Unsupervised learning

Katarzyna Kedzierska

25 October 2019

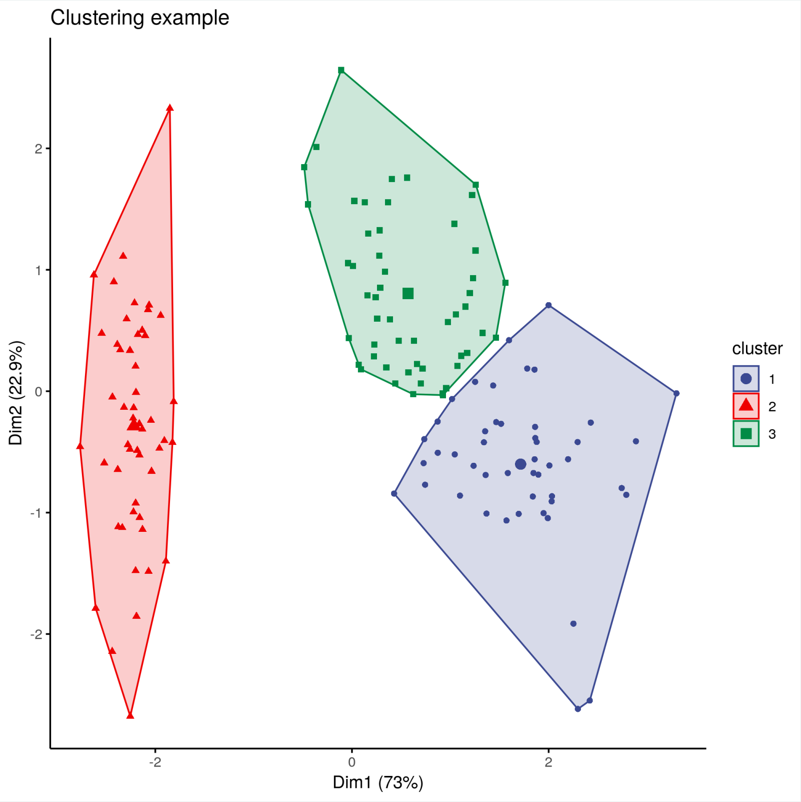
Outline

What will be talking about today?

- Unsupervised learning
- Dimensionality reduction techniques
    - PCA
    - PCoA
    - nMDS
    - tSNE
    - UMAP
- Clustering
    - K-means
    - Hierarchical

Unsupervised learning

The goal: find hidden patterns in unlabeled data.



What does dimensionality reduction mean?

We can reduce dimensionality of our data by for example explaining two measuremnets by one.

Let's say we have the dimenions of a recatngle:

```
rect_dims <- tibble(rect = 1:5,
                    a = sample(1:10, 5, replace = TRUE),
                    b = sample(5:10, 5, replace = TRUE))
rect_dims
```

```
## # A tibble: 5 x 3
##    rect     a     b
##   <int> <int> <int>
## 1     1     5     8
## 2     2     8    10
## 3     3     6     6
## 4     4     2     7
## 5     5     5     8
```

We can reduce dimenions by expressing the `a` and `b` variables by choosing some representation. For example, we can calculate the area:

```
rect_dims %>%
  mutate(area = a * b) %>%
  select(rect, area)
```

```
## # A tibble: 5 x 2
##    rect  area
##   <int> <int>
## 1     1    40
## 2     2    80
## 3     3    36
## 4     4    14
## 5     5    40
```

Why bother you may ask.

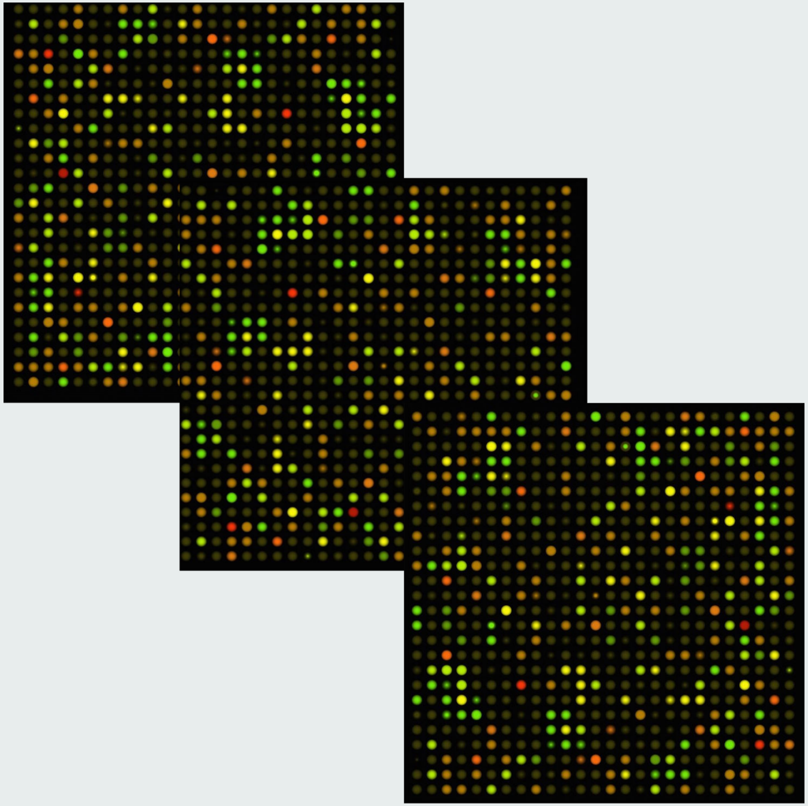
```
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```
cor.test(~ Sepal.Length + Petal.Length, data = iris)
```

```
## 
##     Pearson's product-moment correlation
## 
## data:  Sepal.Length and Petal.Length
## t = 21.646, df = 148, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8270363 0.9055080
## sample estimates:
##       cor 
## 0.8717538
```

Why do dimensionality reduction?



Dimensionality Reduction methods

Note: those are the ones we will touch on in this brief tutorial, there are many more methods! *Note2:* the following classification of the methos is arbitrary and serves only teaching purposes.

- Linear methods

- Principal Component Aanalysis (PCA)
- Classical / Metric Multidimensioanl Scaling == Principal Coordinate Analysis (PCoA)

- Non-linear methos

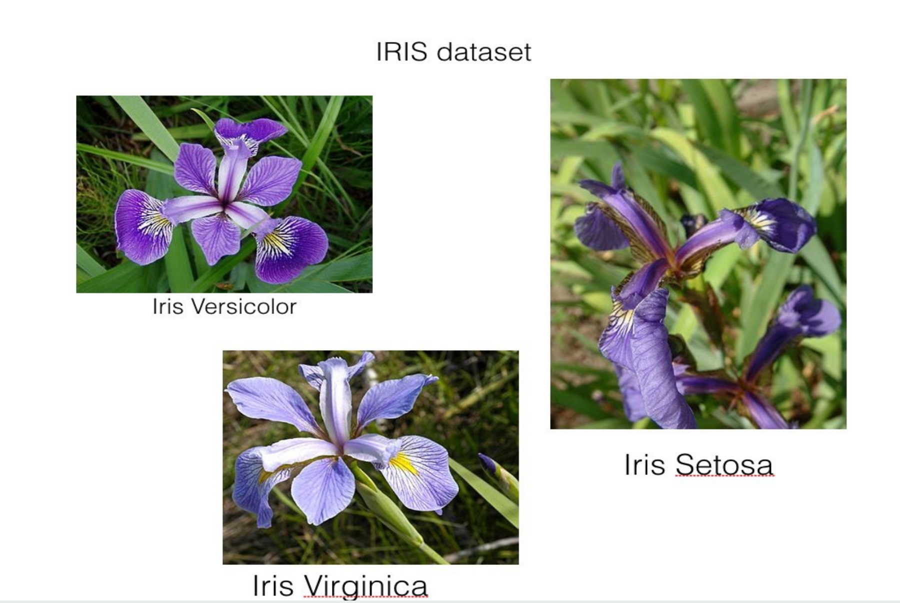
- Non-Metric Multidimensional Scaling (NMDS)
- t-distributed Stochastic Neighbor Embedding (tSNE)
- Uniform Manifold Approximation and Projection (UMAP)

In-depth introdction to DR methods

Iris data set

```
head(iris, n=3)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
```



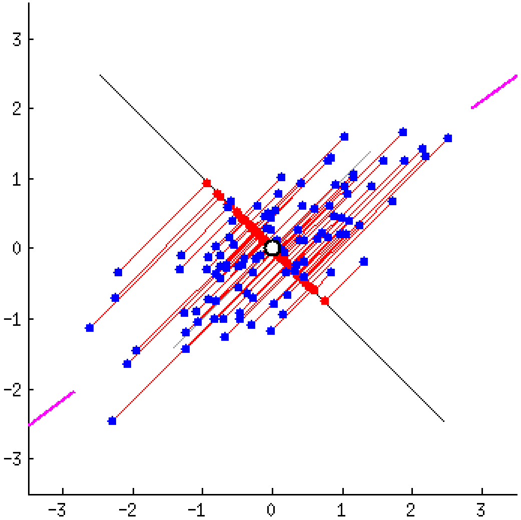
Iris data set

```
summary(iris)
```

```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width   
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100  
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300  
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300  
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199  
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800  
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500  
##        Species  
##  setosa    :50  
##  versicolor:50  
##  virginica :50  
##                 
##                 
##                 
```

Principal Component Analysis (PCA)

The goal of the PCA



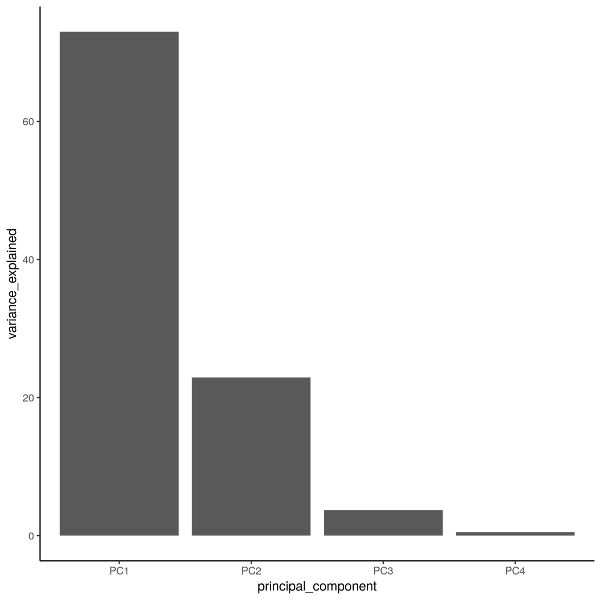
PCA on Setosa, Making sense of PCA on CrossValidated

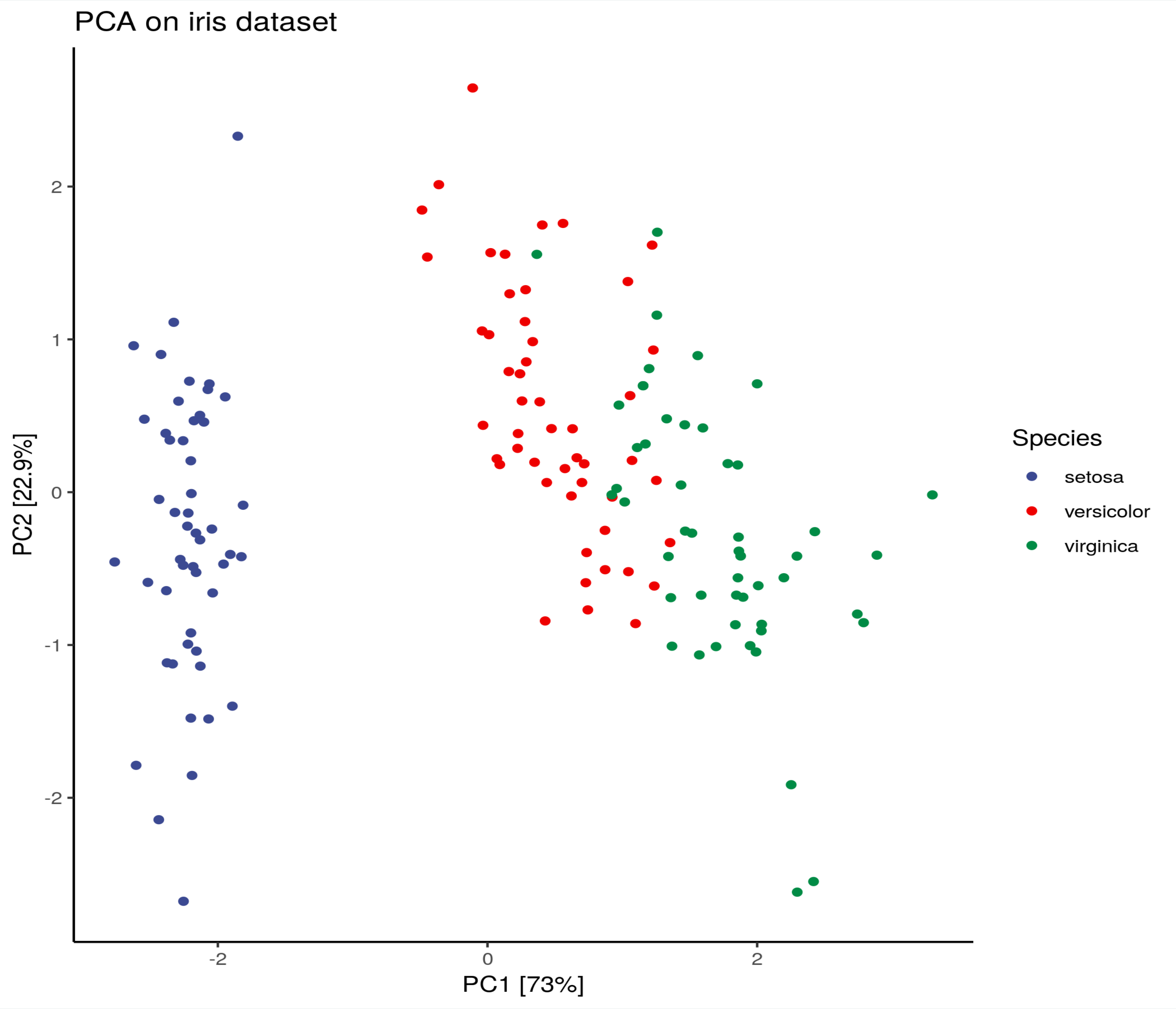
PCA in R

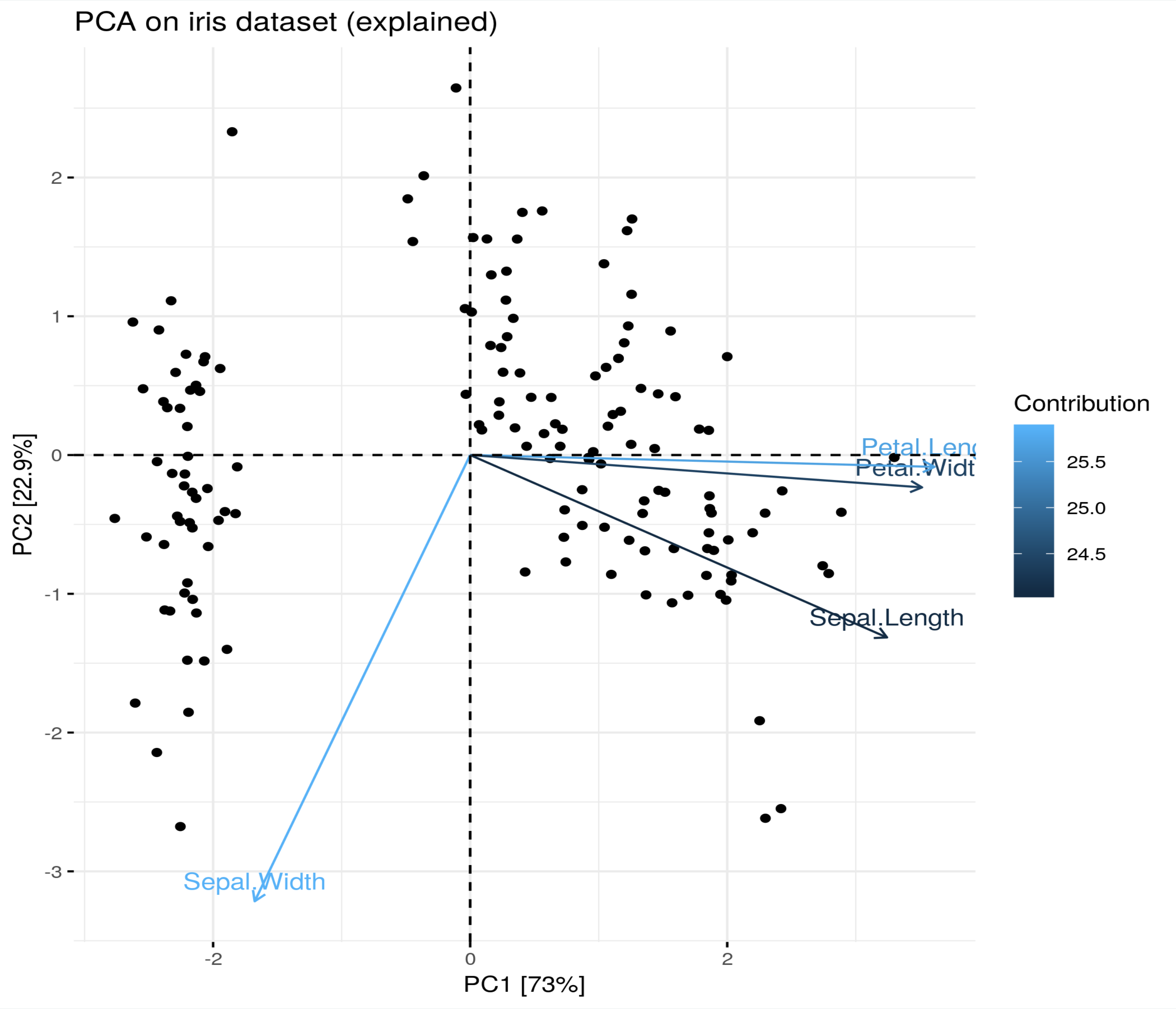
```
iris_pca <- prcomp(iris[,1:4], scale = TRUE, center = TRUE)
```

```
summary(iris_pca)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4
## Standard deviation     1.7084 0.9560 0.38309 0.14393
## Proportion of Variance 0.7296 0.2285 0.03669 0.00518
## Cumulative Proportion  0.7296 0.9581 0.99482 1.00000
```




PCA on iris dataset
PC2 [22.9%]
2
1
0
-1
-2
-2
0
2
PC1 [73%]
Species
setosa
versicolor
virginica

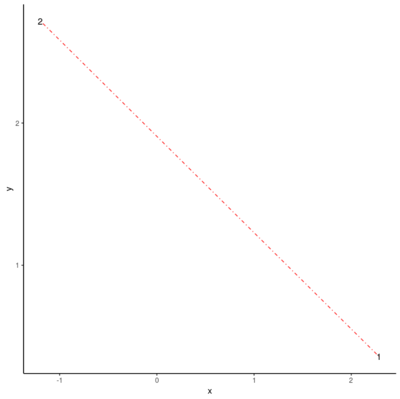

PCA on iris dataset (explained)
PC2 [22.9%]
2
1
0
-1
-2
-3
-2
0
2
PC1 [73%]
Petal.Leng
Petal.Widt
Sepal.Length
Sepal.Width
Contribution
25.5
25.0
24.5

Distances

Exactly what you think it is. :)

```
set.seed(7)
samp_df <- tibble(x = rnorm(10), y = rnorm(10))

samp_df[1:2,] %>%
  dist()
```

```
##          1
## 2 4.207954
```



```
sqrt((x1 - x2)^2 + (y1 - y2)^2)
```

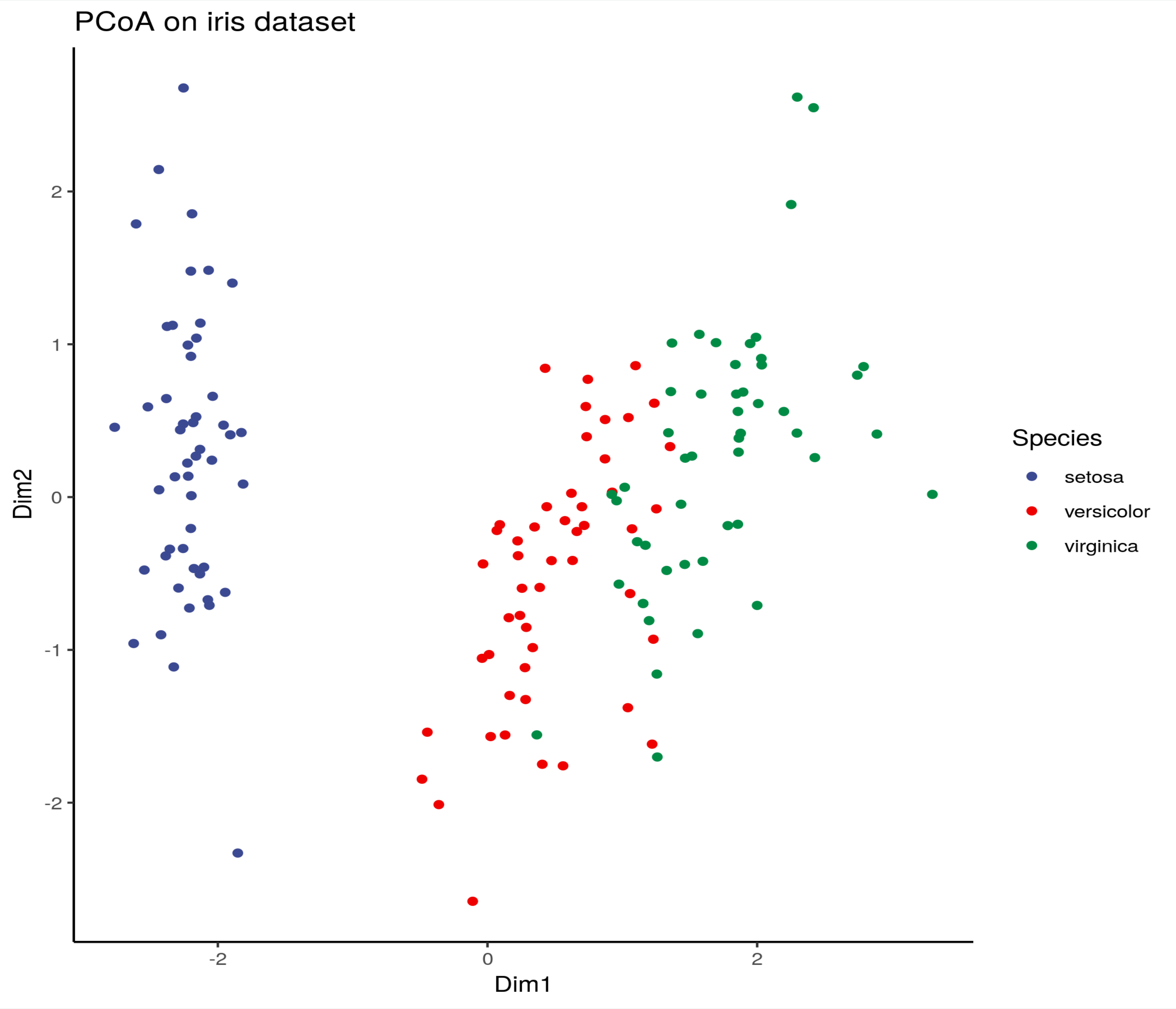
```
##          x
## 1 4.207954
```

Classical / Metric Multidimensioanl Scaling == Principal Coordinate Analysis (PCoA)

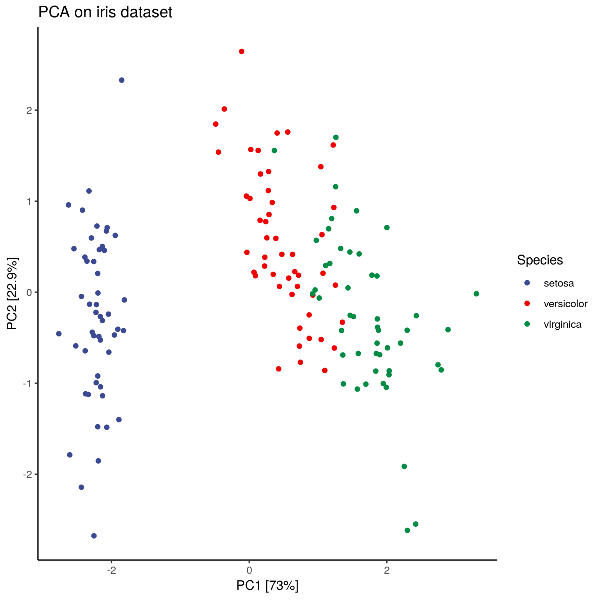
```
iris_dist <- iris[,1:4] %>%
  scale() %>%
  dist()
iris_pcoa <- cmdscale(iris_dist)
```

```
head(iris_pcoa, n = 5)
```

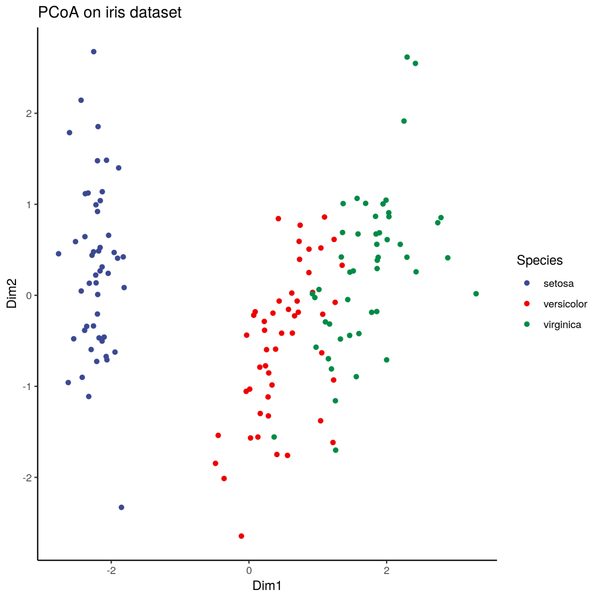
```
##           [,1]       [,2]
## [1,] -2.257141  0.4784238
## [2,] -2.074013 -0.6718827
## [3,] -2.356335 -0.3407664
## [4,] -2.291707 -0.5953999
## [5,] -2.381863  0.6446757
```


PCoA on iris dataset
Dim2
2
1
0
-1
-2
Dim1
-2
0
2
Species
● setosa
● versicolor
● virginica

```
iris_pca_plt
```



```
iris_pcoa_plt
```



However, this method gives us more flexibility. For example, we can use prior knowledge about our dataset and reduce the dimenions using distance appropriate for our method. Like,

- correlation distance for correlated measurements (gene expression?)
- binary distance for multiple binary meaurements (smoking/non-smoking and similiar)
- ranking based distanced to apply rank based normalisation (mutations in genes)

Non linear methods

Non-Metric Multidimensional Scaling (NMDS)

- No assumptions about the linear relationship!
- NMDS tries to find a Euclidean distance matrix in 2D that will best correlate with the distances in the original space.

Here: Kruskal’s non-metric multidimensional scaling

```
iris_nmds <- MASS::isoMDS(iris_dist2)
```

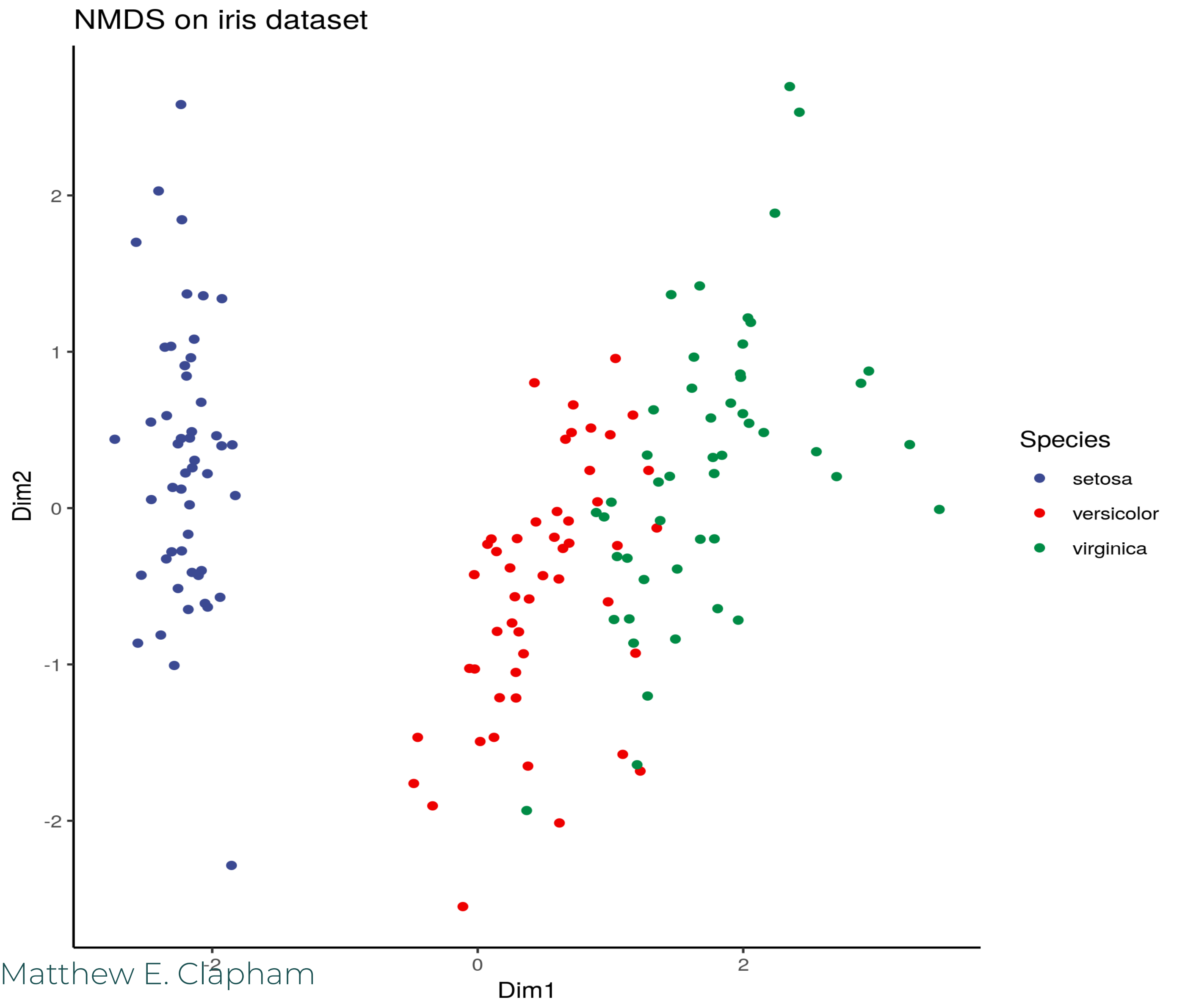
```
## initial  value 4.818162
## final  value 4.817789
## converged
```

```
head(iris_nmds$points, n=3)
```

```
##          [,1]       [,2]
## 1 -2.234280  0.4449669
## 2 -2.054236 -0.6096799
## 3 -2.305528 -0.2795758
```

```
iris_nmds$stress
```

```
## [1] 4.817789
```


NMDS on iris dataset
Dim2
2
1
0
-1
-2
Dim1
-2
0
2
Species
● setosa
● versicolor
● virginica

tSNE

1. Step 1. Determining the similarity of the points. (normal dist)
2. Step 2. Random projection of the data. (t-distribution)
3. Step 3. Moving the points - so Matrix from Step 2 resembles Matrix form Step 1.

Simple, ain't it? :)

"How to Use t-SNE Effectively"

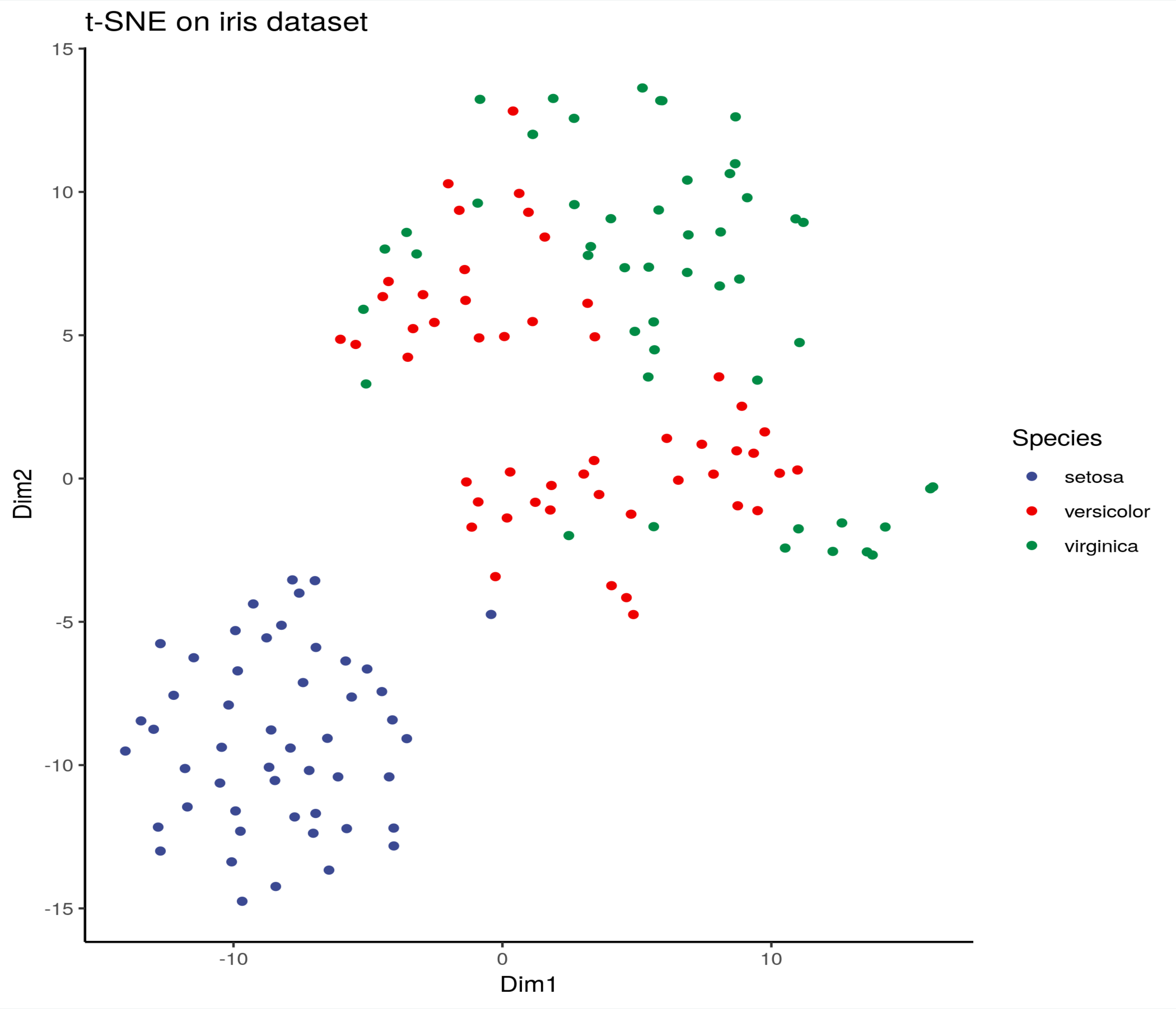
t-SNE

```
iris_tsne <- tsne::tsne(iris[,1:4])
```

```
## sigma summary: Min. : 0.486505661043274 |1st Qu. : 0.587913800179832 |Median : 0.61487243

## Epoch: Iteration #100 error is: 12.652124554884

## Epoch: Iteration #200 error is: 0.205306339588185

## Epoch: Iteration #300 error is: 0.20432551195561

## Epoch: Iteration #400 error is: 0.204287989607803

## Epoch: Iteration #500 error is: 0.204287652103515

## Epoch: Iteration #600 error is: 0.204287647840186

## Epoch: Iteration #700 error is: 0.204287647776791

## Epoch: Iteration #800 error is: 0.204287647776038

## Epoch: Iteration #900 error is: 0.204287647776029

## Epoch: Iteration #1000 error is: 0.204287647776029
```

```
head(iris_tsne)
```

```
##             [,1]       [,2]
## [1,]  -8.678520 -10.075163
## [2,]  -4.091358  -8.422715
```


t-SNE on iris dataset
15
10
5
0
-5
-10
-15
Dim2
-10
0
10
Dim1
Species
setosa
versicolor
virginica

UMAP

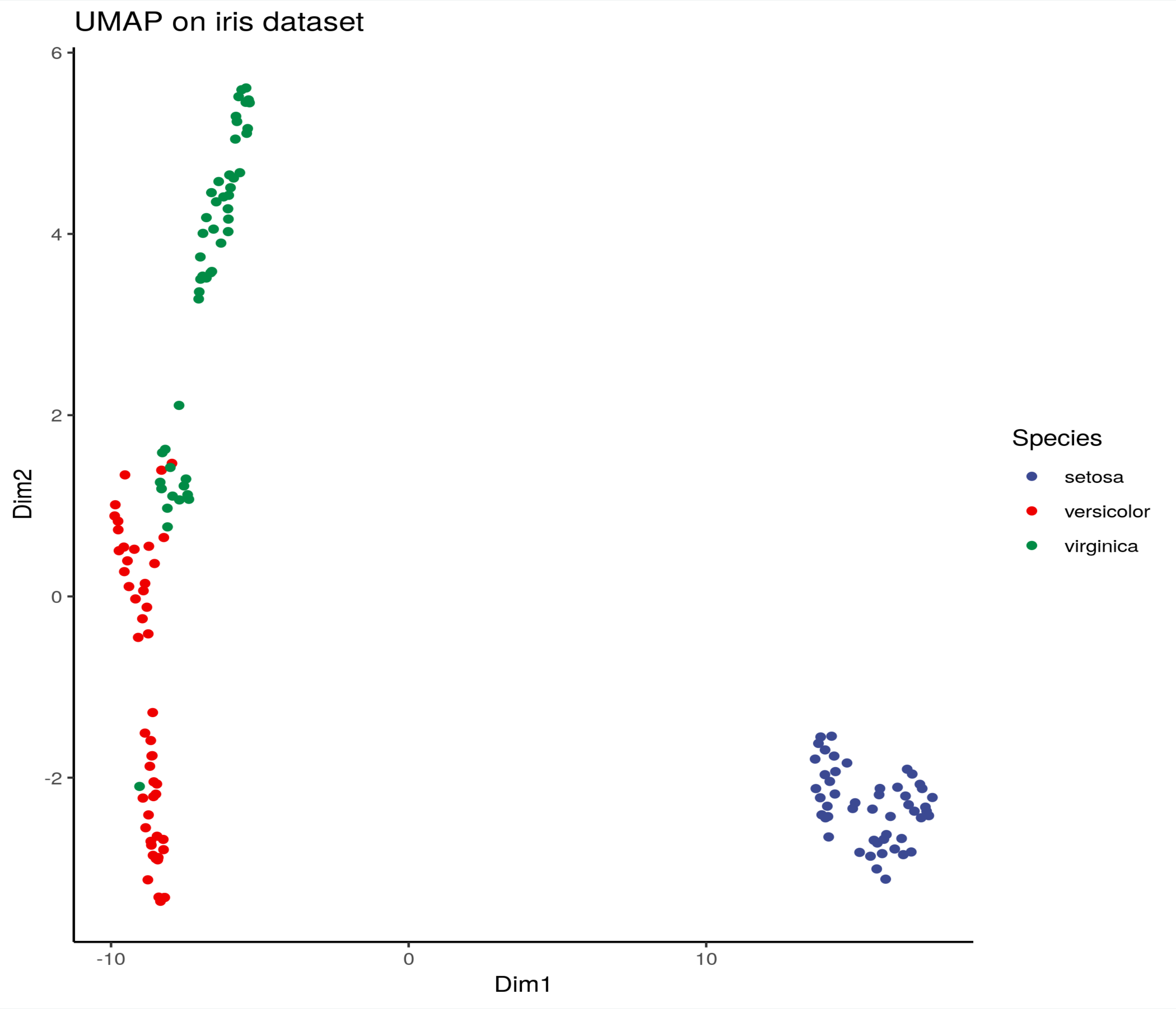
```
iris_umap <- umap::umap(iris[,1:4])
```

```
head(iris_umap$layout)
```

```
##          [,1]      [,2]
## [1,] 15.91344 -2.837810
## [2,] 13.88194 -2.409741
## [3,] 14.15320 -2.041289
## [4,] 13.98945 -1.966901
## [5,] 16.05571 -2.627575
## [6,] 17.18179 -2.073115
```

```
str(iris_umap)
```

```
## List of 4
##  $ layout: num [1:150, 1:2] 15.9 13.9 14.2 14 16.1 ...
##  $ data  : num [1:150, 1:4] 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:4] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"
##  $ knn   :List of 2
##   ..$ indexes  : int [1:150, 1:15] 1 2 3 4 5 6 7 8 9 10 ...
##   ..$ distances: num [1:150, 1:15] 0 0 0 0 0 0 0 0 0 0 ...
##   ..- attr(*, "class")= chr "umap.knn"
##  $ config:List of 23
##   ..$ n_neighbors         : int 15
##   ..$ n_components        : int 2
##   ..$ metric              : chr "euclidean"
##   ..$ n_epochs            : int 200
##   ..$ input               : chr "data"
##   ..$ init                : chr "spectral"
##   ..$ min_dist            : num 0.1
##   ..$ set_op_mix_ratio    : num 1
##   ..$ local_connectivity  : num 1
##   ..$ bandwidth           : num 1
##   ..$ alpha               : num 1
##   ..$ gamma               : num 1
##   ..$ negative_sample_rate: int 5
##   ..$ a                   : num 1.58
##   ..$ b                   : num 0.895
##   ..$ spread              : num 1
##   ..$ random_state        : int 147849767
##   ..$ transform_state     : int NA
##   ..$ knn_repeats         : num 1
##   ..$ verbose             : logi FALSE
##   ..$ umap_learn_args     : logi NA
```


UMAP on iris dataset
Dim2
6
4
2
0
-2
-10
0
10
Dim1
Species
● setosa
● versicolor
● virginica

```r
ggpubr::ggarrange(iris_pca_plt, iris_pcoa_plt,
                  iris_nmds_plt, iris_tsne_plt, iris_umap_plt, common.legend = TRU
```

Clustering

Concept - assing data points into groups.

Kmeans

Biggest disadvantage: you need to know k.

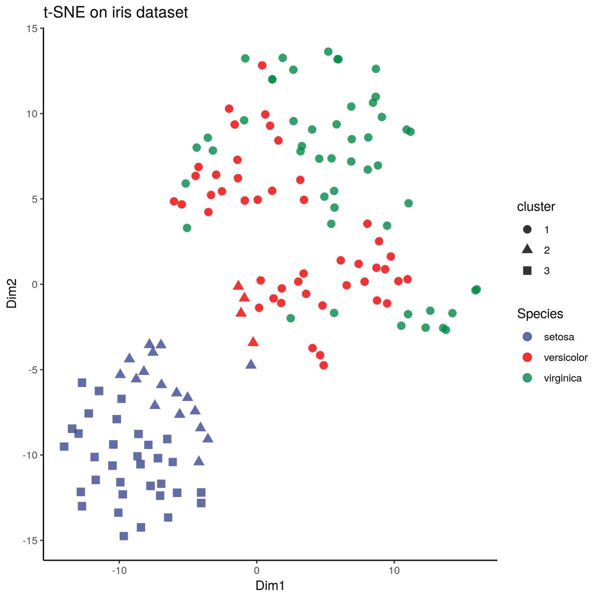
```
iris_kmeans <- kmeans(scale(iris[,1:4]), 3)
iris_kmeans$cluster
```

```
##   [1] 3 2 2 2 3 3 3 3 2 2 3 3 2 2 3 3 3 3 3 3 3 3 3 3 3 2 3 3 3 2 2 3 3 3 2
##  [36] 2 3 3 2 3 3 2 2 3 3 2 3 2 3 3 1 1 1 1 1 1 1 2 1 1 2 1 1 1 1 1 1 1 1 1
##  [71] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 2 1 1 1 1 1 1
## [106] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [141] 1 1 1 1 1 1 1 1 1 1
```

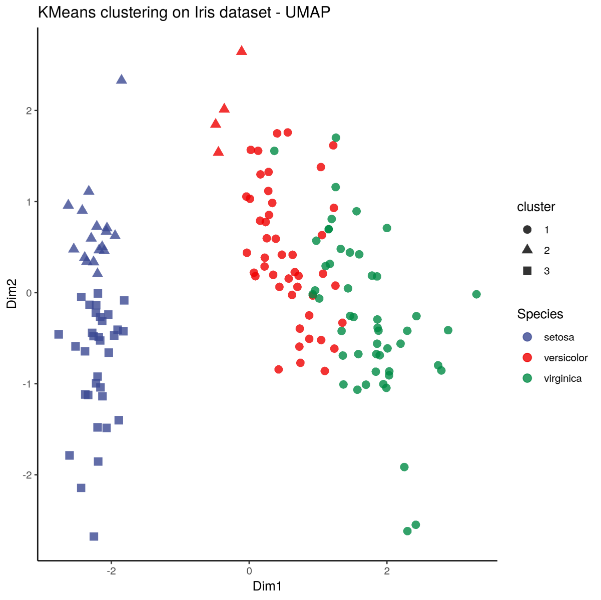
Kmeans

```
iris_umap_plt2 <-
  iris_umap$layout %>%
  as.data.frame() %>%
  mutate(label = iris$Species,
         cluster = as.character(iris_kmeans$cluster)) %>%
  ggplot(., aes(x = V1, y = V2,
                color = label, shape = cluster)) +
  geom_point(size = 3, alpha = 0.8) +
  labs(title = "UMAP on iris dataset",
       color = "Species",
       x = "Dim1",
       y = "Dim2") +
  scale_color_aaas()
iris_umap_plt2
```

```r
iris_tsne_plt2 <-
  iris_tsne %>%
  as.data.frame() %>%
  mutate(label = iris$Species,
         cluster = as.character(iris_kmeans$cluster)) %>%
  ggplot(., aes(x = V1, y = V2,
                color = label, shape = cluster)) +
  geom_point(size = 3, alpha = 0.8) +
  labs(title = "t-SNE on iris dataset",
       color = "Species",
       x = "Dim1",
       y = "Dim2") +
  scale_color_aaas()
iris_tsne_plt2
```

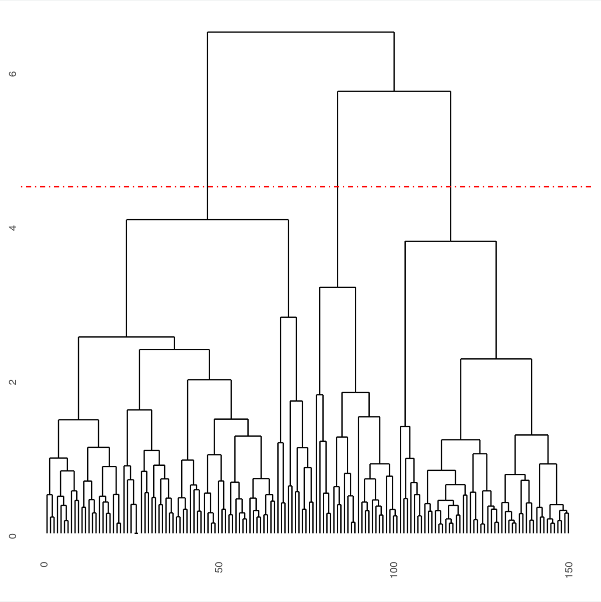


```r
iris_pca_plt2 <-
  iris_pca$x %>%
  as.data.frame() %>%
  mutate(label = iris$Species,
         cluster = as.character(iris_kmeans$cluster)) %>%
  ggplot(., aes(x = PC1, y = PC2,
                color = label, shape = cluster)) +
  geom_point(size = 3, alpha = 0.8) +
  labs(title = "KMeans clustering on Iris dataset - UMAP",
       color = "Species",
       x = "Dim1",
       y = "Dim2") +
  scale_color_aaas()
iris_pca_plt2
```



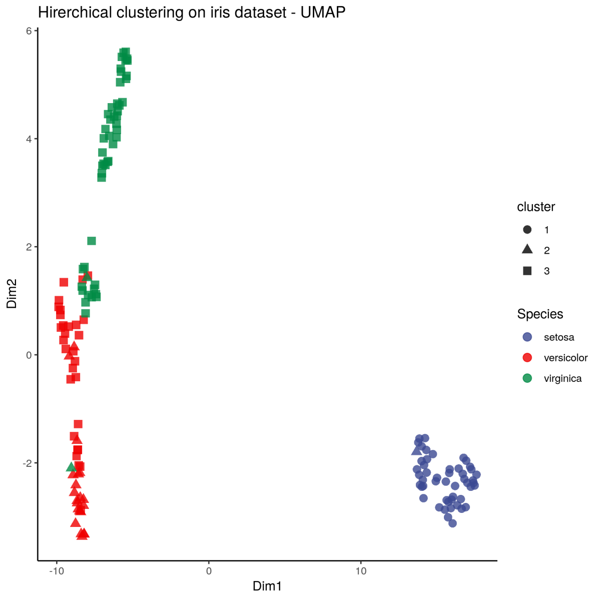
```
iris_hclust <- hclust(iris_dist)
```

```
ggdendro::ggdendrogram(iris_hclust, rotate = FALSE,
                       size = 2, labels = FALSE) +
  geom_hline(yintercept = 4.5, color = "red", linetype = "dotdash")
```



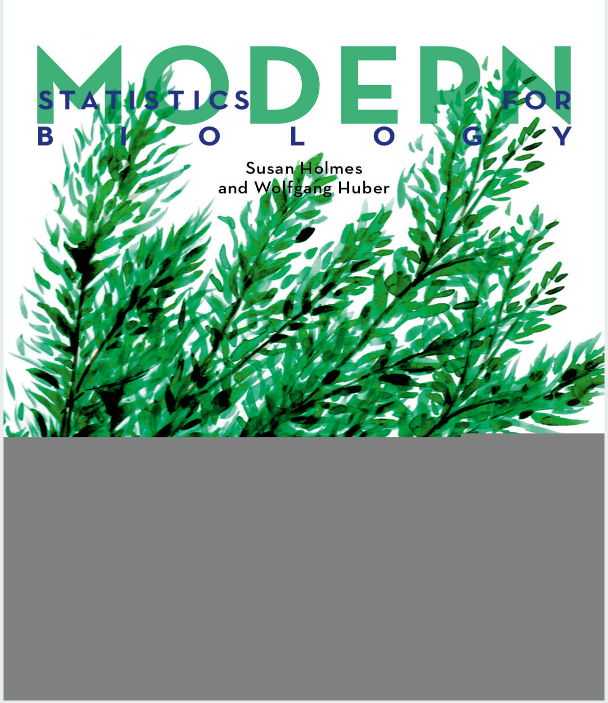
```r
iris_hclust_labels <- dendextend::cutree(iris_hclust, 3)
```

```r
iris_umap_plt3 <-
  iris_umap$layout %>%
  as.data.frame() %>%
  mutate(label = iris$Species,
         cluster = as.character(iris_hclust_labels)) %>%
  ggplot(., aes(x = V1, y = V2,
                color = label, shape = cluster)) +
  geom_point(size = 3, alpha = 0.8) +
  labs(title = "Hirerchical clustering on iris dataset - UMAP",
       color = "Species",
       x = "Dim1",
       y = "Dim2") +
  scale_color_aaas()
iris_umap_plt3
```



Resources

- Modern Statistics for Modern Biology



- In-depth introdction to DR methods
- Making sense of PCA on CrossValidated
- UMAP @ SciPy2018