Machine Learning Seminar

Deep Ensembles

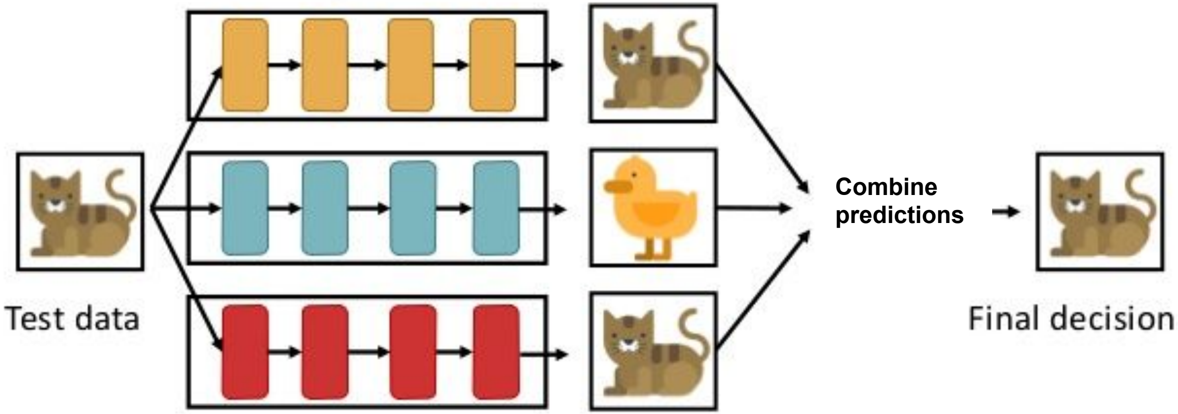
Lorenzo Brigato

Lab Ro.Co.Co - DIAG

SAPIENZA
UNIVERSITÀ DI ROMA

What are Deep Ensembles?

- Ensemble learning
  - Train multiple models to try and solve the same problem
  - Combine the outputs of them to obtain the final decision

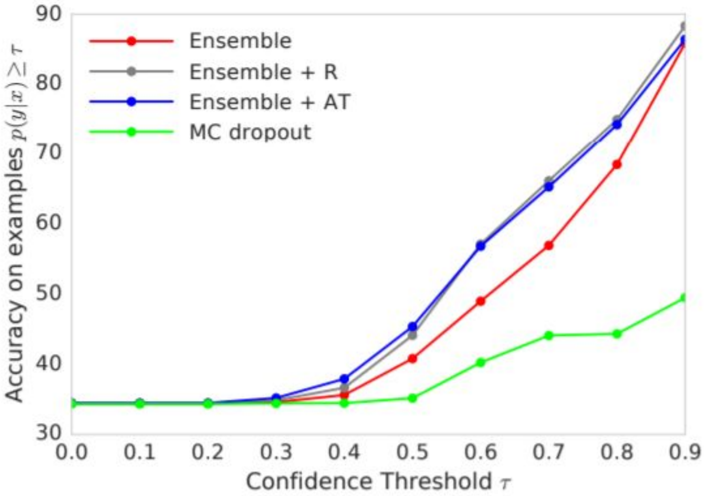


- Bagging [Breiman' 96], boosting [Freund' 99] and mixture of experts [Jacobs' 91]

Strengths and Weaknesses

Strengths:

- Independent training seeks different local minima, hence diverse solutions
- Reduced model variance
  - Better generalization
- Very good at predictive uncertainty



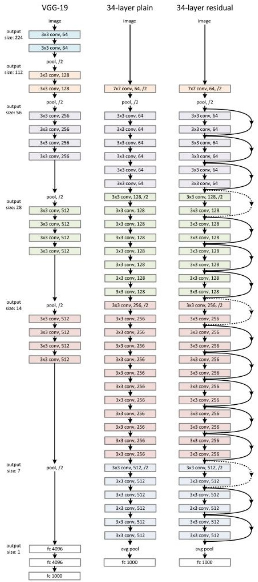
Weaknesses:

- Scaling
  - Training time
  - Memory cost
- Reduced inference time
  - Need to evaluate M nets

Small Data

Great **success** of Deep Learning in many fields:

1. Lots of **data** (e.g. images, text)
2. High **capacity** neural networks (e.g ResNets)



Problem:

1. **Obtaining** data at large scales
   a. time-consuming
   b. difficult
2. **Labeling** data at large scales
   a. expensive

Problem Formulation

We are facing a supervised classification problem $\mathcal{D} = \{(\mathbf{x}_1, \mathbf{y}_1) \ldots, (\mathbf{x}_s, \mathbf{y}_s)\}$

$\mathcal{D}$ is balanced and relatively small (constraining number of samples per class N)

No restriction on the number of classes K

Objective $\mathbf{y} = f_\theta(\mathbf{x})$

In this work:

- $\mathbf{x} \in \mathbb{R}^{H \times W \times D}$
- $N \in \{10, 50, 100, 250\}$

Problem Formulation

Define a set of homogeneous learners:

$$\mathcal{M} = \{g_{\theta_m}(\cdot) : m = 1, \ldots M\}$$

Study deep ensembles making them comparable:

- Fix the total computational cost $\mathcal{C}$
- Vary the complexity of the members

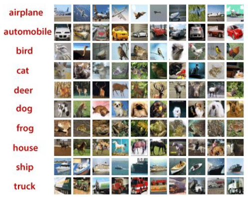
$$\mathcal{M}(g^{(i)}) \sim \mathcal{M}(g^{(j)})$$

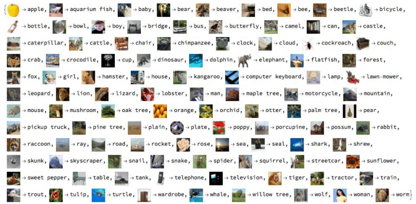
Prediction of our unweighted ensemble with members trained independently:

$$\mathbf{y} = f_\theta(\mathbf{x}) = \frac{1}{M} \sum_{m=1}^{M} \phi\left(g_{\theta_m}(\mathbf{x})\right)$$

Datasets

1. **CIFAR-10/100** - 32x32x3 images with 10/100 classes (e.g. airplane, cat, …)





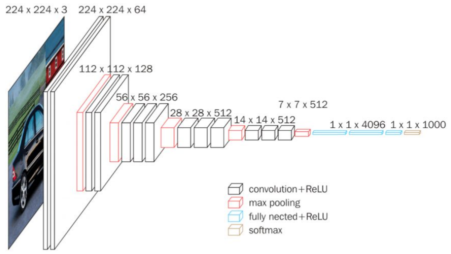
2. **SVHN** - 32x32x3 images of house numbers taken from google street view with 10 classes

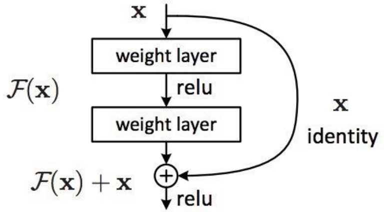
3. **Stanford Dogs** - Larger images (+200 pixels per side) of 120 classes of dogs

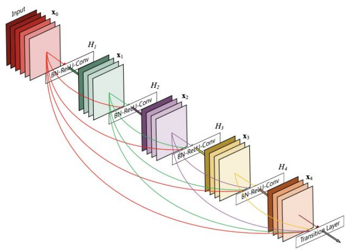
Comparing Ensembles

Ensembles built from **VGG**, **ResNet, DenseNet** families:

- High accuracy on those datasets
- Change model complexity varying depth/width







Defining baselines:

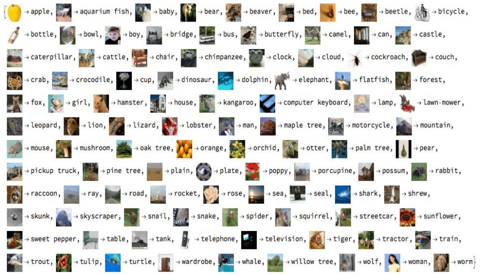
1. A deeper network
2. A shallower and wider network
3. An ensemble of shallower networks (with varying depths)

Notation:

ModelName-Depth-BaseWidth

Comparing to Deeper/Wider Nets

Number of nets	**M**
ResNet-110-16	1
ResNet-8-72	1
ResNet-8-16	20
VGG-9-32	1
VGG-5-76	1
VGG-5-32	5
DenseNet-BC-121, k=32	1
DenseNet-BC-62, k=56	1
DenseNet-BC-62, k=32	3
DenseNet-BC-52, k=12	1
DenseNet-BC-16, k=30	1
DenseNet-BC-16, k=12	6



Regularizing Training

Using standard **data augmentation**:

- Regularization with respect to various transformations
- Cropping, flipping, color distortion (most used)

Also more advanced approaches:

- Random erasing
  - Randomly select a portion of image and add constant or random pixel values

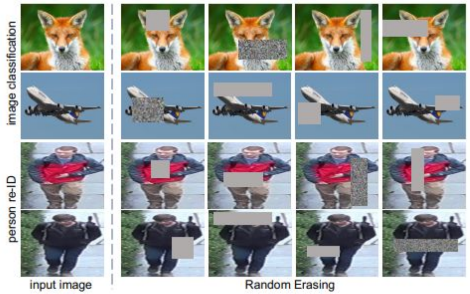


Figure 1. Examples of Random Erasing. In CNN training, we randomly choose a rectangle region in the image and erase its pixels with random values or the ImageNet mean pixel value. Images with various levels of occlusion are thus generated.

Results

Improvements over baselines in **almost all cases** with standard augmentation:

- **Larger gains** on CIFAR with **ResNets** and **VGG** models
- **Significant improvements** of **DenseNets** as well

(a) CIFAR-10

Model	M	N = 10	N = 50	N = 100	N = 250
ResNet-110-16	1	26.06 ± 0.56	41.32 ± 0.58	49.21 ± 1.04	62.5 ± 1.49
ResNet-8-72	1	29.65 ± 1.54	48.0 ± 0.72	58.16 ± 0.37	72.41 ± 0.36
ResNet-8-16	20	**32.83 ± 2.39**	**52.88 ± 0.92**	**63.64 ± 0.61**	**76.23 ± 0.28**
VGG-9-32	1	27.64 ± 1.28	41.74 ± 0.11	47.22 ± 0.42	56.36 ± 1.52
VGG-5-76	1	30.28 ± 1.37	45.39 ± 0.56	51.38 ± 0.72	62.08 ± 1.16
VGG-5-32	5	**31.69 ± 1.03**	**48.61 ± 0.74**	**57.18 ± 0.61**	**68.38 ± 0.47**

(b) CIFAR-100

Model	M	N = 10	N = 50	N = 100	N = 250
ResNet-110-16	1	8.62 ± 1.79	29.44 ± 0.5	40.84 ± 0.41	60.98 ± 1.8
ResNet-8-72	1	16.51 ±0.38	42.52 ± 0.44	54.94 ±0.8	**66.38 ± 0.12**
ResNet-8-16	20	**18.92 ± 0.38**	**46.56 ± 0.41**	**57.37 ± 0.05**	65.56 ± 0.21
VGG-9-32	1	10.22 ± 0.38	23.94 ± 0.34	31.04 ± 0.59	42.09 ± 1.01
VGG-5-76	1	13.25 ± 0.07	26.46 ± 0.36	33.52 ± 0.39	44.84 ± 0.67
VGG-5-32	5	**16.29 ± 0.57**	**34.37 ± 0.33**	**44.04 ± 0.17**	**56.37 ± 0.05**

(c) SVHN

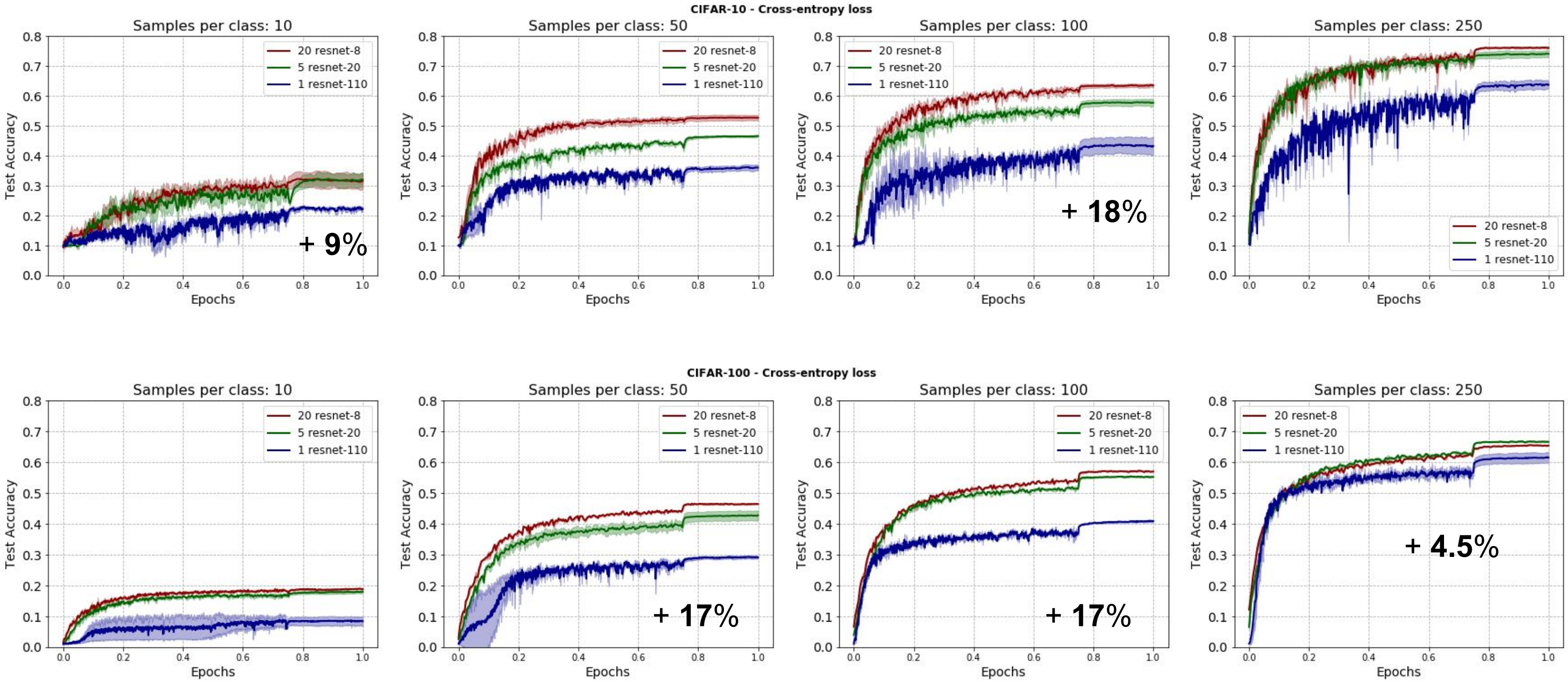
Model	M	N = 10	N = 50	N = 100	N = 250
DenseNet-BC-52, k=12	1	**16.72 ± 1.75**	78.42 ± 1.19	86.52 ± 0.24	89.6 ± 0.7
DenseNet-BC-16, k=30	1	16.44 ± 3.8	76.41 ± 1.65	85.41 ± 0.52	89.28 ± 0.06
DenseNet-BC-16, k=12	6	14.01 ± 2.5	**82.02 ± 1.67**	**87.73 ± 0.44**	**91.61 ± 0.32**

(d) Stanford Dogs

Model	M	N = 10	N = 50	N = 100
DenseNet-BC-121, k=32	1	6.93 ± 0.86	28.32 ± 1.33	47.7 ± 1.17
DenseNet-BC-62, k=56	1	7.33 ± 0.35	29.25 ± 0.76	47.82 ± 0.83
DenseNet-BC-62, k=32	3	**8.42 ± 0.02**	**35.12 ± 0.68**	**53.39 ± 0.45**

Results

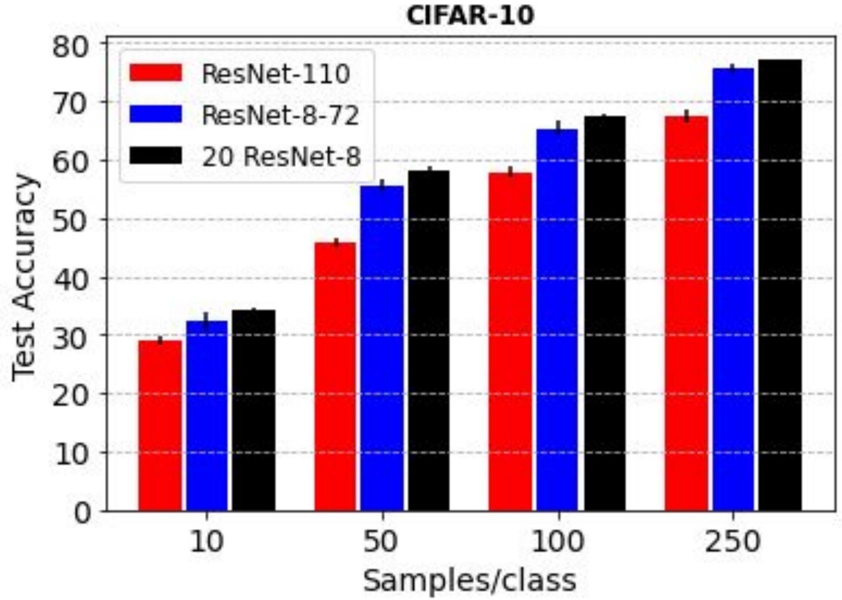
Gains of 20 ResNet-8 over 5 ResNet-20, and 1 ResNet-110 on CIFAR datasets



Results

On CIFAR-10 with aggressive augmentation:

- Still large gaps over deeper model
- Closer value for the wider model



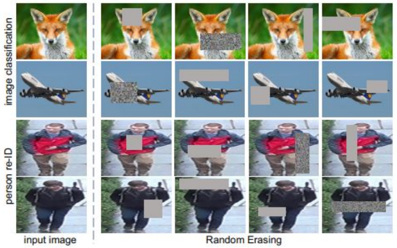
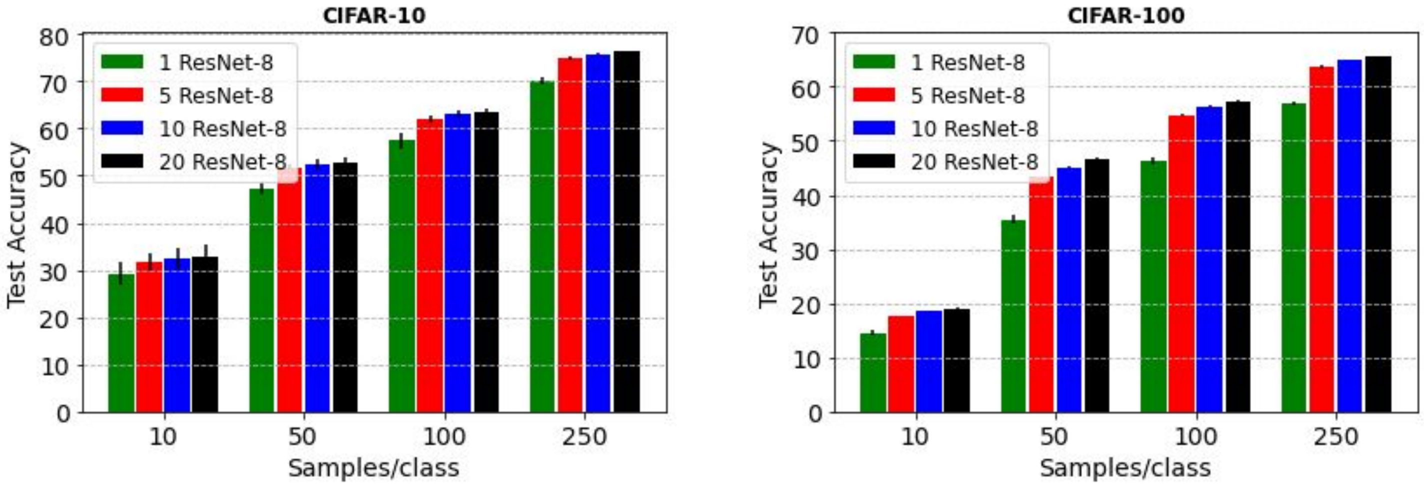


Figure 1. Examples of Random Erasing. In CNN training, we randomly choose a rectangle region in the image and erase its pixels with random values or the ImageNet mean pixel value. Images with various levels of occlusion are thus generated.

How many nets are needed?

Vary the number of nets in the ensembles of ResNets on CIFAR datasets:

- Bigger gap from 1 to 5 nets
- Greater improvements on CIFAR-100



Summary of DE on Small Datasets

Deep ensembles have to be considered when facing a small data problem:

1. Ensembles outperform the wider/deeper single networks
2. Ensembles of small-scale nets outperform smaller ensembles of larger nets

The computational cost is relatively low:

1. Using small-scale networks (e.g. ResNet-8)
2. An ensemble of only 5 ResNet-8 scores already a good performance

Future work for ensembles with small datasets:

1. More complex ensembles techniques (not only simple averaging)
2. Using ensembles to generate more data?