DEVELOPING MACHINE LEARNING AND DEEP LEARNING SOIL MOISTURE MODELS FOR PRECISION AGRICULTURAL APPLICATIONS

A Thesis
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Xiaomo Zhang

In Partial Fulfillment of the Requirements
for the Degree of
MASTER OF SCIENCE

Major Program:
Natural Resources Management

June 2023

Fargo, North Dakota

North Dakota State University
Graduate School

Title

DEVELOPING MACHINE LEARNING AND DEEP LEARNING SOIL MOISTURE MODELS FOR PRECISION AGRICULTURAL APPLICATIONS

By

Xiaomo Zhang

The Supervisory Committee certifies that this ***disquisition*** complies with North Dakota State University's regulations and meets the accepted standards for the degree of

MASTER OF SCIENCE

SUPERVISORY COMMITTEE:

Zhulu Lin
Chair

Xin (Rex) Sun
Co-chair

Thomas DeSutter

Approved:

June 9, 2023
Date

Leon Schumacher
Department Chair

ABSTRACT

Monitoring soil moisture is increasingly becoming a research focus in the fields of agriculture, hydrology, meteorology, and ecology. While soil moisture measurements at points (<1 m^2) and its estimation at larger scales (100-25,000 km^2) have improved considerably, soil moisture modeling at the intermediate scales (10 to 100 m^2) needs more attention. In this study, machine learning and deep learning models including multi-linear regression (MLR), support vector machine (SVM), Gaussian process regression (GPR), and convolutional neural networks (CNN) were built and compared for soil moisture predictions at different depths at the weather stations in the Red River Valley using locations, meteorological data and soil physical properties. The results showed that the GPR (R^2 =0.80-0.90) outperformed other models including MLR (R^2 =68-0.82), SVM (R^2 =0.44-0.60), and CNN (R^2 =0.66-0.84) for soil moisture prediction. The prediction performance in the topsoil was better than in the subsoils. The GPR outperformed SVM when both models used the same kernel functions and kernel parameters.

ACKNOWLEDGMENTS

Firstly, I would like to give my heartfelt thanks to my committee chair, Dr. Zhulu Lin for his professionalism and continuous patience to lead me to the gateway to scientific research, for his encouragement and amiable care to help me to adapt rapidly to the study environment and life in a foreign country, and my other committee members, Dr. Xin (Rex) Sun and Dr. Thomas DeSutter for attentively addressing all my minor confusions using their knowledge during the thesis writing process. I also express my appreciation to Jithin Mathew and Sunil Gc for their assistance in machine learning and deep learning models with this project, and NDAWN staff Barb Mullins for soil moisture data. Thanks for the grant supports by the U.S. Department of Agriculture, agreement number 58-6064-8-023.Without their support, my thesis would not have been finished successfully.

In addition, special appreciation for my family across the ocean, far away from China who sustains my courage to study in the U.S. alone with unfailing love and unwavering support. I am extremely grateful to my mother who was always accompanying me when I was anxious late at night and inspired me to go forward. Furthermore, many thanks go to my classmates in the Department of Agricultural and Biosystems Engineering and my friends, especially Kewen, for their comprehension, inclusion, and accompaniment. Every person I met in Fargo made the place less freezing.

Finally, I would like to thank myself for my efforts and perseverance. I firmly believe that winter will eventually give way to spring, and flowers will bloom along the way. The present and the future will be equally wonderful.

DEDICATION

I would like to dedicate the most important person in my education and life, my mother who encourages me to keep researching in the field I enjoy and specialize in. She also encourages me to dive into more areas to discover my true interests and strong support for my decision. All of these greatly contributed to the completion of my master's thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

°.....Longitude and Latitude

%..... Percentages

ARD..... Automatic Relevance Determination

ANN.....Artificial Neural Networks

℃..... Degrees Celsius

cc.....Cubic Centimeters

cm.....Centimeters

CNN..... Convolutional Neural Networks

DNN.....Deep Neural Networks

DOY.....Day of the Year

℉.....Degrees Fahrenheit

FNN..... Feed-forward Neural Networks

ft..... Feet

g.....Grams

GPR.....Gaussian Process

GPR.....Gaussian Process Regression

in..... Inches

km^2..... Square Kilometers

Ly..... Langley

m..... Meters

m^2..... Square Meters

MAE.....Mean Absolute Error

micro m..... Micrometers

MLR.....Multi-linear Regression

mph..... Miles per Hour

MSE..... Mean Squared Error

NDAWN..... North Dakota Agricultural Weather Network

NN.....Neural Networks

PET..... Precipitation

r^2..... Coefficient of Determination

RBF.....Radial Basis Function

RF.....Random Forest

RMSE.....Root Mean Squared Error

RNN..... Recurrent Neural Networks

RRVN..... Red River Valley of the North

sec..... Seconds

SVM.....Support Vector Machine

SVR.....Support Vector Regression

VWC..... Volumetric Water Content

LIST OF SYMBOLS

b.....Intercept in support vector regression

C.....Coefficients

i/n/N..... Numbers of observations

w.....Regression coefficients in support vector regression

X.....Original values of individual features

$X_1, \dots, X_n / X_i$,..... Independent variables or individual features

X'..... Standardized values of individual features

X_i / X_j..... Input data points

X_{max}..... Maximum value of X

X_{min}..... Minimum value of X

y.....Dependent variables or observation values

y_i.....Observation values

$\widehat{y_i}$.....Prediction values

$\bar{y}$.....Average of observation values

$\alpha_1, \dots, \alpha_m$.....Regression coefficients

β_0.....Constant term

$\beta_1, \dots, \beta_n$.....Regression coefficients

γ..... A positive parameter that controls the width of the kernel

σ..... Permittivity

σ_f..... Signal Variance

σ_l..... Length Scale

σ_{lm} .....Separate length scale for each predictor

ε.....Residuals, vertical distances from predicted values to hyperplane in support vector regression and be defined

ξ.....Residuals, vertical distances from true values to predicted values in support vector regression over the range of $[-\varepsilon, +\varepsilon]$

LIST OF APPENDIX TABLES

LIST OF APPENDIX FIGURES

1. INTRODUCTION

1.1. Background

Soil moisture is usually quantified as the average water content of a given volume of soil in hydrology, which is divided into surface soil moisture in the top 10 cm of soil and root zone soil moisture in the upper 10-200 cm (Ahmad et al., 2010). The overall quantity of various states of soil moisture is small, only around 0.005% of global water allocation (https://earthhow.com/how-much-water-is-on-earth/). However, soil moisture is one of the key variables in hydrology, climatology, and agriculture. It directly impacts evaporation and plant transpiration, which controls the exchange of water and energy between the land surface and atmosphere (Robinson et al., 2008). Since the spatial and temporal changes in soil moisture are closely related to regional dryness, it is critical to accurately measure soil moisture content (Ahmad et al., 2010). The sustainable development of agriculture is inextricably linked to soil water content, so it is important to study, measure and monitor soil water content for sowing, irrigation, and harvesting of crops (Muñoz-Carpena, 2004).

In the Red River Valley of the North (RRVN), rainfall and melting snow are the major sources of water for agriculture. Soil moisture has a crucial role in decision-making for farming activities such as crop selection, planting, weeding, and harvesting. Appropriate decision-making might result in high-quality crops with better yields. Soil moisture is very important before and after planting. High soil water content in the field might result in wet fields, plowing difficulties, and higher fuel consumption. These issues may be mitigated if we can promptly predict the soil water content in agricultural land.

The traditional measurement of soil moisture at a point applying volumetric and gravimetric methods using in situ sensors ($< 1\ m^2$) has greatly improved, while the measurement

at larger scales using the appropriate optical band with remote sensing technology (100-25,000 km^2) has recently been developed. There is a gap in the intermediate scale (10-100 m^2) for the soil moisture measurement, which is normally required when developing variable rate irrigation description maps (Tom Scherer, Personal Communication). This gap may be filled by applying mechanistic or empirical models for soil moisture prediction. While the mechanistic models require accurate presentations of many input variables including meteorology, groundwater, and soil properties, the traditional statistical models are not a reasonable reflection of natural processes because of the strict assumptions of linearity and additiveness (Clapcott et al., 2013; Ali et al., 2015). The machine learning and deep learning models may serve as an alternative.

1.2. Objectives

Machine learning and deep learning models have recently been developed to predict field soil moisture using meteorological data and soil moisture data measured at nearby weather stations in the RRVN (Acharya et al., 2021a). However, due to various financial and/or technological reasons, the soil moisture data measured at weather stations may not always be readily available. Our research objective is to develop machine learning and deep learning models to predict soil moisture at weather stations using meteorological data and soil physical properties data in surface soil and root zone. The data of the features and target variable are collected from the 29 weather stations in the NDAWN (North Dakota Agricultural Weather Network) in and around the RRVN. Specific objectives include:

(1) Develop machine learning and deep learning algorithms to model the soil moisture dynamics at weather stations using meteorological data and soil physical properties data in surface soil and root zone, including multiple linear regression (MLR),

support vector machine (SVM), Gaussian process regression (GPR), and convolutional neural networks (CNN).

(2) Compare effectiveness of different machine learning and deep learning models in predicting soil moisture and investigate the important features affecting soil moisture in surface soil and root zone, and provide valuable information for farmers and participants in the RRVN.

(3) Investigate the geographical patterns in spatial-tempol changes of soil moisture in 29 weather stations in the RRVN.

2. LITERATURE REVIEW

2.1. Factors Affecting Soil Moisture

2.1.1. Meteorology

Previous studies have studied the direct and indirect impacts on soil moisture due to meteorological factors. Evapotranspiration and precipitation affect soil moisture directly while solar radiation and temperature serve an indirect role (Rasheed et al., 2022). Evaporation is the leading source of soil moisture loss when the plants are in their early development stages whereas transpiration has a similar effect in the mature period of plants (Amooh & Bonsu, 2015). Atmospheric evaporation results from temperature, wind speed, radiation, and relative humidity (Amooh & Bonsu, 2015). Gao et al. estimated actual evapotranspiration over heterogeneous terrain, and found that solar radiation changed soil moisture via impacts on the release rates of plant litter and soil nutrients to adjust the ionic composition of soil solutions (Gao et al., 2011; Rasheed et al., 2022). Srivastava et al. concluded that solar radiation had the greatest impact on spatial soil moisture variability while the precipitation distribution played a non-negligible role in soil moisture variability (Srivastava et al., 2018). Lakshmi et al. believed surface temperature changed the result of soil moisture fluctuation (Lakshmi et al., 2003) and the reason for this is that the surface temperature affected the outgoing radiation, sensible, and ground heat fluxes, as well as the latent heat flux through evapotranspiration. An increase in surface temperature normally reduces soil moisture (Lakshmi et al., 2003).

In addition to the meteorological factors mentioned above affecting soil moisture, Grayson et al. divided factors affecting soil moisture spatial patterns into local controls and nonlocal controls (Grayson et al., 1997). Local controls are predominant in dry environments while nonlocal controls are dominant in wet states (Vereecken et al., 2014). Joshi and Mohanty

also identified soil properties and vegetation that were related to vertical flow and evapotranspiration as local controls, and topography that determined the lateral flow as nonlocal control (Joshi & Mohanty, 2010). All these factors are linked to temporal-spatial soil moisture dynamics and should be accounted for among the elements contributing to soil moisture in the field (Amooh & Bonsu, 2015). Complex soil moisture dynamics is not modulated by a single factor but by a variety of factors working collectively.

2.1.2. Topography

Topography is one of the major contributors which should be included in soil water content dynamics analysis, acting as the slope (Easton et al., 2017). Easton et al. expressed that more water in the soil moves laterally downslope with faster speeds in the steep fields than in the flat fields. Otherwise, the profile curvature and aspect were considered as controlling topographic factors for the catchment in spatial soil water variability (Moore et al., 1988). Zaslavsky and Sinai concluded the most critical parameter was profile curvature for soil moisture distribution (Zaslavsky & Sinai, 1981). When the curvature of the soil surface is high, there normally will be more soil moisture heterogeneity (Rasheed et al., 2022). In addition, the variation of aspect and slope drives the fluctuation in evaporation via determining potential insolation (Lee, 1978).

2.1.3. Soil Properties

Robinson et al. reviewed various soil physical properties used to determine soil moisture such as electrical conductivity at the watershed scale (Robinson et al., 2008). Moreover, evaporation that results in soil moisture loss varied based on soil texture and organic matter content (Amooh & Bonsu, 2015). Amooh & Bonsu discovered the higher the organic matter and clay content in the soil, the better soil water conservation. The different soil textures are

associated with different soil pore sizes. For instance, clay soil with micropores is convenient to store water between clay soil particles and eliminate air; sandy soil presents the opposite phenomenon with clay soil depending on the macro soil pores, which are fewer but bigger than clay pores, whereas the water-holding capacity of loamy soil locates between clay soil and sand soil (Amooh & Bonsu, 2015). Hence, loamy soil is suitable for agricultural production because it avoids waterlogging and guarantees ventilation. In addition, the reduction of particle size is frequently accompanied by a decrease in the decomposition rate of organic matter with greater evaporation rates, such that the soil water content is lower (Giardina & Ryan, 2000). Wanas explained that organic matter decomposition enhances the micropores of sandy soils leading to more soil moisture by decreasing large pore space (Wanas, 2002).

2.1.4. Vegetation and Land Use

The correlation between the existence of trees or crop residues above the ground and soil moisture is frequently covered in soil moisture research. Gao et al. explained that tree crowns with a fully closed canopy maintain evapotranspiration (Gao et al., 2011). Coenders-Gerrits et al. disclaimed that tree roots uptake soil water for transpiration while the crown canopies of trees intercept rainfall into the soil, disturb soil evaporation, and impact stem flow (Coenders-Gerrits et al., 2013; Gwak & Kim, 2017). All of these tree-related processes change soil water content and lead to soil moisture redistribution.

Soil moisture dynamics were dependent on distinct land use types, such as livestock grazing mentioned in Zhao et al.'s research, in addition to the factors mentioned above owing to the variability of soil conditions and vegetation cover (Zhao et al., 2011). Zhao et al. discovered soil moisture was lower in grazed fields compared to in un-grazed fields throughout the year with or without vegetation. They found that the evaporation process was very strong in

vegetation-free areas (grazed fields) created by livestock activities whilst the process of infiltration to soil was relatively poor.

2.2. Soil Moisture Measurement Methods

Field methods for soil water content measurements are divided into direct and indirect methods. Thermo-gravimetric and thermo-volumetric are the most common direct methods. Similarly, some indirect methods are used to monitor soil water content, including volumetric and tensiometry methods such as neutron moderation, dielectric methods, time-domain reflectometry, frequency domain, amplitude domain reflectometry, phase transmission, time-domain transmission, tensiometer, and resistance blocks (Muñoz-Carpena, 2004). In addition, remote sensing methods emerge in these years for surface soil moisture estimation.

Remote sensing methods are always indirect including microwave and multispectral remote sensing methods are also available for large-scale soil moisture measurements or estimation. Satellite soil moisture with coarse spatial resolution has spatial information gaps in large regional and global areas (Guevara & Vargas, 2019). Khedri et al. advocate polarimetric synthetic aperture radar (PolSAR) imaging as a powerful tool for soil moisture estimation (Khedri et al., 2017). Downscaling microwave remotely sensed soil moisture is gradually becoming one of the most effective ways to obtain spatially continuous soil moisture with fine resolution on a regional scale (Sun & Cui, 2021).

Another commonly used indirect measurement for soil moisture is the water content reflectometers, which also measure electrical conductivity, dielectric permittivity, and temperature. Their measurement accuracy depends on various contributors mainly in the installation process (Campbell Scientific, Inc., 2018). The closer to a parallel the two probe rods

are, the better the measuring accuracy. The insertion method reduces effectively air voids generation, while the probe is more sensitive to permittivity close to the rods.

Temperature variation greatly impacts the dielectric permittivity of water, which ranges from 88 to 64 when the temperature changes from 0 ℃ to 70 ℃. Water is the dominant factor of soil bulk dielectric permittivity. Topp et al. described the relationship between volumetric water content (VWC) and soil bulk dielectric permittivity as Eq.1 (Topp et al., 1980). Therefore, the change in soil bulk dielectric permittivity with temperature will affect the measurements of VWC. In general, the VWC values will be overestimated when the temperature is lower than 20 ℃ and underestimated when the temperature is higher than 20 ℃.

$$\mathrm{VWC} = -5.3 \times 0.01 + 2.92 \times 0.01 \times \sigma - 5.5 \times 0.0001 \times \sigma^2 + 4.3 \times 0.000001 \times \sigma^3 \quad (1)$$

Where σ is permittivity.

However, the influence of soil's physical properties, such as soil porosity on temperature change, is crucial but lacks comprehensive consideration for all soils. The Topp equation is not suitable for all soils, such as organic, clayed or fine textured soils, although it has a strong performance in mineral soils or inorganic soils (Majcher et al., 2021). The VWC will be underestimated in organic, volcanic, and fine-textured soils. Therefore, calibration is essential for measuring accuracy.

In conclusion, the selections of various methods rely on measurement accuracy, stability, the average cost per site, adaptability to different depths, whether calibration is required or not, scales, safety, and the difficulty of installing equipment. In general, indirect measurements are more often used for monitoring soil water content in the field compared to direct measurements.

2.3. Machine Learning and Deep Learning for Predicting Soil Moisture

Traditional statistics models that build regression functions for soil moisture estimation draw quick results with inputs measured using *in-situ* methods, including meteorological variables and soil physical property variables (Ali et al., 2015). However, the *in-situ* measurements of variables and targets are time-consuming and money-consuming. In addition, the traditional statistics models are not a reasonable reflection of natural processes because of the strict utilization of linear and additive modeling approaches (Clapcott et al., 2013).

Machine learning and deep learning modeling approaches are widely used in analyzing and predicting soil moisture under the influence of meteorological factors and soil physical properties. The benefit of machine learning and deep learning models is that they offer an opportunity to fit the non-linear complex data without prior strict assumptions and statistics background, which bridges the gap between traditional models and better reflects the most realistic functional relationship and natural rules (Ali et al., 2015). Ali et al. argued that machine learning and deep learning methods were more flexible than other traditional statistics models with a large capacity of inputs (Ali et al., 2015). A variety of machine learning and deep learning models have been used for predicting soil moisture, including MLR (Acharya et al., 2021a; Prakash et al., 2018), SVM (Zaman et al., 2012; Dubois et al., 2021), artificial neural networks (ANN) (Hassan-Esfahani et al., 2015; Achieng, 2019), deep neural network (DNN) (Achieng, 2019), recurrent neural network (RNN) (Prakash et al., 2018), random forest (RF) (Dubois et al., 2021), and regression tree-based algorithms (Liu et al., 2020).

Dubois et al. made short-term soil moisture forecasts up to 7 days for potato crop farming using machine learning approaches including neural network, RF, and SVM after selecting the suitable features (Dubois et al., 2021). The soil moisture data was measured by tensiometers in 3

depths including 20, 30, and 40 cm in the crop fields while the meteorological data was provided by weather stations nearby the crop fields (less than 5 km). Coopersmith et al. produced near-surface soil moisture estimates in the depth of 5 cm using the K-nearest-neighbors algorithmusing *in-situ* observations in the depth of 10 cm and antecedent precipitation (Coopersmith et al., 2016). Gill et al. applied SVM and ANN to predict soil moisture after 4 days and after a week using the previous soil moisture data and meteorological features (a day ago), and all of these features on the same day (Gill et al., 2006). The model results showed that the SVM models outperformed the ANN models in soil moisture forecast. Ahmad et al. compared various machine learning models' performances in soil moisture estimation using soil moisture data from remote sensing (Ahmad et al., 2010). The results showed the SVM outperformed ANN and MLR models.

In the RRVN, Acharya et al. applied several machine learning models including classification and regression trees (CART), random forest regression (RFR), boosted regression trees (BRT), MLR, support vector regression (SVR), and ANN for field soil moisture prediction (Acharya et al., 2021a). They took crop types in the field, soil moisture and weather variables measured at the nearby weather stations (within the range of 2000 m), and the distance between the field and the nearest weather statsion as features in the machine models. The distance between the weather stations and nearby crop fields was classified into six classes (0-100 m, 100-200 m, 200-400 m, 400-800 m, 800-1200 m, and 1200-2000 m). The results showed that the RFR and BRT models were the best algorithms. Moreover, the feature that had the highest correlation with soil moisture was soil moisture at the nearby weather stations while the 4-day cumulative rainfall and potential evaporation (PET), bulk density, and saturated hydraulic conductivity played an essential role as well.

Deep learning models include more than one hidden layer compared to machine learning models which only include one hidden layer. Specific deep learning neural network models include feed-forward neural networks (FNN), recurrent neural networks (RNN), convolutional neural networks (CNN), deconvolutional neural networks, and modular neural networks. Prakash et al. applied the RNN models and compared them with other machine learning models including MLR and SVR for soil moisture prediction for 1 day, 2 days, and 7 days ahead (Prakash et al., 2018). The results showed MLR had superior performance.

3. DATA AND MODEL DEVELOPMENT

3.1. Study Area

The RRVN is situated in North America covering parts of the north-central United States and central Canada and has one of the most fertile lands in the world (https://en.wikipedia.org/wiki/Red_River_Valley). The Red River of the North is about 885 kilometers originating from the confluence of the Bois de Sioux and Otter Tail rivers between North Dakota and Minnesota in the United States and flowing into Lake Winnipeg in Canada. The study area along the RRVN covered 19 weather stations in North Dakota and 10 weather stations in Minnesota (Figure 1). The latitude of the study area span from 46.06° to 48.88° N and the longitude span from -95.85° to -100.25° W.

The major areas in the RRVN are warm-summer humid continental climates based on the Köppen-Geiger climate classification, which have distinct four seasons. The summer is from warm to hot and humid while the winter is cold and windy (https://en.wikipedia.org/wiki/Climate_of_North_Dakota). The average annual air temperature is 5 ℃, and annual temperature generally varies from -17 ℃ to 28 ℃ and rarely below -24 ℃ and above 33 ℃. The 30-yr mean annual rainfall is 60cm and snowfall is 317 cm (Acharya et al., 2021b).

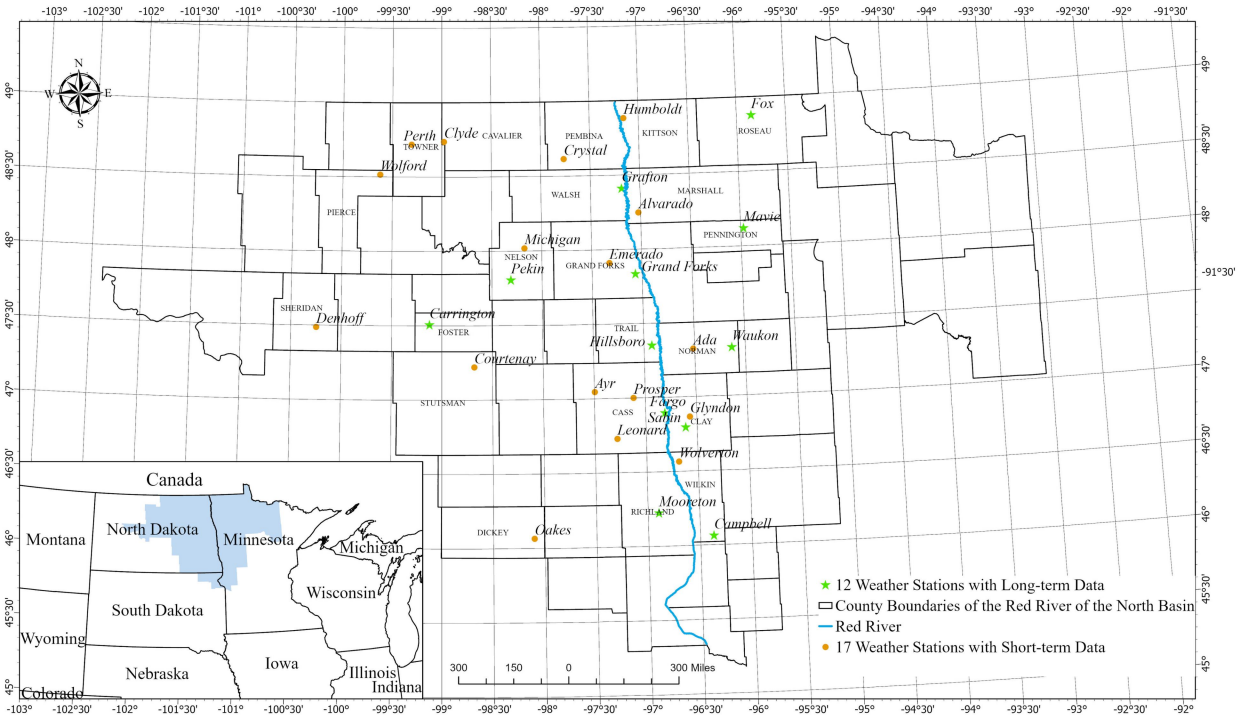

Figure 1. The locations of weather stations in the RRVN were used in this study.
Figure Note: Yellow dots represent 17 weather stations with short-term data while green dots represent 12 weather stations with long-term data. The 29 counties are marked by yellow borders. Weather station names are in italics. The inset shows the location of the entire range of the Red River of the North Basin, where the RRVN is the middle flat part of the basin.

3.2. Data

3.2.1. Data Collection

The soil moisture data and meteorological data were retrieved from the NDAWN https://ndawn.ndsu.nodak.edu/. There are 117 weather stations in NDAWN, including 83 weather stations in North Dakota, 28 weather stations in Minnesota, and 6 weather stations in Montana. Our study used 29 weather stations in and around the RRVN with hourly soil moisture measurement data. Nineteen (19) weather stations are located in 14 North Dakota counties, while 10 are located in 7 Minnesota counties (see Table 1). These weather stations have one to six growing seasons (April to October) of hourly soil moisture data measured at 5 cm, 10 cm, 20 cm, 50 cm, and 100 cm depths. The soil moisture data availability is shown in Table 1, but the soil

moisture measurements at 5 cm were fewer compared to other depths. The daily soil moisture data was calculated by taking the average of the hourly soil moisture.

Table 1. List of 29 weather stations and soil moisture data collection period

Weather Station	County	Soil Moisture Data Collection Period	Number of Growing Seasons
12 weather stations with long-term soil moisture measurement			
Fargo	Cass, ND	04/01/2016-10/31/2021	6
Grand Forks	Grand Forks, ND	04/01/2016-10/31/2021	6
Campbell	Wilkin, MN	04/01/2017-10/31/2021	5
Carrington	Foster, ND	04/08/2016-10/31/2021	6
Fox	Roseau, MN	05/16/2016-10/31/2021	6
Grafton	Walsh, ND	04/08/2017-10/31/2021	5
Hillsboro	Trail, ND	04/01/2017-10/31/2021	5
Mavie	Pennington, MN	04/01/2017-10/31/2021	5
Mooreton	Richland, ND	04/01/2016-10/31/2021	6
Pekin	Nelson, ND	04/01/2017-10/31/2021	5
Sabin	Clay, MN	04/01/2017-10/31/2021	5
Waukon	Norman, MN	04/01/2017-10/31/2021	5
17 weather stations with short-term soil moisture measurement			
Ada	Norman, MN	09/05/2021-10/31/2021	1
Alvarado	Marshall, MN	05/14/2021-10/31/2021	1
Ayr	Cass, ND	05/10/2021-10/31/2021	1
Clyde	Cavalier, ND	05/10/2021-10/31/2021	1
Courtenay	Stutsman, ND	07/25/2019-10/31/2021	3
Crystal	Pembina, ND	04/01/2021-10/31/2021	1
Denhoff	Sheridan, ND	06/20/2019-10/31/2021	3
Emerado	Grand Forks, ND	04/29/2021-10/31/2021	1
Glyndon	Clay, MN	05/11/2021-10/31/2021	1
Humboldt	Kittson, MN	09/24/2021-10/31/2021	1
Leonard	Cass, ND	09/16/2021-10/31/2021	1
Michigan	Nelson, ND	04/01/2021-10/31/2021	1
Oakes	Dickey, ND	08/05/2021-10/31/2021	1
Perth	Towner, ND	05/10/2021-10/31/2021	1
Prosper	Cass, ND	04/01/2021-10/31/2021	1
Wolford	Pierce, ND	05/10/2021-10/31/2021	1
Wolverton	Wilkin, MN	05/10/2021-10/31/2021	1

Soil volumetric water content in this research was monitored by CS655, which are the multiparameter smart sensors (Campbell Scientific, Inc., 2018). CS655 soil water content reflectometer is applied for VWC measurement indirectly. The soil bulk dielectric permittivity is measured within 3 inches (7.5 cm) diameter surrounding the sensor rods and 1.8 inches (4.5 cm) from the end of the rods via using CS655. Topp's equation (Eq. 1) takes advantage of the mathematical relations between dielectric permittivity and VWC to calculate soil moisture. All collected soil moisture data is displayed in percentage.

Meteorological data retrieved from NDAWN included daily average air temperature, average bare soil temperature, average turf soil temperature, average wind speed, total solar radiation, PET, and rainfall. Soil physical properties of the field where the individual NDAWN weather stations were manually retrieved from the Web Soil Survey website (https://websoilsurvey.sc.egov.usda.gov/App/HomePage.htm). The soil properties include sand content, clay content, moist bulk density, saturated hydraulic conductivity, available water capacity, and organic matter at different depths including 5 cm, 10 cm, 20 cm, 50 cm, and 100 cm.

The features that were used for developing the soil moisture machine learning and deep learning models are listed in Table 2. These features were selected using the feature selection process described below. They included one-time feature, two location features, eight soil features and 19 meteorological features. PET and rainfall from the previous 7 days were included due to their accumulation effects on soil water content (Gill et al., 2006; Acharya et al., 2021a). The slope and land use features were not included in this study.

Table 2. Features of machine learning and deep learning models.

Symbol	Feature	Description
	Location	
L1	Latitude	Latitude of the weather station (°)
L2	Longitude	Longitude of the weather station (°)
	Time	
DOY	Day of the year	The sequential day number starting with day 1 on January 1st
	Soil physical properties features	
S1	Sand content	The percentage content of sand (%)
S2	Clay content	The percentage content of clay (%)
S3	Bulk density	Weight of soil (oven dry at 105 °C) per unit volume (g/cc)
S4	Saturated hydraulic conductivity	The ease with which pores of a saturated soil transmit fluid (usually water) (micro m/sec)
S5	Available water capacity	The maximum amount of plant available water a soil can provide (in/in)
S6	Organic matter	The percentage content of organic matter (%)
S7	Average bare soil temperature	4-inch (10 cm) depth in bare soil (devoid of surface vegetation or cover) daily average bare soil temperature (°F)
S8	Average turf soil temperature	4-inch (10 cm) depth daily average turf soil temperature (°F)
	Meteorological features	
M1	Average air temperature	5 ft (1.52 m) above the soil surface daily average air temperature (°F)
M2	Average wind speed	10 ft (3 m) above the soil surface daily wind speed (mph)
M3	Total solar radiation	7 ft (2 m) above the soil surface daily solar radiation (Ly/day)
M4	Potential evaporation (PET)	PET calculated from daily values of solar radiation, dew point temperature, wind speed, and air temperature (inch)
M5	Rainfall	3 ft (1 m) above the soil surface daily rainfall (inch)
M6	PET 1 day ago	Daily PET value 1 day ago (inch)
M7	Rainfall 1 day ago	Daily rainfall value 1 day ago (inch)
M8	PET 2 days ago	Daily PET value 2 days ago (inch)
M9	Rainfall 2 days ago	Daily rainfall value 2 days ago (inch)

Table 2. Features of machine learning and deep learning models (continued).

Symbol	Feature	Description
M10	PET 3 days ago	Daily PET value 3 days ago (inch)
M11	Rainfall 3 days ago	Daily rainfall value 3 days ago (inch)
M12	PET 4 days ago	Daily PET value 4 days ago (inch)
M13	Rainfall 4 days ago	Daily rainfall value 4 days ago (inch)
M14	PET 5 days ago	Daily PET value 5 days ago (inch)
M15	Rainfall 5 days ago	Daily rainfall value 5 days ago (inch)
M16	PET 6 days ago	Daily PET value 6 days ago (inch)
M17	Rainfall 6 days ago	Daily rainfall value 6 days ago (inch)
M18	PET 7 days ago	Daily PET value 7 days ago (inch)
M19	Rainfall 7 days ago	Daily rainfall value 7 days ago (inch)

3.2.2. Data Preprocessing and Pretreatment

To ensure that all features and the target variables (i.e., soil moisture at different depths) had the same number of records, missing observations were identified by plotting raw data in scatter plots and then filled or removed accordingly (van den Berg et al., 2006). For soil moisture, missing data in each depth were calculated using Eq. 1 if the permittivity data were available at the station; otherwise, they were filled with the average of the nearest soil moisture data. For features, missing values were filled using the average of the nearest corresponding feature data.

After preprocessing, feature data were standardized to minimize the impact of differences in the order of magnitude of different features and their distributions. The differences between individual features could be over 1000 times. Normalization was used to scale the data to a small range, such as [0, 1] or [-1, 1], to convert the data into dimensionless numerical data. Min-max normalization was commonly used, as it did not make any assumptions about the distribution of data. The function used is shown below:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (2)$$

Where X is the original data of individual features; X_{min} is the minimum value of the X; X_{max} is the maximum value of the X; $X^{'}$ is the standardized value. All the features were normalized using Eq. 2, while soil moisture data were kept as original.

3.3. Model Development

3.3.1. Feature Selection

Soil moisture is a complex and dynamic variable that exhibits spatial-temporal variability (Coopersmith et al., 2016). To develop accurate machine learning and deep learning models for predicting soil moisture, it is important to consider a wide range of features, including meteorological conditions, topography, soil physical properties, human and animal activities, vegetation cover, time, and more.

One critical step in machine learning and deep learning model development is feature selection. This process helps to eliminate irrelevant or redundant features, improve model function, optimize performance, reduce dimensionality, improve running time, and gain a better understanding of the data structure (Chandrashekar & Sahin, 2014). It is important to note that the data and features themselves determine the upper limit of the model's predictive power, while the algorithms and models serve to approach to this limit.

There are three main types of feature selection methods: filter, wrapper, and embedding. Filter methods score each feature by dispersion or relevance, and set a threshold or number of thresholds to select features. Wrapper methods select or exclude features based on an objective function until the best subset is selected. Embedding methods utilize machine learning and deep learning algorithms to determine the weight coefficients of each feature, and select features based on these coefficients, typically from largest to smallest (Guyon & Elisseeff, 2003).

Embedding methods are similar to filter methods, but with training to determine the merit of the features.

In the literature, researchers usually employed several methods for feature selectin in their studies (Dubois et al., 2021; Zhang et al., 2021). In this study, five feature selection methods were applied to verify the importance of each feature in Python. These methods were tree models, random forest, logistic regression, F-value in the analysis of variance, and mutual information. The features that were selected by multiple feature selection methods were given higher priority in the model building process, as they may have stronger predictive power for soil moisture.

3.3.2. Multiple Linear Regression

MLR is the foundation of multivariable analysis and the entrance to understanding supervised machine learning. It is commonly used in wide research areas because of its simple operation and theoretical basis. MLR is to find the mathematical expression between independent variable (i.e., features) and dependent variable (i.e., target) even though there is no strict and deterministic functional relationship. It attempts to fit a line for all data and minimizes the sum of squares of deviation from the mean (Prakash et al., 2018).

The basic MLR model can be explained as:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n + \varepsilon \quad (3)$$

In Eq. 3, y is the dependent variable which is a random quantitative observation; x_1,…, x_n are independent variables; n is the number of observations; β_0 is a constant term; β_1,…, β_n are partial regression coefficients; ε is a random error, also known as residual, which is part of the change in y that cannot be explained by the independent variables and obeys the N (0, σ^2) distribution.

The `fitlm` algorithm in the MATLAB's Statistic and Machine Learning Toolbox (Mathworks, Boston, MA) was employed to implement the MLR model for soil moisture prediction. After trying several forms of MLR model, the interaction MLR model was found to have the best prediction performance over the validation dataset. The interaction MLR model is shown as below:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n + \alpha_1 x_1 x_2 + \alpha_2 x_1 x_3 + ... + \alpha_m x_{n-1} x_n + \varepsilon \quad (4)$$

All the symbols are defined as in Eq. 3. But Eq. 4 contains quadratic interaction terms such as $x_1 x_2$, $x_1 x_3$, etc.

3.3.3. Support Vector Machine

Support vector machines (SVMs) are a popular machine learning technique for classification and regression problems. They are based on the idea of finding a hyperplane that best separates the data points into different classes or predicts a continuous value. When it is used for regression is sometimes also called support vector regression (SVR) (Drucker et al., 1996). SVR aims to find a function that approximates the relationship between the input variables (or features) and a continuous target variable, while minimizing the prediction error. However, unlike linear regression based on ordinary least squares, SVR does not try to minimize the sum of squared errors, but rather a different loss function called epsilon (ε)-insensitive loss. This loss function ignores errors that are within a certain margin epsilon, and only penalizes errors that exceed this margin. This way, SVR can handle outliers and noise in the data better than linear regression (Smola and Schölkopf, 2004).

To find the optimal function (Eq. 5), SVR solves a convex optimization problem defined in Eq. (6-8) where C is a positive constant that controls the trade-off between the smoothness of

the function and the amount of errors (ξ) tolerated, ε is the margin width that defines the epsilon-insensitive loss.

$$y = wx + b \tag{5}$$

$$\frac{1}{2}\|w\|^2 + c\sum_{i=1}^{N}(\xi_i) \tag{6}$$

Subject to the constraints:

$$y_i - wx_i - b \leq \varepsilon + \xi_i \tag{7}$$

$$wx_i + b - y_i \leq \varepsilon + \xi_i \tag{8}$$

Kernels are an essential part of SVMs, as they allow us to deal with nonlinear and complex relationshiops by transforming the input data into a higher-dimesional space. Common kernel functions include linear kernel, polynomial kernels, and radial basis function (RBF) kernels. The `fitrsvm` algorithm in the MATLAB's Statistic and Machine Learning Toolbox (Mathworks, Boston, MA) was employed to implement the SVM algorithm for soil moisture prediction. After trying several kernels, the SVM model with a linear kernel was found to have the best prediction performance over the validation dataset.

SVR has many advantages over ordinary regression, such as robustness to outliers and noise, ability to handle high-dimensional and nonlinear data, and flexibility in choosing the loss function and kernel. However, SVR also has some disadvantages, such as sensitivity to parameter choices, computational complexity, and lack of interpretability.

3.3.4. Gaussian Process Regression

A Gaussian process is a stochastic process that is a collection of random variables indexed by time or space. A special property of Gaussian process is that any finite collection of these variables also follows a multivariate Gaussion distribution, therefore the Gaussion process is a distribution over an infinite number of variables or a distribution over a continuous function.

The inference of the continuous functions leads to Gaussian process regression (GPR) where the prior Gaussion process model is updated with training data to obtain a posterior Gaussian process distribution. Therefore, the GPR is a non-parametric Bayesian method for modeling a function as a collection of random variables, each with a Gaussion distribution. The mean and covariance of this distribution are specified by a mean function and a covariance function (or a kernel), respectively. The mean function represents the expected value of the function, and the kernel function encodes the similarity between different points in the input space.

To perform GPR, we need to specify a prior distribution over the function, which is usually a zero-mean Gaussian process with a chosen kernel. Given some observed data, we can then compute the posterior distribution over the function using Bayes' rule. The posterior distribution is also a Gaussian process, with a mean and covariance that depend on the observed data, the prior mean, and the prior kernel. The posterior mean gives the best estimate of the function at any new point, and the posterior variance gives the uncertainty around that estimate (Polykovskiy & Novikov, 2017).

Historically, GPR was used for the prediction of time series in the 1940's and it became popular in geostatistics in the 1970's where it is knowns kriging. Recently, the GPR has gained increasing attention in the area of machine learning boosted by rapidly increasing computation power (Rasmussen & Williams, 2006; Beckers, 2021). GPR is commonly used for low and small-sample regression problems, but are also extended for large samples and high-dimensional cases, which are more computationally expensive (Duvenaud, 2014).

The `fitrgp` algorithm in the MATLAB's Statistic and Machine Learning Toolbox (Mathworks, Boston, MA) was employed to implement the GPR algorithm for soil moisture prediction. After trying several kernels, the GPR model with an automatic relevance

determination (ARD) exponential kernel was found to have the best prediction performance over the validation dataset. The ARD exponential GPR defined in Eq (9) allows each predictor have a separate length-scale exponential kernel.

$$K(X_i, X_j) = \sigma_f{}^2 \exp\left(-\sqrt{\sum_{m=1}^{d} \frac{(X_{im} - X_{jm})^2}{\sigma_{lm}{}^2}}\right) \quad (9)$$

Where X_i and X_j are two input data points; σ_f is the signal standard deviation; σ_{lm} is a separate length scale for each predictor m, m = 1, 2, …, d. The larger the length scale, the flatter the regression function. If the length scale is large enough, the function will become a straight line. The introduction of the ARD kernel function in GPR makes the GP model naturally come with the ability of feature selection, which is one of the competitive advantages of the GP model compared to other machine learning models.

3.3.5. Comparisons between SVM and GPR

Since both SVM and GPR are kernel-based machine learning models and the former is generally regarded as the non-probabilistic analog of the latter (Rasmussen and Williams, 2006), we compared the performance of these two ML models in predicting soil moisture when both models taking the equivalent kernel functions. In the current study, the two equivalent kernel functions were the RBF for SVM (Eq. 10) and the squared exponential kernel function for GPR (Eq. 11). As shown in Eq. 10 and Eq. 11, when σ_f is equal to 1, the two kernels are identical, with the kernel parameters (σ in SVM and σ_l in GPR) set as 0.08.

$$G(X_i, X_j) = \exp\left(-\frac{1}{2\sigma^2}\left\|X_i - X_j\right\|^2\right) = \exp\left(-\gamma\left\|X_i - X_j\right\|^2\right) \quad (10)$$

$$K(X_i, X_j) = \sigma_f{}^2 \exp\left(-\frac{1}{2\sigma_l{}^2}\left\|X_i - X_j\right\|^2\right) \quad (11)$$

Where X_i and X_j are two input data points; $\|X_i - X_j\|$ is the Euclidean distance between them; γ is a positive parameter that controls the width of the kernel; σ_f is the signal standard

deviation; σ_l is the characteristic length scale. The larger the length scale, the smaller fluctuations in the regression curve fitted by GPR.

3.3.6. Convolutional Neural Network

The sequential model used in deep learning analysis is an abbreviated version of the functional model. It is designed to process sequential data, where the order of the input elements is important for tasks involving time series data. The structural order is end-to-end and has no bifurcation. Keras implements many layers including a core layer, a convolution layer, a pooling layer, etc. In this research, a sequential model was built using Keras from TensorFlow tools in python. There is a plain stack of layers in the sequential model. Each layer has one input tensor and one output tensor. A sequential model with 4 layers were constructed for the input data in this research. The shape information of the first layer of the sequential model is essential to be definite, the following layers can automatically derive the shape of the intermediate data. The input shape of the first layer was defined as the length of the training dataset, which is also the number of features. The batch sizes were set as 10, 10, 15, and 1 respectively for each layer in order. The first one layer is the input layer, the middle two layers are the hidden layers while the last one is the output layer. The number of the training steps for the model is 2000.

3.4. Model Training, Validation and Evaluation

All data in each depth are divided into 2 datasets: training and validation. The proportion of the 2 datasets is 70%:30%. The training dataset is used for model development while the validation dataset was used to compare model performances. The statistics used in this study for model evaluation include mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and coefficient of determination (r^2). RMSE and MAE are two commonly used loss functions in regression models due to the same order of magnitude. r^2 is an important

statistic reflecting the models' goodness of fit and represents the ratio of the regression sum of squares to the total sum of squares. The range of RMSE, MAE, and r^2 values are [0, +∞), [0, +∞), and [0, 1] respectively. The closer MSE and MAE are to 0, and the closer r^2 is to 1, the better the fitted regression equation.

The functions of calculating RMSE, MAE, and r^2 as follow:

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} = \sqrt{\mathrm{MSE}} \tag{12}$$

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i| \tag{13}$$

$$r^2 = \frac{\sum_{i=1}^{N}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} \tag{14}$$

In the equations above, N is the number of observations; the y_i is the observation values; the $\hat{y}_i$ is the prediction values; the $\bar{y}$ is the average of observation values.

4. RESULTS AND DISCUSSION

4.1. Feature Selection

Five different methods were applied for feature selection in this study. They included tree models, random forest, logistic regression, F-value in the analysis of variance, and mutual information. The feature selection results were presented in Fig. 2. The horizontal axes are different features and the vertical axes are the number of methods selecting that corresponding feature. The taller the vertical bar, which ranged from 0 to 5, meant that the feature was recommended by more methods. For instance, the value of 0 indicated that no feature selection method recommended this feature while the value of 5 meant that all five feature selection methods suggested this corresponding feature important. Figure 2 showed that the location (L1 & L2), time (DOY), and soil property (S1-S8) features were generally selected by more methods than the meteorological features (M1-M19). Especially, DOY (day of the year), S7 (average bare soil temperature), and S8 (average turf soil temperature) were selected by all five methods in various soil depths while S1 (sand content), S2 (clay content), and S4 (saturated hydraulic conductivity) were chosen by four feature selection methods and L1 (latitude), L2 (longitude), S6 (organic matters), M1 (average air temperature), and M2 (average wind speed) were chosen by three methods.

It is also interesting to note that the PET features (M4, M6, M8, M10, M12, M14, M16, and M18) were selected by more methods than the precipitation features (M5, M7, M9, M11, M13, M15, M17, and M19). Almost all PET features were selected by at least one method, while almost none of the precipitation features were chosen by any method, except that M7 (rainfall on the previous day) and M9 (rainfall on the previous two days) were chosen by at least one method for the soil moisture in top three soil depths. However, due to the effect of rainfall on soil

moisture (Acharya et al., 2021a), the rainfall data from the past 4 days were included in the dataset for the subsequent machine learning and deep learning model development. In conclusion, the features included in machine learning and deep learning model developments were location (L1 and L2), time (DOY), soil physical properties (S1-S8), and meteorological data (M1-M14, M16, and M18).

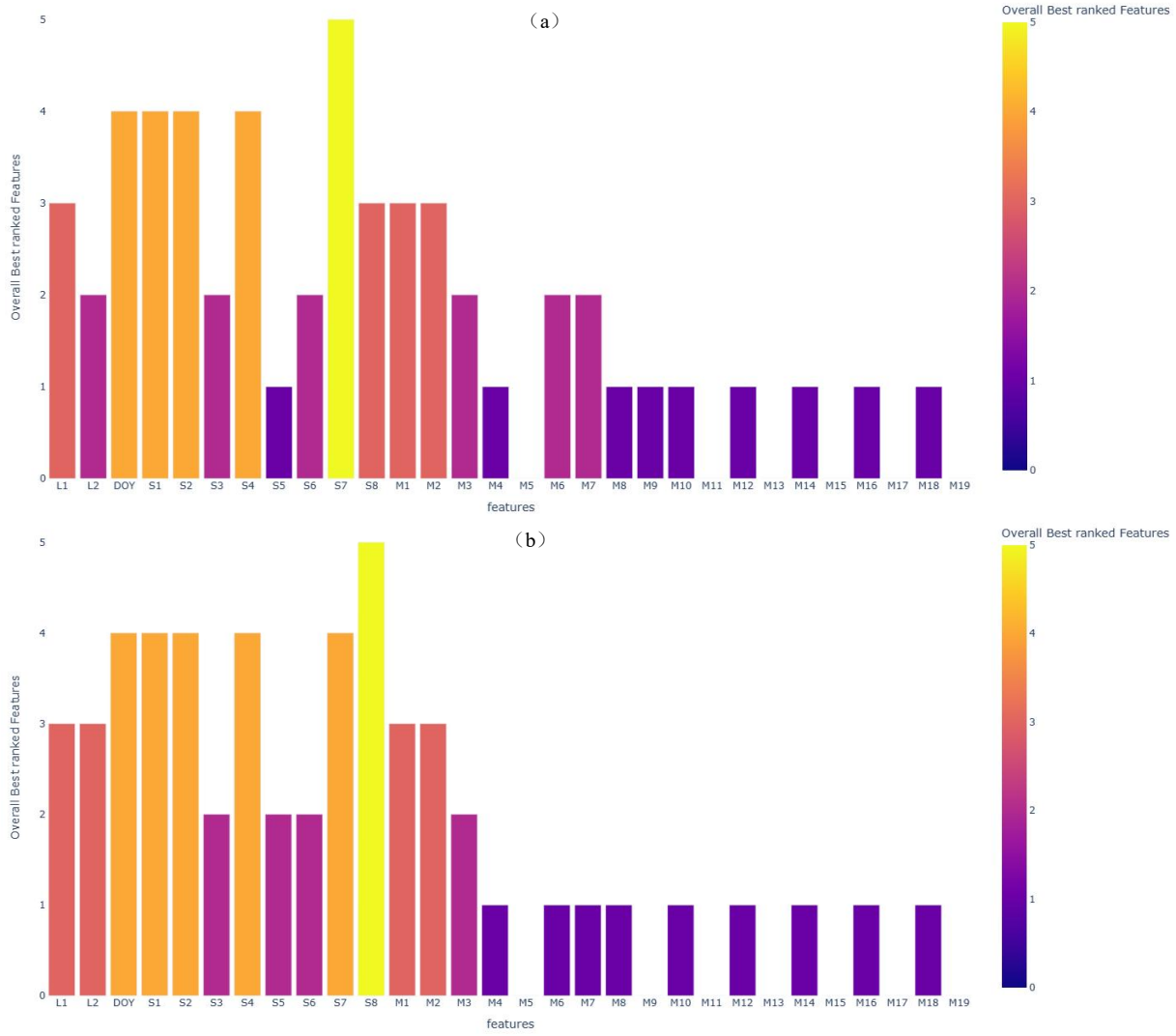

Figure 2. Graphical comparison of feature selection results from 5 feature selection methods at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e).

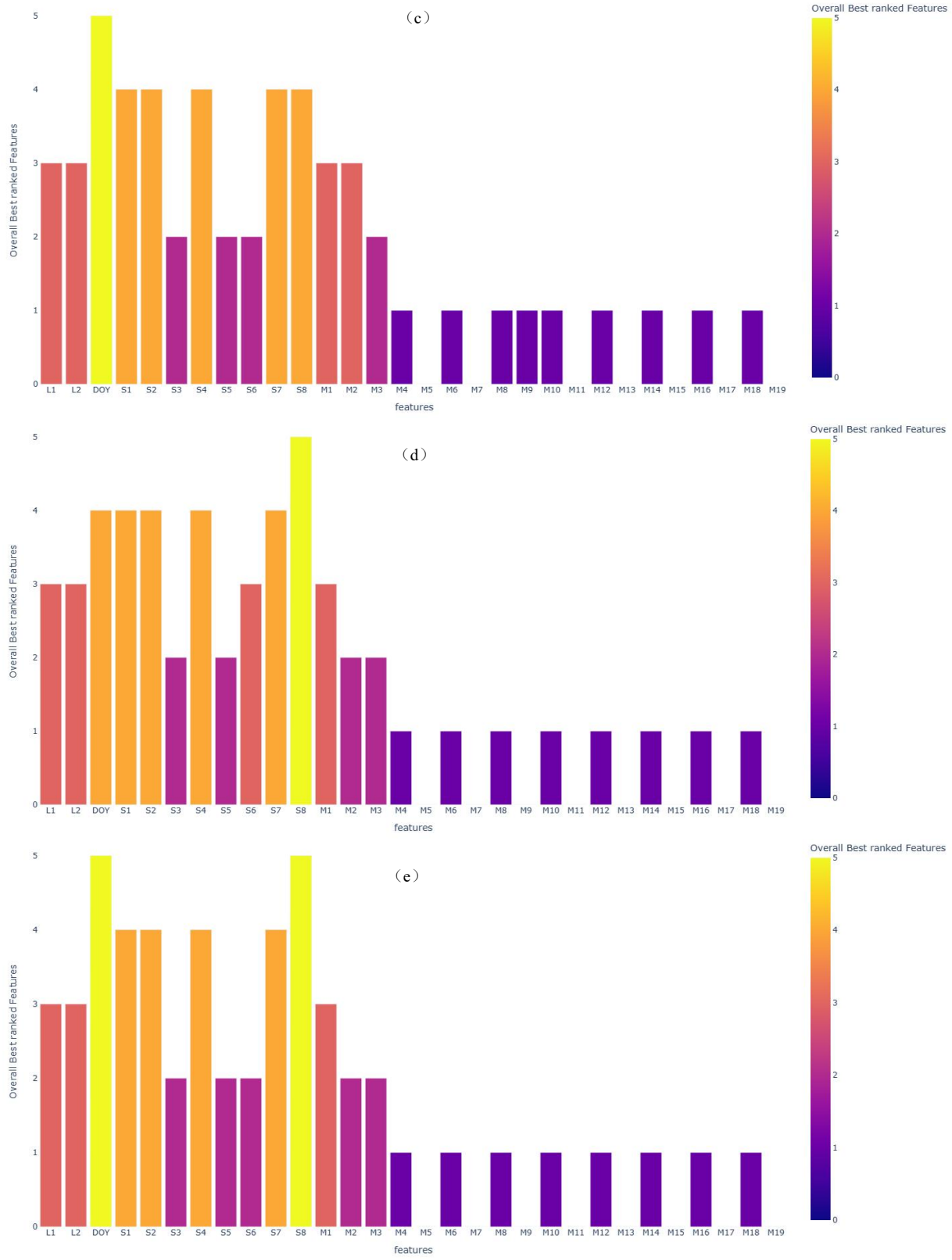

Figure 2. Graphical comparison of feature selection results from 5 feature selection methods at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) (continued).

4.2. Model Performances

4.2.1. All Model Comparisons

All data in each soil depth were randomly divided into the training datasets for model training and the validation datasets for model testing. The ratio of splitting the training datasets and the validation datasets were 70%: 30%. After each model was trained, its performance was calculated against the validation dataset. Table 3 compared four machine learning and deep learning models developed for the soil moisture at the five different soil depths in terms of r^2 values, RMSE, and MAE. These models include the interaction MLR model, the linear SVM, the ARD (automatic relevance determination) exponential GPR and CNN. It should be noted that the three machine learning models were the best-performing models in their respective categories. For deep learning, CNN was the sequential model using Keras. In addition, the comparison results of model performances were also presented graphically in Fig. 3 to Fig. 5 based on r^2 values, RMSE, and MAE, respectively.

In general, the order of the models' performance in predicting soil moisture at all five depth was: GPR>CNN>MLR>SVM. In terms of r^2 values, GPR ranged from 0.7895 to 0.9706, CNN from 0.6769 to 0.9534, MLR from 0.6835 to 0.9095, and SVM from 0.4582 to 0.6209. The reason for this possibility was that the introduction of the ARD kernel made the GPR model naturally come with the ability of feature selection, which is one of the competitive advantages of the GP model compared to other machine learning models. All four models generally did better in predicting soil moisture in topsoil (5 cm and 10 cm) than in subsoils (20 cm, 50 cm, and 100 cm), and all of them did worst in predicting soil moisture at 20 cm depth in terms of r^2. The possible reason might be the influence of both infiltration and evaporation was less significant r in the soil at the depth of 20 cm compared to the surface soil and root zone. It is worth noting

that the best model performance with a r^2 value of 0.97 was resulted from using ARD exponential GPR to predict soil moisture at the depth of 100 cm (Table 3).

Table 3. Comparison of different model testing performances in each soil depth based on r^2, RMSE and MAE. The algorithms include the interaction MLR model, the linear SVM, the ARD exponential GPR, and the sequential CNN.

Soil Depth	Algorithms	r^2	RMSE	MAE
5 cm	MLR	0.7770	0.0490	0.0365
	SVM	0.6092	0.0659	0.0501
	GPR	0.9106	0.0659	0.0501
	CNN	0.7936	0.0469	0.0349
10 cm	MLR	0.6971	0.0553	0.0426
	SVM	0.5309	0.0688	0.0540
	GPR	0.8072	0.0445	0.0329
	CNN	0.7211	0.0529	0.0405
20 cm	MLR	0.6835	0.0565	0.0434
	SVM	0.4582	0.0735	0.0565
	GPR	0.7895	0.0464	0.0338
	CNN	0.6769	0.0566	0.0426
50 cm	MLR	0.8125	0.0455	0.0336
	SVM	0.6115	0.0662	0.0494
	GPR	0.8833	0.0363	0.0256
	CNN	0.8173	0.0447	0.0336
100 cm	MLR	0.9095	0.0359	0.0202
	SVM	0.6209	0.0742	0.0437
	GPR	0.9706	0.0209	0.0130
	CNN	0.9534	0	0.0195

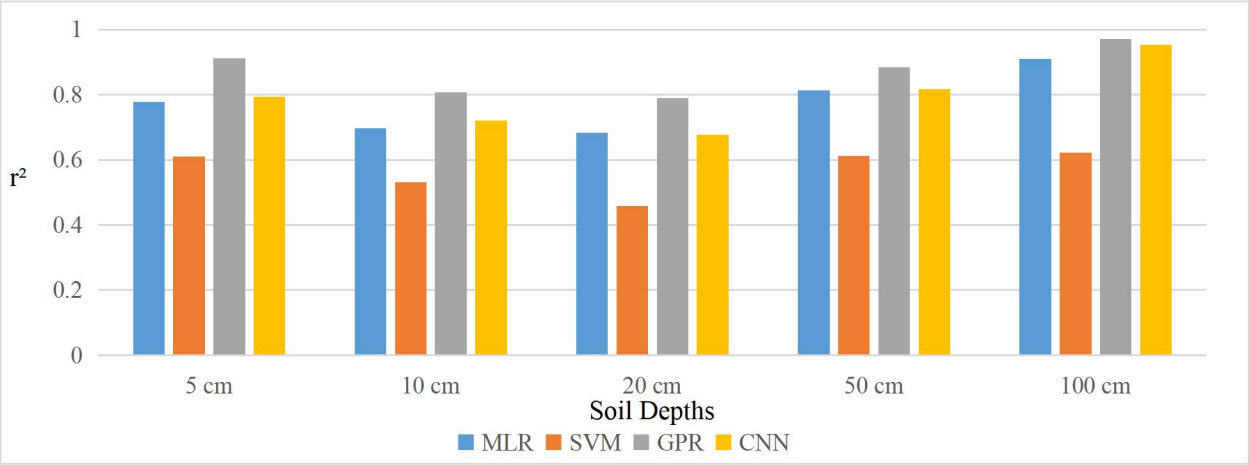

Figure 3. Comparison of interaction MLR, the linear SVM, the ARD exponential GPR, and the sequential CNN model testing performances in five depths based on r^2 values.

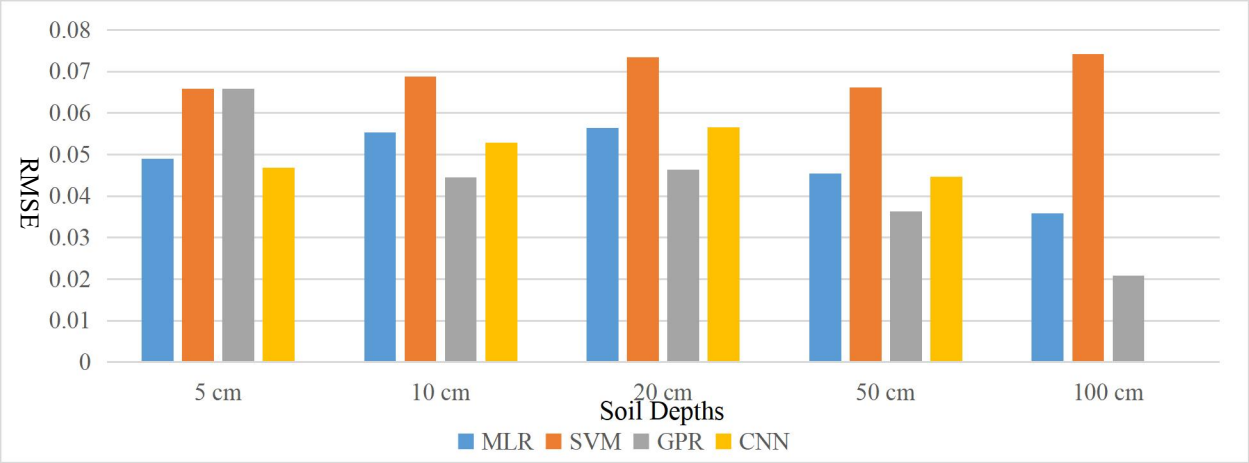

Figure 4. Comparison of interaction MLR, the linear SVM, the ARD exponential GPR, and the sequential CNN model testing performances in five depths based on RMSE values.

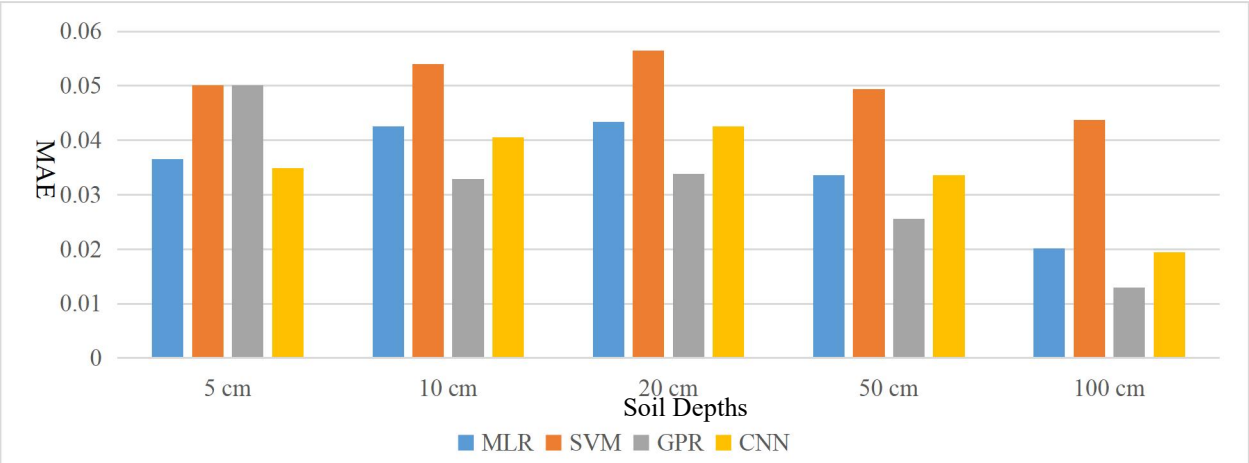

Figure 5. Comparison of MLR, SVM, GPR, and CNN model testing performances in five depths based on MAE values.

4.2.2. SVM and GPR Comparison

In order to compare model performances between SVM and GPR, the same kernel functions and kernel parameters were defined for both models. The kernel functions defined in SVM and in GPR were RBF and the squared exponential kernel function, respectively, as defined in Equations (10) and (11). Besides, the kernel parameter (i.e., σ in SVM and σ_l in GPR) was set as 0.08. The model performances of SVM and GPR were compared in Table 4.

In all soil depths, the GPR model performed better than the SVM model in terms of r^2, RMSE, and MAE. The range of r^2 for GPR was 0.60-0.74 while it was 0.46-0.60 for SVM. On average the r^2 values for GPR were 34.4 % greater than that for SVM in all soil depths. That was probably attributed to the GPR's capability of modeling continuous functions such as time series data well. It is interesting to note that the best model performances of both GPR and SVM occurred at the soil depth of 50 cm while the worst model performances occurred at 20 cm. The SVM model performance in 5 cm was not the best result among the SVM models in all soil depths. In addition, the effects of more infiltration and less evaporation compared to surface soil are possibly another main reason.

Table 4. Comparisons of model testing performances in each depth between RBF SVM and squared exponential GPR.

Soil Depth	Algorithms	r^2	RMSE	MAE
5 cm	SVM	0.4677	0.0751	0.0573
	GPR	0.7194	0.0547	0.0424
10 cm	SVM	0.4639	0.0746	0.0593
	GPR	0.6243	0.0614	0.0485
20 cm	SVM	0.4596	0.0731	0.0569
	GPR	0.5957	0.0643	0.0512
50 cm	SVM	0.5971	0.0668	0.0487
	GPR	0.7369	0.0537	0.0410
100 cm	SVM	0.4756	0.0942	0.0493
	GPR	0.6218	0.0787	0.0492

4.3. Soil Moisture Predictions

4.3.1. GPR Performance at Weather Stations

The overall best-performing machine learning model, the ADR GPR model, was applied to predict soil moistures at various soil depths at the 29 NDAWN weather stations in the RRVN region. The r^2 values of the GPR' performance at these weather stations were shown in Fig. 6. It is not surprising to notice that the GPR performed better in predicting the soil moisture in the topsoil (5 cm and 10 cm) than that in the subsoils (20 – 100 cm).

Fig. 6(a) showed that the GPR model did exceptionally well in predicting soil moisture at 5 cm where the r^2 values were greater than 0.9 at almost all the weather stations, except for three (i.e., Fargo, Grafton, and Ada). The number of stations where the r^2 values were greater than 0.9 decreased to 20 at the depth of 10 cm (Fig. 6(b)) and further decreased to 10-12 for the subsoils (Fig. 6(c-e)). Only in the soil depths of 20 cm and 50 cm via Fig. 6(c) and Fig. 6(d), there were r^2 values of several weather stations lower than 0.6. Overall, the model did less satisfactorily in predicting soil moisture at 20 cm and 50 cm in the RRVN.

However, the r^2 values were consistently greater than 0.9 across all soil depths at seven weather stations, including Wolford, Michigan, Prosper, Humboldt, Emerado, Wolverton, and Oakes. Ada was among the weather stations with worst r^2 values overall in each depth except the soil depth of 100 cm.

In general, the weather stations with worse r^2 values in the same soil depth distributed around the Red River, in the middle of RRVN.

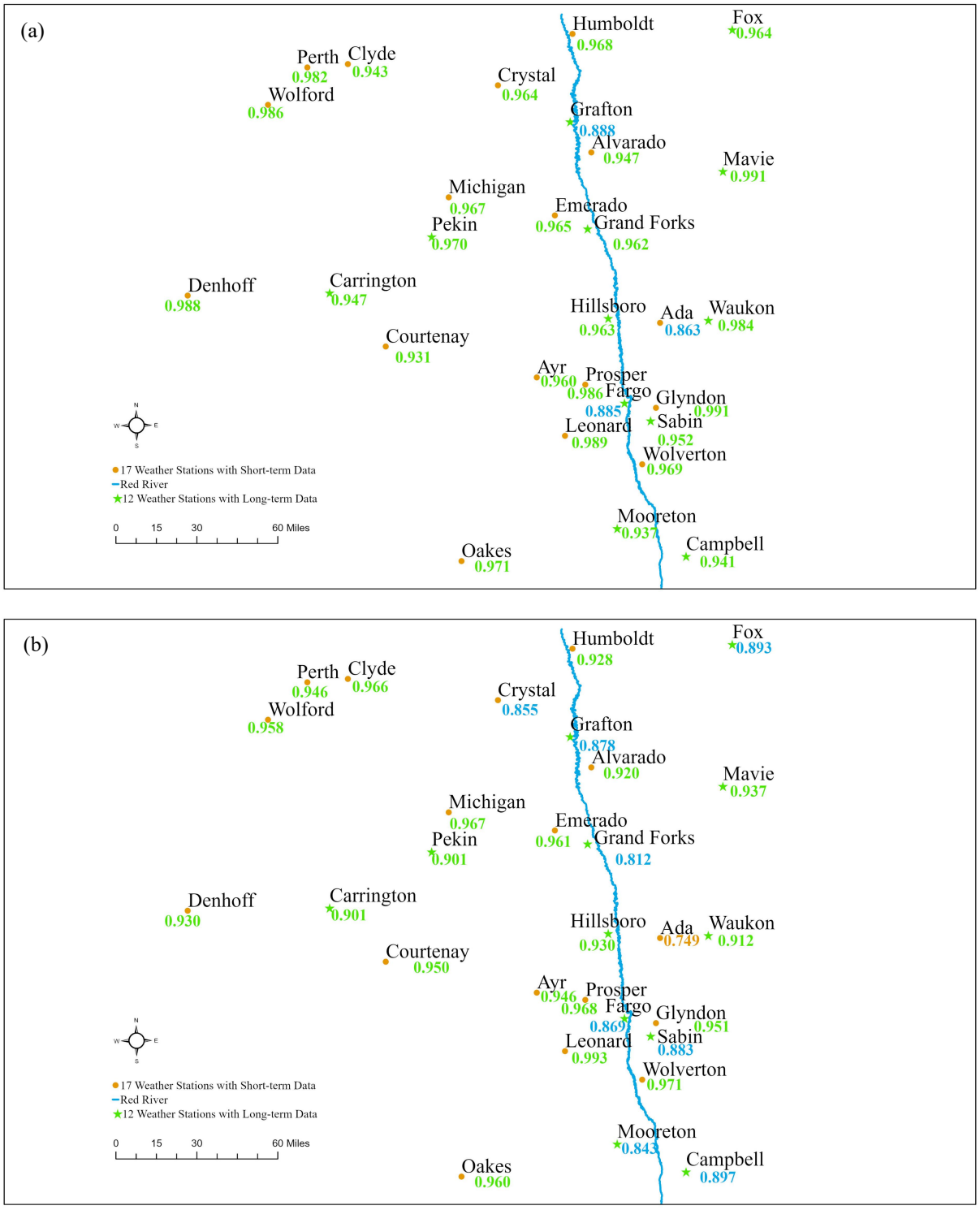

Figure 6. The ADR GPR model performance (r^2) in predicting soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in the Red River Valley of the North. Color codes for r^2 values: green [0.90, 1], blue [0.80, 0.90), orange [0.70, 0.80), pink [0.60, 0.70), and red [0, 0.60). ADR – automatic relevance determination, GPR – Gaussian process regression.

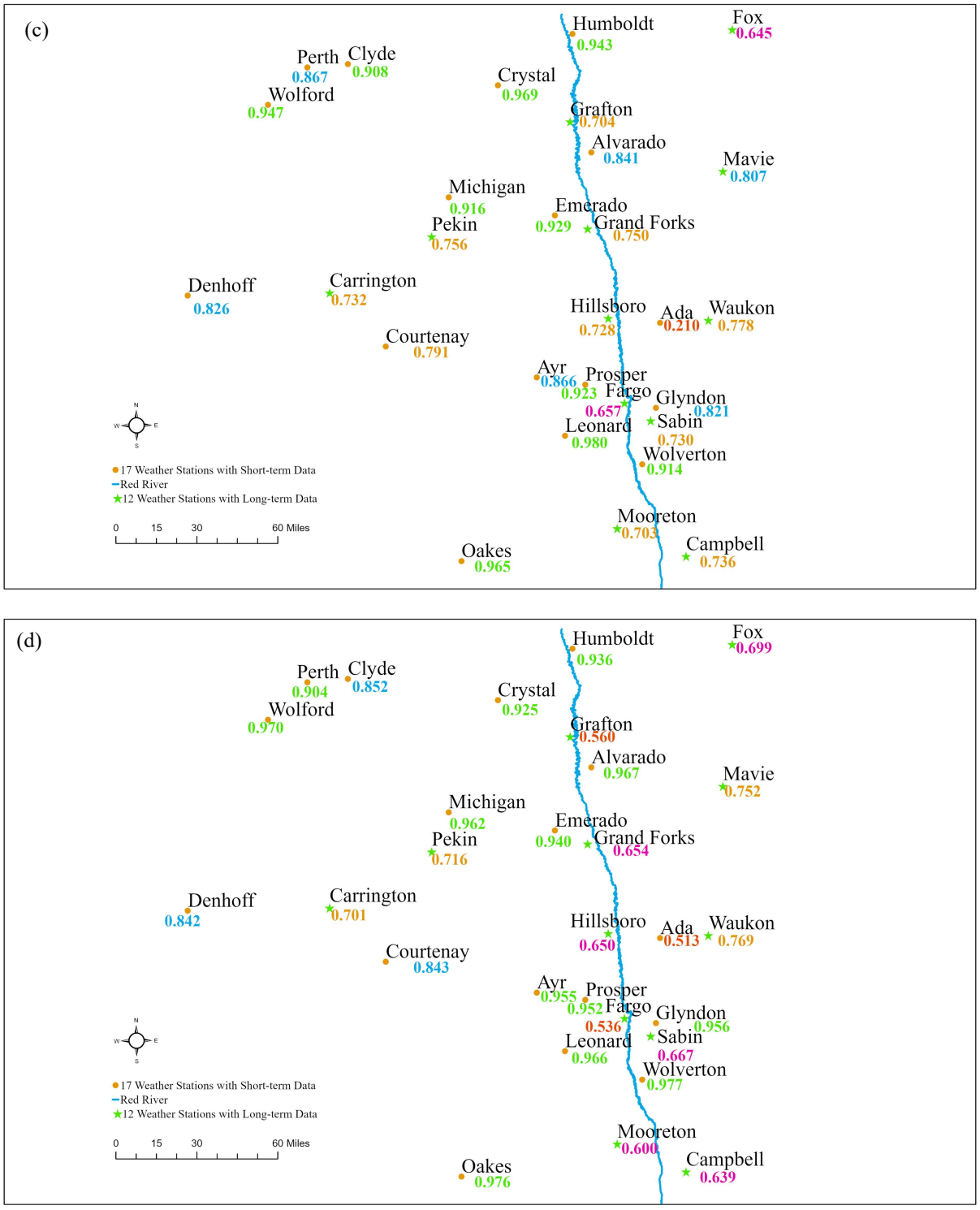

Figure 6. The ADR GPR model performance (r^2) in predicting soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in the Red River Valley of the North. Color codes for r^2 values: green [0.90, 1], blue [0.80, 0.90), orange [0.70, 0.80), pink [0.60, 0.70), and red [0, 0.60). ADR – automatic relevance determination, GPR – Gaussian process regression (continued).

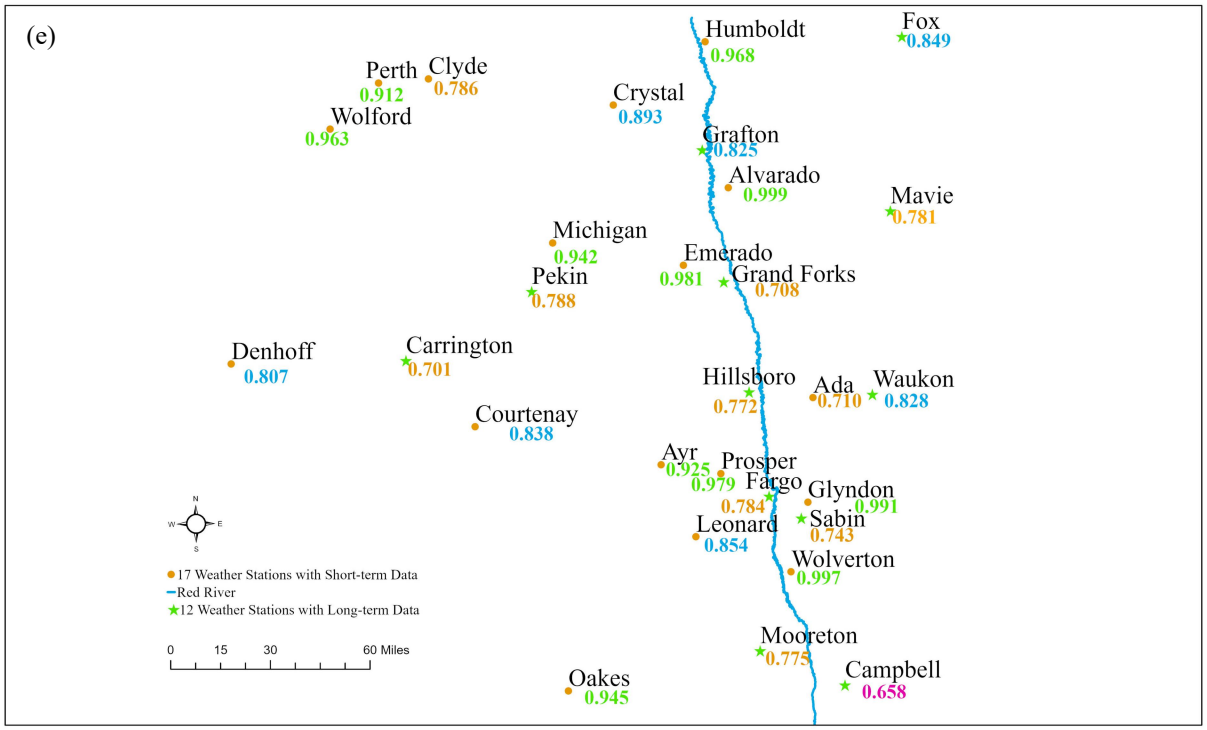

Figure 6. The ADR GPR model performance (r^2) in predicting soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in the Red River Valley of the North. Color codes for r^2 values: green [0.90, 1], blue [0.80, 0.90), orange [0.70, 0.80), pink [0.60, 0.70), and red [0, 0.60). ADR – automatic relevance determination, GPR – Gaussian process regression (continued).

The GPR's predictions of soil moisture at the Fargo (North Dakota) and Grand Forks (North Dakota) weather stations were shown in Fig. 7 and Fig.8 as examples. The Fargo and Grand Forks weather stations had two of the longest records of soil moisture observations. There were 6 growing seasons of soil moisture observations for the 10-100 cm depths (2016-2021) in both of these two stations. Also, there were two and a half growing seasons for the 5 cm depth (2019-2021) in Fargo, while there were one and a half growing seasons for the 5 cm soil depth (2020-2021) in Grand Forks.

Fig. 7(a) showed that the model did very well in predicting soil moisture at 5 cm in all growing seasons, except for the first two weeks in 2019 and the first 6 weeks in 2020 when the model underpredicted the soil moisture in Fargo. A similar performance can also be observed for

soil moisture prediction at 10 cm (Fig. 7(b)). The model's performance dropped considerably for predicting the soil moisture in the subsoils (Fig. 7(c-e)). This trend can also be observed by examining the 95% confidence regions of GPR's predictions. The 95% confidence regions at the 5 cm and 10 cm depths were narrower than those at the 20-100 cm depths (not fully shown), which indicates that the model felt less confident in making predictions for the soil moisture in the subsoils.

In general, the model prediction performances in Grand Forks were slightly better than that in Fargo based on the r^2 values, especially in the soil depth of 5 cm (Fig. 8(a)) with a r^2 value of 0.962 which is higher than all other soil depths in Grand Forks (r2 = 0.64-0.812) and all soil depths in Fargo (r^2 = 0.536-0.885). Similar to Fargo, the model performed better for the surface soil depths (5 cm and 10 cm) than for the subsoil depths (20 – 100 cm). in Grand Forks. The 95% confidence regions at the 20-100 cm soil depths were wider than they were at 5-10 cm, which means there were more confidence for the soil moisture predictions at top soils than that in the root zone.

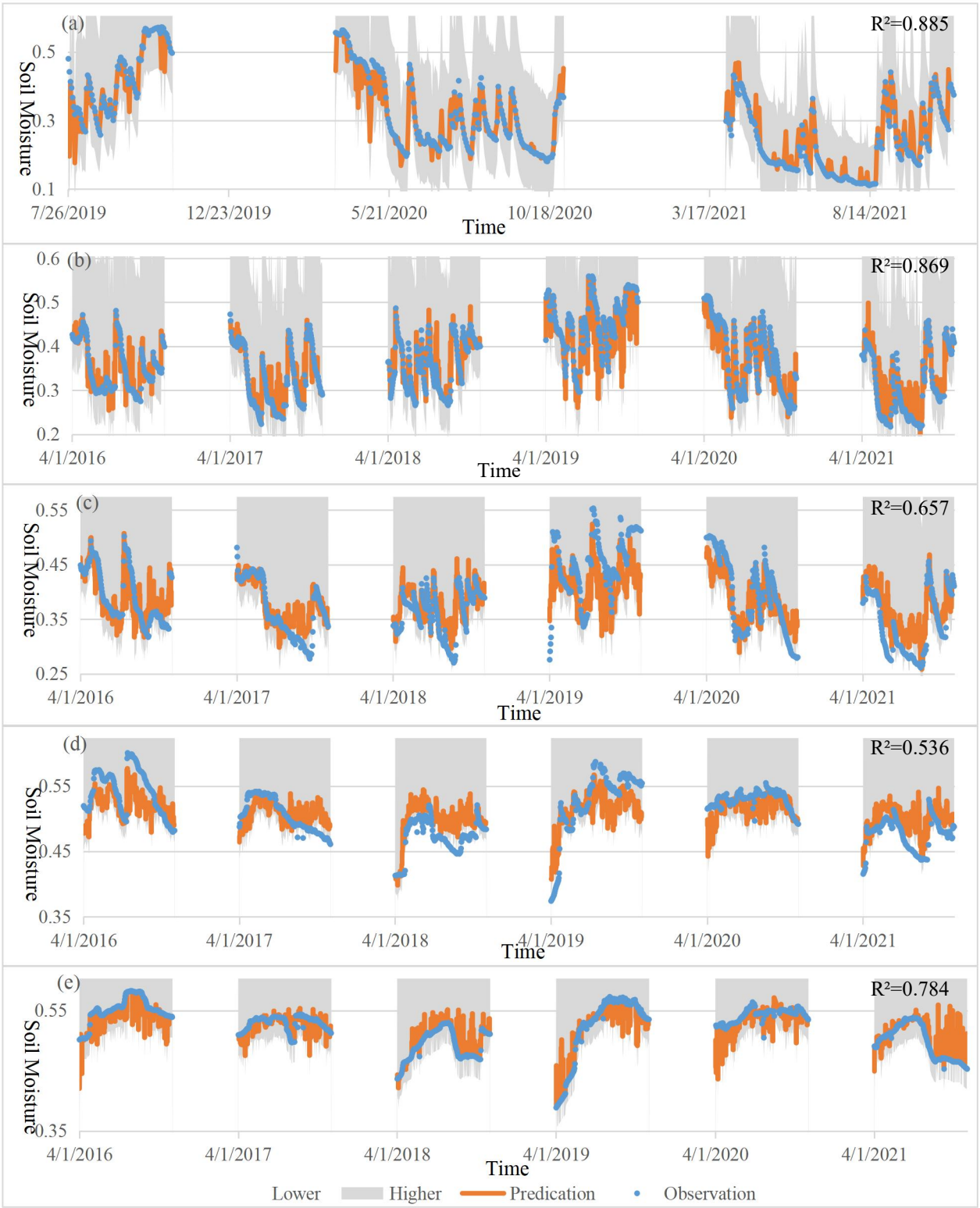

Figure 7. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Fargo. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence regions.

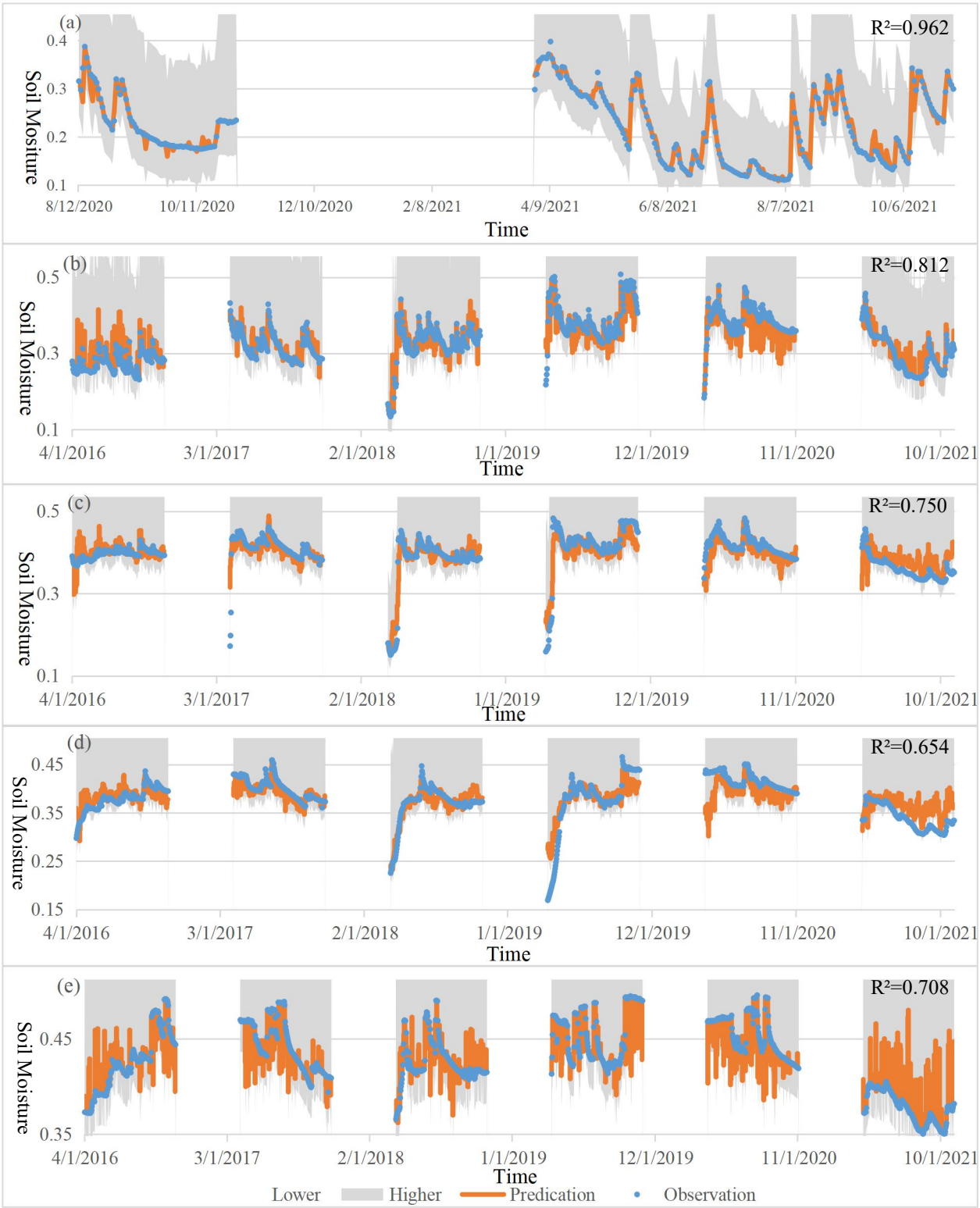

Figure 8. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Grand Forks. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

4.3.2. Effects of Soil Physical Properties and Meteorological Features on Soil Moisture Prediction

To evaluate the effects of soil physical properties and meteorological features on soil moisture predictions, we constructed two new GPR models. In the first model (Model A), the features included L1, L2, DOY, and soil properties (i.e., S1-S8), while in the second model (Model B), the features included L1, L2, DOY, and meteorological features (i.e., M1-M14, M16 and M18). Both Model A and Model B were compared against Model C, the best performing model (i.e., ADR GPR), which included L1, L2, DOY, soil properties (S1-S8) and meteorological features (M1-M14, M16, and M18). The comparison results are shown in Table 5 and Fig. 9.

It is interesting that Model B (with meteorological features) had a similar performance as Model C (a full model) in predicting soil moisture at the soil depths from 5 cm to 50 cm and its performance dropped significantly when predicting the soil moisture at 100 cm. Model B consistently performed better than Model A (with soil properties) at all depths from 5 cm to 50 cm, but its advantage over Model A decreased gradually as the soil depth increased until the depth of 100 cm when the performances of the two models reversed – Model A did better than Model B. This indicates that, collectively, the meteorological features were more important than the soil features in predicting soil moisture when the soil depths were less than 50 cm, but the trend reversed when the depth reached about 100 cm. It is also interesting to note that Model A's performance increased as the soil depth increased except for the surface soil at 5 cm where it performed slightly better than at the depths of 10 cm and 20 cm, while Model B's performance was not affected by the soil depths much with r^2 fluctuating around 0.8.

Table 5. Comparisons of model testing performances which include different features in each depth using ADR exponential GPR. Model (A) included features of L, DOY, and S. Model (B) included features of L, DOY, M1-M14, M16, and M18. Model (C) included features of L, DOY, S1-S8, M1-M7, M8, M10, M12, M14, M16 and M18.

Soil Depth	Model	r^2	RMSE	MAE
5 cm	A	0.6917	0.0561	0.0381
	B	0.9090	0.0311	0.0200
	C	0.8985	0.0336	0.0219
10 cm	A	0.5617	0.0667	0.0528
	B	0.7760	0.0474	0.0350
	C	0.7826	0.0471	0.0342
20 cm	A	0.6243	0.0618	0.0465
	B	0.7723	0.0477	0.0349
	C	0.7750	0.0474	0.0342
50 cm	A	0.8121	0.0463	0.0342
	B	0.8711	0.0386	0.0276
	C	0.8803	0.0370	0.0263
100 cm	A	0.8091	0.0562	0.0226
	B	0.7812	0.0602	0.0241
	C	0.9688	0.0217	0.0143

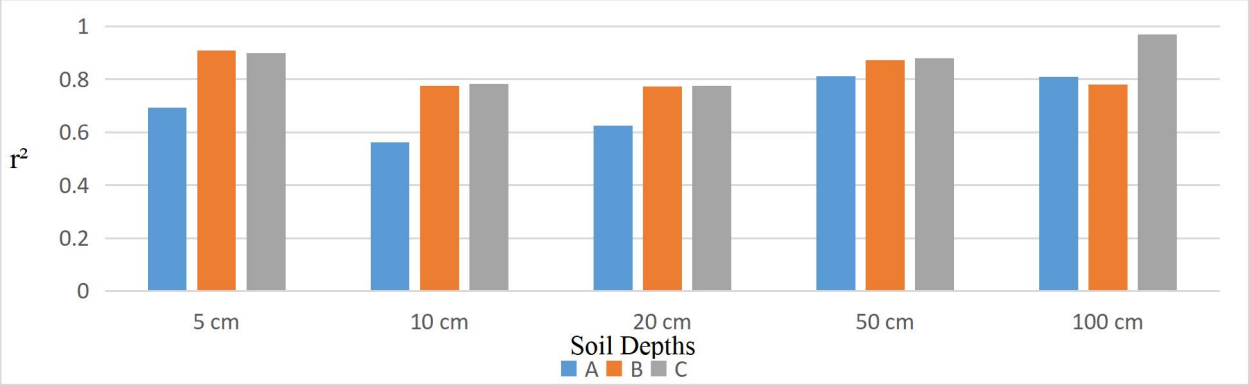

Figure 9. Comparison of ADR exponential GPR model testing performances including different features based on r^2 values. Model (A) included features of L, DOY, and S. Model (B) included features of L, DOY, M1-M14, M16, and M18. Model (C) included features of L, DOY, S1-S8, M1-M7, M8, M10, M12, M14, M16 and M18.

4.3.3. Effect of Individual Features on Soil Moisture Prediction

The correlation heat maps (Fig. A1 to Fig. A5) reveal that several soil physical properties and meteorological features, such as soil particle composition, available water content, wind

speed, PET, rainfall, were correlated with soil moisture contents in all soil depths. Fig. 10 showed the relationship between model's performance and soil physical properties in at the depth of 5cm in all 29 weather stations

There were no notable patterns between the soil's physical properties such as the percentage of sand content (Fig. 10(a)) or available water content (Fig. 10(b)) and the prediction performances of of the GPR model in the 29 weather stations.

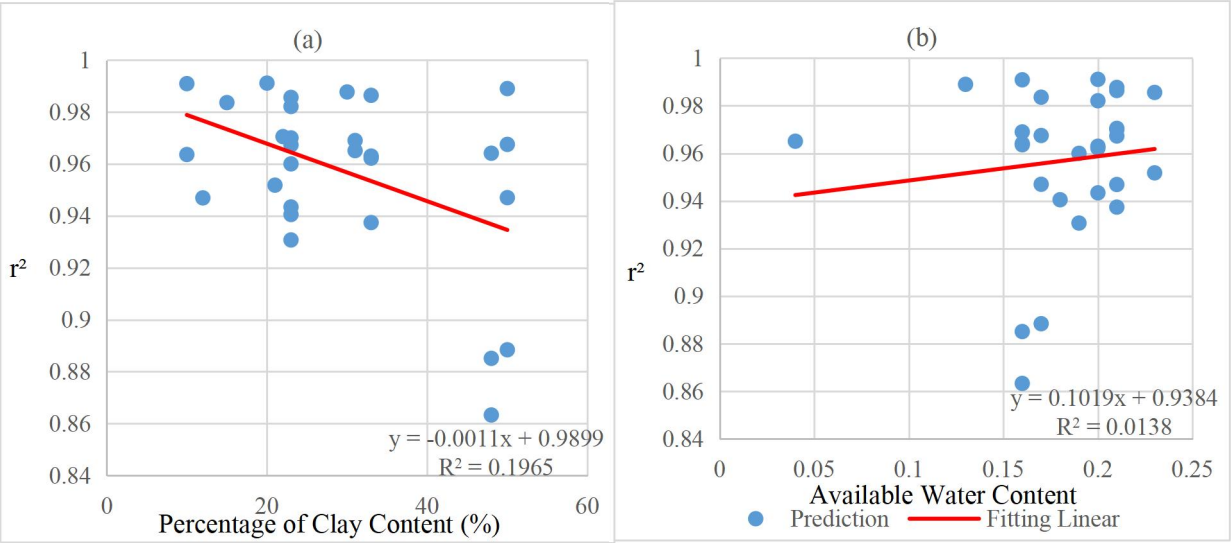

Figure 10. Scatter graphs of the correlation between the model prediction results and different soil physical properties based on r^2. The horizontal coordinate was the percentage of sand content in (a) while it was the available water content in (b). The vertical coordinate was the r^2 values predicted in the soil depth of 5 cm using the ADR exponential GPR model at 29 weather stations.

Fig. 11 showed the correlation between the soil moisture at 5 cm and average bare soil temperature, soil radiation, average wind speed, rainfall, and PET. There were some visible correlations between soil moisture and rainfall (Fig. 11(d)), and PET (Fig. 11(e)). However, the r^2 for the correlations for all features was below 0.1 except for the average bare soil temperature (r^2 =0.1043) in Fig. 11(a).

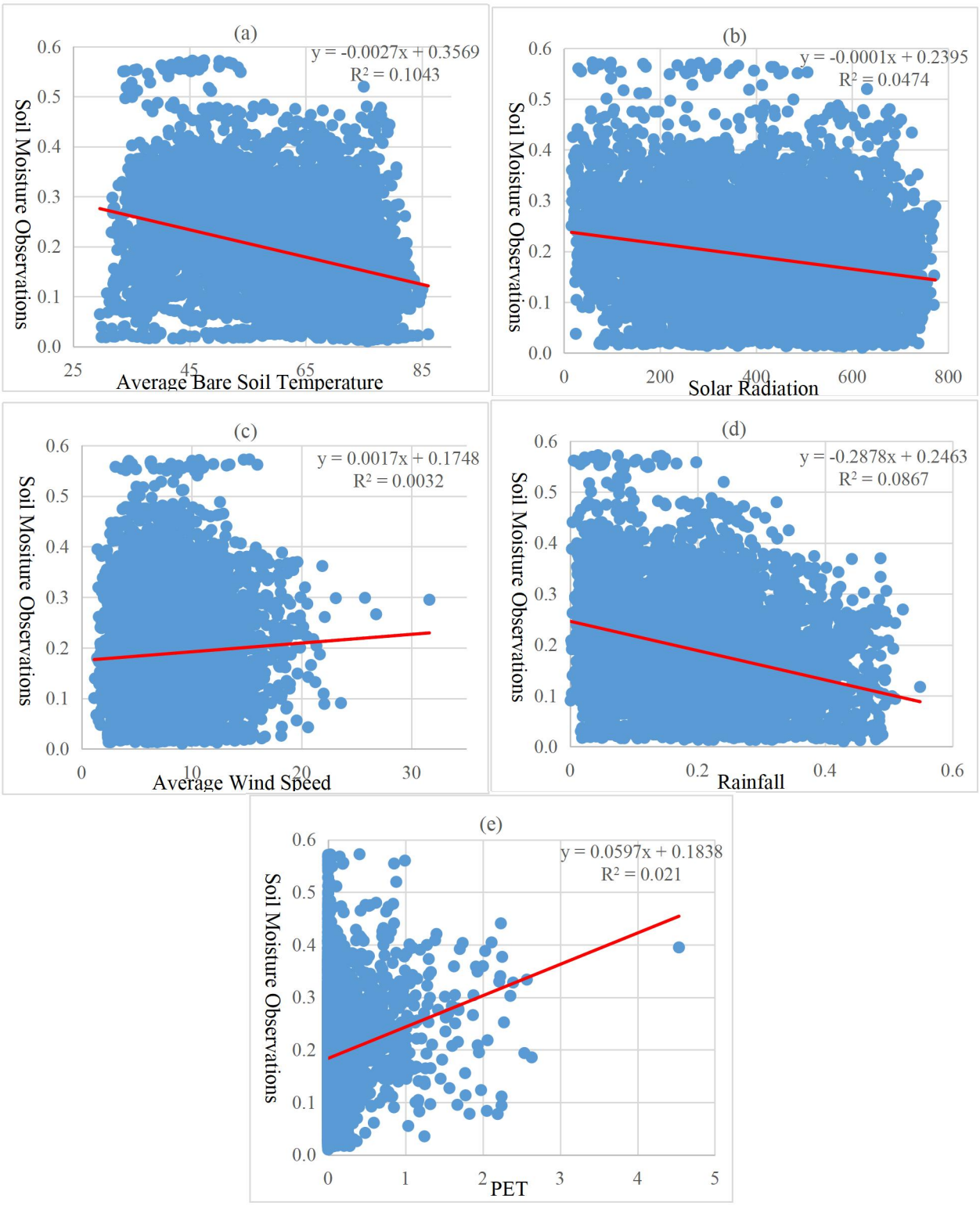

Figure 11. Scatter graphs of the correlation between the soil moisture observations and different features. The horizontal coordinate was the average bare soil temperature in (a), solar radiation in (b), average wind speed in (c), PET in (d), PET a day ago (e), rainfall in (f), and rainfall 1 day ago (g). The vertical coordinate was soil moisture observation values in the soil depth of 5 cm at all 29 weather stations.

5. CONCLUSIONS

It is important to examine the spatial-temporal dynamic change of soil moisture at intermediate scales using machine learning and deep learning algorithms for precision agricultural application in the RRVN. The MLR, SVM, GPR, and CNN models were developed to predict soil moisture at the depths of 5 cm, 10 cm, 20 cm, 50 cm, and 100 cm in the RRVN and its surrounding areas. All models' performances in predicting soil moisture in topsoil (0-10 cm) were better than in subsoils (20-100 cm). It might be because the surface soil is more susceptible to meteorological factors compared to the root zone. The GPR model (r^2= 0.7895 - 0.9706) outperformed the other three models (CNN: r^2= 0.6769 - 0.9534, MLR: r^2= 0.6835 - 0.9095, and SVM: r^2= 0.4582 - 0.6209) in almost all soil depths. The best r^2 value was 0.97, which was the result predicted by using ARD exponential GPR in a soil depth of 100 cm. The natural feature selection from the ARD kernel was probably one of the contributing factors to why it had more prominent model performance than other machine learning models. Besides, the potential reasons possibly were that at 100 cm the soil infiltration is higher and the evaporation in this soil layer was not too much compared to the surface soil.

When taking the equivalent kernel functions with the same values of kernel parameters, the r^2 values for GPR were on average 34.4% higher than SVM across all soil depths. To the best of our knowledge, this is the first time that GPR was applied for soil moisture prediction, and it shows that the GPR has a clear advantage over SVM in modeling soil moisture, especially in the RRVN region. The model building for time series data just like soil moisture might be more appropriate using GPR. With r^2= 0.863 - 0.991 (5 cm) and r^2= 0.749 – 0.993 (10 cm) at individual weather stations, our study demonstrated that the GPR model was capable of predicting soil moisture in the topsoil (0-10 cm) based on location, time, soil and meteorological

features that were readily available at the intermediate scales (10-100 m^2). Our research showed that Gaussian process regression is a promising tool for practitioners, researchers and policymakers to make soil moisture predictions for precision agricultural applications. We will continue to improve the GPR's performance in predicting root zone soil moisture in the future.

REFERENCES

Acharya, U., Daigh, A. L., & Oduor, P. G. (2021a). Machine learning for predicting field soil moisture using soil, crop, and nearby weather station data in the Red River Valley of the North. *Soil Systems*, 5(4), 57.

Acharya, U., Daigh, A. L. M., & Oduor, P. G. (2021b). Factors affecting the use of weather station data in predicting surface soil moisture for agricultural applications. *Canadian Journal of Soil Science*, 102(2), 419-431.

Achieng, K. O. (2019). Modelling of soil moisture retention curve using machine learning techniques: Artificial and deep neural networks vs support vector regression models. *Computers & Geosciences*, 133, 104320.

Ahmad, S., Kalra, A., & Stephen, H. (2010). Estimating soil moisture using remote sensing data: A machine learning approach. *Advances in Water Resources*, 33(1), 69-80.

Ali, I., Greifeneder, F., Stamenkovic, J., Neumann, M., & Notarnicola, C. (2015). Review of machine learning approaches for biomass and soil moisture retrievals from remote sensing data. *Remote Sensing*, 7(12), 16398-16421.

Amooh, M. K., & Bonsu, M. (2015). Effects of soil texture and organic matter on evaporative loss of soil moisture. *J. Glob. Agric. Ecol*, 3, 152-161.

Beckers, T. (2021). An Introduction to Gaussian Process Models. Online available at https://doi.org/10.48550/arXiv.2102.05497.

Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1), 16-28.

Clapcott, J., Goodwin, E., & Snelder, T. (2013). Predictive Models of Benthic Macroinvertebrate Metrics. *Prepared for Ministry for the Environment* (Vol. 2301).

Coenders-Gerrits, A. M. J., Hopp, L., Savenije, H. H., & Pfister, L. (2013). The effect of spatial throughfall patterns on soil moisture patterns at the hillslope scale. *Hydrology and Earth System Sciences*, 17(5), 1749-1763.

Coopersmith, E. J., Cosh, M. H., Bell, J. E., & Boyles, R. (2016). Using machine learning to produce near-surface soil moisture estimates from deeper in situ records at US Climate Reference Network (USCRN) locations: Analysis and applications to AMSR-E satellite validation. *Advances in Water Resources*, 98, 122-131.

Campbell Scientific, Inc. (2018). CS650 and CS655 Water Content Reflectometers Instruction Manual. Campbell Scientific Publishing, Logan, Utah. Online Available at https://s.campbellsci.com/documents/es/manuals/cs650_655.pdf.

Drucker, H., Burges, C. J., Kaufman, L., Smola, A., & Vapnik, V. (1996). Support vector regression machines. *Advances in Neural Information Processing Systems*, 9.

Dubois, A., Teytaud, F., & Verel, S. (2021). Short term soil moisture forecasts for potato crop farming: A machine learning approach. *Computers and Electronics in Agriculture*, 180, 105902.

Duvenaud, D. (2014). Automatic model construction with Gaussian processes. Doctoral dissertation, University of Cambridge. Cambridge, UK.

Easton, Z. M., Bock, E., & Collick, A. S. (2017). Factors when considering an agricultural drainage system. *Virginia Cooperative Extension*, BSE-208.

Gao, Z. Q., Liu, C. S., Gao, W., & Chang, N. B. (2011). A coupled remote sensing and the Surface Energy Balance with Topography Algorithm (SEBTA) to estimate actual evapotranspiration over heterogeneous terrain. *Hydrology and Earth System Sciences*, 15(1), 119-139.

Giardina, C. P., & Ryan, M. G. (2000). Evidence that decomposition rates of organic carbon in mineral soil do not vary with temperature. *Nature*, 404(6780), 858-861.

Gill, M. K., Asefa, T., Kemblowski, M. W., & McKee, M. (2006). Soil moisture prediction using support vector machines 1. *JAWRA Journal of the American Water Resources Association*, 42(4), 1033-1046.

Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep Learning* (vol. 1) Cambridge.

Grayson, R. B., Western, A. W., Chiew, F. H., & Blöschl, G. (1997). Preferred states in spatial soil moisture patterns: Local and nonlocal controls. *Water Resources Research*, 33(12), 2897-2908.

Grimes, D. I. F., Coppola, E., Verdecchia, M., & Visconti, G. (2003). A neural network approach to real-time rainfall estimation for Africa using satellite data. *Journal of Hydrometeorology*, 4(6), 1119-1133.

Guevara, M., & Vargas, R. (2019). Downscaling satellite soil moisture using geomorphometry and machine learning. *PloS one*, 14(9), e0219639.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3(Mar), 1157-1182.

Gwak, Y., & Kim, S. (2017). Factors affecting soil moisture spatial variability for a humid forest hillslope. *Hydrological Processes*, 31(2), 431-445.

Hassan-Esfahani, L., Torres-Rua, A., Jensen, A., & McKee, M. (2015). Assessment of surface soil moisture using high-resolution multi-spectral imagery and artificial neural networks. *Remote Sensing*, 7(3), 2627-2646.

Joshi, C., & Mohanty, B. P. (2010). Physical controls of near‐surface soil moisture across varying spatial scales in an agricultural landscape during SMEX02. *Water Resources Research*, 46(12), W12503.

Khedri, E., Hasanlou, M., & Tabatabaeenejad, A. (2017). Estimating soil moisture using polsar data: a machine learning learning approach. International Archives of the Photogrammetry, *Remote Sensing & Spatial Information Sciences*, 42, 133-137.

Lakshmi, V., Jackson, T. J., & Zehrfuhs, D. (2003). Soil moisture–temperature relationships: results from two field experiments. *Hydrological Processes*, 17(15), 3041-3057.

Lee, R. (1978). Forest microclimatology. Columbia Univ. *Press,* New York., 33-84.

Liu, Y., Xia, X., Yao, L., Jing, W., Zhou, C., Huang, W., ... & Yang, J. (2020). Downscaling satellite retrieved soil moisture using regression tree‐based machine learning algorithms over Southwest France. *Earth and Space Science*, 7(10), e2020EA001267.

Majcher, J., Kafarski, M., Wilczek, A., Szypłowska, A., Lewandowski, A., Woszczyk, A., & Skierucha, W. (2021). Application of a dagger probe for soil dielectric permittivity measurement by TDR. *Measurement*, 178, 109368.

Moore, I. D., Burch, G. J., & Mackenzie, D. H. (1988). Topographic effects on the distribution of surface soil water and the location of ephemeral gullies. *Transactions of the ASAE*, 31(4), 1098-1107.

Muñoz-Carpena, R. (2004). Field devices for monitoring soil water content. *EDIS*, 2004(8).

Polykovskiy, D., & Novikov, A. (2017). Bayesian methods for machine learning. *Coursera and National Research University Higher School of Economics*, 20(1), 8.

Prakash, S., Sharma, A., & Sahu, S. S. (2018). Soil moisture prediction using machine learning. In 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT) (pp. 1-6). *IEEE*, Coimbatore, India, 2018 April.

Rasheed, M. W., Tang, J., Sarwar, A., Shah, S., Saddique, N., Khan, M. U., ... & Sultan, M. (2022). Soil Moisture Measuring Techniques and Factors Affecting the Moisture Dynamics: A Comprehensive Review. *Sustainability*, 14(18), 11538.

Rasmussen, C., & Williams, C. (2006). Gaussian processes for machine learning. MIT Press: Cambridge, MA.

Robinson, D. A., Campbell, C. S., Hopmans, J. W., Hornbuckle, B. K., Jones, S. B., Knight, R., ... & Wendroth, O. (2008). Soil moisture measurement for ecological and hydrological watershed-scale observatories: A review. *Vadose Zone Journal*, 7(1), 358-389.

Smola A.J., Schölkopf B., 2004. A tutorial on support vector regression. *Statistics and Computing,* 14(3):199-222.

Srivastava, A., Yetemen, O., Kumari, N., & Saco, P. M. (2018). Role of Solar Radiation and Topography on Soil Moisture Variations in Semiarid Aspect-Controlled Ecosystems. *Sat*, 1(1).

Sun, H., & Cui, Y. (2021). Evaluating downscaling factors of microwave satellite soil moisture based on machine learning method. *Remote Sensing*, 13(1), 133.

Topp, G. C., Davis, J. L., & Annan, A. P. (1980). Electromagnetic determination of soil water content: Measurements in coaxial transmission lines. *Water Resources Research*, 16(3), 574-582.

Vereecken, H., Huisman, J. A., Pachepsky, Y., Montzka, C., Van Der Kruk, J., Bogena, H., ... & Vanderborght, J. (2014). On the spatio-temporal dynamics of soil moisture at the field scale. *Journal of Hydrology*, 516, 76-96.

Wanas, S. A. (2002). The role of organic compost in alleviating soil compaction. *Egypt. J. Appl. Sci*, 17(6), 363-372.

Zaman, B., McKee, M., & Neale, C. M. (2012). Fusion of remotely sensed data for soil moisture estimation using relevance vector and support vector machines. *International Journal of Remote Sensing*, 33(20), 6516-6552.

Zaslavsky, D., & Sinai, G. (1981). Surface hydrology: I—explanation of phenomena. *Journal of the Hydraulics Division*, 107(1), 1-16.

Zhao, Y., Peth, S., Reszkowska, A., Gan, L., Krümmelbein, J., Peng, X., & Horn, R. (2011). Response of soil moisture and temperature to grazing intensity in a Leymus chinensis steppe, Inner Mongolia. *Plant and Soil*, 340, 89-102.

Zhang, L., Zhang, Z., Xue, Z., & Li, H. (2021). Sensitive feature evaluation for soil moisture retrieval based on multi-source remote sensing data with few in-situ measurements: A case study of the continental US. *Water*, 13(15), 2003.

APPENDIX A. CORRELATIONAL MATRICES

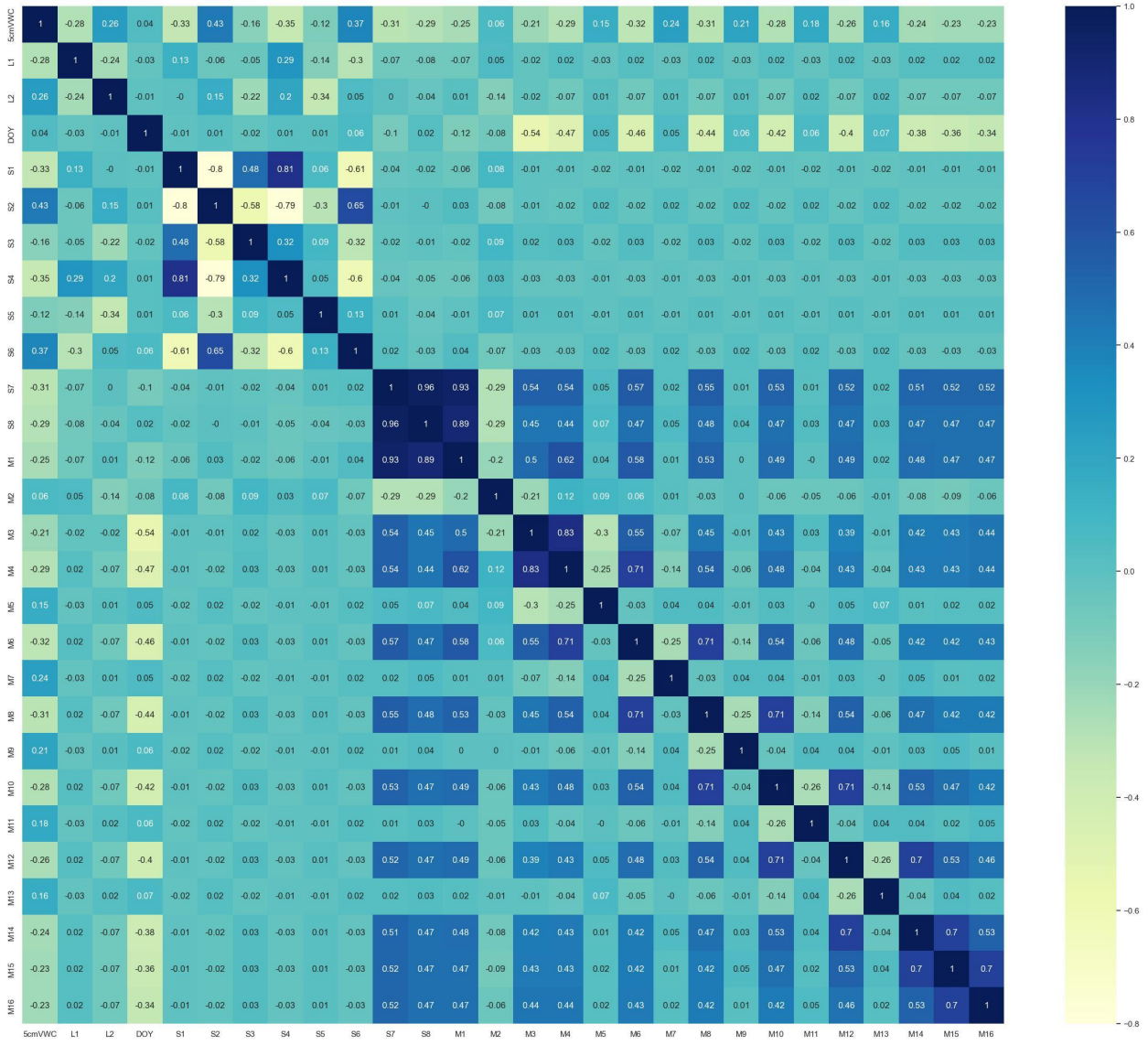

Figure A1. Correlation matrix of the 27 features and soil moisture when soil depth was 5 cm.

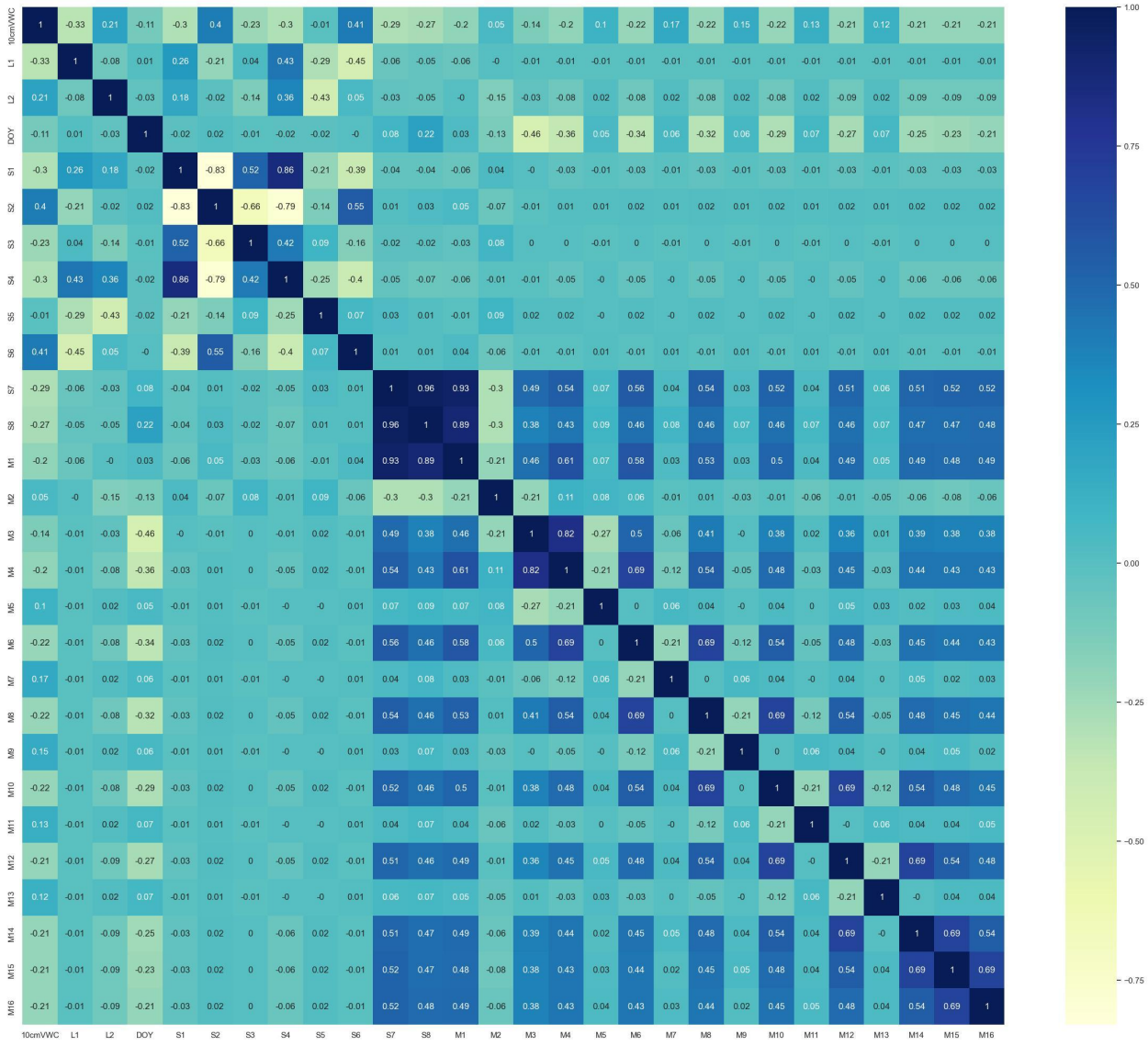

Figure A2. Correlation matrix of the 27 features and soil moisture when soil depth was 10 cm.

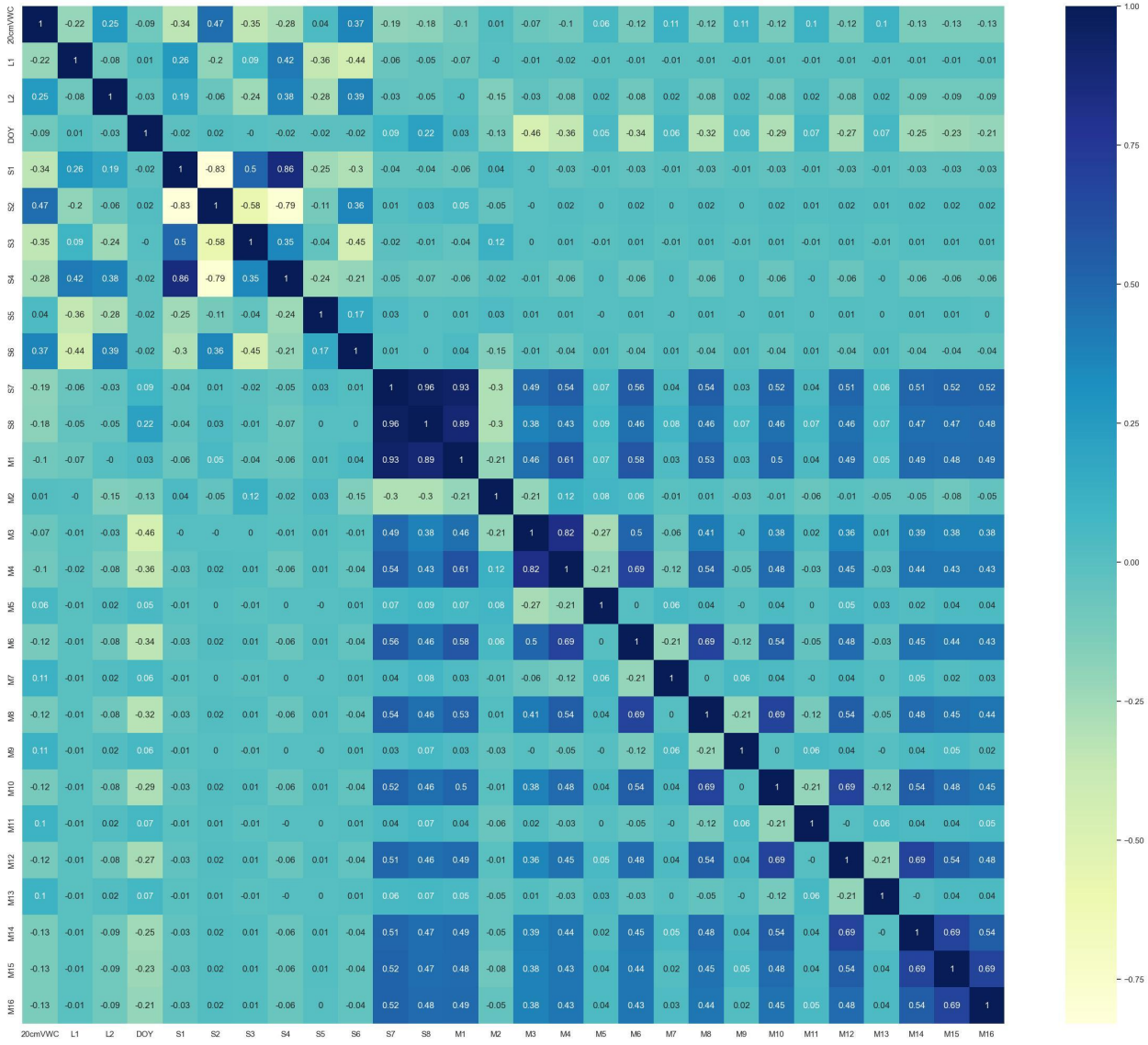

Figure A3. Correlation matrix of the 27 features and soil moisture when soil depth was 20 cm.

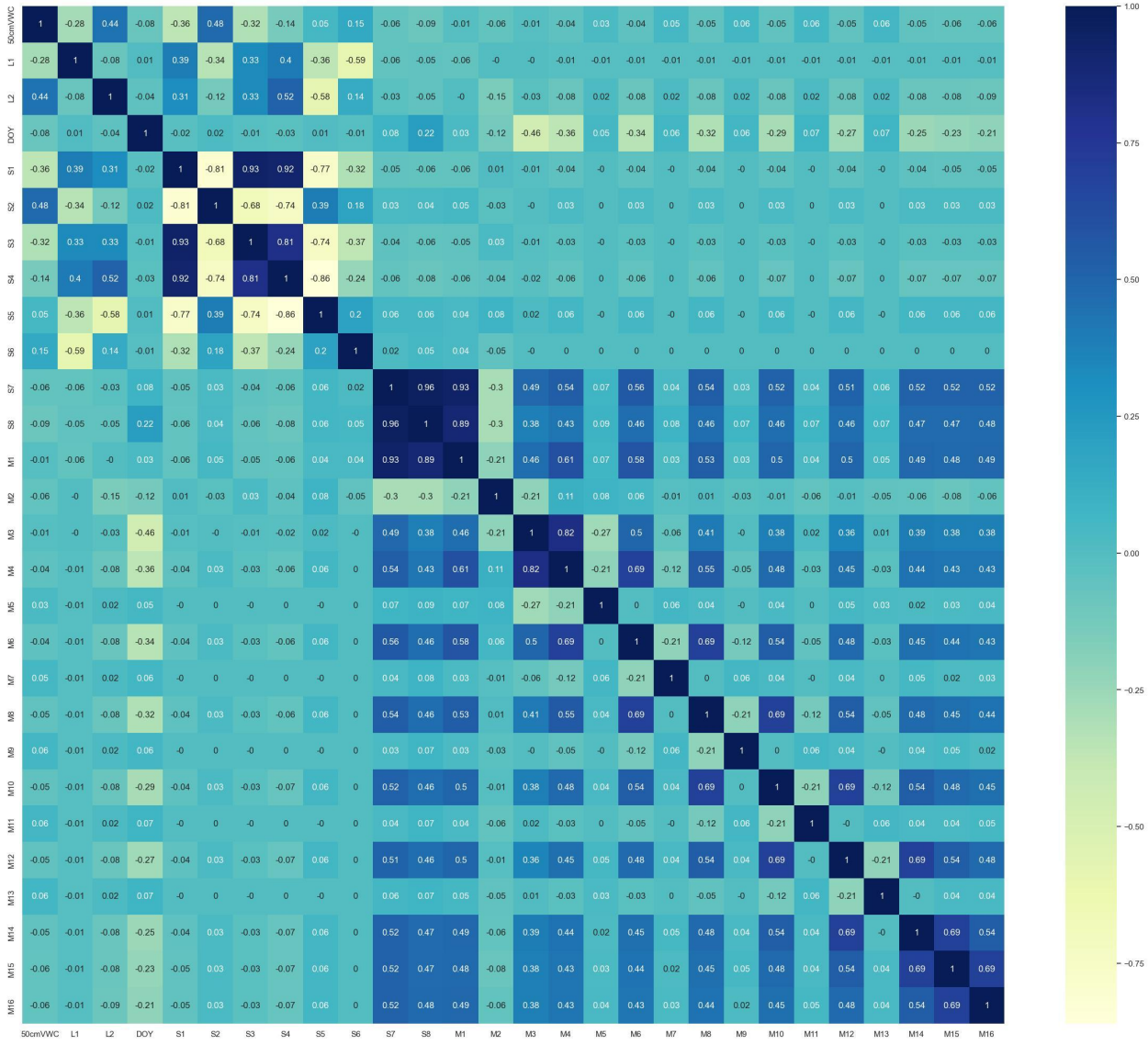

Figure A4. Correlation matrix of the 27 features and soil moisture when soil depth was 50 cm.

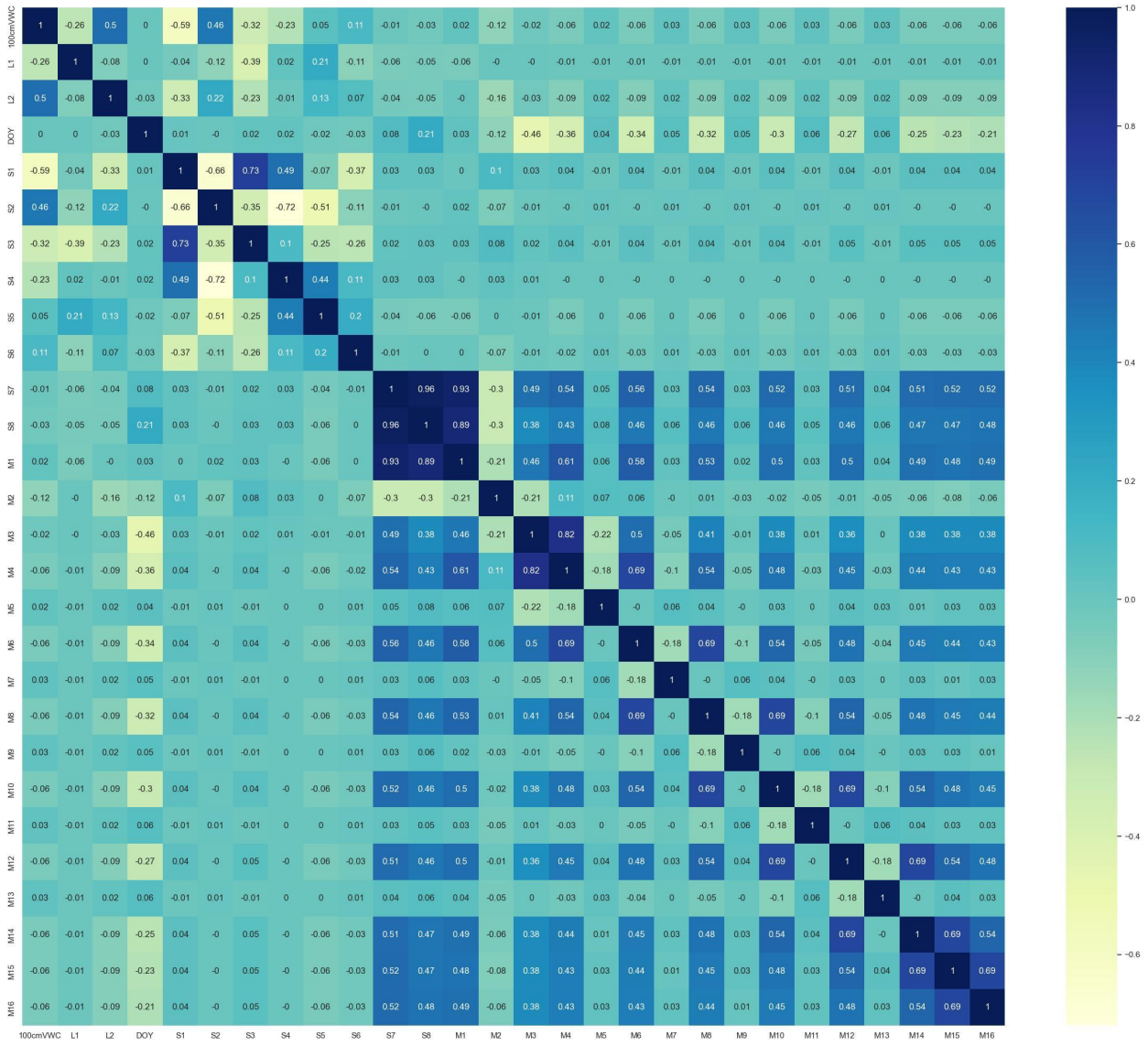

Figure A5. Correlation matrix of the 27 features and soil moisture when soil depth was 100 cm.

APPENDIX B. GRAPHICAL COMPARISONS

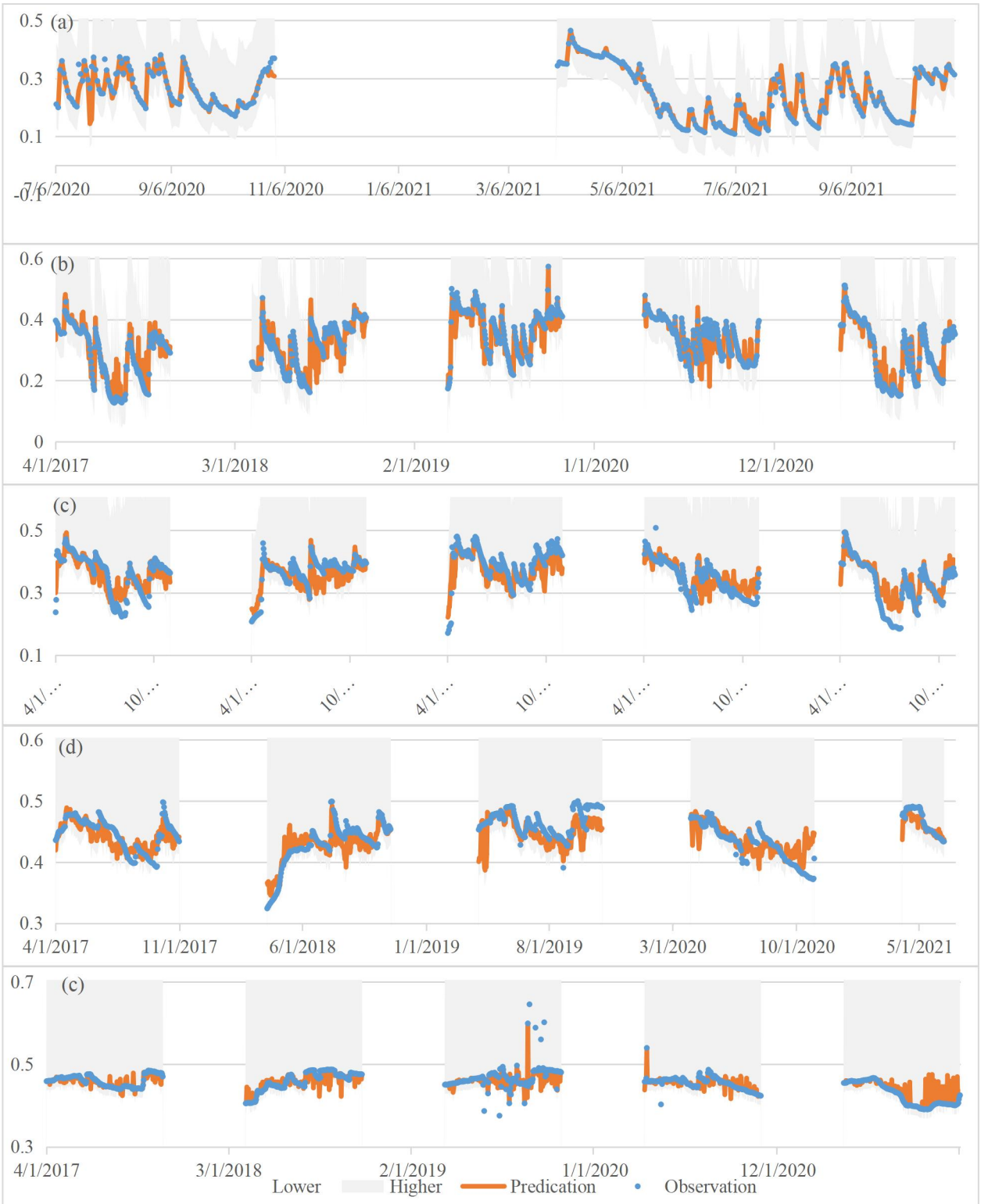

Figure B1. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Campbell. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

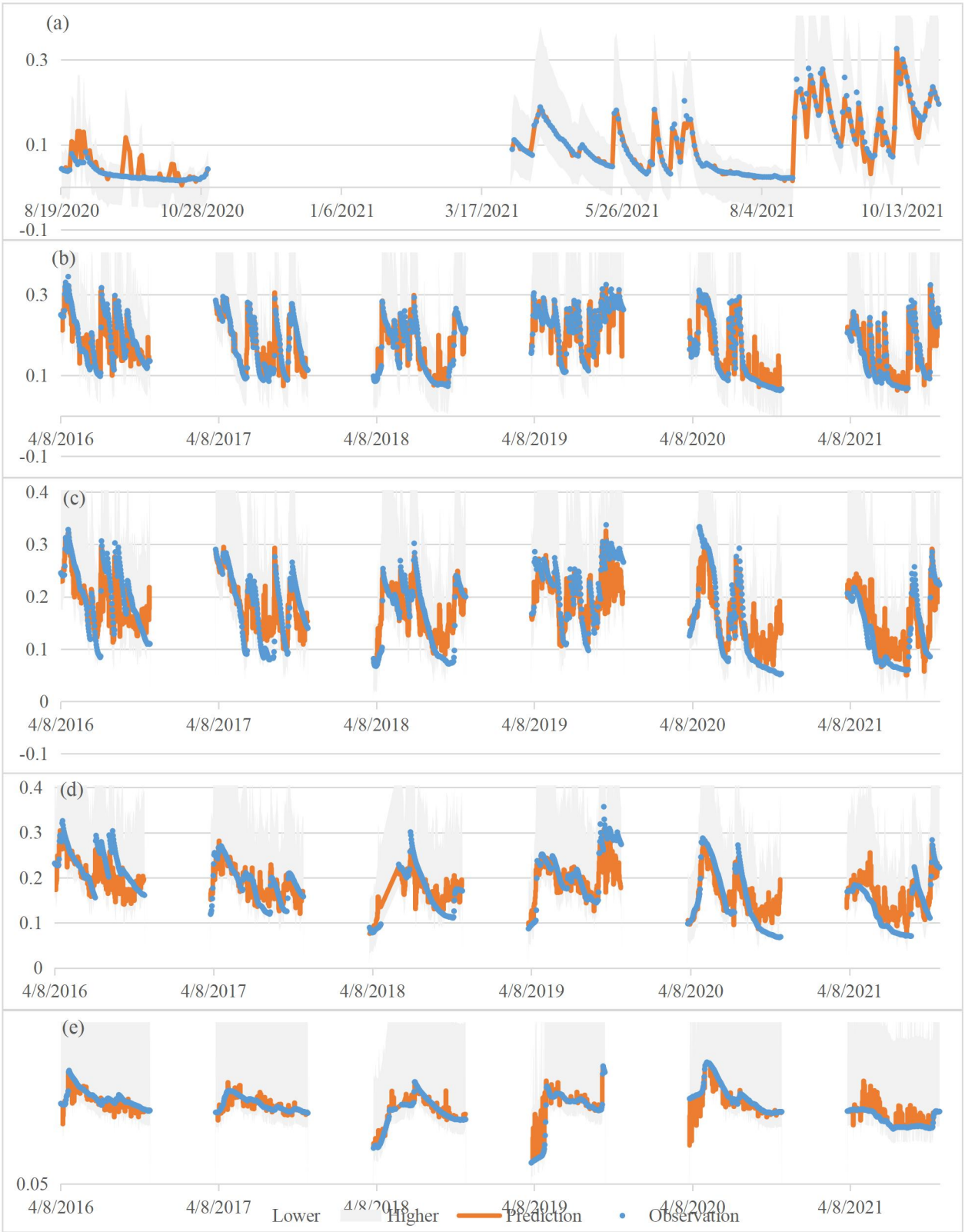

Figure B2. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Carrington. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

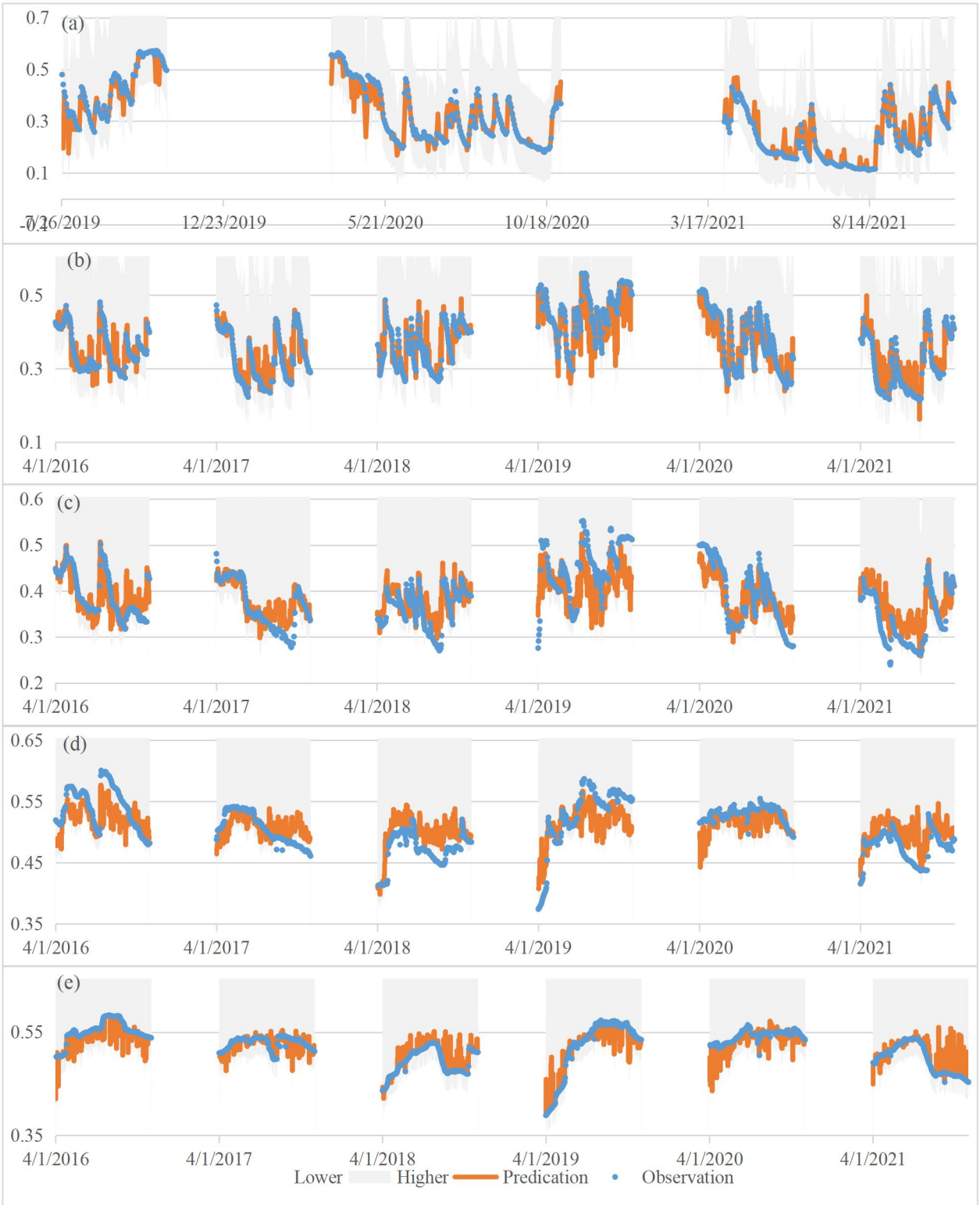

Figure B3. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Fargo. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

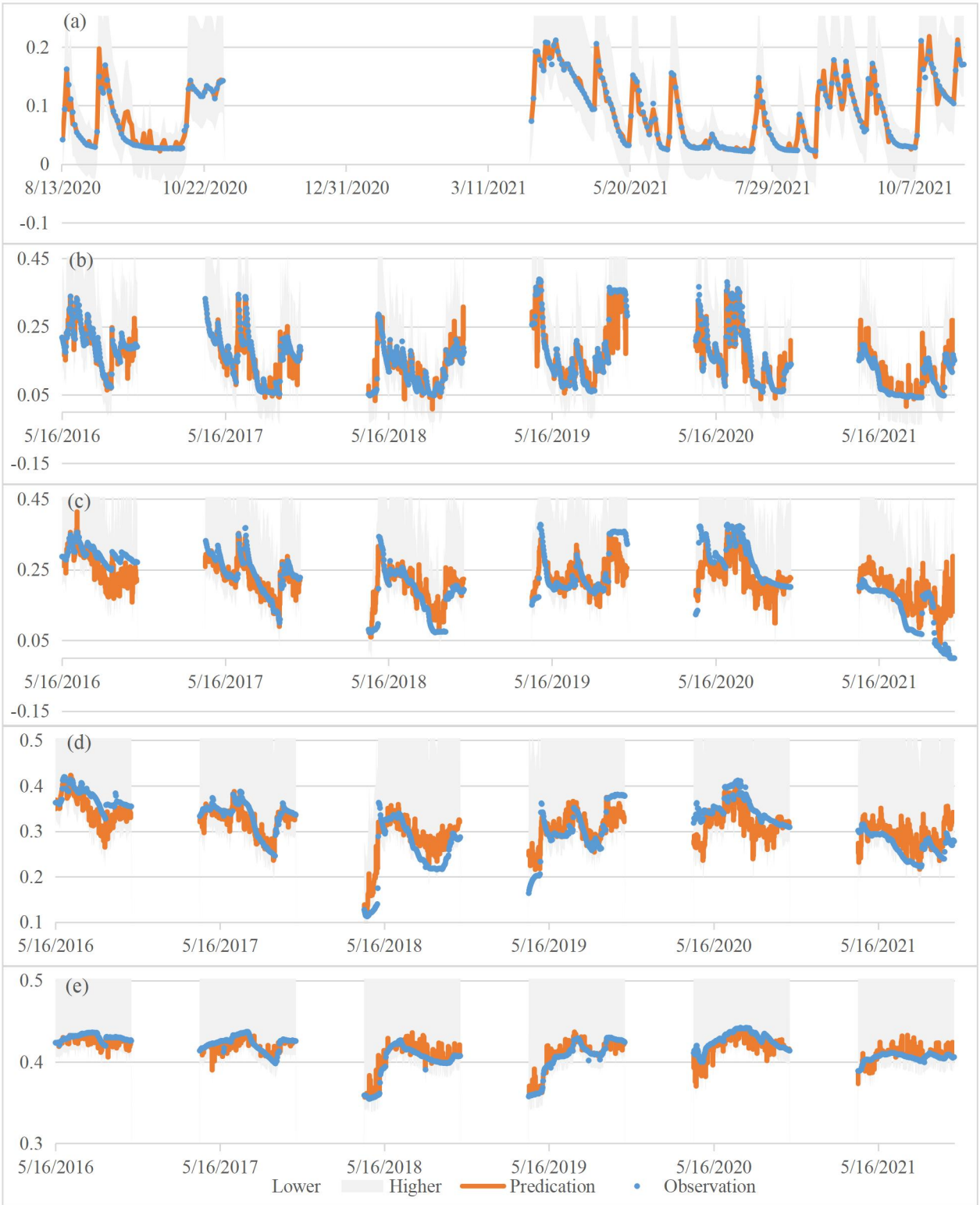

Figure B4. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Fox. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

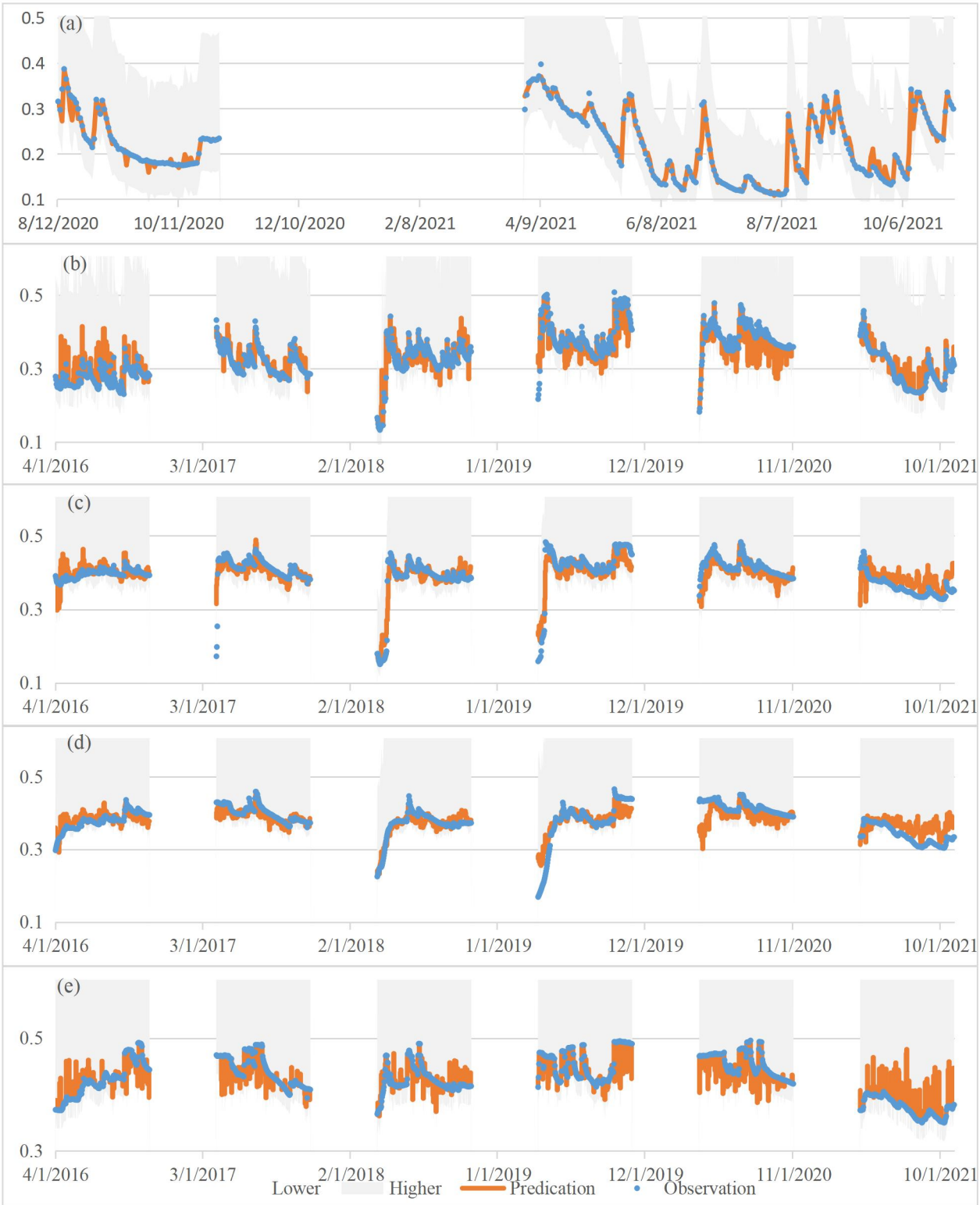

Figure B5. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Grand Forks. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

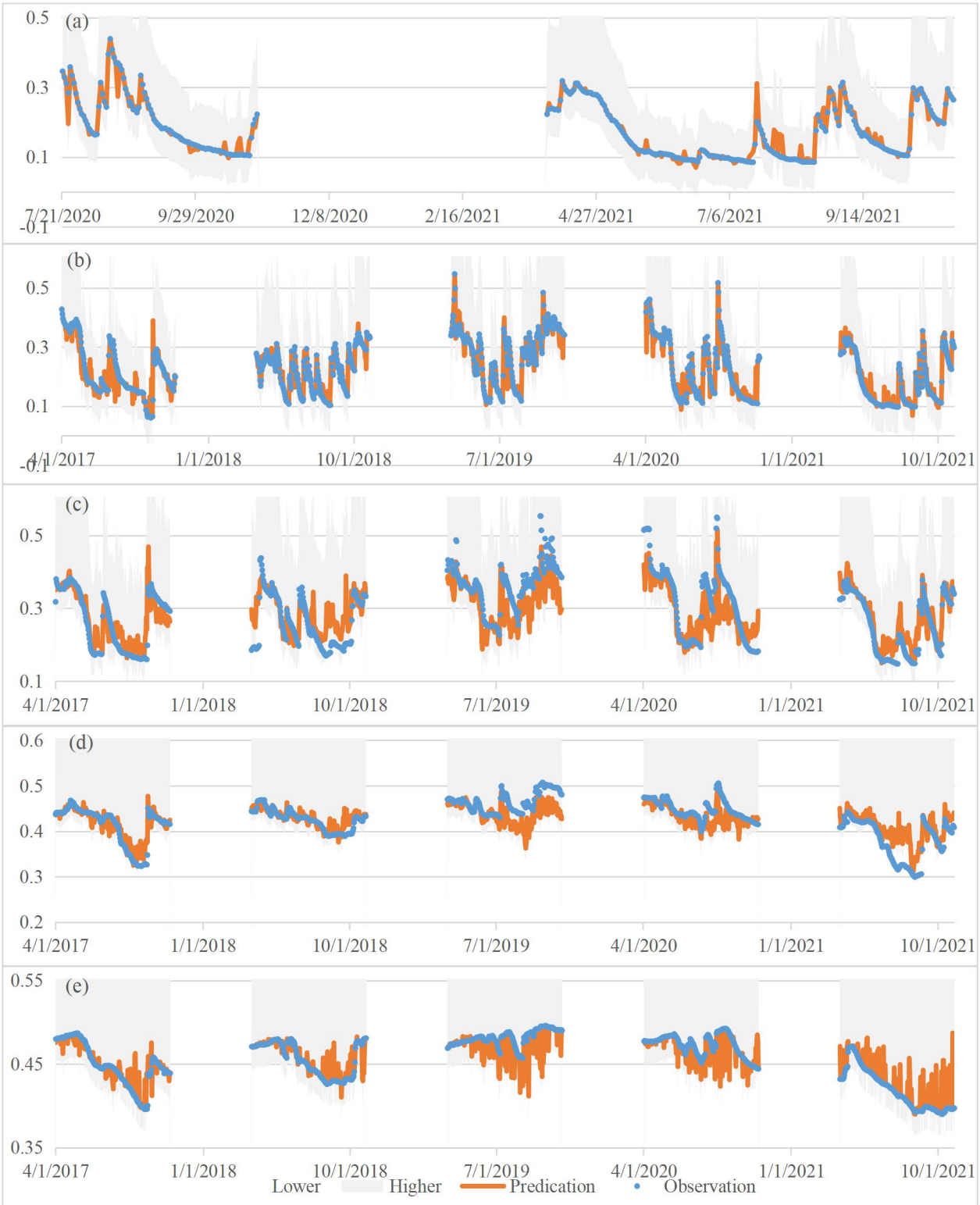

Figure B6. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Hillsboro. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

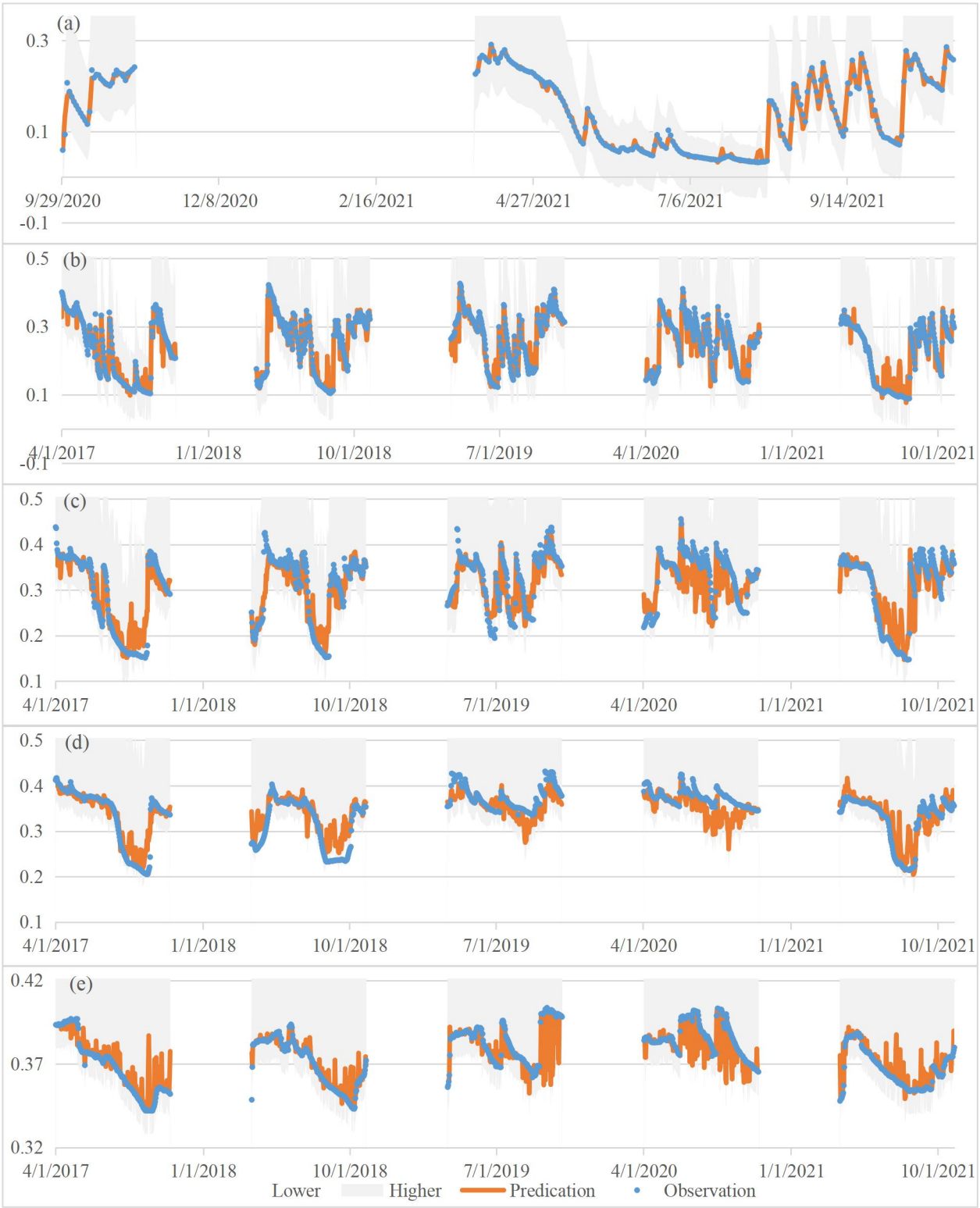

Figure B7. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Mavie. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

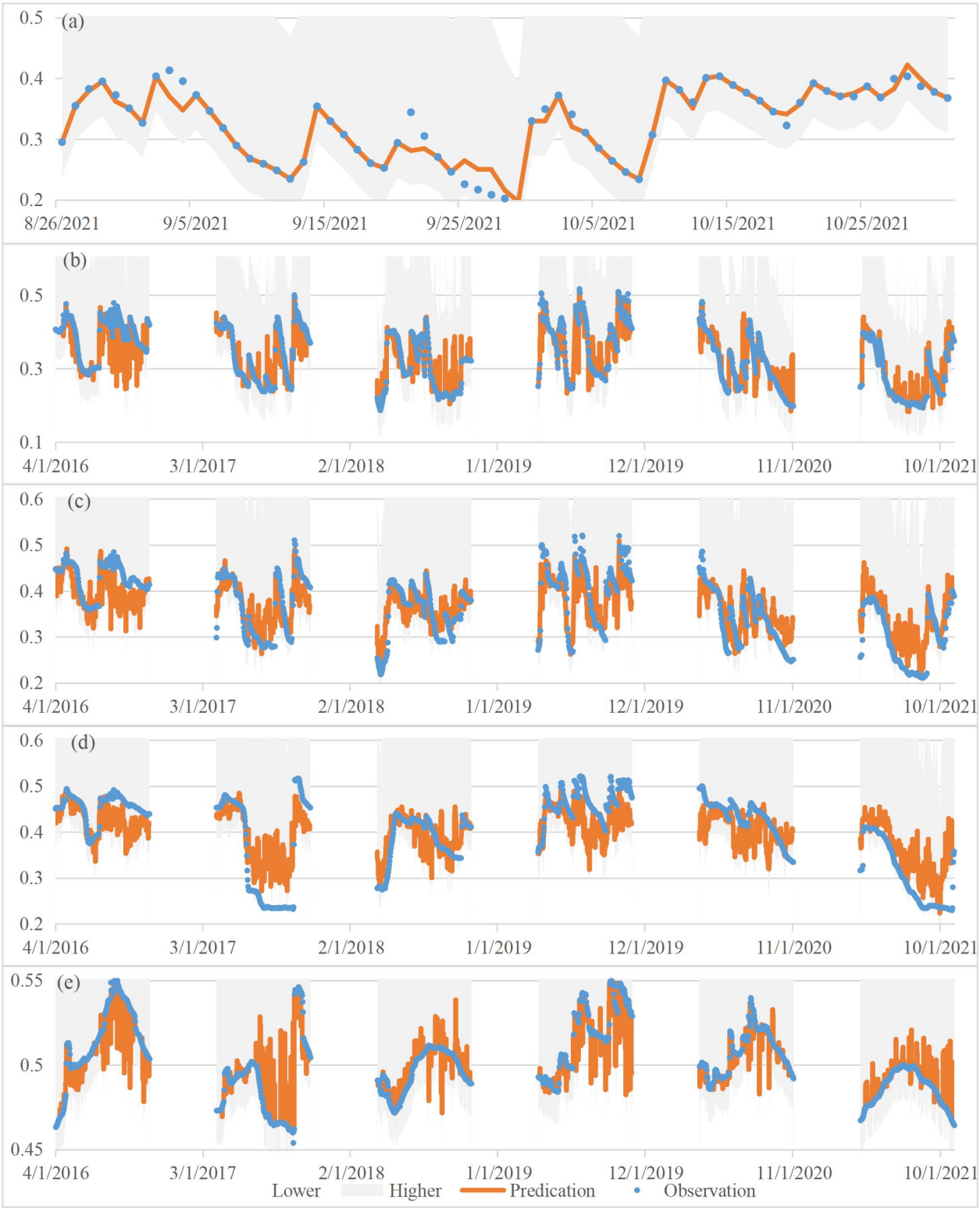

Figure B8. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Mooreton. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

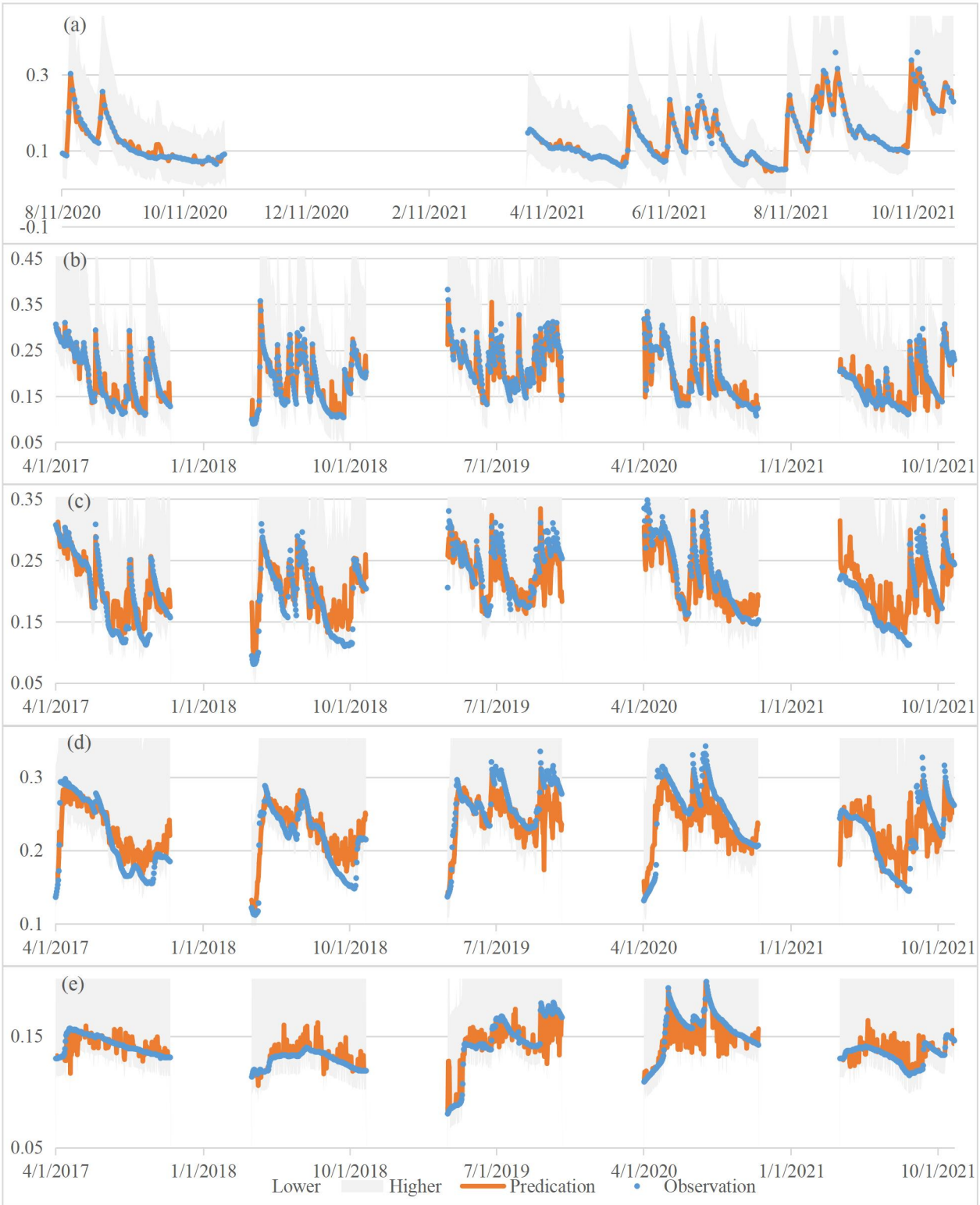

Figure B9. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Pekin. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

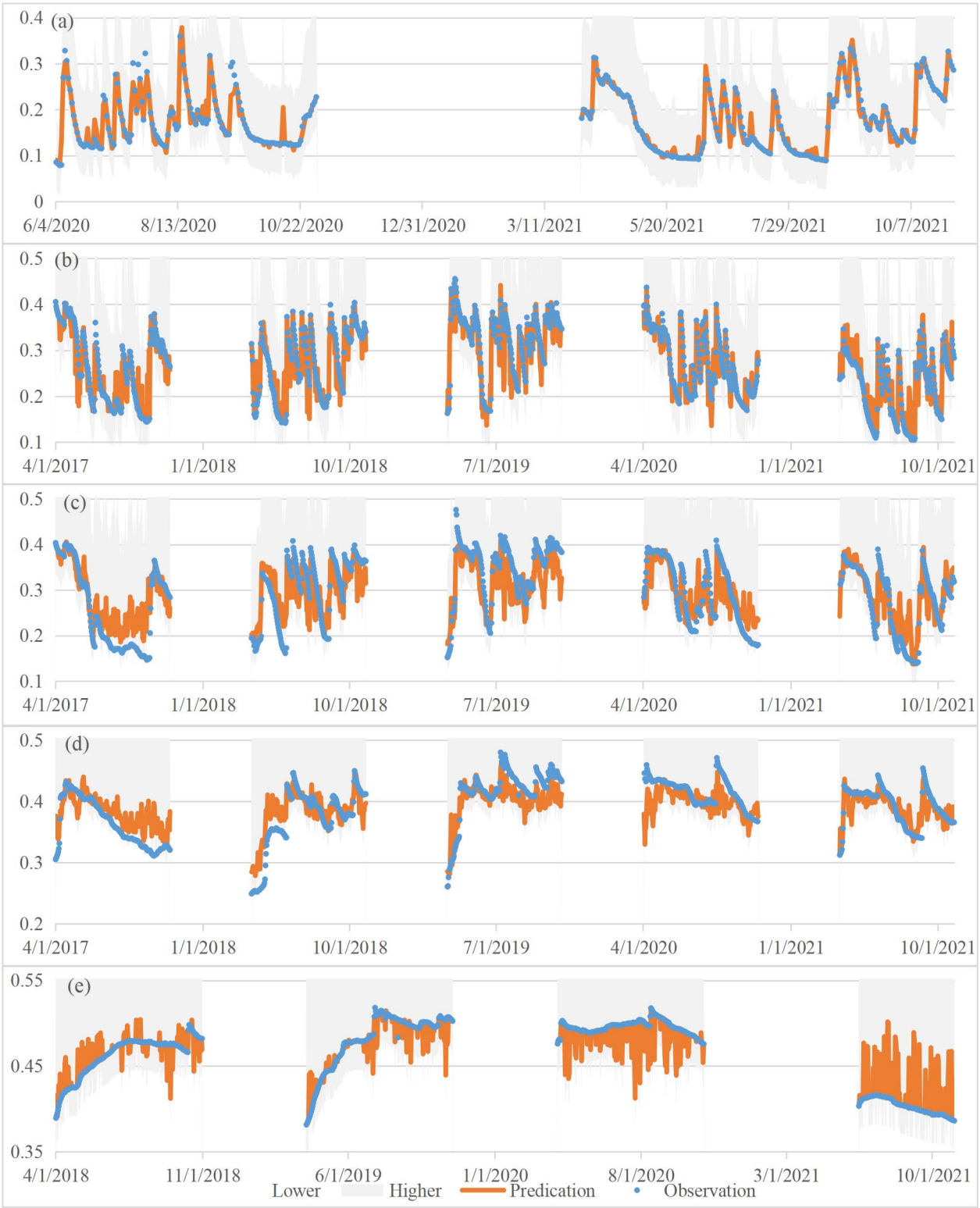

Figure B10. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Sabin. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

Figure B11. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Waukon. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

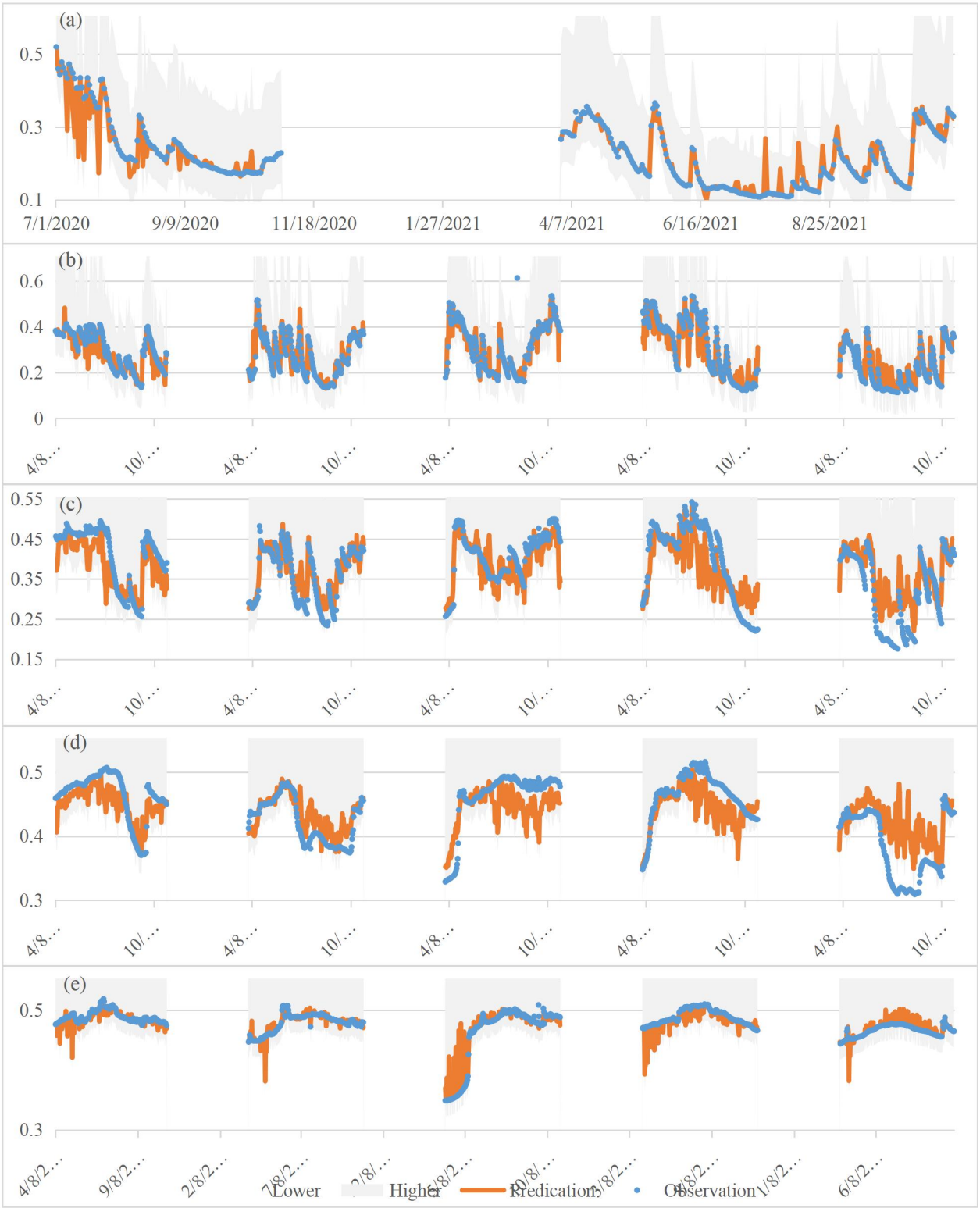

Figure B12. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Grafton. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

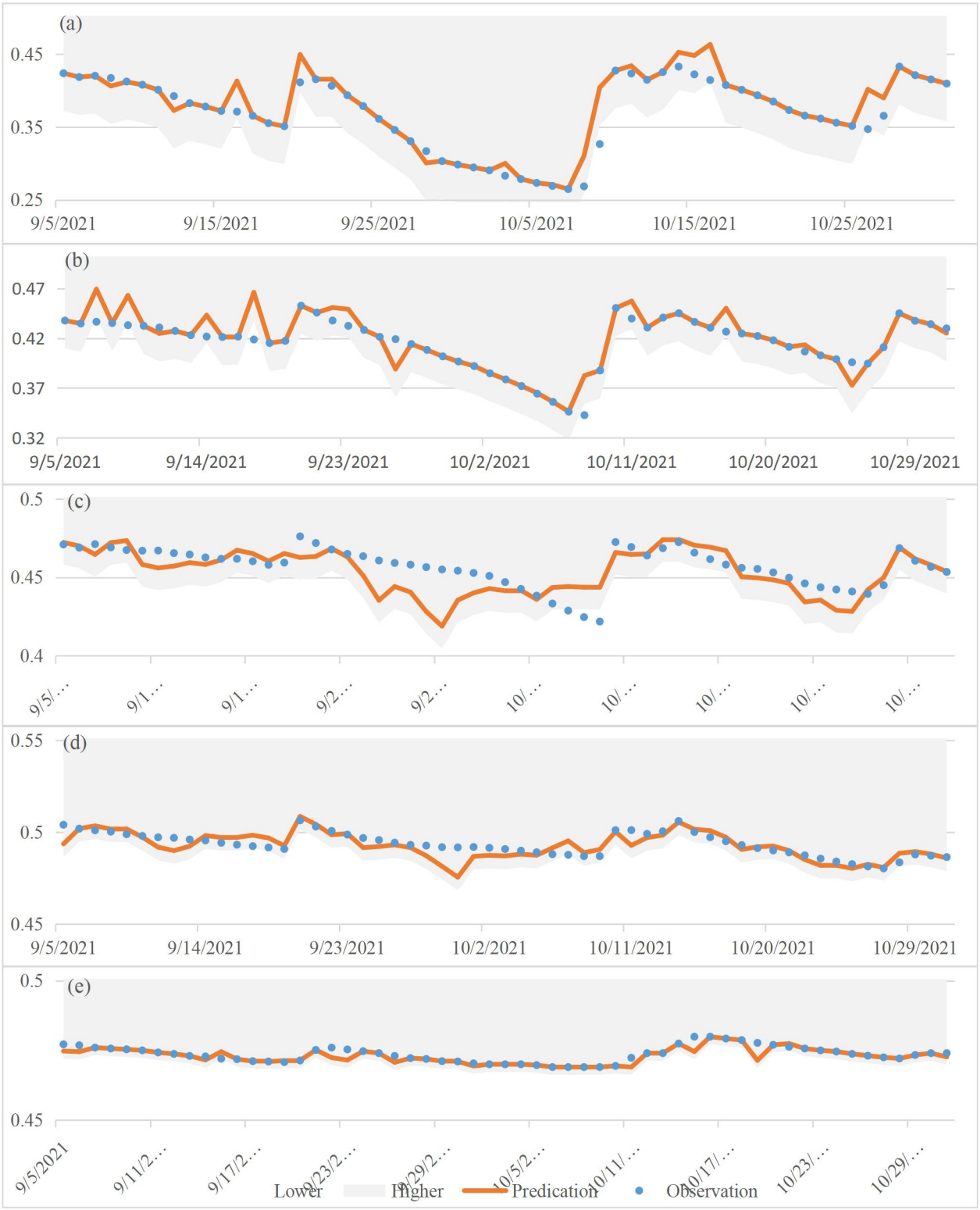

Figure B13. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Ada. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

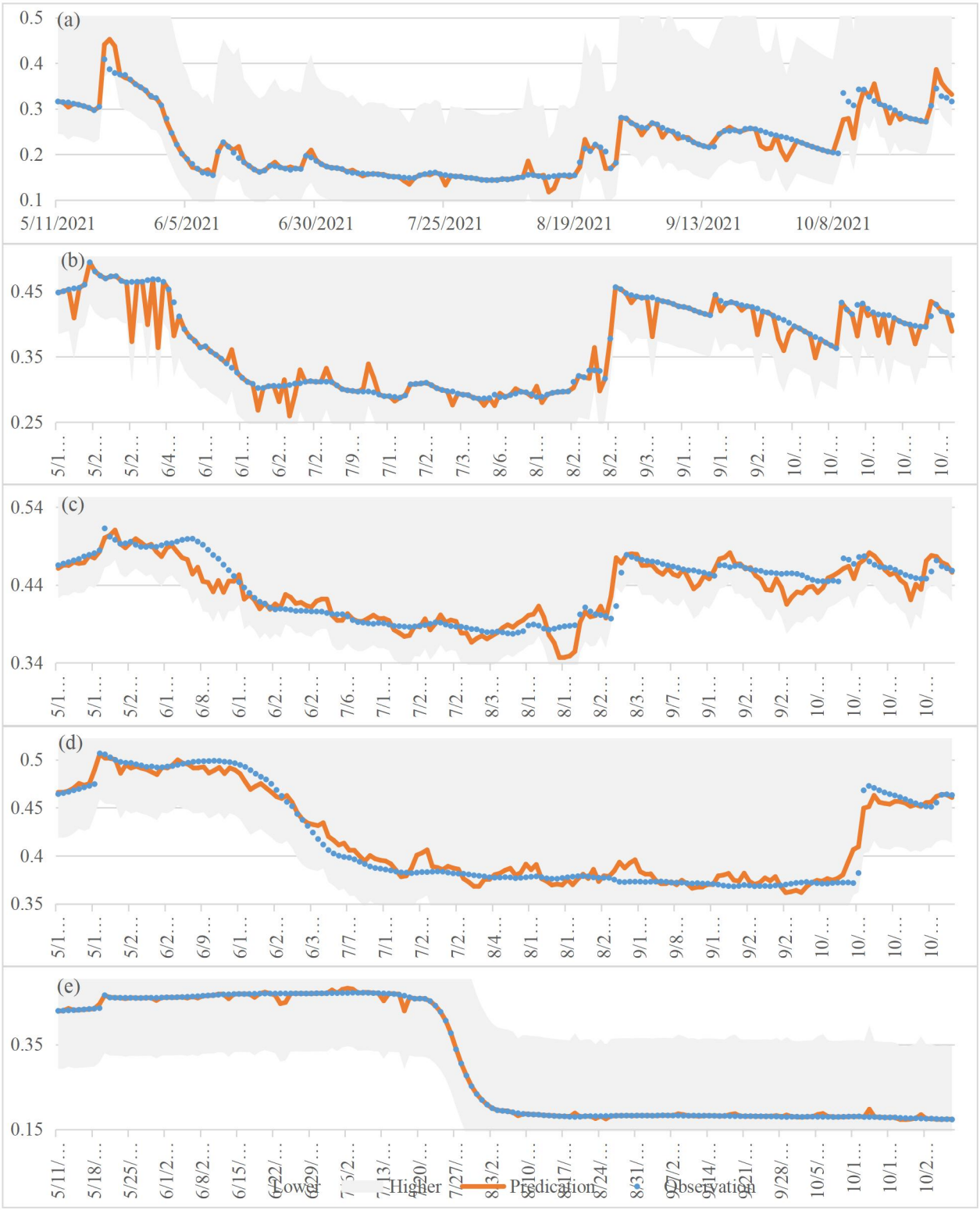

Figure B14. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Alvarado. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

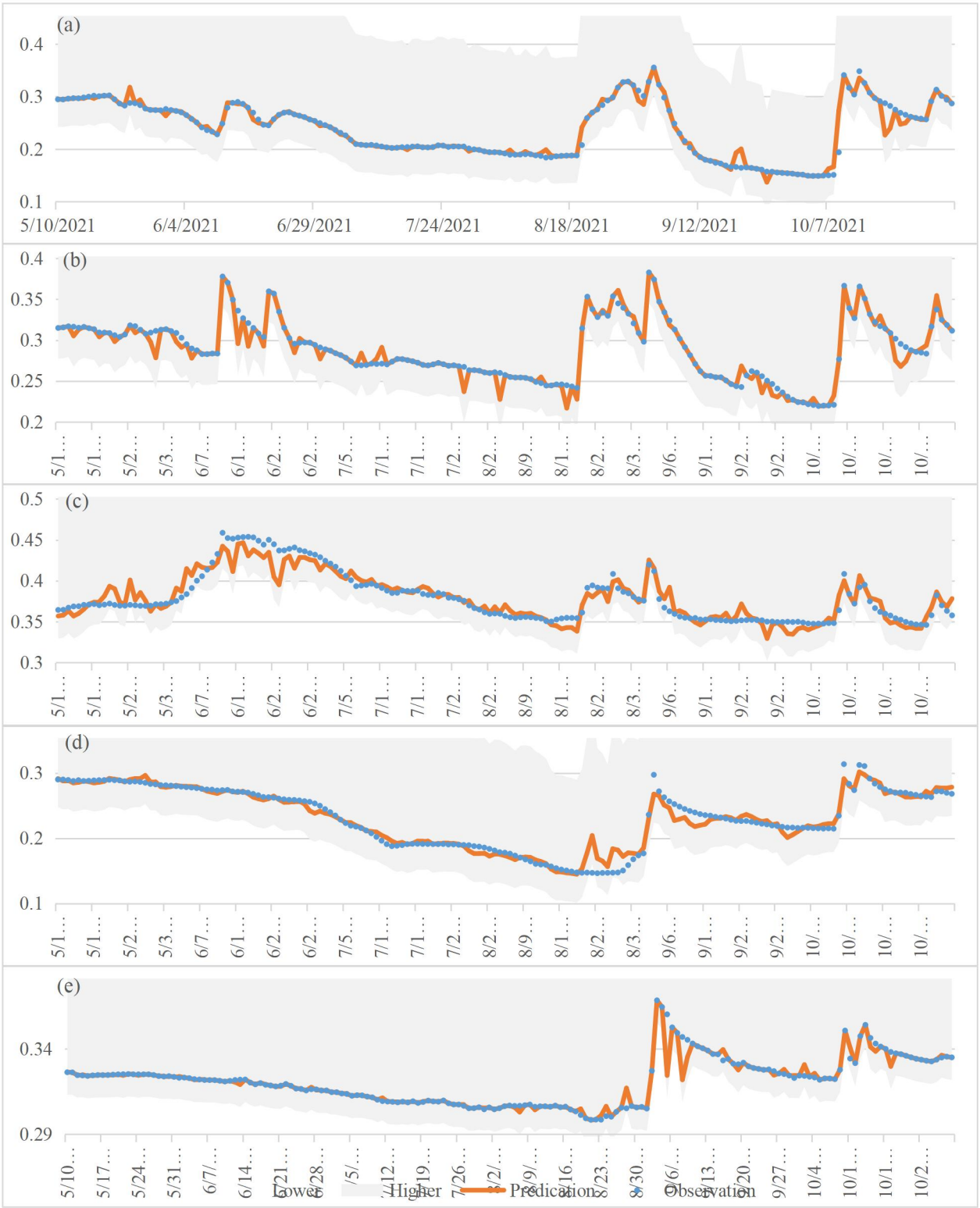

Figure B15. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Ayr. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

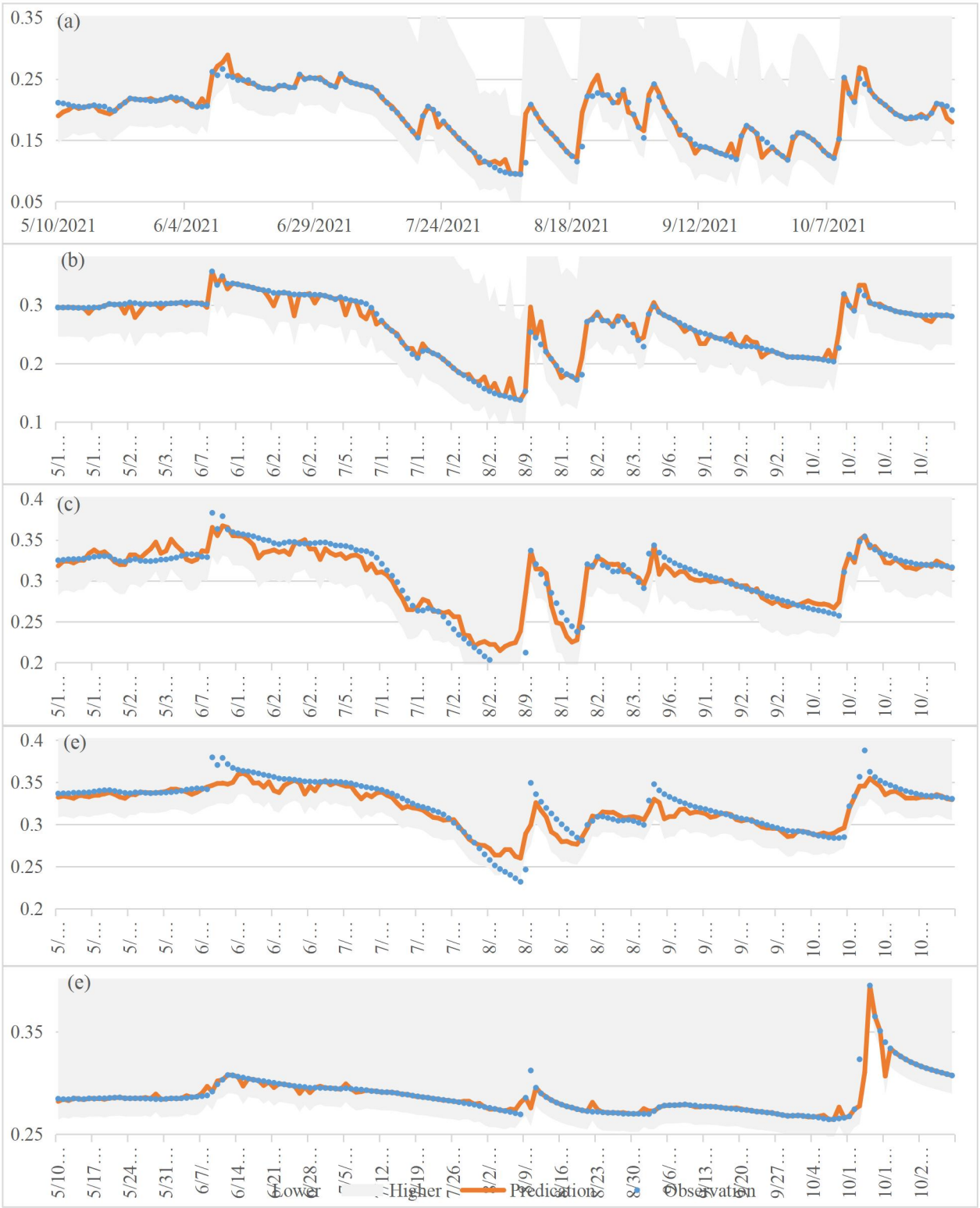

Figure B16. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Clyde. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

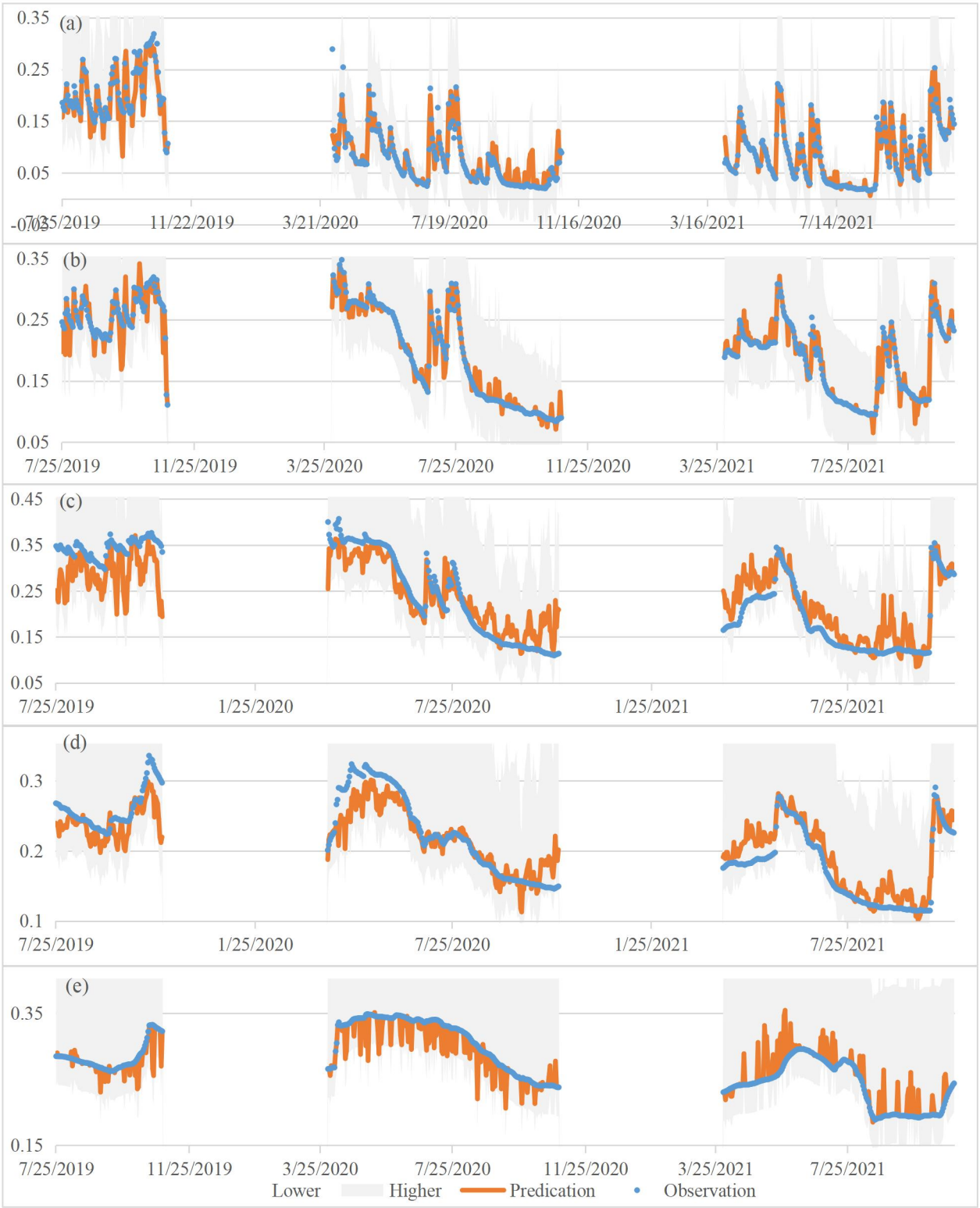

Figure B17. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Courtenay. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

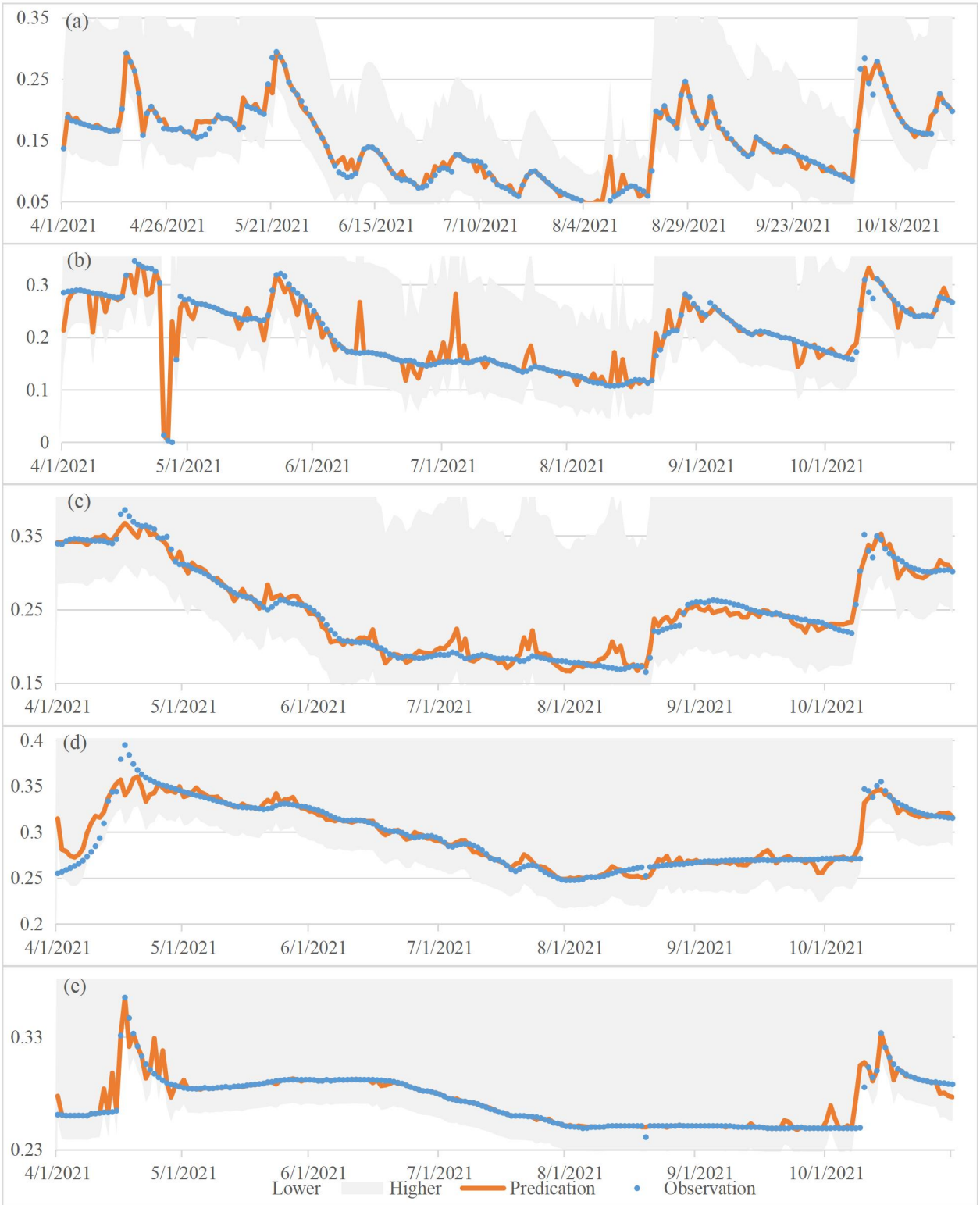

Figure B18. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Crystal. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

Figure B19. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Denhoff. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

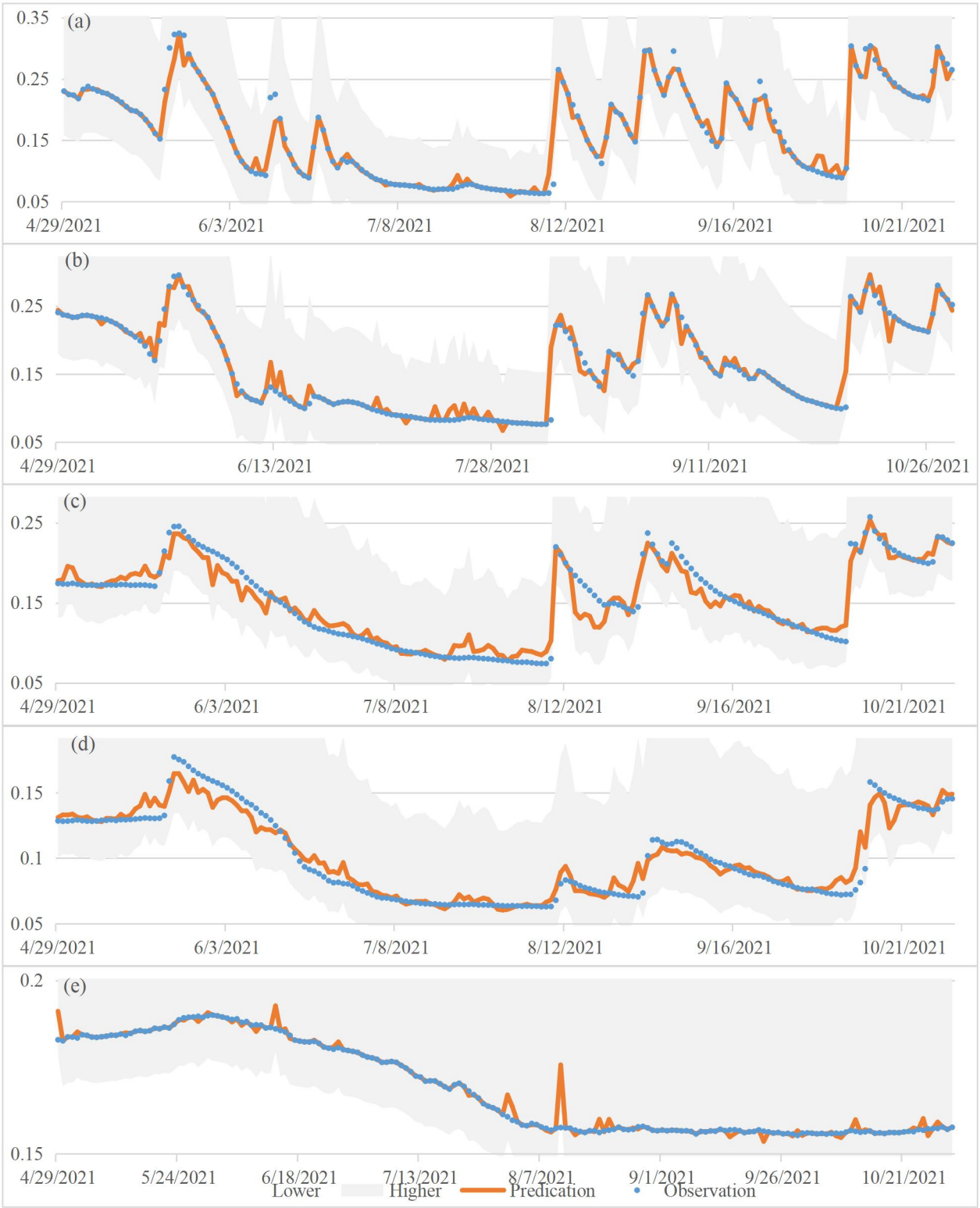

Figure B20. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Emerado. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

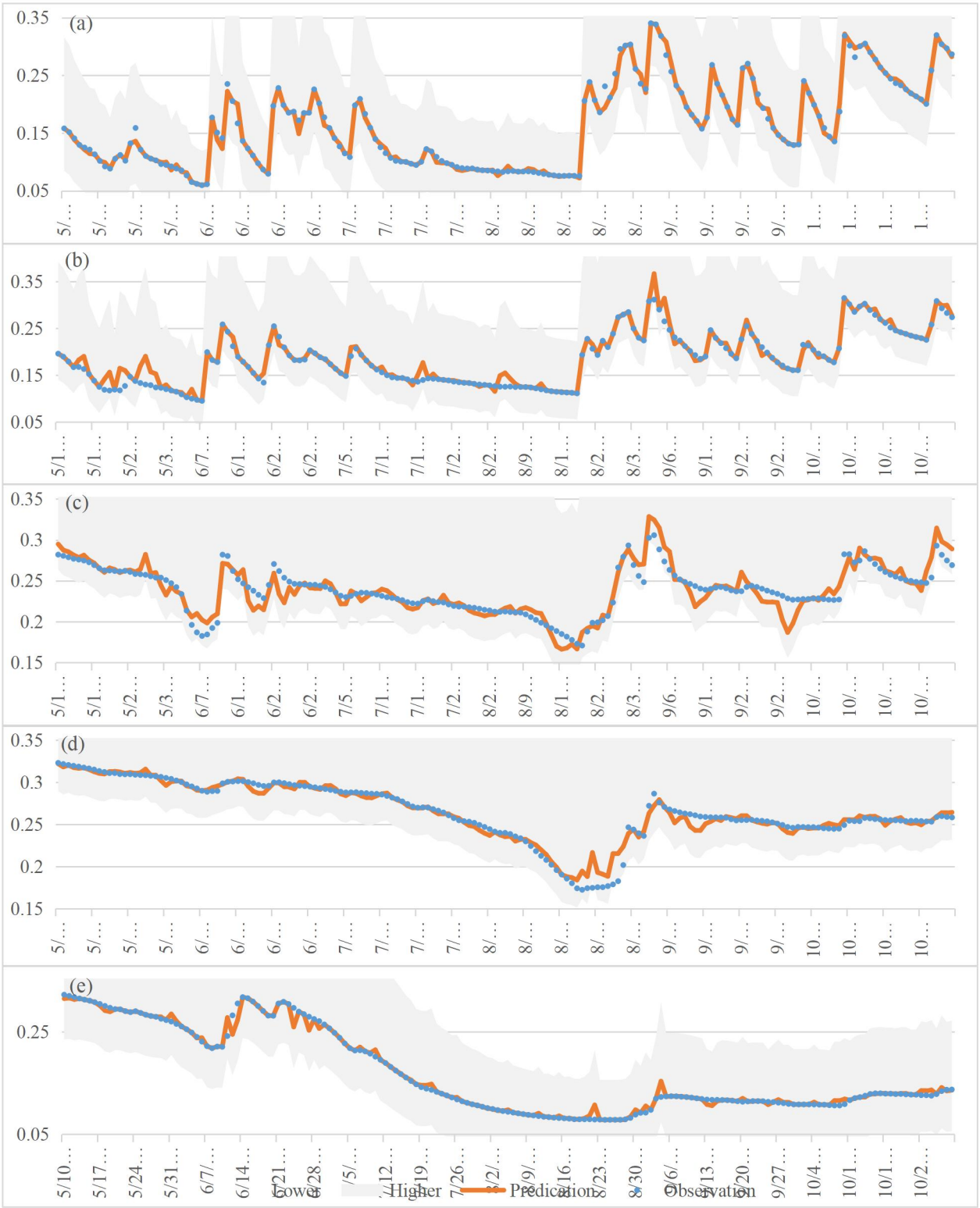

Figure B21. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Glyndon. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

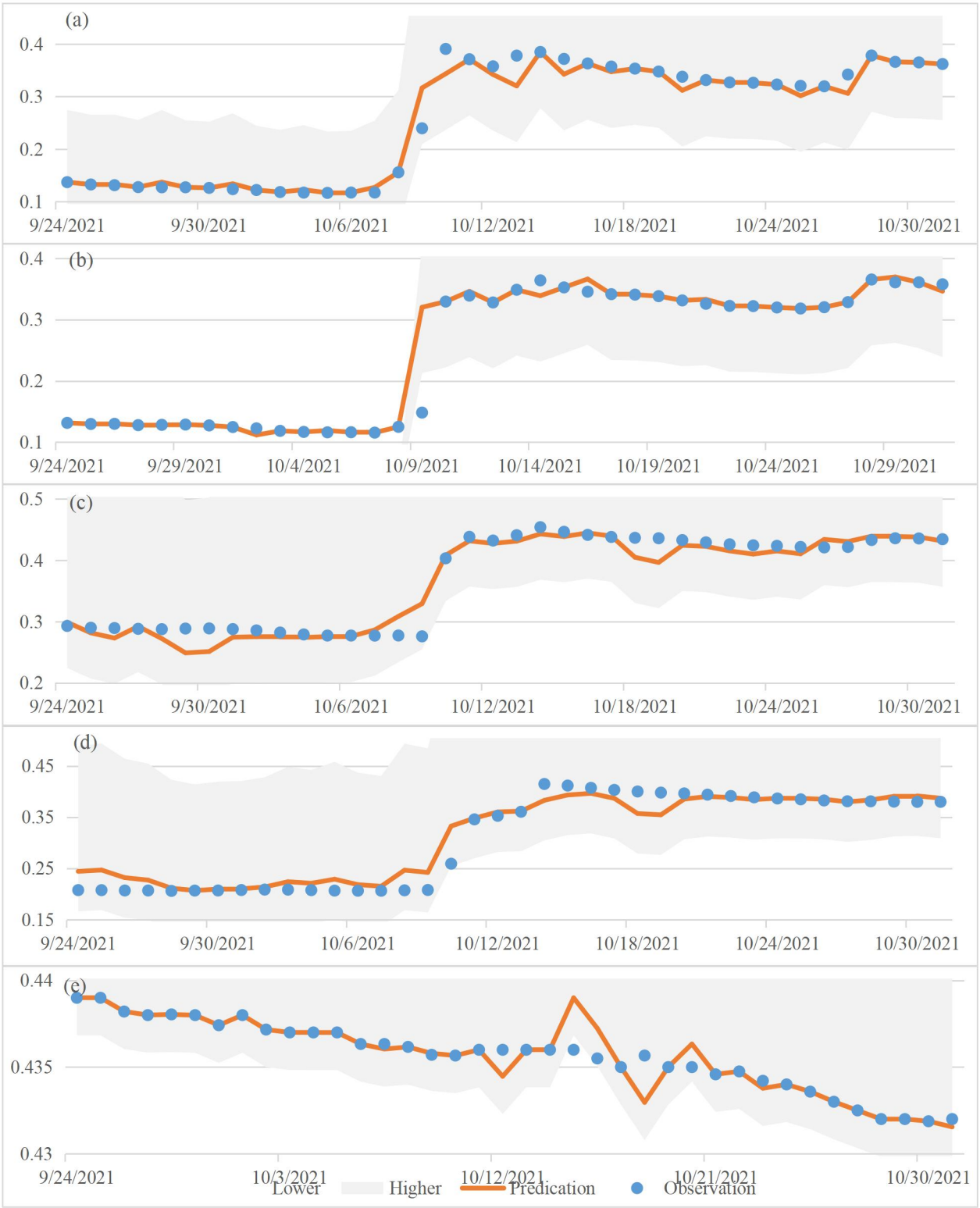

Figure B22. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Humboldt. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

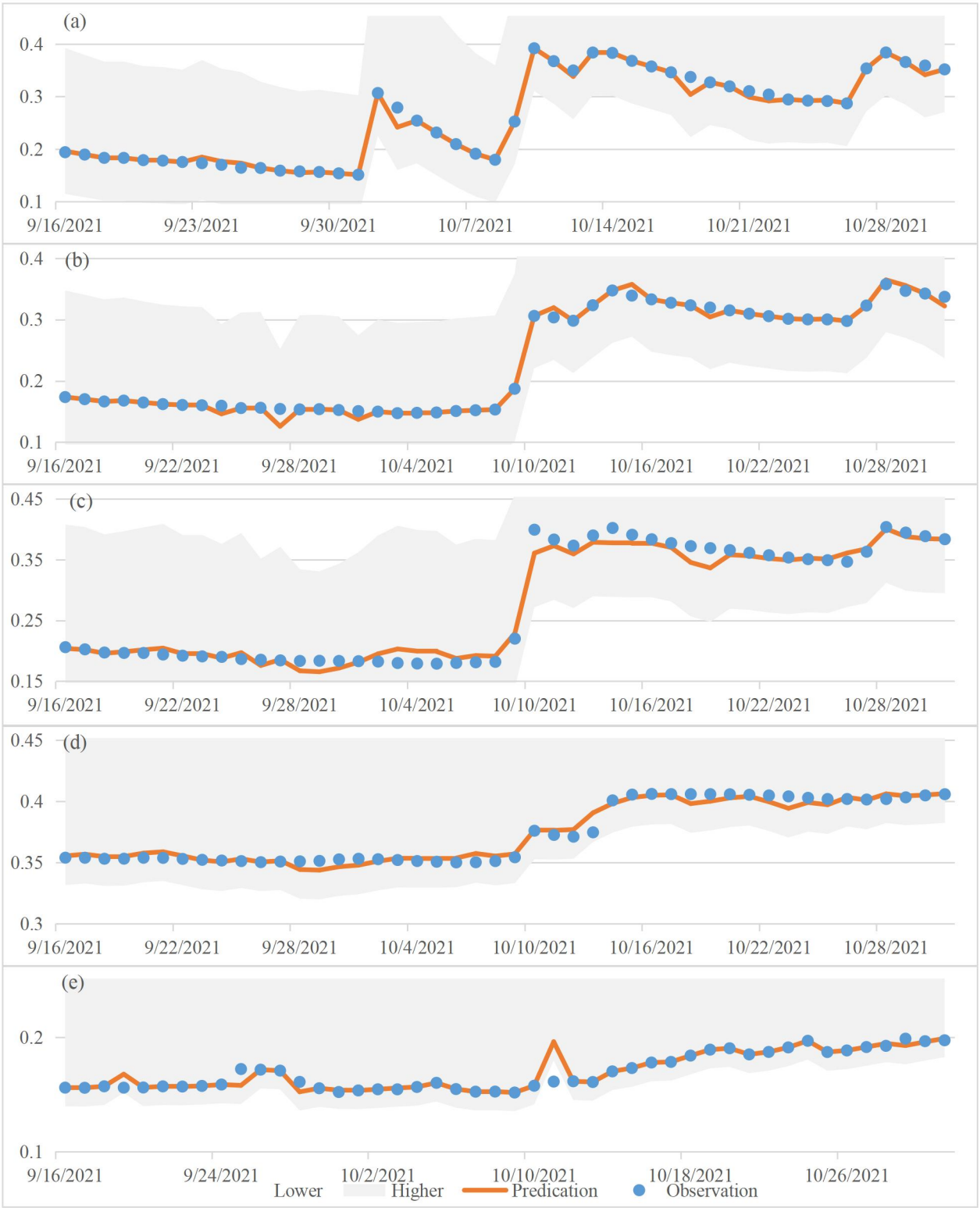

Figure B23. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Leonard. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

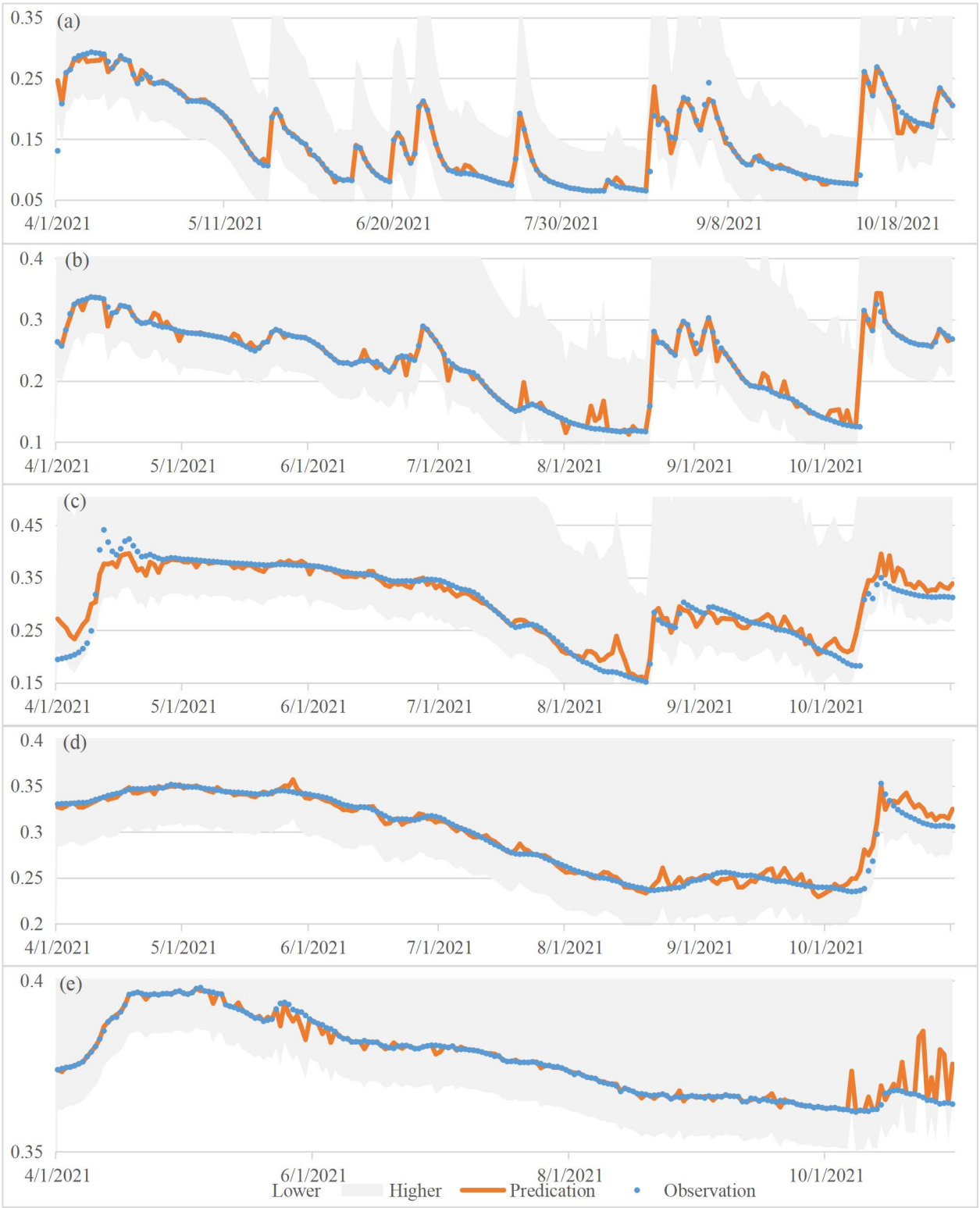

Figure B24. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Michigan. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

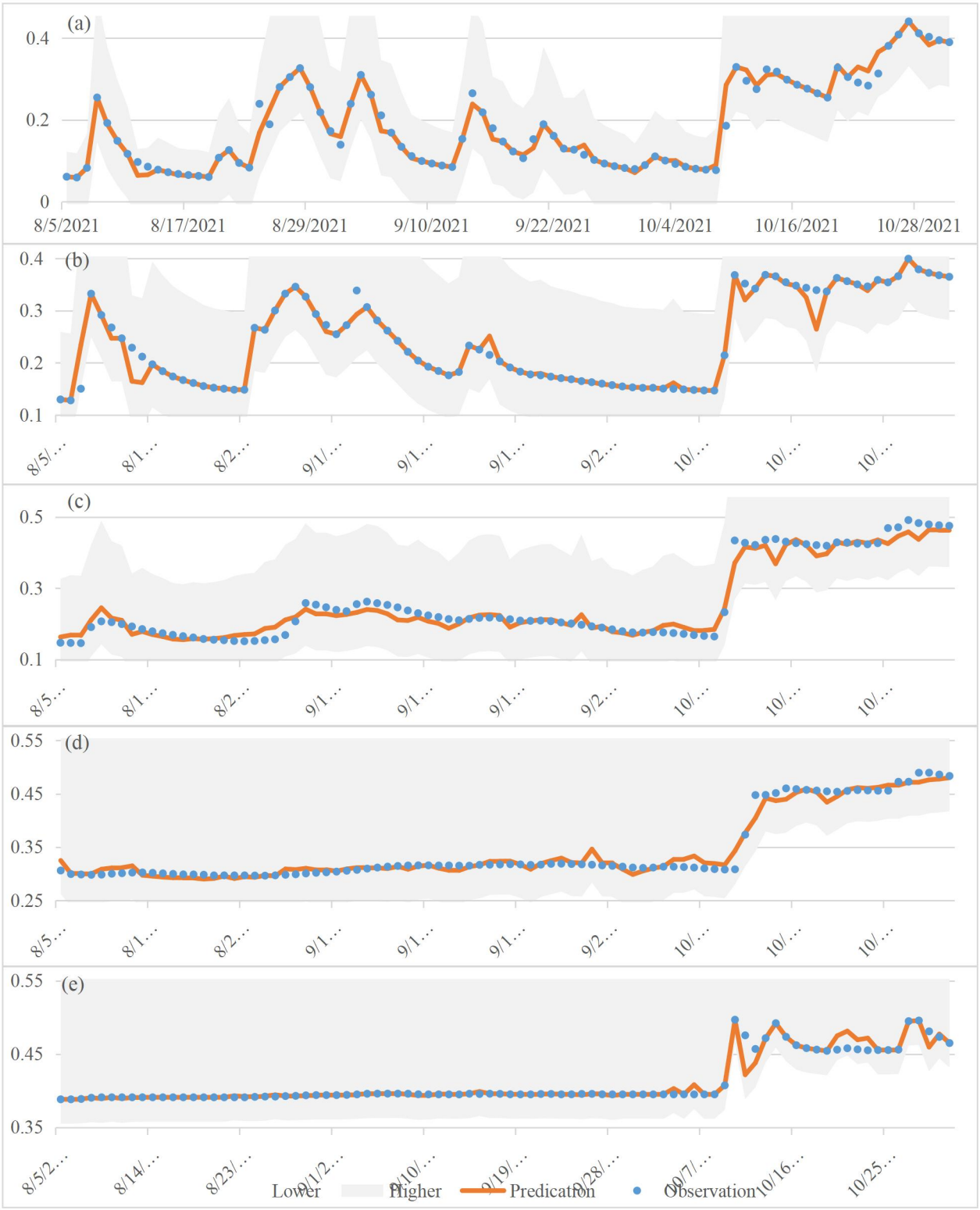

Figure B25. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Oakes. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

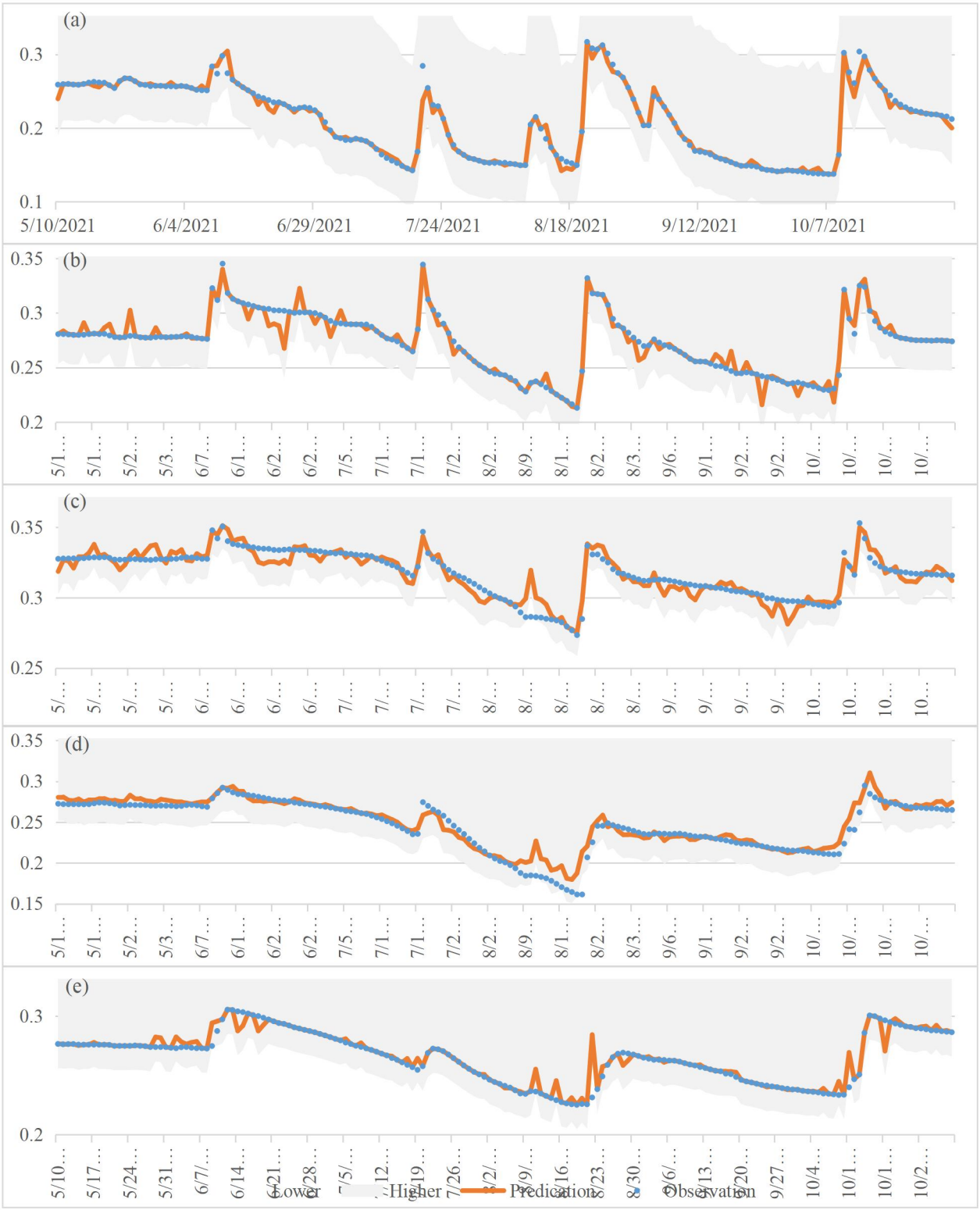

Figure B26. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Perth. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

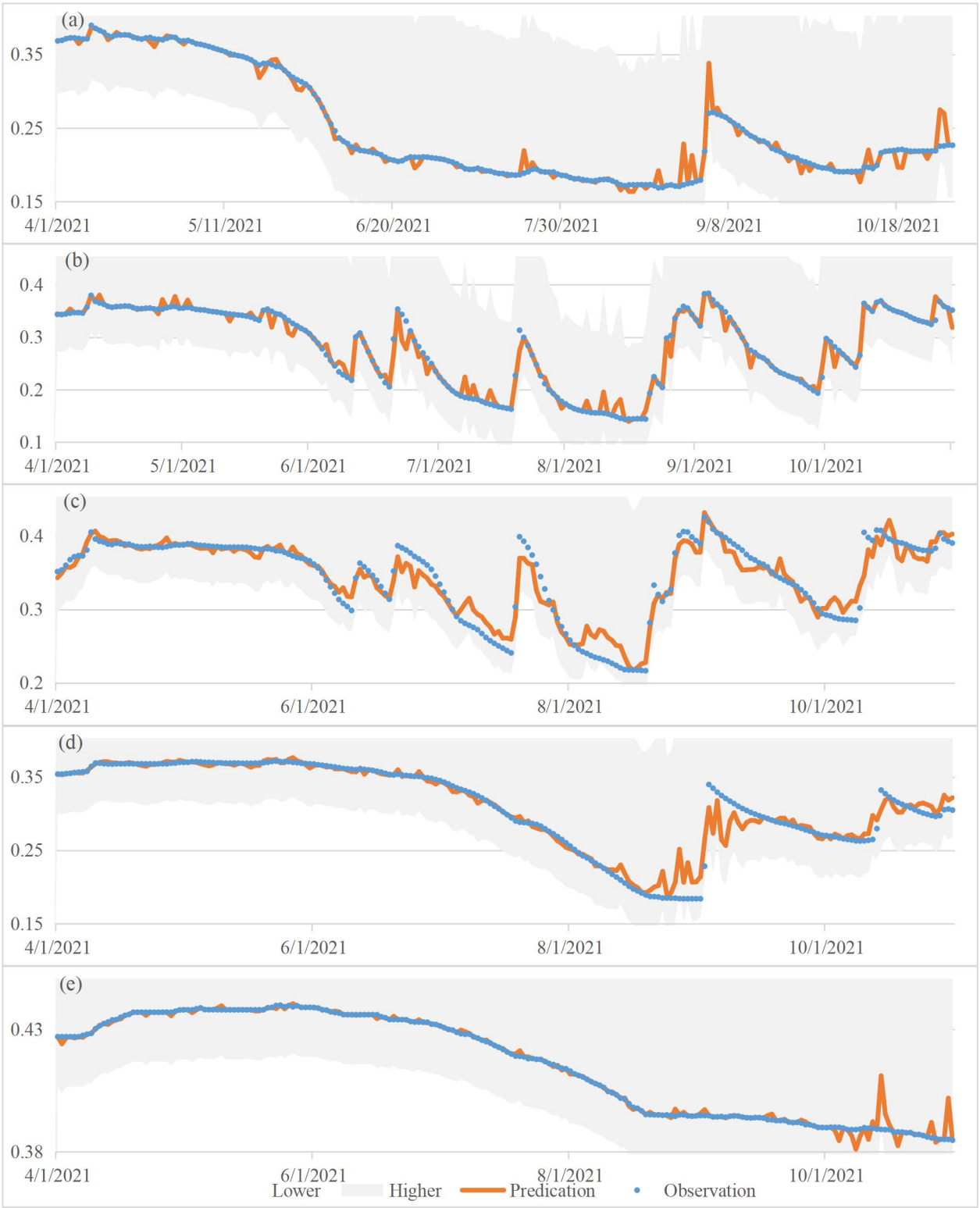

Figure B27. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Prosper. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

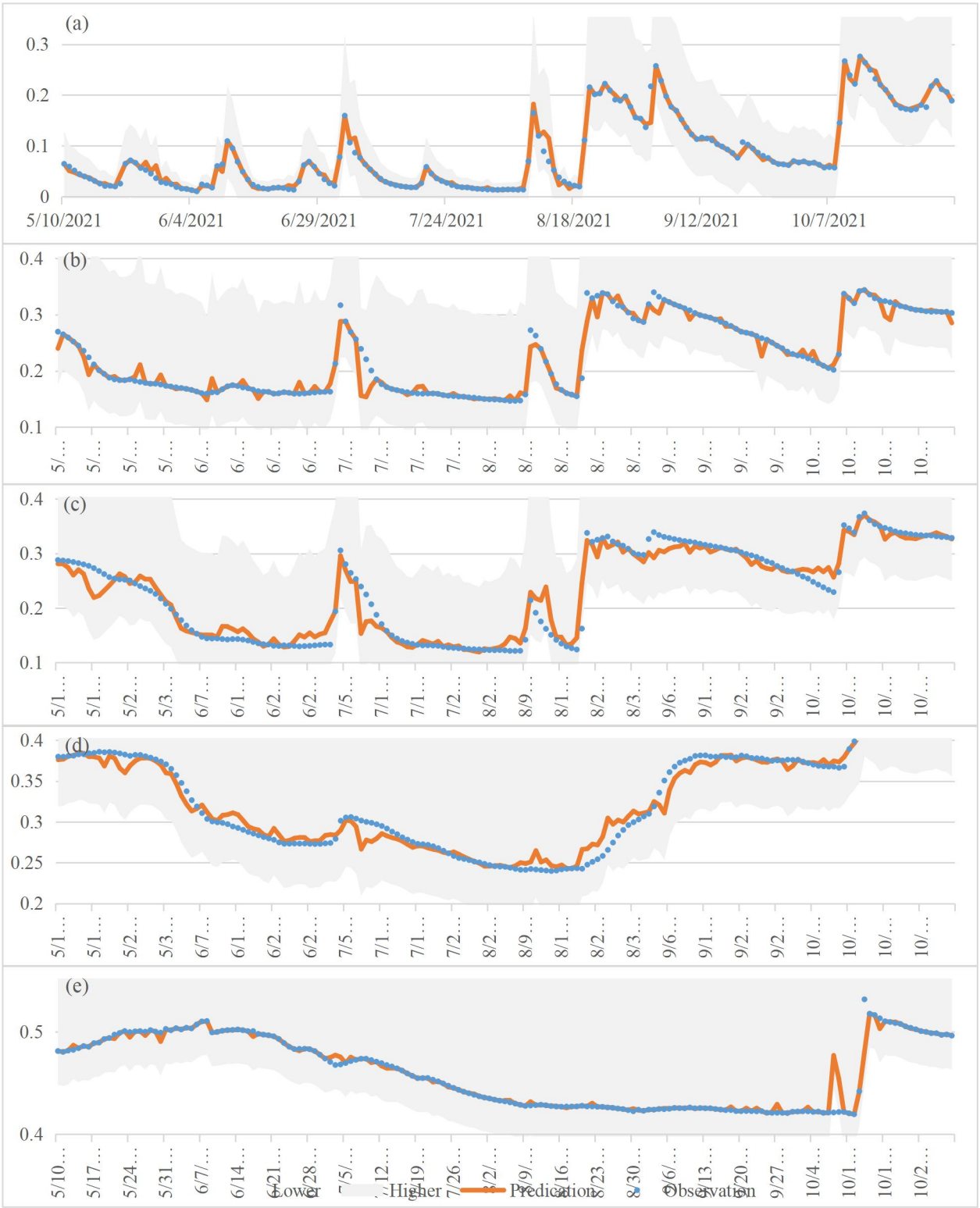

Figure B28. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Wolford. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

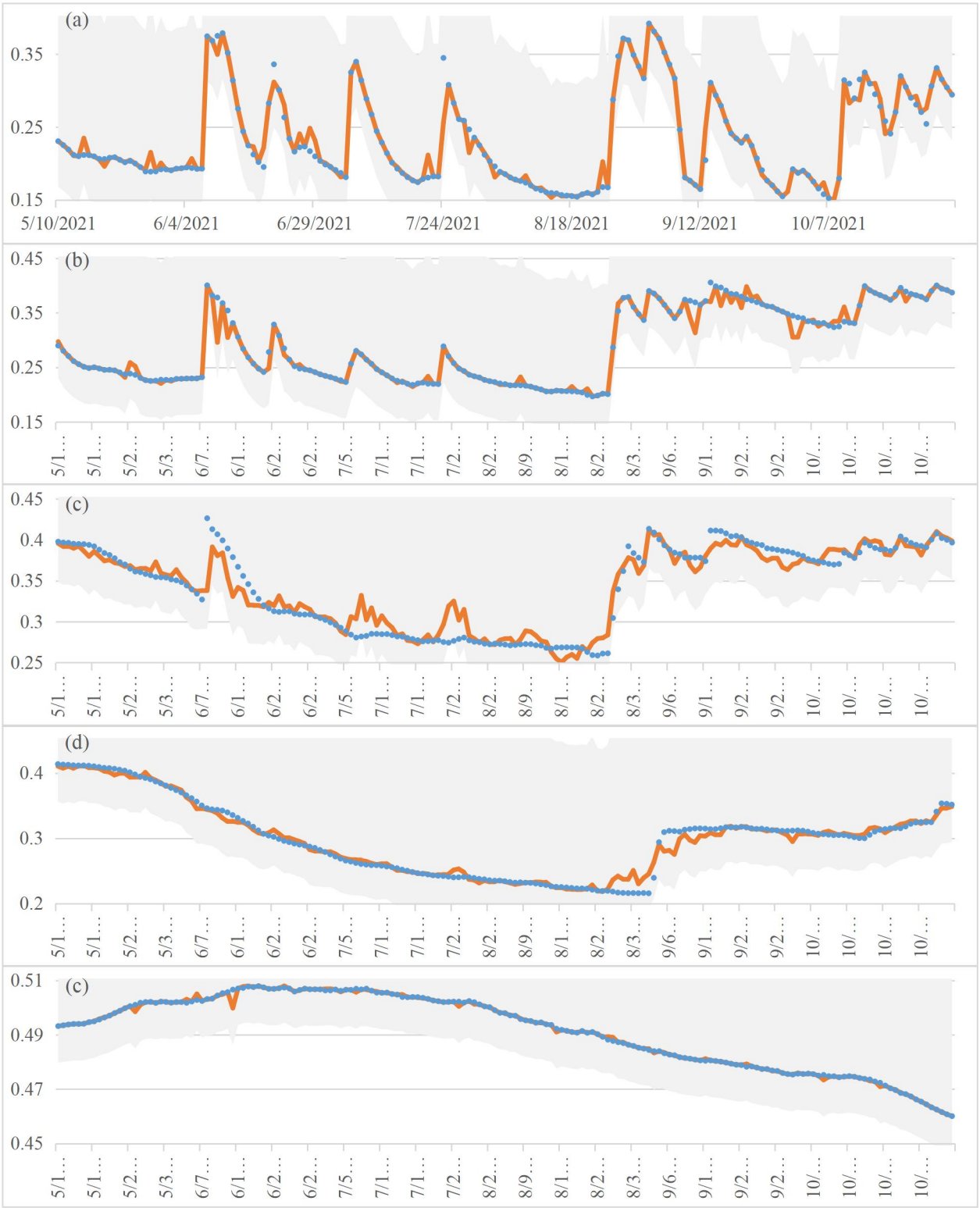

Figure B29. Graphical comparison of observed (blue dots) and GPR-simulated (orange lines) soil moisture at 5 cm (a), 10 cm (b), 20 cm (c), 50 cm (d), and 100 cm (e) in Wolverton. Notes: (1) GPR – Gaussian Process Regression. (2) The shaded areas are 95% confidence region.

APPENDIX C. TABLE OF DATA COLLECTION METHODS OF FEATURES AND TARGET

Table C1. Data collection methods of features and target.

Features and target	Data collection methods	Information source
Meteorological features		
Average bare soil temperature	Type T (copper-constantan) thermocouple	https://ndawn.ndsu.nodak.edu/help-equipment.html#raingauge
Average turf soil temperature	Type T (copper-constantan) thermocouple	
Average air temperature	Vaisala - HMP45C, HMP155A	
Average wind speed	R.M. Young Company - Heavy Duty Wind Monitor HD Model 05108	
Total solar radiation	Apogee - SP-110	
PET	Penman equation: $E_{mass} = \frac{mR_n+\rho_\alpha c_p(\delta_e)g_a}{\lambda_v(m+\gamma)}$	
Rainfall	Texas Electronics - TR-525I	
Target		
Soil Moisture	Campbell Scientific, Inc. - CS655 Soil Moisture Reflectometer	

Where, m is slope of the saturation vapor pressure curve; R_n is net irradiance; ρ_α is density of air; c_p is heat capacity of air; δ_e is vapor pressure deficit; g_a is momentum surface aerodynamic conductance; λ_v is latent heat of vaporization; γ is psychrometric constant.

APPENDIX D. CODES

D.1. MLR Codes

```
clc
clear all

data=readtable('data5cmNormalization.xlsx');
data.Properties.VariableNames =
{'L1','L2','DOY','S1','S2','S3','S4','S5','S6','S7','S8','M1','M2','M3','M4','M5',
'M6','M7','M8','M9','M10','M11','M12','M13','M14','M16','M18','5cmVWC'}

[m,n]=size(data);
percent=0.70;
idx=randperm(m);
dataTrain=data(idx(1:round(percent*m)),:);
dataTest=data(idx(round(percent*m)+1:end),:);

MLRmdl=fitlm(dataTrain,"interactions","RobustOpts","on")
anova(MLRmdl,'summary')

theta=MLRmdl.Coefficients.Estimate

yPred=predict(MLRmdl,dataTest)

dataTest1= table2array(dataTest);

MAE=mean(abs(dataTest1(:,end)-yPred))
RMSE=sqrt(mean((yPred-dataTest1(:,end)).^2))
r2=1-(sum((dataTest1(:,end)-yPred).^2)/sum((dataTest1(:,end)-
mean(dataTest1(:,end))).^2))
```

D.2. Linear SVM Codes

```
clc
clear all

data=readtable('data5cmNormalization.xlsx');
data.Properties.VariableNames =
{'L1','L2','DOY','S1','S2','S3','S4','S5','S6','S7','S8','M1','M2','M3','M4','M5',
'M6','M7','M8','M9','M10','M11','M12','M13','M14','M16','M18','5cmVWC'}


[m,n]=size(data);
percent=0.70;
idx=randperm(m);
dataTrain=data(idx(1:round(percent*m)),:);
dataTest=data(idx(round(percent*m)+1:end),:);
```

```
SVMmdl=fitrsvm(dataTrain,"5cmVWC","Standardize",true,"kernelfunction","linear")
yPred=predict(SVMmdl,dataTest);


dataTest1= table2array(dataTest);


MAE=mean(abs(dataTest1(:,end)-yPred))
RMSE=sqrt(mean((yPred-dataTest1(:,end)).^2))
r2=1-(sum((dataTest1(:,end)-yPred).^2)/sum((dataTest1(:,end)-
mean(dataTest1(:,end))).^2))
```

D.3. RBF SVM Codes

```
clc
clear all


data=readtable('data5cmNormalization.xlsx');
data.Properties.VariableNames =
{'L1','L2','DOY','S1','S2','S3','S4','S5','S6','S7','S8','M1','M2','M3','M4','M5',
'M6','M7','M8','M9','M10','M11','M12','M13','M14','M16','M18','5cmVWC'}


[m,n]=size(data);
percent=0.70;
idx=randperm(m);
dataTrain=data(idx(1:round(percent*m)),:);
dataTest=data(idx(round(percent*m)+1:end),:);


SVMmdl=fitrsvm(dataTrain,"5cmVWC","KernelFunction","rbf","KernelScale",0.08)
yPred=predict(SVMmdl,dataTest);


dataTest1= table2array(dataTest);


MAE=mean(abs(dataTest1(:,end)-yPred))
RMSE=sqrt(mean((yPred-dataTest1(:,end)).^2))
r2=1-(sum((dataTest1(:,end)-yPred).^2)/sum((dataTest1(:,end)-
mean(dataTest1(:,end))).^2))
```

D.4. ARD Exponential GPR Codes

```
clc
clear all


data=readtable('data5cmNormalization.xlsx');
```

```
data.Properties.VariableNames =
{'L1','L2','DOY','S1','S2','S3','S4','S5','S6','S7','S8','M1','M2','M3','M4','M5',
'M6','M7','M8','M9','M10','M11','M12','M13','M14','M16','M18','5cmVWC'}


[m,n]=size(data);
percent=0.70;
idx=randperm(m);
dataTrain=data(idx(1:round(percent*m)),:);
dataTest=data(idx(round(percent*m)+1:end),:);
%Xtrain=dataTrain(:,1:n-1);
%Ytrain=dataTrain(:,n);


GPRmdl5cm=fitrgp(dataTrain,"5cmVWC","KernelFunction","ardexponential","PredictMet
hod","exact")
%save('gpr5cm.mat','GPRmdl5cm');
yPred=predict(GPRmdl5cm,dataTest);
GPRmdlMSE=loss(GPRmdl5cm,dataTest)
[yPred,~,yInt]=predict(GPRmdl5cm,dataTest);


dataTest1= table2array(dataTest);


MAE=mean(abs(dataTest1(:,end)-yPred))
RMSE=sqrt(mean((yPred-dataTest1(:,end)).^2))
r2=1-(sum((dataTest1(:,end)-yPred).^2)/sum((dataTest1(:,end)-
mean(dataTest1(:,end))).^2))


xCampbell=data(1:332,1:end-1);
yCampbell=data(1:332,end);
yPredCampbell=predict(GPRmdl5cm,xCampbell);
yCampbell=table2array(yCampbell);
r2Campbell=1-(sum((yCampbell-yPredCampbell).^2)/sum((yCampbell-
mean(yCampbell)).^2))


xCarrington=data(333:620,1:end-1);
yCarrington=data(333:620,end);
yPredCarrington=predict(GPRmdl5cm,xCarrington);
yCarrington=table2array(yCarrington);
r2Carrington=1-(sum((yCarrington-yPredCarrington).^2)/sum((yCarrington-
mean(yCarrington)).^2))


xFargo=data(621:1146,1:end-1);
yFargo=data(621:1146,end);
yPredFargo=predict(GPRmdl5cm,xFargo);
```

```
yFargo=table2array(yFargo)
r2Fargo=1-(sum((yFargo-yPredFargo).^2)/sum((yFargo-mean(yFargo)).^2))


xFox=data(1147:1440,1:end-1);
yFox=data(1147:1440,end);
yPredFox=predict(GPRmdl5cm,xFox);
yFox=table2array(yFox)
r2Fox=1-(sum((yFox-yPredFox).^2)/sum((yFox-mean(yFox)).^2))


xGrandForks=data(1441:1735,1:end-1);
yGrandForks=data(1441:1735,end);
yPredGrandForks=predict(GPRmdl5cm,xGrandForks);
yGrandForks=table2array(yGrandForks)
r2GrandForks=1-(sum((yGrandForks-yPredGrandForks).^2)/sum((yGrandForks-
mean(yGrandForks)).^2))


xHillsboro=data(1736:2052,1:end-1);
yHillsboro=data(1736:2052,end);
yPredHillsboro=predict(GPRmdl5cm,xHillsboro);
yHillsboro=table2array(yHillsboro)
r2Hillsboro=1-(sum((yHillsboro-yPredHillsboro).^2)/sum((yHillsboro-
mean(yHillsboro)).^2))


xMavie=data(2053:2299,1:end-1);
yMavie=data(2053:2299,end);
yPredMavie=predict(GPRmdl5cm,xMavie);
yMavie=table2array(yMavie)
r2Mavie=1-(sum((yMavie-yPredMavie).^2)/sum((yMavie-mean(yMavie)).^2))


xMooreton=data(2300:2366,1:end-1);
yMooreton=data(2300:2366,end);
yPredMooreton=predict(GPRmdl5cm,xMooreton);
yMooreton=table2array(yMooreton)
r2Mooreton=1-(sum((yMooreton-yPredMooreton).^2)/sum((yMooreton-
mean(yMooreton)).^2))


xPekin=data(2367:2662,1:end-1);
yPekin=data(2367:2662,end);
yPredPekin=predict(GPRmdl5cm,xPekin);
yPekin=table2array(yPekin)
r2Pekin=1-(sum((yPekin-yPredPekin).^2)/sum((yPekin-mean(yPekin)).^2))


xSabin=data(2663:3026,1:end-1);
```

```
ySabin=data(2663:3026,end);
yPredSabin=predict(GPRmdl5cm,xSabin);
ySabin=table2array(ySabin)
r2Sabin=1-(sum((ySabin-yPredSabin).^2)/sum((ySabin-mean(ySabin)).^2))


xWaukon=data(3027:3087,1:end-1);
yWaukon=data(3027:3087,end);
yPredWaukon=predict(GPRmdl5cm,xWaukon);
yWaukon=table2array(yWaukon)
r2Waukon=1-(sum((yWaukon-yPredWaukon).^2)/sum((yWaukon-mean(yWaukon)).^2))


xGrafton=data(3088:3424,1:end-1);
yGrafton=data(3088:3424,end);
yPredGrafton=predict(GPRmdl5cm,xGrafton);
yGrafton=table2array(yGrafton)
r2Grafton=1-(sum((yGrafton-yPredGrafton).^2)/sum((yGrafton-mean(yGrafton)).^2))


xAda=data(3425:3481,1:end-1);
yAda=data(3425:3481,end);
yPredAda=predict(GPRmdl5cm,xAda);
yAda=table2array(yAda)
r2Ada=1-(sum((yAda-yPredAda).^2)/sum((yAda-mean(yAda)).^2))


xAlvarado=data(3482:3655,1:end-1);
yAlvarado=data(3482:3655,end);
yPredAlvarado=predict(GPRmdl5cm,xAlvarado);
yAlvarado=table2array(yAlvarado)
r2Alvarado=1-(sum((yAlvarado-yPredAlvarado).^2)/sum((yAlvarado-
mean(yAlvarado)).^2))


xAyr=data(3656:3830,1:end-1);
yAyr=data(3656:3830,end);
yPredAyr=predict(GPRmdl5cm,xAyr);
yAyr=table2array(yAyr)
r2Ayr=1-(sum((yAyr-yPredAyr).^2)/sum((yAyr-mean(yAyr)).^2))


xClyde=data(3831:4005,1:end-1);
yClyde=data(3831:4005,end);
yPredClyde=predict(GPRmdl5cm,xClyde);
yClyde=table2array(yClyde)
r2Clyde=1-(sum((yClyde-yPredClyde).^2)/sum((yClyde-mean(yClyde)).^2))


xCourtenay=data(4006:4532,1:end-1);
```

```
yCourtenay=data(4006:4532,end);
yPredCourtenay=predict(GPRmdl5cm,xCourtenay);
yCourtenay=table2array(yCourtenay)
r2Courtenay=1-(sum((yCourtenay-yPredCourtenay).^2)/sum((yCourtenay-
mean(yCourtenay)).^2))


xCrystal=data(4533:4758,1:end-1);
yCrystal=data(4533:4758,end);
yPredCrystal=predict(GPRmdl5cm,xCrystal);
yCrystal=table2array(yCrystal)
r2Crystal=1-(sum((yCrystal-yPredCrystal).^2)/sum((yCrystal-mean(yCrystal)).^2))


xDenhoff=data(4759:4862,1:end-1);
yDenhoff=data(4759:4862,end);
yPredDenhoff=predict(GPRmdl5cm,xDenhoff);
yDenhoff=table2array(yDenhoff)
r2Denhoff=1-(sum((yDenhoff-yPredDenhoff).^2)/sum((yDenhoff-mean(yDenhoff)).^2))


xEmerado=data(4863:5048,1:end-1);
yEmerado=data(4863:5048,end);
yPredEmerado=predict(GPRmdl5cm,xEmerado);
yEmerado=table2array(yEmerado)
r2Emerado=1-(sum((yEmerado-yPredEmerado).^2)/sum((yEmerado-mean(yEmerado)).^2))


xGlyndon=data(5049:5223,1:end-1);
yGlyndon=data(5049:5223,end);
yPredGlyndon=predict(GPRmdl5cm,xGlyndon);
yGlyndon=table2array(yGlyndon)
r2Glyndon=1-(sum((yGlyndon-yPredGlyndon).^2)/sum((yGlyndon-mean(yGlyndon)).^2))


xHumboldt=data(5224:5261,1:end-1);
yHumboldt=data(5224:5261,end);
yPredHumboldt=predict(GPRmdl5cm,xHumboldt);
yHumboldt=table2array(yHumboldt)
r2Humboldt=1-(sum((yHumboldt-yPredHumboldt).^2)/sum((yHumboldt-
mean(yHumboldt)).^2))


xLeonard=data(5262:5307,1:end-1);
yLeonard=data(5262:5307,end);
yPredLeonard=predict(GPRmdl5cm,xLeonard);
yLeonard=table2array(yLeonard)
r2Leonard=1-(sum((yLeonard-yPredLeonard).^2)/sum((yLeonard-mean(yLeonard)).^2))
```

```
xMichigan=data(5308:5548,1:end-1);
yMichigan=data(5308:5548,end);
yPredMichigan=predict(GPRmdl5cm,xMichigan);
yMichigan=table2array(yMichigan)
r2Michigan=1-(sum((yMichigan-yPredMichigan).^2)/sum((yMichigan-
mean(yMichigan)).^2))


xOakes=data(5549:5636,1:end-1);
yOakes=data(5549:5636,end);
yPredOakes=predict(GPRmdl5cm,xOakes);
yOakes=table2array(yOakes)
r2Oakes=1-(sum((yOakes-yPredOakes).^2)/sum((yOakes-mean(yOakes)).^2))


xPerth=data(5637:5811,1:end-1);
yPerth=data(5637:5811,end);
yPredPerth=predict(GPRmdl5cm,xPerth);
yPerth=table2array(yPerth)
r2Perth=1-(sum((yPerth-yPredPerth).^2)/sum((yPerth-mean(yPerth)).^2))


xProsper=data(5812:6037,1:end-1);
yProsper=data(5812:6037,end);
yPredProsper=predict(GPRmdl5cm,xProsper);
yProsper=table2array(yProsper)
r2Prosper=1-(sum((yProsper-yPredProsper).^2)/sum((yProsper-mean(yProsper)).^2))


xWolford=data(6038:6212,1:end-1);
yWolford=data(6038:6212,end);
yPredWolford=predict(GPRmdl5cm,xWolford);
yWolford=table2array(yWolford)
r2Wolford=1-(sum((yWolford-yPredWolford).^2)/sum((yWolford-mean(yWolford)).^2))


xWolverton=data(6213:6387,1:end-1);
yWolverton=data(6213:6387,end);
yPredWolverton=predict(GPRmdl5cm,xWolverton);
yWolverton=table2array(yWolverton)
r2Wolverton=1-(sum((yWolverton-yPredWolverton).^2)/sum((yWolverton-
mean(yWolverton)).^2))
```

D.5. Squared Exponential GPR Codes

```
clc
clear all


data=readtable('data5cmNormalization.xlsx');
```

```
data.Properties.VariableNames =
{'L1','L2','DOY','S1','S2','S3','S4','S5','S6','S7','S8','M1','M2','M3','M4','M5',
'M6','M7','M8','M9','M10','M11','M12','M13','M14','M16','M18','5cmVWC'}


[m,n]=size(data);
percent=0.70;
idx=randperm(m);
dataTrain=data(idx(1:round(percent*m)),:);
dataTest=data(idx(round(percent*m)+1:end),:);
%Xtrain=dataTrain(:,1:n-1);
%Ytrain=dataTrain(:,n);


GPRmdl=fitrgp(dataTrain,"5cmVWC","KernelFunction","squaredexponential","Sigma",0.
08)
save('trainedModel.mat','GPRmdl');
yPred=predict(GPRmdl,dataTest);
GPRmdlMSE=loss(GPRmdl,dataTest)
[yPred,~,yInt]=predict(GPRmdl,dataTest);


dataTest1= table2array(dataTest);


MAE=mean(abs(dataTest1(:,end)-yPred))
RMSE=sqrt(mean((yPred-dataTest1(:,end)).^2))
r2=1-(sum((dataTest1(:,end)-yPred).^2)/sum((dataTest1(:,end)-
mean(dataTest1(:,end))).^2))
```

D.6. Deep Learning

```
import pathlib
#from keras.optimizers import sgd
#from tensorflow.keras.optimizers import sgd
from keras.optimizers import gradient_descent_v2
from sklearn.metrics import r2_score
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
import tensorflow as tf
import matplotlib
from tensorflow import keras
from tensorflow.keras import layers
from keras import backend as K
print(tf.__version__)
import tensorflow_docs as tfdocs
import tensorflow_docs.plots
import tensorflow_docs.modeling
```

```
#GPU Setting up############################################################
#config = tf.ConfigProto()
config =tf.compat.v1.ConfigProto()
config.gpu_options.allow_growth = True
#session = tf.Session(config=config)
session = tf.compat.v1.Session(config=config)

#gpu_options = tf.GPUOptions(per_process_gpu_memory_fraction=0.7)
gpu_options =tf.compat.v1.GPUOptions(per_process_gpu_memory_fraction=0.7)
#sess = tf.Session(config=tf.ConfigProto(gpu_options=gpu_options))
sess
=tf.compat.v1.Session(config=tf.compat.v1.ConfigProto(gpu_options=gpu_options))

# Start Reading Files#######################################################
dataframe = pd.read_excel("data_5cm.xlsx")
#dataframe = pd.read_excel("data_5cm_29 weather stations1.xls")
dataframe.head()
dataset = dataframe.copy()
dataset.info()

# CSV Format with Comma Removal ###########################################
dataset = pd.get_dummies(dataset, prefix='', prefix_sep='')
print(dataset)

#####################################################
labels=dataset['5cmVWC']
print(labels)

#Data Normalization#################################################
import numpy as np
def NormalizeData(dataset):
    return (dataset - np.min(dataset)) / (np.max(dataset) - np.min(dataset))
    #return(dataset-np.mean(dataset))/(np.std(dataset))

features1 = dataset.iloc[:,1:27]
features=NormalizeData(features1)
print(features)

sns.set(rc={'figure.figsize':(11.7,8.27)})
sns.distplot(dataset['5cmVWC'], bins=30)
plt.savefig('5cm VWC Density Map5.jpg',bbox_inches='tight')
plt.show()

#########################################################
from sklearn.model_selection import train_test_split
x=features
y=labels

train_dataset, test_dataset, train_labels, test_labels = train_test_split(x, y,
```

```
test_size=0.3, random_state=64, shuffle=True)
print(train_dataset.shape)
print(test_dataset.shape)
print(train_labels.shape)
print(test_labels.shape)


# ## Heat Map ########################################################
import seaborn as sns
import matplotlib.pyplot as plt
corr = dataset.corr().round(2)
plt.figure(figsize=(30, 25))
sns.heatmap(corr,
            xticklabels=corr.columns.values,
            yticklabels=corr.columns.values, annot=True, cmap="YlGnBu",)
#plt.savefig('5cm heat map.png',bbox_inches='tight')
plt.savefig('5cm Heat Map5.jpg',bbox_inches='tight')
plt.show()

# Correlation Chart################################################

sns.pairplot(dataset[["5cmVWC","L1","L2","DOY","S1","S2","S3", "S4", "S5", "S6",
"S7", "S8", "M1", "M2", "M3", "M4", "M5", "M6", "M7", "M8", "M9", "M10", "M11",
"M12", "M13", "M14", "M16", "M18"]], diag_kind="kde", height=1.7)
plt.savefig('5cm Correlation Map5.png',bbox_inches='tight')
plt.show()

# Building Models#################################################
def build_model():
  model = keras.Sequential([
    layers.Dense(10, activation='relu', input_shape=[len(train_dataset.keys())]),
    layers.Dense(10, activation='relu'),
    layers.Dense(15, activation='relu'),
    layers.Dense(1)
  ])
#
  optimizer = tf.keras.optimizers.RMSprop(0.001)

  model.compile(loss='mean_squared_error',
                optimizer=optimizer,
                metrics=['mean_absolute_error', 'mean_squared_error'])
  return model

model = build_model()
model.summary()
print(model.summary)

# Run models#######################################################
EPOCHS = 2000
```

```
history = model.fit(
  train_dataset, train_labels,
  epochs=EPOCHS, validation_split = 0.25, verbose=0,
  callbacks=[tfdocs.modeling.EpochDots()])
#
hist = pd.DataFrame(history.history)
hist['epoch'] = history.epoch
print(hist)
# # #
plt.plot(history.history['mean_absolute_error'])
plt.plot(history.history['val_mean_absolute_error'])
plt.title('model accuracy')
plt.ylabel('mean_absolute_error')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.savefig('5cm Model Accuracy5.jpg',bbox_inches='tight')
plt.show()
# summarize history for loss
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('model loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.savefig('5cm Model Loss5.png',bbox_inches='tight')
plt.show()

# The results of  model regression########################################
loss, mean_absolute_error, mean_squared_error = model.evaluate(test_dataset,
test_labels, verbose=2)
print("Testing set Mean mean_absolute_error Error: {:5.2f}
Yield".format(mean_absolute_error))

# Testing Results#########################################################
test_predictions = model.predict(test_dataset).flatten()

a = plt.axes(aspect='equal')
plt.scatter(test_labels, test_predictions, marker='o')
plt.xlabel('True Values [5cmVWC]')
plt.ylabel('Predictions [5cmVWC]')
lims = [0, 1]
plt.xlim(lims)
plt.ylim(lims)
_ = plt.plot(lims, lims)
plt.savefig('5cm Ture Values vs Predication Values5.jpg',bbox_inches='tight')
plt.show()

#
error = test_predictions - test_labels
plt.hist(error, bins = 25)
```

```
plt.xlabel("Prediction Error [5cmVWC]")
_ = plt.ylabel("Count")
plt.savefig('5cm Prediction Error5.png',bbox_inches='tight')
plt.show()

# R2 ##########################################################
# R2=r2_score(test_labels,test_predictions)
# print(R2)
from sklearn.metrics import r2_score
pred_acc = r2_score(test_labels, test_predictions)
print('pred_acc',pred_acc)
##############################################################

plt.figure(figsize=(8, 4), dpi=80)
plt.plot(range(len(test_labels)), test_labels, ls='-.',lw=2,c='r',label='Ture
Value')
plt.plot(range(len(test_predictions)), test_predictions, ls='-
',lw=2,c='b',label='Predict Value')

# Drawing the grid#####################################################
plt.grid(alpha=0.4, linestyle=':')
plt.legend()
plt.xlabel('Features')
plt.ylabel('5cmVWC')
plt.savefig('5cm Ture Values vs Predication Values Scatter
plot5.png',bbox_inches='tight')
plt.show()

# LinearRegression ####################################################
r2 = r2_score(test_labels, test_predictions)
print(r2)
```