SUPPORT VECTOR MACHINE CLASSIFIERS BY NON-EUCLIDEAN MARGINS

YING LIN AND QI YE*

School of Mathematical Sciences
South China Normal University
Guangzhou, Guangdong 510631, China

ABSTRACT. In this article, the classical support vector machine (SVM) classifiers are generalized by the non-Euclidean margins. We first extend the linear models of the SVM classifiers by the non-Euclidean margins including the theorems and algorithms of the SVM classifiers by the hard margins and the soft margins. Specially, the SVM classifiers by the ∞-norm margins can be solved by the 1-norm optimization with sparsity. Next, we show that the non-linear models of the SVM classifiers by the q-norm margins can be equivalently transferred to the SVM in the p-norm reproducing kernel Banach spaces given by the hinge loss, where 1/p + 1/q = 1. Finally, we illustrate the numerical examples of artificial data and real data to compare the different algorithms of the SVM classifiers by the ∞-norm margin.

1. Introduction. The support vector machine (SVM) classifier, also known as the maximal margin classifier, is a successful method for binary classifications [1, 4]. The classical SVM classifier is constructed to maximize the Euclidean margin defined by the 2-norm distance of the gap of two classes based on the decision hyperplane. In this article, to make use of the information behind data and possess more geometrical structures including the sparsity, we will extend the SVM classifier by the non-Euclidean margin including the linear models and the nonlinear models. The use of non-Euclidean margin rather than Euclidean margin allows the data to be in a Banach space instead of a Hilbert space. In section 2, we verify the algorithms and theorems of the linear SVM classifiers by the non-Euclidean margins such as the maximization of the hard margins and the soft margins. Specially, the SVM classifier by ∞-norm margin can be equivalently transferred to the 1-norm SVM with sparsity similar as in [3, 5, 19, 21, 22]. In Section 3, we show that the nonlinear SVM classifiers by the q-norm margins can also be equivalently transferred to the SVM classifiers in the p-norm reproducing kernel Banach spaces (RKBSs) given by the hinge loss in [18], where 1/p + 1/q = 1. In section 4, we further use the artificial data and real data to test the SVM classifiers by the ∞-norm margin.

2020 Mathematics Subject Classification. Primary: 46E22, 62G08; Secondary: 68Q32.
Key words and phrases. Support vector machine classifier, non-euclidean margin, reproducing kernel Banach space.

The first author is supported by the Innovation Project of Graduate School of South China Normal University. The second author is supported by the Natural Science Foundation of Guangdong Province (2019A1515011995,2020B1515310013) and the National Natural Science Foundation of China (11931003).

* Corresponding author: Qi Ye.
2. Linear support vector machine classifiers by non-Euclidean margins.

2.1. Maximal margin classification. Now we review the basic ideas of the binary classifications by the maximal margins in [17]. In the binary classification, we have a training data set $D := \{(x_i, y_i)\}_{i=1}^N$ composed of input data points $x_1, x_2, \ldots, x_N \in \mathbb{R}^d$ and output data labels $y_1, y_2, \ldots, y_N \in \{\pm 1\}$. We will construct a hyperplane $H := \{x \in \mathbb{R}^d : x^\top \omega + b = 0\}$ to classify the data such that $\text{sign}(x_i^\top \omega + b) > 0$ if $y_i = 1$, and $\text{sign}(x_i^\top \omega + b) < 0$ if $y_i = -1$, for all $i = 1, 2, \ldots, N$.

At first, we assume that $D$ is linearly separable, that is, the separating hyperplane $H$ exists. By the properties of $H$, each class of data points lies on each side of $H$, respectively. To avoid over-fitting, the geometrical feature of $H$ is that the Euclidean distances from the data points to $H$ are needed to be large enough. For fairness of both classes, the decision hyperplane $H$ is chosen in the middle of each class so that the minimal distances from $H$ to both classes are equal. By the classical maximal margin classification, the Euclidean margin is the Euclidean distance of the gap of two classes based on the decision hyperplane, for example, the 2-norm margin in Figure 1a. By the different geometrical structures of the practical applications, we will extend the Euclidean margin to the non-Euclidean margin, that is, the generalization from Euclidean distance to non-Euclidean distance, for example, the $\infty$-norm margin in Figure 1b. Similarly the SVM classifiers by non-Euclidean margins are constructed to maximize the non-Euclidean margins.

The linear model is to construct the decision hyperplane to separate the data by maximizing non-Euclidean margins. Let $\| \cdot \|$ be a norm defined on $\mathbb{R}^d$ and $\| \cdot \|_*$ be its dual norm defined on $\mathbb{R}^d$ such that their dual bilinear product is given by the vector product. By Theorem A.1, we have

$$\text{dist}(x, H) = \frac{|x^\top \omega + b|}{\|\omega\|_*} = \frac{\text{sign}(x^\top \omega + b)(x^\top \omega + b)}{\|\omega\|_*}, \text{ for all } x \notin H,$$
where \( \text{dist}(x, H) \) refers to the distance from a point \( x \) to a hyperplane \( H \). The linear separability of data set guarantees that \( y_i(x_i^\top \omega + b) \geq 0 \) and so \( y_i = \text{sign}(x_i^\top \omega + b) \) for all \( i = 1, 2, \ldots, N \). If \( \|\omega\|_* = 1 \), then we also have

\[
\text{dist}(x_i, H) = y_i(x_i^\top \omega + b), \quad \text{for all } i = 1, 2, \ldots, N.
\]

The idea of maximal margin classification is to maximize the minimum of the distances from \( x_i \) to \( H \) for all \( i = 1, 2, \ldots, N \). Since \( \{x_i\}_{i=1}^N \) is a finite set, there exists a positive \( M \) such that \( \text{dist}(x_i, H) \geq M \) for all \( i \). Thus, the maximal margin classification can be written as a constrained optimization

\[
\max_{\omega \in \mathbb{R}^d, \|\omega\|_* = 1, b \in \mathbb{R}} \quad M
\]
subject to \( y_i(x_i^\top \omega + b) \geq M, \quad i = 1, 2, \ldots, N. \) \hfill (1)

To solve Optimization (1), we transfer the norm constraint \( \|\omega\|_* = 1 \) with \( \|\omega\|_* = 1/M \). Thus, Optimization (1) can be transferred to the constrained optimization

\[
\min_{\omega \in \mathbb{R}^d, b \in \mathbb{R}} \quad \|\omega\|_*
\]
subject to \( y_i(x_i^\top \omega + b) \geq 1, \quad i = 1, 2, \ldots, N. \) \hfill (2)

The objective function of Optimization (2) is a convex function and the constraints are linear inequalities. This assures that Optimization (2) is a convex problem. Next, we show the existence and conditional uniqueness of the solutions to Optimization (2).

**Proposition 1.** If \( D \) is linearly separable, then the solution set to Optimization (2) is non-empty, compact, and convex. In particular, if the norm \( \| \cdot \|_* \) is further strictly convex, Optimization (2) has one and only one solution.

**Proof.** Since \( D \) is linearly separable, there exists a \( \hat{\omega} := (\hat{\omega}, \hat{b}) \in \mathbb{R}^{d+1} \) such that \( y_i(x_i^\top \hat{\omega} + \hat{b}) \geq 1 \) for all \( i \). Thus the feasible solution set to Optimization (2) is non-empty, we denote it by \( \Xi \). Since the constraints are linear inequalities, \( \Xi \) is convex. We can equivalently transfer (2) to

\[
\min_{u = (u, \hat{b}) \in \mathbb{R}^{d+1}} \|\omega\|_* + \delta(u|\Xi), \hfill (3)
\]

where \( \delta(u|\Xi) = 0 \) if \( u \in \Xi \) and \( +\infty \) otherwise. It is worthy pointing out that after introducing \( \delta(\cdot|\cdot) \), the range is now the extended real number line \( \mathbb{R} := \mathbb{R} \cup \{-\infty, -\infty\} \) rather than \( \mathbb{R} \).

To complete the proofs, we only need to discuss the solution set to (3). By Theorem 27.2 in [12], we only need to show that the objective function \( F(u) := \|\omega\|_* + \delta(u|\Xi) \) is a convex closed proper function which has no direction of recession, then the solution set to (2) is non-empty, closed, bounded (so compact) and convex.

We choose \( u_1 := (\omega_1, b_1), u_2 := (\omega_2, b_2) \in \mathbb{R}^{d+1}, \) and \( \lambda \in [0, 1], \) there are two cases to show the convexity. One is that at least one of \( u_1 \) and \( u_2 \) is not in \( \Xi \), then \( \lambda F(u_1) + (1 - \lambda)F(u_2) = \infty \) (by properties and assumes in \( \mathbb{R} \)). In this case, \( F(\lambda u_1 + (1 - \lambda)u_2) \leq \lambda F(u_1) + (1 - \lambda)F(u_2) = \infty. \) The other case is that both \( u_1 \) and \( u_2 \) are in \( \Xi \). In this case, since \( \Xi \) is convex, we have \( F(u_1) = \|\omega_1\|_* , F(u_2) = \|\omega_2\|_* , F(\lambda u_1 + (1 - \lambda)u_2) = \|\lambda \omega_1 + (1 - \lambda)\omega_2\|_*). \) Then

\[
F(\lambda u_1 + (1 - \lambda)u_2) = \|\lambda \omega_1 + (1 - \lambda)\omega_2\|_* \leq \lambda \|\omega_1\|_* + (1 - \lambda)\|\omega_2\|_* = \lambda F(u_1) + (1 - \lambda)F(u_2).
\]

That is, \( F(u) \) is convex.
Since $F(\hat{u})$ is finite and $F(u) \geq 0 > -\infty$, $F(u)$ is proper. By [12, p 52], a convex proper function is closed if it is lower semi-continuous. For each $u_0 := (\omega_0, b_0) \in \mathbb{R}^{d+1}$ and any sequence $\{u_i := (\omega_i, b_i)\}_{i=1}^{\infty}$ converges to $u_0$, there are two cases. One is that $\lim_{i \to \infty} F(u_i) < \infty$, then $F(u_0) = \lim_{i \to \infty} F(u_i)$. The other is that $\lim_{i \to \infty} F(u_i) = \infty$, then $F(u_0) \leq \lim_{i \to \infty} F(u_i)$. So $F(u)$ is semi-continuous then closed. Now $F(u)$ is a convex closed proper function.

Next we show that $F(u)$ has no direction of recession. It is equivalent to show that for any $\bar{u} := (\bar{\omega}, \bar{b}) \neq 0$, there exists a $u_0 := (\omega_0, b_0)$ such that $g(\lambda) := F(u_0 + \lambda \bar{u})$ is not a non-increasing function. That is, there are some $\lambda_0 > \lambda_1$ such that $g(\lambda_0) > g(\lambda_1)$. Given any $\bar{u} \neq 0$, choose a $u_0$ such that there exist $\lambda_0 > \lambda_1$ such that $u_0 + \lambda_0 \bar{u} \in \Xi$ and $u_0 + \lambda_1 \bar{u} \in \Xi$. Then $g(\lambda_0) = \|\omega_0 + \lambda_0 \bar{\omega}\|_*$. Given any $\bar{u} \neq 0$, choose a $u_0$ such that there exist $\lambda_0 > \lambda_1$ such that $u_0 + \lambda_0 \bar{u} \in \Xi$ and $u_0 + \lambda_1 \bar{u} \in \Xi$. Then $g(\lambda_0) = \|\omega_0 + \lambda_0 \bar{\omega}\|_*$. We have

$$
g(\lambda_1) - g(\lambda_0) = \|\omega_0 + \lambda_1 \bar{\omega}\|_* - \|\omega_0 + \lambda_0 \bar{\omega}\|_*
= \|\omega_0 + \lambda_0 \bar{\omega} + (\lambda_1 - \lambda_0)\bar{\omega}\|_* - \|\omega_0 + \lambda_0 \bar{\omega}\|_*
\leq \|\omega_0 + \lambda_0 \bar{\omega}\|_* + (\lambda_1 - \lambda_0)\|\bar{\omega}\|_* - \|\omega_0 + \lambda_0 \bar{\omega}\|_*
= (\lambda_1 - \lambda_0)\|\bar{\omega}\|_* < 0.
$$

Therefore, $F(u)$ has no direction of recession.

We can now directly obtain that the solution set to optimization (2) is non-empty, compact and convex.

Finally, if $\|\cdot\|_*$ is strictly convex, choose any distinct $u_1 := (\omega_1, b_1)$, $u_2 := (\omega_2, b_2)$ in the solution set and denote the minimal of Optimization (2) by $\tau$. Since the solution set is convex, $u_1/2 + u_2/2$ is also in the solution set. We have

$$
\|\omega_1/2 + \omega_2/2\|_* < \|\omega_1\|_*/2 + \|\omega_2\|_*/2 = \tau.
$$

This indicates that $\|\omega_1/2 + \omega_2/2\|_*$ is the new minimal of Optimization (2), which is a contradiction. Therefore, $u_1 = u_2$. The proof of the uniqueness is completed. \qed

2.2. The linear support vector machine classifiers. In real-world applications, because of the noise, the classes overlaps, the data set may not be linearly separable. If $D$ is not linearly separable, Optimization (2) has no solution. Same as the soft margin of the classical SVM classifiers, one way to deal with the overlap is to still maximize the non-Euclidean margin, but we allow some points to be on the wrong side of the decision boundary.

Define the slack variables $\xi = (\xi_i)_{i=1}^{N}$. Figure 2 shows the difference between hard margin and soft margin.

There are two natural ways to modify the constraints in Optimization (1) including $y_i(x_i^\top \omega + b) \geq M - \xi_i$ or $y_i(x_i^\top \omega + b) \geq M(1 - \xi_i)$, and $\xi_i \geq 0$, for $i = 1, 2, \ldots, N$. The first choice seems more natural, but it results in a non-convex optimization problem, while the second is convex. That is because if we use the first choice, the constraints in Optimization (2) will finally become $y_i(x_i^\top \omega + b) \geq 1 - \xi_i/\|\omega\|_*$, which leads to the non-convexity. Thus, we can reform the constraints in Optimization (2) to $y_i(x_i^\top \omega + b) \geq 1 - \xi_i$ and $\xi_i \geq 0$ for $i = 1, 2, \ldots, N$. The $\xi_i$ is 0 in case of no error. If a pattern falls within the margin, the $\xi_i$ for this pattern is positive. Hence, by minimizing the $\sum_{i=1}^{N} \xi_i$, we minimize the total proportional amount by which predictions fall on the wrong side of their margin. As Optimization (2), we can drop the norm constraint on $\omega$ and define $M = 1/\|\omega\|_*$. To maximize the margins
and minimize the mistakes, we rewrite Optimization (1) in the form as

$$\min_{\omega \in \mathbb{R}^d, b \in \mathbb{R}, \xi \in \mathbb{R}^N} \|\omega\|_* + C \sum_{i=1}^N \xi_i$$
subject to
$$y_i(x_i^T \omega + b) \geq 1 - \xi_i, \quad i = 1, 2, \ldots, N,$$
$$\xi_i \geq 0, \quad i = 1, 2, \ldots, N,$$

where the “cost” parameter $C \geq 0$ decides the trade-off between maximizing the margin and minimizing the mistakes. When $C$ is large enough, the soft-margin SVM classifiers behave as the hard-margin SVM classifiers. The linearly separable case corresponds to $C \to \infty$, that is, every $\xi_i$ is constrained to be 0. Now the target variable has changed from $(\omega, b)$ to $(\omega, b, \xi)$.

Remark 1. It has another way to the noises, that is, a constant $\vartheta \geq 0$ is used to control the misclassifications by a constraint $\sum_{i=1}^N \xi_i \leq \vartheta$. As Optimization (2), we can drop the norm constraint on $\omega$ and define $M = 1/\|\omega\|_*$. Then, we have the constrained optimization

$$\min_{\omega \in \mathbb{R}^d, b \in \mathbb{R}} \|\omega\|_*$$
subjected to
$$y_i(x_i^T \omega + b) \geq 1 - \xi_i, \quad i = 1, 2, \ldots, N,$$
$$\xi_i \geq 0, \quad i = 1, 2, \ldots, N,$$
$$\sum_{i=1}^N \xi_i \leq \vartheta.$$

By Theorem 28.1 in [12], using Lagrange multipliers, if $C$ is the Kuhn-Tucker coefficient corresponding to the constraint $\sum_{i=1}^N \xi_i - \vartheta \leq 0$, we can obtain the solution set for Optimization (5) by solving Optimization (4). Since Optimization (4) is computationally easier and it gains more attention than Optimization (5), in this paper we only discuss Optimization (4).

The objective function of Optimization (4) is actually a norm plus a linear sum and the constraints are still linear inequalities. Thus, Optimization (4) is convex. We will show the solution set to Optimization (4) is also non-empty and convex.
Proposition 2. The solution set to Optimization (4) is non-empty, compact and convex. In particular, if \( \| \cdot \|_* \) is further strictly convex, Optimization (4) has one and only one solution.

Proof. Without loss of generality, we can assume that \( C > 0 \). The case of \( C = 0 \) leads to the trivial case that the minimal is 0 and \( \text{convex} \).

By the construction of Optimization (4), we can randomly select \((\tilde{\omega}, \tilde{b}) \in \mathbb{R}^{d+1}\) and let \( \xi_i = \max\{1 - y_i(x_i^\top \tilde{\omega} + \tilde{b}), 0\} \), then \( \tilde{v} := (\tilde{\omega}, \tilde{b}, \xi) \) is a feasible solution to Optimization (4). Thus, the feasible solution set to Optimization (4) is non-empty, we denote it by \( \Xi \). Since the constraints are all linear inequalities, \( \Xi \) is convex. We can equivalently transfer (4) to

\[
\min_{v=(\omega, b, \xi) \in \mathbb{R}^{d+N+1}} \|\omega\|_* + C \sum_{i=1}^{N} \xi_i + \delta(v|\Xi).
\]  

(6)

Similarly, we are now working in \( \tilde{\mathbb{R}} \).

To complete the proofs, we only need to discuss the solution set to Optimization (6). By Theorem 27.2 in [12], we only need to show that the objective function \( F(v) := \|\omega\|_* + C \sum_{i=1}^{N} \xi_i + \delta(v|\Xi) \) is a convex closed proper function which has no direction of recession, then the solution set to (4) is non-empty, closed, bounded (so compact) and convex.

We choose \( v_1 := (\omega_1, b_1, \xi_1), v_2 := (\omega_2, b_2, \xi_2) \in \mathbb{R}^{d+N+1} \), where we denote \( \xi_{1,i}, \xi_{2,i} \) by the \( i \)th element of \( \xi_1, \xi_2 \), by the \( i \)th element of \( \xi_2 \), and \( \lambda \in [0, 1] \), there are two cases to show the convexity. One is that at least one of \( v_1 \) and \( v_2 \) is not in \( \Xi \), then \( \lambda F(v_1) + (1 - \lambda) F(v_2) = \infty \) (by properties and assumes in \( \tilde{\mathbb{R}} \)). In this case, \( F(\lambda v_1 + (1 - \lambda) v_2) \leq \lambda F(v_1) + (1 - \lambda) F(v_2) = \infty \). The other case is that both \( v_1 \) and \( v_2 \) are in \( \Xi \). In this case, since \( \Xi \) is convex, we have \( F(v_1) = \|\omega_1\|_* + C \sum_{i=1}^{N} \xi_{1,i}, F(v_2) = \|\omega_2\|_* + C \sum_{i=1}^{N} \xi_{2,i}, F(\lambda v_1 + (1 - \lambda) v_2) = \|\lambda \omega_1 + (1 - \lambda) \omega_2\|_* + C \sum_{i=1}^{N} (\xi_{1,i} + (1 - \lambda) \xi_{2,i}) \). Then

\[
F(\lambda v_1 + (1 - \lambda) v_2) \leq \lambda \|\omega_1\|_* + (1 - \lambda) \|\omega_2\|_* + C \lambda \sum_{i=1}^{N} \xi_{1,i} + C(1 - \lambda) \sum_{i=1}^{N} \xi_{2,i}
\]

That is, \( F(v) \) is convex.

Since \( F(\tilde{v}) \) is finite and \( F(v) \geq 0 \), \( F(v) \) is proper. By [12, p 52], a convex proper function is closed if it is lower semi-continuous. For each \( v_0 := (\omega_0, b_0, \xi_0) \in \mathbb{R}^{d+N+1} \) and any sequence \( \{v_i := (\omega_i, b_i, \xi_i)\}_{i=1}^{\infty} \) converges to \( v_0 \), there are two cases. One is that \( \lim_{i \to \infty} F(v_i) < \infty \), then \( F(v_0) = \lim_{i \to \infty} F(v_i) \). The other is that \( \lim_{i \to \infty} F(v_i) = \infty \), then \( F(v_0) \leq \lim_{i \to \infty} F(v_i) \). So \( F(v) \) is semi-continuous then closed. Now \( F(v) \) is a convex closed proper function.

Next we show that \( F(\tilde{v}) \) has no direction of recession. It is equivalent to show that for any \( \bar{v} := (\tilde{\omega}, \tilde{b}, \xi) \neq 0 \), there exists a \( v_0 := (\omega_0, b_0, \xi_0) \) such that \( g(\lambda) := F(\omega_0 + \lambda \tilde{v}) \) is not a non-increasing function. That is, there are some \( \lambda_0 > \lambda_1 \) such that \( g(\lambda_0) > g(\lambda_1) \). Given any \( \bar{v} \neq 0 \), choose a \( v_0 \) such that there exist \( \lambda_0 > \lambda_1 \) such that \( v_0 + \lambda_0 \tilde{v} \in \Xi \) and \( v_0 + \lambda_1 \tilde{v} \in \Xi \). Then \( g(\lambda_0) = \|\omega_0 + \lambda_0 \tilde{v}\|_* + C \sum_{i=1}^{N} (\xi_{0,i} + \lambda_0 \xi_{0,i}) \).
\[ \lambda_0 \xi_i \), \quad g(\lambda_1) = \| \omega_0 + \lambda_1 \omega \|_* + C \sum_{i=1}^N (\xi_{0,i} + \lambda_1 \xi_i). \]

We have
\[
g(\lambda_1) - g(\lambda_0) = \| \omega_0 + \lambda_1 \omega \|_* - \| \omega_0 + \lambda_0 \omega \|_* + C \sum_{i=1}^N (\xi_{0,i} - \xi_{0,i} + (\lambda_1 - \lambda_0) \xi_i)
\]
\[
= \| \omega_0 + \lambda_0 \omega + (\lambda_1 - \lambda_0) \omega \|_* - \| \omega_0 + \lambda_0 \omega \|_* + C(\lambda_1 - \lambda_0) \sum_{i=1}^N \xi_i
\]
\[
\leq \| \omega_0 + \lambda_0 \omega \|_* + (\lambda_1 - \lambda_0) \| \omega \|_* - \| \omega_0 + \lambda_0 \omega \|_* + C(\lambda_1 - \lambda_0) \sum_{i=1}^N \xi_i
\]
\[
= (\lambda_1 - \lambda_0)(\| \omega \|_* + C \sum_{i=1}^N \xi_i) < 0.
\]

Therefore, \( F(v) \) has no direction of recession.

We can now directly obtain that the solution set to optimization (4) is non-empty, compact and convex.

Finally, if \( \| \cdot \|_* \) is strictly convex, \( F(v) \) is also strictly convex. We choose any distinct \( v_1 := (\omega_1, b_1, \xi_1), v_2 := (\omega_2, b_2, \xi_2) \) in the solution set and denote the minimal of Optimization (4) by \( \tau \). Since the solution set is convex, \( v_1/2 + v_2/2 \) is also in the solution set. We have
\[
F(v_1/2 + v_2/2) < F(v_1)/2 + F(v_2)/2 = \tau.
\]

This indicates that \( F(v_1/2 + v_2/2) \) is the new minimal of Optimization (4), which is a contradiction. Therefore, \( v_1 = v_2 \). The proof of the uniqueness is completed.

2.3. Regularizations. Now we transfer Optimization (4) to an unconstrained optimization by the Karush Kuhn Tucker (KKT) conditions. Suppose \( v^* = (\omega^*, b^*, \xi^*) \) is one of the solutions to Optimization (4). The Lagrange (primal) function \( \mathcal{L}_P(v) \) of Optimization (4) is
\[
\mathcal{L}_P(v) = \| \omega \|_* + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \alpha_i[y_i(x_i^T \omega + b) - (1 - \xi_i)] - \sum_{i=1}^N \mu_i \xi_i,
\]
where \( \alpha_i \geq 0, \mu_i \geq 0 \) are Lagrange multipliers. Setting the derivative of \( \mathcal{L}_P(v) \) with respect to \( \xi_i \) to zero, we have
\[
\alpha_i = C - \mu_i, \text{ for } i = 1, 2, \ldots, N. \tag{7}
\]

The KKT conditions for \( \mathcal{L}_P(v) \) include
\[
\alpha_i \left[ y_i(x_i^T \omega + b) - (1 - \xi_i) \right] = 0,
\]
\[
\mu_i \xi_i = 0,
\]
\[
y_i(x_i^T \omega + b) - (1 - \xi_i) \geq 0. \tag{8}
\]

Combining Equations (7) and (8), we have that if \( y_i(x_i^T \omega + b) - (1 - \xi_i) > 0 \), then \( \alpha_i = C - \mu_i = 0 \). It implies \( \mu_i = C \neq 0 \), then \( \xi_i = 0 \) and \( y_i(x_i^T \omega + b) > 1 \). Otherwise, \( y_i(x_i^T \omega + b) - (1 - \xi_i) = 0 \), \( \xi_i = 1 - y_i(x_i^T \omega + b) \). Consequently, we have
\[
\xi_i^* = \begin{cases} 
0 & \text{if } 1 - y_i(x_i^T \omega^* + b^*) < 0 \\
1 & \text{if } 1 - y_i(x_i^T \omega^* + b^*) \geq 0.
\end{cases}
\]

That means \( \xi_i^* = [1 - y_i(x_i^T \omega^* + b^*)]_+ = \max\{0, 1 - y_i(x_i^T \omega^* + b^*)\} \). Those data points whose corresponding \( \xi_i^* \)’s are non-zero are the so-called support vectors, by
which the classifier is totally determined. Now although it may seem a bit strange, we can actually rewrite Optimization (4) as an unconstrained optimization problem whose target variable is $(\omega, b)$

$$
\min_{\omega \in \mathbb{R}^d, b \in \mathbb{R}} \|\omega\|_* + C \sum_{i=1}^{N} [1 - y_i(x_i^T \omega + b)]_+.
$$

Letting $\lambda = 1/C$, we get

$$
\min_{\omega \in \mathbb{R}^d, b \in \mathbb{R}} \sum_{i=1}^{N} [1 - y_i(x_i^T \omega + b)]_+ + \lambda \|\omega\|_*.
$$

This is indeed a regularized model which is the sum of a hinge loss function and an arbitrary norm penalty term. It is a convex optimization.

Before we show that Optimization (4) can be rewritten as Optimization (9), we need to show that Optimization (9) has solutions.

**Proposition 3.** The solution set to Optimization (9) is non-empty, compact and convex. In particular, if $\| \cdot \|_*$ is further strictly convex, Optimization (9) has one and only one solution.

**Proof.** Since Optimization (9) is unconstrained, the feasible solution set to it is $\mathbb{R}^{d+1}$ which is non-empty and convex, then there exists a feasible solution $\tilde{u} := (\tilde{\omega}, \tilde{b})$.

By Theorem 27.2 in [12], we only need to show that the objective function $F(u) := \sum_{i=1}^{N} [1 - y_i(x_i^T \omega + b)]_+ + \lambda \|\omega\|_*$ is a convex closed proper function which has no direction of recession, then the solution set to (9) is non-empty, closed, bounded (so compact) and convex.

$F(u)$ is indeed the sum of a hinge loss function and an arbitrary norm penalty term, so it is convex. Since $F(u)$ is finite and $F(u) \geq 0 > -\infty$, $F(u)$ is proper. By [12, p 52], a convex proper function is closed if it is lower semi-continuous. Because $F(u)$ is actually continuous, the lower semi-continuity can be naturally obtained. So $F(u)$ is a convex closed proper function.

Next we show that $F(u)$ has no direction of recession. It is equivalent to show that for any $\tilde{u} := (\tilde{\omega}, \tilde{b}) \neq 0$, there exists a $u_0 := (\omega_0, b_0)$ such that $g(\mu) := F(u_0 + \mu \tilde{u})$ is not a non-increasing function. That is, there are some $\mu_0 > \mu_1$ such that $g(\mu_0) > g(\mu_1)$. Given any $\tilde{u} \neq 0$ and randomly-chose $u_0$ such that $g(0) = F(u_0) < \infty$. Since $g(\mu)$ is continuous and $g(\mu) \geq \lambda \|\omega_0 + \mu \tilde{\omega}\|_* \to \infty (\mu \to \infty), \lim_{\mu \to \infty} g(\mu) = \infty$. That is, for any $G > 0$, there exists a $\mu' > 0$ such that $g(\mu') > G$. Then for $G = g(0)$, there exists a $\mu' > 0$ such that $g(\mu') > g(0)$. It implies that $F(u)$ has no direction of recession.

We can directly obtain that the solution set to Optimization (9) is non-empty, compact and convex.

Finally, if $\| \cdot \|_*$ is strictly convex, $F(u)$ is also strictly convex. We choose any distinct $u_1 := (\omega_1, b_1), u_2 := (\omega_2, b_2)$ in the solution set and denote the minimal of Optimization (4) by $\tau$. Since the solution set is convex, $u_1/2 + u_2/2$ is also in the solution set We have

$$
F(u_1/2 + u_2/2) < F(u_1)/2 + F(u_2)/2 = \tau.
$$

This indicates that $F(u_1/2 + u_2/2)$ is the new minimal of Optimization (4), which is a contradiction. Therefore, $u_1 = u_2$. The proof of the uniqueness is completed. \qed
Next we will discuss the relationship between the solutions to Optimization (4) and the solutions to Optimization (9).

**Theorem 2.1.** (1) If \( v^* = (\omega^*, b^*, \xi^*) \) is one of the solutions to Optimization (4), then \( u^* = (\omega^*, b^*) \) is also one of the solutions to Optimization (9).

(2) If \( u^* = (\omega^*, b^*) \) is one of the solutions to Optimization (9), then let \( \xi^* = [1 - y_i(x_i^T \omega^* + b^*)]_+ \), \( v^* = (\omega^*, b^*, \xi^*) \) is also one of the solutions to Optimization (4).

**Proof.** (1) By the constraints of Optimization (4), we have \( \xi_i \geq 0 \) and \( \xi_i \geq 1 - y_i(x_i^T \omega + b) \), then \( \xi_i \geq [1 - y_i(x_i^T \omega + b)]_+ \). Suppose \( v^* = (\omega^*, b^*, \xi^*) \) is one of the solutions to Optimization (4). For any feasible solution \( v = (\omega, b, \xi) \) to (4), we have

\[
C \sum_{i=1}^{N} \xi_i^* + ||\omega^*||_* \leq C \sum_{i=1}^{N} \xi_i + ||\omega||_*
\]

For any feasible solution \( v = (\omega, b, \xi) \) where \( \xi = ([1 - y_i(x_i^T \omega + b)]_+) \), we obtain

\[
C \sum_{i=1}^{N} [1 - y_i(x_i^T \omega^* + b^*)]_+ + ||\omega^*||_* \leq C \sum_{i=1}^{N} \xi_i^* + ||\omega^*||_* \\
\leq C \sum_{i=1}^{N} \xi_i + ||\omega||_* \\
= C \sum_{i=1}^{N} [1 - y_i(x_i^T \omega + b)]_+ + ||\omega||_*
\]

Then for any feasible solution \( u = (\omega, b) \) to Optimization (9),

\[
C \sum_{i=1}^{N} [1 - y_i(x_i^T \omega^* + b^*)]_+ + ||\omega^*||_* \leq C \sum_{i=1}^{N} [1 - y_i(x_i^T \omega + b)]_+ + ||\omega||_*
\]

Multiply the left and right sides by \( \lambda = 1/C \), we get

\[
\sum_{i=1}^{N} [1 - y_i(x_i^T \omega^* + b^*)]_+ + \lambda||\omega^*||_* \leq \sum_{i=1}^{N} [1 - y_i(x_i^T \omega + b)]_+ + \lambda||\omega||_*
\]

It follows that \( u^* = (\omega^*, b^*) \) is also one of the solutions to Optimization (9).

(2) Suppose \( u^* = (\omega^*, b^*) \) is one of the solutions to Optimization (9). For any feasible solution \( u = (\omega, b) \) to Optimization (9), we have

\[
C \sum_{i=1}^{N} [1 - y_i(x_i^T \omega^* + b^*)]_+ + ||\omega^*||_* \leq C \sum_{i=1}^{N} [1 - y_i(x_i^T \omega + b)]_+ + ||\omega||_*
\]

Let \( \xi_i = [1 - y_i(x_i^T \omega + b)]_+ \), for any \( i = 1, 2, \ldots, N \). Considering the feasible solution \( v^* = (\omega^*, b^*, \xi^*) \) to Optimization (4), we have

\[
C \sum_{i=1}^{N} [1 - y_i(x_i^T \omega^* + b^*)]_+ + ||\omega^*||_* \leq C \sum_{i=1}^{N} [1 - y_i(x_i^T \omega + b)]_+ + ||\omega||_*
\]

Then

\[
C \sum_{i=1}^{N} \xi_i^* + ||\omega^*||_* = C \sum_{i=1}^{N} [1 - y_i(x_i^T \omega^* + b^*)]_+ + ||\omega^*||_*
\]
\[
\leq C \sum_{i=1}^{N} [1 - y_i (x_i^T \omega + b)]_+ + \|\omega\|_* \\
\leq C \sum_{i=1}^{N} \xi_i + \|\omega\|_*..
\]

Thus for any feasible solution \(v = (\omega, b, \xi)\) to Optimization (4),
\[
C \sum_{i=1}^{N} \xi_i^* + \|\omega^*\|_* \leq C \sum_{i=1}^{N} \xi_i + \|\omega\|_*..
\]

It can be concluded that \(v^* = (\omega^*, b^*, \xi^*)\) is one of the solutions to Optimization (4).

2.4. Examples of \(p\)-norm margins. Next we will discuss some special cases, the SVM classifiers by \(p\)-norm margins for \(1 \leq p \leq \infty\), especially for \(p = \infty\) and \(p = m\) where \(m\) is an even and non-zero natural number. We will show how to understand the sparsity brought by 1-norm regularization from the SVM classifier by \(\infty\)-norm margin and introduce a new method to help solve the SVM classifiers by \(m\)-norm margins.

In practice, \(p\)-norm regularization is a crucial technique to prevent over-fitting [2, 7, 13]. Different regularization terms will lead to different performances. For instances, only 1-norm regularization can result in sparsity. That is, \(\|\cdot\|_*\) is 1 norm, the margin is \(\infty\)-norm margin. The reasons for sparsity were discussed in the parameter space in the past. In fact, only the extreme points of the ball of 1 norm are sparse then the model can obtain sparse solutions. Here, we will illustrate the causes of sparsity. Several examples in 2-dimensional space will be provided to help understand the sparsity in data space. It was shown that the decision boundaries are totally determined by a small number of support vectors, so we only need to discuss the case of a few data points, especially 2 distinct points. See Figure 12 to understand the projections by different norms.

There are two different cases for 2 distinct points in 2-dimensional space, case one is that one of their coordinates is equal, case two is that all of their coordinates are not equal. We present these two cases under the sense of 2 norm and \(\infty\) norm in Figure 3a and Figure 3b, respectively. For 2 norm, only in case one the solutions will be sparse, which means the obtained hyperplane is perpendicular to or parallel to a certain coordinate axis. But for \(\infty\) norm, in both two cases the solutions are sparse.

By Proposition 3, since 1 norm is not strictly convex, the uniqueness of the solutions cannot be guaranteed. Now we will discuss the uniqueness in the data space rather than the parameter space, and give a geometrical interpretation of why the solutions may not be sparse when they are not unique. As we can see in Figure 4a, in a 2-dimensional space, when the slope of the line connecting two distinct points is \(\pm 1\), the tangent point of two balls of \(\infty\) norm is the common vertex of two squares. Then any line which lies entirely inside the shadowed area in Figure 4a is one of the solutions to the optimization. These cases correspond to the cases that the parameter points lie in the blue line segment (top right edge of the rhombus) in Figure 4b. In this case, almost all of these solutions are not sparse.

However, it does not mean that the SVM classifier by \(\infty\)-norm margin will never have a unique solution. In fact, it can obtain a unique solution in some special cases. In Figure 5a, the solution obtained is the only possible one, and it is sparse.
In parameter space, the corresponding parameter point is the blue point in Figure 5b, which exactly lie on one of the coordinate axis.

Next, we will discuss the spacial cases for $p = m$ where $m$ is an even natural number. In the standard SVM classifier by Euclidean margin, there is a fast algorithm called Sequential Minimal Optimization (SMO) [10] to solve the Optimization (4). SMO is actually solving the Lagrange (Wolfe) dual function of Optimization (4) where $\| \cdot \|_\ast$ is 2 norm. But the Lagrange (Wolfe) dual function is derived by setting the derivative of the Lagrange (primal) function to zero. So not all norms can guarantee that Optimization (4) can obtain Lagrange (Wolfe) dual function.
We try to find some special norms that can obtain Lagrange (Wolfe) dual function, then the optimization can be solved in other ways.

Let us now consider \( p \) norm where \( p = m \) is an even natural number. Its dual norm is \( \frac{m}{m-1} \) norm. The objective function of Optimization (4) now is \( \frac{m-1}{m} \|\omega\|_{\frac{m}{m-1}} + C \sum_{i=1}^{N} \xi_i \). We can write down the primal Lagrange function

\[
\mathcal{L}_P = \frac{m-1}{m} \|\omega\|_{\frac{m}{m-1}} + C \sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} \alpha_i [y_i (x_i^\top \omega + b) - (1 - \xi_i)] - \sum_{i=1}^{N} \mu_i \xi_i,
\]

where \( \alpha_i \geq 0, \mu_i \geq 0, i = 1, 2, \ldots, N \) are Lagrange multipliers. Let \( \alpha = (\alpha_i)_{i=1}^{N} \).

Since \( m \) is an even natural number, \( m - 1 \) is an odd natural number. We have that for any \( x \in \mathbb{R}, |x|^{\frac{m}{m-1}} = (|x|^m)^{\frac{1}{m-1}} = (x^m)^{\frac{1}{m-1}} = x^{\frac{m}{m-1}} \). It does not hold if \( m \) is an odd natural number. In this way, \( \|\omega\|_{\frac{m}{m-1}} \) is differentiable and its derivative has a simple form. By setting the derivatives with respect to \( \omega, b, \xi_i \) to zero, we get

\[
\omega = \left( \sum_{i=1}^{N} \alpha_i y_i x_i \right)^{m-1}, \quad \text{here we use element-wise power},
\]

\[
0 = \sum_{i=1}^{N} \alpha_i y_i, \quad \alpha_i = C - \mu_i, \quad i = 1, 2, \ldots, N.
\]

Substituting Equation (10) into \( \mathcal{L}_P \), we obtain the Lagrange (Wolfe) dual function

\[
\mathcal{L}_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{m} \sum_{i_1, i_2, \ldots, i_m = 1}^{N} \prod_{k=1}^{m} \left( \alpha_{i_k} y_{i_k} \left( \sum_{j=1}^{m} \prod_{k=1}^{m} x_{i_k, j} \right) \right),
\]

where \( x_{i_k, j} \) is the \( j \)th element of \( x_{i_k} \).

We can exactly write \( \mathcal{L}_D \) in a tensor form. For convenience, the notions and operations of tensors are defined as in the book [11]. Let \( 1 = (1)_{i=1}^{N} \). Considering a
mth order Nth dimension tensor \( A_m := (\prod_{k=1}^{m} y_{ik} (\sum_{j=1}^{d} \prod_{k=1}^{m} x_{ik,j}))_{i_1,i_2,...,i_m=1}^{N,N,...,N} \).

\( \mathcal{L}_D \) can be written as
\[
\mathcal{L}_D = 1^\top \alpha - \frac{1}{m} A_m \alpha^m,
\]
where \( A_m \alpha^m \) is the m-mode product.

Therefore, the Optimization (4) where \( \| \cdot \|_* \) is \( \frac{m}{m^\top} \) norm can be solved by solving Optimization
\[
\min_{\alpha \in [0, \infty)^N} 1^\top \alpha - \frac{1}{m} A_m \alpha^m. \tag{11}
\]

3. Nonlinear support vector machine classifiers.

3.1. Kernel tricks and p-Norm RKBSs. The SVM classifiers by non-Euclidean margins described so far find linear boundaries in the input space. As with other linear methods, we can make the procedure more flexible by enlarging the feature space. Generally linear boundaries in the enlarged space achieve better training-class separation, and translate to nonlinear boundaries in the original space. Once the basis functions \( h := (h_j(x))_{j=1}^n \) are selected, the procedure is the same as before. The only difference is that we now fit the SVM classifiers using input feature \( h(x_i) := (h_j(x_i))_{j=1}^n, \) and obtain the nonlinear function \( h(x)^\top \omega + b \). The classifier is \( \text{sign}(h(x)^\top \omega + b) \).

If we use the basis functions \( h, \) \( \mathcal{L}_D \) has the form
\[
\mathcal{L}_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{m} \sum_{i_1,i_2,...,i_m=1}^{N,N,...,N} \prod_{k=1}^{m} (\alpha_{i_k} y_{i_k} (\sum_{j=1}^{n} \prod_{k=1}^{m} h_j(x_{i_k,j}))).
\]

And by (10), \( \omega = \left( \sum_{i=1}^{N} \alpha_i y_i h(x_i) \right)^{m-1} \), here we still use element-wise power. Thus, we can write the nonlinear function as
\[
\omega = \left( \sum_{i=1}^{N} \alpha_i y_i h(x_i) \right)^{m-1} = \sum_{i_2,i_3,...,i_m=2}^{N} \prod_{k=2}^{m} (\alpha_{i_k} y_{i_k} (\sum_{j=1}^{n} \prod_{k=1}^{m} h_j(x_{i_k,j}) h_j(x_i))) + b. \tag{12}
\]

In (12), letting \( K_m(x, x_{i_2}, \ldots, x_{i_m}) := \sum_{j=1}^{n} \prod_{k=2}^{m} h_j(x_{i_k,j}) h_j(x_i) \), we can obtain the mth order \( \text{tensor kernel function} \). In fact, we need not specify the transformation \( h(x) \) at all, but require only knowledge of the tensor kernel function to find the nonlinear function. That is what we called kernel trick.

If \( m = 2 \), we have
\[
\omega = \sum_{i=1}^{N} \alpha_i y_i h_j(x_i) h_j(x_i) = \sum_{i=1}^{N} \alpha_i y_i \langle h(x), h(x_i) \rangle + b. \tag{13}
\]

The functional (13) involves basis functions \( h(x) \) only through inner products. The inner product can be treated as the standard kernel function, a special case of tensor kernel function. In practice, the standard kernel functions are more commonly used. Henceforth, we only discuss the two-variable kernel functions. Figure 6 shows the geometrical interpretation of the kernel tricks.

Kernel trick is an important and powerful method to fit the nonlinear function. It depends on the data points, especially the relationship between the data points, and not on the choice of the basis functions. In point of fact, by kernel trick, the SVM classifier fitting can be viewed as function estimation in some special function space. For the SVM classifier by Euclidean margin, the space is called Reproducing
Kernel Hilbert Spaces (RKHSs) [13, 16]. It is the inner product form of the standard kernel function that leads to Hilbert space rather than Banach space. To possess more geometrical structures including sparsity, Reproducing Kernel Banach Spaces (RKBSs) are introduced [18, 20], in which the $p$-norm RKBSs are the most popular, see Appendix B for details. In fact, RKBSs to SVM classifiers by non-Euclidean margins are equivalent to RKHSs to SVM classifiers by Euclidean margins. That is, in this paper, we have theoretically supplemented the construction of RKBSs.

The RKBSs are the spaces spanned by the finite or infinite orthonormal basis. For the finite-dimensional case, since any finite-dimensional normed space is Banach space, it is easy to construct. However, we cannot generalize directly from finite-dimensional Banach spaces to infinite-dimensional Banach spaces. A simple counter-example is that in $L_1$ the existence of solutions to a convex optimization problem is easy to prove, but its infinite-dimensional version $l_1$ is not reflexive so the existence of solutions to convex optimization problems in it is difficult to show. Without the reflexivity, the existence of solutions to the SVM classifier fitting problem in the Banach space cannot be guaranteed [6, p 75, Proposition 6]. About the construction of infinite-dimensional $p$-norm RKBSs, we refer the reader to [18].

3.2. SVM classifiers in RKBSs. By Theorem 2.1, we will focus on the unconstrained optimization only.

Once the basis functions are selected to be the kernel functions $K(x, x_i), i = 1, 2, \ldots, N$, we obtain the spanned space

$$B_p = \left\{ f := \sum_{i=1}^{N} a_i K(x, x_i) : (a_i)_{i=1}^{N} \in L_p \right\}$$

equipped with the norm
∥f∥_{B_p} := \left( \sum_{i=1}^{N} |a_i|^p \right)^{1/p},

for 1 ≤ p < ∞. B_p is a finite-dimensional normed space, so it is naturally a Banach space. In the finite-dimensional RKBSs, the non-linear SVM classifiers by non-Euclidean margins have a similar form like the linear classifiers. The corresponding optimization also has the form as Optimization (9)

\begin{equation}
\min_{f \in B_p} \sum_{i=1}^{N} [1 - y_i f(x_i)]_+ + \lambda \|f\|_{B_p}. \tag{14}
\end{equation}

Because \( f \in B_p \) is a finite linear combination of basis functions, the target variable is indeed the finite-dimensional vector \( a := (a_i)^N_{i=1} \). Thus, Optimization (14) is actually a finite-dimensional optimization

\begin{equation}
\min_{a \in L_p} \sum_{i=1}^{N} [1 - y_i \sum_{j=1}^{N} a_j K(x_i, x_j)]_+ + \lambda \|a\|_{L_p}. \tag{15}
\end{equation}

We can still use Proposition 3 to guarantee the existence and conditional uniqueness of its solutions.

The infinite-dimensional \( p \)-norm RKBSs are defined by Equations (17) and (18) in Appendix B. The optimization is the same as Optimization (14), but now it is an infinite-dimensional problem. Proposition 3 cannot be applied to it. For 1 < p < ∞, \( B_p \) which is isometrically equivalent to \( l_p \) is reflexive. The existence and conditional uniqueness of the solutions to Optimization (14) can be guaranteed \[6, p 75, Proposition 6\]. But \( B_1 \), which is isometrically equivalent to \( l_1 \), is not reflexive. Nevertheless, the SVM classifiers by non-Euclidean margins in \( B_p \) still can be guaranteed to have solutions. About the infinite-dimensional RKBSs, we refer to \[18\].

Moreover, there are several representor theorems to present the form of the solution to Optimization (14) in infinite-dimensional RKBSs. Such as \[8, 14, 15, 18, 20\] and so on.

**Remark 2 (The domain of data).** When we discuss how to build the SVM classifiers by non-Euclidean margins, we use Theorem A.1. It requires the domain of the data points is \( \mathbb{R}^d \), a finite-dimensional space. Such a requirement is also used in the proofs of Propositions 1, 2 and 3. In an infinite-dimensional space, the compactness cannot be derived from closeness and boundedness. It fails to extend directly from finite-dimensional domain to infinite-dimensional domain.

In fact, Theorem A.1 can be generalized to the Banach spaces, see \[3, Lemma 2.2\] and \[5, Lemma 1\]. The only difference is the form of the functional used to define the hyperplane. Now, the domains of the target variables of the optimizations can be Banach spaces. And by \[6, p 75, Proposition 6\], the existence of solutions can be guaranteed if the Banach space is reflexive. However, \( l_1 \) is not reflexive. It follows that some constraints are added to the choice of the norm. It should be noted that the finite-dimensional space is a special case of reflexive Banach spaces.

When we turn to the RKBSs, since we are estimating the function in RKBSs, the domain of the data becomes less important. Actually, the domain of the data now can be locally compact Hausdorff space, see \[18\].
4. **Numerical experiments.** We shall show some numerical experiments on both artificial data and real data to evaluate the SVM classifiers by non-Euclidean margins. Due to the interest in sparsity, we choose the SVM classifier by $\infty$-norm margin, which can be equivalently transferred to the 1-norm SVM classifier, in finite-dimensional space as a special example. We have actually introduced three convex optimizations to find the classifiers such as Optimization (2) for linear separable case, Optimization (4) for non-separable case, and Optimization (9) for non-separable case. Thus, the experiments will be performed on noiseless and noisy data sets as well as on MNIST.

The performance and sparsity of the 1-norm SVM classifier are discussed by some recent papers. [3] gave statistical error bounds for 1-norm SVM classifier. [19] discussed the $\infty$-norm margin behind 1-norm SVM classifier, which is actually a special case of non-Euclidean margins. [21] showed the study of sparsity by 1-norm SVM classifier. [22] deeply studied the 1-norm SVM classifier.

4.1. **Experiments on artificial data.** The linear separable sparse data set is generated randomly. Then some noise are added to the separable data set to generate the non-separable data set. Figure 7 shows these two data sets.

![Figure 7. Noiseless and noisy data sets. The classes overlaps in noisy data set.](image)

For the noiseless data set, because of its linear-separability, the hard-margin solution exists. That is, Optimization (2) has solutions. Conversely, for the noisy data set, Optimization (2) does not have solution. Thus, we will perform three experiments on the noiseless data set, and two experiments on the noisy data set. The results are shown in Figure 8 for noiseless data set, and in Figure 9 for noisy data set, respectively. Recall that here $\| \cdot \|$ is 1 norm.

As we can see, the classifiers in Figure 8b and 8c are totally same. And they all allow some data points to fall in the margin and even cross the boundary. Although the soft-margin classifier misclassifies several -1-class data points, intuitively the classifier is close to the real solution, and it is sparse.

Similarly, the two classifiers in Figure 9a and 9b are the same and all sparse. Some data points are allowed to fall in the margin.
4.2. Experiments on MNIST. The MNIST database (http://yann.lecun.com/exdb/mnist/) of handwritten digits has 10 classes corresponding to digits 0, 1, ..., 9. It contains 60000 training data and 10000 testing data. Each data is a row vector in $[0, 1]^{784}$. Since SVM classifiers by non-Euclidean margins are designed for binary classification problems, we choose two digits 6 and 9. Thus, we have 9939 training data and 1967 testing data. Some selected images of these two classes are displayed in Figure 10. We can see that compared to the normal image of 9, some images of 9 are quite strange. We run two experiments on MNIST. One is a linear SVM classifier by $\infty$-norm margin in Optimization (9). The other one is a non-linear SVM classifier in Optimization (15) by Gaussian kernel called it kernel SVM classifier by
\( \infty \)-norm margin here. The results obtained by linear and kernel SVM classifiers by \( \infty \)-norm margin are shown in Table 1.

\[
\begin{array}{|c|c|c|c|}
\hline
\text{MNIST} & \text{Training Errors} & \text{Test Errors} & \text{Sparsity} \\
\hline
\text{Linear SVM classifier by \( \infty \)-norm margin} & 0 / 9939 & 5 / 1967 & 654 / 785 \\
\hline
\text{Kernel SVM classifier by \( \infty \)-norm margin} & 0 / 9939 & 4 / 1967 & 1760 / 9939 \\
\hline
\end{array}
\]

**Table 1.** The results of experiments on MNIST. The row **Linear SVM classifier by \( \infty \)-norm margin** is the result obtained by (9) where \( \lambda = 0.1 \). The row **Kernel SVM classifier by \( \infty \)-norm margin** is the result obtained by (15) where \( \lambda = 0.1 \). The kernel function used here is Gaussian kernel 
\[
K(x, x') = \exp\left(-\frac{\|x - x'\|^2}{\sigma^2}\right)
\]
where \( \sigma = 9.6 \).

4.3. **Experiments on handwritten alphabets database.** The handwritten alphabets database used in this paper comes from kaggle([https://www.kaggle.com/sachinpatel21/az-handwritten-alphabets-in-csv-format](https://www.kaggle.com/sachinpatel21/az-handwritten-alphabets-in-csv-format)), the author of which has collected the images from several different sources. Similar to MNIST, this database of handwritten capital alphabets has 26 classes corresponding to letters A, B,..., Z. It contains more than 372 thousand images. Each data is a row vector in \([2, 255]^{784}\). Due to the property of SVM classifiers by non-Euclidean margins and the restriction of computing resources, we take 5,000 data points from each of O and Q, a total of 10,000 data points, and then randomly divided them into training set and test set in a 7:3 ratio. That means we have a training set with 7,000 data points and a test set with 3,000 data points. Some selected images of these two classes are displayed in Figure 11. It is easy to see that some images are quite indistinguishable. Similar to experiments on MNIST, we also run two experiments on it. The results are shown in Table 2.

\[
\begin{array}{|c|c|c|c|}
\hline
\text{4.3. Experiments on handwritten alphabets database.} & \text{Training Errors} & \text{Test Errors} & \text{Sparsity} \\
\hline
\text{Linear SVM classifier by \( \infty \)-norm margin} & 0 / 9939 & 5 / 1967 & 654 / 785 \\
\hline
\text{Kernel SVM classifier by \( \infty \)-norm margin} & 0 / 9939 & 4 / 1967 & 1760 / 9939 \\
\hline
\end{array}
\]
Figure 11. The comparison between handwritten alphabets O and Q. The first row is O and the second row is Q.

| Handwritten Alphabets | Training Errors | Test Errors | Sparsity |
|------------------------|-----------------|-------------|----------|
| Linear SVM classifier by ∞-norm margin | 0 / 7000 | 69 / 3000 | 401 / 785 |
| Kernel SVM classifier by ∞-norm margin | 0 / 7000 | 14 / 3000 | 1339 / 7000 |

Table 2. The results of experiments on Handwritten Alphabets Database. The row Linear SVM classifier by ∞-norm margin is the result obtained by (9) where λ = 0.001. The row Kernel SVM classifier by ∞-norm margin is the result obtained by (15) where λ = 0.001. The kernel function used here is Gaussian kernel $K(x, x') = \exp\left(-\frac{\|x-x\|^2}{\sigma^2}\right)$ where $\sigma = 2600$.

We can see that both linear and kernel SVM classifiers by ∞-norm margin obtain a sparse result. Although Kernel SVM classifier by ∞-norm margin requires many more computing resources, it obtains a higher accuracy and a higher sparse rate.

Appendix A. Distance from a point to a hyperplane. In this appendix, a theorem is introduced to give an explicit form for the distance from an arbitrary point to a given hyperplane, where the distance is derived from a general norm.

**Theorem A.1** (Arbitrary-norm projection on a plane). Let $x$ be any point in $\mathbb{R}^d$ not on the plane

$$H := \{x \in \mathbb{R}^d : x^T \omega + b = 0 \}.$$

Considering the distance derived from a general norm $\| \cdot \|$, the distance between $x$ and $H$ is given by

$$\text{dist}(x, H) = \frac{|x^T \omega + b|}{\|\omega\|_*},$$

(16)

where $\| \cdot \|_*$ is the dual norm to $\| \cdot \|$, which is defined as $\|z\|_* := \sup\{z^T x : \|x\| < 1\}$.

For the original theorem and the corresponding proof, we refer the reader to [9].
It is worth pointing out that the norm used here is not limited to be $p$ norm for $1 \leq p \leq \infty$, but can be any norm. To help understand the geometrical interpretation, Figure 12 shows some examples of $p$ norms where $p = 1, 2, \infty$. Moreover, the distances would be the same for any $1 \leq p \leq \infty$ in some cases. Such as, in 2-dimensional space, when the slope of the line is $\pm 1$, the distances derived from $p$ norms are the same.

**Figure 12.** The geometrical interpretation of distance from a point to a plane.

**Example.** For the 2 norm $\| \cdot \|_2$, $\text{dist}(x, H) = \frac{|x^\top \omega + b|}{\|\omega\|_2}$.

**Example.** For the $\infty$ norm $\| \cdot \|_\infty$, $\text{dist}(x, H) = \frac{|x^\top \omega + b|}{\|\omega\|_1}$.

**Appendix B. Reproducing Kernel Banach Spaces.** We review some definitions and theorems of Reproducing Kernel Banach Spaces (RKBSs) in [18]. Suppose that $\Omega$ and $\Omega^*$ are locally compact Hausdorff spaces. Let $\langle \cdot, \cdot \rangle_B$ be the dual bilinear product of the Banach space $B$ and its dual space $B^*$.

**Definition B.1 ([18], Definition 2.1).** Let a measurable kernel $K : \Omega \times \Omega^* \rightarrow \mathbb{R}$, and let a normed space $B$ be a Banach space composed of measurable functions $f : \Omega \rightarrow \mathbb{R}$ such that the dual space $B^*$ of $B$ is isometrically equivalent to a normed space $F$ composed of measurable functions $g : \Omega^* \rightarrow \mathbb{R}$.

We call $B$ a right-sided RKBS and $K$ its right-sided reproducing kernel if

1. $K(x, \cdot) \in F \cong B^*$, for each $x \in \Omega$,
2. $\langle f, K(x, \cdot) \rangle_B = f(x)$, for all $f \in B$ and all $x \in \Omega$.

If the Banach space $B$ reproduces from the other side, that is,

3. $K(\cdot, y) \in B$, for each $y \in \Omega^*$,
4. $\langle K(\cdot, y), g \rangle_B$, for any $g \in F \cong B^*$ and all $y \in \Omega^*$,

then $B$ is called left-sided RKBS and $K$ its left-sided reproducing kernel. If two-sided reproducing properties 1-4 as above are satisfied, then we say that $B$ is a two-sided RKBS with the two-sided reproducing kernel $K$. 
Next, we look at the $p$-norm RKBS. Let $\phi_n : \Omega \to \mathbb{R}$ and $\phi^*_n : \Omega^* \to \mathbb{R}$ be linearly independent measurable functions for $n \in \mathbb{N}$. Denote that the space $l_p$ for $1 \leq p < \infty$ and $l_\infty$ are the collection of all countable sequences of scalars with the standard norms $\| \cdot \|_p$ and $\| \cdot \|_\infty$, respectively, and the space $c_0$ is the subspace of $l_\infty$ with all countable sequences of scalars that converge to 0. We define the normed spaces

$$
\mathcal{B}_p = \left\{ f := \sum_{i=1}^{\infty} a_i \phi_i : (a_i)_{i=1}^{\infty} \in l_p \right\}
$$

equipped with the norm

$$
\| f \|_{\mathcal{B}_p} := \left( \sum_{i=1}^{\infty} |a_i|^p \right)^{1/p},
$$

for $1 \leq p < \infty$, and

$$
\mathcal{B}_{\infty}^* := \left\{ g := \sum_{n=1}^{\infty} b_n \phi^*_n : (b_n)_{n=1}^{\infty} \in c_0 \right\}
$$

equipped with the norm

$$
\| g \|_{\mathcal{B}_{\infty}^*} := \sup_{n \in \mathbb{N}} |b_n|.
$$

If $\{\phi_n\}_{n=1}^{\infty}$ and $\{\phi^*_n\}_{n=1}^{\infty}$ satisfy the condition

$$
\sum_{n=1}^{\infty} |\phi_n(x)| < \infty \text{ for all } x \in \Omega, \sum_{n=1}^{\infty} |\phi^*_n(y)| < \infty \text{ for all } y \in \Omega^*,
$$

then [18, Theorems 3.8, 3.20, and 3.21] guarantee that $\mathcal{B}_1$ is a right-sided RKBS with the right-sided reproducing kernel $K(x, y) := \sum_{n=1}^{\infty} \phi_n(x)\phi^*_n(y)$, $\mathcal{B}_p$ for $1 < p < \infty$ is a two-sided RKBS with the two-sided reproducing kernel $K(x, y)$, and $\mathcal{B}_{\infty}^*$ is a two-sided RKBS with the two-sided reproducing kernel $K^*(y, x) := \sum_{n=1}^{\infty} \phi^*_n(y)\phi_n(x)$. Thus, we call $\mathcal{B}_p$, a $p$-norm RKBS for $1 \leq p < \infty$, or $\mathcal{B}_{\infty}^*$ a $\infty$-norm RKBS.

These reproducing kernels $K$ and $K^*$ are generalized Mercer kernels. The functions $\phi_n$ and $\phi^*_n$ are called the left-sided and right-sided expansions of the generalized Mercer kernel $K$.

It is clear that $\mathcal{B}_1$ and $\mathcal{B}_{\infty}^*$ are isometrically equivalent to $l_1$ and $c_0$, respectively. This shows that $\mathcal{B}_1$ is isometrically equivalent to the dual space of $\mathcal{B}_{\infty}^*$. But $\mathcal{B}_{\infty}^*$ is a proper imbedding subspace of the dual space $\mathcal{B}_1$.

REFERENCES

[1] B. E. Boser, I. M. Guyon and V. N. Vapnik, A training algorithm for optimal margin classifiers, in Proceedings of the Fifth Annual Workshop on Computational learning theory, ACM, (1992), 144–152.
[2] P. Bühlmann and S. van de Geer, Statistics for High-Dimensional Data. Methods, Theory and Applications, Springer Series in Statistics, Springer, Heidelberg, 2011.
[3] L. Chen and H. Zhang, Statistical margin error bounds for Ω-norm support vector machines, Neurocomputing, 339 (2019), 210–216.
[4] C. Cortes and V. Vapnik, Support-vector networks, Machine Learning, 20 (1995), 273–297.
[5] R. Der and D. Lee, Large-margin classification in banach spaces, Journal of Machine Learning Research - Proceedings Track, 2 (2007), 91–98.
[6] I. Ekeland and T. Turnbull, Infinite-Dimensional Optimization and Convexity, Chicago Lectures in Mathematics, University of Chicago Press, Chicago, IL, 1983.
[7] T. Hastie, R. Tibshirani and M. Wainwright, Statistical Learning with Sparsity: The Lasso and Generalizations, Monographs on Statistics and Applied Probability, 143. CRC Press, Boca Raton, FL, 2015.
[8] L. Huang, C. Liu, L. Tan and Q. Ye, Generalized representer theorems in Banach spaces, *Anal. Appl. (Singap.)*, (2019).
[9] O. L. Mangasarian, Arbitrary-norm separating plane, *Operations Research Letters*, 24 (1999), 15–23.
[10] J. Platt, Sequential minimal optimization: A fast algorithm for training support vector machines.
[11] L. Q. Qi, H. B. Chen and Y. N. Chen, *Tensor Eigenvalues and Their Applications*, Advances in Mechanics and Mathematics, 39. Springer, Singapore, 2018.
[12] R. T. Rockafellar, *Convex Analysis*, Princeton Mathematical Series, No. 28, Princeton University Press, Princeton, N.J., 1970.
[13] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, The MIT Press, Cambridge, 2001.
[14] G. H. Song and H. Z. Zhang, Reproducing kernel Banach spaces with the $\ell^1$ norm II: Error analysis for regularized least square regression, *Neural Comput.*, 23 (2011), 2713–2729.
[15] G. H. Song, H. Z. Zhang and F. J. Hickernell, Reproducing kernel Banach spaces with the $\ell^1$ norm, *Appl. Comput. Harmon. Anal.*, 34 (2013), 96–116.
[16] I. Steinwart and A. Christmann, *Support Vector Machines*, Information Science and Statistics, Springer, New York, 2008.
[17] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, New York, 1995.
[18] Y. S. Xu and Q. Ye, Generalized mercer kernels and reproducing kernel banach spaces, *Mem. Amer. Math. Soc.*, 258 (2019).
[19] H. Yang, X. Yang, F. Zhang, Q. Ye and X. Fan, Infinite norm large margin classifier, *International Journal of Machine Learning and Cybernetics*, 10 (2019), 2449–2457.
[20] H. Z. Zhang, Y. S. Xu and J. Zhang, Reproducing kernel Banach spaces for machine learning, *J. Mach. Learn. Res.*, 10 (2009), 2741–2775.
[21] L. Zhang and W. Zhou, On the sparseness of 1-norm support vector machines, *Neural Networks*, 23 (2010), 373–385.
[22] J. Zhu, S. Rosset, R. Tibshirani and T. J. Hastie, 1-norm support vector machines, in *Advances in Neural Information Processing Systems*, (2004), 49–56.

Received January 2020; revised May 2020.

E-mail address: linying@live.cn
E-mail address: yeqi@m.scnu.edu.cn