Bayes Optimal Unlabeled Graph Classification: Applications in Statistical Connectomics

Joshua T. Vogelstein and Carey E. Priebe

Abstract—Graph classification algorithms often do not incorporate vertex label information in their classifiers. In this work, we investigate the extent to which discarding vertex labels can hinder classification performance, and for which random graph models it would be expected to matter. Via theory we demonstrate a collection of results. Specifically, if one "shuffles" the graphs prior to classification, the vertex label information is irretrievably lost, which can degrade misclassification performance (and does whenever the vertex labels have class-conditional signal). Thus, while one cannot hope to recover the labels, trying to recover the labels actually results in a consistent estimate of the optimal graph invariant. This approach therefore solves the question of "which invariant to use" for any graph classification problem, at least asymptotically. Via simulation we demonstrate that a finite (and small) number of training samples can be sufficient to achieve this bound. Finally, we apply this approach to a "connectome" classification problem (a connectome is the complete set of connections within a brain). Unshuffling the graphs indeed improves performance, although not over the best performance achievable composing a number of graph invariant and machine learning tools. Thus, given any unlabeled graph classification problem, the relative performance of an unshuffling approach might be difficult to predict with small sample sizes.

Index Terms—statistical inference, graph theory, network theory, structural pattern recognition, connectome.

1 Introduction

This work addresses graph classification with and without vertex labels. Consider the following idealized scenario. Let $\mathbb{G} : \Omega \mapsto \mathcal{G}_n$ be a graph-valued random variable taking values $G \in \mathcal{G}_n$. Let Y be a categorical random variable, $Y : \Omega \mapsto \mathcal{Y} \subseteq \mathbb{Z}$, such that each graph has an associated class. A graph classifier $h : \mathcal{G}_n \mapsto \mathcal{Y}$ is any function that maps from graph space to class space. The *risk* of a classifier is the expected misclassification rate, $L_h = \mathbb{E}[h[G] \neq Y]$. The optimal classifier is the classifier that minimizes risk:

$$h^* = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \, \mathbb{E}[h(\mathbb{G}) \neq Y]. \tag{1}$$

where $\mathcal{H}$ is the set of possible classifiers. Let $L^* = L_{h^*}$ indicate minimal (optimal) risk.

Graph classification differs from classification of vector-valued random variables in several key aspects. First, the structure of a graph may encode information. Second, the vertex labels may or may not be observed. In unobserved scenarios, NP-hard problems rear their ugly heads [1].

2 Graph Shuffling

Let G and G' be isomorphic to another if and only if $\exists \sigma[G] = G'$, where $\sigma : \mathcal{G}_n \mapsto \mathcal{G}_n$ is a vertex permutation function. An *unlabeled graph* is actually a set: $\widetilde{G} = \{\sigma[G] : \forall \sigma \in \Sigma_n\}$, where Σ_n is the set of permutation functions on n vertices. Let $\widetilde{\mathcal{G}}_n$ be unlabeled graph space. A *shuffle channel*, $\mathcal{C} : \mathcal{G}_n \mapsto \widetilde{\mathcal{G}}_n$ is a channel that takes as input a graph, uniformly at random selects a permutation function $\sigma \in \Sigma_n$, and outputs an unlabeled graph. Let $\widetilde{h} : \widetilde{\mathcal{G}}_n \mapsto \mathcal{Y}$ be an unlabeled graph classifier, and $\widetilde{L}^*$ be the unlabeled optimal risk for the unlabeled optimal classifier, $\widetilde{h}^*$.

- *J.T. Vogelstein and C.E. Priebe are with the Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD 21218.*
  E-mail: joshuav@jhu.edu
- *R.J. Vogelstein is with the Johns Hopkins University Applied Physics Laboratory, Laurel, MD, 20723.*

3 Model Based Graph Classifiers

Let $\mathbb{P} = \mathbb{P}_{\mathbb{G},Y}$ indicate a joint distribution of graphs and classes. This joint distribution may be decomposed into the product of a likelihood and prior term: $\mathbb{P}_{\mathbb{G},Y} = \mathbb{P}_{\mathbb{G}|Y}\pi_Y$. Let π_y denote class prior probabilities, $P[Y = y] \triangleq P[\{\omega : Y(\omega) = y\}]$, and let $\mathbb{P}_{\mathbb{G}|Y=y} = \mathbb{P}_y$ denote the class-conditional distribution. Assuming the data were sampled from a distribution, $\mathbb{P}_{\mathbb{G},Y}$, a Bayes classifier which chooses the maximum a posteriori class is optimal [2]:

$$h^*[G] = \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \, \mathbb{P}_y \pi_y. \tag{2}$$

Let $\widetilde{\mathbb{P}} = \mathbb{P}_{\widetilde{\mathbb{G}},Y}$ indicate the joint distribution of unlabeled (or shuffled) graphs and classes. In other words, $\widetilde{\mathbb{P}}$ is the same as $\mathbb{P}$ except that the graphs have been passed through a shuffle channel, so $\mathbb{P}$ is a distribution over graphs, and $\widetilde{\mathbb{P}}$ is a distribution over sets of isomorphic graphs. A Bayes optimal unlabeled graph classifier is:

$$\widetilde{h}^*[\widetilde{G}] = \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \, \mathbb{P}_{\widetilde{\mathbb{G}}|Y=y} \mathbb{P}_{Y=y}. \tag{3}$$

Let L^* and $\widetilde{L}^*$ be the Bayes risk and unlabeled Bayes risk, under $\mathbb{P}$ and $\widetilde{\mathbb{P}}$, respectively.

It may be illustrative to consider the class-conditional distributions of graphs as a categorical distribution: $\mathbb{P}_y[G] = \text{Cat}(G; \boldsymbol{\theta}_y)$, where $\boldsymbol{\theta}_y = (\theta_1, \dots, \theta_d)$, and $\theta_i \geq 0, \sum_{i \in [d]} \theta_i = 1$, $d = |\mathcal{G}_n|$ and $[d] = \{1, 2, \dots, d\}$. After passing $\mathbb{G}$ through a shuffle channel, all graphs that were isomorphic to one another must have the same probability (because the shuffle channel samples a permutation uniformly at random from Σ_n). The distribution resulting from passing $\mathbb{G}$ through a shuffle channel can therefore also be thought of as a categorical distribution, but over sets of isomorphic graphs, $\widetilde{\mathbb{P}}_y[\widetilde{G}] = \mathbb{P}_{\widetilde{\mathbb{G}}|Y=y}[\widetilde{G}] = \text{Cat}(\widetilde{G}; \boldsymbol{\eta}_y)$, where $\boldsymbol{\eta}_y = (\boldsymbol{\eta}_1, \dots, \boldsymbol{\eta}_{\widetilde{d}})$, with similar constraints on $\boldsymbol{\eta}_y$ as $\boldsymbol{\theta}_y$. Note that in general $\widetilde{d} \ll d$.

Thus, the effect of passing $\mathbb{G}$ through a shuffle channel is to change $\boldsymbol{\theta}$ such that graphs isomorphic to one another have identical probabilities, that is $\{\theta_i = \eta_j \, \forall i : G_i \in \widetilde{G}_j\}$. Prior to shuffling, there is no such constraint.

4 Shuffling Can Degrade Optimal Performance

The result of passing the graph through a shuffle channel can only degrade, but not improve, Bayes risk, as proven below.

Theorem 1. *Assuming* $(\mathbb{G}, Y) \sim \mathbb{P}_{\mathbb{G},Y}$, $L^* \leq \widetilde{L}^*$

Proof: Let $\mathbb{P}[\mathbb{G} = G|Y = y] = \mathbb{P}_y[\mathbb{G} = G] = \mathbb{P}_y[G]$, where $\mathbb{P}_y[G] \geq 0$ and $\sum_{G \in \mathcal{G}_n} \mathbb{P}_y[G] = 1$. Therefore:

$$\begin{aligned} L^* &= \mathbb{E}[h^*(\mathbb{G}) \neq Y] = \int_{\mathcal{G}_n} \mathbb{I}\{h^*(\mathbb{G}) \neq Y\} d\mathbb{P} \\ &= \sum_{G \in \mathcal{G}_n} \min_y \mathbb{P}_y[G] \end{aligned} \tag{4}$$

where $\mathbb{I}\{\cdot\}$ is the indicator function taking value unity whenever its argument is true and zero otherwise. When the graphs are shuffled, we have:

$$\begin{aligned} \widetilde{L}^* &= \sum_{G \in \mathcal{G}_n} \min_y \mathbb{P}_y[\sigma[G] = G] = \sum_{G \in \mathcal{G}_n} \min_y \mathbb{P}_y[\sigma[G] \in \widetilde{G}] \\ &= \sum_{\widetilde{G} \in \widetilde{\mathcal{G}}_n} \min_y \mathbb{P}_y[G \in \widetilde{G}] \end{aligned} \tag{5}$$

The result follows from the fact that $\min_y \mathbb{P}_y[G \in \widetilde{G}] \geq \min_y \mathbb{P}_y[\mathbb{G} = G]$ for all $G \in \mathcal{G}_n$. □

5 When Shuffling Necessarily Degrades Bayes Optimal Performance

The above proof demonstrates that shuffling can degrade Bayes risk, but not necessarily. A natural follow-up question is: "under what circumstances does shuffling degrade Bayes risk?" Below we prove that if the labels contain any class-conditional signal, then shuffling necessarily degrades Bayes risk. We write $\mathbb{P}_{\widetilde{\mathbb{G}}|Y} = \mathbb{P}_{\mathbb{G}|Y}$ if and only if $\mathbb{P}[G' \in \widetilde{G}|Y] = \mathbb{P}[\mathbb{G} = G'|Y]$ for all $G' \in \mathcal{G}_n$. In other words, $\mathbb{P}_{\widetilde{\mathbb{G}}|Y} = \mathbb{P}_{\mathbb{G}|Y}$ if and only if $\{\theta_i = \eta_j \, \forall i : G_i \in \widetilde{G}_j\}$.

When shuffling changes the distribution of graphs, such that it was the case that $\{\theta_i \neq \eta_j \, \forall i : G_i \in \widetilde{G}_j\}$ prior to shuffling, then shuffling necessarily degrades misclassification performance.

Theorem 2. *Assume* $\pi_y = 1/|\mathcal{Y}|$ *for all* $y \in \mathcal{Y}$ *without loss of generality. If* $\mathbb{P}_{\widetilde{\mathbb{G}}|Y} \neq \mathbb{P}_{\mathbb{G}|Y}$ *then* $L^* < \widetilde{L}^*$ *(note the strictly less than).*

Proof: If $\mathbb{P}_{\widetilde{\mathbb{G}}|Y} \neq \mathbb{P}_{\mathbb{G}|Y}$, then it must be the case that for at least one $G' \in \mathcal{G}_n$ such that $\mathbb{P}_y[\mathbb{G} = G'] \neq \mathbb{P}_y[G' \in \widetilde{\mathbb{G}}]$. Thus, for all $G' \in \widetilde{G}'$, $\min_y \mathbb{P}_y[G' \in \widetilde{G}] > \min_y \mathbb{P}_y[\mathbb{G} = G']$, which implies that:

$$L^* = \sum_{G \in \mathcal{G}_n} \min_y \mathbb{P}_y[G] < \sum_{\widetilde{G} \in \widetilde{\mathcal{G}}_n} \min_y \mathbb{P}_y[G \in \widetilde{G}] = \widetilde{L}^*. \tag{6}$$

□

6 Bayes Optimal Graph Classification After Shuffling

If the graph G has been passed through a shuffle channel, and one still desires to classify it, one might consider two complementary approaches. First, one might try to "unpermute" the graph, to recover the vertex labels, and then use a Bayes optimal graph classifier. Second, one might try to use a graph-invariant based classifier. A graph invariant is any function: $\psi : \mathcal{G}_n \mapsto \mathbb{R}^d$ such that $\psi(G) = \psi(\sigma(G))$ for all $\sigma \in \Sigma_n$ and $G \in \mathcal{G}_n$. A graph invariant based classifier first projects a graph into an invariant space and then classifies. The Bayes optimal graph invariant classifier minimizes risk over all invariants:

$$h^*_\psi = \operatorname*{argmin}_{h_\psi \in \mathcal{H}_\psi} \mathbb{E}[h_\psi(\mathbb{G}) \neq Y], \tag{7}$$

and L^*_ψ is the Bayes invariant risk.

Below we show two seemingly contradictory results. First, trying to recover the vertex labels is futile: it cannot be done. Second, it is optimal.

Theorem 3. *After passing a graph through a shuffle channel,* $\mathcal{C}(G) = \widetilde{G}$*, all graphs in* $\mathcal{G}$ *are equally likely to have given rise to* $\widetilde{G}$*.*

Proof: The number of different permutation functions $|\Sigma_n|$ is also the number of graphs within an isomorphism class, $|\widetilde{G}|$. A shuffle channel samples a permutation function uniformly at random from Σ_n. Thus, once G is shuffled, the probability of any graph G giving rise the the unlabeled graph $\mathcal{G}$ is $1/|\widetilde{G}|$. □

Thus, there is no hope to recover the original graph. Yet, once the graph has been shuffled, a Bayes optimal unlabeled graph classifier cannot be beat.

Theorem 4. *After passing a graph through a shuffle channel,* $L \geq \widetilde{L}^*$*.*

Proof: After passing a graph through a shuffle channel, the effect on the graph distribution is a normalizing of likelihoods. Specifically, $\mathbb{P}_y[\sigma(G)] = \mathbb{P}_y[G]$ for all $\sigma \in \Sigma_n$ and $G \in \mathcal{G}_n$. Thus, $\widetilde{\mathbb{P}}_y[G] \propto \mathbb{P}_y[G]$ and

$$\operatorname*{argmax}_{y\in\mathcal{Y}} \mathbb{P}[\widetilde{\mathbb{G}} = \widetilde{G}|Y = y] = \operatorname*{argmax}_{y\in\mathcal{Y}} \mathbb{P}[\mathbb{G} = G|Y = y] \quad (8)$$

□

In other words, $\widetilde{L}^* = L^*_{\psi}$, that is, the Bayes unlabeled graph classifier is the optimal invariant-based classifier.

7 A CONSISTENT AND EFFICIENT UNSHUFFLING-BASED CLASSIFIER

The above results show that after passing a graph through a shuffle channel, although nothing can be gained trying to unpermute the vertices, classifying the unlabeled graphs is optimal. Here, we show that we can induce a consistent and efficient classifier from training data, that is, a classifier can be estimated that is asymptotically guaranteed to be optimal.

Assume that a collection of n graph/class pairs are sampled independently and identically from some true but unknown distribution, $(\mathbb{G}_i, Y_i) \overset{iid}{\sim} \mathbb{P}_{\mathbb{G},Y} = \mathbb{P}_Y \pi_Y$, and that each graph has been passed through a shuffle channel. The *training data* is therefore $\mathcal{T}_n = \{(\widetilde{\mathbb{G}}_i, Y_i)_{i\in[n]}$. In a slight abuse of notation, an *induced unlabeled graph classifier*, $h : \widetilde{\mathcal{G}}_n \times (\widetilde{\mathcal{G}}_n \times \mathcal{Y})^n \mapsto \mathcal{Y}$ aims to estimate h^* from the training data. A Bayes plugin unlabeled graph-classifier estimates the likelihood and prior terms and plugs them in to Eq. (3)

$$\hat{h}_{BPI}[G] = \operatorname*{argmax}_{y\in\mathcal{Y}} \hat{\mathbb{P}}_{\widetilde{\mathbb{G}}|Y=y}[G]\hat{\pi}_y \quad (9)$$

Given consistent estimators for $\mathbb{P}_y[G]$ and π_Y, the Bayes plugin classifier is also consistent [2]. Formally, if $\hat{\mathbb{P}}_y[G] \to \mathbb{P}_y[G]$ and $\hat{\pi}_Y \to \pi_Y$ as $n \to \infty$, then $\hat{h}_{BPI} \to h^*$ as $n \to \infty$.

As described above, the class-conditional distributions of unlabeled graphs can be characterized as categorical distributions, $\mathbb{P}_{\widetilde{\mathbb{G}}|Y=y} = \text{Cat}(\boldsymbol{\eta}_y)$. Because a categorical distribution is in the exponential family, the maximum likelihood estimate for its parameters are exist, are unique, consistent, and efficient. Moreover, the class prior can also be represented as a categorical random-variable, and therefore has the same properties. Taken together, these results demonstrate that the Bayes plugin unlabeled graph classifier is consistent and efficient.

8 A PRACTICAL APPROACH TO UNLABELED GRAPH CLASSIFICATION

While the likelihood parameters are available, estimating them requires first solving a computationally hard problem. Specifically, to estimate each η_j one must first determine whether each additional graph is isomorphic to a previously observed graph. Graph isomorphism is not known to be in P or NP, which means there is no known algorithm for solving it in polynomial time (in the worst case). Thus, the above consistent result depends on solving an infinite number of NP problems! While this sounds bad, in practice, many graph isomorphism algorithms are available, including approximate ones [1].

To demonstrate the practically of an isomorphism-based unlabeled graph classifier, consider an independent-edge random graph model, where the class-conditional probability of an edge is given by $\mathbb{P}[A_{uv}|Y = y] = \mathbb{P}_{uv|y}$, yielding a likelihood factorization: $\mathbb{P}_y = \prod_{uv} \mathbb{P}_{uv|y}$. Thus, the class-conditional likelihood is given by a matrix, $\mathbb{P}_y \in (0, 1)^{n\times n}$. We sample $s + 1$ labeled graph/class pairs identically and independently from the above distribution (s training samples, and 1 test sample). Then, we pass each training graph through a shuffle channel, yielding $\{\widetilde{G}_i\}_{i\in[s]}$.

Our classifier proceeds as follows. Try to match each training graph to the test graph using some graph matching strategy, yielding a estimated labeled graph for each training graph, $\hat{G}_i = f(G, \widetilde{G}_i)$. We use a quadratic assignment problem (QAP) approximate algorithm as described in [3]. This is an approximate graph matching algorithm; while solving graph matching is NP-hard [1], this approach scales cubically in n. Nonetheless, we find it to be very effective.

Given the estimated labeled graphs, use $\{\hat{G}_i\}_{i\in[s]}$ to estimate the class conditional likelihood parameters using a robust L-estimator as described in [4], and obtain maximum likelihood estimates of the prior probabilities. Now, we can use a Bayes plugin classifier:

$$\hat{h}_{BPI}[G] = \operatorname*{argmax}_{y\in\mathcal{Y}} \prod_{uv} \hat{\mathbb{P}}_{uv|y}\hat{\pi}_y. \quad (10)$$

Figure 1 shows the performance of this Bayes plugin classifier as a function of the number of training samples. As we had hoped, performance monotonically increases towards optimal performance (gray dot), even though the graph matching algorithm we used was approximate, and only $\mathcal{O}(n^3)$ instead of exponential.

9 UNLABELED CONNECTOME CLASSIFICATION

Inspired by the simulated performance of our unlabeled graph classifier, we decided to try it on a real-world application. A "connectome" is a graph in which vertices correspond to biological neural units, and edges correspond to connections between the units. Diffusion Magnetic Resonance (MR) Imaging and related technologies are making the acquisition of MR connectomes routine [5]. 49 subjects from the Baltimore Longitudinal Study on Aging comprise this data, with acquisition and connectome inference details as reported in [6]. Each connectome yields a 70×70 element binary adjacency matrix. Associated with each graph is class label based on the gender of the individual (24 males, 25 females). Because the vertices are labeled, we can compare the results of having the labels and not having the labels. A k_n

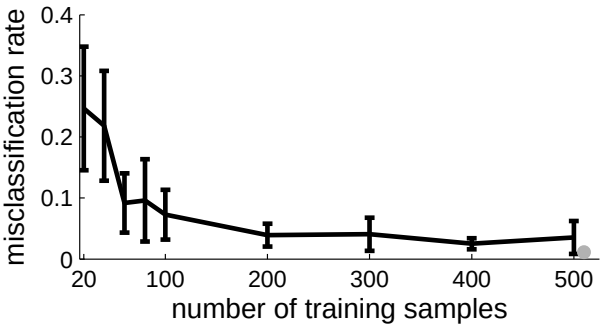

Fig. 1. Inexact graph matching can be used to approximate a consistent unlabeled graph classifier. Data in this simulation was sampled from the independent edge model described above, with $\mathbb{P}_{uv|0}$=Uniform$(0, 0.7)$, and $\mathbb{P}_{uv|1} = \mathbb{P}_{uv|0} + 0.3$ for all (u, v); $\pi_0 = \pi_1 = 0.5$. For each number of training samples, we tested using 5000 test samples, and repeated 10 times. The gray dot indicates Bayes optimal performance.

nearest neighbor ($k$nn) classifier is universally consistent, that is, guaranteed to achieve optimal performance in the limit [7], and therefore seems more appropriate than an independent edge model. Performance is evaluated with leave-one-out misclassification rate and reported in Table 1. When using the vertex labels, a standard $k$nn achieves 20% misclassification rate. Chance performance (only using the estimated prior) on this data is 49%. These two numbers provide bounds on performance. When all graphs are passed through a shuffle channel, we first try to unshuffle the graphs using the above mentioned QAP algorithm. Given the unshuffled graphs, performance changes to 45%, not particularly impressive. The performance of the independent edge model based Bayes plugin classifier for unlabeled graphs is similarly unimpressive. We therefore develop a hybrid approach in which the independent edge model is assumed, and parameters are estimated using the vertex labels. Given these estimates, we can use the QAP algorithm to match each test graph to the two likelihood matrices, and then use the Bayes plugin classifier. This approach yields a 31% misclassification rate. In contrast, a "standard" graph invariant based approach, which computes the graph invariants from [8], and plugs them into various machine learning algorithms (including the winner [9]), yields misclassification rates as low as 25%.

TABLE 1
MR Connectome Leave-One-Out Misclassification Rates

N/A-QAP	1-QAP	48-QAP	1NN-GI
20%	31%	45%	25%

10 Discussion

In this work, we have address both the theoretical and practical limitations of classifying graphs with and without including labels. Specifically, we show that shuffling the vertex labels results in an irretrievable situation, with a possible degradation of classification performance, and a necessary degradation if the vertex labels contained class-conditional signal. Moreover, although one cannot hope to recover the vertex labels, estimating them yields an asymptotically optimal classifier. This suggest that efforts to estimate the vertex labels may yield useful classification results, outperforming "standard" graph-invariant based classifiers. Via simulation we show that an approximate graph matching algorithm converges to the optimal performance with only about 500 training samples for a particular independent edge random graph model. Finally, we demonstrate with connectome data that estimating the vertex labels can be useful, but that there remains room to grow to exceed misclassification performance of a carefully chosen graph invariant $\circ$ machine learning based approach on this data. These connectome data, much like other collections of graphs, can also be equipped with both vertex and edge attributes. As such, we hope to extend the results herein to consider the more general cases.

Acknowledgments

References

[1] D. Conte, P. Foggia, C. Sansone, and M. Vento, "THIRTY YEARS OF GRAPH MATCHING," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 18, no. 3, pp. 265–298, 2004.
[2] L. Devroye, L. Gyorfi, and G. Lugosi, *A probabilistic theory of pattern recognition*. New York: Springer-Verlag, 1996.
[3] J. T. Vogelstein and Priebe, "conroy's QAP," *Submitted for publication*, 2011.
[4] J. T. Vogelstein and C. E. Priebe, "ind edge class paper," *Submitted for publication*, 2011.
[5] P. Hagmann, L. Cammoun, X. Gigandet, S. Gerhard, P. Ellen Grant, V. Wedeen, R. Meuli, J. P. Thiran, C. J. Honey, and O. Sporns, "MR connectomics: Principles and challenges," *J Neurosci Methods*, vol. 194, no. 1, pp. 34–45, 2010. [Online]. Available: http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve\&db=PubMed\&dopt=Citation\&list_uids=20096730
[6] W. R. Gray, J. T. Vogelstein, and R. J. Vogelstein, "Mr. Cap," *Submitted for publication*, 2011.
[7] C. J. Stone, "Consistent Nonparametric Regression," *The Annals of Statistics*, vol. 5, no. 4, pp. 595–620, Jul. 1977. [Online]. Available: http://projecteuclid.org/euclid.aos/1176343886
[8] H. Pao, G. A. Coppersmith, and C. E. Priebe, "Statistical Inference on Random Graphs : Comparative Power Analyses via Monte Carlo," 2010.
[9] K. Crammer, M. Dredze, and F. Pereira, "Exact Convex Confidence-Weighted Learning," *Journal of Multivariate Analysis*, vol. 99, no. 5, pp. 1–8, 2008. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0047259X07000784

Joshua T. Vogelstein Joshua T. Vogelstein is a spritely young man, engorphed in a novel post-buddhist metaphor.

Carey E. Priebe Buddha in training.