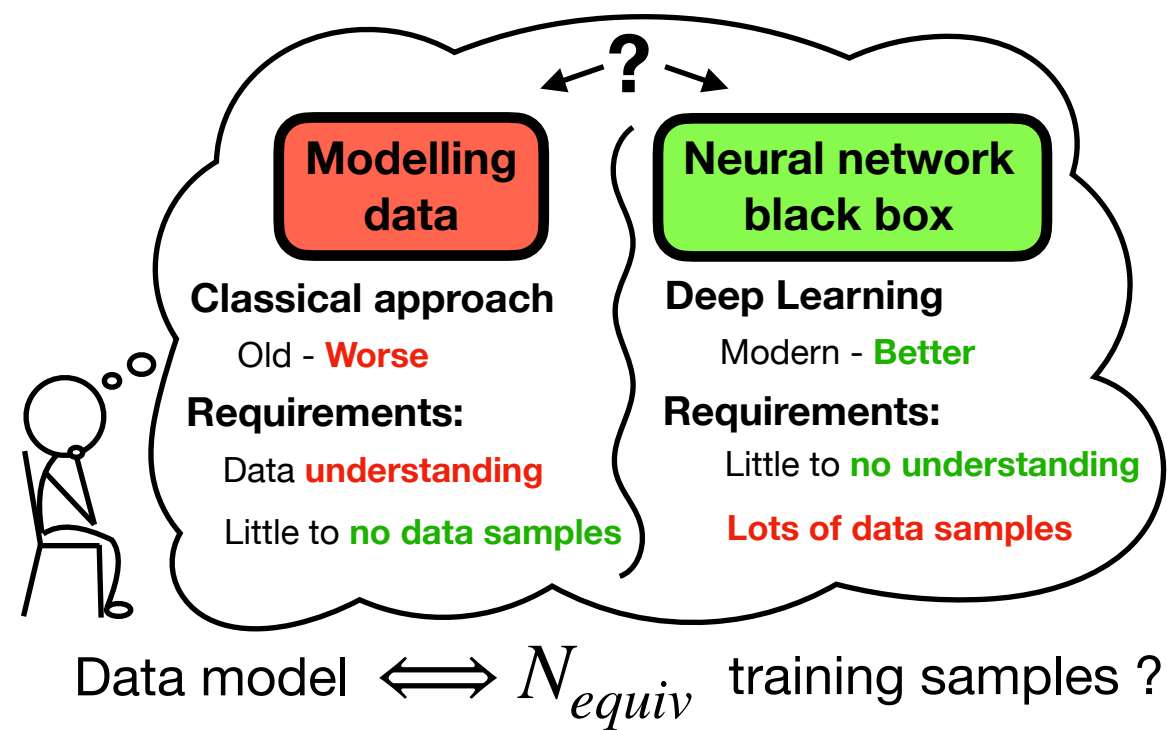
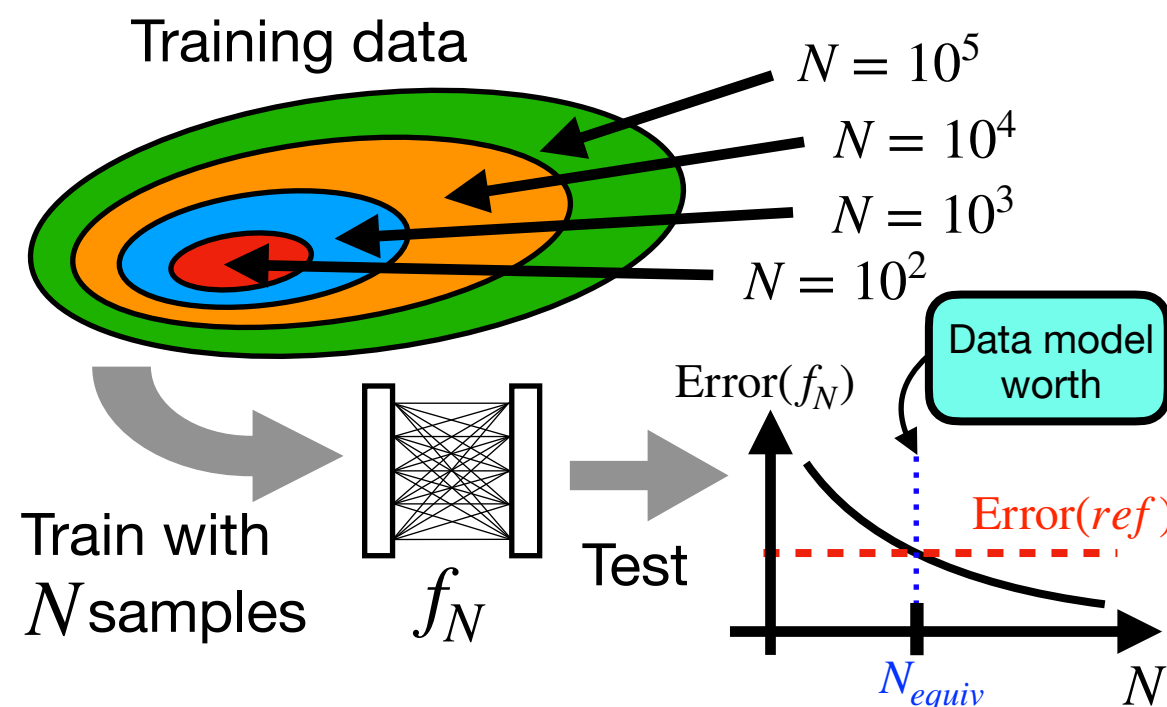


Classical Data Model or NN?

New estimation task:



Neural networks:

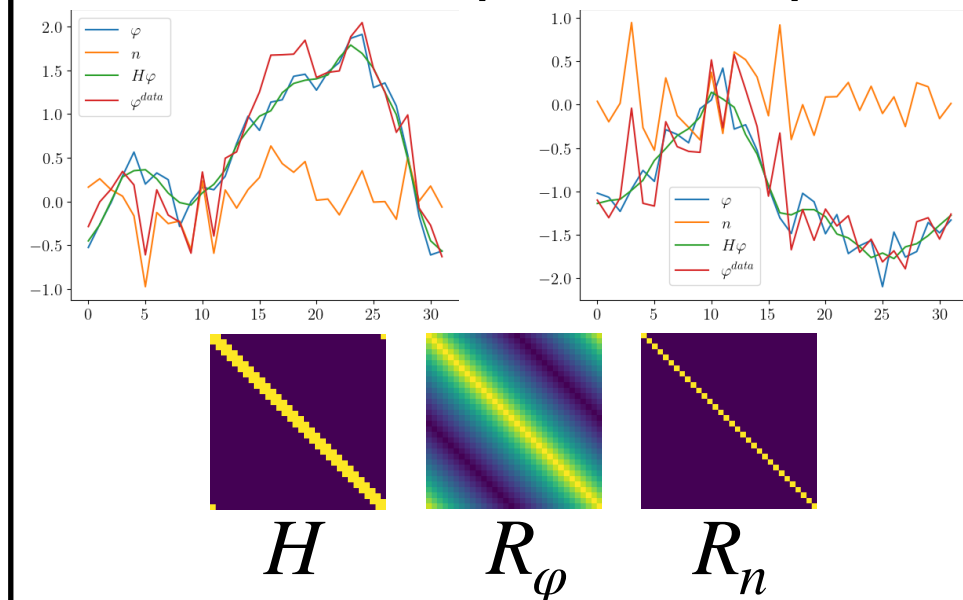


Take-home Message

- On toy examples:
Data model $\iff \geq 10^4$ training samples
- On real problems:
Greater complexity \implies More data needed
- Best practice (not studied):
Combine better of two worlds:
Data model/understanding + NN training

1D Gaussian Signals: Denoising and Deblurring

Data model: $\varphi^{data} = H\varphi + n$

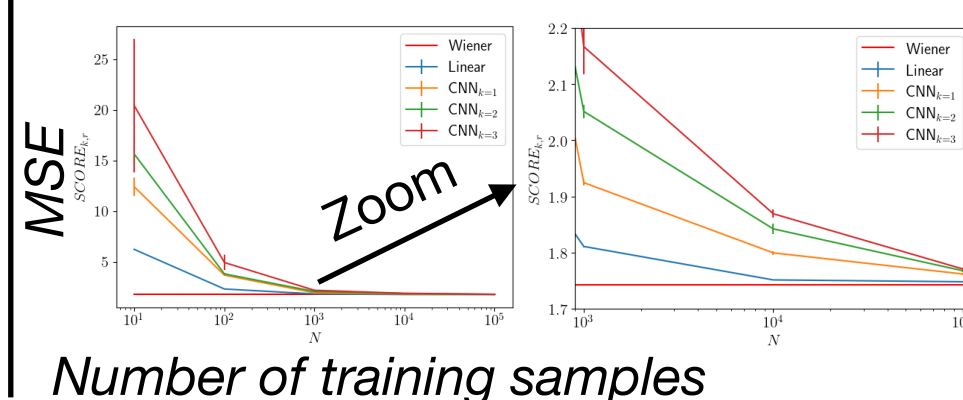


Goal: $\operatorname{argmin} \mathbb{E}(\|\hat{\varphi} - \varphi\|^2)$

Estimation:

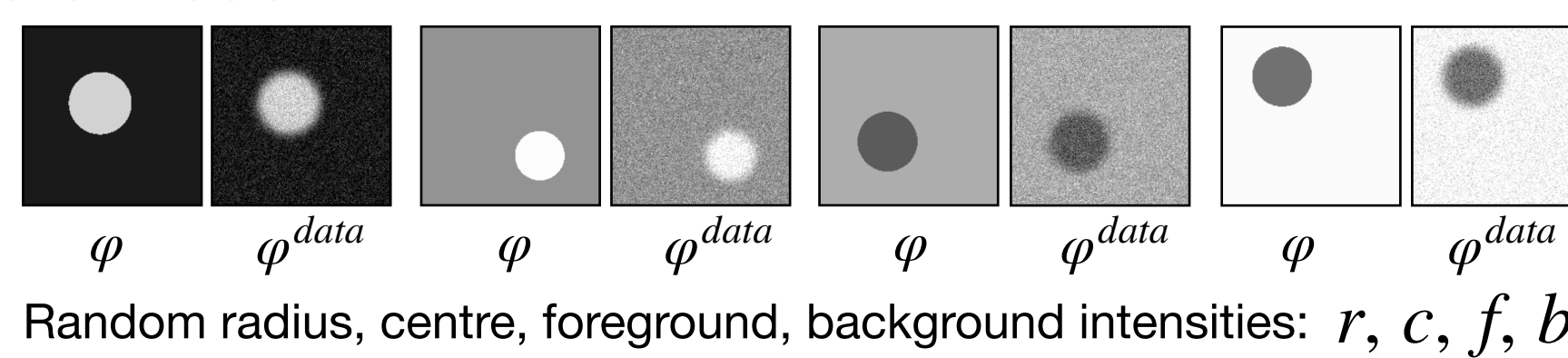
- $\hat{\varphi}^* = W \varphi^{data}$
 $W = R_\varphi H^\top (H R_\varphi H^\top + R_n)^{-1}$
Wiener filter (**optimal** L^2)
 - $\hat{\varphi} = CNN_k(\varphi^{data})$
- Diagram of CNN architecture: Conv Layer \rightarrow ReLU \rightarrow Conv Layer \rightarrow FC layer.

Performance:



2D Disk Images: Geometric Estimation

Data model: $\varphi^{data} = H\varphi + n$

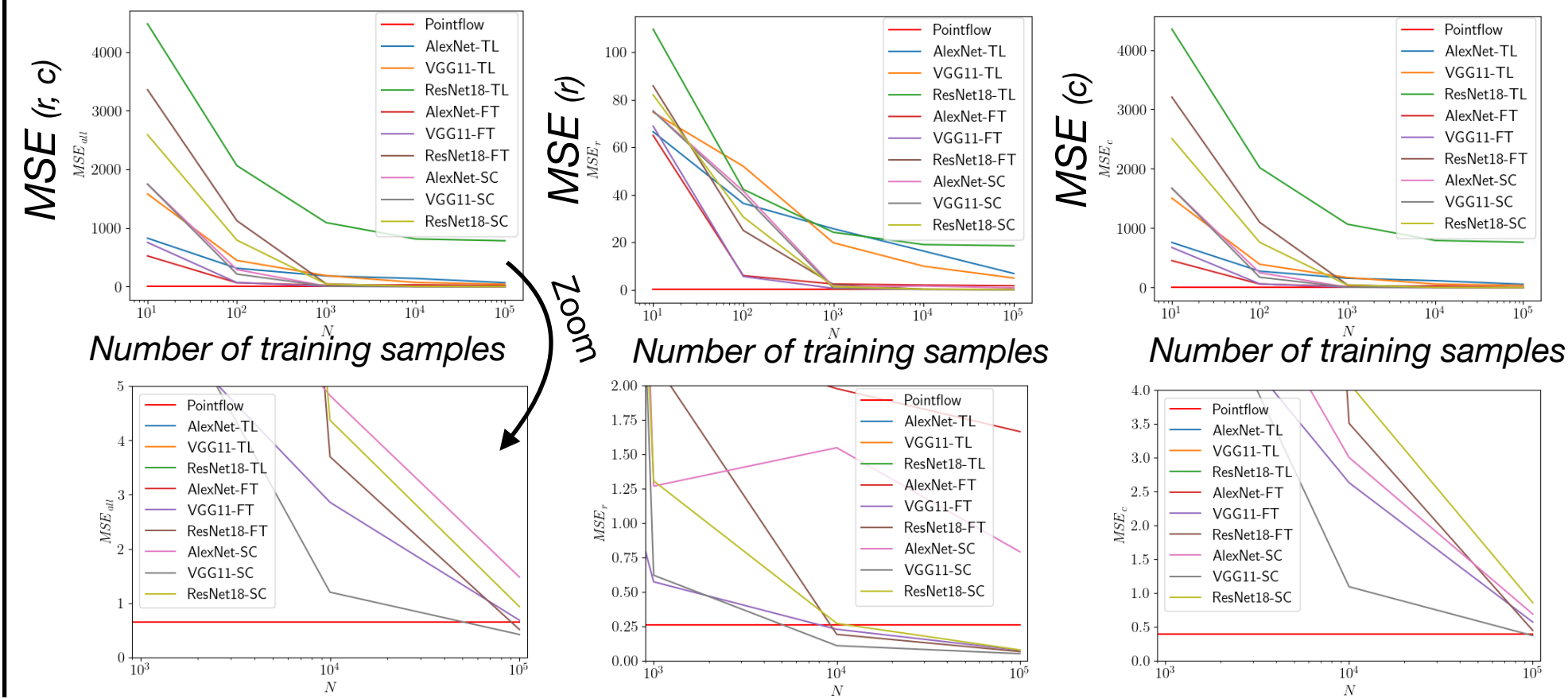


Goal: $\operatorname{argmin} \mathbb{E}(\|(\hat{r}, \hat{c}) - (r, c)\|^2)$

Estimation:

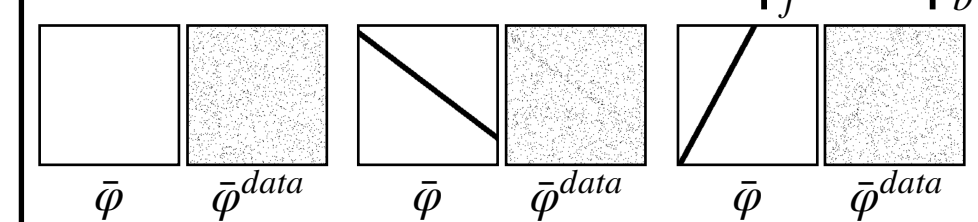
- Pointflow** $V_\pm = \frac{1}{2}(\nabla \|\nabla I_b\| \pm \nabla I_b^\top)$ $\frac{dP}{dt}(t) = V(P(t))$
Yang, F., Cohen, L.D., Bruckstein, A.M.: A model for automatically tracing object boundaries, ICIP (2017)
- NN:** AlexNet, VGG₁₁, ResNet₁₈, ViT_{B16} *From standard Imagenet classification
Training: Transfer-learning* (TL), Finetuning* (FT), From Scratch (SC)

Performance:



2D Random Dots Images: Geometric Classification

Data model: $\mathbb{P}(y_\varphi = 1) = \frac{1}{2}$
 $\varphi^{data} = \varphi_{p_f} \vee n_{p_b}$



Goal: $\operatorname{argmax} AUC(\hat{y}_\varphi, y_\varphi)$

+ Estimation constraint:
“ $\operatorname{argmin} \mathbb{E}(\|(\hat{r}, \hat{\theta}) - (r, \theta)\|^2)$ ”

Classification:

- Hough transform**
Diagram of Hough transform for line detection.
- NN:** AlexNet, VGG₁₁, ResNet₁₈, ViT_{B16}
Training: Transfer-learning* (TL), Finetuning* (FT), From Scratch (SC)
*From standard Imagenet classification

Performance:

