



# 1 Introduction

Application du positionnement multidimensionnel (Multidimensional Scaling – MDS) aux distances entre les villes principales des provinces de Madagascar. Traitements sous R avec les fonctions `cmdscale()` [STATS] et `isoMDS()` [MASS].

Ce tutoriel vient illustrer mon [support de cours consacré au positionnement multidimensionnel](#) (*Multidimensional Scaling* en anglais, MDS). L'objectif est de représenter dans le plan les positions relatives des villes en prenant en entrée les distances kilométriques des routes qui les relient. C'est une application très parlante et/mais ressassée du MDS (exemple pour les [villes de France](#)). J'essaie d'être original ici en traitant les villes de Madagascar.

Mon idée initiale était de récupérer les informations directement sous R via les API des services de cartographie en ligne ([Google Maps](#), [Bing Maps](#), [OpenStreetMap](#), etc.). Mais la tâche ne s'est pas avérée aussi triviale que je ne le pensais. J'en parle justement dans [ce document](#) où je montre comment accéder aux données dans une application R à l'aide des packages spécialisés, en ce qui concerne OpenStreetMap tout du moins. Souhaitant traiter les villes principales des provinces de Madagascar, et leur nombre étant relativement retreint (6 provinces), j'ai finalement décidé de recueillir manuellement les distances sur Google Maps.

## 2 Distances entre les villes principales des provinces malgaches

Madagascar est une île [superbe](#) de l'Océan Indien. Elle est subdivisée en [6 provinces historiques](#) dont les chefs-lieux sont mis en évidence dans la carte ci-dessous (Figure 2). En adoptant les dénominations de Google Maps, nous avons, dans le sens trigonométrique : Antsiranana, Majunga, Tuléar, Fianarantsoa, Tananarive, Toamasina.

Via l'application en ligne, j'ai collecté manuellement les distances entre les villes pour obtenir le tableau suivant (Figure 1), la matrice est symétrique :

Ville	Antsiranana	Majunga	Tuléar	Fianarantsoa	Tananarive	Toamasina
Antsiranana	0	859	2038	1522	1109	1287
Majunga	859	0	1495	979	566	926
Tuléar	2038	1495	0	516	930	1266
Fianarantsoa	1522	979	516	0	414	750
Tananarive	1109	566	930	414	0	355
Toamasina	1287	926	1266	750	355	0

Figure 1- Distances routières (en Km) entre les villes principales des provinces malgaches

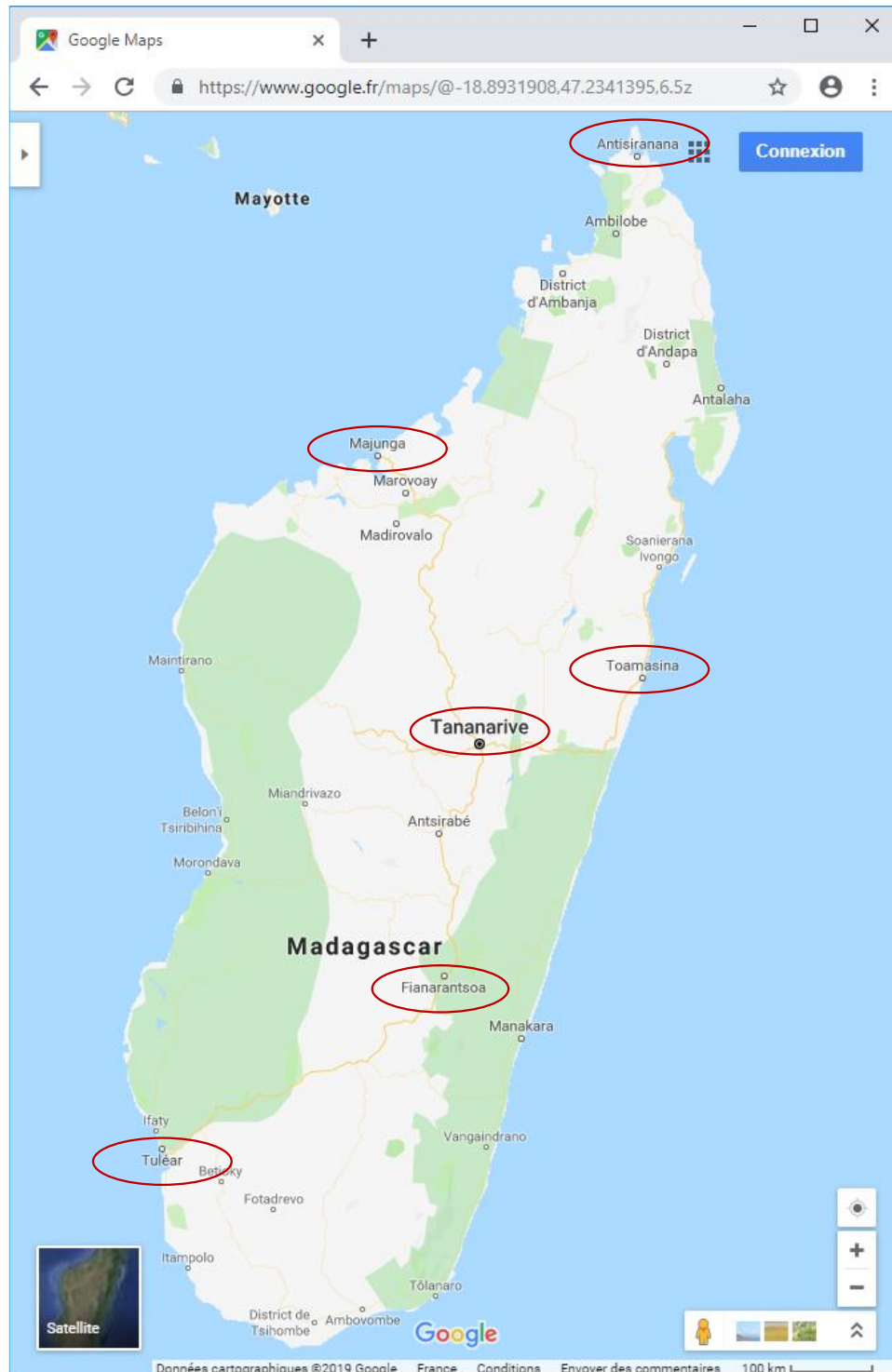


Figure 2 – Carte de Madagascar, avec les villes principales des 6 provinces

Précision importante, certains trajets nécessitent de prendre le ferry (plutôt un bac s'agissant des rivières malgaches). Je les ai exclus pour privilégier les solutions exclusivement routières. Ainsi, dans l'exemple ci-dessous (Figure 3), il faut passer par la capitale Tananarive si on veut utiliser uniquement la voiture pour aller de Fianarantsoa à Toamasina

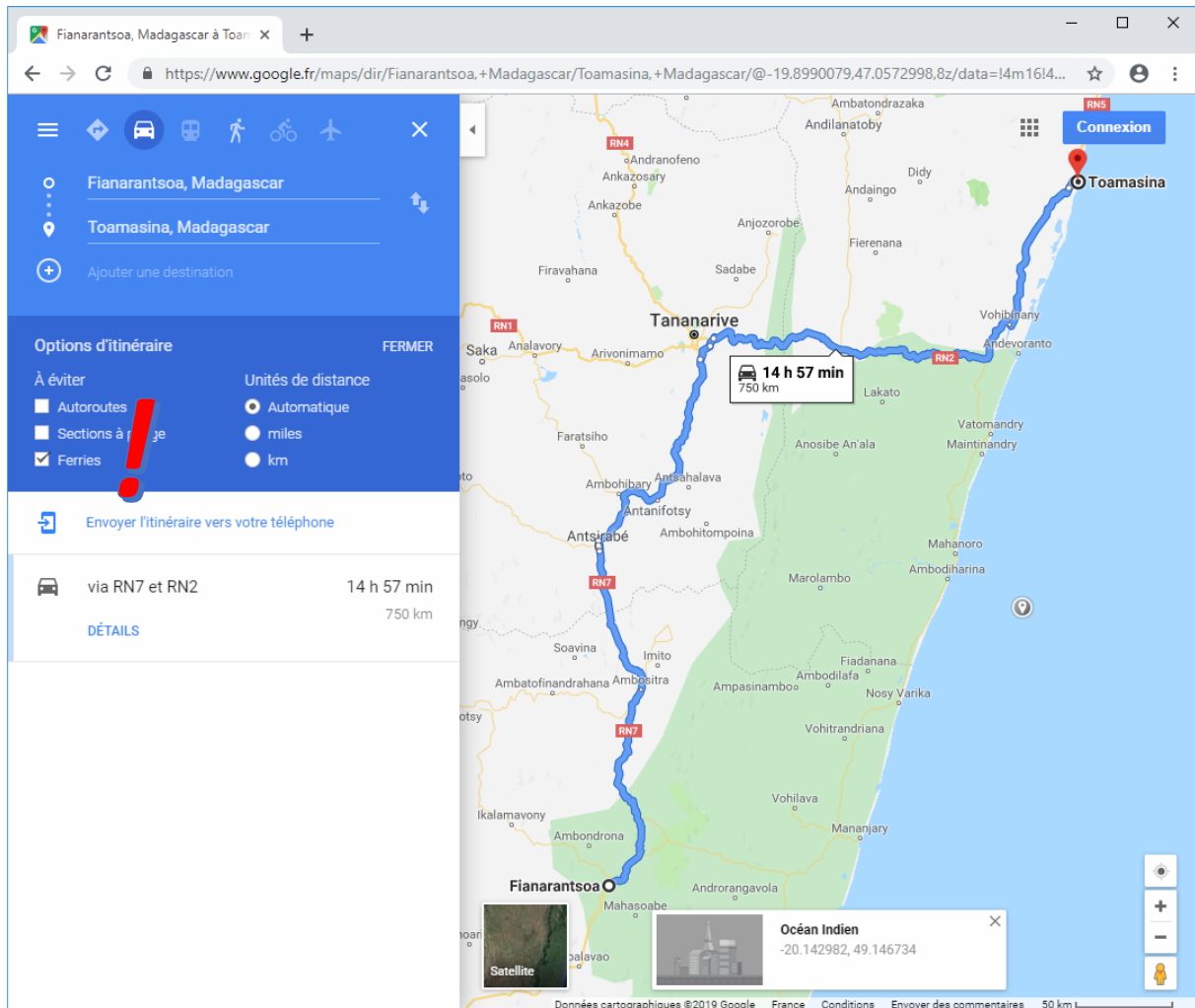


Figure 3 - Trajet de Fianarantsoa à Toamasina, en excluant de prendre les "ferries"

### 3 Positionnement multidimensionnel sous R

#### 3.1 Importation et préparation des données

Nous importons le fichier "provinces\_mcar.xlsx" à l'aide de la fonction `read.xlsx()` du package "openxlsx". Il faut indiquer que la première ligne et la première colonne représentent respectivement des identifiants (`colNames = T`, `rowNames = T`).

```
#library
library(openxlsx)

#chargement
mcar <- openxlsx::read.xlsx("provinces_mcar.xlsx", sheet=1, colNames=T, rowNames=T)

#transf. en matrice de distance
dmcar <- as.dist(mcar)
```



```
print(dmcar)
```

Nous typons `as.dist()` la structure en classe "`dist`" que sait manipuler R. Nous avons uniquement la partie triangulaire inférieure à l'affichage, sachant que la diagonale principale est nulle.

	Antsiranana	Majunga	Tuléar	Fianarantsoa	Tananarive
Majunga	859				
Tuléar	2038	1495			
Fianarantsoa	1522	979	516		
Tananarive	1109	566	930	414	
Toamasina	1287	926	1266	750	355

### 3.2 Positionnement multidimensionnel "classique" (1)

La fonction `cmdscale()` du package "`stats`" (installée et chargée automatiquement) réalise le positionnement multidimensionnel classique. Nous lui passons en entrée : la matrice de distance, le nombre de composantes à générer ( $k = 2$ ), nous souhaitons obtenir les valeurs propres (`eig = TRUE`), et disposer de la matrice B utilisée pour la diagonalisation (`x.ret = TRUE`) (cf. [cours](#), page 8).

```
#cmdscale
pmcar <- cmdscale(d=dmcar,k=2,eig=TRUE,x.ret=TRUE)
print(pmcar)
```

```
$points
      [,1]      [,2]
Antsiranana -1045.550101 -161.54973
Majunga     -439.277753 -181.15370
Tuléar       986.841325 -224.65764
Fianarantsoa 462.012191 -57.48386
Tananarive   41.511637  68.74778
Toamasina    -5.537299 556.09714

$eig
[1] 2.475205e+06 4.266607e+05 2.007539e+05 7.598011e-02 1.178884e-10 -
9.275464e+04

$x
      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
[1,] -2361398.00 -792703.7 1953992.3 924916.33 98733.17 176459.83
[2,] -792703.67 -699771.3 866386.7 397686.67 20021.50 208380.17
[3,] 1953992.33 866386.7 -2037505.3 -963365.33 -104301.50 284793.17
[4,] 924916.33 397686.7 -963365.3 -421737.33 10078.50 52421.17
[5,] 98733.17 20021.5 -104301.5 10078.50 99102.33 -123634.00
[6,] 176459.83 208380.2 284793.2 52421.17 -123634.00 -598420.33

$ac
[1] 0

$GOF
[1] 0.9081458 0.9352953
```

Nous observons dans la sortie R :

**\$points** représentent les coordonnées factorielles des observations. Nous en produirons la représentation graphique plus bas. Nous disposons de 2 colonnes parce que nous avons spécifié ( $k=2$ ) lors de l'appel de la fonction.



**\$eig** correspond aux valeurs propres de la matrice à diagonaliser B (**\$x**). Nous constatons que certaines valeurs propres sont négatives, B n'est pas semi-définie positive. Deux solutions - étudiées plus bas - sont possibles pour dépasser cet obstacle.

**\$ac** représente la correction apportée à la matrice à diagonaliser, qui représente une des pistes justement pour solutionner le problème des valeurs propres négatives. Nous y reviendrons (section 3.3). La matrice n'a pas été corrigée ici (**\$ac = 0**).

**\$GOF** indique la qualité de la représentation dans le plan ( $k = 2$ ). Deux valeurs sont fournies, la première correspond à  $GOF_1 = \frac{\lambda_1 + \lambda_2}{\sum_j |\lambda_j|} = 0.9081$  ; la seconde à  $GOF_2 = \frac{\lambda_1 + \lambda_2}{\sum_j \max(\lambda_j, 0)} = 0.9352$  (c.-à-d. en ne prenant en compte que les valeurs propres positives).

La solution de s'en tenir uniquement aux valeurs propres positives est souvent avancée dans la littérature. Elle est simple à mettre en œuvre et les résultats sont probants souvent. Dans notre cas, nous aurons effectivement :

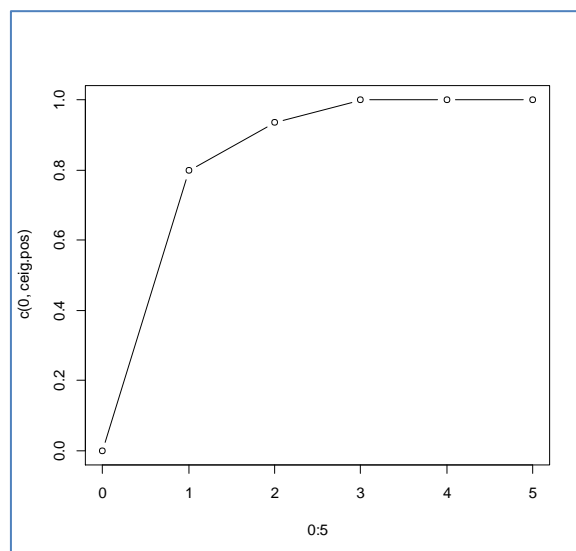
```
#qualité - uniquement v.p. positives
eig.pos <- pmcar$eig[pmcar$eig > 0]

#qualité relative
ceig.pos <- cumsum(eig.pos)/sum(eig.pos)
print(ceig.pos)

0.7977790 0.9352953 1.0000000 1.0000000 1.0000000
```

On apprécie mieux l'évolution en fonction du nombre d'axes dans un graphique :

```
#graphique
plot(0:5,c(0,ceig.pos),type="b")
```



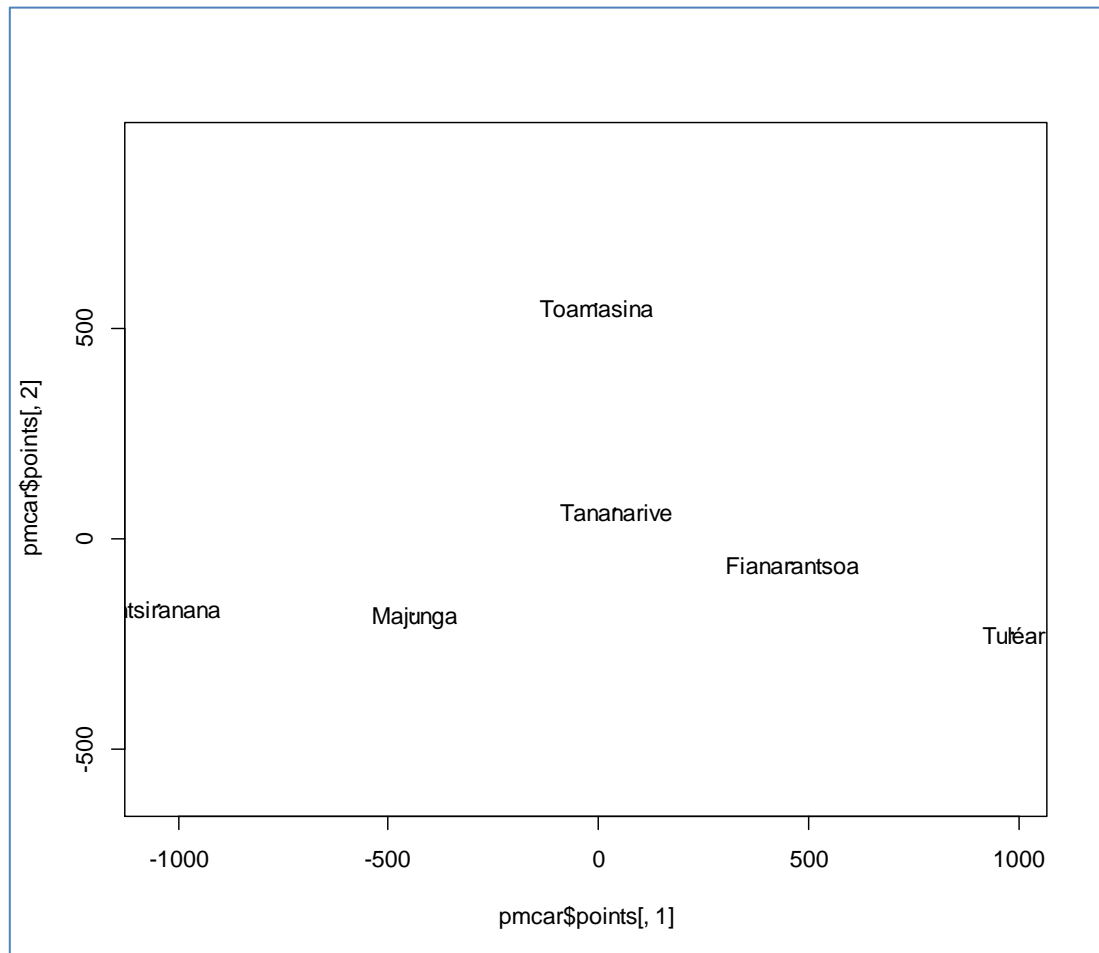
Le choix de ( $k = 2$ ) facteurs paraît légitime pour notre étude. Visualisons les villes dans le plan.



#### #graphique des points dans le plan

```
plot(pmcar$points[,1],pmcar$points[,2],type="p",cex=0.3,asp=1)  
text(pmcar$points[,1],pmcar$points[,2],label=rownames(mcar))
```

Le paramètre (`asp = 1`; aspect ratio) est fondamental dans l'appel de `plot()`. Les qualités de restitution en abscisse et ordonnée étant en lien direct avec les dispersions, il ne faut surtout pas que R fasse une mise à l'échelle automatique pour aligner les points min/max aux bords du graphique, ce qui fausserait leur perception. Avec (`asp = 1`), les unités en abscisse et ordonnées sont identiques.



Nous distinguons les positions relatives des villes. Et l'éloignement sur l'axe Antsiranana-Tuléar est autrement plus prononcé que sur l'axe Toamasina vs. les autres provinces. Mais nous constatons que notre graphique est pivoté par rapport aux positions réelles des villes (Figure 2).

Transposons le graphique en modifiant l'appel de `plot()` et `text()`.

#### #graphique "réarrangé"

```
plot(pmcar$points[,2],-pmcar$points[,1],type="p",cex=0.3,asp=1)  
text(pmcar$points[,2],-pmcar$points[,1],label=rownames(mcar))
```

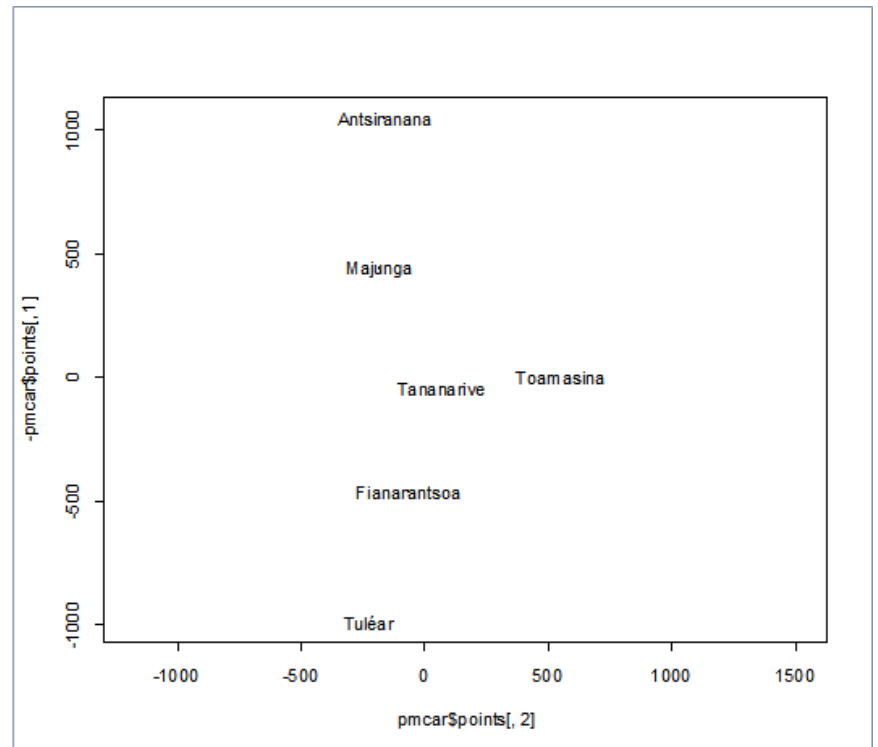


Figure 4 - Carte de Madagascar et positionnement des villes à partir des distances kilométriques

La représentation est assez fidèle. Il y a un axe de circulation nord-sud de Tuléar à Antsiranana qui passe par la capitale Antananarivo, que l'on retrouve avec la requête Google Maps (Figure 5).

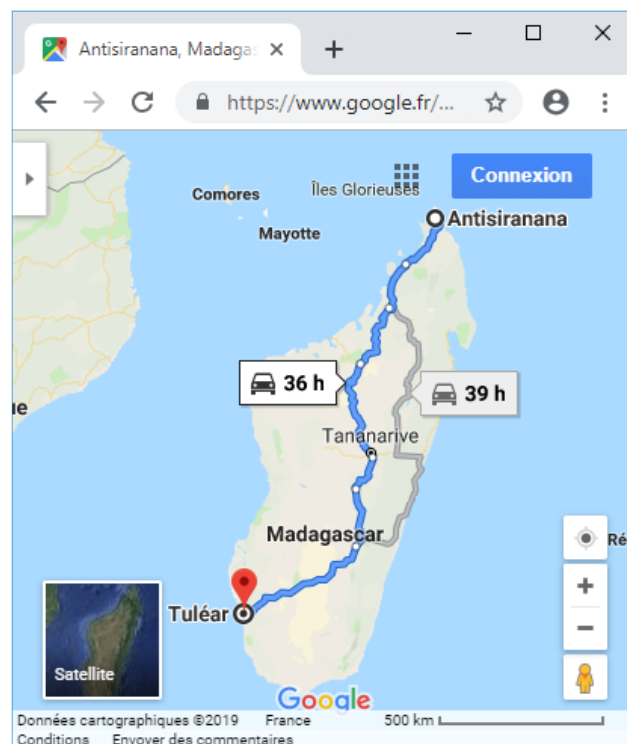


Figure 5 - Axe Nord-Sud (Antsiranana - Tuléar), passant par Tananarive



### 3.3 Positionnement multidimensionnel "classique" (2)

Pour dépasser le problème des valeurs propres négatives de la matrice B, une autre astuce simple consiste à ajouter une constante positive  $c$  aux éléments hors diagonale principale. Elle doit être la plus petite possible pour éviter de détériorer l'information. Sa valeur peut être déterminée analytiquement (Caillez, 1983 ; décrite dans Saporta 2006, cf. [support de cours](#)).

Avec `cmdscale()`, l'option (`add = TRUE`) permet de l'introduire.

```
#pour avoir uniquement des v.p. positifs
pmcar.bis <- cmdscale(d=dmcar,k=2,eig=TRUE,add=TRUE)
print(pmcar.bis)
```

```
$points
      [,1]      [,2]
Antsirana -1148.985060 -243.92285
Majunga   -495.487132 -181.47822
Tuléar     1080.180111 -310.05543
Fianarantsoa 523.516637 -70.42849
Tananarive  47.573111  130.22624
Toamasina  -6.797668  675.65874
```

```
$eig
[1] 3.008842e+06 6.670008e+05 3.860996e+05 8.842601e+04 2.473826e-10 -
5.222927e-11
```

```
$x
NULL
```

```
$ac
[1] 206.5948
```

```
$GOF
[1] 0.8856666 0.8856666
```

Toutes les valeurs propres (**\$eig**) sont positives (la dernière est une erreur de précision simplement). La qualité de représentation (**\$GOF**) dans le plan ( $k = 2$ ) est **88.56%**. Et la constante ajoutée à la matrice B est  $c = 206.5948$ .

On se rend compte que cela ne modifie pas la nature des résultats. Les positions relatives des villes restent fidèles dans le plan (Figure 6).

```
#graphique avec correction
plot(pmcar.bis$points[,2],-pmcar.bis$points[,1],type="p",cex=0.3,asp=1)
text(pmcar.bis$points[,2],-pmcar.bis$points[,1],label=rownames(mcar))
```



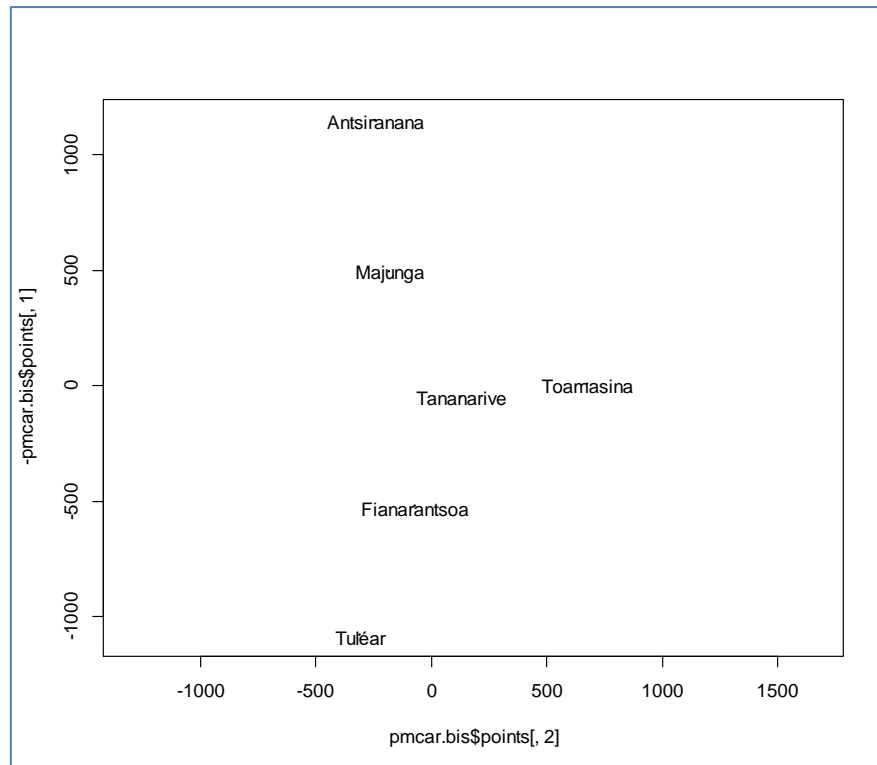


Figure 6 - Positionnement après modification de la matrice à diagonaliser

### 3.4 Positionnement multidimensionnel non-métrique

Le MDS classique tient compte des valeurs des distances pour positionner les objets. Parfois, nous ne disposons que de leur ordre c.-à-d. nous savons que Fianarantsoa est plus proche de Tananarive que Tuléar, etc. Le MDS non-métrique permet de traiter ce type d'information.

Dans cette section, nous transformons la matrice de distance en rangs, et nous lançons le MDS non-métrique. Nous verrons alors ce qu'il en sera des positions relatives des villes.

La transformation de la matrice de distances en matrice de rangs nécessite plusieurs étapes. Tout d'abord, nous linéarisons la structure de distance.

```
#linéariser la matrice de distance
d <- unclass(dmcar)
print(d)
[1] 859 2038 1522 1109 1287 1495 979 566 926 516 930 1266 414 750 355
attr(,"Labels")
[1] "Antsiranana" "Majunga" "Tuléar" "Fianarantsoa" "Tananarive"
"Тоamasina"
attr(,"Size")
[1] 6
attr(,"call")
as.dist.default(m = mcar)
attr(,"Diag")
[1] FALSE
attr(,"Upper")
```



```
[1] FALSE
```

Nous calculons les rangs à partir de ce vecteur **d**.

```
#calcul des rangs des distances
rg <- rank(d)
print(rg)
```

```
6 15 14 10 12 13 9 4 7 3 8 11 2 5 1
```

C.-à-d. 355 correspond à la plus petite valeur des distances (rang = 1), puis 414 (rang = 2), etc. La distance entre les deux villes les plus éloignées est 2038 (rang = 15).

Il faut recharger maintenant ces valeurs dans une matrice symétrique. Il y a certainement des manières plus efficaces de le faire sous R, mais j'ai préféré privilégier la lisibilité du code en utilisant une double boucle très basique.

```
#matrice des rangs
k <- 0
rmcar <- mcar
for (i in 1:(nrow(mcar)-1)){
  for (j in (i+1):ncol(mcar)){
    k <- k+1
    rmcar[i,j] <- rg[k]
    rmcar[j,i] <- rg[k]
  }
}
print(rmcar)
```

	Antsiranana	Majunga	Tuléar	Fianarantsoa	Tananarive	Toamasina
Antsiranana	0	6	15	14	10	12
Majunga	6	0	13	9	4	7
Tuléar	15	13	0	3	8	11
Fianarantsoa	14	9	3	0	2	5
Tananarive	10	4	8	2	0	1
Toamasina	12	7	11	5	1	0

Les bonnes valeurs sont au bon endroit, il était important de le vérifier. Nous transformons cette matrice en structure "dist".

```
#classe dist
drmcars <- as.dist(rmcar)
print(drmcars)
```

	Antsiranana	Majunga	Tuléar	Fianarantsoa	Tananarive
Majunga	6				
Tuléar	15	13			
Fianarantsoa	14	9	3		
Tananarive	10	4	8	2	
Toamasina	12	7	11	5	1

Que nous passons à la fonction `isoMDS()` du package "MASS" (installé automatiquement, mais qu'il faut charger) pour réaliser le positionnement multidimensionnel non-métrique.



```
#MASS
library(MASS)

#isoMDS - MDS non-métrique
pmcar.iso <- isoMDS(drmcar,k=2)
print(pmcar.iso)

$points
      [,1]      [,2]
Antsiranana -8.4162115 -3.5695897
Majunga     -4.8373326  1.3166077
Tuléar       6.7707461 -4.3497154
Fianarantsoa 5.0716331  0.5185552
Tananarive   0.7051342  1.5173974
Toamasina    0.7060307  4.5667448

$stress
[1] 1.846667e-14
```

La représentation est de très bonne qualité puisque le STRESS (**\$stress**) est quasiment nul ([support de cours](#), page 20). Nous disposons des coordonnées des villes avec **\$points**. Nous nous empressons de les placer dans le plan, en veillant à corriger le sens du graphique (Figure 7).

```
#graphique
plot(pmcar.iso$points[,2],-pmcar.iso$points[,1],type="p",cex=0.3,asp=1)
text(pmcar.iso$points[,2],-pmcar.iso$points[,1],label=rownames(mcar))
```

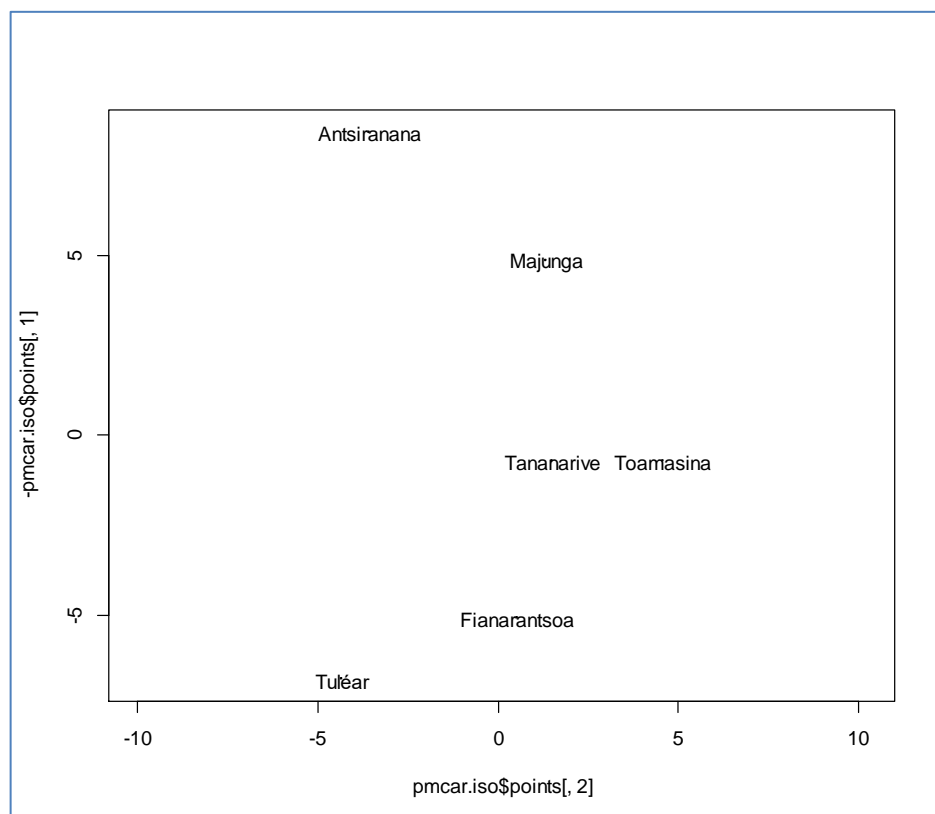


Figure 7 - Position relatives des villes - MDS non-métrique basé sur les rangs



L'information est préservée en passant aux rangs. C'est le principal résultat que nous retiendrons. Les positions relatives sont respectées. Nous constatons néanmoins qu'en occultant les valeurs (et les écarts) des distances, nous observons un rapprochement entre certaines villes (Tananarive vs. Toamasina, Tuléar vs. Fianarantsoa).

## 4 Conclusion

Le positionnement multidimensionnel a de multiples applications (comme nous le verrons dans un prochain document). Dans ce tutoriel, j'ai repris l'exemple particulièrement usité des positions relatives des villes pour illustrer sa mise en œuvre sous R. Les résultats sont probants je trouve, nous avons pu reconstituer la carte de Madagascar (très partiellement, j'en conviens) à partir de la matrice de distances entre les villes principales des 6 provinces qui constituent l'île.

## 5 Références

R. Rakotomalala, "[Positionnement Multidimensionnel - Diapos](#)", avril 2019.