

# [POP] Przeszukiwanie i optymalizacja Travelling Santa 2018 – Dokumentacja wstępna

Adam Kraś 325177, Krzysztof Król 325178

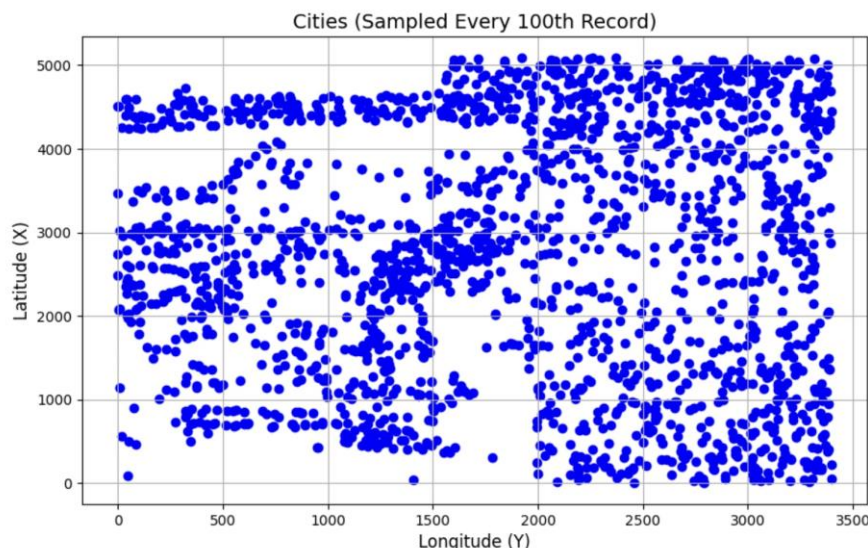
## 1. Cel projektu

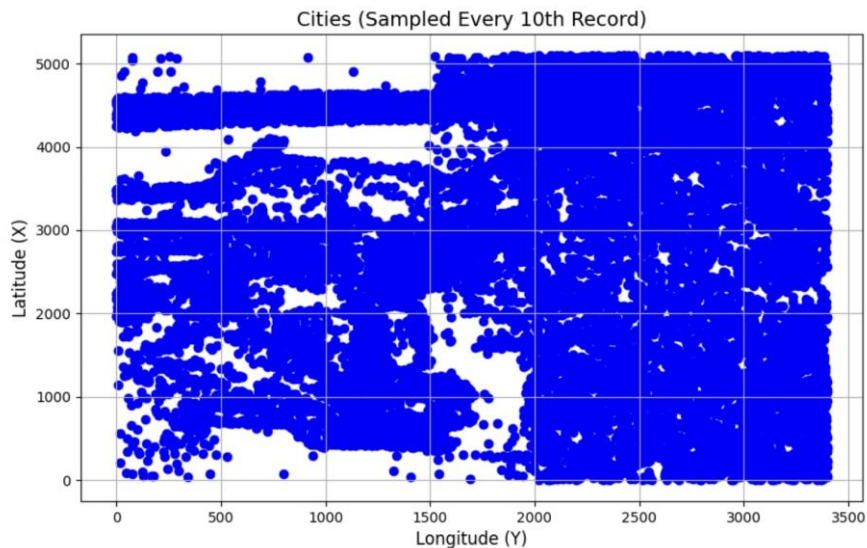
Celem projektu jest zoptymalizowanie trasy świętego Mikołaja, który musi odwiedzić wszystkie lokalizacje z punktów danych na stronie zadania i wrócić do punktu startowego. Problem w gruncie rzeczy jest pewną modyfikacją klasycznego problemu komiwojażera. Różnica polega na tym, iż w przypadku tego zadania, jeżeli co dziesiąte przejście z miasta do miasta, nie będzie z miasta o CityID, które jest liczbą pierwszą, to koszt tego przejścia będzie zwiększony o 10%, względem standardowej sytuacji.

## 2. Analiza danych

Nasze dane mają strukturę złożoną z 3 pól, CityID, X, Y. CityID to ID miasta, gdzie CityID = 0, to biegun północny, czyli wymagany punkt startowy i końcowy szukanej ścieżki. X i Y to natomiast szerokość i wysokość geograficzna. Największym wyzwaniem w implementacji tego zadania jest ilość punktów do odwiedzenia, których jest prawie 200000.

Wyniki próbnej analizy danych:





Jak widać na wykresach, punkty są względnie równolegle rozmieszczone

### 3. Wstępna propozycja rozwiązania

Po konsultacjach, podejście, które spróbujemy zastosować w pierwszej kolejności będzie zakładało zachowanie wszystkich punktów w naszych eksperymentach, które następnie podzielimy na mniejsze podgrupy punktów znajdujących się w określonym mniejszym obszarze. Możliwe to będzie przy pomocy struktury typu k-d tree. Eksperymentalnie znajdziemy ilu krotny podział będzie dawał najlepsze wyniki (czyli na jakiej wielkości obszary najlepiej będzie podzielić nasz zbiór danych), po czym dla każdego obszaru będziemy szukać najbardziej optymalnej trasy lokalnej przy pomocy algorytmu typu TSP (w tym momencie ciężko powiedzieć który konkretnie, czy np. wariant algorytmu ewolucyjnego czy może algorytm mrówkowy). Każda potencjalna trasa będzie reprezentowana jako permutacja miast z danego podobszaru. Następnie, gdy znajdziemy już trasy dla każdego z podobszarów, to przy pomocy analogicznego algorytmu dokonamy połączenia małych tras w jedną dużą trasę zawierającą wszystkie punkty. Zrobimy to korzystając z grafu centroidów, gdzie dany centroid będzie reprezentantem swojego małego obszaru (średnia współrzędnych punktów w danym obszarze), a następnie jeszcze raz rozwiążemy zadanie typu TSP, które zwróci nam ostateczne rozwiązanie.

### 4. Funkcja celu

Funkcja celu która będzie minimalizowana, to całkowita długość trasy z uwzględnieniem ewentualnych kar za niespełnienie warunków związanymi z liczbami pierwszymi. Tzn. koszt co dziesiątego przejścia będzie zwiększony o 10% jeżeli nie będzie wychodził z miasta o ID, które to jest liczbą pierwszą.

$$\sum_{i=1}^{n-1} d(C_i, C_{i+1}) + karaZaLiczbyNiepierwsze$$

Gdzie  $d(C_i, C_{i+1})$ - odległość Euklidesowa między miastami  $C_i$  i  $C_{i+1}$

$n$  – liczba miast do odwiedzenia

$karaZaLiczbyNiepierwsze$  – dodatkowy koszt zależny od tego, czy indeksy miast spełniały wymaganie bycia liczbą pierwszą