Reading Assignment

Last Update: February 2, 2018

Objective:

The objective of the reading assignment is have you learn a few popular algorithms. To achieve this objective, I will ask you to read the papers and write me a summary report. The due dates of the paper summaries will be posted on the website.

Report Format:

Here is a rough guideline for the report format.

- All reports must be typed in LaTeX. Print out the hard copy and hand in during the class.
- Please use LaTeX template of the Project Report when typing your reading assignment.
- Maximum pages: 2 pages. This includes everything. 10 point font please.

What to Include:

There are two pages of the report. While I do not have strict rules on the layout, generally I like to see these on the first page:

1. Problem Statement. Every paper has a problem. You need to understand what they are trying to solve. If it is a tutorial / textbook chapter, describe the theme.
2. Prior Work. Briefly mention what has been done before their work.
3. Contribution. Have a few bullet points summarizing the contributions of the paper. For tutorial / textbook, provide a short summary of what they cover.
4. Key Idea. Describe the key ideas behind the paper. Have one or two most important equations. Do not re-derive the results. If there are convergence results etc, briefly describe the conditions. If there are assumptions, mention their assumption.

For the remaining, please tell me:

1. Your comment. That can be pros and cons of the algorithm.
2. Toy experiments. Most of the papers have codes associated. Feel free to download their code and try. Run a few experiments and report your findings.
3. References, if any.

Tips: My grading criteria depends on how much effort you spend on the assignment. At the very minimal level, you need to demonstrate that you actually understand the paper. So please do not copy word by word from the paper because I am quite good in detecting those. If there is something you do not fully understand, you should honestly tell me. A good way to demonstrate you really understand the paper is to try out their codes. As for programming language, I am open to any language as long as you feel comfortable, that includes R, Python, C++, MATLAB, etc.

© 2018 Stanley Chan. All Rights Reserved.

Reading List

Reading 1 L1LS

S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An Interior-Point Method for Large-Scale L1-Regularized Least Squares," *IEEE Journal of Selected Topics in Signal Processing: Special Issue on Convex Optimization Methods for Signal Processing*, vol. 1, no. 4, pp. 606–617, 2007.

Code: `https://stanford.edu/~boyd/l1_ls/`

Reading 2 Gradient Projection

M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing: Special Issue on Convex Optimization Methods for Signal Processing*, vol. 1, no. 4, pp. 586–598, 2007.

Code: `http://www.lx.it.pt/~mtf/GPSR/`

Reading 3: Basis Pursuit

E. van den Berg and M. P. Friedlander, "Probing the Pareto Frontier for Basis Pursuit Solutions," SIAM Journal on Scientific Computing, vol. 31, no. 2, pp. 890-912, Nov 2008.

Code: `http://www.cs.ubc.ca/~mpf/spgl1/`

Reading 4 Iterative Shrinkage Thresholding

A. Beck and M. Teboulle, "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems," SIAM J. Imaging Sci., vol. 2, no. 1, pp. 183–202, Mar 2009.

Code: `https://people.eecs.berkeley.edu/~yang/software/l1benchmark/`

Reading 5 The FOCUSS Algorithm (Iterated Reweighted Least-Squares)

I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: A re-weighted norm minimization algorithm," *IEEE Trans. Signal Processing*, 45, pp. 600–616, March 1997.

Code: `http://dsp.ucsd.edu/~jfmurray/software.htm`

© 2018 Stanley Chan. All Rights Reserved.

Reading 6: Greedy Algorithms

Greedy algorithms discussed in Elad Chapter 3.1:

- Orthogonal Matching Pursuit
- Matching Pursuit
- Weak Matching Pursuit
- Thresholding

Code: These algorithms are fairly easy to implement by yourselves.
For this assignment you do not need to write the "prior-art" section. Just focus on the key idea section and the comment section. The overview section can be a as few as two or three sentence, like an abstract. Tell me more about your findings besides runtime.

Reading 7 Convolutional Sparse Coding

B. Wohlberg, "Efficient Algorithms for Convolutional Sparse Representations", IEEE Transactions on Image Processing, vol. 25, no. 1, pp. 301-315, Jan 2016

Code: `http://brendt.wohlberg.net/software/SPORCO/`

© 2018 Stanley Chan. All Rights Reserved.