Package 'maptpx'

October 24, 2017

Title MAP Estimation of Topic Models

Version 1.9-3

Author Matt Taddy <taddy@chicagobooth.edu>

Depends R (>= 2.10)

Imports slam, SQUAREM, boot

Suggests MASS

Description Posterior maximization for topic models (LDA) in text analysis,
as described in Taddy (2012) `on estimation and selection for topic models'.
Previous versions of this code were included as part of the textir package.
If you want to take advantage of openmp parallelization, uncomment the relevant
flags in src/MAKEVARS before compiling.

Maintainer Matt Taddy <taddy@chicagobooth.edu>

License GPL-3

URL http://faculty.chicagobooth.edu/matt.taddy/index.html

R topics documented:

`counts` *Utilities for count matrices*

Description

Tools for manipulating (sparse) count matrices.

Usage

```
normalizetpx(x,byrow=TRUE)
stm_tfidf(x)
```

Arguments

x	A simple_triplet_matrix or matrix of counts.
byrow	Whether to normalizetpx by row or column totals.

Value

normalizetpx divides the counts by row or column totals, and stm_tfidf returns a matrix with entries $x_{ij} \log[n/(d_j + 1)]$, where x_{ij} is term-j frequency in document-i, and d_j is the number of documents containing term-j.

Author(s)

Matt Taddy <taddy@chicagobooth.edu>

Examples

```
normalizetpx( matrix(1:9, ncol=3) )
normalizetpx( matrix(1:9, ncol=3), byrow=FALSE )

(x <- matrix(rbinom(15,size=2,prob=.25),ncol=3))
stm_tfidf(x)
```

---

predict.topics *topic predict*

Description

Predict function for Topic Models

Usage

```
## S3 method for class 'topics'
predict( object, newcounts, loglhd=FALSE, ... )
```

Arguments

object	An output object from the topics function, or the corresponding matrix of estimated topics.
newcounts	An nrow(object$theta)-column matrix of multinomial phrase/category counts for new documents/observations. Can be either a simple matrix or a simple_triplet_matrix.
loglhd	Whether or not to calculate and return sum(x*log(p)), the un-normalized log likelihood.
...	Additional arguments to the undocumented internal tpx* functions.

Details

Under the default mixed-membership topic model, this function uses sequential quadratic programming to fit topic weights Ω for new documents. Estimates for each new ω_i are, conditional on object$theta, MAP in the (K-1)-dimensional logit transformed parameter space.

Value

The output is an `nrow(newcounts)` by `object$K` matrix of document topic weights, or a list with including these weights as `W` and the log likelihood as `L`.

Author(s)

Matt Taddy <taddy@chicagobooth.edu>

References

Taddy (2012), *On Estimation and Selection for Topic Models*. http://arxiv.org/abs/1109.4518

See Also

topics, plot.topics, summary.topics, congress109

Examples

```
## Simulate some data
omega <- t(rdir(500, rep(1/10,10)))
theta <- rdir(10, rep(1/1000,1000))
Q <- omega%*%t(theta)
counts <- matrix(ncol=1000, nrow=500)
totals <- rpois(500, 200)
for(i in 1:500){ counts[i,] <- rmultinom(1, size=totals[i], prob=Q[i,]) }

## predict omega given theta
W <- predict.topics( theta, counts )
plot(W, omega, pch=21, bg=8)
```

---

rdir *Dirichlet RNG*

Description

Generate random draws from a Dirichlet distribution

Usage

```
rdir(n, alpha)
```

Arguments

`n`	The number of observations.
`alpha`	A `vector` of scale parameters, such that $E[p_j] = \alpha_j / \sum_i \alpha_i$.

Value

An `n` column matrix containing the observations.

Author(s)

Matt Taddy <taddy@chicagobooth.edu>

Examples

```
rdir(3,rep(1,6))
```

`topics` *Estimation for Topic Models*

Description

MAP estimation of Topic models

Usage

```
topics(counts, K, shape=NULL, initopics=NULL,
  tol=0.1, bf=FALSE, kill=2, ord=TRUE, verb=1,
  admix=TRUE, nbundles=1, use_squarem=FALSE,
  init.adapt=FALSE,  type = "full",
  ind_model_indices = NULL,
  signatures = NULL,
  light=1, method_admix=1,
  sample_init = TRUE, tmax = 10000, ...)
```

Arguments

`counts`	A matrix of multinomial response counts in `ncol(counts)` phrases/categories for `nrow(counts)` documents/observations. Can be either a simple `matrix` or a `simple_triplet_matrix`.
`K`	The number of latent topics. If `length(K)>1`, `topics` will find the Bayes factor (vs a null single topic model) for each element and return parameter estimates for the highest probability K.
`shape`	Optional argument to specify the Dirichlet prior concentration parameter as `shape` for topic-phrase probabilities. Defaults to `1/(K*ncol(counts))`. For fixed single `K`, this can also be a `ncol(counts)` by `K` matrix of unique shapes for each topic element.
`initopics`	Optional start-location for $[\theta_1 ... \theta_K]$, the topic-phrase probabilities. Dimensions must accord with the smallest element of `K`. If `NULL`, the initial estimates are built by incrementally adding topics.
`tol`	Convergence tolerance: optimization stops, conditional on some extra checks, when the *relative* posterior increase over a full paramater set update is less than `tol`.
`bf`	An indicator for whether or not to calculate the Bayes factor for univariate `K`. If `length(K)>1`, this is ignored and Bayes factors are always calculated.
`kill`	For choosing from multiple `K` numbers of topics (evaluated in increasing order), the search will stop after `kill` consecutive drops in the corresponding Bayes factor. Specify `kill=0` if you want Bayes factors for all elements of `K`.

ord	If TRUE, the returned topics (columns of theta) will be ordered by decreasing usage (i.e., by decreasing colSums(omega)).
verb	A switch for controlling printed output. verb > 0 will print something, with the level of detail increasing with verb.
admix	if TRUE, admixture model used, else a mixture model is used. Defaults to TRUE
nbundles	apply the active set method update to the topic proportions omega after these many steps if light=1. Defaults to 1, which essentially means the update is used at each step.
use_squarem	if TRUE, uses the SQUAREM implementation due to Ravi Varadhan. Else uses the usual EM + Quasi Newton Acceleration update.
init.adapt	if TRUE, adpatively initializes the topic distributions theta. Deafults to FALSE.
type	whether to use the "full" version of the topic model or the "independent" version proposed by Shiraishi and Stephens (2015) when the variables comprise of blocks of features.
ind_model_indices	The indices representing the blocks of features in the variables - to be needed for the "independent" version of the model. The "full" version by default sets the them to NULL.
signatures	A matrix of factors with each column representing a factor for each block with the levels reported along the rows for each row.
light	May tale values 0,1,2. If light=2, we select a set of genes after a burn-in period for each cluster and iterate over these selected genes. If light=1, the active set method update occurs after every nbundles steps. If light=0, no active set method is used for updating the topic proportions omega. Defaults to 1.
method_admix	May take values 1 or 2. If method_admix=1, R code is used for EM update. If method_admix=2, C code is used for EM update. Usually C code is preferredfor small datasets, but R code is performs more stablly for large data and is also faster. Defaults to 1.
sample_init	If TRUE, we use a fixed initialization scheme due to Taddy 2014, otherwise we switch to random initialization. Defaults to TRUE.
tmax	Number of iterations used for the topic model. Defaults to 10,000 iterations.
...	Additional arguments to the undocumented internal tpx* functions.

Details

A latent topic model represents each i'th document's term-count vector X_i (with $\sum_j x_{ij} = m_i$ total phrase count) as having been drawn from a mixture of K multinomials, each parameterized by topic-phrase probabilities θ_i, such that

$$X_i \sim MN(m_i, \omega_1\theta_1 + ... + \omega_K\theta_K).$$

We assign a K-dimensional Dirichlet(1/K) prior to each document's topic weights $[\omega_{i1}...\omega_{iK}]$, and the prior on each θ_k is Dirichlet with concentration α. The topics function uses quasi-newton accelerated EM, augmented with sequential quadratic programming for conditional $\Omega|\Theta$ updates, to obtain MAP estimates for the topic model parameters. We also provide Bayes factor estimation, from marginal likelihood calculations based on a Laplace approximation around the converged MAP parameter estimates. If input length(K)>1, these Bayes factors are used for model selection. Full details are in Taddy (2011).

Value

An `topics` object list with entries

`K`	The number of latent topics estimated. If input `length(K)>1`, on output this is a single value corresponding to the model with the highest Bayes factor.
`theta`	The `ncol{counts}` by `K` matrix of estimated topic-phrase probabilities.
`omega`	The `nrow{counts}` by `K` matrix of estimated document-topic weights.
`BF`	The log Bayes factor for each number of topics in the input `K`, against a null single topic model.
`D`	Residual dispersion: for each element of `K`, estimated dispersion parameter (which should be near one for the multinomial), degrees of freedom, and p-value for a test of whether the true dispersion is > 1.
`X`	The input count matrix, in `dgTMatrix` format.

Note

Estimates are actually functions of the MAP (K-1 or p-1)-dimensional logit transformed natural exponential family parameters.

Author(s)

Matt Taddy <taddy@chicagobooth.edu>

References

Taddy (2012), *On Estimation and Selection for Topic Models*. http://arxiv.org/abs/1109.4518

See Also

plot.topics, summary.topics, predict.topics, wsjibm, congress109, we8there

Examples

```
## see also http://faculty.chicagobooth.edu/matt.taddy/teaching/we8there.R

## Simulation Parameters
K <- 10
n <- 100
p <- 100
omega <- t(rdir(n, rep(1/K,K)))
theta <- rdir(K, rep(1/p,p))

## Simulated counts
Q <- omega%*%t(theta)
counts <- matrix(ncol=p, nrow=n)
totals <- rpois(n, 100)
for(i in 1:n){ counts[i,] <- rmultinom(1, size=totals[i], prob=Q[i,]) }

## Bayes Factor model selection (should choose K or nearby)
summary(simselect <- topics(counts, K=K+c(-5:5)), nwrd=0)

## MAP fit for given K
summary( simfit <- topics(counts,  K=K, verb=2), n=0 )
```

```
## Adjust for label switching and plot the fit (color by topic)
toplab <- rep(0,K)
for(k in 1:K){ toplab[k] <- which.min(colSums(abs(simfit$theta-theta[,k]))) }
par(mfrow=c(1,2))
tpxcols <- matrix(rainbow(K), ncol=ncol(theta), byrow=TRUE)
plot(theta,simfit$theta[,toplab], ylab="fitted values", pch=21, bg=tpxcols)
plot(omega,simfit$omega[,toplab], ylab="fitted values", pch=21, bg=tpxcols)
title("True vs Fitted Values (color by topic)", outer=TRUE, line=-2)

## The S3 method plot functions
par(mfrow=c(1,2))
plot(simfit, lgd.K=2)
plot(simfit, type="resid")
```

`topicVar` *topic variance*

Description

Tools for looking at the variance of document-topic weights.

Usage

```
topicVar(counts, theta, omega)
logit(prob)
expit(eta)
```

Arguments

`counts`	A matrix of multinomial response counts, as inputed to the `topics` or `predict.topics` functions.
`theta`	A fitted topic matrix, as ouput from the `topics` or `predict.topics` functions.
`omega`	A fitted document topic-weight matrix, as ouput from the `topics` or `predict.topics` functions.
`prob`	A probability vector (positive and sums to one) or a matrix with probability vector rows.
`eta`	A vector of the natural exponential family parameterization for a probability vector (with first category taken as null) or a matrix with each row the NEF parameters for a single observation.

Details

These function use the natural exponential family (NEF) parametrization of a probability vector $q_0 ... q_{K-1}$ with the first element corresponding to a 'null' category; that is, with $NEF(q) = e_1 ... e_{K-1}$ and setting $e_0 = 0$, the probabilities are

$$q_k = \frac{exp[e_k]}{1 + \sum exp[e_j]}.$$

Refer to Taddy (2012) for details.

Value

topicVar returns an array with dimensions $(K-1, K-1, n)$, where K=ncol(omega)=ncol(theta) and n = nrow(counts) = nrow(omega), filled with the posterior covariance matrix for the NEF parametrization of each row of omega. Utility logit performs the NEF transformation and expit reverses it.

Author(s)

Matt Taddy <taddy@chicagobooth.edu>

References

Taddy (2012), *On Estimation and Selection for Topic Models*. http://arxiv.org/abs/1109.4518

See Also

topics, predict.topics

Index