A Bootstrap Based Neyman–Pearson Test for Identifying Variable Importance

Gregory Ditzler, Robi Polikar, and Gail Rosen

Abstract

Selection of most informative features that leads to a small loss on future data is arguably one of the most important steps in classification, data analysis and model selection. Several feature selection algorithms are available; however, due to noise present in any data set, feature selection algorithms are typically accompanied by an appropriate cross validation scheme. In this work, we propose a statistical hypothesis test derived from the Neyman-Pearson lemma for determining if a feature is statistically relevant. The proposed approach can be applied as a wrapper to any feature selection algorithm, regardless of the feature selection criteria used by that algorithm, to determine whether a feature belongs in the relevant set. Perhaps more importantly, this procedure efficiently determines the number of relevant features given an initial starting point. We provide freely available software implementations of the proposed methodology.

1 Introduction

High dimensional data sets are frequently encountered in real-world machine learning problems. In such scenarios, the feature vectors, $\mathbf{x}$, are represented in a high dimensional space $\mathbb{R}^K$, where some or many of the K features may be irrelevant, carry little or no information about the learning problem while others may be redundant (i.e., carry the same information as other features in regards to the class labels). In either of these scenarios, using fewer features is likely to be sufficient for learning. A plethora of algorithms have been proposed, many well-established, for reducing the number of features to k ($k \ll K$) by optimizing an objective function that selects the k "most informative" features, while minimizing the redundancy of these k features (see [1, 2] for a review of such approaches). While individual feature selection methods vary from each other, many share the general principle: select $k < K$ features through (possibly) an iterative process that optimizes a pre-determined objective function.

Feature selection (FS) algorithms typically fall into one of three categories: *wrapper*–, *embedded*–, and *filter*–based approaches. A FS wrapper is a classifier dependent implementation that selects features minimizing some predictive scoring objective function for a specific classification model. Embedded methods corporate FS into the construction of the classification model – still a classifier dependent model for FS. Finally, filters are independent of the classifier, and select features based on an objective function that is independent of classification loss, such as mutual information or χ^2 statistics.

Selecting the appropriate subset size k is one of the key considerations in feature subset selection. Heuristics may lead to severely suboptimal results, whereas grid searches are infeasible for large data sets. Also of practical importance is whether a post-hoc test can be used to determine the accuracy, or the optimality, of initial selection of k, and taking the appropriate action when warranted. There are existing hypothesis-testing approaches for FS; however, the implementations of these approaches are usually not flexible with respect to other objective functions. For example, the χ^2 test may be used to measure a lack of independence between data variables X and label variables Y; however, the χ^2 based FS does not allow the inspection of objective functions such as mutual information.

In this brief correspondence, we present a Neyman-Pearson hypothesis test for the identification of relevant features. Our approach is derived from a given base FS algorithm that selects k features across several bootstrap data sets. Given the results obtained by running the FS algorithm on n bootstrap data sets, we derive a hypothesis test to infer the number of relevant features k^*, which may in fact be different than the k that was used by the base FS algorithm.

This article is organized as follows: section 2 presents related work. Section 3 presents the proposed approach. Section 4 presents the results on several synthetic and UCI benchmark data sets. Finally, section 5 includes a discussion and concluding remarks.

2 Related Work

FS is a well-researched area that seeks to find an optimal feature subset, cleared from irrelevant and redundant features. Such a feature subset not only improves classification accuracy, but also reduces the computational complexity of the model. Guyon & Elisseeff's tutorial on variable selection covers several FS and subsequent validation methods [1]. Validation is important in evaluating a FS approach, as it allows us to determine the robustness of the approach to variations in its free parameter(s). Selecting and inferring values of such free parameters, such as the number of features a method selects as relevant, is the focus of this brief communication. Brown et al. recently presented an information-theoretic FS framework for maximizing of the conditional likelihood function [3], where they examine the *consistency* to measure the stability of FS methods. However, in their approach k was selected heuristically, and was not optimized for any of their experiments, an issue that is addressed in this communication.

Yang et al. developed a hypothesis test based FS method to find textual abundance features that contribute to the "spam" class for email prediction [4]. Their work presented a methodology that used a Binomial hypothesis test (Bi-test) that was designed to identify features that were highly probable to be in a spam email. However, the approach, while effective, assumes the features of the data are of a particular form, or distribution. Other approaches, such as Relief and Focus, can be used to determine feature relevance [5, 6]; however, these approaches do not allow for the selection of the objective function being optimized.

Some FS methods have the capability to "dynamically" select the number of features based on the χ^2 statistic [7], which measures the lack of independence between random variables X and Y. However, using the χ^2 statistic fixes the objective function for the FS method. Developing a general and versatile framework that allows free choice of the objective function while providing inference on parameter selection appears to be an under explored area.

Kuncheva presents a consistency index for determining the level of stability of a FS algorithm when tested with multiple validation data sets [8]. Kuncheva's consistency index was designed to meet three primary criteria: the consistency index (a) is a monotonically increasing function of the number of features common to two feature sets, (b) is bounded, and (c) has a constant value for independently drawn subsets of features of the same cardinality.

Definition 2.1 (Consistency [8]) *The consistency index for two subsets $\mathcal{A} \subset \mathcal{X}$ and $\mathcal{B} \subset \mathcal{X}$, such that $r = |\mathcal{A} \cap \mathcal{B}|$ and $|\mathcal{A}| = |\mathcal{B}| = k$, where $1 \leq k \leq |\mathcal{X}| = K$, is*

$$\mathcal{I}_C(\mathcal{A}, \mathcal{B}) = \frac{rK - k^2}{k(K-k)}$$

3 Neyman-Pearson Hypothesis Testing for Feature Selection

Different FS algorithms optimize different objective functions, hence, making different assumptions about the dispersion or distribution of the data. Unfortunately, few methods can offer the dynamic selection of k, and fewer yet have the ability to work with other FS objective functions (e.g., they already have a specified filter criteria: see FS with the χ^2 statistic [7]).

In this section, we present an algorithm-independent meta–approach to determine an appropriate level of k using the Neyman-Pearson feature selection (NPFS) hypothesis test. This approach can be used with any FS algorithm. Table 1 contains the mathematical notations used throughout this manuscript.

3.1 Overview of the Proposed Method & Preliminaries

A FS algorithm, $\mathcal{F}$, is run n-times with bootstrap data sets sampled uniformly from $\mathcal{D}$. In this setting, data instances – and not the features – that are sampled randomly. For each bootstrap data set, $\mathcal{F}$ selects k of the K features in the "relevant feature set". For the moment, we assume there is a k^*, the *optimal* number of relevant features. Ideally, the same k features would be found by $\mathcal{F}$ as relevant over each of the n trials; however, this is rarely the case due to initializations and randomness in the bootstrap sample. A consistency index can be used to measure the stability of the relevant feature sets over these n trials. This index, however, is not based on a statistical hypothesis test, nor is it designed to determine if a feature is consistently selected as relevant. In fact, by Kuncheva's formulation, $\mathcal{I}_C(\mathcal{A}, \mathcal{B})$ is a random variable (this is easy to see since $R = r$ is a random variable with a hypergeometric distribution).

Table 1: Mathematical Notations

Notation	Meaning
$\mathcal{X}$	full set of features, $\lvert\mathcal{X}\rvert = K$
$\mathcal{F}$	feature selection algorithm
X_l	Bernoulli random variable indicating if a feature was selected as relevant on the lth bootstrap trial
Z	Binomial random variable
H_0	null hypothesis
H_1	alternative hypothesis
ζ_{crit}	Neyman-Pearson (critical) threshold
k	number of features selected by $\mathcal{F}$
n	number of bootstraps
$T(Z)$	sufficient statistic of random variable Z

3.2 Algorithm Derivation & Implementation

Let us first consider a hypothesis test being applied to a single feature (the proposed test can be applied to each feature individually). At each bootstrap iteration, $\mathcal{F}$, returns a set of indices for the relevant feature set. For each feature in the set $\mathcal{X}$, we mark whether the feature was in the relevant set ($X_l = 1$) or not in the set ($X_l = 0$), where $l \in [n]$ is the bootstrap iteration.

In this situation, we can determine that the random variable X_l is distributed as a Bernoulli random variable with probability p (that is yet to be determined). The n Bernoulli random variables from the n bootstrap data sets form a Binomial distribution with $Z_n = X_1 + \ldots + X_n$ successes ($Z_n = z$ be an observation of the random variable Z_n). If a feature is selected by chance, then the probability for such a feature appearing in the relevant feature set is $p_0 = k/K$. Now, there is the observed probability of a feature appearing in the relevant feature set from the bootstrap trials, which is $p_1 = z/n$. If all features were equally relevant (or equally irrelevant), we would expect these probabilities to be equal to one another. Ultimately, we would like to know if $p_1 > p_0$, or in other words, if the probability of a feature being in the relevant set is greater than the probability of a feature being selected by random chance. Against this background we have a hypothesis test formulated as follows,

$$H_0 : p_0 = p_1$$
$$H_1 : p_1 > p_0$$

where H_0 is the null hypothesis (that all features are equally relevant), and H_1 is the alternative hypothesis (that some features are more relevant than others). We select the Neyman-Pearson test for several reasons: (a) the likelihood functions under H_0 and H_1 can be explicitly computed as shown below, (b) the solution with the Neyman-Pearson lemma is a simple yet elegant result, and (c) perhaps most importantly, the Neyman-Pearson test is the most powerful test available for size α [9]. The Neyman-Pearson lemma states that we reject the null hypothesis if,

$$\Lambda(z) = \frac{\mathbb{P}(z|H_1)}{\mathbb{P}(z|H_0)} > \zeta_{\text{crit}} \tag{1}$$

where $\mathbb{P}(z|H_0)$ is the probability distribution under the null hypothesis, $\mathbb{P}(z|H_1)$ is the probability distribution under the alternative hypothesis, and ζ_{crit} is a threshold such that,

$$\mathbb{P}(T(z) > \zeta_{\text{crit}}|H_0) = \alpha \tag{2}$$

where α is size of the test, and $T(z)$ is the test-statistic. Using $\log\Lambda(z)$ would provide equivalent results since taking the logarithm does not affect the solution. Recall that the random variable Z follows a Binomial distribution. Using equation (1) and the form of the probability distribution on Z, we apply the Neyman–Pearson lemma:

$$\begin{aligned}\frac{\mathbb{P}(Z_n = z|H_1)}{\mathbb{P}(Z_n = z|H_0)} &= \frac{\binom{n}{z} p_1^z (1-p_1)^{n-z}}{\binom{n}{z} p_0^z (1-p_0)^{n-z}} \\ &= \left(\frac{1-p_1}{1-p_0}\right)^n \cdot \left(\frac{p_1(1-p_0)}{p_0(1-p_1)}\right)^z \\ &> \zeta_{\text{crit}}\end{aligned}$$

Since $\left(\frac{1-p_1}{1-p_0}\right)^n$ is simply a constant, which can be moved to the other side of the inequality, resulting in a new threshold ζ'_{crit}. Thus,

$$\left(\frac{p_1(1-p_0)}{p_0(1-p_1)}\right)^z > \zeta'_{\text{crit}}$$

Taking the logarithm gives us

$$z \log \left\{\frac{p_1(1-p_0)}{p_0(1-p_1)}\right\} > \zeta''_{\text{crit}}$$

where, again, the logarithm term is simply a constant and it can be removed to find a scaled threshold ζ'''_{crit}. Thus, we are seeking

$$z > \zeta'''_{\text{crit}}$$

where ζ_{crit} is a critical threshold determined by $\mathbb{P}(z > \zeta'''_{\text{crit}}|H_0) = \alpha$ (note by definition that z is a sufficient statistic for $T(z)$). Since the probability distribution on the null hypothesis is known (i.e., Binomial), we may explicitly solve for ζ'''_{crit}.

$$\mathbb{P}(z > \zeta'''_{\text{crit}}|H_0) = 1 - \underbrace{\mathbb{P}(z \leq \zeta'''_{\text{crit}}|H_0)}_{\text{cumulative distribution function}} = \alpha \tag{3}$$

Since $\mathbb{P}(z \leq \zeta'''_{\text{crit}}|H_0)$ has a closed form expression it can be obtained from a lookup table. Note that α can be used to control how conservative the hypothesis test will be. That is, if α is small, it will become more difficult for a feature to be detected as relevant because ζ'''_{crit} will become large. To summarize, NPFS is implemented as follows:

1. Run a FS algorithm $\mathcal{F}$ on n independently sampled data sets (sampling instances, not features). The independently sampled data sets can be a result of cross-validation or bootstrap samples. Form a matrix $\mathbf{X} \in \{0,1\}^{K \times n}$ where $\{\mathbf{X}\}_{il}$ is the Bernoulli random variable for feature i on trial l.

2. Compute ζ'''_{crit} using equation (3), which requires n, p_0, and the Binomial inverse cumulative distribution function.

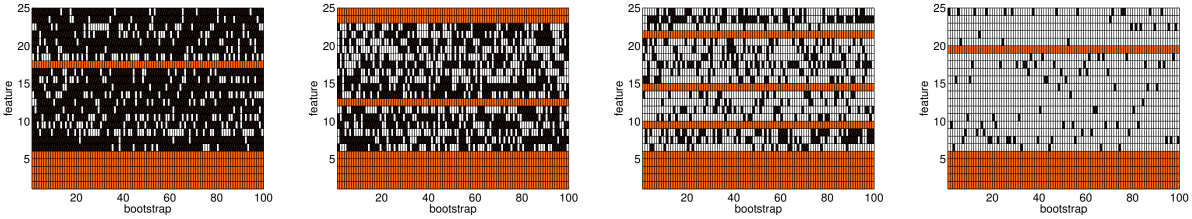
3. Let $\{\mathbf{z}\}_i = \sum_{l=1}^{n} \{\mathbf{X}\}_{il}$. If $\{\mathbf{z}\}_i > \zeta'''_{\text{crit}}$ then feature belongs in the relevant set, otherwise the feature is deemed non-relevant. Use only the features selected by the Neyman-Pearson detector for learning a classification or regression function.

3.3 Advantages of the Proposed Approach

The proposed method for post-analysis of FS offers several capabilities. Let us assume that k was selected to be too large compared to the true number of relevant features, k^*. How can we determine a more accurate value of k? The proposed approach provides a natural solution: simply use the features that Neyman-Pearson detector returns as being relevant. Note that the number of features returned by the Neyman-Pearson detector need not be k: if k were too large, we expect the test to return fewer relevant features. Having such an inference on k can reduce the complexity of the classifier or the regression function. We can also ask the opposite question: what if k – provided as a user-input to the FS algorithm – was selected too small? Could we apply this hypothesis test to determine the subset of K features that are relevant even though $\mathcal{F}$ never selects all of them because k was smaller than k^*? Our experiments, described in Section 4, test these conditions under controlled simulations as well as on data sets obtained from the UCI Machine Learning Repository.

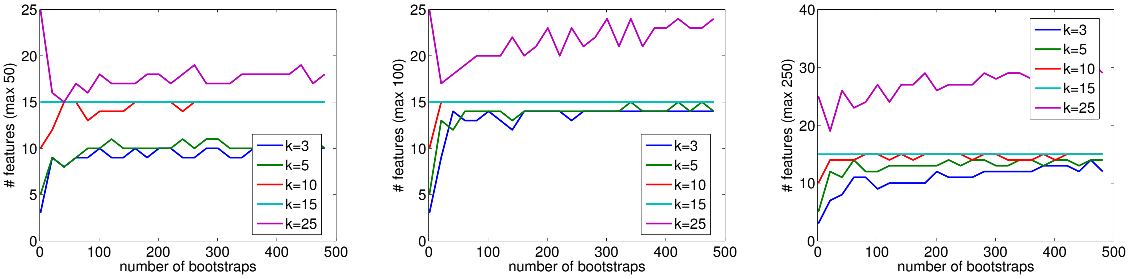
3.4 Upper Bound on Parameter Estimation

An important property of the proposed approach is that if $X_1, \ldots, X_n \sim \text{Bernoulli}(p)$, then we expect the difference between p and its bootstrap estimate $\hat{p}$ to become arbitrarily small as n grows large. The probability of the magnitude of difference between p and $\hat{p}$ being greater than some $\epsilon > 0$ can be upper bounded using Hoeffding's inequality.



(a) $k = 10$ (b) $k = 15$ (c) $k = 20$ (d) $k = 24$

Figure 1: Results of the Neyman-Pearson hypothesis test applied to the synthetic uniform data set for different cardinalities of the relevant feature set. The Neyman-Pearson hypothesis test recovers the original 5 relevant features (first 5 rows of each plot) with only a few additional irrelevant features in the set. This is a visualization of $\mathbf{X}$, where black segments indicate $X_l = 0$, white segments $X_l = 1$, and the orange rows are the features detected as relevant by the Neyman-Pearson test.



(a) $K = 50, k^* = 15$ (b) $K = 100, k^* = 15$ (c) $K = 250, k^* = 15$

Figure 2: Number of features selected by the Neyman-Pearson detector for varying levels of k (too large & too small) when there are 15 relevant features (k^*) in the synthetic data set. The number of features selected by the proposed approach appears to be converging to 15 when k is initially selected too small. Even though the number of selected features diverges when k is selected too big, they undershoot the original guess while the too small k's overshoot their original guesses.

Theorem 3.1 *(Hoeffding's Inequality [10]) Let $Y_1, Y_2, \ldots, Y_n$ be independent random observations such that $\mathbb{E}[Y] = \mu$, $\bar{Y} = \frac{1}{n}\sum_i Y_i$, and $a \leq Y_i \leq b$. For any $\epsilon > 0$, the following inequality holds,*

$$\mathbb{P}(|\bar{Y} - \mu| \geq \epsilon) \leq 2e^{-2n\epsilon^2/(b-a)^2} \tag{4}$$

Hoeffding's inequality is similar to that of Markov's inequality; however, it produces a tighter bound for larger deviations. We may use Hoeffding's inequality with a few assumptions to bound the differences between the bootstrap's estimate $\hat{p}$, and the true probability p. If $X_1, \ldots, X_n \sim \text{Bernoulli}(p)$, then for any $\epsilon > 0$, we have,

$$\mathbb{P}(|\hat{p} - p| \geq \epsilon) \leq 2\text{e}^{-2n\epsilon^2} \tag{5}$$

where $\hat{p} = \frac{1}{n}Z_n$. Thus if $X_1, \ldots, X_n \sim \text{Bernoulli}(p)$, then $\hat{p}$ approaches p exponentially quickly as a function of n. Chebyshev's inequality can also be used to find a bound on $\mathbb{P}(|\hat{p} - p| \geq \epsilon)$; however, Hoeffding's inequality provides a tighter upper bound for larger values of ϵ.

4 Experiments

Our proposed methodology for feature relevance using NPFS was implemented using *joint mutual information* (JMI) as the baseline FS objective function. In this section, we seek to determine the behavior of the hypothesis testing procedure through several experiments on synthetic and real-world data. We wish to answer the following questions:

1. Given a controlled data set, can NPFS correctly identify the truly relevant features?
2. If k were selected too large, can NPFS identify the subset of the k features that should be used instead of the set of k features?
3. If k were selected too small, can NPFS identify all the relevant features that could not be identified as relevant due to k being too small?

We provide a Matlab implementation of NPFS under the GNU GPLv3[1].

4.1 Data Sets and Testing Procedure

The proposed Neyman-Pearson hypothesis testing methodology (NPFS) for any given FS algorithm was tested on a synthetic data set, and a collection of data obtained from the UCI machine learning repository [11] (see Table 2). The synthetic data, described below, allows us to tailor experiments to test the strengths and weaknesses of the proposed approach.

4.1.1 Description of the Uniform Data

M observations are generated with features that are independently and identically distributed (iid) uniform random variables in the interval $[0, 10]$. This data set is referred to as $\mathcal{D}_{\text{uni}}$. Each feature vector $\mathbf{x}_m$ for $m \in [M]$ has K features. The true labeling function, unknown to any algorithm, is given by,

$$y_m = \begin{cases} 1, & \sum_{i=1}^{k^*} \mathbf{x}_m(i) \leq 5 \cdot k^* \\ 0, & \text{otherwise} \end{cases}$$

Hence, only the first k^* features carry information for determining the label y_m of a feature vector $\mathbf{x}_m$. Our goal is to identify, using our hypothesis test, those features (indices $i \in [k^*]$) that are relevant to the classification problem. Note that the threshold for determining the class label is the statistical expectation of the linear combination of the first k^* feature variables (this is easily shown using the properties of the expectation of a linear function). Such a threshold sets the prior probability on each of the classes to approximately $\frac{1}{2}$ for a randomly sampled data set.

There are n bootstrap data sets drawn from $\mathcal{D}_{\text{uni}}$, and the JMI feature selection algorithm is run independently on each sampled bootstrap set. k of K features are selected for each bootstrap data set, and a vector with binary indicators representing whether or not the feature was selected is produced. The n vectors form a $K \times n$ matrix with binary entries (i.e., $\mathbf{X}$). Each row, corresponding to a feature, is the sequence of Bernoulli experiments of success and failures used in NPFS.

4.2 Results on Synthetic Data Sets

Let us start with our questions on appropriate selection of k: if k is selected too large, can k^* be found such that $k^* < k$, and what is *approximately* the ideal value of k given the results from the n bootstraps? In this experiment, 5 features were considered relevant out of 25 features (recall that the features are uniform random variables). The value of k was varied from 10 to 24. For these cases, there are (at least) 5 to 19 irrelevant features are incorrectly selected as relevant at any given bootstrap iteration. We apply the Neyman-Pearson test after 100 bootstraps. Figure 1 shows that the Neyman-Pearson test can identify when irrelevant features are being selected by JMI. In this figure, the matrix $\mathbf{X}$ is visualized with white entries indicating features selected by JMI at different bootstrap iterations. The orange rows highlight the features that Neyman-Pearson method identifies as being relevant. Note that features $\{1, 2, 3, 4, 5\}$ are the only relevant features for this problem. Clearly, the inference provided by the Neyman-Pearson test allows us the ability to reduce k to achieve a much smaller subset of relevant features. In each of these experiments, we find that

[1] `http://github.com/EESI/NPFS`

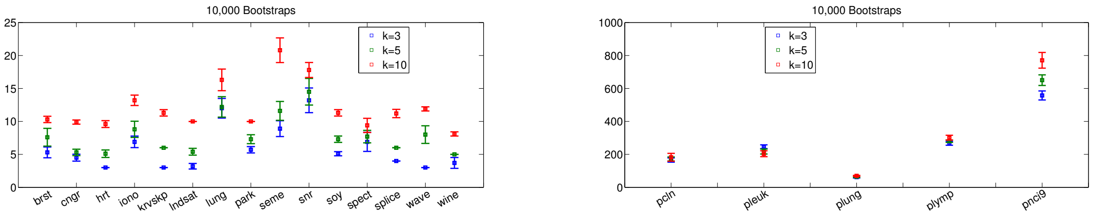


Figure 3: Variation in the Neyman-Pearson's test for the value of k^* given that k may have been selected too small. x-axis represents the data set under test and the y-axis is the predicted k^* by the proposed approach using 10,000 bootstraps.

there are a few features being detected as relevant, which are actually non-relevant. It is possible to tune n and α such that in every experiment only features 1 through 5 are being detected as relevant. In every experiment, however, the proposed method is always recommending the use of fewer features, because many of the features JMI selects at each bootstrap iteration are irrelevant.

The second key question is: can the value of k^* be recovered if k was initially chosen too small, and if so, how many bootstraps are needed? To examine this situation, three more synthetic uniform data sets were generated. All synthetic data sets' features are uniform random variables with 15 relevant features; however, the data set have 50, 100, or 250 features. We apply our Neyman-Pearson test with the number of bootstraps varying between 1 and 500. Furthermore, $k \in \{3, 5, 10, 15, 25\}$ are examined. Figure 2(a) shows that the value k^* selected by the Neyman-Pearson algorithm is approaching the true value for various selections of k. We should note that we can improve these results by increasing the number of observations in the data set. However, if k were too large, there are still a few features left in the relevant set as determined by the Neyman-Pearson detector (as observed previously in Figure 1). Figure 2(c) shows the effect of using 250 features rather than 50 features. Again, if k were selected too small, the Neyman-Pearson detector finds approximately k^* features; however, the method still unable to completely recover all of them with 500 bootstraps.

4.3 Results on UCI Data Sets

In this section, we present the classification error using a base classifier trained on: (i) all features, (ii) trained on the top 10 features selected by JMI, and (iii) trained on features selected by the proposed approach. The data sets are obtained from the UCI machine learning repository [11], and the Peng et al.'s mRMR paper [12]. The naïve Bayes (nb) and CART algorithms are used as baseline classifiers [13, 14]. We use the following notation to denote the classifier and the FS algorithm: nb (naïve Bayes trained on all features), nb-npfs (naïve Bayes trained with features identified by JMI and the proposed NPFS), and nb-jmi (top 10 features selected with JMI). It is important to note that we do not have access to the (true) k^* or the degree of feature relevancy for these data sets, therefore, we must examine the performance of a classifier to evaluate the methods effectiveness.

Table 2 presents each classifier's error and its rank (see [15]). The proposed approach for both the naïve Bayes and CART produces the best average rank. Unfortunately, there is not enough statistical evidence to suggest that the proposed approach provides uniformly the lowest error rate. There is, however, statistical significance between CART-NPFS and CART-JMI, with CART-NPFS out performing CART-JMI with an α-level of 0.1 using the Wilcoxon's signed rank test. The average number of features being selected by the Neyman-Pearson test after 10,000 bootstraps can be found in Figure 3. The UCI data sets do not allow us to control the level of feature relevancy as we did with the synthetic data and it is worth noting that we do not observe NPFS detecting all features as relevant even when the number of bootstraps is quite large.

4.4 Optical Character Recognition

Our final experiment uses the optical character recognition data set collected from UCI Machine Learning Repository. Each image in the experiment consists of 64 pixels represented by 4-bits (i.e., an 8×8 image); however, each image has been corrupted by adding nosiy pixels. The final image is 16×16. Just as before, we run 100 bootstrap trials with the JMI FS algorithm and apply the Neyman-Pearson hypothesis test. In this experiment $k = 64$ and $K = 256$. Each noisy pixel is sampled from a uniform probability mass function taking possible values $\{1, \dots, 16\}$.

Table 2: Classification errors of a Naïve Bayes and CART tested on the UCI data sets (see section 4.3) and rank after 10-fold cross validation. The errors in the table have been truncated; however, the ranks are determined via the untruncated values.

data set	**instances**	**features**	**nb**	**nb-jmi**	**nb-npfs**	**cart**	**cart-jmi**	**cart-npfs**
breast	569	30	0.069 (3)	0.055 (1.5)	0.055 (1.5)	0.062 (3)	0.056 (2)	0.041 (1)
congress	435	16	0.097 (3)	0.088 (1)	0.088 (2)	0.051 (3)	0.051 (1.5)	0.051 (1.5)
heart	270	13	0.156 (1)	0.163 (2)	0.174 (3)	0.244 (3)	0.226 (2)	0.207 (1)
ionosphere	351	34	0.117 (3)	0.091 (2)	0.091 (1)	0.077 (3)	0.068 (1)	0.074 (2)
krvskp	3196	36	0.122 (3)	0.108 (1)	0.116 (2)	0.006 (1)	0.056 (3)	0.044 (2)
landsat	6435	36	0.204 (1)	0.231 (2.5)	0.231 (2.5)	0.161 (1)	0.173 (2)	0.174 (3)
lungcancer	32	56	0.617 (3)	0.525 (1)	0.617 (2)	0.542 (2)	0.558 (3)	0.533 (1)
parkinsons	195	22	0.251 (3)	0.170 (1.5)	0.170 (1.5)	0.133 (1.5)	0.138 (3)	0.133 (1.5)
pengcolon	62	2000	0.274 (3)	0.179 (2)	0.164 (1)	0.21 (1)	0.226 (2.5)	0.226 (2.5)
pengleuk	72	7070	0.421 (3)	0.029 (1)	0.043 (2)	0.041 (2)	0.027 (1)	0.055 (3)
penglung	73	325	0.107 (1)	0.368 (3)	0.229 (2)	0.337 (1)	0.530 (3)	0.504 (2)
penglymp	96	4026	0.087 (1)	0.317 (3)	0.140 (2)	0.357 (3)	0.312 (2)	0.311 (1)
pengnci9	60	9712	0.900 (3)	0.600 (2)	0.400 (1)	0.667 (2)	0.617 (1)	0.783 (3)
semeion	1593	256	0.152 (1)	0.456 (3)	0.387 (2)	0.25 (1)	0.443 (3)	0.355 (2)
sonar	208	60	0.294 (3)	0.279 (2)	0.241 (1)	0.259 (2)	0.263 (3)	0.201 (1)
soybean	47	35	0.000 (2)	0.000 (2)	0.000 (2)	0.020 (2)	0.020 (2)	0.020 (2)
spect	267	22	0.210 (2)	0.206 (1)	0.232 (3)	0.187 (1)	0.210 (2)	0.229 (3)
splice	3175	60	0.044 (1)	0.054 (2)	0.055 (3)	0.085 (3)	0.070 (2)	0.066 (1)
waveform	5000	40	0.207 (3)	0.204 (2)	0.202 (1)	0.259 (3)	0.238 (2)	0.228 (1)
wine	178	13	0.039 (2.5)	0.039 (2.5)	0.034 (1)	0.079 (3)	0.068 (1.5)	0.068 (1.5)
average			2.275	1.900	1.825	2.075	2.1250	1.800

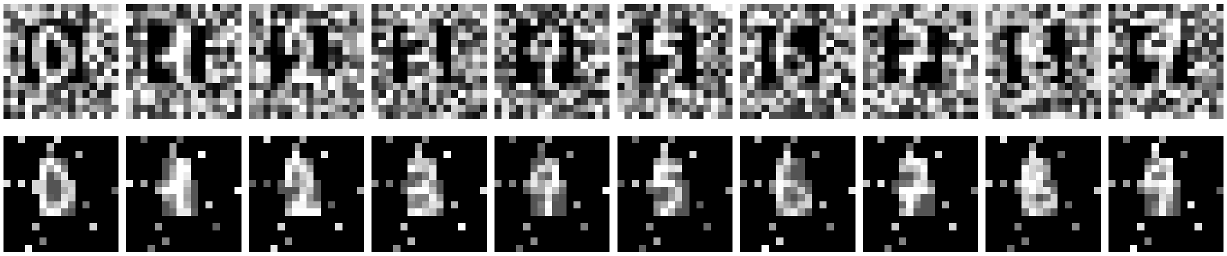


Figure 4: **Top row**: 16×16 image from the OCR data set corrupted with noisy pixels. The actual OCR images are 8×8 and take a 4-bit value. **Bottom row**: Irrelevant features marked by the Neyman-Pearson test are indicated in black. Note ONLY black pixels are irrelevant feature and not the "actual" value of the pixel (i.e., we have scaled the pixel to assure there were not black pixels). The Neyman-Pearson test selects a subset of 52 features in the 16×16 image that are relevant.

Figure 4 presents the NPFS results on OCR data set. The top row of Figure 4 shows the 16×16 images corrupted with noisy pixels. Note that the original OCR images can be observed as they are embedded within the noise. The bottom row of Figure 4 shows the irrelevant features marked in black by the Neyman-Pearson test. Note that only the black pixels are irrelevant features and not the "actual" value of the pixel (i.e., we have scaled the pixel to assure there were not black pixels). The Neyman-Pearson test selects a subset of 52 features in the 16×16 image that are relevant. Thus the Neyman-Pearson test is suggesting that there is a subset of features, fewer than 64, that are relevant for the discrimination between the characters in the image.

5 Conclusion

In this brief communication, we presented a wrapper methodology for validating the selection of k given a FS algorithm using the Neyman-Pearson hypothesis test – uniformly the most powerful hypothesis test. There are no assumptions made about the distribution of the data that the base FS algorithm would not already be making. The approach is easily integrated with existing FS methods, and can be used as a post-hoc test to determine the selection of the free parameter k was appropriate. We demonstrated, on synthetic data sets, that NPFS is cable of identify the correct number of relevant features even when the base-FS method does not select k^* features for each bootstrap, and that NPFS works well in practice on UCI data sets.

References

[1] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.

[2] R. Greiner, A. J. Grove, and A. Kogan, "Knowing what doesn't matter: Exploiting the omission of irrelevant data," *Artificial Intelligence*, vol. 97, pp. 345–380, 1997.

[3] G. Brown, A. Pocock, M.-J. Zhao, and M. Luján, "Conditional likelihood maximisation: A unifying framework for information theoretic feature selection," *Journal of Machine Learning Research*, vol. 13, pp. 27–66, 2012.

[4] J. Yang, Y. Liu, Z. Liu, X. Zhu, and X. Zhang, "A new feature selection algorithm based on binomial hypothesis testing for spam filtering," *Knowledge-Based Systems*, vol. 24, pp. 904–914, 2011.

[5] K. Kira and L. Rendell, "A practical approach to feature selection," in *National Conference on Artificial Intelligence*, 1992.

[6] H. Almuallim and T. Dietterich, "Efficient algorithms for identifying relevant features," in *Canadian Conference on Artificial Intelligence*, 1992.

[7] Y. Yang and J. Pedersen, "A comparative study on feature selection in text categorization," in *International Conference on Machine Learning*, 1997.

[8] L. I. Kuncheva, "A stability index for feature selection," in *International Conference on Artifical Intelligence and Application*, pp. 390–395, 2007.

[9] J. Neyman and E. Pearson, "On the problem of the most efficient tests of statistical hypotheses," *Philosophical Transactions of the Royal Society of London. Series A*, vol. 231, pp. 289–337, 1933.

[10] L. Wasserman, *All of Statistics: A concise course in Statistical Inference*. Springer, 2005.

[11] A. Frank and A. Asuncion, "UCI machine learning repository," 2010.

[12] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, 2005.

[13] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. John Wiley & Sons, Inc., 2nd ed., 2001.

[14] L. Breiman, J, Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. CRC Press, 1984.

[15] J. Demšar, "Statistical comparisions of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.