Alternative Bandwidth Optimization Methods for Geographically Weighted Regression

Tyler D. Hoffman

3 December 2021

1 Introduction

Geographically weighted regression (GWR; [4]) is a widely used statistical technique for analyzing spatial process variation and spatial relationships. The framework broadens a classical regression by allowing for spatially varying coefficients that illustrate different effects in different areas of the spatial domain. These coefficients are calibrated alongside a bandwidth parameter that controls the amount of spatial contextual information at a given location. During the fitting procedure, a local regression is fitted at every spatial observation using the information allotted by the bandwidth. Due to the influence it has on this process, the bandwidth parameter can be thought of as an indicator of the scale at which the process operates.

To find the bandwidth parameter, the fitting procedure makes use of a version of the Akaike information criterion (AIC; [1]) that incorporates a penalty term to disincentivize overly complex models. The GWR-corrected AIC, or AICc, takes the form [4]

$$\text{AICc} = -2\log f(y|\beta,\sigma^2,X) + \frac{2n(k+1)}{n-k-2}$$

where $f(y|\beta,\sigma^2,X)$ is the likelihood function for the model, n is the number of observations (i.e., areal units), and k is the effective number of parameters for the model. This paper regards the AICc as a function of the GWR bandwidth, which controls the amount of data used at a given observation to calibrate its local model. In GWR, the optimal bandwidth—the bandwidth that the model ultimately reports—is the one that minimizes the AICc.

Theoretically, this minimization problem is easy to frame. In practice, however, the computation can get messy. Since the AICc does not have an analytical derivative with respect to the bandwidth, one must use a derivative-free optimization method to find its minimizer. As a consequence, the performance of this optimizer is severely constrained: no derivative information means that an optimization method must learn the shape of the objective on its own or use heuristic or naive methods to find a minimum. These handicaps can be costly in terms of performance and make it difficult to quickly and accurately solve the minimization problem.

Additional constraints or information about an optimization problem always help refine the optimization strategy. By exploiting elements of the problem's structure, it is possible to customize faster and more powerful optimization methods. Such information is even more important when the objective lacks a derivative, as the optimizers must make up for not knowing the slope of the function.

While many improvements have been made to GWR since its inception (in particular, the generalization of the method into MGWR [8]), this paper focuses on the original procedure for simplicity. Indeed, this paper aims to deconstruct the optimization procedure for the GWR bandwidth and to examine the interplay between the structure of the problem and solution methods. By analyzing the problem's structure, it is possible to extract more insights

about viable research directions for more efficient and reliable optimization techniques.

2 Problem Structure

Following ideas developed in [6], it is natural to consider the AICc as a combination of two terms acting independently: the log-likelihood (as a proxy for model fit) and the GWR correction (as a penalty for overfitting). Both of these vary as a function of bandwidth, but the way in which they vary differs between the two. The GWR correction is a monotonically increasing function of k and blows up at $k = n - 2$ due to its denominator:

$$\lim_{k \to n-2} \frac{2n(k+1)}{n-k-2} = \infty$$

but recovers "normal" behavior after moving beyond this asymptote. Recall that k is the effective number of parameters for a particular bandwidth; that is, the number of independent coefficient estimates calibrated by the model. Thanks to the generality of GWR, the model reverts to a global model when the bandwidth is infinite—that is, when all the locations borrow data from each other, $k = p$ (the number of covariates at the global level). Conversely, when the bandwidth is 0, each location has a completely independent regression calibrated at it and therefore there are effectively n independent parameters being calibrated. Hence, $k \in [p, n]$ is a decreasing function of bandwidth and we have that the correction is a decreasing function of the bandwidth as well.

In fact, the GWR correction is a very well-behaved decreasing function of the bandwidth. The log-likelihood, on the other hand, evades such guarantees. As such, the log-likelihood drives the AICc curve

- Decompose the AICc
- Outline the three components that really affect optimization here: minima, steepness, and noise

2.1 Minima

- start with what we know (first three bullets on that slide) which is the foundation for how GWR works
- spend some time discussing multiple minimizers
- explain why this is dangerous for an optimizer / how an optimizer should account for it

2.2 Steepness

- Emphasize that due to fixed (monotonic) nature of the correction, the steepness is nearly completely determined by the log-likelihood. Process variation is the name of the game
- Bandwidth CI issues—CIs can be wide even if the optimum is well-defined. How does Golden section search address this at present?
- explain why this is dangerous for an optimizer / how an optimizer should account for it

2.3 Noise

- explain the way that noise develops in these curves—due to the way areal units work
- explain why this is dangerous for an optimizer / how an optimizer should account for it

3 Methods

3.1 Optimizers

This work tests four optimizers for finding the optimal bandwidth. First, grid search (brute force optimization) is presented as a baseline for performance benchmarking. Next, a version of golden-section search [7] similar to the

implementation in the Python `mgwr` module was tested [8]. Golden-section search looks for the minimum of a unimodal function in a specified interval by repeatedly bisecting that interval according to the golden ratio ($\phi = (1 + \sqrt{5})/2$). By enforcing this proportion, the search algorithm offers a good rate of convergence to the minimum without sacrificing much in the way of accuracy. However, when applied to multimodal functions such as the AICc curve, golden-section search is known to get trapped in local minima. One proposal to avoid this is to use different initial points for golden-section search and to use the best result. More research is required to determine what guarantees this lends to the search method.

Another prototypical derivative-free optimizer, Brent's method, was also tested [3]. Brent's method combines the bisection method and a quadractic version of the secant method (inverse quadractic interpolation instead of linear interpolation) by testing how much each method improved the solution at each iteration. By balancing the two, it has a worse worst-case complexity than the bisection method, but converges superlinearly if the objective is well-behaved.

Finally, particle swarm optimization (PSO) was considered as a heuristic approach to optimization [2]. PSO lets loose a swarm of particles (candidate solutions) in the domain which search for minima by moving with a certain amount of velocity. On each iteration, the particles use this built-in velocity, knowledge of their own best position, and knowledge of the swarm's best position to determine where to move. As a result, PSO can effectively explore local minima: as long as one particle reaches the absolute minimum, the swarm has succeeded. Additionally, due to the highly parallel behavior of the particles, PSO lends itself well to parallel computing implementations. These can improve the runtime of the algorithm, but do not improve the number of objective calls done by the algorithm. While these parallel methods are not explored in this paper, they could offer significant practical speedups in implementation.

3.2 Sample Data

As explained in Section 2, the AICc curves typically look like smooth checkmarks. The baseline curve (Figure 1) was therefore chosen to be $b(x) = (x^2 - x + 600)/2x$ for $x > 0$ as it depicts well this smooth, unimodal shape. Subsequently, this objective was altered along the three axes of study: minima, steepness, and noise. All experiments were conducted with a low amount of added noise to simulate how real AICc curves behave.

To introduce multiple minima, a rule was prescribed that adds more checkmark-shaped curves along fixed displacements (Figure 2). The extra checkmarks take the form

$$g(x) = 0.0005((x - 120)/10)^8 + x - 65$$

and for a given set of new minima locations $\{a_i\}_{i=1}^m$ were displaced to create a set of functions

$$h_i(x) = g(x - a_i) + a_i/2, i = 1, \ldots, m.$$

Finally, for each x, the multiple minimum objective was defined as

$$b_m(x) = \min\{b(x), h_1(x), \ldots, h_m(x)\} + \text{noise}.$$

For steepness, the baseline curve was parametrized with an additional steepness factor that altered the slope of the curve's tail (Figure 3). These curves took the form

$$b_\sigma(x) = (\sigma x^2 - x + 600)/2x$$

where $\sigma > 0$ was the steepness parameter.

To add the noise, the generalized Hénon map was used [5]. The noise needed to be non-random so it could be controlled across all optimizers, so a hyperchaotic dynamical system was utilized to ensure the noise could not be predicted but were replicable. The discrete-time dynamical system takes the form

$$x_i = a - x_{i-1}^2 - bx_{i-2}$$

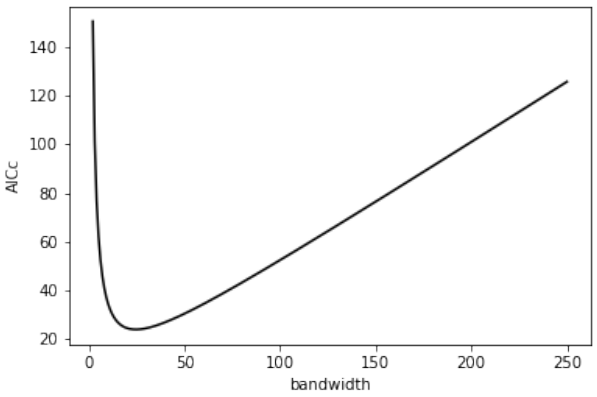

Figure 1: The base curve used for the numerical experiments.

where $a = 1.76$ and $b = 0.1$ are taken to be fixed parameters that are known to generate hyperchaos. To generate the noise, the system was run for 100 iterations using initial conditions $x_0 = 0$ and $x_1 = x$, where x is the point for which the noise was requested (Figure 4).

4 Results and Discussion

present results and explain what they mean. time/accuracy tradeoff (obv)

5 Conclusion

conclude, discuss limitations of this, and point to future work

References

[1] H. Akaike. "Information theory and an extension of the maximum likelihood principle". In: *2nd International Symposium on Information Theory, Tsahkadsor, Armenia, USSR, September 2-8, 1971* (1973), pp. 267–281.

[2] M.R. Bonyadi and Z. Michalewicz. "Particle swarm optimization for single objective continuous space problems: a review". In: *Evolutionary Computation* 1.25 (2017), pp. 1–54.

[3] R.P. Brent. "Algorithms for Minimization without Derivatives". In: Prentice-Hall, 1973. Chap. Chapter 4: An Algorithm with Guaranteed Convergence for Finding a Zero of a Function.

[4] A. S. Fotheringham, C. Brunsdon, and M. Charlton. *Geographically Weighted Regression: The Analysis of Spatially Varying Relationships*. Wiley, 2003.

[5] M. Henon. "A two-dimensional mapping with a strange attractor". In: *Communications in Mathematical Physics* 1.50 (1976), pp. 69–77.

[6] T. D. Hoffman and T. Oshan. "A Supervised Heuristic for a Balanced Approach to Regionalization". In: *GIS Research UK Conference Proceedings* (2021).

[7] J. Kiefer. "Sequential minimax search for a maximum". In: *Proceedings of the American Mathematical Society* 3.4 (1953).

[8] T.M. Oshan et al. "mgwr: A Python implementation of multiscale geographically weighted regression for investigating process spatial heterogeneity and scale". In: *ISPRS Journal of Geo-Information* 6.8 (2019), p. 269.

Large Figures

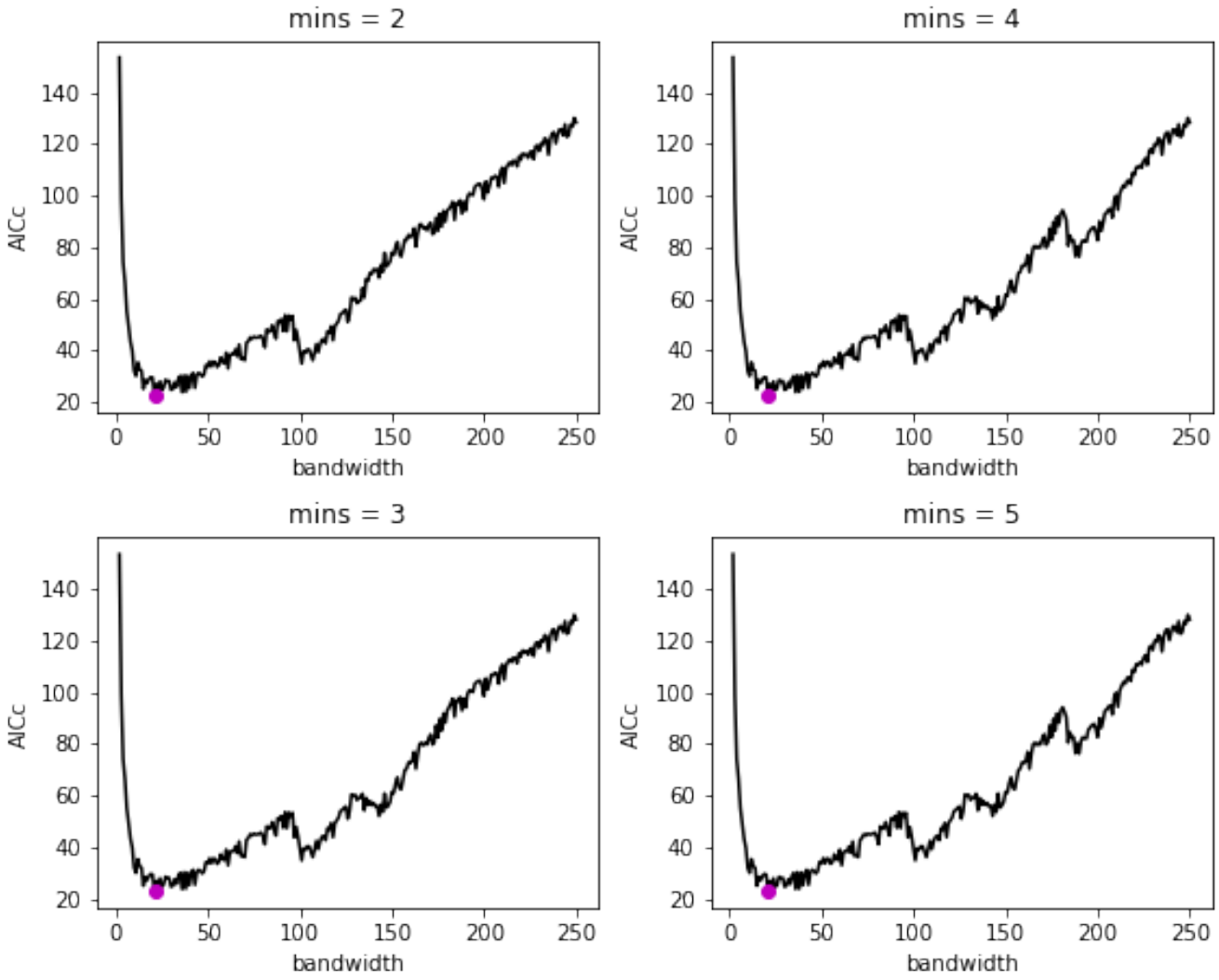

Figure 2: Examples of four numbers of minima used in the numerical experiments.

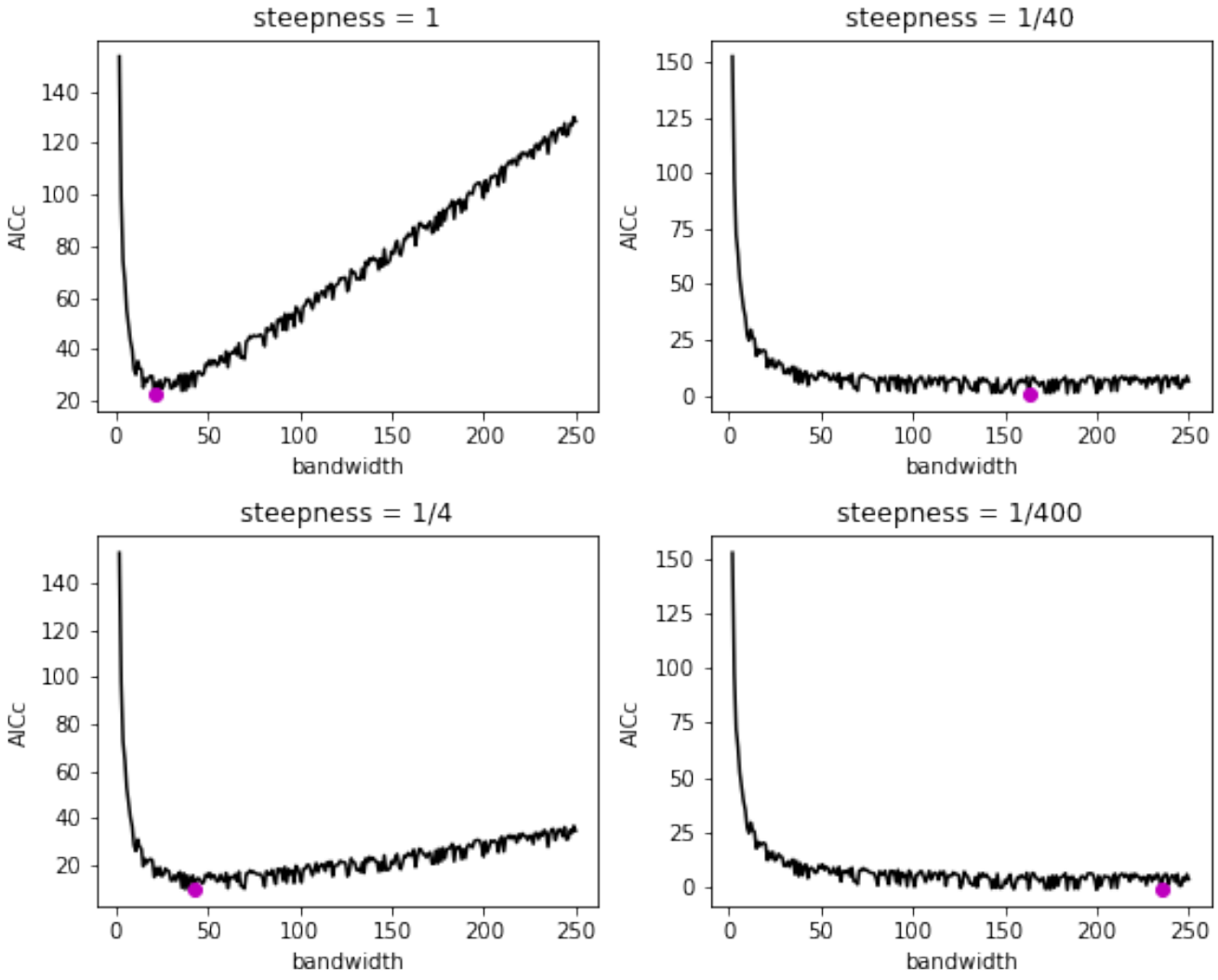

Figure 3: Examples of four steepness levels used in the numerical experiments.

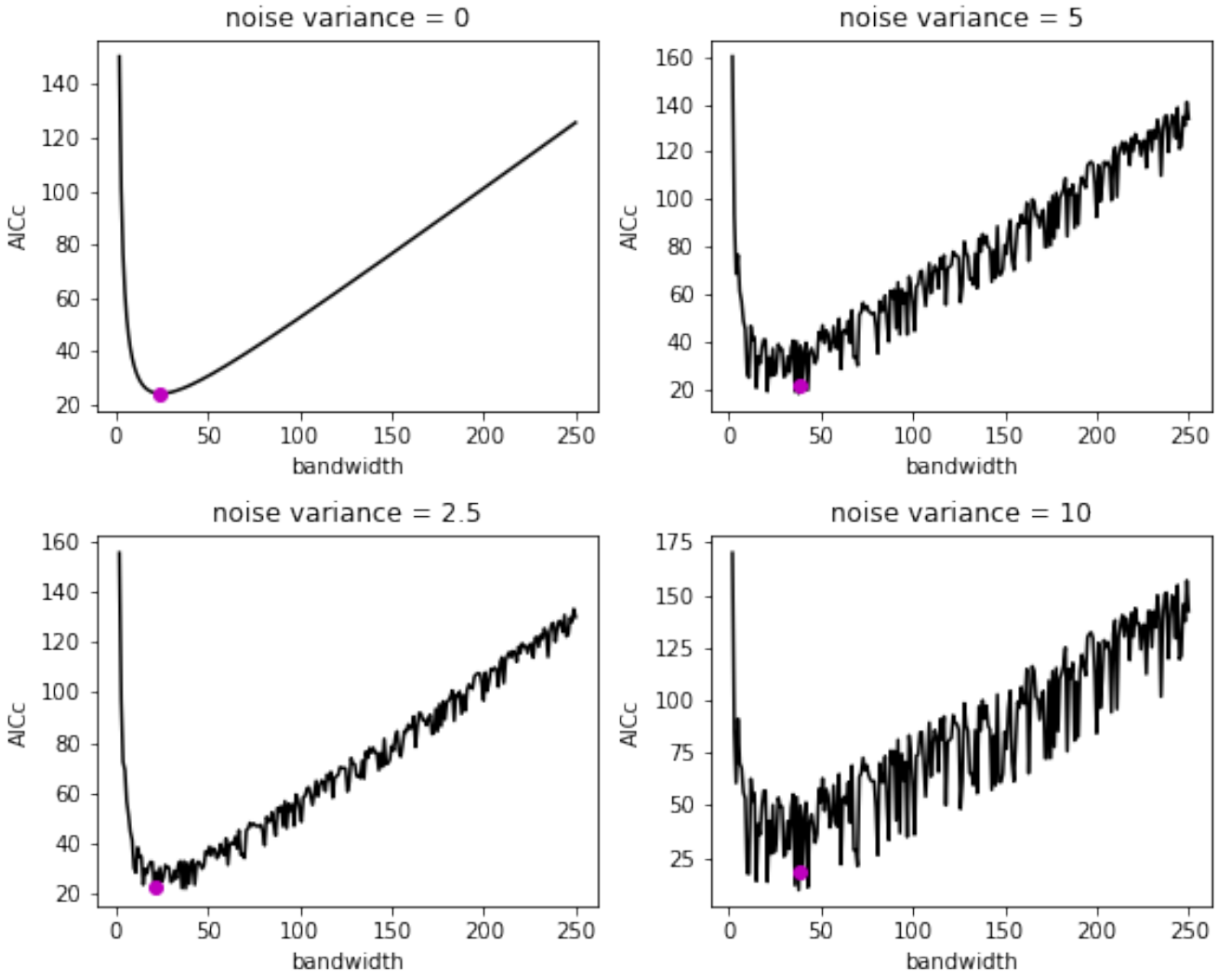

Figure 4: Examples of four noise levels used in the numerical experiments.