ISSN 1936-5330

kcFED Research Working Papers

Understanding Models and Model Bias with Gaussian Processes

Thomas R. Cook and Nathan M. Palmer

June 2023
RWP 23-07
http://doi.org/10.18651/RWP2023-07

Federal Reserve Bank of Kansas City

Understanding Models and Model Bias with Gaussian Processes

Thomas R. Cook[*][†] Nathan M Palmer[†][‡]

June 1, 2023

Abstract

Despite growing interest in the use of complex models, such as machine learning (ML) models, for credit underwriting, ML models are difficult to interpret, and it is possible for them to learn relationships that yield de facto discrimination. How can we understand the behavior and potential biases of these models, especially if our access to the underlying model is limited? We argue that counterfactual reasoning is ideal for interpreting model behavior, and that Gaussian processes (GP) can provide approximate counterfactual reasoning while also incorporating uncertainty in the underlying model's functional form. We illustrate with an exercise in which a simulated lender uses a biased machine model to decide credit terms. Comparing aggregate outcomes does not clearly reveal bias, but with a GP model we can estimate individual counterfactual outcomes. This approach can detect the bias in the lending model even when only a relatively small sample is available. To demonstrate the value of this approach for the more general task of model interpretability, we also show how the GP model's estimates can be aggregated to recreate the partial density functions for the lending model.

[*] Federal Reserve Bank of Kansas City Email: thomas.cook@kc.frb.org

[†] The views expressed in this article are those of the authors and do not necessarily reflect the views of the Federal Reserve Board, the Federal Reserve Bank of Kansas City or the Federal Reserve System.

[‡] Federal Reserve Board of Governors

JEL Classifications: C10, C14, C18, C45

1 Introduction

In recent years there has been a growing appetite to use machine learning for finance and economics. Fintech firms in particular have suggested that ML models, combined with alternative data sources (big data) can help extend credit access.

Despite their appeal approaches that use AI/ML/big data are difficult to interpret. It can be challenging to determine or understand what influences them and it is possible for these approaches to learn relationships (or representations) that yield de facto minority discrimination. The potential for ML models to produce discriminatory outputs is not an abstract concern as there have been many recent examples of machine learning models exhibiting some form of minority bias in several domains from facial recognition (Buolamwini and Gebru, 2018), to speech recognition (Koenecke et al., 2020), to recidivism prediction (Angwin et al., 2016).

How can we understand the behavior of a complex, nonlinear model? How can we determine if a model is exhibiting bias against minority groups? Furthermore, how can these things be done if we are limited in our access to the underlying model? In this paper we argue that counterfactual reasoning is an ideal way of thinking about interpreting model behavior and that gaussian processes can be used to enable approximate counterfactual reasoning about black box models while also enabling us to appreciate the uncertainty over the underlying model's functional form.

This paper builds on several large bodies of reseaerch on methodologies of causal inference, the economics of discrimination and machine learning (computer science). We advocate for an approach similar to the field studies and correspondence studies in labor discrimination (see e.g. Bertrand and Mullainathan, 2004). It is a departure from broader studies on discriminatory lending in fintech, much of which focuses on disparate outcomes in the aggregate sense and in a sense that may mask more complex patterns of bias or discrimination otherwise present at the margins (Bartlett et al., 2022; Bhutta, Hizmo, and Ringo, 2021; Popick, 2022, and others). Counterfactual reasoning has been more widely explored in the computer science literature on model bias, but the focus of this line of research has been primarily on the use of counterfactual reasoning to correct or remediate model bias (see Corbett-Davies and Goel, 2018; Jung et al., 2018; Pierson, Corbett-Davies, and Goel, 2018; Wang, Ustun, and Calmon, 2019). More broadly, there is a long line of research on the use of counterfactuals to understand natural processes, and more recent discussion of how that might be extended to model interpretation (Athey and Imbens, 2019; King, Keohane, and Verba, 1994; Pearl, 2009; Rubin, 1978). We expand on this line of thought by using gaussian processes to approximate counterfactual outcomes when the true model is inaccessible.

We proceed as follows. First we will discuss why counterfactuals are an ideal framework for model interpretation especially when trying to determine if a model exhibits minority bias. We will identify the major benefits of this approach as well as hurdles to its use. In the subsequent sections we will introduce gaussian process (GP) regression as a method to facilitate counterfactual reasoning when (a) new model predictions cannot be generated and (b) the exact structure of the model is not accessible. This approach is illustrated in an exercise in which we use GP regression to identify bias in simulated lending models exhibiting varying degrees of bias.

1.1 model interpretation through counterfactual reasoning

Why not use non-counterfactual methods to explain model behavior? There are a variety of ways we might consider trying to interpret an ML model's behavior. The first and most appealing in the context of classical econometrics is a simple interpretation of fitted model parameters. When using machine learning models, however, such an approach is usually infeasible if for no other reason than the sheer number of fitted parameters and because ML models are frequently not linear in their parameters.

Without interpreting model parameters, we might instead consider describing model behavior in terms of feature importances. This approach, however, is a much weaker way of describing model behavior and, while it might point to which model inputs are most influential on a model's behavior, it does not provide insight into the magnitude of effect of model inputs nor does it account for complex relationships between model inputs.

An alternative approach is to characterize model behavior in terms of aggregate model behavior. For example, we might describe the effect of a variable, p, on a model, f, by calculating the aggregate difference in model outcomes between one class of observations and another, $E[f(x)|x^{(p)} = 1] - E[f(x)|x^{(p)} = 0]$. This risks overlooking important control variables, may mask discrimination where it does occur and runs up against important selection type effects (the independent variables distribution between minority and non-minority classes may be fundamentally different). For additional discussion, see Bertrand and Duflo (2017, pg 7) in which the authors argue that, to ascertain whether a model is biased, we want and need to understand model behavior at the margins (i.e. at the level of the individual observation).

Counterfactual reasoning is an alternative to these approaches. We use the term counteractuals here in the sense in which it is used in King, Keohane, and Verba (1994), Pearl (2009), and Rubin (1978). To be precise, consider a model $y = f(x)$ where x is a P-length vector of inputs and where we can distinguish the p-th entries in x from all other entries in x by writing $x = (x^{(p)}, x^{(\neg p)})$. For a given observation, the counterfactual on p differs from the observation only in terms of p: $x' = (x'^{(p)}, x^{(\neg p)})$. If f is deterministic, then we can understand the change in y under the counterfactual on p to be $f(x) - f(x')$

Counterfactual reasoning is appealing because it allows us to explain the behavior of a model regarding *specific* observations in light of changes to its inputs. That is, it explains the behavior of a model for a particular outcome by contrasting it with model outputs under alternative (*what-if*) scenarios. This allows us to describe model behavior even if it is a black box where the particular shape or nature of f is unknown. Further, we can compare a specific observation to a variety of counterfactuals to recover measures of feature importance and marginal effects and we can aggregate comparisons to counterfactuals across many observations to characterize aggregate effects. Of course, actually employing counterfactual reasoning is often challenging as in nearly all observational settings *the fundamental problem of causal inference* prevents us from directly measuring/observing this causal effect. That is, since we cannot observe both a thing (x) and its counterfactual (x'), we cannot observe the outcomes that follow from a thing ($f(x)$) and its counterfactual ($f(x')$).

It is interesting to note that the fundamental problem of causal inference is primarily limiting when we are talking about causality in the physical world. For the purposes of explaining ML, however, we are talking about causality as it relates to the behavior of mathematical models. If we have access to the model we can easily generate actual ($f(x)$) and counterfactual (i.e. potential) outcomes ($f(x')$) and thereby easily assess the causal effect of changes in model inputs on the model outputs. This makes counterfactual reasoning a compelling and powerful framework to consider when trying to explain the behavior of complex models.

Despite the fact that it is more readily possible to explore counterfactuals when considering model behavior, there are two hurdles that encumber its use in practice. First, we may not have direct access to the model, model software or fitted parameters that would be necessary to generate new model predictions from counterfactual observations. Second, even if we can submit counterfactual observations to the model and generate new model predictions, it is possible that the posited counterfactuals are implausible. Implausible counterfactuals are not very problematic for linear models. For ML models, however, they pose a unique hazard as ML models tend to perform poorly/erratically on data that is too dissimilar from the training data used to fit the model.

The contribution of this paper, therefore is to introduce gaussian processes as a technique that overcomes these hurdles and enables the characterization of model behavior through the lens of counterfactual inference. More specifically, we introduce GP regression as a technique that enables reasoning about a model's behavior without direct access to the model itself and in such a way that we can quantify our uncertainty over the model's functional form. Though we will illustrate this through a discussion of detecting bias in a model, the general technique should be applicable to many other scenarios involving the interpretation of complex model behavior.

2 Counterfactual Inference on Black Boxes

Consider a collection of observations $X = (x_1, x_2 \dots x_N)$, from which we can form a dataset of model inputs and outputs, $D = \{X, f(X)\}$. Further, for simplicity, assume f is deterministic[1]. One way to think about D is as a low-resolution image or

[1] This is not wholly unreasonable since, for our purposes, f represents a fitted machine learning model that, in many cases should admit no random variation. This assumption here serves primarily

representation of the underlying model function f. Thought of this way, counterfactual reasoning is simply a matter of upsampling a low-resolution image or, alternatively, interpolation between known model outcomes[2](See Funke and Gronwald, 2009; King and Zeng, 2006).

There are a number of ways to do interpolation from a finite sample of data. These range form simple approaches such as linear interpolation to more complex approaches such as polynomial regression or even generative adversarial networks. We propose using Gaussian process (GP) regression for this purpose for a few specific reasons. First, GP regression models can be universal function approximators[3]. Second, GP regression can be constructed so that the fitted GP model is constrained to pass through D exactly (and should therefore pass through all observed data points). This is useful since we are interested in describing the behavior of a (deterministic) machine learning model where the model function should exhibit essentially no random variation. Third GP regression allows us a way to quantify uncertainty over the functional form of f and thus a way to express uncertainty over the interpolated outcomes from counterfactuals. Among other things, this is will be important for our ability to assess the plausibility of counterfactuals.

3 Gaussian Processes

In this section, we present an overview of the intuition behind Gaussian process regression. It should not be taken as comprehensive. For further detail, consult Williams and Rasmussen (2006), which provides a much more detailed treatment of the topic.

Consider a function, f, over an arbitrary (potentially infinite) domain $\mathscr{X}$. One way to think about this function is to consider the value of the function at any given

to simplify discussion and can be relaxed without loss of generality.

[2] Where counterfactuals are present in D, they can be used directly for counterfactual reasoning. I.e. where D contains both $x_i, f(x_i)$ and $x'_i, f(x'_i)$ for a deterministic f this may be essentially possible, but the likelihood tends to decrease in the dimensionality of x. This is typically not the case, however, and where f is not deterministic (such as with observations of some natural phenomena), we would generally expect the fundamental problem of causal inference to apply

[3] This is dependent on choice of kernel function. For the purposes of this paper, we focus on kernels that can satisfy the universal approximating property as discussed in Micchelli, Xu, and Zhang (2006).

point, $x \in \mathscr{X}$, as a random variable, $\phi_x := f(x)$, following a Gaussian distribution. We can take the collection of these random variables, $\Phi = \{\phi_x\}$, to define a stochastic process characterizing f on $\mathscr{X}$. If we further define this process as a Gaussian process, then the distribution of any finite collection $\Phi_X \subset \Phi$ is multivariate Gaussian and the marginal distribution of any individual ϕ_x is univariate normal. The Gaussian process is specified by a mean function, $\mu(x)$ and a kernel function $k(x, x')$ and we can write the GP as $\Phi \sim GP(\mu, k)$. For a given (finite) collection of inputs[4], X, we can draw a sample from the GP, yielding $\Phi_X \sim N(\mu^{(X)}, \Sigma^{(X,X)})$, where $\mu_i^{(X)} = \mu(x_i)$ and where $\Sigma^{(X,X)}$ is a covariance matrix with entries $\Sigma_{ij}^{(X,X)} = k(X_i, X_j)$.

Gaussian process regression is a Bayesian method that enables inference about $p(\phi_{x'}|f(x))$ – the distribution of a counterfactual outcome given realized set of inputs and model outputs, $D = \{X, f(X)\}$. With a prior $\Phi \sim GP(\mu, k)$, and given D, the posterior predictive distribution is a GP: $\Phi|D \sim GP(\mu_D, K_D)$ where for all $x' \in \mathscr{X}$

$$\begin{aligned} \mu_D(x') &= \mu(x') + \Sigma^{(x',X)}(\Sigma^{(X)})^{-1}(f(X - \mu(X))) \\ K_D(x') &= \Sigma^{(x',x')} - \Sigma^{(x',X)}(\Sigma^{(X,X)})^{-1}\Sigma^{(X,x')}. \end{aligned} \tag{1}$$

And for a finite collection $X' \subset \mathscr{X}$, the posterior predictive distribution is multivariate normal, i.e. $\Phi_{X'}|D \sim N(\mu_D(X'), \Sigma^{K_D} = K_D(X'))$.

Figure 1 shows functions sampled from a GP prior. Figure 2 shows posterior of this GP after being conditioned on several specific (i.e. known or realized) data points. The mean function is drawn in black. Note from this two things: that the conditioned GP passes exactly through the known points and that the uncertainty about the value of the GP at those points collapses to zero.

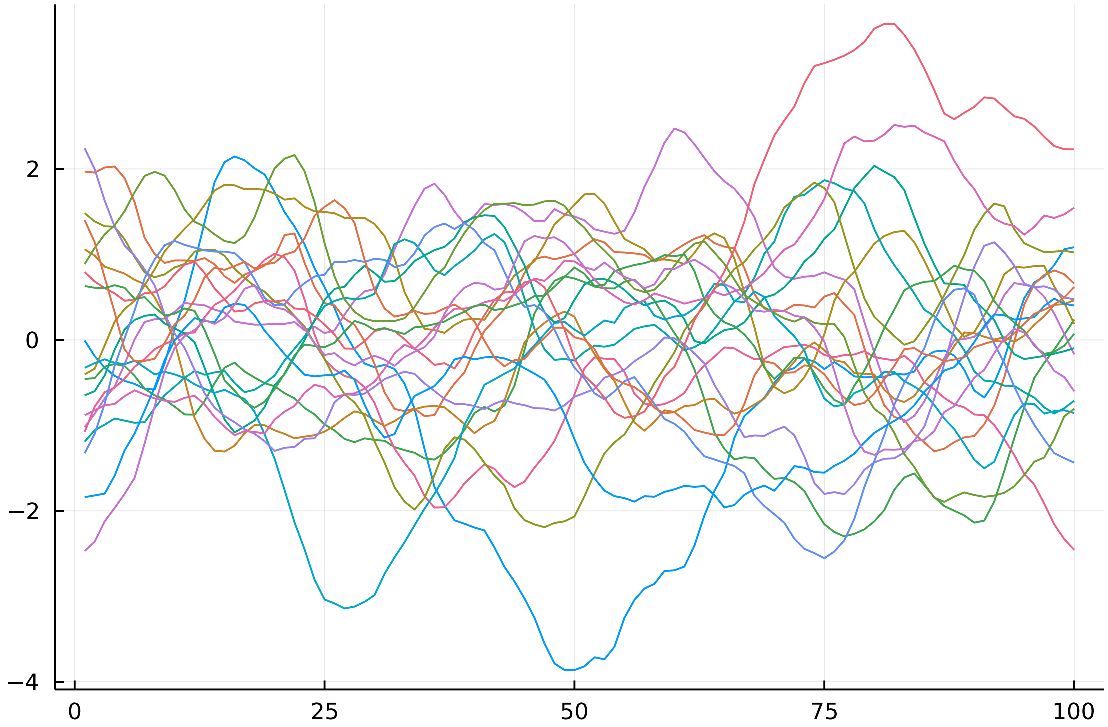
If not at this point clear, the behavior of the Gaussian process is primarily governed by the covariance kernel function[5], k. There are many different kernel functions that we might choose when setting up the GP regression. For our purposes, however, the ideal choice of kernel is one that demonstrates the universal approximation property[6].

[4] i.e. $X = (x_1, x_2, ...)$, where each $x_i \in \mathscr{X}$. Typically, we might think of X as a set of observations, counterfactuals, datapoints of interest, etc..

[5] This is so much the case that it is common to choose $\mu(x) = 0$ in the specification of the GP prior.

[6] Essentially, this property guarantees that a kernel function can approximate any function in its reproducing Hilbert space of continuous functions defined on $\mathscr{X}$ to an arbitrary degree of precision. In simpler terms, we can think of this as meaning that the kernel can be used to represent any

Figure 1: Functions sampled from unconditioned GP



One such kernel is the squared exponential kernel function

$$k_{se}(x,x') = \sigma^2 e^{-\frac{(x-x')^2}{2\ell^2}} \tag{2}$$

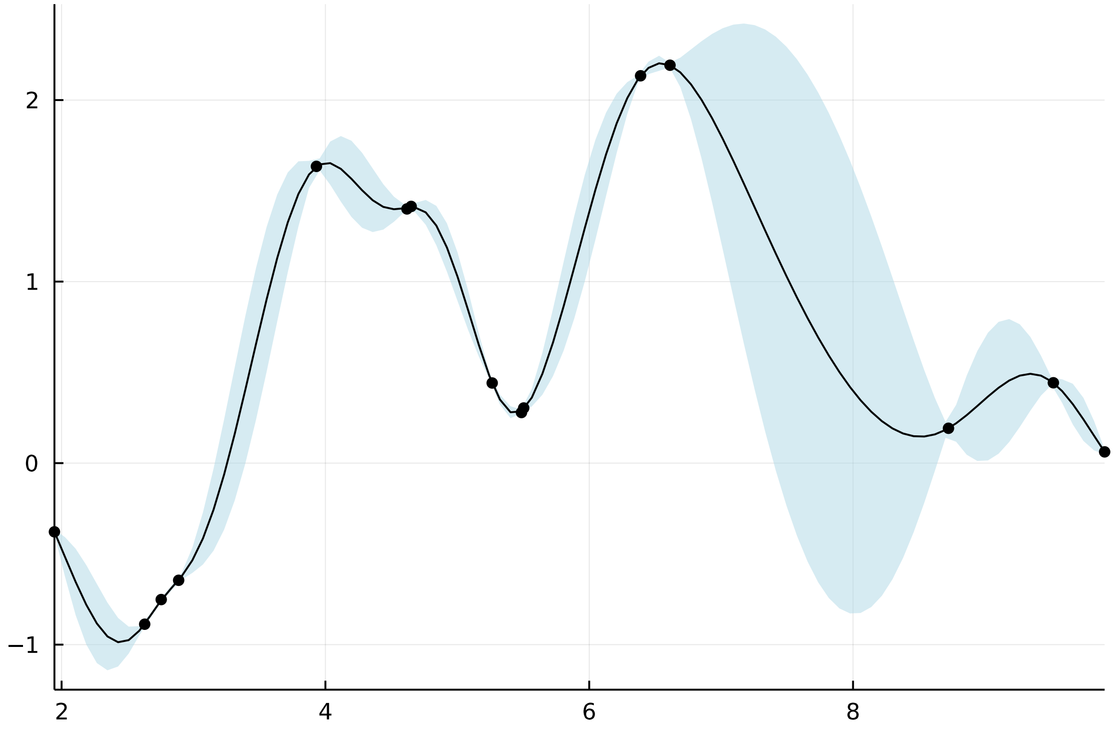
where ℓ is the characteristic length-scale and σ^2 is the variance in the kernel output (a simple scaling factor). We can think of the characteristic length-scale as controlling the radius around x in which we expect $(f(x), f(x)')$ to be similar. Both ℓ and σ^2 are free parameters that can be fit by either frequentest (e.g. maximum likelihood[7]) or Bayesian approaches (e.g. MCMC, variational inference).

The SE kernel produces high values where inputs (x, x') are close together and low values (near zero) as the distance between them increases. The shape of $k_{se}(x = 0, x')$ is loosely Gaussian and is visualized in Figure 3 below. Intuitively, the SE kernel is

continuous functional form. See Micchelli, Xu, and Zhang (2006)

[7] See Karvonen et al. (2020) for discussion of the asymptotic properties of estimating these parameters. Estimation of the scale parameter via maximum likelihood is likely to limit the influence of kernel misspecification and may at worst be ”slowly” overconfident. See Karvonen, Tronarp, and Särkkä (2019) for additional discussion regarding the length-scale parameter.

Figure 2: GP conditioned several specific points



appealing because it conveys the notion that we should be confident that $(f(x), f(x'))$ are similar where (x, x') are similar and much less certain about whether $(f(x), f(x'))$ are close when (x, x') are far apart.

The SE kernel is a special case of the Matern kernel function,

$$k_m(x, x') = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \Big(\sqrt{2\nu}\frac{||x - x'||_2^2}{\ell}\Big)^\nu \mathscr{K}_\nu\Big(\sqrt{2\nu}\frac{||x - x'||_2}{\ell}\Big) \tag{3}$$

where $\mathscr{K}$ is a modified Bessel function and ν is an additional parameter that governs the shape of k_m such that lower values of ν take on a more narrow shape (see figure 3). In practice, higher values of ν drive smoother functions in a fitted GP regression model. The Matern kernel converges to the SE kernel as ν approaches infinity. The Matern kernel is useful when the SE kernel seems to produce unrealistically smooth functions. As with ℓ and σ, the hyperparameter ν can be chosen via maximum likelihood or bayesian inference. However, in practice ν is often chosen to be one of $\{1/2, 3/2, 5/2\}$.

Both Matern and SE kernels contain a length-scale parameter ℓ. If we allow for ℓ to be a vector of equal dimensionality to the input space (i.e. equal in length to x), we rewrite our kernel function to accommodate,

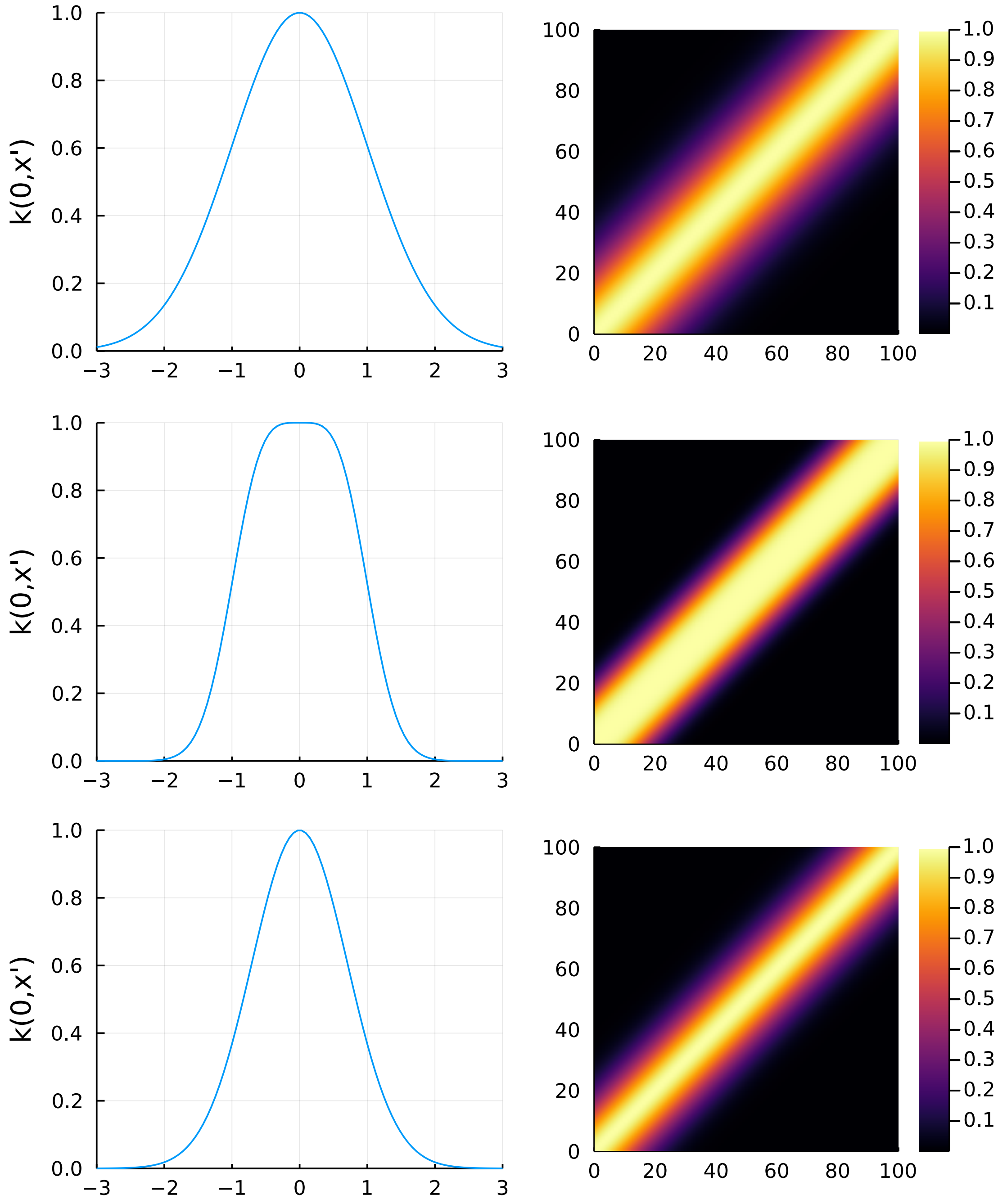
$$k_{se}(x, x') = \sigma^2 e^{-1/2(x-x')^\top (\mathbb{I}\ell)^{-2}(x-x')} \tag{4}$$

$$k_m(x, x') = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \Big(\sqrt{2\nu(x-x')^\top(\mathbb{I}\ell)^{-1}(x-x')}\Big)^\nu \mathscr{K}_\nu\Big(\sqrt{2\nu(x-x')^\top(\mathbb{I}\ell)^{-1}(x-x')}\Big) \tag{5}$$

where $\mathbb{I}$ is a $p \times p$ identity matrix. As written here, we can reconsider the choice over ℓ instead terms of a choice over inverse length scale, $1/\ell^2$. Doing this, we can see that, for a particular feature, p, as the corresponding element, $1/\ell_p^2$ tends towards zero, the contribution of p to $k(x, x')$ also goes to zero. Optimizing the choice of inverse length-scale thus acts as a form of implicit regularization on k, reducing or eliminating the influence of irrelevant features. Kernel functions constructed this way are, fittingly, called automatic relevance discovery[8] (ARD) kernels and are useful when working with a set of input features, P of which the model function of interest only uses some unknown combination of features, $P^* \subset P$.

[8] see Williams and Rasmussen (1998) as an example.

Figure 3: Squared Exponential, Matern (5/2,1/2) kernel comparison



3.1 Aggregation via PDP

To this point we have focused on why it is useful to look at individual counterfactuals when assessing bias. But describing the general (aggregate) behavior of a model can also be useful. Cook et al. (2021) explore ways to do this using partial dependency functions (Friedman, 2001). We briefly discuss here how GP regression can be used to give an estimate of the PDP when access to the model is not provided.

Because the GP regression gives us easy-to-use posterior distributions, we can calculate the pdp quite easily. In principal, the PDP is just the distribution of $y' = \phi_{x'}$ marginalizing over $x^{(\neg p)}$:

$$p(y'|x'^{(p)}) = \int p(y'|x'^{(p)}, x^{(\neg p)})p(x^{(\neg p)})dx^{(\neg p)} \tag{6}$$

To get a empirical estimate of the PDP, we leverage the fact that any finite set of $\phi_{x'}$ is multivariate normal. We can then estimate the PDP from a dataset (denoted $PDPGP$) as a weighted convolution over the counterfactual observations:

$$PDPGP(x'^{(p)} = q) \sim N(\mu'^{(q)}, \sigma^{2'(q)}) \tag{7}$$

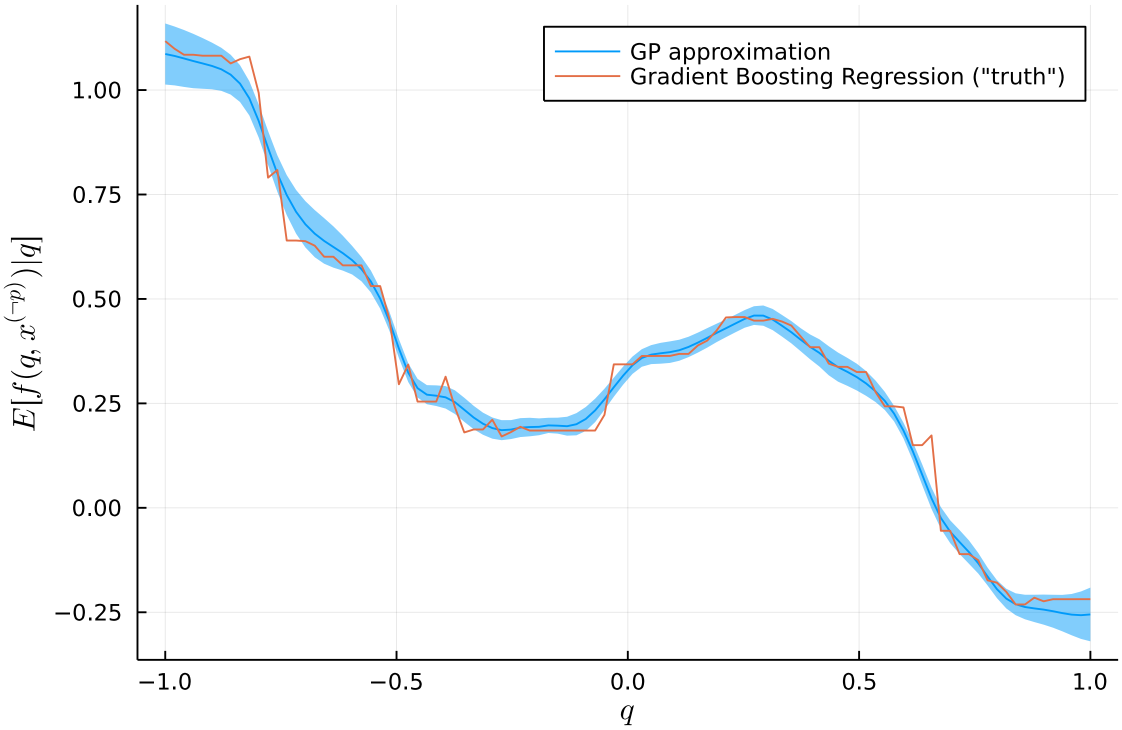
$$\mu'^{(q)} = 1/N \sum_i \mu_D(x'^{(p)} = q, x_i^{(\neg p)}) \tag{8}$$

$$\sigma^{2'(q)} = \frac{1}{N^2} \sum_i \sum_j K_D\big((\mathbf{x}'^{(p)} = q, \mathbf{x}^{(\neg p)})\big)_{ij} \tag{9}$$

We can use $\sigma^{2'(q)}$ to construct confidence intervals, but these require some nuanced interpretation. As constructed here, $\sigma^{2'(q)}$ quantifies uncertainty over the expected mean model prediction when p is held constant at $x'^{(p)} = q$. Specifically, it quantifies uncertainty that is driven by the variance/covariance in $\phi_{(x'^{(p)}=q, x^{(\neg p)})}$ that is in turn driven by both the variance/covariance in D as well as the distance of x' from points in D. Accordingly, $\sigma^{2'(q)}$ can be influenced by the choice of the counterfactual value (q), the observed (i.e. training) dataset D, and the values used for the non-counterfactual part[9] of x', i.e. $x'^{(\neg p)}$.

[9] to this point we have assumed that the non-counterfactual part of x' is the same as the training data – $x'^{(\neg p)} = x^{(\neg p)}$. One might imagine using a separate sample of the data to construct $x'^{(\neg p)}$, though this would likely raise issues in using x' for causal reasoning/inference.

Figure 4: Simulated example of a model (GBM) PDP and approximate Gaussian process PDP



For a given function, f, a given dataset, $D = \{X, f(X)\}$, and a fitted GP, $\Phi_{X'}|D$, the PDPGP will approach the PDP of f estimated from D as D increases in size. To see this, consider that the $\Phi_{X'}|D$ is a universal approximator[10] and that, given sufficient D, can learn to approximate f to an arbitrary degree of precision. Since the estimated PDP of f from D is merely the aggregated evaluations of f at various points, then the evaluation of an approximation of f over the same points should produce a similar aggregation.

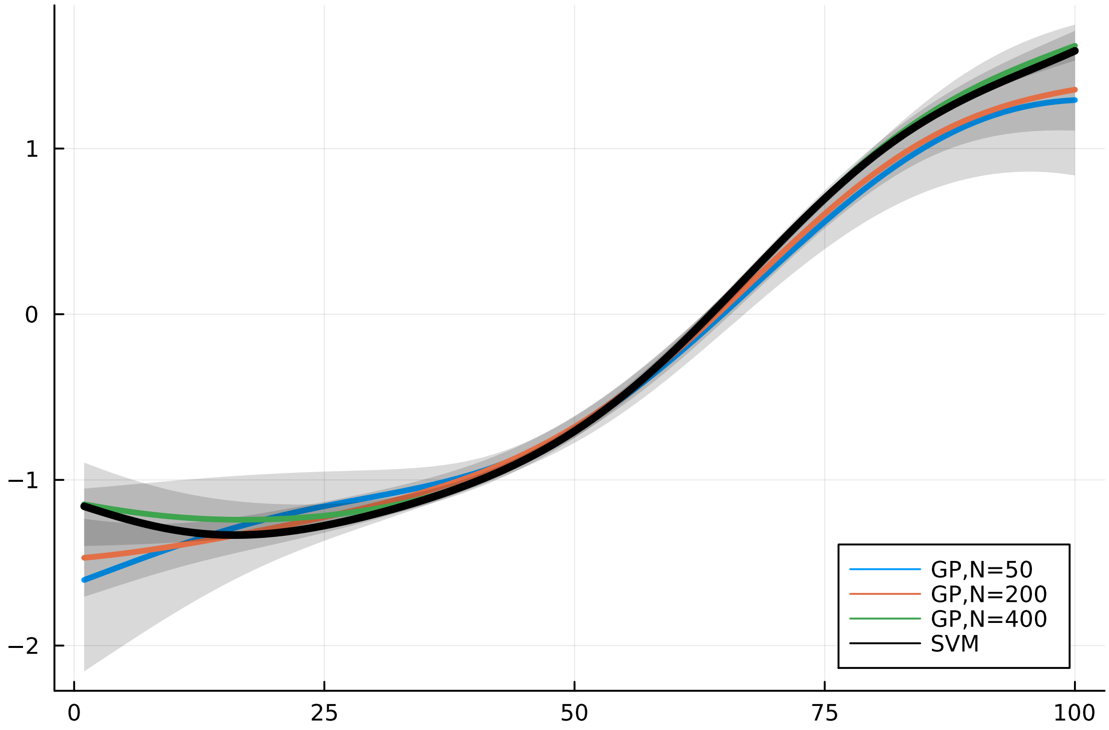
Figure 4 provides an example of a PDP as approximated by the PDPGP of a GP regression. In this figure, outcome values, $Y = (y_1, y_2 \dots y_N)$, from a randomly drawn function of several random input variables[11], $X = (x_1, x_2 \dots x_N)$. A gradient boosting model (GBM), f, was trained on $\{X, Y\}$, producing a dataset of model inputs and

[10] this may require some additional assumptions about the GP, namely that f falls within the RHKS associated to the kernel of the GP.

[11] Specifically, the outcomes are generated from 3 input features and a total of 500 observations.

model predictions, $D = \{X, f(X)\}$. A Gaussian process regression model was fit to D. This figure shows, for one of the input variables, p, the PDP of the GBM model (orange line) quantities of interest, q. This ("true") PDP is then approximated by the PDP from the fitted Gaussian process regression. The GP PDP generally tracks closely to the PDP from the GBM, and exhibits something of a smoothing effect in comparison to the more jagged shape of the GBM PDP. We can attribute this, in part to the choice of kernel[12], and more generally to the notion that Gaussian processes can be thought of as a linear smoothers (Williams and Rasmussen, 2006).

Figure 5: PDP-GP svm overlay



[12] The degree of smoothness is controlled by the choice of kernel. For Matern kernels, this is controlled largely by the choice of ν, with higher values corresponding to greater smoothness. See 3 for illustration. Further discussion is found in Schulz, Speekenbrink, and Krause (2018), section 3.

4 Simulation Exercise

An extensive literature has been published that establishes differences in lending outcomes based protected-class status (e.g. race, gender, etc.). Recent examples include Popick (2022) and Bhutta, Hizmo, and Ringo, 2021, both of which leverage new attributes available in the HMDA dataset. This is preceeded by Bartlett et al. (2022) that looks more narrowly at fintech lending. All studies find at least some disparity in lending outcomes, but each of these models is dedicated to detecting racial disparities in the aggregate, while controlling for potential confounds. Moreover, most studies impose some sort of (usually linear) functional form on their model of lending outcomes. This is a problematic approach when considering machine learning models or other models that are inherently non-linear as the nonlinearities may result in a masking of outcome disparities[13].

To highlight the usefulness of GP regression for detecting bias, we conduct a simulation exercise presented below. In this exercise, a simulated lender uses an ML model to assign interest rates to home mortgages. We induce bias over the ML model and examine the model, using a GP regression model to estimate counterfactual outcomes.

The goals of this simulation exercise are two fold – first, to demonstrate the ability of the GP to approximate the functional form of the lender's ML model sufficiently well as to be able to approximate its PDP (and thereby enable interpretation). Second, to demonstrate the ability of the GP to estimate counterfactuals sufficiently well to detect bias at levels similar to those commonly detected in the literature[14].

[13] We illustrate this in appendix A.2.

[14] For conventional purchase loans, Popick (2022) reports interest rate spreads of about 6 bp, controlling for other variables and as much as 13 basis points when not controlling for other variables. This is similar to interest rate differences in Bartlett et al. (2022)

4.1 Data description and Simulation Process

We begin by collecting a sample of home mortgage applications[15], $\tilde{D} = \{\tilde{X}, \tilde{y}\}$. Specifically, we collect loan to value (LTV) and debt to income (DTI) ratios along with credit score[16], race (white/non-white), gender, income, and age. The data we examine comes from the 10th district banks in the first quarter of 2019. Containing loans to this period and region helps limit the complexity of the DGP that we need to simulate – sampling from a broader sample of loans would not affect general conclusions of this exercise.

Following this, we fit a nonlinear, ML model to the data[17], $\tilde{f}$. The fitted ML model does not include protected status variables and represents the underwriting model of an unbiased lender[18].

Then, we simulate a dataset $D = \{X, y\}$ which consists of simulated inputs into the lending model and (biased) model outputs. The values of X are random draws from a distribution that matches the empirical distribution of $\tilde{X}$. We use X to simulate lending decisions such that $y = \tilde{f}(X) + B(X^{(w)})$, where $B(X^{(w)})$ is a function[19] that adds bias on the basis of a race indicator variable, w that returns 1 if a simulated applicant is white and 0 otherwise. By constructing the data this way, we can control the precise ammount of bias that is exhibited by the model which is helpful in assessing the performance of the GP regression.

For this exercise we will focus on a GP regression,f fitted to a small training set which is a subset of D. This GP model is equipped with a Matern 3/2 kernel with inverse length and scale parameters chosen by maximum likelihood estimation.

[15] Data comes from a merged dataset consisting of Black Knight McDash Data (MCDASH), Equifax Credit Risk Insight Servicing (CRISM), and The Home Mortgage Disclosure Act (HMDA). McDash and Equifax credit reporting information data is anonymized. The HMDA data is similarly anonymized.

[16] For credit score, we use the original form of the FICO® score, which ranges from 350-850.

[17] In the discussion below, this model is a support vector machine, though we have conducted the exercise with other types of ML models and found the GP regression to provide similar levels of performance.

[18] We examined a version of this exercise in which this unbiased model has access to a random data component that represents an alternative data source for the model to leverage for lending decisions with low credit-score borrowers, the results are consistent with the results shown here.

[19] $B(x_i^{(w)}) = x_i^{(w)} b$ for the indiscriminate bias scenario. $B(x_i^{(w)}) = x_i^{(w)} b(1 - \frac{x_i^{(\text{credit score})} - \min(\text{credit score})}{\max(\text{credit score}) - \min(\text{credit score})})$ for the moderated bias scenario. Where b is the size of the bias in basis points.

We explore varying sizes of training set to illustrate the impact of sample size on performance.

4.2 GP Performance and Discussion

The uncertainty over the PDP decreases and the MAP estimate generally becomes more accurate as the size of the training set grows. Figure 6 shows the $PDPGP$ approximation of the PDP of $\tilde{f}$ for applicant credit score Score. Note that even where the number of observations is few (100), the PDP of $\tilde{f}$ falls within the 95% credibility interval. As the number of observations increases, the size of this credibility interval becomes more narrow and the MAP of the $PDPGP$ becomes closer to the PDP of $\tilde{f}$. Figure 7 shows the $PDPGP$ approximation of the PDP of $\tilde{f}$ for other variables in the SVM model using a subset of 500 observations from D. They similarly demonstrate that the $PDPGP$ closely approximates the PDP from the SVM model. Summary performance statistics for all five variables are provided in Table 1.

Figure 6: Estimating the PDP of an SVM ($\tilde{f}(X)$) for Credit Score

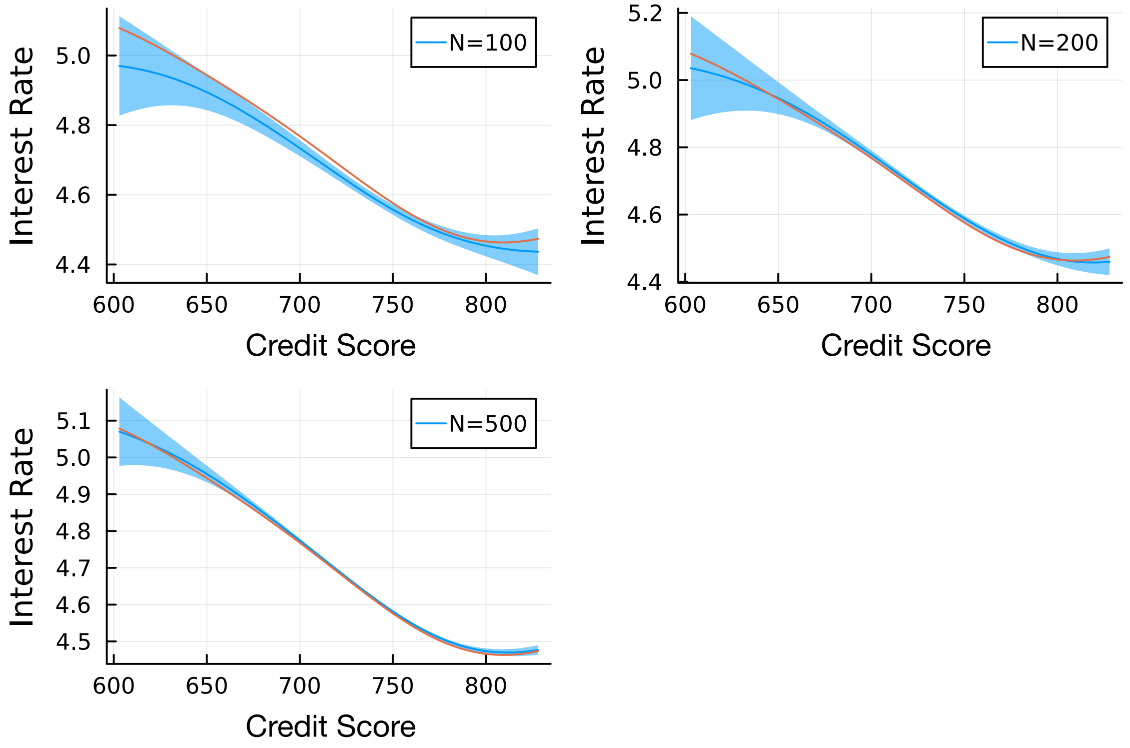
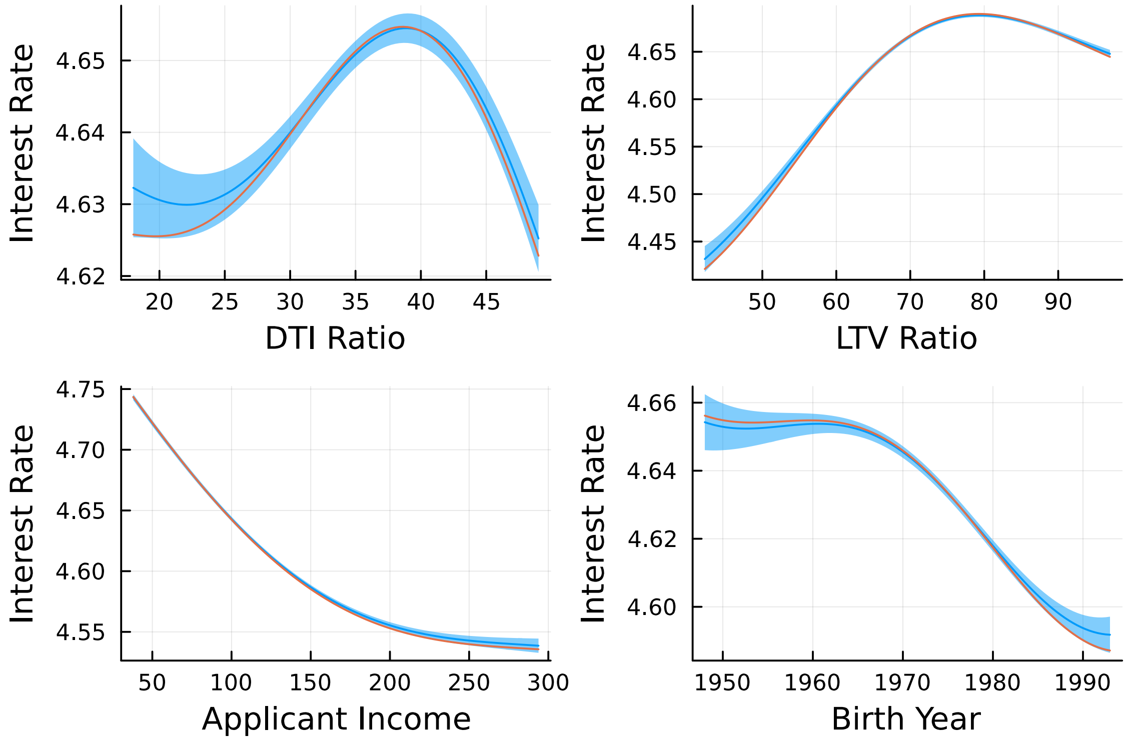


Figure 7: *PDPGP* and PDP of SVM for other Variables



PDPGP shown in blue, PDP of SVM in orange. Shaded regions reflect 95% credibility interval. *PDPGP* estimated from a sample size of N=500

Beyond replicating the PDP of $\tilde{f}$, the main purpose of the exercise is examine the GP regression ability to detect model bias. We induce bias in the simulated lender's ML model, $\tilde{f}$ on the basis of simulated applicant race (white, non-white). Non-white applicants in X are relatively rare, consisting of only about 8% of all simulated observations. This is due to the relative rarity of non-white applicants in the actual dataset, $\tilde{X}$. Nevertheless, the GP model is capable of capturing the presence of bias in the data.

We induce model bias in two different ways in this exercise: indiscriminate and moderated. In the indiscriminate bias implementation, the bias function applies equally to all non-white applicants, $B(X^{(w)}) = (1 - X^{(w)})b$ where b is the size of the bias effect in basis points. For the indiscriminate bias, we explore two levels of bias effect: a small effect at 5 basis points and a larger effect at 18 basis points. The small effect is roughly the size of the size of bias found in the recent literature. The

Table 1: RMSE comparison PDP to $PDPGP$

		RMSE	
DTI	0.0023	0.0119	0.0138
LTV	0.0047	0.0089	0.0054
Credit score	0.0022	0.0125	0.0077
Income	0.0020	0.0057	0.0172
Age	0.0017	0.0094	0.0080
	N=500	N=200	N=100

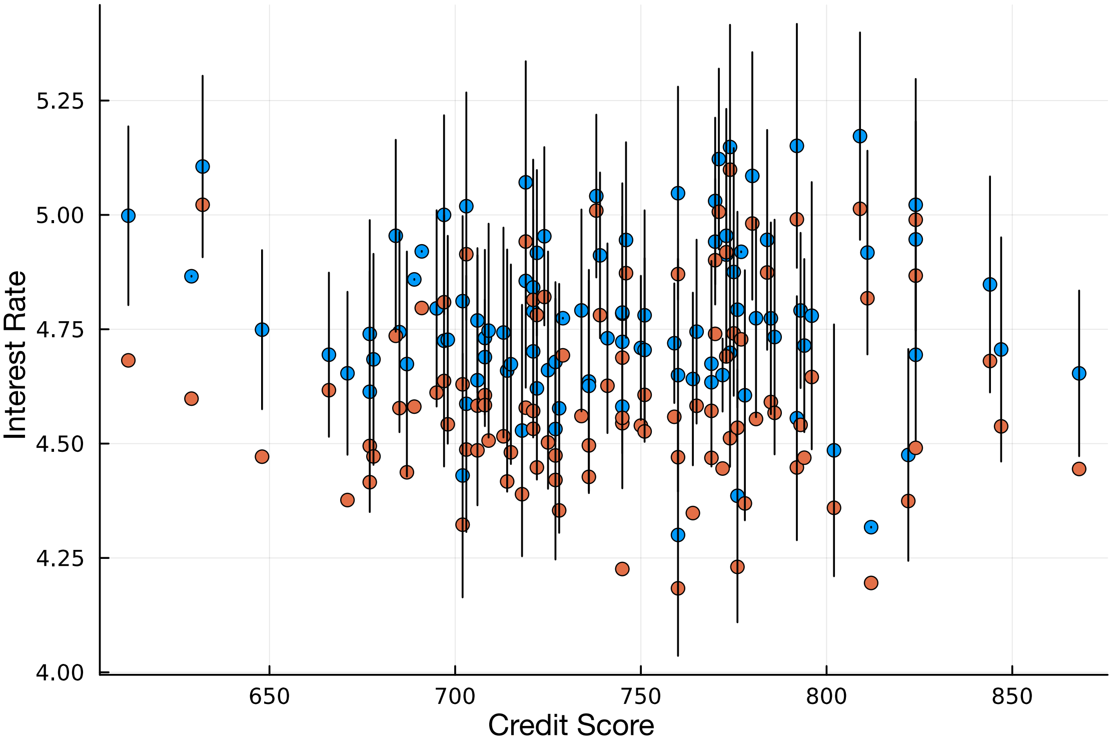
larger effect is roughly the size of the standard deviation of $\tilde{f}(X)$ and is only slightly larger than the size of the (13 bp) unconditional difference in interest rates for white and non-white applicants reported (Popick, 2022, pg 4).

We estimate the bias in the simulated data by comparing the counterfactual outcomes of each observation. We first fit a gaussian process regression to a sample of the data and generate estimates of the interest rate for each observation. We then use the model to predict interest rates for each observation under the counterfactual on the white/non-white indicator variable, w. we can examine the difference between factual and counterfactual outcomes to describe the size of the bias on a per-observation basis. Where the size of the uncertainty over the counterfactual outcome is large, we may consider the counterfactual as implausible or not otherwise well represented in the data. In these circumstances, increases in sample size can help reduce the uncertainty. Figure 8 shows this individual comparison for each observation in a small sample of 200 observations, where the true bias parameter is 18 basis points.

We can aggregate the differences between observed/counterfactual outcomes and compare to the true value of b. This gives us a sense of how well this technique captures the true level of bias induced in the data. Table 2 displays aggregate performance for the indiscriminate bias implementation. Performance is shown for GP regression models fit using varying sizes of sample observations. For each sample size, the aggregated (mean) estimated bias is shown for 5 and 18 basis point values of b As the sample observations grow larger, the detected bias more closely resembles the true value of the bias parameter, b.

In addition to the indiscriminate bias, we also consider a form of bias that is

Figure 8: Counterfacutal Comparision of white and non-white applicants, N=200, b=18



moderated, or scaled, by another attribute. In this scenario, the bias function is implemented as $B(x_i^{(w)}) = x_i^{(w)} b(1 - \frac{x_i^{(\text{credit score})} - \min(\text{credit score})}{\max(\text{credit score}) - \min(\text{credit score})})$. Using this bias function, the extent of the disparity in interest rates between white and non-white applicants decreases linearly in (scaled) credit score with a slope of b. We examine this scenario under low (10), medium (20) and high (50) values for b.

Figure 9 illustrates the estimated counterfactuals and its effect. The difference in interest rate between the observed and counterfactual observations are shown. A black line provides a linear fit of these observations on credit score. In Figure 9 we can see how the GP model estimates of counterfactual outcomes changes (improves) with the sample size. We can also note from this the changes in uncertainty from one observation/counerfactual to the next. Even when the GP model is fit to a sample size of 500, there are many instances where the model exhibits substantial uncertainty over the counterfactual outcome. For the regions where this uncertainty remains, even as sample size increases, it may be the case that the counterfactual is implausible (i.e.

Table 2: bias detection performance

		GP	reference
N=100	small	6.8	5
	large	8.5	18
N=200	small	9.4	5
	large	21.8	18
N=500	small	4.3	5
	large	19.2	18

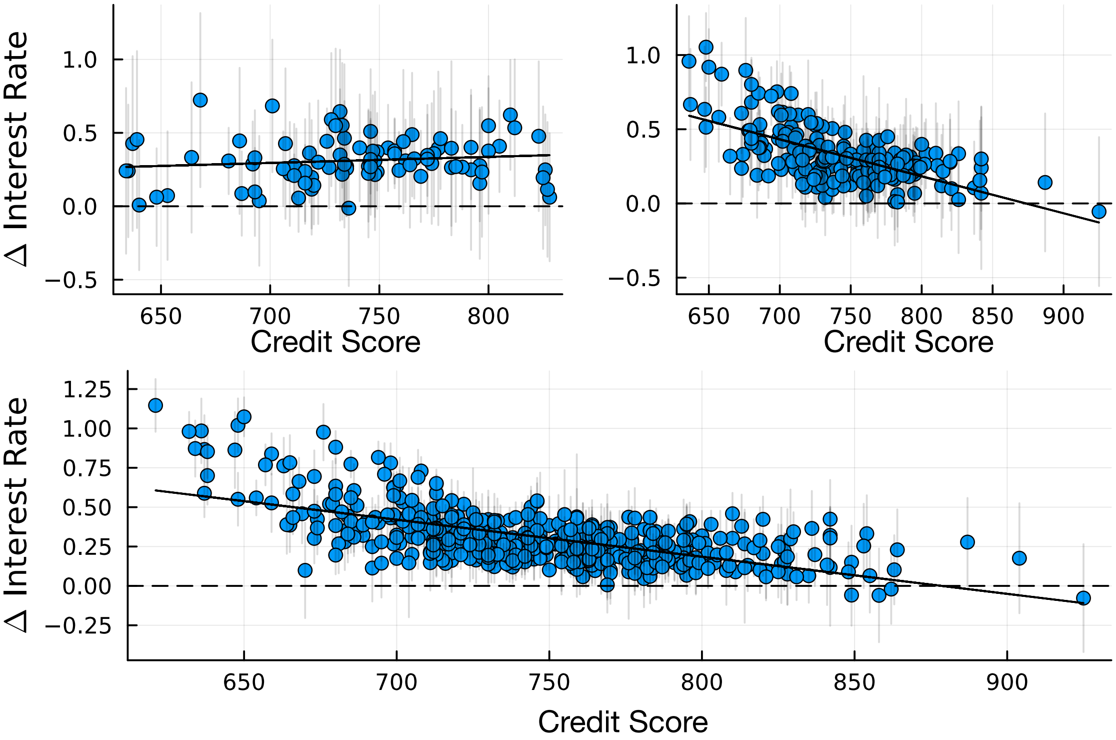
substantially different from the underlying distribution of data used to fit the *true* model, $\tilde{f}$).

If the GP regression does a good job of capturing the moderated bias, then the individual counterfactual predictions from the GP will demonstrate substantial differences between observed/counterfactual predictions where credit score is low, and small differences where credit score is high. Moreover a linear fit of these differences on credit score will exhibit a slope close to b. Table 3 shows the coefficients for a linear fit of differences between observed and predicted counterfactual outcomes on credit score. Where the number of observations is low (100) the GP model is not able to accurately capture the moderated bias, but its performance improves substantially with only moderate increases in the number of observations. The number of observations needed to capture the moderated bias depends on the bias size; the GP model converges to the true value of b more quickly when b is large.

Table 3: GP regression counterfactuals capture the moderated bias effect

N=100	3.6	7.9	8.6
N=200	-29.5	-11.0	-58.4
N=500	-19.6	-28.3	-54.9
Reference	-10	-20	-50

Figure 9: GP Regression counterfactual increase in interest rate for large bias moderated by credit score Score



Y-axis indicates increase in interest rate when applicant is changed from white to non-white. Each panel shows a GP Regression model trained on a different number of observations: (clockwise from top left) 100, 200, 500

5 Additional Considerations and Conclusion

The use of GP regression should work generally across a wide variety of nonlinear models. Its use as described in this paper is subject to a number of important limitations. First, while the ability to approximate the underlying model will generally improve with the number of observations, the computational cost of fitting a GP regression model increases with the number of observations in $O(N^3)$ Williams and Rasmussen (2006, p. 171). Most of this computational cost carries through to inference as the majority of the computational complexity consists of inverting (Σ^X) as in equation 1.

Subsampling the available data can help to reduce the computational cost of model fitting and inference. The key concern when subsampling from available data is to achieve a sample that appropriately covers the domain of interest ($\mathscr{X}$). Random sampling can achieve this, but is likely inefficient as it will not sufficiently sample from the edges of the area of interest and will likely produce more samples than necessary from the center of the distribution. Latin hypercube sampling is one popular approach to more strategically subsampling from a dataset (see Gramacy, 2020, Chapter 4). More specific to GP regression, *inducing point* methods have been proposed to approximation methods can reduce this time. The idea behind these methods is to choose a specific set of points, $m \in \mathscr{X}$ that adequately cover the region of interest, producing a smaller kernel matrix, and reducing computational time from $O(N^3)$ to $O(MN^2)$(Quiñonero-Candela and Rasmussen, 2005, see) with a popular implementation in Wilson and Nickisch (2015).

Perhaps of greater concern than the computational cost is whether the GP model converges to an accurate approximation of the underlying model, f. Wynne, Briol, and Girolami (2021) helps to establish when this should occur and provides advice to encourage convergence. Wang and Jing (2021) provides additional discussion about the convergence of estimates of the smoothness parameter, ν and gives an understanding of convergence for estimating smoothness parameter in terms of number of required observations

Lastly, the discussion provided in this paper is largely focused scenarios regarding on continuous, scalar outcomes. The approach can be applied to other types or problems such as discrete outcomes or classification problems. Doing this, however, requires some additional modeling and inference becomes more complicated. Williams and Rasmussen (2006) provide a very detailed treatment of this topic.

To conclude, understanding models is hard. even linear models, if complicated enough can be difficult to understand. Counterfactual reasoning is a useful framework for interpreting a model's behavior. The approach we have outlined in this paper hopefully allows researchers to combine counterfactual reasoning with model interpretation, even when direct access to the model is not available. In this paper we have focused on model discrimination as an illustrative example of this approach. But the approach can be used much more generally to understand any black-box process and to hypothesize about counterfactual outcomes. Application of the GP regression for counterfactual reasoning should also be considered where it is costly or otherwise difficult to directly calculate the PDP of a model. As we have seen here, the $PDPGP$ can quite closely approximate the PDP of a model even if only a relatively small sample of data is available.

References

Angwin, Julia et al. (2016). *Machine Bias.* Tech. rep. ProPublica. URL: `https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing`.

Athey, Susan and Guido Imbens (2019). "Machine learning methods economists should know about". *arXiv preprint arXiv:1903.10075.*

Bartlett, Robert et al. (2022). "Consumer-lending discrimination in the FinTech era". *Journal of Financial Economics* 143.1, pp. 30–56.

Bertrand, Marianne and Esther Duflo (2017). "Field experiments on discrimination". *Handbook of economic field experiments* 1, pp. 309–393.

Bertrand, Marianne and Sendhil Mullainathan (2004). "Are Emily and Greg more employable than Lakisha and Jamal? A field experiment on labor market discrimination". *American economic review* 94.4, pp. 991–1013.

Bhutta, Neil, Aurel Hizmo, and Daniel Ringo (2021). "How much does racial bias affect mortgage lending? Evidence from human and algorithmic credit decisions". *Evidence from Human and Algorithmic Credit Decisions (July 15, 2021).*

Buolamwini, Joy and Timnit Gebru (2018). "Gender shades: Intersectional accuracy disparities in commercial gender classification". In: *Conference on fairness, accountability and transparency.* PMLR, pp. 77–91.

Cook, Thomas R et al. (2021). *Explaining Machine Learning by Bootstrapping Partial Dependence Functions and Shapley Values.* Tech. rep.

Corbett-Davies, Sam and Sharad Goel (2018). "The Measure and Mismeasure of Fairness: A Critical Review of Fair Machine Learning". *CoRR* abs/1808.00023. arXiv: `1808.00023`. URL: `http://arxiv.org/abs/1808.00023`.

Friedman, Jerome H (2001). "Greedy function approximation: a gradient boosting machine". *Annals of statistics*, pp. 1189–1232.

Funke, Michael and Marc Gronwald (2009). "A Convex hull approach to counterfactual analysis of trade openness and growth".

Gramacy, Robert B. (2020). *Surrogates: Gaussian Process Modeling, Design and Optimization for the Applied Sciences.* `http://bobby.gramacy.com/surrogates/`. Boca Raton, Florida: Chapman Hall/CRC.

Jung, Jongbin et al. (2018). *Omitted and Included Variable Bias in Tests for Disparate Impact.* DOI: 10.48550/ARXIV.1809.05651. URL: https://arxiv.org/abs/1809.05651.

Karvonen, Toni, Filip Tronarp, and Simo Särkkä (2019). "Asymptotics of maximum likelihood parameter estimates for gaussian processes: The ornstein–uhlenbeck prior". In: *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP).* IEEE, pp. 1–6.

Karvonen, Toni et al. (2020). *Maximum likelihood estimation and uncertainty quantification for Gaussian process approximation of deterministic functions.* DOI: 10.48550/ARXIV.2001.10965. URL: https://arxiv.org/abs/2001.10965.

King, Gary, Robert O Keohane, and Sidney Verba (1994). *Designing social inquiry.* Princeton university press.

King, Gary and Langche Zeng (2006). "The dangers of extreme counterfactuals". *Political analysis* 14.2, pp. 131–159.

Koenecke, Allison et al. (2020). "Racial disparities in automated speech recognition". *Proceedings of the National Academy of Sciences* 117.14, pp. 7684–7689.

Micchelli, Charles A, Yuesheng Xu, and Haizhang Zhang (2006). "Universal Kernels." *Journal of Machine Learning Research* 7.12.

Oaxaca, Ronald (1973). "Male-female wage differentials in urban labor markets". *International economic review*, pp. 693–709.

Pearl, Judea (2009). "Causal inference in statistics: An overview". *Statistics surveys* 3, pp. 96–146.

Pierson, Emma, Sam Corbett-Davies, and Sharad Goel (2018). "Fast threshold tests for detecting discrimination". In: *International conference on artificial intelligence and statistics.* PMLR, pp. 96–105.

Popick, Stephen (2022). "Did Minority Applicants Experience Worse Lending Outcomes in the Mortgage Market? A Study Using 2020 Expanded HMDA Data". *FDIC Center for Financial Research Paper* 2022-05.

Quiñonero-Candela, Joaquin and Carl Edward Rasmussen (2005). "A Unifying View of Sparse Approximate Gaussian Process Regression". *Journal of Machine Learning Research* 6.65, pp. 1939–1959. URL: http://jmlr.org/papers/v6/quinonero-candela05a.html.

Rubin, Donald B (1978). "Bayesian inference for causal effects: The role of randomization". *The Annals of statistics*, pp. 34–58.

Schulz, Eric, Maarten Speekenbrink, and Andreas Krause (2018). "A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions". *Journal of Mathematical Psychology* 85, pp. 1–16.

Wang, Hao, Berk Ustun, and Flavio Calmon (2019). "Repairing without retraining: Avoiding disparate impact with counterfactual distributions". In: *International Conference on Machine Learning*. PMLR, pp. 6618–6627. URL: `http://proceedings.mlr.press/v97/wang19l/wang19l.pdf`.

Wang, Wenjia and Bing-Yi Jing (2021). *Convergence of Gaussian process regression: Optimality, robustness, and relationship with kernel ridge regression.* DOI: `10.48550/ARXIV.2104.09778`. URL: `https://arxiv.org/abs/2104.09778`.

Williams, Christopher and Carl Rasmussen (1998). "Gaussian Processes for Regression." In: *Advances in Neural Information Processing Systems*. Ed. by D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo. Vol. 8. MIT Press, pp. 514–520.

Williams, Christopher KI and Carl Edward Rasmussen (2006). *Gaussian processes for machine learning*. Vol. 2. 3. MIT press Cambridge, MA.

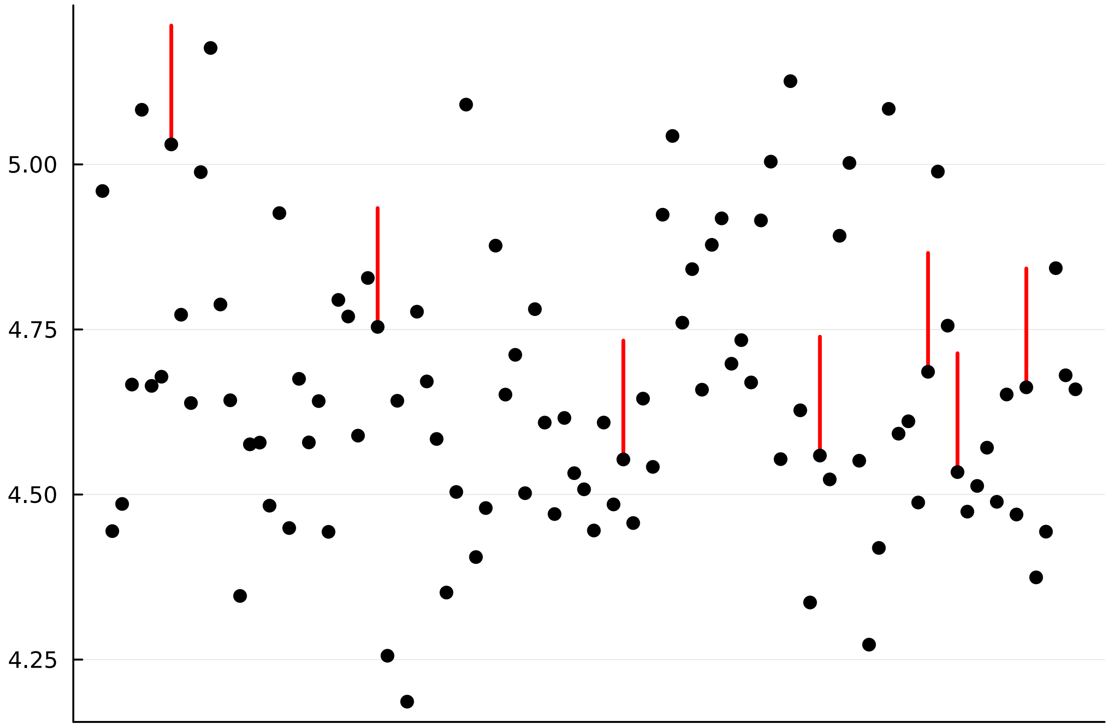
Wilson, Andrew and Hannes Nickisch (2015). "Kernel interpolation for scalable structured Gaussian processes (KISS-GP)". In: *International conference on machine learning*. PMLR, pp. 1775–1784.

Wynne, George, François-Xavier Briol, and Mark Girolami (2021). "Convergence guarantees for Gaussian process means with misspecified likelihoods and smoothness". *The Journal of Machine Learning Research* 22.1, pp. 5468–5507.

A Appendix

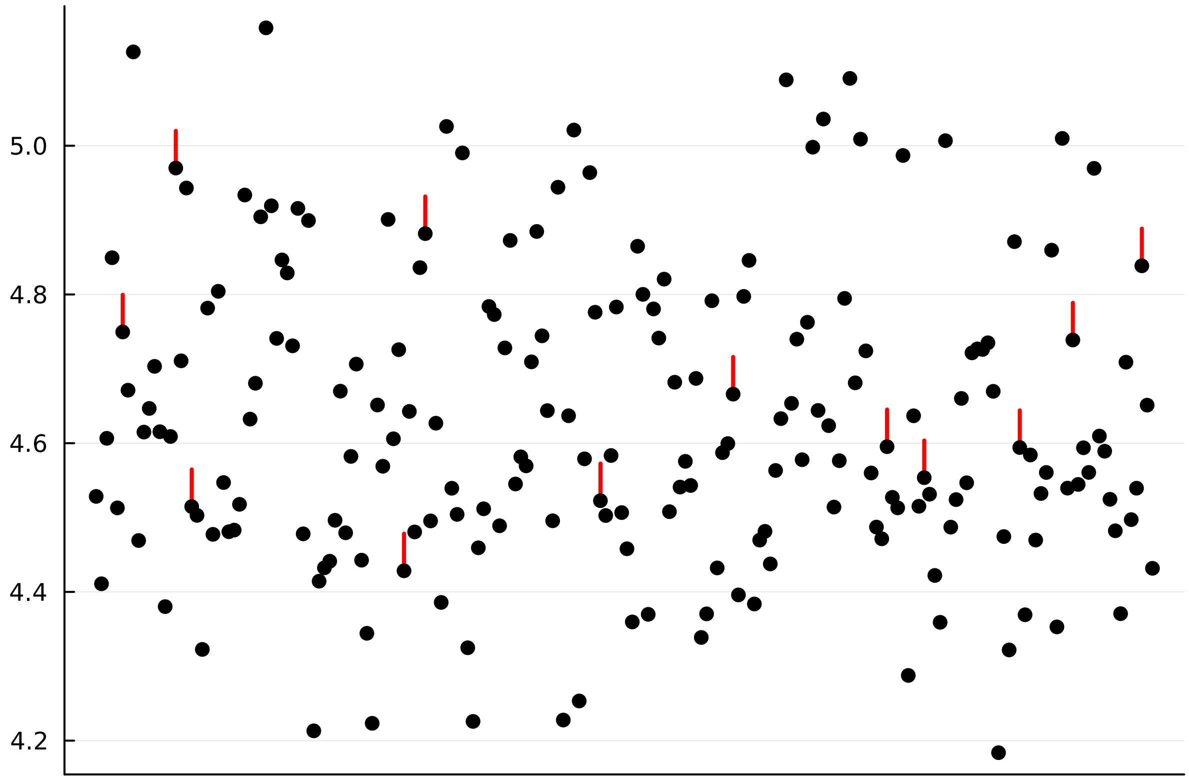
A.1 Additional figures

Figure 10: Implementation of indescriminate bias of 18 bp, N=100



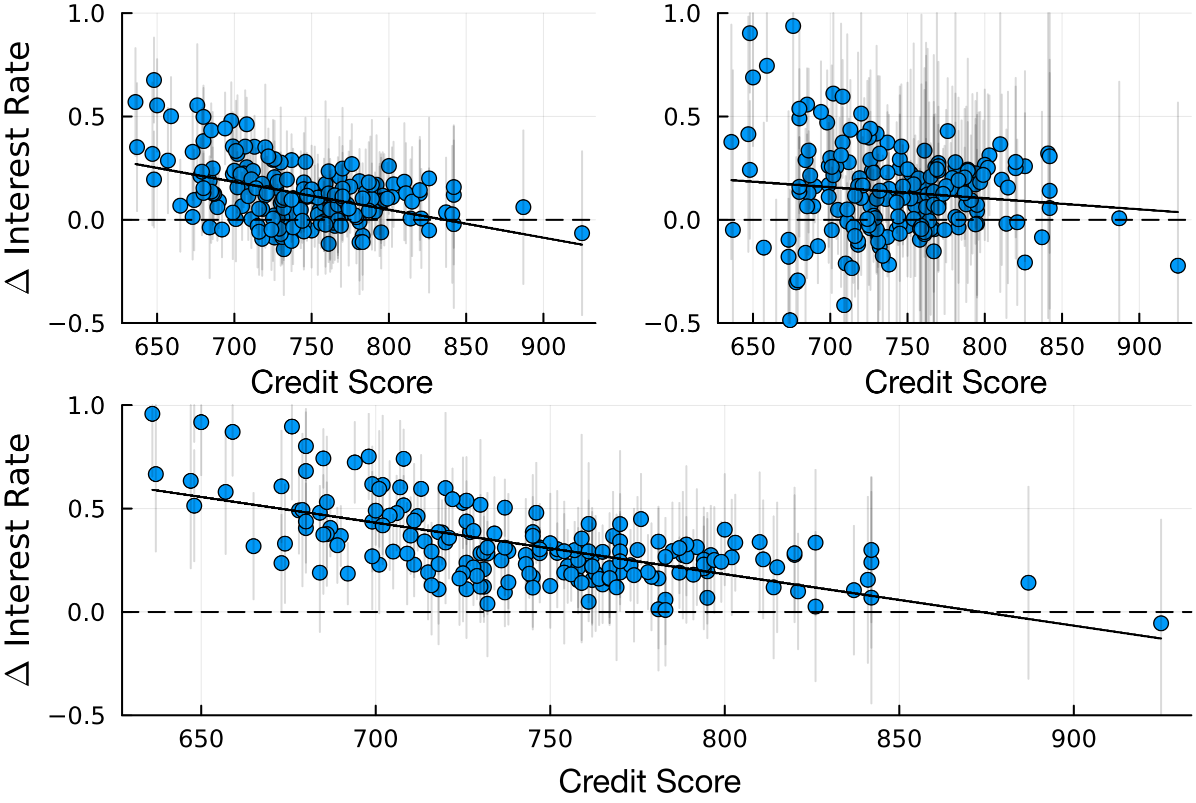
This figure portrays the implementation of an 18bp bias effect. Round dots in the figure show the interest rate that would be assigned to applicants if they were white. Red lines are attached to each non-white observation and indicate the bias effect. The red lines terminate at the interest rate assigned to those observations. The size of the sample shown is N=100. This figure highlights the relative rarity with which bias occurs within the sample provided to the GP model.

Figure 11: Implementation of simple bias, 5 basis points



This figure portrays the implementation of bias as in Figure 10, but for a 5 basis point level of bias, and a sample size N=200.

Figure 12: GP regression counterfacutal increase in interest rate at Varying levels of bias moderated by Credit Score



Y-axis indicates increase in interest rate when applicant is changed from white to non-white. Each panel shows a GP Regression model trained on 200 observations at different levels of bias: (clockwise from top left) 10, 20, 50 basis points.

A.2 Detecting Bias with Oaxaca Blinder Decomposition

The Oaxaca-Blinder decomposition (Oaxaca, 1973) is a commonly used method for detecting bias at an aggregate level. The approach relies upon generating split-sample linear estimates over the dimension of discrimination. In the case of the exercise we undertake in this paper, this would mean separate estimates for $w = 1$ and $w = 0$. Let the subscript of 0 denote nonwhite and 1 denote white. Then let $D_0 = f(X_0), X_0$ and $D_1 = f(X_1), X_1$ and specify linear relationships between $f(X)$ and X:

$$\begin{aligned} f(X_0) &= X_0 B_0 + U_0 \\ f(X_1) &= X_1 B_1 + U_1 \end{aligned} \tag{10}$$

where B_0 and B_1 are vectors of coefficients and U_0 and U_1 are vectors of residual terms. Approximating B_0 and B_1 via OLS as b_0 and b_1, the Oaxaca-Blidner decomposition of Equation (10) is

$$\text{mean}(f(X_0) - f(X_1)) = \underbrace{(b_1 - b_0)}_{\text{unexplained effect}} X_0 + b_0 \underbrace{(X_1 - X_0)}_{\text{endowment effect}} \tag{11}$$

In a correctly specified model, the unexplained effect will capture the level of discrimination. Table 4 reports the unexplained effect for the moderated bias simulation exercise. If the decomposition were to appropriately capture disscrimination, then we would expect the estimated unexplained effect to be statistically significant and range from 0.03 where $b = 10$ to 0.15 where $b = 50$. The results reported in the table are substantially biased downward and the decomposition only reveals a statistically significant unexplained effect when the true bias effect size and sample size are at their largest (50 and 500 respectively). The failure of the decomposition to portray the discrimination effect is unsuprising and is a direct consequence of the fact that the nature underlying data generating process is inherently nonlinear.

Table 4: Oaxaca-Blinder decomposition of moderated bias

	Unexplained Effect ($b_1^{(\text{credit score})} - b_0^{(\text{credit score})}$)		
Bias effect	N=100	N=200	N=500
10	0.001	-0.0	0.001
	(0.305)	(0.58)	(0.101)
20	0.002	0.0	0.001
	(0.269)	(0.316)	(0.061)
50	0.003	0.001	0.002
	(0.197)	(0.098)	(0.028)

Values reported are the estimated Credit Score coefficient differences from a Oaxaca-Blinder decomposition. P-values in parentheses. $*$ indicates $p < 0.05$
.