Unsupervised Feature Selection by Preserving Stochastic Neighbors: Supplemental Material

Table 1: Symbol definitions

Symbol	Definition
$\mathbf{x}_i \in \mathcal{R}^D$	Feature vector of the i-th sample
S_{ij}	Similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$ before feature selection
s_{ij}	Similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$ after feature selection
p_{ij}	The probability that $\mathbf{x}_j$ is $\mathbf{x}_i$'s neighbor
$\mathbf{p}_i$	The probability distribution of other $\mathbf{x}_j$ ($j = 1, \ldots, n$) being $\mathbf{x}_i$'s neighbor before feature selection
$\mathbf{q}_i$	The probability distribution of other $\mathbf{x}_j$ ($j = 1, \ldots, n$) being $\mathbf{x}_i$'s neighbor after feature selection
$\mathbf{w} \in \{0,1\}^D$	Feature selection indicator vector
$N_{(1-\alpha)}$	The number of features that have scores greater than $(1-\alpha)$
$\mathcal{R}_w$	Set of restricted variables in optimization
$\mathcal{F}_w$	Set of free variables in optimization

1 Notations

A summary of symbols used in this paper is shown in Table 1.

2 Optimization

2.1 Gradient Derivation

In the following, we derive the gradient update formula for SNFS. To make the derivation less cluttered, we denote $\exp(s_{ij}/\sigma^2)$ as A_{ij} and the normalization term $\sum_{k\neq i}\exp(s_{ik}/\sigma^2)$ as Z_i. So q_{ij} can be denoted as A_{ij}/Z_i.

Denote $KL(\mathbf{p}_i||\mathbf{q}_i) = \sum_{j\neq i} p_{ij}\log\frac{p_{ij}}{q_{ij}}$ as $\mathcal{L}_i$. The gradient of $\mathcal{L}_i$ w.r.t s_{ij} can be decomposed into two parts.

$$\frac{\partial \mathcal{L}_i}{\partial s_{ij}} = -\frac{\partial}{\partial s_{ij}}(p_{ij}\log q_{ij}) - \sum_{k\neq j}\frac{\partial}{\partial s_{ij}}(p_{ik}\log q_{ik}) \quad (1)$$

We work on these two parts individually.

$$\begin{aligned}\frac{\partial}{\partial s_{ij}}(p_{ij}\log q_{ij}) &= p_{ij}/q_{ij}\cdot\frac{\partial q_{ij}}{\partial A_{ij}}\frac{\partial A_{ij}}{\partial s_{ij}}\\ &= p_{ij}/q_{ij}\frac{Z_i - A_{ij}}{Z_i^2}\cdot\frac{\partial A_{ij}}{\partial s_{ij}}\\ &= p_{ij}\frac{1}{A_{ij}}\frac{Z_i - A_{ij}}{Z_i}\cdot\frac{\partial A_{ij}}{\partial s_{ij}}\\ &= p_{ij}\frac{1}{A_{ij}}\cdot\frac{\partial A_{ij}}{\partial s_{ij}} - p_{ij}\frac{1}{Z_i}\cdot\frac{\partial A_{ij}}{\partial s_{ij}}\end{aligned} \quad (2)$$

$$\begin{aligned}\sum_{k\neq j}\frac{\partial}{\partial s_{ij}}(p_{ik}\log q_{ik}) &= \sum_{k\neq j} p_{ik}/q_{ik}\cdot\frac{\partial q_{ik}}{\partial A_{ij}}\frac{\partial A_{ij}}{\partial s_{ij}}\\ &= \sum_{k\neq j} -p_{ik}/q_{ik}\frac{A_{ik}}{Z_i^2}\cdot\frac{\partial A_{ij}}{\partial s_{ij}}\\ &= \sum_{k\neq j} -p_{ik}\frac{1}{Z_i}\cdot\frac{\partial A_{ij}}{\partial s_{ij}}\end{aligned} \quad (3)$$

We combine them together and get the following by observing $p_{ij} + \sum_{k\neq j} p_{ik} = 1$ and $\frac{\partial A_{ij}}{\partial s_{ij}} = A_{ij}/\sigma^2$.

$$\begin{aligned}\frac{\partial \mathcal{L}_i}{\partial s_{ij}} =& -p_{ij}\frac{1}{A_{ij}}\cdot\frac{\partial A_{ij}}{\partial s_{ij}} + p_{ij}\frac{1}{Z_i}\cdot\frac{\partial A_{ij}}{\partial s_{ij}}\\ &+\sum_{k\neq j} p_{ik}\frac{1}{Z_i}\cdot\frac{\partial A_{ij}}{\partial s_{ij}}\\ =& -(p_{ij}\frac{1}{A_{ij}} - \frac{1}{Z_i})\frac{\partial A_{ij}}{\partial s_{ij}}\\ =& -(p_{ij} - q_{ij})/\sigma^2\end{aligned} \quad (4)$$

It is easy to get the following gradient:

$$\frac{\partial s_{ij}}{\partial w_t} = x_{it}x_{jt} \quad (5)$$

Therefore, the gradient of loss function $\mathcal{L}$ w.r.t w_t (for $w_t > 0$) is calculated as follows.

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial w_t} &= \sum_{i=1}^{n}\sum_{j\neq i}\frac{\partial \mathcal{L}_i}{\partial s_{ij}}\frac{\partial s_{ij}}{\partial w_t} + \lambda\frac{\partial |w_t|}{\partial w_t}\\ &= -\sum_{i=1}^{n}\sum_{j\neq i}(p_{ij} - q_{ij})x_{it}x_{jt}/\sigma^2 + \lambda\end{aligned} \quad (6)$$

Though it takes several steps to get this gradient, the final result is simple. If we use negative euclidean distance as the similarity measure, one can derive the following gradient formula in a similar manner:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial w_t} &= \sum_{i=1}^{n}\sum_{j\neq i}\frac{\partial \mathcal{L}}{\partial s_{ij}}\frac{\partial s_{ij}}{\partial w_t} + \lambda\frac{\partial |w_t|}{\partial w_t} \\ &= \sum_{i=1}^{n}\sum_{j\neq i}(p_{ij}-q_{ij})(x_{it}-x_{jt})^2/\sigma^2 + \lambda\end{aligned} \tag{7}$$

3 Experiment

3.1 Datasets

We use six publicly available datasets: BBC and BBCSport news dataset[1], Guardian news dataset[2], BlogCatalog[3] blog-posts dataset, Newsgroup [4] and TDT2[5],.

The baseline methods UDFS and RSFS are prohibitively slow for large datasets. The original data of the latter three datasets are too large and therefore we use a subset of them.

- BBC: It consists of 2225 documents from the BBC news website corresponding to stories in five topical areas from 2004-2005. Each document belongs to one of 5 classes (*business, entertainment, politics, sport, tech*).
- BBCSport: It consists of 737 documents from the BBC Sport website corresponding to sports news articles in five topical areas from 2004-2005. The dataset has 5 classes: *athletics, cricket, football, rugby, tennis*.
- Guardian: 302 news stories from the period February - April 2009. Each story is annotated with one or more of the six topical labels based on the dominant topic: *business, entertainment, health, politics, sport, tech*.
- BlogCatalog: A subset of users' blogposts from BlogCatalog in the following categories (100 posts for each category): *cycling, military, architecture, commodities/futures, vacation rentals*
- Newsgroup: A subset of Newsgroup dataset on four topics: *comp.graphics, rec.sport.baseball, rec.motorcycles, sci.electronics*
- TDT2: The original corpus (Nist Topic Detection and Tracking corpus) consists of 11201 on-topic documents which are classified into 96 semantic categories. We filter out those documents that appear in two or more categories and randomly sample 100 documents from each of the top 15 categories. Terms with more than 5 occurrence count are retained.

[1] http://mlg.ucd.ie/datasets/bbc.html

[2] http://mlg.ucd.ie/datasets/3sources.html

[3] http://dmml.asu.edu/users/xufei/datasets.html

[4] http://www.cs.umb.edu/~smimarog/textmining/datasets/

[5] http://www.cad.zju.edu.cn/home/dengcai/Data/TextData.html

3.2 Experimental Setting

Definition of Accuracy and NMI Accuracy is defined as follows.

$$Accuracy = \frac{1}{n}\sum_{i=1}^{n}\mathcal{I}(c_i = map(c'_i)) \tag{8}$$

where c'_i is the clustering result of instance i and c_i is its ground truth label. $map(\cdot)$ is a permutation mapping function that maps c'_i to a class label using Kuhn-Munkres Algorithm [1].

Normalized Mutual Information (NMI) is another popular metric for evaluating clustering performance. Let C be the set of clusters from the ground truth and C' obtained from a clustering algorithm. Their mutual information $MI(C, C')$ is defined as follows:

$$MI(C,C') = \sum_{c_i\in C, c'_j\in C'} p(c_i, c'_j)\log\frac{p(c_i, c'_j)}{p(c_i)p(c'_j)} \tag{9}$$

where $p(c_i)$ and $p(c'_j)$ are the probabilities that a random document from the data set belongs to c_i and c'_j, respectively, and $p(c_i, c'_j)$ is the joint probability that the document belongs to the cluster c_i and c'_j at the same time. In our experiments, we use the normalized mutual information.

$$NMI(C,C') = \frac{MI(C,C')}{\max(H(C), H(C'))} \tag{10}$$

where $H(C)$ and $H(C')$ are the entropy of C and C'. Higher value of NMI indicates better quality of clustering.

3.3 Clustering Results

The clustering performance (NMI) on six datasets is shown in Table 2.

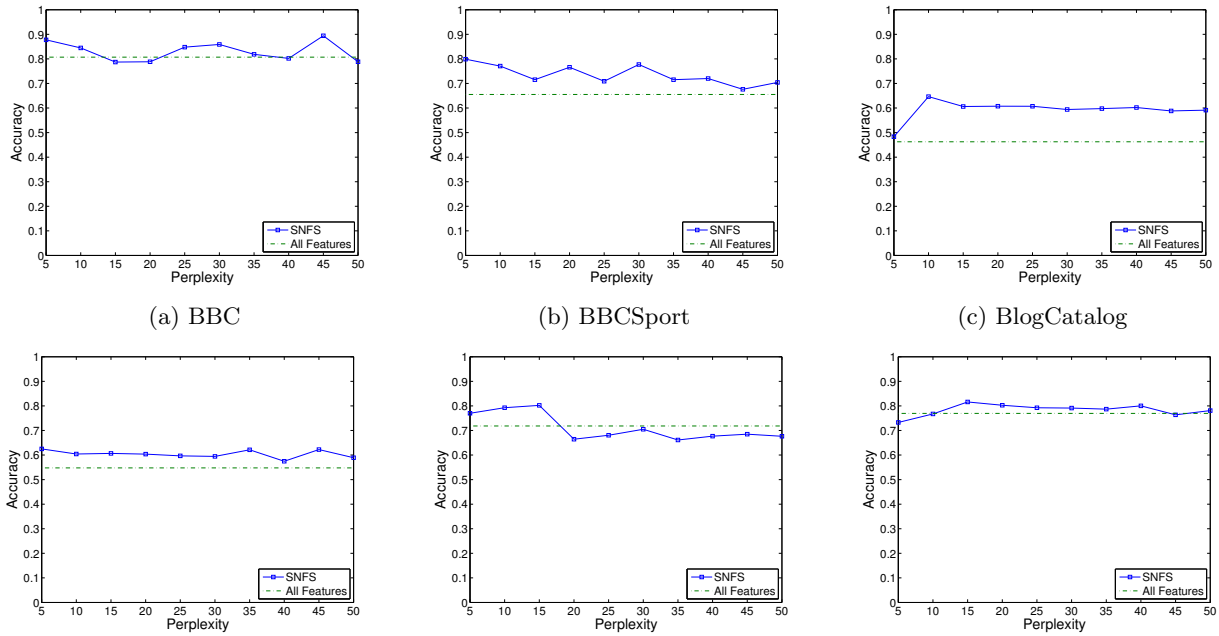
3.4 Sensitivity Analysis

Figure 1 shows the clustering accuracy of SNFS ($N_{0.9}$) over different values of perplexity.

The clustering performance of SNFS ($N_{0.9}$) over different λ is shown in Figure 2.

Table 2: Clustering NMI on six datasets. For UDFS, RUFS, RSFS, median/best performance is reported. SNFS($N_{0.9}$) denotes the performance of SNFS with $N_{0.9}$ features.

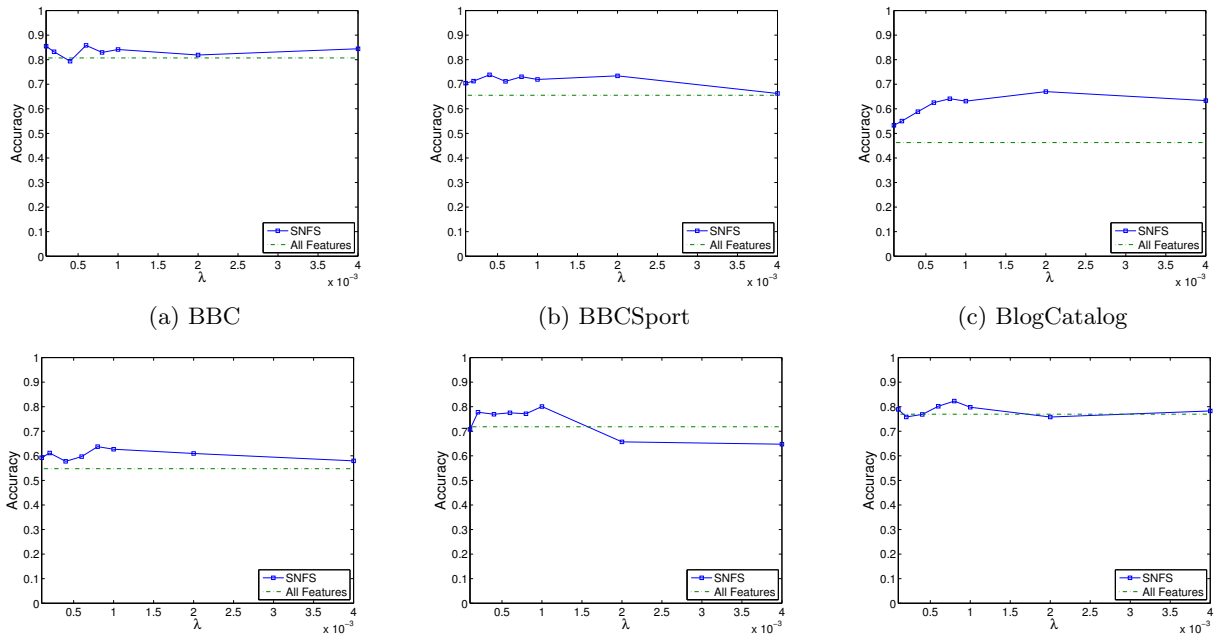
Method	**BBC**				**BBC Sport**			
# features	100	200	400	600	100	200	400	600
All Features	0.7303				0.5190			
LS	0.0119	0.0335	0.2173	0.2649	0.1029	0.1509	0.2745	0.4820
MCFS	0.3820	0.5459	0.6103	0.6665	0.4410	0.5506	0.5685	0.5549
UDFS	0.2167/0.5285	0.4572/0.7001	0.6246/0.7614	0.6715/0.7573	0.2440/0.2741	0.2563/0.3967	0.3767/0.5169	0.3919/0.4706
RUFS	0.2167/0.5285	0.4572/0.7001	0.6246/0.7614	0.6715/0.7573	0.4182/0.6457	0.5384/0.6648	0.5952/0.6383	0.5739/0.6162
RSFS	0.3260/0.5463	0.5669/0.6389	0.6612/0.7393	0.7049/0.7544	0.4705/0.5511	0.529/0.6128	0.5324/0.5701	0.5127/0.5617
SNFS	0.3617	0.5968	0.7027	0.7166	0.3988	0.5046	0.6743	0.5941
SNFS($N_{0.9}$)	0.7358(550)				0.6319(440)			
Method	**BlogCatalog**				**Guardian**			
# features	100	200	400	600	100	200	400	600
All Features	0.2435				0.3711			
LS	0.1040	0.1590	0.1763	0.1796	0.0790	0.2007	0.4967	0.5139
MCFS	0.1189	0.2022	0.1745	0.1635	0.3473	0.3145	0.3751	0.3591
UDFS	0.1295/0.1479	0.1748/0.2363	0.2050/0.2444	0.2696/0.2966	0.1414/0.3109	0.2574/0.3778	0.3358/0.3686	0.3387/0.3759
RUFS	0.2384/0.3110	0.3048/0.3398	0.3303/0.3613	0.3170/0.3504	0.2619/0.3944	0.3767/0.4096	0.3952/0.4295	0.4080/0.4318
RSFS	0.1267/0.2708	0.1884/0.3063	0.2188/0.3337	0.2729/0.3352	0.3649/0.3996	0.3715/0.4307	0.3912/0.4225	0.3971/0.4314
SNFS	0.3230	0.4057	0.3697	0.3604	0.3155	0.4432	0.4996	0.4699
SNFS($N_{0.9}$)	0.4113(230)				0.5144(440)			
Method	**Newsgroup**				**TDT**			
# features	100	200	400	600	100	200	400	600
All Features	0.5031				0.8272			
LS	0.0230	0.0995	0.3865	0.4691	0.6727	0.7613	0.8223	0.8261
MCFS	0.0394	0.1208	0.1640	0.1927	0.6026	0.6659	0.7266	0.7452
UDFS	0.0410/0.0949	0.0730/0.0951	0.1197/0.1272	0.1459/0.3580	0.4182/0.6457	0.5384/0.6648	0.5952/0.6383	0.5739/0.6162
RUFS	0.1635/0.3313	0.1596/0.3701	0.2014/0.3733	0.2431/0.3950	0.3842/0.6647	0.4937/0.8115	0.6373/0.8225	0.7458/0.8473
RSFS	0.1221/0.3060	0.1633/0.3708	0.3046/0.4146	0.3554/0.4449	0.6524/0.7998	0.7750/0.8469	0.8178/0.8578	0.8368/0.8580
SNFS	0.1359	0.1820	0.3993	0.4623	0.7389	0.8266	0.8319	0.8422
SNFS($N_{0.9}$)	0.5815(495)				0.8382(163)			



(a) BBC (b) BBCSport (c) BlogCatalog

(d) Guardian (e) Newsgroup (f) TDT

Figure 1: Clustering accuracy with different perplexity values



(a) BBC (b) BBCSport (c) BlogCatalog

(d) Guardian (e) Newsgroup (f) TDT

Figure 2: Clustering accuracy with different λ

References

[1] J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society of Industrial and Applied Mathematics*, 5(1):32–38, March 1957.