Metamodels for complex structured objects classification

Ilya Zharikov, Roman Isachenko, Artem Bochkarev

Introduction

This project is dedicated to multiclass classification of complex structured objects (for which we don't have explicit features). The problem arises in many applications such as image recognition, signal processing or time series classification. We will focus on multiclass multivariate time series classification. In this setup time series are regarded as complex structured objects without explicit feature description. This is reasonable because we can't operate with original features as time series might be of different size, not aligned [1] or even multiscaled.

We investigate classification of accelerometer time series [2]. The data is time series of acceleration from three axis, which is sensed by mobile phone or other portable device with accelerometer. The task is to predict the activity a person is performing. List of activities might include walking, running, sitting or walking up/down the stairs.

In general the problem of classifying complex structured objects can be split in two distinctive procedures. First, we need to extract informative features, and then we use those features as input to some classifier to obtain final model. For simplicity, we assume that these two procedures can be built and analyzed separately. In our project we focused mainly on comparing different methods of feature generation [3, 4]. Extracted features can be later used for building classifiers and feature selection algorithms.

The first approach for feature generation is calculating expertly defined functions of time series [5]. These functions include average value, standard deviation,

mean absolute deviation and distribution for each component. We consider this approach a baseline, as it is the simplest method we use.

We compare baseline with more sophisticated parametric feature generation methods. One of them is autoregressive model [6]. For each time series we build parametric model and use those parameters as features for classification. We also consider the model of singular spectrum analysis of time series [7]. We use eigenvalues of trajectory matrix as features for building classifier.

In the first part of these report we give definitions and make problem statement. In the second part we describe all of the proposed approaches in more detail. Last, we make the experiment on real accelerometer datasets [8, 9], compare all methods and give conclusions and recommendations for practical use.

Problem Statement

Let $\mathcal{S}$ be a space of complex structured objects, Y is a finite set of class labels. Denote by $\mathfrak{D} = \{(s_i, y_i)\}_{i=1}^m$ – given sample, where $s_i \in \mathcal{S}$ and $y_i \in Y$.

We consider the problem of recovering the function $f : \mathcal{S} \to Y$

$$y = f(s).$$

Let $L(f, \mathfrak{D})$ be an error function which expresses the classification error of the function f over the sample $\mathfrak{D}$. Our goal is to determine function f^* which minimizes the error

$$f^* = \arg\min_f L(f, \mathfrak{D}). \tag{1}$$

We assume that the target function f^* belongs to the class of function compositions $f = g \circ h$, where

- $h : \mathcal{S} \to H$ is a map from the original space $\mathcal{S}$ to the feature space $H \subset \mathbb{R}^n$;

- $g : H \times \Theta \to Y$ is a parametric map from the feature space H to the space of class labels Y. The function g is parametrized by a vector parameter $\boldsymbol{\theta} \in \Theta$.

The determining of the function f^* is equivalent to determining the functions h^* and g^*.

In this project we consider the following ways of generating the feature space H:

- expert functions based on prior knowledge of the original objects. These functions can be expressed as a set of statistics $\{h_i\}_{i=1}^n$, where $h_i : \mathcal{S} \to \mathbb{R}$. Thus, the description $\mathbf{h}$ of the object s is the value of these statistics on the object

$$\mathbf{h} = h(s) = (h_1(s), \ldots, h_n(s)).$$

- hypothesis of data generation. In this case the features are the estimated parameters of the considered hypothesis. Let $S(s, \mathbf{h}, \boldsymbol{\lambda})$ be the error function which specifies the hypothesis, e.g. one could define the function S as negative log-likelihood function [to do link]. The optimal parameters $\widehat{\mathbf{h}}$ for object s is obtained by

$$\widehat{\mathbf{h}} = \arg\min_{\mathbf{h}} S(s, \mathbf{h}, \boldsymbol{\lambda}). \tag{2}$$

The parameter $\boldsymbol{\lambda}$ is external structural parameter for the function S. The equation (2) determines the feature map $h : \mathcal{S} \to H$.

Given appropriate feature space H and feature map h we transform our original sample $\mathfrak{D} = \{s_i, y_i\}_{i=1}^m$ with complex structured objects to the new sample $\mathfrak{D}_H = \{\mathbf{h}_i, y_i\}_{i=1}^m$, where $\mathbf{h}_i = h(s_i) \in H$. The function $g(\mathbf{h}, \boldsymbol{\theta})$ is defined by its parameters vector $\boldsymbol{\theta} \in \Theta$. The optimal parameters $\widehat{\boldsymbol{\theta}}$ are given by

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} L(g(\cdot, \boldsymbol{\theta}), \mathfrak{D}_H),$$

where $L(\cdot, \cdot)$ is an analogue of the function (1).

In our project we consider accelerometer time series as complex objects. Time series is represented in following way:

$$s = (x_1, \ldots, x_T) \in \mathcal{S},$$

where T denotes the length of time series. Now let us expand on different approaches of feature generation.

Feature generation

Expert functions

Given a set of complex objects $\{s_i\}_{i=1}^m$ we can extract features in a non-parametric way with a set of expert functions $\{h_j\}_{j=1}^n$. For time series ![[link],](link) these functions could be mean, standard deviations, mean absolute deviations and distributions of the acceleration. The main drawback of this approach is that we are restricted by our choice of expert functions and for some types of data these functions might be impossible to derive.

Autoregressive model

In this method we assume autoregressive model ![[link]](link) of the order n as a hypothesis for generation of time series s. Each component of the object s is a linear combination of the previous n components

$$x_t = w_0 + \sum_{j=1}^n w_j x_{t-j} + \varepsilon_t,$$

where ε_t is a random noise. Prediction of the autoregressive model is defined by

$$\hat{x}_t = w_0 + \sum_{j=1}^n w_j x_{t-j}. \tag{3}$$

For this method n is a structural parameter and $\boldsymbol{\lambda} = n$.

Feature description $\mathbf{h}$ of the object s is given by optimal parameters of autoregressive model $\widehat{\mathbf{w}} = \{\widehat{w}_j\}_{j=0}^n$ for time series s_i. The hypothesis error function (2) in this case is the squared error between the original object s and its prediction of the model (3).

$$\mathbf{h} = \widehat{\mathbf{w}} = \operatorname*{arg\,min}_{\mathbf{w} \in \mathbb{R}^{n+1}} S(s, \mathbf{w}, \boldsymbol{\lambda}) = \operatorname*{arg\,min}_{\mathbf{w} \in \mathbb{R}^{n+1}} \left(\sum_{t=n+1}^T \|x_t - \hat{x}_t\|^2 \right). \tag{4}$$

The problem (4) could be easily converted to the linear regression problem. Hence, for each initial time series s we have to solve linear regression problem.

Singular spectrum analysis

Alternative hypothesis for generation of time series is SSA (Singular Spectrum Analysis) model. We construct trajectory matrix for each time series $s = (x_1, \ldots x_T)$:

$$\mathbf{X} = \begin{pmatrix} x_1 & x_2 & \ldots & x_n \\ x_2 & x_3 & \ldots & x_{n+1} \\ \ldots & \ldots & \ldots & \ldots \\ x_{T-n+1} & x_{T-n+2} & \ldots & x_T \end{pmatrix}.$$

Here n is an external structural parameter. Let find singular decomposition of matrix $\mathbf{X}^\mathsf{T}\mathbf{X}$:

$$\mathbf{X}^\mathsf{T}\mathbf{X} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\mathsf{T},$$

where $\mathbf{U}$ is a unitary matrix and $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ whose entries λ_i are eigenvalues of $\mathbf{X}^\mathsf{T}\mathbf{X}$. In this case we use the spectrum of the matrix $\mathbf{X}^\mathsf{T}\mathbf{X}$ as feature description $\mathbf{h}$ of the object s

$$\mathbf{h} = (\lambda_1, \ldots, \lambda_n).$$

Splines

One could approximate the time series by splines [link]. Spline is defined by its parameters:

- $\{\xi_\ell\}_{\ell=1}^L$ — the set of knots;

- $\{\mathbf{w}_\ell\}_{\ell=1}^{L-1}$ — parameters of the models are built on the interval $[\xi_\ell; \xi_{\ell+1}]$. The dimension of each parameter vector $\mathbf{w}_\ell$ depends on the spline order.

The feature description of the time series could be assumed as union of these parameters.

$$\boldsymbol{h} = h(s) = (\xi_1, \ldots, \xi_L, \mathbf{w}_1, \ldots, \mathbf{w}_{L-1}).$$

Classification

In this section we will describe our approach to classification of time series using newly generated feature. We use three different classification models: logistic

regression, SVM and random forests.

Quality measure

We chose accuracy score as our main quality measure function. We did that because this approach was used in previous articles [3, 4] and this measure is easy to understand. Accuracy score is a relation correctly labeled objects and the total amount of objects in dataset:

$$\text{accuracy}(y, \hat{y}) = \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}(y_i = \hat{y}_i),$$

where $\hat{y}_i$ is a prediction of the classifier. We calculated this measure for binary case (when we predict each class in particular) and in multilabel case.

Multiclass classification

As we had numerous labels in our datasets we had to choose one of the multiclass approaches to classification. We decided to use one-vs-rest classification as a simple, yet effective approach. The main idea is that we train binary classifiers for each class label and then, on the prediction step, we classify new object according to the most confident classifier.

Logistic regression

The first approach to classification we played with was regularized logistic regression. We seek to minimize the following loss function:

$$L(\boldsymbol{\theta}) = \sum_{i=1}^{m} \log(1 + \exp(-(\boldsymbol{\theta}^T \mathbf{h}_i) y_i)) + \frac{\lambda}{2} \|\boldsymbol{\theta}\|^2$$

$$\boldsymbol{\theta}^* = \arg\min_{\theta} L(\boldsymbol{\theta})$$

We make a prediction for a new object in a following way:

$$\hat{y} = \text{sgn}(\boldsymbol{\theta}^T \mathbf{h})$$

SVM

We also used classical binary SVM. The problem in this case can be formulated in a following way:

$$\boldsymbol{\theta}^* = \begin{pmatrix} \mathbf{w} \\ b \\ \boldsymbol{\xi} \end{pmatrix} = \arg\min_{\mathbf{w}^*,b^*,\boldsymbol{\xi}^*} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m} \xi_i$$

$$\text{s.t.} \quad y_i(\mathbf{w}^T\mathbf{h}_i + b) \geqslant 1 - \xi_i$$

$$\xi_i \geqslant 0, \quad 1 \leqslant i \leqslant m.$$

For new objects we can make a prediction

$$\hat{y} = \text{sgn}(\mathbf{w}^T\mathbf{h} + b)$$

Random Forest

TO DO

TO DO

- the function h is vectorial (discuss notations);

- add explanation(pics) of autoregressive model;

- splines;

- ts example pic.

Experiment

In this paper we consider 2 different smart phone based datasets: WISDM [8] and USC-HAD [9]. Data from smart phone accelerometer consists of information about acceleration along each of three axis. Time difference between measurements equals 50 ms. Each time series belongs to one activity. The distributions of time series activities for each datasets are presented in Table 1.

Table 1: Activities distributions

(a) WISDM

Activity	# objects	
Standing	229	5.3%
Walking	1917	44.4%
Upstairs	466	10.8%
Sitting	277	6.4%
Jogging	1075	24.9%
Downstairs	357	8.3%
Total	4321	

(b) USC-HAD

Activity	# objects	
Walking-downstairs	951	7%
Walking-upstairs	1018	7.4%
Walking-forward	1874	13.8%
Walking-right	1305	9.6%
Walking-left	1280	9.4%
Elevator-up	764	5.6%
Elevator-down	763	5.6%
Standing	1167	8.6%
Sitting	1294	9.5%
Sleeping	1860	13.7%
Jumping	495	3.6%
Running	849	6.2%
Total	13620	

For each dataset we applied the feature generation approaches described above: expert functions, autoregressive model, SSA, splines. We used three different widely used classification model for each generated feature description: logistic regression, support vector machine and random forest. The external structural parameters for feature generation procedures, such as the length n for autoregression and the window width n for SSA, !params for splines!, were tuned using cross validation procedure. The hyperparameters for classification models were also tuned using cross validation procedure.

Expert functions for time series TO DO. Autoregressive parameters + SSA TO DO.

We fit splines for time series using *scipy* [link] python library. The spline order equals 3. The knots $\{\xi_\ell\}_{\ell=1}^L$ for splines were distributed uniformly. The number L were chosen implicitly by fitting smoothing parameter s, it is possible because $L \sim \frac{1}{s}$. The fitting was constructed as follows. Firstly, there was the initialization step to find appropriate bounds for smoothing parameter. The next step is finding the smoothing parameter in this interval using bi-search approach.

The results of the experiments for the both datasets is presented in Figure 1. We also carried out the experiment for union of all generated features.

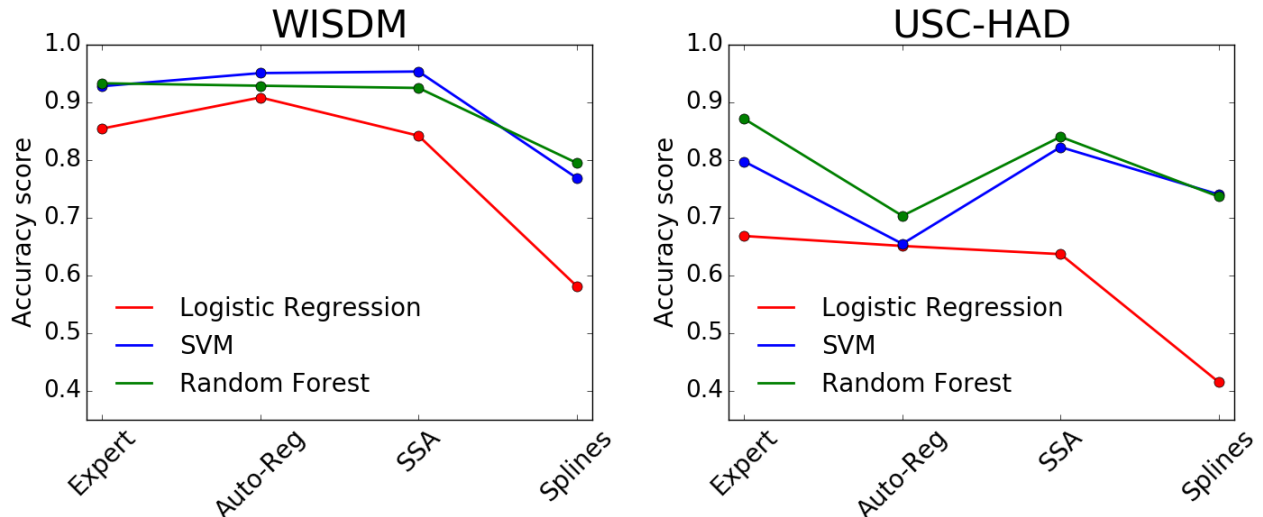Figure 1: Accuracy scores

Table 2: Accuracy scores for WISDM dataset, all features

All	Standing	Walking	Upstairs	Sitting	Jogging	Downstairs
0.983	0.995	0.993	0.993	0.995	0.997	0.993

References

[1] Pierre Geurts. Pattern extraction for time series classification. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 115–127. Springer, 2001.

[2] Wen Wang, Huaping Liu, Lianzhi Yu, and Fuchun Sun. Human activity recognition using smart phone embedded sensors: A linear dynamical systems method. In *Neural Networks (IJCNN), 2014 International Joint Conference on*, pages 1185–1190. IEEE, 2014.

[3] ME Karasikov and VV Strijov. Feature-based time-series classification. *Intelligence*, 24(1):164–181.

[4] M.P. Kuznetsov and Ivkin N.P. Time series classification algorithm using combined feature description. *Machine Learning and Data Analysis*, 1(11):1471–1483.

[5] Jennifer R Kwapisz, Gary M Weiss, and Samuel A Moore. Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter*,

12(2):74–82, 2011.

[6] Yu P Lukashin. Adaptive methods of short-term forecasting of time series. *M.: Finance and statistics*, 2003.

[7] Hossein Hassani. Singular spectrum analysis: methodology and comparison. 2007.

[8] Wisdm dataset. http://www.cis.fordham.edu/wisdm/dataset.php.

[9] The usc human activity dataset. http://www-scf.usc.edu/~mizhang/datasets.html.