Published in final edited form as:
Stat Anal Data Min. 2012 December ; 5(6): 509–522. doi:10.1002/sam.11163.

Predicting Simulation Parameters of Biological Systems Using a Gaussian Process Model

Xiangxin Zhu[1], **Max Welling**[1], **Fang Jin**[2], and **John Lowengrub**[2]

[1]Department of Computing Science, University of California Irvine, Irvine, USA

[2]Department of Mathematics, University of California Irvine, Irvine, USA

Abstract

Finding optimal parameters for simulating biological systems is usually a very difficult and expensive task in systems biology. Brute force searching is infeasible in practice because of the huge (often infinite) search space. In this article, we propose predicting the parameters efficiently by learning the relationship between system outputs and parameters using regression. However, the conventional parametric regression models suffer from two issues, thus are not applicable to this problem. First, restricting the regression function as a certain fixed type (e.g. linear, polynomial, etc.) introduces too strong assumptions that reduce the model flexibility. Second, conventional regression models fail to take into account the fact that a fixed parameter value may correspond to multiple different outputs due to the stochastic nature of most biological simulations, and the existence of a potentially large number of other factors that affect the simulation outputs. We propose a novel approach based on a Gaussian process model that addresses the two issues jointly. We apply our approach to a tumor vessel growth model and the feedback Wright–Fisher model. The experimental results show that our method can predict the parameter values of both of the two models with high accuracy.

Keywords

Gaussian process; regression; biological simulation

1. INTRODUCTION

Systems biology is a rapidly growing research field that aims to understand and model the interactions between the components of biological systems. Their approach often involves the development of mechanistic and probabilistic models, control theory, and simulations. However, because of the large number of parameters, variables and constraints in biology systems, it is usually very difficult to find the optimal parameter values directly.

In a tumor growth model (we will describe the details later), for example, one key parameter, the *diffusion constant*, plays a crucial role in controlling the growth speed and the structure of vessel networks in tumors. How can we efficiently find the optimal value for it that generates a given simulation result? One possible way could be, rather than doing brute force search, finding the relationship between the diffusion constant on the one hand and the simulation outputs on the other, using regression. We can then predict the optimal parameter value using the system's observed outputs as input to the learned regression function.

Correspondence to: Xiangxin Zhu (xzhu@ics.uci.edu).

However, the conventional parametric regression model, $y = f(x, \theta) + \varepsilon$, is not suitable in this case, and the reason is twofold. First, usually little is known about what is the correct form of the function f that describes the relation between the simulation parameters and the system outputs. It is not reasonable to restrict too much the form of functions that we consider. If we are using a model based on a certain class of functions (e.g. linear functions) and the target function cannot be well modeled by this class, then the prediction accuracy will be poor. Second, in biological system simulation, a fixed parameter value may generate multiple different simulation outputs because of the stochastic nature of most biological systems, and the existence of a potentially large number of other parameters which may vary from one simulation to another. Because of this reason, the data oftentimes have a special plateau-like structure (see Fig. 1). Unfortunately, the conventional regression models are not able to take into account the fact that a set of inputs to the regression model $\{x_i\}$ can correspond to a single target y. Any conventional regression model would make different predictions for those different input features, and not use the (known) information that they really correspond to the same target. For example, as we show in Fig. 2, if we fit a linear function to the observed data (black dots), and use this learned linear function (blue line) to make predictions on the new data (red dots), we will predict different values at each of new input (red circles on the blue line indicate the predictions). It is clear that the linear regression function fails to model the fact that all the new inputs actually correspond to a same target. The exact same problem exists for other conventional regression models.

We note that these two issues are very general ones, and not just limited to the two biological models we use to demonstrate our approach in this article. We believe that they are ubiquitous in biological simulation and other sciences. In this work, we propose a novel model based on the Gaussian process that is able to address the aforementioned two problems.

As such, we believe our model may find broad applicability in the biology community.

We apply our approach on two models: a tumor vessel network growth model and the feedback Wright–Fisher model for reproduction of cells. The experimental results show that our approach generates accurate predictions on both the two models, and outperforms several commonly used regression methods.

The rest of this article is organized as follows: In Section 2, we briefly introduce the Gaussian process. In Section 3 we propose our novel regression model. We briefly introduce the two simulation models in Section 4 and describe the features we use for regression in Section 5. The detailed experiments and results are presented in Section 6. We conclude our work in Section 7.

2. GAUSSIAN PROCESSES

A Gaussian process (GP) [1] is a probability distribution over functions $\mathbf{f}(\cdot)$. A GP is specified by a mean function $\mathbf{u}(\cdot)$ and a covariance kernel $K(\cdot, \cdot)$ which are modeled parametrically. Once these are given we can compute the joint probability distribution over any subset of function values (say a pair of points) as follows:

$$\begin{bmatrix} \mathbf{f}(\mathbf{x}) \\ \mathbf{f}(\mathbf{x}') \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{u}(\mathbf{x}) \\ \mathbf{u}(\mathbf{x}') \end{bmatrix}, \begin{bmatrix} K(\mathbf{x}, \mathbf{x}) & K(\mathbf{x}, \mathbf{x}') \\ K(\mathbf{x}', \mathbf{x}) & K(\mathbf{x}', \mathbf{x}') \end{bmatrix} \right). \quad (1)$$

Hence all finite dimensional marginal distributions over subsets of function values are Gaussian distributed. Moreover, the covariance kernel is constructed such that points that are further away from each other are less correlated while points close together are strongly and positively correlated. This ensures that the functions we consider are smooth at small distance scales. The mean function is used to bias the functions to the type of functions one expects to encounter *a priori*.

While a GP is a prior specification of what functions we expect, the data will transform that into a *posterior* distribution over functions that agree with the evidence provided by the data. Note moreover that the GP also quantifies the *uncertainty* over functions consistent with the data. In other words, it tells us not only the most likely regression curve but also a one standard deviation uncertainty band within which the real function may be found. This is clearly a desirable property.

To compute the posterior probability given data we split our points into training points $\{\mathbf{x}_i, \mathbf{y}_i\}$ and testing point $\mathbf{x}_*$, where $\mathbf{y}_i$ is the observed function value subject to noise corruption, $\mathbf{y}_i = \mathbf{f}(\mathbf{x}_i) + \epsilon, \epsilon \sim \mathcal{N}(0, \sigma^2 I)$. The joint distribution over training and testing points is then,

$$\begin{bmatrix} \mathbf{f} \\ f_* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{u} \\ u_* \end{bmatrix}, \begin{bmatrix} K & K_*^{\mathrm{T}} \\ K_* & K_{**} \end{bmatrix} \right), \quad (2)$$

while $\mathbf{y}|\mathbf{f} \sim \mathcal{N}(0, \sigma^2 I)$.

Using Bayes rule, we can then compute the posterior of the unseen test case given the observed data. The posterior can be written in closed-form:

$$p(f_*|\mathbf{y}, \mathbf{x}, \mathbf{x}_*) = \mathcal{N}\left(u_* + K_*\left[K + \sigma^2 I\right]^{-1}(\mathbf{y} - \mathbf{u}), K_{**} - K_*\left[K + \sigma^2 I\right]^{-1} K_*^{\mathrm{T}}\right). \quad (3)$$

A GP is a nonparametric model, which means we do not restrict ourselves to a specific form or parametrized family of functions. Instead our ‘inductive bias’ is expressed by stating something about the smoothness of the functions we like to admit. This means our inductive bias is weak, or in other words: ‘we let the data speak’. However, too much flexibility in a model class means that we may easily overfit to the noise of the data. In a GP one is protected against overfitting because the parameters in the mean function and the covariance kernel are not estimated but integrated over. Thus, there is really no fitting of parameters at all. There is no free lunch of course, and too large a model class may simply lead to very large uncertainties in ones posterior predictions. Thus, we see that more inductive bias will allow us to learn more from fewer data points but if our inductive bias is wrong then we may bias our answer in the wrong direction. We express our inductive bias by (i) choosing a GP in the first place, (ii) choosing a mean function and covariance kernel, and (iii) placing priors over the hyperparameters that govern the mean and covariance functions.

3. GAUSSIAN PROCESS WITH MULTIPLE INPUTS PER TARGET

As we alluded to before, the situation we face when estimating the parameters from multiple stochastic simulations is that we now have potentially many inputs corresponding a single target. In this section, we propose a modified Gaussian process model with multiple inputs per target to address this issue.

3.1. Modeling Multiple Inputs

We will say that our data comes in groups $\mathcal{D} = \{(X_i, y_i)\}_{i=1}^{c}$ where c is the number of the data groups and $X_i = \left[\mathbf{x}_\mathbf{i}^1, \ldots, \mathbf{x}_\mathbf{i}^\mathbf{j}, \ldots, \mathbf{x}_\mathbf{i}^{\mathbf{N_i}}\right]$ is the set of inputs corresponding to group i. Also N_i is the number of training samples in ith group and y_i is the regression target of the ith group.

Finally, let $\mathscr{T} = \{X_*\}$ be the testing inputs, where $X_* = \left[\mathbf{x}_*^1, \ldots, \mathbf{x}_*^{\mathbf{k}}, \ldots, \mathbf{x}_*^{\mathbf{N}_*}\right]$. Similarly, N_* is the number of testing inputs. Assuming that there exists an underlying intrinsic ‘center’ for each group, the hidden variables $\mathscr{Z} = \left\{\{\mathbf{z_i}\}_{i=1}^{c}, \mathbf{z}_*\right\}$ are introduced to represent these ‘centers’. The probability density over the samples in each group is then modeled by a Gaussian distribution centered at these hidden variable $\mathbf{z_i}$:

$$\mathbf{x_i^j} \sim \mathscr{N}(\mathbf{z_i}, \Sigma_i) \qquad \forall j, \quad (4)$$

$$\mathbf{x}_*^{\mathbf{k}} \sim \mathscr{N}(\mathbf{z}_*, \Sigma_*) \qquad \forall k. \quad (5)$$

Assuming the samples in each group are I.I.D. (independently and identically distributed), we thus have:

$$p(X_i|\mathbf{z_i}, \Sigma_i) = \prod_j p\left(\mathbf{x_i^j}|\mathbf{z_i}, \Sigma_i\right), \quad (6)$$

$$p(X_*|\mathbf{z}_*, \Sigma_*) = \prod_k p\left(\mathbf{x}_*^{\mathbf{k}}|\mathbf{z}_*, \Sigma_*\right). \quad (7)$$

3.2. Gaussian Process on the Hidden Variables

We assume that the value of the noisy regression targets y_i depend on the hidden variables as follows: $y_i = f(\mathbf{z}_i) + \epsilon, \epsilon \sim \mathscr{N}\left(0, \sigma_n^2\right)$. The function **f** is modeled as a Gaussian process (see Section 2),

$$p\left(\{f_i\}_{i=1}^{c}, f_* | \{\mathbf{z_i}\}_{i=1}^{c}, \mathbf{z}_*\right) = \mathscr{N}\left(\mathbf{u}, \begin{bmatrix} K & K_*^{\mathrm{T}} \\ K_* & K_{**} \end{bmatrix}\right), \quad (8)$$

where $\mathbf{u} = [u_1, \ldots, u_c, u_*]$ is a vector that denotes the *mean function*. We model this mean function as follows:

$$u_i = \exp\left(\mathbf{w}^{\mathrm{T}} \begin{bmatrix} \mathbf{z_i} \\ 1 \end{bmatrix}\right). \quad (9)$$

where **w** is a $(d + 1) \times 1$ vector of parameters. A ‘1’ is appended to **z** to model an overall scaling factor.

Furthermore, $\begin{bmatrix} K & K_*^{\mathrm{T}} \\ K_* & K_{**} \end{bmatrix}$ is the *covariance matrix*, whose elements are defined as:

$$K = \begin{bmatrix} k(\mathbf{z}_1, \mathbf{z}_1) & k(\mathbf{z}_1, \mathbf{z}_2) & \cdots & k(\mathbf{z}_1, \mathbf{z_c}) \\ k(\mathbf{z}_2, \mathbf{z}_1) & k(\mathbf{z}_2, \mathbf{z_z}) & \cdots & k(\mathbf{z}_2, \mathbf{z_c}) \\ \vdots & \vdots & \ddots & \vdots \\ k(\mathbf{z_c}, \mathbf{z}_1) & k(\mathbf{z_c}, \mathbf{z}_2) & \cdots & k(\mathbf{z_c}, \mathbf{z_c}) \end{bmatrix}_{c \times c}, \quad (10)$$

$$K_* = [k(\mathbf{z}_*, \mathbf{z}_1), k(\mathbf{z}_*, \mathbf{z}_2), \cdots, k(\mathbf{z}_*, \mathbf{z_c})]_{1 \times c}, \quad (11)$$

$$K_{**}=k(\mathbf{z}_*,\mathbf{z}_*), \quad (12)$$

where $k(\cdot,\cdot)$ is the *covariance kernel function*. In this article, we have adopted the Matérn covariance function with noise, which is defined as:

$$k(\mathbf{z_a},\mathbf{z_b})=\sigma_0^2\left(1+\frac{\sqrt{3}\|\mathbf{z_a}-\mathbf{z_b}\|}{l}\right)\exp\left(-\frac{\sqrt{3}\|\mathbf{z_a}-\mathbf{z_b}\|}{l}\right). \quad (13)$$

Putting everything together the joint probability of the training data $\mathscr{D}=\{(X_i,y_i)\}_{i=1}^{c}$, the testing inputs $\mathscr{T}=\{X_*\}$, the hidden variables $\mathscr{Z}=\left\{\{\mathbf{z_i}\}_{i=1}^{c},\mathbf{z}_*\right\}$, the models parameters Θ = { $\psi_{\mathbf{K}}$ $\mathbf{w}$, σ_n, Σ_i, Σ_*}, the hidden variables $\mathscr{F}=\left\{\{f_i\}_{i=1}^{c}\right\}$ and the prediction target f_* can be written as:

$$\begin{aligned}&p(\mathscr{D},\mathscr{T},\mathscr{Z},\Theta,\mathscr{F},f_*)\\&=\left[\prod_i p(X_i|\mathbf{z_i},\Sigma_i)\,p(\mathbf{z_i})\,p(\Sigma_i)\right]p(X_*|\mathbf{z}_*,\Sigma_*)\,p(\mathbf{z}_*)\,p(\Sigma_*)\times p\left(\{f_i\}_{i=1}^{c},f_*|\{\mathbf{z_i}\}_{i=1}^{c},\mathbf{z}_*\right)\left[\prod_i p(y_i|f_i)\right]p(\mathbf{w})\,p(\psi_{\mathbf{K}})\,p(\sigma_n).\end{aligned} \quad (14)$$

In Appendix A we provide more details about the model parameters Θ = { $\psi_{\mathbf{K}}$, $\mathbf{w}$, σ_n, Σ_i, Σ_*} and the priors we used for them. The graphical representation of our model is given in Fig. 3.

3.3. Regression

Our goal is to compute the probability distribution for the variable f_*. Starting from the joint distribution Eq. (14) we find,

$$\begin{aligned}p(f_*|\mathscr{D},\mathscr{T})&=\int\int\int_{\mathscr{Z},\Theta,\mathscr{F}}p(f_*,\mathscr{Z},\Theta,\mathscr{F}|\mathscr{D},\mathscr{T})\ \mathrm{d}\mathscr{Z}\ \mathrm{d}\Theta\ \mathrm{d}\mathscr{F}\\&=\iint_{\mathscr{Z},\Theta}p(f_*|\mathscr{Z},\Theta,\mathscr{D},\mathscr{T})\,p(\mathscr{Z},\Theta|\mathscr{D},\mathscr{T})\ \mathrm{d}\mathscr{Z}\ \mathrm{d}\Theta.\end{aligned} \quad (15)$$

The integral of Eq. (15) is composed of two terms. The first term $p(f_*|\mathscr{Z},\Theta,\mathscr{D},\mathscr{T})$ is the posterior of a standard Gaussian process that can be computed using Eq. (3). The second term $p(\mathscr{Z},\Theta|\mathscr{D},\mathscr{T})$, which is the posterior of the hidden variables and the model parameters given the observed data and testing inputs, has a very complicated form and thus can not be calculated analytically. Therefore computing the exact value of Eq. (15) is nontrivial.

However, the integration of Eq. (15) can be approximated by sampling from $p(\mathscr{Z},\Theta|\mathscr{D},\mathscr{T})$:

$$\begin{aligned}p(f_*|\mathscr{D},\mathscr{T})&=\iint_{\mathscr{Z},\Theta}p(f_*|\mathscr{Z},\Theta,\mathscr{D},\mathscr{T})\,p(\mathscr{Z},\Theta|\mathscr{D},\mathscr{T})\ \mathrm{d}\mathscr{Z}\ \mathrm{d}\Theta\\&\approx\frac{1}{N}\sum_{s=1}^{N}p\left(f_*|\mathscr{Z}^{(s)},\Theta^{(s)},\mathscr{D},\mathscr{T}\right),\end{aligned} \quad (16)$$

where $\mathscr{Z}^{(s)}$ and $\Theta^{(s)}$ is a sample drawn from the posterior distribution $p(\mathscr{Z},\Theta|\mathscr{D},\mathscr{T})$. When N is large enough, It is guaranteed [2] that:

$$\frac{1}{N}\sum_{s=1}^{N}p\left(f_*|\mathscr{Z}^{(s)},\Theta^{(s)},\mathscr{D},\mathscr{T}\right)\xrightarrow{N\to\infty}p(f_*|\mathscr{D},\mathscr{T}). \quad (17)$$

3.4. Inference Using Hybrid Monte Carlo

Hybrid Monte Carlo [3,4] is a tool to draw samples efficiently from a distribution $p(\mathbf{x})$ if it is differentiable and strictly positive everywhere. It incorporates information about the gradient of the target distribution. The main idea is that we simulate according to Hamiltonian dynamics with randomly drawn momentum variables, where the Hamiltonian is defined as $H(\mathbf{x}, \mathbf{p}) = \frac{1}{2}\mathbf{p}^{\mathrm{T}}\mathbf{p} - \log p(\mathbf{x})$.

$$\dot{\mathbf{x}} = \frac{\partial H(\mathbf{x}, \mathbf{p})}{\partial \mathbf{p}} = \mathbf{p}, \qquad \dot{\mathbf{p}} = -\frac{\partial H(\mathbf{p}, \mathbf{x})}{\partial \mathbf{x}} = -\frac{\partial E(\mathbf{x})}{\partial \mathbf{x}}. \quad (18)$$

At every iteration we redraw the momentum variables from a standard normal distribution $\mathbf{p} \sim \mathcal{N}(0, I)$. Their actual values are discarded afterwards. The Hamiltonian dynamics is implemented numerically using a numerical integration scheme known as 'leapfrog steps'. The errors in this numerical integration can be corrected by using an additional accept/reject step at the end of each iteration. For further details we refer to the literature [5].

In our model, $p(\mathbf{x}) \to p(\mathscr{Z}, \Theta|\mathscr{D}, \mathscr{T}) \propto p(\mathscr{D}, \mathscr{T}, \mathscr{Z}, \Theta)$, where

$$\begin{aligned} & p(\mathscr{D}, \mathscr{T}, \mathscr{Z}, \Theta) \\ & \quad = \int_{\mathscr{F}} p(\mathscr{D}, \mathscr{T}, \mathscr{Z}, \Theta, \mathscr{F})\, \mathrm{d}\mathscr{F} = p(X_*|\mathbf{z}_*, \Sigma_*)\, p(\mathbf{z}_*)\, p(\Sigma_*) \prod_i [p(X_i|\mathbf{z_i}, \Sigma_i)\, p(\mathbf{z_i})\, p(\Sigma_i)] \\ & \quad \times p\left(\{y_i\}_{i=1}^{c} | \{\mathbf{z_i}\}_{i=1}^{c}\right) p(\mathbf{w})\, p(\psi_{\mathbf{K}})\, p(\sigma_n). \end{aligned} \quad (19)$$

Defining the negative logarithm of $p(\mathscr{D}, \mathscr{T}, \mathscr{Z}, \Theta)$ (i.e. $E(\mathbf{x})$ in Eq (18)) as

$$E = -\log p(\mathscr{D}, \mathscr{T}, \mathscr{Z}, \Theta), \quad (20)$$

we can compute the derivatives of E with respect to $\mathscr{Z}$ and Θ ($\mathscr{Z}$ and Θ correspond to $\mathbf{x}$ in Eq. (18)), and generate sequence of samples $\mathscr{Z}^{(s)}$ and $\Theta^{(s)}$ using the HMC method. The derivation of these derivatives are put in Appendix B.

4. SIMULATIONS

In this section, we describe the two biological simulation systems we apply our approach to: a tumor vessel network growth model and a feedback Wright–Fisher model for reproduction of cells.

4.1. Tumor Vessel Network Growth Model

The development of a tumor-induced neovasculature network is modeled using a lattice-free, discrete framework developed in ref. 6 together with several modifications that are described below. The angiogenesis model generates a vascular network regulated by tumor angiogenic factors (TAF), e.g. [7]. Here, we model TAF using a continuum variable that describes the net effect of pro-angiogenic regulators. The concentration of TAFs, denoted by c, is governed by the diffusion-reaction equation,

$$0 = D_c \nabla^2 c - \beta_d c + S_c \phi_h (c_{sat} - c), \quad (21)$$

where the diffusion constant D_c is the key parameter in the model we would like to predict using regression. Refer to Appendix C for more details of this equation.

The new capillaries form randomly at sprouts near the tumor boundary following the concentration of TAFs. Vessels are described in terms of the trajectories taken by migrating endothelial cells [8]. A stochastic equation is prescribed for the leading endothelial cell at the vessel tip that describes the motion as a biased random walk:

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t}=s\mathbf{e}+\mathbf{v}_{\text{random}}. \quad (22)$$

This is a stochastic model of the chemotaxis of tip endothelial cells up gradients of TAFs. Find more details in Appendix C.

While there are many parameters in the model (see Table 7 in Appendix C), we focus here on the effect of the TAF diffusion coefficient D_c on the developing neovasculature network and the resulting tumor progression. This models the variable solubility of TAF isoforms.

In particular, it is found that the more soluble isoforms lead to a more disorganized and less functional vessel network than the more insoluble isoforms.

We performed many simulations with TAF diffusion coefficient D_c ranging from 20 to 1, where we kept all other parameters unchanged but varied the initial tumor shape. In particular, the initial tumor shape is taken as a small random perturbation of a unit sphere. A sample of results are shown in Fig. 4. The results are quantified in Fig. 5, where the tumor volumes (a), the vessel lengths (b) and the ratio of the vessel, and tumor volumes (c) are shown. Note that the vessel volume is obtained by assuming that the vessel network is a collection of cylindrical vessel segments, with a radius of 0.05 in nondimensional length (approximately 10 μm in dimensional length).

4.2. Feedback Wright–Fisher Model

The Wright–Fisher model [9] [10] is one the most popular stochastic models for reproduction in population genetics. We have three types of cells in the population, stem cell (SC), transit-amplifying cell (TAC), and terminal differentiated cell (TDC). Denote the number of each type of cells by x_0, x_1, and x_2, respectively. We use feedback Wright–Fisher model to simulate how the population of each kind of cell grows. A parameter k in this model is varied to generate different trajectories.

We use our model to predict k based on the observed x_0, x_1 and x_2. The model is described as follows: The feedback act on p_0 as

$$p_0=\frac{1}{1+kx_2/N}. \quad (23)$$

Suppose we start at a vector (x_0, x_1, x_2) at time $t = n$, the *proportion* of SC, TAC and TDC in the next generation at time $t = n + 1$ will be

$$x_0p_0, \quad x_0(1-p_0)+x_1p_1, \quad x_1(1-p_1). \quad (24)$$

To generate $x_0^{'}, x_1^{'}, x_2^{'}$ we distribute N cells into three groups according to the above ratio. This is done by first generating a binomial random variable

$$x_0^{'}=\mathscr{B}(N,q), \quad (25)$$

with $q = x_0p_0/(x_0 + x_1)$. Then generate another binomial random variable

$$x_1' = \mathscr{B}\left(N - x_0', \widehat{q}\right), \quad (26)$$

with $\widehat{q} = \frac{x_0(1-p_0)+x_1 p_1}{(x_0+x_1)(1-q)}$. And finally $x_2' = N - x_0' - x_1'$. If we repeat this process we get a trajectory of cell populations, as shown in Fig. 6.

In our simulations, k is varied from 2 to 5. N is fixed to be 2000. p_1 is fixed to be 0.1.

5. FEATURES

In this section, we describe which features we extracted from a simulation which acted as the inputs (covariates) for our regression model (i.e. they will represent the input vector X_i in the joint model in Eq (14).

5.1. Tumor Vessel Network Growth Model

Tortuosity—Tortuosity is a property of a curve being tortuous (twisted; having many turns). Tortuosity of blood vessels is known to be used as a medical sign [11]. There have been several attempts to quantify this property [12] [13]. We propose a new measurement of tortuosity: nondominant variance ratio. It is the normalized sum of the variances of nodes in all nondominant directions. We apply principal component analysis (PCA) on the 3D coordinates of all nodes in a branch. The largest eigenvalue corresponds to the variance in the dominant direction. The non-dominant variance ratio is the sum of all eigenvalues except the largest one, divided by the sum of all eigenvalues. Figure 7 (left) illustrates the variances of the node locations in two orthogonal directions in a 2D plane. In this example, the nondominant variance ratio can be computed as $\frac{\sigma_1}{\sigma_0+\sigma_1}$. It is easy to see that the non-dominant variance ratio is a dimensionless quantity in the range [0, 1][1].

The measurement is defined on a vessel branch. The tortuosity of a vessel network is the average tortuosity values over all branches in the network.

Junction-node ratio—The junction node ratio is the number of junction points divided by the total number of nodes in the vessel network, where junction points are the nodes that belong to more than one branch. Figure 7 (right) illustrates a junction node and nonjunction nodes in a vessel network.

Tortuosity and the junction node ratio are used together to characterize the tumor vessel network. A predefined list of diffusion constants D_c is chosen. For each of the diffusion constants, a simulation is run for a fixed amount of time, $t = 40$ days, and then the features are measured for all times series within this run.

Figure 8 shows how these two features change while the tumor grows larger (i.e. more nodes in vessel network). When the tumor grows reasonably large, the curves of both tortuosity and junction node ratio tend to flatten out and the feature values converge to a relatively small range, which indicates that they become insensitive to the size of tumor. Moreover, the value-ranges corresponding to different diffusion constants (shown as different colors in Fig. 8) do not fully overlap, which implies that our features carry useful information for predicting the diffusion constant.

For each large tumor (by large, we mean the tumor is large enough so that the feature values can be considered stable, i.e. independent to the size of tumor), the tortuosity and junction

[1] By definition, the value of the nondominant variance ratio in the 3D case would not be close to 1. It is just a loose upper bound.

node ratio of its vessel network is computed, and visualized in Fig. 9. We can clearly see the plate-like structure of the data in 3D, because different tumors can be generated with a same diffusion constant.

5.2. Feedback Wright–Fisher Model

The number of three kinds of cells (SC, TAC, and TDC) are used as features to predict the parameter k in Eq (23). We visualize the three features and the k values used to generate them in Fig. 10.

6. EXPERIMENTS AND RESULTS

6.1. Tumor Vessels Data

Our dataset consists of around 700 data points from seven distinct diffusion constants (seven groups).

6.1.1. Parameter values—The parameter values we used in the experiment are listed in Table 1:

At the beginning of the HMC sampling process, $\mathscr{F}^{(0)}$ and $\Theta^{(0)}$ are initialized at their maximum likelihood values. $\mu_{\mathbf{w}}$ is initialized using a simple linear least-square regression.

6.1.2. Prediction results—We use a leave-one-out test mechanism. At every round we pick one group of data-points corresponding to one value of D_c as test input, and the rest for training. The ground truth diffusion constants, their predicted value and 95% prediction intervals are summarized in Table 2, and plotted in Fig. 11

We observe that the predictions are quite close to the ground truth with an average prediction error of 0.60. All the ground truth values successfully fall into the 95% prediction interval. In Fig. 11, we observe that our model gives a larger prediction interval to the groups at are located at the far end (e.g. the red group when $D_c = 1$) compared with the groups that are situated more toward the middle. The reason is that when the testing points are relatively far away from the training data, our GP model is less confident on its predictions than when the data are surrounded by other data. In other words, interpolation is easier than extrapolation.

6.1.3. Comparison—We compare our method with several baseline regression models and show the quantitative improvement in prediction accuracy. We run the baselines in two ways: (1) use all data (2) only on the group centers. Our baselines include:

<u>Linear regression:</u> We fit a linear function $y = \mathbf{a}^{\mathrm{T}}\mathbf{x} + b$ to all the training data, and make prediction at each testing input as

$$y_*^i = \mathbf{a}^{\mathrm{T}}\mathbf{x}_*^i + b \qquad i \in \{1, 2, \ldots, N_*\}. \quad (27)$$

The final prediction for this group is the average over the prediction at each sample in this group:

$$y_* = \frac{1}{N_*}\sum_{i=1}^{N_*} y_*^i. \quad (28)$$

Regression using quadratic function: Similar to the linear regression, but we fit a quadratic function instead in this case:

$$y=\mathbf{x}^{\mathrm{T}}A\mathbf{x}+\mathbf{b}^{\mathrm{T}}\mathbf{x}+c. \quad (29)$$

Regression using exponential function: In this case, we fit an exponential function to the training data:

$$y=e^{\mathbf{a}^{\mathrm{T}}\mathbf{x}+b}. \quad (30)$$

Standard Gaussian process regression: The detailed description of the standard Gaussian process regression can be found in Section 2. Similar to the other baselines, the group prediction is computed as the average of the predictions at each testing input. It is worth mentioning that the standard GP will not produce reliable error bars because it does not take the evidence into account in the correct manner.

The quantitative results of the baselines and our method are summarized in Table 3. Our method achieves the overall lowest prediction error of only 0.60. Notably, our model significantly reduces the prediction error of a standard GP from 2.05 to 0.60, which suggests that our method is not a trivial modification of a GP, but can better capture the special structure of the data and produce more accurate predictions.

6.2. Feedback Wright–Fisher Model

6.2.1. Parameter values—The parameter values we used in the experiment are listed in Table 4:

At the beginning of the HMC sampling process, $\mathscr{F}^{(0)}$ and $\Theta^{(0)}$ are initialized with their maximum likelihood values. $\mu_{\mathbf{w}}$ is initialized using a simple linear least-square regression.

6.2.2. Prediction results—We use again a leave-one-out test mechanism. The ground truth value of *k*, as well as its prediction by the model and the 95% prediction intervals are summarized in Table 5, and plotted in Fig. 12

Again, our predictions are very accurate with a small average prediction error of 0.09. All the ground truth values successfully fall into the 95% prediction interval.

6.2.3. Comparison—We compare our method to the same baselines used in our tumor vessel data experiments.

The quantitative results of the baselines and our method are summarized in Table 6. Our method again achieves the overall lowest prediction error of only 0.09.

7. CONCLUSION

We have proposed a fully (nonparametric) Bayesian approach to regression in a situation where multiple inputs (covariates) correspond to a single label (response). In this work, we have focussed on the prediction of the diffusion constant which was an important input parameter for the simulation of tumor growth. We have used two properties of the tumor vessel network, namely tortuosity and Junction Node Ratio to predict the diffusion constant. As a second experiment we have looked at the prediction of *k* in the feedback Wright–Fisher model for the number of stem cells, transit-amplifying cells and terminal differentiated cells.

In both cases our predictions were very accurate and the ground truths lie within the 95% prediction interval predicted by our model. Note that these uncertainty bands provide very useful information beyond the prediction value itself.

This seems to be the first fully Bayesian treatment of regression with multiple covariates per response value. However, we believe this type of regression problem is ubiquitous in biology because due to noise or extreme sensitivity to initial conditions (a.k.a. chaos) in the generating process we are often faced with a many-to-one correspondence between covariates and response variables. As such, our method may find widespread application in this scientific discipline.

APPENDIX

A. PARAMETERS AND PRIORS OF OUR MODEL

The parameters of our model are $\Theta = \{\psi_{\mathbf{K}}, \mathbf{w}, \sigma_n, \Sigma_i, \Sigma_*\}$. $\psi_{\mathbf{K}}=\left[l,\sigma_0^2\right]^{\mathrm{T}}$ are the hyperparameters of the kernel function in Eq (13), l is the scale factor. σ_0^2 is the variance. σ_n^2 implies the strength of the noise. $\mathbf{w} = [w^{(1)}, w^{(2)}, \ldots, w^{(d+1)}]$ is the weight vector of the exponential linear mean function in Eq (9). Σ_i and Σ_* are the covariance matrix of the samples within each group. They can be assumed to be diagonal, if each dimension of the inputs is independent.

$$\Sigma_i=\begin{bmatrix} \sigma_{i1}^2 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & 0 \\ \vdots & 0 & \sigma_{\mathrm{im}}^2 & \vdots \\ 0 & 0 & \cdots & \ddots \end{bmatrix}_{d\times d}, \Sigma_*=\begin{bmatrix} \sigma_{*1}^2 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & 0 \\ \vdots & 0 & \sigma_{*m}^2 & \vdots \\ 0 & 0 & \cdots & \ddots \end{bmatrix}_{d\times d}. \quad (31)$$

The prior on the hyperparameters of the kernel function in Eq (13) is modeled by Gamma distribution $\Gamma(\alpha,\beta)=x^{\alpha-1}\frac{\mathrm{e}^{-x/\beta}}{\beta^{\alpha}\Gamma(\alpha)}$ with $a = 1$:

$$p\left(\psi_K^{(m)}\right)=\Gamma\left(1,\beta_K^{(m)}\right), \quad p\left(\psi_{\mathbf{K}}\right)=\prod_m p\left(\psi_K^{(m)}\right). \quad (32)$$

The prior on the noise over y_i:

$$p\left(\sigma_n\right)=\Gamma\left(1,\beta_{\sigma_n}\right). \quad (33)$$

The prior on the weights of the mean function in Eq (9) is modeled as multivariate Gaussian distribution with diagonal covariance matrix:

$$p(\mathbf{w})=\mathcal{N}\left(\mu_{\mathbf{w}},\Lambda_{\mathbf{w}}\right). \quad (34)$$

The prior on the diagonal element of Σ_i and Σ_* is modeled by Gamma distribution:

$$p\left(\sigma_{\mathrm{im}}^2\right)=\Gamma\left(1,\beta_{\Sigma_{\mathrm{im}}}\right), \quad p\left(\sigma_{*m}^2\right)=\Gamma\left(1,\beta_{\Sigma_{*m}}\right). \quad (35)$$

Assuming that the elements on the diagonal are independent, $p(\Sigma_i)$ and $p(\Sigma_*)$ can be written as:

$$p\left(\Sigma_i\right)=\prod_m p\left(\sigma_{\mathrm{im}}^2\right), \quad p\left(\Sigma_*\right)=\prod_m p\left(\sigma_{*m}^2\right). \quad (36)$$

Let $\mathbf{z_i}=\left[z_i^{(1)},\ldots,z_i^{(m)},\ldots,z_i^{(d)}\right]^{\mathrm{T}}$, $\mathbf{z}_*=\left[z_*^{(1)},\ldots,z_*^{(m)},\ldots,z_*^{(d)}\right]^{\mathrm{T}}$. The prior on each dimension of the hidden variables is modeled to be Gamma-distributed:

$$p\left(z_*^{(m)}\right)=p\left(z_i^{(m)}\right)=\Gamma\left(1,\beta_z^{(m)}\right). \quad (37)$$

Notice that $\beta_z^{(m)}$ depends on the dimension index m but is independent to the group index i.

Since the dimensions of the inputs are independent, $p(\mathbf{z_i})$ and $p(\mathbf{z}_*)$ can be written as:

$$p\left(\mathbf{z_i}\right)=\prod_m p\left(z_i^{(m)}\right), \quad p\left(\mathbf{z}_*\right)=\prod_m p\left(z_*^{(m)}\right). \quad (38)$$

B. DERIVATIVES OF F

Derivatives of F with respect to $\mathscr{Z}$ and Θ for HMC. F is defined in Eq (20):

$$\begin{aligned} F &= -\log p\left(\mathscr{D},\mathscr{T},\mathscr{Z},\Theta\right) \\ &= -\sum_i\left[\log p\left(X_i|\mathbf{z_i},\Sigma_i\right)+\log p\left(\mathbf{z_i}\right)+\log p\left(\Sigma_i\right)\right]-\log p\left(X_*|\mathbf{z}_*,\Sigma_*\right)-\log p\left(\mathbf{z}_*\right)-\log p\left(\Sigma_*\right)-\log p\left(\{y_i\}_{i=1}^{c}|\{\mathbf{z_i}\}_{i=1}^{c}\right)-\log p\left(\mathbf{w}\right)-\log p \\ &= -\sum_i\left[\sum_{j=1}^{N_i}\log p\left(\mathbf{x_i^j}|\mathbf{z_i},\Sigma_i\right)+\sum_{m=1}^{d}\log p\left(z_i^{(m)}\right)+\sum_{m=1}^{d}\log p\left(\sigma_{\mathrm{im}}^2\right)\right]-\sum_{j=1}^{N_*}\log p\left(\mathbf{x}_*|\mathbf{z}_*,\Sigma_*\right)-\sum_{m=1}^{d}\log p\left(z_*^{(m)}\right)-\sum_{m=1}^{d}\log p\left(\sigma_{*m}^2\right)-\log p\left(\{y_i\}_{i=1}^{c}|\{\right. \end{aligned}$$

(

We define:

$$\alpha=K^{-1}\left(\mathbf{y}-\left[\begin{array}{c}\vdots\\ u\left(\mathbf{z_i}\right)\\ \vdots\end{array}\right]\right), \quad (40)$$

$$\bar{\mathbf{x}}_i=\frac{1}{N_i}\sum_{j=1}^{N_i}\mathbf{x_i^j}, \quad \bar{\mathbf{x}}_*=\frac{1}{N_*}\sum_{j=1}^{N_*}\mathbf{x}_*^{\mathbf{j}}. \quad (41)$$

The derivatives are:

$$\frac{\partial F}{\partial \mathbf{z_i}} \propto \frac{1}{\beta_{\mathbf{z}}}+N_i\Sigma_i^{-1}\left(\mathbf{z_i}-\bar{\mathbf{x}}_\mathbf{i}\right)+\frac{1}{2}\mathrm{tr}\left(K^{-1}\frac{\partial K}{\partial \mathbf{z_i}}\right) \quad (42)$$

$$-\frac{1}{2}\alpha^{\mathrm{T}}\frac{\partial K}{\partial \mathbf{z_i}}\alpha-\frac{\partial}{\partial \mathbf{z_i}}\left|\begin{array}{c}\vdots\\ u\left(\mathbf{z_i}\right)\\ \vdots\end{array}\right|\alpha, \quad (43)$$

$$\begin{aligned}\frac{\partial F}{\partial \mathbf{z}_*} &\propto \frac{1}{\beta_{\mathbf{z}}}+N_*\Sigma_*^{-1}\left(\mathbf{z}_* - \bar{\mathbf{x}}_*\right), \\ \frac{\partial F}{\partial \mathbf{w}} &\propto -\frac{\partial}{\partial \mathbf{w}}\begin{bmatrix} \vdots \\ u\left(\mathbf{z_i}\right) \\ \vdots \end{bmatrix}\alpha+\Lambda_{\mathbf{w}}{}^{-1}\left(\mathbf{w}-\mu_{\mathbf{w}}\right),\end{aligned} \tag{44}$$

$$\frac{\partial F}{\partial l} \propto \frac{l}{\beta_K^{(1)}}+\frac{1}{2}\mathrm{tr}\left[\left(K^{-1}-\alpha\alpha^{\mathrm{T}}\right)\frac{\partial K}{\partial l}l\right], \tag{45}$$

$$\frac{\partial F}{\partial \sigma_0^2} \propto \frac{2\sigma_0^2}{\beta_K^{(2)}}+\frac{1}{2}\mathrm{tr}\left[\left(K^{-1}-\alpha\alpha^{\mathrm{T}}\right)\frac{\partial K}{\partial \sigma_0^2}\sigma_0^2\right], \tag{46}$$

$$\frac{\partial F}{\partial \sigma_n^2} \propto \frac{2\sigma_n^2}{\beta_{\sigma_n}}+\frac{1}{2}\mathrm{tr}\left[\left(K^{-1}-\alpha\alpha^{\mathrm{T}}\right)\frac{\partial K}{\partial \sigma_n^2}\sigma_n^2\right], \tag{47}$$

$$\frac{\partial F}{\partial \sigma_{\mathrm{im}}^2} \propto \frac{1}{\beta_{\Sigma_{\mathrm{im}}}}+\frac{N_i}{2\sigma_{\mathrm{im}}^2}-\frac{1}{2}\sum_{j=1}^{N_i}\frac{\left(\mathbf{x}_{\mathbf{i}}^{\mathbf{j}^{(m)}}-z_i^{(m)}\right)^2}{\sigma_{\mathrm{im}}^4}, \tag{48}$$

$$\frac{\partial F}{\partial \sigma_{*m}^2} \propto \frac{1}{\beta_{\Sigma_{*m}}}+\frac{N_*}{2\sigma_{*m}^2}-\frac{1}{2}\sum_{j=1}^{N_*}\frac{\left(\mathbf{x}_*^{\mathbf{j}^{(m)}}-z_*^{(m)}\right)^2}{\sigma_{*m}^4}. \tag{49}$$

C. TUMOR VESSEL NETWORK MODELS

The progression of a vascularized tumor in three dimensions is simulated using a continuum multispecies tumor model developed by Wise *et al.* [14] coupled with a lattice-free discrete model of angiogenesis developed by Frieboes *et al.* [6]. We briefly describe the models here. We refer the readers to the references above, and the book by Cristini and Lowengrub [15], for further details.

C.1. Angiogenesis Model

The development of a tumor-induced neovasculature network is modeled using a lattice-free, discrete framework developed in ref. 6 together with several modifications that are described below. This builds on earlier work by the authors of refs. 8,16–18. The angiogenesis model generates a vascular network regulated by tumor angiogenic factors (TAF) such vascular endothelial growth factor (VEGF), e.g. [7]. Here, we model TAF using a continuum variable that describes the net effect of pro-angiogenic regulators. The concentration of TAFs, denoted by *c*, is governed by the diffusion–reaction equation:

$$0=D_c\nabla^2 c-\beta_d c+S_c\phi_h\left(c_{\mathrm{sat}}-c\right),$$

where D_c is the diffusivity, β_d is the natural decay rate, S_c is the transfer rate of the supply from the hypoxic cells, and c_{sat} denotes the saturation level. The volume fraction of hypoxic cells φ_h is defined as the volume fraction of viable cells where the cell substrate is lower than a specific threshold, which is here set to be the same as the necrotic threshold n_V.

The new capillaries form randomly at sprouts near the tumor boundary following the concentration of TAFs. The scheme first identifies all the sites where the $\varphi_V < 0.2$ and $c > 0.1$, which guarantees that the sites are outside the tumor and close to the tumor/host boundary. Then these sites are weighted by the c and one site is randomly selected from the list. The frequency of site generation was set to 5 per unit time step (day), which was calibrated to yield a reasonable number of vessels over the time course of the simulations presented herein. See ref. 6.

Vessels are described in terms of the trajectories taken by migrating endothelial cells [8]. A stochastic equation is prescribed for the leading endothelial cell at the vessel tip that describes the motion as a biased random walk:

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t}=s\mathbf{e}+\mathbf{v}_{\mathrm{random}}, \quad (50)$$

where $s = s_0|\nabla c|$ is the speed of the tip cell with s_0 a constant, $\mathbf{e} = (1 - w)\mathbf{e}_{\mathrm{old}} + w\nabla c$ is the direction of the tip cell with w a weighting factor and $\mathbf{e}_{\mathrm{old}}$ denotes the previous direction of the tip cell. Further, $\mathbf{v}_{\mathrm{random}}$ denotes a random direction This is a stochastic model of the chemotaxis of tip endothelial cells up gradients of TAFs. The endothelial cells just behind the tip are assumed to proliferate, providing a source of new endothelial cells to populate the growing vessel [8]. For simplicity, we do not consider the effect of haptotaxis (motion up gradients of extracellular matrix) here although this can be easily incorporated [6,8,17,18].

A vessel has a fixed probability of branching at each time step. When branching occurs, the leading endothelial cell splits into two leading cells with the new cells reorienting by a fixed angle of 30°. The two cells then continue to migrate and proliferate into new vessels. If the leading cell of one vessel crosses the trail of another vessel from a different sprout site, then anastomosis may occur (self-intersections are not allowed). This process forms a closed loop and the corresponding vessel segments between the two sprouts can now be a source of cell substrates to the surrounding tumor tissue.

The model presented here currently does not include blood flow rates in the vasculature or the associated morphological changes in the vascular network, such as branching induced by shear stress. Here, we assume GPF extravasation as soon as the vessels anastomose, which models the fact that the flow time scale is much faster than the tumor growth time scale. Simplified models of the blood fluid dynamics in capillary networks have been developed (e.g. see the reviews [19–21]) and will be considered in a future work.

C.2. Simulation

We perform numerical simulations of the model described in the previous subsection using a nondimensionalization described in [6]. In particular, space and time are nondimensionalized by the GPF diffusion length $\mathscr{L}=\left(D/\nu_v^u\right)^{1/2}$ and the mitosis time scale $\mathscr{T}=1/\lambda_v^m$. Note that because of the relations $\varphi_T + \varphi_H = 1$ and $\varphi_T = \varphi_V + \varphi_D$, we need to solve for only two variables. Following refs. 6,14, we solve for φ_T and φ_D. Note that we do not need to solve for φ_W as this variable is slaved to the growth of the tumor but does not influence the tumor progression.

While there are many parameters in the model, see Table 7, we focus here on the effect of the TAF diffusion coefficient D_c on the developing neovasculature network and the resulting tumor progression. This models the variable solubility of TAF isoforms. For example, it is known that due to cleavage by matrix metalloproteinases, VEGF isoforms may display varying degrees of solubility, e.g. see Lee *et al.* [22]. In particular, it is found that the more soluble isoforms lead to a more disorganized and less functional vessel network than the more insoluble isoforms. TAF is set to zero on the boundary of the domain (Dirichlet boundary condition), which models the intravasation of TAF into the vascular network. Indeed, soluble forms of tumor-induced TAF can be found in the blood.

We performed many simulations with TAF diffusion coefficient D_c ranging from 20 to 1, where we kept all other parameters unchanged but varied the initial tumor shape. In particular, the initial tumor shape is taken as a small random perturbation of a unit sphere. A sample of results are shown in Fig. 4. In the figure, the contours $\varphi_V = 0.5$ of the viable tumor volume fraction are plotted together with the neovascular network. Blue vessels denote sprouts which have not yet anastomosed to form a functional network. Vessels colored red denote the looped, or anaostomosed, vessels that are releasing GFPs into the tumor microenvironment. As can be clearly seen in the figure, the tumor size and the number of vessels are decreasing functions of the TAF diffusion coefficient, consistent with experimental observations. The results are quantified in Fig. 5, where the tumor volumes (a), the vessel lengths (b) and the ratio of the vessel and tumor volumes (c) are shown. Note that the vessel volume is obtained by assuming that the vessel network is a collection of cylindrical vessel segments, with a radius of 0.05 in nondimensional length (approximately $10\mu m$ in dimensional length). Again, all these quantities are decreasing functions of the TAF diffusion coefficient, and increasing functions of time.

REFERENCES

1. Rasmussen, C.; Williams, C. The MIT Press; Boston: 2006.
2. Freedman, D.; Purves, R.; Pisani, R. Statistics. 3rd ed.. W.W. Norton & Company; New York: 1998.
3. Duane S, Kennedy A, Pendleton B, Roweth D. Hybrid Monte Carlo. Phys Lett B. 1987; 195(2): 216–222.
4. Andrieu C, De Freitas N, Doucet A, Jordan M. An introduction to MCMC for machine learning. Mach Learn. 2003; 50:5–43.
5. Neal, R. Technical Report CRG-TR-93-1. Dept. of Computer Science, University of Toronto; 1993. Probabilistic inference using Markov chain Monte Carlo methods.
6. Frieboes HB, Jin F, Chuang Y-L, Wise SM, Lowengrub JS, Cristini V. Three dimensional multispecies nonlinear tumor growth II: tumor invasion and angiogenesis. J Theor Biol. 2010; 264:1254–1278. [PubMed: 20303982]
7. Takano S, Yoshii Y, Kondo S, Suzuki H, Maruno T, Shirai S, Nose T. Concentration of vascular endothelial growth factor in the serum and tumor tissue of brain tumor patients. Cancer Res. 1996; 56:2185–2190. [PubMed: 8616870]
8. Anderson ARA, Chaplain MAJ. Continuous and discrete mathematical models of tumor-induced angiogenesis. Bull Math Biol. 1998; 60:857–900. [PubMed: 9739618]
9. Fisher, R. The Genetical Theory of Natural Selection. Clarendon Press; Oxford: 1930.
10. Wright S. Evolution in Mendelian populations. Genetics. 1931; 16:97–159. [PubMed: 17246615]
11. McDonald D. Significance of blood vessel leakiness in cancer. Cancer Res. 2002; 62:5381–5385. [PubMed: 12235011]
12. Bullitt E, Gerig G, Pizer S, Lin W, Aylward S. Measuring tortuosity of the intracerebral vasculature from MRA images. IEEE Trans Med Imaging. 2003; 22(9):1163–1171. [PubMed: 12956271]
13. Hart W, Goldbaum M, Cote B, Kube P, Nelson M. Measurement and classification of retinal vascular tortuosity. Int J Med Inform. 1999; 53:239–252. [PubMed: 10193892]

14. Wise SM, Lowengrub JS, Frieboes HB, Cristini V. Three-dimensional multispecies nonlinear tumor growth: model and numerical method. J Theor Biol. 2008; 253:524–543. [PubMed: 18485374]
15. Cristini, V.; Lowengrub, JS. Multiscale modeling of cancer: An integrated experimental and mathematical modeling approach. Cambridge University Press; Cambridge UL: 2010.
16. McDougall SR, Anderson ARA, Chaplain MAJ, Sherratt J. Mathematical modelling of flow through vascular networks: implications for tumour-induced angiogenesis and chemotherapy strategies. Bull Math Biol. 2002; 64:673–702. [PubMed: 12216417]
17. Plank MJ, Sleeman BD. A reinforced random walk model of tumour angiogenesis and anti-angiogenic strategies. Math Med Biol. 2003; 20(2):135–181. [PubMed: 14636027]
18. Plank MJ, Sleeman BD. Lattice and non-lattice models of tumour angiogenesis. Bull Math Biol. 2004; 66:1785–1819. [PubMed: 15522355]
19. Lowengrub JS, Frieboes HB, Jin F, Chuang Y-L, Li X, Macklin P, Wise SM, Cristini V. Nonlinear modeling of cancer: Bridging the gap between cells and tumors. Nonlinearity. 2010; 23:R1–R91. [PubMed: 20808719]
20. Pries AR, Hopfner M, le Noble F, Dewhirst MW, Secomb TW. The shunt problem: Control of functional shunting in normal and tumor vasculature. Nat Rev Cancer. 2010; 10:587–593. [PubMed: 20631803]
21. Chaplain MAJ, McDougall SR, Anderson ARA. Mathematical modeling of tumor induced angiogenesis. Ann Rev Biomed Eng. 8(1006):233–257. [PubMed: 16834556]
22. Lee S, Jilani SM, Nikolova GV, Carpizo D, Iruela-Arispe ML. Processing of VEGF-A by matrix metalloproteinases regulates bioavailability and vascular patterning in tumors. J Cell Biol. 2006; 169:681–691. [PubMed: 15911882]

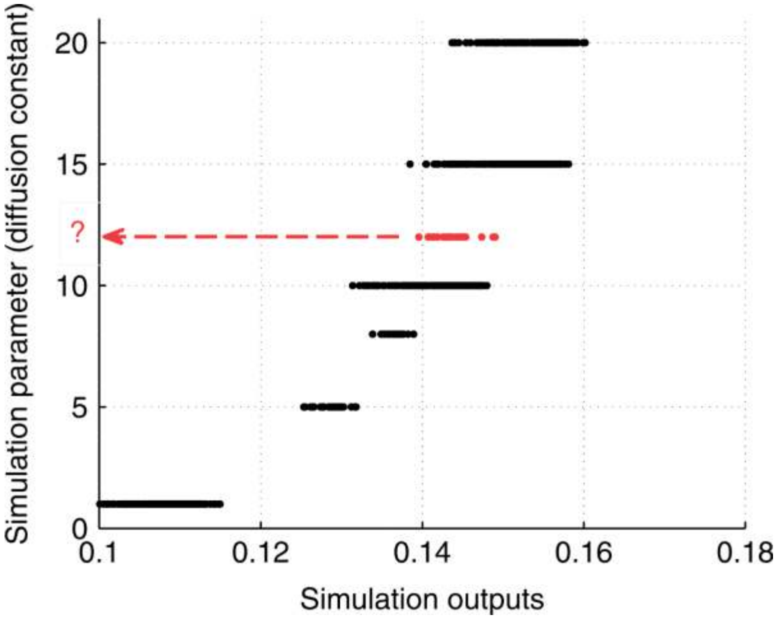

Fig. 1.

One feature of simulation outputs versus a simulation parameter (diffusion constant in the tumor growth model we use). Black dots denote observed data. Given a set of new observations that we know corresponding to a single simulation parameter (plotted in red dots), we want to predict the best parameter value that most likely generated them. Note that the simulation output varies even given a fixed parameter value because of the stochastic nature of the simulation. Likewise, a given simulation output could possibly be generated by more than one parameter value, which makes the conventional regression models not applicable. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

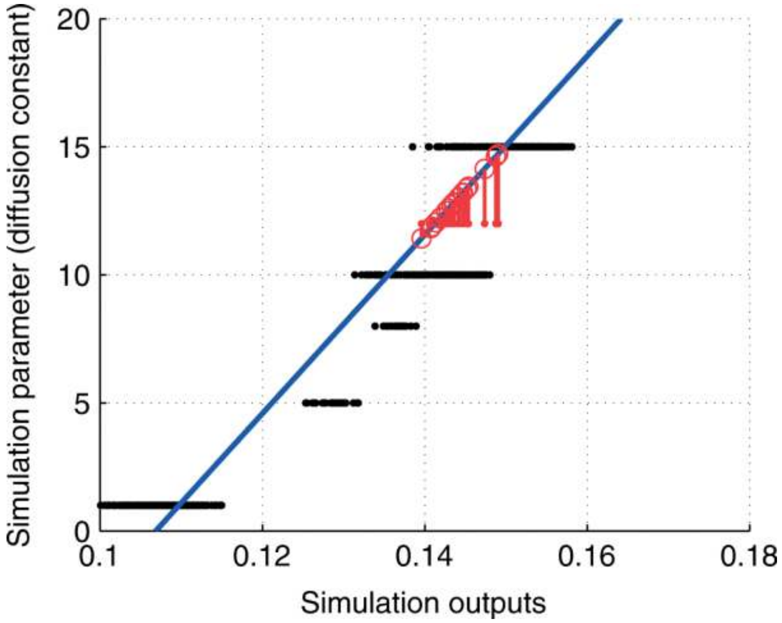

Fig. 2.

If we fit a linear function to the observed data (black dots), and use the learned linear function (blue line) to make predictions on the new data (red dots), we will predict different values at each of new input (red circles on the blue line indicate the predictions). But the linear function is not correctly modeling the special structure of the data, because it fails to take into account the fact that all the new inputs actually correspond to a same target. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

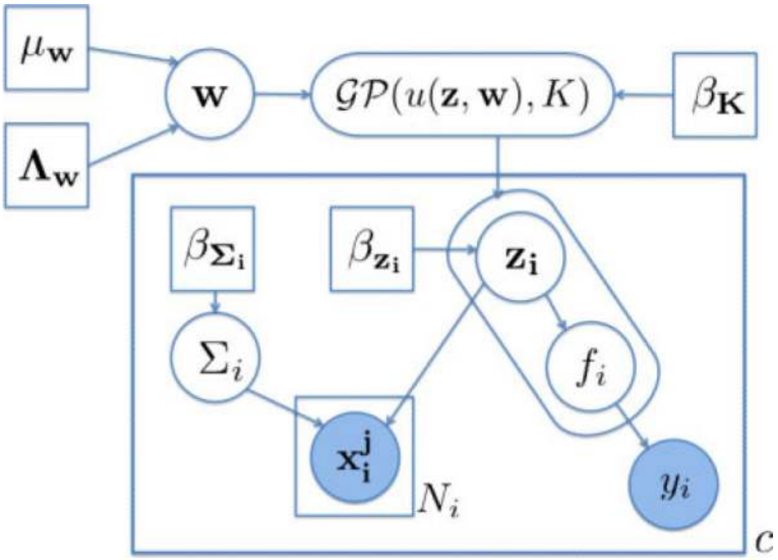

Fig. 3.

The graphical representation of our GP model. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

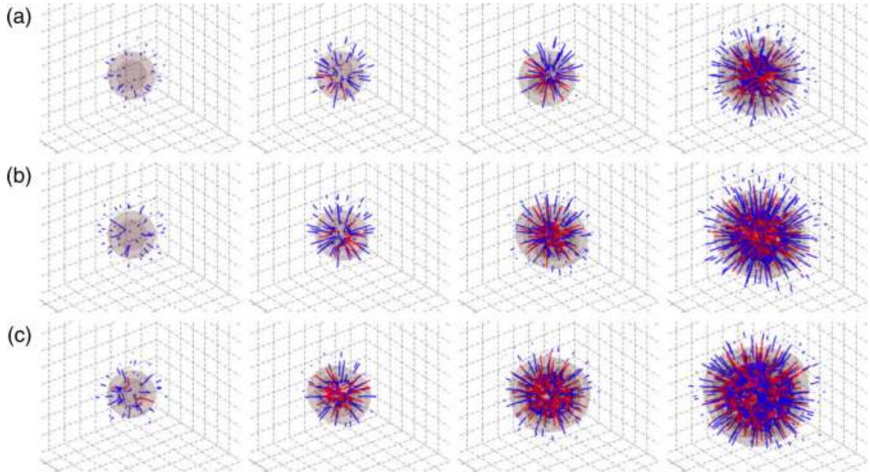

Fig. 4.
Tumor and vessel morphologies at times $t = 10$ (first column), 20 (second column), 30 (third column) and 50 (fourth column), from left to right. In each row, the TAF diffusivity D_c is different. (a) $D_c = 20$; (b) $D_c = 10$; (c). $D_c = 3$. [Color figure can be viewed in online issue, which is available at wileyonlinelibrary.com.]

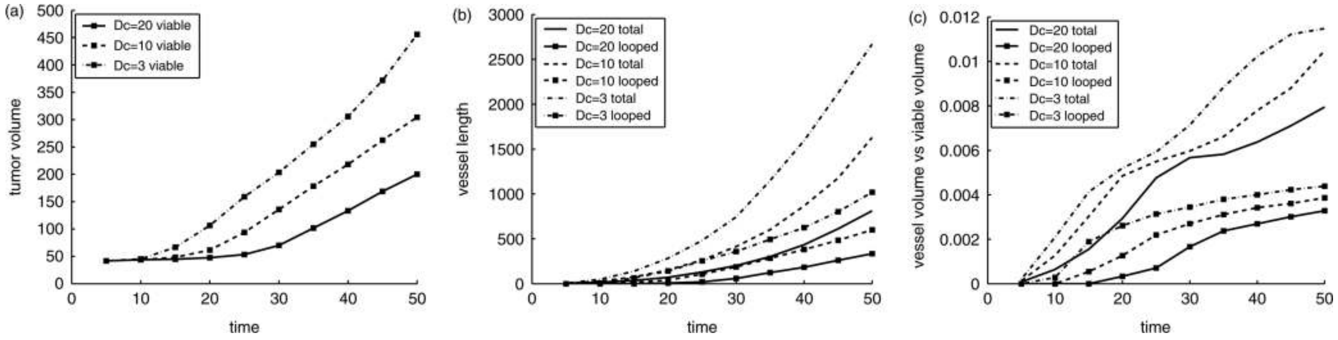

Fig. 5.

Details of the simulations shown in Fig. 4. (a) Tumor volume; (b) Total length of both looped vessels and the total neovascular network; (c) the ratio of the vessel volume to the tumor volume. The TAF diffusion coefficient D_c is labeled.

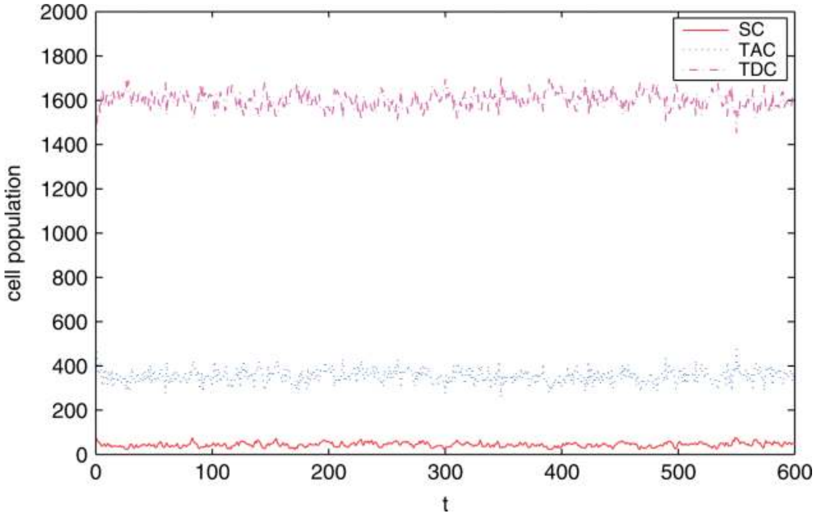

Fig. 6.

The trajectories of cell population in our simulation with the feedback Wright-Fisher model, using parameters $p_1 = 0.1$, $N = 2000$ and $k = 5$. [Color figure can be viewed in the online issue, which is available at wileonlinelibrary.com.]

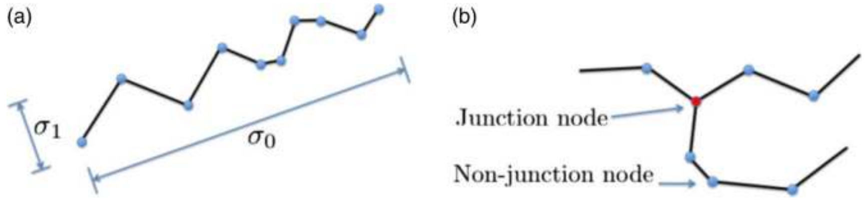

Fig. 7.

(a) Computing the variances of the node locations in the orthogonal directions of a vessel branch. (b) Junction node and nonjunction nodes in a vessel network. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

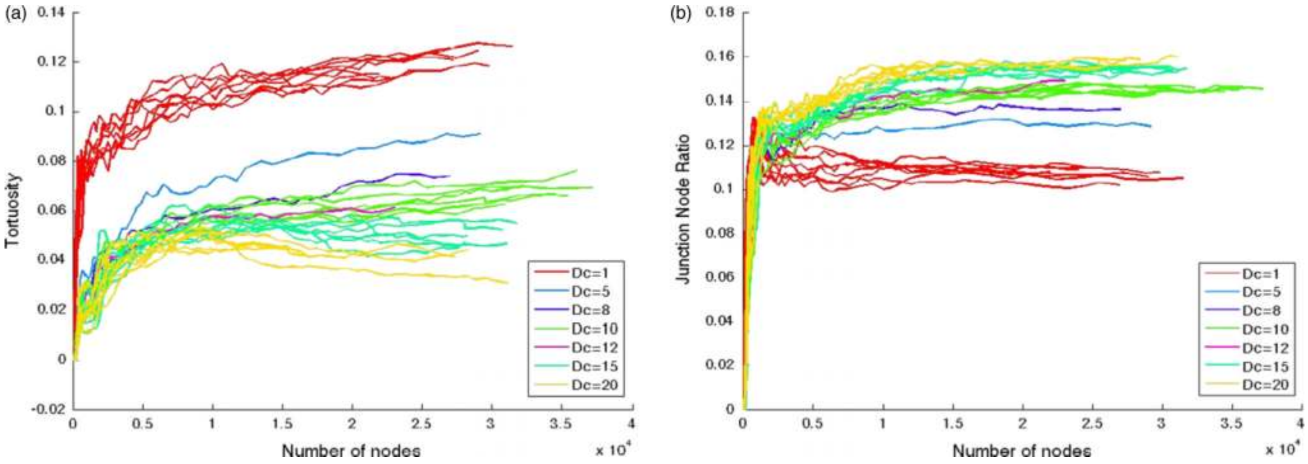

Fig. 8.

Feature values of different size of tumors. (a) Tortuosity versus number of nodes in tumor vessel network. (b) Junction node ratio versus number of nodes in tumor vessel network. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

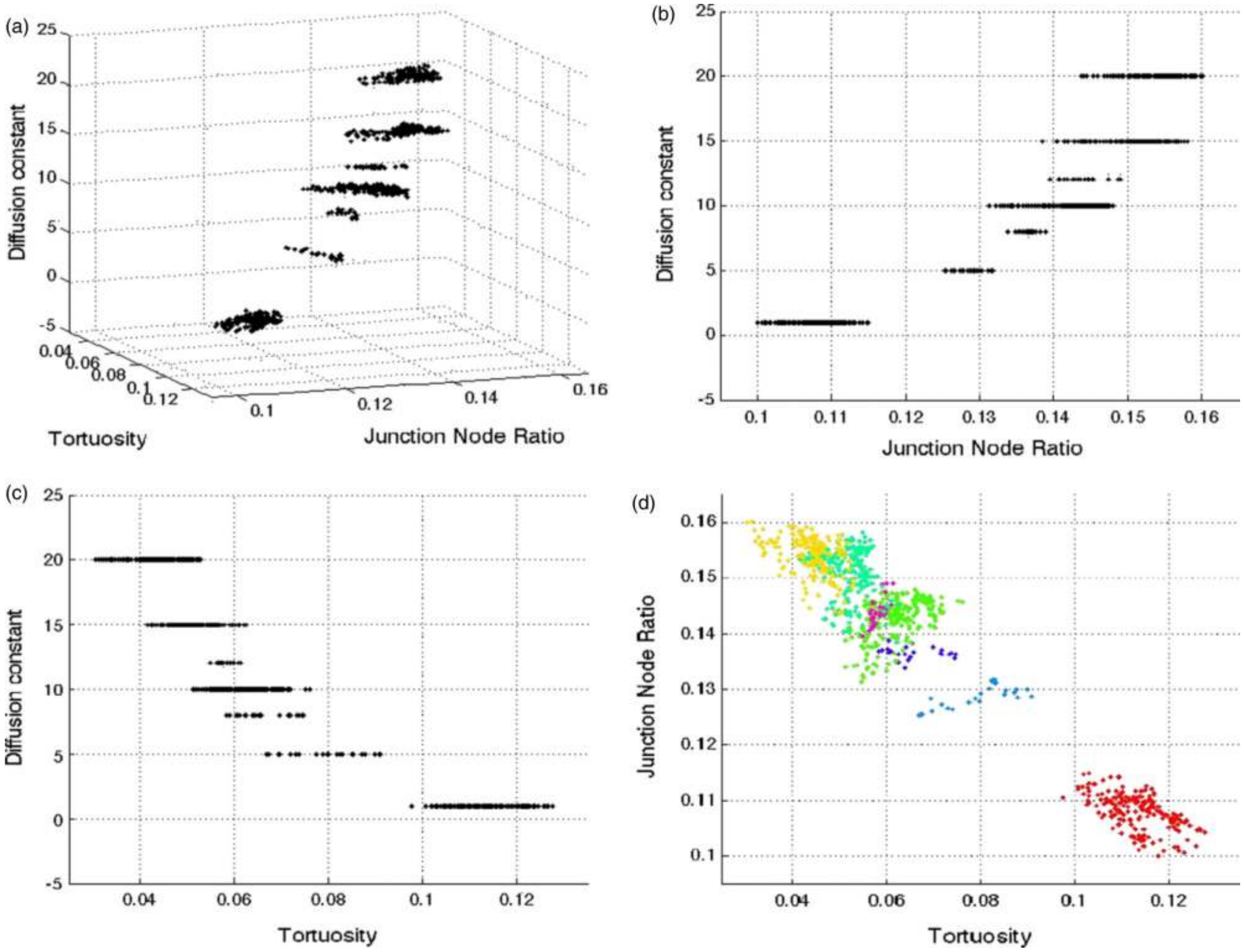

Fig. 9.

Visualization of the two features (tortuosity and junction node ratio) that are used to predict the diffusion constant D_c: (a) 3D plot; (b)–(d) 2D views from three different axes. In (d), each color shows the tumors generated using a same diffusion constant D_c. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

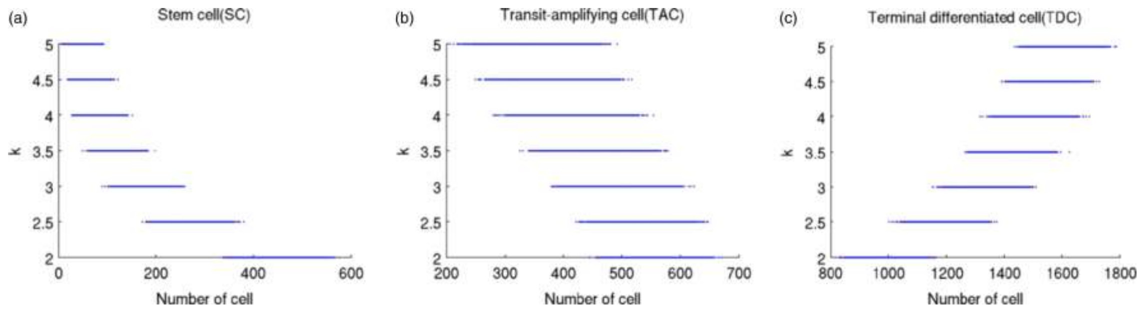

Fig. 10.

Visualization of the three features (SC, TAC, and TDC) that are used to predict *k*: (a) stem cell; (b) transit-amplifying cell; (c) terminal differentiated cell. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

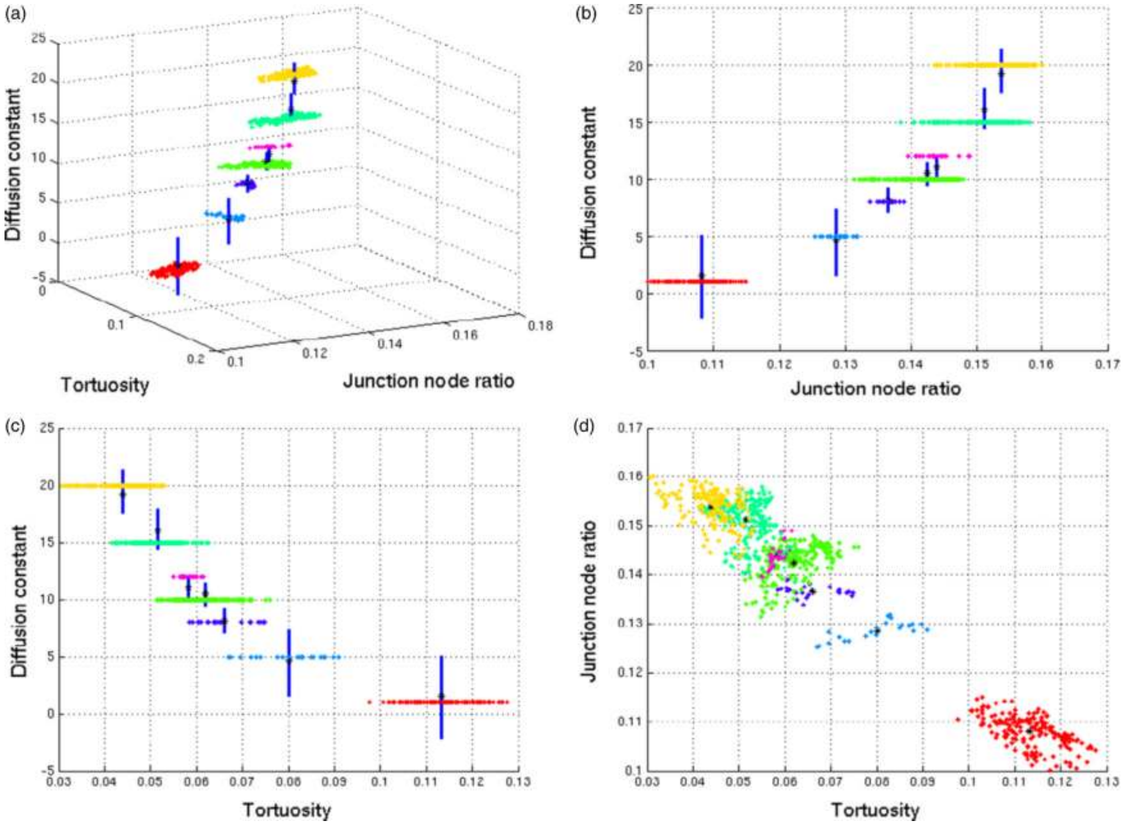

Fig. 11.

The prediction results on the tumor data with 95% prediction interval. Each color shows the tumors generated using a same diffusion constant D_c. The blue error bars give the 95% prediction interval on each group: (a) 3D visualization; (b)–(d) views from three different axes. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

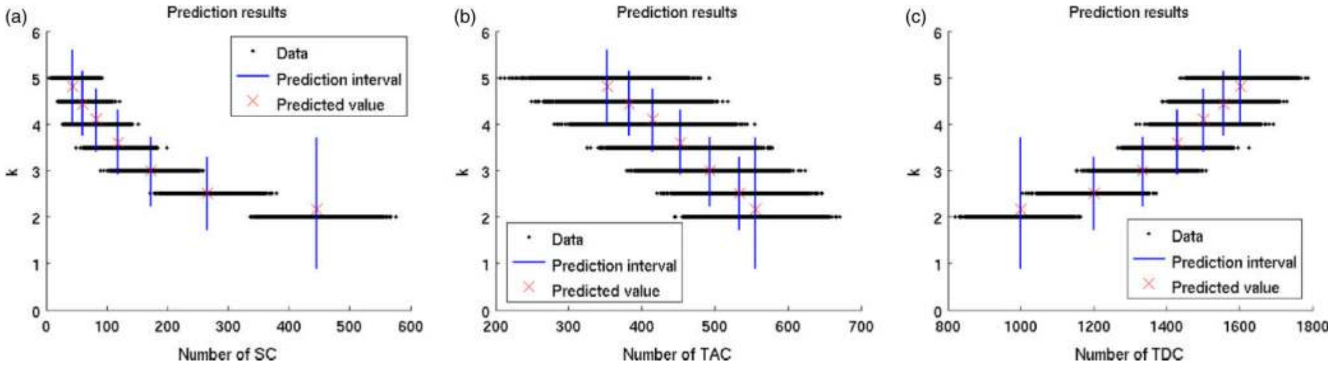

Fig. 12.

The prediction results of *k* with prediction interval. The blue error bars give the 95% prediction interval of each group. (a)–(c) show the same result but for different features. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

Table 1

The parameter values we used in the tumor vessel experiments.

$\boldsymbol{\beta_{\mathbf{z}}^{(m)}}$	**0.5**	$\boldsymbol{\Lambda_{\mathrm{w}}}$	**3*I***
$\beta_{\mathbf{K}}^{(1)}$	1	$\beta_{\Sigma\,\mathrm{im}}$	1
$\beta_{\mathbf{K}}^{(2)}$	2	$\beta_{\Sigma\,*m}$	1
$\beta_{\sigma\,\mathrm{n}}$	0.5		

Table 2

Prediction results on the tumor data using our model.

Groundtruth	Prediction	Prediction interval (95%)	Absolute error
1.00	1.49	[–2.17, 5.03]	0.49
5.00	4.70	[1.56, 7.30]	0.30
8.00	8.10	[7.04, 9.17]	0.10
10.00	10.44	[9.31, 11.44]	0.44
12.00	11.04	[9.97, 12.05]	0.96
15.00	16.05	[14.38, 17.92]	1.05
20.00	19.12	[17.52, 21.33]	0.88
Average absolute error:			0.60

Table 3

Comparison of the prediction accuracy of our method and several baselines on the tumor vessel data. Our method achieves the overall lowest error of 0.60.

	Linear		**Quadratic**		**Exponential**		**Standard GP**		
Ground truth	**All**	**Centers**	**All**	**Centers**	**All**	**Centers**	**All**	**Centers**	**Ours**
1.00	−9.73	−10.08	−4.84	9.43	2.40	2.37	1.59	2.61	1.49
5.00	7.95	6.77	5.02	3.20	5.72	4.80	5.34	3.20	4.70
8.00	11.09	9.63	8.79	9.70	8.71	8.68	9.44	8.72	8.10
10.00	13.33	12.57	11.59	10.39	10.84	10.02	11.84	10.28	10.44
12.00	13.23	12.34	11.96	11.67	11.54	11.34	10.62	11.37	11.04
15.00	15.56	18.18	16.04	16.01	15.79	15.65	18.42	16.41	16.05
20.00	14.75	14.20	16.70	17.70	17.76	20.69	14.63	16.89	19.12
Avg. Error	**3.88**	**3.77**	**1.80**	**2.28**	**1.02**	**0.61**	**2.05**	**1.37**	**0.60**

Table 4

The parameter values we used in the Wright–Fisher model experiments.

$\boldsymbol{\beta}_{\mathbf{z}}^{(\boldsymbol{m})}$	**5**	$\Lambda_{\mathbf{w}}$	**0.005*I***
$\beta_{\mathbf{K}}^{(1)}$	3	$\beta_{\Sigma \mathrm{im}}$	5
$\beta_{\mathbf{K}}^{(2)}$	0.05	$\beta_{\Sigma *m}$	5
$\beta_{\sigma n}$	0.05		

Table 5

Prediction results on the feedback Wright–Fisher data using our model.

Ground truth	Prediction	Prediction interval (95%)	Absolute error
2.00	2.16	[0.90, 3.71]	0.16
2.50	2.51	[1.73, 3.29]	0.01
3.00	2.98	[2.23, 3.74]	0.02
3.50	3.61	[2.91, 4.31]	0.11
4.00	4.11	[3.41, 4.77]	0.11
4.50	4.44	[3.76, 5.14]	0.06
5.00	4.82	[4.01, 5.59]	0.18
Average absolute error:			0.09

Table 6

Comparison of the prediction accuracy of our method and several baselines on the Wright–Fisher data. Our method achieves the overall lowest error of 0.09.

	Linear		Quadratic		Exponential		Standard GP		
Ground truth	**All**	**Centers**	**All**	**Centers**	**All**	**Centers**	**All**	**Centers**	**Ours**
2.00	1.08	2.54	3.44	2.45	1.62	2.18	2.39	2.33	2.16
2.50	2.66	2.36	2.31	2.57	2.63	2.41	2.65	2.40	2.51
3.00	3.28	2.98	3.10	3.16	3.20	3.07	3.16	2.91	2.98
3.50	3.73	3.57	3.69	4.02	3.66	3.59	3.61	3.56	3.61
4.00	4.09	4.09	4.14	3.87	4.05	4.11	4.10	4.08	4.11
4.50	4.36	4.53	4.47	4.48	4.35	4.59	4.55	4.58	4.44
5.00	4.42	4.82	4.57	4.99	4.50	4.87	4.49	4.91	4.82
Avg. Error	**0.34**	**0.15**	**0.36**	**0.19**	**0.22**	**0.11**	**0.21**	**0.12**	**0.09**

Table 7

Nondimensional angiogenesis parameters used for the vascularized tumor simulations shown in Figs. 4 and 5.

ν_{ves}	**0.4**	D_c	**Varied**
β_d	2	S_c	1
c_{sat}	1	r_{ves}	0.4
$\varepsilon_{\mathrm{ves}}$	0.1	C_{ves}	1
σ	$\sqrt{2}$	p_{crush}	0.6
s_0	0.2	w	0.9