# Package 'manifestos'

February 5, 2013

**Type** Package

**Title** Spatial Modelling of Party Manifestos

**Version** 0.7-1

**Date** 2013-02-05

**Author** Martin Elff

**Maintainer** Martin Elff <martin.elff@uni-konstanz.de>

**Description** This package provides an MCEM algorithm to
estimate the parameters of a spatial model for the
reconstruction of policy positions of political actors,such as parties, based on manifesto data or, more generally,based on multinomial counts.

**License** GPL-2

**LazyLoad** Yes

**Depends** Matrix, mvtnorm, munfold, memisc, Rcpp (>= 0.9.4),RcppArmadillo (>= 0.2.0)

**LinkingTo** Rcpp, RcppArmadillo

## R topics documented:

latpos *Dynamic State-Space Models of Coded Political Texts*

## Description

The function latpos allows to specify and estimate a dynamic state-space model of political texts and returns an object from which the (latent) positions, which actors take in their texts, can be predicted.

## Usage

```
  latpos(formula,data,subset,id,time,
         unfold.method="Schoenemann",start=NULL,
         sampler=mvt.sampler(df=7*length(latent.dims)),
         ...)

  ## S3 method for class 'latpos'
predict(object, newdata = NULL, id=NULL, time=NULL,
          type=c("posterior modes","posterior means","simulate"),
          se.fit=FALSE, interval=c("none","normal","percentile"), level=0.95,
          sample.size = object$sample.size,
          sampler=object$sampler,
          maxiter=100,...)
```

## Arguments

| | |
|---|---|
| formula | a formula describing the model. |
| data | an optional data frame that contains data to which the model is fitted. |
| subset | a logical vector; an optional condition that defines a subset of the optiona data frame to which the analysis is to be restricted. |
| id | name of a variable that identifies the actors. |
| time | name of a variable that identifies the occasions on which actors publish their political texts. |
| unfold.method | a text string, name of the method which is used in the unfolding procedure to obtain starting values. |
| start | an optional list giving starting values. |
| sampler | an object that generates random numbers to be used in the MCEM algorithm. |
| ... | further optional arguments, that are passed to latpos.control and latpos.start by the function latpos and ignored by the predict method. |
| object | an object of class "latpos". |
| newdata | an optional data frame to predict actors positions from, based on the fitted model contained in object. |
| type | a character string identifying the type of predictions about the latent positions, either "posterior modes","posterior means", or "simulate". If the type is "posterior means" or "posterior modes", only one prediction is generated per coded text. If the type is "simulate", then sample.size predictions are generated from the empirical Bayes posterior distribution of the positions. |

| | |
|---|---|
| se.fit | logical, should standard errors be provided with the posterior means? |
| interval | a character vector. If "none", then no prediction intervals are returned, if "normal" than prediction intervals are contstructed based on normal quantiles, if "percentile", then predicion intervals are constructed based on the empirical percentiles of samples from the posterior. |
| level | a number between zero and one, the nominal coverage level of the prediction intervals. |
| sample.size | a positive number, the sample size for the simulated posterior means, or the number of imputed values generated from the posterior. |
| maxiter | a positive number, the maximum number of iterations used to compute posterior modes. |

## Value

latpos returns an object of class "latpos" that contains parameter estimates etc.

predict.latpos returns a vector or a matrix depending on the arguments provided to the function.

## Examples

```
## Not run:

load(file="manifesto-counts.RData")
options(latpos.chunk.size=4*1024*1024) # determines the size of
# data chunks used in internal computations.

econ.latpos <- latpos(
   nationalize+controlecon+econplan+marketregul+incentives+econorthodox+freeenterp~econlr,
   data=manifesto.counts,
   id=party,
   time=year*100 + month,
   free.beta=TRUE,
   initial.size=50
 )

summary(econ.latpos)


nonmat.latpos <- latpos(
    laworder+tradmoralpos+tradmoralneg+natwaylifepos+
    democracy+freedmhumrights~authlib+tradmod,
    data=manifesto.counts,
    id=party,
    time=year*100 + month,
    initial.size=50,
    restrictions=set.parms.free(
      authlib=c("natwaylifepos","laworder","democracy","freedmhumrights"),
      tradmod=c("tradmoralpos","tradmoralneg","democracy","freedmhumrights")
    ),
    start = list(
          A=list(
              authlib=c(natwaylifepos=1.4,laworder=.7,democracy=-1),
              tradmod=c(tradmoralpos=1,tradmoralneg=-1)
            )
        )
```

```
)
summary(nonmat.latpos)

## End(Not run)
```

---

`latpos_restrictions` *Specify and Implement Restrictions on the Location Parameters*

---

## Description

`restrictor` creates a linear transformation that maps a small set of linearly unrestriced parameters to a larger set of linearly restricted parameters.

`set.parms.free` specifies a call to `restrictor` in which only the specified parameters are allowed to differ from zero.

## Usage

```
restrictor(C,d=numeric(m),sign=7)

set.parms.free(...)
```

## Arguments

| | |
|---|---|
| `C` | The matrix involved in the linear restriction $C\alpha = d$. |
| `d` | The vector involved in the linear restriction $C\alpha = d$. |
| `sign` | The number of significant digits to use for rounding to compensate finite machine precision in computing the QR decomposition. |
| `...` | several character vector arguments. Each character vector corresponds to one of the axes of the latent space, and each character string in a vector corresponds to the name of a policy objective that can obtain coordinate values different from zero. |

## Value

The values of these functions are for internal use only.

If $C\alpha = d$ then $\alpha = Q\phi + r$. The function `restrictor` returns a list with the elements "reduction" (which equals $Q$) and "offset" (which equals $r$)

The function `set.parms.free` returns a function that generates arguments $C$ and $d$ with which the `restrictor` is called inside of the function `latpos`.

latpos_tunefit *Starting Values and Fine Tuning of the MCEM algorithm*

## Description

Function `latpos.start` is used to construct "good" starting values, while function `latpos.control` provides settings for the numerical aspects of the MCEM algorithm, with reasonable defaults.

## Usage

```
latpos.start(resp,latent.dims,manifest,start,
             unfold.method,restrictions=standard.restrictions,
             maxiter,...)
latpos.control(maxiter=200,initial.size=101,
                           Lambda.alpha=.05,
                           Lambda.eps=1e-7,
                           diff.logLik.eps=1e-7,
                           abs.diff.psi.eps=0,
                           rel.diff.psi.eps=0,
                           max.size=Inf,
                           min.final.size=1000,
                           force.increase=TRUE,
                           Q.linesearch=TRUE,
                           ...)
```

## Arguments

| | |
|---|---|
| `resp` | an internal representation of the observed data. |
| `latent.dims` | a character vector with the names of the axes of the latent space. |
| `manifest` | a character vector with the names of the observed variables, i.e. emphasis counts of policy objectives. |
| `start` | an optional list with starting values for the model parameters |
| `unfold.method` | the unfolding method to be used to generate reasonable starting values. |
| `restrictions` | an object representing restrictions on the model parameters, see `restrictor`. |
| `maxiter` | the maximum number of iterations to use, in `latpos.start` to get initial values for the posterior modes, in `latpos.control` to set the maximum number of MCEM iterations. |
| `initial.size` | a positive number, the simulation sample size to use in the first MCEM iteration. |
| `Lambda.alpha` | a "significance level" for the increase of the Q-function. If the increase is not "statistically significant" at this level, the sample size is automatically increased. |
| `Lambda.eps` | a non-negative number as convergence critierion. If the increase of the Q-function is smaller than this value, convergence of the MCEM is declared. |
| `diff.logLik.eps` | a non-negative number as convergence critierion. If the increase of the marginal log-likelihood is smaller than this value, convergence of the MCEM is declared. |
| `abs.diff.psi.eps` | a non-negative number as an alternative convergence critierion. if the absolute change of the model parameters is smaller than this value, convergence of the MCEM is declared. |

`rel.diff.psi.eps`
: a non-negative number as an alternative convergence critierion. if the absolute change of the model parameters is smaller than this value, convergence of the MCEM is declared.

`max.size`
: a positive number, the maximum simulation sample size to be used.

`min.final.size`
: a positive number, the minimal simulation sample size to be used in the final iterations of the MCEM algorithm.

`force.increase`
: logical; if TRUE and the likelihood or the Q-function cannot be increased then conduct a line search for the optimal step size.

`Q.linesearch`
: logical; if TRUE, force.increase==TRUE and the likelihood or the Q-function cannot be increased then conduct a line search for the optimal step size; if FALSE, but force.increase==TRUE and the likelihood or the Q-function cannot be increased then step back to the values of the previous iteration.

`...`
: other arguments, ignored.

---

`sampler` *Control the Generation of Random Numbers for Sampling from the Proposal Distribution*

---

## Description

The functions `mvt.sampler` and `mvnorm.sampler` can be used to fine-tune the construction and sampling from the proposal distribution in the importance sampling algorithm used in the MCEM algorithm to fit `latpos` models.

## Usage

```
mvt.sampler(df,fix.seed=TRUE) # Multivariate Student's t-distribution
mvnorm.sampler(fix.seed=TRUE) # Multivariate normal distribution
```

## Arguments

`df`
: a number, the degrees of freedom of the multivariate Student's t-distribution.

`fix.seed`
: a logical value; if TRUE the seed of the random number generated is reset in each MCEM iteration to make sure that approximate marginal log-likelihood surface is continuous while the simulation sample size is constant.

## Value

Both functions return a list that contains a function to generate random numbers and to compute the log-density function of the distribution from which random numbers are generated.

# Index