Dropout as a Structured Shrinkage Prior

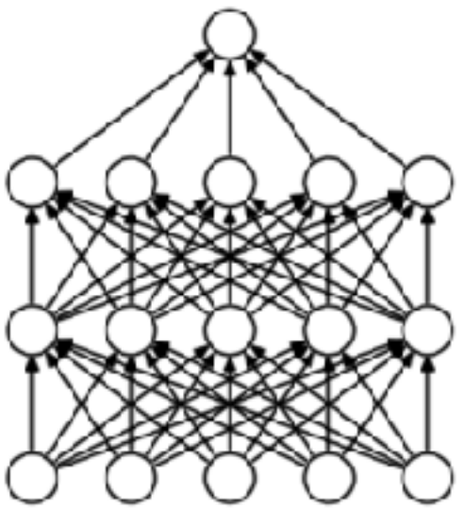
Eric Nalisnick

UNIVERSITY OF CAMBRIDGE

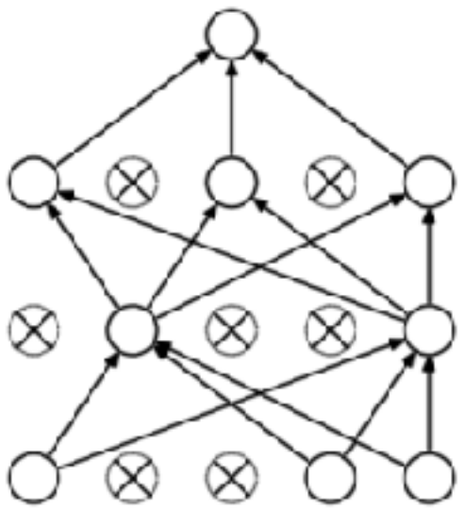
Computational and Biological Learning
University of Cambridge

Dropout Regularization

Randomly set hidden units to zero (i.e. drop them) for every forward pass during training [Hinton et al., 2012].



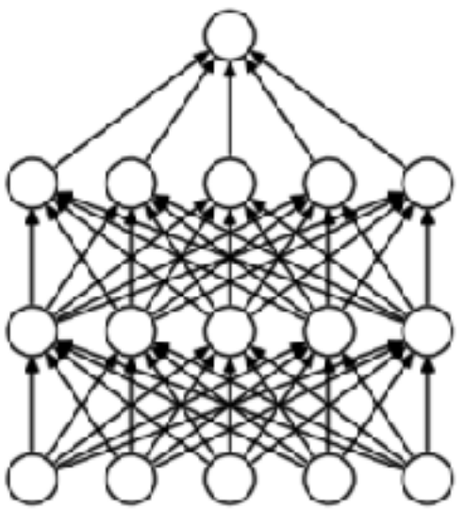
Standard Neural Network



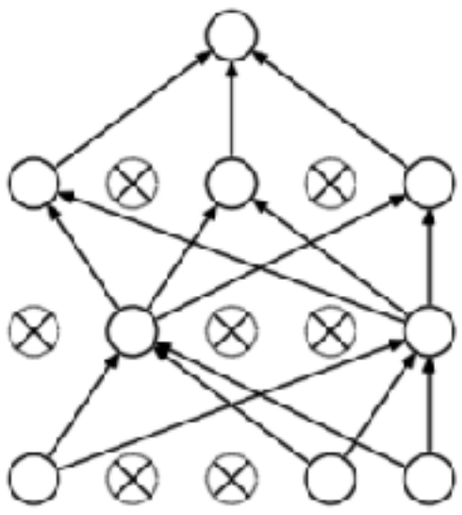
After Applying Dropout

Dropout Regularization

Randomly set hidden units to zero (i.e. drop them) for every forward pass during training [Hinton et al., 2012].



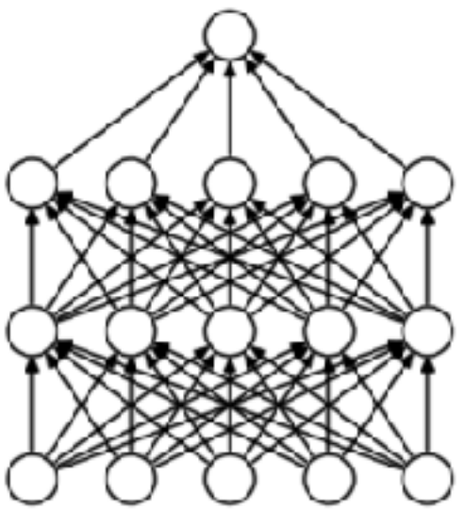
Standard Neural Network



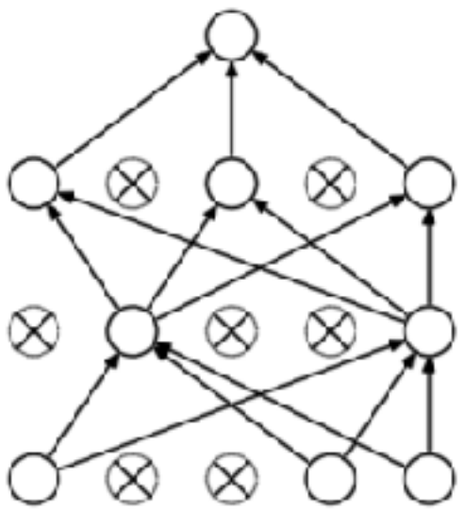
After Applying Dropout

Dropout Regularization

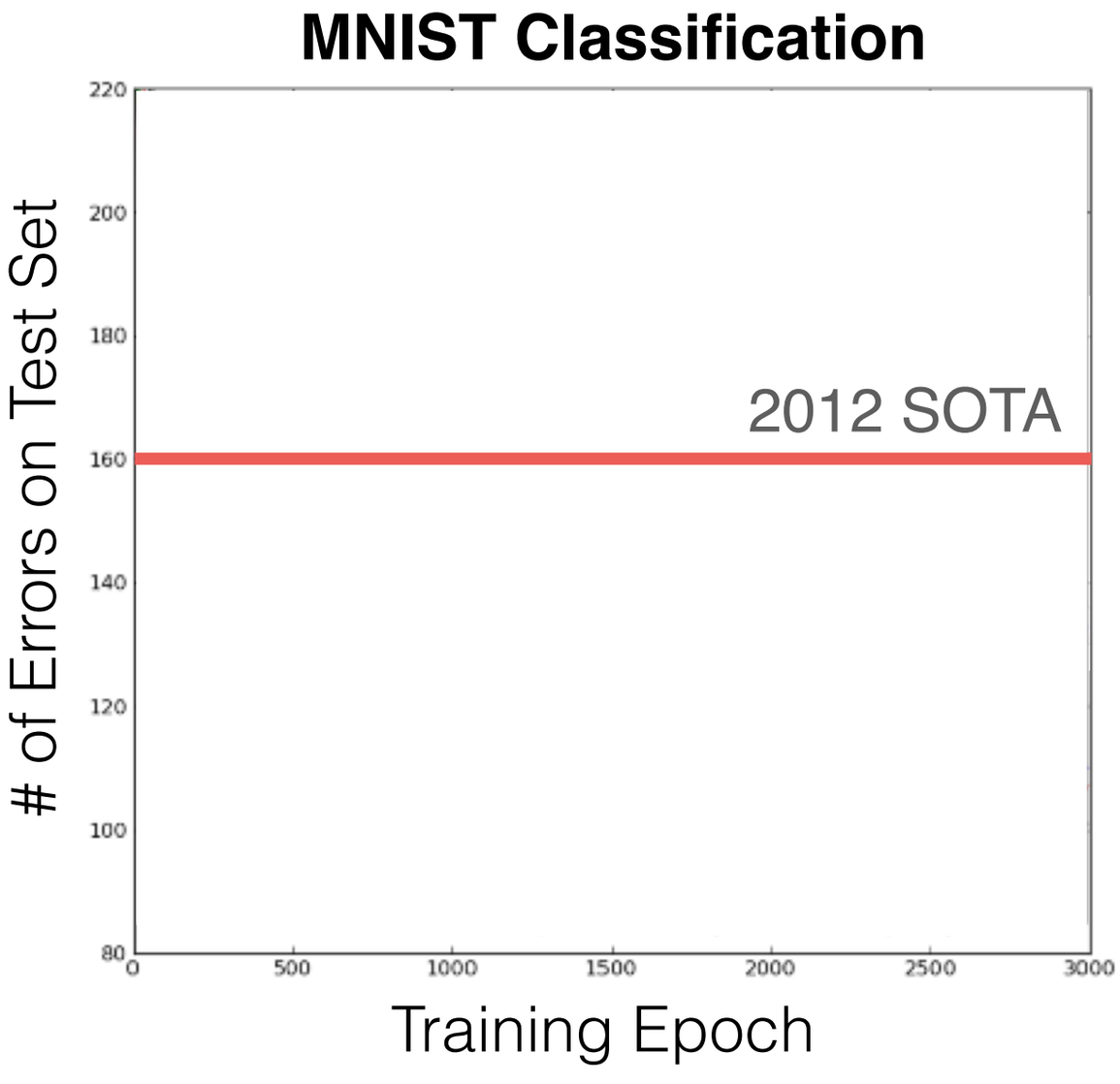
Randomly set hidden units to zero (i.e. drop them) for every forward pass during training [Hinton et al., 2012].



Standard Neural Network

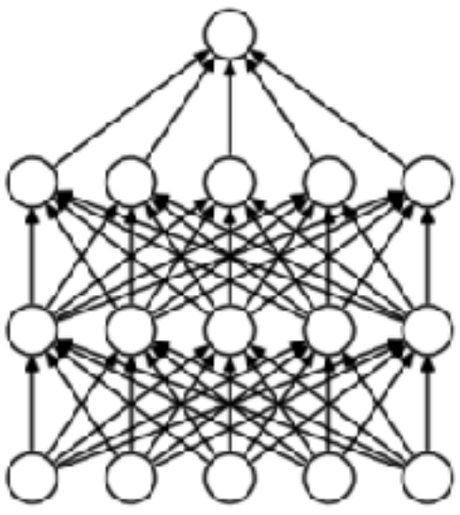


After Applying Dropout

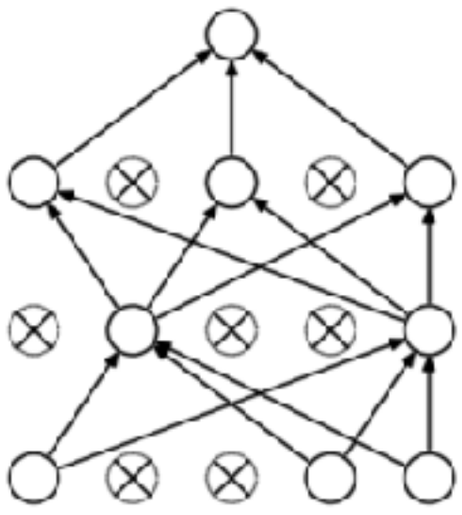


Dropout Regularization

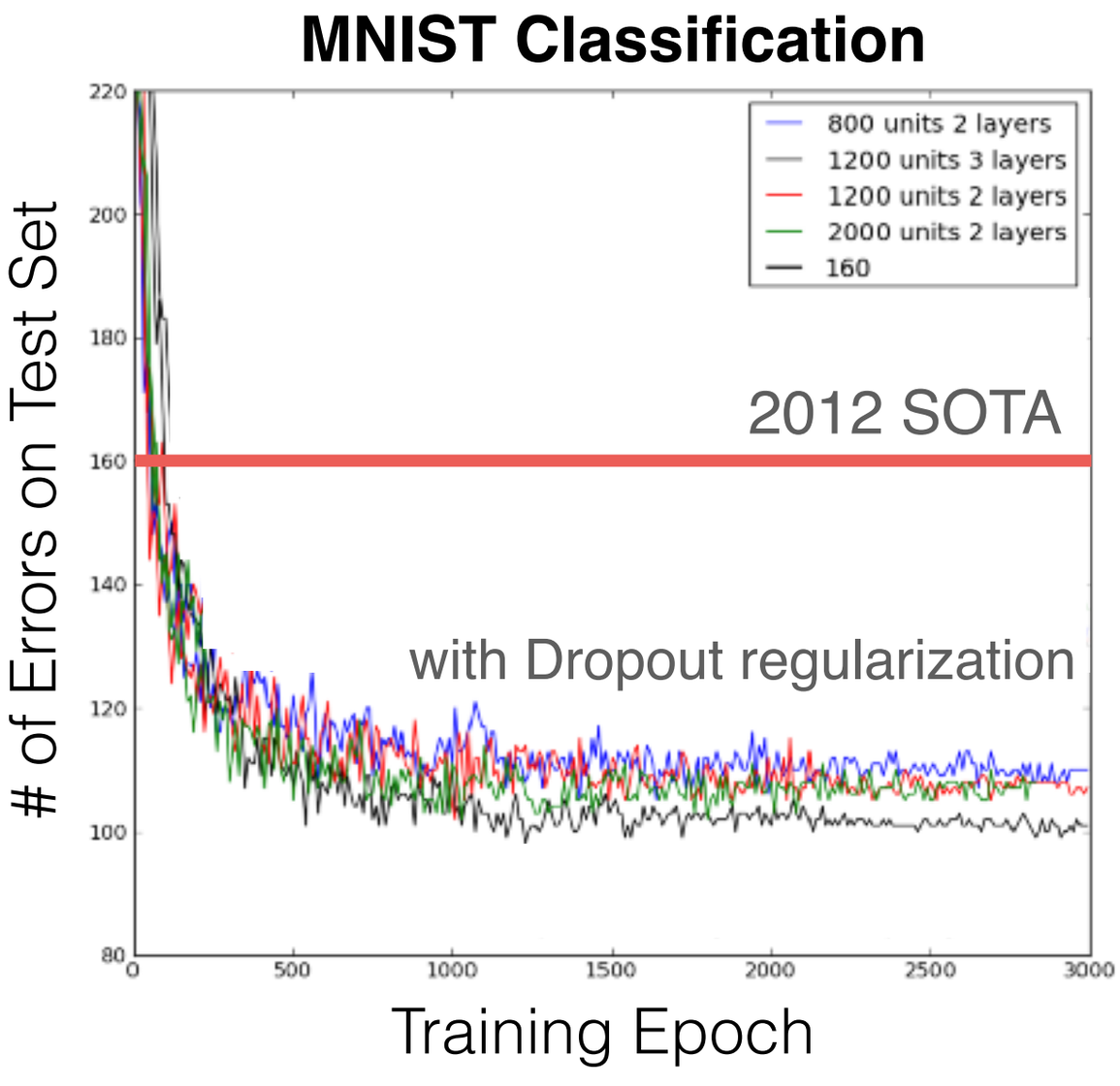
Randomly set hidden units to zero (i.e. drop them) for every forward pass during training [Hinton et al., 2012].



Standard Neural Network



After Applying Dropout



Dropout Regularization

Geoffrey Hinton

2018 Turing Award winner for Deep Learning

Source: https://www.reddit.com/r/MachineLearning/comments/4w6tsv/ama_we_are_the_google_brain_team_wed_love_to/d6dgyse

Dropout Regularization

I went to my bank. The tellers kept changing….I figured it must be because it would require cooperation between employees to successfully defraud the bank. This made me realize that randomly removing… neurons…would prevent conspiracies and thus reduce overfitting.

Geoffrey Hinton

2018 Turing Award winner for Deep Learning

Source: https://www.reddit.com/r/MachineLearning/comments/4w6tsv/ama_we_are_the_google_brain_team_wed_love_to/d6dgyse

Dropout Regularization

I went to my bank. The tellers kept changing….I figured it must be because it would require cooperation between employees to successfully defraud the bank. This made me realize that randomly removing… neurons…would prevent conspiracies and thus reduce overfitting.

Geoffrey Hinton

2018 Turing Award winner for Deep Learning

Source: https://www.reddit.com/r/MachineLearning/comments/4w6tsv/ama_we_are_the_google_brain_team_wed_love_to/d6dgyse

Dropout Regularization

I went to my bank. The tellers kept changing….I figured it must be because it would require cooperation between employees to successfully defraud the bank. *This made me realize that randomly removing… neurons…would prevent conspiracies and thus reduce overfitting.*

Geoffrey Hinton

2018 Turing Award winner for Deep Learning

Source: https://www.reddit.com/r/MachineLearning/comments/4w6tsv/ama_we_are_the_google_brain_team_wed_love_to/d6dgyse

Dropout Regularization

I went to my bank. The tellers kept changing….I figured it must be because it would require cooperation between employees to successfully defraud the bank. This made me realize that randomly removing neurons…would prevent conspiracies and thus reduce overfitting.

Geoffrey Hinton

2018 Turing Award winner for Deep Learning

Source: https://www.reddit.com/r/MachineLearning/comments/4w6tsv/ama_we_are_the_google_brain_team_wed_love_to/d6dgyse

Dropout Regularization

$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\mathbf{W}_l)$$

Dropout Regularization

$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\mathbf{W}_l)$$

Weight Matrix

Hidden Units

‘Activation’ function

Dropout Regularization

Implementation as *multiplicative noise*:

$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\mathbf{\Lambda}_l\mathbf{W}_l)$$

Dropout Regularization

Implementation as *multiplicative noise*:

$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\mathbf{\Lambda}_l\mathbf{W}_l) \qquad \lambda_{l,i,i} \sim p(\lambda)$$

Diagonal Matrix of Random Variables

Dropout Regularization

Implementation as *multiplicative noise*:

$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\mathbf{\Lambda}_l\mathbf{W}_l) \qquad \lambda_{l,i,i} \sim p(\lambda)$$

Diagonal Matrix of Random Variables

- ⊗ Dropout corresponds to **p(λ)** being Bernoulli.

Dropout Regularization

Implementation as *multiplicative noise*:

$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\mathbf{\Lambda}_l\mathbf{W}_l) \qquad \lambda_{l,i,i} \sim p(\lambda)$$

Diagonal Matrix of Random Variables

- ⊗ Dropout corresponds to **p(λ)** being Bernoulli.
- ⊗ Gaussian, beta, and uniform noise also work well.

Dropout Regularization

Implementation as *multiplicative noise*:

$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\boldsymbol{\Lambda}_l\mathbf{W}_l) \qquad \lambda_{l,i,i} \sim p(\lambda)$$

Diagonal Matrix of Random Variables

- ⊗ Dropout corresponds to **p(λ)** being Bernoulli.
- ⊗ Gaussian, beta, and uniform noise also work well.
- ⊗ Optimization objective:

$$\mathbb{E}_\lambda \left[\log\ p\left(\mathbf{y} \,|\, \mathbf{X}, \{\mathbf{W}_l\}_{l=1}^{L+1}, \{\boldsymbol{\Lambda}_l\}_{l=1}^{L+1} \right) \right]$$

Bayesian Interpretation of Dropout

- ⊗ Gal & Ghahramani [2016] argued that dropout can be interpreted as variational Bayesian inference.

  $p(\lambda)$ serves as a variational approximation to the posterior $p(\lambda \mid \mathbf{y}, \mathbf{X})$

Bayesian Interpretation of Dropout

- Gal & Ghahramani [2016] argued that dropout can be interpreted as variational Bayesian inference.

- Obtain predictive uncertainty by averaging over noise samples.

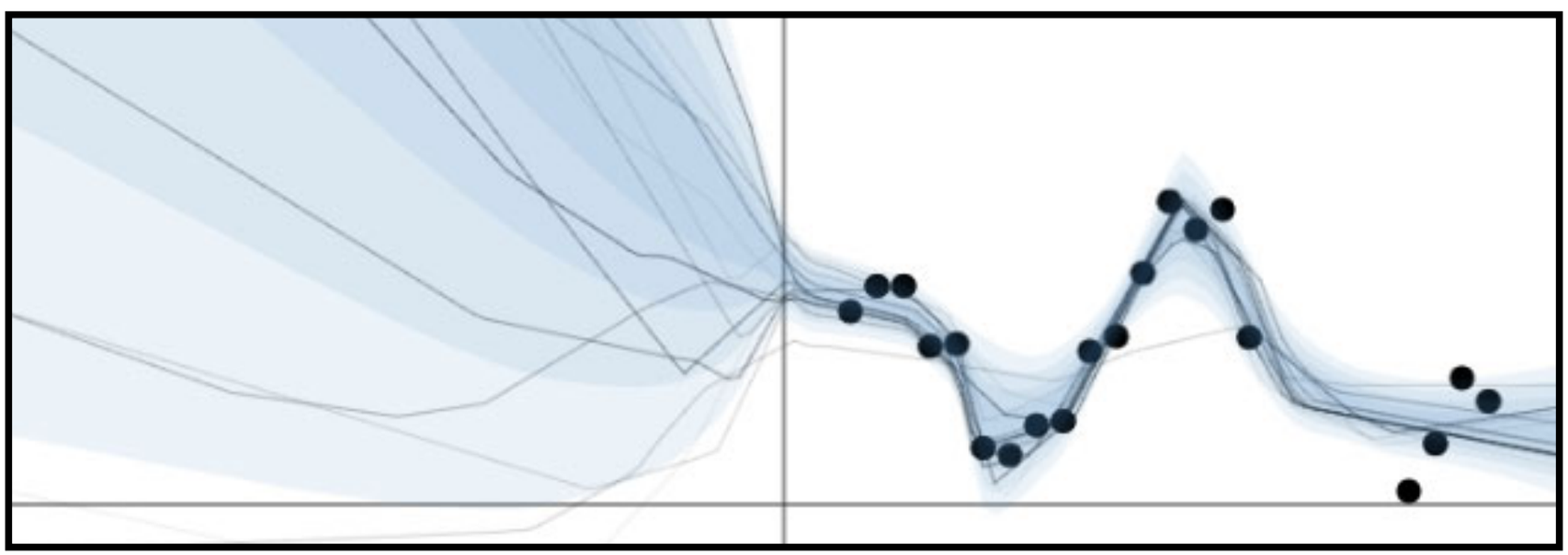
$$p(y^* \mid \mathbf{x}^*, \mathbf{y}, \mathbf{X}) \approx \frac{1}{S} \sum_{s=1}^{S} p(y^* \mid \mathbf{x}^*, \{\hat{\mathbf{\Lambda}}_{l,s}\}_{l=1}^{L+1}), \qquad \hat{\lambda} \sim p(\lambda)$$

Bayesian Interpretation of Dropout

- Gal & Ghahramani [2016] argued that dropout can be interpreted as variational Bayesian inference.

- Obtain predictive uncertainty by averaging over noise samples.

$$p(y^* \mid \mathbf{x}^*, \mathbf{y}, \mathbf{X}) \approx \frac{1}{S} \sum_{s=1}^{S} p(y^* \mid \mathbf{x}^*, \{\hat{\mathbf{\Lambda}}_{l,s}\}_{l=1}^{L+1}), \qquad \hat{\lambda} \sim p(\lambda)$$



Bayesian Interpretation of Dropout

- Gal & Ghahramani [2016] argued that dropout can be interpreted as variational Bayesian inference.

- Obtain predictive uncertainty by averaging over noise samples.

- However, p(λ)—their posterior approximation—**is fixed**, **does not depend on data**.

Bayesian Interpretation of Dropout

- Gal & Ghahramani [2016] argued that dropout can be interpreted as variational Bayesian inference.

- Obtain predictive uncertainty by averaging over noise samples.

- However, p(λ)—their posterior approximation—**is fixed**, **does not depend on data**.

- Subsequent work has attempted to fix this by optimizing the noise distribution. [Maeda, 2014; Kingma et al., 2015; Molchanov et al., 2017; Gal et al., 2017]

The Astrophysical Journal, 859:64 (16pp), 2018 May 20 https://doi.org/10.3847/1538-4357/aabfdb

Detecting Solar-like Oscillations in Red Giants with Deep Learning

Marc Hon[1], Dennis Stello[1,2,3], and Joel C. Zinn[4]
[1] School of Physics, The University of New South Wales, Sydney NSW 2052, Australia; mtyh555@uowmail.edu.au
[2] Sydney Institute for Astronomy (SIfA), School of Physics, University of Sydney, NSW 2006, Australia
[3] Stellar Astrophysics Centre, Department of Physics and Astronomy, Aarhus University, Ny Munkegade 120, DK-8000 Aarhus C, Denmark
[4] Department of Astronomy, The Ohio State University, Columbus, OH 43210, USA
Received 2018 March 12; revised 2018 April 18; accepted 2018 April 19; published 2018 May 24

The Astrophysical Journal, 859:64 (16pp), 2018 May 20 https://doi.org/10.3847/1538-4357/aabfdb

Detecting Solar-like Oscillations in Red Giants with Deep Learning

Marc Hon[1], Dennis Stello[1,2,3], and Joel C. Zinn[4]
[1] School of Physics, The University of New South Wales, Sydney NSW 2052, Australia; mtyh555@uowmail.edu.au
[2] Sydney Institute for Astronomy (SIfA), School of Physics, University of Sydney, NSW 2006, Australia
[3] Stellar Astrophysics Centre, Department of Physics and Astronomy, Aarhus University, Ny Munkegade 120, DK-8000 Aarhus C, Denmark
[4] Department of Astronomy, The Ohio State University, Columbus, OH 43210, USA
Received 2018 March 12; revised 2018 April 18; accepted 2018 April 19; published 2018 May 24

"…[dropout] is effectively a Monte Carlo integration over a Gaussian process posterior approximation (Gal & Ghahramani, 2016)."

~ Hon et al. (2018)

Understanding Dropout: Goals

Understanding Dropout: Goals

- ⊗ **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

Understanding Dropout: Goals

- **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

- **Why drop hidden units?:** dropping weights ('DropConnect') was explored by Wan et al. [2013] but has not found wide use.

Understanding Dropout: Goals

- **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

- **Why drop hidden units?:** dropping weights ('DropConnect') was explored by Wan et al. [2013] but has not found wide use.

- Principles for extension to new architectures?

Understanding Dropout: Goals

- **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

- **Why drop hidden units?:** dropping weights ('DropConnect') was explored by Wan et al. [2013] but has not found wide use.

- Principles for extension to new architectures?

Dropout as a Bayesian Prior

- ⊗ Consider a scalar-version of the dropout / mult. noise equation:

$$h_l = f_l(h_{l-1}\lambda_l w_l), \quad \lambda_l \sim p(\lambda)$$

Dropout as a Bayesian Prior

- Consider a scalar-version of the dropout / mult. noise equation:

$$h_l = f_l(h_{l-1}\lambda_l w_l), \quad \lambda_l \sim p(\lambda)$$

- Assume a Gaussian prior on the weight:

$$w_l \sim \mathbf{N}(0, \sigma_0^2)$$

Dropout as a Bayesian Prior

- Consider a scalar-version of the dropout / mult. noise equation:

$$h_l = f_l(h_{l-1}\lambda_l w_l), \quad \lambda_l \sim p(\lambda)$$

- Assume a Gaussian prior on the weight:

$$w_l \sim \mathbf{N}(0, \sigma_0^2)$$

- What is the distribution of the noise-weight product?

$$h_l = f_l(h_{l-1}\underbrace{\lambda_l w_l})$$

Distribution of this product?

Dropout as a Bayesian Prior

- Consider a scalar-version of the dropout / mult. noise equation:

$$h_l = f_l(h_{l-1}\lambda_l w_l), \quad \lambda_l \sim p(\lambda)$$

- Assume a Gaussian prior on the weight:

$$w_l \sim \mathbf{N}(0, \sigma_0^2)$$

- What is the distribution of the noise-weight product?

$$h_l = f_l(h_{l-1}\underbrace{\lambda_l w_l})$$

Gaussian scale mixture! (expanded param.)

[Beale & Mallows, 1959]

Dropout as a Bayesian Prior

- Gaussian scale mixtures can be reparameterized into the following hierarchical form:

$$h_l = f_l(h_{l-1}\lambda_l w_l), \quad w_l \sim \mathbf{N}(0,\sigma_0^2), \quad \lambda_l \sim p(\lambda)$$

Dropout as a Bayesian Prior

- Gaussian scale mixtures can be reparameterized into the following hierarchical form:

$$h_l = f_l(h_{l-1}\lambda_l w_l), \quad w_l \sim \mathbf{N}(0,\sigma_0^2), \quad \lambda_l \sim p(\lambda)$$

SWITCH TO HIERARCHICAL PARAMETRIZATION

$$h_l = f_l(h_{l-1}\tilde{w}_l), \quad \tilde{w}_l \sim \mathbf{N}(0,\lambda_l^2\sigma_0^2), \quad \lambda_l \sim p(\lambda)$$

Dropout as a Bayesian Prior

- Gaussian scale mixtures can be reparameterized into the following hierarchical form:

$$h_l = f_l(h_{l-1}\lambda_l w_l), \quad w_l \sim \mathbf{N}(0,\sigma_0^2), \quad \lambda_l \sim p(\lambda)$$

↓ SWITCH TO HIERARCHICAL PARAMETRIZATION ↓

$$h_l = f_l(h_{l-1}\tilde{w}_l), \quad \tilde{w}_l \sim \mathbf{N}(0,\lambda_l^2\sigma_0^2), \quad \lambda_l \sim p(\lambda)$$

- Noise moves from likelihood and becomes a scale

Dropout as a Bayesian Prior

- Gaussian scale mixtures can be reparameterized into the following hierarchical form:

$$h_l = f_l(h_{l-1}\lambda_l w_l), \quad w_l \sim \mathbf{N}(0,\sigma_0^2), \quad \lambda_l \sim p(\lambda)$$

↓ SWITCH TO HIERARCHICAL PARAMETRIZATION ↓

$$h_l = f_l(h_{l-1}\tilde{w}_l), \quad \tilde{w}_l \sim \mathbf{N}(0,\lambda_l^2\sigma_0^2), \quad \lambda_l \sim p(\lambda)$$

- Noise moves from likelihood and becomes a scale

- Expanded parameterization has been previously used for Bayesian variable selection [Kuo & Mallik, 1998]:

Bernoulli

$$y_i = \sum_{j=1}^{p} \beta_j \gamma_j x_{ij} + \epsilon_i$$

Dropout as a Bayesian Prior

- ⊗ Reparameterization translates between noise distributions and marginal priors:

Noise Model $p(\lambda)$	Variance Prior $p(\lambda^2)$	Marginal Prior $p(\widetilde{w})$
Bernoulli	Bernoulli	Spike-and-Slab
Gaussian	χ^2	Generalized Hyperbolic
Rayleigh	Exponential	Laplace
Inverse Nakagami	Γ^{-1}	Student-t
Half-Cauchy	Unnamed	Horseshoe

The Astrophysical Journal, 859:64 (16pp), 2018 May 20 https://doi.org/10.3847/1538-4357/aabfdb

Detecting Solar-like Oscillations in Red Giants with Deep Learning

Marc Hon[1], Dennis Stello[1,2,3], and Joel C. Zinn[4]
[1] School of Physics, The University of New South Wales, Sydney NSW 2052, Australia; mtyh555@uowmail.edu.au
[2] Sydney Institute for Astronomy (SIfA), School of Physics, University of Sydney, NSW 2006, Australia
[3] Stellar Astrophysics Centre, Department of Physics and Astronomy, Aarhus University, Ny Munkegade 120, DK-8000 Aarhus C, Denmark
[4] Department of Astronomy, The Ohio State University, Columbus, OH 43210, USA
Received 2018 March 12; revised 2018 April 18; accepted 2018 April 19; published 2018 May 24

"…[dropout] is effectively a Monte Carlo integration over a Gaussian process posterior approximation (Gal & Ghahramani, 2016)."

~ Hon et al. (2018)

The Astrophysical Journal, 859:64 (16pp), 2018 May 20 https://doi.org/10.3847/1538-4357/aabfdb

Detecting Solar-like Oscillations in Red Giants with Deep Learning

Marc Hon[1], Dennis Stello[1,2,3], and Joel C. Zinn[4]
[1] School of Physics, The University of New South Wales, Sydney NSW 2052, Australia; mtyh555@uowmail.edu.au
[2] Sydney Institute for Astronomy (SIfA), School of Physics, University of Sydney, NSW 2006, Australia
[3] Stellar Astrophysics Centre, Department of Physics and Astronomy, Aarhus University, Ny Munkegade 120, DK-8000 Aarhus C, Denmark
[4] Department of Astronomy, The Ohio State University, Columbus, OH 43210, USA
Received 2018 March 12; revised 2018 April 18; accepted 2018 April 19; published 2018 May 24

"…[dropout] is effectively a Monte Carlo integration over a scale prior. [redacted] "

~ Hon et al. (2018)

Understanding Dropout: Goals

- **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

- **Why drop hidden units?:** dropping weights ('DropConnect') was explored by Wan et al. [2013] but has not found wide use.

- Principles for extension to new architectures?

Understanding Dropout: Goals

- **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

Dropout is a scale *prior*, not a posterior.

- **Why drop hidden units?:** dropping weights ('DropConnect') was explored by Wan et al. [2013] but has not found wide use.

- **Principles for extension to new architectures?**

Understanding Dropout: Goals

- **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

Dropout is a scale *prior*, not a posterior.

- **Why drop hidden units?:** dropping weights ('DropConnect') was explored by Wan et al. [2013] but has not found wide use.

- **Principles for extension to new architectures?**

Dropout as Structured Shrinkage

- Now consider the multivariate case:

Dropout as Structured Shrinkage

- Now consider the multivariate case:

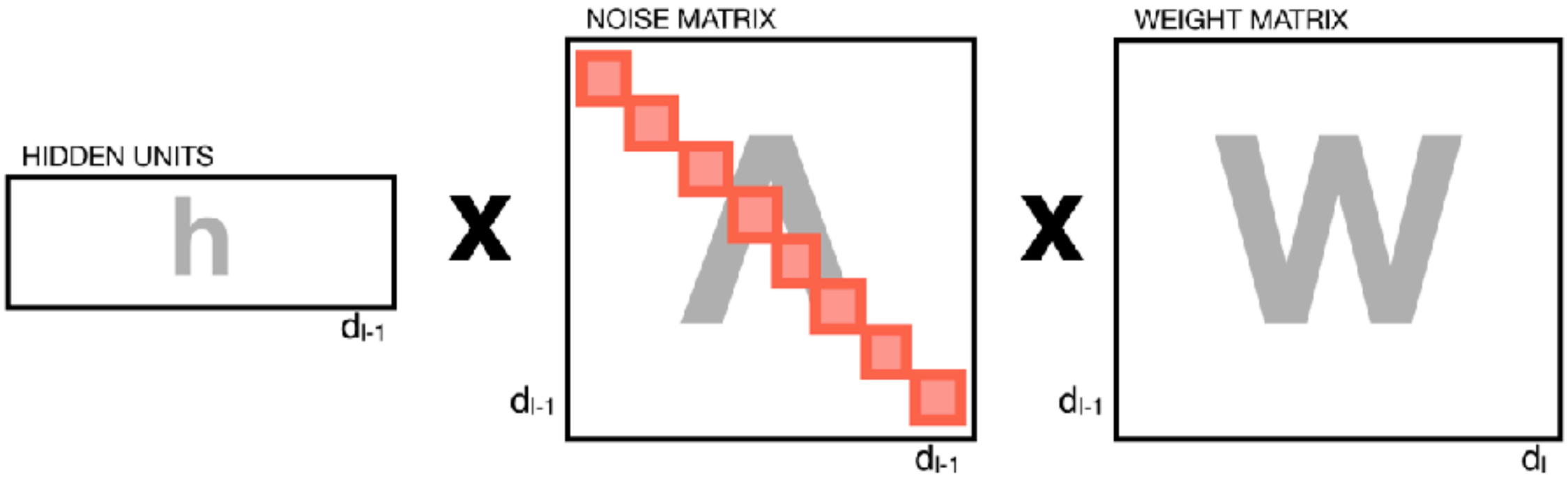
$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\mathbf{\Lambda}_l\mathbf{W}_l) \qquad w_{l,i,j} \sim \mathbf{N}(0,\sigma_0^2)$$

$$\lambda_{l,i,i} \sim p(\lambda)$$

Dropout as Structured Shrinkage

⊗ Now consider the multivariate case:

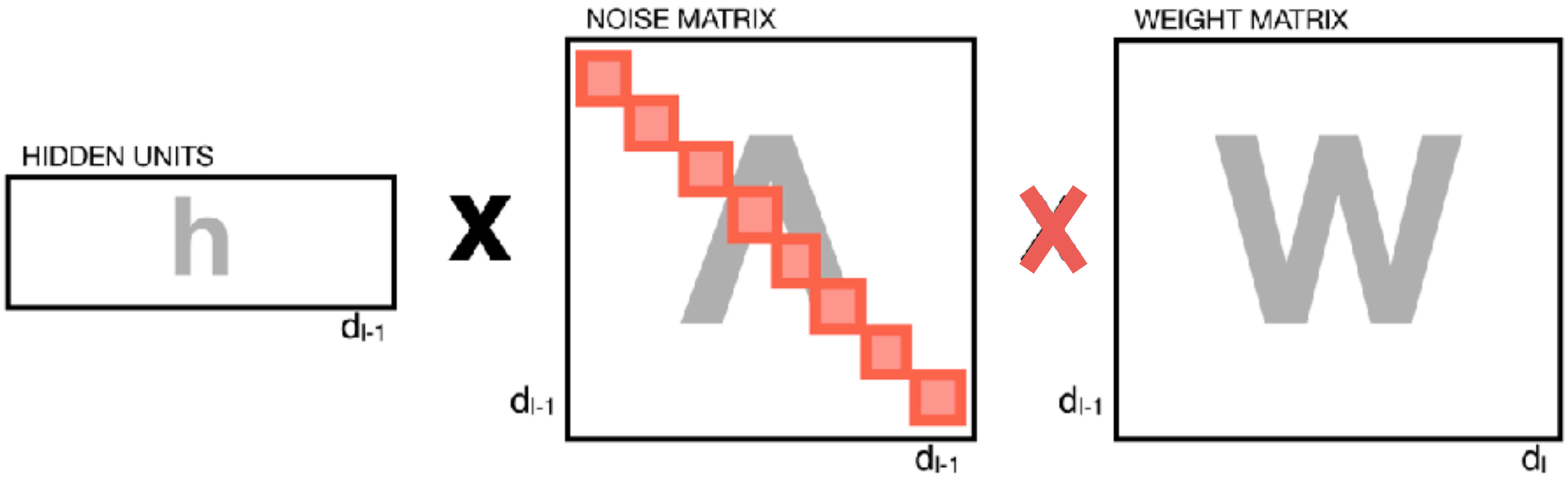
$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\mathbf{\Lambda}_l\mathbf{W}_l) \qquad w_{l,i,j} \sim \mathbf{N}(0,\sigma_0^2) \qquad \lambda_{l,i,i} \sim p(\lambda)$$



Dropout as Structured Shrinkage

⊗ Now consider the multivariate case:

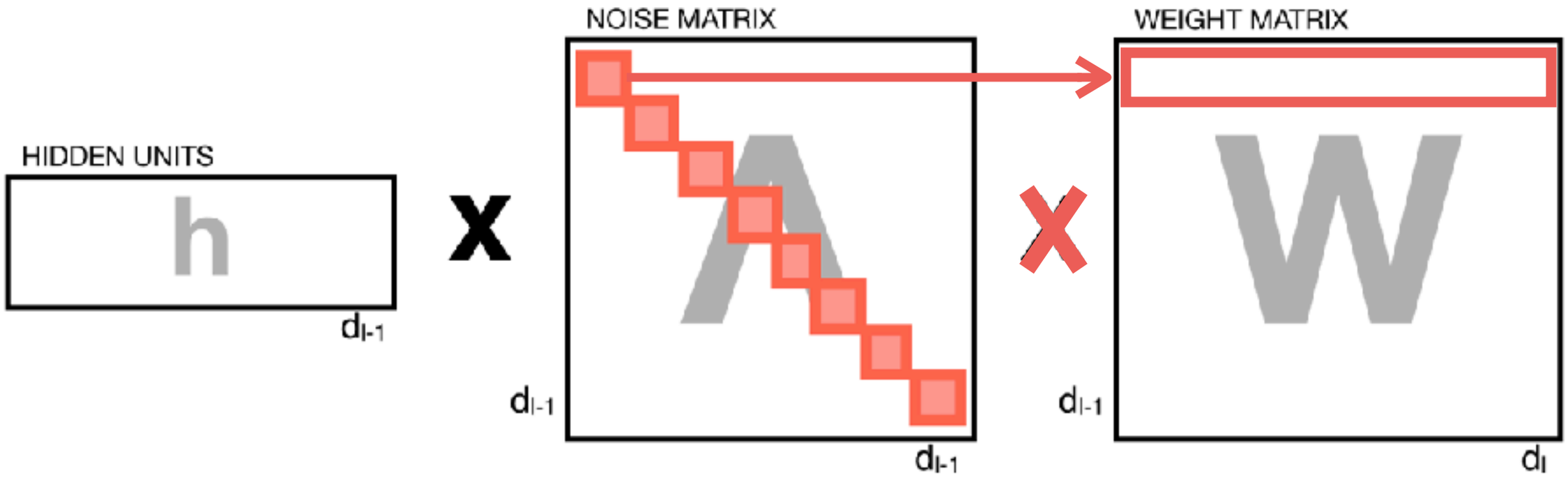
$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\mathbf{\Lambda}_l\mathbf{W}_l) \qquad w_{l,i,j} \sim \mathbf{N}(0,\sigma_0^2) \qquad \lambda_{l,i,i} \sim p(\lambda)$$



Dropout as Structured Shrinkage

⊗ Now consider the multivariate case:

$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\mathbf{\Lambda}_l\mathbf{W}_l) \qquad w_{l,i,j} \sim \mathbf{N}(0,\sigma_0^2) \qquad \lambda_{l,i,i} \sim p(\lambda)$$



Dropout as Structured Shrinkage

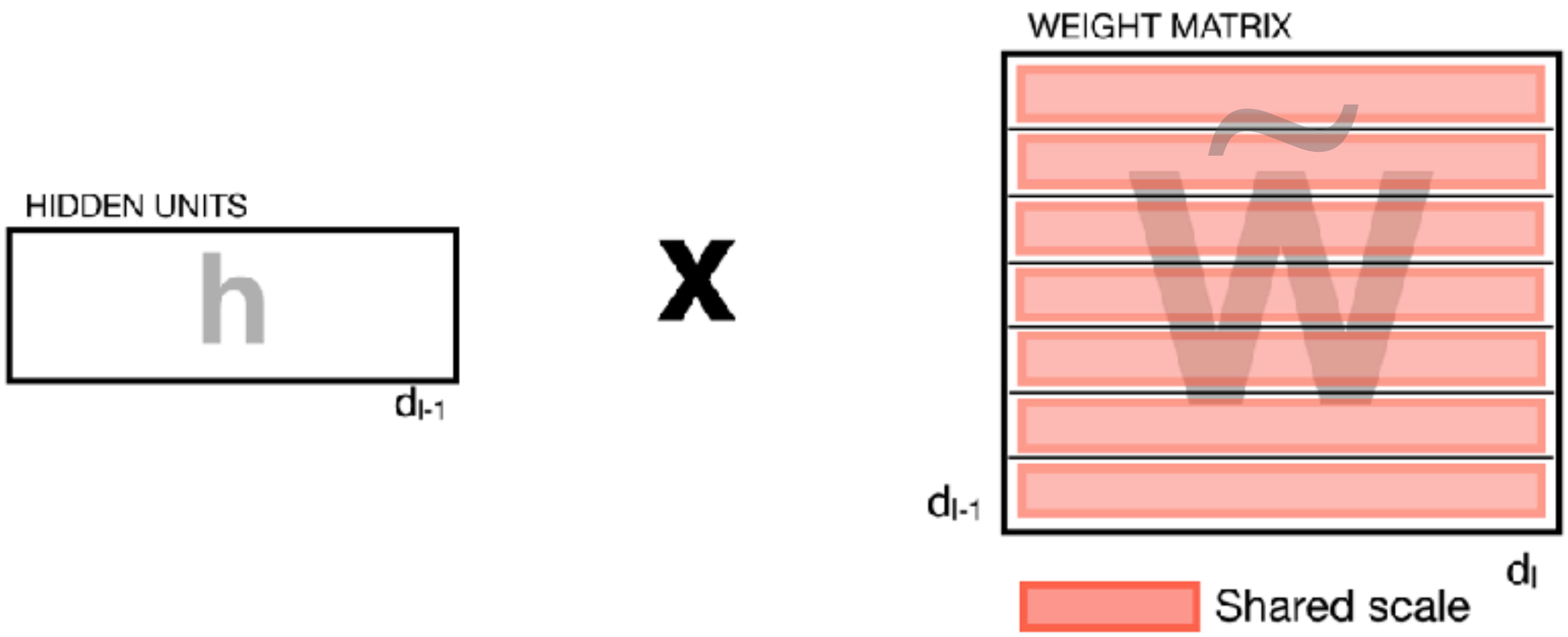
- Reparameterized form:

$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\tilde{\mathbf{W}}_l)$$

$$\tilde{w}_{l,i,j} \sim \mathbf{N}(0, \lambda^2_{l,i,i}\sigma_0^2)$$

$$\lambda_{l,i,i} \sim p(\lambda)$$

i indexes rows



Dropout as Structured Shrinkage

- Reparameterized form:

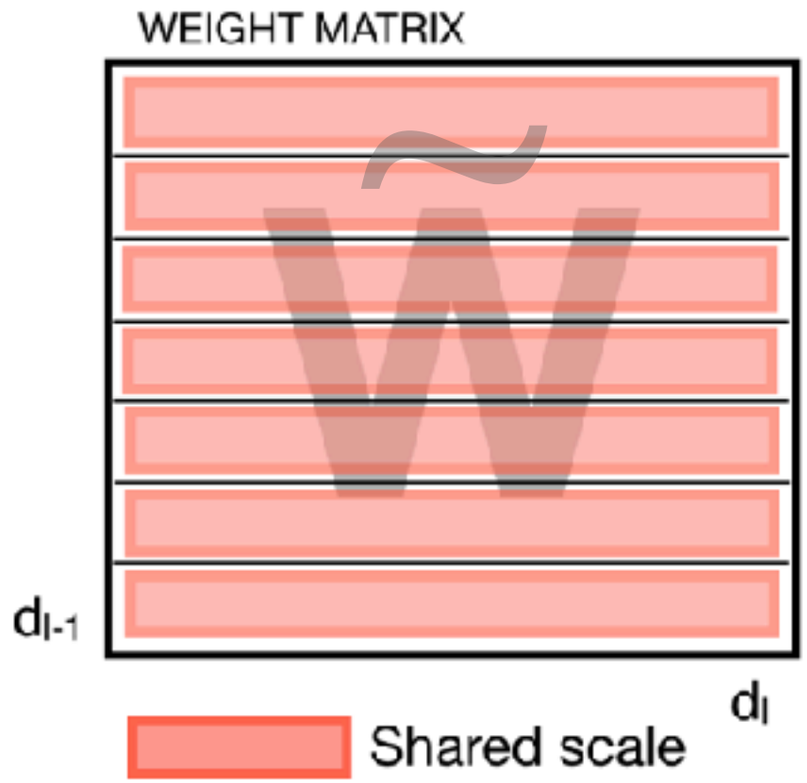
$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\tilde{\mathbf{W}}_l)$$

$$\tilde{w}_{l,i,j} \sim \mathbf{N}(0, \lambda^2_{l,i,i}\sigma_0^2)$$

$$\lambda_{l,i,i} \sim p(\lambda)$$

i indexes rows

Uses Bayesian shrinkage to control the effective number of hidden units.



Dropout as Structured Shrinkage

⊗ Reparameterized form:

$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\tilde{\mathbf{W}}_l)$$

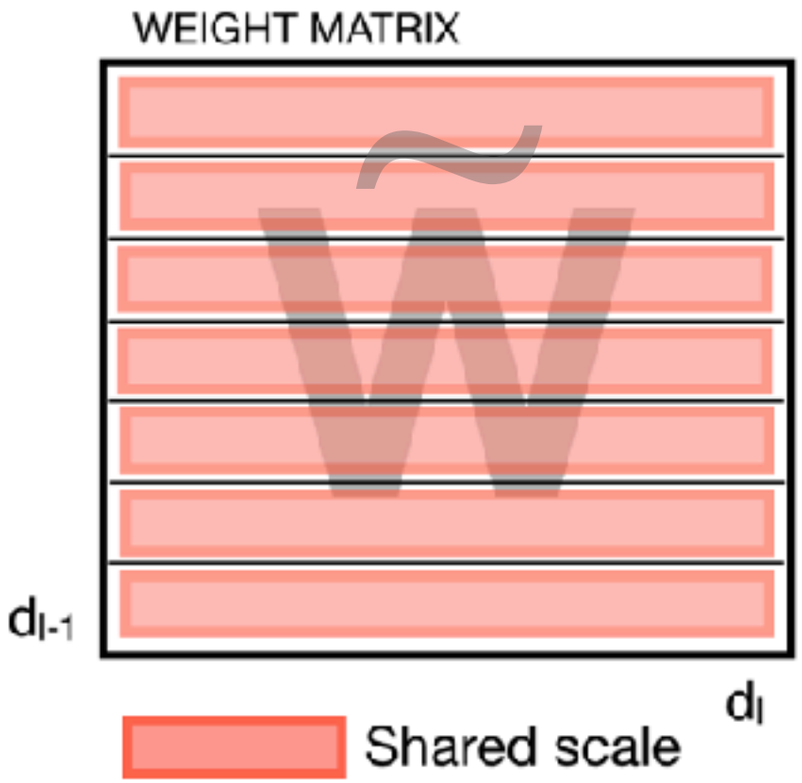
$$\tilde{w}_{l,i,j} \sim \mathbf{N}(0, \lambda^2_{l,i,i}\sigma_0^2)$$

$$\lambda_{l,i,i} \sim p(\lambda)$$

i indexes rows

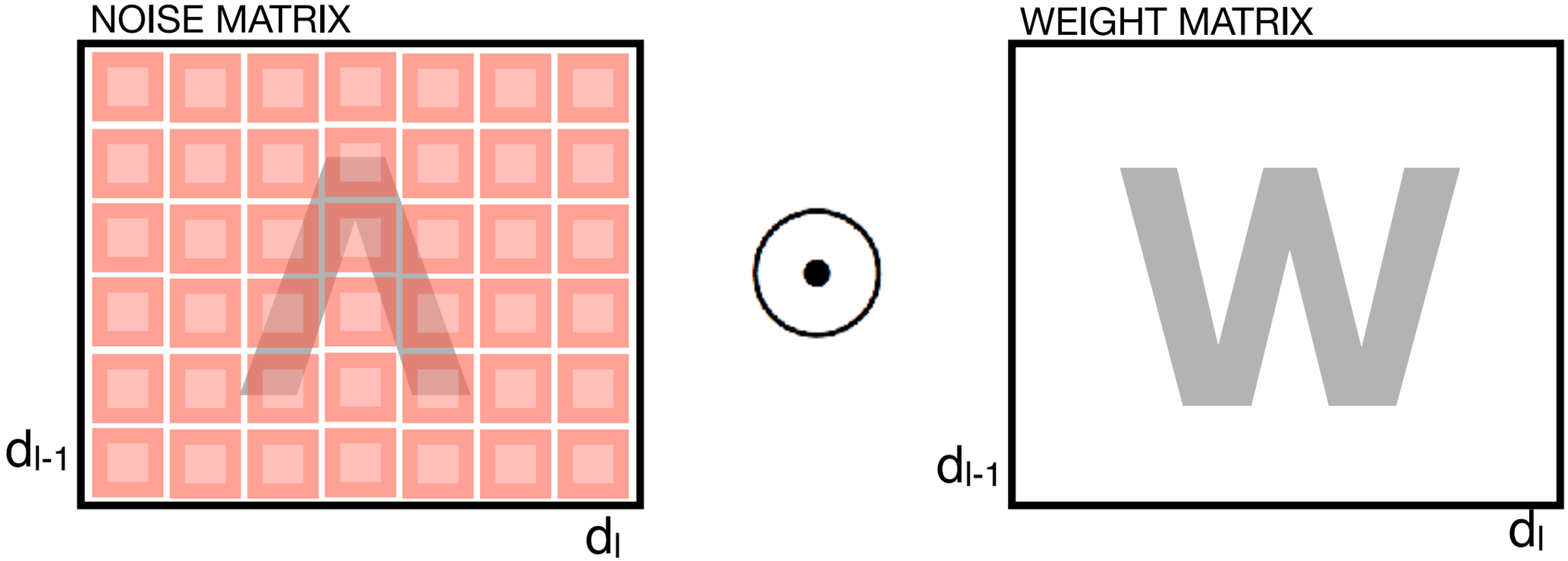
Uses Bayesian shrinkage to control the effective number of hidden units.

Same structure as the **automatic relevance determination (ARD)** prior proposed by D. MacKay and R. Neal for Bayesian NNs [1994].



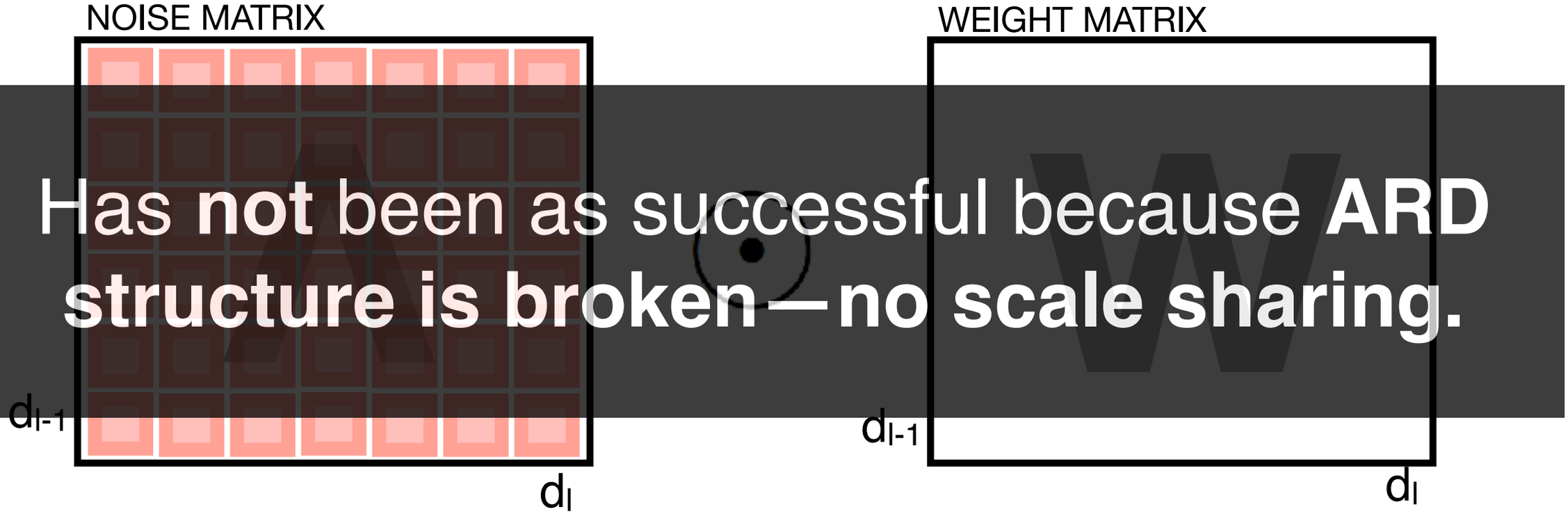
What about DropConnect?

Previous work by Wan et al. [2013] proposed ***dropconnect***, which drops weights independently. Motivated by "co-adaptation" explanation of Hinton et al. [2012]



What about DropConnect?

Previous work by Wan et al. [2013] proposed ***dropconnect***, which drops weights independently. Motivated by "co-adaptation" explanation of Hinton et al. [2012]



Has **not** been as successful because **ARD structure is broken—no scale sharing.**

Understanding Dropout: Goals

- **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

Dropout is a scale *prior*, not a posterior

- **Why drop hidden units?:** dropping weights ('DropConnect') was explored by Wan et al. [2013] but has not found wide use.

- **Principles for extension to new architectures?**

Understanding Dropout: Goals

- **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

Dropout is a scale *prior*, not a posterior

- **Why drop hidden units?:** dropping weights ('DropConnect') was explored by Wan et al. [2013] but has not found wide use.

Induces ARD structure to control effective width

- Principles for extension to new architectures?

Understanding Dropout: Goals

- **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

Dropout is a scale *prior*, not a posterior

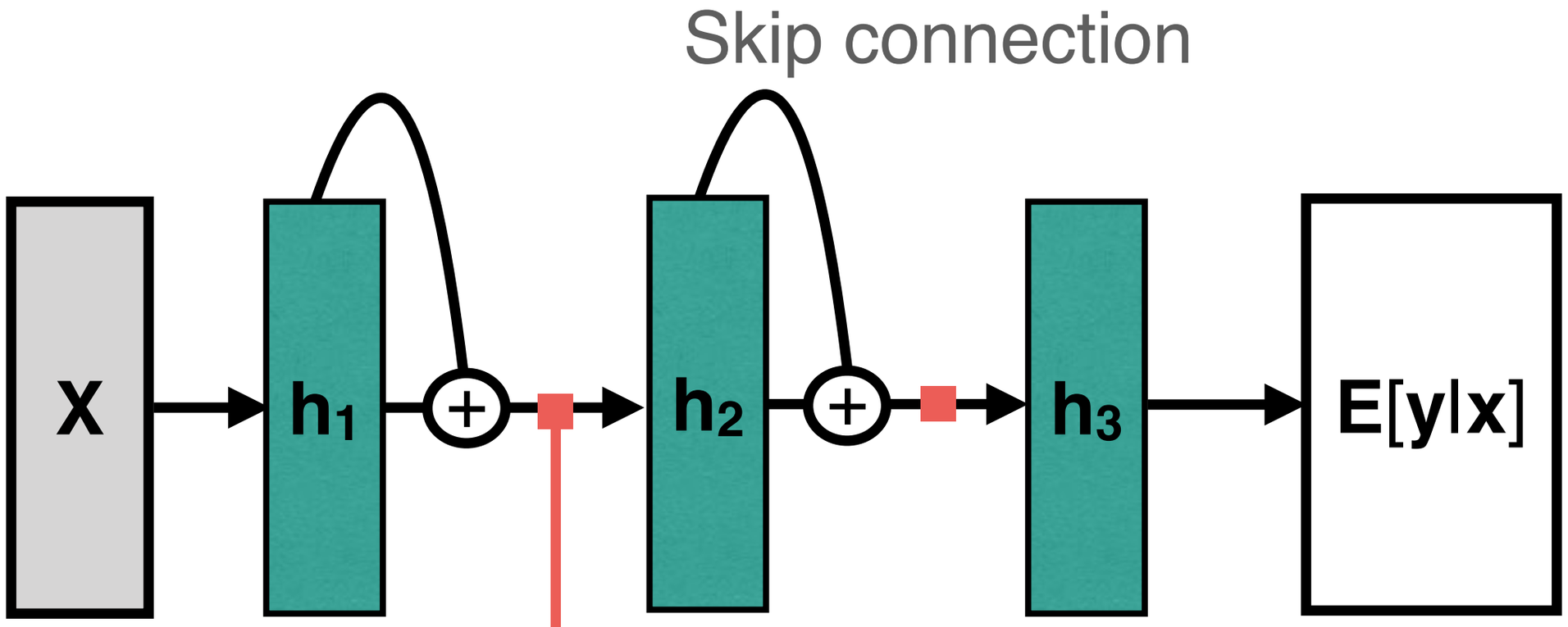
- **Why drop hidden units?:** dropping weights ('DropConnect') was explored by Wan et al. [2013] but has not found wide use.

Induces ARD structure to control effective width

- Principles for extension to new architectures?

Residual Networks and Skip Connections

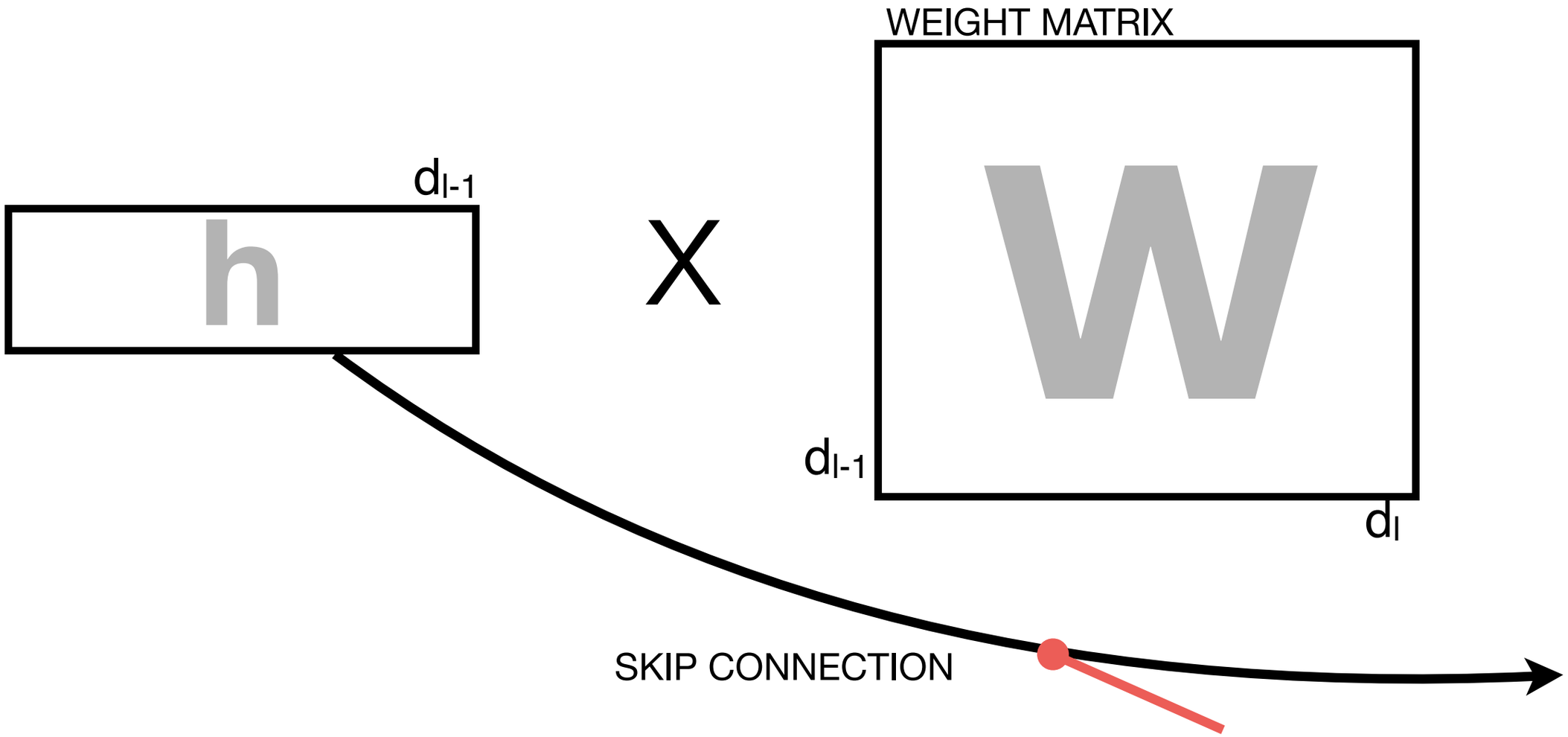
Now what about other architectural modifications?
Say, skip connections?



$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\mathbf{W}_l) + \mathbf{h}_{l-1}$$

Residual Networks and Skip Connections

Now what about other architectural modifications?
Say, skip connections?

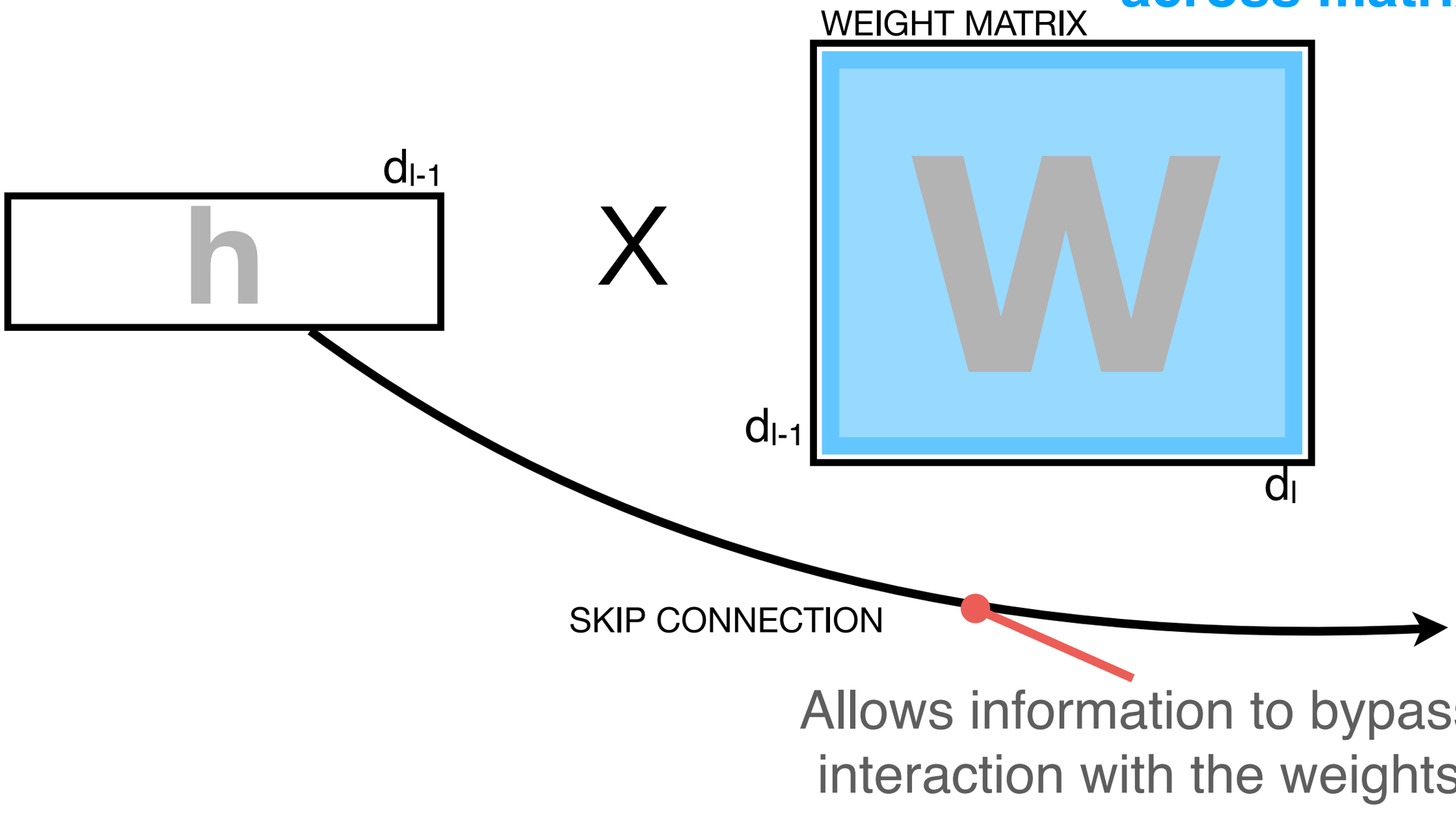


Allows information to bypass interaction with the weights

Residual Networks and Skip Connections

Now what about other architectural modifications?
Say, skip connections?

Scale shared across matrix

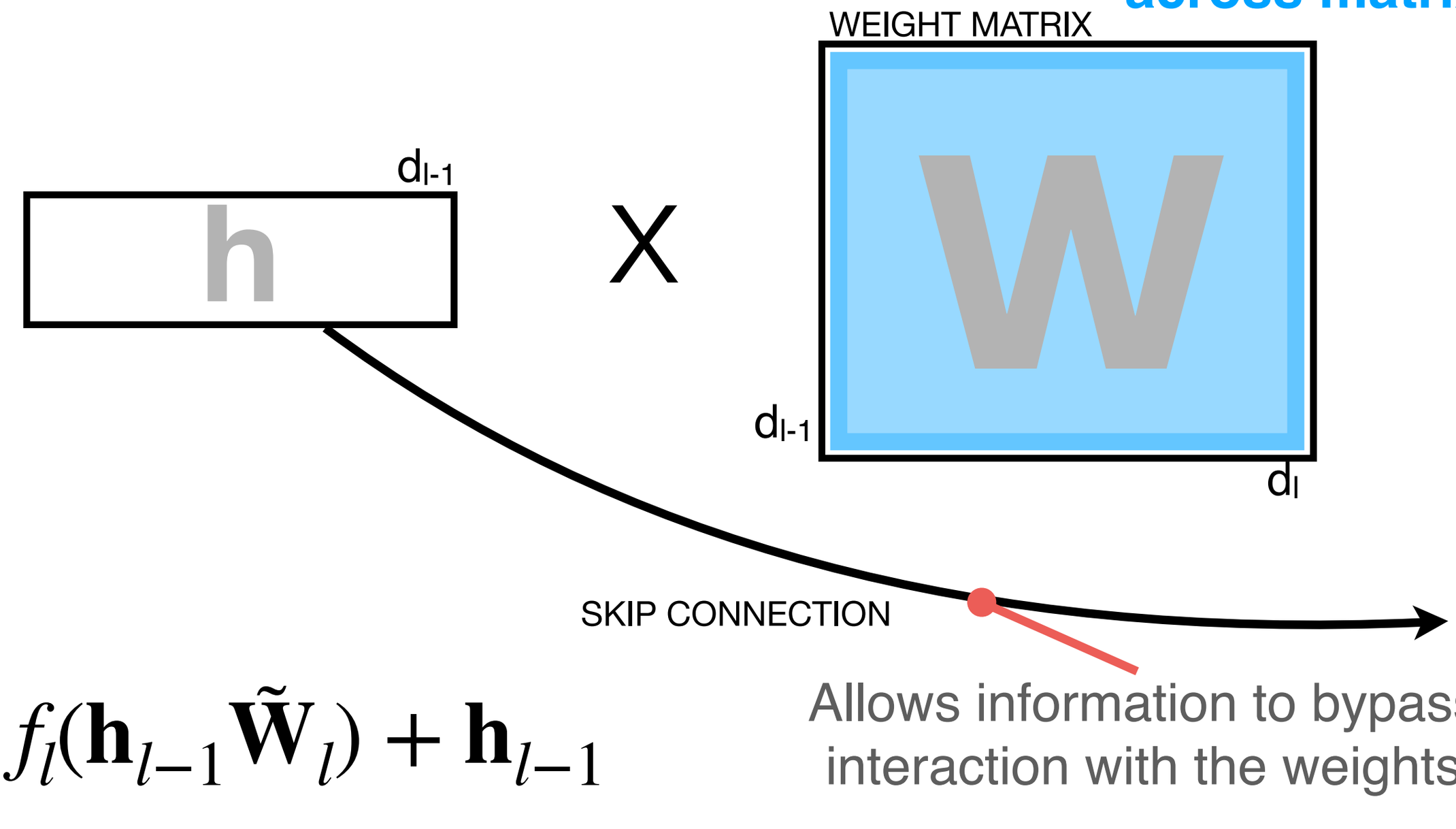


Allows information to bypass interaction with the weights

Residual Networks and Skip Connections

Now what about other architectural modifications?
Say, skip connections?

Scale shared across matrix



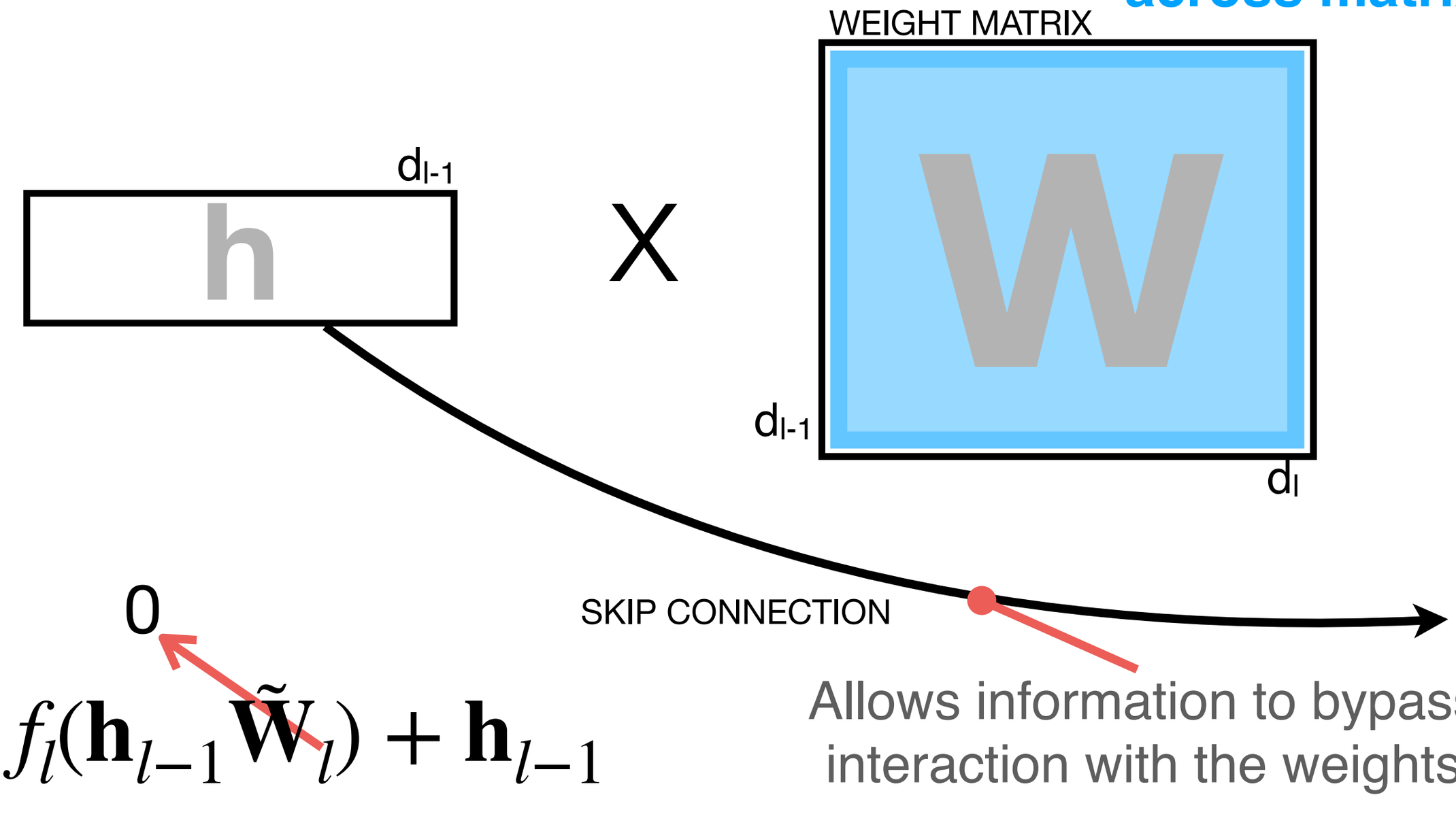
$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\tilde{\mathbf{W}}_l) + \mathbf{h}_{l-1}$$

Allows information to bypass interaction with the weights

Residual Networks and Skip Connections

Now what about other architectural modifications?
Say, skip connections?

Scale shared across matrix



$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\tilde{\mathbf{W}}_l) + \mathbf{h}_{l-1}$$

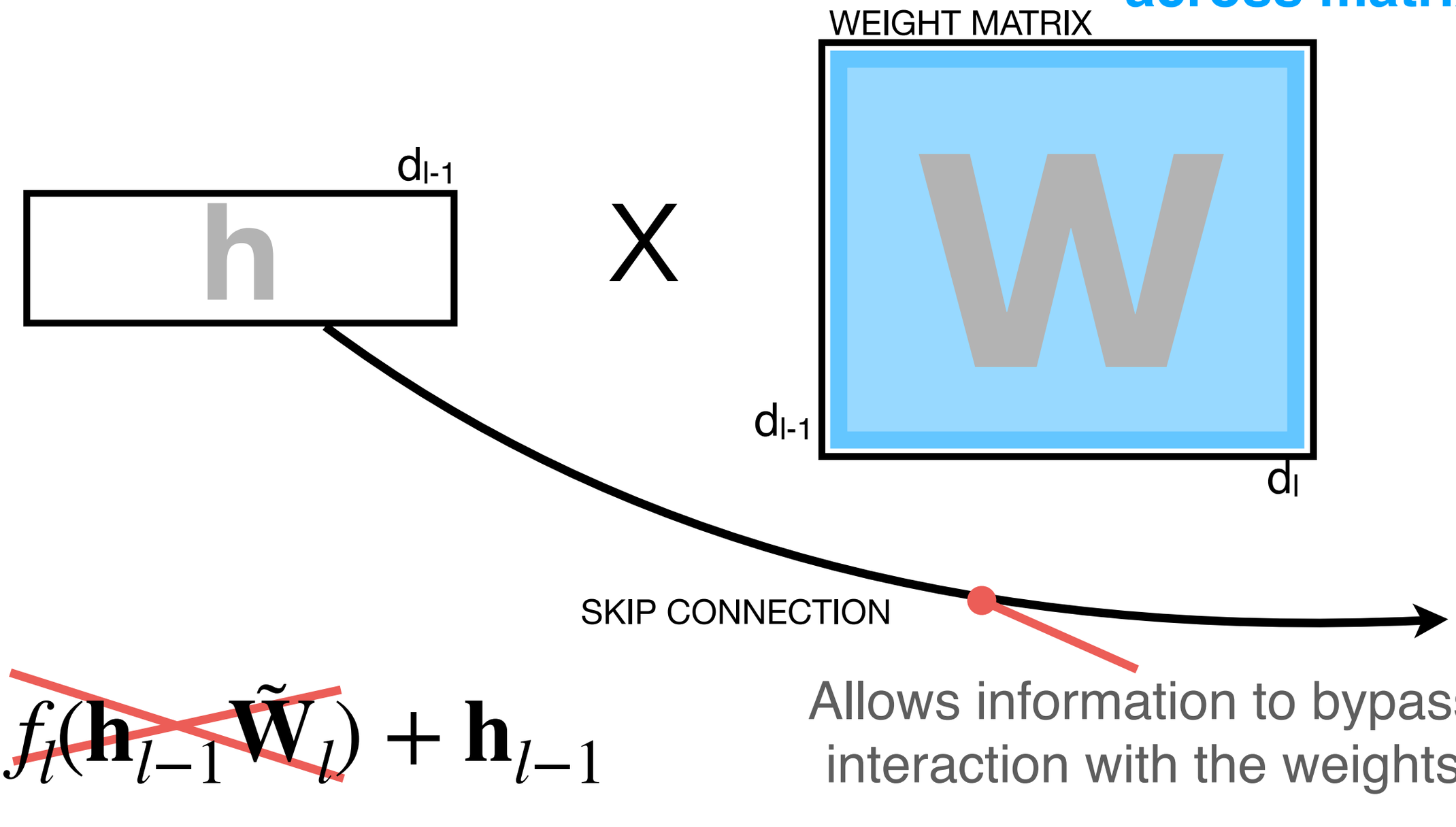
0

Allows information to bypass interaction with the weights

Residual Networks and Skip Connections

Now what about other architectural modifications?
Say, skip connections?

Scale shared across matrix

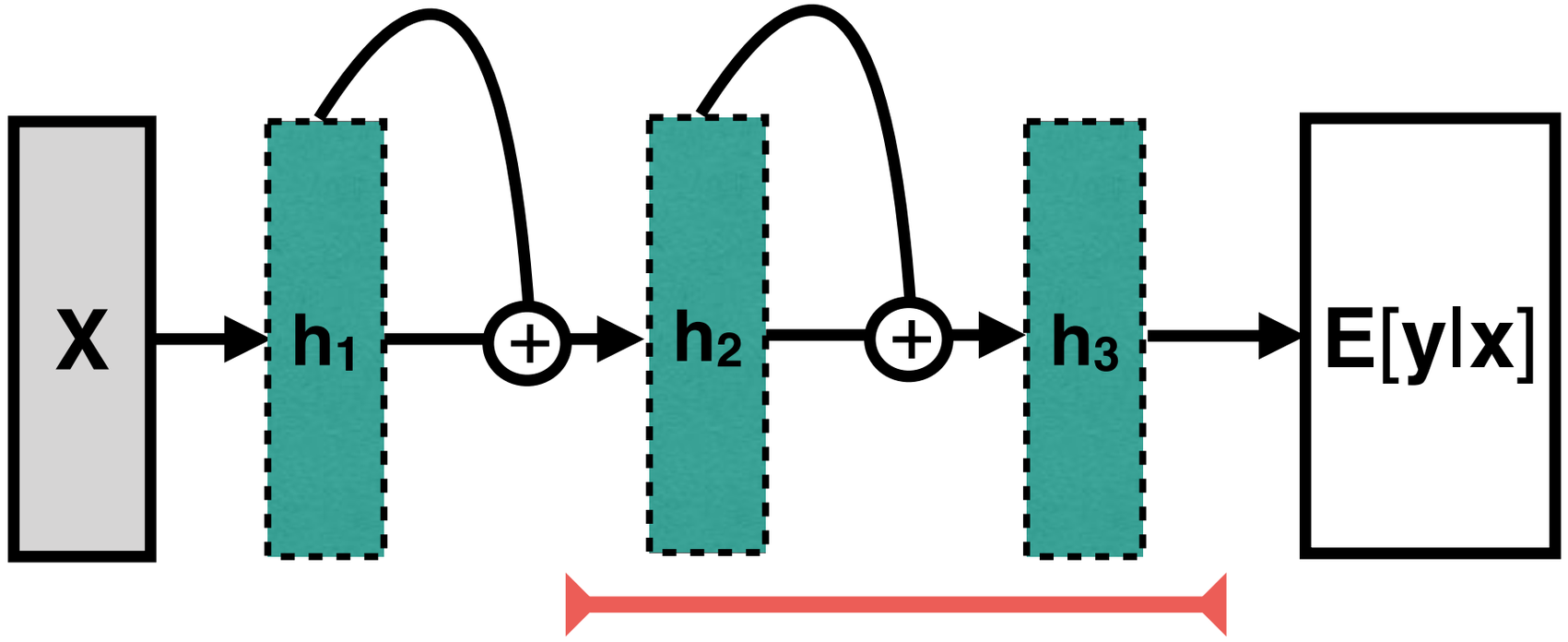


$$\mathbf{h}_l = f_l(\mathbf{h}_{l-1}\tilde{\mathbf{W}}_l) + \mathbf{h}_{l-1}$$

Allows information to bypass interaction with the weights

Automatic Depth Determination

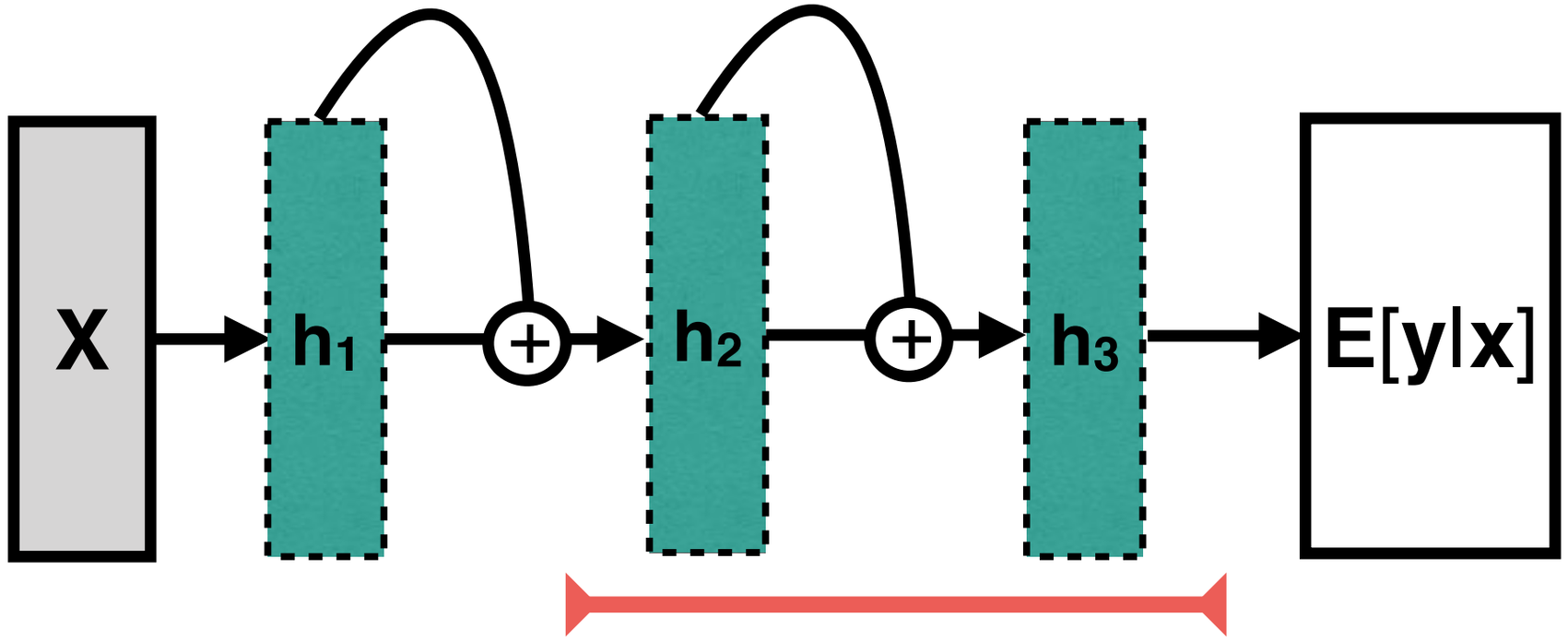
We derive a prior for ResNets called **automatic depth determination** (ADD).



Bayesian shrinkage can control the effective depth of the network

Automatic Depth Determination

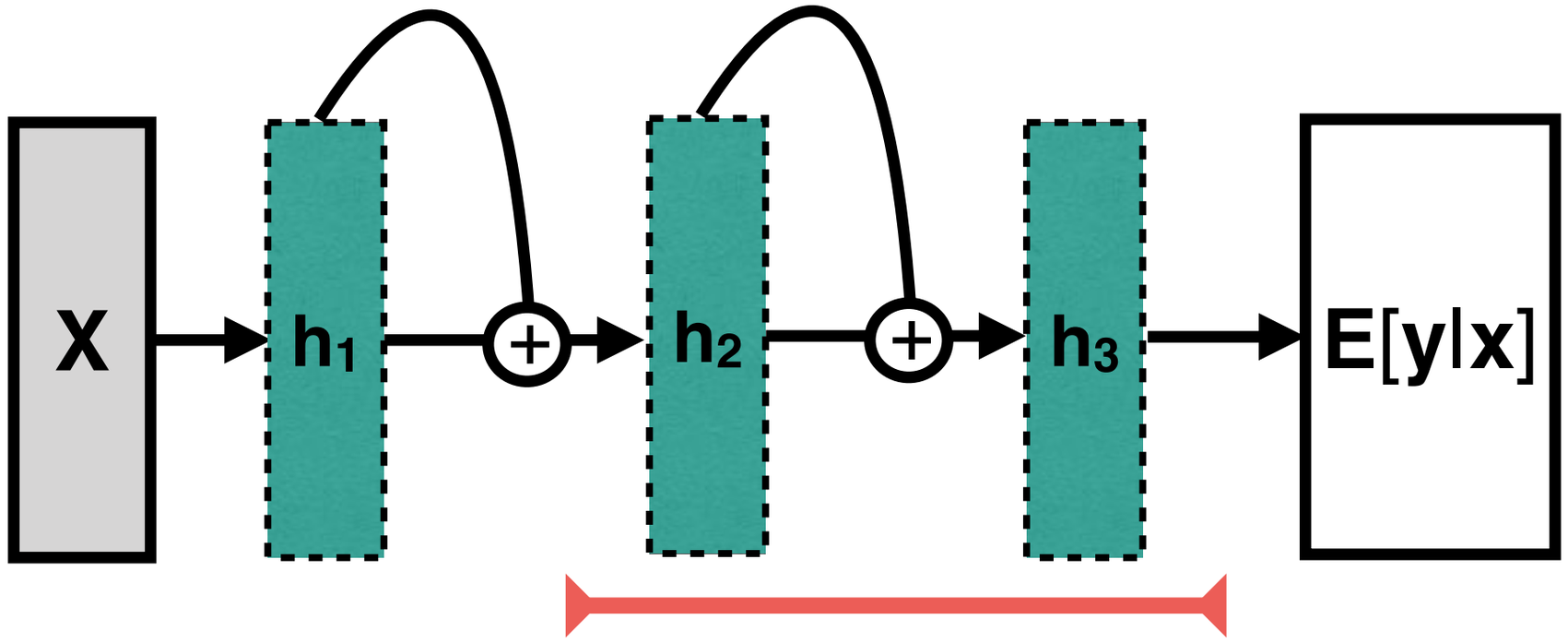
We derive a prior for ResNets called **automatic depth determination** (ADD).



$$\tilde{w}_{l,i,j} \sim \mathbf{N}(0, \tau_l^2 \sigma_0^2) \quad \tau_l \sim p(\tau)$$

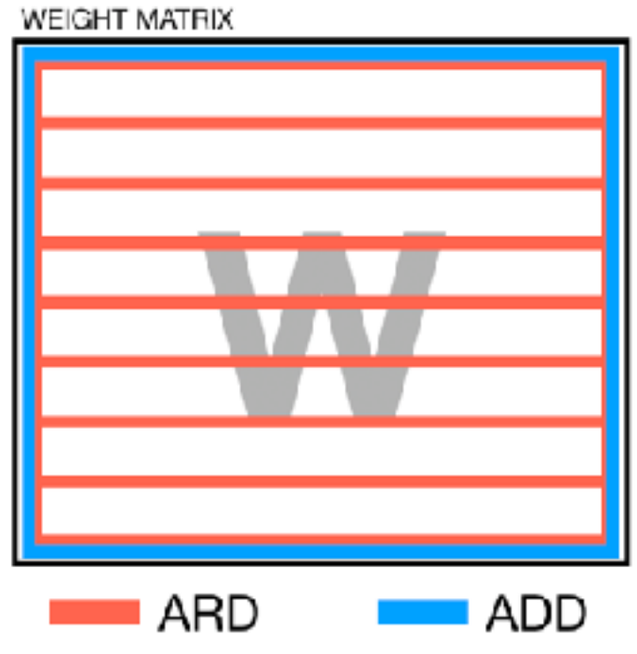
Automatic Depth Determination

We derive a prior for ResNets called **automatic depth determination** (ADD).



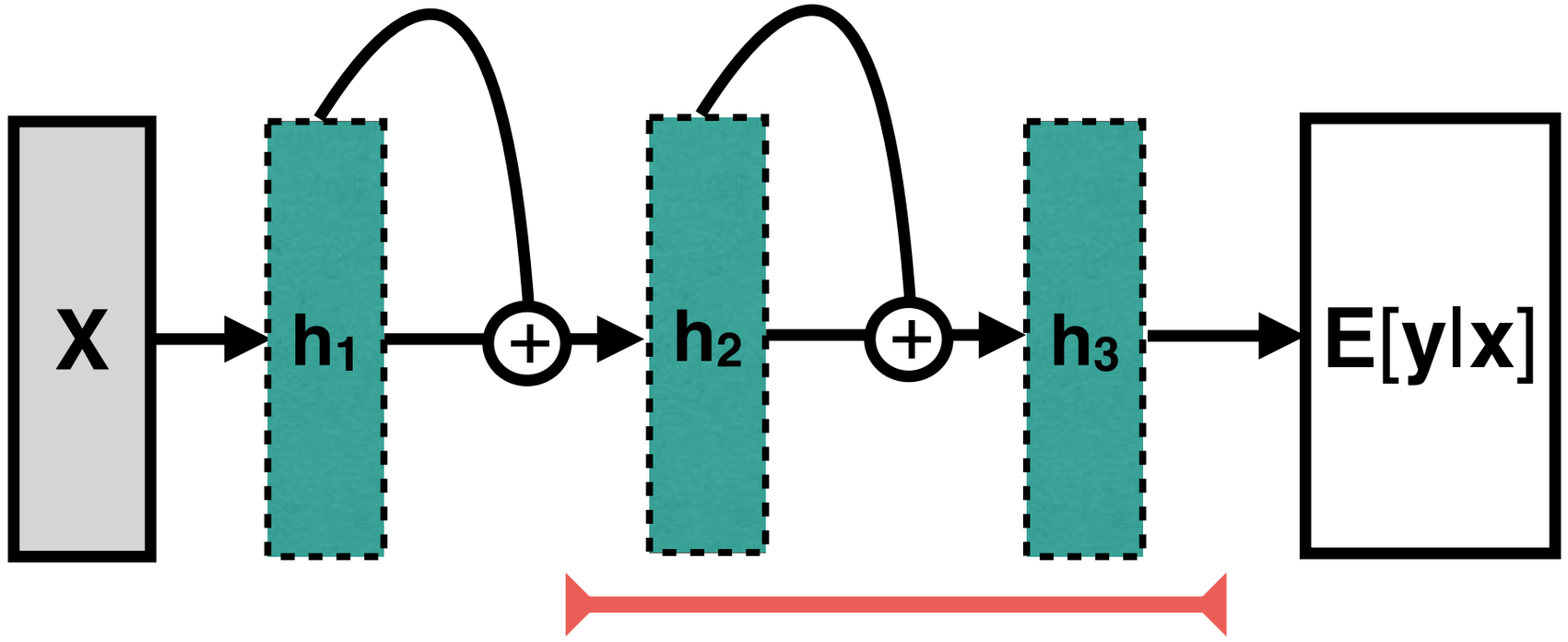
Bayesian shrinkage can control the effective depth of the network

$$\tilde{w}_{l,i,j} \sim \mathbf{N}(0, \tau_l^2 \sigma_0^2) \quad \tau_l \sim p(\tau)$$

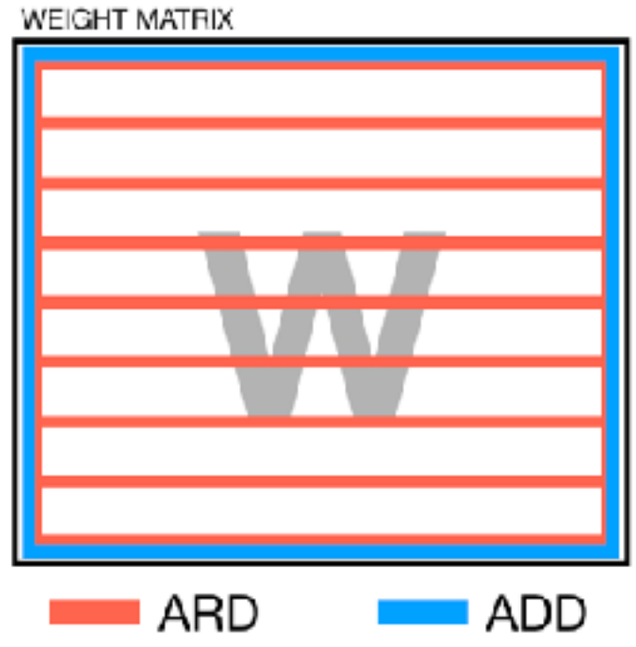


Automatic Depth Determination

We derive a prior for ResNets called **automatic depth determination** (ADD).



Bayesian shrinkage can control the effective depth of the network



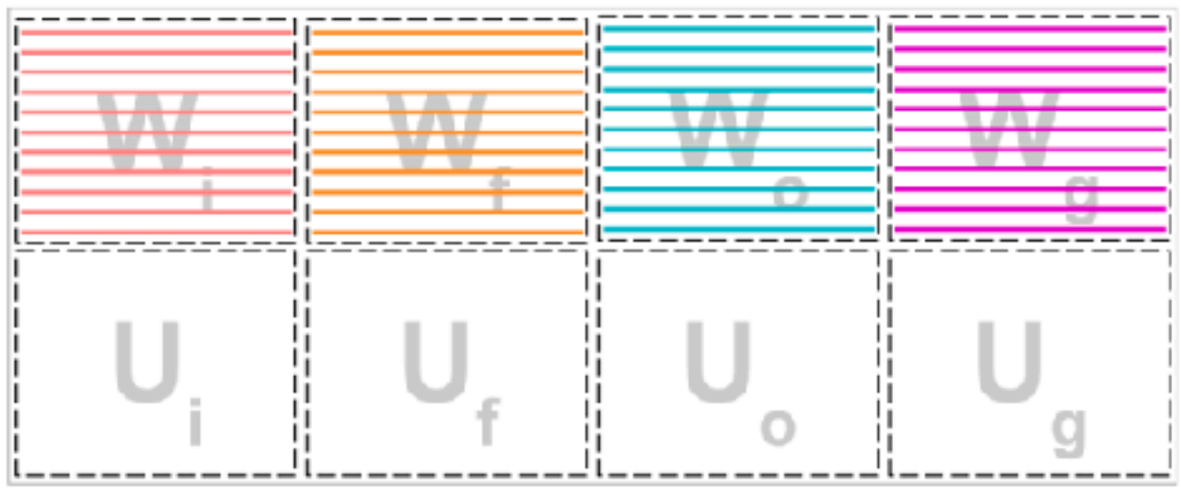
A neural-network-analog of the *global-local shrinkage* prior for robust regression. (See for ref: Polson & Scott [2010])

Extension to More Architectures

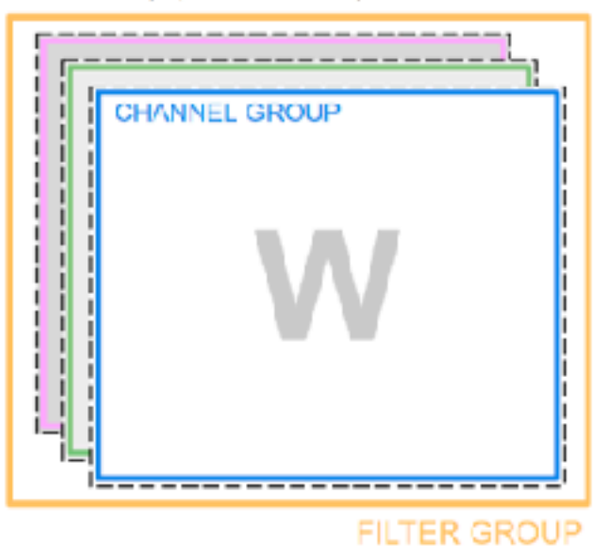
(a) Unstructured (DropConnect)

(b) ARD (Dropout)

(c) ADD (Stochastic Depth)



(d) LSTM ARD



(e) CNN ARD (SpatialDropout)

See ArXiv version (v2) of ICML paper.

Understanding Dropout: Goals

- **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

Dropout is a scale *prior*, not a posterior

- **Why drop hidden units?:** dropping weights ('DropConnect') was explored by Wan et al. [2013] but has not found wide use.

Induces ARD structure to control effective width

- Principles for extension to new architectures?

Understanding Dropout: Goals

- **Revise dropout's Bayesian interpretation:** should be compatible with any inference procedure (not restricted to variational inference).

Dropout is a scale *prior*, not a posterior

- **Why drop hidden units?:** dropping weights ('DropConnect') was explored by Wan et al. [2013] but has not found wide use.

Induces ARD structure to control effective width

- Principles for extension to new architectures?

Automatic depth determination for ResNets

- ⊗ This might help clarify Dropout's assumptions / model definition, but what does this buy us in practice?

- This might help clarify Dropout's assumptions / model definition, but what does this buy us in practice?
- We derived two algorithms that attempt to preserve the simplicity of Dropout:
  - Tail-adaptive importance sampling for marginal-MAP inference
  - Variational EM

- This might help clarify Dropout's assumptions / model definition, but what does this buy us in practice?
- We derived two algorithms that attempt to preserve the simplicity of Dropout:
  - Tail-adaptive importance sampling for marginal-MAP inference
  - Variational EM

Regression

Test Set RMSE

	Dropout	Prob. Backprop	Deep GP	ARD	ADD	ARD-ADD
Boston Concrete Energy Kin8nm Power Wine Yacht	Gal & Ghahramani's results (10 samples train / 500 test)	Variational inference benchmark	Trained via expectation propagation	Trained via variational EM. **Exactly the same hyperparameter settings as Gal & Ghahramani**. Prior chosen via performance on auxiliary data set.		

Regression

Test Set RMSE

	Dropout	Prob. Backprop	Deep GP	ARD	ADD	ARD-ADD
Boston	2.80 ±.13	**2.795** ±.16	2.38 ±.12	**2.158** ±.20	2.343 ±.31	**2.367** ±.18
Concrete	4.50 ±.18	**5.241** ±.12	4.64 ±.11	3.805 ±.28	4.084 ±.34	**3.761** ±.23
Energy	**0.47** ±.01	**0.903** ±.05	0.57 ±.02	**0.852** ±.01	**0.867** ±.11	**0.853** ±.08
Kin8nm	0.08 ±.00	**0.071** ±.00	**0.05** ±.00	0.066 ±.01	0.064 ±.00	0.064 ±.00
Power	3.63 ±.04	**4.028** ±.03	3.60 ±.03	**3.486** ±.10	**3.290** ±.06	**3.236** ±.07
Wine	0.60 ±.01	**0.643** ±.01	**0.50** ±.01	**0.561** ±.03	**0.555** ±.01	**0.538** ±.03
Yacht	0.66 ±.06	**0.848** ±.05	0.98 ±.09	**0.691** ±.12	**0.657** ±.14	**0.604** ±.16

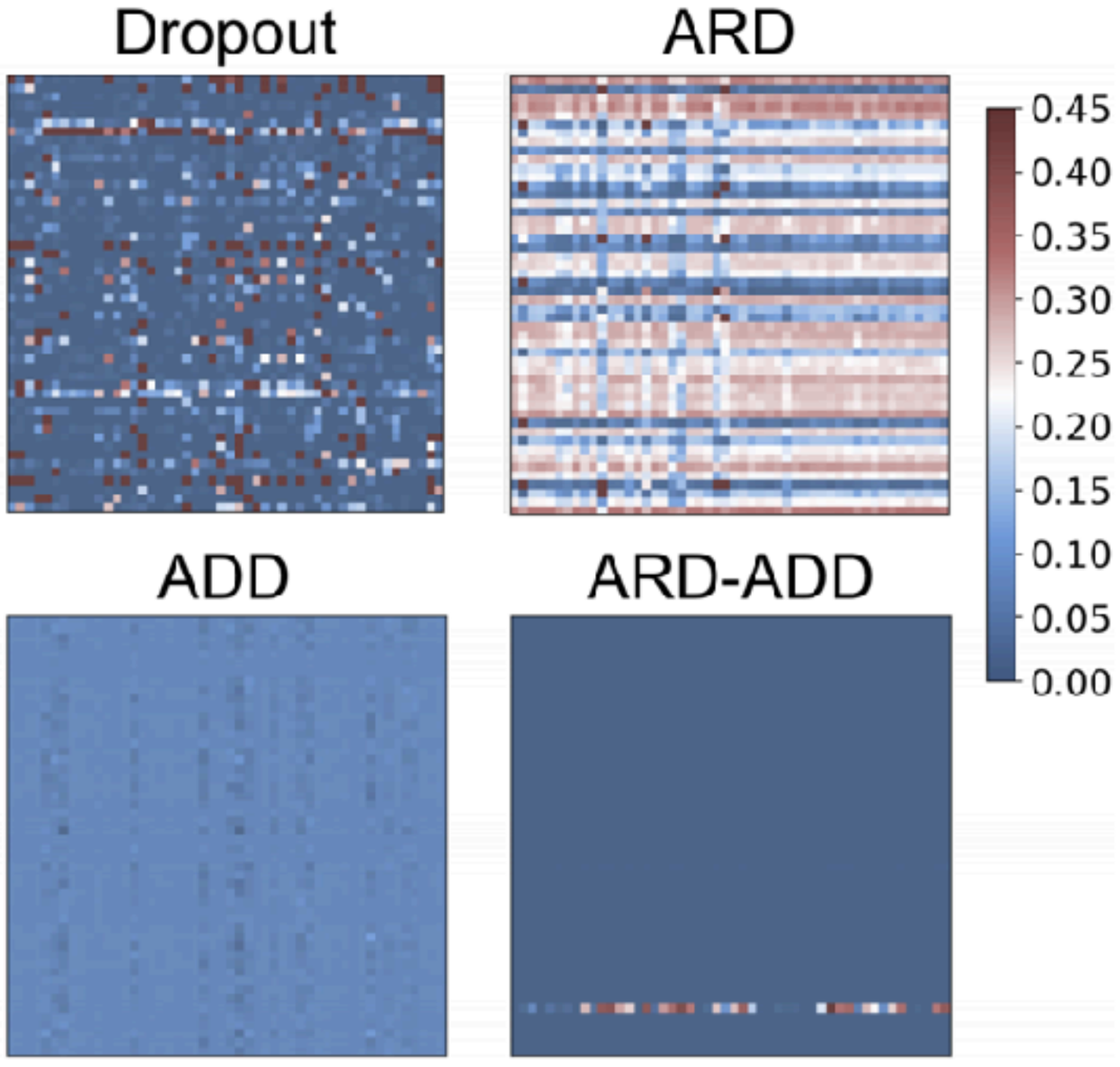
Regression

Test Set RMSE

	Dropout	Prob. Backprop	Deep GP	ARD	ADD	ARD-ADD
Boston	2.80 ±.13	2.795 ±.16	2.38 ±.12	**2.158** ±.20	2.343 ±.31	**2.367** ±.18
Concrete	4.50 ±.18	5.241 ±.12	4.64 ±.11	3.805 ±.28	4.084 ±.34	**3.761** ±.23
Energy	**0.47** ±.01	0.903 ±.05	0.57 ±.02	0.852 ±.01	0.867 ±.11	0.853 ±.08
Kin8nm	0.08 ±.00	0.071 ±.00	**0.05** ±.00	0.066 ±.01	0.064 ±.00	0.064 ±.00
Power	3.63 ±.04	4.028 ±.03	3.60 ±.03	3.486 ±.10	3.290 ±.06	**3.236** ±.07
Wine	0.60 ±.01	0.643 ±.01	**0.50** ±.01	0.561 ±.03	0.555 ±.01	0.538 ±.03
Yacht	0.66 ±.06	0.848 ±.05	0.98 ±.09	0.691 ±.12	0.657 ±.14	**0.604** ±.16
Avg. Rank	4.4 ±1.7	5.6 ±0.5	3.1 ±1.8	3.0 ±1.1	2.9 ±10	**2.0** ±1.1

Posterior Structure

Heat map of summed moments (mean^2 + variance)



Summary

- ⊗ **Clarified** Dropout's modeling assumptions, **generalized** its Bayesian interpretation.

- ⊗ Derived a new prior (ADD) to **control the effective depth** of Residual Networks.

- ⊗ Showed our priors (w/ variational EM) **can serve as direct replacement for dropout** in predictive tasks.

Thank you. Questions?

ICML 2019

enalisnick / dropout_icml2019

In collaboration with:

José Miguel Hernández-Lobato

Padhraic Smyth