EXPECTED KERNEL FOR MISSING FEATURES IN SUPPORT VECTOR MACHINES

Hyrum S. Anderson

Sandia National Laboratories[1]
Data Analysis and Exploitation Department
Albuquerque, NM 87123

Maya R. Gupta

University of Washington
Department of Electrical Engineering
Seattle, WA 98125

ABSTRACT

The expected kernel for missing features is introduced and applied to training a support vector machine. The expected kernel is a measure of the mean similarity with respect to the distribution of the missing features. We compare the expected kernel SVM with the robust second-order cone program (SOCP) SVM, which accounts for missing kernel values by estimating the mean and covariance of missing similarities. Further, we extend the SOCP SVM to utilize the expected kernel by deriving the expected kernel variance. Results show that the expected kernel—used with a traditional SVM solver—shows competitive performance on benchmark datasets to the SOCP SVM at a far-reduced computational burden.

Index Terms— missing features, support vector machine, kernel, expected kernel

1. INTRODUCTION

Data with missing features is a common problem encounted in signal processing, statistics and machine learning. A standard solution is to impute the missing values, for example by replacing the values with the average of all other related data instances, or using k nearest neighbors (k-NN) of each instance to fill in missing values (also called *hot deck imputation*). Farhangfar et al. recently analyzed the effect of imputing missing features in training data using several standard imputation strategies, and found that classification accuracy of support vector machine (SVM) and k-NN classifiers generally benefit from imputation [1].

In multiple imputation, each missing value is replaced by a list of $m > 1$ plausible values, producing m plausible alternative versions of the complete data [2]. Classification or regression algorithms can be run independently on each of the m datasets, then averaged Monte-Carlo style to form an output and a variance that reflects missing-data uncertainty. Multiple imputation has been used to compute support vector machines (SVM) kernels to handle missing features [1].

Rather than form a discrete set of multiple imputation candidates, we propose using the *expected kernel*—the mean similarity value over the distribution of possible candidates—with an SVM. Expected kernels have been used to incorporate uncertainty due to additive and channel noise on test signals [3], and the same formulation has also been termed *marginalized kernel* and used to marginalize out hidden variables when computing kernels between graphs [4]. Here we show that the expected kernel is an effective and efficient approach to the missing features problem.

In related work, Shivaswamy et al. introduced a generalization of the SVM so that the margin-maximization takes into account the uncertainty due to missing features or noise [5]. The resulting second-order cone program (SOCP) SVM is discussed in more detail in section 2.3. The expected kernel and kernel variance—which we introduce in Sec. 2.3—are naturally suited for use in the SOCP SVM.

We compare the proposed expected kernel and the expected kernel SOCP SVM to the SOCP SVM [5] on benchmark datasets with features missing completely at random. We find that the three methods demonstrate similar performance, although training the expected kernel SVM is a simple quadratic program (QP) that can be trained with a fast SVM solver, whereas the robust SVM requires an SOCP solver.

2. EXPECTED KERNEL

We model each feature vector with missing components as a random vector X_i distributed as p_{X_i} with finite mean m_i and covariance Σ_i. Given any kernel $K : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ that measures the similarity between its inputs, the corresponding expected kernel K_{exp} is defined to be the following functional of two distributions [3]:

$$K_{\text{exp}}(p_{X_i}, p_{X_j}) \triangleq \mathrm{E}_{X_i,X_j}\left[K\left(X_i, X_j\right)\right] \quad (1)$$
$$= \iint p_{X_i}(x_i)p_{X_j}(x_j)K(x_i, x_j)dx_i dx_j.$$

The expected kernel averages the similarity of all possible values of x_i and x_j, weighted by their respective probability

[1] Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

densities. The expected kernel is positive definite since it is a weighted average of positive definite (PD) functions $K(\cdot,\cdot)$ on the convex PD cone. Note that when $\Sigma_i = 0$ and $\Sigma_j = 0$ for $i = 1, \ldots, n$ (that is, there is no uncertainty in the two feature vectors, s.t. $X_i = m_i$ and $X_j = m_j$), then the expected kernel reduces to the user-defined kernel $K(x_i, x_j)$.

2.1. Example Expected Kernels

The expected kernel is actually a family of kernels parameterized by the practitioner's choice of $K(\cdot,\cdot)$, p_{X_i} and p_{X_j}. Next, we show that for two standard kernel choices the expected kernel has a closed-form solution that depends only on the means and covariances of p_{X_i} and p_{X_j}.

Expected Inner Product Kernel. For the inner product kernel $K(x_i, x_j) = x_i^T x_j$ (used in the linear SVM), the expected kernel depends only on the mean and covariance of the distributions p_{X_i} and p_{X_j}:

$$K_{\text{exp}}^{\text{lin}}(p_{X_i}, p_{X_j}) = m_i^T m_j + \delta_{i=j} \text{tr } \Sigma_i, \tag{2}$$

where $\text{tr } A$ is the trace of matrix A and $\delta_{i=j} = 1$ if $i = j$ and 0 otherwise.

Proof. For independent random vectors X_i and X_j, (2) follows directly from application of (1). For the case that $i = j$, (1) becomes

$$\begin{aligned} \mathrm{E}_{X_i}\left[K(X_i, X_i)\right] &= \mathrm{E}_{X_i}\left[X_i^T X_i\right] = \mathrm{E}_{X_i}\left[\text{tr }\left(X_i^T X_i\right)\right] \\ &= \mathrm{E}_{X_i}\left[\text{tr}\left(X_i X_i^T\right)\right] = \text{tr}\left(\mathrm{E}_{X_i}\left[X_i X_i^T\right]\right) \\ &= \text{tr } \Sigma_i + m_i^T m_j. \end{aligned}$$

The proof holds for any p_{X_i} and p_{X_j}. □

Expected RBF Kernel. Assuming independent Gaussian random vectors X_i and X_j and radial basis function (RBF) kernel $K(x_i, x_j) = \exp\left(-\frac{\gamma}{2}\|x_i - x_j\|^2\right)$ with bandwidth γ, the expected RBF kernel is given by

$$K_{\text{exp}}^{\text{rbf}}(p_{X_i}, p_{X_j}) = \tag{3}$$
$$\frac{\exp\left(-\frac{1}{2}\left(m_i - m_j\right)^T \left(\Sigma_i + \Sigma_k + \gamma^{-1} I\right)^{-1} \left(m_i - m_j\right)\right)}{\left|\gamma\Sigma_i + \gamma\Sigma_j + I\right|^{\frac{1}{2}}}.$$

Proof. Follows from (1) since $\int \mathcal{N}(x; a, A)\mathcal{N}(x; b, B)\, dx = \mathcal{N}(a; b, A + B)$, where $\mathcal{N}(x; a, A)$ is a normal distribution in x with mean a and covariance A. □

Anderson et al. showed that ignoring the scaling term for the RBF expected kernel in (3) also results in a PD kernel $K_{\text{exp}}^{\text{urbf}}(\cdot,\cdot)$ which has the satisfying property that $K_{\text{exp}}^{\text{urbf}}(p_{X_i}, p_{X_i}) = 1$ (identity along the diagonal of the kernel matrix) [3].

Key to the missing features problem is that the expected kernel adapts the notion of similarity between two feature vectors by accounting for the covariance of the imputation estimate: the expected inner product kernel in (2) increases self-similarity by $\text{tr } \Sigma_i$, and the expected RBF kernel in (3) elongates the standard spherical RBF kernel according to the covariance of the inputs. For both kernels, K_{exp} is a non-decreasing function of the eigenvalues of Σ_i and Σ_j—similarity generally increases with uncertainty.

2.2. Use in an SVM

The expected kernel can be used in a standard SVM [3], which is trained by solving the QP:

$$\begin{aligned} \underset{c,b,\xi}{\text{minimize}} \ & \frac{1}{2} c^T K_{\text{exp}} c + C \sum_{i=1}^{n} \xi_i \\ \text{s.t. } & y_i \left(c^T k_{\text{exp},i} + b\right) \geq 1 - \xi_i \\ & \xi_i \geq 0, \end{aligned} \tag{4}$$

where $k_{\text{exp},i} \triangleq \left[K_{\text{exp}}(p_{X_1}, p_{X_i}), \ldots, K_{\text{exp}}(p_{X_n}, p_{X_i})\right]^T$ is the ith column of the expected kernel matrix K_{exp}; and c, b, ξ, and C are respectively the standard weights, bias and slack variables and soft margin regularization parameter.

2.3. The SOCP SVM and Kernel Covariance

Shivaswamy et al. proposed a generalized form of the SVM to account for missing features [5]. Their approach incorporates the probabilistic uncertainty due to the missing features into the maximization of the margin:

$$\begin{aligned} \underset{c,b,\xi}{\text{minimize}} \ & \frac{1}{2} c^T \hat{K} c + C \sum_{i=1}^{n} \xi_i \\ \text{s.t. } & y_i \left(c^T \hat{k}_i + b\right) \geq 1 - \xi_i + \tau_i \|c\|_{\Sigma_i^k} \\ & \xi_i \geq 0 \end{aligned} \tag{5}$$

where $\hat{k}_i$ (ith column of $\hat{K}$) and Σ_i^k are the mean and covariance (in kernel space) of the ith column of a random kernel matrix whose randomness is a result of missing entries. The presence of the covariance-weighted norm $\|c\|_{\Sigma_i^k} \triangleq \sqrt{c^T \Sigma_j^k c}$ makes (5) an SOCP problem that generalizes the standard SVM: if $\Sigma_i^k = 0$ for $i = 1, \ldots, n$ (no uncertainty in training data), then (5) reduces to the standard primal SVM formulation, a QP problem. The user-specified parameter τ_i is related to the probability of correctly classifying the ith training point; in [5], the authors use $\tau_i = \tau$ for all i.

However, Shivaswamy et al. do not specify a feasible method to estimate the mean and covariance of missing kernel entries[1]. In their experiments they set $\hat{K}_{ij} = K(\hat{x}_i, \hat{x}_j)$, where $\hat{x}_i$ is an imputed estimated of x_i; furthermore, they assume spherical covariance, $\Sigma_i^k = I$ for $i = 1, \ldots, n$ [5, 6].

[1] A different SOCP problem is provided for the linear SVM, in which case the training data are imputed using expectation-maximization [5], but we focus here on the more general kernelized SOCP SVM.

The expected kernel in (1) is precisely the mean similarity in kernel space, and is theoretically well-suited to replace $\hat{K}$ in (5). However, this approach would not capture the variance Σ_i^k of the ith column of the random kernel. Using the proposed expected kernel for $\hat{K}$ and setting $\tau_i = 0$ in (5) results precisely in the proposed (QP) SVM given in (4). For $\tau_i > 0$ we can calculate the variances of the random kernels, and use it for a *diagonal* Σ_i^k in (5). We provide formulas for variances of the the inner product and RBF kernels below. To form a full covariance matrix Σ_i^k requires computing $\text{Cov}\left[K(X_i, X_j), K(X_k, X_j)\right]$, which we do not consider in this paper.

Inner Product Kernel Variance. Assuming independent $X_i \sim \mathcal{N}(x_i; m_i, \Sigma_i), i = 1, \ldots, n$, the variance of the inner product kernel $\text{Var}\left[X_i^T X_j\right]$ is given by

$$\begin{cases} \text{tr}\ (\Sigma_i \Sigma_j) + m_j^T \Sigma_i m_j + m_i^T \Sigma_j m_i & \text{if } i \neq j, \\ 2\,\text{tr}\ (\Sigma_i \Sigma_i) + 4 m_i^T \Sigma_i m_i & \text{if } i = j. \end{cases} \tag{6}$$

Proof. Falls directly from equations (5) (for $i \neq j$) and (6) (for $i = j$) due to Brown and Rutemiller [7]. The Gaussian assumption is necessary for the $i = j$ case.

RBF Kernel Variance. Assuming independent $X_i \sim \mathcal{N}(x_i; m_i, \Sigma_i), i = 1, \ldots, n$, the variance of the RBF kernel $\text{Var}\left[K^{\text{rbf}}(X_i, X_j)\right]$ is given by

$$\begin{aligned} &(\pi\gamma^{-1})^{d/2} \mathcal{N}(a; b, A + B + (2\gamma)^{-1} I) \\ &\quad - \gamma^{d/2} \left|4\pi(A + B + \gamma^{-1} I)\right|^{-1/2} \times \\ &\quad \mathcal{N}\left(a; b, \frac{1}{2}\left(A + B + \gamma^{-1} I\right)\right). \end{aligned} \tag{7}$$

Proof. Given in Appendix A.

3. EXPERIMENTS

We compare *(i)* the standard SVM with the expected kernel; *(ii)* the SOCP SVM using the expected kernel and the kernel variance given above; and *(iii)* the SOCP SVM as implemented by its authors [5, 6] with $\hat{K}_{ij} = K(\hat{x}_i, \hat{x}_j)$ where $\hat{x}_i$ is an imputed estimated of x_i using expectation maximization (EM) [8] and $\Sigma_i^k = I$.

We consider two benchmark datasets—`Ionosphere` and `Heart`—each randomly partitioned 9:1 into disjoint training and test sets. For each dataset, we randomly set as missing 5% to 40% of the entries of each training vector. Results are averaged over 10 independent runs of the experiment. The parameters γ (for RBF kernel), C (for the SVM and SOCP SVM), and $\tau_i = \tau$ are determined via 5-fold crossvalidation.

4. RESULTS

Classification error as a function of missing feature fraction is plotted for each kernel type and dataset in Figure 1. The error rate of the SVM if there had been no missing features is shown for comparison. The results show that using the expected kernel by itself (without the SOCP form) provides roughly the same performance as using the more complicated SOCP formulation, and was dramatically faster to train; 20 times faster on the small (`Heart` dataset, $n = 240$) using a Intel Core 2 machine. Further experiments are needed to judge whether there are practical circumstances where the SOCP is worthwhile, with either the pre-imputed implementation provided by Shivaswamy et al. [5, 6] or with the expected kernel as considered here.

A. DERIVATION OF VARIANCE OF RBF KERNEL

Let X and Y be independent Gaussian random vectors with $X \sim \mathcal{N}(x; a, A)$ and $Y \sim \mathcal{N}(y; b, B)$. Express the RBF kernel as

$$K^{\text{rbf}}(X, Y) = (2\pi\gamma^{-1})^{d/2} \mathcal{N}(X; Y, \gamma^{-1} I),$$

and note that

$$\left[\mathcal{N}(x; y, R)\right]^2 = |4\pi R|^{-1/2} \mathcal{N}(x; y, \frac{1}{2} R).$$

Compute the second moment

$$\begin{aligned} \text{E}\left[K^{\text{rbf}}(X, Y)^2\right] &= \iint\limits_{x\ y} \mathcal{N}(x; a, A) \mathcal{N}(y; b, B) K^{\text{rbf}}(x, y)^2 \\ =&(2\pi\gamma^{-1})^d \iint\limits_{x\ y} \mathcal{N}(x; a, A) \mathcal{N}(y; b, B) \left[\mathcal{N}(x; y, \gamma^{-1} I)\right]^2 \\ =&(\pi\gamma^{-1})^{d/2} \iint\limits_{x\ y} \mathcal{N}(x; a, A) \mathcal{N}(y; b, B) \mathcal{N}(x; y, (2\gamma)^{-1} I) \\ =&(\pi\gamma^{-1})^{d/2} \mathcal{N}(a; b, A + B + (2\gamma)^{-1} I). \end{aligned} \tag{A}$$

Then from (3), compute the square of the expected value

$$\begin{aligned} \text{E}\left[K^{\text{rbf}}(X, Y)\right]^2 &= \left(\gamma^{d/2} \mathcal{N}(a; b, A + B + \gamma^{-1} I)\right)^2 \\ =&\gamma^{d/2} \left|4\pi(A + B + \gamma^{-1} I)\right|^{-1/2} \times \\ &\mathcal{N}\left(a; b, \frac{1}{2}\left(A + B + \gamma^{-1} I\right)\right). \end{aligned} \tag{B}$$

Then, the variance of the RBF kernel in (7) is the difference of equations (A) and (B).

B. REFERENCES

[1] A. Farhangfar, L. Kurgan, and J. Dy, "Impact of imputation of missing values on classification error for discrete data," *Pattern Recognition*, vol. 41, pp. 3692–3705, 2008.

[2] J. L. Schafer and J. W. Graham, "Missing data: Our view of the state of the art," *Psychological Methods*, vol. 7, no. 2, pp. 147–177, 2002.

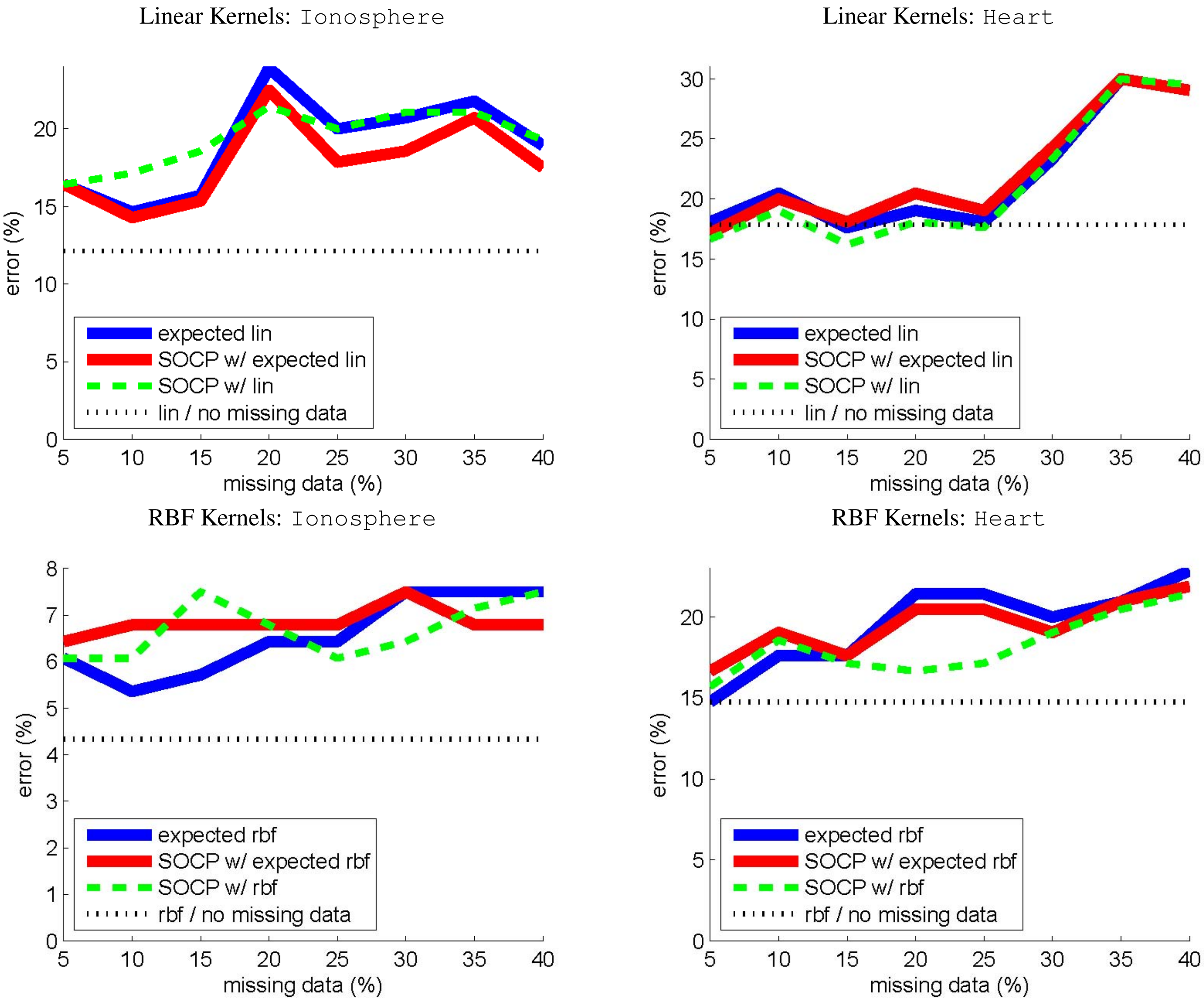

Fig. 1. SVM classification error using RBF kernel for Ionosphere (left) and Heart (right).

[3] H. S. Anderson, M. R. Gupta, E. Swanson, and K. Jamieson, "Channel-robust classifiers," *IEEE Trans. Sig. Proc*, vol. 59, no. 4, pp. 1421.

[4] H. Kashima and K. Tsuda nad A. Inokuchi, "Marginalized kernels between labeled graphs," *Proc. Intl. Conf. Machine Learning*, 2003.

[5] P. K. Shivaswamy, C. Bhattacharyya, and A. J. Smola, "Second order cone programming approaches for handling missing and uncertain data," *J. Mach. Learn. Res.*, vol. 7, pp. 1283–1314, 2006.

[6] Email conversation with Pannanga Shivaswamy, January 2011.

[7] G. G. Brown and H. C. Rutemiller, "Means and variances of stochastic vector products with applications to random linear models," *Man. Sci.*, vol. 24, no. 2, pp. 210–216, 1977.

[8] M. R. Gupta and Y. Chen, *Theory and Use of the EM Method*, Foundations and Trends in Signal Processing Series by NOW Publishers, 2011.