A Relationship between Fisher Information and Hessian

In probabilistic view for deep neural networks (DNNs), Fisher information matrix F is a way of measuring the amount of information about a negative log-likelihood for a joint distribution $p(x, y \mid w)$ parameterized by model parameters w in DNNs as follows: $-\log p(x, y \mid w)$. The Fisher information matrix can be carried from random variables x and y (*i.e.,* input images and their target labels) in this joint distribution, which can be written as:

$$F = \mathbb{E}_{(x,y)\sim p(x,y|w)} \left[\{\nabla_w \log p(x, y \mid w)\}\{\nabla_w \log p(x, y \mid w)\}^T \right]. \tag{1}$$

Here, the joint distribution can be factorized with the conditional distribution $p(y \mid x, w)$ and the prior $p(x)$, where the conditional distribution indicates model prediction in DNNs. In addition, the gradient of the joint distribution equals to that of the conditional distribution as follows: $-\nabla_w \log p(x, y \mid w) = -\nabla_w \log p(y \mid x, w)$. This is because the prior distribution $p(x)$ does not have factors of model parameters w. Then, the following equation is satisfied for the Fisher information matrix with the factorized distribution as:

$$F = \mathbb{E}_{x\sim p(x)} \left[\mathbb{E}_{y\sim p(y|x,w)} \left[\{\nabla_w \log p(y \mid x, w)\}\{\nabla_w \log p(y \mid x, w)\}^T \right] \right]. \tag{2}$$

However, in terms of the prior distribution $p(x)$, we cannot deal with an ideal prior due to limitation of sampling x, thus we replace it with an empirical prior distribution $q(x)$ possibly regarding our practical dataset $\mathcal{D}$. Therefore, the Fisher information matrix can be formulated as follows:

$$F = \mathbb{E}_{x\sim q(x)} \left[\mathbb{E}_{y\sim p(y|x,w)} \left[\{\nabla_w \log p(y \mid x, w)\}\{\nabla_w \log p(y \mid x, w)\}^T \right] \right]. \tag{3}$$

Meanwhile, Hessian for model prediction in DNNs is known to compute with the second derivative of the negative log-likelihood $-\log p(y \mid x, w)$, which can be written as:

$$\mathrm{H} = \mathbb{E}_{x\sim q(x)} \left[\mathbb{E}_{y\sim p(y|x,w)} \left[-\nabla_w^2 \log p(y \mid x, w) \right] \right]. \tag{4}$$

Next, to examine a relationship between Fisher information and Hessian, we expand the second derivative of it as follows:

$$\begin{aligned}
-\nabla_w^2 \log p(y \mid x, w) &= -\nabla_w \left[\frac{\nabla_w p(y \mid x, w)}{p(y \mid x, w)} \right] \\
&= -\frac{p(y \mid x, w)\nabla_w^2 p(y \mid x, w) - \{\nabla_w p(y \mid x, w)\}\{\nabla_w p(y \mid x, w)\}^T}{p(y \mid x, w)^2} \\
&= -\frac{\nabla_w^2 p(y \mid x, w)}{p(y \mid x, w)} + \left\{ \frac{\nabla_w p(y \mid x, w)}{p(y \mid x, w)} \right\} \left\{ \frac{\nabla_w p(y \mid x, w)}{p(y \mid x, w)} \right\}^T \\
&= -\frac{\nabla_w^2 p(y \mid x, w)}{p(y \mid x, w)} + \{\nabla_w \log p(y \mid x, w)\}\{\nabla_w \log p(y \mid x, w)\}^T.
\end{aligned} \tag{5}$$

Then, we apply this expanded second derivative to Eq. (4) which can be formulated as follows:

$$\begin{aligned}
\mathrm{H} &= \mathbb{E}_{x\sim q(x)}\left[\mathbb{E}_{y\sim p(y|x,w)}\left[-\frac{\nabla_w^2 p(y \mid x,w)}{p(y \mid x,w)} + \{\nabla_w \log p(y \mid x,w)\}\{\nabla_w \log p(y \mid x,w)\}^T\right]\right] \\
&= \mathbb{E}_{x\sim q(x)}\left[\mathbb{E}_{y\sim p(y|x,w)}\left[-\frac{\nabla_w^2 p(y \mid x,w)}{p(y \mid x,w)}\right]\right] + F \\
&= \mathbb{E}_{x\sim q(x)}\left[\int_{y\in\mathcal{C}} -\frac{\nabla_w^2 p(y \mid x,w)}{p(y \mid x,w)} p(y \mid x,w) dy\right] + F \quad (\mathcal{C}: \text{a class set}) \\
&= \mathbb{E}_{x\sim q(x)}\left[-\nabla_w^2 \underbrace{\int_{y\in\mathcal{C}} p(y \mid x,w) dy}_{=1}\right] + F \\
&= F.
\end{aligned} \tag{6}$$

Therefore, it is proved that Hessian for the model prediction equals to Fisher Information matrix for the negative log-likelihood. At this point, we focus on how to approximate the second derivative of a loss function $\mathcal{L}$ (cross-entropy) such that $\nabla_w^2\mathcal{L}$ by using an empirical distribution $q(x)$ with respect to $\mathcal{D}$. Once we employ Eq. (6), the second derivative of the loss function can be computed with Hessian and Fisher information as follows:

$$\begin{aligned}
\nabla_w^2\mathcal{L} &\simeq \underbrace{\mathbb{E}_{x\sim q(x)}\left[\mathbb{E}_{y\sim p(y|x,w)}\left[\nabla_w^2\mathcal{L}\right]\right]}_{\mathrm{H}} \\
&= \underbrace{\mathbb{E}_{x\sim q(x)}\left[\mathbb{E}_{y\sim p(y|x,w)}\left[\{\nabla_w\mathcal{L}\}\{\nabla_w\mathcal{L}\}^T\right]\right]}_{F}.
\end{aligned} \tag{7}$$

In conclusion, it satisfies $\nabla_w^2\mathcal{L} \simeq \mathbb{E}_{\mathcal{D}}[\nabla_w\mathcal{L}^2]$ where $w \in \mathbb{R}$.

B A Deterministic Solution of Optimal Brain Surgeon (OBS)

To obtain a deterministic solution for the loss change, OBS deals with the following optimization problem as:

$$\min_{\Delta w \in \mathbb{R}^d} \frac{1}{2} \Delta w^T \mathrm{H} \Delta w, \quad \text{s.t. } e_k^T \Delta w + w_k = 0. \tag{8}$$

Then, we employ Lagrangian relaxation to solve it, which can be formulated as:

$$\mathcal{R}(\Delta w, \lambda) = \frac{1}{2} \Delta w^T \mathrm{H} \Delta w + \lambda (e_k^T \Delta w + w_k), \tag{9}$$

where $\mathcal{R}$ denotes the objective with Lagrangian relaxation, and λ indicates a Lagrangian parameter. Here, we expand the derivative of the objective over the variation Δw as follows:

$$\begin{aligned} \frac{\partial}{\partial \Delta w} \mathcal{R}(\Delta w, \lambda) &= \mathrm{H} \Delta w + \lambda e_k = 0 \\ \Rightarrow \qquad \Delta w^* &= -\lambda \mathrm{H}^{-1} e_k. \end{aligned} \tag{10}$$

We substitute this optimal variation Δw^* for the objective in Eq. (9), which can be written as:

$$\mathcal{R}(\Delta w^*, \lambda) = -\frac{\lambda^2}{2} e_k^T \mathrm{H}^{-1} e_k + \lambda w_k. \tag{11}$$

Next, we differentiate the objective over the Lagrangian parameter λ as follows:

$$\begin{aligned} \frac{\partial}{\partial \lambda} \mathcal{R}(\Delta w^*, \lambda) &= -\lambda e_k^T \mathrm{H}^{-1} e_k + w_k = 0 \\ \Rightarrow \qquad \lambda^* &= \frac{w_k}{e_k^T \mathrm{H}^{-1} e_k} = \frac{w_k}{[\mathrm{H}^{-1}]_{kk}}. \end{aligned} \tag{12}$$

Now, we simply apply this optimal Lagrangian parameter λ^* to the objective in Eq. (11). It can be formulated as follows:

$$\mathcal{R}(\Delta w^*, \lambda^*) = \frac{1}{2} \frac{w_k^2}{[\mathrm{H}^{-1}]_{kk}}. \tag{13}$$

Therefore, we can compute a deterministic solution for OBS by these procedures through the Lagrangian relaxation, such that it satisfies $\Delta \mathcal{L}_{\text{OBS}} = \frac{1}{2} w_k^2 / [\mathrm{H}^{-1}]_{kk}$. In case of Optimal Brain Damage (OBD), its deterministic solution can be also computed through this loss change in Eq. (13), where we consider only diagonal terms in Hessian. Then, it satisfies $\Delta \mathcal{L}_{\text{OBD}} = \frac{1}{2} w_k^2 \mathrm{H}_{kk}$.

C Theoretical Analysis of Taylor Expansion

Taylor approximation represents a number as a polynomial that has a very similar value to the number in a neighborhood around a specified value. It is a powerful tool to approximate a function that can be intractable. It is evaluated as infinite sums and integrals of the function's derivatives at a single point. Basically, it can be written with an arbitrary function $F : \mathbb{R} \to \mathbb{R}$ as follows:

$$F(a + \Delta a) = F(a) + \sum_{k=1}^{\infty} \frac{(\Delta a)^k}{k!} \frac{\partial^k}{\partial a^k} F(a). \tag{14}$$

Once the second derivative of the function is considered, it can be expressed as a second-order polynomial as follows:

$$F(a + \Delta a) = F(a) + \Delta a \frac{\partial}{\partial a} F(a) + \frac{\Delta a^2}{2} \frac{\partial^2}{\partial a^2} F(a), \tag{15}$$

where the difference $\Delta a \in \mathbb{R}$ gets small enough. Here, we extend it to deal with a multi-variable function $\mathcal{F} : \mathbb{R}^M \to \mathbb{R}$ given small $\Delta a \in \mathbb{R}^M$ as follows:

$$\mathcal{F}(a + \Delta a) = \mathcal{F}(a) + \left[\frac{\partial}{\partial a} \mathcal{F}(a) \right]^T \Delta a + \frac{1}{2} \Delta a^T \left\{ \frac{\partial^2}{\partial a^2} \mathcal{F}(a) \right\} \Delta a, \tag{16}$$

where the dimension of $\mathcal{F}(a)$, $\frac{\partial}{\partial a}\mathcal{F}(a)$, and $\frac{\partial^2}{\partial a^2} F(a)$ is each $\mathbb{R}$, $\mathbb{R}^M$, and $\mathbb{R}^{M \times M}$. From this Taylor expansion, once we change variables of $\mathcal{F} \to \mathcal{L}$ and $a \to w$, then Eq. (16) can be written as follows:

$$\begin{aligned} \mathcal{L}(w + \Delta w) &= \mathcal{L}(w) + \underbrace{\left[\frac{\partial}{\partial w} \mathcal{L}(w) \right]^T}_{\approx 0} \Delta w + \frac{1}{2} \Delta w^T \left\{ \frac{\partial^2}{\partial w^2} \mathcal{L}(w) \right\} \Delta w \\ \Rightarrow \quad \Delta \mathcal{L} &= \frac{1}{2} \Delta w^T \left\{ \frac{\partial^2}{\partial w^2} \mathcal{L}(w) \right\} \Delta w. \quad (\Delta \mathcal{L} := \mathcal{L}(w + \Delta w) - \mathcal{L}(w)) \end{aligned} \tag{17}$$

Here, the first gradient of the loss function $\mathcal{L}$ approximates zero due to using well-trained DNNs in standard pruning. In addition, it is the basic equation of OBD and OBS.

For *Masking Adversarial Damage* (MAD), once we change variables of $\mathcal{F} \to \mathcal{L}$, $(a + \Delta a) \to w$, and $a \to w_{m^*}$, then Eq. (16) can be represented as follows:

$$\begin{aligned} \mathcal{L}(w) &= \mathcal{L}(w_{m^*}) + \underbrace{\left[\frac{\partial}{\partial w_{m^*}} \mathcal{L}(w_{m^*}) \right]^T}_{\approx 0} [w - w_{m^*}] + \frac{1}{2} [w - w_{m^*}]^T \left\{ \frac{\partial^2}{\partial w_{m^*}^2} \mathcal{L}(w_{m^*}) \right\} [w - w_{m^*}] \\ \Rightarrow \quad \Delta \mathcal{L}_{\text{MAD}} &= \frac{1}{2} [w - w_{m^*}]^T \left\{ \frac{\partial^2}{\partial w_{m^*}^2} \mathcal{L}(w_{m^*}) \right\} [w - w_{m^*}] \quad (\Delta \mathcal{L}_{\text{MAD}} := \mathcal{L}(w) - \mathcal{L}(w_{m^*})) \\ &= \frac{1}{2} [w \odot (1 - m^*)]^T \left\{ \frac{\partial^2}{\partial w_{m^*}^2} \mathcal{L}(w_{m^*}) \right\} [w \odot (1 - m^*)]. \end{aligned} \tag{18}$$

Through masking optimization in Equation (5) at our manuscript, the first gradient of the loss function $\mathcal{L}$ also approximates zero in adversarial settings. Then, it is verified that Equation (8) at our manuscript is completely aligned with the theoretical analysis of Taylor expansion in Eq. (18), where the provided constraint in Equation (7) at our manuscript is reasonable from this analysis.

D K-FAC for the Fully-Connected Layer

For l^{th} fully-connected layer in DNNs, let us define (activated) layer input $a \in \mathbb{R}^m$, weight matrix $\mathcal{W} \in \mathbb{R}^{m\times n}$, and layer output $z \in \mathbb{R}^n$ such that it satisfies $z = \mathcal{W}^T \times a$, $\nabla_{\mathcal{W}}\mathcal{L} = a\{\nabla_z \mathcal{L}\}^T$, and $F = \mathbb{E}[\{\nabla_{\mathcal{W}}\mathcal{L}\}\{\nabla_{\mathcal{W}}\mathcal{L}\}^T]$. Note that m and n denote the total number of nodes in each l^{th} and $(l+1)^{\text{th}}$ layer.

$$F \approx \mathbb{E}[\{\nabla_z\mathcal{L}\}\{\nabla_z\mathcal{L}\}^T] \otimes \mathbb{E}[aa^T] = \mathcal{Z} \otimes \mathcal{A}, \tag{19}$$

where $\mathcal{Z} \in \mathbb{R}^{n\times n}$ and $\mathcal{A} \in \mathbb{R}^{m\times m}$ stand for correlation of layer output z and layer input a, respectively.

To prune fully-connected layers, we should ensure that MAD does not significantly eliminate model parameters in fully-connected layers so that we prevent nodes in the output layer from being removed which breaks model prediction. Therefore, we limit each node in a fully-connected layer to be cut in equal proportions. In our manuscript, all of experiment results have reflected MAD pruning to whole network parameters on both convolution and fully-connected layers.

E Computing Adversarial Saliency by Block-wise K-FAC

We start from Equation (10) at our manuscript as:

$$\Delta\mathcal{L}_{\text{MAD}} = \frac{1}{2}\Delta w^T \mathcal{Z} \otimes \mathcal{A} \Delta w = \frac{1}{2}\text{Tr}\left[\Delta w \Delta w^T \mathcal{Z} \otimes \mathcal{A}\right], \tag{20}$$

where the variation is denoted by $\Delta w = -w \odot (1-m)$ for MAD constraint, of which vector shape is $\mathbb{R}^{d=c_{\text{out}}c_{\text{in}}k^2}$ in the convolution layer.

Here, we develop a block-wise K-FAC that allows for efficiently computing the adversarial saliency $\Delta\mathcal{L}_{\text{MAD}}$. Since we cannot use full Hessian for all model parameters due to computation limitation and physical memory budget, we instead deal with only of the diagonal terms in the correlation $\mathcal{Z}$ of layer output z, where we consider off-diagonal terms in $\mathcal{Z}$ as zero such that $\mathcal{Z}_{ij} = 0$ $(i \neq j)$. Then, Fisher information matrix can be described with the block-wise K-FAC as follows:

$$F \approx \mathcal{Z} \otimes \mathcal{A} = \begin{bmatrix} \mathcal{Z}_{11}\mathcal{A} & 0 & \cdots & 0 \\ 0 & \mathcal{Z}_{22}\mathcal{A} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \mathcal{Z}_{rr}\mathcal{A} \end{bmatrix}, \tag{21}$$

where $r = c_{\text{out}}$. Once we substitute this Fisher matrix F for Eq. (20), then the following equation is satisfied as:

$$\begin{aligned} \Delta\mathcal{L}_{\text{MAD}} &= \frac{1}{2}\text{Tr}\left[\Delta w \Delta w^T \begin{bmatrix} \mathcal{Z}_{11}\mathcal{A} & 0 & \cdots & 0 \\ 0 & \mathcal{Z}_{22}\mathcal{A} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \mathcal{Z}_{rr}\mathcal{A} \end{bmatrix}\right] \\ &= \frac{1}{2}\text{Tr}\left[\begin{bmatrix} \Delta\mathcal{W}_1 \\ \Delta\mathcal{W}_2 \\ \vdots \\ \Delta\mathcal{W}_r \end{bmatrix} \begin{bmatrix} \Delta\mathcal{W}_1^T & \Delta\mathcal{W}_2^T & \cdots & \Delta\mathcal{W}_r^T \end{bmatrix} \begin{bmatrix} \mathcal{Z}_{11}\mathcal{A} & 0 & \cdots & 0 \\ 0 & \mathcal{Z}_{22}\mathcal{A} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \mathcal{Z}_{rr}\mathcal{A} \end{bmatrix}\right], \end{aligned} \tag{22}$$

where the variation vector Δw can be represented with variation matrix $\Delta\mathcal{W} \in \mathbb{R}^{c_{\text{in}}k^2 \times c_{\text{out}}}$ ($\Delta\mathcal{W}_i$: i^{th} column vector) as:

$$\Delta w = \begin{bmatrix} \Delta\mathcal{W}_1 \\ \Delta\mathcal{W}_2 \\ \vdots \\ \Delta\mathcal{W}_r \end{bmatrix}. \tag{23}$$

Note that $\Delta\mathcal{W} = -\mathcal{W} \odot (1 - \mathcal{M})$, where mask matrix is also reshaped to $\mathcal{M} \in \mathbb{R}^{c_{\text{in}}k^2 \times c_{\text{out}}}$. Then, Eq. (22) is finally expanded as follows:

$$\Delta\mathcal{L}_{\text{MAD}} = \text{Tr}\left[\underbrace{\begin{bmatrix} \frac{1}{2}\mathcal{Z}_{11}\Delta\mathcal{W}_1\Delta\mathcal{W}_1^T\mathcal{A} & 0 & \cdots & 0 \\ 0 & \frac{1}{2}\mathcal{Z}_{22}\Delta\mathcal{W}_2\Delta\mathcal{W}_2^T\mathcal{A} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \frac{1}{2}\mathcal{Z}_{rr}\Delta\mathcal{W}_r\Delta\mathcal{W}_r^T\mathcal{A} \end{bmatrix}}_{\mathcal{B}}\right], \tag{24}$$

where $\mathcal{B} \in \mathcal{R}^{c_{\text{out}}c_{\text{in}}k^2 \times c_{\text{out}}c_{\text{in}}k^2}$ denotes a block matrix, and $\frac{1}{2}\mathcal{Z}_{ii}\Delta\mathcal{W}_i\Delta\mathcal{W}_i^T\mathcal{A}$ $(i = 1, 2, \cdots, r)$ indicates a block diagonal term denoted by $\mathcal{B}_{ii} \in \mathbb{R}^{c_{\text{in}}k^2 \times c_{\text{in}}k^2}$. By using a block-wise K-FAC in convolution layer, we can efficiently compute the adversarial saliency, which can be formulated as follows:

$$\begin{aligned} \Delta\mathcal{L}_{\text{MAD}} &= \text{Tr}\left[\mathcal{B}\right] \\ &= \sum_{i=1}^{r} \text{Tr}\left[\mathcal{B}_{ii}\right] \\ &= \sum_{i=1}^{r} \frac{\mathcal{Z}_{ii}}{2} \text{Tr}\left[\Delta\mathcal{W}_i\Delta\mathcal{W}_i^T\mathcal{A}\right]. \end{aligned} \tag{25}$$

For a block-wise K-FAC in fully-connected layers, we can also calculate the adversarial saliency based on Eq. (19), which can be formulated as follows:

$$\Delta\mathcal{L}_{\text{MAD}} = \sum_{i=1}^{r} \frac{\mathcal{Z}_{ii}}{2} \text{Tr}\left[\Delta\mathcal{W}_i\Delta\mathcal{W}_i^T\mathcal{A}\right], \tag{26}$$

where $r = n$ and we reshape the variation vector $\Delta w \in \mathbb{R}^{d=mn}$ to variation matrix $\Delta\mathcal{W} \in \mathbb{R}^{m \times n}$ ($\Delta\mathcal{W}_i$: i^{th} column vector). Here, it also satisfies $\Delta\mathcal{W} = -\mathcal{W} \odot (1 - \mathcal{M})$, where mask matrix is also reshaped to $\mathcal{M} \in \mathbb{R}^{m \times n}$.

To make sure the benefit of a block-wise K-FAC, in terms of matrix multiplication, we naively calculate the computational complexity of Eq. (20) and Eq. (25) in convolution layer and that of Eq. (20) and Eq. (26) in fully-connected layer. We then identify that the block-wise K-FAC reduces the complexity from $\mathcal{O}(c_{\text{out}}^4 c_{\text{in}}^4 k^8)$ to $\mathcal{O}(c_{\text{out}}^2 c_{\text{in}}^4 k^8)$ in convolution layer and from $\mathcal{O}(n^4 m^4)$ to $\mathcal{O}(n^2 m^4)$ in fully-connected layer. To both layers, the complexity can be lightened by the matrix size r^2 of the correlation $\mathcal{Z}$ for layer output z. Note that we do not regard the complexity of calculating full Hessian in Eq. (20) obtained from the Kronecker product $\otimes$, thus acquiring the adversarial saliency without block-wise K-FAC actually induces more computational burden than $\mathcal{O}(c_{\text{out}}^4 c_{\text{in}}^4 k^8)$ and $\mathcal{O}(n^4 m^4)$.

F Additional Discussion and Ablation Study

Violation of OBD/OBS in Adversarial Settings. As mentioned in section 3.2 at our manuscript, OBD and OBS have achieved outstanding results in standard pruning regime. These methods have proposed a way of discriminating parameter whether the removed parameter truly affects prediction (or loss) according to the loss change from. Nonetheless, it is difficult to directly apply them to adversarial settings, in other words, hard to discriminate whether the parameters are actually salient. The main principle of possibly computing the loss change $\Delta\mathcal{L}$ is based on the fact that the performance of DNNs should be reliable to measure how much the loss change occurs from the predictive criteria $\mathcal{L}(w)$, when a single parameter is removed. However, once adversarial examples are given, $\mathcal{L}(w)$ cannot be operated to be predictive anymore. This aspect is attributed to the performance degradation derived from the fragility of DNNs, since the criteria of adversarial settings is not trustworthy due to the significant drop in accuracy under attack scenario, as seen in Table 1 of our manuscript. This insecurity easily damages loss function so that it causes the first order of Taylor expansion $\frac{\partial \mathcal{L}}{\partial w}$ not to be zero in adversarial settings. Hence, the use of OBD and OBS in adversarial settings brings in the violation of their nature. In line with above theoretical analysis, we have experimented the standard pruning baseline methods (OBD/OBS) in adversarial settings. As in Tab. 1, OBD and OBS shows much poor robustness and even bad benign accuracy compared with those of baselines and MAD in Table 1 at our main manuscript.

Necessity of Block-wise K-FAC. By considering the local geometry of the adversarial loss at multiple parameters points, we investigate which model parameters affect model prediction in adversarial settings. Since the deterministic solution of OBD and OBS for removing multiple parameters at once cannot be derived [49], we propose MAD that considers the combinatorial problem of parameter conjunctions at the multiple points, thereby precisely measuring *adversarial saliency*. For the realization of MAD, we need full second-order information (Hessian) that represents intrinsic correlated connectivity among multiple parameters, but it is intractable to acquire due to high computation complexity of second derivatives. To make it tractable, we introduce a Block-wise K-FAC that can handle a lot more elements than naive diagonal one within physical memory budget. Note that this paper employs K-FAC approximation to Hessian for theoretical scalability among the efficient approximation methods; layer-wise pruning framework (L-OBS) [7], K-FAC approximation [53], and Woodbury formulation (WoodFisher) [49]. Based on Tab. 1, this ablation result confirms that the ground reason for improvements of MAD is due to the numerically estimated adversarial saliency realized with Block-wise K-FAC, by engaging mask optimization for *'look-ahead'* predictive loss.

	VGG						ResNet					
Method	Clean	FGSM	PGD	CW_∞	AP	AA	Clean	FGSM	PGD	CW_∞	AP	AA
OBD	64.1	43.7	41.0	38.5	40.0	37.3	66.5	44.0	41.2	38.7	40.3	37.4
OBS	64.4	44.0	41.5	38.8	40.1	37.4	67.1	44.5	41.8	39.6	41.1	38.6
MAD (**D**)	81.2	54.1	48.9	46.5	47.4	44.6	81.8	56.3	50.6	49.1	49.0	47.0
MAD (**B**)	**81.4**	**57.0**	**51.8**	**47.1**	**50.0**	**45.1**	**82.7**	**58.4**	**53.0**	**50.2**	**51.6**	**48.1**

Table 1: Ablation study of adversarial robustness for MAD using **(D)**iagonal or **(B)**lock-wise K-FAC compared with OBD and OBS on VGG-16 and ResNet-18 with pruning ratio 90% for CIFAR-10.

Feasibility of MAD with a Single Parameter. The main purpose of mask optimization is to make the loss function predictive criteria by controlling multiple parameters so that the loss change can be practically calculated in adversarial settings. Then, it may be natural to throw a question: *'Does MAD adjusting a single parameter instead of multiple ones also work?'*. However, only changing a single parameter does not guarantee that the damaged loss from adversarial examples become a predictive one for all samples in dataset. Therefore, MAD with a single parameter also brings in theoretical limitation, similar to OBD and OBS, which is the violation of the predictive criteria.

Realization of Iterative pruning. Iterative pruning can be another option of MAD, but it can raise two disadvantages. First, every iteration will require calculating Hessian. Although we introduce a Block-wise K-FAC which is an efficient way of computing Hessian, repetitive calculations at every iteration will be computationally striking. Second, the performance of MAD will be dependent on parameter settings such as the number of iterations and controlling pruning ratio per each iteration so that it becomes sensitive according to heuristic choices. Due to these reasons, we propose a one-shot pruning method to be a fundamental pruning work for adversarial settings.