NATIONAL RESEARCH UNIVERSITY

MLDS, 2022

UNSUPERVISED DOMAIN ADAPTATION BY BACKPROPAGATION

Davydkin Nikita

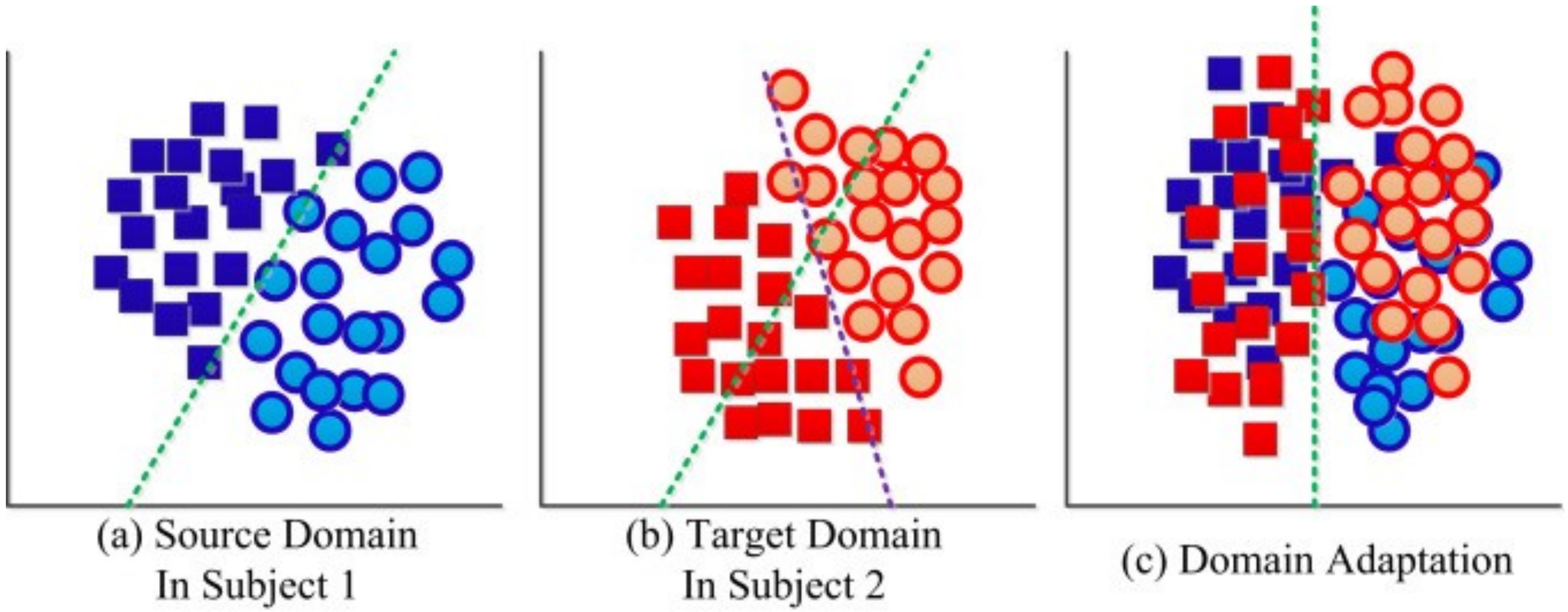
Moscow, 2022

Plan

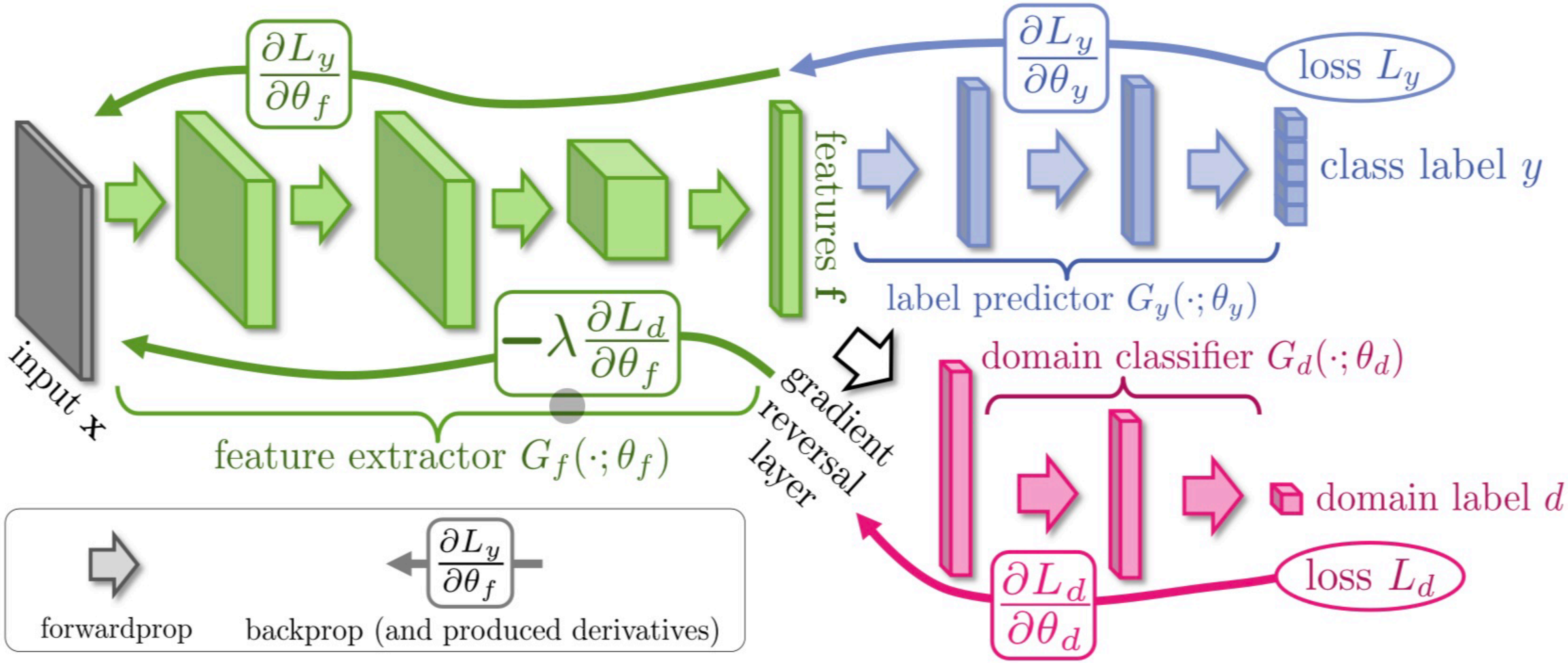
- Domain adaptation
- Model architecture
- Optimisation task
- Gradient reversal layer
- Data
- Experimental design
- Results
- Questions

Domain adaptation

Domain adaptation is the ability to apply an algorithm trained in one or more "source domains" to a different (but related) "target domain".



Model architecture



Goal: obtain **domain invariant features** by learning parameters of feature extractor to maximise domain classifier loss, while learning all parameters to make good label and domain classification.

Optimisation task

$$E(\theta_f, \theta_y, \theta_d) = \sum_{\substack{i=1..N \\ d_i=0}} L_y\left(G_y(G_f(\mathbf{x}_i; \theta_f); \theta_y), y_i\right) -$$

$$\lambda \sum_{i=1..N} L_d\left(G_d(G_f(\mathbf{x}_i; \theta_f); \theta_d), y_i\right) =$$

$$= \sum_{\substack{i=1..N \\ d_i=0}} L_y^i(\theta_f, \theta_y) - \lambda \sum_{i=1..N} L_d^i(\theta_f, \theta_d) \qquad (1)$$

$$(\hat{\theta}_f, \hat{\theta}_y) = \arg\min_{\theta_f, \theta_y} E(\theta_f, \theta_y, \hat{\theta}_d) \qquad (2)$$

$$\hat{\theta}_d = \arg\max_{\theta_d} E(\hat{\theta}_f, \hat{\theta}_y, \theta_d)\,. \qquad (3)$$

Optimisation task

$$\theta_f \quad \longleftarrow \quad \theta_f \; - \; \mu \left(\frac{\partial L_y^i}{\partial \theta_f} - \lambda \frac{\partial L_d^i}{\partial \theta_f} \right) \qquad (4)$$

$$\theta_y \quad \longleftarrow \quad \theta_y \; - \; \mu \frac{\partial L_y^i}{\partial \theta_y} \qquad (5)$$

$$\theta_d \quad \longleftarrow \quad \theta_d \; - \; \mu \frac{\partial L_d^i}{\partial \theta_d} \qquad (6)$$

Standard backpropagation can not be done without any modifications for this optimisation task. However, problem is solved by implementing the gradient reversal layer.

Gradient reversal layer

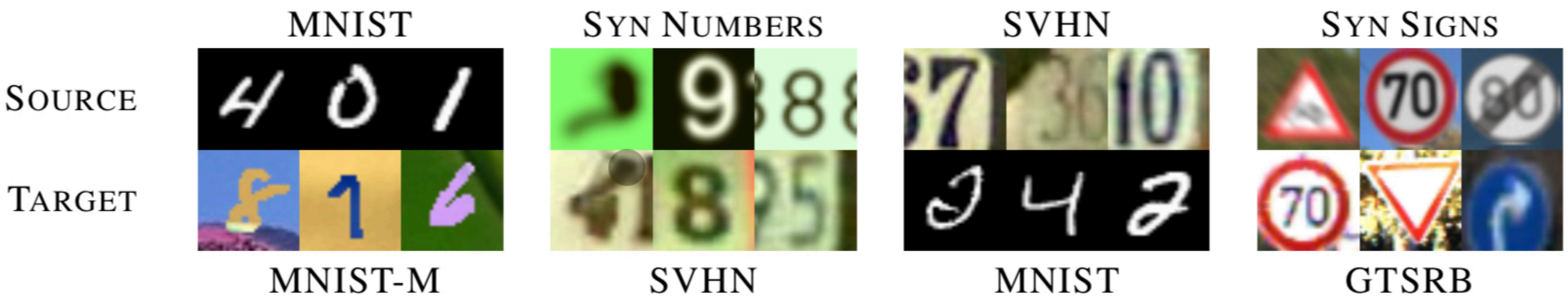
$$R_\lambda(\mathbf{x}) = \mathbf{x} \tag{7}$$

$$\frac{dR_\lambda}{d\mathbf{x}} = -\lambda \mathbf{I} \tag{8}$$

$$\tilde{E}(\theta_f, \theta_y, \theta_d) = \sum_{\substack{i=1..N\\ d_i=0}} L_y\left(G_y(G_f(\mathbf{x}_i;\theta_f);\theta_y), y_i\right) +$$

$$\sum_{i=1..N} L_d\left(G_d(R_\lambda(G_f(\mathbf{x}_i;\theta_f));\theta_d), y_i\right) \tag{9}$$

Data


MNIST
SYN NUMBERS
SVHN
SYN SIGNS
SOURCE
TARGET
MNIST-M
SVHN
MNIST
GTSRB

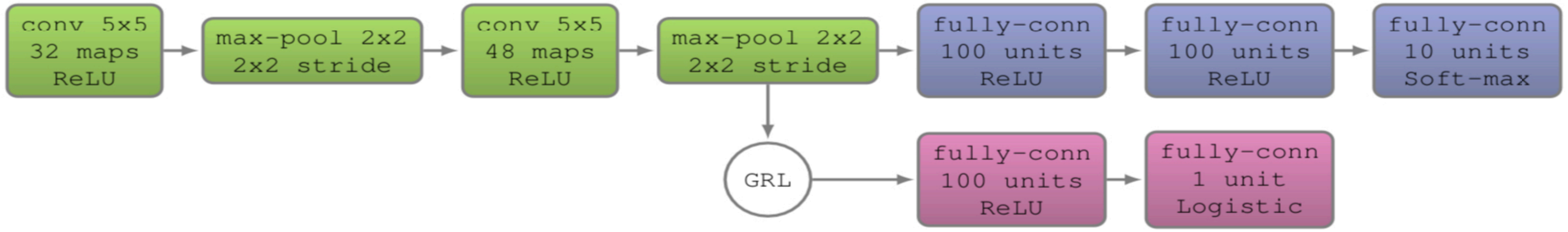

Amazon
Webcam
DSLR

Experimental design

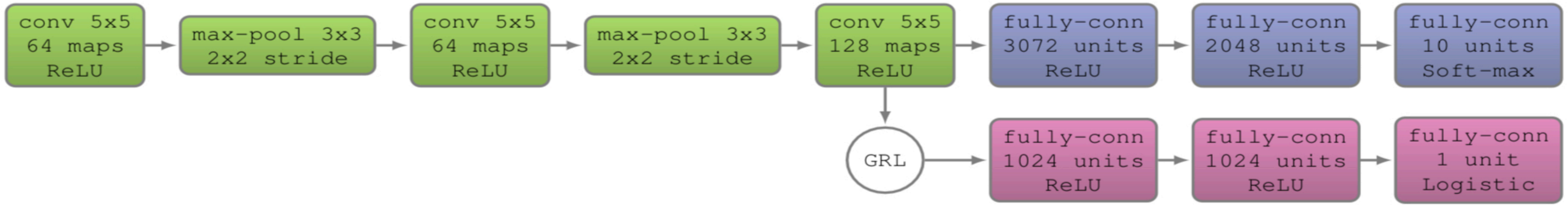
Baseline: source-only model trained without consideration for target-domain data

Upper bound: train-on-target model is trained on the target domain with class labels revealed

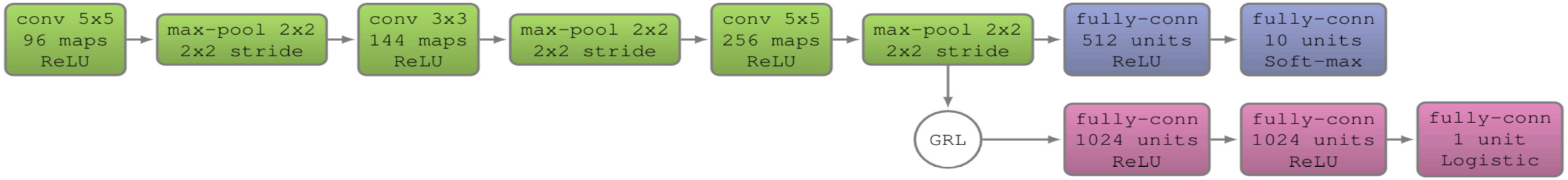
In addition: the approach is compared against the recently proposed unsupervised DA method based on subspace alignment (SA)



(a) MNIST architecture



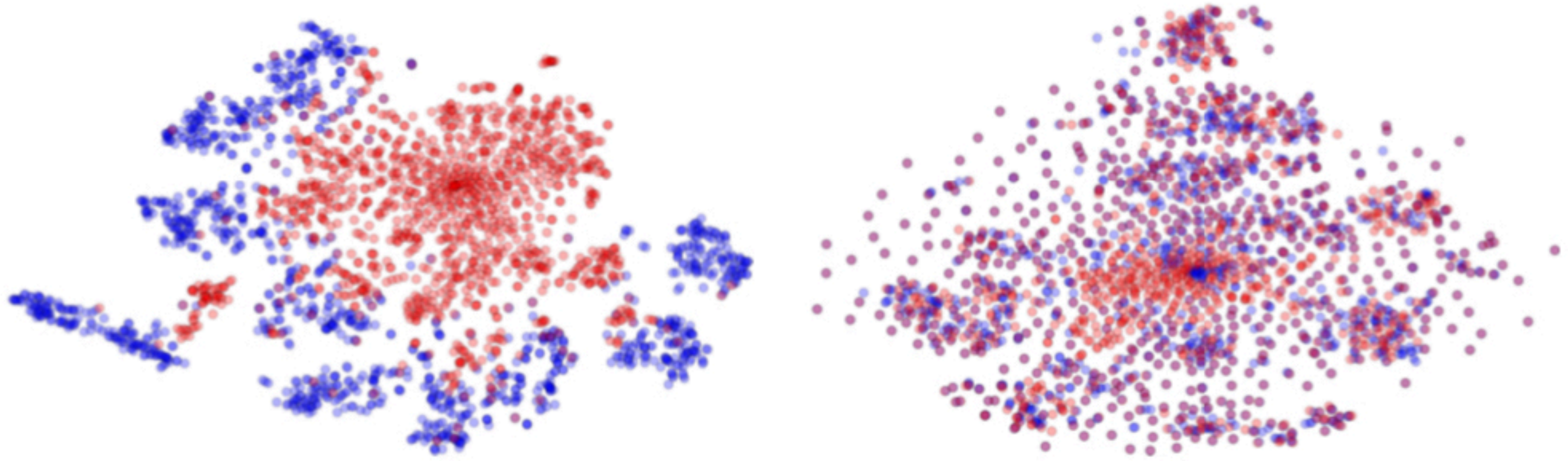
(b) SVHN architecture



(c) GTSRB architecture

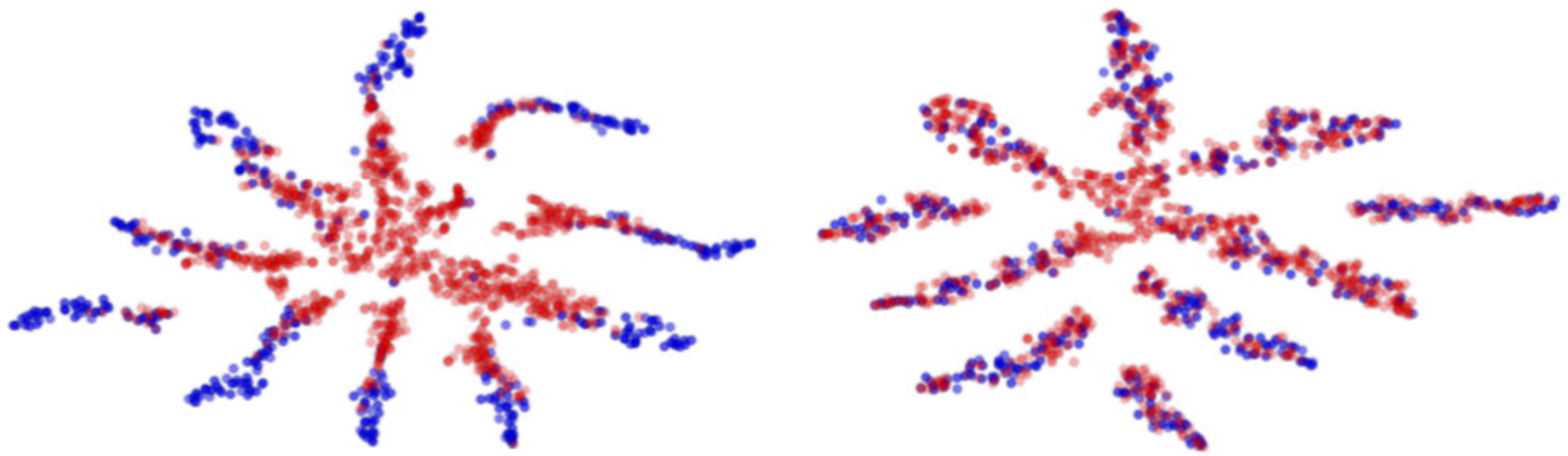
Results

MNIST → MNIST-M: top feature extractor layer



(a) Non-adapted (b) Adapted

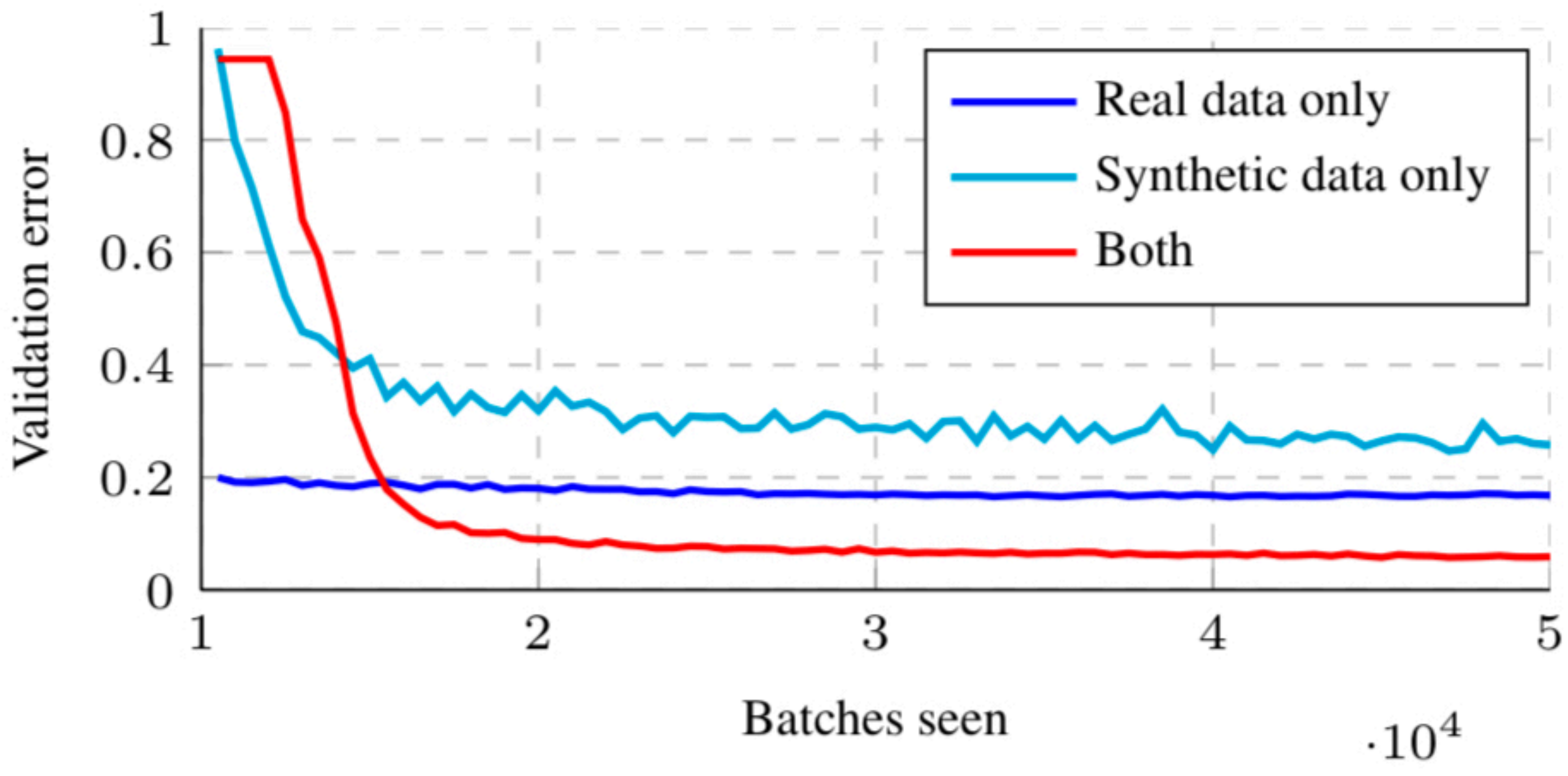
SYN NUMBERS → SVHN: last hidden layer of the label predictor



(a) Non-adapted (b) Adapted

METHOD (SOURCE / TARGET)	MNIST / MNIST-M	SYN NUMBERS / SVHN	SVHN / MNIST	SYN SIGNS / GTSRB
SOURCE ONLY	.5749	.8665	.5919	.7400
SA (FERNANDO ET AL., 2013)	.6078 (7.9%)	.8672 (1.3%)	.6157 (5.9%)	.7635 (9.1%)
PROPOSED APPROACH	**.8149** (57.9%)	**.9048** (66.1%)	**.7107** (29.3%)	**.8866** (56.7%)
TRAIN ON TARGET	.9891	.9244	.9951	.9987

Results



Real data only
Synthetic data only
Both
Validation error
Batches seen
$\cdot 10^4$
1
0.8
0.6
0.4
0.2
0
1
2
3
4
5

Results

Method (Source / Target)	Amazon → Webcam	DSLR → Webcam	Webcam → DSLR
GFK(PLS, PCA) (Gong et al., 2012)	$.464 \pm .005$	$.613 \pm .004$	$.663 \pm .004$
SA (Fernando et al., 2013)	$.450$	$.648$	$.699$
DA-NBNN (Tommasi & Caputo, 2013)	$.528 \pm .037$	$.766 \pm .017$	$.762 \pm .025$
DLID (S. Chopra & Gopalan, 2013)	$.519$	$.782$	$.899$
DeCAF$_6$ Source Only (Donahue et al., 2014)	$.522 \pm .017$	$.915 \pm .015$	–
DaNN (Ghifary et al., 2014)	$.536 \pm .002$	$.712 \pm .000$	$.835 \pm .000$
DDC (Tzeng et al., 2014)	$.594 \pm .008$	$.925 \pm .003$	$.917 \pm .008$
Proposed Approach	$\mathbf{.673 \pm .017}$	$\mathbf{.940 \pm .008}$	$\mathbf{.937 \pm .010}$

Questions

- What is domain adaptation?
- Describe optimisation goals for each bundle of parameters (feature extractor, label predictor, domain classifier) with respect to 2 losses (classification loss, domain loss).
- How does the gradient reversal layer work?

NATIONAL RESEARCH UNIVERSITY

Thank you for your attention !