Weakly Supervised Regression on Uncertain Datasets

Alexander Litvinenko[1],
joint work with
Vladimir Berikov[2], Roman Kozinets[2],
Kirill Kalmutskiy and Layla Cherikbaeva[4]

[1]RWTH Aachen, Germany, [2]Sobolev Institute of Mathematics, [3]
Novosibirsk State University, Russia [4]Al-Farabi Kazakh National
University, Kazakhstan

Plan

After short reviewing of ML tasks, we

- introduce weakly-supervised regression problem
- propose a method which combines graph Laplacian regularization and cluster ensemble (collective decision making) methodologies
- solve an auxiliary minimisation problem
- apply our method for solving two clustering problems.

Machine learning (ML) problems

ML problems can be classified as

- fully supervised,
- unsupervised,
- semi-supervised,
- weakly-supervised.

Motivation: Recognition of acute stroke from CT scans

Manual annotation of a large number of computed tomography (CT) digital images

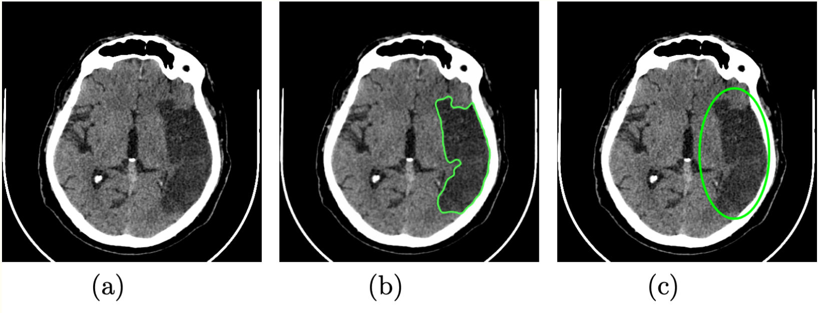


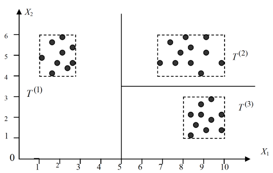
Figure: Example of CT image: (a) initial image, (b) stroke area annotated by a radiologist, (c) inaccurate mask.

V. Berikov, A. Litvinenko, I. Pestunov, and Y.Sinyavskiy, On a Weakly Supervised Classification Problem, AIST 2021 Proceedings, accepted to Springer Nature

Machine learning (ML) problems

Consider a dataset $\mathbf{X} = \{x_1, \ldots, x_n\}$, where $x_i \in \mathbb{R}^d$ is a feature vector, d the dimensionality of feature space, and n the sample size.

Fully supervised learning assumes we are given a set $Y = \{y_1, \ldots, y_n\}$, $y_i \in D_Y$, of target feature labels for each data point.



GOAL: To find a decision function $y = f(x)$ (classifier, regression model) for predicting target feature values for any new data point $x \in \mathbb{R}^d$ from the same statistical population.

The function should be optimal in some sense, e.g., give minimal value to the expected losses.

Unsupervised learning setting

In an unsupervised learning problem, the target feature is not specified.

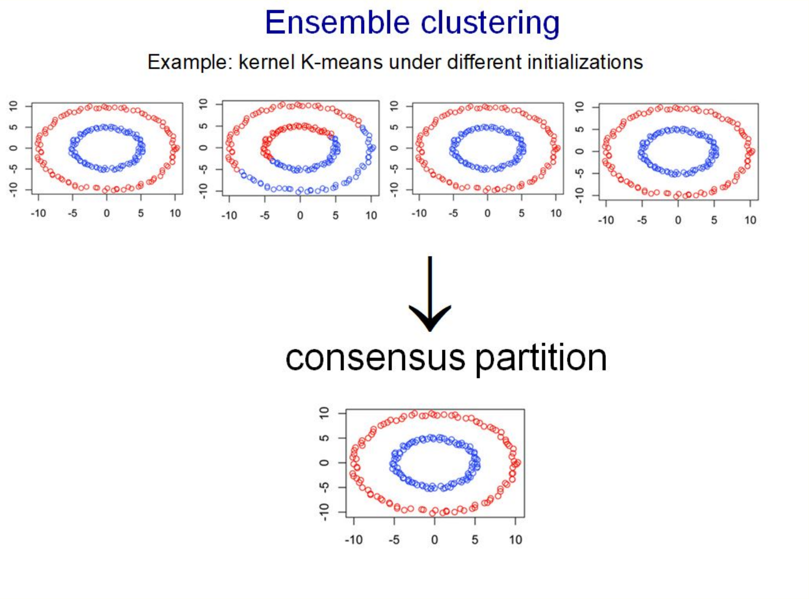
It is necessary to find a meaningful representation of data, i.e., find a partition $P = \{C_1, \ldots, C_K\}$ of $\mathbf{X}$ on a relatively small number K of homogeneous clusters describing the structure of data.

The desired number of clusters is either a predefined parameter or should be found in the best way.

The obtained cluster partition can be uncertain due to:

1. a lack of knowledge about data structure,
2. uncertaity in setting optional parameters of the learning algorithm,
3. dependence on random initializations.

Construction of the consensus partition



Ensemble clustering

Example: kernel K-means under different initializations

consensus partition

Semi-supervised learning problems

The target feature labels are known only for a part of the data set $\mathbf{X}_1 \subset \mathbf{X}$.
We assume that $\mathbf{X}_1 = \{x_1, \ldots, x_{n_1}\}$, and the unlabeled part is $\mathbf{X}_0 = \{x_{n_1+1}, \ldots, x_n\}$.
The set of labels for points from $\mathbf{X}_1$ is denoted by $\mathbf{Y}_1 = \{y_1, \ldots, y_{n_1}\}$.

GOAL: Predict target features $\mathbf{Y}_0 = (y_{n_1+1}, \ldots, y_n)$ either for given unlabeled data $\mathbf{X}_0$ (i.e., perform *transductive learning*) or for arbitrary new observations from the same statistical population (*inductive learning*).

Note: To improve prediction accuracy, the information contained in both labeled and unlabeled datasets is used.

Weakly-supervised learning problems

Def.: For some data points the labels are known, for some unknown, and for others uncertain
(due to lack of resources, random distortions in labels identification, etc).

To model uncertainty in the label identification, we suppose that for each i-th point, $i = 1, \ldots, n_1$, the value y_i of the target feature is a realization of a random variable Y_i with cumulative distribution function (cdf) $F_i(y)$ defined on D_Y.
We suppose that $F_i(y)$ belongs to a given distribution family.

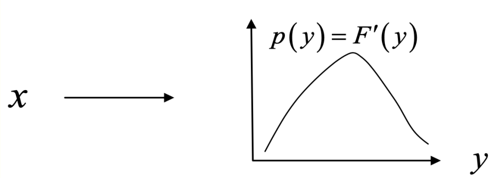
Weakly-supervised learning problems

Further assume:

$$Y_i \sim \mathcal{N}(a_i, \sigma_i), \tag{1}$$

where a_i, σ_i are the mean and the st. dev.
Assume $a_i = y_i$ and $\sigma_i = s_i$ are known for each (weakly) labeled observation, $i = 1, \ldots, n_1$.
For strictly determined observation y_i, we postulate a normal uncertainty model with $a_i = y_i$ and small standard deviation $\sigma_i \approx 0$.

We aim at finding a weak labeling of points from $\mathbf{X}_0$, i.e., determining cfd $F_i(y)$ for $i = n_1 + 1, \ldots, n$ following an objective criterion.



Assumptions:

Semi- and weakly- supervised learning assume:

1. cluster assumption (points from the same cluster often have the same labels or labels close to each other)
2. manifold assumption (points with similar labels belong to a smooth manifold).

Optimization problem

Let $F = \{F_1, \ldots, F_n\}$ denote the set of cdfs for n data points; each cdf F_i is represented by a pair of parameters (a_i, σ_i). We solve

$$\text{find } F^* = \arg\min_F J(F), \text{ where}$$

$$J(F) = \sum_{x_i \in X_1} \mathcal{D}(\mathcal{N}_i, F_i) + \gamma \sum_{x_i, x_j \in \mathbf{X}} \mathcal{D}(F_i, F_j) W_{ij}. \tag{2}$$

Here $\mathcal{D}$ is a statistical distance between two distributions (such as the Wasserstein distance, KLD,...)[1], W_{ij} describes the degree of similarity between two points.

1st term: reduces the dissimilarity on labeled data;
2nd: (smoothing) if two points x_i, x_j are similar, their labeling distribution should not be very different.

[1] Litvinenko, A, Marzouk, Y, Matthies, HG, Scavino, M, Spantini, A. Computing f-divergences and distances of high-dimensional probability density functions. Numer Linear Algebra Appl. 2022;e2467. https://doi.org/10.1002/nla.2467

Objective functional

Use the Wasserstein distance w_p between distributions P and Q over a set D_Y as a measure of their dissimilarity:

$$w_p(P,Q) := \left(\inf_{\gamma\in\Gamma(P,Q)} \int_{D_Y \times D_Y} \rho(y_1,y_2)^p \, \mathrm{d}\gamma(y_1,y_2) \right)^{1/p},$$

where $\Gamma(P,Q)$ is a set of all probability distributions on $D_Y \times D_Y$ with marginal distributions P and Q, ρ a distance metric, and $p \geq 1$.

For normal distributions $P_i = \mathcal{N}(a_i, \sigma_i)$, $Q_j = \mathcal{N}(a_j, \sigma_j)$ and the Euclidean metric, the w_2 distance is equal to

$$w_2(P_i, Q_j) = (a_i - a_j)^2 + (\sigma_i - \sigma_j)^2.$$

Objective functional

Add an L_2 regularizer:

$$J(a, \sigma) = \sum_{i=1}^{n_1} \left((y_i - a_i)^2 + (s_i - \sigma_i)^2 \right) +$$

$$+ \gamma \sum_{i,j=1}^{n} \left((a_i - a_j)^2 + (\sigma_i - \sigma_j)^2 \right) W_{ij} + \beta(\|a\|^2 + \|\sigma\|^2), \qquad (3)$$

where $\beta > 0$ is a regularization parameter, $a = (a_1, \ldots, a_n)^\top$, $\sigma = (\sigma_1, \ldots, \sigma_n)^\top$,

Optimal solution

To find the optimal solution, we differentiate (3) and get:

$$\frac{\partial J}{\partial a_i} = 2(a_i - y_i) + 4\gamma \sum_{j=1}^{n} (a_i - a_j) W_{ij} + 2\beta a_i = 0, \quad i = 1, \ldots, n_1, \tag{4}$$

$$\frac{\partial J}{\partial a_i} = 4\gamma \sum_{j=1}^{n} (a_i - a_j) W_{ij} + 2\beta a_i = 0, \quad i = n_1 + 1, \ldots, n. \tag{5}$$

By $L := D - W$ we denote the standard Graph Laplacian, where D is the diagonal matrix with elements $D_{ii} = \sum_j W_{ij}$.

Optimal solution

Denote $Y_{1,0} = (y_1, \dots, y_{n_1}, \underbrace{0, \dots, 0}_{n-n_1})^\top$ and let B be a diagonal matrix with elements

$$B_{ii} = \begin{cases} \beta+1, \ i=1,\dots,n_1 \\ \beta, \ i=n_1+1,\dots,n. \end{cases}$$

Combining (4), (5) and using vector-matrix notation, we finally get:

$$(B + 2\gamma L)a = Y_{1,0},$$

thus the optimal solution is

$$a^* = (B + 2\gamma L)^{-1} Y_{1,0}. \tag{6}$$

Similarly, one can obtain the optimal value of σ:

$$\sigma^* = (B + 2\gamma L)^{-1} S_{1,0}, \tag{7}$$

where $S_{1,0} = (s_1, \dots, s_{n_1}, \underbrace{0, \dots, 0}_{n-n_1})^\top$.

Speeding up linear algebra operations

Let the similarity matrix be presented in the a low-rank form

$$W = AA^\top, \tag{8}$$

where matrix $A \in \mathbb{R}^{n \times m}$, $m \ll n$. Further, we have

$$B + 2\gamma L = B + 2\gamma D - 2\gamma AA^\top = G - 2\gamma AA^\top, \tag{9}$$

where $G := B + 2\gamma D$.
The following Woodbury matrix identity is well-known in linear algebra:

$$(S + UV)^{-1} = S^{-1} - S^{-1}U(I + VS^{-1}U)^{-1}VS^{-1}, \tag{10}$$

where $S \in \mathbb{R}^{n \times n}$ is an invertible matrix, $U \in \mathbb{R}^{n \times m}$ and $V \in \mathbb{R}^{m \times n}$.

Speeding up linear algebra operations

Let $S = G$, $U = -2\gamma A$ and $V = A^\top$. One can see that

$$G^{-1} = \operatorname{diag}\left(1/(B_{11} + 2\gamma D_{11}), \ldots, 1/(B_{nn} + 2\gamma D_{nn})\right). \qquad (11)$$

From (6), (9), (10) and (11) we obtain:

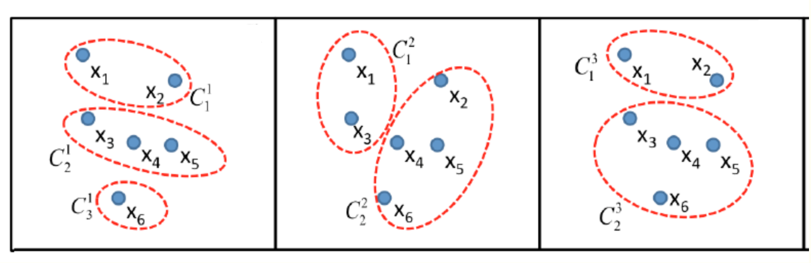
$$a^* = \left(G^{-1} + 2\gamma G^{-1} A (I - 2\gamma A^\top G^{-1} A)^{-1} A^\top G^{-1}\right) Y_{1,0}. \qquad (12)$$

Similarly, from (7), (9), (10) and (11) we have:

$$\sigma^* = \left(G^{-1} + 2\gamma G^{-1} A (I - 2\gamma A^\top G^{-1} A)^{-1} A^\top G^{-1}\right) S_{1,0}. \qquad (13)$$

The comput. complexity of (12), (13) is reduced and can be estimated as $O(nm + m^3)$ instead of $O(n^3)$.

An example of co-association matrix of cluster ensemble

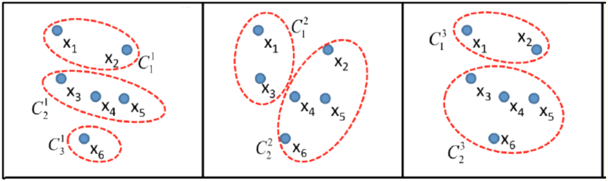


$$W = \frac{1}{3}\begin{bmatrix} 3 & 2 & 1 & 0 & 0 & 0 \\ 2 & 3 & 0 & 1 & 1 & 1 \\ 1 & 0 & 3 & 2 & 2 & 1 \\ 0 & 1 & 2 & 3 & 3 & 2 \\ 0 & 1 & 2 & 3 & 3 & 2 \\ 0 & 1 & 1 & 2 & 2 & 3 \end{bmatrix}$$

$W_{1,2}$ shows how often a pair $\{x_1, x_2\}$ belongs to the same cluster.

An example of co-association matrix of cluster ensemble

$W = \frac{1}{L} A \cdot A^\top$, $A = [A_1, A_2, ..., A_L]$ a block matrix,



$$A = \begin{array}{ccc|cc|cc} C_1^1 & C_2^1 & C_3^1 & C_1^2 & C_2^2 & C_1^3 & C_2^3 \\ 0.6 & 0.3 & 0.1 & 0.7 & 0.3 & 0.6 & 0.4 \\ 0.6 & 0.3 & 0.1 & 0.4 & 0.6 & 0.6 & 0.4 \\ 0.3 & 0.5 & 0.2 & 0.7 & 0.3 & 0.5 & 0.5 \\ 0.2 & 0.7 & 0.1 & 0.5 & 0.5 & 0.3 & 0.7 \\ 0.3 & 0.5 & 0.2 & 0.7 & 0.3 & 0.4 & 0.6 \\ 0.1 & 0.2 & 0.7 & 0.2 & 0.8 & 0.1 & 0.9 \end{array}$$

Co-association matrix of cluster ensemble

Use a co-association matrix of cluster ensemble as a similarity matrix in (3).
Consider partitions $\{P_\ell\}_{\ell=1}^r$, where $P_\ell = \{C_{\ell,1}, \ldots, C_{\ell,K_\ell}\}$, $C_{\ell,k} \subset \mathbf{X}$, $C_{\ell,k} \bigcap C_{\ell,k'} = \varnothing$ and K_ℓ is the number of clusters in ℓ-th partition.
For each partition P_ℓ determine matrix $H_\ell = (h_\ell(i,j))_{i,j=1}^n$ with elements indicating whether a pair x_i, x_j belong to the same cluster in ℓ-th variant or not.
We have

$$h_\ell(i,j) = \mathbb{I}[c_\ell(x_i) = c_\ell(x_j)],$$

and $c_\ell(x)$ is the cluster label assigned to x.
The weighted averaged co-association matrix is

$$H = \sum_{\ell=1}^{r} \omega_\ell H_\ell, \tag{14}$$

where $\omega_1, \ldots, \omega_r$ are weights , $\omega_\ell \geq 0$, $\sum \omega_\ell = 1$.

Graph Laplacian

Graph Laplacian matrix for H can be written in the form:

$$L = D' - H,$$

where $D' = \mathrm{diag}(D'_{11}, \ldots, D'_{nn})$, $D'_{ii} = \sum\limits_{j} H(i,j)$. One can see that

$$D'_{ii} = \sum_{j=1}^{n} \sum_{\ell=1}^{r} \omega_\ell \sum_{k=1}^{K_\ell} Z_\ell(i,k) Z_\ell(j,k) = \sum_{\ell=1}^{r} \omega_\ell N_\ell(i), \qquad (15)$$

where $N_\ell(i)$ is the size of the cluster which includes point x_i in ℓ-th partition variant.

Using H instead of the similarity matrix W in (8), and the matrix D' defined in (15), we obtain cluster ensemble based predictions in the form given by (12), (13).

WSR-LRCM algorithm

Weakly Supervised Regression algorithm based on the Low-Rank representation of the co-association matrix (WSR-LRCM):

Input:
X: dataset
a_i, σ_i, $i = 1, \ldots, n_1$: uncertain input parameters for labeled and inaccurately labeled points;
r, $\mathbf{\Omega}$: number of runs and set of parameters for the k-means clustering (number of clusters, maximum number of iterations, parameters of the initialization process).
Output:
a^*, σ^*: predicted estimates of uncertain parameters for objects from sample **X** (including predictions for the unlabeled sample).

WSR-LRCM algorithm

Steps:

1. Generate r variants of clustering partition for parameters randomly chosen from $\mathbf{\Omega}$; calculate weights $\omega_1, \ldots, \omega_r$.
2. Find graph Laplacian in the low-rank representation;
3. Calculate predicted estimates of uncertainty parameters using (12) and (13) .

end.

Numerical examples

First example with artificial data

Dataset: a mixture of two multidim. normal distributions $\mathcal{N}(m_1, \sigma_X I)$, $\mathcal{N}(m_2, \sigma_X I)$, m_1, $m_2 \in \mathbb{R}^8$, $d = 8$

Noise: 2 independent $\mathcal{U}(0,1)$.

Ground truth: $Y = 1 + \varepsilon$, $Y = 2 + \varepsilon$, $\varepsilon \in \mathcal{N}(0, \sigma_\varepsilon^2)$

MC generates samples of the given size n according to the specified distribution mixture.
Training: 66.6%, $\mathbf{X}_{train}$,
Test: 33.4%, $\mathbf{X}_{test}$.

In the training dataset:
10% fully labeled samples;
20% inaccurately labeled objects;
rest: unlabeled data.

First example with artificial data

This partitioning mimics a typical situation in the weakly supervised learning: a small number of accurately labeled instances, medium sized uncertain labelings and a lot of unlabeled examples.

To model the inaccurate labeling, we use the parameters defined in (1):
$\sigma_i = \delta \cdot \sigma_Y$,
σ_Y is a standard deviation of Y over labeled data,
$\delta > 0$.

First example with artificial data

The ensemble variants are generated by random initialization of centroids; to increase the diversity of base clusterings, we set the number of clusters in each run as $K = 2, \dots, K_{max}$, where $K_{max} = 2 + r$, and $r = 10$ is the ensemble size.
Parameters (Objective functional) $\beta = 0.001$ and $\gamma = 0.001$ were estimated.
The quality of prediction is estimated on the test sample as the Mean Wasserstein Distance between the predicted according to (12), (13) and ground truth values of the parameters:

$$\mathrm{MWD} = \frac{1}{n_{test}} \sum_{x_i \in \mathbf{X}_{test}} \left((a_i^{true} - a_i^*)^2 + \sigma_i^{*2} \right),$$

where n_{test} is the test sample size, and $a_i^{true} = y_i^{true}$ the true value of the target feature.

First example with artificial data

We compare the suggested method WSR-LRCM with its simplified version. The output predictions were calculated according to (6) and (7).

Repeated 40 times.

$m_1 = (0, \ldots, 0)^\top$, $m_2 = (10, \ldots, 10)^\top$, $\sigma_X = 3$, and $\delta = 0.1$.

n	σ_ε	WSR-LRCM		WSR-RBF	
		MWD	time (sec)	MWD	time (sec)
1000	0.01	0.002	0.04	0.007	0.04
	0.1	0.012	0.04	0.017	0.04
	0.25	0.065	0.04	0.070	0.04
5000	0.01	0.001	0.14	0.004	1.71
	0.1	0.011	0.14	0.014	1.72
	0.25	0.064	0.15	0.067	1.75
10000	0.01	0.001	0.33	0.002	9.40
	0.1	0.011	0.33	0.012	9.35
	0.25	0.064	0.33	0.065	9.36
10^5	0.01	0.001	6.72	-	-
10^6	0.01	0.001	89.12	-	-

Weakly supervised WSR-LRCM vs. semi-supervised SSR-RBF

$n = 1000$, $\sigma_\varepsilon = 0.1$

Table: Results of experiments with WSR-LRCM and SSR-RBF algorithms. Averaged MWD estimates are calculated for different values of parameter δ.

δ	0.1	0.25	0.5
WSR-LRCM	0.012	0.017	0.038
SSR-RBF	0.051	0.051	0.051

δ accounts for the degree of uncertainty: the larger its value is, the more similar the results become.

See more in [Berikov, V., Litvinenko, A. (2021). Weakly Supervised Regression Using Manifold Regularization and Low-Rank Matrix Representation. In: Pardalos, P., Khachay, M., Kazakov, A. (eds) MOTOR 2021. LNCS, vol 12755. Springer, Cham. `https://doi.org/10.1007/978-3-030-77876-7_30`]

The second example with real data

Gas Turbine CO and NOx Emission Data Set: UC Irvine Machine Learning Repository: Gas Turbine CO and NOx Emis- sion Data Set. https://archive.ics.uci.edu/ml/datasets/Gas+Turbine+CO+and+NOx+Emission+Data+Set (06 Apr 2021).
11 features (temperature, pressure, humidity, etc.) of a gas turbine.

The monitoring was carried out during 2011-2015.

Predicted outputs: Carbon monoxide (CO) and Nitrogen oxides (NOx).

The second example with real data

We make predictions for CO over the year 2015 (in total, 7384 observations).
Datasets: learning (66.6%) and test (33.4%) samples.

Accurately labeled sample is 1% from the entire dataset;
Inaccurately labeled instances 10% ;
Unlabeled samples: the rest.
Used k-means clustering (the number of clusters varies from 100 to $100 + r$).
Forecast: the averaged MWD for
WSR-LRCM is 1.85
SSR-RBF is 5.18.

The second example with real data

WSR-LRCM vs. fully supervised algorithms,
We calculate the standard Mean Absolute Error (MAE) using estimates of a^* defined in (12) as the predicted feature outputs:

$$\text{MAE} = \frac{1}{n_{test}} \sum_{x_i \in \mathbf{X}_{test}} |y_i^{true} - a_i^*|.$$

	MAE	time
WSR-LRCM, 100+ clusters	0.634	1.99
Random Forest (RF), 300 trees	0.774	0.35
Linear Regression (LR)	0.873	0.38

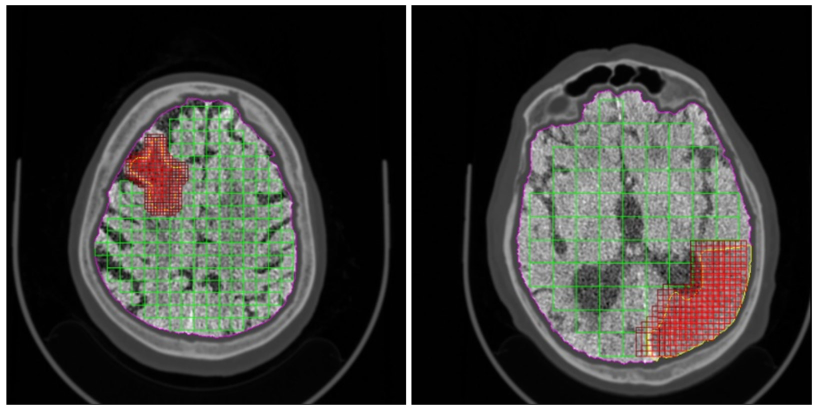
From the experiments, one may conclude that the proposed WSR-LRCM gives more accurate predictions than other compared methods in case of a small proportion of labeled sample.

Conclusion

1. Introduced a weakly supervised regression method (WSR-LRCM) using the manifold regularization technique.
2. To model uncertain labeling, we have used normal distribution
3. The measure of similarity between uncertain labelings was formulated in terms of the Wasserstein distance between probability distributions.
4. Ensemble clustering is used for obtaining the co-association matrix which we consider as the similarity matrix.
5. WSR-LRCM is faster
6. Additional information on uncertain labelings improves the regression quality.

Other applications

Masks of healthy (purple contours) and affected (yellow contours) brain tissues and fragments emplacement for healthy and affected tissues (green and red squares respectively) for $FS = 16$ (left) and 30 (right). Red color saturation indicates the probability of assigning the fragment to the affected tissue.



Nedel'ko V et al. Comparative Analysis of Deep Neural Network and Texture-Based Classifiers for Recognition of Acute Stroke using Non-Contrast CT Images // 2020 Ural Symposium (USBEREIT). – IEEE, 2020, pp 376-379. DOI:10.1109/USBEREIT48449.2020.9117784

1. V. Berikov, I. Pestunov, Ensemble clustering based on weighted co-association matrices: Error bound and convergence properties, Pattern Recognition. 2017. Vol. 63. P. 427-436.
2. V. Berikov, A. Litvinenko, Semi-Supervised Regression using Cluster Ensemble and Low-Rank Co-Association Matrix Decomposition under Uncertainties, arXiv:1901.03919, (2019).
3. V. Berikov, N. Karaev, A. Tewari, Semi-supervised classification with cluster ensemble. In Engineering, Computer and Information Sciences (SIBIRCON), International Multi-Conference. 245-250. IEEE. (2017)
4. V.B. Berikov, Construction of an optimal collective decision in cluster analysis on the basis of an averaged co-association matrix and cluster validity indices. Pattern Recognition and Image Analysis. 27(2), 153-165 (2017)
5. V. Berikov, Cluster Ensemble with Averaged Co-Association Matrix Maximizing the Expected Margin, CEUR Workshop Proceedings, `http://ceur-ws.org/Vol-1623/papercpr1.pdf`, 2019
6. V.B. Berikov, A. Litvinenko, The influence of prior knowledge on the expected performance of a classifier. Pattern recognition letters 24 (15), 2537-2548, (2003)
7. V Berikov, A Litvinenko, Methods for statistical data analysis with decision trees, Novosibirsk, Sobolev Institute of Mathematics, 2003, `http://www.math.nsc.ru/AP/datamine/eng/context.pdf`