Proceedings of Machine Learning Research 128:1–37, 2020 Conformal and Probabilistic Prediction and Applications

Complete Statistical Theory of Learning (Learning Using Statistical Invariants)

Vladimir Vapnik* VLADIMIR.VAPNIK@GMAIL.COM
Columbia University, New York, NY, USA
Rauf Izmailov RIZMAILOV@PERSPECTALABS.COM
Perspecta Labs, Basking Ridge, NJ, USA

Editor: Alexander Gammerman, Vladimir Vovk, Zhiyuan Luo, Evgueni Smirnov and Giovanni Cherubin

Abstract

Statistical theory of learning considers methods of constructing approximations that converge to the desired function with increasing number of observations. This theory studies mechanisms that provide convergence in the space of functions in L_2 norm, i.e., it studies the so-called strong mode of convergence. However, in Hilbert space, along with the convergence in the space of functions, there also exists the so-called weak mode of convergence, i.e., convergence in the space of functionals. Under some conditions, this weak mode of convergence also implies the convergence of approximations to the desired function in L_2 norm, although such convergence is based on other mechanisms.

The paper discusses new learning methods which use both modes of convergence (weak and strong) simultaneously. Such methods allow one to execute the following: (1) select an admissible subset of functions (i.e., the set of appropriate approximation functions), and (2) find the desired approximation in this admissible subset.

Since only two modes of convergence exist in Hilbert space, we call the theory that uses both modes *the complete statistical theory of learning*.

Along with general reasoning, we describe new learning algorithms referred to as *Learning Using Statistical Invariants* (LUSI). LUSI algorithms were developed for sets of functions belonging to Reproducing Kernel Hilbert Space (RKHS); they include the modified SVM method (LUSI-SVM method). Also, the paper presents a LUSI modification of Neural Networks (LUSI-NN). LUSI methods require fewer training examples that standard approaches for achieving the same performance.

In conclusion, the paper discusses the general (philosophical) framework of a new learning paradigm that includes the concept of intelligence.

Keywords: Learning Theory, Weak convergence, Statistical Invariants, Complete solution of learning problem, Reproducing Kernel Hilbert Space, Kernel Machines, Statistical Invariants for Support Vector Classification, Statistical Invariants for Support Vector Regression, Statistical Invariants for Neural Nets, Predicates, Symmetries Invariants.

* This material is based upon the work partially supported by AFRL under contract FA9550-19-1-0124. Any opinions, findings and / or conclusions in this material are those of the authors and do not necessarily reflect the views of AFRL.

1. Introduction

1.1. Main Results of VC Theory of Learning

About fifty years ago, the statistical theory of learning, the so-called VC theory[1] was developed [1]. This theory addressed the following question.

Under what circumstances, when using a finite number of i.i.d. observations

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

generated according to some unknown distribution function $P(x, y)$, one can find, in a *given set of indicator functions*[2] $\{f(x)\} : X \to \{0, 1\}$, a function $f_\ell(x)$ that is close to the one that minimizes expected risk

$$R(f) = \int L(y - f(x))dP(x, y),$$

defined by some nonnegative loss function $L(y - f(x))$.

The answer to that question was that this is possible *if and only if* the measure of capacity (diversity) of the given set of functions $\{f(x)\}$, namely, the so-called VC dimension h of this set of functions (defined below), is finite. The following was proved:

1. If the VC dimension is finite (i.e., $h < \infty$) then, with probability $1 - \eta$, the bound

$$\left|R(f) - R^\ell_{emp}(f)\right| \le \varepsilon\sqrt{1 + \frac{4R^\ell_{emp}(f)}{\varepsilon}}, \quad \varepsilon = O\left(\frac{h - \ln\eta}{\ell}\right) \tag{1}$$

holds true *simultaneously for all functions* $\{f(x)\}$, where

$$R^\ell_{emp}(f) = \frac{1}{\ell}\sum_{i=1}^{\ell} L(y_i - f(x_i)). \tag{2}$$

Bound (1) implies that the inequality

$$R(f) \le R^\ell_{emp}(f) + \varepsilon\sqrt{1 + \frac{4R^\ell_{emp}(f)}{\varepsilon}} \tag{3}$$

holds for all $f \in \{f(x)\}$ with probability $1 - \eta$. Therefore, the smallest guaranteed risk $R(f)$ is realized by the function that minimizes the empirical loss $R_{emp}(f)$.

2. If, however, the VC dimension is infinite (i.e., $h = \infty$), then there exists a generator $P(x)$ of random vectors x such that, for almost any iid generated sequence $x_1, \ldots, x_\ell$ (of any size ℓ), the set of functions $\{f(x)\}$ contains 2^ℓ functions that shatter the sequence in all 2^ℓ possible ways. In this case, one can find a function with training loss equal to 0 and test loss equal to 1. This means that one cannot find a function with small guaranteed risk $R(f)$ in $\{f(x)\}$ by using only given observations.

1. Abbreviation for Vapnik-Chervonenkis theory.
2. Subsequently, this theory was generalized to real-valued functions [2].

Bound (1) is called *uniform convergence* bound (it is uniform over all functions from $\{f(x)\}$) and bound (3) is called *guaranteed risk bound* [1].

Definition of VC dimension. The VC dimension of the set of indicator functions is equal to h if (i) there exist h vectors $x_1, \ldots, x_h$ such that they can be separated (shattered) by functions from this set in all 2^h possible ways, and (ii) there exist no $h+1$ vectors that can be separated (shattered) in 2^{h+1} ways. The VC dimension of the set is infinite if such vectors exist for any number h.

VC dimension of subsets of linear indicator functions. The combinatorial definition of VC dimension allows one to estimate VC dimension of the following subsets of linear functions, which are important for applications.

Theorem [2]. Let vectors $x \in R^n$ belong to a sphere of radius 1. Consider the subset of indicator functions $\theta(f(x,w))$ defined by hyperplanes $f(x) = (w,x)$ with bounded norms $||w||^2 \le B$ of parameter vectors:

$$f_w(x) = \theta[(w,x)], \quad \theta(u) = \begin{cases} 1 & \text{if } u \ge 0, \\ 0 & \text{if } u < 0. \end{cases} \tag{4}$$

The VC dimension h of this subset has the bound

$$h < \min(n, B) + 1. \tag{5}$$

Thus VC dimension is bounded by the smallest of two values: the dimensionality n of R^n and the bound B of weight norms $||w||^2$). That is, according to (5), *VC dimension can be much smaller than the dimensionality of the space* and it can be effectively controlled by the value B. This fact plays an important role for constructing learning algorithms.

In order to prove this theorem, one has to construct the largest simplex (i.e., the simplex with the largest number of vertices) within the sphere of radius 1, where the vertices of that simplex can be separated in two subsets by functions (4) in all 2^h possible ways using weights with norms $||w||^2 \le B$.

Structural Risk Minimization Principle. Using bound (3), the VC theory introduced the Structural Risk Minimization inductive principle for searching for the desired approximation in a given set of functions. Let a structure of nested subsets

$$S_1 \subset S_2 \cdot \cdot \subset S_p \subset \cdots \tag{6}$$

be defined on the given set $\{f(x)\}$, where VC dimensions h_k of subsets S_k form a non-decreasing sequence:

$$h_1 \le h_2 \le \ldots \le h_p \le \ldots.$$

Then, for any subset S_k, the bound

$$R(f) \le R^{\ell}_{emp}(f) + \varepsilon_k \sqrt{1 + \frac{4R^{\ell}_{emp}(f)}{\varepsilon_k}}, \quad \varepsilon_k = O^*\left(\frac{h_k - \ln \eta}{\ell}\right) \tag{7}$$

holds true. Since VC dimension h_k is fixed for any S_k (and thus the value ε_k in (7) is defined), the smallest *guaranteed* bound of expected risk is achieved when one chooses the function $f_k(x)$ which minimizes the empirical risk in S_k.

Remark 1. According to bound (1), for functions with empirical loss $R_{emp}(f^*) = 0$, the expected loss is bounded as

$$R(f^*) \le O^* \left(\frac{h - \ln \eta}{\ell} \right),$$

while for functions with loss $R_{emp}(f^*) \neq 0$, the expected loss is bounded as

$$R(f^*) \le R_{emp}(f^*) + O^* \left(\sqrt{\frac{h - \ln \eta}{\ell}} \right).$$

Therefore, the upper bound of risk depends on the value of empirical loss (the smaller is the empirical loss, the smaller is the confidence interval). This fact plays an important role in constructions of learning algorithms: it increases their reliance on data memorization.

1.2. Beyond Statistical Learning Theory

With all its impact, the statistical learning theory has not addressed the following four questions:

1. How to choose the loss function $L(y - f(x))$ in the target functional $R_{emp}(f)$?
2. How to choose the admissible set of functions $\{f(x)\}$?
3. How to construct the nested structure on the admissible set?
4. How to minimize the target functional on the elements of the nested structure?

In this paper, we provide answers to all these questions. We refer to the resulting extended VC theory, which include answers to these four questions, as *Complete Statistical Learning Theory* (or Complete VC theory).

2. Settings of Pattern Recognition Problem

2.1. Phenomenological Learning Model

Below we consider binary classification problem; its generalization to multiclass classification problems and regression problems is straightforward.

Consider the following model of pattern recognition problem (Figure 1).

> Suppose that some generator $\mathcal{G}$ generates vectors $x \in X$ randomly and independently, according to an *unknown* distribution function $P(x)$. Suppose that some object $\mathcal{O}$ transforms any input vector x into $y \in \{0, 1\}$. We assume that transformation of vector x into value y is carried out according to some *unknown* conditional probability function $P(y|x)$. Without loss of generality, instead of function $P(y|x)$, we consider function $f(x) = P(y = 1|x)$, where $P(y = 0|x) = 1 - P(y = 1|x)$.

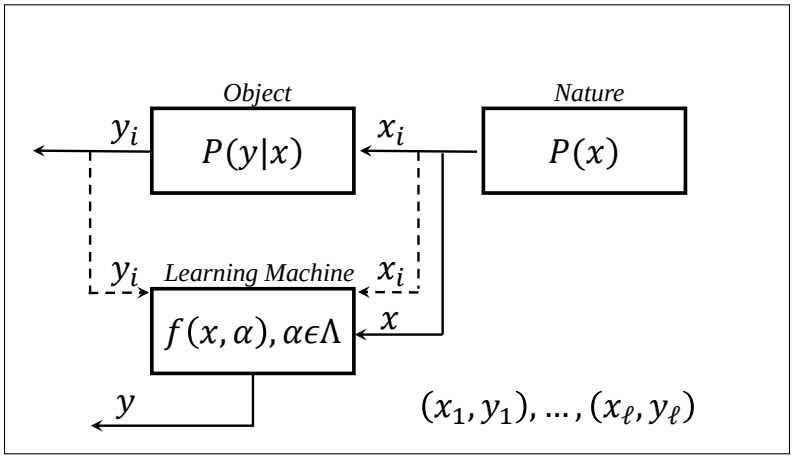


Figure 1: Interaction of *Learning Machine* with data (which is generated by *Nature*) and *Object* (which provides classification of data).

Consider a Learning Machine that can implement functions from some set of admissible indicator functions $\{\theta(f(x))\}$ (here $\{f(x)\}$ is a set of real-valued functions). Let Learning Machine observe ℓ i.i.d. pairs (x_i, y_i), where $i = 1, \ldots, \ell$, generated by an *unknown* joint distribution function $P(y, x) = P(y|x)P(x)$.

The goal of Learning Machine is to select, using ℓ observations, such function $f_0(x)$ in the set of admissible functions that minimizes the probability of error; that is, minimizes the functional

$$R(f) = \int (y - \theta(f(x)))^2 dP(x, y). \tag{8}$$

2.2. Why Minimization of Square Loss Functional Is Not The Best Idea

From a computational point of view, the minimization of functional (8) is a hard computational problem: the loss function $L(y, f(x)) = (y - \theta(f(x))^2$ is discontinuous (it takes only two values, so the gradient of this loss function is either undefined or is equal to 0). Therefore, classical methods replace function $\theta(f(x))$ in (8) with continuous function $f(x) \in \{f(x)\}$ and target minimizing the loss in the set of functions $\{f(x)\}$:

$$R(f) = \int (y - f(x))^2 dP(x, y). \tag{9}$$

Minimum $f_0(x)$ of functional (9) defines conditional probability function $f_0(x) = P(y = 1|x)$ (assuming that $f_0(x) \in \{f(x)\}$). This function defines the optimal decision rule

$$r(x) = \theta(f_0(x) - 0.5). \tag{10}$$

Note that when one tries to estimate the conditional probability function using a small number ℓ of observations, the minimization of (9) may be not the best idea. Indeed, one

can rewrite this functional as follows:

$$R(f) = \int (y - f(x)^2 dP(x,y) =$$

$$\int ((y - f_0(x)) + (f_0(x) - f(x)))^2 dP(x,y) = \int (y - f_0(x))^2 dP(x,y) +$$

$$2\int (y - f_0(x))(f_0(x) - f(x)) dP(x,y) + \int (f_0(x) - f(x))^2 dP(x). \tag{11}$$

Since only the last two integrals in (11) depend on function $f(x)$, the minimum of (9) is defined by the sum of last *two* integrals, and not by the minimum of the last *one*. When one estimates regression from a limited number ℓ of observations, minimization of the sum of two integrals (rather than minimization of only the last one) can slow down the rate of convergence to the desired function.

2.3. Direct Methods of Estimation of Conditional Probability Function

In order to estimate the conditional probability function $f_0(x) = P(y = 1|x)$ on the set of functions $\{f(x)\}$ (where $f_0(x) \in \{f(x)\}$), consider equality

$$P(y = 1|x)p(x) = f_0(x)p(x) = p(y = 1, x); \tag{12}$$

here $p(y = 1, x)$ and $p(x)$ are density functions. From (12), we obtain the following equality for any function $G(x - x') \in L_2$:

$$\int G(x - x')f(x)dP(x) = \int G(x - x')dP(y = 1, x). \tag{13}$$

The solution of Fredholm equation (13) (with respect to $f(x) \in \{f(x)\}$ when the right-hand side of the equation (13) is known) defines conditional probability function $P(y = 1|x)$. The estimation of the conditional probability function from given data is thus realized by solving the corresponding Fredholm integral equation when probability measures $P(y = 1, x)$ and $P(x)$ are unknown but iid data

$$(x_1, y_1), \ldots, (x_\ell, y_\ell), \quad x = (x^1, \ldots, x^n),\ y \in \{0, 1\}$$

are given.

2.3.1. Constructive Equation for Direct Estimation.

In order to solve equation (13) using data, we use the following inductive step: we replace the unknown cumulative distribution functions with their estimates:

$$P_\ell(x) = \frac{1}{\ell}\sum_{i=1}^{\ell} \theta(x - x_i), \qquad \theta\{x - x_i\} = \prod_{k=1}^{n} \theta\{x^k - x_i^k\}.$$

Replacing $P(x)$ and $P(y = 1, x)$ in (13) with their estimates, we obtain

$$\frac{1}{\ell}\sum_{i=1}^{\ell} G(x - x_i)f(x_i) = \frac{1}{\ell}\sum_{i=1}^{\ell} y_i G(x - x_i). \tag{14}$$

In order to estimate the condition probability function, we have to solve equation (14) in the set of functions $\{f(x)\}$.

Remark 2. In classical statistics, Nadaraya-Watson estimator of regression for $y \in \{0, 1\}$ defines conditional probability function using the formula

$$P(y = 1|x) = \frac{\sum_{i=1}^{\ell} y_i G(x - x_i)}{\sum_{i=1}^{\ell} G(x - x_i)},$$

where special kernels (say, Gaussian kernel $\exp\left\{-||x - x_i||^2/(2\sigma^2)\right\}$) are used.

This estimator is the solution of the "corrupted" equation

$$\frac{1}{\ell}\sum_{i=1}^{\ell} G(x - x_i)f(x) = \frac{1}{\ell}\sum_{i=1}^{\ell} y_i G(x - x_i)$$

(which uses a special kernel) rather than the original equation

$$\frac{1}{\ell}\sum_{i=1}^{\ell} G(x - x_i)f(x_i) = \frac{1}{\ell}\sum_{j=1}^{\ell} y_j G(x - x_j),$$

where one can use any kernel $G(x - x')$ from L_2.

The main problem with Nadaraya-Watson estimator is to find the best width parameter $\sigma > 0$ of the kernel. There are several recommendations for choosing the value of this parameter. Since the solution of equation (14), which defines the conditional probability function, exists for any function $G(x - x')$, it seems reasonable to use the parameter that is optimal for Nadaraya-Watson estimator of conditional probability function.

2.3.2. Solution of Equation.

In order to solve equation (14) in the set of functions $\{f(x)\}$, we minimize the distance

$$\rho^2 = \int \left(\sum_{i=1}^{\ell} G(x - x_i)f(x_i) - \sum_{j=1}^{\ell} y_j G(x - x_j)\right)^2 d\mu(x),$$

where $\mu(x)$ is a given (probability) measure which defines the required concept of closeness. This distance can be rewritten as

$$\rho^2 = \sum_{i,j=1}^{\ell} (f(x_i - y_i)(f(x_j) - y_j)v(x_i, x_j), \tag{15}$$

where values $v(x_j, x_j)$ are defined as

$$v(x_i, x_j) = \int G(x - x_i)G(x - x_j)d\mu(x). \tag{16}$$

Elements $v(x_i, x_j)$ form an $(\ell \times \ell)$-dimensional matrix, which we refer to as $\mathcal{V}$-matrix.

Example 1. Consider the case $\mu(x) = P_\ell(x)$, where $P_\ell(x)$ is an empirical estimate of unknown measure $P(x)$. For this measure, we obtain the following elements $v(x_i, x_j)$ of $\mathcal{V}$-matrix:

$$v(x_i, x_j) = \frac{1}{\ell} \sum_{s=1}^{\ell} G(x_s - x_i) G(x_s - x_j).$$

Example 2. Let $G(x - x') = \exp\{-0.5\Delta^2 (x - x')^2\}$. Consider one-dimensional case of $x \in (-a, a)$, where x is uniformly distributed on $(-a, a)$ (i.e., $\mu(x) = (2a)^{-1} x$). For this case,

$$v(x_i, x_j) = \frac{1}{2a} \int \exp\{-0.5\Delta^2 (x - x_i)^2\} \exp\{-0.5\Delta^2 (x - x_j)^2\} dx.$$

After integration, we obtain

$$v(x_i, x_j) = C \exp\left\{ -\frac{(x_i - x_j)^2}{\sigma^2} \right\} \left[\operatorname{erf}\left(\frac{c + \widehat{x}_{i,j}}{\sigma} \right) + \operatorname{erf}\left(\frac{c - \widehat{x}_{i,j}}{\sigma} \right) \right],$$

where we have denoted $(x_i + x_j)/2 = \widehat{x}_{i,j}$.

For multidimensional case of $x = (x^1, ..., x^n) \in [-c_1, c_1] \times \cdots \times [-c_n, c_n]$, the elements $v(x_i, x_j)$ of $\mathcal{V}$-matrix have the form

$$v(x_i, x_j) = C \exp\left\{ -\frac{||x_i - x_j||^2}{\sigma^2} \right\} \prod_{k=1}^{n} \left[\operatorname{erf}\left(\frac{c_k + \widehat{x}^k_{i,j}}{\sigma} \right) + \operatorname{erf}\left(\frac{c_k - \widehat{x}^k_{i,j}}{\sigma} \right) \right].$$

When minimizing (15), one can ignore constant C in $\mathcal{V}$-matrix.

Target Functional in Matrix Form. For simplicity, we introduce matrix notations. Consider $(\ell \times 1)$-dimensional matrix $Y = (y_1, \ldots, y_\ell)^T$, where binary values y_i are labels of the elements x_i in training data. For any function $f(x)$ from the set $\{f(x)\}$, consider $(\ell \times 1)$-dimensional matrix $F(f) = (f(x_1), \ldots, f(x_\ell))^T$. For the selected kernel function $G(x, x')$, consider $(\ell \times \ell)$-dimensional $\mathcal{V}$-matrix of elements $v(x_i, x_j)$.

In these notations, the target functional $R(f)$ has the form

$$R(f) = (F(f) - Y)^T \mathcal{V} (F(f) - Y). \tag{17}$$

Our goal is to minimize (17) in the set of admissible functions $\{f(x)\}$.

Therefore, finding solution of equation (14) in a given set of functions $\{f(x)\}$ requires the minimization of the functional (17) rather than the minimization of the classical least square functional

$$R(f) = (F(f) - Y)^T \mathcal{I} (F(f) - Y),$$

where the identity matrix $\mathcal{I}$ is used instead of $\mathcal{V}$-matrix.

3. Estimation of Conditional Probability Function and Solution of Ill-Posed Problems

In order to find conditional probability function, we need to solve the approximation (14) of Fredholm equation (13) in a given set of functions $\{f(x)\}$. However, it is known that solving the Fredholm integral equation is an *ill-posed problem*, as explained next.

3.1. Well-Posed and Ill-posed Problems

Let A be a linear operator which maps elements f of a metric space E_1 into elements F of a metric space E_2. The solution of operator equation

$$Af = F \tag{18}$$

in the set $\{f\}$ is *well-posed* if the solution (i) *exists*, (ii) is *unique*, and (iii) is *continuous*. That is, if the functions F_1 and F_2 of the right-hand side of equation (43) are close in the metric of space E_2 (i.e., $\rho_{E_2}(F_1, F_2) \leq \varepsilon$), they correspond to the solutions f_1 and f_2 that are close in the metric of space E_1 (i.e., $\rho_{E_1}(f_1, f_2) \leq \delta$). The problem is called *ill-posed* if at least one of the three conditions above is violated. Below we consider the ill-posed problems where unique solutions exist, but the inverse operator

$$f = A^{-1}F$$

could be discontinuous. Inference problems defined by the Fredholm equation

$$\int_0^1 \theta(x-t)f(t)dP(x) = P(y=1,x)$$

are ill-posed. Thus the solution of the problem of statistical inference requires *to solve ill-posed problems described by Fredholm equations with both both right-hand side (i.e, F) and left-hand side (i.e., operator A) of equation (18) defined approximately.*

3.2. Regularization of Ill-Posed Problems

The solution of ill-posed problems is based on the following lemma.

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

Consider a continuous non-negative functional $W(f)$ and the set of functions

$$\mathcal{M}_C = \{f : W(f) \leq C\}. \tag{19}$$

defined by a constant $C > 0$. Let the set of functions $\mathcal{M}_C$ be convex and compact for any C. Suppose that the solution of operator equation belongs to compact sets $\mathcal{M}_C$.

The idea of solving ill-posed equation (18) is to choose an appropriate compact set (i.e., to choose a constant C^* in (19)) and then to minimize the square of distance between left- and right-hand sides of equation (18) on this compact set of functions[3] defined by C^*. In other words, the idea is to minimize the square of distance

$$\rho = \rho^2_{E_2}(Af, F) \tag{20}$$

3. Note that this idea of solving ill-posed problems is the same one as in structural risk minimization in VC theory. In both cases, a structure is defined on the set of functions.When solving well-posed problems, elements of structure should have finite VC-dimension. When solving ill-posed problems, elements of structure should be compact sets.

in E_2 space over functions f subject to the constraint

$$W(f) \leq C^*. \tag{21}$$

The equivalent form of this approach is Tikhonov's regularization method. In this method, the functional

$$R(f) = \rho^2_{E_2}(F(x), Af) + \gamma_{c^*} W(f) \tag{22}$$

is minimized, where $\gamma_{c^*} > 0$ is the regularization constant which depends on the value C^* in (21).

The expression (22) is the Lagrangian functional for the problem of minimizing (20) subject to (21), where parameter γ_{c^*} is defined by the parameter C^* that defines the chosen compact set of functions $\{f(x)\}$ satisfying (21). The parameter $\gamma = \gamma_{c^*}$ in (22) should be chosen in such a way that the equality

$$W(f_*) = C^*$$

holds true for the solution f^* of the minimization problem.

The following theorem holds true for regularization method (22).

Theorem. Let E_1 and E_2 be metric spaces and suppose that, for $F \in E_2$, there exists a solution of the equation

$$Af = F$$

that belongs to the set $f \in \{W_{E_1}(f) \leq C\}$ for $C > C_0$. Let the right-hand side F of this equation be approximated with F_δ such that $\rho(F, F_\delta) \leq \delta$. Suppose that the values of (regularization) parameters $\gamma(\delta)$ are such that

$$\gamma(\delta) \longrightarrow 0, \quad \text{for } \delta \longrightarrow 0$$

$$\lim_{\delta \to 0} \frac{\delta^2}{\gamma(\delta)} \leq r \leq \infty.$$

Then the elements $f_{\gamma(\delta)}$ minimizing the functional

$$R(f) = \rho^2_{E_2}(Af, F(\delta)) + \gamma(\delta) W_{E_1}(f)$$

converge to the exact solution as $\delta \longrightarrow 0$.

Remark 3. This theorem was generalized in [2] for the case where (i) approximately defined right-hand sides of equation converges to the true right-hand side and (ii) approximately defined operators A_ε converge to operator A.

Remark 4. In the terminology of VC theory, regularization corresponds to (i) constructing, in a given set of functions $\{f(x)\}$, a structure of compacts $\mathcal{M}_C = \{f : W(f) \leq C\}$ with bounded VC dimension, and (ii) subsequent selection of an element $\mathcal{M}_C$ and a function in this element that minimizes (22). The choice of γ in (22) is equivalent to the choice of an element of the structure in Structural Risk Minimization.

4. Selection of Admissible Set of Functions

Strong and Weak Modes of Convergence. In order to select an admissible set of functions from a large set of functions (for instance, the set of continuous bounded functions), we use the idea of weak convergence of functions in Hilbert space. It is known that there are two modes of convergence in Hilbert space: the *strong mode* and *the weak mode*.

Strong convergence. The sequence of functions $f_1(x), \ldots, f_\ell(x), \ldots$ converges to function $f_0(x)$ in *strong mode* (in the space of functions) if

$$\lim_{\ell \to \infty} ||f_\ell(x) - f_0(x)|| = 0.$$

Weak convergence. The sequence of functions $f_1(x), \ldots, f_\ell(x), \ldots$ converges to function $f_0(x)$ in weak mode (in the space of functionals) if the equality

$$\lim_{\ell \to \infty} (\phi(x), \{f_\ell(x) - f_0(x)\}) = 0$$

holds true *for all functions* $\phi(x) \in L_2$.

For our problem, the sequence of estimates of conditional probability functions converges to the desired function $f_0(x) = P(y = 1|x)$ in weak mode if

$$\lim_{\ell \to \infty} \int \phi(x) f_\ell(x) dP(x) = \int \phi(x) dP(y = 1, x), \quad \forall \phi(x) \in L_2.$$

It is easy to show (using Cauchy-Schwartz inequality) that if a sequence of functions $f_\ell(x)$ strongly converges to function $f_0(x)$ then it also converges weakly. It is also known that, under some conditions, the converse theorem holds true ([5], Chapter 7.8):

Theorem. If set of functions $\{f(x)\}$ is a compact then weak convergence of estimates implies strong convergence.

In this paper, we estimate conditional probability functions belonging to sets of functions with bounded norm in Reproducing Kernel Hilbert Space (RKHS). This set is a compact.

Admissible Set of Functions. To select an admissible set of functions, we relax the concept of weak convergence. We consider a finite set of functions

$$\phi_s(x), \ s = 1, \ldots, m,$$

which we call *predicates*.

For any predicate $\phi_s(x), s = 1, \ldots, m$, the following equality holds:

$$\int \phi_s(x) P(y = 1|x) dP(x) = \int \phi_s(x) dP(y = 1, x), \quad s = 1, \ldots, m. \tag{23}$$

Suppose that one knows the values of right-hand sides of equations (23). Then one can select, from a large set of functions (for instance, bounded continuous functions), the subset of functions $\{f(x)\}$ that satisfy the equalities

$$\int \phi_s(x) f(x) dP(x) = a_s, \quad s = 1, \ldots, m. \tag{24}$$

This set of functions $\{f(x)\}$ we call the *admissible set of functions.*

Note that the desired condition probability function $f_0(x) = P(y = 1|x)$ belongs to the set of admissible functions and, with the increasing number of m of predicates, the set of admissible functions keeps shrinking monotonically[4].

In reality, we do not know $P(x)$ and $P(y = 1|x)$ in the left- and the right-hand sides of equation (23). However, replacing them with their empirical approximations $P_\ell(x)$ and $P_\ell(y = 1, x)$, we can find the following approximation of (23):

$$\frac{1}{\ell}\sum_{i=1}^{\ell}\phi_s(x_i)f(x_i) = \frac{1}{\ell}\sum_{i=1}^{\ell}y_i\phi_s(x_i), \quad s = 1, \ldots, m. \tag{25}$$

We refer to equalities (25) as *statistical invariants defined on the training set* (x_i, y_i). We consider the functions satisfying (25) as the *admissible set* $\{f(x)\}$.

To simplify notations, we introduce two ℓ-dimensional vectors: vector

$$\Phi_s = (\phi_s(x_1), \ldots .\phi_s(x_\ell))^T$$

of predicate $\phi_s(x)$ and vector

$$F(f) = (f(x_1), \ldots, f(x_\ell))^T,$$

where $F(f)$ transforms any function from admissible set $\{f(x)\}$ into ℓ-dimensional vector of its values defined on training vectors x_i. In these notations, we rewrite equation (25) as

$$\Phi_s^T F(f) = \Phi_s^T Y, \quad s = 1, \ldots, m. \tag{26}$$

Remark 5. Since equation (26) does not depend on the scale of vectors Φ_s, we use normalized vectors Φ_s, so that $||\Phi_s|| = 1$.

5. Complete Solution of Learning Problem

5.1. The Exact Solution

The complete solution of our problem of estimating conditional probability function requires (i) selection of an admissible subset of a large set of functions using both training data and predicates and then (ii) selection of the desired approximation from the admissible subset of functions using training data.

Formally, this requires solving the following constrained optimization problem: in a given set of functions $\{f(x)\}$, minimize the functional

$$R(f) = (F(f) - Y)^T \mathcal{V}(F(f) - Y) \tag{27}$$

subject to the constraints

$$\Phi_s^T F(f) = \Phi_s Y, \quad s = 1, \ldots, m. \tag{28}$$

4. As follows from Section 4, if the conditional probability function belongs to a compact and the set of predicates consists of *all* functions $\phi(x) \in L_2$, the admissible set of functions contains only one function – namely, the desired one.

Below we present closed-form solution of this optimization problem for sets of functions that belong to Reproducing Kernel Hilbert Space.

Remark 6. When the number ℓ of observations is small, the solution of optimization problem (27)-(28) is not a good approximation of the solution of equation (13) with constraints (23).

Indeed, the goal is to minimize the functional (13) in the set of functions satisfying (23). However, since probability measure that defines (23) is unknown, we replace it in (23) with empirical estimate (28).

According to Hoeffding inequality, however, for bounded functions $a \le u(x) \le b$, the inequality

$$P\left\{\left|\int u(x)dP(x) - \frac{1}{\ell}\sum_{i=1}^{\ell} u(x_i)\right| > \varepsilon\right\} \le 2\exp\left\{-\frac{2\varepsilon^2\ell}{(b-a)^2}\right\}$$

holds true. That is, with probability $1-\eta$, the inequality

$$\left|\int u(x)dP(x) - \frac{1}{\ell}\sum_{i=1}^{\ell} u(x_i)\right| \le (a-b)\sqrt{-\frac{\ln\eta/2}{2\ell}}$$

is valid. Using this fact, we conclude that for any fixed $f(x)$

$$\left|\int \phi_s f(x)dP(x) - \frac{1}{\ell}\sum_{i=1}^{\ell} \phi_s(x_i)f(x_i)\right| \le \varepsilon_s$$

$$\left|\int y\phi_s(x)dP(y=1|x) - \frac{1}{\ell}\sum_{i=1}^{\ell} \phi_s(x_i)y_i\right| < \varepsilon_s$$

where

$$\varepsilon_s = (a_s - b_s)\sqrt{\frac{-\ln\eta/2}{\ell}}, \quad a_s = \sup_x \phi_s(x),\ b = \inf_x \phi_s(x).$$

Therefore, for small sample size ℓ, instead of equality constraints (28) it is better to use more accurate inequality constraints

$$\left|\Phi_s^T F(f) - \Phi_s^T Y\right| \le \ell\varepsilon_s, \quad s = 1,\ldots,m. \tag{29}$$

5.2. Approximate Solutions of Complete Learning Problem

Approximation 1. L_2 **Unconstrained Minimization**. In this approximation of the problem, instead of minimizing functional (27) subject to constraints (28) in a given set of functions $\{f(x)\}$, we minimize functional

$$R(f) = \widehat{\tau}(F(f)-Y)^T\mathcal{V}(F(f)-Y) + \frac{\tau}{m}\sum_{s=1}^{m}\left(\Phi_s^T F(f) - \Phi_s^T Y\right)^2,$$

where $\widehat{\tau}+\tau = 1,\ \tau \ge 0$ is a free parameter which describes the relative importance of both terms of functional $R(f)$.

Simple algebra leads to the following equivalent expression of this functional:

$$R(f) = (F(f) - Y)^T(\widehat{\tau}\mathcal{V} + \tau\mathcal{P})(F(f) - Y), \tag{30}$$

where matrix $\mathcal{P}$ is defined as

$$\mathcal{P} = \frac{1}{m}\sum_{s=1}^{m}\Phi_s\Phi_s^T.$$

Approximation 2. L_1 **Constrained Minimization.** Here we consider the problem of minimization in given set of functions $\{f(x)\}$ the functional

$$R(f) = ||f(x)||^2 + C\left(\tau\sum_{s=1}^{m}\xi_s + \widehat{\tau}\sum_{i=1}^{\ell}\xi_{(m+i)}\right) \tag{31}$$

subject to the constraints

$$|y_i - f(x_i)| \leq \varepsilon + \xi_{(m+i)}, \quad i = i, ..., \ell. \tag{32}$$

and constraints

$$|\Phi_s^T F(f) - \Phi_s^T Y| \leq \ell\varepsilon_s + \xi_s, \quad s = 1, ..., m, \tag{33}$$

where $\varepsilon_s > 0$, ε, and τ are free parameters of algorithm.

Approximation 3. Hinge Loss Constrained Minimization. Here we consider the problem of minimization in a given set of functions the functional (31) subject to constraints (32) and constraints

$$(2y_i - 1)(f(x_i) - 0.5) \geq \varepsilon - \xi_{(m+i)}, \quad i = 1, \ldots, \ell, \tag{34}$$

where $(2y_i - 1) \in \{-1, 1\}$. Note that $f(x)$ in (34) is an estimate of conditional probability function and is an estimate of optimal classification rule (10).

In this paper, we solve complete learning problem in two sets of functions:

1) The set of functions that belong to Reproducing Kernel Hilbert Space (RKHS) of kernel $K(x, x')$. For this set of functions, we obtain closed-form solutions for both exact constrained minimization method and approximate unconstrained minimization method.

2) The set of functions defined by a neural network. For this set of functions, we obtain solution for L_2 unconstrained minimization approximation.

6. Complete Solution in a Set of RKHS

In this section, we present solution of complete learning problem in a set of functions that belong to RKHS.

6.1. Reproducing Kernel Hilbert Space

We are looking for solutions of our inference problems in the set of functions $f(x,\alpha), \alpha \in \Lambda$ that belong to Reproducing Kernel Hilbert Space associated with kernel $K(x,x')$, where $K(x,x')$ is a continuous positive semi-definite function of variables $x, x' \in X \subset R^n$:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} K(x_i,x_j)c_i c_j \geq 0 \tag{35}$$

for any $\{x_1,\ldots,x_n\}$ and $\{c_1,\ldots,c_n\}$. Consider linear operator

$$Af = \int_a^b K(x,s)f(s)\,ds \tag{36}$$

that maps elements $f(s)$ into elements $Af(x)$ in space H.

According ro Mercer theorem, for any continuous positive semi-definite kernel, there exists an orthonormal basis $e_i(x)$ consisting of eigenfunctions of $K(x,x')$ of operator (36), and the corresponding sequence of nonnegative eigenvalues λ_i such that kernel $K(x,x')$ has the representation

$$K(x,x') = \sum_{i=1}^{\infty} \lambda_i e_i(x) e_i(x'), \tag{37}$$

where the convergence of the sequence is absolute and uniform.

The set $\{f(x)\}$ of functions $f(x)$ belongs to Reproducing Kernel Hilbert Space (RKHS) associated with kernel $K(x,x')$ if the inner product $(f_1,f_2)_{\mathcal{H}}$ between functions f_1, and f_2 of this set is such that for any function $f(x) \in \{f(x)\}$ the equality

$$f(x') = (K(x,x'),f(x))_{\mathcal{H}} \tag{38}$$

holds true. That is, the inner product of functions from $\{f(x)\}$ with kernel $K(x,x')$ (where variable x' is fixed) has the reproducing property.

Consider the parametric set $\{f(x)\}$ of functions

$$f_c(x) = \sum_{i=1}^{\infty} c_i e_i(x), \quad c = (c_1,c_2,\ldots)^T \in R^{\infty}. \tag{39}$$

According to representation (37), kernel $K(x,x')$ as a function of variable x belongs to set $\{f(x)\}$ (the values $\lambda_i\phi_i(x')$ can be considered as parameters c_i of expansion.)

In order to define Reproducing Kernel Hilbert Space for set (39), we introduce the following inner product between two functions $f_b(x)$, $f_d(x)$, defined by parameters $b = (b_1,b_2,\ldots)$ and $d = (d_1,d_2,\ldots)$ in (39):

$$(f_b(x),f_d(x))_{\mathcal{H}} = \sum_{i=1}^{\infty} \frac{b_i d_i}{\lambda_i}. \tag{40}$$

It is easy to check that, for such inner product, reproducing property (38) of functions from RKHS of kernel $K(x,x')$ holds true and the square of the norm of function $f_b(x) \in \Phi$ is equal to

$$||f_b(x)||^2_{\mathcal{H}} = (f_b(x),f_b(x))_{\mathcal{H}} = \sum_{i=1}^{\infty} \frac{b_i^2}{\lambda_i} = B. \tag{41}$$

6.1.1. Properties of RKHS.

The following three properties of functions from RKHS of kernel $K(x, x')$ make them useful for function estimation problems in high-dimensional spaces:

1. Functions from RKHS with bounded square of norms

$$\sum_{i=1}^{\infty} \frac{b_i^2}{\lambda_i} \le C \tag{42}$$

belong to a compact set and therefore the square of the norm of function can be used as a regularization functional (see Lemma in Section 3.2).

2. (*Representer Theorem*) The function that minimizes the empirical loss

$$R(f) = \sum_{i=1}^{\ell} L(y_i - f(x))$$

in a set of RKHS with bounded norm (42), along with representation (39), has the representation

$$f(x, \alpha) = \sum_{i=1}^{\ell} \alpha_i K(x_i, x), \tag{43}$$

where ℓ is the number of observations.

3. The square of the norm of the chosen function, along with representation (42), has the representation

$$||f(x, \alpha)||_{\mathcal{H}}^2 = (f(x, \alpha), f(x, \alpha))_{\mathcal{H}} = \sum_{i,j=1}^{\ell} \alpha_i \alpha_j K(x_i, x_j). \tag{44}$$

Representation (43) of the function from RKHS and its norm (44) is used to solve inference problems in high-dimensional spaces.

6.1.2. Properties of Kernels.

Kernels $K(x, x')$ (also called *Mercer kernels*) have the following properties:

(1) Linear combination of kernels $K_1(x, x')$ and $K_2(x, x')$ with non-negative weights is the kernel

$$K(x, x') = \alpha_1 K_1(x, x') + \alpha_2 K_2(x, x'), \quad \alpha_1 \ge 0, \alpha_2 \ge 0.$$

(2) Product of the kernels $K_1(x, x')$ and $K_2(x, x')$ is the kernel

$$K(x, x') = K_1(x, x') K_2(x, x').$$

In particular, the product of kernels $K(x^k, x'^k)$ defined on coordinates x^k of vectors $x = (x^1, \ldots, x^m)$ is a multiplicative kernel in m-dimensional vector space $x \in R^m$:

$$K(x, x') = \prod_{s=1}^{m} K_s(x^s, x'^s).$$

(3) Normalized kernel is the kernel

$$K_*(x,x') = \frac{K(x,x')}{\sqrt{K(x,x)K(x',x')}}.$$

6.1.3. Examples of Mercer Kernels.

Gaussian kernel in $x \in R^1$ has the form

$$K(x,x') = \exp\{-\delta(x-x')^2\}, \quad x,x' \in R^1,$$

where $\delta > 0$ is a free parameter of the kernel.

In m-dimensional space $x \in R^m$, Gaussian kernel has the form

$$K(x,x') = \prod_{k=1}^{m} \exp\{-\delta(x^k - x'^k)^2\} = \exp\{-\delta|x-x'|^2\}, \quad x,x' \in R^n.$$

6.2. Exact Solution of Complete Learning Problem in RKHS

In this section, we estimate the conditional probability function in the form

$$f(x) = \psi(x) + c, \tag{45}$$

where $\psi(x)$ belongs to RKHS of kernel $K(x,x')$ and $b \in R^1$ is the bias.

In order to do this, we minimize $\mathcal{V}$-quadratic form (27) in a given set of functions $\{f(x)\}$ subject to constraints (28). Below we look for a solution in the set of functions from RKHS of kernel $K(x,x')$ with bias, using the representation

$$f(x) = \sum_{i=1}^{\ell} a_i K(x_i,x) + c. \tag{46}$$

We consider functions from $RKHS$ that have their norm bounded by value B:

$$||f(x)||^2_{\mathcal{H}} = \left|\left|\sum_{i=1}^{\ell} a_i K(x_i,x)\right|\right|^2_{\mathcal{H}} = \sum_{i,j=1}^{\ell} a_i a_j K(x_i,x_j) \le B. \tag{47}$$

We introduce vector $A = (a_1,\ldots,a_\ell)^T$ and rewrite (47) in the form[5]

$$||f(x)||^2_{\mathcal{H}} = A^T K A \le B. \tag{48}$$

Vector $F(f)$, which defines estimates of conditional probability function on training data $x_1,\ldots,x_\ell$ for RKHS, has the form

$$F(f) = KA + 1_\ell c,$$

where $1_\ell = (1,\ldots,1)^T$ is ℓ-dimensional vector of ones.

5. Note that the functions satisfying (41) constitute the set of smooth functions where smoothness properties are controlled by the value B: the smaller is the value B, the smoother are the functions.

In order to estimate the conditional probability function from RKHS with bounded norm, we minimize target functional (27) subject to constraints (28) written in the form defined for RKHS.

The target functional 27) for RKHS has the form

$$R(A) = (KA + c1_\ell - Y)^T \mathcal{V}(KA + c1_\ell - Y), \tag{49}$$

and constraints (28) has the form

$$\Phi_s^T KA + c\Phi_s^T 1_\ell = \Phi_s^T Y, \quad s = 1, \ldots, m. \tag{50}$$

In order to solve the complete learning problem problem for RKHS, we have to minimize functional (49) subject to constraints (50) and (48). Consider the Lagrangian

$$L(A, c, \mu) = (KA + c1_\ell - Y)^T \mathcal{V}(KA + c1_\ell - Y) + \gamma(A^T KA - B) + \tag{51}$$
$$\sum_{k=1}^{m} \mu_k(\Phi_s^T KA + c\Phi_s^T 1_\ell - \Phi_s^T Y).$$

The necessary conditions of its minimum are as follows:

$$\begin{aligned}
&\frac{\partial L(A, c, \mu)}{\partial A} \implies \mathcal{V}KA + \gamma A + cV1_\ell - \mathcal{V}Y + \sum_{s=1}^{m} \mu_s \Phi_s = 0 \\
&\frac{\partial L(A, c, \mu)}{\partial c} \implies 1_\ell^T \mathcal{V}KA + 1_\ell^T \mathcal{V}1_\ell c - 1_\ell^T \mathcal{V}Y + \sum_{s=1}^{m} \mu_s 1_\ell^T \Phi_s = 0 \\
&\frac{\partial L(A, c, \mu)}{\partial \mu_k} \implies A^T K\Phi_k + c1_\ell^T \Phi_k - Y^T \Phi_k = 0, \quad k = 1, \ldots, m.
\end{aligned} \tag{52}$$

From the first line of (52), we obtain the expression

$$(VK + \gamma I)A = \mathcal{V}Y - c\mathcal{V}1_\ell - \sum_{k=1}^{m} \mu_k \Phi_k \tag{53}$$

and the expression

$$A = (VK + \gamma I)^{-1}(\mathcal{V}Y - c\mathcal{V}1_\ell - \sum_{s=1}^{m} \mu_s \Phi_s). \tag{54}$$

We then compute $(m + 2)$ vectors

$$\begin{aligned}
A_\mathcal{V} &= (\mathcal{V}K + \gamma I)^{-1}\mathcal{V}Y, \\
A_c &= (\mathcal{V}K + \gamma I)^{-1}\mathcal{V}1_\ell \\
A_s &= (\mathcal{V}K + \gamma I)^{-1}\Phi_s, \quad s = 1, \ldots, n.
\end{aligned} \tag{55}$$

The desired vector A has the expression

$$A = A_\mathcal{V} - cA_c - \sum_{s=1}^{m} \mu_s A_s. \tag{56}$$

Putting expression (56) back into the last two lines of (52), we find that, in order to compute coefficient c and n coefficients μ_s of expansion (56), we have to solve the following system of $m+1$ linear equations:

$$\begin{aligned} c[1_\ell^T \mathcal{V} K A_c - 1^T \mathcal{V} 1_\ell] + \sum_{s=1}^{m} \mu_s [1_\ell^T \mathcal{V} K A_s - 1_\ell^T \Phi_s] &= [1_\ell^T \mathcal{V} K A_{\mathcal{V}} - 1_\ell^T \mathcal{V} Y] \\ c[A_c^T K \Phi_k - 1_\ell^T \Phi_k] + \sum_{s=1}^{m} \mu_s A_s^T K \Phi_k &= [A_V^T K \Phi_k - Y^T \Phi_k], \quad k = 1, \ldots, m. \end{aligned} \tag{57}$$

Using estimated vector A and bias c, we obtain the desired function

$$f(x) = A^T \mathcal{K}(x) + c. \tag{58}$$

Parameter γ is a free parameter in the algorithm. It depends on the selected value B in (48): γ is selected in such a way that equality $A^T K A = B$ holds.

6.3. Approximation 1. L_2 Conditional Minimization

In order to find vector A and bias c of of approximation (58), we minimize functional (30) in the set of functions (48). That is, we minimize

$$R(A) = (KA + c1_\ell - Y)^T (\widehat{\tau}\mathcal{V} + \tau\mathcal{P})(KA + c1 - Y)$$

subject to constraint

$$A^T K A \le B.$$

Consider Lagrangian

$$R(A) = (KA + c1_\ell - Y)^T (\widehat{\tau}\mathcal{V} + \tau\mathcal{P})(KA + c1 - Y) + \gamma(A^T K A - B).$$

From the necessary conditions of minima of this Lagrangian

$$K(\widehat{\tau}\mathcal{V} + \tau\mathcal{P})(KA + c1 - Y) + \gamma K A = 0,$$

$$1_\ell^T (\widehat{\tau}\mathcal{V} + \tau\mathcal{P})(KA + c1_\ell - Y) = 0,$$

we obtain vector A and value b of the solution (58):

$$A = ((\widehat{\tau}\mathcal{V} + \tau\mathcal{P})K + \gamma I)^{-1} (\widehat{\tau}\mathcal{V} + \tau\mathcal{P})(Y - c1_\ell)$$

and

$$c = \frac{1_\ell^T (\widehat{\tau}\mathcal{V} + \tau\mathcal{P}) Y_r}{1_\ell^T (\widehat{\tau}\mathcal{V} + \tau\mathcal{P}) 1_\ell}. \tag{59}$$

6.4. Approximation 2. LUSI-Regression SVM

To simplify the formulas, we denote, along with m predicate vectors

$$\Phi_s = (\phi_s(x_1), \ldots, \phi_s(x_\ell))^T, \quad s = 1, \ldots, m,$$

ℓ vectors that indicate the indices i of the training vectors x_i:

$$\Phi_{m+i} = (0, ..., 0, 1, 0, ..., 0)^T, \quad i = 1, ..., \ell.$$

We also set $\tau_s = 1 - \tau$ for $1 \le s \le m$ and set $\tau_s = \tau$ for $s > m$.

Using these notations, consider Approximation 2 based on L_1 unconstrained minimization problem (see Section 5): In order to find parameters A and b of approximation (58), we minimize functional

$$R = A^T K A + C \sum_{s=1}^{m+\ell} \tau_s \xi_s \tag{60}$$

subject to constraints

$$|\Phi_s^T (KA + c1 - Y)| \le \varepsilon_s + \xi_s, \quad s = 1, ..., (m + \ell), \quad \xi_s \ge 0, \tag{61}$$

For $s = (m + 1), ..., (m + \ell)$, let ε_s be a small value $\varepsilon_s = \varepsilon_*$; for $s = 1, \ldots, m$ let ε_s be the value

$$\varepsilon_s = c\sqrt{\ell \ln \eta/2},$$

which is proportional to least square deviation (29) (see Remark 6 in Section 5).

Consider the Lagrangian

$$L(A, b, \xi) = A^T K A + C \sum_{s=1}^{m+\ell} \tau_s \xi_s + \sum_{s=1}^{m+\ell} \nu_s \xi_s + \tag{62}$$

$$\sum_{s=1}^{m+\ell} \alpha_s [-(\varepsilon_s + \xi_s) + \Phi_s^T (KA + c1 - Y)] - \sum_{s=1}^{m+\ell} \beta_s [(\varepsilon_s + \xi_s) + \Phi_s^T (KA + c1 - Y)],$$

where $\gamma \ge 0, \alpha \ge 0, \beta \ge 0, \nu \ge 0$ are Lagrange multipliers.

From necessary conditions of minima of this Lagrangian, we obtain

$$\begin{aligned}
&\frac{\partial L}{\partial A} = 0 \Longrightarrow A = \sum_{s=1}^{m+\ell} (\alpha_s - \beta_s) \Phi_s \\
&\frac{\partial L}{\partial c} = 0 \Longrightarrow \sum_{s=1}^{m+\ell} (\alpha_s - \beta_s) \Phi_s^T 1 = 0 \\
&\frac{\partial L}{\partial \xi_s} = 0 \Longrightarrow \quad 0 \le \alpha_s, \beta_s < C(1 - \tau), \quad s = 1, ..., m \\
&\frac{\partial L}{\partial \xi_s} = 0 \Longrightarrow \quad 0 \le \alpha_s, \beta_s < C\tau, \quad s = (m + 1), ..., (m + \ell)
\end{aligned} \tag{63}$$

Putting the expression for A back into Lagrangian and taking into account (63), we obtain that, in order to minimize the Lagrangian over α and β and ξ, we have to maximize the quadratic form

$$R(\alpha,\beta) = -\sum_{s=1}^{m+\ell}\varepsilon_s(\alpha_s+\beta_s) + \sum_{s=1}^{m+\ell}(\alpha_j-\beta_j)\Phi_s^T Y - \frac{1}{2}\sum_{s,r=1}^{m+\ell}(\alpha_s-\beta_s)\Phi_s^T K\Phi_s(\alpha_r-\beta_r) \quad (64)$$

subject to the constraints

$$\sum_{s=1}^{m+\ell}(\alpha_s-\beta_s)\Phi_s^T 1 = 0 \quad (65)$$

$$\begin{aligned} &0 \le \alpha_s,\beta_s \le C(1-\tau), \quad s=1,...,m \\ &0 \le \alpha_s,\beta_s \le C\tau, \quad s=(m+1),\ldots,(m+\ell). \end{aligned} \quad (66)$$

6.5. Approximation 3. LUSI-Classification SVM

Consider, along with the values $y_i \in \{0,1\}$, the values $\widehat{y}_i = (2y_i-1) \in \{-1,1\}$). In order to minimize functional

$$R(f) = A^T KA + C\sum_{s=1}^{m+\ell}\tau_s\xi_s, \quad \xi_s \ge 0 \quad (67)$$

subject to constraints

$$\widehat{y}_i(A^T K\Phi_{(m+i)} + c - 0.5) \ge \varepsilon_* - \xi_{(m+i)}, \quad i=i,...,\ell, \quad (68)$$

(which is the SVM idea for constructing classification rule) and the constraints

$$-(\varepsilon_s+\xi_s) \le \Phi_s^T KA + c\Phi_s^T 1_\ell - \Phi_s^T Y \le (\varepsilon_s+\xi_s), \quad s=1,\ldots,m. \quad (69)$$

Consider Lagrangian

$$L(\alpha,\beta,\delta) = A^T KA + C\sum_{s=1}^{m+\ell}\tau_s\xi_s + \sum_{s=1}^{m+\ell}\nu_s\xi_s$$

$$+\sum_{s=1}^{m}\beta_s[\Phi_s^T(KA+c1-Y) - (\varepsilon_s+\xi_s)] - \sum_{s=1}^{m}\alpha_s[(\varepsilon_s+\xi_s) + \Phi_s^T(KA+c1-Y)] -$$

$$\sum_{i=1}^{\ell}\delta_i[\widehat{y}_i(\Phi_{(m+i)}^T KA + c - 0.5) - \varepsilon_* + \xi_{(m+i)}].$$

From necessary conditions of minima of Lagrangian, we obtain

$$\begin{aligned}
&\frac{\partial L}{\partial A}=0\Longrightarrow A=\sum_{s=1}^{m}(\alpha_s-\beta_s)\Phi_s+\sum_{i=1}^{\ell}\widehat{y}_i\delta_i\Phi_{m+i}\\
&\frac{\partial L}{\partial c}=0\Longrightarrow\sum_{s=1}^{m}(\alpha_s-\beta_s)\Phi_s^T1+\sum_{i=1}^{\ell}\widehat{y}_i\delta_i=0\\
&\frac{\partial L}{\partial \xi_s}=0\Longrightarrow\quad 0\le\alpha_s,\beta_s<C(1-\tau),\quad s=1,...,m\\
&\frac{\partial L}{\partial \xi_s}=0\Longrightarrow\quad 0\le\delta_i<C\tau,\quad s=(m+1),...,(m+i),...,(m+\ell)
\end{aligned}\tag{70}$$

Putting the expression for A back into Lagrangian and taking into account (70), we obtain that, in order to minimize the Lagrangian over α,β,δ and ξ, we have to maximize the quadratic form

$$\begin{aligned}
R(\alpha,\beta,\delta)=-\sum_{s=1}^{m}\varepsilon_s(\alpha_s+\beta_s)+\varepsilon_*\sum_{i=1}^{\ell}\delta_i+\sum_{s=1}^{m}(\alpha_j-\beta_j)\Phi_s^TY+\frac{1}{2}\sum_{i=1}^{\ell}\widehat{y}_i\delta_i-\qquad(71)\\
\frac{1}{2}\left(\sum_{s,r=1}^{m}(\alpha_s-\beta_s)\Phi_s^TK\Phi_s(\alpha_r-\beta_r)+2\sum_{s=1}^{m}\sum_{i=1}^{\ell}(\alpha_s-\beta_s)\Phi_s^TK\Phi_{m+i}\delta_i\widehat{y}_i+\right.\\
\left.\sum_{i,j=1}^{\ell}\widehat{y}_i\delta_i\Phi_{m+i}^TK\Phi_{m+j}\delta_j\widehat{y}_j\right)
\end{aligned}$$

subject to the constraints

$$\begin{aligned}
&\sum_{s=1}^{m}(\alpha_s-\beta_s)\Phi_s^T1+\sum_{i=1}^{\ell}\widehat{y}_i\delta_i=0\\
&0\le\alpha_s,\beta_s\le C(1-\tau),\quad s=1,\ldots,m\\
&0\le\delta_i\le C\tau,\quad i=1,\ldots,\ell.
\end{aligned}\tag{72}$$

6.5.1. Discussion

1) LUSI Classification SVM method (4) is a reinforcement (using invariant constraints) of the standard SVM method. Indeed, if$\tau=1$, then parameters $\alpha_s=\beta_s=0$ for $s=1,\ldots,m$ (see (72)), $\Phi_s^TK\Phi_r=K(x_i,x_j)$, and $\Phi_i^T1=1$. As a result, we obtain that the desired function has the form

$$f(x)=\sum_{i=1}^{\ell}\delta^*\widehat{y}_iK(x_i,x)+b.$$

To find the parameters of expansion δ^* one has to maximize the functional (71) which, for our particular case, has the form

$$R(\delta^*)=\varepsilon_*\sum_{i=1}^{\ell}\delta_i^*-\frac{1}{2}\sum_{i,j=1}^{\ell}\widehat{y}_i\delta_i^*K(x_i,x_j)\delta_j^*\widehat{y}_j\tag{73}$$

subject to constraint

$$\sum_{j=1}^{\ell} \widehat{y}_i \delta_i^* = 0,$$

and constraints

$$0 \le \delta_i^* \le C^*, \quad i = 1, ..., \ell..$$

Changing the scaling factor for parameters $\delta_i^* = \varepsilon^* \widehat{\delta}_i$, we obtain that, in order to find the desired approximation

$$f(x) = \sum_{i=1}^{\ell} \widehat{\delta}_i \widehat{y}_i K(x_i, x) + b,$$

one has to maximize the functional

$$R(\widehat{\delta}) = \sum_{i=1}^{\ell} \widehat{\delta}_i - \frac{1}{2} \sum_{i,j=1}^{\ell} \widehat{y}_i \widehat{\delta}_i K(x_i, x_j) \widehat{\delta}_j \widehat{y}_j$$

subject to the constraints

$$\sum_{j=1}^{\ell} \widehat{y}_i \widehat{\delta}_i = 0,$$

and constraints

$$0 \le \widehat{\delta}_i \le \frac{C^*}{\varepsilon^*} = \widehat{C}.$$

($\widehat{C}$ is a parameter of the algorithm). This is the standard SVM estimate of parameters for pattern recognition problem.

2) When $0 \le \tau \le 1$, the obtained solution implements both weak and strong modes of convergence. Parameter τ defines the balance between these two modes.

Calibration for Multiclass Classification Problems

Consider n-class LUSI classification problems. For such problems, the values y_i in training set can take one of n values $y_i = 1, \ldots, n$. In order to solve an n-class classification problem, we estimate n conditional probability functions $P(y = p|x),\ p \in \{1, \ldots, n\}$. Using the obtained estimates, we construct the rule

$$r(x) = \text{argmax}\{P_\ell(y = 1|x), \ldots, P_\ell(y = n|x)\}. \tag{74}$$

In order to do this, we consider n different pattern recognition problem defined by the training set (x_i, y_i) (where y_i take n values). For each of these problems, we consider the training set

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

where

$$y_i^s = \begin{cases} y_i^s = 1, & \text{if } y_i = s \\ 0, & \text{otherwise.} \end{cases}$$

We denote by Y_s the vector of values $(y_1^s, \ldots, y_\ell^s)$.

Suppose now that we estimate n conditional probability functions using Approximation 1 method in the same RKHS (defined by the same kernel function $K(x, x')$), using the same set of predicate functions $\phi_1(x), \ldots, \phi_m(x)$ and the same regularization parameter γ. The estimates of conditional probability $P(y = s|x)$ have the form

$$f_s(x) = A_s^T \mathcal{K}(x) + c_s, \quad s = 1, \ldots, n.$$

In order to find n conditional probability functions, one has to estimate n pairs (A_s, c_s). Using training sets (x_i, y_i^s), $s = 1, \ldots, n$, we estimate n conditional probability functions as described above.

Since matrices K and $\mathcal{P}$ do not depend on s (class $y_s^* = s$, for which we estimate the conditional probability), and

$$\sum_{p=1}^{n} Y_p = 1_\ell$$

from (59) we obtain that

$$\sum_{p=1}^{n} c_p = 1, \qquad \sum_{p=1}^{n} A_p = 0_\ell$$

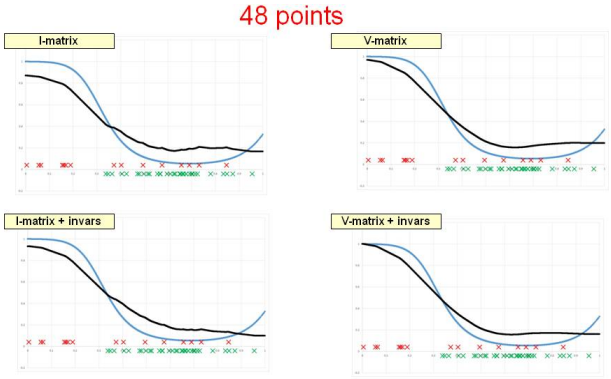
That is,

$$\sum_{p=1}^{n} P_\ell(y = p|x) = 1$$

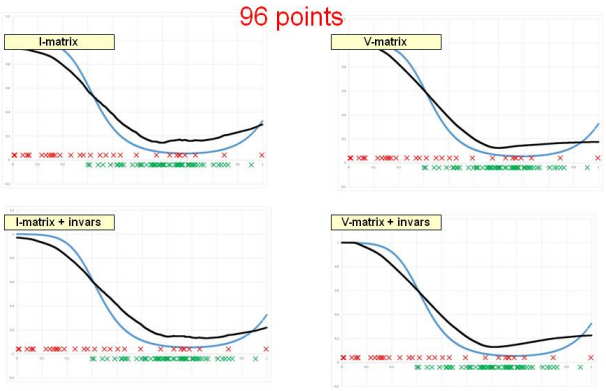
for all $x \in X$. In other words, n unconditional minimization solutions form n jointly calibrated solutions.

6.5.2. Illustrations

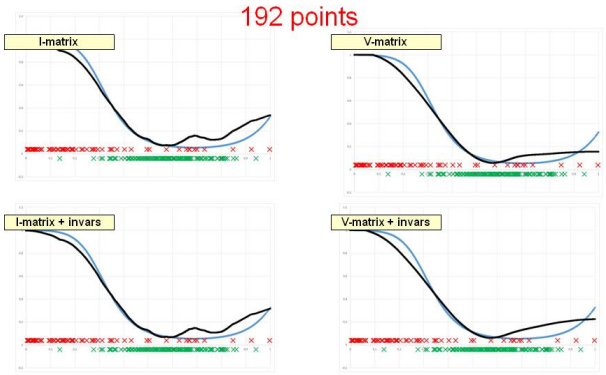
The following illustrations show conditional probability functions (the dashed lines correspond to the same ground truth function across all the cases, and the solid lines correspond to various approximation to that ground truth function) for four algorithms: (1) Least square method (I-matrix method), (2) V-matrix method, (3) I-matrix +Invariants method, and (4) V-matrix+Invariant method; the illustrations demonstrate the accuracy of all these methods for different sizes ℓ of the training set.



I: 0.3756 **V: 0.1432**
$I\&\mathcal{I}$: 0,2166 **$V\&\mathcal{I}$: 0.1017**



I: 0.3212 **V: 0.1207**
$I\&\mathcal{I}$: 0.1808 **$V\&\mathcal{I}$: 0.0778**



I: 0.1672 **V: 0.0689**
$I\&\mathcal{I}$: 0.1072 **$V\&\mathcal{I}$: 0.0609**

6.6. LUSI Using Neural Networks

Neural Networks implements smoothed piecewise linear set of functions $\{f(x)\}$,. constructed as a combination of neurons described by function

$$u = g((w, x) + b), \quad x, w \in R^n,$$

where w is vector of weight parameters, b is bias, and $g(z)$ is a nonlinear function. For example, $g(x)$ can be threshold function $g(z) = \theta(z)$, smoothed threshold function $g(z) = (1 + \exp(-z))^{-1}$, hinge function $g(z) = \max(0, z)$ and so on.

Neural Network consists of several layers of neurons. The initial layer (layer number zero) is the input vector $x_i(0) = (x_i^1(0), \ldots, x_i^n(0))^T$ of dimensionality n. Using n_1 neurons (with n different parameters w, b and the same function $g(u)$) the n-dimensional input vector $x_i(0)$ is transformed into n_1-dimensional vector $x_i(1)$ of the next layer and so on.

We denote parameters $w_1, \ldots, w_n$ of transformation vectors $x_i(0)$ of the initial layer into vectors $x_i(1)$ of the next level by $(n \times n_1)$-dimensional matrix $W(1)$.

Generally, n_k-dimensional vectors $x_i(k)$ of layer k is transformed into n_k-dimensional vectors $x_i(k+1)$ of layer $(k+1)$ using $(n_k \times n_{k+1})$-dimensional matrix $W(k) = (w_1(k), ..., w_{n_k}(k))$ (every column in the matrix describes weights w of one neuron of level k).

The last layer N of network transform vectors of layer $x_I(N-1)$ into scalar $x_i(N)$ using n_N-dimensional vector of weights w_N.

Learning using Neural Networks requires to find such weights matrix $W = (w_1, \ldots, w_N)$ of the network which minimizes the functional

$$R(f) = (Y - F(f))^T (Y - F(f)) = \sum_{i=1}^{\ell} (y_i - f(x_i))^2, \tag{75}$$

where (x_i, y_i) are elements of training set. Using the notation $X(N) = (x_1(N), \ldots, x_\ell(N))^T$ we can rewrite rewrite in (75) vector $F(f)$ as

$$F(f) = X(N) = (x_1(N), \ldots, x_\ell(N))^T$$

6.6.1. Backpropagation Method

Minimization (75) in the set of functions given by the Neuaral Network can be described as a constrained optimization problem. Indeed, our goal is to minimize

$$R = (Y - X(N))^T (Y - X(N)) \tag{76}$$

in the set of functions given by construction of the network. The construction can be described as follows: for all $i = 1, \ldots, \ell$ and all k the equalities

$$x_i(k) = G([W(k)x_i(k-1)]) \tag{77}$$

hold true, where we using the notation

$$G([W(k)x_i(k-1)]) = (g(W_1^T(k)x_i(k-1)), \ldots, g(W_{n_{(k-1)}}^T(k)x_i(k-1)))^T.$$

For $k = 1$, vectors $x_i(0)$ are elements of training data.

In order to minimize functional (76) subject to constraints (77), consider Lagrangian

$$L(W, X, B) = (X(N) - Y)^T (X(N) - Y) + \sum_{i=1}^{\ell} \sum_{k=1}^{N-1} B_i(k)(x_i(k) - G([W(k)x_i(k-1)])),$$

where $B_i(k)$ are Lagrange multipliers. Conditions of minima describe three subconditions

$$\frac{\partial L(W, X, B)}{\partial B_i(k)} = 0, \quad \frac{\partial L(W, X, B)}{\partial x_i(k)} = 0, \quad \frac{\partial L(W, X, B)}{\partial W(k)} = 0.$$

The first subcondition can be decomposed in $\ell \times N$ conditions

$$x_i(k) = G(\{W(k)x_i(k-1)\}), \quad i = 1, ..., \ell, \ k = 1, \ldots, N.$$

The second subcondition can be split into two cases: the case $k = N$ and the case $k \neq N$.

The case $k = N$ defines the boundary condition, that define vector $B(N)$ as:

$$B(N) = 2(Y - X(N)). \tag{78}$$

The case $k \neq N$ leads to the chain

$$B_i(k) = B_i(K+1)W^T(k+1)G'\{W(k)x_i(k))\}B_i(k+1), \quad k = N-1, \ldots, 1,$$

where we used the notation

$$G'\{W(k)x_i(k)\} = \{g'(W_1(k)x_i(k-1)), \ldots, g'_{n_k}(W_{n_k}(k-1)x(k-1))\}^T,$$

The third subcondition requires to find a stationary point with respect to W. This is equivalent to finding a minimum of Lagrangian $L(W, X, W)$, while satisfying the first two subconditions. In order to find a minimum with respect to $W(k)$, the gradient descent procedure

$$W(k) \longleftarrow W(k) - \lambda \frac{\partial L(W, X, B)}{\partial W(k)}$$

is used.

6.7. LUSI Learning: $\mathcal{VP}$-Neural Networks

In this section, we show that, using a slightly modified standard learning procedure for Neural Networks, one can minimize the functional

$$R_*(f) = (Y - F(f))^T(\widehat{\tau}\mathcal{V} + \tau\mathcal{P})(Y - F(f)) \tag{79}$$

(rather than functional (75)) where $(\widehat{\tau}\mathcal{V}+\tau\mathcal{P})$ matrix takes into account statistical invariants (see Section 6.3). We call Neural Network that minimizes (79) $\mathcal{VP}$-Neural Network.

To modify the Neural Network method, we rewrite the problem of minimizing functional (79) in the set of piecewise linear functions as the problem of minimizing the functional

$$R_* = (Y - X(N))^T(\widehat{\tau}\mathcal{V} + \tau\mathcal{P})(Y - X(N))$$

subject to constraints (77). We construct the modified Lagrangian

$$L^*(W, X, B) =$$

$$(X(N) - Y)^T(\widehat{\tau}\mathcal{V} + \tau\mathcal{P})(X(N) - Y) + \sum_{i=1}^{\ell}\sum_{k=1}^{N-1} B_i(k)(x_i(k) - G([W(k)x_i(k-1)])).$$

In order to find the minimum of this Lagrangian, we use the same reasoning as in Section 5.2. This brings us to the same procedure as in the standard Neural Network with just one correction: In the first case of the second subcondition, we have to replace the border conditions (78) with the modified boundary condition

$$B(N) = 2(\widehat{\tau}\mathcal{V} + \tau\mathcal{P})(Y - X(N)). \tag{80}$$

Remark 7. The Lagrangian that define classical Neural Networks have many local minima and Neural Networks use several heuristics to find a local minimum of the Lagrangian, which is close to the smallest possible.

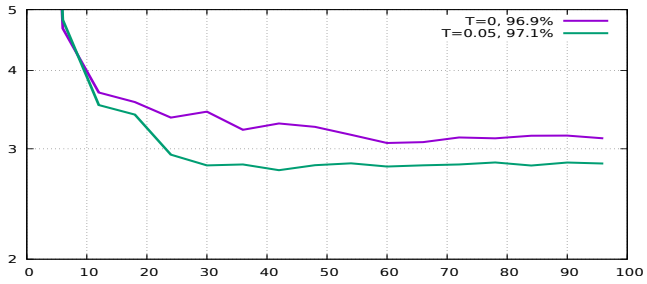
The same remains true for $\mathcal{VP}$-NN: functional (79) also has many local minima; in order to find a small one, $\mathcal{VP}$-NN uses the same heuristics as standard Neural Networks.

6.7.1. Illustration

Below we present the solution of MNIST digit recognition problem using Deep and $\mathcal{VP}$ Neural Networks constructed by Igor Durdanovich from Princeton NEC Research based on the state-of-the-art DNN of NEC.

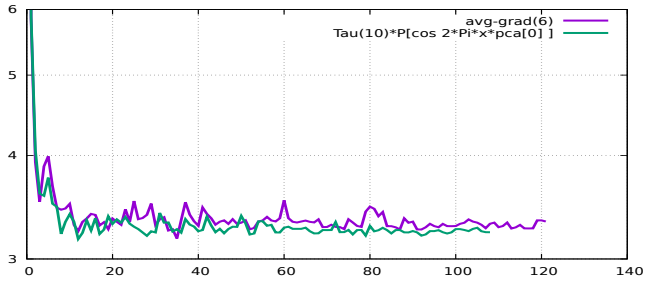
Figures below show rate of convergence depending on number of epoch for DNN and for $\mathcal{IP}$-DNN. In all experiments 1,000 observations (100 per class) and modified back propagation method with batch 6 were used. For simplicity, instead of $\mathcal{V}$ matrix we used $\mathcal{I}$-matrix.

1 Predicate: $\phi(u_i) = 1$.



Error rate: DNNet – 3.1%, $\mathcal{VP}$-NNet – 2.9%

2. Predicate: $\phi(u_i) = \int_0^1 u_i(x^1, x^2) \cos 2\pi x^1$.

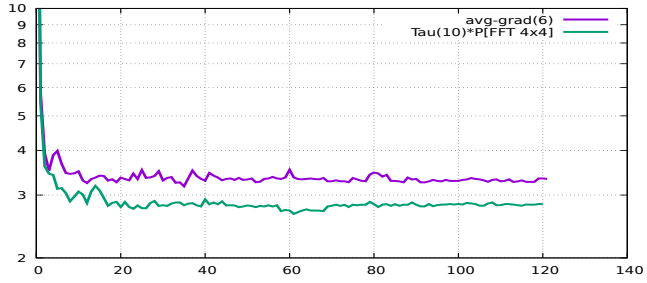


Error rate: DNNet – 3.4%, $\mathcal{VP}$-NNet – 3.3%

3. 16 predicates $(m, n = 1, .., 4)$.

$$\phi_{m,n}(u_i) = \int_0^1 u_i(x^1, x^2) \cos m\pi x^1 \cos n\pi x^2 dx^1 dx^2, \quad m, n = 1, 2, 3, 4.$$

.



Error rate: DNNet – 3.4%, $\mathcal{VP}$-NNet – 2.8%

7. Examples of Predicates

Before introducing examples of predicates, we make the following important remark.

> The concept of predicates is very different from the concept of features used in the classical machine learning. Indeed, with the increasing number of features, the capacity (the VC dimension) of the set of admissible functions constructed using these features *increases* while with the increasing number of invariants (predicates), the capacity of admissible set *decreases*. In the extreme case, when the set of predicates contains all the functions from L_2, the admissible set consists of just one function, the desired one.

There exist two type of predicates: (i) general predicates, which take into account only purely mathematical concepts and (ii) special predicates, which take into account specific properties of the data (for instance, predicates for 2D images recognition can take into account existing understanding of mechanisms of images construction). In this section, we consider examples of both types.

7.1. Examples of General Predicates

1. Predicate $\phi(x) = 1$. This predicate leads to the invariant

$$\sum_{i=1}^{\ell} f(x_i) = \sum_{i=1}^{\ell} y_i, \quad (y_i \in \{1, 0\}.$$

This predicates restricts the functions in the admissible set to those for which the frequency of expected examples of the first class (with $y = 1$) is equal to their frequency observed in the training data.

2. Predicates $\phi(x) = x$ (here $x \in R^n$). This predicate provides n invariants

$$\sum_{i=1}^{\ell} x_i f(x_i) = \sum_{i=1}^{\ell} y_i x_i$$

(invariants are given coordinate-wise). These invariants require that, for any function from admissible set of conditional probabilities, the expectation of the center of mass to be equal to the center of mass of vectors of class $y = 1$ in training data.

3. Predicate $\phi(x) = xx^T$. This predicates form $n(n+1)/2$ invariants

$$\sum_{i=1}^{\ell} x_i x_i^T f(x_i) = \sum_{i=1}^{\ell} y_i x_i x_i^T$$

(equalities are considered element-wise). Expectation of covariance matrix computed using any function from the admissible set is equal to the corresponding matrix computed on training set.

4. Predicates $\rho_s(|x - x_s|)$ (for example, $\rho(u) = u^{-\delta}$ or $\rho(u) = e^{-au}$) define function of distance from vector x to a given vector x^s. They lead to the invariants

$$\sum_{i=1}^{\ell} \rho(|x_i - x^s|) f(x_i) = \sum_{i=1}^{\ell} y_i \rho(|x_i - x^s|).$$

Choosing different vectors x^s, one obtains different predicates $\rho(x - x^s)$. Any predicate $\rho_s(|x - x^s|)$ leads to the invariant that selects the admissible set of conditional probability functions with the same local (in the vicinity of x^s) characteristic.

7.2. Example of Predicates defining structure of $2D$ images

Below we consider examples of predicates that can be used for $2D$ images recognition problems. Consider the images $u_i(x^1, x^2)$ of (say, handwritten digits) on the plane (x^1, x^2).

7.2.1. Predicates Based on Fourier (Wavelet) Image Processing.

Consider the following predicates defined by Fourier image analysis. Suppose we are given the training set of ℓ images $u^j(x^1, x^2)$ and their classifications y_j:

$$(u^1(x^1, x^2), y_1), \ldots, (u^\ell(x^1, x^2), y_\ell).$$

1. Consider $T(T+1)/2$ predicates

$$a^j_{s,r} = \int_\Gamma u_j(x^1, x^2) \cos sx^1 \cos rx^2 dx^1 dx^2, \quad s, r = 0, \ldots, T,$$

(computed numerically) which define the first $T(T+1)$ coefficients of Fourier expansion of image $u^J(x^1, x^2)$

$$u^i(x^1, x^2) = \sum_{s \geq r}^{T} a^i_{s,r} \cos sx^1 \cos rx^2 + O(w(x^1, x^2)).$$

It is known that if $u^j(x^1, x^2)$ is a smooth function (has bounded joint derivatives), then $O(w(x^1, x^2)) \longrightarrow_{T \to \infty} 0$.

The invariants

$$\sum_{i=1}^{\ell} a^i_{s,r} f(u^i) = \sum_{i=1}^{\ell} a^i_{s,r}, \qquad s, r = 0, ..., T$$

corresponding to these predicates allow one to select such set of functions $\{f(u)\}$ for which the equalities holds true.

7.2.2. Predicates defining structure of images.

Let $2D$ images be defined as functions $u(x^1, x^2)$ and let every function is defined by $2D$ Fourier expansion containing T terms with coefficient $A_i,\ i = 1, ..., T$. Then set of $T(T-1)$ predicates $A_i A_j$ defines structure of images for the problem.

7.3. Tangent Distance Based Predicates

Lie operators. Let image be defined by differentiable $2D$ function $f(x^1, x^2)$. Consider small linear transformations of 2D space $(x^1, x^2) \in R^2$:

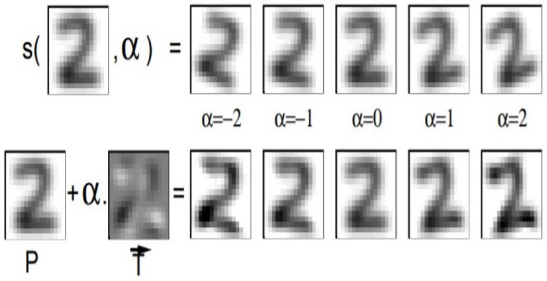
$$t_\alpha \begin{pmatrix} x^1 \\ x^2 \end{pmatrix} \Longrightarrow \begin{pmatrix} x^1 + a_1 x^1 + a_2 x^2 + a_3 \\ x^2 + a_4 x^2 + a_5 x^1 + a_6 \end{pmatrix}$$

For small a_k, the function in transformed space $t_\alpha(x^1, x^2)$ has the following representation in non-transformed space (x^1, x^2):

$$f(t_\alpha(x^1, x^2)) \approx f(x^1, x^2) + \sum_{k=1}^{6} a_k L_k f(x^1, x^2),$$

where $L_k f(x^1, x^2)$ are the so-called *Lie derivatives*. They provide, for small a_k, the following transformations t_α^{-1} of images from space $t_\alpha(x^1, x^2)$ into (x^1, x^2): (1) horizontal translation, (2) vertical translation, (3) rotation, (4) scaling, (5) parallel hyperbolic transformation, (6) diagonal hyperbolic transformation.

Example (from (3)).



Digit 2 in the transformed space and in the original space corrected transformed by the Lie operator of rotation.

1. Horizontal translation

$$t_\alpha^{-1} : \begin{pmatrix} x^1 + a \\ x^2 \end{pmatrix} \Longrightarrow \begin{pmatrix} x^1 \\ x^2 \end{pmatrix}$$

is defined by Lie operator is $L_1 = \frac{\partial}{\partial x^1}$ as

$$f(t_a(x, y)) \approx f(x, y) + a \frac{\partial f(x, y)}{\partial x^1}$$

2. Vertical translation

$$t_a^{-1} : \begin{pmatrix} x^1 \\ x^2 + a \end{pmatrix} \Longrightarrow \begin{pmatrix} x^1 \\ x^2 \end{pmatrix}$$

is defined by Lie operator is $L_2 = \frac{\partial}{\partial x^2}$ as

$$f(t_a(x, y)) \approx f(x, y) + a \frac{\partial f(x, y)}{\partial x^1}$$

3. Rotation transformation

$$t_\alpha^{-1}: \left(\begin{array}{c} x^1 \cos a - x^2 \sin a \\ x^1 \sin a + x^2 \cos a \end{array} \right) \Longrightarrow \left(\begin{array}{c} x^1 \\ x^2 \end{array} \right)$$

is defined by Lie operator $L_3 = x^2 \frac{\partial}{\partial x^1} - x^1 \frac{\partial}{\partial x^2}$ as

$$f(t_a(x^1, x^2)) \approx f(x^1, x^2) + a \left(x^2 \frac{\partial f(x^1, x^2)}{\partial x^1} - x^1 \frac{\partial f(x^1, x^2)}{\partial x^2} \right)$$

4. Scaling transformation

$$t_\alpha^{-1}: \left(\begin{array}{c} x^1 + ax \\ x^2 + ax^2 \end{array} \right) \Longrightarrow \left(\begin{array}{c} x^1 \\ x^2 \end{array} \right)$$

is defined by Lie operator $L_4 = x^1 \frac{\partial}{\partial x^1} + x^2 \frac{\partial}{\partial x^2}$ as

$$f(t_a(x^1, x^2)) \approx f(x^1, x^2) + a \left(x^1 \frac{\partial f(x^1, x^2)}{\partial x^1} + x^2 \frac{\partial f(x^1, x^2)}{\partial x^2} \right)$$

5. Parallel hyperbolic transformation

$$t_\alpha^{-1}: \left(\begin{array}{c} x^1 + a_1 x^1 \\ x^2 - ax^2 \end{array} \right) \Longrightarrow \left(\begin{array}{c} x^1 \\ x^2 \end{array} \right)$$

is defined by Lie operator is $L_5 = x^1 \frac{\partial}{\partial x^1} - x^2 \frac{\partial}{\partial x^2}$ as

$$f(t_a(x^1, x^2)) \approx f(x^1, x^2) + a \left(x^2 \frac{\partial f(x^1, x^2)}{\partial x^1} - x^2 \frac{\partial f(x^1, x^2)}{\partial x^2} \right)$$

6. Diagonal hyperbolic transformation

$$t_\alpha^{-1}: \left(\begin{array}{c} x^1 + ax^2 \\ x^2 + ax^1 \end{array} \right) \Longrightarrow \left(\begin{array}{c} x^1 \\ x^2 \end{array} \right)$$

defined by Lie operator $L_6 = x^2 \frac{\partial}{\partial x^1} + x^1 \frac{\partial}{\partial x^2}$ as

$$f(t_a(x^1, x^2)) \approx f(x^1, x^2) + a \left(x^2 \frac{\partial f(x^1, x^2)}{\partial x^1} + x^2 \frac{\partial f(x^1, x^2)}{\partial x^2} \right)$$

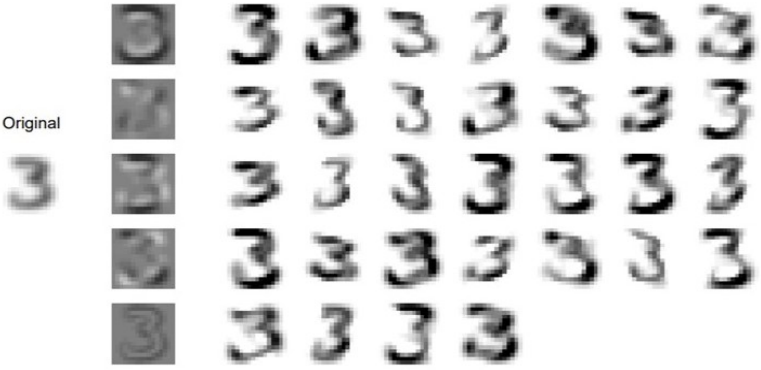
Tangent Distance Based Predicates. Consider two images defined by functions $u(x^1, x^2)$ and $u_s(x^1, x^2)$. We introduce two six-parametric sets of functions

$$\{u(x^1, x^2)\}_a = u(x^1, x^2) + \sum_{k=1}^{6} a_k L_k u(x^1, x^2)$$

$$\{u_s(x^1, x^2)\}_b = u_s(x^1, x^2) + \sum_{k=1}^{6} b_k L_k u_s(x^1, x^2)$$

defined by parameters a_k and b_k, where $k = 1, \ldots, 6$.

Example (from (3))



Digit 3 and its transformations using five Lie operators (scaling, rotation, expansion-compression, diagonal expansion-compression, thickening).

Tangent distance between functions $u(x^1, x^2)$ and $u_s(x^1, x^2)$ is defined by the smallest distance (see [3] for the concept of tangent distance both in continuous and in discrete spaces) between set $\{u(x^1, x^2)\}_a$ and set $\{u_s(x^1, x^2)\}_b$

$$\rho_{tang}(u, u_s) = \min_{a,b} \left| u(x^1, x^2) + \sum_{k=1}^{6} a_k L_k u(x^1, x^2) - u_s(x^1, x^2) - \sum_{k=1}^{6} b_k L_k u_s(x^1, x^2) \right|.$$

Using tangent distance between original image and its transformation, one can construct specific invariants.

7.4. Predicates Describing Levels of Symmetries and Asymmetries of Images.

Let the image $f(x)$ be defined in the discrete 2D space by $(n_1 \times n_2)$ values x of pixels. Consider the following concepts of symmetries and asymmetries in this space.

1. Predicate for vertical symmetry of image x. Consider, along with image x, its vertical mirror transformed image $\widehat{x}$, where last line of pixels in x becoming first line in $\widehat{x}$ and first line in x becoming last line $\widehat{x}$:

$$x = \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nn} \end{bmatrix}, \quad x_v = \begin{bmatrix} x_{n1} & \cdots & x_{nn} \\ \vdots & \ddots & \vdots \\ x_{11} & \cdots & x_{1n} \end{bmatrix}.$$

We consider as predicate of vertical symmetry of image x the tangent distance $\phi_v(x) = d_{tang}(x, \widehat{x}_v)$ between two vectors x and x_v.

2. Predicate for horizontal symmetry of image x. Consider, along with image x, the transformed image:

$$x = \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nn} \end{bmatrix}, \quad x_h = \begin{bmatrix} x_{1n} & \cdots & x_{11} \\ \vdots & \ddots & \vdots \\ x_{nn} & \cdots & x_{n1} \end{bmatrix}.$$

We consider the tangent distance$\phi_h(x) = d_{tang}(x, x_h)$ between vectors x and x_h as the predicate of vertical symmetry of image x.

3. Predicate for horizontal antisymmetry of image x (Example character **S**)). Consider along with image x the transformed image

$$x = \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nn} \end{bmatrix}, \quad \widehat{x}_h = \begin{bmatrix} x_{nn} & \cdots & x_{n1} \\ \vdots & \ddots & \vdots \\ x_{1n} & \cdots & x_{11} \end{bmatrix}.$$

We consider as predicate of vertical antisymmetry of image x the tangent distance $\phi_h(x) = d_{tang}(x, \widehat{x}_h)$ between vectors x and $\widehat{x}_h$.

Similarly, one can introduce concepts of *vertical antisymmetry, diagonal symmetry / antisymmetry (left and right) and many others.*

Using these predicates, one can construct invariants that keep corresponding degree of symmetries/antisymmetries of images.

8. Conclusive Remarks

1. Mathematical mechanisms of learning. The solution of learning problem is based on two mechanisms: (1) the mechanism of strong convergence (convergence of approximations to the desired function in L_2 metric) and (2) the mechanism of weak convergence (convergence of approximations to the desired function in the space of functionals). The strong mode of convergence implies the weak mode convergence. For most learning problems (when a given set of functions is compact), the weak mode implies strong mode as well. Thus two different mechanisms of estimation of the desired function can lead to the same solution.

This paper considers algorithms that use both these mechanisms simultaneously. Since there are only two mechanisms of convergence for functions of Hilbert space, and since we use both of them, we call the corresponding theory *the complete theory of learning.*

2. Concepts of predicates and features in learning models. In order to implement the weak mode of convergence, we select a finite subset of predicate functions from Hilbert space. Using these functions and training data, we construct the set of constraints which allow us to select a set of admissible functions from which we finally choose the desired approximation.

The concept of predicates is mathematically well defined, which is in contrast to the concept of features (which is defined only on an intuitive level). As mentioned above, these concepts play different roles in the models of learning.

With the increasing number of predicates, the capacity of the admissible set of functions *decreases.* (According to the definition of weak convergence, when predicates are all the functions of Hilbert space and when we can estimate the right-hand side of invariant equalities accurately, the admissible set has only one function – the desired one).

With the increasing number of features, the capacity of the set of admissible function *increases.* This requires, generally speaking, an increase of the size of training data.

The only remaining question in the complete statistical learning theory is how to choose a (small) set of predicates. The choice of predicate functions reflects the intellectual part of the learning problem; it reflects our understanding of the nature of Real World problems where Learning Machine acts.

3. Predicate functions for understanding life. In 1928, Vladimir Propp wrote the book "Morphology of the Folk Tale", where he formulated 31 functions into which he decomposed Russian fairy tales. Later, these functions have been successfully applied to other types of narratives in literature, theater, film, television series, games, etc. In other words, Propp's functions reflected understanding of humans relationships not only in World of Russian fairy tales but also in the wider World of human life.

Similarly, the old Chinese book 'The Art of War" (attributed to Sun Tzu, 5th century BC) describes 33 general strategic rules (predicates) which reflect intelligent control of actions in Wars. These rules, however, are wider than just principles for military actions. They are also studied in business schools of management to teach directions of activities in real competitive life.

Our hypothesis is that there exists a relatively small number of predicates (including ones based on idea of structure (Section 7.2) and idea of symmetry (Sections 7.4)) that reflect our understanding of World of 2D black and white images).

These predicates allow one to introduce invariants (constructed based on training data) to use them in LUSI algorithms. This can lead to high performance based on a small size of training data (the additional necessary information is extracted from the invariants).

The solution of the following challenge problem can be one of the first steps in the selection of "universal" predicates for image understanding.

4. The Challenge. The current solutions of 10 class digit recognition problem (MNIST dataset) using the training data of size 60,000 ($\approx 6,000$ examples per class) achieve the error rate of about 0.5%.

The challenge is to get approximately the same level of error rate (using LUSI methods) with the training size that is 100 times smaller (600 examples, that is, only 60 per class). In order to do this, one has to formulate the appropriate predicates. The challenge is to formulate a small number of such predicates that would allow to solve this problem. The hope is that these predicates (as in the Sun Tzu and Propp's case) will be applicable for other 2D graphical pattern recognition problems.

5. Plato's Type of World Model. The philosophy of the methods considered above can be described in the style of Plato's understanding of World as consisting of two parts: World of Ideas and World of Things.

World of Ideas reflects our understanding of the Real World, our intelligence. World of Things is World of actions which is a result of projection of our understanding of the Real World into real situation. In our model, the World of Ideas (which reflects the understanding of Real World) is a set of predicate functions. They allow one to construct invariants for selection of the admissible set of functions for the problem defined by training data. By using universal abstract predicates and training data, one constructs statistical invariants in order to select the admissible set of models for the given problem of interest.

The new element in such Plato's type of model is the interaction of Ideal World with Real World: the mechanism of transformation of universal ideas (predicates) into specific constraints for action (statistical invariants).

6. Imitation of intelligence and the essence of intelligence. In the study of Artificial Intelligence, one can differentiate between two problems.

1. Engineering problem: Problem of *imitation* of intelligence (in order to solve this problem, one has to construct a machine which passes Turing *imitation* test) and

2. Scientific problem: Problem of understanding the *essence of intelligence* (what defines the abstract understanding of Real World?).

In our model, intelligence is defined by a set of predicate functions that reflect our understanding of elements of Real World. The challenge is to find them for different type of problems existing in Real World, in particular, for classification problems in World of 2D graphical images.

References

[1] Vapnik, V.: The Nature of Statistical Learning Theory. Springer-Verlag, New York (1995).

[2] Vapnik, V.: Statistical Learning Theory. John Wiley & Sons, New York (1998).

[3] Patrice Y. Simard, Yann A. LeCun, John S. Denker, Bernard Victorri: Transformation Invariance in Pattern Recognition - Tangent Distance and Tangent Propagation. Neural Networks: Tricks of the Trade, pp. 239-274 (2002).

[4] V.Vapnik, R.Izmailov, Rethinking statistical learning theory: learning using statistical invariants, Machine Learning, 1-43, 2018.

[5] D. Griffel, Applied Functional Analysis. John Wiley & Sons, New York (1995).