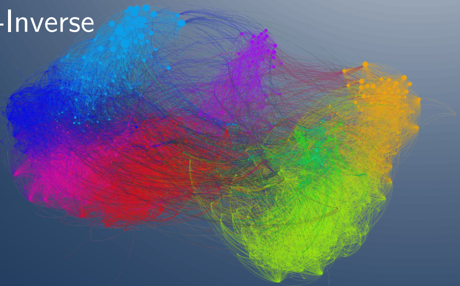
Graphs in Machine Learning

Effective Resistance Computation

Laplacian Connection and Pseudo-Inverse

Michal Valko

Inria & ENS Paris-Saclay, MVA



Partially based on material by: Ulrike von Luxburg, Gary Miller, Doyle & Schnell, Daniel Spielman

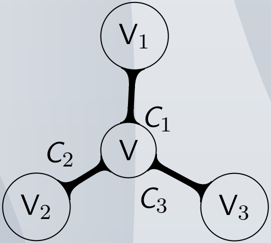
How to compute effective resistance?

How to compute effective resistance?

Kirchhoff's Law $\equiv$ flow in $=$ flow out

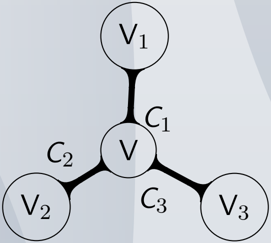
How to compute effective resistance?

Kirchhoff's Law $\equiv$ flow in $=$ flow out



How to compute effective resistance?

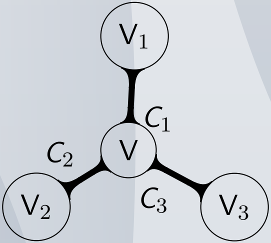
Kirchhoff's Law $\equiv$ flow in $=$ flow out



$V = \frac{C_1}{C} V_1 + \frac{C_2}{C} V_2 + \frac{C_3}{C} V_3$ (convex combination)

How to compute effective resistance?

Kirchhoff's Law $\equiv$ flow in $=$ flow out

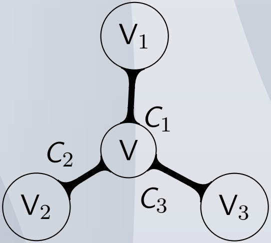


$V = \frac{C_1}{C} V_1 + \frac{C_2}{C} V_2 + \frac{C_3}{C} V_3$ (convex combination)

residual current $= CV - C_1 V_1 - C_2 V_2 - C_3 V_3$

How to compute effective resistance?

Kirchhoff's Law $\equiv$ flow in $=$ flow out



$V = \frac{C_1}{C}V_1 + \frac{C_2}{C}V_2 + \frac{C_3}{C}V_3$ (convex combination)

residual current $= CV - C_1V_1 - C_2V_2 - C_3V_3$

Kirchhoff says: This is zero! **There is no residual current!**

Resistors: Where is the link with the Laplacian?

Resistors: Where is the link with the Laplacian?

General case of the previous!

Resistors: Where is the link with the Laplacian?

General case of the previous! $d_i = \sum_j c_{ij} =$ sum of conductances

Resistors: Where is the link with the Laplacian?

General case of the previous! $d_i = \sum_j c_{ij} =$ sum of conductances

$$\mathbf{L}_{ij} = \begin{cases} d_i & \text{if } i = j, \\ -c_{ij} & \text{if } (i,j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

Resistors: Where is the link with the Laplacian?

General case of the previous! $d_i = \sum_j c_{ij}$ = sum of conductances

$$\mathbf{L}_{ij} = \begin{cases} d_i & \text{if } i = j, \\ -c_{ij} & \text{if } (i,j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

$\mathbf{v}$ = **voltage setting** of the nodes on graph.

Resistors: Where is the link with the Laplacian?

General case of the previous! $d_i = \sum_j c_{ij}$ = sum of conductances

$$\mathbf{L}_{ij} = \begin{cases} d_i & \text{if } i = j, \\ -c_{ij} & \text{if } (i,j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

$\mathbf{v}$ = **voltage setting** of the nodes on graph.

$(\mathbf{L}\mathbf{v})_i$ = residual current at $\mathbf{v}_i$ — as we derived

Resistors: Where is the link with the Laplacian?

General case of the previous! $d_i = \sum_j c_{ij}$ = sum of conductances

$$\mathbf{L}_{ij} = \begin{cases} d_i & \text{if } i = j, \\ -c_{ij} & \text{if } (i,j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

$\mathbf{v}$ = **voltage setting** of the nodes on graph.

$(\mathbf{L}\mathbf{v})_i$ = residual current at $\mathbf{v}_i$ — as we derived

Use: setting voltages and getting the current

Resistors: Where is the link with the Laplacian?

General case of the previous! $d_i = \sum_j c_{ij} =$ sum of conductances

$$\mathbf{L}_{ij} = \begin{cases} d_i & \text{if } i = j, \\ -c_{ij} & \text{if } (i,j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

$\mathbf{v} =$ **voltage setting** of the nodes on graph.

$(\mathbf{L}\mathbf{v})_i =$ residual current at $\mathbf{v}_i$ — as we derived

Use: setting voltages and getting the current

Inverting $\equiv$ injecting current and getting the voltages

Resistors: Where is the link with the Laplacian?

General case of the previous! $d_i = \sum_j c_{ij}$ = sum of conductances

$$\mathbf{L}_{ij} = \begin{cases} d_i & \text{if } i = j, \\ -c_{ij} & \text{if } (i,j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

$\mathbf{v}$ = **voltage setting** of the nodes on graph.

$(\mathbf{Lv})_i$ = residual current at $\mathbf{v}_i$ — as we derived

Use: setting voltages and getting the current

Inverting $\equiv$ injecting current and getting the voltages

The net injected has to be zero $\equiv$ Kirchhoff's Law.

Resistors and the Laplacian: Finding R_{ab}

Let's calculate R_{1N} to get the **movie recommendation score**!

$$\mathbf{L}\begin{pmatrix} 0 \\ v_2 \\ \vdots \\ v_{n-1} \\ 1 \end{pmatrix} = \begin{pmatrix} i \\ 0 \\ \vdots \\ 0 \\ -i \end{pmatrix}$$

$$i = \frac{V}{R} \qquad V = 1 \qquad R = \frac{1}{i}$$

Return $R_{1N} = \dfrac{1}{i}$

Doyle and Snell: Random Walks and Electric Networks

https://math.dartmouth.edu/~doyle/docs/walks/walks.pdf

Resistors and the Laplacian: Finding R_{1N}

$\mathbf{L}\mathbf{v} = (i, 0, \ldots, -i)^\mathsf{T} \equiv$ **boundary valued problem**

Resistors and the Laplacian: Finding R_{1N}

$\mathbf{Lv} = (i, 0, \ldots, -i)^\mathsf{T} \equiv$ **boundary valued problem**

For R_{1N}

Resistors and the Laplacian: Finding R_{1N}

$\mathbf{Lv} = (i, 0, \ldots, -i)^\mathsf{T} \equiv$ **boundary valued problem**

For R_{1N}

V_1 and V_N are the **boundary**

Resistors and the Laplacian: Finding R_{1N}

$\mathbf{Lv} = (i, 0, \ldots, -i)^\mathsf{T} \equiv$ **boundary valued problem**

For R_{1N}

V_1 and V_N are the **boundary**

$(v_1, v_2, \ldots, v_N)$ is **harmonic:**

Resistors and the Laplacian: Finding R_{1N}

$\mathbf{Lv} = (i, 0, \ldots, -i)^\mathsf{T} \equiv$ **boundary valued problem**

For R_{1N}

V_1 and V_N are the **boundary**

$(v_1, v_2, \ldots, v_N)$ is **harmonic:**

$V_i \in$ **interior** (not boundary)

Resistors and the Laplacian: Finding R_{1N}

$\mathbf{Lv} = (i, 0, \ldots, -i)^\mathsf{T} \equiv$ **boundary valued problem**

For R_{1N}

V_1 and V_N are the **boundary**

$(v_1, v_2, \ldots, v_N)$ is **harmonic:**

$V_i \in$ **interior** (not boundary)

V_i is a **convex combination of its neighbors**

Resistors and the Laplacian: Finding R_{1n}

From the properties of electric networks (cf. Doyle and Snell) we inherit the useful properties of the Laplacians!

Resistors and the Laplacian: Finding R_{1n}

From the properties of electric networks (cf. Doyle and Snell) we inherit the useful properties of the Laplacians!

Example: Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions (later in the course)

Resistors and the Laplacian: Finding R_{1n}

From the properties of electric networks (cf. Doyle and Snell) we inherit the useful properties of the Laplacians!

Example: Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions (later in the course)

Maximum Principle

If $\mathbf{f} = \mathbf{v}$ is harmonic then $\min$ and $\max$ are on the boundary.

Resistors and the Laplacian: Finding R_{1n}

From the properties of electric networks (cf. Doyle and Snell) we inherit the useful properties of the Laplacians!

Example: Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions (later in the course)

Maximum Principle

If $\mathbf{f} = \mathbf{v}$ is harmonic then $\min$ and $\max$ are on the boundary.

Proof: $k \in \circ \implies \exists$ neighbors V_i, V_j s.t. $v_i \leq v_k \leq v_j$

Resistors and the Laplacian: Finding R_{1n}

From the properties of electric networks (cf. Doyle and Snell) we inherit the useful properties of the Laplacians!

Example: Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions (later in the course)

Maximum Principle

If $\mathbf{f} = \mathbf{v}$ is harmonic then $\min$ and $\max$ are on the boundary.

Proof: $k \in \circ \implies \exists$ neighbors V_i, V_j s.t. $v_i \leq v_k \leq v_j$

Uniqueness Principle

If $\mathbf{f}$ and $\mathbf{g}$ are harmonic with the same boundary then $\mathbf{f} = \mathbf{g}$

Resistors and the Laplacian: Finding R_{1n}

From the properties of electric networks (cf. Doyle and Snell) we inherit the useful properties of the Laplacians!

Example: Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions (later in the course)

Maximum Principle

If $\mathbf{f} = \mathbf{v}$ is harmonic then $\min$ and $\max$ are on the boundary.

Proof: $k \in \circ \implies \exists$ neighbors V_i, V_j s.t. $v_i \leq v_k \leq v_j$

Uniqueness Principle

If $\mathbf{f}$ and $\mathbf{g}$ are harmonic with the same boundary then $\mathbf{f} = \mathbf{g}$

Proof: $\mathbf{f} - \mathbf{g}$ is harmonic with zero on the boundary
$\implies \mathbf{f} - \mathbf{g} \equiv 0 \implies \mathbf{f} = \mathbf{g}$ (using maximum principle)

Resistors and the Laplacian: Finding R_{1n}

From the properties of electric networks (cf. Doyle and Snell) we inherit the useful properties of the Laplacians!

Example: Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions (later in the course)

Maximum Principle

If $\mathbf{f} = \mathbf{v}$ is harmonic then $\min$ and $\max$ are on the boundary.

Proof: $k \in \circ \implies \exists$ neighbors V_i, V_j s.t. $v_i \leq v_k \leq v_j$

Uniqueness Principle

If $\mathbf{f}$ and $\mathbf{g}$ are harmonic with the same boundary then $\mathbf{f} = \mathbf{g}$

Proof: $\mathbf{f} - \mathbf{g}$ is harmonic with zero on the boundary
$\implies \mathbf{f} - \mathbf{g} \equiv 0 \implies \mathbf{f} = \mathbf{g}$ (using maximum principle)

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}}$$

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N$$

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N$$

Why?

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N \qquad \text{Why?}$$

Question: Does $\mathbf{v}$ exist?

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N$$

Why?

Question: Does $\mathbf{v}$ exist? $\mathbf{L}$ does not have an inverse :(.

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N$$

Why?

Question: Does $\mathbf{v}$ exist? $\mathbf{L}$ does not have an inverse :(.
Not unique:

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N$$

Why?

Question: Does $\mathbf{v}$ exist? $\mathbf{L}$ does not have an inverse :(.
Not unique: $\mathbf{1}$ in the nullspace of $\mathbf{L}$

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N$$

Why?

Question: Does $\mathbf{v}$ exist? $\mathbf{L}$ does not have an inverse :(.
Not unique: $\mathbf{1}$ in the nullspace of $\mathbf{L}$:
$\mathbf{L}(\mathbf{v} + c\mathbf{1}) = \mathbf{L}\mathbf{v} + c\mathbf{L}\mathbf{1} = \mathbf{L}\mathbf{v}$

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N$$

Why?

Question: Does $\mathbf{v}$ exist? $\mathbf{L}$ does not have an inverse :(.
Not unique: $\mathbf{1}$ in the nullspace of $\mathbf{L}$:
$\mathbf{L}(\mathbf{v} + c\mathbf{1}) = \mathbf{L}\mathbf{v} + c\mathbf{L}\mathbf{1} = \mathbf{L}\mathbf{v}$ **Moore-Penrose pseudo-inverse**

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N$$

Why?

Question: Does $\mathbf{v}$ exist? $\mathbf{L}$ does not have an inverse :(.
Not unique: $\mathbf{1}$ in the nullspace of $\mathbf{L}$:
$\mathbf{L}(\mathbf{v} + c\mathbf{1}) = \mathbf{L}\mathbf{v} + c\mathbf{L}\mathbf{1} = \mathbf{L}\mathbf{v}$ **Moore-Penrose pseudo-inverse** solves LS

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N$$

Why?

Question: Does $\mathbf{v}$ exist? $\mathbf{L}$ does not have an inverse :(.
Not unique: $\mathbf{1}$ in the nullspace of $\mathbf{L}$:
$\mathbf{L}(\mathbf{v} + c\mathbf{1}) = \mathbf{L}\mathbf{v} + c\mathbf{L}\mathbf{1} = \mathbf{L}\mathbf{v}$ **Moore-Penrose pseudo-inverse** solves LS
Solution: Instead of $\mathbf{v} = \mathbf{L}^{-1}\mathbf{i}_{\text{ext}}$ we take $\mathbf{v} = \mathbf{L}^{+}\mathbf{i}_{\text{ext}}$
We get: $R_{1N} = v_1 - v_N = \mathbf{i}_{\text{ext}}^{\mathsf{T}}\mathbf{v}$

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N$$

Why?

Question: Does $\mathbf{v}$ exist? $\mathbf{L}$ does not have an inverse :(.
Not unique: $\mathbf{1}$ in the nullspace of $\mathbf{L}$:
$\mathbf{L}(\mathbf{v} + c\mathbf{1}) = \mathbf{L}\mathbf{v} + c\mathbf{L}\mathbf{1} = \mathbf{L}\mathbf{v}$ **Moore-Penrose pseudo-inverse**
solves LS
Solution: Instead of $\mathbf{v} = \mathbf{L}^{-1}\mathbf{i}_{\text{ext}}$ we take $\mathbf{v} = \mathbf{L}^{+}\mathbf{i}_{\text{ext}}$
We get: $R_{1N} = v_1 - v_N = \mathbf{i}_{\text{ext}}^{\mathsf{T}}\mathbf{v} = \mathbf{i}_{\text{ext}}^{\mathsf{T}}\mathbf{L}^{+}\mathbf{i}_{\text{ext}}$.

Resistors and the Laplacian: Finding R_{1N}

Alternative method to calculate R_{1N}:

$$\mathbf{L}\mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \stackrel{\text{def}}{=} \mathbf{i}_{\text{ext}} \quad \text{Return} \quad R_{1N} = v_1 - v_N$$

Why?

Question: Does $\mathbf{v}$ exist? $\mathbf{L}$ does not have an inverse :(.

Not unique: $\mathbf{1}$ in the nullspace of $\mathbf{L}$:

$\mathbf{L}(\mathbf{v} + c\mathbf{1}) = \mathbf{L}\mathbf{v} + c\mathbf{L}\mathbf{1} = \mathbf{L}\mathbf{v}$ **Moore-Penrose pseudo-inverse**

solves LS

Solution: Instead of $\mathbf{v} = \mathbf{L}^{-1}\mathbf{i}_{\text{ext}}$ we take $\mathbf{v} = \mathbf{L}^{+}\mathbf{i}_{\text{ext}}$

We get: $R_{1N} = v_1 - v_N = \mathbf{i}_{\text{ext}}^{\mathsf{T}}\mathbf{v} = \mathbf{i}_{\text{ext}}^{\mathsf{T}}\mathbf{L}^{+}\mathbf{i}_{\text{ext}}$.

Notice: We can reuse $\mathbf{L}^{+}$ to get resistances for any pair of nodes!

What? A pseudo-inverse?

Eigendecomposition of the Laplacian:

What? A pseudo-inverse?

Eigendecomposition of the Laplacian:

$$\mathbf{L} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{\mathsf{T}} =$$

What? A pseudo-inverse?

Eigendecomposition of the Laplacian:

$$\mathbf{L} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{\mathsf{T}} = \sum_{i=1}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^{\mathsf{T}} =$$

What? A pseudo-inverse?

Eigendecomposition of the Laplacian:

$$\mathbf{L} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{\mathsf{T}} = \sum_{i=1}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^{\mathsf{T}} = \sum_{i=2}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^{\mathsf{T}}$$

Pseudo-inverse of the Laplacian:

What? A pseudo-inverse?

Eigendecomposition of the Laplacian:

$$\mathbf{L} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{\mathsf{T}} = \sum_{i=1}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^{\mathsf{T}} = \sum_{i=2}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^{\mathsf{T}}$$

Pseudo-inverse of the Laplacian:

$$\mathbf{L}^{+} =$$

What? A pseudo-inverse?

Eigendecomposition of the Laplacian:

$$\mathbf{L} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\mathsf{T} = \sum_{i=1}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^\mathsf{T} = \sum_{i=2}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^\mathsf{T}$$

Pseudo-inverse of the Laplacian:

$$\mathbf{L}^+ = \mathbf{Q}\boldsymbol{\Lambda}^+\mathbf{Q}^\mathsf{T} =$$

What? A pseudo-inverse?

Eigendecomposition of the Laplacian:

$$\mathbf{L} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\mathsf{T} = \sum_{i=1}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^\mathsf{T} = \sum_{i=2}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^\mathsf{T}$$

Pseudo-inverse of the Laplacian:

$$\mathbf{L}^+ = \mathbf{Q}\boldsymbol{\Lambda}^+\mathbf{Q}^\mathsf{T} = \sum_{i=2}^{N} \frac{1}{\lambda_i} \mathbf{q}_i \mathbf{q}_i^\mathsf{T}$$

Moore-Penrose pseudo-inverse solves a least squares problem:

What? A pseudo-inverse?

Eigendecomposition of the Laplacian:

$$\mathbf{L} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\mathsf{T} = \sum_{i=1}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^\mathsf{T} = \sum_{i=2}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^\mathsf{T}$$

Pseudo-inverse of the Laplacian:

$$\mathbf{L}^+ = \mathbf{Q}\boldsymbol{\Lambda}^+\mathbf{Q}^\mathsf{T} = \sum_{i=2}^{N} \frac{1}{\lambda_i} \mathbf{q}_i \mathbf{q}_i^\mathsf{T}$$

Moore-Penrose pseudo-inverse solves a least squares problem:

$$\mathbf{v} = \arg\min_{\mathbf{x}} \|\mathbf{L}\mathbf{x} - \mathbf{i}_{\text{ext}}\|_2 =$$

What? A pseudo-inverse?

Eigendecomposition of the Laplacian:

$$\mathbf{L} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\mathsf{T} = \sum_{i=1}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^\mathsf{T} = \sum_{i=2}^{N} \lambda_i \mathbf{q}_i \mathbf{q}_i^\mathsf{T}$$

Pseudo-inverse of the Laplacian:

$$\mathbf{L}^+ = \mathbf{Q}\boldsymbol{\Lambda}^+\mathbf{Q}^\mathsf{T} = \sum_{i=2}^{N} \frac{1}{\lambda_i} \mathbf{q}_i \mathbf{q}_i^\mathsf{T}$$

Moore-Penrose pseudo-inverse solves a least squares problem:

$$\mathbf{v} = \arg\min_{\mathbf{x}} \|\mathbf{L}\mathbf{x} - \mathbf{i}_{\text{ext}}\|_2 = \mathbf{L}^+\mathbf{i}_{\text{ext}}$$

Michal Valko

michal.valko@inria.fr

Inria & ENS Paris-Saclay, MVA

https://misovalko.github.io/mva-ml-graphs.html