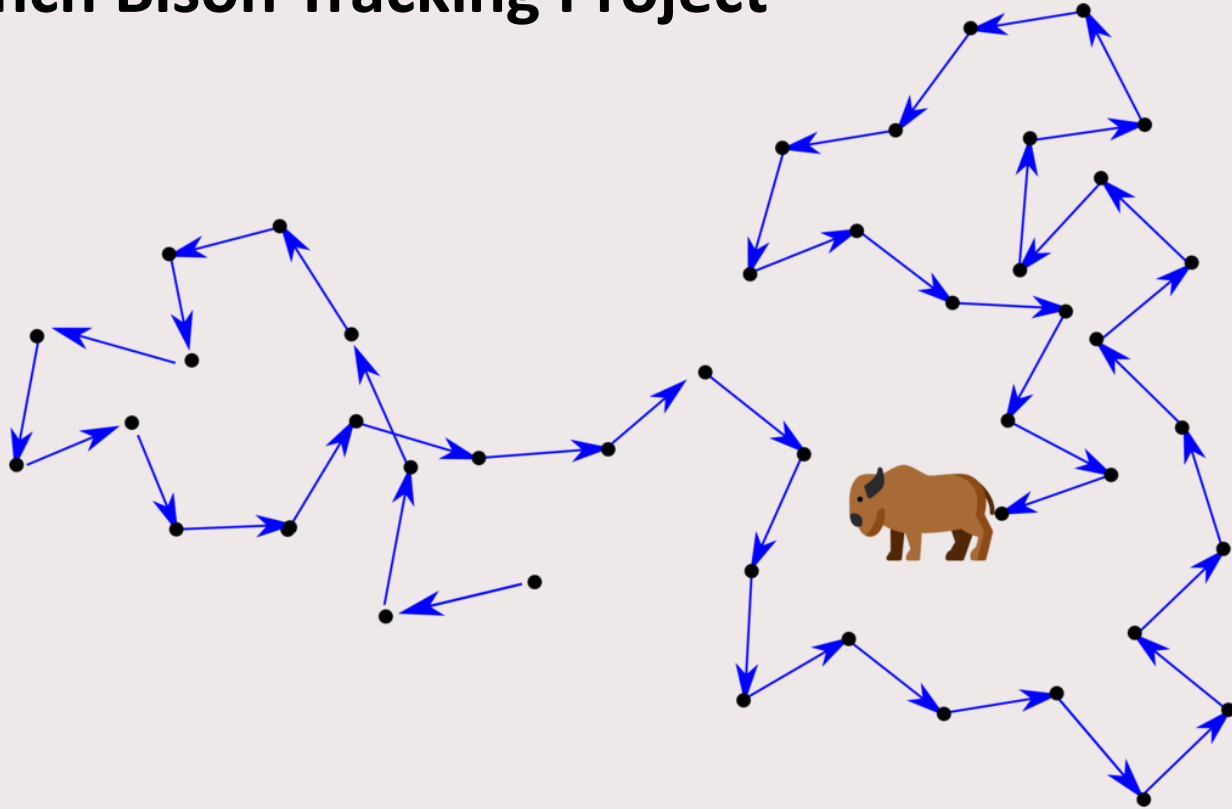




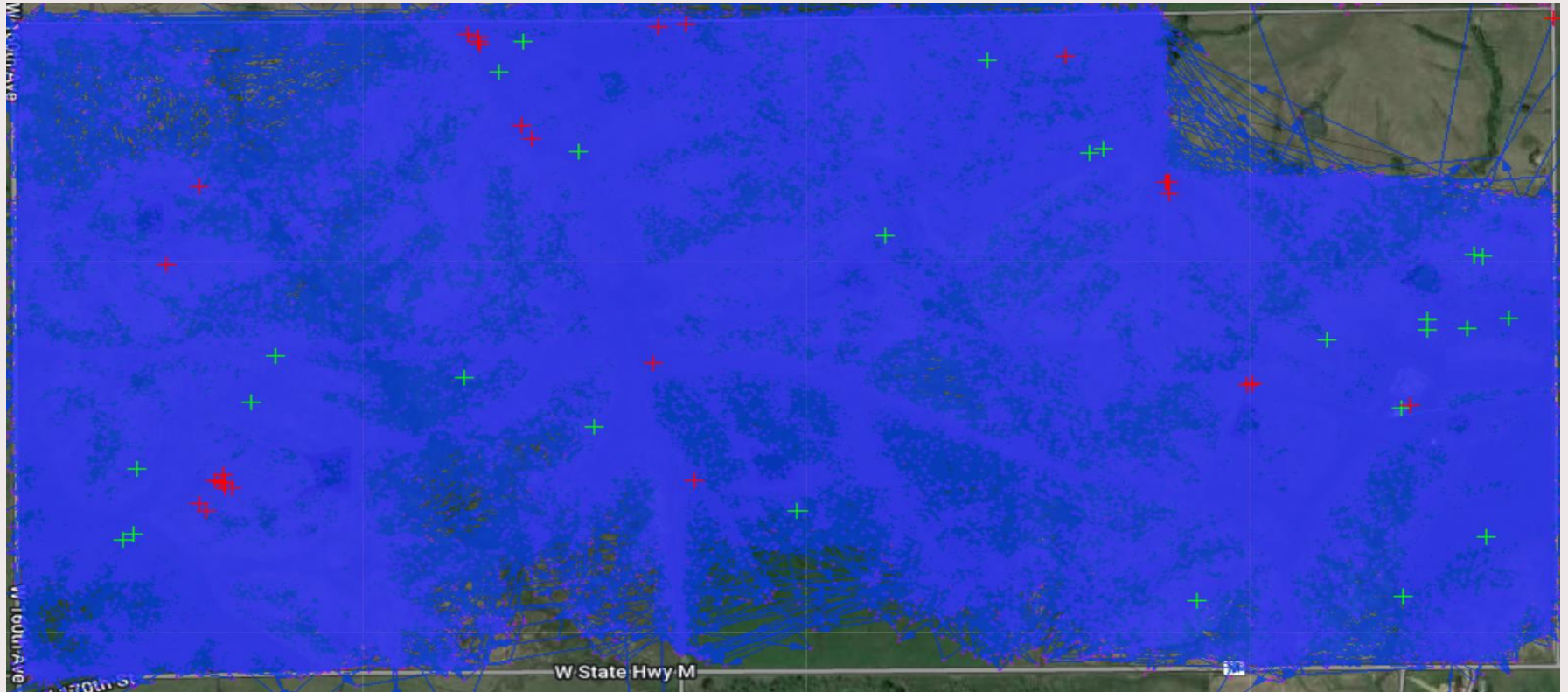
Detection and evaluation of clusters within animal movements

Alexander Van Werde, presented Stochastic Models in Life Science (Eurandom, 2023)

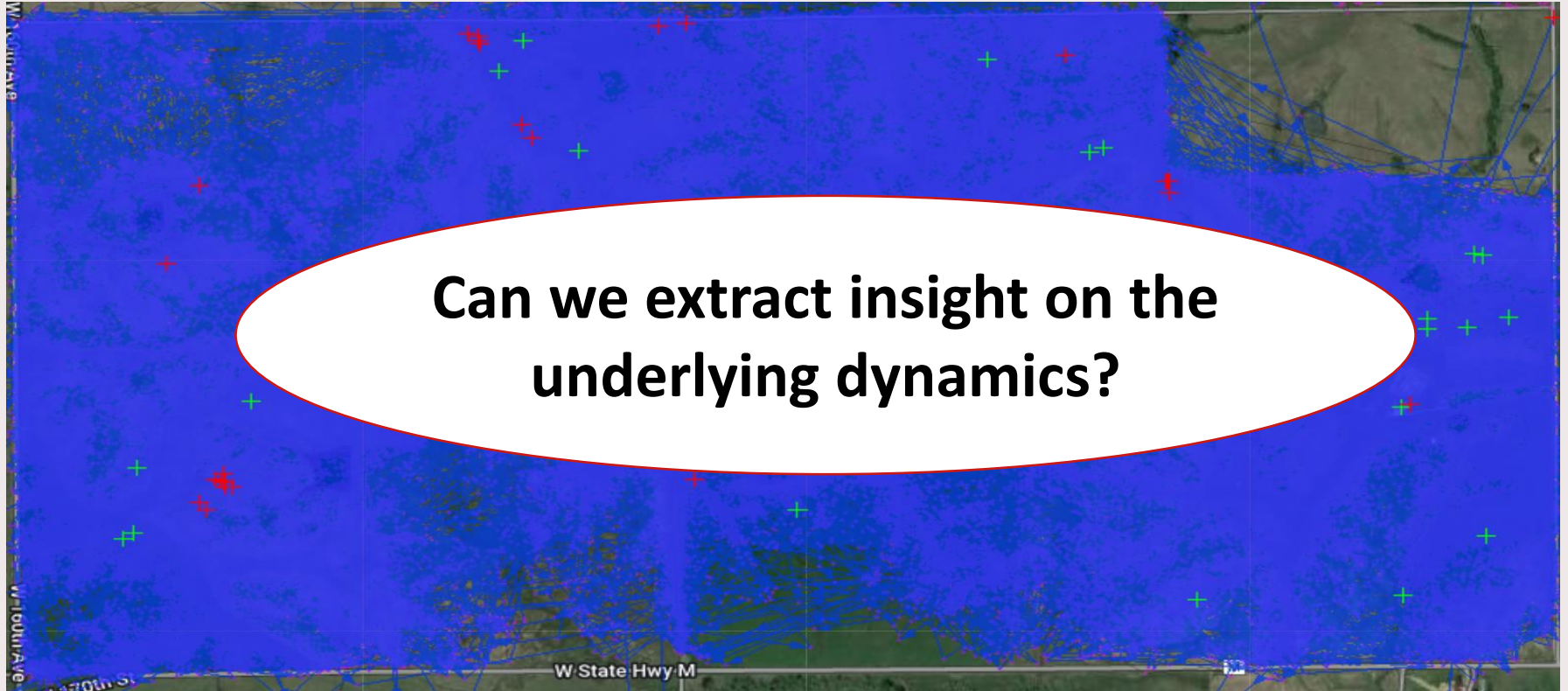
Dunn Ranch Bison Tracking Project



Direct visualization is not sufficient



Direct visualization is not sufficient

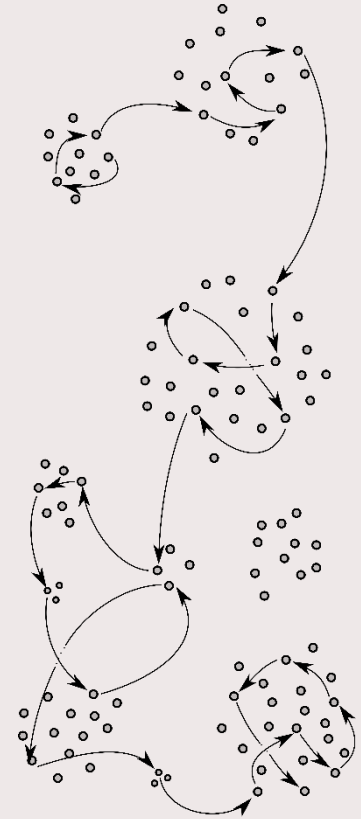


Extracting insight from sequential data

Sequential data can be modelled as a Markov chain.

Typically, the state spaces are large which causes many difficulties:

- Human interpretation becomes difficult
- Overfitting may occur
- Algorithms slow down



Extracting insight from sequential data

Sequential data can be modelled as a Markov chain.

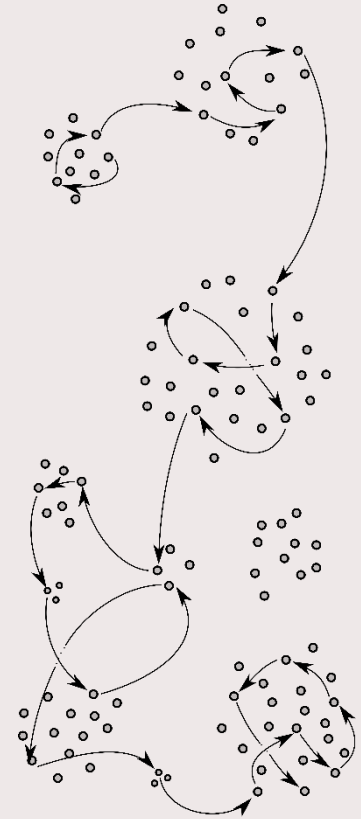
Typically, the state spaces are large which causes many difficulties:

- Human interpretation becomes difficult
- Overfitting may occur
- Algorithms slow down

Also relevant in:

- Natural language processing
- Genomics
- Reinforcement learning

Can one identify a lower-dimensional structure?

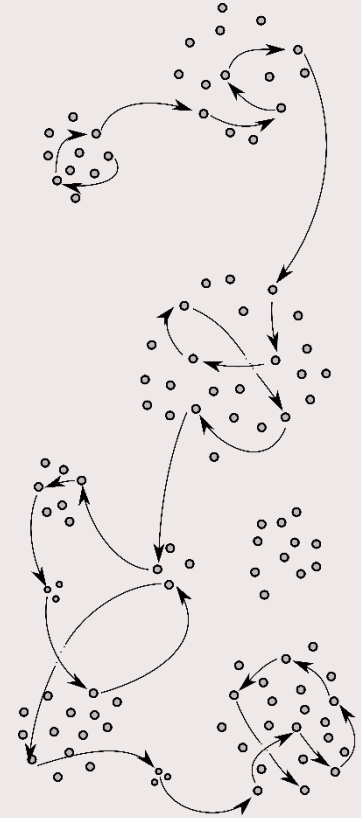


Problem statement

Given sequential data:

$$X_0 \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow \dots \rightarrow X_\ell$$

living on some large (but finite) space $\{1, \dots, n\}$.



Problem statement

Given sequential data:

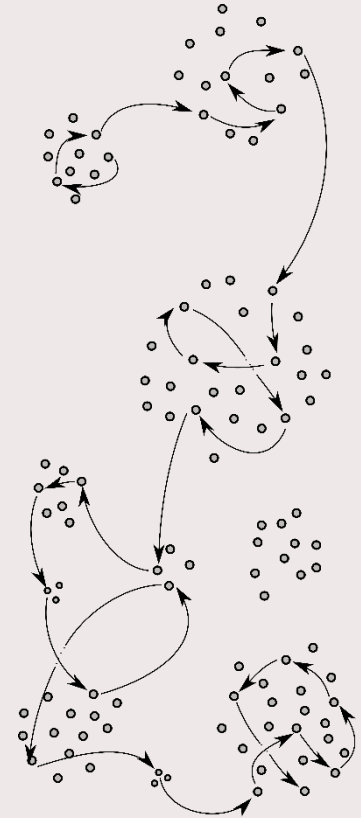
$$X_0 \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow \dots \rightarrow X_\ell$$

living on some large (but finite) space $\{1, \dots, n\}$.

Goal. Find a map $\sigma: \{1, \dots, n\} \rightarrow \{1, \dots, K\}$ with $K \ll n$ such that

$$\sigma(X_0) \rightarrow \sigma(X_1) \rightarrow \sigma(X_2) \rightarrow \dots \rightarrow \sigma(X_\ell)$$

captures most of the dynamics of the process.



Problem statement

Given sequential data:

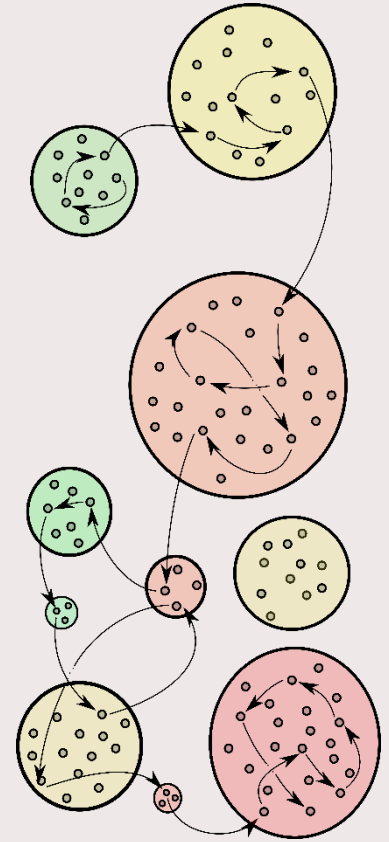
$$X_0 \rightarrow X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow \dots \rightarrow X_\ell$$

living on some large (but finite) space $\{1, \dots, n\}$.

Goal. Find a map $\sigma: \{1, \dots, n\} \rightarrow \{1, \dots, K\}$ with $K \ll n$ such that

$$\sigma(X_0) \rightarrow \sigma(X_1) \rightarrow \sigma(X_2) \rightarrow \dots \rightarrow \sigma(X_\ell)$$

captures most of the dynamics of the process.



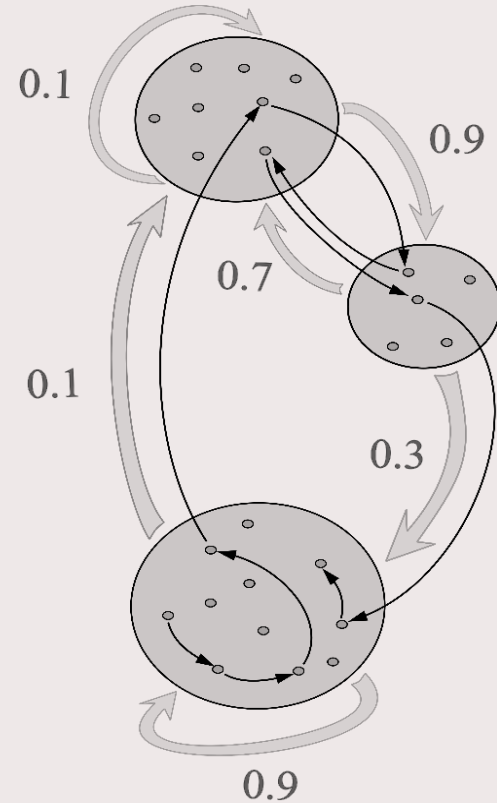
Idealized case: block Markov chains

Fix the following data:

- Positive integer $K \geq 1$
- Stochastic matrix $p \in [0,1]^{K \times K}$
- Partition $\mathcal{V}_1, \dots, \mathcal{V}_K$ for $\mathcal{V} := \{1, \dots, n\}$.

Then, the associated **block Markov chain** is the Markov chain on \mathcal{V} with transition matrix

$$P_{i,j} = \frac{p_{x,y}}{\#\mathcal{V}_y} \quad \forall i \in \mathcal{V}_x, \forall j \in \mathcal{V}_y.$$



Theory on block Markov chains

Clustering algorithms

- **J. Sanders, A. Proutière, S.-Y. Yun (2020)**
Two-step clustering algorithm and information-theoretic limits.
- **A. Zhang, M. Wang (2020)**
Spectral clustering algorithm.

Random matrix-theoretic properties

- **J. Sanders, A. Van Werde (2023)**
Singular value distributions
- **J. Sanders, A. Senen-Cerda (2023)**
Spectral norm bounds
- **A. Van Werde, J. Sanders (2023)**
Sharp concentration inequalities

Reinforcement learning

- **Y. Jedra, J. Lee, A. Proutière (2023)**
Block Markov decision processes

Theory on block Markov chains

Clustering algorithms

- **J. Sanders, A. Proutière, S.-Y. Yun (2020)**
Two-step clustering algorithm and information-theoretic limits.
- **A. Zhang, M. Wang (2020)**
Spectral clustering algorithm.

Random matrix-theoretic properties

- **J. Sanders, A. Van Werde (2023)**
Singular value distributions
- **J. Sanders, A. Senen-Cerda (2023)**
Spectral norm bounds
- **A. Van Werde, J. Sanders (2023)**
Sharp concentration inequalities

Reinforcement learning

- **Y. Jedra, J. Lee, A. Proutière (2023)**
Block Markov decision processes



1. Spectral initial guess
2. Greedy improvements

Theory on block Markov chains

Clustering algorithms

- **J. Sanders, A. Proutière, S.-Y. Yun (2020)**
Two-step clustering algorithm and information-theoretic limits.
- **A. Zhang, M. Wang (2020)**
Spectral clustering algorithm.

Random matrix-theoretic properties

- **J. Sanders, A. Van Werde (2023)**
Singular value distributions
- **J. Sanders, A. Senen-Cerda (2023)**
Spectral norm bounds
- **A. Van Werde, J. Sanders (2023)**
Sharp concentration inequalities

Reinforcement learning

- **Y. Jedra, J. Lee, A. Proutière (2023)**
Block Markov decision processes



1. Spectral initial guess
2. Greedy improvements

Theory on block Markov chains

Clustering algorithms

- **J. Sanders, A. Proutière, S.-Y. Yun (2020)**
Two-step clustering algorithm and information-theoretic limits.
- **A. Zhang, M. Wang (2020)**
Spectral clustering algorithm.

Random matrix-theoretic properties

- **J. Sanders, A. Van Werde (2023)**
Singular value distributions
- **J. Sanders, A. Senen-Cerda (2023)**
Spectral norm bounds
- **A. Van Werde, J. Sanders (2023)**
Sharp concentration inequalities

Reinforcement learning

- **Y. Jedra, J. Lee, A. Proutière (2023)**
Block Markov decision processes



1. Spectral initial guess
2. Greedy improvements

Theory on block Markov chains

Clustering algorithms

- **J. Sanders, A. Proutière, S.-Y. Yun (2020)**
Two-step clustering algorithm and information-theoretic limits.
- **A. Zhang, M. Wang (2020)**
Spectral clustering algorithm.

Random matrix-theoretic properties

- **J. Sanders, A. Van Werde (2023)**
Singular value distributions
- **J. Sanders, A. Senen-Cerda (2023)**
Spectral norm bounds
- **A. Van Werde, J. Sanders (2023)**
Sharp concentration inequalities

Reinforcement learning

- **Y. Jedra, J. Lee, A. Proutière (2023)**
Block Markov decision processes



1. Spectral initial guess
2. Greedy improvements



**Inspired by stochastic
block model**

What about the real world?

Does the algorithm work?

Is the model appropriate?

What insights can be found?

Our contributions

Efficient implementation of the algorithm

Python module with C++ implementation

```
pip install BMCToolkit
```

Identification and preprocessing of datasets

Sequential data demonstrating relevance to a variety of fields.

(Ethology, natural language processing, microbiology, finance)

Model evaluation toolset

Classical as well as new evaluation tools.

For instance: spectral noise evaluation

Our contributions

Efficient implementation of the algorithm

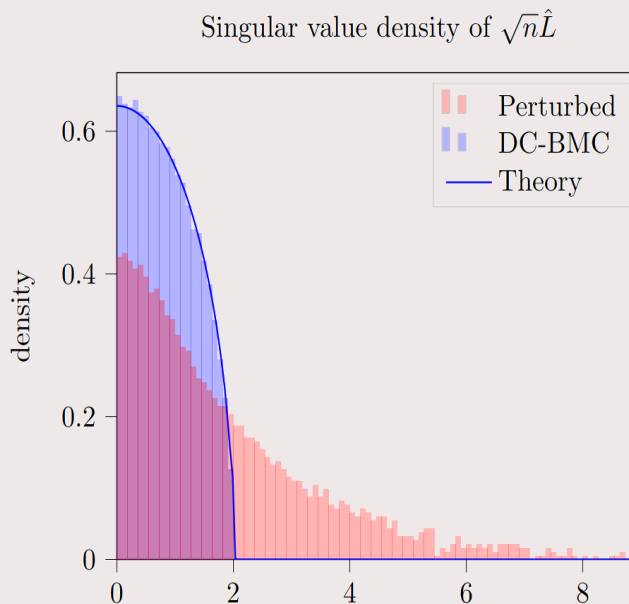
Python module with C++ implementation
`pip install BMCToolkit`

Identification and preprocessing of datasets

Sequential data demonstrating relevance to a variety of fields.
(Ethology, natural language processing, microbiology, finance)

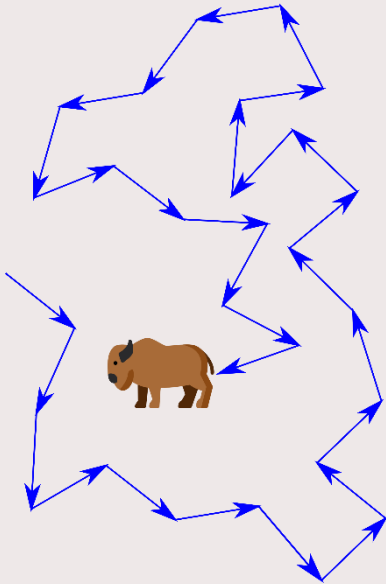
Model evaluation toolset

Classical as well as new evaluation tools.
For instance: spectral noise evaluation

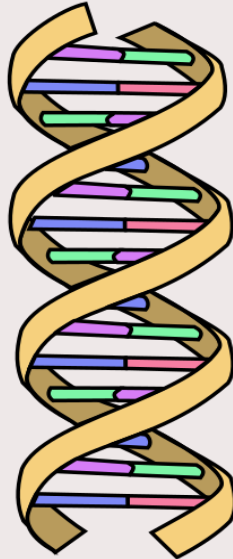


Sequential datasets

Animal movement



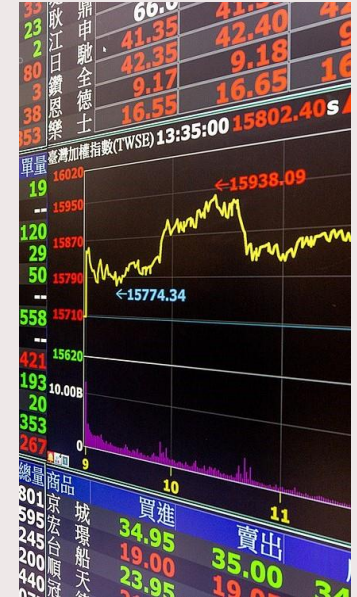
DNA



Text

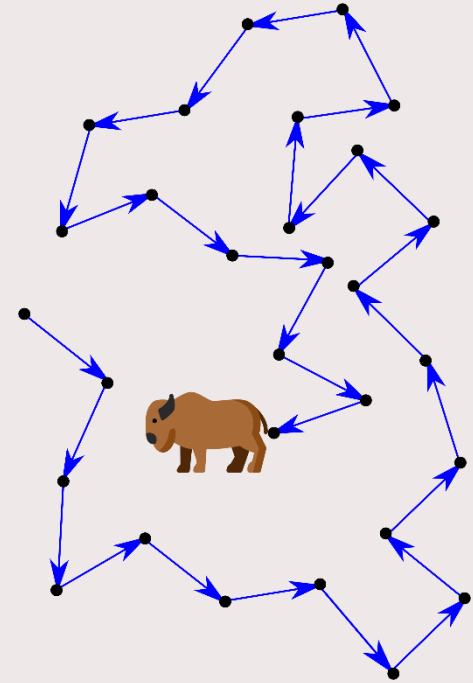
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas in turpis sit amet risus ultricies tincidunt vitae accumsan enim. Pellentesque ac efficitur enim. Praesent dui magna, mattis at bibendum eu, fringilla eu neque. Nulla vel accumsan arcu. Quisque sagittis, massa quis maximus vestibulum, sapien ligula venenatis ex, eu elementum diam ante a neque. Nunc egestas malesuada interdum. Morbi eget sem in neque pretium vehicula. Duis nulla turpis, efficitur sed lacus at, auctor tempor lorem. Maecenas pharetra et erat non viverra. Morbi pellentesque velit nec risus imperdiet, ac fermentum lectus aliquam. Maecenas rutrum nisi ut turpis ultricies, id placerat quam fringilla. Donec non interdum urna. Nam justo mi, malesuada eu sollicitudin vitae, euismod vitae tortor. Donec feugiat ligula sed sem lectus consectetur id non lectus. Donec ac neque sed metus facilisis rhoncus. Pellentesque iaculis, urna et consequat venenatis, nulla felis laoreet metus, vitae porta risus neque non odio. Sed venenatis mauris magna, sed tincidunt enim suscipit vitae. Donec a commodo ipsum. Proin euismod lacus ac metus finibus sagittis. Duis congue lorem quis velit tempus tincidunt. Ut ultrices rhoncus ipsum, eget aliquam ex dapibus et. Mauris porta, quam sit amet tincidunt aliquet, ex ante tincidunt nibh, et ultricies mi diam ac odio. Duis viverra velit ut dolor ornare ultricies. Phasellus non interdum velit, ac egestas ligula. Cras magna odio, vestibulum in ex ut, scelerisque dapibus tellus. Curabitur facilisis nisl quis erat lobortis efficitur. Donec at ullamcorper nulla, ut eleifend purus. Donec euismod odio quis dui ultricies efficitur. Quisque lobortis lectus tortor, in interdum lorem tempor dictum. Curabitur vitae libero sed ex elementum fringilla. Duis vel sagittis nisl. Donec a lectus consectetur quam tincidunt mollis venenatis eget purus. Mauris condimentum id dolor quis rutrum. Etiam egestas sapien sit amet interdum tincidunt. Nam a viverra risus. Nulla in auctor massa, ac lobortis nunc. Nam maximus efficitur purus a laoreet. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos ...

Stock market



Dataset: animal movement data

Dunn ranch bison tracking project: (S. Blake, R. Arndt, D. Ladd)
GPS coordinates of bison over time.
(latitude, longitude, timestamp)



Dataset: animal movement data

Dunn ranch bison tracking project: (S. Blake, R. Arndt, D. Ladd)

GPS coordinates of bison over time.

(latitude, longitude, timestamp)

Preprocessing:

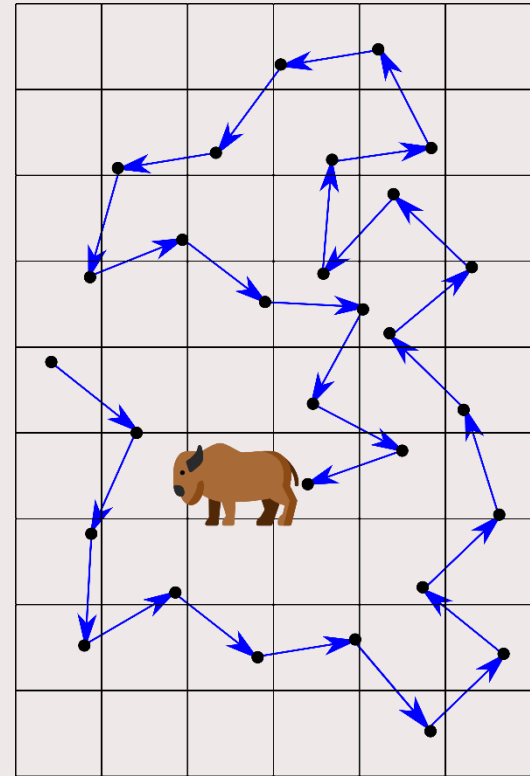
- 1) Discretize using a grid with bins of 40x40 meter.
- 2) Disregard self transitions.
- 3) Prune outliers caused by noisy GPS data.

Resulting dataset:

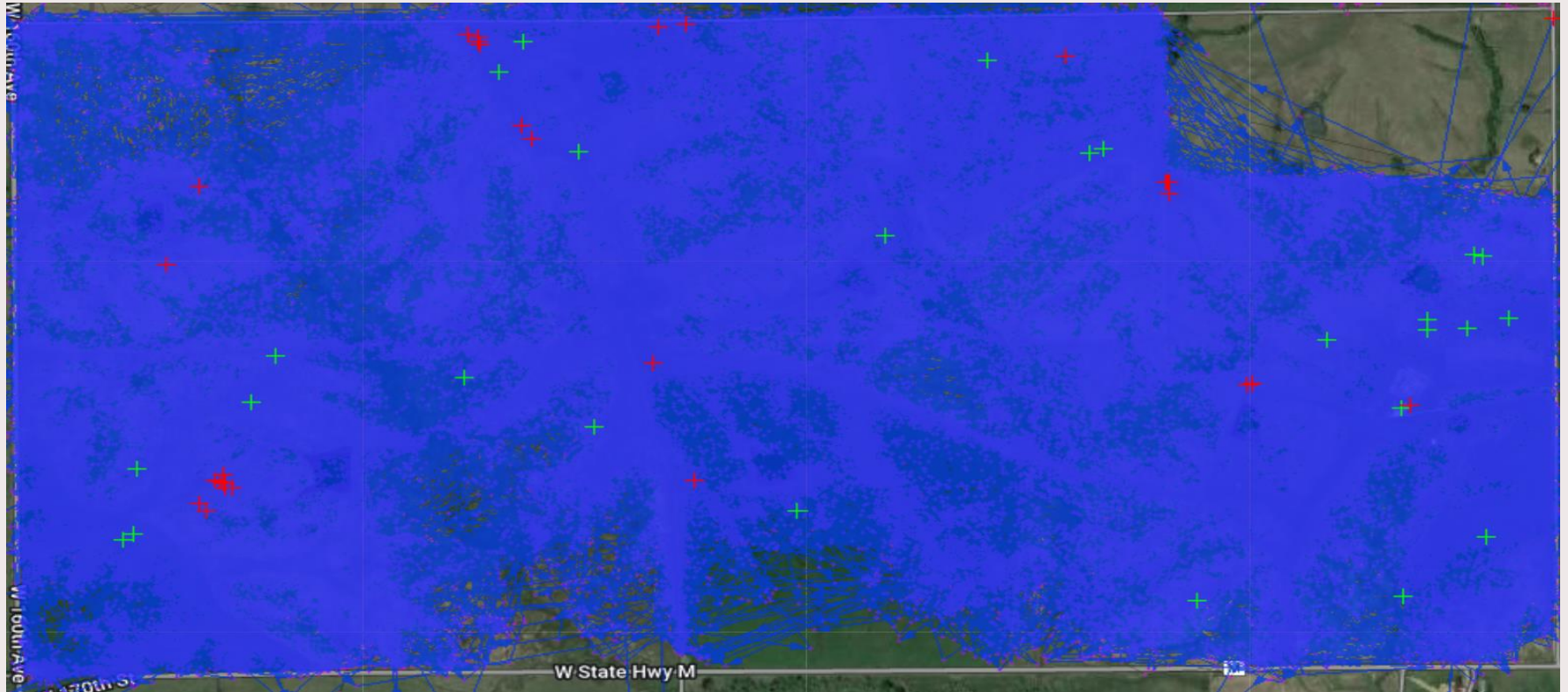
Number of states: $n = 3155$

Sample path length: $\ell \approx 2 \times 10^5$

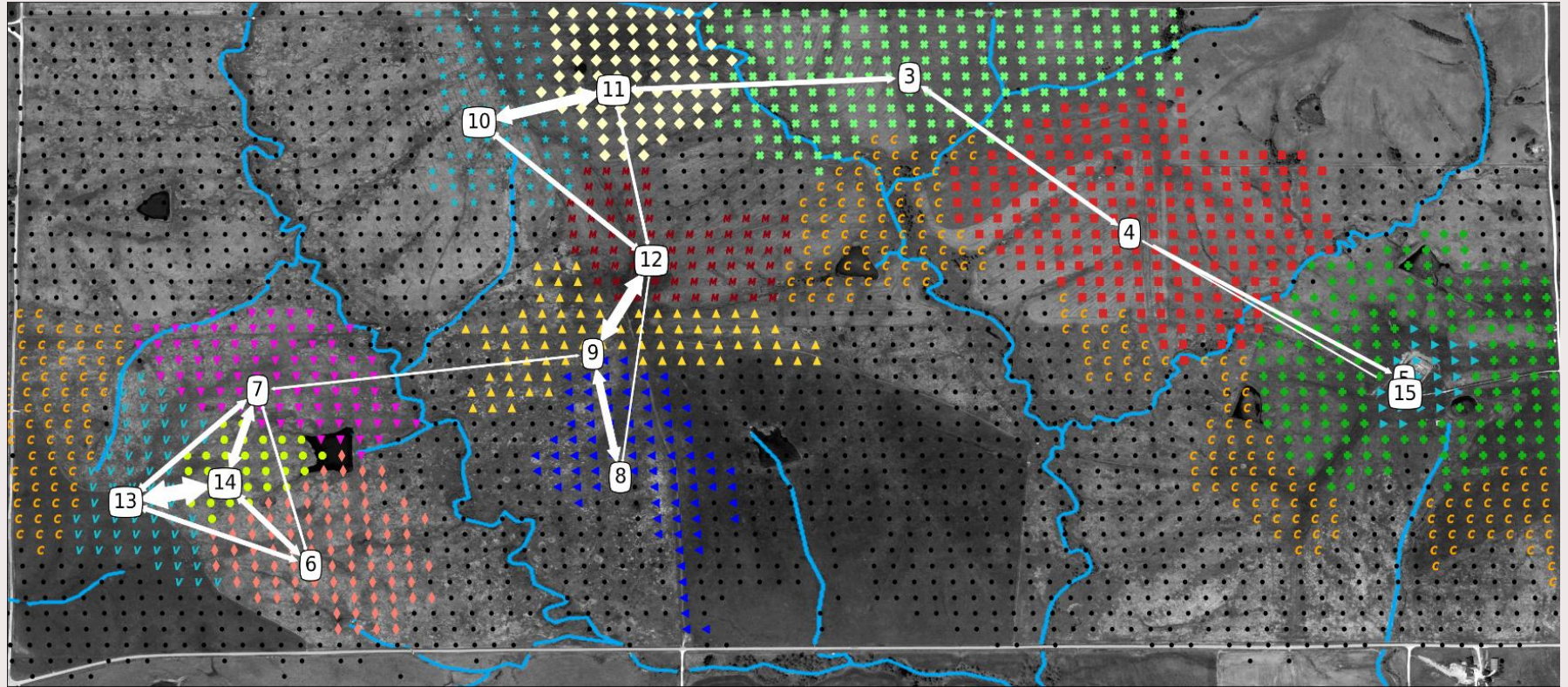
Sparsity: $\ell/n^2 \approx 0.019$



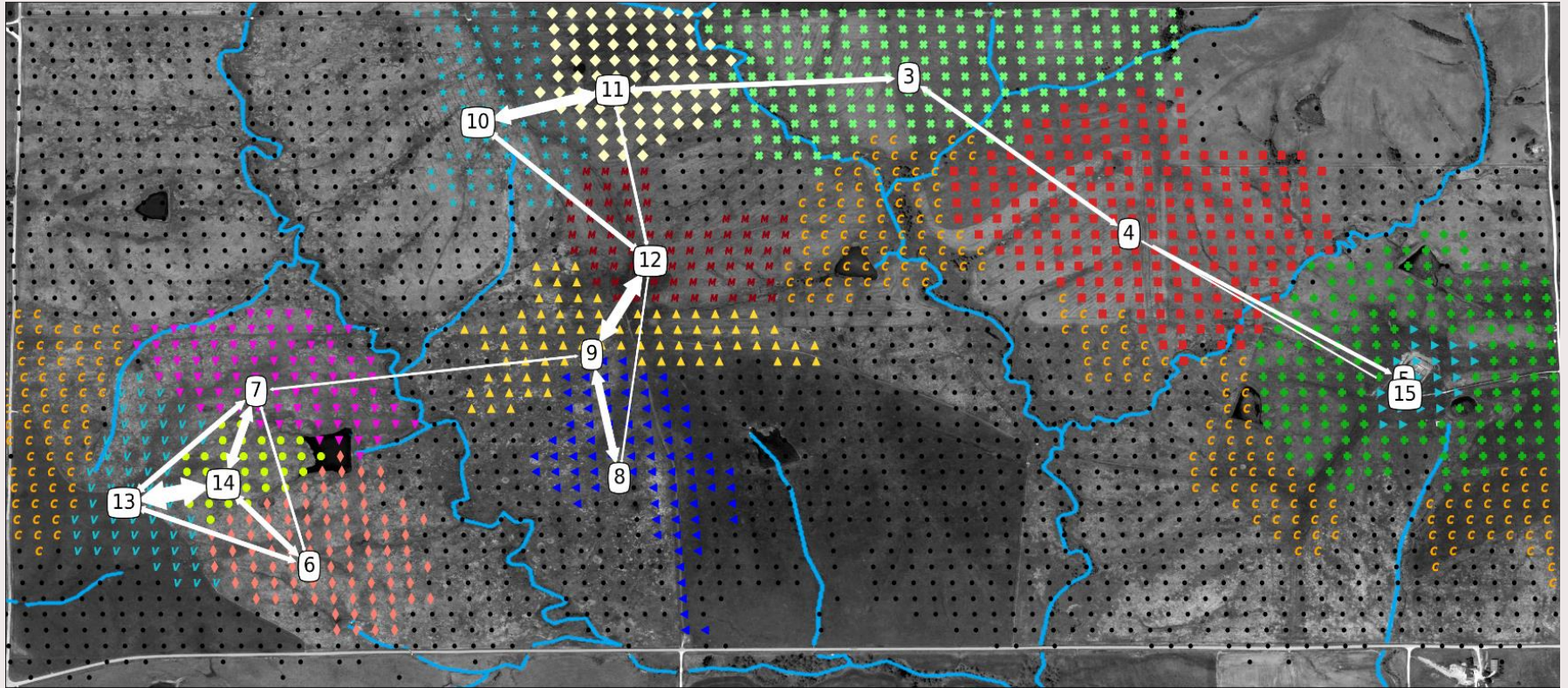
Clusters in animal movements



Clusters in animal movements



Clusters in animal movements Global topology of animal dynamics!

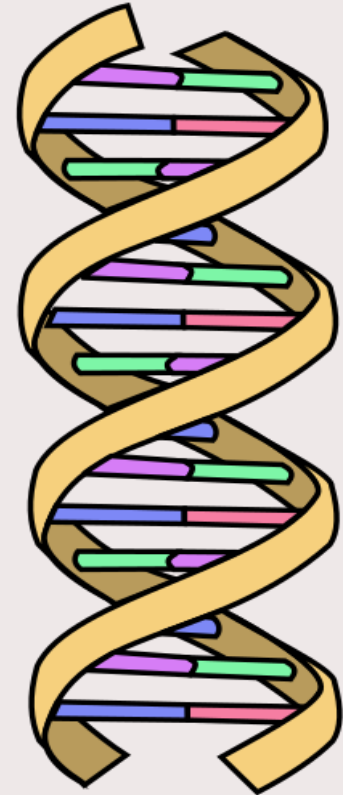


Dataset: DNA

Human gene OCA2:

Sequence of nucleotides:

... GTAGTTAGATCTCCTCTATCC ...



Dataset: DNA

Human gene OCA2:

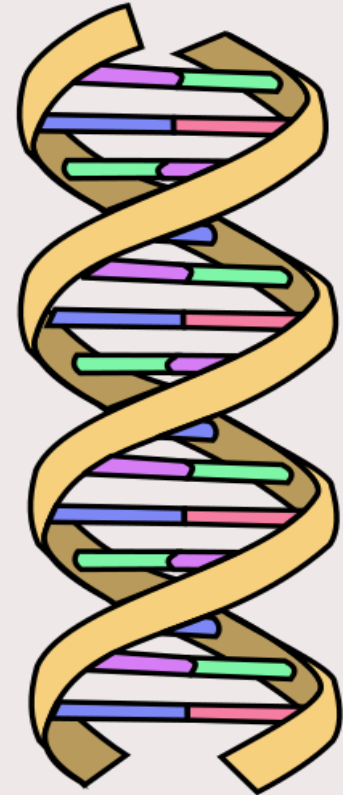
Sequence of nucleotides:

... GTAGTTAGATCTCCTCTATCC ...

Preprocessing:

Extract sequence of *codons*. (Three-letter words)

... → GTA → GTT → AGA → TCT → CCT → CTA → ...



Dataset: DNA

Human gene OCA2:

Sequence of nucleotides:

... GTAGTTAGATCTCCTCTATCC ...

Preprocessing:

Extract sequence of *codons*. (Three-letter words)

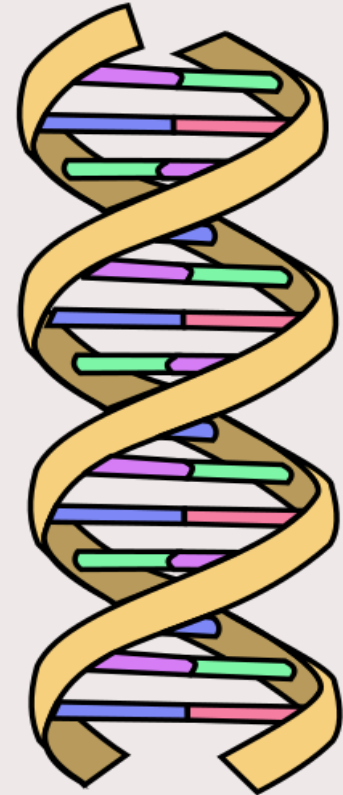
... → GTA → GTT → AGA → TCT → CCT → CTA → ...

Resulting dataset:

Number of states: $n = 64$

Sample path length: $\ell \approx 16 \times 10^4$

Sparsity: $\ell/n^2 \approx 39$



Clusters in DNA

Clusters:

$\mathcal{V}_1 := \{AAA, \dots\}$

$\mathcal{V}_2 := \{CAC, GCC, CCC, TCC, ACC, GTC, CTC, TTC, ATC, TGC, AGC, TAC, AAC, GGC, TAG, CTA, GAC\}$

$\mathcal{V}_3 := \{GTG, GAG, GGT, GCA, GAA, GTA, GGA, GAT, GGG, GTT, GCT\}$

$\mathcal{V}_4 := \{CGA, CGC, ACG, TCG, CCG, GCG, CGT, CGG\}$

$\mathcal{V}_5 := \{TTT\}$

Clusters in DNA

Clusters:

$\mathcal{V}_1 := \{AAA, \dots\}$

$\mathcal{V}_2 := \{CAC, GCC, CCC, TCC, ACC, GTC, CTC, TTC, ATC, TGC, AGC, TAC, AAC, GGC, TAG, CTA, GAC\}$

$\mathcal{V}_3 := \{GTG, GAG, GGT, GCA, GAA, GTA, GGA, GAT, GGG, GTT, GCT\}$

$\mathcal{V}_4 := \{CGA, CGC, ACG, TCG, CCG, GCG, CGT, CGG\}$

$\mathcal{V}_5 := \{TTT\}$

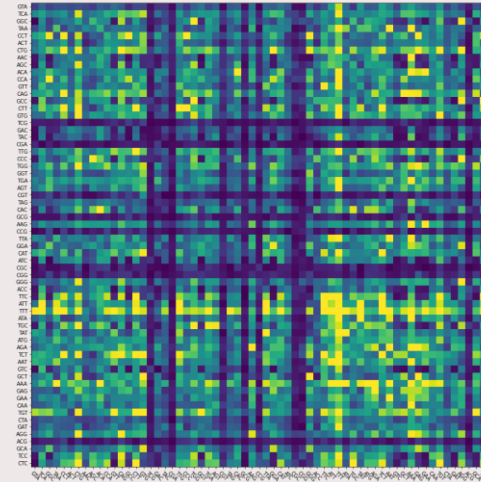
Observation: (codon-pair bias)

Only rarely $\mathcal{V}_2 \rightarrow \mathcal{V}_3$.

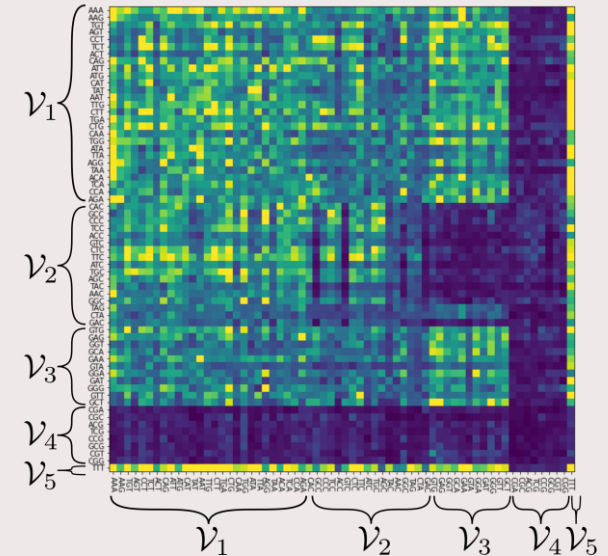
Codons ending with “C” rarely transition to codons starting with “G”.

Sample frequency matrix: $\hat{N}_{i,j} := \#\{\text{transitions } i \rightarrow j\}$

Before clustering:



After clustering:



Clusters in DNA

Clusters:

$\mathcal{V}_1 := \{AAA, \dots\}$

$\mathcal{V}_2 := \{CAC, GCC, CCC, TCC, ACC, GTC, CTC, TTC, ATC, TGC, AGC, TAC, AAC, GGC, TAG, CTA, GAC\}$

$\mathcal{V}_3 := \{GTG, GAG, GGT, GCA, GAA, GTA, GGA, GAT, GGG, GTT, GCT\}$

$\mathcal{V}_4 := \{CGA, CGC, ACG, TCG, CCG, GCG, CGT, CGG\}$

$\mathcal{V}_5 := \{TTT\}$

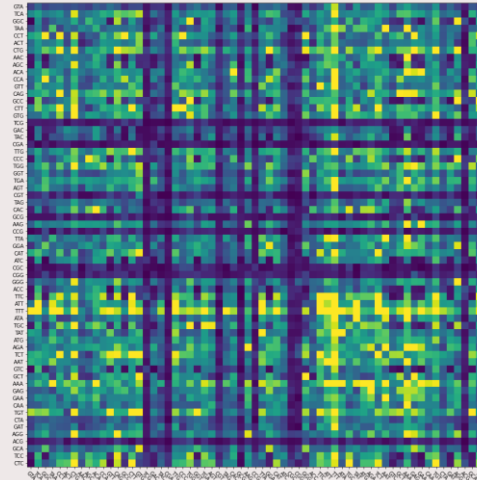
Observation: (codon-pair bias)

Only rarely $\mathcal{V}_2 \rightarrow \mathcal{V}_3$.

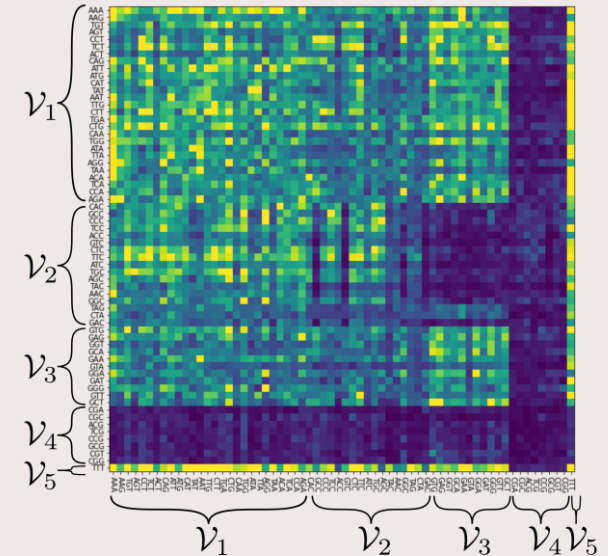
Codons ending with “C” rarely transition to codons starting with “G”.

Sample frequency matrix: $\hat{N}_{i,j} := \#\{\text{transitions } i \rightarrow j\}$

Before clustering:



After clustering:



Clusters in DNA

Rediscover biological phenomenon!

Clusters:

$\mathcal{V}_1 := \{AAA, \dots\}$

$\mathcal{V}_2 := \{CAC, GCC, CCC, TCC, ACC, GTC, CTC, TTC, ATC, TGC, AGC, TAC, AAC, GGC, TAG, CTA, GAC\}$

$\mathcal{V}_3 := \{GTG, GAG, GGT, GCA, GAA, GTA, GGA, GAT, GGG, GTT, GCT\}$

$\mathcal{V}_4 := \{CGA, CGC, ACG, TCG, CCG, GCG, CGT, CGG\}$

$\mathcal{V}_5 := \{TTT\}$

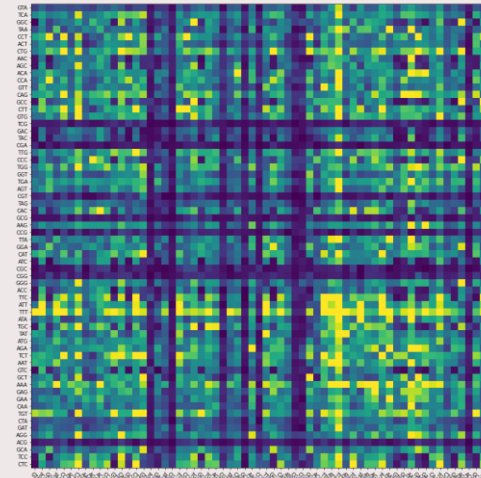
Observation: (codon-pair bias)

Only rarely $\mathcal{V}_2 \rightarrow \mathcal{V}_3$.

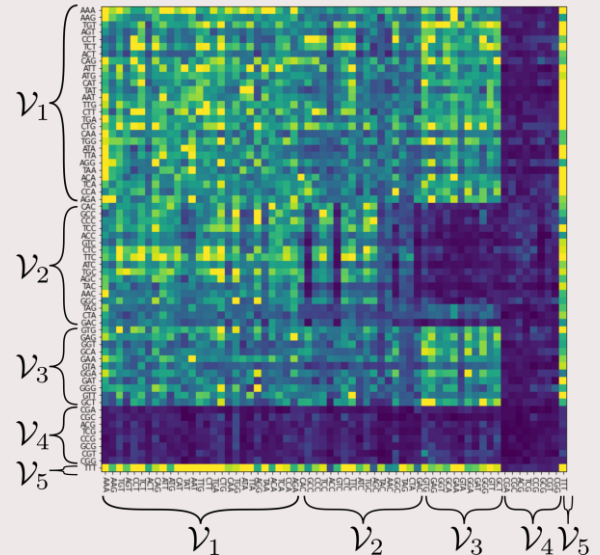
Codons ending with “C” rarely transition to codons starting with “G”.

Sample frequency matrix: $\hat{N}_{i,j} := \#\{\text{transitions } i \rightarrow j\}$

Before clustering:



After clustering:

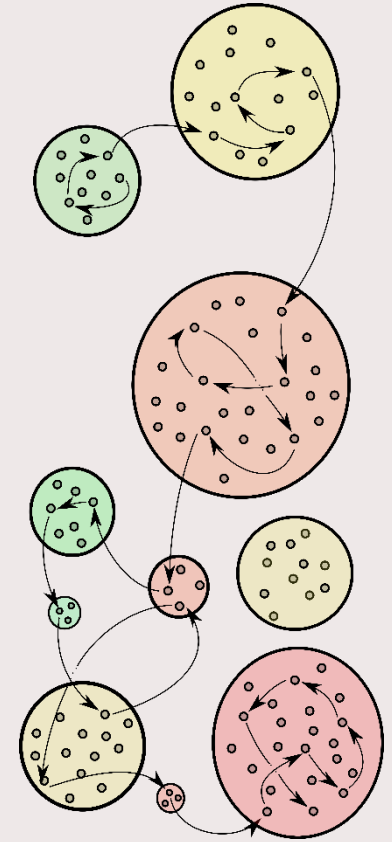


Model evaluation using spectral noise

Evaluation of clusters is non-trivial

In real life, unlike theory or synthetic scenarios, we do not know a ground-truth clustering.

The classical quality measure: “*number of misclassified states*” hence does not make sense.



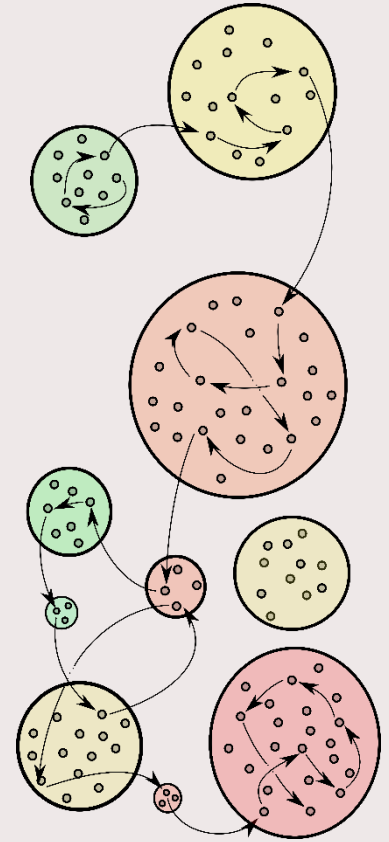
Evaluation of clusters is non-trivial

In real life, unlike theory or synthetic scenarios, we do not know a ground-truth clustering.

The classical quality measure: “*number of misclassified states*” hence does not make sense.

We develop and apply an evaluation toolset including:

1. Benchmarking on downstream tasks (e.g., document classification for text data.)
2. KL divergence on validation data
3. Consistent Akaike Information Criterion (CAIC)
4. Spectral noise evaluation



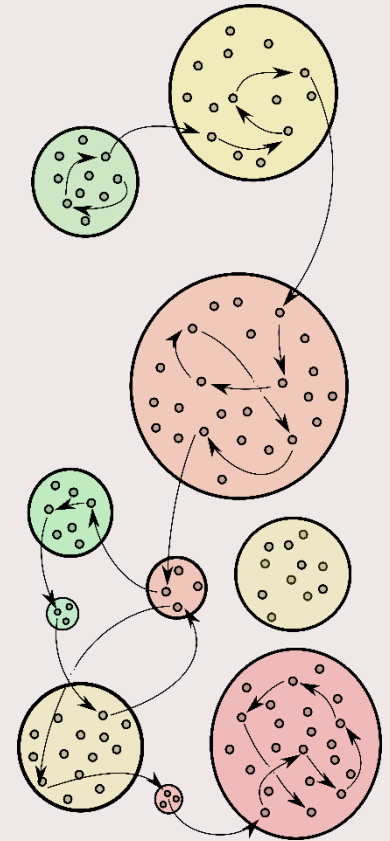
Evaluation of clusters is non-trivial

In real life, unlike theory or synthetic scenarios, we do not know a ground-truth clustering.

The classical quality measure: “*number of misclassified states*” hence does not make sense.

We develop and apply an evaluation toolset including:

1. Benchmarking on downstream tasks (e.g., document classification for text data.)
2. KL divergence on validation data
3. Consistent Akaike Information Criterion (CAIC)
4. **Spectral noise evaluation**



Spectral noise

Definition. The *sample frequency matrix* is the $n \times n$ matrix \hat{N} given by

$$\begin{aligned}\hat{N}_{i,j} &= \#\{\text{transitions } i \rightarrow j\} \\ &= \sum_{t=1}^{\ell} 1\{X_t = i, X_{t+1} = j\}\end{aligned}$$

Spectral noise

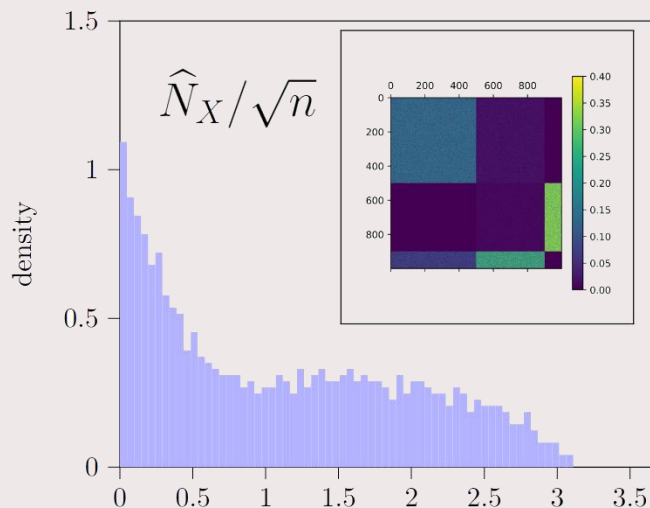
Definition. The *sample frequency matrix* is the $n \times n$ matrix \widehat{N} given by

$$\begin{aligned}\widehat{N}_{i,j} &= \#\{\text{transitions } i \rightarrow j\} \\ &= \sum_{t=1}^{\ell} 1\{X_t = i, X_{t+1} = j\}\end{aligned}$$

Theorem. (J. Sanders, A. Van Werde, 2023)

Assume that $\ell = \Theta(n^2)$. Then, the histogram of singular values of $\sqrt{n/\ell}\widehat{N}$ has a limit as $n \rightarrow \infty$.

The limit can be computed in terms of the parameters of the block Markov chain.



Spectral noise

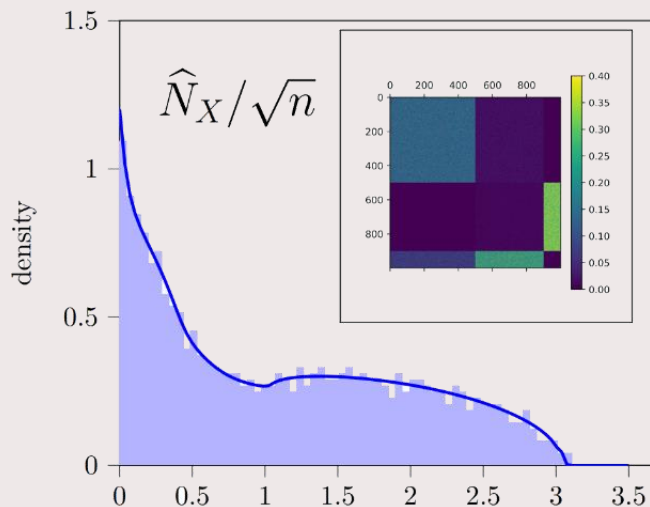
Definition. The *sample frequency matrix* is the $n \times n$ matrix \widehat{N} given by

$$\begin{aligned}\widehat{N}_{i,j} &= \#\{\text{transitions } i \rightarrow j\} \\ &= \sum_{t=1}^{\ell} 1\{X_t = i, X_{t+1} = j\}\end{aligned}$$

Theorem. (J. Sanders, A. Van Werde, 2023)

Assume that $\ell = \Theta(n^2)$. Then, the histogram of singular values of $\sqrt{n/\ell}\widehat{N}$ has a limit as $n \rightarrow \infty$.

The limit can be computed in terms of the parameters of the block Markov chain.

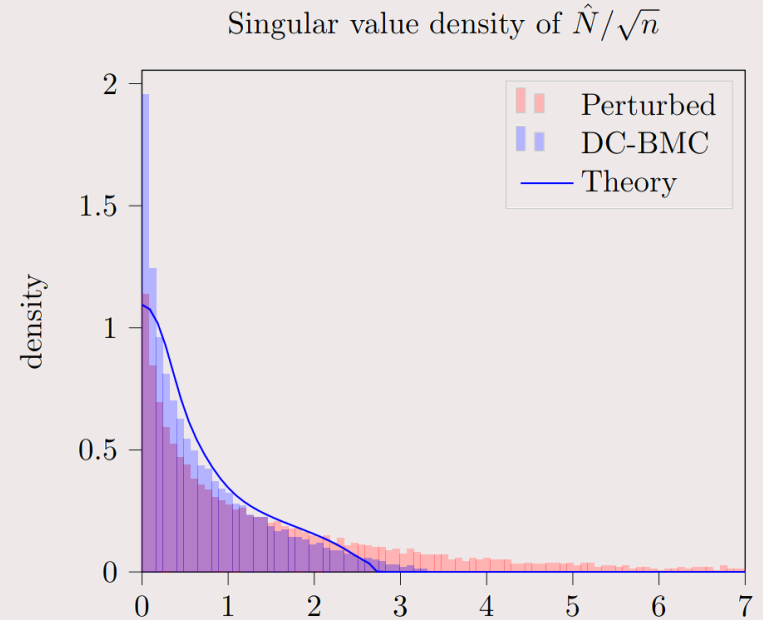


Empirically observed insensitivity

Observation. Real-world data often has a highly inhomogeneous equilibrium distribution.

The spectrum of \hat{N} tends to be dominated by this inhomogeneity.

This makes it insensitive to model violations.



Empirically observed insensitivity

Observation. Real-world data often has a highly inhomogeneous equilibrium distribution.

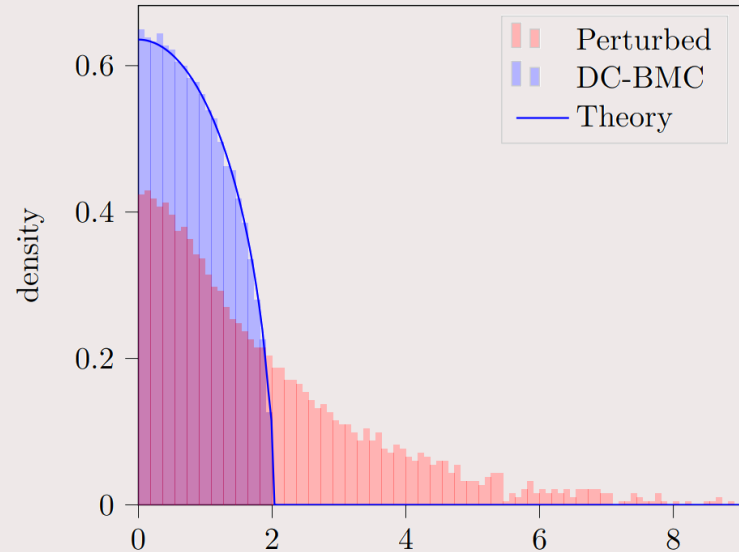
The spectrum of \hat{N} tends to be dominated by this inhomogeneity.

This makes it insensitive to model violations.

Solution: Consider *normalized Laplacian*.
That is, the matrix \hat{L} defined by

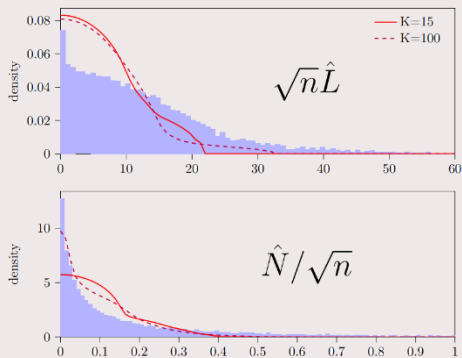
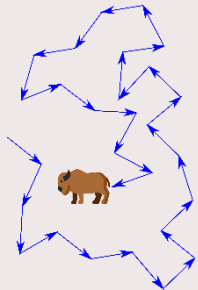
$$\hat{L}_{i,j} := \frac{\hat{N}_{i,j}}{\sqrt{\sum_{k=1}^n \hat{N}_{i,k}} \sqrt{\sum_{k=1}^n \hat{N}_{k,j}}}$$

Singular value density of $\sqrt{n}\hat{L}$

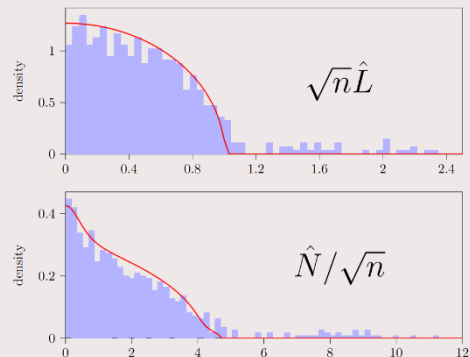
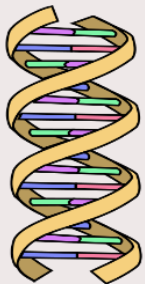


Spectral model evaluation

Animal movement

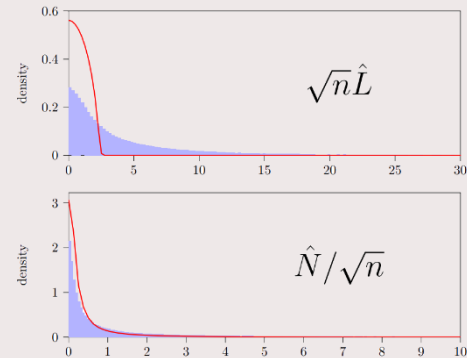


DNA

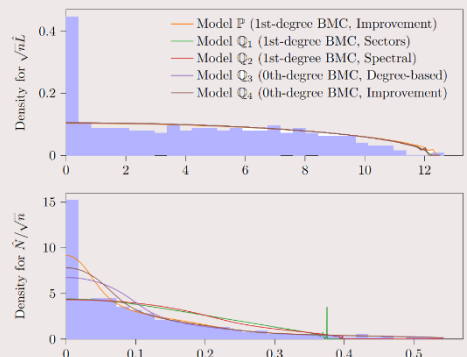


Text

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas sit turpis sit amet risus ultrices, in elementum enim. Pellentesque ac efficitur enim. Praesent dui magna, mattis at bibendum eu, fringilla eu, neque. Nulla vel accumsan arcu. Quisque sagittis, massa quis maximus vestibulum, sapien ligula venenatis, eu elementum diam ante a neque. Nunc egeta malesuada interdum. Morbi eget sem in neque pretium vehicula. Duis nulla turpis, efficitur sed lacus sit, acuter tempus lorem. Maecenas pharetra et erat non viverra. Morbi pellentesque velit nec risus imperdiet, ac fermentum lectus aliquam. Maecenas nunc nisi ut turpis ultrices, id pharetra quam fringilla. Donec non interdum urna. Nunc justo mi, malesuada eu sollicitudin vitae, euismod vitae tector. Donec feugiat ligula sed sem lectus, consectetur id non lectus. Donec ac neque sed metus facilisis rhoncus. Pellentesque iaculis, urna et congequa venenatis, nulla fella laoreet metus, vitae porta risus neque non odio. Sed venenatis mauris magna, sed interdum enim suscipit vitae. Donec a conmodo ipsum. Proin euismod lacus ac metus fames sagittis. Duis congue lorem quis velit tempus tincidunt. Ut ultrices rhoncus ipsum, eget aliquam ex dapibus et. Mauris porta, quam sit amet tincidunt aliquet, ex ante tincidunt nibh, et ultrices mi diam ac odio. Duis viverra velit ut dolor ornare ultrices. Phasellus non interdum velit, ac egestas ligula. Cras magna odio, vestibulum in ex ut, scelerisque dapibus tellus. Curabitur facilisis nisi quis erat lobortis efficitur. Donec at ullamcorper nulla, ut eberidit purus. Donec euismod odio quis dui ultrices efficitur. Quisque lobortis lectus tector, in interdum lorem tempus dicitur. Curabitur vitae libero sed ex elementum fringilla. Duis vel sagittis nisi...



Stock market



Now, let's see some proofs!

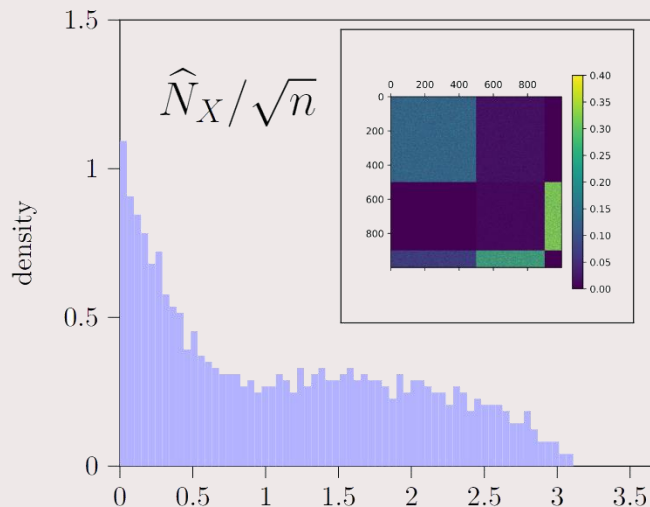
Definition. The *sample frequency matrix* is the $n \times n$ matrix \widehat{N} given by

$$\begin{aligned}\widehat{N}_{i,j} &= \#\{\text{transitions } i \rightarrow j\} \\ &= \sum_{t=1}^{\ell} 1\{X_t = i, X_{t+1} = j\}\end{aligned}$$

Theorem. (J. Sanders, A. Van Werde, 2023)

Assume that $\ell = \Theta(n^2)$. Then, the histogram of singular values of $\sqrt{n/\ell}\widehat{N}$ has a limit as $n \rightarrow \infty$.

The limit can be computed in terms of the parameters of the block Markov chain.



Now, let's see some proofs!

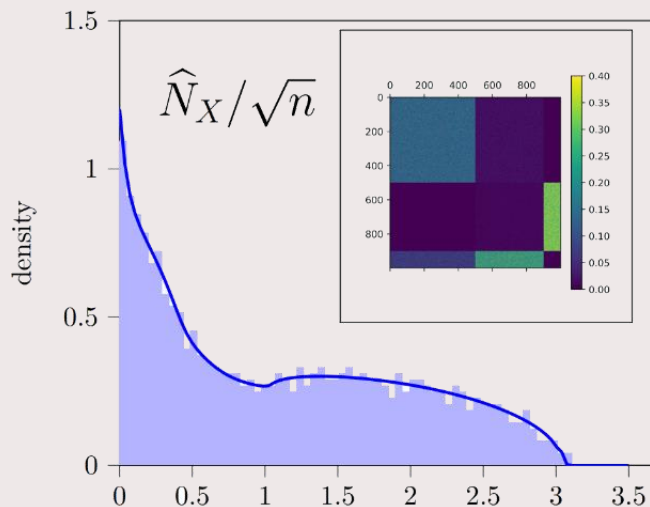
Definition. The *sample frequency matrix* is the $n \times n$ matrix \widehat{N} given by

$$\begin{aligned}\widehat{N}_{i,j} &= \#\{\text{transitions } i \rightarrow j\} \\ &= \sum_{t=1}^{\ell} 1\{X_t = i, X_{t+1} = j\}\end{aligned}$$

Theorem. (J. Sanders, A. Van Werde, 2023)

Assume that $\ell = \Theta(n^2)$. Then, the histogram of singular values of $\sqrt{n/\ell}\widehat{N}$ has a limit as $n \rightarrow \infty$.

The limit can be computed in terms of the parameters of the block Markov chain.



Now, let's see some proofs!

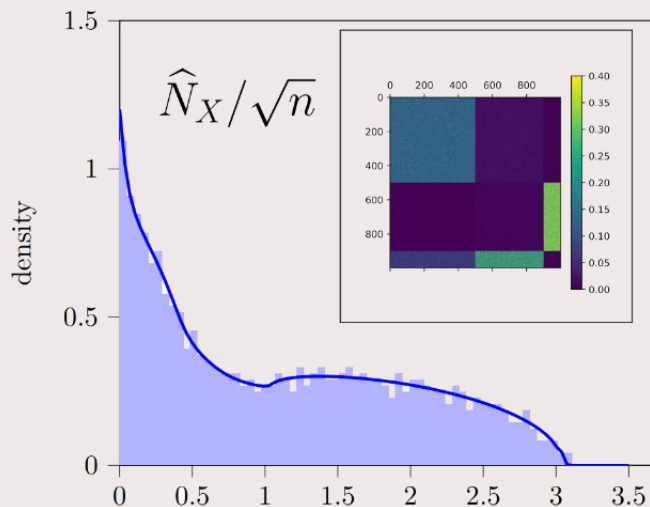
Definition. The *sample frequency matrix* is the $n \times n$ matrix \widehat{N} given by

$$\begin{aligned}\widehat{N}_{i,j} &= \#\{\text{transitions } i \rightarrow j\} \\ &= \sum_{t=1}^{\ell} 1\{X_t = i, X_{t+1} = j\}\end{aligned}$$

Theorem. (A. Van Werde, J. Sanders, 2023)

Assume that $\ell = \omega(n)$. Then, the histogram of singular values of $\sqrt{n/\ell}\widehat{N}$ has a limit as $n \rightarrow \infty$.

The limit can be computed in terms of the parameters of the block Markov chain.



Now, let's see some proofs!

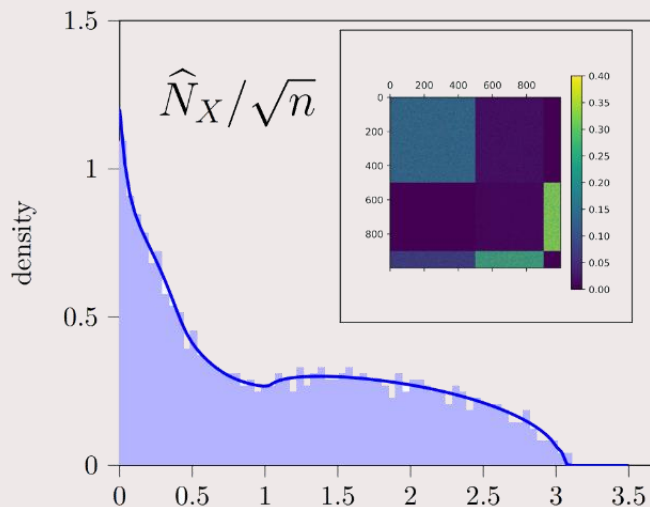
Definition. The *sample frequency matrix* is the $n \times n$ matrix \widehat{N} given by

$$\begin{aligned}\widehat{N}_{i,j} &= \#\{\text{transitions } i \rightarrow j\} \\ &= \sum_{t=1}^{\ell} 1\{X_t = i, X_{t+1} = j\}\end{aligned}$$

Theorem. (A. Van Werde, J. Sanders, 2023)

Assume that $\ell = \omega(n)$. Then, the histogram of singular values of $\sqrt{n/\ell}\widehat{N}$ has a limit as $n \rightarrow \infty$.

The limit can be computed in terms of the parameters of the block Markov chain.



Proof sketch 1: $\ell = \Theta(n^2)$ using coupling

Moment method: Compute $\mathbb{E} \left[\sigma_I(\widehat{\mathbf{N}})^{2k} \right] = \mathbb{E} [\text{Tr}(\widehat{\mathbf{N}}\widehat{\mathbf{N}}^T)^k]$

Main difficulty: Different from classical results, the entries of $\widehat{\mathbf{N}}_{i,j}$ are *dependent*.

We need to show, e.g., that $\mathbb{E} \left[\prod_{r=1}^{2k} \widehat{\mathbf{N}}_{i_r, j_r} \right] \approx \prod_{r=1}^{2k} \mathbb{E} \left[\widehat{\mathbf{N}}_{i_r, j_r} \right]$.

Proof sketch 1: $\ell = \Theta(n^2)$ using coupling

Moment method: Compute $\mathbb{E} \left[\sigma_I(\widehat{\mathbf{N}})^{2k} \right] = \mathbb{E}[\text{Tr}(\widehat{\mathbf{N}}\widehat{\mathbf{N}}^T)^k]$

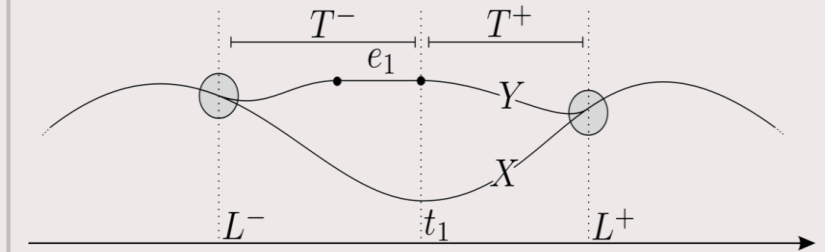
Main difficulty: Different from classical results, the entries of $\widehat{\mathbf{N}}_{i,j}$ are *dependent*.

We need to show, e.g., that $\mathbb{E}[\prod_{r=1}^{2k} \widehat{\mathbf{N}}_{i_r, j_r}] \approx \prod_{r=1}^{2k} \mathbb{E}[\widehat{\mathbf{N}}_{i_r, j_r}]$.

Proof idea. By expanding $\widehat{\mathbf{N}}_{i_1, j_1}$ into indicator variables, it is sufficient to show that

$$\mathbb{E} \left[\prod_{r=2}^{2k} \widehat{\mathbf{N}}_{i_r, j_r} \mid (X_t, X_{t+1}) \right] \approx \mathbb{E}[\prod_{r=2}^{2k} \widehat{\mathbf{N}}_{i_r, j_r}]$$

We use a *coupling argument* and construct a path Y , independent of (X_t, X_{t+1}) , such that $Y_{t'} = X_{t'}$ for most times $t' \neq t$.



Proof sketch 2: $\ell = \omega(n)$ using new general-purpose theory

Moment method: Compute $\mathbb{E} \left[\sigma_I(\hat{\mathbf{N}})^{2k} \right] = \mathbb{E} \left[\text{Tr}(\hat{\mathbf{N}}\hat{\mathbf{N}}^T)^k \right]$

New approach: Represent the matrix as a sum

$$\hat{\mathbf{N}} = \sum_{t=1}^{\ell} \mathbf{E}_t \quad \text{where} \quad (\mathbf{E}_t)_{i,j} = 1\{X_{t-1} = i, X_t = j\}$$

Proof sketch 2: $\ell = \omega(n)$ using new general-purpose theory

Moment method: Compute $\mathbb{E} \left[\sigma_I(\widehat{\mathbf{N}})^{2k} \right] = \mathbb{E} \left[\text{Tr}(\widehat{\mathbf{N}}\widehat{\mathbf{N}}^T)^k \right]$

New approach: Represent the matrix as a sum

$$\widehat{\mathbf{N}} = \sum_{t=1}^{\ell} \mathbf{E}_t \quad \text{where} \quad (\mathbf{E}_t)_{i,j} = 1\{X_{t-1} = i, X_t = j\}$$

Suppose that Z is a ψ -mixing Markov chain and generate a $n \times n$ random matrix as

$$\mathbf{S} = \sum_{t=1}^{\ell} f(Z_t) \quad \text{for a bounded and matrix-valued function } f$$

A *Gaussian model* satisfies $\mathbb{E}[\mathbf{G}] = \mathbb{E}[\mathbf{S}]$ and $\text{Cov}(\mathbf{G}) = \text{Cov}(\mathbf{S})$.

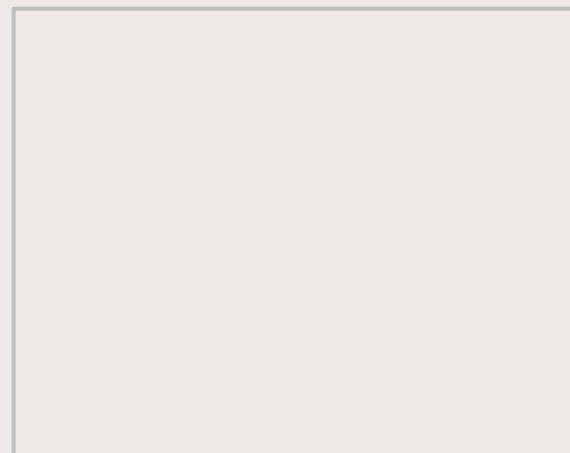
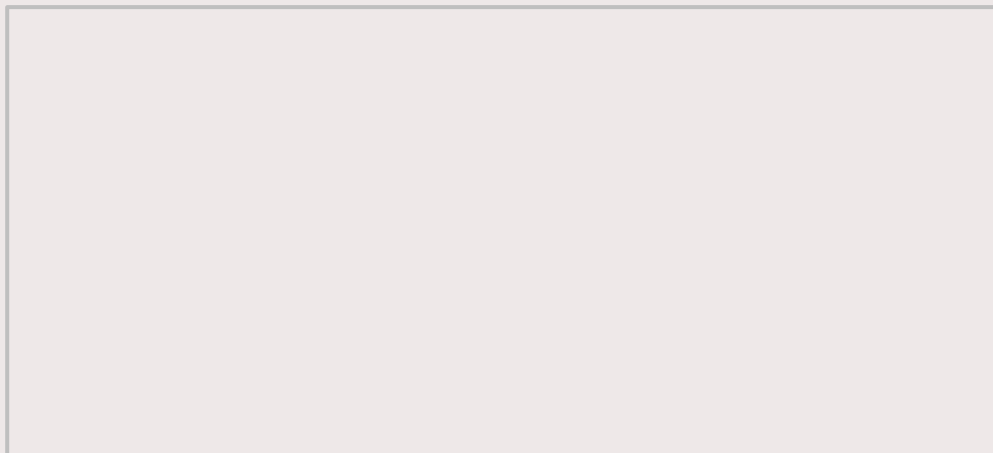
General-purpose theorem. (Informally)

For any fixed $k, n \geq 1$, one has that $\mathbb{E}[\text{Tr}(\mathbf{S}\mathbf{S}^T)^k] \approx \mathbb{E}[\text{Tr}(\mathbf{G}\mathbf{G}^T)^k]$.

Bonus result!

Context. The analysis of the spectral step in the clustering algorithms requires one to show that $\widehat{\mathbf{N}} \approx \mathbb{E}[\widehat{\mathbf{N}}]$.

Combining the moment universality with a free-probabilistic result of Bandeira, Boedihardjo, and van Handel, we can now state *exactly* how good the approximation is.



Bonus result!

Context. The analysis of the spectral step in the clustering algorithms requires one to show that $\widehat{\mathbf{N}} \approx \mathbb{E}[\widehat{\mathbf{N}}]$.

Combining the moment universality with a free-probabilistic result of Bandeira, Boedihardjo, and van Handel, we can now state *exactly* how good the approximation is.

Theorem. (A. Van Werde, J. Sanders, 2023)

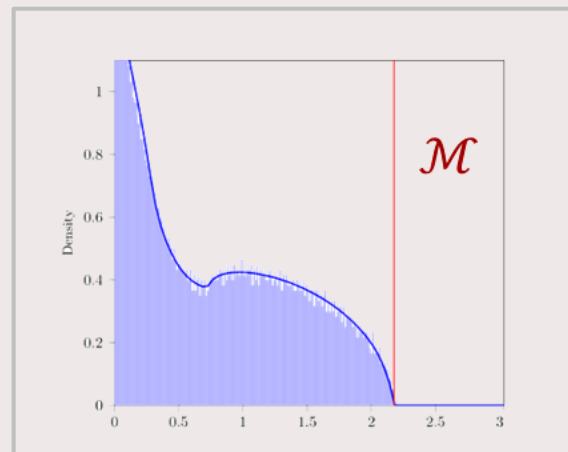
Denote $\mathbf{M} := \sqrt{n/\ell}(\widehat{\mathbf{N}} - \mathbb{E}[\widehat{\mathbf{N}}])$ and assume that $\ell = \omega(n \ln(n)^4)$ as $n \rightarrow \infty$. Then,

$$\|\mathbf{M}\| \rightarrow \mathcal{M} \quad \text{in probability}$$

where

$$\mathcal{M} := \inf_{x \in \mathbb{R}_{>0}^{2K}} \max_{i=1, \dots, 2K} \left\{ \frac{1}{x_i} + \sum_{j=1}^n c_{ij} x_j \right\}$$

with explicit coefficients c_{ij} .

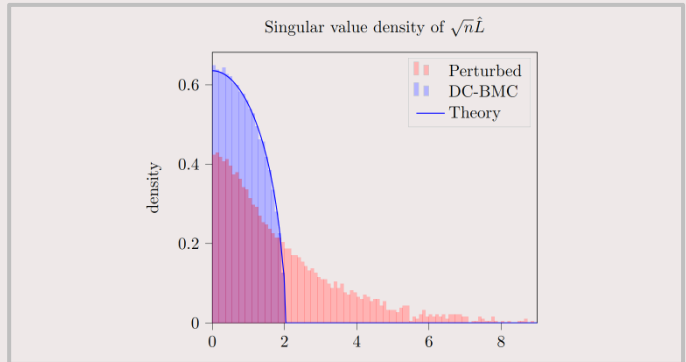
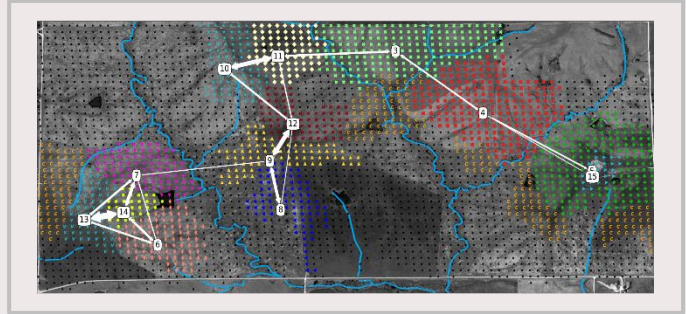


Summary

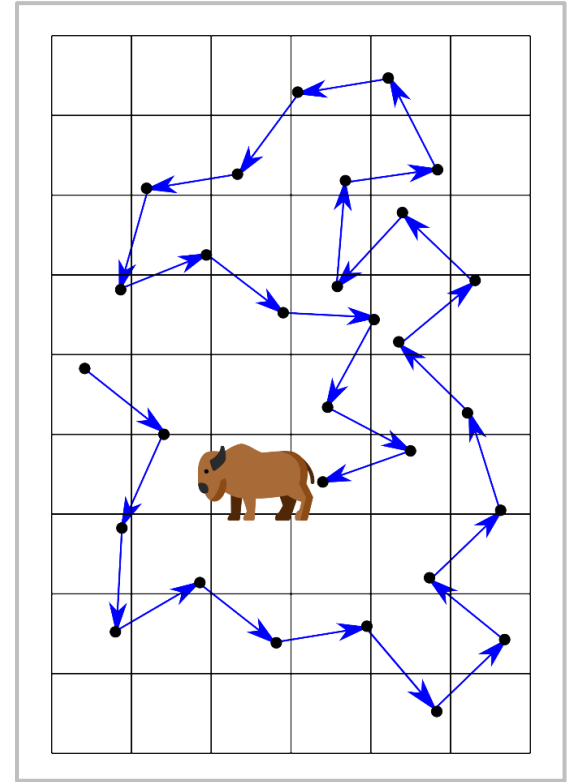
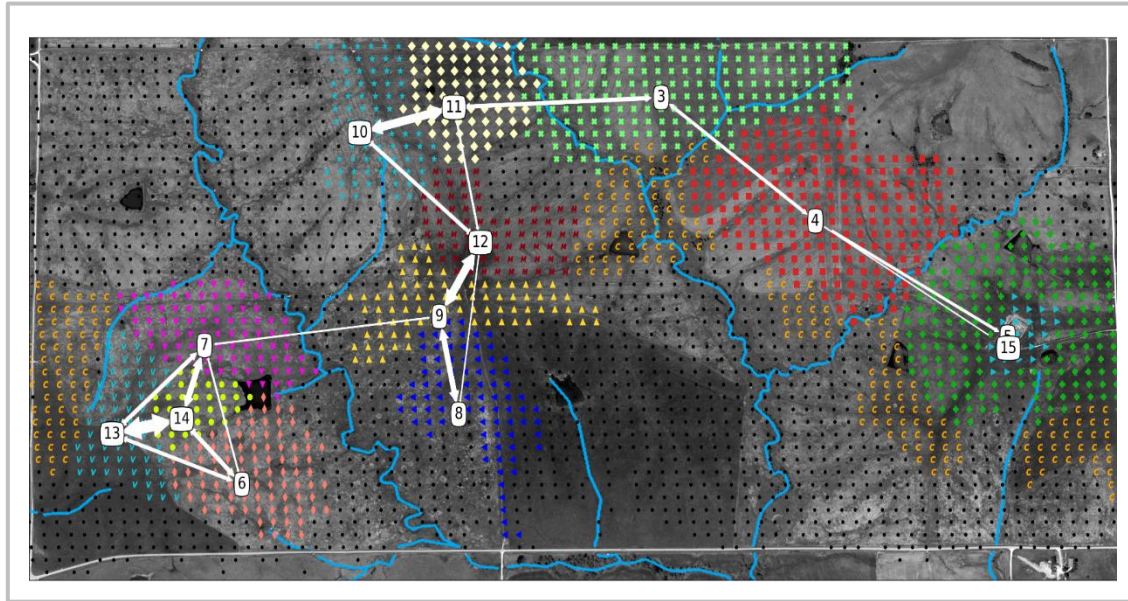
Clustering in real-world sequential data can produce insights!

In spectral model evaluation, it is best to use $\hat{\mathbf{L}}$ instead of $\hat{\mathbf{N}}$.

The theory heavily relies on random matrices with dependence. This gives rise to challenging proofs!



Thank you!



a.van.werde@tue.nl

`pip install BMCToolkit`

TU/e

Dataset: Text

Sequences of text: Wikipedia corpus

Every page corresponds to a sequence of words

Wikipedia → is → an → online → encyclopedia → ...



The screenshot shows the Wikipedia homepage in a browser window. The address bar displays "en.wikipedia.org/wiki/Wikipedia". The page features the Wikipedia logo, a navigation menu, and a sidebar with various links. The main content area includes the title "Wikipedia" and a brief introduction: "From Wikipedia, the free encyclopedia". Below this, there is a detailed paragraph defining Wikipedia as a multilingual free online encyclopedia. To the right, there is a "Screenshot" section with a table of key facts about Wikipedia.

Screenshot	
Type of site	Online encyclopedia
Available in	329 languages
Country of origin	United States
Owner	Wikimedia Foundation
Created by	Jimmy Wales Larry Sanger ^[1]
URL	wikipedia.org ^l
Commercial	No
Registration	Optional ^[note 1]
Users	>291,179 active editors ^[note 2] >104,148,259 registered users
Launched	January 15, 2001 (21 years ago)
Current status	Active
Content license	CC Attribution / Share-Alike 3.0 Most text is also dual-licensed under GFDL, media licensing

Dataset: Text

Sequences of text: Wikipedia corpus

Every page corresponds to a sequence of words

Wikipedia → is → an → online → encyclopedia → ...

Preprocessing:

- 1) Reduce to root words and prune 100 most frequent and extremely infrequent words which occur fewer than 1000 times.
- 2) Remove self transitions.
- 3) Aggregate data: $\hat{N} := \sum_{\text{Wikipedia pages } p} \hat{N}_p$.



Dataset: Text

Sequences of text: Wikipedia corpus

Every page corresponds to a sequence of words

Wikipedia → is → an → online → encyclopedia → ...

Preprocessing:

- 1) Reduce to root words and prune 100 most frequent and extremely infrequent words which occur fewer than 1000 times.
- 2) Remove self transitions.
- 3) Aggregate data: $\hat{N} := \sum_{\text{Wikipedia pages } p} \hat{N}_p$.

Resulting dataset:

Number of states: $n \approx 17\,000$

Sample path length: $\ell \approx 2 \times 10^8$

Sparsity: $\ell/n^2 \approx 1.4$



Clusters in text

200 clusters including:

$\mathcal{V}_{\text{color}} = \{\text{green, white, red, blue, black, gold}\}$

$\mathcal{V}_{\text{letters}} = \{\text{b, c, x, v, iii, g, e, r, f, j, d, k, p, l, w, h}\}$

$\mathcal{V}_{\text{music}} = \{\text{song, top, singl, album, track, band}\}$

$\mathcal{V}_{\text{navigation}} = \{\text{west, east, south, north, american}\}$

$\mathcal{V}_{\text{tech}} = \{\text{motor, magnet, real, nuclear, electr}\}$

$\mathcal{V}_{\text{politics}} = \{\text{socialist, reform, labour, communist, liber, democrat, republican, labor, conserv}\}$

$\mathcal{V}_{\text{compare}} = \{\text{larger, less, below, greater, higher, smaller, above, reduced, lower, low}\}$

Clusters in text

200 clusters including:

$\mathcal{V}_{\text{color}} = \{\text{green, white, red, blue, black, gold}\}$

$\mathcal{V}_{\text{letters}} = \{\text{b, c, x, v, iii, g, e, r, f, j, d, k, p, l, w, h}\}$

$\mathcal{V}_{\text{music}} = \{\text{song, top, singl, album, track, band}\}$

$\mathcal{V}_{\text{navigation}} = \{\text{west, east, south, north, american}\}$

$\mathcal{V}_{\text{tech}} = \{\text{motor, magnet, real, nuclear, electr}\}$

$\mathcal{V}_{\text{politics}} = \{\text{socialist, reform, labour, communist, liber, democrat, republican, labor, conserv}\}$

$\mathcal{V}_{\text{compare}} = \{\text{larger, less, below, greater, higher, smaller, above, reduced, lower, low}\}$

Benchmarking with document classification:

K	Algorithm	AG News	Yahoo!	Wiki	Book	CMU
50	Random	48.3%	27.4%	56.9%	31.0%	67.4%
50	Spectral	66.0%	39.8%	71.1%	44.4%	69.5%
50	Improved	68.5%	40.1%	71.5%	44.7%	71.8%
100	Random	55.5%	33.3%	68.4%	30.0%	67.4%
100	Spectral	72.7%	47.2%	81.6%	45.2%	70.0%
100	Improved	76.8%	49.0%	80.1%	46.3%	70.7%
200	Random	64.0%	41.7%	80.8%	28.2%	66.8%
200	Spectral	78.2%	51.7%	85.6%	44.4%	68.7%
200	Improved	80.7%	54.7%	86.5%	43.4%	69.0%
400	Random	72.8%	49.4%	87.8%	28.9%	66.8%
400	Spectral	81.5%	56.3%	88.0%	42.1%	67.9%
400	Improved	83.1%	58.6%	89.0%	44.4%	68.4%

Clusters in text

200 clusters including:

$\mathcal{V}_{\text{color}} = \{\text{green, white, red, blue, black, gold}\}$

$\mathcal{V}_{\text{letters}} = \{\text{b, c, x, v, iii, g, e, r, f, j, d, k, p, l, w, h}\}$

$\mathcal{V}_{\text{music}} = \{\text{song, top, singl, album, track, band}\}$

$\mathcal{V}_{\text{navigation}} = \{\text{west, east, south, north, american}\}$

$\mathcal{V}_{\text{tech}} = \{\text{motor, magnet, real, nuclear, electr}\}$

$\mathcal{V}_{\text{politics}} = \{\text{socialist, reform, labour, communist, liber, democrat, republican, labor, conserv}\}$

$\mathcal{V}_{\text{compare}} = \{\text{larger, less, below, greater, higher, smaller, abov, reduc, lower, low}\}$

Improvement algorithm is helpful!

Benchmarking with document classification:

K	Algorithm	AG News	Yahoo!	Wiki	Book	CMU
50	Random	48.3%	27.4%	56.9%	31.0%	67.4%
50	Spectral	66.0%	39.8%	71.1%	44.4%	69.5%
50	Improved	68.5%	40.1%	71.5%	44.7%	71.8%
100	Random	55.5%	33.3%	68.4%	30.0%	67.4%
100	Spectral	72.7%	47.2%	81.6%	45.2%	70.0%
100	Improved	76.8%	49.0%	80.1%	46.3%	70.7%
200	Random	64.0%	41.7%	80.8%	28.2%	66.8%
200	Spectral	78.2%	51.7%	85.6%	44.4%	68.7%
200	Improved	80.7%	54.7%	86.5%	43.4%	69.0%
400	Random	72.8%	49.4%	87.8%	28.9%	66.8%
400	Spectral	81.5%	56.3%	88.0%	42.1%	67.9%
400	Improved	83.1%	58.6%	89.0%	44.4%	68.4%

Dataset: Stocks

Sequence of highest daily returning stock on S&P500

Compute daily return per stock as $R_t := \frac{\text{Closing price}}{\text{Opening price}} - 1$.

For every day, consider stock with highest return.

... \rightarrow *GOOGL* \rightarrow *AMZN* \rightarrow *NTAP* \rightarrow *HUM* \rightarrow ...



Dataset: Stocks

Sequence of highest daily returning stock on S&P500

Compute daily return per stock as $R_t := \frac{\text{Closing price}}{\text{Opening price}} - 1$.

For every day, consider stock with highest return.

... $\rightarrow GOOGL \rightarrow AMZN \rightarrow NTAP \rightarrow HUM \rightarrow \dots$

Preprocessing:

- 1) Only consider time range where complete data is available.
That is, approximately from 2001 to 2021.
- 2) Eliminate self transitions.



Dataset: Stocks

Sequence of highest daily returning stock on S&P500

Compute daily return per stock as $R_t := \frac{\text{Closing price}}{\text{Opening price}} - 1$.

For every day, consider stock with highest return.

... $\rightarrow GOOGL \rightarrow AMZN \rightarrow NTAP \rightarrow HUM \rightarrow \dots$

Preprocessing:

- 1) Only consider time range where complete data is available.
That is, approximately from 2001 to 2021.
- 2) Eliminate self transitions.

Resulting dataset:

Number of states: $n = 300$
Sample path length: $\ell \approx 2500$
Sparsity: $\ell/n^2 \approx 0.027$



Clusters in stocks

Comparison to alternative model:

[P] Block Markov chain with $K = 3$ after improvement step.

[Q₄] Sequence of independent random variables with 3 classes having different frequencies.

Clusters in stocks

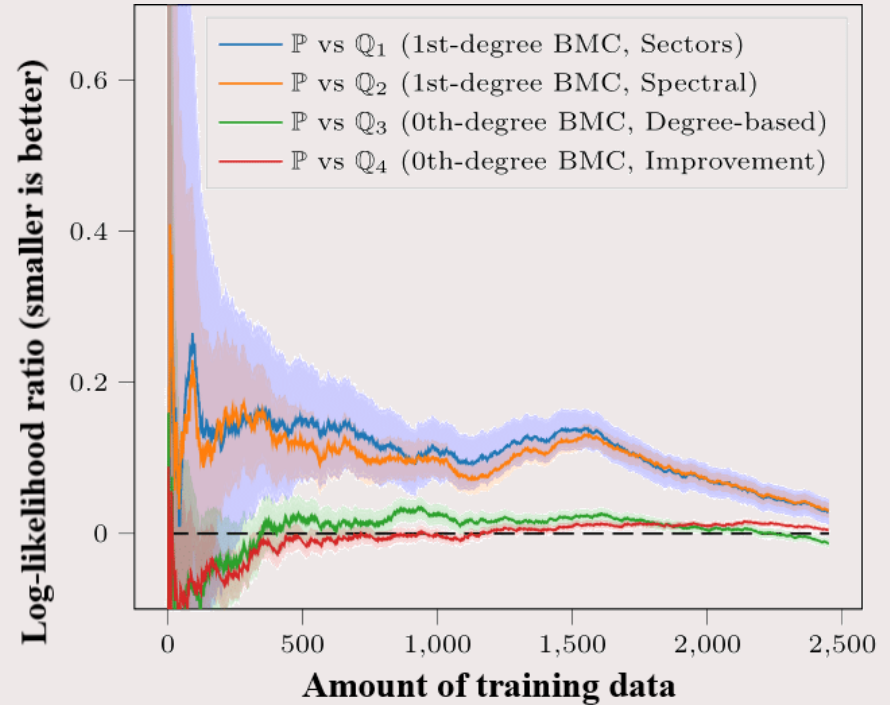
Comparison to alternative model:

[\mathbb{P}] Block Markov chain with $K = 3$ after improvement step.

[\mathbb{Q}_4] Sequence of independent random variables with 3 classes having different frequencies.

Metric for comparison:

Log-likelihood ratio evaluated on validation data.



Clusters in stocks

Comparison to alternative model:

[\mathbb{P}] Block Markov chain with $K = 3$ after improvement step.

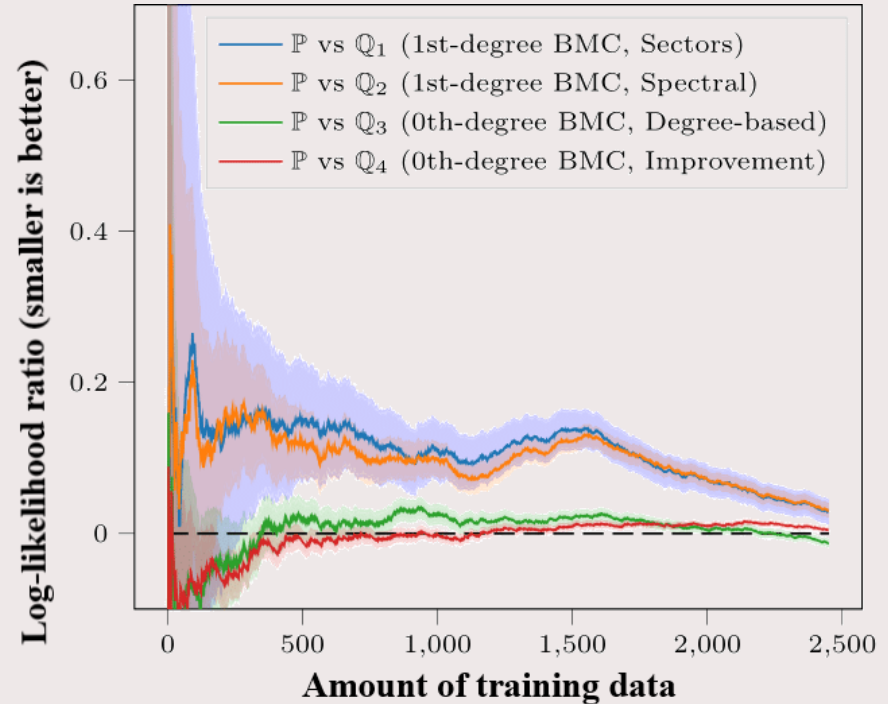
[\mathbb{Q}_4] Sequence of independent random variables with 3 classes having different frequencies.

Metric for comparison:

Log-likelihood ratio evaluated on validation data.

Observation:

The simpler model \mathbb{Q}_4 performs similarly. Possibly because the training data is very sparse.



Clusters in stocks

Comparison to alternative model:

[\mathbb{P}] Block Markov chain with $K = 3$ after improvement step.

[\mathbb{Q}_4] Sequence of independent random variables with 3 classes having different frequencies.

Metric for comparison:

Log-likelihood ratio evaluated on validation data.

Observation:

The simpler model \mathbb{Q}_4 performs similarly. Possibly because the training data is very sparse.

Delicate model evaluation

