Zbiór zadań z modelowania matematycznego w środowisku Wolfram Mathematica [1]*.*

Konrad Kisiel & Grzegorz Siudem

20 lutego 2023

Przedstawiamy zbiór zadań i problemów związanych z dynamiką modeli matematycznych. Wersja ciągle robocza. Będziemy wdzięczni za wszelkie uwagi.

[1] Złożono w klasie `tufte-handout`, `www.ctan.org/pkg/tufte-latex`

Spis treści

Materiały wprowadzające

Poniżej zamieszczamy garść najważniejszych definicji stosowanych podczas zajęć, krótki opis środowiska Wolfram Mathematica, a na zakończenie, opisujemy formalne wymagania, jakie będziemy stawiać wynikom prac domowych.

Zwracamy uwagę, że każda pojawiająca się w tekście nazwa mathematicznej funkcji (np. `D[]`) jest hiperłączem, po kliknięciu w które można przeczytać dokumentację tej funkcji na stronie reference.wolfram.com/language/

Zalecamy przeczytać cały rozdział przed wykonywaniem zadań.

Definicje i oznaczenia

Niech $f\colon X \to X$ będzie pewną funkcją, a X (w zasadzie dowolnym) zbiorem. Mówiąc o dynamice zadanej przez funkcję f mamy na myśli iteracje tej funkcji, czyli jej wielokrotne złożenia

$$f^n(x) = \underbrace{f \circ f \circ \cdots \circ f}_{n \text{ razy}}(x) = \underbrace{f(f(f(\ldots f}_{n \text{ razy}}(x)\ldots))).$$

Aby lepiej opisywać takie złożenia stosujemy wyjaśnijmy poniższe terminy.

W Mathematice złożenie funkcji otrzymamy dzięki funkcji `Nest[]`.

Definicja 1 **Orbitą** *funkcji f w punkcie $x \in X$ nazywamy zbiór*

$$\mathcal{O}(f, x) = \{x, f(x), f^2(x), f^3(x), \ldots\}.$$

Gdy nie prowadzi to do nieporozumień orbitą będziemy także nazywali skończone podzbiory zbioru $\mathcal{O}(f, x)$ postaci

$$\{x, f(x), f^2(x), f^3(x), \ldots, f^N(x)\}.$$

Dzięki temu będziemy mogli pisać o tym, że funkcja `NestList[]` generuje orbity.

Definicja 2 *Orbitę $\mathcal{O}(f, x)$ nazywamy* **okresową** *o okresie k jeśli*

$$\forall_{n\in\mathbb{N}} \quad f^{k+n}(x) = f^n(x).$$

Definicja 3 *Gdy k w poprzedniej definicji przyjmuje wartość $k = 1$, wówczas punkt x nazywamy* **punktem stałym**, *gdyż*

$$f(x) = x.$$

W poszukiwaniu punktów stałych pomoże funkcja `FixedPoint[]`.

Definicja 4 *Punkt $x_p \in X$ nazywamy* **punktem preokresowym** *jeśli $\mathcal{O}(f, f(x_p))$ jest orbitą okresową.*

W kolejnych definicjach zakładamy, że $(X, \mathcal{B}, \mu)$ jest przestrzenią z miarą.

Definicja 5 *Mówimy, że* **przekształcenie** $T : X \to X$ **zachowuje miarę** *μ gdy*

$$\forall B \in \mathcal{B}, \qquad \mu\left(T^{-1}(B)\right) = \mu(B).$$

Definicja 6 *Przekształcenie T nazwiemy* **ergodycznym** *jeśli zachowuje miarę i ponadto spełnia warunek*

$$\forall B \in \mathcal{B}, \qquad T^{-1}(B) = B \implies (\mu(B) = 0 \quad \vee \quad \mu(X \setminus B) = 0).$$

Twierdzenie 1 (Twierdzenie Birkhoffa) *Niech $T : X \to X$ będzie przekształceniem ergodycznym na przestrzeni $(X, \mathcal{B}, \mu)$. Załóżmy dodatkowo, że $\mu(X) < \infty$. Wówczas dla dowolnej funkcji $f \in L^1(\mu)$ zachodzi zbieżność:*

$$\frac{1}{n}\sum_{k=0}^{n-1} f\left(T^{(k)}(x_0)\right) \to \frac{1}{\mu(X)}\int_X f(x)\mathrm{d}\mu(x), \quad \text{dla} \quad \mu\text{-p.w.} \quad x_0 \in X.$$

Definicje i twierdzenia związane z teorią procesów stochastycznych przytaczamy za podręcznikiem Iwanika i Misiewicz [1].

Definicja 7 **Procesem stochastycznym** $\{X_t,\ t \in \mathbb{T}\}$ *nazywamy taką funkcję $(t, \omega) \to X_t(\omega)$, $\omega \in \Omega$, $t \in \mathbb{T}$, że $X_t(\omega)$ jest zmienną losową dla każdego ustalonego $t \in \mathbb{T}$.*

Definicja 8 *Zbiór wszystkich możliwych wartości procesu stochastycznego $\{X_t,\ t \in \mathbb{T}\}$ $\mathcal{S} = \{X_t(\omega) : t \in \mathbb{T}, \omega \in \Omega\}$ nazywamy* **przestrzenią stanów** *procesu lub jego* **przestrzenią fazową**.

Będziemy rozważali co najwyżej przeliczalne przestrzenie stanów.

Definicja 9 (Warunek Markowa) *Proces stochastyczny $\{X_n,\ n \in \mathbb{N}\}$ o wartościach w $\mathcal{S}_m = \{1, ..., m\}$, $m \in \mathbb{N}$ nazywamy* **łańcuchem Markowa** *lub* **procesem Markowa**, *jeśli dla dowolnego $n \in \mathbb{N}$ i dowolnego wyboru stanów $k_0, k_1, ..., k_n \in \mathcal{S}_m$ zachodzi poniższa własność*

$$\mathbb{P}(X_n = k_n | X_{n-1} = k_{n-1}, ..., X_0 = k_0) = \mathbb{P}(X_n = k_n | X_{n-1} = k_{n-1}).$$

Równanie to bywa także nazywane warunkiem Markowa lub warunkiem braku pamięci.

Definicja 10 *Łańcuch Markowa $\{X_n,\ n \in \mathbb{N}\}$ jest* **jednorodny** *lub równoważnie* **ma stacjonarne prawdopodobieństwa przejścia**, *jeśli dla dowolnych stanów $i, j \in \mathcal{S}_m$, $m \in \mathbb{N}$ i dowolnego $n \in \mathbb{N}$ zachodzi równość*

$$\mathbb{P}(X_n = j | X_{n-1} = i) = \mathbb{P}(X_1 = j | X_0 = i).$$

W dalszej części rozważane będą jedynie łańcuchy Markowa o stacjonarnym prawdopodobieństwie przejścia.

Definicja 11 *Prawdopodobieństwa $p_{ji} := \mathbb{P}(X_1 = j | X_0 = i)$, dla $i, j \in \mathcal{S}_m$, $m \in \mathbb{N}$, nazywamy* **prawdopodobieństwami przejścia w jednym kroku**, *a utworzoną z nich macierz*

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1m} \\ p_{21} & p_{22} & p_{23} & \cdots & p_{2m} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ p_{m1} & p_{m2} & p_{m3} & \cdots & p_{mm} \end{bmatrix},$$

macierzą przejścia w jednym kroku.

Uwaga 1 *Przyjęto tutaj notację inną od zawartej w cytowanych podręcznikach do procesów stochastycznych (m. in. Iwanik i Misiewicz [1]), gdzie autorzy zapisywali prawdopodobieństwo przejścia w jednym kroku jako* $\tilde{p}_{ij} := \mathbb{P}\left(X_1 = j | X_0 = i\right)$, *co powodowało, że w ich ujęciu macierz przejścia w jednym kroku była transpozycją macierzy, według defiinicji 11 -* $\tilde{P}^T = P$.

Uwaga 2 *Ujęcie teorii procesów Markowa w formalizm teorii układów dynamicznych często okazuje się pomocne nie tylko pojęciowo, ale też ze względów rachunkowych (porównaj rozdział trzeci i piąty w podręczniku Lasoty i Mackeya [2]). Wówczas macierz przejścia P nazywa się operatorem Markowa. Znane z teorii układów dynamicznych pojęcie asymptotycznej stabilności wiąże się wówczas ściśle z ergodycznością łańcuchów Markowa. W dalszej części termin macierz przejścia P będzie stosowany zamiennie z terminem operator Markowa P.*

Samo pojęcie można zdefiiniować też abstrakcyjnie, ale w rozważanym przypadku wystarczy utożsamiać operator Markowa z operatorem generującym proces Markowa, czyli z macierzą przejścia w jednym kroku.

Z aplikacyjnego punktu widzenia bardzo ważne jest pytanie, jak działa proces Markowa, gdy iterowany jest wielokrotnie[2]. Odpowiedzi na pytanie to udziela poniższe twierdzenie.

[2] To właśnie te wielokrotne iteracje są przyczyną, dla której formalizm teorii układów dynamicznych jest tu z powodzeniem stosowany.

Twierdzenie 2 *Jeżeli proces Markowa P jest iterowany n-krotnie to odpowiada mu wtedy proces Markowa o macierzy przejścia w jednym kroku równej* $Q = P^n$, *przy czym* $q_{ij}(n) = [P^n]_{ij}$, *jest prawdopodobieństwem przejścia w jednym kroku dla procesu będącego n-krotną iteracją procesu wyjściowego.*

Definicja 12 *Macierz* $P \in \mathbb{M}^{m\times m}\left([0,1]\right)$, $m \in \mathbb{N}$, $P = [p_{ij}]_{i,j=1,\dots,m}$ *spełniającą poniższe założenia*

- $p_{ij} \geqslant 0$ *dla dowolnych* i, $j \in \mathcal{S}_m$,
- $\sum\limits_{i=1}^{m} p_{ij} = 1$,

nazywamy **macierzą stochastyczną**.

Widać, że przyjęta defiinicja 11 macierzy P powoduje, że macierzą stochastyczną nazywamy macierz stochastyczną kolumnowo (tzn. w kolumnach sumującą się do 1). Macierze stochastyczne wierszowo są właściwe przy transponowanej defiinicji P (porównaj uwaga 1). Macierz będącą jednocześnie stochastyczną kolumnowo i wierszowo nazywamy **macierzą podwójnie stochastyczną**.

Drugą, po macierzy przejścia w jednym kroku, wygodną metodą reprezentacji procesu Markowa są diagramy przejścia w jednym kroku.

Definicja 13 **Diagramem przejścia w jednym kroku** *procesu Markowa* $\{X_n,\ n \in \mathbb{N}\}$ *nazywamy graf skierowany, w którym wierzchołkami są elementy przestrzeni fazowej, a łuki łączą dwa stany, o ile prawdopodobieństwo przejścia między nimi w jednym kroku jest niezerowe.*

Każdy ze stanów należący do przestrzeni fazowej $\mathcal{S}_m$ dla procesu Markowa może zostać scharakteryzowany ze względu na możliwości przejścia między nim a innymi stanami. Poniżej zaprezentowano stosowaną później charakteryzację stanów.

Definicja 14 *Stan* $i \in \mathcal{S}_m$ *jest* **pochłaniający** *jeśli* $p_{ii} = 1$.

Intuicyjnie stan pochłaniający to taki stan, z którego nie można wyjść.

Definicja 15 *Stan* $i \in \mathcal{S}_m$ *jest* **osiągalny** *ze stanu* $j \in \mathcal{S}_m$ *jeśli istnieje takie* $n \in \mathbb{N}$, *że* $p_{ij}(n) > 0$. *Będzie to oznaczane przez* $j \to i$.

Jeden stan jest dla drugiego osiągalny, o ile można w toku iteracji przejść z jednego do drugiego.

Definicja 16 *Stany* $i, j \in \mathcal{S}_m$ **wzajemnie się komunikują** *jeśli* $j \to i$ *oraz* $i \to j$.

Komunikowanie się jest relacją równoważności (symetryczną, zwrotną i przechodnią), dzieli więc wszystkie stany procesu Markowa na klasy abstrakcji - klasy stanów komunikujące się między sobą.

Definicja 17 *Łańcuch Markowa nazywamy* **nieprzywiedlnym**, *gdy wszystkie jego stany wzajemnie się komunikują.*

Nieprzywiedlność oznacza, że wszystkie stany należą do tej samej klasy abstrakcji względem relacji komunikowania się.

Definicja 18 **Okresem** *stanu* $i \in \mathcal{S}_m$ *nazywamy liczbę*

$$o(i) = NWD(n \in \mathbb{N} : p_{ii}(n) > 0),$$

czyli największy wspólny dzielnik zbioru takich n, *że powrót do stanu* i *jest możliwy po* n *krokach.*

Stan i *jest* **okresowy** *jeśli* $o(i) > 1$ *i* **nieokresowy** *gdy* $o(i) = 1$.

Definicja 19 *Nieprzywiedlny łańcuch Markowa jest* **okresowy** *jeśli wszystkie jego stany są okresowe o tym samym okresie* $d > 1$. *W przeciwnym przypadku mówimy, że łańcuch jest* **nieokresowy**.

Defiinicja 19 jest dobrze określona, co wynika z poniższego twierdzenia, przytoczonego za podręcznikiem Iwanika i Misiewicz [1].

Twierdzenie 3 *W nieprzywiedlnym łańcuchu Markowa wszystkie stany mają ten sam okres.*

Definicja 20 *Rozkład początkowy* $\pi_i = \mathbb{P}(X_0 = i)$, $i \in \mathcal{S}_m$ *łańcucha Markowa* $\{X_n, n \in \mathbb{N}\}$ *jest* **stacjonarny** *lub* **niezmienniczy** *jeśli zachodzi równanie*

$$\pi = P\pi.$$

Rozkład stacjonarny jest więc punktem stałym operatora P.

Definicja 21 *Jednorodny łańcuch Markowa jest* $\{X_n,\ n \in \mathbb{N}\}$ **ergodyczny**, *jeśli dla każdego* $i \in \mathcal{S}_m$ *istnieją i nie zależą od* j *następujące granice*

$$q_i = \lim_{n\to\infty} p_{ij}(n) > 0,$$

oraz $\sum_{i=1}^{m} q_i = 1$. *Otrzymany rozkład* $\mathbf{q} = (q_i)_{i=1,\dots,m}$ *nazywamy* **ergodycznym**.

Ergodyczność w języku operatorów Markowa (porównaj rozdział 5.6 w podręczniku Lasoty i Mackeya [2]) jest równoważna istnieniu asymptotycznie stabilnego rozkładu prawdopodobieństwa dla operatora (macierzy) P.

Twierdzenie 4 *Każdy rozkład ergodyczny dla pewnego łańcucha Markowa jest też rozkładem stacjonarnym tego łańcucha.*

Uwaga 3 *Implikacja z twierdzenia 4 w ogólności nie daje się odwrócić. Twierdzenie odwrotne nie musi być prawdziwe. Łańcuch Markowa może posiadać więcej niż jeden rozkład stacjonarny (porównaj twierdzenie 5), a, zgodnie z definicją 21, rozkład ergodyczny wyznaczony jest jednoznacznie. Na podstawie twierdzenia 5 każdy łańcuch Markowa posiada co najmniej jeden rozkład stacjonarny, ale co do istnienia rozkładu ergodycznego dla dowolnego procesu nie ma gwarancji. (porównaj twierdzenie 6).*

Twierdzenie 5 *Dla każdego procesu Markowa o skończonej liczbie stanów istnieje co najmniej jeden rozkład stacjonarny. Wszystkie rozkłady stacjonarne (jako wektory z przestrzeni* $\mathbb{R}^m$*) należą do podprzestrzeni rozpiętej przez wektory własne macierzy* P *odpowiadające wartości własnej* 1.

Jest to tak naprawdę cecha każdej macierzy stochastycznej, dla której 1 jest na pewno wartością własną.

Twierdzenie 6 *Rozważmy nieokresowy i nieprzywiedlny proces Markowa o skończonej liczbie stanów. Wówczas łańcuch ten posiada dokładnie jeden rozkład stacjonarny* π. *Co więcej rozkład* π *jest też rozkładem ergodycznym tego procesu.*

Jeśli stanów byłoby przeliczalnie wiele wówczas należy jeszcze założyć, że każdy stan jest dodatnio powracający, czyli, że prawdopodobieństwo powrotu do każdego stanu w skończonym czasie jest większe niż 0.

Mathematica

Podstawową jednostką składni w środowisku Mathematica jest komórka. Komórki mogą przyjmować kilka typów, ale dwa najważniejsze z nich to `In[]`, czyli wprowadzany przez użtkownika skrypt oraz `Out[]`, czyli wynik działania `In[]`. Aby wykonać komórkę, w której aktualnie znajduje się kursor należy wcisnąć `[Shift]+[Enter]` lub prawy `[Enter]`.

Komórki wyróżnione są wewnątrz notatnika ramką po prawej stronie.

Zachęcamy do sprawdzenia opisywanej składni już teraz. Można to zrobić nawet bez dostępu do wydziałowych licencji, używając aplikacji Wolfram Programming Lab.

Nazwy wbudowanych funkcji w języku Wolfram zawsze zaczynają się wielką literą według wzoru

```
NazwaFunkcji[arg1, arg2, ...]
```

Zauważmy, że te nazwy dość dokładnie precyzują co dana funkcja robi, np. `Series[]` rozwija w szereg, a `Solve[]` rozwiązuje równania. Podstawowa znajomość języka angielskiego pozwala zatem na dość dokładne przewidywanie nazwy funkcji, której funkcjonalność chcielibyśmy uzyskać.

gdzie argumenty funkcji przekazywane są w kwadratowych nawiasach `[]`. Najważniejszym sposobem poznawania funkcjonalności Mathematiki jest jej bogata pomoc. Jest to także podstawowe narzędzie pracy w tym środowisku. Pomoc wywołuje się klawiszem `[F1]` wewnątrz aktywnego notatnika, lub, nawet bez aktywnej aplikacji, można sprawdzić jej internetową wersję.

Dostępną, przypomnijmy, pod adresem reference.wolfram.com/language/

Jako ćwiczenie wprowadzające do środowiska Wolfram Mathematica proponujemy sprawdzić w dokumentacji do czego mogą służyć następujące funkcje

Zwracamy uwagę na bogactwo przykładów dostępnych w plikach pomocy.

- `Plot[]`
- `Integrate[]`,
- `Animate[]`,
- `Manipulate[]`,
- `Table[]`,
- `Graphics[]`,
- `D[]`,
- `ListPlot[]`,
- `Simplify[]`.

Jak można zauważyć, podczas analizy powyższych przykładów, podstawową strukturą danych w Mathematice jest lista. Lista to nic innego jak pewien uporządkowany zbiór pewnych (niekoniecznie tego samego typu!) elementów. Poniżej zamieszczamy przykłady list

```
{a, b, c, 1, 2, 3}

{{1,0},{0,1}}

Table[1/i,{i,10}]
```

Jaki wyniki dają te kody źródłowe?

Lista list jest sposobem zapisu macierzy w środowisku Mathematica - po więcej szczegółów odsyłamy do dokumentacji (zwłaszcza `Symbolic & Numeric Computation`→`Matrices and Linear Algebra`).

Funkcja `Table[]` jest jedną z częściej używanych, podczas pracy z Mathematiką. Sugerujemy przejrzeć jej dokumentację.

Poniższe kody źródłowe są przykładami wykorzystania środowiska Mathematica do tworzenia grafiki. Sugerujemy je skopiować do notatnika i sprawdzić co daje ich wykonanie. Przejrzenie dokumentacji wykorzystanych funkcji również będzie pouczające.

```
Graphics[Table[Circle[RandomReal[{-10, 10}, {2}]], {80}]]

Graphics[{Blue, FilledCurve[Line[RandomReal[{0, 10}, {10,
2}]]], Red, FilledCurve[Line[RandomReal[{0, 10}, {20, 2}]]],
Green, FilledCurve[Line[RandomReal[{0, 10}, {100, 2}]]]}]
```

Polecamy także wypróbować działania funkcji `Disk[]`, `Polygon[]`, a także możliwości manipulowania obiektami graficznymi (kolory, grubości linii etc.), przykłady Czytelnik znajdzie oczywiście w dokumentacji `Graphics[]`.

Wymagania formalne

Prawidłowo przygotowana praca domowa powinna spełniać następujące wymagania

- Po pierwsze **musi być zgodna z** obowiązującym, umieszczonym w zasobach przedmiotu, **szablonem odpowiedzi**. W szczególności, nie może zajmować więcej niż jednej kartki A4.
- Kody źródłowe powinny być czytelne i poprawne.
- Wszystkie liczby i cyfry występujące na wykresach i rysunkach powinny być w rozmiarach zbliżonych do fontu tekstu oraz kroju, umożliwiającym ich wygodne odczytanie.
- Wszystkie krzywe (w tym osie współrzędnych i wykresy) występujące na rysunkach powinny być pogrubione tak, aby być czytelne i odróżniać się od reszty rysunku.
- Osie wykresów powinny być podpisane.

Obroty

Eppur si muove — Galileo Galilei

Celem ćwiczeń będzie zbadanie podstawowych dynamicznych własności obrotu na okręgu ($\mathcal{S}^1 = [0, 2\pi]/(0 = 2\pi)$), czyli funkcji

$$f_\theta : \mathcal{S}^1 \to \mathcal{S}^1, \quad f_\theta(x) = (x + \theta) \mod 2\pi.$$

Na przykładzie tych badań poznamy też podstawową funkcjonalność środowiska Wolfram Mathematica.

Uwaga: Przed wykonywaniem zadań sugerowane jest przeczytanie pierwszego rozdziału niniejszego skryptu.

Plan badań

1. Implementacja funkcji f_θ (definicja funkcji, definicja zmiennej). — `:=` vs. `=`
2. Rysowanie wykresu funkcji f_θ. — `Plot[]`
3. Rysowanie wykresu orbity f_θ (naiwne). — `ListPlot[]`
4. Łączenie wykresów. — `GraphicsGrid[]`, `Show[]`
5. Dyskusja zależności $f_\theta(x)$ od θ i x. — `Manipulate[]`
6. Iterowanie funkcji f_θ. — `Nest[]`
7. Generowanie orbity f_θ. — `NestList[]` vs. `Table[]`
8. Analiza orbity funkcji f_θ i dyskusja jej zależności od θ. — `Histogram[]` vs. `HistogramList[]`

Pytania i problemy

♣$_1$ Wyjaśnij jak działa i, rysując wykres funkcji sinus, podaj przykład użycia funkcji `Plot[]`.

♣$_2$ Wyjaśnij jak działa i, rysując wykres listy `{3,1,4,1,5,9}`, podaj przykład użycia funkcji `ListPlot[]`.

♣$_3$ Wyjaśnij jak działa i, generując iteracje funkcji $h(x) = 2x$, podaj przykład użycia funkcji `Nest[]`.

♣$_4$ Wyjaśnij jak działa i, rysując wykres listy pierwszych 10 liczb pierwszych, podaj przykład użycia funkcji `ListPlot[]`.

♣$_5$ Wyjaśnij jak działa i, rysując wykres funkcji kosinus, podaj przykład użycia funkcji `Plot[]`.

♣$_6$ Wyjaśnij jak działa i, generując iteracje funkcji $h(x) = x/3$, podaj przykład użycia funkcji `Nest[]`.

♣$_7$ Jak w środowisku Mathematica definiuje się funkcje? Zdefiniuj własną funkcję $\mathrm{sgn}(x) := |x|/x$ dla $x \neq 0$.

$\clubsuit_8$ Wyjaśnij jak działa i, generując listę kolejnych iteracji funkcji sinus, podaj przykład użycia funkcji `NestList[]`.

$\clubsuit_9$ Wyjaśnij jak działa i, wyznaczając histogram listy liczb pseudolosowych, podaj przykład użycia funkcji `Histogram[]`.

Potrzebną listę najprościej wygenerować komendą `li=RandomReal[{0,1},{100}]`

$\clubsuit_{10}$ Wyjaśnij jak działa i, generując listę kolejnych iteracji funkcji $h(x) = x/10$, podaj przykład użycia funkcji `NestList[]`.

$\clubsuit_{11}$ Jak w środowisku Mathematica definiuje się funkcje? Zdefiniuj własną funkcję $h(x) = |x|$.

$\clubsuit_{12}$ Jak w środowisku Mathematica definiuje się zmienne? Podaj przykłady co najmniej 5 typów zmiennych.

$\clubsuit_{13}$ Jak w środowisku Mathematica stworzyć listę? Jak zmienić dokładnie jeden jej element?

$\clubsuit_{14}$ Wyjaśnij jak działa i, generując macierz jednostkową 5×5, podaj przykład użycia funkcji `Table[]`.

$\clubsuit_{15}$ Wyjaśnij jak działa i, generując listę ze współrzędnymi wybranych punktów wykresu funkcji sinus na zbiorze $[0, 2\pi]$, podaj przykład użycia funkcji `Table[]`.

Zwróćmy uwagę, że jest to lista par postaci `{{a,Sin[a]},{b,Sin[b]},...}`

$\clubsuit_{16}$ Jak dodać więcej niz jeden wykres na `Plot[]`? Jako przykład rozważ funkcje sinus i kosinus.

$\clubsuit_{17}$ Jak dodać więcej niz jeden wykres na `ListPlot[]`? Podaj przykład z listami `{1,2,3,4}` i `{{4,5},{3,2},{2,1},{1,0}}`.

$\clubsuit_{18}$ Jak dodać kropki na wykresie typu `Plot[]`?

Sugrowanym przykładem mogą być punkty `{{0.1,0.15},{0.2,0.19},{0.3,0.32},{0.4,0.39},{0.5,0.51},{0.6,0.59}}` i przybliżająca je funkcja $h(x) = x$.

$\clubsuit_{19}$ Powtórz poprzednie zadanie, ale tym razem dodając linię ciągłą na wykresie typu `ListPlot[]`.

$\clubsuit_{20}$ Wyjaśnij jak działa funkcja `Manipulate[]`. Zilustruj jej działanie przykładem.

$\clubsuit_{21}$ Jak pogrubić krzywą wykresu typu `Plot[]`?

W zadaniach $\clubsuit_{21} - \clubsuit_{23}$ sugerujemy uważnie przeczytać rozdział `Options` w dokumentacji funkcji `Plot[]`.

$\clubsuit_{22}$ Jak zmienić kolor krzywej wykresu typu `Plot[]`?

$\clubsuit_{23}$ Jak zmienić styl kreskowania krzywej wykresu typu `Plot[]`?

$\clubsuit_{24}$ Jak zmienić rozmiar punktów wykresu typu `ListPlot[]`?

W zadaniach $\clubsuit_{24} - \clubsuit_{26}$ sugerujemy uważnie przeczytać rozdział `Options` w dokumentacji funkcji `ListPlot[]`.

$\clubsuit_{25}$ Jak zmienić kolor punktów wykresu typu `ListPlot[]`?

$\clubsuit_{26}$ Jak zmienić kształt punktów wykresu typu `ListPlot[]`?

W zadaniach $\clubsuit_{27} - \clubsuit_{30}$ sugerujemy uważnie przeczytać rozdział `Options` w dokumentacji funkcji `ListPlot[]` `i/lub` `Plot[]`.

$\clubsuit_{27}$ Jak dodać do wykresu (dowolnego typu) linie siatki?

♣28 Jak zmienić na wykresie (dowolnego typu) zakres na osi odciętych i rzędnych?

♣29 Jak na wykresie (dowolnego typu) podpisać osie?

♣30 Jak dodać do wykresu (dowolnego typu) linie siatki?

Uwaga: Pamiętaj aby w rozwiązaniach ♠ zastosować opcje zwiększające czytelność (a niekiedy nawet nadające sens) wykresów. Są to opcje takie jak: `PlotRange`, `AxesOrigin`, `PlotStyle`, `PlotLegends` itp.

♠1 Narysuj wykres pierwszych 10 elementów orbity funkcji $h(x) = x^3$ w $x_0 = 0.5$.

♠2 Wykorzystując warunkową instrukcję `If[]` napisz własną implementację funkcji `Mod[]`.

♠3 Narysuj wykres pierwszych 10 elementów orbity funkcji $h(x) = \sqrt{x}$ w $x_0 = 0.5$.

♠4 Narysuj histogram pierwszych 100 elementów orbity funkcji $h(x) = \sqrt{x}$ w $x_0 = 0.5$.

Dobierz odpowiednie rozmiary przedziałów (binów).

♠5 Narysuj wykresy trzech pierwszych iteracji funkcji sinus na zbiorze $[0, 2\pi]$.

♠6 Narysuj histogram pierwszych 100 elementów orbity funkcji $h(x) = x^3$ w $x_0 = 0.5$.

Dobierz odpowiednie rozmiary przedziałów (binów).

♠7 Narysuj wykresy trzech pierwszych iteracji funkcji kosinus na zbiorze $[0, 2\pi]$.

♠8 Narysuj wykres pierwszych 4 elementów orbity funkcji $h(x) = e^x$ w $x_0 = 0.05$.

Żeby rysunek był czytelny zamiast funkcji `ListPlot[]` wykorzystaj `ListLogPlot[]`.

♠9 Narysuj histogram pierwszych 100 elementów orbity funkcji $h(x) = \sin x$ w $x_0 = \pi/2$.

Dobierz odpowiednie rozmiary przedziałów (binów).

♠10 Narysuj wykres drugiej iteracji funkcji wykładniczej na $[0, 1]$.

Zamiast funkcji `Plot[]` wykorzystaj `LogPlot[]`.

♠11 Narysuj wykres pierwszych 50 elementów orbity funkcji $h(x) = 3x(1-x)$ w $x_0 = 0.27$.

♠12 Narysuj wykres czwartej iteracji funkcji $h(x) = 3x(1-x)$ na zbiorze $[0, 1]$.

♠13 Narysuj wykres pierwszych 10 elementów orbity funkcji $h(x) = \sin x$ w $x_0 = \pi/2$.

♠14 Narysuj wykres trzeciej iteracji funkcji $h(x) = \sin(e^x)$ na $[0, 1]$.

♠15 Narysuj histogram pierwszych 100 elementów orbity funkcji $h(x) = 3x(1-x)$ w $x_0 = 0.27$.

Dobierz odpowiednie rozmiary przedziałów (binów).

♠16 Wygeneruj listę wykresów pierwszych 5 iteracji funkcji sinus na zbiorze $[0, 2\pi]$.

♠$_{17}$ Wygeneruj macierz 5×5, o współrzędnych będących sumą numeru wiersza i kolumny.

♠$_{18}$ Narysuj i wyjaśnij jak zależy wykres funkcji $h_C(x) = e^{-Cx^2}$ od parametru $C \in [0,2]$.

Polecamy tutaj funkcję `Manipulate[]`.

♠$_{19}$ Wypisz wszystkie macierze 2×2 o elementach ze zbioru $\{0,1\}$.

♠$_{20}$ Rozwiąż graficznie równanie $x^2 - 2x = 0$.

♠$_{21}$ Jak zależy od wartości parametru C orbita funkcji $h_C(x) = Cx$?

Polecamy tutaj funkcję `Manipulate[]`.

♠$_{22}$ Jak zależy od wartości parametru $C \in [-2,1]$ orbita funkcji $h_C(x) = C\sin(\pi x)$ na zbiorze $x \in [-1,1]$?

Polecamy tutaj funkcję `Manipulate[]`.

♠$_{23}$ Rozwiąż graficznie równanie $e^x + x = 0$.

♠$_{24}$ Rozstrzygnij własną implementacją czy dla wartości początkowych $n = 2, 3, \ldots, 200$ poniższa całkowitoliczbowa funkcja posiada orbitę okresową?

$$h(n) = \begin{cases} \frac{n}{2} & \text{gdy } n \text{ jest parzysta} \\ 3n+1 & \text{gdy } n \text{ jest nieparzysta} \end{cases}.$$

Pojawiająca się w zadaniach ♠$_{24}$ − ♠$_{27}$ funkcja związana jest ze słynnym otwartym problemem Collatza.

♠$_{25}$ Zidentyfikuj punkt stały funkcji z zadania ♠$_{24}$. Rozstrzygnij albo pospekuluj o istnieniu innych punktów stałych.

♠$_{26}$ Znajdź co najmniej dwie orbity funkcji z zadania ♠$_{24}$.

Sugerujemy rozpocząć poszukiwania od $n < 5$.

♠$_{27}$ Narysuj wykres długości orbity funkcji z zadania ♠$_{24}$ zanim osiągnie swoją finalną orbitę okresową w zależności od punktu. początkowego n.

Sugerujemy przyjęcie wartości $n = 1, \ldots, 10$.

♠$_{28}$ Wygeneruj macierz trójkątną, gdzie każdy niezerowy element jest numerem naddiagonali na której lezy.

♠$_{29}$ Wyznacz punkt stały funkcji $f(x) = x/\pi$ dla $x \in \mathbb{R}$.

♠$_{30}$ Narysuj wykres trzeciej iteracji funkcji $h(x) = 3x(1-x)$ na zbiorze $[0,1]$.

Rodzina kwadratowa

A W DODATKU, DAJĘ SŁOWO, MAM RODZINĘ WYJĄTKOWĄ!
J. BRZECHWA

Celem ćwiczeń będzie numeryczne zbadanie dynamiki rozwoju populacji bakterii, modelowane przez rodzinę odwzorowań kwadratowych

$$f_\lambda(x) = \lambda x(1-x), \qquad x \in [0,1]\,,$$

gdzie $\lambda \in [1, 4]$ to parametr.

Plan badań

1. Rodzina kwadratowa jako uproszczony model populacji bakterii.

Ustalmy pojemność środowiska $M > 0$, liczbę $y_0 \in [0, M]$ jako początkową ilość bakterii oraz wielkość $x_0 := y_0/M$.

Dlaczego naiwny model $\hat{f}_\lambda(x) = \lambda x$ jest niefizyczny?

- Niech populacji zmienia się zgodnie z $f_\lambda(x) = \lambda x(1-x)$,
- Dokonaj interpretacji biologicznej rekurencji postaci

$$x_{n+1} = f_\lambda(x_n).$$

2. Dla ustalonych wartości x_0 i λ wygeneruj 30 pierwszych elementów orbity $\mathcal{O}(f_\lambda, x_0)$.

Przy definiowaniu $f_\lambda[x\_] :=$ przydatna może okazać się funkcja `N[]`. Dlaczego?

3. Przedstaw w formie graficznej uzyskaną orbitę.

4. Przy użyciu funkcji `Manipulate[]` przedyskutuj zależność dynamiki od λ i x_0.

Proszę pamiętać też aby uwzględnić możliwość zmiany długości obserwowanej orbity

5. Narysuj kolejne iteracje funkcji $f_\lambda(x)$. Przedyskutuj zależność kolejnych iteracji od parametru λ oraz zinterpretuj uzyskane wyniki.

Pomocne może być naniesienie na badany wykres funkcji $h(x) = x$

6. Diagram bifurkacyjny.

Dla wybranego (dobrze dobranego) punktu startowego x_0 wygeneruj pierwsze n elementów orbity $\mathcal{O}(f_\lambda, x_0)$. Dysponując szeregami czasowymi dla różnych wartości λ nanieś ich „zachowania asymptotyczne" dla kolejnych wartości λ na wykres aby otrzymać poszukiwany diagram bifurkacyjny.

Na diagramie bifurkacyjnym należy umieścić tylko punkty powyżej pewnej liczby iteracji n_0.

Pytania i problemy

$\clubsuit_1$ Jak do wykresu (dowolnego typu) dodać legendę?

$\clubsuit_2$ Opisz co najmniej 3 typy komórek (np. `In`, `Out`, `Text`).

♣3 Wyjaśnij czym różnią się w działaniu np. funkcji `Table[]` liczniki `{i,1,10,1}` i `{i,{1,10,1}}`.

♣4 Czym różnią się przypisania `=` i `:=`? Zilustruj tę różnicę wykorzystując funkcję `RandomInteger[]`.

♣5 Wyjaśnij jak działa funkcja `Show[]`. Podaj przykład jej wykorzystania.

♣6 Wyjaśnij jak działa funkcja `NestWhile[]`. Podaj przykład jej zastosowania.

♣7 Wyjaśnij jak działa funkcja `NestWhileList[]`. Podaj przykład jej zastosowania.

♣8 Wyjaśnij jak działa funkcja `FixedPoint[]` . Podaj przykład jej zastosowania.

♣9 Wyjaśnij jak działa funkcja `Partition[]`. Podaj przykład jej zastosowania.

♣10 Wyjaśnij jak działa funkcja `Riffle[]`. Podaj przykład jej zastosowania.

♣11 Wyjaśnij jak działa funkcja `Join[]`. Podaj przykład jej zastosowania.

♣12 Wyjaśnij jak działa funkcja `Flatten[]`. Podaj przykład jej zastosowania.

♣13 Wyjaśnij jak działa funkcja `Reverse[]`. Podaj przykład jej zastosowania.

♣14 Wyjaśnij jak działa funkcja `Animate[]`. Podaj przykład jej zastosowania.

♣15 Jak działa funkcja `GraphicsGrid[]`? Podaj przykład jej zastosowania.

♣16 Jak działa funkcja `GraphicsRow[]`? Podaj przykład jej zastosowania.

♣17 Jak działa funkcja `GraphicsColumn[]`? Podaj przykład jej zastosowania.

♣18 Wyjaśnij jak działa funkcja `Range[]`. Podaj przykład jej zastosowania.

♣19 Wyjaśnij jak działa funkcja `Directive[]`. Podaj przykład jej zastosowania.

♣$_{20}$ Wyjaśnij na przykładach czym różni się sześć różnych liczników opisanych w dokumentacji funkcji `Manipulate[]`.

♣$_{21}$ Wyjaśnij na przykładach czym różni się sześć różnych liczników opisanych w dokumentacji funkcji `Table[]`.

♣$_{22}$ Wyjaśnij jak działa funkcja `Sort[]`. Podaj przykład jej zastosowania.

♣$_{23}$ Wyjaśnij jak działa, a następnie wykorzystaj w napisaniu własnej funkcji, procedurę `Function[]` w wersji `Function[x,body]`.

♣$_{24}$ Wyjaśnij jak działa, a następnie wykorzystaj w napisaniu własnej funkcji, procedurę `Function[]` w wersji `Function[F[#]]`.

♣$_{25}$ Co w środowisku Mathematica oznacza `#` (`Slot`)?

♣$_{26}$ Wyjaśnij jak działa, a następnie wykorzystaj w napisaniu własnej funkcji, procedurę `F[#]&`.

♣$_{27}$ Jakie znaczenie w środowisku Mathematica mają `#` (`Slot`) i `&` (`Function`)?

♣$_{28}$ Jakie wartości może przyjmować i jak działa (m. in. w `Plot[]` i `ListPlot[]`) dyrektywa `Filling`?

♣$_{29}$ Jakie wartości może przyjmować i jak działa (w `ListPlot[]`) dyrektywa `Joined`?

♣$_{30}$ Jakie wartości może przyjmować i jak działa (w `ListPlot[]`) dyrektywa `Mesh`?

Uwaga: Pamiętaj aby w rozwiązaniach ♠ zastosować opcje zwiększające czytelność (a niekiedy nawet nadające sens) wykresów. Są to opcje takie jak: `PlotRange`, `AxesOrigin`, `PlotStyle`, `PlotLegends` itp.

♠$_{1}$ Analizując wykres orbity wyznacz (empirycznie) wartość λ przy której zachodzi pierwsza bifurkacja w rodzinie kwadratowej.

♠$_{2}$ Analizując wykres orbity wyznacz (empirycznie) wartość λ przy której zachodzi druga bifurkacja w rodzinie kwadratowej.

Do zliczania czasu wykonania funkcji polecamy komendę `AbsoluteTiming[]`.

♠$_{3}$ Porównaj na wykresie, w zależności od $n \in \{1, 2, \ldots, 20\}$, czasy wykonywania komend

```
Table[Nest[λ # (1 - #) &, 2/5, n], {λ, 1, 4, 1/4}]
Table[Nest[λ # (1 - #) &, 2/5, n], {λ, 1, 4, 0.25}]
```

Wyjaśnij różnice.

♠$_{4}$ Analizując wykres orbity wyznacz (empirycznie) wartość λ przy której, dla rodziny kwadratowej, pojawia się pierwsza orbita o okresie 4.

♠5 Analizując $f^4_{3.5}$ Przy pomocy funkcji `Solve[]` wyznacz wartości orbity okresowej o okresach 2 i 4 oraz punkty stałe. Sprawdź wszystkie wyniki bezpośrednio z definicji (wywołując funkcję $f_{3.5}$ odpowiednią liczbę razy).

♠6 Powtórz wykres diagramu bifurkacyjnego z zajęć w lepszej rozdzielczości dla $\lambda \in (1, 3)$. Dopasuj krzywą do obserwowanych punktów.

Można wyliczyć jej wzór, analizując postać funkcji f_λ, odszukać go w literaturze albo dopasować numerycznie np. przy pomocy funkcji `Fit[]` lub `FindFit[]`.

♠7 Wykonując operacje na listach wyznacz wartość λ dla pierwszej bifurkacji podwajania okresu.

♠8 Analizując $f^2_{3.1}$ Przy pomocy funkcji `Solve[]` wyznacz wartości orbity okresowej o okresie 2 oraz punkty stałe. Sprawdź wszystkie wyniki bezpośrednio z definicji (wywołując funkcję $f_{3.1}$ odpowiednią liczbę razy). Narysuj wykres zależności punktów orbity o okresie 2 w funkcji $\lambda \in$ (3, 3.5).

♠9 Wykonując operacje na listach wyznacz wartość λ dla pierwszej bifurkacji podwajania okresu.

♠10 Wychodząc od listy danej funkcją `Range[200]` i stosując operacje na listach, wygeneruj listę liczb mniejszych niż 200 i podzielnych przez 13.

♠11 Odczytaj, zmieniając wartości λ na wykresie $f^3_\lambda(x)$ przy jakiej wartości parametru λ pojawia się orbita o okresie 3.

♠12 Wykonaj animację (względem parametru λ) histogramu orbity funkcji f_λ. W raporcie zamieść kod źródłowy i wybrane, reprezentatywne, klatki tej animacji, nie mniej niż 10.

W zadaniu tym bardzo ważne jest odpowiednie dobranie długości orbity i liczby binów. Zbyt małe zniweczą spodziewany efekt.

♠13 Dodając strzałki do wspólnego wykresu funkcji f_λ i $h(x) = x$ zilustruj pierwsze trzy iteracje metodą graficzną. Zamieść wyniki dla czterech wartości λ generujących odmienną dynamikę.

♠14 Wychodząc od listy danej funkcją `Range[200]` i stosując operacje na listach, wygeneruj listę par postaci `{n,n+1}`, gdzie $n =$ 0, 2, 4, . . . , 200.

♠15 Powtórz wykres diagramu bifurkacyjnego z zajęć w lepszej rozdzielczości dla $\lambda \in (3, 3.7)$. Odczytaj z wykresu możliwie najdokładniej wartości λ odpowiadające trzem pierwszym bifurkacjom podwojenia okresu.

♠16 Analizując wykres orbity wyznacz (empirycznie) wartość λ przy której, dla rodziny kwadratowej, pojawia się pierwsza orbita o okresie 8.

♠17 Odczytaj z diagramu bifurkacyjnego wartości 4 kolejnych bifurkacji, a następnie narysuj histogramy orbity przed i po każdej bifurkacji.

Jeśli λ_b to punkt bifurkacji to przed oznacza $\lambda_b - \varepsilon$, a po $\lambda_b + \varepsilon$ np. dla $\varepsilon = 0.001$.

♠18 Korzystając z licznika postaci `{λ,3,4,0.01}` wygeneruj listę orbit funkcji f_λ, następnie usuń dublujące się elementy i narysuj wykres liczności orbity w funkcji λ.

W zadaniu ♠16 należy odnosić się do kolejnych elementów orbity (co najmniej 200), które występują powyżej pewnej dostatecznie dużej liczby iteracji n_0.

♠19 Wykonując operacje na listach wyznacz wartość λ dla drugiej bifurkacji podwajania okresu.

♠20 Wychodząc od listy danej funkcją `Range[200]` i stosując operacje na listach, wygeneruj listę liczb mniejszych niż 200 i podzielnych przez 7.

♠21 Stosując funkcję `HistogramList` do listy służącej podczas ćwiczeń do narysowania diagramu bifurkacyjnego narysuj funkcją `ArrayPlot` wykres, w którym na osi poziomej będą wartości λ, na osi pionowej wartosci $y \in [0,1]$, a intensywność koloru każdego piksela wykresu będzie zależeć od częstości występowania danych wartości y w orbicie funkcji f_λ.

♠22 Na diagramie bifurkacyjnym z zajęć nanieś strzałki wskazujące miejsca pierwszych trzech bifurkacji.

♠23 Dodając strzałki do wspólnego wykresu funkcji f_λ i $h(x) = x$ zilustruj pierwsze trzy iteracje metodą graficzną. Zamieść wyniki dla czterech wartości λ generujących odmienną dynamikę.

♠24 Powtórz wykres diagramu bifurkacyjnego z zajęć w lepszej rozdzielczości dla $\lambda \in (3.7, 4)$.

♠25 Usuń dublujące się punkty z diagramu bifurkacyjnego z zajęć.

Pamiętaj, że operujesz na liczbach zmiennoprzecinkowych, a nie rzeczywistych!

♠26 Wykonując operacje na listach wyznacz wartość λ dla trzeciej bifurkacji podwajania okresu.

♠27 Wychodząc od listy danej funkcją `Range[200]` i stosując operacje na listach, wygeneruj listę liczb mniejszych niż 200 i podzielnych przez 11.

♠28 Wyznacz graficznie, powiększając odpowiedni fragment diagramu bifurkacyjnego i zwiększając jego rozdzielczość wartość parametru λ, dla którego funkcja f_λ ma orbitę o długości 3.

♠29 Odczytaj z diagramu bifurkacyjnego krotności orbit pojawiających się w pierwszych 4 bifurkacjach.

♠$_{30}$ Stosując funkcję `HistogramList` do listy służącej podczas ćwiczeń do narysowania diagramu bifurkacyjnego narysuj funkcją `ArrayPlot` wykres, w którym na osi poziomej będą wartości λ, na osi pionowej wartosci $y \in [0, 1]$, a intensywność koloru każdego piksela wykresu będzie zależeć od częstości występowania danych wartości y w orbicie funkcji f_λ.

Fraktale

We need to go deeper!
Incepcja

Celem ćwiczeń będzie analiza i wizualizacja zbiorów fraktalnych w tym (trójkowego) zbioru Cantora, zbiorów Julii i Mandelbrota dla rodziny kwadratowej, a także obliczanie wymiaru fraktalnego dla rzeczywistych obiektów.

Plan badań

1. Wyznacz empirycznie wymiar fraktalny dla zbioru Cantora. Zestaw dopasowaną krzywą z danymi empirycznymi na jednym wykresie. Porównaj ze ścisłym wynikiem.

Sugeruję rysować na wykresie `ListLogLogPlot`, a krzywą dopasowywać funkcja `FindFit`.

2. Narysuj graficzną wizualizację zbioru Cantora.

Polecamy funkcje `Line` i wycinanie ręczne lub automatyzacja przy pomocy operacji na listach i/lub funkcji `RealDigits`.

3. Wygeneruj binarny obrazek rzeczywistego obiektu o fraktalnej naturze

Polecamy funkcję `CountryData` i rysowanie konturów państw.

4. Wyznacz eksperymentalnie wymiar fraktalny swojego rysunku.

Pomocna okaże się funkcja `ImageTrim` i operacje na listach.

5. Wykonaj wizualizacje zbiorów Julii (wypełnionych) i Mandelbrota.

Oczywiście wybór `JuliaSetPlot` i `MandelbrotSetPlot` odradzamy – popsułoby to całą zabawę.

Pytania i problemy

♣$_1$ Jak działa funkcja `Graphics`? Podaj przykład jej zastosowania.

♣$_2$ Jak działa funkcja `Line`? Podaj przykład jej zastosowania.

♣$_3$ Jak działa funkcja `Rectangle`? Podaj przykład jej zastosowania.

♣$_4$ Jak działa funkcja `Polygon`? Podaj przykład jej zastosowania.

♣$_5$ Jak działa funkcja `Disk`? Podaj przykład jej zastosowania.

♣$_6$ Jak działa funkcja `Circle`? Podaj przykład jej zastosowania.

♣$_7$ Jak działa funkcja `Arrow`? Podaj przykład jej zastosowania.

♣$_8$ Jak działa funkcja `Thickness`? Podaj przykład jej zastosowania.

♣$_9$ Jak działa funkcja `ArrayPlot`? Podaj przykład jej zastosowania.

♣$_{10}$ Jak działa funkcja `ListLogLogPlot`? Podaj przykład jej zastosowania.

♣$_{11}$ Jak działa funkcja `ImageTrim`? Podaj przykład jej zastosowania.

♣$_{12}$ Jak działa funkcja `ImageCrop`? Podaj przykład jej zastosowania.

♣13 Jak działa funkcja `ImageValue`? Podaj przykład jej zastosowania.

♣14 Czym różni się dyrektywa `Thick` od `Thickness`?

♣15 Jak działa funkcja `CountryData`? Podaj przykład jej zastosowania.

♣16 Jak działa funkcja `FindFit`? Podaj przykład jej zastosowania.

♣17 Jak działa funkcja `ReplaceAll` (`\.`)? Podaj przykład jej zastosowania.

♣18 Jak zmienić grubość lini podczas rysowania grafiki?

♣19 Jak zmienić styl lini podczas rysowania grafiki?

♣20 Jak wczytać obrazek do środowiska Mathematica?

♣21 Jak działa funkcja `NestWhile`? Podaj przykład jej zastosowania.

♣22 Jak działa funkcja `Map`? Podaj przykład jej zastosowania.

♣23 Jak działa funkcja `Apply`? Podaj przykład jej zastosowania.

♣24 Jak działa funkcja `ImageMeasurements`? Podaj przykład jej zastosowania.

♣25 Jak działa dyrektywa `ColorFunctions`? Podaj przykład jej zastosowania.

♣26 Co robi `@`? Podaj przykład zastosowania.

♣27 Co robi `\@`? Podaj przykład zastosowania.

♣28 Co robi `@@`? Podaj przykład zastosowania.

♣29 Co robi `@@@`? Podaj przykład zastosowania.

♣30 Co robi `#` i `&`? Podaj przykład zastosowania.

♠1 Wyznacz empirycznie wymiar fraktalny wypełnionego zbioru Julii dla 3 wybranych wartości c.

♠2 Połącz wykresy wypełnionych zbiorów Julii i Mandelbrota w jeden blok `Manipulate`, w którym klikając na punkt zbioru Mandelbrota kod rysuje odpowiadający temu punktowi zbiór Julii.

♠3 Narysuj wypełnione zbiory Julii w większej rozdzielczości dla 3 wybranych wartości c.

♠4 Narysuj mapę polityczną Afryki, wykorzystując funkcje `CountryData`.

♠5 Narysuj mapę polityczną Ameryki Południowej, wykorzystując funkcje `CountryData`.

♠$_6$ Narysuj w większej rozdzielczości wybrany fragment zbioru Mandelbrota, wyznacz jego empiryczny wymiar fraktalny.

♠$_7$ Narysuj mapę polityczną Ameryki Północnej, wykorzystując funkcje `CountryData`.

♠$_8$ Połącz wykresy wypełnionych zbiorów Julii i Mandelbrota w jeden blok `Manipulate`, w którym klikając na punkt zbioru Mandelbrota kod rysuje odpowiadający temu punktowi zbiór Julii.

♠$_9$ Narysuj wypełnione zbiory Julii w większej rozdzielczości dla 3 wybranych wartości c.

♠$_{10}$ Narysuj mapę polityczną Azji, wykorzystując funkcje `CountryData`.

♠$_{11}$ Narysuj w większej rozdzielczości wybrany fragment zbioru Mandelbrota, wyznacz jego empiryczny wymiar fraktalny.

♠$_{12}$ Narysuj polityczną mapę Europy, wykorzystując funkcje `CountryData`.

♠$_{13}$ Wyznacz empirycznie wymiar fraktalny wypełnionego zbioru Julii dla 3 wybranych wartości c.

♠$_{14}$ Dopasuj parametry wybranej krzywej do ciągu liczb Fibonacciego.

♠$_{15}$ Połącz rysunki wypełnionych zbiorów Julii i Mandelbrota w jeden blok `Manipulate`, w którym klikając na punkt zbioru Mandelbrota kod rysuje odpowiadający temu punktowi zbiór Julii.

♠$_{16}$ Narysuj wypełnione zbiory Julii w większej rozdzielczości dla 3 wybranych wartości c.

♠$_{17}$ Dopasuj parametry wybranej krzywej do ciągu liczb pierwszych.

♠$_{18}$ Narysuj w większej rozdzielczości wybrany fragment zbioru Mandelbrota, wyznacz jego empiryczny wymiar fraktalny.

♠$_{19}$ Dopasuj parametry wybranej krzywej do ciągu liczb Fibonacciego.

♠$_{20}$ Wyznacz empirycznie wymiar fraktalny wypełnionego zbioru Julii dla 3 wybranych wartości c.

♠$_{21}$ Wczytaj plik z grafiką przedstawiającą wybraną roślinę, wyznacz empirycznie jej wymiar fraktalny.

♠$_{22}$ Narysuj wypełnioneg zbiory Julii w większej rozdzielczości dla 3 wybranych wartości c.

♠23 Wczytaj plik z grafiką przedstawiającą wybraną muszlę, wyznacz empirycznie jej wymiar fraktalny.

♠24 Połącz rysunki wypełnionych zbiorów Julii i Mandelbrota w jeden blok `Manipulate`, w którym klikając na punkt zbioru Mandelbrota kod rysuje odpowiadający temu punktowi zbiór Julii.

♠25 Narysuj w większej rozdzielczości wybrany fragment zbioru Mandelbrota, wyznacz jego empiryczny wymiar fraktalny.

♠26 Wczytaj plik z grafiką przedstawiającą deltę wybranej rzeki, wyznacz empirycznie jej wymiar fraktalny.

♠27 Wczytaj plik z grafiką przedstawiającą układ naczyń krwionośnych, wyznacz empirycznie jej wymiar fraktalny.

♠28 Wyznacz empirycznie wymiar fraktalny wypełnionego zbioru Julii dla 3 wybranych wartości c.

♠29 Połącz rysunki wypełnionego zbioru Julii i Mandelbrota w jeden blok `Manipulate`, w którym klikając na punkt zbioru Mandelbrota kod rysuje odpowiadający temu punktowi zbiór Julii.

♠30 Narysuj wypełnione zbiory Julii w większej rozdzielczości dla 3 wybranych wartości c.

Ergodyczność

Wstrząśnięte, nie zmieszane — James Bond

Plan badań

1. Korzystając z twierdzenia Birkhoffa i wyników uzyskanych na pierwszych zajęciach dla obrotu na okręgu (o kąt niewspółmierny do π) oblicz numerycznie poniższe całki. Porównaj uzyskane wyniki z wartością dokładną.

   a) $\int_0^{2\pi} \sin(x) dx$,

   b) $\int_0^{2\pi} \cos^2(x) dx$.

Można użyć funkcji `Integrate[]`.

2. Wyestymuj miarę niezmienniczą dla $f_4 = 4x(1-x)$, w tym celu:

   - Wylosuj wektor m liczb pseudolosowych z rozkładu równomiernego na $[0, 1]$.
   - Na każdej ze współrzędnych rozważanego wektora wywołaj n-tą iterację funkcji f_4 Zadziałaj na każdej współrzędnej tego wektora n razy rozważaną funkcją.
   - Przy użyciu funkcji `HistogramList[]` utwórz z tej listy odpowiednią listę histogramu - sugerujemy samemu popracować nad optymalnym doborem liczby przedziałów i ich szerokości na jakie chcemy podzielić nasze dane.
   - Stosując funkcje `Partition[]` i `Riffle[]` tak przekształć uzyskaną listę aby można było narysować ją z wykorzystaniem funkcji `ListPlot[]`.
   - Porównaj uzyskany wynik z wynikiem dokładnym:

$$h(x) = \frac{1}{\pi\sqrt{x(1-x)}}.$$

Polecamy do tego celu `Nest[]` i `Map[]`.

Warto zapoznać się z opcją `PDF` dla funkcji `HistogramList[]` - będzie to pomocne.

3. Odwzorowanie z rodziny logistycznej $f_4(x)$ jest ergodyczne. Oblicz wartość oczekiwaną $\mathbb{E}\sqrt{X}$ jako wartość oczekiwaną rozkładu danego przez miarę niezmienniczą z poprzedniego zadania

$$h(x) = \frac{1}{\pi\sqrt{x(1-x)}}$$

   oraz poprzez iterację odwzorowania $f_4(x)$.

4. Wyznacz wszystkie deterministyczne układy dynamiczne na zbiorze trójelementowym. Narysuj ich macierze i diagramy przejścia w jednym kroku. Które są ergodyczne w sensie dynamicznym, a które w sensie procesów Markowa?

5. Zaproponuj łańcuch Markowa o 5 stanach, który jest ergodyczny. Znajdź stacjonarny rozkład prawdopodobieństwa, analizując jego macierz przejścia, a następnie ją iterując.

Przydatne mogą być funkcje `Eigensystem[]`, `Eigenvalues[]`, `Eigenvectors[]`

Pytania i problemy

♣$_1$ Wyjaśnij w jaki sposób można obliczyć n-tą potęgę macierzy.

♣$_2$ Wyjaśnij jak działa funkcja `Integrate[]`. Podaj przykład jej zastosowania.

♣$_3$ Wyjaśnij jak działa funkcja `D[]`. Podaj przykład jej zastosowania.

♣$_4$ Wyjaśnij jak działa funkcja `Solve[]`. Podaj przykład jej zastosowania.

♣$_5$ Wyjaśnij jak działa funkcja `NSolve[]`. Podaj przykład jej zastosowania.

♣$_6$ Czym różnią się porównania `==` (`Equal`) i `===` (`SameQ`)?

♣$_7$ Wyjaśnij jak działa funkcja `Eigenvalues[]` oraz zilustruj jej działanie na przykładzie.

♣$_8$ Wyjaśnij jak działa funkcja `Eigenvectors[]` oraz zilustruj jej działanie na przykładzie.

♣$_9$ Wyjaśnij jak działa funkcja `Eigensystem[]` oraz zilustruj jej działanie na przykładzie.

♣$_{10}$ Jaka jest różnica między funkcjami `Map[]` i `Apply[]`.

♣$_{11}$ Wyjaśnij oraz zilustruj na przykładzie działanie funkcji `NIntegrate[]`.

♣$_{12}$ Wyjaśnij oraz zilustruj na przykładzie działanie funkcji `WeightedAdjacencyGraph[]`.

♣$_{13}$ Wyjaśnij jak wykorzystać funkcję `ArrayPlot[]` do narysowania szachownicy 4×4.

♣$_{14}$ Wyjaśnij oraz zilustruj na przykładzie działanie funkcji `MatrixForm[]`.

♣$_{15}$ Wyjaśnij dlaczego kod: `MatrixForm`$[\{1,1\}] == \{2,2\}$ nie zwraca wartości logicznej `True`.

♣$_{16}$ Wyjaśnij jak zmienić kolor wybranej (tylko jednej) krawędzi w grafie generowanym za pomocą funkcji `WeightedAdjacencyGraph[]`.

♣$_{17}$ Wyjaśnij jak działa funkcja `Graph[]` oraz zilustruj jej działanie na przykładzie.

♣$_{18}$ Wyjaśnij jak działa funkcja `Flatten[]` oraz zilustruj jej działanie na przykładzie.

♣$_{19}$ Wyjaśnij oraz zilustruj na przykładzie działanie funkcji `Transpose[]`.

♣$_{20}$ Wyjaśnij jak przy pomocy funkcji `AdjacencyGraph[]` wygenerować graf pełny o 5 wierzchołkach.

♣$_{21}$ Wyjaśnij i zilustruj przykładem działanie funkcji `Total[]`.

♣$_{22}$ Wyjaśnij oraz zilustruj przykładem działanie funkcji `Max[]` i `Min[]`.

♣$_{23}$ Wyjaśnij oraz zilustruj przykładem działanie dyrektywy `GraphLayout`.

♣$_{24}$ Wyjaśnij oraz zilustruj przykładem działanie dyrektywy `VertexyStyle`

♣$_{25}$ Wyjaśnij oraz zilustruj przykładem działanie dyrektywy `EdgeStyle`

♣$_{26}$ Czym różnią się funkcje `DirectedEdge[]` i `UndirectedEdge[]`?

♣$_{27}$ Wyjaśnij oraz zilustruj przykładem działanie funkcji `Mean[]`

♣$_{28}$ Wyjaśnij jak działa funkcja `Histogram[]`, zilustruj jej działanie przykładem.

♣$_{29}$ Jak w środowisku Mathematica wygenerowac liczby pseudolosowe z rozkładu jednostajnego na zadanym przedziale?

♣$_{30}$ Jak w środowisku Mathematica wygenerowac liczby pseudolosowe z zadanego rozkładu?

Proszę szukać wbudowanej funkcji, a nie implementować np. metodę odwrotnej dystrybuanty!

♠$_{1}$ Niech $n \in \mathbb{N}$, parzysta. W jaki sposób w grafie o n wierzchołkach generowanym za pomocą funkcji `WeightedAdjacencyGraph[]` pokolorować parzyste wierzchołki na zielono a nieparzyste na żółto?

Pomocna będzie funkcja `Table[]`.

♠$_{2}$ Zaproponuj proces Markowa o co najmniej 6 stanach posiadający stany o różnym okresie, tak aby graf reprezentujący ten proces był spójny (jako graf **nieskierowany**). Dla zaproponowanego procesu:

- narysuj diagram przejścia w jednym kroku,
- narysuj macierz przejścia w jednym kroku,
- znajdź wszystkie stacjonarne rozkłady prawdopodobieństwa.

♠$_{3}$ Dla dowolnego grafu zapisz macierz przejścia w jednym kroku dla cząstki błądzącej po nim przypadkowo.

♠$_{4}$ Zaproponuj proces Markowa o okresie 2 posiadający co najmniej 6 stanów, tak aby graf reprezentujący ten proces był spójny (jako graf **skierowany**). Dla zaproponowanego procesu:

- narysuj diagram przejścia w jednym kroku,
- narysuj macierz przejścia w jednym kroku,

- znajdź wszystkie stacjonarne rozkłady prawdopodobieństwa.

$\spadesuit_5$ Wykorzystując funkcję `Eigensystem[]` znajdź wszystkie stacjonarne rozkłady prawdopodobieństwa dla grafu zadanego macierzą przejścia **(Rozumianej w myśl definicji 11)**

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0.5 & 0 & 0 & 1 \\ 0 & 0 & 0.5 & 0.5 & 0 \end{pmatrix}.$$

$\spadesuit_6$ Dla dowolnego grafu zapisz macierz przejścia w jednym kroku dla cząstki błądzącej po nim przypadkowo.

$\spadesuit_7$ Powtórz zadanie $\spadesuit_5$ dla macierzy przejścia:

$$\begin{pmatrix} 1 & 0 & 0 & 0.3 & 0 \\ 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 1 \\ 0 & 0 & 0 & 0.2 & 0 \end{pmatrix}$$

$\spadesuit_8$ Wykorzystując funkcje `AdjacencyGraph[]`, `Table[]`, `If[]` napisz program, który dla zadanego $n \in \mathbb{N}$ wygeneruje graf pełny o n wierzchołkach.

$\spadesuit_9$ Wykorzystując funkcje `Graph[]`, `Table[]`, `If[]` napisz program, który dla zadanego $n \in \mathbb{N}$ wygeneruje graf pełny o n wierzchołkach.

$\spadesuit_{10}$ Napisz funkcję, która, wykorzystując funkcję `Total[]` sprawdza czy podana w jej argumencie macierz jest macierzą przejścia w jednym kroku (w myśl definicji 11).

$\spadesuit_{11}$ Napisz funkcję, która, wykorzystując funkcję `Plus[]` sprawdza czy podana w jej argumencie macierz jest macierzą przejścia w jednym kroku (w myśl definicji 11).

$\spadesuit_{12}$ Napisz funkcję która dla zadanego $n \in \mathbb{N}$ zwracać będzie przy pomocy `ArrayPlot[]` graficzną reprezentację macierzy $n \times n$ przedstawiającą pikselowe koło, czyli figurę, w której małymi kwadratami przybliża się z coraz większą dokładnością (wraz ze wzrostem wartości n) wnętrze dwuwymiarowej kuli.

$\spadesuit_{13}$ Wykonaj zadanie $\spadesuit_{12}$ przybliżając wnętrze dwuwymiarowej figury danej następująco:

$$P = \left\{ (x, y) \in \mathbb{R}^2 : 4 + 2x < x^2 + y^2 < 9 \right\}.$$

♠$_{14}$ Wykonaj zadanie ♠$_{12}$ przybliżając wnętrze trójkąta równobocznego.

♠$_{15}$ Wykonaj dwie animacje P^n (wypisanej jako macierz), w funkcji $n = 1, \ldots, 10$, gdzie P to macierz przejścia w jednym kroku dla ergodycznego i okresowego procesu Markowa

♠$_{16}$ Stosując funkcje `Apply[]` i `Reverse[]` przygotuj kod, który pozwoli zmienić kierunek skierowanych krawędzi w wybranym grafie skierowanym.

♠$_{17}$ Stwórz macierz 10×10, której (n, k)-ty elementy to k-ta cyfra po przecinku liczby π^n.

♠$_{18}$ Napisz program, który dla zadanego $n \in \mathbb{N}_+$ zawraca graf będący siatką kwadratową. Zapisz jego macierz sąsiedztwa.

Do wygenerowania grafu polecam użyć funkcji `Graph[]`.

♠$_{19}$ Znajdź proces Markowa, który opisany jest macierzą podwójnie stochastyczną.

♠$_{20}$ Napisz skrypt, który tworzy w losowy sposób macierz przejścia procesu Markowa o 20 stanach. Rozstrzygnij jaki typ dynamiki prezentuje.

♠$_{21}$ Dla wybranego procesu ergodycznego P o co najmniej 5 stanach i losowej wartości początkowej x_0 narysuj wykres $n(\epsilon)$ zależności minimalnej liczby iteracji n, dla których $|P^n x_0 - P^\infty x_0|_2 < \epsilon$.

♠$_{22}$ Posługując się funkcją `GridGraph[]` wyznacz macierz przejścia w jednym kroku dla błądzenia losowego po siatce kwadratowej.

♠$_{23}$ Mając daną macierz przejścia w jednym kroku (definiuje ona w sposób jednoznaczny pewien graf) napisz funkcję zwracającą listę stopni jego wierzchołków. Zilustruj na przykładzie działanie tej funkcji.

♠$_{24}$ Znajdź proces Markowa, który opisany jest macierzą podwójnie stochastyczną.

♠$_{25}$ Mając do dyspozycji wybrany graf nieskierowany zadany za pomocą funkcji `Graph[]` napisz funkcję zwracającą listę stopni jego wierzchołków. Zilustruj działanie tej funkcji na przykładzie.

♠$_{26}$ Dla wybranego procesu ergodycznego P i losowej wartości początkowej x_0 narysuj wykres $\epsilon(n)$ zależności $\epsilon = |P^n x_0|_2$ od liczby iteracji n.

♠$_{27}$ Dla ogólnej postaci macierzy przejścia w jednym kroku dwustanowego procesu Markowa

$$\begin{pmatrix} p & 1-q \\ 1-p & q \end{pmatrix},$$

gdzie $p, q \in [0,1]$, wyznacz symboliczny wzór na rozkład stacjonarny. Wyznacz warunek na p i q kiedy ta macierz jest podwójnie stochastyczna, wypisz wzór na rozkład stacjonarny dla tego przypadku.

♠$_{28}$ Dla rozkładu prawdopodobieństwa danego przez $(p, 1-p)$, gdzie $p \in [0,1]$, przy pomocy funkcji `D[]` znajdź i określ ekstremum entropii danej przez $\mathcal{H}(p_1, p_2, \ldots, p_n) = -\sum_{k=1}^{n} -p_k \log p_k$.

♠$_{29}$ Napisz funkcję, która generuje wszystkie macierze stochastyczne o elementach ze zbioru $\{0, 1\}$ i wymiarze $n \times n$. Pokaż jej działanie na przykładzie $n = 3$.

♠$_{30}$ Znajdź proces Markowa, który opisany jest macierzą podwójnie stochastyczną.

Literatura

[1] A. Iwanik, J.K. Misiewicz. *Wykłady z procesów stochastycznych z zadaniami: Część pierwsza – Procesy Markowa*. Oficyna Wydawnicza Uniwersytetu Zielonogórskiego, 2009.

[2] A. Lasota, M. C. Mackey. *Chaos, Fractals, and Noise. Stochastic Aspects of Dynamics*. Springer-Verlag, 1994.