Kernel Smoothing And Heat Equation

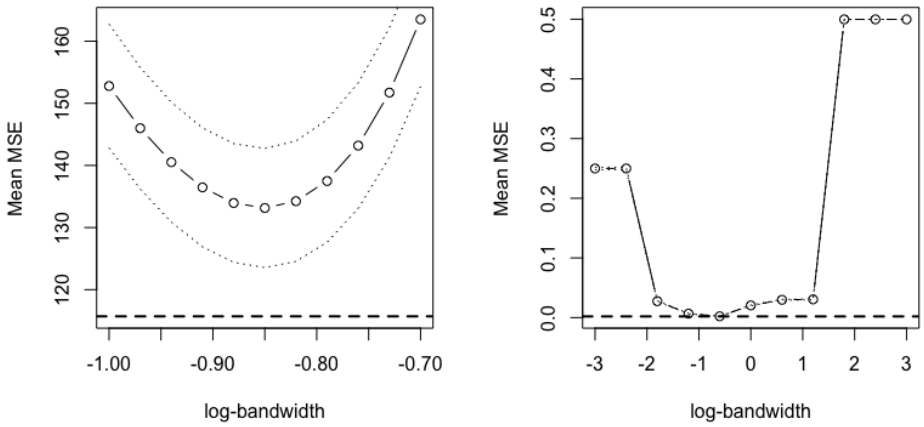
February 9, 2021

1 Notes

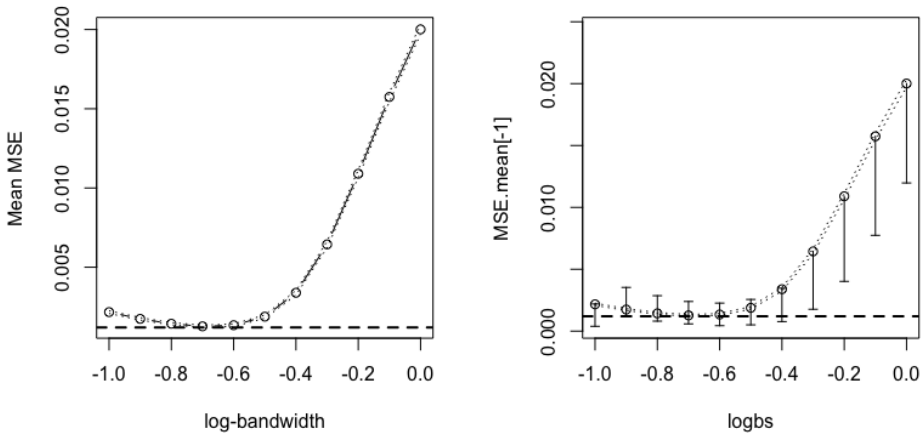
- 02/08/2021.

$$b = \big(\frac{L\sigma^2}{2\sqrt{\pi}(N-1)f''}\big)^{1/5} \tag{1}$$

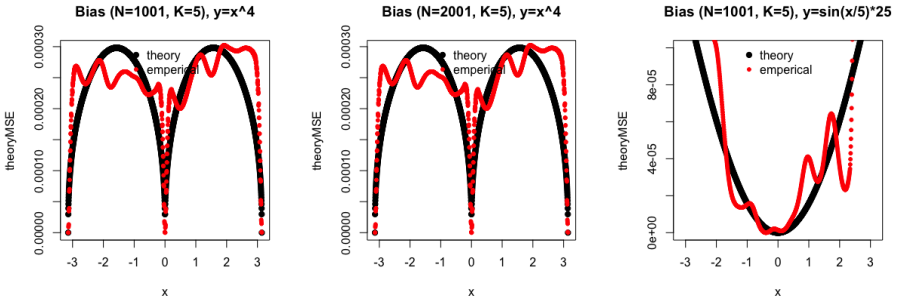
Plot MSE with logbandwidth (left: $y = x^4$; right: $y = \sin(x)$)



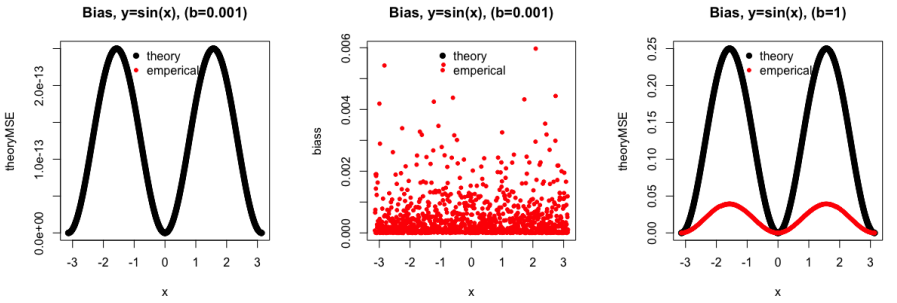
Plot MSE with logbandwidth with confidence interval
($y = \sin(x)$, N = 2001, rep = 200)



Plot sample size N versus bias (using adaptive bandwidth)



We notice that when bandwidth is too small or too large, the values fall out of the smooth window (2 left figures: small fixed b; right figure: large fixed b)



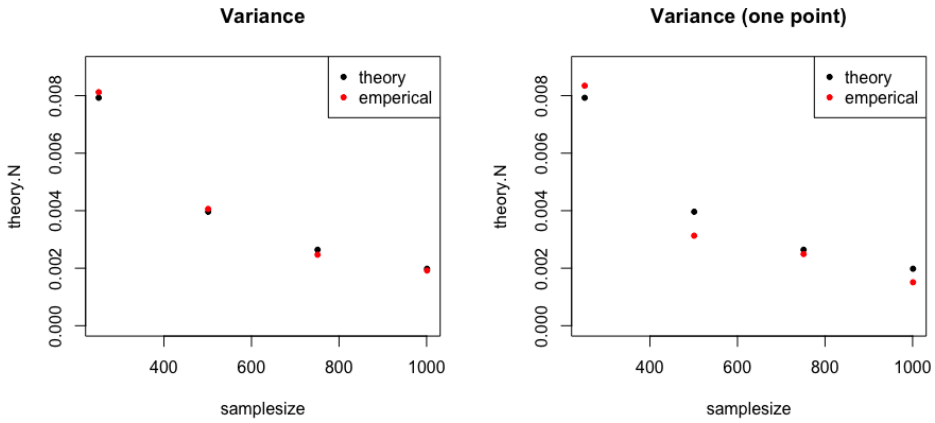
- 02/02/2021.

  We fixed the theoretical formula for both variance and bias:

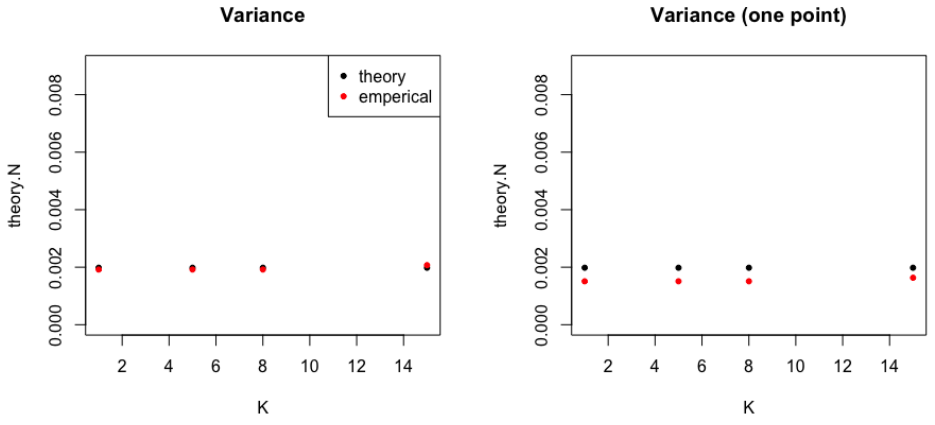
$$\mathrm{MSE}_j = \frac{L}{N-1} \cdot \frac{\sigma^2}{2\sqrt{\pi} \cdot b} + \left[\frac{\partial^2 u}{2\partial x^2} \cdot b^2 \right]^2 \tag{2}$$

  We plot the variance and bias^2 scenarios separately.

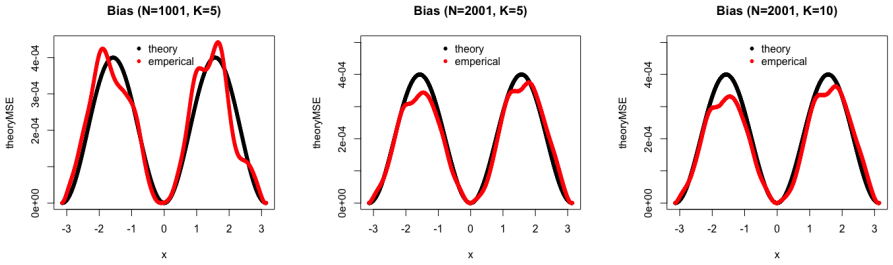
Plot sample size N versus variance (left is the empirical variance mean, and right is the certain point variance.)



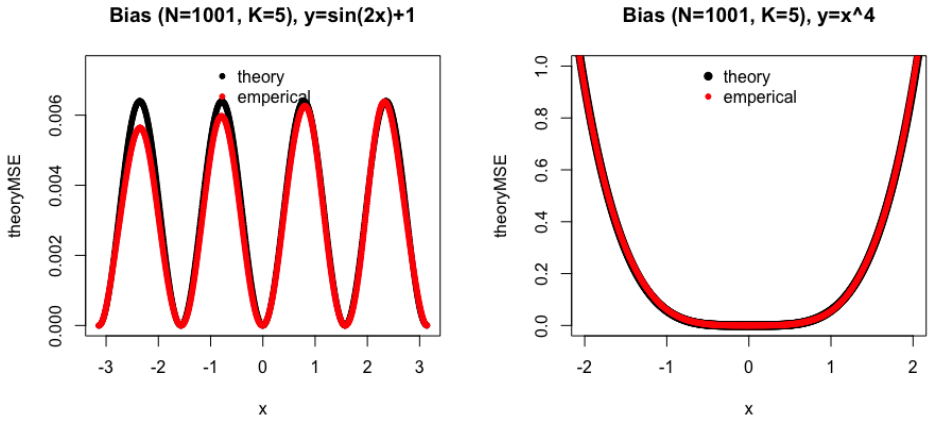
Plot smooth.window parameter K versus variance



Plot sample size N versus bias



Plot sample size N versus bias



TODOLIST:

1. find the optimal smoother and redraw the figures.
2. prove adaptive smoothing is an advantage
3. replace variate kernel smoother to heat equation

$$\begin{aligned}\frac{\partial u}{\partial t} &= \nabla_x(c(x)\nabla_x u) \\ &\approx c(x)\nabla^2 u\end{aligned}$$

4. implement finite differential solver (compare diffusion coefficient)
5. long term: finite different method (FDM) and finite element method (FEM)

- 01/26/2021.

In order to find out the reason that empirical MSE and theoretical MSE look so different, we split them into $\texttt{bias}^2$ and $\texttt{variance}$.

For fixed bandwidth scenario, we tried with the below 3 possible theoretical MSE with sample size $N = 501$ and simulation time $n = 500$:

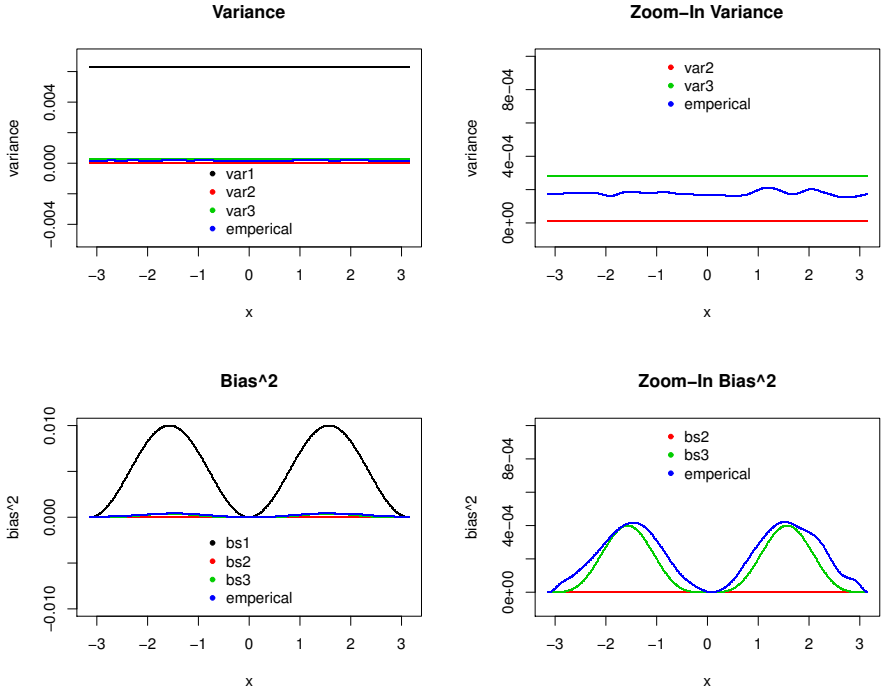
$$\text{MSE}_j = \frac{\sigma^2}{2\sqrt{\pi} \cdot \sqrt{b}} + \left[\frac{\partial^2 u}{2\partial x^2} \cdot b\right]^2 \tag{3}$$

$$\text{MSE}_j = \frac{\sigma^2}{2N\sqrt{\pi} \cdot \sqrt{b}} + \frac{20}{N^2} \cdot \left[\frac{\partial^2 u}{2\partial x^2} \cdot b\right]^2 \tag{4}$$

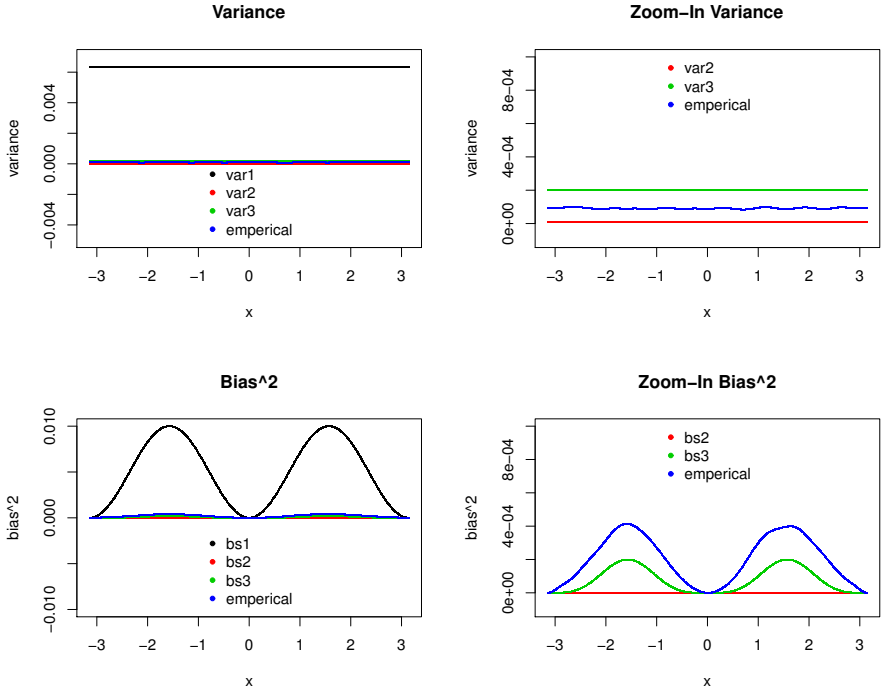
$$\text{MSE}_j = \frac{\sigma^2}{2\sqrt{N}\sqrt{\pi} \cdot \sqrt{b}} + \frac{20}{N} \cdot \left[\frac{\partial^2 u}{2\partial x^2} \cdot b\right]^2 \tag{5}$$

Empirical MSE:

$$\begin{aligned}
\text{MSE}_j &= \mathbb{E}[(\hat{y} - u)^2] \\
&= \mathbb{E}[(\hat{y} - \mathbb{E}(\hat{y}))^2] + \mathbb{E}[\mathbb{E}(\hat{y}) - u]^2 \\
&= \texttt{var}(\hat{y}) + \texttt{bias}^2(\hat{y})
\end{aligned}$$



Increase sample size from N=501 to N=1001:



TODOLIST:

1. plot `variance` vs sample size (N) plot to see the relationship
2. take a close look of `bias` term: Taylor expansion on a fix point (eg.: $f(x) = \sin(x)$), to see if the first term goes to $f(x_0)$, second term goes to 0 and third term goes to $f''(x_0)b^2/2$
3. After solve the fix bandwidth scenario issue, try adaptive bandwidth

- 01/19/2021. Checked theoretical MSE vs empirical MSE at the whole scale. Explored the scenarios with increasing k value (to narrow the smooth.window) and increasing sample size.
  We noticed that the empirical MSE is smaller than theoretical MSE. And theoretical has certain pattern while empirical MSE is more in randomness. We suspect our theoretical derivation (Eq (1)) is incorrect.

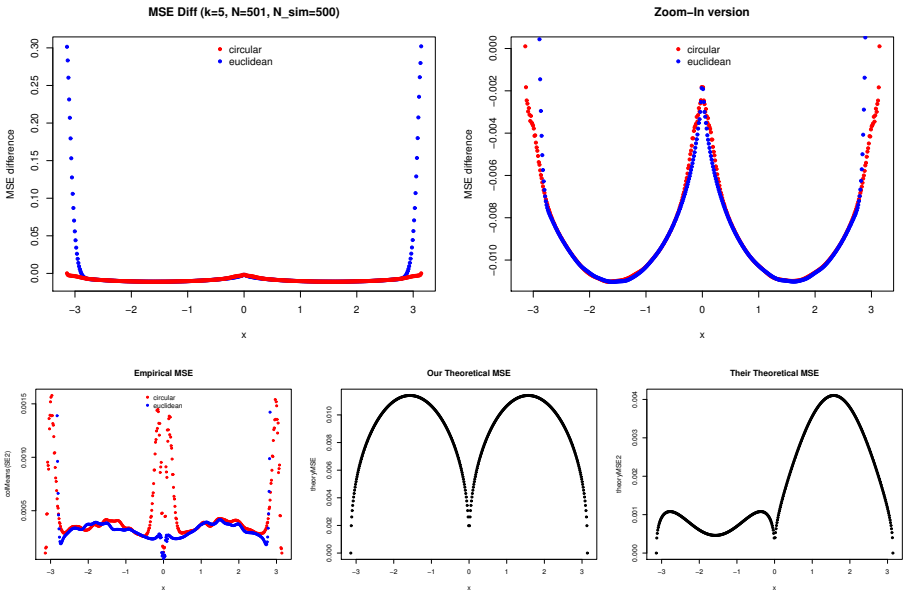
$$\text{MSE}_j = \frac{\sigma^2}{2\sqrt{\pi} \cdot \sqrt{b}} + \left[\frac{\partial^2 u}{2\partial x^2} \cdot b\right]^2 \tag{6}$$

We found the MSE of KDE online (UW lecture notes) expressed as (Eq (3))

$$\text{MSE}_j = (nh)^{-1} f(x) \int K(z)^2 dz + \left(\frac{f''(x)}{2} h^2 \int z^2 K(z) dz\right)^2 \tag{7}$$

$$= \frac{u}{2\sqrt{\pi} \cdot n\sqrt{b}} + \left[\frac{\partial^2 u}{2\partial x^2} \cdot b\right]^2 \tag{8}$$

We then use their theoretical result but it still shows the certain pattern that our empirical MSE didn't reveal. We are wondering if our theoretical result is problematic, why our emperical result, which uses adaptive bandwidth ($b = \left(\frac{\sigma^2}{2\sqrt{\pi} u''^2}\right)^{\frac{2}{5}}$), still generates quite good estimation?



TODOLIST:

1. rederivate formular (N should be in the denominator)
2. plot variance and bias seperately to see which one is more problematic
3. adaptive bandwidth should give constant MSE
4. Think about non-constant adaptive Gaussian smoother and inhomogeneous heat equation:

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}(c \cdot \frac{\partial u}{\partial t}) = c \cdot u''_x$$

when c is not a constant but a function $c(x)$, the equation becomes

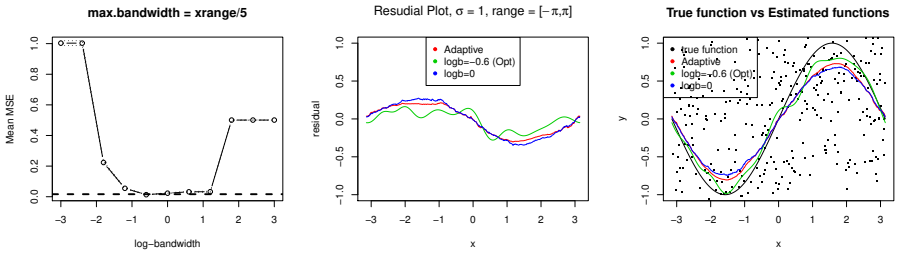
$$\frac{\partial u}{\partial t} = c'(x)u'_x + c \cdot u''_x$$

$c'(x)u'_x$ can be understood as tuning parameter in heat equation, where c shows how fast the heat transform from one locaton to another (spatial discrepancy). We shall explore the relationship between c in heat equation and bandwidth b in kernel smoother.

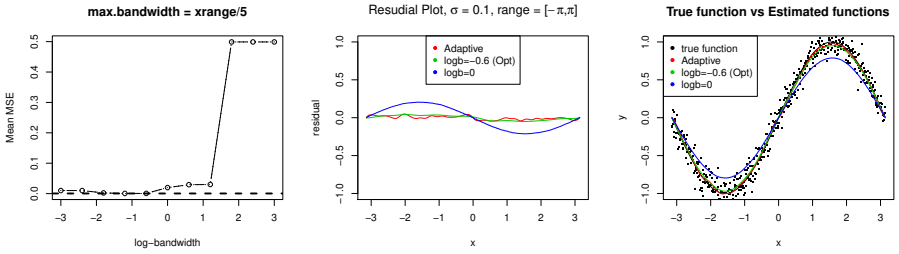
- 12/08/2020. Fixed the boundary issue for circular situation.

k	(theoretical MSE $=1.96 \times 10^{-7}$) empirical MSE ($\sigma = 1$)	(theoretical MSE $=4.93 \times 10^{-9}$) empirical MSE ($\sigma = 0.1$)
0.1	$0.0020\ (6.65 \times 10^{-6})$	$2.02 \times 10^{-5}(7.35 \times 10^{-10})$
1	$0.0021\ (9.69 \times 10^{-6})$	$1.95 \times 10^{-5}(7.00 \times 10^{-10})$
2	$0.0021\ (8.33 \times 10^{-6})$	$1.95 \times 10^{-5}(7.17 \times 10^{-10})$
2.4	$0.0024\ (1.22 \times 10^{-5})$	$2.41 \times 10^{-5}(1.17 \times 10^{-9})$
3	$0.0029\ (1.51 \times 10^{-5})$	$2.82 \times 10^{-5}(1.59 \times 10^{-9})$
4	$0.0041\ (3.52 \times 10^{-5})$	$4.36 \times 10^{-5}(3.42 \times 10^{-9})$
5	$0.0050\ (5.36 \times 10^{-5})$	$5.07 \times 10^{-5}(5.24 \times 10^{-9})$

Table 1: 1000 times simulation for empirical MSE at extreme point at boundary




```
  Adaptive    logb=-3  logb=-2.4  logb=-1.8  logb=-1.2  logb=-0.6
0.01650001 1.00313133 1.00313133 0.22355199 0.05497094 0.01329125
    logb=0   logb=0.6   logb=1.2   logb=1.8   logb=2.4     logb=3
0.02305490 0.03210026 0.03280336 0.50068896 0.50068896 0.50068896
```




```
    Adaptive      logb=-3    logb=-2.4    logb=-1.8    logb=-1.2    logb=-0.6
0.0004494958 0.0100313133 0.0100313133 0.0022356024 0.0005502092 0.0005894302
      logb=0     logb=0.6     logb=1.2     logb=1.8     logb=2.4       logb=3
0.0197699286 0.0292714857 0.0299909440 0.4990188656 0.4990188656 0.4990188656
```

TODOLIST:

1. Check theoretical MSE vs empirical MSE

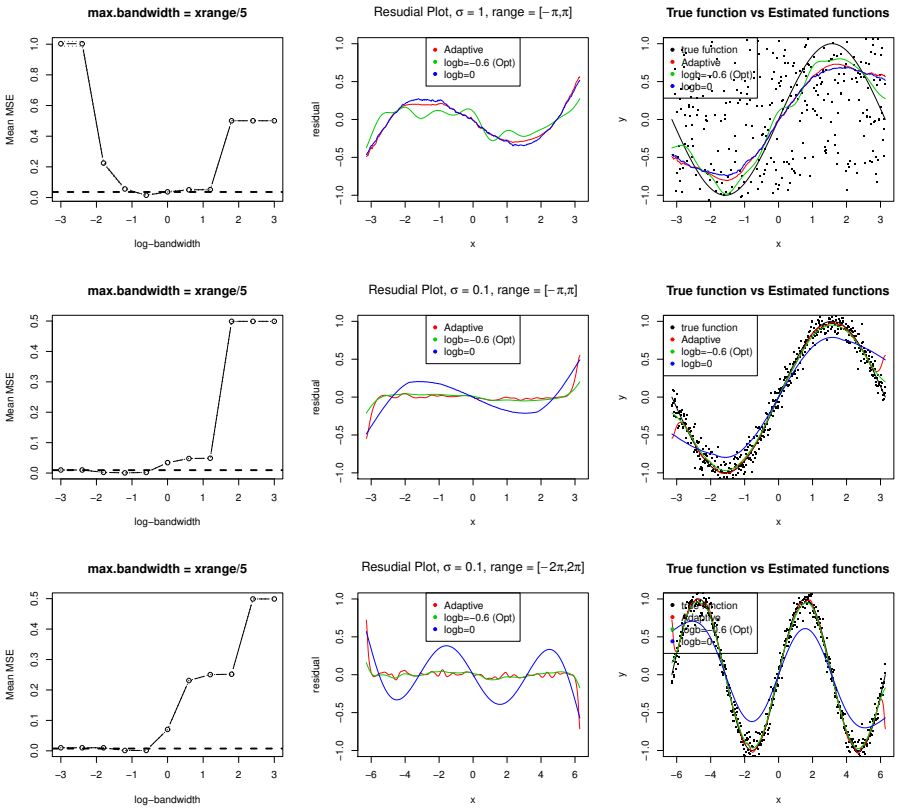
$$\sin^2(t) = \sin(t) - \sin(t_0) - \cos(t_0)(t - t_0)$$

   - $\text{bias}^2 \propto \sin^2(t)$
   - $\text{variance} \propto \frac{\sigma^2}{n_{\texttt{neighbor}}}$

   increase k to narrow the smooth.window $\frac{\texttt{xrange}}{k}$

2. Map non-constant adaptive Gaussian smoother back to inhomogeneous heat equation

- 12/01/2020.



Although the adaptive bandwidth smoother perform well in general, but the boundary behavior of the adaptive bandwidth smoother is not ideal. We then modified `max.window` from `xrange/5` to `xrange/0.1`. The situation gets better.

We explored the extreme points by

1. computing the difference between the empirical value `mean(y.within.window)` and the true value `u`
2. comparing the empirical MSE with the theoretical MSE of the extreme points

Denote $\texttt{max.window} = \frac{\texttt{xrange}}{k}$, we plot the difference between the empirical value `mean(y.within.window)` and the true value `u` with respect to k at extreme points. We noticed when $k > 2$, the difference would increase tremendously (Figure below).

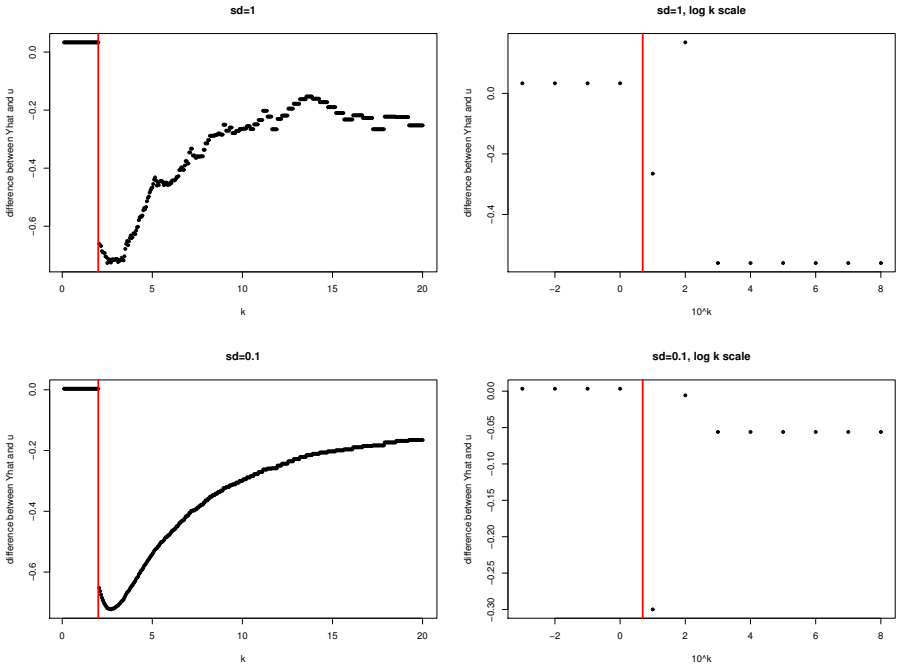


Figure 1: Extreme point (first point) of $y = \sin(x)$ with $\sigma = 1$ and $\sigma = 0.1$ scenarios. We use log-scale plot to see if the difference would converge asymptomatically as $k \to +\infty$. The red line denotes when $k = 2$, the difference has a jump.

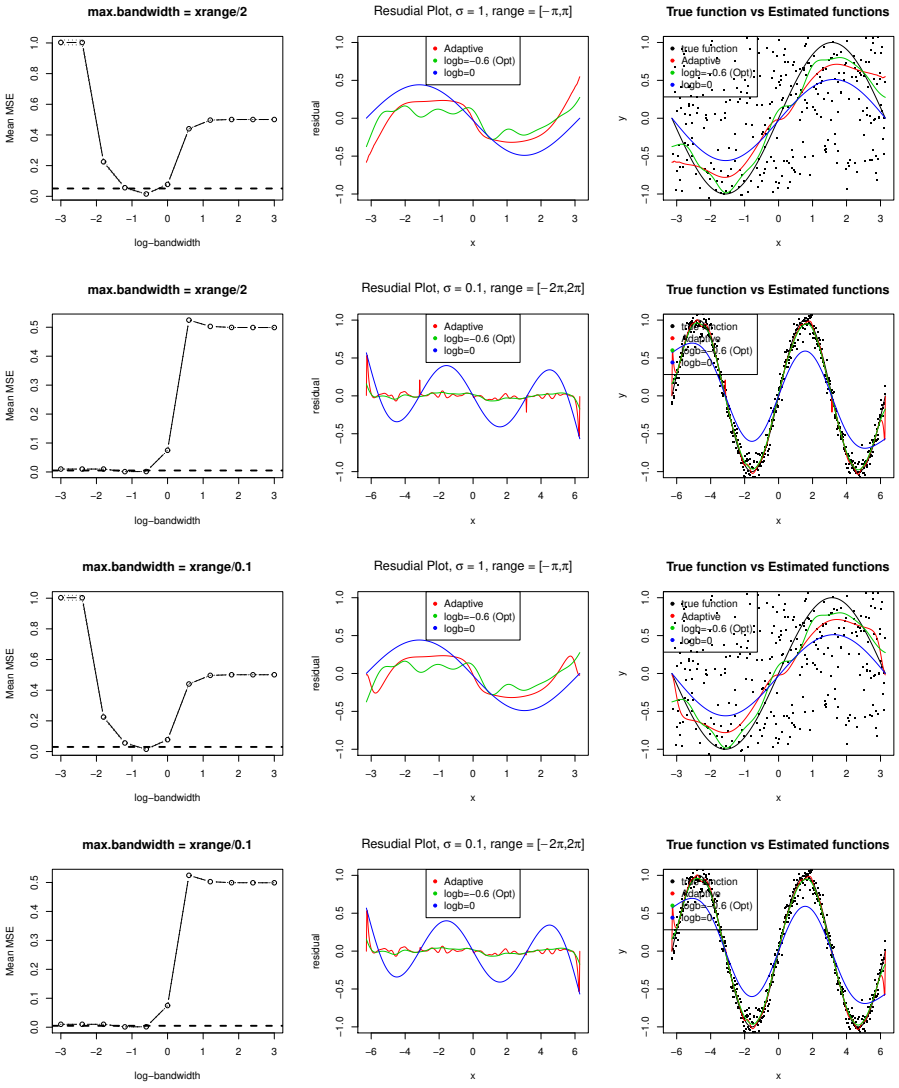
Now we compare the empirical MSE with the theoretical MSE on the extreme point at the boundary.

Flash back:

$$\text{MSE}_j = \frac{\sigma^2}{2\sqrt{\pi \cdot b}} + \left[\frac{\partial^2 u}{2\partial x^2} \cdot b\right]^2$$

k	(theoretical MSE $=1.96 \times 10^{-7}$) empirical MSE ($\sigma = 1$)	(theoretical MSE $=3.95 \times 10^{-7}$) empirical MSE ($\sigma = 0.1$)
0.1	0.0019 (8.15×10^{-6})	0.0021 (7.59×10^{-6})
1	0.0020 (7.80×10^{-6})	0.0019 (6.76×10^{-6})
2	0.0020 (8.09×10^{-6})	0.0019 (7.09×10^{-6})
2.1	0.4448 (0.0072)	0.4408 (0.0071)
3	0.5124 (0.0128)	0.5220 (0.0123)
4	0.4107 (0.0122)	0.4070 (0.0131)
5	0.3116 (0.0124)	0.3147 (0.0136)

Table 2: 1000 times simulation for empirical MSE at extreme point at boundary



- 11/17/2020.
  TODOLIST:
  1. Editor: Emacs, Vim
  2. Learn: Command line, `ssh`, `zsh`,`bash`, to remotely control.
  3. `.eps` → `.eps`; change `postscript()` to `pdf(), png(), tiff()`
  4. Residual trend means under-smoothing (overfitting)
  5. manually compute the 2 extreme points (second order taylor expansion) – empirical MSE and theoretical MSE

- 11/10/2020. We implemented circular smoother on both fixed bandwidth and adaptive bandwidth scenarios; and did the corresponding simulation. We noticed for the circular data, the adaptive bandwidth cannot beat the optimal fixed bandwidth. Fixed bandwidth achieve the smallest MSE in our $y = \sin(x)$ setting. We also noticed the circular smoother performs better than Euclidean smoother in the circular data ($y = \sin(x)$).

```
           euclidean   circular
Adaptive  0.19821445 0.19821445
logb=-3   1.00313133 1.00313133
logb=-2.4 1.00313133 1.00313133
logb=-1.8 1.00313133 1.00313133
logb=-1.2 0.11346202 0.11227971
logb=-0.6 0.02875581 0.02717735
logb=0    0.08004308 0.08145240
logb=0.6  0.44585526 0.52684029
logb=1.2  0.49326980 0.50454607
logb=1.8  0.50018979 0.50094139
logb=2.4  0.50068896 0.50068896
logb=3    0.50068896 0.50068896
```

  TODOLIST:

  1. Compare MSE at 2 boundaries
  2. manually do the smoothing for certain points, compare the two bandwidth (adaptive vs fixed); expected MSE and empirical MSE should agree. Two scenarios:
     (a) manual result doesn't equal to the automatic one, debug the result
     (b) manual result equal to the auto one. Seek the reason: Taylor expansion is not good enough, higher order term is needed (More observation, smaller bandwidth, smaller window)
  3. Future: explore the inhomogeneous heat equation (adaptive $\lambda(x)$)

$$\frac{\partial f(x,t)}{\partial t} = \Delta f = \frac{\partial f}{\partial x}[\lambda(x)\frac{\partial f}{\partial x}] \neq \lambda(x)\frac{\partial^2 f}{\partial x^2}$$

     Google the connection between adaptive smoothing and inhomogeneous heat equation

- 10/27/2020. We reviewed Dr. Qiu's code/documentation. Todos:

  1. Compare our old code with Dr. Qiu's new code one more time.
  2. Try to implement the circular smoother (at least for the fixed bandwidth case).

3. We've already studied the mathematical connections between the classical (homogeneous) heat equation and Gaussian kernel smoothing. We need to document this important connection in this document.
4. We need to think: what is the mathematical connection between adaptive Gaussian kernel smoother and inhomogeneous heat equation? Rationale: we've already derived (based on large sample theory) the "optimal" variable bandwidth for kernel smoother. If the kernel smoother is equivalent to an inhomogeneous differential equation, it means that we can then use an efficient numerical DE solver (e.g., those based on finite element method) to do kernel smoothing, especially for multi-dimensional cases.
5. In the long run, we need to develop a practical estimation procedure for $x(t)$ and $\sigma^2(t)$, better with some model selection procedure so the entire estimation procedure can be automated.
6. Multi-dimensional!! (a) the curse of dimensionality, (b) much more complicated boundary to deal with.

2 Introduction

It is universally acknowledged that observed data with respect to the underlying patterns we are seeking for in the physical world is, to some extent, contaminated with random noise. Commonly, the observed data would be expressed into two parts: the systematic component (i.e.: a true underlying oracle function $u(x)$) and a random component (i.e.: noise ϵ). Yet, due to the fundamental sorrow of the limited observations in an infinite world, we may fail to achieve the oracle function. In fact, we can only estimate it based on our finite noise infested observations.

For decades, statisticians have been proposed various methods to estimate the oracle function. One of the non-parametric ways is to use direct diffusion to smooth away all the noise. In an absolute non-rigorous sense, the direction diffusion process achieves the oracle risk by (weighted) averaging each observation neighbor. Each time we do this, our estimated function will be closer to the true function until at a finite time T we achieve our goal. After that the estimated function will be bounded away from the truth. People coin the term "over-smoothing" to indicate this phenomenon. Our goal is to discover the optimal T where we are closest to the truth given an observed data set.

However, the diffusion process is not widely used among statisticians. Instead, they use kernel smoothing, a nonparametric method proved to be equivalent to the heat equation in physics world. Therefore, finding optimal time T can be translated to the problem of finding the optimal sigma. In statistics literature (**??** reference needed), the optimal sigma is found to be a constant for every point x. This is a significant constraint. Hence, in this paper, we will allow the spatial bandwidth to be different for each point. In the next sec-

tion, we will derive the optimal sigma for each observation point. We will do simulations to show that our "adaptive" bandwidth performs better than the best-fixed bandwidth.

Some literature review...

Goal of this paper...

Structure of the paper...

3 Methodology

The connection between Heat Equation and Kernel Smoothing can be expressed (**??**reference for Equation (9)) as

$$T = \frac{1}{2}\sigma^2 \tag{9}$$

where T denotes the total smoothing time in the direction process, and σ denotes the proportion of the spatial width of the Gaussian smoother in paper **??** (Equation (10)).

$$\sigma = \text{bandwidth} \times 0.3706506 \tag{10}$$

Given $\mathbf{x}_i = (x_1, x_2, ..., x_p)^\top \subseteq \mathbb{R}^p$, $Y_i \subseteq R$, we have the data set

$$\mathcal{D}_n = \big\{(x_1, y_1), ...(x_n, y_n)\big\}$$

Denote $u(\mathbf{x})$ as the oracle function, and $u(\mathbf{x}_i)$ is the first order Taylor expansion

$$u(\mathbf{x}_i) \approx u(\mathbf{x}) + \frac{\partial u}{\partial \mathbf{x}}(\mathbf{x}_i - \mathbf{x}) \tag{11}$$

And given a Gaussian kernel with b as the bandwidth parameter

$$\phi_b(x) = \frac{1}{\sqrt{2\pi b}} \exp(\frac{-x^2}{2b}) \tag{12}$$

We have MSE at $\mathbf{x}_i$ with respect to b

$$\text{MSE}_j = \frac{\sigma^2}{2\sqrt{\pi \cdot b}} + \left[\frac{\partial^2 u}{2\partial x^2} \cdot b\right]^2 \tag{13}$$

Take the derivative with respective to b, we can solve the equation

$$b = \left(\frac{\sigma^2}{2\sqrt{\pi} u''^2}\right)^{\frac{2}{5}} \tag{14}$$

(Detailed calculation in Appendix)

4 Appendix

We let

$$\phi_b(x) = \frac{1}{\sqrt{2\pi b}} \exp(\frac{-x^2}{2b})$$

where b is the band width parameter. Then we have our MSE at x_i

$$\begin{aligned}
(\hat{u}(x_i) - u(x_i)) &= \sigma^2 \sum_{i=1}^{n} w_i^2 + \left[\frac{\partial u}{\partial x} \cdot \sum_{i=1}^{n} w_i(x_i - x)\right] + \left[\frac{\partial^2 u}{2\partial x^2} \cdot \sum_{i=1}^{n} w_i(x_i - x)^2\right] + \sum_{i=1}^{n} \epsilon_i w_i \\
E\left[(\hat{u}(x_i) - u(x_i))^2\right] &= Var\left[(\hat{u}(x_i) - u(x_i))\right] + \left(E\left[(\hat{u}(x_i) - u(x_i))\right]\right)^2 \\
&= \sigma^2 \sum_{i=1}^{n} w_i^2 + \left[\frac{\partial u}{\partial x} \cdot \sum_{i=1}^{n} w_i(x_i - x) + \frac{\partial^2 u}{2\partial x^2} \cdot \sum_{i=1}^{n} w_i(x_i - x)^2\right]^2 \\
MSE_j &= \sigma^2 \sum_{i=1}^{n} \phi_b^2(x_j - y_i) + \left[\frac{\partial u}{\partial x} \cdot \sum_{i=1}^{n} \phi_b(x_j - y_i) \cdot (x_j - y_i) + \frac{\partial^2 u}{2\partial x^2} \cdot \sum_{i=1}^{n} \phi_b(x_j - y_i) \cdot (x_j - y_i)^2\right]^2
\end{aligned}$$

Now we replace summation with integration

$$\begin{aligned}
MSE_j &= \sigma^2 \int_{\omega} \phi_b^2(x_i - y)dy + \left[\frac{\partial u}{\partial x} \cdot \int_{\omega} \phi_b(x_j - y) \cdot (x_j - y)dy + \frac{\partial^2 u}{2\partial x^2} \cdot \int_{\omega} \phi_b(x_j - y) \cdot (x_j - y)^2 dy\right]^2 \\
&= \frac{\sigma^2}{2\sqrt{\pi \cdot b}} + 0 + \left[\frac{\partial^2 u}{2\partial x^2} \cdot \int_{\omega} \phi_b(x_j - y) \cdot (x_j^2 - 2x_j \cdot y + y^2)dy\right]^2 \\
&= \frac{\sigma^2}{2\sqrt{\pi \cdot b}} + \left[\frac{\partial^2 u}{2\partial x^2} \cdot \int_{\omega} \phi_b(x_j - y)(x_j^2)dy - 2\int_{\omega} \phi_b(x_j - y) \cdot (x_j \cdot y)dy + \int_{\omega} \phi_b(x_j - y) \cdot y^2 dy\right]^2 \\
&= \frac{\sigma^2}{2\sqrt{\pi \cdot b}} + \left[\frac{\partial^2 u}{2\partial x^2} \cdot (x_j^2 - 2x_j^2 + b + x_j^2)\right]^2 \\
&= \frac{\sigma^2}{2\sqrt{\pi \cdot b}} + \left[\frac{\partial^2 u}{2\partial x^2} \cdot b\right]^2
\end{aligned}$$

Set $h(b) = \frac{\sigma^2}{2\sqrt{\pi}} \cdot \frac{1}{\sqrt{b}} + \frac{1}{4}u''^2 \cdot b^2$, and take derivative, then set $h'(b) = 0$

$$\begin{aligned}
h'(b) &= -\frac{\sigma^2}{4\sqrt{\pi}} \cdot b^{-\frac{3}{2}} + \frac{1}{2}u'' b = 0 \\
b &= \left(\frac{\sigma^2}{2\sqrt{\pi}u''^2}\right)^{\frac{2}{5}}
\end{aligned}$$