OptNet: Supplementary Material

Brandon Amos   J. Zico Kolter

A. MNIST Experiment

In this section we consider the integration of QP OptNet layers into a traditional fully connected network for the MNIST problem. The results here show only very marginal improvement if any over a fully connected layer (MNIST, after all, is very fairly well-solved by a fully connected network, let alone a convolution network). But our main point of this comparison is simply to illustrate that we can include these layers within existing network architectures and efficiently propagate the gradients through the layer.

Specifically we use a FC600-FC10-FC10-SoftMax fully connected network and compare it to a FC600-FC10-Optnet10-SoftMax network, where the numbers after each layer indicate the layer size. The OptNet layer in this case includes only inequality constraints and the previous layer is only used in the linear objective term $p(z_i) = z_i$. To keep $Q \succ 0$, we use a Cholesky factorization $Q = LL^T + \epsilon I$ and directly learn L (without any information from the previous layer). We also directly learn A and G, and to ensure a feasible solution always exists, we select some learnable z_0 and s_0 and set $b = Az_0$ and $h = Gz_0 + s_0$.

Figure 6 shows that the results are similar for both networks with slightly lower error and less variance in the OptNet network.

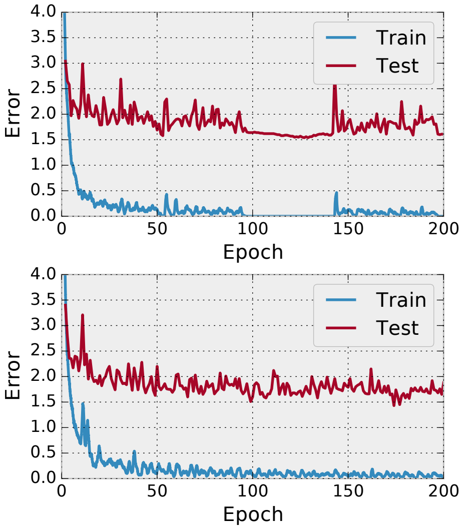

Figure 6. Training performance on MNIST; top: fully connected network; bottom: OptNet as final layer.)

B. Denoising Experiment Details

Figure 7 shows the error of the fully connected network on the denoising task and Figure 8 shows the error of the OptNet fine-tuned TV solution.

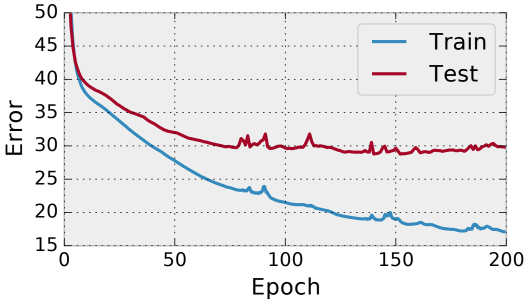

Figure 7. Error of the fully connected network for denoising

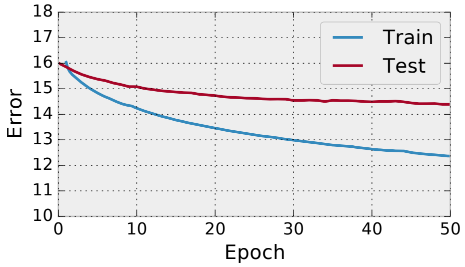

Figure 8. Error rate from fine-tuning the TV solution for denoising

C. Representational power of the QP OptNet layer

This section contains proofs for those results we highlight in Section 3.2. As mentioned before, these proofs are all quite straightforward and follow from well-known properties, but we include them here for completeness.

C.1. Proof of Theorem 1

Proof. The fact that an OptNet layer is subdifferentiable from strictly convex QPs ($Q \succ 0$) follows directly from the well-known result that the solution of a strictly convex QP is continuous (though not everywhere differentiable). Our proof essentially just boils down to showing this fact, though we do so by explicitly showing that there *is* a unique solution to the Jacobian equations (6) that we presented earlier, except on a measure zero set. This measure zero set consists of QPs with degenerate solutions, points where inequality constraints can hold with equality yet also have zero-valued dual variables. For simplicity we assume that A has full row rank, but this can be relaxed.

From the complementarity condition, we have that at a primal dual solution $(z^\star, \lambda^\star, \nu^\star)$

$$\begin{aligned} (Gz^\star - h)_i < 0 \rightarrow \lambda_i^\star = 0 \\ \lambda_i^\star > 0 \rightarrow (Gz^\star - h)_i = 0 \end{aligned} \tag{13}$$

(i.e., we cannot have both these terms non-zero).

First we consider the (typical) case where exactly one of $(Gz^\star - h)_i$ and $\lambda_i^\star$ is zero. Then the KKT differential matrix

$$\begin{bmatrix} Q & G^T & A^T \\ D(\lambda^\star)G & D(Gz^\star - h) & 0 \\ A & 0 & 0 \end{bmatrix} \tag{14}$$

(the left hand side of (6)) is non-singular. To see this, note that if we let $\mathcal{I}$ be the set where $\lambda_i^\star > 0$, then the matrix

$$\begin{aligned} &\begin{bmatrix} Q & G_{\mathcal{I}}^T & A^T \\ D(\lambda^\star)G_{\mathcal{I}} & D(Gz^\star - h)_{\mathcal{I}} & 0 \\ A & 0 & 0 \end{bmatrix} = \\ &\begin{bmatrix} Q & G_{\mathcal{I}}^T & A^T \\ D(\lambda^\star)G_{\mathcal{I}} & 0 & 0 \\ A & 0 & 0 \end{bmatrix} \end{aligned} \tag{15}$$

is non-singular (scaling the second block by $D(\lambda^\star)^{-1}$ gives a standard KKT system (Boyd & Vandenberghe, 2004, Section 10.4), which is nonsingular for invertible Q and $[G_{\mathcal{I}}^T \; A^T]$ with full column rank, which must hold due to our condition on A and the fact that there must be less than n total tight constraints at the solution. Also note that for any $i \notin \mathcal{I}$, only the $D(Gz^\star - h)_{ii}$ term is non-zero for the entire row in the second block of the matrix. Thus, if we want to solve the system

$$\begin{bmatrix} Q & G_{\mathcal{I}}^T & A^T \\ D(\lambda^\star)G_{\mathcal{I}} & D(Gz^\star - h)_{\mathcal{I}} & 0 \\ A & 0 & 0 \end{bmatrix} \begin{bmatrix} z \\ \lambda \\ \nu \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \tag{16}$$

we simply first set $\lambda_i = b_i/(Gz^\star - h)_i$ for $i \notin \mathcal{I}$ and then solve the nonsingular system

$$\begin{bmatrix} Q & G_{\mathcal{I}}^T & A^T \\ D(\lambda^\star)G_{\mathcal{I}} & 0 & 0 \\ A & 0 & 0 \end{bmatrix} \begin{bmatrix} z \\ \lambda_{\mathcal{I}} \\ \nu \end{bmatrix} = \begin{bmatrix} a - G_{\bar{\mathcal{I}}}^T \lambda_{\bar{\mathcal{I}}} \\ b_{\mathcal{I}} \\ c. \end{bmatrix} \tag{17}$$

Alternatively, suppose that we have both $\lambda_i^\star = 0$ and $(Gz^\star - h)_i = 0$. Then although the KKT matrix is now singular (any row for which $\lambda_i^\star = 0$ and $(Gz^\star - h)_i = 0$ will be all zero), there still exists a solution to the system (6), because the right hand side is always in the range of $D(\lambda^\star)$ and so will also be zero for these rows. In this case there will no longer be a *unique* solution, corresponding to the subdifferentiable but not differentiable case. □

C.2. Proof of Theorem 2

Proof. The proof that an OptNet layer can represent any piecewise linear univariate function relies on the fact that we can represent any such function in "sum-of-max" form

$$f(x) = \sum_{i=1}^{k} w_i \max\{a_i x + b, 0\} \tag{18}$$

where $w_i \in \{-1, 1\}$, $a_i, b_i \in \mathbb{R}$ (to do so, simply proceed left to right along the breakpoints of the function adding a properly scaled linear term to fit the next piecewise section). The OptNet layer simply represents this function directly.

That is, we encode the optimization problem

$$\begin{aligned} \underset{z \in \mathbb{R}, t \in \mathbb{R}^k}{\text{minimize}} \quad & \|t\|_2^2 + (z - w^T t)^2 \\ \text{subject to} \quad & a_i x + b_i \leq t_i, \quad i = 1, \ldots, k \end{aligned} \tag{19}$$

Clearly, the objective here is minimized when $z = w^T t$, and t is as small as possible, meaning each t must either be at its bound $a_i x + b \leq t_i$ or, if $a_i x + b < 0$, then $t_i = 0$ will be the optimal solution due to the objective function. To obtain a multivariate but elementwise function, we simply apply this function to each coordinate of the input x.

To see the specific case of a ReLU network, note that the layer

$$z = \max\{Wx + b, 0\} \tag{20}$$

is simply equivalent to the OptNet problem

$$\begin{aligned} \underset{z}{\text{minimize}} \quad & \|z - Wx - b\|_2^2 \\ \text{subject to} \quad & z \geq 0. \end{aligned} \tag{21}$$

□

C.3. Proof of Theorem 3

Proof. The final theorem simply states that a two-layer ReLU network (more specifically, a ReLU followed by a linear layer, which is sufficient to achieve a universal function approximator), can often require exponentially many more units to approximate a function specified by an OptNet layer. That is, we consider a single-output ReLU network, much like in the previous section, but defined for multi-variate inputs.

$$f(x) = \sum_{i=1}^{m} w_i \max\{a_i^T x + b, 0\} \tag{22}$$

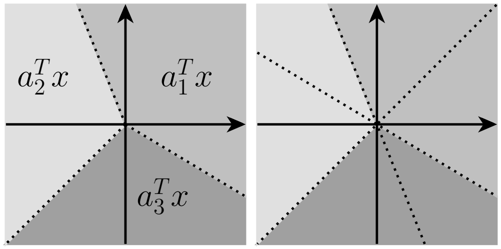

Figure 9. Creases for a three-term pointwise maximum (left), and a ReLU network (right).

Although there are many functions that such a network cannot represent, for illustration we consider a simple case of a maximum of three linear functions

$$f'(x) = \max\{a_1^T x, a_2^T x, a_3^T x\} \tag{23}$$

To see why a ReLU is not capable of representing this function exactly, even for $x \in \mathbb{R}^2$, note that any sum-of-max function, due to the nature of the term $\max\{a_i^T x + b_i, 0\}$ as stated above must have "creases" (breakpoints in the piecewise linear function), than span the entire input space; this is in contrast to the max terms, which can have creases that only partially span the space. This is illustrated in Figure 9. It is apparent, therefore, that the two-layer ReLU cannot exactly approximate the three maximum term (any ReLU network would necessarily have a crease going through one of the linear region of the original function). Yet this max function can be captured by a simple OptNet layer

$$\begin{aligned} \underset{z}{\text{minimize}} \quad & z^2 \\ \text{subject to} \quad & a_i^T x \leq z, \; i = 1, \ldots, 3. \end{aligned} \tag{24}$$

The fact that the ReLU network is a universal function approximator means that the we *are* able to approximate the three-max term, but to do so means that we require a dense covering of points over the input space, choose an equal number of ReLU terms, then choose coefficients such that we approximate the underlying function on this points; however, for a large enough radius this will require an exponential size covering to approximate the underlying function arbitrarily closely. □

Although the example here in this proof is quite simple (and perhaps somewhat limited, since for example the function can be exactly approximated using a "Maxout" network), there are a number of other such functions for which we have been unable to find any compact representation. For example, projection of a point on to the simplex is easily written as the OptNet layer

$$\begin{aligned} \underset{z}{\text{minimize}} \quad & \|z - x\|_2^2 \\ \text{subject to} \quad & z \geq 0, 1^T z = 1 \end{aligned} \tag{25}$$

yet it does not seem possible to represent this in closed form as a simple network: the closed form solution of such a projection operator requires sorting or finding a particular median term of the data (Duchi et al., 2008), which is not feasible with a single layer for any form of network that we are aware of. Yet for simplicity we stated the theorem above using just ReLU networks and a straightforward example that works even in two dimensions.