were added as noise for prediction purposes.), i.e., if the signs of first feature value and second feature value are the same, $y = 1$, otherwise $y = -1$. This is well known as a case in which linear models do not work. For example, L_1-regularized logistic regression produced nearly random outputs where the error rate was 0.421.

We generated one partition for each feature except for the first feature. Each partition became active if the corresponding feature value was greater than 0.0. Therefore, the number of candidate partitions was 19. We used the logistic regression function for loss functions. Hyper-parameters[3] were set as $\lambda_0 = 0.01$ and $\lambda_P = 0.001$. The algorithm was run in $1,000$ iterations.

Figure 2 illustrates results produced by the global and local residual model. The left-hand figure illustrates a learned effective partition (red line) to which the weight vector $a_1 = (10.96, 0.0, \cdots)$ was assigned. This weight a_1 was only applied to the region above the red line. By combining a_1 and the global weight a_0, we obtained the piece-wise linear representation shown in the right-hand figure. While it is yet difficult for existing piece-wise linear methods to capture global structures[4], our convex formulation makes it possible for the global and local residual model to easily capture the global XOR structures.

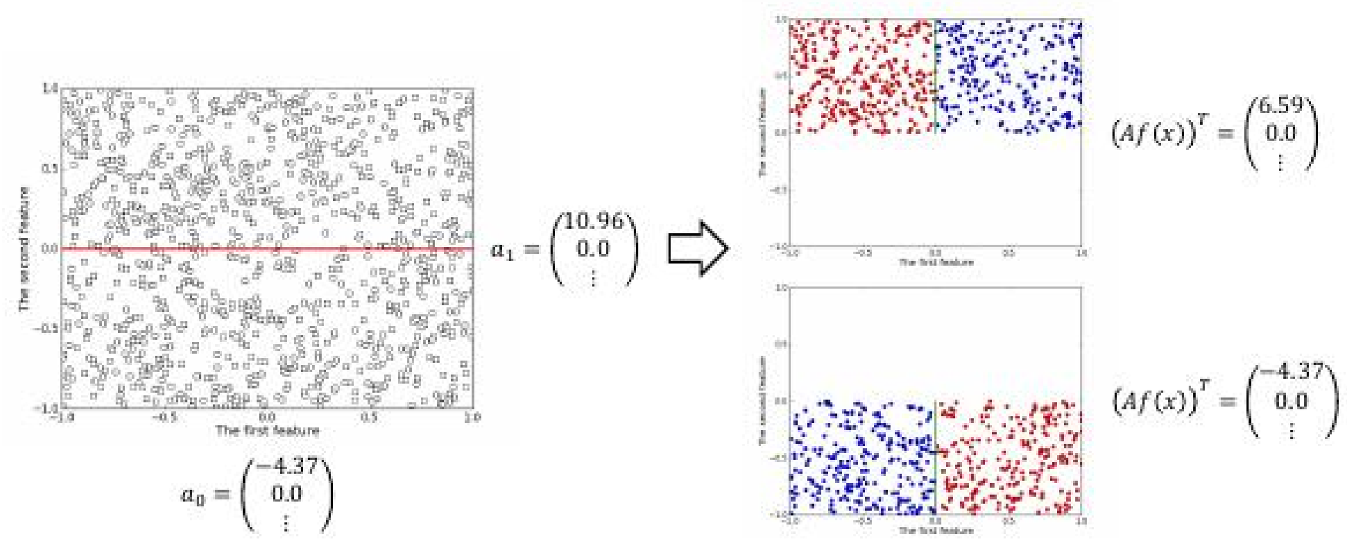

Figure 2: How the global and local residual model classifies XOR data. Red line indicates effective partition; green lines indicate local predictors; red circles indicate samples with $y = -1$; blue circles indicate samples with $y = 1$: This model classified XOR data precisely.

5.2 Comparisons using Benchmark Datasets

We next used benchmark datasets to compare our models with other state-of-the-art region-specific ones. In these experiments, we simply generated partition candidates (activeness functions) as follows. For continuous value features, we calculated all 5-quantiles for each feature and generated partitions at each quantile point. Partitions became active if a feature value was greater than the corresponding quantile value. For binary categorical features, we generated two partitions in which one became active when the feature was 1 (yes) and the other became active only when the feature value was 0 (no).

Table 2: Classification and regression datasets. N is the size of data. D is the number of dimensions. P is the number of partitions. CL/RG denotes the type of dataset (CL: Classification/RG: Regression).

	N	D	P	CL/RG
skin	245,057	[illegible]	12	CL
winequality	6,497	11	44	CL
census_income	45,222	[illegible]	[illegible]	CL
twitter	140,707	11	44	CL
a1a	1,605	113	452	CL
breast_cancer	683	10	40	CL
internet_ad	2,359	[illegible]	[illegible]	CL
energy_heat	768	[illegible]	32	RG
energy_cool	768	[illegible]	32	RG
abalone	[illegible]	[illegible]	[illegible]	RG
kinematics	8,192	[illegible]	32	RG
puma8NH	8,192	[illegible]	32	RG
bank8FM	8,192	8	32	RG
communities	1,994	101	404	RG

We utilized several standard benchmark datasets from UCI datasets (skin, winequality, census_income, twitter, internet_ad, energy_heat, energy_cool, communities), libsvm datasets (a1a, breast_cancer), and LIACC datasets (abalone, kinematics, puma8NH, bank8FM). Table 2 summarizes specifications for each dataset.

5.2.1 Classification

For classification, we compared the global and local residual model (Global/Local) with L_1 logistic regression (Linear), LSL-SP with linear discrimination analysis[5], LDKL supported by L_2-regularized hinge loss[6], FaLK-SVM with linear kernels[7], and C-SVM with RBF kernel[8]. Note that C-SVM is neither a region-specific nor locally linear classification model; it is, rather, non-linear. We compared it with ours as a reference with respect to a common non-linear classification model.

[3] We conducted several experiments on other hyper-parameter settings and confirmed that variations in hyper-parameter settings did not significantly affect results.

[4] For example, a decision tree cannot be used to find a "true" XOR structure since marginal distributions on the first and second features cannot discriminate between positive and negative classes.

[5] The source code is provided by the author of [3].

[6] https://research.microsoft.com/en-us/um/people/manik/code/LDKL/download.html

[7] http://disi.unitn.it/~segata/FaLKM-lib/

[8] We used a libsvm package. http://www.csie.ntu.edu.tw/~cjlin/libsvm/