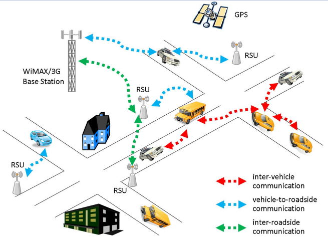
Estimation of arbitrary nonstationary dependencies in a linear observation space

P. Turkov O. Krasotkina V. Mottl

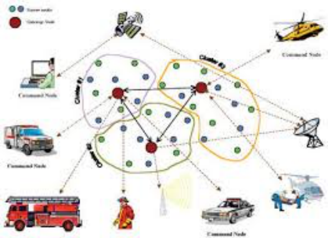
Tula State University
Moscow State University
Computing Centre of Russian Academy of Science

International Conference IIP-11
Barcelona, Spain

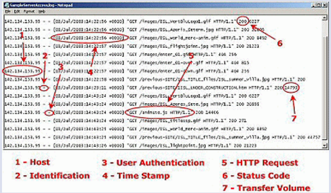
Traffic management



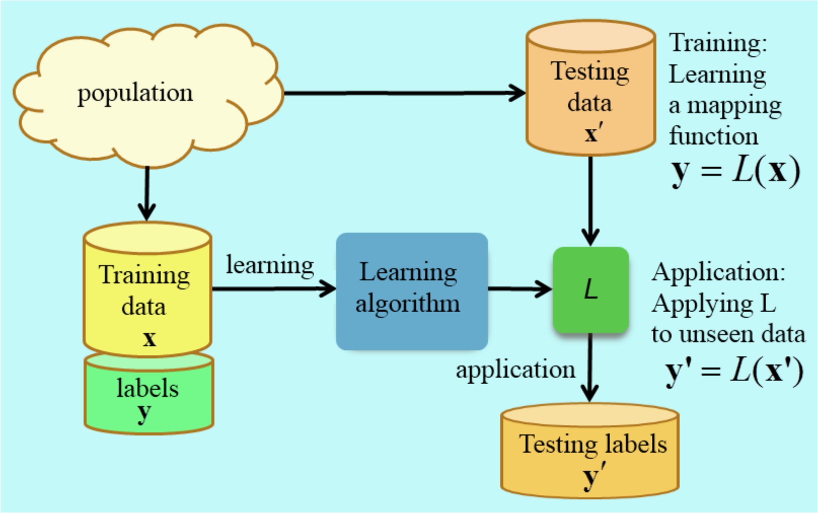
Sensor network

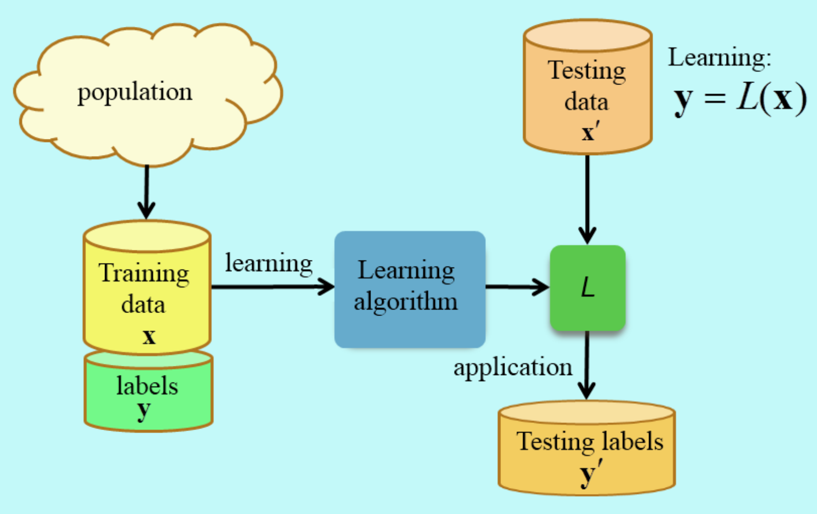


Web logs

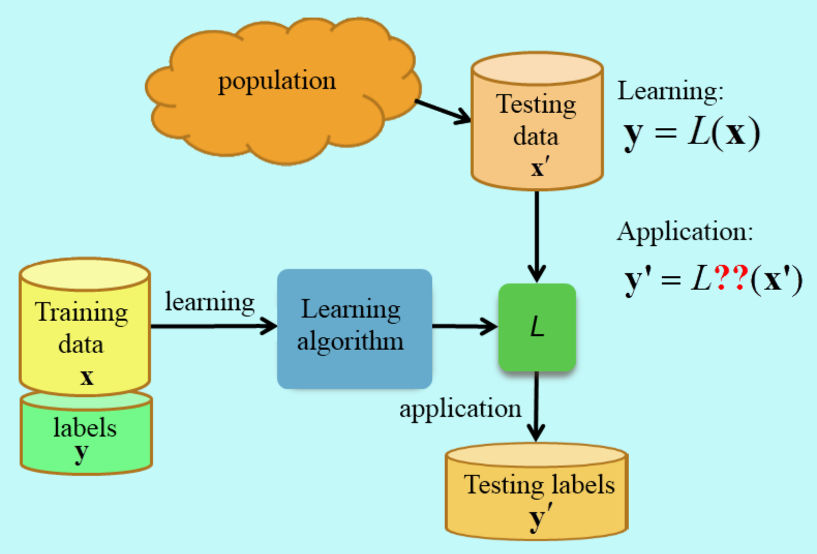


Classical supervised learning




population
Training
data
x
labels
y
learning
Learning
algorithm
L
Testing
data
x′
Learning:
y = L(x)
application
Testing labels
y′

Learning concept drift

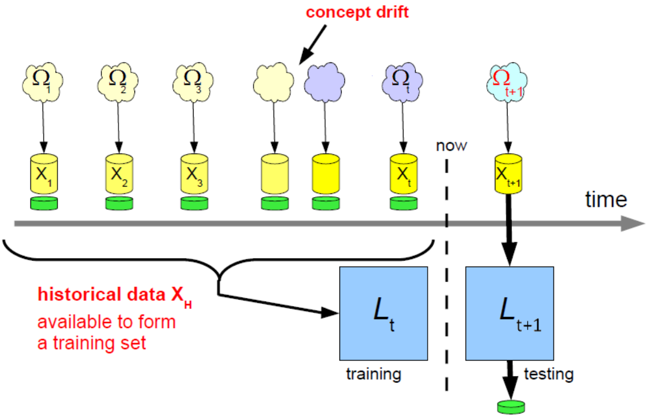


- Let every instance of the environment $\omega \in \Omega$ has hidden property $y \in \{1, -1\}$ and is presented by a point in a linear feature space $\mathbf{x}(\omega) = \big(x^1(\omega), \ldots, x^n(\omega)\big) \in \mathbb{R}^n$.

- In the data stream concept we suppose that instances arrive sequentially in time

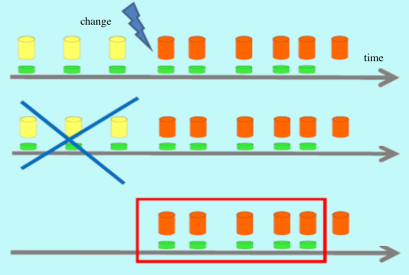
$$\big\{(\mathbf{X}_t,\ \mathbf{Y}_t,\ t)\big\}_{t=1}^{T},$$

$(\mathbf{X}_t, \mathbf{Y}_t) = \{(\mathbf{x}_{k,t}, y_{k,t})\}_{k=1}^{N_t}$ - subset of instances in moment t.


concept drift
Ω_1
Ω_2
Ω_3
Ω_t
Ω_{t+1}
now
X_1
X_2
X_3
X_t
X_{t+1}
time
historical data X_H
available to form
a training set
L_t
training
L_{t+1}
testing

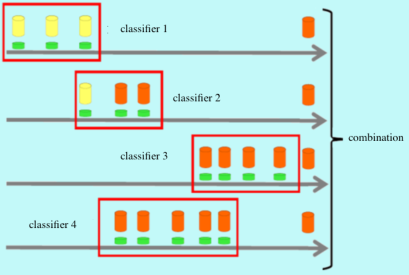
Constant window(Widmer, Kubat, 1996)

Changing window (Patist, 2007)

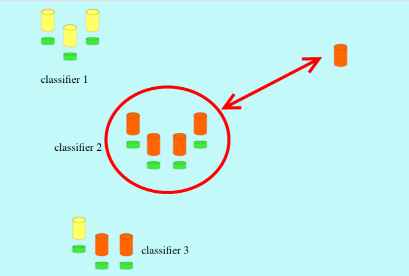


where

Bagging & Boosting (Kolter, 2007)

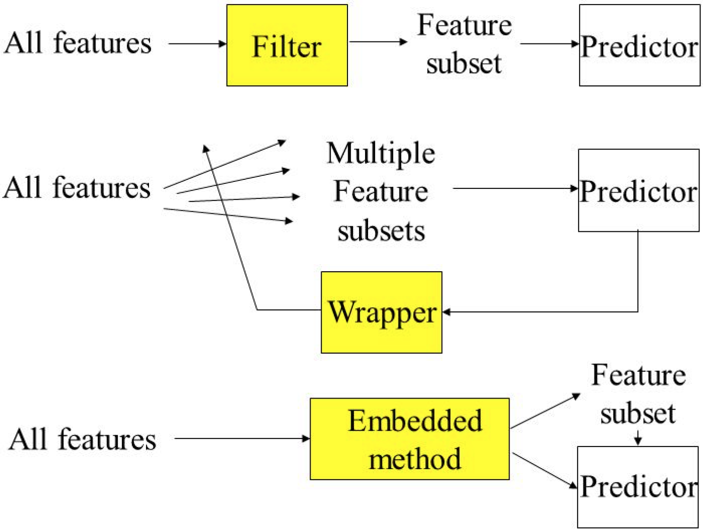


Stacking (Street, 2001)



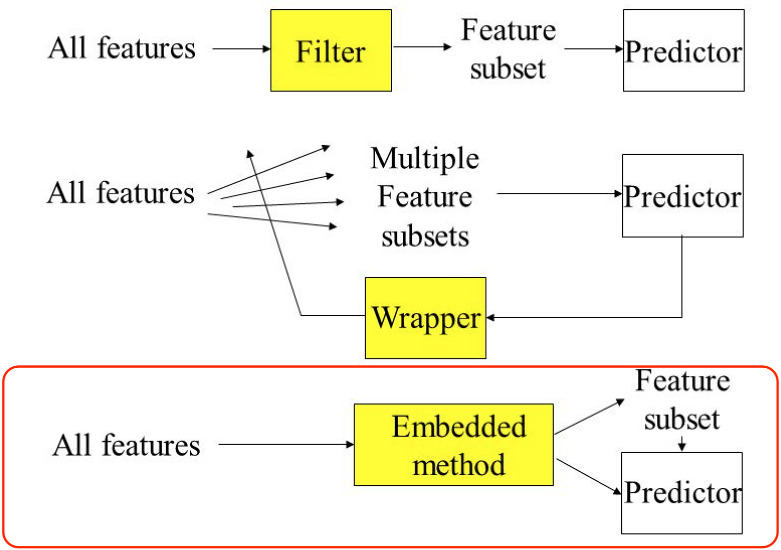
where

Existing data stream classification techniques can't reduce the amount of features being processed!!!

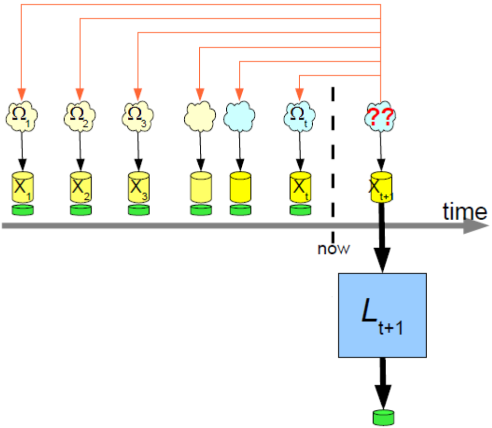

All features
Filter
Feature subset
Predictor
All features
Multiple Feature subsets
Predictor
Wrapper
Feature subset
All features
Embedded method
Predictor

Feature selection techniques in supervised learning

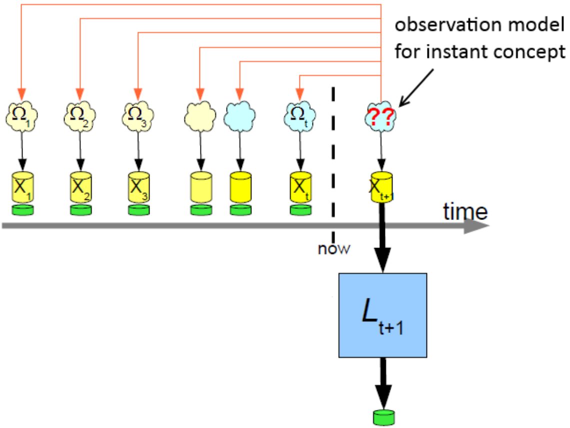
The state of the art



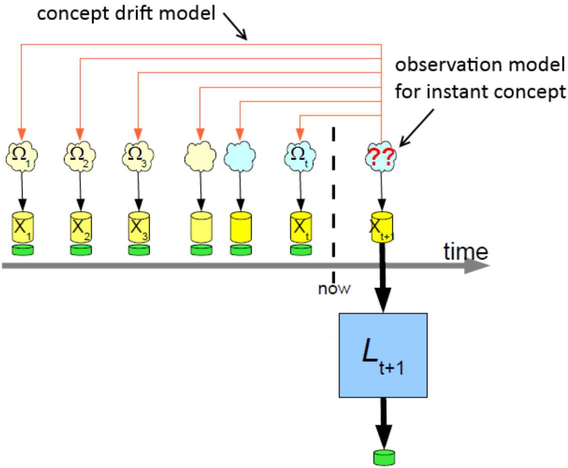
What it need to solve concept drift problem??



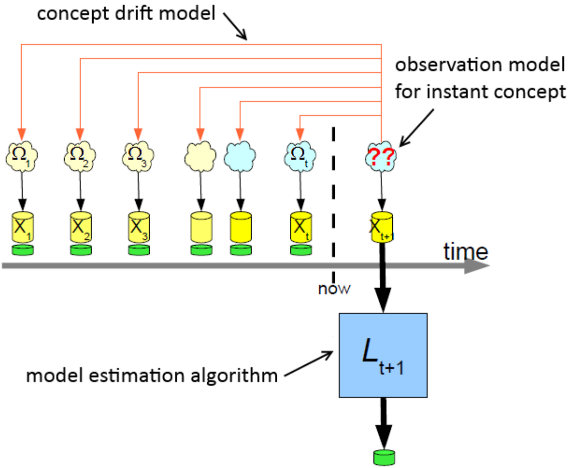
1. Observation model for instant concept



2. Concept drift model



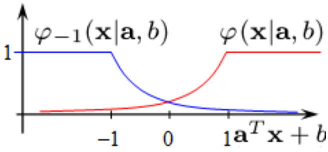
3. Model estimation algorithm



- Observation model for instant concept is the discriminant function:
  $f\big(\mathbf{x}(\omega)\big) = \mathbf{a}^T\mathbf{x} + b > 0$ if $y(\omega) = 1$,and < 0 if $y(\omega) = -1$.

- Concept drift leads to undestanding the parameters $\mathbf{a}$ and b as time functions:
  $\mathbf{a}_t : T \to \mathbb{R}^N; b_t : T \to \mathbb{R}$

As the probabilistic model of the data source, we shall consider two parametric families of distribution densities $\varphi_1(\mathbf{x}|\mathbf{a_t}, b_t)$ and $\varphi_{-1}(\mathbf{x}|\mathbf{a_t}, b_t)$, $\mathbf{a}_t \in \mathbb{R}^n$, $b_t \in \mathbb{R}$, associated with two class indexes $y = \pm 1$.



$$\varphi_1(\mathbf{x}|\mathbf{a_t}, b_t) = \begin{cases} 1, \mathbf{a_t}^T\mathbf{x} + b_t > 1, \\ \exp\left[-c\left(1 - (\mathbf{a_t}^T\mathbf{x} + b_t)\right)\right], \mathbf{a_t}^T\mathbf{x} + b_t < 1, \end{cases}$$

$$\varphi_{-1}(\mathbf{x}|\mathbf{a_t}, b_t) = \begin{cases} 1, \mathbf{a_t}^T\mathbf{x} + b_t < -1, \\ \exp\left[-c\left(1 + (\mathbf{a_t}^T\mathbf{x} + b_t)\right)\right], (\mathbf{a_t}^T\mathbf{x} + b_t) > -1. \end{cases}$$

- A posterior probability density of classes $y_{j,t} = \pm 1$:

$$\varphi_y(\mathbf{x}|\mathbf{a_t}, b_t) = \begin{cases} 1, \mathbf{a_t}^T\mathbf{x} + b_t > 1, \\ \exp\left[-c\left(1 - y(\mathbf{a_t}^T\mathbf{x} + b_t)\right)\right], \mathbf{a_t}^T\mathbf{x} + b_t < 1. \end{cases}$$

- For the training subset $\mathbf{X}_t, \mathbf{Y}_t$ in the time moment t joint distribution function is:

$$\Phi(\mathbf{Y}_t|\mathbf{X}_t, \mathbf{a}_t, b_t) = \prod_{j=1}^{N_t} \varphi_y(\mathbf{x}|\mathbf{a_t}, b_t).$$

- The key element of our approach to the concept drift problem is treating the time-varying parameters of the hyperplane $\mathbf{w}_t = (\mathbf{a}_t, b_t)$ as a hidden random processes, that possesses the Markov property:

$$\begin{aligned} &\mathbf{w}_t = q\mathbf{w}_{t-1} + \boldsymbol{\xi}_t \in \mathbb{R}^{n+1},\ E\left(\boldsymbol{\xi}_t\right) = \mathbf{0}, \\ &E(\boldsymbol{\xi}_t\boldsymbol{\xi}_t^T) = \mathbf{Diag}(d_1, \dots, d_{n+1}), \\ &d_i = (1-q^2)r_i,\ i = 1, \dots, n,\ d_{n+1} = 1-q^2. \end{aligned} \tag{1}$$

  Here $\boldsymbol{\xi}_t = \left(\xi_{i,t}, i = 1, \dots, n+1\right)$ is the vector white noise.

- If $|q| < 1$, each elementary random process $w_{i,t}$ is stationary and ergodic and

$$\begin{aligned} &E(w_{1,t}) = E(w_{n+1,t}) = 0, \\ &Var(w_{i,t}) = Var(a_{i,t}) = \tfrac{d_i}{1-q^2} = r_i,\ i = 1, \dots, n, \\ &Var(w_{n+1,t}) = Var(b_t) = \frac{d_{n+1}}{1-q^2} = 1. \end{aligned} \tag{2}$$

Under the assumption that the white noise is assumed to be normally distributed, the conditional probability density of each hyperplane parameter vector $\mathbf{w}_t$ with respect to its immediately previous value $\mathbf{w}_{t-1}$ will be normal, too:

$$\psi(\mathbf{w}_t|\mathbf{w}_{t-1},\mathbf{r}) \propto \mathcal{N}\left(\mathbf{w}_t|q\mathbf{w}_{t-1},\mathbf{D}_\mathbf{r}\right) = \frac{1}{|\mathbf{D}_\mathbf{r}|^{1/2}(2\pi)^{n/2}}\exp\left(-\frac{1}{2}(\mathbf{w}_t - q\mathbf{w}_{t-1})^T\mathbf{D}_\mathbf{r}^{-1}(\mathbf{w}_t - q\mathbf{w}_{t-1})\right).$$

The a priori distribution density of the hidden sequence of hyperplane parameters:

$$\Psi(\mathbf{w}_t, t=1,...,T|\mathbf{r}) = \prod\nolimits_{t=2}^{T}\psi(\mathbf{w}_t|\mathbf{w}_{t-1},\mathbf{r}).$$

Feature selection model

We shall consider independent a priori gamma distributions of inverse variances

$$\gamma\big((1/r_i)|\alpha,\theta\big) \propto (1/r_i^{\alpha-1})\exp\big(-\theta(1/r_i)\big).$$

Joint a priori distribution density of inverse variances $1/r_i$ is

$$G(1/r_1,\ldots,1/r_p|\alpha,\theta) \propto \Big(\prod_{i=1}^{p}(1/r_i^{\alpha-1})\exp\big(-\theta(1/r_i)\big)\Big).$$

$\alpha = 1 + 1/(2\mu), \beta = 1/(2\mu),$

Feature selection model: properties

if $\mu \to 0$, the values $1/r_i$ are nonrandom $1/r_i \cong ... \cong 1/r_n \cong 1$ because $\left[E(1/r_i)\to 1, Var(1/r_i)\to 0\right]$, and all the squared elements of the direction vector a_i^2 are equally penalized. But the growing parameter $\mu \to \infty$ allows the independent nonnegative values $1/r_i$ to arbitrarily differ from each other $\left[E(1/r_i)\to\infty, Var(1/r_i)\to\infty\right]$, and the requirements $\left[\ln G(\mathbf{r}|\mu)\to \max,\ \ln\Psi(\mathbf{a}|\mathbf{r})\to \max\right]$ enforce their growth $1/r_i\to\infty$.

The training criterion for estimating the concept drift model parameters

The joint distribution of the traning sample, concept drift model and feature selection model:

$$p\Big(\mathbf{w}_t, t=1,...,T, \mathbf{r} | (\mathbf{X}_t, \mathbf{Y}_t), t=1,...,T | c, \mu\Big) \propto \underbrace{\Phi\big(\mathbf{X}_t, t=1,...,T | \mathbf{Y}_t, \mathbf{w}_t, t=1,...,T, c\big)}_{training\ sample} \times \underbrace{\Psi(\mathbf{w}_t, t=1,...,T|\mathbf{r})}_{concept\ drift} \times \underbrace{G(\mathbf{r}|\mu)}_{feature\ selection}.$$

$$(\hat{\mathbf{w}}_t, t=1,...,T, \hat{\mathbf{r}} | c, \mu) = \arg\max_{\mathbf{w}_t, t=1,...,T, \mathbf{r}} p\Big(\mathbf{w}_t, t=1,...,T, \mathbf{r} | (\mathbf{X}_t, \mathbf{Y}_t), t=1,...,T, c, \mu\Big) = \arg\max_{\mathbf{w}_t, t=1,...,T, \mathbf{r}} \Big[\ln \Psi(\mathbf{w}_t, t=1,...,T|\mathbf{r}) + \ln G(\mathbf{r}|\mu) + \ln \Phi\big(\mathbf{X}_t, t=1,...,T | \mathbf{Y}_t, \mathbf{w}_t, t=1,...,T, c\big)\Big].$$

The training criterion for estimating the concept drift model parameters

$$(\hat{\mathbf{w}}_t, t=1,...,T, \hat{\mathbf{r}}|c,\mu) = \underset{\mathbf{w}_t, t=1,...,T,\mathbf{r}}{\arg\min}\ J(\mathbf{w}_t, t=1,...,T,\mathbf{r}|c,\mu),$$

$$J(\mathbf{w}_t, t=1,...,T,\mathbf{r}|c,\mu) = (T-1)\ln|\mathbf{D}_\mathbf{r}| +$$

$$\sum_{t=2}^{T}(\mathbf{w}_t - q\mathbf{w}_{t-1})^T\mathbf{D}_\mathbf{r}^{-1}(\mathbf{w}_t - q\mathbf{w}_{t-1}) - 2\ln G(\mathbf{r}|\mu) +$$

$$2c\sum_{t=1}^{T}\sum_{j=1}^{N_t}\max\left(0,\ 1 - y_{j,t}\mathbf{w}_t^T\mathbf{x}_{j,t}\right).$$

We use the group coordinate descent method for two groups of variables, namely, hyperplane parameters $(\mathbf{w}_t, t=1,...,T)$ and variances $\mathbf{r}=(1,...,n)$.

An approximate dynamic programming procedure for estimation of the drifting hyperplane

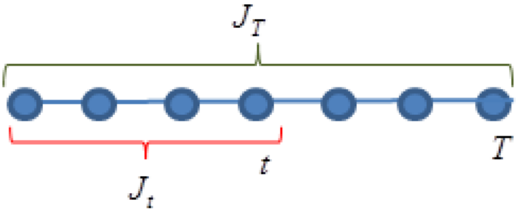
The criterion is pairwise separable, i.e., may be considered as a sum of elementary functions each of which is that of one vector variable $\mathbf{w}_t$ or two variables $(\mathbf{w}_{t-1}, \mathbf{w}_t)$ immediately adjacent in discrete time.

$$J(\mathbf{w}_t, t=1, ..., T|\mathbf{r}, c) = \\ \sum_{t=2}^{T} (\mathbf{w}_t - q\mathbf{w}_{t-1})^T \mathbf{D}_{\mathbf{r}}^{-1} (\mathbf{w}_t - q\mathbf{w}_{t-1}) + \\ 2c \sum_{t=1}^{T} \sum_{j=1}^{N_t} \max\left(0,\ 1 - y_{j,t}\mathbf{w}_t^T \mathbf{x}_{j,t}\right) \to \min,$$

An approximate dynamic programming procedure for estimation of the drifting hyperplane

We use for solving optimization problems of such a kind the well-known principle of dynamic programming. We consider the partial criterion with respect only to the initial part of the entire time interval $s=1,...,t$:

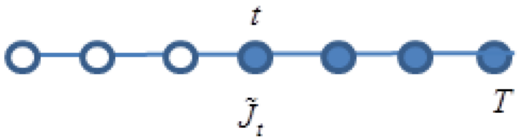
$$J_t(\mathbf{w}_s, s=1,...,t|\mathbf{r}, c) = \sum\nolimits_{s=2}^{t} (\mathbf{w}_s - q\mathbf{w}_{s-1})^T \mathbf{D}_{\mathbf{r}}^{-1} (\mathbf{w}_s - q\mathbf{w}_{s-1}) + 2c \sum\nolimits_{s=1}^{t} \sum\nolimits_{j=1}^{N_s} \max\left(0,\ 1 - y_{j,t} \mathbf{w}_s^T \mathbf{x}_{j,s}\right) \to \min,$$



An approximate dynamic programming procedure for estimation of the drifting hyperplane

If we minimize partial criterion by all the precedent variables $(\mathbf{w}_1, ..., \mathbf{w}_{t-1})$, the result will be function of $\mathbf{w}_t$:

$$\tilde{J}_t(\mathbf{w}_t|\mathbf{r}, c) = \min_{\mathbf{w}_s, s=1,...,t-1} J_t(\mathbf{w}_s, s=1,...,t|\mathbf{r}, c) = \min_{\mathbf{w}_1,...,\mathbf{w}_{t-1}} J_t(\mathbf{w}_1, ..., \mathbf{w}_{t-1}, \mathbf{w}_t|\mathbf{r}, c).$$



$$(\hat{\mathbf{w}}_t|\mathbf{r}, q, c) = \min_{\mathbf{w}_t \in \mathbb{R}^{n+1}} \tilde{J}_t(\mathbf{w}_t|\mathbf{r}, q, c).$$

A trick: Bellman function quadratic approximation

Let's replace the original piece-wise quadratic function $\tilde{J}_t$ on its quadratic approximation

$$\tilde{J}_t(\mathbf{w}_t) \cong \bar{J}_t(\mathbf{w}_t) = (\mathbf{w}_t - \bar{\mathbf{w}}_t)^T \bar{\mathbf{Q}}_t (\mathbf{w}_t - \bar{\mathbf{w}}_t),$$

with the preserving

- the minimum point $\bar{\mathbf{w}}_t = \arg\min \tilde{J}(\mathbf{w_t})$
- the hessian $\bar{\mathbf{Q}}_t = \nabla^2 \tilde{J}(\bar{\mathbf{w}}_t)$

Then Bellman function can be presented in such form:

$$\tilde{J}_t(\mathbf{w}_t) = (\mathbf{w}_t - \tilde{\mathbf{w}}_t)^T \tilde{\mathbf{Q}}_t (\mathbf{w}_t - \tilde{\mathbf{w}}_t) + \tilde{c}_t.$$

Re-estimation of feature weights for the fixed hyperplane drift

$$J(\mathbf{r}|\mathbf{w}_t, t=1, ..., T, \mu) = \\ (T-1)\ln|\mathbf{D}_{\mathbf{r}}^{-1}| + \sum_{t=2}^{T}(\mathbf{w}_t - q\mathbf{w}_{t-1})^T \mathbf{D}_{\mathbf{r}}^{-1}(\mathbf{w}_t - q\mathbf{w}_{t-1}) - \\ 2\ln G(\mathbf{r}|\mu) \to \min,$$

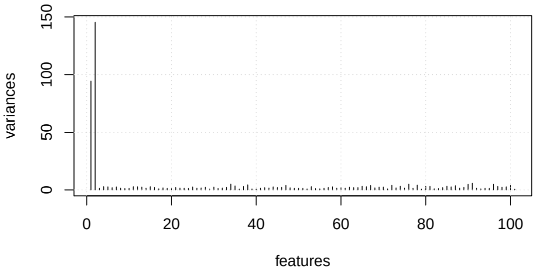
The summands are convex functions, and their differentiation $\partial/\partial(1/r_i)\left[...\right]=0$ yields simple formulas for the solution $\left(\hat{\mathbf{r}}|\mathbf{w}_1, ..., \mathbf{w}_T, \mu\right)$

$$\left(\hat{r}_i|\mathbf{w}_1, ..., \mathbf{w}_T, \mu\right) = \frac{\sum_{t=2}^{T}(w_{i,t})^2 + (1/\mu)}{T-1+(1/\mu)},\ i=1, ..., n.$$

- The synthetic data were generated by two normal distributions.
- Two informative features had been generated by two class-dependent normal distributions.
- 98 synthetic "redundant" features were added to this set. So each entity was characterized by $n = 100$ features, but only two of them were relevant to its class-membership.
- With time the centers of distributions rotated around the origin of coordinates in the two-dimensional feature space.
- We generated 100 consecutive data batches $(\mathbf{X}_t, \mathbf{Y}_t)$, $T = 100$ each containing 20 instances
- To compare the obtained results, we used the concept drift algorithms realized in the software environment Massive Online Analysis (MOA). A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer, *MOA: Massive Online Analysis* `http://sourceforge.net/projects/moa-datastream/`. Journal of Machine Learning Research (JMLR), 2010.

Experiments: "ground-truth" data

Variances



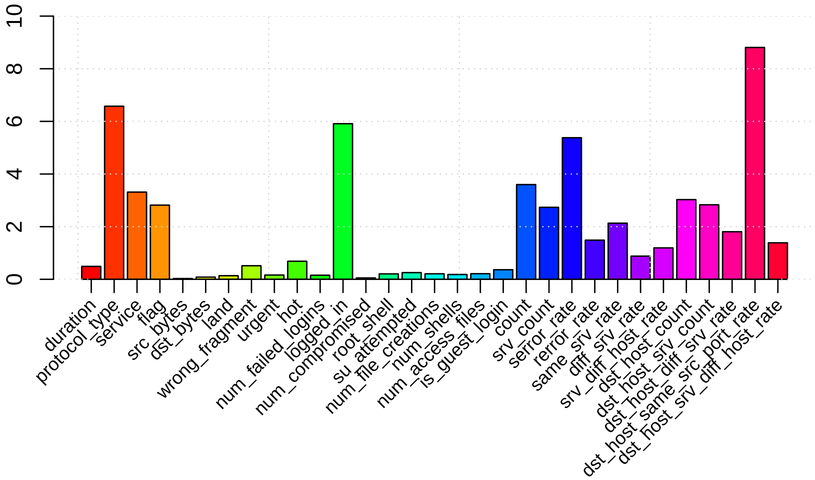
Algorithm	Classification error, %
OzaBagAdwin	12.45
SingleClassifierDrift	17.81
AdaHoeffdingOptionTree	7.22
DriftFeatureSelection	5.7

- The KDDCup'99 dataset (the Third International Knowledge Discovery and Data Mining Tools Competition) is a collection of TCP dumps taken over nine weeks in the framework of DARPA Intrusion Detection Evaluation Program in 1998.
- Each connection has 41 features and is labelled either as normal, or as an attack
- We solve the classification problem of attack detection
- This data set exists in two variants: full with about 5 millions records and its 10-percentage subset.In the current work, we used the 10-percentage set which was normalized and divided into about 10000 batches each containing 20 dumps.

- In the process of online classifier evaluation, before training on the next batch, we computed the error rate on it with the current decision rule. The total error was calculated as the average value of error rates in all the successive batches.
- For comparative some algoritms from the software Massive Online Analysis are used.

Algorithms	Classification error, %
OzaBagAdwin	6,144
SingleClassifierDrift	7,12
AdaHoeffdingOptionTree	1,056
DriftFeatureSelection	0,782

Experiments: real-world data
Variances



Thank you!

Questions?