Diploma thesis

TOWARDS GENE EXPRESSION PREDICTION IN MOUSE BRAINS

September 30, 2021

Tilman Hinnerichs
Matrikelnummer: 4643427
Technische Universität Dresden

Tutor: Dr. Nico Scherf
MPI for CBS

Summer semester 2021

proper title?

Abstract

Contents

To include in some chapter

Predict gene expression per section/structure:
Take region as input and predict gene expression
Challenges:

- how to normalize expression intensity (see discussion in DeepMOCCA paper, Sara Alghamdi), as there are regions with much more activity than others (e.g. bone narrow vs. bone boarder); thresholds for intensity varies across genes
  - over all intensities →
  - per structure →
  - per gene →

  Our model also allows us to test different ways of representing omics data. We tested different ways to normalize values assigned to genes as these normalizations convey different biological information; in the matrix of values assigned to genes from cancer samples, we can normalize values across the entire matrix, across each row (cancer sample), or across each column (gene). While a global normalization is more common, row-based normalization allows us to highlight values that are significantly higher or lower within one sample (e.g., which genes are expressed at high or low levels within a single sample), and column-based normalization allows us to highlight values assigned to a particular gene that are significantly higher or lower within one sample (e.g., whether a gene is expressed at higher or lower levels within one sample compared to all others). We find that column-based normalization performs better than row-based normalization, while the global normalization approach performs close to random. The best results are achieved when combining both row- and column-based normalization (Supplementary Table 2).
- transfer learning working for other structure/regions
- dataset: Allen Mouse brain atlas vs.
  - phenoview impc data
  - mousephenotype
  - HPO/MP project expression data
- structure specific features?
  - structural ontology / closeness
  - developmental hierarchy of tissue

Possible hypotheses

- predict gene expression for a given single structure
- predict structure from gene expression pattern
- predict structure form gene expression and image
- predict cancer type from morphology/pathologic image of cancer
- simulate loss of function/expression by removing one node of graph

Ideas for page-filling plots

- show distribution (histo, mean, median, boxplot?) of expression densities see 'get_ge_structure_mat'

1 Introduction

- Works on mouse brain in general and potential tasks
- works on gene expression in mouse brains
- conservative approaches
- neural networks for this purpose
- gene expression for general tissue

2 Literature review

- Variability and different interpretations of different graph convolutional neural filters [Kipf and Welling, 2016, Li et al., 2020, Hamilton et al., 2017] etc.
- Guilt by association over gene networks [Oliver, 2000, Gillis and Pavlidis, 2012]
- protein function prediction from PPI networks [Vazquez et al., 2003]
- DeepGOPlus for feature generation [Kulmanov and Hoehndorf, 2019]
- discussion of DeepMocca by Sara [Althubaiti et al., 2021]
- discussion of different PPI network databases [Szklarczyk et al., 2014]
- discussion of best neural learning/graph convolutional methods [Paszke et al., 2019, Fey and Lenssen, 2019]
- how to handle highly imbalanced data, metrics, preprocessing, sampling, modification of loss function [Jeni et al., 2013] and optimization over them (with Adam[Kingma and Ba, 2015])
- maybe introduction of PhenomeNET for MP/GO for more sophisticated protein representation [Hoehndorf et al., 2011, Ashburner et al., 2000, Carbon et al., 2020, Smith and Eppig, 2009] and derive features from DL2vec [Chen et al., 2020, Mikolov et al., 2013]
- evaluation of „Using ontology embeddings for structural inductive bias in gene expression data analysis"[Trebacz et al., 2020]
- take some ideas from Zitnik and Leskovec [2017] with title „Predicting multicellular function through multi-layer tissue networks". (OhmNet)
- potentially group results based on InterPro[Blum et al., 2020] families eventually
- RayTune[Liaw et al., 2018] for automated hyperparameter tuning

3 Methods

3.1 Problem description

3.2 Datasets

- Allen mouse brain atlas [Lein et al., 2006]
- STRING for PPI network and how we chose suitable interactions [Szklarczyk et al., 2014]

3.3 Model

3.3.1 Feature generation

3.3.2 Graph convolutional neural layers

We include these molecular and ontology-based sub-models within a graph neural network (GNN) [Kipf and Welling, 2016]. The graph underlying the GNN is based on the protein–protein interaction (PPI) graph. The PPI dataset is represented by a graph $G = (V, E)$, where each protein is represented by a vertex $v \in V$, and each edge $e \in E \subseteq V \times V$ represents an interaction between two proteins. Additionally, we introduce a mapping $x : V \to \mathbb{R}^d$ projecting each vertex v to its node feature $x_v := x(v)$, where d denotes the dimensionality of the node features.

A graph convolutional layer [Kipf and Welling, 2016] consists of a learnable weight matrix followed by an aggregation step, formalized by

$$\mathbf{X}' = \hat{\mathbf{D}}^{-1/2}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-1/2}\mathbf{X}\mathbf{\Theta} \tag{1}$$

where for a given graph $G = (V, E)$, $\hat{A} = A + I$ denotes the adjacency matrix with added self-loops for each vertex, D is described by $\hat{D}_{ii} = \sum_{j=0} \hat{A}_{ij}$, a diagonal matrix displaying the degree of each node, and Θ denotes the learnable weight matrix. Added self-loops enforce that each node representation is directly dependent on its own preceding one. The number of graph convolutional layers stacked equals the radius of relevant nodes for each vertex within the graph.

The update rule for each node is given by a message passing scheme formalized by

$$\mathbf{x}'_i = \mathbf{\Theta} \sum_{j}^{N} \frac{1}{\sqrt{\hat{d}_j \hat{d}_i}} \mathbf{x}_j \tag{2}$$

where both $\hat{d}_i, \hat{d}_j$ are dependent on the edge weights e_{ij} of the graph. With simple, single-valued edge weights such as $e_{ij} = 1\ \forall (i, j) \in E$, all $\hat{d}_i$ reduce to d_i, i.e., the degree of each vertex i. We denote this type of graph convolutional neural layers with GCNConv.

While in this initial formulation of a GCNConv the node-wise update step is defined by the sum over all neighboring node representations, we can alter this formulation to other message passing schemes. We can rearrange the order of activation function σ, aggregation AGG, and linear neural layer MLP with this formulation as proposed by [Li et al., 2020]:

$$\mathbf{x}'_i = \mathrm{MLP}\left(\mathbf{x}_i + \mathrm{AGG}\left(\{\sigma\left(\mathbf{x}_j + \mathbf{e_{ji}}\right) + \epsilon : j \in \mathcal{N}(i)\}\right)\right) \tag{3}$$

where we only consider $\sigma \in \{\mathrm{ReLU}, \mathrm{LeakyReLU}\}$. We denote this generalized layer type as GENConv following the notation of PyTorch Geometric [Fey and Lenssen, 2019]. While the reordering is mainly important for numerical stability, this alteration also addresses the vanishing gradient problem for deeper convolutional networks [Li et al., 2020]. Additionally, we can also generalize the aggregation function to allow different weighting functions such as learnable SoftMax or Power for the incoming signals for each vertex, substituting the averaging step in GCNConv. Hence, while GCNConv suffers from both vanishing gradients and signal fading for large scale and highly connected graphs, each propagation step in GENConv emphasizes signals with values close to 0 and 1. The same convolutional filter and weight matrix are applied to and learned for all nodes simultaneously. We further employ another mechanism to avoid redundancy and fading signals in stacked graph convolutional networks, using residual connections and a normalization scheme [Li et al., 2019] [Li et al., 2020] as shown in Supplementary 3. The residual blocks are reusable and can be stacked multiple times.

3.3.3 Combined prediction model

3.3.4 Hyperparameter tuning

References

S. Althubaiti, M. Kulmanov, Y. Liu, G. V. Gkoutos, P. Schofield, and R. Hoehndorf. Deepmocca: A pan-cancer prognostic model identifies personalized prognostic markers through graph attention and multi-omics data integration. *bioRxiv*, 2021. doi: 10.1101/2021.03.02.433454. URL `https://www.biorxiv.org/content/early/2021/03/02/2021.03.02.433454`.

M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29, May 2000. doi: 10.1038/75556. URL `https://doi.org/10.1038/75556`.

M. Blum, H.-Y. Chang, S. Chuguransky, T. Grego, S. Kandasaamy, A. Mitchell, G. Nuka, T. Paysan-Lafosse, M. Qureshi, S. Raj, L. Richardson, G. A. Salazar, L. Williams, P. Bork, A. Bridge, J. Gough, D. H. Haft, I. Letunic, A. Marchler-Bauer, H. Mi, D. A. Natale, M. Necci, C. A. Orengo, A. P. Pandurangan, C. Rivoire, C. J. A. Sigrist, I. Sillitoe, N. Thanki, P. D. Thomas, S. C. E. Tosatto, C. H. Wu, A. Bateman, and R. D. Finn. The InterPro protein families and domains database: 20 years on. *Nucleic Acids Research*, 49(D1):D344–D354, Nov. 2020. doi: 10.1093/nar/gkaa977. URL `https://doi.org/10.1093/nar/gkaa977`.

S. Carbon, E. Douglass, B. M. Good, D. R. Unni, N. L. Harris, C. J. Mungall, S. Basu, R. L. Chisholm, R. J. Dodson, E. Hartline, P. Fey, P. D. Thomas, L.-P. Albou, D. Ebert, M. J. Kesling, H. Mi, A. Muruganujan, X. Huang, T. Mushayahama, S. A. LaBonte, D. A. Siegele, G. Antonazzo, H. Attrill, N. H. Brown, P. Garapati, S. J. Marygold, V. Trovisco, G. dos Santos, K. Falls, C. Tabone, P. Zhou, J. L. Goodman, V. B. Strelets, J. Thurmond, P. Garmiri, R. Ishtiaq, M. Rodríguez-López, M. L. Acencio, M. Kuiper, A. Lægreid, C. Logie, R. C. Lovering, B. Kramarz, S. C. C. Saverimuttu, S. M. Pinheiro, H. Gunn, R. Su, K. E. Thurlow, M. Chibucos, M. Giglio, S. Nadendla, J. Munro, R. Jackson, M. J. Duesbury, N. Del-Toro, B. H. M. Meldal, K. Paneerselvam, L. Perfetto, P. Porras, S. Orchard, A. Shrivastava, H.-Y. Chang, R. D. Finn, A. L. Mitchell, N. D. Rawlings, L. Richardson, A. Sangrador-Vegas, J. A. Blake, K. R. Christie, M. E. Dolan, H. J. Drabkin, D. P. Hill, L. Ni, D. M. Sitnikov, M. A. Harris, S. G. Oliver, K. Rutherford, V. Wood, J. Hayles, J. Bähler, E. R. Bolton, J. L. D. Pons, M. R. Dwinell, G. T. Hayman, M. L. Kaldunski, A. E. Kwitek, S. J. F. Laulederkind, C. Plasterer, M. A. Tutaj, M. Vedi, S.-J. Wang, P. D'Eustachio, L. Matthews, J. P. Balhoff, S. A. Aleksander, M. J. Alexander, J. M. Cherry, S. R. Engel, F. Gondwe, K. Karra, S. R. Miyasato, R. S. Nash, M. Simison, M. S. Skrzypek, S. Weng, E. D. Wong, M. Feuermann, P. Gaudet, A. Morgat, E. Bakker, T. Z. Berardini, L. Reiser, S. Subramaniam, E. Huala, C. N. Arighi, A. Auchincloss, K. Axelsen, G. Argoud-Puy, A. Bateman, M.-C. Blatter, E. Boutet, E. Bowler, L. Breuza, A. Bridge, R. Britto, H. Bye-A-Jee, C. C. Casas, E. Coudert, P. Denny, A. Estreicher, M. L. Famiglietti, G. Georghiou, A. Gos, N. Gruaz-Gumowski, E. Hatton-Ellis, C. Hulo, A. Ignatchenko, F. Jungo, K. Laiho, P. L. Mercier, D. Lieberherr, A. Lock, Y. Lussi, A. MacDougall, M. Magrane, M. J. Martin, P. Masson, D. A. Natale, N. Hyka-Nouspikel, S. Orchard, I. Pedruzzi, L. Pourcel, S. Poux, S. Pundir, C. Rivoire, E. Speretta, S. Sundaram, N. Tyagi, K. Warner, R. Zaru, C. H. Wu, A. D. Diehl, J. N. Chan, C. Grove, R. Y. N. Lee, H.-M. Muller, D. Raciti, K. V. Auken, P. W. Sternberg, M. Berriman, M. Paulini, K. Howe, S. Gao, A. Wright, L. Stein, D. G. Howe, S. Toro, M. Westerfield, P. Jaiswal, L. Cooper, and J. Elser. The gene ontology resource: enriching a GOld mine. *Nucleic Acids Research*, 49(D1):D325–D334, Dec. 2020. doi: 10.1093/nar/gkaa1113. URL `https://doi.org/10.1093/nar/gkaa1113`.

J. Chen, A. Althagafi, and R. Hoehndorf. Predicting candidate genes from phenotypes, functions and anatomical site of expression. *Bioinformatics*, Oct. 2020. doi: 10.1093/bioinformatics/btaa879. URL `https://doi.org/10.1093/bioinformatics/btaa879`. advance access.

M. Fey and J. E. Lenssen. Fast graph representation learning with pytorch geometric. *CoRR*, abs/1903.02428, 2019. URL `http://arxiv.org/abs/1903.02428`.

J. Gillis and P. Pavlidis. "guilt by association" is the exception rather than the rule in gene networks. *PLoS Computational Biology*, 8(3):e1002444, Mar. 2012. doi: 10.1371/journal.pcbi.1002444. URL `https://doi.org/10.1371/journal.pcbi.1002444`.

W. L. Hamilton, Z. Ying, and J. Leskovec. Inductive representation learning on large graphs. In *NIPS*, 2017.

R. Hoehndorf, P. N. Schofield, and G. V. Gkoutos. PhenomeNET: a whole-phenome approach to disease gene discovery. *Nucleic Acids Research*, 39(18):e119–e119, July 2011. doi: 10.1093/nar/gkr538. URL `https://doi.org/10.1093/nar/gkr538`.

L. A. Jeni, J. F. Cohn, and F. De La Torre. Facing imbalanced data–recommendations for the use of performance metrics. In *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction*, pages 245–251, 2013. doi: 10.1109/ACII.2013.47.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *CoRR*, abs/1609.02907, 2016. URL `http://arxiv.org/abs/1609.02907`.

M. Kulmanov and R. Hoehndorf. DeepGOPlus: improved protein function prediction from sequence. *Bioinformatics*, 36(2):422–429, 07 2019. ISSN 1367-4803. doi: 10.1093/bioinformatics/btz595. URL `https://doi.org/10.1093/bioinformatics/btz595`.

E. S. Lein, M. J. Hawrylycz, N. Ao, M. Ayres, A. Bensinger, A. Bernard, A. F. Boe, M. S. Boguski, K. S. Brockway, E. J. Byrnes, L. Chen, L. Chen, T.-M. Chen, M. C. Chin, J. Chong, B. E. Crook, A. Czaplinska, C. N. Dang, S. Datta, N. R. Dee, A. L. Desaki, T. Desta, E. Diep, T. A. Dolbeare, M. J. Donelan, H.-W. Dong, J. G. Dougherty, B. J. Duncan, A. J. Ebbert, G. Eichele, L. K. Estin, C. Faber, B. A. Facer, R. Fields, S. R. Fischer, T. P. Fliss, C. Frensley, S. N. Gates, K. J. Glattfelder, K. R. Halverson, M. R. Hart, J. G. Hohmann, M. P. Howell, D. P. Jeung, R. A. Johnson, P. T. Karr, R. Kawal, J. M. Kidney, R. H. Knapik, C. L. Kuan, J. H. Lake, A. R. Laramee, K. D. Larsen, C. Lau, T. A. Lemon, A. J. Liang, Y. Liu, L. T. Luong, J. Michaels, J. J. Morgan, R. J. Morgan, M. T. Mortrud, N. F. Mosqueda, L. L. Ng, R. Ng, G. J. Orta, C. C. Overly, T. H. Pak, S. E. Parry, S. D. Pathak, O. C. Pearson, R. B. Puchalski, Z. L. Riley, H. R. Rockett, S. A. Rowland, J. J. Royall, M. J. Ruiz, N. R. Sarno, K. Schaffnit, N. V. Shapovalova, T. Sivisay, C. R. Slaughterbeck, S. C. Smith, K. A. Smith, B. I. Smith, A. J. Sodt, N. N. Stewart, K.-R. Stumpf, S. M. Sunkin, M. Sutram, A. Tam, C. D. Teemer, C. Thaller, C. L. Thompson, L. R. Varnam, A. Visel, R. M. Whitlock, P. E. Wohnoutka, C. K. Wolkey, V. Y. Wong, M. Wood, M. B. Yaylaoglu, R. C. Young, B. L. Youngstrom, X. F. Yuan, B. Zhang, T. A. Zwingman, and A. R. Jones. Genome-wide atlas of gene expression in the adult mouse brain. *Nature*, 445(7124):168–176, Dec. 2006. doi: 10.1038/nature05453. URL `https://doi.org/10.1038/nature05453`.

G. Li, M. Müller, A. Thabet, and B. Ghanem. Deepgcns: Can gcns go as deep as cnns? In *The IEEE International Conference on Computer Vision (ICCV)*, 2019.

G. Li, C. Xiong, A. Thabet, and B. Ghanem. Deepergcn: All you need to train deeper gcns. *CoRR*, abs/2006.07739, 2020.

R. Liaw, E. Liang, R. Nishihara, P. Moritz, J. E. Gonzalez, and I. Stoica. Tune: A research platform for distributed model selection and training, 2018.

T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546, 2013. URL `http://arxiv.org/abs/1310.4546`.

S. Oliver. Guilt-by-association goes global. *Nature*, 403(6770):601–602, Feb. 2000. doi: 10.1038/35001165. URL https://doi.org/10.1038/35001165.

A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

C. L. Smith and J. T. Eppig. The mammalian phenotype ontology: enabling robust annotation and comparative analysis. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 1(3):390–399, Nov. 2009. doi: 10.1002/wsbm.44. URL https://doi.org/10.1002/wsbm.44.

D. Szklarczyk, A. Franceschini, S. Wyder, K. Forslund, D. Heller, J. Huerta-Cepas, M. Simonovic, A. Roth, A. Santos, K. P. Tsafou, M. Kuhn, Peer, L. J. Jensen, and C. von Mering. STRING v10: protein–protein interaction networks, integrated over the tree of life. *Nucleic Acids Research*, 43(D1): D447–D452, Oct. 2014. doi: 10.1093/nar/gku1003. URL https://doi.org/10.1093/nar/gku1003.

M. Trebacz, Z. Shams, M. Jamnik, P. Scherer, N. Simidjievski, H. A. Terre, and P. Liò. Using ontology embeddings for structural inductive bias in gene expression data analysis. *CoRR*, abs/2011.10998, 2020.

A. Vazquez, A. Flammini, A. Maritan, and A. Vespignani. Global protein function prediction from protein-protein interaction networks. *Nature Biotechnology*, 21(6):697–700, May 2003. doi: 10.1038/nbt825. URL https://doi.org/10.1038/nbt825.

M. Zitnik and J. Leskovec. Predicting multicellular function through multi-layer tissue networks. *CoRR*, abs/1707.04638, 2017. URL http://arxiv.org/abs/1707.04638.