Kernelized Design of Experiments

Stefan Rüping
Fraunhofer IAIS
Schloss Birlinghoven, 53754 St. Augustin, Germany
stefan.rueping@iais.fraunhofer.de

Claus Weihs
Statistics Department
University of Dortmund
44221 Dromond, Germany weihs@statistik.uni-dortmund.de

February 18, 2009

Abstract

This paper describes an approach for selecting instances in regression problems in the cases where observations x are readily available, but obtaining labels y is hard. Given a database of observations, an algorithm inspired by statistical design of experiments and kernel methods is presented that selects a set of k instances to be chosen in order to maximize the prediction performance of a support vector machine. It is shown that the algorithm significantly outperforms related approaches on a number of real-world datasets.

1 Introduction

A typical application of regression estimation is to predict some aspect y of real-world entities that is hard to measure by other, more readily available features x. A good example can be found in the description of the abalone data set [1] from the UCI Machine Learning Repository [2]:

> *The age of abalone is determined by cutting the shell through the cone, staining it, and counting the number of rings through a microscope – a boring and time-consuming task. Other measurements, which are easier to obtain, are used to predict the age.*

In some cases, measuring the value of the attribute of interest y may be so hard that it even becomes a problem to obtain enough measurements to learn the prediction function. Suppose it is known in advance that for a data set of n observations x_i only k measurements of the corresponding y can be obtained, where $k << n$. The question is which $x_{i_1}, \ldots, x_{i_k}$ to choose in order to learn the most accurate regression function from the completed sample $(x_{i_1}, y_{i_1}), \ldots, (x_{i_k}, y_{i_k})$.

This problem setting is relevant in several real-world problems. For example, in medical diagnosis measuring the attribute of interest may require a risky medical procedure and hence should be strictly limited to the cases where it is absolutely necessary, while other measurements x may be routinely measured for a large set of patients. When user interaction is required to measure y one often finds that people quickly get bored when they have to answer too many questions without getting an immediate return.

In this paper, kernelized design of experiments is used to select the most informative instances for support vector regression with a given kernel K. The paper is structured as follows: the following section introduces the very basics of support vector machines (SVMs) and kernel PCA (kPCA), as far as they are relevant for this paper. Section 3 introduces the statistical problem of design of experiments (DoE) and the idea of experimental design on observational data. Section 4 contains the new contribution of this paper, kernelized design of experiments, which is empirically evaluated in Section 5. Section 6 discusses related work and 7 concludes.

2 Learning with Kernels

Kernel methods are a very popular and successful area of machine learning. Their common basis is the so-called kernel trick, which can be applied to any linear algorithm which depends on the data only in terms of the inner product of two examples. In this paper we make use of two kernel methods, Support Vector Regression (SVR) and Kernel PCA (kPCA), in order to apply the kernel trick to Design of Experiments.

2.1 Support Vector Regression

Given examples (x_i, y_i) Support Vector Regression ([3]) finds a regression function $f : X \to Y$ by solving the following optimization problem:

$$
\begin{array}{ccc}
||w||^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i') & \to & \min \\
& \text{subject to} & \\
\forall_{i=1}^{n} f(x_i) \geq y_i - \varepsilon - \xi_i & & \forall_{i=1}^{n} f(x_i) \leq y_i + \varepsilon + \xi_i' \\
\forall_{i=1}^{n} \xi_i \geq 0 & & \forall_{i=1}^{n} \xi_i' \geq 0
\end{array}
$$

The regression function f for a kernel K has the form

$$f(x) = w * \Phi(x) + b = \sum_i \alpha_i K(x_i, x) + b.$$

2.2 Kernel PCA

Kernel PCA [4] is an extension of the regular (linear) principal component analysis (PCA). The idea of PCA is shown in Figure 1: Given a set of data, the vector along which the data shows the most variance is the first principal component. Given the first i principal components, the $i+1$-st principal component is the vector orthogonal to the first i principal components along which the data shows the most variance. It follows that the best reconstruction of the data in an i-dimensional subspace is given by the first i principal components.

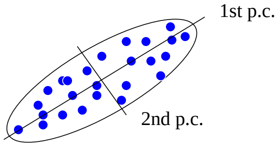

Figure 1: Linear principal component analysis

PCA hence can be kernelized [4]. It can be shown that performing PCA in feature space is equivalent to finding the solutions $\vec{\alpha}^{(1)}, \vec{\alpha}^{(2)}, \ldots$ of

$$n\lambda\vec{\alpha} \quad = \quad K\vec{\alpha}$$

where K is the kernel matrix. To normalize the eigenvectors in feature space one sets $|||\vec{\alpha}^{(k)}||^2 = 1/\lambda_k$. Principal component extraction for an observation x is computed by projecting x on the eigenvectors V_k, i. e. computing

$$V_k\Phi(x) \quad = \quad \sum_{i=1}^{n} \vec{\alpha}_i^{(k)} \Phi(x_i)\Phi(x) = \sum_{i=1}^{n} \vec{\alpha}_i^{(k)} K(x_i, x). \tag{1}$$

For non-centered data, i. e. $\sum_{i=1}^{n} \Phi(x_i) \neq 0$, the $\Phi(x)$ are replaced by their centered counterparts

$$\tilde{\Phi}(x) = \Phi(x) - \frac{1}{n}\sum_{i=1}^{n} \Phi(x_i).$$

The kPCA solution can then be computed using the kernel matrix $\tilde{K}$ corresponding to $\tilde{\Phi}$ [5]. Kernel PCA can be used to extract features from data in order to show up the structure defined by the kernel.

3 Design of Experiments

Given a process which can be described by d features $x^{(j)}$ and a feature of interest y, experimental design describes the task of finding a set of k observations $x_i \in R^d$ that are maximally informative about the dependency $X \to Y$ (it is customary to choose $k = d+1$). The matrix $X = (x_1, \ldots, x_k)^T \in \mathcal{R}^{k \times d}$ is called an experimental plan. Experimental design is particularly useful when very few information about the true model $f : X \to Y$ is known and gathering examples (x_i, y_i) is hard.

In order to extract meaningful information from a small set of observations, some assumptions are needed. In order to asses the effect of each feature independently of the others, it is necessary that the plan is orthogonal:

Definition: A plan X is defined to be orthogonal, iff the matrix $S = X^T X$ is of diagonal form, i.e. for each two columns of the plan $X_{\cdot i} \cdot X_{\cdot j} = 0$ iff $i \neq j$.

It is not always possible to find an orthogonal plan. For example, for binary features it can be shown that such a plan only exists for values of n which are multiples of 4 (Hadamard plans). In this case one has to resort to nearly orthogonal plans, that is plans that maximize some criterion measuring orthogonality. Several criteria based on the non-diagonal entries S_{ij} of S exist, such as:

$$\begin{aligned}
Ave(S^2) &:= \sum_{i<j} S_{ij}^2 / \binom{m}{2} \\
Ave|S| &:= \sum_{i<j} |S_{ij}| / \binom{m}{2} \\
S_{max} &:= \max_{i<j} |S_{ij}| \\
S_{\#} &:= \#\{i, j | i < j, S_{ij} = S_{max}\}
\end{aligned}$$

see [6, 7, 8, 9, 10, 11]. For plans on finite input spaces X, also other criteria which are based on the frequency of value pairs occurring in each two columns of the plan can be used [11],

For the special case of an assumed linear dependency $y = Xw + b$, [12] have used the criterion of D-optimality, which says to select the plan X which maximizes the D-value $\mathcal{D}(X)$:

Definition: For a plan X, the D-value $\mathcal{D}(X)$ is defined as

$$\mathcal{D}(X) \quad := \quad \det(\hat{X}^T \hat{X}) \tag{2}$$

where $\hat{X}$ is defined by

$$\hat{X} = (X_{\cdot 1}/||X_{\cdot 1}||, \ldots, X_{\cdot d}/||X_{\cdot d}||). \tag{3}$$

It has been shown [12, 13] that a D-optimal plan is the one that minimizes the uncertainty about the factor values w by minimizing the volume of the confidence ellipsoid for a fixed confidence level around w with respect to all comparable designs. As an interpretation of the D-value, note that basic linear algebra says that $\mathcal{D}(X)$ is the volume of the parallelepiped that is spanned by the column vectors of $\hat{X}$. As these vectors are normalized observations, this volume is maximized when all vectors are orthogonal to each other.

3.1 Experimental Design in Observational Data

Experimental design assumes that one is free to choose the observations x_i for which the values y_i should be measured. However, there exists situations like the one investigated in this paper, where only a fixed set $x_1, \ldots, x_n$ of observations are available to choose from. For example, when the observations are patients in a medical study, one cannot simply construct new patients that fit the requirements. The problem of selecting k observations from a set of $n >> k$ observations, such that the k observations form an optimal plan, is called experimental design in observational data and has been introduced in [14, 15].

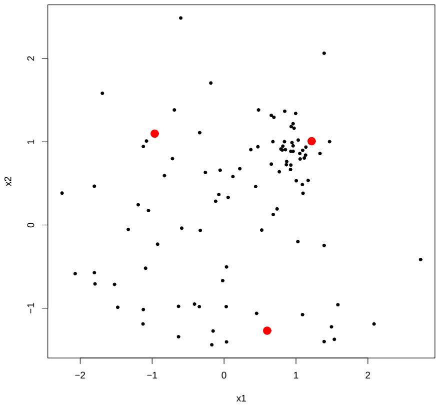

Figure 2: Data set (smaller dots) and optimal plan (larger dots)

For special cases, such as all binary features, efficient algorithms exists that can find subsets of the observations which form orthogonal plans. In the general case of non-binary data and nearly orthogonal plans, no such algorithm is known and hence in this paper we use a heuristic search by a genetic algorithm. Each gene represents a plan and is given by a set of k observation indices in $1, \ldots, n$. We use a standard GA with roulette selection, a pool of 1000 genes, mutation probability 0.1 and crossover probability 0.2 and 1000 iterations. As this paper is not concerned with the efficiency of the approach, these parameters have been chosen in an ad-hoc fashion, and it might be easily possible to find better ones, or a better optimization approach altogether.

4 Kernelized DoE

D-optimality is based on the concept of orthogonality. Obviously, if we have an orthonormal basis $\mathcal{O} = o_1, \ldots, o_d$ of the input space, i.e. $o_i \cdot o_j = \delta_{ij}$, the inner product and hence the orthogonality remains equal if we represent all vectors x in terms of their coefficients of the basis $\mathcal{O}$, that is if we use

$$x' = (o_1 \cdot x, \ldots, o_d \cdot x).$$

This gives rise to a kernelization of D-optimal plans. If we choose an orthonormal basis of the feature space and project all observations onto this basis, we can search for D-optimal plans in feature space. Such a basis, or at least a basis of the subspace of feature space spanned by the observations $(x_1, \ldots, x_n)$ via the implicit mapping Φ, can be found by kPCA. Thus, we can perform the search for D-optimal plans in feature space by using the projections of the observations to the first $k-1$ principal components found by kernel PCA, letting the number of observations be the dimension + 1. Note that the criterion of D-optimality is based on the assumption of a linear model, which for SVMs only holds in feature space, such that observation selection necessarily needs to be done in feature space. This gives rise to the following algorithm:

1. Input: set of observations $x_1, \ldots, x_n$, a kernel K and a desired number of instances k

2. Execute kPCA with kernel K on the $x_1, \ldots, x_n$

3. Select the $k-1$ eigenvectors $V_1, \ldots, V_{k-1}$ with largest eigenvalue and compute the projection of all x_i onto these eigenvectors using Equation 1. This gives the set of transformed observations $x'_1, \ldots, x'_n \in \mathcal{R}^{k-1}$

4. Search for an optimal plan on the transformed observations using the genetic algorithm described in Section 3.1

5. Return the k observations that form the optimal plan

Step 3 is necessary because the feature space dimension can be infinite or at least as large as the number of observations. Thus, dimensionality is reduced by kPCA to the $k-1$ most informative dimensions in the space spanned up by the given kernel K.

5 Experiments

The work presented in this paper has been evaluated on 4 regression data sets from the UCI Machine Learning Repository [2]. We selected data sets with a high number of examples and limited dimensionality. Data sets were pre-processed by dichotomizing nominal attributes and z-scaling numerical attributes. Table 1 gives the statistics of the data sets used in our experiments. For the sake of runtime efficiency, data sets with more than 1000 examples were down-sampled to a size of 1000.

Four different kernel functions were used in the experiments, the linear kernel, the polynomial kernel with degree 2 and radial basis kernels with width

Table 1: Data Sets

Name	#Examples	Dimension
abalone	4177	9
bank8fm	8192	9
cal_housing	20640	9
stock	950	10

$\gamma = 0.1, 0.01$. These kernels cover the most widely used kernel functions in the literature; kernel parameters were chosen by prior knowledge.

The proposed method (called kDoE in the following) has been compared to two other methods for instance selection:

Rand: random sampling has been used as the baseline method

kPCA: in order to evaluate whether the kernel PCA was solely responsible for the performance of the new method, kernel PCA has been applied and for each of the first k principal components, the observations with maximal absolute projection onto this component has been selected.

To estimate the quality of the selected instances, an SVM has been trained on these instances, using 10-fold cross-validation on the selected examples to tune the parameter C. The error of this SVM has been compared to the error of the SVM using the same kernel on all instances in order to account for different levels of noise in the data sets. That is, we based our comparison on the relative error

$$Err_{rel} = \frac{Err}{Err_{fullSVM}}$$

All reported results have been obtained using 10-fold cross-validation.

Results can be seen in Table 2. It is obvious that the new kDoE approach outperforms the other approaches. Figure 3 shows an exemplary learning curve for the three approaches on the Abalone data set with dot kernel. It can be seen that kDoE lower-bounds the other approaches in most of the cases. For larger k, the learning curves converge.

5.1 Statistical Significance

As the relative errors over the different methods in Table 2 are very close, the question of statistical significance arises. To compare the three methods over all data sets, we used both a Binomial test comparing the wins and losses of kDoE relative to the other methods (with respect to the 10-fold cross-validated error) and the Wilcoxon Signed Rank Test on the relative errors. Both tests have been suggested in [16] for the comparison of learners over multiple data sets. The difference between both is that the Wilcoxon test assumes a commensurability of the values (which, due to the normalization by using the relative error may be

Table 2: Relative errors and standard deviation

Data set	**Kernel**	**kDoE**	**kPCA**	**Rand**
abalone	dot	**2.382** (0.433)	2.528 (0.538)	2.863 (1.537)
	radial(0.001)	**2.107** (0.234)	2.322 (0.461)	2.265 (0.385)
	radial(0.01)	2.270 (0.401)	2.717 (0.453)	**2.156** (0.330)
	polynomial(2)	2.439 (0.227)	2.359 (0.253)	**2.257** (0.159)
bank8fm	dot	**0.084** (0.027)	0.183 (0.100)	0.086 (0.042)
	radial(0.001)	**0.057** (0.018)	0.191 (0.108)	0.065 (0.022)
	radial(0.01)	**0.061** (0.038)	0.143 (0.027)	0.067 (0.018)
	polynomial(2)	**0.108** (0.023)	0.261 (0.119)	0.119 (0.015)
cal_housing	dot	**0.435** (0.093)	0.469 (0.088)	0.514 (0.187)
	radial(0.001)	0.475 (0.073)	**0.404** (0.031)	0.536 (0.191)
	radial(0.01)	**0.427** (0.069)	0.612 (0.034)	0.431 (0.082)
	polynomial(2)	**0.489** (0.039)	0.559 (0.099)	0.548 (0.105)
stock	dot	**2.752** (0.167)	9.337 (0.878)	4.141 (2.288)
	radial(0.001)	**3.190** (0.659)	7.903 (1.592)	3.615 (1.198)
	radial(0.01)	**2.537** (0.361)	3.821 (1.305)	3.122 (0.453)
	polynomial(2)	**3.276** (0.339)	3.949 (0.591)	4.106 (0.687)

Table 3: Statistical Comparison of kDoE vs. kPCA

Kernel	**Wins**	**Losses**	**p-Binom**	**p-Wilcox**
dot	4	0	0.0625	0.0625
radial(0.001)	3	1	0.3125	0.125
radial(0.01)	4	0	0.0625	0.0625
polynomial(2)	3	1	0.3125	0.1875
all	14	2	0.0021	0.0003

assumed to hold), while the Binomial test does not depend on this assumption. We will see that in the experiments both tests agree in their decision about significance.

As can be seen in Table 3 for the comparison of kDoE versus kPCA and in Table 3 for the comparison of kDoE versus Rand, both statistical tests confirm the superior performance of kDoE. It can be seen that is has a somewhat less significant performance for the polynomial kernel, and a particularly good performance for the linear kernel.

6 Related Work

The general problem of instance selection describes the problem of selecting a small set of highly informative instances from a larger set of examples [17]. In-

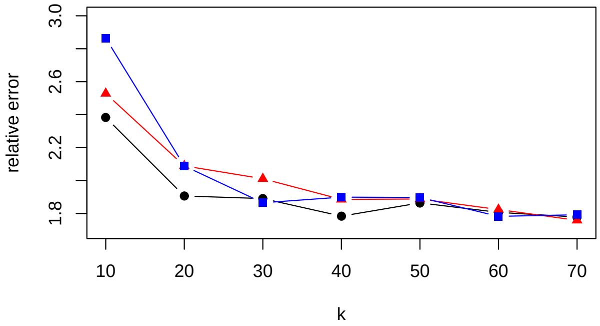

Figure 3: Learning curve for the dot kernel on the Abalone data set (points = kDoE, triangles = kPCA, squares = Rand).

Table 4: Statistical Comparison of kDoE vs. Rand

Kernel	**Wins**	**Losses**	**p-Binom**	**p-Wilcox**
dot	4	0	0.0625	0.0625
radial(0.001)	4	0	0.0625	0.0625
radial(0.01)	3	1	0.3125	0.3125
polynomial(2)	3	1	0.3125	0.3125
all	14	2	0.0021	0.0055

formative examples can be divided into prototypes and discriminating instances. Prototypes are examples which are similar to a large number of examples and can hence be taken as a typical representative of this set of examples. Discriminating instances are examples which are representative of the distinction between different classes of examples.

A conceptual problem of instance selection in a classification setting is a missing general measure of instance importance. Several ad-hoc solutions to instance selection in the context of specific learners exist. For example, k-medoids clustering can be seen as the explicit search for prototypes that represent the structure of the data.

An approach to instance selection based on data squashing is presented by [18]: Assuming the optimal model for the data (x_i, y_i) is represented by a parameter $\theta \in \mathbf{R}^d$, the idea is to inspect a set of similar models $\theta_j = \theta + \delta_j$ and inspect the conditional probabilities $p_{ij} = P_{\theta_j}(y_i|x_i)$ that these models assign to the examples. The number of different models tested and the size of the deviations δ_j are given as a parameter to the method. In contrast to the method described in this paper, the use of the conditional probabilities assumes the labels to be known.

Active learning [19] is another approach to deal with the problem of high cost of assigning labels to observations. In active learning, a measure of certainty of the learner regarding the label of an example is computed, which is then used to query the label of the most uncertain unlabeled observation. The problem is that in order to estimate a meaningful measure of certainty, a reasonably large set of examples must already be available.

Other approaches which make use of unlabeled examples are known under the name of semi-supervised learning [20]. As in active learning, a reasonably large set of examples must already be available in order to use semi-supervised approaches, which suggest the possibility of combining them with kDoE. Instance selection was also recently investigated under the problem of sensor placement [21], however in this approach information about the labels y is assumed to be known to estimate variances.

In conclusion it can be said that the problem of Design of Experiments is the most complex of these tasks, as (1) the only knowledge about the labels y it assumes is the kernel function k, but not any label itself, and (2) it requires to select all k observations simultaneously, ruling out the possibility to take information from previously selected examples into account.

In situations where more information is available, it is obvious to assume that other approaches may outperform kDoE. An optimal combination may be to use kDoE to bootstrap other learning schemes on a non-labeled data set, such that kDoE selects an initial set of examples.

7 Conclusions and Future Work

In this paper, an approach for instance selection in observational data was presented. The selected instances can be used to maximize the predictive performance of a support vector machine learner over training sets of size k when obtaining labels y is hard. It has been shown empirically that the proposed approach significantly outperforms competing algorithms.

Future work may explore the relationship of kDoE with other learning algorithms, such as active learning. It may also be interesting to explore the performance of kDoE on very high dimensional data sets, such as text corpora. Another interesting direction of research includes the investigation of efficient, deterministic algorithms for observation selection based on D-optimality.

Acknowledgement: The financial support of the Deutsche Forschungsgemeinschaft (SFB 475, "Reduction of complexity in multivariate data structures") is gratefully acknowledged.

References

[1] Warwick J. Nash, Tracy L. Sellers, Simon R. Talbot, Andrew J. Cawthorn, and Wes B. Ford. The population biology of abalone (haliotis species) in tasmania. i. blacklip abalone (h. rubra) from the north coast and islands of bass strait. Technical Report 48, Sea Fisheries Division, 1994.

[2] P. M. Murphy and D. W. Aha. UCI repository of machine learning databases, 1994.

[3] V. Vapnik. *Statistical Learning Theory.* Wiley, Chichester, GB, 1998.

[4] B. Schölkopf, A. Smola, and K.-R. Müller. Kernel principal component analysis. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, pages 69–88. MIT Press, 1999.

[5] Bernhard Schölkopf, Alex J. Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.

[6] K. H. V. Booth and D. R. Cox. Some systematic supersaturated designs. *Technometrics*, 4:21–46, 1962.

[7] D. K. J. Lin. A new class of supersaturated designs. *Technometrics*, 35:28–31, 1993.

[8] D. K. J. Lin. Generating systematic supersaturated designs. *Technometrics*, 1995.

[9] C. F. J. Wu. Construction of supersaturated designs through partially aliased interactions. *Biometrika*, 80:661–669, 1993.

[10] L. Y. Deng, D. K. Y. Lin, and J. N. Wang. On resulution rank criterion for supersaturated designs. *Statist. Sinica*, 9:605–610, 1999.

[11] Kai-Tai Fang, Dennis K. J. Lin, and Chang-Xing Ma. On the construction of multi-level supersaturated designs. *Journal of Statistical Planning and Inference*, 86:239–252, 2000.

[12] J. C. Wang and C. F. J. Wu. Nearly orthogonal arrays with mixed levels and small runs. *Technometrics*, 34(5):409–422, 1992.

[13] A. C. Atkinson and A. N. Donev. *Optimum Experimental Design.* Clarendon Press, Oxford, 1992.

[14] C. Pumplün, C. Weihs, and A. Preusser. Experimental design for variable selection in data bases. In C. Weihs and W. Gaul, editors, *Classification - The Ubiquitous Challenge*, pages 192–199. Springer, 2005.

[15] Constanze Pumplün, Stefan Rüping, Katharina Morik, and Claus Weihs. D-optimal plans in observational studies. Technical Report 44/05, SFB475, University of Dortmund, Dortmund, Germany, 2005.

[16] Janez Demsar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[17] Huan Liu and Hiroshi Motoda. *Instance Selection and Construction for Data Mining*. Kluwer Publishers, 2001.

[18] David Madigan, Nandini Raghavan, William DuMouchel, Martha Nason, Christian Posse, and Greg Ridgeway. Likelihood-based data squashing. *Data Mining and Knowledge Discovery*, 6(2):173–190, 2002.

[19] Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, pages 45–66, 2001.

[20] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Annual Conference on Computational Learning Theory (COLT-98)*, 1998.

[21] Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *JMLR*, 9:235–283, Feb 2007.