A. Proofs

A.1. Proof of Theorem 1

Proof. First we show that the multiplication of the input $x \in \mathbb{R}^d$ by a multi-level quantized weight vector $q \in \mathcal{Q}_M^d$ can be represented by the dot product of a function of the input, i.e., $\tilde{x}$ and a binary quantized weight vector u, that is, $q^T x = u^T \tilde{x}$. Here, u is a binary vector of size dM with entries satisfying

$$q_i := \sum_{k=1}^{M} u_{k+(i-1)M},\ i = 1, \ldots, d\,. \tag{27}$$

For instance, for $M = 4$, we have $q_1 = u_1 + u_2 + u_3 + u_4$. Note that because u_j's are from the set $\{-1, +1\}$, we have that $q_1 \in \{-4, -2, 0, 2, 4\}$, which is equal to the set for $(4 + 1 = 5)$-level quantization, i.e., $\mathcal{Q}_4$. The second entry of the q vector similarly satisfies $q_2 = u_5 + u_6 + u_7 + u_8 \in \mathcal{Q}_4$. The same holds for all the entries $q_1, \ldots, q_d$.

Next, plugging in (27) in the dot product $q^T x$ yields

$$\begin{aligned} q^T x = \sum_{i=1}^{d} q_i x_i &= \sum_{i=1}^{d}\sum_{k=1}^{M} u_{k+(i-1)M} x_i \\ &= \sum_{i=1}^{d}\sum_{k=1}^{M} u_{k+(i-1)M} \tilde{x}_{k+(i-1)M} \\ &= u^T \tilde{x} \end{aligned} \tag{28}$$

where we defined $\tilde{x} := \begin{bmatrix} x_1, x_1, \ldots, x_1, x_2, x_2, \ldots, x_2, \ldots, x_d, x_d, \ldots, x_d \end{bmatrix}^T \in \mathbb{R}^{dM}$. This shows that the dot product $q^T x$ is equal to the dot product $u^T \tilde{x}$ where u is a dM-dimensional vector with binary entries.

The input-output relationship for the two-layer fully connected neural network with polynomial activation is $f(x) = \sum_{j=1}^{m} \sigma(x^T q_j)\alpha_j = \sum_{j=1}^{m} \left(a q_j^T x x^T q_j + b q_j^T x + c\right)\alpha_j$ where $q_j \in \mathcal{Q}_M^d$ and $\alpha_j \in \mathbb{R}$, $j = 1, \ldots, m$. Using the fact that we can represent a dot product with multi-level quantized weights as a dot product with binary quantized weights, we equivalently have

$$f(x) = \sum_{j=1}^{m} \left(a u_j^T \tilde{x}\tilde{x}^T u_j + b u_j^T \tilde{x} + c\right)\alpha_j\,. \tag{29}$$

We can rewrite this as a neural network with quadratic activation:

$$\begin{aligned} f(x) &= \sum_{j=1}^{m} \begin{bmatrix} u_j^T & 1 \end{bmatrix} \begin{bmatrix} a\tilde{x}\tilde{x}^T & \frac{b}{2}\tilde{x} \\ \frac{b}{2}\tilde{x}^T & c \end{bmatrix} \begin{bmatrix} u_j \\ 1 \end{bmatrix} \alpha_j \\ &= \sum_{j=1}^{m} \tilde{u}_j^T \mathcal{X} \tilde{u}_j \alpha_j \end{aligned} \tag{30}$$

where we have defined $\tilde{u}_j \in \{-1, +1\}^{dM+1}$, $j = 1, \ldots, m$, and $\mathcal{X} \in \mathbb{R}^{(dM+1)\times(dM+1)}$.

This representation can be seen as a bilinear activation network with $u_j' = u_j$ and $v_j' = u_j$, $j = 1, \ldots, m$. The proof of the converse follows from the symmetrization identity (7). ☐

A.2. Proof of Theorem 3

Proof. We begin by applying the matrix Bernstein concentration bound on the matrices $(u_j v_j^T - \mathbb{E}[u_j v_j^T])$, $j = 1, \ldots, m$, which we note are $(d \times d)$-dimensional zero-mean i.i.d. matrices. We obtain the following upper bound on the spectral norm of these matrices

$$\begin{aligned} \|u_j v_j^T - \mathbb{E}[u_j v_j^T]\| &\leq \|u_j v_j^T\|_2 + \|\,\mathbb{E}[u_j v_j^T]\|_2 \\ &\leq \|u_j v_j^T\|_2 + \mathbb{E}[\|u_j v_j^T\|_2] \\ &= \|u_j\|_2\|v_j\|_2 + \mathbb{E}[\|u_j\|_2\|v_j\|_2] \\ &\leq d + d = 2d\,, \end{aligned} \tag{31}$$

for $j = 1, \dots, m$ where we use the triangle inequality in the first line and Jensen's inequality in the second line. Next, we define $S_j := u_j v_j^T - \mathbb{E}[u_j v_j^T]$ and $S := \sum_{j=1}^m S_j$, then the matrix variance of the sum (which we will plug in the matrix concentration bound formula) is given by

$$\sigma^2 = \max\{\|\mathbb{E}[SS^T]\|_2, \|\mathbb{E}[S^T S]\|_2\} = \max\left\{\left\|\sum_{j=1}^m \mathbb{E}[S_j S_j^T]\right\|_2, \left\|\sum_{j=1}^m \mathbb{E}[S_j^T S_j]\right\|_2\right\} \tag{32}$$

where the second equality follows because S_j's are zero-mean.

$$\begin{aligned}\mathbb{E}[S_j S_j^T] &= \mathbb{E}\left[\left(u_j v_j^T - \mathbb{E}[u_j v_j^T]\right)\left(u_j v_j^T - \mathbb{E}[u_j v_j^T]\right)^T\right] \\ &= d\,\mathbb{E}[u_j u_j^T] - \mathbb{E}[u_j v_j^T]\,\mathbb{E}[v_j u_j^T] \\ &= d\,\mathbb{E}[u_j u_j^T] - (2\gamma/\pi)^2 Z_s^* Z_s^{*T} \\ &= d\,\mathbb{E}[u_j u_j^T] - (2\gamma/\pi Z_s^*)^2 .\end{aligned} \tag{33}$$

Next, we bound the spectral norm of $\mathbb{E}[SS^T]$ as

$$\begin{aligned}\|\mathbb{E}[SS^T]\|_2 = \left\|\sum_{j=1}^m \mathbb{E}[S_j S_j^T]\right\|_2 &= \left\|\sum_{j=1}^m \left(d\,\mathbb{E}[u_j u_j^T] - (2\gamma/\pi Z_s^*)^2\right)\right\|_2 \\ &= \left\|md\,\mathbb{E}[u_1 u_1^T] - m(2\gamma/\pi Z_s^*)^2\right\|_2 \\ &\le md\left\|\mathbb{E}[u_1 u_1^T]\right\|_2 + \left\|m(2\gamma/\pi Z_s^*)^2\right\|_2 \\ &= md\left\|\mathbb{E}[u_1 u_1^T]\right\|_2 + m(2\gamma/\pi)^2\|Z_s^*\|_2^2 \\ &= md(2\gamma/\pi)\|\arcsin(Q_{(11)})\|_2 + m(2\gamma/\pi)^2\|Z_s^*\|_2^2 .\end{aligned} \tag{34}$$

The last line follows from the identity $\mathbb{E}[u_1 u_1^T] = 2\gamma/\pi \arcsin(Q_{(11)})$. We note that the upper bound for $\|\mathbb{E}[SS^T]\|_2$ is also an upper bound for $\|\mathbb{E}[S^T S]\|_2$. Hence, the matrix variance is upper bounded by $\sigma^2 \le c' md + m(2\gamma/\pi)^2\|Z_s^*\|_2^2$ where $c' \ge 0$ is a constant. Applying the matrix Bernstein concentration bound yields

$$\mathbb{P}\left[\left\|\sum_{j=1}^m (u_j v_j^T - \mathbb{E}[u_j v_j^T])\right\|_2 \ge m\epsilon\right] \le 2d\exp\left(\frac{-m^2\epsilon^2}{\sigma^2 + 2dm\epsilon/3}\right) . \tag{35}$$

Plugging in the expression for the variance, we obtain

$$\begin{aligned}\mathbb{P}\left[\left\|\frac{1}{m}\sum_{j=1}^m u_j v_j^T - \mathbb{E}[u_1 v_1^T]\right\|_2 \ge \epsilon\right] &\le 2d\exp\left(\frac{-m^2\epsilon^2}{c' md + m(2\gamma/\pi)^2\|Z_s^*\|_2^2 + 2dm\epsilon/3}\right) \\ &= 2d\exp\left(-\frac{m\epsilon^2}{(2\gamma/\pi)^2\|Z_s^*\|_2^2 + d(c' + 2\epsilon/3)}\right) \\ &= \exp\left(-\frac{m\epsilon^2}{(2\gamma/\pi)^2\|Z_s^*\|_2^2 + d(c' + 2\epsilon/3)} + \log(2d)\right) .\end{aligned} \tag{36}$$

Let us denote the optimal solution of the original non-convex problem as $Z_{nc}^* = \sum_{j=1}^m u_j^*(v_j^*)^T \alpha_j^*$ where the weights $u_j^*, v_j^* \in \{-1, +1\}^d, \alpha_j^* \in \mathbb{R}$, $j = 1, \dots, m$ are optimal network parameters for the non-convex combinatorial problem in (21) . Solving the SDP gives us an unquantized solution Z^* and via the sampling algorithm, we obtain the quantized solution given by $\hat{Z} = \sum_{j=1}^m \hat{u}_j \hat{v}_j^T \hat{\alpha}_j$.

We now introduce some notation. We will denote the loss term in the objective by $L(Z)$ and the regularization term by $R(Z)$, that is,

$$L(Z) := \ell\left(\begin{bmatrix} x_1^T Z x_1 \\ \vdots \\ x_n^T Z x_n \end{bmatrix}, y\right), \quad R(Z) := d\sum_{j=1}^m |\alpha_j| \quad \text{when} \quad Z = \sum_{j=1}^m u_j v_j^T \alpha_j . \tag{37}$$

We now bound the difference between the losses for the unquantized solution of the SDP, i.e., Z^*, and the quantized weights $\hat{Z} = \sum_{j=1}^m \hat{u}_j \hat{v}_j^T \hat{\alpha}_j$:

$$|L(\hat{Z}) - L(Z^*)| \leq L_c \left\| \begin{bmatrix} x_1^T(\sum_{j=1}^m \hat{u}_j \hat{v}_j^T \frac{\rho^* \pi}{\gamma m} - 2Z^*)x_1 \\ \vdots \\ x_n^T(\sum_{j=1}^m \hat{u}_j \hat{v}_j^T \frac{\rho^* \pi}{\gamma m} - 2Z^*)x_n \end{bmatrix} \right\|_\infty \tag{38}$$

where we substituted $\hat{\alpha}_j = \rho^* \frac{\pi}{\gamma m}$. The scaling factor of 2 in front of Z^* is due to the scaling factor in the SDP, i.e., $\hat{y}_i = 2x_i^T Z x_i$. Plugging in $Z^*/\rho^* = Z_s^* = \frac{\pi}{2\gamma} \mathbb{E}[u_1 v_1^T]$ yields

$$\begin{aligned} |L(\hat{Z}) - L(Z^*)| &\leq L_c \left\| \frac{\rho^* \pi}{\gamma} \begin{bmatrix} x_1^T(\frac{1}{m}\sum_{j=1}^m \hat{u}_j \hat{v}_j^T - \mathbb{E}[u_1 v_1^T])x_1 \\ \vdots \\ x_n^T(\frac{1}{m}\sum_{j=1}^m \hat{u}_j \hat{v}_j^T - \mathbb{E}[u_1 v_1^T])x_n \end{bmatrix} \right\|_\infty \\ &= L_c \frac{\rho^* \pi}{\gamma} \max_{i=1,\ldots,n} \left| x_i^T (\frac{1}{m} \sum_{j=1}^m \hat{u}_j \hat{v}_j^T - \mathbb{E}[u_1 v_1^T]) x_i \right| \\ &\leq L_c \frac{\rho^* \pi}{\gamma} \max_{i=1,\ldots,n} (\epsilon \|x_i\|_2^2) = L_c \frac{\rho^* \pi}{\gamma} \epsilon R_m^2 \end{aligned} \tag{39}$$

which holds with probability at least $1 - \exp\left(-\frac{m\epsilon^2}{(2\gamma/\pi)^2 \|Z_s^*\|_2^2 + d(c' + 2\epsilon/3)} + \log(2d)\right)$ as a result of the matrix Bernstein concentration bound. Therefore, when the number of sampled neurons satisfies the inequality

$$\frac{m\epsilon^2}{(2\gamma/\pi)^2 \|Z^*\|_2^2 + d(c' + 2\epsilon/3)} \geq 2\log(2d)\,,$$

this probability is at least $1 - \exp(-\log(2d)) = 1 - \exp(-C\epsilon^2 m/d)$, where $C > 0$ is a constant independent of d, m and ϵ.

Next, we obtain upper and lower bounds on the non-convex optimal value. Since the SDP solution provides a lower bound, and the sampled quantized network provides an upper bound, we can bound the optimal value of the original non-convex problem as follows

$$L(\hat{Z}) + \beta R(\hat{Z}) \geq L(Z_{nc}^*) + \beta R(Z_{nc}^*) \geq L(Z^*) + \beta R(Z^*)\,. \tag{40}$$

We have already established that $|L(\hat{Z}) - L(Z^*)| \leq \frac{\rho^* \pi}{\gamma} L_c R_m^2 \epsilon$ with high probability. It follows

$$\begin{aligned} L(\hat{Z}) - L(Z_{nc}^*) &= L(\hat{Z}) - L(Z^*) + L(Z^*) - L(Z_{nc}^*) \\ &\leq \frac{\rho^* \pi}{\gamma} L_c R_m^2 \epsilon + L(Z^*) - L(Z_{nc}^*) \\ &\leq \frac{\rho^* \pi}{\gamma} L_c R_m^2 \epsilon + \beta R(Z_{nc}^*) \end{aligned} \tag{41}$$

where we have used (40) and that $R(Z^*) \geq 0$ to obtain the last inequality. Furthermore, (40) implies that $L(Z_{nc}^*) - L(\hat{Z}) \leq \beta R(\hat{Z})$. If we pick the regularization coefficient β such that it satisfies $\beta \leq \frac{\frac{\rho^* \pi}{\gamma} L_c R_m^2 \epsilon}{R(Z_{nc}^*)}$ and $\beta \leq \frac{\frac{\rho^* \pi}{\gamma} L_c R_m^2 \epsilon}{R(\hat{Z})}$, we obtain the following approximation error bound

$$|L(Z_{nc}^*) - L(\hat{Z})| \leq 2 \frac{\rho^* \pi}{\gamma} L_c R_m^2 \epsilon\,. \tag{42}$$

Rescaling ϵ by $2\frac{\rho^* \pi}{\gamma} L_c R_m^2$, i.e., replacing ϵ with $\frac{1}{2\frac{\rho^* \pi}{\gamma} L_c R_m^2} \epsilon$, we obtain the claimed approximation result. □

A.3. Duality Analysis for Bilinear Activation

This subsection has the details of the duality analysis that we have carried out to obtain the SDP in (22) for the bilinear activation architecture. The derivations follow the same strategy as the duality analysis in Section 3. The non-convex problem for training such a network is stated as follows:

$$p_b^* = \min_{\text{s.t. } u_j, v_j \in \{-1,1\}^d, \alpha_j \in \mathbb{R}\, \forall j \in [m]} \ell\left(\sum_{j=1}^{m} ((Xu_j) \circ (Xv_j))\alpha_j,\, y\right) + \beta d \sum_{j=1}^{m} |\alpha_j| \,. \tag{43}$$

Taking the convex dual with respect to the second layer weights $\{\alpha_j\}_{j=1}^m$, the optimal value of the primal is lower bounded by

$$p^* \geq d^* = \max_{\max_{u,v \in \{-1,1\}^d} |\nu^T((Xu)\circ(Xv))| \leq \beta d} -\ell^*(-\nu) \tag{44}$$

where $\nu \in \mathbb{R}^n$ is the dual variable.

The constraint $\max_{u,v\in\{-1,1\}^d} |\nu^T((Xu)\circ(Xv))| \leq \beta d$ can be equivalently stated as the following two inequalities

$$\begin{aligned} q_1^* &= \max_{u_i^2 = v_i^2 = 1, \forall i} u^T \left(\sum_{i=1}^{n} \nu_i x_i x_i^T\right) v \leq \beta d \,, \\ q_2^* &= \max_{u_i^2 = v_i^2 = 1, \forall i} u^T \left(-\sum_{i=1}^{n} \nu_i x_i x_i^T\right) v \leq \beta d. \end{aligned} \tag{45}$$

We note that the second constraint $q_2^* \leq \beta d$ is redundant since the change of variable $u \leftarrow -u$ in the first constraint leads to the second constraint:

$$q_1^* = \max_{u_i^2 = v_i^2 = 1, \forall i} u^T \left(\sum_{i=1}^{n} \nu_i x_i x_i^T\right) v = \max_{(-u_i)^2 = v_i^2 = 1, \forall i} (-u)^T \left(\sum_{i=1}^{n} \nu_i x_i x_i^T\right) v = \max_{u_i^2 = v_i^2 = 1, \forall i} u^T \left(-\sum_{i=1}^{n} \nu_i x_i x_i^T\right) v = q_2^*. \tag{46}$$

In the sequel, we remove the redundant constraint $q_2^* \leq \beta d$. The SDP relaxation for the maximization $\max_{u_i^2=v_i^2=1,\forall i} u^T \left(\sum_{i=1}^n \nu_i x_i x_i^T\right) v$ is given by (see, e.g., (Alon & Naor, 2004))

$$\hat{q}_1 = \max_{K = \begin{bmatrix} V & Z \\ Z^T & W \end{bmatrix} \succeq 0,\, K_{jj} = 1, \forall j} \operatorname{tr}\left(\sum_{i=1}^{n} \nu_i x_i x_i^T Z\right) . \tag{47}$$

The dual of the above SDP relaxation can be derived via standard convex duality theory, and can be stated as

$$\min_{z_1, z_2 \text{ s.t. } \bar{1}^T z_1 + \bar{1}^T z_2 = 0} 2d\, \lambda_{\max}\left(\begin{bmatrix} \operatorname{diag}(z_1) & \sum_{i=1}^n \nu_i x_i x_i^T \\ \sum_{i=1}^n \nu_i x_i x_i^T & \operatorname{diag}(z_2) \end{bmatrix}\right) . \tag{48}$$

Then, we arrive at

$$\begin{aligned} d^* \geq d_{SDP} := \max_{\nu, z_1, z_2} \quad & -\ell^*(-\nu) \\ \text{s.t.} \quad & \begin{bmatrix} \operatorname{diag}(z_1) & \sum_{i=1}^n \nu_i x_i x_i^T \\ \sum_{i=1}^n \nu_i x_i x_i^T & \operatorname{diag}(z_2) \end{bmatrix} - \frac{\beta}{2} I \preceq 0 \\ & \bar{1}^T z_1 + \bar{1}^T z_2 = 0 \,. \end{aligned} \tag{49}$$

Next, we will find the dual of the above problem. The Lagrangian is given by

$$\begin{aligned} &L(\nu, z_1, z_2, Q, \rho) = \\ &= -\ell^*(-\nu) - \operatorname{tr}\left(Q \begin{bmatrix} \operatorname{diag}(z_1) & \sum_{i=1}^n \nu_i x_i x_i^T \\ \sum_{i=1}^n \nu_i x_i x_i^T & \operatorname{diag}(z_2) \end{bmatrix}\right) + \frac{\beta}{2}\operatorname{tr}(Q) + \rho \sum_{j=1}^d (z_{1,j} + z_{2,j}) \\ &= -\ell^*(-\nu) - \sum_{j=1}^d (V_{jj} z_{1,j} + W_{jj} z_{2,j}) - 2\sum_{i=1}^n \nu_i x_i^T Z x_i + \frac{\beta}{2}\operatorname{tr}(Q) + \rho \sum_{j=1}^d (z_{1,j} + z_{2,j}) \end{aligned} \tag{50}$$

Maximizing the Lagrangian with respect to ν, z_1, z_2 yields the problem

$$\begin{aligned} \min_{Q,\rho} \quad & \ell\left(\begin{bmatrix} 2x_1^T Z x_1 \\ \vdots \\ 2x_n^T Z x_n \end{bmatrix}, y\right) + \frac{\beta}{2}\operatorname{tr}(Q) \\ \text{s.t.} \quad & V_{jj} = \rho,\, W_{jj} = \rho,\, j = 1, \dots, d \\ & Q = \begin{bmatrix} V & Z \\ Z^T & W \end{bmatrix} \succeq 0\,. \end{aligned} \tag{51}$$

Finally, we obtain the following more concise form for the convex program

$$\begin{aligned} \min_{Q,\rho} \quad & \ell\left(\hat{y},\, y\right) + \beta d \rho \\ \text{s.t.} \quad & \hat{y}_i = 2x_i^T Z x_i,\, i = 1, \dots, n \\ & Q_{jj} = \rho,\, j = 1, \dots, 2d \\ & Q = \begin{bmatrix} V & Z \\ Z^T & W \end{bmatrix} \succeq 0\,. \end{aligned} \tag{52}$$

B. Vector Output Networks

We will assume the following vector output neural network architecture with bilinear activation

$$f(x) = \sum_{j=1}^m (x^T u_j)(x^T v_j)\alpha_j^T \tag{53}$$

where the second layer weights $\alpha_j \in \mathbb{R}^C$, $j = 1, \dots, m$ are C-dimensional vectors. We note that $f(x) : \mathbb{R}^d \to \mathbb{R}^{1\times C}$. The output of the neural network for all the samples in the dataset can be concisely represented as $\hat{Y} = f(X) \in \mathbb{R}^{n\times C}$. We use $Y \in \mathbb{R}^{n\times C}$ to denote the target matrix. The training problem can be formulated as

$$p^* = \min_{u_j, v_j \in \{-1,1\}^d, \alpha_j \in \mathbb{R}^C\; j\in[m]} \ell\left(\sum_{j=1}^m ((Xu_j) \circ (Xv_j))\alpha_j^T,\, Y\right) + \beta d \sum_{j=1}^m \|\alpha_j\|_1\,. \tag{54}$$

Or,

$$p^* = \min_{u_j, v_j \in \{-1,1\}^d,\, j\in[m]} \; \min_{\alpha_j \in \mathbb{R}^C,\, j\in[m],\, \hat{Y}} \ell\left(\hat{Y}, Y\right) + \beta d \sum_{j=1}^m \|\alpha_j\|_1 \quad \text{s.t.} \quad \hat{Y} = \sum_{j=1}^m ((Xu_j) \circ (Xv_j))\alpha_j^T\,. \tag{55}$$

The dual problem for the inner minimization problem is

$$\max_{\nu} -\ell^*(-\nu) \quad \text{s.t.} \quad |\nu_k^T((Xu_j) \circ (Xv_j))| \le \beta d,\, \forall j, k\,. \tag{56}$$

We have introduced the dual variable $\nu \in \mathbb{R}^{n\times C}$ and its columns are denoted by $\nu_k \in \mathbb{R}^n$, $k = 1, \dots, C$. The optimal value of the primal is lower bounded by

$$p^* \ge d^* = \max_{\max_{u,v\in\{-1,1\}^d} |\nu_k^T((Xu)\circ(Xv))| \le \beta d\,, \forall k} -\ell^*(-\nu)\,. \tag{57}$$

The constraints of the above optimization problem can be equivalently stated as the following inequalities

$$
\begin{aligned}
q^*_{1,k} &= \max_{u_i^2=v_i^2=1,\forall i} u^T\left(\sum_{i=1}^n \nu_{k,i}x_ix_i^T\right)v \le \beta d,\ k=1,\dots,C,\\
q^*_{2,k} &= \max_{u_i^2=v_i^2=1,\forall i} u^T\left(-\sum_{i=1}^n \nu_{k,i}x_ix_i^T\right)v \le \beta d,\ k=1,\dots,C\,.
\end{aligned}
\tag{58}
$$

As we have shown in Section A.3, the second set of inequalities $q^*_{2,k} \le \beta d$ are implied by the first and hence we remove them. The SDP relaxation for the maximization $\max_{u_i^2=v_i^2=1,\forall i} u^T\left(\sum_{i=1}^n \nu_{k,i}x_ix_i^T\right)v$ is given by

$$
\hat{q}_{1,k} = \max_{K=\begin{bmatrix} V & Z\\ Z^T & W\end{bmatrix}\succeq 0,\, K_{jj}=1,\forall j} \operatorname{tr}\left(\sum_{i=1}^n \nu_{k,i}x_ix_i^T Z\right)\,. \tag{59}
$$

We have previously given the dual of this problem as

$$
\min_{z_{k,1},z_{k,2}\ \text{s.t.}\ \bar{1}^T z_{k,1}+\bar{1}^T z_{k,2}=0} 2d\,\lambda_{\max}\left(\begin{bmatrix}\operatorname{diag}(z_{k,1}) & \sum_{i=1}^n \nu_{k,i}x_ix_i^T\\ \sum_{i=1}^n \nu_{k,i}x_ix_i^T & \operatorname{diag}(z_{k,2})\end{bmatrix}\right)\,, \tag{60}
$$

where we define the variables $z_{k,1}\in\mathbb{R}^d$, $z_{k,2}\in\mathbb{R}^d$, $k=1,\dots,C$. This allows us to establish the following lower bound

$$
\begin{aligned}
d^* \ge d_{SDP} := \max_{\nu,\{z_{k,1},z_{k,2}\}_{k=1}^C}\quad & -\ell^*(-\nu)\\
\text{s.t.}\quad & \begin{bmatrix}\operatorname{diag}(z_{k,1}) & \sum_{i=1}^n \nu_{k,i}x_ix_i^T\\ \sum_{i=1}^n \nu_{k,i}x_ix_i^T & \operatorname{diag}(z_{k,2})\end{bmatrix} - \frac{\beta}{2}I \preceq 0,\quad k=1,\dots,C\\
& \bar{1}^T z_{k,1} + \bar{1}^T z_{k,2} = 0,\quad k=1,\dots,C\,.
\end{aligned}
\tag{61}
$$

Next, we find the dual of this problem. First, we write the Lagrangian:

$$
\begin{aligned}
&L(\nu,\{z_{k,1},z_{k,2},Q_k,\rho_k\}_{k=1}^C) =\\
&= -\ell^*(-\nu) - \sum_{k=1}^C \operatorname{tr}\left(Q_k\begin{bmatrix}\operatorname{diag}(z_{k,1}) & \sum_{i=1}^n \nu_{k,i}x_ix_i^T\\ \sum_{i=1}^n \nu_{k,i}x_ix_i^T & \operatorname{diag}(z_{k,2})\end{bmatrix}\right) + \frac{\beta}{2}\sum_{k=1}^C \operatorname{tr}(Q_k) + \sum_{k=1}^C \rho_k(\bar{1}^T z_{k,1} + \bar{1}^T z_{k,2})\\
&= -\ell^*(-\nu) - \sum_{k=1}^C \left(\operatorname{diag}(V_k)^T z_{k,1} + \operatorname{diag}(W_k)^T z_{k,2}\right) - 2\sum_{k=1}^C\sum_{i=1}^n \nu_{k,i}x_i^T Z_k x_i + \frac{\beta}{2}\sum_{k=1}^C \operatorname{tr}(Q_k)\\
&\quad + \sum_{k=1}^C \rho_k(\bar{1}^T z_{k,1} + \bar{1}^T z_{k,2})\,,
\end{aligned}
\tag{62}
$$

where we have introduced $Q_k = \begin{bmatrix} V_k & Z_k\\ Z_k^T & W_k\end{bmatrix}$. Maximization of the Lagrangian with respect to $\nu, z_{k,1}, z_{k,2}, k=1,\dots,C$ leads to the dual problem given by

$$
\begin{aligned}
\min_{\{Q_k,\rho_k\}_{k=1}^C}\quad & \ell\left(\begin{bmatrix} 2x_1^TZ_1x_1 & \dots & 2x_1^TZ_Cx_1\\ \vdots & \ddots & \vdots\\ 2x_n^TZ_1x_n & \dots & 2x_n^TZ_Cx_n\end{bmatrix}, Y\right) + \frac{\beta}{2}\sum_{k=1}^C \operatorname{tr}(Q_k)\\
\text{s.t.}\quad & V_{k,jj} = \rho_k,\ W_{k,jj} = \rho_k,\quad k\in[C],\ j\in[d]\\
& Q_k = \begin{bmatrix} V_k & Z_k\\ Z_k^T & W_k\end{bmatrix} \succeq 0,\quad k\in[C].
\end{aligned}
\tag{63}
$$

More concisely,

$$\begin{aligned} \min_{\{Q_k,\rho_k\}_{k=1}^C} \quad & \ell\left(\hat{Y}, Y\right) + \beta d \sum_{k=1}^C \rho_k \\ \text{s.t.} \quad & \hat{Y}_{ik} = 2x_i^T Z_k x_i, \quad i \in [n],\ k \in [C] \\ & Q_{k,jj} = \rho_k, \quad k \in [C],\ j \in [2d] \\ & Q_k = \begin{bmatrix} V_k & Z_k \\ Z_k^T & W_k \end{bmatrix} \succeq 0, \quad k \in [C]. \end{aligned} \tag{64}$$

where V_k, Z_k, W_k are $d \times d$-dimensional matrices.

B.1. Sampling Algorithm for Vector Output Networks

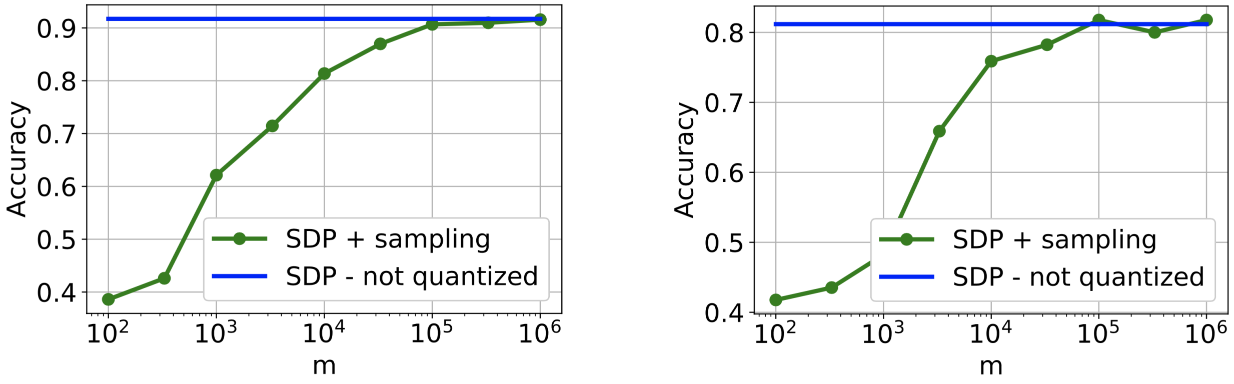
We now give the sampling algorithm:

1. Solve the SDP in (64) and define the matrices $Z_{s,k}^* \leftarrow Z_k^*/\rho_k^*$, $k = 1, \ldots, C$.
2. Find Q_k^*, $k = 1, \ldots, C$ by solving the problem

$$Q_k^* := \arg \min_{Q \succeq 0, Q_{jj}=1 \forall j} \|Q_{(12)} - \sin(\gamma Z_{s,k}^*)\|_F^2 \,. \tag{65}$$

3. Carry out the following steps for each $k = 1, \ldots, C$:
    a. Sample m/C pairs of the first layer weights u_j, v_j via $\begin{bmatrix} u_j \\ v_j \end{bmatrix} \sim \text{sign}(\mathcal{N}(0, Q_k^*))$.
    b. Set the second layer weights for these neurons to $\alpha_j = \rho_k^* C \frac{\pi}{\gamma m} e_k$ where $e_k \in \mathbb{R}^C$ is the k'th unit vector.
4. (optional) Transform the quantized bilinear activation network to a quantized polynomial activation network as described in Section 2.

Figure 4 shows the classification accuracy on a UCI machine learning repository with $C = 4$ classes. We perform one-hot encoding on the output and use the vector output SDP and sampling method developed in this section. We observe that the accuracy of the sampling method approaches the accuracy of the lower bounding SDP as m is increased.



a) Training accuracy

b) Test accuracy

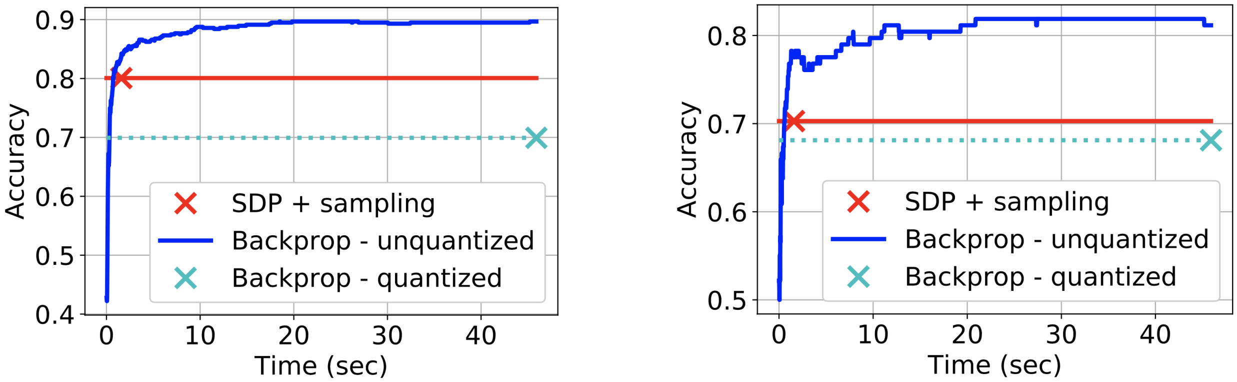
Figure 4. Vector output network experiment showing multiclass classification accuracy against the number of sampled neurons m. The dataset is statlog vehicle multiclass with $C = 4$ classes and dimensions $n = 676, d = 18$. The regularization coefficient is $\beta = 1$. The blue solid line shows the accuracy when we predict the labels using the lower bounding SDP in (64) without quantization. The green curve with circle markers shows the accuracy for the quantized network when we use the sampling method that we designed for the vector output case.

C. Further Details on Step 4 of the Sampling Algorithm

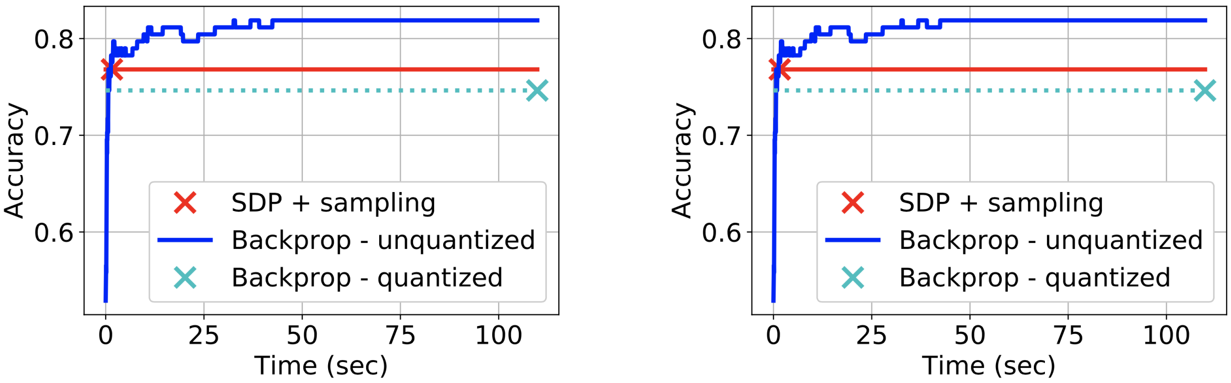
As stated in Step 4 of the sampling algorithm given in subsection 4.1, it is possible to transform the bilinear activation architecture to a quadratic activation neural network with $3m$ neurons. The first layer weights of the quadratic activation network can be obtained, via the symmetrization identity, as $1/2(u_j + v_j) \in \{-1, 0, +1\}^d$, $u_j \in \{-1, +1\}^d$, $v_j \in \{-1, +1\}^d$, $j = 1, \ldots, m$. The second layer weights are picked as stated in Step 3 for the first m neurons and the remaining $2m$ neurons have the opposite sign.

D. Additional Numerical Results

Figure 5 shows the accuracy against time for the credit approval dataset. For this dataset, we similarly observe shorter run times and better classification accuracies for the SDP based sampling method. Furthermore, increasing the number of neurons (plots c,d) improves the accuracy for both methods, which is in consistency with the experiment result shown in Figure 1.



a) Training accuracy, $m = 500$ b) Test accuracy, $m = 500$

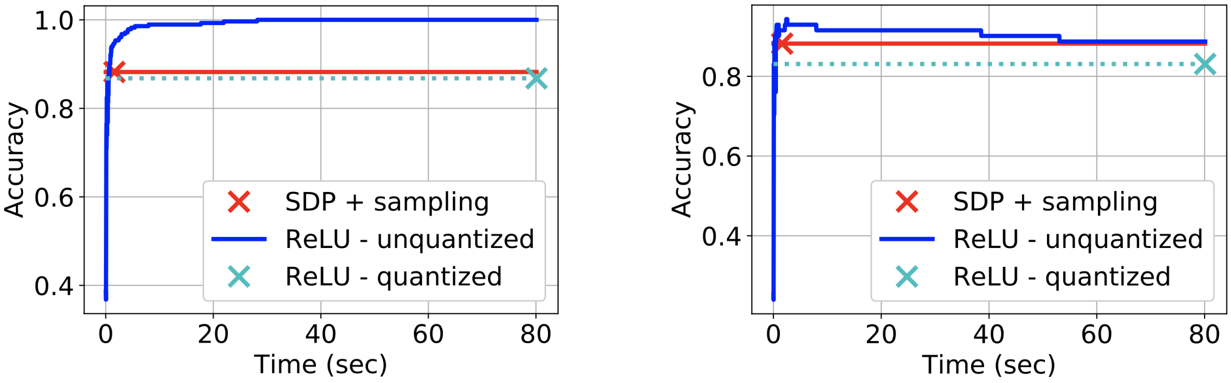


c) Training accuracy, $m = 2500$ d) Test accuracy, $m = 2500$

Figure 5. Classification accuracy against wall-clock time. Credit approval dataset with $n = 552, d = 15$. The number of neurons m is specified in the sub-caption for each plot. The regularization coefficient is $\beta = 10$ for the SDP based method and $\beta = 0.001$ for backpropagation.

D.1. ReLU network comparison

Figure 6 compares the SDP based sampling method with a two-layer ReLU network. We train the ReLU network using backpropagation and quantize the first layer weights post-training. The second layer weights are only scaled to account for the quantization of the first layer weights and not restricted to be identical. Thus, unlike the previous figures, the comparison in Figure 6 unfairly favors the ReLU network. We observe that the SDP approach can still outperform SGD in this case.



a) Training accuracy

b) Test accuracy

Figure 6. Classification accuracy against wall-clock time showing comparison with a two-layer ReLU network. Ionosphere dataset with $n = 280, d = 33$. For the SDP based sampling method, $m = 2500$ and the regularization coefficient is $\beta = 10$. For the ReLU network, $m = 5000$ and $\beta = 10^{-7}$.