Peer-Review of "Entropic Neural Optimal Transport"

Eduard Tulchinskiy

May 21, 2022

1 Introduction

This is a peer-review of second report for project “Entropic Neural Optimal Transport” prepared by Andrew Spiridonov and Polina Karpikova. The reviewed document and source code were obtained from Github repository of that project https://github.com/poliik/ENOT.git.

2 Problem statement and main idea

Authors of the reviewed report aim they work at adding entropy regularization to Neural Optimal Transport (NOT) algorithm in order to improve convergence and achieve better quality of the obtained results. They provide mathematical background necessary for understanding NOT and proposed idea and it is easy to understand, but it also would be great to have some more native explanations and examples.

At the same time, report is quite too much approach-centered. Idea is proposed very clearly, however the problem - why does entropy regularization needed for NOT - is stated vague and briefly. In my opinion, it would be better to extended explanations and make them more accessible to readers who are not yet accustomed to Optimal Transport problems (in particular explain the “mode collapse” that stated as one of the main hazards of the traditional approach).

3 Literature Review

Literature review briefly covers modern approaches to calculating OT plans. Comparison with relevant method is also present. These too parts are very brief and, in my opinion, should be extended if the novelty of the field allows it (were there any other attempts to regularize NOT algorithm? were there any more algorithms to compute OT plans?)

4 Experiments

Report includes results of some initial experiments with proposed solutions. Stochastic maps and loss plots are very illustrative, but it would be great to have any numeric estimations of quality too.

Also, would like to see comparison with the traditional NOT-algorithm and other relevant approaches, but since project is still ongoing it is not a critical issue.

Some parts of experimental protocol is still unclear — what was exact structure of MLP (number of layers) in each experiment? what optimizer was used and with what hyperparameters?

5 General notes

- Report is well structured, its styling and quality are pretty decent.
- Link to project repository at GitHub is present, but i would recommend to insert full URL-address in the report (maybe in the footnotes). In its current state (hyperlink), repository address will be lost for reader if they try to print the paper.
- Readme file contains the structure of the repository and instructions to work with it and run the code.
- I would recommend to add some more commentaries to the IPython notebook with code for experiments.