Manuals for SVCM R Package

Xifeng Wang

University of North Carolina-Chaple Hill

xifeng@live.unc.edu

July 2, 2018

Overview

The SVCM R package can do three following things:

1. Repeat the simulation study in SVCM paper
2. Repeat ADHD data analysis in SVCM paper
3. Do your own Real Data Analysis

Repeat the simulation study in SVCM paper

To repeat the simulation study in SVCM paper, we only need to use the Simtest() function in the R package. The function will save the output into CSV file in working directory

Example

```
pattern=6;
pr=0.5;
nf1=ef=f1=f2=1;
n=60;
Simtest(n, pattern, pr, nf1,ef, f1,f2);
```

Repeat the simulation study in SVCM paper

Note:

- It usually takes about 15-18 hours to run the code. So running the code on server is strongly recommended.
- The Average bias , RMS, SD, and RE of $\beta(d_0)$ parameters in the five ROIs at 3 different scales (h_0, h_5, h_{10}) are showed in "Result" file.
- Estimates (ES) and standard errors (SE) of rejection rates for pixels inside the five ROIs at two different scales (h_0, h_{10}) are showed in "Result1" file.

Repeat ADHD data analysis in SVCM paper

Following the steps in SVCM paper, we just need to use the related functions in the R package.
Note:

- The image data are stored in data folder.
- The grey and white matter image data is stored in mat file so we need Rmatlab package is required.
- Before smoothing the individual residual, we have to set-up the partition of the entire brain image. The dimension of ADHD data is 128*128*96 so we part the image into disjoint smaller image with dimension 8*8*6 each.
- For output display, we can use software like MRIcron(http://people.cas.sc.edu/rorden/mricron/index.html)

Repeat ADHD data analysis in SVCM paper

We put all the data in R working directory and we suppose the working directory is "/lustre/scr/x/i/xifeng/svcmrealdata/".

Example

```
library(oro.nifti)
aa=read.table(file="newid.txt",header = TRUE,sep=',')
aa=as.matrix(aa)
aa=aa[,2]

bb=read.table(file="newNyuDemo.txt")
bb=as.matrix(bb)

library("R.matlab")
path <- ("/lustre/scr/x/i/xifeng/svcmrealdata/")
pathname <- file.path(path, "newNyuGwImg.mat")
cc <- readMat(pathname)
cc=cc[[1]]
```

Repeat ADHD data analysis in SVCM paper

Example (ADHD data code continued)

```
Xdesign=bb;Img=cc;Nonzero=aa;
a1=c(128,128,96); a2=c(8,8,6)
s=smoothResidualRealdata(Xdesign,Img,Nonzero,a1,a2);
covEvalue=s[[3]]; covEvector=s[[4]] ;
mxBeta=s[[1]]; sigmaError=s[[2]]

result2=findVoxelSequenceRealdata(Nonzero,a1);
VSeqId=result2[[1]]
VSeqDist=result2[[2]];
InVSeq=result2[[3]];

results3=smoothParameterRealdata(mxBeta,covEvector,InVSeq,
VSeqId,VSeqDist,Xdesign,covEvalue,sigmaError)
fnlBeta=results3[[1]]; fnlCovb=results3[[2]];
```

Repeat ADHD data analysis in SVCM paper

To accelerate the whole process, we can use parallel computing in finding nearby voxel and smoothing the individual residuals. The code in last page will be replaced by:

Example (ADHD data code continued using parallel computing)

```
core=2
Xdesign=bb;Img=cc;Nonzero=aa;
a1=c(128,128,96); a2=c(8,8,6)
s=smoothResidualRealdata_para(Xdesign,Img,Nonzero,a1,a2,core);
covEvalue=s[[3]]; covEvector=s[[4]] ;
mxBeta=s[[1]]; sigmaError=s[[2]]

result2=findVoxelSequenceRealdata_para(Nonzero,a1,core);
VSeqId=result2[[1]]
VSeqDist=result2[[2]];
InVSeq=result2[[3]];
```

Repeat ADHD data analysis in SVCM paper

Example (ADHD data code continued using parallel computing)

```
results3=smoothParameterRealdata(mxBeta,covEvector,InVSeq,
VSeqId,VSeqDist,Xdesign,covEvalue,sigmaError)
fnlBeta=results3[[1]]; fnlCovb=results3[[2]];
```

Note: The variable "core" is the number of cores will be used in the parallel computing. The value of this variable is based on the how many cores are on the device, which can be get using "detectCores(logical = FALSE)".

Repeat ADHD data analysis in SVCM paper

To save the result into nifti form, we need "oro.nifti" R package again. For example, if we want to see the age and diagnosis interaction just as in the paper, we can use the following code.

Example

```
mask=readANALYZE(fname="newMask");
tempimage1=mask
tempimage1[wmNonzeroId]=fnlBeta[,6];
writeNIfTI(tempimage,filename="smoothbeta6")
```

Note: The above code can help us to store the image data by the same header file with the mask image.So we can keep the same information of image data in the output.

Then we can use the software like MRIcron to display the nifti form data.

Do your own Real Data Analysis

To analyse your own Image data with SVCM method, we need following types of data:

- The 3D image data of all the subjects, should be in nifiti or analyze form.
- Design Matrix: To indicate what kind of covariates you are interested in, such as age ,gender,brain volume. The data can be "txt" or "csv" file.
- Mask image data: To indicate the location of the non-zero element. It should have the same form with the image data of each subject.

Do your own Real Data Analysis

Firstly, we have to read the Mask image to get the nozero location of the image data. And since the 3D image data of each subject are stored seperately, we have to read them together and transform them into a matrix.
We need "oro.nifiti" package in R to read the image data.
To read the Mask image and get the location of non zero element, we can use the following code:

Example

```
library(oro.nifti)
aa=readNIfTI(fname="mask.nii.gz")  #Read mask image
bb=c(aa)
nozeroid=which(bb!=0)       #Get non-zero id in mask image
```

Here we suppose that the Mask image is stored in the working directory and named as "mask.nii.gz".

Do your own Real Data Analysis

Then we try to read the image data of each subject together and transform in to a matrix.
Suppose the image data folder are stored in the following path:
"*/lustre/scr/x/i/xifeng/PNCdata/wangxf/pncmask/segmentation*"
Under this path, there are many data folders and the name of the folder is the subject id. The image data of each subject is stored in each folder and is called "T1".
Then we use the following code to read the image data:

Example

```
mm=list.files(path=a)  #Get all the subject id.

a1=length(nozeroid)
a2=length(mm)
image=matirx(rep(0,a1*a2),a1,a2) #Construct image matrix
```

Do your own Real Data Analysis

Example (Continued)

```
for(i in 1:a2)
{
  hh=paste(a,mm[i],sep='/')
  hh1=paste(hh,"T1",sep='/')
  c=readNIfTI(fname=hh1);
  d=c(c);
  image[,i]=d[nozeroid];
}
```

Then we can see that the we transform all the image data into a matrix and for each column of the matrix, it is the vector of the non-zero element of each subject 's image data.

If your image data is not stored in this way, you can do some modification to the above code,for example , if may use the "paste" and "list.file" function more times to reach the path of each subject's image data.

Do your own Real Data Analysis

Now we have got the image matrix of all the subjects and the location of all the non-zero elements. Then wen can run the SVCM method as we did when we repeat the ADHD data analysis.
Note:

- The demension of the image data we use in the example is 192*256*160 and we part the entire image into smaller image with 8*8*8 dimsension each.
- If your image has different dimension, please set up partition of the entire brain image by yourself appropriately.

Do your own Real Data Analysis

Example

```
bb=read.table(file="dataDemo.txt") #read demographic informati
bb=as.matrix(bb)
Xdesign=bb;
a1=c(192,256,160); #The dimension of the image data
a2=c(8,8,8)  #How to part the entire image on each dimension.
ss=smoothResidualRealdata(Xdesign,image,nozeroid,a1,a2);
covEvalue=ss[[3]]; covEvector=ss[[4]] ;
mxBeta=ss[[1]]; sigmaError=ss[[2]]
```

Do your own Real Data Analysis

Example (Continued)

```
result2=findVoxelSequenceRealdata(nozeroid,a1);
VSeqId=result2[[1]];VSeqDist=result2[[2]];InVSeq=result2[[3]];

results3=smoothParameterRealdata(mxBeta,covEvector,InVSeq,
VSeqId,VSeqDist,Xdesign,covEvalue,sigmaError)

fnlBeta=results3[[1]]; fnlCovb=results3[[2]];
```

When we have the smoothed parameter, we can save them as we did for ADHD data.

And when we get smoothed $\beta(\boldsymbol{d})$ and the covariance matrix, it is also very easy to get Wald statistic.

Do your own Real Data Analysis

Again we can use parallel computing to accelerate the whole process

Example

```
bb=read.table(file="dataDemo.txt") #read demographic informati
bb=as.matrix(bb)
Xdesign=bb;
core=2;
a1=c(192,256,160); #The dimension of the image data
a2=c(8,8,8)  #How to part the entire image on each dimension.
ss=smoothResidualRealdata_para(Xdesign,image,
nozeroid,a1,a2,core);
covEvalue=ss[[3]]; covEvector=ss[[4]] ;
mxBeta=ss[[1]]; sigmaError=ss[[2]]
```

Do your own Real Data Analysis

Example (Continued)

```
result2=findVoxelSequenceRealdata_para(nozeroid,a1,core);
VSeqId=result2[[1]];VSeqDist=result2[[2]];InVSeq=result2[[3]];

results3=smoothParameterRealdata(mxBeta,covEvector,InVSeq,
VSeqId,VSeqDist,Xdesign,covEvalue,sigmaError)

fnlBeta=results3[[1]]; fnlCovb=results3[[2]];
```

References

Zhu, H., Fan, J. and Kong, L. (2014)

Spatially Varying Coefficient Model for Neuroimaging Data with Jump Discontinuities.

Journal of the American Statistical Association Vol. 109, No. 507, 1084-1098