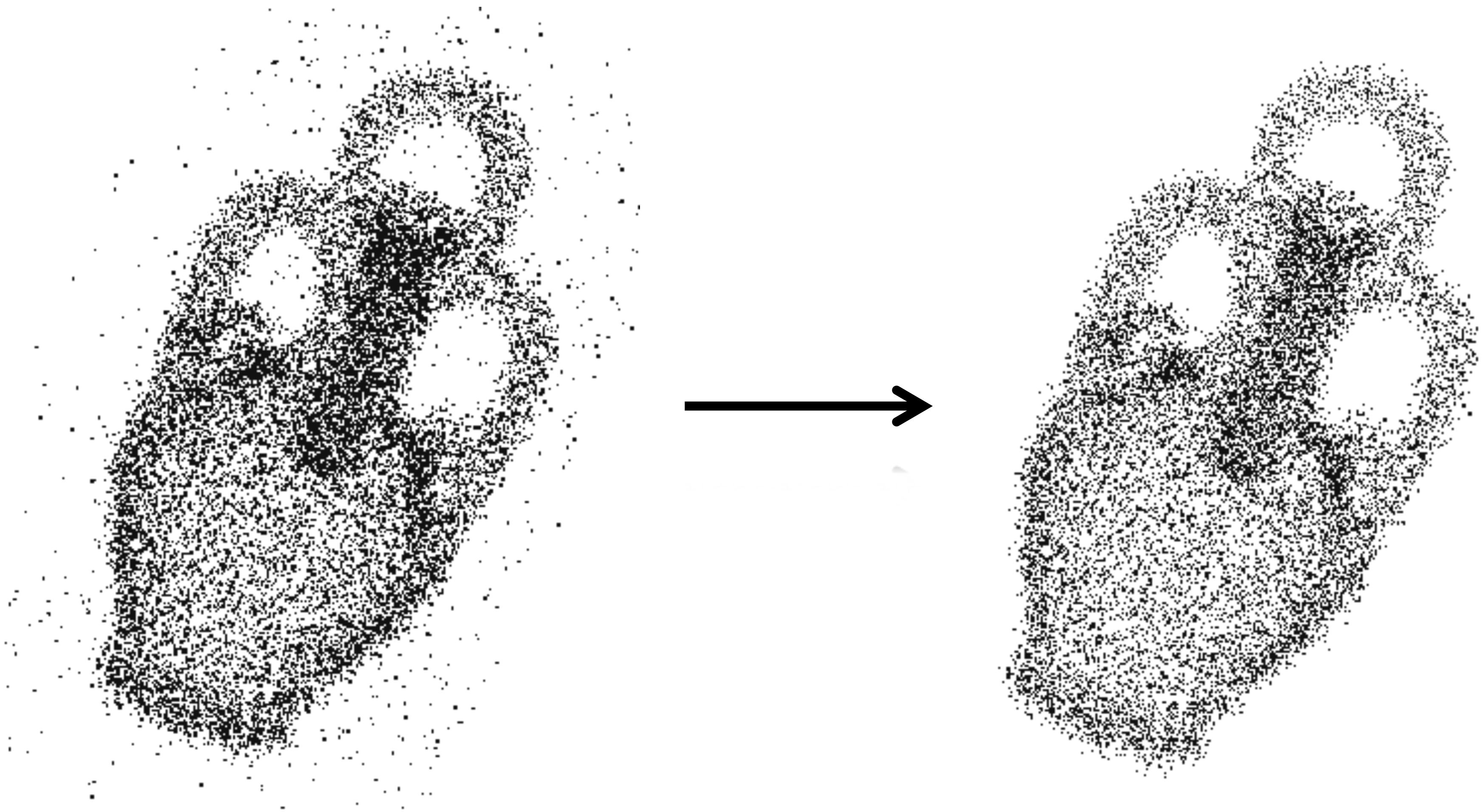
Declutter and resample: Towards parameter free denoising.

Jiayuan Wang
Joint work with:
Mickaël Buchet
Tamal K. Dey
Yusu Wang

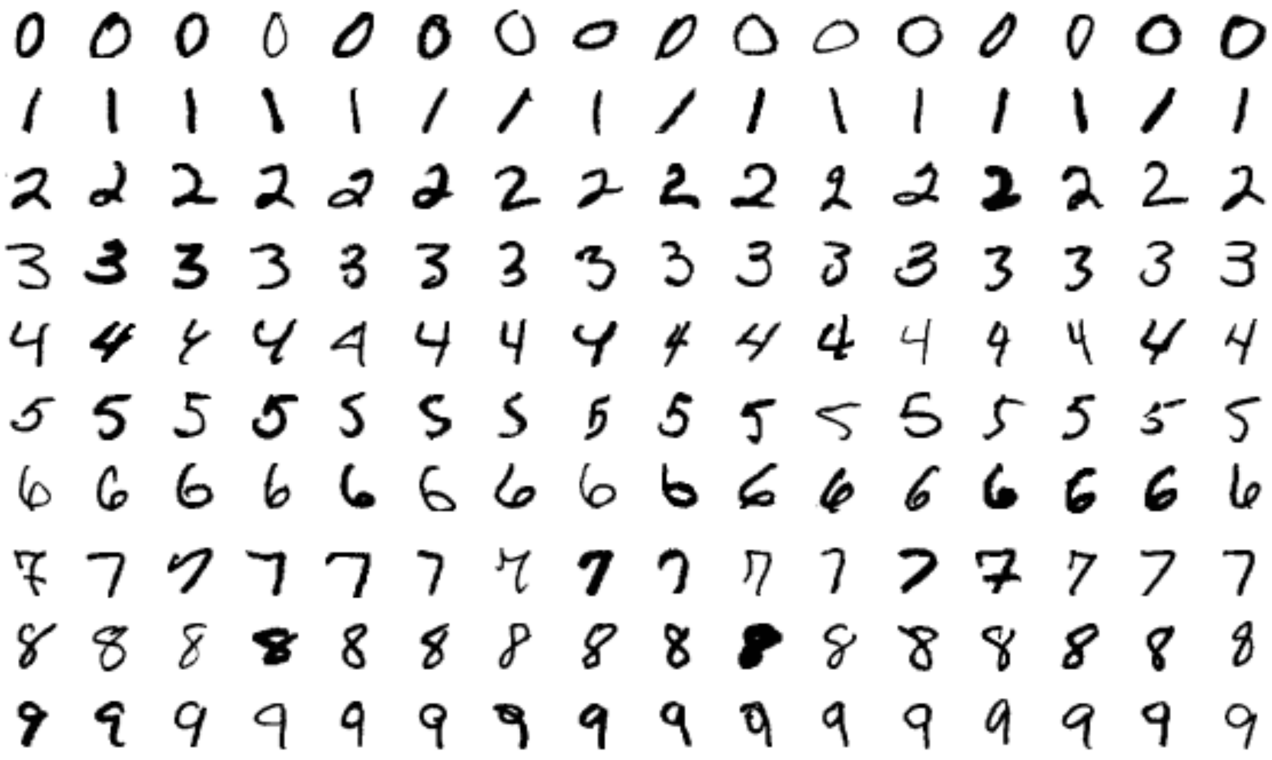
Outline-Parameter-free denoising algorithm

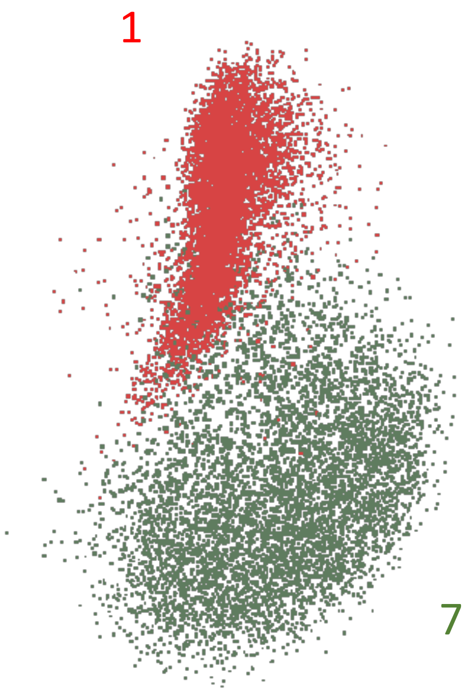
- Parameter-free denoising algorithm
  - Introduction
  - Preliminaries
  - Declutter algorithm
  - Parameter-free algorithm
  - Discussions

Introduction



Introduction





Introduction

- Deconvolution - noise model/parameter
- Thresholding - parameter

Outline-Parameter-free denoising algorithm

- Parameter-free denoising algorithm
  - Introduction
  - Preliminaries
  - Declutter algorithm
  - Parameter-free algorithm
  - Discussions

Preliminaries – sampling condition

- Metric space $(\mathbb{X}, d_{\mathbb{X}})$
- k-distance to a point set P : $d_{P,k}(x) = \sqrt{\frac{1}{k}\sum_{i=1}^{k} d_{\mathbb{X}}\big(x, p_i(x)\big)^2}$
- P is an ϵ_k-noisy sample of K if:
  - $\forall x \in K, d_{P,k}(x) \leq \epsilon_k$
  - $\forall x \in \mathbb{X}, d_{\mathbb{X}}(x, K) \leq d_{P,k}(x) + \epsilon_k$

Preliminaries – adaptive sampling condition

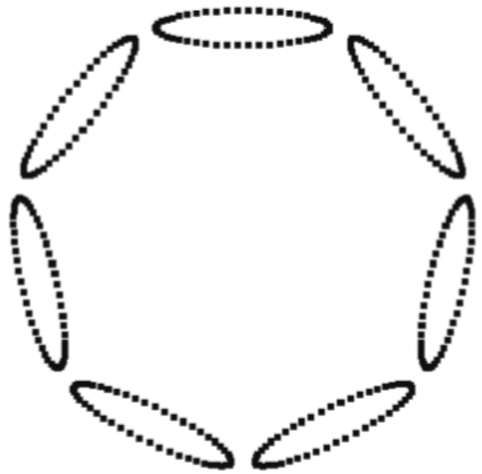
- Metric space $(\mathbb{X}, d_{\mathbb{X}})$
- k-distance to a point set P : $d_{P,k}(x) = \sqrt{\frac{1}{k}\sum_{i=1}^{k} d_{\mathbb{X}}\big(x, p_i(x)\big)^2}$
- P is an ϵ_k-adaptive noisy sample of K if:
  - $\forall x \in K, d_{P,k}(x) \leq \epsilon_k f(x)$
  - $\forall y \in \mathbb{X}, d_{\mathbb{X}}(y, K) \leq d_{P,k}(y) + \epsilon_k f(\bar{y})$

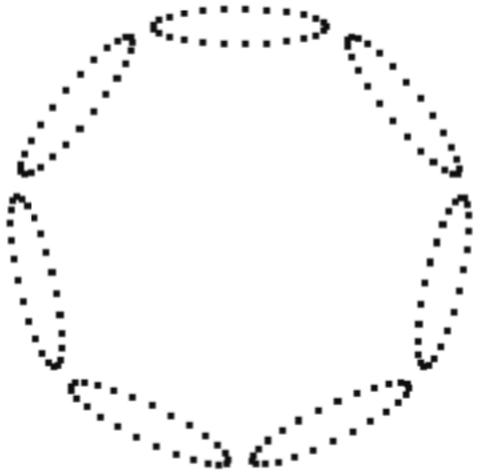
Outline-Parameter-free denoising algorithm

- Parameter-free denoising algorithm
  - Introduction
  - Sampling condition
  - Declutter algorithm
  - Parameter-free algorithm
  - Discussions

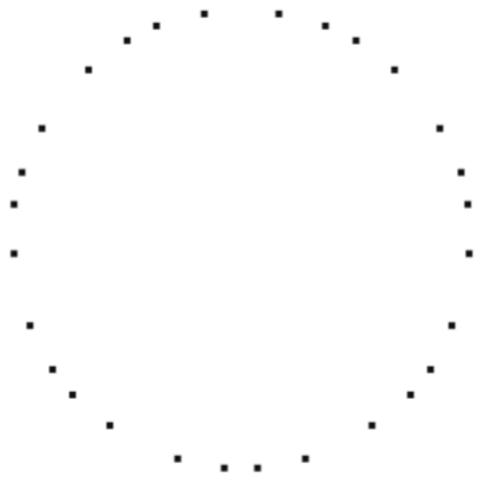
Declutter algorithm

- One parameter is needed





k=2



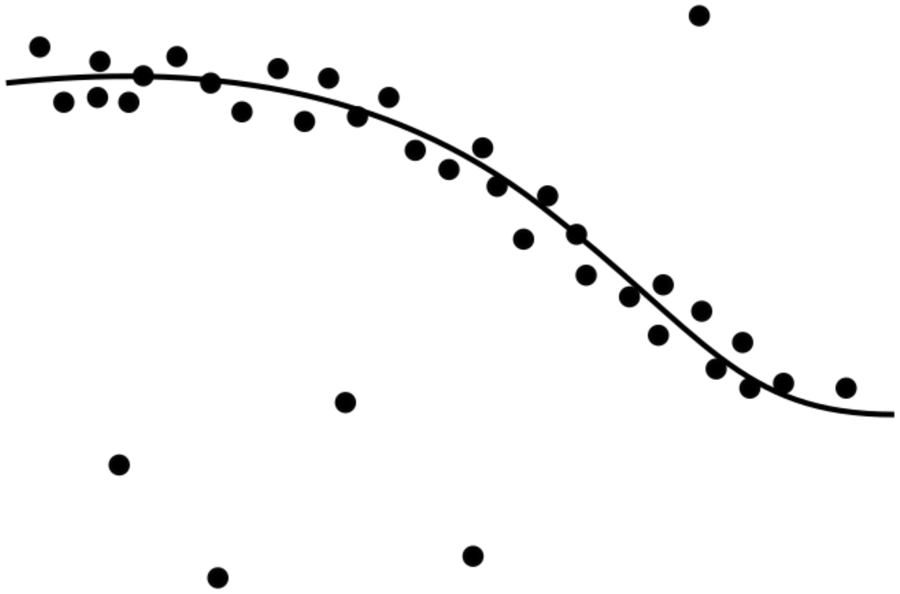
k=10

Declutter algorithm

Algorithm 1: Declutter(P,k)

Data: Point set P, parameter k
Result: Denoised point set Q

```
begin
    sort P such that d_{P,k}(p_1) ≤ ··· ≤ d_{P,k}(p_{|P|}).
    Q_0 ⟵ ∅
    for i ⟵ 1 to |P| do
        if Q_{i-1} ∩ B(p_i, 2d_{P,k}(p_i)) = ∅ then
            Q_i = Q_{i-1} ∪ {p_i}
        else  Q_i = Q_{i-1}
    Q ⟵ Q_n
```



Declutter algorithm

Algorithm 1: Declutter(P,k)

Data: Point set P, parameter k

Result: Denoised point set Q

begin

  sort P such that $d_{P,k}(p_1) \leq \cdots \leq d_{P,k}(p_{|P|})$.

  $Q_0 \longleftarrow \emptyset$

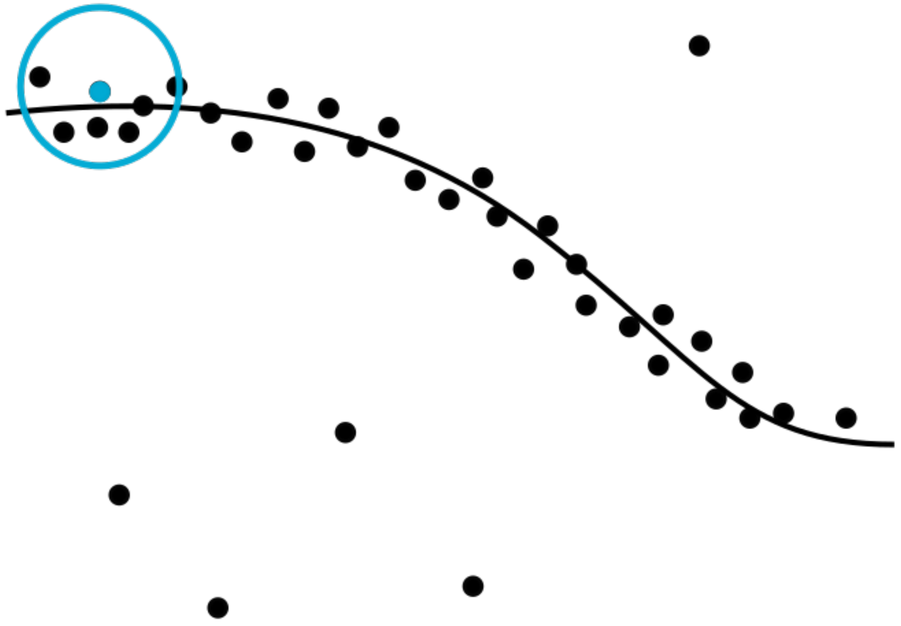
  for $i \longleftarrow 1$ *to* $|P|$ **do**

    if $Q_{i-1} \cap B(p_i, 2d_{P,k}(p_i)) = \emptyset$ **then**

      $Q_i = Q_{i-1} \cup \{p_i\}$

    else $Q_i = Q_{i-1}$

  $Q \longleftarrow Q_n$

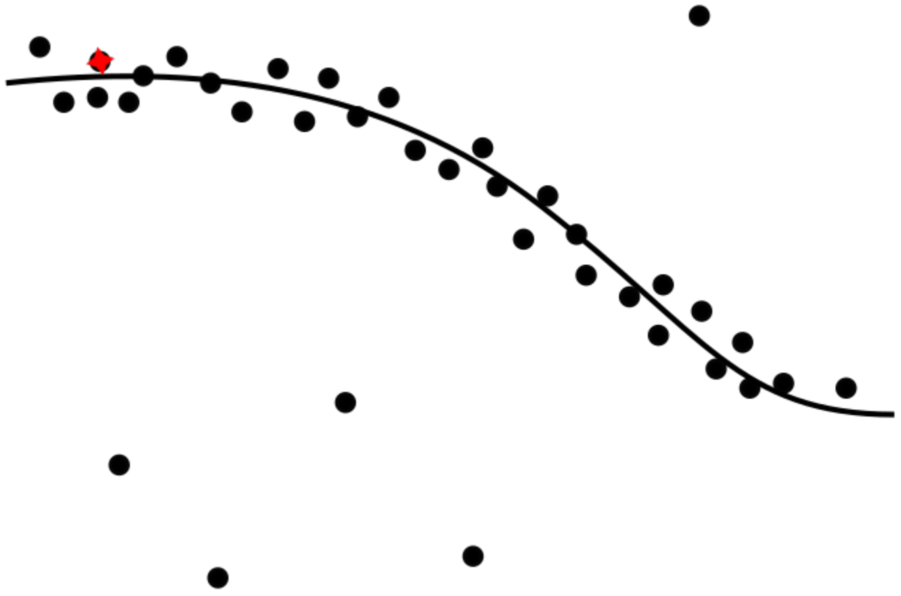


Declutter algorithm

Algorithm 1: Declutter(P,k)

Data: Point set P, parameter k
Result: Denoised point set Q

```
begin
    sort P such that d_{P,k}(p_1) ≤ ··· ≤ d_{P,k}(p_{|P|}).
    Q_0 ⟵ ∅
    for i ⟵ 1 to |P| do
        if Q_{i−1} ∩ B(p_i, 2d_{P,k}(p_i)) = ∅ then
            Q_i = Q_{i−1} ∪ {p_i}
        else  Q_i = Q_{i−1}
    Q ⟵ Q_n
```

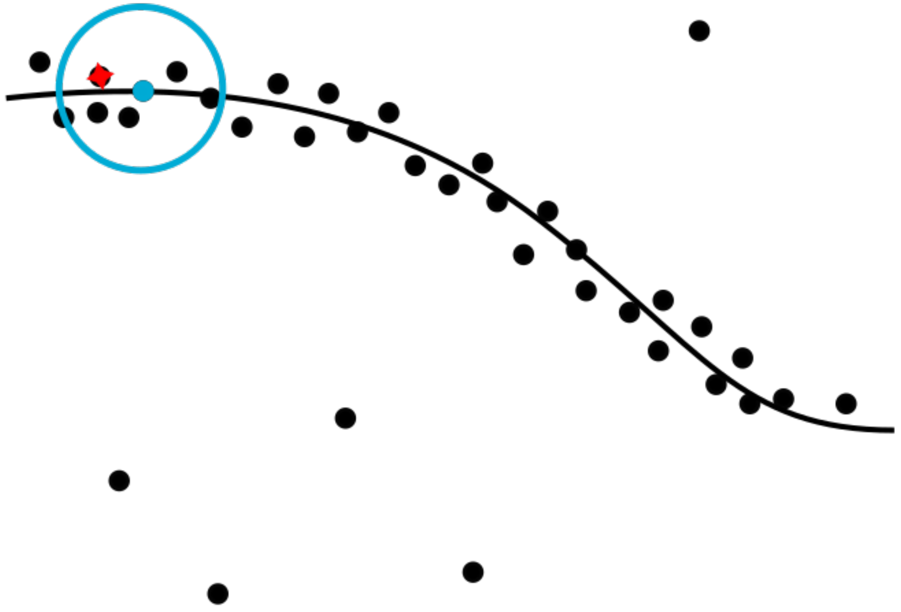


Declutter algorithm

Algorithm 1: Declutter(P,k)

Data: Point set P, parameter k
Result: Denoised point set Q

begin
  sort P such that $d_{P,k}(p_1) \leq \cdots \leq d_{P,k}(p_{|P|})$.
  $Q_0 \longleftarrow \emptyset$
  for $i \longleftarrow 1$ *to* $|P|$ **do**
    if $Q_{i-1} \cap B(p_i, 2d_{P,k}(p_i)) = \emptyset$ **then**
      $Q_i = Q_{i-1} \cup \{p_i\}$
    else $Q_i = Q_{i-1}$
  $Q \longleftarrow Q_n$



Declutter algorithm

Algorithm 1: Declutter(P,k)

Data: Point set P, parameter k

Result: Denoised point set Q

begin

  sort P such that $d_{P,k}(p_1) \leq \cdots \leq d_{P,k}(p_{|P|})$.

  $Q_0 \longleftarrow \emptyset$

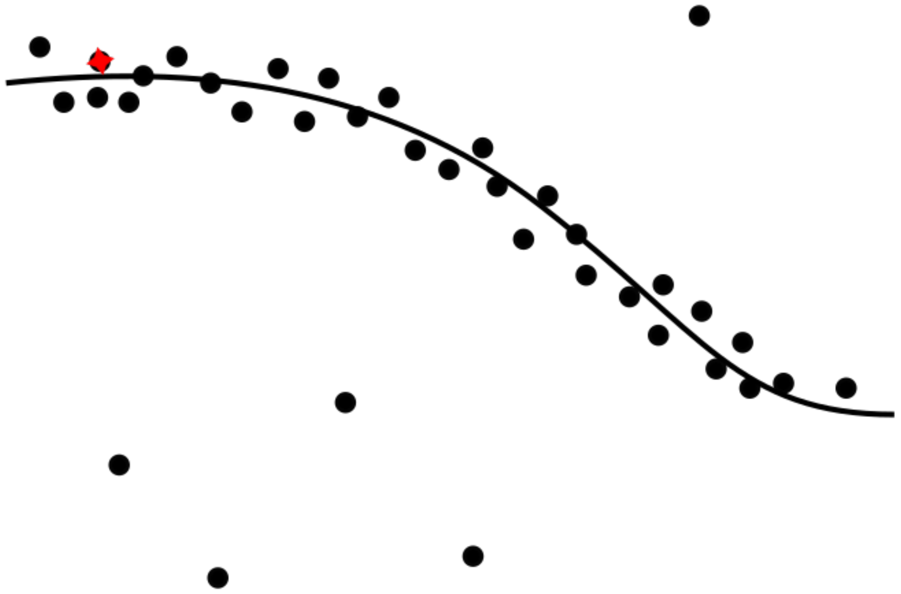
  for $i \longleftarrow 1$ *to* $|P|$ **do**

    if $Q_{i-1} \cap B(p_i, 2d_{P,k}(p_i)) = \emptyset$ **then**

      $Q_i = Q_{i-1} \cup \{p_i\}$

    else $Q_i = Q_{i-1}$

  $Q \longleftarrow Q_n$



Declutter algorithm

Algorithm 1: Declutter(P,k)

Data: Point set P, parameter k

Result: Denoised point set Q

begin

 sort P such that $d_{P,k}(p_1) \leq \cdots \leq d_{P,k}(p_{|P|})$.

 $Q_0 \longleftarrow \emptyset$

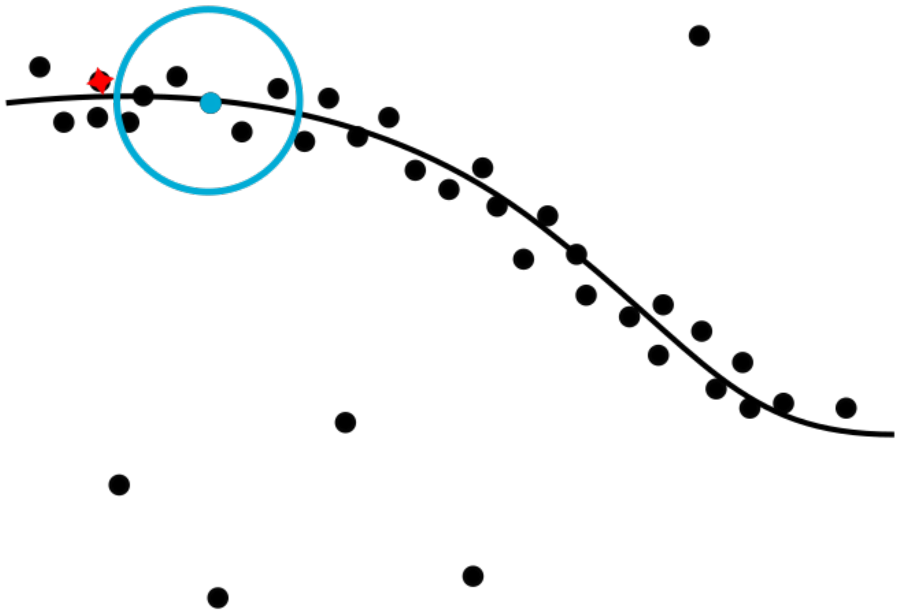
 for $i \longleftarrow 1$ *to* $|P|$ **do**

  if $Q_{i-1} \cap B(p_i, 2d_{P,k}(p_i)) = \emptyset$ **then**

   $Q_i = Q_{i-1} \cup \{p_i\}$

  else $Q_i = Q_{i-1}$

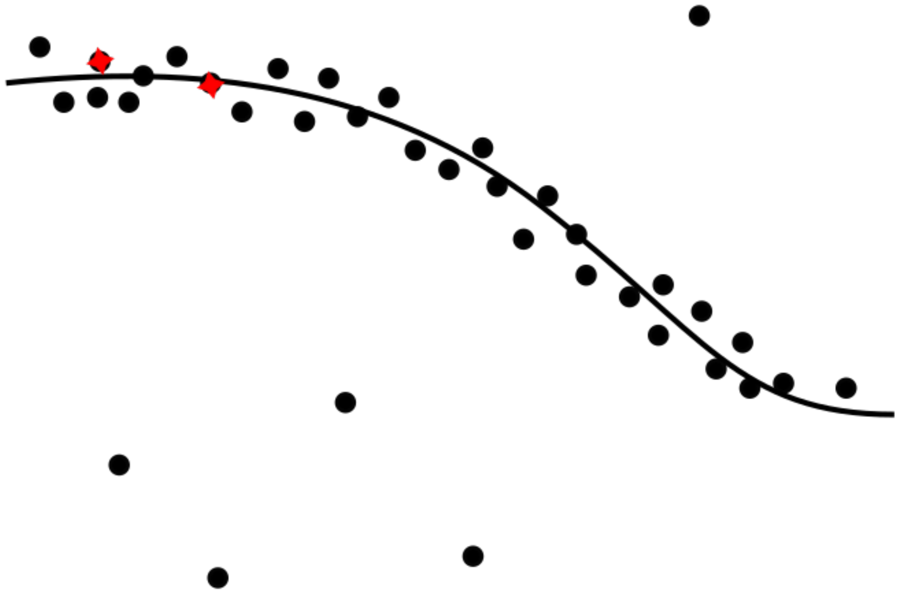
 $Q \longleftarrow Q_n$



Declutter algorithm

Algorithm 1: Declutter(P,k)

```
Data: Point set P, parameter k
Result: Denoised point set Q
begin
    sort P such that d_{P,k}(p_1) ≤ ··· ≤ d_{P,k}(p_{|P|}).
    Q_0 ⟵ ∅
    for i ⟵ 1 to |P| do
        if Q_{i-1} ∩ B(p_i, 2d_{P,k}(p_i)) = ∅ then
            Q_i = Q_{i-1} ∪ {p_i}
        else  Q_i = Q_{i-1}
    Q ⟵ Q_n
```



Declutter algorithm

Algorithm 1: Declutter(P,k)

Data: Point set P, parameter k

Result: Denoised point set Q

begin

  sort P such that $d_{P,k}(p_1) \leq \cdots \leq d_{P,k}(p_{|P|})$.

  $Q_0 \longleftarrow \emptyset$

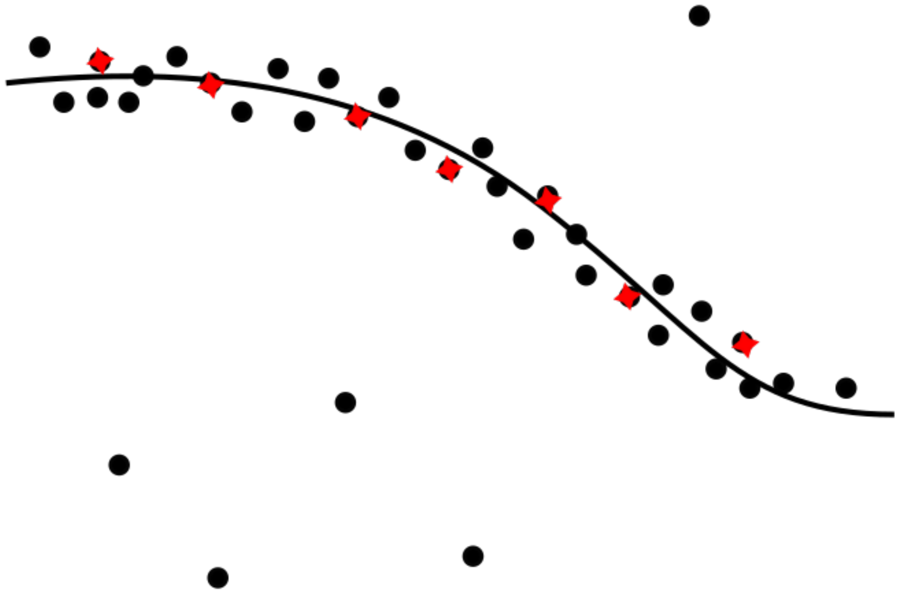
  for $i \longleftarrow 1$ *to* $|P|$ **do**

    if $Q_{i-1} \cap B(p_i, 2d_{P,k}(p_i)) = \emptyset$ **then**

      $Q_i = Q_{i-1} \cup \{p_i\}$

    else $Q_i = Q_{i-1}$

  $Q \longleftarrow Q_n$



Declutter algorithm

Algorithm 1: Declutter(P,k)

Data: Point set P, parameter k

Result: Denoised point set Q

begin

  sort P such that $d_{P,k}(p_1) \leq \cdots \leq d_{P,k}(p_{|P|})$.

  $Q_0 \longleftarrow \emptyset$

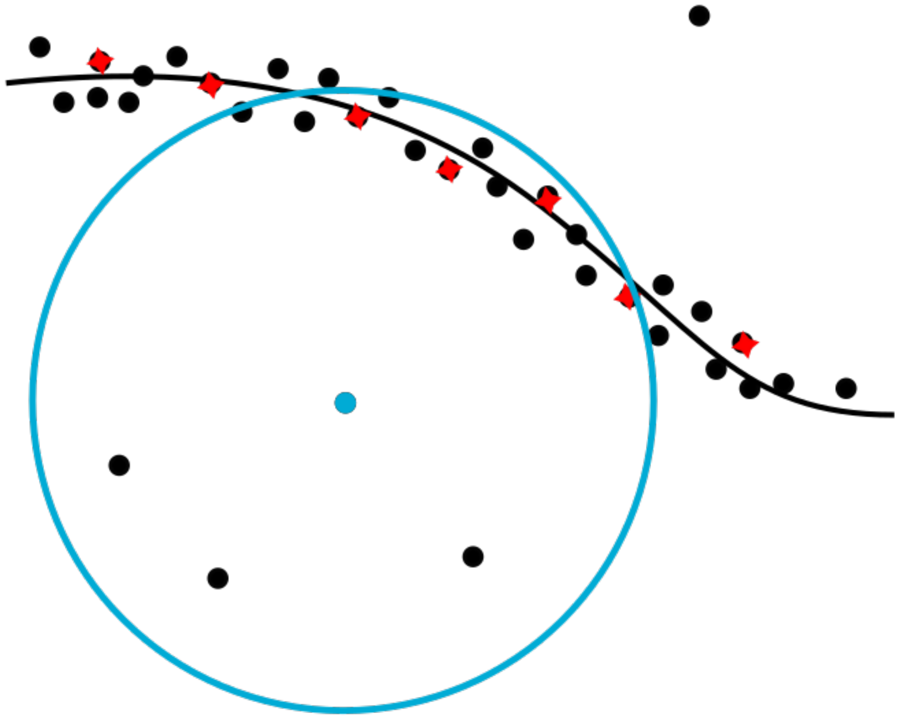
  for $i \longleftarrow 1$ *to* $|P|$ **do**

    if $Q_{i-1} \cap B(p_i, 2d_{P,k}(p_i)) = \emptyset$ **then**

      $Q_i = Q_{i-1} \cup \{p_i\}$

    else $Q_i = Q_{i-1}$

  $Q \longleftarrow Q_n$

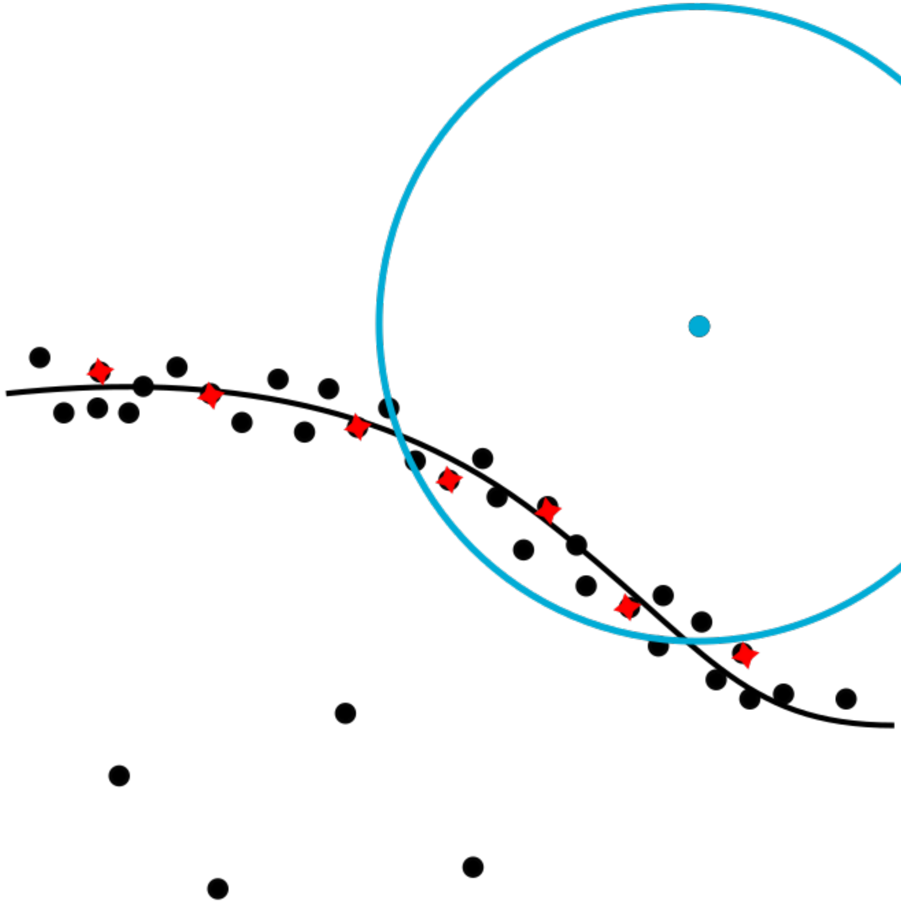


Declutter algorithm

Algorithm 1: Declutter(P,k)

Data: Point set P, parameter k
Result: Denoised point set Q

begin
  sort P such that $d_{P,k}(p_1) \leq \cdots \leq d_{P,k}(p_{|P|})$.
  $Q_0 \longleftarrow \emptyset$
  for $i \longleftarrow 1$ *to* $|P|$ **do**
    if $Q_{i-1} \cap B(p_i, 2d_{P,k}(p_i)) = \emptyset$ **then**
      $Q_i = Q_{i-1} \cup \{p_i\}$
    else $Q_i = Q_{i-1}$
  $Q \longleftarrow Q_n$



Declutter algorithm

Algorithm 1: Declutter(P,k)

Data: Point set P, parameter k

Result: Denoised point set Q

begin

sort P such that $d_{P,k}(p_1) \leq \cdots \leq d_{P,k}(p_{|P|})$.

$Q_0 \longleftarrow \emptyset$

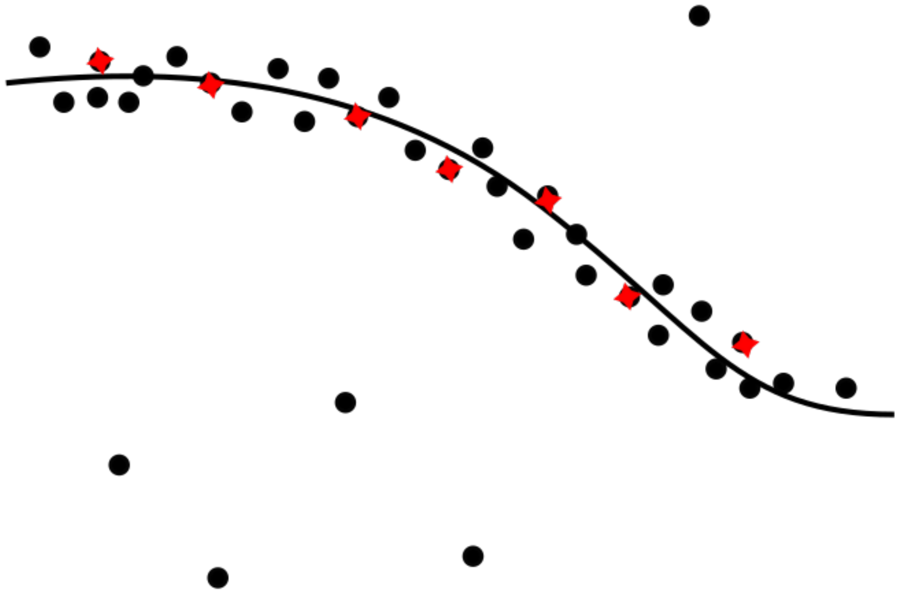
for $i \longleftarrow 1$ *to* $|P|$ **do**

if $Q_{i-1} \cap B(p_i, 2d_{P,k}(p_i)) = \emptyset$ **then**

$Q_i = Q_{i-1} \cup \{p_i\}$

else $Q_i = Q_{i-1}$

$Q \longleftarrow Q_n$



Declutter algorithm

Theorem *Given a point set P which is an ϵ_k-noisy sample of a compact set $K \subseteq \mathbb{X}$, Algorithm* Declutter *returns a set $Q \subseteq P$ such that $\delta_H(K, Q) \leq 7\epsilon_k$.*

Hausdorff distance $\delta\,(K, Q)$ between K and Q

Infimum of δ such that:

$$\forall p \in Q, d_{\mathbb{X}}(p, K) \leq \delta,$$
$$\forall x \in K, d_{\mathbb{X}}(x, Q) \leq \delta$$

Declutter algorithm - adaptive version

Theorem *Given an ϵ_k-adaptive noisy sample P of a compact set $K \subseteq \mathbb{X}$ with feature size f, Algorithm* Declutter *returns a sample $Q \subseteq P$ of K where $\delta_H^f(Q, K) \leq 7\epsilon_k$.*

Infimum of δ such that:

$$\forall p \in Q, d_{\mathbb{X}}(p, K) \leq \delta\, f(\bar{p}),$$
$$\forall x \in K, d_{\mathbb{X}}(x, Q) \leq \delta\, f(x)$$

Declutter algorithm – experiment

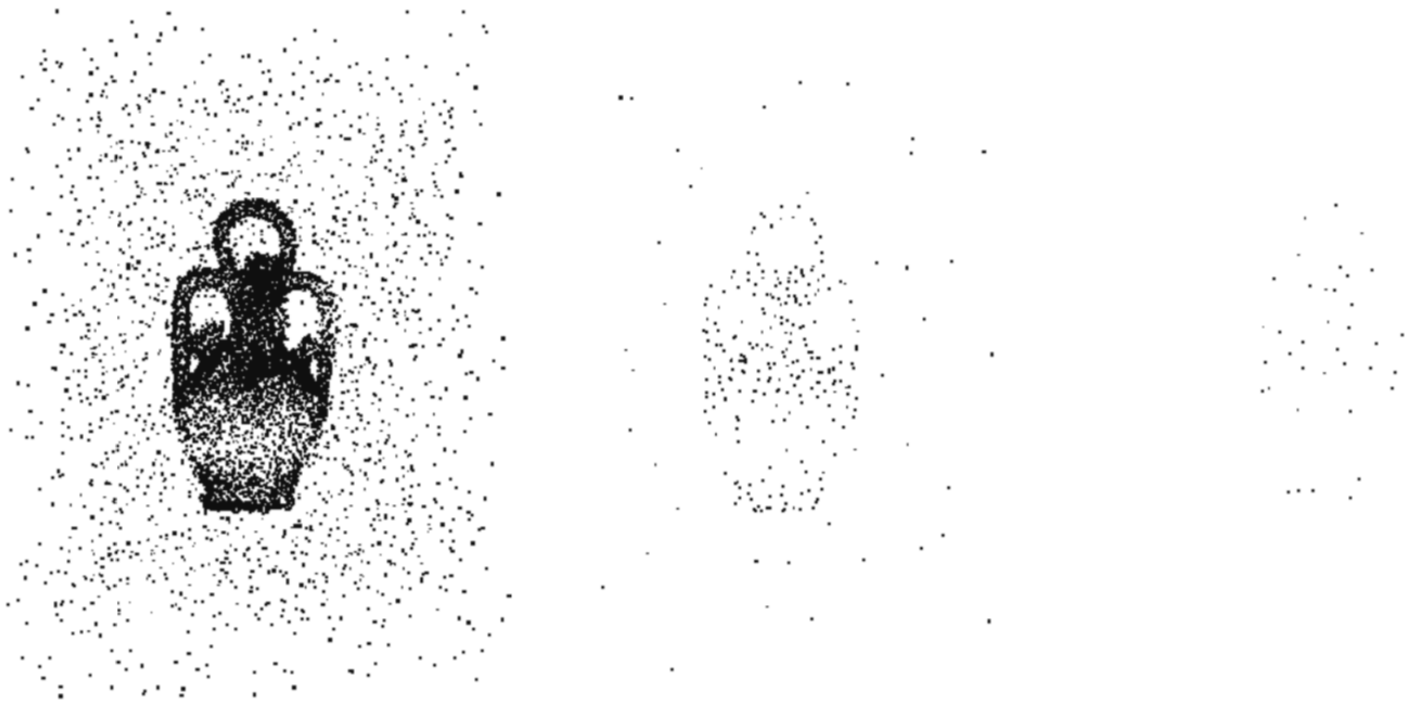


Figure 4: From left to right, the ground truth, the noisy input and the output of Algorithm Declutter for $k = 81$ and $k = 148$

Outline-Parameter-free denoising algorithm

- Parameter-free denoising algorithm
  - Introduction
  - Sampling condition
  - Declutter algorithm
  - Parameter-free algorithm
  - Discussions

Parameter-free algorithm

Algorithm 2: ParfreeDeclutter(P)

Data: Point set P

Result: Denoised point set P_0

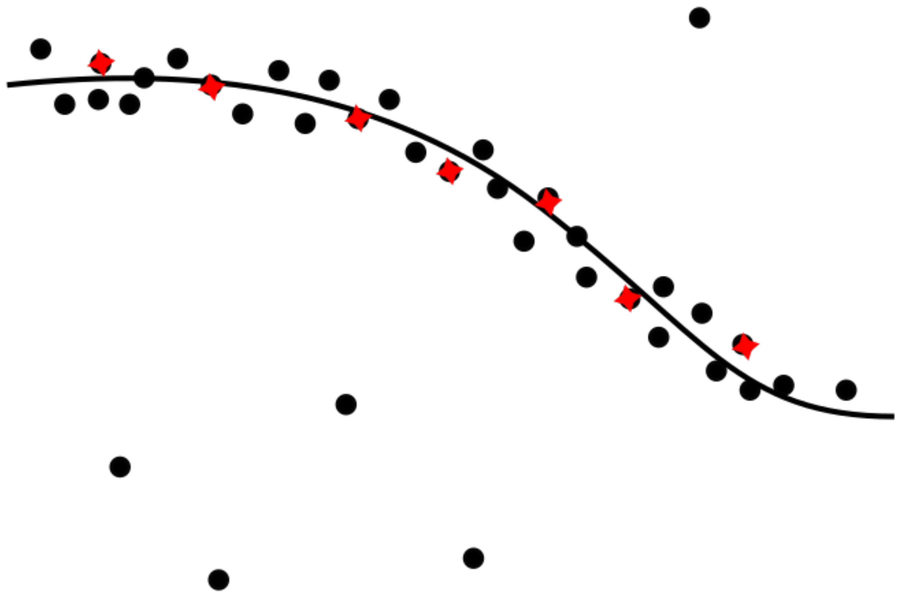
begin

  Set $i_* = \lfloor \log_2(|P|) \rfloor$, and $P_{i_*} \longleftarrow P$

  for $i \longleftarrow i_*$ *to* 1 **do**

    $Q \longleftarrow \text{Declutter}(P_i, 2^i)$

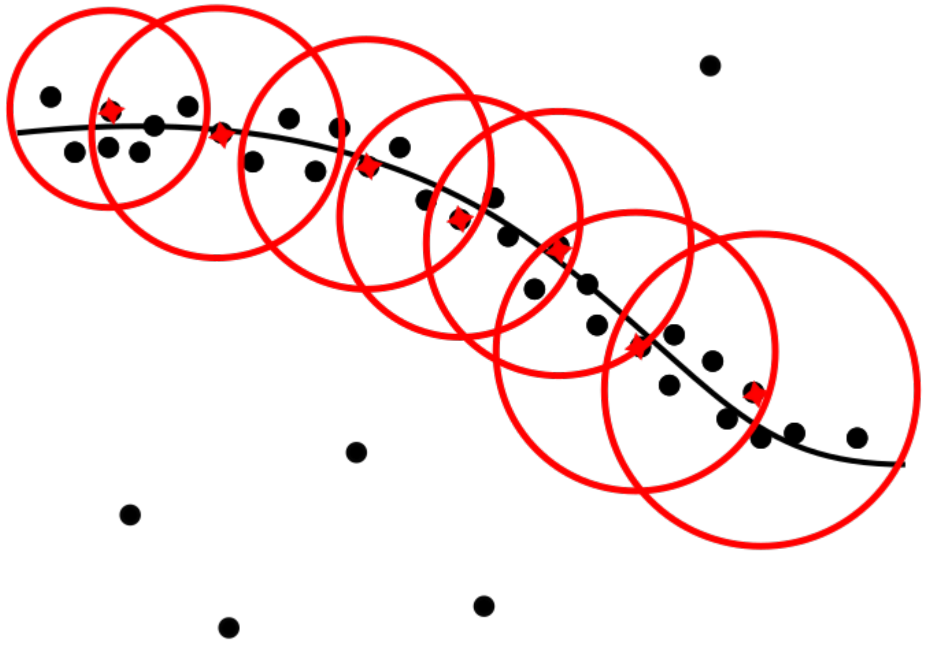
    $P_{i-1} \longleftarrow \cup_{q \in Q} B(q, (10 + 2\sqrt{2}) d_{P_i, 2^i}(q)) \cap P_i$



Parameter-free algorithm

Algorithm 2: ParfreeDeclutter(P)

Data: Point set P
Result: Denoised point set P_0
begin
  Set $i_* = \lfloor \log_2(|P|) \rfloor$, and $P_{i_*} \longleftarrow P$
  for $i \longleftarrow i_*$ *to* 1 **do**
    $Q \longleftarrow \mathsf{Declutter}(P_i, 2^i)$
    $P_{i-1} \longleftarrow \cup_{q \in Q} B(q, (10 + 2\sqrt{2}) d_{P_i, 2^i}(q)) \cap P_i$



Parameter-free algorithm

Algorithm 2: ParfreeDeclutter(P)

Data: Point set P

Result: Denoised point set P_0

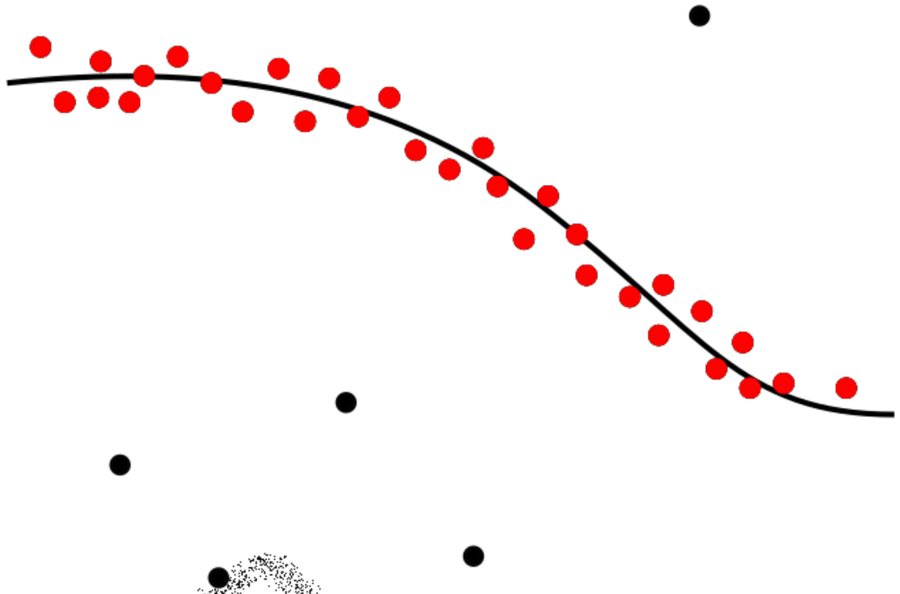
begin

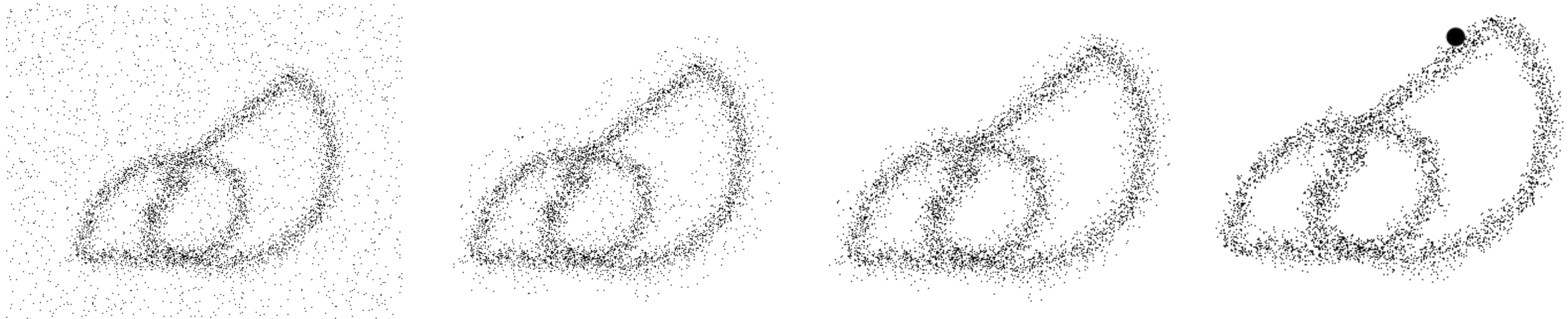
  Set $i_* = \lfloor \log_2(|P|) \rfloor$, and $P_{i_*} \longleftarrow P$

  for $i \longleftarrow i_*$ *to* 1 **do**

    $Q \longleftarrow$ Declutter(P_i,2^i)

    $P_{i-1} \longleftarrow \cup_{q \in Q} B(q, (10 + 2\sqrt{2}) d_{P_i, 2^i}(q)) \cap P_i$





k=128 k=64 k=2

Parameter-free algorithm

Theorem *Given a point set* P *and* i_0 *such that for all* $i > i_0$, P *is a weak uniform* $(\epsilon_{2^i}, 2)$*-noisy sample of* K *and is also a uniform* $(\epsilon_{2^{i_0}}, 2)$*-noisy sample of* K*, algorithm* ParfreeDeclutter *returns a point set* $P_0 \subseteq P$ *such that* $\delta_H(P_0, K) \leq \left(87 + 16\sqrt{2}\right)\epsilon_{2^{i_0}}$.

- P is a uniform (ϵ_k, c)-noisy sample of K if:
  - $\forall x \in K, d_{P,k}(x) \leq \epsilon_k$
  - $\forall x \in \mathbb{X}, d_{\mathbb{X}}(x, K) \leq d_{P,k}(x) + \epsilon_k$
  - $\forall p \in P, d_{P,k}(p) \geq \frac{\epsilon_k}{c}$

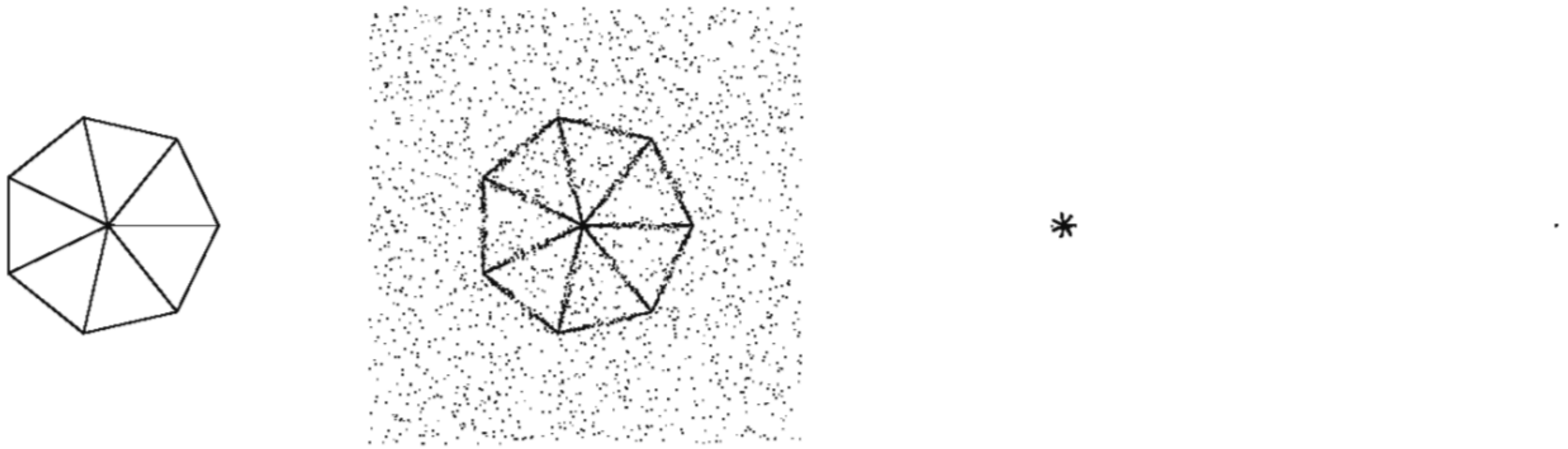
Parameter-free algorithm

Theorem *Given a point set* P *and* i_0 *such that for all* $i > i_0$, P *is a weak uniform* $(\epsilon_{2^i}, 2)$*-noisy sample of* K *and is also a uniform* $(\epsilon_{2^{i_0}}, 2)$*-noisy sample of* K*, algorithm* ParfreeDeclutter *returns a point set* $P_0 \subseteq P$ *such that* $\delta_H(P_0, K) \leq \left(87 + 16\sqrt{2}\right)\epsilon_{2^{i_0}}$.

- P is a weak uniform (ϵ_k, c)-noisy sample of K if:
  - $\forall x \in K, d_{P,k}(x) \leq \epsilon_k$
  - $\cancel{\forall x \in \mathbb{X}, d_{\mathbb{X}}(x, K) \leq d_{P,k}(x) + \epsilon_k}$
  - $\forall p \in P, d_{P,k}(p) \geq \frac{\epsilon_k}{\mathrm{c}}$

Parameter-free algorithm

- Example where the parameter-free algorithm doesn’t work.
- non-uniform



Ground truth | Noisy input | Intermediate step | Final output

Parameter-free algorithm-experiment

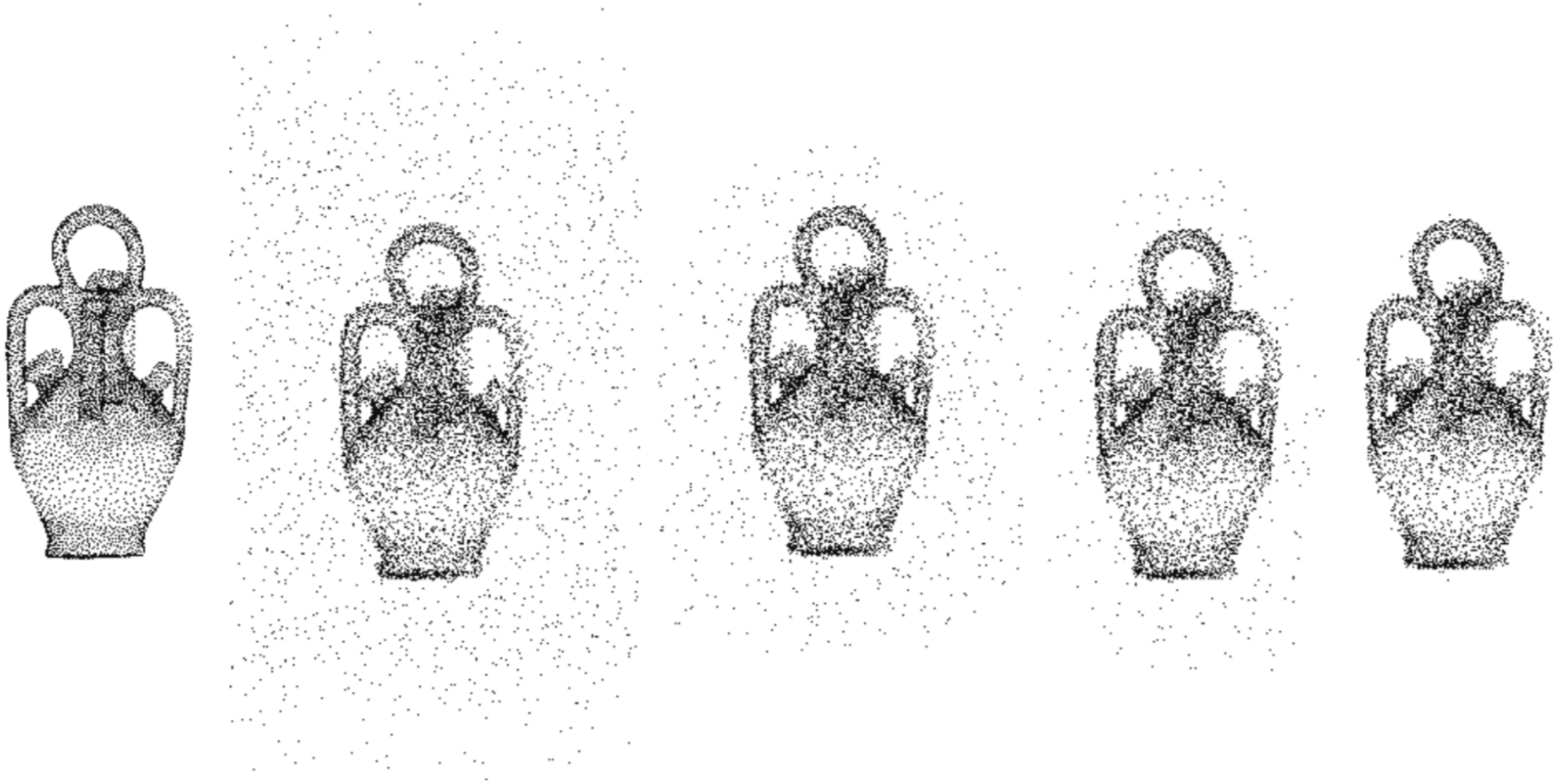


Figure 9: Experiment on a two dimensional manifold. From left to right, the ground truth, the noisy input, two intermediate steps of Algorithm ParfreeDeclutter and the final result.

Parameter-free algorithm-experiment

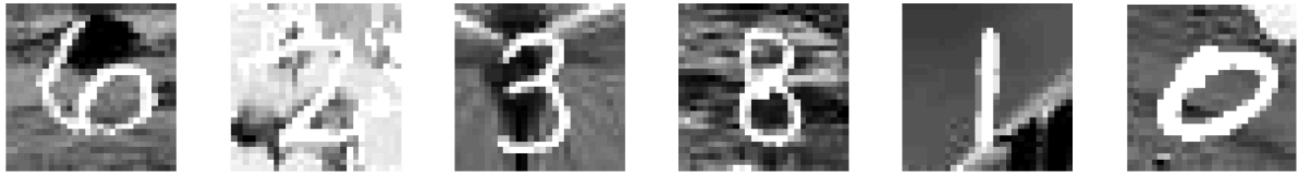
Swap noise:

Label 1 Label 7 Label 7 Label 1

1						Error(%)
2	Original	# Digit 1 1352		# Digit 7 1279		0.6564
3	Swap. Noise	# Mislabelled 1 270		# Mislabelled 7 266		4.0957
4		Digit 1		Digit 7		
5		# Removed	# True Noise	# Removed	# True Noise	
6	L1 Denoising	314	264	17	1	2.4500

Parameter-free algorithm-experiment

background noise:



1				Error(%)
2	Original	# Digit 1 1352	# Digit 7 1279	0.6564

7	Back. Noise	# Noisy 1 250		# Noisy 7 250		1.1464
8		Digit 1		Digit 7		
9		# Removed	# True Noise	# Removed	# True Noise	
10	L1 Denoising	294	250	277	250	0.7488

Outline-Parameter-free denoising algorithm

- Parameter-free denoising algorithm
  - Introduction
  - Sampling condition
  - Declutter algorithm
  - Parameter-free algorithm
  - Discussions

Discussions

- Relax the sampling conditions
- Estimate Hausdorff distance in a parameter free manner
- Adaptive theoretical guarantees for algorithm ParfreeDeclutter