Information Theoretic Clustering via Divergence Maximization among Clusters (Supplementary Material)

Sahil Garg[1,*] **Mina Dalirrooyfard**[1] **Anderson Schneider**[1] **Yeshaya Adler**[1] **Yuriy Nevmyvaka**[1] **Yu Chen**[1]
Fengpei Li[1] **Guillermo Cecchi**[2]

[1]Dept. of Machine Learning Research, Morgan Stanley, New York, New York, USA
[2]IBM T. J. Watson Research Center, Yorktown Heights, New York, USA,
[*] Corresponding Author: sahil.garg@morganstanley.com, sahil.garg.cs@gmail.com

1 PROOFS

1.1

Proof of Theorem 1. First we compute $D(p||q)$ and $D(q||p)$ in terms of entropy.

$$\mathcal{D}(p||q) = -\mathcal{H}(p) - \langle \log q(.) \rangle_p \tag{1}$$
$$= -\mathcal{H}(p) + \mathcal{H}_p(q) \tag{2}$$

And similarly we have

$$\mathcal{D}(q||p) = -\mathcal{H}(q) - \langle \log p(.) \rangle_q \tag{3}$$
$$= -\mathcal{H}(q) + \mathcal{H}_q(p) \tag{4}$$

So with $\lambda = \frac{p(y=q)}{p(y=p)} = \frac{n_q}{n_p}$ we have

$$\text{argmax}_{p,q:\mathcal{H}(\boldsymbol{\mathcal{X}})=c} \ \mathcal{D}(p||q) + \lambda \mathcal{D}(q||p) \tag{5}$$
$$= \text{argmax}_{p,q:\mathcal{H}(\boldsymbol{\mathcal{X}})=c} \ -\mathcal{H}(p) - \lambda \mathcal{H}(q) + \lambda \mathcal{H}_q(p) + \mathcal{H}_p(q) \tag{6}$$
$$= \text{argmax}_{p,q:\mathcal{H}(\boldsymbol{\mathcal{X}})=c} \ -\mathcal{H}(\boldsymbol{\mathcal{X}}|y=p) - \lambda \mathcal{H}(\boldsymbol{\mathcal{X}}|y=q) + \lambda \mathcal{H}_q(p) + \mathcal{H}_p(q) \tag{7}$$
$$= \text{argmax}_{p,q:\mathcal{H}(\boldsymbol{\mathcal{X}})=c} \ -p(y=p)\mathcal{H}(\boldsymbol{\mathcal{X}}|y=p) - p(y=q)\mathcal{H}(\boldsymbol{\mathcal{X}}|y=q) + p(y=q)\mathcal{H}_q(p) + p(y=p)\mathcal{H}_p(q) \tag{8}$$
$$= \text{argmax}_{p,q:\mathcal{H}(\boldsymbol{\mathcal{X}})=c} \ \mathcal{H}(\boldsymbol{\mathcal{X}}) - p(y=p)\mathcal{H}(\boldsymbol{\mathcal{X}}|y=p) - p(y=q)\mathcal{H}(\boldsymbol{\mathcal{X}}|y=q) + p(y=q)\mathcal{H}_q(p) + p(y=p)\mathcal{H}_p(q) \tag{9}$$
$$= \text{argmax}_{p,q:\mathcal{H}(\boldsymbol{\mathcal{X}})=c} \ \mathcal{I}(\boldsymbol{\mathcal{X}}:\mathcal{Y}) + \frac{n_q}{n}\mathcal{H}_q(p) + \frac{n_p}{n}\mathcal{H}_p(q) \tag{10}$$

□

1.2

Proof of Theorem 2. The problem can be seen as, given a subset $S \subseteq S_{\mathcal{X}}$, the input space (i.e., for x in a high dimensional space), we want to solve the optimization problem

$$\text{Opt1}(S) := \max_{\substack{S_1 \cup S_2 = S \\ S_1 \cap S_2 = \emptyset}} \left[\max_{\substack{f \in C^1(S_{\mathcal{X}}) \\ a \le f \le b}} \left[\frac{\sum_{x \in S_1} f(x)}{|S_1|} - \log \frac{\sum_{x \in S_2} e^{f(x)}}{|S_2|} \right] + \max_{\substack{g \in C^1(S_{\mathcal{X}}) \\ a \le g \le b}} \left[\frac{\sum_{x \in S_2} g(x)}{|S_2|} - \log \frac{\sum_{x \in S_1} e^{g(x)}}{|S_1|} \right] \right].$$

Accepted for the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023).

For any given f, by Jesen's inequality, we have

$$\frac{\sum_{x\in S_2} e^{f(x)}}{|S_2|} \geq \exp\big(\frac{\sum_{x\in S_2} f(x)}{|S_2|}\big), \tag{11}$$

so we have

$$\begin{aligned}\max_{\substack{S_1\cup S_2=S\\ S_1\cap S_2=\emptyset}} \max_{\substack{f\in C^1(S_{\mathcal{X}})\\ a\leq f\leq b}} \left[\frac{\sum_{x\in S_1} f(x)}{|S_1|} - \log\frac{\sum_{x\in S_2} e^{f(x)}}{|S_2|}\right] \leq & \max_{\substack{S_1\cup S_2=S\\ S_1\cap S_2=\emptyset}} \max_{\substack{f\in C^1(\mathcal{X})\\ a\leq f\leq b}} \left[\frac{\sum_{x\in S_1} f(x)}{|S_1|} - \frac{\sum_{x\in S_2} f(x)}{|S_2|}\right]\\ \leq & b-a. \end{aligned} \tag{12}$$

Thus, an upper bound for Opt1(S) is simply $2(b-a)$. However, by letting

$$f = \begin{cases} b, \text{ for } x\in S_1\\ a, \text{ for } x\in S_2\end{cases} \quad \text{and} \quad g = \begin{cases} a, \text{ for } x\in S_1\\ b, \text{ for } x\in S_2\end{cases}, \tag{13}$$

we would achieve the optimal value $2(b-a)$, since (11) is tight for such f, g. Thus, the optimal value of Opt1(S) is exactly $2(b-a)$.

Now, if there exists two clusters S_1, S_2 with

$$\begin{aligned} d_{\text{between}} &:= \min_{x_1\in S_1, x_2\in S_2} \|x_1 - x_2\|\\ d_i &:= \max_{x\in S_i, x'\in S_i} \|x - x'\|\\ d_{\text{within}} &:= \max(d_1, d_2)\end{aligned}$$

that satisfies $d_{\text{between}} > d_{\text{within}}$. Then for L that satisfies

$$\frac{b-a}{d_{\text{within}}} < L < \frac{b-a}{d_{\text{between}}},$$

we claim (13) is the unique optimal solution to

$$\text{Opt4}(S) := \max_{\substack{S_1\cup S_2=S\\ S_1\cap S_2=\emptyset}} \Bigg[\max_{\substack{f\in C^1(\mathcal{X})\\ a\leq f\leq b\\ f \text{ is } L\text{-Lipschitz}}} \left[\frac{\sum_{x\in S_1} f(x)}{|S_1|} - \log\frac{\sum_{x\in S_2} e^{f(x)}}{|S_2|}\right] + \max_{\substack{g\in C^1(\mathcal{X})\\ a\leq g\leq b\\ g \text{ is } L\text{-Lipschitz}}} \left[\frac{\sum_{x\in S_2} g(x)}{|S_2|} - \log\frac{\sum_{x\in S_1} e^{g(x)}}{|S_1|}\right].$$

To see this, first note that solution (13) is feasible because $L < \frac{(b-a)}{d_{\text{between}}}$. This solution is optimal because it achieves value $2(b-a)$ which is the upper bound (best possible) of the objective value. This solution is unique because in order to achieve objective value $b-a$, the inequality in (13) and (12) must be tight, which means one set of points must take value b and the other a. One can check if any $x\in S_1$ have $f(x)\neq a$ then either $f(x)\neq b$ which is sub-optimal or $f(x)=b$ which is infeasible because $\frac{b-a}{d_{\text{within}}}$. Thus we must have $f(x)=a$ for all $x\in S_1$. Similarly $f(x)=b$ for all $x\in S_2$. Thus the optimal solution is unique. □

Reminder of Theorem 3 *Consider a DV function f and the associated DV representation of the data. Then the clusters that maximize $\hat{D}_f(P||Q)$ form contiguous clusters on the DV representation of the data points: there is a cut point c such that $i\in P$ if $f(\mathbf{x}_i)\geq c$ and $i\in Q$ if $f(\mathbf{x}_i)<c$.*

Proof of Theorem 3. Recall that:

$$\hat{D}_f(P||Q) = \frac{1}{n_P}\sum_{z\in P} f(z) - \log\frac{1}{n_Q}\sum_{z'\in Q}\exp f(z'). \tag{14}$$

Consider the clusters P and Q that maximize Equation 14 (possibly different from the initial clusters that are used to define the DV function f). Suppose that there are data points $z\in P$ and $z'\in Q$ such that $f(z)<f(z')$. We show that by swapping z and z' from their clusters, Equation 14 increases, which is a contradiction (see figure 1). Note that the size of the clusters

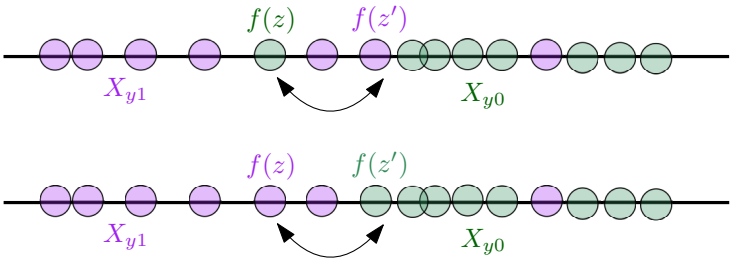

Figure 1: Swapping the cluster labels of two points z and z' to increase KL-D.

does not change. Since $f(z) < f(z')$, after swapping z and z' the first term in Equation 14 increases and the second term (which is negated) decreases. So there are no such z and z', which means that the clusters must be contiguous. Note that if we consider the approximation of (14), namely $\hat{D}_f(P||Q) \approx \frac{1}{n_P}\sum_{z\in P} f(z) - \max_{z'\in Q} f(z') + \log n_Q$, the same proof works.

□

1.3

We first define a total ordering on clusters when the DV representation f is fixed. For any P and Q and fixed DV function f, we say $P > Q$ if the maximum f value of the data points in Q is less than the maximum f value of the data points in P. More formally, $P > Q$ if

$$\max_{x\in P} f(x) > \max_{x\in Q} f(x) \tag{15}$$

Reminder of Theorem 4 *Consider a DV function f and the associated DV representation of the data. Let $P_1^* < \ldots < P_k^*$ be the clusters that maximize the objective*

$$\max_{P_1<\ldots<P_k} \sum_{i=1}^{k-1} \hat{D}_f(P_{>i}||P_i) \tag{16}$$

where $P_{>i} = P_k \cup \ldots \cup P_{i+1}$. Then $P_1^*, \ldots, P_k^*$ form contiguous clusters on the DV representation of the data points: there exist cut points $c_1 < \ldots < c_{k-1}$ such that for all $i < k$ and for all $j \in P_i$ we have $f(\mathbf{x}_j) < c_i$, and for all $i > 1$ and for all $j \in P_i$ we have $f(\mathbf{x}_j) \geq c_i$.

Proof of Theorem 4. Suppose that there are indices t, j, $t < j$ such that for data points x, y with $x \in P_t$ and $y \in P_j$ we have $f(x) > f(y)$. We show that swapping x and y in their clusters increases the function maximized by Equation as follows.

$$\max_{P_1<\ldots<P_k} \sum_{i=1}^{k-1} \hat{D}_f(P_{>i}||P_i)$$

We determine which terms in the Equation above changes with swapping x and y in P_j and P_t. We consider the following for estimating $\hat{D}_f(P||Q)$.

$$\hat{D}_f(P||Q) \approx \frac{1}{n_P}\sum_{x\in P} f(x) - \max_{x\in Q} f(x) + \log n_Q. \tag{17}$$

Note that the size of the clusters stays the same by swapping x and y.

For all $i < t$ and all $i > j$, $\hat{D}_f(P_{>i}||P_i)$ does not change. For $i = t$, $\hat{D}_f(P_{>t}||P_t)$ increases since the first term in Equation 17 increases and the second term (which is negated) either stays the same or decreases. For $t < i < j$, $\hat{D}_f(P_{>i}||P_i)$ increases since the first term increases and the second term doesn't change. For $i = j$, $\hat{D}_f(P_{>j}||P_j)$ stays the same since none of the terms change, because y cannot be the maximum value in P_j by the ordering properties. □

1.4

Reminder of Theorem 5 *Let OPT be the optimal value of the objective $\max_{P_1<\ldots<P_k} \sum_{i=1}^{k-1} \hat{D}_f(P_{>i}||P_i)$. Alg. 3 finds clusters $P_1^* < \ldots < P_k^*$ such that $\sum_{i=1}^{k-1} \hat{D}_f(P_{>i}^*||P_i^*) \geq \frac{e-1}{e} OPT$, where the ordering of clusters is defined with respect*

to the DV representation obtained from a DV function f.

Proof of Theorem 5. Let f be the DV function that defines the fixed DV representation in the theorem. We first rewrite the objective $\max_{P_1<\ldots<P_k} \sum_{i=1}^{k-1} \hat{D}_f(P_{>i}||P_i)$ in terms of cut points that separate contiguous clusters in the DV representation.

Let S be a set of real numbers where for any two values $s_1, s_2 \in S$, $s_1 < s_2$, there is at least one data point x such that $s_1 < f(x) < s_2$, and for any two data points x_1, x_2, $f(x_1) < f(x_2)$, there is a value $s \in S$ such that $f(x_1) < s < f(x_2)$. Note that S can be the set of the value of all the data points in the DV representation minus a very small number. These conditions mean that we cannot have empty clusters, and that any contiguous cluster can be represented uniquely by the interval between two values in S without worrying about including or excluding the endpoints of the interval. Recall that in Alg. 3, we use the DV representation values of the data points as cut points, and we could enforce the non-empty cluster condition by using the set S.

Let c_{min} and c_{max} be the minimum and maximum value in S. Let $C = \{c_1, \ldots, c_{t-1}\} \subseteq S$ be a set of cut points. C defines t clusters, where cluster i is all the data points with DV representation between c_{i-1} and c_i, where we define $c_0 = c_{min}$ and $c_t = c_{max}$.

We rewrite the objective in terms of the cut set C. For any two cut points $c, c' \in S$, $c < c'$, let $n_{c,c'}$ be the number of data points whose DV representation is in (c, c'), and let $\overline{f((c,c'))}$ be the mean of the values of the data points in (c, c'), i.e. $\overline{f((c,c'))} = \frac{1}{n_{c,c'}} \sum_{x:f(x)\in(c,c')} f(x)$. Define $\max f(c,c') = \max_{x:f(x)\in(c,c')} f(x)$. Let $C = \{c_1 < \ldots < c_{t-1}\}$. We define the function $z(\cdot)$ as follows.

$$z(C) = \sum_{i=1}^{t-1} \overline{f(c_i, c_t)} - \sum_{i=0}^{t-1} \max f(c_i, c_{i+1}) + \sum_{i=0}^{t-1} \log n_{c_i,c_{i+1}} \tag{18}$$

Then the objective is to maximize $z(C)$ over cut sets C of size $k-1$. Note that this objective is equivalent to Equation 1.3 because by Theorem 4 we know that the optimal clusters are contiguous.

To prove that the greedy algorithm gives a $\frac{e-1}{e}$ approximation of the optimal solution, we use a result of Nemhauser et al. [1978] stating that if a set function is submodular, then the generic greedy algorithm is a $\frac{e-1}{e}$ approximation of the optimal.

To show the submodularity of function z, we need to prove that for any two sets A and B where $A \subseteq B$ and for any cut point $c \in S$, $c \notin B$, we have $z(A \cup \{c\}) - z(A) \geq z(B \cup \{c\}) - z(B)$.

First we compute $z(A \cup \{c\}) - z(A)$. Sort the points in $A \cup \{c\}$, and let $c_1 < c < c_2$ be the points before and after c in this ordering. Note that c_1 might be c_{min} and c_2 might be c_{max}. From equation 18, we see that

$$\begin{aligned} z(A \cup \{c\}) - z(A) = &\overline{f(c, c_{max})} + \max f(c_1, c_2) - \log n_{c_1,c_2} \\ &- \max f(c, c_2) + \log n_{c,c_2} \\ &- \max f(c_1, c) + \log n_{c_1,c}. \end{aligned}$$

Let $c \notin B$ and let $c'_1 \leq c \leq c'_2$ be the cut points before and after c in B's ordering. Similar to above, we have

$$\begin{aligned} z(B \cup \{c\}) - z(B) = &\overline{f(c, c_{max})} + \max f(c'_1, c'_2) - \log n_{c'_1,c'_2} \\ &- \max f(c, c'_2) + \log n_{c,c'_2} \\ &- \max f(c'_1, c) + \log n_{c'_1,c}. \end{aligned}$$

Note that since $A \subseteq B$, we have $c_1 \leq c'_1 \leq c \leq c'_2 \leq c_2$. Now from the definition of S, we have that $n_{c'_1,c} > 0$ and $n_{c,c'_2} > 0$. So $\max f(c_1, c) = \max f(c'_1, c)$, $\max f(c_1, c_2) = \max f(c, c_2)$ and $\max f(c'_1, c'_2) = \max f(c, c'_2)$. So we have

$$\begin{aligned} z(A \cup \{c\}) - z(A) - (z(B \cup \{c\}) - z(B)) &= -\log n_{c_1,c_2} + \log n_{c_1,c} + \log n_{c,c_2} \\ &\quad + \log n_{c'_1,c'_2} - \log n_{c'_1,c} - \log n_{c,c'_2} \\ &= \log \frac{n_{c'_1,c'_2}}{n_{c'_1,c} n_{c,c'_2}} - \log \frac{n_{c_1,c_2}}{n_{c_1,c} n_{c,c_2}} \\ &= \log(\frac{1}{n_{c'_1,c}} + \frac{1}{n_{c,c'_2}}) - \log(\frac{1}{n_{c_1,c}} + \frac{1}{n_{c,c_2}}) \geq 0 \end{aligned}$$

Note that we used the fact that $n_{c_1,c} + n_{c,c_2} = n_{c_1,c_2}$, $n_{c'_1,c} + n_{c,c'_2} = n_{c'_1,c'_2}$. Moreover, since $c_1 \geq c'_1$, we have $n_{c_1,c} \geq n_{c'_1,c}$. Similarly, we have $n_{c,c_2} \geq n_{c,c'_2}$, and hence we have the last inequality. □

2 EXPERIMENTAL DETAILS

2.1 DATASETS OF NOISY TIMESERIES

Details for some of the datasets which are of high impact but rare to find are as belows.

Neural Activity We used public electrophysiological Neuropixels dataset [Siegle et al., 2021, Institute, 2020]. Multiple high-density extracellular electrophysiology probes were used to simultaneously record spiking neural activity from a wide variety of areas in the mouse brain. We used the data of the animal with session-id `798911424` and included the first 100 out of 195 trials. The first 2000 ms of each trial after stimulus onset was extracted. We time-binned the timestamps with 0.1 ms resolution, giving 443 timeseries, each of length 20,000 timesteps.

Financial time series of returns of US stocks. We started with the 1000 stocks from the constituents of the Russell 3000 index that have the highest liquidity. This dataset is publicly available, though very large in size to be released as a single file. After performing necessary preprocessing and checks on data quality issues, we use 982 of those stocks. The returns are evaluated every 15 minutes, for the period of from May 2021 to May 2022, i.e. 2600 timesteps.

Wind Dataset This dataset is daily average wind speed (in knots = 0.5418 m/s) data collected from year 1961 to 1978 at 12 meteorological stations in the Republic of Ireland (Gneiting 2002).[1]

Rain Dataset Daily data collected for rainy days in 1949–94 across 167 regions in Washington and Oregon states.

Other datasets are available at Kaggle. We also generated a synthetic dataset, with binary timeseries. Script for generating the data will be provided in the code base. All the data files will be provided as part of the codebase.

References

Allen Institute. *Visual Coding - Neuropixels*, 2020.

George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 1978.

Joshua H Siegle, Xiaoxuan Jia, Séverine Durand, Sam Gale, Corbett Bennett, Nile Graddis, Greggory Heller, Tamina K Ramirez, Hannah Choi, Jennifer A Luviano, et al. Survey of spiking in the mouse visual system reveals functional hierarchy. *Nature*, 2021.

[1] `http://lib.stat.cmu.edu/datasets/wind.desc`