Relevance Assignment and Fusion of Multiple Learning Methods Applied to Remote Sensing Image Analysis

Peter Bajcsy[1], Wei-Wen Feng and Praveen Kumar
National Center for Supercomputing Application (NCSA),
University of Illinois at Urbana-Champaign,
Urbana, Illinois, USA
{ pbajcsy, fengww}@ncsa.uiuc.edu, kumar1@uiuc.edu

Abstract

With the advances in remote sensing, various machine learning techniques could be applied to study variable relationships. Although prediction models obtained using machine learning techniques has proven to be suitable for predictions, they do not explicitly provide means for determining input-output variable relevance.

We investigated the issue of relevance assignment for multiple machine learning models applied to remote sensing variables in the context of terrestrial hydrology. The relevance is defined as the influence of an input variable with respect to predicting the output result. We introduce a methodology for assigning relevance using various machine learning methods. The learning methods we use include Regression Tree, Support Vector Machine, and K-Nearest Neighbor. We derive the relevance computation scheme for each learning method and propose a method for fusing relevance assignment results from multiple learning techniques by averaging and voting mechanism. All methods are evaluated in terms of relevance accuracy estimation with synthetic and measured data.

1. Introduction

The problem of understanding relationships and dependencies among geographic variables (features) is of high interest to scientists during many data-driven, discovery-type analyses. Various machine learning methods have for data-driven analyses been developed to build prediction models that represent input-output variable relationships. However, prediction models obtained using machine learning techniques vary in their underlying model representations and frequently do not provide a clear interpretation of input-output variable relationships. Thus, the goal of data-driven modeling is not only accurate prediction but also interpretation of the input-output relationships.

In this paper, we address the problem of data-driven model interpretation to establish relevance of input variables with respect to predicted output variables. First, we introduce the previous work in Section 2 and formulate an interpretation of data-driven models by assigning relevance to input variables in Section 3. Relevance assignments are derived at the sample (local) or model (global) levels based on co-linearity of input variable basis vectors with the normal of regression hyper-planes formed over model-defined partitions of data samples. Second, we propose algorithms for combining relevance assignments obtained from multiple data-driven models in Section 4. Finally, we evaluate accuracy of relevance assignment using (a) three types of synthetic and one set of measured data, (b) three machine learning algorithms, such as Regression Tree (RT), Support Vector Machine (SVM), and K-Nearest Neighbors (KNN), and (c) two relevance assignment fusion methods as documented in Section 5 and summarize our results in Section 6. The novelty of our work lies in developing a methodology for relevance assignment over a set of machine learning models, proposing relevance fusion methods, and demonstrating the accuracy of multiple relevance assignment methods with multiple synthetic and experimental data sets.

2. Previous Work

Interpretation of data-driven models to establish input-output variable relationships is also part of variable (feature) subset selection. Feature subset

[1] POC: Peter Bajcsy, pbajcsy@ncsa.uiuc.edu

selection is one of the classical research areas in machine learning [3][5][14]. Its goal is to pre-select the most relevant features for learning concepts, and improve accuracy of learning [9]. In the work of Pudil et al [11], the authors use a sequential search through all possible combinations of feature subsets, and find an optimal feature subset that yields the best result. With a similar attempt, Perner and his coworkers [10] compare the influence of different feature selection methods in the learning performance for decision tree.

However, the exhaustive search over all possible combinations of feature subsets is not always computationally feasible for large number of features, for instance, in the case of hyperspectral band selection. To address the feasibility aspect, Bajcsy and Groves [8][15] proposed to use a combination of unsupervised and supervised machine learning techniques. Other approaches in spectral classification were proposed by S. De Backer et al. [12].

Another challenge of relevance assignment is related to the multitude of relevance definitions. For example, relevance of an input variable could be defined by traversing a regression tree model, and summing its weighted occurrence in the tree structure as proposed by White and Kumar [2]. In the survey by Blum et al [1], the authors give a conceptual definition of a relevant input variable as a variable that will affect the prediction results if it is modified. In our work, while adhering to the conceptual definition of Blum et al [1], we extend the definition of relevance assignment by numerically quantifying relative importance of input variables. Our relevance assignment is related to the work of Heiler et al [6], in which the authors used co-linearity of basis vectors with the normal of a separating hyper-plane obtained from Support Vector Machine (SVM) method as the metric for relevance assignment. We use the co-linearity property in our relevance assignments derived from a set of regression hyper-planes formed over model-defined partitions of data samples.

3. Relevance Assignments

To analyze input-output relationships in practice, one is presented with discrete data that could be described as a set of M examples (rows of a table) with N variables (columns of a table). Examples represent instances of multiple ground and remotely sensed measurements. Measurements are the variables (features) that might have relationships among themselves and our goal is to obtain better understanding of the relationships.

In this work, we start with a mathematical definition of variable relevance that is consistent with the conceptual understanding of input-output relationships. We define relevance of an input variable $v_i \in \vec{v} = (v_1, v_2, ..., v_N)$ as the partial derivative of an output (predicted) function $f(v_1, v_2, ..., v_N)$ with respect to the input variable v_i. Equation (1) shows a vector of relevance values for all input variables.

$$\vec{R} = \left(\frac{\partial f(v_1, v_2, ..., v_N)}{\partial v_1}; ...; \frac{\partial f(v_1, v_2, ..., v_N)}{\partial v_N} \right) \quad (1)$$

This definition assumes that input and output variables are continuous, and an output function f is C_1 continuous (first derivative exists). In order to follow this mathematical definition in practice, there arise challenges associated with (1) processing discrete samples, such as defining the neighborhood proximity of a sample in the manifold of input variables, (2) representing a data-driven model, such as deriving analytical form of a predicted function f, (3) scaling input and output variables with different dynamic ranges of measurements, (4) removing dependencies on algorithmic parameters of machine learning techniques, (5) understanding variability of relevance as a function of data quality parameters (e.g., sensor noise, cloud coverage during remote sensing), and (6) treating a mix of continuous and categorical variables, just to name a few.

Our approach to the above challenges is (a) to use a mathematically, well defined (analytically described) model, like the multi-variate regression, on sample partitions obtained using a machine learning technique, (b) to scale all variables to the same dynamic range of [0, 1], (c) to investigate dependencies on algorithmic parameters of machine learning techniques and data quality parameters, and (d) to propose the fusion of multi-method relevance results to increase our confidence in the relevance results.. Having analytical description of a function f allows us to derive relevance according to the definition. We have currently constrained our work to processing only continuous variables and foresee the inclusion of categorical variables in our future work.

Based on the above considerations, we define sample and model relevance assignments for a set of discrete samples with measurements of continuous input and output variables. *Example relevance* R_{ij} is the local relevance of each input variable v_i computed at the sample s_j. The computation is defined for three machine learning techniques in the next sub-sections.

Model relevance R_i is the global relevance of each input variable v_i over all examples in the entire data-driven model computed by summing all sample relevancies. To obtain comparable example and model relevancies from multiple data-driven models, we normalize the relevance values by the sum of all model relevancies over all input variables (see Equation (2)). The normalized relevance values are denoted with a tilde.

$$\tilde{R}_{ij} = \frac{R_{ij}}{\sum_i \sum_j R_{ij}}; \quad \tilde{R}_i = \frac{\sum_j R_{ij}}{\sum_i \sum_j R_{ij}} \tag{2}$$

In the next subsections, we introduce relevance assignment for Regression Tree (RT), Support Vector Machine (SVM), and K-Nearest Neighbors (KNN). The main reason for choosing these three methods comes from our requirement for remote sensing data analysis to process continuous input and output variables. Furthermore, the methods represent a set of machine learning techniques with different underpinning principles for data-driven modeling. The KNN method builds models using only close neighbors to a given sample. The SVM method builds models based on all input samples. As for the RT method, it builds its model by hierarchically subdividing all input samples and fitting a local regression model to samples in each divided cell (leaf). Thus, these methods represent a spectrum of supervised machine learning techniques that would be evaluated in terms of relevance assignment accuracy.

3.1. Regression Tree

The process of building a regression tree based model can be described by splitting input examples into sub-groups (denoted as cells or tree leaves) based on a sequence of criteria imposed on individual variables [14]. The splitting criteria, e.g., information gain or data variance, might depend on problem types, and become one of the algorithmic parameters. In this work, we choose variance as our splitting criterion. Once all examples are partitioned into leaves, a multi-variate linear regression model is used to predict output values for examples that fall into the leaf.

For any example e_j, *example relevance* R_{ij} of the variable v_i is computed from the linear regression model associated with the regression tree leaf. The regression model at each regression tree leaf approximates locally the prediction function f with a linear model written as:

$$f(\vec{v}) = \beta_0 + w_1 v_1 + w_2 v_2 + \ldots + w_N v_N \tag{3}$$

The example (local) relevance assignment R_{ij} is then computed as a dot product between the normal $\vec{w}$ of a regression model hyper-plane and the unit vector $\vec{u}_i$ of input variable v_i as described below:

$$R_{ij} = \left| \vec{w} \bullet \vec{u}_i \right| \tag{4}$$

where $\left| \vec{w} \bullet \vec{u}_i \right|$ denotes the absolute value of a dot product of vectors $\vec{w}$ and $\vec{u}_i$.

3.2. K-Nearest Neighbors

K-nearest neighbors is a machine learning method that predicts output values based on K closest examples to any chosen one measured in the space of input variables $\vec{v}$. Predicted values are formed as a weighted sum of those K nearest examples [13][14].

For any example e_j, *example relevance* R_{ij} of the variable v_i is computed from the linear regression model obtained from the K nearest neighbors to the example e_j. The linear regression model approximates locally the prediction function f. The example relevance assignment is performed according to Equations (3) and (4).

3.3. Support Vector Machine

Support vector machine (SVM) is a machine learning method that could build a model for separating examples (classification problem), or for fitting examples (prediction problem). We use SVM as a prediction technique to model input data. The SVM model could be linear or non-linear depending on a SVM kernel. The non-linear models are obtained by mapping input data to a higher dimensional feature space, which is conveniently achieved by kernel mappings [7]. Thus, the prediction function f could be obtained from mathematical descriptions of linear and non-linear kernels. In this paper, we focus only on a linear model in order to simplify the math.

For a SVM method with the linear kernel, the mathematical description of f becomes a hyper-plane as described in Equation (3). The difference between

RT, KNN and SVM methods is in the fact that SVM would use all examples to estimate the parameters of the hyper-plane and it would lead to only one hyper-plane for the entire set of examples. The example relevance assignment for SVM follows Equation (4). In the case of SVM, the example relevance and model relevance are identical.

4. Relevance Fusion

The goal of relevance fusion is to achieve more robust relevance assignment in the presence of noise, variable data quality (e.g., clouds during remote sensing), as well as to remove any bias introduced by a single machine learning method. The latter motivation is also supported by the no-free-lunch (NFL) theorem [16], which states that no single supervised method is superior over all problem domains; methods can only be superior for particular data sets.

In this paper, we propose two different schemes to combine the results of relevance assignment, such as average fusion and rank fusion. *Average fusion* is based on taking the numerical average of relevance values, and using it as the combined relevance value. *Rank fusion,* on the other hand, uses the voting scheme to combine the relative ranks of each input variable determined by each machine learning method.

4.1. Average Fusion

Average fusion is executed by taking the normalized relevance results from multiple machine learning methods, and computing the average of them.

Given $\tilde{R}_{ij}^{k}$ as the normalized example relevance of the *i*-th variable at the example e_j obtained from the *k*-th machine learning method, the example relevance assignment of $\tilde{R}_{ij}^{avg}$ established using average fusion for the example e_j is defined as:

$$\tilde{R}_{ij}^{avg} = \frac{1}{L}\sum_{k=1}^{L} \tilde{R}_{ij}^{k} \qquad (5)$$

where *L* is number of machine learning methods used.

The model relevance assignment of $\tilde{R}_{i}^{avg}$ using average fusion for an input variable v_i is the sum of all example relevancies as defined in Equation (2).

4.2. Rank Fusion

Rank fusion is executed in a similar manner as the average fusion. The difference lies in assigning a relevance rank to each variable v_i based on its relevance ranking by the *k*-th machine learning model. The absolute magnitude of relevance assignment is lost in rank fusion since the meaning of magnitude is converted to relative ranking during the process. The rank fusion approach is expected to be more robust than the average fusion approach when some of the machine learning methods create very incorrect models due to various reasons and skew the correct results of other machine learning methods.

The example relevance assignment using rank fusion is described as follows: For each example e_j and its normalized example relevance $\tilde{R}_{ij}^{k}$, we define the rank of each variable v_i as the index of a sorted list of relevancies from the smallest to the largest; $\underset{i}{rank}\{\tilde{R}_{ij}^{k}\} \subset \{1,2,...N\}$ .The rank fusion based relevance assignment for variable v_i is then computed as shown below:

$$\tilde{R}_{ij}^{rank} = \frac{2}{LN^2}\sum_{k=1}^{L}(N - rank\{\tilde{R}_{ij}^{k}\}) \qquad (6)$$

The model relevance assignment of $\tilde{R}_{i}^{rank}$ using average fusion for an input variable v_i is the sum of all example relevancies $\tilde{R}_{ij}^{rank}$ over all examples as defined in Equation (2).

5. Evaluation Setup

The evaluation of relevance assignments was performed using GeoLearn software that was developed by NCSA and CEE UIUC. GeoLearn allows a user to model input-output variable relationships from multi-variate NASA remote sensing images over a set of boundaries. The machine learning algorithms in the GeoLearn system leverage five software packages, such as, Im2Learn (remote sensing image processing), ArcGIS (georeferencing), D2K software (RT implementation), LibSVM [4] (SVM implementation), and KDTree [13] (KNN implementation).

KNN and SVM methods are sensitive to the scale of input variables, and will favor variables with a wider scale of values. In order to avoid this type of a bias, the dynamic range of all variables is always normalized to

the range between [0, 1] according to the formula below:

$$NormalizedValue = \frac{Value - MinValue}{MaxValue - MinValue} \quad (7)$$

Evaluations were performed with both synthetic and measured data. Synthetic data allow us to simulate three categories of input-output relationships with known ground truth to understand relevance assignment accuracy and relevance dependencies. Measured data was used to demonstrate the application of relevance assignment to studying vegetation changes and the results were verified based on our limited understanding of the phenomena.

The next sub-sections describe synthetic data simulations, model building setup and evaluation metrics to assess relevance assignment accuracy.

5.1. Synthetic Data Simulations

Three sets of input-output relationships were simulated to represent (1) linear additive, (2) non-linear additive and (3) non-linear multiplicative categories of relationships. To introduce irrelevant input variables into the problem, we simulated output using only two input variables v_1, v_2 (the relevant variables) while modeling relationships with four variables, where the additional two input variables v_3, v_4 have values drawn from a uniform distribution of [0,1] (the irrelevant variables). The specific analytical forms for generating the three data sets are provided in Equations (8) -linear additive, (9) – non-linear additive and (10) – non-linear multiplicative.

$$f(v_1, v_2, v_3, v_4) = 4v_1 + v_2 \quad (8)$$

$$f(v_1, v_2, v_3, v_4) = \sin \pi v_1 + \cos \frac{\pi}{2} v_2 \quad (9)$$

$$f(v_1, v_2, v_3, v_4) = v_1 v_2^2 \quad (10)$$

In addition to simulating multiple input-output relationships and relevant-irrelevant variables, we added noise to generated output values to test the noise robustness of relevance assignments. Noise is simulated to be an additive variable following 1D Gaussian distribution with zero mean μ and standard deviation σ; $N(\mu = 0, \sigma)$. The standard deviation was parameterized as $\sigma = \alpha d$, where α is the percentage of the dynamic range d of an output variable.

In our experiments, we used $\alpha = 0.1$ and $\alpha = 0.3$ to generate the total of nine synthetic data sets (3 without noise, 3 with additive noise $\alpha = 0.1$, and 3 with additive noise $\alpha = 0.3$).

5.2. Model Building Setup

Model building setup is concerned with optimization of algorithmic parameters and cross validation. First, we set the algorithmic parameters to the following values: (1) RT - variance error as a criterion for splitting, minimum number of examples per leaf to eight, maximum tree depth to 12; (2) KNN – $K = N + 3$, where N is the dimension of all input variables. The reason for setting K slightly larger than the input variable dimensions is to meet the least-square fitting requirements for estimating a hyper-plane from K examples; (3) SVM - linear kernel, cost factor C=1.0, and termination criteria Epsilon = 0.001. The optimization of KNN's parameter "K" and RT's parameter "maximum tree depth" was investigated experimentally.

Second, we omitted cross validation of models in our experiments and rather computed input variable relevance based on all available examples. We will investigate in the future the accuracy of input variable relevance assignment from examples selected by cross validation or all available examples.

5.3. Evaluation Metrics

To evaluate the accuracy of input variable relevance assignment using multiple machine learning methods, we introduced two metrics, such as "percentage of correctness" and "error distance." The evaluations are conducted only for the synthetic data against the ground truth values of normalized example relevance $\tilde{R}_{ij}^{GT}$ and normalized model relevance $\tilde{R}_i^{GT}$. The ground truth values are obtained by computing partial derivatives of Equations (8), (9) and (10) according to Equation (1).

The "percentage of correctness" metric is defined in Equation (11) as:

$$PC = \frac{\sum_{j=1}^{M} \delta_j}{M} \times 100\% \quad (11)$$

where δ_j is 1 if $\max_i \tilde{R}_{ij}^{GT} = \max_i \tilde{R}_{ij}$, and is 0 otherwise. The error distance metric is defined in Equation (12) as the Euclidean distance between the true model relevance derived from partial derivative and the relevance estimation from our methods. This

Table 1 : Relevance table obtained from synthetic data without noise by RT, KNN and SVM methods.

Model	Var	Linear		Non-Linear Add.		Non-Linear Multi.	
		Relevance	Correct % / Distance Error	Relevance	Correct % / Distance Error	Relevance	Correct % / Distance Error
Regression Tree	v1	0.8	100%	0.667415	95.5%	0.423898	91.83%
	v2	0.2		0.330931		0.575511	
	v3	3.80E-17	4.09373E-29	8.54E-04	2.79E-06	2.73E-04	0.001057
	v4	3.72E-17		8.01E-04		3.17E-04	
K-Nearest Neighbors	v1	0.8	100%	0.616814	95.07%	0.35364	96.94%
	v2	0.2		0.326967		0.568512	
	v3	7.48E-14	2.24784E-25	0.027992	0.004263	0.039011	0.00621
	v4	3.53E-13		0.028227		0.038838	
Support Vector Machine (Linear)	v1	0.748753	100%	0.040722	20.69%	0.391619	75%
	v2	0.249007		0.95396		0.598363	
	v3	6.29E-04	0.005030969	0.005292	0.781345	0.001454	0.000168
	v4	0.00161		2.69E-05		0.008564	

metric does not apply to the relevance results obtained using rank fusion since the results are categorical.

$$ErrorDist. = \sum_{i=1}^{N} (\tilde{R}_i^{GT} - \tilde{R}_i)^2 \qquad (12)$$

6. Experiment Results

In this section, we present evaluations with synthetic and measured data in two forms. First, we report a *relevance image* that shows the color of an input variable with maximum relevance value at each pixel location. The color coding schema maps red, green, blue and yellow colors to (v_1, v_2, v_3, v_4) . Second, we provide a *relevance table* with model relevance value for each input variable.

6.1. Synthetic Data

The relevance assignment results using RT, KNN and SVM methods from synthetic data without noise are summarized in Figure 1 and Table 1. The results obtained using the fusion methods for the same data are summarized in Figure 2 and Table 2. We also evaluated the methods with synthetic data with noise and the results were omitted for brevity.

6.2. Measured Data

We processed measured remotely sensed data from NASA acquired in 2003, at spatial resolution 1000m per pixel and at the location (latitude x longitude) = ([35.34, 36.35] x [-91.54, -93.32]. We modeled output Fpar variable (fraction of Photosynthetic active radiation) as a function of input variables consisting of LST (Land Surface Temperature), LAI (Leaf Area Index), Latitude, and Longitude. For this geo-spatial location, we anticipated LAI to be more relevant to Fpar than LST, and both Latitude and Longitude to be almost irrelevant. The relevance results are summarized in Figure 3 and Table 3

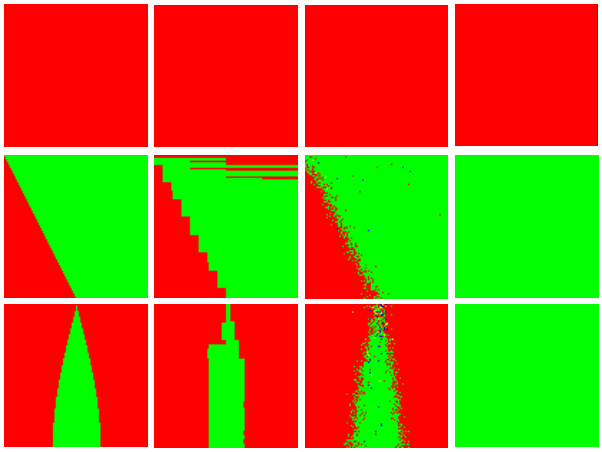

Figure 1. Relevance images obtained from synthetic data without noise by RT, KNN and SVM methods. Rows correspond to categories of relationships (top – linear additive, middle - non-linear multiplicative, bottom - non-linear additive). Columns correspond to methods (left - ground truth, second left – RT, second right KNN and right – SVM).

7. Discussion

The results presented in the previous section are discussed by comparing individual machine learning methods and fusion methods.

Table 2 : Relevance table obtained from synthetic data without noise by average and rank fusion methods.							
Model	Var	Linear		Non-Linear Add.		Non-Linear Multi.	
		Relevance	Correct % Distance Error	Relevance	Correct % Distance Error	Relevance	Correct % Distance Error
Fusion Average	v1	0.782918	100%	0.44165	68.67%	0.389719	94.72%
	v2	0.216336		0.537286		0.580795	
	v3	2.10E-04	0.000558	0.011379	0.093948	0.013579	0.000893
	v4	5.37E-04		0.009685		0.015906	
Fusion Rank	v1	0.4	100%	0.3	97.58%	0.3	97.62%
	v2	0.3		0.4		0.4	
	v3	0.1	NA	0.133333	NA	0.166667	NA
	v4	0.2		0.166667		0.133333	

7.1. Comparison of Individual Methods

From the experimental results, we observe that RT is a reliable hybrid method that usually gives accurate relevance estimation. KNN is flexible under different data type and always gives good relevance estimation in terms of correctness. However, it is very sensitive to noise, and we observe significant performance drops when noise presents. SVM usually yields more robust result under noise, but is restricted by its expressivity since we use only linear kernel here. There is no single best method for every dataset. The correctness of relevance assignment strongly depends on the type of data and the learning method we use. The fusion methods are proposed based on this motivation.

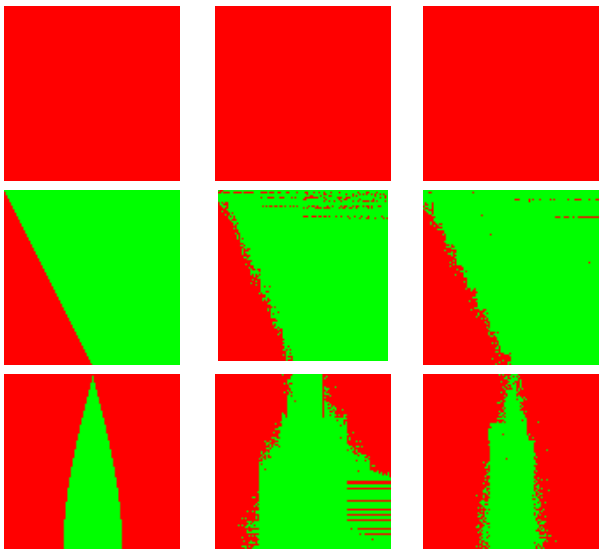

Figure 2. Relevance images obtained from synthetic data without noise by using average and rank fusion methods. Rows correspond to categories of relationships (top – linear additive, middle - non-linear multiplicative, bottom - non-linear additive). Columns correspond to methods (left - ground truth, middle – average fusion, and right – rank fusion).

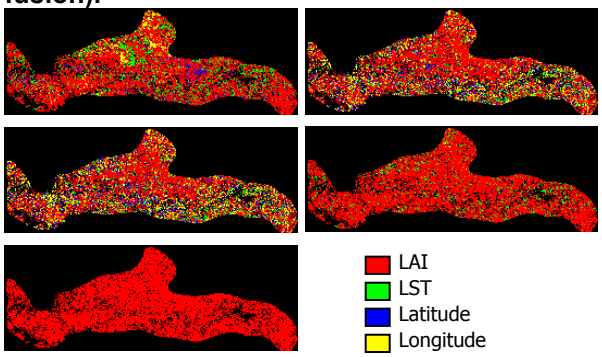

Figure 3. Relevance images for NASA data obtained using RT (left top), KNN (left middle), SVM (left bottom), average fusion (right top) and rank fusion (right middle). The most relevant input variable at each pixel is encoded according to the color scheme in the legend (right bottom). The majority of pixels is labeled red (LAI).

7.2. Comparison of Fusion Methods

Fusion methods usually outperform any single method in terms of relevance assignment correctness (% Correct) for synthetic datasets without noise. Under this setting, the difference between average fusion and rank fusion is not obvious.

Although we have not included the results for synthetic data with noise for brevity reasons, we would comment on the results obtained. For synthetic datasets with noise, the correctness of relevance assignment for fusion methods is still more stable than for a single method (RT or KNN or SVM). However, fusion methods do not give the optimal estimation comparing to the best single method.

8. Conclusion and Future Work

In this paper, we proposed a framework for computing input variable relevance with respect to a

predicted output variable from multiple machine learning methods. Following the conceptual definition of relevance in the literature, we defined partial derivatives of input-output dependencies as our relevance assignment approach. The estimation of two types of relevancies, such as example and model relevancies, were implemented for regression tree, K-nearest neighbors, and support vector machine methods. Additional fusion schemes for combining the relevance results from multiple methods were evaluated together with single methods by using synthetic and measured data and two metrics. Based on three categories of synthetic input-output simulations, including linear additive, non-linear additive and non-linear multiplicative relationships, without or with noise added, we concluded that the relevance assignment using fusion approaches demonstrate more robust performance than the assignment using a single machine learning method.

Table 3 : Relevance assignment results for remote sensing data set from NASA.

Model	Var	Relevance
Regression Tree	LAI	0.996014
	LST	0.003058
	Latitude	1.07E-04
	Longitude	8.20E-04
K-Nearest Neighbors	LAI	0.449047
	LST	0.178427
	Latitude	0.17511
	Longitude	0.197416
Support Vector Machine (Linear)	LAI	0.885942
	LST	0.073454
	Latitude	0.040576
	Longitude	2.82E-05
Average Fusion	LAI	0.777001
	LST	0.08498
	Latitude	0.071931
	Longitude	0.066088
Rank Fusion	LAI	0.4
	LST	0.266667
	Latitude	0.133333
	Longitude	0.2

In the future, we would like to extend the fusion methods to include the results from other learning methods and to understand the dependencies of relevance assignment on model building setups.

10. References

[1] Blum A.L. and P. Langley, 1997. "Selection of Relevant Features and Examples in Machine Learning." *Artificial Intelligence 97*, 1997 Elsevier Science.

[2] White A. and P. Kumar, 2005. "Dominant Influences on Vegetation Greenness Over the Blue Ridge Ecoregion" *Submitted to Ecosystems.*

[3] Jain A. and D. Zongker, 1997. "Feature selection: evaluation, application, and small sample performance." *IEEE Trans. Pattern Anal. Machine Intell.,* 19:153–189, 1997.

[4] Lin C.J., 2006. "LIBSVM -- A Library for Support Vector Machines." **http://www.csie.ntu.edu.tw/~cjlin/libsvm/**.

[5] Kudo M. and J. Sklansky, 2000. "Comparison of algorithms that select features for pattern classifiers." *Pattern Recognition*, 33:25-41, 2000.

[6] Heiler M., D. Cremers, and C. Schnorr, 2001. "Efficient Feature Subset Selection for Support Vector Machines." *Technical Report 21/2001* Dept. of Math and Computer Science, University of Mannheim.

[7] Cristianini N. and J. S. Taylor, 2000. "An Introduction to Support Vector Machines and Other Kernel-based Learning Methods," Cambridge University Press, March 2000.

[8] Bajcsy P. and P. Groves, 2004. "Methodology For Hyperspectral Band Selection," Photogrammetric Engineering and Remote Sensing journal, Vol. 70, Number 7, July 2004, pp. 793-802.

[9] Perner P. and C. Apte, 2000. "Empirical Evaluation of Feature Subset Selection Based on a Real World Data Set," *Principles of Data Mining and Knowledge Discovery*, Springer Verlag 2000, p. 575-580.

[10] Perner P., 2001. "Improving the Accuracy of Decision Tree Induction by Feature Pre-Selection," *Applied Artificial Intelligence 2001*, vol. 15, No. 8, p. 747-760.

[11] Pudil P., J. Navovicova, and J.Kittler, 1994. "Floating search methods in feature selection," *Pattern Recognition Letters,* 15(1994), 1119-1125.

[12] De Backer S., P. Kempeneers, W. Debruyn and P. Scheunders, 2005. "A Band Selection Technique for Spectral Classification," *IEEE Geoscience and Remote Sensing Letters*, Vol. 2, Nr. 3, p. 319-323, July (2005).

[13] Levy S. D., 2006, "A Java class for KD-tree search." http://www.cs.wlu.edu/~levy/

[14] Han, J., and Kamber, M., 2001. *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, San Francisco, CA.

[15] Groves P. and P. Bajcsy, 2003. "Methodology for Hyperspectral Band and Classification Model Selection," IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data, Washington DC, October 27, 2003.

[16] Duda, R., P. Hart and D. Stork, 2001. *Pattern Classification,* Second Edition, Wiley-Interscience.