Domain adaptation: How far are we from the solution ?

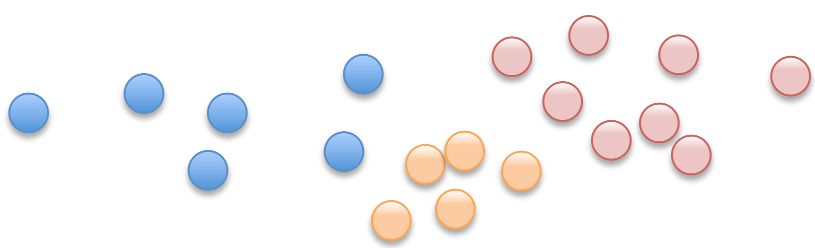
Tinne Tuytelaars
KU Leuven

With the help of:
Basura Fernando and Tatiana Tommasi

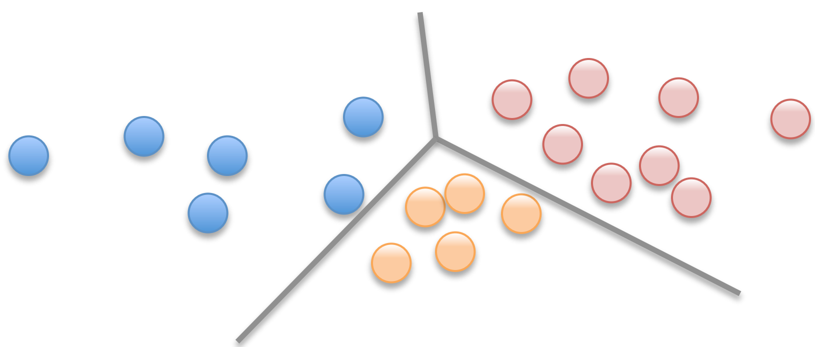
Outline

1. Update on our work on unsupervised DA by subspace alignment
2. Study of interplay between image representations and DA
3. Discussion: how far are we from solving the problem ?

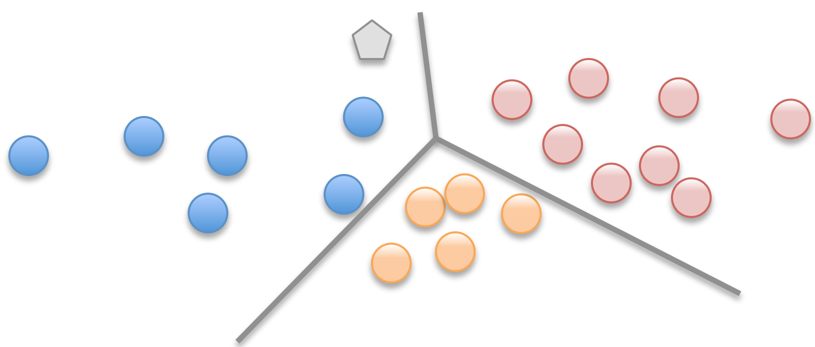
Domain adaptation: basics



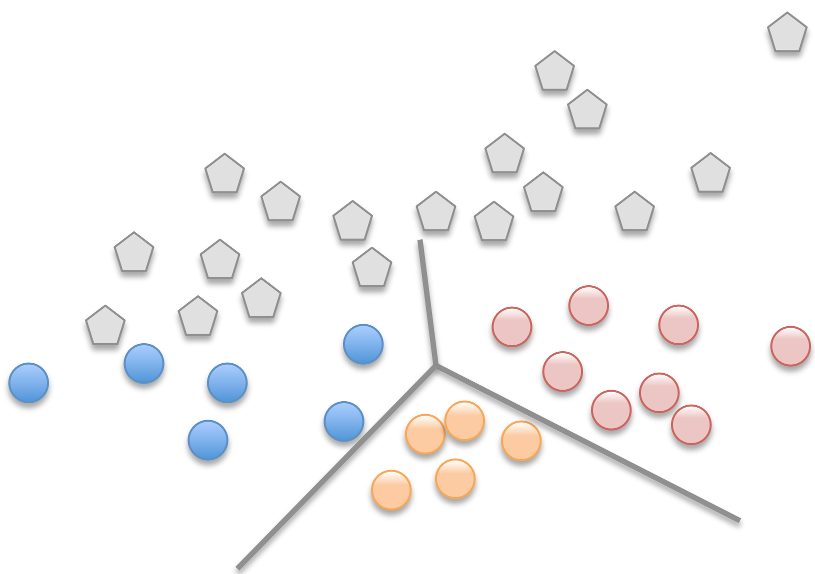
Domain adaptation: basics



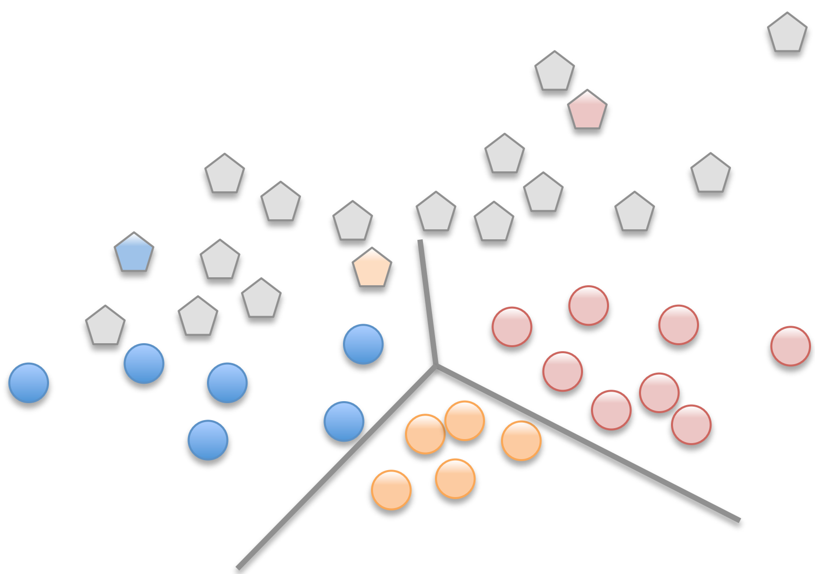
Domain adaptation: basics



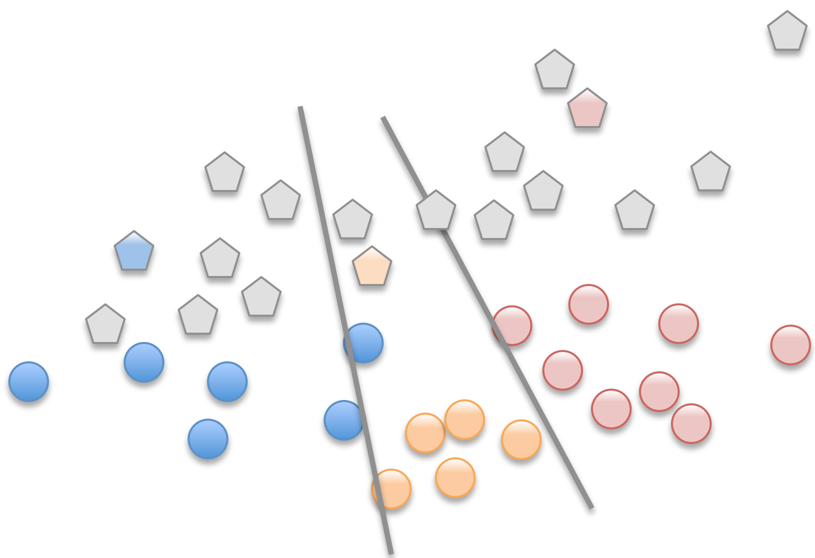
Domain adaptation: basics



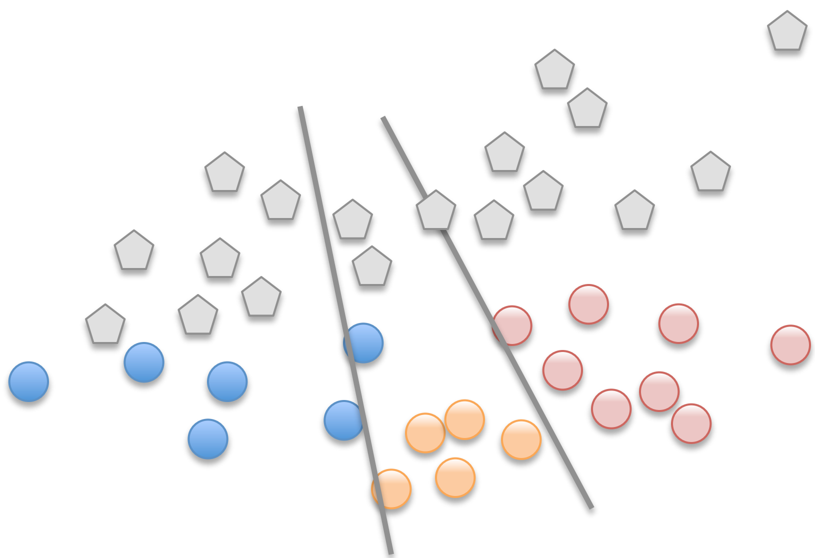
Domain adaptation: basics



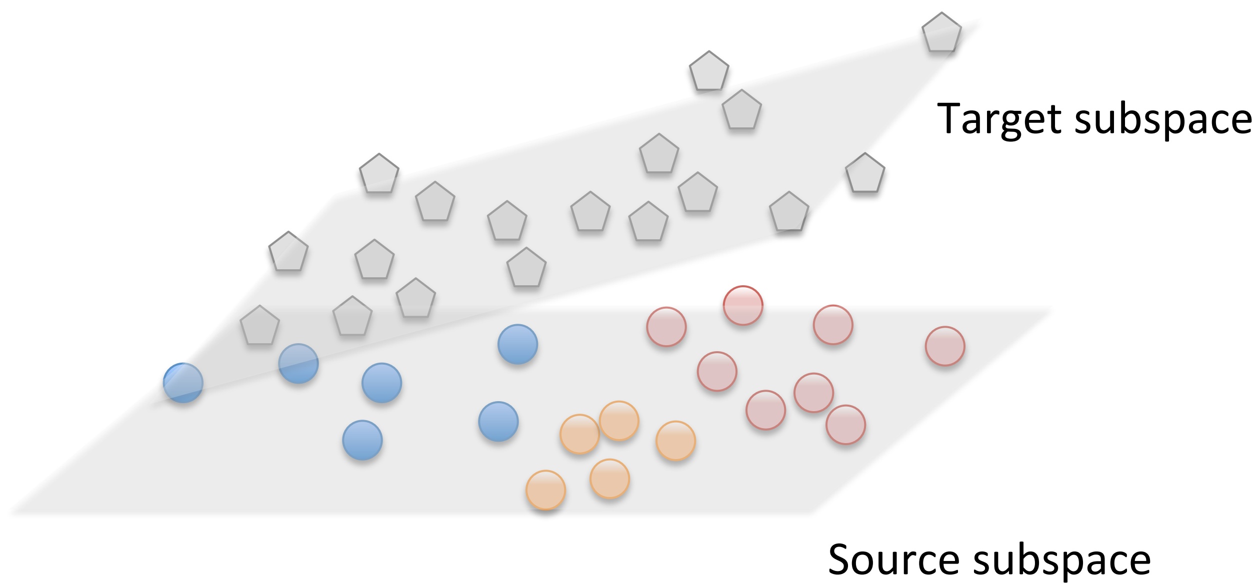
Domain adaptation: basics



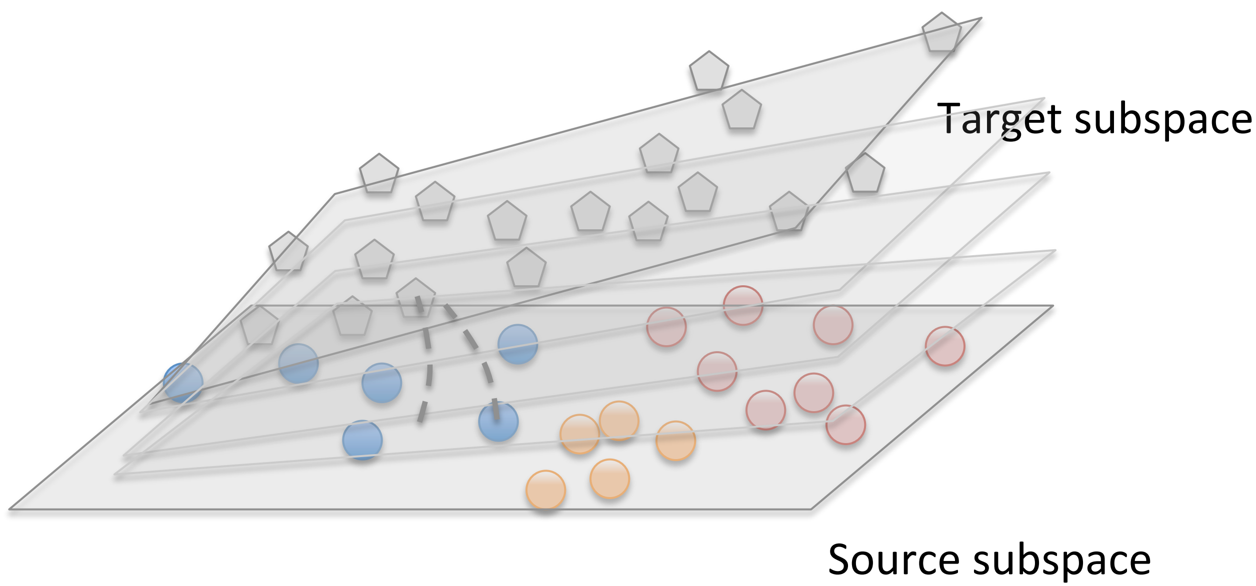
Domain adaptation: basics



Domain adaptation: subspace based methods

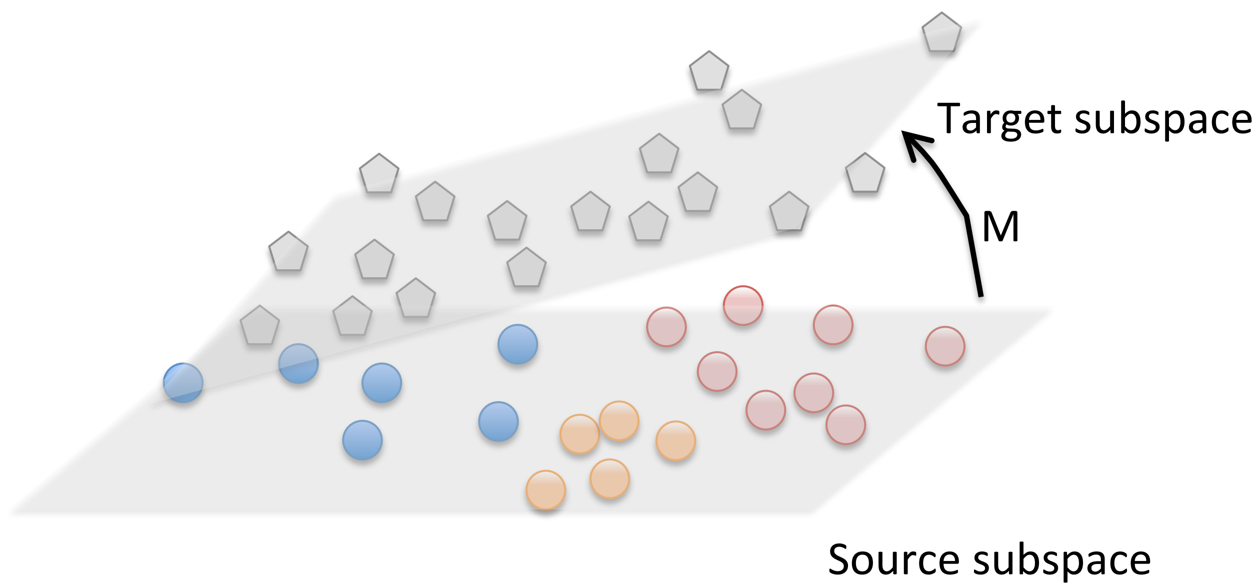


Domain adaptation: GFS and GFK

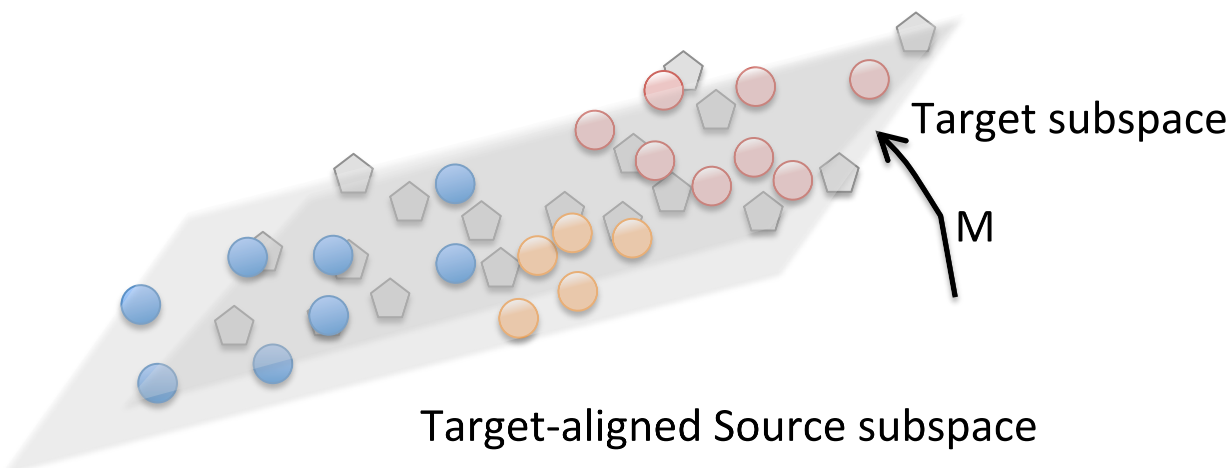


R. Gopalan, R. Li, R. Chellapa, Domain Adaptation for Object Recognition: An Unsupervised Approach, ICCV11
B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic Flow Kernel for Unsupervised Domain Adaptation, CVPR12

Domain adaptation: subspace alignment



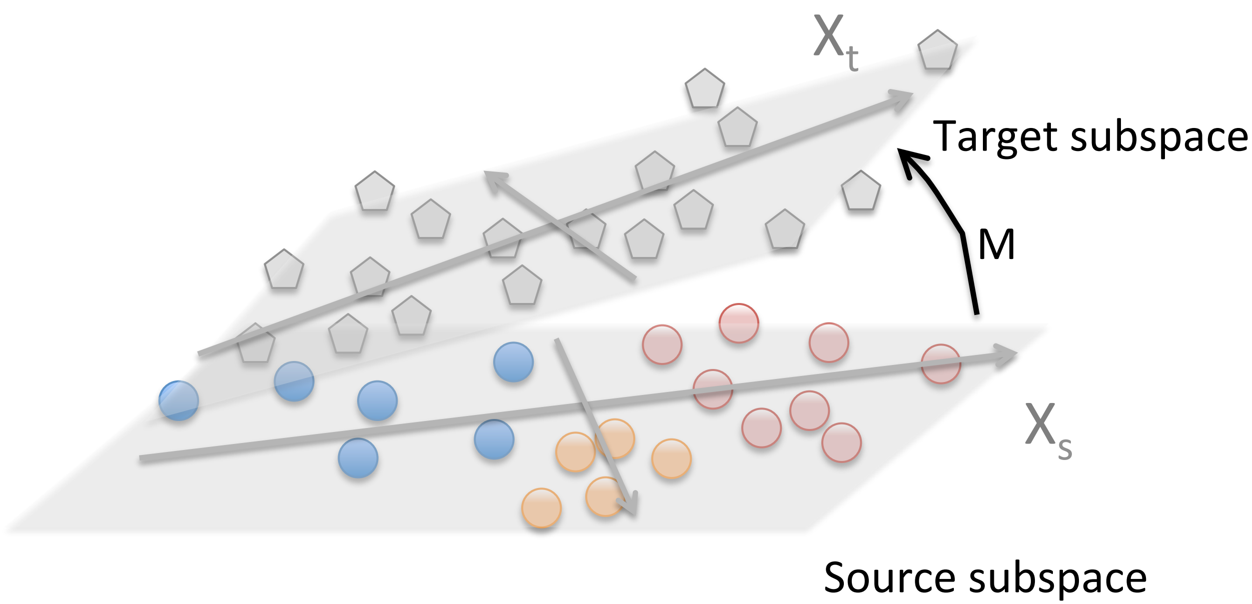
Domain adaptation: subspace alignment



Domain adaptation: subspace alignment

Minimize $F(M) = ||X_S M - X_T||_F^2$

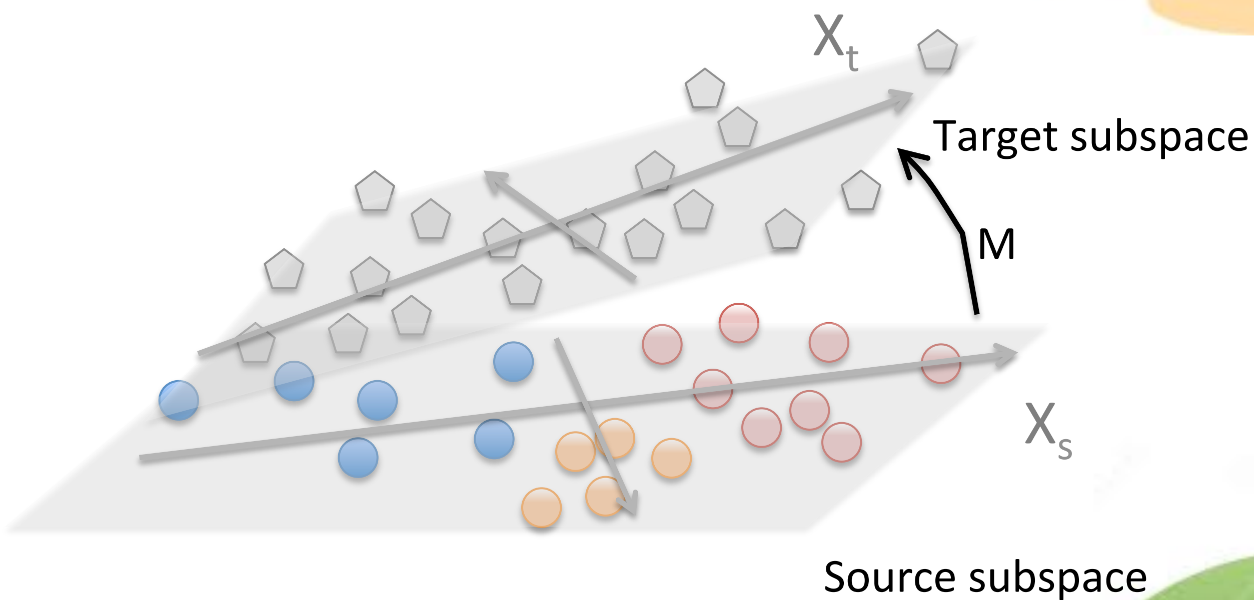
$$M^* = X_S' X_T$$



Domain adaptation: subspace alignment

Minimize $F(M) = ||X_S M - X_T||_F^2$

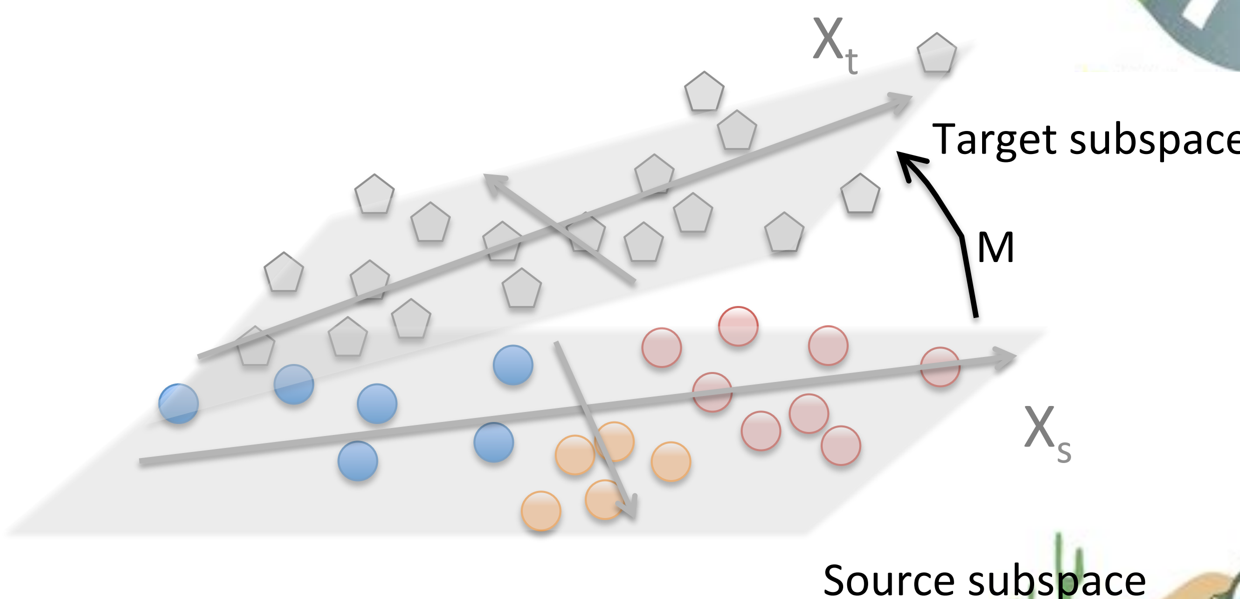
$$M^* = X_S' X_T$$



Domain adaptation: subspace alignment

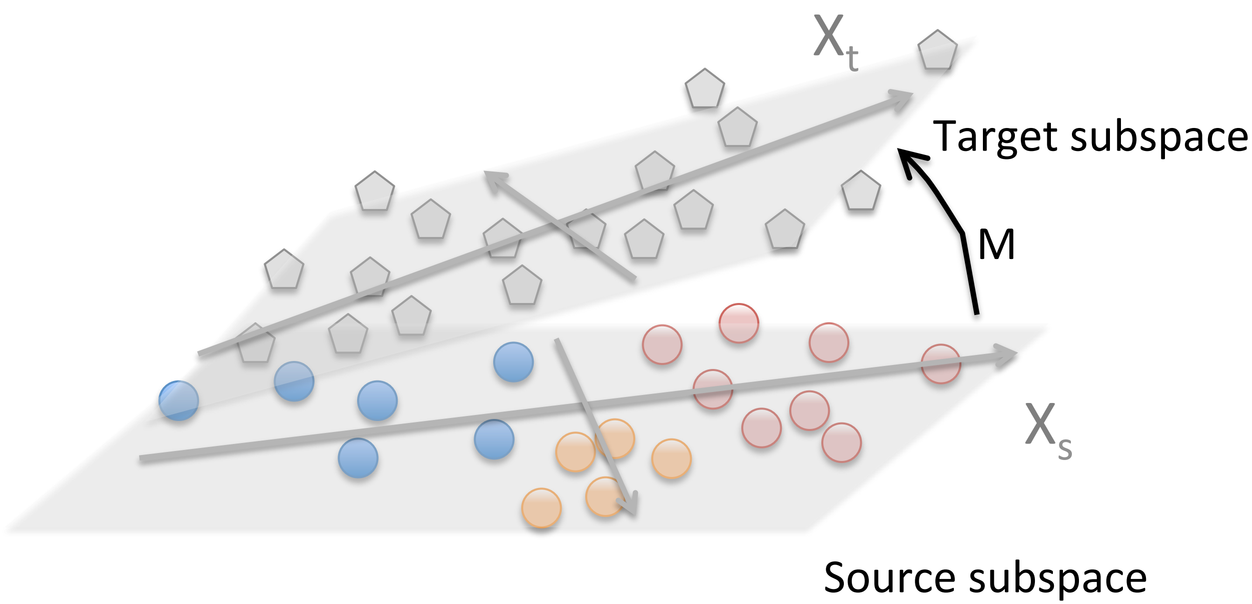
Minimize $F(M) = ||X_S M - X_T||_F^2$

$M^* = X_S' X_T$



Domain adaptation: subspace alignment

$$\begin{aligned} Sim(\mathbf{y_S},\mathbf{y_T}) &= (\mathbf{y_S}X_SM^*)(\mathbf{y_T}X_T)' = \mathbf{y_S}X_SM^*X_T'\mathbf{y_T}' \\ &= \mathbf{y_S}A\mathbf{y_T}', \end{aligned}$$



Office / Caltech10 dataset

(SURF – BoW)

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-S	38.0	29.8	35.5	30.9	29.6	31.3
Baseline-T	**40.5**	33.0	**38.0**	33.3	31.2	31.9
GFS	36.9	32	27.5	35.3	29.4	21.7
GFK	36.9	32.5	31.1	**35.6**	29.8	27.2
TCA	34.7	27.5	34.1	28.8	28.8	30.5
SA	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-S	34.6	37.4	71.8	35.1	33.5	74.0
Baseline-T	34.7	36.4	72.9	36.8	34.4	78.4
GFS	30.7	32.6	54.3	31.0	30.6	66.0
GFK	35.2	35.2	70.6	34.4	33.7	74.9
TCA	30.4	34.7	64.4	30.3	28.8	70.9
SA	**37.6**	**39.6**	**80.3**	**38.6**	**36.8**	**83.6**

Table 2 Recognition accuracy with unsupervised DA using NN classifier (Office dataset + Caltech10).

TCA: S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain Adaptation via Transfer Component Analysis, IJCAI09
GFK: B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic Flow Kernel for Unsupervised Domain Adaptation, CVPR12

Office / Caltech10 dataset

(SURF – BoW)

Method	C→A	D→A	W→A	A→C	D→C	W→C
Baseline-S	44.3	36.8	32.9	36.8	29.6	24.9
Baseline-T	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
TCA	**47.2**	38.8	34.8	**40.8**	33.8	30.9
SA	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-S	36.1	38.9	73.6	**42.5**	34.6	75.4
Baseline-T	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
TCA	36.4	39.2	72.1	38.1	36.5	80.3
SA	**38.8**	**39.4**	**77.9**	39.6	**38.9**	**82.3**

Table 3 Recognition accuracy with unsupervised DA using SVM classifier(Office dataset + Caltech10).

TCA: S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain Adaptation via Transfer Component Analysis, IJCAI09
GFK: B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic Flow Kernel for Unsupervised Domain Adaptation, CVPR12

Office / Caltech10 dataset
(SURF – BoW)

Method	NA	Baseline-S	Baseline-T	GFK	SA
TDAS	1.25	3.34	2.74	2.84	**4.26**
HΔH	98.1	99.0	99.0	74.3	**53.2**

Table 1 Several distribution discrepancy measures averaged over 12 DA problems using Office dataset.

ImageNet, LabelMe and Caltech256

(SURF – BoW)

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
Baseline-S	24.2	27.2	46.9	41.8	35.7	33.8	34.9
Baseline-T	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
TCA	25.7	27.5	43.1	38.8	29.6	26.8	31.9
SA	**49.1**	**41.2**	**47.0**	39.1	**39.4**	**54.5**	**45.0**

Table 4 Recognition accuracy with unsupervised DA with NN classifier (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
Baseline-S	50.5	42.0	39.1	48.3	44.0	59.7	47.3
Baseline-T	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
TCA	46.7	39.4	37.9	47.2	41.0	56.9	44.9
SA	**52.9**	**43.9**	**43.8**	**50.9**	**46.3**	**62.8**	**50.1**

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

ImageNet – Pascal VOC
(SURF – BoW)

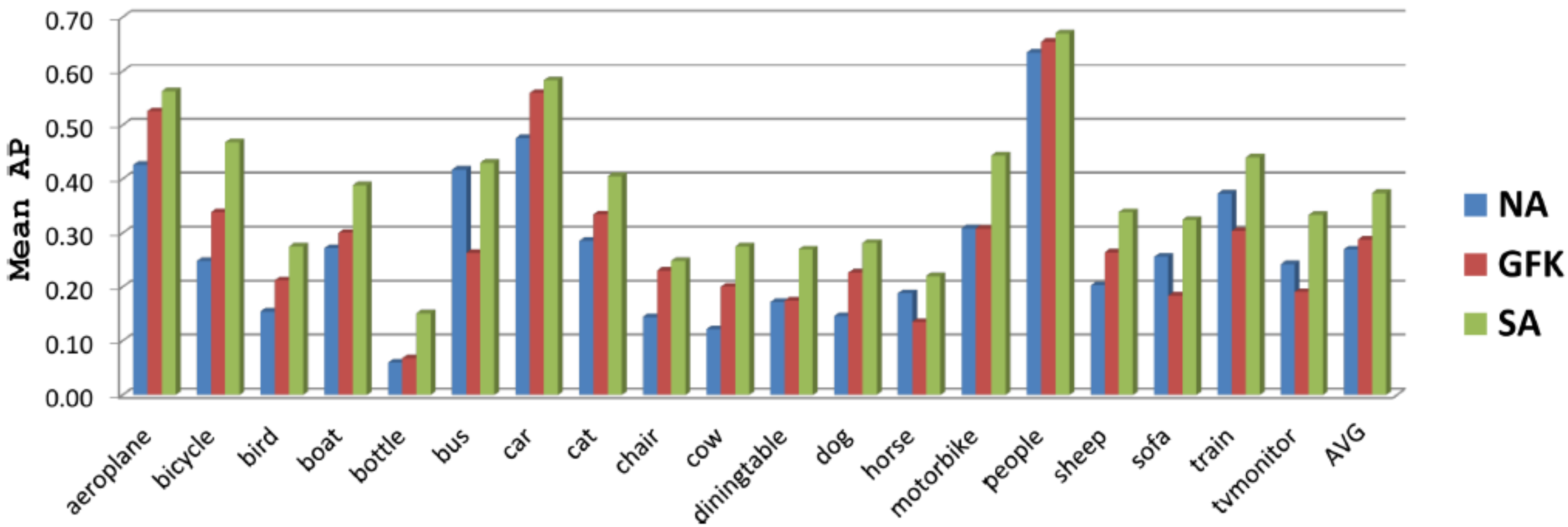


Fig. 6 Train on ImageNet and classify PASCAL-VOC-2007 images using unsupervised DA with a linear SVM classifier. Average average precision over 20 object classes is reported.

Further analysis

- How to determine subspace dimensionality ?
- How to best learn the subspaces ?
- What's the optimal representation ?

How to determine the subspace dimensionality ?

ICCV2013: upper bound based on consistency theorem + crossvalidation

- Nice from a theoretical point of view, but ...
- Can be computationally expensive in high-dimensional spaces (e.g. Fisher Vectors)

How to determine the subspace dimensionality ?

Alternative: find optimal dimensionality using MLE

- Choose the *domain intrinsic dimensionality*

$$\widehat{d}(\mathbf{y}) = \left[\frac{1}{N(R,\mathbf{y})} \sum_{j=1}^{N(R,\mathbf{y})} \log \frac{R}{\Theta_j(\mathbf{y})} \right]^{-1}$$

- Can be done for source and target separately

- Compute Eucl. distance between projections

$$D_{SA-MLE}(\mathbf{y}_s, \mathbf{y}_t) = ||\mathbf{y}_s X_S M - \mathbf{y}_t X_T||_2.$$

E. Levina and P. J. Bickel, Maximum Likelihood Estimation of Intrinsic Dimension, NIPS 2004

Office

(SURF Fisher Vectors)

Method	D → W	W → D	A → W	AVG
NA	62.7 ± 1.1	64.7 ± 2.2	17.1 ± 2.1	48.2 ± 1.7
GFK	44.1 ± 2.1	44.3 ± 2.7	16.3 ± 3.3	34.9 ± 2.8
TCA	14.1 ± 2.5	24.3 ± 1.1	10.7 ± 1.7	16.4 ± 1.8
SA*	49.7 ± 0.6	51.3 ± 2.1	**19.4 ± 1.1**	40.1 ± 1.1
SA-MLE	**68.9 ± 2.1**	**68.0 ± 1.3**	16.7 ± 1.4	**51.2 ± 1.6**

Table 7 Recognition accuracy% obtained on the Office dataset when images are encoded with SURF features and Fisher Vectors of 64 GMM components. *Note that due to computational complexity we use the subspace dimensionality obtained by subspace disagreement measure [12].

TCA: S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain Adaptation via Transfer Component Analysis, IJCAI09
B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic Flow Kernel for Unsupervised Domain Adaptation, CVPR12

How to best learn the subspaces ?

- Exploit class labels available for Source data
- Linear Discriminant Analysis or Partial Least Squares ?
  - Max. dimensionality is nb of class labels (LDA)
  - Bigger discrepancy between source and target subspaces if we use different algorithm for both
  - PLS used in Geodesic Flow Kernel work, but no good results in our experiments

B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic Flow Kernel for Unsupervised Domain Adaptation, CVPR12

How to best learn the subspaces ?

- Apply ITML metric learning PRIOR to PCA

Data: Source data S, Source labels L_S
Result: Source subspace X_s
1. Learn the source metric $W \leftarrow ITML(S, L_S)$;
2. Apply cholesky-decomposition to W. $W_c \leftarrow chol(W)$;
3. Project source data to W_c induced space. $S_w \leftarrow SW_c$;
4. $X_s \leftarrow PCA(S_w)$;

J.V. Davis, B. Kulis, P. Jain, S. Sra, I.S. Dhillon, Information Theoretic Metric Learning, ICML07

Office / Caltech10 dataset
(SURF – BoW)

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-S	38.0	29.8	35.5	30.9	29.6	31.3
Baseline-T	**40.5**	33.0	**38.0**	33.3	31.2	31.9
GFS	36.9	32	27.5	35.3	29.4	21.7
GFK	36.9	32.5	31.1	**35.6**	29.8	27.2
TCA	34.7	27.5	34.1	28.8	28.8	30.5
SA	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-S	34.6	37.4	71.8	35.1	33.5	74.0
Baseline-T	34.7	36.4	72.9	36.8	34.4	78.4
GFS	30.7	32.6	54.3	31.0	30.6	66.0
GFK	35.2	35.2	70.6	34.4	33.7	74.9
TCA	30.4	34.7	64.4	30.3	28.8	70.9
SA	**37.6**	**39.6**	**80.3**	**38.6**	**36.8**	**83.6**

Table 2 Recognition accuracy with unsupervised DA using NN classifier (Office dataset + Caltech10).

TCA: S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain Adaptation via Transfer Component Analysis, IJCAI09
GFK: B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic Flow Kernel for Unsupervised Domain Adaptation, CVPR12

Office / Caltech10 dataset

(SURF – BoW)

Method	C→A	D→A	W→A	A→C	D→C	W→C
GFK (PLS,PCA)	40.4	36.2	35.5	37.9	32.7	29.3
GFK (LDA,PCA)	41.6	36.2	39.9	31.9	31.0	36.7
TCA (LDA,PCA)	35.0	28.2	32.7	28.3	28.4	30.0
SA (ITML-PCA,PCA)	47.1	**40.0**	**42.4**	**41.1**	**38.1**	**39.8**
SA (LDA,PCA)	**48.3**	37.6	41.6	35.7	34.3	39.2
Method	**A→D**	**C→D**	**W→D**	**A→W**	**C→W**	**D→W**
GFK (PLS,PCA)	35.1	**41.1**	71.2	35.7	35.8	79.1
GFK (LDA,PCA)	35.5	37.1	68.9	37.0	37.1	76.9
TCA (LDA,PCA)	35.5	37.1	68.9	37.0	37.1	76.9
SA (ITML-PCA,PCA)	**43.7**	40.4	**83.0**	**43.5**	42.8	**84.5**
SA (LDA,PCA)	32.0	34.0	58.6	35.2	**46.0**	73.6

Table 6 Recognition accuracy with unsupervised DA using NN classifier (Office dataset + Caltech10) using the supervised dimensionality reduction in the source domain.

TCA: S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain Adaptation via Transfer Component Analysis, IJCAI09

GFK: B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic Flow Kernel for Unsupervised Domain Adaptation, CVPR12

What's the best representation ?

- SURF Bag-of-Words
- SURF Fisher Vectors
- CNN-based features (Decaf)

Office

(SURF Fisher Vectors)

Method	D → W	W → D	A → W	AVG
NA	62.7 ± 1.1	64.7 ± 2.2	17.1 ± 2.1	48.2 ± 1.7
GFK	44.1 ± 2.1	44.3 ± 2.7	16.3 ± 3.3	34.9 ± 2.8
TCA	14.1 ± 2.5	24.3 ± 1.1	10.7 ± 1.7	16.4 ± 1.8
SA*	49.7 ± 0.6	51.3 ± 2.1	**19.4 ± 1.1**	40.1 ± 1.1
SA-MLE	**68.9 ± 2.1**	**68.0 ± 1.3**	16.7 ± 1.4	**51.2 ± 1.6**

Table 7 Recognition accuracy% obtained on the Office dataset when images are encoded with SURF features and Fisher Vectors of 64 GMM components. *Note that due to computational complexity we use the subspace dimensionality obtained by subspace disagreement measure [12].

TCA: S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain Adaptation via Transfer Component Analysis, IJCAI09
B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic Flow Kernel for Unsupervised Domain Adaptation, CVPR12

Office
(Decaf)

Method	D→W	W→D	A→W
NA	86.4 ± 1.1	88.6 ± 1.2	42.8 ± 0.9
GFK	69.2 ± 2.1	61.8 ± 2.3	39.5 ± 2.1
TCA	57.1 ± 1.7	51.1 ± 1.8	24.8 ± 3.2
SA	**86.8 ± 1.0**	**89.3 ± 0.8**	**44.7 ± 0.7**

Table 8 Recognition accuracy with unsupervised DA using NN classifier on Office dataset using $Decaf_6$ features.

Method	D→W	W→D	A→W
NA	91.3 ± 1.2	91.6 ± 1.6	**47.9 ± 2.9**
GFK	87.2 ± 1.3	88.1 ± 1.5	46.8 ± 1.8
TCA	89.0 ± 1.4	87.9 ± 1.9	44.6 ± 3.0
SA	**91.8 ± 0.9**	**92.4 ± 1.7**	47.2 ± 1.5

Table 9 Recognition accuracy with unsupervised DA using SVM classifier on Office dataset using $Decaf_6$ features.

TCA: S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain Adaptation via Transfer Component Analysis, IJCAI09
GFK: B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic Flow Kernel for Unsupervised Domain Adaptation, CVPR12

Office + Caltech 10 (SURF + BoW)

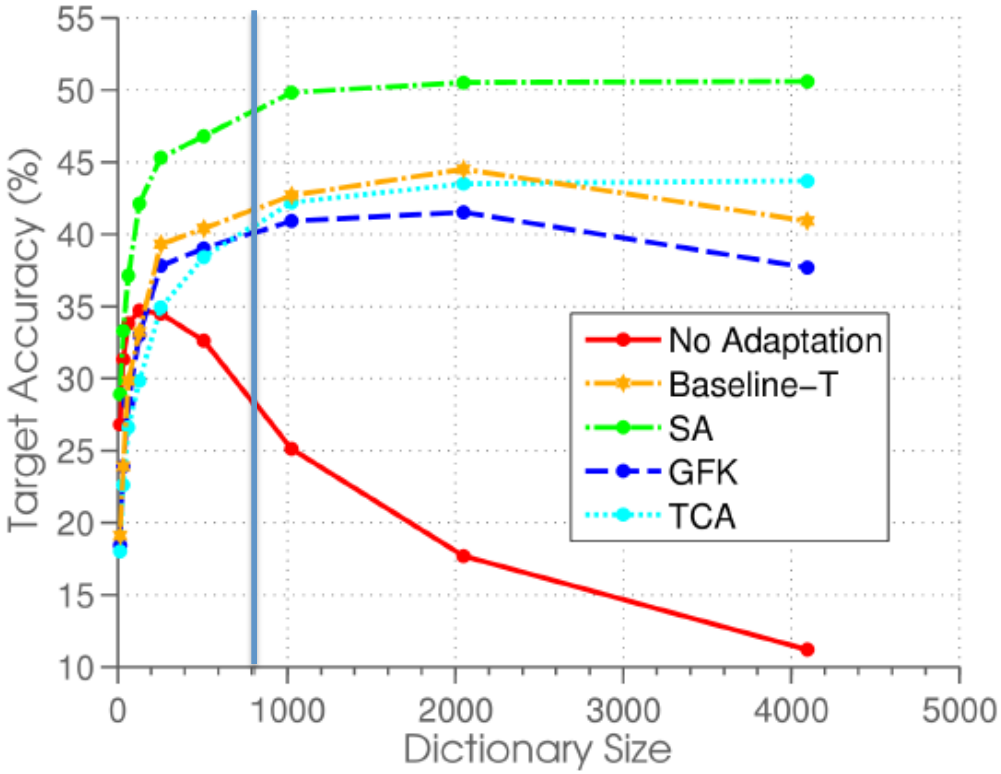


Fig. 7 Mean accuracy on target domain using NN classifier for different dictionary sizes on *Office+Caltech-10* dataset (unsupervised domain adaptation).

Office + Caltech10
(SURF – BoW)

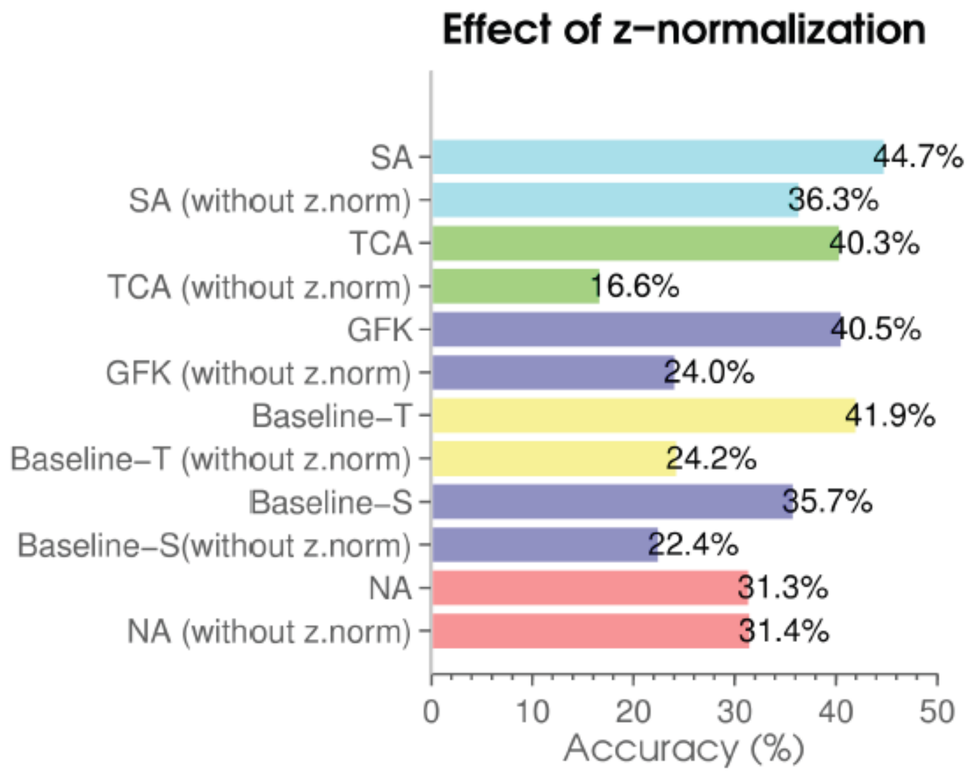


Fig. 8 The effect of z-normalization on subspace-based DA methods. Mean classification accuracy on 12 DA problems on Office+Caltech dataset is shown with and without z-norm.

TCA: S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain Adaptation via Transfer Component Analysis, IJCAI09
GFK: B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic Flow Kernel for Unsupervised Domain Adaptation, CVPR12

So, how far are we from the solution ?

- Dataset bias
- Domain adaptation

So, how far are we from the solution ?

- Dataset bias
  - Most problems can be solved by choosing a robust representation (learnt from a superset)
  - DA dataset setups not sufficient for studying DA algorithms

So, how far are we from the solution ?

- Dataset bias
- Other problems
  - DA from Amazon images to Caltech/DLSR/Webcam
  - DA from images to (very) old pictures
  - DA from images to paintings, cartoons or sketches
  - DA from images to multispectral
  - …
- Real world problems
  - incl. non-overlapping classes between source and target, different class priors
- Go beyond classification: detection, attributes, pose, …

Thank you !

Questions ?