ResNEsts and DenseNEsts: Block-based DNN Models with Improved Representation Guarantees

Kuan-Lin Chen[1], Ching-Hua Lee[1], Harinath Garudadri[2], and Bhaskar D. Rao[1]

[1]Department of Electrical and Computer Engineering, [2]Qualcomm Institute
University of California, San Diego

NeurIPS 2021
December 3, 2021

Outline

1. Motivation
   - Deep learning and the degradation problem
   - Block-based Deep Neural Network (DNN) models
   - Residual blocks are powerful. But why? Are they provably better?
2. Residual Nonlinear Estimator (ResNEst), a generalized and analyzable DNN
   - ResNEsts vs. ResNets
   - New insights and interpretations
   - Non-convex loss landscapes
3. Augmented ResNEsts (A-ResNEsts)
   - Bounding empirical risks via augmentation
   - Wide ResNEsts with bottleneck blocks attain empirical risk lower bounds
   - Bottleneck condition
   - Improved representation guarantees
   - Guarantees on saddle points
   - Empirical results
4. Densely connected Nonlinear Estimators (DenseNEsts)
   - DenseNEsts are wide ResNEsts with bottleneck residual blocks equipped with orthogonalities
5. References

Outline

1. Motivation
   - Deep learning and the degradation problem
   - Block-based Deep Neural Network (DNN) models
   - Residual blocks are powerful. But why? Are they provably better?
2. Residual Nonlinear Estimator (ResNEst), a generalized and analyzable DNN
   - ResNEsts vs. ResNets
   - New insights and interpretations
   - Non-convex loss landscapes
3. Augmented ResNEsts (A-ResNEsts)
   - Bounding empirical risks via augmentation
   - Wide ResNEsts with bottleneck blocks attain empirical risk lower bounds
   - Bottleneck condition
   - Improved representation guarantees
   - Guarantees on saddle points
   - Empirical results
4. Densely connected Nonlinear Estimators (DenseNEsts)
   - DenseNEsts are wide ResNEsts with bottleneck residual blocks equipped with orthogonalities
5. References

Deep learning and the degradation problem

Constructing deep neural network (DNN) models by stacking layers unlocks the field of deep learning, leading to the success in AlexNet (Krizhevsky et al., 2012), VGG (Simonyan and Zisserman, 2015), etc.

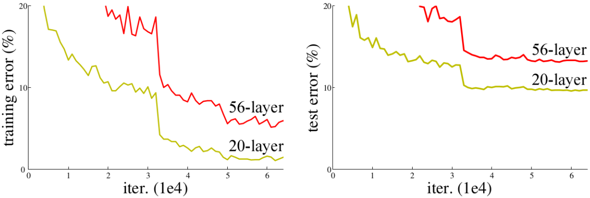


Figure: The degradation problem (He et al., 2016a).

- Stacking more and more layers can suffer from the **degradation problem**.
- Optimization landscapes quickly transition from being nearly convex to being highly chaotic (Li et al., 2018). Stacking more and more layers in DNN models can easily converge to poor local minima.

Outline

Block-based DNN models

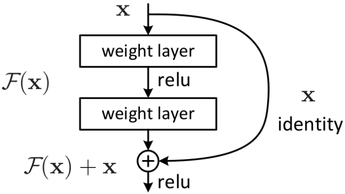


Figure: A residual block (He et al., 2016a).

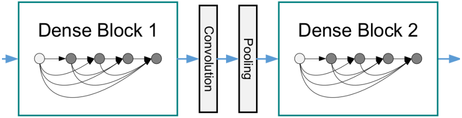


Figure: A dense block (Huang et al., 2017).

- Modern deep learning paradigm has shifted to designing DNN models based on **blocks of the same kind in cascade**.
- A block comprises specific operations on a stack of layers to avoid the degradation problem.
- For example, residual blocks in the ResNet (He et al., 2016a,b; Zagoruyko and Komodakis, 2016; Kim et al., 2016; Xie et al., 2017; Xiong et al., 2018), dense blocks in the DenseNet (Huang et al., 2017), attention blocks in the Transformer (Vaswani et al., 2017), etc.
- ResNets can be even scaled up to 1001 **layers or** 333 **bottleneck residual blocks**, and still improve performance (He et al., 2016b).

Outline

1. Motivation
   - Deep learning and the degradation problem
   - Block-based Deep Neural Network (DNN) models
   - Residual blocks are powerful. But why? Are they provably better?
2. Residual Nonlinear Estimator (ResNEst), a generalized and analyzable DNN
   - ResNEsts vs. ResNets
   - New insights and interpretations
   - Non-convex loss landscapes
3. Augmented ResNEsts (A-ResNEsts)
   - Bounding empirical risks via augmentation
   - Wide ResNEsts with bottleneck blocks attain empirical risk lower bounds
   - Bottleneck condition
   - Improved representation guarantees
   - Guarantees on saddle points
   - Empirical results
4. Densely connected Nonlinear Estimators (DenseNEsts)
   - DenseNEsts are wide ResNEsts with bottleneck residual blocks equipped with orthogonalities
5. References

Residual blocks are powerful. But why? Are they provably better?

- Many applications also adopt residual blocks into their architectures, e.g., Transformer in machine translation (Vaswani et al., 2017), T-GSA in speech enhancement (Kim et al., 2020), V-Net in medical image segmentation (Milletari et al., 2016), etc.
- Despite the huge success, our understanding of ResNets is very limited.

Question 1 (No theory has addressed the following question)

Is learning better ResNets as easy as stacking more blocks?

Outline

1. Motivation
   - Deep learning and the degradation problem
   - Block-based Deep Neural Network (DNN) models
   - Residual blocks are powerful. But why? Are they provably better?
2. Residual Nonlinear Estimator (ResNEst), a generalized and analyzable DNN
   - ResNEsts vs. ResNets
   - New insights and interpretations
   - Non-convex loss landscapes
3. Augmented ResNEsts (A-ResNEsts)
   - Bounding empirical risks via augmentation
   - Wide ResNEsts with bottleneck blocks attain empirical risk lower bounds
   - Bottleneck condition
   - Improved representation guarantees
   - Guarantees on saddle points
   - Empirical results
4. Densely connected Nonlinear Estimators (DenseNEsts)
   - DenseNEsts are wide ResNEsts with bottleneck residual blocks equipped with orthogonalities
5. References

ResNEsts vs. ResNets

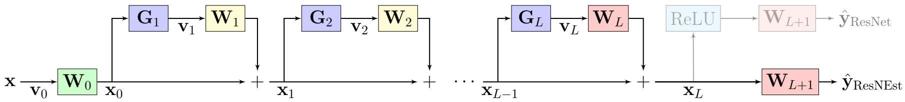


Figure: A generic vector-valued ResNEst that has a chain of L residual blocks (or units).

We consider the proposed ResNEst model shown above whose i-th residual block has the input-output relationship

$$\boldsymbol{x}_i = \boldsymbol{x}_{i-1} + \boldsymbol{W}_i \boldsymbol{G}_i \left(\boldsymbol{x}_{i-1}; \boldsymbol{\theta}_i \right) \tag{1}$$

for $i = 1, 2, \cdots, L$.

- The **nonlinearity** at the final residual representation is **dropped**.
- **Expand** the input space to $\mathbb{R}^M$ to accommodate nonlinear features by $\boldsymbol{W}_0$.
- ResNEsts are **more general** than the models in (Hardt and Ma, 2017; Shamir, 2018; Kawaguchi and Bengio, 2019; Yun et al., 2019).

Outline

Interpretation of basis function modeling in ResNEsts

The input-output relationship for the ResNEst is given by

$$\hat{\boldsymbol{y}}_{L\text{-ResNEst}}(\boldsymbol{x}) = \boldsymbol{W}_{L+1} \sum_{i=0}^{L} \boldsymbol{W}_i \boldsymbol{v}_i(\boldsymbol{x}) \tag{2}$$

where

$$\boldsymbol{v}_i(\boldsymbol{x}) = \boldsymbol{G}_i(\boldsymbol{x}_{i-1}; \boldsymbol{\theta}_i) = \boldsymbol{G}_i\left(\sum_{j=0}^{i-1} \boldsymbol{W}_j \boldsymbol{v}_j; \boldsymbol{\theta}_i\right) \tag{3}$$

for $i = 1, 2, \cdots, L$.

- We define $\boldsymbol{v}_0 = \boldsymbol{v}_0(\boldsymbol{x}) = \boldsymbol{x}$ as the linear feature and regard $\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_L$ as nonlinear features of the input $\boldsymbol{x}$, since $\boldsymbol{G}_i$ is in general nonlinear.
- We do not impose any requirements for each $\boldsymbol{G}_i$.
- The output of a ResNEst $\hat{\boldsymbol{y}}_{L\text{-ResNEst}}$ now can be viewed as a linear function of all these features or **a basis function modeling with a trainable (data-driven) basis**.

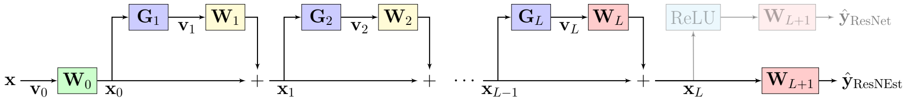


Figure: ResNEst block diagram.

- As opposed to traditional nonlinear methods, the ResNEst jointly finds features and a linear predictor function by solving the ERM problem denoted as (P) on $(\boldsymbol{W}_0, \cdots, \boldsymbol{W}_{L+1}, \boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_L)$.
- Unlike a basis function modeling, the linear predictor function in the ResNEst is not entirely independent of the basis generation process.
- We call such a phenomenon as **a coupling problem** which can **handicap** the performance of ResNEsts.
- The set of parameters $\phi = \{\boldsymbol{W}_{i-1}, \boldsymbol{\theta}_i\}_{i=1}^{L}$ needs to be fixed to sufficiently guarantee that the basis is not changed with different linear predictor functions.
- We refer to $\boldsymbol{W}_L$ and $\boldsymbol{W}_{L+1}$ as prediction weights and $\phi = \{\boldsymbol{W}_{i-1}, \boldsymbol{\theta}_i\}_{i=1}^{L}$ as feature finding weights in the ResNEst.

Because $\boldsymbol{G}_i$ is quite general in the ResNEst, any direct characterization on the landscape of ERM problem seems intractable. Thus, we analyze the following ERM problem

$$(\mathrm{P}_\phi) \min_{\boldsymbol{W}_L, \boldsymbol{W}_{L+1}} \mathcal{R}\left(\boldsymbol{W}_L, \boldsymbol{W}_{L+1}; \phi\right) \tag{4}$$

where

$$\mathcal{R}\left(\boldsymbol{W}_L, \boldsymbol{W}_{L+1}; \phi\right) = \frac{1}{N}\sum_{n=1}^{N} \ell\left(\hat{\boldsymbol{y}}^{\phi}_{L\text{-ResNEst}}\left(\boldsymbol{x}^n\right), \boldsymbol{y}^n\right) \tag{5}$$

for any fixed feature finding weights ϕ.

Remark 1

Since the set of all local minima of (P_ϕ) using any possible features is a superset of the set of all local minima of the original ERM problem (P), any characterization of (P_ϕ) can then be translated to (P).

Outline

Non-convex loss landscapes

Assumption 1

$\sum_{n=1}^{N} \boldsymbol{v}_L(\boldsymbol{x}^n) \boldsymbol{y}^{nT} \neq \boldsymbol{0}$ *and* $\sum_{n=1}^{N} \boldsymbol{v}_L(\boldsymbol{x}^n) \boldsymbol{v}_L(\boldsymbol{x}^n)^T$ *is full rank.*

Proposition 1

If ℓ is the squared loss and Assumption 1 is satisfied, then in (P_ϕ): (i) the objective function is non-convex and non-concave. (ii) every critical point that is not a local minimum is a saddle point.

- The optimization problem (P) is also **non-convex and non-concave**.
- This non-convex loss landscape in (P) immediately raises issues about **suboptimal local minima** in the loss landscape.
- This leads to an important question: Can we guarantee the quality of local minima with respect to some reference models that are known to be good enough?

Outline

1. Motivation
   - Deep learning and the degradation problem
   - Block-based Deep Neural Network (DNN) models
   - Residual blocks are powerful. But why? Are they provably better?
2. Residual Nonlinear Estimator (ResNEst), a generalized and analyzable DNN
   - ResNEsts vs. ResNets
   - New insights and interpretations
   - Non-convex loss landscapes
3. Augmented ResNEsts (A-ResNEsts)
   - Bounding empirical risks via augmentation
   - Wide ResNEsts with bottleneck blocks attain empirical risk lower bounds
   - Bottleneck condition
   - Improved representation guarantees
   - Guarantees on saddle points
   - Empirical results
4. Densely connected Nonlinear Estimators (DenseNEsts)
   - DenseNEsts are wide ResNEsts with bottleneck residual blocks equipped with orthogonalities
5. References

Augmented ResNEsts (A-ResNEsts)

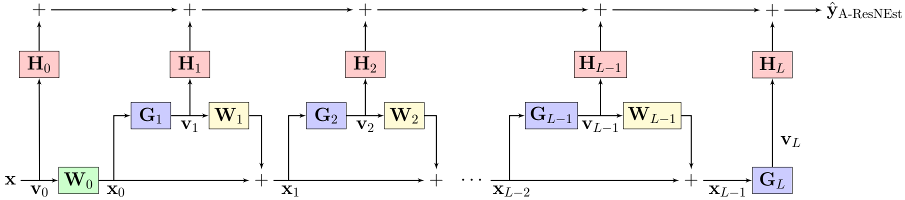


Figure: The proposed Augmented ResNEst or A-ResNEst. A set of new prediction weights $\boldsymbol{H}_0, \boldsymbol{H}_1, \cdots, \boldsymbol{H}_L$ are introduced on top of the features in the ResNEst (see Figure 4).

In the A-ResNEst, (2) is replaced by

$$\hat{\boldsymbol{y}}_{L\text{-A-ResNEst}}(\boldsymbol{x}) = \sum_{i=0}^{L} \boldsymbol{H}_i \boldsymbol{v}_i(\boldsymbol{x}) . \qquad (6)$$

- A-ResNEsts **avoid the coupling problem** that appears in ResNEsts.

Assumption 2

The loss function $\ell(\hat{\boldsymbol{y}}, \boldsymbol{y})$ is differentiable and convex in $\hat{\boldsymbol{y}}$ for any $\boldsymbol{y}$.

Proposition 2

Let $\left(\boldsymbol{H}_0^*, \cdots, \boldsymbol{H}_L^*\right)$ be any local minimizer of the following optimization problem:

$$(PA_\phi) \min_{\boldsymbol{H}_0, \cdots, \boldsymbol{H}_L} \mathcal{A}\left(\boldsymbol{H}_0, \cdots, \boldsymbol{H}_L; \phi\right) \tag{7}$$

where $\mathcal{A}\left(\boldsymbol{H}_0, \cdots, \boldsymbol{H}_L; \phi\right) = \frac{1}{N} \sum_{n=1}^{N} \ell\left(\hat{\boldsymbol{y}}^{\phi}_{\text{L-A-ResNEst}}\left(\boldsymbol{x}^n\right), \boldsymbol{y}^n\right)$. If Assumption 2 is satisfied, then the above optimization problem is convex and

$$\epsilon\left(\boldsymbol{W}_L^*, \boldsymbol{W}_{L+1}^*; \phi\right) = \mathcal{R}\left(\boldsymbol{W}_L^*, \boldsymbol{W}_{L+1}^*; \phi\right) - \mathcal{A}\left(\boldsymbol{H}_0^*, \cdots, \boldsymbol{H}_L^*; \phi\right) \geq 0 \tag{8}$$

for any local minimizer $\left(\boldsymbol{W}_L^*, \boldsymbol{W}_{L+1}^*\right)$ of (P_ϕ) using arbitrary feature finding parameters ϕ.

Necessary condition for strictly improved residual representations

Question 2

What properties are fundamentally required for features to be good, i.e., able to strictly improve the residual representation over blocks?

- A fundamental answer is they need to be at least **linearly unpredictable**.
- Note that $\boldsymbol{v}_i$ must be linearly unpredictable by $\boldsymbol{v}_0, \cdots, \boldsymbol{v}_{i-1}$ if

$$\mathcal{A}\left(\boldsymbol{H}_0^*, \cdots, \boldsymbol{H}_{i-1}^*, \boldsymbol{0}, \cdots, \boldsymbol{0}, \phi^*\right) > \mathcal{A}\left(\boldsymbol{H}_0^*, \boldsymbol{H}_1^*, \cdots, \boldsymbol{H}_i^*, \boldsymbol{0}, \cdots, \boldsymbol{0}, \phi^*\right) \quad (9)$$

  for any local minimum $\left(\boldsymbol{H}_0^*, \cdots, \boldsymbol{H}_L^*, \phi^*\right)$ in (PA).
- Fortunately, the linearly unpredictability of $\boldsymbol{v}_i$ is usually satisfied when $\boldsymbol{G}_i$ is nonlinear.

Outline

1. Motivation
   - Deep learning and the degradation problem
   - Block-based Deep Neural Network (DNN) models
   - Residual blocks are powerful. But why? Are they provably better?
2. Residual Nonlinear Estimator (ResNEst), a generalized and analyzable DNN
   - ResNEsts vs. ResNets
   - New insights and interpretations
   - Non-convex loss landscapes
3. Augmented ResNEsts (A-ResNEsts)
   - Bounding empirical risks via augmentation
   - Wide ResNEsts with bottleneck blocks attain empirical risk lower bounds
   - Bottleneck condition
   - Improved representation guarantees
   - Guarantees on saddle points
   - Empirical results
4. Densely connected Nonlinear Estimators (DenseNEsts)
   - DenseNEsts are wide ResNEsts with bottleneck residual blocks equipped with orthogonalities
5. References

Assumption 3

$M \geq N_o$ where N_o is the output dimension of the network.

Assumption 4

The linear inverse problem $\boldsymbol{x}_{L-1} = \sum_{i=0}^{L-1} \boldsymbol{W}_i \boldsymbol{v}_i$ has a unique solution.

Theorem 1

If Assumption 2 and 3 are satisfied, then in (P_ϕ) under any ϕ such that Assumption 4 holds: (i) every critical point with full rank $\boldsymbol{W}_{L+1}$ is a global minimum. (ii) $\epsilon\left(\boldsymbol{W}_L^*, \boldsymbol{W}_{L+1}^*; \phi\right) = 0$ for every local minimizer.

- Every local minimum of (P_ϕ) is also a **global** minimum despite its non-convex landscape (Proposition 1).
- Replacing "in (P_ϕ) under any ϕ" with just "(P)" in Theorem 1 produces the same results. May gain more clarity, but more restricted.
- **Not limited to fixing any weights during training**; and it applies to both normal training and blockwise training procedures.

Outline

Bottleneck condition

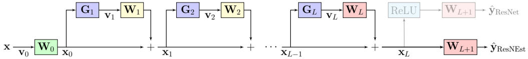
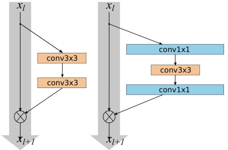


Figure: ResNEst block diagram.



(a) basic (b) bottleneck

Figure: Basic vs. bottleneck.

- A ResNEst needs to be wide enough such that

$$M \geq \sum_{i=0}^{L-1} K_i \tag{10}$$

to necessarily satisfy Assumption 4.

- We call such a sufficient condition on the width and feature dimensionalities as a **bottleneck condition**.
- Without the expansion, the dimenionality of the residual representation is always limited to the input dimension. As a result, Assumption 4 cannot be satisfied for $L > 1$.

Outline

Improved representation guarantees

Remark 2

Let Assumption 2 and 3 be true. Any local minimizer obtained in (P) such that Assumption 4 is satisfied guarantees:

- *(i) monotonically improved (no worse) residual representations over blocks.*
- *(ii) every residual representation is better than the input representation in the linear prediction sense.*

- Although there may exist suboptimal local minima in the optimization problem (P), Remark 2 suggests that such minima **still improve residual representations over blocks** under practical conditions.

Corollary 2

Let Assumption 2 and 3 be true. Any local minimum of $(P_{\boldsymbol{\alpha}})$ *is smaller than or equal to any local minimum of* $(P_{\boldsymbol{\beta}})$ *under Assumption 4 for any* $\boldsymbol{\alpha} = \{\boldsymbol{W}_{i-1}, \boldsymbol{\theta}_i\}_{i=1}^{L_\alpha}$ *and* $\boldsymbol{\beta} = \{\boldsymbol{W}_{i-1}, \boldsymbol{\theta}_i\}_{i=1}^{L_\beta}$ *where* L_α *and* L_β *are positive integers such that* $L_\alpha > L_\beta$.

Corollary 3

Let $\left(\boldsymbol{W}_0^*, \cdots, \boldsymbol{W}_{L+1}^*, \boldsymbol{\theta}_1^*, \cdots, \boldsymbol{\theta}_L^*\right)$ *be any local minimizer of (P) and* $\phi^* = \{\boldsymbol{W}_{i-1}^*, \boldsymbol{\theta}_i^*\}_{i=1}^{L}$. *If Assumption 2, 3 and 4 are satisfied, then (i)*

$$\mathcal{R}\left(\boldsymbol{W}_0^*, \cdots, \boldsymbol{W}_{L+1}^*, \boldsymbol{\theta}_1^*, \cdots, \boldsymbol{\theta}_L^*\right) \leq \min_{\boldsymbol{A} \in \mathbb{R}^{N_o \times N_{in}}} \frac{1}{N} \sum_{n=1}^{N} \ell\left(\boldsymbol{A}\boldsymbol{x}^n, \boldsymbol{y}^n\right) \tag{11}$$

and (ii) the above inequality is strict if $\mathcal{A}\left(\boldsymbol{H}_0^*, \mathbf{0}, \cdots, \mathbf{0}, \phi^*\right) > \mathcal{A}\left(\boldsymbol{H}_0^*, \cdots, \boldsymbol{H}_L^*, \phi^*\right)$.

Outline

Theorem 4

If ℓ is the squared loss, and Assumption 1 and 3 are satisfied, then in the optimization problem (P_ϕ) under any ϕ such that Assumption 4 holds: (i) $\boldsymbol{W}_{L+1}$ is rank-deficient at every saddle point. (ii) there exists at least one direction with strictly negative curvature at every saddle point.

- Although (P_ϕ) is a non-convex optimization problem according to Proposition 1 (i), Theorem 4 (ii) suggests a desirable property for saddle points in the loss landscape.
- Again, we require the **bottleneck condition** to be satisfied in order to guarantee such a nice property about saddle points.
- Theorem 4 is not limited to fixing any weights during training; and it applies to both normal training and blockwise training procedures.

Outline

Empirical results

Archit. \ Type	Standard	ResNEst	BN-ResNEst	A-ResNEst
WRN-16-8	95.56% (11M)	94.39% (11M)	95.48% (11M)	95.29% (8.7M)
WRN-40-4	95.45% (9.0M)	94.58% (9.0M)	95.61% (9.0M)	95.48% (8.4M)
ResNet-110	94.46% (1.7M)	92.77% (1.7M)	94.52% (1.7M)	93.97% (1.7M)
ResNet-20	92.60% (0.27M)	91.02% (0.27M)	92.56% (0.27M)	92.47% (0.24M)

Table: CIFAR-10.

Archit. \ Type	Standard	ResNEst	BN-ResNEst	A-ResNEst
WRN-16-8	79.14% (11M)	75.43% (11M)	78.99% (11M)	78.74% (8.9M)
WRN-40-4	79.08% (9.0M)	75.16% (9.0M)	78.97% (9.0M)	78.62% (8.7M)
ResNet-110	74.08% (1.7M)	69.08% (1.7M)	73.95% (1.7M)	72.53% (1.9M)
ResNet-20	68.56% (0.28M)	64.73% (0.28M)	68.47% (0.28M)	68.16% (0.27M)

Table: CIFAR-100.

- A-ResNEsts empirically exhibit competitive performance to standard ResNets.
- Keeping the batch normalization and simply dropping the ReLU at the final residual representation in standard pre-activation ResNets gives competitive performance.

Outline

1. Motivation
   - Deep learning and the degradation problem
   - Block-based Deep Neural Network (DNN) models
   - Residual blocks are powerful. But why? Are they provably better?
2. Residual Nonlinear Estimator (ResNEst), a generalized and analyzable DNN
   - ResNEsts vs. ResNets
   - New insights and interpretations
   - Non-convex loss landscapes
3. Augmented ResNEsts (A-ResNEsts)
   - Bounding empirical risks via augmentation
   - Wide ResNEsts with bottleneck blocks attain empirical risk lower bounds
   - Bottleneck condition
   - Improved representation guarantees
   - Guarantees on saddle points
   - Empirical results
4. Densely connected Nonlinear Estimators (DenseNEsts)
   - DenseNEsts are wide ResNEsts with bottleneck residual blocks equipped with orthogonalities
5. References

Densely connected Nonlinear Estimators (DenseNEsts)

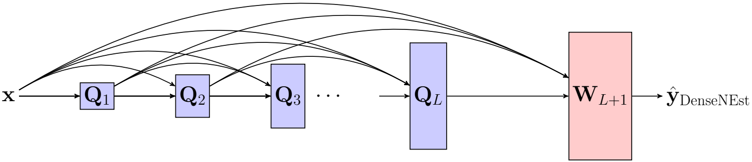


Figure: DenseNEst block diagram.

Proposition 3

If Assumption 2 is satisfied, then any local minimum of (PD) is smaller than or equal to the minimum empirical risk given by any linear predictor of the input.

- **No special architectural design in a DenseNEst is required** to make sure it always outperforms the best linear predictor.

DenseNEsts are wide ResNEsts with bottleneck residual blocks equipped with orthogonalities

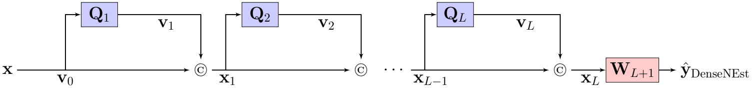


Figure: DenseNEst block diagram.

Proposition 4

Given any L-block DenseNEst $\hat{\boldsymbol{y}}_{L\text{-DenseNEst}}$, there exists a wide L-ResNEst with bottleneck residual blocks $\hat{\boldsymbol{y}}_{L\text{-ResNEst}}$ such that $\hat{\boldsymbol{y}}_{L\text{-ResNEst}}(\boldsymbol{x}) = \hat{\boldsymbol{y}}_{L\text{-DenseNEst}}(\boldsymbol{x})$ for all $\boldsymbol{x} \in \mathbb{R}^{N_{in}}$ and $\epsilon = 0$ for all local minima.

- Any DenseNEst can be viewed as a ResNEst satisfying Assumption 4.
- Proposition 4 can be regarded as a **theoretical support** for why standard DenseNets (Huang et al., 2017) are in general better than standard ResNets (He et al., 2016b).

References I

Hardt, M. and Ma, T. (2017). Identity matters in deep learning. In *International Conference on Learning Representations*.

He, K., Zhang, X., Ren, S., and Sun, J. (2016a). Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition*, pages 770–778. IEEE.

He, K., Zhang, X., Ren, S., and Sun, J. (2016b). Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pages 630–645. Springer.

Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Conference on Computer Vision and Pattern Recognition*, pages 4700–4708. IEEE.

Kawaguchi, K. and Bengio, Y. (2019). Depth with nonlinearity creates no bad local minima in resnets. *Neural Networks*, 118:167–174.

Kim, J., El-Khamy, M., and Lee, J. (2020). T-GSA: Transformer with gaussian-weighted self-attention for speech enhancement. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6649–6653. IEEE.

Kim, J., Kwon Lee, J., and Mu Lee, K. (2016). Accurate image super-resolution using very deep convolutional networks. In *Conference on Computer Vision and Pattern Recognition*, pages 1646–1654. IEEE.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105.

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. (2018). Visualizing the loss landscape of neural nets. In *Advances in Neural Information Processing Systems*, pages 6389–6399.

Milletari, F., Navab, N., and Ahmadi, S.-A. (2016). V-Net: Fully convolutional neural networks for volumetric medical image segmentation. In *International Conference on 3D Vision*, pages 565–571. IEEE.

Shamir, O. (2018). Are resnets provably better than linear predictors? In *Advances in Neural Information Processing Systems*, pages 507–516.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Xie, S., Girshick, R., Dollár, P., Tu, Z., and He, K. (2017). Aggregated residual transformations for deep neural networks. In *Conference on Computer Vision and Pattern Recognition*, pages 1492–1500. IEEE.

Xiong, W., Wu, L., Alleva, F., Droppo, J., Huang, X., and Stolcke, A. (2018). The Microsoft 2017 conversational speech recognition system. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5934–5938. IEEE.

Yun, C., Sra, S., and Jadbabaie, A. (2019). Are deep resnets provably better than linear predictors? In *Advances in Neural Information Processing Systems*, pages 15686–15695.

Zagoruyko, S. and Komodakis, N. (2016). Wide residual networks. In *British Machine Vision Conference (BMVC)*, pages 87.1–87.12. BMVA Press.