User's ideal
User's ideal pdf
Uth
user's ideal
user's ideal pdf
V
V0
W
wgs
Y
y0
y1
Z
zu
zy
zyy

alpha component weights

To simulate a two-component mixture, it is sufficient to set the value $\alpha_1 = \alpha \in [0, 1]$ defining the weight of the first component. The other component will have the weight $\alpha_2 = 1 - \alpha$.

cchns channel in condition

Vector, possibly empty, containing indexes of the channels in condition, used in mixture projection (conditioning, marginalization). Formally, let m contain indexes of some modelled channels $\{1, \ldots, \mathring{d}\}$ and $\overline{m}$ contain the rest of the system channels. Then, the evaluated mixture projection is the pdf $f(d_{m;t}|cchns, d(t-1))$ with $cchn = d_{\overline{m}}$.

cove noise covariance

cove > 0 defines covariance of zero mean white Gaussian noise. The inequality > 0 expresses positive-definiteness in matrix case.

covy output covariance

covy $\equiv r > 0$ defines covariance of Gaussian output. The inequality > 0 expresses positive-definiteness in matrix case.

covu input covariance

covu $\equiv r > 0$ defines covariance of zero mean white Gaussian noise used as the system input. The inequality > 0 expresses positive-definiteness in matrix case.

Cth L'DL decomposition of the covariance of least squares estimates of regression coefficients.

diac diagonal element of covariance matrix

diac > 0 defines the constructed covariance matrix as diac$\times$unit matrix.

diam common diagonal value of the table specifying transition probabilities between components diam $\in (0,1)$ and $\overline{diam} \equiv \frac{1-diam}{number\ of\ components}$ define the following transition probabilities:

$$\begin{bmatrix} diam & \overline{diam} & \overline{diam} & \dots & \overline{diam} & \overline{diam} \\ \overline{diam} & diam & \overline{diam} & \dots & \overline{diam} & \overline{diam} \\ \vdots & & \ddots & & & \vdots \\ \overline{diam} & & \dots & & \overline{diam} & diam \end{bmatrix}.$$

This Markov dependence of component occurrence is used in realistic simulations of dynamic systems.

diaCth common diagonal value of the covariance matrix Cth of least squares estimates of regression coefficients. diaCth > 0 defines the constructed covariance matrix as diaCth$\times$unit matrix.

dfm degrees of freedom

dfm > 0 can be interpreted as an effective number of data processed for parameter estimation. More formally, dfm $\equiv \nu - 2$ is the number of degrees of freedom co-defining the parameter estimate, which is conjugated to exponential family of parameterized models.

Eth point estimate of unknown parameters:
For *Gaussian model*, they are regression coefficients.
For *Markov chain model* they are transition probabilities.

Ealpha point estimate of component weight

Usually, Ealpha coincides with the expected value of alpha conditioned by the available experience.

frg forgetting rate

frg $= \lambda \in (0, 1]$ specifies probability that the estimated parameters are time-invariant.

frga alternative forgetting rate

frga $\in (0, 1]$ denotes forgetting rate of an alternative model, i.e. specifies the probability that the estimated parameters of the alternative model are time invariant. Its use in approximate estimation is called branching by forgetting.

gain prior knowledge on static gain of the system

gain $= [\underline{g}, \overline{g}]$ =[lower bound, upper bound] on a priori expected range of the static gain.

hor optimization horizon

the design is performed till $hor_1 \times hor_2$, where hor_2 determines grouping extent enforced by the requirement that the recommended pointers should not change for hor_2 steps. Omission of hor_2 is identical with setting $hor_2 = 1$, i.e. no grouping is required.

ipar switch selecting the type of the simulated system

ipar = 1 the simulated system exhibits almost deterministic response
ipar = 2 the simulated system response is uncertain but controllable
ipar = 3 the simulated system response is uncertain and poorly controllable.

n number of cutting points

n $= \mathring{\tau}$ is the number of cutting points defining uniform grid on time scale. The data records till a cutting point are used for learning and the rest for model validation.

ndat number of data records

ndat $\equiv \mathring{t}$ (> highest model order) determines the number of data records d_t, each labelled by discrete time $t = 1, \ldots, \mathring{t}$.

ncom number of components

ncom $= \mathring{c}$ specifies the number of uni-modal pdfs called components forming a finite probabilistic mixture.

ncomi initial guess of the number of components

ncomi $\in \{1, 2, \ldots\}$ is the initial guess of the unknown number of mixture components ncom $= \mathring{c}$.

niter number of iterations

niter $\equiv \mathring{n} \in \{1, 2, \ldots\}$ is the number of iterations used in iterative learning.

niteri number of nested iterations

niteri $\equiv \mathring{n} \in \{1, 2, \ldots\}$ is the number of iterations, used in iterative learning, applied whenever iterations are nested.

nitere number of nested iterations

nitere $\equiv \mathring{n} \in \{1, 2, \ldots\}$ is the number of iterations, used in iterative learning, applied whenever iterations are nested.

nchn number of modelled channels

nchn $= \mathring{\Delta} \in \{1, 2, \ldots, \mathring{d}\}$ is dimension of predicted and thus modelled data entries, called in software context channelcolorblues.

nsk parameter used in data grouping

nsk $\in \{1, 2, \ldots\}$ number of data stacked, see grouped data.

nstep number of steps

nstep $\in \{1, 2, \ldots\}$ is prediction horizon.

ny number of possible values of the discrete output
y. The considered discrete-valued $\mathrm{y} \in \{1, 2, \ldots \mathring{y}\}$, $\mathring{y} \equiv \mathrm{ny} > 1$.

nu number of possible values of the discrete input
u. The considered discrete-valued u $\in \{1, 2, \ldots \mathring{u}\}$, $\mathring{u} \equiv$ nu > 1.

nhor receding design horizon

is the key parameter of the receding horizon strategy that performs optimization nhor step ahead, applies the first decision and, ideally with enriched experience, repeats the optimization.

opt computational options

opt is the common name for a group of optional, case specific, parameters of various functions.

ord model order

ord $\equiv \partial \in \{0, 1, \ldots\}$ determines the number of delayed data records of the state ϕ_{t-1}, i.e. $\phi_{t-1} = [d_{t-1}, \ldots, d_{t-\partial}, 1]$.

ordi initial guess of model order

ordi $\in \{0, 1, \ldots\}$ is an initial guess at the estimated model order ord.

om input penalization

The partial loss in additive quadratic loss contains the term $\text{input}^2 \times \text{om}$, $\text{om} \in (0, \infty]$.

pchns predicted channels

pchns is a vector storing indexes of channels, whose values are predicted. Formally, let m contains channel indexes in $\{1, \ldots, \mathring{d}\}$. Then $f(d_{m;t}|d(t-1))$ is computed.

psi0 value of zero-delayed regressor

psi0 is a (possibly empty) sub-vector of non-delayed part of regression vector. Its values are supplied externally when predicting.

pau pause between subsequent figures (in seconds) rg range of uniformly distributed random input

```
u = 2*rg*(rand-0.5)
```

ro threshold value

ro> 0 denotes a threshold value, its particular meaning depends on the function, which uses it.

scale switch of data scaling

scale $\in \{0, 1\} \equiv \{$do not scale, scale$\}$ the processed data.

seed seed of the random generator

seed $\in \{1, 2, \ldots\}$ fixes a realization of random-numbers generator. For seed $= 0$ the realizations are not fixed.

typ switch between different types of processing

typ $\in \{0, 1, 2, \ldots\}$; particular set of possible values of typ depends on the place where it is used.

th parameter

th is software name for a multivariate parameter θ.
For Gaussian model, it coincides with regression coefficients.
For Markov chain model, it denotes free parameters of the transition probability. For instance, if $d_t \in \{1, 2\}$, then the vector th contains probabilities of $d_t = 1$ for respective discrete values of regression vector.

Uth parameters of the true user's ideal pdf
For *Gaussian model*, they are means of the true user's ideal pdf ${}^{\lfloor U}f(d_{o;t}|d(t-1)) = \mathcal{N}_{d_{o;t}}(Uth, Ucov)$.
Thus, the desired range of quantities of interest is $d_{oi;t} \in [Uth_i - \sqrt{Ucov_{ii}}, Uth_i + \sqrt{Ucov_{ii}}]$.
For *Markov chain model*, they completely define the desired transition probabilities. For instance, if $d_t \in \{1, 2\}$, then the vector Uth contains probabilities of $d_t = 1$ for respective discrete values of regression vector.

Ucov covariance matrix of the true user's ideal pdf

$$^{\lfloor U}f(d_{o;t}|d(t-1)) = \mathcal{N}_{d_{o;t}}(Uth, Ucov)$$

Ufc user's ideal pdf on recommended pointers

This design tuning knob specifies preferences among components. Time and data invariant version of Ufc $\equiv {}^{\lfloor U}f(c_t) \equiv {}^{\lfloor U}f(c_t|d(t-1))$ is considered. The exclusion of a dangerous component $\bar{c}$ is reached by setting ${}^{\lfloor U}f(c_t = \bar{c}) = 0$.

V0 prior statistics

V0 $\equiv V_0 > 0$ is initial occurrence table used in Markov chain estimation and extended information matrix in estimation of normal models.

wgs switch for use of data dependent weights in prediction

wgs $\in \{0, 1\} \equiv \{$do not use, use$\}$ a heuristic predictor with data dependent weights.

y1 one-step delayed output, i.e. y_{t-1}.

y0 initial output

y0 = y_0 specifies initial value of the output for the first order model.

zyy partial loss of output increment

zyy ≥ 0 penalizes the increment $y_t - y_{t-1}$ of system output.

zy partial loss of output

zy ≥ 0 penalizes the system output y_t.

zu partial loss of input

v-log-likelihood is obtained as the product of one-step-ahead predictors with inserted measured data. It serves as an universal measure of model quality and represents posterior probability of data-records sequence. Model of a better quality possesses a larger value of the v-log-likelihood.

normalized v-log-likelihood is v-log-likelihood normalized by the number of data records processed. It compensates for its almost linear increase with time and allows to compare models obtained for different numbers of data records.

Normalized v-log-likelihood is v-log-likelihood normalized by the number of data records processed. It compensates for its almost linear increase with time and allows to compare models obtained for different numbers of data records.

prediction-errors norm serves for judgement of predictive abilities of the inspected model. It is defined as sample mean of squared prediction errors normalized by variance of the predicted channel. This scale invariant index indicates gain in using the inspected model comparing to plain prediction by sample mean of data values. It is computed as follows:

```
yp  = ...                       %
dd  = ...                       %
ep  = dd-yp;                    %
sy  = std(dd');                 %
er  = std(ep');                 %
se  = (er./sy)';                %
se  = se+abs(mean(ep'))';       %
```

The function `mean` computes mean of a trajectory, `std` computes standard deviation.

Prediction-errors norm serves for judgement of predictive abilities of the inspected model. It is defined as sample mean of squared prediction errors normalized by variance of the predicted channel. This scale invariant index indicates gain in using the inspected model comparing to plain prediction by sample mean of data values. It is computed as follows:

```
yp  = ...                    %
dd  = ...                    %
ep  = dd-yp;                 %
sy  = std(dd');              %
er  = std(ep');              %
se  = (er./sy)';             %
se  = se+abs(mean(ep'))';    %
```

The function mean computes mean of a trajectory, std computes standard deviation.

Markov jumps among components is a simulation option. This option can be applied to a multi-component mixture. The components are simulated in usual way, but the components' switching follows Markov chain, defined by the table of transition probabilities. The algorithm slows down a wild switching of components typical for the assumed fully independent jumps. The algorithm provides better description of reality as a multi-modal dynamic system often stays around a single mode over some period of functioning.

grouped data preprocessing creates new data records $^{\lfloor groupped}d'_t = [d'_t, d'_{t-1}, \ldots, d'_{t-n}]$, $n =$ nsk, consisting of groups of original data records $d_\tau,\ \tau \le t$. Grouping serves for predictions with dynamic component weights obtained after conditioning of model of grouped data with static weights. The extreme grouping of all data records influencing the data vector Ψ_t into a single grouped data vector $^{\lfloor groupped}d_t = \Psi_t$ allows to treat dynamic models completely as static ones.

The way of grouping is seen on the following example:

```
DATA =
     1     2     3     4     5     6     7     8     9    10
    11    12    13    14    15    16    17    18    19    20

... grouping operation

DATA =
     1     2     3     4     5     6     7     8     9
    11    12    13    14    15    16    17    18    19
     2     3     4     5     6     7     8     9    10
    12    13    14    15    16    17    18    19    20
```

Grouped data preprocessing creates new data records ${}^{\lfloor groupped}d_t' = [d_t', d_{t-1}', \ldots, d_{t-n}']$, $n =$ nsk, consisting of groups of original data records $d_\tau,\ \tau \le t$. Grouping serves for predictions with dynamic component weights obtained after conditioning of model of grouped data with static weights. The extreme grouping of all data records influencing the data vector Ψ_t into a single grouped data vector ${}^{\lfloor groupped}d_t = \Psi_t$ allows to treat dynamic models completely as static ones.

The way of grouping is seen on the following example:

```
DATA =
     1     2     3     4     5     6     7     8     9    10
    11    12    13    14    15    16    17    18    19    20

... grouping operation

DATA =
     1     2     3     4     5     6     7     8     9
    11    12    13    14    15    16    17    18    19
     2     3     4     5     6     7     8     9    10
    12    13    14    15    16    17    18    19    20
```

multi-step prediction

The following diary documents the prediction several steps ahead

```
DATA=1; for t=2:8, DATA(t)=DATA(t-1)*0.9999; end
-----------------------------------------------------------------
TIME                  3
DATA    1  0.9999  0.9998  0.9997  0.9996  0.9995  0.9994  0.9993
prediction         0.9998  0.9997  0.9996  0.9995  0.9994
nstep                1       2       3       4       5
-----------------------------------------------------------------
```

Multi-step prediction

The following diary documents the prediction several steps ahead

```
DATA=1; for t=2:8, DATA(t)=DATA(t-1)*0.9999; end
-----------------------------------------------------------------
TIME                  3 DATA    1  0.9999  0.9998  0.9997  0.9996
0.9995  0.9994  0.9993 prediction         0.9998  0.9997  0.9996
0.9995  0.9994 nstep                 1       2       3       4
5
-----------------------------------------------------------------
```

richest model order is used in connection with structure estimation. Structure estimation maximizes a posteriori probability of hypotheses about structures of the involved regression vectors. These hypotheses represent possible regression vectors taken from the richest regression vector corresponding to the richest data vector. The richest data vector contains current data record and delayed ones up to the *richest model order*.

Richest model order is used in connection with structure estimation. Structure estimation maximizes a posteriori probability of hypotheses about structures of the involved regression vectors. These hypotheses represent possible regression vectors taken from the richest regression vector corresponding to the richest data vector. The richest data vector contains current data record and delayed ones up to the *richest model order*.

iugn selects the following, pre-prepared, input generators in experiments with controlled Markov chain stimulated by discrete valued inputs.

iugn	**meaning**	
1	all ones	$u_t = 1$
2	all twos	$u_t = 2$
3	last output	$u_t = y_{t-1}$
4	minus last output	$u_t = 3 - y_{t-1}$
5	random	$u_t =$ round(rand)+1

where rand produces random mutually independent numbers, uniformly distributed on (0,1) and round convert them either to 0 or 1.

systemp System characteristics

```
function [Eth,str,ts,uref,yref,ssig,sigma,gain,tcons,msig,wcutc,freqc]= system1
msig=0;
chanu = 2;

uref=0;  yref=0;

A=[1.0000   -2.7145    2.4562   -0.7408];
B=[0.0001547   0.0005740    0.0001331 0];
ts    = 0.1;
ssig  = sum(B)/2;
sigma = sum(B)/2;

lendata = 100;        %
lenpri  = 1000;       %
lenst   = 100;        %
randn('seed', 123);   %

gain =(B*ones(max(size(B)),1))/(A*ones(max(size(A)),1));
lA=length(A); lB=length(B);
while  B(lB)==0,
   lB=lB-1;
end

if chanu==1, Eth  = [B(1:lB), -A(:,2:length(A)) ];
else
   Eth  = [-A(:,2:length(A)) B(1:lB)];
end

if chanu==1
   str1 = [1+ones(1,lB), ones(1,lA-1)];
   str2 = [0:(lB-1), 1:(lA-1)];
else
   str1 = [ones(1,lA-1),1+ones(1,lB)];
   str2 = [1:(lA-1),0:(lB-1)];
end
str  = [str1;str2];
str_org=str;
  ii= find(str(1,:)==1);
   a  = Eth(ii);
```

```
   A  = [1, -a];
   rr  = roots(A); r=[];
   len=length(rr); j=1;
   while 1
       if isreal(rr(j)), r=[r,rr(j)];      j=j+1;
       else,             r=[r,abs(rr(j))]; j=j+2;
       end
       if j>len, break; end
   end
   jj = find(abs(r)>1e-10);
   r  = r(jj);
   r  = -log(r);
   r  = abs(r);
   t1 = 1/r(1) * ts;
   if length(r)>1,
      t2 = 1/r(2) * ts;
      if abs(t1-t2)>1e-4
         if t1>t2, tt=[t2,t1]; else, tt=[t1,t2]; end
      end
   else, tt = t1;
   end

   if length(tt)>1
     tcons=[tt(1)*0.99, tt(1)*1.01, tt(2)*0.99, tt(2)*1.01];
     else tcons=[tt*0.99 tt*1.1];
   end

sysd=tf(B, A,ts );
sysc=d2c(sysd);
[num,den]=tfdata(sysc,'v');

[m, ph,w]=bode(sysc);
kk=find(m<=0.01*m(1)); wcutc=w(kk(1));
ll=find(m<=0.5623*m(1)); freqc=w(ll(1));
```

General conventions

f *is the letter reserved for probability (density) functions (p(d)f).*

x^* *denotes the range of* x, $x \in x^*$.

$\mathring{x}$ *denotes the number of members in the countable set* x^* or the *number of entries in the vector* x.

$\equiv$ *means the equality by definition.*

x_t *is a quantity* x *at the discrete time instant labelled by* $t \in t^* \equiv \{1, \ldots, \mathring{t}\}$.

$\mathring{t} \le \infty$ *is called (decision, learning, prediction, control, advising)* horizon.

$x_{i;t}$ *is an* i*-th entry of the array* x *at time* t.

$x(k \ldots l)$ *denotes the sequence* $x_k, \ldots, x_l$, *i.e.* $x(k \ldots l) \equiv x_k, \ldots, x_l$ *for* $k \le l$.

$x(k \ldots l)$ *is an empty sequence and reflects just the prior information if* $l < k$

$x(t) \equiv x(1 \ldots t) \equiv x_1, \ldots, x_t$ is the sequence from the initial time moment till time instance t.

$x_{k \ldots lc} \equiv x_{kc \ldots lc}$ *denotes the sequence* $x_{kc}, \ldots, x_{lc}$.

$\mathrm{supp}\,[\,f(x)]$ *is the support of the pdf* $f : x^* \to [0, \infty]$, *i.e. the subset of* x^* *on which* $f(x) > 0$.

$\setminus$ is the set subtraction or an omission of a term from a sequence.

System is the part of the world that is of interest for a decision maker who should either describe or influence it.

system is the part of the world that is of interest for a decision maker who should either describe or influence it.

Decision (or equivalently action) is the value of a quantity that can be directly chosen by the decision maker for reaching decision maker's aims.

decision (or equivalently action) is the value of a quantity that can be directly chosen by the decision maker for reaching decision maker's aims.

Ignorance concerns knowledge considered by the decision maker but unavailable for the choice of the decision.

ignorance concerns knowledge considered by the decision maker but unavailable for the choice of the decision.

Experience is knowledge about the system, available to the decision maker, for the selection the particular decision.

experience is knowledge about the system, available to the decision maker, for the selection the particular decision.

Loss function quantifies the degree of achievement of the decision-making aim by assigning to each realization of behavior a number on real line.

loss function quantifies the degree of achievement of the decision-making aim by assigning to each realization of behavior a number on real line.

Quantity is a multivariate mapping (corresponding to the notion of random variable). Its value for a fixed argument is called *realization*.

quantity is a multivariate mapping (corresponding to the notion of random variable). Its value for a fixed argument is called *realization*.

Behavior consists of all possible realizations, i.e. values of all quantities considered by the decision maker within the time span determined by the horizon of interest and related to the system.

behavior consists of all possible realizations, i.e. values of all quantities considered by the decision maker within the time span determined by the horizon of interest and related to the system.

Input (more precisely system input) is a decision, which is supposed to influence the ignorance part of the behavior.

input (more precisely system input) is a decision, which is supposed to <u>influence</u> the ignorance part of the behavior.

Output (more precisely system output) is an observable quantity that provides the decision maker information about the system behavior.

output (more precisely system output) is an observable quantity that provides the decision maker information about the system behavior.

Innovation belongs to the ignorance of the current action and to the experience of any subsequent action.

innovation belongs to the ignorance of the current action and to the experience of any subsequent action.

Decision rule is a mapping that assigns a decision to the behavior.

decision rule is a mapping that assigns a decision to the behavior.

Causal decision rule assigns the decision to experience.

causal decision rule assigns the decision to experience.

Strategy is a sequence of decision rules.

strategy is a sequence of decision rules.

Causal strategy is a sequence of causal decision rules.

causal strategy is a sequence of causal decision rules.

Admissible strategy is *causal* strategy that meets physical constraints, i.e. the ranges of its decision rules are in pre-specified subsets of respective sets of decisions.

Admissible strategies are *causal* strategies that meet physical constraints, i.e. the ranges of the decision rules are in pre-specified subsets of respective sets of decisions.

admissible strategy is *causal* strategy that meets physical constraints, i.e. the ranges of its decision rules are in pre-specified subsets of respective sets of decisions.

admissible strategies are *causal* strategies that meet physical constraints, i.e. the ranges of the decision rules are in pre-specified subsets of respective sets of decisions.

Advising strategy is the admissible strategy of advisory system generating advices, i.e. its actions (decisions). For academic design, advices are the recommended pointers c_t to components and the strategy is described by the pdf ${}^{\lfloor I}f(c_t|d(t-1))$. For industrial design, advices consist of recommended recognizable actions $u_{o;t}$ and the strategy is described by the pdf ${}^{\lfloor I}f(u_{o;t}|d(t-1))$. The simultaneous design provides the advising strategy ${}^{\lfloor I}f(c_t, u_{o;t}|d(t-1))$ generating both of them.

advising strategy is the admissible strategy of advisory system generating advices, i.e. its actions (decisions). For academic design, advices are the recommended pointers c_t to components and the strategy is described by the pdf ${}^{\lfloor I}f(c_t|d(t-1))$. For industrial design, advices consist of recommended recognizable actions $u_{o;t}$ and the strategy is described by the pdf ${}^{\lfloor I}f(u_{o;t}|d(t-1))$. The simultaneous design provides the advising strategy ${}^{\lfloor I}f(c_t, u_{o;t}|d(t-1))$ generating both of them.

advising strategies are the admissible strategies of advisory system generating advices, i.e. its actions (decisions). For academic design, advices are the recommended pointers c_t to components and the strategy is described by the pdf ${}^{\lfloor I}f(c_t|d(t-1))$. For industrial design, advices consist of recommended recognizable actions $u_{o;t}$ and the strategy is described by the pdf ${}^{\lfloor I}f(u_{o;t}|d(t-1))$. The simultaneous design provides the advising strategy ${}^{\lfloor I}f(c_t, u_{o;t}|d(t-1))$ generating both of them.

Advising strategies are the admissible strategies of advisory system generating advices, i.e. its actions (decisions). For academic design, advices are the recommended pointers c_t to components and the strategy is described by the pdf ${}^{\lfloor I}f(c_t|d(t-1))$. For industrial design, advices consist of recommended recognizable actions $u_{o;t}$ and the strategy is described by the pdf ${}^{\lfloor I}f(u_{o;t}|d(t-1))$. The simultaneous design provides the advising strategy ${}^{\lfloor I}f(c_t, u_{o;t}|d(t-1))$ generating both of them.

Decision-making means design and application of decision rules.

decision-making means design and application of decision rules.

DM abbreviates decision-making

Channel denotes a particular data record entry. It is used mainly in software context.

channel denotes a particular data record entry. It is used mainly in software context.

Component is a uni-modal pdf. A finite probabilistic mixture is composed of the components entering with respective weights.

component is a uni-modal pdf. A finite probabilistic mixture is composed of the components entering with respective weights.

Factor is a pdf describing particular data entry within a component.

factor is a pdf describing particular data entry within component.

KL divergence is Kullback–Leibler divergence of pdf $f(x)$ from pdf $g(x)$ defined as $D(f||g) \equiv \int_{x^*} f(x) \ln \left(\frac{f(x)}{g(x)} \right) \, dx$

KLD is Kullback–Leibler divergence of pdf $f(x)$ from pdf $g(x)$ defined as $D(f||g) \equiv \int_{x^*} f(x) \ln\left(\frac{f(x)}{g(x)}\right) \, dx$

Kullback–Leibler divergence of pdf $f(x)$ from pdf $g(x)$ is defined by $D(f||g) = \int_{x^*} f(x) \ln\left(\frac{f(x)}{g(x)}\right)\, dx$

CT model
Coin Tossing (CT) model is described by the pdf

$$f(y_t|\theta) = \theta^{\delta_{y_t,1}}(1-\theta)^{\delta_{y_t,2}}$$

where

$y_t \in y^* = \{1, 2\}$ and $u_t \in u^* = \{1, 2\}$, $t \in t^*$ are output and input;

$\theta \in (0, 1)$ is a parameter of the CT model;

$\delta_{y_t,k}$ is Kronecker symbol.

CT parameter

Parameter of the CT model generally is denoted by θ, its Matlab transcription is “th”. It represents probability of obtaining “head” after a coin toss. Indeed, the probability of “tail” is $1-\theta$. It must hold

$$\theta \in (0,1)$$

CCM system
Controlled Coin with Memory System models controlled experiment of tossing of a coin. Particular tosses are independent and probabilities on different outputs are not uniform. The system has discrete two-valued output $y_t \in \{1, 2\}$ and input $u_t \in \{1, 2\},\ t = 1, 2, \ldots, \mathring{t}$.

CCM model is described by the pdf

$$f(y_t|\psi_t,\Theta) = \prod_{\psi\in\psi^*} \left[\Theta_{1|\psi}^{\delta_{y_t,1}} \Theta_{2|\psi}^{\delta_{y_t,2}}\right]^{\delta_{\psi_t,\psi}}$$

where

$y_t \in y^* = \{1,2\}$ and $u_t \in u^* = \{1,2\},\ t \in t^*$ are output and input;

$\psi_t \equiv [u_t, y_{t-1}]' \in \psi^* = \{[1,1]', [1,2]', [2,1]', [2,2]'\}$ is regression vector;

$\Theta = \Theta_{y|\psi}$ is a parameter of the CCM model, see CCM parameter;

$\delta_{\bullet,*}$ is Kronecker symbol.

The set of possible model parameters Θ^* is defined by Cartesian product of sets of conditional probabilities $\Theta_\psi^* \subset \{[\Theta_{1|\psi}, \Theta_{2|\psi}]:\ \Theta_{1|\psi} > 0, \Theta_{2|\psi} > 0, \Theta_{1|\psi} + \Theta_{2|\psi} = 1\}$:

$$\Theta^* = \bigotimes_{\psi} \Theta_\psi^*.$$

th1

The parameter of the CCM model contains constant probabilities of the coin output y_t conditioned on the regression vector ψ_t. Parameter Θ is described by (4×2) matrix with rows containing probabilities of the particular outputs, i.e.

$$\Theta = f(y_t|u_t, y_{t-1}) = f(y_t|\psi_t)$$

$\psi_t = [u_t,\ y_{t-1}]$	$y_t = 1$	$y_t = 2$
$[1,\ 1]$	$\Theta_{1\|11}$	$\Theta_{2\|11} = 1 - \Theta_{1\|11}$
$[1,\ 2]$	$\Theta_{1\|12}$	$\Theta_{2\|12} = 1 - \Theta_{1\|12}$
$[2,\ 1]$	$\Theta_{1\|21}$	$\Theta_{2\|21} = 1 - \Theta_{1\|21}$
$[2,\ 2]$	$\Theta_{1\|22}$	$\Theta_{2\|22} = 1 - \Theta_{1\|22}$

The Θ is fully determined by its first column $\Theta_1 \geq 0$ (entry-wise), which is labelled *th1* in software context.

CCM parameter

The multivariate parameter of the CCM model contains constant probabilities of the coin output y_t conditioned on the regression vector ψ_t. Θ is described by (4×2) matrix with rows containing probabilities of the particular outputs, i.e.

$$\Theta = f(y_t|u_t, y_{t-1}) = f(y_t|\psi_t)$$

$\psi_t = [u_t,\ y_{t-1}]$	$y_t = 1$	$y_t = 2$
$[1,\ 1]$	$\Theta_{1\|11}$	$\Theta_{2\|11} = 1 - \Theta_{1\|11}$
$[1,\ 2]$	$\Theta_{1\|12}$	$\Theta_{2\|12} = 1 - \Theta_{1\|12}$
$[2,\ 1]$	$\Theta_{1\|21}$	$\Theta_{2\|21} = 1 - \Theta_{1\|21}$
$[2,\ 2]$	$\Theta_{1\|22}$	$\Theta_{2\|22} = 1 - \Theta_{1\|22}$

The matrix is fully determined by its first column called "th1" ≥ 0 (entry-wise), which is labelled *th1* in software context.

CCM prior

Prior pdf for the CCM model has Dirichlet pdf 10.1{380}

$$f(\Theta) = Di_{\Theta}(V) \propto \prod_{\psi} \Theta_{1|\psi}^{V_{1|\psi;0}-1} \Theta_{2|\psi}^{V_{2|\psi;0}-1} \chi_{\Theta^*}(\Theta),$$

with $\propto$ denoting proportionality, the indicator $\chi_{\Theta^*}(\Theta)$ of Θ^* and the occurrence table

$$V_0 = V_{y|\psi;0} : \quad \begin{array}{|c||c|c|} \hline \psi_t = [u_t, y_{t-1}] & y_t = 1 & y_t = 2 \\ \hline \hline 1 \quad 1 & V_{1|11} & V_{2|11} \\ 1 \quad 2 & V_{1|12} & V_{2|12} \\ 2 \quad 1 & V_{1|21} & V_{2|21} \\ 2 \quad 2 & V_{1|22} & V_{2|22} \\ \hline \end{array} .$$

Particular entries of the occurrence table express optional prior knowledge about estimated parameters.

CCM loss

Loss function corresponding to the CCM model is considered in additive form with the partial loss z given by the following table

$$z_{y|\psi}: \quad \begin{array}{|cc||c|c|} \hline u_t, & y_{t-1} & y_t = 1 & y_t = 2 \\ \hline 1 & 1 & z_{1|11} & z_{2|11} \\ 1 & 2 & z_{1|12} & z_{2|12} \\ 2 & 1 & z_{1|21} & z_{2|21} \\ 2 & 2 & z_{1|22} & z_{2|22} \\ \hline \end{array}.$$

The entries of the loss function $z_{y|\psi} \geq 0$ express user's aims and is subject to change (user's option).

CCM posterior

Posterior pdf for the CCM model has the form of Dirichlet pdf 10.1{384} as well:

$$f(\Theta|d(t)) \propto \prod_{\psi} \Theta_{1|\psi}^{V_{1|\psi;t}-1} \Theta_{2|\psi}^{V_{2|\psi;t}-1}, \ \propto \text{ means proportionality},$$

with the following updating of $V_{y_t|\psi_t;t}$, for $t = 1, 2, \ldots, \mathring{t}$, starting from the prior occurrence table $V_{y_t|\psi_t;0}$

$$V_{y_t|\psi_t;t} = V_{y_t|\psi_t;t-1} + 1.$$

CCM predictive pdf
Predictive pdf for the CCM model is

$$f(y_{t+1}|\psi_{t+1}, d(t)) = \frac{V_{y_{t+1}|\psi_{t+1};t}}{\sum_{y_t \in y^*} V_{y_{t+1}|\psi_{t+1};t}}$$

where $\psi_{t+1} = [u_{t+1}, y_t]$ and the set y^* contains possible values of the output y_t. For CCM model, $y^* \equiv \{1, 2\}$.

CCM prediction

Point prediction with the CCM model, which minimizes squared deviation of the prediction and predicted output, is

$$\hat{y}_{t+1} = \mathcal{E}[y_{t+1}|\psi_{t+1}, d(t)] = \frac{1 \times V_{1|\psi_{t+1};t} + 2 \times V_{2|\psi_{t+1};t}}{V_{2|\psi_{t+1};t} + V_{1|\psi_{t+1};t}}.$$

CCM hypotheses

Test of hypotheses about the CCM model $\Theta \in \Theta_h^* = (\theta_{\psi,h}^l, \theta_{\psi,h}^u)$ with prior pdf on hypotheses $f(h)$ has the form

$$\begin{aligned} f(h|d(t)) &= \frac{\mathcal{I}_h(d(t))}{\mathcal{I}_h(d(t-1))} f(h), \\ \mathcal{I}_h(d(t)) &= \prod_{\psi} \Big[\mathcal{B}_{V_{\mathbf{1}|\psi;t}, V_{\mathbf{0}|\psi;t}}(\theta_{\psi,h}^u) - \mathcal{B}_{V_{\mathbf{1}|\psi;t}, V_{\mathbf{0}|\psi;t}}(\theta_{\psi,h}^l) \Big] \end{aligned}$$

CCM control
One-step-ahead control with known CCM model parameters and for given CCM loss function $\mathcal{Z}(d(t)) = z_{y_t,u_t}$, the control criterion can be defined as

$$\mathcal{E}[Z(d(t))|u, \phi_{t-1}] = \mathcal{V}_{u;\phi_{t-1}} = z_{1,u}\Theta_{1|u,\phi_{t-1}} + z_{2,u}\Theta_{2|u,\phi_{t-1}}.$$

The optimal one-step-ahead strategy generates the inputs according to the rule

$$u_t = \begin{cases} 2 & \text{for} \quad \mathcal{V}_{1;\phi_{t-1}} \geq \mathcal{V}_{2;\phi_{t-1}} \\ 1 & \text{for} \quad \mathcal{V}_{1;\phi_{t-1}} < \mathcal{V}_{2;\phi_{t-1}}. \end{cases}$$

quasi-Bayes is basic algorithm for approximate mixture estimation, see Section 6.5.1{156} and Section 8.5.1{291}.

Quasi-Bayes is basic algorithm for approximate mixture estimation, see Section 6.5.1{156} and Section 8.5.1{291}.

QB is basic algorithm for approximate mixture estimation, see Section 6.5.1{156} and Section 8.5.1{291}.

batch quasi-Bayes is processing-order-independent algorithm for approximate mixture estimation, see Section 6.5.3{163} and Section 8.5.3{295}.

Batch quasi-Bayes is processing-order-independent algorithm for approximate mixture estimation, see Section 6.5.3{163} and Section 8.5.3{295}.

BQB is processing-order-independent algorithm for approximate mixture estimation, see Section 6.5.3{163} and Section 8.5.3{295}.

branching by forgetting is an algorithm used for mixture initialization, see Section 6.4.7{139} and Section 8.4.6{279}. It complements quasi-Bayes estimation algorithm.

Branching by forgetting is an algorithm used for mixture initialization, see Section 6.4.7{139} and Section 8.4.6{279}. It complements quasi-Bayes estimation algorithm.

BFRG is an algorithm used for mixture initialization, see Section 6.4.7{139} and Section 8.4.6{279}. It complements quasi-Bayes estimation algorithm.

Recommended pointer is the pointer to the component that is recommended by the advisory system to be active.

recommended pointer is the pointer to the component that is recommended by the advisory system to be active.

Recognizable action is the action available to the user and known to the advisory system as user's actions.

recognizable action is the action available to the user and known to the advisory system as user's actions.

User's ideal pdf is the true user's ideal extended on the data space not available to the user.

user's ideal pdf is the true user's ideal extended on the data space not available to the user.

User's ideal is the true user's ideal extended on the data space not available to the user.

user's ideal is the true user's ideal extended on the data space not available to the user.

True user's ideal pdf is the pdf characterizing desired distribution of data available to the user and reflecting user's aims.

true user's ideal pdf is the pdf characterizing desired distribution of data available to the user and reflecting user's aims.

True user's ideal is the pdf characterizing desired distribution of data available to the user and reflecting user's aims.

true user's ideal is the pdf characterizing desired distribution of data available to the user and reflecting user's aims.

Academic design selects recommended pointers to the components. The design searches for such probabilities of the components that the resulting mixture has the smallest KL divergence from the user's ideal.

academic design selects recommended pointers to the components. The design searches for such probabilities of the components that the resulting mixture has the smallest KL divergence from the user's ideal.

Industrial design selects recommended recognizable actions.

industrial design selects recommended recognizable actions.

Simultaneous design optimizes both recommended pointers to the components and recommended recognizable actions.

simultaneous design optimizes both recommended pointers to the components and recommended recognizable actions.

Probability density function is the basic manipulated object describing uncertain knowledge, decision aims, constraints and random mappings realized by system or decision strategy.

Abbreviation pdf is widely used. The notion is often used even for discrete-valued quantities when the term probability function (pf) would be more appropriate.

probability density function is the basic manipulated object describing uncertain knowledge, decision aims, constraints and random mappings realized by system or decision strategy.

Abbreviation pdf is widely used. The notion is often used even for discrete-valued quantities when the term probability function (pf) would be more appropriate.

pdf abbreviates the term probability density function.

Pdf abbreviates the term Probability density function.

Pf abbreviates the term Probability function.

pf abbreviates the term probability function.

Probability function is the basic manipulated object describing uncertain knowledge, decision aims, constraints and random mappings realized by system or decision strategy when the described quantities are discrete valued, see Probability density function.

probability function is the basic manipulated object describing uncertain knowledge, decision aims, constraints and random mappings realized by system or decision strategy when the described quantities are discrete valued, see probability density function.

Gauss-inverse-Wishart pdf is the pdf describing unknown parameters of normal ARX model $\mathcal{N}_y(\theta'\psi, r)$, i.e. regression coefficients θ, weighting regression vector ψ, and noise variance r.

This pdf is conjugated to the parameterized model and it is determined by the extended information matrix and by the number of degrees of freedom

GiW pdf is Gauss-inverse-Wishart pdf.

Gaussian pdf of a vector x, having expectation μ and covariance R, is denoted $\mathcal{N}_x(\mu, R)$. It is also called normal pdf.

Normal pdf of a vector x, having expectation μ and covariance R, is denoted $\mathcal{N}_x(\mu, R)$. It is also called Gaussian pdf.

normal pdf of a vector x, having expectation μ and covariance R, is denoted $\mathcal{N}_x(\mu, R)$. It is also called Gaussian pdf.

normal or equivalently Gaussian refers to properties related to normal pdf.

Normal or equivalently Gaussian refers to properties related to normal pdf.

Gaussian or equivalently normal refers to properties related to normal pdf.

Extended information matrix forms, together with the number of degrees of freedom, statistics determining GiW pdf serving for estimation of parameters of ARX model (factor).

It is positive definite matrix whose $L'DL$ decomposition is propagated by algorithm equivalent to (weighted) recursive least squares.

extended information matrix forms, together with the number of degrees of freedom, statistics determining GiW pdf serving for estimation of parameters of ARX model (factor).

It is positive definite matrix whose $L'DL$ decomposition is propagated by algorithm equivalent to (weighted) recursive least squares.

Number of degrees of freedom forms, together with the extended information matrix, statistics determining GiW pdf serving for estimation of parameters of ARX model (factor).

It is positive number that can be interpreted as effective number of processed data records.

number of degrees of freedom forms, together with the extended information matrix, statistics determining GiW pdf serving for estimation of parameters of ARX model (factor).

It is positive number that can be interpreted as effective number of processed data records.

Dirichlet pdf is the pdf describing unknown component weight α or transition probabilities $\Theta_{y|\psi}$ of controlled Markov chains conditioned by regression vector ψ.

This pdf is conjugated to the parameterized model and it is determined by the occurrence table and by the number of degrees of freedom

ARX model describes the random mapping $(\psi \to y) \equiv$ (regression vector $\to$ output) by normal pdf $\mathcal{N}_y(\theta'\psi, r)$ where θ are regression coefficients and r noise variance.

Regression coefficients θ weight entries of regression vector ψ: the expected value of the output is $\theta'\psi$.

regression coefficients θ weight entries of regression vector ψ: the expected value of the output is $\theta'\psi$.

Regression vector contains delayed output values (auto-regression) and variables externally supplied (input and entries of other outputs) and their delayed values.

regression vector contains delayed output values (auto-regression) and variables externally supplied (input and entries of other outputs) and their delayed values.

Data vector Ψ is coupling of the predicted output y and regression vector ψ.

data vector Ψ is coupling of the predicted output y and regression vector ψ.

data record contains data measured at particular time moment.

Data record contains data measured at particular time moment.

Occurrence table is sufficient statistics determining Dirichlet pdf. Its positive entries counts the number of observed transitions from discrete values of regression vector to discrete values of output.

occurrence table is sufficient statistics determining Dirichlet pdf. Its positive entries counts the number of observed transitions from discrete values of regression vector to discrete values of output.

Kronecker symbol is defined

$$\delta_{a,b} = \begin{cases} 1 & \text{if } a = b \\ 0 & \text{otherwise} \end{cases}$$

Mixture refers to finite probabilistic mixture that defines the joint pdf $f(d(\mathring{t})|\Theta) \equiv \prod_{t\in t^*} f(d_t|\phi_{t-1},\Theta)$ of all observed data records $d(\mathring{t})$. The pdf $f(d_t|\phi_{t-1},\Theta)$ is the mixture if it has the form

$$f(d_t|\phi_{t-1},\Theta) \equiv \sum_{c\in c^*} \alpha_c f(d_t|\phi_{c;t-1},\Theta_c,c),\ c^* = \{1,\ldots,\mathring{c}\},\ \mathring{c} < \infty,$$

$f(d_t|\phi_{c;t-1},\Theta_c,c)$ is called component given by parameters Θ_c and the state $\phi_{c;t} = \Phi_c(\phi_{c;t-1},d_t),$ i.e. the state $\phi_{c;t-1}$can be recursively updated data d_t,

$\alpha_c \equiv$ the probabilistic component weight

$\Theta \equiv$ mixture parameter formed by component parameters and weights in

$$\Theta^* \equiv \left\{\{\Theta_c \in \Theta_c^*\}_{c\in c^*},\ \alpha \equiv [\alpha_1,\ldots,\alpha_{\mathring{c}}] \in \alpha^* \equiv \left\{\alpha_c \geq 0,\ \sum_{c\in c^*} \alpha_c = 1\right\}\right\}.$$

mixture refers to finite probabilistic mixture that defines the joint pdf $f(d(\mathring{t})|\Theta) \equiv \prod_{t \in t^*} f(d_t|\phi_{t-1}, \Theta)$ of all observed data records $d(\mathring{t})$. The pdf $f(d_t|\phi_{t-1}, \Theta)$ is the mixture if it has the form

$$f(d_t|\phi_{t-1}, \Theta) \equiv \sum_{c \in c^*} \alpha_c f(d_t|\phi_{c;t-1}, \Theta_c, c),\ c^* = \{1, \ldots, \mathring{c}\},\ \mathring{c} < \infty,$$

$f(d_t|\phi_{c;t-1}, \Theta_c, c)$ is called component given by parameters Θ_c and the state $\phi_{c;t} = \Phi_c(\phi_{c;t-1}, d_t)$, i.e. the state $\phi_{c;t-1}$can be recursively updated data d_t,

$\alpha_c \equiv$ the probabilistic component weight

$\Theta \equiv$ mixture parameter formed by component parameters and weights in

$$\Theta^* \equiv \left\{ \{\Theta_c \in \Theta_c^*\}_{c \in c^*},\ \alpha \equiv [\alpha_1, \ldots, \alpha_{\mathring{c}}] \in \alpha^* \equiv \left\{ \alpha_c \geq 0,\ \sum_{c \in c^*} \alpha_c = 1 \right\} \right\}.$$

State (more precisely state in the phase form) contains delayed data records up to the order ord.

state (more precisely state in the phase form) contains delayed data records up to the order ord.

Static mixture is a mixture whose components have constant or empty state ϕ. It describes mutually independent data records $d(\mathring{t})$.

static mixture is a mixture whose components have constant or empty state ϕ. It describes mutually independent data records $d(\mathring{t})$.

Static refers to cases dealing with static mixtures.

static refers to cases dealing with static mixtures.

Dynamic refers to cases dealing with dynamic mixtures.

dynamic refers to cases dealing with dynamic mixtures.

Dynamic mixture is a mixture with at least single component having nontrivial (past data dependent) state. It describes mutually dependent data records $d(\mathring{t})$.

dynamic mixture is a mixture with at least single component having nontrivial (past data dependent) state. It describes mutually dependent data records $d(\mathring{t})$.

Model structure defines uniquely considered functional form of the pdf modelling relationship of innovations to experience and estimated parameters. For instance, for normal ARX model, the structure reduces to the list of regressors in the regression vector. For normal mixture, structures of individual factors (which coincide with ARX models), structures of components (the orders of factors in respective components) and the number of components form the mixture structure.

model structure defines uniquely considered functional form of the pdf modelling relationship of innovations to experience and estimated parameters. For instance, for normal ARX model, the structure reduces to the list of regressors in the regression vector. For normal mixture, structures of individual factors (which coincide with ARX models), structures of components (the orders of factors in respective components) and the number of components form the mixture structure.

Structure defines uniquely considered functional form of the pdf modelling relationship of innovations to experience and estimated parameters. For instance, for normal ARX model, the structure reduces to the list of regressors in the regression vector. For normal mixture, structures of individual factors (which coincide with ARX models), structures of components (the orders of factors in respective components) and the number of components form the mixture structure.

structure defines uniquely considered functional form of the pdf modelling relationship of innovations to experience and estimated parameters. For instance, for normal ARX model, the structure reduces to the list of regressors in the regression vector. For normal mixture, structures of individual factors (which coincide with ARX models), structures of components (the orders of factors in respective components) and the number of components form the mixture structure.

Optimization horizon is determined by the optional design parameter hor or nhor.

optimization horizon is determined by the optional design parameter hor or nhor.

Parameter is a collection of quantities defining system or its model. For mixtures, it consists of component weights and collection of parameters describing factors in respective components. The normal ARX factor is parameterized by regression coefficients and noise variance. The Markov chain factor is parameterized by transition probability.

parameter is a collection of quantities defining system or its model. For mixtures, it consists of component weights and collection of parameters describing factors in respective components. The normal ARX factor is parameterized by regression coefficients and noise variance. The Markov chain factor is parameterized by transition probability.

Component weight is the probability with which the component in a mixture is active, i.e. generates the data record.

component weight is the probability with which the component in a mixture is active, i.e. generates the data record.