Neuroinformatics
https://doi.org/10.1007/s12021-018-9371-3

ORIGINAL ARTICLE

Fast, Accurate, and Stable Feature Selection Using Neural Networks

James Deraeve[1] · William H. Alexander[1]

Abstract
Multi-voxel pattern analysis often necessitates feature selection due to the high dimensional nature of neuroimaging data. In this context, feature selection techniques serve the dual purpose of potentially increasing classification accuracy and revealing sets of features that best discriminate between classes. However, feature selection techniques in current, widespread use in the literature suffer from a number of deficits, including the need for extended computational time, lack of consistency in selecting features relevant to classification, and only marginal increases in classifier accuracy. In this paper we present a novel method for feature selection based on a single-layer neural network which incorporates cross-validation during feature selection and stability selection through iterative subsampling. Comparing our approach to popular alternative feature selection methods, we find increased classifier accuracy, reduced computational cost and greater consistency with which relevant features are selected. Furthermore, we demonstrate that importance mapping, a technique used to identify voxels relevant to classification, can lead to the selection of irrelevant voxels due to shared activation patterns across categories. Our method, owing to its relatively simple architecture, flexibility and speed, can provide a viable alternative for researchers to identify sets of features that best discriminate classes.

Keywords Feature selection · fMRI · MVPA · Machine learning

Introduction

The last decade has seen a marked increase in the number of imaging studies utilizing multi-voxel pattern analysis (MVPA). MVPA is a collection of machine learning techniques that allows a model-free approach to decoding mental states from distributed patterns of activity in imaging studies. This is in contrast to traditional univariate statistics, which look at the relationship between cognitive variables and BOLD activity, typically using a General Linear Modeling approach. Several notable benefits of MVPA compared to traditional analyses are more sensitive detection of cognitive states, increased temporal resolution allowing us to relate brain activity to behavior on a short timescale and characterizing how the brain represents cognitive states (Norman et al. 2006).

While MVPA approaches have the potential to reveal cognitive states underlying BOLD activity with much more sensitivity than traditional univariate approaches, a number of methodological obstacles must be addressed before the full impact of MVPA methods can be realized. A critical obstacle in this respect is the relative sparseness of observations relative to the number of features (Guyon and Elisseeff 2003). Due to the coarse temporal resolution of fMRI, which typically requires 2 s to record BOLD activity throughout the entire brain, only a limited number of observations are feasible for a specific condition in a reasonable fMRI study. In contrast, the number of potential features (voxels) available for use in MVPA classification is quite high due to the fine spatial resolution allowed by fMRI. The asymmetry between the number of features and number of observations to classify is problematic because, as the dimensionality of the feature space increases, the space in which the observations are to be classified grows geometrically. As a result, observations appear to be sparsely distributed within this high-dimensional space, and distinct from each other, with the consequence that classifiers used in MVPA may be unable to estimate accurate decision boundaries, or may overfit on training data. Thus, the performance of a classifier will decrease as the dimensionality of the features becomes too large, because the classifier picks up on peculiarities of the data and loses generalizability to new observations.

✉ James Deraeve
james.deraeve@ugent.be

[1] Department of Experimental Psychology, Ghent University, Henri Dunantlaan 2, B-9000 Ghent, Belgium

Published online: 21 March 2018

A number of methods have been devised to ameliorate this problem, known as the curse of dimensionality (Mahmoudi et al. 2012). One approach is cross-validation, in which the original data is partitioned into complementary subsets: a training set, a validation set, and a test set. During classifier training, the validation set is used to tune parameters and monitor for overfitting, while the unrelated test set is then used to assess performance of the classifier on data it was not trained on. A second method for addressing the curse of dimensionality is through reducing the dimensionality directly (Cao and Chong 2002). One class of approaches is feature extraction, where the original set of features is replaced by a smaller set of new features derived from the original features. A popular feature extraction method is principal component analysis where the number of features is transformed into a new set of features that are linearly uncorrelated (principal components) and are sorted in terms of how much variance in the data they account for.

A third approach to dimensionality reduction is feature selection, where an optimal subset is chosen from among the original features. These are usually categorized as either wrapper, embedded or filter techniques based on how the selection algorithm and model building are combined (Mwangi et al. 2014; Saeys et al. 2007). Wrapper methods evaluate the "utility" of features based on classifier performance, and thus directly work to solve the problem of optimizing classification accuracy, which is often the end goal of feature selection. A popular wrapper approach is recursive feature elimination (RFE) (De Martino et al. 2008; Guyon et al. 2002), which is a backward feature selection procedure that prunes irrelevant features until optimal accuracy is obtained. A major drawback to this approach is the computational cost of iterative learning steps and the need for cross-validation. In contrast, filter approaches are much faster but do not attempt to maximize accuracy (Das 2001). Instead, they apply statistical measures to score each feature on its "relevance". Examples of this are correlation-based feature selection (Hall 1998), ReliefF (I. Kononenko and Simec 1995) and mutual information (Vergara and Estévez 2014). Lastly, embedded methods are similar to wrapper methods in that they are also used to optimize performance of a learning algorithm (Chandrashekar and Sahin 2014). However, with embedded methods, the subset size selection is an inherent part of the model and not done separately. The most common embedded feature selection methods are regularization methods, which use constraints and penalizations to eliminate features during model building, e.g. LASSO regression (Ma and Huang 2008). Beyond their utility in alleviating the curse of dimensionality, feature selection methods can also decrease the time needed for classification, produce results that facilitate interpretation and, perhaps most importantly, increase the accuracy of classification (Chandrashekar and Sahin 2014). With respect to this last point, it has been found that increases in accuracy following feature selection can be inconsistent and are highly dependent on the nature of the data and on the methods used (Chu et al. 2012; Kerr et al. 2014).

The past two decades have seen an explosion of proposed feature selection algorithms and choosing the method that best fits your data can be a daunting task. According to Li et al. (2017), traditional feature selection algorithms for generic data can be grouped into four categories based on the techniques adopted during the feature selection process: similarity based, information theoretical based, sparse learning based and statistical based algorithms. Similarity based algorithms determine feature importance by looking at how well features preserve data similarity. The aforementioned ReliefF method can be seen as a similarity based algorithm and other examples include SPEC (Zheng Zhao and Liu 2007) and the Trace Ratio Criterion (Nie et al. 2008). A potential downside of these methods is that they do not remove redundant features during the selection process. Redundancy occurs when features are highly correlated. Thus, including redundant features, even though they are relevant, imparts no additional information, which can lead to increased training times and decreased accuracy (Ding and Peng 2005). When dealing with redundant data, theoretic based algorithms are often considered. Examples of such methods which consider both feature relevance and feature redundancy are conditional mutual information maximization (Fleuret 2004) and minimum redundancy maximum relevance (Peng et al. 2005). Sparse learning based methods are essentially embedded feature selection methods, although RFE-SVM is also included in this category due to its iterative pruning of features as part of the algorithm. Lastly, statistical based methods analyze features individually and rely on statistical measures to assess the importance of a feature. The F-score (Wright 1965) and Chi-Square score (Liu and Setiono 1995) are popular and widely used examples of this category. An in-depth overview of these methods and more can be seen in the review paper by Li et al. (2017).

While feature selection's primary function is usually to improve predictive performance, it can also be used to identify meaningful sets of features that are important contributors to classification (Norman et al. 2006). Such subsets of features can then more efficiently predict new data, while effects from noise or irrelevant features are reduced. A smaller amount of highly informative voxels may also improve interpretation and provide insight into how cognitive states are represented anatomically. However, in order to aid interpretation, it is vital that the set of voxels selected in this way is robust despite variance in training data (Dernoncourt et al. 2014). This sensitivity to changes in the training set is called stability and has been extensively studied in learning algorithms (Turney 1995), but only more recently investigated for feature selection (Alexandros Kalousis et al. 2007). Stability is generally measured by looking at relations between feature sets, rankings or weights. Metrics include Pearson correlation, Spearman rank correlation, Hamming distance (Saeys et al.

2008), Jaccard similarity (Fan and Chou 2016) and entropy (Křížek et al. 2007). General frameworks and methods to increase stability for feature selection have also been developed recently. Examples of this are ensemble classification methods (Saeys et al. 2008), stability selection (Meinshausen and Bühlmann 2010), Random Subspace Bayesian Learning (Yan et al. 2014) and combined performance/stability metrics (Kirk et al. 2013).

While feature selection methods with high stability might consistently select the same features, there is no guarantee that these features are relevant, i.e. a stable feature selection method might consistently identify features that do not improve classification accuracy. Conversely, a method can improve accuracy, but select wildly varying sets of features each iteration, since different sets of features can result in similar accuracy scores, which can occur when there are many correlated variables or features with similar information content (Křížek et al. 2007). Thus, the problem of feature selection is to find methods which can reliably extract all voxels important to classification as this will lead to the better interpretability and good predictive performance.

Unfortunately, in many cases, we do not have knowledge of what the important features are, otherwise feature selection would be unnecessary. This is especially true for Big Data sets, such as fMRI, in which thousands of features might contribute to classification accuracy, but it is unknown a priori which features those may be. Assessing the relative performance of feature selection methods on real data is thus problematic, and may benefit from the use of simulated data resembling experimental data so that we know the precise data generation procedure and which features are truly important to classification. Such data sets can serve an important role in benchmarking feature selection methods (Bolón-Canedo et al. 2013), yet they are rarely used compared to real data or alongside it for the purpose of testing feature selection methods.

In short, while feature selection methods have the potential to greatly enhance the application of MVPA to fMRI data, a number of questions must be addressed before this can take place.

Goal of the Current Study

In this study, we propose a novel procedure for consistently extracting relevant features from simulated and fMRI data. This procedure is based on a single-layer neural network with cross-validation after weight-updating. Our procedure makes use of a variant of stability selection by summing weights across iterations and choosing only those features whose weights pass a predefined threshold. The method is simple, fast, and straightforward to implement. We compare our approach to existing methods for feature selection or importance mapping, on simulated and real experimental data and find that our approach consistently outperforms the alternative methods on stability, detection of relevant voxels and predictive accuracy.

Methods

Classifiers and Feature Selection Methods

Support Vector Machines

Support vector machines (SVM) (Boser et al. 1992; Cortes and Vapnik 1995) are a popular supervised machine learning technique used in classification and regression problems. It attempts to separate points in n-dimensional space by finding the hyperplane that maximizes the distance between the nearest points of each class (support vectors). SVM's can handle non-linear decision boundaries by making use of the kernel trick, but in this study we limit ourselves to linear SVM's. The formulation for the optimization of the soft margin SVM used here is as follows:

$$\begin{aligned} &\min_{\boldsymbol{w}\in\mathbb{R}^d,\xi_i\in\mathbb{R}^+} \|\boldsymbol{w}\|^2 + C\sum_i^N \xi_i \\ &\text{Subject} \quad \text{to}: y_i\left(\boldsymbol{w}^T x_i + b\right) \geq 1-\xi_i \text{ for } i = 1\ldots N \end{aligned} \tag{1}$$

Where ξ_i are "slack" variables that allow misclassified data points and C is a regularization parameter so that if C is small, constraints can easily be ignored and if C is large they are hard to ignore. This ensures convergence even when the problem is not perfectly separable. The case where $C = \infty$ is equal to the hard margin SVM.

Classifying new trials x_{new} can be done by evaluating:

$$sign\left(w^T x_{new} + b\right) \tag{2}$$

In this study, SVM's were used in conjunction with RFE, to establish the feature subset size (Haxby data) and to test performance after feature selection. A multi-class linear SVM was used for the Haxby data set and a binary SVM for the simulated data.

RFE-SVM

RFE is a wrapper approach which begins with the full feature set and iteratively removes a prespecified number of features after evaluation. It was originally created to be used in tandem with SVM's (De Martino et al. 2008; Guyon et al. 2002). SVM weight values are used as a ranking criterion for backward feature elimination. Here we added feature rankings across iterations to determine those most important for classification and then used a feed-forward cross-validation procedure (Haxby data) or the 20 most important features (simulation) to determine subset size.

RFE-SVM is a well-established technique that has been extensively benchmarked and compared to other techniques (Bolón-Canedo et al. 2013; Chu et al. 2012; Dittman et al. 2011; Fan and Chou 2016; Haury et al. 2011; Alexandros Kalousis et al. 2007; Stiglic and Kokol 2010; Tohka et al.

2016). A general outline of the RFE-SVM procedure can be seen in Table 1. This procedure is embedded in a general procedure that is common to all methods described in this paper (Tables 3 and 4).

Importance Mapping

While RFE-SVM is a current state-of-the-art feature selection technique that can be used to detect relevant features for classification, early proposals for determining important features that employ neural networks (NN) are still widely used, possibly because it is less computationally intensive to identify features by the weights learned in neural network approaches. One such approach is "importance mapping" (Polyn et al. 2005) used extensively in fMRI studies, in which a single-layer NN is utilized as a classifier for MVPA, after which the weights of the network are multiplied by average feature activity for each class to determine feature importance.

$$imp_{ij} = w_{ij} * avg_{ij} \tag{3}$$

Where w_{ij} is the weight between input i and output unit j and avg_{ij} is the average activity of input i for category j.

This method has been used in several papers since (e.g. Johnson et al. 2009; Lewis-Peacock et al. 2011; McDuff et al. 2009; Saarimäki et al. 2016), and is also featured in the Princeton MVPA toolbox (http://code.google.com/p/princeton-mvpa-toolbox/), one of the most popular toolboxes for machine learning with neuroimaging data. However, its detection efficacy has, to our knowledge, never been properly tested. While the rationale for multiplying weight strengths by average voxel activity in order to identify important features is that the effect a particular feature has on classification, in the NN framework, is the strength of the input multiplied by the strength of the weight, informal tests in our lab suggest a potential problem with this approach. Specifically, using simulated data, we observed that multiplication by average voxel activity might allow voxels with high activity values to be deemed important even though they do not contribute meaningfully to the classification. An example of this would be voxels that show a similar pattern of increased activity for two categories that are being classified. These "overlapping" voxels do not carry any discriminating information, yet have high activity values, leading to their possible identification as important features. While this method is technically not used as feature selection (since no subsequent analyses are performed on the identified features), it has a similar goal: to determine which features contribute to the classification. If this method were to be used for feature selection, the extracted irrelevant features could increase dimensionality without a commensurate increase in predictive accuracy, the opposite of the goal of feature selection techniques. We implemented the importance mapping method in the same manner as our NN feature selection method (see below), with one difference: the multiplication by average voxel activity of the weights specified by the importance mapping approach.

Iterative NN with Cross-Validation

Our version of a NN feature selection method is again a wrapper approach, because it looks at classification accuracy to determine important features. We used a single-layer neural network where the features serve as inputs, with output nodes for each of the categories. Each run lasted 50 training epochs and on every epoch weights w_i were batch updated according to the delta rule:

$$\triangle w_i = \alpha(t - g(h))g'(h)x_i \tag{4}$$

Where α is the learning rate (fixed at 0.01), $g()$ is the sigmoid function and x_i are the input values. Weights were initialized by drawing from a normal distribution with mean 0 and standard deviation 0.01.

Table 1: Pseudocode outline for the RFE-SVM procedure. This procedure returns a sorted ranking starting with the most discriminating classification feature.

Algorithm Feature selection: RFE-SVM

```
for 20 iterations do
    train_samples = 90 % of training set
    while # features > 0 do
        train SVM on train_samples
        sort features based on abs(weights)
        eliminate feature with smallest score
    end while
    save ranked list of eliminated features
end for
summed_ranks = sum(feature_ranks)
sorted_importance = sort(summed_ranks)
return sorted_importance
```

At each epoch, performance of the neural network was tested on a validation set to prevent overfitting. Following the completion of a training run, weights from the best performing epoch on the validation set were stored, and weights were summed over all training runs. These summed weights were then sorted according to their absolute value, with the highest absolute values belonging to the most important classifying features. An outline of the procedure can be seen in Table 2.

F-Test

The F-test is one of the oldest and most widely used forms of feature selection because of its simple formulation, easy interpretation and inclusion in most statistical packages. It has been used extensively with regard to fMRI research (Cox and Savoy 2003; Zeithamova et al. 2017) and is directly available in many libraries and packages for machine learning (Hebart et al. 2015; Pedregosa et al. 2011). This method decomposes the variability of the data in terms of between- and within-class variability. The ratio of these variabilities for each voxel gives that voxel's F-value, which is used to rank its importance.

$$F = \frac{\text{between-class variability}}{\text{within-class variability}} \tag{5}$$

Our implementation of the F-test is very similar to that of the other techniques mentioned here (Tables 1 and 2), with F-values for each separate voxel summed across all 20 iterations, resulting in a ranking of feature importance.

Mutual Information

Mutual information is an information theoretic measure for the relationship between two random variables. Noteworthy advantages of this measure are: 1) its invariance to transformations of the feature space that preserve order of the original elements, 2) its ability to measure any sort of relation between random variables, including non-linear relationships, and 3) its easy interpretation as the amount of shared information between the variables (Vergara and Estévez 2014). Owing to these desirable properties, it's often been used in various forms for feature selection (Michel et al. 2008; Sayres et al. 2005). Here, we limit ourselves to the form described in Ross (2014), which relies on nonparametric methods based on entropy estimation from k-nearest neighbor distances to measure the dependency between a continuous and discrete data set (i.e. feature values and class labels). We used the default implementation of this algorithm (with k neighbors = 3) from the scikit-learn package. The procedure is again very similar to that of the other methods (Tables 1 and 2), where on each iteration, the mutual information is estimated for each voxel and then summed to return an array of feature importance.

ReliefF

Our last feature selection method is ReliefF, which is a multivariate filter technique based on measuring features' capability in preserving sample similarity (Zhao et al. 2013). It is an extension of the Relief algorithm, which is only suitable for binary problems. A big advantage of this technique is that it is able to detect conditional dependencies between features (Igor Kononenko et al. 1997). Within ReliefF, the importance of a feature increases if it is more similar to itself in nearby instances of the same class (nearest hits) than in nearby instances of other classes (nearest misses). A more exact formulation of ReliefF, as described in Zhao et al. (2013) is as follows.

Assuming c classes with l instances in each class; all features have been normalized to unit length; and both $NH(\boldsymbol{x})$ and $NM(\boldsymbol{x})$ have k instances, the evaluation criterion of ReliefF is equivalent to:

Table 2: Pseudocode outline for the NN procedure. This procedure returns a sorted ranking starting with the most discriminating classification feature.

```
Algorithm Feature selection: Neural network
for 20 iterations do
    train_samples = 90 % of training set
    for 50 epochs do
        forward propogation using train_samples
        update weights with delta rule
        weights = updated weights
        val_acc = acc on validation set
    end for
    best_weights = weights for epoch with max(val_acc)
end for
summed_weights = abs(sum(best_weights))
sorted_importance = sort(summed_weights)
return sorted_importance
```

$$\sum_{i=1}^{n}\left(\sum_{j=1}^{k}\frac{1}{k}\left(f_i-f_{NH(\boldsymbol{x}_i)_j}\right)^2-\sum_{y\neq yi}\frac{\sum_{j=1}^{k}\left(f_i-f_{NM(\boldsymbol{x}_i,y)_j}\right)^2}{(c-1)k}\right) \tag{6}$$

With f_i as the value of the feature **f** on the ith instance, $\boldsymbol{x}_i$; $NH(\boldsymbol{x}_i)_j$ denotes the jth nearest hit of $\boldsymbol{x}_i$; and $NM(\boldsymbol{x}_i, y)_j$ denotes the jth nearest miss of $\boldsymbol{x}_i$ in class y.

We used this version of ReliefF with k = 5 neighboring instances (nearest hits/misses). We used the same general procedure as with the other methods, where we calculate the ReliefF values for all voxels on each iteration and sum these, resulting in a sorted array of feature importance.

Data Sets

Simulated Data Set

The 3 methods were first compared on 5 simulated data sets with varying signal-to-noise ratios (SNR). The advantage of this is that it offers a fully controlled environment where we know beforehand which features contain relevant classification information. Each of these 5 data sets consisted of 15 × 3 subsets (training, test and validation) of 100 trials and 300 features. The 100 trials were split up in 50 trials of category A and 50 trials of category B. In category A trials, 280 noise features were created by sampling from a normal distribution with mean 0 and standard deviation 1, while 20 signal features were created by sampling with a mean of 0.2, 0.4, 0.6 or 0.8 depending on the data set (low or high SNR) and a standard deviation of 1. Category B trials were created similarly, but 10 of the signal features overlapped with 10 signal features of category A. Thus, both category A and B had 10 category-specific features, and 10 overlapping ones. There were therefore 20 features (out of 300) relevant for classification, since the overlapping signal features carried no discriminatory information. The goal of our feature selection methods was to extract the relevant non-overlapping features.

Haxby Data Set

We used the Haxby data set (Haxby et al. 2001) as a benchmark for comparison, as it has been repeatedly reanalyzed for evaluation and comparison of machine learning approaches (Chou et al. 2014; Do et al. 2015; Fan and Chou 2016; Kuncheva et al. 2010; O'Toole et al. 2005). The Haxby study examined face and object representations in human ventral temporal cortex in a blocked-design fMRI study. A total of 6 subjects are included in the dataset, and each subject experienced 12 experimental runs (24 s each). During each run, participants passively viewed 8 images of 8 different objects (e.g. faces, houses, scissors, bottles, etc.). The images were shown for 500 ms and then followed by a 1500 ms inter-stimulus interval. The experiment thus had a total of 12 × 8 = 96 samples from each individual, except for subject 5, where one of the runs was corrupted and not used in the analysis. The same ventral temporal mask was applied as was used in the original paper. The data set underwent standard preprocessing for MVPA: motion correction, linear detrending and z-scoring.

Zeithamova Data Set

We also used a more recent data set to compare our feature selection methods on (Zeithamova et al. 2017). This data was obtained from the OpenfMRI database under accession number ds000238 and was originally used to investigate experimental design optimization by looking at the effect of various trial-timings on decoding accuracy. Designs varied in the number of trials and onset-to-onset ranging from slow 12 s trials with two repetitions of each item to quick 6 s trials with four repetitions per item. A trial was composed of viewing a 2 s image of either an animal or a tool and participants were instructed to encode these images for a subsequent recognition task. For their analyses, the researchers had pre-defined ROI's in which they used F-test feature selection to select 100 voxels as input for an SVM classification. They found equal performance across all timing conditions for category decoding, while item-level information was better detected using slow trial timings.

For computational purposes, we only used the first 15 participants and the medium (8 s) to slow (12 s) onset-to-onset trial timings. We also limited the analyses to a pre-defined region of visual cortex for which Zeithamova et al. found optimal decoding accuracy. Furthermore, because of the higher resolution of the data here (2 × 2 × 2 voxels) and the resulting higher amount of voxels, we found it unfeasible to perform RFE-SVM due to time constraints. The same standard preprocessing steps as for the Haxby data were applied and because image acquisition was time-locked to stimulus presentation, we took the average of the third and fourth TR after stimulus presentation, corresponding to peak HRF as our event of interest.

General Procedure

All analyses were performed using python version 2.7.6 with Nilearn (for Haxby data set), scikit-learn (for out-of-the-box RFE-SVM) (Abraham et al. 2014) and custom code on a server running Linux Ubuntu 14.04 LTS (Kernel version 3.13.0 with 10 × 16 GB RDIMM and 2 x Intel Xeon E5–2620 processors).

All feature selection methods were embedded in the same general procedure for an apples-to-apples comparison. First the data was divided in 3 subsets. These subsets are then assigned to be a training, validation or test set according to a cross-validation scheme. This is replicated 6 times in total, so

that each of the 3 subsets is chosen exactly once as training, test or validation set.

The procedure can be summarized as follows:

1. Randomly take a subset (90% of samples) out of training set.
2. Run the feature selection method on this subset. Save weights for neural network, save rankings for RFE-SVM.
3. Repeat steps 2 and 3 20 times. At the end, use the summed weights (NN, ReliefF), rankings (RFE-SVM), F-values (F-test) or mutual information values (MI) to sort the features in order of importance.
4. For the simulation data set, select the 20 most important features, for the Zeithamova data set, select the 100 most important features. If on the Haxby data set, use a forward selection scheme, starting with the most important feature, each time classifying on the validation set. Take all features from the step where classification accuracy is at the median.
5. Train an SVM with the selected features on the training + validation set.
6. Evaluate the performance on the test set.
7. Replicate steps 1–6 6 times, mixing up the cross-validation scheme so that each set is used once for training, once for validation and once for testing.
8. Measure the pairwise Jaccard index for each selected feature set of each replication and average to get an overall stability metric.

Metrics

Accuracy

Accuracy for all methods was assessed after feature selection using an SVM, trained on our training and validation set, for classification; and accuracy was evaluated on a test set. This was replicated six times, so that each set was a training, validation or test set, and the average across replications and its standard deviation were reported.

Stability

Stability refers to the ability of feature selection methods to select the same features from the data across replications. A methodological framework for stability selection, incorporating resampling, was developed and validated by Meinshausen and Bühlmann (2010). It entails repeated re-sampling of subsets of the original data and performing feature selection on each of them. Features with a frequency of being selected higher than a user-defined threshold are then selected for inclusion in a final set of features. This results in a more stable set of features than would be obtained by a single instance of feature selection. The procedure can be used in conjunction with any feature selection procedure and classifying algorithm.

We used the Jaccard index as a measure of the stability of the feature selection across replications. This metric is commonly used to compare the similarity or diversity of finite sample sets. It is defined as the size of the intersection of two sets, divided by their union and so always has values between 0 and 1.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{7}$$

Because the Jaccard index correlates positively with the proportion of features selected, we established a baseline against which our results were compared. Specifically, we computed the expected Jaccard index by comparing 1000 random sets of features. The size of these sets was taken from a truncated normal distribution (ranging between 1: max amount of features) with mean equal to the average subset size for the specific subject and method and standard deviation equal to the standard deviation of the subset size. The Jaccard index was computed for all pairwise combinations of random subsets and the average taken as our estimate of what a Jaccard index under random feature selection would constitute.

Time

Finally, as noted above, some feature selections can be computationally intensive. We therefore also measured the time needed for the method to run. Computation time can be relevant when dealing with particularly large data sets, such as fMRI, or when embedding the feature selection method in an iterative procedure as is the case here. Three of the methods are wrapper approaches, which are generally not known for their high computational efficiency, but we expect RFE-SVM especially to be slower because of the backward selection procedure.

Data

Haxby Data Set

For the Haxby data, a subset size selection procedure was necessary to determine the optimum number of features (Table 3). After feature selection, features are ordered by the 'importance' of each feature, and we used a forward selection scheme which incrementally adds the next most 'important' feature on each iteration after which classification performance is assessed. For each of these feature sets, an SVM was trained on the training set and performance assessed on the validation set. We then chose the subset size for which accuracy was at the median.

Table 3: Pseudocode outline for the general procedure on the Haxby data.

```
Algorithm General procedure: Haxby data
for J = 1:6 replications do
    CV - assign training, validation, test set
    run feature selection algorithm
    for for K = 1:total number of features do
        subset_voxels = sorted_importance[1:K]
        train SVM on training set, using subset_voxels
        subset_acc[K] = trained SVM on validation set
    end for
    subset = sorted_importance[1:L] if subset_acc[L] ==
median(subset_acc)
    train SVM on train and validation set, using subset
    predict_acc = SVM accuracy on test set
end for
calculate mean accuracy and sd across replications
calculate all pairwise jaccard index
return mean_acc, sd, subset_size, mean_jaccard_index
```

Simulation Data Set

The procedure for the simulation data was similar to that of the Haxby data (Table 4), the only difference being that there is no subset size selection here. Since the number of discriminative features is known for our simulated data, we selected the 20 most 'important' features as determined by the feature selection procedure as the final feature set for testing.

Zeithamova Data Set

For this data set, our procedure was nearly identical to that of the simulation data set (Table 4), except we chose the 100 most relevant features at each step which corresponds to the number of features selected in the original paper. This also allows easier comparison of the stability metric as it will not be confounded with feature subset size.

Results

Simulation

An overview of the results can be seen in Table 5. Mean accuracy overall is highest for the F-test and our NN, and lowest for importance mapping (NN-Polyn) and mutual information (MI). Since the only difference between the NN and importance mapping method is the average voxel activity multiplier, this seems to suggest a problem with this approach. Stability measures reveal an even larger difference between the methods. Note that two stability measures were assessed here: one is the similarity between selected feature sets across replications; the other is the similarity of the selected feature sets with the truly important features (that we know a priori) (Figs. 1 and 2). In our comparisons based on simulated data, there is no variation in the number of features selected each time; during selection, the 20 best features were always

Table 4: Pseudocode outline for the general procedure on the simulation data.

```
Algorithm General procedure: Simulation data
for J = 1:6 replications do
    CV - assign training, validation, test set
    run feature selection algorithm
    subset = sorted_importance[1:20]
    train SVM on train and validation set, using subset
    predict_acc = SVM accuracy on test set
end for
calculate mean accuracy and sd across replications
calculate all pairwise jaccard index
return mean_acc, sd, mean_jaccard_index
```

chosen, corresponding to the number of (actual) relevant features in our simulated data. Fixing the number of features in this manner allows for cleaner comparisons since the number of selected features is correlated with the similarity index (see below). The Jaccard index across replications is consistently higher for the NN and importance mapping than for the other methods (Table 5). This stability across replications scales well with the signal of the data for the NN, RFE-SVM, F-test and ReliefF, while for importance mapping and MI it is very high no matter the signal-to-noise ratio (SNR) and does not increase much with increasing SNR. Furthermore, when we look at the similarity of the selected feature sets with the truly important features in our simulated data, we see much worse results for importance mapping and MI compared to the other techniques. This suggests that, although importance mapping and MI are stable, they consistently pick up irrelevant features from this data set. Additionally, the Jaccard index with the set of truly relevant voxels is highest for the NN and F-test indicating that these are better at finding the correct features. This is also reflected in the standard deviation of the

Table 5 Results on the simulated data sets. Mean accuracy without feature selection is the same for all methods, since the procedure is the same; it is the classification accuracy on the test set using an SVM, without any feature selection

Method	SNR of the relevant features	Jaccard index across replications	Jaccard index with the set of relevant voxels	Mean accuracy (%)	STD (%)	Mean accuracy without feature selection (%)
NN	0.00	0.76	No relevant voxels	50.17	3.40	50.63
NN-Polyn		0.87		49.98	3.37	
RFE-S-VM		0.51		51.03	3.75	
MI		0.78		51.51	3.19	
F-test		0.41		50.10	3.78	
ReliefF		0.75		50.18	4.03	
NN	0.20	0.75	0.16	56.27	3.61	55.10
NN-Polyn		0.86	0.11	55.49	3.97	
RFE-S-VM		0.53	0.13	55.48	4.12	
MI		0.78	0.06	54.51	3.79	
F-test		0.47	0.18	57.21	4.12	
ReliefF		0.75	0.06	53.03	4.41	
NN	0.40	0.84	0.52	73.68	3.16	67.67
NN-Polyn		0.86	0.21	67.42	4.40	
RFE-S-VM		0.59	0.30	68.52	4.36	
MI		0.78	0.09	60.61	5.31	
F-test		0.66	0.49	75.92	2.80	
ReliefF		0.80	0.20	67.43	5.46	
NN	0.60	0.94	0.84	88.68	2.54	81.10
NN-Polyn		0.89	0.32	80.08	2.87	
RFE-S-VM		0.69	0.48	83.36	2.44	
MI		0.80	0.21	76.67	4.15	
F-test		0.98	0.84	89.47	1.76	
ReliefF		0.87	0.49	85.27	2.98	
NN	0.80	0.99	1.00	95.11	1.90	89.90
NN-Polyn		0.92	0.36	86.95	2.38	
RFE-S-VM		0.74	0.54	91.21	2.31	
MI		0.84	0.42	90.16	2.69	
F-test		0.96	0.97	95.26	1.20	
ReliefF		0.92	0.76	94.43	1.63	

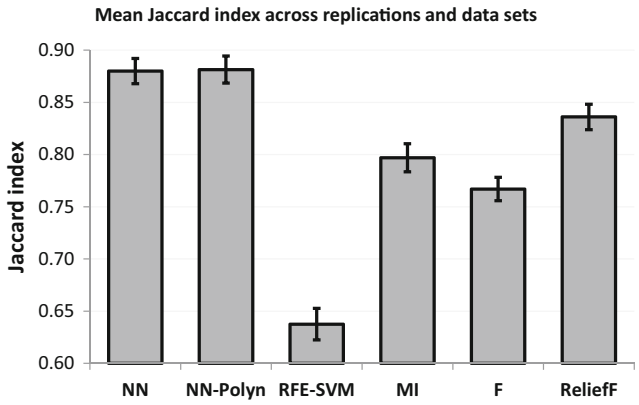

Fig. 1 Average Jaccard index computed from looking at the pairwise similarities between the sets of selected features for each replication collapsed over all strictly positive SNR data sets. Error bars represent the average of the standard errors of the mean for the strictly positive SNR data sets

mean accuracy, which is usually lower for the NN and F-test. While the NN and F-test perform better than the RFE-SVM, MI and ReliefF in both our measures of performance (accuracy and stability), it should be noted that both feature selection methods lead to an increase in accuracy compared to no feature selection. In contrast, feature selection using importance mapping or MI show no consistent improvement over classification without feature selection. Comparing algorithm run time also shows a large difference between methods. As expected, the RFE-SVM took considerably longer than the NN and importance mapping. Specifically, it took on average 54 h for the RFE-SVM to run on 1 of the simulated data sets, while the NN, ReliefF, F-test and importance mapping took no longer than 3 min for the same data. MI also took longer, clocking in at 43 min on average, but this did not result in increased performance.

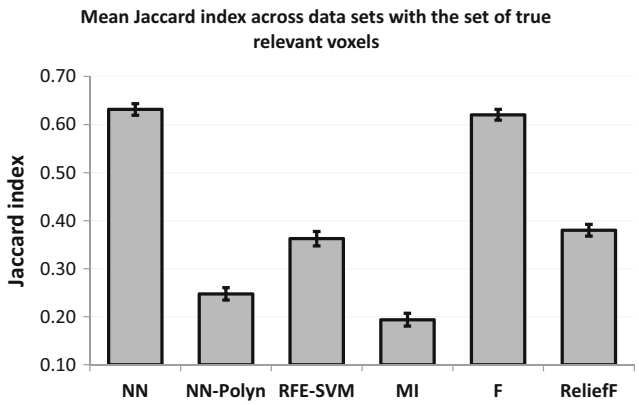

Fig. 2 Average Jaccard index across data sets with strictly positive SNR computed from looking at the similarities between the sets of selected features for each replication and the set of truly relevant voxels. Error bars represent the average of the standard errors of the mean for the positive SNR data sets

Haxby Data

While our results using simulated data are promising, a critical issue is how well they translate to real fMRI data. We tested the RFE-SVM, F-test, MI, ReliefF and NN methods using the Haxby data set (described above) and observed very high stability for ReliefF and comparably high decoding accuracy for the F-test, NN and MI. Unlike the simulation data, we did not fix the number of selected features here since the true number of relevant voxels is not known a priori. Instead, we used a subset size selection procedure (see Methods) to determine the number of selected voxels. Even though accuracy for the NN was comparable or higher than that of the other methods, the number of voxels extracted was lower for most subjects, indicating that the NN was more likely to rank more informative features higher than the other methods (Table 6). In contrast to the simulated data set, feature selection did not lead to an improvement in classification accuracy compared to no feature selection. A possible explanation for this may be that most of the ventral temporal cortex ROI used here is highly informative; nearly all features contain classifying information in varying degrees of importance, and thus additional features always provide more information relevant to classification (Chu et al. 2012).

Because we had an average and standard deviation for the number of selected features on each iteration across replications, we were able to estimate an expected Jaccard index for this subset size if features were selected randomly. This was calculated by repeated random sampling of features and averaging all pairwise Jaccard indexes, with the subset size taken each time by a random value of a normal distribution with mean equal to the empirical subset size and standard deviation equal to the standard deviation of the subset size across iterations. We observed consistently higher stability across replications for the NN and F-test compared to RFE-SVM and MI. Interestingly, stability for ReliefF was exceedingly high here for all subjects, but this was also accompanied by the lowest decoding accuracy while using the most voxels. Unlike the simulation, varying numbers of selected features were possible here due to the subset selection procedure. But even though a larger number of selected features correlates positively with the Jaccard index (as can be seen in the expected Jaccard index column, Table 6), the NN overall has high stability and accuracy despite a lower number of selected features for most subjects. The difference between the expected Jaccard index by chance and the measured Jaccard index is also substantially larger for the NN and F-test compared to MI and RFE-SVM, but not compared to ReliefF (Table 6).

Zeithamova Data

Results from the Zeithamova data set show a slightly different picture. Here, the NN does not perform as well as the ReliefF

Table 6 Results from the methods on the 6 subjects of the Haxby data set

Subject	Method	Average number of selected features	STD of number of selected features	Jaccard index	Expected Jaccard Index for # of selected features	Mean accuracy (%)	STD (%)	Mean accuracy without feature selection (%)
1	NN	175/577	44.95	0.55	0.17	88.63	1.62	93.63
	RFE-S-VM	227/577	46.98	0.35	0.24	88.08	4.22	
	MI	220/577	36.04	0.42	0.23	89.00	1.26	
	F-test	235/577	18.92	0.59	0.25	88.19	0.90	
	ReliefF	255/577	13.66	0.86	0.28	89.76	0.77	
2	NN	196/464	29.24	0.61	0.26	77.78	2.07	85.24
	RFE-S-VM	206/464	23.67	0.36	0.28	78.94	3.12	
	MI	213/464	11.26	0.51	0.30	80.38	1.71	
	F-test	213/464	18.66	0.57	0.30	79.51	1.64	
	ReliefF	201/464	19.73	0.81	0.28	77.20	3.15	
3	NN	142/307	8.16	0.56	0.30	69.62	2.47	77.08
	RFE-S-VM	129/307	16.04	0.33	0.26	66.32	1.55	
	MI	132/307	16.67	0.47	0.27	67.82	2.74	
	F-test	132/307	13.39	0.65	0.27	67.01	3.00	
	ReliefF	124/307	16.25	0.75	0.28	64.24	3.91	
4	NN	214/675	37.25	0.55	0.19	76.13	1.82	83.91
	RFE-S-VM	242/675	42.48	0.34	0.22	72.51	2.10	
	MI	245/675	44.36	0.39	0.22	75.98	4.24	
	F-test	280/675	42.21	0.53	0.26	77.20	2.72	
	ReliefF	285/675	32.54	0.80	0.27	75.93	2.14	
5	NN	183/422	27.98	0.58	0.27	79.39	1.83	86.36
	RFE-S-VM	183/422	27.24	0.37	0.27	76.64	3.36	
	MI	185/422	16.56	0.44	0.28	80.18	2.37	
	F-test	198/422	9.06	0.53	0.31	79.67	2.95	
	ReliefF	191/422	12.28	0.87	0.29	78.79	2.09	
6	NN	128/348	24.52	0.50	0.18	72.51	3.90	80.90
	RFE-S-VM	140/348	27.05	0.32	0.24	70.43	2.12	
	MI	148/348	23.11	0.50	0.27	72.80	4.66	
	F-test	126/348	19.03	0.65	0.22	72.97	1.87	
	ReliefF	169/348	6.05	0.91	0.29	73.09	3.43	
AV-RG.	NN	173/465	28.68	0.56	0.24	77.34	3.12	84.52
	RFE-S-VM	188/465	30.57	0.34	0.25	75.49	2.74	
	MI	191/465	24.67	0.46	0.26	77.70	2.83	
	F-test	197/465	20.21	0.59	0.27	77.43	2.18	
	ReliefF	204/465	16.75	0.83	0.28	76.50	2.58	

or F-test when it comes to stability, however it does have the highest decoding accuracy, suggesting that in this case it might be picking up on interchangeable sets of very relevant voxels. Note that, for simplicity and ease of comparison, no subset size selection procedure was done. Each feature selection method only had 100 voxels to work with. Interestingly, again

we find MI has the lowest stability and decoding accuracy of all the methods tested here, which is similar to its performance on the simulation data. We also note that decoding accuracy after feature selection is worse than no feature selection. This is similar to what we found for the Haxby data set and might again be due to the use of a visual cortex ROI, where most features contain some relevant information in varying degrees (Table 7).

Violating Homoscedasticity

Across the various data sets we tested, we noticed a very comparable performance for the F-test and our NN. We decided to test if this similarity holds for cases where we violate an assumption of the F-test, namely that of homogeneity of variance. We did this by tripling the standard deviation (thus decreasing SNR) of relevant voxels for one of the classes. An overview of the results can be seen in Table 8. We found no large differences in decoding accuracy between the two methods, though the NN is consistently slightly better. However, there is a large difference now between the methods in favor of the NN when it comes to stability (Jaccard index across replications) and the detection of the set of true voxels (Jaccard index with the set of relevant voxels). The NN thus seems to be less affected by violations of this assumption and is able to better detect the truly relevant voxels, and this might be hard to tell by merely looking at the decoding accuracy.

Discussion

In this study, we evaluated a novel method of feature selection based on a single-layer neural network which incorporates cross-validation during feature selection and stability selection through iterative subsampling. This method was compared on simulated data sets, a recent data set from the open fMRI project and the Haxby data set, which has been used extensively for benchmarking fMRI and MVPA analyses. For comparison, we focused on several widely used and tested feature selection methods such as importance mapping, RFE-SVM, Mutual information, F-test and ReliefF.

Using the simulated data set we looked at classification accuracy, time, stability and correct detection of relevant voxels for the 3 methods. In general, we found superior performance on all 3 measures for the NN, achieving higher accuracy, stability, and in less time than the alternative methods. Notably, stability, defined as the similarity of selected feature subsets when the training data is perturbed, was lowest for RFE-SVM, and similar for importance mapping and our NN.

While the stability of importance mapping was comparable to the NN, this does not necessarily indicate that the features identified by importance mapping were informative, as can be seen in the accuracy score and feature subset correspondence with the set of truly informative voxels. Important to note is that the algorithm used in importance mapping is exactly the same as the one used for the NN, barring the multiplier to the weights. Thus, any differences in classifier accuracy following feature selection, stability or detection of relevant voxels should be due to this multiplier. Though importance mapping may work well in some situations, it may fail for situations in which voxels have similar (increased) activation for multiple categories being classified due to the multiplier causing irrelevant voxels to be selected based primarily on their activity. The same is true for mutual information, which shows decent stability, but has very poor detection of true relevant voxels, resulting in lower accuracies than using no feature selection. In contrast to these, RFE-SVM suffers from much lower stability, but truly relevant voxels have a slightly higher chance of being identified, reflected by accuracy scores.

Our feature selection methods, applied to real data from the Haxby data set, largely replicate our results using simulated data. Here, we again find the lowest stability for RFE-SVM, with Jaccard similarity coefficients closer to what is expected by chance. A similar finding of poor stability for RFE-SVM has been reported many times in the literature (Dittman et al. 2011; Haury et al. 2011; A. Kalousis et al. 2005; Stiglic and Kokol 2010), where they compared RFE-SVM with other feature selection methods. They suggest that this low stability is due to multiple iterations of eliminating features and reassigning weights to remaining features. Different subsets of samples can result in different feature rankings and elimination of different features at a given step in the selection procedure. Hence, multiple iterations can cause higher instability of the top k selected features.

Besides its low stability, RFE-SVM is computationally expensive due to its iterative procedure. The implementation of RFE-SVM used in this study is part of the Scikit-learn toolbox

Table 7 Averaged results from the methods on the 15 subjects of the Zeithamova data set

Method	Mean Jaccard index	STD Mean Jaccard index	Mean accuracy (%)	Mean of STD (%)	Mean accuracy without feature selection (%)
NN	0.16	0.04	74.27	4.93	79.48
MI	0.15	0.01	67.20	6.04	
F-test	0.21	0.03	72.50	5.45	
ReliefF	0.20	0.03	73.51	4.97	

Table 8 Results of simulation data for NN and F-test after violating the assumption of homoscedasticity

Method	SNR of the relevant features for category 1 (SNR of the relevant features for category 2)	Jaccard index across replications	Jaccard index with the set of relevant voxels	Mean accuracy (%)	STD (%)	Mean accuracy without feature selection (%)
NN	0.00 (0.00)	0.72	No relevant voxels	51.41	3.52	49.47
F-test		0.46		50.64	3.60	
NN	0.20 (0.07)	0.74	0.13	54.46	3.25	53.23
F-test		0.49	0.10	52.78	3.83	
NN	0.40 (0.13)	0.81	0.39	67.09	3.31	61.23
F-test		0.58	0.30	67.03	3.73	
NN	0.60 (0.20)	0.91	0.65	81.28	2.60	71.30
F-test		0.90	0.49	79.74	3.18	
NN	0.80 (0.27)	0.93	0.80	88.31	2.09	80.97
F-test		0.74	0.61	87.91	2.29	

(Pedregosa et al. 2011), and was not optimized for speed or efficiency. Various adjustments to the out-of-the-box.

RFE-SVM routine, e.g. a simple adjustment of the number or percentage of features eliminated at each step of the backward selection may be able to improve efficiency, albeit at a potential cost of precise feature rankings. Similarly, the NN algorithm described here was not optimized for either the simulated or real data sets, e.g. by selection of optimal meta-parameters or training times. Nevertheless, despite the lack of optimization of our NN method, overall it outperformed widely-used feature selection approaches on our metrics.

Surprisingly, ReliefF showed very high stability on the real data, greater than that of the NN, contrary to what we found for the simulation data. However, it seems to be doing less with more since it selects on average the largest voxel subset of all methods, yet has worse accuracy than our NN, which selects the smallest subset. This suggests that ReliefF is not selecting the most relevant voxels, but a set of voxels that have high correspondence across replications. Mutual information on the other hand shows poor stability but high accuracy for the Haxby data. On the Zeithamova data, meanwhile, it performs considerably worse in terms of accuracy and also has the lowest stability, similar results as for the simulated data set.

Finally, the F-test showed similar results as the NN on both the real data sets and the simulated data, showing the best overall performance in terms of accuracy, speed and correct detection of relevant features, while having very good stability.

While they seem comparable here, there are two reasons why one might prefer the NN over the F-test. First, we showed that the NN is more robust to violations of homoscedasticity, an underlying assumption of the F-test. Second, the F-test is a very inflexible method compared to the NN, which can be more easily tweaked and optimized (e.g. through selection of appropriate hyper-parameters) to better accommodate the data. However, due to its all-around good performance in terms of decoding accuracy, speed, stability and also its ease of implementation, the F-test is a viable alternative to the NN for feature selection with fMRI data, provided that the underlying assumptions of the F-test are satisfied. For both real data sets, we found lower classification accuracy after feature selection for all techniques used here. This is likely due to the nature of the data and the methods used (Chu et al. 2012; Kerr et al. 2014). Since the use of the ventral temporal and visual cortex masks are already a form of knowledge-driven feature selection, most voxels within these ROI are likely diagnostic about category labels. Additionally, due to the nature of the task (identification of visual information), it is also likely that many of these features are redundant. Since none of the methods described here remove redundant features as part of their algorithm, these redundant features that are nevertheless highly informative, will be retained; while features providing less diagnostic, but unique information, will be excluded (Kerr et al. 2014). Additionally, redundant features can cause unstable rankings, which can also explain the poor stability of RFE-SVM (Toloşi and Lengauer 2011). Therefore, it is likely that RFE-SVM is not well suited to imaging data without controlling for redundancy first.

Many feature selection techniques strip redundant features in an attempt to obtain an optimal feature subset which has minimal cardinality while maximizing performance (Vergara and Estévez 2014). These techniques usually boast high accuracy scores on data sets with many correlated features, as is the case with neuroimaging data due to spatial feature correlations (i.e. nearby voxels showing similar activation patterns). However, if the goal is to obtain a set of interpretable voxels possessing discriminative information pertaining to the categories, redundant variables should not be discarded (Wang et al. 2015). Methods such as the ones used in this paper do not strip redundant features, so they serve this goal well. Similar to importance mapping, our NN can be used to determine which voxels are highly relevant for classification, but unlike importance mapping, our NN approach will not select irrelevant (non-discriminative) features due to shared BOLD-

activation between categories. And since a prerequisite of an interpretable set of voxels is that it is stable, i.e. invariant to perturbations in the training data, the NN is more fitting to use than RFE-SVM.

Neural networks are rarely used for feature selection in imaging data, likely due to the popularity and availability of feature selection methods such as RFE-SVM, F-test and other out-of-the-box methods. The neural network implementation used here has a very basic architecture, yet merely adding stability selection through subsampling and cross-validation during learning can yield better performance than the more popular methods shown here. Utilizing the inherent flexibility of neural networks, future research could pair stability selection with various more complex and powerful neural network implementations to determine what parameter settings yield optimal stability and detection of relevant voxels. Furthermore, the use of simulations in exploring and comparing feature selection techniques is underappreciated. Data sets with varying properties (e.g. feature amount, redundancy, relevance, noise, non-linearity, number of samples) that resemble BOLD-activation patterns can be tested and benchmarked on to inform researchers of which feature selection methods and parameters are most appropriate for their data and research question.

Conclusion

In this study, we evaluated a novel method of feature selection based on a single-layer neural network which incorporates cross-validation during feature selection and stability selection through iterative subsampling. This method was compared to several popular feature selection techniques, including importance mapping, a technique used to determine which voxels contribute meaningfully to classification. On a simulated data set, we found superior performance of our method on computational time, accuracy, stability and detection of truly relevant voxels compared to the alternative methods. Importantly, we found that importance mapping consistently selects irrelevant voxels, leading to poor accuracy. We also found very poor stability and less accurate classification on benchmark data for RFE-SVM compared to the NN. Future research can explore how best to optimize neural networks for a stable detection of relevant features in various data settings.

Information Sharing Statement

The data sets used in this article are freely available online. The Haxby data set was made available under the terms of the Creative Commons Attribution-Share Alike 3.0 license and separate tarballs for each subject can be found at http://data.pymvpa.org/datasets/haxby2001. The Zeithamova data was obtained from the OpenfMRI database (RRID:SCR_005031). Its accession number is ds000238 and it was made available under the ODC Public Domain Dedication and License. Code for generating the simulated data can be found at the public GitHub repository (https://github.com/deraevejames/data-generation_FS). RFE-SVM, mutual information and F-test methods were taken from the scikit-learn library for Python (RRID:SCR_002577), while ReliefF code was used from scikit-feature, an open-source feature selection repository for Python (http://featureselection.asu.edu/, RRID:SCR_016141).

Acknowledgments This research was supported by FWO-Flanders Odysseus II Award #G.OC44.13 N to WHA.

Compliance with Ethical Standards

Conflict of Interest We report no conflicts of interest.

References

Abraham, A., Pedregosa, F., Eickenberg, M., Gervais, P., Muller, A., Kossaifi, J., … Varoquaux, G. (2014). Machine learning for neuroimaging with Scikit-learn. *arXiv:1412.3919 [Cs,* Stat*]*. Retrieved from http://arxiv.org/abs/1412.3919

Bolón-Canedo, V., Sánchez-Maroño, N., & Alonso-Betanzos, A. (2013). A review of feature selection methods on synthetic data. *Knowledge and Information Systems, 34*(3), 483–519. https://doi.org/10.1007/s10115-012-0487-8.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on computational learning theory* (pp. 144–152). New York: ACM. https://doi.org/10.1145/130385.130401.

Cao, L. J., & Chong, W. K. (2002). Feature extraction in support vector machine: a comparison of PCA, XPCA and ICA. In *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP '02* (Vol. 2, pp. 1001–1005 vol. 2). https://doi.org/10.1109/ICONIP.2002.1198211.

Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering, 40*(1), 16–28. https://doi.org/10.1016/j.compeleceng.2013.11.024.

Chou, C. A., Kampa, K., Mehta, S. H., Tungaraza, R. F., Chaovalitwongse, W. A., & Grabowski, T. J. (2014). Voxel selection framework in multi-voxel pattern analysis of fMRI data for prediction of neural response to visual stimuli. *IEEE Transactions on Medical Imaging, 33*(4), 925–934. https://doi.org/10.1109/TMI.2014.2298856.

Chu, C., Hsu, A.-L., Chou, K.-H., Bandettini, P., & Lin, C. (2012). Does feature selection improve classification accuracy? Impact of sample size and feature selection on classification using anatomical magnetic resonance images. *NeuroImage, 60*(1), 59–70. https://doi.org/10.1016/j.neuroimage.2011.11.066.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*(3), 273–297. https://doi.org/10.1007/BF00994018.

Cox, D. D., & Savoy, R. L. (2003). Functional magnetic resonance imaging (fMRI) "brain reading": Detecting and classifying distributed patterns of fMRI activity in human visual cortex. *NeuroImage, 19*(2), 261–270. https://doi.org/10.1016/S1053-8119(03)00049-1.

Das, S. (2001). Filters, wrappers and a boosting-based hybrid for feature selection. In *Proceedings of the eighteenth international conference*

on machine learning (pp. 74–81). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. Retrieved from http://dl.acm.org/citation.cfm?id=645530.658297.

De Martino, F., Valente, G., Staeren, N., Ashburner, J., Goebel, R., & Formisano, E. (2008). Combining multivariate voxel selection and support vector machines for mapping and classification of fMRI spatial patterns. *NeuroImage, 43*(1), 44–58. https://doi.org/10.1016/j.neuroimage.2008.06.037.

Dernoncourt, D., Hanczar, B., & Zucker, J.-D. (2014). Analysis of feature selection stability on high dimension and small sample data. *Computational Statistics & Data Analysis, 71*, 681–693. https://doi.org/10.1016/j.csda.2013.07.012.

Ding, C., & Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. *Journal of Bioinformatics and Computational Biology, 3*(2), 185–205. https://doi.org/10.1142/S0219720005001004.

Dittman, D., Khoshgoftaar, T. M., Wald, R., & Wang, H. (2011). Stability Analysis of Feature Ranking Techniques on Biological Datasets. In *2011 I.E. International Conference on Bioinformatics and Biomedicine* (pp. 252–256). https://doi.org/10.1109/BIBM.2011.84.

Do, L.-N., Yang, H.-J., Kim, S.-H., Lee, G.-S., & Kim, S.-H. (2015). A multi-voxel-activity-based feature selection method for human cognitive states classification by functional magnetic resonance imaging data. *Cluster Computing, 18*(1), 199–208. https://doi.org/10.1007/s10586-014-0369-9.

Fan, M., & Chou, C.-A. (2016). Exploring stability-based voxel selection methods in MVPA using cognitive neuroimaging data: A comprehensive study. *Brain Informatics, 3*(3), 193–203. https://doi.org/10.1007/s40708-016-0048-0.

Fleuret, F. (2004). Fast binary feature selection with conditional mutual information. Journal of Machine Learning Research, 5(Nov), 1531–1555.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. Journal of Machine Learning Research, 3(Mar), 1157–1182.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for Cancer classification using support vector machines. *Machine Learning, 46*(1–3), 389–422. https://doi.org/10.1023/A:1012487302797.

Hall, M. A. (1998). Correlation-based feature selection for machine learning.

Haury, A.-C., Gestraud, P., & Vert, J.-P. (2011). The influence of feature selection methods on accuracy, stability and interpretability of molecular signatures. *PLoS One, 6*(12), e28210. https://doi.org/10.1371/journal.pone.0028210.

Haxby, J. V., Gobbini, M. I., Furey, M. L., Ishai, A., Schouten, J. L., & Pietrini, P. (2001). Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science, 293*(5539), 2425–2430. https://doi.org/10.1126/science.1063736.

Hebart, M. N., Görgen, K., & Haynes, J.-D. (2015). The decoding toolbox (TDT): A versatile software package for multivariate analyses of functional imaging data. Frontiers in Neuroinformatics, 8. https://doi.org/10.3389/fninf.2014.00088.

Johnson, J. D., McDuff, S. G. R., Rugg, M. D., & Norman, K. A. (2009). Recollection, familiarity, and cortical reinstatement: A multi-voxel pattern analysis. *Neuron, 63*(5), 697–708. https://doi.org/10.1016/j.neuron.2009.08.011.

Kalousis, A., Prados, J., & Hilario, M. (2005). Stability of feature selection algorithms. In *Fifth IEEE International Conference on Data Mining (ICDM'05)* (p. 8 pp.-). https://doi.org/10.1109/ICDM.2005.135.

Kalousis, A., Prados, J., & Hilario, M. (2007). Stability of feature selection algorithms: A study on high-dimensional spaces. *Knowledge and Information Systems, 12*(1), 95–116. https://doi.org/10.1007/s10115-006-0040-8.

Kerr, W. T., Douglas, P. K., Anderson, A., & Cohen, M. S. (2014). The utility of data-driven feature selection: Re: Chu et al. 2012. *NeuroImage, 84*, 1107–1110. https://doi.org/10.1016/j.neuroimage.2013.07.050.

Kirk, P., Witkover, A., Bangham, C. R. M., Richardson, S., Lewin, A. M., & Stumpf, M. P. H. (2013). Balancing the robustness and predictive performance of biomarkers. *Journal of Comparative Biology, 20*(12), 979–989. https://doi.org/10.1089/cmb.2013.0018.

Kononenko, I., & Simec, E. (1995). Induction of decision trees using Relieff. In *Proceedings of the ISSEK94 workshop on mathematical and statistical methods in artificial intelligence* (pp. 199–220). Springer, Vienna. https://doi.org/10.1007/978-3-7091-2690-5_14.

Kononenko, I., Šimec, E., & Robnik-Šikonja, M. (1997). Overcoming the myopia of inductive learning algorithms with RELIEFF. *Applied Intelligence, 7*(1), 39–55. https://doi.org/10.1023/A:1008280620621.

Křížek, P., Kittler, J., & Hlaváč, V. (2007). Improving stability of feature selection methods. In Computer Analysis of Images and Patterns (pp. 929–936). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-74272-2_115, Improving Stability of Feature Selection Methods.

Kuncheva, L. I., Rodriguez, J. J., Plumpton, C. O., Linden, D. E. J., & Johnston, S. J. (2010). Random subspace ensembles for fMRI classification. *IEEE Transactions on Medical Imaging, 29*(2), 531–542. https://doi.org/10.1109/TMI.2009.2037756.

Lewis-Peacock, J. A., Drysdale, A. T., Oberauer, K., & Postle, B. R. (2011). Neural evidence for a distinction between short-term memory and the focus of attention. *Journal of Cognitive Neuroscience, 24*(1), 61–79. https://doi.org/10.1162/jocn_a_00140.

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. (2017). Feature Selection: A Data Perspective. *ACM Computing. Surveys*, *50*(6), 94:1–94:45. :https://doi.org/10.1145/3136625.

Liu, H., & Setiono, R. (1995). Chi2: feature selection and discretization of numeric attributes. In *Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence* (pp. 388–391). https://doi.org/10.1109/TAI.1995.479783.

Ma, S., & Huang, J. (2008). Penalized feature selection and classification in bioinformatics. *Briefings in Bioinformatics, 9*(5), 392–403. https://doi.org/10.1093/bib/bbn027.

Mahmoudi, A., Takerkart, S., Regragui, F., Boussaoud, D., & Brovelli, A. (2012). Multivoxel pattern analysis for fMRI data: A review. Computational and Mathematical Methods in Medicine, 2012, e961257. https://doi.org/10.1155/2012/961257.

McDuff, S. G. R., Frankel, H. C., & Norman, K. A. (2009). Multivoxel pattern analysis reveals increased memory targeting and reduced use of retrieved details during single-agenda source monitoring. *Journal of Neuroscience, 29*(2), 508–516. https://doi.org/10.1523/JNEUROSCI.3587-08.2009.

Meinshausen, N., & Bühlmann, P. (2010). Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 72*(4), 417–473. https://doi.org/10.1111/j.1467-9868.2010.00740.x.

Michel, V., Damon, C., & Thirion, B. (2008). Mutual information-based feature selection enhances fMRI brain activity classification. In *2008 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro* (pp. 592–595). https://doi.org/10.1109/ISBI.2008.4541065.

Mwangi, B., Tian, T. S., & Soares, J. C. (2014). A review of feature reduction techniques in neuroimaging. *Neuroinformatics, 12*(2), 229–244. https://doi.org/10.1007/s12021-013-9204-3.

Nie, F., Xiang, S., Jia, Y., Zhang, C., & Yan, S. (2008). Trace ratio criterion for feature selection. In *In AAAI* (pp. 671–676).

Norman, K. A., Polyn, S. M., Detre, G. J., & Haxby, J. V. (2006). Beyond mind-reading: Multi-voxel pattern analysis of fMRI data. *Trends in Cognitive Sciences, 10*(9), 424–430. https://doi.org/10.1016/j.tics.2006.07.005.

O'Toole, A. J., Jiang, F., Abdi, H., & Haxby, J. V. (2005). Partially distributed representations of objects and faces in ventral temporal cortex. *Journal of Cognitive Neuroscience, 17*(4), 580–590. https://doi.org/10.1162/0898929053467550.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(Oct), 2825–2830.

Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 27*(8), 1226–1238. https://doi.org/10.1109/TPAMI.2005.159.

Polyn, S. M., Natu, V. S., Cohen, J. D., & Norman, K. A. (2005). Category-specific cortical activity precedes retrieval during memory search. *Science, 310*(5756), 1963–1966. https://doi.org/10.1126/science.1117645.

Ross, B. C. (2014). Mutual information between discrete and continuous data sets., Mutual Information between Discrete and Continuous Data Sets. *PloS One, PLoS ONE*, *9*, *9*(2, 2), e87357–e87357. https://doi.org/10.1371/journal.pone.0087357, https://doi.org/10.1371/journal.pone.0087357.

Saarimäki, H., Gotsopoulos, A., Jääskeläinen, I. P., Lampinen, J., Vuilleumier, P., Hari, R., Sams, M., & Nummenmaa, L. (2016). Discrete neural signatures of basic emotions. *Cerebral Cortex, 26*(6), 2563–2573. https://doi.org/10.1093/cercor/bhv086.

Saeys, Y., Inza, I., & Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. *Bioinformatics, 23*(19), 2507–2517. https://doi.org/10.1093/bioinformatics/btm344.

Saeys, Y., Abeel, T., & Peer, Y. V. de. (2008). Robust feature selection using ensemble feature selection techniques. In Machine Learning and Knowledge Discovery in Databases (pp. 313–325). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-87481-2_21, Robust Feature Selection Using Ensemble Feature Selection Techniques.

Sayres, R., Ress, D., & Grill-Spector, K. (2005). Identifying distributed object representations in human Extrastriate visual cortex. In *Proceedings of the 18th international conference on neural information processing systems* (pp. 1169–1176). Cambridge: MIT Press Retrieved from http://dl.acm.org/citation.cfm?id=2976248.2976395.

Stiglic, G., & Kokol, P. (2010). Stability of ranked gene lists in large microarray analysis studies. *BioMed Research International, 2010*, e616358. https://doi.org/10.1155/2010/616358.

Tohka, J., Moradi, E., Huttunen, H., & Initiative, A. D. N. (2016). Comparison of feature selection techniques in machine learning for anatomical brain MRI in dementia. *Neuroinformatics, 14*(3), 279–296. https://doi.org/10.1007/s12021-015-9292-3.

Toloşi, L., & Lengauer, T. (2011). Classification with correlated features: Unreliability of feature ranking and solutions. *Bioinformatics, 27*(14), 1986–1994. https://doi.org/10.1093/bioinformatics/btr300.

Turney, P. (1995). Technical note: Bias and the quantification of stability. *Machine Learning, 20*(1–2), 23–33. https://doi.org/10.1023/A:1022682001417.

Vergara, J. R., & Estévez, P. A. (2014). A review of feature selection methods based on mutual information. *Neural Computing and Applications, 24*(1), 175–186. https://doi.org/10.1007/s00521-013-1368-0.

Wang, Y., Li, Z., Wang, Y., Wang, X., Zheng, J., Duan, X., & Chen, H. (2015). A Novel Approach for Stable Selection of Informative Redundant Features from High Dimensional fMRI Data. *arXiv: 1506.08301 [Cs, Stat]*. Retrieved from http://arxiv.org/abs/1506.08301

Wright, S. (1965). The interpretation of population structure by F-statistics with special regard to Systems of Mating. *Evolution, 19*(3), 395–420. https://doi.org/10.1111/j.1558-5646.1965.tb01731.x.

Yan, S., Yang, X., Wu, C., Zheng, Z., & Guo, Y. (2014). Balancing the stability and predictive performance for multivariate voxel selection in fMRI study. In Brain Informatics and Health (pp. 90–99). Springer, Cham. https://doi.org/10.1007/978-3-319-09891-3_9, Balancing the Stability and Predictive Performance for Multivariate Voxel Selection in fMRI Study.

Zeithamova, D., de Araujo Sanchez, M.-A., & Adke, A. (2017). Trial timing and pattern-information analyses of fMRI data. *NeuroImage*, *153*(Supplement C), 221–231. https://doi.org/10.1016/j.neuroimage.2017.04.025.

Zhao, Z., & Liu, H. (2007). Spectral feature selection for supervised and unsupervised learning. In *Proceedings of the 24th international conference on machine learning* (pp. 1151–1157). New York: ACM. https://doi.org/10.1145/1273496.1273641.

Zhao, Z., Wang, L., Liu, H., & Ye, J. (2013). On similarity preserving feature selection. *IEEE Transactions on Knowledge and Data Engineering, 25*(3), 619–632. https://doi.org/10.1109/TKDE.2011.222.