Common Q1: Theoretical justification on why AWP works.

A1: Based on previous work on PAC-Bayes bound (Neyshabur et al., NeurIPS 2017), in adversarial training, let $\ell(\cdot,\cdot)$ be 0-1 loss, then $\rho(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n}\max_{\|\mathbf{x}_i'-\mathbf{x}_i\|_p\leq\epsilon}\ell(f_{\mathbf{w}}(\mathbf{x}_i'),y_i) \in [0,1]$. Given a "prior" distribution P (a common assumption is zero mean, σ^2 variance Gaussian distribution) over the weights, the expected error of the classifier can be bounded with probability at least $1-\delta$ over the draw of n training data: $\mathbb{E}_{\{\mathbf{x}_i,y_i\}_{i=1}^n,\mathbf{u}}[\rho(\mathbf{w}+\mathbf{u})] \leq \rho(\mathbf{w}) + \big\{\mathbb{E}_{\mathbf{u}}[\rho(\mathbf{w}+\mathbf{u})] - \rho(\mathbf{w})\big\} + 4\sqrt{\frac{1}{n}KL(\mathbf{w}+\mathbf{u}\|P) + \ln\frac{2n}{\delta}}$. Following (Neyshabur et al., NeurIPS 2017), we choose $\mathbf{u}$ as a zero mean spherical Gaussian perturbation with variance σ^2 in every direction, and set the variance of the perturbation to the weight with respect its magnitude $\sigma = \alpha\|\mathbf{w}\|$, which makes the third term become a constant $4\sqrt{\frac{1}{n}(\frac{1}{2\alpha}+\ln\frac{2n}{\delta})}$. Thus, the robust generalization gap is bounded by the second term that is the expectation of the flatness of weight loss landscape. Considering the optimization efficiency and effectiveness on expectation, $\mathbb{E}_{\mathbf{u}}[\rho(\mathbf{w}+\mathbf{u})] \leq \max_{\mathbf{u}}[\rho(\mathbf{w}+\mathbf{u})]$. AWP exactly optimizes the worst-case of the flatness of weight loss landscape $\{\max_{\mathbf{u}}[\rho(\mathbf{w}+\mathbf{u})] - \rho(\mathbf{w})\}$ to control the above PAC-Bayes bound, which theoretically justifies why AWP works.

Common Q2: Overhead of AWP.

A2: In the default setting of AWP ($A=1, K_1=10, K_2=1$), we first generate adversarial examples (10 back-propagations), and then perturb the weights (AWP) based on these generated adversarial examples (**no need to re-generate adversarial examples**, only 1 back-propagation), finally update the weights (1 back-propagation). Therefore, compared to the vanilla AT, AWP only introduces one extra back-propagation whose overhead is $\sim 8\%$.

R#1 Q1: The weights are constantly perturbed in the worst case, the model may find it difficult to learn.

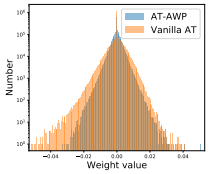
A1: The weight perturbation is constrained in a small region by γ. With properly chosen γ, it encourages the model to explore the blind spots in the local subspace, and its convergence is demonstrated by the learning curve in Fig. 3(d).

R#1 Q2: How do the baseline methods that do implicit weight perturbations differ from AWP?

A2: We did not claim that "baseline methods do the implicit weight perturbations". Instead, we only stated that (Line 150-152) they **implicitly flatten** the weight loss landscape via different techniques (additional data, regularization, and so on), while AWP explicitly performs weight perturbations to flatten the weight loss landscape.

R#1 Q3: What is the difference of weights learned by AT-AWP and vanilla AT?

A3: We plot the histogram of weight values in different layers, and find that AT-AWP and vanilla AT are similar in shallower layers, while AT-AWP has smaller magnitudes and a more symmetric distribution in deeper layers. The right figure demonstrates the distribution of weight values in the last convolutional layer of PreAct ResNet-18 on CIFAR-10 dataset.



R#2 Q1: Only CIFAR-10 and single neural networks are tested.

A1: We have tested several network architectures and datasets in the main body and appendix, *e.g.*, PreAct ResNet-18, VGG-19, and WideResNet-34 on CIFAR-10, SVHN, and CIFAR-100 in Figures 1 & 7 & 8 and Tables 1 & 2.

R#2 Q2: In Fig. 1, the α value in the loss landscape is embed into training or post-training?

A2: It is post-training. Given a model $f_{\mathbf{w}}$, we first perturb its weights by α, and then calculate the adversarial loss on adversarial examples that are specifically generated on the perturbed model $f_{\mathbf{w}+\alpha\mathbf{d}}$ rather than the original model $f_{\mathbf{w}}$ (Line 118-125). Therefore, there is no gradient obfuscation and underestimated post-attack loss here.

R#2 Q3: The trade-off (natural vs. attack accuracy) and experimental configuration for Table 2.

A3: In Table 2, AWP is the default setting $A=1, K_1=10, K_2=1$. For AT and RST, they demonstrate the trade-off between the natural accuracy and the attack accuracy, while TRADES, MART, and Pre-training do not. This is caused by the perturbation constraint γ, we can use smaller γ (*e.g.*, 2.5×10^{-3}) to improve AT and RST on both metrics.

R#3 Q1: What would be the effect of combining AWP with certified defense? The relation to AAAI 2020 paper?

A1: Taking randomized smoothing as an example, the certified radius of provably robust classifier is related to the resistance of original classifier against large Gaussian input perturbations. AWP can improve the generalization of original classifier against perturbations, which will benefit the certified robustness. AWP and AAAI paper both use adversarial weight perturbations during training, but AAAI paper improves the inference performance of weight-perturbed model (caused by quantization), while AWP improves inference performance under no weight perturbations.

R#4 Q1: Why not use $|\rho(\mathbf{w}+\mathbf{v})-\rho(\mathbf{w})|$ or $(\rho(\mathbf{w}+\mathbf{v})-\rho(\mathbf{w}))^2$ as the criteria of the flatness?

A1: We agree that $|\rho(\mathbf{w}+\mathbf{v})-\rho(\mathbf{w})|$ is a better criterion. Actually, we first do AWP before minimizing $\mathbf{w}$, while AWP always increases the loss and makes the term $\rho(\mathbf{w}+\mathbf{v})-\rho(\mathbf{w})$ non-negative, thus $|\cdot|$ can be removed.

R#4 Q2: The difference to previous studies [44, 63] is not clear.

A2: As stated in Line 42-47 and Line 121-125, [44, 63] both tried but failed to draw the connection of weight loss landscape and robust generalization gap due to the unreasonably generated adversarial examples. Further, [44] reminded us to rethink the robust generalization, and [63] turned to explore the input loss landscape.

R#4 Q3: In Eq. (11), why not use the gradient with respect to $\mathbf{w}$, without addition and subtraction of $\mathbf{v}$?

A3: Thanks for your suggestion. They are equivalent. We will modify it in the revision.