Supplementary Material to paper 'Systematic comparison of five machine-learning methods in classification and interpolation of soil particle size fractions using different transformed data'

Mo Zhang, Wenjiao Shi

This supplementary material consists of equations of methods, tables and prediction maps in the paper and in the following 5 sections. **Section S1** is the equations description of machine-learning models. **Section S2** shows parameter adjustment of machine-learning models. **Section S3** includes the uncertainty assessment of soil PSFs interpolation. **Section S4** includes the prediction maps of silt and clay fractions. **Section S5** demonstrates the indirect classification maps using ALR and CLR transformation methods.

Supplementary Material

Section S1 The equations description of machine-learning models

For K-nearest neighbor (KNN), for a train set of observed data $L = \{(y_i, x_i), i = 1, \dots, n_L\}$, class $y_i \in \{1, \dots, c\}$, and the predictor values $x'_i = (x_{i1}, \dots, x_{ip})$. For a new observation (y, x), the nearest neighbor $(y_{(1)}, x_{(1)})$ is based on the distance function which is as follows:

$$d(x, x_{(1)}) = min_i(\, d(x, x_i)), \tag{S1.1}$$

and $\hat{y} = y_{(1)}$ refers to the nearest neighbor, which is the prediction for y. Value $x_{(j)}$ and $y_{(j)}$ are the jth nearest neighbor of x and class of training set, respectively.

For multilayer perceptron neural network (MLP), each neuron j sums input environmental covariate in our study x_i after multiplying them by the connection weights w_{ji} respectively, and calculates its output y_j (soil PSFs components or texture class) as a function of the sum:

$$y_j = f(\sum w_{ji} x_i), \tag{S1.2}$$

where f is the activation function, which can be a linear or logistic function. The sum of squared differences between the predicted values and observed values of the output results of neurons E is defined as follows:

$$E = \frac{1}{2} \sum_j (y_{pj} - y_{oj})^2, \tag{S1.3}$$

where y_{pj} and y_{oj} is the predicted and observed value of output neuron j, respectively. Each w_{ji} is adjusted to reduce E and the adjustment of w_{ji} depends on the training algorithm.

For random forest (RF), the equations for Gini index and minimizing the sum of the squares of the mean deviations (M) are as follows:

$$Gini = 1 - \sum_{k=1}^{K} p_k^2, \tag{S1.4}$$

$$Gini(D, A) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2), \tag{S1.5}$$

$$\mathrm{M} = min_A[\, min_{c_1} \sum_{x_i \in D_1(A)} (y_i - c_1)^2 + min_{c_2} \sum_{x_i \in D_2(A)} (y_i - c_2)^2], \tag{S1.6}$$

where p_k refers to the proportion of kth class in the data set on the current node, for feature $A = a$, data set D is divided into two parts (D_1 and D_2), D_1 describes the data set which meets the condition

$A = a$ and D_2 is the opposite of D_1; $Gini(D, A)$ represents the uncertainty of set D after binary split; y_i is the predicted value of input value x_i; c_1 and c_2 is the mean of data set D_1 and D_2, respectively.

In support vector machine (SVM), for a data set $\{x_i, y_i\}$, $i = 1, \dots, k$, $x \in R$ and x refers to an n-dimensional vector, $y \in \{-1, +1\}$ is the class corresponding to x, the equation for calculating a hyperplane of SVM is defined as follows:

$$\min_{w,b,\xi} \frac{1}{2} w^T \times w + C \sum_{i=1}^{k} \xi_i,$$

$$\text{s.t. } y_i(w^T \times \phi(x_i) + b) \geq 1 - \xi_i,\ \xi_i \geq 0,\ i = 1, \dots, k, \tag{S1.7}$$

where $\phi(x_i)$ refers to the mapping from the input space to the feature space, $C > 0$ is penalty factor (cost), w, b, and ξ are the parameters need to be optimized during the process of model training, which can be determined by the Lagrange multipliers:

$$f(x) = sgn(y_i a_i k(x_i, x) + b^*), \tag{S1.8}$$

where a_i refers to the support vector, $k(x_i, x)$ refers to the kernel function, and b^* is the bias.

For extreme gradient boosting (XGB), the general prediction function at step t is defined as follows:

$$f_i^{(t)} = \sum_{k=1}^{t} f_k(x_i) = f_i^{(t-1)} + f_t(x_i), \tag{S1.9}$$

where $f_t(x_i)$ refers to the tree (learner) at step t, $f_i^{(t)}$ and $f_i^{(t-1)}$ refer to the predicted values at steps t and t − 1, and x_i is the input value.

$$Obj^{(t)} = \sum_{k=1}^{n} l(\overline{y_i}, y_i) + \sum_{k=1}^{n} \Omega(f_i), \tag{S1.10}$$

where $Obj^{(t)}$ is the regularized objective, $\overline{y_i}$ and y_i refer to the prediction value and observed value, l refers to the loss function, n is the number of data set, and Ω refers to the regularization term, which equation is defined as follows:

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \|\omega\|^2, \tag{S1.11}$$

where ω refers to the weight vector, T denotes the total number of features, λ is the regularization term, and γ is the minimum loss.

Supplementary Material

Section S2 Parameter adjustment of machine-learning models

For the parameter adjustment in Table S2.1, for KNN, the kmax was 15; the distance was 1; the kernel was rectangular. For MLP, the size ranged between 5 and 10. For RF, the ntree was 1000; the mtry fluctuated from 9 to 11. For SVM, the gamma was 0.01; the cost was 1. For XGB, the max_depth was 3 – 4; the eta was 0.05 – 0.1; the colsample_bytree was 0.6 – 0.8, the nrounds was 30; the subsample was 0.8 – 1; the gamma was 0 – 0.4; the min_child_weight was 0.6 – 0.8.

Table S2.1 Adjusted parameters for different machine-learning methods. “rectan” is short for rectangular, “opt” is short for optimal and “ep” is short for epanechnikov.

Models	Parameters	alr1	alr2	clr1	clr2	clr3	ilr1	ilr2	sand	silt	clay	class
KNN	kmax	13	13	14	14	15	15	14	14	15	15	15
	distance	1	1	1	1	1	1	1	1	1	1	1
	kernel	rectan	rectan	rectan	rectan	rectan	rectan	opt	rectan	rectan	ep	rectan
MLP	size	5	5	5	5	5	5	5	10	10	10	5
RF	ntree	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
	mtry	9	9	9	9	9	9	9	6	11	11	7
SVM	gamma	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.01
	cost	1	1	1	1	1	1	1	1	1	1	1
XGB	max_depth	3	3	3	3	3	3	3	3	4	3	4
	eta	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.05	0.05	0.05	0.1
	colsample_bytree	0.6	0.6	1	0.8	0.6	0.6	1	0.6	0.6	0.6	0.8
	nrounds	20	30	40	40	30	20	30	30	30	30	30
	subsample	1	1	0.8	1	0.6	0.8	0.8	0.6	0.6	0.6	1
	gamma	0.6	1	0.7	0.4	0.7	0	0.3	0.8	0.8	0.8	0.1
	min_child_weight	0.6	0.8	0.6	1	0.6	1	1	0.8	0.8	0.8	0.6

Supplementary Material

Section S3 Uncertainty assessment of soil PSFs interpolation

For the assessment of the uncertainties of models, Table S3.1 showed that ORI delivered lower SDs than those of log ratio methods among five machine-learning models for sand, silt and clay. Moreover, the ranges of 95 % confidence interval (CI) of indicators were also computed, which indicated relatively low values compared with assessment indicators (Table S3.1). For KNN, MLP and RF, ORI method showed lower values of CI of RMSE, MAE and R^2 than those of log ratio methods, and for SVM and XGB, SVM_CLR and XGB_CLR revealed slight better performance compared with ORI of sand (CI_RMSE: 0.49 %; CI_MAE: 0.33 %) and silt (CI_MAE: 0.44 %), respectively. For the values of the ranges of 95 % CI of AD and STRESS, all models generated the same results (AD: 0.03, STRESS: 0.01) aside from RF_ILR (AD: 0.02), showing better performance. Thus, the estimators' variabilities had reasonable order of magnitudes for the values of the estimates and these indicators were representative of the actual errors on independent test sets.

Table S3.1. The standard deviation of prediction, the ranges of 95 % confidence interval (CI) of indicators for different machine-learning models combined with original and transformed data.

	SD			CI_RMSE (%)			CI_MAE (%)			CI_R^2 (%)			CI_AD	CI_STRESS
	Sand	Silt	Clay	Sand	Silt	Clay	Sand	Silt	Clay	Sand	Silt	Clay		
KNN_ALR	0.18	0.14	0.08	0.71	0.65	0.25	0.51	0.44	**0.16**	4.45	5.03	4.18	0.03	**0.01**
KNN_CLR	0.18	0.14	0.08	0.71	0.64	0.26	0.47	0.41	**0.16**	4.57	4.95	4.23	0.03	**0.01**
KNN_ILR	0.18	0.14	0.08	0.73	0.64	0.27	0.48	0.41	**0.16**	4.78	5.18	4.4	0.03	**0.01**
KNN_ORI	**0.15**	**0.11**	0.07	0.55	0.51	0.28	0.38	0.37	0.19	3.41	3.48	4	0.03	**0.01**
MLP_ALR	0.17	0.13	**0.06**	0.65	0.67	0.33	0.38	0.41	0.2	4.21	5.07	5.44	0.03	**0.01**
MLP_CLR	0.16	0.13	**0.06**	0.64	0.65	0.32	0.38	0.41	0.19	4.07	4.96	5.12	0.03	**0.01**
MLP_ILR	0.16	0.13	**0.06**	0.64	0.65	0.32	0.37	0.41	0.2	4.04	4.95	5.04	0.03	**0.01**
MLP_ORI	**0.15**	**0.11**	**0.06**	0.65	0.58	0.23	0.37	0.4	0.17	3.72	4.02	**2.72**	0.03	**0.01**
RF_ALR	0.18	0.15	0.08	0.62	0.54	0.25	0.42	0.38	0.17	4.03	3.91	4.03	0.03	**0.01**
RF_CLR	0.18	0.15	0.07	0.66	0.64	0.27	0.42	0.42	0.18	4.25	4.45	4.12	0.03	**0.01**
RF_ILR	0.18	0.15	0.08	0.69	0.66	0.27	0.44	0.42	0.18	4.34	4.75	4.31	**0.02**	**0.01**
RF_ORI	**0.15**	0.12	0.07	0.53	0.54	0.25	0.4	0.41	**0.16**	2.95	3.47	3.06	0.03	**0.01**
SVM_ALR	0.17	0.12	**0.06**	**0.45**	**0.49**	0.25	0.35	0.43	0.17	3.27	3.74	2.82	0.03	**0.01**
SVM_CLR	0.16	0.12	**0.06**	0.49	0.5	0.27	**0.33**	**0.35**	0.18	3.05	3.35	3.47	0.03	**0.01**
SVM_ILR	0.16	0.12	**0.06**	0.51	0.51	0.25	0.34	0.36	0.18	3.07	3.38	3.18	0.03	**0.01**
SVM_ORI	**0.15**	**0.11**	**0.06**	0.51	**0.49**	0.25	0.34	**0.35**	0.17	**2.92**	**3.14**	2.95	0.03	**0.01**
XGB_ALR	0.17	0.14	0.07	0.67	0.57	**0.23**	0.48	0.41	**0.16**	4.07	3.97	3.6	0.03	**0.01**
XGB_CLR	0.19	0.15	0.07	0.73	0.65	0.25	0.44	0.44	**0.16**	4.9	5	3.82	0.03	**0.01**
XGB_ILR	0.17	0.13	0.08	0.72	0.69	0.26	0.46	0.48	0.19	4.52	4.86	4.44	0.03	**0.01**
XGB_ORI	0.16	0.12	**0.06**	0.6	0.61	0.24	0.41	0.46	**0.16**	3.4	4.03	2.9	0.03	**0.01**

Supplementary Material

Section S4 Prediction maps of silt and clay fractions

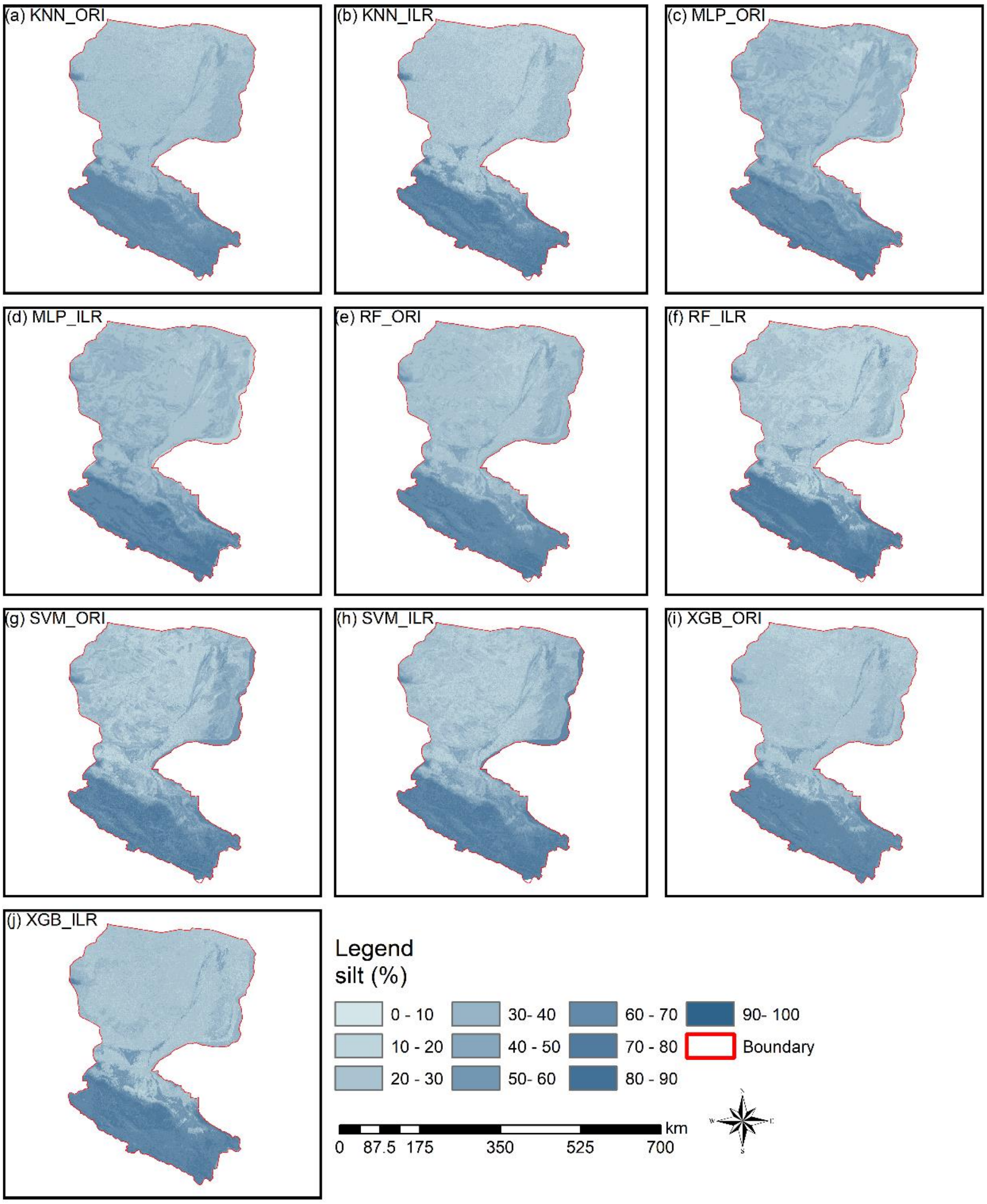

Figure S4.1. The prediction maps of silt fraction using five machine-learning models with ORI and ILR data.

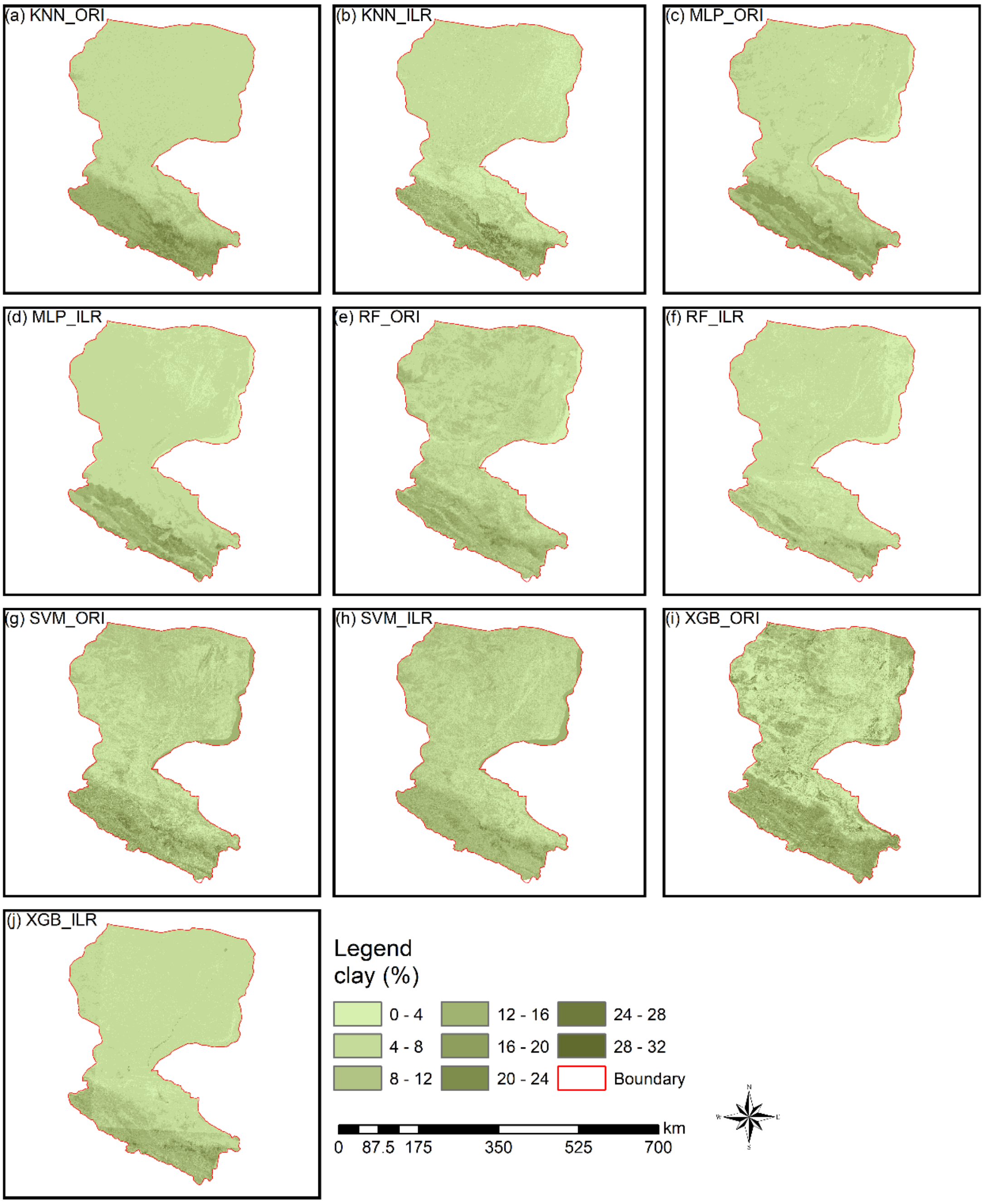

Figure S4.2. The prediction maps of clay fraction using five machine-learning models with ORI and ILR data.

Section S5 Indirect classification maps using ALR and CLR transformation methods

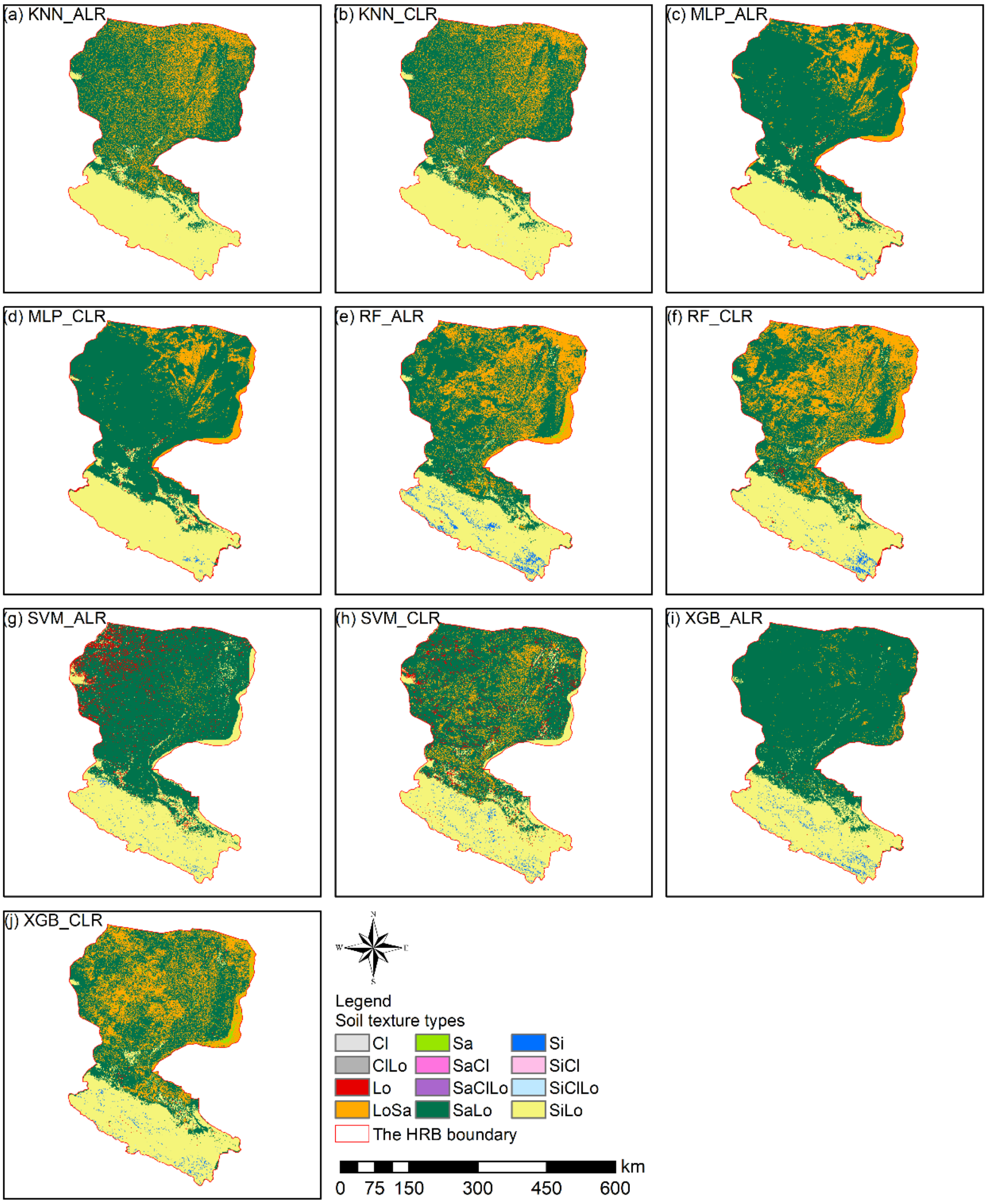

Figure S5.1. Soil texture classification prediction maps by soil PSFs interpolation (ALR and CLR log ratio transformation methods) of KNN, MLP, RF, SVM and XGB.