

# Package ‘exdqlm’

July 20, 2021

**Title** Extended Dynamic Quantile Linear Models

**Version** 0.1.2

**Author** Raquel Barata [aut, cre],  
Raquel Prado [ths],  
Bruno Sanso [ths]

**Maintainer** Raquel Barata <raquel.a.barata@gmail.com>

**Description** Routines for Bayesian estimation and analysis of dynamic quantile linear models utilizing the extended asymmetric Laplace error distribution, also known as extended dynamic quantile linear models (exDQLM) described in Barata et al (2020) <<https://www.soe.ucsc.edu/research/technical-reports/UCSC-SOE-20-14>>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**Depends** R (>= 2.10)

**Imports** stats, methods, graphics, dlm, coda, tictoc, magic, crch,  
truncnorm, HyperbolicDist, GeneralizedHyperbolic, brms, FNN,  
LaplacesDemon, grDevices

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2021-07-20 08:20:02 UTC

## R topics documented:

BTflow	2
combineMods	2
compPlot	3
dlmMod	4
ELJanoms	5
exdqlmChecks	5
exdqlmForecast	7

exdqlmISVB . . . . .	8
exdqlmMCMC . . . . .	11
exdqlmPlot . . . . .	13
nino34 . . . . .	14
polytrendMod . . . . .	14
scIVTmag . . . . .	15
seasMod . . . . .	16
transfn_exdqlmISVB . . . . .	16

<b>Index</b>	<b>20</b>
--------------	-----------

---

BTflow	<i>Monthly time-series of water flow at Big Tree water gauge.</i>
--------	---

---

### Description

Average monthly natural water flow (cubic feet per second) at the Big Tree gauge of the San Lorenzo river in Santa Cruz County, CA from 1937 through 2014.

### Usage

BTflow

### Format

A time series of length 936.

### Source

<https://waterdata.usgs.gov/nwis/>

### References

U.S. Geological Survey (2016). National Water Information System data available on the World Wide Web (USGS Water Data for the Nation). <https://waterdata.usgs.gov/nwis/>.

---

combineMods	<i>Combines state space blocks of an exDQLM</i>
-------------	---

---

### Description

The function combines two models into a single state space model for an exDQLM.

### Usage

combineMods(m1, m2)

**Arguments**

- m1                    List containing the first model to be combined.  
 m2                    List containing the second model to be combined.

**Value**

List containing the new combined state space model components.

**Examples**

```
trend.comp = polytrendMod(2,rep(0,2),10*diag(2))
seas.comp = seasMod(365,c(1,2,4),C0=10*diag(6))
model = combineMods(trend.comp,seas.comp)
# using dlm package
library(dlm)
model = combineMods(dlmModPoly(order=2,C0=10*diag(2)),dlmModTrig(365,2,C0=10*diag(4)))
```

---

 compPlot

---

*Plot a component of an exDQLM*


---

**Description**

The function plots the dynamic MAP estimates and 95% credible intervals (CrIs) of a specified component of an exDQLM. Alternatively, if just.theta=TRUE the MAP estimates and 95% credible intervals (CrIs) of a single element of the dynamic state vector are plotted.

**Usage**

```
compPlot(
  y,
  m1,
  index,
  add = FALSE,
  col = "purple",
  just.theta = FALSE,
  cr.percent = 0.95
)
```

**Arguments**

- y                    A univariate time-series.  
 m1                    An object of class "exdqlm".  
 index                Index of the component or element of the state vector to be plotted.  
 add                    If TRUE, the dynamic component will be added to existing plot.  
 col                    Color of dynamic component to be plotted. Default is purple.

just.theta	If TRUE, the function plots the dynamic distribution of the index element of the state vector. If just.theta=TRUE, index must have length 1.
cr.percent	Percentage used in the calculation of the credible intervals.

### Value

A list of the following is returned:

- map.comp - MAP estimate of the dynamic component (or element of the state vector).
- lb.comp - Lower bound of the 95% CrIs of the dynamic component (or element of the state vector).
- ub.comp - Upper bound of the 95% CrIs of the dynamic component (or element of the state vector).

### Examples

```

y = scIVTmag[1:365]
trend.comp = polytrendMod(2,rep(0,2),10*diag(2))
seas.comp = seasMod(365,c(1,2,4),C0=10*diag(6))
model = combineMods(trend.comp,seas.comp)
M0 = exdq1mISVB(y,p0=0.85,model,df=c(0.98,1),dim.df = c(2,6),
               gam.init=-3.5,sig.init=15,tol=0.05)
# plot first harmonic component
compPlot(y,M0,index=c(3,4),col="blue")

```

---

d1mMod

*Create state space model of exDQLM from DLM*

---

### Description

The function creates a state space model of an exDQLM from "d1m" object.

### Usage

```
d1mMod(m)
```

### Arguments

**m** An object of class "d1m" representing the DLM version of the desired exDQLM state space model. Only time-invariant d1m objects are currently considered.

### Value

List containing only the components of **m** needed for the exDQLM state space model.

**Examples**

```
library(dlm)
m = dlmModPoly(order=2,C0=10*diag(2)) + dlmModTrig(365,2,C0=10*diag(4))
model = dlmMod(m)
```

---

ELIanomns

*Daily time-series of ELI anomalies.*

---

**Description**

ELI anomalies on the daily time-scale from January 1, 1979 to December 31, 2019 with all February 29ths omitted.

**Usage**

ELIanomns

**Format**

A time series of length 14965.

**Source**

<https://portal.nersc.gov/archive/home/projects/cascade/www/ELI>

**References**

Patricola, C.M., O'Brien, J.P., Risser, M.D. et al. *Maximizing ENSO as a source of western US hydroclimate predictability*. *Clim Dyn* 54, 351–372 (2020). doi: [10.1007/s00382019050048](https://doi.org/10.1007/s00382019050048)

---

exdqlmChecks

*exDQLM Diagnostics*

---

**Description**

The function computes the following for the model(s) provided: the posterior predictive loss criterion based off the check loss, the one-step-ahead distribution sequence and its KL divergence from normality. The function also plots the following: the qq-plot and ACF plot corresponding to the one-step-ahead distribution sequence, and a time series plot of the MAP standard forecast errors.

**Usage**

```

exdqImChecks(
  y,
  m1,
  m2 = NULL,
  plot = TRUE,
  cols = c("grey", "grey"),
  ref = NULL
)

```

**Arguments**

<code>y</code>	A univariate time-series.
<code>m1</code>	An object of class "exdqIm".
<code>m2</code>	An optional additional object of class "exdqIm" to compare with <code>m1</code> .
<code>plot</code>	If TRUE, the following will be plotted for <code>m1</code> and <code>m2</code> (if provided): a qq-plot and ACF plot of the MAP one-step-ahead distribution sequence, and a time series plot of the standardized forecast errors.
<code>cols</code>	Color(s) used to plot diagnostics.
<code>ref</code>	Reference sample of size <code>length(y)</code> from a standard normal distribution used to compute the KL divergence.

**Value**

A list containing the following is returned:

- `m1.uts` - The one-step-ahead distribution sequence of `m1`.
- `m1.KL` - The KL divergence of `m1.uts` and a standard normal.
- `m1.pp1c` - The posterior predictive loss criterion of `m1` based off the check loss function.
- `m1.qq` - The ordered pairs of the qq-plot comparing `m1.uts` with a standard normal distribution.
- `m1.acf` - The autocorrelations of `m1.uts` by lag.

If `m2` is provided, analogous results for `m2` are also included in the list.

**Examples**

```

y = scIVTmag[1:100]
model = polytrendMod(1,mean(y),10)
M0 = exdqImISVB(y,p0=0.85,model,df=c(0.95),dim.df = c(1),
               gam.init=-3.5,sig.init=15)
check.out = exdqImChecks(y,M0,plot=FALSE)
check.out$m1.KL
check.out$m1

```

---

exdqlmForecast	<i>k-step-ahead Forecast</i>
----------------	------------------------------

---

### Description

The function estimates and plots the k-step-ahead forecasted quantile distribution from the filtered quantile estimates.

### Usage

```
exdqlmForecast(
  y,
  start.t,
  k,
  m1,
  fFF = NULL,
  fGG = NULL,
  plot = TRUE,
  add = FALSE,
  cols = c("purple", "magenta"),
  cr.percent = 0.95
)
```

### Arguments

<code>y</code>	A univariate time-series.
<code>start.t</code>	Time index at which to start the forecast.
<code>k</code>	Number of k-steps-ahead to forecast.
<code>m1</code>	An object of class "exdqlm".
<code>fFF</code>	State vector for the forecast steps. <code>fFF</code> must have either 1 (non-time-varying) or <code>k</code> (time-varying) columns. The dimension of <code>fFF</code> must match the estimated <code>exdqlm</code> in <code>m1</code> .
<code>fGG</code>	Evolution matrix for the forecast steps. <code>fGG</code> must be either a matrix (non-time-varying) or an array of depth <code>k</code> (time-varying). The dimensions of <code>fGG</code> must match the estimated <code>exdqlm</code> in <code>m1</code> .
<code>plot</code>	If <code>TRUE</code> the forecasted quantile estimates and 95% credible intervals are plotted, along with the filtered quantile estimates and 95% credible intervals for reference. Default is <code>TRUE</code> .
<code>add</code>	If <code>TRUE</code> , the forecasted quantile will be added to the existing plot. Default is <code>FALSE</code> .
<code>cols</code>	Two colors used to plot filtered and forecasted quantile estimates respectively. Default is <code>c("purple", "magenta")</code> .
<code>cr.percent</code>	Percentage used in the calculation of the credible intervals.

**Value**

A list containing the following is returned:

- fa - The forecasted state mean vectors.
- fR - The forecasted state covariance matrices.
- ff - The forecasted quantile mean estimates.
- fQ - The forecasted quantile variances.

**Examples**

```
y = scIVTmag[1:100]
model = polytrendMod(1,quantile(y,0.85),10)
M0 = exdqlmISVB(y,p0=0.85,model,df=c(0.98),dim.df = c(1),
               gam.init=-3.5,sig.init=15)
exdqlmForecast(y,start.t=90,k=10,M0)
```

---

exdqlmISVB

*exDQLM - ISVB algorithm*

---

**Description**

The function applies an Importance Sampling Variational Bayes (ISVB) algorithm to estimate the posterior of an exDQLM.

**Usage**

```
exdqlmISVB(
  y,
  p0,
  model,
  df,
  dim.df,
  fix.gamma = FALSE,
  gam.init = NA,
  fix.sigma = TRUE,
  sig.init = NA,
  dqlm.ind = FALSE,
  exps0,
  tol = 0.1,
  n.IS = 500,
  n.samp = 200,
  PriorSigma = NULL,
  PriorGamma = NULL,
  verbose = TRUE
)
```



**Arguments**

<code>y</code>	A univariate time-series.
<code>p0</code>	The quantile of interest, a value between 0 and 1.
<code>model</code>	List of the state-space model including GG, FF, prior parameters $m_0$ and $C_0$ .
<code>df</code>	Discount factors for each block.
<code>dim.df</code>	Dimension of each block of discount factors.
<code>fix.gamma</code>	Logical value indicating whether to fix gamma at <code>gam.init</code> . Default is FALSE.
<code>gam.init</code>	Initial value for gamma (skewness parameter), or value at which gamma will be fixed if <code>fix.gamma=TRUE</code> .
<code>fix.sigma</code>	Logical value indicating whether to fix sigma at <code>sig.init</code> . Default is TRUE.
<code>sig.init</code>	Initial value for sigma (scale parameter), or value at which sigma will be fixed if <code>fix.sigma=TRUE</code> .
<code>dqlm.ind</code>	Logical value indicating whether to fix gamma at 0, reducing the exDQLM to the special case of the DQLM. Default is FALSE.
<code>exps0</code>	Initial value for dynamic quantile. If <code>exps0</code> is not specified, it is set to the DLM estimate of the $p_0$ quantile.
<code>tol</code>	Tolerance for convergence of dynamic quantile estimates. Default is <code>tol=0.1</code> .
<code>n.IS</code>	Number of particles for the importance sampling of joint variational distribution of sigma and gamma. Default is <code>n.IS=500</code> .
<code>n.samp</code>	Number of samples to draw from the approximated posterior distribution. Default is <code>n.samp=200</code> .
<code>PriorSigma</code>	List of parameters for inverse gamma prior on sigma; shape <code>a_sig</code> and scale <code>b_sig</code> . Default is an inverse gamma with mean 1 (or <code>sig.init</code> if provided) and variance 10.
<code>PriorGamma</code>	List of parameters for truncated student-t prior on gamma; center <code>m_gam</code> , scale <code>s_gam</code> and degrees of freedom <code>df_gam</code> . Default is a standard student-t with 1 degree of freedom, truncated to the support of gamma.
<code>verbose</code>	Logical value indicating whether progress should be displayed.

**Value**

A list of the following is returned:

- `run.time` - Algorithm run time in seconds.
- `iter` - Number of iterations until convergence was reached.
- `dqlm.ind` - Logical value indicating whether gamma was fixed at 0, reducing the exDQLM to the special case of the DQLM.
- `model` - List of the state-space model including GG, FF, prior parameters  $m_0$  and  $C_0$ .
- `p0` - The quantile which was estimated.
- `df` - Discount factors used for each block.
- `dim.df` - Dimension used for each block of discount factors.

- `sig.init` - Initial value for sigma, or value at which sigma was fixed if `fix.sigma=TRUE`.
- `seq.sigma` - Sequence of sigma estimated by the algorithm until convergence.
- `samp.theta` - Posterior sample of the state vector variational distribution.
- `samp.post.pred` - Sample of the posterior predictive distributions.
- `map.standard.forecast.errors` - MAP standardized one-step-ahead forecast errors.
- `samp.sigma` - Posterior sample of scale parameter sigma variational distribution.
- `samp.vts` - Posterior sample of latent parameters, `v_t`, variational distributions.
- `theta.out` - List containing the variational distribution of the state vector including filtered distribution parameters (`fm` and `fC`) and smoothed distribution parameters (`sm` and `sC`).
- `vts.out` - List containing the variational distributions of latent parameters `v_t`.

If `dqlm.ind=FALSE`, the list also contains:

- `gam.init` - Initial value for gamma, or value at which gamma was fixed if `fix.gamma=TRUE`.
- `seq.gamma` - Sequence of gamma estimated by the algorithm until convergence.
- `samp.gamma` - Posterior sample of skewness parameter gamma variational distribution.
- `samp.sts` - Posterior sample of latent parameters, `s_t`, variational distributions.
- `gammasig.out` - List containing the IS estimate of the variational distribution of sigma and gamma.
- `sts.out` - List containing the variational distributions of latent parameters `s_t`.

Or if `dqlm.ind=TRUE`, the list also contains:

- `sig.out` - List containing the IS estimate of the variational distribution of sigma.

## Examples

```
y = scIVTmag[1:1095]
trend.comp = polytrendMod(1,mean(y),10)
seas.comp = seasMod(365,c(1,2,4),C0=10*diag(6))
model = combineMods(trend.comp,seas.comp)
M0 = exdqImISVB(y,p0=0.85,model,df=c(1,1),dim.df = c(1,6),
               gam.init=-3.5,sig.init=15,tol=0.05)
```

---

 exdqImMCMC

*exDQLM - MCMC algorithm*


---

### Description

The function applies a Markov chain Monte Carlo (MCMC) algorithm to sample the posterior of an exDQLM.

### Usage

```

exdqImMCMC(
  y,
  p0,
  model,
  df,
  dim.df,
  fix.gamma = FALSE,
  gam.init = NA,
  fix.sigma = FALSE,
  sig.init = NA,
  dqIm.ind = FALSE,
  Sig.mh,
  joint.sample = FALSE,
  n.burn = 2000,
  n.mcmc = 1500,
  init.from.isvb = TRUE,
  PriorSigma = NULL,
  PriorGamma = NULL,
  verbose = TRUE
)

```

### Arguments

<code>y</code>	A univariate time-series.
<code>p0</code>	The quantile of interest, a value between 0 and 1.
<code>model</code>	List of the state-space model including GG, FF, prior parameters $m_0$ and $C_0$ .
<code>df</code>	Discount factors for each block.
<code>dim.df</code>	Dimension of each block of discount factors.
<code>fix.gamma</code>	Logical value indicating whether to fix gamma at <code>gam.init</code> . Default is FALSE.
<code>gam.init</code>	Initial value for gamma (skewness parameter), or value at which gamma will be fixed if <code>fix.gamma=TRUE</code> .
<code>fix.sigma</code>	Logical value indicating whether to fix sigma at <code>sig.init</code> . Default is TRUE.
<code>sig.init</code>	Initial value for sigma (scale parameter), or value at which sigma will be fixed if <code>fix.sigma=TRUE</code> .

<code>dqlm.ind</code>	Logical value indicating whether to fix gamma at $\emptyset$ , reducing the exDQLM to the special case of the DQLM. Default is FALSE.
<code>Sig.mh</code>	Covariance matrix used in the random walk MH step to jointly sample sigma and gamma.
<code>joint.sample</code>	Logical value indicating whether or not to recompute <code>Sig.mh</code> based off the initial burn-in samples of gamma and sigma. Default is FALSE.
<code>n.burn</code>	Number of MCMC iterations to burn. Default is <code>n.burn = 2000</code> .
<code>n.mcmc</code>	Number of MCMC iterations to sample. Default is <code>n.mcmc = 1500</code> .
<code>init.from.isvb</code>	Logical value indicating whether or not to initialize the MCMC using the ISVB algorithm. Default is TRUE.
<code>PriorSigma</code>	List of parameters for inverse gamma prior on sigma; shape <code>a_sig</code> and scale <code>b_sig</code> . Default is an inverse gamma with mean 1 (or <code>sig.init</code> if provided) and variance 10.
<code>PriorGamma</code>	List of parameters for truncated student-t prior on gamma; center <code>m_gam</code> , scale <code>s_gam</code> and degrees of freedom <code>df_gam</code> . Default is a standard student-t with 1 degree of freedom, truncated to the support of gamma.
<code>verbose</code>	Logical value indicating whether progress should be displayed.

### Value

A list of the following is returned:

- `run.time` - Algorithm run time in seconds.
- `model` - List of the state-space model including GG, FF, prior parameters  $m_0$  and  $C_0$ .
- `p0` - The quantile which was estimated.
- `df` - Discount factors used for each block.
- `dim.df` - Dimension used for each block of discount factors.
- `samp.theta` - Posterior sample of the state vector.
- `samp.post.pred` - Sample of the posterior predictive distributions.
- `map.standard.forecast.errors` - MAP standardized one-step-ahead forecast errors.
- `samp.sigma` - Posterior sample of scale parameter sigma.
- `samp.vts` - Posterior sample of latent parameters,  $v_t$ .
- `theta.out` - List containing the distributions of the state vector including filtered distribution parameters (`fm` and `fC`) and smoothed distribution parameters (`sm` and `sC`).

If `dqlm.ind=FALSE`, the list also contains the following:

- `samp.gamma` - Posterior sample of skewness parameter gamma.
- `samp.sts` - Posterior sample of latent parameters,  $s_t$ .
- `init.log.sigma` - Burned samples of log sigma from the random walk MH joint sampling of sigma and gamma.
- `init.logit.gamma` - Burned samples of logit gamma from the random walk MH joint sampling of sigma and gamma.
- `accept.rate` - Acceptance rate of the MH step.
- `Sig.mh` - Covariance matrix used in MH step to jointly sample sigma and gamma.

**Examples**

```

y = scIVTmag[1:100]
trend.comp = polytrendMod(1,mean(y),10)
seas.comp = seasMod(365,c(1,2,4),C0=10*diag(6))
model = combineMods(trend.comp,seas.comp)
M2 = exdqImMCMC(y,p0=0.85,model,df=c(1,1),dim.df = c(1,6),
               gam.init=-3.5,sig.init=15,
               n.burn=100,n.mcmc=150)

```

---

exdqImPlot

*Plot exDQLM*


---

**Description**

The function plots the MAP estimates and 95% credible intervals (CrIs) of the dynamic quantile of an exDQLM.

**Usage**

```
exdqImPlot(y, m1, add = FALSE, col = "purple", cr.percent = 0.95)
```

**Arguments**

y	A univariate time-series.
m1	An object of class "exdqIm".
add	If TRUE, the dynamic quantile will be added to existing plot.
col	Color of dynamic quantile to be plotted. Default is purple.
cr.percent	Percentage used in the calculation of the credible intervals.

**Value**

A list of the following is returned:

- map.quant - MAP estimate of the dynamic quantile.
- lb.quant - Lower bound of the 95% CrIs of the dynamic quantile.
- ub.quant - Upper bound of the 95% CrIs of the dynamic quantile.

**Examples**

```

y = scIVTmag[1:100]
model = polytrendMod(1,quantile(y,0.85),10)
M0 = exdqImISVB(y,p0=0.85,model,df=c(0.98),dim.df = c(1),
               gam.init=-3.5,sig.init=15)
exdqImPlot(y,M0,col="blue")

```

---

nino34	<i>Monthly Niño 3.4 Index.</i>
--------	--------------------------------

---

**Description**

Monthly Niño 3.4 sea surface temperature (SST) Index time series.

**Usage**

```
nino34
```

**Format**

A time series of length 936.

**Source**

[https://psl.noaa.gov/gcos\\_wgsp/Timeseries/Nino34/](https://psl.noaa.gov/gcos_wgsp/Timeseries/Nino34/)

**References**

Rayner N. A., D. E. Parker, E. B. Horton, C. K. Folland, L. V. Alexander, D. P. Rowell, E. C. Kent, A. Kaplan. *Global analyses of sea surface temperature, sea ice, and night marine air temperature since the late nineteenth century*. J. Geophys. Res., 108 (D14), 4407 (2003). doi: [10.1029/2002JD002670](https://doi.org/10.1029/2002JD002670)

---

polytrendMod	<i>Create an n-th order polynomial exDQLM component</i>
--------------	---

---

**Description**

The function creates an n-th order polynomial exDQLM component.

**Usage**

```
polytrendMod(order, m0, C0)
```

**Arguments**

order	The order of the polynomial model.
m0	Prior mean of the state vector. Default is $m0 = \text{rep}(0, \text{order})$ .
C0	Prior covariance of the state vector. Default is $C0 = 1e3 * \text{diag}(\text{order})$ .

**Value**

A list of the following:

- FF - Observational vector.
- GG - Evolution matrix.
- m0 - Prior mean of the state vector.
- C0 - Prior covariance of the state vector.

**Examples**

```
# create a second order polynomial component  
trend.comp = polytrendMod(2, rep(0, 2), 10 * diag(2))
```

---

scIVTmag

*Time series of daily average magnitude IVT in Santa Cruz, CA.*

---

**Description**

ECMWF Re-Analysis 5 (ERA5) daily average magnitude IVT in Santa Cruz, CA (approximately 22 N, 122 W) from January 1, 1979 to December 31, 2019 with all February 29ths omitted.

**Usage**

```
scIVTmag
```

**Format**

A time series of length 14965.

**Source**

<https://cds.climate.copernicus.eu>

**References**

Hersbach, H, Bell, B, Berrisford, P, et al. *The ERA5 global reanalysis*. Q J R Meteorol Soc. 2020; 146: 1999– 2049. doi: [10.1002/qj.3803](https://doi.org/10.1002/qj.3803)

---

seasMod                      *Create Fourier representation of a periodic exDQLM component*

---

### Description

The function creates a Fourier form periodic component for given period and harmonics.

### Usage

```
seasMod(p, h, m0, C0)
```

### Arguments

p	The period.
h	Vector of harmonics to be included.
m0	Prior mean of the state vector.
C0	Prior covariance of the state vector.

### Value

A list of the following:

- FF - Observational vector.
- GG - Evolution matrix.
- m0 - Prior mean of the state vector.
- C0 - Prior covariance of the state vector.

### Examples

```
# create a seasonal component with first, second and fourth harmonics of a period of 365
seas.comp = seasMod(365,c(1,2,4),C0=10*diag(6))
```

---

transfn\_exdqImISVB            *Transfer Function exDQLM - ISVB algorithm*

---

### Description

The function applies an Importance Sampling Variational Bayes (ISVB) algorithm to estimate the posterior of an exDQLM with exponential decay transfer function component.



**Usage**

```

transfn_exdqlmISVB(
  y,
  p0,
  model,
  X,
  df,
  dim.df,
  lam,
  tf.df,
  fix.gamma = FALSE,
  gam.init = NA,
  fix.sigma = TRUE,
  sig.init = NA,
  dqlm.ind = FALSE,
  exps0,
  tol = 0.1,
  n.IS = 500,
  n.samp = 200,
  PriorSigma = NULL,
  PriorGamma = NULL,
  tf.m0 = rep(0, 2),
  tf.C0 = diag(1, 2),
  verbose = TRUE
)

```

**Arguments**

y	A univariate time-series.
p0	The quantile of interest, a value between 0 and 1.
model	List of the state-space model including GG, FF, prior parameters m0 and C0.
X	A univariate time-series which will be the input of the transfer function component.
df	Discount factors for each block.
dim.df	Dimension of each block of discount factors.
lam	Transfer function rate parameter lambda, a value between 0 and 1.
tf.df	Discount factor(s) used for the transfer function component.
fix.gamma	Logical value indicating whether to fix gamma at gam.init. Default is FALSE.
gam.init	Initial value for gamma (skewness parameter), or value at which gamma will be fixed if fix.gamma=TRUE.
fix.sigma	Logical value indicating whether to fix sigma at sig.init. Default is TRUE.
sig.init	Initial value for sigma (scale parameter), or value at which sigma will be fixed if fix.sigma=TRUE.
dqlm.ind	Logical value indicating whether to fix gamma at 0, reducing the exDQLM to the special case of the DQLM. Default is FALSE.

<code>exps0</code>	Initial value for dynamic quantile. If <code>exps0</code> is not specified, it is set to the DLM estimate of the $p_0$ quantile.
<code>tol</code>	Tolerance for convergence of dynamic quantile estimates. Default is <code>tol=0.1</code> .
<code>n.IS</code>	Number of particles for the importance sampling of joint variational distribution of sigma and gamma. Default is <code>n.IS=500</code> .
<code>n.samp</code>	Number of samples to draw from the approximated posterior distribution. Default is <code>n.samp=200</code> .
<code>PriorSigma</code>	List of parameters for inverse gamma prior on sigma; shape <code>a_sig</code> and scale <code>b_sig</code> . Default is an inverse gamma with mean 1 (or <code>sig.init</code> if provided) and variance 10.
<code>PriorGamma</code>	List of parameters for truncated student-t prior on gamma; center <code>m_gam</code> , scale <code>s_gam</code> and degrees of freedom <code>df_gam</code> . Default is a standard student-t with 1 degree of freedom, truncated to the support of gamma.
<code>tf.m0</code>	Prior mean of the transfer function component.
<code>tf.C0</code>	Prior covariance of the transfer function component.
<code>verbose</code>	Logical value indicating whether progress should be displayed.

### Value

A list of the following is returned:

- `run.time` - Algorithm run time in seconds.
- `iter` - Number of iterations until convergence was reached.
- `dqlm.ind` - Logical value indicating whether gamma was fixed at  $\theta$ , reducing the exDQLM to the special case of the DQLM.
- `model` - List of the augmented state-space model including GG, FF, prior parameters `m0` and `C0`.
- `p0` - The quantile which was estimated.
- `df` - Discount factors used for each block, including transfer function component.
- `dim.df` - Dimension used for each block of discount factors, including transfer function component.
- `lam` - Transfer function rate parameter lambda.
- `sig.init` - Initial value for sigma, or value at which sigma was fixed if `fix.sigma=TRUE`.
- `seq.sigma` - Sequence of sigma estimated by the algorithm until convergence.
- `samp.theta` - Posterior sample of the state vector variational distribution.
- `samp.post.pred` - Sample of the posterior predictive distributions.
- `map.standard.forecast.errors` - MAP standardized one-step-ahead forecast errors.
- `samp.sigma` - Posterior sample of scale parameter sigma variational distribution.
- `samp.vts` - Posterior sample of latent parameters, `v_t`, variational distributions.
- `theta.out` - List containing the variational distribution of the state vector including filtered distribution parameters (`fm` and `fC`) and smoothed distribution parameters (`sm` and `sC`).
- `vts.out` - List containing the variational distributions of latent parameters `v_t`.

- `median.kt` - Median number of time steps until the effect of  $X_t$  is less than or equal to  $1e-3$ .

If `dqlm.ind=FALSE`, the list also contains:

- `gam.init` - Initial value for gamma, or value at which gamma was fixed if `fix.gamma=TRUE`.
- `seq.gamma` - Sequence of gamma estimated by the algorithm until convergence.
- `samp.gamma` - Posterior sample of skewness parameter gamma variational distribution.
- `samp.sts` - Posterior sample of latent parameters,  $s_t$ , variational distributions.
- `gammasig.out` - List containing the IS estimate of the variational distribution of sigma and gamma.
- `sts.out` - List containing the variational distributions of latent parameters  $s_t$ .

Or if `dqlm.ind=TRUE`, the list also contains:

- `sig.out` - List containing the IS estimate of the variational distribution of sigma.

## Examples

```
y = scIVTmag[1:1095]
X = ELIanom[1:1095]
trend.comp = polytrendMod(1,mean(y),10)
seas.comp = seasMod(365,c(1,2,4),C0=10*diag(6))
model = combineMods(trend.comp,seas.comp)
M1 = transfn_exdqlmISVB(y,p0=0.85,model=model,
                        X,df=c(1,1),dim.df = c(1,6),
                        gam.init=-3.5,sig.init=15,
                        lam=0.38,tf.df=c(0.97,0.97))
```

# Index

## \* datasets

- BTflow, [2](#)
- ELIanom, [5](#)
- nino34, [14](#)
- scIVTmag, [15](#)

BTflow, [2](#)

combineMods, [2](#)  
compPlot, [3](#)

dImMod, [4](#)

ELIanom, [5](#)  
exdqlmChecks, [5](#)  
exdqlmForecast, [7](#)  
exdqlmISVB, [8](#)  
exdqlmMCMC, [11](#)  
exdqlmPlot, [13](#)

nino34, [14](#)

polytrendMod, [14](#)

scIVTmag, [15](#)  
seasMod, [16](#)

transfn\_exdqlmISVB, [16](#)