How to write a *Technometrics* paper

Robert B. Gramacy*

May 22, 2023

Abstract

This is an unofficial template for, and guide to, writing a *Technometrics* paper. The instructions here should serve you well for any methodology-focused paper, but are biased toward elements that *Tech* values highly: light theory, strong intuition/illustration, and empirics. I'll try to make suggestions for variations that are common. The emphasis here is on creative aspects, but I even get into nitty-gritty like typesetting. I'm not using the official, recommended style file, however the style I am using is similar. I strongly suggest working with 11-12pt font and ordinary, one-inch margins. Never allow anything (math, figures, prose) to run into the margins. *Tech* prefers double-spaced submissions to ease review. I have not done that here. This document is more like a version of the paper that I'd put on arXiv. I often joke that I can spot a *Tech* paper from ten feet away. If I've done a good job here, you won't be able to tell from ten feet away that this is *not* a *Tech* paper.

Keywords: choose 3–6 words which are not in your title that best describe your contribution

1 Introduction

An introduction serves to motivate a need for the methodology proposed in the main body of the paper. Don't begin here by pasting from your Abstract, or from anywhere else in the document. If the paper is application/case study-focused, then the Intro should introduce the case. Often an Intro must do both: survey an application area to motivate a novel method whose main ideas are immediately outlined. Even if you only have a single application in mind, make sure to connect to a wider class of problems. Likely your reader will only care about your application in-so-far-as it might seem similar to one they're working on. For *Technometrics*, that class of problems should be well-aligned with those listed on the Aims & Scope page.[1][2] I almost always have no sub-sections in the introduction section. Keep it simple. You don't want to lose your reader on the first page.

This is a good – but certainly not the only – place to provide a literature review, covering the state-of-the-art in order to establish a pedigree to frame your contributions. Cite (who

*Corresponding author: Department of Statistics, Virginia Tech, rbg@vt.edu

[1] https://www.tandfonline.com/action/journalInformation?show=aimsScope&journalCode=utch20

[2] Note that American Statistical Associate (ASA) style discourages footnotes, but they are technically allowed as long as they're short and don't have math in them.

you want as) friends here, and be nice to them. Always be nice. Many Associate Editors (AEs) will choose at least one referee from your bibliography. For example, this tutorial was inspired by a similar one for INFORMS' *Journal on Data Science* (Shmueli, 2021), and by the instructions for authors exemplifying the *Biometrika* style (Fearnhead et al., 2021).

The most important thing, for your Intro, is to extensively foreshadow later development. Be clear about what's novel in your contribution. You're not writing a mystery; suspense is *not* an appropriate presentation device for a research paper. Let your reader know what you'll do and why up front, leaving only details and nuance for later. Provide evidence throughout. Sometimes evidence is theoretical, sometimes empirical, sometimes an appeal to intuition, sometimes part of the canon in the literature. The Intro should focus on intuition, supported where possible by references to other literary sources. Make it clear what's what: which ideas are your insights; and which have been suggested by others.

Cite liberally. This goes for the whole paper, but especially here and in Section 2. It's extremely rare that a page of text would go by without any citations, except when the page is filled with math or figures. A typical page of text may have 3–4 cites; some review paragraphs may have at least one per sentence. It's common for bibliographies to have many more than thirty citations, spanning several pages. Ensure that yours is diverse. Even though I use several of my own papers as examples here (Sauer et al., 2022, 2023), the bibliography spreads love much farther afield. Be careful not to give the impression that your work is old-fashioned by citing too many "old" papers, and not enough "new" ones, besides your own recent work. There's almost no harm in being over-inclusive. *Technometrics*, like most journals, doesn't count the reference list when measuring up against page budgets. Make sure you're using the right citation format. If it's not fully parenthetical (those in this paragraph, via `\citep` in LaTeX), like the Shmueli (2021) reference at the end of paragraph two above (via `\citet`), make sure it reads grammatically as a proper noun in your sentence.

About grammar: take advantage of writing tools. This goes for everyone, not just non-native English speakers. Google provides free grammar and spell checking in their Docs/Gmail environments. There are many other resources out there. Typos happen, and it's not a big deal. But things become problematic when the same mistakes are made over and over again. Most good referees will work hard to overlook grammatical issues to focus on the science. But even the best ones are only human. It's easy to be distracted by poor writing when reading a lengthy paper. Some issues won't be caught by grammar checkers, like out-of-order logic or holes in a narrative development. It can really help to have someone "referee" your paper before it goes to the journal. They should be an expert in the field, but not someone who is intimately familiar with your recent work.

Keep in mind that the message in this document is entirely my opinion. There's nothing official about it. I'm sure many of my colleagues would disagree with at least some of the recommendations here, though I doubt they'd reject it's spirit wholesale. I hope this will be useful for young researchers, and those new to *Technometrics*. Even when I'm not sending the paper to *Tech*, I generally follow the scheme described herein. Once I'm done, and have perhaps decided on another outlet, I might or might not make adjustments to suit.

I usually end the Introduction with a roadmap for the rest of the paper, outlining what the reader can expect to find in each subsequent section. Below, I shall outline sections in the order that I typically have them in my *Technometrics* papers. I'll explain what goes in each, how things are laid out therein, and set expectations about flow. This document is clearly different, because I'm not disseminating my research, but rather attempting to give my opinion in format of a research paper. So I'll be taking some liberties. In an effort to hold my reader's attention by entertaining, the writing style here is less formal than for an ordinary research paper. Nevertheless, my message is serious. Often I'll make commentary, like the paragraph on grammar above, which is relevant to other sections as well. Page-length requirements will usually dictate that some material is relegated to an online supplement. These material include proofs of theorems, auxiliary experiments, etc. Don't forget to mention the supplement in your roadmap.

The remainder of the paper is outlined as follows. Section 2 describes how to review relevant background, mixing examples with advice. Section 3 provides suggestions for how to introduce novel methodological aspects in the paper, again mixing examples and advice. Section 4 is labeled for the development of a second, complementary method. But it's really about other topics including typesetting, technical and writing suggestions. Section 5 discusses the delivery of implementation details, including reproducibility and benchmarking on synthetic examples. Validation on a motivating example, case study, or other "real data" might be fleshed out in a separate Section 6. Finally, I'll conclude the tutorial with a brief discussion in Section 7. An electronic supplement, which at the time of submission (and in this example), is located as an appendix after the references, describes how I would write a rebuttal when a revision is encouraged.

2 Review of basic elements

This is where you set notation, and introduce the basics required to jump into your methodological contribution in Section 3 and beyond. Always outline what you're going to do before you do it. I think it's tacky to start a section without any prose before immediately jumping to a sub-section. Be sure to use appropriately named sub-sections to systematically iterate over relevant material. Don't allow any section, or sub-section heading to run over onto a second line. Be brief with your headings; they are phrases, not sentences.

2.1 First topic

Start foundationally. You may find that you are duplicating earlier references. That's fine. Try to be thorough. This is as much about introducing notation, and being concrete about motivation, as it is about providing background. Aim for completeness, but also be careful not to introduce any quantities that you don't need later. You can assume your reader is familiar with the subject area, but perhaps is a little rusty, or might not notate things the same way you do. Take the opportunity to foreshadow later developments, point out flaws that will be addressed in due course, etc. You'll probably need to provide some math here, especially if your methodological innovations later require math.

For example, many of my papers are on Gaussian process (GP) surrogate modeling of simulation experiments. I might write the following. Let $f : \mathbb{R}^d \to \mathbb{R}$ denote a computer model simulation with d real-valued inputs and a scalar output. If f is expensive to evaluate, then one may conduct an experiment where n inputs and outputs are collected for the purposes of fitting a so-called *surrogate* model (Gramacy, 2020). Let $\mathbf{X}_n$ denote an $n \times d$ training design of inputs and $\mathbf{Y}_n = f(\mathbf{X}_n)$ denote the corresponding simulation outputs. The canonical GP surrogate assumes a multivariate normal distribution (MVN) over $\mathbf{Y}_n$,

$$\mathbf{Y}_n \sim \mathcal{N}_n\left(\boldsymbol{\mu}, \Sigma(\mathbf{X}_n)\right). \tag{1}$$

To streamline notation, denote $\mathbf{\Sigma}_n = \Sigma(\mathbf{X}_n)$. When $\Sigma(\cdot)$ is based only on pairwise distances in its argument(s), the GP is said to be *stationary.* The mean $\boldsymbol{\mu}$ may be linear in columns of $\mathbf{X}_n$, but $\boldsymbol{\mu} = \mathbf{0}$ is often sufficient after centering (Santner et al., 2018). Given these data $\mathbf{D}_n = (X_n, Y_n)$, the predictive distribution for an $n' \times d$ matrix of testing locations $\mathcal{X}$ has a closed form under multivariate normal conditioning.[3]

$$Y(\mathcal{X}) \mid \mathbf{D}_n \sim \mathcal{N}_{n'}\left(\mu_Y(\mathcal{X}), \Sigma_Y(\mathcal{X})\right), \quad \text{where} \quad \begin{aligned} \mu_Y(\mathcal{X}) &= \Sigma(\mathcal{X}, \mathbf{X}_n)\mathbf{\Sigma}_n^{-1}\mathbf{Y}_n \\ \Sigma_Y(\mathcal{X}) &= \Sigma(\mathcal{X}) - \Sigma(\mathcal{X}, \mathbf{X}_n)\mathbf{\Sigma}_n^{-1}\Sigma(\mathbf{X}_n, \mathcal{X}) \end{aligned}$$

Above, $\Sigma(\mathbf{X}_n, \mathcal{X})$ is an $n \times n'$ matrix obtained by extending $\Sigma(\cdot)$ from Eq. (1) to $\Sigma(\cdot, \cdot)$ so that it may be applied across training and testing elements, $\mathbf{X}$ and $\mathcal{X}$ respectively.

There are a few important points of note demonstrated in the previous paragraph. Observe that I only number the equations that are actually referred to later. I use bold for matrices and vectors. Actually, I prefer not to use bold in general, but it is required for *Technometrics.* Notice that my equations are aligned, and everything is tidy and makes efficient use of space. This is important. If I didn't need to refer to Eq. (1) later, then inlining as $\mathbf{Y}_n \sim \mathcal{N}_n\left(\boldsymbol{\mu}, \Sigma(\mathbf{X}_n)\right)$ would save space.

I find it helpful to pause for an illustration at some point in the review. You don't want your reader to wait until the empirical work in Section 5 for visual stimulation. I'll often make a figure that contrasts the state-of-the-art with my proposed methodology, and place it here rather than later in the paper. It's true that I haven't introduced those new ideas yet, but that's ok. You can tell your reader that the details will come later. They won't be able to help but look at the whole figure and get excited.

Figure 1 is from Sauer et al. (2022). Section 2 in that paper introduces two paradigms for GP regression (first two panels). The paper is motivating the need for something fancier, a deep Gaussian process (DGP)[4] in the final panel, but this is not introduced until Section 3. Make sure that your reader knows what you want them to see in the figure by explaining it to them in the main body of text, not in the figure caption. Use the figure caption as a heading, and to explain anything that's missing or left ambiguous by the legend. Ideally, the reader wouldn't need to see the figure to appreciate what it reveals. The opposite is also

[3] `https://en.wikipedia.org/wiki/Multivariate_normal_distribution#Conditional_distributions`

[4] Introduce all acronyms before you use them.

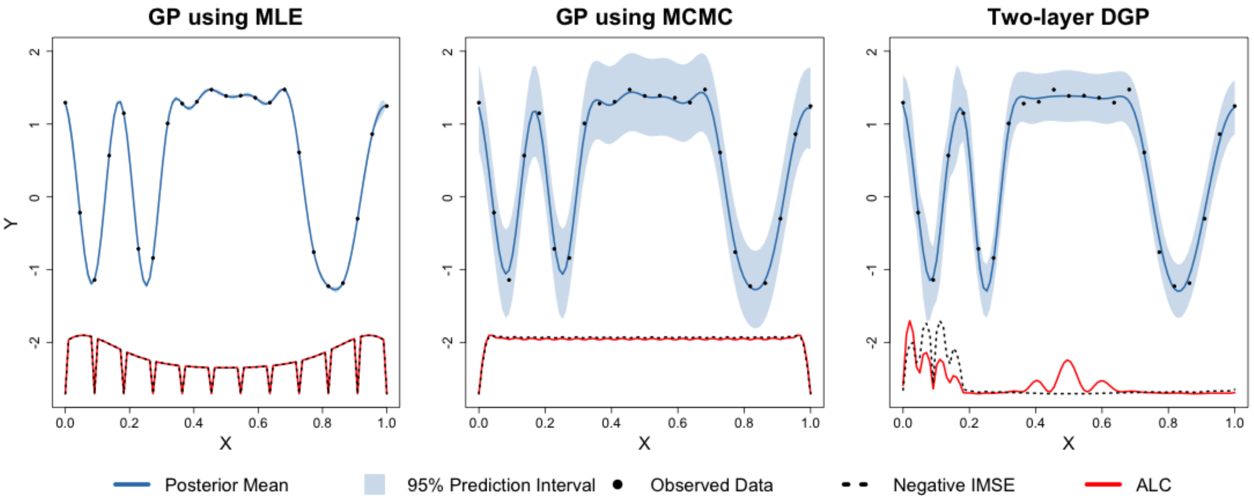


Figure 1: An example figure appearing in Section 2 of Sauer et al. (2022). The first two panels were introduced here; the last one is saved for later.

true. They should be able to appreciate many aspects of the figure by simply looking at it, without narrative. Think about what might look intriguing from ten feet away.

Be careful about how you use vertical space in figures, pay close attention to the ink-to-void ratio, and don't "squoosh". Ensure resolution is high, and fonts readable. In LaTeX, use `\begin{figure}[ht!]`, and place that markup *before* the figure's first reference to ensure that it floats near the top of the page (or following page) containing that text. Preserve original aspect ratios for plots generated by software; these should be roughly square if they have a main title above them, or no fatter than 4:3 (wide:tall) if they don't.[5] Vertical space is precious. Make sure your figure is worth its space in words. Try to show, and draw attention to, more than one interesting takeaway message in the figure. Sometimes that means by supporting narrative from disparate parts of the paper. Since we're talking about managing space, the easiest way to shorten a paper is to look for paragraphs whose final line is short, being only one or two words long. It's usually possible to re-word things earlier in the paragraph more economically in order to remove that dangling final line. Now that you're aware of it, you'll notice that I don't have any of those in this paper.

2.2 Second topic

If you have one sub-section in your section, make sure you have at least two. Otherwise, you don't need sub-sections at all. Consider re-arranging things so that they're either more hierarchical, or flat like Section 1. Hierarchy is important for longer sections, flatness for shorter ones. But if it's too short, maybe it should be subsumed into another, longer sub-section. This sub-section here is possibly too short, but I've included it as an illustration.

[5] Of course, there are always exceptions, but this is a good rule to start out with. If you find that you consistently require an exception, then you don't know what it means to be exceptional.

Some papers don't have a background/related work section at all, preferring to jump straight into novel development, mixing in previous work when necessary, and/or relying on a lengthier Introduction. I generally think this is a mistake. Take the opportunity here to introduce your notation, be clear about what has come before that you need for your development, identify gaps, and motivate your novel contributions coming in Section 3. Sauer et al. (2022) used Section 2.1 to introduce GPs, and Section 2.2 for active learning (AL) heuristics – those black and red lines at the bottom of Figure 1. Do you see what I did there? I delayed talking about a part of an earlier figure until a later sub-section, here.

I like to end Section 2 in a provocative manner, making it obvious it's time to transition to novel methodology. This isn't a steadfast rule, but it can be an effective device. You need a transition passage anyways, and it's usually pretty easy to do if you've been hinting at flaws along the way in your review.

3 Main novel contribution

This is the meat of your paper. Just like in the review of Section 2, break things into pieces and arrange them hierarchically. Don't forget you are leading your reader on a journey. Find a balance between entertaining and being a useful reference to those just browsing.

3.1 Methodological development

If the paper has any math in it, most of it will be here. Be matter-of-fact, but provide intuition. Remember that many *Technometrics* readers are practitioners, not theoreticians. At times, highly technical arguments are required, but the best *Tech* papers don't make a fuss about theory. A flow that has multiple propositions, lemmas, theorems, proofs (likely relegated to the supplement), and remarks seldom connects well with our readers and will bias our AEs/referees against you. The problem with this setup, for practitioners anyways, is that the presentation is backwards; it puts the result (theorem) before the intuition/evidence (proof). Our readers would prefer a less formal setup. They'd like to have the intuition first, and then the result in the form of a cute formula or mathematical relationship between quantities, that they can consult/extract later if desired.

It's important to clarify that this doesn't mean *Technometrics* papers lack mathematical sophistication. Although presentation can be less formal, emphasizing grace over acrobatics, it's still important that your arguments be specific, with precise and articulate notation and a comprehensive description of relationships between quantities. The "tech" in *Technometrics* is about technology, but I think it is equally about being technical in terms of arguments. Many feel that this is what sets *Tech* apart from its competitors with a similar remit, like the *Journal of Quality Technology* (JQT), for example. This can only really be appreciated by having a look through our archives, which I highly encourage.

Rather than duplicate an overly-mathy example here, like I did in Section 2, I'll go over some of the highlights while at the same time using the space here to provide more advice. In addition to prose and mathematics, it's often helpful to have a diagram. Try to aim for at least one diagram in your paper, which differs from a plot by being more like a drawing. Use it to highlight a flow or a relationship between quantities. Use software or other precision-tools. Don't draw by hand, and don't force it if a diagram doesn't seem natural for what you're doing. Use reasoning similar to that provided around Figure 1. Offer your readers visual stimulus, especially to complement a highly technical development. As with any figure, make sure your diagram is readable and appropriately scaled. Whereas plots are typically square (or 4:3), diagrams can sometimes be shaped more advantageously, especially in the horizontal direction. (Orienting a diagram taller than wide is a waste of vertical space.)

For example, Figure 2 shows a diagram of a three-layer DGP developed in Section 3 of Sauer et al. (2022). Again, make sure to verbalize what you want the reader to see in the diagram. It should both stand alone, and be appreciated via the narrative at a glance. In this case, latent quantities $\mathbf{Z}$ and $\mathbf{W}$, each with GP setups like in Eq. (1), are inserted between outputs $\mathbf{Y}_n$ and inputs $\mathbf{X}_n$, warping the inputs so that a cascade of ordinary, stationary GP models can accommodate highly flexible, non-stationary input–output dynamics. The equations and details for parameters $(\theta_{(\cdot)}, g, \tau^2)$ would be a distraction to go over here and can be found in Sauer et al., who offer an excellent example of a technical narrative that's not overly formal, and instead focused on intuition (though I'm biased).

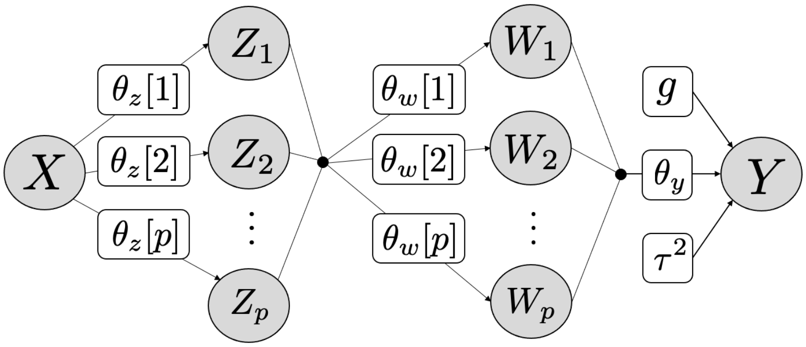


Figure 2: Model structure for a three-layer DGP with p latent nodes in $\mathbf{Z}$ and $\mathbf{W}$.

It might also be appropriate to provide a pseudo code in a formal Algorithm environment. But again, don't force it. If you choose to include one, be careful to describe the algorithm mathematically, using and symbols and operators, with reference to equations as much as possible. Use comments ("//") to describe and detail, but also don't forget to narrate your algorithm in the main prose. Use pseudo-code structures, like loops, conditionals and indentation, but remember that this is for understanding, not for reproduction. You'll be commenting more on reproduction and implementation later in Section 5.1.

Algorithm 1 details the Gibbs sampling scheme for DGPs. You guessed it – from Sauer et al. (2022). Many of the details would be too much of a distraction to explain here, but there are still important high-level observations that can be made, exemplifying best practice. Notice that everything is well-defined in the flow of the algorithm. There isn't a reference to a quantity that wasn't either calculated in an earlier step, earlier iteration in the **for** loop, or provided as initialization. The comments provide equation numbers, e.g., (11), from that paper, so that you can specifically reference what's involved in that calculation, and more

Algorithm 1: Gibbs sampling procedure for three-layer DGP

```
initialize $g^{(1)}$, $\theta_y^{(1)}$, $\theta_w^{(1)}$, $\theta_z^{(1)}$, $\mathbf{W}^{(1)}$,and $\mathbf{Z}^{(1)}$
for $t = 2, \ldots, T$ do
    $g^{(t)} \sim \pi(g \mid \mathbf{Y}_n, \mathbf{W}^{(t-1)}, \theta_y^{(t-1)}, g^{(t-1)})$    // MH (11) via $\mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \theta_y \;\; g)$
    $\theta_y^{(t)} \sim \pi(\theta_y \mid \mathbf{Y}_n, \mathbf{W}^{(t-1)}, \theta_y^{(t-1)}, g^{(t)})$    // MH (11) via $\mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \theta_y \;\; g)$
    for $i = 1, \ldots, p$ do
        $\theta_w[i]^{(t)} \sim \pi(\theta_w[i] \mid \mathbf{W}_i^{(t-1)}, \mathbf{Z}^{(t-1)}, \theta_w[i]^{(t-1)})$    // MH (11) via $\mathcal{L}(\mathbf{W}_i \mid \mathbf{Z}, \theta_w[i])$
    for $i = 1, \ldots, p$ do
        $\theta_z[i]^{(t)} \sim \pi(\theta_z[i] \mid \mathbf{Z}_i^{(t-1)}, \mathbf{X}_n, \theta_z[i]^{(t-1)})$    // MH (11) via $\mathcal{L}(\mathbf{Z}_i \mid \mathbf{X}_n, \theta_z[i])$
    for $i = 1, \ldots, p$ do
        $\mathbf{W}_i^{(t)} \sim \pi(\mathbf{W} \mid \mathbf{Y}_n, \mathbf{W}_{\geq i}^{(t-1)}, \mathbf{W}_{<i}^{(t)}, \mathbf{Z}^{(t-1)}, g^{(t)}, \theta_y^{(t)}, \theta_w^{(t)})$    // ESS via (13)
    for $i = 1, \ldots, p$ do
        $\mathbf{Z}_i^{(t)} \sim \pi(\mathbf{Z} \mid \mathbf{X}_n, \mathbf{W}^{(t)}, \mathbf{Z}_{\geq i}^{(t-1)}, \mathbf{Z}_{<i}^{(t)}, \theta_w^{(t)}, \theta_z^{(t)})$    // ESS via (14)
```

intimately connect to the main text. Although not exemplified in Algorithm 1, if you plan to have multiple algorithms, perhaps developing macros or subroutines to be used in a bigger code later, be sure to indicate what quantities are returned by the subroutine. Point your reader to the specific Algorithm where that subroutine can be found.

3.2 Further development

Remember, since you had one sub-section [Section 3.1], you must now have two [Section 3.2]. Maybe you have a special case, or a generalization to demonstrate or reveal. Perhaps, after first introducing the method you might, here, provide theory or an analysis of computational complexity, or approximations that help circumvent challenges to applying the method at scale. Avoid a smattering of facts. Whatever you provide should be motivated by the desires or concerns of a practitioner, or the needs of a motivating application. Maybe you wish to clarify how your method serves a wider class of problems, or offer an analysis that gives additional insight, or connects your method to others' on a technical level.

So that this looks like a *Technometrics* paper from ten feet away, I shall paste more from Sauer et al. (2022) here. Although the calculation is rather textbook, e.g., see Gramacy (2020, Section 5.5, exercise 1), the concentrated log likelihood for a two-layer DGP with latent node $\mathbf{W}$ is provided below for concreteness:

$$\log \mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \theta_y, g) \propto -\frac{n}{2} \log \left(n\hat{\tau}^2\right) - \frac{1}{2} \log |K_{\theta_y}(\mathbf{W}) + g\mathbb{I}_n|, \tag{2}$$

with $\hat{\tau}^2 = \frac{1}{n}\left(\mathbf{Y}^\top (K_{\theta_y}(\mathbf{W}) + g\mathbb{I}_n)^{-1}\mathbf{Y}\right)$. Throughout, the symbol "$\propto$" for log likelihoods indicates that an additive constant has been dropped. For DGPs with two and three layers, we additionally require the following

$$
\begin{aligned}
\log \mathcal{L}(\mathbf{W} \mid \mathbf{Z}, \theta_w) &\propto \sum_{i=1}^{p} \log \mathcal{L}\left(\mathbf{W}_i \mid \mathbf{Z}, \theta_w[i]\right) \\
&\propto \sum_{i=1}^{p} \left(-\frac{1}{2}\log |K_{\theta_w[i]}(\mathbf{Z})| - \frac{1}{2}\mathbf{W}_i^\top \left(K_{\theta_w[i]}(\mathbf{Z})\right)^{-1} \mathbf{W}_i\right).
\end{aligned} \tag{3}
$$

In two-layers, take $\mathbf{Z} \equiv \mathbf{X}_n$ and collect

$$
\log \mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \mathbf{X}_n, \theta, g) = \log \mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \theta_y, g) + \log \mathcal{L}(\mathbf{W} \mid \mathbf{X}_n, \theta_w),
$$

combining Eqs. (2–3). Of course, in the context above of unknown (latent) $\mathbf{W}$, the quantity $\log \mathcal{L}(\mathbf{W} \mid \mathbf{X}_n, \theta_w)$ is actually a log prior, but its form is nevertheless given by Eq. (3), so it's convenient to notate using marginal log likelihoods.

In the three layer case, with trivial extension beyond,

$$
\log \mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \mathbf{Z}, \mathbf{X}_n, \theta, g) = \log \mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \theta_y, g) + \log \mathcal{L}(\mathbf{W} \mid \mathbf{Z}, \theta_w) + \log \mathcal{L}(\mathbf{Z} \mid \mathbf{X}_n, \theta_z),
$$

where the third term follows Eq. (3) for $\mathbf{Z}$ and $\mathbf{X}_n$ instead of $\mathbf{W}$ and $\mathbf{Z}$. The posterior distribution is completed with the hyperparameter priors:

$$
\pi(\mathbf{W}, \mathbf{Z}, \theta, g \mid \mathbf{D}_n) \propto \mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \mathbf{Z}, \mathbf{X}_n, \theta, g) \times \pi(\theta, g).
$$

Ok, that's probably enough of that. Obviously, there isn't enough above to really understand what's going on. But you have to admit that the paper looks more legit now as a technical statistics publication, even though there are no propositions, lemmas, theorems or remarks. As earlier, take note of the cleanly formatted and aligned equations, references to earlier equations, and explanation of notation and other quantities being used. Notice the balanced tall parentheses in Eq. (3), via `\left(` and `\right)` markups in LaTeX.

Illustration

I like to cap off my methodological sections with something visual. This is partly to break up the flow of the development, but also to force the reader to re-engage with the material and check/reinforce their understanding. If you provided a warm-up visual earlier, say in your review in Section 2, then this would be a good time to refer back to that figure to fill in detail or any gaps. For example, the final panel of Figure 1 was left un-narrated. Sauer et al. (2022) explain that the two-layer GP in that panel is able to capture, separately, the wiggly outer regions and the inner, flat one, by warping inputs $\mathbf{X}$ with latent layer $\mathbf{W}$. Consequently, AL heuristics provided at the bottom of each panel demonstrate that future sampling would be more focused in the interesting region, whereas in the first two panels they were (unproductively) rather uniform by comparison. However, this last detail is not narrated until their Section 4. So hold that thought for a moment.

Remembering to complete the description of an earlier figure is important, but doesn't do much to add novel visual stimulus here. Sauer et al. provide a new figure to augment this example, showing how latent $\mathbf{W}$s warp $\mathbf{X}$s. At the same time they demonstrate, making good use of horizontal figure space, that their proposed method, based on elliptical slice sampling (ESS; Murray et al., 2010), offers higher efficiency compared to ordinary Metropolis–Hastings (MH) sampling. I've decided not to duplicate that figure here to save space.

4 Novel contribution two

Most good papers offer two distinct ideas. Often the second one is "icing on the cake", or "but wait there's more". This works for research papers as in marketing on television. But it could equally be the other way around; the second one is the big idea, but you need the first one first for the second one to make sense. This is how it is in Sauer et al. (2022), my running example. Follow the outline already provided for Section 3, including hierarchy, and ending with visual stimulus/illustration. Rather than duplicate that here, I'm going to use the space for a digression. A secondary aim of this tutorial – a second novel contribution if you will – is to comment on the tools that make for an effective, and professional presentation.

4.1 Typesetting

Although *Technometrics* will accept submissions typeset in Word, I would strongly discourage it. Word documents don't look professional for papers with equations, mathematical illustrations and figures. I don't know why this is. Presumably it's possible to make things look good, but I've never seen it in academic circles. Word's menus and buttons are friendly and, for many, familiar. But this gloss masks an insidiously cumbersome interface for producing a technical document. Mathematical typsetting in Word is particularly problematic. This is especially shocking when you consider that this "problem" was solved in the 1970s, more than fifty years ago with TEX, and now LATEX, well before Microsoft, or any other visual document editor for that matter. If you're new to LATEX or rusty, let me encourage you to invest in climbing the learning curve. It's not hard, especially with modern collaborative tools like OverLeaf. Using anything else, while technically allowed, will make your contribution look amateurish, and bias referees against you.

I really like markdown-based tools, e.g., `Rmarkdown` (Allaire et al., 2018) for R (R Core Team, 2019) for code demonstration. I wrote my entire book (Gramacy, 2020) in `bookdown` (Xie, 2016, 2018), a variant/add-on for `Rmarkdown`. These tools can output LATEX and other formats, and are great for making reproducible, living documents. By the way, notice how I'm using `typewriter` font for packages/libraries, and Sans Serif for languages, and that I've cited all of them. This is standard and expected. For example, the Journal of Statistical Software has a style guide[6] going over just this. Their template provides macros `\pkg` and `\proglang` which are wrappers around `\texttt` and `\textsf`, respectively. Ok, that was a big digression – maybe don't do that in your papers. `Rmarkdown`, notebooks (e.g., Jupyter)

[6] https://www.jstatsoft.org/style/

and similar mixed-code and text environments that come with RStudio, are democratizing the dissemination, and organization of scientific communication and collaboration, especially on the web. They're fantastic for what they are, but not for journal papers because they limit control over formatting, particularly the use of precious space. Perhaps more practically, re-running code chunks to revise figures, tables, and other output is cumbersome, even when taking advantage of caching features. Unless showing code plays a pivotal role in your presentation, which would be rare at *Technometrics*, stick to raw LaTeX.

As a highly personal preference, avoid macros in LaTeX. You might think that by mapping `\E` to `\mathbb{E}` you're saving yourself time and from repetitive stress injury by eliminating a few keystrokes if you need to take a lot of expectations: $\mathbb{E}\{Y\}$, say. But you're not really. You're going to annoy your co-authors by forcing them to learn yet another idiosyncratic short-hand. You're going to annoy your future self when you try to paste this TeX into another collaborative project, like a grant application (where plagiarizing your own work is allowed, even encouraged), and discover your macros clash with your colleague's macros. You'll try to merge them and end up in a hot macro mess. I've learned this lesson the hard way through hours of undoing my own macros, or reconciling them with others'. Save yourself from future hassle and type it all out. You'll get faster with practice.

Here are a couple of other mistakes that I see folks make over and over again with LaTeX. Both involve control over space which – I know I'm a broken record – is at a premium. Don't put blank-line carriage returns between your displayed equations and the text immediately above and below. This will put extra white space above them, and automatically indent to start a new paragraph below them. That blank space is a waste, and you probably don't want to start a new paragraph, especially if you're continuing a sentence after the equation. If that's the case, don't forget to put a comma after your equation. If you are ending a sentence, put a full-stop (period). It's quite rare that you would want to finish a paragraph with displayed equation, which is what you're doing with those extra carriage returns.

Sometimes you may want to use full-stop mark ".", but you're not ending a sentence. In that case, put a tilde character ("$\sim$") after it rather than an ordinary space. If you don't, LaTeX will put two spaces rather than one because it thinks you intend to start a new sentence. This mistake is most common with abbreviations. For example, suppose we wish to refer Eq. (2.1) or Prof. Gramacy. Notice how there's extra space in both compared to Eq. (2.1) and Prof. Gramacy. A tilde after the period will also prevent a break across lines in the middle of your abbreviation if formatting your sentence requires a wrap.

4.2 Miscellaneous dos and don'ts

Try not to promise a "general", "unifying" or "systematic" treatment of anything, like in the title, abstract or into. I could have titled this paper "General guide to writing a research paper", but I didn't. There's power in specificity, which allows you to hone things. It's much better to be humble in the salesmanship of your offering, and then over-deliver. The opposite can be devastating from a refereeing perspective. Your contribution could be powerful and important, but if you over-sell it, especially when setting expectations early on, referees will rake you over the coals for it in their reports. They won't be able to accurately measure the absolute value of the merits of your contribution because they'll be distracted by a feeling that your paper doesn't measure up to the overly-hyped, high bar you set early on. If you do a good job with your writing, in particular using a breadth of empirical evidence, then the generality of your contribution will be implicit to the reader without forcing it. Don't promise to be the final word on anything. You'll eat those words later, even if they make it into print, when someone bests you. That's inevitable, so just embrace it at the outset.

I try to avoid bulleted or enumerated lists throughout. I could have used a bulleted list for this sub-section, maybe even for the entire tutorial, but I chose not to and I think it looks ok. In-line lists are fine if (a) the items are short; (b) you don't need to refer back to them often; and/or (c) you can think of a third thing for the purposes of this illustration. You don't need a list at all if there are fewer than three things! The main reason for this is that – you guessed it! – bulleted lists take up too much space. They're the first things to go when the Editor asks for you to find economies, so why not get rid of them proactively? I have one caveat to this recommendation in Section 5. Similarly, don't provide a list/table of notation. *Technometrics* won't publish it. I'm a firm believer in YPYW,[7] but with this indulgence you're screaming to the editorial team that you don't know what a *Tech* paper looks like. If you feel compelled to have a notation key for "convenience", put it in the online supplement. Make sure you introduce all quantities when you first use them. If you do a good job of that, your reader will be able to find them, either at a glance or with Ctrl-F.

It's almost never a good idea to have a paragraph comprised of a single sentence, or a section comprised of a single paragraph. An exception to the former might be as a heading to start a section, briefly laying out the sub-sections to follow. It's especially problematic to have multiple one-sentence paragraphs in a row. Find a way to weave small ideas (sentences) into a cohesive theme that can be delivered together (in a paragraph). While we're on the topic of "nevers", never speculate outside of the final discussion section [Section 7], i.e., at the end of the paper. If you think there's potential for your implementation to be parallelized, or implemented on a GPU, save it for later unless you've actually done it and can provide empirical demonstration. Don't speculate that it would be trivial to adapt your 1d presentation for higher dimension. You're over-selling if you're describing your 2d method as "multi-dimensional". Use bivariate, or spatial. If you think your method for regression can be adapted to classification without much hassle (a logistic link), save it for the discussion unless you're prepared to demonstrate it. Things are never that simple when rubber meets the road. You don't get to own an extension of an idea simply by mentioning it offhandedly. You're not a wolf. Overcome the temptation to mark your territory.

5 Implementation and benchmarking

There's substantial variation in how empirical results are presented in a stats paper, but at *Technometrics* there's rather less variability. Most of the papers that get accepted at *Tech* will have illustration and benchmarking on both synthetic and real-data examples, and both

[7] Your paper, your way (YPYW). See: `https://www.elsevier.com/authors/tools-and-resources/your-paper-your-way`

will draw quantitative comparisons to a representative set of the methods believed to be state-of-the-art for the class of problems being studied. I like to present the idealized, synthetic examples first [here in Section 5], and then turn to real/motivating examples next [Section 6]. Others prefer the other way around. But first, be clear about what you're evaluating, and acknowledge that you're stress-testing the engineering of the implementation behind the methods at least as much as the methods themselves. You've been promising those details since Sections 3–4, so that's a good place to start.

Before we jump in, a disclaimer may be warranted. The passages here and in Section 6 attempt weave advice and examples (from other papers) rather more seamlessly than earlier sections. It's not my intention that you follow the scientific content of the examples, although in a real paper that's of course important. My goal here is to introduce and demonstrate tools – statistical, narrative, visual – that make for a successful presentation of empirical evidence, all while looking like a *Technometrics* paper from ten feet away. An important takeway is how much – in terms of sheer volume – empirical work is expected at *Tech*.

5.1 Implementation details

My goal in this sub-section is usually four-fold: to (i) describe important considerations, and how they were handled, when bridging the gap between a mathematical description (say Algorithm 1), and its numerical analog in code; (ii) explain how it's coded, like in what language and with libraries/subroutines; (iii) introduce competing methods; and (iv) explain how all of the methods will be evaluated (i.e., what metrics will be used) so that a concrete and unambiguous numerical, and statistical comparison can be drawn between them. Sometimes it makes sense to do each of those separately, like in their own (un-numbered) sub-sub-section. I find (i)–(ii) go hand-in hand, but note that your reader is most interested in (i). Points (iii)–(iv) are also often intertwined. I'll present them here grouped in this way, but in an otherwise flattened narrative that avoids additional hierarchy.

The cornerstone of these passages should be a pointer to where the reader can go to get the code to reproduce your experiments, for all methods under consideration. Reproduction is a requirement at *Technometrics*, as well as most other top outlets. For example, code to reproduce all results from the Sauer et al. (2022) paper, and a follow-on one (Sauer et al., 2023) can be found in the Git repository linked below.

`https://bitbucket.org/gramacylab/deepgp-ex/`

You don't need to have a public repository for consideration at *Tech*. But I can't see any downside to having one except with double-blinding, about which I shall say more in Section 7. Some authors worry about the time it will take to make code "presentable" for an online repo. But you need your code to be that presentable anyway, even if you just submit a zipped archive rather than linking to a repo. If a *Tech* AE can't get it to run, and reproduce results from the paper without hassle, acceptance of your paper could be delayed. It may entirely be derailed, though that is unlikely.

Sauer et al. (2022) explain that all implementation of (novel) methods described in that paper are provided for `R` in the `deepgp` package (Sauer, 2020) on CRAN. Notice that I have cited the package. It's not sufficient to cite the main paper only. The authors for packages may not be the same as those for the original paper. Sauer et al. go on to discuss defaults in the packages, such as settings for proposal and prior distributions, numbers and width of intermediate layers, etc., ending with suggestions that are not dictated by defaults. They then give an example of how to fit DGP models using functions from the package.

You do not need to have a package (for R, Python, MATLAB, etc.), but again it doesn't hurt. Making it practical for others to use your methods pays dividends in impact. This is important for you and for the journal. Besides evaluating the merit and correctness of the methods proposed in the paper, referees are attempting to gauge impact. They'll try to assess what influence your work will have on the work of others, and ultimately on citations which affect the journal's impact factor.[8] A package on a prominent repository can help make the case for impact, and thus will increase your chances of being accepted for publication. I'd guess about half of my Google Scholar citations came because another author has used my software in a benchmarking capacity, like I'm about to describe momentarily. You can't be included in such comparisons without accessible software. The higher the barriers are, the lower the impact and the fewer the cites.

Next introduce your comparators, which probably means links to both software and the original methodological papers. This is the one place in the paper where it can be advantageous to utilize a bulleted list, so that it's easy for readers to reference what comes from where and why. Sauer et al. (2023)'s follow-on paper has one of these, although the original does not. This list is paraphrased as a demonstration below.

- DGP VEC: Sauer et al. (2023)'s Vecchia-DGP, via `deepgp` using defaults, Matèrn $\nu = 5/2$ kernel, independent predictions, and "warped" conditioning sets.

- DGP FULL: full, un-approximated analog of DGP VEC, with everything otherwise identical (also via `deepgp`). This comparator was not feasible for all data sizes.

- DGP DSVI: from Salimbeni and Deisenroth (2017), implemented in `gpflux` (Dutordoir et al., 2021), with Matèrn $\nu = 5/2$ kernel. We follow package examples in using 100 k-means located inducing points. For numerical stability we required `eps = 1e-4`.

- DGP HMC: from Havasi et al. (2018) using 100 inducing points. This code only supports a squared exponential kernel and estimates (does not fix) the noise parameter.

- GP: stationary un-approximated GP following Eq. (2.1) with Matérn $\nu = 5/2$ kernel and anisotropic lengthscales estimated through MLE. Not feasible for all data sizes.

- GP SVEC: scaled Vecchia-GP of Katzfuss et al. (2020a), via `GPvecchia` (Katzfuss et al., 2020b) and `GpGp` (Guinness et al., 2021). This is a fast "shallow" GP where kernel hyperparameters are estimated via lengthscale-adjusted conditioning sets.

Finally, wrap up the implementation section by describing how the experiments, beginning in Section 5.2 but also continuing with real-data in Section 6, will be conducted. There

[8] `https://en.wikipedia.org/wiki/Impact_factor`

may be differences, or multiple kinds of experiments, but it can help for things to be uniform and to describe as much as possible up front. You can always explain variations and exceptions later. For example, Sauer et al. (2022) explain that they assess predictive accuracy, over the course of AL iterations, via RMSE and proper score (Gneiting and Raftery, 2007, Eq. 25), and over multiple Monte Carlo (MC) repetitions. You could provide the formulas for your metrics. RMSE is rather straightforward (smaller is better). Proper score is less common, but notice that I provided a reference to an equation (larger scores are better). Sauer et al. (2023) explain that they prefer the continuously ranked probability score (CRPS; smaller is better), also from Gneiting and Raftery (2007). Given a Gaussian predictor with mean $\mu^\star$ and point-wise standard deviations $\sigma^\star$, CRPS is defined as

$$\text{CRPS}\left(y^{\text{true}} \mid \mu^\star, \sigma^\star\right) = \sigma^\star \left[\frac{1}{\sqrt{\pi}} - 2\phi(z) - z\left(2\Phi(z) - 1\right)\right] \quad \text{for} \quad z = \frac{y^{\text{true}} - \mu^\star}{\sigma^\star} \tag{4}$$

where ϕ and Φ are the standard Gaussian pdf and cdf, respectively.

These specific metrics might not be relevant in your particular analysis, because your methodological development might not involve prediction. But make sure you are measuring something, drawing an empirical comparison to something else, and that you're using least one metric that incorporates uncertainty which is a hallmark – an essential aspect – of statistical inquiry. Find a visually stimulating way to convey results, avoiding tables if at all possible. Ensure that computing time, storage/memory is included in your analysis.

5.2 Synthetic experiments

Empirical evaluation is important to *Technometrics*, and a key component of that is validation in a controlled setting, where you can focus on aspects of a data-generating mechanism suited to your proposed methodology. If you can't find, or engineer, examples where your proposed method compares favorably to others, then you're not yet ready to submit to *Tech*. Remember, the selections here are your choice. In the ideal scenario, you should be able to pull familiar data-generating mechanisms from a library of common benchmarks, such as the *Virtual library of simulation experiments* (VLSE; Surjanovic and Bingham, 2013). This is just an example, and may not apply to your class of problems. Choose something that connects to your literature, so that your contribution can be vetted on a common footing with other, similar and contemporary methods. If you must engineer a new example, that's fine. Its appropriation by others can be a great way to generate citations and expand impact down the line, but you risk a weaker connection with your reader.

Aim for at least three examples total, including the real-data one described in Section 6. Make sure to explain everything about each experiment that isn't already introduced earlier, say in Section 5.1. Relevant quantities include problem size, like the number of records n and input dimension d. If there are multiple variations of a classic example, be clear about which one you're considering. Are you using coded inputs or scaled outputs? If so, say so. Be thorough but not pedantic. Minutia can be left to your code supplement. Descriptions of an experimental apparatus and the results they yield are hard to write so that they're interesting to read. But it can be done. At least one of the examples here should be simple/small enough to be reproduced by the editorial team (and readers) without cumbersome computation. That means standard hardware and less than an hour of computing time using code provided in the supplement. It may also mean pairing things down for one, but not all examples. If you provided illustrations early on, like in Figures 1 and 2 in Section 2, your first example could be on the same/similar problem, and this may be the first place where you're discussing the details of that illustration. Sauer et al. (2022)'s first (1d) example in their Section 5 introduces the data-generating mechanism behind those figures, and then provides an analysis similar to that described below.

2d. Their second example is from Gramacy and Lee (2009), who introduced the treed Gaussian process (TGP), an important comparator.[9] The data-generating mechanism can be found on the VLSE:[10], but Sauer et al. nonetheless quote it as follows.

$$f(x_1, x_2) = 10x_1 \exp\left(-x_1^2 - x_2^2\right) \quad \text{for} \quad x_1, x_2 \in [-2, 4]. \tag{5}$$

For the purposes of training and testing, outputs $Y(x) = f(x_1, x_2) + \varepsilon$ are observed with $\varepsilon \overset{\text{iid}}{\sim} \mathcal{N}(0, 0.1^2)$ noise. The left panel of Figure 3 provides a visual of the mean/noiseless surface, hilighting two distinct regimes using color along the x-axes. Sauer et al. describe an AL experiment that's initialized with random LHSs (Morris and Mitchell, 1995) of size $n_0 = 10$, followed by ALC acquisitions (200 candidates/references) up to $n = 80$. Figure 4 shows RMSE and score calculated after each iteration for one- (ordinary GP), two-, and three-layer (DGP) models as well as TGP. I'll spare you the detailed analysis here except to note that both DGPs excel at placing design points in the region of interest [Figure 3, right panel] and also yield lower RMSEs/higher scores.

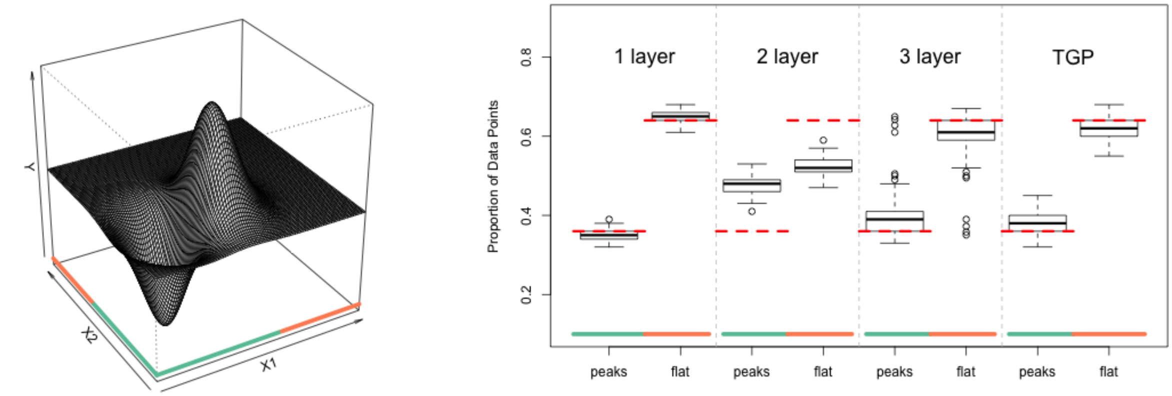


Figure 3: *Left:* $f(x_1, x_2)$ from Eq. (5). Two input regions are marked by the green and orange lines along the x-axes. *Right:* Proportion of AL acquisitions falling in each region over 100 repetitions. Red dashed lines mark the actual proportions occupied by each region.

[9] Note that this comparator is not in the bulleted list in Section 5.1, which is from a different paper. We shall return to those comparators momentarily.

[10] `https://www.sfu.ca/~ssurjano/grlee08.html`

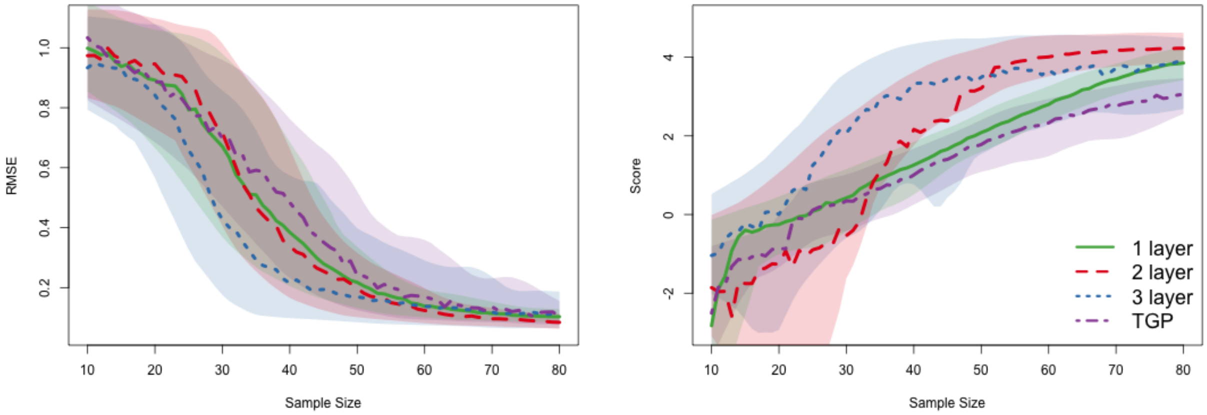


Figure 4: RMSE (left) and score (right) for $f(x_1, x_2)$ from Eq. (5). Solid lines represent the average over 100 repetitions. Shaded regions represent 90% quantiles.

This latter trend is clearer visually for score (i.e., incorporating uncertainty) than for RMSE (accuracy only). In such situations, it's not uncommon to provide additional (non-visual), statistical evidence of the superiority of one method over another. A paired Wilcox or t-test of the RMSE values for a slice along the x-axis in the left panel of Figure 4 could help here. See the table provided in Figure 4 of Cole et al. (2022) or Table 1 in Gramacy and Apley (2015). Paired tests only make sense if all other aspects of the experiment are held constant – only method(s) are varied.

G-function. Sauer et al. (2023) consider the G-function (Marrel et al., 2009), which can be found on the VLSE.[11] Unlike the previous example, which is from another paper, they don't quote the function here, in-line with their discussion. However they do provide it in their online supplement. The G-function is defined in arbitrary dimension. Earlier in their paper – not here, but in their Section 2 review – they utilized a $d = 2$ version supporting the visuals of an illustration. For their empirical work later they opt for a harder, $d = 4$ case. Their experiment involved fitting GP and DGP models to noise-free realizations on LHS samples of training sizes $n \in \{3000, 5000, 7000\}$ and testing on separately generated LHSs of size $n_p = 5000$. Results for twenty MC repetitions are displayed in Figure 5. Their supplement provides an additional, noisy variation.

The comparators along the x-axis were introduced earlier in the bulleted-list of Section 5.1. The left panel shows RMSE; the right has CRPS (4). Again, I shall spare you an analysis here. But don't forget to narrate what you want the reader to see in the plot. E.g., that the DGP VEC model has the lowest RMSEs and CRPSs. Notice results were only provided for the full GP for the smallest sample sizes, n, owing to the cubic cost of that method.

[11] `https://www.sfu.ca/~ssurjano/gfunc.html`

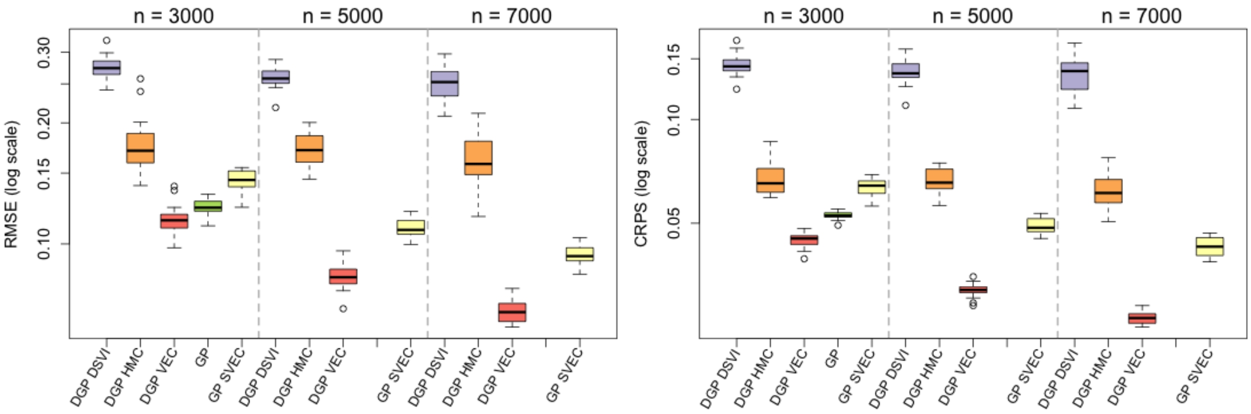


Figure 5: RMSE (left) and CRPS (right) on log scales for the 4d G-function.

6 Empirical motivating examples

It's exceedingly rare that papers are accepted with only examples on a case study, i.e., no synthetic/toy examples [Section 5.2]. But sometimes this is allowed, especially if the case study is rather involved, and requires much space to analyze. In such situations – and it would have to be a pretty intricate case study to warrant foregoing proof that your method applies more widely – you might only have one empirical section [Section 5], although you'll probably still need to go over implementation details and metrics somewhere. Similarly, it's possible you don't have a genuine motivating, real-data example. Usually this is for lack of imagination, and therefore will bias referees against you. Many textbooks provide real-data examples ready for appropriation. Chapter 2 of my *Surrogates* book (Gramacy, 2020) outlines four good examples involving computer simulation experiments, with full data/code for download. Sauer et al. (2022) vet their DGP on two of these examples, and highlights will be provided momentarily.[12] Repositories like those from UCI (Asuncion and Newman, 2007) are a nice place to get challenging real-data benchmark problems.

6.1 Langley glide-back-booster

This example actually appears in the supplementary material of Sauer et al. (2022). Originally it was in the main body of the paper, but the editorial team asked for expanded narrative/examples in other parts of the paper without adding length. So we opted to relocate this one, keeping the satellite drag example coming next in Section 6.2. Both serve as good illustrative examples, and allow me to highlight relevant aspects of a real data analysis here. The first step is to succinctly describe the problem, and include any references to the domain-literature from which the problem/data originate. You may have already introduced this problem in your Section 1. Don't duplicate that here. Whereas that introduction should be high-level, these passages should provide more detail, especially about how you intend to

[12] These examples are presented in Sauer et al.'s Section 5, as Section 5.3, alongside the other synthetic examples rather than separately in a Section 6.

bring your novel methods to bear now that your reader has a full appreciation of those ideas from earlier in the paper. Below I shall paraphrase by copying aspects from Sauer et al. (2022), interleaved with commentary.

The Langley Glide-Back Booster (LGBB) computer model was designed by NASA to assess the movement of a rocket booster gliding back to Earth for re-use after launching a payload into orbit. A thorough review is provided in Pamadi et al. (2004). The LGBB simulator has three inputs – speed upon entry (mach), angle of attack (alpha), and side-slip angle (beta) – and produces six deterministic response values (lift, drag, pitch, side, yaw, and roll). This model yields non-stationary response surface because the sound barrier at mach 1 imparts regime changes on aeronautic dynamics. The TGP model was specifically designed for this simulator, so it makes for a formidable comparator.

Figure 6 provides a visual of the lift response for a fixed beta value. Notice how the sound barrier causes a sharp ridge at mach 1. The region with mach values less than 2 is more “interesting”. After initializing with a random uniform (sub-) design size $n_0 = 50$, Sauer et al. entertained AL for acquisitions up to $n = 300$. RMSE on a 1000-point out-of-sample testing set was evaluated after every 10th acquisition. Creating a train–test partition is an important aspect of any real-data analysis, allowing for out-of-sample comparisons, say in terms of accuracy. The partition should be random, and repeated in order to study the distribution of behavior/performance metrics under study via MC.

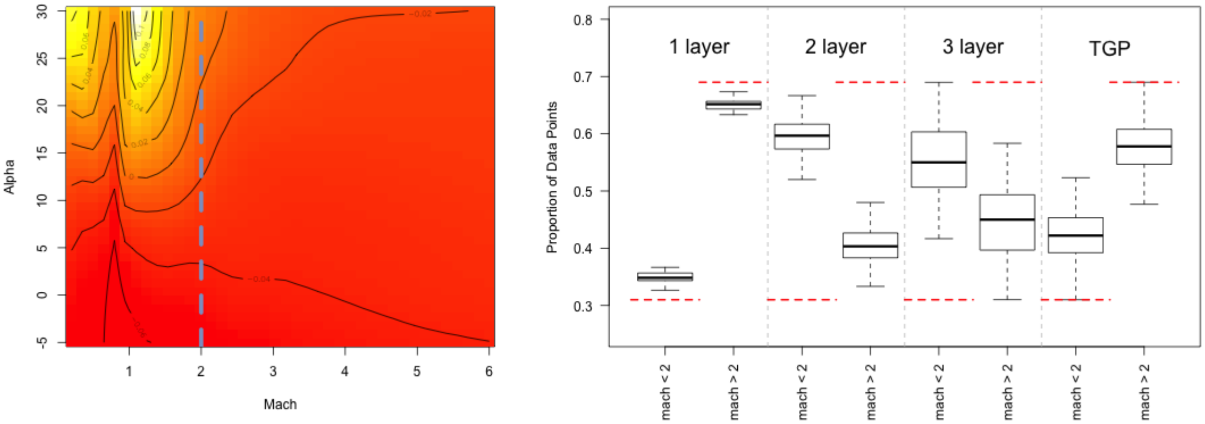


Figure 6: *Left:* Visual of the LGBB side response for fixed beta = 4. White/high, red/low. A vertical dotted line splits the input region at mach 2. *Right:* Proportion of 300 sampled points falling left/right of mach 2. Boxplots represent spread over 30 repetitions. Red dotted lines mark actual proportion of input space.

Figure 7 tracks out-of-sample RMSE via averages over thirty such MC repetitions (left), and the spread of values after the final AL acquisition (right). The relative prowess of each method can be explained by differences in the allocation of design points in the two input regions of Figure 6 [right panel]. The stars in the figure were an addition requested by the editorial team. These correspond to a static, non-AL, design of size $n = 300$, making clear the value of AL across the board. They drive home one of the main takeaway messages; AL

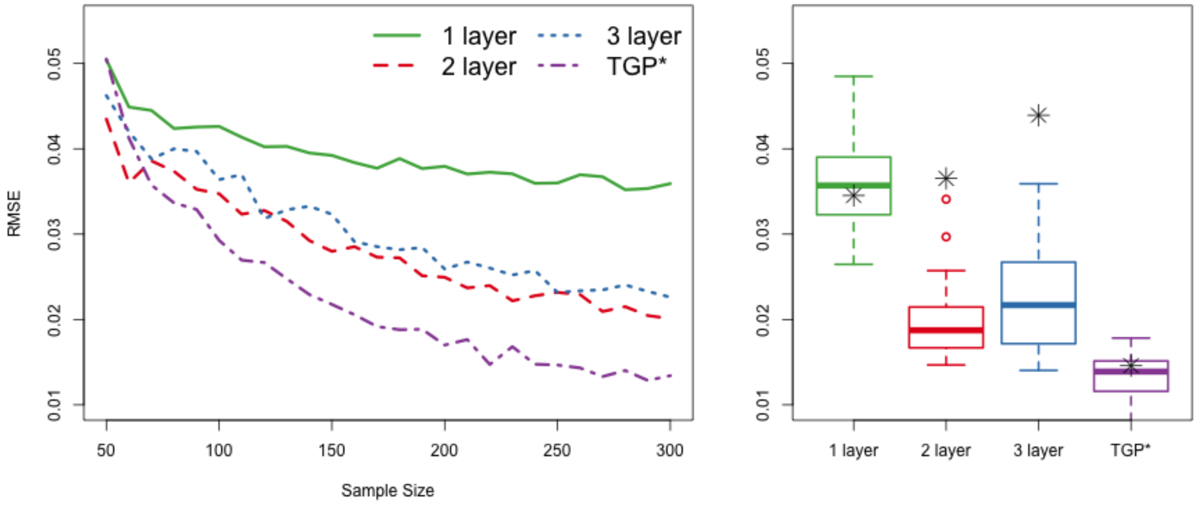


Figure 7: *Left:* RMSE on out-of-sample 1000 point testing sets for the LGBB computer experiment, averaged over 30 repetitions. *Right:* spread of final RMSE at $n = 300$ across 30 repetitions. Black stars denote the RMSE obtained from static 300-point designs. Both one-layer MLE variants performed equivalently to the one-layer MCMC fit.

benefits DGP models, and vice versa. If you're paying attention, you'll notice in Figure 7 that TGP wins the comparisons, but it benefits less from AL. (An ordinary, one-layer GP doesn't benefit at all.) It's hard to beat a model that leverages hard partitioning when the data exhibit sharp regime changes. I was attempting to manage the reader's expectations earlier when I said that TGP "makes for a formidable comparator" on this problem.

6.2 Satellite drag

Both Sauer et al. (2022) and Sauer et al. (2023) looked at this example. The passages here paraphrase both for variety. I begin with a description of the simulator, followed by AL and large-data surrogate modeling examples.

Researchers at Los Alamos National Laboratory developed the *Test Particle Monte Carlo* simulator to compute drag coefficients for satellites in low Earth orbit. These are useful in the development of positioning and collision avoidance systems. Sun et al. (2019) provide an R wrapper (`https://bitbucket.org/gramacylab/tpm/`) and several caches of runs. The simulator relies on a geometric satellite specification, atmospheric composition, and seven input variables. See Sun et al. (2019) and Gramacy (2020, Chapter 2) for a thorough discussion of the simulator and its inputs. Both examples below consider the GRACE satellite, and both may be found in the authors' Git repo, provided earlier.

Active learning

Mehta et al. (2014) showed that over a restricted portion of the input space, a GP surrogate trained via 1000-point LHS is able to predict drag within 1% root mean square percentage error (RMSPE). Sauer et al. (2022) began with a 7d LHS of size $n_0 = 100$, plus ten randomly chosen replicates (effectively $n_0 = 100$) to help separate signal from noise. For out-of-sample testing, via RMSPE, they engaged a novel LHS of size $n_p = 1000$. Figure 8 shows these RMSEs over the course of acquisitions, $n = n_0, \ldots, 5000$, in ten repetitions, for variations on one and two-layer DGPs (different numbers of nodes p of $\mathbf{W}$), and TGP.

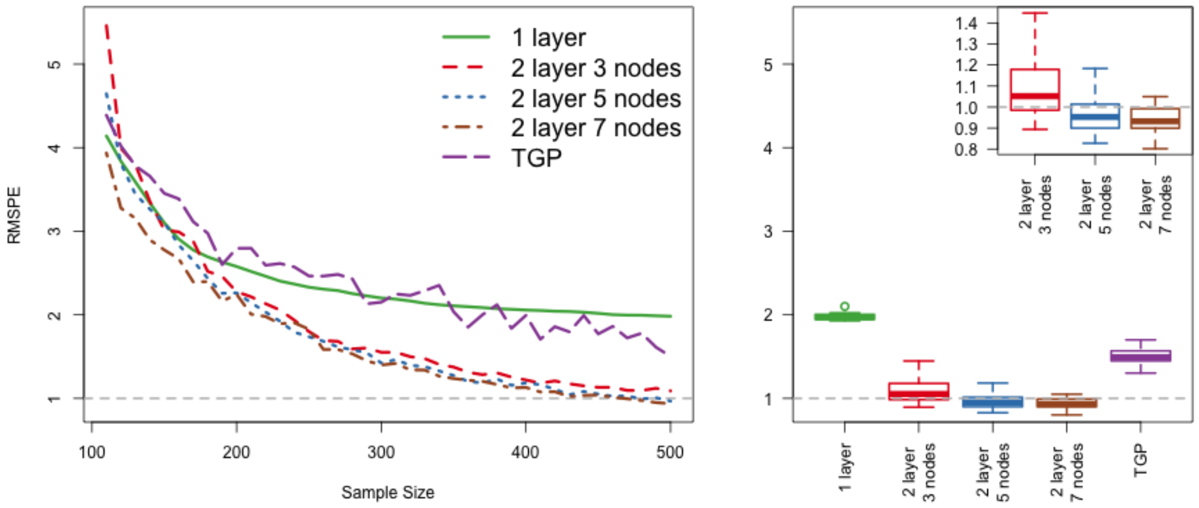


Figure 8: *Left*: average RMSPE for sequential design of the satellite drag simulator over ten repetitions. *Right*: spread of final RMSPE at $n = 500$ across 10 repetitions. Dashed lines highlight the 1% goal. The top-right sub-panel zooms in on the best comparators.

Observe that all but the one-layer (ordinary GP) and TGP comparators were able to achieve the 1% benchmark using fewer than 500 runs of the simulator. Interestingly, it is possible to accomplish this feat with a latent dimension of $p = 3 \ll d = 7$. Accuracy improves with wider $\mathbf{W}$, but $p = 3$ is two times faster than $p = 7$ on an 8-core machine.

Scaling up

Whereas Mehta et al. (2014) involved small designs ($n \leq 1000$) and a restricted input space, the analysis here extends to the full input space. Sun et al. (2019) used locally approximated GPs (Gramacy and Apley, 2015) and required one million runs to reach the 1% RMSPE benchmark. Later, Katzfuss et al. (2020a) improved on both with Vecchia-GPs (GP SVEC). Here, we reproduce the analysis showing that Sauer et al. (2023)'s Vecchia-DGP is able to beat the 1% RMSPE benchmark with as few as $n = 50,000$ (and can beat it consistently with $n = 100,000$), and provide better UQ than the stationary GP SVEC alternative.

Results for thirty MC repetitions are displayed in Figure 9. DGP DSVI and DGP HMC results are omitted because they were not competitive, yielding RMSPE's upwards of 35%. DGP VEC consistently outperforms the shallow/stationary GP SVEC and is able to achieve the 1% RMSPE goal with as few as 50,000 training observations. Compared to GP SVEC counterparts (with matched training/testing data), DGP VEC models have lower CRPS in

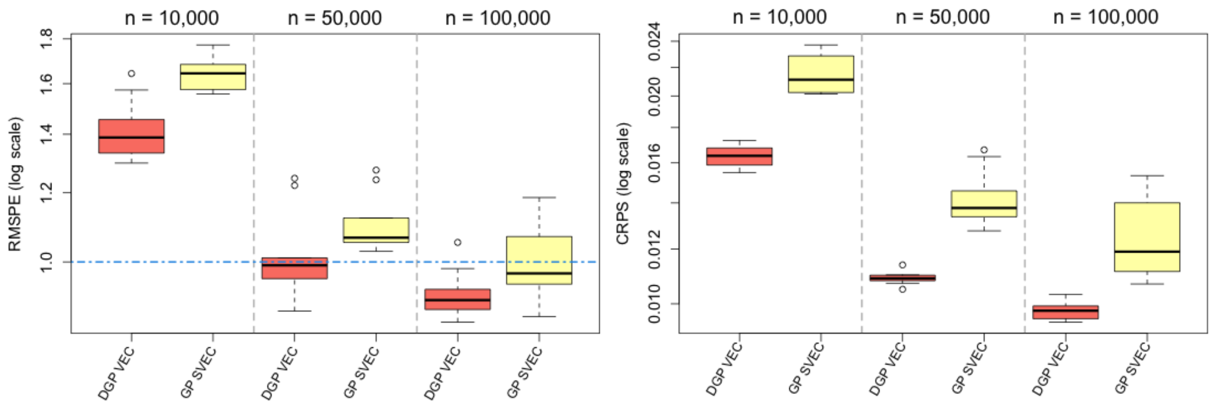


Figure 9: RMPSE (left) and CRPS (right) on log scales for satellite drag fits. DGP DSVI and DGP HMC are omitted. Horizontal line marks 1% RMSPE.

all thirty MC repetitions, and lower RMSPE in 28 of the 30. Computation times are provided with Sauer et al.'s supplementary material.

7 Discussion

It's customary to begin the final section with a summary. Reviewing important points at the end of a lengthy presentation is a good idea, especially in a talk (in case you put your audience to sleep), but also in a paper. But take care to offer something unique here. You don't want to waste your readers' time with a "re-abstract". They can quickly flip to the first page for that. After you review things, you might offer some reflections on extensions, special cases, address any speculative insights that you carefully avoided earlier. I'll try to wrap up my tutorial here after that fashion.

This paper attempts to be a weave of advice and examples, providing the structure and substance of a research paper while at the same time being a tutorial, for how to write a paper for *Technometrics*. I think this advice is applicable more widely than that, especially if your contribution aims toward practical, and empirically demonstrated, novel methodological contribution. The document is quite opinionated, and I'm sure my colleagues will quibble with aspects. The advice is certainly not official, but it has worked for me. You might ultimately find that you prefer another recipe. But if you're new to statistical publishing, or new to *Technometrics*, it can't hurt to begin by emulating someone who has had success before going maverick. As with any research paper, it's important to read for inspiration, and with extrapolation in mind. Your methodology is novel, so aspects of your presentation will also be novel no matter how hard you try to follow a familiar roadmap, such as the one outlined here. That's fine, but I think this is still a good place to start.

A typical *Technometrics* paper is less than thirty double-spaced pages in 11-12pt font. This tutorial uses 12pt font, and is single-spaced. So it's about the right length, perhaps a little on the long side. It's always better to start long and then pare back. It's common for referees to ask for more in the same amount of space. Asking them to read too much will bias them against you; take advantage of the supplement. My supplement will discuss writing a rebuttal to accompany an invited revision. Most *Tech* papers, like this example, are thinner in the middle (e.g., Sections 3–4 on mathematical exposition, theory, etc.) than at the tails (Sections 1–2 and 5–7) because those latter sections provide the intuition and empirical evidence, which *Tech* readers prefer over math and theory. That doesn't mean you shouldn't have meat in your sandwich. Just remember who your audience is. Write with mathematical precision and sophistication, but don't go off on a theoretical tangent just to show us how acrobatic you are with sums.

The last thing I want to touch on is the submission process. *Technometrics* requires double-blind reviewing, so that referees (not Editor or AE) can't easily determine the authors' identities, and vice versa. That means not including author names on the title page, nor referring to anything in the document that makes it obvious who the authors are, e.g., references to software/data on the internet. This last bit is hard, almost impossible. The best solution is to provide broken, blinded links to the code, and submit a blinded archive along with the paper. You can explain in your cover letter. Also, your cover letter should explain anything else that might not be obvious from the abstract/introduction. For example, if most of the work was done by a student, it doesn't hurt to say so. Don't bore the Editor/AE with a copy of the abstract, but this is a good place to let them know about potential conflicts of interest, or why you think *Tech* is a particularly good outlet.

One may naturally wonder if enforcing double-blinding is really practical in modern times. Anyone who wants to hide their identity can do so – even from the Editor/AE – if desired. Just remove names/write pseudonyms on the title page and cover letter, and similarly provide other fake details (pseudo-Gmail address, etc.) at submission time. On the other hand, anyone who's proud of their research will ensure that it gets out there, with their real names attached. They'll want to expand their impact by promoting their work early and often: giving talks and other presentations at conferences and workshops with public scope (advertised on the internet); putting it on their website/arXiv[13] so that curious attendees/readers can find the details. Publishing on arXiv is a great way to ensure you aren't scooped. Hiding your idea from the public during the review process exposes you to risk on multiple fronts. Any referee who doesn't Google the paper title and other keywords in order to investigate the degree of novelty and pedigree of your ideas isn't fully executing their role. In so doing, they'll likely find your paper and discover your identity. Or at least one should hope so. Suppose they find nothing. Is that a good thing? They've just learned that you, the author, are not excited about your work. They might do their best to ignore this, and they should if they're being fair, but they're only human. Putting faith in their ability to overlook your shyness about publicizing your ideas is naïve. On aggregate, the unavoidable sentiment left by this void will negatively affect your scores. You're better off putting it out there with pride.

[13] `https://arxiv.org/`

Acknowledgements

Refer to your grants, with funding institution and reference numbers here. This is probably a requirement you agreed to when you, or your advisor, accepted the funding. This work was supported by U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research and Office of High Energy Physics, Scientific Discovery through Advanced Computing (SciDAC) program under Award Number 0000231018. Thank your referees, and other members of the editorial team after revision. If anyone read over this for you (thanks Gramacy Lab reading group!), or was involved in other discussions, thank them too. I'd like to thank Matt Plumlee and Kamran Paynabar for planting the seed of the idea for this tutorial. I'm sure it turned out different than they'd have imagined, and perhaps they'd vehemently disagree with many of my opinions. So be it.

References

Allaire, J., Xie, Y., McPherson, J., Luraschi, J., Ushey, K., Atkins, A., Wickham, H., Cheng, J., and Chang, W. (2018). `rmarkdown`*: Dynamic Documents for R*. R package version 1.10.

Asuncion, A. and Newman, D. (2007). "UCI machine learning repository."

Cole, D. A., Gramacy, R. B., and Ludkovski, M. (2022). "Large-scale local surrogate modeling of stochastic simulation experiments." *Computational Statistics & Data Analysis*, 174, 107537.

Dutordoir, V., Salimbeni, H., Hambro, E., McLeod, J., Leibfried, F., Artemev, A., van der Wilk, M., Hensman, J., Deisenroth, M. P., and John, S. (2021). "GPflux: A library for deep Gaussian processes." *arXiv preprint arXiv:2104.05674*.

Fearnhead, P., Davidson, A., Gessner, R., and Titterington, D. (2021). "Some notes on Biometrika style." See biometrika.zip link on `https://academic.oup.com/biomet/pages/General_Instructions`.

Gneiting, T. and Raftery, A. E. (2007). "Strictly proper scoring rules, prediction, and estimation." *Journal of the American statistical Association*, 102, 477, 359–378.

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

Gramacy, R. B. and Apley, D. W. (2015). "Local Gaussian Process Approximation for Large Computer Experiments." *Journal of Computational and Graphical Statistics*, 24, 2, 561–578.

Gramacy, R. B. and Lee, H. K. H. (2009). "Adaptive Design and Analysis of Supercomputer Experiments." *Technometrics*, 51, 2, 130–145.

Guinness, J., Katzfuss, M., and Fahmy, Y. (2021). *GpGp: Fast Gaussian Process Computation Using Vecchia's Approximation*. R package version 0.4.0.

Havasi, M., Hernández-Lobato, J. M., and Murillo-Fuentes, J. J. (2018). "Inference in deep gaussian processes using stochastic gradient hamiltonian monte carlo." In *Advances in neural information processing systems*, 7506–7516.

Katzfuss, M., Guinness, J., and Lawrence, E. (2020a). "Scaled Vecchia approximation for fast computer-model emulation." *arXiv preprint arXiv:2005.00386*.

Katzfuss, M., Jurek, M., Zilber, D., and Gong, W. (2020b). *GPvecchia: Scalable Gaussian-Process Approximations*. R package version 0.1.3.

Marrel, A., Iooss, B., Laurent, B., and Roustant, O. (2009). "Calculations of Sobol indices for the Gaussian process metamodel." *Reliability Engineering & System Safety*, 94, 3, 742–751.

Mehta, P. M., Walker, A., Lawrence, E., Linares, R., Higdon, D., and Koller, J. (2014). "Modeling satellite drag coefficients with response surfaces." *Advances in Space Research*, 54, 8, 1590–1607.

Morris, M. D. and Mitchell, T. J. (1995). "Exploratory designs for computational experiments." *Journal of statistical planning and inference*, 43, 3, 381–402.

Murray, I., Adams, R. P., and MacKay, D. J. C. (2010). "Elliptical slice sampling." In *The Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, vol. 9 of *JMLR: W&CP*, 541–548. PMLR.

Pamadi, B., Covell, P., Tartabini, P., and Murphy, K. (2004). "Aerodynamic characteristics and glide-back performance of Langley glide-back booster." In *22nd Applied Aerodynamics Conference and Exhibit*, 5382.

R Core Team (2019). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Salimbeni, H. and Deisenroth, M. (2017). "Doubly stochastic variational inference for deep Gaussian processes." *arXiv preprint arXiv:1705.08933*.

Santner, T., Williams, B., and Notz, W. (2018). *The Design and Analysis of Computer Experiments, Second Edition*. New York, NY: Springer–Verlag.

Sauer, A. (2020). `deepgp`*: Sequential Design for Deep Gaussian Processes using MCMC*. R package version 0.1.0.

Sauer, A., Cooper, A., and Gramacy, R. B. (2023). "Vecchia-approximated Deep Gaussian Processes for Computer Experiments."

Sauer, A., Gramacy, R. B., and Higdon, D. (2022). "Active learning for deep Gaussian process surrogates." *Technometrics*, 0, 0, 1–15.

Shmueli, G. (2021). "INFORMS Journal on Data Science (IJDS) Editorial #1: What Is an IJDS Paper?" *INFORMS Journal on Data Science*, 0, 0, 1–3.

Sun, F., Gramacy, R. B., Haaland, B., Lawrence, E., and Walker, A. (2019). "Emulating satellite drag from large simulation experiments." *SIAM/ASA Journal on Uncertainty Quantification*, 7, 2, 720–759.

Surjanovic, S. and Bingham, D. (2013). "Virtual library of simulation experiments: test functions and datasets." `http://www.sfu.ca/~ssurjano`.

Xie, Y. (2016). `bookdown`*: Authoring Books and Technical Documents with Rmarkdown*. Chapman and Hall/CRC.

— (2018). `bookdown`*: Authoring Books and Technical Documents with Rmarkdown*. R package version 0.9.

SUPPLEMENTARY MATERIAL

Technometrics has special instructions for how to organize supplementary material, and I won't rehash those here. If you've gotten so far through the process that they're micromanaging you about your supplement, then you're going to be ok. What I want to talk about here is what to do when your original submission gets rejected, but the Editor has expressed interest in seeing a substantially revised version. This happens for about 25% of submissions. The other 75% are outright rejects. In that first 25% of cases, the verbiage in the letter from the Editor is often harsh, but it's actually good news. This is a tough thing to wrap your head around as a new author. Final acceptance is by no means guaranteed, but at the same time final acceptance is yours to loose. Don't give up at this stage, which happens in about half (of those 25%) of cases. Do all the things s/he and the rest of the editorial team say to do, and you'll probably get in. Take this next step seriously.

How to revise/write a rebuttal

Begin with the attitude that the referee is always right, even though that's of course nonsense. Every query from a referee should result in a meaningful upgrade in the paper, and a polite response in a rebuttal document explaining what you did. If the referee has misunderstood something, that's probably your fault. Explain it to them, and to your readers, by wording things better. Telling a referee they have misunderstood will get you nowhere. Thank them for their insightful comment, and make a change in the paper.

It can help to highlight upgrades with color. Blue for additions is common, using the add-on LaTeX `changes` package. You can also use red for passages that were deleted, but this is less common. Show the editorial team, via lots of blue text that can be seen at a glance, that you've made many upgrades. In your rebuttal, point them to specific passages (sections, pages, paragraphs, equations) where they can find revisions you've made. Often times you'll need to find economies, or move material to a supplement, to make space for additions in your revised manuscript. It's not uncommon for an Editor to ask for both new material, and a reduced (or no larger) footprint. You may have to get creative.

Sometimes you can address a referee's concern without making a change/addition to the paper. Perhaps they're wondering what would happen in a certain scenario, but you think that addressing that in the paper would be a distraction. Remember, you are the author of the paper, and your vision for the paper is paramount. The best way to make a case for *not* including the suggested changes in the paper is to do them – run the new simulation, or develop the new math – and put the results in the rebuttal, and explain that they wouldn't fit well in the paper (or the supplement). If you say "no" without doing the work, the team will either think that you can't do the work, or that you are hiding what it might reveal, and they will ding you. Don't make that mistake.

If you're not addressing all referee comments in a way that would seem favorable to all members of the editorial team, then you risk getting a final rejection. Pay attention to the spirit and the letter of requests. If they ask for something big, like a new simulation, do it. If they ask for something little, no matter how silly it is, do it. Referees often complain about my use of colloquialisms in my writing. I like them because I think they're disarming, and bring life to an otherwise stoic presentation. But not everyone does, and when a referee says so I gratefully thank them and make the change(s). Take your time, be thorough, make a list, and knock things off one at a time. *Technometrics* gives you six months to make a revision because they realize that these things take time. But don't procrastinate. Don't spend six months doing two weeks worth of work.

Example rebuttal entries

Start a rebuttal document by pasting comments from the editorial team. I work in LaTeX for the rebuttal, because I usually need to provide figures and math, but this is not essential. Sometimes a plain text file is just fine. The important thing is that it be easy to follow. Make sure it's clear which comment is coming from which member of the editorial team. I use separate section headings for the Editor, AE, and Referee 1 and 2, in that order. Each of those sections begins with a polite header thanking them for their insights, explaining the layout to follow, and covering any broad themes as necessary. Be clear to delineate their comments from your responses. I like to put the comment in italics, and the response in plain text. Below are a few examples.

The examples below come from a rebuttal between the first and second submissions of Sauer et al. (2023). The rebuttal document totaled 11 pages in the same format as this tutorial (12pt, single-spaced). So it was quite substantial. It opens with a brief overview to the Editor, then iterates through comments/responses to AE, Referee 1 and Referee 2. One example is shown from each.

AE *My main concern is the novelty of this paper because neither the Bayesian DGP inference for computer experiments nor Vecchia's approximation is new. While the proposed method seems interesting and useful, it is built on existing methods. The authors are suggested to provide further developments or more convincing results.*

Thank you for carefully reading our paper. Your assessment is accurate – the core of our contribution is in the combination of two existing frameworks (Bayesian DGPs and Vecchia approximation). Although neither of the two components are "novel", we believe their integration (with careful attention to computational considerations) will have broad impact. This kind of integration is easy to say, but hard to do.

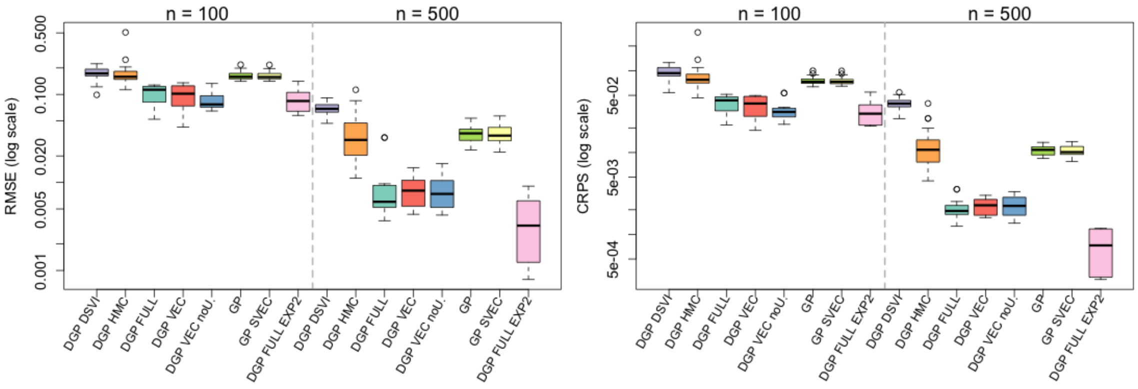
We have implemented several upgrades to the manuscript in an effort to provide further developments and more convincing results. First, we have updated our illustrative figures to better reveal our particular approach to DGP modeling and inference, which (compared to other frameworks) is uniquely well-suited to Vecchia approximation, and to improve accessibility to a wider audience. We doubled the number of the benchmarking MC exercises in Section 5.1 and added an additional un-approximated GP comparator (for the data sizes that were small enough to be feasible). We investigated the choice of conditioning set size m in a new Supplement D. We included a larger, six-dimensional simulated exercise in a new Supplement E. We also now report computation times for all simulated exercises in Supplement F.

Ref 1 *The numerical results look good, but I'm frustrated not knowing how long the various methods take to run. This is a computational journal after all. Vecchia approximations have a built-in mechanism for trading off speed vs accuracy, namely the number of neighbors in the conditioning set. If one method is faster, can it be made to be more accurate (and faster) by increasing m? I'm wondering if that's true for GP SVEC.*

We have added a section in the supplementary material to report the computation times for each of our simulation studies. You're right – in theory a larger m should result in more accurate predictions. To study whether this pays off in practice we included a full-GP comparator (i.e., $m = n - 1$) for the data sizes which were feasible. We found that the full GP compared similarly to the scaled Vecchia GP on the two-dimensional Schaffer function (Figure 4), but outperformed it on the four-dimensional G function (Figure 5). A fuller study varying m is now in Supplement D. These preliminary results suggest that m may play a larger role in higher-dimensional problems. We added such discussion and recommendations to Section 6.

Ref 2 *On page 20, the authors mentioned "Finally, at the outset we expected the stationary GP (GP SVEC) to perform poorly given the complexity of the response surface, but were surprised to see GP SVEC holding its own against DGP HMC." I am not sure if this statement is fair because DGP HMC used the squared exponential kernel, but GP SVEC used Matèrn. It may be attributed to the effect of using two different kernels.*

To investigate the effect of the two different kernels, we ran our DGP FULL model with both the Matèrn $\nu = 5/2$ kernel (DGP FULL) and the squared exponential kernel (DGP FULL EXP2). The figure below shows the RMSE/CRPS for the DGP FULL EXP2 model next to the results from the 7 models that we reported for the two-dimensional example of Figure 4. We were surprised to see that our DGP performed better with the squared exponential kernel, not worse. As you suggested, most people assume the Matèrn $\nu = 5/2$ kernel would outperform the squared exponential kernel. But it also makes sense that this can't *always* be the case, and it would appear that the Schaffer function is an example which benefits from an infinite smoothness assumption.



> Software limitations restrict us from further comparisons of the kernels (specifically for our Vecchia approximation; our implementation relies on some dependencies that are not implemented for the squared exponential kernel, though maybe they should be in light of these results). More to the point of addressing your query: these results suggest that the DGP HMC model, if anything, should have an advantage from the use of the squared exponential kernel in this instance. It's surprising that it does not.

Notice that in each case our response is polite, acknowledges that the referee has a valid point, thanks them for it, and goes on to describe how we did new work to address the concern. The first two describe new simulations that were performed to investigate the phenomena under question. Specific pointers are made to where in the paper the changes where made/results could be found. In the last case, in response to Referee #2, work is described and shown directly in the supplement, rather than as a revision in the paper. We thought it would be a distraction to describe these results in the paper. I believe the author should reserve the right to do this when the they feel strongly about it. But you have to bring the editorial team onto your side. Although we did not include these results in the paper, we still did the work, and I think that's key! The results actually indicate the referee was wrong, but note we did not point that out. Instead we described them as "surprising". In this instance, that surprise was both genuine and tactical. Also, we apologized for not being able to go further. Sometimes referees ask, explicitly or implicitly, for too much. It's ok to let them know that you can't do it all at this time, but do so sparingly. In this case, we made sure to demonstrate that we had done substantial new work, and that it could have been taken further, but there were logistical obstacles.

Sometimes all the referee wants is for you to fix a typo or add some verbal explanation. Those usually don't take as much work. But it still pays to describe what you did to make clear that what was done, and to take the opportunity to show the referee you value their effort. Simply writing "Thanks" at the start of every response is lazy, and not good enough. Make it obvious with your effort that the remark was valuable, and that addressing it led to improvements in the paper without going overboard in saying so explicitly.