Regularized deep learning with non-convex penalties

Sujit Vettam[a,1], Majnu John[b,c,d,2]

[a] *Booth School of Business, University of Chicago, IL.*

[b] *Department of Mathematics, Hofstra University, Hempstead, NY.*

[c] *Center for Psychiatric Neuroscience, Feinstein Institute of Medical Research, Manhasset, NY.*

[d] *Division of Psychiatry Research, Zucker Hillside, Northwell Health System, Glen Oaks, NY.*

Abstract

Regularization methods are often employed in deep learning neural networks (DNNs) to prevent overfitting. For penalty based methods for DNN regularization, typically only convex penalties are considered because of their optimization guarantees. Recent theoretical work have shown that non-convex penalties that satisfy certain regularity conditions are also guaranteed to perform well with standard optimization algorithms. In this paper, we examine new and currently existing non-convex penalties for DNN regularization. We provide theoretical justifications for the new penalties and also assess the performance of all penalties on DNN analysis of real datasets.

Keywords: deep learning, neural network, regularization, lasso, non-convex penalty

1. Introduction

The success of DNNs in learning complex relationships between the inputs and outputs may be mainly attributed to multiple non-linear hidden layers [1,2]. As a consequence of having multiple layers, DNNs typically have tens of thousands of parameters, sometimes

[1]Corresponding author, e-mail: sjv@chicagobooth.edu, svettam@uchicago.edu

[2]Corresponding author, address: 350 Community Drive, Manhasset, NY 11030. e-mail: mjohn5@northwell.edu, majnu.john@hofstra.edu, Phone: +01 718 470 8221, Fax: +01 718 343 1659

even millions. Such large number of parameters gives the method incredible amount of flexibility. However on the downside, this may lead to overfitting the data, especially if the training sample is not large enough. Overfitting means that the method may work well in the training set but not in the test set. Since overfitting is a typical problem for DNNs, many methods have been suggested to reduce it. Adding weight penalties to the cost function, drop-out, early stopping, max-norm regularization and data augmentation are some of the popular regularization methods used to avoid overfitting. In this paper, we narrow our focus to regularization methods based on weight penalties appended to the cost function.

Two most commonly considered penalties for DNN regularization are the L_1 and L_2 penalties. In statistical literature, these two penalties are known as Lasso [3] and Ridge penalties [4,5] respectively. One of the main advantages of working with these two penalties is convexity of the optimization problem which guarantees that a local optima will always be a global optimum. L_1 penalization is also a selection procedure as it sets many parameters to zero. L_2 penalization does not have this property. All the parameters after L_2 penalization and all non-zero parameters after L_1 penalization are shrunk towards zero. The resulting bias in the regularized solution of the above convex penalties has motivated a few authors to consider nonconvex penalties [6,7], which have the potential to yield nearly unbiased estimates for the parameters. Recent theoretical work [8,9] has also shown that although nonconvex regularizers may yield multiple local optima they are essentially as good as a global optima from a statistical perspective.

In this paper we present nonconvex penalty functions which could be utilized as regularizers of the parameters in a DNN. In the method section we motivate the definition of these penalty functions based on the L_0 norm. The main focus of our paper is to compare the performance of DNN with regularization based on nonconvex penalties with regularization based on convex penalty functions. We provide theoretical justifications for our proposed regularization approaches and also assess their performance on real datasets.

The paper is structured as follows. In section 2, we motivate and introduce our method for regularizing DNNs, and justify based on theoretical considerations. In section 3, we apply our method to real datasets and compare its performance with L_1 regularization. Finally we make our conclusions in section 4.

2. Methods

2.1. Background and Motivation

Consider a classifier $f_w : x \to y$ parameterized by the weight vector w, for input x and categorical output y. Optimal weights in a non-regularized setting are obtained by minimizing a cost function $\mathcal{L}(w)$. Typically the negative log-likelihood is taken as the cost function; in the case of a categorical output it will be the cross-entropy function. One general approach for regularizing DNNs is to append a penalty function $P_\theta(w)$ to the cost function, where θ denotes the vector of tuning parameters associated with the penalty function. As done in most of the literature we will be restricting our attention to co-ordinate separable penalty

functions which could be expressed as a sum

$$P_\theta(w) = \sum_{j=1}^{p} p_\theta(w_j),\ w = (w_1, \ldots, w_p) \in \mathcal{R}^p.$$

Thus the regularized optimization problem that we are interested in is

$$\underset{w \in \mathcal{R}^p}{\operatorname{argmin}}\{\mathcal{L}(w) + P_\theta(w)\} = \underset{w \in \mathcal{R}^p}{\operatorname{argmin}}\{\mathcal{L}(w) + \sum_{j=1}^{p} p_\theta(w_j)\}. \tag{1}$$

The most commonly discussed approach, known as the 'canonical selection procedure' is based on

$$P_\theta(w) = \sum_{j=1}^{p} I(w_i \neq 0),\ \text{ where } I \text{ denotes the indicator function;} \tag{2}$$

the penalty function in this case is referred to as the L_0 norm. The key ideas behind Akaike's, Bayesian and Minimax Risk based Information Criteria (AIC, BIC and RIC), and Mallow's C_p are all based on the above L_0 norm. However, it is intractable for DNN applications because finding the minimum of the objective function in (1) with the penalty function in (2) is in general NP hard. It is combinatorial in nature and has exponential complexity as it requires an exhaustive search of order $O(2^p)$.

The above-mentioned intractability has led to considerations of approximations for the penalty function in (2). The most widely considered approximations are of the class of Bridge functions [10,11]

$$\sum_{j=1}^{p} \left|w_j\right|^\gamma,\ \gamma > 0,$$

motivated by the fact that

$$\lim_{\gamma \to 0} \sum_{j=1}^{p} \left|w_j\right|^\gamma = \sum_{j=1}^{p} I(w_j = 0).$$

$\gamma = 1$ and $\gamma = 2$ cases (L_1 and L_2 penalties) are known in the literature as Lasso and Ridge penalties. Note that the penalty function in (2) is singular at zero and the optimization problem based on it is non-convex. Bridge penalty functions are convex when $\gamma \geq 1$ and non-convex for $0 < \gamma < 1$. Bridge functions are singular at zero only in the case $0 < \gamma \leq 1$. Thus Lasso is the only case among the class of Bridge functions which is both convex and has a singularity at origin. Convex relaxation of a non-convex problem has its advantage in the optimization setting based on the simple fact that the local minimum of a convex function is also a global minimum. Singularity at origin for the penalty function is essentially what guarantees the sparsity of the solution (i.e. setting to zero small estimated weights to reduce model complexity).

Although Lasso has the above-mentioned advantages over other Bridge estimators, it differs from the L_0 norm in a crucial aspect: where as the L_0 norm is constant for any nonzero

argument the L_1 norm increases linearly with the absolute value of the argument. This linear increase results in a bias for the L_1-regularized solution [6] which in turn could lead to modeling bias. As mentioned in [6], in addition to unbiasedness and sparsity, a good penalty function should result in an estimator with continuity property. Continuity is necessary to avoid instability in model prediction. Note that the penalty function in (2) does not satisfy the continuity criterion. None of the Bridge penalty functions satisfy simultaneously all of the preceding three required properties. The solution for Bridge penalties is continuous only when $\gamma \geq 1$. However, when $\gamma > 1$ the Bridge penalties do not produce sparse solutions. When $\gamma = 1$ (i.e. Lasso) it produces continuous and sparse solution, but this comes at the price of shifting the resulting estimator by a constant (i.e. bias).

The above issues for the Bridge functions have led to considerations of other approximations for the penalty function in (2) (especially non-convex approximations) with the hope that these new approximations will satisfy (or nearly satisfy) all the three desirable properties mentioned above. In this paper we present two non-convex approximation functions:

$$\sum_{j=1}^{p} p_\theta(w_i) = \lambda \sum_{j=1}^{p} (1 - \varepsilon^{|w_i|}), \quad \theta = (\lambda, \varepsilon),\ \varepsilon \in (0,1),\ \lambda > 0, \tag{3}$$

$$\sum_{j=1}^{p} p_\theta(w_i) = \lambda \sum_{j=1}^{p} \frac{2}{\pi} \arctan(\gamma |w_i|), \quad \theta = (\lambda, \gamma),\ \gamma > 0, \lambda > 0. \tag{4}$$

The first penalty has appeared previously in the medical imaging literature in a method for magnetic resonance image reconstruction [12], and it has been referred to as the Laplace penalty function. See also [13]. The second penalty function based on arctan has not been considered in the literature so far to the best of our knowledge. Two other non-convex penalties that currently exist in the literature are the SCAD penalty,

$$\sum_{j=1}^{p} p_{a,\lambda}(w_i), \text{ where } a > 2 \text{ and } p_{a,\lambda}(t) = \begin{cases} \lambda |t|, & \text{for } |t| \leq \lambda \\ -(t^2 - 2a\lambda|t| + \lambda^2)/(2(a-1)), & \text{for } \lambda < |t| \leq a\lambda, \\ (a+1)\lambda^2/2, & \text{for } |t| > a\lambda, \end{cases}$$

developed by Fan and Li (2001) and the MCP regularizer (Zhang 2010),

$$\sum_{j=1}^{p} p_{b,\lambda}(w_i), \text{ where } b > 0 \text{ and } p_{b,\lambda}(t) = \text{sign}(t)\lambda \int_0^{|t|} \left(1 - \frac{z}{\lambda b}\right)_+ dz.$$

There are two other non-convex penalties that have appeared in the literature before that we do not consider in this paper. We present these two penalties in a later section and provide reasons for not considering them.

Although non-convex penalties are worth considering in DNN applications, they rarely get as much attention as the convex penalty functions. For example, textbooks such as [14] mention only L_1 and L_2 as regularization methods based on weight penalties. In this paper, we compare the performance of non-convex regularizers (including ours, SCAD and MCP) with L_1 regularizer for DNNs.

2.2. Theoretical considerations

Properties of SCAD and MCP penalties have been studied in the original papers in which they were presented. Below we present a few properties satisfied by Laplace and arctan penalty functions. These properties will help us to apply theorems from existing literature [6,8,9] that guarantees that any local optimum $\tilde{w}$ lies close to the target vector $w^* = \underset{w \in \mathcal{R}^p}{\operatorname{argmin}} \mathbb{E}(\mathcal{L}(w))$. These properties are easy to see from plots, but we give proofs.

Properties of the Laplace penalty function

We begin with a useful lemma.

Lemma 2.1. *For $\varepsilon \in (0, 1]$ and $x \geq 0$,*

$$\varepsilon^x \leq \frac{1}{1 - x \log \varepsilon}. \tag{5}$$

Proof. Let $y = -x \log \varepsilon$. Note that $y \geq 0$ based on the assumptions. Taking logarithm on both sides of the inequality (1), we get $x \log \varepsilon \leq -\log(1 - x \log \varepsilon)$. Multiplying by -1 on both sides and substituting y, we get $y \geq \log(1 + y)$. But this follows from the inequality $z - 1 \geq \log z$ for all $z > 0$ (in particular for $z \geq 1$) and the fact that $y \geq 0$. □

We present a few properties satisfied by the penalty function,

$$p_\lambda(t) = \lambda p(t) = \lambda(1 - \varepsilon^{|t|}), \quad \varepsilon \in (0, 1), \quad \lambda > 0.$$

(P1) $p_\lambda(0) = 0$ and $p_\lambda(t)$ is symmetric around zero. It is easily verified.

(P2) $p_\lambda(t)$ is increasing for $t \in [0, \infty)$. It is easy to see that $p'(t) = -\log(\varepsilon)\varepsilon^t$ is positive for $\varepsilon \in (0, 1)$ and $t \geq 0$.

(P3) For $t > 0$, the function $g(t) = p(t)/t$ is non-increasing in t. Since, for $t > 0$,

$$g'(t) = \frac{t\left[-(\log \varepsilon)\varepsilon^t\right] - \left[1 - \varepsilon^t\right]}{t^2} = \frac{\varepsilon^t\left[1 - t \log \varepsilon\right] - 1}{t^2}$$

it suffices to show that the numerator $\varepsilon^t\left[1 - t \log \varepsilon\right] - 1 \leq 0$ for $t > 0$. But this follows from Lemma 3.1 above.

(P4) The function p_λ is differentiable for all $t \neq 0$ and subdifferentiable at $t = 0$, with $\lim_{t \to 0+} p'_\lambda(t) = \lambda L$, $L = -\log \varepsilon$. It is easy to see that any point in the interval $[\lambda \log \varepsilon, -\lambda \log \varepsilon]$ is a subgradient of $p_\lambda(t)$ at $t = 0$.

(P5) There exists $\mu > 0$ such that $p_{\lambda,\mu}(t) = p_\lambda(t) + (\mu t^2/2)$ is convex: $\mu = \lambda(\log \varepsilon)^2$ will work. μ is a measure of the severity of non-convexity of the penalty function.

Since the penalty function $p_\lambda(t)$ satisfy the properties (P1) to (P5), we have $q_\lambda(t) := \lambda|t| - p_\lambda(t)$ is everywhere differentiable. These properties also imply that p_λ is λL-Lipschitz as

a function of t [8]. In particular, all subgradients and derivatives of p_λ are bounded in magnitude by λL [8]. We also see that for empirical loss $\mathcal{L}_n$ satisfying restricted strong convexity condition and conditions for λ and sample size in Theorem 1 in [8], the squared l_2-error of the estimator grows proportionally with the number of nonzeros in the target parameter and with λ^2. One condition that is not satisfied by $p_\lambda(t)$ is

(P6) There exists $\gamma \in (0, \infty)$ such that $p'_\lambda(t) = 0, \forall t > \gamma\lambda$. It is clear that such a γ does not exist for our penalty function. However, we note that $p'_\lambda(|t|)$ can be made arbitrarily close to zero for large $|t|$. In other words, the following property is satisfied.

(P6′) $\lim_{|t|\to\infty} p'_\lambda(t) = 0$.

(P6) and (P6′) are related to unbiasedness as mentioned in Fan and Li (2009). (P6) guarantees unbiasedness (and (P6′) near-unbiasedness) when the true unknown parameter is large to avoid unnecessary modeling bias.

Our penalty function depends on two parameters λ and ε, while as Lasso and ridge penalties depend on only λ alone. In our case, is there an optimal choice of λ that depends on ε alone? The following considerations based on Fan and Li shed some light into this. According to Fan and Li [6] a good penalty function $p_\lambda(t)$ should have the following two properties.

(P7) Minimum of the function $|t| + p'_\lambda(|t|)$ is positive. This property guarantees sparsity, at least in the L_2 empirical loss case; that is, the resulting estimator is a thresholding rule.

(P8) Minimum of the function $|t| + p'_\lambda(|t|)$ is attained at $t = 0$. This property, at least in the L_2 empirical loss case, is related to the continuity of the resulting estimator. Continuity helps to avoid instability in model prediction.

Consider the function $f(t) = |t| + p'_\lambda(|t|) = |t| - \lambda(\log\varepsilon)\varepsilon^{|t|}$, which is symmetric around zero, so that if a minimum is attained at t_{min}, then it is attained at $-t_{min}$ as well. This allows us to restrict our attention to the domain $0 \le t < \infty$. Note that in this domain $f(t) = t - \lambda(\log\varepsilon)\varepsilon^t$ and $t_{min} \ge 0$. We have $f'(t) = 1 - \lambda(\log\varepsilon)^2\varepsilon^t$ so that $\varepsilon^{t_{min}} = 1/\lambda(\log\varepsilon)^2$ or

$$t_{min} = \frac{\log\left[\lambda(\log\varepsilon)^2\right]}{(-\log\varepsilon)}. \tag{6}$$

In order that $t_{min} \ge 0$, we require that $\lambda \ge 1/(\log\varepsilon)^2$; t_{min} satisfies (P8) if

$$\lambda = \frac{1}{(\log\varepsilon)^2}. \tag{7}$$

For any t_{min} given by eq.(2) (i.e. with any $\lambda \ge 1/(\log\varepsilon)^2$)

$$f(t_{min}) = \frac{1}{(-\log\varepsilon)}\left\{\log\left[\lambda(\log\varepsilon)^2\right] + 1\right\} > 0,$$

so, in particular, $f(t_{min})$ corresponding to $\lambda = 1/(\log\varepsilon)^2$ is positive. Thus for a given ε, choosing λ based on eq.(3) will ensure that properties (P7) and (P8) are satisfied by our penalty function.

Theorem 1 in Fan and Li [6] provides required conditions for the $\sqrt{n}$-consistency of the estimator in a maximum likelihood framework and generalized linear models setting. The main assumption on the penalty function is stated as the following property.

(P9) $\max\left\{\left|p''_\lambda(|t_s|)\right| : t_s \neq 0, s = 1, \ldots, p\right\} \longrightarrow 0$ as $\lambda \to 0^+$. In our case, this property is satisfied because $\left|p''_\lambda(|t|)\right| = \lambda(\log\varepsilon)^2\varepsilon^{|t|} \leq \lambda(\log\varepsilon)^2 \longrightarrow 0$ as $\lambda \to 0^+$.

Lemma 3.1 suggests considering another penalty function $p_{1,\lambda}(t) = \lambda p_1(t)$ where

$$p_1(t) = \frac{-|t|\log\varepsilon}{1-|t|\log\varepsilon}.$$

This penalty function (p_1) is equivalent to Geman's penalty function mentioned in [15]; also mentioned in [12,13]. Most of the properties listed above are satisfied by the penalty function p_1 as well. For example,

$$p'_1(t) = \frac{-\log\varepsilon}{(1-t\log\varepsilon)^2} > 0, \quad \text{for } t \geq 0 \text{ verifying (P2).}$$

In order to check (P3) we consider

$$g_1(x) = \frac{1}{x(1-x\log\varepsilon)^2}.$$

For $0 < x_1 \leq x_2$, it is easy to check that

$$g_1(x_1) - g_1(x_2) = \frac{(x_2-x_1)\left\{[1-(\log\varepsilon)(x_2-x_1)]^2 + 3(\log\varepsilon)^2 x_1 x_2\right\}}{x_1x_2(1-x_1\log\varepsilon)^2(1-x_2\log\varepsilon)^2} \geq 0.$$

$$\text{Also } p''_1(t) = \frac{-2(\log\varepsilon)^2}{(1-t\log\varepsilon)^3} < 0 \text{ for } t \geq 0.$$

However, this suggests that the μ required for (P5) is $2\lambda(\log\varepsilon)^2$ which is twice as that for $p_\lambda(t)$. That is, non-convexity for $p_{1,\lambda}$ is twice as severe for p_λ. Also $p'_1(t)$ converges to zero (as do $p'(t)$) for large $|t|$ satisfying (P6′) for near-unbiasedness. However, since it can be shown that

$$\varepsilon^t < \frac{1}{(1-t\log\varepsilon)^2} \text{ for large } t, \text{ using the fact that } \frac{y}{2} > \log(1+y) \text{ for large } y,$$

we see that the convergence for $p'(t)$ is faster. Hence we do not consider the latter penalty function, $p_{1,\lambda}(t)$ in this paper.

We may also generalize our penalty function to $\lambda\left(1-\varepsilon^{\gamma|t|}\right)$ for $\gamma \geq 1$. However, the μ corresponding to this function will be $\gamma(\log\varepsilon)^2$, making it more severely non-convex similar to the Bridge penalty function when $\gamma < 1$. Hence, in this paper we focus only on $\gamma = 1$ case. Further comparison with bridge penalty is given in the subsection below.

Properties of the arctan penalty function

Here we check properties for the arctan penalty,

$$p_\lambda(t) = \lambda p(t) = \lambda \left(\frac{2}{\pi} \arctan(\gamma|t|) \right), \quad \gamma \in (0, \infty), \quad \lambda > 0.$$

Property P1 (- $p_\lambda(0) = 0$ and $p_\lambda(t)$ is symmetric around zero -) is again easily verified.

(P2): For $t \geq 0$,

$$p'(t) = \frac{2\gamma}{\pi} \frac{1}{1 + \gamma^2 t^2}$$

is positive for $\gamma > 0$. Hence $p_\lambda(t)$ is increasing for $t \in [0, \infty)$.

We state as a lemma a well-known fact about arctan function.

Lemma 2.2. *For $y \geq 0$,*

$$\frac{y}{1 + y^2} \leq \arctan(y) \leq y. \tag{8}$$

Proof. If we take $f(y) = y - \arctan(y)$, then $f'(y) = y^2/(1 + y^2) \geq 0$ for $y \geq 0$ and hence $f(\cdot)$ is non-decreasing in the interval $[0, \infty)$. In particular $f(y) \geq f(0) = 0$, which proves the right inequality. Similarly, by writing $f(y) = \arctan(y) - [y/(1 + y^2)]$, we have $f'(y) = 2y^2/(1 + y^2)^2 \geq 0$ for $y \geq 0$, proving the left inequality. □

(P3) For $t > 0$ consider the function $g(t) = p(t)/t$.

$$g'(t) = \frac{2}{\pi t^2} \left[\frac{\gamma t}{1 + \gamma^2 t^2} - \arctan(\gamma t) \right].$$

Thus $g'(t) \leq 0$ by the above lemma and hence $p(t)/t$ is non-increasing.

(P4) $\lim_{t \to 0+} p'_\lambda(t) = \lambda L$, $L = 2\gamma/\pi$. Any point in the interval $\left[-2\lambda\gamma/\pi, 2\lambda\gamma/\pi\right]$ is a subgradient of $p_\lambda(t)$ at $t = 0$.

(P5) $\mu = 2\lambda\gamma^2/\pi > 0$ makes $p_{\lambda,\mu}(t) = p_\lambda(t) + (\mu t^2/2)$ convex.

(P6') It is easy to check that $\lim_{|t| \to \infty} p'_\lambda(t) = 0$.

Also easy to see that $\left|p''_\lambda(t)\right| \leq 2\lambda\gamma^2/\pi \to 0$ which guarantees that (P9) is satisfied.

Convergence of the Laplace and arctan approximation functions

Here we present heuristic justifications for using the Laplace or arctan penalties over the bridge penalties by considering their respective error in approximating the indicator function involved in the canonical selection procedure (eq. (2)).

Lemma 2.3. *Consider the approximation functions for $I(x \neq 0)$, $f(x) = |x|^{\varepsilon}$ and $g(x) = 1 - \varepsilon^{|x|}$ for some fixed $\varepsilon \in (0,1)$. The overall error for $f(x)$ is much larger than that of $g(x)$.*

Proof. We give a proof based on heuristic analysis. Because of symmetry we just focus on the right side of origin on the x-axis for error analysis. For an interval $[a, a+h]$, with $a > 1$, $h > 0$, we have $f(x) = x^{\varepsilon}$, so that the area under the curve in this interval is

$$\int_a^{a+h} f(x)dx = \frac{(a+h)^{\varepsilon+1} - a^{\varepsilon+1}}{\varepsilon+1}.$$

The area under the curve for the indicator function $I(x \neq 0)$ in this interval is h, so that the error in approximation is

$$\frac{(a+h)^{\varepsilon+1} - a^{\varepsilon+1}}{\varepsilon+1} - h \approx \frac{(\varepsilon+1)a^{\varepsilon}}{(\varepsilon+1)}h - h = (a^{\varepsilon} - 1)h,$$

where we used the leading term of the Taylor series approximation to the function $(a+x)^{\varepsilon+1}$. For the approximation function $g(x)$ the area under the curve in the interval $[a, a+h]$ with $a > 1$ is

$$\int_a^{a+h} g(x)dx = h - \frac{\varepsilon^a \left(\varepsilon^h - 1\right)}{\log(\varepsilon)}.$$

Using the leading term in the Taylor series approximation, $\varepsilon^h - 1$ can be approximated by $\log(\varepsilon)h$ so that the absolute value of the error is approximately $\varepsilon^a h$. Thus the absolute value of the error for $f(x)$ in a unit interval $[a, a+1]$ (with $a > 1$) is approximately $a^{\varepsilon} - 1$ and that for $g(x)$ in the same interval is approximately ε^a. For a fixed ε, the former can be made arbitrarily large, and the latter arbitrarily small by increasing a. The error for $g(x)$ is larger than that for $f(x)$ in the interval $[0,1]$ but the difference in this interval is bounded. □

Lemma 2.4. *Consider the approximation functions for $I(x \neq 0)$, $f(x) = |x|^{\varepsilon}$ and $g(x) = (2/\pi)\arctan(\gamma x)$ where $\gamma = \varepsilon^{-1}$, $\varepsilon \in (0,1)$ is fixed. The overall error for $f(x)$ is much larger than that of $g(x)$.*

Proof. In this case

$$\begin{aligned}
\int_a^{a+h} g(x)dx &= \frac{2}{\pi}\left[(a+h)\arctan(\gamma(a+h)) - a\arctan(a\gamma)\right] \\
&\quad - \frac{1}{\pi\gamma}\left[\log(1+\gamma^2(a+h)^2) - \log(1+\gamma^2 a^2)\right] \\
&\approx \frac{2h}{\pi}\left[\arctan(\gamma a) + \frac{\gamma a}{1+\gamma^2 a^2}\right] - \frac{2h}{\pi}\left[\frac{\gamma a}{1+\gamma^2 a^2}\right] \\
&= \frac{2h}{\pi}\left[\arctan(\gamma a)\right],
\end{aligned}$$

where the approximate equality above was obtained using the leading term in the Taylor series expansion of the each of the following functions:

$$f_1(x) = (a+x)\arctan[\gamma(a+x)] \text{ and } f_2(x) = \frac{1}{2\gamma}\log[1+\gamma^2(a+x)^2].$$

Thus the absolute value of the error for $g(x)$ in a unit interval $[a, a+1]$ (with $a > 1$) is approximately

$$\left|1 - \frac{2}{\pi}\arctan(\gamma a)\right|$$

which can be made arbitrarily small by increasing a, since for $x \geq 0$, $\arctan(x) < \pi/2$ and increases to $\pi/2$ as $x \to \infty$. On the other hand, as shown in the previous lemma, the corresponding error for $f(x)$ can be made arbitrarily large by increasing a. Also the error for $g(x)$ and $f(x)$ in the interval $[0, 1]$ is bounded. □

Two other non-convex penalties

Here we mention two other non-convex penalties that have appeared in the regularization literature previously. The first one is the Geman-McClure function

$$p(t) = \frac{|t|}{\sigma + |t|};$$

this function is exactly same as the function $p_1(t)$ mentioned above if we replace σ with $1/(-\log(\varepsilon))$. As mentioned above, the function $p_1(t)$ is related to the Laplace penalty via Lemma 2.1. For the same parameter ε the non-convexity for p_1 is twice as that for the Laplace penalty. It can also be shown that the derivative of the Laplace penalty converges to zero at a faster rate than the that of p_1. Based on these considerations we did not study the Geman-McClure function in this paper.

Yet another non-convex penalty that has appeared in the literature is the concave logarithmic penalty

$$p(t) = \log\left(\frac{|t|}{\sigma} + 1\right), \ \sigma > 0.$$

This function increases with the absolute value of the argument like L_1 and L_2 penalties; although the increase is at a lower rate than L_1 and L_2 for large $|t|$, it is still an increasing function thereby resulting in bias. Hence we do not consider this latter penalty as well in this paper.

3. Experimental results

We assess the performance of regularized DNNs with the non-convex penalty functions presented in this paper, by applying them on a real dataset (MNIST). Details of the analysis and description of the MNIST dataset are given below.

The optimal weights of the fitted deep neural networks (DNN) were estimated by minimizing the total cross entropy loss function. We used batch gradient descent algorithm with early stopping. To avoid the vanishing/exploding gradients problem, the weights were initialized to values obtained from a normal distribution with mean zero and variance $4/(n_i + n_{(i-1)})$ where n_i is the number of neurons in the i^{th} layer [16, 17]. Rectified linear units (ReLU) function was used as the activation function.

The training data was randomly split into multiple batches. During each epoch, the gradient descent algorithm was sequentially applied to each of these batches resulting in new weights estimates. At the end of each epoch, the total validation loss was calculated using the validation set. When twenty consecutive epochs failed to improve the total validation loss, the iteration was stopped. The maximum number of epochs was set at 250. The weights estimate that resulted in the lowest total validation loss was selected as the final estimate. Since there was a random aspect to the way the training sets were split into batches, the whole process was repeated three times with seed values 1, 2, and 3. The reported test error rates are the median of the three test error rates obtained using each of these seed values.

A triangular learning rate schedule was used because it produced the lowest test error rates [18]. The learning rates varied from a minimum of 0.01 to a maximum of 0.25 (see figure 1 below).

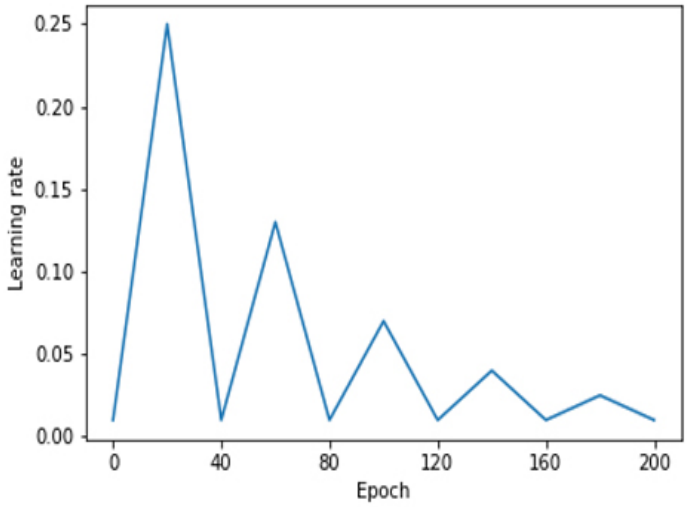

Figure 1: Learning rate plot

For all penalty functions the optimal λ was found by fitting models with logarithmically equidistant values in a grid. We used Python ver. 3.6.7rc2 and TensorFlow ver.1.12.0 for the calculations.

The models were fit with no regularization, L_1 and L_2 regularizations and the non-convex regularization methods. The results based on new non-convex penalty functions were comparable to L_1 and L_2 regularization in all the datasets. A general overview of the dataset and the DNN model specifications is given in Table 1. The models were intentionally over-parameterized to better contrast the effects of various types of regularization methods.

Table 1. Overview of dataset and DNN model specifications							
Dataset	Domain	Dimensionality	Classes	DNN Specifications	Training Set	Validation Set	Test Set
MNIST	Visual	784 (28×28 greyscale)	10	5 layers, 1024 units	48000	2000	10000

MNIST:
Modified National Institute of Standards and Technology (MNIST) dataset is a widely used toy dataset of 60,000 grey-scale images of hand-written digits. Each image has $28 \times 28 = 784$ pixels. The intensity measures of these 784 pixels form the input variables of the model. The dataset was split into 48,000 training set, 2000 validation set, and 10,000 test set.

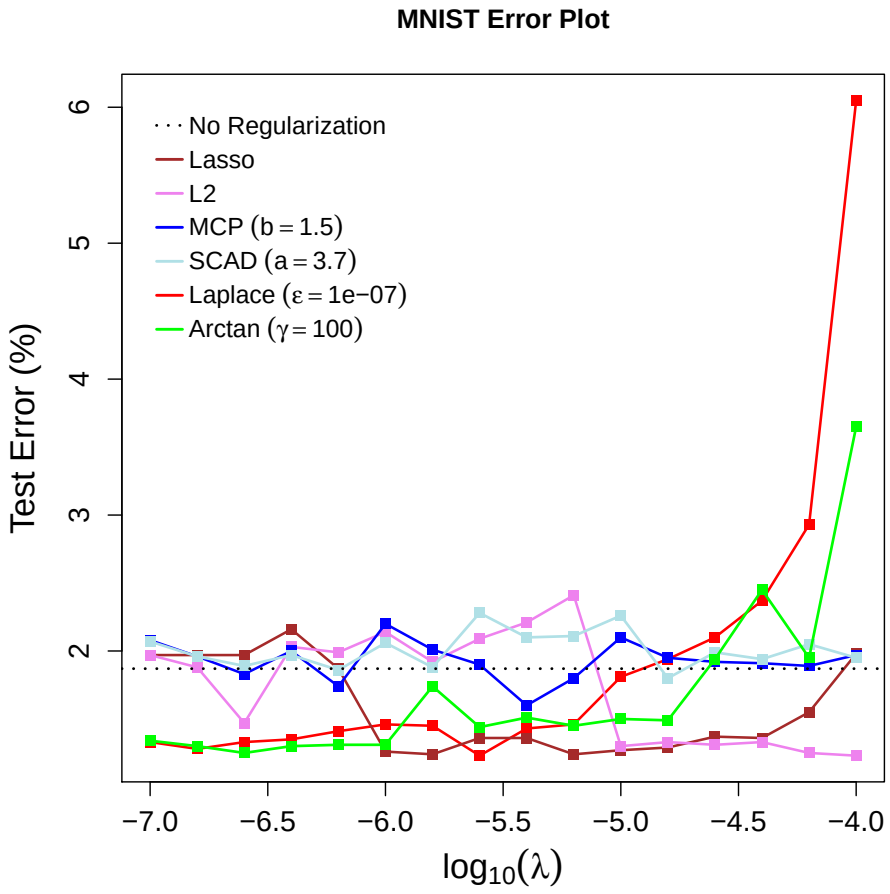

Figure 2: Error plot for MNIST dataset

The test error rate obtained with no regularization was 1.87%. With Lasso regularization, the test error reduced to 1.24% and with Ridge regularization the test error was 1.23% The Laplace and Arctan methods gave test error rates of 1.23% and 1.25% which were comparable to Lasso and Ridge.

All the results mentioned above are summarized in Table 2 below.

Table 2. Median test error rates at optimal λ							
Dataset	Regularization method						
	None	L_1	L_2	SCAD	MCP	Laplace	Arctan
MNIST	1.87	1.24	1.23	1.80	1.60	1.23	1.25

4. Conclusion

Non-convex regularizers were originally considered in statistical literature after observing certain limitations of the convex regularizers from the class of Bridge functions. Yet, non-convex regularizers never gained as much popularity as their convex counterparts in DNN applications, mainly because of certain perceived computational and optimization limitations - that is, in the presence of local optima which are not global optima, in the case of non-convex functions, iterative methods such as gradient or coordinate descent may terminate undesirably in local optima. However, recent theoretical work [8,9] that established regularity conditions under which both local and global minimum lie within a small neighborhood of the true minimum have brought the limelight back onto non-convex regularizers. The new theory eliminates the need for specially designed optimization algorithms for most non-convex regularizers as it implies that standard first-order optimization methods will converge to points within statistical error of the truth. In other words, non-convex regularizers that satisfy such regularity conditions enjoy guarantees for both statistical accuracy and optimization efficiency.

Penalty functions typically considered for regularization of DNN are convex. In this paper, we present non-convex penalty functions (Laplace, Arctan, SCAD and MCP) that are typically not considered in the DNN literature. Arctan penalty function has not been considered in any statistic literature previously to the best of our knowledge. We study the performance of the non-convex penalty functions while applying DNN on a large dataset (MNIST). Test error rates for Laplace and Arctan penalty functions were comparable to that obtained by the convex penalties.

References

References

[1] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. Nature, 323, 533-536.

[2] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning internal representations by error propagation. In Rumelhart, D. E. and McClelland, J. L., editors, Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1: Foundations Volume 1: Foundations, MIT Press, Cambridge, MA.

[3] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. J. Royal. Statist. Soc B., Vol. 58, No. 1, pages 267-288.

[4] Hoerl, A.E. and Kennard, R. (1970). Ridge regression: Biased estimation for nonorthogonal problems. Technometrics,12:55-67

[5] Frank, I. and Friedman, J. (1993). A statistical view of somechemometrics regression tools.Technometrics,35, 109–148.

[6] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. Journal of the American Statistical Association, 96:1348–1360, 2001.

[7] C.-H. Zhang. Nearly unbiased variable selection under minimax concave penalty. Annals of Statistics, 38(2):894–942, 2010.

[8] P. Loh and M.J. Wainwright. Regularized M-estimators with nonconvexity: Statistical and algorithmic theory for local optima. Journal of Machine Learning Research 16 (2015) 559–616.

[9] P. Loh and M. J. Wainwright. Support recovery without incoherence: A case for nonconvex regularization. Annals of Statistics 45(6): 2455-2482, 2017.

[10] Fu, W. J.(1998). Penalized regressions: the Bridge versus the Lasso.J. Comput. Graph. Statist.7397–416.

[11] Knight K and Fu W (2000). Asymptotics for lasso-type estimators Ann. Statist. 28, no. 5, 1356-1378.

[12] Trzasko, J and Manduca, A. Highly undersampled magnetic resonance image reconstruction via homotopic L0-minimization. IEEE Transactions on Medical Imaging. Vol 28, Issue: 1, Jan. 2009

[13] Lu, C., Tang, J., Yan, S. and Lin, Z. 2014. Generalized Nonconvex Nonsmooth Low-Rank Minimization. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14). IEEE Computer Society, Washington, DC, USA, 4130-4137.

[14] Geron, A. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media; 1 edition (2017)

[15] Geman, D and Yang, C. Nonlinear image recovery with half-quadratic regularization. IEEE Transactions on Image Processing (Volume: 4, Issue: 7, Jul 1995.

[16] Glorot, X., Bengio, Y. Understanding the difficulty of training deep feedforward neural networks

[17] He, K., Zhang, X., Ren, S., Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification.

[18] Smith, L.N. Cyclical Learning Rates for Training Neural Networks. DOI: 10.1109/WACV.2017.58 https://ieeexplore.ieee.org/document/7926641

[19] Lewis, D. D., Yang, Y., Rose, T. G., Li, F. (2004). RCV1: A new benchmark collection for text categorization research. The Journal of Machine Learning Research, 5, 361-397.

[20] Tarigan, B. and van de Geer, S. (2006) Classifiers of support vector machine type withl1complexity regularization.Bernoulli,12, 1045–1076.

[21] L. Meier, S. van de Geer and P. Buehlmann (2008). The group Lasso for logistic regression. JRSS, Series B, 70, 53-71

Appendix A: Statistical Consistency

Statistical consistency results for the weight estimates based on penalty functions in eq.(3) and eq.(4) can be obtained by modifying slightly existing theoretical results [20,21] in the literature. Asymptotic results for SCAD and MCP are presented in [6] and [7]. So, we focus on only Laplace and arctan penalties. Consider the class of logistic classifiers,

$$\mathcal{F} = \{\eta_\beta : \mathbb{R}^p \to \mathbb{R} \mid \eta_\beta(\mathbf{x}) = \mathbf{x}^T\beta,\ \mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^p,\ \beta \in \mathcal{B} \subseteq \mathbb{R}^p\}.$$

Classification is done based on the sign of of the function $f_\beta : \mathbb{R}^p \to \mathbb{R}$ defined as

$$f_\beta(\mathbf{x}) = \frac{1}{1+\exp(-\eta_\beta(\mathbf{x}))} - \frac{1}{2}.$$

Here

$$\pi_\beta(\mathbf{x}) = \frac{1}{1+\exp(-\eta_\beta(\mathbf{x}))} = \frac{\exp(\eta_\beta(\mathbf{x}))}{1+\exp(\eta_\beta(\mathbf{x}))}$$

denotes the class probability. If there are $K > 2$ classes, then the class probability for the k^{th} class may be modeled as

$$\frac{\exp(\eta_{\beta_k}(\mathbf{x}))}{\sum_{m=1}^{K}\exp(\eta_{\beta_m}(\mathbf{x}))},$$

but for simplicity, we just focus on binary classification.

We assume that $\mathcal{X}$ is endowed with a probability measure ν and let $||\cdot||_{p,\nu}$ be the $L_p(\nu)$ norm ($1 \le p < \infty$). Denote $\Sigma = \mathbb{E}_\nu(\mathbf{x}^T\mathbf{x})$. Design matrix $\mathbf{X} = [\mathbf{x}_1^T, \ldots, \mathbf{x}_n^T]^T$ consists of n copies of $\mathbf{x}$. The empirical logistic (also known as cross-entropy) loss is

$$R_n(\eta_\beta) = \frac{1}{n}\sum_{i=1}^{n} l(y_i, \eta_\beta(\mathbf{x}_i)),\ \text{where } l(y_i, \eta_\beta(\mathbf{x}_i)) = -[y_i\pi_\beta(\mathbf{x}_i) + (1-y_i)(1-\pi_\beta(\mathbf{x}_i))]$$

and theoretical loss

$$R(\eta_\beta) = \mathbb{E}(R_n(\eta_\beta)).$$

Let

$$\eta^* = \underset{\text{all } \eta}{\operatorname{argmin}}\, R(\eta).$$

We assume the following three conditions given in [20].

(C1): There exists constants $\sigma > 0$ and $\kappa \geq 1$, such that for all $\eta \in \mathcal{F}$,

$$R(\eta) - R(\eta^*) \geq \frac{||\eta - \eta^*||_{1,\nu}^{\kappa}}{\sigma^{\kappa}}.$$

(C2): The smallest eigenvalue ρ^2 of Σ is non-zero.

(C3): $\max_{1\leq k\leq p} ||\mathbf{x}e_k||_\infty \leq \sqrt{\frac{n}{\log n}}$, $\max_{1\leq k\leq p} ||\mathbf{x}e_k||_2 < \infty$, $p < n^D$ for some D. Here e_k denotes the unit vector with 1 as the k^{th} element and 0's elsewhere.

The following theorem holds for $p_\lambda(\beta)$, where $p_\lambda(\beta)$ equals either the Laplace penalty function given in eq (3) or the arctan penalty function given in eq. (4).

Theorem 4.1. *Assume conditions C1 to C3 hold and that* $||\eta_\beta||_\infty \leq K$, $\forall \eta_\beta \in \mathcal{F}$. *Then for universal constants* c, c_1,

$$\mathbb{P}(R(\hat{\eta}_n) - R(\eta^*) > \varepsilon_n) \leq \frac{c_1}{n^2},$$

where

$$\hat{\eta}_n = \underset{\eta_\beta \in \mathcal{F}}{\operatorname{argmin}}\{R_n(\eta_\beta) + p_{\hat{\lambda}_n}(\beta))\},\ \hat{\lambda}_n = c(\hat{C}_n \vee 4)DK^2\sqrt{\frac{\log(n)}{n}},\ \hat{C}_n = \frac{1}{n}\max_{1\leq k\leq p} ||\mathbf{x}e_k||_2^2;$$

$$\varepsilon_n = (1+4\delta)\inf\left\{R(\eta_\beta) - R(\eta^*) + V_n(N(\beta)) + 2\lambda_n K\sqrt{\frac{\log(n)}{n}} : \eta_\beta \in \mathcal{F}\right\},\ \delta \in (0, 0.5];$$

$$V_n(N) = 2\delta^{-1/(2\kappa-1)}(18\sigma\lambda_n^2 LNDK/\rho^2)^{\kappa/(2\kappa-1)},\ \kappa \geq 1;\ N(\beta) = \#(\text{nonzero elements in } \beta). \tag{9}$$

The constant L *in eq. (7) depends on the penalty function:* $L = (\log(\varepsilon))^2$ *for the Laplace penalty and* $L = (2\gamma/\pi)^2$ *for the arctan penalty.*

Proof. We give only a sketch of the proof, as the proof is the same as the lengthy proof given in [20], with only minor differences. First of all note that the only difference in the statement of the above theorem from the statement of the theorem 1 in [20] is the constant L in eq. (9).

Although the loss function used in [20] was Hinge-loss function, the steps in their proof follows for logistic loss also (-actually it becomes easier-) as pointed out in [21]. Thus, the only difference in the steps in the proof that we need to focus are those corresponding to the penalty functions. Their theorem is stated for the Lasso penalty. The triangle inequality satisfied by the Lasso penalty is used in certain steps of the proof. But, since both Laplace and arctan penalties are subadditive (- concave in the positive real line, with $p_\lambda(0) = 0$ -), those steps hold true for these two penalties as well.

The only other step we need to focus is Lemma 5.2 in their proof, where the key inequality used is

$$\left(\sum_{t=1}^{T}|\beta_t|\right)^2 \leq T\sum_{t=1}^{T}\beta_t^2. \tag{10}$$

Instead, we use the inequalities

$$\left(\sum_{t=1}^{T}\left(1-\varepsilon^{|\beta_t|}\right)\right)^2 \leq LT\sum_{t=1}^{T}\beta_t^2, \text{ where } L=(\log(\varepsilon))^2 \tag{11}$$

and

$$\left(\sum_{t=1}^{T}\left(\frac{2}{\pi}\arctan(\gamma|\beta_t|)\right)\right)^2 \leq LT\sum_{t=1}^{T}\beta_t^2, \text{ where } L=\left(\frac{2\gamma}{\pi}\right)^2. \tag{12}$$

Inequality in (11) follows from the inequality in (10) and the fact that

$$1-\varepsilon^x \leq -(\log\varepsilon)x, \text{ for } x\geq 0,$$

which follows easily by considering the function $f(x)=-(\log\varepsilon)x-1+\varepsilon^x$ and noting that $f(0)=0$ and

$$f'(x)=-\log\varepsilon+\varepsilon^x\log\varepsilon=(-\log\varepsilon)(1-\varepsilon^x)\geq 0, \text{ for } x\geq 0 \text{ and } \varepsilon\in(0,1).$$

Inequality in (12) follows from the inequality in (10) and the right inequality in Lemma 2.2.

□

Appendix B: Tables

DNN analysis was repeated for multiple seed values. The test error rate presented in section 3.3 was the median of the test error rates from all seed values. Detailed results (i.e. test error rates for each grid point and seed value) used for compiling the summarized table in section 3.3 are presented below.

4.1. MNIST

MNIST results with no regularization											
Seed	1	2	3	4	5	6	7	8	9	10	Median
Error	1.69	2.54	1.76	2.38	1.70	1.98	2.34	1.72	2.56	1.76	1.87

MNIST results with Lasso regularization				
$\log(\lambda)$	Seed = 1	Seed = 2	Seed = 3	Median
-4.00	1.98	1.98	1.69	1.98
-4.20	1.55	1.55	1.52	1.55
-4.40	1.45	1.36	1.32	1.36
-4.60	1.37	1.34	1.41	1.37
-4.80	1.28	1.40	1.29	1.29
-5.00	1.44	1.27	1.24	1.27
-5.20	1.16	1.34	1.24	1.24
-5.40	1.45	1.36	1.35	1.36
-5.60	1.36	1.42	1.34	1.36
-5.80	1.24	1.32	1.21	1.24
-6.00	1.26	1.31	1.24	1.26
-6.20	1.72	2.09	1.87	1.87
-6.40	2.16	1.78	2.31	2.16
-6.60	2.14	1.97	1.41	1.97
-6.80	1.97	1.84	2.45	1.97
-7.00	1.84	1.97	2.03	1.97

MNIST results with L_2 regularization				
$\log(\lambda)$	Seed = 1	Seed = 2	Seed = 3	Median
-4.00	1.26	1.23	1.22	1.23
-4.20	1.36	1.25	1.22	1.25
-4.40	1.45	1.33	1.29	1.33
-4.60	1.31	1.28	1.34	1.31
-4.80	1.33	1.35	1.26	1.33
-5.00	1.90	1.30	1.27	1.30
-5.20	1.70	2.45	2.41	2.41
-5.40	2.21	2.16	2.43	2.21
-5.60	2.09	2.23	2.02	2.09
-5.80	2.43	1.71	1.92	1.92
-6.00	2.14	2.07	2.15	2.14
-6.20	1.99	1.96	2.38	1.99
-6.40	1.83	2.05	2.03	2.03
-6.60	2.02	1.38	1.47	1.47
-6.80	1.88	2.40	1.83	1.88
-7.00	2.19	1.40	1.97	1.97

MNIST results with Laplace (ε = 1e-07) regularization				
$\log(\lambda)$	Seed = 1	Seed = 2	Seed = 3	Median
-4.00	6.05	5.78	6.43	6.05
-4.20	3.07	2.70	2.93	2.93
-4.40	2.29	2.37	2.41	2.37
-4.60	2.33	2.10	1.93	2.10
-4.80	2.04	1.77	1.94	1.94
-5.00	1.81	1.86	1.79	1.81
-5.20	1.46	1.53	1.39	1.46
-5.40	1.35	1.43	1.45	1.43
-5.60	1.18	1.36	1.23	1.23
-5.80	1.47	1.42	1.45	1.45
-6.00	1.46	1.46	1.35	1.46
-6.20	1.45	1.40	1.41	1.41
-6.40	1.33	1.35	1.35	1.35
-6.60	1.33	1.36	1.30	1.33
-6.80	1.25	1.28	1.34	1.28
-7.00	1.19	1.40	1.33	1.33

MNIST results with Arctan (γ = 100) regularization				
$\log(\lambda)$	Seed = 1	Seed = 2	Seed = 3	Median
-4.00	3.43	3.65	5.71	3.65
-4.20	1.95	3.62	1.75	1.95
-4.40	2.57	1.66	2.45	2.45
-4.60	1.94	2.64	1.36	1.94
-4.80	1.51	1.43	1.49	1.49
-5.00	1.47	1.50	2.11	1.50
-5.20	1.41	1.45	2.02	1.45
-5.40	1.51	1.51	1.50	1.51
-5.60	1.44	1.39	1.57	1.44
-5.80	1.88	1.74	1.57	1.74
-6.00	1.30	1.34	1.31	1.31
-6.20	1.35	1.31	1.31	1.31
-6.40	1.38	1.30	1.29	1.30
-6.60	1.33	1.25	1.25	1.25
-6.80	1.39	1.30	1.27	1.30
-7.00	1.34	1.27	1.34	1.34

MNIST results with SCAD ($a = 3.7$) regularization				
$\log(\lambda)$	Seed = 1	Seed = 2	Seed = 3	Median
-4.00	1.95	2.29	1.92	1.95
-4.20	2.07	2.05	1.78	2.05
-4.40	1.94	2.29	1.92	1.94
-4.60	2.08	1.99	1.67	1.99
-4.80	1.70	1.80	1.95	1.80
-5.00	2.34	2.26	2.08	2.26
-5.20	2.28	2.11	1.94	2.11
-5.40	2.29	2.01	2.10	2.10
-5.60	2.28	2.39	1.83	2.28
-5.80	2.02	1.88	1.58	1.88
-6.00	2.32	2.06	1.94	2.06
-6.20	2.12	1.74	1.86	1.86
-6.40	2.20	1.97	1.94	1.97
-6.60	1.89	1.61	2.12	1.89
-6.80	2.00	1.44	1.96	1.96
-7.00	2.07	2.08	2.00	2.07

MNIST results with MCP ($b = 1.5$) regularization				
$\log(\lambda)$	Seed = 1	Seed = 2	Seed = 3	Median
-4.00	1.72	1.97	2.09	1.97
-4.20	2.19	1.89	1.84	1.89
-4.40	2.16	1.88	1.91	1.91
-4.60	1.94	1.92	1.85	1.92
-4.80	1.82	2.38	1.95	1.95
-5.00	2.10	1.93	2.34	2.10
-5.20	1.76	2.28	1.80	1.80
-5.40	2.09	1.33	1.60	1.60
-5.60	1.80	2.12	1.90	1.90
-5.80	2.01	1.83	2.12	2.01
-6.00	2.24	2.20	1.65	2.20
-6.20	2.16	1.46	1.74	1.74
-6.40	2.00	1.67	2.15	2.00
-6.60	2.02	1.78	1.83	1.83
-6.80	2.02	1.96	1.89	1.96
-7.00	2.02	2.08	2.56	2.08