# Parameter Regularization Schemes for AutoDL Transfer Learning

Li Xuhong

Big Data Lab, Baidu Research

# Transfer Learning

## Biological Motivation

Transfer the knowledge from one or more **source** tasks to one or more **target** tasks.

## Definitions

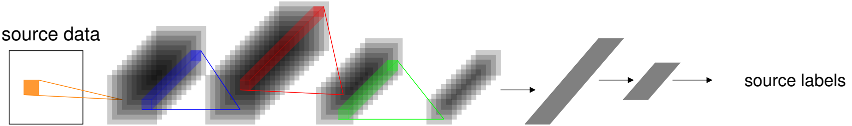
- Domain : data space ($\boldsymbol{x}$) and the corresponding distribution
- Task : label space ($\boldsymbol{y}$) and the parametric function to optimize for predicting the labels $\hat{\boldsymbol{y}} = f(\boldsymbol{x}; \boldsymbol{w})$

## Different Settings

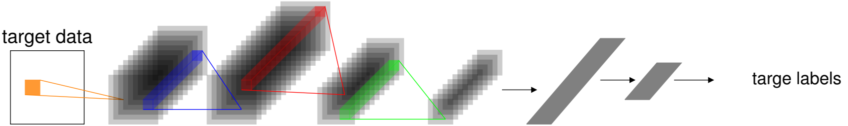
- Multi-Task : learning several tasks simultaneously
- Lifelong Learning : learning several tasks successively
- Domain Adaptation : different domains but the same task
- **Inductive Transfer** : the same domain but different tasks

# Transfer Learning and Fine-Tuning

## Fine-tuning : A practical method for (inductive) transfer learning




**Train** the model from scratch for solving the source task ;
**Adjust** the *pretrained* parameters for solving the target task.




**Catastrophic Forgetting**
=> not the objective of transfer learning

# Proposed Solution through Parameter Regularization Approaches

## Regularization

- Supplementary information added to the task

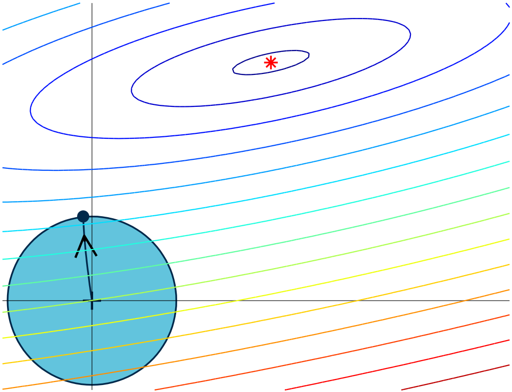
During the transfer, we apply the proposed regularization approaches to preserve the learned source knowledge.

## Regularized Loss Function (Supervised Learning)

- $L(f(\boldsymbol{x};\boldsymbol{w}^{(t)}),\boldsymbol{y}) + \alpha\Omega(\boldsymbol{w}^{(t)})$

# Standard Parameter Regularization (example of $L^2$)

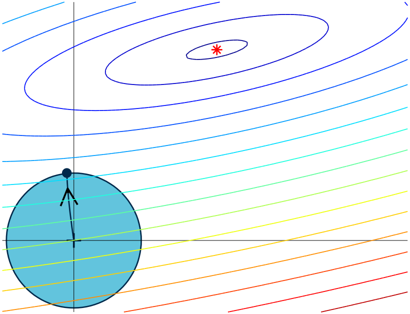
Weight Decay : $\Omega(\boldsymbol{w}) = \frac{\alpha}{2} \|\boldsymbol{w}\|_2^2$



+ Starting point $\boldsymbol{w}^0$
✳ Unregularized minimum
● Regularized minimum

# Weight Decay in Transfer Learning

## No Transfer

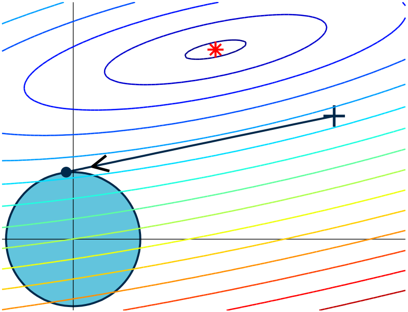


+ Starting point $\boldsymbol{w}^0$

✳ Unregularized minimum

● Regularized minimum

## With Transfer Learning

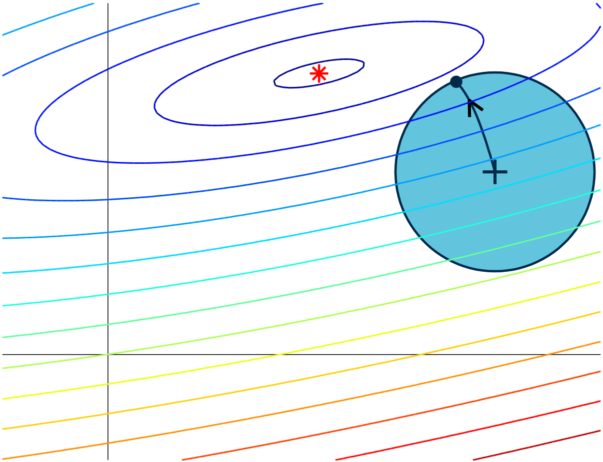


Staring point = pretrained parameters

$L^2$ effect => driving towards the origin

Not adapted to transfer learning

# Regularization with the Pretrained Model as Reference



+ Starting point $\boldsymbol{w}^0$
✳ Unregularized minimum
● Regularized minimum

The pretrained model is used
(1) as the starting point (-*SP*) for fine-tuning,
(2) also as the reference for the parameter regularization.

$L^2$-*SP* :
$$\Omega(\boldsymbol{w}) = \frac{\alpha}{2} \left\| \boldsymbol{w} - \boldsymbol{w}^0 \right\|_2^2$$

# -*SP* Regularization For Transfer Learning

## Standard Regularization $L^2$ (weight decay)

$$L^2 : \Omega(\boldsymbol{w}) = \frac{\alpha}{2} \|\boldsymbol{w}\|_2^2$$

## Regularization -*SP* (Starting Point)

- $L^2$-*SP* : $\Omega(\boldsymbol{w}) = \frac{\alpha}{2} \left\|\boldsymbol{w} - \boldsymbol{w}^0\right\|_2^2$
- $L^1$-*SP* : $\Omega(\boldsymbol{w}) = \alpha \left\|\boldsymbol{w} - \boldsymbol{w}^0\right\|_1$
- *GL-SP* : $\Omega(\boldsymbol{w}) = \alpha \sum_{g=1}^{G} s_g \left\|\boldsymbol{w}_{\mathcal{G}_g} - \boldsymbol{w}^0_{\mathcal{G}_g}\right\|_2$

$\boldsymbol{w}$ : parameter vector to learn ;
$\boldsymbol{w}^0$ : pretrained parameter vector ;
$\mathcal{G}_g$ : group of parameters ;
$s_g$ : predefined constant for balancing the groups.

# Fisher Information Metric

## Estimated Fisher Information Matrix

$$\hat{F}_{jj} = \frac{1}{m} \sum_{i=1}^{m} \sum_{c=1}^{C} f_c(\boldsymbol{x}_i; \boldsymbol{w}^0) \left( \frac{\partial}{\partial w_j} \log f_c(\boldsymbol{x}_i; \boldsymbol{w}^0) \right)^2$$

$i$ : example index
$c$ : class index
$j$ : parameter index

Fisher information mesures the model's sensibility on the source task w.r.t. the parameters.

## Regularization Approaches with Fisher information

- $L^2$-*SP* : $\Omega(\boldsymbol{w}) = \frac{\alpha}{2} \left\| \boldsymbol{w} - \boldsymbol{w}^0 \right\|_2^2$
- $L^2$-*SP-Fisher* : $\Omega(\boldsymbol{w}) = \frac{\alpha}{2} \sum_j \hat{F}_{jj} \left( w_j - w_j^0 \right)^2$
- *GL-SP* : $\Omega(\boldsymbol{w}) = \alpha \sum_{g=1}^{G} s_g \left\| \boldsymbol{w}_{\mathcal{G}_g} - \boldsymbol{w}^0_{\mathcal{G}_g} \right\|_2$
- *GL-SP-Fisher* : $\Omega(\boldsymbol{w}) = \alpha \sum_{g=1}^{G} s_g \left( \sum_{j \in \mathcal{G}_g} \hat{F}_{jj} \left( w_j - w_j^0 \right)^2 \right)^{1/2}$

## Application to Image Classification – Experimental Settings

**Source Datasets**

- ImageNet
- Places365

**Target Datasets**

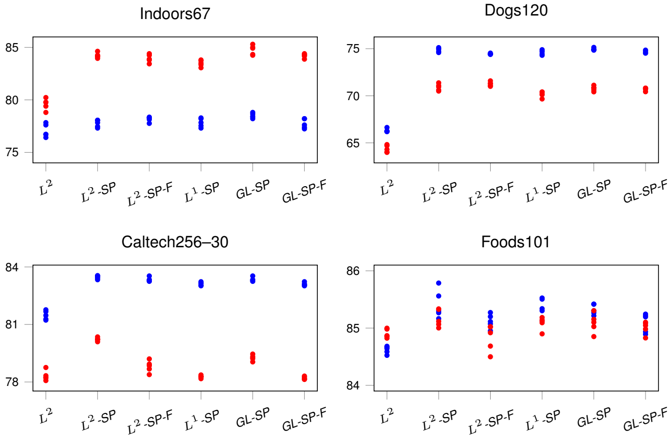
- Indoors67
- Dogs120
- Caltech256
- Foods101

**Network Structure**

- ResNet-101

# Application to Image Classification – Experimental Results

- source : ImageNet
- source : Places365




## Remarks

- *-SP* is always better than $L^2$.
- Similarity helps transfer.
- Performances of $L^1$ et *GL* are not better than $L^2$ (due to the discontinuity of gradients ?).
- $L^2$-*SP* is efficient.

# $L^2$-*SP* Application to Image Semantic Segmentation

**Cityscapes**

| Approach | $L^2$ | $L^2$-*SP* |
|---|---|---|
| FCN | 66.9 | **67.9** |
| ResNet-101 | 68.1 | **68.7** |
| DeepLab | 68.6 | **70.4** |
| DeepLab-COCO | 72.0 | **73.2** |
| PSPNet | 78.2 | **79.4** |
| PSPNet-extra | 80.9 | **81.2** |

**Pascal VOC**

78.3 vs **79.9** on the validation set

image RGB


segmentation PSPNet-extra + $L^2$-*SP*

# More Experiments with $L^2$-SP

## Cooperation with three other research teams (Tsinghua, Amazon et Cornell)

The performances are systematically improved by $L^2$-SP.

| **Approach** | **Target Data** | **Task** | **Metric** | $L^2$ | $L^2$-SP |
|---|---|---|---|---|---|
| EncNet-50 [1] | PASCAL Context | image segmentation | mIoU | 50.84 | **51.17** |
| EncNet-101 | PASCAL Context | image segmentation | mIoU | 54.10 | **54.12** |
| SegFlow* [2] | DAVIS | video segmentation | IoU | 65.5 | **66.2** |
| SegFlow | DAVIS | video segmentation | IoU | 67.4 | **68.0** |
| SegFlow | Monkaa *Final* | optical flow | EPE | 7.90 | **7.17** |
| SegFlow | Driving *Final* | optical flow | EPE | 37.93 | **30.31** |
| DSTL [3] | Birds200 | image classification | accuracy | 88.47 | **89.19** |
| DSTL | Flowers102 | image classification | accuracy | 97.21 | **97.68** |
| DSTL | Cars196 | image classification | accuracy | 90.19 | **90.67** |
| DSTL | Aircraft100 | image classification | accuracy | 85.89 | **86.83** |
| DSTL | Food101 | image classification | accuracy | 88.16 | **88.75** |
| DSTL | NABirds | image classification | accuracy | 87.64 | **88.32** |

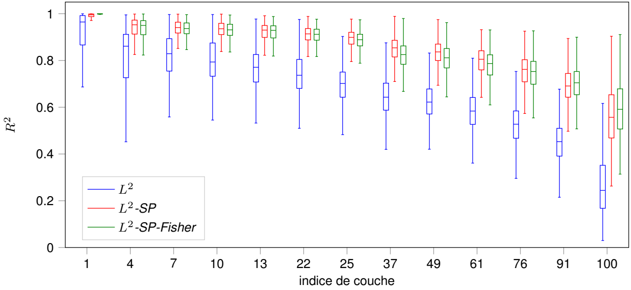
1. Hang Zhang et al. “Context encoding for semantic segmentation”. In : **CVPR**. 2018.
2. Jingchun Cheng et al. “SegFlow : Joint learning for video object segmentation and optical flow”. In : **ICCV**. 2017.
3. Yin Cui et al. “Large scale fine-grained categorization and domain-specific transfer learning”. In : **CVPR**. 2018.

# Analyses of the $L^2$-*SP* Regularization

- Performance drops in (%) on the source task after *fine-tuning*

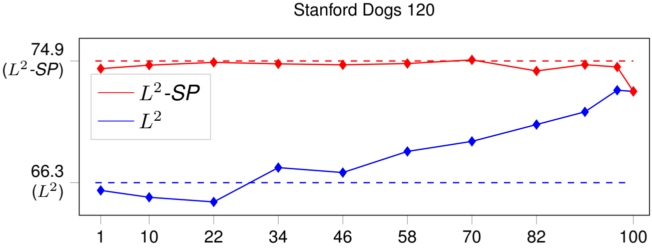
| | $L^2$ | $L^2$-*SP* | $L^2$-*SP-Fisher* |
|---|---|---|---|
| MIT Indoors 67 | 24.1 | 5.3 | **4.9** |
| Caltech 256–30 | 15.4 | 4.2 | **3.6** |
| Caltech 256–60 | 16.9 | 3.6 | **3.2** |
| Stanford Dogs 120 | 14.1 | 4.7 | **4.2** |
| Foods 101 | 68.6 | 64.5 | **53.2** |

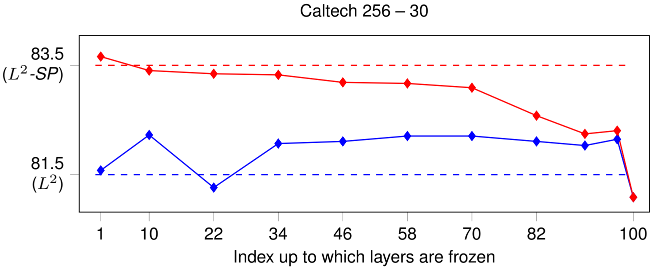
- Effect of $L^2$-*SP* on the activations



# Analyses of the $L^2$-*SP* Regularization

- Freezing first layers when fine-tuning - ResNet-101



# Conclusion

**Conclusion**

- weight decay in transfer learning
- use starting point (-*SP*) as the reference
- experiments on different tasks, with different networks
- analyses