Database alignment: fundamental limits and efficient algorithms

Negar Kiyavash
École Polytechnique Fédérale de Lausanne

Joint work with
Daniel Cullina - Penn State
Osman Emre Dai - Georgia Tech

November 11th, 2021

Motivation

We are subject to ubiquitous data collection.

Motivation

We are subject to ubiquitous data collection.

- Diverse data sources
  - Data junction might offer great benefits

Motivation

We are subject to ubiquitous data collection.

- Diverse data sources
  - Data junction might offer great benefits
- Data junction often not directly possible
  (E.g. anonymized data)

Motivation

We are subject to ubiquitous data collection.

- Diverse data sources
  - Data junction might offer great benefits
- Data junction often not directly possible
  (E.g. anonymized data)
- Correlation among data
  - Possibility of non-obvious alignment and inference

Motivation

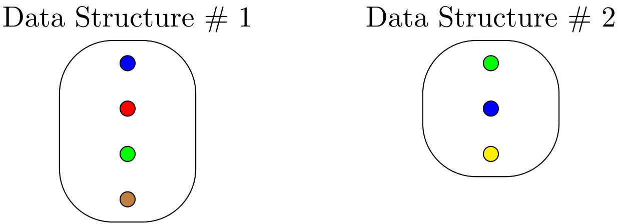
We are subject to ubiquitous data collection.

- Diverse data sources
  - Data junction might offer great benefits
- Data junction often not directly possible
  (E.g. anonymized data)
- Correlation among data
  - Possibility of non-obvious alignment and inference
- Risks on privacy
  - Crucial to understand conditions that allow or prevent privacy breaches

Alignment problem

The alignment problem studies the scenario where

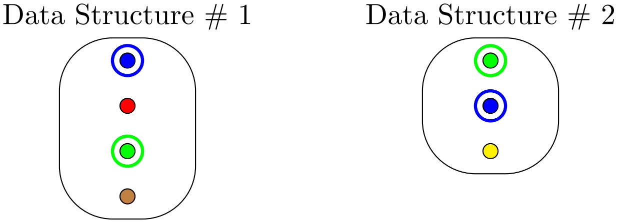
- multiple data sources are present,



Alignment problem

The alignment problem studies the scenario where

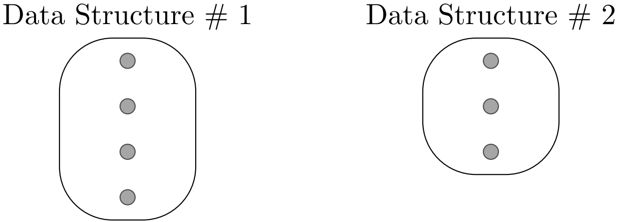
- multiple data sources are present,
- there are ‘users’ whose data is available from each source,



Alignment problem

The alignment problem studies the scenario where

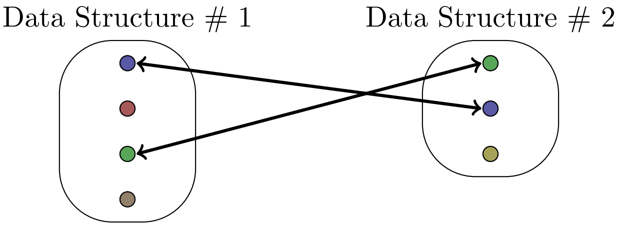
- multiple data sources are present,
- there are 'users' whose data is available from each source,
- correspondence between data sources is obfuscated or unknown,



Alignment problem

The alignment problem studies the scenario where

- multiple data sources are present,
- there are ‘users’ whose data is available from each source,
- correspondence between data sources is obfuscated or unknown,
- it is possible to identify the correspondences *if* the correlation between data is strong enough.

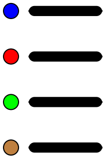


Alignment Problem

Structure of data

- Data associated to single users: database alignment
  E.g. medical records

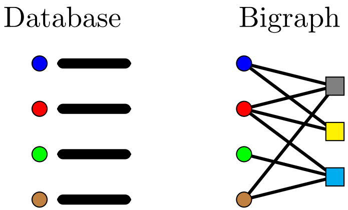
Database



Alignment Problem

Structure of data

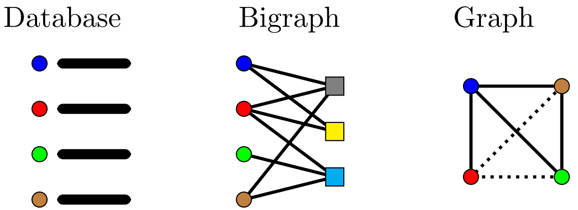
- Data associated to single users: database alignment
  E.g. medical records
- Interactions between users and objects: bipartite alignment
  E.g. customer movie ratings



Alignment Problem

Structure of data

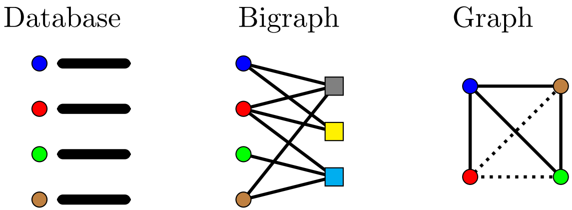
- Data associated to single users: database alignment
  E.g. medical records
- Interactions between users and objects: bipartite alignment
  E.g. customer movie ratings
- Interactions among users: graph alignment
  E.g. connections on social media websites



Alignment Problem

Structure of data

- Data associated to single users: database alignment
  E.g. medical records
- Interactions between users and objects: bipartite alignment
  E.g. customer movie ratings
- Interactions among users: graph alignment
  E.g. connections on social media websites
- Or any combination of these



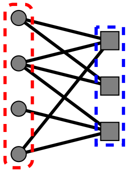
Illustrative example

Bipartite alignment: Movie ratings

Illustrative example

Bipartite alignment: Movie ratings

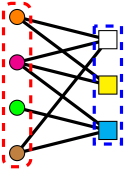
- Netflix prize dataset
  User IDs, movie IDs, movie ratings



Illustrative example

Bipartite alignment: Movie ratings

- Netflix prize dataset
  User IDs, movie IDs, movie ratings
- IMDB user ratings
  Usernames , movie names, movie ratings



Illustrative example

- Common for IMDB users to register with full name

[1] Arvind Narayanan and Vitaly Shmatikov, "How To Break Anonymity of the Netflix Prize Dataset," S&P 2006

Illustrative example

- Common for IMDB users to register with full name
- Such users may avoid publicly rating certain movies of interest.

[1] Arvind Narayanan and Vitaly Shmatikov, "How To Break Anonymity of the Netflix Prize Dataset," S&P 2006

Illustrative example

- Common for IMDB users to register with full name
- Such users may avoid publicly rating certain movies of interest.
- Movie interests may reveal insight on personal information.

[1] Arvind Narayanan and Vitaly Shmatikov, "How To Break Anonymity of the Netflix Prize Dataset," S&P 2006

Illustrative example

- Common for IMDB users to register with full name
- Such users may avoid publicly rating certain movies of interest.
- Movie interests may reveal insight on personal information.
    - Political views

[1] Arvind Narayanan and Vitaly Shmatikov, "How To Break Anonymity of the Netflix Prize Dataset," S&P 2006

Illustrative example

- Common for IMDB users to register with full name
- Such users may avoid publicly rating certain movies of interest.
- Movie interests may reveal insight on personal information.
  - Political views
  - Religious beliefs

[1] Arvind Narayanan and Vitaly Shmatikov, "How To Break Anonymity of the Netflix Prize Dataset," S&P 2006

Illustrative example

- Common for IMDB users to register with full name
- Such users may avoid publicly rating certain movies of interest.
- Movie interests may reveal insight on personal information.
  - Political views
  - Religious beliefs
  - Sexual orientation

[1] Arvind Narayanan and Vitaly Shmatikov, "How To Break Anonymity of the Netflix Prize Dataset," S&P 2006

Illustrative example

- Common for IMDB users to register with full name
- Such users may avoid publicly rating certain movies of interest.
- Movie interests may reveal insight on personal information.
  - Political views
  - Religious beliefs
  - Sexual orientation
- It was shown that many Netflix user IDs can be matched with public IMDB profiles. [1]

[1] Arvind Narayanan and Vitaly Shmatikov, "How To Break Anonymity of the Netflix Prize Dataset," S&P 2006

Illustrative example

- Common for IMDB users to register with full name
- Such users may avoid publicly rating certain movies of interest.
- Movie interests may reveal insight on personal information.
  - Political views
  - Religious beliefs
  - Sexual orientation

- It was shown that many Netflix user IDs can be matched with public IMDB profiles. [1]
- Netflix faced class action lawsuit and canceled sequal to competition.

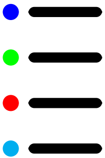
---

[1] Arvind Narayanan and Vitaly Shmatikov, "How To Break Anonymity of the Netflix Prize Dataset," S&P 2006

Database alignment

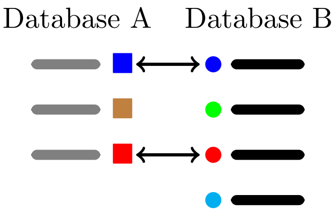
Correlated database model

Database: (unordered) set of features, each associated with a user



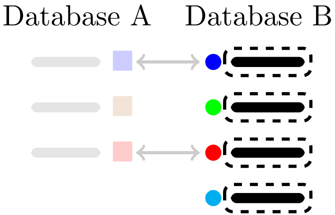
Correlated database model

Pair of databases with correlated data
Some users might not be present on both databases.



Correlated database model

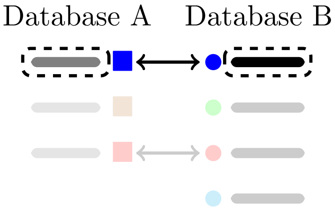
Features in a database are i.i.d.



independent
and
identically
distributed

Correlated database model

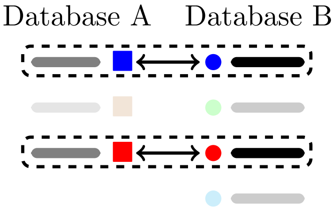
Features associated with the same user are jointly distributed.



jointly
distributed

Correlated database model

Features associated with the same user are jointly distributed.
Each pair is i.i.d.



independent

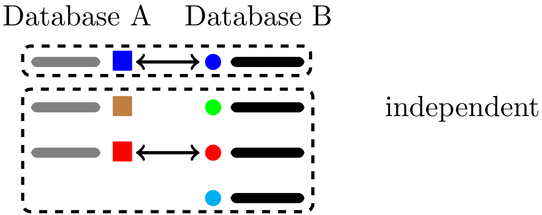
and

identically

distributed

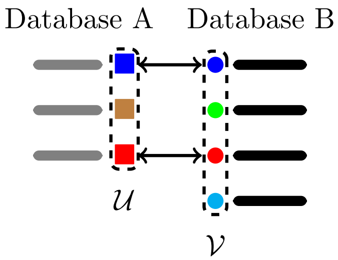
Correlated database model

Features of a user are independent from all other features.



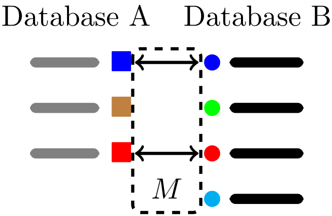
Likelihood function

$\mathcal{U}$ and $\mathcal{V}$: Sets of user identifiers.



Likelihood function

$\mathcal{U}$ and $\mathcal{V}$: Sets of user identifiers.
M: Mapping between identifiers

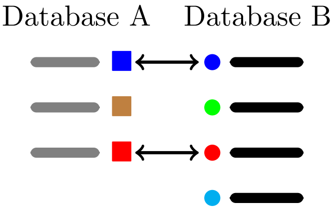


Likelihood function

$\mathcal{U}$ and $\mathcal{V}$: Sets of user identifiers.

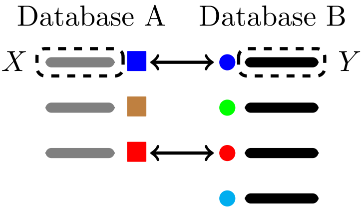
Given some mapping M:

- $\mathcal{W}_M = \{({\color{blue}\blacksquare},{\color{blue}\bullet}), ({\color{red}\blacksquare},{\color{red}\bullet})\}$: pairs mapped by M
- $\mathcal{U}_M = \{{\color{blue}\blacksquare},{\color{red}\blacksquare}\}$: mapped users from A
- $\mathcal{V}_M = \{{\color{blue}\bullet},{\color{red}\bullet}\}$: mapped users from B



Likelihood function

- $X = A(\blacksquare)$: arbitrary feature from database A
- $Y = B(\bullet)$: arbitrary feature from database B

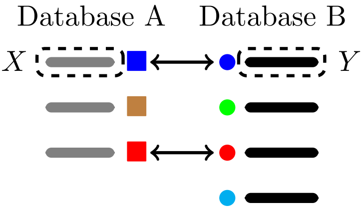


Likelihood function

- $X = A(\blacksquare)$: arbitrary feature from database A
- $Y = B(\bullet)$: arbitrary feature from database B

- f_X, f_Y: marginal pdfs (or pmfs) of features
- $f_{XY|M}$: joint pdf (or pmf) of features of a user, given M

The distributions are known.



Likelihood function

- $X = A(u)$: arbitrary feature from database A
- $Y = B(v)$: arbitrary feature from database B

- f_X, f_Y: marginal pdfs (or pmfs) of features
- $f_{XY|M}$: joint pdf (or pmf) of features of a user, given M

Then the log likelihood of databases A, B given M is

$$\sum_{(u,v)\in\mathcal{W}_M} \log f_{XY|M}\big(A(u), B(v)\big) + \sum_{u\in\mathcal{U}\setminus\mathcal{U}_M} \log f_X\big(A(u)\big) + \sum_{v\in\mathcal{V}\setminus\mathcal{V}_M} \log f_Y\big(B(v)\big)$$

Likelihood function

$$\sum_{(u,v)\in\mathcal{W}_M} \log f_{XY|M}\big(A(u), B(v)\big) + \sum_{u\in\mathcal{U}\setminus\mathcal{U}_M} \log f_X\big(A(u)\big) + \sum_{v\in\mathcal{V}\setminus\mathcal{V}_M} \log f_Y\big(B(v)\big)$$

This can be rewritten as

$$\sum_{(u,v)\in\mathcal{W}_M} \log \frac{f_{XY|M}\big(A(u), B(v)\big)}{f_X\big(A(u)\big) f_Y\big(B(v)\big)} + \sum_{u\in\mathcal{U}} \log f_X\big(A(u)\big) + \sum_{v\in\mathcal{V}} \log f_Y\big(B(v)\big)$$

Maximum likelihood estimation

$$\sum_{(u,v)\in\mathcal{W}_M} \log \frac{f_{XY|M}\big(A(u),B(v)\big)}{f_X\big(A(u)\big)\,f_Y\big(B(v)\big)}$$
$$+ \sum_{u\in\mathcal{U}} \log f_X\big(A(u)\big)$$
$$+ \sum_{v\in\mathcal{V}} \log f_Y\big(B(v)\big)$$

The last two terms do not depend on M.

We only need to consider the first term to maximize likelihood.

Maximum likelihood estimation

$$\sum_{(u,v)\in\mathcal{W}_M} \log \frac{f_{XY|M}\big(A(u), B(v)\big)}{f_X\big(A(u)\big)\, f_Y\big(B(v)\big)}$$

Maximum likelihood estimation

$$\sum_{(u,v)\in\mathcal{W}_M} \log \frac{f_{XY|M}\big(A(u), B(v)\big)}{f_X\big(A(u)\big)\, f_Y\big(B(v)\big)}$$

$\mathbf{G} \in \mathbb{R}^{\mathcal{U}\times\mathcal{V}}$:

Information density matrix s.t. $G_{u,v} = \log \frac{f_{XY|M}\big(A(u),B(v)\big)}{f_X\big(A(u)\big)\, f_Y\big(B(v)\big)}$

Maximum likelihood estimation

$$\sum_{(u,v)\in\mathcal{W}_M} \log \frac{f_{XY|M}\big(A(u),B(v)\big)}{f_X\big(A(u)\big)\,f_Y\big(B(v)\big)}$$

$\mathbf{G} \in \mathbb{R}^{\mathcal{U}\times\mathcal{V}}$:
Information density matrix s.t. $G_{u,v} = \log \frac{f_{XY|M}\big(A(u),B(v)\big)}{f_X\big(A(u)\big)\,f_Y\big(B(v)\big)}$

$\mathbf{M} \in \{0,1\}^{\mathcal{U}\times\mathcal{V}}$:
Matrix encoding of mapping M s.t. $M_{u,v} = 1 \iff M$ maps (u,v)

Maximum likelihood estimation

$$\sum_{(u,v)\in\mathcal{W}_M} \log \frac{f_{XY|M}\big(A(u), B(v)\big)}{f_X\big(A(u)\big) f_Y\big(B(v)\big)}$$

$\mathbf{G} \in \mathbb{R}^{\mathcal{U}\times\mathcal{V}}$:

Information density matrix s.t. $G_{u,v} = \log \frac{f_{XY|M}\big(A(u),B(v)\big)}{f_X\big(A(u)\big) f_Y\big(B(v)\big)}$

$\mathbf{M} \in \{0,1\}^{\mathcal{U}\times\mathcal{V}}$:
Matrix encoding of mapping M s.t. $M_{u,v} = 1 \iff M$ maps (u,v)

The summation can be written as an inner product:

$$\sum_{(u,v)\in\mathcal{W}_{\color{blue}M}} \log \frac{f_{XY|M}\big({\color{red}A}(u),{\color{red}B}(v)\big)}{f_X\big({\color{red}A}(u)\big) f_Y\big({\color{red}B}(v)\big)} = \langle {\color{red}\mathbf{G}}, {\color{blue}\mathbf{M}} \rangle$$

Maximum likelihood estimation

- **G** is a function of the random databases A and B and is computable.

[2] Lyle Ramshaw and Robert E. Tarjan, On minimum-cost assignments inunbalanced bipartite graphs, HP Labs 2012

Maximum likelihood estimation

- $\mathbf{G}$ is a function of the random databases A and B and is computable.
- $\mathbf{M}$ is a (partial) permutation matrix.

[2] Lyle Ramshaw and Robert E. Tarjan, On minimum-cost assignments inunbalanced bipartite graphs, HP Labs 2012

Maximum likelihood estimation

- $\mathbf{G}$ is a function of the random databases A and B and is computable.
- $\mathbf{M}$ is a (partial) permutation matrix.
- If $|M|$ is known, maximizing $\langle \mathbf{G}, \mathbf{M} \rangle$ is equivalent to the (unbalanced) linear assignment problem.

[2] Lyle Ramshaw and Robert E. Tarjan, On minimum-cost assignments inunbalanced bipartite graphs, HP Labs 2012

Maximum likelihood estimation

- $\mathbf{G}$ is a function of the random databases A and B and is computable.
- $\mathbf{M}$ is a (partial) permutation matrix.
- If $|M|$ is known, maximizing $\langle \mathbf{G}, \mathbf{M} \rangle$ is equivalent to the (unbalanced) linear assignment problem.
- This can be solved using the Hungarian algorithm in $\mathcal{O}\big(|\mathcal{U}| \cdot |\mathcal{V}| \cdot |M|\big)$-time [2].

[2] Lyle Ramshaw and Robert E. Tarjan, On minimum-cost assignments inunbalanced bipartite graphs, HP Labs 2012

Maximum likelihood estimation

- $\mathbf{G}$ is a function of the random databases A and B and is computable.
- $\mathbf{M}$ is a (partial) permutation matrix.
- If $|M|$ is known, maximizing $\langle \mathbf{G}, \mathbf{M} \rangle$ is equivalent to the (unbalanced) linear assignment problem.
- This can be solved using the Hungarian algorithm in $\mathcal{O}\big(|\mathcal{U}| \cdot |\mathcal{V}| \cdot |M|\big)$-time [2].
- Therefore maximum likelihood estimation is possible in polynomial time.

[2]Lyle Ramshaw and Robert E. Tarjan, On minimum-cost assignments inunbalanced bipartite graphs, HP Labs 2012

Other estimators

MLE optimizes over all mappings.
Simpler and faster approaches exist for aligning small subset.

Other estimators

MLE optimizes over all mappings.
Simpler and faster approaches exist for aligning small subset.
Other algorithms:

Maximum row alignment

Other estimators

MLE optimizes over all mappings.
Simpler and faster approaches exist for aligning small subset.
Other algorithms:

Maximum row alignment

- Given user in database A, find feature in database B.

Other estimators

MLE optimizes over all mappings.
Simpler and faster approaches exist for aligning small subset.
Other algorithms:

Maximum row alignment

- Given user in database A, find feature in database B.
- Ignore all other features in A.
  Pick feature in B that maximizes likelihood.

Other estimators

MLE optimizes over all mappings.
Simpler and faster approaches exist for aligning small subset.
Other algorithms:

Maximum row alignment

- Given user in database A, find feature in database B.
- Ignore all other features in A.
  Pick feature in B that maximizes likelihood.
- Equivalent to picking the max entry in a row of $\mathbf{G}$.

Other estimators

MLE optimizes over all mappings.
Simpler and faster approaches exist for aligning small subset.
Other algorithms:

Thresholding

Other estimators

MLE optimizes over all mappings.
Simpler and faster approaches exist for aligning small subset.
Other algorithms:

<u>Thresholding</u>

- Decide if given feature pair is correlated.

Other estimators

MLE optimizes over all mappings.
Simpler and faster approaches exist for aligning small subset.
Other algorithms:

Thresholding

- Decide if given feature pair is correlated.
- Perform likelihood-ratio test with some threshold.

Other estimators

MLE optimizes over all mappings.
Simpler and faster approaches exist for aligning small subset.
Other algorithms:

Thresholding

- Decide if given feature pair is correlated.
- Perform likelihood-ratio test with some threshold.
- Equivalent to accepting if corresponding entry in $\mathbf{G}$ is above some threshold.

Database alignment

Results

Finite alphabet databases

- Features in A and B take values from finite alphabets $\mathcal{X}$ and $\mathcal{Y}$.

[3] Daniel Cullina, Prateek Mittal and Negar Kiyavash, Fundamental limits of database alignment, ISIT 2018

Finite alphabet databases

- Features in A and B take values from finite alphabets $\mathcal{X}$ and $\mathcal{Y}$.
- Critical information theoretical measure: I_2°

[3] Daniel Cullina, Prateek Mittal and Negar Kiyavash, Fundamental limits of database alignment, ISIT 2018

Finite alphabet databases

- Features in A and B take values from finite alphabets $\mathcal{X}$ and $\mathcal{Y}$.
- Critical information theoretical measure: I_2°

  Asymptotic case: $|M| = |\mathcal{U}| = |\mathcal{V}| = n \to \infty$

[3] Daniel Cullina, Prateek Mittal and Negar Kiyavash, Fundamental limits of database alignment, ISIT 2018

Finite alphabet databases

- Features in A and B take values from finite alphabets $\mathcal{X}$ and $\mathcal{Y}$.
- Critical information theoretical measure: I_2°

  Asymptotic case: $|M| = |\mathcal{U}| = |\mathcal{V}| = n \to \infty$
- **Achievability**: [3] MLE finds the correct mapping with probability $1 - o(1)$ as long as

$$I_2^\circ \geq 2 \log n + \omega(1) \text{ [3]}.$$

[3] Daniel Cullina, Prateek Mittal and Negar Kiyavash, Fundamental limits of database alignment, ISIT 2018

Finite alphabet databases

- Features in A and B take values from finite alphabets $\mathcal{X}$ and $\mathcal{Y}$.
- Critical information theoretical measure: I_2°

  Asymptotic case: $|M| = |\mathcal{U}| = |\mathcal{V}| = n \to \infty$
- **Achievability**: [3] MLE finds the correct mapping with probability $1 - o(1)$ as long as

$$I_2^\circ \geq 2\log n + \omega(1) \text{ [3].}$$

- **Converse**: [3] Any algorithm fails to find the correct mapping with probability $1 - o(1)$ if

$$I_2^\circ \leq 2\log n(1 - \Omega(1)) \text{ [3].}$$

[3] Daniel Cullina, Prateek Mittal and Negar Kiyavash, Fundamental limits of database alignment, ISIT 2018

Finite alphabet databases

What is I_2°?

Finite alphabet databases

What is I_2°?

- When $|M| = |\mathcal{U}| = |\mathcal{V}|$, the smallest number of errors the estimator makes is 2.
  (If one user is mapped to a wrong user, then the same holds for that second user.)

Finite alphabet databases

What is I_2°?

- When $|M| = |\mathcal{U}| = |\mathcal{V}|$, the smallest number of errors the estimator makes is 2.
  (If one user is mapped to a wrong user, then the same holds for that second user.)
- Consider two mappings m and m' that differ only for two pairs of users:
  m maps $u_1 \sim v_1$ and $u_2 \sim v_2$.
  m' maps $u_1 \sim v_2$ and $u_2 \sim v_1$.

Finite alphabet databases

What is I_2°?

- When $|M| = |\mathcal{U}| = |\mathcal{V}|$, the smallest number of errors the estimator makes is 2.
  (If one user is mapped to a wrong user, then the same holds for that second user.)
- Consider two mappings m and m' that differ only for two pairs of users:
  m maps $u_1 \sim v_1$ and $u_2 \sim v_2$.
  m' maps $u_1 \sim v_2$ and $u_2 \sim v_1$.
- I_2° is the **Bhattacharyya distance** between the distribution of the databases under m and under m'.

Finite alphabet databases

- Features in A and B take values from finite alphabets $\mathcal{X}$ and $\mathcal{Y}$.
- Critical information theoretical measure: I_2°

  Consider the case where $|M| = |\mathcal{U}| = |\mathcal{V}| = n \to \infty$
- **Achievability**: [3] MLE finds the correct mapping with probability $1 - o(1)$ as long as

$$I_2^\circ \geq 2\log n + \omega(1) \text{ [3]}.$$

- **Converse**: [3] Any algorithm fails to find the correct mapping with probability $1 - o(1)$ if

$$I_2^\circ \leq 2\log n(1 - \Omega(1)) \text{ [3]}.$$

[3] Daniel Cullina, Prateek Mittal and Negar Kiyavash, Fundamental limits of database alignment, ISIT 2018

Gaussian databases

- Features in A and B take values in vector spaces.
- Features of a user are multivariate Gaussian.

[4] Osman Dai, Daniel Cullina, and Negar Kiyavash, Database alignment with Gaussian features, AISTATS 2019

Gaussian databases

- Features in A and B take values in vector spaces.
- Features of a user are multivariate Gaussian.
- Critical information theoretical measure:
  Mutual information between correlated features I_{XY}

[4] Osman Dai, Daniel Cullina, and Negar Kiyavash, Database alignment with Gaussian features, AISTATS 2019

Gaussian databases

- Features in A and B take values in vector spaces.
- Features of a user are multivariate Gaussian.
- Critical information theoretical measure:
  Mutual information between correlated features I_{XY}
  I_{XY} is equal to the **Bhattacharyya distance** between the distribution of databases for 'adjacent' mappings.

[4] Osman Dai, Daniel Cullina, and Negar Kiyavash, Database alignment with Gaussian features, AISTATS 2019

Gaussian databases

- Features in A and B take values in vector spaces.
- Features of a user are multivariate Gaussian.
- Critical information theoretical measure:
  Mutual information between correlated features I_{XY}
  I_{XY} is equal to the **Bhattacharyya distance** between the distribution of databases for 'adjacent' mappings.
  Asymptotic case: $|M| = |\mathcal{U}| = |\mathcal{V}| = n \to \infty$
- **Achievability**: [4] MLE finds the correct mapping with probability $1 - o(1)$ as long as

$$I_{XY} \geq 2 \log n + \omega(1) \text{ [4]}.$$

- **Converse**: [4] Any algorithm fails to find the correct mapping with probability $1 - o(1)$ if

$$I_{XY} \leq 2 \log n (1 - \Omega(1)) \text{ [4]}.$$

[4] Osman Dai, Daniel Cullina, and Negar Kiyavash, Database alignment with Gaussian features, AISTATS 2019

Gaussian databases

- These results consider estimation a success only if it gives the exact mapping.

Gaussian databases

- These results consider estimation a success only if it gives the exact mapping.
- What if the estimate is not exact but the ratio of errors is vanishingly small?

Gaussian databases

- Exact alignment
  - **Achievability**: If $I_{XY} \geq 2\log n + \omega(1)$, then MLE finds the correct mapping with probability $1 - o(1)$ [4].
  - **Converse**: If $I_{XY} \leq 2\log n(1 - \Omega(1))$, then any algorithm fails to find the correct mapping with probability $1 - o(1)$ [4].

- Almost-exact alignment
  - **Achievability**: If $I_{XY} \geq \log n + \omega(1)$, then MLE makes $o(n)$ errors in expectation [4].
  - **Converse**: If $I_{XY} \leq \log n(1 - \Omega(1))$, then any algorithm makes $\Omega(n)$ errors in expectation [4].

[4] Osman Dai, Daniel Cullina, and Negar Kiyavash, Database alignment with Gaussian features, AISTATS 2019

Gaussian alignment

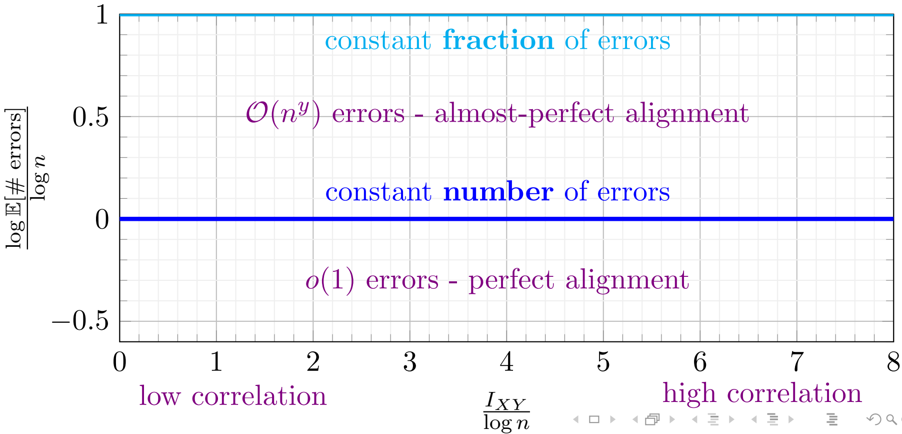
Asymptotic regime: $n \to \infty$

x-axis: mutual information

$I_{XY} = x \log n$

y-axis: error exponent

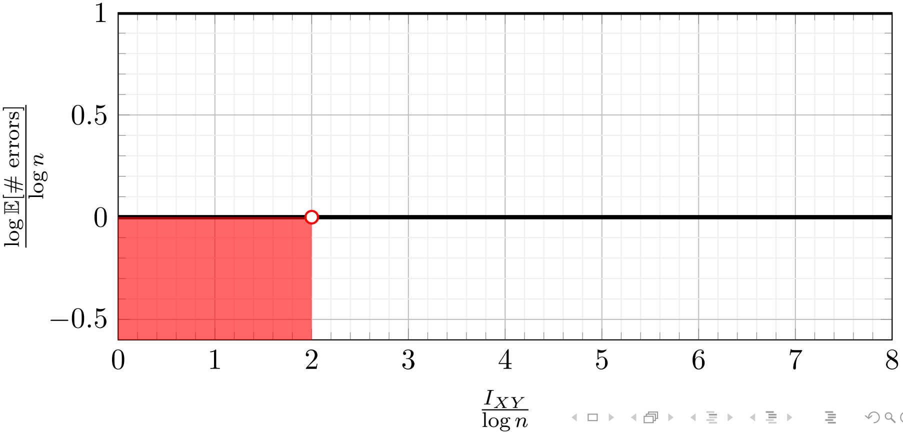
$\mathbb{E}[\#\text{errors}] = n^y$



Gaussian alignment

Converse:

- $I_{XY} \leq 2\log n(1-\Omega(1)) \implies \mathbb{E}[\#\text{errors}] \geq \Omega(1)$
- $x < 2 \implies y \geq 0$

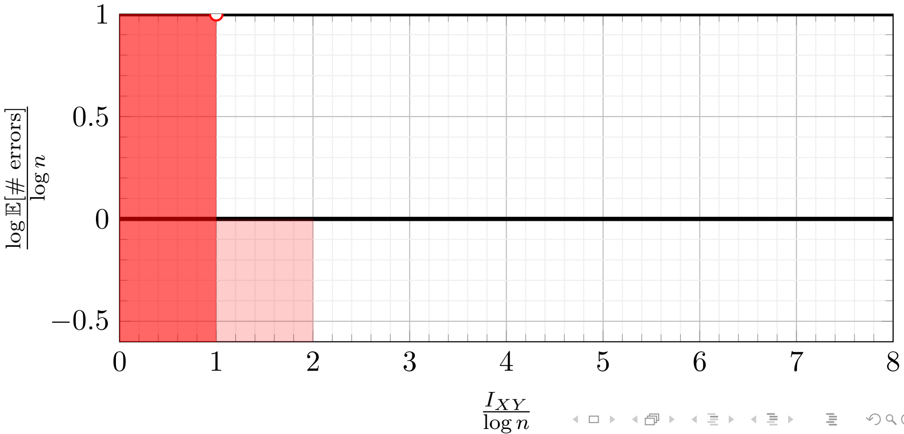


Gaussian alignment

Converse:

- $I_{XY} \leq \log n(1 - \Omega(1)) \implies \mathbb{E}[\#\text{errors}] \geq \Omega(n)$
- $x < 1 \implies y \geq 1$

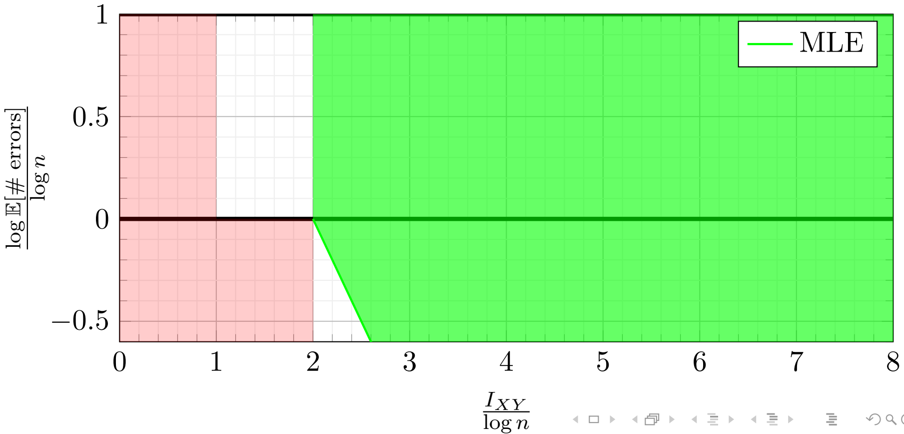
+



Gaussian alignment

Achievability (MLE - high correlation):

- $I_{XY} \geq 2\log n + \omega(1)$
  $\implies \mathbb{E}[\#\text{errors}] \leq 2\exp(2\log n - I_{XY})(1 + o(1))$
- $x > 2 \implies y \leq 2 - x$

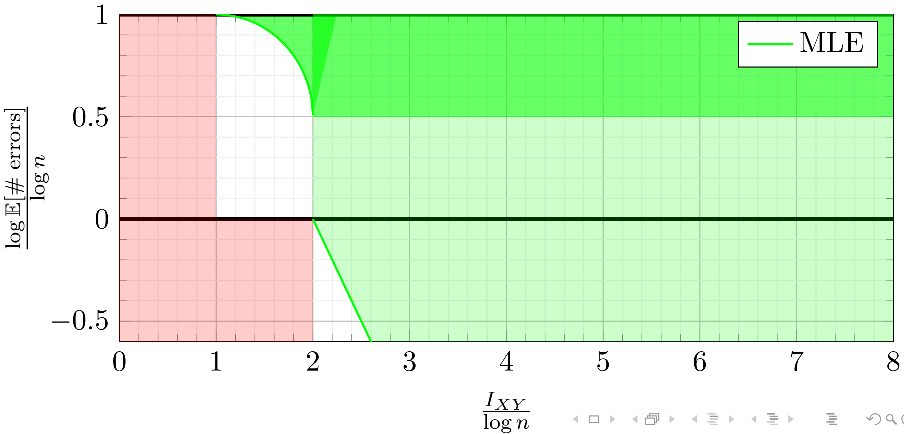


Gaussian alignment

Achievability (MLE - low correlation) [5]:

- $x > 1 \implies (x-1)^2 + (2y-1)^2 \leq 1$

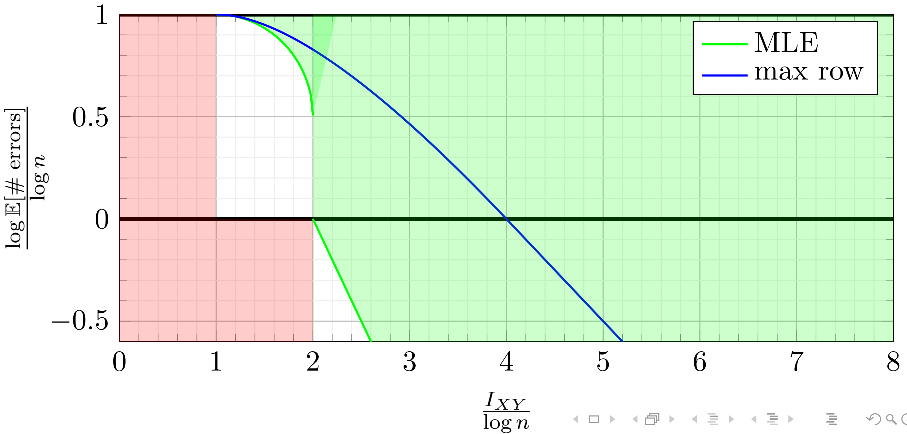
[5] Osman Dai, Daniel Cullina, and Negar Kiyavash, Achievability of nearly-exact alignment for correlated Gaussian databases, ISIT 2020



Gaussian alignment

Achievability (maximum row alignment):

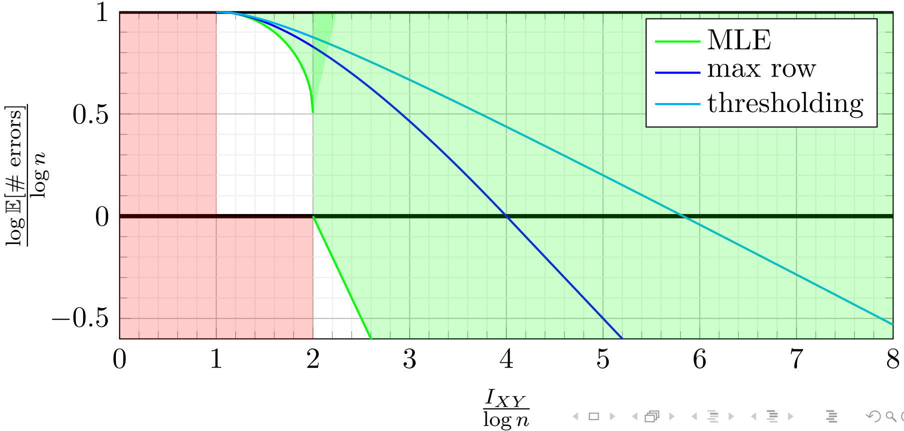
- High correlation: $x \geq 2 \implies y < 2 - \frac{x}{2}$
- Low correlation: $2 \geq x > 1 \implies y < 1 - (1 - \sqrt{x})^2$



Gaussian alignment

Achievability (thresholding):

- $x > 1 \implies y < 1 - \frac{x}{4} \cdot (1 - 1/x)^2$

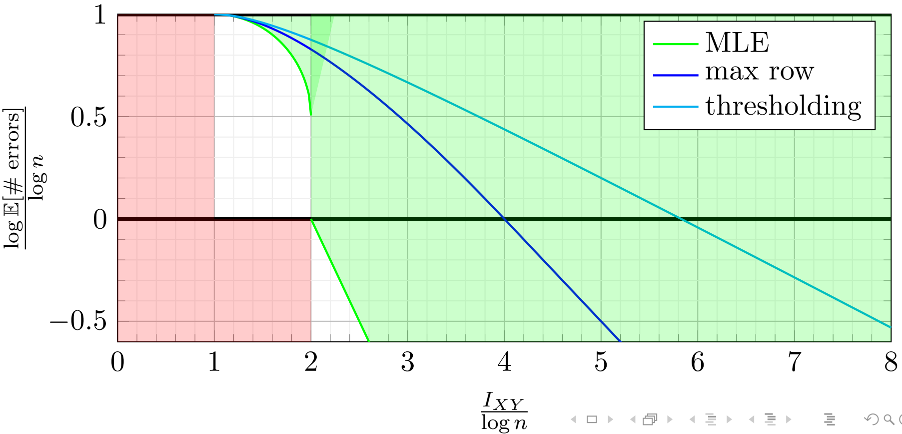


Gaussian alignment

All three algorithms make $o(n)$ errors right above the converse (Consistent with [6]).

---

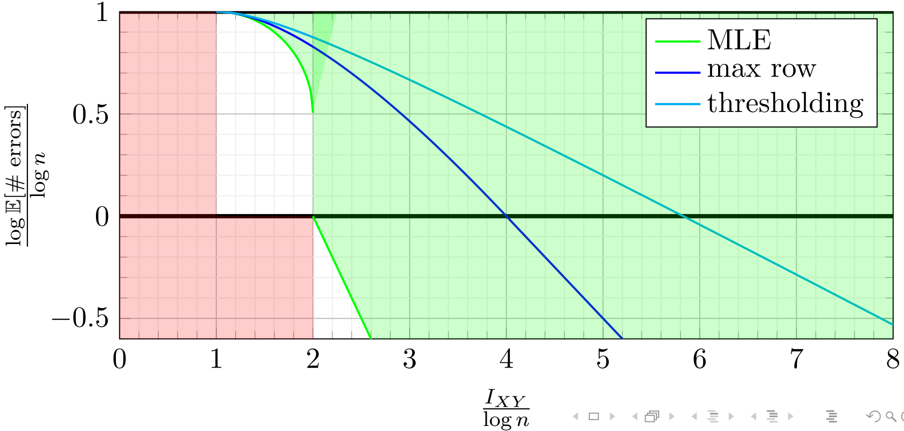
[6] Farhad Shirani, Siddharth Garg, and Elza Erkip on discrete features, A Concentration of Measure Approach to Database De-anonymization, ISIT 2019



Gaussian alignment

All three algorithms make $o(n)$ errors right above the converse (Consistent with [6]).

[6] Farhad Shirani, Siddharth Garg, and Elza Erkip on discrete features, A Concentration of Measure Approach to Database De-anonymization, ISIT 2019



Database alignment

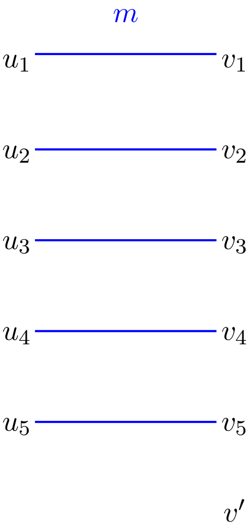
- Understanding the problem of database alignment provides insight on information theoretic quantities that characterize limits to alignment in general.

Outline of analysis

High-correlation achievability for MLE

Decomposition of misalignments

Graph representation

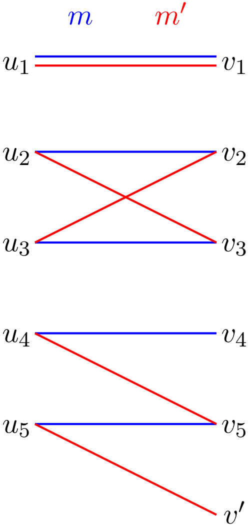


Matrix representation

$$\begin{array}{c}\mathbf{m}\\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0 & 0\\ 0 & 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}\end{array}$$

Decomposition of misalignments

Graph representation



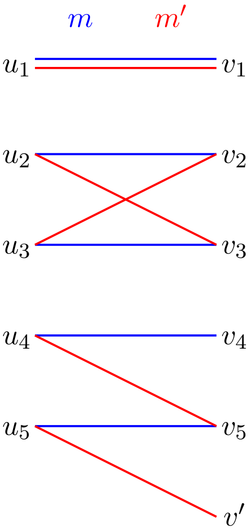
Matrix representation

$\mathbf{m}'$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Decomposition of misalignments

Graph representation



Matrix representation

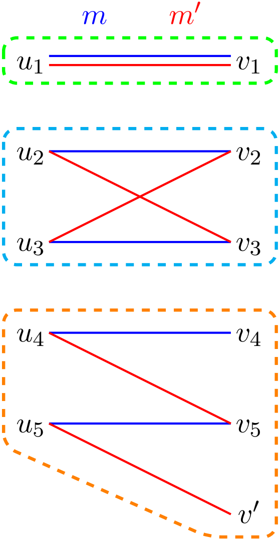
$$\mathbf{m'} - \mathbf{m}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & +1 & 0 & 0 & 0 \\ 0 & +1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & +1 & 0 \\ 0 & 0 & 0 & 0 & -1 & +1 \end{bmatrix}$$

Decomposition of misalignments

Graph representation Matrix representation

Components $\longleftrightarrow$ Blocks



$$\mathbf{m}' - \mathbf{m}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & +1 & 0 & 0 & 0 \\ 0 & +1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & +1 & 0 \\ 0 & 0 & 0 & 0 & -1 & +1 \end{bmatrix}$$

Decomposition of misalignments

- Consider m' such that $\mathbf{m}' - \mathbf{m}$ induces multiple blocks:
  $\mathbf{m}' - \mathbf{m} = \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & \mathbf{H}_2 \end{bmatrix}.$

Decomposition of misalignments

- Consider m' such that $\mathbf{m'} - \mathbf{m}$ induces multiple blocks:
  $\mathbf{m'} - \mathbf{m} = \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & \mathbf{H}_2 \end{bmatrix}$.
- There exist m'_1 and m'_2 for which
  $\mathbf{m'_1} - \mathbf{m} = \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & 0 \end{bmatrix}$ and $\mathbf{m'_2} - \mathbf{m} = \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{H}_2 \end{bmatrix}$. These correspond to a 'partition' of the components that correspond to the blocks.

Decomposition of misalignments

- Consider m' such that $\mathbf{m'} - \mathbf{m}$ induces multiple blocks: $\mathbf{m'} - \mathbf{m} = \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & \mathbf{H}_2 \end{bmatrix}$.
- There exist m'_1 and m'_2 for which $\mathbf{m'_1} - \mathbf{m} = \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & 0 \end{bmatrix}$ and $\mathbf{m'_2} - \mathbf{m} = \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{H}_2 \end{bmatrix}$. These correspond to a 'partition' of the components that correspond to the blocks.
- The difference in $\langle \mathbf{G}, \cdot \rangle$ decomposes:

$$\left\langle \mathbf{G}, \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & \mathbf{H}_2 \end{bmatrix} \right\rangle = \left\langle \mathbf{G}, \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & 0 \end{bmatrix} \right\rangle + \left\langle \mathbf{G}, \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{H}_2 \end{bmatrix} \right\rangle$$

$$\langle \mathbf{G}, \mathbf{m}' - \mathbf{m} \rangle = \langle \mathbf{G}, \mathbf{m}'_1 - \mathbf{m} \rangle + \langle \mathbf{G}, \mathbf{m}'_2 - \mathbf{m} \rangle$$

Decomposition of misalignments

- Consider m' such that $\mathbf{m'} - \mathbf{m}$ induces multiple blocks: $\mathbf{m'} - \mathbf{m} = \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & \mathbf{H}_2 \end{bmatrix}$.
- There exist m'_1 and m'_2 for which $\mathbf{m'_1} - \mathbf{m} = \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & 0 \end{bmatrix}$ and $\mathbf{m'_2} - \mathbf{m} = \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{H}_2 \end{bmatrix}$. These correspond to a 'partition' of the components that correspond to the blocks.
- The difference in $\langle \mathbf{G}, \cdot \rangle$ decomposes:

$$\left\langle \mathbf{G}, \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & \mathbf{H}_2 \end{bmatrix} \right\rangle = \left\langle \mathbf{G}, \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & 0 \end{bmatrix} \right\rangle + \left\langle \mathbf{G}, \begin{bmatrix} 0 & 0 \\ 0 & \mathbf{H}_2 \end{bmatrix} \right\rangle$$
$$\langle \mathbf{G}, \mathbf{m'} - \mathbf{m} \rangle = \langle \mathbf{G}, \mathbf{m'}_1 - \mathbf{m} \rangle + \langle \mathbf{G}, \mathbf{m'}_2 - \mathbf{m} \rangle$$

- Therefore, if $\langle \mathbf{G}, \mathbf{m'} \rangle - \langle \mathbf{G}, \mathbf{m} \rangle = \langle \mathbf{G}, \mathbf{m'} - \mathbf{m} \rangle \geq 0$,

  then either $\langle \mathbf{G}, \mathbf{m'}_1 - \mathbf{m} \rangle \geq 0$
  or $\langle \mathbf{G}, \mathbf{m'}_2 - \mathbf{m} \rangle \geq 0$.

Decomposition of misalignments

- If $\langle \mathbf{G}, \mathbf{m}' \rangle - \langle \mathbf{G}, \mathbf{m} \rangle = \langle \mathbf{G}, \mathbf{m}' - \mathbf{m} \rangle \geq 0$,

  then either $\langle \mathbf{G}, \mathbf{m}'_1 - \mathbf{m} \rangle \geq 0$
  or $\langle \mathbf{G}, \mathbf{m}'_2 - \mathbf{m} \rangle \geq 0$.

Decomposition of misalignments

- If $\langle \mathbf{G}, \mathbf{m'} \rangle - \langle \mathbf{G}, \mathbf{m} \rangle = \langle \mathbf{G}, \mathbf{m'} - \mathbf{m} \rangle \geq 0$,

  then either $\langle \mathbf{G}, \mathbf{m'}_1 - \mathbf{m} \rangle \geq 0$
  or $\langle \mathbf{G}, \mathbf{m'}_2 - \mathbf{m} \rangle \geq 0$.

- Then we can limit our attention to 'single-component'/'single-block' misalignments.

Decomposition of misalignments

- Let m be the true mapping and m' some false mapping.

Decomposition of misalignments

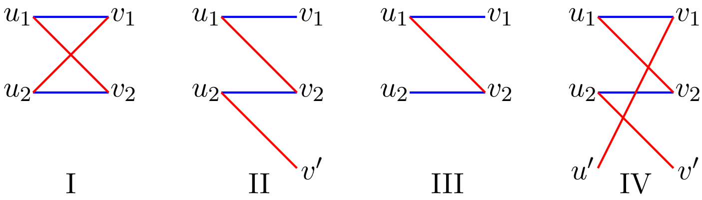
- Let m be the true mapping and m' some false mapping.
- Every user is mapped at most once by m and at most once by m', i.e. every node has at most one blue and at most one red edge.

Decomposition of misalignments

- Let m be the true mapping and m' some false mapping.
- Every user is mapped at most once by m and at most once by m', i.e. every node has at most one blue and at most one red edge.
- Then all components are paths or cycles.

Decomposition of misalignments

- All components are paths or cycles.

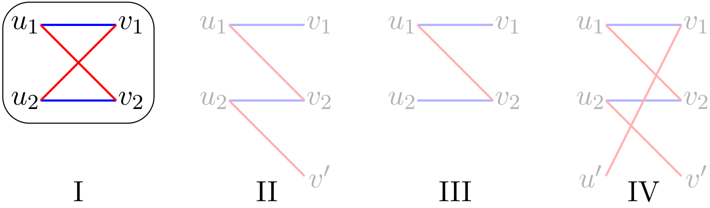


- I: cycle
  II: balanced path

  III: m-dominant path
  IV: m'-dominant path

Decomposition of misalignments

- All components are paths or cycles.



- I: cycle
  II: balanced path
  III: m-dominant path
  IV: m'-dominant path
- If $M = m$ matches all users, i.e. $|M| = |\mathcal{U}| = |\mathcal{V}|$, then there are no paths and all components are cycles.

Error bounds

Assume $|M| = |\mathcal{U}| = |\mathcal{V}|$, so only consider cycle-inducing false mappings.
If m' a false mapping that induces a **cycle of length 4**
with true mapping $M = m$, then the Chernoff bound gives us exactly:

Error bounds

Assume $|M| = |\mathcal{U}| = |\mathcal{V}|$, so only consider cycle-inducing false mappings.
If m' a false mapping that induces a **cycle of length 4**
with true mapping $M = m$, then the Chernoff bound gives us exactly:

- Finite alphabet database:

$$\mathbb{P}[\langle \mathbf{G}, \mathbf{m}' - \mathbf{m} \rangle \geq 0 | M = m] \leq \exp\left(-I_2^\circ\right)$$

Error bounds

Assume $|M| = |\mathcal{U}| = |\mathcal{V}|$, so only consider cycle-inducing false mappings.
If m' a false mapping that induces a **cycle of length 4** with true mapping $M = m$, then the Chernoff bound gives us exactly:

- Finite alphabet database:

$$\mathbb{P}[\langle \mathbf{G}, \mathbf{m}' - \mathbf{m} \rangle \geq 0 | M = m] \leq \exp\left(-I_2^{\circ}\right)$$

- Gaussian database:

$$\mathbb{P}[\langle \mathbf{G}, \mathbf{m}' - \mathbf{m} \rangle \geq 0 | M = m] \leq \exp\left(-I_{XY}\right)$$

Error bounds

Assume $|M| = |\mathcal{U}| = |\mathcal{V}|$, so only consider cycle-inducing false mappings.
If m' a false mapping that induces a **cycle of length** 2δ with true mapping $M = m$,
then Chernoff bound upper bounded by:

- Finite alphabet database:

$$\mathbb{P}[\langle \mathbf{G}, \mathbf{m'} - \mathbf{m} \rangle \geq 0 | M = m] \leq \exp\left(-I_2^\circ \cdot \frac{\delta}{2}\right)$$

- Gaussian database:

$$\mathbb{P}[\langle \mathbf{G}, \mathbf{m'} - \mathbf{m} \rangle \geq 0 | M = m] \leq \exp\left(-I_{XY} \cdot \frac{\delta}{2}\right)$$

Error bounds

Assume $|M| = |\mathcal{U}| = |\mathcal{V}|$, so only consider cycle-inducing false mappings.
If m' a false mapping that induces a **cycle of length** 2δ
with true mapping $M = m$,
then Chernoff bound upper bounded by:

- Finite alphabet database:

$$\mathbb{P}[\langle \mathbf{G}, \mathbf{m'} - \mathbf{m} \rangle \geq 0 | M = m] \leq \exp\left(-I_2^\circ \cdot \frac{\delta}{2}\right)$$

- Gaussian database:

$$\mathbb{P}[\langle \mathbf{G}, \mathbf{m'} - \mathbf{m} \rangle \geq 0 | M = m] \leq \exp\left(-I_{XY} \cdot \frac{\delta}{2}\right)$$

Henceforth we use to I without subscript to refer to either I_2° or I_{XY}.

Error bounds

- $|M| = |\mathcal{U}| = |\mathcal{V}| = n$
  There are $\binom{n}{\delta}(\delta-1)!$ false mappings m' that induce a cycle of length 2δ on m.

  $\binom{n}{\delta}(\delta-1)!$ is upper bounded by $\frac{n^\delta}{\delta}$.

Error bounds

- $|M| = |\mathcal{U}| = |\mathcal{V}| = n$
  There are $\binom{n}{\delta}(\delta-1)!$ false mappings m' that induce a cycle of length 2δ on m.

  $\binom{n}{\delta}(\delta-1)!$ is upper bounded by $\frac{n^\delta}{\delta}$.

- Let $\widehat{m}$ denote the ML estimate.
  The expected number of cycles of length 2δ contained in $\widehat{m} - m$ is bounded by

$$\frac{n^\delta}{\delta}\exp\left(-\delta\cdot\frac{I}{2}\right) = \frac{1}{\delta}\exp\left(-\frac{\delta}{2}(I - 2\log n)\right)$$

Error bounds

- $|M| = |\mathcal{U}| = |\mathcal{V}| = n$
  There are $\binom{n}{\delta}(\delta-1)!$ false mappings m' that induce a cycle of length 2δ on m.

  $\binom{n}{\delta}(\delta-1)!$ is upper bounded by $\frac{n^\delta}{\delta}$.

- Let $\widehat{m}$ denote the ML estimate.
  The expected number of cycles of length 2δ contained in $\widehat{m} - m$ is bounded by

$$\frac{n^\delta}{\delta}\exp\left(-\delta \cdot \frac{I}{2}\right) = \frac{1}{\delta}\exp\left(-\frac{\delta}{2}(I - 2\log n)\right)$$

- A cycle of length 2δ results in δ errors.
  The total number of errors in expectation is upper bounded by:

$$\sum_{\delta=2}^{n}\exp\left(-\frac{\delta}{2}(I - 2\log n)\right)$$

Error bounds

- The total number of errors in expectation is upper bounded by:

$$\sum_{\delta=2}^{n} \exp\left(-\frac{\delta}{2}(I - 2\log n)\right)$$

Error bounds

- The total number of errors in expectation is upper bounded by:

$$\sum_{\delta=2}^{n} \exp\left(-\frac{\delta}{2}(I - 2\log n)\right)$$

- If $I > 2\log n$,
  the expression is upper bounded by $\frac{a^2}{1-a} \leq \mathcal{O}(1)$,
  where $a = \exp\left(-\frac{I-2\log n}{2}\right)$.

Error bounds

- The total number of errors in expectation is upper bounded by:

$$\sum_{\delta=2}^{n} \exp\left(-\frac{\delta}{2}(I - 2\log n)\right)$$

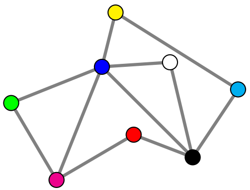
- If $I > 2\log n$,
  the expression is upper bounded by $\frac{a^2}{1-a} \leq \mathcal{O}(1)$,
  where $a = \exp\left(-\frac{I-2\log n}{2}\right)$.
- If $I > 2\log n + \omega(1)$, this converges to $o(1)$.

Graph alignment

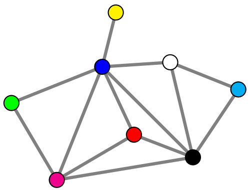
Graph alignment

In graph alignment, the data consists of edge information.

Network # 1



Network # 2

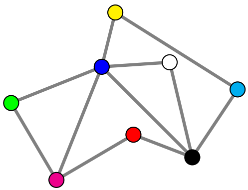


Graph alignment

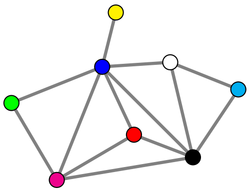
In graph alignment, the data consists of edge information.

The correlated graph model:

Network # 1



Network # 2



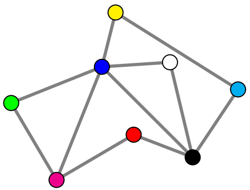
Graph alignment

In graph alignment, the data consists of edge information.

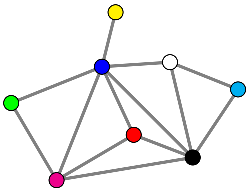
The correlated graph model:

- Edges are i.i.d. random variables.

Network # 1



Network # 2



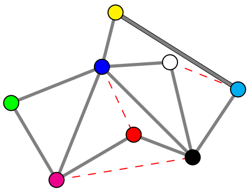
Graph alignment

In graph alignment, the data consists of edge information.

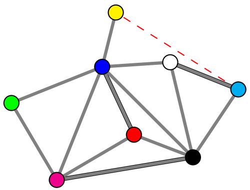
The correlated graph model:

- Edges are i.i.d. random variables.
- Edges are more likely to appear in both graphs,
  than to **appear in one** but not the other.

Network # 1



Network # 2



Graph alignment

Classical scenario: Edges are Bernouilli random variables taking values in {edge,non-edge}.

Graph alignment

Classical scenario: Edges are Bernouilli random variables taking values in {edge,non-edge}.

Other settings of interest:

Graph alignment

Classical scenario: Edges are Bernouilli random variables taking values in {edge,non-edge}.

Other settings of interest:

- Arbitrary discrete alphabet.

  Allows modelling of different types of connections in networks.

  This is a generalization of the Bernouilli case.

Graph alignment

Classical scenario: Edges are Bernouilli random variables taking values in {edge,non-edge}.

Other settings of interest:

- Arbitrary discrete alphabet.

  Allows modelling different types of connections in networks.

  This is a generalization of the Bernouilli case.

- Continuous vector space.

  One case of particular interest is that of graphs with Gaussian weights [7], [8].

[7] Zhou Fan, Cheng Mao, Yihong Wu and Jiaming Xu, Spectral Graph Matching and Regularized Quadratic Relaxations I:The Gaussian Model, 2019

[8] Luca Ganassali, Sharp Threshold for Alignment of Graph Databases with Gaussian Weights 2020

Graph alignment - likelihood function

- f_X, f_Y: marginal pdfs (or pmfs) of edges in two graphs
- f_{XY}: joint pdf (or pmf) of correlated edge pair
- $\mathcal{W}_M \subset \mathcal{U} \times \mathcal{V}$: vertex pairs mapped by M

Let $\binom{\mathcal{S}}{2}$ denote set of all subsets of $\mathcal{S}$ of size 2.

Proxy for log-likelihood of graphs:

$$\sum_{\{(u_i,v_i),(u_j,v_j)\}\in\binom{\mathcal{W}_M}{2}} \log \frac{f_{XY}(A\{u_i,u_j\}, B\{v_i,v_j\})}{f_X(A\{u_i,u_j\}) f_Y(B\{v_i,v_j\})}$$

Graph alignment - likelihood function

- f_X, f_Y: marginal pdfs (or pmfs) of edges in two graphs
- f_{XY}: joint pdf (or pmf) of correlated edge pair
- $\mathcal{W}_M \subset \mathcal{U} \times \mathcal{V}$: vertex pairs mapped by M

Let $\binom{\mathcal{S}}{2}$ denote set of all subsets of $\mathcal{S}$ of size 2.

Proxy for log-likelihood of graphs:

$$\sum_{\{(u_i,v_i),(u_j,v_j)\} \in \binom{\mathcal{W}_M}{2}} \log \frac{f_{XY}(A\{u_i,u_j\}, B\{v_i,v_j\})}{f_X(A\{u_i,u_j\}) f_Y(B\{v_i,v_j\})} = \frac{1}{2} \overrightarrow{M}^\top \mathbf{G} \overrightarrow{M}$$

where $\mathbf{G} \in \mathbb{R}^{(\mathcal{U}\times\mathcal{V})\times(\mathcal{U}\times\mathcal{V})}$ information density matrix,
and $\overrightarrow{M} \in \{0,1\}^{(\mathcal{U}\times\mathcal{V})}$ encodes the mapping M.

Graph alignment - MLE

MLE for graph alignment is equivalent to the following optimization:

$$\max_{\vec{m}} \vec{m}^\top \mathbf{G} \vec{m} \qquad \text{s.t.} \qquad \begin{aligned} &\sum_v m_{(u,v)} = 1 && \forall u \in \mathcal{U} \\ &\sum_u m_{(u,v)} = 1 && \forall v \in \mathcal{V} \\ &\vec{m} \in \{0,1\}^{(\mathcal{U} \times \mathcal{V})} \end{aligned}$$

Databases vs Graphs

Database alignment

MLE given by linear optimization:

$\max \langle \mathbf{G}, \mathbf{m} \rangle = \operatorname{tr}(\mathbf{G}^\top \mathbf{m})$
over $\mathbf{m} \in \{0,1\}^{\mathcal{U}\times\mathcal{V}}$ with row and column sums equal to 1.

$\sim$ linear assignment problem
$\mathcal{O}(n^3)$

Graph alignment

MLE given by quadratic optimization:

$\max \overrightarrow{m}^\top \mathbf{G} \overrightarrow{m} = \operatorname{tr}(\mathbf{G} \overrightarrow{m} \overrightarrow{m}^\top)$
over $\overrightarrow{m} \in \{0,1\}^{(\mathcal{U}\times\mathcal{V})}$ with 'row' and 'column' sums equal to 1.

$\sim$ quadratic assignment problem
NP-hard

Graph alignment

Most well studied model: Correlated Erdős-Rényi

Graph alignment

Most well studied model: Correlated Erdős-Rényi

- Pair of graphs $G_A = (V; E_A)$ and $G_B = (V; E_B)$ on $|V| = n$ vertices.

Graph alignment

Most well studied model: Correlated Erdős-Rényi

- Pair of graphs $G_A = (V; E_A)$ and $G_B = (V; E_B)$ on $|V| = n$ vertices.
- Each graph Erdős-Rényi with average degree np.

Graph alignment

Most well studied model: Correlated Erdős-Rényi

- Pair of graphs $G_A = (V; E_A)$ and $G_B = (V; E_B)$ on $|V| = n$ vertices.
- Each graph Erdős-Rényi with average degree np.
- Intersection of graphs $(V; E_A \cap E_B)$ is Erdős-Rényi with avg. deg. nps

Graph alignment

Most well studied model: Correlated Erdős-Rényi

- Pair of graphs $G_A = (V; E_A)$ and $G_B = (V; E_B)$ on $|V| = n$ vertices.
- Each graph Erdős-Rényi with average degree np.
- Intersection of graphs $(V; E_A \cap E_B)$ is Erdős-Rényi with avg. deg. nps
  Difference of graphs $(V; E_A \setminus E_B)$ is Erdős-Rényi with avg. deg. $np(1-s)$

Graph alignment

Most well studied model: Correlated Erdős-Rényi

- Pair of graphs $G_A = (V; E_A)$ and $G_B = (V; E_B)$ on $|V| = n$ vertices.
- Each graph Erdős-Rényi with average degree np.
- Intersection of graphs $(V; E_A \cap E_B)$ is Erdős-Rényi with avg. deg. nps
  Difference of graphs $(V; E_A \setminus E_B)$ is Erdős-Rényi with avg. deg. $np(1-s)$

Generate graphs by independently generating edge random variable for each pair of vertices.

$$\Pr(\text{edge in } E_A \cap E_B) = ps$$
$$\Pr(\text{edge in } E_A \setminus E_B) = p(1-s)$$
$$\Pr(\text{edge in } E_B \setminus E_A) = p(1-s)$$
$$\Pr(\text{edge not in } E_A \cup E_B) = 1 - p(2-s)$$

Graph alignment

Results

Graph alignment

y-axis: strength of signal

0

average degree $\Theta(n^y)$ in intersection graph

$\frac{\log ps}{\log n}$

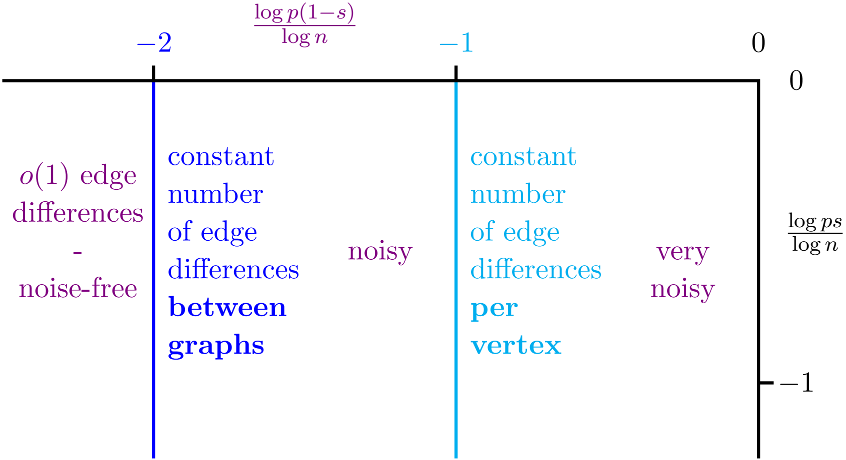
constant average degree in intersection graph

-1

average degree $o(1)$ in intersection graph

Graph alignment

x-axis: strength of noise $\quad$ y-axis: strength of signal



Graph alignment

Graphs are positively correlated if

$$\Pr(\text{in neither})\Pr(\text{in both}) > \Pr(\text{in } E_A \setminus E_B)\Pr(\text{in } E_A \setminus E_B)$$

Graph alignment

Graphs are positively correlated if

$$\begin{aligned} &\Pr(\text{in neither})\Pr(\text{in both}) > \Pr(\text{in } E_A \setminus E_B)\Pr(\text{in } E_A \setminus E_B) \\ &\iff [1 - p(2-s)]ps > p^2(1-s)^2 \end{aligned}$$

Graph alignment

Graphs are positively correlated if

$$\begin{aligned} \Pr(\text{in neither}) \Pr(\text{in both}) &> \Pr(\text{in } E_A \setminus E_B) \Pr(\text{in } E_A \setminus E_B) \\ \iff [1 - p(2-s)]ps &> p^2(1-s)^2 \\ \iff s &> p \end{aligned}$$

Graph alignment

Graphs are positively correlated if

$$
\begin{aligned}
\Pr(\text{in neither})\Pr(\text{in both}) &> \Pr(\text{in } E_A \setminus E_B)\Pr(\text{in } E_A \setminus E_B) \\
\iff [1-p(2-s)]ps &> p^2(1-s)^2 \\
\iff s &> p \\
\iff \frac{y}{x} = \frac{\log ps/\log n}{\log p(1-s)/\log n} &> \frac{\log p^2}{\log p(1-p)}
\end{aligned}
$$

Graph alignment

Graphs are positively correlated if

$$
\begin{aligned}
\Pr(\text{in neither})\Pr(\text{in both}) &> \Pr(\text{in } E_A \setminus E_B)\Pr(\text{in } E_A \setminus E_B) \\
\iff [1-p(2-s)]ps &> p^2(1-s)^2 \\
\iff s &> p
\end{aligned}
$$

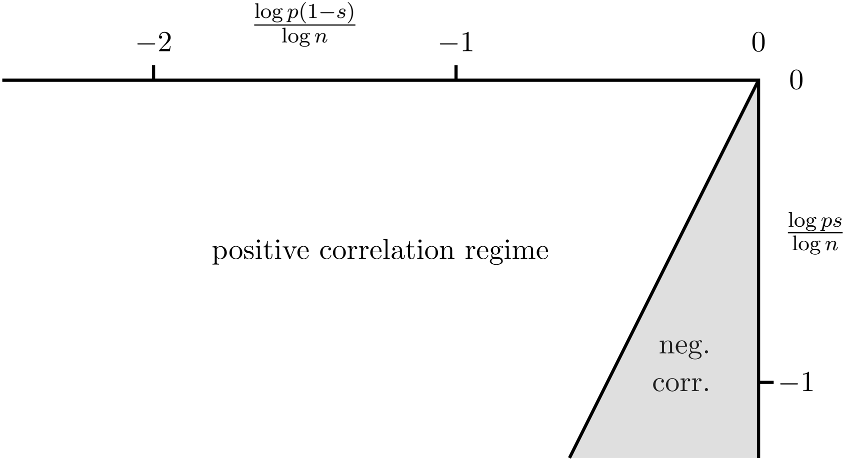
$$\iff \frac{y}{x} = \frac{\log ps/\log n}{\log p(1-s)/\log n} > \frac{\log p^2}{\log p(1-p)}$$

In the sparse regime $p \leq o(1)$, this corresponds to

$$\frac{y}{x} = \frac{\log ps/\log n}{\log p(1-s)/\log n} > 2.$$

Graph alignment

We only present results on positively correlated graphs.



Graph alignment

Noiseless case: $p(1-s) \leq o(n^2)$

Edward M. Wright - 1971 [9]

Sufficient and **necessary** condition for noiseless case:

Alignment possible with probability $1 - o(1)$ if and only if

$$np \geq \log n + \omega(1).$$

[9] Graphs on unlabeled nodes with a given number of edges, Acta Mathematica 1971

Graph alignment

Noiseless case: $p(1-s) \leq o(n^2)$

Edward M. Wright - 1971 [9]

Sufficient and **necessary** condition for noiseless case:
Alignment possible with probability $1 - o(1)$ if and only if

$$np \geq \log n + \omega(1).$$

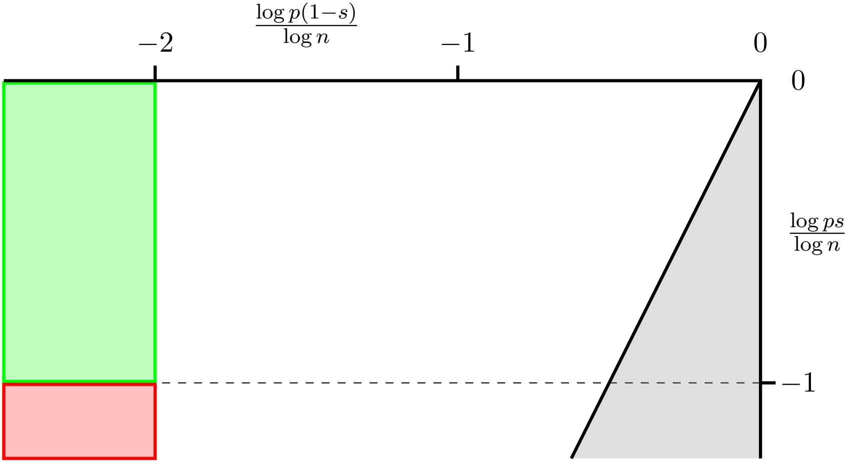
The cut-off corresponds to the line

$$y = \frac{\log ps}{\log n} = -1 + \frac{\log \log n}{\log n} + \frac{\log s}{\log n} = -1 \pm o(1)$$

[9] Graphs on unlabeled nodes with a given number of edges, Acta Mathematica 1971

Graph alignment

x-axis: strength of noise $\quad$ y-axis: strength of signal



Graph alignment

Noiseless case: $p(1-s) \leq o(n^2)$

Sufficient cond. for **polynomial-time** alignment:

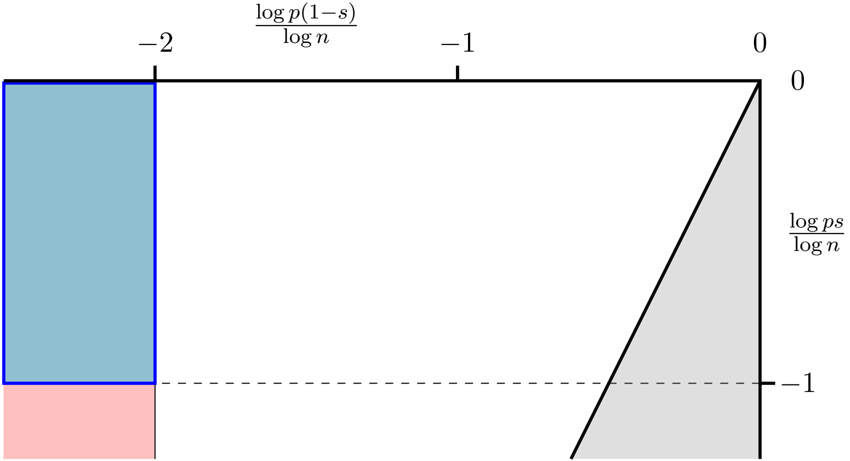
Polynomial-time algorithms that achieve alignment with probability $1 - o(1)$ if $np \geq \log n + \omega(1)$

[10] Bla Bollobs, Distinguishing vertices of random graphs, North-Holland Mathematics Studies 1982

[11] Tomek Czajka and Gopal Pandurangan, Improved random graph isomorphism, Journal of Discrete Algorithms 2008

Graph alignment

x-axis: strength of noise y-axis: strength of signal



Graph alignment

Information theoretic bound formulations for sparse regime $p = o(1)$

- Pedram Pedarsani and Matthias Grossglauser 2011 [12]
  Sufficient condition: $nps\left(\frac{s}{2-s}\right) \geq 8\log n + \omega(1)$

[12] Pedram Pedarsani and Matthias Grossglauser, On the Privacy of Anonymized Networks, SIGKDD 2011

Graph alignment

Information theoretic bound formulations for sparse regime $p = o(1)$

- Pedram Pedarsani and Matthias Grossglauser 2011 [12]

  Sufficient condition: $nps\left(\frac{s}{2-s}\right) \geq 8\log n + \omega(1)$

  This implies

$$ps > \omega(1/n) \text{ and } \frac{(ps)^2}{ps + 2p(1-s)} > \omega(1/n)$$

[12] Pedram Pedarsani and Matthias Grossglauser, On the Privacy of Anonymized Networks, SIGKDD 2011

Graph alignment

Information theoretic bound formulations for sparse regime $p = o(1)$

- Pedram Pedarsani and Matthias Grossglauser 2011 [12]
  Sufficient condition: $nps\left(\frac{s}{2-s}\right) \geq 8\log n + \omega(1)$
  This implies

$$ps > \omega(1/n) \text{ and } \frac{(ps)^2}{ps + 2p(1-s)} > \omega(1/n)$$

For small ps, the latter implies

$$2\log ps - \log p(1-s) > -\log n$$

or

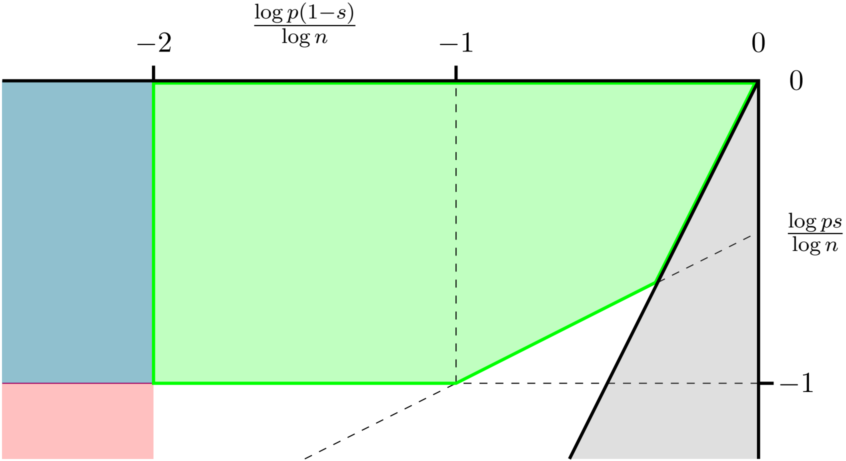
$$2y - x = -1$$

---

[12] Pedram Pedarsani and Matthias Grossglauser, On the Privacy of Anonymized Networks, SIGKDD 2011

Graph alignment

x-axis: strength of noise y-axis: strength of signal



Graph alignment

Information-theoretic bounds for sparse regime $p = \mathcal{O}(1/\log n)$

- Pedram Pedarsani and Matthias Grossglauser 2011 [12]

  Sufficient condition: $nps\left(\frac{s}{2-s}\right) \geq 8\log n + \omega(1)$

---

[12] Pedram Pedarsani and Matthias Grossglauser , On the Privacy of Anonymized Networks, SIGKDD 2011

[13] Daniel Cullina and Negar Kiyavash, Improved Achievability and Converse Bounds for Erdos-Renyi Graph Matching, Sigmetrics 2016

[14] Daniel Cullina and Negar Kiyavash, Exact alignment recovery for correlated Erdos-Renyi graphs, 2017

Graph alignment

Information-theoretic bounds for sparse regime $p = \mathcal{O}(1/\log n)$

- Pedram Pedarsani and Matthias Grossglauser 2011 [12]
  Sufficient condition: $nps\left(\frac{s}{2-s}\right) \geq 8\log n + \omega(1)$
- Daniel Cullina and Negar Kiyavash 2016 [13]
  Sufficient condition: $nps\left(1 - \frac{p(1-s)}{\sqrt{ps}}\right)^2 \geq 2\log n + \omega(1)$
  Necessary condition: $nps > \log n(1 - \Omega(1))$

[12] Pedram Pedarsani and Matthias Grossglauser , On the Privacy of Anonymized Networks, SIGKDD 2011

[13] Daniel Cullina and Negar Kiyavash, Improved Achievability and Converse Bounds for Erdos-Renyi Graph Matching, Sigmetrics 2016

[14] Daniel Cullina and Negar Kiyavash, Exact alignment recovery for correlated Erdos-Renyi graphs, 2017

Graph alignment

Information-theoretic bounds for sparse regime $p = \mathcal{O}(1/\log n)$

- Pedram Pedarsani and Matthias Grossglauser 2011 [12]
  Sufficient condition: $nps\left(\frac{s}{2-s}\right) \geq 8\log n + \omega(1)$
- Daniel Cullina and Negar Kiyavash 2016 [13]
  Sufficient condition: $nps\left(1 - \frac{p(1-s)}{\sqrt{ps}}\right)^2 \geq 2\log n + \omega(1)$
  Necessary condition: $nps > \log n(1 - \Omega(1))$
- Daniel Cullina and Negar Kiyavash - 2017 [14]
  Sufficient condition: $nps \geq \log n + \omega(1)$

$$p(1-s) \leq \mathcal{O}(1/\log n) \text{ and } \frac{p(1-s)}{\sqrt{ps}} \leq \mathcal{O}(1/\log^{3/2} n)$$

---

[12] Pedram Pedarsani and Matthias Grossglauser , On the Privacy of Anonymized Networks, SIGKDD 2011

[13] Daniel Cullina and Negar Kiyavash, Improved Achievability and Converse Bounds for Erdos-Renyi Graph Matching, Sigmetrics 2016

[14] Daniel Cullina and Negar Kiyavash, Exact alignment recovery for correlated Erdos-Renyi graphs, 2017

Graph alignment

Information-theoretic bounds for sparse regime $p = \mathcal{O}(1/\log n)$

- Pedram Pedarsani and Matthias Grossglauser - 2011 [12]
  Sufficient condition: $nps\left(\frac{s}{2-s}\right) \geq 8\log n + \omega(1)$
- Daniel Cullina and Negar Kiyavash - 2016 [13]
  Sufficient condition: $nps\left(1 - \frac{p(1-s)}{\sqrt{ps}}\right)^2 \geq 2\log n + \omega(1)$
  Necessary condition: $nps > \log n(1 - \Omega(1))$
- Daniel Cullina and Negar Kiyavash - 2017 [14]
  Sufficient condition: $nps \geq \log n + \omega(1)$

$$\frac{p(1-s)}{\sqrt{ps}} \leq \mathcal{O}(1/\log^{3/2} n)$$

---

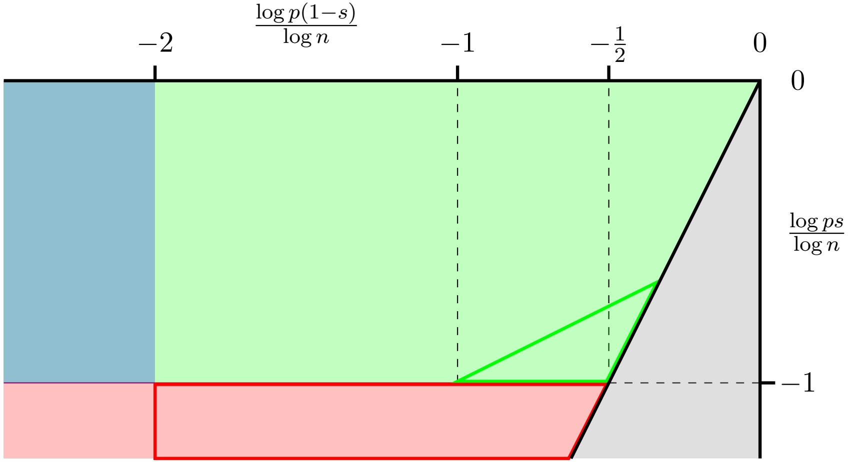
[12] Pedram Pedarsani and Matthias Grossglauser , On the Privacy of Anonymized Networks, SIGKDD 2011

[13] Daniel Cullina and Negar Kiyavash, Improved Achievability and Converse Bounds for Erdos-Renyi Graph Matching, Sigmetrics 2016

[14] Daniel Cullina and Negar Kiyavash, Exact alignment recovery for correlated Erdos-Renyi graphs, 2017

Graph alignment

x-axis: strength of noise $\qquad$ y-axis: strength of signal



Graph alignment

Regime of particular interest:

- Most algorithmic results focus on

$$-\log p \geq \Omega(\log n) \text{ and } -\log s \leq o(\log n)$$

  sparse graphs and s does not go to zero too quickly.

Graph alignment

Regime of particular interest:

- Most algorithmic results focus on

$$-\log p \geq \Omega(\log n) \text{ and } -\log s \leq o(\log n)$$

  sparse graphs and s does not go to zero too quickly.

Graph alignment

Regime of particular interest:

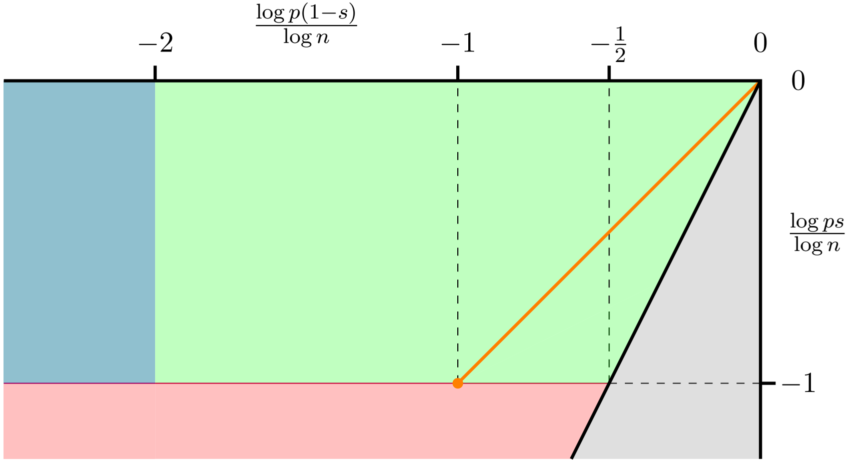
- Most algorithmic results focus on

$$-\log p \geq \Omega(\log n) \text{ and } -\log s \leq o(\log n)$$

  sparse graphs and s does not go to zero too quickly.
- This entire regime of interest is contained on line $\frac{y}{x} = \frac{\log p + \log s}{\log p + \log(1-s)} = 1$.

Graph alignment

x-axis: strength of noise $\qquad$ y-axis: strength of signal



Graph alignment

Polynomial-time algorithms for exact alignment in the regime $(1-s) \le \mathcal{O}(1)$:

- Jian Ding, Zongming Ma, Yihong Wu and Jiaming Xu [15]
  $np \ge (\log n)^c$ and $(1-s) \le (\log n)^{-c}$
- Zhou Fan, Cheng Mao, Yihong Wu and Jiaming Xu [16]
  $np \ge (\log n)^c$ and $(1-s) \le (\log n)^{-c}$
- Cheng Mao, Mark Rudelson and Konstantin Tikhomirov [17]
  $np \ge (\log n)^c$ and $(1-s) \le (\log \log n)^{-c}$
- Cheng Mao, Mark Rudelson and Konstantin Tikhomirov [18]
  $n^{o(1)} \ge np \ge \log n(1+\varepsilon)$ and $(1-s) \le \min\{\text{constant}, \varepsilon\}$

---

[15] Efficient random graph matching via degree profiles, Probability Theory and Related Field 2021

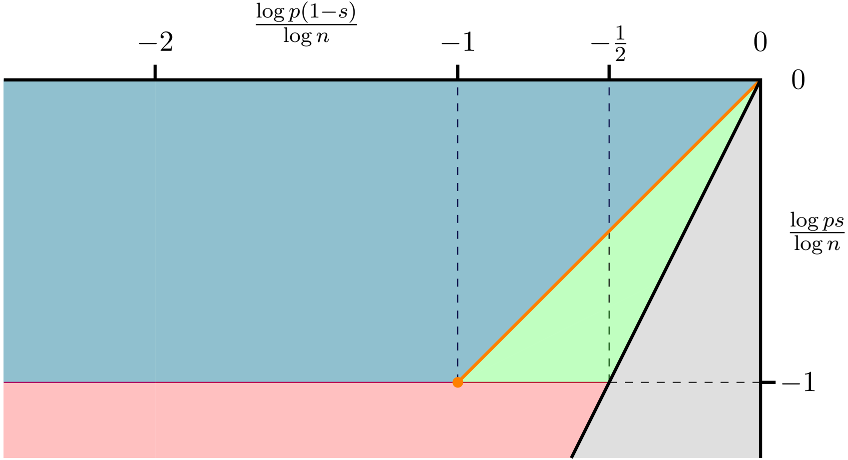
[16] Spectral graph matching and regularized quadratic relaxations II: ErdosRenyi graphs and universality, 2019

[17] Random graph matching with improved noise robustness, Conference on Learning Theory 2021

[18] Exact Matching of Random Graphs with Constant Correlation, Conference on Learning Theory 2021

Graph alignment

x-axis: strength of noise y-axis: strength of signal



Graph alignment

All these polynomial-time algorithms have guarantees in the regime where s is bounded away from 0.

[19] (Nearly) Efficient Algorithms for the Graph Matching Problem on Correlated Random Graphs, Advances in Neural Information Processing Systems 2019

Graph alignment

All these polynomial-time algorithms have guarantees in the regime where s is bounded away from 0.

Quasi-polynomial time algorithm for exact alignment:

- Boaz Barak, Chi-Ning Chou, Zhixian Lei, Tselil Schramm and Yueqi Sheng [19]
  $np \geq n^{o(1)}$ and $s \geq (\log n)^{-o(1)}$

[19] (Nearly) Efficient Algorithms for the Graph Matching Problem on Correlated Random Graphs, Advances in Neural Information Processing Systems 2019

Graph alignment

All these polynomial-time algorithms have guarantees in the regime where s is bounded away from 0.

Quasi-polynomial time algorithm for exact alignment:

- Boaz Barak, Chi-Ning Chou, Zhixian Lei, Tselil Schramm and Yueqi Sheng [19]
  $np \geq n^{o(1)}$ and $s \geq (\log n)^{-o(1)}$

Unlike the polynomial algorithms, this algorithm allows $s \to 0$.

[19] (Nearly) Efficient Algorithms for the Graph Matching Problem on Correlated Random Graphs, Advances in Neural Information Processing Systems 2019

Graph alignment

All these polynomial-time algorithms have guarantees in the regime where s is bounded away from 0.

Quasi-polynomial time algorithm for exact alignment:

- Boaz Barak, Chi-Ning Chou, Zhixian Lei, Tselil Schramm and Yueqi Sheng [19]
  $np \geq n^{o(1)}$ and $s \geq (\log n)^{-o(1)}$

Unlike the polynomial algorithms, this algorithm allows $s \to 0$. This is still within the regime $-\log s \leq o(\log(n))$ and therefore does guarantee any region to the right of the $y = x$ line.

[19] (Nearly) Efficient Algorithms for the Graph Matching Problem on Correlated Random Graphs, Advances in Neural Information Processing Systems 2019

Graph alignment - partial alignment

- **Exact alignment**:
  No misaligned vertices

Graph alignment - partial alignment

- **Exact alignment**:
  No misaligned vertices
- **Almost-exact alignment**:
  Vanishing fraction of misaligned vertices

Graph alignment - partial alignment

- **Exact alignment**:
  No misaligned vertices
- **Almost-exact alignment**:
  Vanishing fraction of misaligned vertices
- **Partial alignment**:
  Constant fraction of misaligned vertices

Graph alignment - partial alignment

Necessary condition:

- Exact alignment [13]

$$nps > \log n(1 - \Omega(1))$$

- Almost-exact alignment [20]

$$nps > \mathcal{O}(1)$$

- Partial alignment [20]

$$nps > 1$$

[13] Daniel Cullina and Negar Kiyavash, Improved Achievability and Converse Bounds for Erdos-Renyi Graph Matching, Sigmetrics 2016

[20] Cullina, Daniel, Negar Kiyavash, Prateek Mittal and H. Vincent Poor, Partial Recovery of Erdos-Renyi Graph Alignment via k-Core Alignment, Sigmetrics 2020

[21] Luca Ganassali, Marc Lelarge and Laurent Massoulie, Impossibility of Partial Recovery in the Graph Alignment Problem, Annual Conference on Learning Theory 2021

Graph alignment - partial alignment

Sufficient condition:

- Exact alignment [13]

$$nps \geq \log n + \omega(1)$$

- Almost-exact alignment [20]

$$nps \geq \omega(1)$$

- Partial alignment [22]

$$nps \geq \max\left\{4, \frac{2\log n}{\log(s/p)}\right\}(1 + \mathcal{O}(1))$$

---

[14] Daniel Cullina and Negar Kiyavash, Exact alignment recovery for correlated Erdos-Renyi graphs, 2017

[20] Cullina, Daniel, Negar Kiyavash, Prateek Mittal and H. Vincent Poor, Partial Recovery of Erdos-Renyi Graph Alignment via k-Core Alignment, Sigmetrics 2020

[22] Yihong Wu, Jiaming Xu and Sophie H. Yu, Settling the Sharp Reconstruction Thresholds of Random Graph Matching, 2021

Graph alignment - dense graphs

Recent work improved the information theoretic bound for dense graphs with $p/s = \Theta(1)$.

---

[13] Improved Achievability and Converse Bounds for Erdos-Renyi Graph Matching, Sigmetrics 2016

[22] Yihong Wu, Jiaming Xu and Sophie H. Yu, Settling the Sharp Reconstruction Thresholds of Random Graph Matching, 2021

Graph alignment - dense graphs

Recent work improved the information theoretic bound for dense graphs with $p/s = \Theta(1)$.

Sufficient and necessary conditions:

- Daniel Cullina and Negar Kiyavash - 2016 [13] In the regime where $p \leq \mathcal{O}(1/\log n)$

$$nps\left(1-(1-s)\sqrt{p/s}\right)^2 \geq 2\log n + \omega(1)$$

$$nps > \log n(1-\Omega(1))$$

- Yihong Wu, Jiaming Xu and Sophie H. Yu - 2021 [21]

$$nps\left(1-\sqrt{p/s}\right)^2 \geq \log n(1+o(1))$$

$$nps\left(1-\sqrt{p/s}\right)^2 > \log n(1-o(1))$$

[13] Improved Achievability and Converse Bounds for Erdos-Renyi Graph Matching, Sigmetrics 2016

[22] Yihong Wu, Jiaming Xu and Sophie H. Yu, Settling the Sharp Reconstruction Thresholds of Random Graph Matching, 2021

Thank you.