

PROCEEDINGS

Open Access

Scoring relevancy of features based on combinatorial analysis of Lasso with application to lymphoma diagnosis

Habil Zare^{1,2*}, Gholamreza Haffari³, Arvind Gupta², Ryan R Brinkman⁴

From The Eleventh Asia Pacific Bioinformatics Conference (APBC 2013)
Vancouver, Canada. 21-24 January 2013

Abstract

One challenge in applying bioinformatic tools to clinical or biological data is high number of features that might be provided to the learning algorithm without any prior knowledge on which ones should be used. In such applications, the number of features can drastically exceed the number of training instances which is often limited by the number of available samples for the study. The Lasso is one of many regularization methods that have been developed to prevent overfitting and improve prediction performance in high-dimensional settings. In this paper, we propose a novel algorithm for feature selection based on the Lasso and our hypothesis is that defining a scoring scheme that measures the “quality” of each feature can provide a more robust feature selection method. Our approach is to generate several samples from the training data by bootstrapping, determine the best relevance-ordering of the features for each sample, and finally combine these relevance-orderings to select highly relevant features. In addition to the theoretical analysis of our feature scoring scheme, we provided empirical evaluations on six real datasets from different fields to confirm the superiority of our method in exploratory data analysis and prediction performance. For example, we applied FeaLect, our feature scoring algorithm, to a lymphoma dataset, and according to a human expert, our method led to selecting more meaningful features than those commonly used in the clinics. This case study built a basis for discovering interesting new criteria for lymphoma diagnosis. Furthermore, to facilitate the use of our algorithm in other applications, the source code that implements our algorithm was released as FeaLect, a documented R package in CRAN.

Introduction

To build a robust classifier, the number of training instances is usually required to be more than the number of features. In many real life applications such as bioinformatics, natural language processing, and computer vision, a high number of features might be provided to the learning algorithm without any prior knowledge about which ones should be used. Therefore, the number of features can drastically exceed the number of training instances and the model is subject to overfit the training data. Many regularization methods have been developed to prevent

overfitting and to improve the generalization error bound of the predictor in this learning situation.

Most notably, Lasso [1] is an ℓ_1 -regularization technique for linear regression which has attracted much attention in machine learning and statistics. The same approach is useful in classification because any binary classification problem can be reduced to a regression problem by treating the class labels as real numbers, and consider the sign of the model prediction as the class label. The features selected by the Lasso depends on the regularization parameter, and the set of solutions for all values of this free parameter is provided by regularization path [2]. Although efficient algorithms exist for recovering the whole regularization path for the Lasso [3], finding a subset of highly *relevant* features which leads to a robust predictor is a prominent research question.

* Correspondence: zare@u.washington.edu

¹Department of Genome Sciences, University of Washington, Seattle, Washington, USA

Full list of author information is available at the end of the article

In this paper, we propose a novel algorithm for feature selection based on the Lasso and our hypothesis is that defining a scoring scheme that measures the “quality” of each feature can provide a more robust feature selection method. Our approach is to generate several samples from the training data by bootstrapping, determine the best relevance-ordering of the features for each sample, and finally combine these relevance-orderings to select highly relevant features. In addition to the theoretical analysis of our feature scoring scheme, we provided empirical evaluations using a real-life lymphoma dataset as well as several UCI datasets, which confirms the superiority of our method in exploratory data analysis and prediction performance.

Background and previous work

Lasso is an ℓ_1 -regularization technique for least-square linear regression:

$$\mathcal{L} := \sum_{i=1}^n \frac{1}{2n} \|y_i - w^T \cdot x_i\|_2^2 + \lambda \|w\|_1 \quad (1)$$

where the response random variable $Y \in \mathbb{R}$ is dependent on a d -dimensional covariate $X \in \mathbb{R}^d$, and the training data $D = \{(x_i, y_i)\}_{i=1}^n$ is independently and identically sampled from a fixed joint distribution P_{XY} . It is well known that the ℓ_1 -regularization term shrinks many components of the solution to zero, and thus performs feature selection [4]. There has been also some variants, such as elastic nets [5], to select highly-correlated predictive features. The number of selected features in eqn (1) is controlled by the regularization parameter λ .

A common practice is to find the best value for λ by cross-validation to maximize the prediction accuracy. Having found the best value for the regularization parameter, the features are selected based on the non-zero components of the global and unique minimizer of the training objective in equation (1). However, recent research on the *consistency* of the Lasso [4,6-10] shows that a fixed value of λ for all n will not result in a *consistent* estimate for the parameter vector [7]. Now, the question is what would be a proper value for λ as a function of n with a theoretical basis?

Various decaying schemes of the regularization parameter were studied [4,7,8,11] and it is shown that under specific settings, Lasso selects the *relevant* features with probability one and the *irrelevant* features with a positive probability less than one, provided that the number of training instances tends to infinity. To do a better feature selection, note that each run of the cross-validation gives the value of the regularization parameter λ and the corresponding selected-features. If several samples were available from the underlying data distribution, irrelevant features could be removed by simply *intersecting* the set of

selected features for each sample. The idea in [7] is to provide such datasets by resampling with replacement from the given training dataset using the *bootstrap* method [12]. This approach leads to Bolasso algorithm for feature selection that is theoretically motivated by the proposition 1.

Proposition 1. [7] *Suppose P_{XY} satisfies some mild assumptions and let $\lambda = \mu_0 n^{-\frac{1}{2}}$ for a fixed constant $\mu_0 > 0$. Let \mathbf{J} represents the index of the **true** relevant features, and $\hat{\mathbf{j}}$ denote the index of relevant features found by Bolasso. Then, the probability that Bolasso does not select the correct model is upper-bounded by:*

$$\Pr(\hat{\mathbf{j}} \neq \mathbf{J}) \leq mA_1 e^{-A_2 n} + A_3 \frac{\log n}{n^{\frac{1}{2}}} + A_4 \frac{\log m}{m},$$

where $m > 1$ is the number of bootstrap samples, and all A_i s are positive constants.

Now, if we send m to infinity slower than $e^{A_2 n}$, then with probability tending to one Bolasso will select \mathbf{J} , exactly the relevant features. The proposition 1 guarantees the performance of Bolasso only asymptotically, i.e. when $n \rightarrow \infty$. However, in real applications where the number of training samples is often limited, the probability of selecting relevant features can be significantly less than 1. One of the main goals of our proposed framework in this paper is to address this problem by scoring the features.

Previous studies have shown that there is room for improving Bolasso [7,8]. For example, while on synthetic data it outperforms similar methods such as ridge regression, Lasso, and bagging of Lasso estimates [13], Bolasso is sometimes too strict on real data because it requires the relevant features to be selected in all bootstrap runs. Bolasso-S, a soft version of Bolasso, performs better in practice because it relaxes this condition and selects a feature if it is chosen in at least a user-defined fraction of the bootstrap replicates (a threshold of 90% is considered to be enough). Bolasso-S is more flexible and thus, more appropriate for the practical models that are not extremely sparse [8].

Our contributions

In this paper, we develop FeaLect algorithm that is softer than Bolasso in the following three directions:

- For each bootstrap sample, Bolasso considers only one model that minimizes the training objective \mathcal{L} in eqn (1), whereas we include information provided by the whole regularization path,
- Instead of making a binary decision of inclusion or exclusion, we compute a score value for each feature that can help the user to select the more relevant ones,
- While Bolasso-S relies on a threshold, our theoretical study of the behaviour of irrelevant features leads

to an analytical criterion for feature selection without using any pre-defined parameter.

We compared the performance of Bolasso, FeaLect, and Lars algorithms for feature selection on six real datasets in a systematic manner. The source code that implements our algorithm was released as FeaLect, a documented R package in CRAN.

Feature scoring and mathematical analysis

In this section, we describe our novel algorithm that scores the features based on their performance on samples obtained by bootstrapping. Afterwards, we present the mathematical analysis of our algorithm which builds the theoretical basis for its proposed automatic thresholding in feature selection.

The FeaLect algorithm

Our feature selection algorithm is outlined in Figure 1 and described in Algorithm 1. Let B be a random sample with size γm generated by choosing from the given training data D without replacement, where $n = |D|$ and $\gamma \in (0, 1)$ is a parameter that controls the size of sample sets. Using a training set B , we apply the Lars algorithm to recover the whole regularization path efficiently [3]. Let F_k^B be the set of selected features by the Lasso when λ allows exactly k features to be selected. The number of selected features is decreasing in λ and we have:

$$\emptyset = F_0^B \subset \dots \subset F_k^B \subset F_{k+1}^B \subset \dots \subset F_d^B = F.$$

For each feature f , we define a scoring scheme depending on whether or not it is selected in F_k^B :

$$S_k^B(f) := \begin{cases} \frac{1}{k} & \text{if } f \in F_k^B \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

The above randomized procedure is repeated several times for various random subsets B to compute the average score of f when exactly k features are selected, i.e. $\mathbb{E}_B[S_k^B(f)]$ is estimated empirically. According to our experiments, the convergence rate to the expected score is fast and there is no significant difference between the average scores computed by 100 or 1000 samples (Figure 2). The total score for each feature is then defined as the sum of average scores:

$$S(f) := \sum_k \mathbb{E}_B[S_k^B(f)] \quad (3)$$

Algorithm 1 Feature Scoring

- 1: **for** $t = 1$ to m **do**
- 2: Sample (without replacement) a random subset $B \subset D$ with size $\gamma|D|$
- 3: Run Lars on B to obtain F_1^B, \dots, F_d^B
- 4: Compute S_1^B, \dots, S_d^B using eqn (2)
- 5: **for** $k \in \{1, \dots, d\}$ **do**
- 6: Update the feature scores for all feature $f : S(f) \leftarrow S(f) + S_k^B(f)/m$
- 7: **end for**
- 8: **end for**

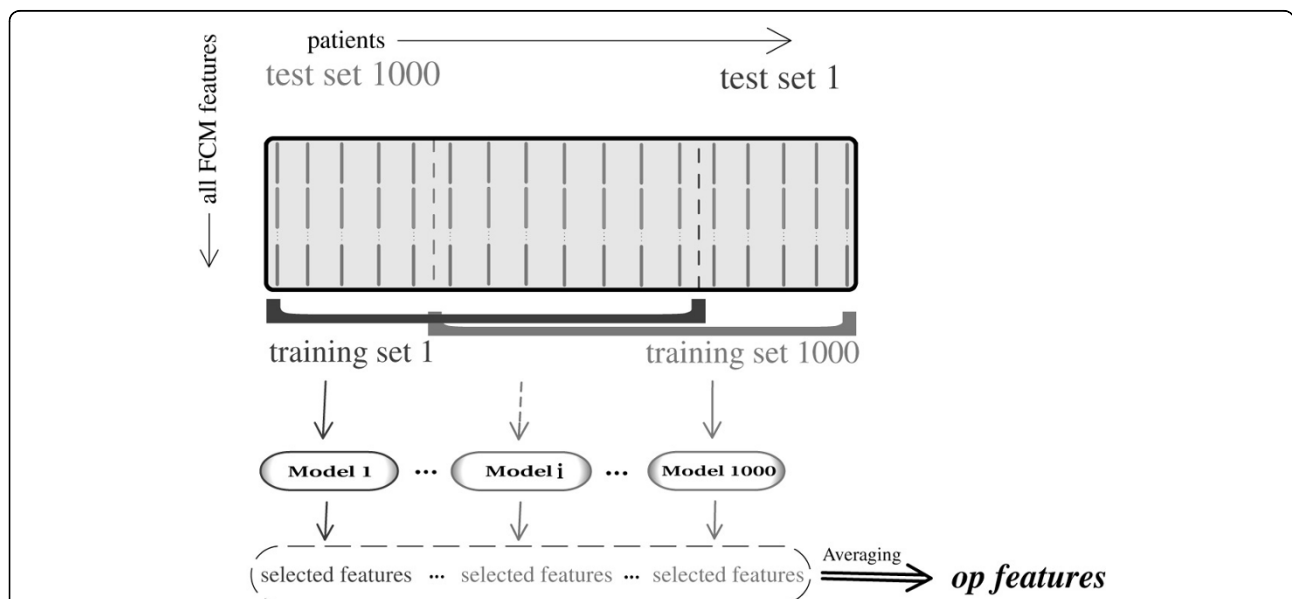
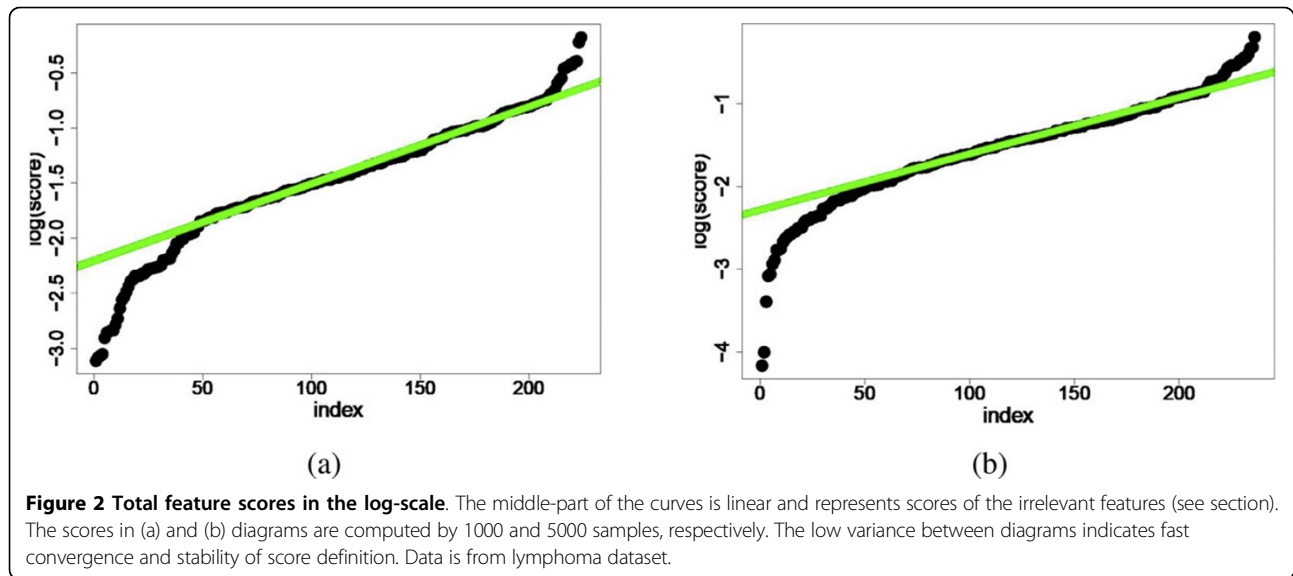


Figure 1 Overview of bootstrapping performed by FeaLect. A row and a column of the gray data matrix correspond to a feature and a case, accordingly. 1000 models are trained, each fitted to a random subset that contains $\frac{3}{4}$ of cases using Lasso technique [1]. Without any assumption from a-priori knowledge, all features are included for training the models. Then the selected features are scored by computing an average vote (eq. 3) to select the most predictive ones.



9: Fit a 3-segment spline ($g_1(\cdot)$, $g_2(\cdot)$, $g_3(\cdot)$) on log-scale feature score curve (see the text for more information)

10: return features corresponding to g_3 as informative features

Before describing the rest of the algorithm, let us have a look at the feature scores for our lymphoma classification problem (The task and data set is described in details in the Experiment section). Figure 2 depict the total score of features in log-scale, where features are sorted according to their increasing total scores. The feature score curve is almost linear in the middle and bending at both ends. We hypothesize that features with a “very high score” in the top non-linear and bending part of the curve are good candidates for informative features. Furthermore, the linear middle-part of the curve consists of features that are responsible for the model to get overfitted and therefore we call them *irrelevant* features. In the next section, a formal definition will be provided to clarify this intuitive idea and we show how this insight can be very helpful in identifying informative features.

The final step of our feature selection algorithm is to fit a 3-segment spline model to the feature score curve: the first quadratic lower-part captures the low score features, the linear middle-part captures irrelevant features, and the last quadratic upper-part captures high-score informative features. As discussed below, the middle linear-part provides an analytic threshold for the score of relevant features: The features with score above this threshold are reported as informative features which can be used for training the final predictor and/or explanatory data analysis.

The analysis

The aim of this analysis is to provide a mathematical explanation for the linearity of the middle part of the scoring function (Figure 2), and also a justification for why the features corresponding to this part can be excluded. We first present a probabilistic interpretation of the feature scores, and then we provide a precise definition of an irrelevant feature. Our definition formalizes the fact that such a feature is selected by the Lasso if and only if a particular fixed finite subset U of instances is included in the random training set, whereas a relevant feature should be selected for almost any general U . We estimate the probability that a random sample $B \subset D$ contains U as n grows to infinity. Finally, we show that asymptotically, the log of the scores for irrelevant features is linear in $|U|$. This explains the linearity of the middle part of the feature score curve in Figure 2.

Proposition 2. Suppose $\Pr(\mathbf{f} = f_i)$ is the probability of selecting a feature f_i by the Lasso in some stage of our feature selection method in Algorithm 1. Then, the probability distribution of the random variable \mathbf{f} is given by:

$$\Pr(\mathbf{f} = f_i) = \frac{1}{d} S(f_i)$$

Proof. By conditional probability:

$$\begin{aligned} \Pr(\mathbf{f} = f_i) &= \sum_B \Pr(\mathbf{f} = f_i | B) \Pr(B) \\ &= \sum_B \sum_{k=1}^d (\Pr(\mathbf{f} = f_i | \mathbf{f} \in F_k^B) \Pr(\mathbf{f} \in F_k^B) \Pr(B)) \\ &= \sum_B \sum_{k=1}^d S_k^B(f_i) \Pr(\mathbf{f} \in F_k^B) \Pr(B) \end{aligned}$$

Since we have not imposed any prior assumption, we put a uniform distribution on $\Pr(\mathbf{f} \in F_k^B)$ to get:

$$\begin{aligned} \Pr(\mathbf{f} = f_i) &= \frac{1}{d} \sum_B \sum_k S_k^B(f_i) \Pr(B) \\ &= \frac{1}{d} \mathbb{E}_B \left(\sum_k S_k^B(f_i) \right) \\ &= \frac{1}{d} S(f_i). \end{aligned}$$

□

The following definition formalizes the idea that irrelevant features depend only on a specific subset of the whole data set.

Definition 3. For any subset of samples $U \subseteq A$ and any feature f_i , we say that f_i over-fits on U if:

$$\forall k, \forall B : f_i \in F_k^B \Leftrightarrow U \subseteq B$$

In words, f_i is selected in F_k^B if and only if B contains U . Next, we derive the probability of including a specific set U in a randomly generated sample.

Lemma 4. For any $U \subseteq A$, we have:

$$\lim_{n \rightarrow \infty} \Pr_B(U \subseteq B) = \gamma^r$$

where r is the number of samples in U and γ is the fraction of samples chosen for a random set B .

Proof. Assuming B has γn members chosen without replacement, we have:

$$\begin{aligned} \Pr_B(U \subseteq B) &= \frac{\binom{n-r}{\gamma n}}{\binom{n}{\gamma n}} \\ &= \frac{(n-r)! (\gamma n)!}{n! (\gamma n - r)!} \\ &= \left(\prod_{i=1}^{n-r} i \right) \left(\prod_{i=1}^{\gamma n} i \right) \left(\prod_{i=1}^{n-r} i^{-1} \right) \left(\prod_{i=1}^{\gamma n-r} i^{-1} \right) \\ &= \left(\prod_{i=1}^{n-r} i \right) \cdot \left(\prod_{i=1}^{\gamma n-r} i \right) \cdot \left(\prod_{i=\gamma n-r+1}^{\gamma n} i \right)^{-1} \times \\ &\quad \left(\prod_{i=1}^{n-r} i^{-1} \right) \cdot \left(\prod_{i=n-r+1}^n i^{-1} \right) \cdot \left(\prod_{i=1}^{\gamma n-r} i^{-1} \right) \\ &= \left(\prod_{i=\gamma n-r+1}^{\gamma n} i \right) \cdot \left(\prod_{i=n-r+1}^n i^{-1} \right) \\ &= \prod_{i=0}^{r-1} [(\gamma n - i)(n - i)^{-1}] \\ &= \gamma^r \prod_{i=0}^{r-1} \left(\frac{n-i}{n-i} \right) \\ &= \gamma^r \prod_{i=0}^{r-1} \left(1 + \frac{i(1-\gamma)}{n-i} \right) \\ &= \gamma^r (1 + O(n^{-1})). \end{aligned}$$

The first line of the above proof relies on the assumption that the members of the random set B are chosen

without replacement, and the claim derives from the fact that γ is a fixed constant. □

The following theorem concludes our argument for the exponential behavior of total score of irrelevant features. It relates the probability of selecting a feature f_i irrelevant on U to the probability of including U in the sample.

Theorem 5. If a feature f_i over-fits on a set of samples U with size r , then:

$$\lim_{n \rightarrow \infty} S(f_i) = d\gamma^r.$$

Proof. From proposition 2 we have:

$$\begin{aligned} S(f_i) &= d\Pr(\mathbf{f} = f_i) \\ &= d \sum_B \Pr(f_i \in F^B) \Pr(B) \sum \Pr \\ &= d \Pr(U \subseteq B) \\ &= d(\gamma^r + O(n^{-1})). \end{aligned}$$

The last equation was proved in lemma 4, and the one before that from definition 3. □

Although we presented the above arguments for the Lasso, it also should work for any other feature selection algorithm which exhibits linearity in its feature score curve. That is, features corresponding to the linear part of the scoring curve are indeed the irrelevant features for that algorithm, and therefore, the features on non-linear upper-part should be considered as informative ones. Obviously the features on the non-linear lower-part are not interesting for the any prediction task because their scores are even less than the irrelevant features. We speculate that these features do not present a linear behavior because not only they are not relevant to the outcome, but also they are not associated with any particular set U , meaning they are not even included in an over-fitted model. A follow-up study may investigate this hypothesis further.

Experiment with real data

We applied FeaLect on several datasets to test the performance of our feature selection algorithm in real life conditions.

Lymphoma

Lymphoma is a cancer that begins in the lymphatic cells of the immune system, and is presented as a solid tumor of lymphoid cells [14]. Just as cancer represents many different diseases, lymphoma represents many different cancers of lymphocytes [15]. We applied our algorithm for automatic diagnosis of lymphoma types based on flow cytometry (FCM) data [16]. Usually 15-30 markers are used for each patient, where each marker distinguishes a particular

cell type based on its protein content. We analyzed flow cytometry data of 85 lymphoma patients who had been diagnosed at BC Cancer Agency, Vancouver, Canada between 2004-2007. The patients were grouped into four top-level disease subgroups and the goal was to build a classifier that could diagnosis 20 test patients based on their FCM data. For each group, we trained a classifier to distinguish that group versus the others. These four classifiers were then combined to provide the top-level diagnosis.

Data preparation and feature extraction

The blood sample of each patient was divided into 7 portions, and each portion is examined in a different tube by the cytometer. Each tube gives 5 dimensional data of 20,000-70,000 blood cells. In the first analysis step, we used a spectral clustering approach to cluster the cells in each tube into cell populations. It was not possible to directly apply classical spectral clustering [17-20] to the lymphoma data because it involved computing eigenvectors of a big n -by- n matrix where n ranges from 20,000 to 70,000. Instead, we have made use of SamSPECTRAL that is our enhanced spectral clustering method capable of analyzing large amount of data in a reasonable amount of time; it has also a good memory footprints [21].

SamSPECTRAL performs a specific sampling stage called *faithful sampling* to reduce the size of data for spectral clustering. Our data reduction scheme is designed to preserve density information and can be briefly stated as follows:

1. Set all points to be unregistered and assume the parameter h is adjusted appropriately.
2. Pick a random unregistered point p (the representative of a community) and find all unregistered data points within distance h from p .
3. Put all of these points in a set called community p , and label them as registered.
4. Repeat the above two steps until no unregistered points are left.

After the above steps, the similarity between the communities is defined by summing up similarities between their members, and the resulting similarity matrix is passed to a classical spectral clustering algorithm. Because this matrix is much smaller than the original similarity matrix (3000-by-3000 instead of 20,000-by-20,000 in our experiments), its eigenvectors can be efficiently computed in reasonable time.

Each cluster computed by SamSPECTRAL was regarded as a "cell population" that could potentially have information about the lymphoma type. Without imposing any *a priori* knowledge on the importance of any population, we considered their sizes and their means in all dimensions as

features. In total, 276 features were obtained and ignoring those with very low variance, 224 were kept for feature selection and classification.

Feature selection and classification

Since the number of features was considerably larger than the number of training samples ($p = 224$, $n = 85$), a careful feature selection scheme was needed. To reduce the computation time required, we imposed a pre-defined upper bound 60 on the number of features based on *a priori* knowledge from the biology. We initially applied ℓ_1 -regularization technique, and it was not by its own enough to prevent overfitting. Reducing the regularization parameter did not improve the results as we observed that some of the features that were known to be biologically and clinically interesting were ignored. We also applied Bolasso [7] to select relevant features. For most bootstrap samples of our data, the global error defined by equation (1) was minimized when only a few (less than 4) features were selected. Because the intersection of selected features from several (more than 10) samples was empty, Bolasso could not result in appropriate feature selection.

Next, we applied our feature selection algorithm. In our experiment, we set $\gamma = \frac{3}{4}$ to be the fraction of instances used in each iteration for training. Figures 2(a) and 2(b) depict the resulting feature scores for follicular lymphoma type after $m = 1000$ random runs. The log-scale plot consisted of a linear part that confirmed our hypothesis experimentally. Similarly, the plots for other types of lymphoma had also linear parts. Furthermore, we re-ran the experiments with $m = 5000$ random samples and the results did not vary significantly indicating a fast convergence rate for the feature scores.

To select the informative features, we fitted a 3-segment spline model to each curve. The features corresponding to the middle linear segment were considered as irrelevant ones, and ignored for the rest of analysis. Features with score higher than score of these irrelevant features were selected as informative features. We observed that unlike the pure Lasso, all features that were known to be biologically and clinically interesting were selected by our approach. Prediction accuracy was improved confirming the efficiency of our feature selection method. We used our selected features to build a linear classifier that had precision, recall and F-measure 98%, 94% and 96%, respectively while the best result we obtained with the pure Lasso was 93%, 82% and 87%, respectively.

For further evaluation in a data exploratory setting, we interrogated the *selected features* together with our clinical collaborators for novel biomarker discovery. A task which would be challenging otherwise, due to large number of features and huge amount of clinical work required to evaluate each individual feature. We narrowed down our attention to those features which were relevant to

Table 1 Comparison of area under the ROC curve between FeaLect, lars, and Bolasso on six different datasets.

Dataset	Total samples	# of features	20 training samples			40 training samples			Reference
			Bolasso	lars	FeaLect	Bolasso	lars	FeaLect	
Lymphoma	258	505	0.62	0.81	0.84	0.67	0.87	0.88	current
Colon	62	2000	0.50	0.57	0.65	0.47	0.64	0.75	[23]
Arcene	100	10000	0.51	0.59	0.64	0.50	0.66	0.72	[26] (UCI)
SECOM	208	590	0.51	0.57	0.61	0.52	0.61	0.64	[25] (UCI)
Connectionist	208	60	0.63	0.76	0.78	0.67	0.78	0.79	[27] (UCI)
ISOLET	479	617	0.90	0.99	1.00	0.91	1.00	1.00	[28] (UCI)

lymphoma types based on our feature selection algorithm, but were not previously reported to be biologically relevant. This approach resulted in the discovery of interesting new criteria for lymphoma diagnosis that have clinical applications in practice [22].

Additional real datasets

In addition to our lymphoma flow cytometry data, we validated the performance of FeaLect on five other datasets including the well-known colon gene expression (Table 1). Colon dataset contains expression of 2000 genes in 22 normal and 40 colon cancer tissues [23] and it is a benchmark for gene expression analysis [24]. All additional four datasets are from UCI (University of California, Irvine) Machine Learning Repository [25]. Arcene contains mass-spectrometric data for cancer and normal cases [26], variables of SECOM were collected from sensors and process measurements in complex modern semi-conductors with the goal of enhancing current business improvement techniques [25]. We used a version of SECOM dataset balanced by randomly selecting equal

number of positive and negative samples. The learning task for Connectionist dataset is to train a network to discriminate between sonar signals bounced off a metal cylinder and those bounced off a roughly cylindrical rock [27]. ISOLET is a natural language processing dataset generated from speech of 150 subjects with the goal of identifying which letter-name was spoken [28]. In the current study, we only considered letters A and B as positive and negative samples and excluded the rest of samples.

Table 1 compares the performance of Bolasso, pure lars, and FeaLect on the studied datasets. Training samples were selected uniformly at random and area under the ROC curves (AUC) were computed using the rest of samples (Figure 3). For each dataset, we repeated this procedure 100 times and reported the average AUC to avoid any dependency on the random selection of train-test sets. Both FeaLect and lars always outperformed Bolasso. When only 20 random training samples were provided, FeaLect provides significantly better than pure lars except ISOLET dataset. The number of samples in

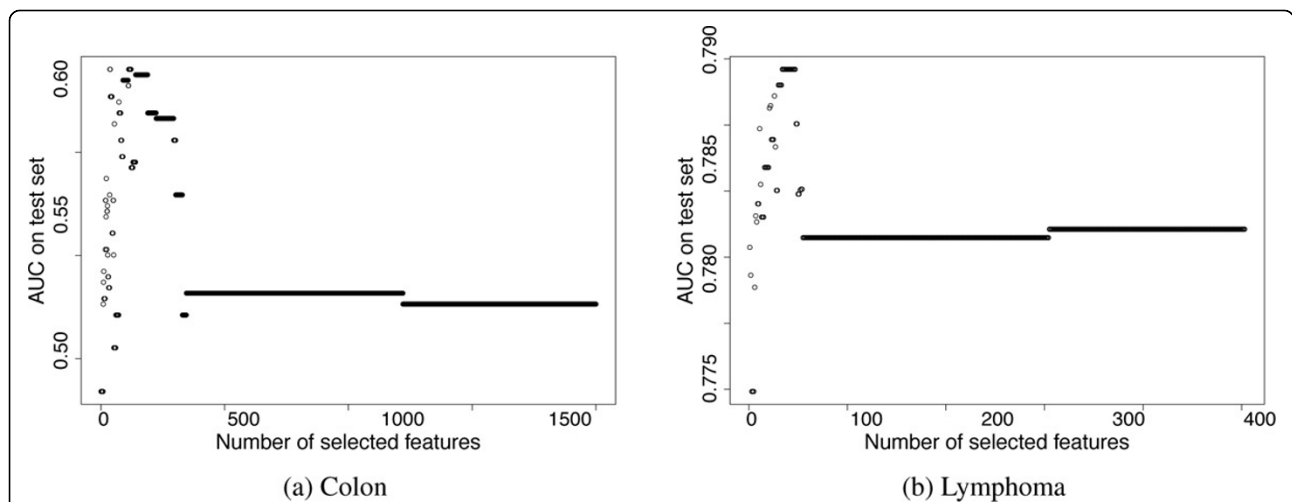
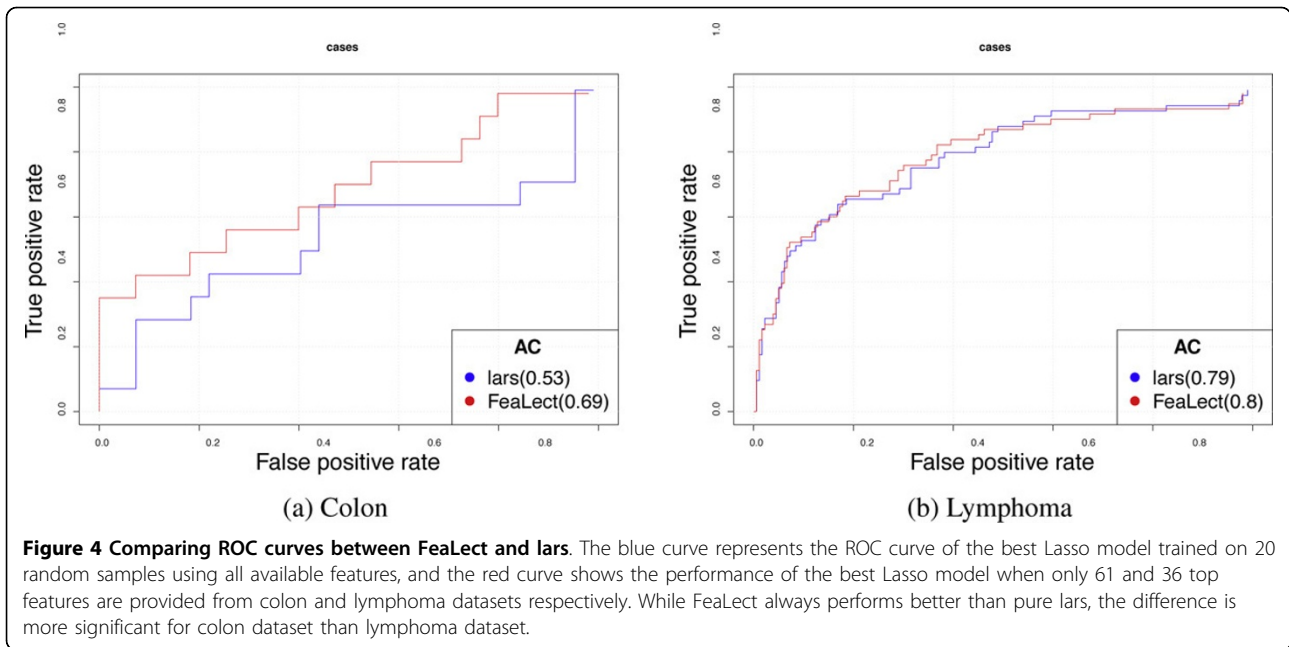


Figure 3 Variation of area under the ROC curve when different number of features are used. The features are sorted by applying FeaLect on 20 random training samples. Then, the training samples and the highly scored features are considered to build linear classifiers by lars. The best AUC is reported by testing on a set of validating samples disjoint from the training set. For both lymphoma and colon datasets, the performance of the optimum classifier decreases if all features are provided to lars. This observation practically shows the advantage of using a limited number of highly scored features over pure lars.

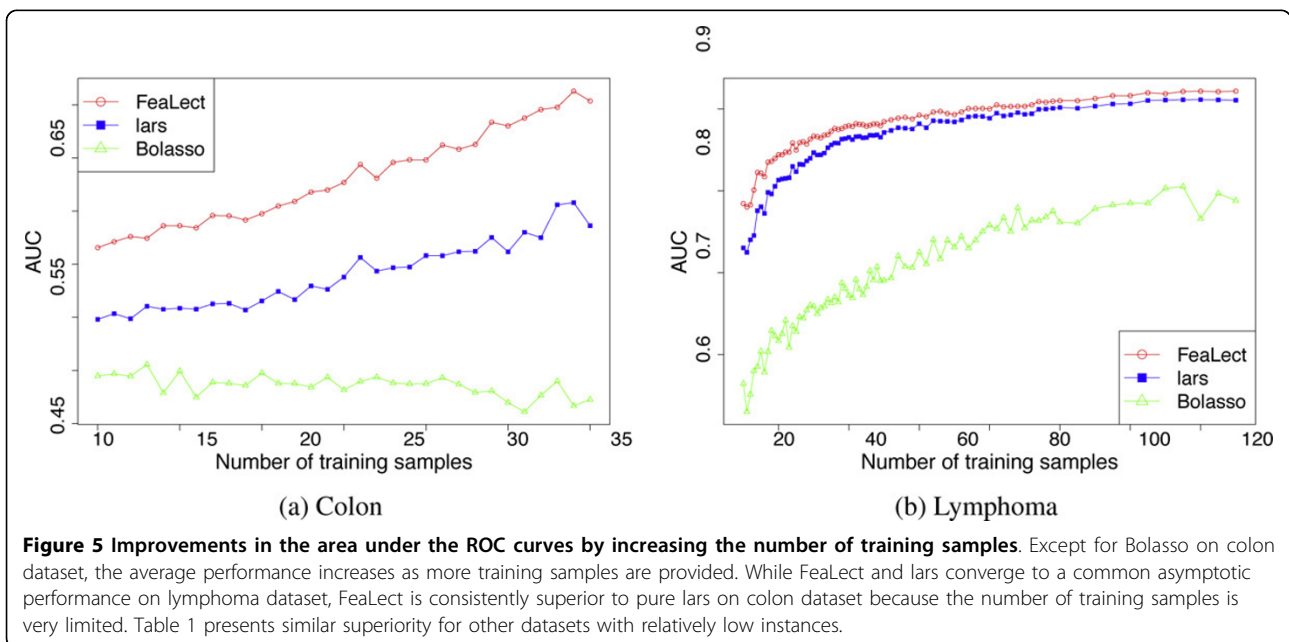


ISOLET dataset is more than other datasets, enough such that both methods performed well. The superiority of FeaLect over lars decreases as the number of training samples increases from 20 to 40, except for Colon and Arcene datasets for which FeaLect is still better than lars by .09 and 0.6, accordingly. Interestingly, these two datasets are the ones with 2000 and 10000 features that are considerably higher dimensional than other datasets. This observation reassures that FeaLect is advantageous over lars in high-dimensional settings and their performance

converges as “adequate” number of samples are provided (Figures 4 and 5).

Conclusion

We have presented FeaLect, a novel feature selection algorithm, based on Lasso (Figure 1). The idea of FeaLect is to combine the selected feature sets to score the features according to their relevancy and prediction power. An advantage of FeaLect compared to many other feature selection methods is to provide a ranking for features



relevance, which can be leveraged in better prediction models and/or exploratory data analysis. We reported a cancer classification problem (lymphoma diagnosis) for which distinguishing the most relevant features is of great interest from the biological and clinical point of view. FeaLect has led to the discovery of novel biomarkers for this disease to help clinicians in lymphoma sub-type diagnosis [22]. The log-scale score curve can be studied in more detail and explaining its behavior in the non-linear parts is potentially a source of insight. Shedding more light on the Lasso performance by studying feature scores is a possible future direction of this study.

Furthermore, we provided empirical and quantitative evaluations on five other real-world datasets (from different fields) to confirm the superiority of our method, in prediction performance, compared to the baselines.

Acknowledgements

The authors would like to thank Andrew Weng and Randy Gascoyne for providing data and valuable clinical insight, and Nima Aghaeepour for his scientific comments.

Author details

¹Department of Genome Sciences, University of Washington, Seattle, Washington, USA. ²Department of Computer Science, University of British Columbia, Vancouver, BC, Canada. ³Faculty of Information Technology, Monash University, VIC, Australia. ⁴Medical Genetics, University of British Columbia, Vancouver, BC, Canada.

Authors' contributions

AG and RB supervised the project and motivated the study by providing scientific insight. HZ developed the idea of scoring features and performed the experiments. HZ and GH designed the mathematical analysis. RB provided data and computing facilities. All authors read, edited and approved the final manuscript.

Declarations

The research and publication costs for this article were funded by NIH grants 1R01EB008400 and 1R01EB005034, the Michael Smith Foundation for Health Research, the National Science and Engineering Research Council and the MITACS Network of Centres of Excellence.

This article has been published as part of *BMC Genomics* Volume 14 Supplement 1, 2013: Selected articles from the Eleventh Asia Pacific Bioinformatics Conference (APBC 2013): Genomics. The full contents of the supplement are available online at <http://www.biomedcentral.com/bmcgenomics/supplements/14/S1>.

Competing interests

The authors declare that they have no competing interests.

Published: 21 January 2013

References

1. Tibshirani R: **Regression Shrinkage and Selection via the Lasso.** *Journal of the Royal Statistical Society (Series B)* 1996, **58**:267-288.
2. Hastie T, Rosset S, Tibshirani R, Zhu J: **The entire regularization path for the support vector machine.** *Journal of Machine Learning Research* 2005, **5**(2):1391.
3. Efron B, Hastie T, Johnstone L, Tibshirani R: **Least angle regression.** *Annals of Statistics* 2004, **32**:407-499.
4. Zhao P, Yu B: **On Model Selection Consistency of Lasso.** *J Mach Learn Res* 2006, **7**:2541-2563.
5. Zou H, Hastie T: **Regularization and variable selection via the Elastic Net.** *Journal of the Royal Statistical Society, Series B* 2005, **67**:301-320.
6. Wainwright MJ: **Sharp thresholds for high-dimensional and noisy sparsity recovery using l_1 -constrained quadratic programming (Lasso).** *IEEE Trans Inf Theor* 2009, **55**(5).
7. Bach FR: **Bolasso: model consistent Lasso estimation through the bootstrap.** *ICML '08: Proceedings of the 25th international conference on Machine Learning* 2008.
8. Bach F: **Model-consistent sparse estimation through the bootstrap.** *Tech rep, HAL-00354771* 2009.
9. Meinshausen N, Yu B: **Lasso-type recovery of sparse representations for high-dimensional data.** *Annals of Statistics* 2009, **37**:246-270.
10. Lounici K, et al: **Sup-norm convergence rate and sign concentration property of Lasso and Dantzig estimators.** *Electronic Journal of statistics* 2008, **2**:90-102.
11. Meinshausen N, Bühlmann P: **Consistent neighborhood selection for sparse high-dimensional graphs with the Lasso.** *Statist Surv* 2004, **2**:61-93.
12. Efron B, Tibshirani RJ: *An Introduction to the Bootstrap* (Chapman & Hall/CRC Monographs on Statistics & Applied Probability) Chapman and Hall/CRC; 1998.
13. Breiman L: **Bagging predictors.** *Machine learning* 1996, **24**(2):123-140.
14. Matasar MJ, Zelenetz AD: **Overview of Lymphoma Diagnosis and Management.** *Radiologic Clinics of North America* 2008, **46**(2):175-198.
15. Hiddemann W, Longo D, Coiffier B, Fisher R, Cabanillas F, Cavalli F, Nadler L, De Vita V, Lister T, Armitage J: **Lymphoma classification-the gap between biology and clinical management is closing.** *Blood* 1996.
16. Hawley TS, Hawley RG: *Flow Cytometry Protocols, Methods in Molecular Biology*. 2 edition. Humana Press; 2005.
17. von Luxburg U: **A tutorial on spectral clustering.** *Statistics and Computing* 2007, **17**(4):395-416.
18. von Luxburg U, Belkin M, Bousquet O: **Consistency of Spectral Clustering.** *Annals of Statistics* 2008, **36**(2):555-586.
19. Ng AY, Jordan MI, Weiss Y: **On Spectral Clustering: Analysis and an algorithm.** *Advances in Neural Information Processing Systems* 14 2001.
20. Azran A, Ghahramani Z: **Spectral Methods for Automatic Multiscale Data Clustering.** *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, Volume 1* 2006, 190-197.
21. Zare H, Shooshtari P, Gupta A, Brinkman R: **Data reduction for spectral clustering to analyze high throughput flow cytometry data.** *BMC Bioinformatics* 2010, **11**:403.
22. Zare H, Bashashati A, Kridel R, Aghaeepour N, Haffari G, Connors J, Gascoyne R, Gupta A, Brinkman R, Weng A: **Automated analysis of multidimensional flow cytometry data improves diagnostic accuracy between mantle cell lymphoma and small lymphocytic lymphoma.** *American Journal of Clinical Pathology* 2012, **137**:75-85.
23. Alon U, Barkai N, Notterman D, Gish K, Ybarra S, Mack D, Levine A: **Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays.** *Proc Natl Acad Sci USA* 1999, **96**(12):6745-6750.
24. Stingo F, Vannucci M: **Variable selection for discriminant analysis with Markov random field priors for the analysis of microarray data.** *Bioinformatics* 2011, **27**(4):495.
25. Frank A, Asuncion A: **UCI Machine Learning Repository.** 2010.
26. Guyon I, Hur AB, Gunn S, Dror G: **Result analysis of the NIPS 2003 feature selection challenge.** *Advances in Neural Information Processing Systems* 17 MIT Press; 2004, 545-552.
27. Gorman R, Sejnowski T: **Analysis of hidden units in a layered network trained to classify sonar targets.** *Neural networks* 1988, **1**:75-89.
28. Fandy MA, Cole R: **Spoken Letter Recognition.** *NIPS* 1990, 220.

doi:10.1186/1471-2164-14-S1-S14

Cite this article as: Zare et al.: Scoring relevancy of features based on combinatorial analysis of Lasso with application to lymphoma diagnosis. *BMC Genomics* 2013 **14**(Suppl 1):S14.