Supervised Feature Selection via Dependence Estimation

Le Song
NICTA and University of Sydney, Australia
lesong@it.usyd.edu.au

Alex Smola
SML, National ICT Australia
alex.smola@nicta.com.au

Arthur Gretton
MPI for Biological Cybernetics, Germany
arthur.gretton@tuebingen.mpg.de

Karsten Borgwardt
Ludwig-Maximilians-University, Germany
borgwardt@dbs.ifi.lmu.de

Abstract

We introduce a framework of feature filtering for supervised learning. It employs the Hilbert-Schmidt Independence Criterion (HSIC) as a measure of dependence between data and labels. The key idea is that good features should maximize such dependence. Feature selection for various supervised learning problems (including binary, multiclass and regression problems) can be unified under this framework, and the solution is approximated using a backward-elimination algorithm. Particularly, for binary problems, HSIC is also related to criteria such as Pearson's correlation, signal-to-noise ratio, Maximum Mean Discrepancy and the Kernel-Target Alignment. We conducted experiments on various real world data, which demonstrate the usefulness of this framework.

1 Introduction

The problem of supervised feature selection can be cast as a combinatorial optimisation problem. We have a full set of features, denoted $\mathcal{S}$ (each element in $\mathcal{S}$ corresponds to one dimension of the data). We use these features to predict a particular outcome, for instance the presence of cancer: clearly, only a subset $\mathcal{T}$ of features will be relevant. Suppose the relevance of a feature subset to the outcome is quantified by $\mathcal{Q}(\mathcal{T})$ and it is computed by restricting the data to the dimensions in $\mathcal{T}$. Feature selection can then be formulated as:

$$\mathcal{T}_0 = \arg\max_{\mathcal{T}\subseteq\mathcal{S}} \mathcal{Q}(\mathcal{T}) \qquad \text{s.t.} \quad |\mathcal{T}| \leq t \tag{1}$$

where $|\cdot|$ computes the cardinality of a set and t upper bounds the number of selected features. Two important aspects of problem (1) are the choice of the criterion $\mathcal{Q}(\mathcal{T})$ and the selection algorithm. We therefore begin with a description of our criterion, and later introduce the feature selection algorithm based on this criterion.

2 Feature Selection Criterion: HSIC

We define $\mathcal{X}$ and $\mathcal{Y}$ broadly as two domains from which we draw samples (x, y): these may be real valued, vector valued, class labels, strings [1], graphs [2], and so on (see [3] for further examples in bioinformatics). We define a (possibly nonlinear) mapping $\phi(x) \in \mathcal{F}$ from each $x \in \mathcal{X}$ to a feature space $\mathcal{F}$, such that the inner product between the features is given by a kernel function $k(x, x') := \langle \phi(x), \phi(x') \rangle$: $\mathcal{F}$ is called a reproducing kernel Hilbert space (RKHS). Likewise, let $\mathcal{G}$ be a second RKHS on $\mathcal{Y}$ with kernel $l(\cdot, \cdot)$ and feature map $\psi(y)$. We may now define a cross-covariance operator between these feature maps, which is a linear operator $\mathcal{C}_{xy} : \mathcal{G} \longmapsto \mathcal{F}$ such that

$$\mathcal{C}_{xy} = \mathsf{E}_{xy}[(\phi(x) - \mu_x) \otimes (\psi(y) - \mu_y)], \tag{2}$$

where $\otimes$ is the tensor product (see [4] for more detail). The square of the Hilbert-Schmidt norm of the cross-covariance operator (HSIC), $\|\mathcal{C}_{xy}\|^2_{\mathrm{HS}}$, is then used as our feature selection criterion $\mathcal{Q}(\mathcal{T})$.

HSIC was shown in [5] to be expressible in terms of kernels as

$$\begin{aligned}\mathrm{HSIC}(\mathcal{F},\mathcal{G},\Pr_{xy}) = \|\mathcal{C}_{xy}\|^2_{\mathcal{H}} = \mathsf{E}_{xx'yy'}[k(x,x')l(y,y')] + \mathsf{E}_{xx'}[k(x,x')]\mathsf{E}_{yy'}[l(y,y')] \\ - 2\mathsf{E}_{xy}[\mathsf{E}_{x'}[k(x,x')]\mathsf{E}_{y'}[l(y,y')]].\end{aligned} \tag{3}$$

2.1 Unbiased Estimator of HSIC

Given a sample $Z = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ of size m drawn from $\Pr_{xy}$, an empirical estimator of HSIC was shown in [5] to be

$$\mathrm{HSIC}(\mathcal{F},\mathcal{G},Z) = (m-1)^{-2}\mathsf{Tr}(\mathbf{KHLH}) \tag{4}$$

where $\mathbf{K}, \mathbf{L} \in \mathbb{R}^{m\times m}$ are the kernel matrices for the data and the labels respectively, and $\mathbf{H}_{ij} = \delta_{ij} - m^{-1}$ centres the data and the label features. See [6] for a different interpretation of a related criterion used in independence testing. Note that the estimator in equation (4) is biased. Here we derive an unbiased estimator of HSIC as

$$\mathrm{HSIC}(\mathcal{F},\mathcal{G},Z) = \frac{1}{m(m-3)}\left[\mathsf{Tr}(\mathbf{KL}) + \frac{1}{(m-1)(m-2)}\mathbf{1}^\top\mathbf{K}\mathbf{1}\mathbf{1}^\top\mathbf{L}\mathbf{1} - \frac{2}{m-2}\mathbf{1}^\top\mathbf{K}\mathbf{L}\mathbf{1}\right] \tag{5}$$

where the diagonal entries of $\mathbf{K}$ and $\mathbf{L}$ are set to zeros to obtain an unbiased estimator. We now present two theorems which support our using HSIC as a feature selection criterion (the proofs are presented in the appendix). They shows the unbiasedness and the concentration of the empirical HSIC in (5). This implies that if the empirical HSIC is large, then given sufficient samples it is very probable that the population HSIC is also large; likewise, a small empirical HSIC likely corresponds to a small population HSIC. Moreover, the same features should consistently be selected to achieve high dependence if the data is repeatedly drawn from the same distribution.

Theorem 1 (Unbiasedness of Estimator) *Let E_Z denote the expectation taken over m independent observations (x_i, y_i) drawn from $\Pr_{xy}$. Then*

$$\mathrm{HSIC}(\mathcal{F},\mathcal{G},\Pr_{xy}) = \mathsf{E}_Z\left[\mathrm{HSIC}(\mathcal{F},\mathcal{G},Z)\right].$$

Theorem 2 (Concentration of Estimator) *Assume that $k(\cdot,\cdot)$ and $l(\cdot,\cdot)$ are bounded almost everywhere by 1, and are non-negative. Then for $m > 1$ and all $\delta > 0$, with probability at least $1-\delta$, for all $\Pr_{xy}$,*

$$|\mathrm{HSIC}(\mathcal{F},\mathcal{G},Z) - \mathrm{HSIC}(\mathcal{F},\mathcal{G},\Pr_{xy})| \leq \sqrt{\tfrac{\log(6/\delta)}{\alpha^2 m}},\ \alpha > 0.28.$$

Furthermore, when $\mathcal{F}, \mathcal{G}$ RKHSs with universal [7] kernels k, l on respective compact domains $\mathcal{X}$ and $\mathcal{Y}$, then $\mathrm{HSIC}(\mathcal{F},\mathcal{G},\Pr_{xy}) = 0$ if and only if x and y are independent [8, Theorem 4].

In terms of our feature selection setting, using a universal kernel such as the Gaussian RBF kernel or the Laplace kernel, HSIC is zero if feature values and class labels are independent; clearly we want to reach the opposite result, namely strong dependence between feature values and class labels. Hence we try to select features that maximise HSIC.

2.2 Variance of the Unbiased Estimator of HSIC

Sometimes, it is useful to assess the significance of the functional dependence between data and labels before actually learning it. We show that HSIC is asymptotically normal and derive its variance, so that statistics can be formulated for a significance test. First, we express HSIC using the U-statistics:

$$\mathrm{HSIC} = \frac{(m-4)!}{m!}\sum_{(i,j,q,r)\in\mathbf{i}_4^m}^{m} h(i,j,q,r), \tag{6}$$

where $\mathbf{i}_r^m$ denotes the set of all m-tuples drawn without replacement from $\{1,\ldots,m\}$, and h is the kernel of the U-statistics defined by:

$$\begin{aligned}h(i,j,q,r) =& \frac{1}{6}[\mathbf{K}_{ij}(\mathbf{L}_{ij}+\mathbf{L}_{qr}) + \mathbf{K}_{iq}(\mathbf{L}_{iq}+\mathbf{L}_{jr}) + \mathbf{K}_{ir}(\mathbf{L}_{ir}+\mathbf{L}_{jq}) + \mathbf{K}_{jq}(\mathbf{L}_{jq}+\mathbf{L}_{ir}) \\ &+ \mathbf{K}_{jr}(\mathbf{L}_{jr}+\mathbf{L}_{qi}) + \mathbf{K}_{qr}(\mathbf{L}_{qr}+\mathbf{L}_{ij})] - \frac{1}{12}\sum_{(t,u,v)}^{(i,j,q.r)}\mathbf{K}_{tu}[\mathbf{L}_{tv}+\mathbf{L}_{uv}]\end{aligned} \tag{7}$$

Note that the sum in (7) represents all ordered triples (t, u, v) selected without replacement from (i, j, q, r). Then according to [9], HSIC is asymptotically normal with estimated variance:

$$\sigma^2_{\mathrm{HSIC}} = \frac{16}{m}\left(R - \mathrm{HSIC}^2\right) \quad \text{where} \quad R = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{(m-3)!}{m!}\sum_{(j,q,r)\in \mathbf{i}_3^m \setminus \{i\}} h(i,j,q,r)\right)^2. \quad (8)$$

3 Feature Selection Algorithm: Backward Elimination

BAHSIC Having defined our feature selection criterion, we now describe an algorithm that conducts feature selection on the basis of this dependence measure. Using HSIC, we can perform both forward and backward selection of the features. In particular, when we use a linear kernel on the data and labels, forward selection and backward selection are equivalent. However, although forward selection is computationally more efficient, backward elimination in general yields better features, since the quality of the features is assessed within the context of all other features. Hence we present the backward elimination (BA) version of our algorithm here.

Our feature selection algorithm (BAHSIC) appends the features from $\mathcal{S}$ to the end of a list $\mathcal{S}^\dagger$ so that the elements towards the end of $\mathcal{S}^\dagger$ have higher relevance to the learning task. The feature selection problem in (1) can be solved by simply taking the last t elements from $\mathcal{S}^\dagger$. Our algorithm produces $\mathcal{S}^\dagger$ using a backward elimination procedure. It proceeds recursively, eliminating the least relevant features from $\mathcal{S}$ and adding them to the end of $\mathcal{S}^\dagger$ in each iteration.

Algorithm 1 Feature Selection via Backward Elimination

Input: The full set of features $\mathcal{S}$
Output: An ordered set of features $\mathcal{S}^\dagger$

1: $\mathcal{S}^\dagger \leftarrow \varnothing$
2: **repeat**
3: $\quad \sigma_0 \leftarrow \arg\max_\sigma \ \mathrm{HSIC}(\sigma, \mathcal{S}), \ \ \sigma \in \Xi$
4: $\quad i \leftarrow \arg\max_i \ \mathrm{HISC}(\sigma_0, \mathcal{S} \setminus \{i\}), \ \ i \in \mathcal{S}$
5: $\quad \mathcal{S} \leftarrow \mathcal{S} \setminus \{i\}$
6: $\quad \mathcal{S}^\dagger \leftarrow \mathcal{S}^\dagger \cup \{i\}$
7: **until** $\mathcal{S} = \varnothing$

Step 3 of the algorithm optimises over all possible choices of kernel parameters in the set Ξ. Note that Ξ is chosen such that the kernels are bounded. If we have no prior knowledge regarding the nature of the nonlinearity in the data, then optimising over Ξ is essential: it allows us to adapt to the scale of the nonlinearity present in the (feature-reduced) data. If we have prior knowledge about the type of nonlinearity, we can use a kernel with fixed parameters for BAHSIC. In this case, step 3 can be omitted since there will be no parameter to tune. For faster elimination of features, we can choose a group of features at step 4 and delete them in one shot at step 5.

4 Connection to Other Approaches

In this section we will show that several feature selection criteria are special cases of BAHSIC, and thus BAHSIC is capable of finding and exploiting dependence of a much more general nature (for instance, dependence between data and labels with graph and string values).

We first define the symbols used in the following subsections. Let $\mathbf{X}$ be the full data matrix with each row being a sample and each column a feature, $\mathbf{x}$ be a column of $\mathbf{X}$, and x_i be the entries in $\mathbf{x}$. Let $\mathbf{y}$ be the vector of labels with entries y_i. When the labels are multidimensional, we express them as a matrix $\mathbf{Y}$, with each row for a datum and each column for a dimension. The kth column of $\mathbf{Y}$ is then $\mathbf{Y}(k)$.

Suppose the number of data points is m. We denote the mean of a particular feature of the data as $\bar{x}$, and its standard deviation as s_x. For two-class data, let the number of the positive and negative samples be m_+ and m_-, respectively ($m = m_+ + m_-$). In this case, denote the mean of the samples from the positive and the negative classes by $\bar{x}_+$ and $\bar{x}_-$, respectively, and the corresponding standard deviations by s_{x+} and s_{x-}. For multiclass data, we let m_i be the number of samples in class i, where $i \in \mathbb{N}^*$ and $m = \sum_i m_i$. Finally, let $\mathbf{1}_k$ be a column vector of all ones with length k and $\mathbf{0}_k$ be a column vector of all zeros.

4.1 Maximum Mean Discrepancy and Kernel Target Alignment

For binary classification, an alternative criterion for selecting features is to check whether the distributions $\Pr(x|y=1)$ and $\Pr(x|y=-1)$ differ. For this purpose one could use Maximum Mean Discrepancy (MMD) [10]. Likewise, one could use Kernel Target Alignment (KTA) [11] to test directly whether there exists any correlation between data and labels.[1] For the kernel $l(y,y') = \rho(y)\rho(y')$, where $\rho(y) = \frac{1}{m_+}$ for $y = 1$ and $\frac{-1}{m_-}$ otherwise, Theorem 3 shows that $(m-1)^{-2}$MMD and $(m-1)^{-2}$KTA are closely related to HSIC (see appendix for the proof). Under certain conditions, both of them converges to HSIC with rate $1/m$. Note that, however, HSIC is a more general criterion and not equal to $(m-1)^{-2}$MMD or $(m-1)^{-2}$KTA in other cases.

Theorem 3 (Connection to MMD and KTA) *Assume the kernel $k(x,x')$ for the data is bounded and the kernel for the labels is $l(y,y') = \rho(y)\rho(y')$. Then*

$$\left|\mathrm{HSIC} - (m-1)^{-2}\mathrm{MMD}\right| = \mathcal{O}(m^{-1}) \text{ and } \left|\mathrm{HSIC} - (m-1)^{-2}\mathrm{KTA}\right| = \mathcal{O}(m^{-1}). \tag{9}$$

4.2 Pearson Correlation

Pearson's correlation is commonly used in microarray analysis [13, 14], and is defined as $r_{xy} = (\sum_{i=1}^m (x_i - \bar{x})(y_i - \bar{y}))/(s_x s_y)$, for each column $\mathbf{x}$ of $\mathbf{X}$ (scores are computed separately for each feature). The link between HSIC and Pearson's correlation is straightforward: we first normalise the data and the labels by s_x and s_y respectively, and apply a linear kernel in both domains. HSIC then becomes

$$\begin{aligned}\mathsf{Tr}(\mathbf{KHLH}) &= \mathsf{Tr}(\mathbf{x}\mathbf{x}^\top \mathbf{H}\mathbf{y}\mathbf{y}^\top \mathbf{H}) = ((\mathbf{H}\mathbf{x})^\top(\mathbf{H}\mathbf{y}))^2 \\ &= \left(\sum_{i=1}^m \left(\frac{x_i}{s_x} - \frac{\bar{x}}{s_x}\right)\left(\frac{y_i}{s_y} - \frac{\bar{y}}{s_y}\right)\right)^2 = \left(\frac{\sum_{i=1}^m (x_i - \bar{x})(y_i - \bar{y})}{s_x s_y}\right)^2.\end{aligned} \tag{10}$$

The above equation is just the square of Pearson's correlation (pc).

4.3 Mean Difference and its Variants

The difference between the sample means of the positive and negative classes, $(\bar{x}_+ - \bar{x}_-)$, is useful for selecting discriminative features. With different normalisation of the data and the labels, many variants can be derived. For example, the centroid [15](lin), t-statistic (t), moderated t-score (m-t), signal-to-noise ratio (snr), and B-statistics (lods) [16] all belong to this subfamily.

We will start by showing that $(\bar{x}_+ - \bar{x}_-)^2$ is a special case of HSIC. This is straightforward if we assign $\frac{1}{m_+}$ as the labels to the positive samples and $\frac{-1}{m_-}$ to the negative samples. Applying a linear kernel on both domains leads to the equivalence

$$\begin{aligned}\mathsf{Tr}(\mathbf{KHLH}) &= \mathsf{Tr}(\mathbf{x}\mathbf{x}^\top \mathbf{y}\mathbf{y}^\top) = (\mathbf{x}^\top \mathbf{y})^2 \\ &= \left(\frac{1}{m_+}\sum_{i=1}^{m_+} x_i - \frac{1}{m_-}\sum_{i=1}^{m_-} x_i\right)^2 = (\bar{x}_+ - \bar{x}_-)^2.\end{aligned} \tag{11}$$

Note that the centring matrix $\mathbf{H}$ disappears because the labels are already centred (i.e. $\mathbf{y}^\top \mathbf{1}_m = 0$, and thus $\mathbf{HLH} = \mathbf{L}$).

The t-statistic is defined as $t = \frac{\bar{x}_+ - \bar{x}_-}{\bar{s}}$, where $\bar{s} = (\frac{s_{x+}^2}{m_+} + \frac{s_{x-}^2}{m_-})^{\frac{1}{2}}$. The square of the t-statistic is equivalent to HSIC if the data is normalised by $(\frac{s_{x+}^2}{m_+} + \frac{s_{x-}^2}{m_-})^{\frac{1}{2}}$. The signal-to-noise ratio, moderated t-statistic, and B-statistic are three variants of the t-test. They differ only in their respective denominators, and are thus special cases of HSIC if we normalise the data accordingly. For example, we obtain the signal-to-noise ratio if the data is normalised by $(s_{x+} + s_{x-})$.

4.4 Shrunken Centroid

The shrunken centroid (pam) method [17, 18] performs feature ranking using the differences from the class centroids to the centroid of all the data. This is also related to HSIC if specific preprocessing of the data and labels is performed. Here we will focus on constructing appropriate labels, as the

[1] KTA, $\langle \mathbf{K}, \mathbf{L}\rangle_{\mathbf{F}} := \mathsf{Tr}(\mathbf{KL})$ has been used for feature selection in [12]. For computational convenience, the normalization, $\sqrt{\langle \mathbf{K}, \mathbf{K}\rangle_F \langle \mathbf{L}, \mathbf{L}\rangle_F}$, is omitted in practice. It is the unnormalized KTA that we will discuss.

normalisation of the data is similar to the previous section. For two-class problems, we use the 2-dimensional label matrix

$$\mathbf{Y} = \begin{pmatrix} \frac{\mathbf{1}_{m_+}}{m_+} - \frac{\mathbf{1}_{m_+}}{m}, & -\frac{\mathbf{1}_{m_+}}{m} \\ -\frac{\mathbf{1}_{m_-}}{m}, & \frac{\mathbf{1}_{m_-}}{m_-} - \frac{\mathbf{1}_{m_-}}{m} \end{pmatrix}_{m \times 2}. \tag{12}$$

The labels are centred (i.e. $\mathbf{Y}^\top \mathbf{1_m} = \mathbf{0}_2$), and thus

$$\begin{aligned} \mathsf{Tr}(\mathbf{KHLH}) &= \mathsf{Tr}(\mathbf{x}\mathbf{x}^\top \mathbf{Y}\mathbf{Y}^\top) = \mathbf{Y}(1)^\top \mathbf{x}\mathbf{x}^\top \mathbf{Y}(1) + \mathbf{Y}(2)^\top \mathbf{x}\mathbf{x}^\top \mathbf{Y}(2) \\ &= \left(\frac{1}{m_+} \sum_{i=1}^{m_+} x_i - \frac{1}{m} \sum_{i=1}^{m} x_i \right)^2 + \left(\frac{1}{m_-} \sum_{i=1}^{m_-} x_i - \frac{1}{m} \sum_{i=1}^{m} x_i \right)^2 \\ &= (\bar{x}_+ - \bar{x})^2 + (\bar{x}_- - \bar{x})^2 \end{aligned} \tag{13}$$

This is in essence the shrunken centroid method.

4.5 Multiclass

In addition to scoring features for two-class data, our method can readily be applied to multiclass data, by constructing an appropriate label space kernel using the class label assignments. For instance, we can score a feature for the multiclass classification problem by applying linear kernels to the following label feature vectors (3-class example):

$$\mathbf{Y} = \begin{pmatrix} \frac{\mathbf{1}_{m_1}}{m_1} & \frac{\mathbf{1}_{m_1}}{m_2 - m} & \frac{\mathbf{1}_{m_1}}{m_3 - m} \\ \frac{\mathbf{1}_{m_2}}{m_1 - m} & \frac{\mathbf{1}_{m_2}}{m_2} & \frac{\mathbf{1}_{m_2}}{m_3 - m} \\ \frac{\mathbf{1}_{m_3}}{m_1 - m} & \frac{\mathbf{1}_{m_3}}{m_2 - m} & \frac{\mathbf{1}_{m_3}}{m_3} \end{pmatrix} \quad \text{or} \quad \mathbf{Y} = \begin{pmatrix} \frac{\mathbf{1}_{m_1}}{\sqrt{m_1}} & \mathbf{0}_{m_1} & \mathbf{0}_{m_1} \\ \mathbf{0}_{m_2} & \frac{\mathbf{1}_{m_2}}{\sqrt{m_2}} & \mathbf{0}_{m_2} \\ \mathbf{0}_{m_3} & \mathbf{0}_{m_3} & \frac{\mathbf{1}_{m_3}}{\sqrt{m_3}} \end{pmatrix} \tag{14}$$

The $\mathbf{Y}$ on the left is equivalent to one-versus-the-rest scoring of the features, while that on the right is geared towards selecting features that recover the block structure of the kernel matrix in the data space.

4.6 Regression

BAHSIC can also be used to select features for regression problems, except that in this case the labels are continuous variables. Again we can use different kernels on both the data and the labels and apply BAHSIC. In this context, feature selection using ridge regression can also be viewed as a special case of BAHSIC. In ridge regression [19], we predict the outputs $\mathbf{y}$ using the predictor $\mathbf{Vw}$ by minimising the objective function $R = (\mathbf{y} - \mathbf{Vw})^2 + \lambda \|\mathbf{w}\|^2$, where the second term is known as the regulariser. Our discussion encompasses two cases: first, the linear model, in which $\mathbf{V} = \mathbf{X}$; and second, the nonlinear case, in which each of the m rows of $\mathbf{V}$ is a vector of nonlinear features of a particular observation x_i, and $f(x_i) = \sum_j w_j v_j(x_i)$. Recursive feature elimination combined as an embedded method with ridge regression removes the feature which causes the smallest increase in R. Equivalently, after minimising R, this is the feature which has the smallest absolute weight $|w_i|$.

The minimum of this objective function with respect to $\mathbf{w}$ is

$$R^* = \mathbf{y}^\top \mathbf{y} - \mathbf{y}^\top \mathbf{V}(\mathbf{V}^\top \mathbf{V} + \lambda \mathbf{I})^{-1} \mathbf{V}^\top \mathbf{y} = \mathbf{y}^\top \mathbf{y} - \mathsf{Tr}(\mathbf{V}(\mathbf{V}^\top \mathbf{V} + \lambda \mathbf{I})^{-1} \mathbf{V}^\top \mathbf{y}\mathbf{y}^\top) \tag{15}$$

Therefore recursively removing the feature which minimises the increase in R^* is equivalent to maximising the HSIC, when using $\mathbf{K} = \mathbf{V}(\mathbf{V}^\top \mathbf{V} + \lambda \mathbf{I})^{-1} \mathbf{V}^\top$ as the kernel matrix on the data and the linear kernel on the labels.

The final case we consider is kernel ridge regression, which differs from the above in that the space of nonlinear features of the input may be infinite dimensional, and the regulariser becomes a smoothness constraint on the functions from this space to the output. Specifically, the inputs are mapped to a *different* feature space $\mathcal{H}$ with kernel $\hat{k}(x, x')$, in which a linear prediction is made of the label y. Without going into further detail, we use standard kernelisation methods [20] to obtain that the minimum objective is $R^* = \mathbf{y}^\top \mathbf{y} - \mathbf{y}^\top (\hat{\mathbf{K}} + \lambda \mathbf{I})^{-1} \hat{\mathbf{K}} \mathbf{y}$. This is equivalent to defining a feature space $\mathcal{F}$ with kernel $(\hat{\mathbf{K}} + \lambda \mathbf{I})^{-1} \hat{\mathbf{K}}$ on the data, and then selecting features by maximising HSIC.

5 Experimental Results

We conduct experiments to test BAHSIC family of feature selection methods. These experiments use real world data ranging from binary classification, multiclass classification to regression. In these experiments, the BAHSIC family of methods demonstrates excellent performance.

5.1 Microarray Data

We ran our experiments on 28 gene expression datasets, 15 of which are two-class datasets and 13 are multiclass datasets. These datasets are assigned a reference number for convenience. Two-class datasets have a reference number less than or equal to 15, and multi-class datasets have reference numbers of 16 and above. Only one dataset (ref no.19) has feature dimension less than 1000 (79 features). All other datasets have dimensions ranging from approximately 2000 to 25000. The number of samples varies depending on datasets between approximately 50 and 300 samples. A summary of the datasets and their sources is in the appendix.

In addition to the BAHSIC family of feature selection algorithms, we compare against three methods that are not members of the BAHSIC family: mutual information (mi), recursive feature elimination SVM (rfe) [21], and L1-SVM for feature selection (l1) [22].

We used stratified 10-fold cross-validation and SVMs to evaluate the predictive performance of the top 10 features selected by each method. For two-class datasets, a non-linear SVM with an RBF kernel, $k(x, x') = \exp(-\frac{\|x-x'\|^2}{2\sigma^2})$, was used. The regularisation constant C and the kernel width σ were tuned on a grid of $\{0.1, 1, 10, 10^2, 10^3\} \times \{1, 10, 10^2, 10^3\}$. Classification performance is measured as the fraction of misclassified samples. For multiclass datasets, all procedures are the same except that we used the SVM in a one-versus-the-rest fashion. Two new BAHSIC methods are included in the comparison, with kernels $\exp(-\frac{\|x-x'\|}{2\sigma^2})$ (rbf) and $\|x - x'\|^{-1}$ (dis) on the data.

The classification results for binary and multiclass datasets are reported in Table 1 and Table 2, respectively. In addition to error rate we also report the overlap between the top 10 gene lists created in each fold. The tables are presented separately as some older members of the BAHSIC family, and some competitors, are not naturally extensible to multiclass datasets. From the experiments we make the following observations:

1. The BAHSIC family obtains the lowest classification error (not necessarily significant) in 12 out of 15 of the two-class datasets and all 13 of the multiclass datasets.
2. The BAHSIC family obtains the greatest overlap in all but one datasets. This suggests that genes selected by the BAHSIC family can be more stable.
3. The BAHSIC family with non-linear kernels obtains the lowest classification error in 7 datasets and the greatest overlap in 7 datasets.

5.2 Brain-computer Interface Data

In this experiment, BAHSIC is used to select frequency band for a brain-computer interface (BCI) dataset (IVa) from the Berlin BCI group [23]. This dataset contains EEG signals (118 channels, sampled at 100 Hz) from five healthy subjects ('aa', 'al', 'av', 'aw' and 'ay') recorded during two types of motor imaginations. The task is to classify the imagination for individual trial.

Our experiment proceeds in 3 steps: (**I**) Fast Fourier transformation (FFT) is performed on each channel and the power spectrum is computed. (**II**) The power spectra from all channels are averaged to obtain a single spectrum for each trial. (**III**) BAHSIC is used to select the top 5 discriminative frequency components based on the power spectrum. The selected 5 frequencies and their 4 nearest neighbors are used to reconstruct the temporal signals (with all other Fourier coefficients set to zeros). This is equivalent to bandpass filter the EEG signals. The resulting signals are then passed to normal CSP method for feature extraction and then classified using linear SVM.

Automatic filtering using BAHSIC is then compared to other filtering approaches: normal CSP method with manual filtering (8-40 Hz), the CSSP method [24] and the CSSSP method [25]. All results presented in Table 3 are obtained using 50×2-fold cross-validation. Our method is very competitive and obtains the first and second place for 4 of the 5 subjects. While the CSSP and the CSSSP methods are embedded methods (into the CSP method) for frequency selection. Our method decouples from the CSP method completely but still contributes to it positively.

In Figure 1, we use HSIC to visualize the responsiveness of different frequency bands to motor imagination. The horizontal and the vertical axes in each subfigure represent the lower and the upper bounds for a frequency band respectively. HSIC is computed for each of these bands. It is reported earlier that μ rhythm (around 12 Hz) of EEG is most responsive to motor imagination, and β rhythm (around 22 Hz) is also responsive [25]. We expect that HSICs will create a strong peaks at μ rhythm and a weaker peak at β rhythm, and the absence of other responsive frequency components

Table 1: Two-class datasets: classification error (%) and number of common genes (overlap) for 10-fold cross-validation using the top 10 selected features. Each *row* shows the results for a dataset, and each *column* is a method. Each entry in the table contains two numbers separated by "|": the first number is the classification error and the second number is the number of overlaps. Best results are in bold. The last *row* summarises the number of times a method was the best. The last *column* shows in which datasets the BAHSIC family obtains the best performance (indicated by a ✓).

	BAHSIC family									**Others**			
Ref.#	**pc**	**snr**	**pam**	**t**	**m-t**	**lods**	**lin**	**rbf**	**dis**	**rfe**	**l1**	**mi**	
1	12.7|3	11.4|3	11.4|**4**	12.9|3	12.9|**4**	12.9|**4**	15.5|3	19.1|1	13.9|2	14.3|0	**7.7**|0	26.1|0	-|✓
2	33.2|1	33.9|2	33.9|1	29.5|1	29.5|1	**27.8**|1	32.9|2	31.5|**3**	32.8|2	34.2|0	32.5|1	29.9|0	✓|✓
3	37.4|0	37.4|0	37.4|0	**34.6**|**6**	**34.6**|**6**	**34.6**|**6**	37.4|1	37.4|0	37.4|0	37.4|0	37.4|0	36.4|0	✓|✓
4	41.6|0	38.8|0	41.6|0	40.7|**1**	40.7|0	**37.8**|0	41.6|0	41.6|0	39.7|0	41.6|0	41.6|0	40.6|0	✓|✓
5	27.8|0	**26.7**|0	27.8|0	**26.7**|**2**	**26.7**|**2**	**26.7**|**2**	27.8|0	27.8|0	27.6|0	27.8|0	27.8|0	27.8|0	✓|✓
6	30.0|2	**25.0**|0	31.7|0	**25.0**|**5**	**25.0**|**5**	**25.0**|**5**	30.0|0	31.7|0	30.0|1	30.0|0	33.3|0	33.3|0	✓|✓
7	**2.0**|**6**	**2.0**|5	**2.0**|5	28.7|4	26.3|4	26.3|4	**2.0**|3	**2.0**|4	30.0|0	**2.0**|0	**2.0**|0	**2.0**|2	✓|✓
8	3.3|3	**0.0**|4	**0.0**|4	**0.0**|4	3.3|**6**	3.3|**6**	3.3|2	3.3|1	6.7|2	**0.0**|0	3.3|0	6.7|1	✓|✓
9	10.0|**6**	10.0|**6**	**8.7**|4	34.0|5	37.7|**6**	37.7|**6**	12.0|3	10.0|5	12.0|1	10.0|0	17.0|1	12.0|3	✓|✓
10	16.0|2	18.0|2	**14.0**|2	**14.0**|8	22.0|**9**	22.0|**9**	16.0|2	16.0|0	18.0|0	32.5|0	**14.0**|0	20.5|1	✓|✓
11	12.9|5	12.9|5	12.9|5	19.5|0	22.1|0	33.6|0	11.2|4	**9.5**|**6**	16.0|4	19.0|0	17.4|0	11.2|4	✓|✓
12	30.3|2	36.0|2	31.3|2	26.7|**3**	35.7|0	35.7|0	**18.7**|1	35.0|0	33.0|1	29.7|0	30.0|0	23.0|2	✓|✓
13	8.4|5	11.1|0	7.0|5	22.1|3	27.9|**6**	15.4|1	7.0|2	9.6|0	11.1|0	**4.3**|1	5.5|2	7.0|4	-|✓
14	20.8|1	20.8|1	20.2|0	20.8|**3**	20.8|**3**	20.8|**3**	20.8|0	20.2|0	19.7|0	20.8|0	20.8|1	**19.1**|1	-|✓
15	**0.0**|7	0.7|1	**0.0**|5	4.0|1	0.7|**8**	0.7|**8**	**0.0**|3	**0.0**|2	2.0|2	**0.0**|1	**0.0**|1	**0.0**|7	✓|✓
#best	2|2	4|1	5|1	5|6	3|10	5|9	3|0	3|2	0|0	4|0	4|0	3|0	

(pc=Pearson's correlation, snr=signal-to-noise ratio, pam=shrunken centroid, t=t-statistics, m-t=moderated t-statistics, lods=B-statistics, lin=centroid, rbf=$\exp(-\frac{\|x-x'\|}{2\sigma^2})$, dis=$\|x - x'\|^{-1}$)

Table 2: Multiclass datasets: in this case *columns* are the datasets, and *rows* are the methods. The remaining conventions follow Table 1.

Ref.#	16	17	18	18	20	21	22	23	24	25	26	27	28	**#best**
lin	36.7|1	**0.0**|3	5.0|**3**	10.5|6	35.0|3	**37.5**|**6**	**18.6**|1	**40.3**|**3**	**28.1**|**3**	26.6|**6**	**5.6**|**6**	27.9|7	45.1|1	6|6
rbf	33.3|**3**	5.1|4	**1.7**|**3**	**7.2**|**9**	33.3|0	40.0|1	22.1|0	72.5|0	39.5|0	**24.7**|4	**5.6**|**6**	22.1|**10**	**21.5**|3	5|5
dis	**29.7**|2	28.8|**5**	6.7|0	8.2|**9**	**29.4**|**7**	38.3|4	43.4|**4**	66.1|0	40.8|0	38.9|4	7.6|1	**8.2**|8	31.6|3	3|4
mi	42.0|1	11.4|3	**1.7**|2	7.7|8	39.4|4	38.3|3	30.3|1	57.3|2	37.6|1	40.8|2	6.5|**6**	22.6|3	23.3|**6**	1|2
	✓|✓	✓|✓	✓|✓	✓|✓	✓|✓	✓|✓	✓|✓	✓|✓	✓|✓	✓|✓	✓|✓	✓|✓	✓|-	

Table 3: Classification errors (%) on BCI data after selecting a frequency range.

Subject	aa	al	av	aw	ay
CSP(8-40Hz)	17.5±2.5	3.1±1.2	32.1±2.5	7.3±2.7	**6.0±1.6**
CSSP	14.9±2.9	2.4±1.3	33.0±2.7	**5.4±1.9**	*6.2±1.5*
CSSSP	**12.2±2.1**	*2.2±0.9*	*31.8±2.8*	6.3±1.8	12.7±2.0
BAHSIC	*13.7±4.3*	**1.9±1.3**	**30.5±3.3**	*6.1±3.8*	9.0±6.0

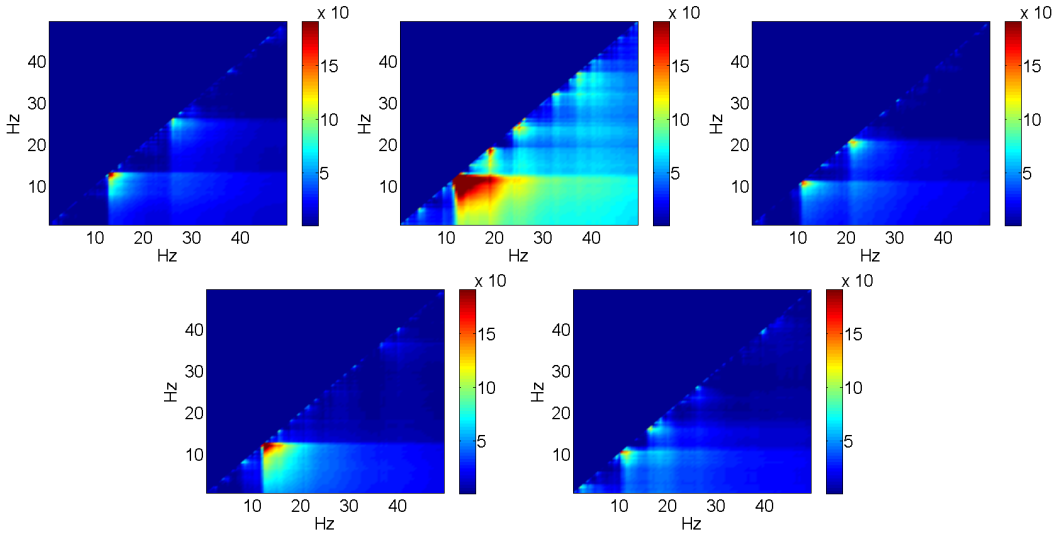

Figure 1: HSIC computed at different frequency bands. Ordered from upper row to bottom row, and left to right are the results for subject 'aa', 'al', 'av', 'aw' and 'ay' respectively.

will create block patterns. Both predictions are confirmed in Figure 1. Furthermore, the large area of the red region for subject 'al' indicates good responsiveness of his μ rhythm. This also corresponds well with the lowest classification error obtained for him in Table 3.

5.3 Regression Data

BAHSIC is also applied to 3 regression datasets. Two datasets, Pyrimidines (Pyrim) and Triazines (Triaz), are from the UCI repository,[2] and the third one, Bodyfat, is from the Statlib repository.[3] The dimension, the number of samples and the number of selected features are listed in Table 4. All regression results presented in Table 4 are obtained using 10-fold cross-validation and ϵ-support vector regression (ϵ-SVR with $\epsilon = 10^{-3}$ and a Gaussian RBF kernel). For comparison, the regression results using all features and randomly selected features are also presented (ϵ-SVR and RAND column). It is clear from Table 4 that selecting features using BAHSIC still produce comparable root mean square error (RMSE) in the subsequent regression.

Table 4: Root mean square error (RMSE) of support vector regression with and without HSIC

Method	Sample	Dim	Selected #	ϵ-SVR	RAND	BAHSIC
Pyrim	55	27	5	0.112±0.067	0.092±0.073	**0.085±0.066**
Triaz	186	60	2	0.147±0.027	0.157±0.036	**0.144±0.033**
Bodyfat	227	14	7	0.0019±0.0026	0.0019±0.0026	**0.0019±0.0024**

6 Conclusion

This paper proposes a backward elimination procedure for feature selection using the Hilbert-Schmidt Independence Criterion (BAHSIC). The basic idea of BAHSIC is to choose the feature subset that maximizes the dependence between the data and the labels. With this interpretation, BAHSIC provides a unified framework for supervised feature selection problems. This framework

[2] `http://www.ics.uci.edu/~mlearn/MLSummary.html`
[3] `http://lib.stat.cmu.edu/datasets/`

includes several well-known feature selection methods, which differ only by the choice of the preprocessing and the kernel function. Our experiments show that the BAHSIC family of feature selection algorithms performs well in practise, both in terms of accuracy and robustness, and that the more sophisticated members of this family are directly applicable to multi-class and regression problems.

The BAHSIC family represents a step towards establishing theoretical links between the huge set of feature selection algorithms in the literature. Only if we fully understand these theoretical connections can we hope to explain why different methods select different features, and to choose feature selection methods that yield the most meaningful results.

References

[1] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins. Text classification using string kernels. *Journal of Machine Learning Research*, 2:419–444, February 2002.

[2] T. Gärtner, P.A. Flach, and S. Wrobel. On graph kernels: Hardness results and efficient alternatives. In B. Schölkopf and M. K. Warmuth, editors, *Proc. Annual Conf. Computational Learning Theory*. Springer, 2003.

[3] B. Schölkopf, K. Tsuda, and J.-P. Vert. *Kernel Methods in Computational Biology*. MIT Press, Cambridge, MA, 2004.

[4] C. Baker. Joint measures and cross-covariance operators. *Transactions of the American Mathematical Society*, 186:273–289, 1973.

[5] A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf. Measuring statistical dependence with Hilbert-Schmidt norms. In *Proc. Intl. Conf. on Algorithmic Learning Theory*, pages 63–78, 2005.

[6] Andrey Feuerverger. A consistent test for bivariate dependence. *International Statistical Review*, 61(3):419–433, 1993.

[7] I. Steinwart. On the influence of the kernel on the consistency of support vector machines. *Journal of Machine Learning Research*, 2:67–93, 2002.

[8] A. Gretton, O. Bousquet, A.J. Smola, and B. Schölkopf. Measuring statistical dependence with Hilbert-Schmidt norms. In S. Jain and W.-S. Lee, editors, *Proceedings Algorithmic Learning Theory*, 2005.

[9] R. Serfling. *Approximation Theorems of Mathematical Statistics*. Wiley, New York, 1980.

[10] K. M. Borgwardt, A. Gretton, M.J. Rasch, H.P. Kriegel, B. Schölkopf, and A.J. Smola. Integrating structured biological data by kernel maximum mean discrepancy. In *Proc. of Intelligent Systems in Molecular Biology (ISMB)*, Fortaleza, Brazil, 2006.

[11] N. Cristianini, J. Shawe-Taylor, A. Elisseeff, and J. Kandola. On kernel-target alignment. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 367–373, Cambridge, MA, 2002. MIT Press.

[12] J. Neumann, C. Schnörr, and G. Steidl. Combined SVM-based feature selection and classification. *Machine Learning*, 61:129–150, 2005.

[13] L. J. van 't Veer, H. Dai, M. J. van de Vijver, Y. D. He, A. A. M. Hart, et al. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415:530–536, 2002.

[14] L. Ein-Dor, O. Zuk, and E. Domany. Thousands of samples are needed to generate a robust gene list for predicting outcome in cancer. *Proc Natl Acad Sci U S A*, 103(15):5923–5928, Apr 2006.

[15] Justin Bedo, Conrad Sanderson, and Adam Kowalczyk. An efficient alternative to svm based recursive feature elimination with applications in natural language processing and bioinformatics. In *Artificial Intelligence*, 2006. to appear.

[16] Gordon K. Smyth. Linear models and empirical bayes methods for assessing differential expressionin microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, 3, 2004.

[17] R. Tibshirani, T. Hastie, B. Narasimhan, and G. Chu. Diagnosis of multiple cancer types by shrunken centroids of gene expression. In *National Academy of Sciences*, volume 99, pages 6567–6572, 2002.

[18] R. Tibshirani, T. Hastie, B. Narasimhan, and G. Chu. Class prediction by nearest shrunken centroids, with applicaitons to dna microarrays. *Stat Sci*, 18:104–117, 2003.

[19] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, New York, 2001.

[20] B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002.

[21] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46:389–422, 2002.

[22] R. Tibshirani. Regression selection and shrinkage via the lasso. Technical report, Department of Statistics, University of Toronto, June 1994. ftp://utstat.toronto.edu/pub/tibs/lasso.ps.

[23] G. Dornhege, B. Blankertz, G. Curio, and K. Müller. Boosting bit rates in non-invasive EEG single-trial classifications by feature combination and multi-class paradigms. *IEEE Trans. Biomed. Eng.*, 51:993–1002, 2004.

[24] S. Lemm, B. Blankertz, G. Curio, and K.-R. Mülller. Spatio-spectral filters for improving the classification of single trial EEG. *IEEE Trans. Biomed. Eng.*, 52:1541–1548, 2005.

[25] G. Dornhege, B. Blankertz, M. Krauledat, F. Losch, G. Curio, and K. Müller. Optimizing spatio-temporal filters for improving BCI. In *Advances in Neural Information Processing Systems 18*, 2006.