

0. Function preparation

```
create_B=function(d,random=FALSE){#a function to initialize B's
  l=vector('list',k)
  if(random){for(i in 1:k) l[[i]]=matrix(rnorm(dimension[i]^2),dimension[i])}
  else {for(i in 1:k) l[[i]]=diag(dimension[i])}
  return(l)
}
multiple_ttm=function(x,b){#a function to perform consecutive mode-d products
  if(length(b)>1) return(ttm(multiple_ttm(x,b[-length(b)]),b[[length(b)]],length(b)))
  return(ttm(x,b[[1]],1))
}
```

1. Generate Data

```
library(rTensor) #https://cran.r-project.org/web/packages/rTensor/index.html
set.seed(5)
dimension=c(3,4,10)
k=length(dimension)-1
e=rand_tensor(dimension)
X=100*rand_tensor(dimension)
B=create_B(dimension,random=TRUE) #generate random B's
Y=multiple_ttm(X,B)+e
X2=100*rand_tensor(dimension)
e2=rand_tensor(dimension)
Y2=multiple_ttm(X2,B)+e2
#X;B;e;Y # view generated sample data
```

2. Regression

```
N=500 #number of iterations
B_hat=create_B(dimension) #initialize B's as identical matrices
#B_hat=lapply(1:k, function(i) return(B[[i]]+matrix(rnorm(dimension[i]^2),dimension[i])/10))
#B;B_hat
B_hats=list()
Error_Y_hat=numeric(0) #store the prediction errors comparing to Y
Error_Y2_hat=numeric(0) #store the prediction errors comparing to Y2
for(i in 1:N){
  for(j in 1:k){ #each regression iteration
    X_j=k_unfold(X,j)@data
    temp=B_hat[-j]
    temp[[k]]=diag(dimension[k+1])
    X_tilde=X_j%*%t(kronecker_list(rev(temp)))
    Y_j=k_unfold(Y,j)@data
    B_hat[[j]]=Y_j%*%t(X_tilde)%*%solve(X_tilde%*%t(X_tilde))
    #print(B_hat)
  }
  B_hats[[i]]=B_hat #update after each small iteration
  Y_hat=multiple_ttm(X,B_hat)
```

```

Y2_hat=multiple_ttm(X2,B_hat)
Error_Y_hat=c(Error_Y_hat,fnorm(Y_hat-Y))
Error_Y2_hat=c(Error_Y2_hat,fnorm(Y2_hat-Y2))
#cat(i,fnorm(Y_hat-Y),fnorm(Y2_hat-Y2),'\\n')
if(tail(Error_Y_hat,1)<50) break
}
par(mar=c(5,4,4,4)+0.3) #draw the graph
plot(Error_Y_hat,type='l',col='coral',axes=FALSE,xlab='\\ # of iterations',
      ylab='error of Y_hat',main='Prediction Error')
axis(side=2,at=pretty(range(Error_Y_hat)),las=1)
box()
par(new=TRUE)
plot(Error_Y2_hat,type='l',col='skyblue',axes=FALSE,xlab='',ylab='',lty=2)
axis(side=4,at=pretty(range(Error_Y2_hat)),las=1)
mtext("error of Y2_hat", side=4, line=3)
axis(1,pretty(range(1:length(Error_Y_hat))))
legend("right",legend=c("error of Y_hat","error of Y2_hat"),
      text.col=c("coral","skyblue"),lty=c(1,2),col=c("coral","skyblue"),cex=.8,bty="n")

```

Prediction Error

