Taming numerical imprecision by adapting the KL divergence to negative probabilities[a]

Simon Pfahler[1], Peter Georg[1], Rudolf Schill[2], Maren Klever[3], Lars Grasedyck[3], Rainer Spang[1], Tilo Wettig[1]

[1]University of Regensburg, [2]ETH Zürich, [3]RWTH Aachen University

August 28, 2024

Slides: **pfahler.online**

[a]Pfahler et al. in *Stat. Comput.* **34** (2024)

Introduction

Introduction - Example

- Consider the SVD of a matrix:

$$\mathrm{p} \sim \begin{pmatrix} 1 & 2 & 2 \\ 2 & 2 & 1 \\ 2 & 1 & 0.01 \end{pmatrix} \approx \begin{pmatrix} -0.62 & 0.68 & 0.40 \\ -0.66 & -0.18 & -0.73 \\ -0.42 & -0.72 & 0.56 \end{pmatrix} \begin{pmatrix} 4.51 & 0 & 0 \\ 0 & 1.63 & 0 \\ 0 & 0 & 0.14 \end{pmatrix} \begin{pmatrix} -0.62 & -0.66 & -0.42 \\ -0.68 & 0.18 & 0.72 \\ 0.40 & -0.73 & 0.56 \end{pmatrix}$$

- Truncating the smallest singular value, we get the approximation

$$\tilde{\mathrm{p}} \sim \begin{pmatrix} 0.98 & 2.04 & 1.97 \\ 2.04 & 1.93 & 1.06 \\ 1.97 & 1.06 & \text{-0.03} \end{pmatrix} \approx \begin{pmatrix} -0.62 & 0.68 \\ -0.66 & -0.18 \\ -0.42 & -0.72 \end{pmatrix} \begin{pmatrix} 4.51 & 0 \\ 0 & 1.63 \end{pmatrix} \begin{pmatrix} -0.62 & -0.66 & -0.42 \\ -0.68 & 0.18 & 0.72 \end{pmatrix}$$

- $\tilde{\mathrm{p}}$ can not be used to calculate a KL divergence, even though it is an approximation of a probability distribution

Introduction - Approach

$$D_{\mathrm{KL}}(\mathrm{p}\|\mathrm{q}) = \sum_{i=1}^{n} p_i \log \frac{p_i}{q_i}$$

- we consider the case where p is our data and q is our model
- negative entries can occur in q due to various reasons:
  1. approximations
  2. assumptions of a theory
  3. rounding errors
- if a KL-like divergence is needed, we can
  (a) give up
  (b) reformulate
  (c) devise a workaround
- this talk follows approach (c)

shifted KL divergence

sKL divergence - Idea

$$D_{\mathrm{sKL}}(\mathrm{p}\|\mathrm{q}) = \sum_{i=1}^{n} (p_i + \varepsilon_i) \log \frac{p_i + \varepsilon_i}{q_i + \varepsilon_i}$$

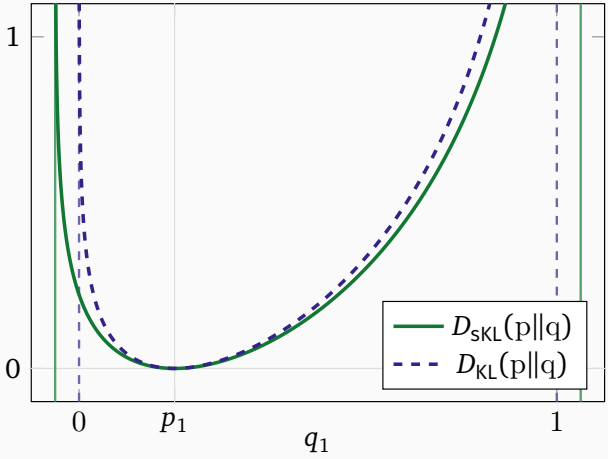
- Negative entries cause a problem because $\log x$ is only well-defined for $x > 0$
- How can we adapt the KL divergence so we can use it as an objective function in an optimization task?
- Can we just shift the negative entries? → Almost!
- In order to retain important properties of the KL divergence, p also has to be shifted

sKL divergence - Properties

$$D_{\mathrm{sKL}}(\mathrm{p}\|\mathrm{q}) = \sum_{i=1}^{n} (p_i + \varepsilon_i) \log \frac{p_i + \varepsilon_i}{q_i + \varepsilon_i}$$

The sKL divergence retains many properties of the KL divergence:

- The sKL divergence is a statistical divergence, i.e.
  (1) $D_{\mathrm{sKL}}(\mathrm{p}\|\mathrm{q}) \geq 0$
  (2) $D_{\mathrm{sKL}}(\mathrm{p}\|\mathrm{q}) = 0 \iff \mathrm{p} = \mathrm{q}$
  (3) $\frac{\mathrm{d}^2 D_{\mathrm{sKL}}(\mathrm{p}\|\mathrm{q})}{\mathrm{d}q_i \mathrm{d}q_j}|_{\mathrm{q}=\mathrm{p}}$ is a positive definite matrix
- The sKL divergence is convex in the pair of its arguments

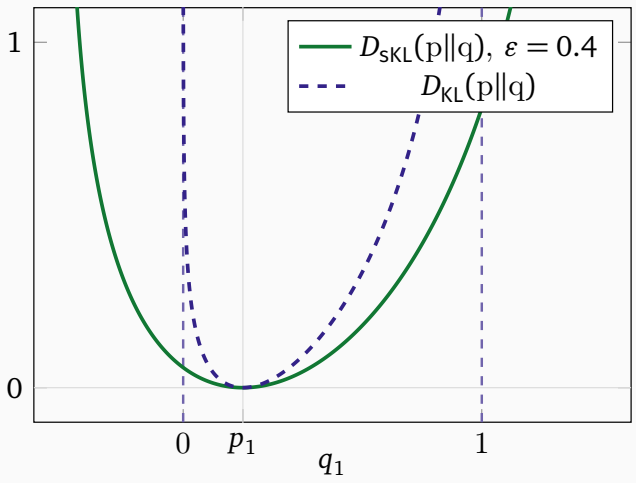


2D example with $\mathrm{p} = (0.2, 0.8)$, $\mathrm{q} = (q_1, 1 - q_1)$ and $\varepsilon = (0.05, 0.05)$

Parameter choice

Parameter choice - General observations

$D_{\mathrm{SKL}}(\mathrm{p}\|\mathrm{q}) = \sum_{i=1}^{n}(p_i+\varepsilon_i)\log\frac{p_i+\varepsilon_i}{q_i+\varepsilon_i}$



- Especially in high-dimensional data, often many entries of the data distribution p are zero → choose $\varepsilon_i = 0$ if $p_i = 0$
- The gradient with respect to q_i is

$$\frac{\partial D_{\mathrm{SKL}}(\mathrm{p}\|\mathrm{q})}{\partial q_i} = -\frac{p_i+\varepsilon_i}{q_i+\varepsilon_i} \xrightarrow{\varepsilon_i\to\infty} -1$$

→ large ε_i lead to a loss of gradient information

Simplest idea: $\varepsilon_i = \begin{cases} 0\,, & p_i = 0\,, \\ \varepsilon\,, & \text{else} \end{cases}$

⊕ Pros:

- ε does not depend on q, so the sKL divergence is the same function for any q
- Use in higher-order optimizers is possible

⊖ Cons:

- A reasonable value for ε has to be determined before the sKL divergence can be used
- Negative values $q_i < -\varepsilon$ still render the sKL divergence undefined

$$\text{Dynamic choice: } \varepsilon_i = \begin{cases} 0, & p_i = 0 \text{ or } q_i > 0, \\ |q_i| + f(|q_i|), & \text{else} \end{cases}$$

with $f : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$, e.g. $f(x) = \delta \cdot x$ with $\delta > 0$

⊕ Pros:
- No prior knowledge about q is needed
- Less gradient information is lost
- Average of the sKL divergence under Gaussian noise is controlled → next slide

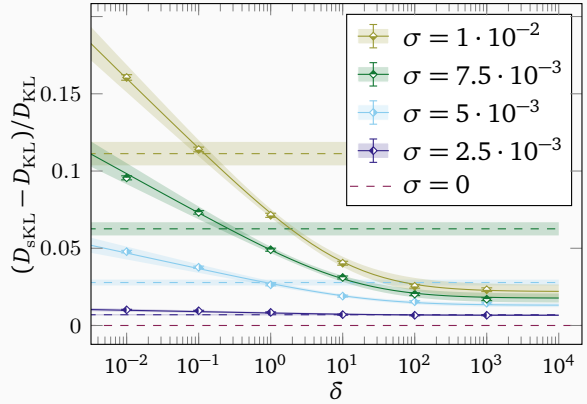
⊖ Cons:
- The sKL divergence is a different function for every q
- Only first-order optimizers can be used

Parameter choice - Gaussian noise

$D_{\mathrm{sKL}}(\mathrm{p}\|\mathrm{q}) = \sum_{i=1}^{n}(p_i+\varepsilon_i)\log\frac{p_i+\varepsilon_i}{q_i+\varepsilon_i}$

Consider a vector x of i.i.d. Gaussian random variables with mean 0 and standard deviation σ.
With the dynamic choice of ε, we get

$$\langle D_{\mathrm{sKL}}(\mathrm{p}\|\mathrm{q}+\mathrm{x})\rangle_{\mathrm{x}} = D_{\mathrm{KL}}(\mathrm{p}\|\mathrm{q}) + \sigma^2\sum_i \frac{p_i}{2q_i^2} + \mathcal{O}(\sigma^4)$$



example for $\mathrm{p},\mathrm{q}\in\mathbb{R}^{10}$

numerical result σ^2-term

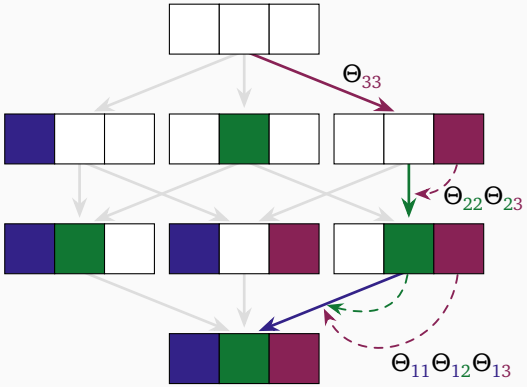
Application

Application - Mutual Hazard Networks[b]

- Modeling tumor progression as a Markov chain
- d binary events (mutations, CNAs) → 
- Tumors start healthy and accumulate events one by one with transition rate

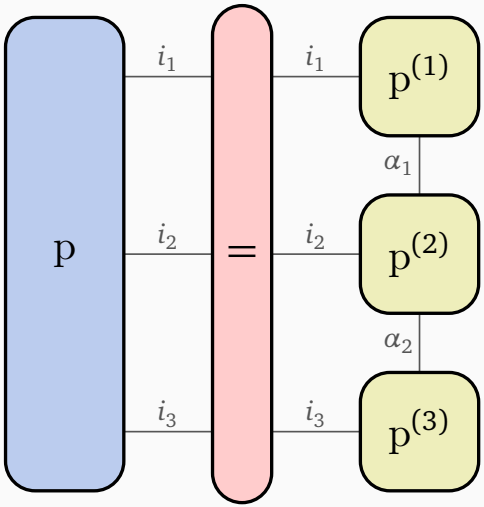
$$Q_{x^{+i},x} = \Theta_{ii} \prod_{x_j=1} \Theta_{ij}$$

- The model is parameterized by $\Theta \in \mathbb{R}^{d \times d}$, which we can learn from patient data



[b] Schill et al. in *Bioinformatics* **36** (2020)

Application - Mutual Hazard Networks



- The model parameters Θ are optimized by minimizing the KL divergence of

$$p_\Theta = (I - Q_\Theta)^{-1} p(t = 0)$$

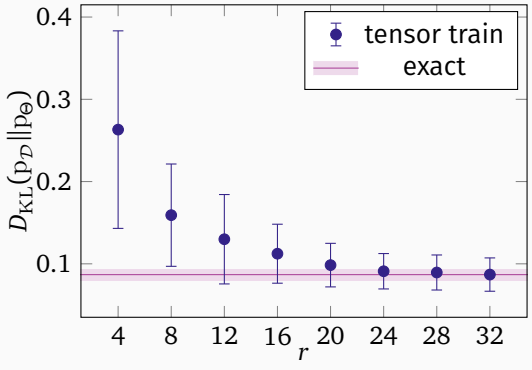
  from a given patient data distribution $p_{\mathcal{D}}$
- The size of $Q_\Theta \in \mathbb{R}^{2^d \times 2^d}$ and $p_\Theta \in \mathbb{R}^{2^d}$ grows exponentially with d → Efficient format needed!
- We use the tensor train format[c], which approximates the probability distribution using d 3D tensors

[c] Oseledets in *SIAM J. Sci. Comput.* **33** (2011)

Application - Results

- Tests on datasets with 20 events, so we can compare to the exact result
- Tensor train rank r is used to control the approximation quality (tensor trains would be exact for $r = 1024$)

$\Rightarrow$ Results quickly converge when increasing the approximation quality



Summary

Summary

- The sKL divergence provides a statistical divergence measure which
  1. shares important theoretical properties with the KL divergence
  2. is compatible with approximations of probability vectors, even if negative values occur
  3. is applicable to a wide range of problems
- Optimization tasks on high-dimensional data can be performed using the sKL divergence as an objective function
- Mutual Hazard Networks with up to ~ 40 events (compared to ~ 25 before) can be learned using tensor trains
  $\rightarrow$ for more events, we are working on another approach

Team

University of Regensburg

Tilo Wettig

Peter Georg

Marco Huber

Physics

Bioinformatics

Rainer Spang

Linda Hu

Andreas Lösch

RWTH Aachen University

Lars Grasedyck

Maren Klever

ETH Zürich

Rudolf Schill

Kevin Rupp