Journal of Machine Learning Research 17 (2016) 1-33 Submitted 3/16; Revised 7/16; Published 8/16

In memory of Alexey Chervonenkis

Synergy of Monotonic Rules

Vladimir Vapnik VLADIMIR.VAPNIK@GMAIL.COM
Columbia University
New York, NY 10027, USA
Facebook AI Research
New York, NY 10017, USA

Rauf Izmailov RIZMAILOV@APPCOMSCI.COM
Applied Communication Sciences
Basking Ridge, NJ 07920-2021, USA

Editor: Andreas Christmann

Abstract

This article describes a method for constructing a special rule (we call it synergy rule) that uses as its input information the outputs (scores) of several monotonic rules which solve the same pattern recognition problem. As an example of scores of such monotonic rules we consider here scores of SVM classifiers.

In order to construct the optimal synergy rule, we estimate the conditional probability function based on the direct problem setting, which requires solving a Fredholm integral equation. Generally, solving a Fredholm equation is an ill-posed problem. However, in our model, we look for the solution of the equation in the set of monotonic and bounded functions, which makes the problem well-posed. This allows us to solve the equation accurately even with training data sets of limited size.

In order to construct a monotonic solution, we use the set of functions that belong to Reproducing Kernel Hilbert Space (RKHS) associated with the INK-spline kernel (splines with Infinite Numbers of Knots) of degree zero. The paper provides details of the methods for finding multidimensional conditional probability in a set of monotonic functions to obtain the corresponding synergy rules. We demonstrate effectiveness of such rules for
1) solving standard pattern recognition problems,
2) constructing multi-class classification rules,
3) constructing a method for knowledge transfer from multiple intelligent teachers in the LUPI paradigm.

Keywords: conditional probability, synergy, ensemble learning, intelligent teacher, privileged information, knowledge transfer, support vector machines, SVM+, classification, learning theory, kernel functions, regression

1. Introduction

The standard setting of pattern recognition problem requires, in the given set of functions $f(x, \alpha), \alpha \in \Lambda$ defined in the space $X \in R^n$, to find the function $f(x, \alpha_0)$ such that the indicator function $y = \theta(f(x, \alpha)) \in \{0, 1\}$ (in this paper, the indicator function is defined

as follows: $\theta(x) = 0$ for $x < 0$ and $\theta(x) = 1$ for $x \geq 0$) minimizes the loss functional

$$R(\alpha) = \int |y - f(x, \alpha)| dp(x, y)$$

if the probability measure $p(x, y),\ x \in X,\ y \in \{0, 1\}$ is unknown but iid data

$$(x_1, y_1), \ldots, (x_\ell, y_\ell), \quad x_i \in X,\ y_i \in \{0, 1\}$$

generated according to $p(x, y) = p(y|x)p(x)$ are given (in the standard pattern recognition terminology, conditional probability $P(y|x)$ defines an unknown law of classification given by Teacher and $P(x)$ defines an unknown generator of events that should be classified by the learning machine).

In this article, we illustrate our approach using Support Vector Machine (SVM) algorithms. The SVM algorithm construct an approximation of the desired classification function by first mapping vectors $x \in X$ into vectors $z \in Z$ and then constructing a separating hyperplane in space (Z, y). The obtained rule is used for classification of unknown iid vectors distributed according to the same unknown probability measure $p(x, y)$.

The conditional probability of class $y = 1$ given x depends on the position of vector z relative to the obtained hyperplane

$$s_i = (w, z_i) + b,$$

where w, b are the parameters estimated by SVM: if $s_i \geq 0$, vector x_i belongs to class $y_i = 1$, otherwise it belongs to the opposite class $y_i = 0$.

As Platt (1999) observed, the smaller is the (negative) score s_i for vector z_i, the closer is the conditional probability $P(y = 1|s_i)$ to zero and, the larger is the (positive) score s_i, the closer is the conditional probability $P(y = 1|s_i)$ to one. Platt introduced a method for mapping SVM scores into values of conditional probability based on two hypotheses, a general one and a special one.

> *The general hypothesis:* Conditional probability function $p(y = 1|s)$ is a monotonic function of variable s.
>
> *The special hypothesis:* Conditional probability function can be approximated well with sigmoid functions with two parameters:
>
> $$P(y = 1|s) = \frac{1}{1 + \exp\{-As + B\}}, \quad A, B \in R^1.$$
>
> Using the maximum likelihood technique, Platt (1999) introduced effective methods to estimate both parameters A, B (see Lin et al., 2007).

Platt's approach was shown to be useful for calibration of SVM scores. Nevertheless, this method has certain drawbacks: even if the conditional probability function for SVM is monotonically increasing, it does not necessarily have the form of a two-parametric sigmoid function.

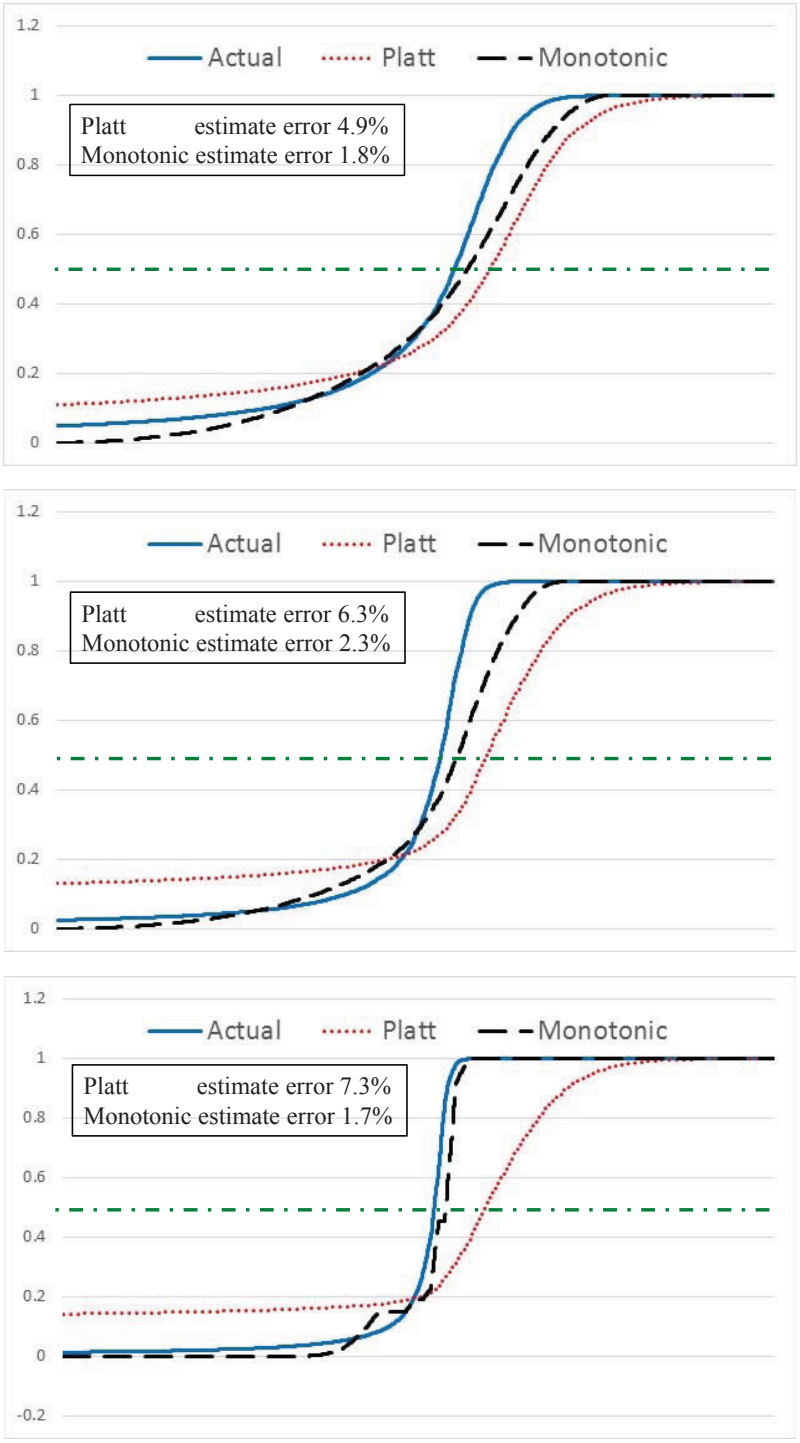

Figure 1: Comparison of conditional probability estimates.

It is easy to construct examples where suggested sigmoid function does not approximate well the desired monotonic conditional probability function. Figure 1 illustrates the conditional probability approximations (for a sample consisting of 96 random numbers that are evenly split between both classes) for Platt's approach and for the special one-dimensional case of the algorithm described further in this paper.

The one-dimensional problem mentioned in the previous paragraph has the following form: given pairs (values s_i of SVM scores and corresponding classifications y_i)

$$(s_1, y_1), \ldots, (s_\ell, y_\ell),$$

find an accurate approximation of the monotonic conditional probability function $p(y = 1|s)$. Section 3 describes a technique for construction of a monotonic approximation of the desired function. This approximation provides a more accurate estimate than the one based on sigmoid functions.

In this paper, we consider a more general (and more important) problem than this one-dimensional one. Suppose we have d different SVMs, solving the same classification problem. Also, suppose that the probability of class $y = 1$ given scores $s = (s^1, \ldots, s^d)$ of d SVMs is a multidimensional monotonic conditional probability function: for any coordinate k and any fixed values of the other coordinates $(s^1, \ldots, s^{k-1}, s^{k+1}, ...s^d)$, the higher is the value of score s^k, the higher is the probability $P(y = 1|s)$.

The goal of this article is to find a method for estimation of the *monotonic* conditional probability function $P(y = 1|s)$ for multidimensional vectors $s = (s^1, \ldots, s^d)$; that is, to combine, in a single probability value, the results of multiple (namely, d) SVMs. We show that estimating conditional probability function in a set of monotonic functions has a significant advantage over estimating conditional probability function in a general, non-monotonic set of functions: it forms a *well-posed* problem rather than an *ill-posed* problem.

The decision rule for a two-class pattern recognition problem can be obtained using the estimated conditional probability function $P(y = 1|s)$ as

$$y = \theta\left(P(y = 1|s) - \frac{1}{2}\right).$$

This article is organized as follows. In Section 2, we consider the problem of estimating conditional probability function. We show that the problem of conditional probability estimation in general sets of functions is *ill-posed*. However, this problem is *well-posed* for sets of nonnegative bounded (by 1) monotonic functions. Therefore, the problem of estimating the monotonic conditional probability function can be solved more accurately than the general problem of estimating conditional probability function. In Section 3, we describe methods of estimating monotonic conditional probability functions. In Section 4, we apply methods of estimating monotonic conditional probability function based on the scores generated by several different SVMs solving the same pattern recognition problem. Here we estimate monotonic conditional probability function of class $y = 1$ given all the scores and we obtain the so-called *synergy rule* of classification. In Section 5, we consider a method for knowledge transfer from multiple intelligent teachers.

Remark. It is important to note that, in classical machine learning literature, there are *ensemble methods* that combine several rules (see Dietterich (2000), Zhang and Ma (2012),

Tsybakov (2003), Lecué (2007)). The difference between ensemble rules and synergy rules is in the following:

1) Ensemble rule is a result of structural combination (such as voting or weighted aggregation) of several classification rules.

2) Synergy rule defines the *optimal* solution to the problem of combining several scores of *monotonic rules*. It is based on effective methods of conditional probability estimation in the set of monotonic functions.

3) Synergy rule is constructed only for *monotonic rules* (such as SVM) in contrast to ensemble rule which combines *any* rules. Synergy is the property of monotonicity of the solution.

2. Overview of Methods

In this section, we present a short overview of the direct constructive setting of estimation of conditional probability, as presented in (Vapnik and Izmailov, 2015c) and (Vapnik and Izmailov, 2015a). The method is quite general, and, in this section, we do not even assume that the probability has to belong to $[0, 1]$.

2.1 Glivenko-Cantelli Theory

In (Vapnik et al., 2015), (Vapnik and Izmailov, 2015c), (Vapnik and Izmailov, 2015a), we introduced direct constructive methods for solving the main problems of statistical inference. All these methods are based on Glivenko-Cantelli theory, which forms the foundation of classical statistics. This theory states that *the joint cumulative distribution function of several variables* $X = (X^1, \ldots, X^n)$

$$F(x) = P\{X^1 \le x^1, \ldots, X^n \le x^n\}$$

can be estimated from the observations

$$X_1, \ldots, X_\ell$$

by *empirical cumulative distribution function*

$$F_\ell(x) = \frac{1}{\ell} \sum_{i=1}^{\ell} \prod_{k=1}^{n} \theta(x^k - X_i^k), \tag{1}$$

where $\theta(x^k - X_i^k)$ is the step function (indicator function for $x \ge 0$).

Classical statistical theory provides bounds on the rate of convergence of $F_\ell(x)$ to the desired function $F(x)$ for the one-dimensional case (Massart, 1990):

$$P\{\sup_x |F_\ell(x) - F(x)| \ge \varepsilon\} \le 2 \exp\{-2\varepsilon^2 \ell\}. \tag{2}$$

Application of VC theory to n-dimensional case (Vapnik, 1998) gives the bound

$$P\{\sup_x |F_\ell(x) - F(x)| \ge \varepsilon\} \le \exp\left\{-\left(\varepsilon^2 - \frac{n \ln \ell}{\ell}\right) \ell\right\}. \tag{3}$$

2.2 Direct Setting of Conditional Probability Estimation

Estimation of cumulative distribution function using empirical data is a foundation for estimation of more sophisticated characteristics of stochastic events such as *density function, conditional density function, regression function, conditional probability function etc.*

1. We call function $p(x)$ *the density function* of the random events $X \sim F(x)$ if its integral defines the cumulative distribution function

$$\int \theta(x - X)p(X)dX = F(x), \quad X \in R^n.$$

2. Let pair $(x, y),\ x \in X,\ y \in Y \in R^1$ be a random event. We call

$$p(y|x) = \frac{p(x, y)}{p(x)},\ \ p(x) > 0,$$

the conditional density function; it defines the conditional density of the value of y given observation x.

3. Let pair $(x, y),\ x \in X,\ y \in \{0, 1\}$ be a random event. We call

$$p(y = 1|x) = \frac{p(x, y = 1)}{p(x)},\ \ p(x) > 0,$$

the conditional probability function; it defines conditional probability of $y = 1$ given observation x.

4 We call the integral

$$f(x) = \int yp(y|x)dy,$$

the regression function $f(x)$; it defines conditional expectation of value y given observation x.

The definition of conditional probability can be rewritten (see Vapnik and Izmailov (2015c), Vapnik and Izmailov (2015a)) in the form of the solution of integral equation

$$\int \theta(x - X)p(y = 1|X)dF(X) = F(x, y = 1). \tag{4}$$

These papers describe the direct constructive way of estimation of the conditional probability function as solving a multidimensional Fredholm integral equation (4) when cumulative distribution functions $F(x)$ and $F(x, y = 1)$ are unknown but data

$$(x_1, y_1), \ldots, (x_\ell, y_\ell)$$

are given. This setting is called *direct* because it is based on the definition of conditional probability. It is called *constructive* because there exists an *empirical cumulative distribution function*, defined as (1), that converges to the real cumulative distribution function with the rates (2) and (3).

In order to find the conditional probability (i.e., to solve equation (4)), we use the approximations $F_\ell(x)$ and $F_\ell(x, y = 1) = F_\ell(x|y = 1)P_\ell(y = 1)$ instead of unknown functions $F(x),\ F(x, y = 1)$. As shown in (Vapnik and Izmailov, 2015c) and (Vapnik and Izmailov, 2015a), statistical inference problems, such as (1) conditional density estimation, (2) conditional probability estimation, (3) regression estimation, (4) ratio of two densities estimation, can be formulated as follows: solve the integral equation

$$\int \theta(z - Z)f(z)dF(z) = (Ey)^{-1}F^*(z)$$

in the situation when the cumulative distribution functions $F(z),\ F^*(z)$, and the value $E(y)$ are unknown but their approximations in the form of empirical cumulative functions $F_\ell(z),\ F^*_\ell(z)$ and empirical average $P_\ell(y = 1)$ can be obtained using data

$$(z_1, y_1), \ldots, (z_\ell, y_\ell).$$

2.3 Fredholm Integral Equations of the First Kind

We consider the linear operator equations

$$Af = \Phi, \tag{5}$$

where A maps elements f of the metric space E_1 into elements Φ of the metric space E_2. Let A be a continuous one-to-one operator, which maps a set $\mathcal{M} \subset E_1$ onto a set $\mathcal{N} \subset E_2$, i.e., $A\mathcal{M} = \mathcal{N}$. The solution of such operator equation exists and is unique, i.e., inverse operator A^{-1} is defined:

$$\mathcal{M} = A^{-1}\mathcal{N}.$$

The crucial question is whether this inverse operator A^{-1} is continuous. If it is, then close functions in $\mathcal{N}$ are mapped by A^{-1} to close functions in $\mathcal{M}$; that is, a "small" change in the right-hand side of (5) results in a "small" change of its solution. In this case, the operator A^{-1} is called *stable* (Tikhonov and Arsenin, 1977). If, however, the inverse operator is discontinuous, then "small" changes in the right-hand side of (5) can cause a significant change of its solution. In this case, the operator A^{-1} is *unstable*.

The equation (5) is *well-posed* if its solution (1) exists, (2) is unique, and (3) is stable. Otherwise, the equation (5) is *ill-posed*.

We are interested in the situation when the solution of operator equation *exists*, and *is unique*. In this case, the *stability* of the operator A^{-1} determines whether (5) is ill-posed or well-posed. If the operator is unstable, then, generally speaking, any numerical solution of (5) is meaningless (a small error in the right-hand side of (5) can cause a large change of its solution).

Here we consider the linear integral operator

$$Af(x) = \int_a^b K(x, u)f(u)du$$

defined by the kernel $K(t, u)$, which is a symmetric positive definite function that is continuous almost everywhere on $a \le t \le b,\ \ c \le x \le d$. This kernel maps the set of functions

$\{f(t)\}$, continuous on $[a, b]$, unto the set of functions $\{\Phi(x)\}$, also continuous on $[c, d]$. The corresponding Fredholm equation of the first kind (Tikhonov and Arsenin, 1977)

$$\int_a^b K(x, u)f(u)du = \Phi(x)$$

requires finding the solution $f(u)$ given the right-hand side $\Phi(x)$. It is known that these integral equations are ill-posed.

In our problem, not only the right-hand side of (5) is an approximation but also the operator of (5) is defined approximately. In (Vapnik, 1995), such equations are called stochastic ill-posed problems.

2.4 Methods of Solving Ill-Posed Problems

In this subsection, we consider methods for solving ill-posed operator equations.

2.4.1 Inverse Operator Lemma

The following Inverse Operator Lemma (see Tikhonov and Arsenin, 1977) is the key enabler for solving ill-posed problems.

Lemma. *If A is a continuous one-to-one operator defined on a compact set $\mathcal{M}^* \subset \mathcal{M}$, then the inverse operator A^{-1} is continuous on the set $\mathcal{N}^* = A\mathcal{M}^*$.*

It is known that bounded monotonic functions form a compact set. Therefore, if we restrict the set of solutions of Fredholm integral equations to the class of bounded monotonic functions, we will make the corresponding equation well-posed. This is exactly the reason of our targeting the *monotonic* solutions in this paper.

Thus, as follows from Inverse Operator Lemma, the conditions of existence and uniqueness of the solution of an operator equation imply that the problem is well-posed on the compact $\mathcal{M}^*$. The third condition (stability of the solution) is automatically satisfied. This lemma is the basis for all constructive ideas of solving ill-posed problems. We describe one of them in the next subsection.

2.4.2 Regularization Method

Suppose that we have to solve the operator equation (4) defined by a continuous one-to-one operator A mapping $\mathcal{M}$ into $\mathcal{N}$, where we assume that the solution of (5) exists. Also, suppose that, instead of the right-hand side $\Phi(x)$, we are given its approximations $\Phi_\delta(x)$, where

$$\rho_{E_2}(\Phi(x), \Phi_\delta(x)) \leq \delta.$$

Our goal is to solve the equations

$$Af = \Phi_\delta$$

when $\delta \to 0$.

Consider a lower semi-continuous functional $W(f)$ (called the *regularizer*) that has the following three properties:

1. the solution of (5) belongs to the domain $D(W)$ of the functional $W(f)$;

2. the values $W(f)$ of W are non-negative in the domain of W;

3. the sets $\mathcal{M}_c = \{f : W(f) \leq c\}$ are compact for any $c \geq 0$.

The idea of regularization is to find a solution for (5) as an element minimizing the so-called regularized functional

$$R_\gamma(\hat{f}, \Phi_\delta) = \rho^2_{E_2}(A\hat{f}, \Phi_\delta) + \gamma_\delta W(\hat{f}), \qquad \hat{f} \in D(W) \tag{6}$$

with *regularization parameter* $\gamma_\delta > 0$.

The following theorem holds true (Tikhonov and Arsenin, 1977).

Theorem 1. *Let E_1 and E_2 be metric spaces, and suppose that, for $\Phi \in \mathcal{N}$, there exists a solution of (5) that belongs to compact $\mathcal{M}_c$ for some c. Suppose that, instead of the exact right-hand side Φ in (5), its approximations*[1] *$\Phi_\delta \in E_2$ are such that $\rho_{E_2}(\Phi, \Phi_\delta) \leq \delta$. Consider the sequence of parameters γ such that*

$$\gamma(\delta) \longrightarrow 0 \ \text{ for } \ \delta \longrightarrow 0, \ \text{ and } \ \lim_{\delta \longrightarrow 0} \frac{\delta^2}{\gamma(\delta)} \leq r < \infty. \tag{7}$$

Then the sequence of solutions $f_\delta^{\gamma(\delta)}$ minimizing the functionals $R_{\gamma(\delta)}(f, \Phi_\delta)$ on $D(W)$ converges to the exact solution f (in the metric of space E_1) as $\delta \longrightarrow 0$.

In a Hilbert space, the functional $W(f)$ may be chosen as $||f||^2$ for a linear operator A. Although the sets $\mathcal{M}_c$ are only weakly compact in this case, regularized solutions converge to the desired one. Such a choice of regularized functional is convenient since its domain $D(W)$ is the whole space E_1. In this case, however, the conditions on the parameters γ are more restrictive than in the case of Theorem 1: γ should converge to zero slower than δ^2.

Thus the following theorem holds true (Tikhonov and Arsenin, 1977).

Theorem 2. *Let E_1 be a Hilbert space and $W(f) = ||f||^2$. Then, if $\gamma(\delta)$ satisfies (7) with $r = 0$, the regularized elements $f_\delta^{\gamma(\delta)}$ converge to the exact solution f in E_1 as $\delta \to 0$.*

2.4.3 Stochastic Ill-Posed Problems

Let A_ℓ be approximations of A and Φ_ℓ be approximations of Φ. In order to solve stochastic ill-posed problems

$$A_\ell f = \Phi_\ell, \tag{8}$$

we will also use the regularization method minimizing the functional

$$T_\ell f = ||A_\ell f - \Phi_\ell||^2_{E_2} + \gamma_\ell \Omega(f). \tag{9}$$

Here, with increasing number of observations ℓ, functions Φ_ℓ converge to the actual function Φ and operator A_ℓ converges to the actual operators A in the sense that

$$||A_\ell - A||^2 = \sup_{f \in \{\Omega(f) \leq C\}} \frac{||Af - A_\ell f||^2}{\Omega(f)} \longrightarrow_{\ell \to \infty} 0. \tag{10}$$

As was shown in (Vapnik (1998)), if the desired solution belongs to one of the compacts $\{f;\ \Omega(f) \leq C\}$, the sequence of approximations Φ_ℓ of the right-hand side of equation and

1. The elements Φ_δ do not have to belong to the set $\mathcal{N}$.

the sequence of approximations A_ℓ of the operators converge in probability to, respectively, Φ and A, and $\gamma_\ell \to 0$ in (9) is such that

$$\lim_{\ell \to \infty} \frac{||A_\ell - A||^2}{\gamma_\ell} = 0,$$

then the sequence of minima converges to the desired function.

In (Vapnik and Izmailov (2015c)), (Vapnik and Izmailov (2015a)), it was shown that our specific integral equations satisfy the required conditions.

2.5 V-Matrix Method of Estimation of Conditional Probability Function

In order to find the conditional probability from the observations, we solve stochastic ill-posed problem (8) using regularization method (9), where approximations of the right-hand side of equation Φ and operator A are defined. We define two terms of (9) as:

1. the square of the distance $\rho^2(A_\ell f, \Phi_\ell)$ in space E_2 between functions (we omit the common normalizing multiplier $1/\ell$ in these definitions since it does not affect the subsequent derivations):

$$A_\ell f(x) = \sum_{i=1}^{\ell} \theta(x - X_i) f(X_i) \quad \text{and} \quad \Phi_\ell(x) = \sum_{j=1}^{\ell} y_j \theta(x - X_j),$$

where $(X_i, y_i), \ldots, i = 1, \ldots, \ell,\ X_i \in R^d,\ y_i \in \{0, 1\}$ are training data;

2. the regularization functional $\Omega(f)$, to be defined below.

2.5.1 Choice of Distance and Definition of V-Matrix

Below, we use L_2-distance in space E_2 in the general form

$$\rho^2(A_\ell f, \Phi_\ell) = \int (A_\ell f(x) - \Phi_\ell(x))^2 \sigma(x) d\mu(x),$$

where $\sigma(x)$ is a non-negative function and $\mu(x)$ is a probability measure; some choices for $\sigma(x)$ and $\mu(x)$ were considered in (Vapnik and Izmailov (2015c)), (Vapnik and Izmailov (2015a)). To simplify computations, we chose

$$\sigma(x) = \prod_{k=1}^{d} \sigma_k(x^k) \quad \text{and} \quad \mu(x) = \prod_{k=1}^{d} \mu_k(x^k), \quad \text{where } x = (x^1, \ldots, x^d).$$

Then we can rewrite the square of distance ρ^2 in the explicit form

$$\int \left(\sum_{i=1}^{\ell} f(X_i) \prod_{k=1}^{d} \theta(x^k - X_i^k) - \sum_{j=1}^{\ell} y_j \prod_{k=1}^{d} \theta(x^k - X_j^k) \right)^2 \prod_{k=1}^{d} \sigma_k(x^k) d\mu_k(x^k) =$$

$$\sum_{i,j=1}^{\ell} f(X_i) f(X_j) V_{i,j} - 2 \sum_{i,j=1}^{\ell} y_j f(X_i) V_{i,j} + \sum_{i,j=1}^{\ell} y_i y_j V_{i,j},$$

where we have denoted

$$V_{i,j} = \prod_{k=1}^{d} V_{i,j}^k, \quad V_{i,j}^k = \int \theta\left(x^k - \min\{X_i^k, X_j^k\}\right) \sigma_k(x^k) d\mu(x^k).$$

We denote by V the $(\ell \times \ell)$-dimensional matrix of elements V_{ij}, by $\mathbf{f}$ the ℓ-dimensional vector $\mathbf{f} = (f(X_1), \ldots, f(X_\ell))^T$, and by Y the ℓ-dimensional vector $Y = (y_1, \ldots, y_\ell)^T$. In matrix notations, we can rewrite the square of distance as follows:

$$\rho^2 = \mathbf{f}^T V \mathbf{f} - 2\mathbf{f}^T V Y + Y^T V Y.$$

2.5.2 Choice of Regularization Functional

Suppose that the solution of our integral equation (4) belongs to the RKHS (Reproducing Kernel Hilbert Space) associated with kernel $K(x, x^*)$ (symmetric positive definite function of vector variables $x, x^* \in X$). This means that RKHS has inner product such that for any function $f(x)$ from the space, the equality

$$(K(\cdot, y), f) = f(y)$$

holds true. According to Mercer theorem, any positive definite kernel $K(x, x^*)$ can be represented as

$$K(x, x^*) = \sum_{k}^{\infty} \lambda_k \phi_k(x) \phi_k(x^*),$$

where $\{\phi_k(x)\}$ is a system of orthonormal functions in E_1 and $\{\lambda_k\}$ is a sequence of non-negative values converging to zero, where $k = 1, 2, \ldots$.

It is easy to check that the functions

$$f(x, a) = \sum_{k=1}^{\infty} a_k \phi_k(x),$$

belong to RKHS associated with kernel $K(x, x^*)$ if the inner product between two functions $f(x, a)$ and $f(x, b)$ has the form

$$(f(x, a), f(x, b)) = \sum_{k=1}^{\infty} \frac{a_k b_k}{\lambda_k},$$

and, therefore, the norm of function $f(x, a)$ is

$$||f(x, a)||^2 = \sum_{k=1}^{\infty} \frac{a_k^2}{\lambda_k}. \tag{11}$$

We will chose the norm of function from RKHS as the regularizer $\Omega(f) = ||f||^2$. As follows from (11), the set of functions with their norm bounded by C

$$||f(x, a)||^2 = \sum_{k=1}^{\infty} \frac{a_k^2}{\lambda_k} \leq C$$

is a compact. Therefore, we use as a regularizer in (9) the norm of function in RKHS

$$\rho^2 + \gamma_\ell ||f||^2 = \mathbf{f}^T V \mathbf{f} - 2\mathbf{f}^T VY + Y^T VY + \gamma_\ell ||f||^2. \tag{12}$$

An important property of RKHS for applicatons is defined by the so-called Representer Theorem (Kimeldorf and Wahba (1970)), according to which the minimum of (12) has an expansion on elements $K(x_i, x)$ defined on the training data $x_1, \ldots, x_\ell$

$$f(x, a) = \sum_{i=1}^{\ell} \alpha_i K(x_i, x), \tag{13}$$

and the norm of function f in RKHS is defined as

$$||f||^2 = \sum_{i,j=1}^{\ell} \alpha_i \alpha_j K(x_i, x_j). \tag{14}$$

To simplify the notations, we introduce ℓ-dimensional vector $\Lambda = (\alpha_1, \ldots, \alpha_\ell)$, ℓ-dimensional vector functions $\mathcal{K}(x) = (K(x_1, x), \ldots, K(x_\ell, x))^T$ and $(\ell \times \ell)$-dimensional matrix $K = (K(x_i, x_j))$.

Using these notations, we can rewrite (13) and (14) as

$$f(x) = \mathcal{K}^T(x)\Lambda, \quad \mathbf{f} = K\Lambda, \quad ||f||^2 = \Lambda^T K \Lambda.$$

2.5.3 V-Matrix Kernel Regression

In order to solve our integral equation using the regularization technique, we have to minimize, with respect to vector Λ, the functional

$$W(\Lambda) = \Lambda^T KVK\Lambda - 2\Lambda^T KVY + \gamma_\ell \Lambda^T K \Lambda; \tag{15}$$

in this functional, the third term of (12) was omitted since it does not depend on Λ. The solution has the form

$$f(x) = \Lambda^T \mathcal{K}(x), \tag{16}$$

where one has to minimize functional (15) in order to find Λ.

The minimum of (15) has the closed-form representation

$$\Lambda = (VK + \gamma_\ell I)^{-1} VY. \tag{17}$$

Note that for $y_i \in \{0, 1\}$ in $Y = (y_1, .., y_\ell)^T$, expression (17) estimates the conditional probability; for $y_i \in R^1$, this expression estimates the regression.

2.6 Estimation in a Set of Functions with Bias Term

Below we consider sets of functions $\{f(x) + b\}$, where b is a value of bias (to be estimated from data) and function $f(x)$ belongs to RKHS (note that $f(x) + b$ does not have to belong to RKHS). Replacing $f(x)$ with $f(x) + b$ in (16), we can rewrite (15) in the form

$$W = (K\Lambda + b\mathbf{1}_\ell)^T V(K\Lambda + b\mathbf{1}_\ell) - 2(K\Lambda + b\mathbf{1}_\ell^T)VY + \gamma_\ell \Lambda^T K \Lambda. \tag{18}$$

Finding the expression for b by minimizing (18)

$$b = \frac{1}{\ell}\mathbf{1}_\ell^T (Y - K\Lambda) \tag{19}$$

and putting it into equation (18), we obtain the functional for minimization:

$$W = \left(K\Lambda + \frac{1}{\ell}\mathbf{1}_\ell\mathbf{1}_\ell^T(Y - K\Lambda)\right)^T V \left(K\Lambda + \frac{1}{\ell}\mathbf{1}_\ell\mathbf{1}_\ell^T(Y - K\Lambda)\right) - \tag{20}$$

$$2\left(K\Lambda + \frac{1}{\ell}\mathbf{1}_\ell\mathbf{1}_\ell^T(Y - K\Lambda)\right)^T VY + \gamma_\ell\Lambda^T K\Lambda.$$

In order to simplify this expression, we introduce the notations

$$E = I - \frac{1}{\ell}\mathbf{1}_\ell\mathbf{1}^\mathbf{T} \text{ and } \mathcal{V} = EVE.$$

Then

$$W = \Lambda^T K\mathcal{V}K\Lambda - 2\Lambda^T K\mathcal{V}Y + \gamma\Lambda^T K\Lambda + C, \tag{21}$$

where C are the terms that do not depend on Λ. Taking the derivative of W over Λ and equating it to zero, we see that, in order to minimize (21), vector Λ has to satisfy the equation

$$2K\mathcal{V}K\Lambda - 2K\mathcal{V}Y + 2\gamma K\Lambda = 0.$$

Solving this equation with respect to Λ, we obtain the closed-form solution

$$\Lambda = (\mathcal{V}K + \gamma I)^{-1}\mathcal{V}Y, \tag{22}$$

which differs from (17) just by using matrix $\mathcal{V}$ instead of matrix V.

Therefore, in order to find the conditional probability in the form $f(x) = \Lambda^T\mathcal{K}(x) + b$, we have to estimate the vector Λ using (22) and estimate the bias b using (19).

Remark. The described solution for conditional probability is also applicable for estimating regression. In that case, coordinates y_i of vector $Y = (y_1, \ldots, y_\ell)^T$ belong to R^1 and the set of functions $f(x, \alpha), \alpha \in \Lambda$ is a set of real-valued functions from RKHS.

2.7 Indirect Methods of Estimation of Conditional Probability

In addition to direct setting of conditional probability problem, indirect settings also exist. They are based on the fact that for some loss functions $\rho(y - f(x, \alpha))$, under a wide range of conditions, the minimum of the functional

$$R = \int \rho(y - f(x, \alpha))dp(x, y) \tag{23}$$

in the set $f(x, \alpha),\ \alpha \in \Lambda$ defines conditional probability function $f(x, \alpha_0)$ (provided that $\alpha_0 \in \Lambda$). In order to estimate the conditional probability, one has to find the function that minimizes functional (23) if the probability measure $Pp(x, y)$ is unknown but iid sample

$$(x_1, y_1), \ldots, (x_\ell, y_\ell) \tag{24}$$

is given. The standard idea for solving this problem is to minimize the functional

$$\sum_{i=1}^{\ell} \rho(y_i - f(x_i, \alpha)) + \gamma ||f(x, \alpha)||^2.$$

There are two classical ideas of choosing the term $\rho(y - f(x, \alpha))$:

1. $\rho(y - f(x, \alpha)) = (y - f(x, \alpha))^2$, which leads to regularized *kernel least square* method.

2. $\rho(y - f(x, \alpha)) = |y - f(x, \alpha)|$, which leads to a more robust regularized *kernel least modulo* method.

2.7.1 Regularized Kernel Least Square Method

We minimize the functional (23) based on empirical data (24) in the set of functions belonging to RKHS associated with the kernel $K(x, x^*)$. For this set, we minimize the empirical functional

$$\sum_{i=1}^{\ell} (y_i - f(x_i, \alpha) - b)^2 + \gamma ||f(x, \alpha)||^2.$$

Minimizing this expression over b, we obtain

$$\sum_{i=1}^{\ell} (f(x_i, \alpha) + b) = \sum_{i=1}^{\ell} y_i,$$

where we again assume that functions $f(x, \alpha), \alpha \in \Lambda$ belong to RKHS associated with kernel $K(x, x^*)$. Using the same reasoning as in the previous section, we obtain that the solution has the form (16), with the expansion coefficients $\Lambda = (\alpha_1, \ldots, \alpha_\ell)^T$ maximizing the functional

$$W = \Lambda^T K \mathcal{I} K \Lambda - 2 \Lambda^T K \mathcal{I} Y + \gamma \Lambda^T K \Lambda,$$

where we have denoted

$$\mathcal{I} = EIE.$$

The vector of coefficients Λ in closed form is

$$\Lambda = (\mathcal{I} K + \gamma I)^{-1} \mathcal{I} Y.$$

2.7.2 Regularized Kernel Least Modulo Method

In classical statistics, besides L_2-norm loss function for estimating regression, L_1-norm loss is considered as well. In many situations, L_1-norm regression has an advantage over L_2-norm: it provides the so-called *robust regression* (Andersen (2008)). As in previous sections, we estimate the regression in the set of functions $\{f(x, \alpha) + b\}$, where each $f(x, \alpha)$ belongs to RKHS associated with kernel $K(x, x^*)$. In order to do that, we minimize the functional

$$R = C \sum_{i=1}^{\ell} |y_i - f(x_i, \alpha) - b| + ||f(x, \alpha)||^2.$$

We rewrite this problem in an equivalent form: we map vectors $x \in X$ into Hilbert space $z \in Z$ defined by the inner product $(z_i, z_j) = K(x_i, x_j)$ given by a non-negative definite kernel $K(x, x_*)$. We look for a solution in the form $f(x, \alpha) = (w, z) + b$, where $w, z \in Z$. In these notations, we rewrite our minimization problem as follows: minimize the functional

$$R = C \sum_{i=1}^{\ell} \xi_i + (w, w)$$

subject to the constraints

$$-\xi_i \leq y_i - (w, z_i) - b \leq \xi_i, \quad i = 1, \ldots, \ell.$$

Using Lagrange multiplier method, we construct the Lagrangian

$$\mathcal{L} = C \sum_{i=1}^{\ell} \xi_i + (w, w) - \sum_{i=1}^{\ell} \alpha_i [(y_i - (w, z_i) - b) + \xi_i] - \sum_{i=1}^{\ell} \alpha_i^* [(-y_i + (w, z_i) + b) + \xi_i],$$

the saddle point of which (minimum with respect to ξ and w and maximum with respect to α) defines the solution.

The solution has the form

$$f(x, \alpha) = \sum_{i=1}^{\ell} \delta_i K(x_i, x) + b,$$

where, in order to find $\delta_i = \alpha_i^* - \alpha_i$, one has to maximize the functional

$$R = \sum_{i=1}^{\ell} y_i \delta_i - \frac{1}{2} \sum_{i,j=1}^{\ell} \delta_i \delta_j K(x_i, x)$$

subject to the constraints

$$-C \leq \delta_i \leq C, \qquad \sum_{i=1}^{\ell} \delta_i = 0.$$

The bias b can be computed as

$$b = y_k - \sum_{i=1}^{\ell} \delta_i K(x_i, x_k),$$

where k is an index for which $|\delta_k| \neq C$.

3. Estimation of Monotonic Conditional Probability Functions

Our goal is to minimize functional (15) in the set of monotonically increasing functions. We do this by using expansion of desired function on kernels that generate splines with infinite number of knots (INK-spline) of degree zero. The reason we use these kernels is that they enable an efficient and straightforward construction of multidimensional monotonic functions; it is possible that some other kernels might be used for that purpose as well.

3.1 Kernels for Estimating INK-Splines

According to the definition in the one-dimensional case, splines of degree r with m knots are defined by the expansion (in this section, we assume that $0 \le x \le 1$)

$$S(x|r,m) = \sum_{s=0}^{r} c_s x^s + \sum_{k=0}^{m} e_k (x - a_k)_+^r, \tag{25}$$

where

$$(x - a_k)_+^r = \begin{cases} (x - a_k)^r & \text{if } x - a_k \ge 0 \\ 0 & \text{otherwise} \end{cases}$$

We generalize expansion (25) using infinite number of knots:

$$S_\infty(x) = \sum_{s=0}^{r} c_s x^s + \int_0^\infty g(\tau)(x - \tau)_+^r d\tau.$$

Following the approach from (Vapnik (1998)), (Izmailov et al. (2013)), we define the kernel with infinite number of knots (INK-spline) of degree r for expansion of the function of one variable $x \ge 0$ in the form

$$K_r(x_i, x_j) = \int_0^\infty (x_i - \tau)_+^r (x_j - \tau)_+^r \, d\tau = \sum_{k=0}^{r} \frac{C_r^k}{2r - k + 1} [\min\{x_i, x_j\}]^{2d-k+1} |x_i - x_j|^k$$

(here we modified the definition of INK-kernel from (Vapnik (1998)), (Izmailov et al. (2013)) by omitting its polynomial portion).

For $r = 0$, the INK-spline kernel has the form

$$K_0(x_i, x_j) = \min\{x_i, x_j\}; \tag{26}$$

for $r = 1$, the INK-spline kernel has the form

$$K_1(x_i, x_j) = \frac{1}{3} (\min\{x_i, x_j\})^3 + \frac{1}{2} (\min\{x_i, x_j\})^2 |x_i - x_j|.$$

In the multidimensional case, the INK-spline of degree r is defined as

$$K_r(x_i, x_j) = \prod_{k=1}^{d} K_r(x_i^k, x_j^k), \quad x = (x^1, \ldots, x^d).$$

3.2 Estimating One-Dimensional Monotonic Conditional Probability Function

In classical statistics, there are methods for estimation of monotonic (isotonic) regression (see Best and Chakravarti, Mair et al. (2009), Sysoev et al. (2011), Meyer (2013)), focusing on maintaining the monotonicity on the observed sample points. Below we describe a method of estimating conditional probability that is a monotonic function in the whole space; the method is based on INK-splines with infinite number of knots.

We estimate the monotonic conditional probability function in the set of INK-splines of degree zero (piecewise constant spline function with infinite number of knots). We start with one-dimensional case where $x \geq 0$. For this kernel, the solution is defined as

$$f(x) = \Lambda^t \mathcal{K}(x) + b = \sum_{i=1}^{\ell} \alpha_i \min\{x_i, x\} + b. \tag{27}$$

To specify monotonically increasing function, we impose additional constraints for (27): specifically, we consider the subset of functions (27) for which the inequality

$$\frac{df(x,\alpha)}{dx} \geq 0, \quad \forall x \geq 0 \tag{28}$$

is valid. Since any function (27) is a piecewise linear continuous function, in order for it to be monotonic, it is sufficient for that function to satisfy the constraints

$$\begin{aligned} \frac{df(x_j,\alpha)}{dx} &= \sum_{i=1}^{\ell} \alpha_i \theta(x_i - x_j) \geq 0, \quad j = 1, \ldots, \ell, \\ \frac{df(0,\alpha)}{dx} &= \sum_{i=1}^{\ell} \alpha_i \geq 0. \end{aligned} \tag{29}$$

Indeed, consider three possible cases:

1. Let $x \leq \min\{x_1, \ldots, x_L\}$. Then, since $\theta(x_i - x) = 1$ for all $i = 1, \ldots, L$, the value (28) is non-negative according to the second inequality in (29).

2. Let $\min\{x_1, \ldots, x_L\} < x < \max\{x_1, \ldots, x_L\}$. Without loss of generality, assume the ordering $x_1 \leq x_2 \leq \ldots x_L$ and the position of x within that ordering as $x_j \leq x \leq x_{j+1}$. The function (28) is linear on the interval (x_j, x_{j+1}) and its values at the ends of the interval are
$$\sum_{i=1}^{L} \alpha_i \theta(x_i - x_j) \quad \text{and} \quad \sum_{i=1}^{L} \alpha_i \theta(x_i - x_{j+1}),$$
which are non-negative, according to the first inequality in (29). Therefore, the function is also non-negative at any internal point x of the interval (x_j, x_{j+1}).

3. Let $x \geq \max\{x_1, \ldots, x_L\}$. Then, since $\theta(x_i - x) = 0$ for all $i = 1, \ldots, L$, the value (28) is zero.

We introduce the notations

$$\Theta(x_j) = (\theta(x_1 - x_j), \ldots, \theta(x_\ell - x_j))^T, \quad j = 1, \ldots, \ell.$$

Using these notations, we rewrite the constraints (29) in the form

$$\Lambda^T \Theta(0) \geq 0, \quad \Lambda^T \Theta(x_j) \geq 0, \quad j = 1, \ldots, \ell. \tag{30}$$

3.2.1 Estimating Monotonic Conditional Probability Using V-Matrix Method

In order to find a monotonic solution, we use our method for estimating conditional probability function with INK-spline kernel of degree zero with additional ℓ monotonicity constraints (30). That is, we have to minimize the functional

$$W = (K\Lambda + b\mathbf{1}_\ell)^T V(K\Lambda + b\mathbf{1}_\ell) - 2(K\Lambda + b\mathbf{1}_\ell)^T VY + \gamma_\ell \Lambda^T K\Lambda$$

(here coordinates of vector Y are $y_i \in \{0, 1\}$ subject to $\ell + 1$ inequality constraints

$$\Lambda^T \Theta(0) \geq 0, \quad \Lambda^T \Theta(x_j) \geq 0, \quad j = 1, \ldots, \ell. \tag{31}$$

Let $x \geq 0$. Then, in order to construct the conditional probability in the set of nonnegative monotonic functions bounded by the value 1, we have to enforce the constraint $P(y = 1|x) \leq 1$. Thus, taking into account nonnegativity and monotonicity (31), we add the constraint

$$\Lambda^T \mathcal{K}(x = 1) + b = \Lambda^T \vec{\mathbf{x}} + b \leq 1, \tag{32}$$

where $\vec{\mathbf{x}} = (x_1, ...x_\ell)^T$.

3.2.2 Estimating Monotonic Conditional Probability Using L_2-Norm SVM

Using L_2-norm SVM for estimating monotonic conditional probability function, we minimize the functional

$$W(\Lambda) = \Lambda^T KK\Lambda - 2\Lambda^T KY + \gamma_\ell \Lambda^T K\Lambda,$$

with coordinates of Y are $y_i \in \{0, 1\}$ subject to $\ell + 2$ inequality constraints (31), (32).

3.2.3 Estimating Monotonic Conditional Probability Using L_1-Norm SVM

We look for a solution of the following quadratic optimization problem: minimize the functional

$$C \sum_{i=1}^{\ell} |y_i - f(x_i, \alpha) - b| + ||f(x, \alpha)||^2$$

subject to $\ell + 2$ inequality constraints (31) and (32), where $f(x, \alpha)$ belongs to RKHS associated with the kernel INK-spline of degree zero $K(x_i, x_j) = \min(x_i, x_j)$.

In matrix form, this problem can be rewritten as follows: minimize the functional

$$R(\xi, \Lambda) = C\mathbf{1}_\ell^T \vec{\xi} + \gamma \Lambda^T K\Lambda$$

subject to the constraints

$$-\vec{\xi} \leq Y - K\Lambda - b\mathbf{1}_\ell \leq \vec{\xi},$$

and $\ell + 2$ constraints

$$\Lambda^T \Theta(0) \geq 0, \quad \Lambda^T \Theta(x_i) \geq 0, \quad i = 1, \ldots, \ell,$$

$$\Lambda^T \vec{\mathbf{x}} + b \leq 1,$$

where we have denoted

$$\vec{\xi} = (\xi_1, \ldots, \xi_\ell)^T, \quad \vec{x} = (x_1, \ldots, x_\ell)^T, \quad Y = (y_1, \ldots, y_\ell)^T.$$

3.3 Estimating Multidimensional Monotonic Conditional Probability Functions

3.3.1 Estimation of Monotonic Functions from RKHS Associated with Multiplicative Kernels

In multidimensional case, $x_i = (x_i^1, \ldots, x_i^d)^T \in H \subset R^d$, where we can assume (by proper normalization) that $H = [0,1]^d$. We consider the solution of the equation in the form

$$f(x) = \sum_{i=1}^{\ell} \alpha_i K(x_i, x) + b,$$

where the kernel generating d-dimensional INK-spline of degree zero has the multiplicative form

$$K(x_i, x) = \prod_{k=1}^{d} \min(x_i^k, x^k).$$

We have

$$f(x) = \sum_{i=1}^{\ell} \alpha_i \prod_{k=1}^{d} \min(x_i^k, x^k) + b. \tag{33}$$

Note that the set of functions (33) is monotonic in H if d inequalities

$$\frac{df(x)}{dx^k} = \sum_{i=1}^{\ell} \alpha_i \theta(x_i^k - x^k) \prod_{m \neq k} \min(x_i^m, x^m) \geq 0, \quad k = 1, \ldots, d \tag{34}$$

hold true for an function and any $x = (x^1, \ldots, x^d) \in H$. To keep the matrix notations, we consider diagonal matrix D_k with diagonal elements h_{ii}

$$h_{ii} = \prod_{m \neq k} \min(x_i^m, x^m), \ i = 1, \ldots, \ell, \ k = 1, \ldots, d.$$

Using this notation, we rewrite (34) as

$$\frac{df(x)}{dx^k} = \Lambda^T D_k \Theta(x^k) \geq 0, \quad k = 1, \ldots, d \tag{35}$$

for all $x \in H$.

In order to find the monotonic conditional probability function, we minimize

$$R(\Lambda, b) = (K\Lambda + b\mathbf{1}_\ell)^T V (K\Lambda + b\mathbf{1}_\ell) - 2 (K\Lambda + b\mathbf{1}_\ell)^T VY + \gamma(\Lambda^T K \Lambda) \tag{36}$$

subject to d constrains

$$\inf_{x \in H} \Lambda^T D_k \Theta(x^k) \geq 0, \quad k = 1, \ldots, d.$$

This is a difficult problem to solve. Instead, one could construct an approximate solution by using Monte Carlo ideas: consider $N \approx n^d$ random (or pseudo-random) elements $x_t =$

$(x_t^1, \ldots, x_t^d)^T, t = 1, \ldots, N$ belonging to H (Sobol points (Jäckel (2004)) could be used, for instance) and instead of d constraints (35) consider Nd constraints

$$\Lambda^T D_k \Theta(x_t^k) \geq 0, \quad k = 1, \ldots, d, \ t = 1, \ldots, N. \tag{37}$$

As in the one-dimensional case, in order to enforce that the value of conditional probability does not exceed one, we add one more constraint

$$\Lambda^T \vec{\mathbf{x}^*} + b \leq 1, \tag{38}$$

where we have denoted by $\vec{\mathbf{x}^*}$ the ℓ-dimensional vector of products

$$\vec{\mathbf{x}^*} = \left[\left(\prod_k^d x_1^k \right), \ldots, \left(\prod_k^d x_\ell^k \right) \right]^T.$$

Therefore, in order to construct d-dimensional *approximation* of monotonic conditional probability function, one has to minimize functional (36), subject to Nd constraints (37) and one constraint (38).

3.3.2 Estimating Monotonic Functions from RKHS Associated with Additive Kernels

Along with functions defined by (multiplicative) INK-spline kernels of degree zero (26) that can construct approximations to monotonic functions, we consider functions defined by the additive kernel (which is a sum of one-dimensional kernels)

$$f(x) = \sum_{i=1}^{\ell} \sum_{k=1}^{d} \alpha_i^k \min(x_i^k, x^k) + b. \tag{39}$$

In order to find $d \times \ell$ coefficients $\alpha_i^k, \ k = 1, \ldots, d, \ i = 1, \ldots, \ell$ of expansion in estimating conditional probability function in the direct setting, we minimize the functional

$$R(\Lambda_1, \ldots \Lambda_d, b) = \left[\sum_{k=1}^{d} K_k \Lambda_k + b \mathbf{1}_\ell \right]^T V \left[\sum_{k=1}^{d} K_k \Lambda_k + b \mathbf{1}_\ell \right] - \tag{40}$$

$$2 \left[\sum_{k=1}^{d} K_k \Lambda_k + b \mathbf{1}_\ell \right]^T VY + \gamma \sum_{k=1}^{d} (\Lambda_k^T K_k \Lambda_k)$$

subject to $d \times (\ell + 1)$ inequality constraints

$$\begin{aligned} \frac{\partial f(x_j, \alpha)}{\partial x^k} &= \sum_{i=1}^{\ell} \alpha_i^k \theta(x_i^k - x_j^k) = \Lambda_k^T \Theta(x_j^k) \geq 0, \quad j = 1, \ldots, \ell, \quad k = 1, \ldots, d, \\ \frac{\partial f(\vec{0}, \alpha)}{\partial x^k} &= \sum_{i=1}^{\ell} \alpha_i^k = \Lambda_k^T \Theta(\vec{0}^k) \geq 0, \quad k = 1, \ldots, d, \end{aligned} \tag{41}$$

where we have denoted by Λ_k the ℓ-dimensional vector of $\alpha^k = (\alpha_1^k, \ldots, \alpha_\ell^k)^T,\ k = 1, \ldots, d$, by K_k the $(\ell \times \ell)$-dimensional matrix of elements $K_k(x_i^k, x_j^k) = \min(x_i^k, x_j^k)$, and by

$$\Theta(x_j^k) = (\theta(x_1^k - x_j^k), \ldots, \theta(x_\ell^k - x_j^k))^T, \quad j = 1, \ldots, \ell, \quad k = 1, \ldots, d$$

we have denoted the $d \times \ell$ vectors of dimensionality ℓ.

Let vector $x = (x^1, \ldots, x^d)$ have bounded coordinates $0 \le x^k \le c_k,\ k = 1, \ldots, , d$. Since conditional probability does not exceed 1, we need one more constraint $P(y = 1|c_1, \ldots, c_d) \le 1$. That is, we have to add the constraint

$$\sum_{k=1}^{d} \Lambda_k^T \mathbf{X}^k + b \le 1, \tag{42}$$

where we have denoted $\mathbf{X}^k = (x_1^k, \ldots, x_\ell^k)^T$. A function satisfying the conditions (41) and (42), is monotonic (it can be proven in the same way it was done in Section 3.2).

3.3.3 Estimation of Multidimensional Monotonic Conditional Probability Using L_2-Norm SVM

In order to estimate the conditional probability in indirect setting, one minimizes functional (36), subject to constraints (37), (38) for multiplicative kernel (or functional (40) subject to constraints (41), (42) for additive kernel), where V-matrix is replaced with identity matrix (I-matrix).

3.3.4 Estimation of Multidimensional Monotonic Conditional Probability Using L_1-Norm SVM

In order to estimate multidimensional conditional monotonic function using L_1-norm and multiplicative INK-spline kernel (33), one minimizes the functional

$$R(\xi, \Lambda) = C\mathbf{1}_\ell^T \vec{\xi} + \gamma \Lambda^T K \Lambda$$

subject to the constraints

$$-\vec{\xi} \le Y - K\Lambda - b\mathbf{1}_\ell \le \vec{\xi},$$

and constraints (37), (38). Here we used the notations

$$\vec{\xi} = (\xi_1, \ldots, \xi_\ell)^T, \quad Y = (y_1, \ldots, y_\ell)^T.$$

To estimate multidimensional conditional monotonic function using L_1-norm and additive INK-spline kernel (39), one minimizes the functional

$$R(\xi, \Lambda) = C\mathbf{1}_\ell^T \vec{\xi} + \gamma \left[\sum_{k=1}^{d} \Lambda_k^T K_k \Lambda_k \right]$$

subject to the constraints

$$-\vec{\xi} \le Y - \sum_{k=1}^{d} K_k \Lambda_k - b\mathbf{1}_\ell \le \vec{\xi},$$

and constraints (41), (42).

3.3.5 Computational Issues

Quadratic optimization problem. In order to estimate a multidimensional monotonic function using multiplicative kernel, one has to solve a quadratic optimization problem of order ℓ subject to $N = \ell d$ inequality constraints.

With additive kernel, one has to estimate $d \times (\ell + 1)$ parameters under $d \times (\ell + 1)$ constraints. To decrease the computation amount:

1. One can replace V-matrix with I-matrix.

2. For additive kernel, one can estimate multidimensional conditional probability function in the *restricted set of functions* where $\alpha_i^t = \alpha_i$, for some or for all t.

3. One can consider linear structure of the solution using d one-dimensional estimates of conditional probability $P(y = 1|s^t)$ obtained by solving one-dimensional estimation problems as described in Section 3.2.1, and then approximate the multidimensional conditional probability function as

$$P(y = 1|s^1, \ldots, s^d) = \sum_{t=1}^{d} \beta_t P(y = 1|s^t),$$

where its weights $\beta_t \geq 0$, $\sum \beta_i = 1$ are computed by solving an d-dimensional quadratic optimization problem under $d + 1$ constraints. That optimization problem is formulated as follows: minimize the functional

$$B^T PVPB - 2B^T PVY + \gamma B^T B$$

subject to the constraints

$$B \geq 0, \;\; B^T \mathbf{1}_\ell = 1,$$

where we have denoted by B vector of coefficients $B = (\beta_1, \ldots, \beta_d)^T$, by P the $(d \times \ell)$-dimensional matrix $P = p(x_i^t), \; t = 1, \ldots, d, \; i = 1, ..., \ell$.

Estimation of both the SVMs and the conditional probability using the same data set. In the examples considered in this section, we construct synergy rules for SVMs where we use the same training set both for constructing SVM rules $s_k = f_t(x), \; t = 1, \ldots, d$ and for estimating the conditional probability $P(y = 1|s_1, \ldots, s_d)$.

Suppose that our rules were constructed using different SVM kernels $K_t(x, y)$ and the same training set

$$(x_1, y_1), \ldots, (x_\ell, y_\ell) \tag{43}$$

and let

$$s_1^t, \ldots, s_\ell^t, \quad t = 1, \ldots, d$$

be the scores $s^t = f_t(x)$ obtained using vectors x from (43).

Note that these scores are statistically different from the scores obtained using ℓ elements of test set (support vectors s^* are biased: in the separable case, all $|s^*| = 1$). Therefore, it is reasonable to use scores obtained in the procedure of k-fold cross-validation for estimating parameters of SVM algorithm.

Also, note that while individual components of the same d-dimensional vector $S^t = (s_1^t, \ldots, s_d^t)$ are interdependent, the vectors S^t themselves are not (they are i.i.d), so the general theory developed in the previous sections is applicable here for computing conditional probabilities.

4. Synergy of Several SVMs

In this section, we consider several examples of synergy of d SVM rules obtained under different circumstances:

1. Synergy of d rules obtained using the same training data but different kernels.
2. Synergy of d rulse obtained using different training data but the same kernel.
3. Synergy of d classes classification problem using d *one versus the rest* rules.

In all these examples, the synergy of the rules is based on estimating the corresponding monotonic conditional probability function from RKHS associated with additive kernel, as described in Section 3.3.2.

4.1 Synergy of SVM Rules with Different Kernels

In this section, we show that the accuracy of classification using synergy of SVM rules that use different kernels can be much higher than the accuracy of a rule based on any kernel.[2] The effect of synergy, which is estimated by the number of additional training examples in training data required to achieve comparable to synergy level of accuracy, can be significant.

We selected the following 9 calibration data sets from UCI Machine Learning Repository (Lichman (2013)): Covertype, Adult, Tic-tac-toe, Diabetes, Australian, Spambase, MONK's-1, MONK's-2, and Bank marketing. Our selection of these specific data sets was driven by the desire to ensure statistical reliability of targeted estimates, which translated into availability of relatively large test data set (containing at least 150 samples). Specific breakdowns for the corresponding training and test sets are listed in Table 1.

For each of these 9 data sets, we constructed 10 random realizations of training and test data sets; for each of these 10 realizations, we trained three SVMs with different kernels: with RBF kernel, with INK-Spline kernel, and with linear kernel. The averaged test errors of the constructed SVMs are listed in Table 1.

Constructed SVMs provide binary classifications y and scores s. Additional performance improvements are possible by intelligent leveraging of the results of these classifications.

We compared our approach with the baseline method of voting on classification results of all three classifications obtained from three different kernels (since we had odd number of kernels, we did not need any tie-breaking in that vote). The first column of Table 2 shows the averaged test errors of that voting approach.

The second column of Table 2 shows the averaged test errors of our synergy approach. Specifically, the data in the second column are based on constructing a 3-dimensional mono-

2. The idea of using several SVMs as ensemble SVM (such as (Wang et al. (2009)) and Stork et al. (2013)) was used in the past for providing improved classification performance; however, these approaches did not leverage the main monotonicity property of SVM.

Data set	Training	Test	Features
Covertype	300	3000	54
Adult	300	26147	123
Tic-tac-toe	300	658	27
Diabetes	576	192	8
Australian	517	173	14
Spambase	300	4301	57
MONK's-1	124	432	6
MONK's-2	169	432	6
Bank	300	4221	16

Table 1: Calibration data sets from UCI Machine Learning repository.

Data set	Voting	Synergy	Gain
Covertype	27.83%	28.96%	-4.05%
Adult	20.07%	19.08%	4.93%
Tic-tac-toe	1.95%	1.75%	10.16%
Diabetes	24.53%	23.39%	4.67%
Australian	12.02%	12.54%	-4.33%
Spambase	8.96%	8.44%	5.80%
MONK's-1	22.80%	20.16%	11.57%
MONK's-2	19.31%	16.23%	15.95%
Bank	12.79%	11.73%	8.29%

Table 2: Synergy of SVMs with RBF, INK-spline, and linear kernels.

tonic conditional probability function from RKHS associated with additive kernel, as described in Section 3.3.2, on triples of SVM scores s. In this column, we assigned the classification labels y based on the sign of the difference between 3-dimensional conditional probability and the threshold value 1/2.

The last column of Table 2 contains relative performance gain (i.e., relative decrease of error rate) delivered by the proposed synergy approach over the benchmark voting algorithm.

The results demonstrate the consistent performance advantage of synergy approach over its empirical alternative in most of the cases (for 7 data sets out of 9); for some data sets this advantage is relatively small, but for others it is substantial (in relative terms).

This substantial performance improvement of synergy can be also viewed as a viable alternative to brute force approaches relying on accumulation of (big) data. Indeed, for the already considered Adult data set, we compared results of our synergy approach on a training data set consisting of 300 samples to an alternative approach relying on training SVM algorithms on larger training data sets. Specifically, we trained SVMs with RBF kernel and INK-Spline kernel on Adult data sets containing 1,000 and 3,000 samples. The results, shown in Table 3, suggest that synergy of two rules, even on training data set of

Training size	300	1000	3000
RBF	20.95%	19.21%	18.49%
INK-Spline	19.77%	18.72%	18.38%
Synergy	17.92%	-	-

Table 3: Synergy versus training size increase.

limited size, can be better that straightforward SVMs on training data sets of much larger sizes (in this example, equivalent to the increase of training sample by more than a factor of 10).

4.2 Synergy of SVM Rules Obtained on Different Training Data

Suppose we are dealing with "big data" situation, where the number L of elements in the training data set

$$(x_1, y_1), ..., (x_L, y_L), \tag{44}$$

is large. Consider the SVM method that uses a universal kernel[3]. Generally speaking, with the increase of size ℓ of training data, the expected error rate of the obtained SVM rule monotonically converges to the Bayesian rule (here the expectation is taken both over the rules obtained from different training data of the same size ℓ and over test data). The typical *learning curve* shows the dependence of that expected error rate on the size ℓ of training data as a hyperbola-looking curve consisting of two parts: the beginning of the curve, where the error rate falls steeply with the increase of ℓ, and the tail of the curve, where the error rate slowly converges to the Bayesian solution. Suppose that the transition from the "steeply falling" part of the curve to the "slowly decreasing" part of the curve (sometimes referred to as the "knee" of the curve) occurs for some ℓ^*. Assuming that large number L in (44) is greater than ℓ^*, we partition the training data (44) into J subsets containing ℓ elements each (here $L = J\ell$ and $\ell > \ell^*$ as well):

$$(x_{(t-1)\ell+1}, y_{(t-1)\ell+1}), \ldots, (x_{t\ell}, y_{t\ell}), \quad t = 1, \ldots, J \tag{45}$$

On each of these J disjoint training subsets we construct its own SVM rule (independent of other rules)

$$y = \theta(f_t(x, \alpha_\ell)), \qquad t = 1, ..., J.$$

For each of these SVM rules, we construct (as described in Section 3.3.2) its own one-dimensional monotonic conditional probability function $P_t(y = 1|s^t),\ t = 1, \ldots, J$.

Then, using these J one-dimensional monotonic condition probability functions, we construct the J-dimensional ($s = (s^1, ..., s^J)$) conditional probability function as follows:

$$P_{syn}(y = 1|s) = \frac{1}{J}\sum_{t=1}^{J} P_t(y = 1|s^t). \tag{46}$$

3. A universal kernel (for example, RBF) can approximate well any bounded continuous function.

Training size	300	300	300	900	1000	3000
RBF SVM	20.77%	19.06%	21.40%	20.01%	19.21%	18.49%
Voting on 3 subsets	N/A	N/A	N/A	19.44%	-	-
Synergy on 3 subsets	N/A	N/A	N/A	18.52%	-	-

Table 4: Synergy versus training size increase.

The Synergy decision rule in this case has the form

$$y = \theta\left(P_{syn}(y=1|s) - \frac{1}{2}\right).$$

Note that (46) forms an unbiased estimate of the values of learning curve describing conditional probability for training data of (different) size ℓ. Since the training data (45) for different t are independent, the averaging of J conditional probability values decreases the variance of resulting conditional probability by a factor of J. In this approach, by choosing an appropriate value of ℓ, one can optimally solve the bias-variance dilemma.

To illustrate this approach, we again used Adult data set. Specifically, we trained SVMs with RBF kernel on Adult data sets containing 900, 1,000 and 3,000 samples. For the first of these samples (containing 900 elements), we also executed the following procedure: we split it into three subsets containing 300 elements each, trained RBF SVM on each of them, and then constructed two combined decision rules: (1) voting on the labels of three auxiliary SVMs, and (2) synergy of three SVMs as described in this section. The results, shown in Table 4, suggest that Synergy of rules on disjoint data sets can be better that straightforward SVMs on training data sets of much larger sizes (in this example, equivalent to the increase of training sample by a factor of 3).

Comparison of Table 3 and Table 4 suggests that synergy of SVMs with different SVM kernels obtained on the same data set may be more beneficial (equivalent to ten-fold increase of training sample size) than the synergy of SVMs with the same kernel obtained on different subsets of of that data set (equivalent to three-fold increase of training sample size).

Thus it is reasonable to assume that, for big data set (44), Synergy of SVM rules obtained on different training data and Synergy of SVM rules with different kernels (described in previous Section 4.1) can be unified to create an even more accurate synergy rule. This unification can be implemented in the following manner.

Consider d kernels $K_r(x, x'),\ k = 1, \ldots, d$. For each of these kernels, using the method described in Section 3.3.2, we construct the corresponding condition probability function

$$P_{syn}(y=1|s(r)) = \frac{1}{J}\sum_{t=1}^{J} P_t(y=1|s^t(r)),$$

where we have denoted by $P_t(y=1|s^t(r))$ the conditional probability function estimated for the rule with kernel $K_r(x, x')$ and for the jth subset of training data (44) with the fixed t. Let introduce the vector $p = (p^1, ..., p^r)$ where

$$p^r = P_{syn}(y=1|s(r)), \quad r = 1, \ldots, d.$$

Using these vectors, we estimate the d-dimensional conditional probability function $P_{syn}(y = 1|p) = P_{syn}(y = 1|p^1, ..., p^d)$.

The resulting double reinforced Synergy rule has the form

$$y = \theta\left(P_{syn}(y = 1|p) - \frac{1}{2}\right).$$

4.3 Multi-Class Classification Rules

Constructing decision rules for multi-class classification is an important problem in pattern recognition. In contrast to methods for constructing two-class classification rules, which have solid statistical justifications, existing methods for constructing $d > 2$ class classification rules are based on heuristics.

One of the most popular heuristics, *one versus rest* (OVR), suggests first to solve the following d two-class classification problems: in problem number k (where $k = 1, \ldots, d$), the examples of class k are considered as examples of the first class and examples of the all other classes $1, \ldots, (k-1), (k+1), \ldots, d$ are considered as the second class. Using OVR approach, one constructs d different two-class classification rules

$$y = \theta(f_k(x)) \quad k = 1, \ldots, d.$$

The new object x_* is assigned to the class k, where kth rule provides the maximum score for x_*:

$$k = \text{argmax}\{s_*^1, \ldots, s_*^d\}, \quad \text{where} \quad s_*^t = f_t(x_*).$$

This method of d-class classification is not based on a clear statistical foundation[4].

Here we implement the following multi-class classification procedure. For every k (where $k = 1, \ldots, d$), we solve the corresponding OVR SVM problem, for which all the elements with the original label k are marked with $y = 1$, while all the other elements are marked with $y = 0$. Upon solving all these d problems, we can, for any given vector x and any class k, compute its score $s_k(x)$ provided by the kth SVM rule.

After that, for every k (where $k = 1, \ldots, d$) we use the obtained scores for estimating conditional probability of the class k based on the scores $(\overline{s}^1, \ldots, \overline{s}^d)$ where

$$\overline{s}^m = \begin{cases} s_m & \text{if } m = k \\ -s_m & \text{if } m \neq k \end{cases}$$

This transformation of scores is used to maintain the monotonicity of the overall conditional probability function. To estimate the function

$$P(k) = P(k|\overline{s}^1, \ldots, \overline{s}^d),$$

as in Section 3.3.2, we use the representation

$$P(k|\overline{s}^1, \ldots, \overline{s}^d) = \sum_{i=1}^{\ell} \left[\alpha_i \min\{\overline{s}_i^k, \overline{s}^k\} + \beta_i \sum_{t \neq k} \min\{\overline{s}_i^t, \overline{s}^t\}\right], \quad k = 1, \ldots, d.$$

4. Another common heuristics called *one versus one* (OVO): it suggests to solve C_d^2 two-class classification problems separating all possible pairs of classes. To classify a new object x^*, one uses a voting scheme based on the obtained C_d^2 rules.

Data set	Classes	Features	Training	Test	OVR	Synergy	Gain
Vehicle	4	18	709	236	17.45%	14.15%	18.91%
Waveform	3	40	200	4800	20.10%	18.31%	8.90%
Cardiotocography	3	21	300	1826	15.83%	12.05%	23.87%

Table 5: Synergy for multi-class classification.

Finally, we replace the heuristic procedure of choosing the class k based on maximization of underlying scores with the following procedure that is based on the framework described above; this procedure uses estimated d conditional probabilities $P(k|s_1, \ldots, s_d)$ (probability of class $k = 1, \ldots, d$ given all d scores) and chooses the class t corresponding to the maximum value of the conditional probability:

$$t = \text{argmax}\{P(1|\overline{s}_*^1, \ldots, \overline{s}_*^d), \ldots, P(d|\overline{s}_*^1, \ldots, \overline{s}_*^d)\}.$$

We compared our synergy approach with the standard OVR approach for the data sets Vehicle, Waveform, and Cardiotocography from UCI Machine Learning Repository (Lichman (2013)). Training and test sets were selected randomly from these data sets; the number of elements in each are shown in Table 5; the table also shows the error rates achieved by OVR and synergy algorithm, along with relative performance gain obtained with our approach. The results confirm the viability of our framework.

5. Synergy of Learning from Several Intelligent Teachers

In (Vapnik and Izmailov (2015d)), (Vapnik and Izmailov (2015b)), we introduced the concept of *knowledge transfer* from Intelligent Teacher to student. Knowledge transfer is possible in the framework of Learning Using Privileged Information (LUPI) paradigm introduced in (Vapnik (2006)) and (Vapnik and Vashist (2009)). According to this paradigm, iid training examples are generated by some unknown generator $P(x), x \in X$ and Intelligent Teacher who supplies vectors x with information (x^*, y) according to some (unknown) *Intelligence generator* $P(x^*, y|x), x^* \in X^*, y \in \{-1, 1\}$, forming training triplets

$$(x_1, x_1^*, y_1), \ldots, (x_\ell, x_\ell^*, y_\ell). \tag{47}$$

Vector x_i^* corresponding to vector x_i is called *privileged information*, and generator $P(x^*, y|x)$ is called generator of intelligent (due to x^*) information. Privileged information is available only for training examples and is *not available* for test examples. In contrast to LUPI, classical learning paradigm considers a primitive teacher that just generates classification y for any x according to $P(y|x)$ (with no additional explanation x^*), forming training pairs

$$(x_1, y_1), \ldots, (x_\ell, y_\ell).$$

Knowledge transfer mechanism. Consider the second[5] mechanism in LUPI paradigm the *knowledge transfer mechanism* to construct a better decision rule. Given triplets (47), we can consider two pattern recognition problems:

1. Pattern recognition problem defined in space X: Using data, $(x_1, y_1), \ldots, (x_\ell, y_\ell)$, find in the set of functions $f(x, \alpha), \alpha \in \Lambda$ the rule $y = \text{sgn}\{f_\ell(x)\}$ that minimizes the probability of test errors (in space X).

2. Pattern recognition problem defined in space X^*: Using data, $(x_1^*, y_1), \ldots, (x_\ell^*, y_\ell)$, find in the set of functions $f^*(x^*, \alpha^*), \alpha^* \in \Lambda^*$ the rule $y = \text{sgn}\{f_\ell^*(x^*)\}$ that minimizes the probability of test errors (in space X^*).

Suppose that, in space X^*, one can find a rule $y = \text{sgn}\{f_0^*(x^*)\}$ that is better (more accurate) than the corresponding rule $y = \text{sgn}\{f_0(x)\}$ in space[6] X.

The question arises: *Can the knowledge about a good rule*

$$f_\ell^*(x^*) = \sum_{i=1}^{\ell} y_i \alpha_i^* K^*(x_i^*, x^*) + b^* \tag{48}$$

in space X^* *help to find a good rule*

$$f_\ell(x) = \sum_{i=1}^{\ell} y_i \alpha_i K(x_i, x) + b \tag{49}$$

in space X?

Consider the following example. Suppose that our goal is to classify images x_i of biopsy in pixel space X into two categories: cancer and non-cancer.

Suppose that, along with images x_i in pixel space X, we are given description of the images $x_i^* \in X^*$ (privileged information), reflecting the existing model of developing cancer:

- *Aggressive proliferation of A-cells into B-cells.*
- *Absence of any dynamic in standard picture of sells distribution.*

Since pixel space X is universal (it can be used for many problems, for example, in pixel space, one can distinguish male faces from female ones), and space of descriptions X^* reflects just the model of cancer development[7], the VC dimension of the corresponding set of functions in X space has to be larger than the VC dimension of the corresponding set of functions in X^*.

Therefore the rule constructed from ℓ examples in space X^* will be more accurate than the rule constructed from ℓ examples in space X. That is why transferring the rule from space X^* into space X can be helpful.

5. The first mechanism is called *similarity control* described in (Vapnik and Izmailov (2015d)), (Vapnik and Izmailov (2015b)). The second mechanism of knowledge transfer, described there and further in this paper, is related to SVM technology. However, the idea of knowledge transfer is general and can be implemented for other learning algorithms.
6. This is always the case if space X is a subset of X^*.
7. In this example, generator $P(x^*, y|x)$ is intelligent since for any *picture* of the event x it describes the *essence* of the event. Using descriptions of the essence of an event makes classification of the event a relatively easy problem.

Knowledge representation in space X^*. To transfer knowledge from space X^* into space X, we use three elements of knowledge representation developed in 1950's in Artificial Intelligence (see Brachman and Levesque, 2004):

1. Fundamental elements of the knowledge in X^*.
2. Main frames (fragments) of the knowledge in X^*.
3. Structure of knowledge: combination of the frames in X^*.

For LUPI using SVM[8]:

1. The *fundamental elements* are defined by k support vectors of rule (48) in X^*.
2. The *frames* in X^* are defined by the functions $K^*(x_s^*, x^*),\ s = 1, \ldots, k$.
3. The structure of the knowledge (48) is linear in the frames.

Algorithm for knowledge transfer. In order to transfer knowledge from space X^* to space X, one has to make two transformations in the training triplets (47):

1. To transform n-dimensional vectors of $x_i = (x_i^1, \ldots, x_i^n)^T$ into k-dimensional vectors $\mathcal{F}x_i = (\phi_1(x_i), \ldots, \phi_k(x_i))^T$;
2. Use the target values $f_\ell^*(x_i^*)$ obtained for x_i^* in rule (48) instead of the values y_i given for x_i in triplet (47).

(1) In order to transform vector x, one constructs k-dimensional space as follows: for any frame $K^*(x^*, x_s^*), s = 1, \ldots, k$ in space X^*, one constructs its image (function) $\phi_s(x)$ in space X that is defined by the relationship

$$\phi_s(x) = \int K(x_s^*, x^*) P(x^*|x) dx^*, \quad s = 1, \ldots, k.$$

This requires to solve the following regression estimation problem: given data

$$(x_1, z_1^s), \ldots, (x_\ell, z_\ell^s), \quad \text{where} \quad z_i^s = K(x_s^*, x_i^*),$$

find k regression functions $\phi_s(x),\ s = 1, \ldots, k$, forming the space $\mathcal{F}(x) = (\phi_1(x), \ldots, \phi_k(x))^T$.

(2) Replace target value y_i in triplets (47) with scores $f_\ell^*(x^*)$ given (48).

Therefore the knowledge transfer algorithm transforms the training triplet[9]

$$((\mathcal{F}x_1, x_1^*, f_\ell^*(x_1^*)), \ldots, (\mathcal{F}x_\ell, x_\ell^*, f_\ell^*(x_\ell^*))). \tag{50}$$

It uses triplets (50) instead of triplets (47).

Synergy of several Intelligent Teachers. Suppose now that Student tries to learn how to solve the same problem from several (say two) Intelligent Teachers. For simplicity, let both Teachers use the same training data $(x_i, y_i),\ i = 1, \ldots, \ell$ but different privileged information (different explanations)

$$(x_1, x_1^*, y_1), \ldots, (x_\ell, x_\ell^*, y_\ell)$$

and

$$(x_1, x^{**}, y_1), \ldots, (x_\ell, x_\ell^{**}, y_\ell).$$

Constructing, using these triplets, two different rules and corresponding synergy rule, one obtains the synergy effect of two Intelligent Teachers.

8. Different concepts of fundamental elements, frames, and structure of knowledge can be applied for different algorithms.
9. In the simplified version, pairs $(\mathcal{F}x_i, s_i^*),\ i = 1, \ldots, \ell$.

6. Conclusion

In this paper, we showed that:

1. Scores $s = (s^1, \ldots, s^d)$ of several monotonic classifiers (for example, SVMs) that solve the same pattern recognition problem can be transformed into multi-dimensional monotonic conditional probability functions $P(y|s)$ (probability of class y given scores s).

2. There exists an effective algorithm for such transformation.

3. Classification rules obtained on the basis of constructed conditional probability functions significantly improve performance, especially in multi-class classification cases.

Acknowledgments

This material is based upon work partially supported by AFRL and DARPA under contract FA8750-14-C-0008, and by AFRL under contract FA9550-15-1-0502. Any opinions, findings and / or conclusions in this material are those of the authors and do not necessarily reflect the views of AFRL and DARPA.

The authors thank the reviewers for their outstanding diligence, which helped to correct multiple errors and typos in the paper.

References

R. Andersen. *Modern methods for robust regression.* Quantitative Applications in the Social Sciences. SAGE, 2008.

M. Best and N. Chakravarti. Active set algorithms for isotonic regression; a unifying framework. *Mathematical Programming*, 47(1):425–439.

R. Brachman and H. Levesque. *Knowledge Representation and Reasoning.* Morgan Kaufman Publishers, San Francisco, CA, 2004.

T. Dietterich. Ensemble methods in machine learning. In *Multiple Classifier Systems, LBCS-1857*, pages 1–15. Springer, 2000.

R. Izmailov, V. Vapnik, and A. Vashist. Multidimensional splines with infinite number of knots as SVM kernels. In *International Joint Conference on Neural Networks*, pages 1096–1102, 2013.

P. Jäckel. *Monte Carlo methods in finance.* Wiley, 2004.

G. Kimeldorf and G. Wahba. A correspondence between bayesian estimation on stochastic processes and smoothing by splines. *The Annals of Mathematical Statistics*, pages 495–502, 1970.

G. Lecué. Optimal rates of aggregation in classification under low noise assumption. *Bernoulli*, 13(4):1000–1022, 11 2007.

M. Lichman. UCI machine learning repository, 2013. URL `http://archive.ics.uci.edu/ml`.

H. Lin, C. Lin, and R. Weng. A note on platt's probabilistic outputs for support vector machines. *Machine Learning*, 68(3):267–276, 2007.

P. Mair, K. Hornik, and J. de Leeuw. Isotone optimization in R: pool-adjacent-violators algorithm (PAVA) and active set methods. *Journal of Statistical Software*, 32(5):1–24, October 2009.

P. Massart. The tight constant in the Dvoretzky-Kiefer-Wolfowitz inequality. *The Annals of Probability*, 18(3):1269–1283, 1990.

M. Meyer. Semi-parametric additive constrained regression. *Journal of Nonparametric Statistics*, 25(3):715–730, 2013.

J. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 1999.

J. Stork, R. Ramos, P. Koch, and W. Konen. SVM ensembles are better when different kernel types are combined. In *ECDA*, Studies in Classification, Data Analysis, and Knowledge Organization, pages 191–201. Springer, 2013.

O. Sysoev, O. Burdakov, and A. Grimvall. A segmentation-based algorithm for large-scale partially ordered monotonic regression. *Computational Statistics & Data Analysis*, 55(8): 2463–2476, 2011.

A. Tikhonov and V. Arsenin. *Solutions of Ill-Posed Problems*. W.H. Winston, 1977.

A. Tsybakov. *Optimal Rates of Aggregation*, pages 303–313. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

V. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.

V. Vapnik. *Estimation of Dependencies Based on Empirical Data*. Springer–Verlag, 2nd edition, 2006.

V. Vapnik and R. Izmailov. V-matrix method of solving statistical inference problems. *Journal of Machine Learning Research*, 16:1683–1730, 2015a.

V. Vapnik and R. Izmailov. Learning using privileged information: Similarity control and knowledge transfer. *Journal of Machine Learning Research*, 16:2023–2049, 2015b.

V. Vapnik and R. Izmailov. Statistical inference problems and their rigorous solutions. In A. Gammerman, V. Vovk, and H. Papadopoulos, editors, *Statistical Learning and Data Sciences*, volume 9047 of *Lecture Notes in Computer Science*, pages 33–71. Springer International Publishing, 2015c.

V. Vapnik and R. Izmailov. Learning with intelligent teacher: Similarity control and knowledge transfer. In A. Gammerman, V. Vovk, and H. Papadopoulos, editors, *Statistical Learning and Data Sciences*, volume 9047 of *Lecture Notes in Computer Science*, pages 3–32. Springer International Publishing, 2015d.

V. Vapnik and A. Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5-6):544–557, 2009.

V. Vapnik, I. Braga, and R. Izmailov. Constructive setting for problems of density ratio estimation. *Statistical Analysis and Data Mining*, 8(3):137–146, 2015.

S. Wang, A. Mathew, Y. Chen, L. Xi, L. Ma, and J. Lee. Empirical analysis of support vector machine ensemble classifiers. *Expert Systems with Applications*, 36(3 Pt2):6466–6476, 2009.

C. Zhang and Y. Ma. *Ensemble Machine Learning: Methods and Applications*. Springer New York, 2012.