Main Questions

1. **Data Clustering**
   What is the $k$-means method?

2. **Smoothed Analysis**
   What can we do when worst case analysis is too pessimistic?

3. **Smoothed Analysis of $k$-Means Method**
   What is the smoothed complexity of the $k$-means method?

4. **Extensions and Conclusions**
Input:
point set $X \subseteq \mathbb{R}^d$, $|X| = n$
number of clusters $k$
Data Clustering

**Input:** point set $X \subseteq \mathbb{R}^d$, $|X| = n$
number of clusters $k$

**Output:** partition $C_1, \ldots, C_k$
centers $c_1, \ldots, c_k \in \mathbb{R}^d$
Data Clustering

**Input:** point set $X \subseteq \mathbb{R}^d$, $|X| = n$

data points from a $d$-dimensional space with $n$ points

**Output:** partition $C_1, \ldots, C_k$

clusters of points

**Goal:**

$$\min \sum_{i=1}^k \sum_{x \in C_i} \|x - c_i\|^2$$

$\ell_2$-norm of the distance to the closest center

David Arthur, Bodo Manthey, Heiko Röglin

Smoothed Analysis of the $k$-Means Method
Data Clustering

**Input:** point set $X \subseteq \mathbb{R}^d$, $|X| = n$
number of clusters $k$

**Output:** partition $C_1, \ldots, C_k$
centers $c_1, \ldots, c_k \in \mathbb{R}^d$

**Goal:** $\min \sum_{i=1}^{k} \sum_{x \in C_i} \|x - c_i\|^2$

**Theory:** The problem is NP-hard, but a PTAS exists.
running time is exponential in $k$

**Practice:** $k$-Means Method.
Local Search

$k$-Means Method

Local search based on two observations:

1. clusters $C_i$ fixed

$\Rightarrow$ centers $c_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$
Local Search

**k-Means Method**

Local search based on two observations:

1. clusters $C_i$ fixed
   $\Rightarrow$ centers $c_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$
$k$-Means Method

Local search based on two observations:

1. clusters $C_i$ fixed
   $\Rightarrow$ centers $c_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$

2. centers $c_i$ fixed
   $\Rightarrow$ clusters $C_i$ fixed
Local Search

\textit{k-Means Method}

Local search based on two observations:

1. clusters $C_i$ fixed
   \Rightarrow centers $c_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$

2. centers $c_i$ fixed
   \Rightarrow clusters $C_i$ fixed
**k-Means Method**

**Input:** $X \subseteq \mathbb{R}^d$, $k$

1. **choose** $c_1, \ldots, c_k \in \mathbb{R}^d$

2. Repeat

3. partition $X$ into $C_1, \ldots, C_k$

4. $c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x$

5. Until stable
**k-Means Method**

**Input:** \( X \subseteq \mathbb{R}^d, k \)

1. choose \( c_1, \ldots, c_k \in \mathbb{R}^d \)
2. Repeat
3. **partition** \( X \) into \( C_1, \ldots, C_k \)
4. \( c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x \)
5. Until stable
**k- Means Method**

**Input:** $X \subseteq \mathbb{R}^d$, $k$

1. choose $c_1, \ldots, c_k \in \mathbb{R}^d$

2. Repeat

3. partition $X$ into $C_1, \ldots, C_k$

4. $c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x$

5. Until stable
**k-Means Method**

**Input:** $X \subseteq \mathbb{R}^d$, $k$

1. choose $c_1, \ldots, c_k \in \mathbb{R}^d$
2. Repeat
3. **partition $X$ into $C_1, \ldots, C_k$**
4. $c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x$
5. Until stable
**k- Means Method**

**Input:** \( X \subseteq \mathbb{R}^d, k \)

1. choose \( c_1, \ldots, c_k \in \mathbb{R}^d \)
2. Repeat
3. partition \( X \) into \( C_1, \ldots, C_k \)
4. \( c_i \leftarrow \frac{1}{|C_i|} \sum_{x \in C_i} x \)
5. Until stable
**k-Means Method**

**Input:** \(X \subseteq \mathbb{R}^d, k\)

1. choose \(c_1, \ldots, c_k \in \mathbb{R}^d\)

2. Repeat

3. **partition** \(X\) **into** \(C_1, \ldots, C_k\)

4. \(c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x\)

5. Until stable

---

David Arthur, Bodo Manthey, Heiko Röglin  
Smoothed Analysis of the *k*-Means Method
### k-Means Method

**Input:** $X \subseteq \mathbb{R}^d$, $k$

1. choose $c_1, \ldots, c_k \in \mathbb{R}^d$
2. Repeat
3. partition $X$ into $C_1, \ldots, C_k$
4. $c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x$
5. Until stable

---

David Arthur, Bodo Manthey, Heiko Röglin

Smoothed Analysis of the k-Means Method
**k-Means Method**

**Input:** \( X \subseteq \mathbb{R}^d, k \)

1. choose \( c_1, \ldots, c_k \in \mathbb{R}^d \)
2. Repeat
3. **partition** \( X \) into \( C_1, \ldots, C_k \)
4. \( c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x \)
5. Until stable
**k-Means Method**

**Input:** $X \subseteq \mathbb{R}^d$, $k$

1. choose $c_1, \ldots, c_k \in \mathbb{R}^d$
2. Repeat
3. partition $X$ into $C_1, \ldots, C_k$
4. $c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x$
5. Until stable
**k-Means Method**

**Input:** $X \subseteq \mathbb{R}^d$, $k$

1. choose $c_1, \ldots, c_k \in \mathbb{R}^d$
2. Repeat
3. partition $X$ into $C_1, \ldots, C_k$
4. $c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x$
5. Until stable
**k-Means Method**

**Input:** $X \subseteq \mathbb{R}^d$, $k$

1. choose $c_1, \ldots, c_k \in \mathbb{R}^d$
2. Repeat
3. partition $X$ into $C_1, \ldots, C_k$
4. $c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x$
5. Until stable

---

David Arthur, Bodo Manthey, Heiko Röglín  
Smoothed Analysis of the $k$-Means Method
**k-Means Method**

**Input:** $X \subseteq \mathbb{R}^d$, $k$

1. choose $c_1, \ldots, c_k \in \mathbb{R}^d$
2. Repeat
3. partition $X$ into $C_1, \ldots, C_k$
4. $c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x$
5. **Until stable**
**k-Means Method**

**k-Means Method**

**Input:** $X \subseteq \mathbb{R}^d$, $k$

1. choose $c_1, \ldots, c_k \in \mathbb{R}^d$
2. Repeat
3. partition $X$ into $C_1, \ldots, C_k$
4. $c_i \leftarrow \frac{1}{|C_i|} \cdot \sum_{x \in C_i} x$
5. Until stable

“by far the most popular clustering algorithm used in scientific and industrial applications” (Berkhin 2002)

“in practice the number of iterations is generally much less than the number of points” (Duda et al. 2001)
Running Time

**Upper Bound:** At most \((k^2 n)^{kd}\) iterations.
No clustering can occur twice.

**Lower Bound:** At least \(2^{\Omega(k)}\) iterations for \(d \geq 2\).

[Andrea Vattani (SoCG’09)]
Running Time

Upper Bound: At most \((k^2 n)^{kd}\) iterations. No clustering can occur twice.

Lower Bound: At least \(2^{\Omega(k)}\) iterations for \(d \geq 2\).

[Andrea Vattani (SoCG’09)]

Quality

Local optima can be arbitrarily bad.
**Running Time**

Upper Bound: **At most** \((k^2 n)^{kd}\) iterations.
No clustering can occur twice.

Lower Bound: **At least** \(2^{\Omega(k)}\) iterations for \(d \geq 2\).

[Andrea Vattani (SoCG’09)]

**Quality**

Local optima can be **arbitrarily bad**.

⇒ **Huge discrepancy between theory and practice.**

(Focus of this talk: running time.)
Main Questions

1. Data Clustering
   What is the $k$-means method?

2. Smoothed Analysis
   What can we do when worst case analysis is too pessimistic?

3. Smoothed Analysis of $k$-Means Method
   What is the smoothed complexity of the $k$-means method?

4. Extensions and Conclusions
Smoothed Analysis

- Worst-case analysis is too pessimistic. Typical instances are not adversarial.
Smoothed Analysis

- Worst-case analysis is too pessimistic. Typical instances are not adversarial.
- Average-case analysis unrealistic. What is the right probability distribution?
Worst-case analysis is too pessimistic. Typical instances are not adversarial.

Average-case analysis unrealistic. What is the right probability distribution?
Worst-case analysis is too pessimistic. Typical instances are not adversarial.

Average-case analysis unrealistic. What is the right probability distribution?

Smoothed Analysis
Less powerful adversary:

1. Adversary chooses instance $I$. 

Smoothed Analysis of the $k$-Means Method

David Arthur, Bodo Manthey, Heiko Röglin
Worst-case analysis is too pessimistic. Typical instances are not adversarial.

Average-case analysis unrealistic. What is the right probability distribution?

Smoothed Analysis
Less powerful adversary:

1. Adversary chooses instance $I$.
2. Small amount of random noise is added to $I$.
   $\sigma = \text{amount of noise}$
Worst-case analysis is too pessimistic. Typical instances are not adversarial.

Average-case analysis unrealistic.
What is the right probability distribution?

Smoothed Analysis
Less powerful adversary:

1. Adversary chooses instance $I$.
2. Small amount of random noise is added to $I$.

$\sigma = \text{amount of noise}$

$$T(n, \sigma) = \max_{I, |I| = n} \mathbb{E}(\text{running time}(\text{per}_\sigma(I)))$$
Smoothed Analysis

- Worst-case analysis is too pessimistic. Typical instances are not adversarial.
- Average-case analysis unrealistic. What is the right probability distribution?

Smoothed Analysis

Less powerful adversary:

1. Adversary chooses instance $I$.
2. Small amount of random noise is added to $I$.

$\sigma = \text{amount of noise}$

$$T(n, \sigma) = \max_{\substack{I, |I| = n}} \mathbb{E}(\text{running time}(\text{per}_\sigma(I)))$$

- models, e.g., measurement errors or numerical imprecision
- Smoothed compl. low $\Rightarrow$ bad performance unlikely in practice
Smoothed Analysis introduced for the Simplex Method:

- Linear Programming:
  \[
  \max c \cdot x \text{ such that } Ax \leq b
  \]
Smoothed Analysis introduced for the Simplex Method:

- **Linear Programming:**
  \[ \max c \cdot x \text{ such that } Ax \leq b \]

- **Simplex Method:** Follow edges of the polytope. In the worst case slow.
Smoothed Analysis introduced for the Simplex Method:

- **Linear Programming:**\[ \max c \cdot x \text{ such that } Ax \leq b \]
- **Simplex Method:** Follow edges of the polytope. In the worst case slow.
Smoothed Analysis introduced for the Simplex Method:

- **Linear Programming:**
  \[ \text{max } c \cdot x \text{ such that } Ax \leq b \]

- **Simplex Method:** Follow edges of the polytope. In the worst case slow.
Smoothed Analysis introduced for the Simplex Method:

- **Linear Programming:**
  \[ \text{max } c \cdot x \text{ such that } Ax \leq b \]

- **Simplex Method:** Follow edges of the polytope. In the worst case slow.
Smoothed Analysis introduced for the Simplex Method:

- **Linear Programming:**
  \[
  \max c \cdot x \text{ such that } Ax \leq b
  \]

- **Simplex Method:** Follow edges of the polytope. In the worst case slow.
Smoothed Analysis introduced for the Simplex Method:

- **Linear Programming:** \( \max c \cdot x \) such that \( Ax \leq b \)
- **Simplex Method:** Follow edges of the polytope. In the worst case slow.
- **Perturbation:** Add independent Gaussian with standard deviation \( \sigma \) to every entry in matrix \( A \) and vector \( b \).
Smoothed Analysis introduced for the Simplex Method:

- **Linear Programming:**
  \[ \max c \cdot x \text{ such that } Ax \leq b \]

- **Simplex Method:** Follow edges of the polytope. In the worst case slow.

- **Perturbation:** Add independent Gaussian with standard deviation \( \sigma \) to every entry in matrix \( A \) and vector \( b \).

**Theorem [Spielman, Teng 2001]**

The smoothed running time of the simplex method with shadow vertex pivot rule is polynomial in \( n, m, \) and \( 1/\sigma \).
Smoothed Analysis introduced for the Simplex Method:

- **Linear Programming:**
  \[ \max c \cdot x \text{ such that } Ax \leq b \]

- **Simplex Method:** Follow edges of the polytope. In the worst case slow.

- **Perturbation:** Add independent Gaussian with standard deviation \( \sigma \) to every entry in matrix \( A \) and vector \( b \).

**Theorem [Spielman, Teng 2001]**

The smoothed running time of the simplex method with shadow vertex pivot rule is polynomial in \( n, m, \) and \( 1/\sigma \).

In 2006, Roman Vershynin improved bound from
\[ O^* \left( m^{86} n^{55} \sigma^{-30} \right) \text{ to } O \left( \max\{n^5 \log^2 m, n^9 \log^4 n, n^3 \sigma^{-4} \} \right). \]

Further results on condition number of matrices, interior-point method, Gaussian elimination, etc.
Outline

Main Questions

1. Data Clustering
   What is the $k$-means method?

2. Smoothed Analysis
   What can we do when worst case analysis is too pessimistic?

3. **Smoothed Analysis of the $k$-Means Method**
   What is the smoothed complexity of the $k$-means method?

4. Extensions and Conclusions
Model: Every point is perturbed by independent \( d \)-dimensional Gaussian with standard deviation \( \sigma \).

\[
T(n, \sigma) = \max_{X, |X|=n} E(\#\text{Iterations}(\text{per}_\sigma(X)))
\]
Smoothed Analysis of $k$-Means

**Model:** Every point is perturbed by independent $d$-dimensional Gaussian with standard deviation $\sigma$.

$$T(n, \sigma) = \max_{X, |X|=n} \mathbb{E}(\#\text{Iterations}(\text{per}_\sigma(X)))$$

Arthur, Vassilvitskii (FOCS 2006)
Smoothed number of iterations is at most $\text{poly}(n^k, 1/\sigma)$.

Manthey, Röglin (SODA 2009)
Smoothed number of iterations is at most
- $\text{poly}(n^{\sqrt{k}}, 1/\sigma)$
- $k^{kd} \cdot \text{poly}(n, 1/\sigma)$
Smoothed Analysis of $k$-Means

**Model:** Every point is perturbed by independent $d$-dimensional Gaussian with standard deviation $\sigma$.

$$T(n, \sigma) = \max_{X, |X| = n} \mathbb{E}(\text{#Iterations}(\text{per}_{\sigma}(X)))$$

Arthur, Vassilvitskii (FOCS 2006)

Smoothed number of iterations is at most $\text{poly}(n^k, 1/\sigma)$.

Manthey, Röglin (SODA 2009)

Smoothed number of iterations is at most

- $\text{poly}(n^{\sqrt{k}}, 1/\sigma)$
- $k^{kd} \cdot \text{poly}(n, 1/\sigma)$

Our Result (FOCS 2009)

Smoothed number of iterations is $\text{poly}(n, 1/\sigma)$. 
Initial Potential: After first iteration whp

\[ \Phi = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - c_i\|^2 = O(\text{poly}(n)) \]
General Approach

1. **Initial Potential**: After first iteration whp
   \[\Phi = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - c_i\|^2 = O(\text{poly}(n))\]

2. Define **smallest possible improvement**:
   \[\Delta = \min_{\text{iteration}: C \rightarrow \text{succ}(C)} (\Phi(C) - \Phi(\text{succ}(C)))\].

   \[\Rightarrow \text{at most } O(\text{poly}(n)/\Delta) \text{ steps.}\]
1. **Initial Potential**: After first iteration whp
   \[ \Phi = \sum_{i=1}^{k} \sum_{x \in C_i} \| x - c_i \|^2 = O(poly(n)) \]

2. Define smallest possible improvement:
   \[ \Delta = \min \limits_{\text{iteration: } C \to \text{succ}(C)} (\Phi(C) - \Phi(\text{succ}(C))) \cdot \]
   \[ \Rightarrow \text{at most } O(poly(n)/\Delta) \text{ steps.} \]

In the worst case: \( \Delta \) arbitrarily small.
General Approach

1. Initial Potential: After first iteration whp
   \[ \Phi = \sum_{i=1}^{k} \sum_{x \in C_i} \| x - c_i \|^2 = O(poly(n)) \]

2. Define smallest possible improvement:
   \[ \Delta = \min_{\text{iteration: } C \rightarrow \text{succ}(C)} (\Phi(C) - \Phi(\text{succ}(C))) \].

   \[ \Rightarrow \text{at most } O(poly(n)/\Delta) \text{ steps.} \]

   In the worst case: \( \Delta \) arbitrarily small.

Lemma

For \( d \geq 2 \), for every \( X \subseteq [0, 1]^d \), in the model of smoothed analysis:

\[ \mathbb{E} \left[ \frac{1}{\Delta} \right] = \text{poly}(n, 1/\sigma). \]

\[ \Rightarrow \max_{X, |X|=n} \mathbb{E}(\#\text{Iterations}(\text{per}_\sigma(X))) = \text{poly}(n, 1/\sigma). \]
When does the potential drop?

1) center moves by $\varepsilon$

$\Rightarrow$ improvement by $\varepsilon^2$
When does the potential drop?

1) center moves by $\varepsilon$

$\Rightarrow$ improvement by $\varepsilon^2$

2) point with distance $\varepsilon$ to bisector changes assignment

$\Rightarrow$ improvement by $2\varepsilon\delta$
When does the potential drop?

1) center moves by $\varepsilon$

$\Rightarrow$ improvement by $\varepsilon^2$

2) point with distance $\varepsilon$ to bisector changes assignment

$\Rightarrow$ improvement by $2\varepsilon\delta$

**Goal**: Show that in every iteration

1. either a center moves significantly
2. or a reassigned point is significantly far from bisector.
Configuration $C$ is $\varepsilon$-bad if $\Phi(C) - \Phi(\text{succ}(C)) \leq \varepsilon$.

Naive approach: **Union Bound** over all configurations.

$$\Pr[\exists \text{Configuration } C : C \text{ is } \varepsilon\text{-bad}] \leq \sum_{\text{Configuration } C} \Pr[C \text{ is } \varepsilon\text{-bad}]$$
Configuration \( C \) is \( \varepsilon \)-bad if \( \Phi(C) - \Phi(\text{succ}(C)) \leq \varepsilon \).

Naive approach: **Union Bound** over all configurations.

\[
\Pr[\exists \text{Configuration } C: C \text{ is } \varepsilon\text{-bad}] \leq \sum_{\text{Configuration } C} \Pr[C \text{ is } \varepsilon\text{-bad}]
\]

Problem: **Too many configurations**: \( k^n \).
How large is $\Delta$?

Configuration $\mathcal{C}$ is $\varepsilon$-bad if $\Phi(\mathcal{C}) - \Phi(\text{succ}(\mathcal{C})) \leq \varepsilon$.

Naive approach: Union Bound over all configurations.

$$\Pr[\exists \text{Configuration } \mathcal{C} : \mathcal{C} \text{ is } \varepsilon\text{-bad}] \leq \sum_{\text{Configuration } \mathcal{C}} \Pr[\mathcal{C} \text{ is } \varepsilon\text{-bad}]$$

Problem: Too many configurations: $k^n$.
How large is $\Delta$?

Configuration $C$ is $\varepsilon$-bad if $\Phi(C) - \Phi(\text{succ}(C)) \leq \varepsilon$.

Naive approach: Union Bound over all configurations.

$$\Pr\left[\exists \text{Configuration } C : C \text{ is } \varepsilon\text{-bad}\right] \leq \sum_{\text{Configuration } C} \Pr\left[C \text{ is } \varepsilon\text{-bad}\right]$$

Problem: Too many configurations: $k^n$.

$$\Pr\left[\exists \text{Configuration } C : C \text{ is } \varepsilon\text{-bad}\right] \leq \sum_{i=1}^{5} \Pr\left[\mathcal{F}_i\right]$$
Transition Graph $G = (V, E)$:

$V$: clusters  

$E$: labeled directed edge for each reassigned point

David Arthur, Bodo Manthey, Heiko Röglin

Smoothed Analysis of the $k$-Means Method
Transition Graph $G = (V, E)$:

- $V$: clusters
- $E$: labeled directed edge for each reassigned point

Approach: Union bound over different transition blueprints.
Transition Graph $G = (V, E)$:

$V$: clusters  \quad E$: labeled directed edge for each reassigned point

Approach: Union bound over different transition blueprints.

- First glance: Natural idea.
- Second glance: Not enough information.
  E.g.: No information about positions of centers and bisectors.
Transition Graph $G = (V, E)$:

- $V$: clusters
- $E$: labeled directed edge for each reassigned point

Approach: Union bound over different transition blueprints.

- First glance: Natural idea.
- Second glance: Not enough information. E.g.: No information about positions of centers and bisectors.
- Third glance: Enough information!
**Goal**: Show that in every iteration

1. either a center moves significantly
2. or a reassigned point is significantly far from bisector.

\[ \| \text{cm}((C \cup A) \setminus B) - \text{cm}(C) \| \geq \frac{1}{n^2} \| \text{cm}(C) - \left( \frac{|B| \text{cm}(B) - |A| \text{cm}(A)}{|B| - |A|} \right) \| \]
**Goal**: Show that in every iteration

1. either a center moves significantly
2. or a reassigned point is significantly far from bisector.

\[
\| \text{cm}((C \cup A) \setminus B) - \text{cm}(C) \| \geq \frac{1}{n^2} \| \text{cm}(C) - \left( \frac{|B| \text{cm}(B) - |A| \text{cm}(A)}{|B| - |A|} \right) \|
\]

small potential drop \(\Rightarrow\) \(\text{cm}(C)\) must be close to \(\text{approx}(A, B)\)
Proof of Main Theorem

Theorem

Smoothed number of iterations is $\text{poly}(n, 1/\sigma)$.
Proof of Main Theorem

Theorem

Smoothed number of iterations is \( \text{poly}(n, 1/\sigma) \).

- number of blueprints with \( m \) edges: \( (k^2n)^m \).
Proof of Main Theorem

Theorem

Smoothed number of iterations is $\text{poly}(n, 1/\sigma)$.

- number of blueprints with $m$ edges: $(k^2 n)^m$.
- probability that a fixed data point has distance at most $\varepsilon$ from its approximate bisector: $\leq \varepsilon/\sigma$. 
Proof of Main Theorem

Theorem

Smoothed number of iterations is \( \text{poly}(n, 1/\sigma) \).

- number of blueprints with \( m \) edges: \( (k^2 n)^m \).
- probability that a fixed data point has distance at most \( \varepsilon \) from its approximate bisector: \( \leq \varepsilon / \sigma \).
- probability that all \( m \) data points are \( \varepsilon \)-close to their approximate bisectors: \( \leq (\varepsilon / \sigma)^m \)
Proof of Main Theorem

**Theorem**

Smoothed number of iterations is \( \text{poly}(n, 1/\sigma) \).

- number of blueprints with \( m \) edges: \( (k^2 n)^m \).
- probability that a fixed data point has distance at most \( \varepsilon \) from its approximate bisector: \( \leq \varepsilon / \sigma \).
- probability that all \( m \) data points are \( \varepsilon \)-close to their approximate bisectors: \( \leq (\varepsilon / \sigma)^m \).

**Union Bound:**

\[
\Pr[\exists \varepsilon\text{-bad blueprint}] \leq (k^2 n)^m \cdot (\varepsilon / \sigma)^m
\]

Very unlikely for \( \varepsilon = 1 / \text{poly}(n, 1/\sigma) \).
Proof of Main Theorem

**Theorem**

Smoothed number of iterations is $\text{poly}(n, 1/\sigma)$.

- number of blueprints with $m$ edges: $(k^2 n)^m$.
- probability that a fixed data point has distance at most $\epsilon$ from its approximate bisector: $\leq \epsilon/\sigma$.
- probability that all $m$ data points are $\epsilon$-close to their approximate bisectors: $\leq (\epsilon/\sigma)^m$
- Union Bound:

$$
\Pr[\exists \epsilon\text{-bad blueprint}] \leq (k^2 n)^m \cdot (\epsilon/\sigma)^m
$$

Very unlikely for $\epsilon = 1/\text{poly}(n, 1/\sigma)$.

**Technical Difficulties:** Data points are not independent from approx. bisectors, approximate centers not defined for balanced clusters, blueprints must have enough edges, . . .
Main Questions

1. **Data Clustering**
   What is the $k$-means method?

2. **Smoothed Analysis**
   What can we do when worst case analysis is too pessimistic?

3. **Smoothed Analysis of $k$-Means Method**
   What is the smoothed complexity of the $k$-means method?

4. **Extensions and Conclusions**
Text Classification and Bag-of-Words Model:

- $S$ set of all words
- count words and normalize: prob. distribution $p: S \rightarrow [0, 1]$
Kullback-Leibler Divergence

Text Classification and Bag-of-Words Model:

- $S$ set of all words
- Count words and normalize: prob. distribution $p: S \rightarrow [0, 1]$

Kullback-Leibler divergence (relative entropy):

$$KLD(p, q) = \sum_{i=1}^{d} p_i \log \left( \frac{p_i}{q_i} \right)$$

- number of bits to encode $p$ with Huffman code for $q$
- number of bits to encode $p$ with Huffman code for $p$
Bregman Divergences

**Bregman divergences** are distance measures that generalize squared Euclidean distances and the Kullback-Leibler divergence.

**Ackermann, Blömer, Sohler (SODA 2008)**

**Approximation Scheme** for special cases of Bregman divergences (e.g., Kullback-Leibler divergence).
Bregman divergences are distance measures that generalize squared Euclidean distances and the Kullback-Leibler divergence.

Ackermann, Blömer, Sohler (SODA 2008)
Approximation Scheme for special cases of Bregman divergences (e.g., Kullback-Leibler divergence).

Manthey, Röglin (ISAAC 2009)
For any well-behaved Bregman divergence:
Smoothed number of iterations is at most
- \(\text{poly}(n^{\sqrt{k}}, 1/\sigma)\)
- \(k^{kd} \cdot \text{poly}(n, 1/\sigma)\)
- \(\text{poly}(n, 1/\sigma)\) if \(d, k = O(\sqrt{\log n / \log \log n})\) (or \(d = 1\))
Bregman Divergences

Bregman divergences are distance measures that generalize squared Euclidean distances and the Kullback-Leibler divergence.

Ackermann, Blömer, Sohler (SODA 2008)

Approximation Scheme for special cases of Bregman divergences (e.g., Kullback-Leibler divergence).

Manthey, Röglin (ISAAC 2009)

For any well-behaved Bregman divergence:
Smoothed number of iterations is at most

- $\text{poly}(n^{\sqrt{k}}, 1/\sigma)$
- $k^{kd} \cdot \text{poly}(n, 1/\sigma)$
- $\text{poly}(n, 1/\sigma)$ if $d, k = O(\sqrt{\log n / \log \log n})$ (or $d = 1$)

Polynomial bound does not extend as it uses special properties of Gaussian perturbations.
Summary:

- Worst-case instances are often fragile.
- Smoothed Analysis often leads to better understanding of observed practical behavior.
Further Results and Open Questions

Summary:
- Worst-case instances are often fragile.
- Smoothed Analysis often leads to better understanding of observed practical behavior.

Future Research:
- improve exponents for $k$-means (currently $\approx n^{30}$)
- better understanding of dynamics seems necessary for this explanation for good approximation ratio
- better analysis of Bregman divergences
- more systematic theory of smoothed local search
- Are all local search problems in PLS easy in smoothed analysis?