Xinming Zhang

MDS

12/02/2021

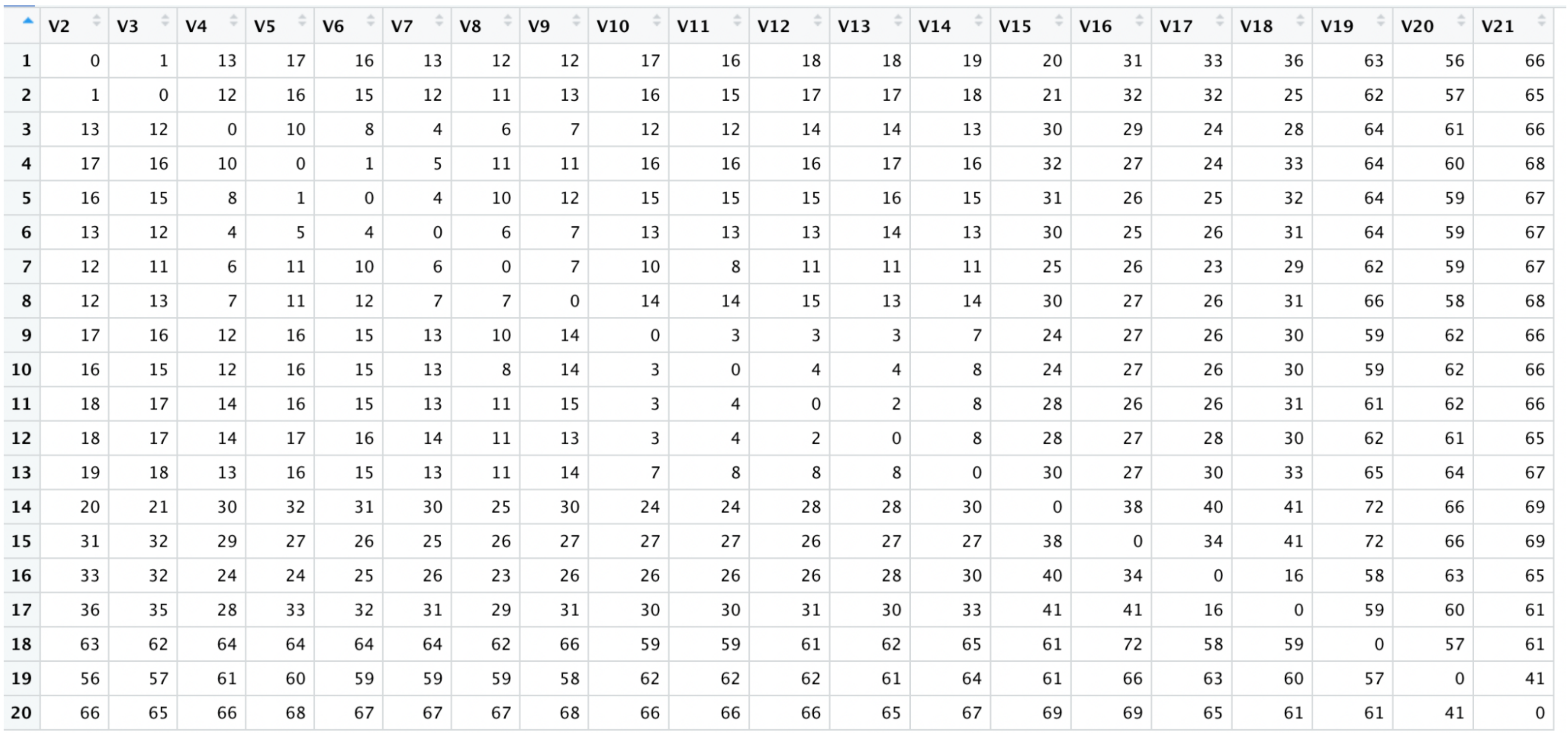
This document will be about multidimensional scaling, and the data set I chose is #29 from the link provided in the assignment. The data set is about mutation distances. According to John Hartigan, the distance between two species is the number of positions in the protein molecule cytochrome-c, where the two species have different amino acids.

The dataset table is showing following:

Text table:

"Species"	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
"Man"	0	1	13	17	16	13	12	12	17	16	18	18	19	20	31	33	36	63	56	66
"Monkey"	1	0	12	16	15	12	11	13	16	15	17	17	18	21	32	32	25	62	57	65
"Dog"	13	12	0	10	8	4	6	7	12	12	14	14	13	30	29	24	28	64	61	66
"Horse"	17	16	10	0	1	5	11	11	16	16	16	17	16	32	27	24	33	64	60	68
"Donkey"	16	15	8	1	0	4	10	12	15	15	15	16	15	31	26	25	32	64	59	67
"Pig"	13	12	4	5	4	0	6	7	13	13	13	14	13	30	25	26	31	64	59	67
"Rabbit"	12	11	6	11	10	6	0	7	10	8	11	11	11	25	26	23	29	62	59	67
"Kangaroo"	12	13	7	11	12	7	7	0	14	14	15	13	14	30	27	26	31	66	58	68
"Peking Duck"	17	16	12	16	15	13	10	14	0	3	3	3	7	24	27	26	30	59	62	66
"Pigeon"	16	15	12	16	15	13	8	14	3	0	4	4	8	24	27	26	30	59	62	66
"Chicken"	18	17	14	16	15	13	11	15	3	4	0	2	8	28	26	26	31	61	62	66
"King Penguin"	18	17	14	17	16	14	11	13	3	4	2	0	8	28	27	28	30	62	61	65
"Snapping Turtle"	19	18	13	16	15	13	11	14	7	8	8	8	0	30	27	30	33	65	64	67
"Rattlesnake"	20	21	30	32	31	30	25	30	24	24	28	28	30	0	38	40	41	72	66	69
"Tuna"	31	32	29	27	26	25	26	27	27	27	26	27	27	38	0	34	41	72	66	69
"Screwworm Fly"	33	32	24	24	25	26	23	26	26	26	26	28	30	40	34	0	16	58	63	65
"Moth"	36	35	28	33	32	31	29	31	30	30	31	30	33	41	41	16	0	59	60	61
"Baker's Mold"	63	62	64	64	64	64	62	66	59	59	61	62	65	61	72	58	59	0	57	61
"Bread Yeast"	56	57	61	60	59	59	59	58	62	62	62	61	64	61	66	63	60	57	0	41
"Skin Fungus"	66	65	66	68	67	67	67	68	66	66	66	65	67	69	69	65	61	61	41	0

Table in R:



	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	V21
1	0	1	13	17	16	13	12	12	17	16	18	18	19	20	31	33	36	63	56	66
2	1	0	12	16	15	12	11	13	16	15	17	17	18	21	32	32	25	62	57	65
3	13	12	0	10	8	4	6	7	12	12	14	14	13	30	29	24	28	64	61	66
4	17	16	10	0	1	5	11	11	16	16	16	17	16	32	27	24	33	64	60	68
5	16	15	8	1	0	4	10	12	15	15	15	16	15	31	26	25	32	64	59	67
6	13	12	4	5	4	0	6	7	13	13	13	14	13	30	25	26	31	64	59	67
7	12	11	6	11	10	6	0	7	10	8	11	11	11	25	26	23	29	62	59	67
8	12	13	7	11	12	7	7	0	14	14	15	13	14	30	27	26	31	66	58	68
9	17	16	12	16	15	13	10	14	0	3	3	3	7	24	27	26	30	59	62	66
10	16	15	12	16	15	13	8	14	3	0	4	4	8	24	27	26	30	59	62	66
11	18	17	14	16	15	13	11	15	3	4	0	2	8	28	26	26	31	61	62	66
12	18	17	14	17	16	14	11	13	3	4	2	0	8	28	27	28	30	62	61	65
13	19	18	13	16	15	13	11	14	7	8	8	8	0	30	27	30	33	65	64	67
14	20	21	30	32	31	30	25	30	24	24	28	28	30	0	38	40	41	72	66	69
15	31	32	29	27	26	25	26	27	27	27	26	27	27	38	0	34	41	72	66	69
16	33	32	24	24	25	26	23	26	26	26	26	28	30	40	34	0	16	58	63	65
17	36	35	28	33	32	31	29	31	30	30	31	30	33	41	41	16	0	59	60	61
18	63	62	64	64	64	64	62	66	59	59	61	62	65	61	72	58	59	0	57	61
19	56	57	61	60	59	59	59	58	62	62	62	61	64	61	66	63	60	57	0	41
20	66	65	66	68	67	67	67	68	66	66	66	65	67	69	69	65	61	61	41	0

There are two species in the dataset, and the number at (i,j) box means the mutation distance between species i and species j. For example, at (1,4) it is 17. That means the mutation distance between species man and species horse is 17. As we see, the table we have now is in distance form, so we do not need to standardize or normalize anymore. Then we can use R to work with multidimensional scaling.

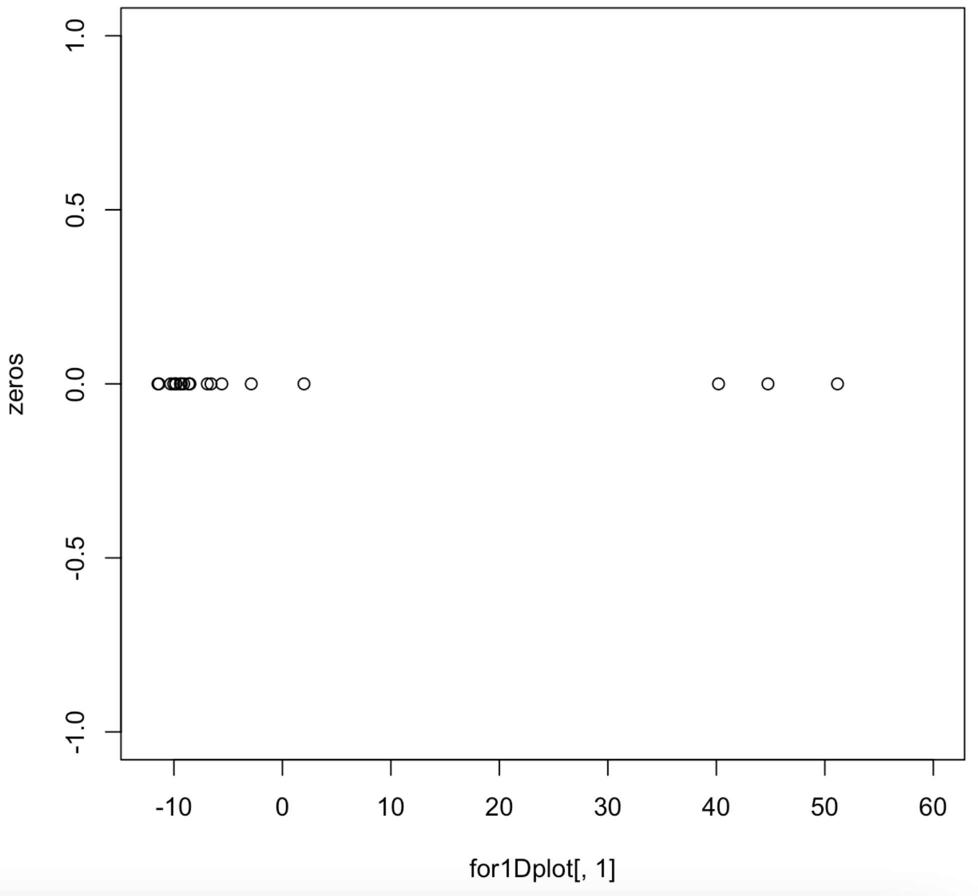
Here is the R code:

```
#read the dataset from cvs
mydata <- read.csv("file29.csv",header=FALSE)
#col.names=FALSE

#data <- subset(mydata, select = -V1 )
#remove the first column which is names
d <- mydata[1:20,2:21]
#set name as the names of the rows and columns
name <- mydata[,1]
#new dataset without last three points
newD <- d[-c(18,19,20),-c(18,19,20)]
#a vector of 20 zeros for plotting 1D model
zeros <- c(1:20)*0

#d_n <- normalize(d,method="standardize",margin=2)
#col.names=FALSE
#d2 <- as.matrix(data)
#d3 <- dist(data)

#original 1D model
model1 <- cmdscale(d, k =1,eig=TRUE)
#eigenvalue of 1D model
eig1 <- cmdscale(d,k=1,eig=TRUE)$eig
#Goodness of fit of 1D model
GOF1 <- cmdscale(d,k=1,eig=TRUE)$GOF

#new 1D model
model1_1 <- cmdscale(newD, k =1,eig=TRUE)
#eigenvalue of new 1D model
eig1_1 <- cmdscale(newD,k=1,eig=TRUE)$eig
#Goodness of fit of new 1D model
GOF1_1 <- cmdscale(newD,k=1,eig=TRUE)$GOF

#original 2D model
model2 <- cmdscale(d, k =2,eig=TRUE)
#eigenvalue of 2D model
eig2 <- cmdscale(d,k=2,eig=TRUE)$eig
#Goodness of fit of 2D model
GOF2 <- cmdscale(d,k=2,eig=TRUE)$GOF

#new 2D model
model2_1 <- cmdscale(newD, k =2)
#eigenvalue of 2D new model
eig2_1 <- cmdscale(newD,k=2,eig=TRUE)$eig
#Goodness of fit of new 2D model
GOF2_1 <- cmdscale(newD,k=2,eig=TRUE)$GOF

#original 3D model
model3 <- cmdscale(d, k =3, eig=TRUE)
#eigenvalue of 3D model
eig3 <- cmdscale(d,k=3,eig=TRUE)$eig
#Goodness of fit of 3D model
```

```
GOF3 <- cmdscale(d,k=3,eig=TRUE)$GOF

#new 3D model
model3_1 <- cmdscale(newD, k =3, eig=TRUE)
#eigenvalue of 3D new model
eig3_1 <- cmdscale(newD,k=3,eig=TRUE)$eig
#Goodness of fit of new 3D model
GOF3_1 <- cmdscale(newD,k=3,eig=TRUE)$GOF

…
#Some code are deleted for saving the space, it would the 4D model to 7D model, and code would be the similar

#original 8D model
model8 <- cmdscale(d, k =8, eig=TRUE)
#Goodness of fit of 8D model
GOF8 <- cmdscale(d,k=8,eig=TRUE)$GOF

#new 8D model
model8_1 <- cmdscale(newD, k =8, eig=TRUE)
#Goodness of fit of new 8D model
GOF8_1 <- cmdscale(newD,k=8,eig=TRUE)$GOF

#plot the 1D model of distance with names
for1Dplot <- data.frame(model1$points,0)
for1Dplot_1 <- data.frame(model1_1$points,0)
library(wordcloud)
plot(for1Dplot[,1],zeros,xlim=c(-12,60))
#textplot(for1Dplot[,1],zeros, name,xlim=c(-12,60),cex=0.6)
textplot(for1Dplot[,1],zeros,
        gsub("(^[^\\s]+\\s{1})","",
            name,perl=TRUE),
        xlim=c(-12,60),ylim=c(-10,10),
        cex=0.6)
plot(for1Dplot_1, ylim=c(-12,30),xlim=c(-12,30))



#plot 2D model of distances with names
textplot(model2$points[,1],model2$points[,2],
        gsub("(^[^\\s]+\\s{1})","",
            name,perl=TRUE),
        asp=1,xlim=c(-40,60),ylim=c(-40,60),
        cex=0.6)
plot(model2$points,
    ylim=c(-40,60),xlim=c(-40,60),
    ylab="y",xlab="x")
plot(model2$eig)
plot(model2_1,
    ylim=c(-20,25),xlim=c(-20,25),
    ylab="y",xlab="x")
textplot(model2_1[,1],model2_1[,2],
        gsub("(^[^\\s]+\\s{1})","",
            name,perl=TRUE),
        asp=1,ylim=c(-20,25),xlim=c(-20,25),
        cex=0.6)
```

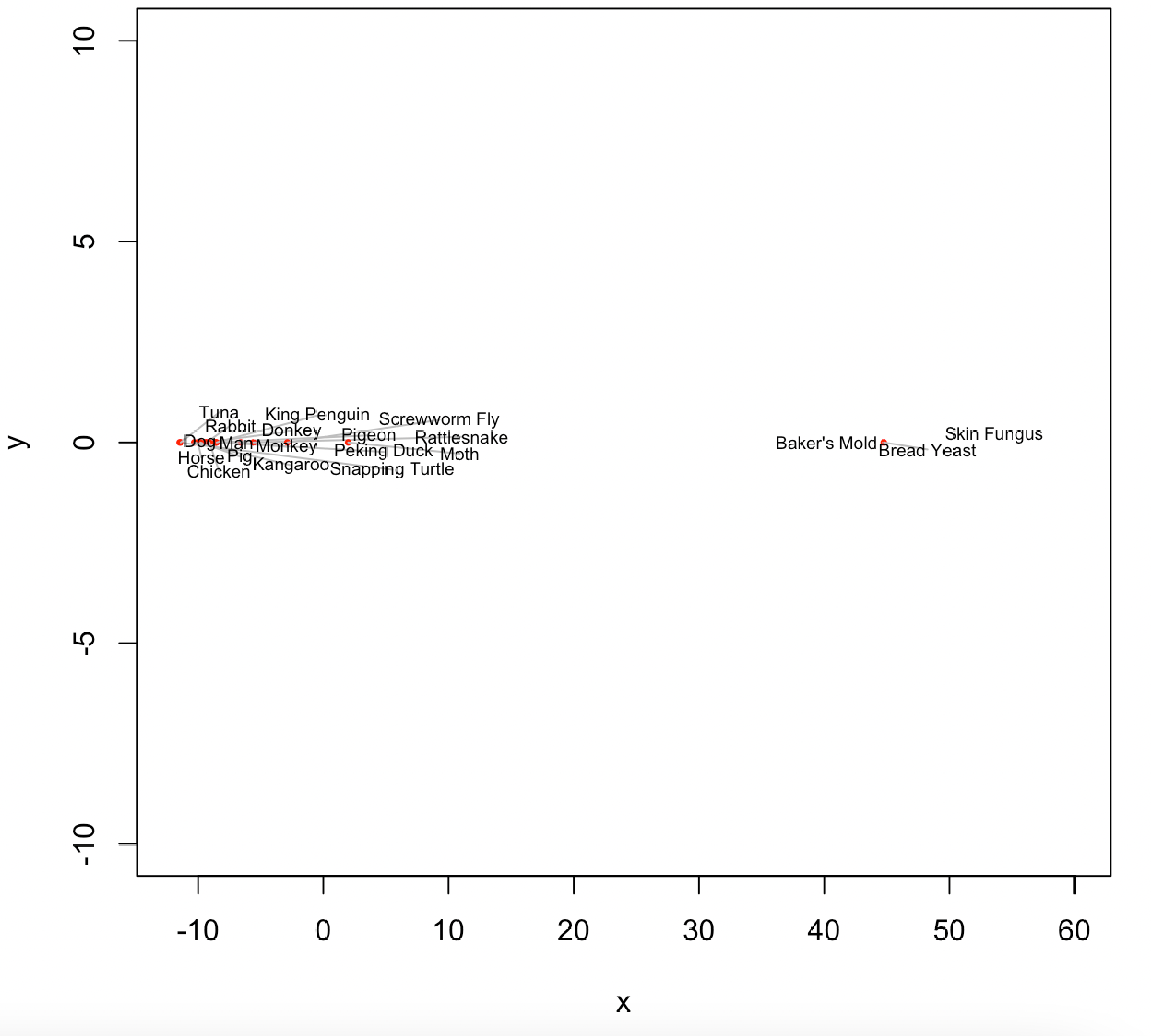
```
#show points of 3D model
model3$points
#plot the eigenvalues of original distances and new ones
plot(model1$eig)
plot(model1_1$eig)

#plot original distance with dist function
plot(as.matrix(d),as.matrix(dist(model1$points)))
abline(0,1)
plot(as.matrix(d),as.matrix(dist(model2$points)))
abline(0,1)
plot(as.matrix(d),as.matrix(dist(model3$points)))
abline(0,1)
#plot new distance with dist function
plot(as.matrix(newD),as.matrix(dist(model2_1)))
abline(0,1)

#mean absolute difference
mean(abs(as.matrix(d)-as.matrix(dist(model1$points))))
…
mean(abs(as.matrix(d)-as.matrix(dist(model8$points))))

mean(abs(as.matrix(newD)-as.matrix(dist(model1_1$points))))
…
mean(abs(as.matrix(newD)-as.matrix(dist(model8_1$points))))

#correlation with every species
 cor <- c(cor(model8$points[,1],d[,1]),
         …
          cor(model8$points[,1],d[,20]))
 as.matrix(cor)
```

Then we can generate the code get following information:
At first, we can get our 1D model as below:



1D model of the original distances(with no names)

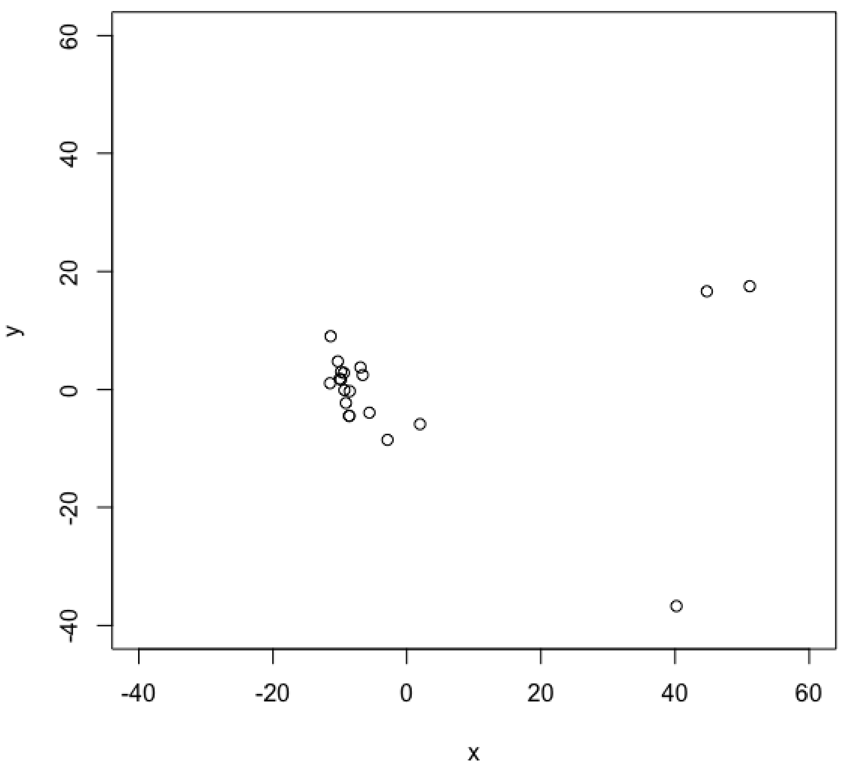
The model above can tell us some information already. We can see there are three data points that are far away from the “main” group on the left. From looking at the dataset in the beginning, without putting the names on the point, we can still be confident that these three “species” are Baker’s Mold, Bread Yeast, and Skin Fungus. In the table, they are the three that are really different from any other species. It does make sense, because they are not considered as “animals”, and other species in the dataset are all “animals” or human. Then we can proof our hypothesis by plotting the named 1D model like below:



Named 1D model of original distances

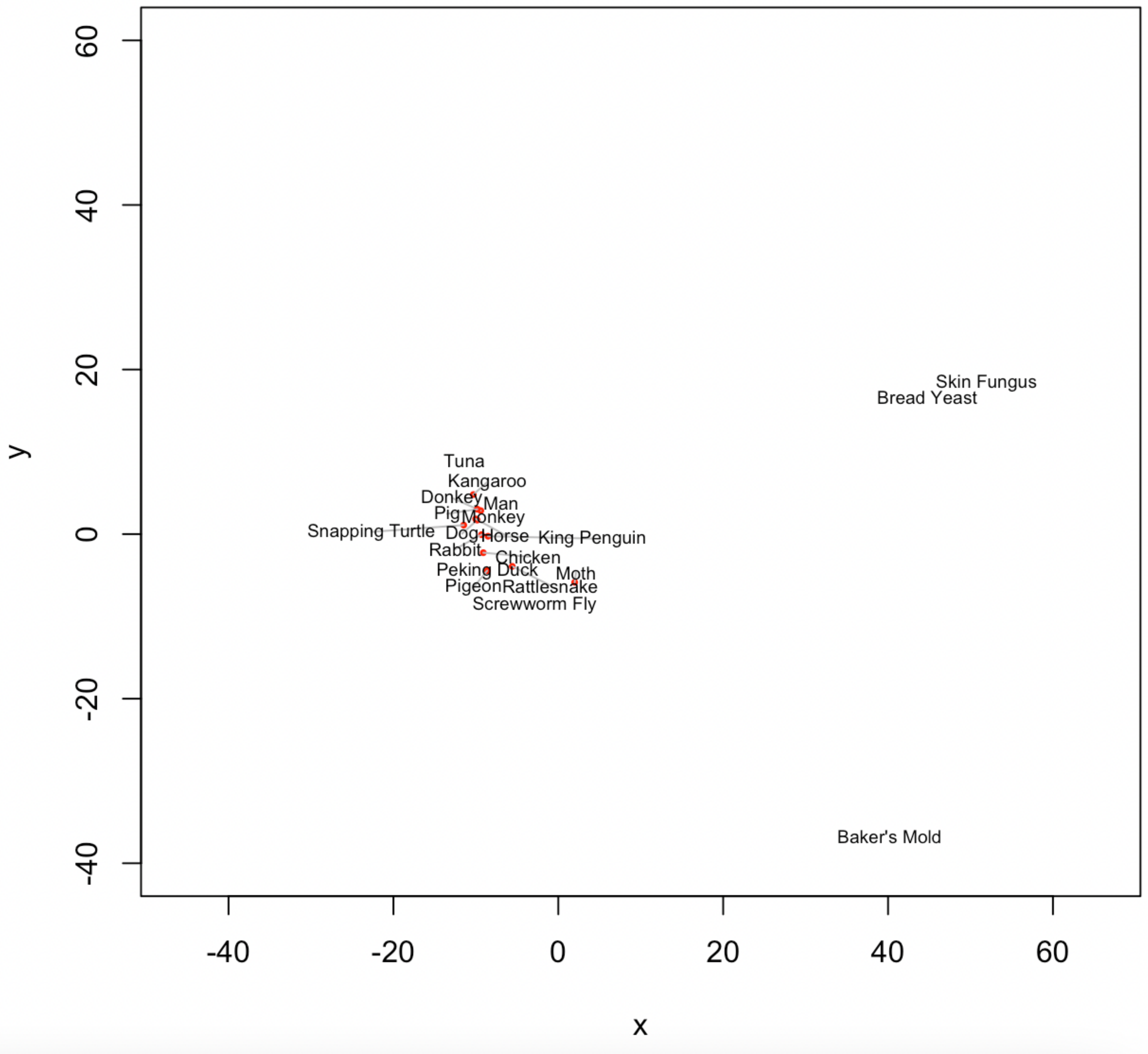
Our hypothesis about those species is right. This also showed our explanation is right, because they are supposed to be different from “animals” in the dataset which are on our 1D model. Then, if we want to discuss more about only animal species dataset, we can simply remove those three data points. I will talk about it after the 2D and 3D models of the original distances dataset.

Then, we can move on to 2D model, and the graph without names is shown as following:



2D model without names

From looking at the 2D model, we can get some similar information as the 1D model, and there is some more new information. Then, we can see the names model just to clear which species we are talking about.



Named 2D model of original distances

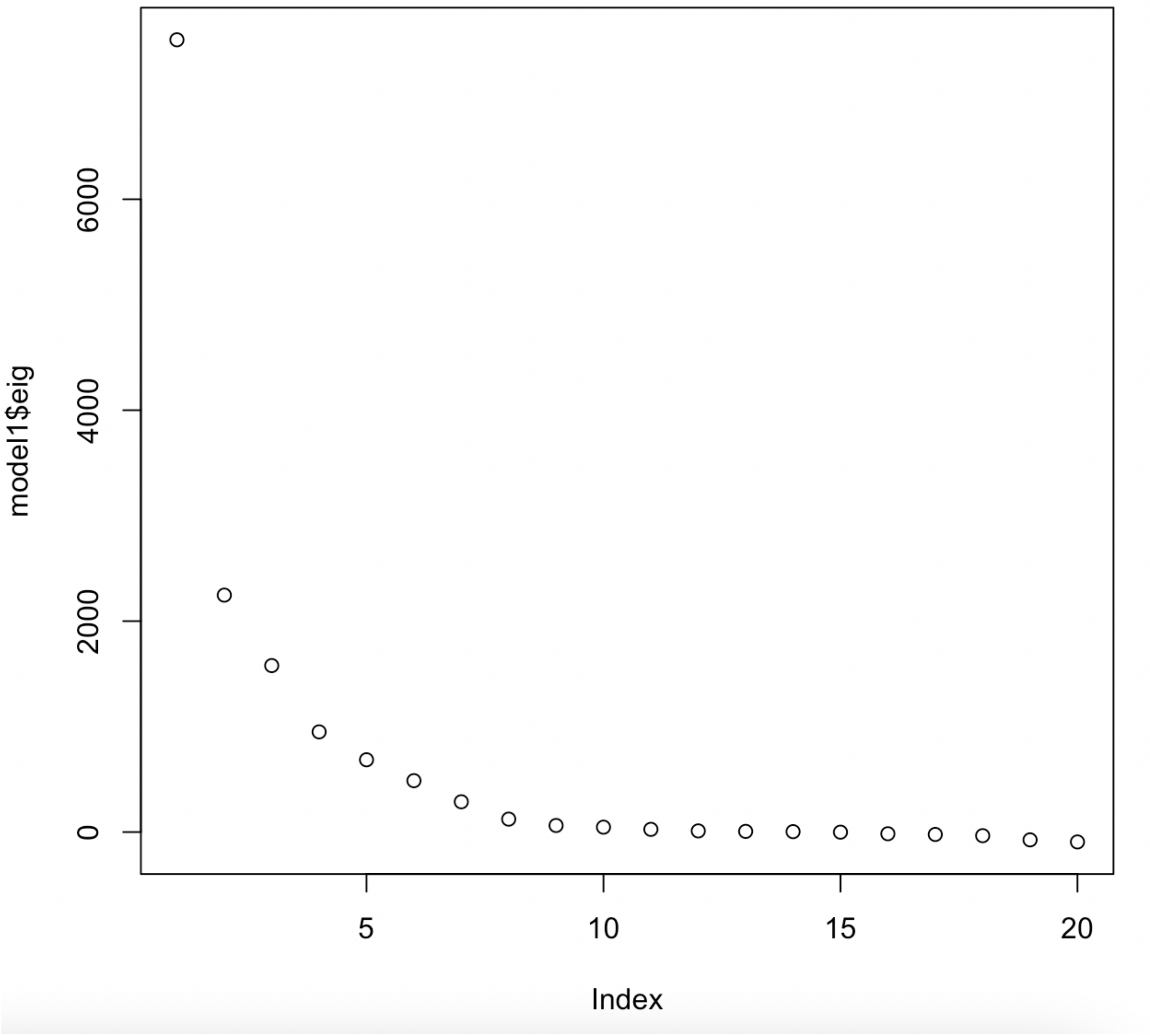
As we discussed in the 1D model, the non_animal species are still far away from the “main” group. One interesting thing is Baker’ Mold is also far away from Bread Yeast and Skin Fungus. After checking the difference among them, everything makes more sense. Baker’s Mold is a mold which is considered as multi celled fungi. On the other hand, Bread Yeast and Skin Fungus are considered as single celled fungi. Those species are in the same group which is fungi, but mold has more cells than yeast or fungus. This also tells us, the 2 dimensional model could give us more information, even if we find some in the 1D model.

The other thing is about the Moth and Screwworm fly. They are also not in the big group on the left. Compared to the three species we just talked about, they are much closer to the group. However, they must have some significant differences with other species. From looking at the names, we can see that they are insects, and all other species are non-insects. Our distances are about mutation, so it does make sense. Every type of animal will have a different mutation system for sure. Also on the top left, Tuna could be seen clearly. At first, I thought it was caused by it living in the water (sea), but there is Snapping Turtle in the dataset. Turtle is among the big group by just looking at the model. Then I think it is caused by Tuna being a fish, but turtle is an amphibian.

There are some interesting things we can see by just looking at the 2D model, and the main thing is we can distinguish some different types of species from the gap. However, if we want to distinguish them better, we have to increase the dimensions to know more about it.

After talking about 1D and 2D models, we have already received lots of useful information, and then I went ahead and did 3D to 8D models. Before I talk about those I will show the eigenvalues of the distance matrix.

Here is the eigenvalues plot for the 1D model, and they are the same for the models I did.

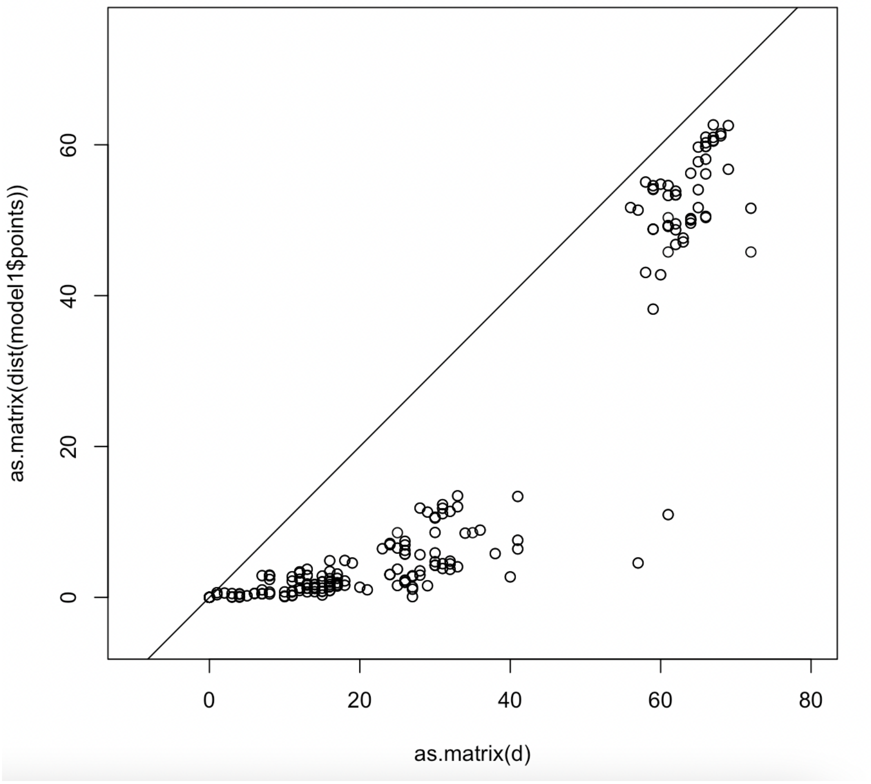


Eigenvalue plot

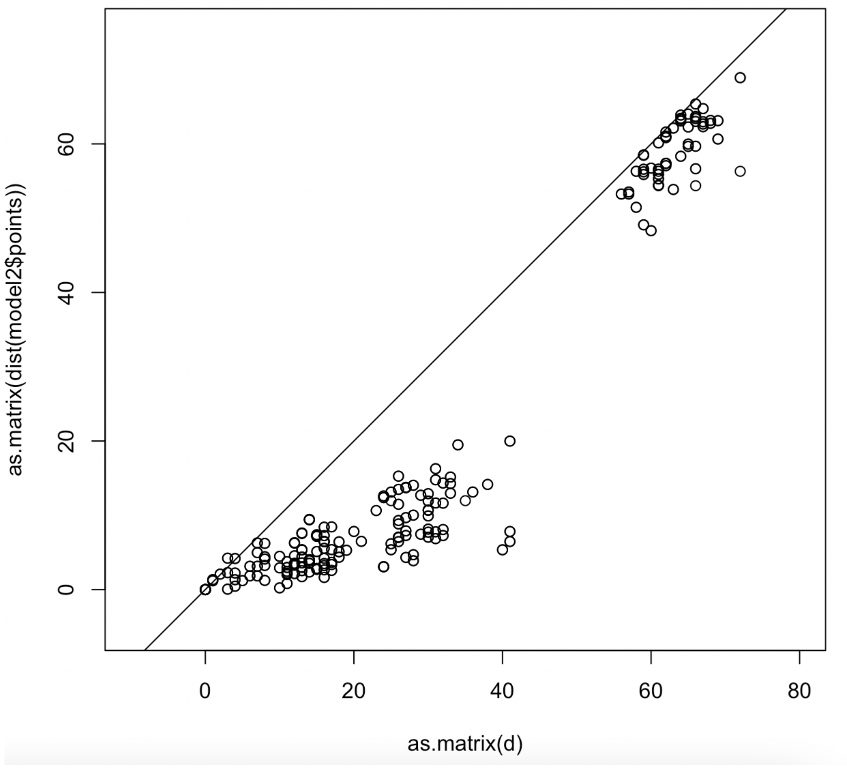
The eigenvalues plot looks good for the first 14 indexes (non-negative), because it is going to zero in those indexes. That means those models are good and reasonable. We can also do some more tests to see if our models are getting better when we increase the dimensions.

On the other hand, there is a big drop from the first dimension to the second dimension. The drops of further dimensions, the gaps are not as big as the first one anymore. That means the 2D model is a lot better than the 1D model, but when we increase the dimension further, the model is not getting significantly better anymore.

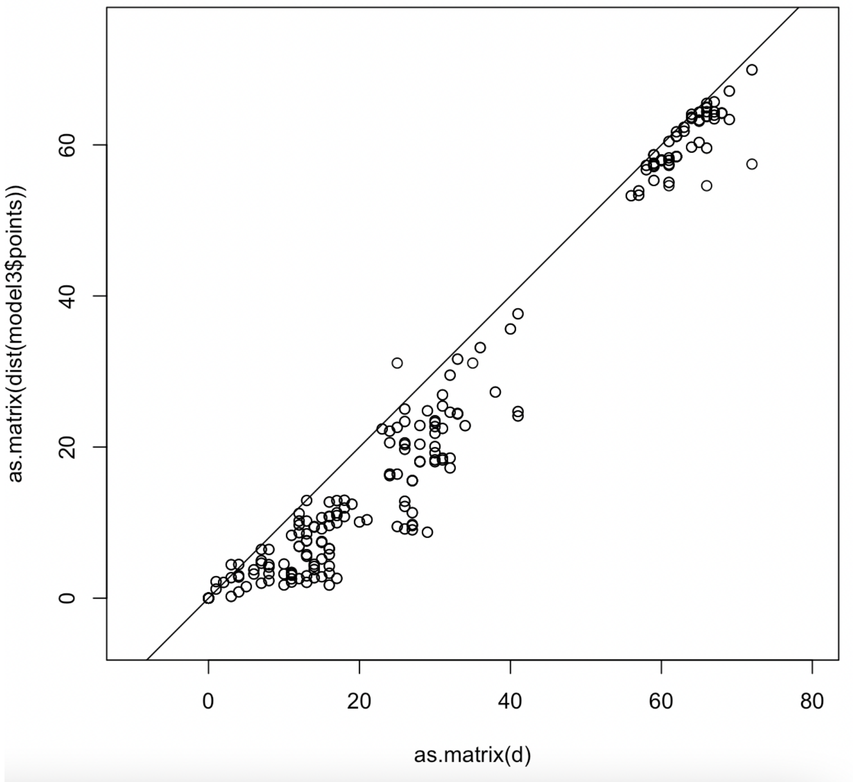
The eigenvalues after index 14 shows that the matrix is not using euclidean distance, because my distance matrix is the input matrix. The distance depends on the distance between two species is the number of positions in the protein molecule cytochrome-c, so it makes sense that it is not euclidean. Then the later analysis will show the same as well.



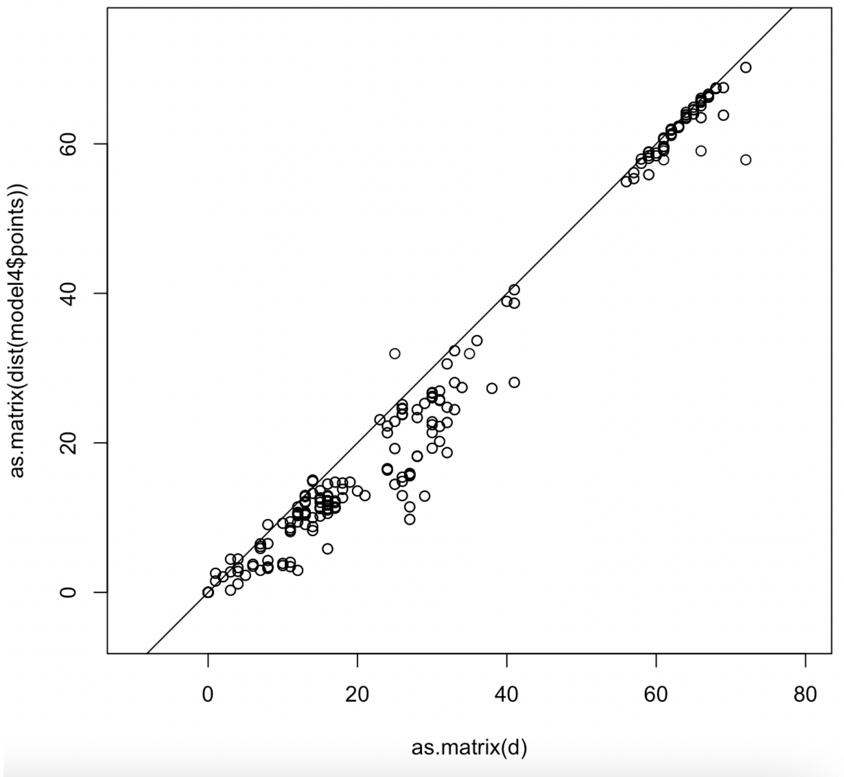
1D vs dist(1D)



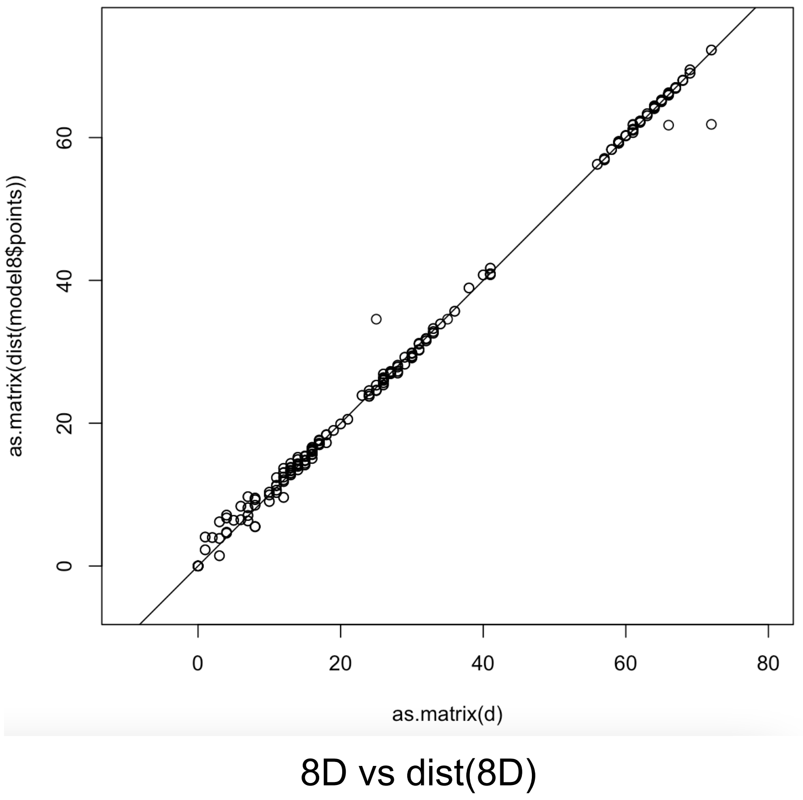
2D vs dist(2D)



3D vs dist(3D)



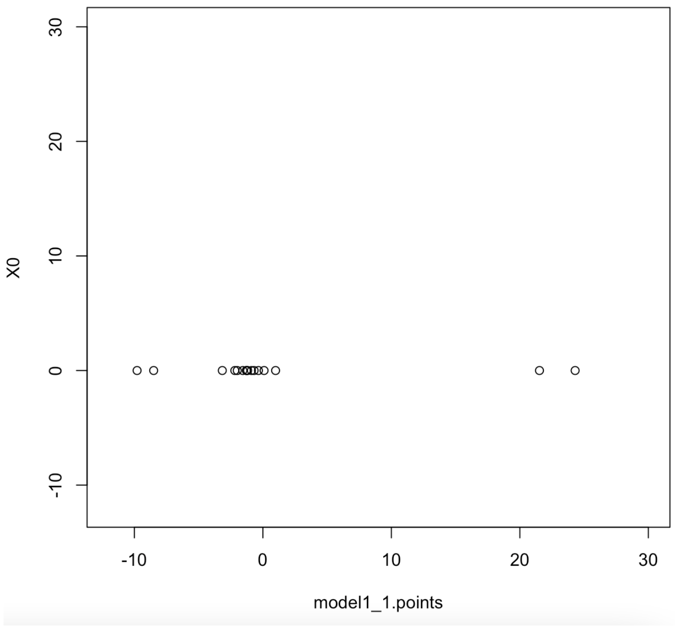
4D vs dist(4D)



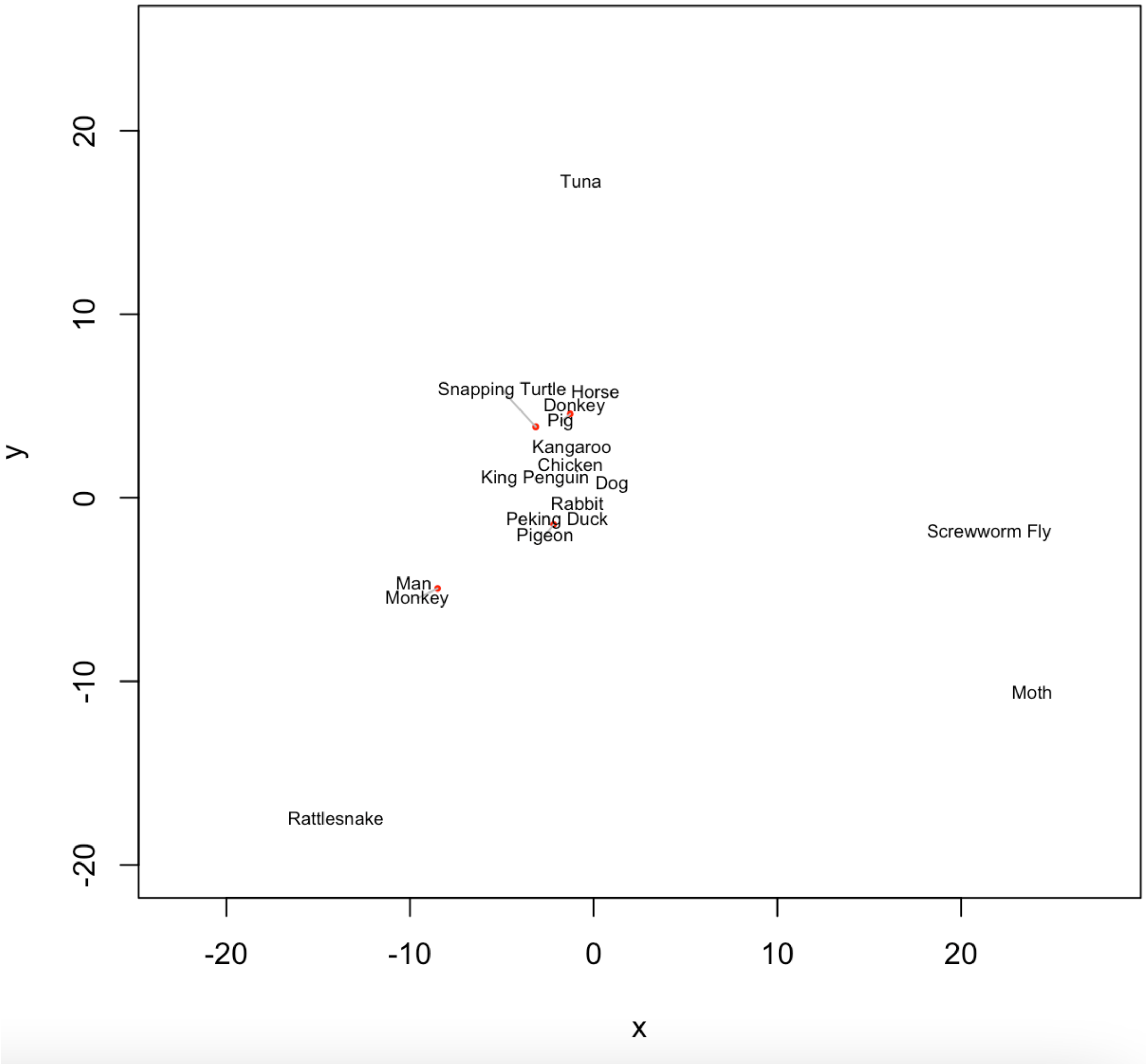
8D vs dist(8D)

As we see above, it is getting better and better when the dimension is increased. Then when we come to the 8 dimensional model, the model is getting closer to the line, but there are three data points that are a bit far from the y=x line. I would think that is caused by at least two of them being in the group “fungi”. The plot looks better when the dimension is increasing, but there are few points above the y=x line, so it also proved that the matrix is euclidean.

After investing those models, we can move on to the “new” dataset which is the one without those fungi (Baker’s Mold, Bread Yeast, and Skin Fungus), then I did model 1 through 8 (1-dimensional to 8 dimensional). Also, I made new plots for new 1D and 2D models.



New model 1 plot



New 2D model plot

Looking at the new plots, we can see a little bit more information even though it is the same plot but without those three fungi. Our thoughts about Tuna, Moth, and Fly are proofed again by those two plots, because we can see closer and clearer. Then we can find some more interesting findings. The first one is Rattlesnake, and it belongs to reptiles. Turtles are also considered reptiles, but they are also amphibians. The snapping turtle is a lot closer to the big group in the middle, but Rattlesnake is the other way. I think that is caused by amphibians, but I'm not sure (only a guess). After the snake, we see man and monkey. They are close to each other, but they are not that close to the group in the middle. Monkey and man are primates, so that could be the factor influencing the result. That makes sense, they would have absolutely different mutation systems than any other animals. I would think the new dataset has better fitting than the original one, because we can see it clearer and get more information. Then I went ahead to check the GOF (Goodness of fit), and the result was different than I thought.

GOF table for original distance

Dimensions	1	2	3	4	5	6	7	8
GOF(3 digits)	0.527, 0.536	0.684, 0.696	0.795, 0.808	0.861, 0.876	0.909, 0.925	0.943, 0.960	0.964, 0.980	0.972, 0.989

GOF table for new distances

Dimensions	1	2	3	4	5	6	7	8
GOF(3 digits)	0.373, 0.386	0.600, 0.620	0.758, 0.783	0.879, 0.908	0.921, 0.952	0.940, 0.971	0.952, 0.983	0.958, 0.990

There are two GOFs for each dimension, so it also proved the matrix is not euclidean.

Compare two GOF tables of two dataset from 1 to 8 dimensions. For the first three dimensions, the original distances have larger GOFs than the new ones. Then, for 4 and 5 dimensions, the new ones have the larger GOFs. From 6, 7 and 8 dimensions, original ones have larger lower bound, and new ones have larger higher bound. The result is different from what I was thinking. They are really good while we are increasing the dimensions. On the other hand, they could be good in different ways.

The difference of GOFs for the new dataset is larger than the original one, because it is not as “simple” as the original one. Even the last points are far away from the main group, but they make the model “simpler”. The new dataset is missing three data points, and it is making the number of various dimensions go up. Then we need more demison models to analyze it to get a better result. However, from looking at the 2D model plot, we can get some information out from looking at either dataset.

The mean absolute difference between distance matrix and dist() function could show how close the points are.

Mean absolute difference between original distances and dist() function

Dimensions	1	2	3	4	5	6	7	8
MAD	13.79208	9.623504	5.529233	3.404789	1.989259	1.146627	0.6361978	0.5460138

Mean absolute difference between new distances and dist() function

Dimensions	1	2	3	4	5	6	7	8
MAD	9.410564	5.279504	2.916215	1.054024	0.6037273	0.4866692	0.4790979	0.5256918

From looking at both tables, the GOF of original distances is then the dimension increases, and it is really good for both 8-dimensional models. Then, we know the 8-dimensional model can be a good fit for both dataset. One thing is that in the second dataset which is without the fungi points, the absolute difference of 7-dimension is lower than 8-dimensional one. Then that proves the new dataset with the fungi is also not euclidean.

Overall, for the original dataset, we can get lots of information from the 2-dimensional model. It is actually simpler than the one without the fungi points from looking at the GOFs. 8-dimensional model could be good as well, but it is hard to explain the axis, because the distance matrix is not euclidean. If I have to choose only one dimension to analyze this dataset, then I will choose the 2-dimensional model. However, this dataset is interesting, and if I can find a more interesting one, I would do it for the last assignment.