Integer-arithmetic-only Certified Robustness for Quantized Neural Networks: Supplementary Material

Appendix A. Additional Details of Training

Dataset Details

CIFAR-10 [31] consists of 50,000 training images and 10,000 test images, where each image is of 32×32 resolution. For data pre-processing, we do horizontal flips and take random crops from images padded by 4 pixels on each side, filling missing pixels with reflections of original images.

Caltech-101 [11] is a more challenging dataset than CIFAR-10 since it contains 9,144 images of size 300×200 pixels in 102 categories (one of which is background). We use 101 categories for classification (without the background category). We randomly split 80% for training and the remaining images for testing. Following [27], all images are resized and center cropped into 224×224. We train on the training dataset and test on the testing for both dataset.

Training details

On CIFAR-10, we trained using SGD on one GeForce RTX 2080 GPU. We train for 90 epochs. We use a batch size of 256, and an initial learning rate of 0.1 which drops by a factor of 10 every 30 epochs. On Caltech-101 we trained with SGD on one TITAN RTX GPU. We train for 90 epochs. We use a batch size of 64, and an initial learning rate of 0.1 which drops by a factor of 10 every 30 epochs. The models used in this paper are similar to those used in Cohen et al. [6] except we use a smaller model on CIFAR10. On CIFAR-10, we used a 20-layer residual network from https://github.com/bearpaw/pytorch-classification. On Caltech-101 our base classifier used the pretrained ResNet-50 architecture provided in torchvision.

Appendix B. Additional Details and Results of Figure 1: The Demonstration Example

For Figure 1 in the paper, we use a well-studied adversarial perturbation attack method: projected gradient descent (PGD) to find adversarial examples against the base classifier f and assess the performance of the attack on full-precision model and quantized model. We set iterations equal to 7 and vary ε which is the maximum allowed l_∞ perturbation of the input from 0.001 to 0.05. Here we present the adversarial examples we found under different ε in Figure 9. For $\varepsilon = 0.05$, the adversarial examples generated by PGD attack is visually indistinguishable from the original image, but completely distorts both the full-precision and quantized classifiers' prediction.

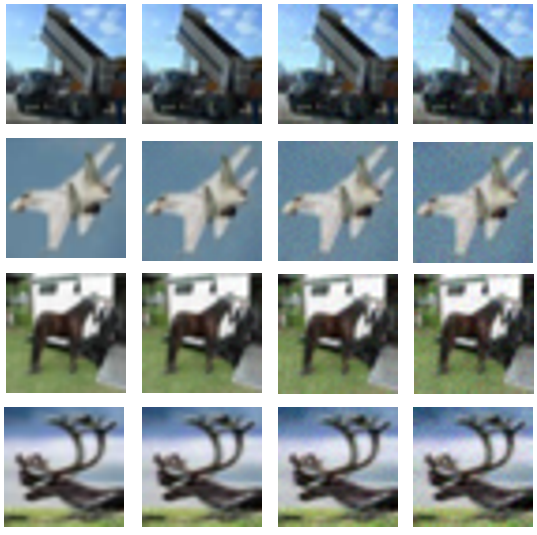

Figure 9. CIFAR-10 adversarial images corrupted generated by PGD attack with varying levels of perturbations

Appendix C. Practical Prediction and Experiment Results

Here we describe how to get the smoothed classifier's prediction. We use the same prediction algorithm as in [6]. **Prediction** draws n samples of $f(\mathbf{x}+\mathbf{n})$ and return the class as its predicted label which appeared much more often than any other class. If such class doesn't exist, **Prediction** will abstain. The pseudocode is in Algorithm 2.

We also analyze the effect of the number of Monte-Carlo samples n in **Prediction** on quantized model. Table 1 shows the performance of **Prediction** as the number of Monte Carlo samples n is varied between 100 and 10000 on CIFAR-10. When N increases, the time spent on **Prediction** also increases. We observe from Table 1 that when n is small, the smooth classifier is more likely to make abstentions for both full-precision (FloatRS-fp) and quantized (IntRS-quant) model.

Algorithm 2 Monte-Carlo estimation and aggregated evaluation for certified robust prediction

Input: Base function $f(\cdot)$, inference sample $\mathbf{x}$, Gaussian noise std σ, repeated number N, and confidence level α.

Prediction:

1: Repeat N inferences on $f(\mathbf{x}+\mathbf{n})$, where $\mathbf{n} \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I}_d)$.
2: Collect prediction results: $(n_A, \hat{c}_A)$: highest prediction count and its label; $(n_B, \hat{c}_B)$: second highest prediction count and its label;
3: **if** Binomial p-value test of given $n_A, n_A + n_B$ is no greater than 0.5 **then**
4: **Return** $\hat{c}_A$;
5: **else**
6: ABSTAIN
7: **end if**

Table 1. Performance of Prediction when n is varied. The column presents the result on CIFAR-10 and set $\sigma = 0.25, \alpha = 0.001$ The column is "correct" if Prediction returns the label without abstention and the labels matches with the ground-truth label

FloatRS-fp			IntRS-quant		
n	correct	abstain	n	correct	abstain
100	0.74	0.16	100	0.73	0.15
1000	0.79	0.03	1000	0.77	0.05
10000	0.81	0.02	10000	0.80	0.02
100000	0.82	0.00	100000	0.80	0.00

Appendix D. Additional Experiments with Different Types of Adversarial Perturbation Attacks

In this appendix, we use one of the strongest attacks (i.e., projected gradient descent (PGD)) under ℓ_2 abll to generate adversarial perturbations and evaluate **Prediction** performance. For **Prediction**, we set $n = 1000, \alpha = 0.001, \sigma = 0.25$. For PGD, we set 20 iterations and vary $\varepsilon = \{0.0, 0.12, 0.25, 0.50, 1.00\}$. Here ε is the maximum allowed ℓ_2 perturbation of the input. Figure 10 demonstrates the results of prediction accuracy on adversarial examples of CIFAR-10 on full-precision model and our quantized model.

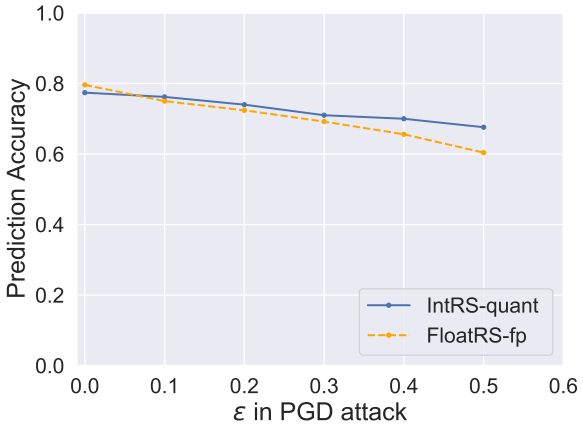

Figure 10. Prediction accuracy on CIFAR-10 adversarial examples of FloatRS-fp and IntRS-quantized model.

Appendix E. Effect of the Confidence Level Parameter α

In this section, we show the effect of confidence level parameter α on certified accuracy on the full-precision model and our quantized model. We can observe that the certified accuracy of each model has not been vastly affected by choice of α.

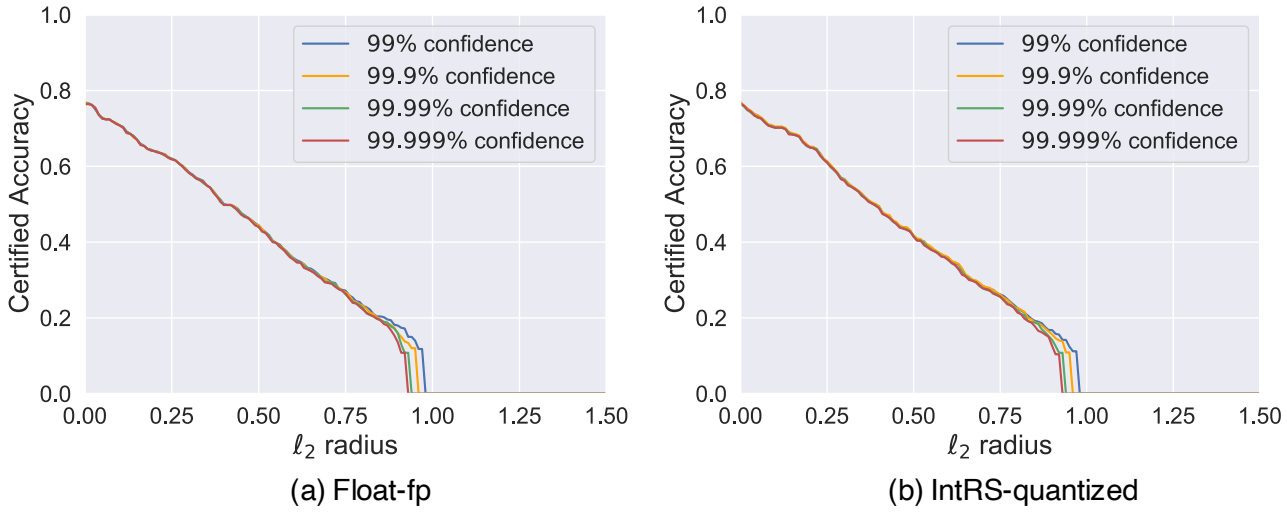

Figure 11. Certified accuracy of varying $bm\alpha$. The experiment is performed on CIFAR-10 with $\sigma = 0.25$

Appendix F. Detailed Results on Dataset: Report Table

In Table 2, 3, we summarize the certified accuracy under different noise level σ at different radius r. In Table 4, 5, we vary certification noise while holding training noise fixed at $\sigma = 0.12, 0.25$ on CIFAR-10 to evaluate the effects of Gaussian noise for training base classifier f on certification performance. Note for the quantized model, the accuracy of base model f would be slightly lower than that of the full-precision model. Our goal is to achieve comparably certified accuracy for IntRS-quant compared with FloatRS-fp model.

Table 2. Certified test accuracy on CIFAR-10 with different σ. Each column represents the certified accuracy at different radius r

FloatRS-fp	$r = 0.25$	$r = 0.5$	$r = 0.75$	$r = 1.0$	$r = 1.25$	$r = 1.5$
$\sigma = 0.12$	0.59	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.25$	0.62	0.44	0.27	0.00	0.00	0.00
$\sigma = 0.50$	0.54	0.43	0.32	0.22	0.15	0.09
$\sigma = 1.00$	0.39	0.33	0.28	0.22	0.18	0.15
IntRS-quant	$r = 0.25$	$r = 0.5$	$r = 0.75$	$r = 1.0$	$r = 1.25$	$r = 1.5$
$\sigma = 0.12$	0.59	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.25$	0.61	0.42	0.26	0.00	0.00	0.00
$\sigma = 0.50$	0.52	0.39	0.29	0.22	0.15	0.08
$\sigma = 1.00$	0.35	0.28	0.23	0.18	0.16	0.12
FloatRS-quant	$r = 0.25$	$r = 0.5$	$r = 0.75$	$r = 1.0$	$r = 1.25$	$r = 1.5$
$\sigma = 0.12$	0.56	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.25$	0.59	0.42	0.23	0.00	0.00	0.00
$\sigma = 0.50$	0.43	0.33	0.25	0.18	0.11	0.06
$\sigma = 1.00$	0.19	0.14	0.12	0.09	0.07	0.05

Table 3. Certified test accuracy on Caltech-101 with different σ

FloatRS-fp	$r = 0.25$	$r = 0.5$	$r = 0.75$	$r = 1.0$	$r = 1.25$	$r = 1.5$	$r = 1.75$	$r = 2.0$
$\sigma = 0.12$	0.65	0.00	0.00	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.25$	0.56	0.54	0.00	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.50$	0.62	0.58	0.55	0.52	0.00	0.00	0.00	0.00
$\sigma = 1.00$	0.51	0.51	0.48	0.47	0.46	0.45	0.45	0.41
IntRS-quant	$r = 0.25$	$r = 0.5$	$r = 0.75$	$r = 1.0$	$r = 1.25$	$r = 1.5$	$r = 1.75$	$r = 2.0$
$\sigma = 0.12$	0.61	0.00	0.00	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.25$	0.58	0.56	0.00	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.50$	0.64	0.59	0.51	0.46	0.00	0.00	0.00	0.00
$\sigma = 1.00$	0.56	0.56	0.56	0.54	0.53	0.52	0.52	0.52
FloatRS-quant	$r = 0.25$	$r = 0.5$	$r = 0.75$	$r = 1.0$	$r = 1.25$	$r = 1.5$	$r = 1.75$	$r = 2.0$
$\sigma = 0.12$	0.59	0.00	0.00	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.25$	0.60	0.56	0.00	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.50$	0.61	0.56	0.00	0.00	0.00	0.00	0.00	0.00
$\sigma = 1.00$	0.20	0.18	0.18	0.18	0.16	0.14	0.12	0.12

Table 4. Certified Accuracy of varying σ used in certification. The base model f is trained on CIFAR-10 using Gaussian noise augmentation with $\sigma = 0.12$

FloatRS-fp	$r = 0.25$	$r = 0.5$	$r = 0.75$	$r = 1.0$	$r = 1.25$	$r = 1.5$	$r = 1.75$
$\sigma = 0.12$	0.59	0.00	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.25$	0.19	0.11	0.07	0.00	0.00	0.00	0.00
$\sigma = 0.50$	0.09	0.09	0.08	0.07	0.04	0.01	0.00
$\sigma = 1.00$	0.10	0.09	0.09	0.08	0.06	0.04	0.03
IntRS-quant	$r = 0.25$	$r = 0.5$	$r = 0.75$	$r = 1.0$	$r = 1.25$	$r = 1.5$	$r = 1.75$
$\sigma = 0.12$	0.59	0.00	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.25$	0.36	0.24	0.11	0.00	0.00	0.00	0.00
$\sigma = 0.50$	0.15	0.12	0.11	0.08	0.05	0.01	0.00
$\sigma = 1.00$	0.11	0.10	0.10	0.09	0.09	0.09	0.07

Table 5. Certified Accuracy of varying σ used in certification. The base model f is trained trained on CIFAR-10 using Gaussian noise augmentation with $\sigma = 0.25$

FloatRS-fp	$r = 0.25$	$r = 0.5$	$r = 0.75$	$r = 1.0$	$r = 1.25$	$r = 1.5$	$r = 1.75$
$\sigma = 0.12$	0.57	0.00	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.25$	0.62	0.44	0.27	0.00	0.00	0.00	0.00
$\sigma = 0.50$	0.19	0.15	0.10	0.05	0.02	0.01	0.00
$\sigma = 1.00$	0.10	0.09	0.08	0.08	0.06	0.04	0.02
IntRS-quant	$r = 0.25$	$r = 0.5$	$r = 0.75$	$r = 1.0$	$r = 1.25$	$r = 1.5$	$r = 1.75$
$\sigma = 0.12$	0.57	0.00	0.00	0.00	0.00	0.00	0.00
$\sigma = 0.25$	0.61	0.42	0.26	0.00	0.00	0.00	0.00
$\sigma = 0.50$	0.31	0.23	0.15	0.08	0.02	0.01	0.00
$\sigma = 1.00$	0.14	0.11	0.10	0.07	0.03	0.02	0.01

Appendix G. Examples of Noisy Images

In this section, we demonstrate examples of CIFAR-10 and Caltech-101 images corrupted with varying levels of noise in Gaussian noise. Since it is hard to visualize the quantized input, we only present the input corrupted by $\mathcal{N}(0, \sigma^2)$.

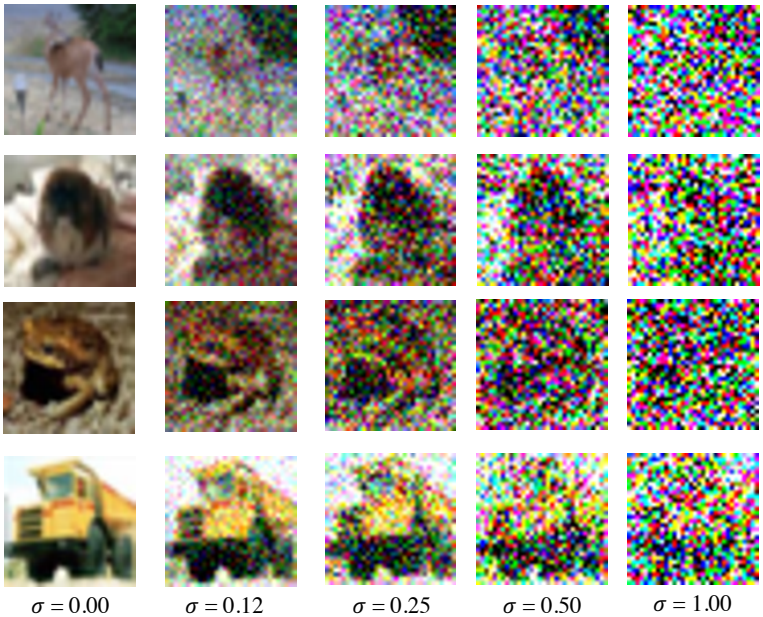

Figure 12. An illustration of CIFAR-10 images generated by adding Gaussian noise with various σPixel values greater than 1.0 or less than 0.0 were clipped to 1.0 or 0.0.

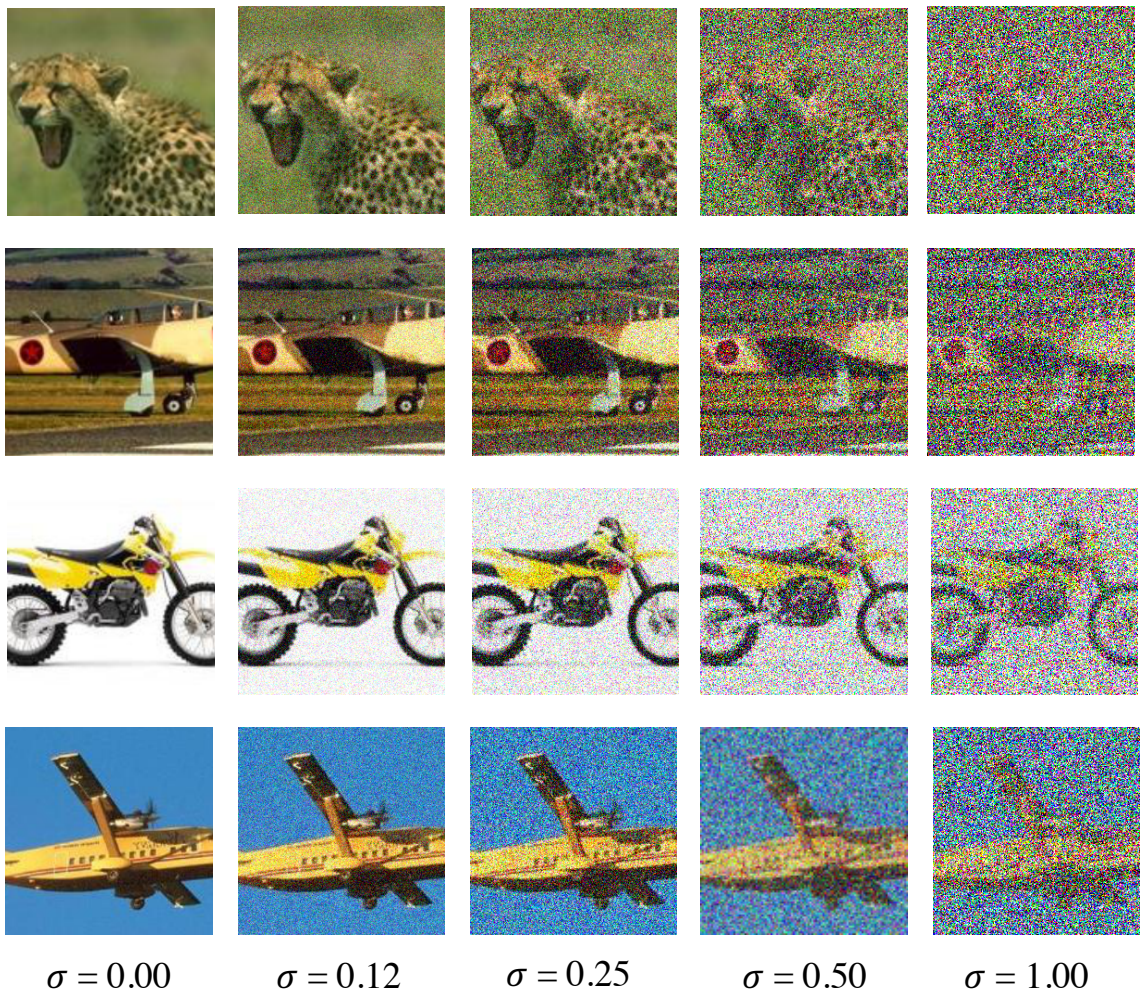

Figure 13. An illustration of Caltech-101 images generated by adding Gaussian noise with various σ. Pixel values greater than 1.0 or less than 0.0 were clipped to 1.0 or 0.0.

Appendix H. Omitted Proof

H.1. Proof of Proposition 3.1

Proof. The proof follows the Neyman-Pearson lemma (especially its form for discrete distribution in Lemma 3.1). We want to show that the condition in this Proposition is equivalent to the condition with respect to the likelihood ratio statistic, which takes the form:

$$\mathscr{L}(\mathbf{z}) = \frac{\left(\prod_{i=1}^{d} \frac{e^{-(\mathbf{z}_i-(\mathbf{x}_i+\boldsymbol{\delta}_i))^2/2\sigma^2}}{\sum_{\mathbf{h}_i\in\mathbb{H}} e^{-(\mathbf{h}_i-(\mathbf{x}_i+\boldsymbol{\delta}_i))^2/2\sigma^2}}\right)}{\left(\prod_{i=1}^{d} \frac{e^{-(\mathbf{z}_i-\mathbf{x}_i)^2/2\sigma^2}}{\sum_{\mathbf{h}_i\in\mathbb{H}} e^{-(\mathbf{h}_i-\mathbf{x}_i)^2/2\sigma^2}}\right)}. \tag{11}$$

Due to the discrete nature of the inference stage, we have $\boldsymbol{\delta}_i \in \mathbb{H}$. Based on this fact and by $\sum_{\mathbf{h}_i\in\mathbb{H}} e^{-(\mathbf{h}_i-(\mathbf{x}_i+\boldsymbol{\delta}_i))^2/2\sigma^2}$ is periodic for $\boldsymbol{\delta}_i \in \mathbb{H}$, we have

$$\sum_{\mathbf{h}_i\in\mathbb{H}} e^{-(\mathbf{h}_i-(\mathbf{x}_i+\boldsymbol{\delta}_i))^2/2\sigma^2} = \sum_{\mathbf{h}_i\in\mathbb{H}} e^{-(\mathbf{h}_i-\mathbf{x}_i)^2/2\sigma^2}. \tag{12}$$

Then, we further have

$$\begin{aligned}\mathscr{L}(\mathbf{z}) &= \frac{e^{-\sum_{i=1}^{d}(\mathbf{z}_i-(\mathbf{x}_i+\boldsymbol{\delta}_i))^2/2\sigma^2}}{e^{-\sum_{i=1}^{d}(\mathbf{z}_i-\mathbf{x}_i)^2/2\sigma^2}} \\ &= e^{\frac{1}{2\sigma^2}\sum_{i=1}^{d}(2\mathbf{z}_i\boldsymbol{\delta}_i-\delta_i-2\mathbf{x}_i\boldsymbol{\delta}_i)} \\ &= e^{\frac{1}{\sigma^2}\langle\mathbf{z},\boldsymbol{\delta}\rangle-\frac{1}{2\sigma^2}(\|\boldsymbol{\delta}\|_2^2+2\langle\mathbf{x},\boldsymbol{\delta}\rangle)}.\end{aligned} \tag{13}$$

Thus, in order to carry out the likelihood ratio test, we have the following equivalent relationship.

$$\mathscr{L}(\mathbf{z}) \le \alpha \Longleftrightarrow \langle\mathbf{z},\boldsymbol{\delta}\rangle \le \sigma^2\ln\alpha + \frac{1}{2}(\|\boldsymbol{\delta}\|_2^2 + 2\langle\mathbf{x},\boldsymbol{\delta}\rangle) \tag{14}$$

$$\mathscr{L}(\mathbf{z}) \ge \alpha \Longleftrightarrow \langle\mathbf{z},\boldsymbol{\delta}\rangle \ge \sigma^2\ln\alpha + \frac{1}{2}(\|\boldsymbol{\delta}\|_2^2 + 2\langle\mathbf{x},\boldsymbol{\delta}\rangle). \tag{15}$$

The remaining follows Lemma 3.1. □