Local affinity construction for dimension reduction methods

Ajay Dheeraj, Inchan Hwang, Tyler Lian, Joseph Saldutti

DOmath 2018*

July 21, 2018

Abstract

Accompanying the rise of large data in high dimensions (buzzword: “big data”) is an increasing necessity for effective dimension reduction methods to make that data more amenable to study. In general, the goal of dimension reduction is to reduce the dimension of the data while preserving or extracting its key features or other properties of interest. Spectral graph-based kernel techniques like diffusion maps rely on the convergence of the graph Laplacian matrix to the Laplace-Beltrami operator on a manifold to preserve local geometry information of the original data to the low dimension embedding. Extending previous research on variable bandwidth kernels, this project studied the converge of a new proposed kernel. The main desired advantage of this new kernel is to remove the computational dependence of the intrinsic dimension of the given data (which is difficult to derive reliably) by utilizing a $k$-nearest neighbors method.

1 Introduction

Accompanying the rise of large data in high dimensions (buzzword: “big data”) is an increasing necessity for effective dimension reduction methods to make that data more amenable to study. Usually, dimension reduction

---

* sponsored by DOmath 2018 of the Duke Math Dept. Special thanks to Didong Li, Dr. Xiuyuan Cheng, and Dr. Hau-Tieng Wu for pushing us forward and making this possible.
1 INTRODUCTION

is necessary as a first step for visualization, classification, or other machine learning applications of high-dimension data. In general, the goal of dimension reduction is to reduce the dimension of the data while preserving or extracting its key features or other properties of interest. That is to say that the low-dimension projection should faithfully represent the original data in some way in order to be meaningful.

Diffusion maps and eigenmaps ([3], [1]) are one such approach to dimension reduction. These are nonlinear, graph-based kernel techniques that aim to preserve the local neighborhood geometry of the data (as opposed to classical techniques like PCA). A weighted graph is constructed on the data points using a kernel that decays as the Euclidean distance between two points increases. Eigenvectors of the resulting (random walk) graph Laplacian matrix are then the optimal projections of the data to low-dimension space such that the cost of mapping neighboring points far apart is minimized. Namely, these eigenvectors are the mappings that best preserve the pairwise diffusion distances between points, which is a measure of the connectivity between points on the random walk. (See Coifman and Lafon, [3]).

These algorithms are justified by their connection to the Laplace-Beltrami operator on a manifold, and have geometric interpretations. In these techniques, high-dimension data points are assumed to have the structure of an underlying lower-dimension topological manifold. Most notably, the Laplacian matrix of a graph is analogous and converges to the Laplace-Beltrami operator of a manifold. Further, eigenvectors of the graph Laplacian converge pointwise to eigenfunctions of the Laplace-Beltrami operator. This convergence is a central topic of study for those interested in these dimension reduction techniques.

In [2], the convergence analysis of past research is extended to variable bandwidth kernels of the form

$$K_{\epsilon}(x, y) = h \left( \frac{\|x - y\|^2}{\epsilon \rho(x) \rho(y)} \right),$$

(1)

where $h$ decays exponentially. Kernels of this form are used in most applications of kernel techniques rather than the fixed bandwidth kernels ($\rho(x) = 1$) described in the original theory. The advantage of variable bandwidth kernels (a.k.a. self-tuning kernels) over fixed bandwidth ($\rho = 1$) kernels is that the non-constant bandwidth function $\rho$ regularizes the output of the kernel to better handle regions of sparse sampling.
The results of Berry and Harlim in [2], though rigorous, are hindered by a dependence on the intrinsic dimension $d$ of the data in their numerical algorithm. Though techniques exist to estimate $d$, this value usually cannot be easily or accurately divined from the data in practice.

This motivates the research of this report, which analyzes the convergence of the graph Laplacian matrix constructed using a new original kernel that takes advantage of the properties of the $k$-nearest neighbor algorithm. Since the $k$-nearest neighbor algorithm is used as a kernel density estimation, this implementation does not require the estimation of the intrinsic dimension $d$ and simpler to compute on the whole.

This report is organized as follows: Section 2 details the main theory behind our new kernel of interest. Section 3 outlines the procedure used for the numerical implementation of the algorithm and provides results of empirical experiments on $S_t$. Section 4 applies the new kernel to more real-world MNIST data.

2 Convergence analysis

We begin by extending the analysis of [2] to a new kernel that we call $K_\rho$. Then, we make a connection between $K_\rho$ and our kernel of interest $K_\sigma$, which utilizes the $k$-nearest neighbors algorithm to construct a variable bandwidth function.

2.1 Main theorem

**Theorem 1.** Suppose $p$ is a probability density function defined on a manifold $\mathcal{M} \subset \mathbb{R}^n$. Let $\{x_i\}_{i=1}^N$ be points sampled independently from $p$. For the kernel $K_\rho$ defined to be

$$K_\rho(x, y) = \frac{\exp(-\|x - y\|^2/\epsilon \rho(x) \rho(y))}{\rho(x) \rho(y)} \quad (2)$$

where $\rho = p^{-1/d}$, then

$$L_{n, \epsilon}^p \equiv \frac{1}{\epsilon m \rho^2} L_{rw} = \Delta f(x_i) + \frac{\nabla p}{p} \cdot \nabla f(x_i) + O \left( \epsilon, \|\nabla f(x_i)\| \epsilon^{-d/4-1/2} \right), \quad (3)$$
with high probability, where $L_{rw} = I - D^{-1}W$ is the random walk graph Laplacian matrix.

Proof. The proof of this theorem is given in the appendices. The limiting operator and the bias term are derived in Appendix A. The variance term is derived in Appendix B.

Here, we see that the matrix $L^\varrho_\epsilon$ converges to the biased Laplacian operator $\Delta + \frac{\nabla p}{p} \cdot \nabla$. For convenience, we will refer to this operator as $\Delta_\rho$.

2.2 Introducing $k$-nearest neighbor bandwidth

Notice that the use of this kernel $K_\rho$ still requires an estimation of the intrinsic dimension $d$, as $\rho = p^{-1/d}$. We can make a connection, however, between $\rho$ and the $k$-nearest neighbor algorithm.

First, we define a $k$-nearest neighbors bandwidth function $\sigma$.

**Definition 2.1.** Let $\{x_i\}_{i=1}^n$ be a set of points. Define the $k$-th nearest neighbor variable bandwidth function $\sigma$ to be

$$\sigma(x_i) = d(x_i, x^{(k)}),$$

that is the Euclidean distance between $x_i$ and $x^{(k)}$, the $k$th-nearest neighbor to $x_i$.

We can see that $\sigma$ satisfies the general properties necessary for a useful variable bandwidth function. For densely sampled regions, $\sigma$ is small, and for sparsely sampled regions, $\sigma$ is large. Its only parameter is $k$, and most importantly, it can computed without any required knowledge of the intrinsic dimension of the points or the underlying sampling distribution.

**Proposition 2.** Suppose $\sigma$ is defined as above, and $\{x_i\}_{i=1}^n$ with intrinsic dimension $d$ is sampled from a probability density $p$, with $k > 0$. Then, for any $x_i$, as $n$ increases,

$$\sigma(x_i) \approx \left(\frac{k}{c_d n}\right)^{1/d} p(x_0)^{-1/d},$$

where $c_d \in \mathbb{R}$ is dependent solely on $d$. 

4
2 CONVERGENCE ANALYSIS

Proof. Let $x_0$ be any point from the set. The $k$-nearest neighbors of $x_0$ are enclosed in a $d$-ball of radius $\sigma(x_0)$ centered at $x_0$. For convenience of notation, let $r = \sigma(x_0)$. Then, as $n \to \infty$,

$$\frac{k}{n} = \int_{B_r^d(x_0)} p(y)dy,$$

where $B_r^d(x_0)$ is a $d$-dimensional ball with radius $r$ centered at $x_0$. This approximation should hold as long as $n$ is large enough to minimize sampling bias from $p$.

We want to solve for $r$ to obtain an explicit formula for $\sigma$. Performing a Taylor expansion around the center $x_0$ of the ball, we have

$$p(y) = p(x_0) + (\nabla p(x_0))^T(y - x_0) + \frac{1}{2}(y - x_0)^T H_p(x_0)(y - x_0) + \ldots$$

for $r << 1$, where $H_p$ is the Hessian matrix.

This can be integrated term by term. Considering the first term only gives

$$\int_{B_r^d(x_0)} p(x_0)dy = c_d r^d p(x_0),$$

where $c_d$ is a constant such that $c_d r^d$ is the volume of $B_r^d(x_0)$. In other words, $c_d$ is the volume of the unit $d$-ball.

To estimate the error of this expansion, we calculate the second term

$$\int_{B_r^d(x_0)} \sum_i \frac{\partial p}{\partial y_i}(x_0)(y - x_0)_i dy = 0,$$

by symmetry.

Moving to the third term, we compute the matrix product with the Hessian and pull out the summation and the $\frac{1}{2} \frac{\partial^2 p}{\partial y_i \partial y_j}(x)$ terms from the integral, resulting in

$$\frac{1}{2} \sum_{i,j} \left( \frac{\partial^2 p}{\partial y_i \partial y_j}(x_0) \int_{B_r^d(x_0)} (y - x_0)_i(y - x_0)_j dy \right).$$

When $i \neq j$, by the symmetry of $B_r^d(x_0)$, we have both $y - x$ and $y - x$ with sign of $j$'th component flipped within the ball. Then they cancel each other out for any $(y - x)_i$, leaving us only with $y - x$ with zero $j$'th component, which also vanishes. Hence, the sum reduces to
\[
\frac{1}{2} \sum_i \left( \frac{\partial^2 p}{\partial y_i^2}(x_0) \int_{B_i^\epsilon(x_0)} (y - x_0)^2 \, dy \right).
\]

Since \( \frac{\partial^2 p}{\partial y_i^2} \) is bounded, we can say there exists an \( M \) such that \( \frac{\partial^2 p}{\partial y_i^2}(x_0) \leq M \) for all \( i \). Then,

\[
\frac{1}{2} \sum_i \left( \frac{\partial^2 p}{\partial y_i^2}(x_0) \int_{B_i^\epsilon(x_0)} (y - x_0)^2 \, dy \right) \leq \frac{M}{2} \sum_i \int_{B_i^\epsilon(x_0)} (y - x_0)^2 \, dy,
\]

which is simply

\[
\frac{M}{2} \int_{B_i^\epsilon(x_0)} \| y - x_0 \|^2 \, dy = O(r^{d+2}).
\]

Hence, we have

\[
k \frac{1}{n} = c_d r^d p(x_0) + O(r^{d+2}),
\]
as \( r \to 0 \). Note that \( r \to 0 \) as \( n \to \infty \), as long as \( \frac{k}{n} \to 0 \).

Solving for \( r \),

\[
\sigma(x_0) \approx \left( \frac{k}{c_d n} \right)^{1/d} p(x_0)^{-1/d}
\]
as stated. \( \square \)

Therefore, if we let \( \rho = p^{-1/d} \) and \( \sqrt{\epsilon} = \left( \frac{k}{c_d n} \right)^{1/d} \), then

\[
\sigma \approx \sqrt{\epsilon} \rho \quad \text{and} \quad \sigma(x)\sigma(y) \approx \epsilon \rho(x)\rho(y).
\]

Recall the formulation of the kernel \( K_\rho \) of Theorem 1. Then, analogous to (2), we can define a new kernel

\[
K_\sigma(x, y) = \frac{\exp(-\|x - y\|^2/\sigma(x)\sigma(y))}{\sigma(x)\sigma(y)}.
\]

The primary advantage of this variable bandwidth kernel is that, using a \( k \) nearest neighbor approach with \( \sigma \), there is no dependence on the intrinsic dimension of the data in the construction and computation of the graph Laplacian matrix. Empirically, we can show that this kernel at least...
2 CONVERGENCE ANALYSIS

resembles the limiting operator in Theorem 1 (see the next section), though we expect the rate of convergence to differ as the use of $\sigma$ to approximate $\rho$ introduces new bias and variance terms. More work needs to be done to derive this rate of convergence theoretically.
2 CONVERGENCE ANALYSIS

Figure 1: The $k$-nearest neighbor bandwidth function $\sigma$ (b) appears to resemble that of $\rho = p^{-1/d}$ (a), differing by some scale that can be tuned via $\epsilon$, for the $d = 1$ modulus normal distribution with $n = 10000$. In (c), the double-log pointwise convergence plot between $\rho$ and $\sigma$ at $x_0 = 0.1$ shows a convergence rate of approximately $O(n^{-1})$. 
3 Numerical implementation and results

3.1 Algorithm

The implementation of the algorithm keeps to the standard procedure of most spectral graph methods. For the sake of transparency, the specific steps used in the numerical results of this report are outlined below:

1. Given a set of data points \( \{x_i\}_{i=1}^{n} \) and a positive integer \( k \), we compute \( \sigma(x_i) \) for each \( x_i \), the distance from \( x_i \) to its \( k \)th nearest neighbor using the Euclidean distance metric.

2. We construct the affinity matrix \( W \), where \( W_{ij} = K_{\sigma}(x_i, x_j) \) as defined in Definition 4. Note that \( W \) is a symmetric matrix. In practice, for each \( x_i \), the kernel values of \( K_{\sigma}(x_i, \cdot) \) can be approximated as 0 beyond a certain threshold of nearest neighbors of \( x_i \). With this computation optimization, \( W \) is a sparse matrix. Since \( k \)-nearest neighbors is not necessarily a symmetric algorithm, the affinity matrix is symmetrized by \( (W + W^\top)/2 \).

3. From Theorem 1, we are interested in the eigendecomposition of the bandwidth-normalized random walk graph Laplacian matrix \( \frac{1}{\sigma^2} L_{rw} = \Sigma^{-2}(I - D^{-1}W) \), where \( \Sigma \) and \( D \) are diagonal matrices such that \( \Sigma_{ii} = \sigma(x_i) \) and \( D_{ii} = \sum_{j=1}^{n} W_{ij} \). In practice, however, since the bandwidth normalization disrupts the symmetry of the graph Laplacian, we compute the conjugation of this matrix, as in [2]. Let \( S = \Sigma D^{1/2} \). Then, the conjugation is

\[
L_{\text{conj}} = S^{-1}\left(\frac{1}{\sigma^2} L_{rw}\right)S = \Sigma^{-2} - S^{-1}WS.
\]

We symmetrize the conjugation by \( (L_{\text{conj}} + L_{\text{conj}}^\top)/2 \). The rationale of computing this conjugation becomes clear in the next step.

4. We find the eigendecomposition of \( L_{\text{conj}} = V_{\text{conj}}^\top \Lambda_{\text{conj}} V_{\text{conj}} \). It can easily be verified that \( V = S^{-1}V_{\text{conj}} \) and \( \Lambda = \Lambda_{\text{conj}} \), where \( V \) and \( \Lambda \) are respectively the eigenvectors and eigenvalues of \( \frac{1}{\sigma^2} L_{rw} \). Eigendecomposition algorithms can run faster since \( L_{\text{conj}} \) is sparse. The advantage of using the conjugation \( L_{\text{conj}} \) is that it is symmetric, guaranteeing real-valued eigenvalues, and thus numerical spectral convergence, by the spectral theorem.
To verify the convergence of the graph eigenvectors, we approximate the analytic solutions of the eigenvalue problem $\Delta_p f = -\lambda f$ using a finite difference matrix. For a fair comparison between the graph and analytic curves, the analytic eigenfunctions are rescaled by a scalar and phase-shifted (if necessary). Rescaling is valid since any scalar multiple of an eigenfunction is still an eigenfunction associated with the same eigenvalue. This justifies omitting the scalar $m$ (recall from Theorem 1) in the numerical implementation; $\epsilon$ is already implied when we perform bandwidth normalization, via Proposition 2.

### 3.2 Verification and numerical experiments

The algorithm was run on points sampled from $(0, 1) \in \mathbb{R}^1$ and mapped to $\mathcal{M} = S_1$, the unit circle. To introduce a region of sparseness on $S_1$, we sample points from what we will refer to as the “modulus normal distribution,” so named because it is constructed from taking the modulus of the normal distribution so that it is bounded on $(0, 1)$.

![Density](image1.png)  
![Map to S_1](image2.png)

**Figure 2:** Points are sampled from the $d = 1$ modulus normal distribution and mapped to the unit circle. A visible sparseness occurs at $\theta \approx \pi$, which corresponds to the sparse region at around 0.5 in the interval.

**Verification of Theorem 1.** Eigenvectors (second through ninth; the first is trivial and thus excluded) of the graph Laplacian matrix derived from
the $K_\rho$ kernel defined in (2) are computed and compared to the analytic
eigenfunction solutions in Figure 3.

![Figure 3:](image)

(a) Eigenvectors  
(b) Convergence plot

**Figure 3:** In (a), eigenvectors of the graph Laplacian (red) appear to converge
to the analytical eigenfunctions (blue), for $n = 10000$ on the modulus normal
distribution. In (b), the double log convergence plot of $n$ against pointwise error is
very jagged and does not seem to align with the theoretical point wise convergence
of Theorem 1. Slopes above the plots are the empirical convergence rates in the
form of $O(n^{slope})$

Visually, the graph and analytic eigenvectors appear very similar, sug-
gesting convergence to the limiting operator proposed in Theorem ??.

For $d = 1$, we would expect a slope of $\frac{2}{3}$ on the double-log plot based on
the results of Theorem 1. Unfortunately, Figure 3 shows slopes much worse
(that is, more positive) for the convergence of the second through ninth graph
eigenvectors, constructed using $K_\rho$. In addition, the plots are very jagged
and pointwise error does not appear to decrease monotonically with $n$.

If this is the case, this would pose serious obstacles justifying the use of
the $K_\rho$ (and $K_\sigma$) kernel for graph dimension reduction methods. Though
more troubleshooting needs to occur, issues in the empirical convergence
rate could be due to algorithmic errors in the normalization steps performed
before comparing the graph and analytical eigenvectors. Further, we notice
that adjacent pairs of eigenvectors can have the tendency to switch order
(when comparing their eigenvalues) when $\epsilon$ is large. This leads to large
errors in convergence calculations, but relatively harmless sources of error
when related to the theory itself. However, these convergence plots could
also be indications of more underlying sources of error present in the theory. More investigation of this topic is necessary.

**Demonstration of the \( k \)-nearest neighbor kernel.** Again, the eigenvectors of the graph Laplacian are compared to the analytic eigenfunctions in Figure 4, this time using the \( K_\sigma \) kernel defined in (4).

![Figure 4:](image)

(a) Eigenvectors  
(b) Convergence plot

**Figure 4:** In (a), eigenvectors of the graph Laplacian (red) appear to converge to the analytical eigenfunctions (blue), for \( n = 10000, k = 20 \) on the modulus normal distribution. In (b), the double log convergence plot of \( n \) against pointwise error shows that the pointwise error does not seem to converge as \( n \) increases. Slopes above the plots are the empirical convergence rates in the form of \( O(n^{slope}) \)

In Figure 4, we see results similar to Figure 3. Though the graph and analytic eigenvectors appear to converge to the limiting operator for the second through ninth eigenvectors, the empirical convergence rates are highly variable and much slower (if at all) than those of [2] and previous works.

Though we expect the use of the \( k \)-nearest neighbor \( \sigma \) bandwidth function to approximate \( \rho \) will introduce new error terms to our Theorem 1, more theory will need to be developed to derive those terms with more precision.

## 4 Application to MNIST

For a more real-world application, we test this algorithm on the MNIST data set, a database of images of handwritten digits (0-9). This is an example of high dimension data that may have underlying structure. (For example, all
4 APPLICATION TO MNIST

the ‘ones’ have similar features.) To start, we run our using the $K_\sigma$ kernel algorithm to cluster two numbers via diffusion map ([3]). For visualization purposes, diffusion maps use the eigenvectors of the graph Laplacian matrix to project data into two or three dimensions, as in Figure 5.

![Diffusion maps](image)

(a) Two-dimensional diffusion map for one (blue) and five (red).

(b) Three-dimensional diffusion map for zero (blue) and seven (red).

**Figure 5:** Diffusion map techniques using $K_\sigma$ appear to perform well on the MNIST data, to cluster two numbers

Visually, we can see that the diffusion map technique using the $K_\sigma$ kernel is able to separate distinct numbers into relatively distinct clusters in space. Support vector machines (SVM) trained on this projected data classify the clusters with 90-95 percent accuracy, for various combinations of two numbers.

We also offer a comparison between fixed and variable bandwidth algorithms performed on the MNIST data. To compare the effect of the fixed and variable versions of the algorithm on both the local and global geometry, we use all 10 numbers of the MNIST data set. Figure 6 shows the three-dimensional diffusion maps for 500 data points for each number. Note: Some colors are used more than once.

The most noticeable difference is that for the variable bandwidth algorithm each cluster seems to be similar in size and structure, unlike the fixed bandwidth diffusion map. This is expected due to the “self-tuning” nature of the variable bandwidth function. Specifically, we can observe the orange band (on the left side of (a) and towards the center of (b)), corresponding to the number ‘one.’ In the fixed bandwidth diffusion map, the points are
tightly packed together relative to other clusters. However, this is not the case in the variable bandwidth map.

We can see that $\sigma$ is significantly lower for the number 'one' than for other numbers. This means that 'ones' are denser in the original high-dimensional space. This intuitively makes sense because we would not expect handwritten 'ones' to have much variation. Then, the fixed bandwidth algorithm would cluster 'ones' more closely. We cannot objectively say which method is "better." We can say, however, that our variable bandwidth kernel is competitive with well-established fixed bandwidth methods and can curb the trade off between local and global geometry of the data.
Figure 6: In (a) and (b), differences can be seen between the diffusion maps constructed using fixed and variable bandwidth kernels. In (c), the $x$-axis values of 500-1000 correspond to the number ‘one.’
5 Conclusion

We have demonstrated that our kernel has some very desirable advantages. The first, and perhaps most notable, is that using our kernel removes the necessity to estimate the dimension of the manifold in computation, without resorting to the fixed bandwidth case. This is extremely important because it is at the moment still difficult to reliably estimate dimension from given data. Thus, we expect to see results that more consistently converge to the correct operator for more complicated datasets. Secondly, our kernel has a simple and easy to choose tuning parameter, $k$. This is an improvement over previous fixed bandwidth methods because the choice of $k$ is intuitive based on the number of data points. In all of our numerical experiments, we chose $k$ to be slightly less than $1/100th$ of $n$, the number of data points. In the fixed bandwidth case, choosing $\epsilon$ may be hard. It depends on the magnitude of the data, and has ranged in this report from about $10^{-4}$ to $10^1$. Our parameter $k$ is also an improvement over other variable bandwidth methods. Specifically, [2] used four parameters, $\alpha$, $\beta$, $d$, and $\epsilon$. These parameters are also harder to interpret and therefore harder to choose.

We do note, however, that one disadvantage of consolidating parameters is that we diminish the ability to tune our kernel. Another disadvantage is that the convergence on our kernel using both $K_\rho$ and $K_\sigma$ has not been properly numerically verified. More research is necessary to confirm our convergence theory on $K_\rho$ and derive the theoretical convergence of the $K_\sigma$.

References


Appendix A  Proof of limiting operator and bias term

The purpose of this appendix is to rigorously derive the limiting operator in Theorem 1 of the graph Laplacian matrix constructed using the kernel $K_\rho$, as defined in (2), as $\epsilon \to 0$. In addition, the convergence term of the continuous operators (i.e. the bias) will be shown simultaneously. Discussion of the convergence terms when discrete random sampling is introduced can be found in the next appendix.

In [2], Berry & Harlim extended the convergence analysis of fixed bandwidth kernels to variable bandwidth kernels of the form (1). Their proof is built over several steps, starting by first considering “left” and “right formulations” of the symmetric variable bandwidth kernel $K^S_\epsilon$ before combining these results to recover the symmetric kernel. These left and right formulations are defined respectively as

$$K^L_\epsilon (x,y) = h\left( \frac{\|x-y\|^2}{\epsilon \rho(x)} \right) \quad \text{and} \quad K^R_\epsilon (x,y) = h\left( \frac{\|x-y\|^2}{\epsilon \rho(y)} \right).$$

The proof detailed in this appendix adhere to their theoretical approach; however, in many cases, our derivations are actually less complicated than those of [2].

We begin with a crucial result from [3] for the fixed width kernel.

**Lemma 3.** Let $f$ be a smooth, real-valued function on an embedded $d$-dimensional compact manifold $\mathcal{M} \subseteq \mathbb{R}^n$. Suppose $h$ is non-negative and has fast decay. Then,

$$G_\epsilon f(x) = \epsilon^{-d/2} \int_{\mathcal{M}} h\left( \frac{\|x-y\|^2}{\epsilon} \right) f(y) dy$$

$$= m_0 f(x) + \epsilon m_2 (\omega(x) f(x) + \Delta f(x)) + O(\epsilon^2),$$

where $m_0 = \int_{\mathbb{R}^d} h(\|z\|^2) dz$ and $m_2 = \frac{1}{2} \int_{\mathbb{R}^d} z^2 h(\|z\|^2) dz$ are constants determined by $h$ and $\omega$ is determined by the induced geometry of $\mathcal{M}$. The operator $\Delta$ is defined as the Laplace-Beltrami operator on $\mathcal{M}$. 


A PROOF OF LIMITING OPERATOR AND BIAS TERM

A.1 Left formulation for uniformly sampled data
For the left formulation of the variable bandwidth kernel $K_{\epsilon}^L$, define the integral operator

$$G_{\epsilon}^L f(x) = \epsilon^{-d/2} \int_{\mathcal{M}} h\left(\frac{\|x - y\|^2}{\epsilon \rho(x)}\right) f(y)dy.$$  

For now, assume uniform sampling on $\mathcal{M}$. Let $\tilde{\epsilon} = \epsilon \rho(x)$. Substituting,

$$G_{\epsilon}^L f(x) = \rho(x)^{d/2} \tilde{\epsilon}^{-d/2} \int_{\mathcal{M}} h\left(\frac{\|x - y\|^2}{\tilde{\epsilon}}\right) f(y)dy$$

$$= \rho(x)^{d/2} \left[ m_0 f(x) + \tilde{\epsilon} m_2 (\omega(x) f(x) + \Delta f(x)) + \mathcal{O}(\epsilon^2) \right]$$

$$= \rho(x)^{d/2} \left[ m_0 f(x) + \epsilon \rho(x) m_2 (\omega(x) f(x) + \Delta f(x)) \right] + \mathcal{O}(\epsilon^2), \quad (5)$$

by substituting the result of Lemma 3.

A.2 Right formulation for uniformly sampled data
Continuing to assume uniform sampling, the left formulation will be necessary to analyze the right formulation $K_{\epsilon}^R$ of the kernel.

In this section, we define an equivalence relation $W$ for two functions $f$ and $g$, such that $f \equiv_W g$ if $\forall h$

$$\langle f, h \rangle = \langle g, h \rangle,$$

for the inner product $\langle f, h \rangle = \int f(x)h(x) \, dx$. We say $f$ and $g$ are equal in the weak sense.

To expand

$$G_{\epsilon}^R f(x) = \epsilon^{-d/2} \int_{\mathcal{M}} h\left(\frac{\|x - y\|^2}{\epsilon \rho(y)}\right) f(y)dy,$$
we first consider, for an arbitrary smooth function $g$,

$$
\langle g, G^R_\epsilon f \rangle = \int_M g(x) \left[ \epsilon^{-d/2} \int_M h \left( \frac{\|x-y\|^2}{\epsilon \rho(y)} \right) f(y) dy \right] dx \\
= \int_M f(y) \left[ \epsilon^{-d/2} \int_M h \left( \frac{\|x-y\|^2}{\epsilon \rho(y)} \right) g(x) dx \right] dy \\
= \int_M f(y) \left[ \rho(y)^{d/2} \left[ m_0 g(y) + \epsilon \rho(y)m_2 (\omega(y)g(y) + \Delta g(y)) \right] + \mathcal{O}(\epsilon^2) \right] dy \\
= \langle g, m_0 \rho^{d/2} f + \epsilon m_2 \rho^{d+1} \omega f + \epsilon m_2 \Delta (\rho^{d+1} f) + \mathcal{O}(\epsilon^2) \rangle
$$

The first equality applies the definition of the inner product. The second equality changes the order of integration; we can do this because $h$ decays quickly. The third equality substitutes the results of the left formulation (5). The fourth equality splits the integral term-wise. The fifth equality sums the inner products and uses the fact that the Laplacian operator is self-adjoint.

Thus, we can say that

$$G^R_\epsilon f \equiv_W m_0 \rho^{d/2} f + \epsilon m_2 \rho^{d+1} \omega f + \epsilon m_2 \Delta (\rho^{d+1} f) + \mathcal{O}(\epsilon^2).$$

This weak equivalence is strong enough for our purposes.

Expanding $\Delta (\rho^{d+1} f)$,

$$\Delta (\rho^{d+1} f) = \Delta (\rho^d f) + \rho^d \Delta f + 2 \nabla (\rho^{d+1}) \nabla f$$

$$= \Delta (\rho^d f) + \rho^d \Delta f + 2 \left( \frac{d}{2} + 1 \right) \rho^{d/2} \nabla \rho \nabla f.$$

Substituting,

$$G^R_\epsilon f = m_0 \rho^{d/2} f + \epsilon m_2 \rho^{d+1} \omega f + \epsilon m_2 \left( \Delta (\rho^d f) + \rho^d \Delta f + (d+2) \rho^{d/2} \nabla \rho \nabla f \right) + \mathcal{O}(\epsilon^2)$$

$$= \rho^{d/2} \left[ m_0 f + \epsilon m_2 \rho \left( \tilde{\omega} f + \Delta f + (d+2) \nabla \rho \nabla f \right) \right] + \mathcal{O}(\epsilon^2), \quad (6)$$

where $\tilde{\omega} = \omega + \rho^{-3d/2} \Delta (\rho^{3d/2})$. 

19
A PROOF OF LIMITING OPERATOR AND BIAS TERM

A.3 Symmetric bandwidth for uniformly sampled data

Combining the results of the left and right formulations, we define

\[ G^S_\epsilon f(x) = \epsilon^{-d/2} \int_M h \left( \frac{\|x - y\|^2}{\epsilon \rho(x) \rho(y)} \right) f(y) dy. \]

Again, letting \( \tilde{\epsilon} = \epsilon \rho(x) \), we substitute,

\[ G^S_\epsilon f(x) = \rho \left[ m_0 f + \tilde{\epsilon} m_2 \rho \left( \tilde{\omega} f + \Delta f + (d + 2) \frac{\nabla \rho}{\rho} \nabla f \right) + O(\tilde{\epsilon}^2) \right] \]

\[ = \rho \left[ m_0 f + \epsilon m_2 \rho \left( \omega f + \Delta f + (d + 2) \frac{\nabla \rho}{\rho} \nabla f \right) \right] + O(\epsilon^2), \quad (7) \]

substituting the result from (6).

A.4 Generalization to non-uniformly sampled data

Generalizing the previous result to non-uniform sampling, the symmetric operator is now

\[ G^S_\epsilon f(x) = \rho^{d/2} \epsilon^{-d/2} \int_M h \left( \frac{\|x - y\|^2}{\epsilon \rho(x) \rho(y)} \right) f(y) p(y) dy. \]

We can substitute \( f(y) p(y) \) into the result of (7), giving

\[ G^S_\epsilon f = \rho \left[ m_0 f p + \epsilon m_2 \rho \left( \omega f p + \Delta f p + (d + 2) \frac{\nabla \rho}{\rho} \nabla (f p) \right) \right] + O(\epsilon^2) \]

\[ = \rho \left[ m_0 f p + \epsilon m_2 \rho \left( \omega f p + \Delta f p + 2 \nabla f \nabla p + (d + 2) \frac{\nabla \rho}{\rho} \left( f \nabla p + p \nabla f \right) \right) \right] + O(\epsilon^2) \]

\[ = \rho \left[ (m_0 p + O(\epsilon)) f + \epsilon m_2 \rho \left( \Delta f + 2 \frac{\nabla p}{\rho} \nabla f + (d + 2) \frac{\nabla \rho}{\rho} \nabla f \right) \right] + O(\epsilon^2). \quad (8) \]

Equation (8) is derived by combining all the terms containing \( f \) only. This will be useful later.
A PROOF OF LIMITING OPERATOR AND BIAS TERM

A.5 Application to the $K_{\rho}$ kernel

Finally, we consider the integral operator derived from the $K_{\rho}$ kernel, defined as

$$G^\rho \epsilon f(x) = \epsilon^{-d/2} \int_M h \left( \frac{\|x - y\|^2}{\epsilon \rho(x) \rho(y)} \right) \frac{1}{\rho(x) \rho(y)} f(y) p(y) dy$$

$$= \frac{\epsilon^{-d/2}}{\rho(x)} \int_M h \left( \frac{\|x - y\|^2}{\epsilon \rho(x) \rho(y)} \right) \frac{f(y)}{\rho(y)} p(y) dy.$$

Here, $h(z) = \exp(-z)$, which satisfies the necessary conditions for $h$ in Lemma 3. Substituting in (8) and replacing $f$ with $f/\rho$,

$$G^\rho \epsilon f = \rho^d \rho^{-1} [(m_0 p + O(\epsilon)) \left( \frac{f}{\rho} \right) +$$

$$\epsilon m_2 \rho^2 p \left( \Delta \left( \frac{f}{\rho} \right) + 2 \frac{\nabla p}{p} \nabla \left( \frac{f}{\rho} \right) + (d + 2) \frac{\nabla \rho}{\rho} \nabla \left( \frac{f}{\rho} \right) \right)] + O(\epsilon^2).$$

$$= \rho^{d-1} [(m_0 p + O(\epsilon)) \left( \frac{f}{\rho} \right) +$$

$$\epsilon m_2 \rho^2 p \left( \Delta f - 2 \frac{\nabla \rho}{\rho} \nabla f + 2 \frac{\nabla p}{p} \nabla f + (d + 2) \frac{\nabla \rho}{\rho} \nabla f \right)] + O(\epsilon^2).$$

$$= \rho^{d-2} p [(m_0 + O(\epsilon)) f +$$

$$\epsilon m_2 \rho^2 \left( \Delta f + 2 \frac{\nabla p}{p} \nabla f + d \frac{\nabla \rho}{\rho} \nabla f \right)] + O(\epsilon^2),$$

by again combining all the terms containing $f$ only.

Recall that in our kernel, $\rho = p^{-1/d}$. Thus,

$$\frac{\nabla \rho}{\rho} = \frac{\nabla (p^{-1/d})}{p^{-1/d}} = \left( -\frac{1}{d} \right) p^{-\frac{1}{d}-1} \frac{\nabla p}{p^{-1/d}} = \left( -\frac{1}{d} \right) \frac{\nabla p}{p}.$$

Thus,

$$G^\rho \epsilon f = \rho^{d-2} p [(m_0 + O(\epsilon)) f +$$

$$\epsilon m_2 \rho^2 \left( \Delta f + 2 \frac{\nabla p}{p} \nabla f + \left( -\frac{1}{d} \right) \frac{\nabla p}{p} \nabla f \right)] + O(\epsilon^2),$$

$$= \rho^{d-2} p \left[ (m_0 + O(\epsilon)) f + \epsilon m_2 \rho^2 \left( \Delta f + \frac{\nabla p}{p} \nabla f \right) \right] + O(\epsilon^2). \quad (9)$$
A.6 Derivation of limiting operator and bias term

At last, we define the continuous operator

$$L^\rho f(x) = \frac{1}{\epsilon m\rho(x)^2} \left( \frac{G^K_\epsilon f(x)}{G^K_\epsilon 1(x)} - f(x) \right),$$

where $m = m_2/m_0$.

From (9) we have

$$G^\rho f = \rho^{d-2} \left[ (m_0 + C_\epsilon) f + \epsilon m_2 \rho^2 \left( \Delta f + \frac{\nabla p}{p} \nabla f \right) \right] + O(\epsilon^2)$$

$$G^\rho 1 = \rho^{d-2} \left[ (m_0 + C_\epsilon) \right] + O(\epsilon^2),$$

where $C_\epsilon$ is simply shorthand for a collection of known terms all of the order $O(\epsilon)$.

Substituting gives

$$L^\rho f = \frac{1}{\epsilon m\rho^2} \left( \rho^{d-2} \left[ (m_0 + C_\epsilon) f + \epsilon m_2 \rho^2 \left( \Delta f + \frac{\nabla p}{p} \nabla f \right) \right] + O(\epsilon^2) \right) - f$$

$$= \frac{1}{\epsilon m\rho^2} \left( \epsilon m_2 \rho^2 \left( \Delta f + \frac{\nabla p}{p} \nabla f \right) + O(\epsilon^2) \right)$$

$$= \Delta f + \frac{\nabla p}{p} \nabla f + O(\epsilon).$$

The terms $m_0$ and $C_\epsilon$ cancel exactly in the subtraction of $f$. Therefore, the limiting operator and bias term given in (11) is as stated in Theorem 1.

Appendix B Proof of the variance term

We now consider the discrete operator

$$L^\rho_{n,\epsilon} f(x_i) = \frac{1}{\epsilon m\rho^2} \left( \sum_{j \neq i} K(x_i, x_j) f(x_j) \frac{\sum_{j \neq i} K(x_i, x_j)}{\sum_{j \neq i} K(x_i, x_j)} - f(x_i) \right).$$
B PROOF OF THE VARIANCE TERM

B.1 Bounding the variance

We explicitly separate the bias from variance via

$$\frac{1}{\epsilon m \rho^2} \left( \sum_{j \neq i} K(x_i, x_j) f(x_j) - \frac{\mathbb{E}(K(x_i, \cdot) f(\cdot))}{\mathbb{E}(K(x_i, \cdot))} + \frac{\mathbb{E}(K(x_i, \cdot) f(\cdot))}{\mathbb{E}(K(x_i, \cdot))} - f(x_i) \right).$$

The bias was already considered in Appendix A. Thus, the expression of interest for this section is

$$\frac{1}{\epsilon m \rho^2} \left( \sum_{j \neq i} K(x_i, x_j) f(x_j) - \frac{\mathbb{E}(K(x_i, \cdot) f(\cdot))}{\mathbb{E}(K(x_i, \cdot))} \right)$$

Instead of calculating the expectation directly, we try to find the probability that this term exceeds $\alpha$, which is equivalent to the quantity

$$\mathbb{P} \left( \left( \sum_{j \neq i} K(x_i, x_j) f(x_j) - \frac{\mathbb{E}(K(x_i, \cdot) f(\cdot))}{\mathbb{E}(K(x_i, \cdot))} \right) > \alpha \epsilon m \rho^2 \right)$$

Defining $F(\cdot) = K(x_i, \cdot) f(\cdot)$, and $G(\cdot) = K(x_i, \cdot)$, we can simply write this quantity as

$$\mathbb{P} \left( \mathbb{E}(G) \sum_{j \neq i} F(x_j) - \mathbb{E}(F) \sum_{j \neq i} G(x_j) > \alpha \epsilon m \rho^2 \mathbb{E}(G) \sum_{j \neq i} G(x_j) \right)$$

Now, we define $Y(x_j) = \mathbb{E}(G) F(x_j) - \mathbb{E}(F) G(x_j) + \alpha \epsilon m \rho^2 \mathbb{E}(G)(\mathbb{E}(G) - G(x_j))$, obtaining the form

$$\mathbb{P} \left( \sum_{j \neq i} Y(x_j) > (N - 1) \alpha \epsilon m \rho^2 (\mathbb{E}(G))^2 \right)$$

We want this probability to be small. Since $Y(x_j)$ is i.i.d. and $\mathbb{E}(Y) = 0$, we can use central limit theorem to state that

$$\mathbb{P} \left( \sum_{j \neq i} Y(x_j) > 4\sqrt{N - 1} \sqrt{\text{Var}(Y)} \right)$$
B PROOF OF THE VARIANCE TERM

is small. Hence by equating these two right-hand sides, we get the error bound

$$\alpha = \frac{4\sqrt{Var(Y)}}{\sqrt{N - 1} \epsilon m \rho^2 (E(G))^2}$$

that will be violated only with low probability.

B.2 Variance of $Y$

Since $E(Y) = 0$, the variance of $Y$ is $E(Y^2)$. We first calculate

$$Y(x_j)^2 = E(G)^2 F(x_j)^2 + E(F)^2 G(x_j)^2 - 2 E(F) E(G) F(x_j) G(x_j) + \alpha \epsilon m \rho^2 E(G) \left[ E(G)^2 F(x_j) - E(G) F(x_j) G(x_j) - E(F) E(G) G(x_j) + E(F) G(x_j)^2 \right] + O(\alpha^2 \epsilon^2)$$

and then

$$E(Y^2) = E(G)^2 E(F^2) + E(F)^2 E(G^2) - 2 E(F) E(G) E(FG) + \alpha \epsilon m \rho^2 E(G) \left[ E(G)^2 E(F) - E(G) E(FG) - E(F) E(G)^2 \right] + O(\alpha^2 \epsilon^2)$$

$$= E(G)^2 E(F^2) + E(F)^2 E(G^2) - 2 E(F) E(G) E(FG) + \alpha \epsilon m \rho^2 E(G) \left[ E(F) E(G^2) - E(G) E(FG) \right] + O(\alpha^2 \epsilon^2)$$

(12)

So we see that we need $E(F), E(G), E(F^2), E(G^2), E(FG)$. From appendix A, we have

$$G^K \epsilon f = \epsilon^{-d/2} \int_{\mathcal{M}} h \left( \frac{\|x-y\|^2}{\epsilon \rho(x) \rho(y)} \right) f(y) p(y) dy$$

$$= m_0 \rho^{d-2} p(f + \epsilon m \rho^2 (f \bar{w} + \Delta_p f)) + O(\epsilon^2)$$
Using this operator, we calculate these five quantities.

\[ \mathbb{E}(F) = \epsilon^{d/2} G^K f = \epsilon^{d/2} m_0 \rho^{d-2} p (f + \epsilon p \rho^2 (f \bar{w} + \Delta p f)) + \mathcal{O}(\epsilon^{d+2}) \]
\[ \mathbb{E}(G) = \epsilon^{d/2} G^K 1 = \epsilon^{d/2} m_0 \rho^{d-2} p (1 + \epsilon p \rho^2 \bar{w}) + \mathcal{O}(\epsilon^{d+2}) \]
\[ \mathbb{E}(F^2) = (\epsilon/2)^{d/2} \rho^{-1} G_{\epsilon/2}^{K} \frac{f^2}{\rho} \]
\[ = \epsilon^{d/2} m_0 \rho^{d-3} p (\frac{f^2}{\rho}) + \frac{\epsilon}{2} m \rho^2 (\frac{f^2}{\rho} \bar{w} + \Delta p (\frac{f^2}{\rho})) + \mathcal{O}(\epsilon^{d+2}) \]
\[ \mathbb{E}(G^2) = (\epsilon/2)^{d/2} \rho^{-1} G_{\epsilon/2}^{K} 1 \]
\[ = \epsilon^{d/2} m_0 \rho^{d-3} p (\frac{1}{\rho}) + \frac{\epsilon}{2} m \rho^2 (\frac{1}{\rho} \bar{w} + \Delta p (\frac{1}{\rho})) + \mathcal{O}(\epsilon^{d+2}) \]
\[ \mathbb{E}(FG) = (\epsilon/2)^{d/2} \rho^{-1} G_{\epsilon/2}^{K} \frac{f}{\rho} \]
\[ = \epsilon^{d/2} m_0 \rho^{d-3} p (\frac{f}{\rho}) + \frac{\epsilon}{2} m \rho^2 (\frac{f}{\rho} \bar{w} + \Delta p (\frac{f}{\rho})) + \mathcal{O}(\epsilon^{d+2}) \]

Then, representing each quantity in the form \( a_i \epsilon^{d/2} + b_i \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2}) \)
indexed respect to the order of quantities above,

\[ a_1 = m_0 \rho^{d-2} p f \quad b_1 = m_2 \rho^d p (f \bar{w} + \Delta p f) \]
\[ a_2 = m_0 \rho^{d-2} p \quad b_2 = m_2 \rho^d \bar{w} \]
\[ a_3 = m_0 \rho^{d-4} 2^{-d/2} p f^2 \quad b_3 = m_2 \rho^{d-1} 2^{-d/2-1} p (\bar{w} + \Delta p) \frac{f^2}{\rho} \]
\[ a_4 = m_0 \rho^{d-4} 2^{-d/2} p \quad b_4 = m_2 \rho^{d-1} 2^{-d/2-1} p (\bar{w} + \Delta p) \frac{1}{\rho} \]
\[ a_5 = m_0 \rho^{d-4} 2^{-d/2} p f \quad b_5 = m_2 \rho^{d-1} 2^{-d/2-1} p (\bar{w} + \Delta p) \frac{f}{\rho} \]

We note here that \( a_1 a_5 = a_2 a_3 \) and \( a_1 a_4 = a_2 a_5 \).

Now we calculate (12), starting with the fourth term, we calculate the quantity inside the bracket.
\[ \mathbb{E}(F)\mathbb{E}(G^2) - \mathbb{E}(G)\mathbb{E}(FG) = \\
(a_1 \epsilon^{d/2} + b_1 \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2}))(a_4 \epsilon^{d/2} + b_4 \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2})) - \\
(a_2 \epsilon^{d/2} + b_2 \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2}))(a_5 \epsilon^{d/2} + b_5 \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2})) \]

Collecting the \( \epsilon^d \) terms, we have \( a_1 a_4 - a_2 a_5 \) which is 0. Noting that \( \mathbb{E}(G) \) is of \( \mathcal{O}(\epsilon^{d/2}) \), the fourth term can be bounded by

\[ \alpha c m \rho^2 \mathbb{E}(G) \left[ \mathbb{E}(F)\mathbb{E}(G^2) - \mathbb{E}(G)\mathbb{E}(FG) \right] = \mathcal{O}(\alpha \epsilon^{3d/2+2}) \]

We now compute the rest of (12).

\[ \mathbb{E}(G)^2 \mathbb{E}(F^2) + \mathbb{E}(F)^2 \mathbb{E}(G^2) - 2\mathbb{E}(F)\mathbb{E}(G)\mathbb{E}(FG) = \\
(a_2 \epsilon^{d/2} + b_2 \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2}))(a_3 \epsilon^{d/2} + b_3 \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2})) + \\
(a_1 \epsilon^{d/2} + b_1 \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2}))(a_4 \epsilon^{d/2} + b_4 \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2})) - \\
2(a_1 \epsilon^{d/2} + b_1 \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2}))(a_2 \epsilon^{d/2} + b_2 \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2}))(a_3 \epsilon^{d/2} + b_3 \epsilon^{d/2+1} + \mathcal{O}(\epsilon^{d/2+2})) \]

Collecting the \( \epsilon^{3d/2} \) terms, we have \( a_2^2 a_3 + a_1^2 a_4 - 2a_1 a_2 a_5 = 0. \) For \( \epsilon^{3d/2+1} \) terms, we have \( a_2^2 a_3 + 2a_2 a_3 b_2 + a_1^2 b_4 + 2a_1 a_4 b_1 - 2(a_1 a_2 b_3 + a_1 a_3 b_2 + a_2 a_3 b_1) = \\
a_2^2 b_3 + a_1^2 b_4 - 2a_1 a_2 b_5 \) from which we get

\[ (m_0 \rho^{d-2} p)^2 m_2 \rho^{d-1} 2^{-d/2-1} p(\bar{w} + \Delta_p)\left(\frac{f^2}{\rho}\right) + \\
(m_0 \rho^{d-2} p)^2 f^2 m_2 \rho^{d-1} 2^{-d/2-1} p(\bar{w} + \Delta_p)\left(\frac{1}{\rho}\right) - \\
2(m_0 \rho^{d-2} p)^2 f m_2 \rho^{d-1} 2^{-d/2-1} p(\bar{w} + \Delta_p)\left(\frac{f}{\rho}\right) \]

Gathering the terms and canceling \( \bar{w} \) terms, we end up with

\[ m_0^2 m_2 2^{-d/2-1} \rho^{3d-5} p^3 [\Delta_p(f^2 \rho^{-1}) + f^2 \Delta_p(\rho^{-1}) - 2f \Delta_p(f \rho^{-1})] \]

Recalling that \( \Delta_p = \Delta + \frac{\nabla p}{p} \cdot \nabla \), the expression inside the bracket can be simplified to \( 2\|\nabla f\|^2 \rho^{-1} \). Hence, assuming \( \alpha \) is small and noting that \( p = \rho^{-d} \), we can finally express \( \mathbb{E}(Y^2) \) as
We can now compute the error bound $\alpha$.

$$\alpha = \frac{4 \sqrt{\text{Var}(Y)}}{\sqrt{N - 1} \epsilon m^2} = \frac{4 m_0 \sqrt{m_2} 2^{-d/4} \rho^{-3} \|\nabla f\| \epsilon^{3d/4 + 1/2} + O(\epsilon^{3d/2 + 2})}{\sqrt{N - 1} \epsilon m^2 (E(G))^2}$$

Define $C = m_2^{-1/2} 2^{-d/4 - 2}$, the error bound is approximately (up to $O(\epsilon^2)$)

$$\frac{C \|\nabla f\| \epsilon^{-d/4 - 1/2}}{\sqrt{N} p^{-1/d}}$$