Subject: Re: help IDL with some equations implementation and optimized coding Posted by Jeremy Bailin on Tue, 01 Jun 2010 12:56:40 GMT

View Forum Message <> Reply to Message

```
On Jun 1, 1:30 am, lila hadji <lhad...@gmail.com> wrote:
> Dear everybody,
>
> I tried to implement the dictionary update step in the Method of
> Optimal Directions, as explained in the paper (basically equations 8,
> 9, 17, 18 and 25)
>
 %
          "Method of optimal directions for frame design",
           K. Engan, S. O. Aase, and J. H. Husøy,
> %
           in Proc. ICASSP, Phoenix, AZ, Mar. 1999, pp. 2443-2446.
>
>
  First link of the page:
>
   http://scholar.google.com/scholar?hl=fr&q=%22Method+of+o ptimal+direct...
>
 The problem is that I didn't obtain good results - as it is expected
> within the paper - with my implementation and I was wondering of
> anyone can check out the code and the equations in the paper and help
> me out why my code does not give the expected results and if there is
> any improvements I can do on the code to get rid of the for-loop.
> please?
>
> Here is the version with the loop.
> ;;#IDL Code#
> $Main$
> ;N=16 : size of each sample
> ;m=64: number of samples
 ;k=32:number of atoms in the dictionary to be learned
>
> :Y=randomn(seed,m,n) : samples
> ;D=randomn(seed, k,n): dictionary to be learned
> ;X=randomn(seed,m,k): sparse codes to construct in the domain D
>
> B=dblarr(N,k,/nozero)
> ones=dblarr(N,/nozero)
 ones(*)=1
>
> A=invert(transpose(x)#X,/double)
 R=Y-D##X
>
>
 for i=0,k-1 do begin
    xi=X(*,i)
```

```
> B(*,i)=total((ones##xi)*R,1)
> end
>
> delta=A##B
> D=D+transpose(delta)
> end
>
> ;; #IDL Code#
>
> Thank you very much for your help in advance.
> Cheers
> Lila
I don't know about the first question, but as for the second you can replace lots of that code including the for loop with:
B = matrix_multiply(R, X, /atranspose)
-Jeremy.
```

Subject: Re: help IDL with some equations implementation and optimized coding Posted by lila hadji on Tue, 01 Jun 2010 16:50:13 GMT View Forum Message <> Reply to Message

```
On 1 juin, 14:56, Jeremy Bailin <astroco...@gmail.com> wrote:
> On Jun 1, 1:30 am, lila hadji <lhad...@gmail.com> wrote:
>
>
   Dear everybody,
>> I tried to implement the dictionary update step in the Method of
>> Optimal Directions, as explained in the paper (basically equations 8,
>> 9, 17, 18 and 25)
            "Method of optimal directions for frame design",
>> %
>> %
            K. Engan, S. O. Aase, and J. H. Husøy,
>> %
            in Proc. ICASSP, Phoenix, AZ, Mar. 1999, pp. 2443Đ2446.
>> First link of the page:
>> http://scholar.google.com/scholar?hl=fr&q=%22Method+of+o ptimal+direct...
>> The problem is that I didn't obtain good results - as it is expected
>> within the paper - with my implementation and I was wondering of
>> anyone can check out the code and the equations in the paper and help
>> me out why my code does not give the expected results and if there is
```

```
>> any improvements I can do on the code to get rid of the for-loop,
>> please?
>> Here is the version with the loop.
>> ;;#IDL Code#
>> $Main$
>> ;N=16 : size of each sample
>> ;m=64: number of samples
>> ;k=32:number of atoms in the dictionary to be learned
>> ;Y=randomn(seed,m,n) : samples
>> ;D=randomn(seed, k,n): dictionary to be learned
>> ;X=randomn(seed,m,k): sparse codes to construct in the domain D
>
>> B=dblarr(N,k,/nozero)
>> ones=dblarr(N,/nozero)
>> ones(*)=1
>> A=invert(transpose(x)#X,/double)
>> R=Y-D##X
>> for i=0,k-1 do begin
      xi=X(*,i)
      B(*,i)=total((ones##xi)*R,1)
>>
   end
>>
>> delta=A##B
>> D=D+transpose(delta)
>> end
>
>> ;; #IDL Code#
>> Thank you very much for your help in advance.
>> Cheers
>> Lila
> I don't know about the first question, but as for the second you can
  replace lots of that code including the for loop with:
>
 B = matrix_multiply(R, X, /atranspose)
> -Jeremy.
```

Thank you very much, it's actually the same. I didn't see the matrices' multiplication in the first place. It actually works and is loop-free besides the outer loop that is necessary for the

convergence. Thank you very much for your help! Cheers Lila