Graph Learning for Inverse Landscape Genetics

Prathamesh Dharangutte
Tandon School of Engineering
New York University
prathamesh.d@nyu.edu

Christopher Musco
Tandon School of Engineering
New York University
cmusco@nyu.edu

Abstract

The problem of inferring unknown graph edges from numerical data at a graph’s nodes appears in many forms across machine learning. We study a version of this problem that arises in the field of landscape genetics, where genetic similarity between populations of organisms living in a heterogeneous landscape is explained by a weighted graph that encodes the ease of dispersal through that landscape.

Our main contribution is an efficient algorithm for inverse landscape genetics, which is the task of inferring this graph from measurements of genetic similarity at different locations (graph nodes). We reduced the problem to that of inferring graph edges from noisy measurements of effective resistances between graph nodes, which have been observed to correlate well with genetic similarity.

Building on [15], we develop an efficient first-order optimization method for solving this problem. Despite its non-convex nature, extensive experiments on synthetic and real genetic data establish that our method provides fast and reliable convergence, significantly outperforming existing heuristics used in the field.

1 Introduction

Many datasets can be modeled as a weighted, undirected graph: $G = (V, E)$ with nodes $V = \{v_1, \ldots, v_n\}$ and additional numerical data vectors $x_1, \ldots, x_n \in \mathbb{R}^d$ at each node. For example, in social networks, each node is a user, each edge is a connection or interaction between users, and $x_i$ might contain demographic information about user $i$ like age, gender, or expressed political party.

Often, node data is correlated with $G$’s connectivity structure: if $v_i$ and $v_j$ are strongly connected, $x_i$ and $x_j$ tend to be more similar than for poorly connected nodes [20, 31]. Formally, connectivity between two nodes can be quantified in many of ways, from simple statistics like shortest path distance or number of common neighbors, to more advanced metrics like personalized PageRank [33, 19], SimRank [18], or DeepWalk distance [34]. While these measures depend solely on $G$’s structure (i.e. edges and their weights), they often align with measured similarities between $x_1, \ldots, x_n$.

This observation leads to an interesting possibility: even when edges in $G$ are unknown or partially observed, node data can be useful in inferring edges and weights, or at least in inferring a graph whose connectivity structure is consistent with the observed data. This possibility has been explored across statistics, machine learning, and network science [37, 35, 9, 22, 15]. In many cases, pairwise measures of connectivity can reveal a striking amount of information about $G$, and by proxy, so can similarity information between $x_1, \ldots, x_n$ [15]. Applications of graph inference from node data include understanding structured statistical correlation, link prediction, and phylogeny reconstruction.

In this work, we examine an application of graph inference that arises in landscape genetics, a field at the intersection of landscape ecology, spatial statistics, and population genetics [39]. We refer the reader to [24] for an introduction. Landscape genetics seeks to explain genetic differences between populations of the same species that live at different geographic locations. In particular, the goal is to understand how ease of movement between these geographic locations (i.e., through the landscape)
affects population genetics. Geographically isolated populations tend to have highly differentiated gene pools, whereas ease of travel and intermixing between populations leads to genetic similarity.

Early methods in landscape genetics correlate genetic similarity with simple measures of geographic isolation, like the Euclidean distance between populations [49, 42], or distance along an oriented direction or curve, leading to concepts like clines and ring species [10, 16]. These tried-and-true approaches have been successfully applied to understanding genetic variation in a variety of species, including humans [29]. More recently, however, work in landscape genetics considers finer-grained measures of landscape-driven isolation, largely based on modeling the landscape as an undirected graph (the landscape graph). Each location (spatial cell) in the landscape is associated with a graph node, and each node is connected by an edge to all geographically adjacent nodes (see Fig. 1).

Edge weights are chosen to reflect the ease of organism dispersal between adjacent nodes: we follow the convention that high weight indicates ease of dispersal and low weight indicates inhibition to movement, although note that the opposite meaning is sometimes used [7]. Weights are tailored to specific species: e.g., an edge across a span of water would have low weight for a ground-dwelling species which cannot easily traverse the edge. For an organism that prefers low-land environments, edges crossing areas of high elevation might receive lower weight than those crossing low-land areas.

In addition to the landscape graph, we consider numerical genetic data about populations of organisms living at different nodes of the graph. Usually this data is sparse, meaning we only have information for a subset of nodes [32]. Regardless, the goal is to correlate pairwise genetic similarity between these nodes with pairwise connectivity in the underlying landscape graph. For example, the weight of the least cost path between two nodes is commonly used as a connectivity measure, and shown to correlate with genetic similarity (measured using e.g., the fixation index) [2, 7, 47]. More recently, the influential paper Isolation by Resistance by McRae popularized the use of effective resistance distance as a connectivity measure in landscape genetics [26, 50]. Effective resistances better model organism dispersal, and thus correlate more closely with genetic differences across landscapes [27].

2 Our Contributions

So where does graph inference come in? Most studies that use landscape graphs to model species dispersal construct these graphs based on expert knowledge [26, 40]. Knowledge of a species' behavioral preferences (e.g., preferred elevation, vegetation cover, or climate) are used to determine edge weights, which are then used to compute pairwise connectivities like least cost paths or effective resistances. Multiple landscape graphs proposed by experts can be tested for fit [48, 23, 47, 41], but achieving high levels of correlation with genetic data ultimately requires significant background information on a species (which may be imperfect) and laborious hand-tuning of the landscape graph.

To address this issue, there has been interest in moving beyond expert opinion, by algorithmically determining optimal edge weights [51, 36]. Specifically, the goal is to learn a function that maps measurable landscape parameters for each edge (e.g., what vegetation cover it goes through, or if there is human development along the edge) to edge weights. The resulting weighted graph should have connectivity structure that correlates as well as possible with genetic differences across the landscape.
We call this parameterized graph inference problem inverse landscape genetics. Not only does this exciting problem offer the possibility of refining expert-designed landscape graphs, but a solution would allow ecologists to infer information about species dispersal based purely on collected genetic data [32], as opposed to the traditional perspective of explaining genetic data with known ecological knowledge. Genetic information could be used to understand species habitat preferences, find bottlenecks in migration, or understand if human development is impeding species movement [25]. However, despite interest in the inverse landscape genetics problem, few effective algorithms have been developed for its solution. Most current approaches optimize landscape graphs (i.e. find a graph most consistent with observed genetic data) using variants of brute force search. For example, a common approach is to rely on expert opinion to obtain an initial graph and then search over a small set of nearby weight functions to obtain an improved fit [40]. We refer the reader to [51] for a survey of existing techniques. There has been some work on more systematic algorithms. Peterman et al. [35] introduce a framework for optimizing landscape graphs using a genetic algorithm and compare their method with other approaches [36]. Graves et al. develop an approach based on local search heuristics, using Nelder-Mead and Newton line search algorithms to optimize landscape graphs [13].

A differentiable approach. Our main contribution is to show that one of the most common formalizations of the inverse landscape genetics problem can be solved efficiently and reliably using gradient based optimization methods. In particular, we consider a version of the problem which correlates the effective resistance between two nodes (a measure of graph connectivity) with the fixation index between genetic data at those nodes (a measure of genetic differentiation). We build on recent work of Hoskins et al. that studies the problem of learning graph edges based on noisy measurements of effective resistances in the graph [15]. As in that result, we show how to compute a gradient for an appropriately chosen graph-learning loss involving the effective resistances, and in our case, fixation index values. To do so, we need to differentiate through the effective resistances computation, which involves the pseudoinverse of a graph Laplacian. We implement this step efficiently using an iterative linear system solver for positive semidefinite matrices. Our approach is detailed in Section 3.

To the best of our knowledge, our method is the first for the inverse landscape genetics problem that uses a gradient based optimization method. In Section 4 we compare it against local search heuristics used in prior work [13], showing that it obtains much more reliable convergence on both synthetic and real-world data sets. As an application of our fast algorithm, we are able to explore questions of statistical complexity that have been raised in the landscape genetics literature [32]. In particular, there are concerns that algorithmic methods might overfit the landscape graph if learned using genetic data from an insufficient number of nodes. By varying the amount of data available in a sequence of large synthetic data experiments, we empirically explore the precise number of samples required to obtain a generalizing solution, showing that in some cases, as few as 25 populations are needed to reliably fit the parameters of a landscape graph involving 1000s of nodes.

Additional related work. Relevant work on landscape genetics is included in the previous sections. We discuss additional related work on graph learning to Appendix C in the supplemental material.

3 Proposed Method

We first describe notation needed to formalize the inverse landscape genetics problem from Section 2.

Graph and Genetic Data Notation. We denote the weighted, undirected landscape graph by $G = (V, E, w)$, where $V = \{v_1, \ldots, v_n\}$ is the vertex set, $E$ is the edge set, and $w$ is a vector of weights assigned to each edge. Let $m$ denote $m = |E|$. Typically $m \ll \binom{n}{2}$ since for most landscapes $G$ will be a grid graph with $m = O(n)$. We index both $E$ and $w$ by their terminal nodes: edges are $e_{i_1j_1}, \ldots, e_{i_mj_m}$ and weights are $w_{i_1j_1}, \ldots, w_{i_mj_m}$. It is often helpful to view graphs as electrical networks where $e_{ij}$ represents an electrical connection with conductance $w_{ij}$ between nodes $v_i$ and $v_j$ [44]. Let $r_{ij} = 1/w_{ij}$ denote the resistance of the connection.

For a subset $S \subseteq V$ of nodes we have measured vectors of population genetic data $x_{1}, \ldots, x_{|S|} \in \mathbb{R}^d$. We only interact with this data through a black-box measure of genetic dissimilarity: the specific choice is not important. In keeping with prior work, our experiments use the fixation index, typically denoted $F_{ST}$. For two populations, $i$ and $j$ a high $F_{ST}$ (close to 1) indicates greater difference between
the measured genetic information in $x_i$ and $x_j$. Let $F \in \mathbb{R}^{[S] \times [S]}$ contain pairwise $F_{ST}$ (or another dissimilarity) for all nodes in $S$. Let $F_{i,i} = 0$ for all diagonal entries.

It has been established that the values in $F$ will correlate well with the effective resistances of an appropriately chosen landscape graph $G$ \cite{26}. To define these measures, let $D \in \mathbb{R}^{n \times n}_+$ be the diagonal degree matrix with $D_{i,i} = \sum_{j \neq i} D_{ij}$. Let $A$ be the adjacency matrix with $A_{ij} = A_{ji} = w_{ij}$ for all $c_{ij} \in E$, and 0 otherwise. Let $\bar{L}$ be the weighted graph Laplacian as $D - A$.

**Definition 1 (Effective resistance).** The effective resistance $R_{ij}$ between two nodes $i$ and $j$ satisfies

$$R_{ij} = b^T_i \bar{L}^+ b^T_j$$

where $\bar{L}^+$ is the Moore-Penrose pseudoinverse of the graph laplacian $\bar{L}$ and $b_{ij} \in \mathbb{R}^n$ is the vector with a 1 at position $i$, a $-1$ at position $j$ and 0’s elsewhere.

The effective resistance between two nodes $v_i$ and $v_j$ is lower when there exist more low-resistance paths (i.e., high weight paths) between $v_i$ and $v_j$. It is known to be equal to the commute time between $v_i$ and $v_j$ for a random walk with steps taken proportional to edges weights \cite{6}, which gives some intuition for why the measure effectively quantifies organism dispersal through a landscape. We refer the reader to \cite{25} for further discussion of the important of effective resistances in landscape ecology.

Let $R$ be the matrix of all pairwise effective resistances and note that $R_{ij} = R_{ji}$, which can be thought of as the resistance surface for the landscape. $R$ is 0 along its diagonal. Let $R_S \in [S] \times [S]$ be the principal submatrix of $R$ containing only the pairwise effective resistances between nodes in $S$. The main problem we study is as follows:

**Problem 1 (Inverse Landscape Genetics).** Given landscape graph nodes $V$ and edges $E$ we are given a vector of environmental parameters $C_{i,j,k} \in \mathbb{R}^p$ for each $e_{i,j,k} \in E$ and a function class $P$ from $\mathbb{R}^p \rightarrow \mathbb{R}_+$ which maps these parameters to a weight for each edge. Assume $P$ in parameterized by parameters $\theta$ and denote functions in the class by $p_\theta \in P$. For $p_\theta$, let $p_\theta(E) = \{p_\theta(C_{i1,j1}),\ldots,p_\theta(C_{im,jm})\}$. Our goal is to find $\theta^*$ minimizing the loss:

$$\theta^* = \arg \min_{\theta} \mathcal{L}(\theta) = \arg \min_{\theta} \|R_S(p_\theta(E)) - F\|_F^2,$$

where $R_S(p_\theta(E))$ is the effective resistance matrix for the graph $G = (V, E, p_\theta(E))$ (restricted to nodes in $S$). $\|A\|_F^2 = \sum_i \sum_j A_{ij}^2$ denotes the standard Frobenius norm.

Note that both $R_S(p_\theta(E))$ and $F$ have zeros on the diagonal, so the Frobenius norm above is equal to $2 \times$ the standard squared loss between effective resistances and genetic dissimilarities. Other natural choices could be used instead of $\mathcal{L}$, e.g. the inverse of the Mantel correlation between $R_S(p_\theta(E))$ and $F$ \cite{13}. In either case, the goal is to find edge weights such that the landscape graph $G$ produces effective resistances between nodes in $S$ which are as close as possible to the measure genetic dissimilarities in $F$. Alternatively, under the assumption that genetic dissimilarities represent noisy measurements of the true effective resistances for some unknown landscape graph $G^*$, then Problem 1 can be viewed as the task of recovering that graph.

### 3.1 Example functional forms

The problem is stated under the constraint that weights in the learned graph are a function $p_\theta$ of $q$ environmental parameters $C_{i,j,k}$ about each edge. This function can take any form: we only require that it is differentiable with respect to its parameters. For example, prior work often considers $C_{i,j,k}$ which is a single continuous scale parameter like edge elevation or temperature. A typical choice (see e.g. \cite{13}) is to assume that $1/w_{i,j,k} = r_{i,j,k}$ follows an inverted Gaussian relation governed by parameters $\theta = [\beta, \beta_{opt}, \beta_{SD}]$:

$$\frac{1}{w_{i,j,k}} = r_{i,j,k} = \beta + 1 - \beta \exp \left(\frac{-(C_{i,j,k} - \beta_{opt})^2}{2\beta_{SD}^2}\right)$$

(2)

This form captures the fact that many species have for example a preferred “ideal” elevation $\beta_{opt}$ and are more likely to travel along edges of similar elevation: the resistance to dispersal $r_{i,j,k}$ increases as $C_{i,j,k}$ moves further from $\beta_{opt}$. Other papers consider slightly different functions, but they typically have the same general structure as \cite{2} \cite{35}.


Another common functional form is linear. We simply let:

$$\frac{1}{w_{ik,jk}} = r_{ik,jk} = \alpha^T C_{ik,jk}. \tag{3}$$

For instance $C_{ik,jk}$ might be a vector of one-hot-encoded categorical data indicating what landcover an edge traverses (e.g., water, forest, marshland, tundra). Each entry in $\alpha$ is a scalar associated with each category type that conveys how permeable the category is for movement. When continuous and discrete data at nodes is considered in unison, it is natural to add multiple functional forms linearly:

$$r_{ik,jk} = r_{ik,jk}^E + r_{ik,jk}^L C_{ik,jk} \text{ where } r_{ik,jk}^E \text{ is an elevation term in the form of } \theta^2, \text{ and } r_{ik,jk}^L C_{ik,jk} \text{ is a landcover term in the form of } \theta^2.$$

### 3.2 Gradient computation

Due to its non-convex nature, there is no closed form solution for $L(\theta)$. The cornerstone of our approach is to instead find approximate solution by using projected gradient descent to minimize $L(\theta)$. To do so, we need an efficient method for computing the gradient of this loss.

**Proposition 1.** Let $n_\theta$ denote the number of parameters in $\theta$ (typically a small constant) and let $J \in \mathbb{R}^{n \times n_\theta}$ denote the Jacobian with $J_{k,h} = \frac{\partial w_{ik,jk}}{\partial \theta_h}$. Let $B \in \mathbb{R}^{n \times n}$ denote the edge-vertex incidence matrix of $G$ with $k^{th}$ row equal to $e_i - e_j$, where $i_k$ and $j_k$ are the terminal nodes of $G$’s $k^{th}$ edge.

$$\nabla_\theta L = \sum_{v_i, v_j \in S} (F_{lk} - R(u^\theta)_{lk}) \cdot 2J \cdot (b^T_{lk} L_\theta^2 B)^{\circ 2},$$

where $\circ^2$ denote the Hadamard power (i.e., square every vector element entrywise) and $L_\theta$ denotes the Laplacian of the landscape graph with edge weights $p_\theta(E)$.\footnote{\textcolor{red}{[15]}}

**Proof.** For given parameters $\theta$, let $w^0 = p_\theta(E)$, where $p_\theta(E)$ is as defined in Proposition 1. We have:

$$\nabla_{w^\theta} L = -2 \sum_{v_i, v_k \in S} (F_{lk} - R(w^\theta)_{lk}) \cdot \nabla_{w^\theta} R(w^\theta)_{lk} \tag{4}$$

As in Problem 1, $R(w^\theta)$ is the matrix of all pairwise effective resistances for the graph $G = (V, E, w^\theta)$. From the definition for effective resistance, $R(w^\theta)_{lk} = b^T_{lk} L_\theta^2 b_{lk}$. As in $\cite{15}$, we can obtain a partial derivative for entries of $L^+$ with respect to $w^\theta$ via the Sherman-Morrison formula for rank one updates to the pseudoinverse. Specifically, we have $\frac{\partial L^+_{\theta}}{\partial w^\theta_{ij}} = -L_\theta^{-1} b_{ij} b_{ij}^T L_\theta^+ \text{ and thus}$

$$\frac{\partial R(u^\theta)_{lk}}{\partial w^\theta_{ij}} = -b^T_{lk} \left( L_\theta^+ b_{ij} b_{ij}^T L_\theta^+ \right) b_{lk} = -(b^T_{ij} L_\theta^+ b_{lk})^2$$

It follows that $\nabla_{w^\theta} R(u^\theta)_{lk} = -(b^T_{ij} L_\theta^+ B)^{\circ 2}$. The proposition follows from plugging this equation into (1) and noting that $\nabla_{\theta} L = J^T \nabla_{w^\theta} L$. \hfill \Box

**Efficient computation of the gradient.** Proposition 1 yields an efficient algorithm for computing $\nabla_{\theta} L$. In particular, since $n_\theta$ is typically a small constant computing the Jacobian $J$ is efficient for any differentiable functional form $p_\theta$. Then, ignoring the cost of computing $b^T_{lk} L_\theta^+ b_{lk}$ for all $v_i, v_k \in S$, the gradient can be computed in $O(|S|^2 \cdot m \cdot n_\theta)$ time. Note that since every row in $B$ is 2 sparse, $b^T_{ij} L^+ B$ can be computed in $O(m)$ time once $b^T_{ij} L^+$ is computed. Since $m = O(n)$ in most landscape genetics applications, the bottle neck is therefore computing each $L_\theta^+ b_{lk}$.

This would naively require inverting the $n \times n$ Laplacian $L_\theta$, which would be computationally intensive and impractical for large graphs. We instead approximate the matrix-vector product $L_\theta^+ b_{lk}$ using an iterative solve for positive definite linear systems ($L$ is positive definite). In our experiments we use the standard MINRES method. To optimize the approach further, we note that $L_\theta^+ b_{lk} = L(w)^+ e_l + L_\theta^+ e_k$ where $e_l$ and $e_k$ are the $l^{th}$ and $k^{th}$ standard basis vectors. Accordingly, we only need to solve $|S|$ linear systems (either $e_l$ as the right hand side for all $v_i \in S$), and can then recombining those solutions to return all $\binom{|S|}{2}$ vectors $L(w)^+ b_{lk}$ needed for the gradient computation.
4 Empirical Results

With an efficient gradient oracle in hand for the loss function in Problem 1 we test a gradient based optimization approach on both synthetic and real genetic data. Real genetic data is obtained for the North American wolverine (Gulo gulo) from [21], which provides \(^{ST}F\) values for 6 populations living across a region in Alaska. Our goal is to understand the interplay between genetic variation in this region and the underlying landscape. Specifically, we obtain elevation data [30] and land cover data [14], which will be used as the basis for selecting edge weights in a landscape graph.

The landscape graph graph is constructed by dividing the Alaska region into a grid of square cells. In previous landscape genetics studies of the North American wolverine, cell sizes of 5 km and 50 km have been used [27]. We choose a resolution of 15 km, which lead to a graph \(G = (V, E)\) with \(|V| = 24035\) and \(|E| = 47746\). For each cell we create a node in the grid graph, and connect adjacent nodes with edges (as in Figure 1). Our landscape data comes as raster images, with each pixel corresponding to a region of 100 × 100 meters for elevation data and 30 × 30 meters for landcover data, so we have multiple pixels of information within each landscape cell. This data was resampled to cell resolution using standard GIS methods (see Appendix A for details).

Continuous and discrete environmental parameters are then collected for each edge in the graph. For edge \(k\), edge elevation \(C_{ik,jk}\) is taken as the average elevation at cells \(i\) and \(j\) and scaled to lie within the range 0-10. For each edge we also construct a vector of one-hot-encoded landcover data \(\mathbf{C}_{ik,jk}\), which has 17 entries for landcover types like evergreen forest, barren land, or open water. Each entry in \(\mathbf{C}_{ik,jk}\) is given values as follows: 0 if the landcover type is absent at cell \(i\) and \(j\), 0.5 if present at either cell \(i\) or \(j\), or 1 if present at both cells \(i\) and \(j\). We model edge weights as a function of these parameters by linearly combining equation (2) for elevation data and (3) for landcover data. So, the final parameter vector we hope to learn when solving Problem 1 is \(\theta = \{\beta, \beta_{SD}, \alpha \in \mathbb{R}^{17}\}\).

To minimize 1, we implement a projected gradient descent method with RMSProp step size adjustment, which adjusts learning rate by a decaying average of squared gradients [46]. Since edge weights are constrained to be non-negative, and all edge data is non-negative, we project parameters to \(\max(0, \theta)\) at each gradient step. All experiments were run on server with 2vCPU @2.2GHz and 13 GB main memory.

**Synthetic data:** Our first set of data experiments uses the real landscape data from Alaska, but in conjunction with carefully simulated genetic data, which makes it possible to better assess the performance of our method. Specifically, we selected a random 50 × 50 subgrid of our Alaska graph to obtain a grid graph with \(|V| = 2500\) and \(|E| = 4900\). We then constructed a ground truth graph by randomly sampling a set of parameters, \(\theta_{true}\), and evaluating the weights for all edges in \(E\). The goal in our synthetic experiments is to recover this ground truth, which is a common set up in testing algorithms for inverse landscape genetics as real ground truth data is never available [13].

In particular, we construct the pairwise effective resistance matrix \(R_{S}(w^{true})\) for a set of nodes \(S\) with \(N = |S| \ll |V|\). For the nodes in \(S\), we produce a simulated genetic similarity matrix \(F\) by setting \(F_{ik} = \left[R_{S}(w^{true})\right]_{ik} + \tilde{z}\) where \(\tilde{z} \sim N(0, \tilde{\sigma})\). We run experiments with \(\tilde{\sigma} = \{0, 0.05\mu, 0.2\mu\}\), where \(\mu\) is the mean of the resistances in \(R_{S}(w^{true})\). These cases (no, low, and high noise) range

\[0.00\quad 0.05\quad 0.10\quad 0.15\quad 0.20\]

\[0.00\quad 0.05\quad 0.10\quad 0.15\quad 0.20\]

\(\text{Number of nodes sampled } |S|\)

\[25\quad 50\quad 100\quad 150\]

Figure 2: Relative error between recovered parameters and true parameters for synthetic data experiments with different numbers of nodes sampled \(N\) and noise standard deviation \(\tilde{\sigma}\). Parameter recovery improves with more samples (i.e., more locations with genetic similarity data), and generally with less noise (i.e., more highly correlated resistance and genetic data).

---

1In fact, we found that projecting to \(\max(\epsilon, \theta)\) where \(\epsilon \leq 1\) was helpful in preventing the method from getting stuck at local minima. This choice also ensures non-zero resistance value for all landcover types, which is a constraint often imposed in prior work.
We conclude that, as $N$ increases, our method obtains high quality approximations to the true parameters $\theta_{\text{true}}$, even in the high noise regime. For example, $N = 150$ was sufficient for fitting the graph parameters in all cases. This is a pretty typical number of samples for a landscape genetics study (e.g. [40] obtains genetic data for mountain goats from $N = 149$ locations over a comparably sized area). Accordingly, even for a reasonably large number of landscape parameters, reliable learning of landscape data should be possible with existing data collection methods.

It has been noted in the literature that a potential concern with optimizing landscape graphs is overfitting when $N$ is small, i.e., the landscape graph fit to $F$ does not generalize to new data [52]. To validate our method against overfitting, we randomly split nodes into sets $S_{\text{train}}$ and $S_{\text{test}}$. We learn parameters $\theta$ for nodes in $S_{\text{train}}$ and evaluate these parameters against pairwise effective resistances in $S_{\text{test}}$. Even in the high noise setting, with $\bar{\sigma} = 0.2\mu$, test loss converges along with train loss when $N$ is as low as 25, (see Fig. 3). This implies good generalization and a lack of overfitting, even though we do not accurately recover all parameters in $\theta_{\text{true}}$. This is not necessarily surprisingly: it indicates that, while the inverse landscape genetics problem may be poorly conditioned with respect to $\theta$ (a fact observed in [13]) it is still possible to obtain reliable predictive models with little data.

### Comparison with existing approaches
We compare gradient-based optimization to the Nelder-Mead method, which has been used in prior work on inverse landscape genetics [13]. We observe that our method is faster in terms of convergence and also better at recovering true parameters with enough data. Nelder-Mead eventually achieves comparable performance in terms of train loss but fails at recovering the true parameters (Figure 5). To ensure a fair comparison, we choose the same random initialization of parameters and non-negativity constraints.

### North American wolverine (gulo gulo)
For experiments on real-world data, the $F_{\text{ST}}$ values range from 0 to 1 and we have access to genetic data at 15 nodes out of 24035 nodes. After fitting $\theta$ with our gradient based method, we compute the $R^2$ value for a linear fit between recovered resistances and $F_{\text{ST}}$ values, a metric used

---

Note that Nelder-Mead is an unconstrained optimization method, so we add a projection step to ensure interpretable parameters are found. This does not noticeably effect the behavior of convergence in our experiments.
(a) Relative loss vs. iteration for Nelder-Mead and gradient-based optimization. After 5000 iterations, the loss value is 0.193 for Nelder-Mead and 0.19 for gradient-based optimization.

(b) Relative parameter error between recovered parameters and true parameters with iteration. Gradient-based optimization is better at recovering true parameters.

Figure 5: Comparison of proposed method to a heuristic optimization technique. Gradient-based optimization is faster in convergence and better at recovering true parameters with enough data. Experiments are for synthetic data with high noise setting with $N = 150$ and $\tilde{\sigma} = 0.2 \mu$.

Figure 6: $R^2$ values were computed for a linear fit between $F_{ST}$ and effective resistances in the final learnt landscape graphs. Gradient-based optimization obtains a slightly better fit.

in prior work [27]. We obtain an $R^2$ value of 0.7396 using gradient-based optimization and 0.6203 using Nelder-mead after learning, in comparison to 0.68 (5km resolution) and 0.71 (50km resolution) obtained by [27] using expert opinions. Note that [27] uses a binary map as habitat/nonhabitat for underlying landscape with 12 populations whereas we use a multivariate surface with continuous and discrete data with 6 populations. We provide the final parameters $\theta$ in Table 1. The solutions for Nelder-Mead optimization and our gradient descent method largely agree: landcover types that allow for movement under cover (i.e. forests are assigned low resistances values) and open water is assigned the highest resistance value. There is a notable difference between learned parameters for barren land, and sedge/herbaceous landscape, which would be interesting to explore further.

| Parameter         | Nelder Mead | Gradient based optimization |
|-------------------|-------------|-----------------------------|
| $\beta$           | 0           | 0                           |
| $\beta_{opt}$     | 10          | 9                           |
| $\beta_{SD}$      | 0           | 0                           |
| Open water        | 227         | 502                         |
| Barren Land       | 151         | 5                           |
| Deciduous forest  | 0           | 0                           |

Table 1: Final parameter values after optimization, round to nearest whole number. We do not report for landcover types which were present at less that 2% of nodes in the graph.
Conclusion

We believe this work has high potential for positive broader impacts related to environmental protection and conversation. Landscape ecology and landscape genetics are important in understanding and discovering impediments to species dispersal which may threaten biodiversity or long-term species survival [1,8]. Effective resistance-based landscape ecology has been especially important in understanding the effects of climate change on species dispersal and migration [25]. We refer the reader to https://circuitscape.org/pubs.html for more information. By providing researchers working in landscape genetics with a powerful, general purpose algorithmic tool, we hope our work will have a positive impact on accelerating their research.

We also hope this paper helps highlight an interesting applications of graph-inference that we believe is not well known to the machine learning community. Storfer et al. emphasize the need for forming bridges between research areas with different technical expertise in order to move landscape genetics forward [45]. Already there has been successful cross-field collaboration between ecologists and those working in spatial statistics. We hope to bring the machine learning community into the fold.

With those benefits in mind, our work does have some potential for negative impact. In particular, the graph inference problem studied has potential applications to de-anonymizing edges in social networks [15,22]. Our contributions would probably have limited impact on this sort of application (since our methods are developed specifically for parameterized, planar graphs used in modeling landscapes). Nevertheless, continued work in the area could have negative privacy implications.

Acknowledgments

The authors would like to thank Cameron Musco and Charalampos E. Tsourakakis for early discussions about this work, as well as Uthsav Chitra who provided assistance in formalizing the inverse landscape genetics problem studied.

References

[1] R Ament, R Callahan, M McClure, M Reuling, and G Tabor. Wildlife connectivity: Fundamentals for conservation action. Center for Large Landscape Conservation, Bozeman, Montana, 2014.

[2] Jean-François Arnaud. Metapopulation genetic structure and migration pathways in the land snail helix aspersa: influence of landscape heterogeneity. Landscape Ecology, 18(3):333–346, 2003.

[3] Hagai Attias. A variational bayesian framework for graphical models. In Advances in neural information processing systems, pages 209–215, 2000.

[4] Vladimir Batagelj, Tomaž Pisanski, and JMS Sim [otilde] es Pereira. An algorithm for tree-realizability of distance matrices. International Journal of Computer Mathematics, 34(3-4):171–176, 1990.

[5] Tony Cai, Weidong Liu, and Xi Luo. A constrained \( \ell_1 \) minimization approach to sparse precision matrix estimation. Journal of the American Statistical Association, 106(494):594–607, 2011.

[6] Ashok K Chandra, Prabhakar Raghavan, Walter L Ruzzo, Roman Smolensky, and Prasoon Tiwari. The electrical resistance of a graph captures its commute and cover times. Computational Complexity, 6(4):312–340, 1996.

[7] A Coulon, JF Cosson, JM Angibault, B Cargnelutti, M Galan, N Morellet, E Petit, Stéphane Aulagnier, and AJM Hewison. Landscape connectivity influences gene flow in a roe deer population inhabiting a fragmented landscape: an individual–based approach. Molecular ecology, 13(9):2841–2850, 2004.

[8] Brett G Dickson, Christine M Albano, Ranjan Anantharaman, Paul Beier, Joe Fargione, Tabitha A Graves, Miranda E Gray, Kimberly R Hall, Josh J Lawler, Paul B Leonard, et al. Circuit-theory applications to connectivity science and conservation. Conservation Biology, 33(2):239–249, 2019.
[9] Hilmi E Egilmez, Eduardo Pavez, and Antonio Ortega. Graph learning from data under laplacian and structural constraints. *IEEE Journal of Selected Topics in Signal Processing*, 11(6):825–841, 2017.

[10] John A Endler. *Geographic variation, speciation, and clines*. Princeton University Press, 1977.

[11] Joseph Felsenstein. Confidence limits on phylogenies: an approach using the bootstrap. *Evolution*, 39(4):783–791, 1985.

[12] Joseph Felsenstein and Joseph Felsenstein. *Inferring phylogenies*, volume 2. Sinauer associates Sunderland, MA, 2004.

[13] Tabitha A Graves, Paul Beier, and J Andrew Royle. Current approaches using genetic distances produce poor estimates of landscape resistance to interindividual dispersal. *Molecular ecology*, 22(15):3888–3903, 2013.

[14] Collin Homer, Jon Dewitz, Suming Jin, George Xian, Catherine Costello, Patrick Danielson, Leila Gass, Michelle Funk, James Wickham, Stephen Stehman, et al. Conterminous united states land cover change patterns 2001–2016 from the 2016 national land cover database. *ISPRS Journal of Photogrammetry and Remote Sensing*, 162:184–199, 2020.

[15] Jeremy Hoskins, Cameron Musco, Christopher Musco, and Babis Tsourakakis. Inferring networks from random walk-based node similarities. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 3704–3715. Curran Associates, Inc., 2018.

[16] Richard John Huggett. *Fundamentals of biogeography*. Routledge, 2004.

[17] Arun Jambulapati, Kirankumar Shiragur, and Aaron Sidford. Efficient structured matrix recovery and nearly-linear time algorithms for solving inverse symmetric $m$-matrices. *arXiv preprint arXiv:1812.06295*, 2018.

[18] Glen Jeh and Jennifer Widom. Simrank: a measure of structural-context similarity. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 538–543, 2002.

[19] Glen Jeh and Jennifer Widom. Scaling personalized web search. In *Proceedings of the 12th international conference on World Wide Web*, pages 271–279, 2003.

[20] Vassilis Kalofolias. How to learn a graph from smooth signals. In *Artificial Intelligence and Statistics*, pages 920–929, 2016.

[21] CJ Kyle and C Strobeck. Genetic structure of north american wolverine (gulo gulo) populations. *Molecular Ecology*, 10(2):337–347, 2001.

[22] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031, 2007.

[23] N Lugon-Moulin and J Hausser. Phylogeographical structure, postglacial recolonization and barriers to gene flow in the distinctive valais chromosome race of the common shrew (sorex araneus). *Molecular ecology*, 11(4):785–794, 2002.

[24] Stéphanie Manel, Michael K Schwartz, Gordon Luikart, and Pierre Taberlet. Landscape genetics: combining landscape ecology and population genetics. *Trends in ecology & evolution*, 18(4):189–197, 2003.

[25] BH McRae, Viral Shah, and Alan Edelman. Circuitscape: modeling landscape connectivity to promote conservation and human health. *The Nature Conservancy*, 14, 2016.

[26] Brad H McRae. Isolation by resistance. *Evolution*, 60(8):1551–1561, 2006.

[27] Brad H McRae and Paul Beier. Circuit theory predicts gene flow in plant and animal populations. *Proceedings of the National Academy of Sciences*, 104(50):19885–19890, 2007.
[28] Karthik Mohan, Mike Chung, Seungyeop Han, Daniela Witten, Su-In Lee, and Maryam Fazel. Structured learning of gaussian graphical models. In Advances in neural information processing systems, pages 620–628, 2012.

[29] John Novembre, Toby Johnson, Katarzyna Bryc, Zoltán Kutalik, Adam R Boyko, Adam Auton, Amit Indap, Karen S King, Sven Bergmann, Matthew R Nelson, et al. Genes mirror geography within europe. Nature, 456(7218):98–101, 2008.

[30] National Atlas of the United States. 100-meter resolution elevation of alaska, albers projection. http://purl.stanford.edu/sg962yb7367, 2012.

[31] Antonio Ortega, Pascal Frossard, Jelena Kovačević, José MF Moura, and Pierre Vanderheynst. Graph signal processing: Overview, challenges, and applications. Proceedings of the IEEE, 106(5):808–828, 2018.

[32] Sara J Oyler-McCance, Bradley C Fedy, and Erin L Landguth. Sample design effects in landscape genetics. Conservation Genetics, 14(2):275–285, 2013.

[33] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[34] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 701–710, 2014.

[35] William E Peterman. ResistanceGA: An r package for the optimization of resistance surfaces using genetic algorithms. Methods in Ecology and Evolution, 9(6):1638–1647, 2018.

[36] William E Peterman, Kristopher J Winiarski, Chloe E Moore, Carolina da Silva Carvalho, Anthony L Gilbert, and Stephen F Spear. A comparison of popular approaches to optimize landscape resistance surfaces. Landscape Ecology, 34(9):2197–2208, 2019.

[37] Garvesh Raskutti, Bin Yu, Martin J Wainwright, and Pradeep K Ravikumar. Model selection in gaussian graphical models: High-dimensional consistency of \( \ell_1 \)-regularized MLE. In Advances in Neural Information Processing Systems, pages 1329–1336, 2009.

[38] Lev Reyzin and Nikhil Srivastava. Learning and verifying graphs using queries with a focus on edge counting. In International Conference on Algorithmic Learning Theory, pages 285–297. Springer, 2007.

[39] James Sanderson. Landscape ecology: a top down approach. CRC Press, 2020.

[40] AJ Shirk, DO Wallin, SA Cushman, CG Rice, and KI Warheit. Inferring landscape effects on gene flow: a new model selection framework. Molecular ecology, 19(17):3603–3619, 2010.

[41] RA Short Bull, SA Cushman, R Mace, T Chilton, KC Kendall, EL Landguth, MK Schwartz, Kevin McKelvey, Fred W Allendorf, and G Luikart. Why replication is important in landscape genetics: American black bear in the rocky mountains. Molecular ecology, 20(6):1092–1107, 2011.

[42] Robert R Sokal and Neal L Oden. Spatial autocorrelation in biology: 2, some biological implications and four applications of evolutionary and ecological interest. Biological Journal of the Linnean Society, 10(2):229–249, 1978.

[43] Daniel A Spielman. Trees and distances. University Lecture, 2012.

[44] Daniel A Spielman and Nikhil Srivastava. Graph sparsification by effective resistances. SIAM Journal on Computing, 40(6):1913–1926, 2011.

[45] A Storfer, MA Murphy, JS Evans, CS Goldberg, S Robinson, SF Spear, R Dezzani, E Delmelle, L Vierling, and LP Waits. Putting the ‘landscape’ in landscape genetics. Heredity, 98(3):128–142, 2007.
[46] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

[47] Sacha N Vignieri. Streams over mountains: influence of riparian connectivity on gene flow in the pacific jumping mouse (zapus trinotatus). *Molecular Ecology*, 14(7):1925–1937, 2005.

[48] Claire C Vos, AG Antonisse-De Jong, PW Goedhart, and MJM Smulders. Genetic similarity as a measure for connectivity between fragmented populations of the moor frog (rana arvalis). *Heredity*, 86(5):598–608, 2001.

[49] Sewall Wright. Isolation by distance. *Genetics*, 28(2):114, 1943.

[50] Luh Yen, Francois Fouss, Christine Decaestecker, Pascal Francq, and Marco Saerens. Graph nodes clustering based on the commute-time kernel. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 1037–1045. Springer, 2007.

[51] Katherine A Zeller, Kevin McGarigal, and Andrew R Whiteley. Estimating landscape resistance to movement: a review. *Landscape ecology*, 27(6):777–797, 2012.
A Experimental Details

The resolution for elevation data is 100 x 100 meters and that of landcover data is 30 x 30 meters. To have a reasonably sized graph, we change the resolution and convert pixels to cell corresponding to 15km resolution, compared to previously used 5km and 50km [27], using the re-sampling tool from ArcMap with the ‘nearest’ technique for land cover data and ‘bilinear’ technique for elevation data.

For experiments with real-world genetic data, we set the edge weight to 0 if any node for the edge had landcover type as ‘unclassified’ whereas for synthetic data we set corresponding α parameter to high value which results in minuscule edge weight. For optimization, we initialize β as 1 and all other parameters (θ_true) randomly from a discrete uniform distribution, as a small initial value for β provided us with most consistent results. For setting the true parameters θ_true and initialization, we sample β_opt and β_SD from range 0-10 and α from range 0-100 for synthetic data experiments. For experiments with real-world data, we initialize β_opt and β_SD similarly but for alpha we use the range 1-10. We run all experiments with RMSProp for 5000 iterations using learning rate of 0.1 and 0.9 for 15km resolution, compared to previously used 5km and 50km [27], using the re-sampling tool from ArcMap with the ‘nearest’ technique for land cover data and ‘bilinear’ technique for elevation data.

For experiments with real-world genetic data, we set the edge weight to 0 if any node for the edge had landcover type as ‘unclassified’ whereas for synthetic data we set corresponding α parameter to high value which results in minuscule edge weight. For optimization, we initialize β as 1 and all other parameters (θ_true) randomly from a discrete uniform distribution, as a small initial value for β provided us with most consistent results. For setting the true parameters θ_true and initialization, we sample β_opt and β_SD from range 0-10 and α from range 0-100 for synthetic data experiments. For experiments with real-world data, we initialize β_opt and β_SD similarly but for alpha we use the range 1-10. We run all experiments with RMSProp for 5000 iterations using learning rate of 0.1 and 0.9 for 15km resolution, compared to previously used 5km and 50km [27], using the re-sampling tool from ArcMap with the ‘nearest’ technique for land cover data and ‘bilinear’ technique for elevation data.

For optimization, we initialize β as 1 and all other parameters (θ_true) randomly from a discrete uniform distribution, as a small initial value for β provided us with most consistent results. For setting the true parameters θ_true and initialization, we sample β_opt and β_SD from range 0-10 and α from range 0-100 for synthetic data experiments. For experiments with real-world data, we initialize β_opt and β_SD similarly but for alpha we use the range 1-10. We run all experiments with RMSProp for 5000 iterations using learning rate of 0.1 and 0.9 for 15km resolution, compared to previously used 5km and 50km [27], using the re-sampling tool from ArcMap with the ‘nearest’ technique for land cover data and ‘bilinear’ technique for elevation data.

For experiments with real-world genetic data, we set the edge weight to 0 if any node for the edge had landcover type as ‘unclassified’ whereas for synthetic data we set corresponding α parameter to high value which results in minuscule edge weight. For optimization, we initialize β as 1 and all other parameters (θ_true) randomly from a discrete uniform distribution, as a small initial value for β provided us with most consistent results. For setting the true parameters θ_true and initialization, we sample β_opt and β_SD from range 0-10 and α from range 0-100 for synthetic data experiments. For experiments with real-world data, we initialize β_opt and β_SD similarly but for alpha we use the range 1-10. We run all experiments with RMSProp for 5000 iterations using learning rate of 0.1 and 0.9 for 15km resolution, compared to previously used 5km and 50km [27], using the re-sampling tool from ArcMap with the ‘nearest’ technique for land cover data and ‘bilinear’ technique for elevation data.

For experiments with real-world genetic data, we set the edge weight to 0 if any node for the edge had landcover type as ‘unclassified’ whereas for synthetic data we set corresponding α parameter to high value which results in minuscule edge weight. For optimization, we initialize β as 1 and all other parameters (θ_true) randomly from a discrete uniform distribution, as a small initial value for β provided us with most consistent results. For setting the true parameters θ_true and initialization, we sample β_opt and β_SD from range 0-10 and α from range 0-100 for synthetic data experiments. For experiments with real-world data, we initialize β_opt and β_SD similarly but for alpha we use the range 1-10. We run all experiments with RMSProp for 5000 iterations using learning rate of 0.1 and 0.9 for 15km resolution, compared to previously used 5km and 50km [27], using the re-sampling tool from ArcMap with the ‘nearest’ technique for land cover data and ‘bilinear’ technique for elevation data.

Note that for reporting R^2 values for linear fit between pairwise effective resistances of learnt landscape graph and F_st values, [27] use 1 - F_st instead of F_st values whereas we use F_st, but as F_st << 1 for the data under consideration, the R^2 values are approximately equal.

B Additional Experiments

B.1 Synthetic data experiments

We present results for experiments where we consider landscape data to be comprised only of elevation data (continuous surface) or landcover data (discrete surface). We note that when only elevation data is considered, we consistently obtain good recovery for β_opt and β_SD but not for β. Although the recovery is not consistent across all parameters, β_opt and β_SD are typically more meaningful to researchers, indicating the preferred elevation range and range of elevation of the species. Similar to optimizing for landscape comprised of both continuous and discrete data, we conclude that as the number of nodes sampled(N) increases, we obtain good approximations to the true parameters θ_true (refer Appendix Figure[7]).

[These are the recommended techniques from the software documentation. We refer the reader to https://desktop.arcgis.com/en/arcmap/10.3/tools/data-management-toolbox/resample.htm]
We compare Nelder-mead without projection and gradient-based optimization for learning the parameters for landscape graph. We observe that the gradient-based approach obtains better approximation to true parameters \( \theta \) (refer Appendix Figure 12).

### B.3 Comparison with existing approaches

We compare Nelder-mead without projection and gradient-based optimization for learning the parameters for landscape graph. We observe that the gradient-based approach obtains better approximation to true parameters \( \theta_{\text{true}} \) (refer Appendix Figure 12).

### C Additional Related Work

Relevant related work on landscape genetics is discussed in the introduction. Here we add important comments on additionally related work in graph learning. In particular, in Section 3 we frame the inverse landscape genetics problem as a problem of learning edge weights in a graph from (noisy) measurements of the effective resistances. This problem was directly addressed in [15], which our paper builds on. It has also been studied elsewhere. For example, it is well known that you can recover a graph exactly if you know effective resistances between all pairs of nodes.
value of $N$ descent iteration, for various values of noise standard deviation $\sigma$. Higher value of $N$ provides better recovery.

Figure 9: Relative error between recovered parameters and true parameters for continuous data with gradient descent iteration, for various values of noise standard deviation $\sigma$, for landscape graph with continuous data. Higher value of $N$ provides better recovery.

(a) Landscape with continuous data and $N = 25$.
(b) Landscape with continuous data and $N = 50$.
(c) Landscape with continuous data and $N = 100$.
(d) Landscape with continuous data and $N = 150$.

(a) Landscape with discrete data and $N = 25$.
(b) Landscape with discrete data and $N = 50$.
(c) Landscape with discrete data and $N = 100$.

Figure 10: Relative error between recovered parameters and true parameters for discrete data with gradient descent iteration, for various values of noise standard deviation $\sigma$, for landscape graph with discrete data. Higher value of $N$ provides better recovery and we observe overfitting for high-noise with $N = 25$. 

15
Applications include the reconstruction of phylogenetic trees in genetics [11, 12]. This can be done in polynomial time [43]: access to all effective resistances allows you to reconstruct like graph sparsity [37, 5]. Our work makes a structural assumption that the graph underlying our many samples are needed to learn the structure of this graph, often under additional assumptions is related to the adjacency or Laplacian matrix of an unknown graph. Several work explore how processing [9, 31]. The common assumption is that the correlation matrix between data at each node has been studied in different formulations across machine learning, statistics, and graph signal processing [9][31]. The common assumption is that the correlation matrix between data at each node is related to the adjacency or Laplacian matrix of an unknown graph. Several work explore how many samples are needed to learn the structure of this graph, often under additional assumptions like graph sparsity [37][5]. Our work makes a structural assumption that the graph underlying our

This can be done in polynomial time [43]: access to all effective resistances allows you to reconstruct the pseudoinverse of the graph Laplacian, which can then be inverted using a generic $O(n^3)$ time method, or more efficient algorithms [17]. Unfortunately, when only a subset of effective resistances are known, no polynomial time algorithm is known for recovering a graph consistent with those measurements. However, as observed in [15] and this paper (where we study a somewhat different parameterized problem) graph recovery can be framed as an optimization problem and solved to a global optimal with first order methods, despite inherent non-convexity. Recovering edge information from effective resistances has also been studied for the special case of tree graphs. In a tree, the effective resistance is the inverse of the shortest path distance between nodes $i$ and $j$. There has been a lot of interest in reconstructing trees from actual and partial measurements of these distances [38][4]. Applications include the reconstruction of phylogenetic trees in genetics [11][12].

Finally, we note that our problem is related to that of inferring graphical models [3][28], which has been studied in different formulations across machine learning, statistics, and graph signal processing [9][31]. The common assumption is that the correlation matrix between data at each node is related to the adjacency or Laplacian matrix of an unknown graph. Several work explore how many samples are needed to learn the structure of this graph, often under additional assumptions like graph sparsity [37][5]. Our work makes a structural assumption that the graph underlying our
data has both a simple edge structure (i.e., its a grid graph) and that edges weights are functions of relatively low-dimensional edge data (i.e., landscape information). An interesting direction for future work is theoretically exploring the implications of these strong assumptions on bounding the sample complexity of the inverse landscape genetics problem.