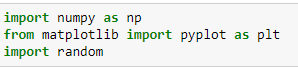
***MOWNIT - Zestaw 5 Całkowanie Monte Carlo***

*Opracował: Mateusz Woś*

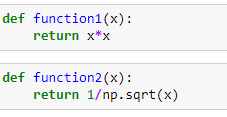
Większość zadań została zaimplementowana w języku Python. W ostatnim zadaniu ze względu na brak wymienionych metod w standardowych bibliotekach Pythona skorzystałem z biblioteki gsl, natomiast wykresy z wygenerowanych już danych rysowałem w matplotlibie.

1. *Napisać funkcję liczącą całkę metodą "hit-and-miss". Czy będzie ona dobrze działać dla funkcji 1/sqrt(x)?*

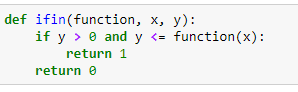
Zaimportowanie bibliotek:



Następnie zdefiniowałem pierwszą i drugą funkcję:



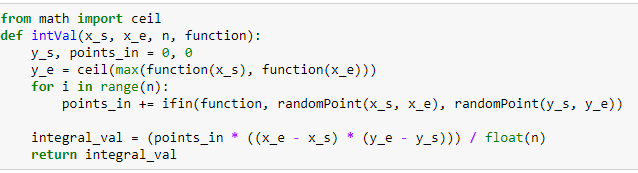
Funkcja sprawdzająca czy punkt leży pod wykresem:



Funkcja generująca losowy punkt z wyznaczonego obszaru:



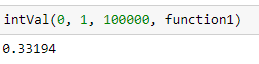
Funkcja hit and miss:



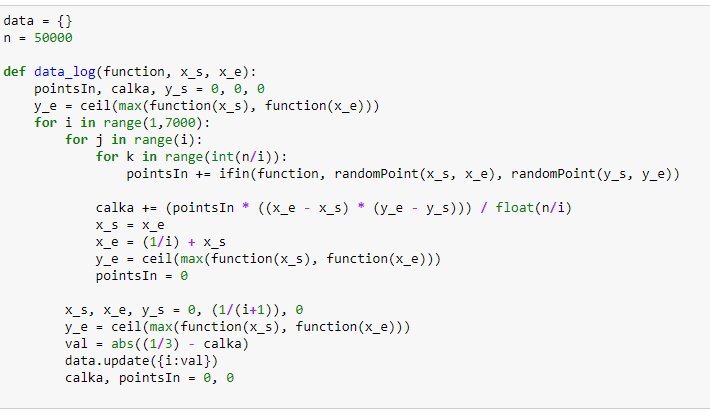
Funkcja hit and miss nie poradzi sobie z drugą funkcją: , ponieważ nie da się poprawnie wyznaczyć wartości zakresów wysokość w tym przypadku. Funkcja dla wartości x = 0 będzie dążyć do nieskończoności.

1. *Policzyć całkę przy użyciu napisanej funkcji. Jak zmienia się błąd wraz ze wzrostem liczby podprzedziałów? Narysować wykres tej zależności przy pomocy Gnuplota. Przydatne będzie skala logarytmiczna.*

Wartość pierwszej funkcji, używając wcześniej napisanej funkcji. Przedział [0,1] dla 100000 ilość prób trafień.

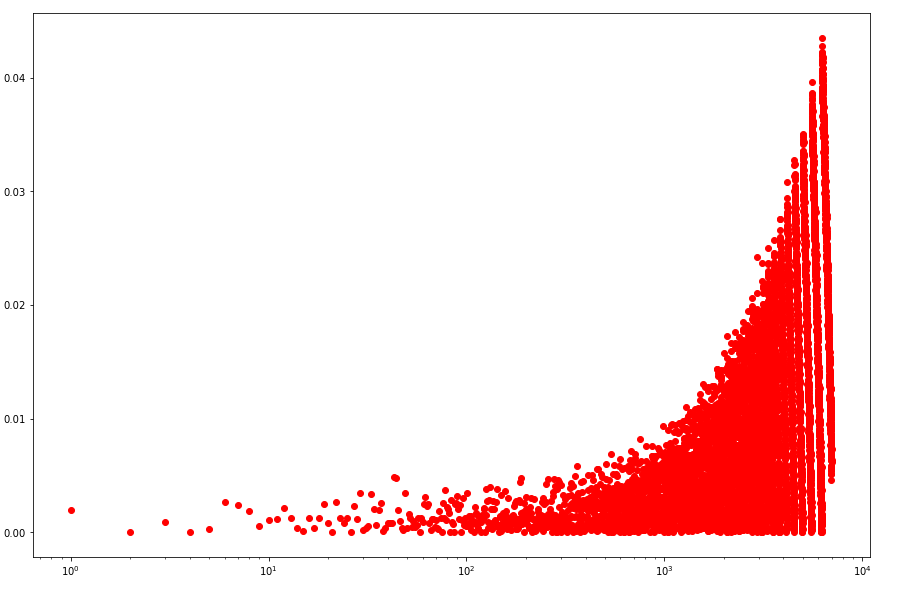


Do obliczenia błędów napisałem, poniższą funkcję:



Obliczenia wykonałem dla 7000 przedziałów i dla N = 50000. Nie wiem czy słusznie, ale ilość możliwych trafień zawsze dzieliłem pomiędzy podprzedziały.

Z otrzymanych danych wyrysowałem wykres:

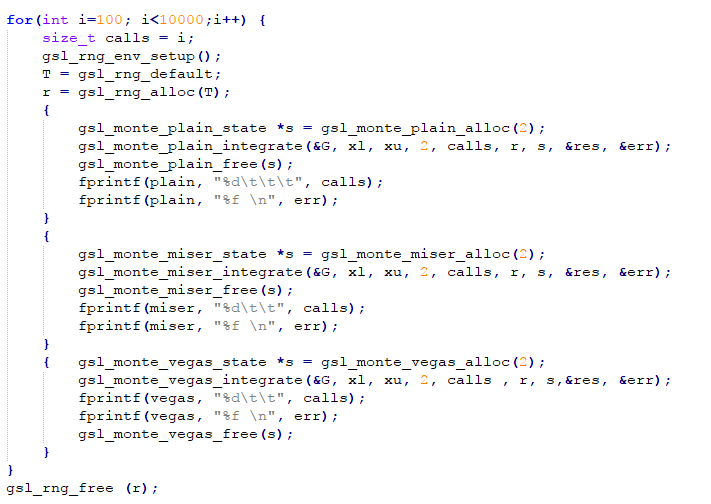


1. *Policzyć wartość całki korzystając z funkcji Monte Carlo z GSL. Narysować wykres zależności błędu od ilości wywołań funkcji dla różnych metod (PLAIN, MISER, VEGAS).*

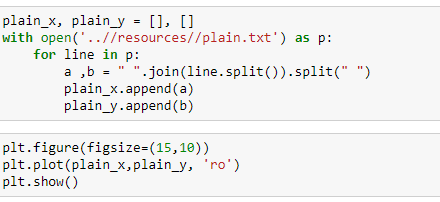
Posłużyłem się teraz biblioteką gsl. Wygenerowałem dane z wszystkich metod do plików .txt i narysowałem wykresy w Pythonie.

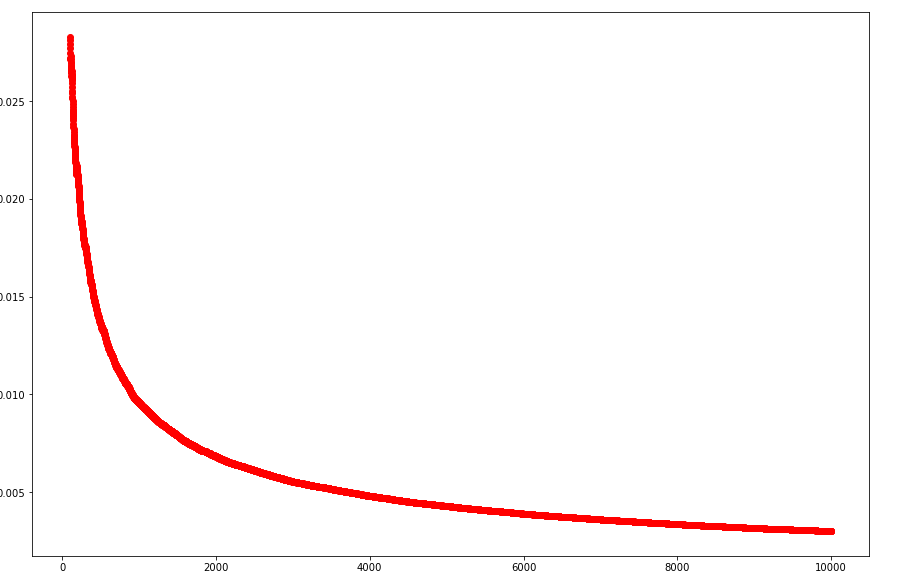
Wartość pierwszej funkcji przy użyciu metod z gsl’a była bardzo podbna do tego co otrzymałem z mojej implementacji hit and miss.

Kod generujący wartości błędów metod plain, miser i vegas:

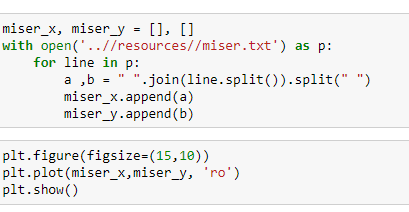


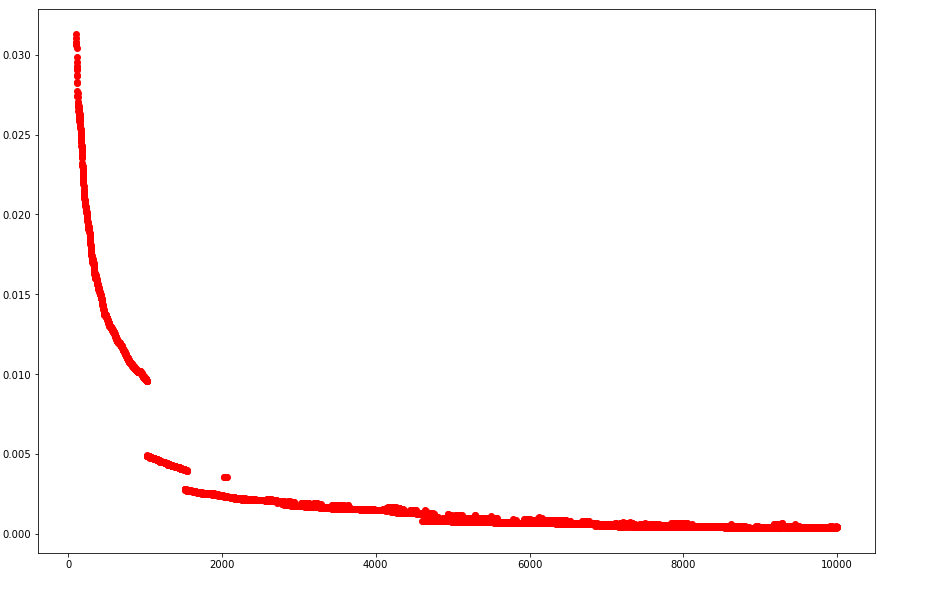
**Metoda plain:**



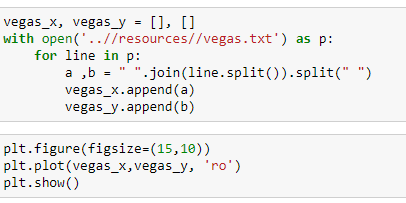


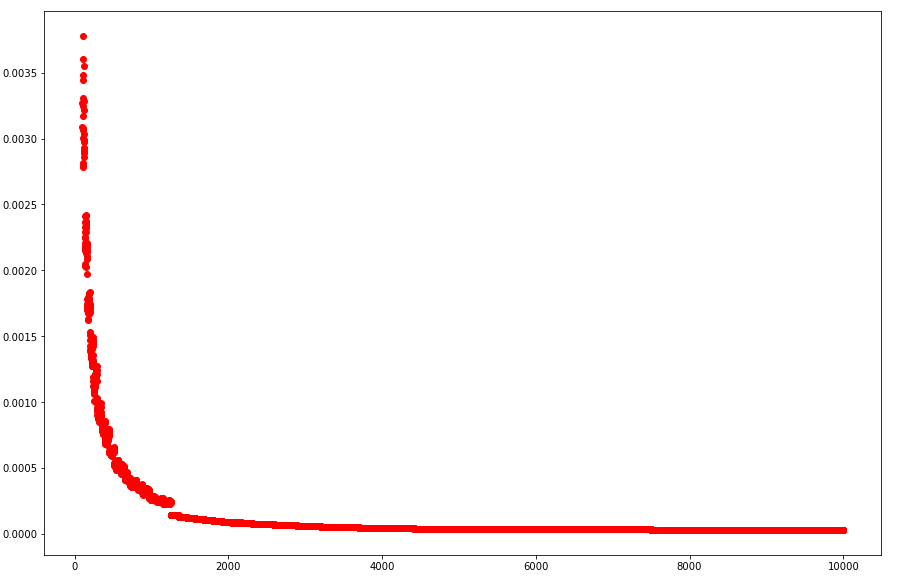
**Metoda miser:**

****

****

**Metoda vegas:**

****

****