Wykorzystanie programu SAS do optymalizacji podróży

Projekt zaliczeniowy w ramach przedmiotu “SAS - Probabilistyczne i deterministyczne modele optymalizacji”

Kamil Morański (432426)

Weronika Golik (430228)

1. Opis problemu

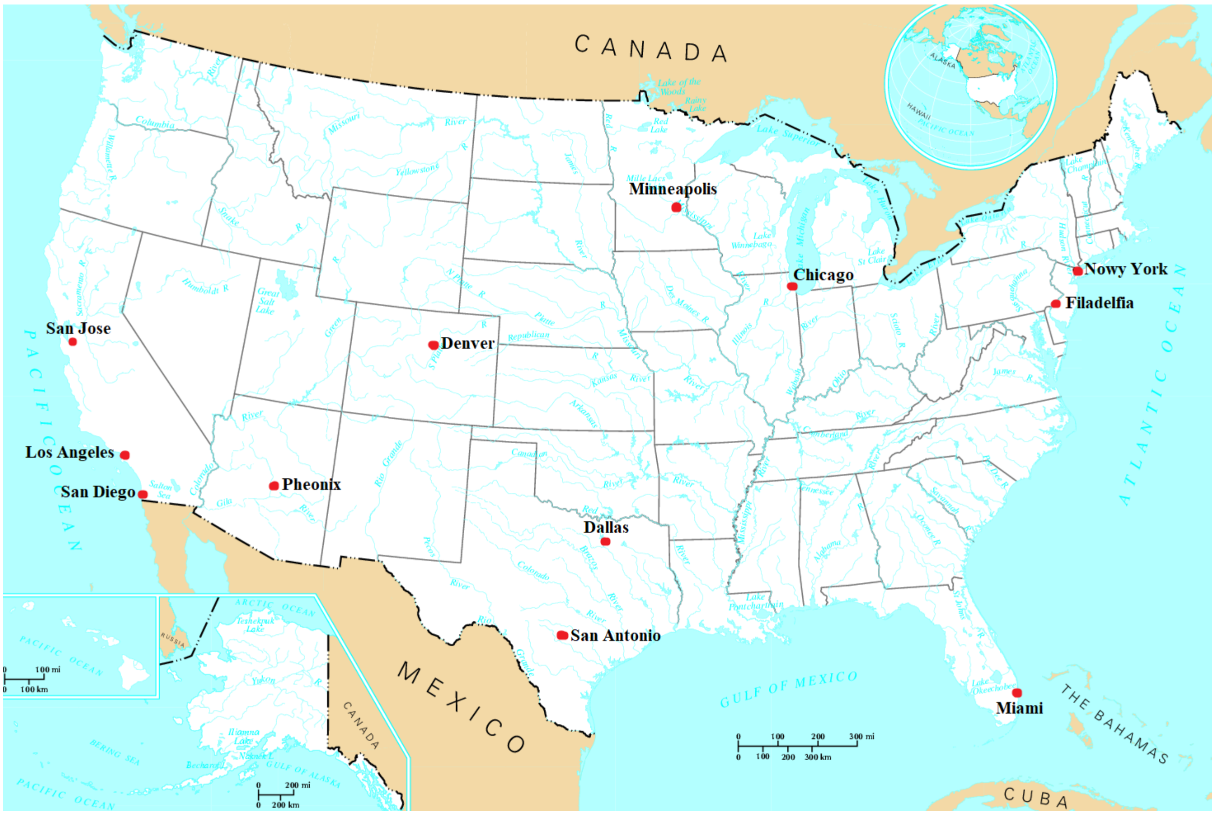
Często napotyka się sytuację gdy występuje potrzeba odwiedzenia wielu miejsc jednego po drugim. Z takim przypadkiem można spotkać się nawet w życiu codziennym, gdy w obrębie jednego miasta trzeba odwiedzić wiele miejsc w określonych sprawach. W takim wypadku bardzo przydatnym jest zbudowanie trasy lub kolejności, w której dane zadania zostaną wykonane. O ile poprzedni przykład był dość prosty i ewentualna nieefektywna trasa nie spowoduje dużych strat, o tyle istnieją przypadki gdy znalezienie optymalnego rozwiązania potrafi zaoszczędzić wiele czasu i środków. Mowa tutaj o np. trasach koncertowych lub kampaniach wyborczych gdzie osoba lub zespół, muszą odwiedzić wiele miejsc w określonym czasie. W tym przypadku koszty operacji znacznie wzrastają ze względu na transport czy też organizację wydarzeń. Określenie odpowiedniej trasy minimalizującej czas czy też koszty jest tutaj kluczowe i pozwala uzyskać dużo większy zysk, zmieniając tylko kolejność wizyt. Minimalizację czasu podróży można dodatkowo rozumieć w kontekście kosztu alternatywnego: im więcej dana osoba spędza czasu w podróży, tym mniej czasu może poświęcać na inne aktywności, które mogą przynosić zysk.

W tej pracy zajęto się nieco bardziej codziennym przykładem. Omawiany jest przypadek turysty, który chce zobaczyć określone miasta w Stanach Zjednoczonych. Zakładamy, że nie ma on ograniczeń finansowych, dlatego może podróżować dowolną linią lotniczą, lecz chce zminimalizować łączny czas podróży pomiędzy miastami.

2. Opis danych

Aby rozwiązać problem należy uzyskać dane na temat czasu podróży pomiędzy każdym z wybranych miast. W tym przypadku turysta chce zwiedzić 12 miast, czyli interesuje nas czas przelotu pomiędzy jednym miastem i 11 pozostałymi. Miasta do przeprowadzenia analizy wybrano w następujący sposób: wybrano 10 miast o największej populacji, po czym do próby dodano Minneapolis, Miami, Denver ze względu na oddalenie od innych miejsc wybranych w próbie oraz bliskość Parków Narodowych, które mogą stanowić atrakcję turystyczną dla turysty. Z próby wyłączono Houston, ponieważ miasto to znajduje się bardzo blisko innych metropolii w Texasie, czyli Dallas i San Antonio, przez co nie stanowi interesującej destynacji. Miasta zostały ukazane na grafice numer 1. Z Europy najłatwiej jest przylecieć do Nowego Yorku, dlatego to tutaj rozpocznie się przygoda turysty. Miasto, które zostanie zwiedzone jako ostatnie nie jest istotne.. Dodatkowo przyjęto kilka założeń, które ułatwiają analizę podróży. Zakłada się że turysta podróżuje jedynie połączeniami bezpośrednimi, są one najszybsze i najwygodniejsze. Nie wzięto pod uwagę odległości lotniska od miasta lub dodatkowego czasu na przyjechanie wcześniej na lotnisko przed lotem, każde lotnisko jest inne i wywołałoby to zbyt duże skomplikowanie w danych.

Grafika 1. Wybrane miasta w Stanach Zjednoczonych.



źródło: [*http://www.worldmap.pl/stany-zjednoczone/*](http://www.worldmap.pl/stany-zjednoczone/)

Dane zostały zaczerpnięte ze strony https://www.google.com/travel/flights, oraz ręcznie przepisane do tabeli w Microsoft Excel. Wyszukano dostępne loty pomiędzy każdymi dwoma miastami, brano pod uwagę największe lotniska. Daną którą wzięto pod uwagę jest czas lotu, zaokrąglono go z dokładnością do pół godziny. Dane przedstawiono w tabeli numer 1 oraz 2, de facto jest to jedna tabela, lecz ze względów edycyjnych została podzielona na dwie części. W pierwszej kolumnie wypisano miasta z których startuje dany lot, natomiast pierwszy wiersz zawiera cel podróży. Z oczywistych powodów student nie chce startować i lądować w tym samym mieście, dlatego przecięciu miast o tej samej nazwie widnieje kropka, którą należy rozumieć jako brak możliwości podróży.

Tabela 1 Czas przelotu w godzinach pomiędzy wybranymi miastami USA

	Nowy York	Los Angeles	Chicago	Minneapolis	Pheonix	Filadelfia
Nowy York	.	6	3,5	2,5	6	1
Los Angeles	6	.	4	4	1,5	5
Chicago	3,5	4	.	1,5	4	2
Minneapolis	2,5	4	1,5	.	3,5	2,5
Pheonix	6	1,5	4	3,5	.	4,5
Filadelfia	1	5	2	2,5	4,5	.
San Antonio	4	3	3	3	2	4
San Diego	6	1	4,5	4	1	6
Dallas	4	3	2,5	2,5	2	4
San Jose	6,5	1,5	5	4	2	6
Miami	3	5	3	4	4,5	3
Denver	4,5	2	2,5	2	2	4

Tabela 2 Czas przelotu w godzinach pomiędzy wybranymi miastami USA

	San Antonio	San Diego	Dallas	San Jose	Miami	Denver
Nowy York	4	6	4	6,5	3	4,5
Los Angeles	3	1	3	1,5	5	2
Chicago	3	4,5	2,5	5	3	2,5
Minneapolis	3	4	2,5	4	4	2
Pheonix	2	1	2	2	4,5	2
Filadelfia	4	6	4	6	3	4
San Antonio	.	3	1	4	3	2
San Diego	3	.	3	1,5	5	2,5
Dallas	1	3	.	4	3	2
San Jose	4	1,5	4	.	5,5	2,5
Miami	3	5	3	5,5	.	4,5
Denver	2	2,5	2	2,5	4,5	.

3. Model matematyczny

Przedstawiony w poprzednim rozdziale problem, tak jak większość zjawisk i procesów decyzyjnych można opisać za pomocą odpowiedniego modelu matematycznego. Poniżej zaprezentowano dwa podejścia do matematycznego modelowania optymalizacji czasu podróży.

3.1. Podejście I

Mamy n obiektów (miast przylotu, O), które mają być przydzielone w sposób jednoznaczny do m obiektów (miast wylotu, D), tzn. jeden obiekt Oj może być przydzielony do jednego obiektu Di i wszystkie obiekty muszą mieć “parę” (n=m). Dana jest macierz C=[cij], i=1,…,m, j=1,…,n, opisująca koszt (w naszym wypadku: czas) wynikający z połączeń obiektów. Zmienne zerojedynkowe xij określają, czy obiekty Di i Oj zostały połączone. Naszym celem jest minimalizacja kosztu (czasu) jaki turysta ponosi przy każdym przelocie. Matematycznie możemy zapisać funkcję celu w następujący sposób:

$$f(x) = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \rightarrow \max/\min$$

$$\sum_{j=1}^{n} x_{ij} \leq 1, \quad i = 1,\ldots, m$$

$$\sum_{i=1}^{m} x_{ij} = 1, \quad j = 1,\ldots, n$$

$$x_{ij} = \begin{cases} 0 \\ 1 \end{cases}$$

3.2. Podejście II

Możemy rozpatrzyć opisywany problem również jako zadanie transportowe. Przyjmijmy, że danych jest m dostawców (miast wylotu turysty, D) i n odbiorców (miast przylotu, O). Każdy dostawca Di posiada podaż Ai = 1 i każdy odbiorca Oj ma popyt Bj = 1.

Dana jest macierz C=[cij], i=1,…,m, j=1,…,n, określająca koszt lotu od i-tego dostawcy do j-tego odbiorcy. Zadanie jest zbilansowane, gdyż A = B.

$$A = \sum_{i=1}^{m} A_i \qquad\qquad B = \sum_{i=1}^{n} B_j$$

Ponownie chcemy minimalizować sumaryczny koszt (czas przelotu) turysty, za pomocą funkcji celu:

$$f(x) = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \rightarrow \max/\min$$

$$\sum_{j=1}^{n} x_{ij} \le a_i, \quad i = 1,...,m$$
$$\sum_{i=1}^{m} x_{ij} \ge b_j, \quad j = 1,...,n$$
$$x_{ij} \ge 0, \quad i = 1,...,m, \quad j = 1,...,n$$

4. Rozwiązanie problemu

Problem postanowiono rozwiązać za pomocą języka SAS, który oferuje wiele możliwości z zakresie optymalizacji. Ze względu na charakter omawianego przypadku, można rozwiązać go na dwa sposoby: Za pomocą procedury Assign, która przydziela obiekty nawzajem do siebie w zależności od warunków oraz za pomocą procedury Trans, która jest używana w zadaniach związanych z transportem. Każdy ze sposobów będzie wymagał indywidualnego podejścia i zmodyfikowania danych wejściowych w ten sposób, aby wynik miał odzwierciedlenie w realnym świecie.

4.1. Rozwiązanie za pomocą przydziału.

Pierwszym pomysłem na rozwiązanie problemu jest zastosowanie metody przydziału, za pomocą procedury Assign. Metoda ta pozwala na przyporządkowanie dokładnie jednego obiektu z listy do dokładnie jednego obiektu z drugiej listy, w ten sposób aby zminimalizować koszt ich połączenia. W przypadku turysty obie listy są takie same, znajdują się na nich miasta, jednak jedna lista określa miejsca wylotu, a druga miejsca lądowania. Po przydzieleniu wszystkich miast miejsce lądowania rozumie się jako miejsce wylotu w następnej “rundzie”, ponieważ turysta nie może się “teleportować” pomiędzy miastami. Kolejne punkty podróży są zapisywane, aż turysta odwiedzi wszystkie miejsca na liście. Uruchomiony kod SAS-owy znajduje się w załączniku 1, a jego wynik przedstawiono w tabeli 3:

Tabela 3:

Wylot	Przylot	Czas lotu
Nowy York	Filadelfia	1.0
Los Angeles	San Jose	1.5
Chicago	Miami	3.0
Minneapolis	Denver	2.0
Pheonix	San Diego	1.0
Filadelfia	Nowy York	1.0
San Antonio	Dallas	1.0
San Diego	Pheonix	1.0
Dallas	San Antonio	1.0
San Jose	Los Angeles	1.5
Miami	Chicago	3.0
Denver	Minneapolis	2.0
		19.0

Na tym etapie okazało się, że dane rozwiązanie jest kompletnie niepraktyczne. Niektóre miasta leżą od siebie na tyle blisko, że program przyporządkowuje je do siebie nawzajem. Biorąc za przykład tabelę 3, turysta odbywa lot z Nowego Yorku do Filadelfii, a potem z Fildadelfii do Nowego Yorku. W tym przypadku postanowiono zmodyfikować kod w ten sposób, aby uniemożliwiał powrót do miejsca w którym turysta już był. Druga wersja kodu znajduje się w załączniku numer 2, a jej wynik prezentuje tabela 4.

Tabela 4.

Wylot	Przylot	Czas lotu
Nowy York	Filadelfia	1.0
Los Angeles	San_Jose	1.5
Chicago	Miami	3.0
Minneapolis	Chicago	1.5
Pheonix	San_Diego	1.0
Filadelfia	Minneapolis	2.5
San Antonio	Dallas	1.0
San Diego	Los_Angeles	1.0
Dallas	Denver	2.0
San Jose	Nowy_York	6.5
Miami	San_Antonio	3.0
Denver	Pheonix	2.0
		26.0

W tej postaci kod zadziałał poprawnie, a żadne miasto nie powtarza się. Analizując tabelę 4, widać, że turysta wylatując z Nowego Yorku leci do Filadelfii, później z Filadelfii do Minneapolis itd. Ostatecznie lądując w San Jose, skąd poleciałby do Nowego Yorku, lecz tutaj nie jest to istotne, gdyż turysta zaczynał z tego miasta swoją podróż. Nie możemy też całkowicie wykluczyć możliwości podróży do Nowego Yorku, ponieważ wtedy program nie będzie miał gdzie przyporządkować tego miasta. Wynik funkcji celu to 26, a nie biorą pod uwagę lotu powrotnego do Nowego Jorku, jest to 20 godzin podróży samolotami, aby zwiedzić najważniejsze miasta Stanów Zjednoczonych.

4.2. Rozwiązanie za pomocą transportu

Innym sposobem rozwiązania omawianego problemu jest rozpatrzenie go jako klasyczne zadanie transportowe, w którym dostawcami są miasta wylotu turysty, zaś odbiorcami - miasta przylotu. Czas przelotu interpretujemy jako koszt. Zarówno podaż jak i popy w tym przypadku równe są 1. Zadanie transportowe jest szczególnym przypadkiem problemu przepływu w sieciach i może być zapisane jako szczególny przypadek zadania programowania liniowego. Za pomocą procedury TRANS wyznaczamy optymalne rozwiązanie. Zadanie jest zbilansowane, ponieważ liczba miast wylotu jest równa liczbie miast przylotu. Aby uniknąć powtarzania się miast, tak jak w poprzednim rozwiązaniu, kod napisano narzucając ograniczenia, które uniemożliwiają powrót do tego samego miasta. Kod SAS-owy znajduje się w załączniku nr 3, zaś wyniki procedury zostały przedstawione w tabeli numer 5:

Tabela 5:

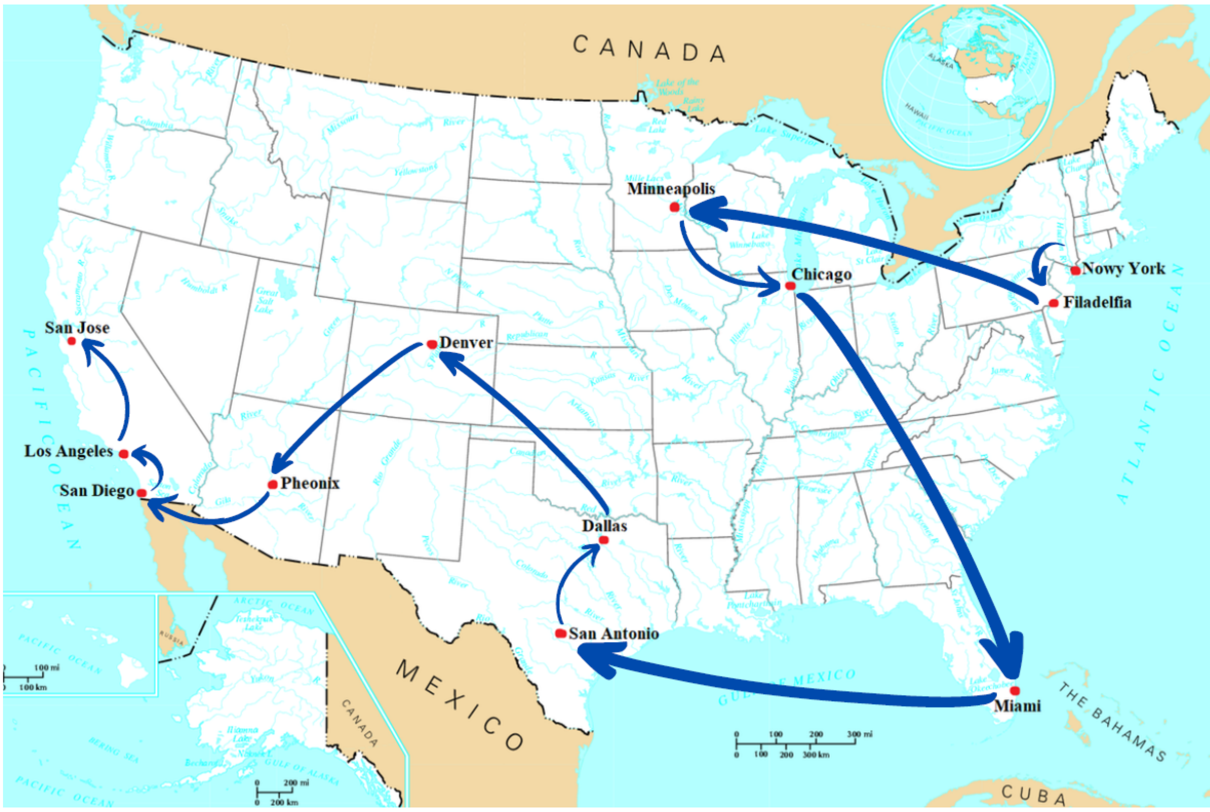
Obs.	City	sup	New_York	Los_Angeles	Chicago	Minneapolis	Pheonix	Filadelfia	San_Antonio	San_Diego	Dallas	San_Jose	Miami	Denver	_DUAL_
1	_DEMAND_	.	1.0	1.0	1	1.0	1	1	1	1	1	1	1	1	.
2	NY	1	.	0.0	0	0.0	0	1	0	0	0	0	0	0	-1.0
3	LA	1	0.0	.	0	0.0	0	0	0	0	0	1	0	0	-1.5
4	CHI	1	.	0.0	.	.	0	0	0	0	0	0	1	0	-1.0
5	MIN	1	.	0.0	1	.	0	0	0	0	0	0	0	0	-1.5
6	PHE	1	.	0.0	0	0.0	.	0	0	1	0	0	0	0	-1.0
7	FIL	1	.	0.0	0	1.0	0	.	0	0	0	0	0	0	-1.0
8	SA	1	.	0.0	0	0.0	0	0	.	0	1	0	0	0	-1.0
9	SD	1	0.0	1.0	0	0.0	.	0	0	.	0	0	0	0	-1.5
10	DAL	1	.	0.0	0	0.0	0	0	.	0	.	0	0	1	-1.0
11	SJ	1	1.0	0.0	0	0.0	0	0	0	.	0	.	0	0	-1.0
12	MIA	1	.	0.0	0	0.0	0	0	1	0	0	0	.	0	0.0
13	DEN	1	.	0.0	0	.	1	0	0	0	0	0	0	.	-1.0
14	_DUAL_	.	7.5	2.5	3	3.5	3	2	3	2	2	3	4	3	.

W powyższej tabeli jedynką zostały oznaczone komórki których nazwa wiersza jest miastem wylotu, zaś nazwa kolumny - miastem do którego leci turysta. Optymalna wartość funkcji celu wynosi 26. Po odjęciu 6-cio godzinnej podróży z San Jose do Nowego Jorku, czas spędzony przez turystę w podróży wyniesie 20 godzin. Na podstawie tabeli możemy rozpisać trasę wyznaczoną przez procedurę Trans. Okazuje się, że trasa ta pokrywa się z tą, wyznaczoną za pomocą procedury Assign. Turysta zwiedzi miasta w następującej kolejności: Nowy Jork - Filadelfia - Minneapolis - Chicago - Miami - San Antonio - Dallas - Denver - Phoenix - San Diego - Los Angeles - San Jose.

5. Wyniki

Przedstawiony problem optymalizacji czasu podróży został rozwiązany przy pomocy dwóch podejść matematycznych oraz odpowiadających im procedur SAS-owych: Assign i Trans. Obydwa podejścia doprowadziły do identycznych rezultatów. Możemy zatem wnioskować o istnieniu jednego najbardziej optymalnego rozwiązania, dla którego wartość funkcji celu wynosi 26, zaś trasa, jaką pokona turysta – student jest taka, jak na grafice nr 2 przedstawionej poniżej.

Grafika 2 Trasa turysty ustalona za pomocą metody przydziału



Na grafice numer 2 przedstawiono trasę, którą pokonał turysta. Ostatecznie całość podróży z metą w San Jose trwała 20 godzin, co można uznać za dobry wynik zważywszy na fakt, że niektóre loty w danych wejściowych trwały nawet po 6 i pół godziny, a lotów odbyło się aż 11. Najdłuższe z nich trwały po 3 godziny i były to loty z Chicago do Miami, oraz z Miami do San Antonio, co widać po oddaleniu tego miasta od reszty. Najkrótsze zaś loty odbyły się na trasach: Nowy Jork - Filadelfia, Phoenix - San Diego, San Antonio - Dallas oraz San Diego - Los Angeles.

Jak widać, wykorzystany w owej pracy kod SAS-owy pozwala na zoptymalizowanie czasu podróży. Przez wprowadzanie różnych danych może być ponownie wykorzystywany w celu efektywnego planowania podróży. Za pomocą SAS-a stworzono pomocne narzędzie o szerokim zastosowaniu.