Re: How can we find (approximatly) pricipal component without explicit calculating covariance matrix ?

Source: http://sci.tech−archive.net/Archive/sci.math.num−analysis/2006−04/msg00071.html

---

- *From*: spellucci@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx (Peter Spellucci)
- *Date*: Thu, 6 Apr 2006 11:04:57 +0000 (UTC)

In article <e12j0r$a5$1@xxxxxxxxxxxxxxxxxx>,
"shna" <nsh1979@xxxxxxxxxxxxx> writes:

> "Peter Spellucci" <spellucci@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in
> message news:e10a27$9dq$1@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
>
> > In article <e0vft9$eo4$1@xxxxxxxxxxxxxxxxxx>,
> > "shna" <nsh1979@xxxxxxxxxxxxx> writes:
> >
> > > Hi, all.
> > >
> > > Recently, I have been front of a problem which is formulated
> > > as follows .
> > >
> > > Let v_1 and v_2 be given vectors.
> > >
> > > C = v_1 v_1^t + v_2 v_2 ^ t
> > >
> > > is semi−positive definite matrix.
> >
> > C has rank two at most. hence there are n−2 eigenvalues zero, (at
> > least),
> > with the invariant subspace as the orthogonal complement to span of v_1
> > and
> > v_2.
> >
> > > We want to vector v to approximate well C
> > >
> > > which decomposed into v v^t.
> > >
> > > In other words,

v = argmin || C − v v ^t ||

where ||.|| is matrix norm (F−norm).

In fact, v will be the pricipal vector given C.

However, the problem is that feature vector v_1 and v_2 are very
high−dimensional matrix (more than 100,000).
So, C cannot be calculated effectively.

We want to approximate v without calculate C.

in which form do you "have" C???
maybe as given x −> some blackbox −> y=C*x out ??
you are sure that this gives y without noise? (means: the dimension of span C*x , x arbitrary is truly 2?

I mean that C is defined in the above formula. In other words,
C is exactly a form of v_1 v_1^t + v_2 v_2 ^ t and C is not a blackbox.
Simply, I did pointout that C is too big to store because its dimension is high.
There is no noise in C itself.
But, after reading your messsage below,
The power method seems to need only calculate the form C*x.

yes

Clearly, calcuating C*x is possible without explicitly using C.

Do you have any idea or reference for this procedure ??

For example (it is only an example),

I can derive the following procedure for approximating v.

First, let's rewrite the above formula as follows.

C = v v^t (Here, we used '=' instead of '~'.) − (A)

>>> v_1 v_1^t + v_2 v_2 ^ t = v v^t
>>>
>>> If we multiply an arbitrary vector x on both side,
>>> then
>>> v_1 v_1^t x + v_2 v_2 ^ t x = v v^t x − (B)

>> this is meant in terms of approximately equal !

>>> From the above formula,
>>> we obtain a fixed point equation as like.
>>>
>>> v <− (v_1 (v_1^t x) + v_2 (v_2^t x)) / (v^t x) − (*)

>> no no, but you are on the way of reinventing the power method

>>> Then, this solution (if it can be one of solution) is calcuated
>>> without
>>> explictly C.
>>>
>>> My second question is, is there any theory how (*)−like
>>> equation is
>>> effective???.

>> yes, the theory of the power method is completely known

> Thank you !
> I didn't know that it is one of power method, before you mention.
> Currently, I want that (*) is rewritten as following.
> By setting x = v, Cx = x x^t x
> Then, iteration process at one step is derived as
> x <− C x / || C x ||^2 −−−−−−−−−−−−−−−−−−− (C)
>
> I think that (C) is not a standard power method.

yes, you do not normalize , but normalize by the square of then norm
this will not converge in the sense of giving an eigenvector:
if the dominant eigenvalue of C is larger than one, it will "converge"
to zero, but in direction it will converge correctly (provided the
initial vector is sufficiently general)

>But, do you mean the above iteration (C) is reasonble although it is not
>standard ?
no, x<−Cx/||Cx||
is the correct way of doing this
>Is it the power method sufficently general to contain (C)−like formula ?
>
>> >
>> >For third question, (It is somewhat related to the second question),
>> >let v be a vector such that (B) is satisfied for all x.
> > you forgot that your equality is approximately only
>> hence in what sense this "approximately should be meant?
>> e.g. for x on the unit sphere???
>
>Yes. maybe i mean that assumption is "approximately" satisfied.

And, I want to know 'approximation' theory.

Then, is there any meaning to find such v for obtaining real
solution of
(A).
("Meaning" means that two solution is related with a strong
inequility,
...
something like that).

Thank you.

From
Seung−Hoon Na

given a sufficiently random x0 of euclidean length 1:
compute
x1=C*x0,
x2=C*x1
now you have three vectors .
if your assumptions hold, then
C=lambda_1*u_1*u_1'+ lambda_2*u_2*u_2' , u_1, u_2 of lenght 1 and
u_1'*u_2=0
and lambda_1>=lambda_2>0 or lambda_1> lambda_2=0 (this is the rank one
case,
v_2 a multiple of v_1) . what you want to get is u_1.
now, you can develop x0 into a sum
x_0 = alpha_1*u_1+alpha_2*u_2+ sum_{i=3 to n} alpha_i*u_i
where u_3 to u_n are an orthonormal basis of the orthogonal complement to

span
{u_1,u_2}.
"sufficiently random" means alpha_1 and alpha_2 not =0.

inserting this into the procedure above you find
x1= (lambda_1*alpha_1*u_1+lambda_2*alpha_2*u_2)
x2= (lambda_1^2*alpha_1*u_1+lambda_2^2*alpha_2*u_2)
hence there are c0,c1,c2 such that

c0*x0+c1*x1+c2*x2=0 c0,c1,c2 not all zero.

you get these from the svd of the matrix (x0,x1,x2) (the economy size one)

(x0,x1,x2)=W*diag(s1,s2,0)*V' V 3 times 3, W n times 3 orthogonal

V orthonormal.
namely

(x0,x1,x2)*V=[w_1*s1,w_2*s2, 0]

as the last column of V. in matlab notation V(:,3)=[c0;c1;c2].
now your lambda_1, lambda_2 are the zeros of the quadratic
c0+c1*lambda+c2*lambda^2=0

OK. I can follow that.

let lambda_1>=lambda_2

if lambda_1=lambda_2, then you can take x0/norm(x0)*sqrt(lambda_1)
as your desired v. otherwise, you must compute
u_1=(lambda_2*x0−x1)/norm of numerator * sqrt(lambda_1)
as v

if u_1 and u_2 are the eigenvectors , normalized, for lambda_1
and lambda_2, then from your original assumption on C

C=lambda_1*u_1*u_1' + lambda_2*u_2*u_2'
hence your v_1=sqrt(lambda_1)*u_1 etc
now, given x0,x1 and lambda_1, lambda_2 already computed,
I simply eliminate u_1 from the the two equations for x0=... and x1=...
hth
peter

How was v derived? Could you explain detaily of derivation process ?
Or, I will try to follow your formulation after understanding sufficently
the power method.

done

however, should noise be in your black box, then the svd will not deliver
the third singular value as zero, but in some sense "small". it is then
your
problem to decide whether "small" is "small enough".

hth
peter

.