Componentes principales con introducción explícita de regresores no lineales: ejemplo Matlab.

Presentación en vídeo en http://personales.upv.es/asala/YT/V/pcanlx.html

Este código ejecutó correctamente en Matlab R2018b.

Tabla de Contenidos

1.-Motivación y esbozo de teoría

Si se forma un vector de regresores (características, "f*eature vector*") que pueda incluir no-linealidades, $\phi(x)$, el PCA puede descubrir que los datos se agrupan en variedades ("*manifolds*") con curvatura, descritas por una combinación lineal de las características no lineales.

El PCA lineal sólo puede determinar subespacios (hiperplanos) donde se agrupen los datos.

Por lo tanto, dados (x1, x2, ... xn), en vez de formar X=[x1 x2 .... xn] para PCA lineal, formamos $X=[\phi(x_1)\ \ \phi(x_2)\ \ \dots\ \ \phi(x_n)]$ como matriz de datos. La descomposición en valores singulares de Xproveerá de información sobre si algunas combinaciones lineales de estas características tienen una desviación típica muy pequeña, que podamos, por tanto, suponer que son un "modelo no lineal" identificado, generalizando la idea tras el PCA o Total Least Squares lineal.

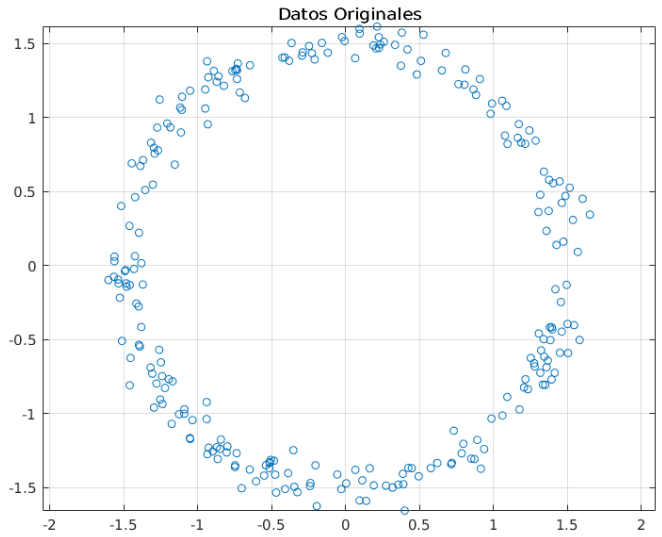
El PCA no-lineal puede considerar unas características de **más dimension** que los datos originales para modelar mejor: $\phi(x)=(x,\|x\|,sin(x_1)log(x_1-x_2)\dots)$.

Nota: aunque los "ruidos de medida" en los elementos de x podrían considerarse no correlados, en las características no lineales aparecerán repetidas veces (introduciendo correlación) y una hipotética "distribución normal" del ruido de medida sería deformada en $\phi(x)$. Por tanto, la interpretación estadística (TLS) ya no es formalmente válida, pero sigue siendo útil en la práctica.

2.-Ejemplo

```
N=250;
r=1.5+randn(N,1)*0.075;
th=rand(N,1)*2*pi;
x=r.*cos(th);
y=r.*sin(th);
```

```
plot(x,y,'o'), grid on, title('Datos Originales'), axis equal
```



PCA lineal

```
X0=[x y]
```

```
X0 = 250×2
    1.4257    0.1397
    1.2148    0.8221
   -1.3708   -0.1275
    1.3973   -0.4300
    0.7959   -1.2023
    1.3220   -0.5720
    1.4557   -0.2455
   -1.1052    1.1401
   -0.6462    1.3533
    0.3176   -1.4983
     ⋮
```

```
medias=mean(X0)
```

```
medias = 1×2
   -0.0708   -0.0757
```

```
desvtip=std(X0)
```

```
desvtip = 1×2
    1.0460    1.0751
```

```
X=(X0-medias)*inv(diag(desvtip));

[U,S,V]=svd(X,'econ');
diag(S)'
```

```
ans = 1×2
   15.7990   15.7604
```

No detecta dimensionalidad reducida, todos los valores son similares.

PCA no lineal

Sospechamos que podría existir alguna relación cuadrática entre los datos. Definimos un vector de características:

```
phi=@(x1,x2) [x1 x2 x1.^2 x2.^2 x1.*x2];
```

y formamos la matriz de datos extendida.

```
X0=phi(x,y)
```

```
X0 = 250×5
    1.4257    0.1397    2.0326    0.0195    0.1991
    1.2148    0.8221    1.4758    0.6758    0.9987
   -1.3708   -0.1275    1.8791    0.0163    0.1748
    1.3973   -0.4300    1.9525    0.1849   -0.6009
    0.7959   -1.2023    0.6334    1.4456   -0.9569
    1.3220   -0.5720    1.7478    0.3272   -0.7562
    1.4557   -0.2455    2.1190    0.0603   -0.3573
   -1.1052    1.1401    1.2215    1.2999   -1.2601
   -0.6462    1.3533    0.4176    1.8314   -0.8745
    0.3176   -1.4983    0.1009    2.2448   -0.4759
     ⋮
```

```
medias=mean(X0)
```

```
medias = 1×5
   -0.0708   -0.0757    1.0947    1.1570    0.0081
```

```
desvtip=std(X0)
```

```
desvtip = 1×5
    1.0460    1.0751    0.7990    0.8016    0.8101
```

```
X=(X0-medias)*inv(diag(desvtip)) %escalado media cero, desv.típ. unidad.
```

```
X = 250×5
    1.4307    0.2004    1.1739   -1.4190    0.2358
    1.2291    0.8351    0.4770   -0.6002    1.2228
   -1.2429   -0.0481    0.9818   -1.4230    0.2057
    1.4036   -0.3295    1.0736   -1.2126   -0.7518
    0.8286   -1.0479   -0.5773    0.3600   -1.1912
    1.3316   -0.4616    0.8174   -1.0352   -0.9435
    1.4594   -0.1579    1.2821   -1.3681   -0.4511
   -0.9890    1.1309    0.1587    0.1782   -1.5655
   -0.5501    1.3292   -0.8475    0.8413   -1.0895
    0.3714   -1.3231   -1.2438    1.3571   -0.5975
     ⋮
```

X contiene el vector de características normalizado para que todos sus componentes tengan media cero y varianza unidad.

$$\phi_{normalizado}(x) = (\phi(x) - mean(\phi(x))./std(\phi(x))$$

```
[U,S,V]=svd(X,'econ')
```

```
U = 250×5
   -0.0848   -0.0260    0.0620   -0.0630    0.0543
   -0.0338   -0.0056    0.0292   -0.1285    0.0261
   -0.0738    0.0483   -0.0637    0.0258    0.1001
   -0.0772   -0.0492    0.0812    0.0067    0.0314
    0.0258   -0.0375    0.0830    0.0752    0.0494
   -0.0637   -0.0521    0.0831    0.0236    0.0494
   -0.0883   -0.0430    0.0771   -0.0165    0.0191
   -0.0024   -0.0781   -0.0934    0.0617   -0.0713
    0.0512   -0.0806   -0.0768    0.0195    0.0056
    0.0814    0.0032    0.0722    0.0715   -0.0265
     ⋮
S = 5×5
   22.1208         0         0         0         0
         0   16.8914         0         0         0
         0         0   15.7567         0         0
         0         0         0   14.5707         0
         0         0         0         0    3.1256
V = 5×5
   -0.0403   -0.4272    0.7982   -0.4227   -0.0019
   -0.0129   -0.5688   -0.6018   -0.5604    0.0043
   -0.7054    0.0426   -0.0202   -0.0107   -0.7072
    0.7051   -0.0504    0.0144    0.0141   -0.7070
    0.0587    0.6997    0.0004   -0.7120   -0.0056
```

Hemos detectado que hay **un** componente (el último) de mucha menor desviación típica que el resto. Por tanto, los datos podrían estar próximos a una variedad de dimensión 1 (una curva en el plano) expresable como "[polinomio de grado 2] = 0".

Reconstrucción de la curva (variedad) en coordenadas originales

La ecuación de la curva, en concreto, sería $\phi_{normalizada}(x) * V(:,5) = 0$.

Para escribirlo en las coordenadas de los datos, hay que deshacer la normalización:

$$0 = \frac{\phi(x) - media}{desvtip} * V(:,5) = \phi(x) * \frac{V(:,5)}{desvtip} - media * \frac{V(:,5)}{desvtip} = \phi(x) * w - m$$

```
w=inv(diag(desvtip))*V(:,5)
```

```
w = 5×1
   -0.0018
    0.0040
   -0.8851
   -0.8820
   -0.0069
```

```
m=medias*w
```

```
m = -1.9895
```

```
phitrans=@(x,y) phi(x,y)*w-m
```

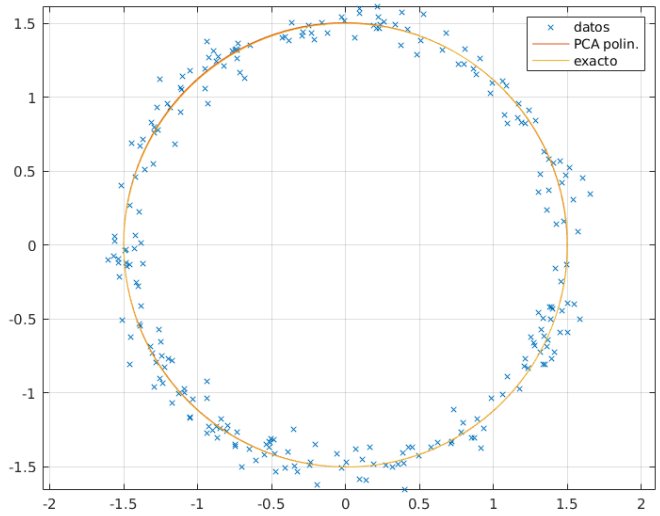
```
phitrans = function_handle with value:

    @(x,y)phi(x,y)*w-m
```

```
plot(x,y,'x') %datos
hold on
fimplicit(@(x,y) phi(x,y)*w-m)   %modelo estimado
```

```
Warning: Function behaves unexpectedly on array inputs. To improve performance, properly vectorize
your function to return an output with the same size and shape as the input arguments.
```

```
fimplicit(@(x,y) x.^2+y.^2-1.5^2) %somos DIOS, y sabemos de verdad qué originó los dato
hold off, grid on, axis equal
legend('datos','PCA polin.','exacto')
```



3.- Conclusiones

Si se conoce el tipo de no-linealidad subyacente en un conjunto de datos, puede incorporarse al PCA para obtener mejores resultados en identificación y posterior detección de cambios.