Eigenwerte, Eigenvektoren

StochastischeProzesse Haftendorn 2011

Eigenwerte (Zur Erzeugung der stochastischen Matrizen siehe 2. Problem)

Es gibt eingebaute Befehle dafür $\mathrm{eigVl}(matrix)$ $eigVec(matrix)$. Weiter gibt es eine Bibliothek linalgcas.tns, sie enthält **linalgcas\eigenvals**$(matrix)$ und **linalgcas\eigenvects**$(matrix)$.

aa$:=\begin{bmatrix} 0.73 & 0.13 & 0.14 \\ 0.43 & 0.43 & 0.14 \\ 0.15 & 0.15 & 0.7 \end{bmatrix}$ Die Matrix ist mit tool\stomat(3) erzeugt, dann aber mit copy und paste dem aa direkt zugewiesen, damit bei anderer Auswertung keine andere Matrix erzeugt wird.

$\mathrm{eigVl}(\mathbf{aa}) \blacktriangleright \{1.,0.3,0.56\}$ Der eingebaute Befehl versagt aber bei komplexen Eigenwerten, wenn er im real−Modus ist.In der Stochastik interessiert sowieso nur der Eigenwert 1, mit

$\mathbf{ee}:=\mathrm{identity}(3) \blacktriangleright \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

und an $\det(\mathbf{aa}-\mathbf{ee}) \blacktriangleright 0$ kann man sehen, dass die Matrix aa den Eigenwert 1 wirklich hat.

Eine Alternative ist aus dem Bibliotheksprogramm linalscas.tns zu nehmen
linalgcas\eigenvals(aa) $\blacktriangleright \{0.3,0.56,1\}$, aber man weiß es ja schon, es gibt den Eigenwert 1.

Übrigens kann man mit den anderen Eigenwerten keine stochastischen Prozesse beschreiben, weil v auf z.B. 0.3*v abgebildet würde, das ja ja kein stochastischer Vektor ein, wenn v einer ist.

1.1

Eigenvektoren ..

Am einfachsten bestimmt man den stabilen Vektor v, der $v\cdot\mathbf{aa}=v$ erfüllt, aus einer hohen Matrixpotenz **stabil**$:=\mathbf{aa}^{30}$ ▸ $\begin{bmatrix} 0.487013 & 0.194805 & 0.318182 \\ 0.487013 & 0.194805 & 0.318182 \\ 0.487013 & 0.194805 & 0.318182 \end{bmatrix}$

Wenn die Zeilen sind gleich sind, ist der Exponent hoch genug gewählt. Jede Zeile ist ist der gesuchte Eigenvektor.

Probe: **evp**$:=$**stabil**$[1]$ ▸ $\begin{bmatrix} 0.487013 & 0.194805 & 0.318182 \end{bmatrix}$

$\mathbf{evp}\cdot\mathbf{aa}^{30}$ ▸ $\begin{bmatrix} 0.487013 & 0.194805 & 0.318182 \end{bmatrix}$

Theoretisch fundierte Bestimmung: Das übliche Eigenvektor−Konzept arbeitet mit Spaltenvektoren. Darum ist es hier einfacher, den einzigen Eigenvektor, der interessiert, nänmlich den zum Eigenwert 1 direkt aus einer Gleichung zu bestimmen. Die Gleichung v **aa**$=v$ muss also erfüllt werden und v muss ein stochastischer Vektorsein, also Spaltensumme 1 haben. Dazu müssen wir mit mat▸list aus der Vektorgleichung ein Gleichungssystem machen, dem wir die Normierungsgleichung noch hinzufügen.

loo$:=\mathrm{solve}\left(\mathrm{augment}\left(\mathrm{mat}\blacktriangleright\mathrm{list}\left(\begin{bmatrix} x & y & z \end{bmatrix}\cdot\mathbf{aa}=\begin{bmatrix} x & y & z \end{bmatrix}\right),\{x+y+z=1\}\right),\{x,y,z\}\right)$

▸ $x=0.487013$ and $y=0.194805$ and $z=0.318182$

Dies sind die Komponentendes stabilen Vektors, des Eigenvektors.

1.2

Weitere Beispiele, Matrizen erzeugt mit tool\stomat(3)

$\mathbf{a1}:=\begin{bmatrix}0.32 & 0.32 & 0.36\\ 0.86 & 0.05 & 0.09\\ 0.93 & 0.06 & 0.01\end{bmatrix} ▸ \begin{bmatrix}0.32 & 0.32 & 0.36\\ 0.86 & 0.05 & 0.09\\ 0.93 & 0.06 & 0.01\end{bmatrix}$ $\det(\mathbf{a1}-\mathbf{ee}) ▸ 0$ also 1 ist Eigenvektor

Bestimmung des zugörigen Eigenvektors

$\mathbf{lo1}:=\text{solve}(\text{augment}(\text{mat▸list}([x\ \ y\ \ z]\cdot\mathbf{a1}=[x\ \ y\ \ z]),\{x+y+z=1\}),\{x,y,z\})$

$▸\ x=0.568692 \text{ and } y=0.205802 \text{ and } z=0.225506$

Der Eigenvektor lässt sich auch aus der hohen Potenz ablesen.

$\mathbf{stabil1}:=\mathbf{a1}^{30} ▸ \begin{bmatrix}0.568692 & 0.205802 & 0.225506\\ 0.568692 & 0.205802 & 0.225506\\ 0.568692 & 0.205802 & 0.225506\end{bmatrix}$

$\mathbf{eva1}:=\mathbf{stabil1}[1] ▸ [0.568692\ \ 0.205802\ \ 0.225506]$

Probe $\mathbf{eva1}\cdot\mathbf{a1} ▸ [0.568692\ \ 0.205802\ \ 0.225506]$ Dies ist der stabile Vektor

$[333\ \ 333\ \ 334]\cdot\mathbf{a1}^{30} ▸ [568.692\ \ 205.802\ \ 225.506]$

Von 1000 Leuten mit zunächst gleichverteilten Merkmalen haben auf lange Sicht 567 Merkmal 1, 206 Merkmal 2 und 225 haben Merkmal 3.

1.3

Weitere Beispiele, Matrizen erzeigt mit tool\stomat(3)

$\mathbf{a2} \blacktriangleright \begin{bmatrix} 0.12 & 0.12 & 0.76 \\ 0.11 & 0.11 & 0.78 \\ 0.4 & 0.4 & 0.2 \end{bmatrix}$ $\det(\mathbf{a2}-\mathbf{ee}) \blacktriangleright 0$ also 1 ist Eigenwert

Bestimmung des zugörigen Eigenvektors

$\mathbf{lo2}:=\text{solve}\left(\text{augment}\left(\text{mat}\blacktriangleright\text{list}\left(\begin{bmatrix} x & y & z \end{bmatrix}\cdot\mathbf{a2}=\begin{bmatrix} x & y & z \end{bmatrix}\right),\{x+y+z=1\}\right),\{x,y,z\}\right)$

$\blacktriangleright x=0.254777 \text{ and } y=0.254777 \text{ and } z=0.490446$

Der Eigenvektor lässt sich auch aus der hohen Potenz ablesen.

$\mathbf{stabil2}:=\mathbf{a2}^{30} \blacktriangleright \begin{bmatrix} 0.254777 & 0.254777 & 0.490446 \\ 0.254777 & 0.254777 & 0.490446 \\ 0.254777 & 0.254777 & 0.490446 \end{bmatrix}$

$\mathbf{eva2}:=\mathbf{stabil2}[1] \blacktriangleright \begin{bmatrix} 0.254777 & 0.254777 & 0.490446 \end{bmatrix}$

Probe $\mathbf{eva2}\cdot\mathbf{a2} \blacktriangleright \begin{bmatrix} 0.254777 & 0.254777 & 0.490446 \end{bmatrix}$ Dies ist der stabile Vektor

$\begin{bmatrix} 333 & 333 & 334 \end{bmatrix}\cdot\mathbf{a2}^{30} \blacktriangleright \begin{bmatrix} 254.777 & 254.777 & 490.446 \end{bmatrix}$

Von 1000 Leuten mit zunächst gleichverteilten Merkmalen haben auf lange

Sicht 255 Merkmal 1, 255 Merkmal 2 und 490 haben Merkmal 3.

1.4

Die in der Bibliothek linalgcas.tns vorhandene Funktion

linalgcas\eigenvects(**aa**,1) ▸ $\begin{bmatrix}1\\1.\\1.\end{bmatrix}$ liefet hier erstmal Unsinn, da sich das Konzept auf

Spaltenvektoren bezieht. Darum wird aa transponiert, das Zeichen ᵀ ist bei den Sonderzeichen.

linalgcas\eigenvects($\mathbf{aa}^T$,1) ▸ $\begin{bmatrix}1\\0.4\\0.653333\end{bmatrix}$

nor:=sum(mat▸list(*evt*)) ▸ Fehler: Ungültiger Datentyp $\left(\frac{evt}{\mathbf{nor}}\right)^T$ ▸ Fehler: Ungültiger Datentyp

Also mit dem Trick schafft man es. Ebenso mit der im TI Nspire eingebauten Funktion.

eigVc($\mathbf{aa}^T$) ▸ $\begin{bmatrix}-0.79384 & -0.731792 & 0.707107\\-0.317536 & 0.052271 & -0.707107\\-0.518642 & 0.679521 & -1.44254\text{E}-15\end{bmatrix}$ **eve**:= $-\begin{bmatrix}-0.79384\\-0.317536\\-0.518642\end{bmatrix}$ ▸ $\begin{bmatrix}0.79384\\0.317536\\0.518642\end{bmatrix}$

noe:=sum(mat▸list(**eve**)) ▸ 1.63002 $\left(\frac{\mathbf{eve}}{\mathbf{noe}}\right)^T$ ▸ $\begin{bmatrix}0.487013 & 0.194805 & 0.318182\end{bmatrix}$

Die erste Spalte musste man nehmen, da eigvl die 1 zuerst genannt hatte(s.o.). Außerdem können die anderen wegen der Vorzeichen nicht auf stochastische Vektoren umgeschrieben werden. Aber wie gesagt, der Aufwand lohnt sich nicht, aber schön wenn alles passt.

1.5

Stochastische Matrizen, einfach

Stochastische Matrizen (einfache Version) Haftendorn 2011

Im Umgang mit stochastischen Prozessen, wie Markov−Ketten, gibt es Zustandsvektoren und stochastische Matrizen. In der Fachliteratur sind die ersteren Zeilenvektoren und die Matrizen haben Zeilensumme 1. In Schulbücher wird es z.T. anders herum gemacht, um keinen Wechsel zu den geometrisch gedeutenen Vektoren der linearen Algebra zu haben. Das hat didaktischen Wert.

Hier folge ich der Fachliteartur. Zustandsvektor z.B. $\mathbf{v}:=\begin{bmatrix}0.1 & 0.3 & 0.6\end{bmatrix} \blacktriangleright \begin{bmatrix}0.1 & 0.3 & 0.6\end{bmatrix}$

Unten ist die Progammierung für die Funktion stma(n) angegeben.

$\mathbf{sa}:=\mathbf{stma}(3) \blacktriangleright \begin{bmatrix}0.1 & 0.1 & 0.8\\ 0.95 & 0.03 & 0.02\\ 0.76 & 0.06 & 0.18\end{bmatrix}$ und nun $\mathbf{v}\cdot\mathbf{sa} \blacktriangleright \begin{bmatrix}0.751 & 0.055 & 0.194\end{bmatrix}$ Vektor links!!!!!

Stochastische Matrizen haben Zeilensumme 1

Im Bibliotheksprogramm tool.tns ist der entsprechende Befehl $\mathbf{tool\backslash stomat}(3) \blacktriangleright \begin{bmatrix}0.42 & 0.42 & 0.16\\ 0.59 & 0.4 & 0.01\\ 1. & 0. & 0.\end{bmatrix}$

den braucht man, wenn man im mehreren Problemen einer oder mehrerer Dateien stochastische Zufalls−Matrizen braucht.

Allerdings haben die Komponenten nicht den Erwartungswert 1/n wie es eigentlich sein müsste.

Dieser Mangel wird von der Programmierung in der Datei **StochastischeMatrizen.tns** behoben.

2.1

Beispiel für stochastische Matrix stomat(n) ..

aa:=**stma**(3) ▸ $\begin{bmatrix} 0.67 & 0.27 & 0.06 \\ 0.02 & 0.02 & 0.96 \\ 0.78 & 0.11 & 0.11 \end{bmatrix}$ erzeugt stochastische nxn Zufallsmatrix

Wenn einem die nicht gefällt, kann man sie ja modifizieren. Siehe auch "Stoch− Matrizen"

Dabei sind die Zeilensummen 1 **start**:=$\begin{bmatrix} 10 & 30 & 40 \end{bmatrix}$ ▸ $\begin{bmatrix} 10 & 30 & 40 \end{bmatrix}$

start·aa ▸ $\begin{bmatrix} 38.5 & 7.7 & 33.8 \end{bmatrix}$ **aa**^{20} ▸ $\begin{bmatrix} 0.575742 & 0.18543 & 0.238828 \\ 0.575742 & 0.18543 & 0.238828 \\ 0.575742 & 0.18543 & 0.238828 \end{bmatrix}$

Die **end**:=**start·aa**^{20} ▸ $\begin{bmatrix} 46.0593 & 14.8344 & 19.1063 \end{bmatrix}$

$\sum_{i=1}^{3} (\mathbf{start}[1,i])$ ▸ 80 $\sum_{i=1}^{3} (\mathbf{end}[1,i])$ ▸ 80.

aa^{80} ▸ $\begin{bmatrix} 0.575742 & 0.18543 & 0.238828 \\ 0.575742 & 0.18543 & 0.238828 \\ 0.575742 & 0.18543 & 0.238828 \end{bmatrix}$ gibt die langfristige Verteilung der 80 Leute an.

2.2

stma 2/15

```
Define LibPub stma(n)=
Func
© stma(n) gibt eine quadr. stochastische Matrixdim
  nxn aus.
Local i,j,ma,lima,li,z:  ma:=newMat(n,n):   lima:={ }
 For i,1,n
  z:=randInt(0,100):    li:={z}
  For j,2,n−1
     While sum(li)+z>100
        z:=randInt(0,100)
      EndWhile
    li:=augment(li,{z})
    EndFor
    z:=100−sum(li):  li:=augment(li,{z}): li:=tool\perm(li
    lima:=augment(lima,li)
  EndFor
  lima:=lima·0.01
 Return list▸mat(lima,n)
EndFunc
```

Die stochastische Matrix wird zeilenweise aus n stochastischen Vektoren erzeugt:

Es wird in der Einheit % von 0 bis 100 ein erster Wert zufällig erzeugt und in die Liste geschrieben. Ein weiter Wert kann den Rest bis 100 ausschöpfen u.s.w..

Damit nicht immer die großen Werte vorn stehen, wird die Liste li noch zufällig permutiert. Damit ist ein stochastischer Vektor erzeugt.

Am Ende werden die Prozente noch richtig als Dezimalzahlen zwischen 0 und 1 angegeben.

2.3