

Proste modele ze złożonym zachowaniem czyli o chaosie

29 kwietnia 2014

Komputer jest narzędziem coraz częściej stosowanym przez naukowców do ukazywania skrzętnie ukrywanych przez naturę tajemnic. Symulacja, obok eksperymentu i teorii, staje się trzecim sposobem badania świata.

Na Uniwersytecie Śląskim trzy lata temu rozpoczęliśmy program integracji metod komputerowych z edukacją. W efekcie powstało wiele niezwykle fascynujących materiałów dydaktycznych pozwalających na łatwiejsze i bardziej dogłębne poznanie wielu tematów. Jako podstawowe narzędzie został wybrany język Python, który wraz z potęgą dostępnych w nim bibliotek naukowych, stanowi obecnie chyba najlepsze rozwiązanie do “komputerowego eksperymentowania” z równaniami, obrazami czy danymi. Jedną z ciekawszych implementacji kompletnego środowiska pracy jest program Sage[2]. Stanowi on otwartą integrację systemu algebry komputerowej z językiem Python, ponadto umożliwia rozpoczęcie zabawy od razu, korzystając z przeglądarki internetowej i jednej z możliwych opcji dostępu poprzez usługę chmurową[3] lub server pojedynczych obliczeń, na którym bazuje interaktywna wersja tego artykułu[4].

1 Chaos w ekologii

W latach siedemdziesiątych XX wieku na Uniwersytecie w Oxford australijski uczonek Robert May zajmował się teoretycznymi aspektami dynamiki populacyjnej. Swoje prace podsumował w artykule, który ukazał się w *Nature* pod prowokującym tytułem “Proste modele matematyczne z bardzo skomplikowaną dynamiką”

[1]. Artykuł ten po latach stał się jedną z najczęściej cytowanych prac z teoretycznej ekologii. Co wzbudziło tak wielkie zainteresowanie w tej pracy?

Klasycznym zadaniem dynamiki populacyjnej jest obliczenie populacji pewnego gatunku w przyszłości znając jego obecny stan. Matematycznie najprostszym rodzajem ekosystemów wydawały się takie, w których życie jednego pokolenia populacji trwa jeden sezon. Dobrym przykładem jest populacja owadów, które w ciągu jednego sezonu przechodzą pełną metamorfozę, np. motyle. Czas jest w naturalny sposób podzielony na dyskretne¹ okresy, odpowiadające cyklom życia populacji. Równania opisujące taki ekosystem mają więc w naturalny sposób tzw. dyskretny czas, tj. $t = 1, 2, 3, \dots$. Robert May zajmował się między innymi właśnie taką dynamiką. W swoich rozważaniach uprościł ekosystem do jednego gatunku, w którym populacja była funkcją kwadratową populacji w roku poprzednim. Skąd taki model?

Najprostszym równaniem dyskretnym opisującym ewolucję populacji jest model liniowy:

$$N_{i+1} = \alpha N_i, \quad (1)$$

gdzie N_i to liczebność w i -tym sezonie, a N_{i+1} opisuje populację w następnym sezonie. Łatwo się przekonać, że takie równanie może prowadzić to trzech scenariuszy. Dla $\alpha = 1$ ewolucja nie będzie zmieniać stanu liczebności populacji, $\alpha < 1$ prowadzi do wyginięcia, a przypadek $\alpha > 1$ oznacza nieograniczony wzrost populacji. To prowadziło do zachwiania równowagi w przyrodzie. Ponieważ wszystko w przyrodzie jest ograniczone, warto poprawić to równanie w taki sposób aby brało pod uwagę ograniczoną ilość zasobów. Wyobraźmy sobie, że szkodniki zjadają zboże, którego jest dokładnie ta sama ilość co roku. Jeżeli owadów jest mało w porównaniu do ilości pożywienia to mogą rozmnażać się z pełną siłą rozrodczą matematycznie określoną przez stałą $\alpha > 1$. Jednak w miarę wzrostu liczebności szkodników, pożywienia nie będzie wystarczać i siła rozrodcza będzie maleć. W krytycznym przypadku można sobie wyobrazić że owadów rodzi się tyle, że zjadają całe zboże przed osiągnięciem zdolności rozrodczej i populacja ginie. Model uwzględniający ten efekt ograniczonego dostępu do pożywienia został po raz pierwszy zaproponowany przez Verhulsta w roku 1838. W modelu tym tempo wzrostu α nie jest stałe,

¹W matematyce “dyskretny” to przyjmujący wartości z określonego przeliczalnego zbioru. Przeciwieństwem jest “ciągły”

ale zależy od stanu populacji:

$$\alpha = \alpha(N_i) \quad (2)$$

Zależność tempa wzrostu α od N_i powinna mieć następującą własność: jeżeli populacja wzrasta, tempo wzrostu powinno maleć ponieważ jest utrudniony dostęp do pożywienia. Oczywiście jest wiele funkcji o tej własności: to są funkcje malejące. Verhulst zaproponował taką oto zależność:

$$\alpha = a \left(1 - \frac{N_i}{K}\right) \quad (3)$$

gdzie $a > 0$ oraz stała $K > 0$ charakteryzuje zasoby pożywienia i nazywa się pojemnością środowiska. Jak zmiana K wpływa na tempo wzrostu populacji? Jeżeli K rośnie to N_i/K maleje. Z kolei to powoduje że $1 - N_i/K$ rośnie, czyli α rośnie. Oznacza to, że tempo wzrostu rośnie i populacja rozrasta się szybciej. Zmodyfikujmy więc poprzedni model (1) zakładając, że tempo wzrostu α zmienia się tak jak w równaniu (3). Wówczas otrzymamy równanie

$$N_{i+1} = a \left(1 - \frac{N_i}{K}\right) N_i \quad (4)$$

Równanie to można zapisać w postaci równania rekurencyjnego

$$x_{i+1} = ax_i(1 - x_i), \quad (5)$$

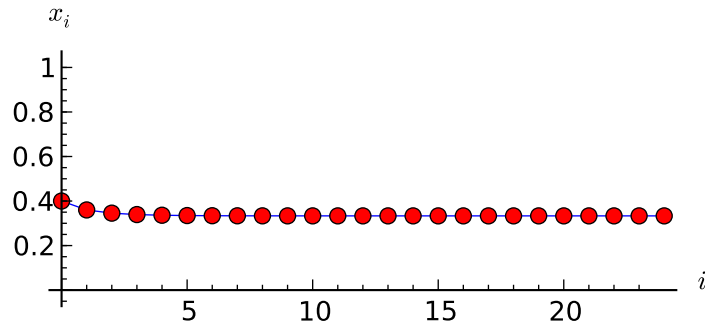
gdzie $x_i = N_i/K$ oraz $x_{i+1} = N_{i+1}/K$ oznaczają przeskalowane liczebności populacji w czasie i oraz w czasie $i+1$. Równanie (5) nazywa się *równaniem logistycznym*.

Mogłoby się wydawać, że przy tak niewielkiej modyfikacji nasz model jest prosty do analizy. Sprawdźmy to. Rozważmy równanie (5) dla parametru $a = 0.5$ startując z początkowej populacji $x_0 = 0.45$. Kolejne wartości populacji można otrzymać stosując równanie rekurencyjne (5):

$$\begin{aligned} x_1 &= ax_0(1 - x_0), \\ x_2 &= ax_1(1 - x_1), \\ x_3 &= ax_2(1 - x_2), \quad \text{itd} \end{aligned} \quad (6)$$

Aby ułatwić obliczenia w (6) możemy skorzystać z poniższego programu² symulującego nasz model:

²Program jest napisany w języku Python i można go uruchomić m.in. na platformie Sage. Zachęcamy do lektury książki <http://icse.us.edu.pl/e-book/>



Rysunek 1: Dla $a = 1.5$ ekosystem osiąga dynamiczną równowagę.

```

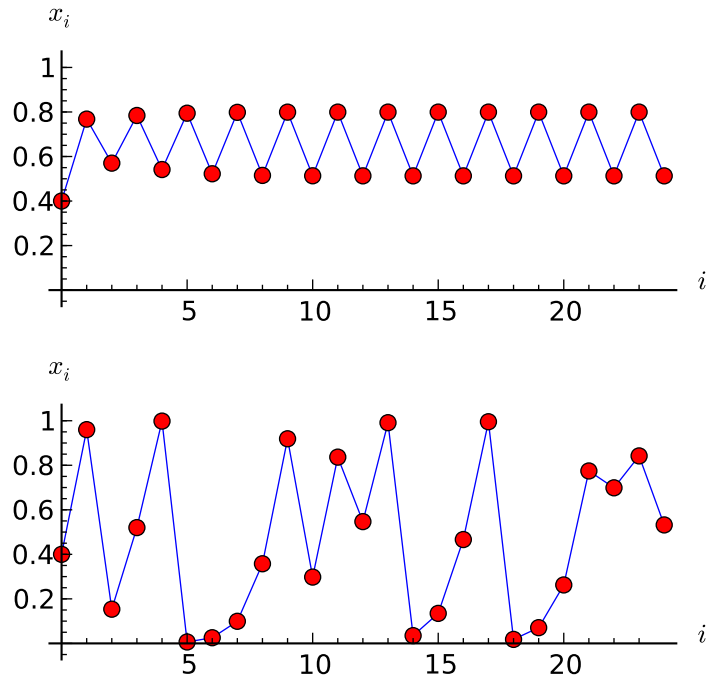
a = 0.5
x = 0.45
for i in range(10):
    x = a*x*(1-x)
    print x

```

Obliczamy kolejne wartości x_i i zauważamy, że maleją one do zera. Eksperymentując z powyższym kodem łatwo też jest się przekonać, że dzieje się tak od wartości początkowej x_0 . Oznacza to, że populacja zawsze ginie.

W drugim kroku analizy zwiększamy wartość parametru a do dowolnej wartości z przedziału $a \in (1, 3)$. Okazuje się, że wtedy ciąg x_i dąży do pewnej wielkości $x^* > 0$. Interpretując to w kategoriach ekologii możemy powiedzieć, że wielkość populacji ustala się na pewnym poziomie, który nie zmienia się z sezonu na sezon. Warto zauważyć, że wartość x^* nie zależy od stanu początkowego x_0 . Jest to efekt dążenia ekosystemu do stabilizacji - populacja dostosowuje swoją liczebność do możliwości wyżywienia się. Matematycznie mówimy, że układ dąży do stabilnego punktu stałego, tj. spełniającego równość $x^* = f(x^*)$ (oznacza to że w chwili następnej stan jest taki sam jak w chwili poprzedniej). Z pomocą programu Sage możemy przedstawić tę ewolucję graficznie rysując wykres liczebności populacji od czasu.

Taki efekt stabilizacji był oczekiwany przez badaczy i równanie logistyczne (5) nie przyciągnęłoby szczególnej uwagi gdyby nie pewna niespodzianka. Okazało się



Rysunek 2: Dla $a = 3.2$ (góra) ewolucja układu prowadzi to oscylacji. W tym przypadku populacja co drugi sezon jest taka sama. Dla $a = 4$ (dół) liczba osobników w kolejnych latach zdaje się być losowa!

bowiem, że dla pewnych wartości parametru a model (5) nie zachowuje się w przewidywalny sposób. Po pierwsze, pojawiają się stany okresowe i wielo-okresowe. Po drugie, z każdym krokiem czasowym populacja zmienia się w nieregularny sposób, podobny do losowego ruchu. Po trzecie, pojawia się duża wrażliwość na warunki początkowe: dwa prawie nierozróżnialne warunki początkowe prowadzą do całkowicie różnej ewolucji populacji. Wszystkie te cechy są charakterystyczne dla zachowania które przypomina ruch całkowicie losowy i nazywa się *deterministycznym chaosem*.

Zbadajmy tę własność!

Na początek ustalmy wartość parametru na $a = 3.2$ i przyjrzyjmy się ewolucji. Zaskoczeniem może być fakt, że tym razem populacja nie osiąga jednej wartości, ale dwie, które występują kolejno po sobie co drugi sezon. Okazało się jednak, że to nie koniec kłopotów. Dla $a = 4$ układ przestaje być przewidywalny. Popatrzmy na rysunek (2) lub wygenerujmy sami ciąg liczb za pomocą komputera. Wyniki wy-

glądają na czysto przypadkowe i dla nieco różniących się populacji początkowych są kompletnie różne. Uważny czytelnik powinien jednak zaprotestować. Jak układ, który jest opisany przez deterministyczne³ równanie, do tego nawet całkiem proste, może mieć nieprzewidywalne zachowanie? Otóż może. Własnością tego układu jest niezwykła czułość na warunki początkowe. Wystarczy wystartować z dwóch warunków początkowych różniących się o jedną milionową i już po kilku krokach otrzymamy całkiem inne wartości populacji. Sprawdźmy to przy pomocy komputera:

```
a=4.0
x=0.123
y=0.123+0.000001
pkts = []
for i in range(25):
    x = a*x*(1-x)
    y = a*y*(1-y)
    print x,y
```

Oto mamy prosty model deterministycznej ewolucji. Ale ten determinizm jest złudny, jest determinizmem tylko matematycznym. Z praktycznego punktu widzenia układ zachowuje się w sposób nieprzewidywalny ponieważ nigdy nie możemy zadać warunków początkowych w matematycznie dokładny sposób. W rzeczywistości wszystko jest określane z pewną dokładnością: każdy przyrząd pomiarowy ma określoną dokładność i to może być przyczyną praktycznej nieprzewidywalności w deterministycznych układach posiadających własność chaotyczności. Przykładem niech będą modele prognozy pogody które zawsze wykazują własność chaotyczności. Dlatego tak kiepskie bywają prognozy pogody w dłuższym przedziale czasowym.

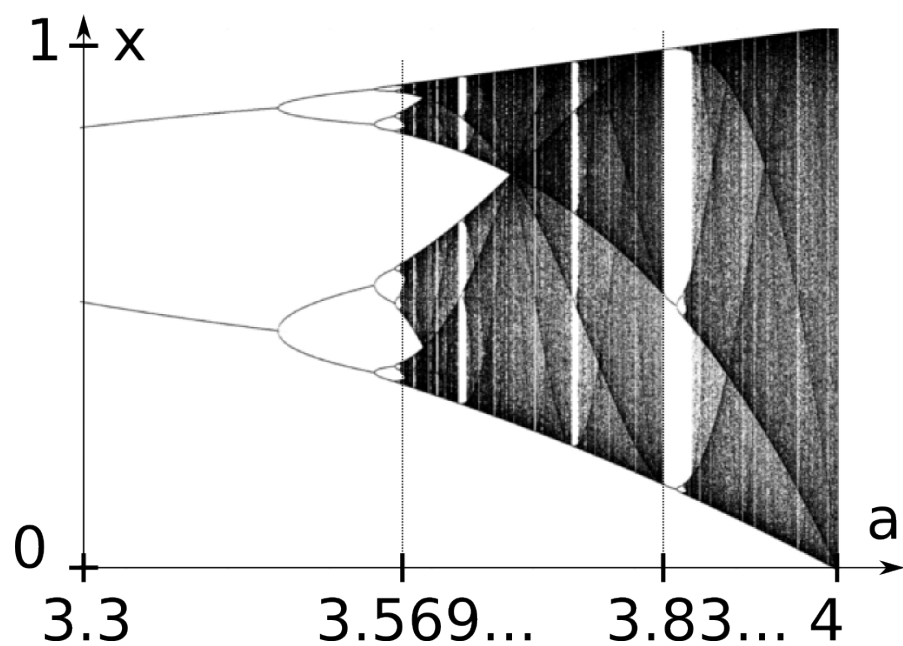
Analiza układów chaotycznych jest niezwykle trudna. Możemy jednak dość łatwo odkryć wiele tajemnic chaotyczności stosując symulacje komputerowe. Narysujmy tak zwany diagram bifurkacyjny, na którym będziemy odkładać na osi odciętych wartości parametru a a na osi rzędnych stabilne punkty stałe odwzorowania logistycznego. Stabilne punkty otrzymamy symulując dużą ilość układów

³Prawo deterministyczne to takie w którym przyszłość jest jednoznacznie określona przez stan początkowy. Antonimem jest prawo probabilistyczne.

jednocześnie i rysując wartości po wieku krokach obliczeń. Jak można się domyślić wymaga to wykonania dużej ilości obliczeń. Spróbujmy “ostrożnie” postępować z następującymi wartościami:

```
import numpy as np
Nx = 300
Na = 500
x = np.linspace(0,1,Nx)
x = x + np.zeros((Na,Nx))
x = np.transpose(x)
a=np.linspace(1,4,Na)
a=a+np.zeros((Nx,Na))
for i in range(100):
    x=a*x*(1-x)
pt = [[a_,x_] for a_,x_ in \
      zip(a.flatten(),x.flatten())]
point(pt,size=1,figsize=(7,5))
```

Powinniśmy otrzymać coś podobnego do rysunku (3). Jak interpretować ten rysunek? Np. dla wartości parametru $a = 3.3$ mamy 2 stabilne punkty stałe (co drugi sezon populacja ma tą samą liczebność). Np. dla parametru $a = 3.5$ mamy 4 punkty stałe (co czwarty sezon populacja ma tą samą liczebność). Np. dla parametru $a = 3.56$ mamy 8 punktów stałych (co ósmy sezon populacja ma tą samą liczebność). Ale dla parametru $a \approx 3.57$ mamy nieskończenie wiele punktów stałych (liczebność w populacji nigdy się nie powtarza i zmienia się w nieprzewidywalny sposób). Mając jednak program komputerowy, możemy zmienić zakres parametru a i eksplorować własnoręcznie niekończącą się strukturę geometryczną owego diagramu. To tylko czubek góry lodowej. Na temat tego równania napisano tysiące prac naukowych i mimo tego wciąż skrywa ono swoje tajemnice. Z pomocą symulacji komputerowej można, nawet bez stosowania zaawansowanej matematyki pobawić się w odkrywcę świata dynamiki nieliniowej. Zapraszamy do lektury wersji online, zawierającej szczegółowy wielu ciekawych własności równania logistycznego oraz interesujące sposoby ich wizualizacji.



Rysunek 3: Diagram bifurkacyjny dla równania logistycznego

Literatura

- [1] May, R. M. "Simple mathematical models with very complicated dynamics".
Nature 261 (5560): 459–467,1976.
- [2] <http://sagemath.org>
- [3] <http://cloud.sagemath.com>
- [4] <http://visual.icse.us.edu.pl/Warsztaty>