Minimizing Travel Costs

A MATLAB report by Z and Charlie

Preparing a Function:

Creating Coordinates:

To begin, we needed to find the coordinates of each of the cities. To do so I googled the decimal form coordinates for Middlebury, Dallas, Detroit, and Tampa, with those coordinates being:

```
Midd_Coords = [44.0151, -73.1670]
```

```
Midd_Coords = 1×2
   44.0151  -73.1670
```

```
Dallas_Coords = [32.7767, -96.7970]
```

```
Dallas_Coords = 1×2
   32.7767  -96.7970
```

```
Detroit_Coords = [42.3333, -83.0992]
```

```
Detroit_Coords = 1×2
   42.3333  -83.0992
```

```
Tampa_Coords = [27.9506, -82.4572]
```

```
Tampa_Coords = 1×2
   27.9506  -82.4572
```

Because coordinates are written (Longitude, Latitude), X is measuring north/south, while Y is measuring east/west. Then, I took the Dallas Y coordinate and Tampa X coordinate and set them as (0,0) so that the points would all be in the first quadrant. Then, to solve for the coordinates in a corresponding X-Y Plane, I used the conversion $(X - \mathrm{X1}) * 70, (Y - \mathrm{Y1}) * 50$:

```
Tampa = [(Tampa_Coords(1,1)-Tampa_Coords(1,1))*70; (Tampa_Coords(1,2)-
Dallas_Coords(1,2))*50]
```

```
Tampa = 2×1
         0
  716.9900
```

```
Dallas = [(Dallas_Coords(1,1)-Tampa_Coords(1,1))*70; (Dallas_Coords(1,2)-
Dallas_Coords(1,2))*50]
```

```
Dallas = 2×1
  337.8270
         0
```

```
Detroit = [(Detroit_Coords(1,1)-Tampa_Coords(1,1))*70; (Detroit_Coords(1,2)-
Dallas_Coords(1,2))*50]
```

```
Detroit = 2×1
10^3 ×
```

```
1.0068
0.6849
```

```
Midd = [(Midd_Coords(1,1)-Tampa_Coords(1,1))*70; (Midd_Coords(1,2)-
Dallas_Coords(1,2))*50]
```

```
Midd = 2×1
10^3 ×
    1.1245
    1.1815
```

Now, we have our coordinates in a usable (x,y) coordinate system where we can use calculus to find the minimum and maximum.

Creating an Equation:

With these coordinates, we need to figure out how to represent the total cost to fly the jet to each location. The first step to doing this is finding the distance from the home base to each location, weighted by the frequency with which they'll have to travel there. To do this, I can use the following equation, where X1 and Y1 represent the coordinates of a city, and C is the number of visis per month:

$$C\sqrt{(X-X_1)^2+(Y-Y_1)^2}$$

After rounding the distances to remove decimals, this results in an overall distance formula of:

$$\text{Cost} = 6.74 * \left(4\sqrt{(X)^2+(Y-717)^2}+3\sqrt{(X-1125)^2+(Y-1182)^2}+2\sqrt{(X-338)^2+(Y)^2}+\sqrt{(X-1007)^2+(Y)^2}\right)$$

Now that we have a function, we can derive the non-constant parts in order to find the minimum location:

Finding the Minimum:

Derivation:

There isn't anything particularly unusual about finding the minimum for this matrix -- while the derivatives are difficult, they're fairly standard. We just need to find the derivative with respect for X, then the derivative with respect to Y, and then find when both of those equal 0.

```
Distance_Traveled = sqrt((y - 685)^2 + (x - 1007)^2) + 3*sqrt((y - 1182)^2
+ (x - 1125)^2) + 2*sqrt((x - 338)^2 + y^2) + 4*sqrt((y - 717)^2 + x^2)
```

Distance_Traveled =

$$\sqrt{(x-1007)^2+(y-685)^2}+3\sqrt{(x-1125)^2+(y-1182)^2}+2\sqrt{(x-338)^2+y^2}+4\sqrt{(y-717)^2+x^2}$$

```
Derivativex = diff(Distance_Traveled)
```

Derivativex =

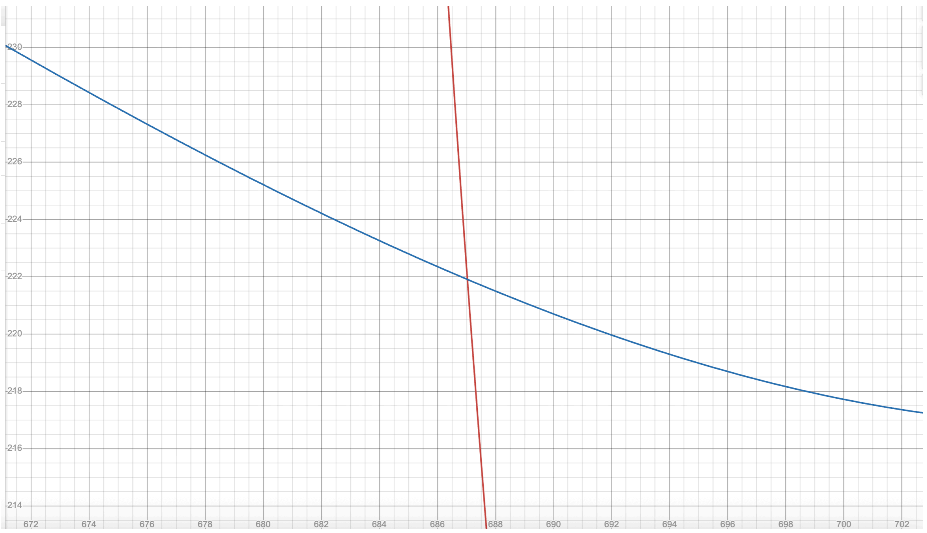
$$\frac{2\,x-676}{\sqrt{(x-338)^2+y^2}}+\frac{4\,x}{\sqrt{(y-717)^2+x^2}}+\frac{2\,x-2014}{2\,\sqrt{(x-1007)^2+(y-685)^2}}+\frac{3\,(2\,x-2250)}{2\,\sqrt{(x-1125)^2+(y-1182)^2}}$$

```
Derivativey = diff(Distance_Traveled,y)
```

```
Derivativey =
```

$$\frac{2\,(2\,y-1434)}{\sqrt{(y-717)^2+x^2}}+\frac{2\,y}{\sqrt{(x-338)^2+y^2}}+\frac{2\,y-1370}{2\,\sqrt{(x-1007)^2+(y-685)^2}}+\frac{3\,(2\,y-2364)}{2\,\sqrt{(x-1125)^2+(y-1182)^2}}$$

Now that we have the derivatives, we need to solve for when they both equal 0. Because these curves are so complex, it would be extremely difficult to find a perfect algebraic solution. Rather, because this is a practical application and we don't need infinite precision, we can find the intersection of the lines formed when they are set equal to 0. This resulted in the following intersection:



When looked at further, this intersection approximates to X = 221.90, Y = 687.03, which is a local extrema, but we still don't know if it's a minimum or a maximum.

Minimum or Maximum?

To find out whether it's a minimum or maximum, we can look at whether the Hessian matrix has all positive or negative eigenvalues. To find these eigenvalues, we can use MATLAB to find the second derivative matrix, before plugging in our points and solving for the eigenvalues.

```
MinimumX = 221.90;
MinimumY = 687.03;

Derivativexx = diff(Derivativex);
Derivativexy = diff(Derivativex,y);
Derivativeyx = diff(Derivativey);
Derivativeyy = diff(Derivativey, y);
Minimumxx = double(subs(Derivativexx, {x, y}, {MinimumX,MinimumY}))
```

```
Minimumxx =
```

```
0.0038
```

```
Minimumxy = double(subs(Derivativexy, {x, y}, {MinimumX,MinimumY}))
```

```
Minimumxy =
0.0016
```

```
Minimumyx = double(subs(Derivativeyx, {x, y}, {MinimumX,MinimumY}))
```

```
Minimumyx =
0.0016
```

```
Minimumyy = double(subs(Derivativeyy, {x, y}, {MinimumX,MinimumY}))
```

```
Minimumyy =
0.0211
```

```
Hessian = [Minimumxx, Minimumxy; Minimumyx, Minimumyy];
```

Next, we just need to find the eigenvalues. Luckily, MATLAB has a function for this!

```
Eigs_Hessian = eigs(Hessian)
```

```
Eigs_Hessian = 2×1
    0.0213
    0.0036
```

Since both of the eigenvalues are positive, this means that we have found a minimum, and the coordinates represented by X = 221.90, Y = 687.03 are the optimal location for the home base.

Final Location:

Finding the cost:

We can then take this location and solve for the minimum distance we would have to travel in a month, before multiplying this value by $6.74 per mile to find the cost:

```
MinimumDistance = double(subs(Distance_Traveled, {x, y},
{MinimumX,MinimumY}))
```

```
MinimumDistance =
6.1638e+03
```

```
MinimumCost = MinimumDistance*6.74
```

```
MinimumCost =
4.1544e+04
```

Next, we need to compare this to the cost at each of the four cities:

```
DallasDistance = double(subs(Distance_Traveled, {x, y},
{Dallas(1,1),Dallas(2,1)}));
DallasCost = DallasDistance*6.74
```

```
DallasCost =
5.6540e+04
```

```
TampaDistance = double(subs(Distance_Traveled, {x, y},
{Tampa(1,1),Tampa(2,1)}));
TampaCost = TampaDistance*6.74
```

```
TampaCost =
4.2090e+04
```

```
MiddDistance = double(subs(Distance_Traveled, {x, y},
{Midd(1,1),Midd(2,1)}));
MiddCost = MiddDistance*6.74
```

```
MiddCost =
5.5387e+04
```

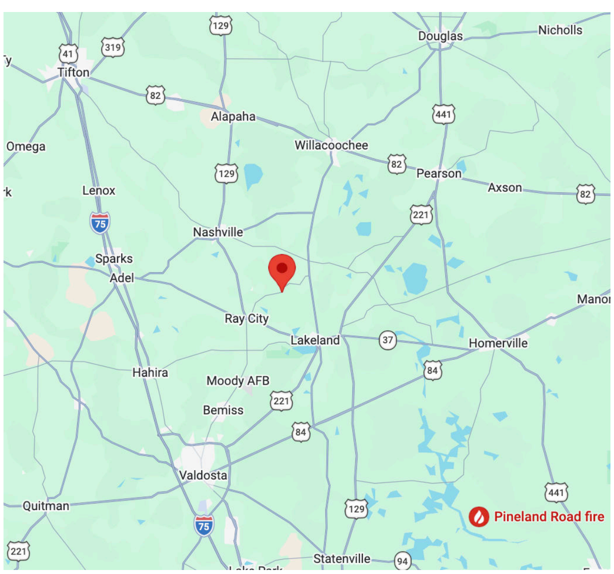
```
DetroitDistance = double(subs(Distance_Traveled, {x, y},
{Detroit(1,1),Detroit(2,1)}));
DetroitCost = DetroitDistance*6.74
```

```
DetroitCost =
5.0394e+04
```

As seen in these costs, by building the home base at the optimized coordinates, we're spending around 41,544 on jet fuel each month -- saving roughly 500 compared to if our home base was in Tampa (which would be the most efficient city). In comparison, building the home base in Dallas would cost almost 15,000 more per month, which is a signfiicant increase. The others, Midd and Detroit, are both somewhere in the middle: something which makes sense considering the importance of Tampa and that the other cities are all far away from dallas.

Physical Location:

Since we now have the coordinates in our own (X,Y) plane, we need to convert them back into Latitude and Longitude in order to find the location and nearest city. First, I need to divide the distances by their respective mileage per point of latitude/longitude. This results in 3.17 degrees of latitude and 13.66 degrees of longitude, which we can then add to the starting coordinates of (27.9506, -96.9790), resulting in a final location of (31.1175, 83.1356). This, when found in google maps, is a location in georgia next to the Florida border:



The nearest town is Ray City, but the nearest city with an airport is Valdosta (where Z's mother went to college), meaning that we would likely want to set up our home base there instead, where the technicians will have to fly 6,164 miles each month, at a cost of $41,544.

There were a couple assumptions made during this process. The first is that the distance per point of latitude will stay the same as we go north, which is not true -- it will get slightly smaller as you go further north. Additionally, it's assuming that the cost of fuel and flights will be constant in every region, and that the cost of building the home base is negligible. Lastly, it's assuming that we don't need perfect precision throughout all of the problem; using the intersection of curves and rounding to decimals, while it makes the process much easier, does make the answer less precise. Despite these assumptions, this location is likely very close to the optimal location, and thus still a good spot to build the home base.