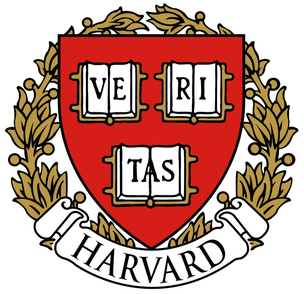
uai2023

On Minimizing the Impact of Dataset Shifts on Actionable Explanations

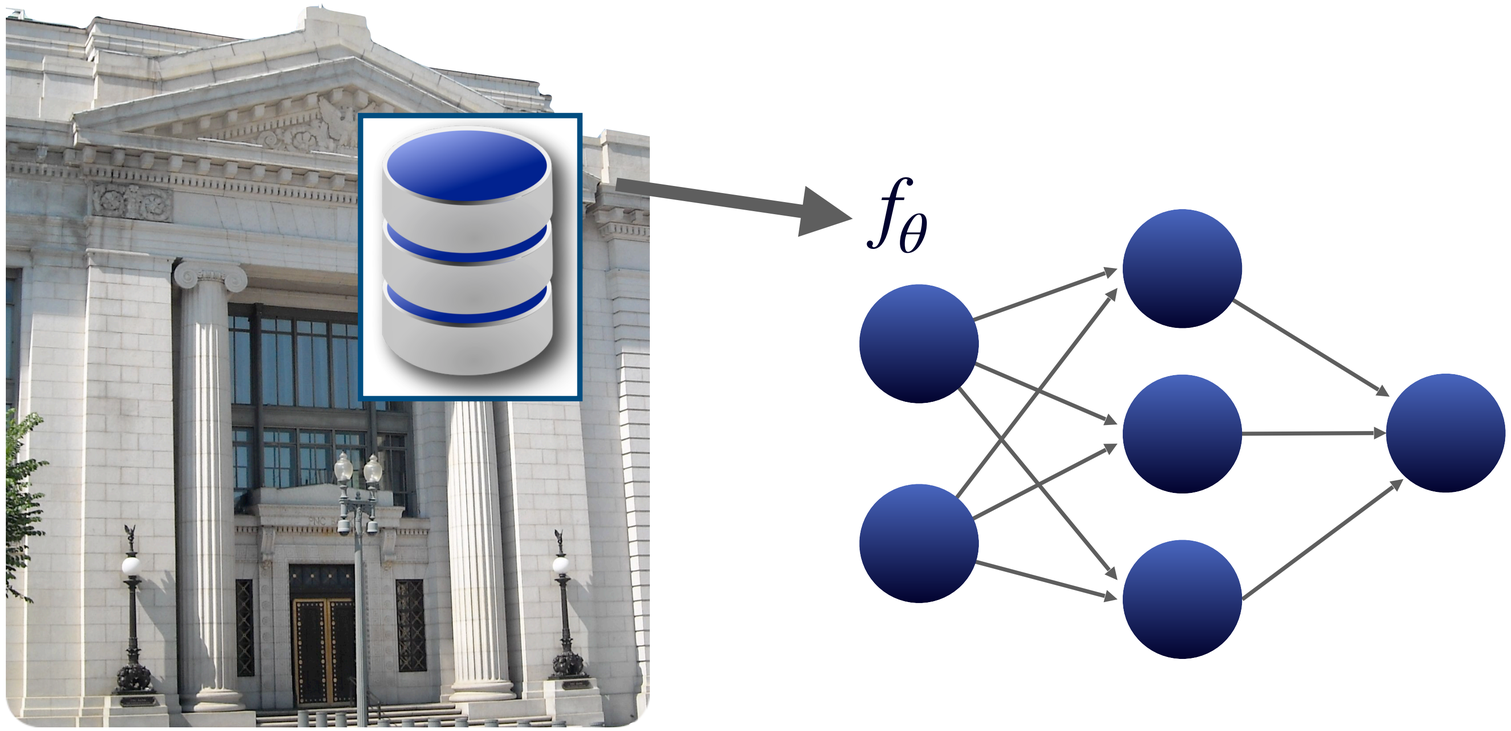
Anna P. Meyer* // Dan Ley* // Suraj Srinivas // Himabindu Lakkaraju

UAI 2023

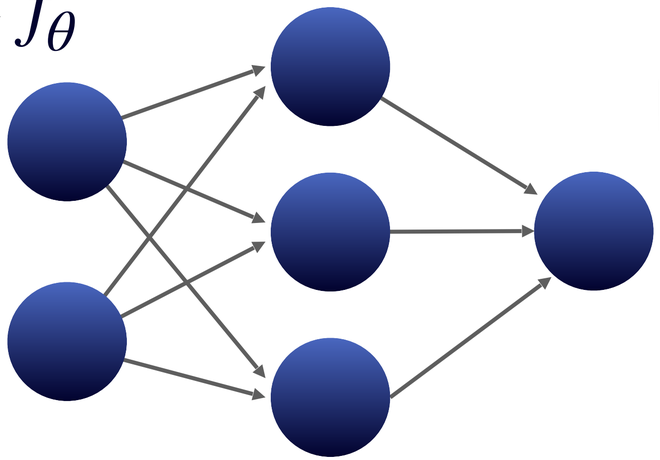
* Equal contribution



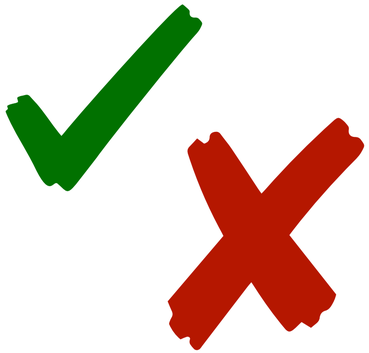
madPL


f_θ

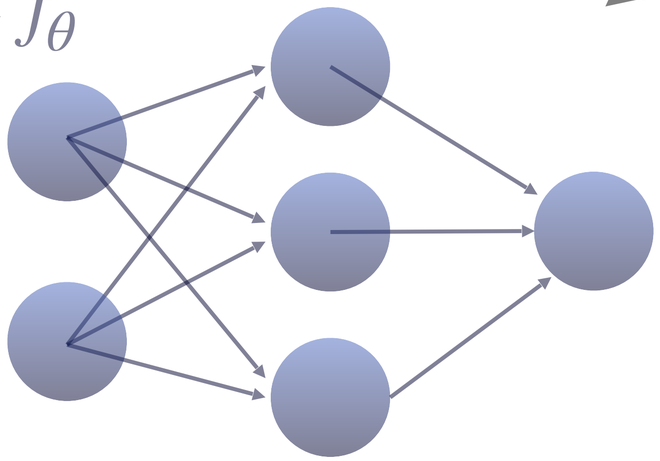
f_θ

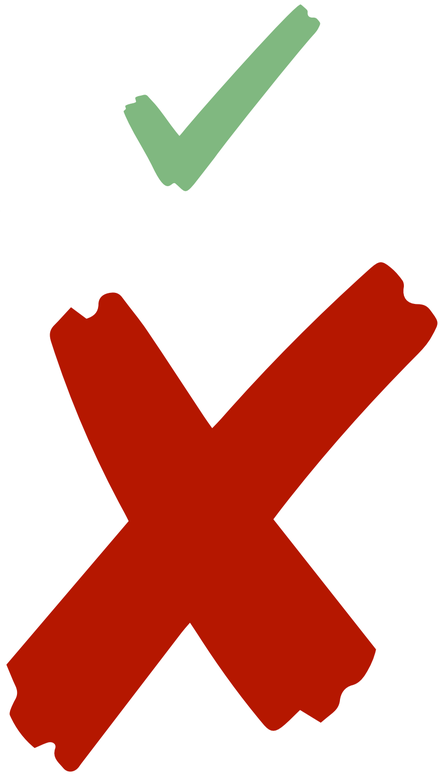


New loan application



f_θ





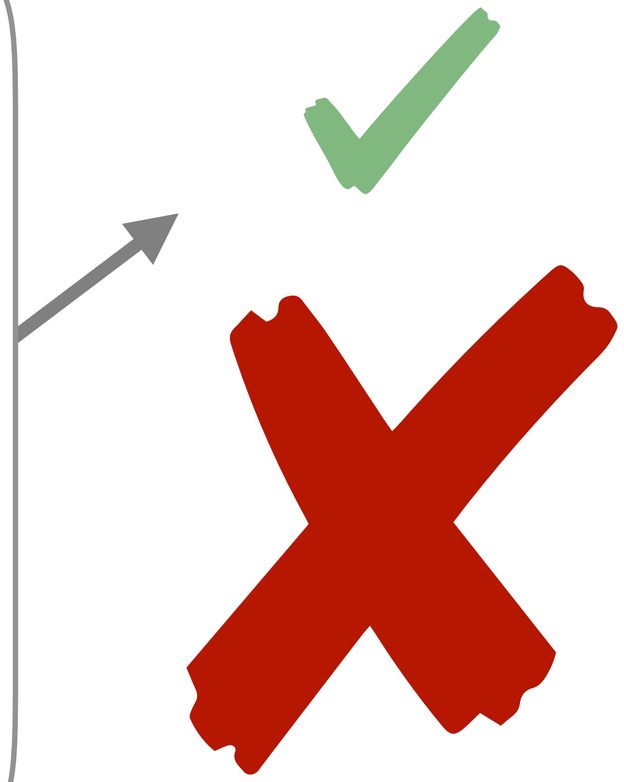
Why ✗ ?

Right to an Explanation

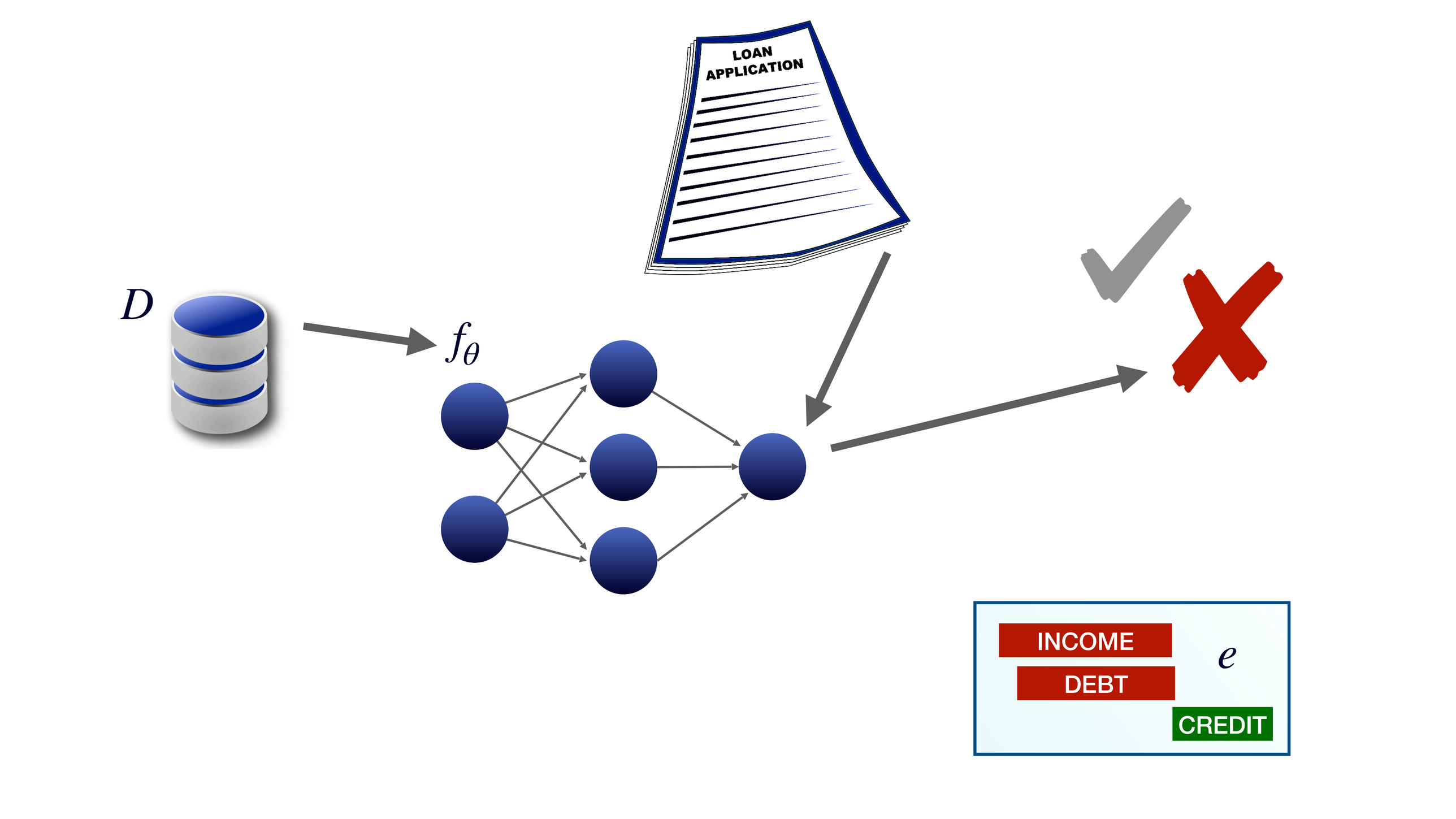
USA’s Equal Credit Opportunity Act (Regulation B)

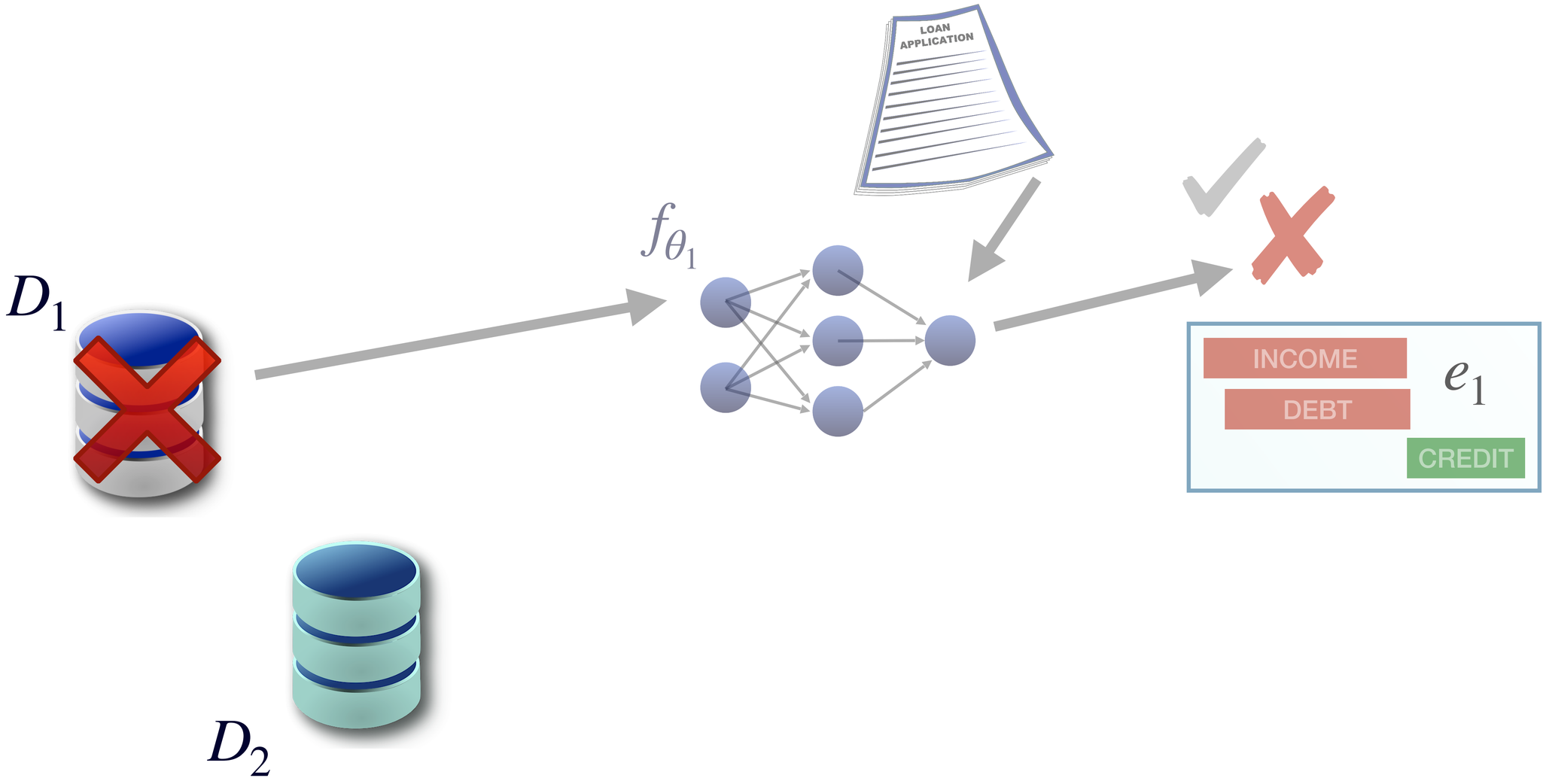
The European Union’s General Data Protection Regulation (Article 22)

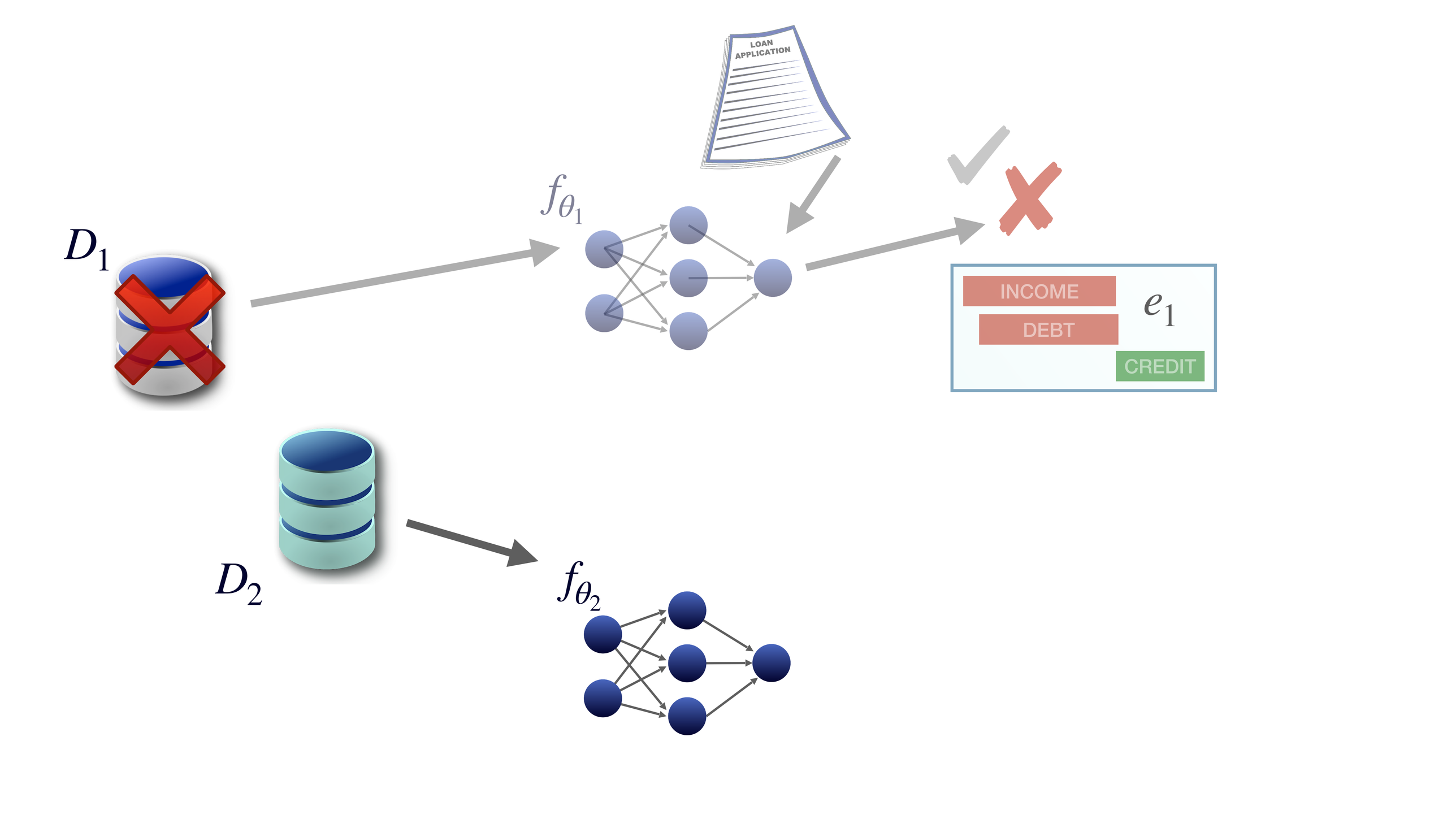
and more

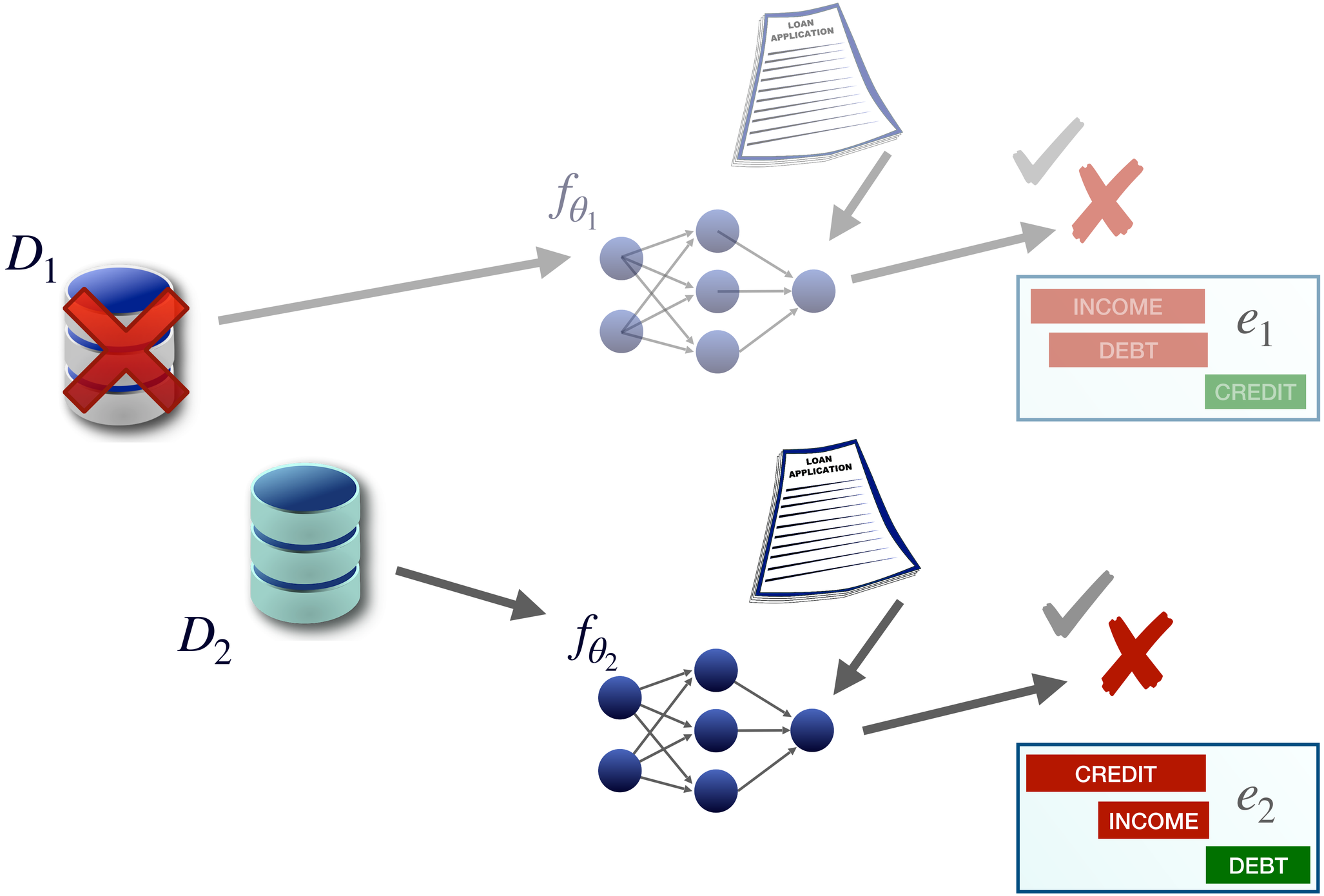


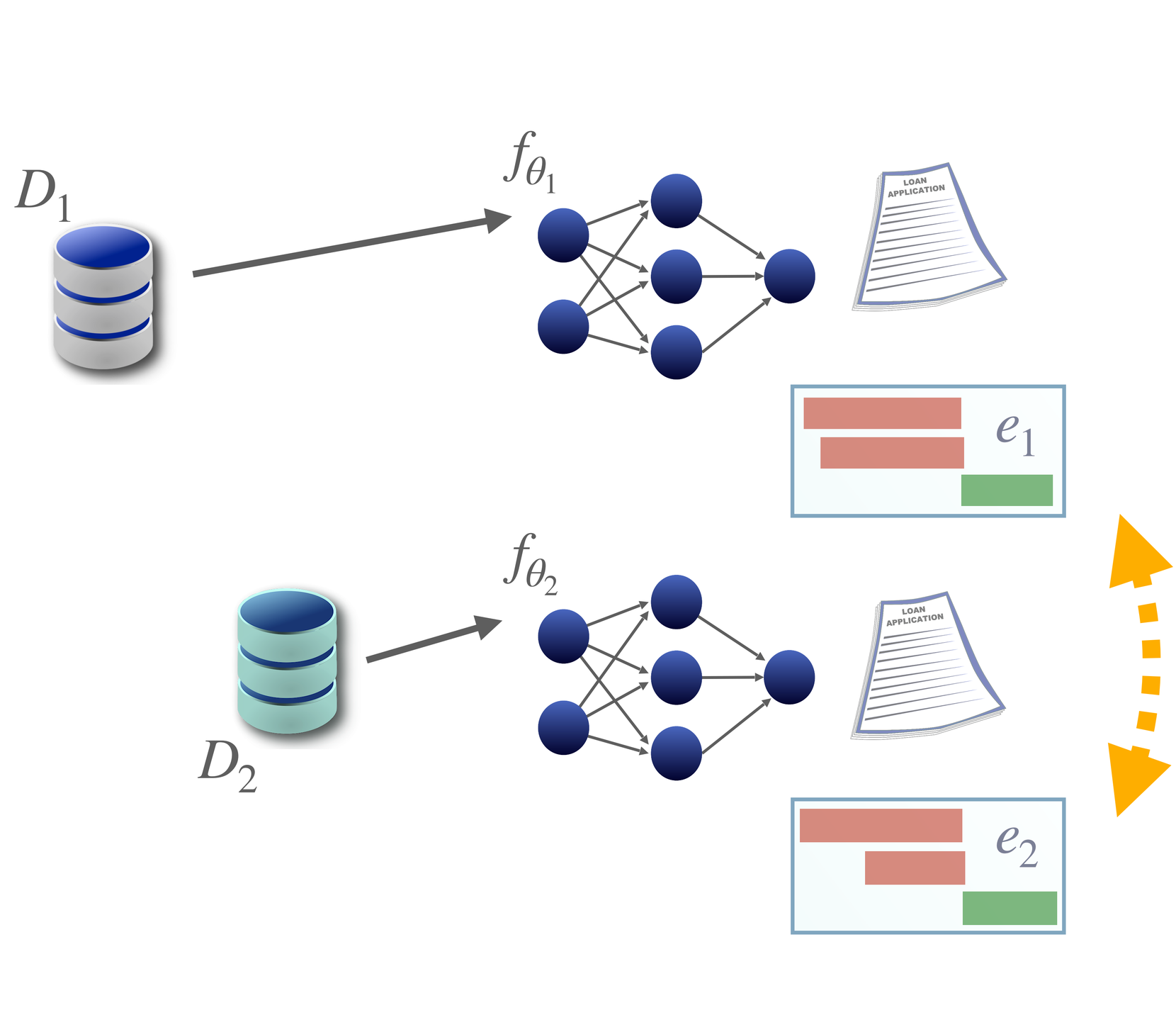
Why ✗ ?

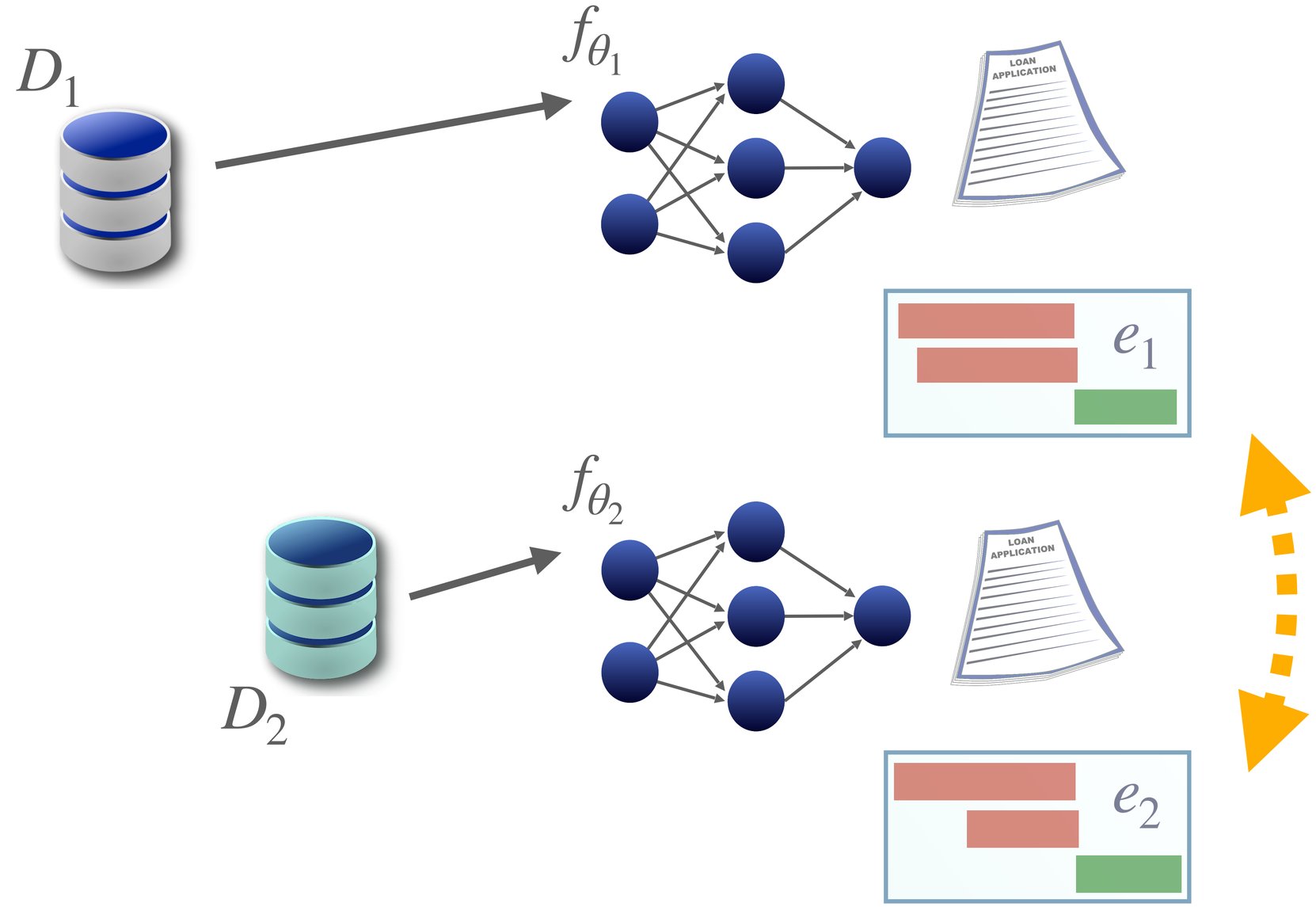

LOAN
APPLICATION
D
f_θ
INCOME
DEBT
CREDIT
e


LOAN APPLICATION
f_{θ_1}
D_1
D_2
INCOME
DEBT
CREDIT
e_1

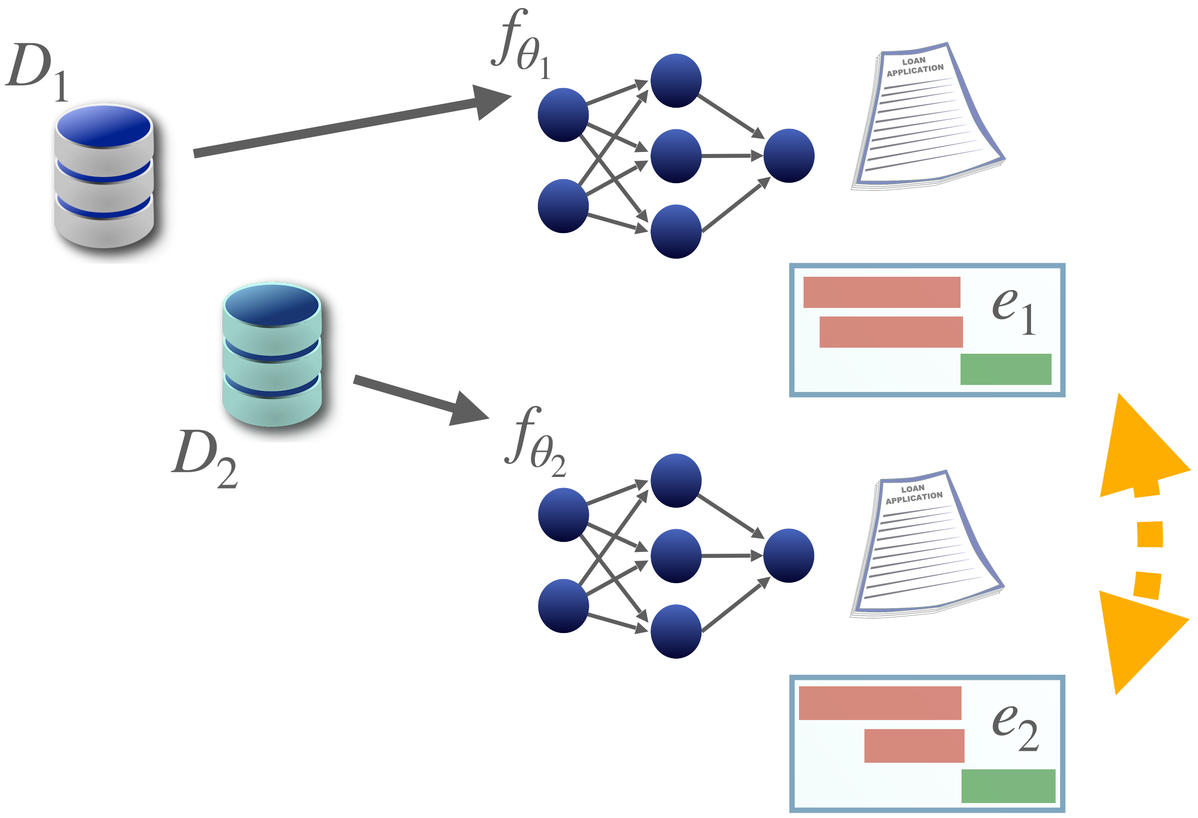

LOAN APPLICATION
f_{θ_1}
D_1
INCOME
DEBT
CREDIT
e_1
D_2
f_{θ_2}


LOAN APPLICATION
f_{θ_1}
D_1
INCOME
DEBT
CREDIT
e_1
LOAN APPLICATION
D_2
f_{θ_2}
CREDIT
INCOME
DEBT
e_2


D_1
f_{θ_1}
LOAN APPLICATION
e_1
D_2
f_{θ_2}
LOAN APPLICATION
e_2

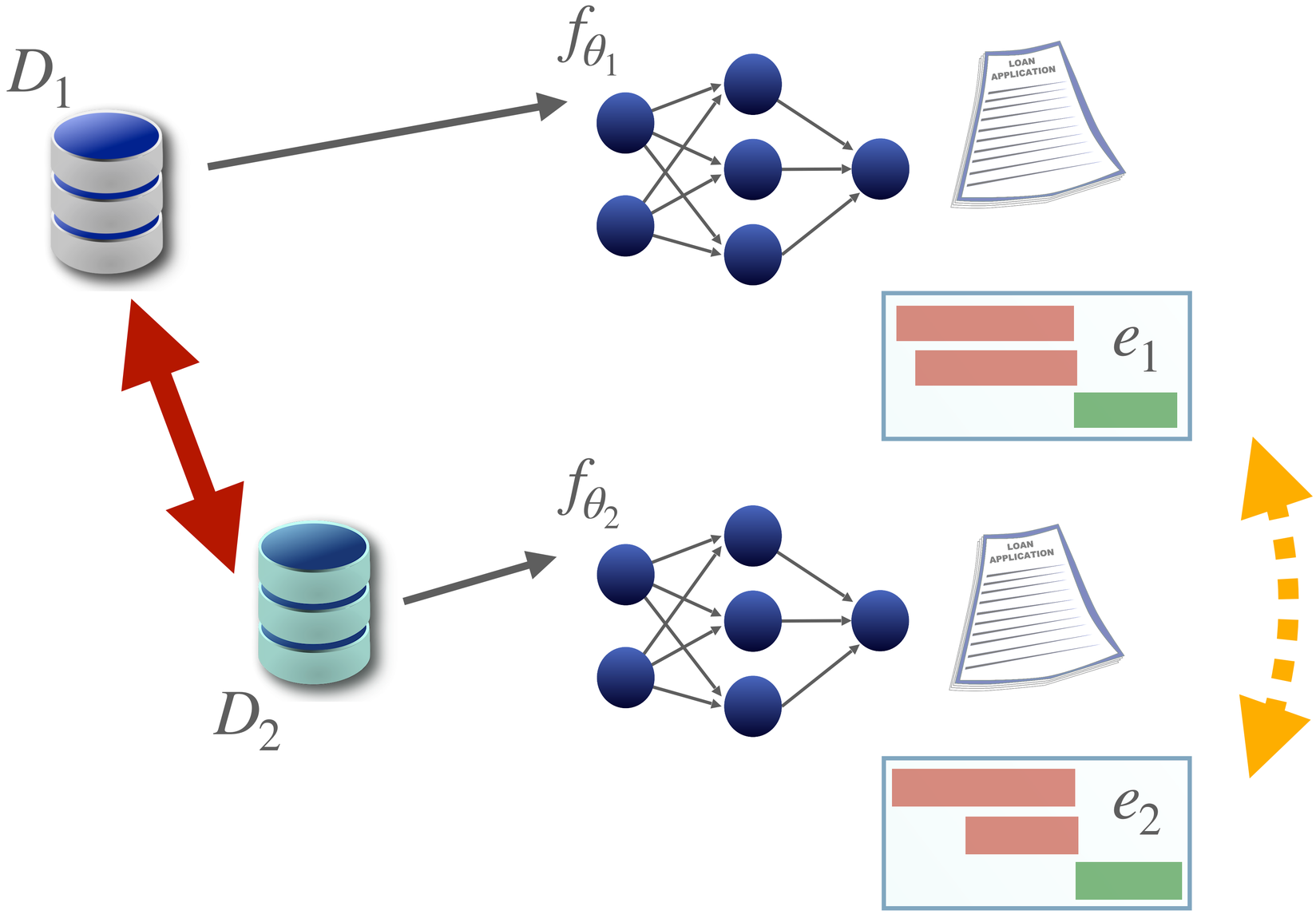


Will e_1 remain **actionable** after retraining?



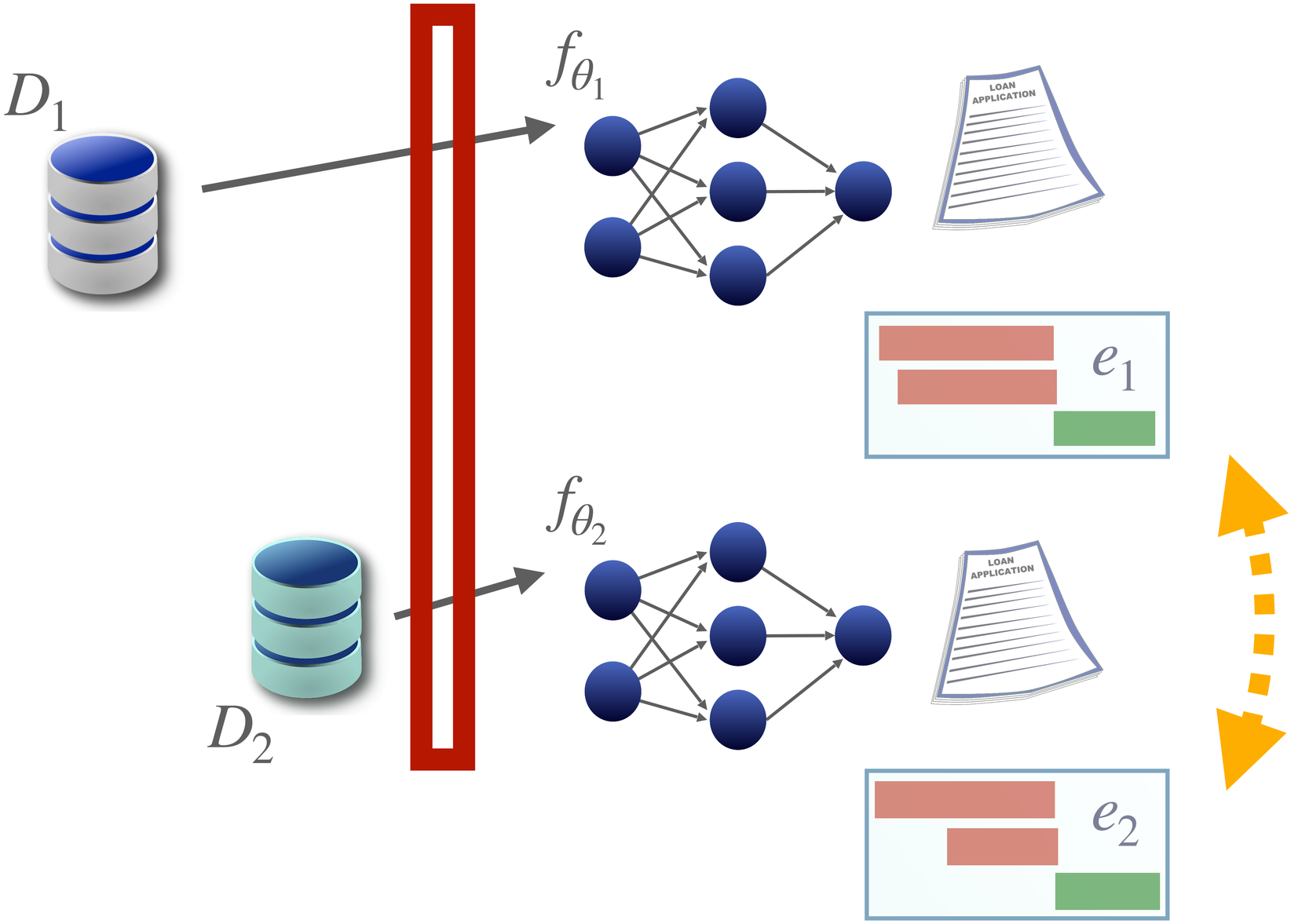
How much do the explanations for a model f_{θ_1} trained on dataset D_1 change when retraining on a slightly shifted dataset D_2 resulting in a new model f_{θ_2}?

Explanation shift is affected by…



1. Dataset shift size

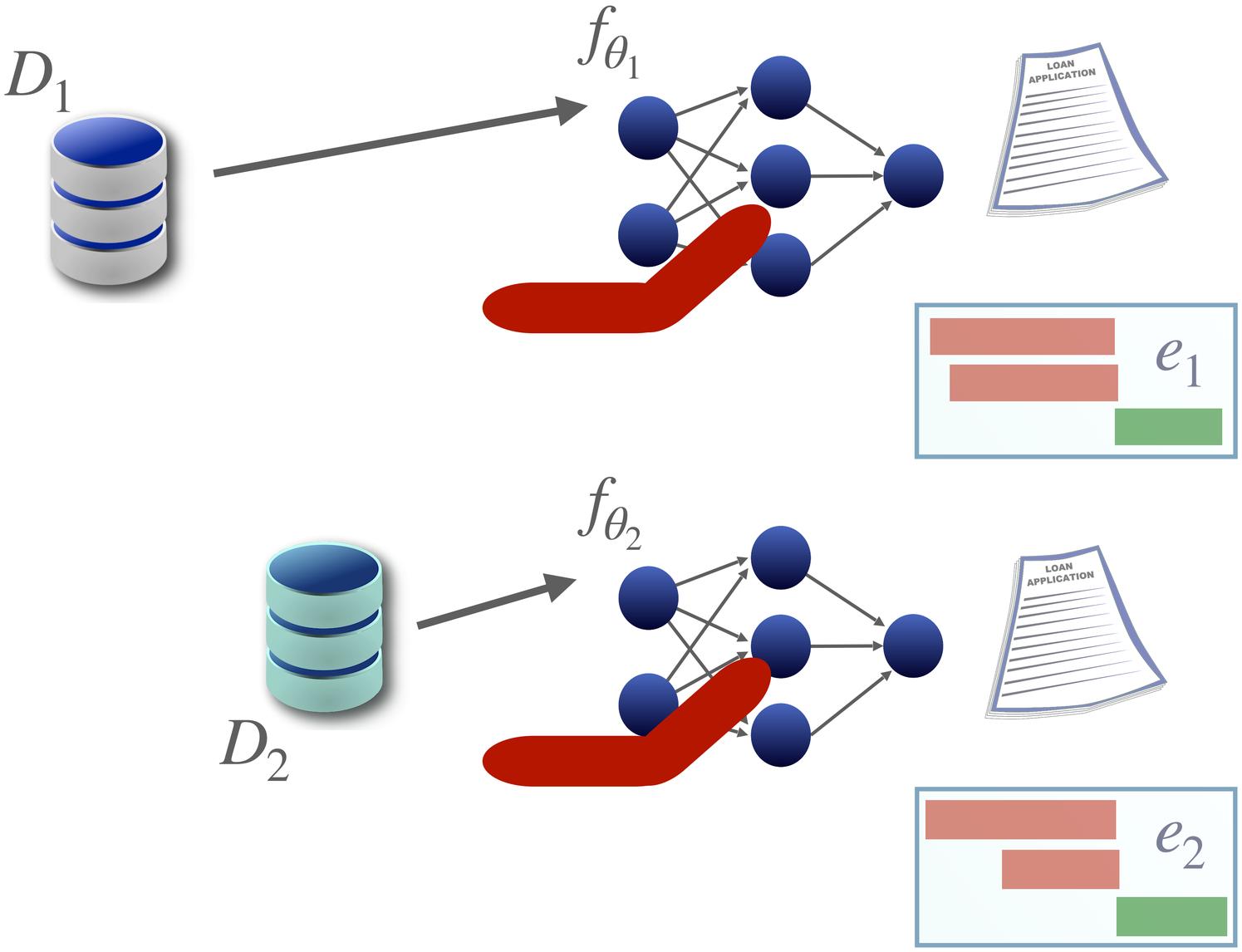
Explanation shift is affected by…



1. Dataset shift size

2. Weight decay parameter

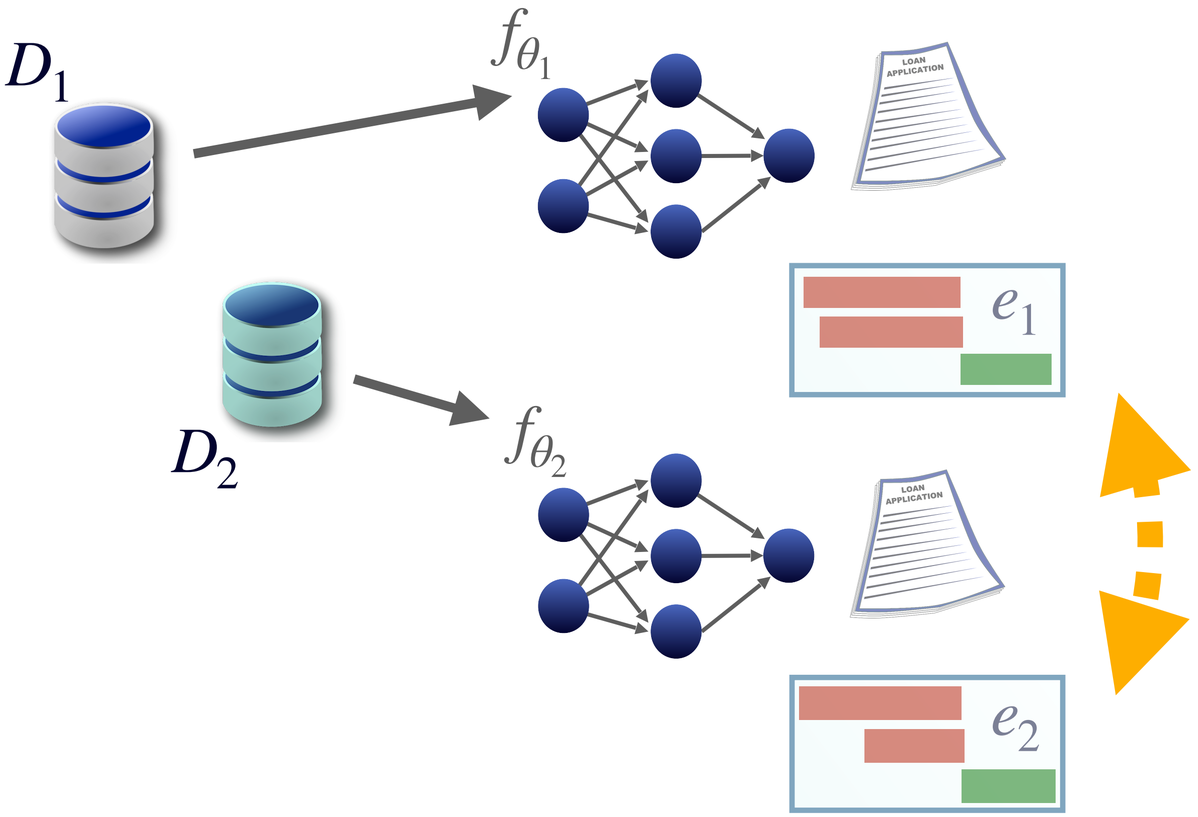
Explanation shift is affected by…

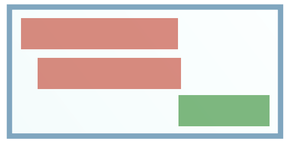


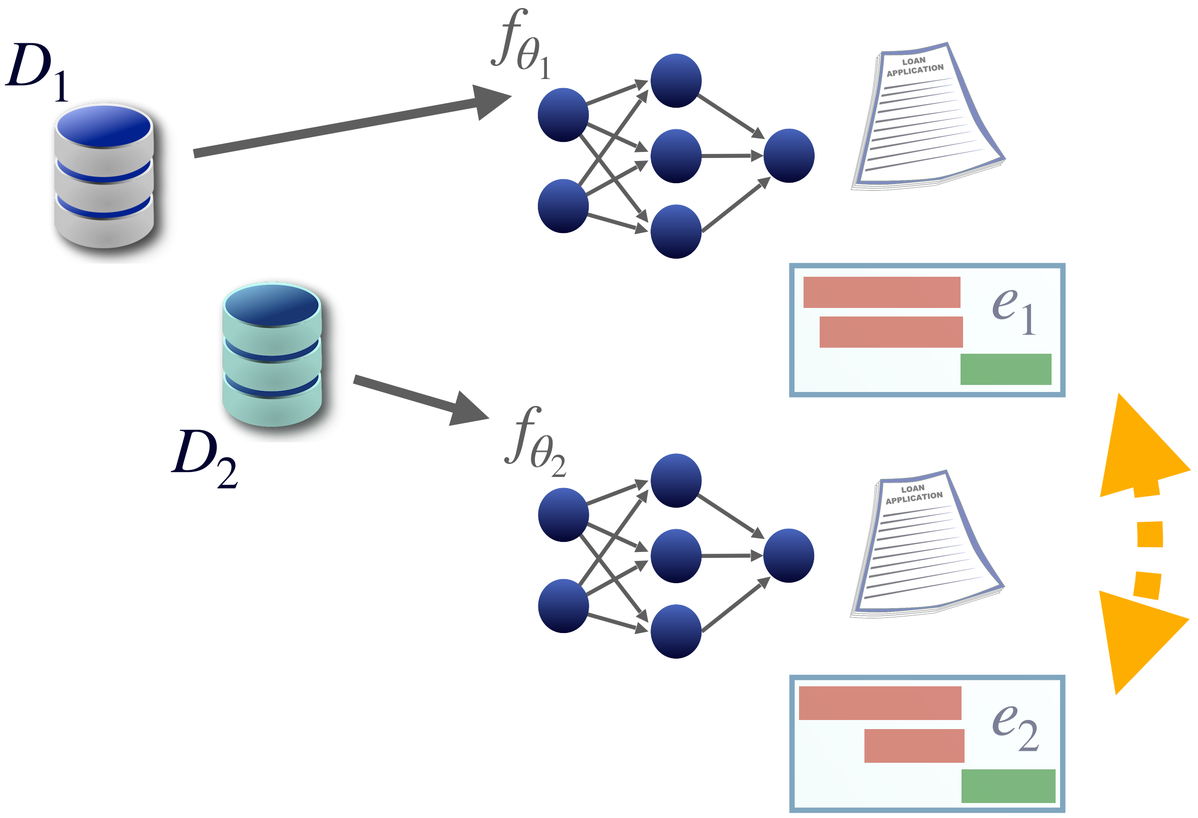
1. Dataset shift size

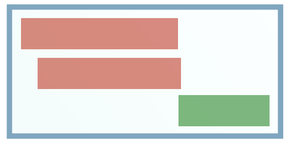
2. Weight decay parameter

3. Smoothness of the activation function

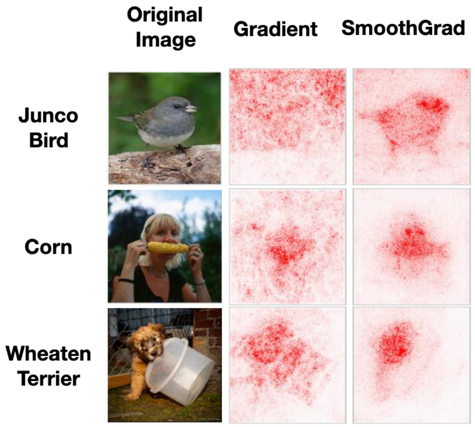

D_1
f_{θ_1}
LOAN APPLICATION
e_1
D_2
f_{θ_2}
LOAN APPLICATION
e_2

What is  ?

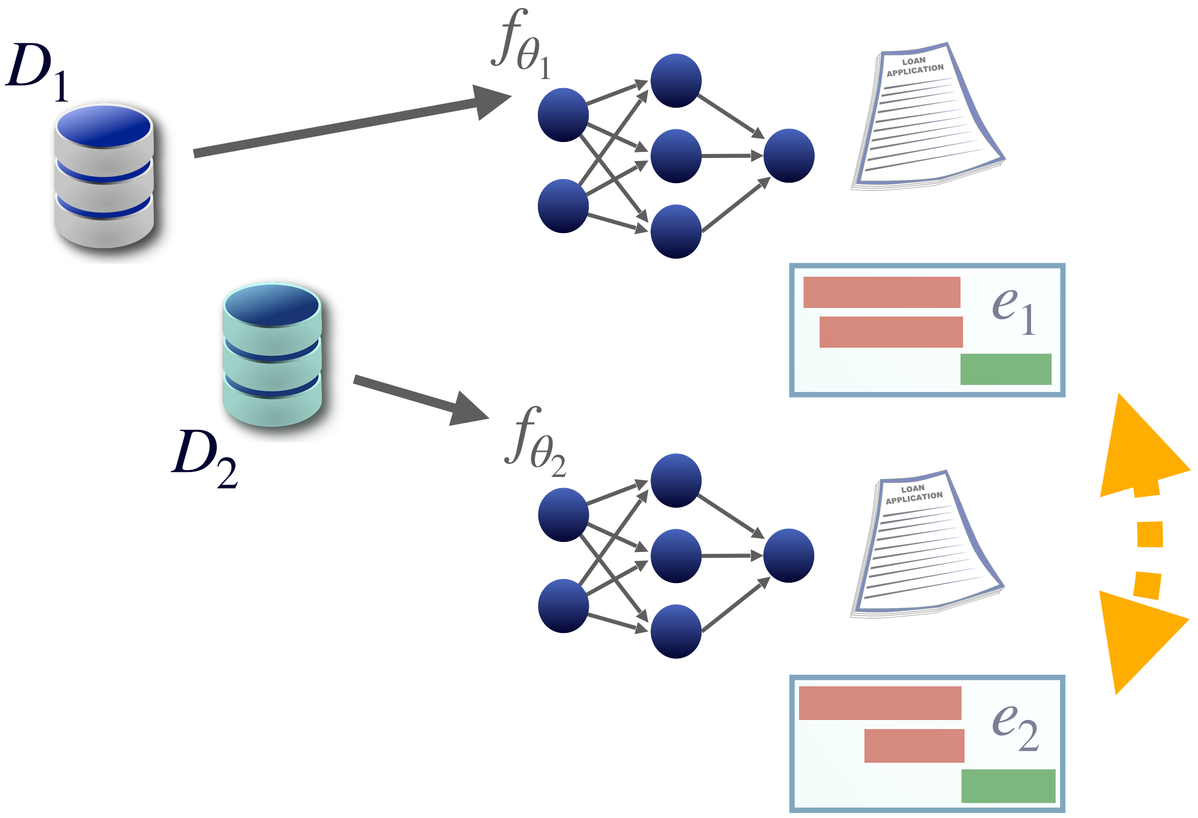


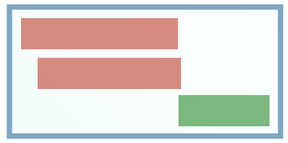
What is  ?

- Gradient-based (input gradients, SmoothGrad)

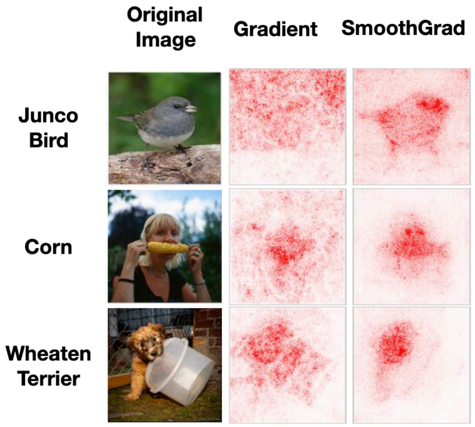


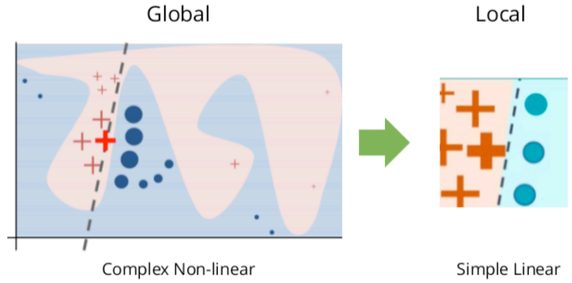
Images from
https://c3.ai/glossary/data-science/lime-local-interpretable-model-agnostic-explanations/
and "Sanity Checks for Saliency Maps," Adebayo et al., 2018.



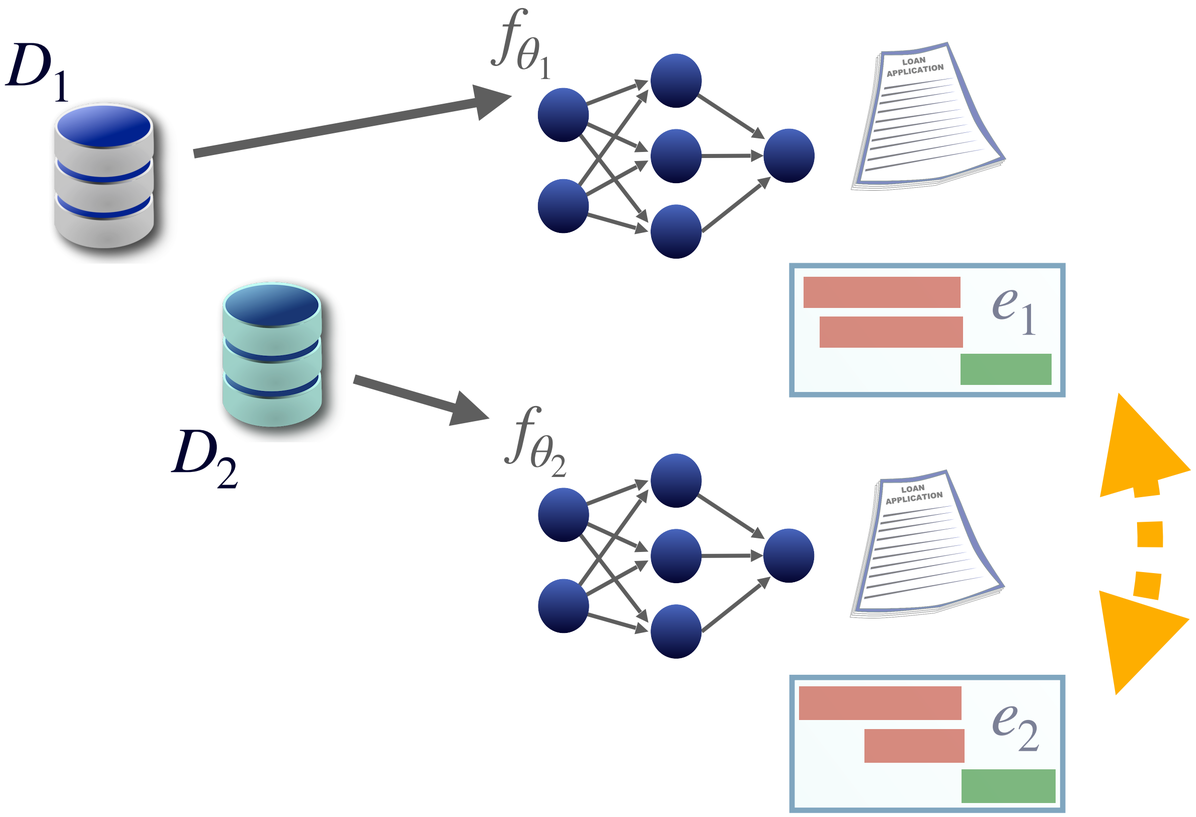
What is  ?

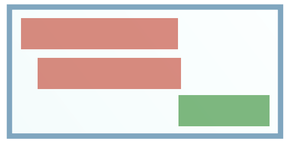
- Gradient-based (input gradients, SmoothGrad)
- Perturbation based (LIME, SHAP)





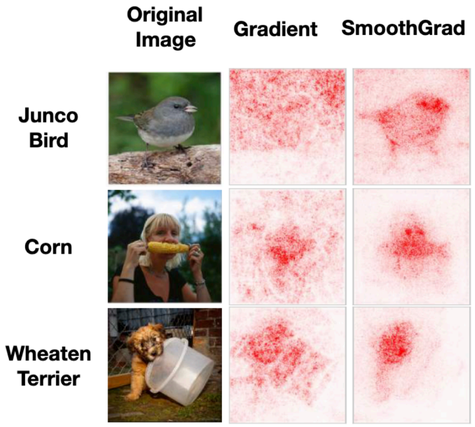
Images from
https://c3.ai/glossary/data-science/lime-local-interpretable-model-agnostic-explanations/
and "Sanity Checks for Saliency Maps," Adebayo et al., 2018.

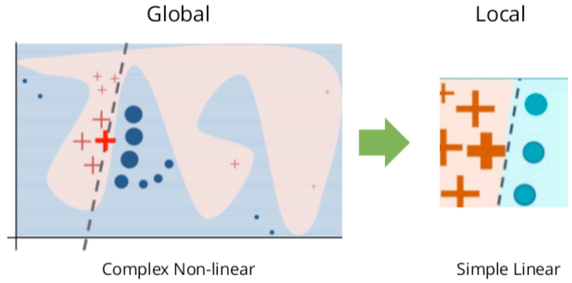


What is  ?

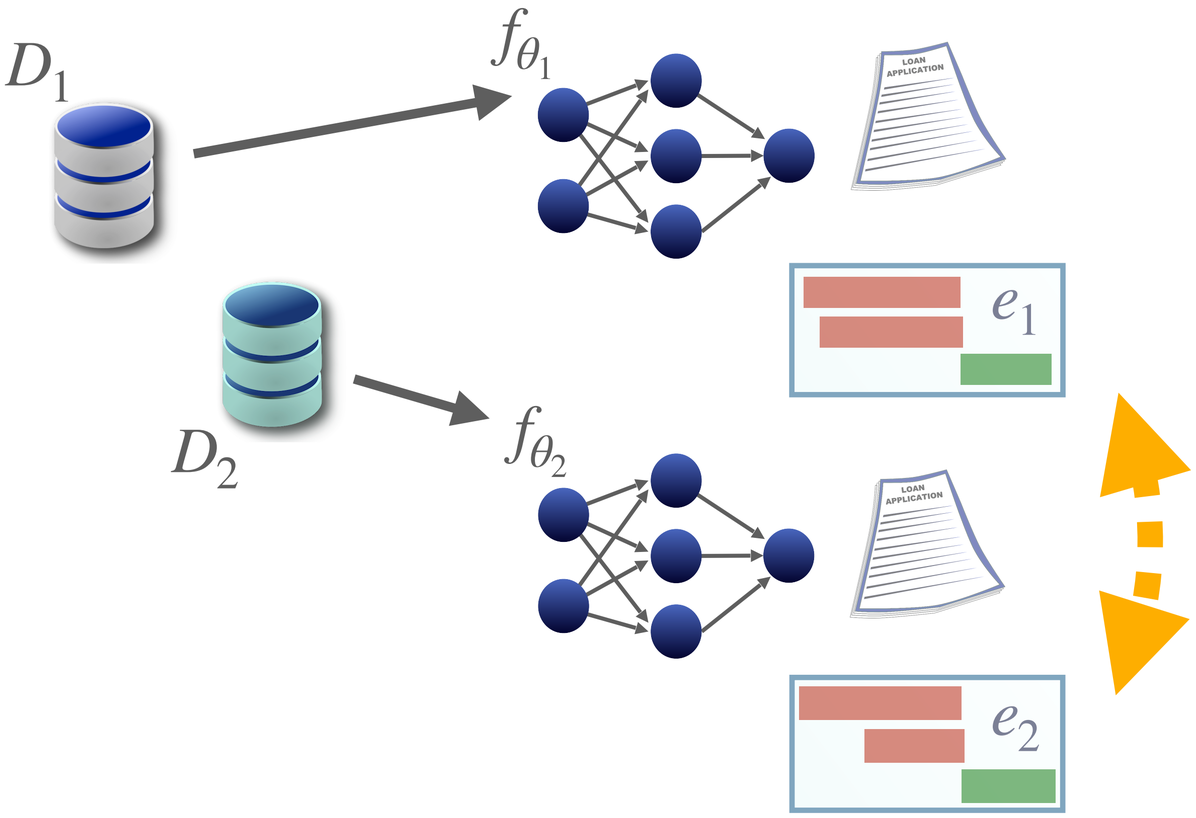
- Gradient-based (input gradients, SmoothGrad)
- Perturbation based (LIME, SHAP)

We focus on **gradient-based** (see Han et al., NeurIPS 2022)





Images from
https://c3.ai/glossary/data-science/lime-local-interpretable-model-agnostic-explanations/
and "Sanity Checks for Saliency Maps," Adebayo et al., 2018.



gradients

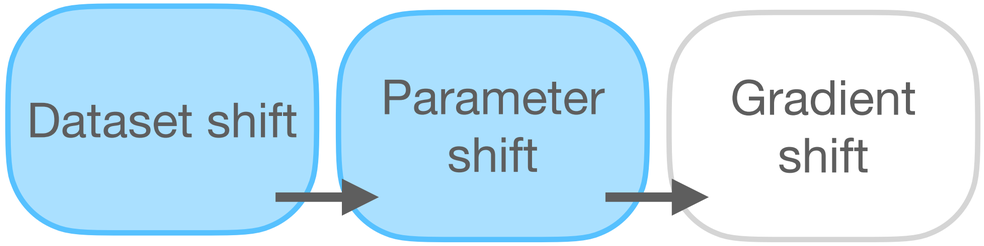
How much do the ~~explanations~~ for a model f_{θ_1} trained on dataset D_1 change when retraining on a slightly shifted dataset D_2 resulting in a new model f_{θ_2}?

How much do the gradients for a model f_{θ_1} trained on dataset D_1 change when retraining on a slightly shifted dataset D_2 resulting in a new model f_{θ_2}?

What is the dataset shift $d(D_1, D_2)$?

What is the parameter shift $\|\theta_2 - \theta_1\|_2$?

What is the gradient shift $\|\nabla_{\mathbf{x}} f_{\theta_2}(x) - \nabla_{\mathbf{x}} f_{\theta_1}(x)\|$?

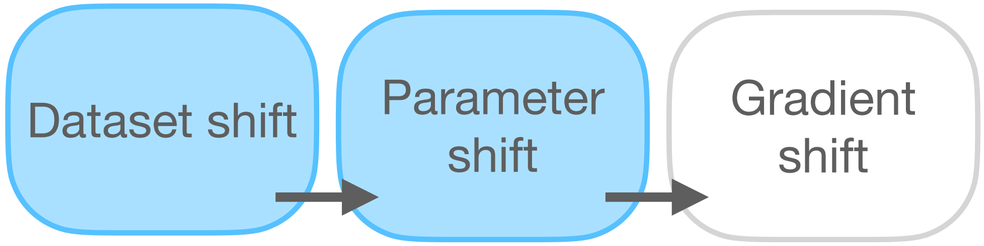


Assumptions

1. We minimize a regularized loss l_{reg} that involves weight decay, i.e., $l_{reg}(\theta) = l(\theta) + \gamma\|\theta\|_2^2$
2. The loss l is locally quadratic
3. The learning algorithm returns a unique minimum θ given a dataset D

Theorem 1: Given the assumptions stated to the left, we have:

$$\|\theta_2 - \theta_1\|_2 \leq \sqrt{\frac{L_x(\theta_1)\ d(D_1, D_2)}{\gamma}} + C$$



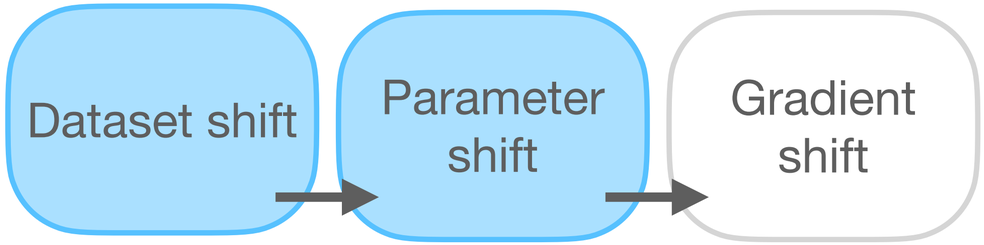
Theorem 1: Given the assumptions stated to the left, we have:

$$\|\theta_2 - \theta_1\|_2 \leq \sqrt{\frac{L_x(\theta_1)\, d(D_1, D_2)}{\gamma}} + C$$

Assumptions

1. We minimize a regularized loss l_{reg} that involves weight decay, i.e., $l_{reg}(\theta) = l(\theta) + \gamma\|\theta\|_2^2$
2. The loss l is locally quadratic
3. The learning algorithm returns a unique minimum θ given a dataset D

$L_{\mathbf{x}}(\theta_1)$ is the Lipschitz constant of the original model



Assumptions

1. We minimize a regularized loss l_{reg} that involves weight decay, i.e., $l_{reg}(\theta) = l(\theta) + \gamma\|\theta\|_2^2$
2. The loss l is locally quadratic
3. The learning algorithm returns a unique minimum θ given a dataset D

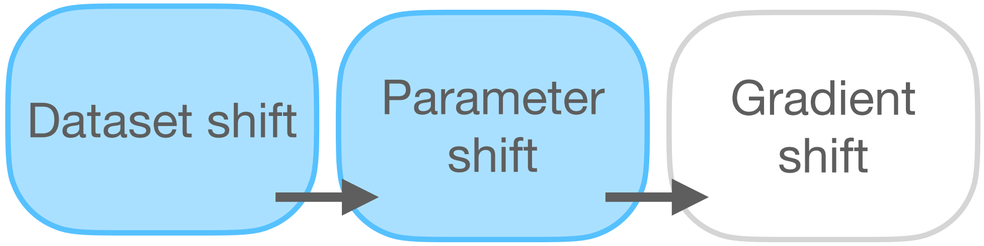
$L_{\mathbf{x}}(\theta_1)$ is the Lipschitz constant of the original model

Theorem 1: Given the assumptions stated to the left, we have:

$$\|\theta_2 - \theta_1\|_2 \leq \sqrt{\frac{L_x(\theta_1)\, d(D_1, D_2)}{\gamma}} + C$$

$d(D_1, D_2)$ is the minimum average L2 distance over all possible matchings of samples in D_1 and D_2, i.e.,

$$d(D_1, D_2) = \min_{P(D_2)} \sum_{i=1}^{N} \|x_i - x_i'\|_2$$



Theorem 1: Given the assumptions stated to the left, we have:

$$\|\theta_2 - \theta_1\|_2 \leq \sqrt{\frac{L_x(\theta_1)\ d(D_1, D_2)}{\gamma}} + C$$

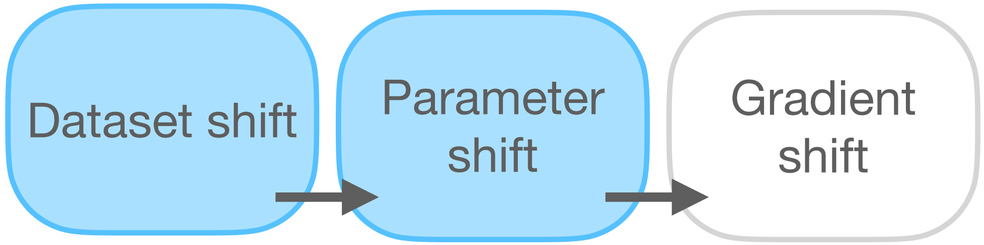
Assumptions

1. We minimize a regularized loss l_{reg} that involves weight decay, i.e., $l_{reg}(\theta) = l(\theta) + \gamma\|\theta\|_2^2$
2. The loss l is locally quadratic
3. The learning algorithm returns a unique minimum θ given a dataset D

$L_{\mathbf{x}}(\theta_1)$ is the Lipschitz constant of the original model

$d(D_1, D_2)$ is the minimum average L2 distance over all possible matchings of samples in D_1 and D_2, i.e.,

$$d(D_1, D_2) = \min_{P(D_2)} \sum_{i=1}^{N} \|x_i - x_i'\|_2$$



Assumptions

1. We minimize a regularized loss l_{reg} that involves weight decay, i.e., $l_{reg}(\theta) = l(\theta) + \gamma\|\theta\|_2^2$
2. The loss l is locally quadratic
3. The learning algorithm returns a unique minimum θ given a dataset D

Theorem 1: Given the assumptions stated to the left, we have:

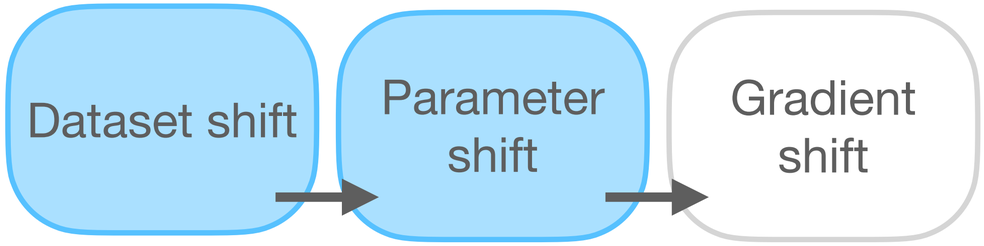
$$\|\theta_2 - \theta_1\|_2 \leq \sqrt{\frac{L_x(\theta_1)\ d(D_1, D_2)}{\gamma}} + C$$

C is a small problem-dependent constant

$L_{\mathbf{x}}(\theta_1)$ is the Lipschitz constant of the original model

$d(D_1, D_2)$ is the minimum average L2 distance over all possible matchings of samples in D_1 and D_2, i.e.,

$$d(D_1, D_2) = \min_{P(D_2)} \sum_{i=1}^{N} \|x_i - x_i'\|_2$$



Theorem 1: Given the assumptions stated to the left, we have:

$$\|\theta_2 - \theta_1\|_2 \leq \sqrt{\frac{L_x(\theta_1)\ d(D_1, D_2)}{\gamma}} + C$$

Assumptions

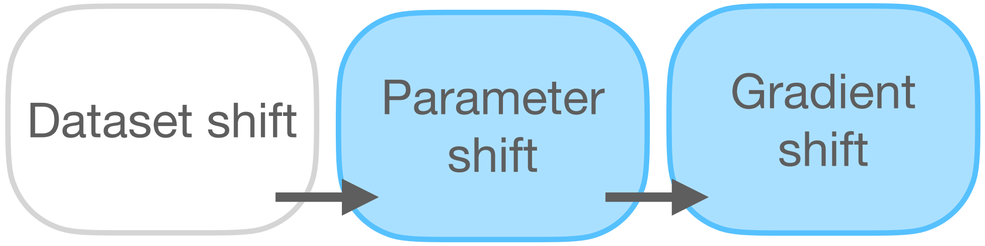
1. We minimize a regularized loss l_{reg} that involves weight decay, i.e., $l_{reg}(\theta) = l(\theta) + \gamma\|\theta\|_2^2$
2. The loss l is locally quadratic
3. The learning algorithm returns a unique minimum θ given a dataset D

C is a small problem-dependent constant

$L_{\mathbf{x}}(\theta_1)$ is the Lipschitz constant of the original model

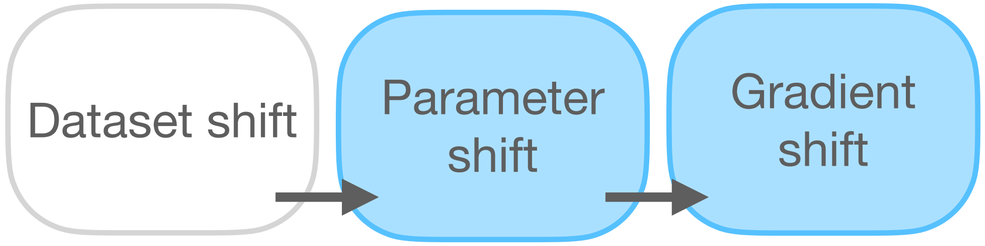
$d(D_1, D_2)$ is the minimum average L2 distance over all possible matchings of samples in D_1 and D_2, i.e.,

$$d(D_1, D_2) = \min_{P(D_2)} \sum_{i=1}^{N} \|x_i - x_i'\|_2$$



Lemma 1: The average gradient shift across all data samples $\mathbf{x} \in D$ is upper bounded by the parameter shift $\|\theta_2 - \theta_1\|_2$ times the gradient-parameter Lipschitz constant $L_{\Theta,D}$, i.e.,

$$\mathbb{E}_{\mathbf{x}\sim D}\|\nabla_{\mathbf{x}} f_{\theta_1}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\theta_2}(\mathbf{x})\|_2 \leq L_{\Theta,D} \times \|\theta_2 - \theta_1\|_2$$



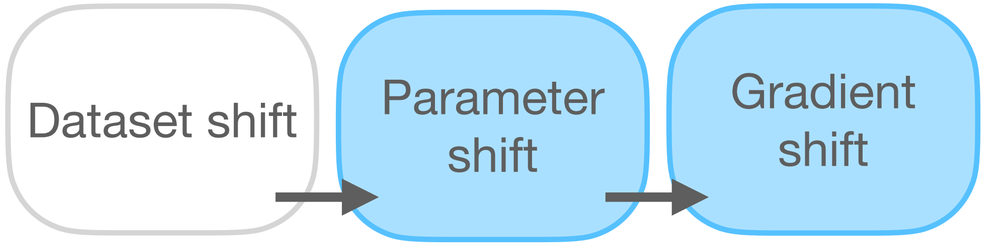
Gradient-parameter Lipschitz constant

We define the gradient-parameter Lipschitz constant, $L_{\Theta,D}$, w.r.t. an input distribution D and a parameter set Θ as

$$L_{\Theta,D} = \mathbb{E}_{\theta\in\Theta}\mathbb{E}_{\mathbf{x}\sim D}\|\nabla_\theta \nabla_{\mathbf{x}} f(\mathbf{x},\theta)\|_2$$

Lemma 1: The average gradient shift across all data samples $\mathbf{x} \in D$ is upper bounded by the parameter shift $\|\theta_2 - \theta_1\|_2$ times the gradient-parameter Lipschitz constant $L_{\Theta,D}$, i.e.,

$$\mathbb{E}_{\mathbf{x}\sim D}\|\nabla_{\mathbf{x}} f_{\theta_1}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\theta_2}(\mathbf{x})\|_2 \leq L_{\Theta,D} \times \|\theta_2 - \theta_1\|_2$$



Gradient-parameter Lipschitz constant

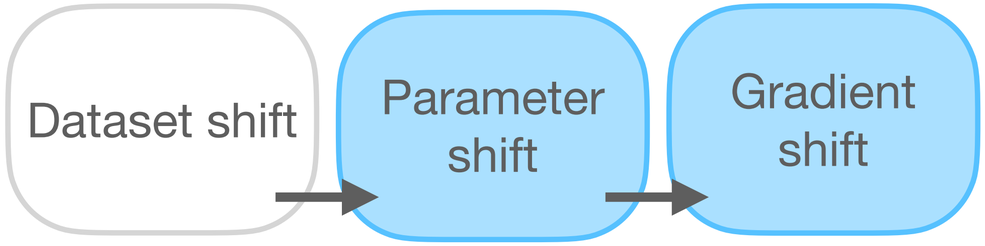
We define the gradient-parameter Lipschitz constant, $L_{\Theta,D}$, w.r.t. an input distribution D and a parameter set Θ as

$$L_{\Theta,D} = \mathbb{E}_{\theta\in\Theta}\mathbb{E}_{\mathbf{x}\sim D}\|\nabla_\theta \nabla_{\mathbf{x}} f(\mathbf{x},\theta)\|_2$$

$$\Theta = \{\lambda\theta_1 + (1-\lambda)\theta_2 \mid \lambda \in [0,1]\}$$

Lemma 1: The average gradient shift across all data samples $\mathbf{x} \in D$ is upper bounded by the parameter shift $\|\theta_2 - \theta_1\|_2$ times the gradient-parameter Lipschitz constant $L_{\Theta,D}$, i.e.,

$$\mathbb{E}_{\mathbf{x}\sim D}\|\nabla_{\mathbf{x}} f_{\theta_1}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\theta_2}(\mathbf{x})\|_2 \leq L_{\Theta,D} \times \|\theta_2 - \theta_1\|_2$$



Gradient-parameter Lipschitz constant

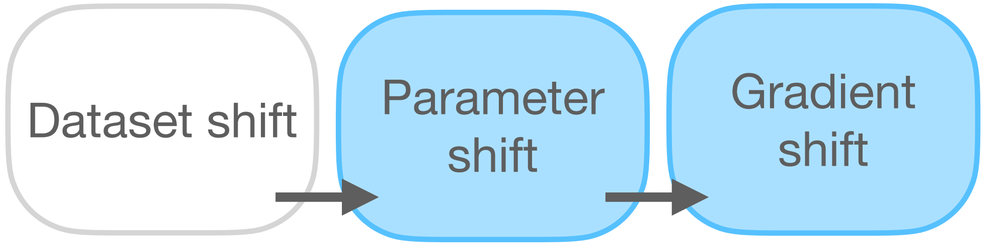
We define the gradient-parameter Lipschitz constant, $L_{\Theta,D}$, w.r.t. an input distribution D and a parameter set Θ as

$$L_{\Theta,D} = \mathbb{E}_{\theta\in\Theta}\mathbb{E}_{\mathbf{x}\sim D}\|\nabla_\theta \nabla_{\mathbf{x}} f(\mathbf{x},\theta)\|_2$$

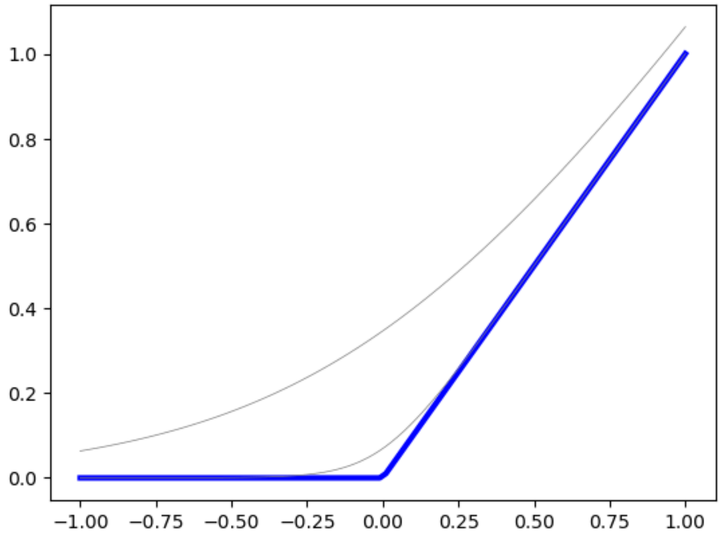
$$\Theta = \{\lambda\theta_1 + (1-\lambda)\theta_2 \mid \lambda \in [0,1]\}$$

Lemma 1: The average gradient shift across all data samples $\mathbf{x} \in D$ is upper bounded by the parameter shift $\|\theta_2 - \theta_1\|_2$ times the gradient-parameter Lipschitz constant $L_{\Theta,D}$, i.e.,

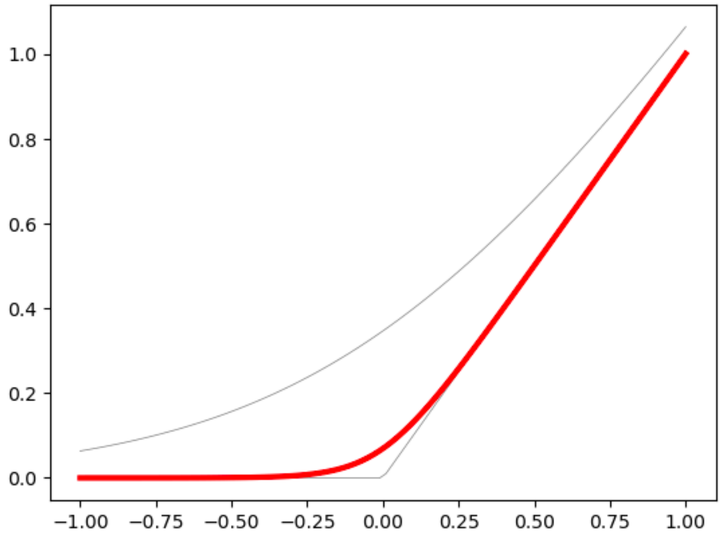
$$\mathbb{E}_{\mathbf{x}\sim D}\|\nabla_{\mathbf{x}} f_{\theta_1}(\mathbf{x}) - \nabla_{\mathbf{x}} f_{\theta_2}(\mathbf{x})\|_2 \leq L_{\Theta,D} \times \|\theta_2 - \theta_1\|_2$$



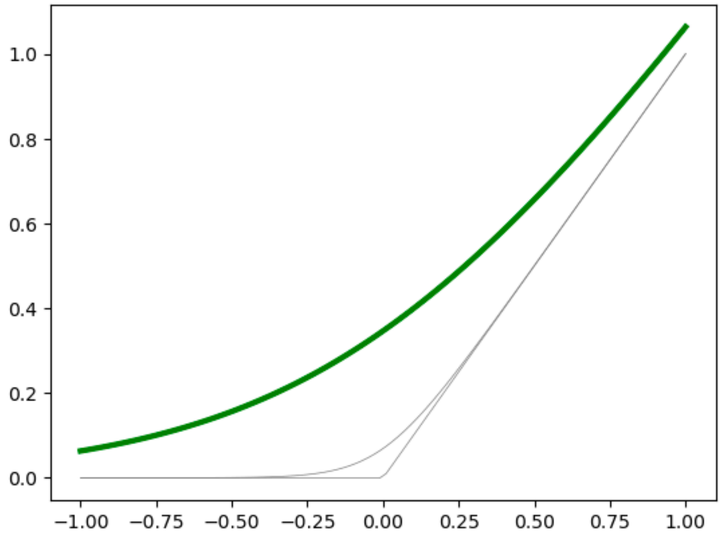
Intuition: The Gradient-Parameter Lipschitz Constant relates to the model's curvature



ReLU:
$L_{\Theta,D}$ is **infinity**

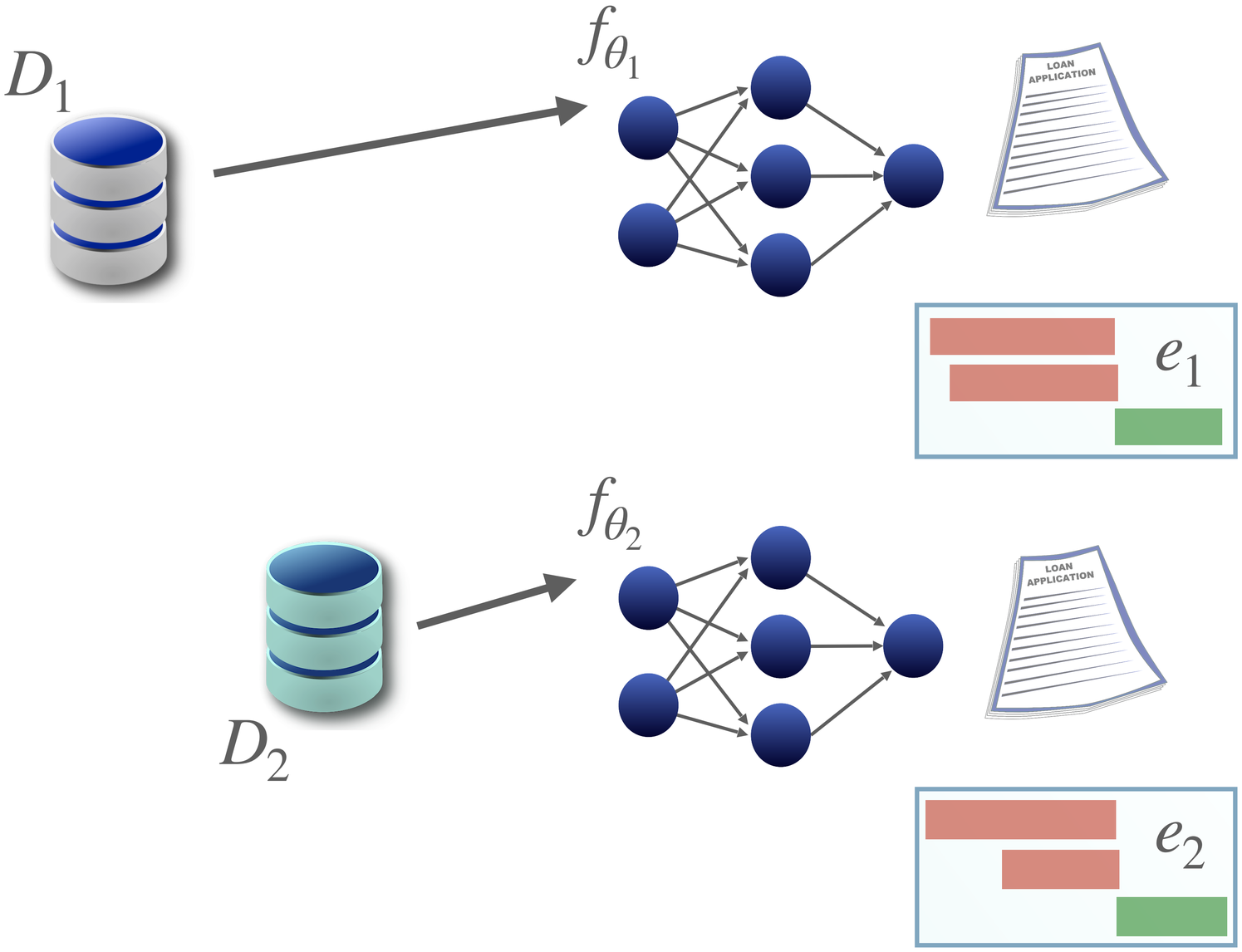


Softplus ($\beta = 10$):
$L_{\Theta,D}$ is **large**



Softplus ($\beta = 2$):
$L_{\Theta,D}$ is **small**

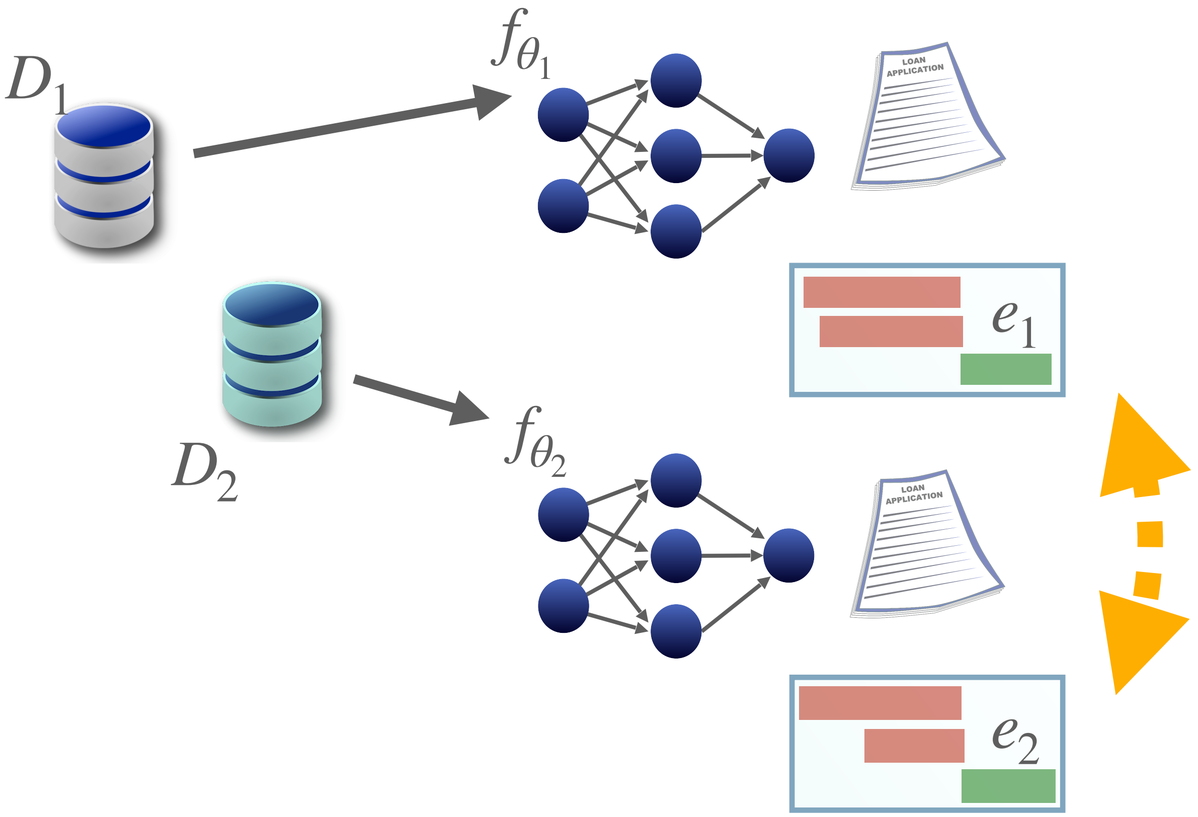
Gradient shift is affected by…



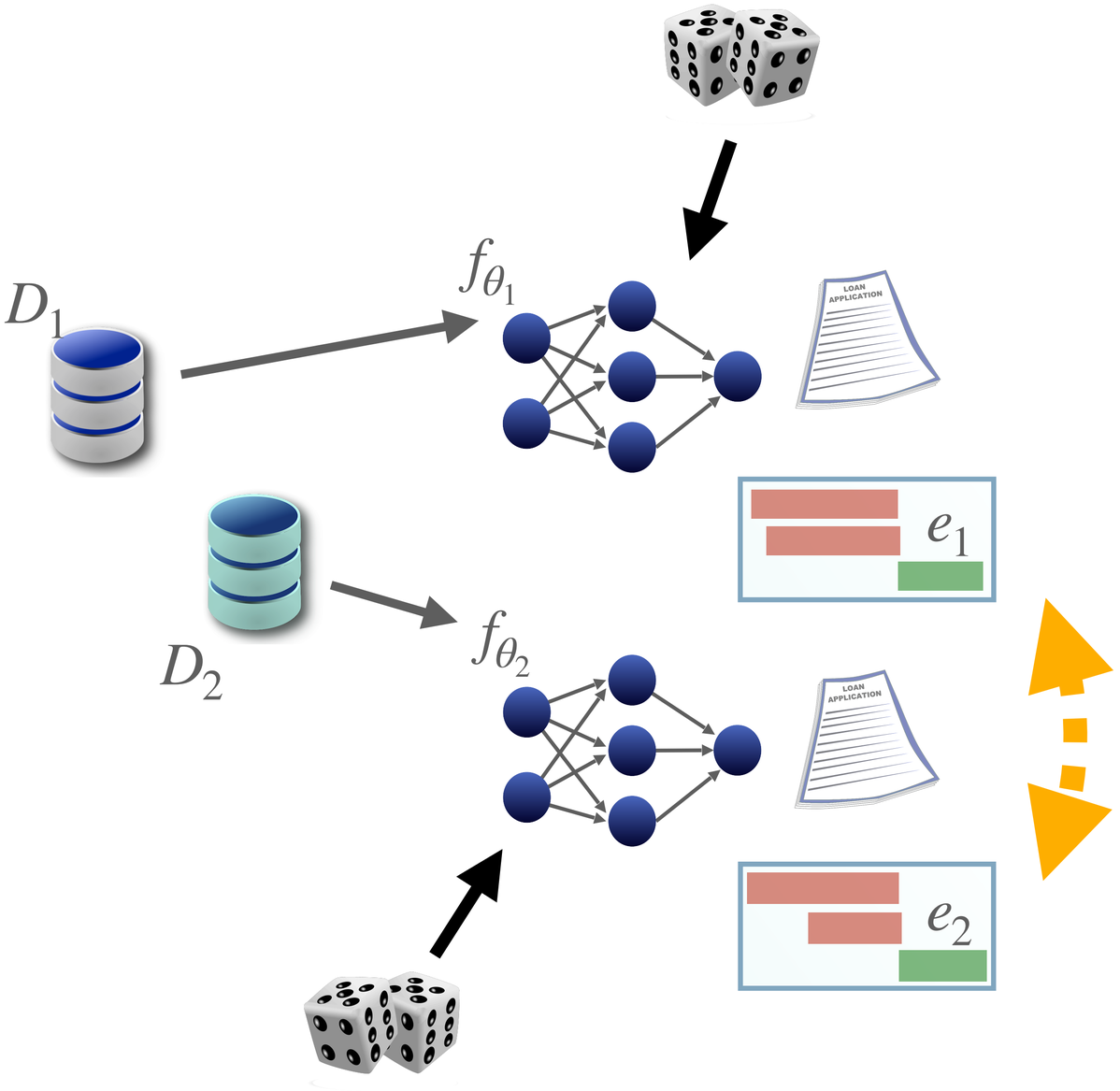
1. Dataset shift size

2. Weight decay parameter

3. Smoothness of the activation function



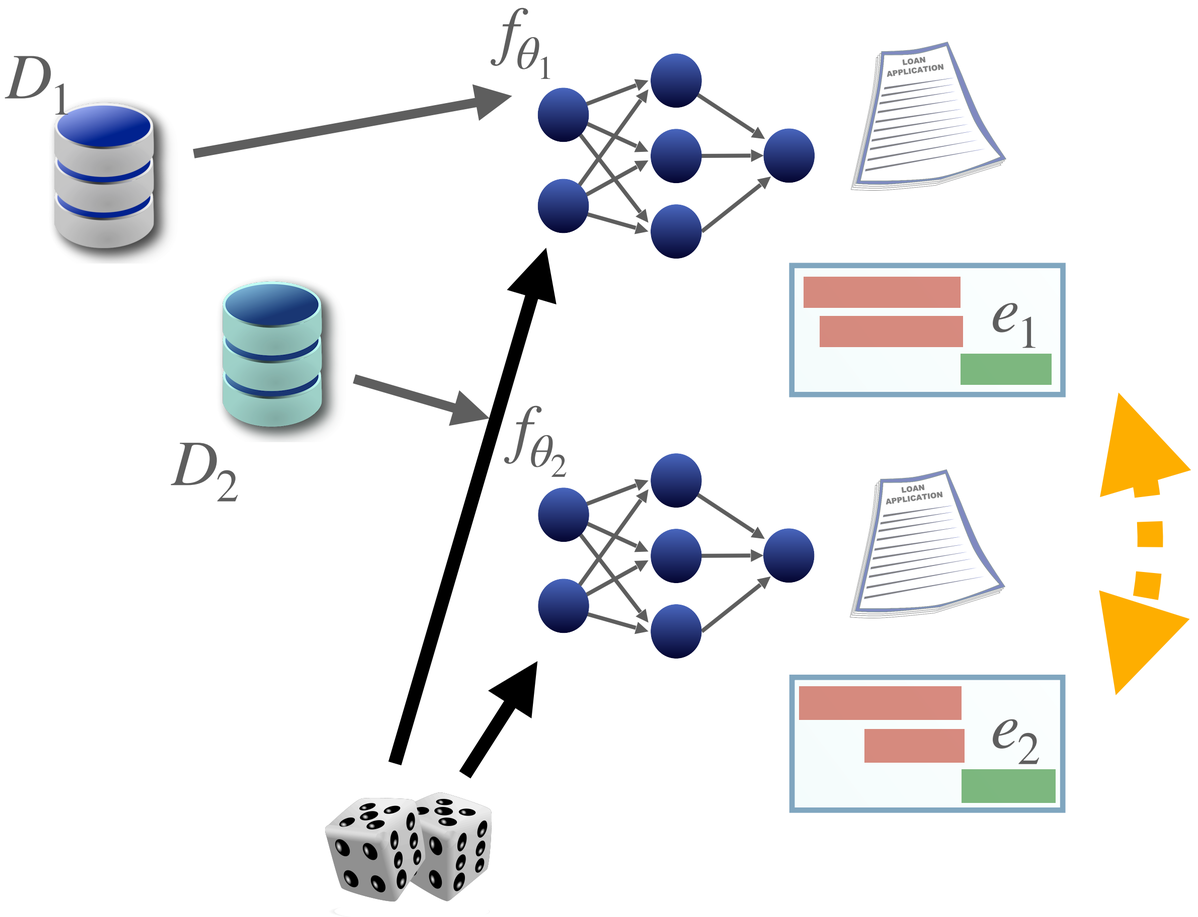
Practical considerations



Practical considerations

Model multiplicity

Random model weight initializations for both f_{θ_1} and f_{θ_2} will lead to explanation instability

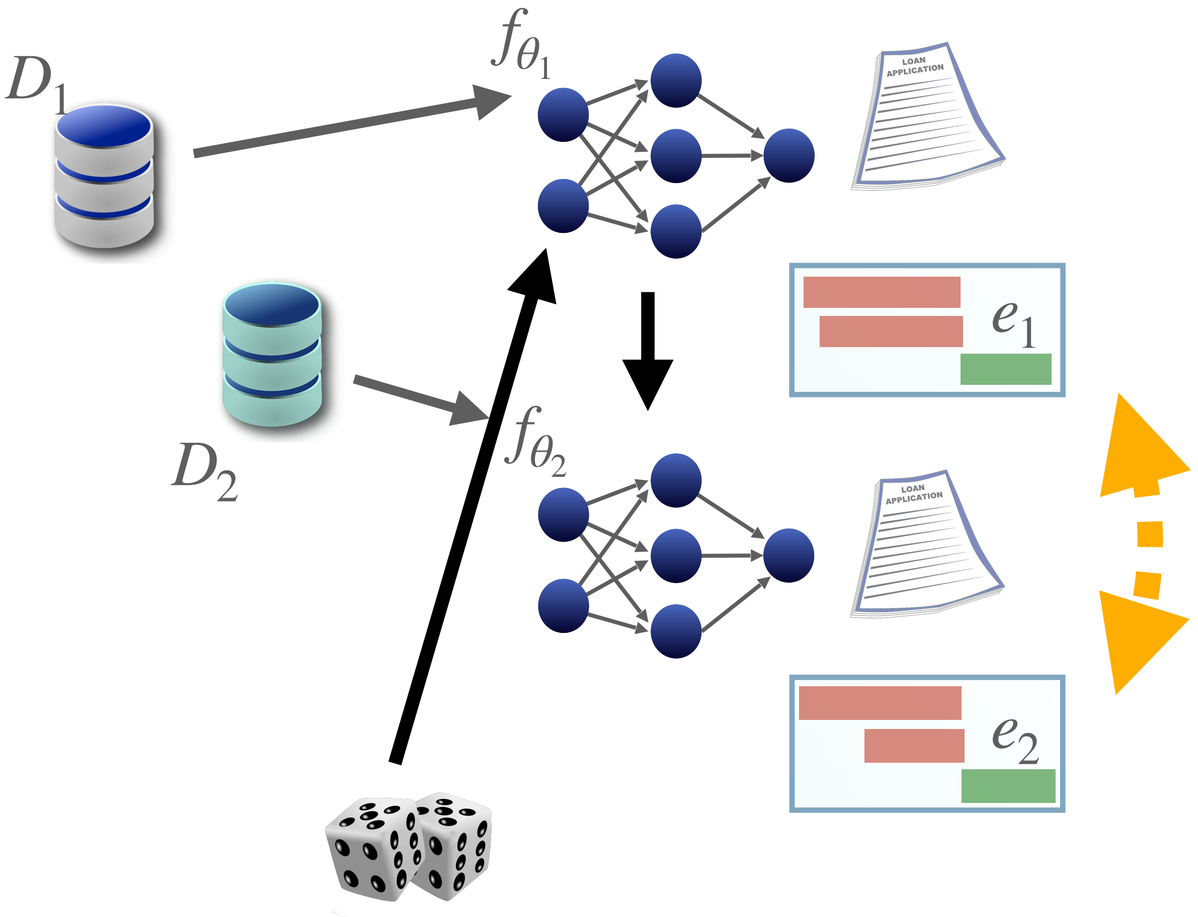


Practical considerations

Model multiplicity

Random model weight initializations for both f_{θ_1} and f_{θ_2} will lead to explanation instability

Solution: use the same random initialization, or *finetune* f_{θ_2} starting with f_{θ_1}'s weights

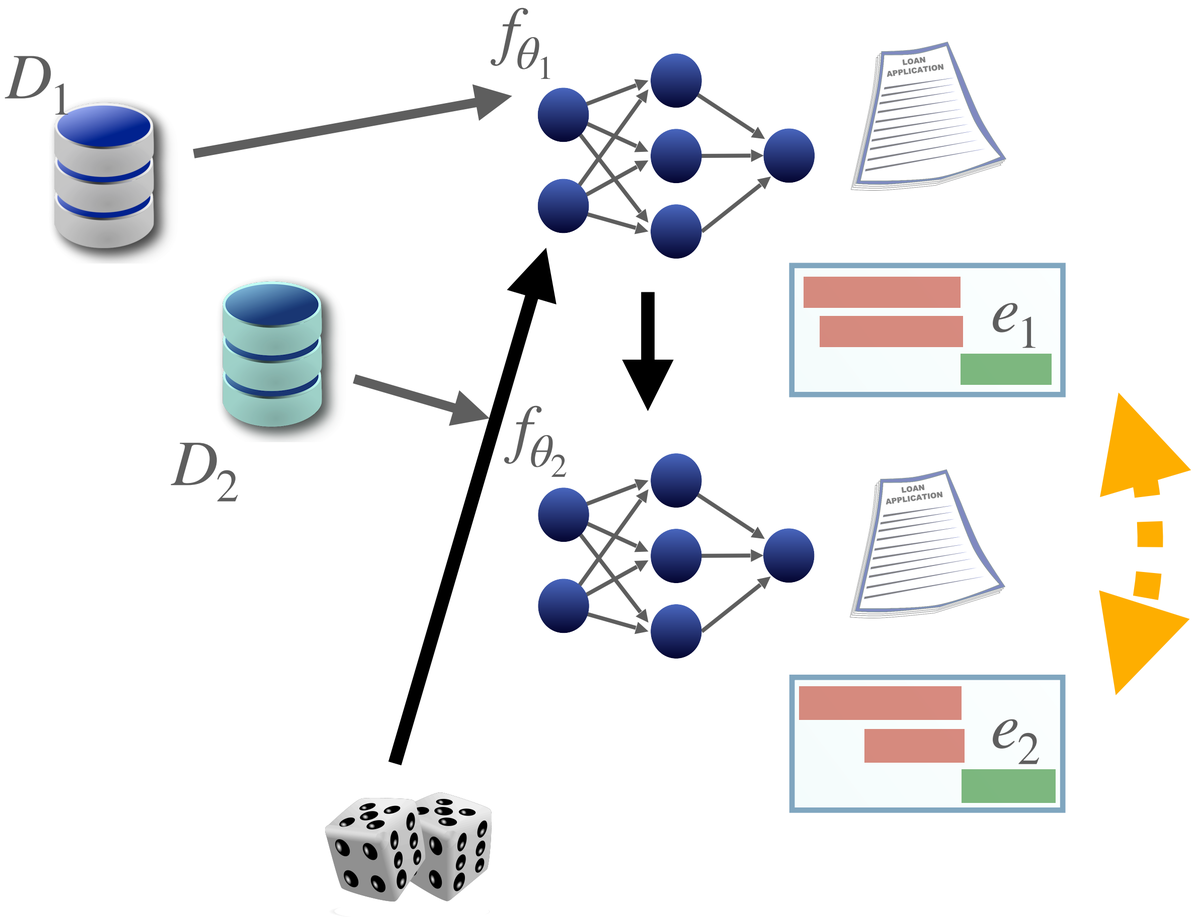


Practical considerations

Model multiplicity

Random model weight initializations for both f_{θ_1} and f_{θ_2} will lead to explanation instability

Solution: use the same random initialization, or *finetune* f_{θ_2} starting with f_{θ_1}'s weights



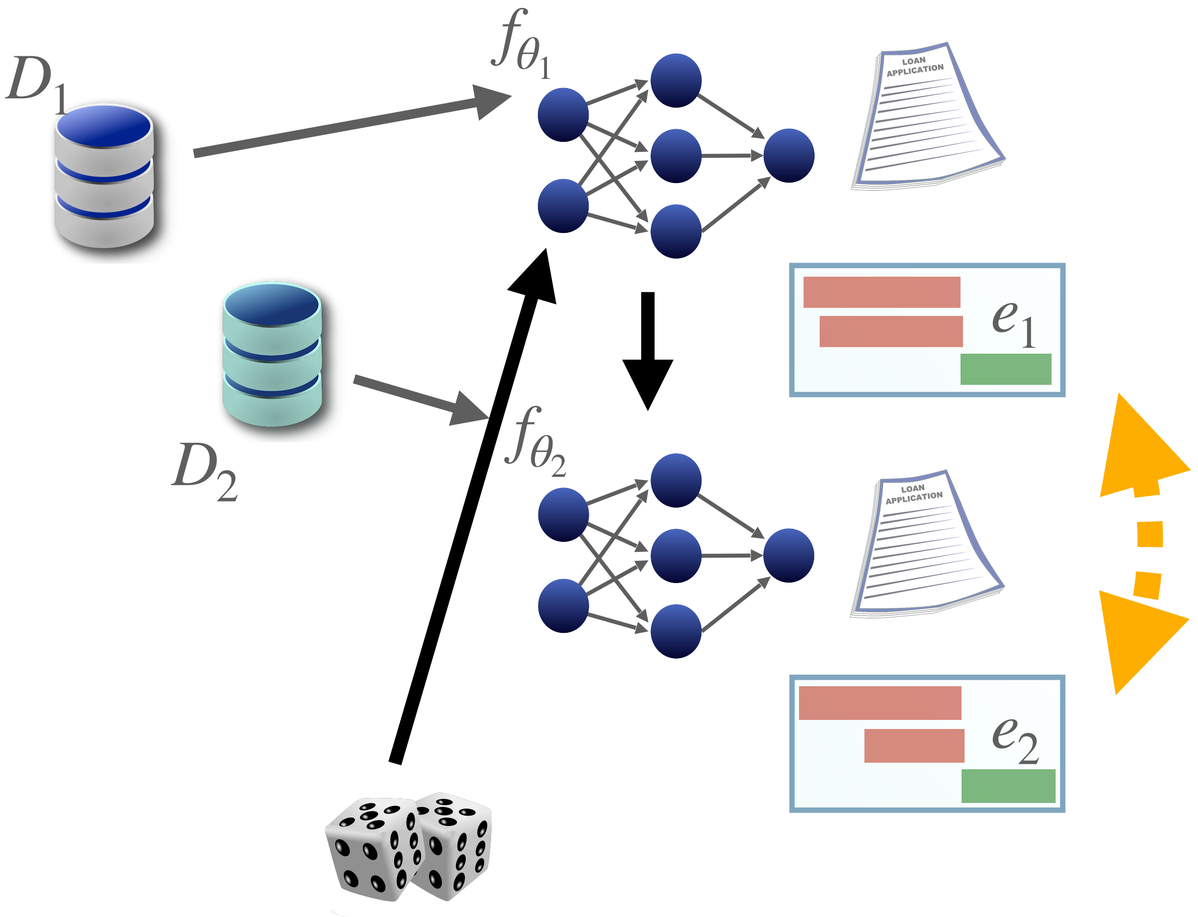
Practical considerations

Model multiplicity

Random model weight initializations for both f_{θ_1} and f_{θ_2} will lead to explanation instability

Solution: use the same random initialization, or *finetune* f_{θ_2} starting with f_{θ_1}'s weights

Reaching an exact minimum of f



Practical considerations

Model multiplicity

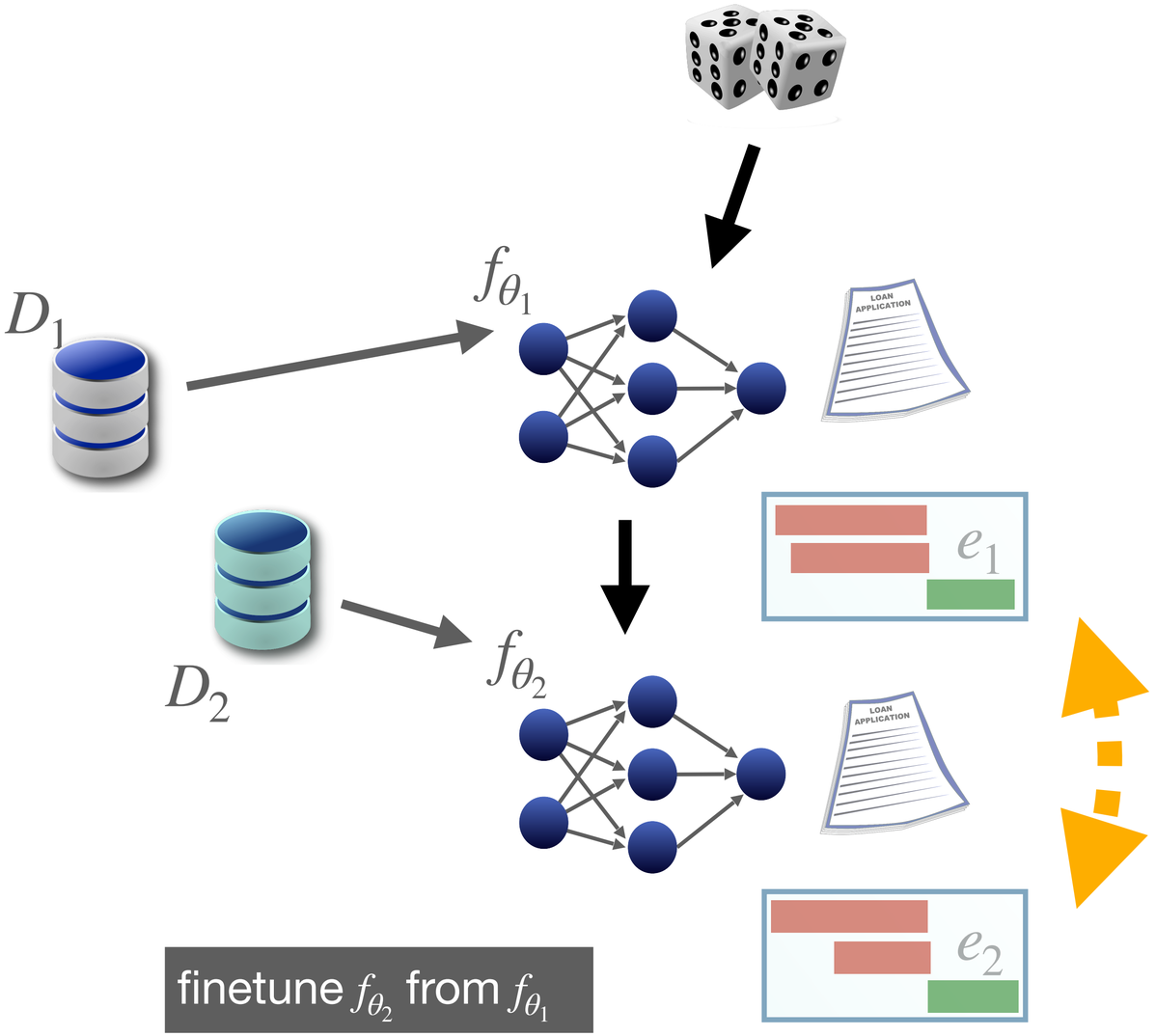
Random model weight initializations for both f_{θ_1} and f_{θ_2} will lead to explanation instability

Solution: use the same random initialization, or *finetune* f_{θ_2} starting with f_{θ_1}'s weights

Reaching an exact minimum of f

Not likely in practice (e.g., due to SGD)

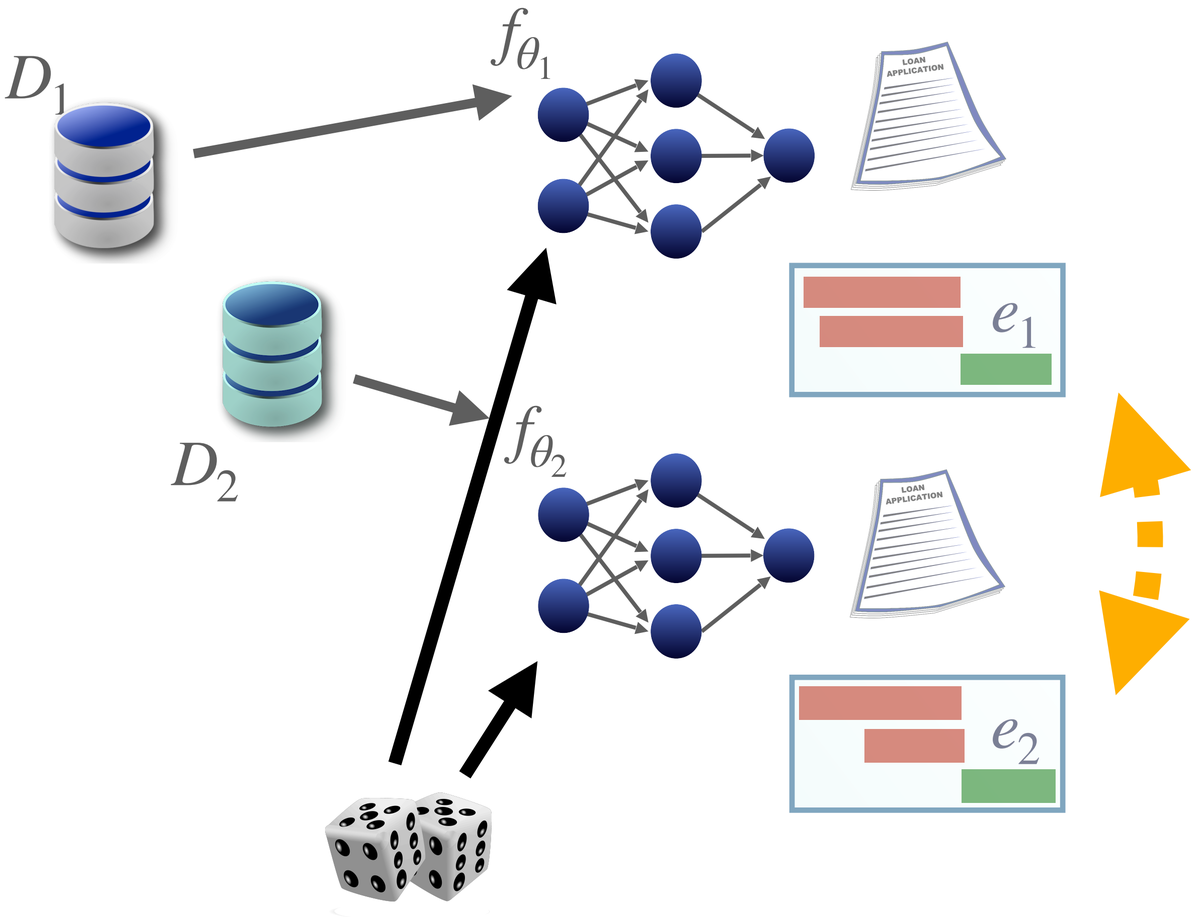
Solution: train for many epochs with a small learning rate



Four sets of experiments

1. **Validate theoretical findings**
2. Extend to realistic training setups
3. Extend to complex explanations
4. Probe the importance of other hyperparameters

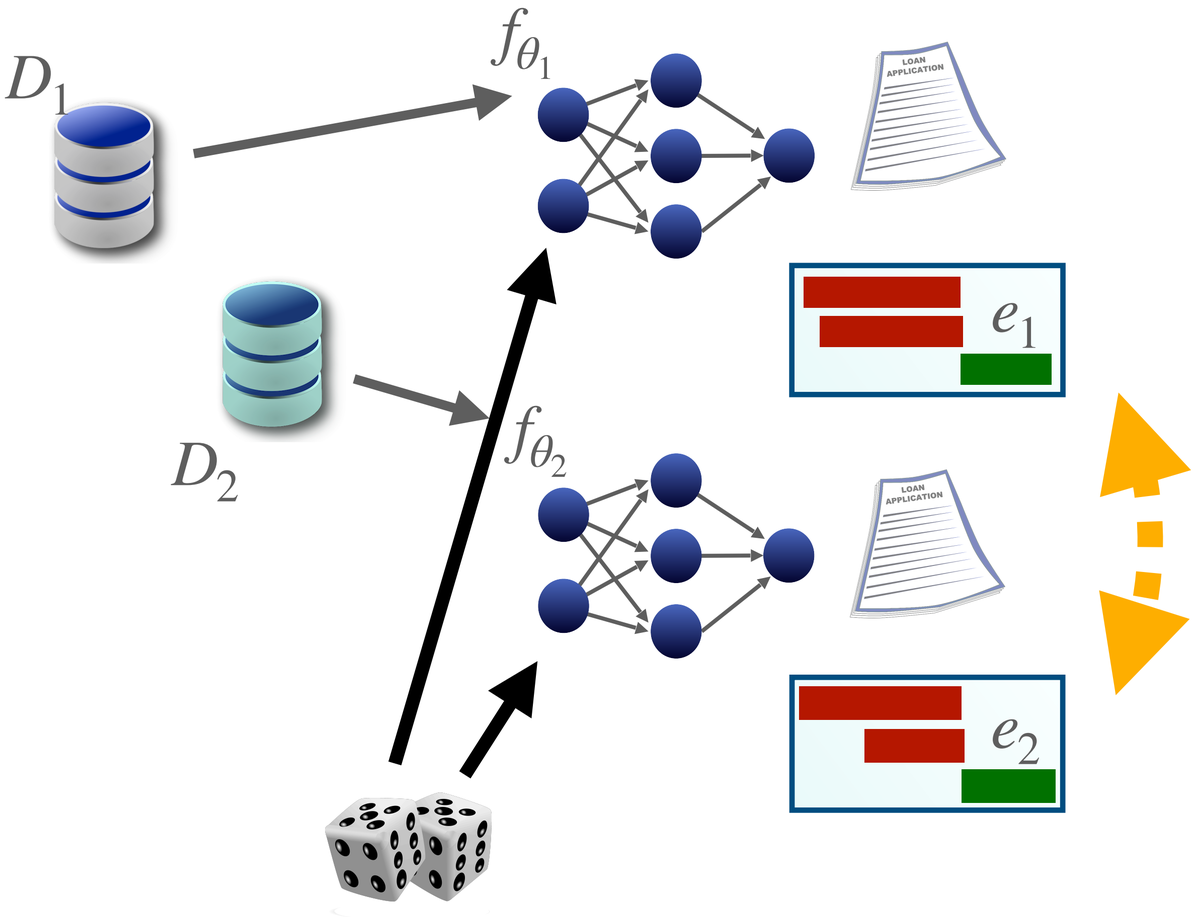
Four sets of experiments



Same random initialization for f_{θ_1} and f_{θ_2}

1. Validate theoretical findings
2. **Extend to realistic training setups**
3. Extend to complex explanations
4. Probe the importance of other hyperparameters

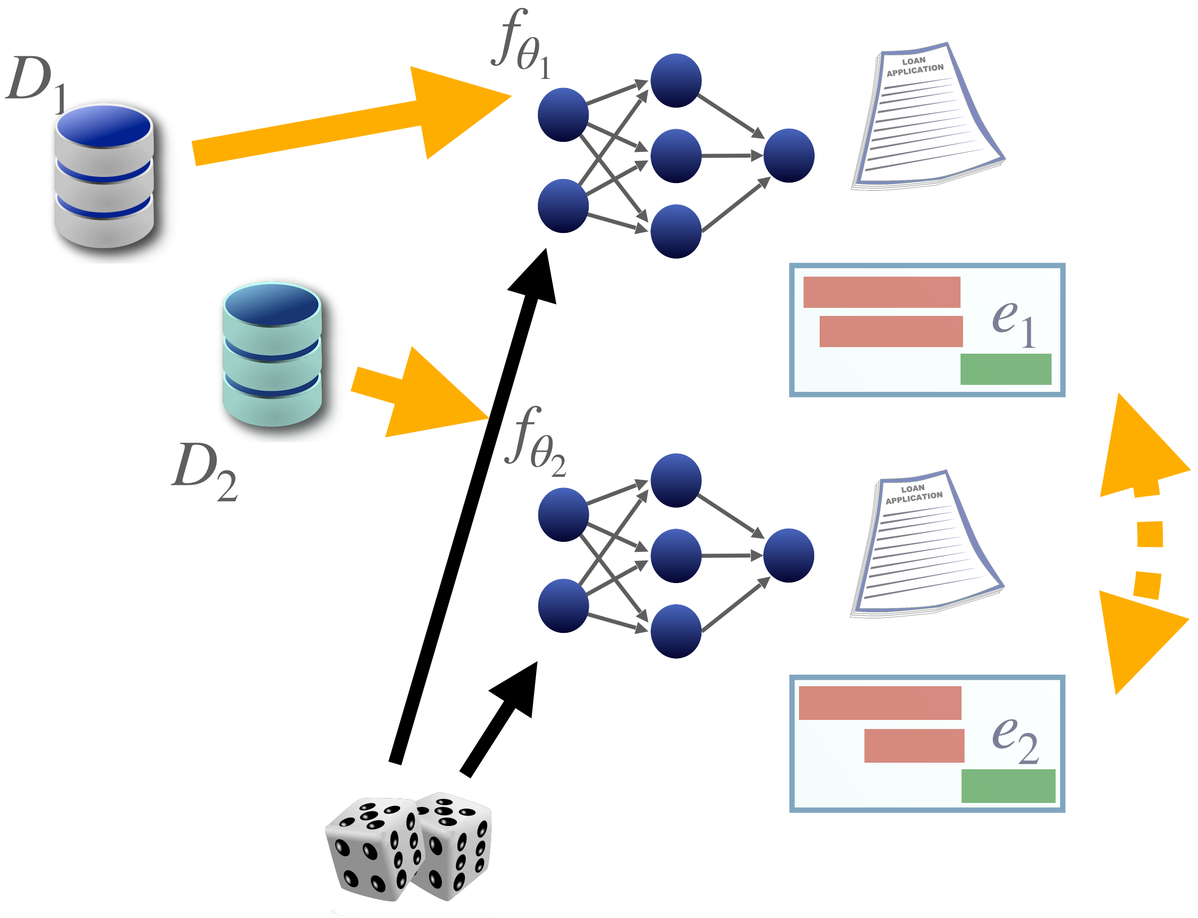
Four sets of experiments



1. Validate theoretical findings
2. Extend to realistic training setups
3. **Extend to complex explanations**
4. Probe the importance of other hyperparameters

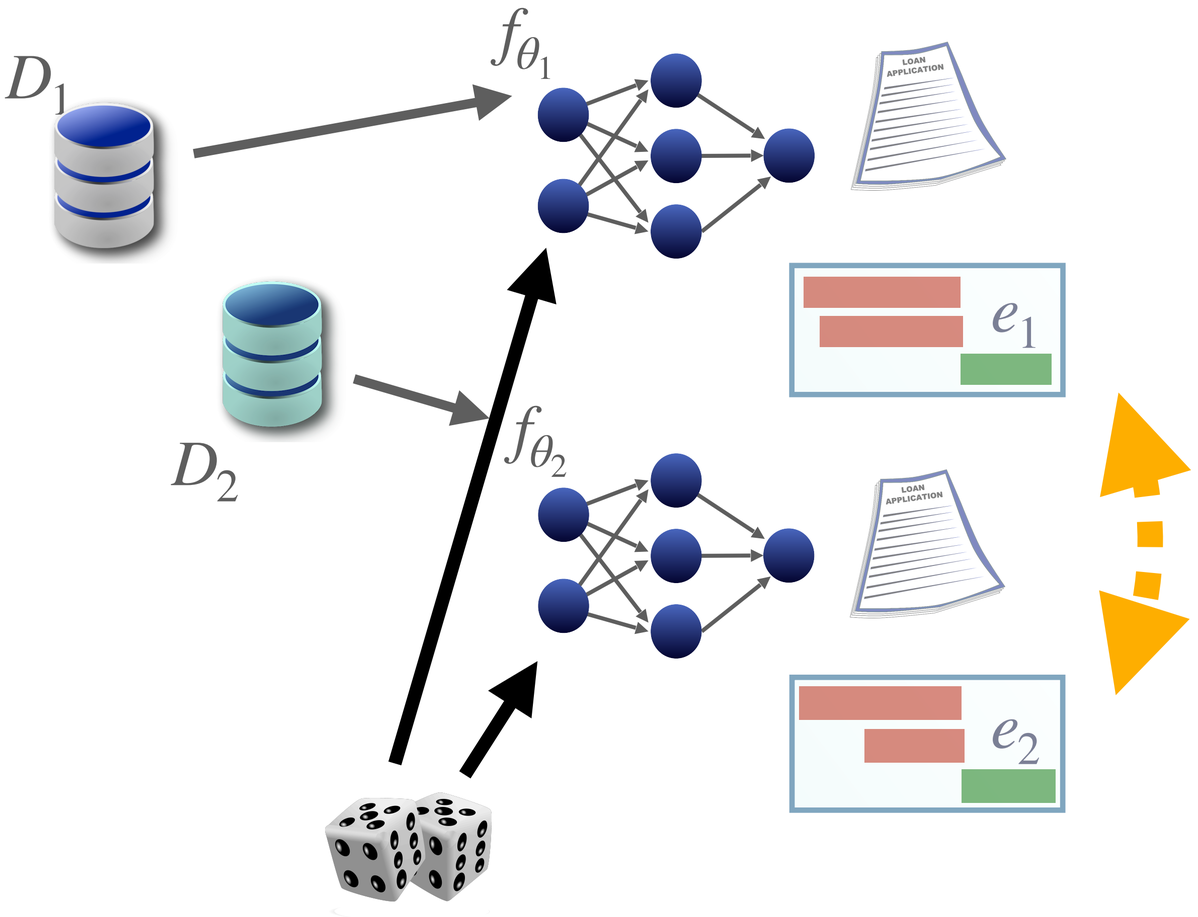
Top-K feature explanations, not gradients

Vary other training choices



Four sets of experiments

1. Validate theoretical findings
2. Extend to realistic training setups
3. Extend to complex explanations
4. **Probe the importance of other hyperparameters**



What are  and  ?

Three datasets

1. **Adult**
2. **HELOC**
3. **WHO**

Two shift paradigms

1. **Gaussian** noise
2. Real-world

   temporal

Training setup	Dataset shift	Interventions
Strict assumptions	Gaussian	Dataset shift & weight decay

Training setup

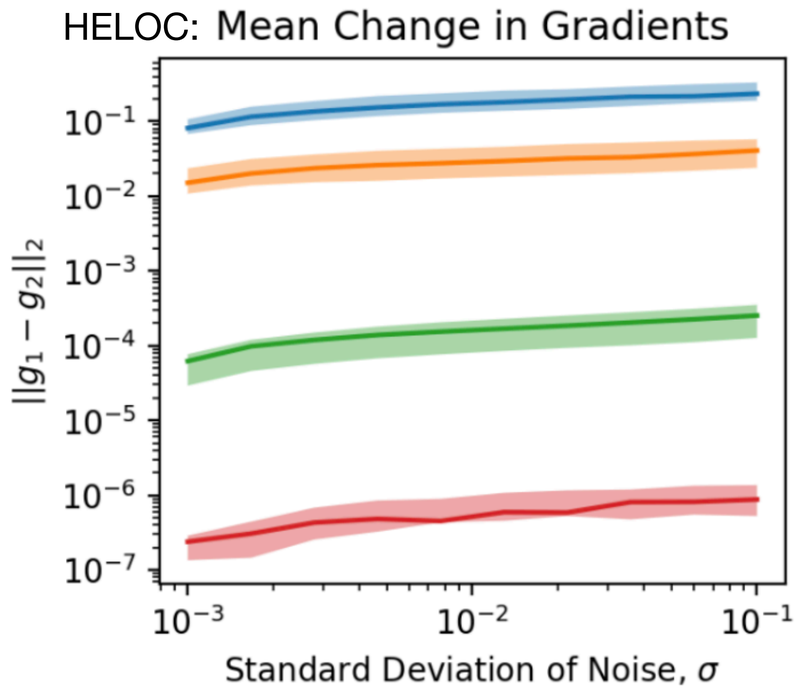
Strict assumptions

Dataset shift

Gaussian

Interventions

Dataset shift & weight decay



Weight decay: $\gamma = 0$ $\gamma = 0.001$ $\gamma = 0.005$ $\gamma = 0.01$

Training setup

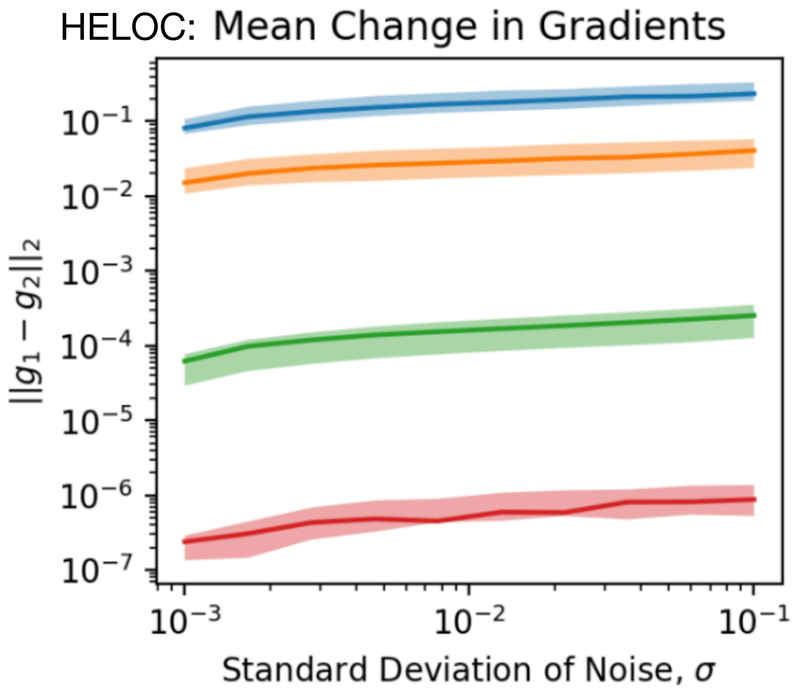
Strict assumptions

Dataset shift

Gaussian

Interventions

Dataset shift & weight decay



1. As dataset shift increases, gradient shift increases

Weight decay: $\gamma = 0$ $\gamma = 0.001$ $\gamma = 0.005$ $\gamma = 0.01$

Training setup

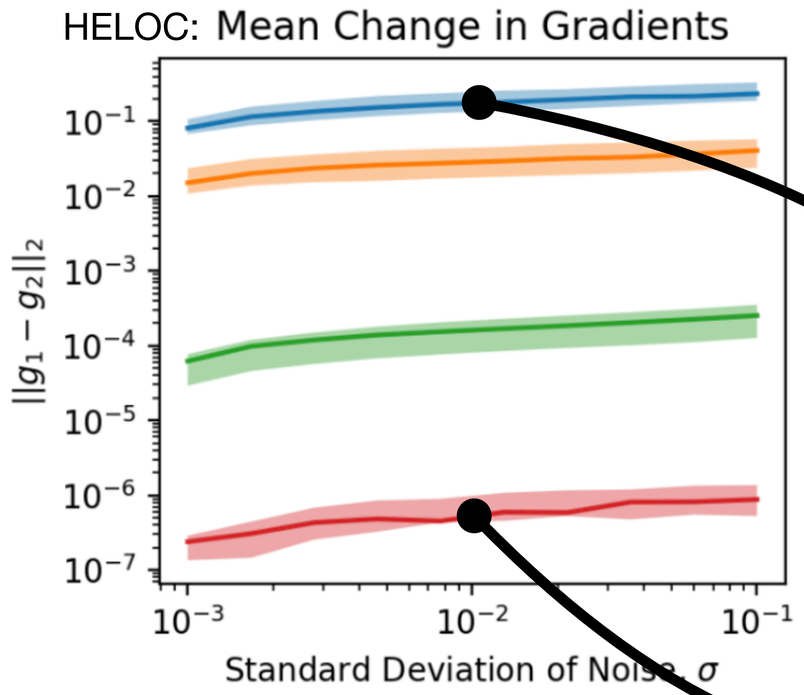
Strict assumptions

Dataset shift

Gaussian

Interventions

Dataset shift & weight decay



1. As dataset shift increases, gradient shift increases
2. Smaller values of weight decay yield larger gradient shifts

$\gamma = 0$ has avg. gradient shift of 10^{-1}

$\gamma = 0.01$ has avg. gradient shift of 10^{-6}

Weight decay: $\gamma = 0$ $\gamma = 0.001$ $\gamma = 0.005$ $\gamma = 0.01$

Training setup

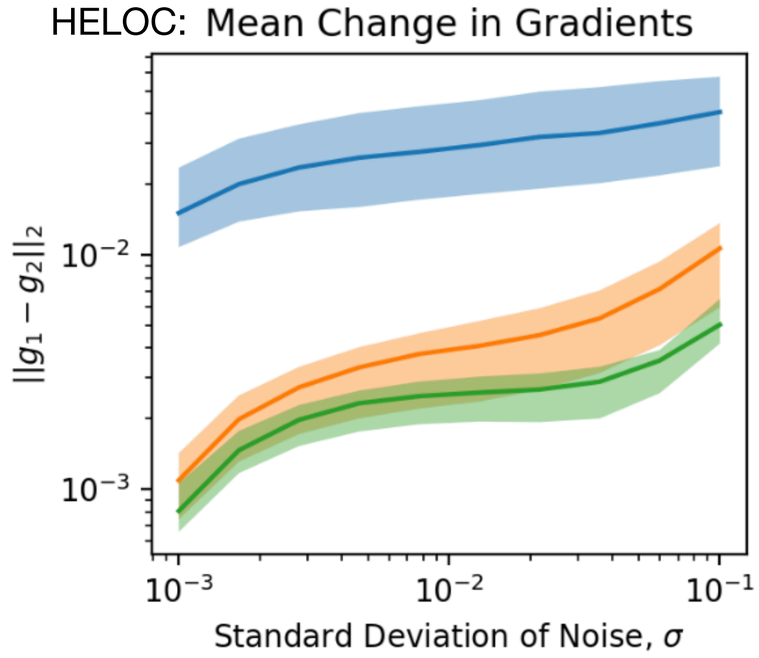
Strict assumptions

Dataset shift

Gaussian

Interventions

Dataset shift & smoothness



Activation Function: Relu Softplus (β=5) Softplus (β=2)

Training setup

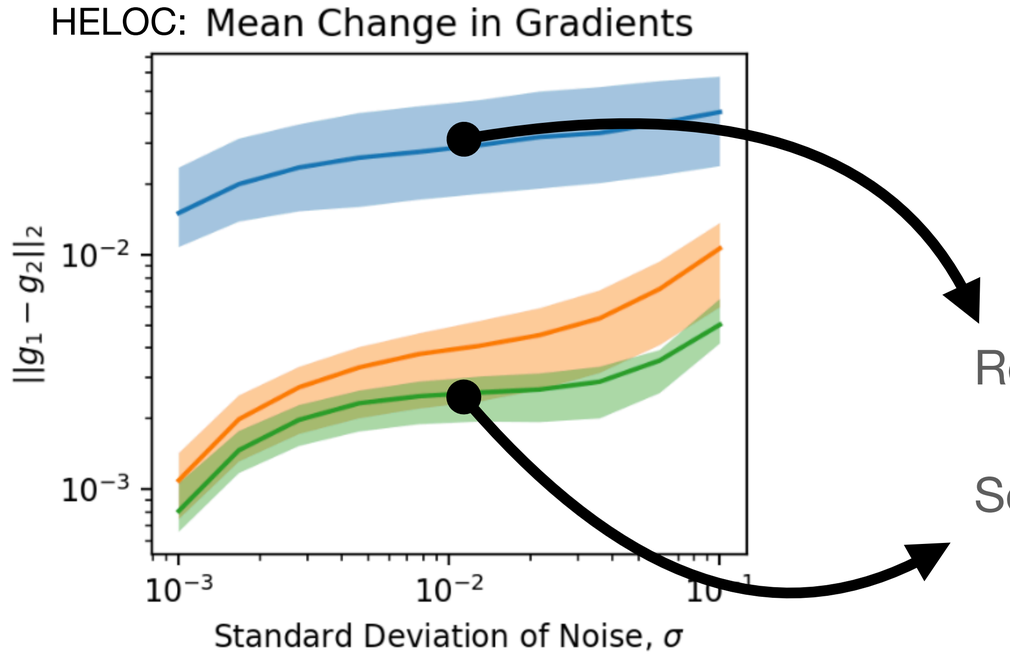
Strict assumptions

Dataset shift

Gaussian

Interventions

Dataset shift & smoothness



Larger β values yield larger gradient shifts

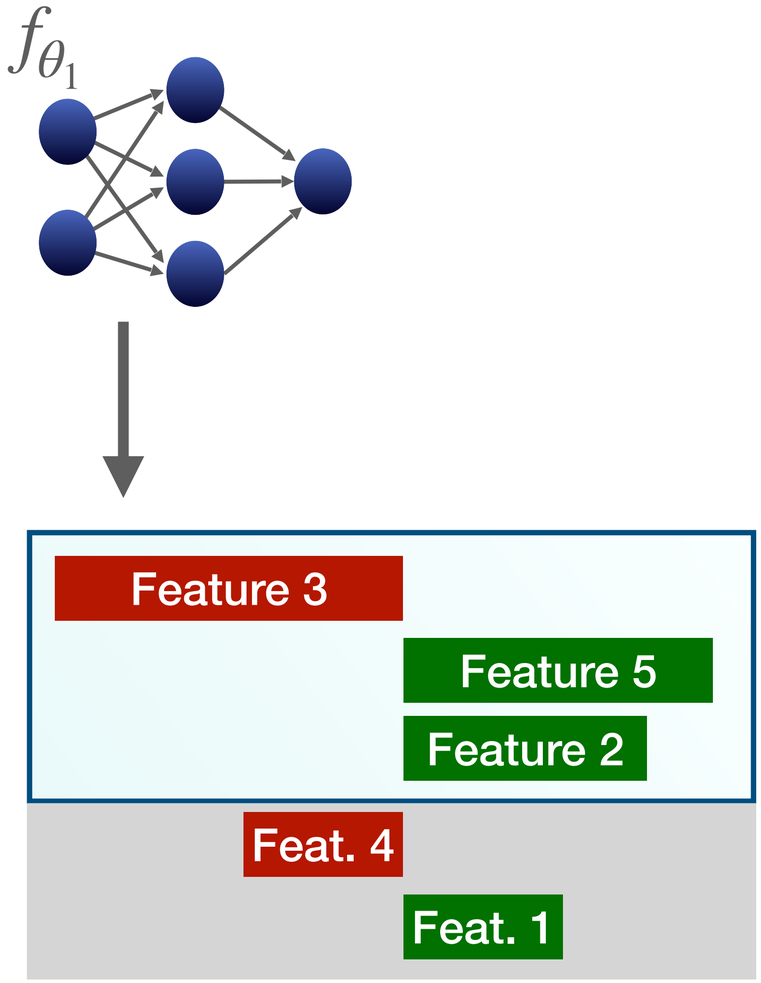
ReLU ($\beta = \infty$) has avg. gradient shift of 0.03

Softplus ($\beta = 2$) has avg. gradient shift of 0.002

Activation Function: Relu Softplus (β=5) Softplus (β=2)

Training setup

Complex explanations

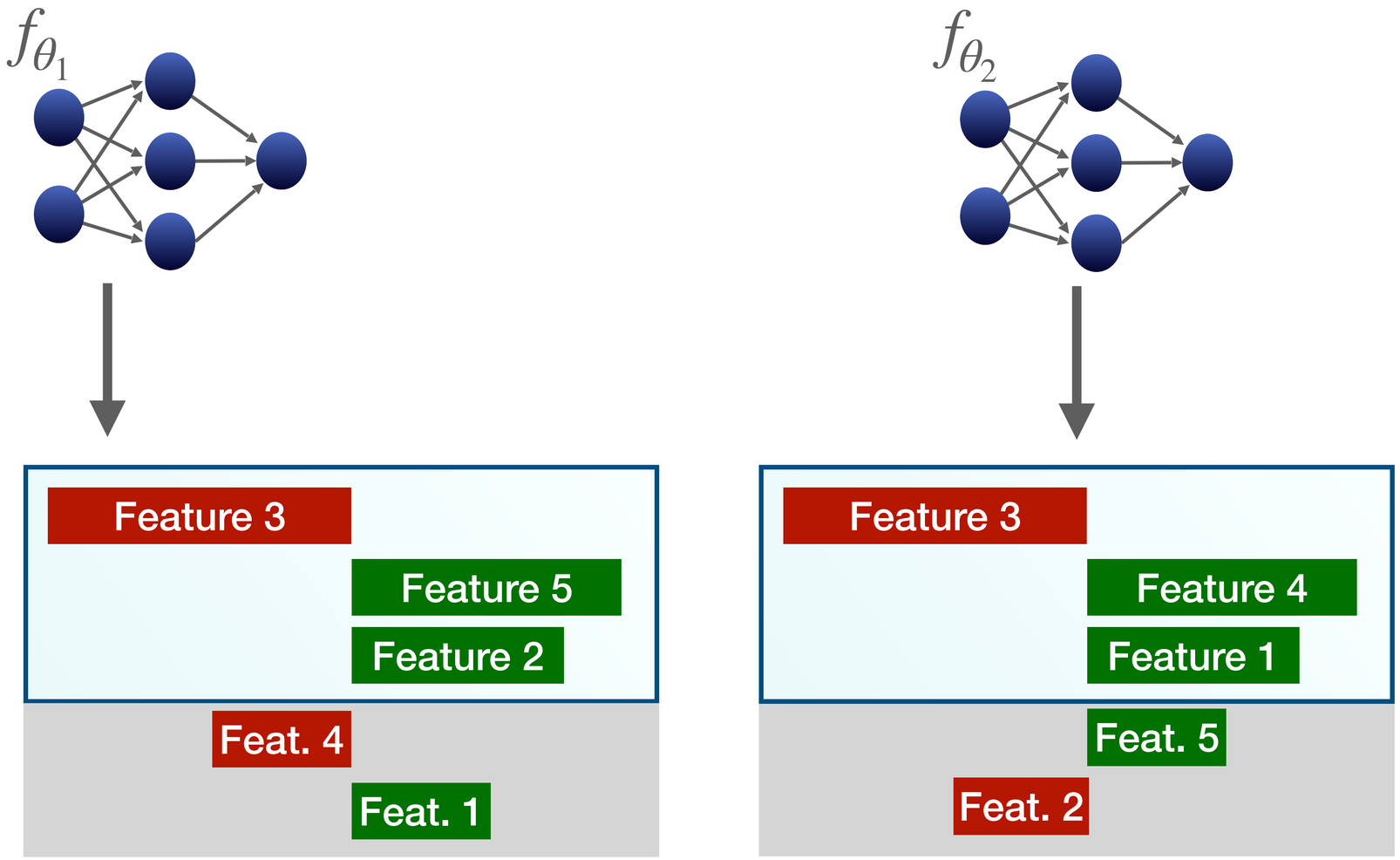


Top-K feature explanations: Sign agreement (SA)

Training setup

Complex explanations

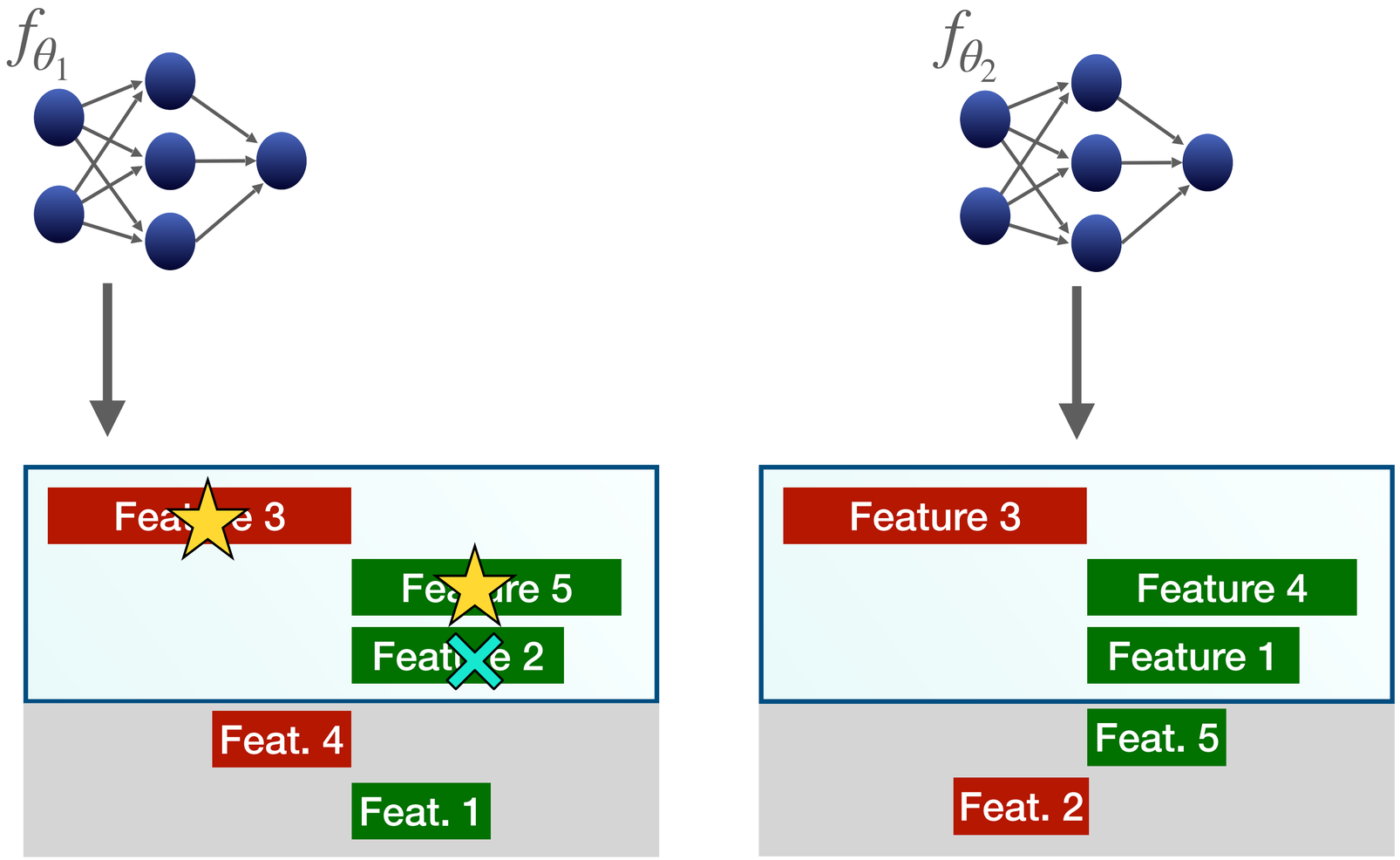
Top-K feature explanations: Sign agreement (SA)



Training setup

Complex explanations

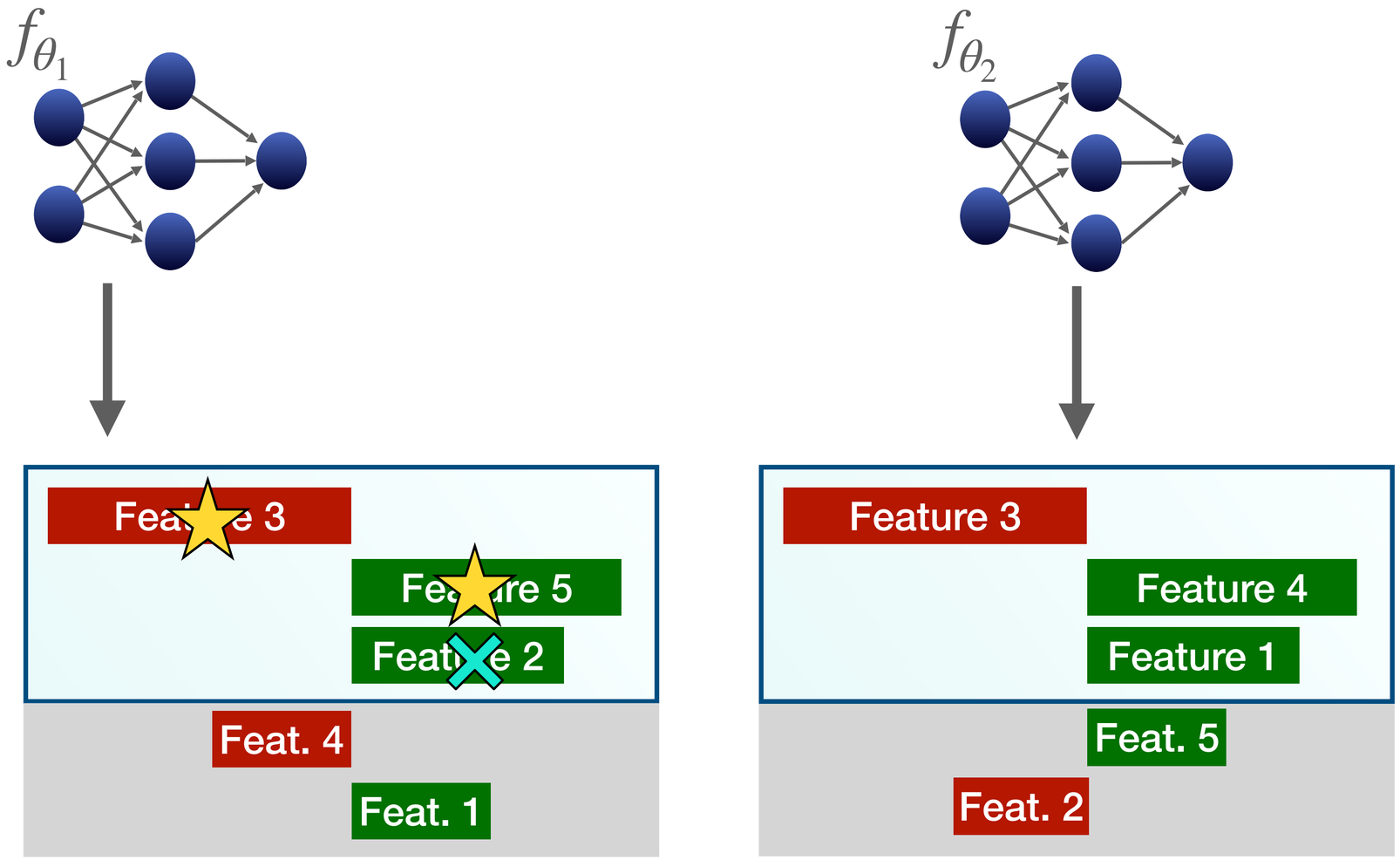
Top-K feature explanations: Sign agreement (SA)



Training setup

Complex explanations

Top-K feature explanations: Sign agreement (SA)



$$\text{SA} = \frac{2}{3}$$

Training setup

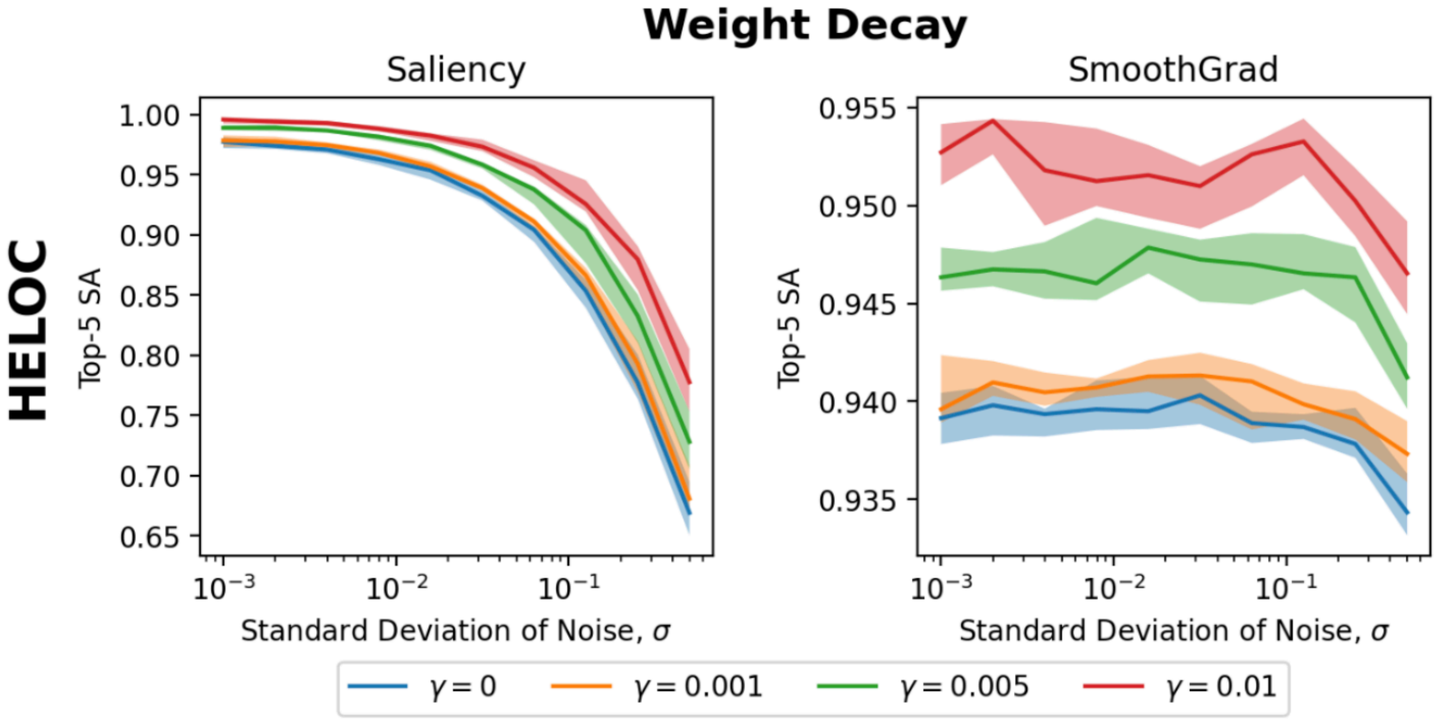
Complex explanations

Dataset shift

Gaussian

Interventions

Dataset shift & weight decay & explanation technique



Training setup

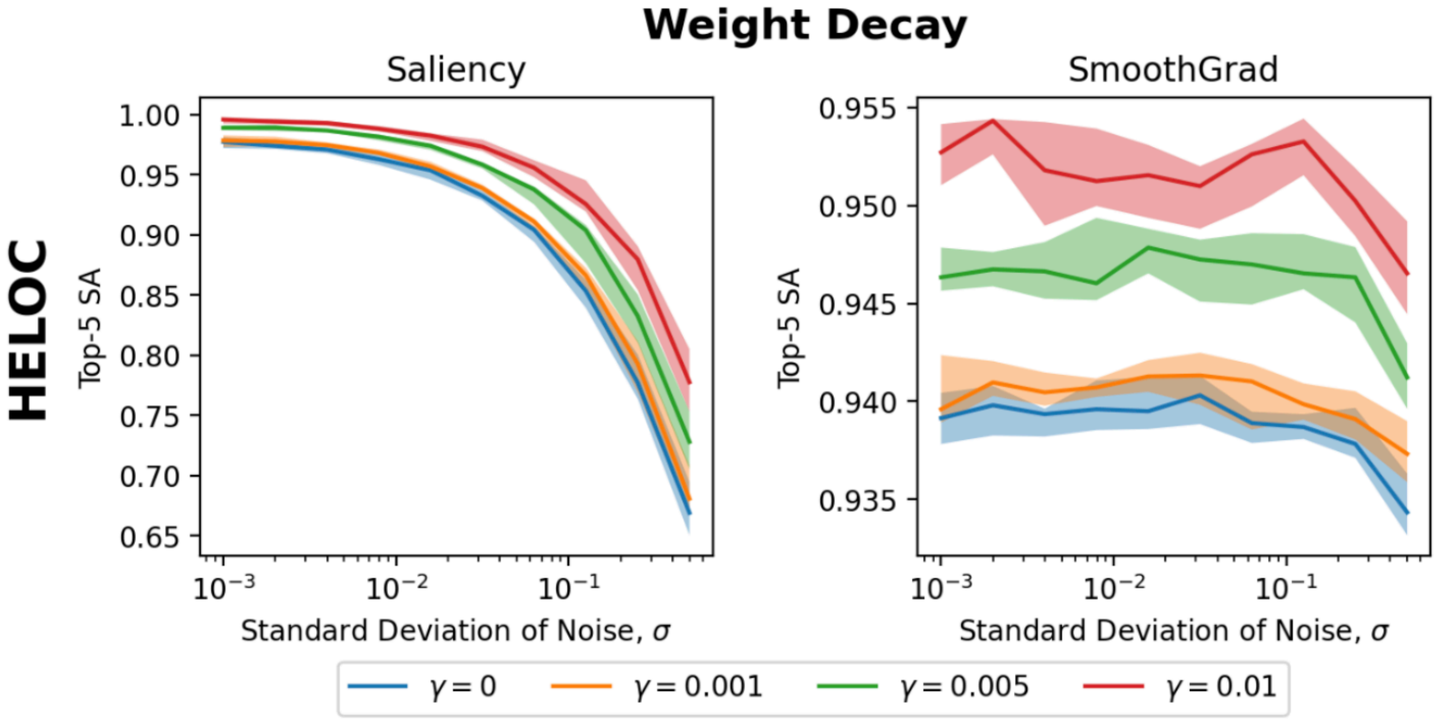
Complex explanations

Dataset shift

Gaussian

Interventions

Dataset shift & weight decay & explanation technique



1. As dataset shift increases, **explanation similarity** decreases

Training setup

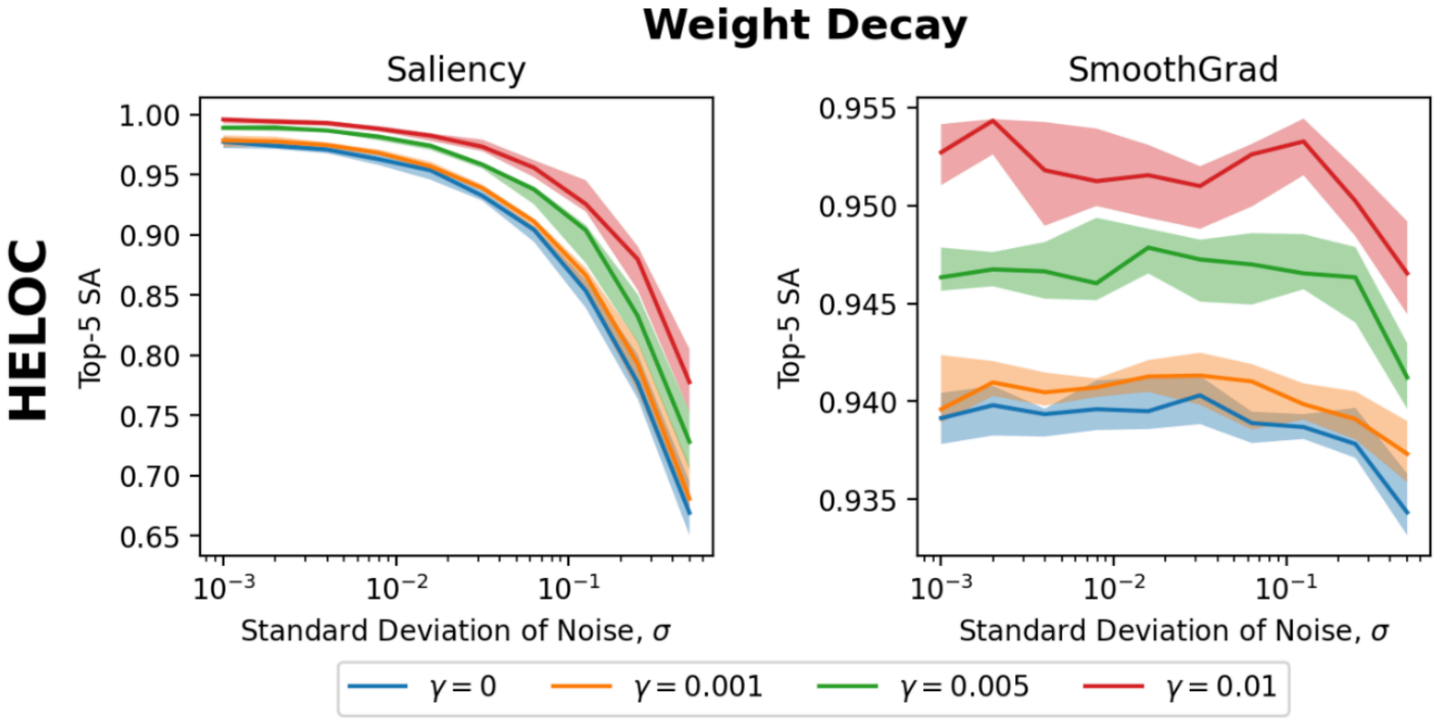
Complex explanations

Dataset shift

Gaussian

Interventions

Dataset shift & weight decay & explanation technique



1. As dataset shift increases, **explanation similarity** decreases
2. Weight decay matters a little

Training setup

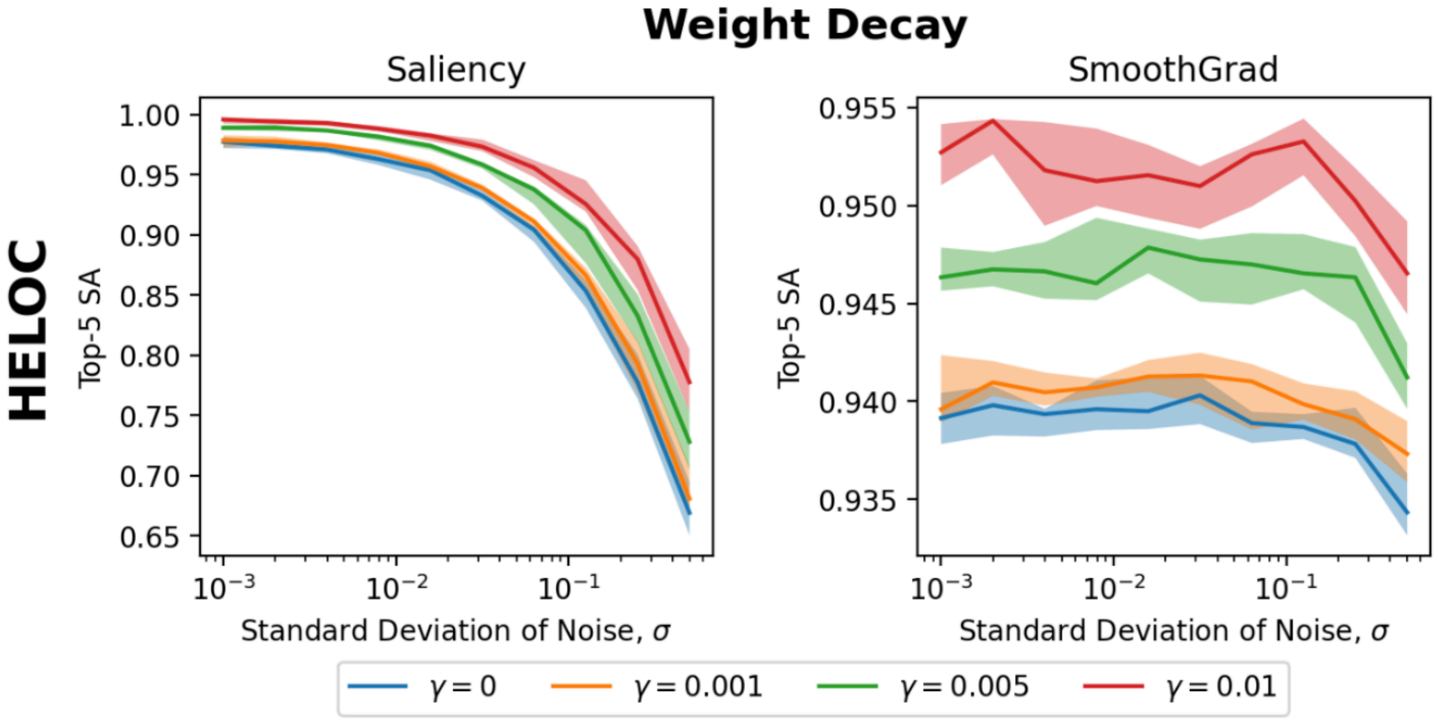
Complex explanations

Dataset shift

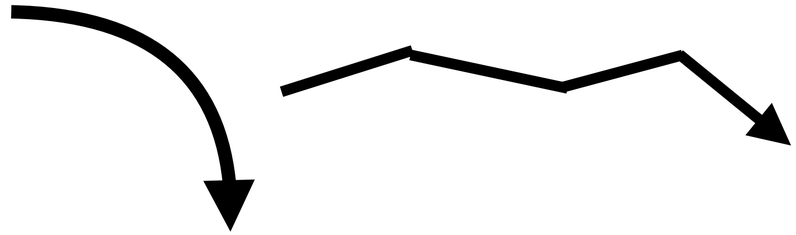
Gaussian

Interventions

Dataset shift & weight decay & explanation technique



1. As dataset shift increases, **explanation similarity** decreases
2. Weight decay matters a little
3. SmoothGrad is noisier than Saliency



Training setup

Complex explanations

Dataset shift

Temporal

Interventions

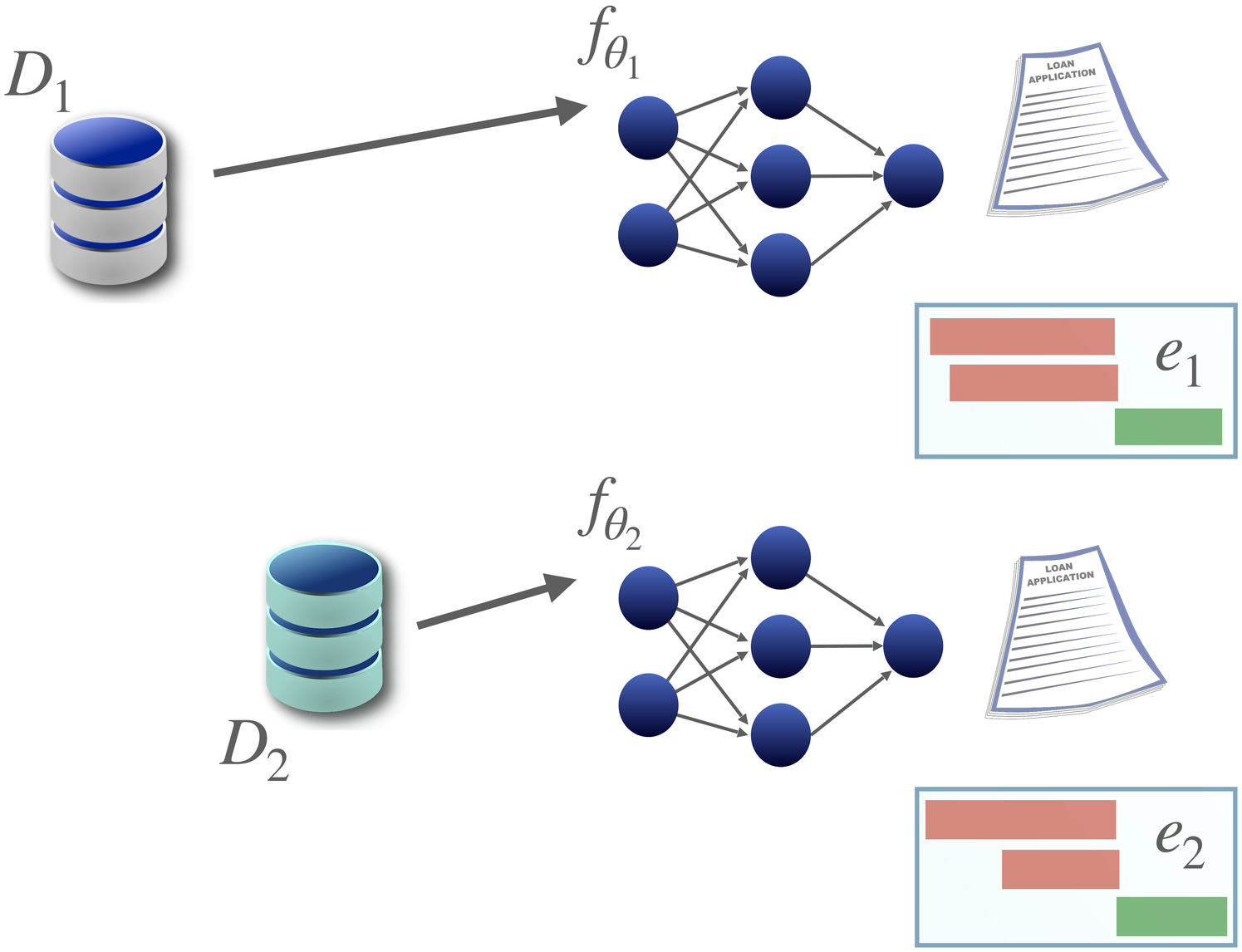
Explanation technique

Dataset	Explanation technique	Top-3 SA	Top-3 CDC	Top-3 SSA
WHO	Saliency	0.63±0.01	0.83±0.03	0.18±0.03
	SmoothGrad	0.94±0.00	0.94±0.01	0.91±0.00
	LIME	0.69±0.09	0.35±0.24	0.27±0.20
	K.SHAP	0.58±0.10	0.60±0.39	0.13±0.08

Explanation choice matters!

(SmoothGrad outperforms other techniques)

Explanation shift is affected by…



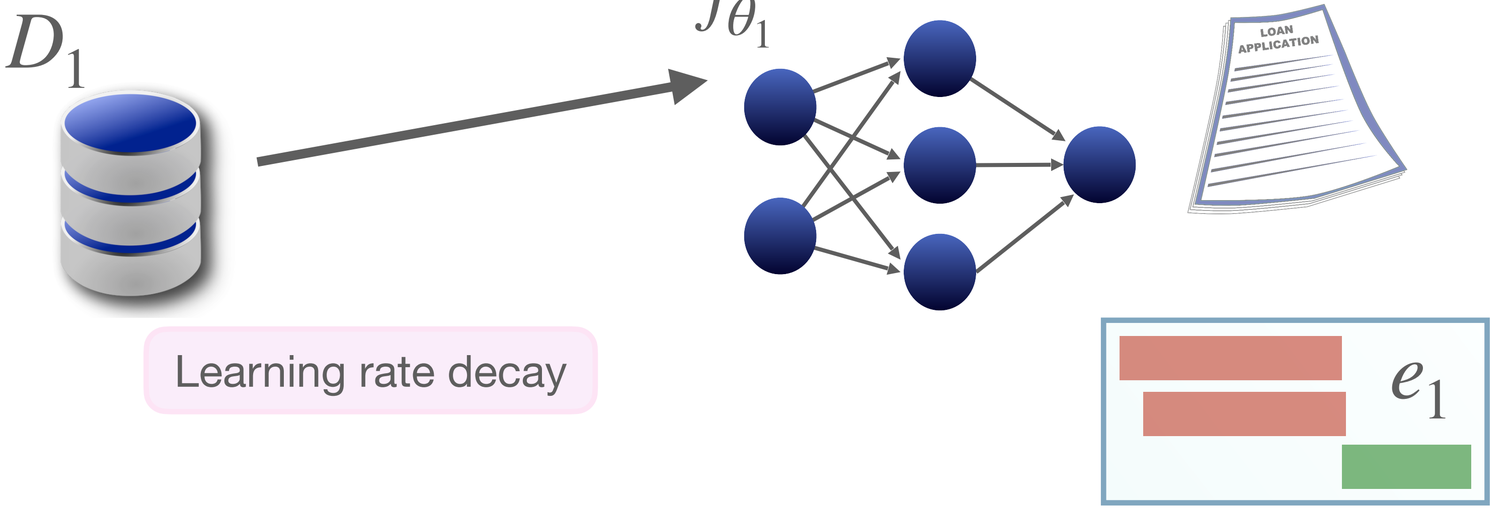
1. Dataset shift size

2. Weight decay parameter

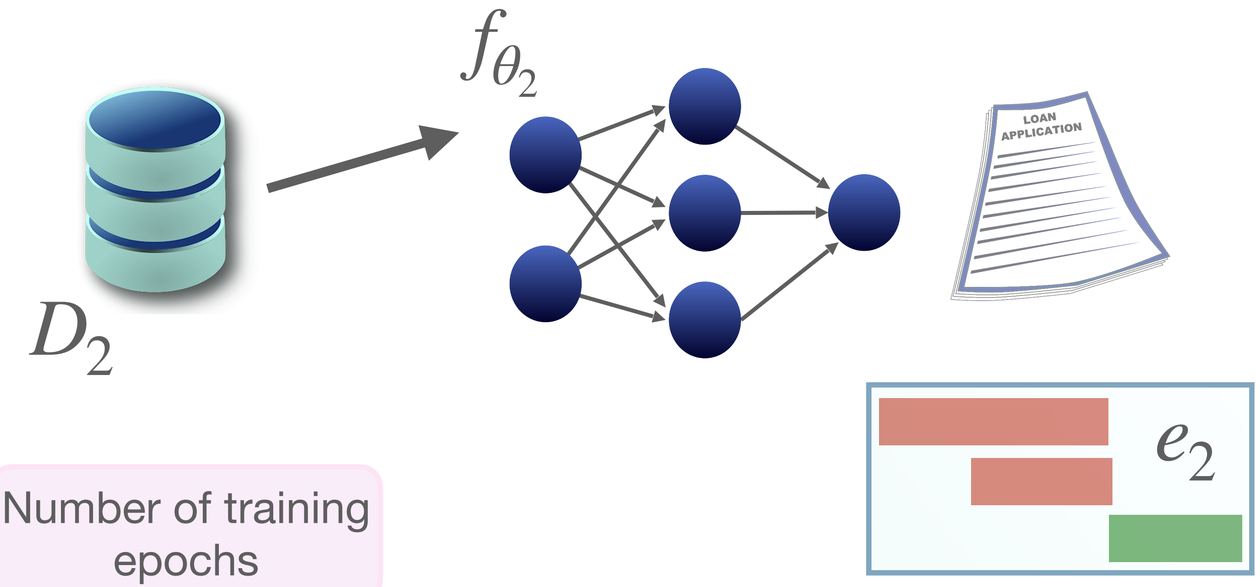
3. Smoothness of the activation function

Explanation shift is affected by…

Learning rate



Learning rate decay



Number of training epochs

1. Dataset shift size

2. Weight decay parameter

3. Smoothness of the activation function

Batch size

Training setup

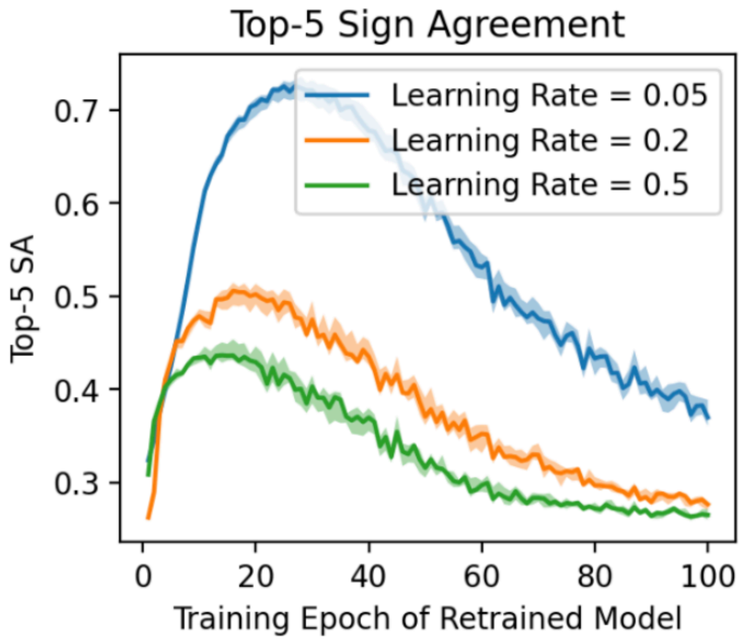
Complex explanations

Dataset shift

Temporal

Interventions

Other training hyperparameters



Training setup

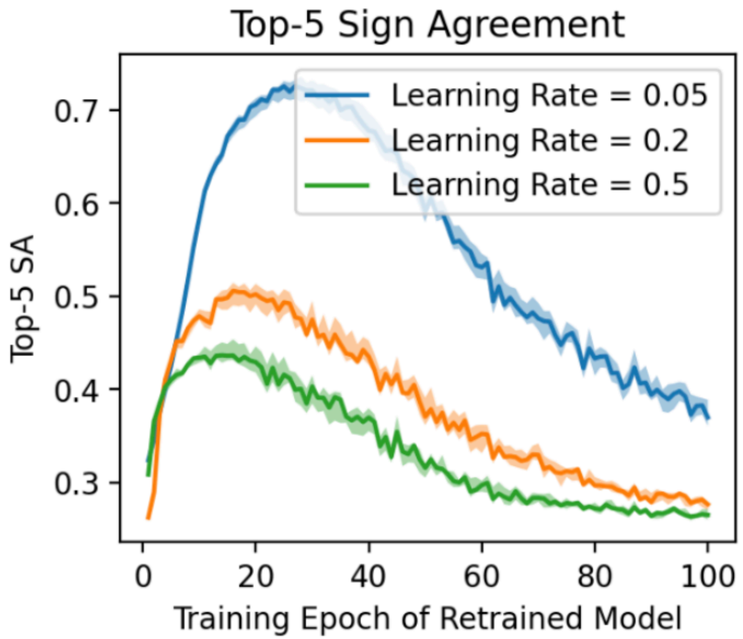
Complex explanations

Dataset shift

Temporal

Interventions

Other training hyperparameters



Peaks occur at different epochs

Training setup

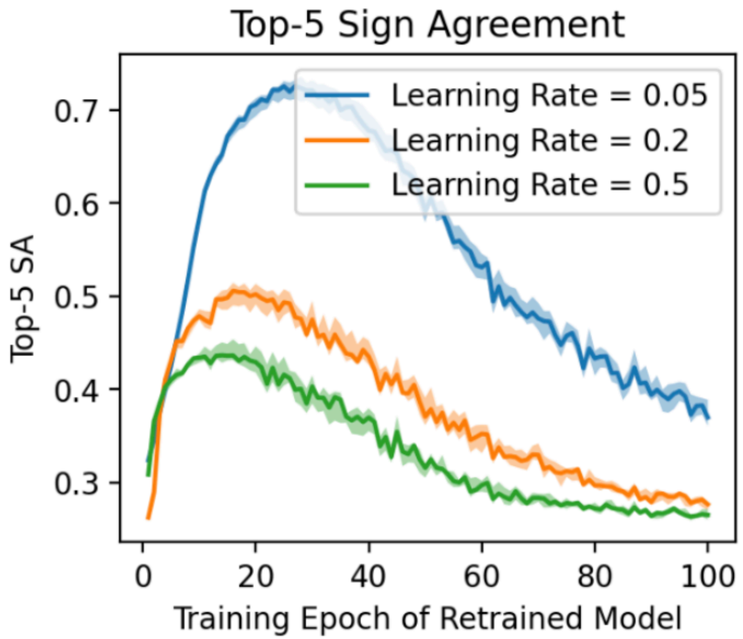
Complex explanations

Dataset shift

Temporal

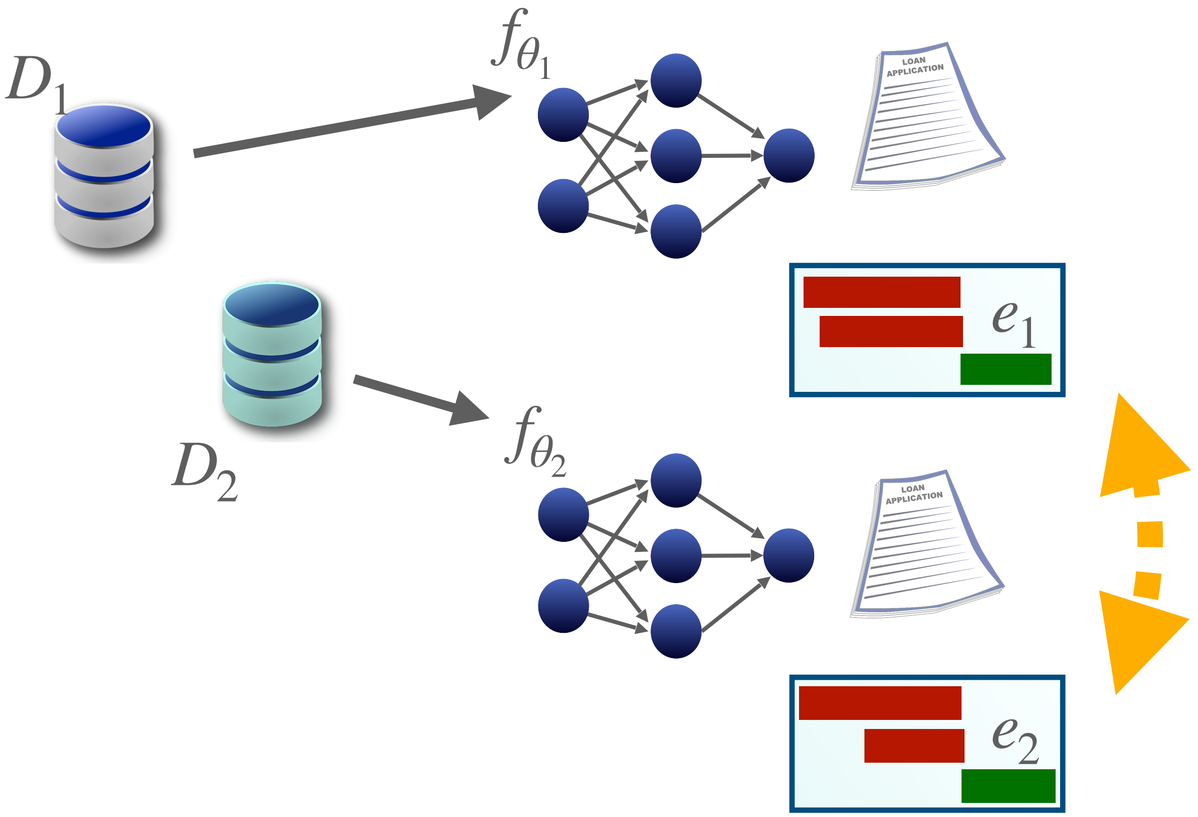
Interventions

Other training hyperparameters



Peaks occur at different epochs

Choice of hyperparameters is complex

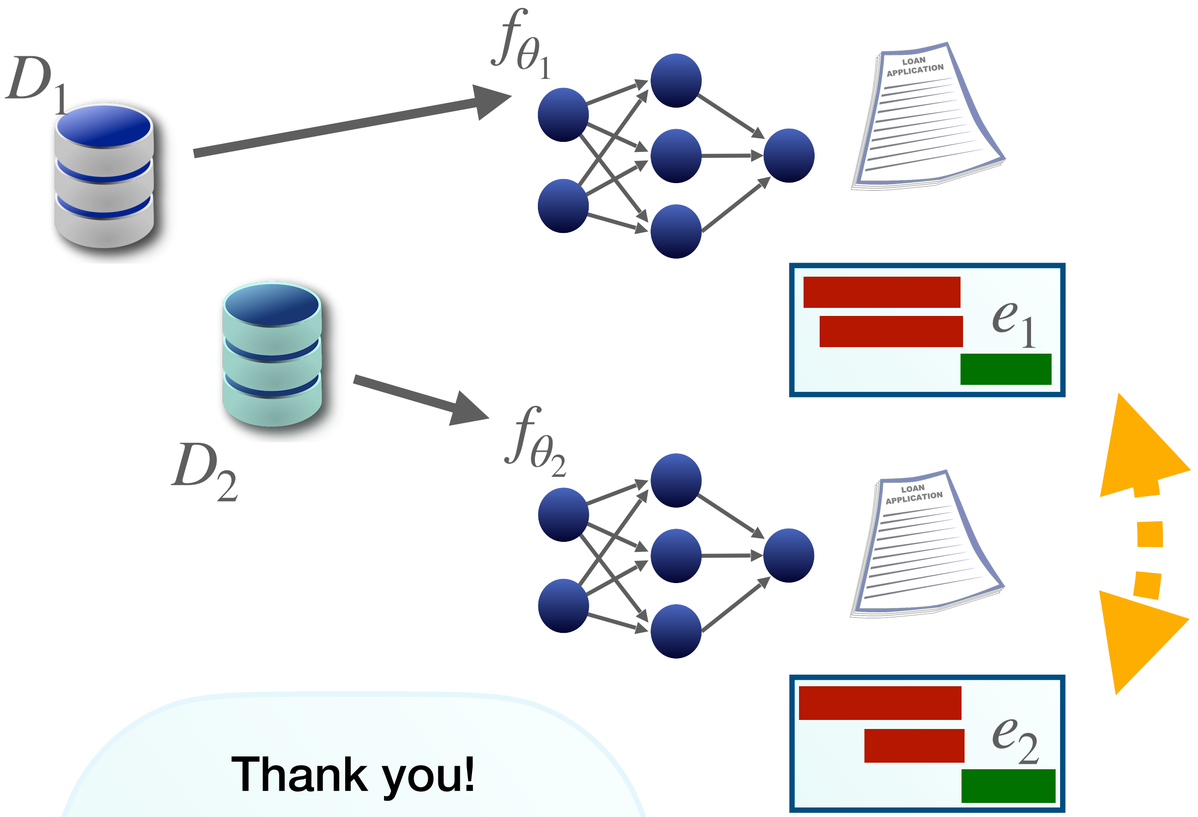


1. Theoretically proved what factors impact explanation stability

2. Validated that the theoretical results hold in practice

3. Extended the theoretical claims to real-world explanations

4. Empirically shown how other training hyperparameters impact explanation stability



1. Theoretically proved what factors impact explanation stability
2. Validated that the theoretical results hold in practice
3. Extended the theoretical claims to real-world explanations
4. Empirically shown how other training hyperparameters impact explanation stability

Thank you!

You can reach us at:
apmeyer4@wisc.edu
dley@g.harvard.edu

Our code is available at https://github.com/AI4LIFE-GROUP/robust-grads