Trust Estimation in Forecasting-Based Knowledge Fusion*

Miroslav Kárný[1][0000−0002−7440−6041] and Daniel Karlík[1][0000−0001−8571−7534]

The Czech Academy of Sciences, Institute of Information Theory and Automation
18200 Prague 8, Czech Republic, school@utia.cas.cz, daniel@karlik.cz
https://www.utia.cas.cz/AS

Abstract. Inference and *d*ecision *m*aking (DM) are ultimate goals of the artificial-intelligence use. Complexity of DM tasks is the main barrier of their efficient solutions. Complex tasks are solved by dividing them among cooperating agents. This requires a knowledge fusion at a solution stage. It always has to cope with uncertainty. The used Bayesianism quantifies the uncertain knowledge by a *p*robability *d*ensity (pd) of modelled variables. The knowledge accumulation evolves the posterior pd of a parameter in the parametric model of observations. Bayes' rule updates the posterior pd. It provides a lossless compression of the knowledge in the observed data. An extended Bayes' rule enables the use of knowledge coded in a forecaster of the modelled observations supplied by an agent's neighbour. This rule exploits a weight expressing the trust into the forecaster. The paper offers yet-missing, algorithmic, data-based choice of this weight. It applies Bayesian estimation while assuming an invariant trust weight. Simulated examples illustrate behaviour of the resulting algorithm. They inspect its sensitivity to violation of the assumed credibility invariance. This prepares solutions coping with volatile knowledge sources.

Keywords: Trust · Knowledge sharing · Forecasting · Fusion · Decision making · Bayesianism.

1 INTRODUCTION

Complex decision-making (DM) tasks are solved by dividing them among cooperating agents[1], [7]. This requires a knowledge fusion at a solution stage, [33]. An agent locally models its environment. It selects its actions according to its local — in information space and time — aims. The efficiency of such an adaptive agent is enhanced (if not enabled at all) by sharing a knowledge with its neighbours in the information space. The neighbours are imperfect and may even act as adversaries. This makes the use of the shared knowledge strongly dependent on the *trust* assigned to neighbours. The trust quantification is actively studied in various contexts, [8,11,34], but it is far from being matured. The paper contributes to an improvement of this state. It deals with a specific, but well-applicable, knowledge-sharing scenario. The sharing supports an agent estimating a parametric model by using observations and Bayes' rule, [24]. Its

* Supported by MŠMT LTC18075 and EU-COST Action CA16228

[1] They are humans, technical tools and their mixed groups. The agent is referred by "it".

neighbour irregularly offers a forecaster of the same observation. It adds the number indicating how many data items the forecaster reflects. The agent processes them by the extended Bayes' rule. This rule has its origin in [15]. Its advanced, formally derived, versions are in [14,26]. They use a trust weight assigned to the neighbour.

The vital, but yet-unsolved, choice of the trust weight is addressed here.

Layout: Sec. 2 makes the paper self-reliant by recalling the used theory. Sec. 3 solves the addressed problem. Simulations in Sec. 4 illustrate the solution and inspect its sensitivity to the adopted invariance assumption. Sec. 5 touches the case of volatile credibility of the neighbour. Concluding remarks are in Sec. 6.

Notation: The text applies the next agreements:
$\{x\}$ is a set of x's, its nature is only revealed if need be; $|x|$ is cardinality of $\{x\}$;
$:=$ defines by assigning; $\propto$ is equality up to the normalisation; t marks discrete time;
✓ random variables, their values and realisations are formally undistinguished;
✓ models are *p*robability *d*ensities (pds[2]) marked by $\mathsf{sansmath}$ fonts as all mappings;
✓ functions with different arguments are different; the text prefers mnemonic labels;
$\mathsf{g}(x_t, y_{t-1}) := \mathsf{g}_t(x_t, y_{t-1})$: the time index of a function g drops if it is at its argument;
$\mathsf{p}_{t-1}(p)$ is the posterior pd of an unknown parameter $p \in \{p\}$, entering the parametric model; it is conditioned on the knowledge processed up to time $t-1$;
$\mathsf{p}(p|w, \mathsf{f}_t)$ enriches the condition of $\mathsf{p}_{t-1}(p)$ by the forecaster f_t with the trust weight w.

2 Preliminaries

An agent uses a parametric model $\mathsf{m}_t(o|r, p)$. This conditional pd relates the observation $o \in \{o\}$ to the regressors $r \in \{r\}$ and to an unknown parameter $p \in \{p\}$. The relation depends on time $t \in \{t\} := \{1, 2, \dots\}$. The posterior (conditional) pd $\mathsf{p}_{t-1}(p)$ quantifies the agent's knowledge about the unknown parameter $p \in \{p\}$ gained up to time $t-1$. Having data $d_t := (o_t, r_t)$, the pd $\mathsf{p}_{t-1}(p)$ updates by Bayes' rule, [24], to

$$\mathsf{p}_t(p) = \frac{\mathsf{m}(o_t|r_t, p)\mathsf{p}_{t-1}(p)}{\mathsf{m}(o_t|r_t, \mathsf{p}_{t-1})}, \quad \mathsf{m}_t(o|r, \mathsf{p}) := \int_{\{p\}} \mathsf{m}_t(o|r, p)\mathsf{p}(p)\,\mathrm{d}p, \quad t \in \{t\}. \tag{1}$$

The normalising pd $\mathsf{m}_t(o|r, \mathsf{p})$ models the observation o for the given regressors r and the knowledge about unknown parameter $p \in \{p\}$ stored in the pd $\mathsf{p}(p)$. It is agent's forecasting model. A subjective prior pd p_0, [29], starts the recursion (1).

In the inspected knowledge sharing, a neighbour provides to the agent its forecaster $\mathsf{f}_t(o)$ of the observations $o_t \in \{o\}$. This *non-normalised* pd should reflect the situation with the same regressors r_t as those used by the agent for forecasting of o_t. The number $\nu_t := \int_{\{o\}} \mathsf{f}_t(o)\,\mathrm{d}o \in (0, \infty)$ enhances the knowledge stored in the pd $\mathsf{f}_t(o)/\nu_t$. It declares the amount of data items used for creating the forecaster.

The neighbour forecasts using other knowledge resources than the agent. It means other models, theories, data sets, processing ways, expert's opinions, simulations, etc.

[2] Pd means Radon-Nikodým derivative, [28], i.e. both a probability density and mass function.

The theory we rely on, see Prop. 3 in [14], exploits the forecaster f_t by the extended Bayes' rule. It corrects the posterior pd $p_{t-1}(p)$ to the pd denoted $p(p|w_t, f_t)$

$$p(p|w_t, f_t) \propto p_{t-1}(p) \exp\left[w_t \int_{\{o\}} f_t(o) \ln[m(o|r_t, p)]\, \mathrm{d}o\right], \quad \text{where} \tag{2}$$

$w_t \in [0,1]$ is the agent's trust weight assigned to the neighbour's forecaster f_t. The pd $p(p|w_t, f_t)$ is conditioned on the knowledge entering p_{t-1} enriched by the forecaster f_t weighted by w_t. The relation (2) indeed extends Bayes' rule as a fully trustable, $w_t = 1$, single, $\nu_t = 1$, crisp observation o_t is modelled by Kronecker's (Dirac's) pd

$$\delta(o, o_t) := \begin{cases} 1 & \text{if } o = o_t \\ 0 & \text{otherwise} \end{cases} \quad \text{and reduces (2) to (1) as } p(p|w_t := 1, \delta_t) \overbrace{\propto}^{(2)}$$

$$p_{t-1}(p) \exp\left[1 \times \int_{\{o\}} \delta(o, o_t) \ln[m(o|r_t, p)]\, \mathrm{d}o\right] = p_{t-1}(p)[m(o_t|r_t, p)]^1 \overbrace{\propto}^{(1)} p_t(p).$$

It is practically important that for parametric models from *e*xponential *f*amily (EF, [4]), the functional rule (2) reduces to an algebraic updating of values of a sufficient statistic. EF consists of the parametric models of the form

$$m_t(o|r, p) := \exp\langle a_t(d), b(p)\rangle, \quad d := (o, r). \tag{3}$$

They are instantiated by multivariate functions a_t, b with their values entering the scalar product $\langle\cdot,\cdot\rangle$. In thought cases, the scalar product has the simple form

$$\langle a_t(d), b(p)\rangle := \sum_{i \in \{i\}} a_{ti}(d) b_i(p), \quad |i| < \infty, \; t \in \{t\}, \tag{4}$$

where a_{ti}, b_i are known real-valued functions.

The used posterior pd p_t, conjugated to the model (3), [5], is given by the value of the $|i|$-dimensional statistic $\sigma_t = (\sigma_{ti})_{i\in\{i\}}$ with real-valued σ_{ti}. The pd reads

$$p_t(p) := c(p|\sigma_t) := \frac{\exp\langle \sigma_t, b(p)\rangle}{n(\sigma_t)}, \quad n(\sigma) := \int_{\{p\}} \exp\langle \sigma, b(p)\rangle\, \mathrm{d}p < \infty. \tag{5}$$

Updating by the extended Bayes' rule (2) preserves the form (5). It holds

$$\begin{aligned} p_{t-1}(p) &= c(p|\sigma_{t-1}) \quad \overbrace{\Rightarrow}^{(2)} \quad p(p|w_t, f_t) = c(p|\sigma(w_t, f_t)) \\ \sigma_i(w_t, f_t) &= \sigma_{(t-1)i} + w_t a_i(f_t, r)\delta(r, r_t) \\ a_i(f_t, r) &:= \int_{\{o\}} f_t(o) a_{ti}(o, r)\, \mathrm{d}o, \;\; t \in \{t\}, \;\; i \in \{i\}, \;\; r \in \{r\}. \end{aligned} \tag{6}$$

This important case exemplifies the influence of the trust weight $w_t \in [0,1]$.

Markov's chain as a member of EF: Markov's chain models the evolution of data with a finite number of possible values. Its parametrisation takes all transition probabilities as the unknown parameter. The next expression uses Kronecker's δ and $d = (o, r)$

$$p_{o_t|r_t} := \mathsf{m}(o_t|r_t, p) = \prod_{d\in\{d\}} p_{o|r}^{\delta(d,d_t)} = \exp\left[\overbrace{\sum_{d\in\{d\}} \underbrace{\delta(d, d_t)}_{\mathsf{a}_{o|r}(d_t)} \underbrace{\ln(p_{o|r})}_{\mathsf{b}_{o|r}(p)}}^{\langle \mathsf{a}(d_t),\mathsf{b}(p)\rangle}\right]. \tag{7}$$

This is an EF member (3), (4) with $i := o|r$. Its conjugated pd (5) is Dirichlet's pd $\mathsf{c}(p|\sigma) \propto \prod_{r\in\{r\}} \prod_{o\in\{o\}} p_{o|r}^{\sigma_{o|r}-1}$. The positive values of the statistic $\sigma := (\sigma_{o|r})_{o\in\{o\},r\in\{r\}}$ describe this pd. They enter the normalisation $\mathsf{n}(\sigma)$ (5), [13],

$$\mathsf{n}(\sigma) = \prod_{r\in\{r\}} \frac{\prod_{o\in\{o\}} \Gamma(\sigma_{o|r})}{\Gamma\left(\sum_{o\in\{o\}} \sigma_{o|r}\right)}, \quad \Gamma(v) := \int_0^\infty z^{v-1} \exp(-z)\, \mathrm{d}z, \quad v > 0. \tag{8}$$

The agent's forecasting model $\mathsf{m}_t(o|r, \mathsf{p})$ (1), found by (8) and $\Gamma(v + 1) = v\Gamma(v)$, [1], is

$$\mathsf{m}(o|r, \mathsf{p}) = \mathsf{m}(o|r, \sigma) = \frac{\sigma_{o|r}}{\sum_{\tilde{o}\in\{o\}} \sigma_{\tilde{o}|r}}.$$

For $w_t \in [0, 1]$, $i = o|r$, $r, r_t \in \{r\}$, $o \in \{o\}$, the rule (6) gives the sufficient statistic

$$\sigma_{o|r}(w_t, \mathsf{f}_t) = \sigma_{(t-1)o|r} + w_t \mathsf{f}_t(o)\delta(r, r_t), \;\; o \in \{o\},\; r \in \{r\}.$$

3 Estimation of the Trust Weight

The unknown trust weight w_t in (2) is a hidden variable. Non-linear stochastic filtering, [10], estimates it optimally. It needs, however, the rarely-available time-evolution model and quite complex evaluations. This makes us to use local modelling, typical for adaptive systems. The inspected case of the *invariant trust*, $w = w_t$, $\forall t \in \{t\}$, prepares the general solution. Sec. 5 comments the volatile case.

The invariant w extends the parameter $p \in \{p\}$ to unknowns $(p, w) \in (\{p\}, [0, 1])$ entering the parametric model and the knowledge processing. A joint pd

$$\mathsf{p}_{t-1}(p, w) = \mathsf{p}_{t-1}(p|w)\beta_{t-1}(w) \tag{9}$$

describes the knowledge about (p, w) after time $t - 1$ and before $t \in \{t\}$. The factorisation in (9) is the chain rule for pds, [24]. The conditional pd $\mathsf{p}_{t-1}(p|w)$ accumulates the knowledge about the unknown $p \in \{p\}$ when assigning the fixed trust weight w to the knowledge provided by the neighbour through forecasters offered before time t. The pd $\beta_{t-1}(w)$ expresses the agent's belief that w is the proper trust weight for the neighbour. The neighbour's forecaster $\mathsf{f}_t(o)$ enters the conditional version of (2)

$$\mathsf{p}(p|w, \mathsf{f}_t) \;\propto\; \mathsf{p}_{t-1}(p|w) \exp\left[w\zeta_t \int_{\{o\}} \mathsf{f}_t(o) \ln[\mathsf{m}(o|r_t, p)]\, \mathrm{d}o\right]$$
$$\zeta_t := \begin{cases} 1 & \text{if the forecaster } \mathsf{f}_t \text{ is available} \\ 0 & \text{otherwise} \end{cases}. \tag{10}$$

The introduced indicator ζ_t allows us to respect irregularity of processing of neighbour's forecasters without making the notation too complex. The agent's forecasting model, normalising (1) for the given $\mathsf{p}_{t-1}(p|w)$, is

$$\mathsf{m}(o|r, \mathsf{p}_{t-1}, w) := \int_{\{p\}} \mathsf{m}_t(o|r,p)\mathsf{p}_{t-1}(p|w)\,\mathrm{d}p. \tag{11}$$

In (10), (11), the weight w concerns the neighbour and thus it enters the posterior pds $\mathsf{p}_{t-1}(p|w)$ but not the agent's parametric model $\mathsf{m}(o|r,p)$.

The data-based updating of the belief $\beta_{t-1}(w)$ (9) into trust weights $w \in [0,1]$ may realise after observing how much the neighbour's knowledge has contributed to the forecasting quality. The standard Bayes' rule gives, cf. (11),

$$\beta_t(w) \propto \mathsf{m}(o_t|r_t, \mathsf{p}_{t-1}, w)\beta_{t-1}(w). \tag{12}$$

The implementation of the recursion (10), (11), (12) is generally hard. It is simple for the discretised trust weight, [19]. The next proposition summarises such updating.

Proposition 1 (Parameter and Trust-Weight Estimation). *Let imminent trust weights be $w \in (w_k)_{k\in\{k\}}$, $\{k\} := \{1,\ldots,|k|\}$, $|k| < \infty$. They condition pds $\big(\mathsf{p}_{t-1}(p|w_k)\big)_{k\in\{k\}}$ quantifying the knowledge about the unknown parameter $p \in \{p\}$ of the pd $\mathsf{m}(o_t|r_t,p)$.*

The knowledge includes past data collected up to and including time $t-1$. It is enriched by irregularly available neighbour's forecasters with weights $(w_k)_{k\in\{k\}}$.

The values $(w_k)_{k\in\{k\}}$ express the neighbour's, supposedly invariant, credibility. They enter the updating of $\mathsf{p}_{t-1}(p|w_k)$ by the neighbour's forecaster f_t

$$\mathsf{p}(p|w_k,\mathsf{f}_t) \propto \mathsf{p}_{t-1}(p|w_k)\exp\left[w_k\zeta_t\int_{\{o\}}\mathsf{f}_t(o)\ln[\mathsf{m}(o|r_t,p)]\,\mathrm{d}o\right],\ \ p\in\{p\},\ k\in\{k\},$$

with $\zeta_t = 1$ if f_t is available and zero otherwise, cf. (10).

Let beliefs into trust weights w_k be $\beta_{t-1}(w_k)$, $k \in \{k\}$, see (9). Then, the updating of the pds β_{t-1}, p_{t-1} by data $d_t = (o_t, r_t)$ via the standard Bayes' rule reads, cf. (11),

$$\begin{aligned}
\beta_t(w_k) &= \frac{\mathsf{m}(o_t|r_t,\mathsf{p}_{t-1},w_k)}{\mathsf{m}(o_t|r_t,\mathsf{p}_{t-1},\beta_{t-1})}\beta_{t-1}(w_k)\\
\mathsf{m}(o|r,\mathsf{p}_{t-1},w_k) &:= \int_{\{p\}}\mathsf{m}_t(o|r,p)\mathsf{p}_{t-1}(p|w_k)\,\mathrm{d}p \\
\mathsf{m}(o|r,\mathsf{p}_{t-1},\beta_{t-1}) &:= \sum_{k\in\{k\}}\mathsf{m}(o|r,\mathsf{p}_{t-1},w_k)\beta_{t-1}(w_k)\\
\mathsf{p}_t(p|w_k) &\propto \mathsf{m}(o_t|r_t,p)\mathsf{p}(p|w_k,\mathsf{f}_t).
\end{aligned} \tag{13}$$

Prop. 1 applied to EF (3) maps both Bayes' functional recursions to algebraic handling of the finite-dimensional statistic.

Proposition 2 (Estimation of Parameter and Trust Weight in Exponential Family). *Let trust weights $(w_k)_{k\in\{k\}}$ condition conjugated pds $\mathsf{p}_{t-1}(p|w_k) = \mathsf{c}(p|\sigma_{t-1}(w_k))$,* *(5). Let $(\beta_{t-1}(w_k))_{k\in\{k\}}$ be beliefs assigned to the trust weights. Their updating by the forecaster $f_t(o)$, preserves the conjugated form (5) and reads*

$$
\begin{aligned}
p(p|w_k, f_t) &= c(p|\sigma(w_k, f_t)) = \frac{\exp\langle\sigma(w_k, f_t), \beta(p)\rangle}{n(\sigma(w_k, f_t))}, \quad n(\sigma) = \int_{\{p\}} \exp\langle\sigma, b(p)\rangle \, \mathrm{d}p \\
\sigma_i(w_k, f_t) &= \sigma_{(t-1)i}(w_k) + w_k \zeta_t a_i(f_t, r)\delta(r, r_t) \qquad (14)\\
a_i(f_t, r) &:= \int_{\{o\}} f_t(o) a_i(o, r) do, \qquad r \in \{r\}, \ i \in \{i\}, \ k \in \{k\},
\end{aligned}
$$

with ζ_t (10) respecting irregular availability of forecasters. The updating by the standard Bayes' rule, after having data $d_t = (o_t, r_t)$, see (6) and (14), reads

$$\sigma_{ti}(w_k) = \sigma_i(w_k, f_t) + a_i(d_t), \quad \beta_t(w_k) \propto \frac{n(\sigma(w_k, f_t))}{n(\sigma_{t-1}(w_k))} \beta_{t-1}(w_k), \ k \in \{k\}. \qquad (15)$$

Thus, we have to store values of statistics $(\sigma(w_k), \beta(w_k))_{k\in\{k\}}$. The increments $a(f_t, r_t)$ (14) and $a(d_t) = a(\delta_t, r_t)$ (15) are evaluated once.

Trust estimation for Markov's chain: Specialisation of Prop. 2 and Sec. 2 imply that Dirichlet's pd is conjugated to the Markov's chain (7). Its degrees of freedom and beliefs into respective trust weights evolve, for $i = o|r$, as follows

$$
\begin{aligned}
\sigma_{o|r}(w_k, f_t) &= \sigma_{(t-1)o|r}(w_k) + w_k \zeta_t f_t(o) \delta(r, r_t) \\
\sigma_{(t)o|r}(w_k) &= \sigma_{o|r}(w_k, f_t) + \delta((o, r), (o_t, r_t)) \qquad (16)\\
\beta_t(w_k) &\propto \frac{\sigma_{(t)o_t|r_t}(w_k)}{\sum_{o\in\{o\}} \sigma_{(t)o|r_t}(w_k)} \beta_{t-1}(w_k), \ (o, r) = d \in \{d\}, \ k \in \{k\},
\end{aligned}
$$

where ζ_t (10) respects irregular offers of f_t.

Formulae (16) have strong intuitive appeal:

- ▶ the forecaster distributes its mass over possible observations $o \in \{o\}$ according to the probabilities $f_t(o)$ it assigns them, cf. quasi-Bayes techniques, [31];
- ▶ the agent attenuates f_t by the trust weight $w_k \in [0, 1]$ (discarding it for $w_k = 0$);
- ▶ the beliefs to weights reflect the neighbour's contribution to the forecasting quality.

The exploitation of the gained posterior pds depends on the DM task. For instance:

- ▶ a point estimate of the trust weight can be constructed, say, $\hat{w}_t := \sum_{k\in\{k\}} w_k \beta_t(w_k)$;
- ▶ Bayesian averaging may estimate parameter $p \in \{p\}$, say, via the marginal pd $p_t(p)$ $:= \sum_{k\in\{k\}} p_t(p|w_k)\beta_t(w_k)$ or similarly to forecast the observation $o_t \in \{o\}$ without specifying a point estimate of the weight;
- ▶ the trust estimate may serve to other, neighbour-related, inference or DM tasks.

4 Illustrative Experiments

Experiments illustrate the presented theory and show the sensitivity of the found estimator to the key assumption that the credibility of the neighbour's forecasters is invariant.

4.1 Simulation and Evaluation Conditions

The modelled environment was simulated by a discretised version of 2[nd] order auto-regressive-regressive Gaussian model

$$y_t = 1.9600 y_{t-1} - 0.9604 y_{t-2} + 0.0004 a_t + 0.0004 \varepsilon_t,$$

where ε_t was white zero-mean noise with unit variance; ε_t was independent of the past values $y_{\tau-1}, a_\tau, \tau \leq t$. The dynamics corresponds with the double real pole 0.98 and the unit static gains of actions and of the noise, [3]. Five-valued, integer, uniformly distributed, independent actions a_t were used, $|a| = 5$. A realisation of 10^5 samples, initiated by $y_0 = y_1 = 1$, was linearly mapped on positive values and discretised to ten-valued integer observations o_t, $|o| = 10$. The sequence $(o_t, a_t)_{t=2}^{10^5}$ was used for the choice of the *simulated* transition probability $p(o_t|o_{t-1}, o_{t-2}, a_t)$. The point estimate of this pd from the said realisation was used. Work [25] inspired this choice. The 2[nd] order Markov model was gained. The agent estimated 1[st] order model $p(o_t|o_{t-1}, a_t, p) = p_{o_t|o_{t-1}, a_t}$, $r_t = (o_{t-1}, a_t)$, (7), i.e. the realistic mismodelling error was faced.

The neighbour's forecaster used the simulated transition probability with the inserted condition o_{t-1}, o_{t-2}, a_t. In the sensitivity-oriented experiments, this ideal forecaster was partially replaced by a randomly generated one, see below.

The trust-weight values $(w_k)_{k \in \{k\}} := \{0, 0.5, 1\}$, $|k| = 3$, Prop. 1, were inspected.

Prior statistics σ_0 (15) had randomly and independently assigned values 1 or 2.

Evaluations used 1000 *M*onte *C*arlo (MC) runs each lasting $|t| = 500$ steps, giving:

▶ Histograms of *beliefs* $\beta_t(w_k)$ (9) and of the *estimates*

$$\hat{w}_t := \sum_{k \in \{k\}} w_k \beta_t(w_k) \tag{17}$$

of weights at the simulation end. Figures with time courses show their medians.

▶ Histograms of *forecast errors* per step compared to the best available forecast $\hat{o}_t^{\mathrm{i}}$ provided by the simulated transition probability

$$\Delta := \frac{1}{|t|} \sum_{t \in \{t\}} \Big| |o_t - \hat{o}_t| - |o_t - \hat{o}_t^{\mathrm{i}}| \Big|. \tag{18}$$

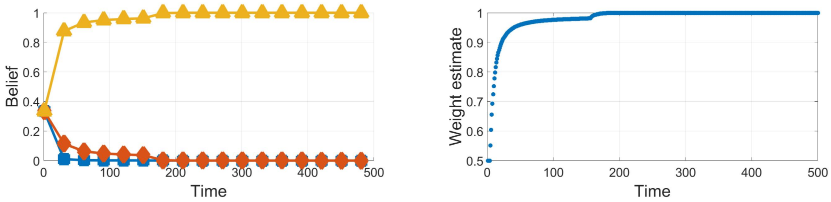
There, o_t is the observation at the time t and judged $\hat{o}_t$ are the forecasts given by $m(o|r, p_{t-1}, w_k)$, $\forall\, k \in \{k\}$, (11) and by $m(o|r, p_{t-1}, \beta_{t-1})$ (13).

▶ Tables of basic *statistics* of the forecast errors (18) at the end of simulations. Their median, mean, standard deviation (STD) and root mean square error (RMS) are shown. RMS is taken as the primary indicator of quality when comparing the results.

4.2 Invariant Ideal and Bad Neighbour's Forecasters

This part shows the behaviour of the proposed processing under met assumptions.

Ideal Neighbour's Forecaster: The neighbour's forecaster was the best possible one, i.e. the simulated $f_t(o) := p(o_t = o|o_{t-1}, o_{t-2}, a_t)$, $o \in \{o\}$, at realised o_{t-1}, o_{t-2}, a_t.

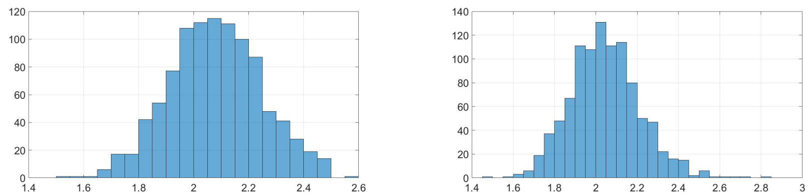


(a) Medians of beliefs in particular weights. (b) Median of the weight estimates.

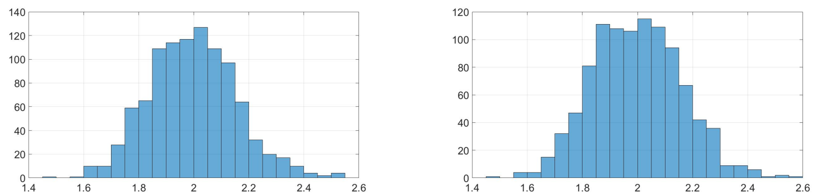
Fig. 1: shows medians of: (a) the beliefs $\beta_t(w_1)$ ■, $\beta_t(w_2)$ ♦, $\beta_t(w_3)$ ▲. (b) the weight estimate (17). It reflects 10^3 MC runs with the **ideal** neighbour's forecaster.

Results: Fig. 1 shows a fast convergence of the beliefs. The median of $\beta_t(w_3 = 1)$ raised rapidly to 1 and stayed there. Thus, the weight estimate (17) converged to 1, too.

Fig. (2) shows histograms of forecast errors Δ (18). They are presented for completeness only. The differences are better seen on statistic values shown in Tab. 1.



(a) Forecast errors of agent using $w_1 = 0.0$. (b) Forecast errors of agent using $w_2 = 0.5$.



(c) Forecast errors of agent using $w_3 = 1.0$. (d) Forecast errors of the proposed way.

Fig. 2: has counts of errors Δ (18) on the vertical axis and values of Δ on the horizontal axis. It reflects 10^3 MC runs with the **ideal** neighbour's forecaster.

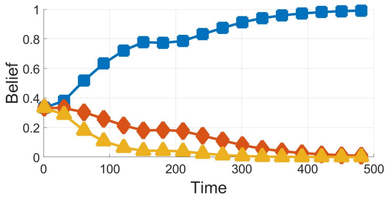
Discussion: The results confirm the desirable behaviour of the trust estimator. The high convergence rate is plausible. As predictable, the best quality is obtained for the fixed full weight assigned to the ideal forecaster. The proposed way is only slightly worse. The poorer performance is the cost for the lack of the knowledge of the proper weight.

Bad Neighbour's Forecaster: In this case, the neighbour's forecaster was chosen as useless as it was selected randomly without any relation to the simulated environment.

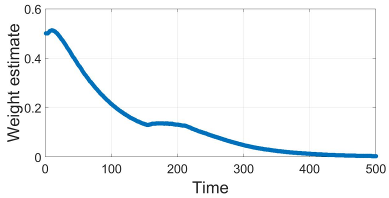
Table 1: Forecast errors Δ (18) with the **ideal** neighbour's forecaster.

Forecaster	Median	Mean	STD	RMS
Agent using $w_1 = 0.0$	2.076	2.079	0.167	2.086
Agent using $w_2 = 0.5$	2.038	2.035	0.165	2.042
Agent using $w_3 = 1.0$	1.992	1.996	0.159	2.002
Proposed way	1.997	1.998	0.161	2.005

Results: Fig. 3 shows that the proposed way behaves as desirable. The medians of beliefs into non-zero weights go quickly to 0. The point estimate $\hat{w}$ (17) goes also to 0.



(a) Medians of beliefs in particular weights.



(b) Median of the weight estimates.

Fig. 3: shows medians of: (a) $\beta_t(w_1)$ ■, $\beta_t(w_2)$ ♦, $\beta_t(w_3)$ ▲. (b) the weight estimate (17). It reflects 10^3 MC runs with the **bad** neighbour's forecaster.

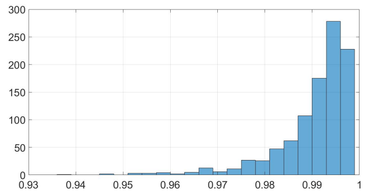
Histograms of forecast errors are poorly informative and they are left out. Their statistics are in Tab. 2. The best result is gained for the fixed zero weight ignoring the bad forecaster. The proposed way is close to it. It needed some data to recognise that the neighbour's forecaster is useless.

Table 2: Forecast errors Δ (18) with the **bad** neighbour's forecaster.

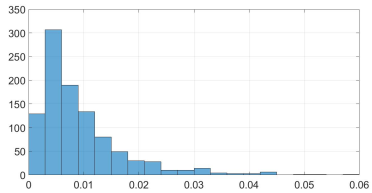
Forecaster	Median	Mean	STD	RMS
Agent using $w_1 = 0.0$	2.069	2.076	0.164	2.083
Agent using $w_2 = 0.5$	2.096	2.094	0.167	2.101
Agent using $w_3 = 1.0$	2.118	2.116	0.165	2.122
Proposed way	2.080	2.081	0.165	2.088

Fig. 4 complements the picture by presenting histograms of beliefs and the weight estimates (17) at the ends of simulation runs. They show quite small variations.

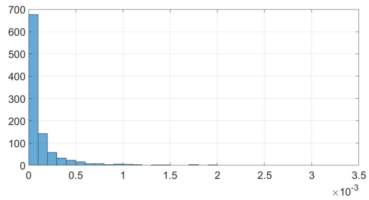
Discussion: The results confirm the expected desirable behaviour. Similarly as with the ideally forecasting neighbour, the poor forecasting was quickly recognised. As pre-



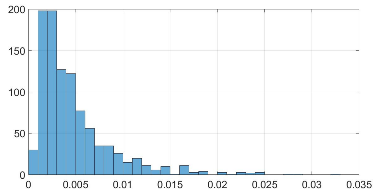
(a) Histogram of belief $\beta(w_1)$.



(b) Histogram of belief $\beta(w_2)$.



(c) Histogram of belief $\beta(w_3)$.



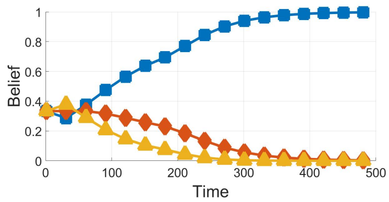
(d) Histogram of the weight estimate.

Fig. 4: shows counts of values on the vertical axis, the final values of $\beta_{|t|}$ and $\hat{w}_{|t|}$ on the horizontal axes. It reflects 10^3 MC runs with the **bad** neighbour's forecaster.

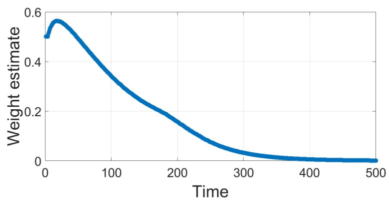
dictable, the best quality is obtained for the fixed zero weight assigned to the bad forecaster. The proposed way is only slightly worse. It pays for the lack of the knowledge.

4.3 Neighbour's Forecasters of Varying Reliability

Randomly Failing Forecaster: In this experiment, the neighbour's forecaster consists of ideal forecasters in one half of randomly chosen time moments and of meaningless forecasters in the remaining half. The distribution of these choices were uniform. It is tempting to expect that the proper weight given to the forecaster will be around 0.5.
Results: Fig. 5 shows a small initial rise of the median of the belief $\beta_t(w_3)$. Since $t = 25$, it decreases to 0, which reached around $t = 400$. The median of the belief $\beta_t(w_2)$ behaves similarly. It leads to the weight estimates decreasing to 0.

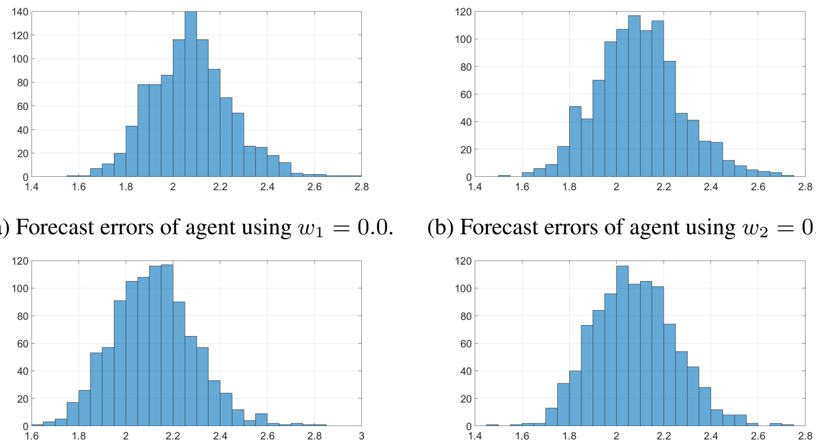


(a) Medians of beliefs in particular weights.



(b) Median of the weight estimates.

Fig. 5: shows medians of: (a) $\beta_t(w_1)$ ■, $\beta_t(w_2)$ ♦, $\beta_t(w_3)$ ▲. (b) the weight estimate (17). It reflects 10^3 MC runs with the **randomly failing** neighbour's forecaster.



(a) Forecast errors of agent using $w_1 = 0.0$. (b) Forecast errors of agent using $w_2 = 0.5$.

(c) Forecast errors of agent using $w_3 = 1.0$. (d) Forecast errors of the proposed way.

Fig. 6: shows counts of errors Δ (18) on the vertical axis and the values of Δ on the horizontal axis. It reflects 10^3 MC runs with the **randomly failing** neighbour's forecaster.

Fig. 6 presents forecast errors. The only visible difference in Fig. 6 seems to be in Fig. 6d exhibiting a smaller amount of outliers. This might be a random effect so that statistics in Tab. 3 are more informative. Fig. 7 shows beliefs in the respective weights at the ends of simulation runs.

Table 3: Forecast errors Δ (18) with the **randomly failing** neighbour's forecaster.

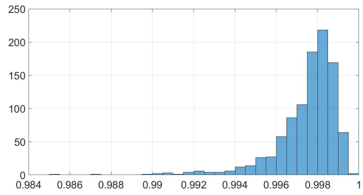
Forecaster	Median	Mean	STD	RMS
Agent using $w_1 = 0.0$	2.068	2.074	0.170	2.081
Agent using $w_2 = 0.5$	2.086	2.094	0.181	2.102
Agent using $w_3 = 1.0$	2.116	2.119	0.173	2.126
Proposed way	2.072	2.076	0.173	2.083

Discussion: Against the expectation, the ignoring of unreliable neighbour's forecaster is the optimal policy. The weight $w_1 = 0.0$ gives the best result. The proposed way converges to it giving the second best results.

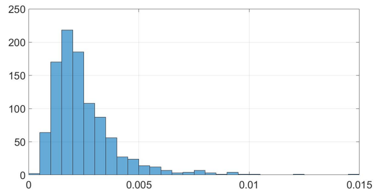
Improving Forecaster: In this experiment, the forecaster begins with a bad quality and slowly throughout the simulation it is improving towards ideal reliability. Again, it is tempting to expect that the weight estimate $\hat{w}_t$ will converge to one.

Results: Fig. 8 shows a quite volatile evolution of beliefs. They oscillate before reaching (probably) stabilised values. The oscillations project into the weight estimate (17).

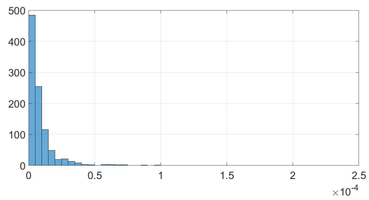
Tab. 4 summarises the forecast errors. It favourises to neglect the offered forecaster, $w_1 = 0.0$. The proposed way follows this and it is again the second best.



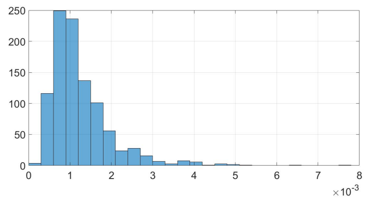
(a) Histogram of belief $\beta(w_1)$.



(b) Histogram of belief $\beta(w_2)$.

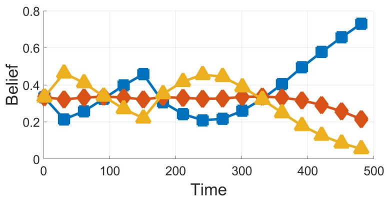


(c) Histogram of belief $\beta(w_3)$.

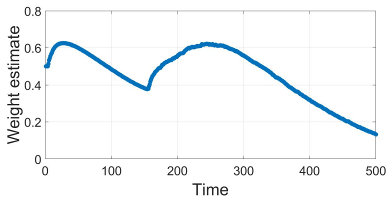


(d) Histogram of the weight estimate.

Fig. 7: shows counts of values on the vertical axis, the final values of $\beta_{|t|}$ and $\hat{w}_{|t|}$ on the horizontal axes. It reflects 10^3 MC runs with the **randomly failing** forecaster.



(a) Medians of beliefs in particular weights.

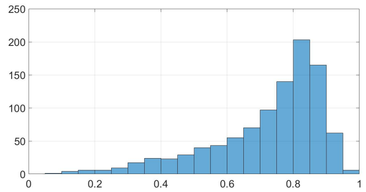


(b) Median of the weight estimates.

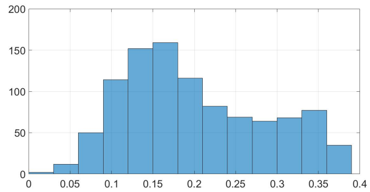
Fig. 8: shows medians of: (a) $\beta_t(w_1)$ ■, $\beta_t(w_2)$ ♦, $\beta_t(w_3)$ ▲. (b) the weight estimate (17). It reflects 10^3 MC runs with the **improving** neighbour's forecaster.

Table 4: Forecast errors Δ (18) with the **improving** forecaster.

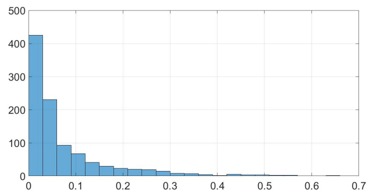
Forecaster	Median	Mean	STD	RMS
Agent using $w_1 = 0.0$	2.078	2.081	0.171	2.088
Agent using $w_2 = 0.5$	2.080	2.088	0.172	2.095
Agent using $w_3 = 1.0$	2.088	2.097	0.166	2.104
Proposed way	2.072	2.084	0.169	2.091



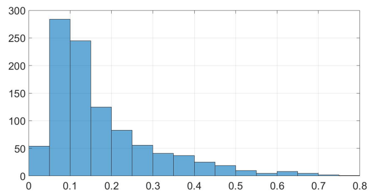
(a) Histogram of belief $\beta(w_1)$.



(b) Histogram of belief $\beta(w_2)$.



(c) Histogram of belief $\beta(w_3)$.



(d) Histogram of the weight estimate.

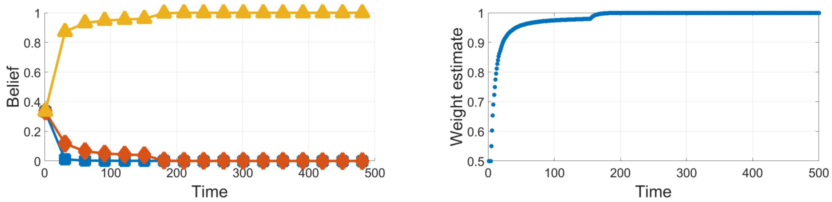
Fig. 9: shows counts of values on the vertical axis, the final values of $\beta_{|t|}$ and $\hat{w}_{|t|}$ on the horizontal axes. It reflects 10^3 MC runs with the **improving** neighbour's forecaster.

Fig. 9 confirms volatility of results in this scenario. It shows quite varying beliefs at the end of respective simulations.

Discussion: The results discard the over-simplified expectation formulated above. The estimation dynamics and the forecaster-quality changes influence the results in a quite complex way. This confirms the need to relax the invariance assumption, see Sec. 5.

Deteriorating Forecaster: In this experiment, the neighbour's forecaster started as the ideal one and gradually deteriorated into the bad forecaster. The weight estimate (17) was expected to rise rapidly to 1 and then to decline to 0.

Results: Fig. 10 confirms the expectation for the initial phase but the weight estimate does not track the deterioration and stay close to 1 until the simulation end.
Tab. 5 evaluates the forecast errors and shows that the best results are gained when using fully the neighbour's forecaster all the time. The proposed way follows this pattern.
Discussion: The experiment confirmed that over-simplified expectations are violated when the estimation dynamics and the neighbour's forecaster with a varying reliability are combined. This makes the further progress outlined in Sec. 5 inevitable.



(a) Medians of beliefs in particular weights. (b) Median of the weight estimates.

Fig. 10: shows medians of: (a) $\beta_t(w_1)$ ■, $\beta_t(w_2)$ ♦, $\beta_t(w_3)$ ▲. (b) the weight estimate (17). It reflects 10^3 MC runs with the **deteriorating** neighbour's forecaster.

Table 5: Forecast errors Δ (18) with the **deteriorating** forecaster.

Forecaster	Median	Mean	STD	RMS
Agent using $w_1 = 0.0$	2.076	2.081	0.173	2.088
Agent using $w_2 = 0.5$	2.043	2.046	0.168	2.053
Agent using $w_3 = 1.0$	1.999	2.006	0.163	2.013
Proposed way	2.002	2.009	0.161	2.015

5 Towards Handling Volatile Credibility

The assumed invariance of the estimated parameter fits to the assumed invariant trust weight. Adaptive systems [3] have a long tradition and experience how to cope with a slowly varying estimated parameter. Various types of forgetting (age-weighting) were proposed [22] and used even in connection with a trust handling [32].

The forgetting was recognised as a sort of flattening the evaluated posterior pds [16,17,18,23]. Thus, it can be directly applied both to p and w estimation, possibly using the idea of partial forgetting [6]. There are well-established rule of thumbs for the choice of forgetting factor. In critical cases, it may extend the estimated parameter, but it increases the computational complexity.

Possible abrupt changes of the estimated quantities were counteracted by adding a detector of such changes [9]. Recently, the problem was successfully and efficiently addressed by applying minimum *expected* relative principle [12]. Its tailoring to the discussed problem will be elaborated and published elsewhere.

6 Conclusions

Done: The paper contributes to a trustable knowledge sharing in a specific but widely applicable scenario. In it, a neighbour offers its forecaster of the observations handled by the supported agent. It complements the recent knowledge-sharing scenario [14] by the feasible estimation of the trust weight with which the neighbour's forecaster should be used. It primarily deals with the invariant weight quantifying the neighbour's credibility. The case fits the assumption that the parameter estimated by the agent is invariant. The performed, partially presented, experiments illustrate that the results are not extremely sensitive to this assumption. The proposed solution, presented experiments and the discussion in Sec. 5 prepare general solutions for slowly as well as abruptly varying credibility of the neighbour's forecaster.

A comment on related works: The used knowledge-sharing way, complemented by the above trust learning, has unique inter-related features: ► it combines pds operating on *partially* overlapping domains, i.e. the agent and neighbour process the knowledge quite freely; ► the roles of the agent and the neighbour may swap, i.e. their mutual trust may even substantially differ. Such a support of agents cooperating without a mediator allows an unlimited scalability of the network interacting adaptive agents.

Future work: The need for the cooperation respecting credibility of the shared knowledge and the positive experience with the presented results make worthwhile to:

- ✓ perform extensive experiments delimiting the applicability range of the proposed technique, cf. no free lunch theorem, [35];
- ✓ apply the technique to important particular cases, say selected according to [11,34];
- ✓ elaborate the general solution to linear-in-unknown-parameter Gaussian model, [24], which is an important EF member suitable for modelling of dynamic environments with continuous-valued observations [27];
- ✓ extend the technique to other models like mixtures of EF members [20,21] requiring an approximate recursive estimation, [2];
- ✓ tailor the technique to other knowledge-sharing scenarios, up to an algorithm comparison [30], requiring an estimation of the trust weight [8];
- ✓ complete solutions coping with the volatile trustability.

You are invited to contribute to this important research. We are ready to cooperate and DK will share the relevant experimental software with you.

References

1. Abramowitz, M., Stegun, I.: Handbook of Mathematical Functions. Dover Publ., N.Y. (1972)
2. Andrýsek, J.: Estimation of Dynamic Probabilistic Mixtures. Ph.D. thesis, FNSPE CTU, 18208 Prague 8, Czech Republic (2005)
3. Åström, K., Wittenmark, B.: Adaptive Control. Addison-Wesley (1994)
4. Barndorff-Nielsen, O.: Information and Exponential Families in Statistical Theory. Wiley, N.Y. (1978)
5. Berger, J.: Statistical Decision Theory and Bayesian Analysis. Springer (1985)
6. Dedecius, K., Nagy, I., M.Kárný: Parameter tracking with partial forgetting method. Int. J. Adapt Control Signal Process. **26**(1), 1–12 (2012)
7. Dorri, A., Kanhere, S., Jurdak, R.: Multi-agent systems: A survey. IEEE Access **6**, 28573 – 28593 (2018)
8. Drawel, N.: Model Checking Trust-based Multi-Agent Systems. Ph.D. thesis, The Concordia Institute for Information Systems Engineering (2019)
9. Holst, J., Poulsen, N.: Self tuning control of plants with abrupt changes. IFAC Proc. Volumes **17**(2), 923 – 928 (1984), 9th IFAC World Congr.
10. Jazwinski, A.: Stochastic Processes and Filtering Theory. Ac. Press (1970)
11. Jiao, J., Zhou, F., Gebraeel, N., Duffy, V.: Towards augmenting cyber-physical-human collaborative cognition for human-automation interaction in complex manufacturing and operational environments. Int. J. of Production Research **58**(16), 5089–5111 (2020)

12. Kárný, M.: Minimum expected relative entropy principle. In: Proc. of the 18th ECC. pp. 35–40. Sankt Petersburg (2020)
13. Kárný, M., Böhm, J., Guy, T., Jirsa, L., Nagy, I., Nedoma, P., Tesař, L.: Optimized Bayesian Dynamic Advising: Theory and Algorithms. Springer, London, UK (2006)
14. Kárný, M., Hůla, F.: Fusion of probabilistic unreliable indirect information into estimation serving to decision making. Intern. Journal of Machine Learning and Cybernetics (2021). https://doi.org/10.1007/s13042-021-01359-9
15. Kracík, J., Kárný, M.: Merging of data knowledge in Bayesian estimation. In: Filipe, J., et al (eds.) Proc. of the 2nd Int. Conf. on Informatics in Control, Automation and Robotics. pp. 229–232. Barcelona (2005)
16. Kulhavý, R.: Restricted exponential forgetting in real-time identification. Automatica **23**(5), 589–600 (1987)
17. Kulhavý, R., Kraus, F.J.: On duality of regularized exponential and linear forgetting. Automatica **32**, 1403–1415 (1996)
18. Kulhavý, R., Zarrop, M.B.: On a general concept of forgetting. Int. J. of Control **58**(4), 905–924 (1993)
19. Lainiotis, D.: Partitioned estimation algorithms, I: Nonlinear estimation. Inf. Sci. **7**, 203–235 (1974)
20. McLachlan, G., Peel, D.: Finite Mixture Models. Wiley Series in Probab. & Stat., Wiley, N.Y. (2000)
21. McNicholas, P.: Mixture model-based classification. CRC Press, Boca Raton, N.Y. (2017)
22. Milek, J., Kraus, F.: Time-varying stabilized forgetting for recursive least squares identification. In: Bányász, C. (ed.) IFAC Symp. ACASP'95, pp. 539–544. IFAC, Budapest (1995)
23. Peterka, V.: Subjective probability approach to real-time identification. In: The 4th IFAC Symp. on Identification and System Parameter Estimation, vol. 1, pp. 83–99. Tbilisi (1976)
24. Peterka, V.: Bayesian system identification. In: Eykhoff, P. (ed.) Trends & Progress in System Identification, pp. 239–304. Perg. Press (1981)
25. Podlesná, Y., Kárný, M.: Combination of forecasters in parameter estimation. Tech. Rep. 2385, ÚTIA AV ČR, Prague, Czech Republic (2020)
26. Quinn, A., Kárný, M., Guy, T.: Fully probabilistic design of hierarchical Bayesian models. Inf. Sci. **369**, 532–547 (2016)
27. Quinn, A., Kárný, M., Guy, T.: Optimal design of priors constrained by external predictors. Int. J. Approximate Reasoning **84**, 150–158 (2017)
28. Rao, M.: Measure Theory and Integration. J. Wiley (1987)
29. Savage, L.: Foundations of Statistics. Wiley (1954)
30. Schlender, T., Spanakis, G.: Thy algorithm shalt not bear false witness': An evaluation of multiclass debiasing methods on word embeddings. In: Baratchi, M., Cao, L., Kosters, W., Lijffijt, J., van Rijn, J., Takes, F. (eds.) BNAIC/Benelearn 2020. vol. 1398. Springer (2021)
31. Smith, A., Makov, U.: A quasi-Bayes sequential procedures for mixtures. J. of the Royal Statistical Society **40**(1), 106–112 (1978)
32. Vasilomanolakis, E., Habib, S., Milaszewicz, P., , Malik, R., Mühlhäuser, M.: Towards trust-aware collaborative intrusion detection: Challenges and solutions. In: Steghöfer, J., Esfandiari, B. (eds.) Trust Management XI. pp. 94–109. Springer Int. Publ. (2017)
33. Wang, P., Yang, L., Li, J., Chen, J., Hu, S.: Data fusion in cyber-physical-social systems: State-of-the-art and perspectives. Information Fusion **51**, 42 – 57 (2019)
34. Wang, Y.: Trust quantification for networked cyber-physical systems. IEEE Internet of Things Journal **5**(3), 2055–2070 (2018)
35. Wolpert, D., Macready, W.: No free lunch theorems for optimization. IEEE Trans. on Evolutionary Computation **1**(1), 67–82 (1997)