# Optimal European Trip

John Gilbertson, Shian Su, Ria Szeredi and Kenneth Young

## Abstract

Planning a trip to Europe is not a trivial task. It bears similarity to the Travelling Salesperson and Knapsack Problems. We formulate an integer linear program to optimise travel costs and enjoyment. The implementation of the model is assisted by greedy heuristics to improve computational time. Our model is extended to investigate several real-world scenarios. Methods to overcome nonlinearity are also explored. This model can be used to assist travel agencies or any prospective traveller planning a trip to Europe. It has the potential to be built into a website where the traveller can freely plan their dream European holiday.

Keywords: personalised tourist guide, linear programming, transportation planning, trip generation, decision making

# Contents

# 1 Introduction

Planning a trip to Europe is not a trivial task as there are many factors which must be considered. For example, comparing the cost of airfares, the cost of hotels, deciding on which places to visit and the order in which to visit them. Prospective travellers seek the help of travel agencies to assist them in finding their optimal trip. However, using travel agencies comes with a fee and many are associated with certain suppliers. Hence the results they provide their customers are biased and not necessarily the optimal solution.

With this project we aim to formulate and create a linear program which optimises a trip to Europe. What classifies the optimality of the trip will be defined by the traveller; whether it is minimal cost, maximal enjoyment or a combination of the two. This LP could be built into a website where travellers can input their requirements, such as maximum trip length, longest and shortest time spent in any one city and a budget constraint. The LP seeks only optimality and so would have no bias towards particular airlines or hotels; a feature which would certainly be appealing to any traveller. One who has already planned their trip could use the website to estimate costs or even find a cheaper or more enjoyable configuration.

To model this program we require data on the daily cost of living for each city, the cost of airfares between all cities and the daily utility of staying at each city. The first models we consider are greedy heuristics and from their output we motivate the use of a formal linear programming approach. Two integer linear programs are then proposed: one simply seeking the minimal cost and the other seeking the minimal cost given that enjoyment must first be maximised.

The model is extended in various ways to increase its applicability in real-world scenarios. We consider travellers who have an affinity for particular acitvities, such as going to the beach or visiting museums, and use our program to maximise their enjoyment. The model is also extended to handle trips involving multiple people. One person may want to join/avoid a friend/enemy who has already planned their trip to Europe. A program formulation which can plan a trip involving many people at once is also proposed. This LP can be used to ensure that everyone is always with at least one other person and the total enjoyment is maximised.

Finally, we present our concluding remarks on the project and give recommendations to possible further work which could be done.

# 2 Data

In this section we present data on the airfares between all cities, the daily living expenses of any city, the cost of flights to and from Melbourne for any city and the base utility of staying at a city.

For the sake of brevity, we shall only provide data sets for the case of five cities here. This will be enough to demonstrate how we stored and used our data. The full data set can be found in the appendix. All values are in Australian Dollars.

Our model will need to know the flight costs between all the cities. Initially we assume that flights are uniform across all days of the trip and at all times of each day. Our flight data is sourced from `momondo.com.au` [1]. We select only the cheapest flights between any two cities.

| City | Moscow | Paris | London | Madrid | Rome |
|---|---|---|---|---|---|
| **Moscow** | 0 | 146 | 126 | 202 | 146 |
| **Paris** | 227 | 0 | 60 | 143 | 93 |
| **London** | 213 | 82 | 0 | 249 | 160 |
| **Madrid** | 188 | 86 | 136 | 0 | 144 |
| **Rome** | 223 | 80 | 125 | 146 | 0 |

Table 1: Airfares between Five Cities

For simplicity we assume that a person can fly directly between all cities, even though in some cases the cheapest flight is not direct.

Aside from the cost of airfares, the cost of living is another contributor to our model's objective value. All our daily living expenses data are sourced from `budgetyourtrip.com` [2]. To simplify the computation of our daily costs, we assume that there is no reduction in cost if a traveller books accommodation at a single hotel over multiple nights.

| City | Moscow | Paris | London | Madrid | Rome |
|---|---|---|---|---|---|
| **Low** | 36 | 81 | 110 | 57 | 68 |
| **Mid** | 92 | 223 | 298 | 148 | 169 |
| **High** | 233 | 657 | 845 | 393 | 423 |

Table 2: Daily Costs of Five Cities

The last contribution to our model's objective value comes from the flights from and to Melbourne at the begining and end of the trip. We assume that the traveller buys two one-way tickets for these flights as it results in a simpler model. This flight data is again sourced from `momondo.com.au` [1].

We assume that all flights occur in the morning. We make this simplification so we do not have to calculate fractional daily costs. Further to this, we assume that the return flight to Melbourne occurrs on the morning after the last day of the trip. This is again to simplify the calculation of daily costs. If the traveller leaves a city in the early morning then we assume that there is no living expenses for that day.

| City | Moscow | Paris | London | Madrid | Rome |
|---|---|---|---|---|---|
| **To Melbourne** | 1390 | 1090 | 1007 | 1175 | 1082 |
| **From Melbourne** | 803 | 844 | 875 | 1061 | 934 |

Table 3: Melbourne Flight Costs of Five Cities

| Moscow | Paris | London | Madrid | Rome |
|---|---|---|---|---|
| 71 | 93 | 100 | 77 | 85 |

Table 4: Base Utilities of Five Cities

The true utility of staying in a city is highly subjective and very difficult to measure. For a basic model our utility is based on the overall popularity of a city as measured by a "bednights" statistic [5], which measures the number of nights tourists stayed in a given city. Because the difference in the raw data was quite large, the fifth root was taken such that all values were of similar magnitude and then rescaled to be a percentages of the highest utility city. This reflects the behaviour of a naive traveller who makes their decisions on general popularity. In Section 4.2 we explore specific preferences.

# 3 Solution Methodology

## 3.1 Heuristics

This problem has many similarities with the Travelling Salesman Problem (TSP), which is known not to be computationally efficient to solve. To avoid this issue, it can be useful to solve the problem using a greedy heuristic.

### 3.1.1 Cheap Heuristic

The cheap heuristic attempts to find the cheapest 15 day trip with 15 possible cities. Let $\alpha$ denote the maximum number of days allowed in any one city. For each city, the heuristic calculates the cost of travelling from Melbourne to that city and remaining in that city for $\alpha$ days. The algorithm then chooses the city that has the minimum cost and requires the traveller to remain there for $\alpha$ days. In the same way, cities are added iteratively until there are no days left. The traveller then flies back to Melbourne from their final city.

**Algorithm 1** Cheap Heuristic

```
Begin in Melbourne
for each city i do
    cost(i) = costFromMelb(i) + α × costDaily(i)
end for
Go to city i with the minimum cost and stay for α days
daysLeft = days − α
while daysLeft > 0 do
    step = min(α, daysLeft)
    for each city i not yet visited do
        cost(i) = costTravel(currentCity, i) + step × costDaily(i)
    end for
    Go to city i with the minimum cost and stay for step days
    Decrement daysLeft by step
end while
Return to Melbourne from final city
```

Using $\alpha = 4$ and the medium daily costs defined in Section 2, the output of this heuristic is shown in Table 5. The heuristic solution is to spend the first 4 days in Istanbul, then fly to Moscow for 4 days, followed by Prague for 4 days and finally Venice for 3 days. The cost of this trip is \$4,060 including Melbourne flights, flights within Europe and daily costs. The utility of this trip is 1,102.

### 3.1.2 Maximum Utility Heuristic

The maximum utility heuristic attempts to find the 15 day trip with the maximum utility regardless of cost. At each iteration, the algorithm chooses the city with the maximum

| Days | # of Days | City |
|---|---|---|
| 1-4 | 4 | Istanbul |
| 5-8 | 4 | Moscow |
| 9-12 | 4 | Prague |
| 13-15 | 3 | Venice |
| **Total Cost ($)** | | 4060 |
| **Total Utility** | | 1102 |

Table 5: Cheap Heuristic

utility from the set of all cities. The utility of staying another day in the chosen city is multiplied by a decay factor. This will be further explained in Section 3.2.2. This heuristic is strikingly similar to the knapsack algorithm.

**Algorithm 2** Maximum Utility Heuristic

```
Begin in Melbourne
for j = 1, ..., days do
  Go to city i ∈ C with the maximum utility
  Reduce the utility of the current city by the decay factor
end for
Sort cities in alphabetical order
Return to Melbourne from final city
```

We sort the cities in alphabetical order so that we do not visit a city twice. The order is arbitrary.

Using a decay factor of 0.9 and the medium daily costs defined in Section 2, the output of this heuristic is shown in Table 6. The utility of this trip is $1,238$, which is 12% greater than the utility of the trip found using the cheap heuristic. The cost of this trip is \$5,636, which is \$1,576 greater than the cheap heuristic.

| Days | # of Days | City |
|---|---|---|
| 1 | 1 | Barcelona |
| 2-3 | 2 | Berlin |
| 4 | 1 | Istanbul |
| 5-7 | 3 | London |
| 8 | 1 | Madrid |
| 9-10 | 2 | Paris |
| 11 | 1 | Prague |
| 12-13 | 2 | Rome |
| 14 | 1 | Venice |
| 15 | 1 | Vienna |
| **Total Cost ($)** | | 5636 |
| **Total Utility** | | 1238 |

Table 6: Maximum Utility Heuristic

### 3.2 Integer Linear Program (ILP)

#### 3.2.1 Base Model

In this section we introduced our base model which minimises the cost of a trip to Europe.

**Variables**
$x_{id}$: [binary] equal to 1 if the traveller is in city $i$ on day $d$.
$y_i$: [binary] equal to 1 if the traveller ever visits city $i$.
$m_{ijd}$: [binary] equal to 1 if the traveller moves from city $i$ to city $j$ on day $d$.

**Parameters**
$D$: the total number of days.
$\mathcal{C}$: the set of cities available for visiting.
$\mu_{ij}$: the cost to move from city $i$ to city $j$.
$\lambda_i$: the cost to move from Melbourne to city $i$.
$\nu_i$: the cost to move from city $i$ to Melbourne.
$\delta_i$: the daily cost of city $i$.
$\alpha$: the minimum number of days allowed in a city.
$\omega$: the maximum number of days allowed in a city.

$$\text{Min} \ \sum_{d=1}^{D}\sum_{i\in\mathcal{C}} \delta_i \cdot x_{id} + \sum_{d=1}^{D-1}\sum_{i\in\mathcal{C}}\sum_{j\in\mathcal{C}} \mu_{ij}\cdot m_{ijd} + \sum_{i\in\mathcal{C}} (x_{i,1}\cdot\lambda_i + x_{i,D}\cdot\nu_i)$$

$$\text{s.t} \quad \sum_{d=1}^{D} x_{id} \geq \alpha\cdot y_i \qquad \forall i\in\mathcal{C} \tag{1}$$

$$\sum_{d=1}^{D} x_{id} \leq \omega\cdot y_i \qquad \forall i\in\mathcal{C} \tag{2}$$

$$x_{id} + x_{j(d+1)} \leq 1 + m_{ijd} \qquad \forall i,j\in\mathcal{C}, d=1..(D-1) \tag{3}$$

$$\sum_{i\in\mathcal{C}} x_{id} = 1 \qquad d=1..D \tag{4}$$

$$\sum_{i\in\mathcal{C}}\sum_{j\in\mathcal{C}} m_{ijd} = 1 \qquad d=1..(D-1) \tag{5}$$

$$\sum_{d=1}^{D-1}\sum_{j\in\mathcal{C}} m_{ijd} \leq 1 + \sum_{d=1}^{D-1} m_{iid} \qquad \forall i\in\mathcal{C} \tag{6}$$

$$x_{j1} + \sum_{d=1}^{D-1}\sum_{i\in\mathcal{C}} m_{ijd} \leq 1 + \sum_{d=1}^{D-1} m_{jjd} \qquad \forall j\in\mathcal{C} \tag{7}$$

$$x_{id}, y_i, m_{ijd} \in \{0,1\} \qquad \forall i,j\in\mathcal{C}, d=1..D \tag{8}$$

The components of the objective function are the daily expenses, travel costs between cities, and the travel costs to and from Melbourne.

| Constraint | Explanation |
|---|---|
| 1 | If the traveller visits city $i$, he must stay there for at least the minimum number of days allowed. |
| 2 | If the traveller visits city $i$, he must stay there for no more than the maximum number of days allowed. |
| 3 | The variable $m_{ijd}$ is forced to be 1 if the traveller is in city $i$ on day $d$ and in city $j$ on day $d+1$. |
| 4 | On any given day, the traveller must be in exactly one city. |
| 5 | On any given day, the traveller must move exactly once (including a move to the same city). |
| 6 | The traveller can only leave a visited city once. |
| 7 | The traveller can only enter a visited city once. This together with constraint 6 ensures the traveller does not return to an already visited city. |

The base model is a simple minimisation of cost during a trip through Europe. It is clear that the optimal solution would be to simply fly to the cheapest city and stay there for the duration of the planned trip. Thus, we impose constraints 1 and 2 to set limits on how long we can stay in a particular city.

There are a number of simple extensions we can add to the base model. Suppose the traveller wishes to spend a total of 4 days in Paris (city 2). We would require the constraint:

$$\sum_{d=1}^{D} x_{2,d} = 4$$

If the traveller had a flight already booked from London (city 3) to Berlin (city 8) on day 5, and needed to plan the rest of the trip around this flight, we would add the constraint:

$$m_{3,8,5} = 1$$

Similarly, suppose there was a football match in Madrid (city 4) on day 8 that the traveller wanted to go to. Then we would add:

$$x_{4,8} = 1$$

If the traveller does not want to visit Istanbul (city 13) then we can add the following constraint:

$$y_{13} = 0$$

### 3.2.2 Decaying Enjoyment

We explore the assumption that the enjoyment of staying in a city decays with the number of days stayed. In its natural form this does not yield a problem that is linear in our main decision variable $x_{id}$.

Let $u$ be some base level utility and $r$ be some decay factor. Let $N_i = \sum_{d=1}^{D} x_{id}$ be the total number of days we stay in city $i$. To find out how much utility we would obtain for a given city we would need to compute the sum

$$\sum_{d=1}^{N_i} u \cdot r^{d-1}, \quad i \in \mathcal{C}.$$

This is a basic geometric sum with closed form $u \cdot \frac{1-r^{N_i}}{1-r}$ which is clearly not linear in $x_{id}$. We can however linearise this problem by precomputing the utility values as a matrix for each day in each city. This is only possible because we are working in discrete time steps. With this linearisation method in mind, we add the following to the model.

**Variables**

$s_{id}$: [binary] equal to 1 if traveller stays in city $i$ for $d$ days.

**Parameters**

$u_{id}$: The total utility of staying in city $i$ for $d$ days.

We now no longer need constraint (2) from the base model since we expect that the traveller naturally leaves the city once the marginal enjoyment is low enough. We retain constraint (1) since it is still reasonable to set a length of minimum stay.

The new objective function is

$$\text{Max} \sum_{i \in \mathcal{C}} \sum_{d=1}^{D} s_{id} \cdot u_{id}$$

with additional constraints

$$\sum_{d=1}^{D} s_{id} = y_i \qquad \forall i \in \mathcal{C} \tag{9}$$

$$s_{ig} \cdot g \leq \sum_{d=1}^{D} x_{id} \qquad \forall i \in \mathcal{C}, g = 1...D \tag{10}$$

| Constraint | Explanation |
|---|---|
| 9 | This ensures that if $y_i$ =1 then one $s_{id}$ is equal to 1. |
| 10 | This ensures $s_{id} = 1$ if we stay in city $i$ for $d$ days. |

Once the optimal enjoyment has been found we then optimise again to minimise cost using the following objective function.

$$\text{Min} \sum_{d=1}^{D} \sum_{i \in \mathcal{C}} \delta_i \cdot x_{id} + \sum_{d=1}^{D-1} \sum_{i \in \mathcal{C}} \sum_{j \in \mathcal{C}} \mu_{ij} \cdot m_{ijd} + \sum_{i \in \mathcal{C}} (x_{i,1} \cdot \lambda_i + x_{i,D} \cdot \nu_i)$$

With the constraint that the utility is equal to the optimal objective value previously found:

$$\sum_{i \in \mathcal{C}} \sum_{d=1}^{D} s_{id} \cdot u_{ij} = \left( \sum_{i \in \mathcal{C}} \sum_{d=1}^{D} s_{id} \cdot u_{id} \right)_{optimal}$$

In practice, solving the model in FICO Xpress takes a substantial amount of time as the number of cities and days of the trip grows. Therefore we impose some additional constraints.

1. Lower bound on the utility: First we note that with unconstrained budget, the utility is non-decreasing with the trip duration, eg. the optimal utility of an eight day trip is always less than that of a nine day trip. Thus we can solve a shorter trip and use its optimal utility as a lower bound for the utility in the longer trip.

2. Upper bound on the utility: The previous heuristic from Section 3.1.2 gives maximal utility if we ignore the minimum days constraint. This will be an upper bound for our model.

3. Optimal duration in each city: For an optimal utility, the number of days in each city will not change in the cost optimisation. Therefore we can use this as a constraint for improving the performance of the cost optimisation:

$$\sum_{d=1}^{D} s_{id} = \left( \sum_{d=1}^{D} s_{id} \right)_{optimal} \qquad \forall i \in C.$$

### 3.2.3 Decaying Utility Results

We first run the model using a decay factor of 0.9 (i.e 10% rate of decay per day) with the constraint that we stay at least 2 days in any city that we visit. Below is the trip that maximises the utility for 15 days under such conditions.

| Days | # of Days | City |
|---|---|---|
| 1-2 | 2 | Rome |
| 3-4 | 2 | Barcelona |
| 5-6 | 2 | Venice |
| 7-10 | 3 | London |
| 11-12 | 2 | Berlin |
| 13-15 | 2 | Paris |
| **Total Cost ($)** | | 5195 |
| **Total Utility** | | 1219 |

Table 7: Trip plan with optimised decaying utility for 15 days

We explore some different rates of utility decay, this will affect the duration of stay in each city.

| Days | # of Days | City |
|---|---|---|
| 1-4 | 4 | London |
| 5-6 | 2 | Barcelona |
| 7-8 | 2 | Rome |
| 9-11 | 3 | Berlin |
| 12-15 | 4 | Paris |

Table 8: 0.95 Decay Factor

| Days | # of Days | City |
|---|---|---|
| 1-8 | 8 | London |
| 9-10 | 2 | Berlin |
| 11-15 | 5 | Paris |

Table 9: 0.98 Decay Factor

We see that as one would expect, slower rates of utility decay lead to longer staying durations in the cities that are visited. This allows for a more natural control of stay durations than simply setting maximum and minimum days. For longer trips it may make sense to set a slower rate of utility decay. Given that the model precomputes the utilities for each day in each city, it is simple to alter the way that utility is calculated. A different rate of change may be derived for each city based on statistics on average lengh of stay and more complex models may be used to define the utility without the need to alter the ILP.

Also of interest is how the model reacts to a limited budget, $\beta$. We wish to see how the traveller adapts to not being able to follow the trip with the optimised utility. To do this we add another constraint to our model:

$$\sum_{d=1}^{D}\sum_{i\in\mathcal{C}}\delta_i \cdot x_{id} + \sum_{d=1}^{D-1}\sum_{i\in\mathcal{C}}\sum_{j\in\mathcal{C}}\mu_{ij}\cdot m_{ijd} + \sum_{i\in\mathcal{C}}\left(x_{i,1}\cdot\lambda_i + x_{i,D}\cdot\nu_i\right) \leq \beta$$

Recall that for a decay factor of 0.9 we had a trip which optimised utility at 1219 and a cost of $5195. We see in Table 10 the new optimal trip with a budget constraint of $4500.

| Days | # of Days | City |
|---|---|---|
| 1-2 | 2 | London |
| 3-5 | 3 | Paris |
| 6-9 | 4 | Berlin |
| 10-15 | 6 | Istanbul |
| **Total Cost ($)** | | 4341 |
| **Total Utility** | | 1102 |

Table 10: Trip plan with decay factor of 0.9 and budget of $4500 for 15 days

Upon inspection of the data, it would appear that Istanbul has a high $\frac{\text{utility}}{\text{daily cost}}$ ratio and also offers a very cheap flight back to Melbourne, then it is no surprise that in many of the budget constrained scenarios we end up with significant time spent in Istanbul.

# 4 Extensions and Results Discussion

Currently our base model takes limited input from the propective traveller; only the number of days of the trip and the minimum and maximum number of days spent in any one city. To make the model more applicable in real world scenarios we propose some extensions which can handle more varied inputs.

## 4.1 Varying Airfares

So far, we have assumed that the cost of flights was uniform across all days. We can ensure our model returns more realistic results by considering how the cost of airfares vary across the week. We use `momondo.com.au` again to source the airfare data [1]. For the sake of computational efficiency, we arbitrarily set the first day of the trip to Sunday.

To model the variation in airfares, the travel cost parameter $\mu_{ij}$ needs an additional dimension to store the day of the week. So we now redefine $\mu$ as follows:

$\mu_{ijd}$: The cost to fly from city $i$ to city $j$ on day $d$, $\forall i, j \in \mathcal{C}$, $d = 1...7$.

If there are $n$ cities, then we now have a 3-dimensional array of size $n \times n \times 7$. To define a 3-dimensional array in the Mosel language is rather inelegant. In the case of 15 cities, the parameter definition Section becomes longer than the model definition itself.

Seeking an alternate and more compact implementation, we take the current flight data, $\mu_{ij}$, as the base price to travel between any two cities. We then construct a new parameter which stores how the flight cost is augmented for each day of the week. So now, for each of the cities, we have a size $n \times 7$ array which stores these augmentation values as percentages:

$a_{ijd}$: The percentage increase in airfares from city $i$ to city $j$ on day $d$, $\forall i, j \in \mathcal{C}$, $d = 1...7$.

To demonstrate this we provide Moscow's augmentation array, $a_{1jd}$. An entry of zero indicates that there is no increase in price for the flight on that day. The full table can be found in the appendix.

| City | Sun | Mon | Tue | Wed | Thu | Fri | Sat |
|---|---|---|---|---|---|---|---|
| **Moscow** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Paris** | 4 | 1 | 0 | 3 | 8 | 13 | 10 |
| **London** | 11 | 3 | 0 | 2 | 6 | 8 | 6 |
| **Madrid** | 10 | 3 | 0 | 6 | 13 | 17 | 14 |
| **Rome** | 4 | 1 | 0 | 3 | 11 | 12 | 12 |

Table 11: Airfare Augmentation Data for Moscow

Now adding this to the objective function gives

$$\text{Min} \sum_{d=1}^{D}\sum_{i\in\mathcal{C}}\delta_i\cdot x_{id}+\sum_{d=1}^{D-1}\sum_{i\in\mathcal{C}}\sum_{j\in\mathcal{C}}m_{ijd}\cdot\left(\mu_{ij}+\frac{\mu_{ij}\cdot a_{ij(1+\ d \bmod 7)}}{100}\right)+\sum_{i\in\mathcal{C}}(x_{i,1}\cdot\lambda_i+x_{i,D}\cdot\nu_i)$$

The modulus function is used here in the subscript of the augmentation array. This is clearly so that on the eighth and later days of the trip we correctly access the corresponding element of the array. We divide by 100 here to convert from percentages.

As an illustration, now consider a traveller planning a 13 day trip, who needs to fly from Florence to Moscow on the first Sunday of their trip (day 7). They also want to spend the last day of their trip in Crete. Adding the following constraints imposes the traveller's desires on the model,

$$m_{10,1,7} = 1$$
$$x_{6,13} = 1.$$

Below, we list the results from the base model and varying airfares extension.

| **Days** | **# of Days** | **City** |
|---|---|---|
| 1-4 | 4 | Istanbul |
| 5-6 | 2 | Prague |
| 7-8 | 2 | Florence |
| 9-11 | 3 | Moscow |
| 12-13 | 2 | Crete |
| **Total Cost ($)** | | 4253 |

Table 12: Base Model Output

| **Days** | **# of Days** | **City** |
|---|---|---|
| 1-4 | 4 | Istanbul |
| 5-6 | 2 | Prague |
| 7-8 | 2 | Florence |
| 9-11 | 3 | Moscow |
| 12-13 | 2 | Crete |
| **Total Cost ($)** | | 5715.16 |

Table 13: Varying Airfares Output

We can see that both models have returned the same trip, although their minimal costs vary greatly. The base model returns a total cost of $4253. Whereas with the variable airfare data added to the model, the minimum cost increases by $1462.16 (34.38%) to $5715.16. This cost increase primarily comes from the 39% increase in flight costs from Florence to Moscow on a Sunday and a significant 249.56% increase in flight cost from Crete to Melbourne on a Friday. Removing the assumption that flight costs are uniform across all days has shown that our base model's cost estimate is approximately $1500 lower than this extension's more accurate estimation.

## 4.2 Criteria Preference

We now consider a more realistic representation of utility. We derive a new measure of utility based on a traveller's enjoyment of various characteristics of a city. For the purposes of differentiating this from the previously mentioned utility we define this measure as enjoyment.

We can extend the base model by calculating the traveller's enjoyment based on various criteria, such as beaches, sport and architecture. For each criteria, we give each city a rating out of 10. Let $\kappa_{ic}$ denote the rating for criteria $c$ for city $i$. The ratings are displayed in Table 31 in the appendix. These ratings were gathered from various online sources such as `tripadvisor.com.au` [3] and `hostelworld.com` [4].

The traveller then specifies a rating out of 10 for each of the criteria, as well as a budget for the entire trip. Let $\rho_c$ denote the traveller's rating for criteria $c$ and let $\beta$ denote the budget. We assume that the traveller's enjoyment is proportional to the product of the criteria rating, the city rating and the number of days spent in that city. Our objective function becomes:

$$\text{Max} \sum_{d=1}^{D} \sum_{i \in \mathcal{C}} \sum_{c=1}^{8} \rho_c \cdot \kappa_{ic} \cdot x_{id}$$

As in Section 3.2.2, we need a budget constraint:

$$\sum_{d=1}^{D} \sum_{i \in \mathcal{C}} \delta_i \cdot x_{id} + \sum_{d=1}^{D-1} \sum_{i \in \mathcal{C}} \sum_{j \in \mathcal{C}} \mu_{ij} \cdot m_{ijd} + \sum_{i \in \mathcal{C}} (x_{i,1} \cdot \lambda_i + x_{i,D} \cdot \nu_i) \leq \beta$$

Suppose the traveller wishes to travel for 15 days on a \$6,000 budget and their criteria ratings are:

| **Beach** | **History** | **Museums** | **Architecture** | **Nightlife** | **Food** | **Sport** | **Theatre** |
|---|---|---|---|---|---|---|---|
| 10 | 0 | 1 | 2 | 10 | 1 | 2 | 1 |

Table 14: Criteria ratings

That is, beaches and nightlife are very important to the traveller compared to all other criteria. The model yields the traveller's optimal trip displayed in Table 15. The cost of the trip is \$5,329. As expected, Barcelona and Crete, with beach ratings of 8 and 10 respectively, are included in the trip, as well as London and Amsterdam, which both have nightlife ratings of 9.

| **Day** | **# of Days** | **City** |
|---|---|---|
| **1-4** | 4 | Barcelona |
| **5-8** | 4 | Crete |
| **9-11** | 3 | Amsterdam |
| **12-15** | 4 | London |

Table 15: 15 day trip maximising criteria utility

We can now set the enjoyment to be fixed at the optimal value of this trip, and solve a new problem which minimises cost. This results in the trip in Table 16 with a new cost of \$5,044. The number of days in each city remains the same while the order in which the cities are visited is changed to minimise the total travel cost.

| Day | # of Days | City |
|---|---|---|
| **1-4** | 4 | London |
| **5-8** | 4 | Barcelona |
| **9-12** | 4 | Crete |
| **13-15** | 3 | Amsterdam |

Table 16: 15 day trip maximising criteria utility, minimising cost

## 4.3 Two Travellers

### 4.3.1 Meeting Up with Another Traveller

Often a person decides to travel to Europe because their friend has planned a trip. The aim of the second person's trip is to spend as many days with the first person as possible. However, due to budget constraints, it might not be possible for the second person to spend their whole trip with the first person.

This problem can be solved by extending our model. We introduce a new variable $t_{id}$, which is a binary variable that equals 1 if the two people are together in city $i$ on day $d$. We also add an additional subscript $p$ to the location and move variables to indicate the person they refer to [i.e., $x_{pid}$ and $m_{pijd}$]. When the second person tries to maximise the number of days they are with the first person, their objective function is:

$$\text{Max} \sum_{i \in \mathcal{C}} \sum_{d=1}^{D} t_{id}$$

For this model, we need the additional constraint:

$$\sum_{p=1}^{2} x_{pid} \geq 2 \cdot t_{id} \qquad \forall i \in \mathcal{C}, d = 1..D \tag{11}$$

As we are maximising $t_{id}$, this constraint will ensure that $t_{id} = 1$ if both $x_{1id} = 1$ and $x_{2id} = 1$ (i.e., both people are in city $i$ on day $d$), otherwise $t_{id} = 0$.

This problem requires five optimisation steps:

1. Maximise utility of Person 1's trip
2. Minimise cost of Person 1's trip given that utility
3. Maximise number of days where Person 1 and Person 2 are together given Person 1's trip

4. Maximise utility of Person 2's trip given that number of days together
5. Minimise cost of Person 2's trip given that utility and number of days together

The first two optimisation steps decide the first person's trip independently of the second person. The last three optimisation steps decide the second person's trip given that the first person's trip has already been decided.

To reduce computational time, this model was run for a 10 day trip with constant utility on each day (i.e., the initial utility values of each city in the base model). The first person's budget is \$5,000, and the second person's budget is \$3,500. The results are presented in the two tables below.

| **Days** | **Number** | **City** |
|---|---|---|
| 1-4 | 4 | London |
| 5-6 | 2 | Berlin |
| 7-10 | 4 | Paris |
| **Total Cost (\$)** | | 4,355 |
| **Total Utility** | | 944.00 |

Table 17: Person 1

| **Days** | **Number** | **City** |
|---|---|---|
| 1-2 | 2 | London |
| 3-6 | 4 | Berlin |
| 7-10 | 4 | Istanbul |
| **Total Cost (\$)** | | 3,417 |
| **Total Utility** | | 856.73 |

Table 18: Person 2

As the second person has a smaller budget, they are only able to spend 4 out of the 10 days with the first person. The two people are together on days 1-2 in London, and on days 5-6 in Berlin. The second person's trip is \$938 cheaper than the first person's trip, but has a 10% lower utility.

### 4.3.2 Avoiding Another Traveller

The previous section explored the problem of the second person wanting to spend as many days with the first person as possible. A similar problem could be that the second person wants to *minimise* the number of days that they are in the same city as the first person.

This problem can be solved using a similar model. The new objective function is:

$$\text{Min} \sum_{i \in \mathcal{C}} \sum_{d=1}^{D} t_{id}$$

The additional constraint in the previous section needs to be adjusted to:

$$\sum_{p=1}^{2} x_{pid} \leq 1 + t_{id} \qquad \forall i \in \mathcal{C}, d = 1..D \tag{12}$$

As we are minimising $t_{id}$, this constraint will ensure that $t_{id} = 1$ if both $x_{1id} = 1$ and $x_{2id} = 1$ (i.e., both people are in city $i$ on day $d$), otherwise $t_{id} = 0$.

This problem requires the same five optimisation steps as the previous section, except step 3 is now a minimisation step.

1. Maximise utility of Person 1's trip
2. Minimise cost of Person 1's trip given that utility
3. **Minimise** number of days where Person 1 and Person 2 are together given Person 1's trip
4. Maximise utility of Person 2's trip given that number of days together
5. Minimise cost of Person 2's trip given that utility and number of days together

The model was run over 10 days with constant utility, and both people having a budget of \$4,200. The results are presented in the two tables below.

| **Days** | **# of Days** | **City** |
|---|---|---|
| 1-4 | 4 | London |
| 5-8 | 4 | Berlin |
| 9-10 | 2 | Paris |
| **Total Cost (\$)** | | 4,169 |
| **Total Utility** | | 933.7 |

Table 19: Person 1

| **Days** | **Number** | **City** |
|---|---|---|
| 1-2 | 2 | Istanbul |
| 3-4 | 2 | Berlin |
| 5-8 | 4 | Paris |
| 9-10 | 2 | London |
| **Total Cost (\$)** | | 4,132 |
| **Total Utility** | | 899.01 |

Table 20: Person 2

The second person is able to afford a trip where they are never in the same city on the same day as the first person. However, the second person's trip has a 4% lower utility than the first person's trip even though both trips cost a similar amount. The second person is foregoing 4% of their potential utility in order to never spend a day in the same city as the first person.

## 4.4 Multiple Person Trip

Suppose we now have a set of people, $\mathcal{P}$, planning a trip together, each with their own budget, $\beta_p$ and their own criteria ratings, $\rho_p$. We need to add another dimension to each variable corresponding to each person. For instance, $x_{id}$ becomes $x_{pid}$ and denotes whether person $p$ is in city $i$ on day $d$. Furthermore, we require another binary variable, $t_{pqid}$ representing whether person $p$ and person $q$ are together in city $i$ on day $d$. To ensure this variable is in fact equal to one when two people are together, we require the following constraints (these constraints are similar to those in Section 4.3):

$$x_{pid} + x_{qid} \leq 1 + t_{pqid} \qquad \forall p, q \in \mathcal{P}, i \in \mathcal{C}, d = 1..D$$
$$2 \cdot t_{pqid} \geq x_{pid} + x_{qid} \qquad \forall p, q \in \mathcal{P}, i \in \mathcal{C}, d = 1..D$$

The new objective function maximises the total enjoyment of all people:

$$\text{Max} \sum_{p\in\mathcal{P}} \sum_{d=1}^{D} \sum_{i\in\mathcal{C}} \sum_{c=1}^{8} \rho_{pc} \cdot \kappa_{ic} \cdot x_{pid}$$

We also need to adjust the budget constraint to handle mutiple people:

$$\sum_{d=1}^{D} \sum_{i\in\mathcal{C}} \delta_i \cdot x_{pid} + \sum_{d=1}^{D-1} \sum_{i\in\mathcal{C}} \sum_{j\in\mathcal{C}} \mu_{ij} \cdot m_{pijd} + \sum_{i\in\mathcal{C}} (x_{i,1} \cdot \lambda_i + x_{i,D} \cdot \nu_i) \leq \beta_p \qquad \forall p \in \mathcal{P}$$

In addition to the criteria rating, the enjoyment objective function might also depend on each persons' preference for travelling companions. Perhaps there are two people who dislike each other or there is one person that everyone would like to spend time with. We can introduce a new parameter, $\gamma_{pq}$ representing how much person $p$ enjoys the company of person $q$. We will restrict $\gamma_{pq}$ to be between -5 and 5 inclusive and set $\gamma_{pp} = 0$. The use of negative values ensures that a traveller's enjoyment decreases if they spend time with someone they dislike. We will use a scale factor, $\sigma$, to make the "person enjoyment" comparable to the "place enjoyment". The objective function becomes:

$$\text{Max} \sum_{p\in\mathcal{P}} \sum_{d=1}^{D} \sum_{i\in\mathcal{C}} \sum_{c=1}^{8} \rho_{pc} \cdot \kappa_{ic} \cdot x_{pid} + \sum_{d=1}^{D} \sum_{i\in\mathcal{C}} \sum_{p\in\mathcal{P}} \sum_{q\in\mathcal{P}} \sigma \cdot \gamma_{pq} \cdot t_{pqid}$$

Suppose we set $\rho_{pc}$ and $\gamma_{pq}$ as the values shown in Tables 21 and 22. In this extreme case, persons 1 and 2 have the same interests, as do persons 3 and 4. Furthermore, all four people rate person 5 highly. On the other hand, person 5 shares no interests with his fellow travellers as well as having low ratings for all of them. We choose $\sigma$ to be 25. To reduce computation time, we set the total number of days to be 7 and the maximum number of days allowed in a city to be 3.

| **Person** | **Beach** | **History** | **Museums** | **Architecture** | **Nightlife** | **Food** | **Sport** | **Theatre** |
|---|---|---|---|---|---|---|---|---|
| **1** | 1 | 10 | 10 | 1 | 1 | 1 | 1 | 1 |
| **2** | 1 | 10 | 10 | 1 | 1 | 1 | 1 | 1 |
| **3** | 1 | 1 | 1 | 1 | 1 | 1 | 10 | 10 |
| **4** | 1 | 1 | 1 | 1 | 1 | 1 | 10 | 10 |
| **5** | 10 | 0 | 0 | 0 | 10 | 0 | 0 | 0 |

Table 21: Criteria ratings for 5 people

The optimal trip is displayed in Table 23. Due to the low values of $\gamma$ for person 5, the model allocates him a trip in which he is always by himself. Furthermore, persons 1 and 2 have identical trips due to their similar interests and high $\gamma$ rating. This is also the case for persons 3 and 4.

| Person | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| **1** | 0 | 3 | -1 | 1 | 5 |
| **2** | 3 | 0 | -1 | 1 | 5 |
| **3** | -1 | -1 | 0 | 3 | 5 |
| **4** | 1 | 1 | 3 | 0 | 5 |
| **5** | -5 | -5 | -5 | -5 | 0 |

Table 22: Person ratings for 5 people

| Day | Person 1 | Person 2 | Person 3 | Person 4 | Person 5 |
|---|---|---|---|---|---|
| **1** | Florence | Florence | Madrid | Madrid | London |
| **2** | Florence | Florence | Madrid | Madrid | London |
| **3** | Berlin | Berlin | London | London | Barcelona |
| **4** | Berlin | Berlin | London | London | Barcelona |
| **5** | Rome | Rome | London | London | Barcelona |
| **6** | Rome | Rome | Paris | Paris | Crete |
| **7** | Rome | Rome | Paris | Paris | Crete |
| **Enjoyment** | 1985 | 1985 | 1903 | 1903 | 1195 |
| **Total Cost** | $3,865 | $3,865 | $3,873 | $3,873 | $3,565 |

Table 23: 7 day trip for 5 people

Now, suppose that no traveller is allowed to be alone at any point during their trip. In other words, if person $p$ is in city $i$ on day $d$ (ie. $x_{pid} = 1$) then we require the "together variable" to be greater than or equal to two.

$$\sum_{q \in \mathcal{P}} t_{pqid} \geq 2 \cdot x_{pid} \qquad \forall p \in \mathcal{P}, i \in \mathcal{C}, d = 1..D$$

Running the model with this new constraint gives the trip shown in Table 24. Since person 5 is forced to be with someone at all times, his enjoyment is much lower in comparison to the enjoyment of his companions.

| Day | Person 1 | Person 2 | Person 3 | Person 4 | Person 5 |
|---|---|---|---|---|---|
| **1** | Rome | Rome | Barcelona | Barcelona | Barcelona |
| **2** | Rome | Rome | Barcelona | Barcelona | Barcelona |
| **3** | Rome | Rome | Barcelona | Barcelona | Barcelona |
| **4** | Florence | Florence | London | London | London |
| **5** | Florence | Florence | London | London | London |
| **6** | Berlin | Berlin | Paris | Paris | Berlin |
| **7** | Berlin | Berlin | Paris | Paris | Berlin |
| **Enjoyment** | 2235 | 2235 | 2483 | 2483 | -589 |
| **Total Cost** | $2,689 | $2,689 | $2,635 | $2,635 | $2,675 |

Table 24: 7 day trip for 5 people, no traveller is ever allowed to be alone

Maximising the total enjoyment is perhaps not the fairest way to measure enjoyment; there may be one traveller who is assigned a trip with a low enjoyment value compared to his fellow travellers. To avoid this problem, we could maximise the minimum enjoyment:

$$\text{Max } \min_{p\in\mathcal{P}} \left\{ \sum_{d=1}^{D} \sum_{i\in\mathcal{C}} \sum_{c=1}^{8} \rho_{pc} \cdot \kappa_{ic} \cdot x_{pid} + \sum_{d=1}^{D} \sum_{i\in\mathcal{C}} \sum_{q\in\mathcal{P}} \gamma_{pq} \cdot t_{pqid} \right\}$$

This is now nonlinear, thus we linearise it with a new variable $\xi$.

$$\text{Max } \xi$$

$$\text{s.t} \qquad \xi \leq \sum_{d=1}^{D} \sum_{i\in\mathcal{C}} \sum_{c=1}^{8} \rho_{pc} \cdot \kappa_{ic} \cdot x_{pid} + \sum_{d=1}^{D} \sum_{i\in\mathcal{C}} \sum_{q\in\mathcal{P}} \gamma_{pq} \cdot t_{pqid} \qquad \forall p \in \mathcal{P}$$

In all cases, once the enjoyment has been maximised, we set the enjoyment to be fixed at the optimal value ant then minimise cost.

# 5 Conclusion and Recommendations

We have developed a flexible model to solve the problem of planning an optimal trip to Europe. Our model allows simple modification through additional constraints to explore real-world scenarios, such as being in a city on a certain day of the trip or never visiting a certain city. The flexibility of the formulation allows extensions to replicate more complex situations by building on our base model.

Heuristics and valid inequalities were developed to help reduce computation time in situations where the model was inefficient. Using the greedy heuristics, we were able to derive upper and lower bounds on the objective value of our program, which improved computational efficiency.

When we implement this program on a website, the model will be adjusted in several ways. Firstly, the base utility will need to be derived using a more sophisticated method, or alternatively, we could incorporate the criteria ratings into our utility data. Moreover, our flight data is currently implemented as static values. Flight data is constantly changing due to factors such as fluctuating exchange rates and competition between airlines, so storage of static information is inappropriate for our problem. Hence, we will gather our flight data in real-time.

In this project we consider air travel as the only mode of transport. However, we will extend the model to handle trains, private cars and naval transport as in some cases these would be preferable with regards to cost or speed.

One notable recommendation we make is that changing the time scale from days to hours is ill-advised. However, once a traveller knows the days on which they travel between cities, a time scale of hours could be used to optimise the exact time of the day that they travel.

Finally, our website will allow the user to input data to indicate their preferences towards cities or criteria. This will constrain our model further, allowing for reduced computation time.

In summary, with a few minor tweaks, our model has potential to be built into a website where any traveller can freely find their dream European holiday.

# 6 Acknowledgements

We would like to thank our lecturer Dr. Alysson Costa for the assistance he provided at many stages of our project. Specifically, when we were confronted with the issue that our decaying utility function could lead to non-linear constraints, Dr. Costa recommended we precompute the decayed utility values. His suggestion and hints to reduce computational time regarding floating point inequalities were extremely helpful.

# 7 Appendix

Table 25: Airfares from Melbourne for Twenty Five Cities

| City | Airfare |
|---|---|
| Moscow | 1390 |
| Paris | 1090 |
| London | 1007 |
| Madrid | 1175 |
| Rome | 1082 |
| Crete | 1298 |
| Barcelona | 1101 |
| Berlin | 1376 |
| Budapest | 1399 |
| Florence | 1611 |
| Amsterdam | 1376 |
| Prague | 1236 |
| Istanbul | 1140 |
| Vienna | 1044 |
| Venice | 1362 |
| Goreme | 1448 |
| Lisbon | 1387 |
| Nice | 1376 |
| Reykjavik | 1842 |
| Edinburgh | 1226 |
| Dublin | 1253 |
| Krakow | 1546 |
| Copenhagen | 1356 |
| Athens | 1465 |
| Munich | 1230 |

Table 26: Airfares to Melbourne for Twenty Five Cities

| City | Airfare |
|---|---|
| Moscow | 803 |
| Paris | 844 |
| London | 875 |
| Madrid | 1061 |
| Rome | 934 |
| Crete | 922 |
| Barcelona | 983 |
| Berlin | 921 |
| Budapest | 1033 |
| Florence | 1186 |
| Amsterdam | 622 |
| Prague | 1058 |
| Istanbul | 748 |
| Vienna | 836 |
| Venice | 1076 |
| Goreme | 918 |
| Lisbon | 788 |
| Nice | 1119 |
| Reykjavik | 1260 |
| Edinburgh | 986 |
| Dublin | 880 |
| Krakow | 1140 |
| Copenhagen | 827 |
| Athens | 876 |
| Munich | 1016 |

Table 27: Low Daily costs of Fifteen Cities

| City | Cost |
|---|---|
| Moscow | 36 |
| Paris | 81 |
| London | 110 |
| Madrid | 57 |
| Rome | 68 |
| Crete | 69 |
| Barcelona | 49 |
| Berlin | 50 |
| Budapest | 75 |
| Florence | 56 |
| Amsterdam | 68 |
| Prague | 39 |
| Istanbul | 32 |
| Vienna | 56 |
| Venice | 54 |

Table 28: Mid Daily costs of Fifteen Cities

| City | Cost |
|---|---|
| Moscow | 92 |
| Paris | 223 |
| London | 298 |
| Madrid | 148 |
| Rome | 169 |
| Crete | 181 |
| Barcelona | 126 |
| Berlin | 130 |
| Budapest | 145 |
| Florence | 142 |
| Amsterdam | 165 |
| Prague | 100 |
| Istanbul | 85 |
| Vienna | 158 |
| Venice | 134 |

Table 29: High Daily costs of Fifteen Cities

| City | Cost |
|---|---|
| Moscow | 233 |
| Paris | 657 |
| London | 845 |
| Madrid | 393 |
| Rome | 423 |
| Crete | 418 |
| Barcelona | 321 |
| Berlin | 340 |
| Budapest | 354 |
| Florence | 356 |
| Amsterdam | 432 |
| Prague | 251 |
| Istanbul | 244 |
| Vienna | 494 |
| Venice | 325 |

Table 30: Base Utility for Fifteen Cities

| City | Base Utility |
|---|---|
| Moscow | 71.017 |
| Paris | 92.66 |
| London | 100 |
| Madrid | 77.356 |
| Rome | 85.238 |
| Crete | 50.707 |
| Barcelona | 78.964 |
| Berlin | 87.116 |
| Budapest | 67.093 |
| Florence | 74.819 |
| Amsterdam | 73.17 |
| Prague | 77.126 |
| Istanbul | 77.067 |
| Vienna | 75.922 |
| Venice | 77.436 |

| | **Beach** | **History** | **Museums** | **Architecture** | **Nightlife** | **Food** | **Sport** | **Theatre** |
|---|---|---|---|---|---|---|---|---|
| **Moscow** | 0 | 2 | 4 | 8 | 5 | 6 | 5 | 9 |
| **Paris** | 0 | 7 | 8 | 7 | 6 | 9 | 7 | 8 |
| **London** | 0 | 7 | 7 | 8 | 9 | 3 | 8 | 10 |
| **Madrid** | 0 | 6 | 7 | 5 | 6 | 7 | 9 | 6 |
| **Rome** | 0 | 10 | 8 | 6 | 4 | 10 | 7 | 7 |
| **Crete** | 10 | 9 | 2 | 5 | 2 | 6 | 2 | 1 |
| **Barcelona** | 8 | 5 | 7 | 10 | 9 | 8 | 8 | 5 |
| **Berlin** | 0 | 10 | 7 | 5 | 9 | 5 | 7 | 4 |
| **Budapest** | 0 | 5 | 7 | 8 | 8 | 5 | 4 | 5 |
| **Florence** | 2 | 8 | 10 | 7 | 3 | 10 | 2 | 5 |
| **Amsterdam** | 0 | 5 | 7 | 7 | 9 | 4 | 6 | 6 |
| **Prague** | 0 | 4 | 6 | 6 | 7 | 5 | 5 | 4 |
| **Istanbul** | 0 | 5 | 7 | 6 | 4 | 5 | 4 | 4 |
| **Vienna** | 0 | 8 | 6 | 8 | 3 | 5 | 4 | 3 |
| **Venice** | 7 | 6 | 3 | 8 | 3 | 8 | 2 | 1 |

Table 31: City Ratings

Table 32: Airfares between Twenty Five Cities (Part I)

| From\ To | Moscow | Paris | London | Madrid | Rome | Crete | Barcelona | Berlin | Budapest | Florence | Amsterdam |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Moscow | 0 | 146 | 126 | 202 | 146 | 168 | 143 | 156 | 207 | 291 | 154 |
| Paris | 227 | 0 | 60 | 143 | 93 | 123 | 139 | 110 | 162 | 85 | 52 |
| London | 213 | 82 | 0 | 249 | 160 | 284 | 135 | 107 | 163 | 224 | 124 |
| Madrid | 188 | 86 | 136 | 0 | 144 | 188 | 69 | 70 | 96 | 138 | 103 |
| Rome | 223 | 80 | 125 | 146 | 0 | 177 | 39 | 96 | 69 | 84 | 94 |
| Crete | 233 | 81 | 223 | 188 | 58 | 0 | 131 | 123 | 58 | 241 | 200 |
| Barcelona | 200 | 121 | 79 | 76 | 32 | 165 | 0 | 143 | 108 | 103 | 90 |
| Berlin | 130 | 53 | 81 | 97 | 83 | 153 | 110 | 0 | 83 | 214 | 92 |
| Budapest | 249 | 133 | 125 | 96 | 41 | 123 | 96 | 76 | 0 | 164 | 157 |
| Florence | 309 | 246 | 162 | 172 | 86 | 301 | 90 | 186 | 204 | 0 | 227 |
| Amsterdam | 155 | 158 | 73 | 141 | 136 | 211 | 117 | 106 | 125 | 229 | 0 |
| Prague | 112 | 122 | 100 | 112 | 48 | 193 | 116 | 108 | 138 | 144 | 97 |
| Istanbul | 123 | 101 | 127 | 154 | 67 | 173 | 122 | 86 | 41 | 235 | 66 |
| Vienna | 145 | 162 | 122 | 202 | 90 | 75 | 145 | 117 | 175 | 171 | 90 |
| Venice | 167 | 65 | 17 | 117 | 84 | 191 | 97 | 119 | 110 | 152 | 75 |
| Goreme | 172 | 206 | 205 | 237 | 137 | 253 | 181 | 142 | 86 | 522 | 136 |
| Lisbon | 282 | 80 | 77 | 50 | 135 | 220 | 107 | 123 | 117 | 173 | 55 |
| Nice | 278 | 127 | 119 | 143 | 48 | 220 | 50 | 90 | 138 | 173 | 121 |
| Reykjavik | 566 | 190 | 284 | 337 | 242 | 484 | 217 | 260 | 246 | 454 | 246 |
| Edinburgh | 233 | 194 | 71 | 135 | 204 | 371 | 179 | 115 | 151 | 390 | 168 |
| Dublin | 196 | 87 | 58 | 102 | 146 | 265 | 107 | 86 | 143 | 253 | 138 |
| Krakow | 191 | 135 | 94 | 126 | 142 | 286 | 107 | 176 | 138 | 181 | 143 |
| Copenhagen | 258 | 174 | 71 | 158 | 160 | 315 | 95 | 56 | 157 | 200 | 152 |
| Athens | 118 | 149 | 156 | 135 | 72 | 39 | 156 | 109 | 91 | 230 | 109 |
| Munich | 188 | 120 | 136 | 163 | 69 | 112 | 135 | 110 | 126 | 192 | 170 |

Table 33: Airfares between Twenty Five Cities (Part II)

| From\ To | Prague | Istanbul | Vienna | Venice | Goreme | Lisbon | Nice | Reykjavik | Edinburgh | Dublin | Krakow |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Moscow | 146 | 115 | 131 | 129 | 167 | 283 | 200 | 406 | 266 | 263 | 236 |
| Paris | 165 | 150 | 140 | 68 | 375 | 178 | 114 | 240 | 124 | 106 | 152 |
| London | 171 | 197 | 179 | 152 | 267 | 205 | 160 | 183 | 116 | 116 | 177 |
| Madrid | 124 | 180 | 143 | 119 | 168 | 57 | 152 | 336 | 127 | 125 | 107 |
| Rome | 94 | 74 | 105 | 71 | 157 | 155 | 62 | 180 | 155 | 174 | 147 |
| Crete | 114 | 120 | 164 | 169 | 265 | 154 | 144 | 394 | 178 | 211 | 214 |
| Barcelona | 110 | 141 | 103 | 74 | 203 | 69 | 50 | 241 | 151 | 117 | 100 |
| Berlin | 123 | 99 | 176 | 104 | 197 | 118 | 108 | 245 | 100 | 102 | 128 |
| Budapest | 113 | 50 | 149 | 97 | 100 | 104 | 130 | 265 | 117 | 104 | 107 |
| Florence | 155 | 279 | 229 | 150 | 470 | 197 | 233 | 421 | 223 | 228 | 107 |
| Amsterdam | 85 | 121 | 155 | 95 | 200 | 137 | 73 | 252 | 88 | 101 | 134 |
| Prague | 0 | 85 | 152 | 65 | 186 | 174 | 135 | 256 | 134 | 114 | 141 |
| Istanbul | 108 | 0 | 59 | 65 | 33 | 140 | 126 | 337 | 143 | 162 | 149 |
| Vienna | 181 | 121 | 0 | 194 | 152 | 254 | 166 | 284 | 199 | 211 | 147 |
| Venice | 96 | 177 | 152 | 0 | 274 | 153 | 200 | 316 | 135 | 146 | 127 |
| Goreme | 206 | 27 | 55 | 230 | 0 | 257 | 130 | 384 | 189 | 198 | 440 |
| Lisbon | 178 | 260 | 161 | 123 | 441 | 0 | 66 | 328 | 142 | 108 | 174 |
| Nice | 96 | 180 | 157 | 128 | 224 | 116 | 0 | 313 | 175 | 97 | 189 |
| Reykjavik | 285 | 334 | 370 | 323 | 332 | 325 | 334 | 0 | 175 | 97 | 189 |
| Edinburgh | 140 | 294 | 370 | 127 | 613 | 174 | 168 | 192 | 0 | 39 | 139 |
| Dublin | 110 | 217 | 156 | 146 | 396 | 114 | 138 | 254 | 34 | 0 | 137 |
| Krakow | 180 | 247 | 197 | 205 | 362 | 171 | 139 | 341 | 133 | 119 | 0 |
| Copenhagen | 96 | 234 | 229 | 192 | 363 | 191 | 154 | 270 | 138 | 149 | 149 |
| Athens | 62 | 98 | 105 | 130 | 141 | 173 | 137 | 263 | 179 | 138 | 149 |
| Munich | 194 | 103 | 194 | 165 | 176 | 159 | 117 | 263 | 142 | 143 | 154 |

Table 34: Airfares between Twenty Five Cities (Part III)

| From\ To | Copenhagen | Athens | Munich |
|---|---|---|---|
| Moscow | 162 | 141 | 152 |
| Paris | 75 | 226 | 68 |
| London | 130 | 145 | 133 |
| Madrid | 108 | 169 | 184 |
| Rome | 143 | 160 | 28 |
| Crete | 176 | 46 | 144 |
| Barcelona | 104 | 146 | 93 |
| Berlin | 43 | 174 | 110 |
| Budapest | 123 | 133 | 122 |
| Florence | 164 | 268 | 158 |
| Amsterdam | 60 | 128 | 183 |
| Prague | 86 | 143 | 153 |
| Istanbul | 51 | 70 | 83 |
| Vienna | 177 | 154 | 149 |
| Venice | 184 | 181 | 101 |
| Goreme | 111 | 120 | 147 |
| Lisbon | 192 | 171 | 124 |
| Nice | 103 | 175 | 87 |
| Reykjavik | 218 | 175 | 266 |
| Edinburgh | 159 | 273 | 220 |
| Dublin | 60 | 186 | 155 |
| Krakow | 110 | 176 | 220 |
| Copenhagen | 0 | 128 | 218 |
| Athens | 131 | 0 | 145 |
| Munich | 131 | 166 | 0 |

For brevities sake, we shall only include the tables detailing the airfare augmentation data for Moscow and Florence.

Table 35: Airfare Augmentation Data for Moscow

| City | Sunday | Monday | Tuesday | Wednesday | Thursday | Friday | Saturday |
|---|---|---|---|---|---|---|---|
| Moscow | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Paris | 4 | 1 | 0 | 3 | 8 | 13 | 10 |
| London | 11 | 3 | 0 | 2 | 6 | 8 | 6 |
| Madrid | 10 | 3 | 0 | 6 | 13 | 17 | 14 |
| Rome | 4 | 1 | 0 | 3 | 11 | 12 | 12 |
| Crete | 2 | 0 | 0 | 2 | 7 | 13 | 5 |
| Barcelona | 12 | 3 | 0 | 7 | 13 | 16 | 14 |
| Berlin | 1 | 0 | 2 | 7 | 12 | 7 | 5 |
| Budapest | 0 | 2 | 2 | 7 | 10 | 7 | 5 |
| Florence | 5 | 0 | 5 | 15 | 19 | 15 | 7 |
| Amsterdam | 0 | 0 | 4 | 11 | 15 | 8 | 3 |
| Prague | 0 | 0 | 3 | 7 | 11 | 7 | 3 |
| Istanbul | 2 | 2 | 9 | 11 | 12 | 7 | 0 |
| Vienna | 0 | 0 | 7 | 12 | 16 | 14 | 7 |
| Venice | 2 | 0 | 5 | 14 | 22 | 14 | 6 |
| Goreme | 4 | 1 | 0 | 2 | 5 | 6 | 15 |
| Lisbon | 4 | 0 | 3 | 8 | 13 | 10 | 5 |
| Nice | 3 | 0 | 4 | 12 | 16 | 17 | 6 |
| Reykjavik | 1 | 0 | 6 | 4 | 6 | 7 | 1 |
| Edinburgh | 1 | 0 | 2 | 8 | 16 | 12 | 10 |
| Dublin | 6 | 0 | 3 | 10 | 20 | 17 | 15 |
| Krakow | 0 | 0 | 6 | 6 | 16 | 6 | 2 |
| Copenhagen | 0 | 0 | 3 | 6 | 10 | 10 | 10 |
| Athens | 2 | 0 | 3 | 6 | 13 | 15 | 3 |
| Munich | 1 | 0 | 5 | 11 | 15 | 18 | 7 |

Table 36: Airfare Augmentation Data for Florence

| City | Sunday | Monday | Tuesday | Wednesday | Thursday | Friday | Saturday |
|---|---|---|---|---|---|---|---|
| Moscow | 20 | 7 | 0 | 5 | 18 | 28 | 39 |
| Paris | 20 | 6 | 0 | 5 | 18 | 25 | 39 |
| London | 17 | 0 | 0 | 10 | 12 | 13 | 40 |
| Madrid | 17 | 10 | 3 | 0 | 4 | 5 | 9 |
| Rome | 0 | 15 | 2 | 11 | 8 | 2 | 8 |
| Crete | 5 | 0 | 2 | 3 | 4 | 15 | 6 |
| Barcelona | 6 | 0 | 2 | 7 | 10 | 9 | 11 |
| Berlin | 18 | 7 | 0 | 13 | 10 | 27 | 28 |
| Budapest | 1 | 2 | 2 | 4 | 2 | 0 | 5 |
| Florence | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Amsterdam | 4 | 3 | 2 | 5 | 0 | 12 | 18 |
| Prague | 5 | 0 | 4 | 7 | 3 | 4 | 5 |
| Istanbul | 4 | 0 | 1 | 3 | 4 | 7 | 8 |
| Vienna | 8 | 2 | 0 | 5 | 12 | 15 | 35 |
| Venice | 5 | 0 | 5 | 15 | 23 | 14 | 14 |
| Goreme | 5 | 2 | 0 | 1 | 4 | 6 | 10 |
| Lisbon | 12 | 0 | 1 | 8 | 12 | 11 | 13 |
| Nice | 5 | 3 | 2 | 0 | 1 | 1 | 9 |
| Reykjavik | 24 | 12 | 0 | 1 | 5 | 6 | 26 |
| Edinburgh | 10 | 0 | 3 | 8 | 15 | 16 | 25 |
| Dublin | 5 | 1 | 0 | 2 | 2 | 5 | 11 |
| Krakow | 26 | 9 | 3 | 0 | 0 | 5 | 11 |
| Copenhagen | 10 | 0 | 1 | 4 | 7 | 15 | 22 |
| Athens | 8 | 0 | 0 | 3 | 7 | 16 | 13 |
| Munich | 6 | 5 | 5 | 0 | 4 | 6 | 20 |

Here we give the table detailing how the flight costs of returning to Melbourne from Europe vary over the week. As we have arbitrarily set the trip to beginning on a Sunday, we do not need to consider how the flight costs to Europe from Melbourne vary.

Table 37: Varying Flight Costs to Melbourne

| City | Sunday | Monday | Tuesday | Wednesday | Thursday | Friday | Saturday |
|---|---|---|---|---|---|---|---|
| Moscow | 803 | 734 | 742 | 734 | 795 | 744 | 734 |
| Paris | 844 | 802 | 942 | 844 | 1057 | 1080 | 971 |
| London | 875 | 699 | 800 | 793 | 1044 | 1113 | 1101 |
| Madrid | 1061 | 880 | 1011 | 1083 | 1082 | 1082 | 1048 |
| Rome | 934 | 803 | 845 | 895 | 915 | 983 | 943 |
| Crete | 922 | 869 | 882 | 956 | 1017 | 2301 | 884 |
| Barcelona | 983 | 797 | 1059 | 978 | 1023 | 1082 | 1050 |
| Berlin | 921 | 853 | 804 | 905 | 905 | 918 | 934 |
| Budapest | 1033 | 1000 | 903 | 945 | 1013 | 945 | 1008 |
| Florence | 1186 | 1144 | 1026 | 1064 | 1066 | 1142 | 1086 |
| Amsterdam | 622 | 1003 | 612 | 819 | 956 | 956 | 1129 |
| Prague | 1058 | 891 | 990 | 1031 | 1039 | 956 | 1090 |
| Istanbul | 748 | 581 | 735 | 590 | 682 | 682 | 590 |
| Vienna | 836 | 895 | 809 | 1137 | 1136 | 1087 | 1249 |
| Venice | 1076 | 1036 | 887 | 937 | 1076 | 1074 | 1078 |
| Goreme | 918 | 763 | 1082 | 750 | 1056 | 1110 | 1079 |
| Lisbon | 788 | 939 | 783 | 941 | 934 | 960 | 960 |
| Nice | 1119 | 785 | 927 | 923 | 1015 | 1003 | 927 |
| Reykjavik | 1260 | 1002 | 1168 | 1241 | 1252 | 1114 | 1098 |
| Edinburgh | 986 | 780 | 885 | 874 | 986 | 1063 | 1051 |
| Dublin | 880 | 872 | 872 | 872 | 846 | 911 | 892 |
| Krakow | 1140 | 970 | 1126 | 1128 | 1140 | 1148 | 1152 |
| Copenhagen | 827 | 823 | 726 | 896 | 911 | 797 | 1010 |
| Athens | 876 | 857 | 830 | 870 | 954 | 902 | 764 |
| Munich | 1016 | 1015 | 1031 | 1028 | 1016 | 1041 | 1036 |

# References

[1] momondo.com.au, *Global Travel Search Engine*, accessed 6 May 2015. Available online: `http://www.momondo.com.au/`

[2] Budget Your Trip, *Travel Cost Search*, accessed 13 May 2015. Available online: `http://www.budgetyourtrip.com/`

[3] tripadvisor.com.au, *Travel Reviews and Advice*, accessed 14 April 2015. Available online: `http://www/tripadvisor.com.au/`

[4] hostelworld.com, *Travel Guides*, accessed 16 April 2015. Available online: `http://www/tripadvisor.com.au/`

[5] europeancitiesmarketing, *ECM*, accessed 23 May 2015. Available online: `http://www.europeancitiesmarketing.com`