Binning in Gaussian Kernel Regularization

Tao Shi and Bin Yu

Department of Statistics, University of California

Abstract

Gaussian kernel regularization is widely used in the machine learning literature and proven successful in many empirical experiments. The periodic version of the Gaussian kernel regularization has been shown to be minimax rate optimal in estimating functions in any finite order Sobolev spaces. However, for a data set with n points, the computation complexity of the Gaussian kernel regularization method is of order $O(n^3)$.

In this paper we propose to use binning to reduce the computation of Gaussian kernel regularization in both regression and classification. For the periodic Gaussian kernel regression, we show that the binned estimator achieves the same minimax rates of the unbinned estimator, but the computation is reduced to $O(m^3)$ with m as the number of bins. To achieve the minimax rate in the k-th order Sobolev space, m needs to be in the order of $O(kn^{1/(2k+1)})$, which makes the binned estimator computation of order $O(n)$ for $k = 1$ and even less for larger k. Our simulations show that the binned estimator (binning 120 data points into 20 bins in our simulation) provides almost the same accuracy with only 0.4% of computation time.

For classification, binning with the $L2$-loss Gaussian kernel regularization and the Gaussian kernel Support Vector Machines is tested in a polar cloud detection problem. With basically the same computation time, the $L2$-loss Gaussian kernel regularization on 966 bins achieves better test classification rate (79.22%) than that (71.40%) on 966 randomly sampled data. Using the OSU-SVM Matlab package, the SVM trained on 966 bins has a comparable test classification rate as the SVM trained on 27,179 samples, but reduces the training time from 5.99 hours to 2.56 minutes. The SVM trained on 966 randomly selected samples has a similar training time as and a slightly worse test classification rate than the SVM on 966 bins, but has 67% more support vectors so takes 67% longer to predict on a new data point. The SVM trained on 512 cluster centers from the k-mean algorithm reports almost the same test classification rate and a similar number of support vectors as the SVM on 512 bins, but the k-mean clustering itself takes 375 times more computation time than binning.

KEY WORDS: Asymptotic minimax risk, Binning, Gaussian reproducing kernel, Regularization, Rate of convergence, Sobolev space, Support Vector Machines.

1 Introduction

The method of regularization has been widely used in the nonparametric function estimation problem. The problem begins with estimating a function f using data (x_i, y_i), $i = 1, \cdots, n$, from a nonparametric regression model:

$$y_i = f(x_i) + \epsilon_i,\ i = 1, \cdots, n, \tag{1}$$

where $x_i \in R^d$, $i = 1, \cdots, n$, are regression inputs or predictors, y_i's are the responses, and ϵ_i's are *i.i.d.* $N(0, \sigma^2)$ Gaussian noises. The method of regularization takes the form of finding $f \in \mathcal{F}$ that minimizes

$$L(f, \text{data}) + \lambda J(f) \tag{2}$$

where L is an empirical loss, often taken to be the negative log-likelihood. $J(f)$ is the penalty functional, usually a quadratic functional corresponding to a norm or semi-norm of a Reproducing Kernel Hilbert Space (RKHS) $\mathcal{F}$. The regularization parameter λ trades off the empirical loss with the penalty $J(f)$. In the regression case we may take $L(f,\text{data}) = \sum_{i=1}^{n}(y_i - f(x_i))^2$ and the penalty functional $J(f)$ usually measures the smoothness.

In the nonparametric statistics literature, the well-known smoothing spline (cf Wahba 1990) is an example of regularization method. The reproducing kernel Hilbert space used in smoothing spline is a Hilbert Sobolev space and the penalty $J(f) = \int [f^{(m)}(x)]^2 dx$ is the norm or semi-norm in the space. The reproducing kernel of the Hilbert Sobolev space was nicely covered in Wahba (1990), and the commonly used cubic spline corresponds to the case $m = 2$.

In the machine learning literature, Support Vector Machines (SVM) and regularization networks, which are both regularization methods, have been used successfully in many practical applications. Smola et al (1998), Wahba (1999), and Evgeniou et al (2000) make the connection between both methods and the methods of regularization in the Reproducing Kernel Hilbert Space (**RKHS**). SVM uses a hinge loss function $L(f, data) = \sum_{i=1}^{n}(1 - y_i f(x_i))^+$ in (2) with labels y_i coded as $\{-1, 1\}$ in the two-class case. The penalty functional $J(f)$ used in SVM is the norm of the **RKHS** (see Vapnik 1995 and Whaba et al 1999 for details).

One particularly popular reproducing kernel used in the machine learning literature is the Gaussian kernel, which is defined as $G(s,t) = (2\pi)^{-1/2}\omega^{-1}exp((s-t)^2/2\omega^2)$. Girosi et al (1993) and Smola et al (1998) showed that the Gaussian kernel corresponds to the penalty functional

$$J_g(f) = \sum_{m=0}^{\infty} \frac{\omega^{2m}}{2^m m!} \int_{-\infty}^{\infty} [f^{(m)}(x)]^2 dx. \tag{3}$$

Smola et al (1998) also introduced the periodic Gaussian reproducing kernel for estimating 2π-periodic functions in $(-\pi, \pi]$ as the kernel corresponding to penalty functional

$$J_{pg}(f) = \sum_{m=0}^{\infty} \frac{\omega^{2m}}{2^m m!} \int_{-\pi}^{\pi} [f^{(m)}(x)]^2 dx \tag{4}$$

Using the equivalence between the nonparametric regression and Gaussian white noise model shown in Brown and Low (1996), Lin and Brown (2004) showed asymptotic properties of the regularization using a periodic Gaussian kernel. The periodic Gaussian kernel regularization is rate optimal in estimating functions in all finite order Sobolev spaces. It is also asymptotically minimax for estimating functions in the infinite order Sobolev Space and the space of analytic functions. These asymptotic results on the periodic Gaussian kernel gave a partial explanation of the success of the Gaussian reproducing kernel in practice. In section 2, we describe the periodic Gaussian kernel regularization in nonparametric regression setup and review the asymptotic results, which will be compared to the binning results in section 4. Although having good statistical properties, the Gaussian kernel regularization method is computationally very expensive, usually of order $O(n^3)$ on n data points. It is computationally infeasible when n is too large.

In this paper, motivated by the application of binning technique in nonparametric regression (cf Hall et al 1998), we study the effect of binning in the periodic Gaussian kernel regularization. We first show the eigen structure of the periodic Gaussian kernel in the finite sample case, then the eigen structure is used to prove the asymptotic minimax rates of the binned periodic Gaussian kernel regularization estimator. The results on the kernel matrix are given in section 3.

In section 4, we show the binned estimator achieves the same minimax rates of the unbinned estimator, while the computation is reduced to $O(m^3)$ with m as the number of bins. To achieve the minimax rate in the k-th order Sobolev space, m needs to be in the order of $O(kn^{1/(2k+1)})$, which makes the binned estimator computation to be $O(n)$ for $k = 1$ and even less for larger k. For estimating functions in the Sobolev space of infinite order, the number of bins m only needs to be in the order of $O(\sqrt{log(n)})$ to achieve the minimax risk. For the simple average binning, the optimal regularization parameter λ_B for binned data has a simple relationship with the optimal λ for the unbinned data, $\lambda_B \approx m\lambda/n$ and ω stays the same. In practice, choosing parameters (λ_B, ω) by Mallow's C_p statistics achieves the asymptotic rates.

In section 5, experiments are carried out to assess the accuracy and the computation reduction of the binning scheme in regression and classification problems. We first run simulations to test binning periodic Gaussian kernel Regularization in the nonparametric regression setup. Four periodic functions with different order of smoothness are used the simulation. Comparing to the unbinned estimators on 120 data points, the binned estimators (6 data in each bin) provide the same accuracy, but requires only 0.4% of computation.

For classification, binning on the $L2$-loss Gaussian kernel regularization and the Gaussian kernel Support Vector Machines are tested in a polar cloud detection problem. With the same computation time, the $L2$-loss Gaussian kernel regularization on 966 bins achieves better accuracy (79.22%) than that (71.40%) on 966 randomly sampled data. Using the OSU-SVM Matlab package, the SVM trained on 966 bins has a comparable test classification rate as the SVM trained on 27,179 samples, and reduces the training time from 5.99 hours to 2.56 minutes. The SVM trained on 966 randomly selected samples has a similar training time as and a slightly worse test classification rate than the SVM on 966 bins, but has 67% more support vectors so takes 67% longer to predict on a new data point.

Compare to k-mean clustering, another possible SVM training sample-size reduction scheme proposed in Feng and Mangasarian (2001), binning is much faster than k-mean clustering. The SVM trained on 512 cluster centers from the k-mean algorithm reports almost the same test classification rate and a similar number of support vectors as the SVM on 512 bins, but the k-mean clustering itself takes 375 times more computation time than binning. Therefore, for both the regression and classification in practice, binning Gaussian kernel regularization reduces the computation and keeps the estimation or classification accuracy. Section 6 contains summaries and discussions.

2 Periodic Gaussian Kernel Regularization

Lin and Brown (2004) studied the asymptotic properties of the periodic Gaussian kernel regularization in estimating 2π-periodic functions on $(-\pi, \pi]$ in three different function spaces. Using the asymptotic equivalence between the nonparametric regression and the Gaussian white noise model (see Brown and Low 1996), the asymptotic properties of the periodic Gaussian kernel regularization are proved in the Gaussian white noise model. In this section, we introduce the periodic Gaussian regularization and review the asymptotic results by Lin and Brown (2004) in the nonparametric regression setting.

2.1 Nonparametric Regression

In this paper, we consider estimating periodic function on $(0, 1]$ using periodic Gaussian regularization. With data (x_i, y_i), $i = 1, \cdots, n$, observed from model (1) at equal space designed points x_i's, the method of periodic Gaussian kernel regularization with $L2$ loss estimates f by a periodic function $\hat{f}_\lambda$ that minimizes

$$\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda J_{pg}(f) \tag{5}$$

where $J_{pg}(f)$ is the norm of the corresponding **RKHS** $\mathcal{F}_K$ of the periodic Gaussian kernel (Smola et al, 1998)

$$K(s,t) = 2\sum_{l=0}^{\infty}\exp(-l^2\omega^2/2)\cos(2\pi l(s-t)). \tag{6}$$

The theory of reproducing kernel Hilbert space guarantees that the solution to (5) over $\mathcal{F}_K$ falls in the finite dimensional space spanned by $\{K(x_i, \cdot),\ i = 1, \cdots, n\}$ (see Wahba 1990 for an introduction to the theory of reproducing kernels). Therefore, we can write the solution to (5) as $\hat{f}(x) = \sum_{i=1}^{n} \hat{c}_i K(x_i, x)$ and the minimization problem can then be solved in this finite dimensional space. Using the finite expression, (5) becomes

$$\min_c [(y - G^{(n)}c)^T(y - G^{(n)}c) + \lambda c^T G^{(n)} c], \tag{7}$$

where $y = (y_1, \cdots, y_n)^T$, $c = (c_1, \cdots, c_n)^T$, and $G^{(n)}$ as a $n \times n$ matrix $K(x_i, x_j)$. The solution is $\hat{c} = (G^{(n)} + \lambda I)^{-1}y$ with I being a $n \times n$ identity matrix. The fitted values are $\hat{y} = G^{(n)}\hat{c} = G^{(n)}(G^{(n)} + \lambda I)^{-1}y \triangleq Sy$, which is a linear estimator. The asymptotic risk of this estimator is shown in the next section.

2.2 Asymptotic Properties

We briefly review the asymptotic results shown in Lin and Brown (2004) in this section and compare them to the binned estimators in section 4. The asymptotic risk of periodic Gaussian regularization is studied in estimating periodic function from three types of function spaces: spaces of Sobolev ellipsoid of finite order, ellipsoid spaces of analytic functions, and Sobolev spaces of infinite order, which are defined as follows. (Instead of working with 2π-periodic functions on $(-\pi, \pi]$, we study periodic functions on (0,1] in this paper.)

The first type of function space, k-th order Sobolev ellipsoid $H^k(Q)$, is defined as

$$H^k(Q) = \{f \in L^2(0,1) : f \text{ is periodic}, \int_0^1 [f(t)]^2 + [f^{(k)}(t)]^2 dt \leq Q\}. \tag{8}$$

It has an alternative definition in the Fourier space as:

$$H^k(Q) = \{f : f(t) = \sum_{l=0}^{\infty} \theta_l \phi_l(t), \sum_{l=0}^{\infty} \gamma_l \theta_l^2 \leq Q, \gamma_0 = 1, \gamma_{2l-1} = \gamma_{2l} = l^{2k} + 1\}, \tag{9}$$

where $\phi_0(t) = 1$, $\phi_{2l-1}(t) = \sqrt{2}\sin(2\pi lt)$, $\phi_{2l}(t) = \sqrt{2}\cos(2\pi lt)$ are the classical trigonometric basis in $L^2(0,1)$ and $\theta_l = \int_0^1 f(t)\phi_l(t)dt$ is the corresponding Fourier coefficient.

The second function space being considered is the ellipsoid space of analytic functions $A_\alpha(Q)$, which corresponds to (9) with the exponentially increasing sequence $\gamma_l = exp(\alpha l)$. The third function space is the infinite order Sobolev space $H_\omega^\infty(Q)$, which corresponds to (9) with the sequence $\rho_0 = 1$ and $\gamma_{2l-1} = \gamma_{2l} = e^{l^2\omega^2/2}$. Note that the penalty functional J_{pg} of periodic Gaussian kernel regularization is the norm of $H_\omega^\infty(Q)$.

The asymptotic risk of $\hat{y}$ is determined by the tradeoff between the variance and the bias. The asymptotic variance of $\hat{y} = G^{(n)}\hat{c} = G^{(n)}(G^{(n)} + \lambda I)^{-1}y$ depends only on λ and ω, while $G^{(n)}$ denotes matrix $K(x_i, x_j)$. In the mean time, the asymptotic bias depends not only on λ and ω, but also on the function f itself. Lin and Brown (2004) proved the following lemma using the equivalence between the nonparametric regression and the Gaussian white noise model (Brown an Law 1996).

Lemma 1 *(Lin and Brown 2004) In nonparametric regression, the solution $\hat{y}$ to the periodic Gaussian kernel regularization problem (7) has an asymptotic variance:*

$$\frac{1}{n}\sum var(\hat{y}_i) = (1/n)\sum(1 + \lambda\beta_l)^{-2} \sim 2\sqrt{2}\omega^{-1}n^{-1}(-\log\lambda)^{1/2}, \tag{10}$$

for $\beta_l = exp(l^2\omega^2/2)$ *as* λ *goes to zero. The asymptotic bias is:*

$$\frac{1}{n}\sum bias^2(\hat{y}_l) \sim \sum \lambda^2\beta_l^2(1+\lambda\beta_l)^{-2}\theta_l^2. \tag{11}$$

when estimating function $f(t) = \sum_{l=0}^{\infty}\theta_l\phi_l(t)$.

Based on (10) and (11), following asymptotic results about the periodic Gaussian kernel regularization are shown.

Lemma 2 *(Lin and Brown 2004) For estimating functions in the* k*-th order Sobolev space* $H^k(Q)$, *the periodic Gaussian kernel regularization has a minimax risk:*

$$(2k+1)k^{-2k/(2k+1)}Q^{1/(2k+1)}n^{-2k/(2k+1)},$$

achieved when $log(n/\lambda)/\omega^2 \sim (knQ)^{2/(2k+1)}/2$. *The minimax rate for estimating functions in* $A_\alpha(Q)$ *is*

$$2n^{-1}\alpha^{-1}(\log n),$$

and the rate is

$$2\sqrt{2}\omega^{-1}n^{-1}(\log n)^{1/2}$$

for estimating functions in $H_\omega^\infty(Q)$.

It is well known that the asymptotic minimax risk over $H^k(Q)$ is $[2k/(k+1)]^{2k/(2k+1)}(2k+1)^{1/(2k+1)}Q^{1/(2k+1)}n^{-2k/(2k+1)}$. If we calculate the efficiency of the periodic Gaussian kernel regularization in terms of sample sizes need to achieve the same risk, the efficiency goes to one when the function gets smoother. Therefore, the estimator is rate optimal in this case. For estimating functions in $A_\alpha(Q)$ and $H_\omega^\infty(Q)$, the periodic Gaussian kernel regularization achieves the minimax risk (see Johnstone 1998 for the proof of minimax risk in $A_\alpha(Q)$). These asymptotic rates in Lemma 2 are compared with the binning results in section 4.

3 The Eigen Structure of the Projection Matrix

Instead of working with the Gaussian white noise model, we directly prove Lin and Brown's asymptotic results in the nonparametric regression model in this section. Although the results stated in section 2.2 are proved more easily in the Gaussian white noise model than in the regression model, knowing the eigen structure of the projection matrix S (defined as $\hat{y} = G^{(n)}(G^{(n)} + \lambda I)^{-1}y \triangleq Sy$ in section 2.1) helps us understand the binned estimators in section 4. To study the variance-bias trade-off of the periodic Gaussian regularization, we first derive the eigen-values and eigen-vectors of $G^{(n)} = K(x_i, x_j)$ and make the connections between them to the functional eigen-values and eigen-functions of the reproducing kernel $K(\cdot, \cdot)$.

For a general reproducing kernel $R(\cdot, \cdot)$ that satisfies $\int\int R^2(x, y)dxdy < \infty$, there exist an orthonormal sequence of eigen-functions ϕ_1, ϕ_2, $\cdots$, and eigen-values $\rho_1 \geq \rho_2 \geq \cdots \geq 0$, with

$$\int_a^b R(s,t)\phi_l(s)ds = \rho_l\phi(t), \quad l = 1, 2, ... \tag{12}$$

and $R(s,t) = \sum_{l=1}^{\infty} \rho_l\phi_l(s)\phi_l(t)$. When equally spaced points $x_1, \cdots, x_n$ are taken in $(a, b]$, we get a Gram matrix $R_{i,j}^{(n)} = R(x_i, x_j)$. The eigen-vectors and eigen-values of $R^{(n)}$ are defined as a sequence of orthonormal n by 1 vectors v_1, $\cdots$, v_n and values $d_1 \geq \cdots \geq> 0 d_n$, that satisfy

$$R^{(n)}V_l^{(n)} = d_l^{(n)}V_l^{(n)}, \quad l = 1, 2, ...n \tag{13}$$

and $R^{(n)} = \sum_{l=1}^{n} d_l^{(n)}V_l^{(n)}V_l^{(n)^T}$. The eigen-values $d_l^{(n)}$ have limits: $lim_{n\to\infty} d_l^{(n)}(b-a)/n = \rho_l$ (c.f. Williams and Seeger 2000).

On $(0, 1]$, the eigen-functions of the periodic Gaussian kernel K are the classical trigonometric basis functions $\phi_0(t) = 1$, $\phi_{2l-1}(t) = \sqrt{2}\sin(2\pi lt)$, $\phi_{2l}(t) = \sqrt{2}\cos(2\pi lt)$, with the corresponding eigen-values $\rho_0 = 1$ and $\rho_{2l-1} = \rho_{2l} = exp(-l^2\omega^2/2)$ (For notation simplicity, the labels of eigen-values and eigen-functions start from 0 instead of 1). It is straightforward to see the eigen function decomposition when we rewrite $K(s,t)$ as:

$$\begin{aligned} K(s,t) &= 2\sum_{l=0}^{\infty}\exp(-l^2\omega^2/2)\cos(2\pi l(s-t)) \\ &= \sum_{l=0}^{\infty} e^{-l^2\omega^2/2}[\sqrt{2}\sin(2\pi ls)\sqrt{2}\sin(2\pi lt)+\sqrt{2}\cos(2\pi ls)\sqrt{2}\cos(2\pi lt)] \\ &= \sum_{l=0}^{\infty}\rho_l\phi_l(s)\phi_l(t) \end{aligned}$$

where $\phi_l(t)$'s are orthonormal on $(0,1]$. When n equally spaced data points are taken over $(0,1]$, such as $x_i = -\frac{1}{2n}+\frac{i}{n}$, the Gram matrix $G^{(n)} = K(x_i,x_j)$ has the following property:

Theorem 1 *The Gram matrix $G^{(n)} = K(x_i,x_j)$ at equal-spaced data points $x_1,\cdots,x_n$ over $(0,1]$ has eigen-vectors $V_0^{(n)}, V_1^{(n)},\cdots,V_{n-1}^{(n)}$ (indexed from 0 to $n-1$):*

$$V_0^{(n)} = \sqrt{1/n}(1,\cdots,1)^T = \sqrt{1/n}(\phi_0(x_1),\cdots,\phi_0(x_n))^T,$$

$$V_l^{(n)} = \sqrt{2/n}(\sin(2\pi hx_1),\cdots,\sin(2\pi hx_n))^T = \sqrt{1/n}(\phi_l(x_1),\cdots,\phi_l(x_n))^T, \quad \textit{for odd } l,$$

$$V_l^{(n)} = \sqrt{2/n}(\cos(2\pi hx_1),\cdots,\cos(2\pi hx_n))^T = \sqrt{1/n}(\phi_l(x_1),\cdots,\phi_l(x_n))^T \quad \textit{for even } l,$$

where $h = \lceil (l+1)/2\rceil$, $l = 1,\cdots,n-1$, and $\lceil a\rceil$ stands for the integer part of a. Their corresponding eigen-values are:

$$d_0^{(n)} = 2n\sum_{k=0}^{\infty}(-1)^k\rho_{2kn}$$

$$d_l^{(n)} = n\{\rho_l + \sum_{k=1}^{\infty}(-1)^k[\rho_{kn+h}+(-1)^{l-2h}\rho_{kn-h}]\}$$

The proof is given in the appendix. This theorem shows that the eigen-vector $V_l^{(n)}$ is exactly the evaluation of eigen-function $\phi_l(\cdot)$ at $x_1,\cdots,x_n$, scaled by $\sqrt{(1/n)}$. It is worth to point out that this exact relationship between eigen-functions and eigen-vectors does not generally hold for other kernels and data distributions. For general kernels and data distributions, one can only prove that the eigen-vectors converges to the corresponding eigen-functions as the sample size goes to infinity. Therefore, the eigen-vectors can not be explicitly written out in the finite sample case.

With the eigen decomposition of $G^{(n)}$, we now study the variance-bias trade-off of the periodic Gaussian kernel regularization. Using the matrix notation, let $V^{(n)} \triangleq (V_0^{(n)}, \cdots, V_{n-1}^{(n)})$ and let $D^{(n)} \triangleq diag(d_0^{(n)}, \cdots, d_{n-1}^{(n)})$ be an n by n diagonal matrix , then $G^{(n)} = V^{(n)} D^{(n)} V^{(n)T}$.

For the variance term, recall $S = G^{(n)}(G^{(n)} + \lambda I)^{-1}$, we have $S = V^{(n)} diag(\frac{d_l^{(n)}}{d_l^{(n)}+\lambda}) V^{(n)T}$. Therefore, the variance term is:

$$\frac{1}{n}\sum var(\hat{y}_i) = \frac{1}{n} trace(S^T S) = \frac{1}{n}\sum_{l=0}^{n-1}(\frac{d_l^{(n)}}{d_l^{(n)}+\lambda})^2 = \frac{1}{n}\sum_{l=0}^{n-1}(\frac{d_l^{(n)}/n}{d_l^{(n)}/n+\lambda/n})^2.$$

Since $lim_{n\to\infty} d_l^{(n)}/n = \rho_l$ for $l > 0$ and $\rho_l = 1/\beta_l$, we get

$$\frac{1}{n}\sum var(\hat{y}_i) \sim \frac{1}{n}\sum(\frac{\rho_l}{\rho_l + (\lambda/n)})^2 = \frac{1}{n}\sum(1+\beta_l(\frac{\lambda}{n}))^{-2},$$

which is the same as the asymptotic variance shown in (10).

For the bias term, we expand $f(t)$ as $f(t) = \sum_{l=0}^{\infty} \theta_l \phi_l(t)$. Using the relationship between $V^{(n)}$ and $\phi(\cdot)$ in Theorem 1, we can write vector $F = (f(x_1), \cdots, f(x_n))^T$ as

$$F = \sum_{l=0}^{n-1} \Theta_l^{(n)} V_l^{(n)} = V^{(n)}\Theta^{(n)},$$

where

$$\Theta_0^{(n)} = \sqrt{n}\sum_{k=0}^{\infty}(-1)^k \theta_{2kn},$$

$$\Theta_l^{(n)} = \sqrt{n}\{\theta_l + \sum_{k=1}^{\infty}(-1)^k[\theta_{kn+h} + (-1)^{l-2h}\theta_{kn-h}]\},$$

for $1 \leq l \leq n-1$ and $h = \lceil (l+1)/2 \rceil$. Thus, the bias term is written as:

$$\begin{aligned}
\frac{1}{n}\sum Bias^2(\hat{y}_i) &= \frac{1}{n}((S-I)F)^T((S-I)F) \\
&= \frac{1}{n}(V^{(n)} diag(\frac{\lambda}{d_l^{(n)}+\lambda})V^{(n)T}F)^T(V^{(n)} diag(\frac{\lambda}{d_l^{(n)}+\lambda})V^{(n)T}F) \\
&= \frac{1}{n}\sum_{l=0}^{n-1}(\frac{\Theta_l^{(n)}\lambda}{d_l^{(n)}+\lambda})^2 = \frac{1}{n}\sum_{l=0}^{n-1} n(\frac{\Theta_l^{(n)}}{\sqrt{n}})^2(\frac{\lambda/n}{d_l^{(n)}/n+\lambda/n})^2
\end{aligned}$$

$$
\begin{aligned}
&\sim \sum \theta_l^2 (\frac{\lambda/n}{\rho_l + \lambda/n})^2 \\
&= \sum \theta_l^2 (\frac{\beta_l \lambda/n}{1 + \beta_l \lambda/n})^2,
\end{aligned}
$$

since $lim_{n\to\infty}\Theta_l^{(n)}/\sqrt{n} = \theta_l$, $lim_{n\to\infty} d_l^{(n)}/n = \rho_l$, and $\rho_l = 1/\beta_l$.

Both the variance and bias are the same as in Lemma 1 derived by using the Gaussian white noise model in Lin and Brown (2004). Although it is easier to prove Lemma 1 and 2 in the Gaussian white noise model than through the eigen expansion derived above, binned estimators in the nonparametric regression setup do not directly convert to the Gaussian white noise model. Therefore, the eigen expansion is used to prove the asymptotic properties of binning the periodic Gaussian kernel regularization in the next section.

4 Binning Periodic Gaussian Kernel Regularization

Although the periodic Gaussian regularization method has good asymptotic properties, the computation of the estimator $\hat{y} = G^{(n)}(G^{(n)} + \lambda I)^{-1} y$ is very expensive, taking $O(n^3)$ to invert the n by n matrix $G^{(n)} + \lambda I$. When the sample size gets large, the computation is not even feasible. In nonparametric regression estimation, Hall et al (1998) studied the binning technique. In this section, we use the explicit eigen structure of the periodic Gaussian kernel to study the effect of binning on the asymptotic properties of periodic Gaussian regularization.

4.1 Simple Binning Scheme

Let us take equally spaced n data points in $(0, 1]$, say $x_i = -\frac{1}{2n} + \frac{i}{n}$. Without loss of generality, we assume the number of design point n equals $m \times p$, while m is the number of bins and p is number of data points in each bin. Using equally spaced binning scheme, let us denote the centers of bins as $\bar{x}_j = (x_{(j-1)\times p+1} + \cdots + x_{(j-1)\times p+p})/p$ and the average of observations in each bin as $\bar{y}_j = (y_{(j-1)\times p+1} + \cdots + y_{(j-1)\times p+p})/p$, for $j = 1, \cdots, m$. When we apply the periodic Gaussian regularization to the binned data, the estimated function is in the form of $\hat{f}(x) = \sum_{j=1}^{m} \hat{c}_j K(x, \bar{x}_j)$, where $\hat{c}$ is the solution of

$$\min_c (\bar{y} - G^{(m)}c)^T(\bar{y} - G^{(m)}c) + \lambda_B c^T G^{(m)} c, \tag{14}$$

with $G^{(m)}_{i,j} = K(\bar{x}_i, \bar{x}_j)$, $\bar{y} = (\bar{y_1}, \cdots, \bar{y_m})$ and λ_B as the regularization parameter. Similar to the estimator derived in section 2.1, the solution of (14) is $\hat{c} = (G^{(m)} + \lambda_B I)^{-1}\bar{y}$. With this explicit form of the binned estimator, we study its asymptotic properties next. Let

$$B^{(m,n)} = \begin{pmatrix} m/n & \cdots & m/n & 0 & \cdots & & & \cdots & 0 \\ 0 & \cdots & 0 & m/n & \cdots & m/n & 0 & \cdots & 0 \\ & \cdots & & & & & & \cdots & \\ 0 & \cdots & & & \cdots & 0 & m/n & \cdots & m/n \end{pmatrix}_{m\times n}. \tag{15}$$

The binned estimator can be written as $\hat{y} = G^{(n,m)}(G^{(m)} + \lambda_B I)^{-1}B^{(m,n)}y = S_B y$ with $G^{(n,m)}_{i,j} = K(x_i, \bar{x}_j)$ being a n by m matrix. From this expression, it is straightforward to see that the computation is reduce to $O(m^3)$ since the matrix inversion is taken on an m by m matrix now. The additional computation for binning the data itself is around $O(n)$, which is trivial comparing to the matrix inversion computation $O(n^3)$.

Using this matrix expression, The variance of the estimator can be written as:

$$\frac{1}{n}\sum var(\hat{y_i}) = \frac{1}{n} trace(S_B^T S_B) = \frac{1}{n} trace(S_B S_B^T) \tag{16}$$

Based on the following proposition, the variance term can be explicitly written out using the eigen-decomposition of S_B.

Proposition 1 *Suppose $n = mp$, $x_i = -\frac{1}{2n} + \frac{i}{n}$, and $\bar{x}_j = (x_{(j-1)\times p+1} + \cdots + x_{(j-1)\times p+p})/p$. The eigen-vectors $V^{(m)}$ of $G^{(m)}$ and the eigen-vectors $V^{(n)}$ of $G^{(n)}$ satisfy*

$$G^{(n,m)}V_k^{(m)} = d_k^{(m)}\sqrt{\frac{n}{m}}V_k^{(n)} \ \textit{for} \ k = 0, 1, \cdots, m.$$

The proof is provided in the appendix. This proposition shows that the eigen-vector of $G^{(m)}$ are projected to the corresponding eigen-vector of $G^{(n)}$ by the matrix $G^{(n,m)}$ (Unfortunately, this property does not hold for general kernels). Following this relationship, the asymptotic variance of the binned estimator is provided in the following theorem:

Theorem 2 *The asymptotic variance of the binned estimator* $\hat{y} = G^{(n,m)}(G^{(m)}+\lambda_B I)^{-1}B^{(m,n)}y$ *in the equally spaced binning scheme is:*

$$\frac{1}{n}\sum var(\hat{y}_i) \sim \frac{1}{n}\sum(1+\frac{\beta_l\lambda_B}{m})^{-2} \sim 2\sqrt{2}w^{-1}n^{-1}(-\log(\lambda_B/m))^{1/2}, \quad (17)$$

as $m \to \infty$, $n \to \infty$ *and* $\lambda_B \to 0$. *The expression is the same as the asymptotic variance of the original estimator when* $\lambda_B = m\lambda/n$.

See the proof in the appendix. Now we focus on the bias term, which depends not only on the projection operation, but also on the smoothness of the underline function f itself. We have the following theorem for the bias term:

Theorem 3 *In the equally spaced binning scheme, as* $m \to \infty$, $n \to \infty$, $m/n \to 0$ *and* $\lambda_B \to 0$, *the bias of the binned estimator is:*

$$\frac{1}{n}\sum Bias^2(\hat{y}_i) \sim \sum_{j=0}^{m-1}\theta_j^2(\frac{\beta_j\lambda_B/m}{1+\beta_j\lambda_B/m})^2 + \sum_{j=m}^{\infty}\theta_j{}^2. \quad (18)$$

when estimating function $f(t) = \sum_{l=0}^{\infty}\theta_l\phi_l(t)$.

The theorem is proved in the appendix. With the asymptotic variance and bias obtained, we show in the next section that the asymptotic minimax rates of the periodic Gaussian kernel regularization are kept after binning the data.

4.2 Asymptotic Rates of Binned Estimators

In this section, we study the asymptotic rates of the binned periodic Gaussian kernel regularization in estimating functions in spaces defined in section 2.2. We start with the infinite order Sobolev space. As shown in the next theorem, the binned estimator also achieves the minimax rate as the original estimator does in this space.

Theorem 4 *The minimax rate of the binned estimator* $\hat{y} = G^{(n,m)}(G^{(m)} + \lambda_B I)^{-1}B^{(m,n)}y$ *for estimating functions in the infinite order Sobolev space* $H_w^\infty(Q)$ *is:*

$$\min \max_{\theta \in H_w^\infty(Q)} E[\frac{1}{n}(\hat{y}-y)^T(\hat{y}-y)] \sim 2\sqrt{2}w^{-1}n^{-1}(\log n)^{1/2},$$

which is the same rate of the unbinned estimator. This rate is achieved when $m/n \to 0$*, and* m *is large enough so that* $w^2m^2/2 > \log(4m/\lambda_B)$*. The parameter* $\lambda_B = \lambda_B(n,m)$ *satisfies* $\log(m/\lambda_B) \sim \log n$*,* $\lambda_B/m = o(n^{-1}(\log n)^{1/2})$*. This leads* m *to be in an order of* $O(\sqrt{log(n)})$*.*

Proof: As shown in Theorem 3, the bias of the binned estimator is:

$$\begin{aligned}
\frac{1}{n}\sum Bias^2(\hat{y}_i) &\sim \sum_{l=0}^{m-1}\theta_l^2(\frac{\beta_l\lambda_B/m}{1+\beta_l\lambda_B/m})^2 + \sum_{l=m}^{\infty}{\theta_l}^2 \\
&\le \frac{\lambda_B}{4m}\sum_{l=0}^{m-1}\beta_l\theta_l^2 + \sum_{l=m}^{\infty}\theta_l^2 \\
&\le \frac{\lambda_B}{4m}\sum_{l=0}^{\infty}\beta_l\theta_l^2 \qquad\qquad (\text{when } \frac{\lambda_B\beta_m}{4m} > 1) \\
&\le \frac{\lambda_B}{4m}\,Q,
\end{aligned}$$

and $\lambda_B\beta_m/4m > 1$ is satisfied as $w^2m^2/2 > \log(4m/\lambda_B)$. Then the asymptotic risk is

$$\frac{1}{n}E[(\hat{y}-y)^T(\hat{y}-y)] \le \frac{1}{n}\sum(1+\frac{\beta_l\lambda_B}{m})^{-2} + \frac{\lambda_B}{4m}\,Q \sim 2\sqrt{2}w^{-1}n^{-1}(\log n)^{1/2},$$

when $\log(m/\lambda_B) \sim \log n$ and $\lambda_B/m = o(n^{-1}(\log n)^{1/2})$. □

The theorem shows the binned estimator achieves the same asymptotic rate of the original estimator when m is the order of $O(\sqrt{log(n)})$. Therefore, the computation complexity of the binned estimator is around $O(\log n)^{3/2}$. In practice, we do not expect m can be as small as in this order, since this type of function is not realistic in common applications. Next we study the case of estimating functions in the Sobolev space $H^k(Q)$ with finite order k.

Theorem 5 *The minimax rate of the binned estimator* $\hat{y} = G^{(n,m)}(G^{(m)} + \lambda_B I)^{-1} B^{(m,n)} y$ *for estimating functions in the k-th order Sobolev space* $H^k(Q)$ *is:*

$$\min_{m,w,\lambda_B} \max_{\theta \in H^k(Q)} \frac{1}{n} E[(\hat{y} - y)^T (\hat{y} - y)] \sim (2k+1) k^{-2k/(2k+1)} Q^{1/(2k+1)} n^{-2k/(2k+1)},$$

which is the same rate of the unbinned estimator. This rate is achieved when: $m/n \to 0$ *and* m *is large so that* $m > \sqrt{2} w^{-1} (-\log(\lambda_B/m))^{1/2}$*. The parameter* $\lambda_B = \lambda_B(n, m, w)$ *satisfies* $\log(m/\lambda_B)/w^2 \sim (knQ)^{2/(2k+1)}/2$*. The condition leads* m *to be in an order of* $O(kn^{1/(2k+1)})$*.*

Proof: We first study the bias term, let $\lambda_m = \lambda_B/m$

$$\begin{aligned} B(m, w, \lambda_m) &= \max_{\theta \in H^k(Q)} \sum_{l=0}^{m-1} \theta_l^2 (\frac{\beta_l \lambda_m}{1 + \beta_l \lambda_m})^2 + \sum_{l=m}^{\infty} {\theta_l}^2 \\ &= \max_{\theta \in H^k(Q)} \sum_{l=0}^{m-1} (1 + \beta_l^{-1} \lambda_m^{-1})^{-2} \rho_l^{-1} (\rho_l \theta_l^2) + \sum_{l=m}^{\infty} \rho_l^{-1} (\rho_l \theta_l^2) \end{aligned}$$

Here $\rho_{2l-1} = \rho_{2l} = 1 + l^{2k}$ are the coefficients in the definition (8) of Sobolev ellipsoid $H^k(Q)$. The maximum is achieved by putting all mass Q at the l term that maximizes $\sum_{l=0}^{m-1} (1 + \beta_l^{-1} \lambda_m^{-1})^{-2} \rho_l^{-1} + \sum_{l=m}^{\infty} \rho_l^{-1}$. First let us find the maximizer of

$$A_{\lambda_m}(x) = [1 + \lambda_m^{-1} exp(-x^2 w^2/2)]^{-2} (1 + x^{2k})^{-1} \qquad \text{over} \quad x \geq 0$$

As shown in Lin and Brown (2004), the maximizer x_0 satisfies $x_0^2 w^2/2 \sim (-\log \lambda_m)$ and the maximum $A_{\lambda_m}(x_0) \sim x_0^{-2k} \sim 2^{-k} w^{2k} (-\log \lambda_m)^{-k}$. When $m > x_0$ and $m \geq \sqrt{2} w^{-1} (-\log \lambda_m)^{1/2}$, we have $(1+m^{2k})^{-1} < 2^{-k} w^{2k} (-\log \lambda_m)^{-k}$. Therefore, the maximum value of $B(m, w, \lambda_m) \sim Q 2^{-k} w^{2k} (-\log \lambda_m)^{-k}$. Thus we have the following:

$$\max_{\theta \in H^k(Q)} \frac{1}{n} E[(\hat{y} - y)^T (\hat{y} - y)] \sim Q 2^{-k} w^{2k} (-\log \lambda_m)^{-k} + 2\sqrt{2} w^{-1} n^{-1} (-\log \lambda_m)^{1/2}.$$

This asymptotic rate $(2k+1) k^{-2k/(2k+1)} Q^{1/(2k+1)} n^{-2k/(2k+1)}$ is achieved when the parameters satisfy $\log(m/\lambda_B)/w^2 \sim (knQ)^{2/(2k+1)}/2$ and $m > \sqrt{2} w^{-1} (-\log(\lambda_B/m))^{1/2}$. □

The theorem shows the binned estimator achieves the same minimax rate of the original estimator in the finite order Sobolev space. The same result also holds in the ellipsoid $A_\alpha(Q)$

of analytic functions but we would not prove it here. Comparing the order of smallest m needed to achieve the optimal rates for estimating functions with different order of smoothness, we find that the smoother functions require a smaller number of bins. For instance, the optimal rate of estimating a function in the k-th order Sobolev space can be achieved by binning the data into $m = O(kn^{1/(2k+1)})$ number of bins. The number of bins m decreases as k increases. Binning reduces the computation from $O(n^3)$ to $O(m^3) = O(n)$ for $k = 1$, to $O(n^{3/5})$ for $k = 2$, and even less for larger k values.

5 Experiments

Simulations and real data experiments are conducted in this section to study the effect of binning in regressions and classifications. We first use simulations to study binning in estimating periodic functions in the nonparametric regression setup. The results show that the accuracy of binned estimators are no worse than the original estimators when function are smooth enough. Meanwhile, the computation is reduced to 0.4% of the computation original estimator, when the original 120 data points in binned into 20 bins.

For classification, we test the binning idea on a problem raised in a polar cloud detection problem (cf Shi et al 2004). The $L2$ loss and hinge loss functions are both tested in this experiment. In both cases, the binned classifier provide competitive results to classifiers trained from full data. Furthermore, the computation time is significantly reduced by binning. As an illustration, the time for training SVM on 966 bins is 2.56 minutes, only 0.071% of 5.99 hours that is used to train SVM on 27179 samples, which provide slightly better accuracy than the SVM on 966 bins.

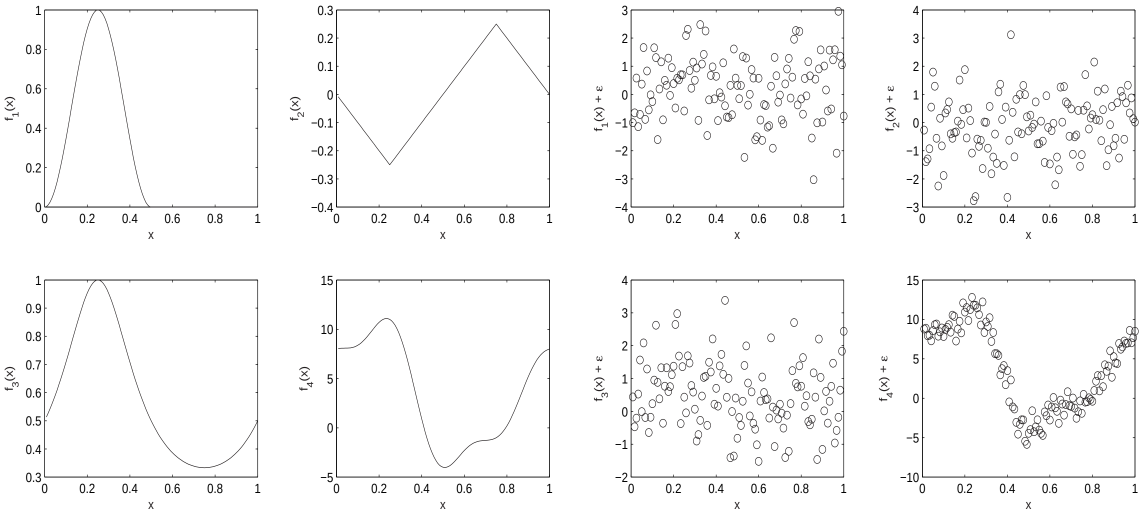

Figure 1: Regression functions and data used in the simulations.

5.1 Non-parametric Regression

Data are simulated from the regression model (1) with noise $N(0,1)$, using four periodic functions on $(0,1]$ with different order of smoothness.

$$f_1(x) = \sin^2(2\pi x)1_{(x\le 1/2)}$$

$$f_2(x) = -x + 2(x-1/4)1_{(x\ge 1/4)} + 2(-x+3/4)1_{(x\ge 3/4)}$$

$$f_3(x) = 1/(2-\sin(2\pi x))$$

$$f_4(x) = 2 + \sin(2\pi x) + 2\cos(2\pi x) + 3\sin^2(2\pi x) + 4\cos^3(2\pi x) + 5\sin^3(2\pi x)$$

The plots of the functions are given the left of Figure 1 and the data are plotted at the right. The first function has a second order of smoothness. The second function has the first order of smoothness. The third function is infinitely smooth. The fourth function is even smoother: it has a Fourier series that only contains finitely many terms. In our simulation, the sample size n is set to 120 and the number of bins are $m = 60, 40, 30, 24, 20, 15, 12$, with corresponding numbers in each bin as $p = 2, 3, 4, 5, 6, 8, 10$. All simulations are done in Matlab 6.

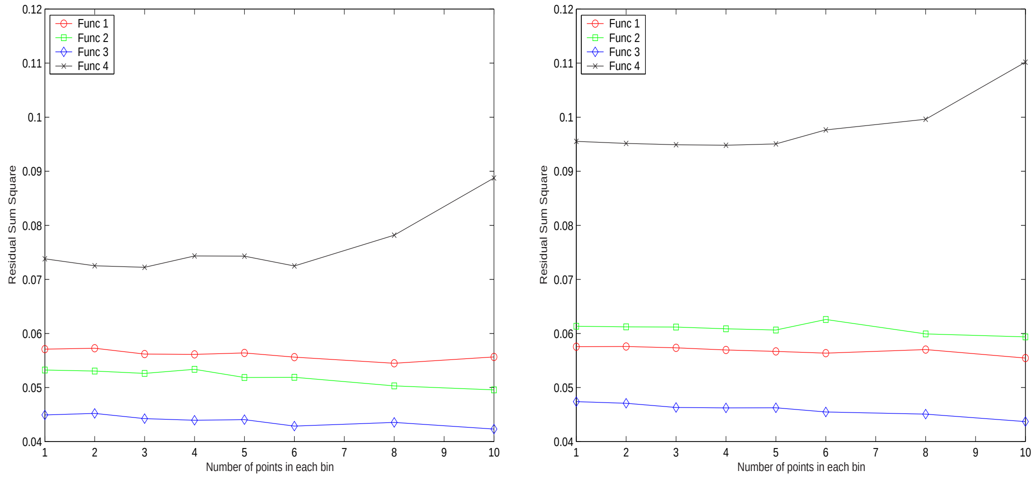

Figure 2: Mean square errors of the binned estimators v.s the number of data points in each bin. Left: Binned Periodic Gaussian kernel regularization; Right: Binned Gaussian kernel regularization. In both plots, unbinned estimators are those with 1 data in each bin.

The computation of the periodic Gaussian regularization is sketched as follows. We follow Lin and Brown (2004) to approximate the periodic Gaussian kernel defined in (6). A Gaussian kernel $G(s,t) = (2\pi)^{-1/2}\omega^{-1}exp((s-t)^2/2\omega^2)$ is used to approximate $K(s,t)$. It is shown in Willamson et al (2001) that $K(s,t) = \sum_{k=-\infty}^{\infty} G((s-t-2k\pi)/2\pi)$. Actually $G^J(s,t) = \sum_{k=-J}^{J} G((s-t-2k\pi)/2\pi)$ for $J=1$ is already a good approximation to $K(s,t)$ with

$$0 < K(s,t) - G^1(s,t) < 2.1 \times 10^{-20}, \qquad \forall (s-t) \in (0,1] \ \text{ for } \ w \leq 1.$$

Therefore, we use $G^1(s,t)$ as an easily computable proxy of $K(s,t)$ in the simulation.

Over the data generated from the regression model (1) on the four functions considered, we compare the mean squared errors of the binned estimator and the original estimator. For periodic Gaussian kernel regularization, we search over $w = 0.3k_1 - 0.1$ for $k_1 = 1, \cdots, 10$; and $\lambda = exp(-0.4k_2 + 7)$, for $k_2 = 1, \cdots, 50$. Then we compute the binned estimator for the number data in each bin p as $2, 3, 4, 5, 6, 8, 10$ separately, The parameters are set to be ω and $\lambda_B = m\lambda/n$. In both cases, we use the minimal point of Mallow's C_p to choose the parameter (w, λ_B).

The simulation runs 300 times. The left panel of Figure 2 shows the averaged mean squared errors against the number of data points in each bin for the four functions (with the unbinned estimators shown as those with one data in each bin in the plot). In most of the cases, the average errors of binned estimators are not significantly higher than those the original estimators, while the computation is reduced from $O(120^3)$ to $O(m^3)$. For example, let us consider the estimator using 6 data points in each bin (m=20). The standard error (not shown in the plot) of the average errors are computed and two sample t tests are conducted to compare the binned estimator to the original estimator. For all four functions, the p-values are all larger than 0.1, which says no significant loss of accuracy by binning the data to 20 bins in this experiment. In the mean time, the computation complexity is reduced to $O(20^3)$, 0.4% of $O(120^3)$ on full data.

In our experiment, the periodic Gaussian kernel is replaced by a Gaussian kernel, which is most common in practice. We repeat the same experiments again and get the average mean square errors plotted in the right panel of Figure 2. The errors from using the Gaussian kernel are generally higher than those from the periodic Gaussian kernel, since the Gaussian kernel does not take in account of the fact that our functions are periodic. However, the binned estimators have almost the same accuracy as the unbinned ones when there are enough number (say 24) of bins in this simulation. The computation reduction of Gaussian kernel is the same as in the periodic Gaussian case.

5.2 Cloud Detection over Snow and Ice Covered Surface

In this section, we test binning in a real classification problem using Gaussian kernel regularization. By reducing the variance, binning the data is expected to keep the classification accuracy as well as relieving the computation burden even in classification. Here we illustrate the effect of binning using a polar cloud detection problem arising in atmospheric science. In polar regions, detecting clouds using satellite remote sensing data is difficult, because the surface is covered by snow and ice that have similar reflecting signatures as clouds. In Shi

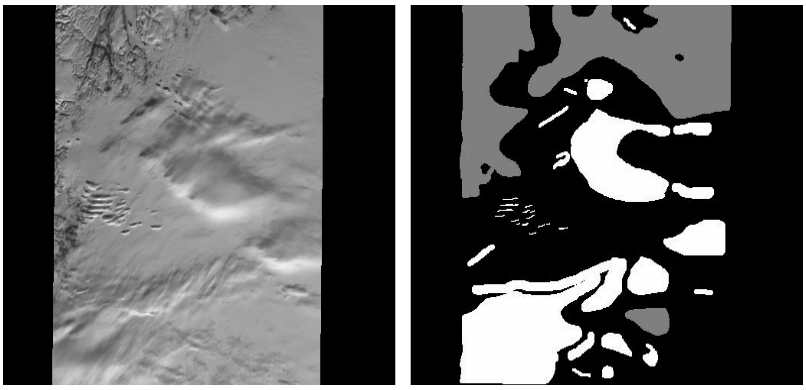

Figure 3: MISR image and expert labels

et al (2004), the Enhanced Linear Correlation Matching Classification (ELCMC) algorithm based on three features was developed for polar cloud detection using data collected by the Multi-angle Imaging SpectroRadiometer (MISR).

Thresholding the features, the ELCMC algorithm has an average accuracy about 91% (compare to expert labels) over 60 different scenes, with around 55,000 valid pixels in each scene. However, there are some scenes that are very hard to classify using the the simple threshold method. The data set we investigate in this paper is collected in MISR orbit 18528 blocks 22-24 over Greenland in 2002, with only a 75% accuracy rate by the ELCMC method. The MISR red channel image of the data is shown in the left panel of Figure 3. It is not easy to separate clouds from the surface because the scene itself is very complicated. There are several types of clouds in this scene, such as low clouds, high clouds, transparent high clouds above low clouds. Moreover, this scene also contains different types of surfaces, such as smooth snow covered terrains, rough terrains, frozen rivers, and cliffs.

Right now, the most reliable way to get large volume of validation data for polar cloud detection is by expert labelling, since there are not enough ground measurements in polar region. The expert labels from our collaborator Prof. Eugene Clothiaux (Department of Meteorology, PSU) are shown at the right panel with white pixels denoting "cloudy", gray pixels for "clear" and black for "not sure". There are 54879 pixels with "cloudy" or "clear" labels in this scene and we use half of these labels for training and half for testing different classifiers. Each pixels is associated with a three dimensional vector $X = (log(SD), CORR, NDAI)$, computed from the original MISR data as described in Shi et al (2004). Hence we build and test classifiers based on these three features.

We test binning on the Gaussian kernel regularization with two different type of loss functions. One is the $L2$ loss function as we studied in this paper, and the other is the hinge loss function corresponding to the Support Vector Machines. In both case, we binned the data based on the empirical marginal distribution of the three predictors. For each predictor, we found the 10%, 20%, $\cdots$, 90% percentiles of the empirical distribution and these percentiles serve are the split points for each predictor. Therefore, we get 1000 bins in the three dimensional space. In those bins, 966 of them contain data and 34 are empty. Thus, the 966 bin centers are our binned data in the experiments. The computation is carried out in Matlab 6 on a desktop computer with a Pentium 4 2.4GHz CPU and 512M memory.

5.2.1 Binning on Gaussian Kernel Regularization with $L2$ loss

The Gaussian kernel regularization with the $L2$ loss function is tested with three different setups for training data. The first setup is random sampling a small proportion of data as training data and train classification over them. This is the common approach to deal with large data sets and it serves as a baseline for our comparison. In the second setup, the bin centers and the majority vote of the labels in each bin are used as training data and responses. Thus, each bin center is treated as one data point in this case. In the last setup, the training data and labels are the bin centers and the proportion of 1's in each bin. To reflect the fact that different bins may have different number of data points, we also give a weight to each bin center in the loss function in the last setup. In all there setups, a half of the 54879 data points are left out for choosing the best parameters ω and λ.

In the first setup, we randomly sample 966 data points from the full data (54879 data points) and use the corresponding label y (0 and 1) to train the classifier $\hat{y} = K_\omega(K_\omega + \lambda I)^{-1} y$.

	random sample size 966		GKR-L2	GKR-L2 on 966 bin
	GKR-L2	Bagged GKR-L2	996 bin centers	centers with fuzzy labels
Accuracy	71.40% *	77.77%	75.86%	79.22%
Comp Time (seconds)	81×26.24 = 35.42 minutes	$81 \times 21 \times 26.24$ = 12.40 hours	$3.87+ 81 \times 26.24$ = 35.48 minutes	$3.87 + 81 \times 26.24$ = 35.48 minutes

Table 1: Binning $L2$ Gaussian kernel regularization for cloud detection. Note: * The average accuracy of 21 runs.

The predicted labels are set to be the indicator function $I(y > 0.5)$. Cross-validation is performed to chose the parameters (ω, λ) from $\omega = 0.8 + (i-5) \times 0.05$ and $\lambda = .1 + (j-5) \times 0.005$ for $i, j = 1, \cdots, 9$. For each (ω, λ) pair, this procedure is repeated for 21 times and the average classification rate is reported. The best average classification rate is 71.40% (with SE 0.43%). With the classification results from the 21 runs, we also take the majority vote over the results to build a "bagged" classifier, which improve the accuracy to 77.77%. As discussed in Breiman (1996) and Bühlmann and Yu (2002), Bagging reduces the classification error by reducing the variance.

In the second setup, the 966 bin centers are used as training data. Cross-validation is carried out to find the best parameters (ω, λ) over the same range as in the first setup. The classifier is then applied to the full data to get an accuracy rate. The best set of parameters lead to 75.86% of accuracy rate.

In the third setup, we solve the following minimization problem: $min_c \sum_{i=1}^{996} (y - Kc)^T W (y - Kc) + \lambda c^T Kc$, with the weight in the diagonal matrix W being proportional to the number of data points in each bin. It leads to the solution are $c = (K + \lambda W^{-1})^{-1} y$. Doing cross-validation over the same range of parameter, we achieve a 79.22% of accuracy, which is the best results with sample size 966 in $L2$-loss.

We compare the computation time (in Matlab) of those setups in Table 1 as well. Training and testing the $L2$ Gaussian kernel regularization on 966 data points takes 26.24 seconds on average. Using cross-validation to chose the best parameters, it takes about 35.42 minutes (26.24 $\times$ the number of parameter pairs tested) for the simple classifier in the first setup, and the "bagged" classifier takes 12.40 hours. In the second and third setups, binning the data in 966 bins takes 3.87 seconds and the training process takes 35.42 minutes. So the computation of binning classifiers takes only 4.77% (35.48min/12.40hr) of the time for training the "bagging" classifier, but it provides better estimation results. It is worthwhile to point out that the training step of these classifiers involves inverting an n by n matrix, the computer runs out of memory when the training data size is larger than 3000.

5.2.2 Binning on Gaussian Kernel SVM

Gaussian kernel Support Vector Machines is a regularization method using the hinge loss function in expression (2). Because of the hinge loss function, a large proportion of the parameter $c_1, \cdots, c_n$ are zeros, and the non-zeros data points are called support vectors (see Vapnik 1995 and Whaba et al 1999 for details). In this section, we study the effect of binning on the Gaussian kernel SVM for the polar cloud detection problem, even though our theoretical results only cover the $L2$ loss.

The software that we used to train the SVM is the Ohio State University SVM Classifier Matlab Toolbox (Junshui Ma et al. `http://www.eleceng.ohio-state.edu/∼maj/osu_smv/`). The OSU SVM toolbox implements SVM classifiers in C++ using the LIBSVM algorithm of Chih-Chung Chang and Chih-Jen Lin (`http://www.csie.ntu.edu.tw/`*sim*`cjlin/libsvm/`). The LIBSVM algorithm breaks the large SVM Quadratic Programming (QP) optimization problem into a series of small QP problems to allow the training data size to be very large. The computational complexity of training LIBSVM is empirically around $O(n_1^2)$, where n_1 is the training sample size. The complexity of testing is $O(s\,n_2)$ where n_2 is the test size and s is the number of support vectors, which usually increases linearly with the size of the training data set.

Similar to the L2 Gaussian kernel regularization in section 5.2.1, the Gaussian kernel

	random sample size 966		SVM on 966	SVM
	SVM	Bagged SVM	bin centers	size 27179
Accuracy	*85.09%	86.07%	86.08%	86.46%
Comp Time (seconds)	81×1.85 $= 2.5$ minutes	$21 \times 81 \times 1.85$ $= 52.11$ minutes	$3.87 + 81 \times 1.85$ $= 2.56$ minutes	81×266.06 $= 5.99$ hours
# Support Vectors	350	~ 7350	210	8630

Table 2: Binning SVM for cloud detection. Note: * average rate of 21 runs with SE 0.18%

SVM is tested in three different types of training data. The first two setups are identical to the ones used in the $L2$ loss case. However, the third setup in the $L2$ case is not easy to carry out in OSU SVM, since the OSU SVM training package does not take the fuzzy labels or support adding weights to each individual points. Hence we replace the third setup by randomly sampling half of the data (27179 points) and compare the accuracy of SVM trained from this huge sample to the ones from the first two types of training data. For all tested classifiers, The accuracy, computation time and number of support vectors are given in table 2.

The first observation from the table is that the SVM with all the data (27179 points) provides the best test classification rate, but requires the longest computation time. The accuracy rates of the bagging SVM and the SVM on bin centers are comparable, but the bagging SVM needs 20 times more computation time. The time for training SVM on 966 bin centers is 2.56 minutes, only 0.71% of 5.99 hours that is used to train SVM on 27179 samples. With the same amount of computation, the accuracy of SVM on bin centers (86.08%) is significantly higher (5 SE above the average) than the average accuracy (85.09%) of the same sized SVM on random samples. Therefore, SVM on bin centers are better than SVM on the same sized data randomly sampled from full data. Thus, SVM on the bin centers is the computationally most efficient method for training SVM and it provides almost the same accuracy to the full size SVM.

Besides the training time, the number of support vectors determines the computation time need to classify new data. As shown in the table, SVM on bin centers has the fewest number of support vectors, so it is the fastest to classify a new point. From the comparison, it is clear that SVM on the binning data provides almost the best accuracy, fast training, and fast prediction.

At last, we compare binning with another possible sample-size reduction scheme, clustering. Feng and Mangasarian (2001) has proposed to use the k-mean clustering algorithm to pick a small proportion of training data for SVM. This method first cluster the data into m clusters. Although this method reduces the size of training data as well, the computation of k-means clustering itself is very expensive comparing to that of training SVM, or even not feasible due to the memory usage when data size is too large. In the cloud detection problem, clustering 27179 training data into 512 groups takes 21.65 minutes, and the time increases dramatically when the number of centroid increases. The increase in the requirement of computer memory is even worse than the increase in the computation time. The computer memory runs out when we try to cluster the data into 966 clusters.

Just for comparison, clustering-SVM and binning-SVM on 512 groups provides very close classification rates, 85.72% and 85.64% respectively, but binning is much faster than clustering. Running in Matlab, the clustering itself takes 21.65 minutes, which is 376 times of the computation time (3.45 seconds) of binning data to 512 bins. The number of support vectors of clustering-SVM and binning-SVM are very close, 145 and 143 respectively, so their testing times are about the same. Thus binning is more preferred in reducing the computation for SVM than clustering.

6 Summaries

To reduce the computational burden of the Gaussian kernel regularization methods, we propose binning on training data. The binning effect on the periodic Gaussian kernel regularization method is studied in the nonparametric regression. While reducing the computation complexity from $O(n^3)$ to the order of $O(n)$ or less, the binned estimator keeps the asymptotic minimax rates of the periodic Gaussian regularization in the Sobolev spaces.

Simulations in the finite sample regression case suggests that the performance of the binned periodic Gaussian kernel regularization estimator is comparable to the original estimator in terms of the estimation error. In our simulation of binning 120 data points in 20 bins, computing the binned estimator only takes 0.4% of the computation time of the unbinned estimator, but the binned estimators provide almost the same accuracy. Binning the Gaussian kernel regularization also gives error rates close to those using the full data in our simulation.

In the polar cloud detection problem, we tested the binning method on the $L2$ loss Gaussian kernel regularization and the Gaussian kernel SVM. With the same computation time, the $L2$-loss Gaussian kernel regularization on 966 bins achieves better accuracy (79.22%) than that (71.40%) on 966 randomly sampled data. For SVM, binning reduces the computation time (from 5.99 hours to 2.56 minutes in our example), keeps the classification accuracy, and speeds up the testing step by providing simpler classifiers with fewer support vectors (from 8630 to 210). The SVM trained on 966 randomly selected samples has a similar training time as and a slightly worse test classification rate than the SVM on 966 bins, but has 67% more support vectors so takes 67% longer to predict on a new data point. The SVM trained on 512 cluster centers from the k-mean algorithm reports almost the same test classification rate and a similar number of support vectors as the SVM on 512 bins, but the k-mean clustering itself takes 375 times more computation time than binning.

In summary, binning can be used as an effective method for dealing with large number of training data in Gaussian kernel regularization methods. Binning is also more preferred in reducing the computation for SVM than clustering.

Acknowledgements

Tao Shi is partially supported by NSF grant CCR-0106656. Bin Yu is partially supported by NSF grant CCR-0106656, NSF grant DMS-0306508, ARO grant DAAD 19-01-01-0643, and the Miller Institute as a Miller Research Professor in Spring 2004. MISR data were obtained at the courtesy of the NASA Langley Research Center Atmospheric Sciences Data Center. We'd like to thank Prof. Eugene Clothiaux for providing hand labels of MISR data and Mr. David Purdy for helpful comments on the presentation of the paper.

Appendix

Proof of Theorem 1:

As shown in section 3, the expansion of kernel $K(.,.)$ in equation (6) leads to

$$G_{i,j}^{(n)} = K(x_i, x_j) = 2\sum_{l=0}^{\infty} e^{-l^2\omega^2/2}[\sin(2\pi l x_i)\sin(2\pi l x_j) + \cos(2\pi l x_i)\cos(2\pi l x_j)] \tag{19}$$

with $x_i = \frac{i}{n} - \frac{1}{2n}$.

In case n is an odd number ($n = 2q+1$), any non-negative integer l can be written as $l = kn - h$ or $l = kn + h$, where both k and h are integers satisfying $k \geq 0$ and $0 \leq h \leq q$. For any $k \geq 1$, $h > 0$, and all i,

$$\begin{aligned}
\sin(2\pi(kn+h)x_i) &= \sin(2\pi knx_i + 2\pi hx_i) \\
&= \sin(2\pi knx_i)\cos(2\pi hx_i) + \cos(2\pi knx_i)\sin(2\pi hx_i) \\
&= \sin(2ki\pi - k\pi)\cos(2\pi hx_i) + \cos(2ki\pi - k\pi)\sin(2\pi hx_i) \\
&= (-1)^k \sin(2\pi hx_i)
\end{aligned}$$

In the same way, we get $\sin(2\pi(kn-h)x_i) = (-1)^{k+1}\sin(2\pi hx_i)$, and $\cos(2\pi(kn+h)x_i) = \cos(2\pi(kn-h)x_i) = (-1)^k\cos(2\pi hx_i)$. In case $h = 0$, we have $\sin(2\pi knx_i) = 0$ and $\cos(2\pi knx_i) = (-1)^k$ for all i. Therefore, the Gram matrix $G^{(n)}$ can be written as:

$$G_{i,j}^{(n)} = d_0^C + \sum_{h=1}^{q}[d_h^S \sin(2\pi hx_i)\sin(2\pi hx_j) + d_h^C \cos(2\pi hx_i)\cos(2\pi hx_j)] \tag{20}$$

where

$$d_0^C = 2\sum_{k=0}^{\infty}(-1)^k e^{-(kn)^2\omega^2/2},$$

$$d_h^S = 2\{e^{-h^2\omega^2/2} + \sum_{k=1}^{\infty}(-1)^k(e^{-(kn+h)^2\omega^2/2} - e^{-(kn-h)^2\omega^2/2})\},$$

$$d_h^C = 2\{e^{-h^2\omega^2/2} + \sum_{k=1}^{\infty}(-1)^k(e^{-(kn+h)^2\omega^2/2} + e^{-(kn-h)^2\omega^2/2})\}.$$

Let $V_0^{(n)} = \sqrt{\frac{1}{n}}(1, \cdots, 1)^T$, $V_{2h-1}^{(n)} = \sqrt{\frac{2}{n}}(\sin(2\pi h x_1), \cdots, \sin(2\pi h x_n))^T$, and $V_{2h}^{(n)} = \sqrt{\frac{2}{n}}(\cos(2\pi h x_1), \cdots, \cos(2\pi h x_n))^T$, for $h = 1, \cdots, q$. Using the following orthogonal relationships

$$\begin{aligned}
\sum_{i=1}^{n} \sin(2\pi\mu x_i)\sin(2\pi\nu x_i) &= n/2 \quad \mu = \nu = 1, \cdots, q \\
&= 0 \quad \mu \neq \nu; \mu, \nu = 0, \cdots, q \\
\sum_{i=1}^{n} \cos(2\pi\mu x_i)\cos(2\pi\nu x_i) &= n/2 \quad \mu = \nu = 1, \cdots, q \\
&= 0 \quad \mu \neq \nu; \mu, \nu = 0, \cdots, q \\
\sum_{i=1}^{n} \cos(2\pi\mu x_i) &= 0 \quad \mu = 1, \cdots, q \\
\sum_{i=1}^{n} \sin(2\pi\mu x_i) &= 0 \quad \mu = 1, \cdots, q
\end{aligned}$$

we can easily see that $V_0, \cdots, V_{2q}$ are orthonormal vectors. Furthermore, they are the eigen-vectors of $G^{(n)}$ with corresponding eigen-values $d_0^{(n)} = nd_0^C$, $d_{2h-1}^{(n)} = nd_h^S/2$, and $d_{2h}^{(n)} = nd_h^C/2$, since $G^{(n)} = \sum_{l=0}^{2q} d_l^{(n)} V_l^{(n)} V_l^{(n)T}$. It completes the proof for odd number $n = 2q + 1$.

For even number $n = 2q$ observations , the eigen-vectors of $G^{(n)}$ are $V_0^{(n)}, \cdots, V_{2q-1}^{(n)}$ and eigen-values are $d_0^{(n)}, \cdots, d_{2q-1}^{(n)}$, while both are the same as defined in the odd number case. All the proofs for odd numbers n hold here except $\sin(2\pi kq x_i) = \sin(2\pi kq(i/n - 2/n)) = 0$ for all $k > 0$, which leaves $V_0, \cdots, V_{2q-2}, V_{2q}$ as the $2q$ eigen-vectors. The eigen-vectors are slightly different of those for odd number n, but this difference does not affect the asymptotic results at all. Therefore, we will use the eigen-structure for odd number observations in the rest of the paper.

To simple the notation, we can write them eigen-values d_l in terms of ρ_l:

$$\begin{aligned}
d_0^{(n)} &= 2n \sum_{k=0}^{\infty} (-1)^k \rho_{2kn} \\
d_l^{(n)} &= n\{\rho_l + \sum_{k=1}^{\infty} (-1)^k [\rho_{kn+h} + (-1)^{l-2h} \rho_{kn-h}]\}
\end{aligned}$$

where $l = 1, \cdots, n - 1$, and $h = \lceil (l + 1)/2 \rceil$, while $\lceil a \rceil$ stands for the integer part of a. $\square$

Proof of Proposition 1:

For $x \in (0,1]$, $\bar{x}$ as defined in proposition 1, and $k \geq 0$

$$
\begin{aligned}
& \sum_{j=1}^{m} K(x,\bar{x}_j) cos(2\pi k \bar{x}_j) \\
= & \sum_{j=1}^{m} \{2 \sum_{l=0}^{\infty} \exp(-l^2\omega^2/2) cos(2\pi l(x-\bar{x}_j)) cos(2\pi k \bar{x}_j)\} \\
= & \ 2 \sum_{l=0}^{\infty} \exp(-l^2\omega^2/2) \{\sum_{j=1}^{m} cos(2\pi l(x-\bar{x}_j)) cos(2\pi k \bar{x}_j)\} \\
= & \sum_{l=0}^{\infty} \exp(-l^2\omega^2/2) \{\sum_{j=1}^{m} cos(2\pi(lx+(k-l)\bar{x}_j)) + cos(2\pi(lx-(k+l)\bar{x}_j))\}
\end{aligned}
$$

For any integer r,

$$
\begin{aligned}
\sum_{j=1}^{m} cos(2\pi(lx + r\bar{x}_j)) &= \sum_{j=1}^{m} cos(2\pi lx - \frac{r}{m}\pi + 2\pi \frac{r}{m} j) \\
&= \begin{cases} 0 & \text{when } \frac{r}{m} \text{ is not an integer;} \\ m(-1)^{r/m} cos(2\pi lx) & \text{when } \frac{r}{m} \text{ is an integer.} \end{cases}
\end{aligned}
$$

Therefore,

$$\sum_{j=1}^{m} K(x,\bar{x}_j) cos(2\pi k \bar{x}_j) = d_k^{(m)} cos(2\pi kx)$$

while $d_k^{(m)}$ follows the definition in Proposition **??**. It is also true that $\sum_{j=1}^{m} K(x,\bar{x}_j) sin(2\pi k \bar{x}_j) = d_k^{(m)} sin(2\pi kx)$. As shown in Proposition **??**, the eigen-vector $V_k^{(m)}$ of $G^{(m)}$ is $\sqrt{2/m} cos(2\pi \bar{x}_j)$ or $\sqrt{2/m} sin(2\pi \bar{x}_j)$. Therefore,

$$G^{(n,m)} V_k^{(m)} = d_k^{(m)} \sqrt{\frac{n}{m}} V_k^{(n)}$$

for all $k = 0, 1, \cdots, m$. □

Proof of Theorem 2:

Following the relationship shown in proposition 1 $G^{(n,m)} V^{(m)} = \sqrt{\frac{n}{m}} V^{(n,m)} diag(d_l^{(m)})$, with $V^{(n,m)}$ as the n by m matrix formed by the first m eigen-vectors of $G^{(m)}$. The projection matrix $S_B = G^{(n,m)} V^{(m)} diag(\frac{1}{d_l^{(m)} + \lambda_B}) V^{(m)^T} B^{(m,n)} = \sqrt{\frac{n}{m}} V^{(n,m)} diag(\frac{d_l^{(m)}}{d_l^{(m)} + \lambda_B}) V^{(m)^T} B^{(m,n)}$.

Since $B^{(m,n)}B^{(m,n)^T} = diag(m/n)$, the asymptotic variance of the estimator is:

$$\begin{aligned}
& \frac{1}{n}\sum var(\hat{y}_i) = \frac{1}{n}\ trace(S_B^T S_B) \\
= & \frac{1}{n}\ trace(\frac{n}{m}V^{(n,m)}diag(\frac{d_l^{(m)}}{d_l^{(m)}+\lambda_B})V^{(m)^T}B^{(m,n)}B^{(m,n)^T}V^{(m)}diag(\frac{d_l^m}{d_l^{(m)}+\lambda_B})V^{(n,m)^T}) \\
= & \frac{1}{n}\ trace(diag(\frac{d_l^{(m)}}{d_l^{(m)}+\lambda_B})^2) = \frac{1}{n}\sum_{l=0}^{m-1}(\frac{d_l^{(m)}}{d_l^{(m)}+\lambda_B})^2
\end{aligned}$$

As proved before, $lim_{m\to\infty}d_l^{(m)}/m = \rho_l$ for $l > 0$ and $\rho_l = 1/\beta_l$, we get

$$\frac{1}{n}\sum var(\hat{y}_i) \sim \frac{1}{n}\sum(\frac{\rho_l}{\rho_l + (\lambda_B/m)})^2 = \frac{1}{n}\sum(1+\frac{\beta_l\lambda_B}{m})^{-2}$$

Proof of Theorem 3:

The bias of the binned estimator is $\frac{1}{n}\sum Bias^2(\hat{y}_i) = \frac{1}{n}((S_B-I)F)^T((S_B-I)F)$. Let $C^{(n,m)}$ denote a n by m matrix of $(I_{m\times m} : 0_{m\times(n-m)})^T$. The term $(S_B - I)F$ is expanded as:

$$\begin{aligned}
(S_B - I)F &= (G^{(n,m)}(G^{(m)}+\lambda_B I)^{-1}B^{(m,n)} - I)V^{(n)}\Theta^{(n)} \\
&= \sqrt{\frac{n}{m}}V^{(n,m)}diag(\frac{d_l^{(m)}}{d_l^{(m)}+\lambda_B})V^{(m)^T}B^{(m,n)}V^{(n)}\Theta^{(n)} - V^{(n)}\Theta^{(n)} \\
&= V^{(n)}C^{(n,m)}diag(\sqrt{\frac{n}{m}}\frac{d_l^{(m)}}{d_l^{(m)}+\lambda_B})V^{(m)^T}B^{(m,n)}V^{(n)}\Theta^{(n)} - V^{(n)}\Theta^{(n)} \\
&= V^{(n)}(C^{(n,m)}diag(\sqrt{\frac{n}{m}}\frac{d_l^{(m)}}{d_l^{(m)}+\lambda_B})V^{(m)^T}B^{(m,n)}V^{(n)} - I^{(n)})\Theta^{(n)} \\
&\triangleq V^{(n)}A^{(n,n)}\Theta^{(n)}
\end{aligned}$$

Now, let us study $V^{(m)^T}B^{(m,n)}V^{(n)}$. We first start with one of the $V^{(n)}$'s eigen-vectors: $\sqrt{2/n}(cos2\pi kx_1,\cdots,cos2\pi kx_n)^T$.

$$\begin{aligned}
& B^{(m,n)}(cos2\pi kx_1,\cdots,cos2\pi kx_n)^T \\
= & ((cos2\pi kx_1+\cdots+cos2\pi kx_p)/p,\cdots,(cos2\pi kx_{n-p+1}+\cdots+cos2\pi kx_n)/p)^T \\
= & w_k^{(m,n)}(cos2\pi k\bar{x}_1,\cdots,cos2\pi k\bar{x}_m)^T,
\end{aligned}$$

while $w_k^{(m,n)}$ is a constant as function of n, m, and k. When $p = n/m$ is an odd number, $(x_{rp+1},\cdots,x_{rp+p})$ is expressed as $(\bar{x}_r-(p-1)/2n,\cdots,\bar{x}_r,\cdots,\bar{x}_r+(p-1)/2n)$. Thus,

$cos2\pi kx_{rp+1} + \cdots + cos2\pi kx_{rp+p} = [1 + 2cos\frac{2\pi k}{n} + \cdots + 2cos\frac{2\pi k((p-1)/2)}{n}]cos2\pi k\bar{x}_r$. Therefore, $w_k^{(m,n)} = (1 + \sum_{j=1}^{(p-1)/2} 2cos\frac{2\pi kj}{n})/p$ for odd number p. It is straightforward to show that $w_k^{(m,n)} = (\sum_{j=1}^{p/2} 2cos\frac{2\pi kj-\pi k}{n})/p$ for even number p. In the same way, we have

$$B^{(m,n)}(sin2\pi kx_1, \cdots, sin2\pi kx_n)^T = w_k^{(m,n)}(sin2\pi k\bar{x}_1, \cdots, sin2\pi k\bar{x}_m)^T$$

Let $j^0 = \lceil (j+1)/2 \rceil$ for $0 \le j \le n-1$. Following the proof of proposition 1 and assuming $m = 2q+1$ as an odd number, we can write any j^0 as $j^0 = hm - i^0$ or $j^0 = hm + i^0$ with $0 \le i^0 \le q$, where i^0 is a function of j and m. For the situation of odd number j and $j^0 = hm + i^0$, we have

$$\begin{aligned} B^{(m,n)}V_j^{(n)} &= B^{(m,n)}\sqrt{2/n}(sin2\pi j^0 x_1, \cdots, sin2\pi j^0 x_n)^T \\ &= w_{j^0}^{(m,n)}\sqrt{2/n}(sin2\pi j^0\bar{x}_1, \cdots, sin2\pi j^0\bar{x}_n)^T \\ &= w_{j^0}^{(m,n)}\sqrt{2/n}(-1)^h(sin2\pi i^0\bar{x}_1, \cdots, sin2\pi i^0\bar{x}_n)^T \\ &= w_{j^0}^{(m,n)}\sqrt{m/n}(-1)^h V_{2i^0-1}^{(m)}. \end{aligned}$$

Similarly, we can derived the equation for even number j and $j^0 = hm - i^0$. So the structure of ${V_i^{(m)}}^T B^{(m,n)} V_j^{(n)}$ is:

$${V_i^{(m)}}^T B^{(m,n)} V_j^{(n)} = \sqrt{m/n}\; w_{j^0}^{(m,n)} c_{i,j}^{m,n}\; {V_i^{(m)}}^T V_{2i^0+((-1)^j-1)/2}^{(m)}$$

where the constant $c_{i,j}^{m,n}$ equals $(-1)^h$ when (1) j is even or (2) j is odd and $j^0 = hm + i^0$, and it equals $(-1)^{h+1}$ otherwise. Therefore, the matrix is nonzero only when $i = 2i^0 + ((-1)^j - 1)/2 \triangleq \hat{j}$. Let us denote $\mu_{ij} = w_{j_0}^{(m,n)} c_{i,j}^{m,n}$ for the nonzero elements of matrix ${V^{(m)}}^T B^{(m,n)} V^{(n)}$, which is in the following shape:

$$\sqrt{\frac{m}{n}}\begin{pmatrix} \mu_{0,0} & 0 & \cdots & 0 & 0 & 0 & \cdots & \mu_{0,2m-1} & 0 & 0 & \cdots \\ 0 & \mu_{1,1} & 0 & 0 & 0 & 0 & \cdots & 0 & \mu_{1,2m} & 0 & \cdots \\ 0 & 0 & \cdots & 0 & \mu_{m-2,m} & 0 & \cdots & 0 & 0 & \cdots & \cdots \\ 0 & 0 & \cdots & \mu_{m-1,m-1} & 0 & \mu_{m-1,m+1} & 0 & \cdots & 0 & 0 & \cdots \end{pmatrix}_{m\times n}$$

Since $A^{(n,n)} = C^{(n,m)}diag(\sqrt{\frac{n}{m}}\frac{d_l^{(m)}}{d_l^{(m)}+\lambda_B}){V^{(m)}}^T B^{(m,n)} V^{(n)} - I^{(n)}$, the entry of of $A^{(n,n)}$ is $a_{ij} =$

$\frac{d_i^{(m)}}{d_i^{(m)}+\lambda_B}\mu_{ij} - I(j=i)$ for $0 \le i < m$, and $a_{ij} = -I(j=i)$ for all $i \ge m$. So $A^{(n,n)}$ is

$$\begin{pmatrix} diag(\frac{d_l^{(m)}\mu_{ll}}{d_l^{(m)}+\lambda_B}) - I^{(m,m)} & A^{(m,n-m)} \\ 0 & -I^{(n-m,n-m)} \end{pmatrix}_{n\times n}$$

Now let us study of the bias as a whole term.

$$\begin{aligned}
\frac{1}{n}\sum Bias^2(\hat{y}_i) &= \frac{1}{n}\ ((S_B - I)F)^T((S_B - I)F) \\
&= \frac{1}{n}\ (V^{(n)}A^{(n,n)}\Theta^{(n)})^T V^{(n)}A^{(n,n)}\Theta^{(n)} \\
&= \frac{1}{n}\ (\frac{\Theta^{(n)}}{\sqrt{n}})^T A^{(n,n)T} A^{(n,n)}\frac{\Theta^{(n)}}{\sqrt{n}} \\
&= \sum_{j=0}^{m-1}(\frac{\Theta_j^{(n)}}{\sqrt{n}})^2(\frac{d_j^{(m)}\mu_{jj}}{d_j^{(m)}+\lambda_B} - 1)^2 + \sum_{j=m}^{n-1}(\frac{\Theta_j^{(n)}}{\sqrt{n}})^2(1 + (\frac{d_{\hat{j}}^{(m)}\mu_{\hat{j}j}}{d_{\hat{j}}^{(m)}+\lambda_B})^2) \\
&\quad + \sum_{k=0}^{m-1}\sum_{j=m}^{n-1}\frac{\Theta_k^{(n)}\Theta_j^{(n)}}{n}(\frac{d_k^{(m)}\mu_{kk}}{d_k^{(m)}+\lambda_B} - 1)(\frac{d_k^{(m)}\mu_{kj}}{d_k^{(m)}+\lambda_B})I(k=\hat{j}) \\
&\quad + \sum_{k=m}^{n-1}\sum_{j=0}^{m-1}\frac{\Theta_k^{(n)}\Theta_j^{(n)}}{n}(\frac{d_k^{(m)}\mu_{kj}}{d_k^{(m)}+\lambda_B})(\frac{d_j^{(m)}\mu_{jj}}{d_j^{(m)}+\lambda_B} - 1)I(j=\hat{k}) \\
&\quad + \sum_{k=m}^{n-1}\sum_{j=m}^{n-1}\frac{\Theta_k^{(n)}\Theta_j^{(n)}}{n}(\frac{d_k^{(m)}\mu_{kj}}{d_k^{(m)}+\lambda_B})(\frac{d_k^{(m)}\mu_{kj}}{d_k^{(m)}+\lambda_B})I(\hat{j}=\hat{k})I(j\neq k) \\
&\sim \sum_{j=0}^{m-1}(\frac{\Theta_j^{(n)}}{\sqrt{n}})^2(\frac{d_j^{(m)}\mu_{jj}}{d_j^{(m)}+\lambda_B} - 1)^2 + \sum_{j=m}^{n-1}(\frac{\Theta_j^{(n)}}{\sqrt{n}})^2(1 + (\frac{d_{\hat{j}}^{(m)}\mu_{\hat{j}j}}{d_{\hat{j}}^{(m)}+\lambda_B})^2)
\end{aligned}$$

when $n \to \infty$, $m \to \infty$ and $d^m/(d^m+\lambda) \to 0$. Since $c_{j,j}^{m,n} = 1$ for $j = 1, \cdots, m$ and $w_j^{(m,n)} \to 1$ as $m/n \to 0$, we have $\mu_{jj} \to 1$. Therefore,

$$\begin{aligned}
\frac{1}{n}\sum Bias^2(\hat{y}_i) &\sim \sum_{j=0}^{m-1}\theta_j^2(\frac{\rho_j}{\rho_j + \lambda_B/m} - 1)^2 + \sum_{j=m}^{\infty}{\theta_j}^2 \\
&= \sum_{j=0}^{m-1}\theta_j^2(\frac{\lambda_B/m}{\rho_j + \lambda_B/m})^2 + \sum_{j=m}^{\infty}{\theta_j}^2 \\
&= \sum_{j=0}^{m-1}\theta_j^2(\frac{\beta_j\lambda_B/m}{1 + \beta_j\lambda_B/m})^2 + \sum_{j=m}^{\infty}{\theta_j}^2
\end{aligned}$$

References

[1] Breiman, L. (1996a). Bagging Predictors. *Machine Learning*, **24** No. 2 123-140.

[2] Brown, L.D. and Low, M.G. (1996). Asymptotic Equivalence of Nonparametric Regression and White Noise. *Annals of Statistics* **24** 2384-2398.

[3] Bühlmann, P. and Yu, B. (2002). Analyzing Bagging. *Annals of Statistics* **30** 927-961.

[4] Evgeniou, T., Pontil, M., and Poggio, T. (2000). Regularization Networks and Support Vector Machines. *Advances in Computational Mathematics* **13** 1-50.

[5] Feng, G. and Mangasarian, O. L. (2001) Semi-Supervised Support Vector Machines for Unlabeled Data Classification. *Optimization Methods and Software.* Kluwer Academic Publishers, Boston.

[6] Girosi Girosi, F., Jones, M. and Poggio, T. (1993). Priors, Stabilizers and Basis Functions: from regularization to radial, tensor and additive splines. *M.I.T. Artificial Intelligence Laboratory Memo* 1430, C.B.C.I. Paper 75.

[7] Hall, P., Park, B. U., and Turlach, B.A. (1998) A note on design transformation and binning in nonparametric curve estimation. *Biometrika*, **85(2)** 469-476.

[8] Johnstone, I. M. (1998). Function Estimation in Gaussian Noise: Sequence Models. Draft of a monograph. http://www-stat.stanford.edu/

[9] Lin, Y. and Brown, L. D. (2004). Statistical Properties of the Method of Regularization with Periodic Gaussian Reproducing Kernel. *Annals of Statistics.* **32** 1723-1743.

[10] Mallows, C.L. (1973). Some comments on C_p. *Technometrics* **15** 661-675.

[11] Shi, T., Yu, B., Clothiaux, E.E., Braverman, A. (2004) Cloud Detection over Snow and Ice based on MISR Data. *Technical Report 663, Department of Statistics, University of California.*

[12] Smola, A.J., Schoölkopf, B., and Müller, K. (1998). The connection between regularization operators and support vector kernels. *Neural Networks* **11** 637-649.

[13] Vapnik, V.(1995). *The Nature of Statistical Learning Theory.* Springer, N.Y., 1995.

[14] Wahba, G. (1990). *Spline Models for Obersvational Data.* Society for Industrial and Applied Mathematics. Philadelphia, Pennsylvania.

[15] Wahba, G., Lin, Y., and Zhang, H. (1999). GACV for Support Vector Machines, or Another Way to Look at Margin-Like Quantities, *Advanced in Large Margin Classifiers.*

[16] Williamson, R. C., Smola, A. J. and Schölkopf, B. (2001). Generalization performance of regularization networks and support vector machines via entropy numbers of compact operators. *IEEE Transactions on Information Theory* **47**, 2516-2532.

[17] Williams, C., and Seeger, M. (2000). The Effect of the Input Density Distribution on Kernel-based Classifiers. *International Conference on Machine Learning* **17**, 1159-1166.