Stochastic Activation Pruning for Robust Adversarial Defense

Guneet S. Dhillon[1,3], Kamyar Azizzadenesheli[4], Aran Khanna[1], Jeremy Bernstein[1,2]
Jean Kossaifi[1,5], Zachary C. Lipton[1,6], Anima Anandkumar[1,2]

[1]Amazon AI, [2]Caltech, [3]UT Austin, [4]UC Irvine, [5]Imperial College London, [6]Carnegie Mellon University

Adversarial Examples

Neural network: h **Parameters:** θ
Input: x **True output:** y

Adversary's objective:

$$\Delta x = \arg\max_{r \sim \rho} J(\theta, x + r, y)$$

where perturbations are **bounded by** l_∞-norm.

Using the Taylor expansion up to the first order term,

$$\Delta x = \arg\max_{r \sim \rho} [J(\theta, x, y) + r^\top \mathcal{J}(\theta, x, y)] \quad \text{where } \mathcal{J} = \frac{\partial J}{\partial x}$$

$$\implies \Delta x = \arg\max_{r \sim \rho} r^\top \mathcal{J}(\theta, x, y)$$

The adversary chooses r to be **in the direction of** $\mathcal{J}(\theta, x, y)$.

So the adversary can choose $\Delta x = \lambda \text{sign}(\mathcal{J}(\theta, x, y))$ (the **fast gradient sign method**).

Minimax Zero-sum Game

Adversary-defender problem (game-theoretic perspective):

$$\pi^*,\ \rho^* := \arg\min_{p \sim \pi} \max_{r \sim \rho} \mathbb{E}_{\pi,\rho}[J(M_p(\theta), x + r, y)]$$

where

- **adversary** tries to **maximize** the loss by **perturbing the input** under policy ρ.
- **defender** tries to **minimize** the loss by **changing model parameters** under policy π.

The optimization problem is a **minimax zero-sum game** between the adversary and defender.
Optimal strategies $(\pi^*,\ \rho^*)$, in general, are mixed Nash equilibrium, i.e. **stochastic policies**.

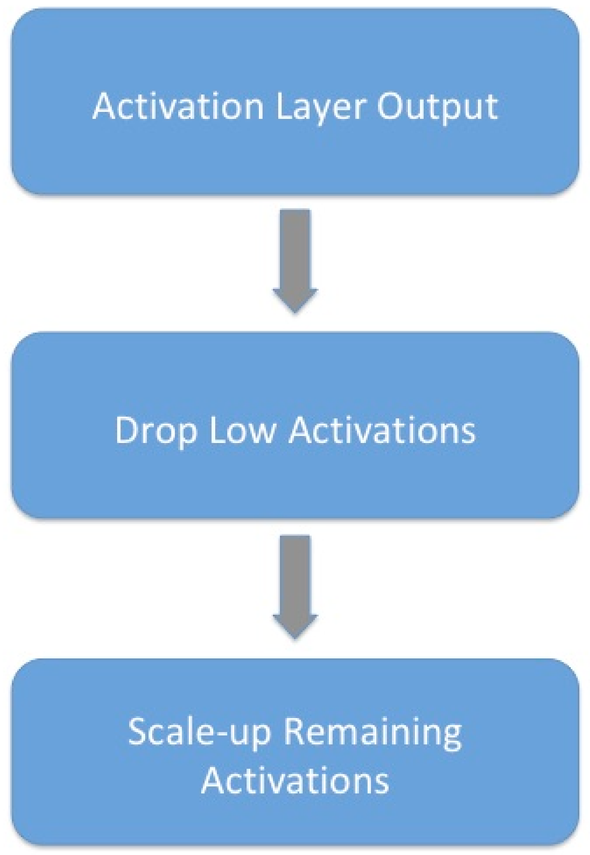
Stochastic Activation Pruning (SAP)

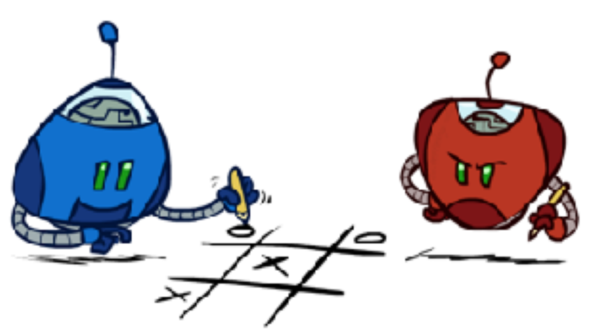
Intuitive idea: stochastically drop out nodes in each layer during forward propagation.

- **Retain nodes** with probabilities **proportional to the magnitude** of their activation.
- **Scale up the surviving nodes** to preserve the dynamic range of the activations in each layer.

- **Preserves the accuracy** of the original model.
- **Better accuracy and calibration** on perturbed input data.
- Can be **applied post-hoc to already-trained models**.

Stochastic Activation Pruning on each Activation Layer





Stochastic Activation Pruning on Pre-Trained Models

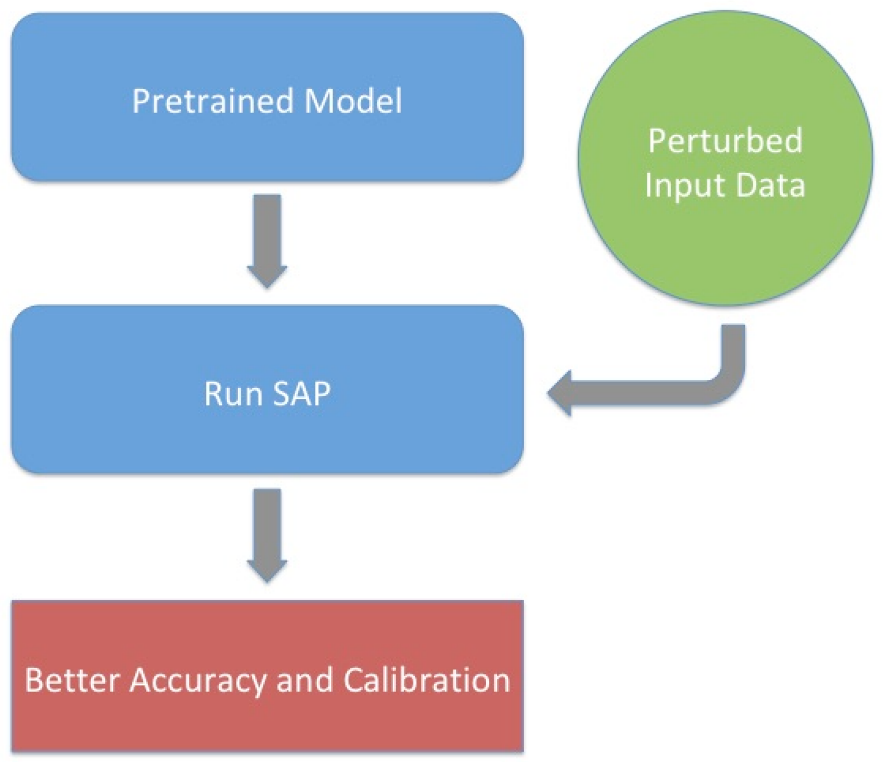
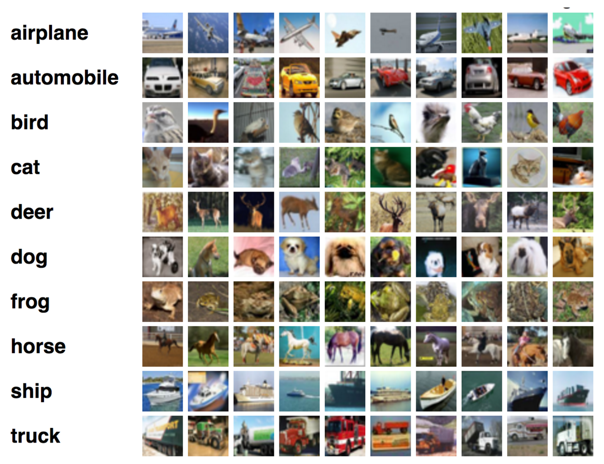


Image Classification



Reinforcement Learning

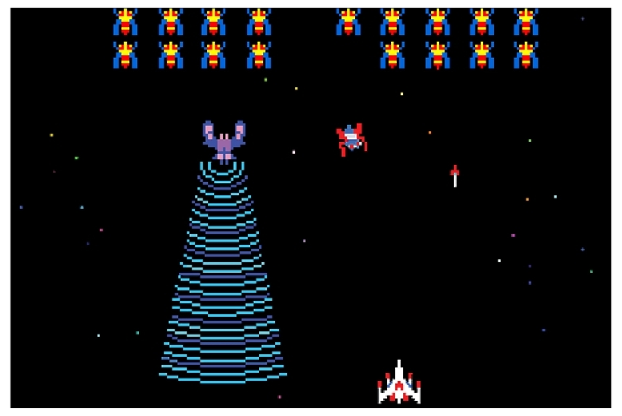
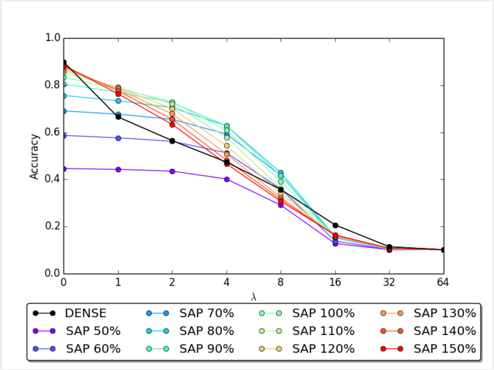
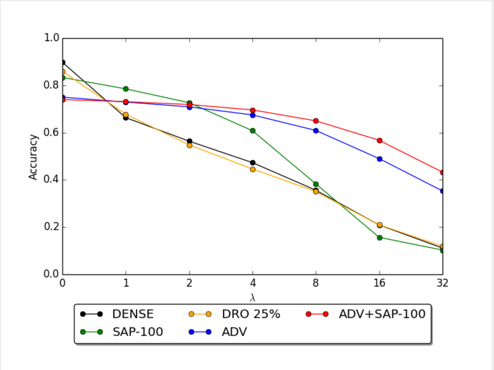


Image Classification

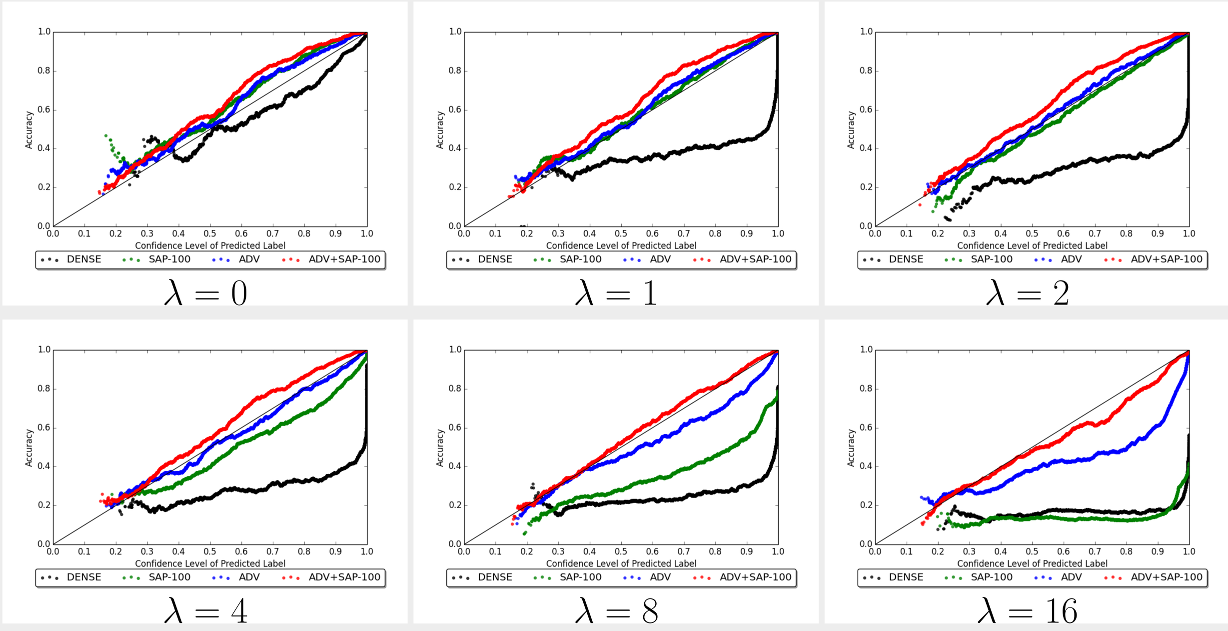
Dataset: Cifar-10 **Model:** ResNet-20
Loss function: Cross-entropy **Non-linearity:** ReLU



Accuracy-λ plots for SAP models. The legend indicates percentage of samples drawn.



Accuracy-λ plots for dense, SAP-100, ADV and ADV+SAP-100 models.



Calibration plots for dense, SAP-100, ADV and ADV+SAP-100 models.

Reinforcement Learning

Games: Multiple Atari games **Model:** Double DQN

Percentage relative increase in rewards gained for SAP-100 compared to original model while playing different Atari games.

λ	Assault	Asterix	BankHeist	BattleZone	BeamRider	Bowling
0	-12.2	-33.4	-59.2	-65.8	-15.8	-4.5
1	10.4	13.3	131.7	-22.0	164.5	3425.9
2	9.8	20.8	204.8	110.1	92.3	
4	12.4	14.0	1760.0	202.6		
8	16.6	7.4	60.9	134.8		

Discussion

- Results in improvements in the robustness of pre-trained models.
- Does not require any additional training.
- In the image classification domain, both the accuracy and calibration of the model improved.
- Can be combined with adversarial training, to compound the benefits.
- In the reinforcement learning domain, is able to defend against adversarial examples better than original model.

Email: guneetdhillon@utexas.edu, kazizzad@uci.edu, arankhan@amazon.com, bernstein@caltech.edu, jean.kossaifi@gmail.com, zlipton@cmu.edu, animakumar@gmail.com