

FINAL REPORT

Using Unmanned Aerial Systems to Model Spatially-Mediated Heterogeneity in 3D Microclimate Landscapes

Anna Carter
Iowa State University

May 2022

This report was prepared under contract to the Department of Defense Strategic Environmental Research and Development Program (SERDP). The publication of this report does not indicate endorsement by the Department of Defense, nor should the contents be construed as reflecting the official policy or position of the Department of Defense. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the Department of Defense.

REPORT DOCUMENTATION PAGE					<i>Form Approved OMB No. 0704-0188</i>	
The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.						
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.						
1. REPORT DATE (DD-MM-YYYY) 10-05-2022		2. REPORT TYPE SERDP Final Report			3. DATES COVERED (From - To) 9/27/2018 - 9/20/2021	
4. TITLE AND SUBTITLE Using Unmanned Aerial Systems to Model Spatially-Mediated Heterogeneity in 3D Microclimate Landscapes					5a. CONTRACT NUMBER 18-P-0071	
					5b. GRANT NUMBER	
					5c. PROGRAM ELEMENT NUMBER	
					5d. PROJECT NUMBER RC18-1021	
6. AUTHOR(S) Anna L. Carter					5e. TASK NUMBER	
					5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Iowa State University of Science and Technology 1350 Beardshear Hall 515 Morrill Rd Ames, IA 50011-2105					8. PERFORMING ORGANIZATION REPORT NUMBER RC18-1021	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Strategic Environmental Research and Development Program 4800 Mark Center Drive, Suite 17D08 Alexandria VA 22350-3600					10. SPONSOR/MONITOR'S ACRONYM(S) SERDP	
					11. SPONSOR/MONITOR'S REPORT NUMBER(S) RC18-1021	
12. DISTRIBUTION/AVAILABILITY STATEMENT DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited.						
13. SUPPLEMENTARY NOTES Prepared with co-PI Fredric Janzen.						
14. ABSTRACT This project sought proof-of-concept for driving an ultra fine-resolution microclimate model with UAV-collected geospatial data, with the main goal of determining whether increasing spatial resolution - from 1m - .13m in this case - could detect micro-scale variation in soil temperatures. Substantial spatial differentiation between model outputs at these resolutions indicates that spatial resolution can affect the detection of biologically important variation in microclimate conditions, likely without significantly impacting model accuracy. An important implication of this result is the likely correctness of the ubiquitous assumption that spatial mismatch could be leading to erroneous ecological conclusions and conservation/management decisions. Given this likelihood, conservation and management decisions made using relatively coarse-resolution models should be considered tentative, at best. Preferably, the indications of a coarse model would be followed up with data that are not mismatched. The continuation of this research is critical for verifying and extending the findings to diverse and novel environments.						
15. SUBJECT TERMS climate change, microclimate ecology, remote sensing, unoccupied aerial vehicle (UAV)						
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 100	19a. NAME OF RESPONSIBLE PERSON Anna L. Carter	
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U			19b. TELEPHONE NUMBER (Include area code) (515) 735-2151	

Table of Contents

List of Tables.....	iii
List of Figures	iv
List of Acronyms.....	v
Keywords	vi
Acknowledgements	vii
Abstract	8
Executive Summary	9
Introduction	9
Objectives	10
Technical Approach	11
Results and Discussion	15
Implications for Future Research and Benefits	17
Objective	18
Background	19
Materials and Methods	20
Results and Discussion.....	24
Conclusions and Implications for Future Research/Implementation	27
Literature Cited	30
Appendices	33
A. Supporting Data	33
B. List of Scientific/Technical Publications	74
C. Other Supporting Materials	75

List of Tables

Table 1. Summary of predicted above-ground conditions output by the NicheMapR microclimate model. For this project, each variable will be output hourly, over one year, for each modeled spatial resolution. 13

List of Figures

- Figure 1.** Overlapping frequency histograms of maximum daily predicted substrate temperatures over 12 months [Jan (top left) – Dec (bottom right), 2017] at (a) 2.5cm and (b) 10cm depths. The black bars are predictions made using terrain inputs at a 1m spatial resolution, and the overlaid transparent purple bars show predictions made at 13cm. 15
- Figure 2.** Snapshot of microclimate model output, summarized from hourly data for a representative day (Julian Day 45: 14 Feb 2017). Data are maximum daily soil temperature at depths of (I) 2.5cm and (II) 10cm. The square plot shows data for a 50x50m area at a spatial resolution of 1m, with the white inset magnified at spatial resolutions of (a) 1m and (b) 13cm. The gray marginal plots show means across space. Note the ‘cool spot’ in Ib that is not apparent in the 1m plot, an example of variation in substrate temperatures detected only in the finer-resolution model. See additional plots in Appendix A..... 16
- Figure 3.** Flowchart showing the inputs required to run the NicheMapR microclimate model. The circled parameters are the focus of this project. The spatial resolution of output microclimate conditions is determined by the terrain inputs. The impacts of data availability (Task 1.4) will focus on the daily climate data. 21
- Figure 4.** Image showing the 50x50m data logger grid (N = 50) designed for a) KONZ and b) NIWO, as yellow points superimposed over a shaded surface relief of the 1km² sampling grid (surface data from NEONe)..... 23
- Figure 5.** Overlapping frequency histograms of maximum daily predicted substrate temperatures over 12 months [Jan (top left) – Dec (bottom right)] at (a) 2.5cm and (b) 10cm depths. The black bars are predictions made using terrain inputs at a 1m spatial resolution, and the overlaid transparent purple bars show predictions made at 13cm. 24
- Figure 6.** Snapshot of microclimate model output, summarized from hourly data for a representative day (Julian Day 45: 14 Feb 2017). Data are maximum daily soil temperature at depths of (I) 2.5cm and (II) 10cm. The square plot shows data for a 50x50m area at a spatial resolution of 1m, with the white inset magnified at spatial resolutions of (a) 1m and (b) 13cm. The gray marginal plots show means across space. The remaining plots are in the Appendices. 25
- Figure 7.** Raw substrate temperatures at two depths for KONZ, showing (a) all data for 10cm, (b) July data for 10cm, (c) all data for 20cm, and (d) July data for 20cm. Data logger IDs are matched with the grid coordinates of model predictions. 26

List of Acronyms

2D	two-dimensional
3D	three-dimensional
3DEP	3D Elevation Program
AED	above elevation data
AOP	aerial observation platform
ATP	automatic tie point
CORS	continuously operating reference station
CPU	central processing unit
CU	University of Colorado [Boulder]
DOD	[United States] Department of Defense
DSM	digital surface model
DTM	digital terrain model
FAA	Federal Aviation Administration
GADS	global aerosol dataset
GCP	ground control point
GIS	geographic information system(s)
GNSS	global navigation satellite system
GPS	global positioning system
GSD	ground sampling distance
IR	infrared
ISU	Iowa State University
KONZ	Konza Prairie [Biological Research Station]
KSU	Kansas State University
LiDAR	light detection-and-ranging
LTER	long-term ecological research
MSU	Michigan State University
NEON	National Ecological Observatory Network
NIWO	Niwot Ridge [Mountain Research Station]
NW	northwest
PI	principal investigator
PPK	post-processed kinematic
R	Pearson's correlation
RAM	random access memory
RGB	red-green-blue
RINEX	receiver independent exchange format
RMSD	root-mean-squared deviation
RTK	real-time kinematic
SERDP	Strategic Environmental Research and Development Program
SON	statement of need
SW	southwest
UAS/V	unoccupied aerial system(s)/vehicles(s)
USGS	United States Geological Survey
VO	visual observer
WGS84	World Geodetic System 1984

Keywords

aerial imagery, digital elevation model, drone, ecological niche modeling, ecophysiology, microclimate, NicheMap*, photogrammetry, remote sensing, unoccupied aerial system (UAS), UAV

Acknowledgements

We would like to acknowledge staff at the Kansas State University Konza Prairie Biological Research Station and the University of Colorado Mountain Research Station for assistance with field work logistics and Tristan Goulden, at the National Ecological Observatory (NEON), for continued advice about the aerial observation platform data. NEON also provided travel funding for the PI to attend a conference and workshop as, respectively, an Early Career Scholar and Accelerate Scholar. We would also like to thank Ryan Sanderson and Levi Baber of ISU ResearchIT for their continuing IT support and four determined field assistants, two of whom worked completely independently at 11,000ft ASL due to COVID-19 restrictions on backcountry field research. All drone flights were conducted using FAA Part 107 Remote Pilot Certificate #4111324 and Small UAS Registration #FA3MXM77NH and approved by Iowa State University. All field work was conducted under permits from, respectively, CU and KSU, and flights at Konza Prairie were conducted under KSU Authorization Certificate #00035. This project was funded by SERDP grant #RC18-C2-1021.

Abstract

Introduction & Objectives

This research examines how spatially mediated uncertainty impacts the predictive value of ecological models we use to assess habitat and species vulnerability. A major challenge of relying on models to identify ecological consequences is spatial mismatch – the area represented by a grid square is larger than the scale at which microclimate conditions vary and most organisms experience their environments. As a result, models capture little of the fine-scale environmental variation that drives critical physiological and ecological processes. However, our understanding of how model spatial resolution affects predictive models is limited, impairing our ability to make evidence-based management decisions.

The broad scientific objective of this research is to understand how model resolution drives our ability to detect variation in microclimate conditions over space and time and how that variation drives organisms' vulnerability to changing conditions. The technical objective of this project was proof-of-concept for a novel application of UAS: examining how the spatial resolution of a model impacts our capacity to assess microclimate-scale heterogeneity at ultra-fine resolutions. The driving hypothesis of this research is that sub-meter scale variation in microclimate conditions can impact model predictions, effects that may be moderated by terrain complexity. If this hypothesis is true and testable using current modeling approaches, this project allows us to (i) quantify that variation and (ii) identify the spatial resolution(s) at which sampling *should* be targeted or whether ultra-fine scale variation can be modeled at all.

Technical Approach

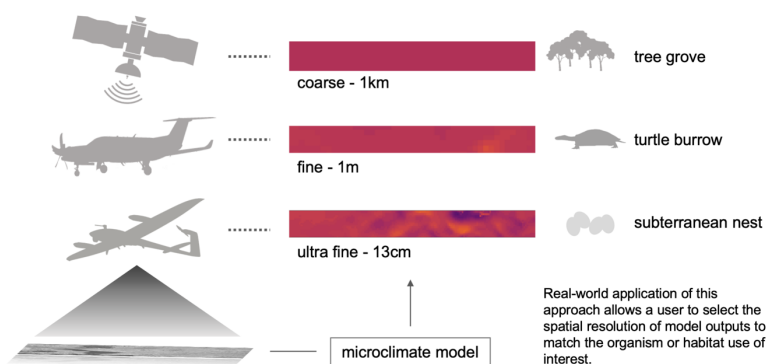
We used emerging methods in UAS remote-sensing to quantify the effects of spatial resolution on our ability to detect and predict the microclimate-scale effects of broad-scale environmental variation. UAS-collected terrain data were used to drive a process-based model of hourly substrate temperatures over one year. The values and spatial distributions of model predictions at spatial resolutions of 1m and 13cm were compared to determine whether cm-scale variation in microclimate conditions could be detected.

Results

Substrate temperatures generally did not vary by spatial resolution but were spatially differentiated by resolution; the finer-resolution model showed finer-scale heterogeneity in microclimate conditions. The implications of our results are that, at least at fine and ultra-fine spatial resolutions, model accuracy is independent of resolution. Even without direct validation, we can be reasonably confident in models' relative accuracy. Likewise, increasing the spatial resolution of a model above 1m is unlikely to benefit research without an explicit, ultra-fine spatial component. However, our findings confirm the assumption that spatial mismatch can mask substantial heterogeneity in microclimate conditions, without which ecological assumptions and management decisions may be incorrect. Finally, our findings illustrate that microclimate heterogeneity is detectable at ultra-fine, cm-scales without direct measurement.

Benefits

Continuation of this research is critical for validating our findings at the continental scale, with the goal of linking spatial and temporal resolution to model uncertainty within a reliable statistical framework. By addressing a key issue in using gridded data in ecological modeling, our results will provide DoD resource managers with insight into the drivers of microclimate landscapes. Our modeling approach is spatially and temporally transferable. It has the potential to increase the precision of management decisions and reduce the costs associated with conservation-focused data collection.



Executive Summary

Introduction

Ecological niche models are ecologists' multi-tools for understanding and predicting the effects of global climate change and habitat modification on biodiversity. To predict range shifts, local extinctions, life-history shifts, and other potential ecological impacts of environmental change, ENMs rely on climate layers that summarize important environmental variables over a grid of pixels that are spatially linked with either (i) data on a species' current or historic distribution, which act as a proxy for its fundamental physiological limitations (Guisan & Zimmerman 2000), or (ii) data that explicitly describe how a species' physiology constrains its distribution (Kearney & Porter 2004, Porter et al 2010). Continental and global-scale gridded layers are widely available and freely downloadable (Hijmans et al 2005, Kearney et al 2014, Wang et al 2016). However, a major barrier to building ecologically realistic ENMs is mismatch between the spatial resolution of those layers, with each grid square (or pixel) representing between 1-225 km² or more, and the relatively small size of the area *actually* used by most species (Hutchinson & MacArthur 1959, Porter et al. 2010, Faye et al 2016, Kotamarthi et al 2016). In addition, the spatial resolution of underlying layers can significantly impact the direction and strength of modeled relationships between ecological measurements and climate variables (Guisan et al 2007, Sears et al 2011, Leempoel et al 2015). Our ability to assess the overall vulnerability of any particular habitat to climate change can also be greatly impaired if the pixel size of an ENM is similar to or larger than the full spatial extent of the habitat, or habitat patches, being studied (Carter et al 2015). The area of DoD-managed sites varies from less than 10km² to more than 300km². Ultimately, if a population is either (1) bounded within one or only a few 1km² and/or (2) sparsely distributed throughout any number of pixels, then very little of that population's environment, which may vary on a scale of a few meters or less, can be captured by a 1km² model, even though 1km² environmental layers are, on a global scale, extremely fine-resolution (Potter et al 2013, Faye et al 2016, Kotamarthi et al 2016, Meineri & Hylander 2016).

While the term 'microclimate' was originally defined primarily to differentiate near-ground environmental conditions from those in higher atmospheric layers (Geiger et al 2009), both 'microclimate' and 'microenvironment' have consistently been used in the literature in reference to relatively small-scale sites that may provide unique microclimate conditions and result in climatic 'pockets' that are environmentally distinct from surrounding areas (Wilson 1998; Korb 2003, Angiletta et al 2009; Briscoe et al 2014; Hannah et al 2014). As an environmentally distinct space, a microclimate can provide a micro-refuge, either from inhospitable conditions in the surrounding microclimate layer or from unfavorable macroclimate conditions like extreme temperatures or drought (Geiger et al 2009, Rull 2009, Mosblech et al 2011, Hannah et al 2014). For example, koala (*Phascolarctos cinereus*) in Australia use species-specific tree limbs with surface temperatures that are cool, relative to the surrounding environment, to facilitate conductive thermoregulation during extreme heat events (Briscoe et al 2014). Use of 'microclimate/microenvironment' in reference to location-specific environmental conditions implies that the climate space of interest is experienced by an organism at a relatively small scale, while the use of 'microclimate' to refer to the atmospheric microclimate layer is not strictly spatial scale-dependent. Gridded microclimate layers can be used to describe the microclimate conditions underlying species' distributions on a global scale or within a single, small microclimate. Whether a microclimate surface can be reasonably labeled as 'high/fine resolution' is relative to the spatial extent of an area of interest. A horizontal spatial resolution of 30" latitude, approximately 1km, can be considered very high resolution for a global or continental-scale model (Wang et al 2016) but would be low-resolution if the area being modeled is a 500m² patch of habitat. Similarly, a 1 km² gridded model is high-resolution, relative to the home range of a large mammal. However, the same model would capture little environmental heterogeneity on a scale relevant for a small, soil-dwelling invertebrate. That is, the same model can be ecologically relevant for some applications but uninformative for others.

The coarser the spatial resolution of a predictive surface, the more within-pixel variation is masked and, as a result, model uncertainty increased (Carter et al. 2015). From a conservation or resource-management

perspective, ENMs built with *relatively* coarse-resolution environmental layers (i.e. those that display spatial mismatch) may be unable to detect (i) microrefugia, the patches of suitable habitat in which species can persist during periods of otherwise unsuitable conditions (Rull 2009, Lenoir et al 2016) and (ii) microclimatic range edges, the dynamic boundaries between suitable and unsuitable patches of habitat that indicate the potential for range expansions/contractions or act as barriers to dispersal, indicating the potential for local extinctions (Wiens 2016). Importantly, the coarser its spatial resolution, the more likely that a spatially-explicit model would over- or underestimate the availability of suitable microhabitats for a target species (Loiselle et al 2003, Hannah et al 2014, Stubbs et al 2014, Faye et al 2016). As a result, management actions based on lower-resolution ENMs, that are ecologically informative for fewer taxa, may have an increased risk of failure (Nadeau et al 2017). Despite recognition of these limitations (Potter et al 2013, Hannah et al 2014, Carter et al 2015, Maréchaux et al 2016), we understand very little about the impacts of spatially-mediated uncertainty on our ability to model organism-microclimate interactions (Potter et al 2013, Carter et al 2015).

For some applications, predicting bounded possibilities in potential microclimates, such as detecting rough distributional limits based on the range of environments that are available to an organism at a variety of depths below ground, may be highly informative. However, answering questions about responses to climate extremes (Kearney et al 2012) or quantifying habitat configuration explicitly (Sears et al 2011) requires (i) reducing spatial mismatch by increasing the spatial resolution of the gridded layers used to build an ENM or (ii) accounting for spatially-mediated uncertainty when interpreting a spatially-mismatched ENM. Very fine-resolution (e.g., 1-5m²) microclimate models and ENMs *have* been built to assess vulnerability to climate change for specific, relatively small habitats (Mitchell et al 2010, Carter et al 2015). However, the methods used to model microclimate conditions at such fine spatial resolutions are time-consuming and, currently, computationally prohibitive for use at sites larger than about 50km² (Kearney & Porter 2009, Kearney et al 2014, Palmer 2014). Quantitatively accounting for spatial mismatch in lower-resolution ENMs may be able to bridge the gap between the practical limitations of ecological modeling and the requirements of assessing habitat vulnerability at ecologically relevant scales under rapidly changing climatic conditions. Here, we used a novel application of UAS-facilitated remote sensing to ecological modeling to reconcile the spatial resolution of an ENM with its ability to detect the fine-scale heterogeneity that drives habitat vulnerability. In this proof-of-concept study, we compared models built with UAS-collected terrain data to quantify how the resolution of underlying layers affects model predictions and – more critically – whether sub-meter variation in microclimate conditions can be modeled with existing computational approaches.

Objectives

The FY2018 Statement of Need (SON) *RC-18-C2: Climate Change Vulnerability Assessment of Major Habitats On and Around DOD Lands* recognizes that effective resource management requires understanding the potential (i.e., future) effects of climate change on the ecological processes underlying natural communities. Our project provides proof-of-concept for filling a critical gap in current methods of predictive modeling and sets the stage for continuing research to respond to all key objectives listed in the SON: (1) assessing habitat vulnerability to climate change and identifying factors that drive vulnerability; (2) improving understanding of the spatial variability in drivers of vulnerability across a species' range; (3) improving understanding of the relationship between climate change and key ecological processes; and (4) developing methodologies that translate research into practical information for guiding conservation and management of sensitive habitats. Our research focuses on how spatially mediated uncertainty impacts the predictive value of ecological models we use to assess microclimate vulnerability.

Because climate change is likely to create novel environments, our capacity for responding effectively to the impacts of climate change in managed habitats will rely increasingly on gridded, i.e., spatially explicit, predictive ecological models. A major challenge of relying on these models is spatial mismatch. That is, the area represented by each grid square is typically much larger than the scale at which (i)

microclimate conditions vary and (ii) most organisms experience and interact with their environment. As a result, model predictions capture little of the micro-environmental variation that drives critical physiological and ecological processes. However, our understanding of how model spatial resolution affects predictive models is limited, impairing our ability to make evidence-based management decisions.

The broad scientific objective of this research is to understand how the spatial scale at which models are built – and their corresponding resolution – drives our ability to detect variation in microclimate conditions over space and time *and* how that variation drives organisms' vulnerability to changing conditions. More specifically, we aim to understand how spatial resolution, the resolution of climate data, and microclimate-scale characteristics themselves interact to affect model uncertainty and, in turn, the strength and direction of model predictions. To meet this goal, the technical objective of this project is to provide proof-of-concept for a novel application of UAVs: examining how the spatial resolution of a predictive model impacts our capacity to assess microclimate-scale heterogeneity.

The driving hypothesis of this research is that sub-meter scale variation in microclimate conditions can impact model predictions, effects that may be moderated by terrain complexity. If this hypothesis is true – and testable using current modeling approaches – this project should allow us to (i) quantify that variation and (ii) identify the spatial resolution(s) at which sampling *should* be targeted or, at a minimum, whether ultra-fine scale variation can be modeled at all.

Technical Approach

We leveraged emerging methods in UAS remote-sensing to quantify the effects of spatial resolution on our ability to detect and predict the microclimate-scale effects of broad-scale environmental variation. Although well-established in the field of precision agriculture, UAS have only recently been used for ecology and conservation, including for collection of microclimate data (Anderson & Gaston 2013, He et al 2015, Pimm et al 2015). Notably, Faye et al (2016) used UAS-collected, thermal-infrared imagery to find significant spatial-temporal heterogeneity in surface thermal conditions at spatial resolutions of 1-25cm². Thus, we know microclimate conditions can vary on the cm-scale. However, the use of UAS-collected data for predictive ecological modeling, rather than solely for direct measurement, had not been explored. The key aim of this study was to provide proof-of-concept of UAS as a tool for detecting heterogeneity in 3D microclimate landscapes, with the goal of eventually understanding how uncertainty in ecological models affects conservation and management decisions.

We built and tested a series of microclimate models with NicheMapR, a global-scale, open-source model that uses a one-dimensional, finite-difference algorithm to solve heat-mass balance equations for microclimate conditions, including substrate temperatures at ten depths, at any specified location and time (Kearney & Porter 2016). The input data for the microclimate model include: geospatial parameters (longitude, latitude, elevation, slope, aspect, viewshed) for calculating clear-sky solar radiation using the integrated SOLRAD radiation subroutine (McCullough & Porter 1971; Porter et al 2002); daily maximum and minimum values of seasonally dynamic climate variables (wind speed, air temperature, humidity, rainfall, cloud cover); and values for relatively constant, site-specific environmental variables (physical properties of soils, shade availability). Optionally, values of soil properties can be modified to simulate a layer of organic topsoil, or 'organic cap,' which reduces below-surface conduction of thermal energy for sites with high turnover rates. NicheMapR also incorporates scattering of solar radiation due to atmospheric aerosols using a modified version of the Global Aerosol Data Set (GADS) (Koepke et al 1997; Kearney & Porter 2016). By modifying the 'global' functions of NicheMapR so the spatial resolution of model outputs is determined by that of the terrain inputs (*sensu* Carter et al 2015), we can explicitly test the effects of spatial resolution, while controlling for the effects of all other input parameters.

Model outputs include one year of hourly, above and below-ground microclimate conditions for 'shaded' and 'unshaded' scenarios represented by 0% and 90% available canopy shading (Kearney et al 2014, Kearney & Porter 2016). Outputs for below-ground conditions include temperature, moisture, water

potential, and relative humidity for ten, user-specified depths (**Table 1**). For this study, we modeled conditions for soil depths of 0, 2.5, 5, 10, 15, 20, 30, 50, 100, and 200cm.

To drive the model, we collected aerial imagery from two sites within the conterminous U.S. that have distinct, complex terrain profiles: the Rocky Mountain Biological Station (NIWO) in central Colorado and the Konza Prairie Long-Term Ecological Research site (KONZ) in northeast Kansas. The Iowa State University Horticultural Research Station in central Iowa acted as a flat, protocol-testing site. The two study sites were selected from broad latitudinal bands using semivariogram analysis of terrain roughness (Trevisani & Rocca 2015); representing a range of topographic complexity within and between field sites will be directly informative for transitioning into further research. In addition, NIWO and KONZ contain plots maintained by the National Ecological Observatory Network (NEON), a continent-scale, long-term research effort that will be a key partner in expanding this research beyond the pilot stage.

We designed our image acquisition plan using eMotion v 3.7.5^a software on a Windows Surface Pro 6 tablet^b (Windows 10 x64, Intel® Core™ i7-8650U CPU @ 1.90GHz 2.11GHz, 16GB RAM) to capture aerial RGB imagery using an eBeeX^a fixed-wing drone (Fig 1a) and integrated AeriaX^a (F/2.8-16 18.5mm / 28mm equivalent: 35 mm) photogrammetry camera. The same software was used for flight control. To achieve a goal GSD of 2.5cm, image acquisition was planned for a flight altitude of 118.8m AED. The recommended minimum image overlap for a terrestrial, horizontal mapping project with heterogeneous visual content is 75% lateral and 60% longitudinal overlap^c. Since optimal flight patterns are influenced by wind direction, we chose a conservative overlap of 80% lateral and 75% longitudinal and used interlaced flight lines at 30m spacing to decrease total flight time and reserve battery life. Prior to undertaking test flights at the Iowa State University Horticultural Research Station, we ran in-software simulations to confirm all parameters. We conducted single-block test flights at the ISU Horticultural Research Station [42.11°N 93.59°W] to ensure the image acquisition plan was feasible, paying particular attention to battery life and the specification of landing vectors. The land cover at this research station is comprised largely of agricultural crops, so the visual content is relatively homogenous. This allowed us to confirm that our planned image overlap would be more than sufficient to build 3D models for our main study areas. Based on our test flights, we also elected to use the eBeeX' standalone PPK capability, rather than setting GCPs and using RTK positioning, which would result in substantial savings of person-hours at the main study locations, where the terrain was much more variable and difficult to access, with negligible impact on absolute model accuracy.

Based on the spatial extent of the sampling areas at the main NEON sampling sites, we divided the total flight area (1km²) into 8 overlapping blocks of 0.25km² each, with 440 images collected over a flight distance of 13,782m and flight time of 00:19:22 for each block. Thus, each image covered an area of 150x100m, with 25m between adjacent images. Aerial imagery was collected at KONZ from 23-27 May 2019 and at NIWO from 23 June – 2 July 2019 by the PI and one field assistant to act as visual observer to the UAS flights. Following acquisition, images and flight logs were uploaded into the control software. Image geotags were post-processed using RINEX v 2.11 data downloaded from the nearest active station in the CORS network. For KONZ imagery, we used correction data from station KSU1 [37.65°N 96.57°W], which samples GPS data at a 15sec sampling rate. For NIWO, we used correction data from station P041 [43.42°N 100.82°W], which samples GPS + GNSS data at 1sec. Following post-processing, absolute accuracy of the image geotags increased from 2-3m to 3-4cm.

^a senseFly SA, Cheseaux-Lausanne, Switzerland 2018; www.sensefly.com

^b Microsoft Corporation, Redmond, WA 2019

^c Pix4Dmapper 4.1 User Manual, Pix4D SA, Lausanne, Switzerland 2018; www.pix4d.com

Table 1. Summary of predicted above-ground conditions output by the NicheMapR microclimate model. For this project, each variable will be output hourly, over one year, for each modeled spatial resolution.

variable #	variable	description	units
1	JULDAY	Julian day over one year	d
2	TIME	time of day for 24 hours	min
3	TALOC	air temperature at local height	°C
4	TAREF	air temperature at reference height	°C
5	RHLOC	relative humidity at local height	%
6	RH	relative humidity at reference height	%
7	VLOC	wind speed at local height	m s ⁻¹
8	VREF	wind speed at reference height	m s ⁻¹
9	SNOWMELT	amount of snowmelt	mm
10	POOLDEP	water pooling on surface	mm
11	PCTWET	soil surface wetness	%
12	ZEN	zenith angle of sun	deg
13	SOLR	solar radiation	W m ⁻²
14	TSKYC	sky radiant temperature	°C
15	DEW	dew presence/absence	1/0
16	FROST	frost presence/absence	1/0
17	SNOWFALL	amount of snow fallen	mm
18	SNOWDEP	snow depth	cm
19	SNOWDENS	snow density	g cm ⁻³
20-29	SOIL	soil temperature (at 10 depths)	°C
30-39	SHADSOIL	soil temperature, shaded (at 10 depths)	°C
40-49	SOILMOIST	soil moisture (at 10 depths)	m ³ m ⁻³
50-59	SHADMOIST	soil moisture, shaded (at 10 depths)	m ³ m ⁻³
60-69	SOILPOT	soil water potential (at 10 depths)	J kg ⁻¹
70-79	SHADPOT	soil water potential, shaded (at 10 depths)	J kg ⁻¹
80-89	HUMID	soil relative humidity (at 10 depths)	%
90-99	SHADHUMID	soil relative humidity, shaded (at 10 depths)	%
100	TRANS	plant transpiration rate	kg m ⁻² s ⁻¹
101	LEAFPOT	leaf water potential	J kg ⁻¹
102-111	RPOT	root water potential (at 10 depths)	J kg ⁻¹
112-119	SN1	snow temperature (of 8 potential layers)	°C
120-230	DRLAM	wavelength-specific direct solar radiation	W m ⁻² nm
231-340	DRRLAM	wavelength-specific direct Rayleigh solar radiation	W m ⁻² nm
341-440	SRLAM	wavelength-specific scattered solar radiation	W m ⁻² nm

Images were processed in Pix4Dmapper Pro v 4.1 software^a running on a Dell Precision 3630 desktop computer^d (Windows 10 x64 Intel® Xeon® E-2186G CPU @ 3.80GHz 3.79GHz 64GB RAM w/ NVIDIA® Quadro® P5000 16GB video card) using the ‘3D model’ template. The conservative overlap in our image acquisition plan meant that initial processing generated enough ATPs (i.e., matched points among multiple images) to generate a dense point cloud without manually adding more points. Point cloud densification was done with a matching window size of 7x7 pixels. For later comparison, we processed images both with and without PPK corrections to generate an orthomosaic, densified point cloud (Fig 9), DSM using inverse-distance weighting (Fig 10), and DTM, which was down-sampled to 5x the spatial resolution of the DSM, giving us elevation data at a horizontal spatial resolution of ~13cm (Fig 10) from a DSM of ~2.6cm per pixel. Because the RGB sensor cannot ‘see’ through the vegetation canopy (i.e., in contrast to LiDAR), we used a gradient-boosted tree algorithm to classify points based on geometric and color features⁵, then filter out points classified as ‘high vegetation’ prior to generating the DTM (Fig 9). We also filtered out human-made objects (e.g., metal plot markers, weather station, etc.). Finally, we clipped the DTM to match the extent of the 1km² NEON grid.

From the filtered DTM (Fig 10), we derived slope and aspect models (Fig 11, 12) using either the ‘raster’^e R-package or in ArcGIS Desktop^f v 10.5.1 software, using the Horn algorithm, which is the same one implemented by NEON to produce derivative geospatial layers (Horn 1981). Thus, the difference between outputs is negligible (Goulden et al. 2016). We then used the horizonSearch() function in R-package ‘horizon’^g to calculate horizon elevation angles (i.e., viewshed) from the DTM (Fig 13) for each hour across 24 hours.

We then used the terrain variables to drive the microclimate model included with the NicheMapR¹ package for biophysical modeling, run in R v 3.5.0^h, generating 3D microclimate conditions within each 1km² grid. To generate hourly microclimate predictions that would fulfill project aims, we modified the default micro_usa() function to be more flexible in its acceptance of inputs from different sources. Terrain inputs were provided as 27-layer raster stacks (i.e., elevation, slope, aspect, and 24x horizon elevation angles) in netcdf (.nc) format at spatial resolutions of 1m and 13cm. Climate inputs were provided via a link to the gridMET¹² database of daily climate data, interpolated to a spatial resolution of ~4km. In addition to customizing the microclimate model itself, we wrote a looping function to run the microclimate model across a 2D grid of any dimensions (see Appendix C for the raster-stacking and looping functions). More details about model parameterization, including justification for using default generalized soil properties, are in Carter et al (2015, 2018)^{3,4}. Fine and ultra-fine resolution microclimate models (1m and 13cm, respectively) were run on the Condor high-performance computing facility at Iowa State University. Because we were unable to parallelize the model, the total spatial area we could run is < 1km². For the 13cm model, we ran a ‘snapshot’ of the larger area.

We also deployed 49 and 34, respectively (out of 50 planned), wireless data loggers^j (HOBO® MX2303; accuracy +/- 0.2°C) on a 50x50m grid at KONZ and NIWO that systematically avoided NEON sampling plots and areas containing substantial vegetation other than grasses or low scrub. Probes were installed at 10cm and 20cm depths to collect soil temperature data at 1hr intervals and calculate statistics at 1min intervals. Holes for the data-logger probes were drilled using a 1.3cm-diameter, steel soil probe. Data loggers were mounted onto plastic landscaping stakes and marked with neon flagging. We located the

^d Dell Incorporated, Round Rock, TX 2019; www.dell.com

^e Hijmans, R. raster: geographic data analysis and modeling. R-package v 2.9-23. (2019)

^f Esri®, Redlands, CA 2017; www.esri.com

^g Van doninck, J. horizon: horizon search algorithm. R-package v 1.2. (2018)

^h The R Foundation for Statistical Computing 2018

ⁱ <http://www.climatologylab.org/gridmet.html>

^j Onset® Computer Corporation, Bourne, MA 2018; www.onset.com

points for data-logger deployment using a handheld GPS receiver^k (Oregon® 750t; horizontal accuracy +/- 3m). All data were collected in the WGS84 coordinate system.

Results and Discussion

In general, the values of predictions did not vary by spatial resolution (**Figure 1**). The implication of this result is that, at least at these spatial resolutions, model accuracy is independent of resolution. That is, even without direct validation, we can be confident in models' relative accuracy compared with measured values. Thus, increasing the spatial resolution of a model above 1m is unlikely to benefit research without an ultra-fine spatial component.

In contrast, the spatial component of substrate temperatures was differentiated between 1m and 13cm (**Figure 2**). That is, the finer-resolution model showed finer-scale heterogeneity in microclimate conditions. Prior to this study, we could not have expected this result a priori. Indeed, based on the preponderance of ecological studies in which the resolution of spatial data is mismatched to the target organism(s), the expectation that microclimate heterogeneity would be undetectable at such fine resolutions would be reasonable. Research in spatial ecology has proceeded with the implicit assumption that, although spatial mismatch may be an important but unknown contributor to observed variation in natural populations, it is undetectable. Otherwise, much previous effort would have been put into detecting it. Our findings illustrate that microclimate heterogeneity is detectable at ultra-fine, cm-scales.

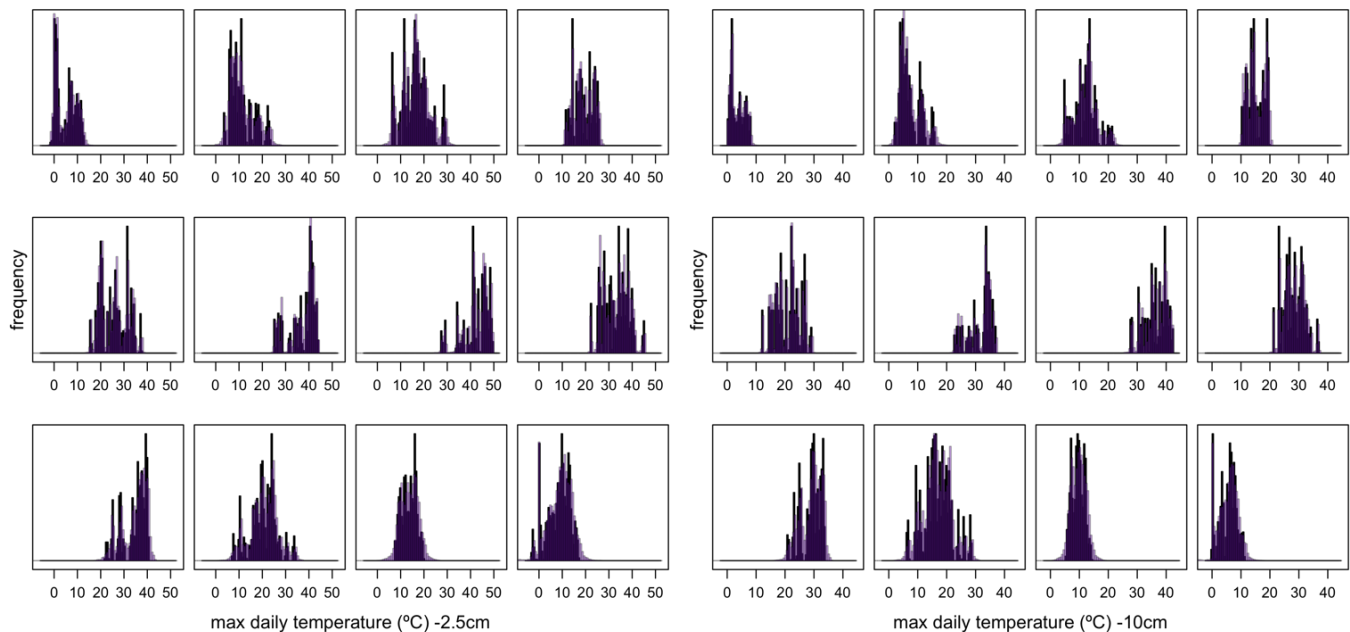


Figure 1. Overlapping frequency histograms of maximum daily predicted substrate temperatures over 12 months [Jan (top left) – Dec (bottom right), 2017] at (a) 2.5cm and (b) 10cm depths. The black bars are predictions made using terrain inputs at a 1m spatial resolution, and the overlaid transparent purple bars show predictions made at 13cm.

^k Garmin Ltd., Schaffhausen, Switzerland 2018; www.garmin.com

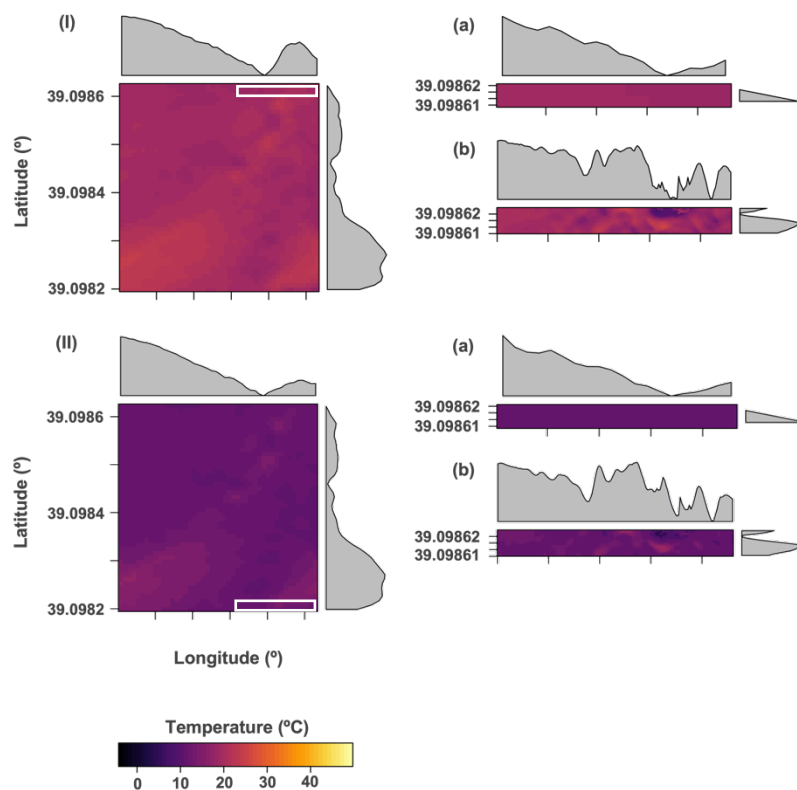


Figure 2. Snapshot of microclimate model output, summarized from hourly data for a representative day (Julian Day 45: 14 Feb 2017). Data are maximum daily soil temperature at depths of (I) 2.5cm and (II) 10cm. The square plot shows data for a 50x50m area at a spatial resolution of 1m, with the white inset magnified at spatial resolutions of (a) 1m and (b) 13cm. The gray marginal plots show means across space. Note the 'cool spot' in Ib that is not apparent in the 1m plot, an example of variation in substrate temperatures detected only in the finer-resolution model. See additional plots in Appendix A.

Implications for Future Research and Benefits

Our primary hypothesis was supported by the results. Substantial spatial differentiation between model outputs at 1m vs. 13cm indicates that spatial resolution can affect the detection of biologically important variation in microclimate conditions, likely without significantly impacting model accuracy. An important implication of this result is that spatial mismatch could be leading to erroneous ecological assumptions and incorrect management decisions (i.e., the suspicion driving this research). Given this conclusion, conservation and management decisions made using relatively coarse resolution models – that is, the vast majority of ecological niche models – should be considered tentative, at best. Preferably, the indications of a coarse model would be verified with spatially matched data. In addition, the continuation of this research is critical for validating our findings at the continental scale, with the goal of linking spatial and temporal resolution to model uncertainty within a reliable statistical framework. By addressing a key issue in using gridded data in ecological modeling, our results will provide DoD resource managers with insight into the spatially-mediated drivers of microclimate landscapes. Our modeling approach emphasizes the potential for spatial and temporal transferability of results, a framework that can improve targeting of management efforts and reduce costs associated with conservation-focused data collection.

Objective

The FY2018 Statement of Need (SON) *RC-18-C2: Climate Change Vulnerability Assessment of Major Habitats On and Around DOD Lands* recognizes that effective resource management requires understanding the potential (i.e., future) effects of climate change on the ecological processes underlying natural communities. Our project provides proof-of-concept for filling a critical gap in current methods of predictive modeling and sets the stage for continuing research to respond to all key objectives listed in the SON: (1) assessing habitat vulnerability to climate change and identifying factors that drive vulnerability; (2) improving understanding of the spatial variability in drivers of vulnerability across a species' range; (3) improving understanding of the relationship between climate change and key ecological processes; and (4) developing methodologies that translate research into practical information for guiding conservation and management of sensitive habitats. Our research focuses on how spatially mediated uncertainty impacts the predictive value of ecological models we use to assess microclimate vulnerability.

Because climate change is likely to create novel environments, our capacity for responding effectively to the impacts of climate change in managed habitats will rely increasingly on gridded, i.e., spatially explicit, predictive ecological models. A major challenge of relying on these models is spatial mismatch: the area represented by each grid square is typically much larger than both the scale at which (i) microclimate conditions vary and (ii) most organisms experience and interact with their environments. As a result, model predictions capture little of the micro-environmental variation that drives critical ecological processes. However, our understanding of how model spatial resolution affects predictive models is limited, impairing our ability to make evidence-based management decisions.

The broad scientific objective of this research is to understand how the spatial scale at which models are built – and their corresponding resolution – drives our ability to detect variation in microclimate conditions over space and time *and* how that variation drives organisms' vulnerability to changing conditions. More specifically, we aim to understand how spatial resolution, the resolution of climate data, and microclimate-scale characteristics themselves interact to affect model uncertainty and, in turn, the strength and direction of model predictions. To meet this goal, the technical objective of this project is to provide proof-of-concept for a novel application of UAVs: examining how the spatial resolution of a predictive model impacts our capacity to assess microclimate-scale heterogeneity. The driving hypothesis of this research is that sub-meter scale variation in microclimate conditions can substantially impact model predictions, effects that may be moderated by terrain complexity. If this hypothesis is true – and testable using current modeling approaches – this project should allow us to (i) quantify that variation and (ii) identify the spatial resolution(s) at which sampling *should* be targeted or, at a minimum, whether ultra-fine scale variation can be modeled at all.

This report is associated with a go/no-go decision point following modeling of microclimate conditions. A 'go' decision to pursue further modeling would be justified by identification of significant, detectable variation in model outputs between at least two models built at different spatial resolutions within at least one grid for at least one of our main study sites. In addition, we have described modifications to our original protocols that were required to comply with restrictions put in place during the COVID-19 pandemic, (i) challenges identified and lessons learned, and (ii) next steps for expanding the scope and scale of this research, with the goal of achieving its broad scientific objective.

Background

To predict range shifts, local extinctions, life-history shifts, and other potential ecological impacts of environmental change, ecological niche models (ENMs) rely on climate layers that summarize important environmental variables over a grid of pixels spatially linked with (i) data on a species' current or historic distribution, which act as a proxy for its fundamental physiological limitations (Guisan & Zimmerman 2000), or (ii) data that explicitly describe how a species' physiology constrains its distribution (Kearney & Porter 2004, Porter et al 2010). Continental and global-scale gridded layers are widely available and freely downloadable (Hijmans et al 2005, Kearney et al 2014, Wang et al 2016). However, a major barrier to building ecologically realistic niche models is mismatch between the spatial resolution of those layers and the relatively small size of the area *actually* used by most species (Hutchinson & MacArthur 1959, Porter et al. 2010, Faye et al 2016, Kotamarthi et al 2016). In addition, the spatial resolution of underlying layers can significantly impact the direction and strength of modeled relationships between ecological measurements and climate variables (Guisan et al 2007, Sears et al 2011, Leempoel et al 2015). Our ability to assess the overall vulnerability of a habitat to climate change can also be greatly impaired if the pixel size of an ENM is similar to the spatial extent of the entire habitat (Carter et al 2015). The area of DoD-managed land varies from $<10 \text{ km}^2$ to $>300 \text{ km}^2$. If a population is either bounded within one, or only a few, 1 km^2 pixels or sparsely distributed throughout any number of pixels, then very little of that population's environment can be captured by a 1 km^2 model, even though 1 km^2 environmental layers are extremely fine resolution from a global perspective (Potter et al 2013, Faye et al 2016, Kotamarthi et al 2016, Meineri & Hylander 2016).

'Microclimate' was originally defined primarily to differentiate near-ground environmental conditions from those in higher atmospheric layers (Geiger et al 2009). However, both 'microclimate' and 'microenvironment' have been used in the contemporary literature to reference small-scale sites that provide unique microclimate conditions – climatic 'pockets' that are environmentally distinct from surrounding areas (Wilson 1998; Korb 2003, Angiletta et al 2009; Briscoe et al 2014; Hannah et al 2014). A microclimate can also be a micro-refuge from inhospitable conditions in the surrounding microclimate layer or unfavorable macroclimate conditions like extreme temperatures or drought (Geiger et al 2009, Rull 2009, Mosblech et al 2011, Hannah et al 2014). Furthermore, gridded microclimate layers can be used to describe the microclimate conditions underlying species' distributions on a global scale or within a single, small microclimate. However, whether a microclimate surface is fine resolution is relative. A horizontal spatial resolution of 30" latitude, approximately 1km, can be considered very high resolution for a global or continental-scale model (Wang et al 2016) but would be low-resolution if the area being modeled is a 500 m^2 patch of habitat. Similarly, a 1 km^2 gridded model is high-resolution relative to the home range of a large mammal. However, the same model would capture little environmental heterogeneity on a scale relevant for a small, soil-dwelling invertebrate. That is, the same model can be ecologically relevant for some applications but not for others.

The coarser the spatial resolution of a predictive surface, the more within-pixel variation is masked and, as a result, model uncertainty increased (Carter et al. 2015). From a conservation or resource-management perspective, ENMs built with *relatively* coarse-resolution environmental layers (i.e. those that display spatial mismatch) may be unable to detect (i) microrefugia, the patches of suitable habitat in which species can persist during periods of otherwise unsuitable conditions (Rull 2009, Lenoir et al 2016) and (ii) microclimatic range edges, the dynamic boundaries between suitable and unsuitable patches of habitat that indicate the potential for range expansions/contractions or act as barriers to dispersal, indicating the potential for local extinctions (Wiens 2016). Importantly, the coarser its spatial resolution, the more likely that a spatially-explicit model would over or underestimate the availability of suitable microhabitats for a target species (Loiselle et al 2003, Hannah et al 2014, Stubbs et al 2014, Faye et al 2016). As a result, management actions based on lower-resolution ENMs and that are ecologically informative for fewer taxa, may have an increased risk of failure (Nadeau et al 2017). Despite recognition of these limitations (Potter et al 2013, Hannah et al 2014, Carter et al 2015, Maréchaux et al 2016), we understand very little

about the impacts of spatially-mediated uncertainty on our ability to model organism-microclimate interactions (Potter et al 2013, Carter et al 2015).

For some applications, predicting bounded possibilities in potential microclimates, such as detecting rough distributional limits based on the range of environments that are available to an organism at a variety of depths below ground, may be highly informative. However, answering questions about responses to climate extremes (Kearney et al 2012) or quantifying habitat configuration explicitly (Sears et al 2011) requires (i) reducing spatial mismatch by increasing the spatial resolution of the gridded layers used to build an ENM or (ii) accounting for spatially-mediated uncertainty when interpreting a spatially-mismatched ENM. Very fine-resolution (e.g., 1-5m²) microclimate models and ENMs *have* been built to assess vulnerability to climate change for specific, relatively small habitats (Mitchell et al 2010, Carter et al 2015). However, the methods used to model microclimate conditions at such fine spatial resolutions are time-consuming and, currently, computationally prohibitive for use at sites larger than about 50km² (Kearney & Porter 2009, Kearney et al 2014, Palmer 2014). Accounting for spatial mismatch in lower-resolution ENMs may bridge the gap between the practical limitations of ecological modeling and the requirements of assessing habitat vulnerability at ecologically relevant scales under rapidly changing climatic conditions. Here, we used a novel application of UAS-facilitated remote sensing to ecological modeling to reconcile the spatial resolution of an ENM with its ability to detect the fine-scale heterogeneity that drives habitat vulnerability. In this proof-of-concept study, we compared models built with UAS-collected terrain data to quantify how the resolution of underlying layers affects model predictions and – more critically – whether sub-meter variation in microclimate conditions can be modeled with existing computational approaches.

Materials and Methods

General approach

Our general technical approach was to build and compare a series of microclimate models using the NicheMapR package for biophysical modeling¹. NicheMapR is a global-scale, open-source model that uses a one-dimensional, finite-difference algorithm to solve heat-mass balance equations for microclimate conditions, including substrate temperatures at ten depths, at any specified location and time (Kearney & Porter 2016). The input data for the microclimate model include geospatial parameters (longitude, latitude, elevation, slope, aspect, viewshed) for calculating clear-sky solar radiation using the integrated SOLRAD radiation subroutine (McCullough & Porter 1971; Porter et al 2002); daily maximum and minimum values of seasonally dynamic climate variables (wind speed, air temperature, humidity, rainfall, cloud cover); and values for relatively constant, site-specific environmental variables (physical properties of soils, shade availability). Optionally, values of soil properties can be modified to simulate a layer of organic topsoil, or 'organic cap,' which reduces below-surface conduction of thermal energy for sites with high turnover rates. NicheMapR also incorporates scattering of solar radiation due to atmospheric aerosols using a modified version of the Global Aerosol Data Set (GADS) (Koepke et al 1997; Kearney & Porter 2016; **Figure 3**). By modifying the 'global' functions of NicheMapR so the spatial resolution of model outputs is determined by that of the terrain inputs (*sensu* Carter et al 2015), we can explicitly test the effects of spatial resolution, while controlling for the effects of all other input parameters. By testing the impacts of varying the spatial and temporal resolutions of the requisite inputs on the strength and direction of model predictions, we can determine maximum resolution(s) at which we can reasonably detect microclimatic heterogeneity and, thus, the resolution(s) at which sampling and modeling efforts can/should be focused. Ultra-fine spatial resolutions (i.e., cm-scale) are achieved using UAV-collected data. Model accuracy is assessed using hourly data collected with soil-temperature data loggers.

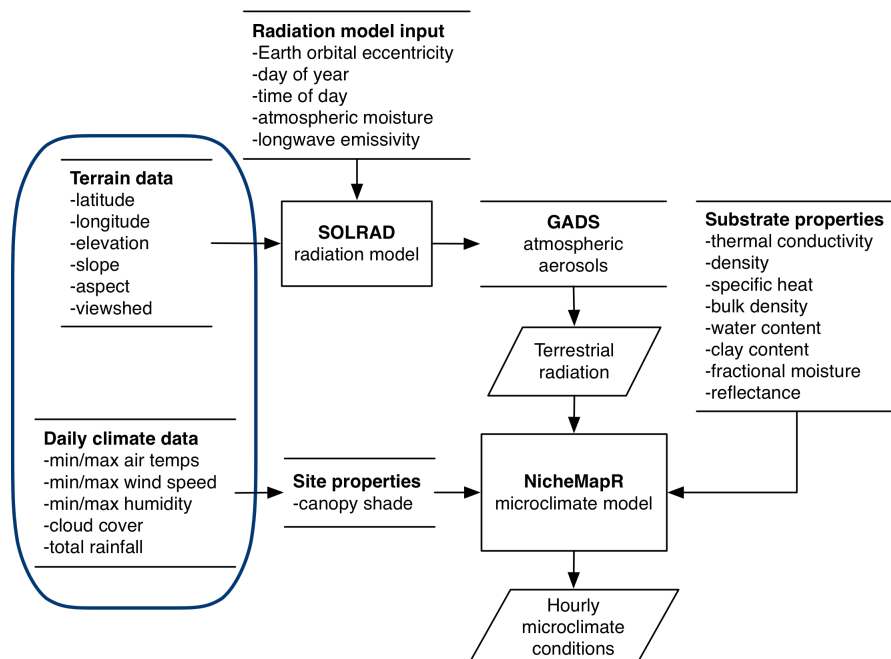


Figure 3. Flowchart showing the inputs required to run the NicheMapR microclimate model. The circled parameters are the focus of this project. The spatial resolution of output microclimate conditions is determined by the terrain inputs. The impacts of data availability (Task 1.4) will focus on the daily climate data.

microclimate model

One of the main constraints on predicting microclimate conditions at ultra-fine spatial resolutions, particularly when using a mechanistic model, is computational power. Per grid square, the NicheMapR microclimate model – itself an R wrapper for FORTRAN code (McCullough & Porter 1971) – requires approximately 30 seconds of runtime on a consumer laptop (Macbook Pro: 4-core, 2.7 GHz Intel Core i7 processor, 16 GB 2133 MHz LPDDR3 memory). At our ultra-fine spatial resolution, 1 km² contains roughly 1 million pixels and would require one year to finish processing at that rate. In addition, the model outputs ~12.4MB of data *per pixel* per year. A 1km² grid would output 12.4TB of data per year, which scales to at least 248 TB per year for 20 sites. Thus, scaling this project will have substantial data processing and storage needs, the latter comparable to institutional scale.

In the interests of optimizing the available computing resources and eventually scaling this project to 20 sites (one site per NEON domain), we pursued parallelization of the microclimate model across the Condo high-performance computing cluster. Although NicheMapR has been run on several different compute clusters, it has not previously been run in parallel (M. Kearney, pers comms). With advice from the ResearchIT department at ISU, our approach was to split a site into an integer number of same-sized jobs (i.e., ‘chunks’ of data containing the same number of pixels), appending the remainder into an extra job. The job numbers were then passed as a list of parameters, along with a call to the microclimate model, to `slurm_apply()` from the `rslurm` package¹.

Building microclimate surfaces

Models built at a spatial resolution of 1m² and finer were run on Condo^m, a high-performance computing server managed by ISU ResearchIT. Condo consists primarily of 192 SuperMicro servers, each with two 8-core Intel Haswell processors, 128 GB of memory, and 2.5 TB of available local disk. In addition to these compute nodes, there are three large memory nodes, two with four 8-core Intel Ivy Bridge processors and 1 TB of main memory. The third large memory node has four 10-core Ivy Bridge

¹ Philippe Marchand, Ian Carroll, Mike Smorul, Rachael Blake and Quentin Read (2021). `rslurm`: Submit R Calculations to a ‘Slurm’ Cluster. R package version 0.6.0. <https://CRAN.R-project.org/package=rslurm>

^m As of July 2021, the Condo cluster has been transitioned to free access.

processors and 2 TB of main memory. All nodes and storage are connected via Intel/Qlogic QDR InfiniBand (40 Mb/s) switch. There is also an accelerator node containing a pair of NVIDIA Tesla K20c GPU cards. The three service nodes include a login node, data transfer node, and management node.

The maximum spatial resolution of each model is equal to that of the terrain inputs. For this project, ‘fine-resolution’ models were driven by terrain layers derived from the NEON AOP LiDAR elevation data (DTM) at a horizontal spatial resolution of 1m, and ‘ultra-fine’ models were obtained using UAV-collected data. Availability of NEON climate data varied spatially and temporally. We elected to build the first set of models for 2017, because NEON data for that year are more complete than 2018 (i.e., fewer months are missing from the published dataset). However, for the initial model runs, we used NicheMapR. Ultimately, however, we would use NEON data for all 20 sites.

Testing data availability

Our original plan was to test the impacts of data availability on model outputs once modeling was completed. However, since none of the climate inputs to the microclimate model are optional (unlike, for example, regressions and other common statistical approaches), “data availability” can only be investigated in terms of resolution. For example, lowering the temporal resolution of the climate variables from hourly to daily to monthly, i.e., removing the normal fluctuating conditions experienced at a site, is one indirect approach to testing data availability. The spatial resolution of input data can be utilized similarly. This loss of resolution simulates a realistic lack of data, since – at a minimum – monthly climate normals are available at spatial resolutions of 10’ – 30” for the entire planetⁿ and finer for many regions of the U.S. We expect that – in line with findings from previous work – loss of resolution in climate data would reduce model accuracy, most noticeably during periods of relatively higher fluctuations in the flattened variable (Carter et al 2015).

ⁿ Fick, S.E. & Hijmans, R.J. WorldClim v 2.0: New 1km spatial resolution climate surfaces for global land areas. *International Journal of Climatology* (2016); www.worldclim.org

Model validation

Data collection

We deployed 49 and 34 (out of 50 planned) high-accuracy, wireless data loggers^o (HOBO® MX2303; accuracy $\pm 0.2^{\circ}\text{C}$) on a 50x50m grid (**Figure 4**) at, respectively, KONZ and NIWO that systematically avoided NEON sampling plots and areas containing substantial vegetation other than grasses or low scrub. Probes were installed at 10cm and 20cm depths to collect soil temperature data at 1hr intervals and calculate statistics at 1min intervals.

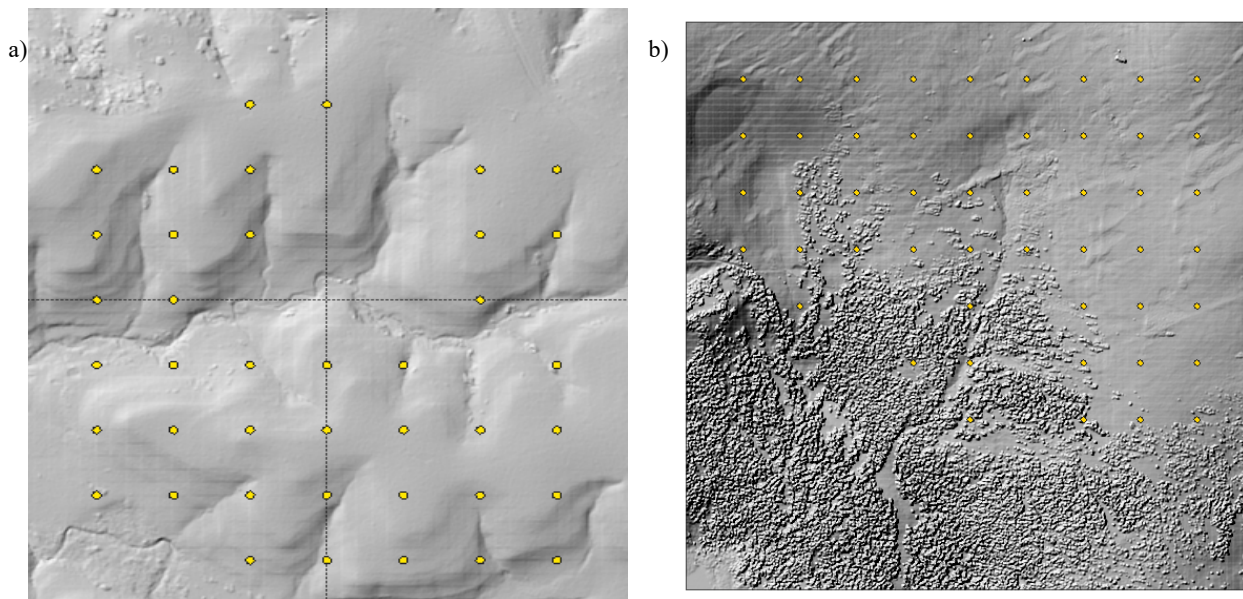


Figure 4. Image showing the 50x50m data logger grid ($N = 50$) designed for a) KONZ and b) NIWO, as yellow points superimposed over a shaded surface relief of the 1km² sampling grid (surface data from NEONE).

Holes for the data-logger probes were drilled using a 1.3cm-diameter, steel soil probe. Data loggers were mounted onto plastic landscaping stakes and marked with neon flagging. We located the points for data-logger deployment using a handheld GPS receiver^p (Oregon® 750t; horizontal accuracy $\pm 3\text{m}$). On retrieval, deployment points will be marked using an RTK-enabled GNSS receiver^q (Reach RS+; horizontal accuracy $\pm < 1\text{cm}$) and post-processed to maximize absolute accuracy, relative to the microclimate models. All data were collected in the WGS84 coordinate system.

Data loggers installed in summer 2019 were retrieved from the KONZ site in early spring 2020 (13-16 March), providing around nine months of soil temperature data for model validation. Data loggers were retrieved from NIWO in late summer (14-15 August). No data were retrieved from the NIWO data loggers, most of which showed substantial evidence of mammal interference with the cables.

Data processing

Hourly temperature data from KONZ were organized in R and extreme outliers (negative values indicating a software/hardware glitch at the time of deployment) were removed. Fine and ultra-fine resolution microclimate models (1m and 13cm, respectively) were built using the Condor high-performance computing facility. Because we were unable to parallelize the model, the spatial area we could run is $< 1\text{km}^2$. For the ultra-fine model, we ran a ‘snapshot’ of the larger area.

^o Onset® Computer Corporation, Bourne, MA 2018; www.onset.com

^p Garmin Ltd., Schaffhausen, Switzerland 2018; www.garmin.com

^q Emlid Ltd., Hong Kong 2018; www.emlid.com

Model validation

Model validation can be done for 2019-2020, matching the window of time during which the data loggers were deployed. Modeled soil temperatures for 10cm and 20cm depths will be compared with the field-recorded temperatures using R and RMSD statistics.

Results and Discussion

Building fine-resolution microclimate surfaces

In general, the values of predictions did not vary by spatial resolution (**Figure 5**). The implication of this result is that, at least at fine and ultra-fine spatial resolutions, model accuracy is independent of resolution. That is, even without direct validation, we can be confident in models' relative accuracy (i.e., compared with measured values). Thus, increasing the spatial resolution of a model above 1m (and, thus, the computational resources required to run it) is unlikely to benefit research without an ultra-fine spatial component.

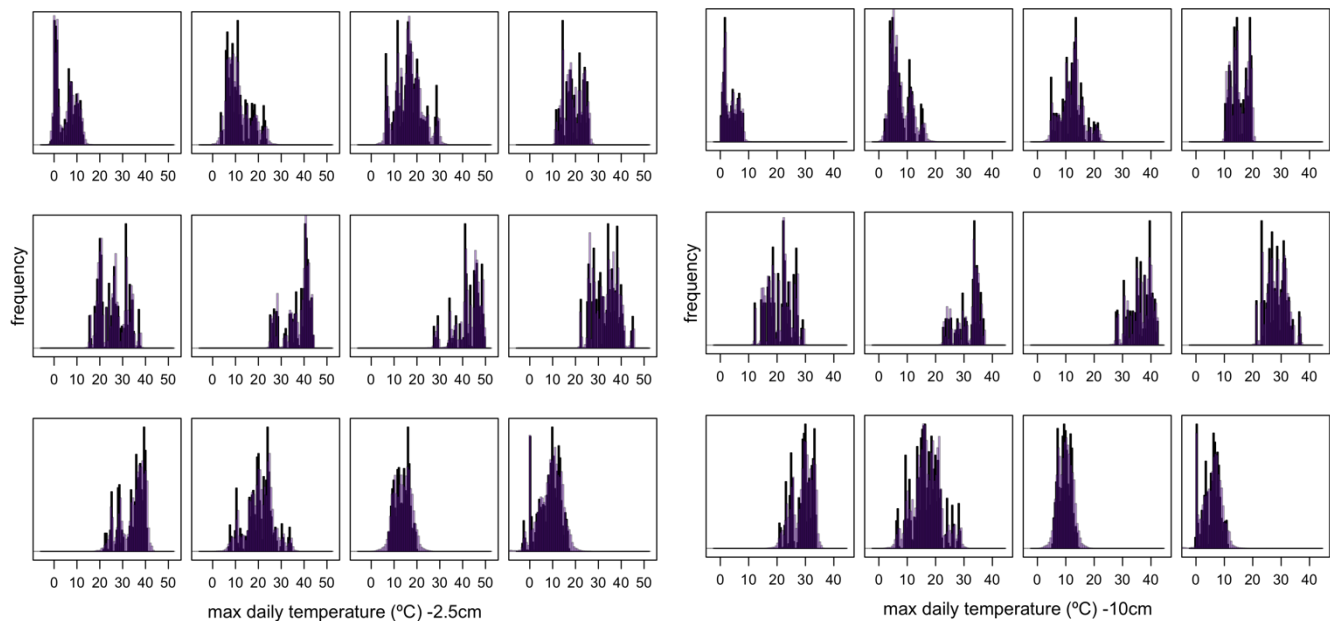
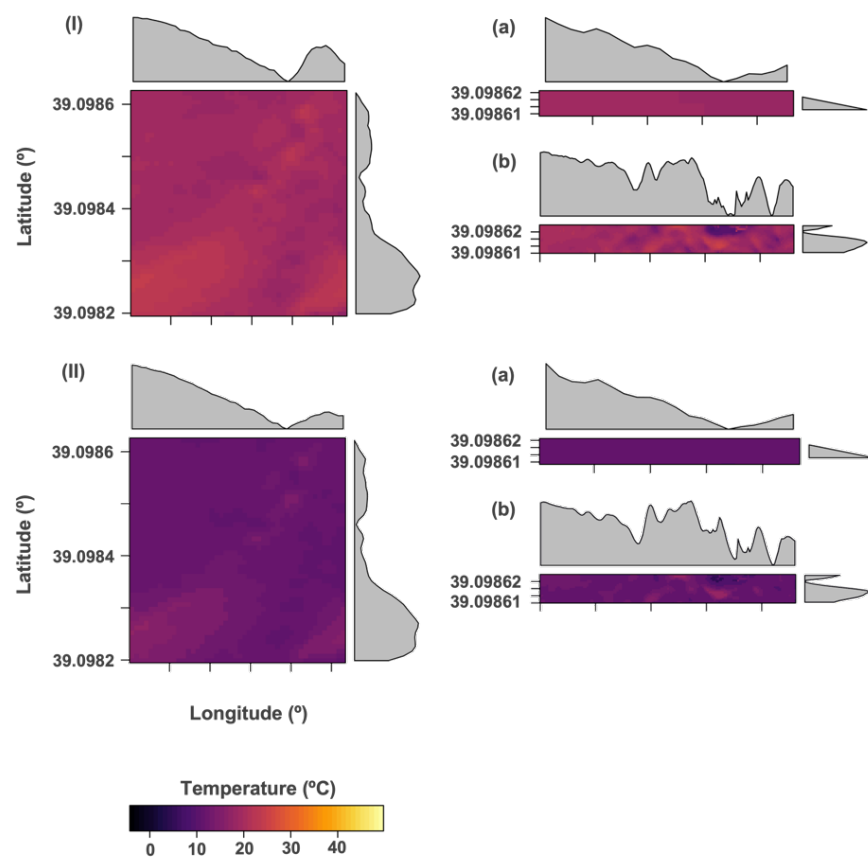


Figure 5. Overlapping frequency histograms of maximum daily predicted substrate temperatures over 12 months [Jan (top left) – Dec (bottom right)] at (a) 2.5cm and (b) 10cm depths. The black bars are predictions made using terrain inputs at a 1m spatial resolution, and the overlaid transparent purple bars show predictions made at 13cm.

In contrast, the spatial component of substrate temperatures was differentiated between 1m and 13cm (**Figure 6**). That is, the finer-resolution model showed finer-scale heterogeneity in conditions. Prior to this study, we could not have expected this result a priori. Indeed, based on the preponderance of ecological studies in which the resolution of spatial data is mismatched to the target organism(s), the expectation that microclimate heterogeneity would be undetectable at such fine resolutions would be reasonable. That is, research in spatial ecology has proceeded with the implicit assumption that, although spatial mismatch may be an important but unknown contributor to observed variation in natural populations, it is undetectable (otherwise, much previous effort would have been put into detecting it). Our findings illustrate that microclimate heterogeneity is both detectable and predictable at cm-scales.

Figure 6. Snapshot of microclimate model output, summarized from hourly data for a representative day (Julian Day 45: 14 Feb 2017). Data are maximum daily soil temperature at depths of (I) 2.5cm and (II) 10cm. The square plot shows data for a 50x50m area at a spatial resolution of 1m, with the white inset magnified at spatial resolutions of (a) 1m and (b) 13cm. The gray marginal plots show means across space. The remaining plots are in the Appendices.



Model Validation

While we have data logger data sufficient to validate accuracy of the 1m² model for KONZ, this task was not a priority for proof-of-concept. The raw data are shown in **Figure 7**. See the section on challenges posed by COVID-19 for further discussion.

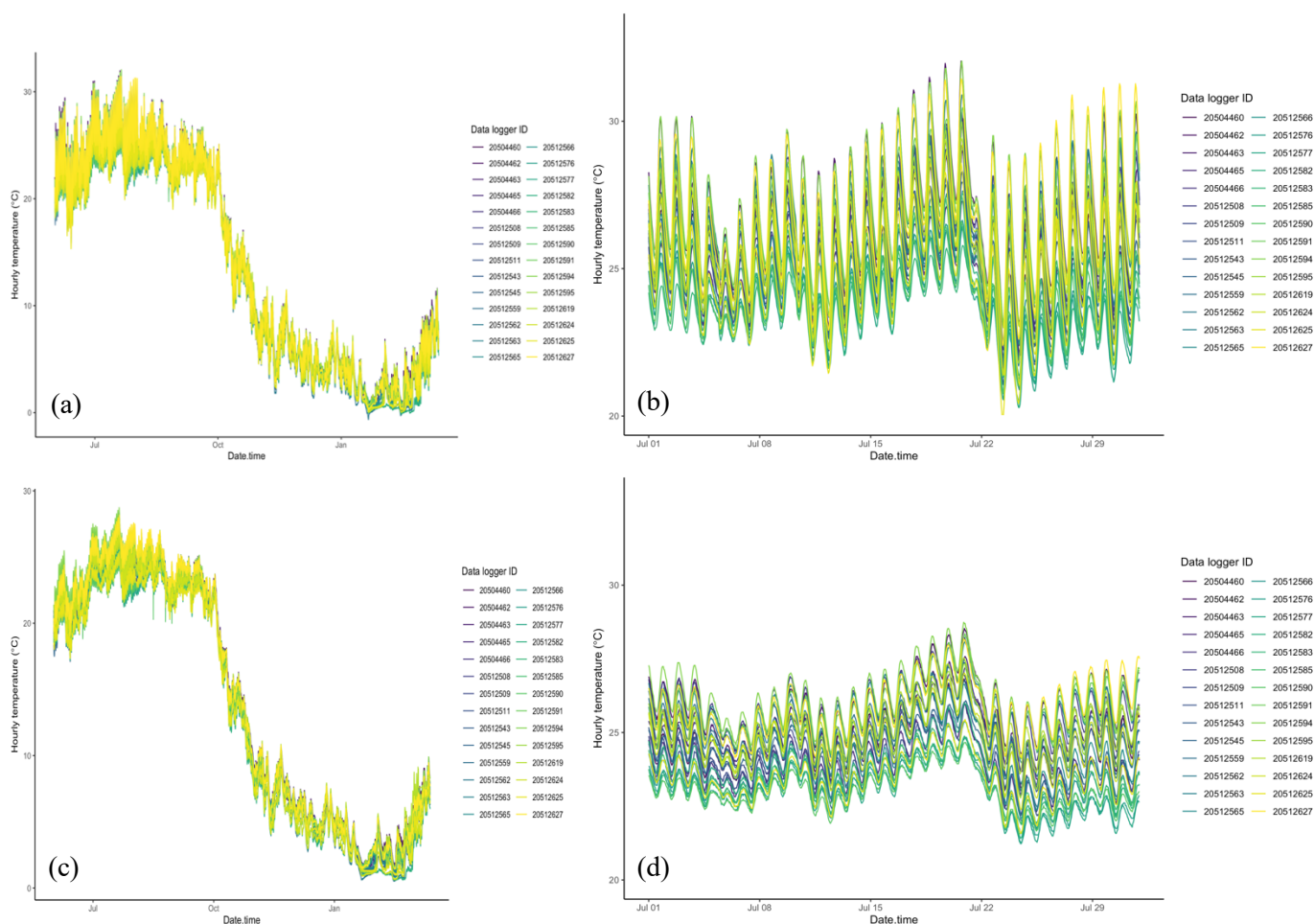


Figure 7. Raw substrate temperatures at two depths for KONZ, showing (a) all data for 10cm, (b) July data for 10cm, (c) all data for 20cm, and (d) July data for 20cm. Data logger IDs are matched with the grid coordinates of model predictions.

Conclusions and Implications for Future Research/Implementation

Our primary hypothesis is supported by the results. Substantial spatial differentiation between model outputs at 1m vs. 13cm indicates that spatial resolution can affect the detection of biologically important variation in microclimate conditions, likely without significantly impacting model accuracy. An important implication of this result is that the ubiquitous assumption that spatial mismatch could be leading to erroneous ecological conclusions and conservation and management decisions may indeed be incorrect. Given this – we predict, high – likelihood, we assert that (1) conservation and management decisions made using relatively coarse resolution models should be considered tentative, at best. Preferably, the indications of a coarse model would be followed up with data that are not mismatched. In addition, (2) the continuation of this research is critical for verifying and extending our findings.

Challenges and Lessons Learned

COVID-19 pandemic

The COVID-19 pandemic presented two significant challenges to this project – to our fieldwork and the resultant extent to which we were able to meet our research objectives. Along with the logistical limitations that directly affected the project were the less tangible, indirect impacts. Most importantly, the pandemic facilitated a shift in research priorities, namely that resources were shifted to research that was most directly relevant for the pandemic response. Many university-based researchers, ourselves included, paused non-critical research to shift our attention to protecting the health of ourselves, families, and communities. Although we have not completed all planned components of this project, proof-of-concept depends most significantly on the differentiation of models by spatial resolution, which was fulfilled.

Fieldwork

Our field objectives for spring/summer 2020 were to retrieve data loggers from both KONZ and NIWO, collect coordinates of data loggers using RTK GPS, and re-fly NIWO to collect aerial imagery from the remaining grids. Fieldwork at KONZ was scheduled for the weekend before the plots were scheduled to be burned as part of the prairie LTER. This weekend was also the point at which ISU restricted employee domestic travel due to the pandemic and halted in-person instruction. We received special permission to complete fieldwork and were able to retrieve most of the data loggers from KONZ. However, weather conditions for the location deteriorated and we made the decision not to collect RTK coordinates.

Restrictions on travel meant we were unable to collect additional data from NIWO. Two former lab members residing local to NIWO were approved by the station manager to retrieve the data loggers for us following approval of a comprehensive field safety plan. However, most of the data logger cables were damaged, presumably by rodents, and none of the data have been accessible so far.

Research Objectives

The inability to collect RTK coordinates for data loggers meant that we would be unable to validate the model at both 1m and 13cm resolutions and, thus, unable to compare accuracy between them. The inability to collect additional aerial data from NIWO meant that we would be unable to compare predictions based on terrain heterogeneity. However, this aim was not central to proof-of-concept. Indeed, the greatest lesson of the pandemic for this research was that, with the re-focusing of resources on other matters, it forced us to rethink which components were critical to proof-of-concept vs. those that were beneficial but inessential for this scope.

Site access

Data collection at the NIWO site proved challenging, providing a contrast with the relatively straightforward KONZ site that will be invaluable for planning follow-on sampling protocols. Success at this and other sites with similar environmental constraints (wind, precipitation, altitude, etc.) is strongly situationally dependent. The follow-on project can accommodate uncertainty in conditions by allowing

for flexibility in the sampling schedule, most easily by increasing the time allotment for ‘difficult’ sites and, as a result, the potential total sampling period for the follow-on project.

Alternatively – or in addition to – a flexible sampling schedule is the potential benefit of increasing the size of the field team. Minimally, collecting data from all 20 planned sites will require two teams of two [experienced] field technicians, both with UAV certificates. The addition of a third field team or creation of a larger team specifically to help collect data from the more difficult sites should be considered. In general, the amount of time reserved for field work at each site should be optimized to allow for the best chance of collecting the requisite data across all 20 sites without rushing the ‘easy’ sites like KONZ.

Finally, the intersection of the pandemic with other time constraints on field work, such as the KONZ plot burning schedule, highlighted the necessity of having dedicated field teams, regardless of their size, to carry out this project at an expanded scope. Outsourcing some of the fieldwork to experienced contractors may be feasible. However, completing all components would likely require two subcontracts per site – one to install/retrieve data loggers and one to collect the aerial imagery (due to the requirements of specialized UAV experience and an FAA certificate).

UAV equipment limitations

Unlike NIWO and KONZ, most sites are relatively heavily vegetated, which will require LiDAR-based, rather than RGB-based, data collection. UAV-mounted LiDAR units have been developed for multirotor UAVs that can accept a heavier payload but are less stable in high winds than the relatively lightweight fixed-wing unit used in this study. Thus, overcoming the challenges of UAV-facilitated data collection at difficult sites like NIWO will require a conservative sampling schedule that is as flexible as possible, given the constraints inherent in scheduling data collection at up to 20 geographically separated sites.

Another serious constraint on UAV-facilitated research is battery life. Completing the flight plan for KONZ (i.e., collecting aerial data within the entire 1km² grid) required six battery changes, even in relatively calm weather conditions. Where sites cannot be directly accessed by vehicle (as KONZ can), field teams must carry either 2x the minimum required number of batteries or a charger with high-enough capacity to re-charge all batteries at least once in a minimal amount of time. The scope of this requirement will increase if using a multirotor UAV, which are less efficient than fixed-wing craft. We used the latter strategy at both KONZ and NIWO and determined that an aluminum pack frame is the most efficient and safest means of carrying the charger and drone over difficult terrain. The ability to store some of the equipment at an intermediary site would also be useful at NIWO and other sites that may need to be accessed on foot.

RTK equipment limitations

The original plan was to collect coordinates of all data loggers using a receiver capable of differential corrections (i.e., RTK). This approach would increase coordinate accuracy to cm-scale. However, the weather and pandemic-constrained fieldwork timeline restricted our capacity to capture those coordinates with anything other than a handheld GPS unit, limiting the usefulness of model validation. That is, without cm-scale accuracy of data logger coordinates, we cannot compare accuracy of the 1m vs. 13cm models. This outcome is not an equipment issue, per se. However, it does highlight the necessity of building sufficient time into the fieldwork plan to allow for conscientious equipment set up, testing, and troubleshooting.

Model parallelization

Although we were unable to run the microclimate model in parallel across the Condo system, there were no indications that this aim would not be achievable (even relatively quickly) with a deeper computational skillset. Alternatively, there may be more efficient approaches to running the model that reduce

computation time similarly. The expanded project will prioritize this aim as its first go/no-go and include a programming specialist among its personnel.

Next Steps

Build new collaborations

In line with our mutual experience and interests, we originally conceived this research within the context of predicting the impacts of climate change on vertebrate species, particularly reptiles. However, discussions with colleagues at NEON and elsewhere have generated considerable interest in our project among plant and soil biologists. Thus, rather than focusing specifically on reptiles, we will propose to use ‘containers’ corresponding to different organismal shapes, a common approach in ecophysiology. Broadening the biological scope of the project does not change the overall modeling approach (i.e., microclimate and dynamic energy budget models to capture organisms throughout their life cycles) but should make it more immediately accessible to a wider range of ecological subfields.

We will also seek to solidify our partnership with NEON, which collects and manages extensive biological datasets in addition to environmental data. Our aim will be to obtain 20 letters of support (i.e., from each of the 20 domain management partners), including approval of a preliminary plan for site access and post-project data integration and dissemination, focusing on the plant and invertebrate datasets. NEON input will also be beneficial for server design and data archiving

Finally, Dr. Kya Dahlin, Assistant Professor of Geography at Michigan State University, will join the research team as a co-PI. Dr. Dahlin uses UAV remote sensing and process-based modeling to understand post-disturbance vegetation patterns, and her expertise will ensure that this research is maximally relevant to plant biologists.

Prototype data logger cages

The level of (presumably rodent) damage to data loggers was much higher than expected (100%) at NIWO. Thus, rodent-proof protective barriers will be required for future installations. As our current research program permits, we will be designing and testing prototype barrier cages *ex situ* in a fluctuating incubation chamber and *in situ* to confirm they do not affect temperature readings (i.e., because they will likely need to be metal).

Design data server

The data generated by the expanded project will require a dedicated server, which will be designed with advice from NEON and Information Technology staff at Michigan State University.

Draft and submit expanded proposal

The expanded-scope project will propose to collect aerial imagery from sites in all 20 NEON ecological domains. Our general approach will be much the same. However, we will include a go/no-go point for model parallelization.

Literature Cited

- Abatzoglou JT (2013) Development of gridded surface meteorological data for ecological applications and modelling. *International Journal of Climatology* 33, 121–131.
- Anderson K & Gaston KJ (2013) Lightweight unmanned aerial vehicles will revolutionize spatial ecology. *Frontiers in Ecology & the Environment* 11(3): 138-146.
- Angilletta Jr MJ, Winters RS, & Dunham AE (2009) Thermal effects on the energetics of lizard embryos: implications for hatchling phenotypes. *Ecology* 81: 2957-2968.
- Becker C, Häni N, Rosinskaya E, d'Angelo E & Strecha C (2017) Classification of Aerial Photogrammetric 3D Point Clouds. *arXiv:1705.08374 [cs]*.
- Briscoe NJ, Handasyde KA, Griffiths SR, Porter WP, Krockenberger A & Kearney MR (2014) Tree-hugging koalas demonstrate a novel thermoregulatory mechanism for arboreal mammals. *Biology Letters* 10: 20140235.
- Carter A, Kearney MR, Mitchell NJ, Hartley S, Porter WP & Nelson NJ (2015) Modelling the soil microclimate: does the spatial or temporal resolution of input parameters matter? *Frontiers in Biogeography* 7: 138-154.
- Carter AL, Kearney MR, Hartley S, Porter WP & Nelson NJ (2018) Geostatistical interpolation can reliably extend coverage of a very high-resolution model of temperature-dependent sex determination. *Journal of Biogeography* 45, 652–663.
- Faye E, Rebaudo F, Yáñez-Cajo D, Cauvy-Fraunié D & Dangles O (2016) A toolbox for studying thermal heterogeneity across spatial scales: from unmanned aerial vehicle imagery to landscape metrics. *Methods in Ecology and Evolution* 7: 437-446.
- Geiger R, Aron RH & Todhunter P (2009) The climate near the ground. 7th Ed. Lanham, MD: Rowman & Littlefield. 642 pp.
- Goulden T, Hopkinson C, Jamieson R & Sterling S (2016) Sensitivity of DEM, slope, aspect and watershed attributes to LiDAR measurement uncertainty. *Remote Sensing of Environment* 179, 23–35.
- Guisan A, Graham CH, Elith J, Huettmann F (2007) Sensitivity of predictive species distribution models to change in grain size. *Diversity and Distributions* 13: 332-340.
- Guisan A & Zimmerman NE (2000) Predictive habitat distribution models in ecology. *Ecological Modelling* 135: 147-186.
- Hannah L, Flint L, Syphard AD, Moritz MA, Buckley LB & McCullough IM (2014) Fine-grain modeling of species' response to climate change: holdouts, stepping-stones, and microrefugia. *Trends in Ecology and Evolution* 29: 390--397.
- He KS, Bradley BA, Cord AF, Rocchini D, Tuanmu M-N, Schmidtlein S, Turner W, Wegmann M & Pettorelli N (2015) Will remote sensing shape the next generation of species distribution models? *Remote Sensing in Ecology and Conservation* doi: 10.1002/rse2.7.
- Hijmans RJ, Cameron SE, Parra JL, Jones PG & Jarvis A (2005) Very high resolution interpolated climate surfaces for global land areas. *International Journal of Climatology* 25: 1965–1978.
- Horn BKP (1981) Hill shading and the reflectance map. *Proceedings of the IEEE* 69, 14–47.
- Hutchinson GE & MacArthur RH (1959) A theoretical ecological model of size distributions among species of animals. *The American Naturalist* 93: 117-125.

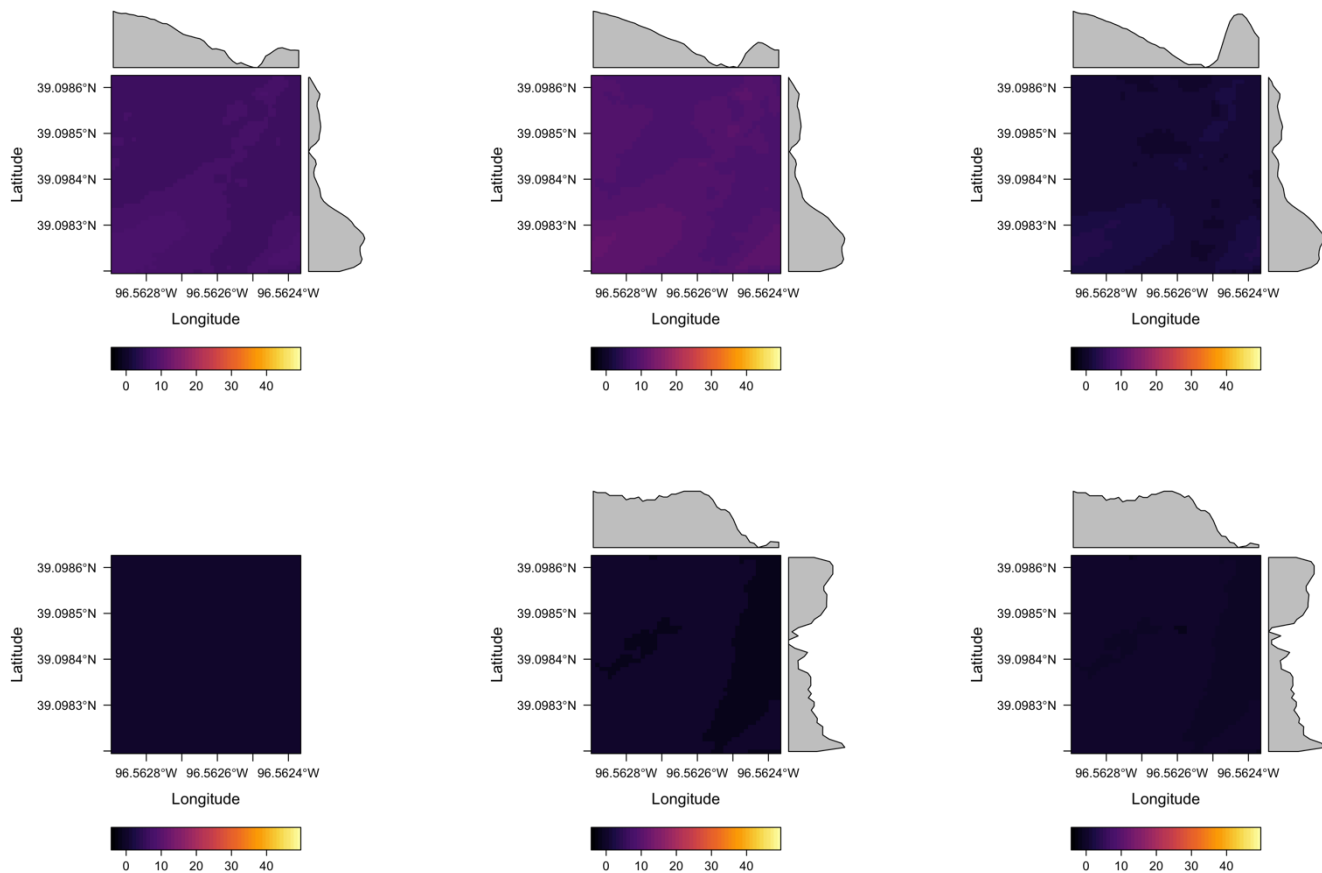
- Kearney MR, Isaac AP & Porter WP (2014) microclim: global estimates of hourly microclimate based on long-term monthly climate averages. *Scientific Data* 1 doi: [10.1038/sdata.2014.6].
- Kearney MR, Matzelle A & Helmuth B (2012) Biomechanics meets the ecological niche: the importance of temporal data resolution. *The Journal of Experimental Biology* 215: 922-933.
- Kearney MR & Porter WP (2004) Mapping the fundamental niche: physiology, climate, and the distribution of a nocturnal lizard. *Ecology* 85: 3119-3131.
- Kearney MR & Porter WP (2009) Mechanistic niche modelling: combining physiological and spatial data to predict species' ranges. *Ecology Letters* 12: 334-50.
- Kearney MR & Porter WP (2016) NicheMapR- an R package for biophysical modelling: the microclimate model. *Ecography* doi: 10.1111/ecog.02360.
- Kearney MR & Porter WP (2017) NicheMapR - an R package for biophysical modelling: the microclimate model. *Ecography* 40, 664–674.
- Koepke P, Hess M, Schult I & Shettle EP (1997) Global Aerosol Data Set. *Report No. 243, Max-Planck-Institut für Meteorologie*, Hamburg. ISSN 0937-1060.
- Korb J (2003) Thermoregulation and ventilation of termite mounds. *Naturwissenschaften* 90: 212-219.
- Kotamarthi R, Mearns L, Hayhoe K, Castro CL & Wuebbles D (2016) Use of climate information for decision-making and impacts research: state of our understanding. Prepared for the Department of Defense, Strategic Environmental Research and Development Program. 55pp.
- Leempoel K, Parisod C, Geiser C, Daprà L, Vittoz P & Joost S (2015) Very high resolution digital elevation models: are multi-scale derived variables ecologically relevant? *Methods in Ecology and Evolution* doi: 10.1111/2041-210X.12427.
- Lenoir J, Hattab T & Pierre G (2016) Climatic microrefugia under anthropogenic climate change: implications for species redistribution. *Ecography* doi: 10.1111/ecog.02788.
- Loiselle BA, Howell CA, Graham CH, Goerck JM, Brooks T, Smith KG & Williams PH (2003) Avoiding pitfalls of using species distribution models in conservation planning. *Conservation Biology* 17: 1591–1600.
- Maréchaux I, Rodrigues ASL & Charpentier A (2016) The value of coarse species range maps to inform local biodiversity conservation in a global context. *Ecography* 10.1111/ecog.02598.
- McCullough EC & Porter WP (1971) Computing clear day solar radiation spectra for the terrestrial ecological environment. *Ecology* 52: 1008-1015.
- Meineri E & Hylander K (2016) Fine-grain climate models based on climate station and comprehensive topographic information improve microrefugia detection. *Ecography* doi: 10.1111/ecog.02494.
- Mitchell NJ, Allendorf FW, Keall SN, Daugherty CH & Nelson NJ (2010) Demographic effects of temperature-dependent sex determination: will tuatara survive global warming? *Global Change Biology* 16: 60-72.
- Mosblech NAS, Bush MB, & van Woesik R (2011) On metapopulations and microrefugia: paleoecological insights. *Journal of Biogeography* 38(30): 419-429.
- Palmer T (2014) Build high-resolution global climate models. *Nature* 515: 338-339.
- Pimm SL, Alibhai S, Bergl R, Dehgan A, Giri C, Jewell Z, Joppa L, Kays R & Loarie S (2015) Emerging technologies to conserve biodiversity. *Trends in Ecology & Evolution* 30(11): 685-696.
- Potter KA, Woods HA & Pincebourde S (2013) Microclimatic challenges in global change biology. *Global Change Biology* 19: 2932-2939.

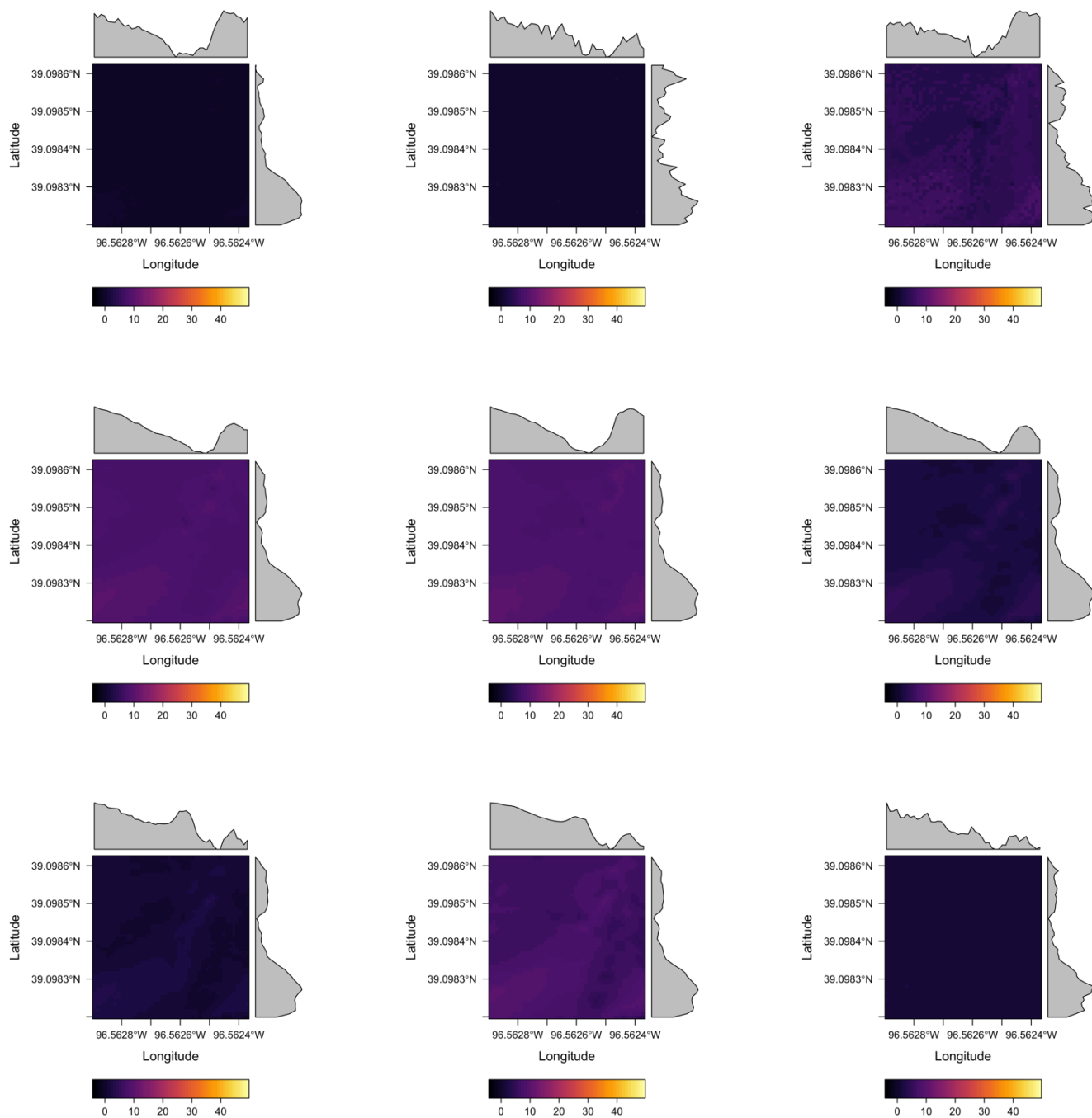
- Porter WP, Ostrowski S & Williams JB (2010) Modeling animal landscapes. *Physiological and Biochemical Zoology* doi: 10.1086/656181.
- Porter WP, Sabo JL, Tracy CR, Reichman OJ & Ramankutty N (2002) Physiology on a landscape scale: plant-animal interactions. *Integrative and Comparative Biology* 42: 431-453.
- Rull V (2009) Microrefugia. *Journal of Biogeography* 36(3): 481-484.
- Sears MW, Raskin E & Angiletta Jr MJ (2011) The world is not flat: defining relevant thermal landscapes in the context of climate change. *Integrative and Comparative Biology* 51: 666-675.
- Stubbs JL, Kearney MR, Whiting SD & Mitchell NJ (2014) Models of primary sex ratios at a major flatback turtle rookery show an anomalous masculinising trend. *Climate Change Responses* doi:10.1186/s40665-014-0003-3.
- Trevisani S & Rocca M (2015) MAD: robust image texture analysis for applications in high resolution geomorphometry. *Computers & Geosciences* 81: 78-92.
- Wang T, Hamann A, Spittlehouse D & Carroll C (2016) Locally downscaled and spatially customizable climate data for historical and future periods for North America. *PLoS ONE* 11(6): e0156720.
- Wiens JJ (2016) Climate-related local extinctions are already widespread among plant and animal species. *PLoS Biology* 14(12): e2001104.
- Wilson D (1998) Nest-site selection: microhabitat variation and its effects on the survival of turtle embryos. *Ecology* 79: 1884-1892.

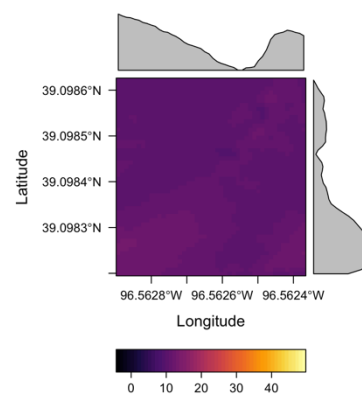
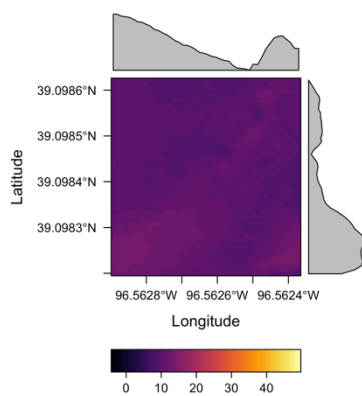
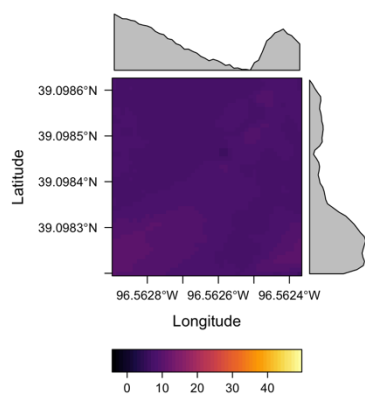
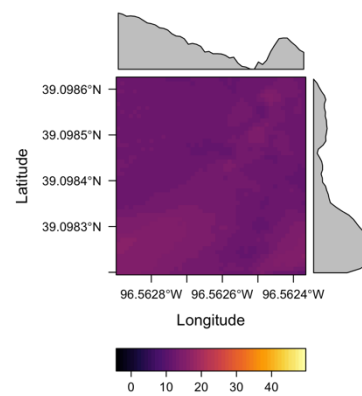
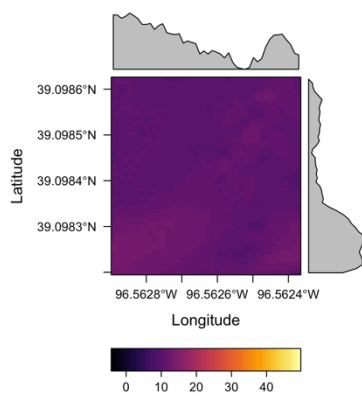
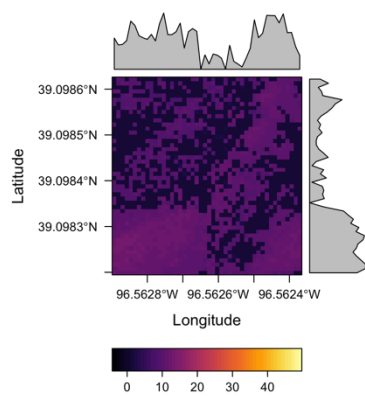
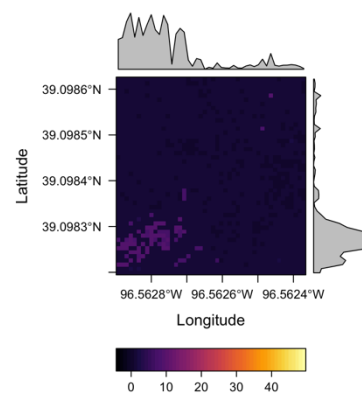
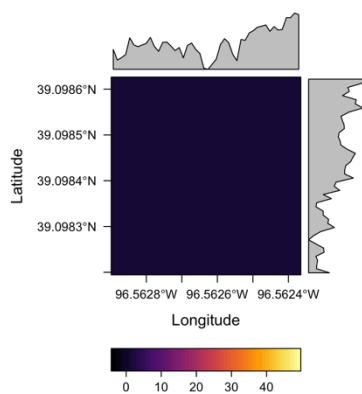
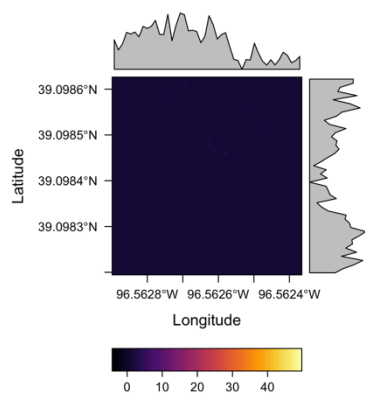
Appendices

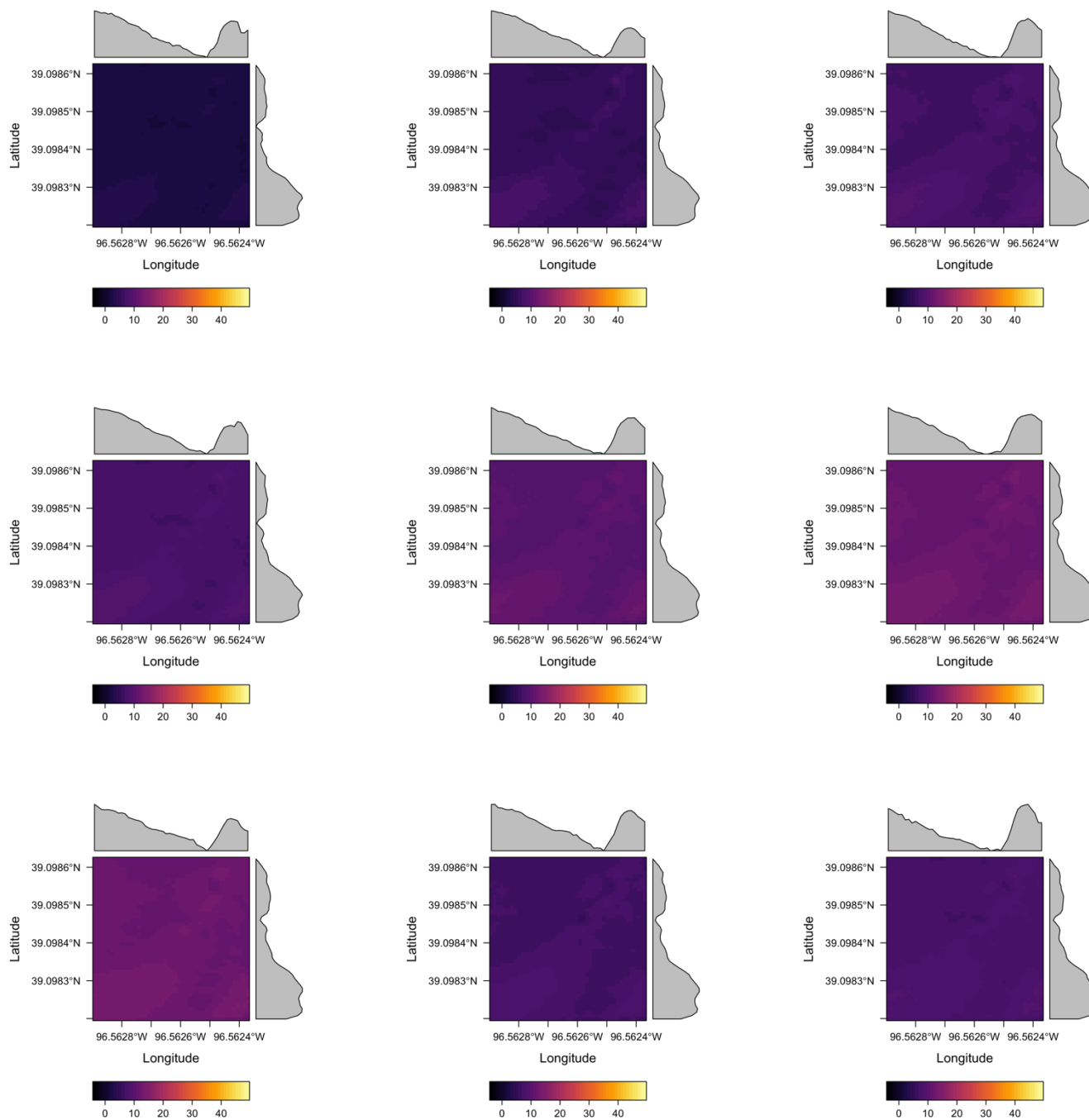
A. Supporting Data

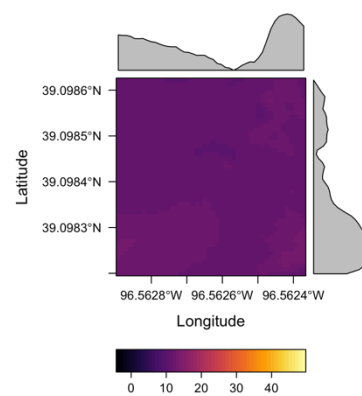
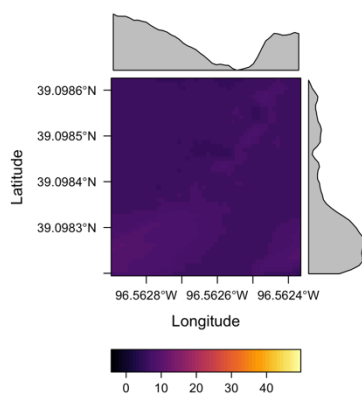
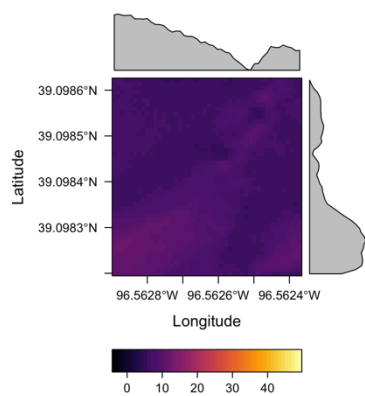
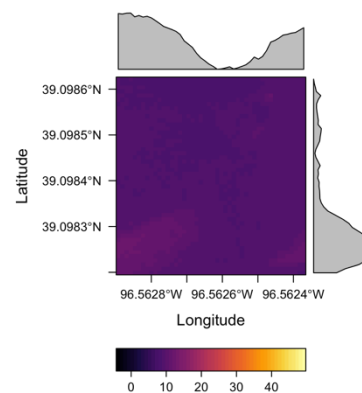
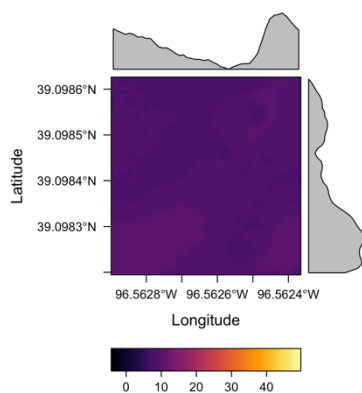
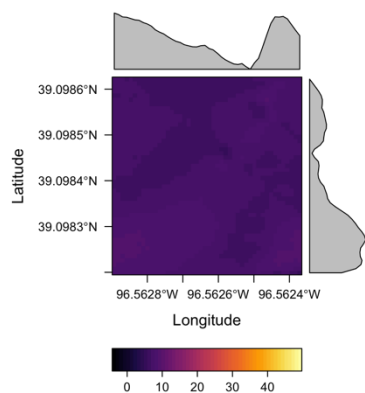
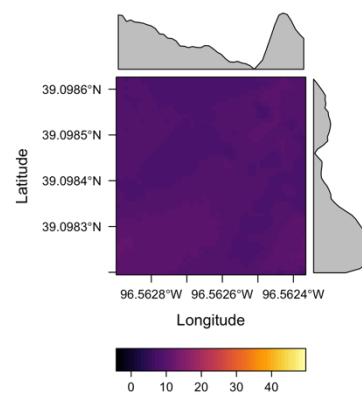
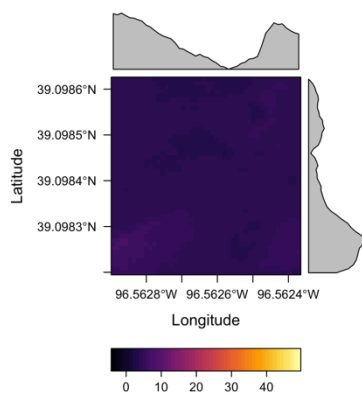
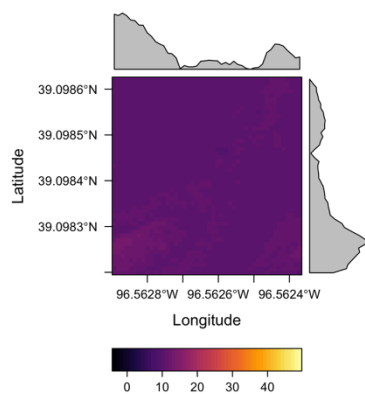
This set of plots shows a series of maximum daily substrate temperature snapshots for KONZ in 2017 at a substrate depth of 2.5cm and spatial resolution of 1m, with each plot representing the maximum [hourly] temperature (°C) for each 24hr period. Plots are ordered left-to-right across a page, starting with 01-01-2017. The gray marginal plots show means across 2D space. These 365 plots are provided as an example of the wealth of data created by a single run of the microclimate model. The complete raw data contains 8,760 hourly points of up to 440 variables (~35MB) *per pixel* of terrain input.

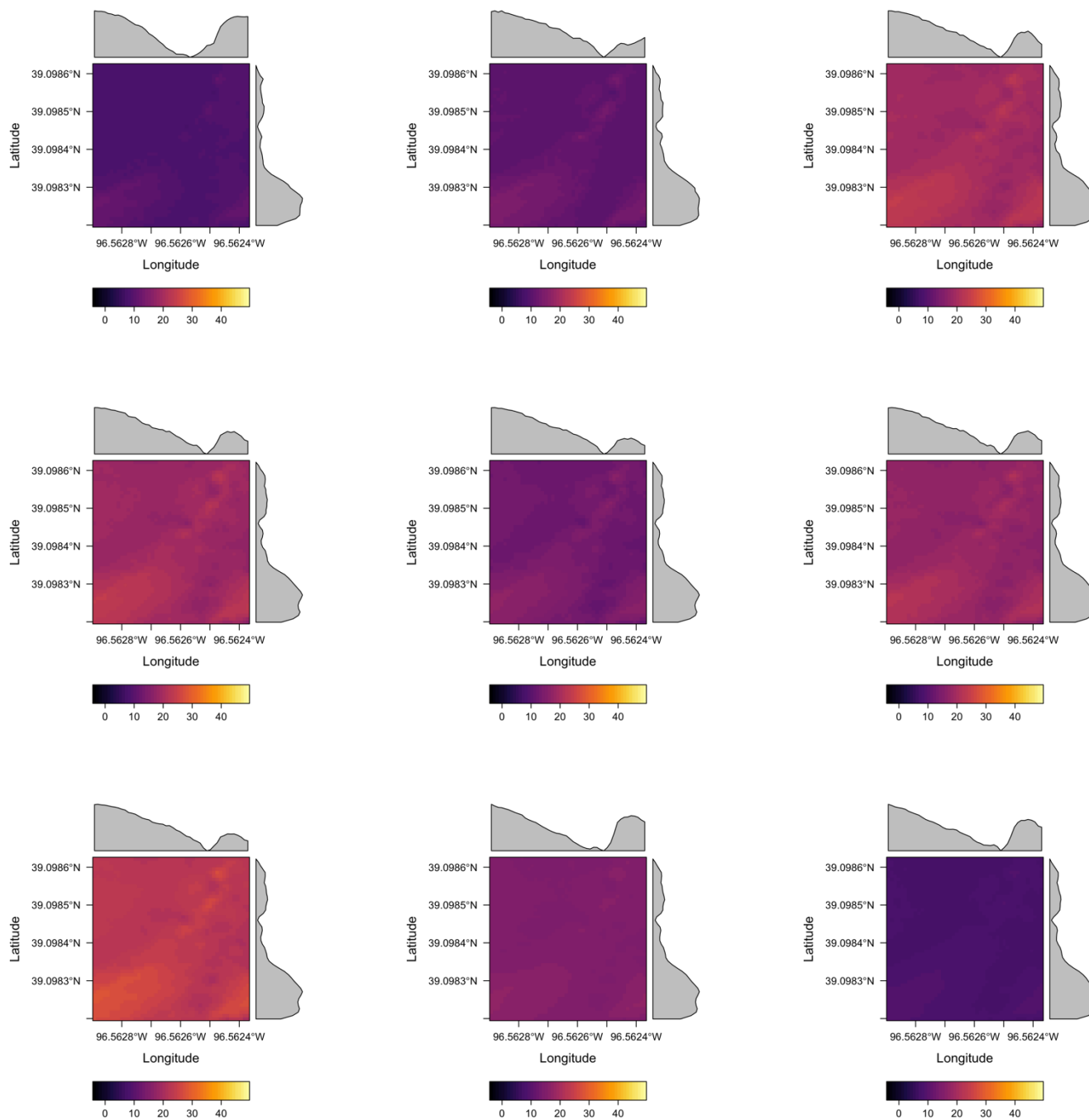




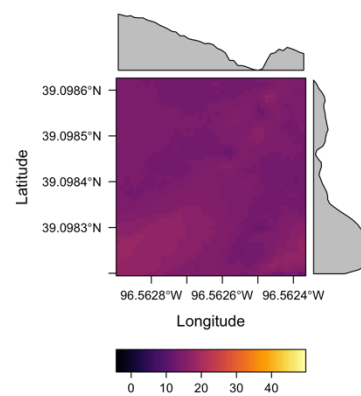
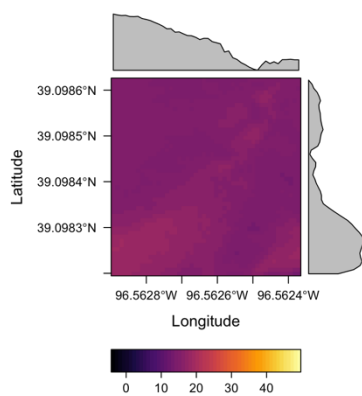
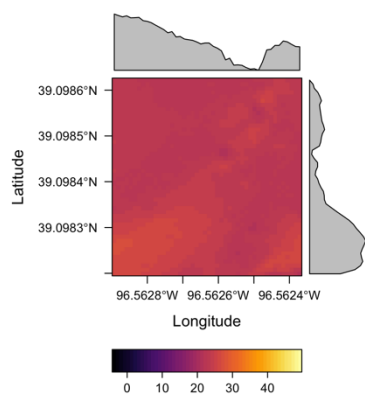
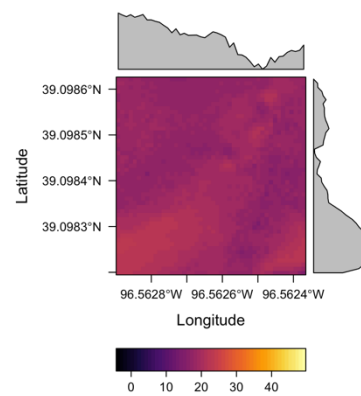
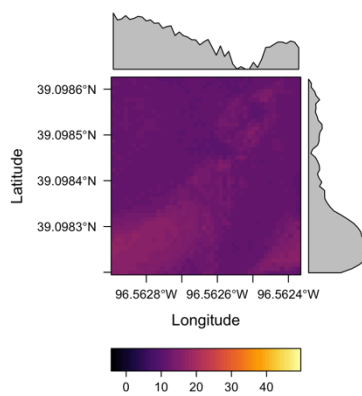
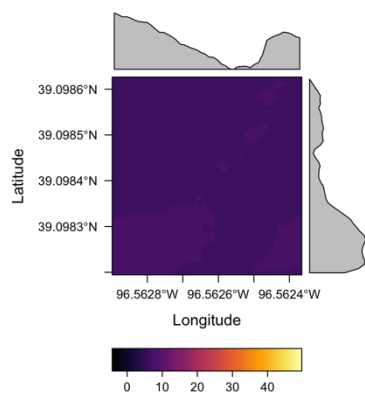
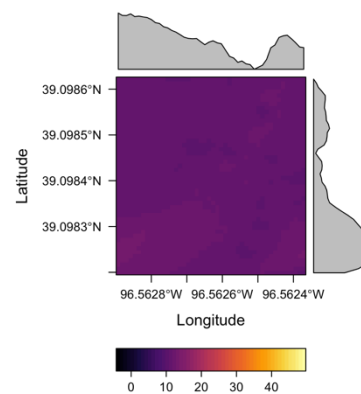
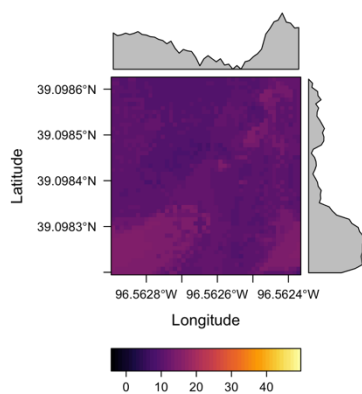
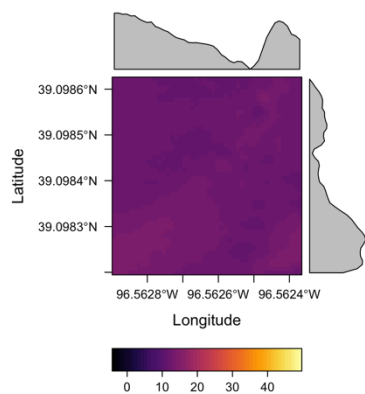


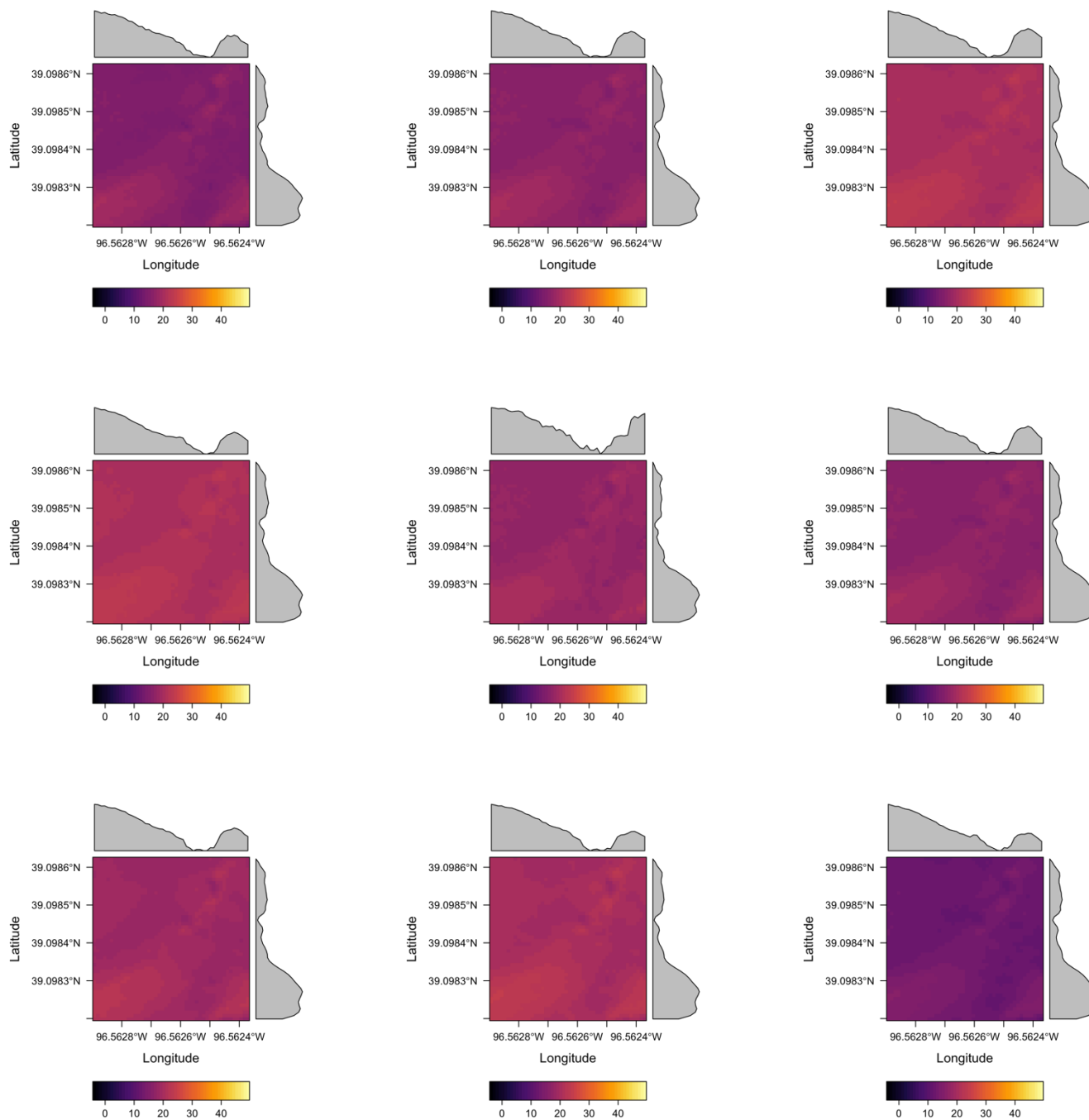


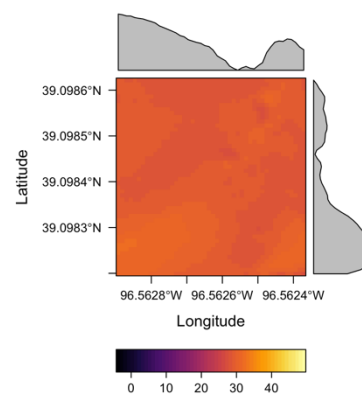
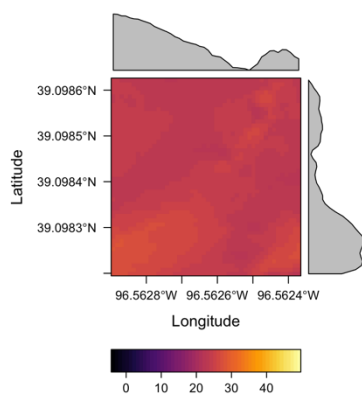
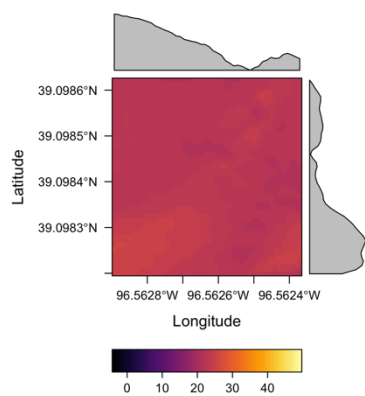
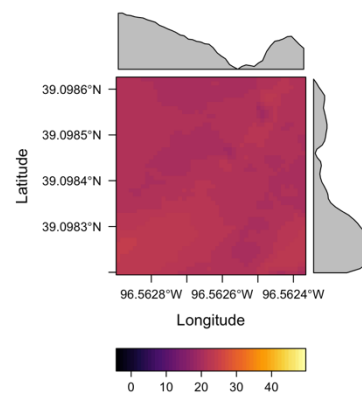
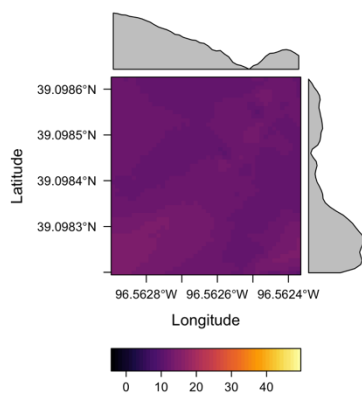
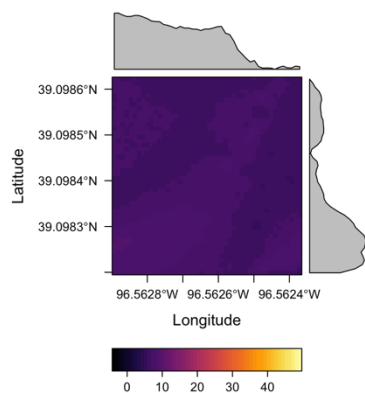
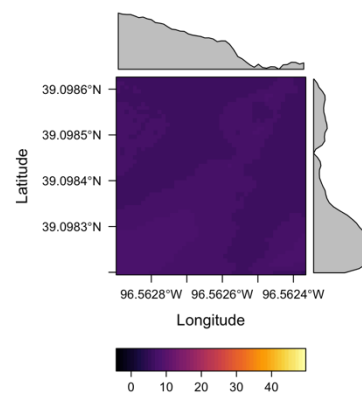
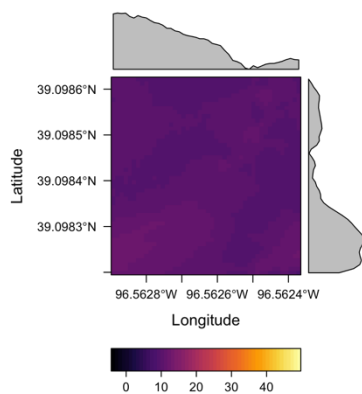
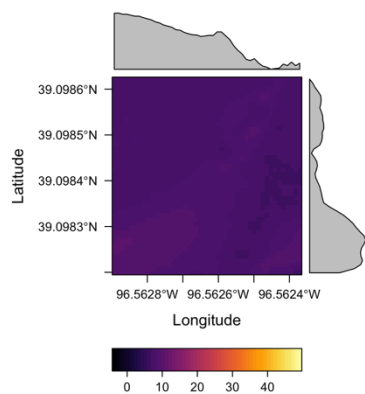


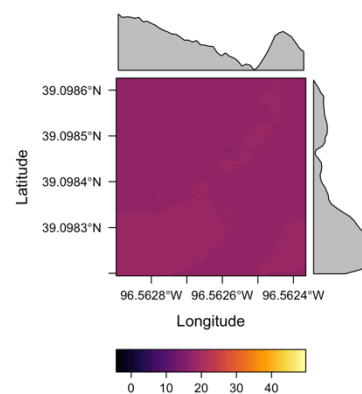
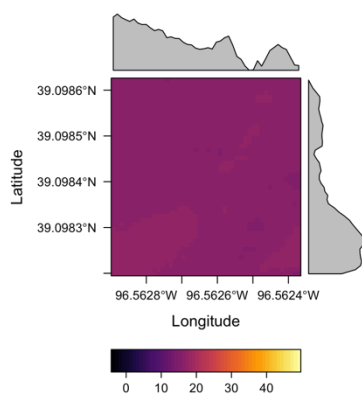
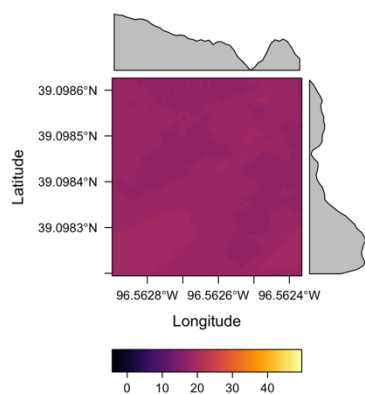
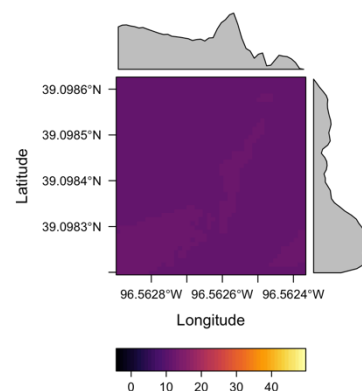
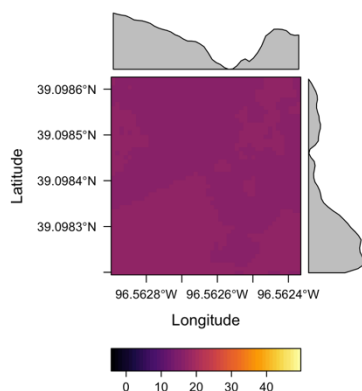
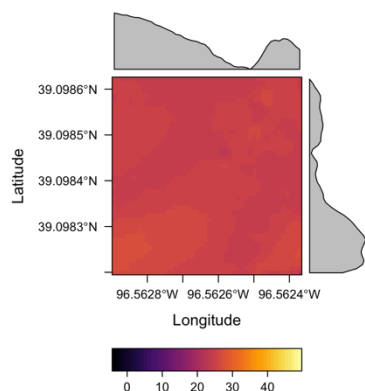
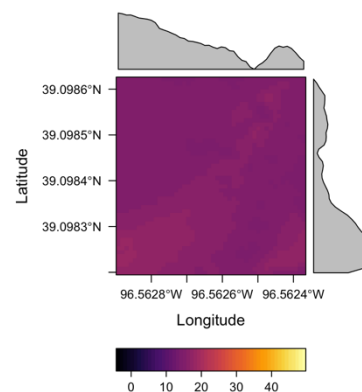
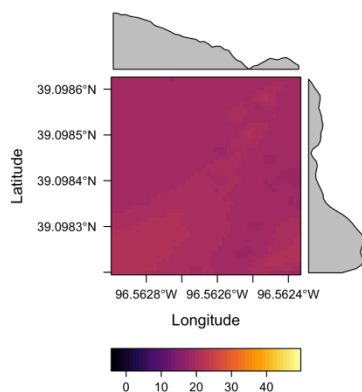
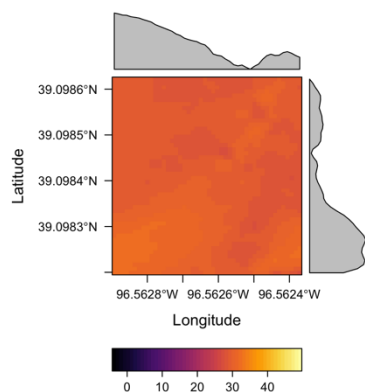


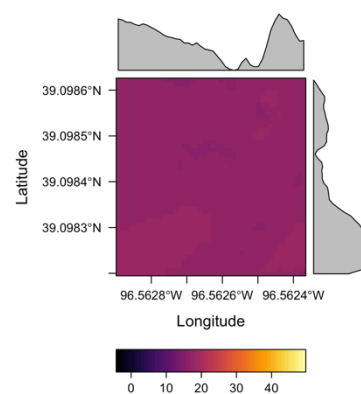
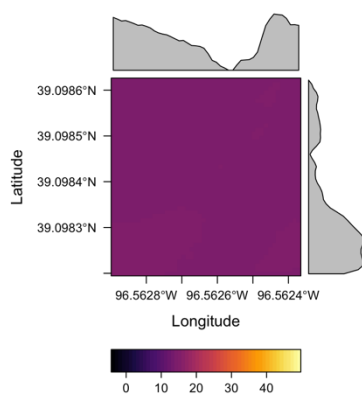
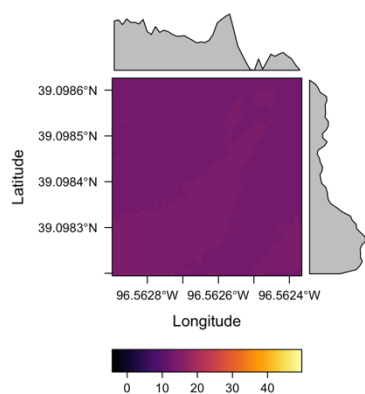
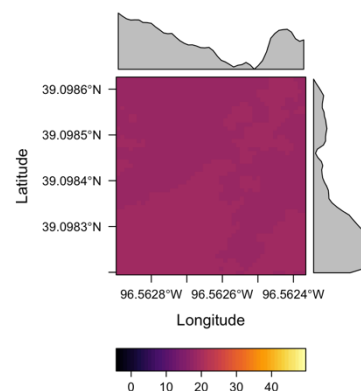
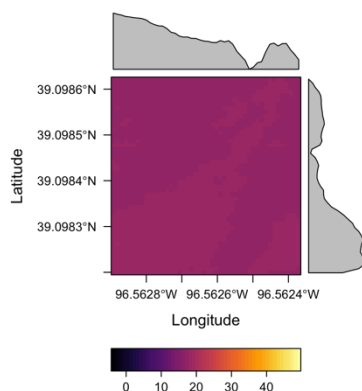
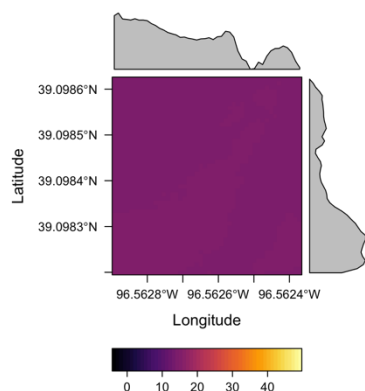
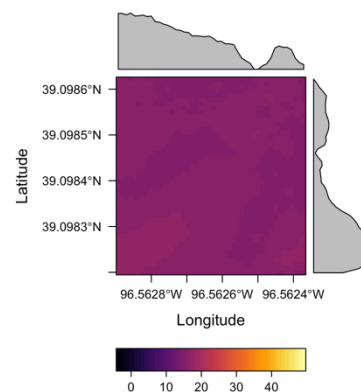
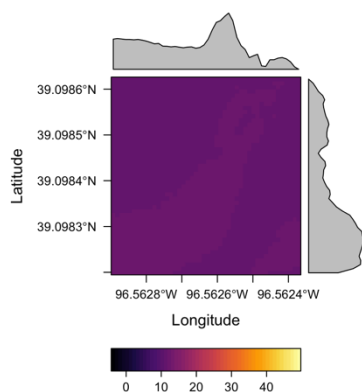
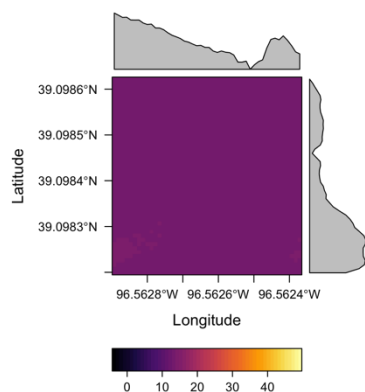
03-01-2017

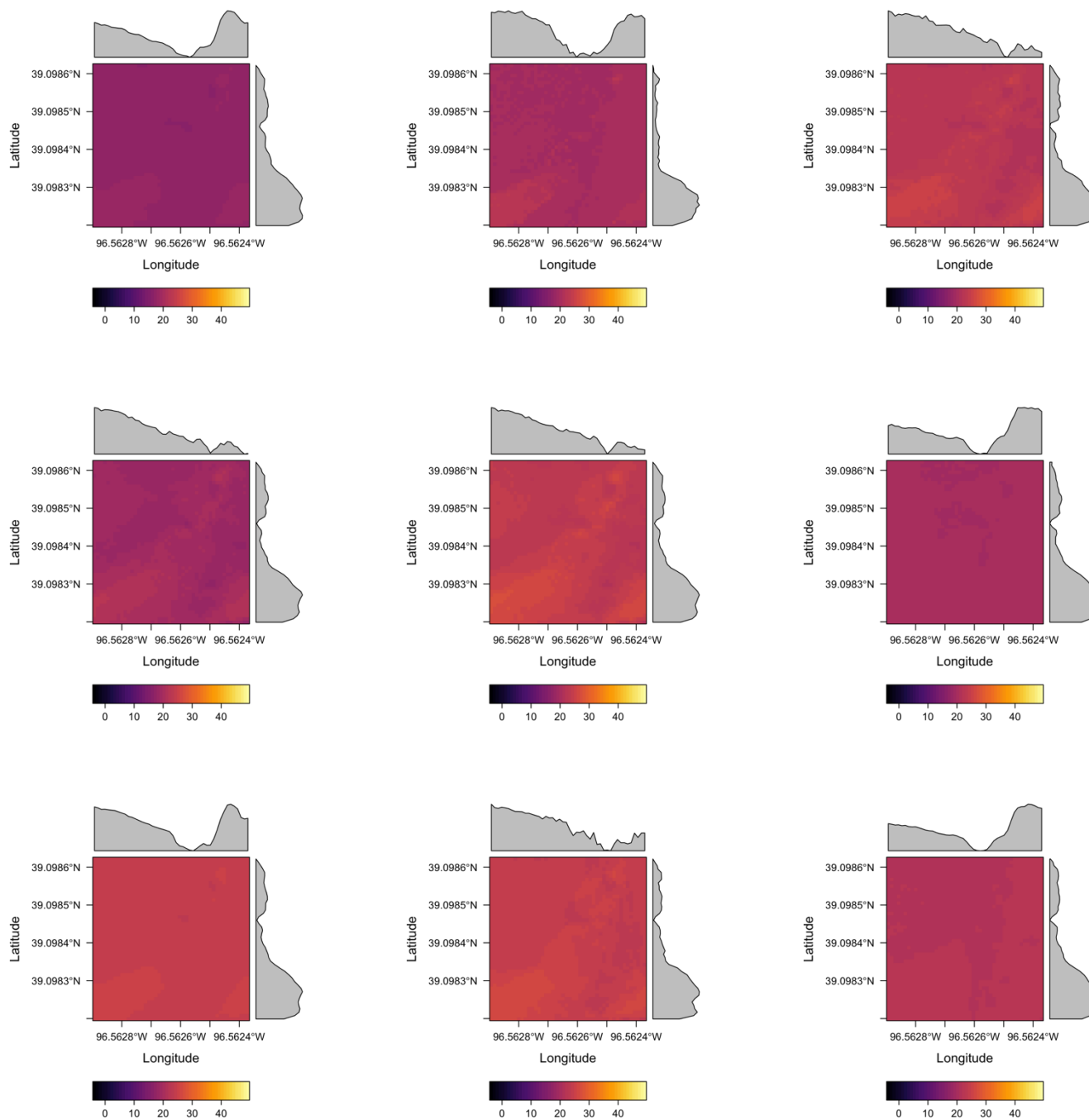


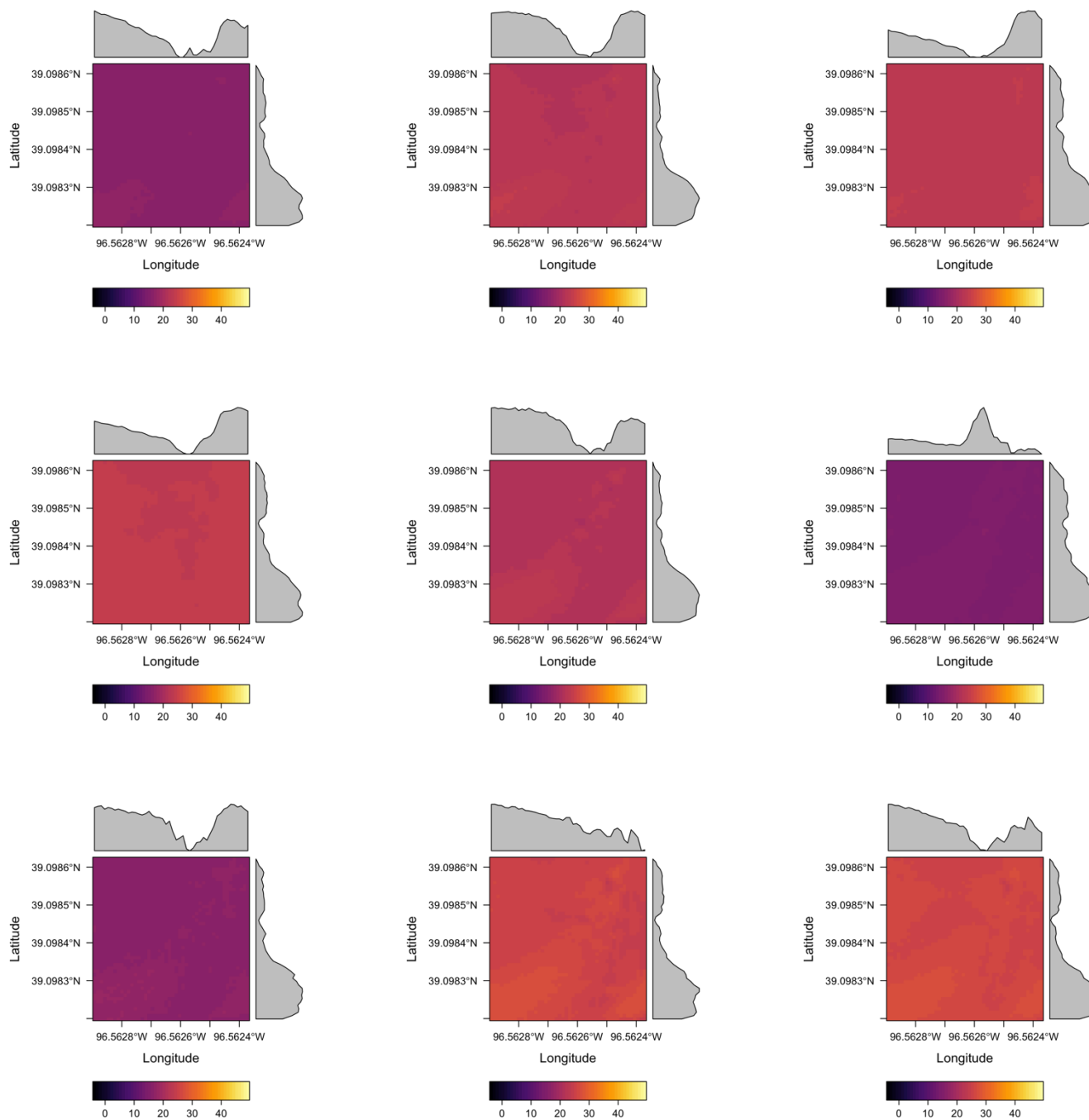


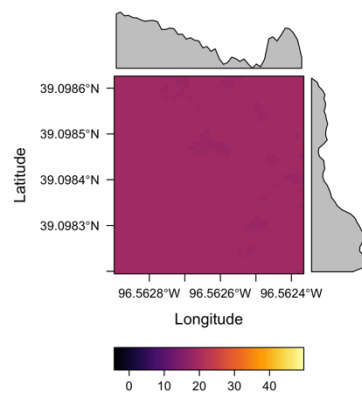
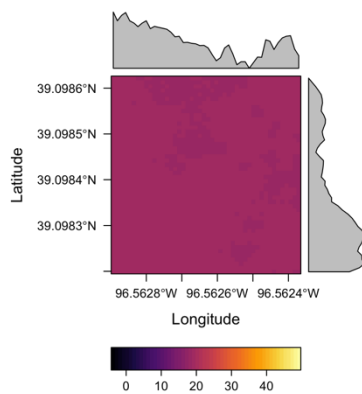
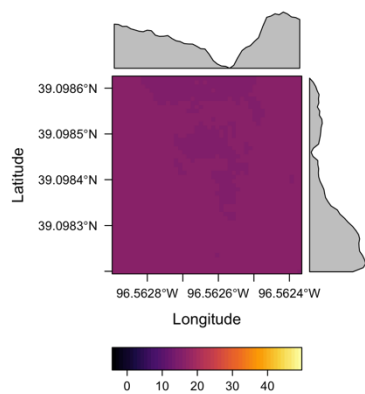
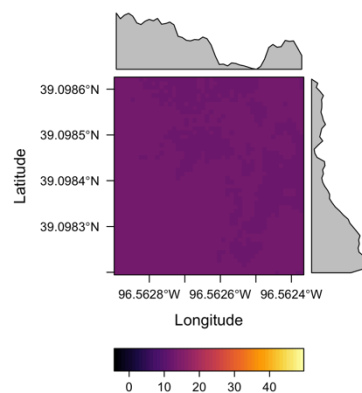
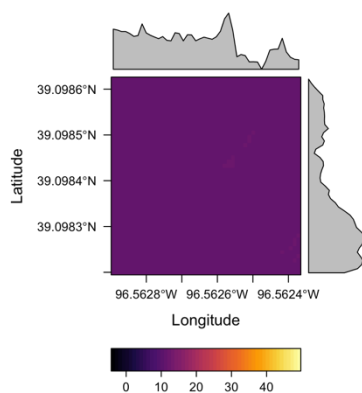
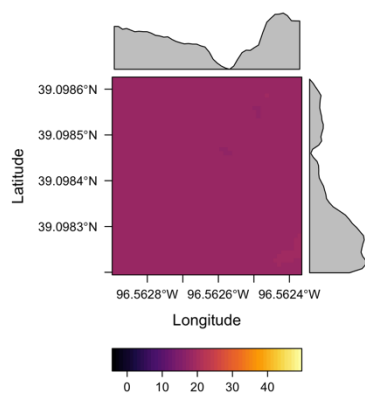
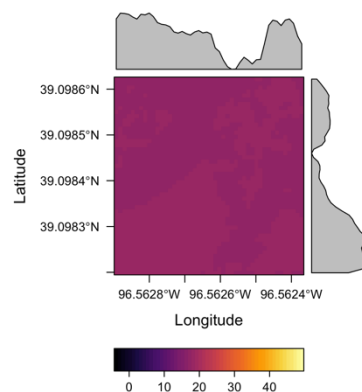
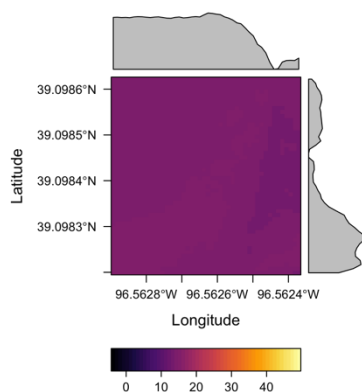
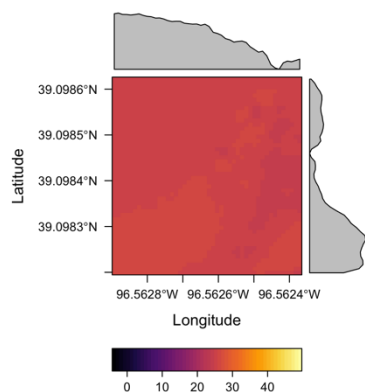


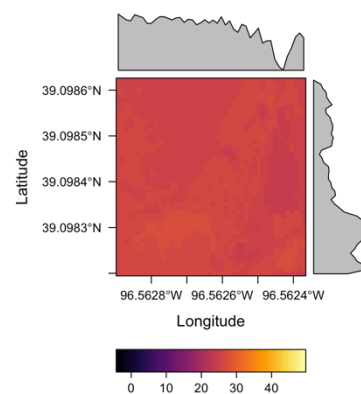
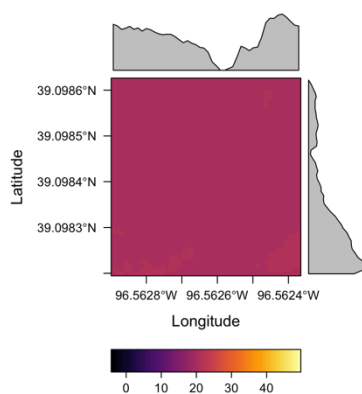
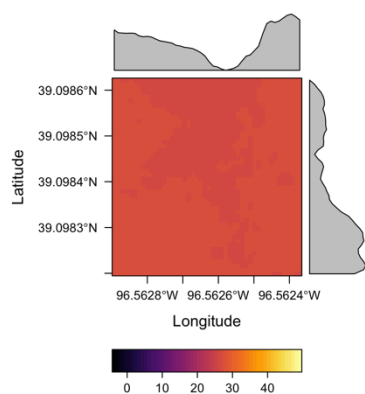
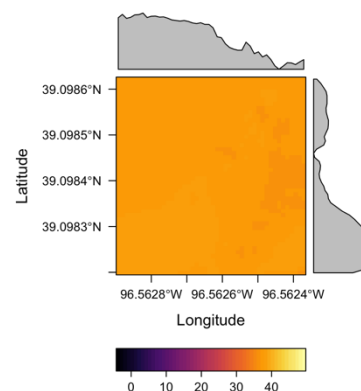
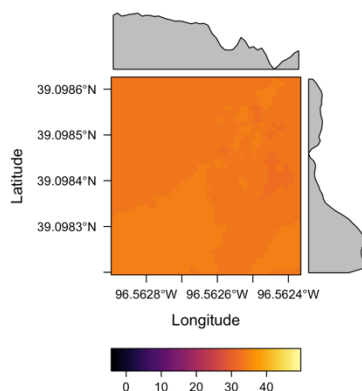
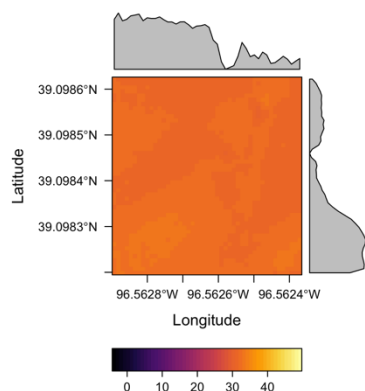
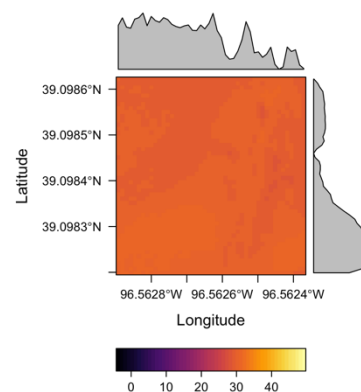
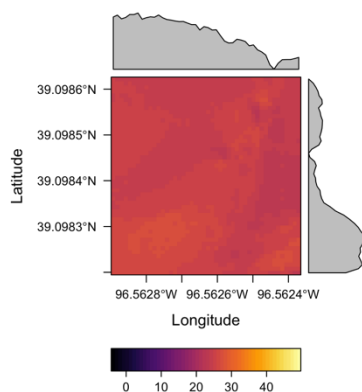
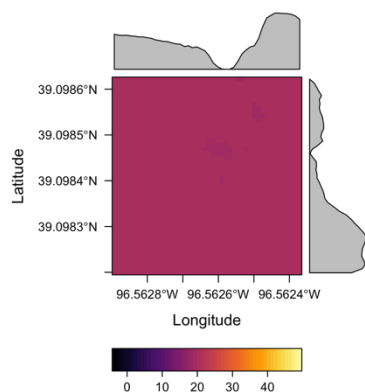


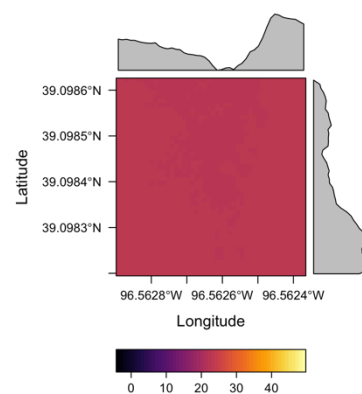
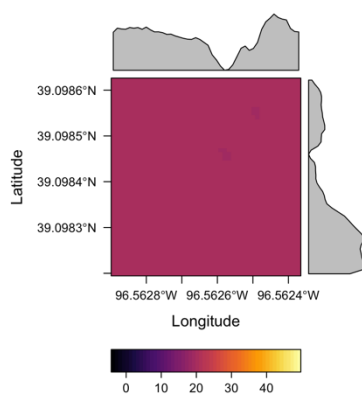
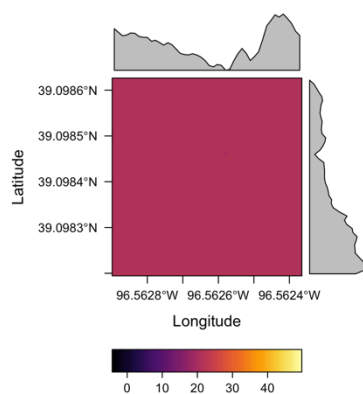
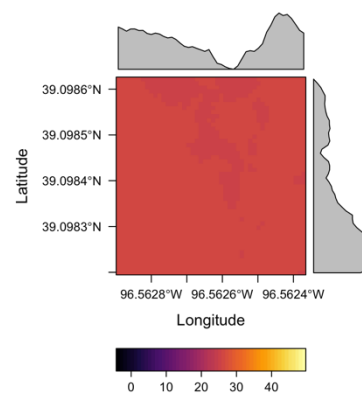
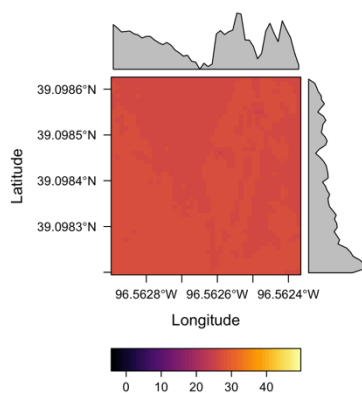
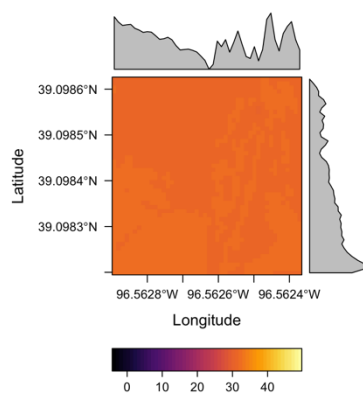
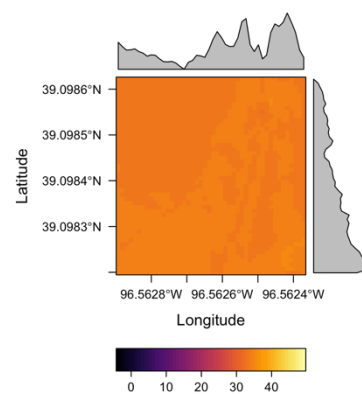
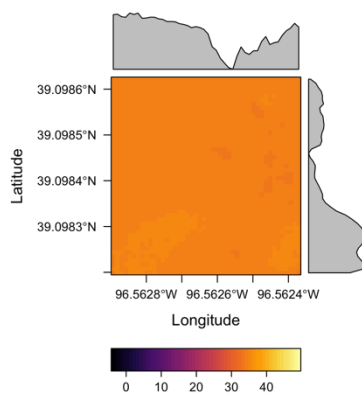
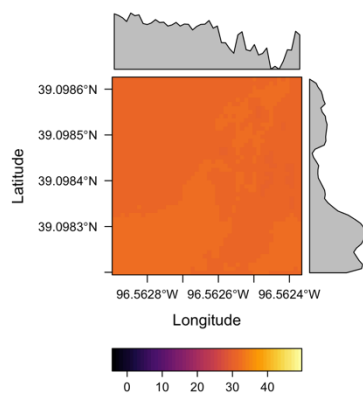


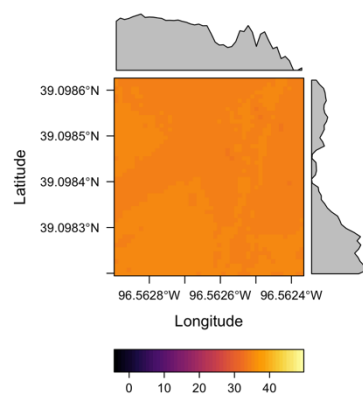
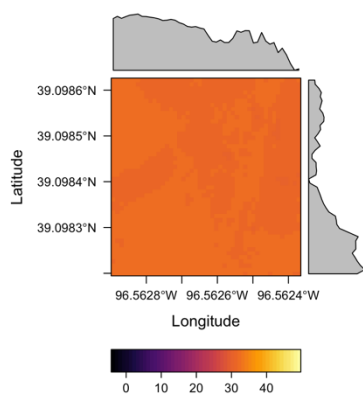
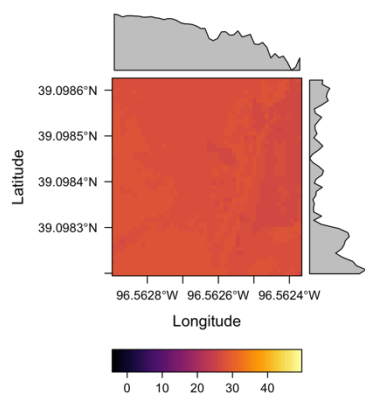
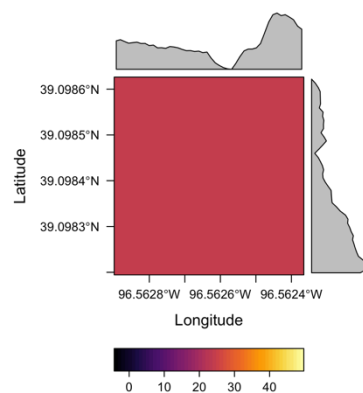
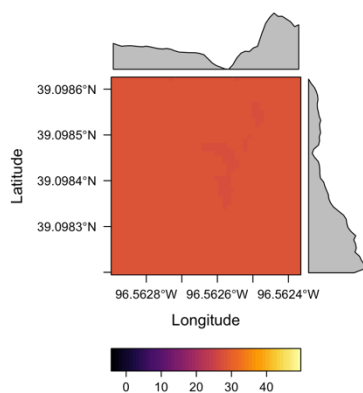
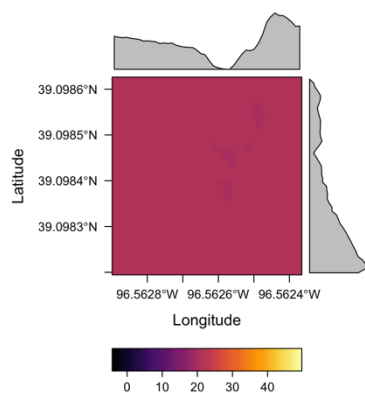
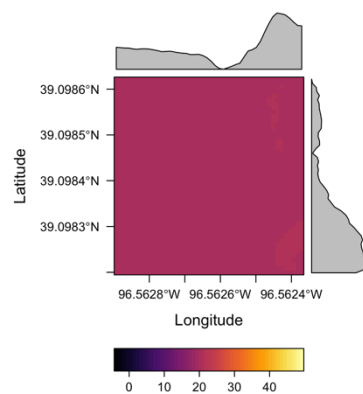
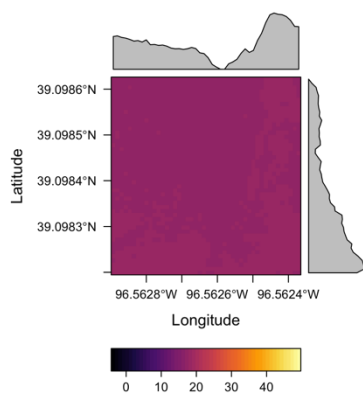
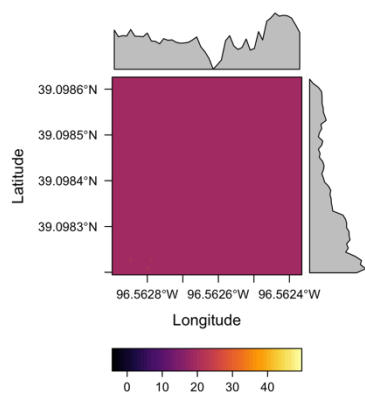


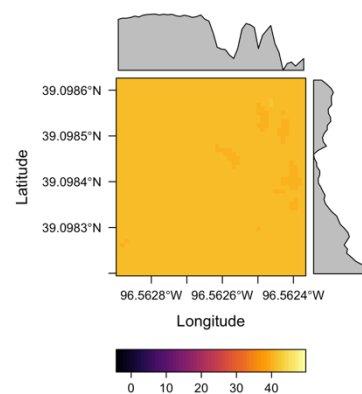
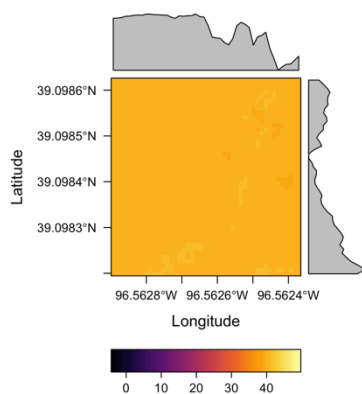
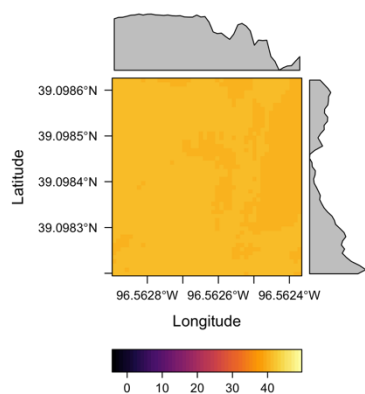
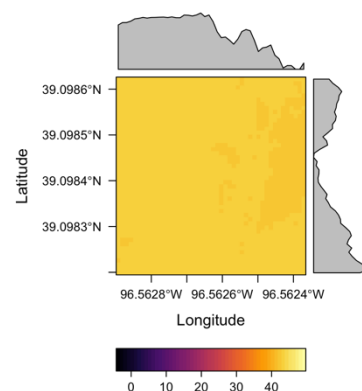
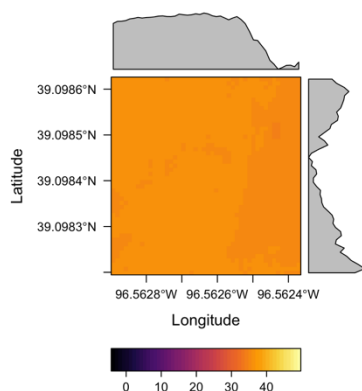
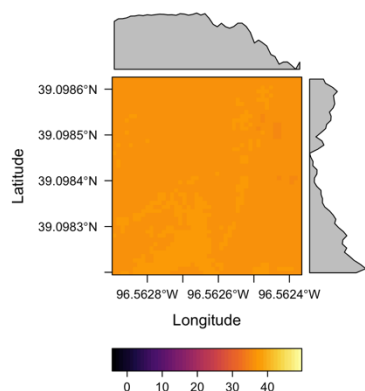
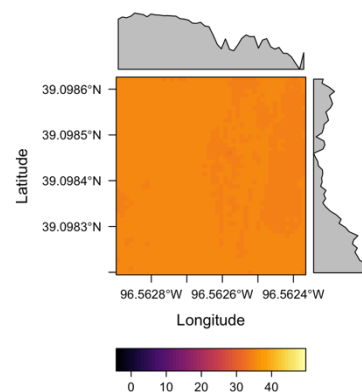
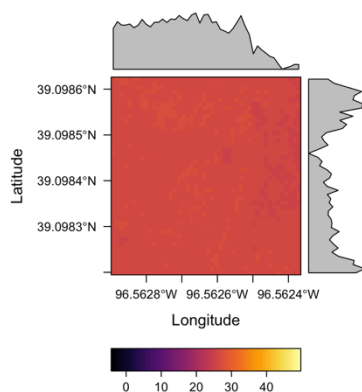
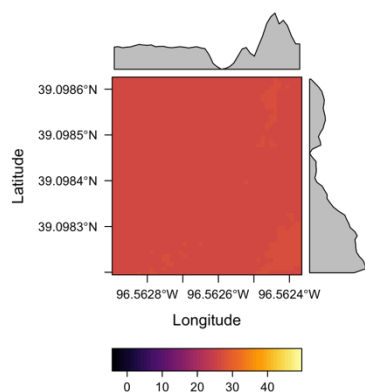


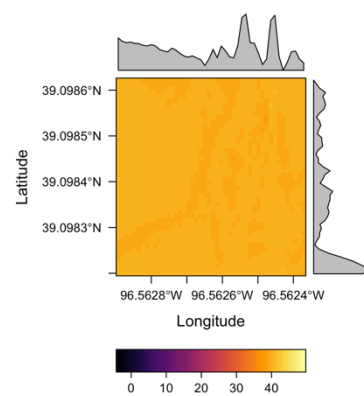
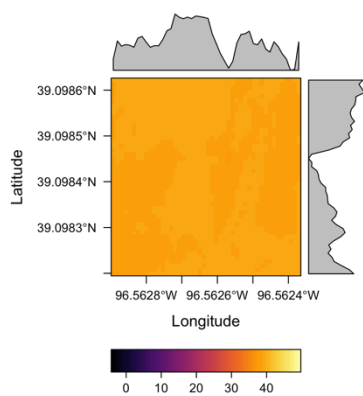
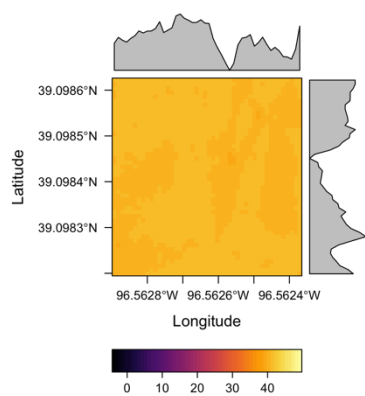
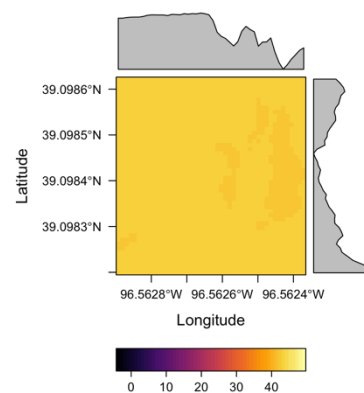
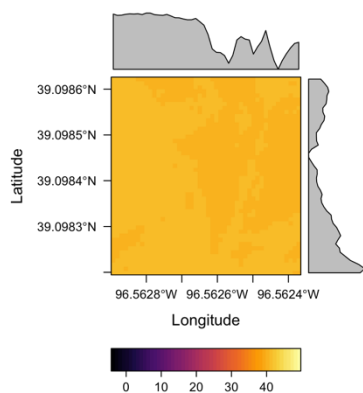
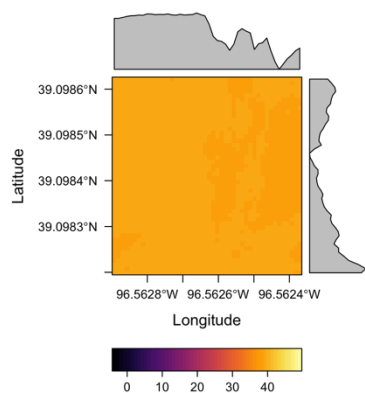
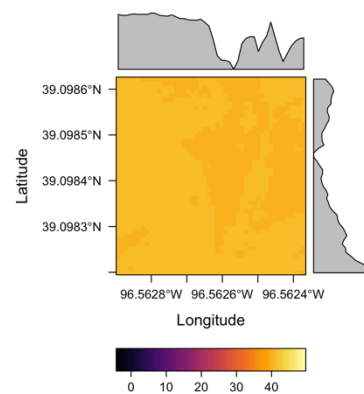
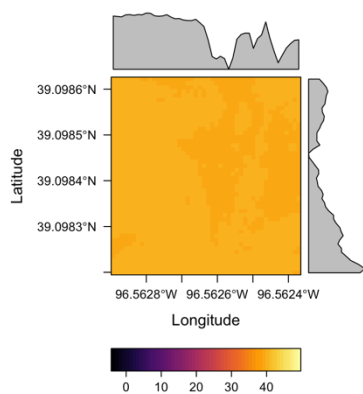
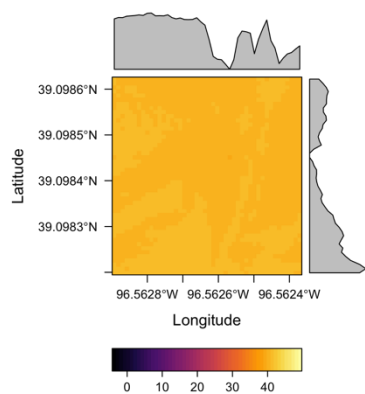


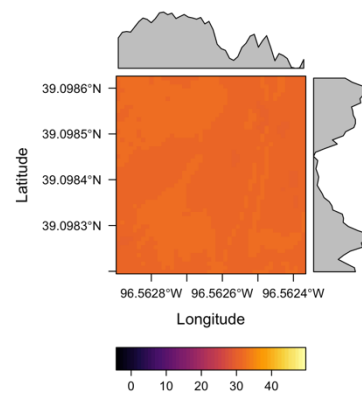
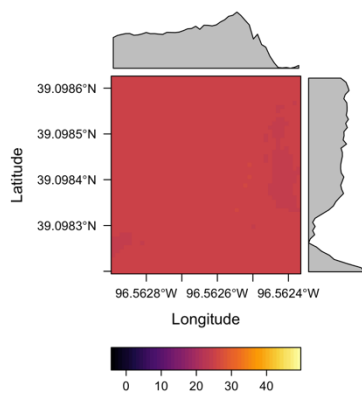
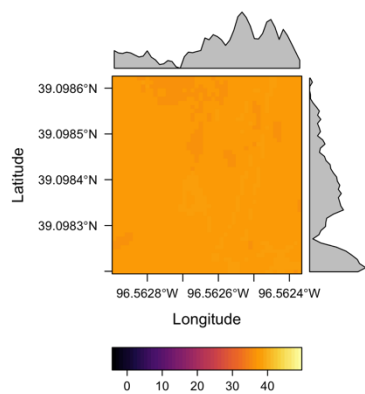
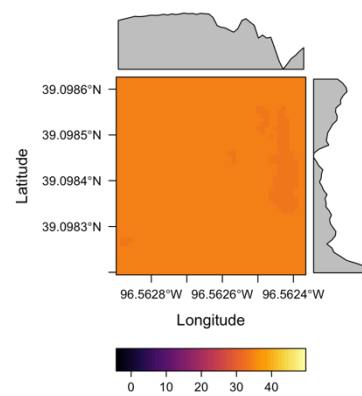
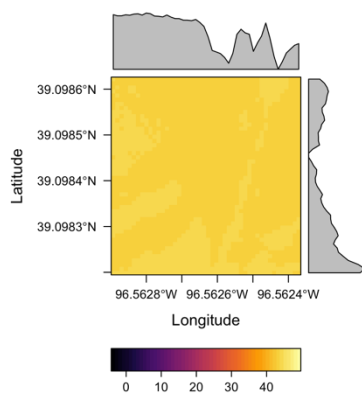
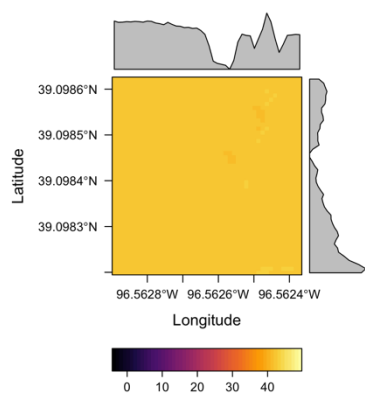
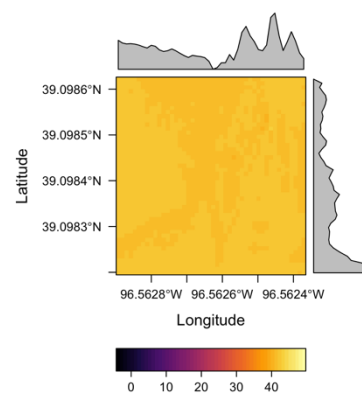
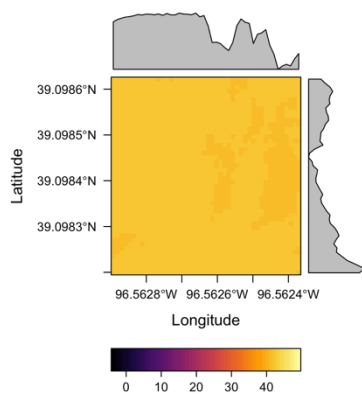
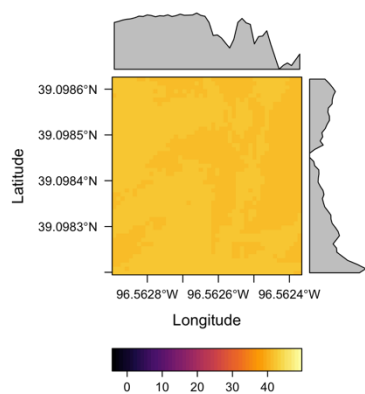


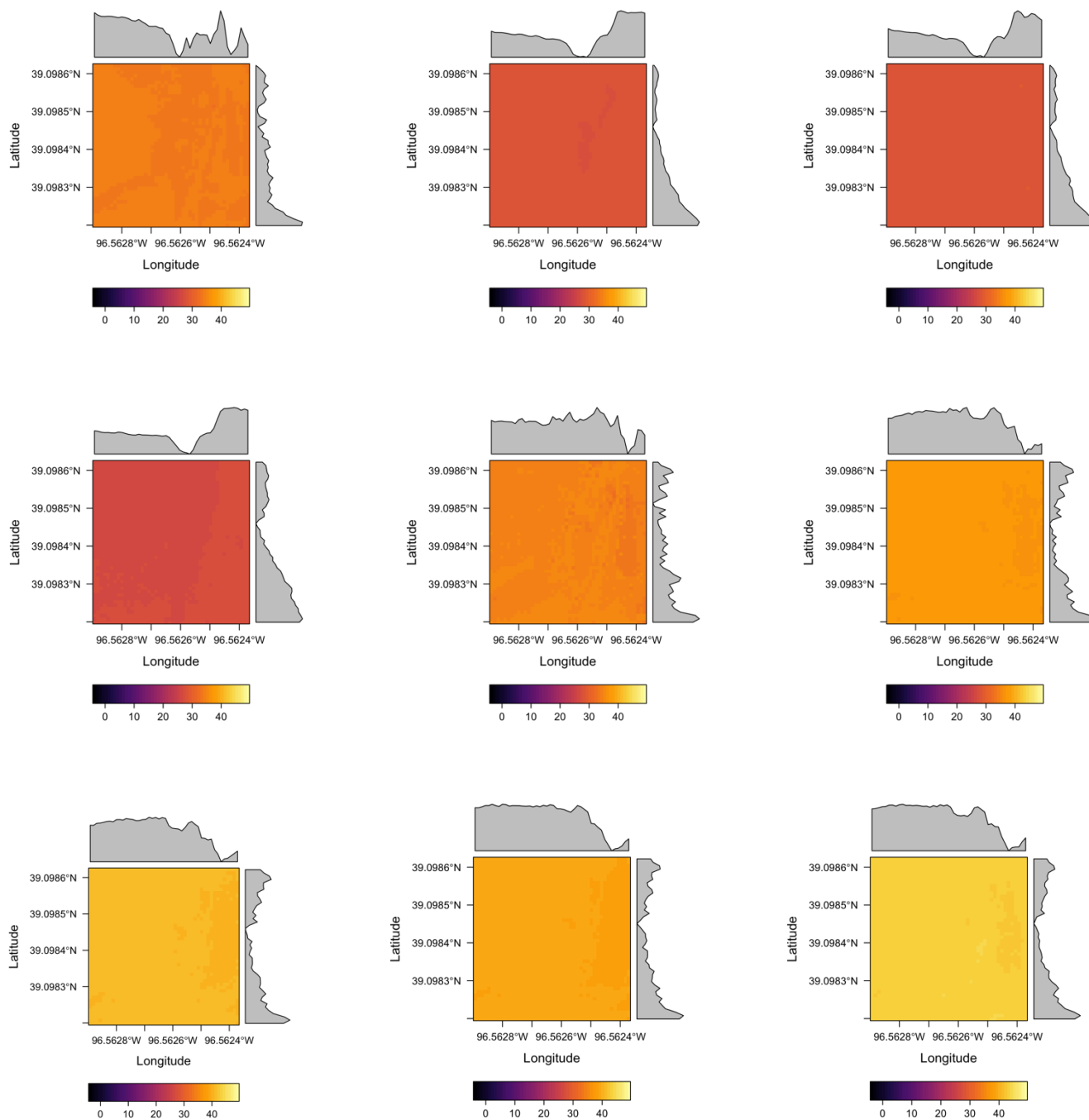


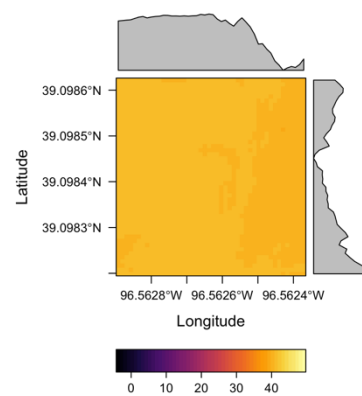
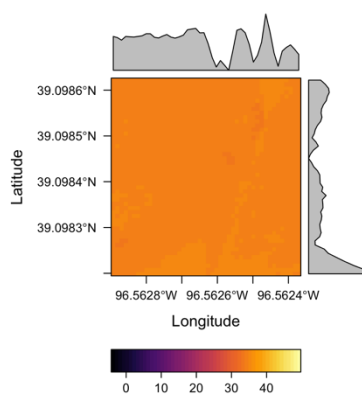
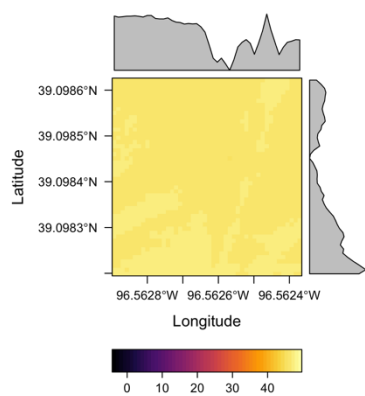
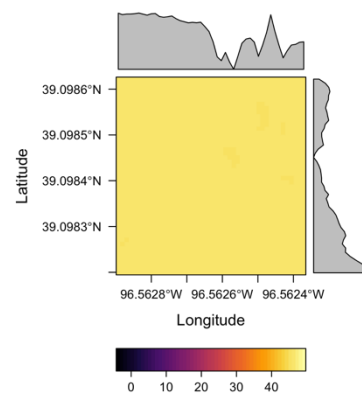
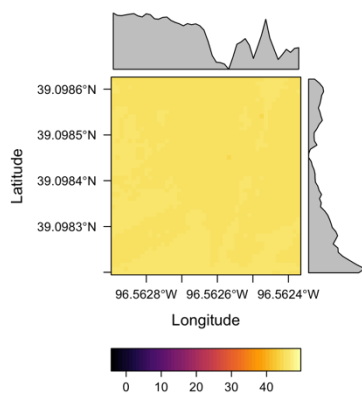
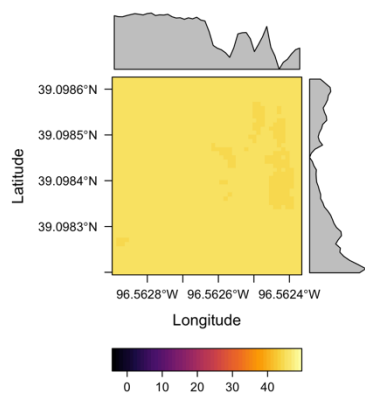
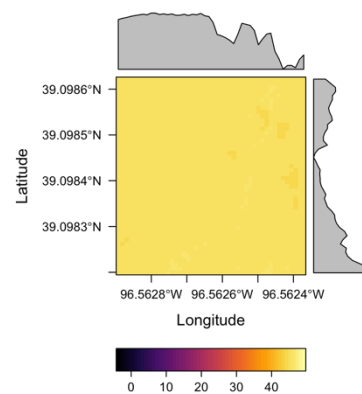
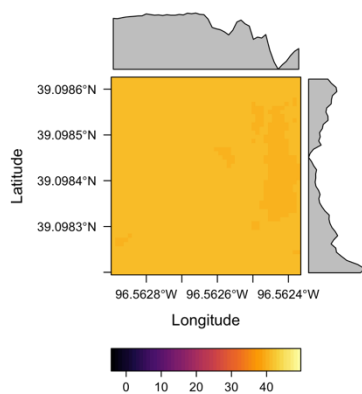
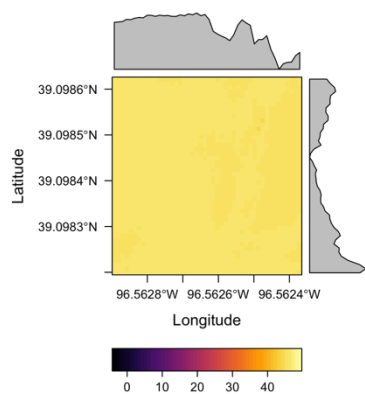


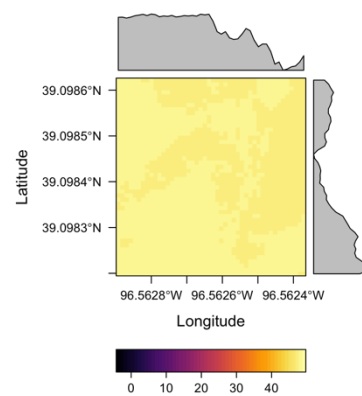
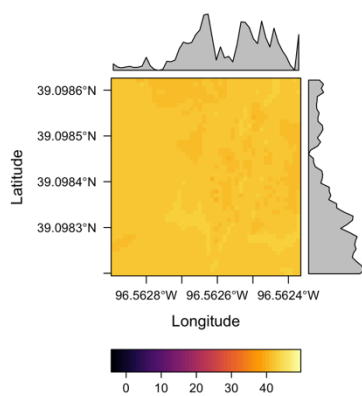
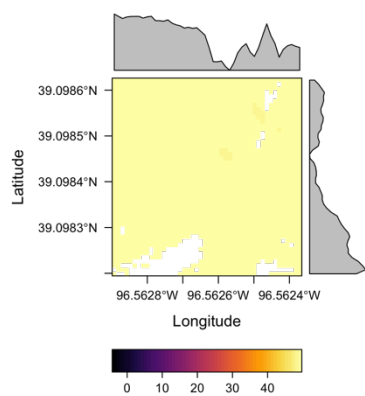
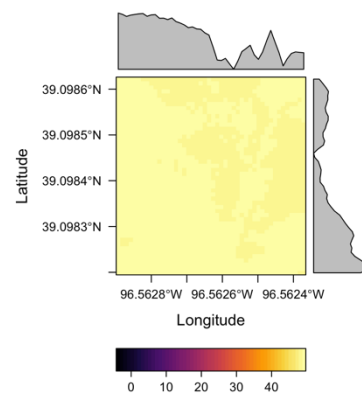
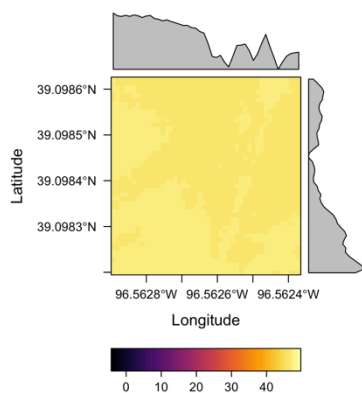
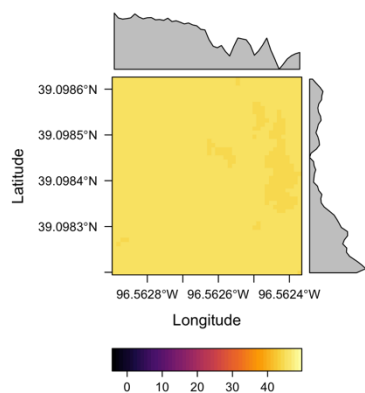
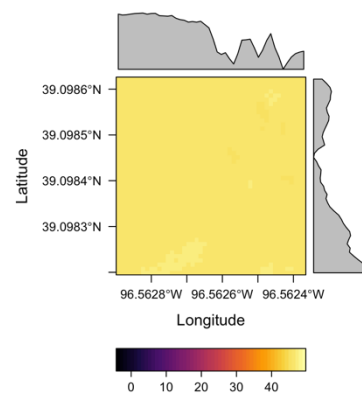
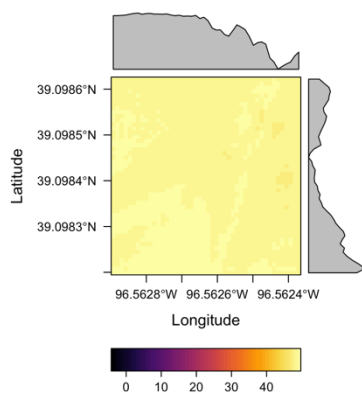
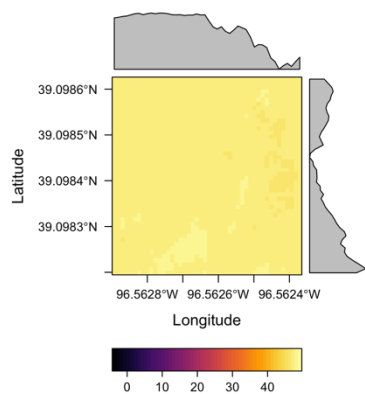


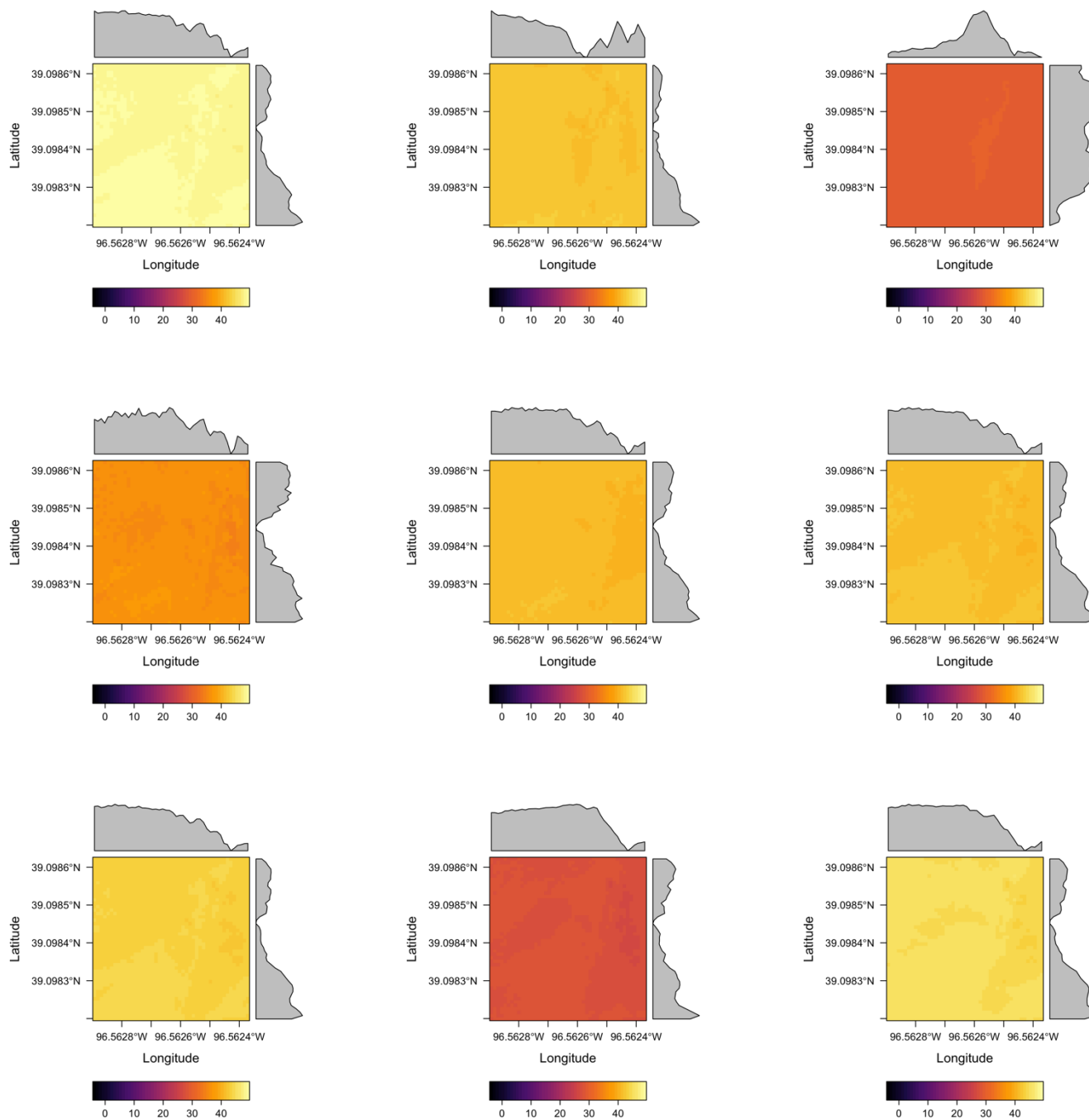


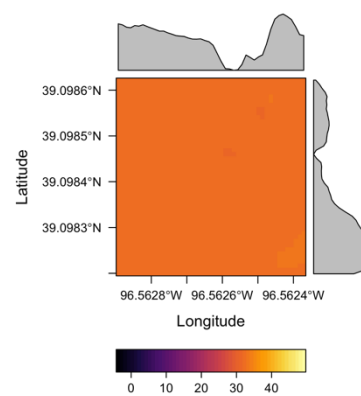
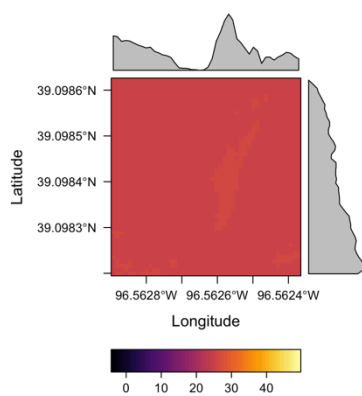
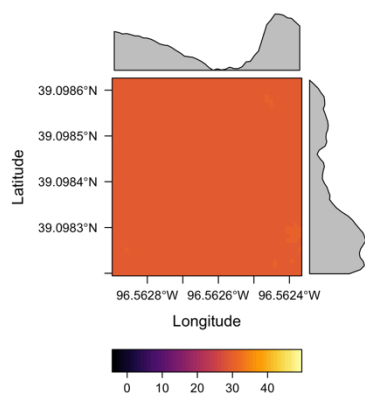
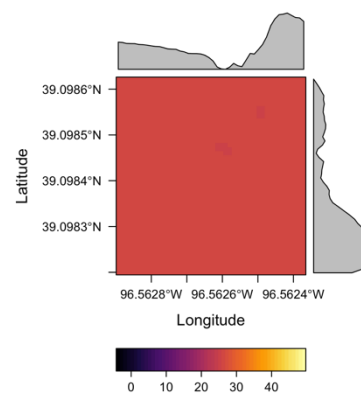
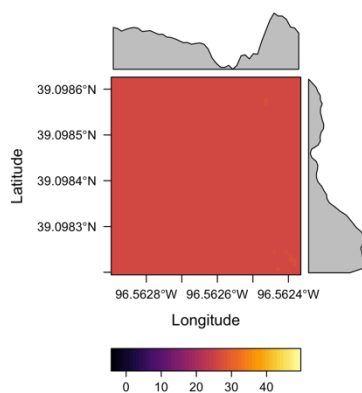
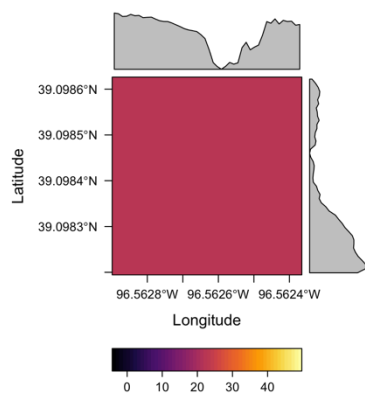
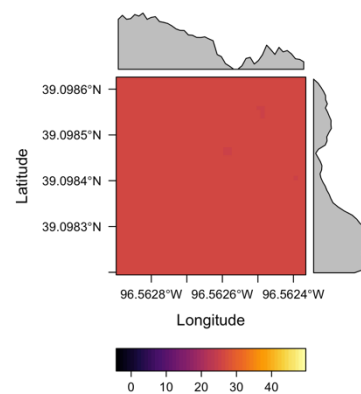
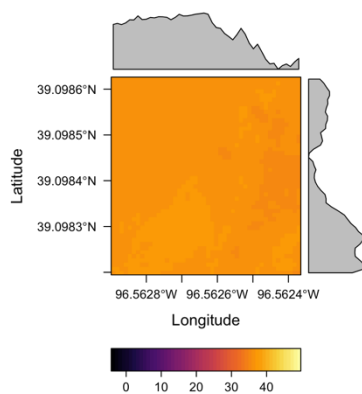
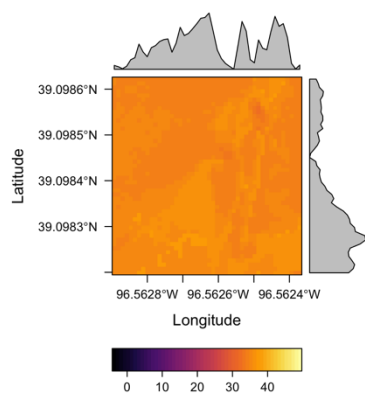


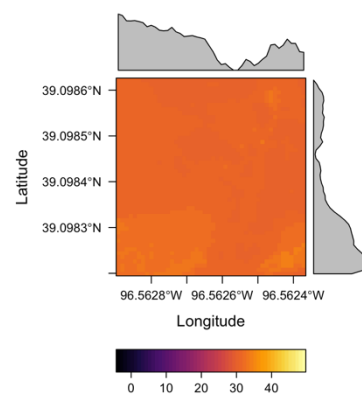
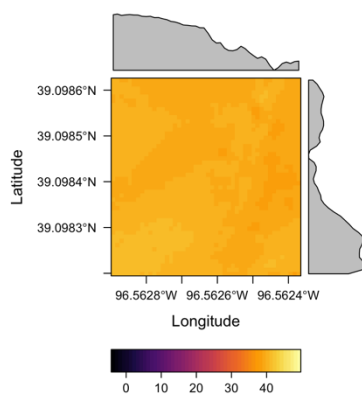
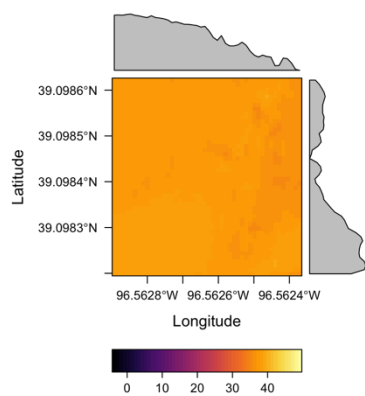
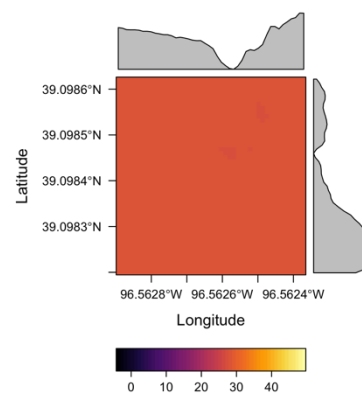
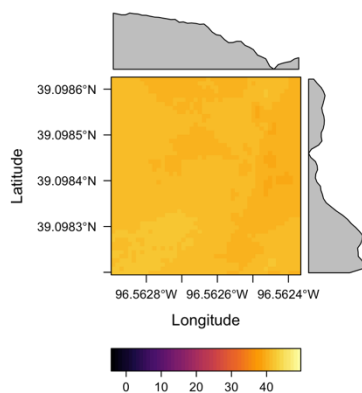
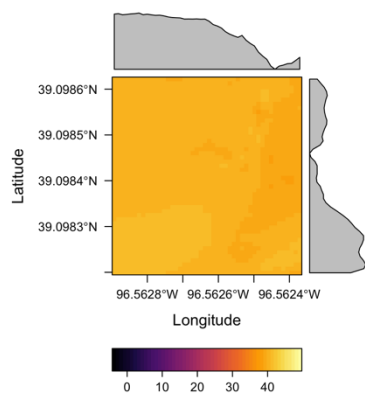
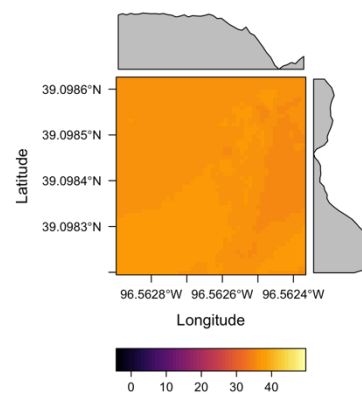
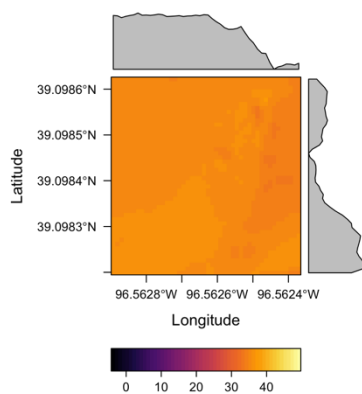
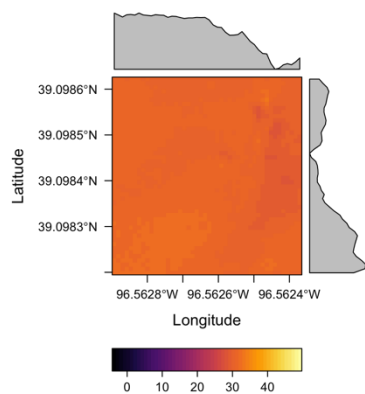


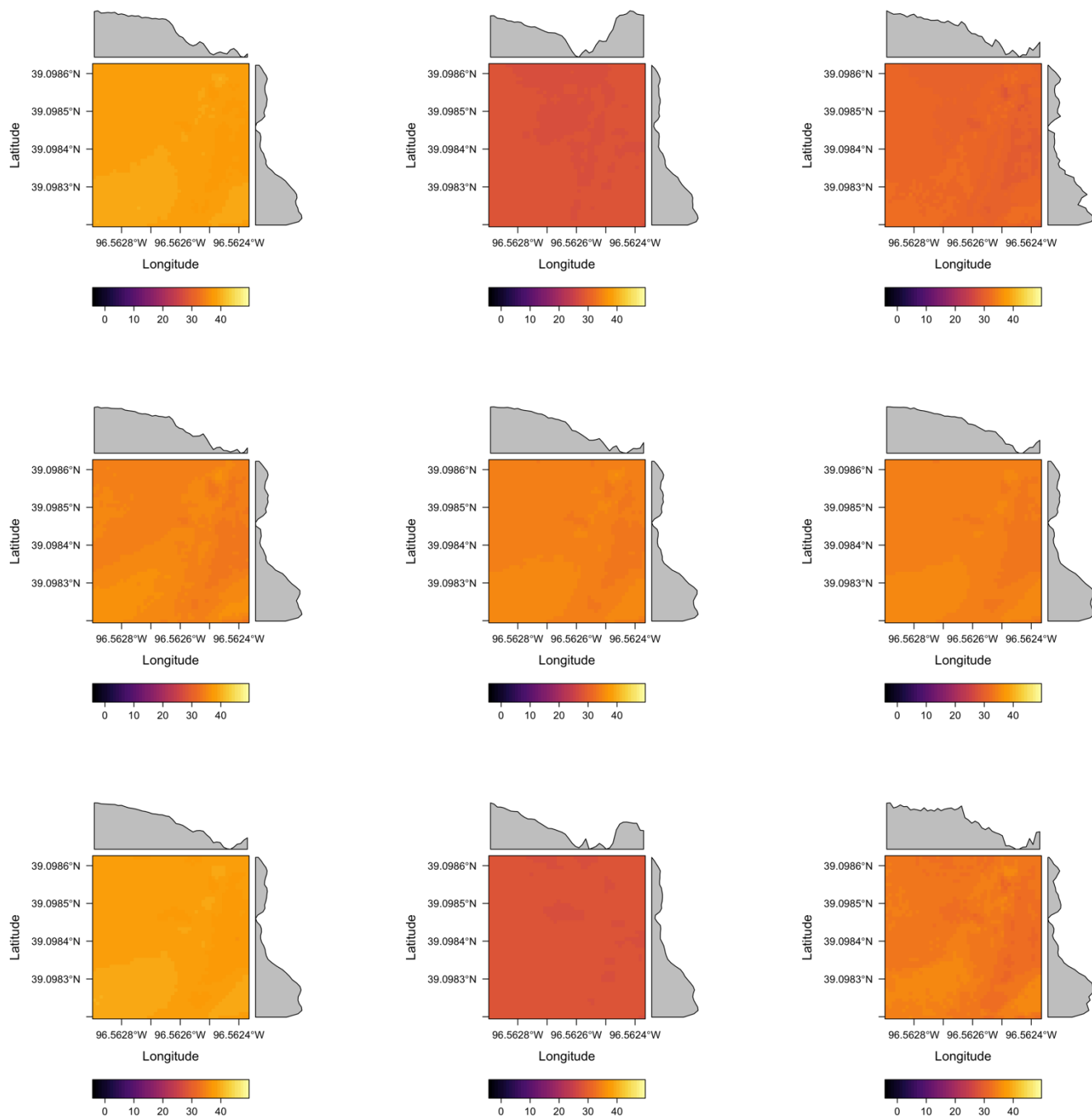


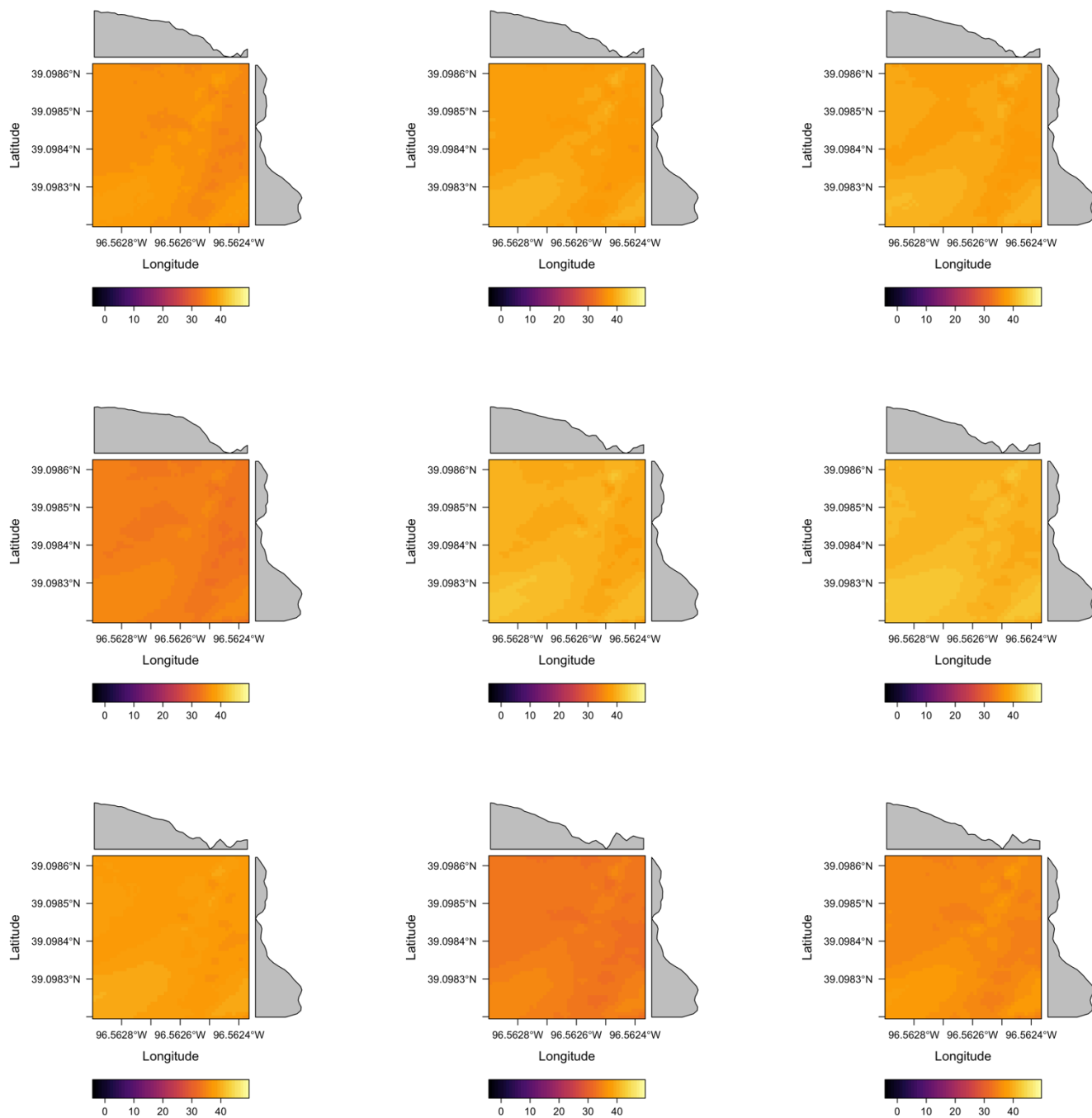


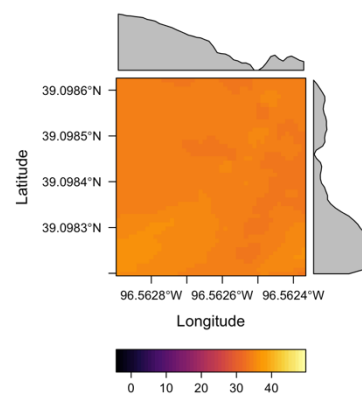
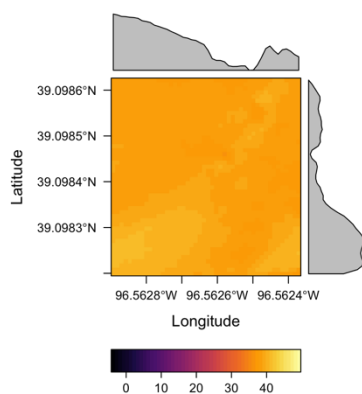
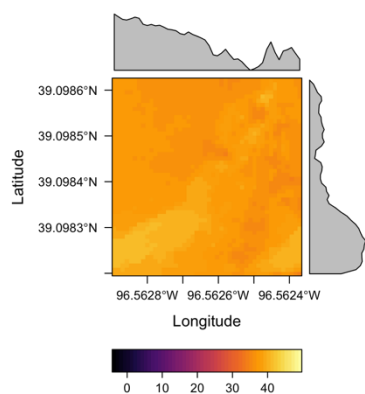
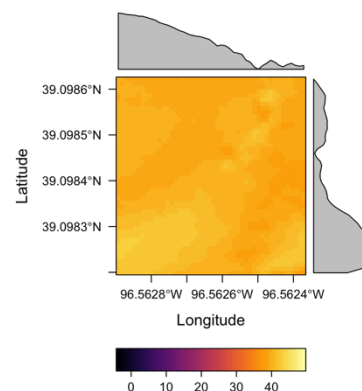
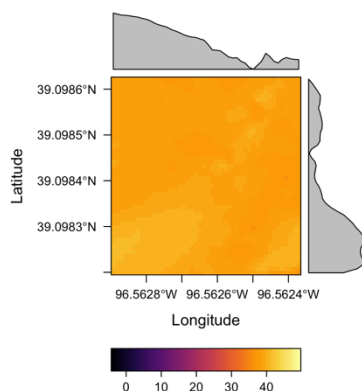
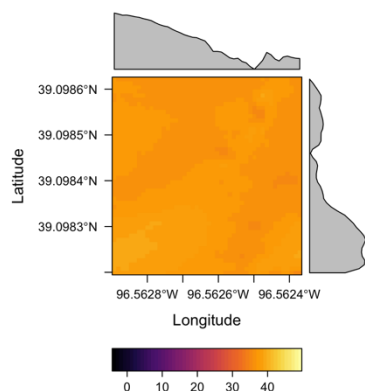
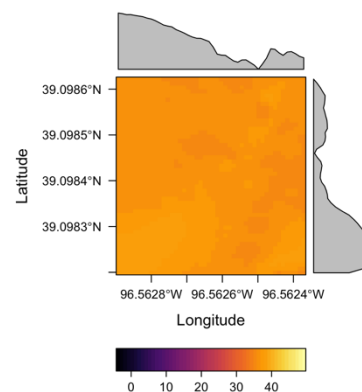
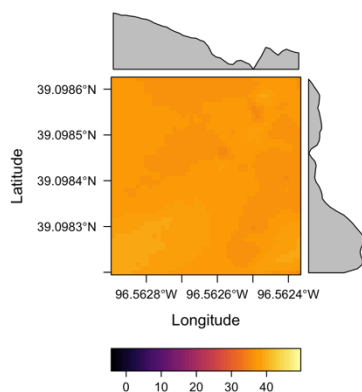
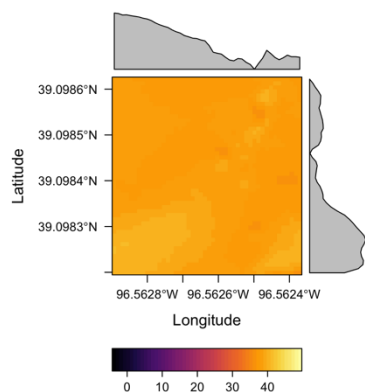


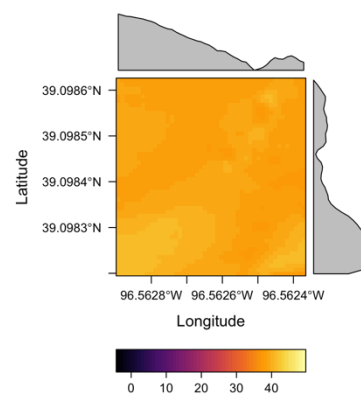
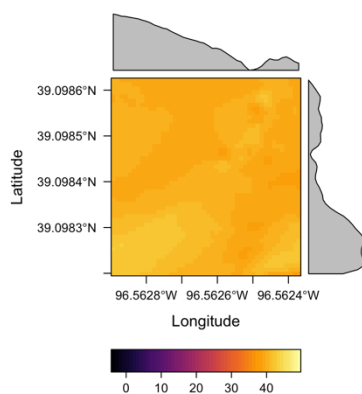
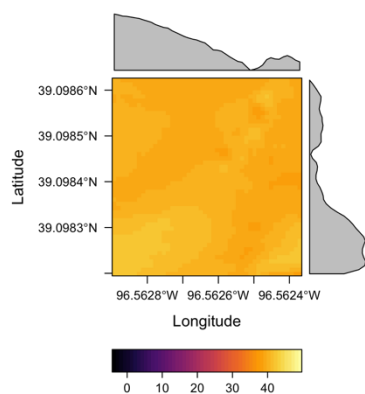
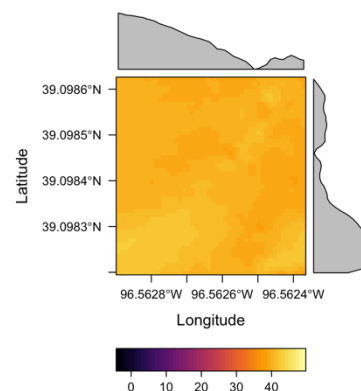
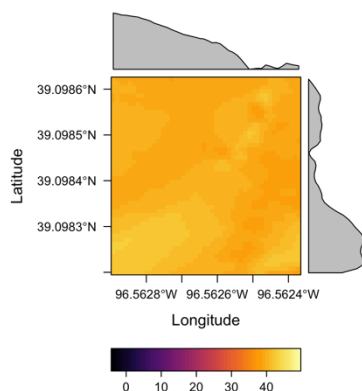
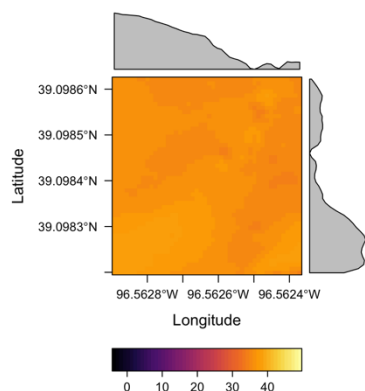
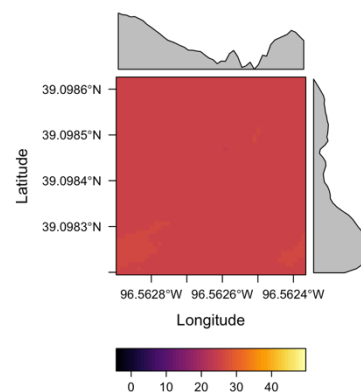
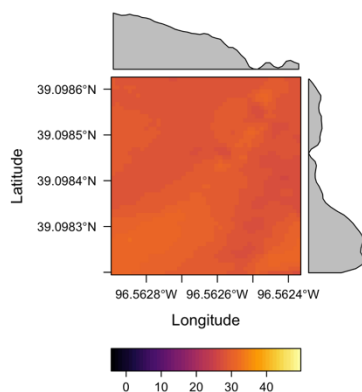
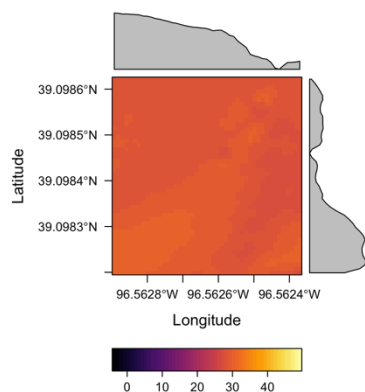


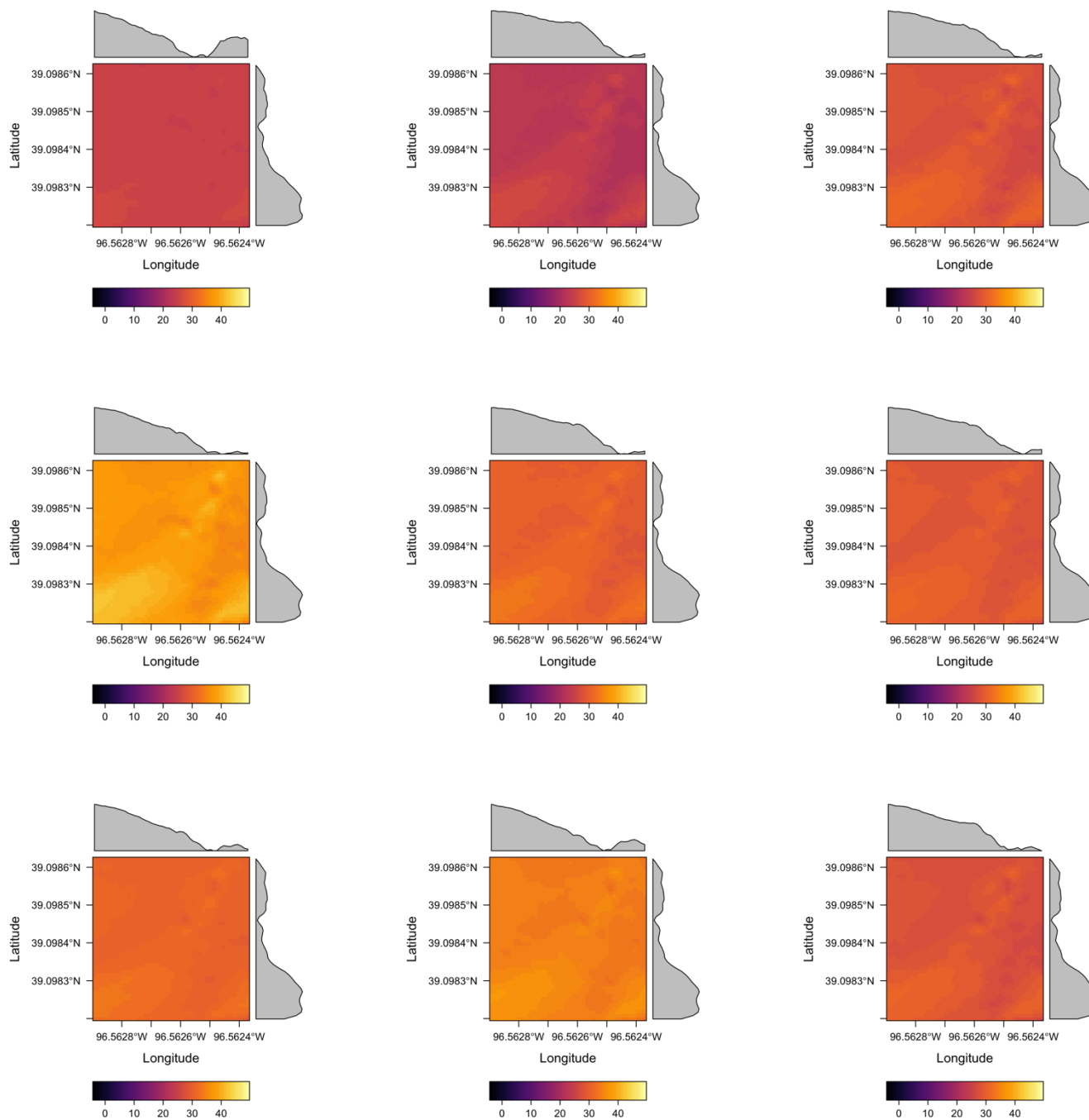


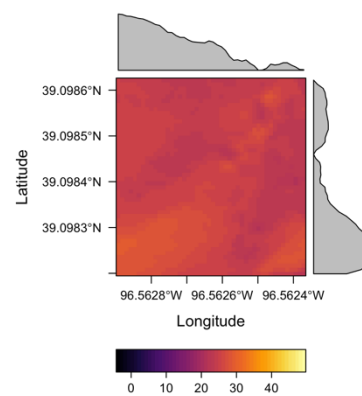
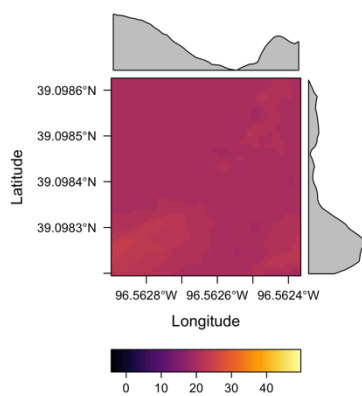
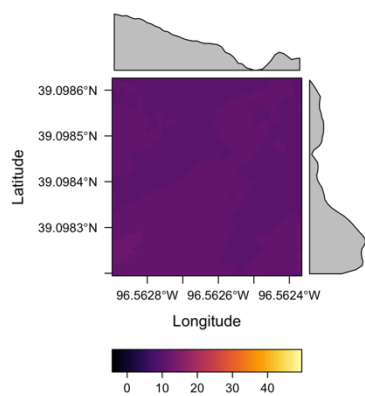
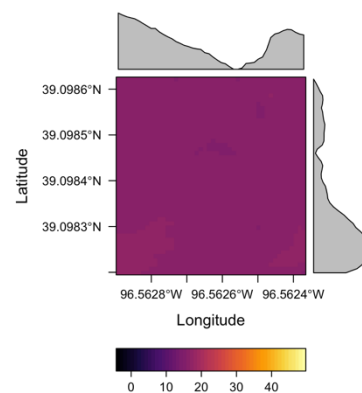
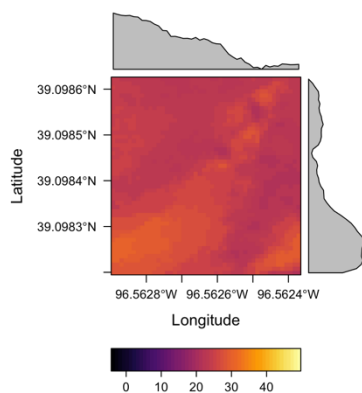
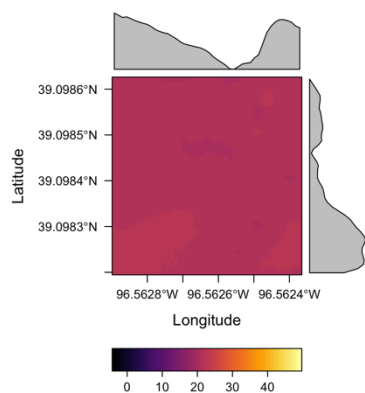
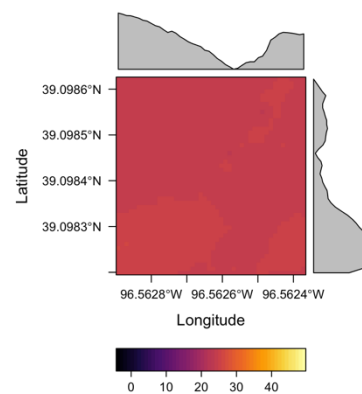
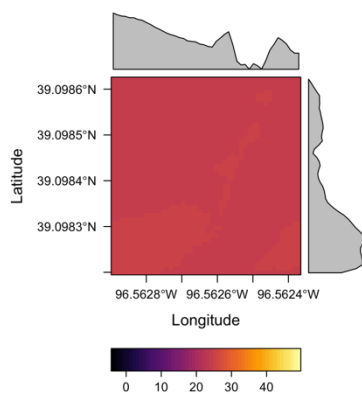
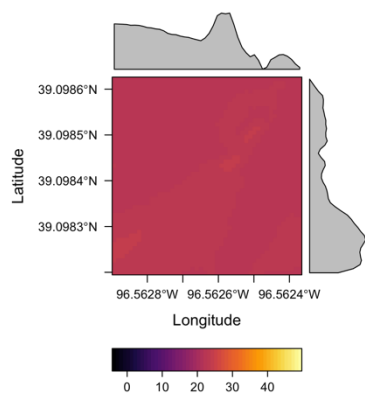


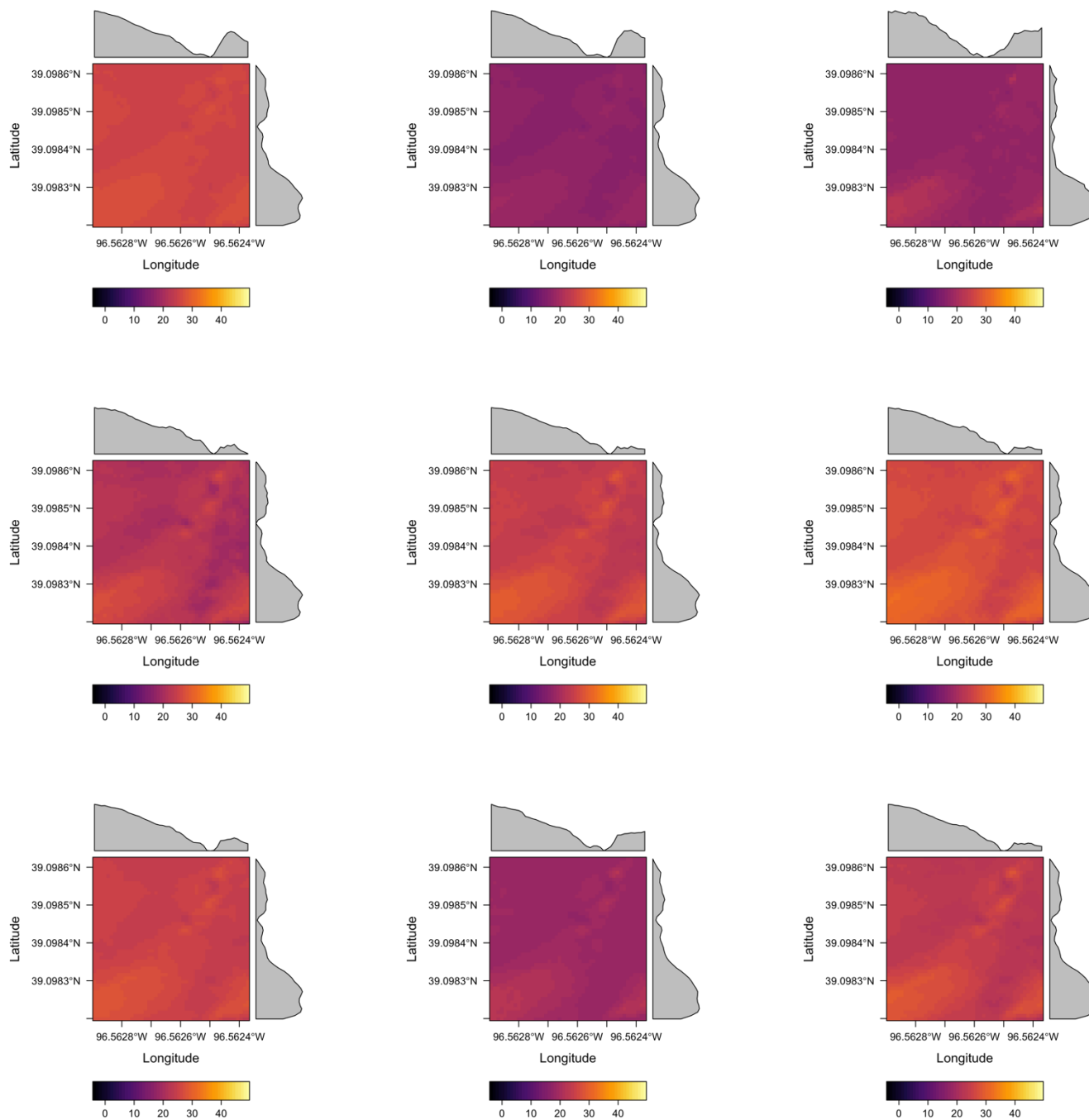


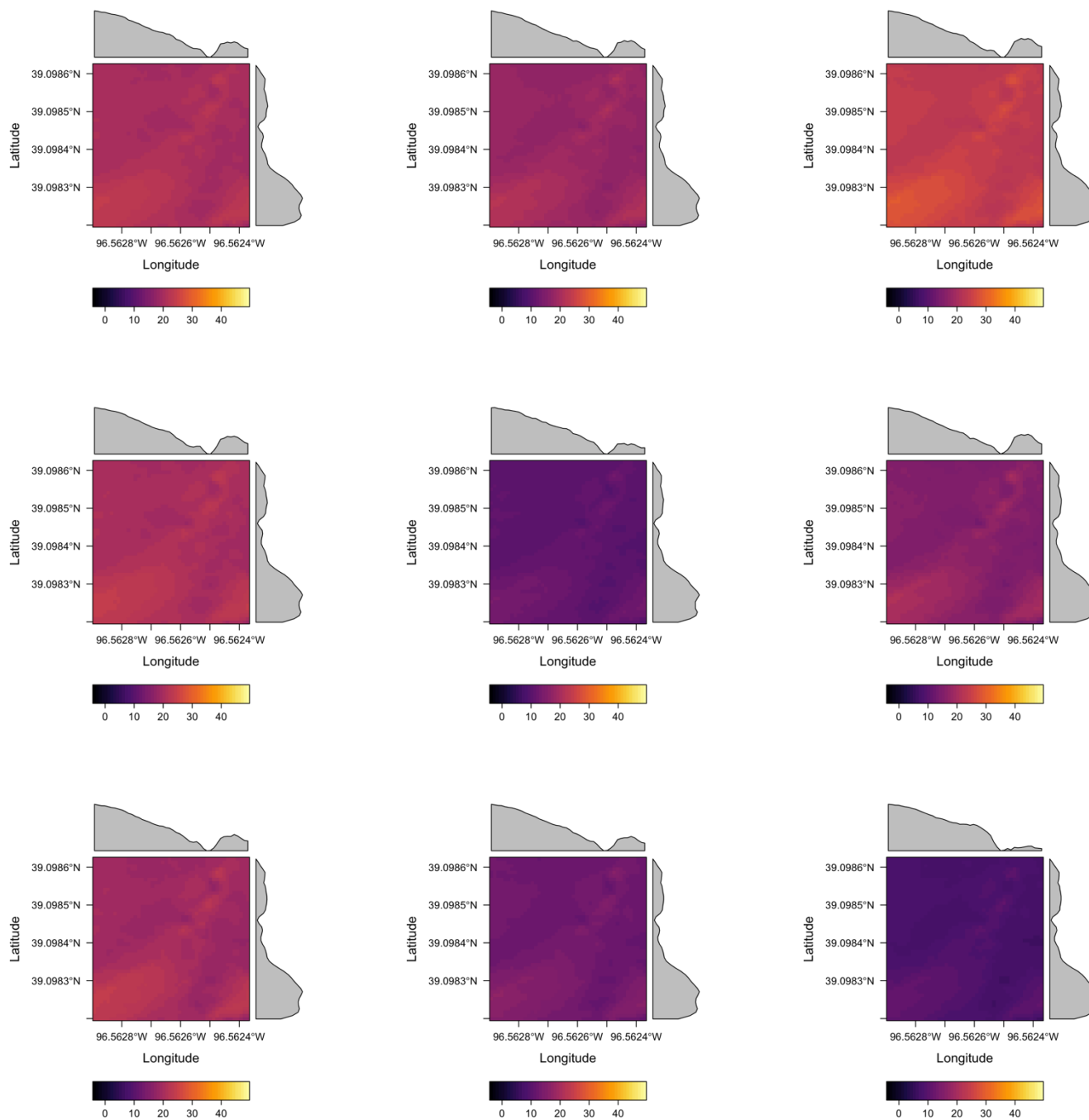




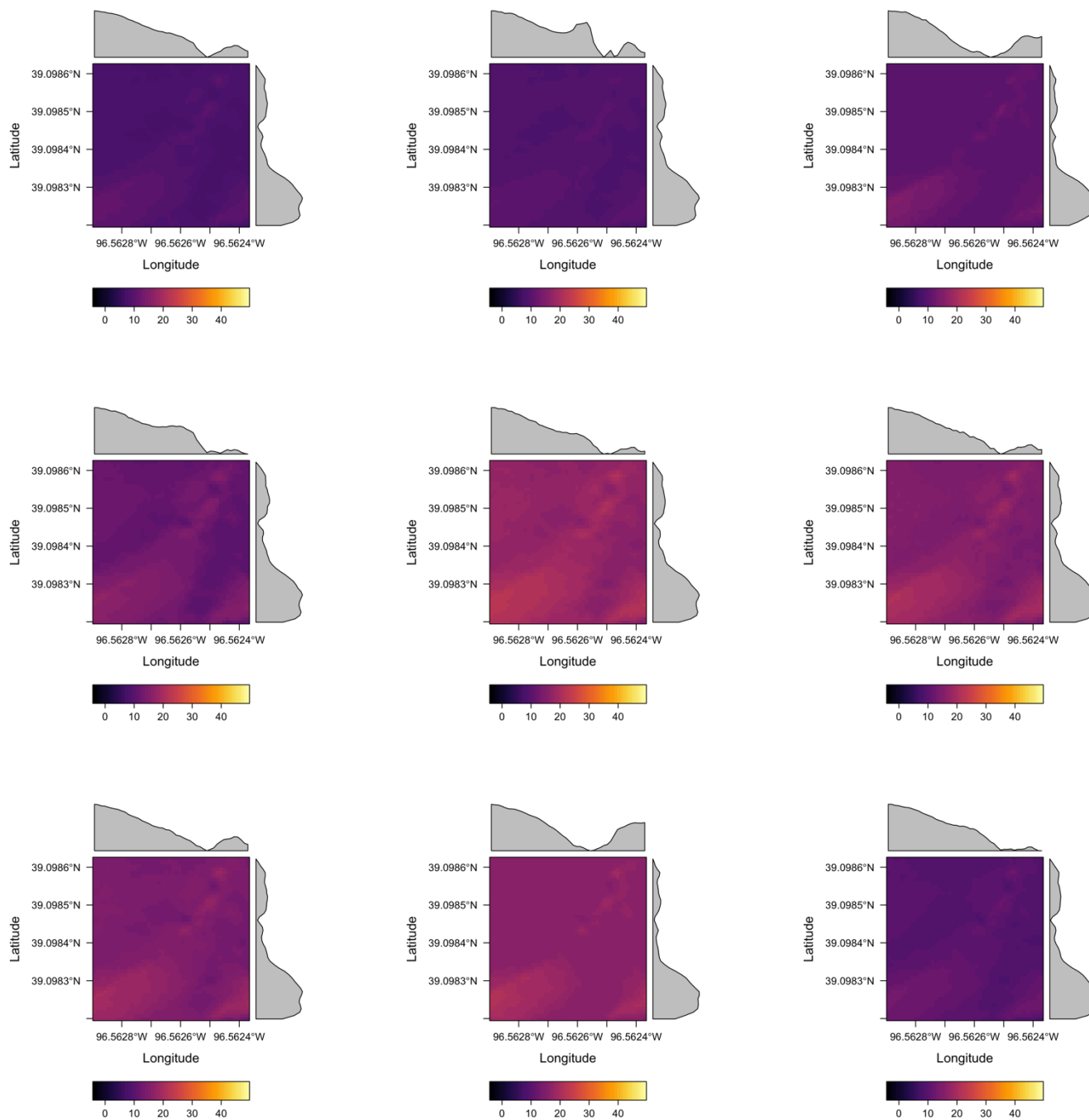


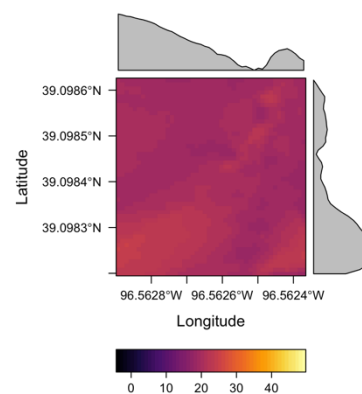
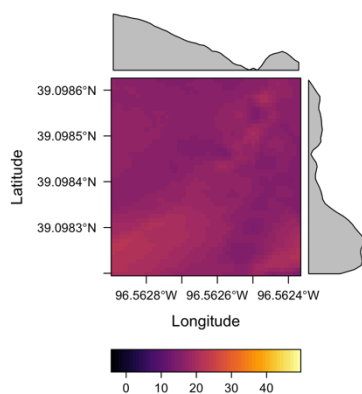
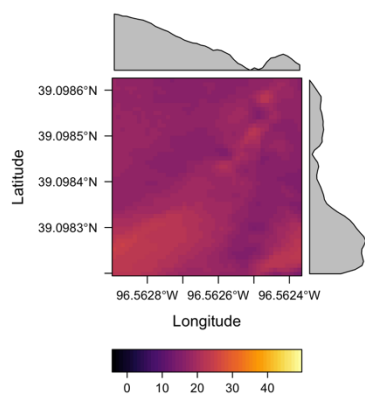
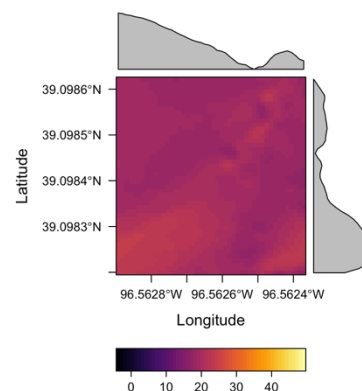
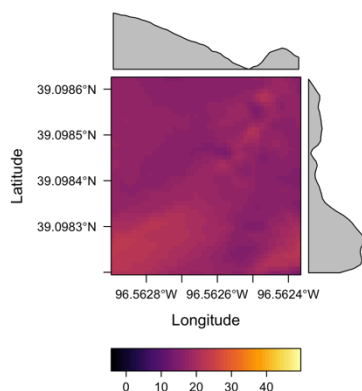
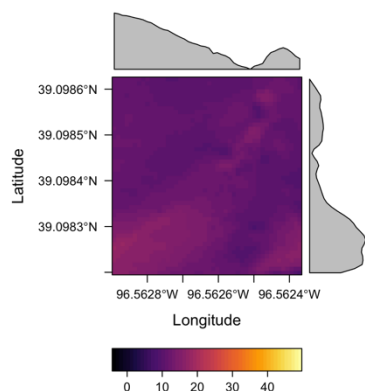
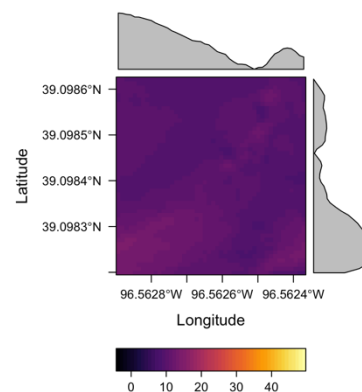
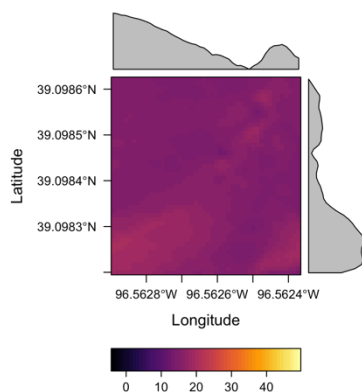
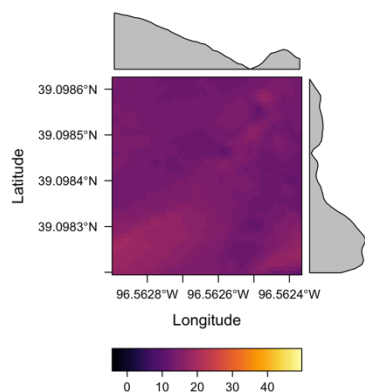


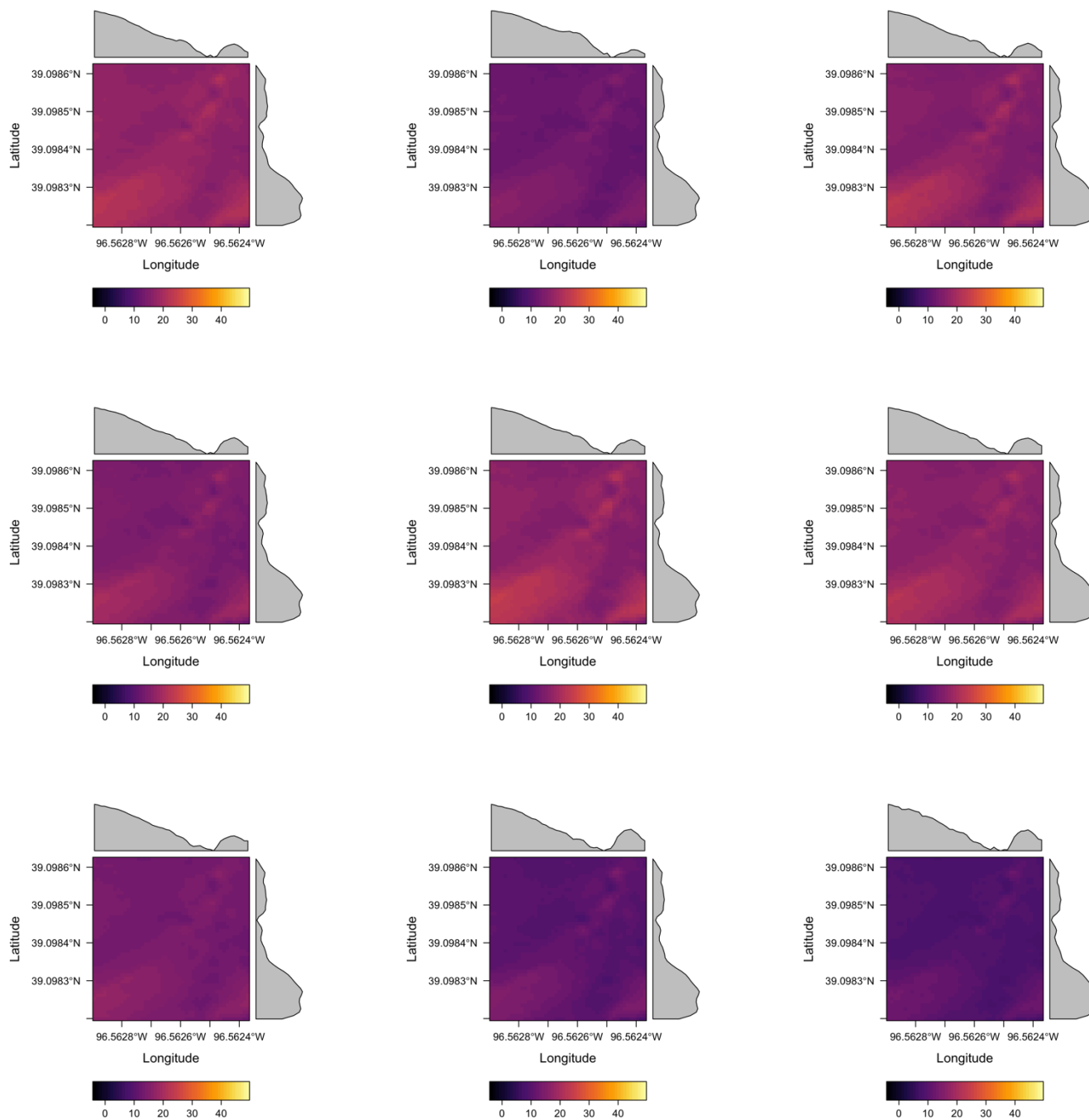


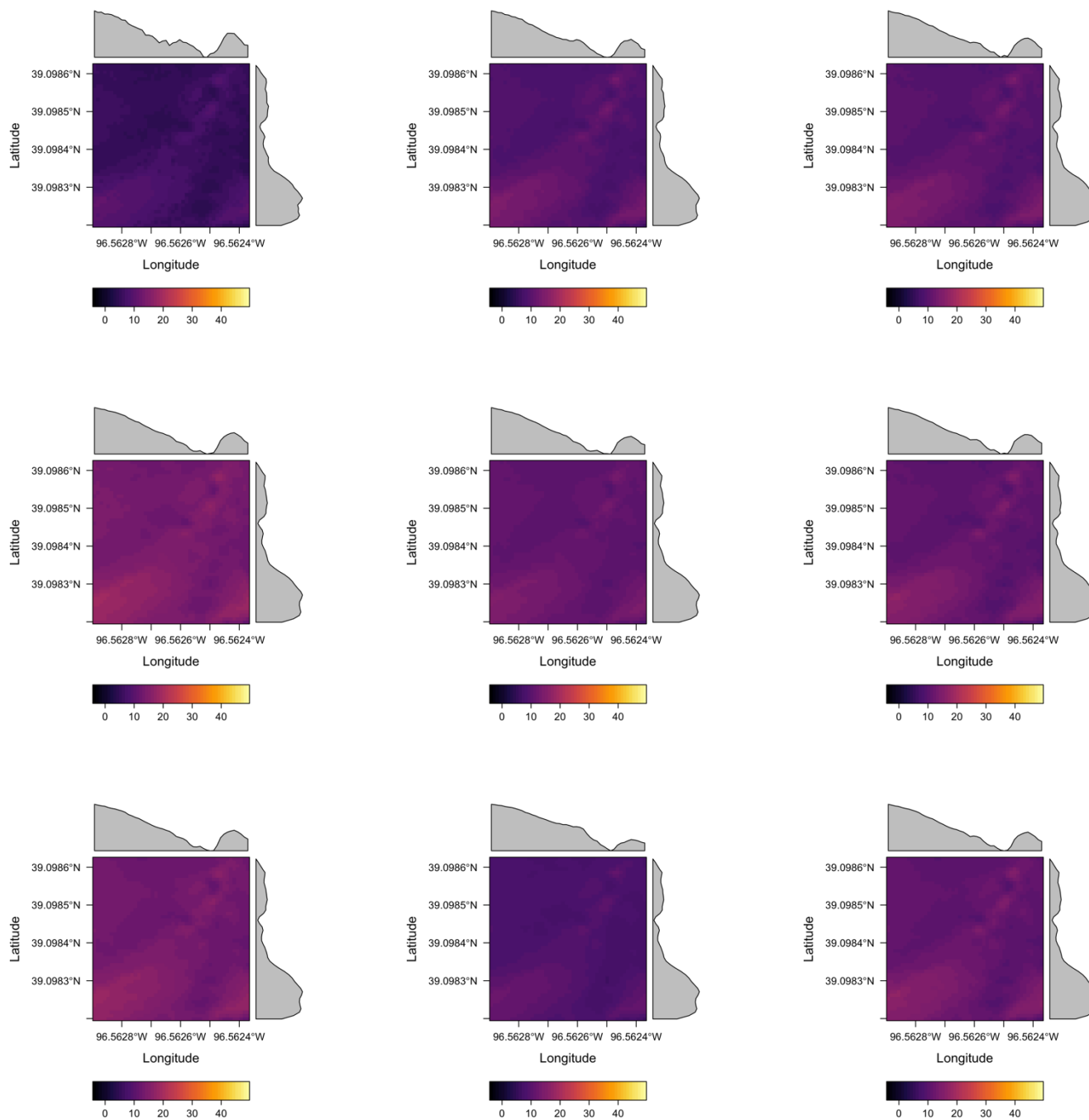


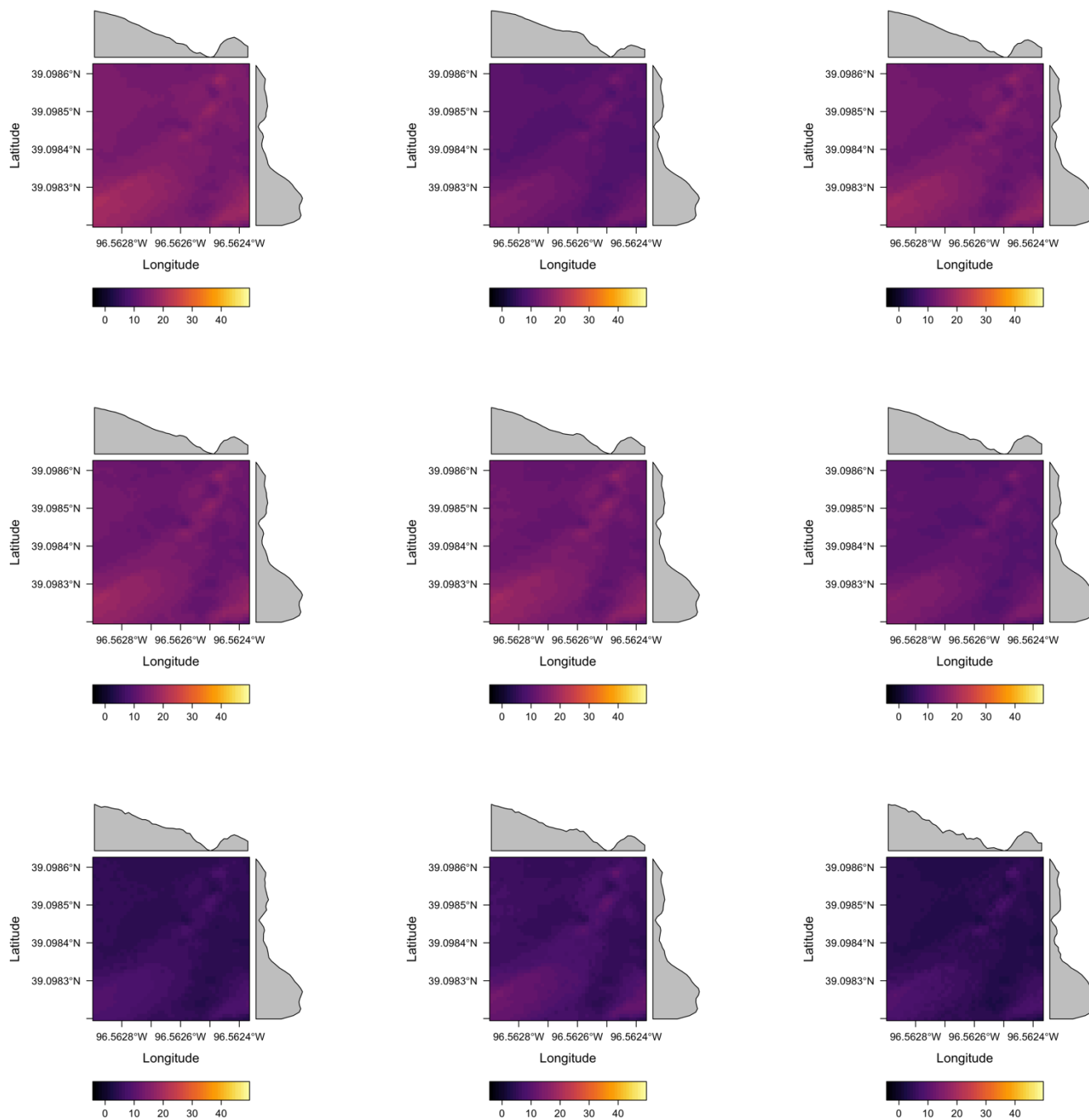


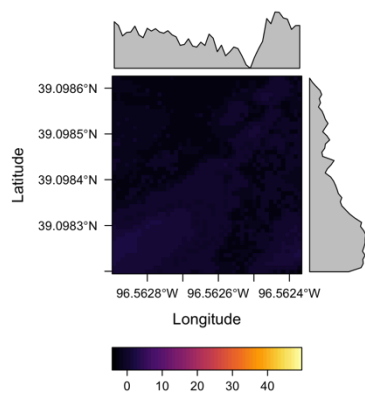
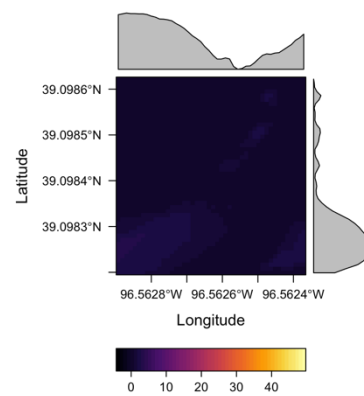
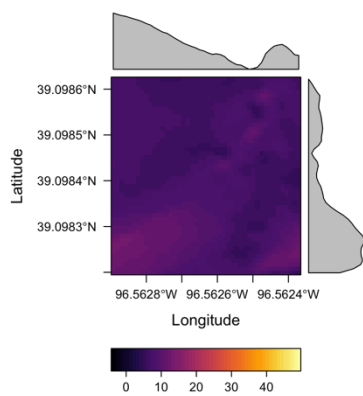
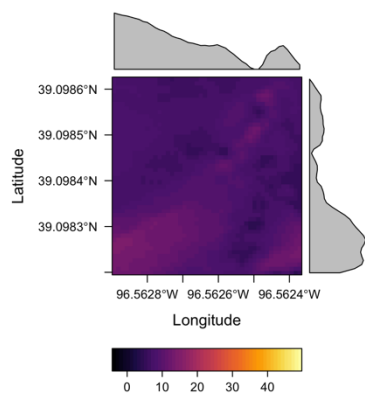
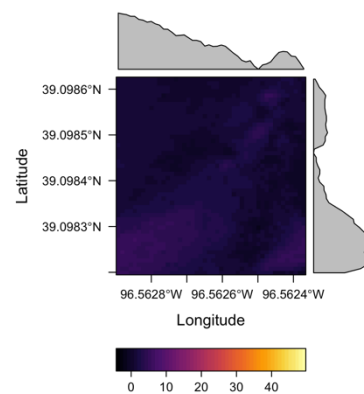
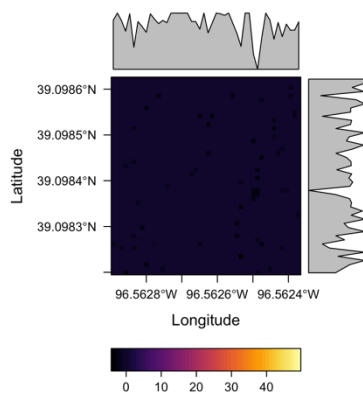
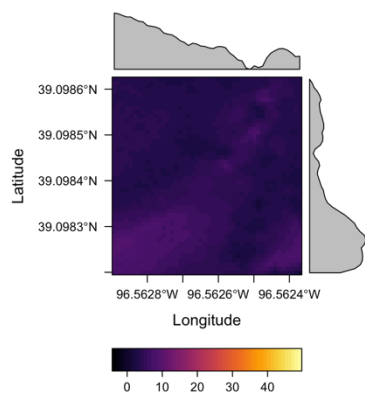












B. List of Scientific/Technical Publications

Carter A. *Using unoccupied aerial vehicles to model thermal microclimates for temperature-dependent traits. Invited symposium presentation in 'Remote Sensing in Ecology and Conservation: five years on,' Zoological Society of London, 19-20 November 2020. (Conference re-scheduled to 2022 due to COVID-19)*

Carter A., N.J. Mitchell, M.R. Kearney, S. Hartley & N.J. Nelson. Modeling impacts of climate change on 'invisible' thermal traits. Invited symposium presentation in 'Temperature, moisture and environmental change: conservation physiology of lepidosaurs in an age of climate disruption,' 9th World Congress of Herpetology, 5-10 Jan 2020.

Carter A. & F.J. Janzen. Using unoccupied aerial systems to model spatially-mediated heterogeneity in 3D thermal landscapes. Ecological Society of America Annual Meeting, 12-16 Aug 2019. Invited symposium presentation.

Carter A. Modeling microclimate conditions across eco-climatic domains. Ecological Society of America Annual Meeting, 12-16 Aug 2019. Invited 'Inspire' presentation.

C. Other Supporting Materials

C.1: R-code to build a raster stack to provide terrain inputs to the NicheMapR microclimate model

This code creates the elevation, slope, aspect, horizons stack for inputting into NicheMapR.

```
#### Load packages
library(horizon)
library(raster)
library(ncdf4)

### In future, get data from .zip files downloaded from NEON Data Portal via library(nneo)

## Create 24 horizon layers (~ 1 hr for 1mil cells)
hori.fun <- function(elev.rast) {
  require(svMisc)
  asp <- seq(0,345,15) # create vector of 24 angles to calculate horizon height
  hori <- stack()
  for(i in 1:length(asp)) {
    progress(i, progress.bar=T)
    Sys.sleep(0.01)
    if(i==101) {
      cat('Done\n')
    }
    x <- horizonSearch(elev.rast, azimuth = asp[i], maxDist = 1000, degrees = T, ll = F)
    hori <- stack(hori, x)
  }
  return(hori)
}

## Create the stack for KONZ
elev1 <- raster('NEON_D06_KONZ_DP3_710000_4330000_DTM.tif')
slope1 <- raster('NEON_D06_KONZ_DP3_710000_4330000_slope.tif')
aspect1 <- raster('NEON_D06_KONZ_DP3_710000_4330000_aspect.tif')
hori1 <- hori.fun(elev1)

## Stack rasters
elevslophasphori <- stack(elev1,slope1,aspect1,hori1)
elevslophasphori.wgs84 <- projectRaster(from=elevslophasphori, crs='+proj=longlat +ellps=WGS84 +datum=WGS84
+no_defs +towgs84=0,0,0') # project coordinates from utm (m) to degrees

## Write raster stack
writeRaster(elevslophasphori.wgs84, filename='elevslophasphori_D06_KONZ_13sec.nc', format='CDF', overwrite=T,
NAflag=-9999, xname='lon', yname='lat') # Does not preserve names

# end
```

C.2: R-code to run a custom *micro_site()* function over a grid of cells.

This code extracts cell-by-cell terrain data from a 27-layer .nc file as inputs then runs the modified *micro_usa()* function (*micro_site()*) for each cell and writes each output to a delimited text file.

```
run_micro <- function(X, lon.name='longitude', lat.name='latitude',
                     spatial='spatial', out.dir='out') {
  library(ncdf4)
  library(raster)
  library(NicheMapR)

  xy <- paste0(spatial, '/elev/', X)

  n.sites <- dim(ncdf4::ncvar_get(nc_open(xy),lon.name)) *
    dim(ncdf4::ncvar_get(nc_open(xy),lat.name)) # get number of sites for loop
  terrain.nc <- raster::brick(xy) # create raster object from netcdf

  write.table(data.frame(), paste0(out.dir,'/metouts.csv'), row.names=F, sep=',', quote=F) # write empty .csv to dump
  output
  write.table(data.frame(), paste0(out.dir,'/shadmets.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/soils.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/shadsoils.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/soilmoists.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/soilpots.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/shadpots.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/humids.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/shadhumids.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/plants.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/shadplants.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/drlams.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/drrlams.csv'), row.names=F, sep=',', quote=F)
  write.table(data.frame(), paste0(out.dir,'/srlams.csv'), row.names=F, sep=',', quote=F)

  source('micro_site_usa.R') # Runs the microclimate model

  for (i in 1:n.sites) {

    cat('Testing site #', i, '\n')

    xy.site <- xyFromCell(terrain.nc, cell=i) # get lon/lat for single cell
    xy.terrain <- extract(terrain.nc, xy.site) # extract terrain data for single cell

    if (anyNA(xy.terrain)==T) { next } # skip cells with incomplete data

    micro <- micro_site(loc=xy.site, elev=round(xy.terrain[1],2), slope=round(xy.terrain[2],2),
                       aspect=round(xy.terrain[3],2), hori=round(xy.terrain[4:27],0), spatial='spatial')

    lon <- xy.site[1]
    lons <- rep(lon, times=dim(micro$metout)[1])
    lat <- xy.site[2]
    lats <- rep(lat, times=dim(micro$metout)[1])
    site <- i
    sites <- rep(i, times=dim(micro$metout)[1])

    metout <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$metout))
    shadmet <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$shadmet))
  }
}
```

```

soil <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$soil))
shadsoil <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$shadsoil))
soilmoist <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$soilmoist))
shadmoist <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$shadmoist))
soilpot <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$soilpot))
shadpot <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$shadpot))
humid <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$humid))
shadhumid <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$shadhumid))
plant <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$plant))
shadplant <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$shadplant))
drlam <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$drlam))
drrlam <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$drrlam))
srlam <- as.data.frame(cbind(sitenum=sites, lon=lons, lat=lats, micro$srlam))

if (i == 1) {
  write.table(metout, paste0(out.dir, '/metouts.csv'), row.names=F, sep=',', quote=F)
  write.table(shadmet, paste0(out.dir, '/shadmets.csv'), row.names=F, sep=',', quote=F)
  write.table(soil, paste0(out.dir, '/soils.csv'), row.names=F, sep=',', quote=F)
  write.table(shadsoil, paste0(out.dir, '/shadsoils.csv'), row.names=F, sep=',', quote=F)
  write.table(soilmoist, paste0(out.dir, '/soilmoists.csv'), row.names=F, sep=',', quote=F)
  write.table(shadmoist, paste0(out.dir, '/shadmoists.csv'), row.names=F, sep=',', quote=F)
  write.table(soilpot, paste0(out.dir, '/soilpots.csv'), row.names=F, sep=',', quote=F)
  write.table(shadpot, paste0(out.dir, '/shadpots.csv'), row.names=F, sep=',', quote=F)
  write.table(humid, paste0(out.dir, '/humids.csv'), row.names=F, sep=',', quote=F)
  write.table(shadhumid, paste0(out.dir, '/shadhumids.csv'), row.names=F, sep=',', quote=F)
  write.table(plant, paste0(out.dir, '/plants.csv'), row.names=F, sep=',', quote=F)
  write.table(shadplant, paste0(out.dir, '/shadplants.csv'), row.names=F, sep=',', quote=F)
  write.table(drlam, paste0(out.dir, '/drlams.csv'), row.names=F, sep=',', quote=F)
  write.table(drrlam, paste0(out.dir, '/drrlams.csv'), row.names=F, sep=',', quote=F)
  write.table(srlam, paste0(out.dir, '/srlams.csv'), row.names=F, sep=',', quote=F)
} else {
  write.table(metout, paste0(out.dir, '/metouts.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(shadmet, paste0(out.dir, '/shadmets.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(soil, paste0(out.dir, '/soils.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(shadsoil, paste0(out.dir, '/shadsoils.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(soilmoist, paste0(out.dir, '/soilmoists.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(shadmoist, paste0(out.dir, '/shadmoists.csv'), append=T, row.names=F, sep=',', quote=F,
col.names=F)
  write.table(soilpot, paste0(out.dir, '/soilpots.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(shadpot, paste0(out.dir, '/shadpots.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(humid, paste0(out.dir, '/humids.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(shadhumid, paste0(out.dir, '/shadhumids.csv'), append=T, row.names=F, sep=',', quote=F,
col.names=F)
  write.table(plant, paste0(out.dir, '/plants.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(shadplant, paste0(out.dir, '/shadplants.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(drlam, paste0(out.dir, '/drlams.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(drrlam, paste0(out.dir, '/drrlams.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
  write.table(srlam, paste0(out.dir, '/srlams.csv'), append=T, row.names=F, sep=',', quote=F, col.names=F)
}
}
} #end

```

C.3 R-code to run custom micro_site() function

```

micro_site_usa <- function (loc = c(-96.56309, 39.10077), dstart = "01/01/2017",
  dfinish = "31/12/2017", nyears = as.numeric(substr(dfinish,
    7, 10)) - as.numeric(substr(dstart, 7, 10)) + 1, REFL = 0.15,
  elev = NA, slope = 0, aspect = 0, lapse_max = 0.0077, lapse_min = 0.0039,
  DEP = c(0, 2.5, 5, 10, 15, 20, 30, 50, 100, 200), minshade = 0,
  maxshade = 90, Refhyt = 2, Ushrht = 0.1, Z01 = 0, Z02 = 0,
  ZH1 = 0, ZH2 = 0, runshade = 0, clearsky = 0, solonly = 0,
  run.gads = 1, write_input = 0, writescv = 0, terrain = 0,
  dailywind = 1, windfac = 1, adiab_cor = 1, warm = 0, spatial = 'spatial',
  ERR = 1.5, RUF = 0.004, EC = 0.0167238, SLE = 0.95, Thcond = 2.5,
  Density = 2.56, SpecHeat = 870, BulkDensity = 1.3, PCTWET = 0,
  rainwet = 1.5, cap = 0, CMH2O = 1, hori = rep(0, 24), TIMAXS = c(1,
    1, 0, 0), TIMINS = c(0, 0, 1, 1), timezone = 0, runmoist = 1,
  PE = rep(1.1, 19), KS = rep(0.0037, 19), BB = rep(4.5, 19),
  BD = rep(BulkDensity, 19), DD = rep(Density, 19), maxpool = 10000,
  rainmult = 1, evenrain = 0, SoilMoist_Init = c(0.1, 0.12,
    0.15, 0.3, 0.4, 0.4, 0.4, 0.4, 0.4, 0.4, 0.4), L = c(0, 0,
    8.2, 8, 7.8, 7.4, 7.1, 6.4, 5.8, 4.8, 4, 1.8, 0.9, 0.6,
    0.8, 0.4, 0.4, 0, 0) * 10000, R1 = 0.001, RW = 2.5e+10,
  RL = 2e+06, PC = -1500, SP = 10, IM = 1e-06, MAXCOUNT = 500,
  LAI = 0.1, snowmodel = 1, snowtemp = 1.5, snowdens = 0.375,
  densfun = c(0.5979, 0.2178, 0.001, 0.0038), snowmelt = 1,
  undercatch = 1, rainmelt = 0.0125, shore = 0, tides = 0,
  scenario = "", year = "", hourly = 0, rainhourly = 0, rainhour = 0,
  rainoff = 0, lamb = 1, IUUV = 0, opendap = 0, soilgrids = 0,
  IR = 0, message = 0, fail = nyears * 24 * 365, save = 0,
  snowcond = 0, intercept = maxshade/100 * 0.4, grassshade = 0)
{
  ystart <- as.numeric(substr(dstart, 7, 10))
  yfinish <- as.numeric(substr(dfinish, 7, 10))
  yearlist <- seq(ystart, (ystart + (nyears - 1)), 1)
  errors <- 0
  if (DEP[2] - DEP[1] > 3 | DEP[3] - DEP[2] > 3) {
    cat("warning, nodes might be too far apart near the surface, try a different spacing if the program is crashing\n")
  }
  if (DEP[2] - DEP[1] < 2) {
    cat("warning, nodes might be too close near the surface, try a different spacing if the program is crashing\n")
  }
  if (is.numeric(loc[1])) {
    if (loc[1] > 180 | loc[2] > 90) {
      cat("ERROR: Latitude or longitude (longlat) is out of bounds.\n      Please enter a correct value.",
        "\n")
      errors <- 1
    }
  }
  if (timezone %in% c(0, 1) == FALSE) {
    cat("ERROR: the variable 'timezone' be either 0 or 1.\n      Please correct.",
      "\n")
    errors <- 1
  }
  if (run.gads %in% c(0, 1) == FALSE) {
    cat("ERROR: the variable 'run.gads' be either 0 or 1.\n      Please correct.",
      "\n")
  }
}

```

```

    errors <- 1
  }
  if (write_input %in% c(0, 1) == FALSE) {
    cat("ERROR: the variable 'write_input' be either 0 or 1.\n    Please correct.",
      "\n")
    errors <- 1
  }
  if (EC < 0.0034 | EC > 0.058) {
    cat("ERROR: the eccentricity variable (EC) is out of bounds.\n    Please enter a correct value (0.0034 -
0.058).",
      "\n")
    errors <- 1
  }
  if (RUF < 1e-04) {
    cat("ERROR: The roughness height (RUF) is too small ( < 0.0001).\n    Please enter a larger value.",
      "\n")
    errors <- 1
  }
  if (RUF > 2) {
    cat("ERROR: The roughness height (RUF) is too large ( > 2).\n    Please enter a smaller value.",
      "\n")
    errors <- 1
  }
  if (DEP[1] != 0) {
    cat("ERROR: First soil node (DEP[1]) must = 0 cm.\n    Please correct",
      "\n")
    errors <- 1
  }
  if (length(DEP) != 10) {
    cat("ERROR: You must enter 10 different soil depths.",
      "\n")
    errors <- 1
  }
  for (i in 1:9) {
    if (DEP[i + 1] <= DEP[i]) {
      cat("ERROR: Soil depth (DEP array) is not in ascending size",
        "\n")
      errors <- 1
    }
  }
  if (DEP[10] > 500) {
    cat("ERROR: Deepest soil depth (DEP array) is too large (<=500 cm)",
      "\n")
    errors <- 1
  }
  if (Thcond < 0) {
    cat("ERROR: Thermal variable conductivity (THCOND) is negative.\n    Please input a positive value.",
      "\n")
    errors <- 1
  }
  if (Density < 0) {
    cat("ERROR: Density variable (Density) is negative.\n    Please input a positive value.",
      "\n")
    errors <- 1
  }
  if (SpecHeat < 0) {

```

```

    cat("ERROR: Specific heat variable (SpecHeat) is negative.\n    Please input a positive value.",
        "\n")
    errors <- 1
  }
  if (min(BulkDensity) < 0) {
    cat("ERROR: Bulk density value (BulkDensity) is negative.\n    Please input a positive value.",
        "\n")
    errors <- 1
  }
  if (REFL < 0 | REFL > 1) {
    cat("ERROR: Soil reflectivity value (REFL) is out of bounds.\n    Please input a value between 0 and 1.",
        "\n")
    errors <- 1
  }
  if (slope < 0 | slope > 90) {
    cat("ERROR: Slope value (slope) is out of bounds.\n    Please input a value between 0 and 90.",
        "\n")
    if (slope < 0) {
      slope <- 0
    } else {
      if (slope > 90) {
        slope <- 90
      }
    }
  }

  # errors <- 1
}
if (aspect < 0 | aspect > 360) {
  cat("ERROR: Aspect value (aspect) is out of bounds.\n    Please input a value between 0 and 365.",
      "\n")
  if (aspect < 0) {
    aspect <- 0
  } else {
    if (aspect > 360) {
      aspect <- 360
    }
  }
}

# errors <- 1
}
if (max(hori) > 90 | min(hori) < 0) {
  cat("ERROR: At least one of your horizon angles (hori) is out of bounds.\n    Please input a value between 0
and 90",
      "\n")

  for (i in 1:length(hori)) {
    if (hori[i] < 0) {
      hori[i] <- 0
    } else {
      if (hori[i] > 90) {
        hori[i] <- 90
      }
    }
  }
}

# errors <- 1

```

```

}
if (length(hori) != 24) {
  cat("ERROR: You must enter 24 horizon angle values.",
      "\n")
  errors <- 1
}
if (SLE < 0.5 | SLE > 1) {
  cat("ERROR: Emissivity (SLE) is out of bounds.\n      Please enter a correct value (0.05 - 1.00).",
      "\n")
  errors <- 1
}
if (ERR < 0) {
  cat("ERROR: Error bound (ERR) is too small.\n      Please enter a correct value (> 0.00).",
      "\n")
  errors <- 1
}
if (Usrhyt < RUF) {
  cat("ERROR: Reference height (Usrhyt) smaller than roughness height (RUF).\n      Please use a larger height
above the surface.",
      "\n")
  errors <- 1
}
if (Usrhyt < 0.005 | Usrhyt > Refhyt) {
  cat("ERROR: Local height (Usrhyt) is out of bounds.\n      Please enter a correct value (0.005 - Refhyt).",
      "\n")
  errors <- 1
}
if (CMH2O < 0.5 | CMH2O > 2) {
  cat("ERROR: Precipitable water in air column (CMH2O) is out of bounds.\n      Please enter a correct value
(0.1 - 2cm).",
      "\n")
  errors <- 1
}
if (max(TIMAXS) > 24 | min(TIMAXS) < 0) {
  cat("ERROR: At least one of your times of weather maxima (TIMAXS) is out of bounds.\n      Please input a
value between 0 and 24",
      "\n")
  errors <- 1
}
if (max(TIMINS) > 24 | min(TIMINS) < 0) {
  cat("ERROR: At least one of your times of weather minima (TIMINS) is out of bounds.\n      Please input a
value between 0 and 24",
      "\n")
  errors <- 1
}
if (minshade > maxshade | minshade == maxshade) {
  cat("ERROR: Your value for minimum shade (minshade) is greater than or equal to the maximum shade
(maxshade).\n      Please correct this.",
      "\n")
  errors <- 1
}
if (minshade > 100 | minshade < 0) {
  cat("ERROR: Your value for minimum shade (minshade) is out of bounds.\n      Please input a value between
0 and 100.",
      "\n")
  errors <- 1
}

```

```

}
if (maxshade > 100 | maxshade < 0) {
  cat("ERROR: Your value for maximum shade (maxshade) is out of bounds.\n    Please input a value between
0 and 100.",
    "\n")
  errors <- 1
}
if (errors == 0) {
  doys12 <- c(15, 46, 74, 105, 135, 166, 196, 227, 258,
    288, 319, 349)
  microdaily <- 1
  daystart <- 1
  idayst <- 1
  if (!require("raster", quietly = TRUE)) {
    stop("package 'raster' is needed. Please install it.",
      call. = FALSE)
  }
  if (!require("ncdf4", quietly = TRUE)) {
    stop("package 'ncdf4' is needed. Please install it.",
      call. = FALSE)
  }
  longlat <- loc
  x <- t(as.matrix(as.numeric(c(loc[1], loc[2]))))
  require("raster")
  require("ncdf4")
  if (timezone == 1) {
    if (!require("geonames", quietly = TRUE)) {
      stop("package \"geonames\" is required to do a specific time zone (timezone=1). Please install it.")
    }
    ALREF <- (geonames::GNtimezone(longlat[2], longlat[1])[4]) *
      -15
  }
  else {
    ALREF <- abs(trunc(x[1]))
  }
  HEMIS <- ifelse(x[2] < 0, 2, 1)
  ALAT <- abs(trunc(x[2]))
  AMINUT <- (abs(x[2]) - ALAT) * 60
  ALONG <- abs(trunc(x[1]))
  ALMINT <- (abs(x[1]) - ALONG) * 60
  azimuth <- aspect
  if (save != 2) {
    if (opendap == 1) {
      if (!require("futile.logger", quietly = TRUE)) {
        stop("package \"futile.logger\" is required for this opendap extraction process. Please install it.")
      }
      require("utils")
      retry <- function(expr, isError = function(x) "try-error" %in%
        class(x), maxErrors = 50, sleep = 1) {
        attempts = 0
        retval = try(eval(expr))
        while (isError(retval)) {
          attempts = attempts + 1
          if (attempts >= maxErrors) {
            msg = sprintf("retry: too many retries [[%s]]",
              capture.output(str(retval)))

```

```

        futile.logger::flog.fatal(msg)
        stop(msg)
    }
    else {
        msg = sprintf("retry: error in attempt %i/%i [[%s]]",
            attempts, maxErrors, capture.output(str(retval)))
        futile.logger::flog.error(msg)
        warning(msg)
    }
    if (sleep > 0)
        Sys.sleep(sleep)
    retval = try(eval(expr))
}
return(retval)
}
cat("extracting elevation via opendaps \n")
baseurl <- "http://thredds.northwestknowledge.net:8080/thredds/dodsC/MET/"
# nc <- ncdf4::nc_open(paste0(baseurl, "elev/metdata_elevationdata.nc"))
nc <- ncdf4::nc_open(paste0(spatial, 'elev/metdata_elevationdata.nc'))
lon <- ncdf4::ncvar_get(nc, "lon")
lat <- ncdf4::ncvar_get(nc, "lat")
flat = match(abs(lat - x[2]) < 1/48, 1)
latindex = which(flat %in% 1)
flon = match(abs(lon - x[1]) < 1/48, 1)
lonindex = which(flon %in% 1)
start <- c(lonindex, latindex, 1)
count <- c(1, 1, -1)
USADEM <- retry(as.numeric(ncdf4::ncvar_get(nc,
    varid = "elevation", start = start, count = count)))
ncdf4::nc_close(nc)
}
else {
    USADEM <- extract(raster::raster(paste0(spatial,
        "/elev/metdata_elevationdata.nc")), x)
}
if (save == 1) {
    cat("saving DEM data for later \n")
    save(USADEM, file = "USADEM.Rda")
}
}
if (save == 2) {
    cat("loading DEM data from previous run \n")
    load("USADEM.Rda")
}
ALTITUDES <- NA
if (is.na(elev) == FALSE) {
    ALTITUDES <- elev
}
if (is.na(ALTITUDES) == TRUE) {
    ALTITUDES <- USADEM
}
if (save != 2) {
    if (soilgrids == 1) {
        cat("extracting data from SoilGrids \n")
        require(jsonlite)
        ov <- jsonlite::fromJSON(paste0("https://rest.soilgrids.org/query?lon=",

```

```

    x[1, "&lat=", x[2, ], "&attributes=BLDFIE,SLTPPT,SNDPPT,CLYPPT"),
    flatten = TRUE)
  if (length(ov) > 3) {
    soilpro <- cbind(c(0, 5, 15, 30, 60, 100, 200),
      unlist(ov$properties$BLDFIE$M)/1000, unlist(ov$properties$SLTPPT$M),
      unlist(ov$properties$SNDPPT$M), unlist(ov$properties$CLYPPT$M))
    colnames(soilpro) <- c("depth", "blkdens",
      "clay", "silt", "sand")
    soil.hydro <- pedotransfer(soilpro = as.data.frame(soilpro),
      DEP = DEP)
    PE <- soil.hydro$PE
    BB <- soil.hydro$BB
    BD <- soil.hydro$BD
    KS <- soil.hydro$KS
    BulkDensity <- BD[seq(1, 19, 2)]
  }
  else {
    cat("no SoilGrids data for this site, using user-input soil properties \n")
  }
}
}
else {
  if (soilgrids == 1) {
    cat("loading SoilGrids data from previous run \n")
    load("PE.Rda")
    load("BB.Rda")
    load("BD.Rda")
    load("KS.Rda")
    load("BulkDensity.Rda")
  }
}
if (save == 1 & soilgrids == 1) {
  cat("saving SoilGrids data for later \n")
  save(PE, file = "PE.Rda")
  save(BB, file = "BB.Rda")
  save(BD, file = "BD.Rda")
  save(KS, file = "KS.Rda")
  save(BulkDensity, file = "BulkDensity.Rda")
}
if (terrain == 1) {
  cat("extracting terrain data \n")
  HORIZONS <- elevslpasphori[4:27]
  SLOPES <- elevslpasphori[2]
  AZMUTHS <- elevslpasphori[3]
  HORIZONS <- (ifelse(is.na(HORIZONS), 0, HORIZONS))/10
  HORIZONS <- data.frame(HORIZONS)
}
else {
  HORIZONS <- hori
  HORIZONS <- data.frame(HORIZONS)
  VIEWF_all <- 1 - sum(sin(as.data.frame(hori) * pi/180))/length(hori)
  SLOPES <- rep(slope, length(x[, 1]))
  AZMUTHS <- rep(aspect, length(x[, 1]))
}
hori <- HORIZONS
row.names(hori) <- NULL

```

```

hori <- as.numeric(as.matrix(hori))
VIEWF <- VIEWF_all
delta_elev = USADEM - ALTITUDES
adiab_corr_max <- delta_elev * lapse_max
adiab_corr_min <- delta_elev * lapse_min
if (save != 2) {
  if (opendap == 1) {
    cat("extracting weather data \n")
    dates <- Sys.time() - 60 * 60 * 24
    curyear <- as.numeric(format(dates, "%Y"))
    days <- seq(as.POSIXct(dstart, format = "%d/%m/%Y",
      origin = "01/01/1900"), as.POSIXct(dfinish,
      format = "%d/%m/%Y", origin = "01/01/1900"),
      by = "days")
    alldays <- seq(as.POSIXct("01/01/1900", format = "%d/%m/%Y",
      origin = "01/01/1900"), Sys.time() - 60 * 60 *
      24, by = "days")
    startday <- which(as.character(format(alldays,
      "%d/%m/%Y")) == format(as.POSIXct(dstart, format = "%d/%m/%Y",
      origin = "01/01/1900"), "%d/%m/%Y"))
    endday <- which(as.character(format(alldays,
      "%d/%m/%Y")) == format(as.POSIXct(dfinish,
      format = "%d/%m/%Y", origin = "01/01/1900"),
      "%d/%m/%Y"))
    countday <- endday - startday + 1
    # baseurl <- "http://thredds.northwestknowledge.net:8080/thredds/dodsC/agg_met_"
    cat(paste0("reading weather input for ", dstart,
      " to ", dfinish, " \n"))
    cat(paste0("tmin weather input \n"))
    # nc <- ncdf4::nc_open(paste0(baseurl, "tmmn_1979_CurrentYear_CONUS.nc"))
    nc <- ncdf4::nc_open(paste0(spatial, "/tmmn/tmmn_", ystart, ".nc"))
    day <- retry(as.numeric(ncvar_get(nc, varid = "day")))
    lon <- ncdf4::ncvar_get(nc, "lon")
    lat <- ncdf4::ncvar_get(nc, "lat")
    flat <- match(abs(lat - x[2]) < 1/48, 1)
    latindex <- which(flat %in% 1)
    flon <- match(abs(lon - x[1]) < 1/48, 1)
    lonindex <- which(flon %in% 1)
    start <- c(lonindex, latindex, which(day == startday) -
      1)
    count <- c(1, 1, countday)
    Tmin <- retry(as.numeric(ncvar_get(nc, varid = "daily_minimum_temperature",
      start = start, count = count))) - 273.15
    nc_close(nc)
    cat(paste0("tmax weather input \n"))
    nc <- nc_open(paste0(baseurl, "tmmx_1979_CurrentYear_CONUS.nc"))
    # nc <- ncdf4::nc_open(paste0(spatial, "/tmmx/tmmx_", ystart, ".nc"))
    Tmax <- retry(as.numeric(ncdf4::ncvar_get(nc,
      varid = "daily_maximum_temperature", start = start,
      count = count))) - 273.15
    ncdf4::nc_close(nc)
    cat(paste0("rhmin weather input \n"))
    nc <- ncdf4::nc_open(paste0(baseurl, "rmin_1979_CurrentYear_CONUS.nc"))
    # nc <- ncdf4::nc_open(paste0(spatial, "/rmin/rmin_", ystart, ".nc"))
    rhmin <- retry(as.numeric(ncdf4::ncvar_get(nc,
      varid = "daily_minimum_relative_humidity",

```

```

      start = start, count = count)))
ncdf4::nc_close(nc)
cat(paste0("rhmax weather input \n"))
nc <- ncdf4::nc_open(paste0(baseurl, "rmax_1979_CurrentYear_CONUS.nc"))
# nc <- ncdf4::nc_open(paste0(spatial, "/rmax/rmax_", ystart, ".nc"))
rhmax <- retry(as.numeric(ncdf4::ncvar_get(nc,
  varid = "daily_maximum_relative_humidity",
  start = start, count = count)))
ncdf4::nc_close(nc)
cat(paste0("rain weather input \n"))
nc <- ncdf4::nc_open(paste0(baseurl, "pr_1979_CurrentYear_CONUS.nc"))
# nc <- ncdf4::nc_open(paste0(spatial, "/pr/pr_", ystart, ".nc"))
Rain <- retry(as.numeric(ncdf4::ncvar_get(nc,
  varid = "precipitation_amount", start = start,
  count = count)))
ncdf4::nc_close(nc)
cat(paste0("solar weather input \n"))
nc <- ncdf4::nc_open(paste0(baseurl, "srad_1979_CurrentYear_CONUS.nc"))
# nc <- ncdf4::nc_open(paste0(spatial, "/srad/srad_", ystart, ".nc"))
solar <- retry(as.numeric(ncdf4::ncvar_get(nc,
  varid = "daily_mean_shortwave_radiation_at_surface",
  start = start, count = count)))
ncdf4::nc_close(nc)
cat(paste0("wind weather input \n"))
nc <- ncdf4::nc_open(paste0(baseurl, "vs_1979_CurrentYear_CONUS.nc"))
# nc <- ncdf4::nc_open(paste0(spatial, "/vs/vs_", ystart, ".nc"))
Wind <- retry(as.numeric(ncdf4::ncvar_get(nc,
  varid = "daily_mean_wind_speed", start = start,
  count = count)))
ncdf4::nc_close(nc)
}
else {
  cat("extracting weather data \n")
  nc <- ncdf4::nc_open(paste(spatial, "/tmmx/tmmx_",
    yearlist[1], ".nc", sep = ""))
  lon <- matrix(ncdf4::ncvar_get(nc, "lon"))
  lat <- matrix(ncdf4::ncvar_get(nc, "lat"))
  lon_1 <- as.numeric(longlat[1])
  lat_1 <- as.numeric(longlat[2])
  dist1 <- abs(lon - lon_1)
  index1 <- which.min(dist1)
  dist2 <- abs(lat - lat_1)
  index2 <- which.min(dist2)
  start <- c(index1, index2, 1)
  count <- c(1, 1, -1)
  for (j in 1:nyears) {
    if (j == 1) {
      cat(paste("reading weather input for ", yearlist[j],
        "\n", sep = ""))
      nc <- ncdf4::nc_open(paste(spatial, "/tmmn/tmmn_",
        yearlist[j], ".nc", sep = ""))
      tmin <- as.numeric(ncdf4::ncvar_get(nc, varid = "air_temperature",
        start = start, count = count))
      ncdf4::nc_close(nc)
      nc <- ncdf4::nc_open(paste(spatial, "/tmmx/tmmx_",
        yearlist[j], ".nc", sep = ""))
    }
  }
}

```

```

tmax <- as.numeric(ncdf4::ncvar_get(nc, varid = "air_temperature",
  start = start, count = count))
ncdf4::nc_close(nc)
nc <- ncdf4::nc_open(paste(spatial, "/rmin/rmin_",
  yearlist[j], ".nc", sep = ""))
rhmin <- as.numeric(ncdf4::ncvar_get(nc,
  varid = "relative_humidity", start = start,
  count = count))
ncdf4::nc_close(nc)
nc <- ncdf4::nc_open(paste(spatial, "/rmax/rmax_",
  yearlist[j], ".nc", sep = ""))
rhmax <- as.numeric(ncdf4::ncvar_get(nc,
  varid = "relative_humidity", start = start,
  count = count))
ncdf4::nc_close(nc)
nc <- ncdf4::nc_open(paste(spatial, "/pr/pr_",
  yearlist[j], ".nc", sep = ""))
Rain <- as.numeric(ncdf4::ncvar_get(nc, varid = "precipitation_amount",
  start = start, count = count))
ncdf4::nc_close(nc)
nc <- ncdf4::nc_open(paste(spatial, "/srad/srad_",
  yearlist[j], ".nc", sep = ""))
solar <- as.numeric(ncdf4::ncvar_get(nc,
  varid = "surface_downwelling_shortwave_flux_in_air",
  start = start, count = count))
ncdf4::nc_close(nc)
nc <- ncdf4::nc_open(paste(spatial, "/vs/vs_",
  yearlist[j], ".nc", sep = ""))
Wind <- as.numeric(ncdf4::ncvar_get(nc, varid = "wind_speed",
  start = start, count = count))
ncdf4::nc_close(nc)
Tmax <- tmax - 273.15
Tmin <- tmin - 273.15
}
else {
cat(paste("reading weather input for ", yearlist[j],
  "\n", sep = ""))
nc <- ncdf4::nc_open(paste(spatial, "/tmmn/tmmn_",
  yearlist[j], ".nc", sep = ""))
tmin <- as.numeric(ncdf4::ncvar_get(nc, varid = "air_temperature",
  start = start, count = count))
ncdf4::nc_close(nc)
nc <- ncdf4::nc_open(paste(spatial, "/tmmx/tmmx_",
  yearlist[j], ".nc", sep = ""))
tmax <- as.numeric(ncdf4::ncvar_get(nc, varid = "air_temperature",
  start = start, count = count))
ncdf4::nc_close(nc)
nc <- ncdf4::nc_open(paste(spatial, "/rmin/rmin_",
  yearlist[j], ".nc", sep = ""))
rhmin <- c(rhmin, as.numeric(ncdf4::ncvar_get(nc,
  varid = "relative_humidity", start = start,
  count = count)))
ncdf4::nc_close(nc)
nc <- ncdf4::nc_open(paste(spatial, "/rmax/rmax_",
  yearlist[j], ".nc", sep = ""))
rhmax <- c(rhmax, as.numeric(ncdf4::ncvar_get(nc,

```

```

    varid = "relative_humidity", start = start,
    count = count)))
ncdf4::nc_close(nc)
nc <- ncdf4::nc_open(paste(spatial, "/pr/pr_",
  yearlist[j], ".nc", sep = ""))
Rain <- c(Rain, as.numeric(ncdf4::ncvar_get(nc,
  varid = "precipitation_amount", start = start,
  count = count)))
ncdf4::nc_close(nc)
nc <- ncdf4::nc_open(paste(spatial, "/srad/srad_",
  yearlist[j], ".nc", sep = ""))
solar <- c(solar, as.numeric(ncdf4::ncvar_get(nc,
  varid = "surface_downwelling_shortwave_flux_in_air",
  start = start, count = count)))
ncdf4::nc_close(nc)
nc <- ncdf4::nc_open(paste(spatial, "/vs/vs_",
  yearlist[j], ".nc", sep = ""))
Wind <- c(Wind, as.numeric(ncdf4::ncvar_get(nc,
  varid = "wind_speed", start = start, count = count)))
ncdf4::nc_close(nc)
Tmax <- c(Tmax, tmax - 273.15)
Tmin <- c(Tmin, tmin - 273.15)
}
}
}
cat("running micro_global to get clear sky solar\n")
micro_clearsky <- micro_global(loc = c(x[1], x[2]),
  clearsky = 1, timeinterval = 365)
clearskyrad <- micro_clearsky$metout[, c(1, 13)]
clearsky_mean1 <- aggregate(clearskyrad[, 2], by = list(clearskyrad[,
  1]), FUN = mean)[, 2]
leapyears <- seq(1900, 2100, 4)
for (j in 1:nyears) {
  if (yearlist[j] %in% leapyears) {
    clearsky_mean <- c(clearsky_mean1[1:59], clearsky_mean1[59],
      clearsky_mean1[60:365])
  }
  else {
    clearsky_mean <- clearsky_mean1
  }
  if (j == 1) {
    allclearsky <- clearsky_mean
  }
  else {
    allclearsky <- c(allclearsky, clearsky_mean)
  }
}
if (opendap == 1 & save != 2) {
  cut <- as.numeric(days[1] - as.POSIXct(paste0("01/01/",
    ystart), format = "%d/%m/%Y") + 1)
  allclearsky <- allclearsky[cut:(cut + countday -
    1)]
}
cloud <- (1 - solar/allclearsky) * 100
cloud[cloud < 0] <- 0
cloud[cloud > 100] <- 100

```

```

if (clearsky == 1) {
  cloud <- cloud * 0
}
CCMAXX <- as.numeric(cloud)
CCMINN <- CCMAXX
CCMINN <- CCMINN * 0.5
CCMAXX <- CCMAXX * 2
CCMINN[CCMINN > 100] <- 100
CCMAXX[CCMAXX > 100] <- 100
Wind[Wind == 0] <- 0.1
if (save == 1) {
  cat("saving met data for later \n")
  save(CCMAXX, file = "CCMAXX.Rda")
  save(CCMINN, file = "CCMINN.Rda")
  save(Wind, file = "Wind.Rda")
  save(Tmax, file = "Tmax.Rda")
  save(Tmin, file = "Tmin.Rda")
  save(rhmax, file = "rhmax.Rda")
  save(rhmin, file = "rhmin.Rda")
  save(Rain, file = "Rain.Rda")
}
}
else {
  cat("loading met data from previous run \n")
  load("CCMAXX.Rda")
  load("CCMINN.Rda")
  load("Wind.Rda")
  load("Tmax.Rda")
  load("Tmin.Rda")
  load("rhmax.Rda")
  load("rhmin.Rda")
  load("Rain.Rda")
}
ndays <- length(Tmax)
doynum <- ndays
leapyears <- seq(1900, 2100, 4)
for (k in 1:leapyears) {
  if (k == 1) {
    cyear <- ystart
  }
  else {
    cyear <- cyear + 1
  }
  if (cyear %in% leapyears) {
    dinyear <- 366
  }
  else {
    dinyear <- 365
  }
  if (k == 1) {
    doy <- seq(1, dinyear)
  }
  else {
    doy <- c(doy, seq(1, dinyear))
  }
}
}

```

```

if(opendap == 1 & save != 2) {
  doy <- doy[cut:(cut + countday - 1)]
}
ida <- ndays
idayst <- 1
ndays <- length(Tmin)
maxshades = rep(maxshade, ndays)
minshades = rep(minshade, ndays)
shademax <- maxshades
if (is.na(ALTITUDES) != TRUE) {
  if (run.gads == 1) {
    relhum <- 1
    optdep.summer <- as.data.frame(rungads(longlat[2],
      longlat[1], relhum, 0))
    optdep.winter <- as.data.frame(rungads(longlat[2],
      longlat[1], relhum, 1))
    optdep <- cbind(optdep.winter[, 1], rowMeans(cbind(optdep.summer[,
      2], optdep.winter[, 2])))
    optdep <- as.data.frame(optdep)
    colnames(optdep) <- c("LAMBDA", "OPTDEPTH")
    a <- lm(OPTDEPTH ~ poly(LAMBDA, 6, raw = TRUE),
      data = optdep)
    LAMBDA <- c(290, 295, 300, 305, 310, 315, 320,
      330, 340, 350, 360, 370, 380, 390, 400, 420,
      440, 460, 480, 500, 520, 540, 560, 580, 600,
      620, 640, 660, 680, 700, 720, 740, 760, 780,
      800, 820, 840, 860, 880, 900, 920, 940, 960,
      980, 1000, 1020, 1080, 1100, 1120, 1140, 1160,
      1180, 1200, 1220, 1240, 1260, 1280, 1300, 1320,
      1380, 1400, 1420, 1440, 1460, 1480, 1500, 1540,
      1580, 1600, 1620, 1640, 1660, 1700, 1720, 1780,
      1800, 1860, 1900, 1950, 2000, 2020, 2050, 2100,
      2120, 2150, 2200, 2260, 2300, 2320, 2350, 2380,
      2400, 2420, 2450, 2490, 2500, 2600, 2700, 2800,
      2900, 3000, 3100, 3200, 3300, 3400, 3500, 3600,
      3700, 3800, 3900, 4000)
    TAI <- predict(a, data.frame(LAMBDA))
  }
  else {
    TAI <- c(0.0670358341290886, 0.0662612704779235,
      0.065497075238002, 0.0647431301168489, 0.0639993178022531,
      0.0632655219571553, 0.0625416272145492, 0.0611230843885423,
      0.0597427855962549, 0.0583998423063099, 0.0570933810229656,
      0.0558225431259535, 0.0545864847111214, 0.0533843764318805,
      0.0522154033414562, 0.0499736739981675, 0.047855059159556,
      0.0458535417401334, 0.0439633201842001, 0.0421788036108921,
      0.0404946070106968, 0.0389055464934382, 0.0374066345877315,
      0.0359930755919066, 0.0346602609764008, 0.0334037648376212,
      0.0322193394032758, 0.0311029105891739, 0.0300505736074963,
      0.0290585886265337, 0.0281233764818952, 0.0272415144391857,
      0.0264097320081524, 0.0256249068083005, 0.0248840604859789,
      0.0241843546829336, 0.0235230870563317, 0.0228976873502544,
      0.0223057135186581, 0.0217448478998064, 0.0212128934421699,
      0.0207077699817964, 0.0202275105711489, 0.0197702578594144,
      0.0193342605242809, 0.0189178697551836, 0.0177713140039894,
      0.0174187914242432, 0.0170790495503944, 0.0167509836728154,

```

```

0.0164335684174899, 0.0161258546410128, 0.0158269663770596,
0.0155360978343254, 0.0152525104459325, 0.0149755299703076,
0.0147045436435285, 0.0144389973831391, 0.0141783930434343,
0.0134220329447663, 0.0131772403830191, 0.0129356456025128,
0.0126970313213065, 0.0124612184223418, 0.0122280636204822,
0.01199745718102, 0.0115436048739351, 0.0110993711778668,
0.0108808815754663, 0.0106648652077878, 0.0104513876347606,
0.0102405315676965, 0.00982708969547694, 0.00962473896278535,
0.00903679230300494, 0.00884767454432418, 0.0083031278398166,
0.00796072474935954, 0.00755817587626185, 0.00718610751850881,
0.00704629977586921, 0.00684663903049612, 0.00654155580333479,
0.00642947339729728, 0.00627223096874308, 0.00603955966866779,
0.00580920937536261, 0.00568506186880564, 0.00563167068287251,
0.00556222005081865, 0.00550522989971023, 0.00547395763028062,
0.0054478983436216, 0.00541823364504573, 0.00539532163908382,
0.00539239864119488, 0.00541690124712384, 0.00551525885358836,
0.00564825853509463, 0.00577220185074264, 0.00584222986640171,
0.00581645238345584, 0.00566088137411449, 0.00535516862329704,
0.00489914757707667, 0.00432017939770409, 0.0036813032251836,
0.00309019064543606, 0.00270890436501562, 0.00276446109239711,
0.00356019862584603)
}
if (adiab_cor == 1) {
  TMAXX <- as.matrix(Tmax + adiab_corr_max)
  TMINN <- as.matrix(Tmin + adiab_corr_min)
}
else {
  TMAXX <- as.matrix(Tmax)
  TMINN <- as.matrix(Tmin)
}
if (warm != 0) {
  TMAXX <- TMAXX + seq(0, ndays - 1)/(ndays - 1) *
    warm
  TMINN <- TMINN + seq(0, ndays - 1)/(ndays - 1) *
    warm
}
RAINFALL <- Rain + rainoff
es <- WETAIR(db = TMAXX, rh = 100)$esat
e <- WETAIR(db = Tmax, rh = rhmin)$e
RHMINN <- (e/es) * 100
RHMINN[RHMINN > 100] <- 100
RHMINN[RHMINN < 0] <- 0.01
es <- WETAIR(db = TMINN, rh = 100)$esat
e <- WETAIR(db = Tmin, rh = rhmax)$e
RHMAXX <- (e/es) * 100
RHMAXX[RHMAXX > 100] <- 100
RHMAXX[RHMAXX < 0] <- 0.01
ALLMINTemps <- TMINN
ALLMAXTEMPs <- TMAXX
ALLTEMPs <- cbind(ALLMAXTEMPs, ALLMINTemps)
WNMAXX <- Wind * windfac
WNMINN <- Wind * windfac
MAXSHADES <- maxshades
MINSHADES <- minshades
REFLS <- rep(REFL, ndays)
PCTWET <- rep(PCTWET, ndays)

```

```

soilwet <- RAINFALL
soilwet[soilwet <= rainwet] = 0
soilwet[soilwet > 0] = 90
PCTWET <- pmax(soilwet, PCTWET)
Intrvls <- rep(0, ndays)
Intrvls[1] <- 1
Numtyps <- 1
Numint <- 1
Nodes <- matrix(data = 0, nrow = 10, ncol = ndays)
Nodes[1, 1] <- 10
ALREF <- abs(trunc(x[1]))
HEMIS <- ifelse(x[2] < 0, 2, 1)
ALAT <- abs(trunc(x[2]))
AMINUT <- (abs(x[2]) - ALAT) * 60
ALONG <- abs(trunc(x[1]))
ALMINT <- (abs(x[1]) - ALONG) * 60
if (adiab_cor == 1) {
  ALTT <- ALTITUDES
}
else {
  ALTT <- UKDEM
}
SLOPE <- SLOPES
AZMUTH <- AZMUTHS
avetemp <- (sum(TMAXX) + sum(TMINN))/(length(TMAXX) *
  2)
soilinit <- rep(avetemp, 20)
tannul <- mean(unlist(ALLTEMPS))
if (nyears == 1) {
  avetemp <- (sum(TMAXX) + sum(TMINN))/(length(TMAXX) *
    2)
  tannulrun <- rep(avetemp, ndays)
}
else {
  if (nrow(TMAXX) == 1) {
    avetemp <- colMeans(cbind(TMAXX, TMINN), na.rm = TRUE)
  }
  else {
    avetemp <- rowMeans(cbind(TMAXX, TMINN), na.rm = TRUE)
  }
  if (length(TMAXX) < 365) {
    tannulrun <- rep((sum(TMAXX) + sum(TMINN))/(length(TMAXX) *
      2), length(TMAXX))
  }
  else {
    tannulrun <- raster::movingFun(avetemp, n = 365,
      fun = mean, type = "to")
    yearone <- rep((sum(TMAXX[1:365]) + sum(TMINN[1:365]))/(365 *
      2), 365)
    tannulrun[1:365] <- yearone
  }
}
WNMAXX <- WNMAXX * (2/10)^0.15
WNMINN <- WNMINN * (2/10)^0.15
SLES <- matrix(nrow = ndays, data = 0)
SLES <- SLES + SLE

```

```

moists2 <- matrix(nrow = 10, ncol = ndays, data = 0)
moists2[1, ndays] <- 0.2
moists <- moist2
if (runmoist == 1) {
  moist2 <- matrix(nrow = 10, ncol = ndays, data = 0)
  moist2[1:10, ] <- SoilMoist_Init
  moist <- moist2
}
soilprops <- matrix(data = 0, nrow = 10, ncol = 5)
soilprops[, 1] <- BulkDensity
soilprops[, 2] <- min(0.26, 1 - BulkDensity/Density)
soilprops[, 3] <- Thcond
soilprops[, 4] <- SpecHeat
soilprops[, 5] <- Density
if (cap == 1) {
  soilprops[1:2, 3] <- 0.2
  soilprops[1:2, 4] <- 1920
}
if (cap == 2) {
  soilprops[1:2, 3] <- 0.1
  soilprops[3:4, 3] <- 0.25
  soilprops[1:4, 4] <- 1920
  soilprops[1:4, 5] <- 1.3
  soilprops[1:4, 1] <- 0.7
}
ALTT <- as.numeric(ALTT)
ALREF <- as.numeric(ALREF)
ALMINT <- as.numeric(ALMINT)
ALONG <- as.numeric(ALONG)
AMINUT <- as.numeric(AMINUT)
ALAT <- as.numeric(ALAT)
microinput <- c(ndays, RUF, ERR, Usrhyt, Refhyt,
  Numtys, Z01, Z02, ZH1, ZH2, idayst, ida, HEMIS,
  ALAT, AMINUT, ALONG, ALMINT, ALREF, slope, azimuth,
  ALTT, CMH2O, microdaily, tannul, EC, VIEWF, snowtemp,
  snowdens, snowmelt, undercatch, rainmult, runshade,
  runmoist, maxpool, evenrain, snowmodel, rainmelt,
  writcsv, densfun, hourly, rainhourly, lamb,
  IUUV, RW, PC, RL, SP, R1, IM, MAXCOUNT, IR, message,
  fail, snowcond, intercept, grassshade, solonly)
if (hourly == 0) {
  TAIRhr = rep(0, 24 * ndays)
  RHhr = rep(0, 24 * ndays)
  WNhr = rep(0, 24 * ndays)
  CLDhr = rep(0, 24 * ndays)
  SOLRhr = rep(0, 24 * ndays)
  ZENhr = rep(-1, 24 * ndays)
  IRDhr = rep(-1, 24 * ndays)
}
if (rainhourly == 0) {
  RAINhr = rep(0, 24 * ndays)
}
else {
  RAINhr = rainhour
}
doy1 = matrix(data = 0, nrow = ndays, ncol = 1)

```

```

SLES1 = matrix(data = 0, nrow = ndays, ncol = 1)
MAXSHADES1 = matrix(data = 0, nrow = ndays, ncol = 1)
MINSHADES1 = matrix(data = 0, nrow = ndays, ncol = 1)
TMAXX1 = matrix(data = 0, nrow = ndays, ncol = 1)
TMINN1 = matrix(data = 0, nrow = ndays, ncol = 1)
CCMAXX1 = matrix(data = 0, nrow = ndays, ncol = 1)
CCMINN1 = matrix(data = 0, nrow = ndays, ncol = 1)
RHMAXX1 = matrix(data = 0, nrow = ndays, ncol = 1)
RHMINN1 = matrix(data = 0, nrow = ndays, ncol = 1)
WNMAXX1 = matrix(data = 0, nrow = ndays, ncol = 1)
WNMINN1 = matrix(data = 0, nrow = ndays, ncol = 1)
REFLS1 = matrix(data = 0, nrow = ndays, ncol = 1)
PCTWET1 = matrix(data = 0, nrow = ndays, ncol = 1)
RAINFALL1 = matrix(data = 0, nrow = ndays, ncol = 1)
tannul1 = matrix(data = 0, nrow = ndays, ncol = 1)
moists1 = matrix(data = 0, nrow = 10, ncol = ndays)
doy1[1:ndays] <- doy
SLES1[1:ndays] <- SLES
MAXSHADES1[1:ndays] <- MAXSHADES
MINSHADES1[1:ndays] <- MINSHADES
TMAXX1[1:ndays] <- TMAXX
TMINN1[1:ndays] <- TMINN
CCMAXX1[1:ndays] <- CCMAXX
CCMINN1[1:ndays] <- CCMINN
RHMAXX1[1:ndays] <- RHMAXX
RHMINN1[1:ndays] <- RHMINN
WNMAXX1[1:ndays] <- WNMAXX
WNMINN1[1:ndays] <- WNMINN
REFLS1[1:ndays] <- REFLS
PCTWET1[1:ndays] <- PCTWET
RAINFALL1[1:ndays] <- RAINFALL
tannul1[1:ndays] <- tannul
moists1[1:10, 1:ndays] <- moist
if (length(LAI) < ndays) {
  LAI <- rep(LAI[1], ndays)
  LAI1 <- LAI
}
if (shore == 0) {
  tides <- matrix(data = 0, nrow = 24 * ndays,
    ncol = 3)
}
micro <- list(tides = tides, microinput = microinput,
  doy = doy, SLES = SLES1, DEP = DEP, Nodes = Nodes,
  MAXSHADES = MAXSHADES1, MINSHADES = MINSHADES1,
  TIMAXS = TIMAXS, TIMINS = TIMINS, TMAXX = TMAXX1,
  TMINN = TMINN1, RHMAXX = RHMAXX1, RHMINN = RHMINN1,
  CCMAXX = CCMAXX1, CCMINN = CCMINN1, WNMAXX = WNMAXX1,
  WNMINN = WNMINN1, TAIRhr = TAIRhr, RHhr = RHhr,
  WNhr = WNhr, CLDhr = CLDhr, SOLRhr = SOLRhr,
  RAINhr = RAINhr, ZENhr = ZENhr, IRDhr = IRDhr,
  REFLS = REFLS1, PCTWET = PCTWET1, soilinit = soilinit,
  hori = hori, TAI = TAI, soilprops = soilprops,
  moist = moist1, RAINFALL = RAINFALL1, tannulrun = tannulrun,
  PE = PE, KS = KS, BB = BB, BD = BD, DD = DD,
  L = L, LAI = LAI1)
if (write_input == 1) {

```

```

if (dir.exists("micro csv input") == FALSE) {
  dir.create("micro csv input")
}
write.table(as.matrix(microinput), file = "micro csv input/microinput.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(doy, file = "micro csv input/doy.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(SLES, file = "micro csv input/SLES.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(DEP, file = "micro csv input/DEP.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(Nodes, file = "micro csv input/Nodes.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(MAXSHADES, file = "micro csv input/Maxshades.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(MINSHADES, file = "micro csv input/Minshades.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(TIMAXS, file = "micro csv input/TIMAXS.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(TIMINS, file = "micro csv input/TIMINS.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(TMAXX, file = "micro csv input/TMAXX.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(TMINN, file = "micro csv input/TMINN.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(RHMAXX, file = "micro csv input/RHMAXX.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(RHMINN, file = "micro csv input/RHMINN.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(CCMAXX, file = "micro csv input/CCMAXX.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(CCMINN, file = "micro csv input/CCMINN.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(WNMAXX, file = "micro csv input/WNMAXX.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(WNMINN, file = "micro csv input/WNMINN.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(REFLS, file = "micro csv input/REFLS.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(PCTWET, file = "micro csv input/PCTWET.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(soilinit, file = "micro csv input/soilinit.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(hori, file = "micro csv input/hori.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(TAI, file = "micro csv input/TAI.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(soilprops, file = "micro csv input/soilprop.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(moists, file = "micro csv input/moists.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(RAINFALL, file = "micro csv input/rain.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(tannulrun, file = "micro csv input/tannulrun.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(PE, file = "micro csv input/PE.csv",

```

```

    sep = ",", col.names = NA, qmethod = "double")
write.table(BD, file = "micro csv input/BD.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(DD, file = "micro csv input/DD.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(BB, file = "micro csv input/BB.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(KS, file = "micro csv input/KS.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(L, file = "micro csv input/L.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(LAI, file = "micro csv input/LAI.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(tides, file = "micro csv input/tides.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(TAIRhr, file = "micro csv input/TAIRhr.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(RHhr, file = "micro csv input/RHhr.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(WNhr, file = "micro csv input/WNhr.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(CLDhr, file = "micro csv input/CLDhr.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(SOLRhr, file = "micro csv input/SOLRhr.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(RAINhr, file = "micro csv input/RAINhr.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(ZENhr, file = "micro csv input/ZENhr.csv",
  sep = ",", col.names = NA, qmethod = "double")
write.table(IRDhr, file = "micro csv input/IRDhr.csv",
  sep = ",", col.names = NA, qmethod = "double")
}
if (is.numeric(loc[1])) {
  location <- paste("long", loc[1], "lat", loc[2])
}
else {
  location <- loc
}
cat(paste("running microclimate model for", ndays,
  "days from", dstart, " to ", dfinish, " at site ",
  location, "\n"))
ptm <- proc.time()
microut <- microclimate(micro)
print(proc.time() - ptm)
metout <- microut$metout
shadmet <- microut$shadmet
soil <- microut$soil
shadsoil <- microut$shadsoil
if (runmoist == 1) {
  soilmoist <- microut$soilmoist
  shadmoist <- microut$shadmoist
  humid <- microut$humid
  shadhumid <- microut$shadhumid
  soilpot <- microut$soilpot
  shadpot <- microut$shadpot
  plant <- microut$plant

```

```

    shadplant <- microut$shadplant
  }
  else {
    soilpot <- soil
    soilmoist <- soil
    shadpot <- soil
    shadmoist <- soil
    humid <- soil
    shadhumid <- soil
    plant <- cbind(soil, soil[, 3:4])
    shadplant <- cbind(soil, soil[, 3:4])
    soilpot[, 3:12] <- 0
    soilmoist[, 3:12] <- 0.5
    shadpot[, 3:12] <- 0
    shadmoist[, 3:12] <- 0.5
    humid[, 3:12] <- 0.99
    shadhumid[, 3:12] <- 0.99
    plant[, 3:14] <- 0
    shadplant[, 3:14] <- 0
  }
  if (snowmodel == 1) {
    sunsnow <- microut$sunsnow
    shdsnow <- microut$shdsnow
  }
  if (max(metout[, 1] == 0)) {
    cat("ERROR: the model crashed - try a different error tolerance (ERR) or a different spacing in DEP",
        "\n")
  }
  dates <- seq(as.POSIXct(paste0("01/01/", ystart),
    format = "%d/%m/%Y", tz = "America/Los_Angeles"),
    as.POSIXct(paste0("31/12/", yfinish), format = "%d/%m/%Y",
    tz = "America/Los_Angeles"), by = "hours")
  dates2 <- seq(as.POSIXct(paste0("01/01/", ystart),
    format = "%d/%m/%Y", tz = "America/Los_Angeles"),
    as.POSIXct(paste0("31/12/", yfinish), format = "%d/%m/%Y",
    tz = "America/Los_Angeles"), by = "days")
  if (lamb == 1) {
    drlam <- as.data.frame(microut$drlam)
    drrlam <- as.data.frame(microut$drrlam)
    srlam <- as.data.frame(microut$srlam)
    if (snowmodel == 1) {
      return(list(soil = soil, shadsoil = shadsoil,
        metout = metout, shadmet = shadmet, soilmoist = soilmoist,
        shadmoist = shadmoist, humid = humid, shadhumid = shadhumid,
        soilpot = soilpot, shadpot = shadpot, sunsnow = sunsnow,
        shdsnow = shdsnow, plant = plant, shadplant = shadplant,
        RAINFALL = RAINFALL, ndays = ndays, elev = ALTT,
        REFL = REFL[1], MAXSHADES = MAXSHADES, longlat = c(x[1],
        x[2]), nyears = nyears, minshade = minshade,
        maxshade = maxshade, DEP = DEP, drlam = drlam,
        drrlam = drrlam, srlam = srlam, dates = dates,
        dates2 = dates2))
    }
  }
  else {
    return(list(soil = soil, shadsoil = shadsoil,
      metout = metout, shadmet = shadmet, soilmoist = soilmoist,

```

```

        shadmoist = shadmoist, humid = humid, shadhumid = shadhumid,
        soilpot = soilpot, shadpot = shadpot, plant = plant,
        shadplant = shadplant, RAINFALL = RAINFALL,
        ndays = ndays, elev = ALTT, REFL = REFL[1],
        MAXSHADES = MAXSHADES, longlat = c(x[1],
        x[2]), nyears = nyears, minshade = minshade,
        maxshade = maxshade, DEP = DEP, drlam = drlam,
        drrlam = drrlam, srlam = srlam, dates = dates,
        dates2 = dates2))
    }
}
else {
  if (snowmodel == 1) {
    return(list(soil = soil, shadsoil = shadsoil,
    metout = metout, shadmet = shadmet, soilmoist = soilmoist,
    shadmoist = shadmoist, humid = humid, shadhumid = shadhumid,
    soilpot = soilpot, shadpot = shadpot, sunsnow = sunsnow,
    shdsnow = shdsnow, plant = plant, shadplant = shadplant,
    RAINFALL = RAINFALL, ndays = ndays, elev = ALTT,
    REFL = REFL[1], MAXSHADES = MAXSHADES, longlat = c(x[1],
    x[2]), nyears = nyears, minshade = minshade,
    maxshade = maxshade, DEP = DEP, dates = dates,
    dates2 = dates2))
  }
  else {
    return(list(soil = soil, shadsoil = shadsoil,
    metout = metout, shadmet = shadmet, soilmoist = soilmoist,
    shadmoist = shadmoist, humid = humid, shadhumid = shadhumid,
    soilpot = soilpot, shadpot = shadpot, plant = plant,
    shadplant = shadplant, RAINFALL = RAINFALL,
    ndays = ndays, elev = ALTT, REFL = REFL[1],
    MAXSHADES = MAXSHADES, longlat = c(x[1],
    x[2]), nyears = nyears, minshade = minshade,
    maxshade = maxshade, DEP = DEP, dates = dates,
    dates2 = dates2))
  }
}
}
}
} #end

```