Domain Adaptation and Multi-view Learning: using subspace alignment and landmark projections

Rémi Emonet

Summer School on Transfer Learning

2018-06-06


Team Data Intelligence @ LabHC
(at some point in the past, not exaustive)

Disclaimer

In a nutshell

- Transfer learning has multiple facets
  - multi-task
  - multi-view
  - multi-domain
- Domain adaptation by bringing distributions together
  - by aligning subspaces obtained from PCA
  - non-linearly by using projection on landmarks
- Landmarks can also be used for multiview-learning
  - random landmark selection
  - non-linear projection on the landmarks
  - fast linear model

LABORATOIRE HUBERT CURIEN
UMR • CNRS • 5516 • SAINT-ETIENNE

Transfer Learning:
Multi-* Learning

Multi-Task Learning

- Covered a lot in this summer school
- (At least), different output for each task, e.g.,
  - different classification task: dog-vs-cat and domestic-vs-wild
  - different output kind: image segmentation and image classification
  - ...

Multi-View Learning

- Input have multiple views, e.g.
  - different viewpoints of an object
  - multi-modal perception
  - different medical tests on a patient
  - different sets of features extracted from images
  - ...
- There could be missing views for some input data

(we'll come back to this)

Multi-domain Learning?

Domain Adaptation: What and Why?

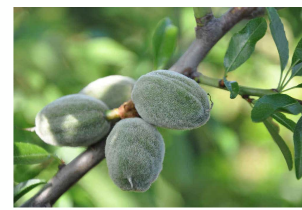
When do we need Domain Adaptation (DA)?

- The training distribution is different from the testing distribution

Example Domain Adaptation task?

- Given: labeled images (e.g., fruits images)
- Task: what fruit appears on this unlabeled images of trees

⇒



Blueberry Almond Blueberry Almond

- How can we learn, from one distribution,
  a low-error classifier on another distribution?

Overview

- The Multiple Facets of Transfer Learning
- Domain Adaptation by Subspace Alignment

  Landmark-based Kernelized Subspace Alignment
- Deep Multi-Domain Multi-Task Learning
- Random Landmark projection for Multi-View Learning

Overview

- The Multiple Facets of Transfer Learning
- Domain Adaptation by Subspace Alignment

  Landmark-based Kernelized Subspace Alignment
- Deep Multi-Domain Multi-Task Learning
- Random Landmark projection for Multi-View Learning

LABORATOIRE HUBERT CURIEN
UMR • CNRS • 5516 • SAINT-ETIENNE

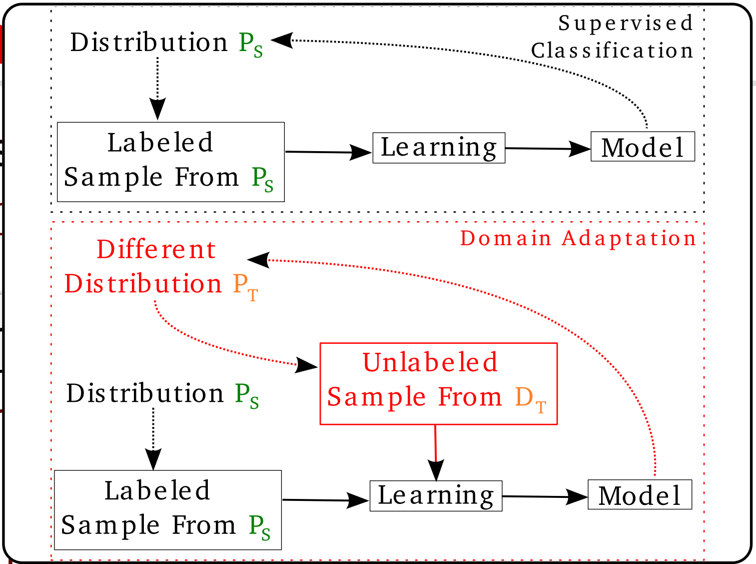
Unupervised Domain Adaptation

by Subspace Alignment: B. Fernando

and Landmark Selection: R. Aljundi

Domain Adaptation: ta

- Typical binary classification tas
  - X : input space, $Y = \{-1, +1$



Typical supervised classification

- P_S source domain: distribution over
- $S = \{(x_i^s, y_i^s)\}_{i=1}^{m_s} \sim (P_S)^{m_s}$: a sa
- Goal: Find a classifier $h \in \mathcal{H}$ with a low source error

$$R_{P_S}(h) = \mathbf{E}_{(x^s, y^s) \sim P_S} \; \mathbf{I}\left[h(x^s) \neq y^s\right]$$

Domain Adaptation

- P_T target domain: distribution over $X \times Y$, (D_T: marginal over X)
- $T = \{(x_i^t)\}_{j=1}^{m_t} \sim (D_T)^{m_t}$: a sample of unlabeled target points
- Goal: Find a classifier $h \in \mathcal{H}$ with a low target error

$$R_{P_T}(h) = \mathbf{E}_{(x^t, y^t) \sim P_T} \; \mathbf{I}\left[h(x^t) \neq y^t\right]$$

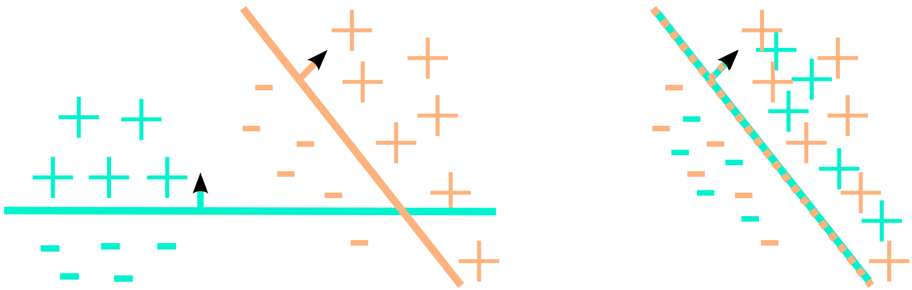
Domain Adaptation – Domain Divergence

Labeled source samples S
drawn *i.i.d.* from P_S

Unlabeled target samples T
drawn *i.i.d.* from P_T

- h is learned on the source, how does it perform on the target?

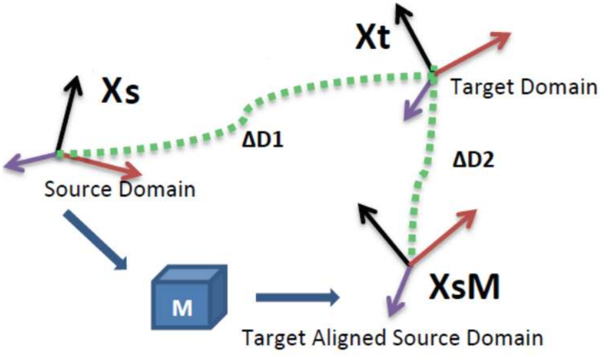
  ⇒ it depends on the closeness of the domains



Adaptation Bound [Ben-David et al., MLJ'10, NIPS'06]

- $\forall h \in \mathcal{H}, \quad R_{P_T}(h) \;\leq\; R_{P_S}(h) \;+\; \frac{1}{2} d_{\mathcal{H}\,\Delta\,\mathcal{H}}(D_S, D_T) \;+\; \nu$
- Domain divergence: $d_{\mathcal{H}\,\Delta\,\mathcal{H}}(D_S, D_T) \;=\; 2 \sup_{(h,h')\in\mathcal{H}^2} \left| R_{D_T}(h,h') - R_{D_S}(h,h') \right|$
- Error of the joint optimal classifier: $\nu = \inf_{h'\in\mathcal{H}} \left(R_{P_S}(h') + R_{P_T}(h') \right)$

- Intuition for unsupervised domain adaptation
  - principal components of the domains may be shared
  - principal components should be re-aligned
- Principle
  - extract a source subspace (d largest eigen vectors)
  - extract a target subspace (d largest eigen vectors)
  - learn a linear mapping function
    that aligns the source subspace with the target one



Subspace Alignment – Algorithm

Algorithm

- **Input:** Source data S, Target data T, Source labels L_S

  Input: Subspace dimension d

  Output: Predicted target labels L_T
- $X_S \leftarrow PCA(S, d)$ *(source subspace defined by the first d eigenvectors)*
- $X_T \leftarrow PCA(T, d)$ *(target subspace defined by the first d eigenvectors)*
- $M \leftarrow {X_S}' X_T$ *(closed form alignment)*
- $X_a \leftarrow X_S M$ *(operator for aligning the source subspace to the target one)*
- $S_a = S X_a$ *(new source data in the aligned space)*
- $T_T = T X_T$ *(new target data in the aligned space)*
- $L_T \leftarrow Classifier(S_a, L_S, T_T)$

- A natural similarity: $Sim(\mathbf{x}_s, \mathbf{x}_t) = \mathbf{x}_s X_S M X_T' \mathbf{x}_t' = \mathbf{x}_s A \mathbf{x}_t'$

Subspace Alignment – Recap.

- Good
  - Very simple and intuitive method
  - Totally unsupervised
  - Theoretical results for dimensionality detection
  - Good results on computer vision datasets
  - Can be combined with supervised information
- Bad
  - Cannot be directly kernelized to deal with non linearity
  - Actually assumes that spaces are relatively close
- Ugly
  - Assumes that all the source and target examples are relevant

Overview

- The Multiple Facets of Transfer Learning
- Domain Adaptation by Subspace Alignment

  Landmark-based Kernelized Subspace Alignment
- Deep Multi-Domain Multi-Task Learning
- Random Landmark projection for Multi-View Learning

Subspace Alignment – Recap.

- Good
  - Very simple and intuitive method
  - Totally unsupervised
  - Theoretical results for dimensionality detection
  - Good results on computer vision datasets
  - Can be combined with supervised information
- Bad
  - Cannot be directly kernelized to deal with non linearity
  - Actually assumes that spaces are relatively close
- Ugly
  - Assumes that all the source and target examples are relevant
- **Idea:** *Select landmarks from both source and target domains to project the data in a common space using a kernel w.r.t those chosen landmarks. Then the subspace alignment is performed.*

Principle of Landmarks

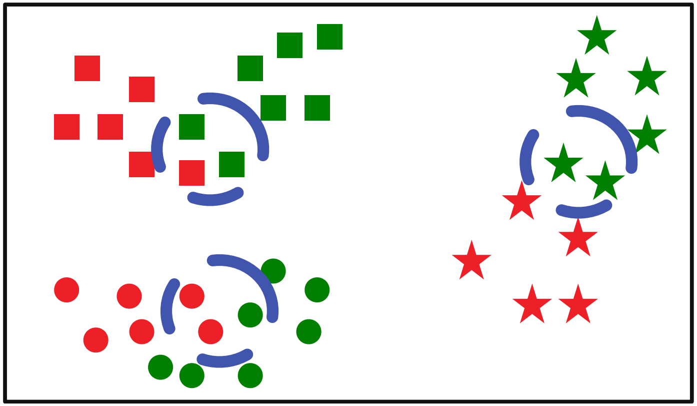
JMLR 2013 – *Connecting the Dots with Landmarks: Discriminatively Learning Domain-Invariant Features for Unsupervised Domain Adaptation*

- Boqing Gong, Kristen Grauman, Fei Sha

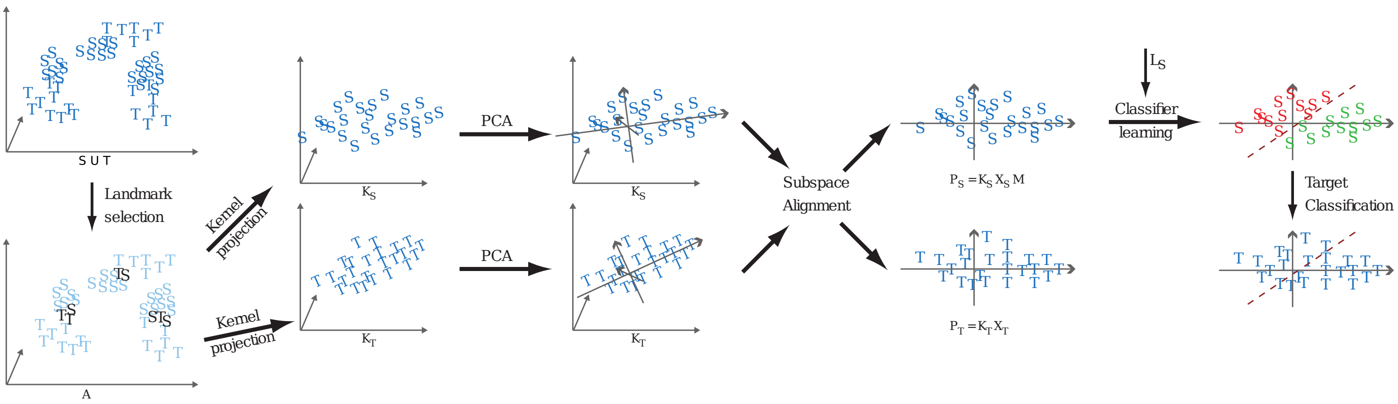
- Principle: find source points (the landmarks) such that the domains are similarly distributed “around”

- Optimization problem: $$\min_{\alpha} \left\| \frac{1}{\sum_m \alpha_m} \sum_m \alpha_m \phi(x_m) - \frac{1}{N} \sum_n \phi(x_n) \right\|^2$$
  - α: binary landmark indicator variables
  - $\phi(.)$: nonlinear mapping, maps every x to a RKHS
  - minimize the difference in sample-means
  - a constraint: *labels should be balanced among the landmarks*

- Intuition for landmarks-based alignment
  - subspace alignment does not handle non-linearity
  - subspace alignment cannot “ignore” points
  - landmarks can be a useful to handle locality and non-linearity
- Challenges
  - selecting landmarks in a unsupervised way
  - choosing the proper Gaussian-kernel scale



Proposed Approach – Workflow



- Overall approach
  - 2 new steps: *landmark selection*, *projection* on landmarks
  - subspace alignment

Multiscale Landmark Selection

- Select landmarks among all points, $S \cup T$
- Greedy selection
  - consider each candidate point c and a set of possible scales s
  - criteria to promote the candidate
    - after projection on the candidate
    - the overlap between source and target distributions is above a threshold
- Projection: a point is projected with $K(c, p) = \exp\left(\frac{-\|c-p\|^2}{2s^2}\right)$
- Overlap
  - project source and target points
  - fit two Gaussians (one for each)
  - $overlap(\mu_S, \sigma_S; \mu_T, \sigma_T) = \frac{\mathcal{N}(\mu_S - \mu_T \mid 0, \sigma^2_{sum})}{\mathcal{N}(0 \mid 0, \sigma^2_{sum})}$
    - normalized integral of product
    - with $\sigma^2_{sum} = {\sigma_S}^2 + {\sigma_T}^2$, and $\mathcal{N}(. \mid 0, \sigma^2_{sum})$ centered 1d-Gaussian

LABORATOIRE HUBERT CURIEN
UMR • CNRS • 5516 • SAINT-ETIENNE

Landmark-Based Alignment – Overall

- Select landmarks among all points, $S \cup T$
  - greedy selection
  - multi-scale selection
  - maximize domain overlap
- Project all points on the landmarks
  - use a Gaussian kernel
  - $\sigma \leftarrow median\_distance(S \cup T)$
- Subspace-align the projected points
  - PCA on source domain
  - PCA on target domain
  - compute the alignment M

Landmark-Based Alignment – Results

- Is landmark-based kernelization useful?

Comparison (in terms of accuracy) of unsupervised DA methods. C: Caltech, A: Amazon, W: Webcam, D: Dslr. NA: No Adaptation; KPCA+SA: two independent KPCA are performed on the source and target data, then a subspace alignment is applied; GFK: Geodesic Flow Kernel; SA: one step Subspace Alignment; TJM: Joint Matching Transfer; LSSA: our approach.

Method	A→W	A→D	A→C	C→D	C→W	C→A	W→D	W→A	W→C	D→W	D→C	D→A	Avg
NA	31.5	40.7	45.4	38.2	30.2	50.1	80.2	32.4	31.2	67.8	28.3	30.8	42.2
KPCA+SA	10.1	5.1	7.7	7.6	10.5	10.4	7.6	10.4	11.8	7.2	8.5	7.5	8,7
GFK	38.6	35.7	40.1	44.6	39.0	54.1	81.2	36.6	28.9	80.3	39.2	33.1	45.9
SA	40.7	46.4	41.6	49.0	42.7	52.7	78.9	39.4	34.7	83.4	44.8	38.0	49.3
TJM	42.0	45.8	45.7	49.0	48.8	58.6	83.4	40.8	34.8	82.0	39.6	35.1	50.5
LSSA	42.4	47.2	44.8	54.1	48.1	58.4	87.2	39.4	34.7	87.1	45.7	38.1	52.6

- Is our landmark-selection any good?

Table 1. Comparison (in terms of accuracy) of 5 landmark selection methods on 12 unsupervised DA subproblems. C: Caltech, A: Amazon, W: Webcam, D: Dslr. RD: Random Selection; All: all the source and target examples are used; σ-LS: our selection method with a f xed σ; CDL: Connecting Dots with Landmarks; MLS: our approach. In red, one reports the best method.

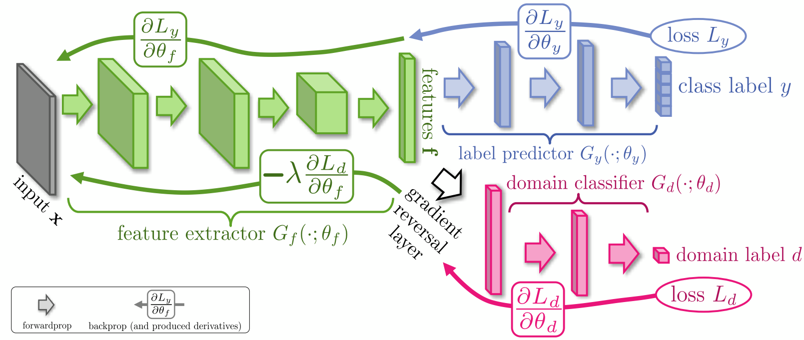
Method	A→W	A→D	A→C	C→D	C→W	C→A	W→D	W→A	W→C	D→W	D→C	D→A	Avg
RD	40.3	38.8	42.3	41.2	40.6	47.5	84.0	32.9	28.4	81.8	36.8	32.3	45.6
All	41.0	39.4	44.7	41.4	41.6	49.6	85.3	33.0	29.2	82.7	38.6	31.3	46.5
σ-LS	39.3	37.5	43.8	42.7	31.5	52.4	80.3	32.6	29.5	82.0	38.6	31.2	45.1
CDL	38.3	38.8	43.9	45.8	45.4	51.7	77.7	35.3	30.9	72.5	33.9	33.3	45.6
MLS	41.1	39.5	45.0	45.2	44.1	53.6	84.7	35.9	31.6	82.4	39.2	34.5	48.1

“Deep” Domain Adapation

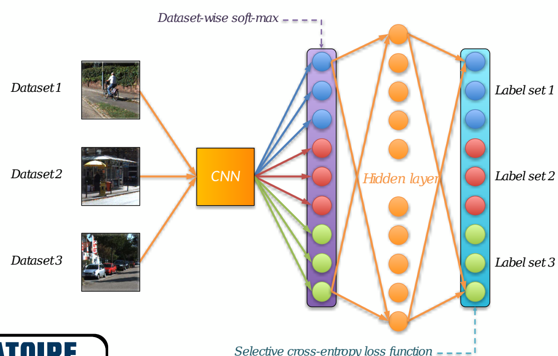
Domain Adaptation in Deep Neural Nets

Based on the same core principles: bring distributions together

- See Elisa Fromont's talk
  - Domain-Adversarial Training… Ganin et al. (JMLR 2016)
  - ADDA
  - chairlifts
  - avoiding negative transfer using domain distances
- Batch normalization and AdaBN
- AutoDIAL
- Multitask-multidomain semantic segmentation (Damien Fourure)



Unsupervised Domain Adaptation by Backpropagation



Overview

- The Multiple Facets of Transfer Learning
- Domain Adaptation by Subspace Alignment

  Landmark-based Kernelized Subspace Alignment
- Deep Multi-Domain Multi-Task Learning
- Random Landmark projection for Multi-View Learning

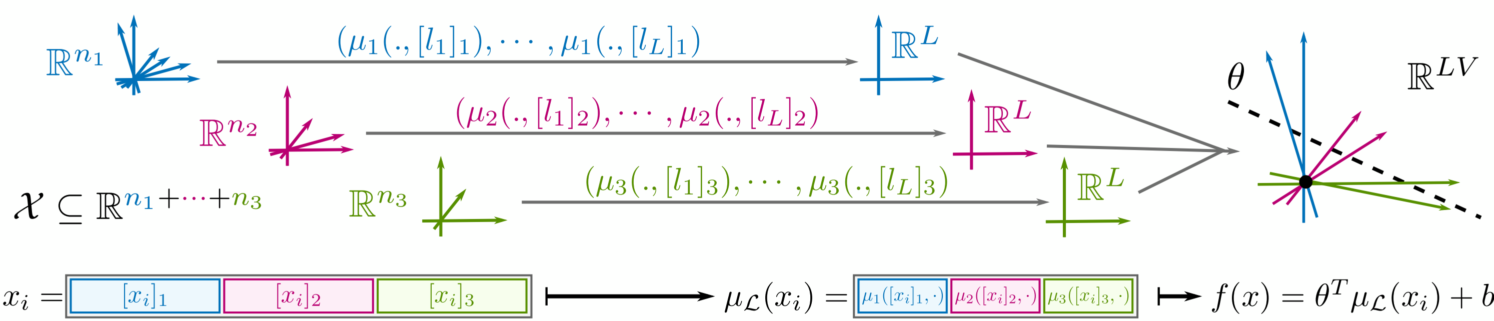
Multi-view Classification with Landmark-based SVM

by **Valentina Zantedeschi**, Rémi Emonet, Marc Sebban

as part of the ANR LIVES project (multiview)

MVL-SVM Principle

- Randomly select landmarks
  - L points $l_1, l_2, \cdots, l_L$ from the dataset
  - with no missing views
- Project all points on this landmarks
  - use an arbitrary μ similarity measure
- Learn a model (classifier)
  - in the joint projected space
  - fast and linear (non-linearity already in the projection)



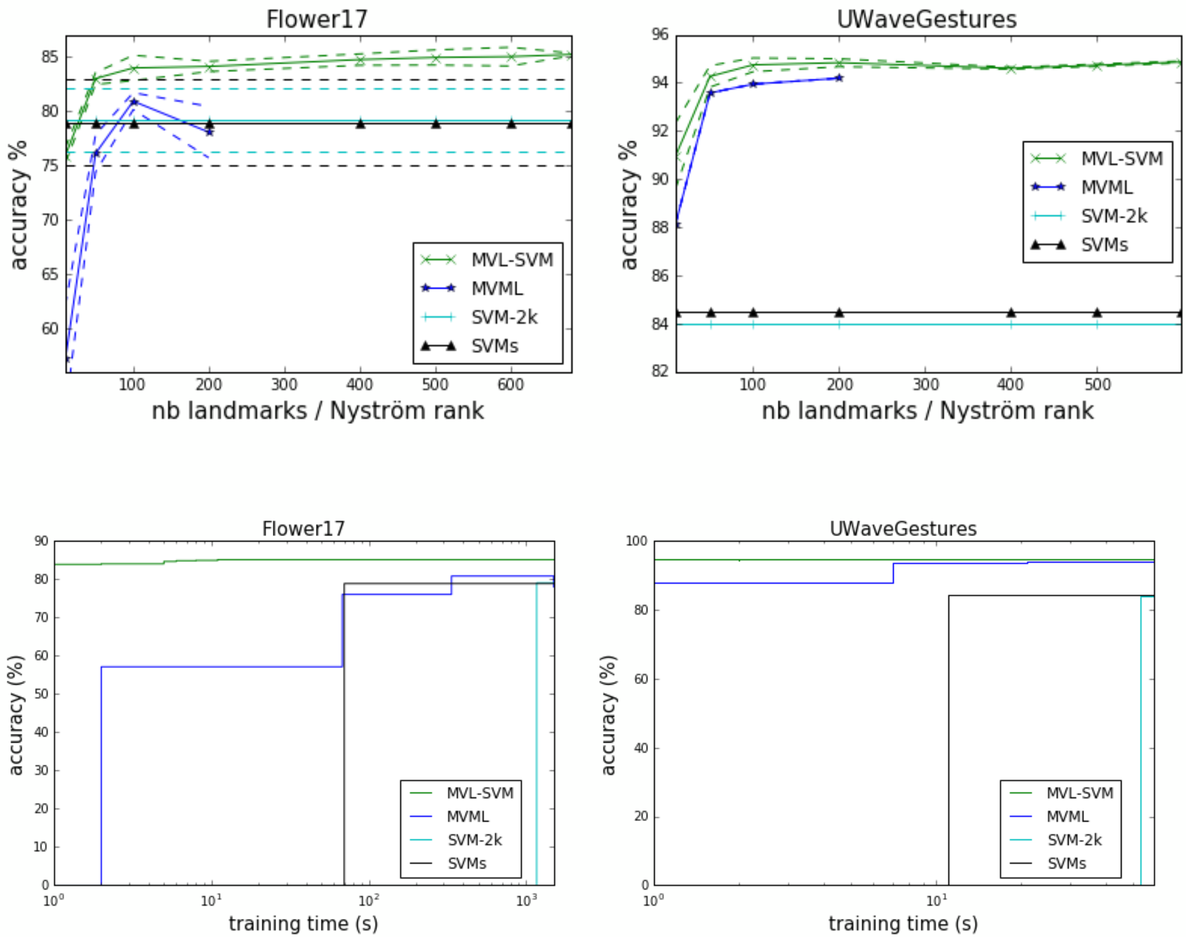
Generalization Bound

- The generalization bound of MVL-SVM, derived using the Uniform Stability framework:

$$R_{\mathcal{D}}(f) \leq \hat{R}_S(f) + \frac{cLVM^2}{m} + \left(2cLVM^2 + 1 + 2c\sqrt{LV}M\right)\sqrt{\frac{\ln\frac{1}{\delta}}{2m}}$$

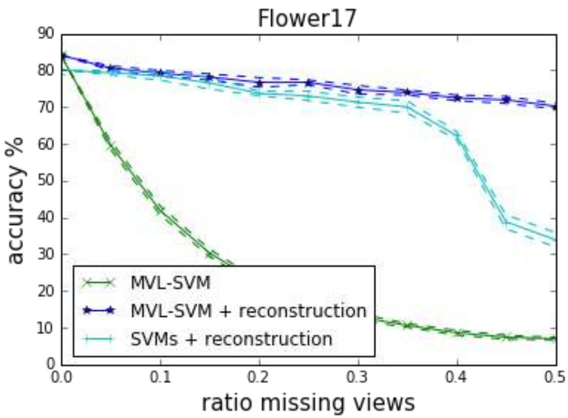
- L number of landmarks
- M number of views
- m number of samples
- NB
    - stable if $L \ll \dfrac{m}{V}$
    - the lower L, the more stable

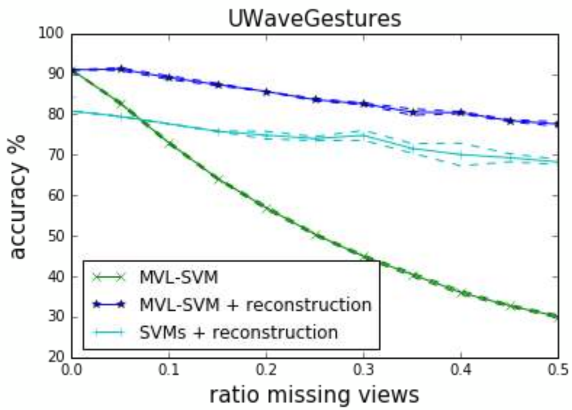
MVL-SVM Results



Missing Views?

- Landmark-based missing view reconstruction method
- Allow to maintain accuracy and scalability





In a nutshell

- Transfer learning has multiple facets
  - multi-task
  - multi-view
  - multi-domain
- Domain adaptation by bringing distributions together
  - by aligning subspaces obtained from PCA
  - non-linearly by using projection on landmarks
- Landmarks can also be used for multiview-learning
  - random landmark selection
  - non-linear projection on the landmarks
  - fast linear model

Supp. on source and target risk

Link the Target Risk to the Source?

$$
\begin{aligned}
R_{P_T}(h) &= \mathbf{E}_{(x^t,y^t)\sim P_T} \mathbf{I}\left[h(x^t) \neq y^t\right] \\
&= \mathbf{E}_{(x^t,y^t)\sim P_T} \frac{P_S(x^t,y^t)}{P_S(x^t,y^t)} \mathbf{I}\left[h(x^t) \neq y^t\right] \\
&= \sum_{(x^t,y^t)} P_T(x^t,y^t) \frac{P_S(x^t,y^t)}{P_S(x^t,y^t)} \mathbf{I}\left[h(x^t) \neq y^t\right] \\
&= \mathbf{E}_{(x^t,y^t)\sim P_S} \frac{P_T(x^t,y^t)}{P_S(x^t,y^t)} \mathbf{I}\left[h(x^t) \neq y^t\right]
\end{aligned}
$$

Supp. on covariate shift

Domain Adaptation – Covariate Shift?

- $R_{P_T}(h) \;=\; \mathbf{E}_{(x^t,y^t)\sim P_S} \frac{P_T(x^t,y^t)}{P_S(x^t,y^t)} \mathbf{I}\left[h(x^t) \neq y^t\right]$
- The target risk can be rewritten as an expectation on the source

Covariate Shift

- When $P_S(y^t|x^t) = P_T(y^t|x^t)$ (covariate shift assumption)
- Very strong assumption
- We can estimate a ratio between unlabeled data

$$
\begin{aligned}
R_{P_T}(h) &= \mathbf{E}_{(x^t,y^t)\sim P_S} \frac{D_T(x^t)P_T(y^t|x^t)}{D_S(x^t)P_S(y^t|x^t)} \mathbf{I}\left[h(x^t) \neq y^t\right] \\
&= \mathbf{E}_{(x^t,y^t)\sim P_S} \frac{D_T(x^t)}{D_S(x^t)} \mathbf{I}\left[h(x^t) \neq y^t\right]
\end{aligned}
$$

⇒ **Approach**: density estimation and instance re-weighting

Supp. on Subspace Alignment Results

Subspace Alignment – Experiments

- Comparison on visual domain adaptation tasks
  - adaptation from Office/Caltech-10 datasets (four domains to adapt)
  - adaptation on ImageNet, LabelMe and Caltech-256 datasets: one is used as source and one as target
- Other methods
  - Baseline 1: projection on the source subspace
  - Baseline 2: projection on the target subspace
  - 2 related methods:
    - GFS [Gopalan et al.,ICCV'11]
    - GFK [Gong et al., CVPR'12]

LABORATOIRE HUBERT CURIEN
UMR • CNRS • 5516 • SAINT-ETIENNE

Subspace Alignment – Results

- Office/Caltech-10 datasets

Method	C→ A	D→ A	W→ A	A→ C	D→ C	W→ C
NA	21.5	26.9	20.8	22.8	24.8	16.4
Baseline 1	38.0	29.8	35.5	30.9	29.6	31.3
Baseline 2	**40.5**	33.0	**38.0**	33.3	31.2	31.9
GFS [8]	36.9	32	27.5	35.3	29.4	21.7
GFK [7]	36.9	32.5	31.1	**35.6**	29.8	27.2
OUR	39.0	**38.0**	37.4	35.3	**32.4**	**32.3**
Method	A→ D	C→ D	W→ D	A→ W	C→ W	D→ W
NA	22.4	21.7	40.5	23.3	20.0	53.0
Baseline 1	34.6	37.4	71.8	35.1	33.5	74.0
Baseline 2	34.7	36.4	72.9	36.8	34.4	78.4
GFS [8]	30.7	32.6	54.3	31.0	30.6	66.0
GFK [7]	35.2	35.2	70.6	34.4	33.7	74.9
OUR	**37.6**	**39.6**	**80.3**	**38.6**	**36.8**	**83.6**

Table 2. Recognition accuracy with unsupervised DA using a NN classif er (Off ce dataset + Caltech10).

Method	C→ A	D→ A	W→ A	A→ C	D→ C	W→ C
Baseline 1	44.3	36.8	32.9	36.8	29.6	24.9
Baseline 2	44.5	38.6	34.2	37.3	31.6	28.4
GFK	44.8	37.9	37.1	38.3	31.4	29.1
OUR	**46.1**	**42.0**	**39.3**	**39.9**	**35.0**	**31.8**
Method	A→ D	C→ D	W→ D	A→ W	C→ W	D→ W
Baseline 1	36.1	38.9	73.6	**42.5**	34.6	75.4
Baseline 2	32.5	35.3	73.6	37.3	34.2	80.5
GFK	37.9	36.1	74.6	39.8	34.9	79.1
OUR	**38.8**	**39.4**	**77.9**	39.6	**38.9**	**82.3**

Table 3. Recognition accuracy with unsupervised DA using a SVM classif er(Off ce dataset + Caltech10).

- ImageNet (I), LabelMe (L) and Caltech-256 (C) datasets

Method	L→ C	L→ I	C→ L	C→ I	I→ L	I→ C	AVG
NA	46.0	38.4	29.5	31.3	36.9	45.5	37.9
Baseline1	24.2	27.2	46.9	41.8	35.7	33.8	34.9
Baseline2	24.6	27.4	**47.0**	**42.0**	35.6	33.8	35.0
GFK	24.2	26.8	44.9	40.7	35.1	33.8	34.3
OUR	**49.1**	**41.2**	**47.0**	39.1	**39.4**	**54.5**	**45.0**

Table 4. Recognition accuracy with unsupervised DA with NN classif er (ImageNet (I), LabelMe (L) and Caltech-256 (C)).

Method	L→ C	L→ I	C→ L	C→ I	I→ L	I→ C	AVG
NA	49.6	40.8	36.0	45.6	41.3	58.9	45.4
Baseline1	50.5	42.0	39.1	48.3	44.0	59.7	47.3
Baseline2	48.7	41.9	39.2	48.4	43.6	58.0	46.6
GFK	52.3	43.5	39.6	49.0	45.3	61.8	48.6
OUR	**52.9**	**43.9**	**43.8**	**50.9**	**46.3**	**62.8**	**50.1**

Table 5. Recognition accuracy with unsupervised DA with SVM classif er (ImageNet (I), LabelMe (L) and Caltech-256 (C)).

Attribution


CC by genevieveromier (Flickr)
CC by genevieveromier (Flickr)


CC by sgillies (Flickr)
CC by sgillies (Flickr)