

# Wytyczne i wskazówki

Poniższe uwagi są pomyślane jako ogólne i użyteczne przy przeprowadzaniu i prezentowaniu wszelkiego rodzaju projektów programistycznych.

## 1 Przeprowadzanie symulacji

Wskazówki, jak sprawić, aby wyniki symulacji były prawidłowe.

- Działanie symulacji lepiej jest najpierw wstępnie sprawdzać, a dopiero później puszczać je w pełnej wersji. Np. kiedy generujemy próbę losową, lepiej jest zacząć od wygenerowania kilkunastu tysięcy punktów, a dopiero gdy wynik okaże się na oko rozsądny, wygenerować próbę większej długości. W ten sposób nie tracimy czasu na wielokrotne powtarzanie dłuższej symulacji, gdy coś jest nie tak i musimy ją poprawiać.
- Bardzo łatwo zużyć cały RAM i zawiesić komputer, gdy działamy na tablicach 2D. Zwiększenie wymiaru  $n$  razy zajmuje  $n^2$  więcej pamięci, trzeba uważać.
- Wyniki symulacji Monte Carlo **są losowe**. Nie możemy bezwarunkowo im ufać. Na przykład, kiedy generujemy 2 próby losowe, mierzymy ich korelację, i otrzymujemy wynik około 0.2, to wcale nie znaczy, że są dodatnio skorelowane. Może się okazać, że jeśli powtórzymy symulację, otrzymamy np. -0.3! Generalnie, im dłuższej próby losowej użyjemy, tym mniejsze będą losowe fluktuacje oszacowanych parametrów, ale zawsze warto powtórzyć symulację przynajmniej 1,2 razy i upewnić się, że wyniki nie wahają się znacząco. Pozwala to też określić, z jaką dokładnością powinniśmy podać wynik. Na przykład, jeśli powtarzając symulację uzyskujemy kolejno liczby 2.1423, 2.1289 oraz 2.0987, to widoczne jest, że wynik to 2.1, a dalszych miejsc po przecinku nie powinniśmy podawać, bo są nieznaczące.
- Sprawdzanie poprawności wyników symulacji Monte Carlo tylko 1 sposobem nigdy nie wystarczy, nawet najlepsza metoda statystyczna daje tylko częściowy obraz sytuacji. Im więcej jak najbardziej niezwiązanych ze sobą nawzajem estymatorów/testów, tym lepiej, 2 lub 3 to minimum.

## 2 Czytelność kodu

W większości przypadków napisany kod powinien być zrozumiały dla innych osób, które będą z niego korzystać. Nawet, jeśli będzie potrzebny tylko na własny użytek, to samemu po kilku miesiącach nie będzie się pamiętało szczegółów.

- Nazwy zmiennych powinny być czytelne, jeśli mówimy o obiektach matematycznych, dobrze trzymać się istniejącej konwencji nazewnictwa, np. ruch Browna zwykle oznaczany jest przez  $B$ , więc warto go też tak nazwać w symulacji. Z nazwy funkcji powinno wynikać, co ona robi.
- W kodzie mają być komentarze, ale powinny być **krótkie**. Nikomu nie chce się czytać esejów, najważniejszy jest kod. Przy najważniejszych zmiennych trzeba napisać, czym są, przy każdej zadeklarowanej funkcji co pobiera i co zwraca.

### 3 Wykresy

- Każdy wykres musi mieć podpisane osie, w przypadku obecności wielu linii legendę.
- Każdy wykres jest załączony w konkretnym celu. Powinien być więc opisany - czytelnik musi wiedzieć, co z niego wynika i dlaczego wynika. Np. „na poniższym wykresie wyestymowana gęstość prawdopodobieństwa zgadza się z teoretyczną gęstością dla rozkładu  $\mathcal{N}(0, 1)$  co pokazuje, że symulacja poprawnie zwraca próbę z tego rozkładu.”
- Jeśli porównujemy jakieś funkcje, rozsądnie jest umieścić je na 1 wykresie, o ile tylko się da. Jeśli różne wykresy zawierają uzupełniające się informacje, warto je umieścić obok siebie.

### 4 Opisy

- Zawsze piszemy co robimy, jak robimy i czym się skończyły nasze działania.
- Kiedy używamy powszechnie znanych algorytmów, nie ma sensu tracić miejsca oraz czasu na ich opisywanie punkt po punkcie. Starczy nazwa albo krótki identyfikujący opis. Jeśli przedstawiamy coś własnego, należy oczywiście prostymi słowami wyjaśnić, co się robi (patrz punkt wyżej), ale zamieszanie pseudokodu nie ma sensu, szczególnie, gdy czytelnik ma do dyspozycji czytelny kod implementacji w konkretnym języku.
- Podobnie, nie trzeba wnikać w szczegóły działania powszechnie znanych metod statystycznych. Natomiast trzeba zawsze precyzować, której się używa. Powszechne są błędy typu pisanie „przedstawiamy dystrybuantę wysymulowanej próby losowej.” Nie ma czegoś takiego! Dystrybuantę próby można oszacować, używając konkretnej metody, np. **dystrybuanty empirycznej**. Podobnie nie prezentujemy gęstości próby, tylko **jądrowy estymator gęstości** lub **histogram**, które estymują gęstość. Dokładne funkcje i wartości parametrów możemy podać wyłącznie dla rozkładu teoretycznego, tzn. takiego, jaki powinien wyjść.