

Modelowanie i Identyfikacja Opis tematów

prowadzący: M. Filiński, K. Kowalczyk, G. Maik

1 Wymagania ogólne

1. Zespoły:

- Projekty realizowane w zespołach 4-osobowych. W szczególnych przypadkach dopuszczalne są grupy większe lub mniejsze.
- Temat projektu nie mogą się powtarzać w obrębie jednej grupy zajęciowej.
- W przypadku zespołów mieszanych wymagana jest prezentacja wyników w każdej grupie zajęciowej.

2. Ocena projektu:

- Plan projektu – powinien zawierać informacje o zakresie pracy (wstępna decyzja o opcjonalnych punktach – nie jest ostateczna) oraz wstępny plan pracy z podziałem na członków grupy. Do przygotowania tego punktu powinna zostać przeprowadzony wstępny przegląd literatury i zapoznanie się z tematem projektu – termin oddania dwa tygodnie od pierwszych zajęć.
- Prezentacja¹ 1 – przegląd literatury i pierwsze wyniki symulacji – termin indywidualny dla każdej grupy, mniej więcej w połowie semestru.
- Prezentacja¹ 2 – podsumowanie wyników i prezentacja wykonanego zadania – ostatnie zaplanowane zajęcia
- Ocena samego projektu – wykonanie projektu i jego dokumentacja. Wykonanie dodatkowych opcjonalnych zadań lub rozszerzenie projektu o inne dodatkowe elementy (najlepiej skonsultować z prowadzącym w celu uniknięcia nieporozumienia podczas wystawiania oceny)
- Potępy w wykonanych zadaniach powinny być zarejestrowane, przykładowo wkład w kod projektu może być rejestrowany na githubie, a przegląd literatury w postaci notatek, raportów.

¹prezentacja powinna trwać nie dłużej niż 12 minut + 5 minuty pytania.

2 Spis tematów – ogólny

Tematy będą miały przypisany szacowany poziom trudności wykonania, który częściowo determinuje możliwe oceny do zdobycia. Należy je rozumieć jako ocenę do uzyskania przy średnim nakładzie pracy i powierzchownym zrozumieniem tematu. Realizacja każdego projektu umożliwia zdobycie dowolnej oceny.

1. (5.0) Modelowanie efektów dyfuzji informacji w mediach społecznościowych przy użyciu symulacji Monte Carlo
2. (5.0) Wykorzystanie modeli agentowych do symulacji rzeczywistych zjawisk (przykładowo symulator ruchu drogowego)
3. (4.5) Przegląd metod optymalizacji wypukłej w modelowaniu statystycznym wraz z prezentacją i wizualizacją ich działania.
4. (4.5) Aplikacja do kompresji danych jednowymiarowych i dwuwymiarowych wraz z prezentacją i wizualizacją ich działania.
5. (4.5) Generator próbek z rozkładów bazując na metodach Markov chain Monte Carlo (MCMC)
6. (4.5) Przegląd metod redukcji wymiarów w danych wielowymiarowych wraz z prezentacją i wizualizacją ich działania.
7. (4.0) Analiza i implementacja generatorów liczb pseudolosowych wraz z badaniem jakości wygenerowanych danych.

3 Spis tematów – szczegółowy

3.1 Modelowanie efektów dyfuzji informacji w mediach społecznościowych przy użyciu symulacji Monte Carlo.

Cel projektu

Przygotowanie aplikacji, która umożliwi przygotowanie, wykonanie i przedstawienie wyników działania dyfuzji informacji w mediach społecznościowych. Aplikacja powinna zawierać interfejs użytkownika, który umożliwi ustawienie parametrów symulacji, danych wejściowych (przykładowo dane rzeczywiste), wybór modelu oraz uruchomienie symulacji. Dodatkowo aplikacja powinna mieć możliwość wyświetlenia wyników, w zależności od prezentacji (np. graf reprezentujący sieć społeczną, wykresy przedstawiające dynamikę rozprzestrzeniania się informacji, statystyki symulacji).

Główne zadania

1. Budowa sieci społecznej:

- budowa struktury sieci (grafu) reprezentującego interakcje między użytkownikami.
- *Opcjonalnie*: wykorzystanie danych rzeczywistych (np. dane z API mediów społecznościowych, publiczne zbiory danych).

2. Model dyfuzji:

- Wybór modelu dyfuzji (np. Independent Cascade, Linear Threshold). Implementacja
- Implementacja symulacji Monte Carlo do modelowania rozprzestrzeniania się informacji w sieci.

3. Projekt i implementacja interfejsu użytkownika:

- Zaprojektowanie graficznego interfejsu umożliwiającego łatwe ustawianie parametrów symulacji (takich jak prawdopodobieństwo propagacji, liczba iteracji, wybór modelu).
- Możliwość wyboru źródła danych (np. dane symulowane vs. dane rzeczywiste).
- Wizualizacja wyników:
 - Graficzne przedstawienie sieci społecznej (np. graf węzłów i krawędzi).
 - Wykresy ilustrujące dynamikę rozprzestrzeniania się informacji oraz statystyki symulacji.

3.2 Wykorzystanie modeli agentowych do symulacji rzeczywistych zjawisk (przykładowo symulator ruchu drogowego)

UWAGA! Opis przygotowany jest dla symulatora ruchu drogowego. Grupa może ustalić zmianę głównego zastosowania modelu.

Cel projektu

Celem projektu jest stworzenie symulatora ruchu drogowego opartego na modelu agentowym, w którym każdy pojazd (agent) podejmuje decyzje dotyczące prędkości, zmiany pasa czy zachowania względem innych uczestników ruchu. System powinien umożliwiać wizualizację przebiegu symulacji oraz analizę wybranych aspektów ruchu (np. korki, przepustowość).

Główne zadania

1. Model ruchu i zachowania agentów:

- Zaprojektowanie uproszczonego modelu ruchu drogowego (np. siatka dróg, skrzyżowania).
- Zdefiniowanie zachowań kierowców (przyspieszanie, hamowanie, reakcja na inne pojazdy, sygnalizację świetlną).

2. Implementacja modelu agentowego:

- Reprezentacja pojazdów jako agentów podejmujących decyzje na podstawie lokalnych informacji.
- Mechanizmy kolizji, zatorów i reagowania na sygnały świetlne.

3. Symulacja i wizualizacja:

- Graficzna prezentacja układu dróg i poruszających się pojazdów (np. w formie 2D).
- *Opcjonalne:* Wykorzystanie rzeczywistych danych do stworzenia układu dróg np. OpenStreetMap.
- *Opcjonalne:* Wykorzystanie dostępnych informacji dotyczących zachowań kierowców do dopasowania modelu agentowego do rzeczywistości.
- Możliwość śledzenia wybranych statystyk (np. średnia prędkość, natężenie ruchu, liczba wypadków).

4. Eksperymenty i analiza:

- Przeprowadzenie symulacji dla różnych natężeń ruchu, konfiguracji skrzyżowań czy zachowań kierowców.
- Analiza danych (np. zależność między liczbą pojazdów a czasem przejazdu, formowanie się korków).

3.3 Przegląd metod optymalizacji wypukłej w modelowaniu statystycznym wraz z prezentacją i wizualizacją ich działania.

Cel projektu

Celem projektu jest przygotowanie aplikacji lub zestawu narzędzi wykorzystujących metody optymalizacji wypukłej do rozwiązywania wybranych problemów modelowania statystycznego. Projekt powinien umożliwiać dopasowywanie modeli do danych (np. modele regresji, klasyfikacji), a także prezentować i wizualizować proces optymalizacji oraz uzyskane wyniki.

Główne zadania

1. **Przegląd literatury i metod optymalizacji wypukłej:** Zapoznanie się z podstawami teoretycznymi optymalizacji wypukłej (np. gradient descent, metoda Newtona), a także z ich zastosowaniami w wybranych zagadnieniach statystycznych.
2. **Wybór i implementacja algorytmów:**
 - Implementacja co najmniej trzech metod optymalizacji wypukłej.
 - Zastosowanie tych metod w konkretnych problemach (np. regresja liniowa, regresja logistyczna).
3. **Przygotowanie danych:**
 - Stworzenie lub pozyskanie zbioru danych (symulowanych), na których będą testowane algorytmy.
 - Wstępne przetwarzanie danych (standaryzacja, oczyszczanie, podział na zbiór treningowy/testowy).
 - *Opcjonalnie:* Wykorzystanie rzeczywistych danych do przeprowadzenia optymalizacji.
4. **Implementacja i wizualizacja wyników:**
 - *Opcjonalnie:* Implementacja metod "od podstaw", tzn. wykorzystując równania do implementacji, a nie gotowe pakiety.
 - Prezentacja zbieżności algorytmów (np. wartość funkcji celu w kolejnych iteracjach).
 - Wyznaczenie i omówienie jakości uzyskanych modeli (np. błędy predykcji, macierz pomyłek).

3.4 Aplikacja do kompresji danych jednowymiarowych i dwuwymiarowych wraz z prezentacją i wizualizacją ich działania.

Cel projektu

Celem projektu jest stworzenie aplikacji umożliwiającej kompresję danych jednowymiarowych (np. sygnały) oraz dwuwymiarowych (np. obrazy), a następnie prezentację i wizualizację procesu kompresji oraz efektów końcowych. Aplikacja powinna oferować porównanie różnych metod kompresji oraz pozwalać na modyfikację parametrów wpływających na jakość i stopień kompresji.

Główne zadania

1. **Przegląd metod kompresji:**
 - Metody bezstratne (np. Huffman).

- Metody stratne (np. DCT dla obrazów, transformata falkowa).

2. Implementacja wybranych algorytmów:

- Implementacja metod kompresji.
- *Opcjonalne:* Implementacja metod kompresji "od podstaw", tzn. wykorzystując równania do implementacji, a nie gotowe pakiety.
- Zaimplementowanie obu metod w aplikacji.

3. Kompresja danych jednowymiarowych i dwuwymiarowych:

- Testowanie wybranych metod na różnego rodzaju danych (np. sygnały audio, obrazy).
- Analiza jakości i wielkości uzyskanych plików.

4. Interfejs i wizualizacja:

- Graficzny interfejs użytkownika pozwalający na załadowanie plików, wybór metody kompresji i jej parametrów.
- Prezentacja efektów kompresji (np. oryginalny vs skompresowany obraz, sygnał).
- Porównanie jakości (MSE) oraz wskaźników rozmiaru (stopień kompresji).

3.5 Generator próbek z rozkładów bazując na metodach Markov chain Monte Carlo (MCMC)

Cel projektu

Celem projektu jest stworzenie narzędzia (biblioteki lub aplikacji) służącego do generowania próbek z zadanego rozkładu przy pomocy metod Markov Chain Monte Carlo (MCMC), ze szczególnym naciskiem na rozkłady warunkowe (np. Gibbs sampling). Projekt powinien umożliwiać prezentację procesu generowania, wizualizację uzyskanych rozkładów oraz analizę zbieżności łańcuchów Markowa.

Główne zadania

1. **Przegląd teorii MCMC:** Zapoznanie się z podstawami teoretycznymi metod typu MCMC (Metropolis-Hastings, Gibbs).
2. **Implementacja algorytmów:**
 - Wybór co najmniej dwóch metod (np. Metropolis-Hastings i Gibbs sampler).
 - Implementacja generowania próbek z wykorzystaniem rozkładów warunkowych w metodach MCMC.
3. **Przykłady testowe i wizualizacja:**
 - Przygotowanie przykładowych funkcji gęstości (np. wielowymiarowe rozkłady Gaussa, rozkład banana shape).
 - Wizualizacja kolejnych próbek (np. w przestrzeni 2D) oraz ocena jakości próbkowania.
 - *Opcjonalne:* Wykorzystanie rzeczywistych danych do estymacji niezbędnych rozkładów.

3.6 Przegląd metod redukcji wymiarów w danych wielowymiarowych wraz z prezentacją i wizualizacją ich działania.

Cel projektu

Celem projektu jest dokonanie przeglądu i implementacji wybranych metod redukcji wymiarów dla danych wielowymiarowych (np. PCA, t-SNE, UMAP). Aplikacja powinna umożliwiać wczytywanie danych, przeprowadzanie redukcji wymiarów oraz wizualizację wyników (np. w 2D lub 3D), a także porównanie efektywności poszczególnych metod.

Główne zadania

1. Przegląd literatury i metod redukcji wymiarów:

- Podstawowe metody liniowe.
- Metody nieliniowe.

2. Implementacja wybranych algorytmów:

- Zaimplementowanie co najmniej jednej metody liniowej i jednej nieliniowej.
- *Opcjonalne:* Implementacja metod kompresji "od podstaw", tzn. wykorzystując równania do implementacji, a nie gotowe pakiety.

3. Przygotowanie danych testowych:

- Wybór rzeczywistych lub syntetycznych zbiorów danych o dużej liczbie wymiarów.
- Wstępne przetwarzanie danych (standaryzacja, normalizacja).

4. Analiza i porównanie metod:

- Wizualizacja wyników w przestrzeni o zredukowanym wymiarze (2D, 3D).
- Ocena jakości redukcji.

3.7 Analiza i implementacja generatorów liczb pseudolosowych wraz z badaniem jakości wygenerowanych danych.

Cel projektu

Celem projektu jest przygotowanie aplikacji umożliwiającej generowanie liczb pseudolosowych z rozkładu jednostajnego przy użyciu różnych metod. Aplikacja powinna również oferować możliwość generowania liczb pseudolosowych za pomocą metod opartych na dystrybuancie oraz funkcji gęstości prawdopodobieństwa, a także prezentować działanie tych metod wraz z wizualizacją procesu generowania i uzyskanych wyników. Każda z metod musi umożliwiać modyfikację parametrów. Ostatnim etapem projektu jest przetestowanie, czy wygenerowane rozkłady odpowiadają teoretycznym rozkładom.

Główne zadania

1. Przegląd literatury:

- Zapoznanie się z teorią generatorów liczb pseudolosowych, rozkładem jednostajnym, dystrybucją oraz funkcją gęstości prawdopodobieństwa.
- Przegląd metod generowania liczb pseudolosowych z rozkładu jednostajnego.

- Przegląd metod generowania liczb pseudolosowych z innych rozkładów (metoda odrzucania i odwrotnej dystrybucyjności) oraz testowania ich jakości.

2. Implementacja generatorów:

- Zaimplementowanie różnych metod generowania liczb pseudolosowych z rozkładu jednostajnego.
- Implementacja metod opartych na dystrybucji oraz funkcji gęstości prawdopodobieństwa
- *Opcjonalne:* Implementacja metody, która wygeneruje rozkład bazując na podanej dystrybucji (odwrotna dystrybucyjność wyznaczona numerycznie) – przykładowo rozkład normalny.

3. Projekt interfejsu użytkownika:

- Zaprojektowanie graficznego interfejsu umożliwiającego wybór metody, modyfikację parametrów oraz wizualizację procesu generowania i uzyskanych wyników (np. wykresy, histogramy, animacje).

4. Testowanie wygenerowanych rozkładów:

- *Opcjonalne:* Przeprowadzenie testów statystycznych (np. test chi-kwadrat) w celu porównania wygenerowanych rozkładów z teoretycznymi.