

$$\ell^\text{l}(z, y) := -\log \sigma(yz) = \log\left(1 + e^{-yz}\right). \tag{27}$$

We consider the regularizers in (20) and (21), respectively, which give rise to the following definition of the *Nash logistic regression* (NLR).

Definition 10 *The* Nash logistic regression *(NLR) is an instance of the Nash prediction game with nonempty, compact, and convex action spaces $\mathcal{W} \subset \mathbb{R}^m$ and $\phi(\mathcal{X})^n \subset \mathbb{R}^{m \cdot n}$ and cost functions*

$$\begin{aligned}
\hat{\theta}^\text{l}_{-1}(\mathbf{w}, \dot{\mathbf{x}}) &:= \sum_{i=1}^n c_{-1,i}\ell^\text{l}(\mathbf{w}^\mathsf{T}\dot{\mathbf{x}}_i, y_i) + \rho_{-1}\frac{1}{2}\|\mathbf{w}\|_2^2 \\
\hat{\theta}^\text{l}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) &:= \sum_{i=1}^n c_{+1,i}\ell^\text{l}(\mathbf{w}^\mathsf{T}\dot{\mathbf{x}}_i, -1) + \rho_{+1}\frac{1}{n}\sum_{i=1}^n \frac{1}{2}\left\|\dot{\mathbf{x}}_i - \mathbf{x}_i\right\|_2^2
\end{aligned}$$

where ℓ^l is specified in (27)*.*

In the above definition, column vectors $\mathbf{x} := [\mathbf{x}_1^{\mathsf{T}}, \ldots, \mathbf{x}_n^{\mathsf{T}}]^{\mathsf{T}}$ and $\dot{\mathbf{x}} := [\dot{\mathbf{x}}_1^{\mathsf{T}}, \ldots, \dot{\mathbf{x}}_n^{\mathsf{T}}]^{\mathsf{T}}$ again denote the concatenation of the original and the transformed training instances, respectively, which are mapped into the feature space by $\mathbf{x}_i := \phi(x_i)$ and $\dot{\mathbf{x}}_i := \phi(\dot{x}_i)$.

As in our introductory discussion, the data generator's loss function $\ell_{+1}(z, y) := \ell^{\mathrm{l}}(z, -1)$ penalizes positive decision values independently of the class label y. In contrast, instances that pass the classifier, *i.e.,* instances with negative decision values, incur little or almost no costs. By the above definition, the Nash logistic regression obviously satisfies Assumption 2, and according to the following corollary, also satisfies Assumption 3 for suitable regularization parameters.

Corollary 11 *Let the Nash logistic regression be specified as in Definition 10 with positive regularization parameters ρ_{-1} and ρ_{+1} which satisfy*

$$\rho_{-1}\rho_{+1} \geq n\mathbf{c}_{-1}^{\mathsf{T}}\mathbf{c}_{+1}, \tag{28}$$

then Assumption 2 and 3 hold, and consequently, the Nash logistic regression possess a unique Nash equilibrium.

Proof. By Definition 10, both players employ the logistic loss with $\ell_{-1}(z, y) := \ell^{\mathrm{l}}(z, y)$ and $\ell_{+1}(z, y) := \ell^{\mathrm{l}}(z, -1)$ and the regularizers in (20) and (21), respectively. Let

$$\begin{array}{rcl|rcl} \ell'_{-1}(z, y) & = & -y\frac{1}{1+e^{yz}} & \ell'_{+1}(z, y) & = & \frac{1}{1+e^{-z}} \\ \ell''_{-1}(z, y) & = & \frac{1}{1+e^{z}}\frac{1}{1+e^{-z}} & \ell''_{+1}(z, y) & = & \frac{1}{1+e^{z}}\frac{1}{1+e^{-z}} \end{array} \tag{29}$$

denote the first and second derivatives of the players' loss functions with respect to $z \in \mathbb{R}$. Further, let

$$\begin{array}{rcl|rcl} \nabla_{\mathbf{w}}\hat{\Omega}_{-1}(\mathbf{w}) & = & \mathbf{w} & \nabla_{\dot{\mathbf{x}}}\hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}}) & = & \frac{1}{n}(\dot{\mathbf{x}} - \mathbf{x}) \\ \nabla^2_{\mathbf{w},\mathbf{w}}\hat{\Omega}_{-1}(\mathbf{w}) & = & \mathbf{I}_m & \nabla^2_{\dot{\mathbf{x}},\dot{\mathbf{x}}}\hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}}) & = & \frac{1}{n}\mathbf{I}_{m\cdot n} \end{array} \tag{30}$$

denote the gradients and Hessians of the players' regularizers. Assumption 2 holds as:

1. The the second derivatives of $\ell_{-1}(z, y)$ and $\ell_{+1}(z, y)$ are positive and continuous for all $z \in \mathbb{R}$ and $y \in \mathcal{Y}$. Consequently, $\ell_v(z, y)$ is convex and twice continuously differentiable with respect to z for $v \in \{-1, +1\}$ and fixed y.

2. The Hessians of the players' regularizers are fixed, positive definite matrices and consequently both regularizers are twice continuously differentiable and uniformly strongly convex in $\mathbf{w} \in \mathcal{W}$ and $\dot{\mathbf{x}} \in \phi(\mathcal{X})^n$ (for any fixed $\mathbf{x} \in \phi(\mathcal{X})^n$), respectively.

3. By Definition 10, the players' action sets are nonempty, compact, and convex subsets of finite-dimensional Euclidean spaces.

Assumption 3 holds as for all $z \in \mathbb{R}$ and $y \in \mathcal{Y}$:

1. The second derivatives of $\ell_{-1}(z, y)$ and $\ell_{+1}(z, y)$ in (29) are equal.

2. The sum of the first derivatives of the loss functions is bounded,

$$\ell'_{-1}(z,y) + \ell'_{+1}(z,y) = -y\frac{1}{1+e^{yz}} + \frac{1}{1+e^{-z}} = \begin{cases} \frac{1-e^{-z}}{1+e^{-z}} & \text{, if } y=+1 \\ \frac{2}{1+e^{-z}} & \text{, if } y=-1 \end{cases} \in (-1,2),$$

which together with Equation 12 gives

$$\tau = \sup_{(\mathbf{x},y)\in\phi(\mathcal{X})\times\mathcal{Y}} \frac{1}{2}\left|\ell'_{-1}(f_{\mathbf{w}}(\mathbf{x}),y) + \ell'_{+1}(f_{\mathbf{w}}(\mathbf{x}),y)\right| < 1.$$

The supremum τ is strictly less than 1 since $f_{\mathbf{w}}(\mathbf{x})$ is finite for compact action sets $\mathcal{W}$ and $\phi(\mathcal{X})^n$. The smallest eigenvalues of the players' regularizers are $\lambda_{-1} = 1$ and $\lambda_{+1} = \frac{1}{n}$, such that inequalities

$$\rho_{-1}\rho_{+1} \geq n\mathbf{c}_{-1}^{\mathsf{T}}\mathbf{c}_{+1} > \tau^2\frac{1}{\lambda_{-1}\lambda_{+1}}\mathbf{c}_{-1}^{\mathsf{T}}\mathbf{c}_{+1}$$

hold.

3. The partial gradient $\nabla_{\dot{\mathbf{x}}_i}\hat{\Omega}_{+1}(\mathbf{x},\dot{\mathbf{x}}) = \frac{1}{n}(\dot{\mathbf{x}}_i - \mathbf{x}_i)$ of the data generator's regularizer is independent of $\dot{\mathbf{x}}_j$ for all $j \neq i$ and $i = 1,\ldots,n$.

As Assumptions 2 and 3 are satisfied, the existence of a unique Nash equilibrium follows immediately from Theorem 8. ■

Recall, that the weighting factors $c_{v,i}$ are strictly positive with $\sum_{i=1}^{n} c_{v,i} = 1$ for both players $v \in \{-1,+1\}$. In particular, it therefore follows that in the unweighted case where $c_{v,i} = \frac{1}{n}$ for all $i = 1,\ldots,n$ and $v \in \{-1,+1\}$, a sufficient condition to ensure the existence of a unique Nash equilibrium is to set the learner's regularization parameter to $\rho_{-1} \geq \frac{1}{\rho_{+1}}$.

5.3 Nash Support Vector Machine

The Nash logistic regression tends to non-sparse solutions. This becomes particularly apparent if the Nash equilibrium $(\mathbf{w}^*,\dot{\mathbf{x}}^*)$ is an interior point of the joined action set $\mathcal{W}\times\phi(\mathcal{X})^n$ in which case the (partial) gradients in (6) and (7) are zero at $(\mathbf{w}^*,\dot{\mathbf{x}}^*)$. For regularizer (21), this implies that $\mathbf{w}^*$ is a linear combination of the transformed instances $\dot{\mathbf{x}}_i$ where all weighting factors are non-zero since the first derivative of the logistic loss as well as the cost factors $c_{-1,i}$ are non-zero for all $i = 1,\ldots,n$. The *support vector machine* (SVM), which employs the *hinge loss*,

$$\ell^{\mathrm{h}}(z,y) := \max(0, 1-yz) = \begin{cases} 1-yz & \text{, if } yz < 1 \\ 0 & \text{, if } yz \geq 1 \end{cases},$$

does not suffer from non-sparsity, however, the hinge loss obviously violates Assumption 2 as it is not twice continuously differentiable. Therefore, we propose a twice continuously differentiable loss function that we call *trigonometric loss*, which satisfies Assumptions 2 and 3.

Definition 12 *For any fixed smoothness factor $s > 0$, the* trigonometric loss *is defined by*

$$\ell^{\mathrm{t}}(z, y) := \begin{cases} -yz & , \textit{if } \ yz \ < -s \\ \frac{s-yz}{2} - \frac{s}{\pi}\cos\left(\frac{\pi}{2s}yz\right) & , \textit{if } |yz| \leq s \\ 0 & , \textit{if } \ yz \ > s \end{cases} . \tag{31}$$

The trigonometric loss is similar to the hinge loss in that it, except around the decision boundary, penalizes misclassifications in proportion to the decision value $z \in \mathbb{R}$ and attains zero for correctly classified instances. Analog to the once continuously differentiable Huber loss where a polynomial is embedded into the hinge loss, the trigonometric loss combines the perceptron loss $\ell^{\mathrm{p}}(z, y) := \max(0, -yz)$ with a trigonometric function. This trigonometrical embedding yields a convex, twice continuously differentiable function.

Lemma 13 *The trigonometric loss $\ell^{\mathrm{t}}(z, y)$ is convex and twice continuously differentiable with respect to $z \in \mathbb{R}$ for any fixed $y \in \mathcal{Y}$.*

Proof. Let

$$\ell^{\mathrm{t}\prime}(z, y) = \begin{cases} -y & , \text{if } \ yz \ < -s \\ -\frac{1}{2}y + \frac{1}{2}y\sin\left(\frac{\pi}{2s}yz\right) & , \text{if } |yz| \leq s \\ 0 & , \text{if } \ yz \ > s \end{cases} \tag{32}$$

$$\ell^{\mathrm{t}\prime\prime}(z, y) = \begin{cases} 0 & , \text{if } \ yz \ < -s \\ \frac{\pi}{4s}\cos\left(\frac{\pi}{2s}yz\right) & , \text{if } |yz| \leq s \\ 0 & , \text{if } \ yz \ > s \end{cases} \tag{33}$$

denote the first and second derivatives of $\ell^{\mathrm{t}}(z, y)$, respectively, with respect to $z \in \mathbb{R}$. The trigonometric loss $\ell^{\mathrm{t}}(z, y)$ is convex in $z \in \mathbb{R}$ (for any fixed $y \in \mathcal{Y}$) as the second derivative $\ell^{\mathrm{t}\prime\prime}(z, y)$ is strictly positive if $|z| = |yz| < s$ and zero otherwise. Moreover, since the second derivative is continuous,

$$\lim_{|z| \to s-} \ell^{\mathrm{t}\prime\prime}(z, y) = \frac{\pi}{4s}\cos\left(\pm\frac{\pi}{2}\right) = 0 = \lim_{|z| \to s+} \ell^{\mathrm{t}\prime\prime}(z, y),$$

the trigonometric loss is also twice continuously differentiable. ■

Because of the similarities of the loss functions, we call the Nash prediction game that is based upon the trigonometric loss *Nash support vector machine* (NSVM) where we again consider the regularizers in (20) and (21).

Definition 14 *The* Nash support vector machine *(NSVM) is an instance of the Nash prediction game with nonempty, compact, and convex action spaces $\mathcal{W} \subset \mathbb{R}^m$ and $\phi(\mathcal{X})^n \subset \mathbb{R}^{m \cdot n}$ and cost functions*

$$\hat{\theta}^{\mathrm{t}}_{-1}(\mathbf{w}, \dot{\mathbf{x}}) := \sum_{i=1}^{n} c_{-1,i}\ell^{\mathrm{t}}(\mathbf{w}^{\mathsf{T}}\dot{\mathbf{x}}_i, y_i) + \rho_{-1}\frac{1}{2}\|\mathbf{w}\|_2^2$$

$$\hat{\theta}^{\mathrm{t}}_{+1}(\mathbf{w}, \dot{\mathbf{x}}) := \sum_{i=1}^{n} c_{+1,i}\ell^{\mathrm{t}}(\mathbf{w}^{\mathsf{T}}\dot{\mathbf{x}}_i, -1) + \rho_{+1}\frac{1}{n}\sum_{i=1}^{n}\frac{1}{2}\|\dot{\mathbf{x}}_i - \mathbf{x}_i\|_2^2$$

where ℓ^{t} is specified in (31).

The following corollary states sufficient conditions under which the Nash support vector machine satisfies Assumptions 2 and 3, and consequently has a unique Nash equilibrium.

Corollary 15 *Let the Nash support vector machine be specified as in Definition 14 with positive regularization parameters ρ_{-1} and ρ_{+1} which satisfy*

$$\rho_{-1}\rho_{+1} > n\mathbf{c}_{-1}^{\mathsf{T}}\mathbf{c}_{+1}, \tag{34}$$

then Assumptions 2 and 3 hold, and consequently, the Nash support vector machine has a unique Nash equilibrium.

Proof. By Definition 14, both players employ the trigonometric loss with $\ell_{-1}(z, y) := \ell^{\mathrm{t}}(z, y)$ and $\ell_{+1}(z, y) := \ell^{\mathrm{t}}(z, -1)$ and the regularizers in (20) and (21), respectively. Assumption 2 holds:

1. According to Lemma 13, $\ell^{\mathrm{t}}(z, y)$, and consequently $\ell_{-1}(z, y)$ and $\ell_{+1}(z, y)$, are convex and twice continuously differentiable with respect to $z \in \mathbb{R}$ (for any fixed $y \in \{-1, +1\}$).
2. The regularizers of the Nash support vector machine are equal to that of the Nash logistic regression and possess the same properties as in Theorem 11.
3. By Definition 14, the players' action sets are nonempty, compact, and convex subsets of finite-dimensional Euclidean spaces.

Assumption 3 holds:

1. The second derivatives of $\ell_{-1}(z, y)$ and $\ell_{+1}(z, y)$ are equal for all $z \in \mathbb{R}$ since

   $$\ell^{\mathrm{t}\prime\prime}(z, y) = \begin{cases} \frac{\pi}{4s}\cos\left(\frac{\pi}{2s}z\right) & \text{, if } |z| \leq s \\ 0 & \text{, if } |z| > s \end{cases}$$

   does not dependent on $y \in \mathcal{Y}$.
2. The sum of the first derivatives of the loss functions is bounded as for $y = -1$:

   $$\ell'_{-1}(z, -1) + \ell'_{+1}(z, -1) = 2\ell^{\mathrm{t}\prime}(z, -1) = \begin{cases} 0 & \text{, if } z < -s \\ 1 - \sin\left(-\frac{\pi}{2s}z\right) & \text{, if } |z| \leq s \\ 2 & \text{, if } z > s \end{cases} \in [0, 2],$$

   and for $y = +1$:

   $$\ell'_{-1}(z, +1) + \ell'_{+1}(z, +1) = \begin{cases} -1 & \text{, if } z < -s \\ \sin\left(\frac{\pi}{2s}z\right) & \text{, if } |z| \leq s \\ 1 & \text{, if } z > s \end{cases} \in [-1, 1].$$

   Together with Equation 12, it follows that

   $$\tau = \sup_{(\mathbf{x},y)\in\phi(\mathcal{X})\times\mathcal{Y}} \frac{1}{2}\left|\ell'_{-1}(f_{\mathbf{w}}(\mathbf{x}), y) + \ell'_{+1}(f_{\mathbf{w}}(\mathbf{x}), y)\right| \leq 1.$$

The smallest eigenvalues of the players' regularizers are $\lambda_{-1} = 1$ and $\lambda_{+1} = \frac{1}{n}$, such that inequalities

$$\rho_{-1}\rho_{+1} > n\mathbf{c}_{-1}^{\mathsf{T}}\mathbf{c}_{+1} \geq \tau^2 \frac{1}{\lambda_{-1}\lambda_{+1}} \mathbf{c}_{-1}^{\mathsf{T}}\mathbf{c}_{+1}$$

hold.

3. As for Nash logistic regression, the partial gradient $\nabla_{\dot{\mathbf{x}}_i}\hat{\Omega}_{+1}(\mathbf{x}, \dot{\mathbf{x}}) = \frac{1}{n}(\dot{\mathbf{x}}_i - \mathbf{x}_i)$ of the data generator's regularizer is independent of $\dot{\mathbf{x}}_j$ for all $j \neq i$ and $i = 1, \dots, n$.

Because Assumptions 2 and 3 are satisfied, the existence of a unique Nash equilibrium follows immediately from Theorem 8. ■

6. Experimental Evaluation

The goal of this section is to explore the relative strengths and weaknesses of the discussed instances of the Nash prediction game and existing baseline methods in the context of email spam filtering. We compare a regular *support vector machine* (SVM), *logistic regression* (LR), the *feature-deleted regularized optimization problem* (FDROP, Globerson and Roweis 2006), and the Nash instances *Nash logistic regression* (NLR) and *Nash support vector machine* (NSVM).

We use four corpora of chronologically sorted emails detailed in Table 1: The first data set contains emails of an email service provider (ESP) collected between 2007 and 2010. The second (Mailinglist) is a collection of emails from publicly available mailing lists augmented by spam emails from Bruce Guenter's spam trap of the same time period. The third corpus (Private) contains newsletters and spam and non-spam emails of the authors. The last corpus is the NIST TREC 2007 spam corpus.

data set	*instances*	*features*	*delivery period*
ESP	169,612	541,713	01/06/2007 - 27/04/2010
Mailinglist	128,117	266,378	01/04/1999 - 31/05/2006
Private	108,178	582,100	01/08/2005 - 31/03/2010
TREC 2007	75,496	214,839	04/08/2007 - 07/06/2007

Table 1: Data sets used in the experiments.

Feature mapping $\phi(x)$ is defined such that email $x \in \mathcal{X}$ is tokenized with the X-tokenizer (Siefkes et al., 2004) and converted into the m-dimensional binary bag-of-word vector $\mathbf{x} := [0, 1]^m$. The value of m is determined by the number of distinct terms in the data set where we have removed all terms which occur only once. For each experiment and each repetition, we then construct the PCA mapping (24) with respect to the corresponding n training emails using the linear kernel $k(\mathbf{x}, \mathbf{x}') := \mathbf{x}^{\mathsf{T}}\mathbf{x}'$ resulting in n-dimensional training instances $\phi_{\mathrm{PCA}}(x_i) \in \mathbb{R}^n$ for $i = 1, \dots, n$. To ensure the convexity as well as the compactness requirement in Assumption 2, we notionally restrict the players' action sets by defining $\phi(\mathcal{X}) := \{\phi_{\mathrm{PCA}}(x) \in \mathbb{R}^n \mid \|\phi_{\mathrm{PCA}}(x)\|_2^2 \leq \kappa\}$ and $\mathcal{W} := \{\mathbf{w} \in \mathbb{R}^n \mid \|\mathbf{w}\|_2^2 \leq \kappa\}$ for some fixed constant κ. Note that by choosing an arbitrarily large κ, the players' action sets become effectively unbounded.

For both algorithms, ILS and EDS, we set $\sigma := 0.001$, $\beta := 0.2$, and $\epsilon := 10^{-14}$. The algorithms are stopped if l exceeds 30 in line 6 of ILS and line 5 in EDS, respectively; in this case, no convergence is achieved. In all experiments, we use the F-measure—that is, the harmonic mean of precision and recall—as evaluation measure and tune all parameters with respect to likelihood. The particular protocol and results of each experiment are detailed in the following sections.

6.1 Convergence

Corollaries 12 (for Nash logistic regression) and 16 (for the Nash support vector machine) specify conditions on the regularization parameters ρ_{-1} and ρ_{+1} under which a unique Nash equilibrium necessarily exists. When this is the case, both the ILS and EDS algorithms will converge on that Nash equilibrium. In the first set of experiments, we study whether repeated restarts of the algorithm converge on the same equilibrium when the bounds in Equations 28 and 34 are satisfied, and when they are violated to increasingly large degrees.

We set $c_{v,i} := \frac{1}{n}$ for $v \in \{-1, +1\}$ and $i = 1, \ldots, n$, such that for $\rho_{-1} > \frac{1}{\rho_{+1}}$ both bounds (Equations 28 and 34) are satisfied. For each value of ρ_{-1} and ρ_{+1} and each of 10 repetitions, we randomly draw 400 emails from the data set and run EDS with a randomly chosen initial solution $(\mathbf{w}^{(0)}, \dot{\mathbf{x}}^{(0)})$ until convergence. We run ILS on the same training set; in each repetition we randomly choose a distinct initial solution, and after each iteration k we compute the Euclidean distance between the EDS solution and the current ILS iterate $\mathbf{w}^{(k)}$.

Figure 1 reports on these average Euclidean distances between distinctly initialized runs. The blue curves ($\rho_{-1} = 2\frac{1}{\rho_{+1}}$) satisfy Equations 28 and 34, the yellow curves ($\rho_{-1} = \frac{1}{\rho_{+1}}$) lie exactly on the boundary; all other curves violate the bounds. Dotted lines show the Euclidean distance between the Nash equilibrium and the solution of logistic regression.

Our findings are as follows. Logistic regression and regular SVM never coincide with the Nash equilibrium—the Euclidean distances lie in the range between 10^{-2} and 2. ILS and EDS always converge to identical equilibria when (28) and (34) are satisfied (blue and yellow curves). The Euclidean distances lie at the threshold of numerical computing accuracy. When Equations 28 and 34 are violated by a factor up to 4 (turquoise and red curves), all repetitions still converge on the same equilibrium, indicating that the equilibrium is either still unique or a secondary equilibrium is unlikely to be found. When the bounds are violated by a factor of 8 or 16 (green and purple curves), then some repetitions of the learning algorithms do not converge or start to converge to distinct equilibria. In the latter case, learner and data generator may attain distinct equilibria and may experience an arbitrarily poor outcome when playing a Nash equilibrium.

6.2 Regularization Parameters

The regularization parameters ρ_v of the players $v \in \{-1, +1\}$ play a major role in the prediction game. The learner's regularizer determines the generalization ability of the predictive model and the data generator's regularizer controls the amount of change in the data generation process. In order to tune these parameter, one would need to have access to labeled data that are governed by the transformed input distribution. In our second

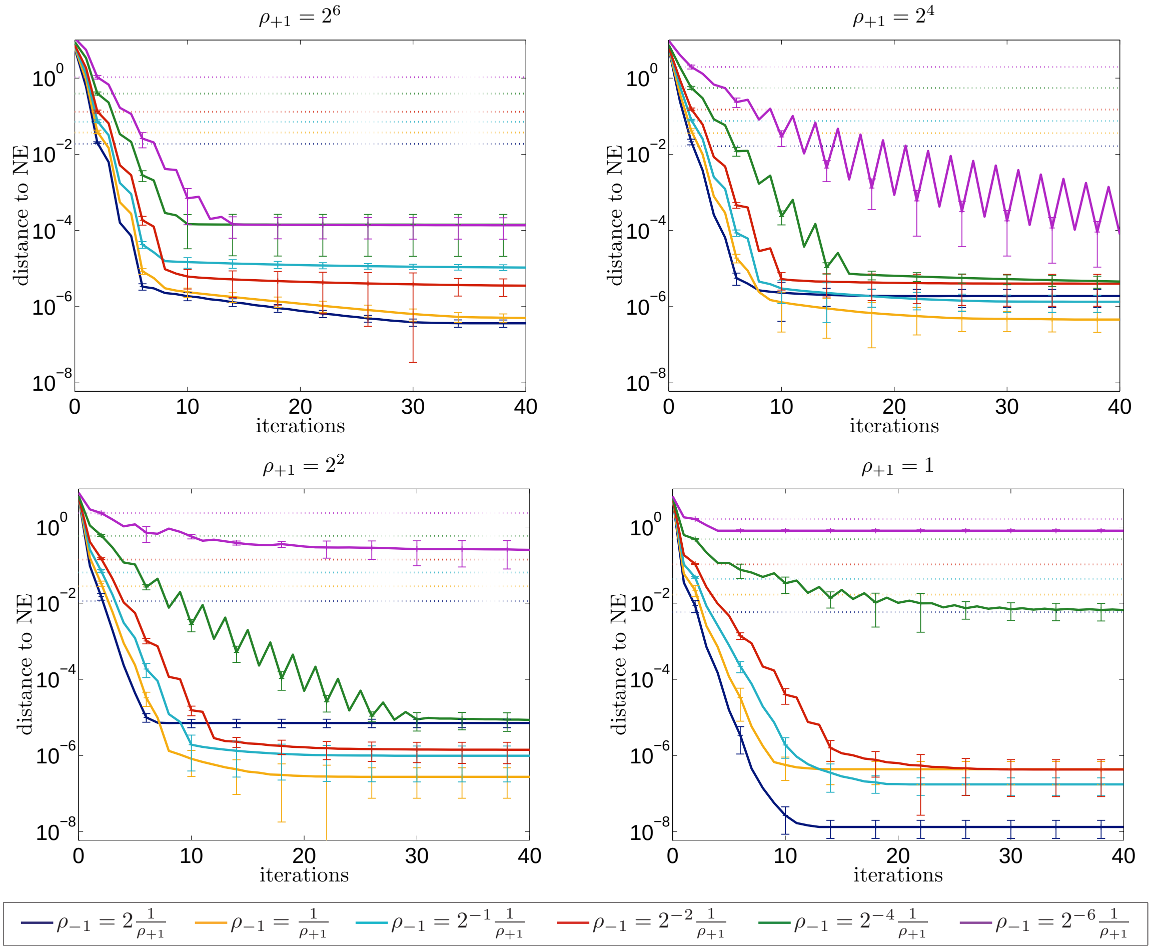

Figure 1: Average Euclidean distance (solid lines) between the EDS solution and the ILS solution at iteration $k = 0, \ldots, 40$ for Nash logistic regression on the ESP corpus. The dotted lines show the distance between the EDS solution and the solution of logistic regression. Error bars indicate standard deviation.

experiment, we will explore to which extend those parameters can be estimated using a portion of the newest training data. Intuitively, the latest training data may more similar to the test data than older training data.

We split the data set into three parts: The 2,000 oldest emails constitute the training portion, we use the next 2,000 emails as hold-out portion on which the parameters are tuned, and the remaining emails are used as test set. We randomly draw 200 spam and 200 non-spam messages from the training portion and draw another subset of 400 emails from the hold-out portion. Both NPG instances are trained on the 400 training emails and evaluated against all emails of the test portion. To tune the parameters, we conduct a grid search maximizing the likelihood on the 400 hold-out emails. We repeat this experiment 10 times for all four data sets and report on the found parameters as well as the "optimal" reference parameters according to the maximal value of F-measure on the test set. Those

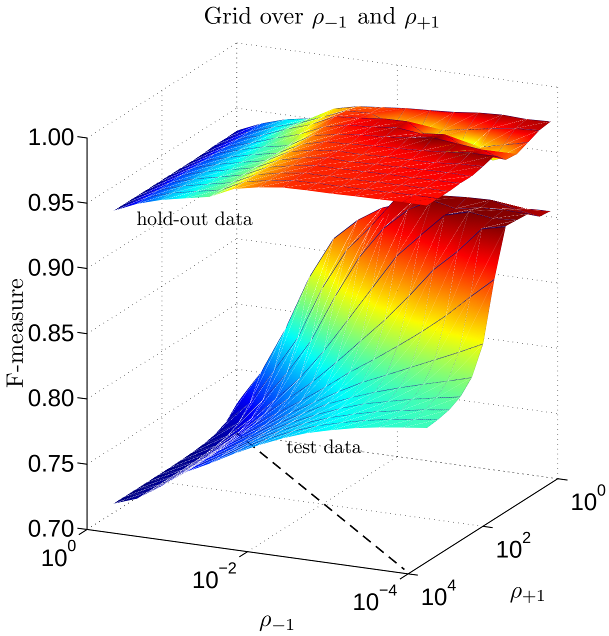

Figure 2a: Performance of NLR on the hold-out and the test data with respect to regularization parameters.

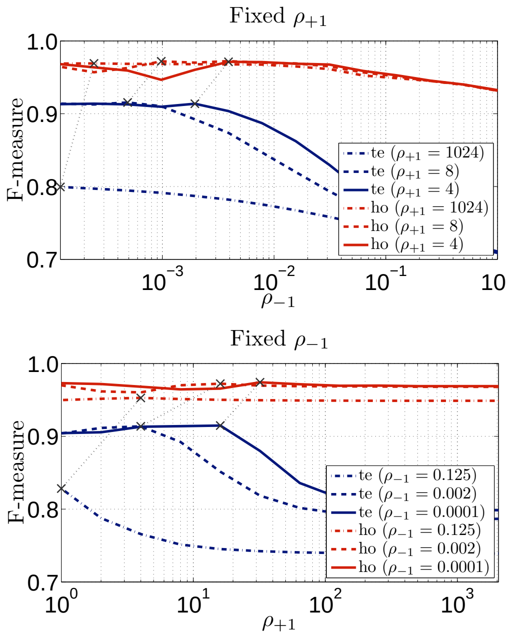

Figure 2b: Performance of NLR on the hold-out data (ho) and the test data (te) for fixed values of ρ_v.

optimal regularization parameters are not used in later experiments. The intuition of the experiment is that the data generation process has already been changed between the oldest and the latest emails. This change may cause a distribution shift which is reflected in the hold-out portion. We expect that one can tune each players' regularization parameter by tuning with respect to this hold-out set.

In Figure 2a we plot the performance of the Nash logistic regression (NLR) on the hold-out and the test data against the regularization parameters ρ_{-1} and ρ_{+1}. The dashed line visualizes the bound in (28) on the regularization parameters for which NLR is guaranteed to possess a unique Nash equilibrium. Figure 2b shows sectional views of the plot in Figure 2a along the ρ_{-1}-axis (upper diagram) and the ρ_{+1}-axis (lower diagram) for several values of ρ_{+1} and ρ_{-1}, respectively. As expected, the effect of the regularization parameters on the test data is much stronger than on the hold-out data.

It turns out that the data generator's ρ_{+1} has almost no impact on the value of F-measure on the hold-out data set (see lower diagram of Figure 2b). Hence, we conclude that estimating ρ_{+1} without access to labeled data from the test distribution or additional knowledge about the data generator is difficult for this application; the latest training data are still too different from the test data. In all remaining experiments and for all data sets we set $\rho_{+1} = 8$ for NLR and $\rho_{+1} = 2$ for NSVM. For those choices the Nash models performed generally best on the hold-out set for a large variety of values of ρ_{-1}. For FDROP the regularization of the data generator's transformation is controlled explicitly by the number K of modifiable attributes per positive instance. We conducted the same experiment for

FDROP resulting in an optimal value of $K = 25$, *i.e.,* the data generator is allowed to remove up to 25 tokens of each spam email of the training data set.

From the upper diagram of Figure 2b we see that estimating ρ_{-1} for any fixed ρ_{+1} seems possible. Even if we slightly overestimate the learner's optimal regularization parameter—to compensate for the distributional difference between the transformed training sample and the marginal shifted hold-out set—the determined value of ρ_{-1} is close to the optimum for all four data sets.

6.3 Evaluation for Nash-Playing Adversary

We evaluate both, a regular classifier trained under the *i.i.d.* assumption and the Nash-equilibrial models against both, an adversary who does not transform the input distribution and an adversary who executes the Nash-equilibrial transformation on the input distribution. Since we cannot be certain that actual spam senders play a Nash equilibrium, we use the following semi-artificial setting.

The learner observes a sample of 200 spam and 200 non-spam emails drawn from the training portion of the data and estimates the Nash-optimal prediction model with parameters $\dot{\mathbf{w}}$; the trivial baseline solution of regularized empirical risk minimization (ERM) is denoted by $\mathbf{w}$. The data generator observes a *distinct* sample D of 200 spam and 200 non-spam messages, also drawn from the training portion, and computes their Nash-optimal response $\dot{D}$.

We again set $c_{v,i} := \frac{1}{n}$ for $v \in \{-1, +1\}$ and $i = 1, \ldots, n$ and study the following four scenarios:

- $(\mathbf{w}, D)$: Both players ignore the presence of an opponent; that is, the learner employs a regular classifier and the sender does not change the data generation process.

- $(\mathbf{w}, \dot{D})$: The learner ignores the presence of an active data generator who changes the data generation process such that D evolves to $\dot{D}$ by playing a Nash strategy.

- $(\dot{\mathbf{w}}, D)$: The learner expects a rational data generator and chooses a Nash-equilibrial prediction model. However, the data generator does not change the input distribution.

- $(\dot{\mathbf{w}}, \dot{D})$: Both players are aware of the opponent and play a Nash-equilibrial action to secure lowest costs.

We repeat this experiment 100 times for all four data sets. Table 2 reports on the average values of F-measure over all repetitions and both NPG instances and corresponding baselines; numbers in boldface indicate significant differences ($\alpha = 0.05$) between the F-measures of $f_{\mathbf{w}}$ and $f_{\dot{\mathbf{w}}}$ for fixed sample D and $\dot{D}$, respectively.

As expected, when the data generator does not alter the input distribution, the regularized empirical risk minimization baselines, logistic regression and the SVM, are generally best. However, the performance of those baselines drops substantially when the data generator plays the Nash-equilibrial action $\dot{D}$. The Nash-optimal prediction models are more robust against this transformation of the input distribution and significantly outperform the reference methods for all four data sets.

	ESP			Mailinglist			Private			TREC 2007		
NLR		$\mathbf{w}$	$\dot{\mathbf{w}}$		$\mathbf{w}$	$\dot{\mathbf{w}}$		$\mathbf{w}$	$\dot{\mathbf{w}}$		$\mathbf{w}$	$\dot{\mathbf{w}}$
vs.	D	**0.957**	0.924	D	**0.987**	0.984	D	**0.961**	0.944	D	**0.980**	**0.979**
LR	$\dot{D}$	0.912	**0.925**	$\dot{D}$	0.958	**0.976**	$\dot{D}$	0.903	**0.912**	$\dot{D}$	0.955	**0.961**

	ESP			Mailinglist			Private			TREC 2007		
NSVM		$\mathbf{w}$	$\dot{\mathbf{w}}$		$\mathbf{w}$	$\dot{\mathbf{w}}$		$\mathbf{w}$	$\dot{\mathbf{w}}$		$\mathbf{w}$	$\dot{\mathbf{w}}$
vs.	D	**0.955**	0.939	D	**0.987**	0.985	D	**0.961**	0.957	D	0.979	**0.981**
SVM	$\dot{D}$	0.928	**0.939**	$\dot{D}$	0.961	**0.976**	$\dot{D}$	0.932	**0.936**	$\dot{D}$	0.960	**0.968**

Table 2: Nash predictor and regular classifier against passive and Nash-equilibrial data generator.

6.4 Case Study on Email Spam Filtering

To study the performance of the Nash prediction models and the baselines for email spam filtering, we evaluate all methods *into the future* by processing the test set in chronological order. The test portion of each data set is split into 20 chronologically sorted disjoint subsets. We average the value of F-measure on each of those subsets over the 20 models (trained on different samples drawn from the training portion) for each method and perform a paired t-test.

Figure 3 shows that, for all data sets, the NPG instances outperform logistic regression and the SVM that do not explicitly factor the adversary into the optimization criterion. Especially for the ESP corpus, the Nash logistic regression (NLR) and the Nash support vector machine (NSVM) are superior. On the TREC 2007 data set, the methods behave comparably with a slight advantage for the Nash support vector machine. The period over which the TREC 2007 data have been collected is very short; we believe that the training and test instances are governed by nearly identical distributions. Consequently, for this data set, the game-theoretic models do not gain a significant advantage over logistic regression and the SVM that assume *i.i.d.* samples.

method vs. *method*	*SVM*	*LR*	*FDROP*	*NLR*	*NSVM*
SVM	0:*0*	40:*2*	13:*16*	8:*57*	2:*65*
LR	2:*40*	0:*0*	7:*22*	5:*59*	2:*71*
FDROP	16:*13*	22:*7*	0:*0*	4:*22*	3:*24*
NLR	57:*8*	59:*5*	22:*4*	0:*0*	22:*30*
NSVM	65:*2*	71:*2*	24:*3*	30:*22*	0:*0*

Table 3: Results of paired t-test over all corpora: Number of trials in which each method (row) has significantly outperformed each other *method (column)* vs. number of times it *was outperformed.*

Table 3 shows aggregated results over all four data sets. For each point in each of the diagrams of Figure 3, we conduct a pairwise comparison of all methods based on a paired t-test at a confidence level of $\alpha = 0.05$. When a difference is significant, we count this

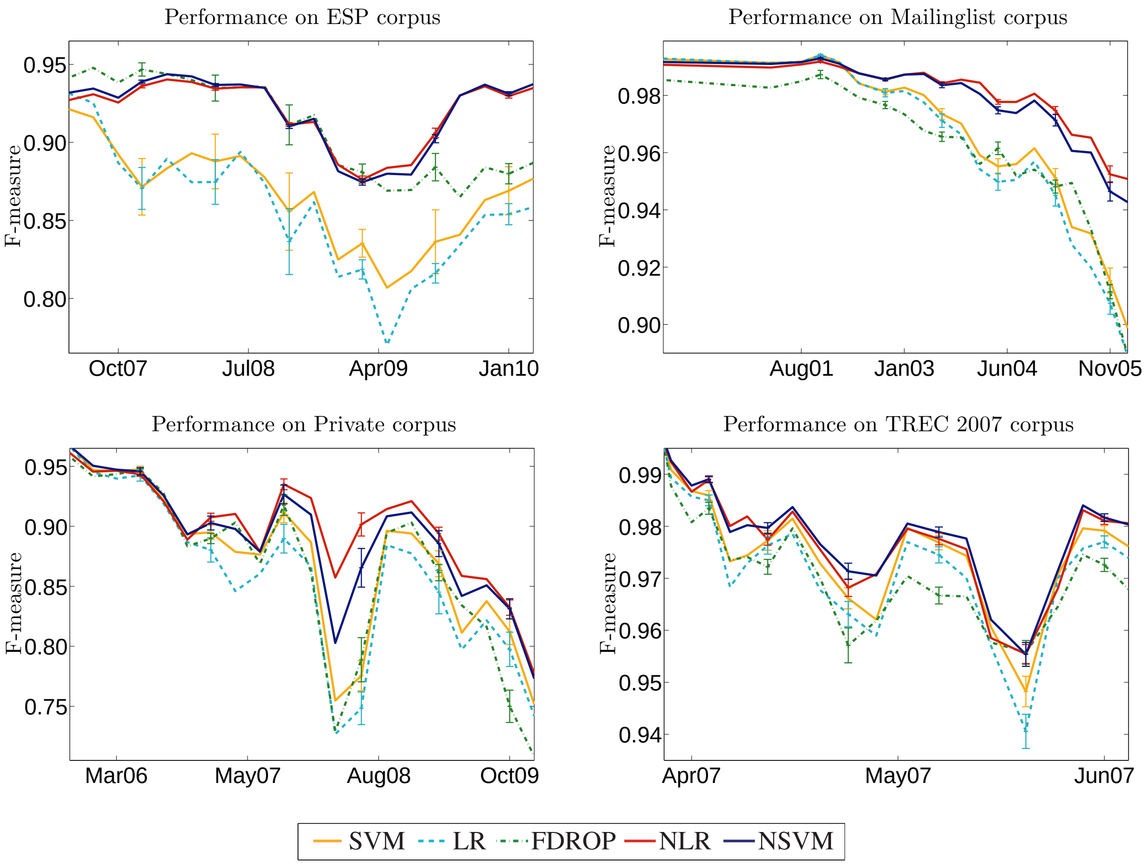

Figure 3: Value of F-measure of predictive models. Error bars indicate standard errors.

as a win for the method that achieves a higher value of F-measure. Each line of Table 3 details the wins and, set in italics, the *losses* of one method against all other methods. The Nash logistic regression and the Nash support vector machine have more wins than they have losses against each of the other methods. The ranking continues with FDROP, the regular SVM, and logistic regression which loses more frequently than it wins against all other methods.

6.5 Efficiency versus Effectiveness

To assess the predictive performance as well as the execution time as a function of the sample size, we train the baselines and the two NPG instances for a varying number of training examples. We report on the results for the ESP data set in Figure 4. The game-theoretic models significantly outperform the trivial baseline methods logistic regression and the SVM, especially for small data sets. However, this comes at the price of considerably higher computational cost. The ILS algorithm requires in general only a couple of iterations to converge; however in each iteration several optimization problems have to be solved such that the total execution time is up to a factor 150 larger than that of the corresponding ERM baseline. In contrast to the ILS algorithm, a single iteration of the EDS algorithm does not require solving nested optimization problems. However, the execution time of the EDS

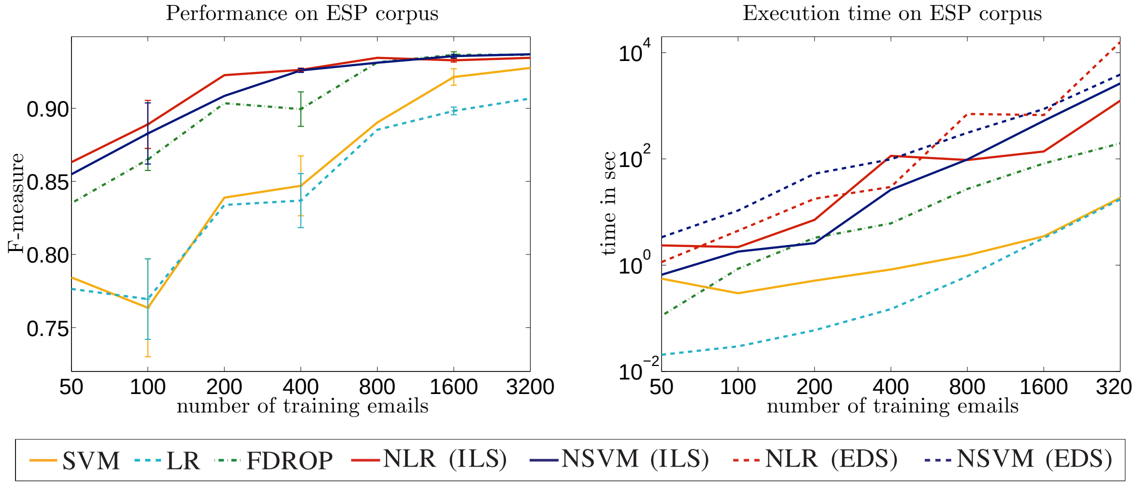

Figure 4: Predictive performance (left) and execution time (right) for varying sizes of the training data set.

algorithm is still higher as it often requires several thousand iterations to fully converge. For larger data sets, the discrepancy in predictive performance between game-theoretic models and *i.i.d.* baseline decreases. Regarding the whether ILS or EDS is faster at solving the optimization problems that lead to the Nash equilibria our results are not conclusive. We conclude that the benefit of the NPG prediction models over the classification baseline is greatest for small to medium sample sizes.

6.6 Nash-Equilibrial Transformation

In contrast to FDROP, the Nash models allow the data generator to modify non-spam emails. However in practice most senders of legitimate messages do not deliberately change their writing behavior in order to bypass spam filters, perhaps with the exception of senders of newsletters who must be careful not to trigger filtering mechanisms. In a final experiment, we want to study whether the Nash model reflects this aspect of reality, and how the data generator's regularizer effects this transformation.

The training portion contains again $n_{+1} = 200$ spam and $n_{-1} = 200$ non-spam instances randomly chosen from the oldest $4,000$ emails. We determine the Nash equilibrium and measure the number of additions and deletions to spam and non-spam emails in $\dot{D}$:

$$\begin{array}{rcl|rcl}
\Delta_{-1}^{\text{add}} & := & \frac{1}{n_{-1}} \sum\limits_{i:y_i=-1} \sum\limits_{j=1}^{m} \max(0, \dot{\mathbf{x}}_{i,j} - \mathbf{x}_{i,j}) & \Delta_{+1}^{\text{add}} & := & \frac{1}{n_{+1}} \sum\limits_{i:y_i=+1} \sum\limits_{j=1}^{m} \max(0, \dot{\mathbf{x}}_{i,j} - \mathbf{x}_{i,j}) \\
\Delta_{-1}^{\text{del}} & := & \frac{1}{n_{-1}} \sum\limits_{i:y_i=-1} \sum\limits_{j=1}^{m} \max(0, \mathbf{x}_{i,j} - \dot{\mathbf{x}}_{i,j}) & \Delta_{+1}^{\text{del}} & := & \frac{1}{n_{+1}} \sum\limits_{i:y_i=+1} \sum\limits_{j=1}^{m} \max(0, \mathbf{x}_{i,j} - \dot{\mathbf{x}}_{i,j})
\end{array}$$

where $\mathbf{x}_{i,j}$ indicates the presence of token j in the i-th training email, that is, Δ_v^{add} and Δ_v^{del} denote the average number of word additions and deletions per spam and non-spam email performed by the sender.

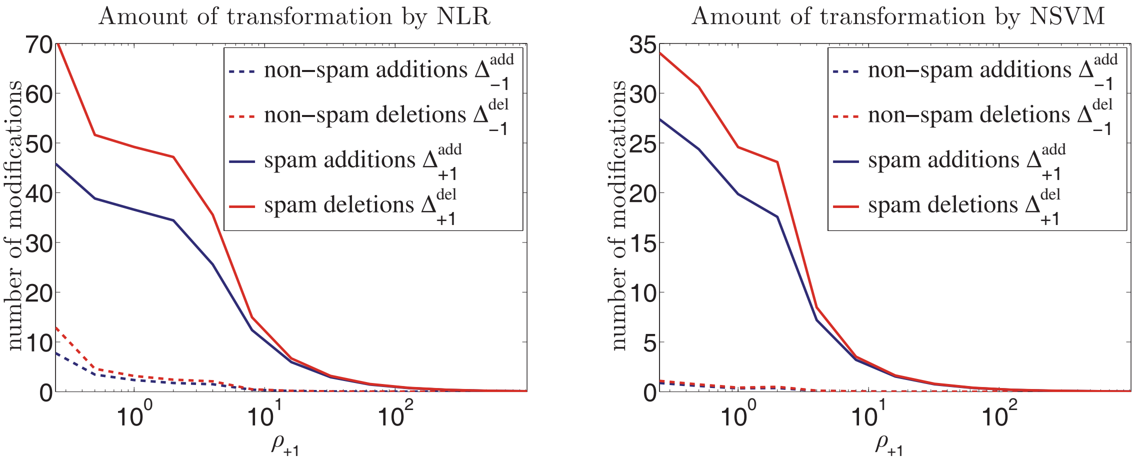

Figure 5: Average number of additions and deletions per spam/non-spam email for NLR (left) and NSVM (right) with respect to the adversary's regularization parameter ρ_{+1} for fixed $\rho_{-1} = n^{-1}$.

Figure 5 shows the number of additions and deletions of the Nash transformation as a function of the adversary's regularization parameter for the ESP data set. Table 4 reports on the average number of word additions and deletions for all data sets. For FDROP, we set the number of possible deletions to $K = 25$.

ESP

game	*non-spam*		*spam*	
model	*add*	*del*	*add*	*del*
FDROP	0.0	0.0	0.0	24.8
NLR	0.7	1.0	22.5	31.2
NSVM	0.4	0.5	17.9	23.8

Mailinglist

game	*non-spam*		*spam*	
model	*add*	*del*	*add*	*del*
FDROP	0.0	0.0	0.0	23.9
NLR	0.3	0.4	8.6	10.9
NSVM	0.3	0.3	6.9	8.4

Private

game	*non-spam*		*spam*	
model	*add*	*del*	*add*	*del*
FDROP	0.0	0.0	0.0	24.2
NLR	0.4	0.2	24.3	11.2
NSVM	0.1	0.1	15.6	7.3

TREC 2007

game	*non-spam*		*spam*	
model	*add*	*del*	*add*	*del*
FDROP	0.0	0.0	0.0	24.7
NLR	0.2	0.2	15.0	11.4
NSVM	0.2	0.1	11.1	8.4

Table 4: Average number of word additions and deletions per training email.

The Nash-equilibrial transformation imposes almost no changes on any non-spam email; the number of modifications declines as the regularization parameter grows (see Figure 5). We observe for all data sets that even if the total amount of transformation differs for NLR and NSVM, both instances behave similarly insofar as the number of word additions and deletions continues to grow when the adversary's regularizer decreases.

7. Conclusion

We studied prediction games in which a learner and a data generator have conflicting but not necessarily directly antagonistic cost functions. We focused on static games in which learner and data generator have to commit simultaneously to a predictive model and a transformation on the input distribution, respectively. The cost-minimizing action of each player depends on the opponent's move; in the absence of information about the opponent's move, players may choose to play a Nash equilibrium which constitutes a cost-minimizing move for each player *if* the other player follows the equilibrium as well. Because a combination of actions from distinct equilibria may lead to arbitrarily high costs for either player, we have studied conditions under which a prediction game can be guaranteed to possess a unique Nash equilibrium. Lemma 1 identifies conditions under which at least one equilibrium exists and Theorem 8 elaborates on when this equilibrium is unique. We propose an inexact linesearch approach and a modified extragradient approach to identifying this unique equilibrium. Empirically, both approaches turned out to perform quite similarly.

We derived Nash logistic regression and Nash support vector machine models, and derived kernelized versions of these methods. Corollaries 12 and 16 specialize Theorem 8 and elaborate conditions on the player's regularization parameters under which the Nash logistic regression and the support vector machine converge on a unique Nash equilibrium. Empirically, we find that both methods identify unique Nash equilibria when the bounds laid out in Corollaries 12 and 16 are satisfied or violated by a factor of up to 4. From our experiment on several email corpora we conclude that Nash logistic regression and the support vector machine outperform their *i.i.d.* baselines and FDROP for the problem of classifying future emails based on training data from the past.

Acknowledgments

This work was partially supported by the German Science Foundation DFG under grant SCHE540/12-1 and by STRATO AG. We also thank Niels Landwehr and Christoph Sawade for constructive comments and suggestions.

References

Tamer Basar and Geert J. Olsder. *Dynamic Noncooperative Game Theory.* Society for Industrial and Applied Mathematics, 1999.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

Michael Brückner and Tobias Scheffer. Stackelberg games for adversarial prediction problems. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), San Diego, CA, USA*. ACM, 2011.

Michael Brückner and Tobias Scheffer. Nash equilibria of static prediction games. In *Advances in Neural Information Processing Systems.* MIT Press, 2009.

Ofer Dekel and Ohad Shamir. Learning to classify with missing and corrupted features. In *Proceedings of the International Conference on Machine Learning.* ACM, 2008.

Ofer Dekel, Ohad Shamir, and Lin Xiao. Learning to classify with missing and corrupted features. *Machine Learning*, 81(2):149–178, 2010.

Carl Geiger and Christian Kanzow. *Theorie und Numerik restringierter Optimierungsaufgaben*. Springer, 1999.

Laurent El Ghaoui, Gert R. G. Lanckriet, and Georges Natsoulis. Robust classification with interval data. Technical Report UCB/CSD-03-1279, EECS Department, University of California, Berkeley, 2003.

Amir Globerson and Sam T. Roweis. Nightmare at test time: Robust learning by feature deletion. In *Proceedings of the International Conference on Machine Learning*. ACM, 2006.

Amir Globerson, Choon Hui Teo, Alex J. Smola, and Sam T. Roweis. *Dataset Shift in Machine Learning*, chapter An adversarial view of covariate shift and a minimax approach, pages 179–198. MIT Press, 2009.

Patrick T. Harker and Jong-Shi Pang. Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications. *Mathematical Programming*, 48(2):161–220, 1990.

Roger A. Horn and Charles R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, Cambridge, 1991.

Gert R. G. Lanckriet, Laurent El Ghaoui, Chiranjib Bhattacharyya, and Michael I. Jordan. A robust minimax approach to classification. *Journal of Machine Learning Research*, 3: 555–582, 2002.

J. B. Rosen. Existence and uniqueness of equilibrium points for concave n-person games. *Econometrica*, 33(3):520–534, 1965.

Bernhard Schölkopf and Alex J. Smola. *Learning with Kernels*. The MIT Press, Cambridge, MA, 2002.

Bernhard Schölkopf, Ralf Herbrich, and Alex J. Smola. A generalized representer theorem. In *COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers*, 2001.

Christian Siefkes, Fidelis Assis, Shalendra Chhabra, and William S. Yerazunis. Combining Winnow and orthogonal sparse bigrams for incremental spam filtering. In *Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD)*, volume 3202 of *Lecture Notes in Artificial Intelligence*, pages 410–421. Springer, 2004.

Choon Hui Teo, Amir Globerson, Sam T. Roweis, and Alex J. Smola. Convex learning with invariances. In *Advances in Neural Information Processing Systems*. MIT Press, 2007.

Anna von Heusinger and Christian Kanzow. Relaxation methods for generalized Nash equilibrium problems with inexact line search. *Journal of Optimization Theory and Applications*, 143(1):159–183, 2009.