NeuroData SIMPLEX Report: May 2017

The following report documents the progress made by the labs of PI Joshua T. Vogelstein and Co-PIs Randal Burns and Carey Priebe at Johns Hopkins University towards goals set by the DARPA SIMPLEX grant.

Contents

1 Bibliography

Manuscripts

[1] C. E. Priebe, Y. Park, M. Tang, A. Athreya, V. Lyzinski, J. T. Vogelstein, Y. Qin, B. Cocanougher, K. Eichler, M. Zlatic et al., "Semiparametric spectral modeling of the Drosophila connectome," arXiv preprint arXiv:1705.03297, 2017.

[2] D. Fishkind, S. Adali, H. Patsolic, L. Meng, V. Lyzinski, and C. Priebe, "Seeded Graph Matching," arXiv preprint arXiv:1209.0367, 2017.

[3] H. Patsolic, Y. Park, V. Lyzinski, and C. Priebe, "Vertex Nomination Via Local Neighborhood Matching," arXiv preprint arXiv:1705.00674, 2017.

2 Statistical Theory and Methods

2.1 meda @JesseLP

A docker container https://hub.docker.com/r/neurodata/synaptograms/ has been developed to view multi-channel muti-dimensional image data. Array tomography (AT) provides a good use-case. AT data consist of (3+c)-dimensional data, where c is the number of features/channels collected. A putative synapse location corresponds to an 11x11x11 pixel cube in the image and this cube will have multiple corresponding channels. These channels can express the presence or absence of a certain protein marker such as Synapsin or VGlut2.

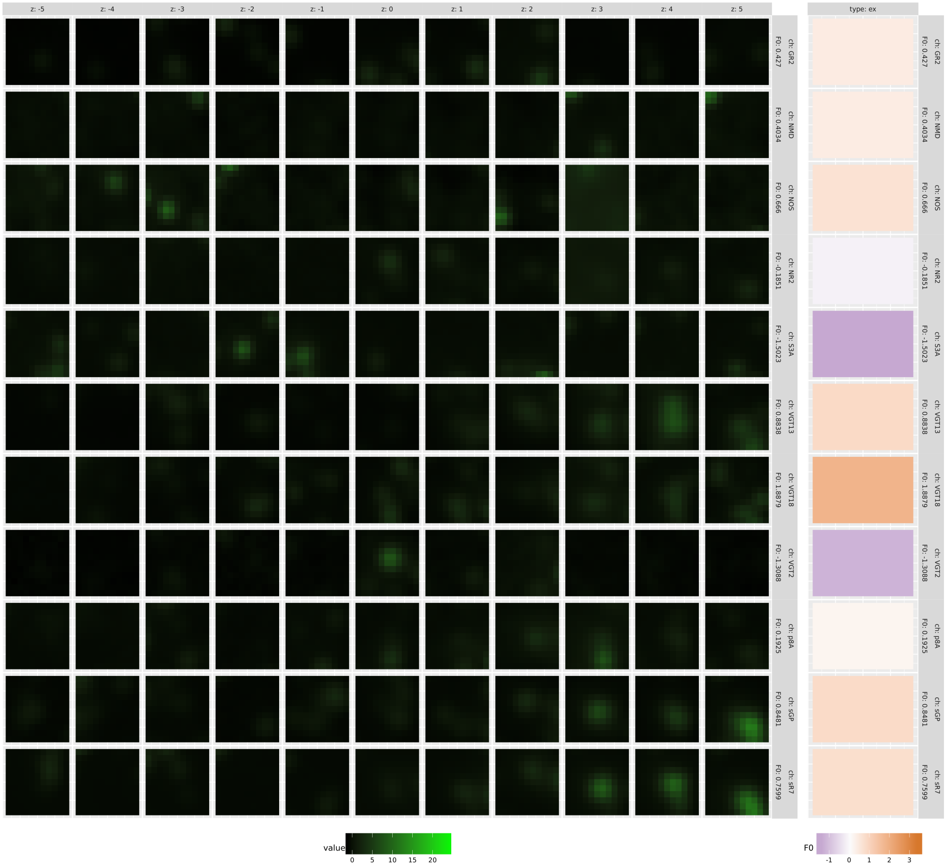


Figure 1: Using a synapse detection algorithm we believe that there is a synapse at this location. The location is defined by an 11x11x11 cube with 11 feature channels. Each row of the figure corresponds to the cube over a specific feature channel. The cube has been splayed out in the z-direction over the columns. The far right column corresponds to the sum statistic of each row. This plot enables viewing of an 11x11x11xc object nicely on screen, where c in this case happens to be 11.

2.2 Randomer Forest (RerF)

Most recently, we have implemented a method for computing the relative importance of features used to make splits in RerF. The motivation for this is two-fold: 1) model interpretability/knowledge extraction and 2) improving model performance by focusing construction of future trees using the most important split features. Feature importance is computed as follows. For each feature, the values of that feature in the training set are randomly permuted. Then out-of-bag error is computed using the permuted data. Feature importance is the relative increase in out-of-bag error when using the permuted data over the original (non-permuted) data. The idea is that more important features should increase the error when perumuted more than less important features. The method is simple but can be computationally expensive if many different features were used in the forest. The feature importance of the top 25 features for the Sparse Parity synthetic dataset with n = 1000 training observations and p = 10 dimensions is shown below. 6,038 features were evaluated in total. In this dataset, only three of the dimensions are informative. Therefore, we would expect features with a high importance value to only be linear combinations of the three informative dimensions. Sure enough, the top three features are the single dimensions having signal, and the other 25 are all linear combinations of just these dimensions.

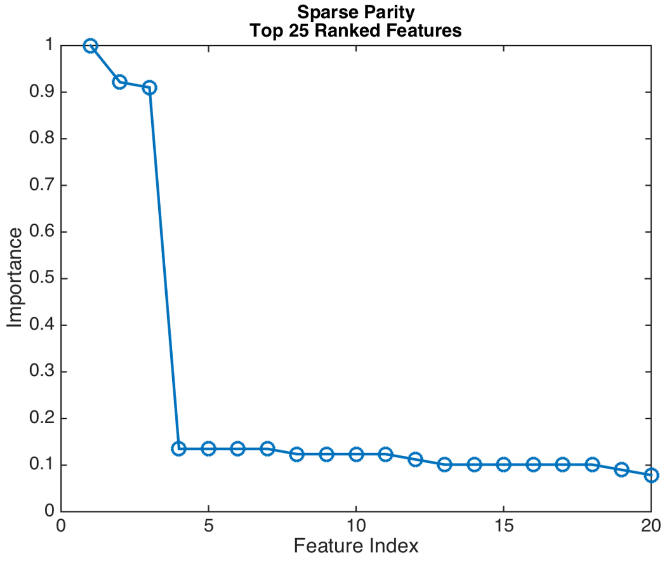


Figure 2: Importance of the 25 features with the highest importance as found by RerF. The top three features precisely the three dimensions that have signal. The remaining 22 features are linear combinations of these three dimensions.

2.3 Law of Large Graphs

We note that low-rank methods can often be more easily interpreted. Moreover, eigenmode is observed among the embedded latent positions, with respect to different lobes in particular. This suggests to use low-rank methods from another perspective. For all the 70 different regions based on the Desikan atlas (35 for each hemisphere), each one is assigned to one of the 10 lobes (5 for each hemisphere), i.e. Frontal, Parietal, Occipital, Temporal, and other. And we do a permutation test as following.

70 vertices are connected spatially as in the Desikan atlas. Let the adjacency matrix of these 70 vertices to be A. $A_{ij} = 1$ means vertex i and vertex j are spatially connected. We say vertex j is a neighbor of vertex i if $A_{ij} = 1$. We define l_i be the lobe i.d. for vertex i.

We define a uniform 1-flip to be:

- Select a pair of adjacent vertices (vertex i_1 and vertex j_1) across the boundary of lobes uniformly, i.e. $A_{i_1 j_1} = 1$ and $l(i_1) \neq l(j_1)$;
- Uniformly select another pair of adjacent vertices (vertex i_2 and vertex j_2 where $i_1 \neq i_2$ and $j_1 \neq j_2$) across the same boundary of lobes uniformly, i.e. $A_{i_2 j_2} = 1$ and $l(i_1) = l(i_2)$ and $l(j_1) = l(j_2)$;
- Reassign vertex j_1 to lobe l_{i_1} and reassign vertex i_2 to lobe l_{j_2}.

By the definition, after a uniform 1-flip, the number of vertices in each lobe keeps the same, where only two vertices are changed to a different lobe.

We define a uniform k-flip to be:

- Sequentially run the uniform 1-flip k times.

Note that after a uniform k-flip, the number of vertices in each lobe still keeps the same.

Let $X = [X_1, \cdots, X_n]^\top$ be the latent positions, where X_i is the latent position for vertex i sampled from distribution f. Test statistic $T(X, l)$ is defined as:

$$T(X,l) = \frac{\sum_{i \neq j, l(i)=l(j)} \|X_i - X_j\|_2}{\sum_{i \neq j, l(i)=l(j)} 1} - \frac{\sum_{i \neq j, l(i) \neq l(j)} \|X_i - X_j\|_2}{\sum_{i \neq j, l(i) \neq l(j)} 1}$$

H_0: Differences between latent positions within lobes are the same compared to across lobes, i.e. $E_f[T(X,l)] = 0$.

H_A: Differences between latent positions within lobes are smaller compared to across lobes, i.e. $E_f[T(X,l)] < 0$.

We ran 1000 simulations for each number of flips and plot the results for the permutation test as in Figure 3. The x-axis represents the different number of flips, while the y-axis represents the measure according to the lobe assignment, i.e. within lobes distances minus the across lobe distances. The dashed line is the baseline for the measure based on the true lobe assignment without any flipping. As the number of flips increases, we can see a clear evidence that H_0 is rejected with respect to H_A. Thus the embedded latent positions based on the low-rank method reflect the eigenmode with respect to different lobes.

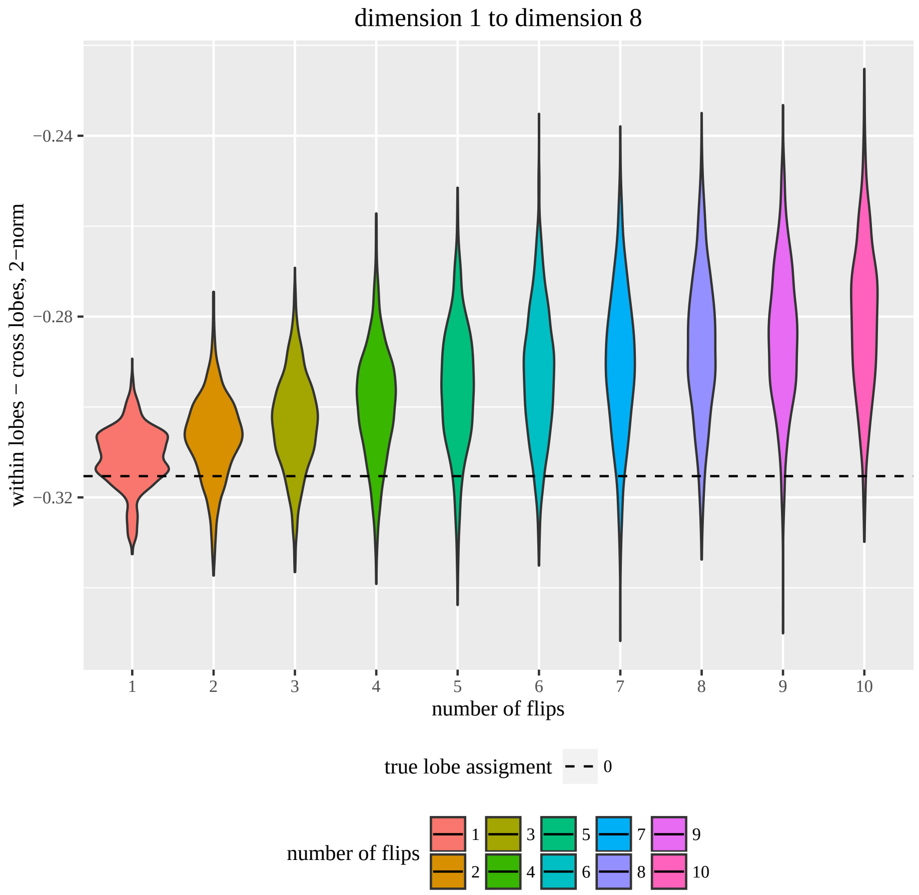


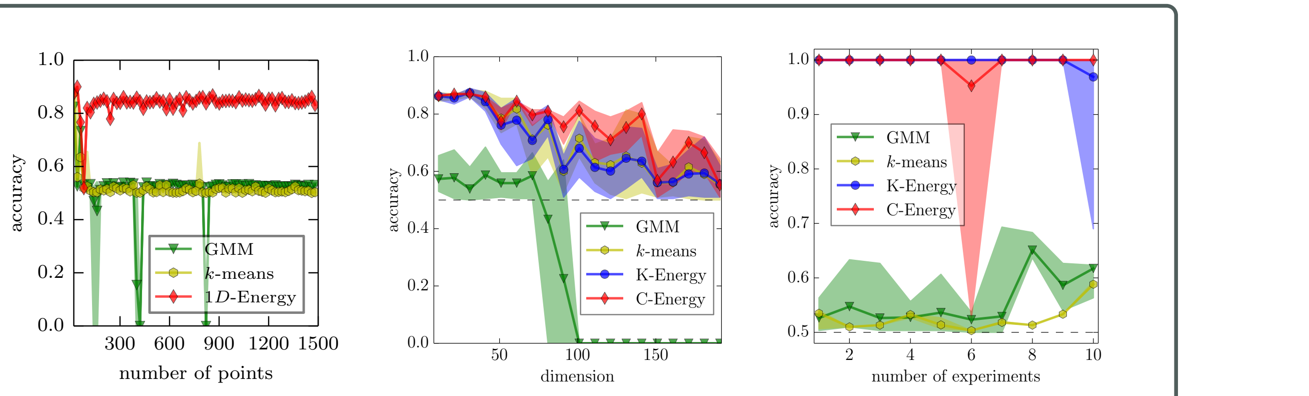
Figure 3: **Violin plot of the permutation test.** We ran 1000 simulations for each number of flips. The x-axis represents the different number of flips, while the y-axis represents the measure according to the lobe assignment, i.e. within lobes distances minus the across lobe distances. The dashed line is the baseline for the measure based on the true lobe assignment without any flipping. As the number of flips increases, we can see a clear evidence that H_0 is rejected with respect to H_A. Thus the embedded latent positions based on the low-rank method reflect the eigenmode with respect to different lobes.

2.4 Non-Parametric Shape Clustering

We proposed a clustering algorithm based on energy statistics. Energy statistics provides a nonparametric test-statistic for equality of distributions in the sense that it does not assume any distribution of the data. We used this formalism to propose a clustering algorithm consistent with energy statistics. Our method can be shortly summarized as follows. Suppose we have data $\{x_i\}_{i=1}^n$, where $x_i \in \mathcal{X}$ lives in an arbitrary space endowed with a semimetric of negative type $\rho : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, and we want to find k disjoint partitions $\{\mathcal{C}_1, \dots, \mathcal{C}_k\}$. We minimize the within energy function $W = \sum_{i=1}^k \frac{n_j}{2} g(\mathcal{C}_j, \mathcal{C}_j)$, where $g(\mathcal{C}_i, \mathcal{C}_j) = \frac{1}{n_i n_j} \sum_{x \in \mathcal{C}_i} \sum_{y \in \mathcal{C}_j} \rho(x, y)$. We demonstrated that this is equivalent to the following QCQP problem:

$$\max_{Y} \operatorname{Tr} \left\{ Y^\top K Y \right\} \qquad \text{s.t. } Y^\top Y = I, Y \geq 0, Y Y^\top e = e \tag{1}$$

where $e = (1, 1, \dots, 1)^\top$ is the all-ones vector. Above, the kernel matrix K is computed based on ρ. The above problem also appears in connection to kernel k-means, spectral clustering, and normalized cuts, thus we bring energy statistics based clustering into this broad picture. We developed an iterative algorithm to find local optimizers of (1), which is NP-hard thus finding global solutions are not feasible. Our method is (i) nonparametric, (ii) does not use the concept of a cluster mean, which can be problematic for some datasets such as images, (iii) can work on arbitrary spaces $\mathcal{X}$, and (iv) flexible enough to incorporate prior information about the data by choosing a suitable ρ. We compare our method to standard k-means and GMM, which are by large the most used methods. In Figure 4 we can see how energy clustering can outperform these two methods even on gaussian settings. We are currently finalizing a paper on this topic which should be submitted for publication soon.



(a) (b) (c)

Figure 4: We compute the clustering accuracy based on the true labels. Our algorithm correspond to the red dots in the plots, which in these experiments is always superior than k-means or GMM . (a) Two log-normal distributions with $\mu_1 = 0$, $\sigma_1 = 0.3$, $\mu_2 = -1.5$, and $\sigma_2 = 1.5$. (b) Two normal distributions where we increase the number of dimensions and keep the number of points fixed. We have $\Sigma_1 = \Sigma_2 = I$, $\mu_1 = 0$, $\mu_2 = 0.7 \times (1, \cdots, 1, 0, \cdots, 0)$ where we have signal in the first 10 dimensions only. (c) Two concentric circles with some noise.

3 Scalable Algorithm Implementations

3.1 FlashX

We re-implement Gaussian Mixture Model (GMM) with FlashR, which follows the implementation in scikit-learn. This GMM implementation is much more stable and mature than the previous one and supports covariance matrices with different constraints. To demonstrate the speed and scalability of our GMM implementation, we run our implementation on the k15f0 dataset, which has over a million data points, and compare it against mclust. As shown in Table 1, the GMM implementation in FlashR takes a few minutes to converge on the dataset, while mclust takes hours and eventually fails to converge. Figure 5 further shows the BIC value for different GMM models on the dataset, which suggests that the dataset have about 20 clusters.

Table 1: Runtime of FlashR GMM vs. mclust on the k15f0 dataset with over a million data points.

	FlashR	mclust
k=40, cov.type=”full”	11 min	hours and fails to converge
k=10, cov.type=”full”	2.5 min	hours and fails to converge
k=40, cov.type=”diag”	7.6 min	hours and fails to converge
k=10, cov.type=”diag”	1.2 min	hours and fails to converge

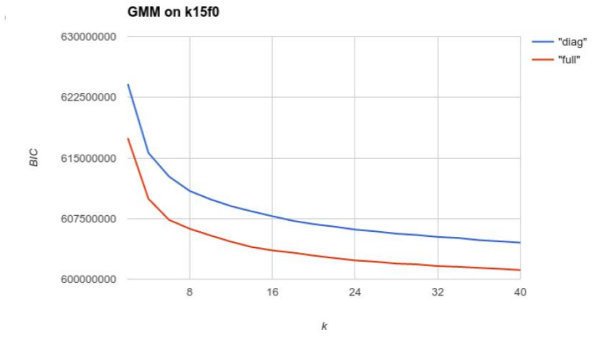


Figure 5: Model selection for GMM on the dataset with BIC. We vary the number of clusters (k) and the type of covariance matrices. ”diag” means diagonal covariance matrices and ”full” means unconstrained covariance matrices.

3.2 knor: K-means NUMA Optimized Routines

knor is a highly optimized k-means library that performs one to two orders of magnitude better than other state-of-the-art machine learning libraries like Spark's MLlib, H_2O and Dato. As part of our commitment to developing user-friendly and easily integratable opensource tools, we developed both R and Python bindings for **knor**. We have one line-installable R package that has all the in-memory functionality of **knor**. We link the Github repo https://github.com/flashxio/knorR. We also develop Python bindings that have the same functionality as those within R. Our bindings are integrated into our main repo https://github.com/flashxio/knor/tree/dev/python. We measure the performance of our bindings relative to the native C++ and find them to compare well when we perform experiments with data on disk for our in-memory routine called **knori**. Figure 6 displays our performance. We provide base docker images to test the functionality and performance of our bindings at https://hub.docker.com/r/flashxio/knorr-base/. Figure 7 displays our pullable docker images.

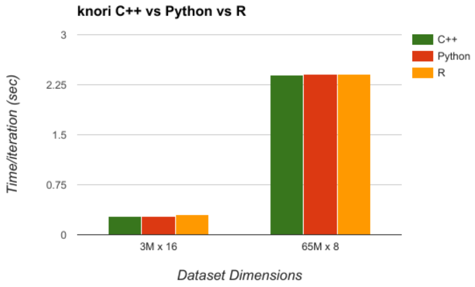
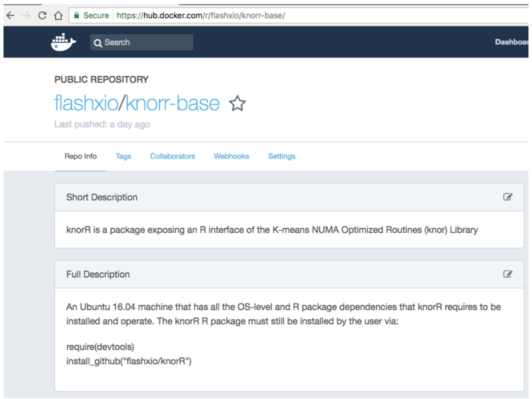


Figure 6: The performance of bindings when data are located on disk.



PUBLIC REPOSITORY

flashxio/knorr-base ☆

Last pushed: a day ago

Repo Info | Tags | Collaborators | Webhooks | Settings

Short Description

knorR is a package exposing an R interface of the K-means NUMA Optimized Routines (knor) Library

Full Description

An Ubuntu 16.04 machine that has all the OS-level and R package dependencies that knorR requires to be installed and operate. The knorR R package must still be installed by the user via:

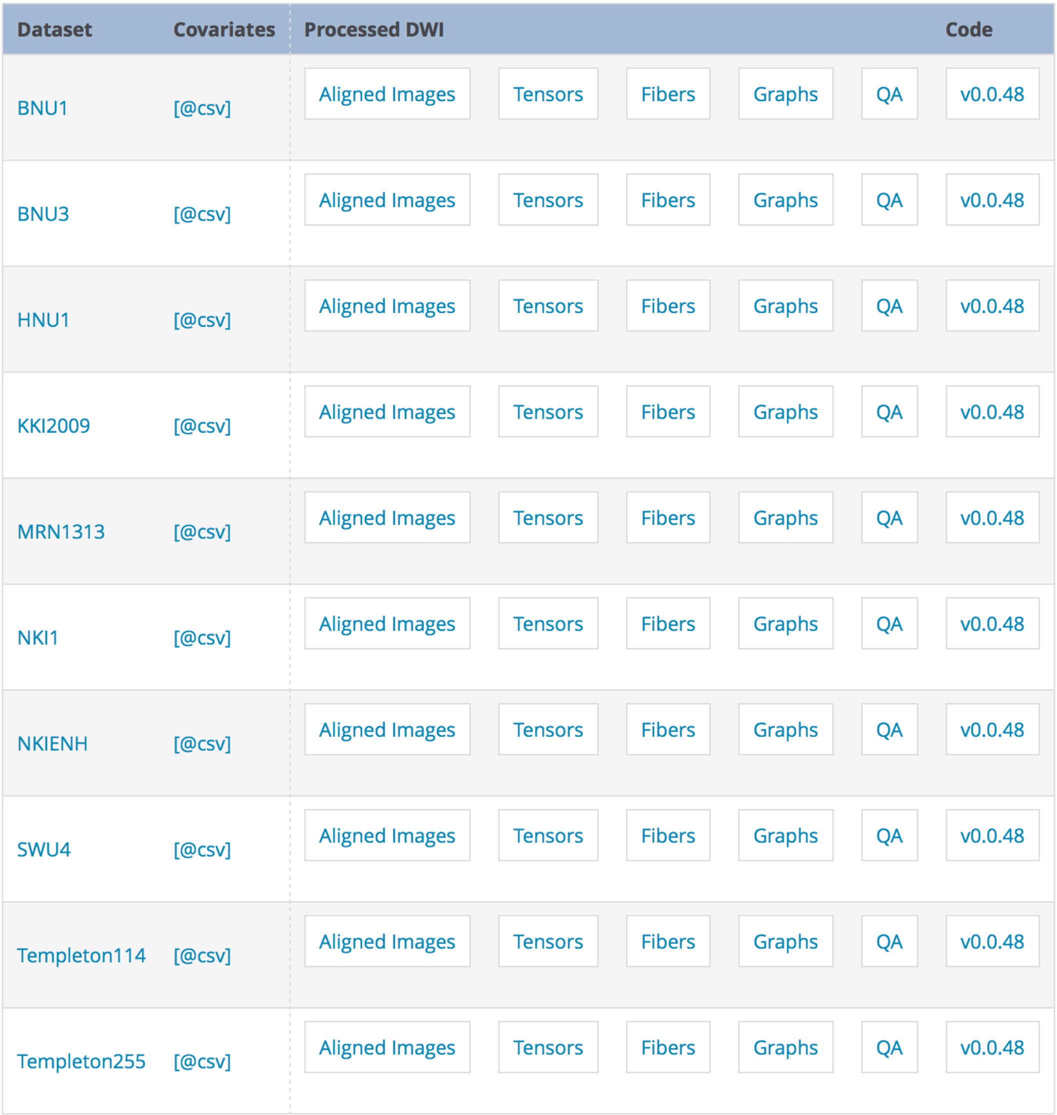
```
require(devtools)
install_github("flashxio/knorR")
```

Figure 7: The docker hub base R image.

4 Data: What's in the Cloud

We have now pushed several of our canonical datasets to the cloud, including 8 whole brain CLARITY specimens and two electron microscopy datasets: bock11, and kasthuri11:

Reference	Modality	Species	Bits	Proj	Ch	T	GV	Res	GB
Bhatla[1]	EM	C. elegans	8	3	3	1	437	6	248
Bock[2]	EM	M. musculus	8	1	1	1	20,249	11	13,312
Harris[3]	EM	R. rattus	8	3	3	1	19	4	9
Kasthuri[4]	EM	M. musculus	8	1	1	1	1,063	8	577
Lee[5]	EM	M. musculus	8	1	1	1	22,334	8	11,264
Ohyama[6]	EM	D. melanogaster	8	1	1	1	2,609	7	2,458
Takemura[7]	EM	D. melanogaster	8	1	1	1	190	5	203
Bloss[8]	AT	M. musculus	8	1	3	1	363	4	215
Collman[9]	AT	M. musculus	8	1	14	1	13	4	2
Unpublished	AT	M. musculus	16	1	24	1	29	3	23
Weiler[10]	AT	M. musculus	16	12	288	1	215	3	141
Vladimirov[11]	Ophys	D. rerio	16	1	1	100	9	4	9
Dyer[12]	XCT	M. musculus	8	1	1	1	3	3	3
Randlett[13]	LM	D. rerio	16	1	28	1	4	2	4
Kutten[14]	CL	M. musculus	16	1	23	1	7,191	6	6,727
Grabner[15]	MR	H. sapiens	16	1	3	1	<1	1	< 1
Totals	–	–	–	29	349	–	47,508	–	28,441



Dataset	Covariates	Processed DWI					Code
BNU1	[@csv]	Aligned Images	Tensors	Fibers	Graphs	QA	v0.0.48
BNU3	[@csv]	Aligned Images	Tensors	Fibers	Graphs	QA	v0.0.48
HNU1	[@csv]	Aligned Images	Tensors	Fibers	Graphs	QA	v0.0.48
KKI2009	[@csv]	Aligned Images	Tensors	Fibers	Graphs	QA	v0.0.48
MRN1313	[@csv]	Aligned Images	Tensors	Fibers	Graphs	QA	v0.0.48
NKI1	[@csv]	Aligned Images	Tensors	Fibers	Graphs	QA	v0.0.48
NKIENH	[@csv]	Aligned Images	Tensors	Fibers	Graphs	QA	v0.0.48
SWU4	[@csv]	Aligned Images	Tensors	Fibers	Graphs	QA	v0.0.48
Templeton114	[@csv]	Aligned Images	Tensors	Fibers	Graphs	QA	v0.0.48
Templeton255	[@csv]	Aligned Images	Tensors	Fibers	Graphs	QA	v0.0.48

Figure 8: A current snapshot of the diffusion magnetic resonance imaging data resulting from the pipeline m2g for generating human connectomes at scale.

5 Scientific Pipelines: Infrastructure & Dataset Specific Progress

5.1 fngs

Through the fngs pipeline, we develope a robust processing pipeline and web service for providing automated acquisition of functional connectomes from structural and functional MRI. The fngs pipeline was developed around the glass-box principle; each step of the pipeline produces intuitive and descriptive quality assurance so users can be confident that the internals of the pipeline are performing properly. We provide an open-source docker container, links to all code, and have numerous tutorials and demos available Tutorials for users to receive a step-by-step introduction to the pipeline. Moreover, we provide an interactive schematic of the pipeline Schematic. Note that the headings in each box at the bottom link to the documentation for the respective steps of the pipeline. Using the one-click cloud deployment of fngs, we were able to analyze 1200 human connectomes in 7 hours for a total of $80. The cloud deployment procedure of the fngs pipeline can be seen in Figure (9).

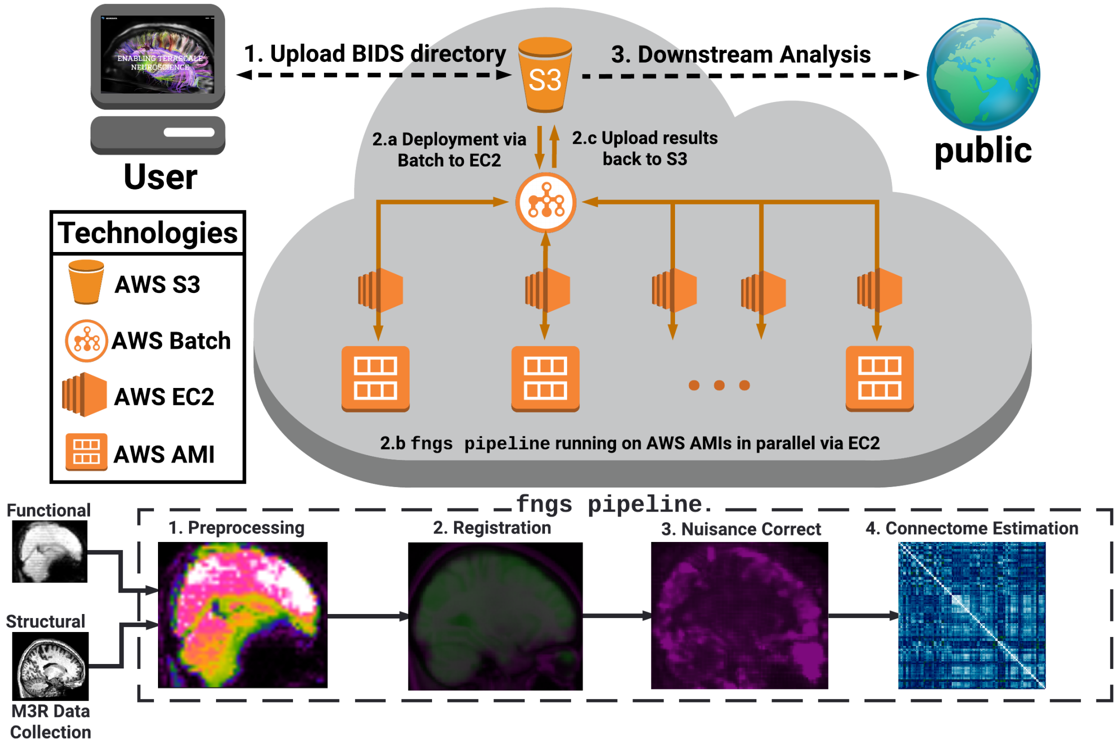


Figure 9: **FNGS workflow**. The deployment workflow for the FNGS pipeline. Users provide a directory according to the BIDs spec to the FNGS cloud controller locally via the docker container at (1). The data is uploaded directly to AWS S3 cloud drives. The controller then initiates the Batch deployment procedure at (2.a), which interfaces between S3 and EC2 cloud computers to provide the MRI scans to EC2 instances pre-loaded with the FNGS pipeline for analysis. After the scans are finished being analyzed on the EC2 instances at (2.b), the results are then re-uploaded back to the S3 cloud drive at (2.c). The user can then navigate to the S3 cloud drive for downstream analyses.

6 Reference Datasets

References

[1] N. Bhatla, R. Droste, S. R. Sando, A. Huang, and H. R. Horvitz, "Distinct neural circuits control rhythm inhibition and spitting by the myogenic pharynx of c. elegans," Current Biology, vol. 25, no. 16, pp. 2075 – 2089, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0960982215007460

[2] D. D. Bock, W.-C. A. Lee, A. M. Kerlin, M. L. Andermann, G. Hood, A. W. Wetzel, S. Yurgenson, E. R. Soucy, H. S. Kim, and R. C. Reid, "Network anatomy and in vivo physiology of visual cortical neurons," Nature, vol. 471, no. 7337, pp. 177–182, 03 2011.

[3] K. M. Harris, J. Spacek, M. E. Bell, P. H. Parker, L. F. Lindsey, A. D. Baden, J. T. Vogelstein, and R. Burns, "A resource from 3D electron microscopy of hippocampal neuropil for user training and tool development," Scientific Data, vol. 2, p. 150046, Aug 2015.

[4] N. Kasthuri, K. J. Hayworth, D. R. Berger, R. L. Schalek, J. Conchello, S. Knowles-Barley, D. Lee, A. Vázquez-Reina, V. Kaynig, T. R. Jones, M. Roberts, J. L. Morgan, J. C. Tapia, H. S. Seung, W. G. Roncal, J. T. Vogelstein, R. Burns, D. L. Sussman, C. E. Priebe, H. Pfister, and J. W. Lichtman, "Saturated reconstruction of a volume of neocortex," Cell, vol. 162, pp. 648–661, 05 2016. [Online]. Available: http://dx.doi.org/10.1016/j.cell.2015.06.054

[5] W.-c. A. Lee, V. Bonin, M. Reed, B. J. Graham, G. Hood, K. Glattfelder, and R. C. Reid, "the visual cortex," Nature, vol. 532, no. 7599, pp. 370–374, 2016. [Online]. Available: http://dx.doi.org/10.1038/nature17192

[6] T. Ohyama, C. M. Schneider-Mizell, R. D. Fetter, J. V. Aleman, R. Franconville, M. Rivera-Alba, B. D. Mensh, K. M. Branson, J. H. Simpson, J. W. Truman, A. Cardona, and M. Zlatic, "A multilevel multimodal circuit enhances action selection in drosophila," Nature, vol. 520, no. 7549, pp. 633–639, 04 2015.

[7] S.-y. Takemura, A. Bharioke, Z. Lu, A. Nern, S. Vitaladevuni, P. K. Rivlin, W. T. Katz, D. J. Olbris, S. M. Plaza, P. Winston, T. Zhao, J. A. Horne, R. D. Fetter, S. Takemura, K. Blazek, L.-A. Chang, O. Ogundeyi, M. A. Saunders, V. Shapiro, C. Sigmund, G. M. Rubin, L. K. Scheffer, I. A. Meinertzhagen, and D. B. Chklovskii, "A visual motion detection circuit suggested by drosophila connectomics," Nature, vol. 500, no. 7461, pp. 175–181, 08 2013. [Online]. Available: http://dx.doi.org/10.1038/nature12450

[8] E. B. Bloss, M. S. Cembrowski, B. Karsh, J. Colonell, R. D. Fetter, and N. Spruston, "Structured dendritic inhibition supports branch-selective integration in ca1 pyramidal cells," Neuron, vol. 89, no. 5, pp. 1016 – 1030, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0896627316000544

[9] F. Collman, J. Buchanan, K. D. Phend, K. D. Micheva, R. J. Weinberg, and S. J. Smith, "Mapping synapses by conjugate light-electron array tomography," The Journal of Neuroscience, vol. 35, no. 14, pp. 5792–5807, 2015.

[10] N. C. Weiler, F. Collman, J. T. Vogelstein, R. Burns, and S. J. Smith, "Molecular architecture of barrel column synapses following experience-dependent plasticity." Nature Scientific Data, 2014.

[11] J. Freeman, N. Vladimirov, T. Kawashima, Y. Mu, N. J. Sofroniew, D. V. Bennett, J. Rosen, C.-T. Yang, L. L. Looger, and M. B. Ahrens, "Mapping brain activity at scale with cluster computing," Nature Methods, no. July, jul 2014. [Online]. Available: http://www.nature.com/doifinder/10.1038/nmeth.3041

[12] E. L. Dyer, W. Gray Roncal, H. L. Fernandes, D. Gürsoy, X. Xiao, J. T. Vogelstein, C. Jacobsen, K. P. Körding, and N. Kasthuri, "Quantifying mesoscale neuroanatomy using x-ray microtomography," "arXiv", "2016".

[13] O. Randlett, C. L. Wee, E. A. Naumann, O. Nnaemeka, D. Schoppik, J. E. Fitzgerald, R. Portugues, A. M. B. Lacoste, C. Riegler, F. Engert, and A. F. Schier, "Whole-brain activity mapping onto a zebrafish brain atlas," Nat Meth, vol. 12, no. 11, pp. 1039–1046, 11 2015.

[14] K. S. Kutten, J. T. Vogelstein, N. Charon, L. Ye, K. Deisseroth, and M. I. Miller, "Deformably Registering and Annotating Whole CLARITY Brains to an Atlas via Masked LDDMM," in Proceedings SPIE 9896, Optics, Photonics and Digital Technologies for Imaging Applications IV, P. Schelkens, T. Ebrahimi, G. Cristóbal, F. Truchetet, and P. Saarikko, Eds., 2016.

[15] G. Grabner, A. L. Janke, M. M. Budge, D. Smith, J. C. Pruessner, and D. L. Collins, "Symmetric atlasing and model based segmentation: An application to the hippocampus in older adults," in Medical Image Computing and Computer-Assisted Intervention - MICCAI 2006, 2006, pp. 58–66.