Stability of Feature Selection Algorithms

Alexandros Kalousis,
Jullien Prados,
Phong Nguyen

Melanie Hilario

Artificial Intelligence Group
Department of Computer Science
University of Geneva

Motivation

- Provide a method to study and better understand the behavior of feature selection algorithms.
- Present users with a quantification of the resilience-robustness of the selected features.
- A lot of work for classification algorithms but nothing for feature selection algorithms.

Stability of Classification Algorithms

- Stability has been extensively studied in the context of classification algorithms.
- The main tool has been the *Bias-Variance* decomposition of the error.
- Variance measures sensitivity of a classification algorithm to different training sets.
- It does not measure how different are the models created from different datasets but only how different their predictions are.

Nevertheless:

- BV needs predictions, feature selection algorithms alone do not provide predictions.
- We could couple a feature selection with a classification algorithm and perform BV but then we would be measuring their joint BV profile.

Feature Preferences

From a vector of features, $f = (f_1, f_2, ..., f_m)$, a feature selection algorithm outputs *feature preferences* which can be:

- Weightings-scorings of features, $w = (w_1, w_2, ..., w_m)$
- Rankings of features, $w = (r_1, r_2, ..., r_m)$
- Subsets of selected features, $s = (s_1, s_2, ..., s_m), s_i \in \{0, 1\}$

This is as close as we get to a classification model
but
it does not output predictions
so
if we want to directly quantify the stability of feature selection algorithms we should directly operate in this output.

Stability of feature selection algorithms

- We define the *stability* of a feature selection algorithm as the *sensitivity* of the feature preferences it produces to different training sets drawn from the same generating distribution $P(X, C)$.
- To measure *sensitivity* we need similarity measures between feature preferences. We define one for each type of feature preference:
  - $S_W(w, w')$ based on Pearson's correlation coefficient.
  - $S_R(r, r')$ based on Spearman's correlation coefficient.
  - $S_S(s, s')$ based on Tanimoto's distance between sets.
- The sensitivity of feature selection algorithms that output:
  - Weightings-scorings, can be measured using all three.
  - Rankings, can be measured using S_R and S_S
  - Subsets of features, can be measured using only S_S

Estimating stability

- Draw k different training sets from $P(X, C)$.
- Construct the corresponding k feature preferences.
- Compute the $\frac{k(k-1)}{2}$ pairwise feature preference similarities.
- The average, $\overline{S}$, of feature preference similarities is the estimated value of stability.
- Since we do not have enough training sets we rely on resampling. We use 10-fold cross validation, but any other resampling method would do.

Coupling stability with error estimation

- In practice a feature selection algorithm is applied together with a classification algorithm and we get an error estimation of the coupled application.
- We want to couple this error estimation with a stability estimation in order to select among the most accurate configurations the one that is most stable.
- We use 10-fold cross-validation for error estimation and stability is estimated on each fold by an inner 10-fold cross-validation as described previously.

Experiments

- 11 datasets from proteomics, genomics and text mining.
- We examined five well known feature selection approaches:
  - Three univariate: Information Gain (IG), CHI-Square (CHI), Symmetrical Uncertainty (SYM),
  - Two multivarate:
    - ReliefF (feature weights are determined based on their contribution on the euclidean distance) and
    - SVM-RFE (linear svm with recursive feature elimination)
- We also examined the stability of a simple linear support vector machine (SVMONE) to demonstrate that the notion of stability can be also applied to any classification algorithm that produces a set of feature weights.
- To build the final classification models we used a linear SVM.
- $\overline{S_W}$ and $\overline{S_R}$ were computed on the complete feature preferences.
- $\overline{S_S}$ was computed on the set of the *ten* best features given by each algorithm.

Datasets

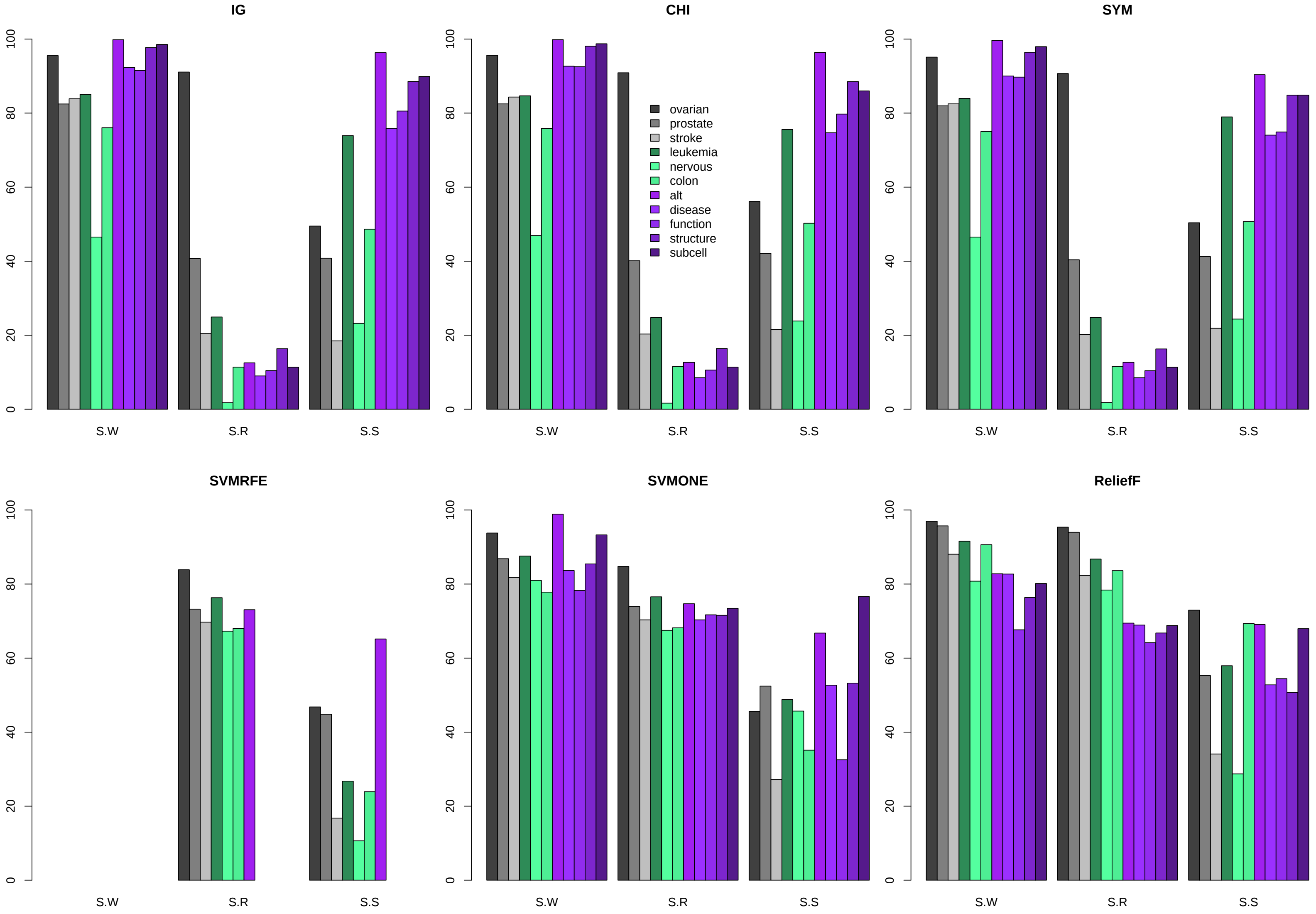
- Proteomics, genomics and text mining datasets.
- Proteomics and genomics datasets are typical for their high dimensionality small sample size.
- The text mining datasets have a high dimensionality but also a high number of training instances.

dataset	class 1	# class 1	class 2	# class 2	# features
ovarian	normal	91	diseased	162	824
prostate	normal	253	diseased	69	2200
stroke	normal	101	diseased	107	4928
leukemia	ALL	47	AML	25	7131
nervous	survival	21	failure	39	7131
colon	normal	22	tumor	40	2000
alt	relevant	1425	not	2732	2112
disease	relevant	631	not	2606	2376
function	relevant	818	not	3089	2708
structure	relevant	927	not	2621	2368
subcell	relevant	1502	not	6475	4031

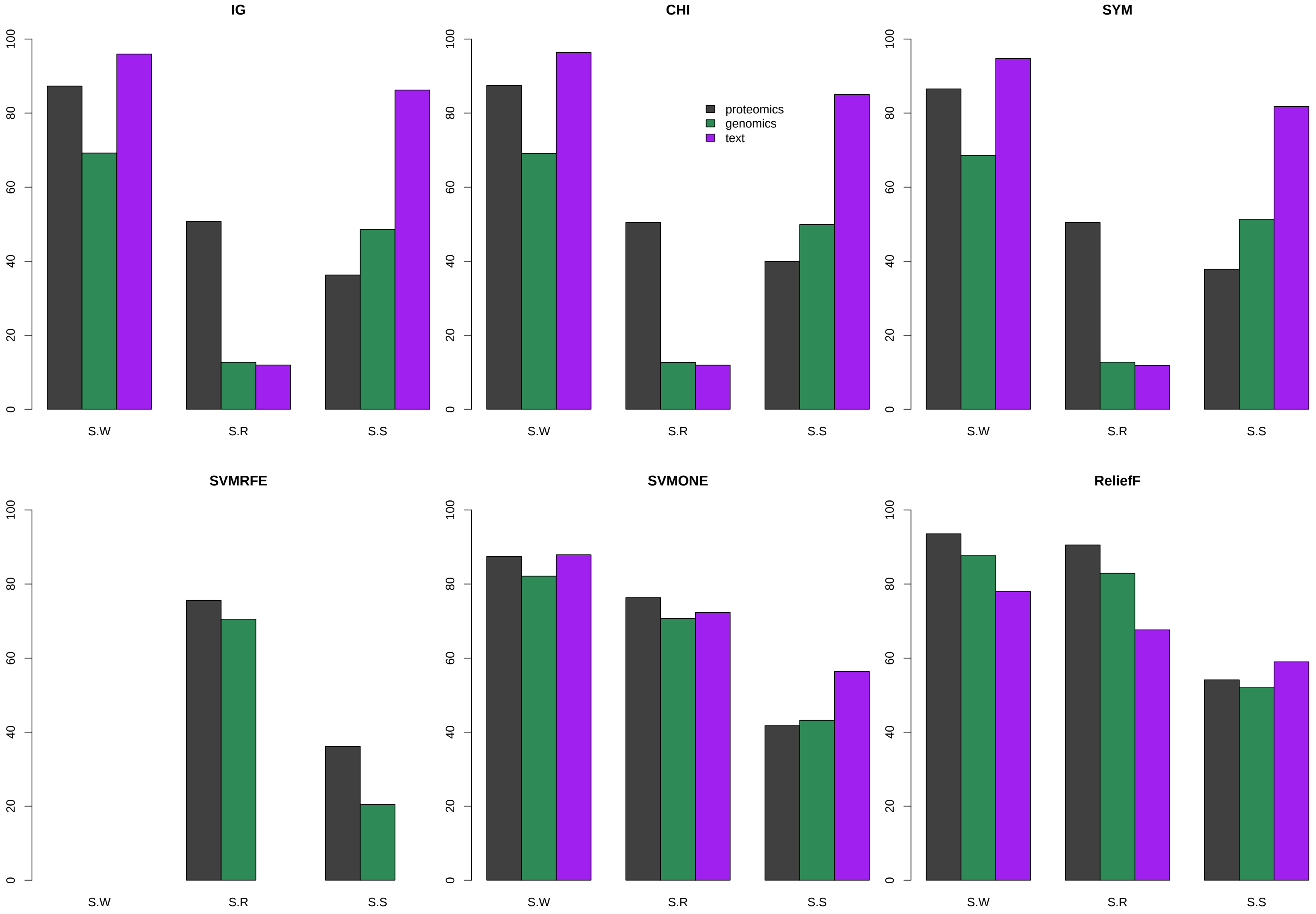
Can we say which stability measure is more appropriate?

- $\overline{S_W}, \overline{S_R}$, are applied on the complete feature preferences; they provide a global view.
- $\overline{S_S}$, is applied on the selected set of k features; it provides a focused view.
- $\overline{S_W}, \overline{S_R}$, treat all weights-ranks equally. However differences or similarities on the highest weighted-ranked features should be emphasized.
- $\overline{S_S}$ focuses on what is most important, i.e. the selected features.
- As a result estimates of stability $\overline{S_W}, \overline{S_R} > \overline{S_S}$ (although it does not make sense to compare their values).
- In practice *Feature subsets* are more interesting then *Rankings* which in turn are more interesting than *Weightings-Scorings*.
- $\overline{S_R}, \overline{S_S}$, can be compared meaningfully among different algorithms, $\overline{S_W}$ cannot due to differences in scales-intervals of weights-scores.
- So it seems that, at least for the moment, the winner is $\overline{S_S}$ (in the next few slides the feature subset size will be fixed at 10).

Stability behavior of FS algorithms



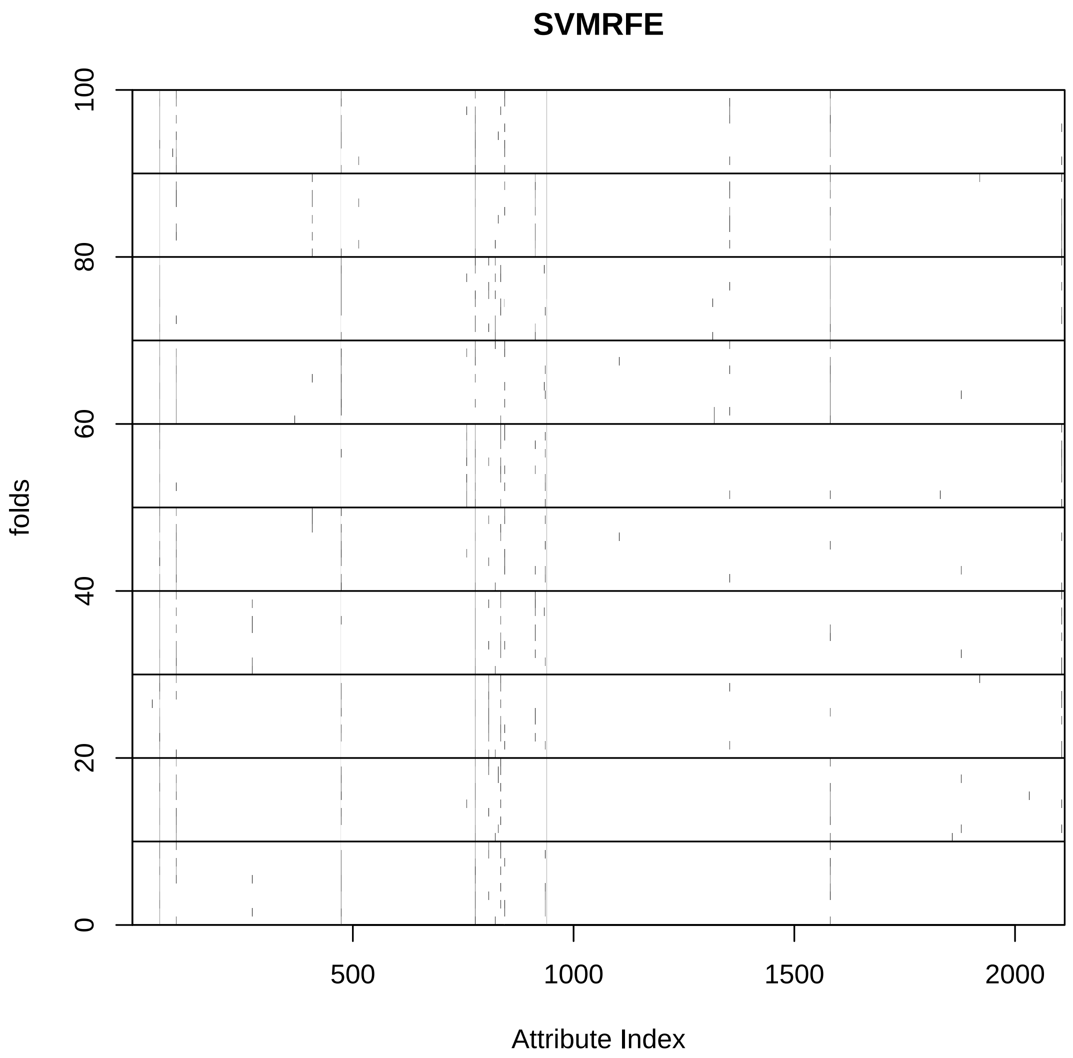
Stability behavior of FS algorithms, averaged per dataset category



Stability performance of feature selection algorithms, Comments

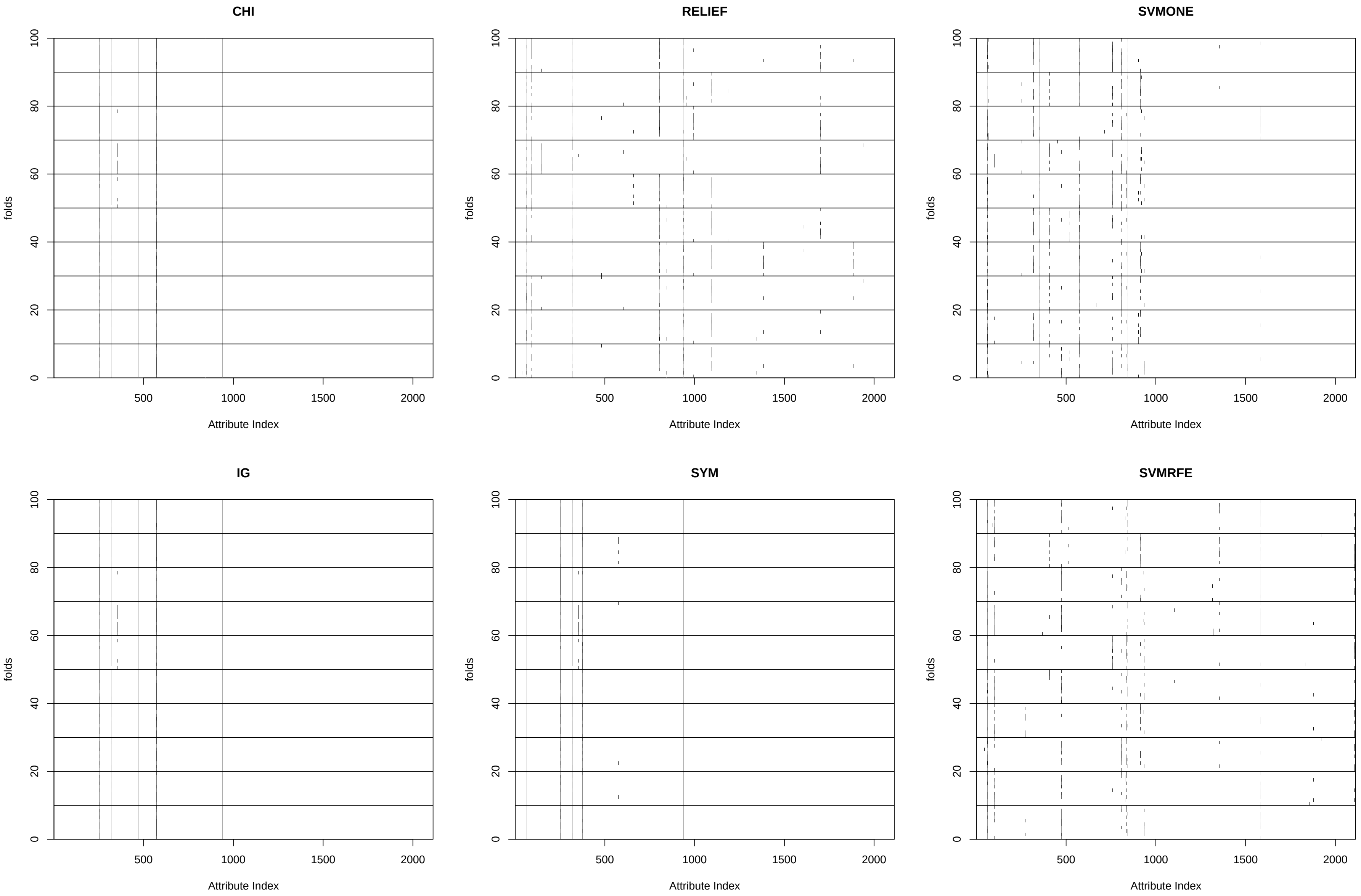
- Stability is not only algorithm dependent but also, and probably mainly, dataset dependent.
- Strong singal provided that the algorithm is able to capture it will give stable models.
- Stability behavior of the three univariate FS algorithms is indistinguishable.
- Poor ranking behavior, $\overline{S_R}$, of univariate algorithms on proteomics and text is explained by the descritization process, which converts many attributes to single value non-informative attributes. Their good ranking behavior on the proteomics datasets should be seen with caution, heavily affected by the ovarian dataset known to have quality problems.
- The univariate FS algorithms produce the most stable feature subsets, $\overline{S_S}$, on the text mining data.
- SVMRFE does not produce weights this is why we have no performance bars on $\overline{S_W}$, moreover it does not complete execution on the text mining datasets due to their "large" size.
- SVMRFE is more unstable with respect to SVMONE on feature subset selection, $\overline{S_S}$, can be explained by the recursive nature of SVMRFE.
- ReliefF seems to produce more stable feature subsets, $\overline{S_S}$, than SVMONE and SVMRFE.

Visualizing stability performance, $\overline{S_S}$, and the feature models

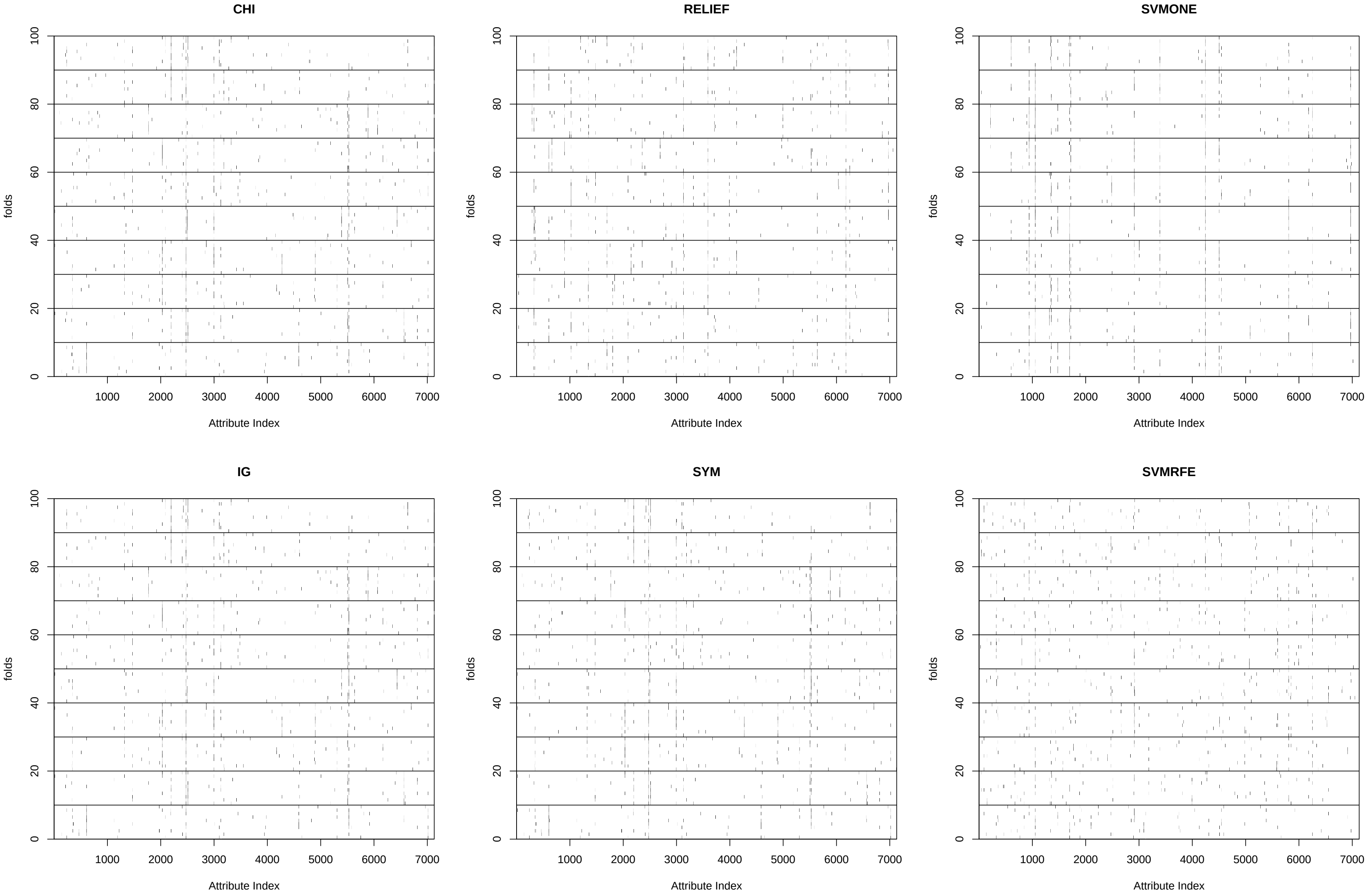


- X-axis=features, Y-axis=folds
- Two consecutive horizontal lines contain the ten inner cv-folds of a single outer cv-fold.
- A point indicates that the corresponding feature was selected.
- The more *complete* vertical lines (i.e. same feature selected among different folds) the more stable the algorithm.
- Clear picture of stability behavior *and also* of which features are important.

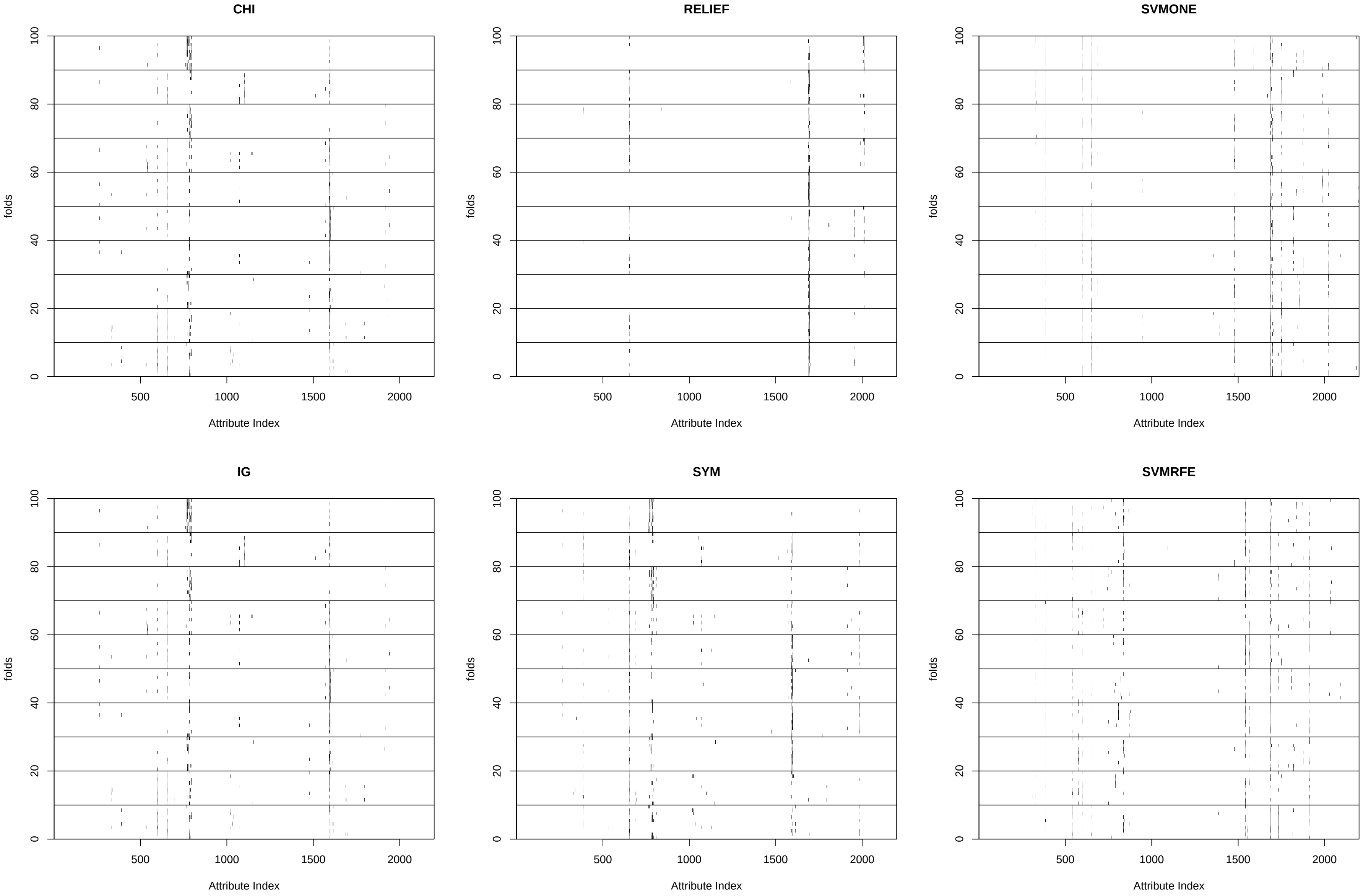
Visualizing stability performance, Alt, text dataset, high stability



Visualizing stability performance, Central Nervous, genomics dataset, low stability



Visualizing stability performance, Prostate, proteomics dataset, average stability



A closer look on feature subset stability, $\overline{S_S}$

- Remember that S_S is given by

$$S_S(s, s') = 1 - \frac{|s| + |s'| - 2|s \cap s'|}{|s| + |s'| - |s \cap s'|}.$$

- Note that as the number of selected features increases

$$|s| \rightarrow |S| \text{ and } |s'| \rightarrow |S|$$

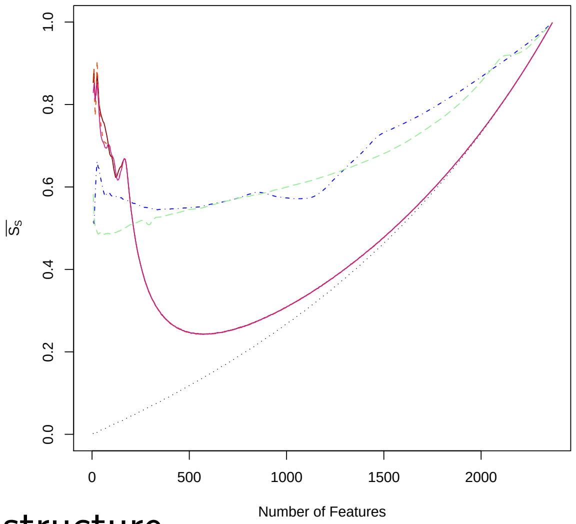
$$\text{then } |s \cap s'| \rightarrow |S| \text{ and } S_S \rightarrow 1$$

  the feature subset stability increases trivially.

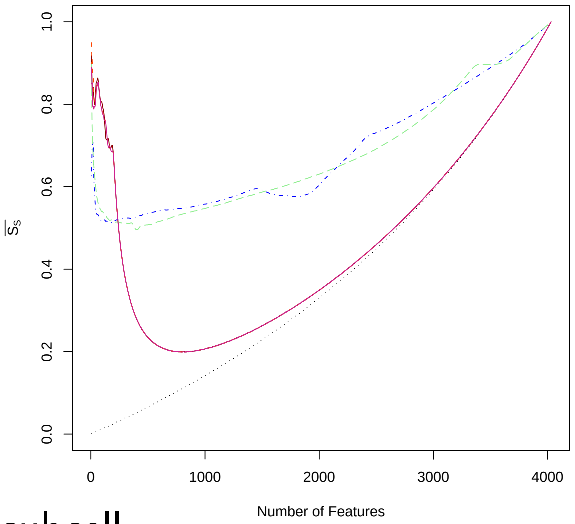
- As a consequence the stability of a random feature selector will increase as the number of selected features approaches $|S|$

- We will create the complete $\overline{S_S}$ stability profile for $|s| = |s'| = 10$ step=5 to=$|S|$ for all feature selection algorithms, including a random one as the baseline.

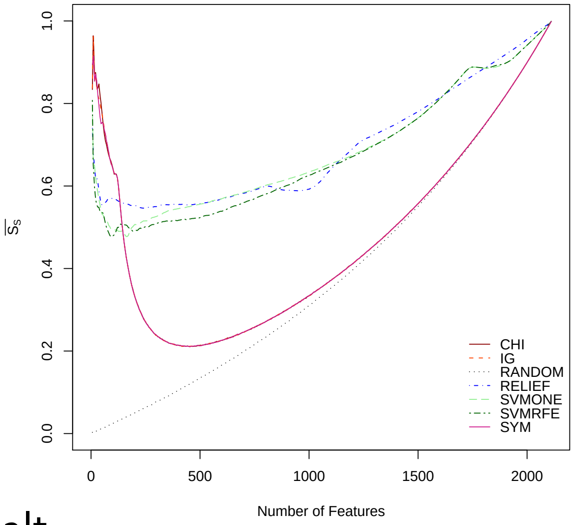
Stability profiles with $\overline{S_S}$, text mining datasets



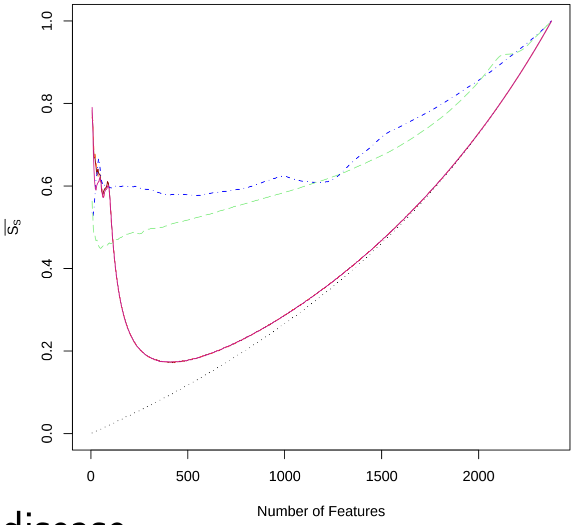
structure



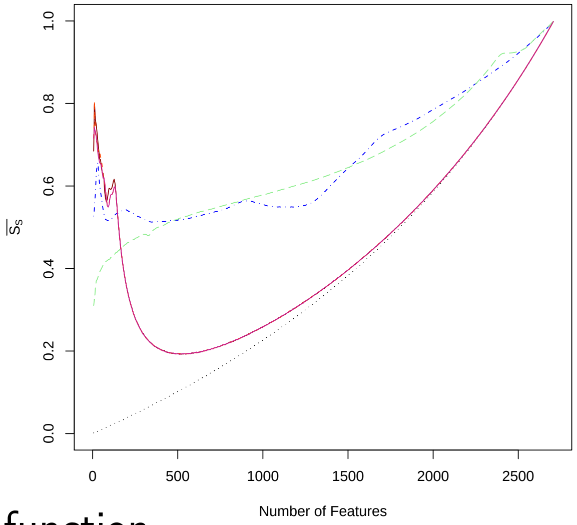
subcell



alt

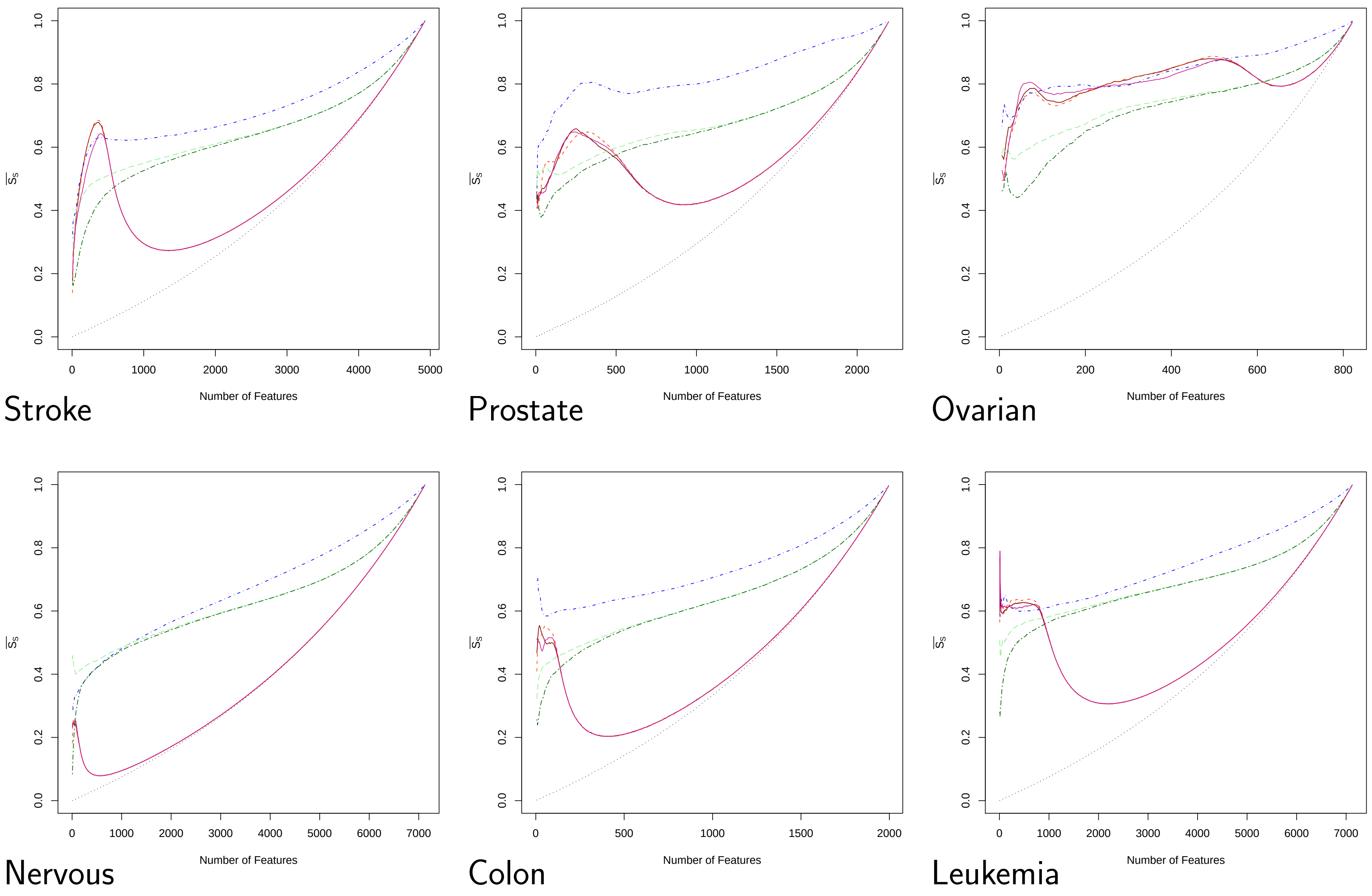


disease



function

Stability profiles with $\overline{S_S}$, genomics and proteomics datasets



Stability profiles with $\overline{S_S}$, comments

- Univariate algorithms have indistinguishable profiles.
- Univariate algorithms exhibit a dramatic drop in stability indicating random addition of features, explained by the discretization process.
- SVMRFE and SVMONE have very similar profiles and diverge on low feature set cardinalities, reasonable if one considers the similarities of the two methods.
- ReliefF has one of the most stable behaviors for large range of feature subset cardinalities.
- All algorithms reach a "knot" point after which stability grows slowly only because of the increase in $|s|$ the number of selected features.
- The knot probably indicates inclusion of the most robust-stable features. After that features are included randomly. Could provide the basis for determining the optimal $|s|$.

Stability and Classification performance

- Stability alone is not enough
- In real world applications selection of algorithms is guided by classification performance
- We propose the following selection strategy:
  - Couple the feature selection algorithms with a classification algorithm
  - Estimate stability and classification performance
  - Among the combination that achieve top classification performance and which performances is not significantly different choose the one with the bigger stability.

Stability and Classification performance (Example)

		Stroke		
N	IG	Relief	SVM	SVMRFE
10	1.5-32.22-0.1847	1.5-30.29-0.3410	1.0-37.02-0.2721	2.0-26.45-0.1678
20	1.0-31.73-0.2612	1.0-28.85-0.3670	1.0-35.10-0.3101	3.0-21.64-0.1679
30	1.5-27.89-0.2944	1.5-27.41-0.3830	1.5-28.37-0.3390	1.5-23.56-0.1802
40	1.5-29.81-0.3261	1.5-25.97-0.3887	1.5-25.00-0.3583	1.5-25.49-0.1886
50	1.5-27.89-0.3576	1.5-28.37-0.4013	1.5-26.45-0.3801	1.5-25.49-0.1997
		Ovarian		
N	IG	Relief	SVM	SVMRFE
10	1.0-10.28-0.4948	1.0-10.28-0.7296	1.0-07.11-0.5965	3.0-01.19-0.4680
20	1.0-05.53-0.6111	1.0-05.93-0.6933	1.5-03.95-0.5897	2.5-01.19-0.4749
30	0.0-04.74-0.6567	2.0-01.58-0.6966	2.0-01.19-0.5631	2.0-00.40-0.4498
40	0.5-03.16-0.7011	1.5-01.58-0.7080	2.0-00.40-0.5682	2.0-00.40-0.4401
50	1.5-02.77-0.7496	1.5-01.58-0.7368	1.5-00.40-0.5825	1.5-00.40-0.4473
		Prostate		
N	IG	Relief	SVM	SVMRFE
10	1.0-18.64-0.4073	1.0-18.95-0.5842	1.0-18.02-0.5308	3.0-13.05-0.4417
20	1.0-17.71-0.4299	1.0-17.09-0.6044	1.0-16.46-0.5131	3.0-11.50-0.4006
30	1.0-16.46-0.4639	1.0-15.84-0.6170	1.0-14.91-0.5193	3.0-10.87-0.3786
40	1.0-16.15-0.5044	1.0-14.91-0.6214	1.0-13.36-0.5280	3.0-09.01-0.3848
50	1.0-14.60-0.5374	1.0-13.36-0.6304	1.0-13.05-0.5343	3.0-09.32-0.3890

- Classification algorithm linear SVM.
- Cells contain:
  - Class. Perf. Ranking
  - Error
  - Stability ($\overline{S_S}$).
- Class. Perf. Ranking is the sum of significant wins plus 0.5 for each tie.
- In red all cases with top and not significantly different classification performance.

Stability and Classification performance (Example)

		Stroke		
N	IG	Relief	SVM	SVMRFE
10	1.5-32.22-0.1847	1.5-30.29-0.3410	1.0-37.02-0.2721	2.0-26.45-0.1678
20	1.0-31.73-0.2612	1.0-28.85-0.3670	1.0-35.10-0.3101	3.0-21.64-0.1679
30	1.5-27.89-0.2944	*1.5-27.41-0.3830*	1.5-28.37-0.3390	1.5-23.56-0.1802
40	1.5-29.81-0.3261	*1.5-25.97-0.3887*	1.5-25.00-0.3583	1.5-25.49-0.1886
50	1.5-27.89-0.3576	*1.5-28.37-0.4013*	1.5-26.45-0.3801	1.5-25.49-0.1997
		Ovarian		
N	IG	Relief	SVM	SVMRFE
10	1.0-10.28-0.4948	1.0-10.28-0.7296	1.0-07.11-0.5965	3.0-01.19-0.4680
20	1.0-05.53-0.6111	1.0-05.93-0.6933	1.5-03.95-0.5897	2.5-01.19-0.4749
30	0.0-04.74-0.6567	*2.0-01.58-0.6966*	2.0-01.19-0.5631	2.0-00.40-0.4498
40	0.5-03.16-0.7011	1.5-01.58-0.7080	*2.0-00.40-0.5682*	2.0-00.40-0.4401
50	*1.5-02.77-0.7496*	1.5-01.58-0.7368	1.5-00.40-0.5825	1.5-00.40-0.4473
		Prostate		
N	IG	Relief	SVM	SVMRFE
10	1.0-18.64-0.4073	1.0-18.95-0.5842	1.0-18.02-0.5308	3.0-13.05-0.4417
20	1.0-17.71-0.4299	1.0-17.09-0.6044	1.0-16.46-0.5131	3.0-11.50-0.4006
30	1.0-16.46-0.4639	1.0-15.84-0.6170	1.0-14.91-0.5193	3.0-10.87-0.3786
40	1.0-16.15-0.5044	1.0-14.91-0.6214	1.0-13.36-0.5280	3.0-09.01-0.3848
50	1.0-14.60-0.5374	1.0-13.36-0.6304	1.0-13.05-0.5343	3.0-09.32-0.3890

- Classification algorithm linear SVM.
- Cells contain:
  - Class. Perf. Ranking
  - Error
  - Stability $(\overline{S_S})$.
- Class. Perf. Ranking is the sum of significant wins plus 0.5 for each tie.
- In red all cases with top and not significantly different classification performance.
- So among the top classification performers choose the *most stable*

Stability, Classification Performance and Redundancy

		Stroke		
N	IG	Relief	SVM	SVMRFE
10	1.5-32.22-0.1847	1.5-30.29-0.3410	1.0-37.02-0.2721	2.0-26.45-0.1678
20	1.0-31.73-0.2612	1.0-28.85-0.3670	1.0-35.10-0.3101	3.0-21.64-0.1679
30	1.5-27.89-0.2944	*1.5-27.41-0.3830*	1.5-28.37-0.3390	1.5-23.56-0.1802
40	1.5-29.81-0.3261	*1.5-25.97-0.3887*	1.5-25.00-0.3583	1.5-25.49-0.1886
50	1.5-27.89-0.3576	*1.5-28.37-0.4013*	1.5-26.45-0.3801	1.5-25.49-0.1997

		Ovarian		
N	IG	Relief	SVM	SVMRFE
10	1.0-10.28-0.4948	1.0-10.28-0.7296	1.0-07.11-0.5965	3.0-01.19-0.4680
20	1.0-05.53-0.6111	1.0-05.93-0.6933	1.5-03.95-0.5897	2.5-01.19-0.4749
30	0.0-04.74-0.6567	*2.0-01.58-0.6966*	2.0-01.19-0.5631	2.0-00.40-0.4498
40	0.5-03.16-0.7011	1.5-01.58-0.7080	*2.0-00.40-0.5682*	2.0-00.40-0.4401
50	*1.5-02.77-0.7496*	1.5-01.58-0.7368	1.5-00.40-0.5825	1.5-00.40-0.4473

		Prostate		
N	IG	Relief	SVM	SVMRFE
10	1.0-18.64-0.4073	1.0-18.95-0.5842	1.0-18.02-0.5308	3.0-13.05-0.4417
20	1.0-17.71-0.4299	1.0-17.09-0.6044	1.0-16.46-0.5131	3.0-11.50-0.4006
30	1.0-16.46-0.4639	1.0-15.84-0.6170	1.0-14.91-0.5193	3.0-10.87-0.3786
40	1.0-16.15-0.5044	1.0-14.91-0.6214	1.0-13.36-0.5280	3.0-09.01-0.3848
50	1.0-14.60-0.5374	1.0-13.36-0.6304	1.0-13.05-0.5343	3.0-09.32-0.3890

- low or lower stability is not related to low classification performance.
- High classification performance does not imply high stability.
- High classification performance + low stability $\longrightarrow$ feature redundancy

Summary

- Introduce the concept of stability of feature selection algorithms.
- Examine the stability profiles of different feature selection algorithms.
- Provide an eloquent visualization of stability performance and feature resilience.
- Couple classification performance based algorithm selection with stability based algorithm selection.

Future Work back then...

- Examine how we can use stability to choose the optimal number of features.
- Examine how we can use stability to detect feature redundancies.
- Combine feature sets to increase stability (the analogue of ensemble methods in classification algorithms).

Related Work

- Different stability measures have been proposed
- Efforts to increase feature selection stability

Sources of Feature Selection Instability

- Small datasets which result in statistics that are sensitive to data perturbations
- Low signal in the features
- Redundant feature sets that can lead to different models of equal predictive power
- The algorithms themselves:
  - algorithms that depend on initial conditions,
  - multiple local optima
  - more aggressive feature selection algorithms result in more unstable models, e.g. l_1 norm regularization is more unstable compared to l_2. With l_1 out of a set of relevant but redundant features only one will be selected the algorithm does not care which one that will be, in fact in the case of redundant and relevant features there is not a unique solution; with l_2 all of them will be but with lower weights.

Increasing Feature Selection Stability

Different methods have been proposed to increase stability

- Managing feature redundancy in a principled manner:
  - Feature clustering methods, (Yu et al., 2008), which cluster features to groups of similar-redundant features and then apply feature selection in the clustering results.
  - Elastic net, (Zou & Hastie, 2005), which combines l_1 and l_2 regularization that has as result to select all together a group of redundant features or none of them in what is known as the *grouping effect*. Strict convexity has the grouping effect.
- Feature model agregation, e.g. ensemble methods, (Sayes et al., 2008; Loscalzo et al., 2009), which agregate feature models obtained from different subsamples of the training set to obtain a single stable feature model.

Increasing stability especially if this is done outside the FS algorithm using, e.g. ensemble methods, does not mean we can expect accuracy gains.

Our current work

- Define more appropriate stability measures
- Explore the effect of sample size in stability
- Place the concept of feature stability in a broader concept of model similarity

References

Kalousis, A., Prados, J., & Hilario, M. (2007). Stability of feature selection algorithms: a study on high dimensional spaces. *Knowledge and Information Systems*, *12*, 95–116.

Loscalzo, S., Yu, L., & Ding, C. (2009). Consensus group based stable feature selection. *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD-2009*.

Sayes, Y., Abeel, T., & Peer, Y. (2008). Robust feature selection using ensemble feature selection techniques. *Proceedings of the European conference on Machine Learning and Knowledge Discovery in Databases*.

Yu, L., Ding, C., & Loscalzo, S. (2008). Stable feature selection via dense feature groups. *Proceedings of the KDD 2008*.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society B*, *67*.