Measuring the Stability of Feature Selection

Sarah Nogueira and Gavin Brown

School of Computer Science, University of Manchester,
Manchester M13 9PL, UK
{sarah.nogueira,gavin.brown}@manchester.ac.uk

Abstract. In feature selection algorithms, "stability" is the sensitivity of the chosen feature set to variations in the supplied training data. As such it can be seen as an analogous concept to the statistical variance of a predictor. However unlike variance, there is no unique definition of stability, with numerous proposed measures over 15 years of literature. In this paper, instead of defining a new measure, we start from an axiomatic point of view and identify what properties would be desirable. Somewhat surprisingly, we find that the simple Pearson's correlation coefficient has all necessary properties, yet has somehow been overlooked in favour of more complex alternatives. Finally, we illustrate how the use of this measure in practice can provide better interpretability and more confidence in the model selection process.

Keywords: Stability, Feature selection.

1 Introduction

High-dimensional datasets can be very expensive in terms of computational resources and of data collection. Predictive models in this situation often suffer from the *curse of dimensionality* and tend to overfit the data. For these reasons, feature selection (FS) has become an ubiquitous challenge that aims at selecting a "useful" set of features [8].

Stability of FS is defined as the sensitivity of the FS procedure to small perturbations in the training set. This issue is of course extremely relevant with small training samples, e.g. in bioinformatics applications - if the alteration/exclusion of just one training example results in a very different choice of biomarkers, we cannot justifiably say the FS is doing a reliable job. In early cancer detection, stability of the identified markers is a strong indicator of reproducible research [6],[12] and therefore selecting a stable set of markers is said to be equally important as their predictive power [7].

The study of stability poses several problems such as: What impacts stability? How can we make FS procedures more stable? How can we quantify it? A large part of the literature is dedicated to the later, which is the focus of this paper. Indeed, at a literature search conducted at the time of writing, we identified at least 10 different measures used to quantify stability [4],[8],[10],[11], [13],[14],[16],[17],[19],[21]. The existence of so many different measures without

known properties may lead to an incorrect interpretation of the stability values obtained.

As described by [8], FS procedures can have 3 types of outputs: *a weighting* on the features also called scoring (e.g. ReliefF), a *ranking* on the features (e.g. ranking by mutual information of the features with the target class) or a *feature set* (e.g. any wrapper approach). A weighting can be mapped into a ranking, and by applying a threshold on a ranking, a ranking can be mapped into a feature set; but the reverse is clearly not possible. For this reason, there exist stability measures for each type of output. In this paper, we focus on FS procedures that return a feature set.

An Example

Imagine we have $d = 5$ features to choose from. We can model the output feature set of the FS procedure by a binary vector $\mathbf{s}$ of length 5, where a 1 at the f^{th} position means the f^{th} feature has been selected and a 0 means it has not been selected. For instance, the vector $\begin{bmatrix}1\ 1\ 1\ 0\ 0\end{bmatrix}$ means that features 1-3 have been selected and features 4-5 have not been selected. Now imagine we apply two distinct FS procedures P_1 and P_2 to $M = 3$ different samples of the data and that we get the following output:

$$\mathcal{A}_1 = \begin{bmatrix}\mathbf{s}_1\\ \mathbf{s}_2\\ \mathbf{s}_3\end{bmatrix} = \begin{bmatrix}1\ 1\ 1\ 0\ 0\\ 1\ 1\ 1\ 0\ 0\\ 1\ 1\ 1\ 0\ 0\end{bmatrix} \quad \mathcal{A}_2 = \left.\begin{bmatrix}1\ 1\ 1\ 0\ 0\\ 1\ 0\ 1\ 1\ 0\\ 1\ 0\ 1\ 0\ 0\end{bmatrix}\right\} M = 3 \text{ feature sets} \tag{1}$$

where the rows of $\mathcal{A}_1$ and $\mathcal{A}_2$ represent the feature sets respectively returned by P_1 and P_2. All the feature sets in $\mathcal{A}_1$ are identical, therefore there is no variation in the output of the procedure. Each column of the matrix $\mathcal{A}_1$ represents the selection of each one of the 5 features. The observed frequency of the first three features is equal to 1 while the one of the two last features is equal to 0. This situation corresponds to a fully stable selection. Now let us look at $\mathcal{A}_2$. In that situation, we can see that there is some variation in the output of the FS procedure since the feature sets in $\mathcal{A}_2$ are different. If we look at the second and fourth columns of $\mathcal{A}_2$ corresponding to the selection of the second and fourth feature over the 3 feature sets, we can see that they are selected with a frequency equal to $\hat{p}_2 = \hat{p}_4 = \frac{1}{3}$, which shows some instability in the FS.

Quantifying the stability of FS consists in defining a function $\hat{\Phi}$ that takes the output $\mathcal{A}$ of the FS procedure as an input and returns a stability value. It is important to note that this is an *estimate* of a quantity, as the true stability is a random variable. We present the general framework to quantify stability in section 2. Coming from an axiomatic point of view, we derive a set of properties that we argue necessary for a stability measure and show that none of the existing measures have all desired properties in section 3. In section 4, we propose the use of the sample Pearson's correlation coefficient showing that it has all required properties and we provide an interpretation of the quantity estimated using this

measure. Finally, we illustrate the use of stability in the context of FS by a $L1$-regularized logistic regression and show how when coupled with the error of the model, it can help select a regularizing parameter.

2 Background

2.1 General Framework

To quantify the stability of FS, the following steps are carried out [1]:

1. Take M perturbed versions of the original dataset $\mathcal{D}$ (e.g. by using a resampling technique [3] such as bootstrap or noise injection [2]).
2. Apply the FS procedure to each one of the M samples obtained. This gives a sequence $\mathcal{A} = [\mathbf{s}_1, ..., \mathbf{s}_M]^T$ of M feature sets.
3. Define a function $\hat{\Phi} : \{0,1\}^{M \times d} \to \mathbb{R}$ taking the sequence of feature sets $\mathcal{A}$ as an input and measuring the stability of the feature sets in $\mathcal{A}$.

The main challenge here lies on the definition of an appropriate function $\hat{\Phi}$ that measures the stability in the choice of features in $\mathcal{A}$. Before looking into the approaches taken in the literature to define such a function $\hat{\Phi}$, we first establish the following notations that will be used in the remainder of the paper. We can denote the elements of the binary matrix $\mathcal{A}$ representing the M feature sets as follows:

$$\mathcal{A} = \begin{bmatrix} \mathbf{s}_1 \\ \vdots \\ \mathbf{s}_M \end{bmatrix} = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,d} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{M,1} & x_{M,2} & \cdots & x_{M,d} \\ \uparrow & \uparrow & & \uparrow \\ X_1 & X_2 & & X_d \end{pmatrix}$$

- For all $f \in \{1, ..., d\}$, the selection of the f^{th} feature is modelled by a Bernoulli variable[1] X_f with unknown parameter p_f. Therefore, each column of the matrix $\mathcal{A}$ can be seen as a realisation of the variable X_f, from which we will assume they are random samples.
- For all f in $\{1, ..., d\}$, $\hat{p}_f = \frac{1}{M} \sum_{i=1}^{M} x_{i,f}$ is the observed frequency of the f_{th} feature and is the *maximum likelihood estimator* of p_f.
- For all i in $\{1, ..., M\}$, $k_i = |\mathbf{s}_i|$ is the cardinality of feature set $\mathbf{s}_i$ (i.e. the number of features in $\mathbf{s}_i$). When all feature sets in $\mathcal{A}$ are of identical cardinality, we will simply denote the cardinality of the sets by k.
- For all (i, j) in $\{1, ..., M\}^2$, $r_{i,j}$ denotes the size of the intersection between feature sets $\mathbf{s}_i$ and $\mathbf{s}_j$ (i.e. the number of features they have in common).

[1] We therefore have a set of d correlated Bernoulli variables $(X_1, ..., X_d)$.

2.2 Quantifying Stability

The main approach that can be found in the literature is the *similarity-based approach*. It consists in defining the stability as **the average pairwise similarities** between the feature sets in $\mathcal{A}$ [8]. Let $\phi : \{0,1\}^d \times \{0,1\}^d \rightarrow \mathbb{R}$ be a function that takes as an input two feature sets $\mathbf{s}_i$ and $\mathbf{s}_j$ and returns a similarity value between these two sets. Then the stability $\hat{\Phi}(\mathcal{A})$ is defined as [2]:

$$\hat{\Phi}(\mathcal{A}) = \frac{1}{M(M-1)} \sum_{i=1}^{M} \sum_{\substack{j=1 \\ j \neq i}}^{M} \phi(\mathbf{s}_i, \mathbf{s}_j).$$

This approach has been very popular in the literature and many similarity measures ϕ have been proposed to that end. Popular examples of similarity measures are the *Jaccard index* [8] defined as follows:

$$\phi_{Jaccard}(\mathbf{s}_i, \mathbf{s}_j) = \frac{|\mathbf{s}_i \cap \mathbf{s}_j|}{|\mathbf{s}_i \cup \mathbf{s}_j|} = \frac{r_{i,j}}{k_i + k_j - r_{i,j}}.$$

For instance, if we take back the examples given in equation 1, using the Jaccard index we get the stability values of:

$$\hat{\Phi}_{Jaccard}(\mathcal{A}_1) = \frac{1}{3}\left(\phi_{Jaccard}(\mathbf{s}_1, \mathbf{s}_2) + \phi_{Jaccard}(\mathbf{s}_1, \mathbf{s}_3) + \phi_{Jaccard}(\mathbf{s}_2, \mathbf{s}_3)\right) = 1$$
$$\hat{\Phi}_{Jaccard}(\mathcal{A}_2) = \frac{1}{3}\left(\frac{2}{4} + \frac{2}{3} + \frac{2}{3}\right) = \frac{11}{18} \simeq 0.61.$$

As expected, we get a smaller stability value in the second case.

Nevertheless, as we further discuss in section 3, this similarity measure has been shown to provide stability estimates $\hat{\Phi}$ that are biased by the cardinality of the feature sets [11]. Based on this observation, Kuncheva [11] identifies a set of desirable properties and introduces a new similarity measure $\phi_{Kuncheva}$ between two feature sets $\mathbf{s}_i$ and $\mathbf{s}_j$ of identical cardinality as follows:

$$\phi_{Kuncheva}(\mathbf{s}_i, \mathbf{s}_j) = \frac{r_{i,j} - \mathbb{E}_{\nabla}[r_{i,j}]}{max(r_{i,j}) - \mathbb{E}_{\nabla}[r_{i,j}]} = \frac{r_{i,j} - \frac{k^2}{d}}{k - \frac{k^2}{d}},$$

where $\mathbb{E}_{\nabla}[r_{i,j}]$ is a correcting term equal to the expected value of $r_{i,j}$ when the FS procedure randomly selects k_i and k_j features from the d available features. As the random intersection of two sets of k_i and k_j objects follows a hypergeometric distribution, this term is known to be equal to $\frac{k_i k_j}{d}$ which is equal to $\frac{k^2}{d}$ here since $k_i = k_j = k$. This measure has been very popular in the literature because of its known properties. Nevertheless, because it is only defined for feature sets $\mathbf{s}_i$ and $\mathbf{s}_j$ of identical cardinality, it can only be used to measure the stability

[2] ϕ is not necessarily symmetric.

of FS algorithms that are guaranteed to select a constant number of features. As we have illustrated in example (1), the output of an FS procedure is not always guaranteed to be of constant cardinality. Examples of such FS procedures are in feature selection by hypothesis testing [15]. For this reason, several attempts at extending this measure to feature sets of varying cardinality have been made in the literature, somehow losing some of the important properties. Even though most similarity measures used to measure stability are increasing functions of the size of the intersection between the feature sets, they have shown to lack of some other required properties.

Other approaches have been taken in the literature to define a function $\hat{\Phi}$, without going through the definition of a similarity measure. A popular measure in this category is Somol's measure CW_{rel} [16] (also called Relative Weighted Consistency Measure). Its definition is a direct function of the observed frequencies of selection of each feature $\hat{p}_f$. This is the only measure in this category that is not biased by the cardinality of the feature sets in $\mathcal{A}$ and holds the property of *correction for chance*.

Due to the multitude of stability measures, it is necessary to discriminate between them with principled reasons which is the purpose of the next section.

3 Required Properties of a FS Stability Measure

In this section, we identify and argue for 4 properties which all stability measures should possess. These properties we will argue are necessary for a sensible measure of stability and if missing even one, a measure will behave nonsensically in certain situations. We will later demonstrate that from 10 stability measures published and widely used in the literature, none of them possesses all these properties.

Property 1: Fully Defined
Imagine we have an FS procedure: Procedure P. Procedure P sometimes returns 4 features, but sometimes 5, so the returned set size *varies*. It would seem sensible to have a stability measure which accounts for this. Unfortunately not all do - Krízek's and Kuncheva's measures [10, 11] are *undefined* in this scenario.

Property 2: Upper/Lower Bounds
For useful *interpretation* of a stability measure and comparison across problems, the range of values of a stability measure should be finite. Imagine we wanted to evaluate the stability of an FS procedure and that we got a value of 0.9. How can we interpret this value? If we know that the stability values can take values in $[0, 1]$, then this corresponds to a fairly high stability value as it is close to its maximum 1. Let us imagine now that we have a stability value that can take values in $(-\infty, +\infty)$. A value of 0.9 is not meaningful any more.

Property 3:

(a) **Deterministic Selection → Maximum Stability**
Imagine that Procedure P selects the same k features every time, regardless of the supplied data. This is a completely *stable* method, so it would seem sensible that any stability *measure* should reflect this, returning its maximum value. Surprisingly, this is not always the case. Figure 1 [LEFT] shows the stability value using Lustgarten's measure [13] when for different values of k. The result clearly *varies* with k. That is, if Procedure P_1 were to repeatedly select features 1-4 and Procedure P_2 then repeatedly selects features 1-5: this measure judges P_1 and P_2 to have *different degrees of stability*, even though they are both completely deterministic procedures.

(b) **Maximum Stability → Deterministic Selection**
The converse to the above should also hold. If a measure has a maximum possible value C, it should *only* return that value when Procedure P is deterministic. For example, imagine Procedure P selects features 1-4 half the time, and 1-5 the rest of the time. Wald's measure and CW_{rel} return a value of 1 in this scenario – their maximum possible value, even though clearly there is some variation in the feature sets. Figure 1 [RIGHT] illustrates this.

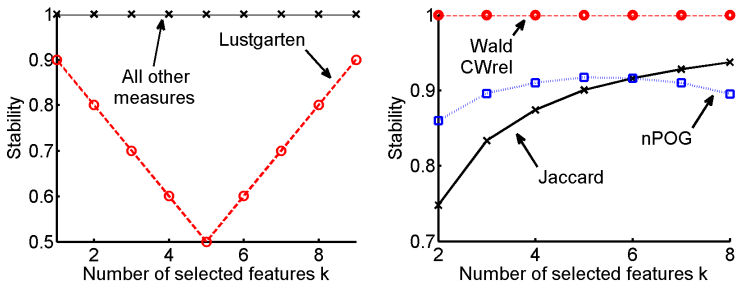


Fig. 1: Illustration of Property 3. Demonstration that Lustgarten's measure violates Property 3a [LEFT] by giving the stability when all feature sets in $\mathcal{A}$ are identical against k for $d = 10$. Demonstration that Wald's measure and CW_{rel} violate Property 3b [RIGHT]. Features $[1, ..., k]$ are selected half of the time and feature $[1, ..., k-1]$ are selected the other half of the time. Stability values against k for $d = 10$ and $M = 100$.

Property 4: Correction for Chance
This was first noted by Kuncheva [11]. This ensures that when the FS is **random**, the expected value of the stability estimate is constant, which we have set here to 0 by convention. Imagine that a procedure P_1 **randomly** selects 5 features and that a procedure P_2 randomly selects 6 features, the stability value should be the same. As illustrated by Figure 2, this is not the case for all measures.

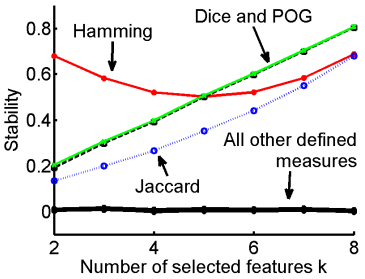


Fig. 2: Demonstration that Hamming, Jaccard, *POG* and Dice violate Property 4. **Random** selection of k features with probability 50% and of $k-1$ and $k+1$ features with probability 25% each. Stability against k for $d = 10$ and $M = 100$.

Summary

We provide a formal description of the required properties and sum up the properties of the different existing stability measures[3] in Table 1. We can observe that none of the measures satisfy all four desired properties.

1. **Fully defined.** $\hat{\Phi}$ is defined for any sequence $\mathcal{A}$ of feature sets.
2. **Bounds.** $\hat{\Phi}$ is bounded by constants.
3. **Maximum.** $\hat{\Phi}$ reaches its maximum $\iff$ All feature sets in $\mathcal{A}$ are identical.
4. **Correction for chance.** $\mathbb{E}_{\nabla}[\hat{\Phi}(\mathcal{A})] = 0$ when the selection is random.

Table 1: Properties of Stability Measures

	Fully defined	Bounds	Maximum	Correction for chance	
Jaccard [8]	✓	✓	✓		Similarity-based
Hamming [4]	✓	✓	✓		
Dice [19]	✓	✓	✓		
POG [14]	✓	✓	✓		
Kuncheva [11]		✓	✓	✓	
nPOG [21]	✓		✓	✓	
Lustgarten [13]	✓	✓		✓	
Wald [17]	✓			✓	
Krízek [10]			✓		
CW_{rel} [16]	✓	✓		✓	

[3] Sketches of proofs are given in the supplementary material available online at `www.cs.man.ac.uk/~nogueirs/files/supplementary-material-ECML-2016.pdf`.

4 The Sample Pearson's Correlation Coefficient

In this section, we first demonstrate that the stability measure using the sample Pearson's correlation coefficient[4] as a similarity measure satisfies all 4 properties. The sample Pearson's correlation coefficient between two feature sets $\mathbf{s}_i$ and $\mathbf{s}_j$ is by definition:

$$\phi_{Pearson}(\mathbf{s}_i, \mathbf{s}_j) = \frac{\frac{1}{d}\sum_{f=1}^{d}(x_{i,f} - \bar{x}_{i,.})(x_{j,f} - \bar{x}_{j,.})}{\sqrt{\frac{1}{d}\sum_{f=1}^{d}(x_{i,f} - \bar{x}_{i,.})^2}\sqrt{\frac{1}{d}\sum_{f=1}^{d}(x_{j,f} - \bar{x}_{j,.})^2}},$$

where $\forall i \in \{1, ..., M\}, \bar{x}_{i,.} = \frac{1}{d}\sum_{f=1}^{d} x_{i,f} = \frac{k_i}{d}$.

As other similarity measures, we can point out that $\phi_{Pearson}(\mathbf{s}_i, \mathbf{s}_j)$ is an increasing function of the size of the intersection of the selected features $r_{i,j}$ between the feature sets $\mathbf{s}_i$ and $\mathbf{s}_j$. Moreover, the sample Pearson correlation coefficient is already the similarity measure used when the FS outputs a scoring on the features[8], even though it has never been used or studied in the context of feature sets. The use of Pearson's correlation coefficient is therefore going towards a unification of the assessment of stability of FS.

The sample Pearson's correlation also subsumes other measures when the cardinality of the feature sets is constant, as stated by Theorem 2. This result is quite surprising, knowing that coming from an axiomatic point of view on a set of desirable properties, Kuncheva defined a measure that is indeed a specific case of the well-known sample Pearson's correlation coefficient $\phi_{Pearson}$.

Theorem 1. *For all $(i, j) \in \{1, ..., M\}^2$, the sample Pearson's correlation coefficient can be re-written:*

$$\phi_{Pearson}(\mathbf{s}_i, \mathbf{s}_j) = \frac{r_{i,j} - \mathbb{E}_{\nabla}[r_{i,j}]}{d\ \upsilon_i \upsilon_j} = \frac{r_{i,j} - \frac{k_i k_j}{d}}{d\ \upsilon_i \upsilon_j}, \tag{2}$$

where $\forall i \in \{1, ..., M\}, \upsilon_i = \sqrt{\frac{k_i}{d}(1 - \frac{k_i}{d})}$. Therefore it possesses the property of correction for chance.

Proof. *The proof is provided in the supplementary material.*

Theorem 2. *When k is constant, the stability using Pearson's correlation is equal to some other measures, that is:*

$$\hat{\Phi}_{Pearson} = \hat{\Phi}_{Kuncheva} = \hat{\Phi}_{Wald} = \hat{\Phi}_{nPOG}.$$

Proof. *Straightforward using Theorem 1 and the definition of the other similarity measures given in the supplementary material.*

[4] Also called the *Phi coefficient* in this case since we are dealing with binary vectors.

4.1 Required properties

Property 1: Fully Defined
As most of the other similarity measures, we can see in Equation 2 that the given expression presents indeterminate forms for $k_i = 0$, $k_j = 0$, $k_i = d$ and $k_j = d$. Because these indeterminate forms correspond to situations in which either all features or none of them are selected, these indeterminate forms are not critical in the context of feature selection since the main aim of FS is to identify a non-empty strict subset of relevant features taken from the available features. Nevertheless, for completeness, following the works on the correlation coefficient in [5], we set $\phi_{Pearson}$ to 0 when:

- $k_i = 0$ and $k_j \neq 0$ or vice-versa;
- $k_i = d$ and $k_j \neq d$ or vice-versa.

When $k_i = k_j = 0$ or $k_i = k_j = d$, then the feature sets are identical (either empty set $\emptyset$ or full set Ω) and in that case, we set $\phi_{Pearson}$ to be equal to 1 so it meets the property of maximum. Therefore, the resulting stability $\hat{\Phi}_{Pearson}$ has the property of being fully defined.

Property 2: Bounds
$\phi_{Pearson}$ is known to take values between -1 and 1: the similarity between two sets is minimal (i.e. equal to -1) when the two sets are fully anti-correlated (i.e. when $\mathbf{s}_i$ and $\mathbf{s}_j$ are complementary sets) and maximal (equal to 1) when the two sets are fully correlated (i.e. identical). Since $\hat{\Phi}_{Pearson}$ is the average value of $\phi_{Pearson}$ over all the possible pairs in $\mathcal{A}$, $\hat{\Phi}_{Pearson}$ will also be in the interval -1 and 1 and is therefore bounded by constants.

Theorem 3. *The stability estimate $\hat{\Phi}_{Pearson}$ is asymptotically in the interval $[0, 1]$ as M approaches infinity.*

Proof. *The proof is provided in the supplementary material.*

The asymptotic bounds on the stability estimates make the stability values obtained more interpretable. Indeed, knowing how the stability values behave as M increases allows us to understand better how to interpret these values. Theorem 3 tackles the misconception according to which negative stability values correspond to FS algorithms worse than random: asymptotically, any FS procedure will have a positive estimated stability.

Property 3: Maximum
When $\mathbf{s}_i = \mathbf{s}_j$, we have $\phi_{Pearson}(\mathbf{s}_i, \mathbf{s}_j) = 1$ and therefore $\hat{\Phi}_{Pearson} = 1$ when all the feature sets in $\mathcal{A}$ are identical. Conversely, $\phi_{Pearson}(\mathbf{s}_i, \mathbf{s}_j) = 1$ implies $\mathbf{s}_i = \mathbf{s}_j$, which gives us that $\hat{\Phi}_{Pearson} = 1$ implies all sets in $\mathcal{A}$ are identical.

Property 4: Correction for Chance
This property is given by Theorem 1.

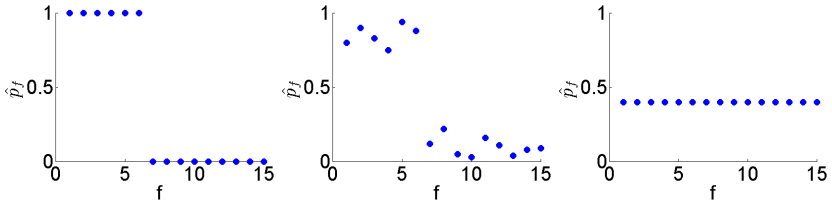
4.2 Interpreting Stability

In this section, we aim at providing an interpretation of the stability value when using the sample Pearson's correlation. For simplicity, we focus on the case where the FS selects a constant number of features k. Hereafter, $\hat{\Phi}$ will denote $\hat{\Phi}_{Pearson}$. By phrasing the concept of stability in this way, it highlights an important point - that we are *estimating* a quantity. The stability is a random variable, from which we have a sample of size M.

Let $\widehat{Var}(X_f) = \frac{M}{M-1}\hat{p}_f(1-\hat{p}_f)$ be the unbiased sample variance of the variable X_f. When the cardinality of the feature sets is constant, we can re-write the stability using the sample Pearson's correlation coefficient as follows:

$$\hat{\Phi}_{Pearson} = 1 - \frac{S}{S_{max}}, \tag{3}$$

where the average total variance $S = \frac{1}{d}\sum_{f=1}^{d}\widehat{Var}(X_f)$ is a measure of the variability in the choice of features and where $S_{max} = \frac{k}{d}\left(1-\frac{k}{d}\right)$ the maximal value of S given that the FS procedure is selecting k features per feature set. In this situation, Equation 3 shows that the stability decreases monotonically with the average variance of X_f.



(a) Scenario 1: $\hat{\Phi} = 1$ (b) Scenario 2: $\hat{\Phi} = 0.58$ (c) Scenario 3: $\hat{\Phi} = 0$

Fig. 3: The parameters $\hat{p}_f$ of the random variables X_f in 3 scenarios for $d = 15$

Because $\widehat{Var}(X_f) = 0$ whenever $\hat{p}_f = 0$ or $\hat{p}_f = 1$, the maximum stability is achieved when all features are selected with an observed frequency equal to 0 or 1. Figure 3 illustrates how to interpret the value $\hat{\Phi}$ in 3 scenarios. Let us assume we have an FS procedure selecting $k = 6$ features out of $d = 15$ features. Scenario 1 illustrates the situation in which the FS algorithm always returns the same feature set made of the first k features. In that situation, the probability of selection of the k first features is equal to 1 and the one of the remaining features is equal to 0, which gives $S = 0$ and therefore a stability $\hat{\Phi}$ equal to its maximal value 1. Scenario 2 illustrates the case where the FS is not completely stable, even though we can still distinguish two group of features. In that scenario, the stability is equal to $\hat{\Phi} = 0.58$. Scenario 3 is the limit case scenario in which the

selection of the k features is random. In that scenario, the d features have a frequency of selection all equal to $\hat{p}_f = \frac{k}{d} = \frac{6}{15}$. In that situation, the variance $Var(X_f) = \frac{k}{d}(1 - \frac{k}{d}) = 0.24$ of each of the random variables X_f is maximal. This gives $S = S_{max}$ and therefore $\hat{\Phi} = 0$. These scenarios illustrate the need to rescale the mean total variance by the one of a random FS procedure and give a useful interpretation of the estimated stability using Pearson's correlation.

5 Experiments

In the previous section we argued for an axiomatic treatment of stability measures — and demonstrated that the simple solution of using Pearson's correlation coefficient allows for all desirable properties.

In this section, we illustrate how stability can be used in practice to select a regularizing parameter in the context of feature selection by a $L1$-regularized regression. We show how without sacrificing a significant amount in terms of error, a regularizing parameter corresponding to a higher stability can be chosen. On the artificial dataset considered, we show how an increase in stability can help discarding the use of irrelevant features in the final model.

5.1 Description of Dataset

We use a synthetic dataset [9] – a binary classification problem, with 2000 instances and $d = 100$ features, where only the first 50 features are relevant to the target class. Instances of the positive class are i.i.d. drawn from a normal distribution with mean $\mu_+ = (\underbrace{1, ..., 1}_{50}, \underbrace{0, ..., 0}_{50})$ and covariance matrix:

$$\Sigma = \begin{bmatrix} \Sigma^*_{50\times 50} & \mathbf{0}_{50\times 50} \\ \mathbf{0}_{50\times 50} & \mathbf{I}_{50\times 50} \end{bmatrix}$$

where $\Sigma^*_{50\times 50}$ is the matrix with ones on the diagonal and ρ, a parameter taken in $[0, 1]$ controlling the degree of redundancy everywhere else. The mean for the negative class is taken equal to $\mu_- = (\underbrace{-1, ..., -1}_{50}, \underbrace{0, ..., 0}_{50})$. The larger the value of ρ, the more the 50 relevant features will be correlated to each other.

5.2 Experimental Procedure and Results

We use $L1$-regularized logistic regression with 100 different regularizing parameters on the synthetic dataset for different degrees of redundancy ρ. The $L1$-regularization results in some coefficients being forced to zero – any coefficients left as non-zero after fitting the model are regarded as "selected" by the model.[5]

[5] You can reproduce these experiments in Matlab with the code given at `https://github.com/nogueirs/ECML2016`

Our experimental procedure is as follows. We take the 2000 samples and divide into 1000 for model selection (the regularizing parameter λ) and 1000 for selection of the final set of features. The model selection set can be used simply to optimize error, or to optimize error/stability simultaneously – the experiments will demonstrate that the latter provides a lower false positive rate in the final selection of features.

For each regularizing parameter λ, we take $M = 100$ bootstrap samples to train our models. We then compute the stability $\hat{\Phi}$ and the out-of-bag (OOB) estimate of the error[6] using the coefficients returned.

Figure 4 shows the OOB error [LEFT] and the stability [RIGHT] versus the regularization parameter λ for a degree of redundancy $\rho = 0$ (i.e. the relevant features are independent from each other). On this case, picking up a value of λ that minimizes the OOB error is also the value of λ that maximizes the stability. Indeed for $\lambda = 4.12 \times 10^{-4}$, we get an error of 0.30 and a stability of 0.98, which means the same features are picked up on nearly all bootstrap samples.

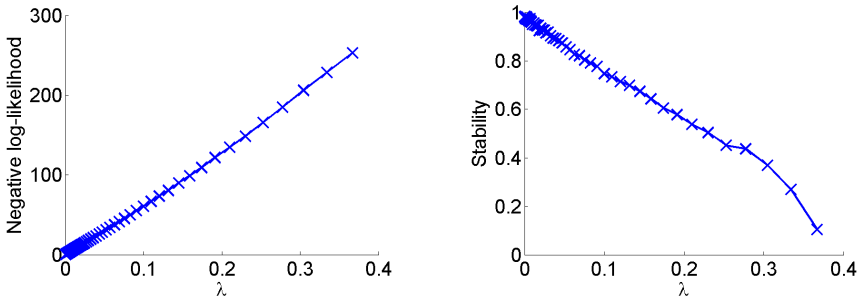


Fig. 4: Results for $\rho = 0$. Each point on the line corresponds to a different regularizing parameter λ. We can see that both high stability and low error are reached for $\lambda = 4.12 \times 10^{-4}$.

Let us now take a degree of redundancy $\rho = 0.3$. In a normal situation, we would choose the regularizing parameter that minimizes the error which is $\lambda = 0.009$, shown in the left of Figure 5. The right of the same figure shows the pareto optimal front, the trade-off of the two objectives – if we sacrifice some error, we can drastically increase stability.

Figure 6 gives the observed frequencies of selection $\hat{p}_f$ of each feature over the $M = 100$ bootstraps for $\lambda = 0.009$ [LEFT] and $\lambda = 0.023$ [RIGHT]. We

[6] Here, the error is taken to be the negative log-likelihood, a measure of goodness-of-fit of the model. The lower the value, the better the model.

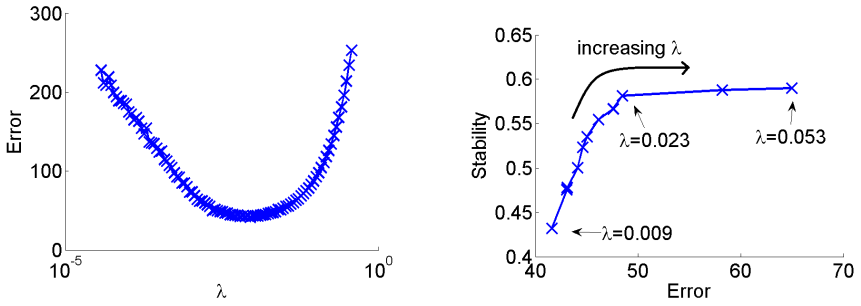
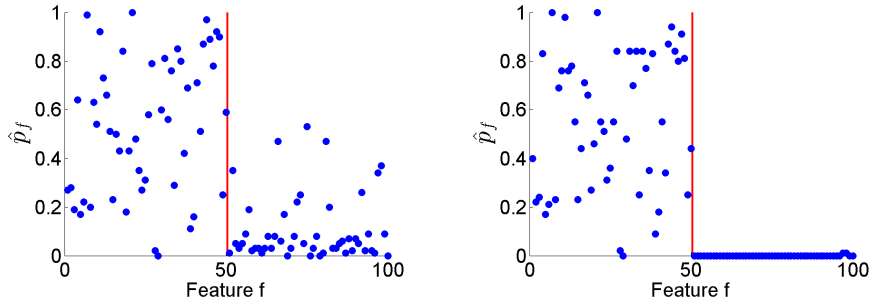


Fig. 5: If we optimize just OOB error [LEFT] we obtain $\lambda = 0.009$, but if we optimize a trade-off [RIGHT] of error/stability, sacrificing a small amount of error we get $\lambda = 0.023$, and can significantly increase feature selection stability.



(a) $\lambda = 0.009$ (b) $\lambda = 0.023$

Fig. 6: The observed frequencies of selection $\hat{p}_f$ for each feature for two values of λ in the pareto front for $\rho = 0.3$. The Features on the left of the red vertical line correspond to relevant features and the ones on the right to irrelevant ones.

can see on the right figure that nearly all irrelevant features have a frequency of selection of 0. Only two irrelevant features have a frequency of selection different from 0 with $\hat{p}_f = 0.01$, which means they have been selected on one of the 100 bootstrap samples only. From looking at the values of $\hat{p}_f$ for the value of λ minimizing the error on the left, we cannot discriminate the set of relevant features from the set of irrelevant ones by looking at the frequencies of selection. Even though $\lambda = 0.023$ does not provide a *high* stability value, we can see how we can benefit from taking $\lambda = 0.023$ instead of $\lambda = 0.009$. The features used in the model (the ones with a non-zero coefficient) are indeed relevant to the target class. As explained in section 4.2, the closer the observed frequencies are to 0 or 1, the higher the stability value will be.

Final feature set chosen: The model selection procedure on the first 1000 examples has suggested $\lambda = 0.009$ and $\lambda = 0.023$. We can now use these on

the final 1000 holdout set to select a set of features, again with $L1$ logistic regression, and compare the 2 feature sets returned. Table 2 shows the false positives (irrelevant features that were falsely identified as relevant) and the false negatives (relevant features that were missed), for three different degrees of increasing redundancy. In all cases, the methodology involving stability reduces the FP rate to zero, with no significant effect on FN rate.

Table 2: False positives and false negatives for different degrees of redundancy ρ

Redundancy	λ_{error}	λ_{ϕ}
low	$FP = 4$, $FN = 17$	$FP = 0$, $FN = 17$
medium	$FP = 7$, $FN = 24$	$FP = 0$, $FN = 25$
high	$FP = 5$, $FN = 35$	$FP = 0$, $FN = 33$

This case study also shows that feature redundancy is a source of instability of FS, as hypothesized by [8],[9],[18]. Similar results have been obtained for $\rho = 0.5$ and $\rho = 0.8$, with smaller stability values for the data points in the pareto front as we increased the degree of redundancy ρ.

6 Conclusions and Future Work

There are many different measures to quantify stability in the literature – we have argued for a set of properties that should be present in any measure, and found that several existing measures are lacking in this respect. Instead, we suggest the use of Pearson's correlation as a similarity measure, in the process showing that it is a generalization of the widely used Kuncheva index. We provide an interpretation of the quantity estimated through the typical procedure and illustrate its use in practice. We illustrate on synthetic datasets how stability can be beneficial and provides more confidence in the feature set returned.

Depending on the type of application, we might want the stability measure to take into account feature redundancy. Such measures attempt to evaluate the stability of the *information* in the feature sets returned by the FS procedure rather than the stability of the feature sets themselves [20, 21]. These measures are generalizations of POG, $nPOG$ (called $POGR$ and $nPOGR$ [21]) and of the Dice coefficient [19] and reduce to these when there is no redundancy between the features. Because their simpler versions do not have the set of desired properties as shown in Table 1, we leave this type of measures to future work.

Acknowledgments. This work was supported by the Engineering and Physical Sciences Research Council, through a Centre for Doctoral Training [EP/I028099/1] and a project grant [EP/L000725/1]. **Data access statement**: All research data supporting this publication are directly available within this publication.

References

1. Alelyani, S., Zhao, Z., Liu, H.: A dilemma in assessing stability of feature selection algorithms. In: HPCC (2011)
2. Altidor, W., Khoshgoftaar, T.M., Napolitano, A.: A noise-based stability evaluation of threshold-based feature selection techniques. In: IRI'11 (2011)
3. Boulesteix, A.L., Slawski, M.: Stability and aggregation of ranked gene lists. Briefings in Bioinformatics (2009)
4. Dunne, K., Cunningham, P., Azuaje, F.: Solutions to instability problems with sequential wrapper-based approaches to feature selection. Tech. rep., Journal of Machine Learning Research (2002)
5. Edmundson, H.P.: A correlation coefficient for attributes or events. In: Proc. Statistical association methods for mechanized documentation (1966)
6. He, Z., Yu, W.: Review article: Stable feature selection for biomarker discovery. Comput. Biol. Chem. (2010)
7. Jurman, G., Merler, S., Barla, A., Paoli, S., Galea, A., Furlanello, C.: Algebraic stability indicators for ranked lists in molecular profiling. Bioinformatics (2008)
8. Kalousis, A., Prados, J., Hilario, M.: Stability of feature selection algorithms: a study on high-dimensional spaces. Knowl. Inf. Syst. (2007)
9. Kamkar, I., Gupta, S.K., Phung, D., Venkatesh, S.: AI 2015: Advances in Artificial Intelligence: 28th Australasian Joint Conference, Canberra, ACT, Australia, Proceedings, chap. Stable Feature Selection with Support Vector Machines (2015)
10. Krízek, P., Kittler, J., Hlavác, V.: Improving stability of feature selection methods. In: CAIP (2007)
11. Kuncheva, L.I.: A stability index for feature selection. In: Artificial Intelligence and Applications (2007)
12. Lee, H.W., Lawton, C., Na, Y.J., Yoon, S.: Robustness of chemometrics-based feature selection methods in early cancer detection and biomarker discovery. Statistical Applications in Genetics and Molecular Biology (2012)
13. Lustgarten, J.L., Gopalakrishnan, V., Visweswaran, S.: Measuring stability of feature selection in biomedical datasets. AMIA Annu Symp Proc (2009)
14. MAQC consortium: The MicroArray quality control project shows inter- and intraplatform reproducibility of gene expression measurements. Nat Biotech. (2006)
15. Sechidis, K., Brown, G.: Markov blanket discovery in positive-unlabelled and semi-supervised data. In: ECML (2015)
16. Somol, P., Novovičová, J.: Evaluating stability and comparing output of feature selectors that optimize feature subset cardinality. IEEE Transactions on Pattern Analysis and Machine Intelligence (2010)
17. Wald, R., Khoshgoftaar, T.M., Napolitano, A.: Stability of filter- and wrapper-based feature subset selection. In: International Conference on Tools with Artificial Intelligence. IEEE Computer Society (2013)
18. Woznica, A., Nguyen, P., Kalousis, A.: Model mining for robust feature selection. In: KDD (2012)

19. Yu, L., Ding, C.H.Q., Loscalzo, S.: Stable feature selection via dense feature groups. In: KDD (2008)
20. Yu, L., Han, Y., Berens, M.E.: Stable gene selection from microarray data via sample weighting. IEEE/ACM Trans. Comput. Biology Bioinform. (2012)
21. Zhang, M., Zhang, L., Zou, J., Yao, C., Xiao, H., Liu, Q., Wang, J., Wang, D., Wang, C., Guo, Z.: Evaluating reproducibility of differential expression discoveries in microarray studies by considering correlated molecular changes. Bioinformatics (2009)