PCA.vs.tSNE

Wen-Ching Calvin Chan

7/3/2018

主成分(分量)分析 Principal Component Analysis (PCA)

From the detection of outliers to predictive modeling, PCA has the ability of projecting the observations described by pvariables into few orthogonal components defined at where the data 'stretch' the most, rendering a simplified overview.

簡單來說，主成分分析是一種透過線性變換來簡化數據集的技術。 因為每個變數都在不同程度上反映了所研究問題的某些信息，並且指標之間彼此有一定的相關性，因而所得的統計數據反映的信息在一定程度上有重疊。 而變數太多會增加計算量和增加分析問題的複雜性。

選用具有代表性的綜合指標

降維(Dimension reduction)即當資料維度數 (變數) 很多的時候，有沒有辦法讓維度數 (變數) 少一點，但資料特性不會差太多。

> 主成份分析是希望資料投影後資料的**變異量會最大化**，獨立成份 (分量) 分析 (Independent components analysis, ICA) 則是希望資料投影後，投影的資料軸跟軸之間彼此是統計獨立。

數學式 Mathematical Equations

假設有n的樣本點$\{x1, x2, \ldots, xn\}$ $xi \in R^d$ 投影軸為 v

在線性代數中，其實就是利用奇異值分解 (singular value decomposition, SVD) 求得 C (共變異數矩陣) 的特徵值 (eigenvalue, λ) 和特徵向量 (eigenvector, v)。

> 解出來的eigenvalue就是變異量(variance)，eigenvector就是讓資料投影下去會有最大變異量的投影軸。

PCA is particularly powerful in dealing with multicollinearity and variables that outnumber the samples (p ≫ n).

累積貢獻比率 (Cumulative Proportion) 是什麼?

統計上，累積貢獻比率 (Cumulative Proportion) 將決定需要取多少主要成分出來。

The cumulative proportion for a value x in a distribution is the proportion of observations in the distribution that lie at or below x.

將主成份從最重要往次要排序後的變異量百分筆累積

統計上，因素分析 (Factor Analysis) 則進一步分析每個主成份所蘊含的因素

缺點

> - 因為 PCA 是對資料求共變異數矩陣，之後進行奇異值分解。因此會被資料的差異性影響，無法很好表現相似性以及分佈。
> - 且 PCA 是一種線性 (linear) 降維的方式，但如果特徵與特徵間的關聯是非線性關係的話，用 PCA 可能會導致欠擬合（underfitting）的情形發生。

鳶尾花(iris)

這個R內建的鳶尾花(iris)資料集是非常著名的生物資訊資料集之一，取自美國加州大學歐文分校的機械學習資料庫 http://archive.ics.uci.edu/ml/datasets/Iris，資料的筆數為150筆，共有五個欄位：

1. 花萼長度(Sepal Length)：計算單位是公分。
2. 花萼寬度(Sepal Width)：計算單位是公分。

3. 花瓣長度(Petal Length) ：計算單位是公分。
4. 花瓣寬度(Petal Width)：計算單位是公分。
5. 類別(Class)：可分為Setosa，Versicolor和Virginica三個品種。

```
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
##
##
##
```

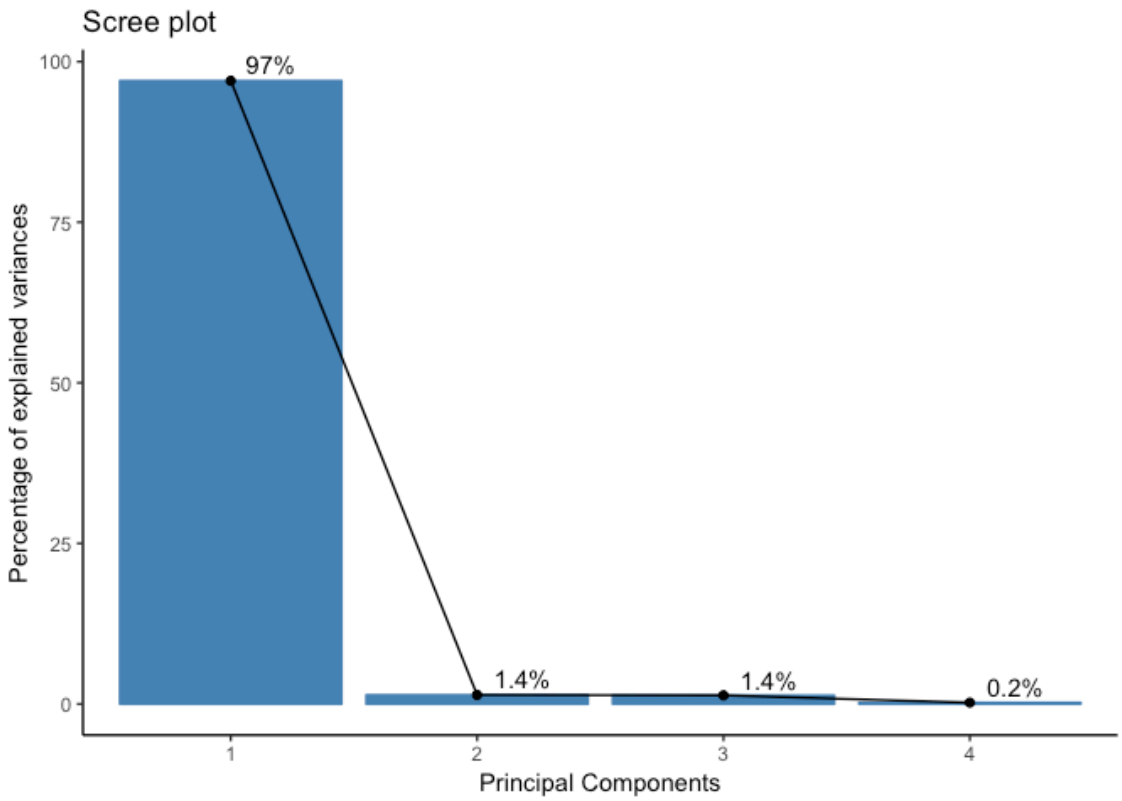
Iris

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa

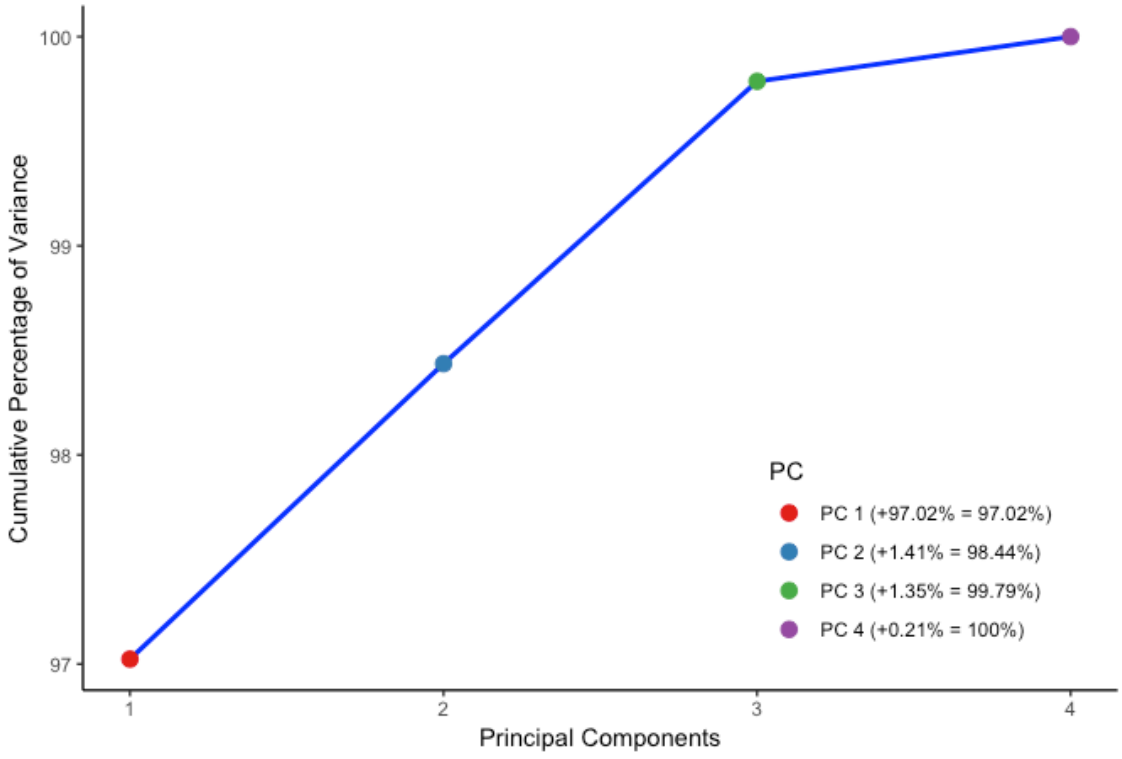
```
summary(
  lst.pca <- FactoMineR::PCA(
    log(iris[, 1:4]),
    scale.unit = FALSE,
    ncp = 5,
    graph = FALSE
  )
) # ~ princomp(cor = FALSE) or prcomp()
```

```
##
## Call:
## FactoMineR::PCA(X = log(iris[, 1:4]), scale.unit = FALSE, ncp = 5,
##      graph = FALSE)
##
##
## Eigenvalues
##                        Dim.1   Dim.2   Dim.3   Dim.4
## Variance               1.306   0.019   0.018   0.003
## % of var.             97.023   1.413   1.350   0.214
## Cumulative % of var.  97.023  98.436  99.786 100.000
##
## Individuals (the 10 first)
##                  Dist    Dim.1    ctr   cos2    Dim.2    ctr   cos2
## 1             |  1.675 | -1.674  1.430  0.998 |  0.021  0.015  0.000 |
## 2             |  1.672 | -1.669  1.422  0.996 | -0.068  0.162  0.002 |
## 3             |  1.716 | -1.714  1.500  0.998 |  0.020  0.014  0.000 |
## 4             |  1.646 | -1.642  1.377  0.995 | -0.096  0.326  0.003 |
## 5             |  1.679 | -1.677  1.436  0.998 |  0.037  0.047  0.000 |
## 6             |  1.019 | -0.983  0.494  0.932 |  0.258  2.326  0.064 |
## 7             |  1.354 | -1.336  0.911  0.973 |  0.184  1.191  0.019 |
## 8             |  1.641 | -1.639  1.372  0.998 | -0.043  0.066  0.001 |
## 9             |  1.687 | -1.678  1.437  0.989 | -0.087  0.268  0.003 |
## 10            |  2.271 | -2.229  2.536  0.963 | -0.405  5.747  0.032 |
##                Dim.3    ctr   cos2
## 1              0.060  0.134  0.001 |
## 2             -0.069  0.176  0.002 |
## 3             -0.075  0.207  0.002 |
## 4             -0.047  0.082  0.001 |
## 5              0.071  0.184  0.002 |
## 6              0.067  0.165  0.004 |
## 7             -0.117  0.502  0.007 |
## 8              0.059  0.130  0.001 |
## 9             -0.146  0.783  0.007 |
## 10             0.165  1.001  0.005 |
##
## Variables
##                   Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3    ctr
## Sepal.Length |  0.115  1.018  0.671 |  0.000  0.000  0.000 |  0.066 23.928
## Sepal.Width  | -0.066  0.332  0.213 |  0.079 33.006  0.309 |  0.096 50.988
## Petal.Length |  0.577 25.530  0.964 | -0.095 47.210  0.026 |  0.058 18.226
## Petal.Width  |  0.977 73.120  0.995 |  0.061 19.784  0.004 | -0.035  6.858
##                cos2
## Sepal.Length  0.220 |
## Sepal.Width   0.456 |
## Petal.Length  0.010 |
## Petal.Width   0.001 |
```

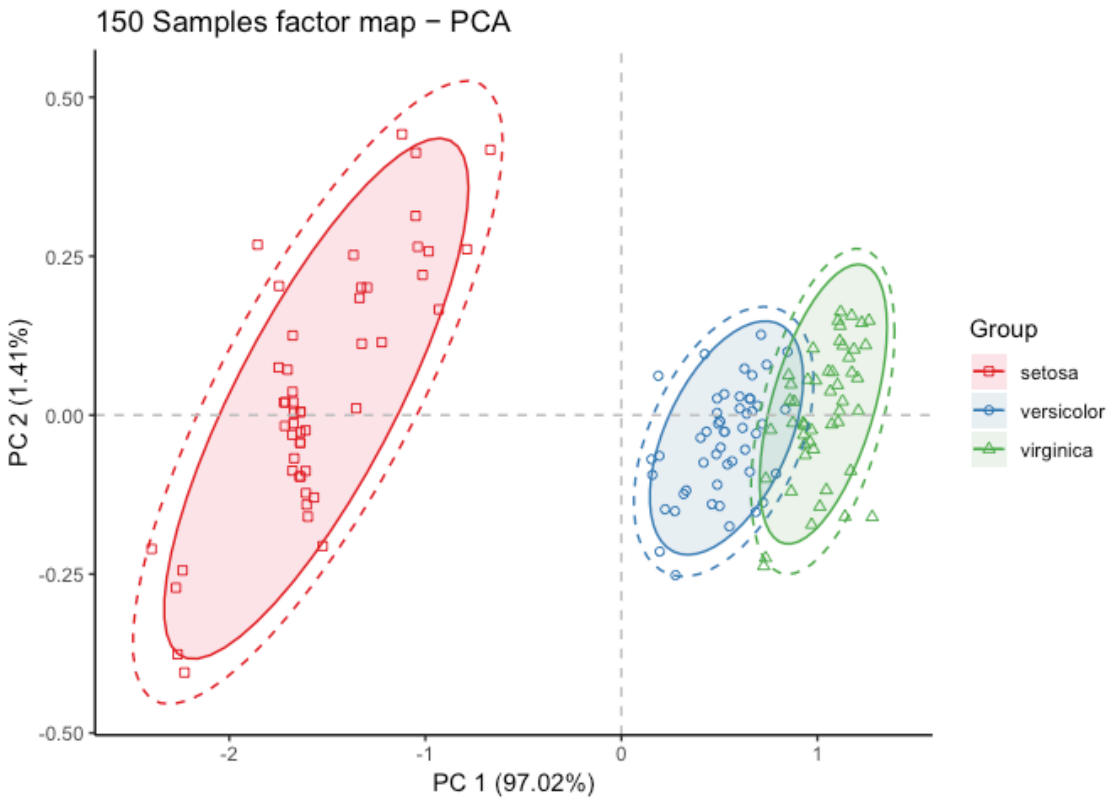
陡坡圖 (Scree Plot)



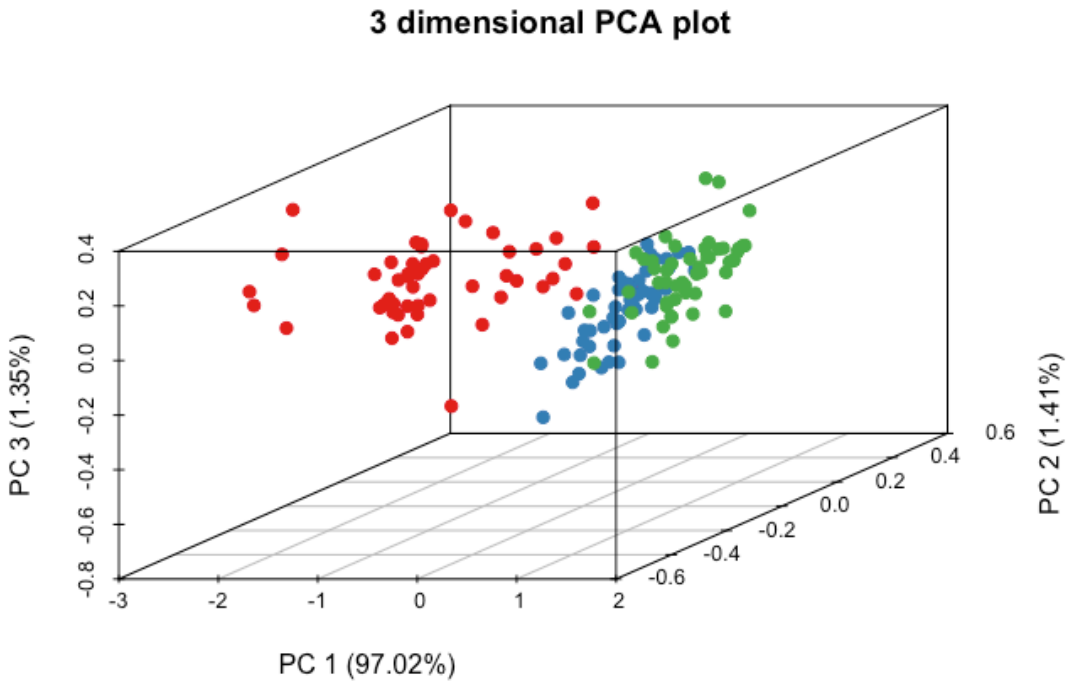
累積貢獻比率 (Cumulative Proportion; Pareto Plot)



PCA 2D Plot



PCA 3D Plot



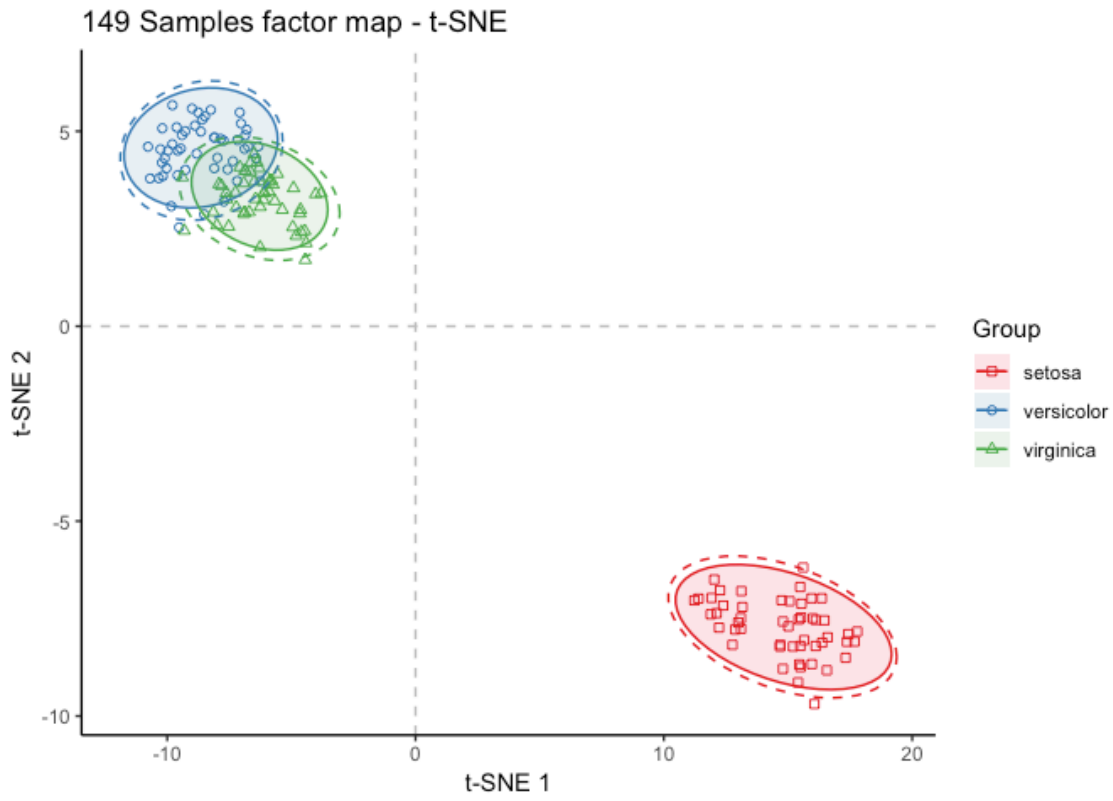
t-分布非線性降维 tSNE (t-Distributed Stochastic Neighbour Embedding)

t-SNE 主要是將高維的數據用高斯分佈的機率密度函數近似，而低維數據的部分使用 t 分佈的方式來近似，在使用 KL 距離 (Kullback-Leibler Divergence) or 相對熵（Relative Entropy） 計算相似度 (cost function)，最後再以梯度下降（或隨機梯度下降）求最佳解 。

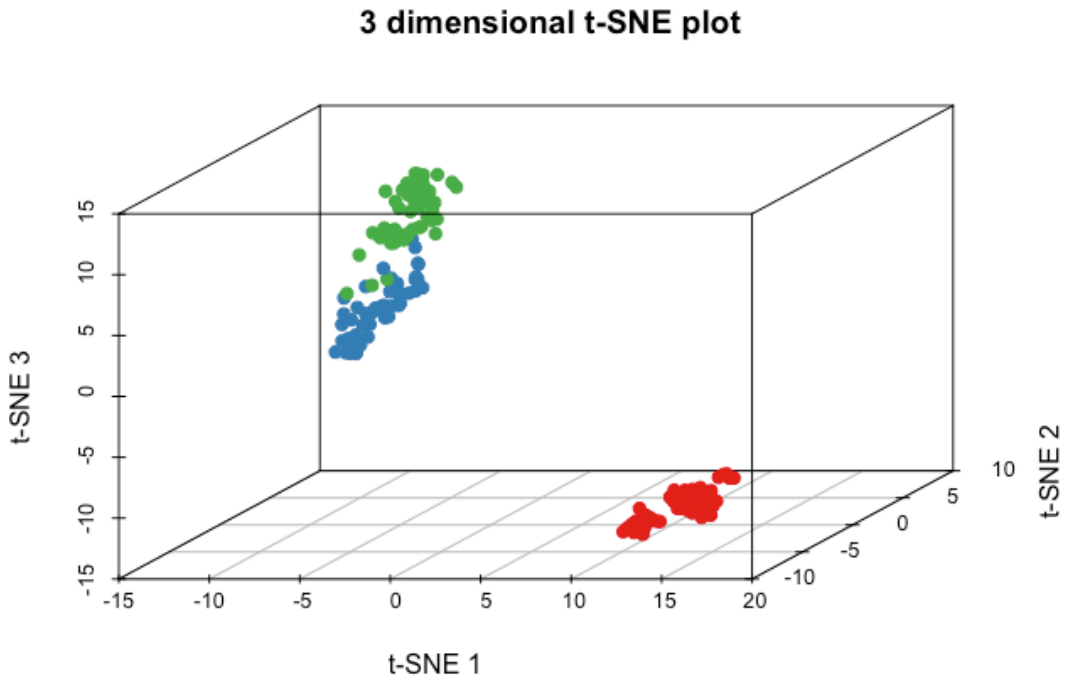
```
iris_unique <- unique(iris) # Remove duplicates before running TSNE.
set.seed(42) # Sets seed for reproducibility
lst.tsne <- Rtsne::Rtsne(
  log(as.matrix(iris_unique[,1:4])),
  dims = min(3, i.npcs)
) # Run TSNE
```

```
## $N
## [1] 149
##
## $Y
##          [,1]      [,2]       [,3]
## [1,] 16.11313 -8.203840  -9.596087
## [2,] 14.67853 -8.220295  -9.396950
## [3,] 15.50229 -8.744928  -9.100665
## [4,] 15.01283 -7.700390  -8.738312
## [5,] 16.37266 -8.117400  -9.521690
## [6,] 12.26175 -6.774772 -12.223199
##
## $costs
## [1] -0.0006986974 -0.0002552980 -0.0003674619 -0.0004297655 -0.0001049897
## [6] -0.0001767622
##
## $itercosts
## [1] 44.2399421 41.6604784 43.6271920 42.8904956 43.6344668  0.1926183
##
## $origD
## [1] 4
##
## $perplexity
## [1] 30
##
## $theta
## [1] 0.5
##
## $max_iter
## [1] 1000
##
## $stop_lying_iter
## [1] 250
##
## $mom_switch_iter
## [1] 250
##
## $momentum
## [1] 0.5
##
## $final_momentum
## [1] 0.8
##
## $eta
## [1] 200
##
## $exaggeration_factor
## [1] 12
```

t-SNE Plot



t-SNE 3D Plot



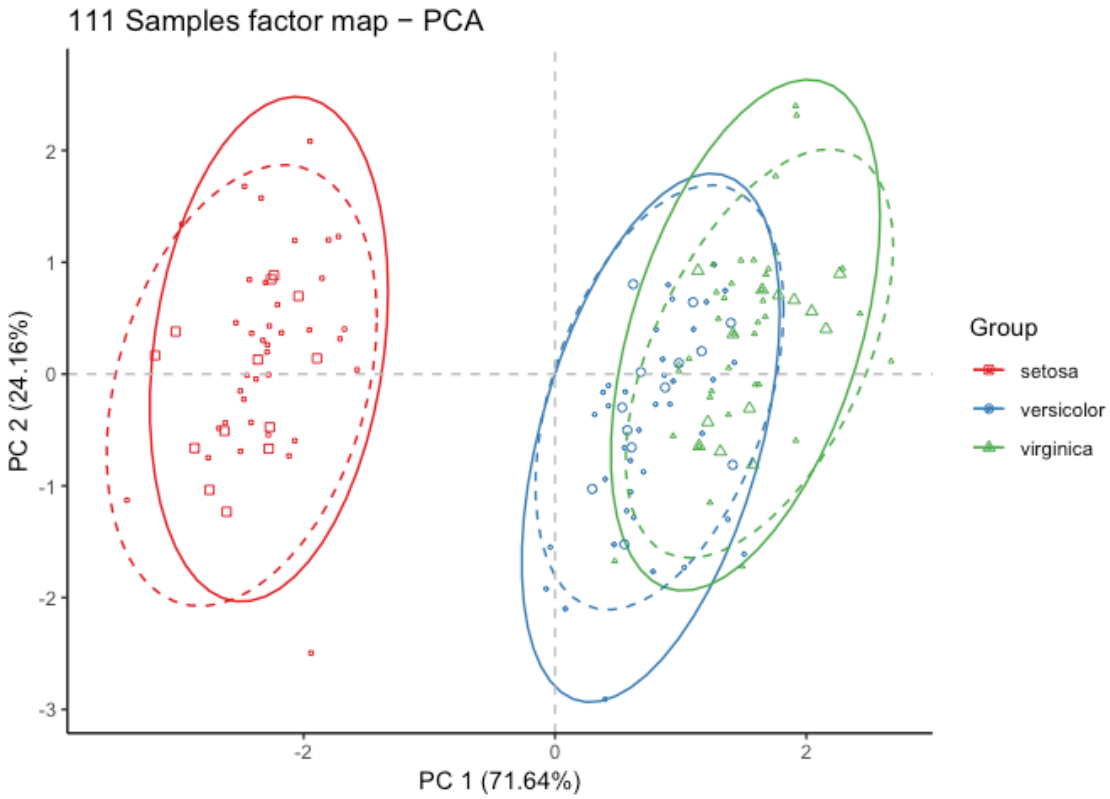
Prediction

```
lst.label <- split(x = iris, f = iris$Species) # stratified data based on classification labels
set.seed(42) # Sets seed for reproducibility
lst.idx.train <- sapply(lst.label, function(x) { sort(sample(nrow(x), nrow(x)*0.75)) }, simplify = F)
iris.train <- as.data.frame(do.call(rbind, lapply(names(lst.label), function(x) { lst.label[[x]][lst.idx.train
[[x]],] })))
iris.valid <- as.data.frame(do.call(rbind, lapply(names(lst.label), function(x) { lst.label[[x]][-lst.idx.trai
n[[x]],] })))

pca <- prcomp(log(iris.train[,1:4]), retx = TRUE, center = TRUE, scale = TRUE) # conduct PCA on training datas
et
expl.var <- pca$sdev^2/sum(pca$sdev^2) # percent explained variance

pred <- predict(pca, newdata = log(iris.valid[,1:4])) # prediction of PCs for validation dataset
```

PCA 2D Plot



優缺點 (Pros and Cons)

- 當特徵數量過多時，使用 PCA 可能會造成降維後的特徵欠擬合 (underfitting) ，這時可以考慮使用 t-SNE 來降維，因為 t-SNE 可高維度資料做可視化降維
- t-SNE 的需要比較多的時間執行
- PCA 可建立 predictive model, 但 t-SNE 目前沒有這樣的功能 (只能 train & test 一起重新跑)

- reference
  - Computing and visualizing PCA in R (https://tgmstat.wordpress.com/2013/11/28/computing-and-visualizing-pca-in-r/)
  - PCA
    - Principal Component Methods in R: Practical Guide (http://www.sthda.com/english/articles/31-principal-component-methods-in-r-practical-guide/118-principal-component-analysis-in-r-prcomp-vs-princomp/)
    - 機器/統計學習:主成分分析(Principle Component Analysis, PCA) (https://medium.com/@chih.sheng.huang821/%E6%A9%9F%E5%99%A8-

%E7%B5%B1%E8%A8%88%E5%AD%B8%E7%BF%92-%E4%B8%BB%E6%88%90%E5%88%86%E5%88%86%E6%9E%90-principle-component-analysis-pca-58229cd26e71)
    - 主成分分析法- MBA智库百科 (http://wiki.mbalib.com/zh-tw/%E4%B8%BB%E6%88%90%E5%88%86%E5%88%86%E6%9E%90%E6%B3%95)
    - R筆記–(7)主成份分析(2012美國職棒MLB) (http://rpubs.com/skydome20/R-Note7-PCA)
- t-SNE
    - tsne (https://lvdmaaten.github.io/tsne/)
    - Rtsne (https://github.com/jkrijthe/Rtsne)
    - Excellent Tutorial (https://github.com/oreillymedia/t-SNE-tutorial)
- Iris
    - Iris Data Set @ UCI (http://archive.ics.uci.edu/ml/datasets/Iris)
    - R統計分析與資料探勘入門—以鳶尾花資料集為例 - 計資中心 (http://www.cc.ntu.edu.tw/chinese/epaper/0031/20141220_3105.html)
- PCA vs. t-SNE
    - 淺談降維方法中的 PCA 與 t-SNE (https://medium.com/d-d-mag/%E6%B7%BA%E8%AB%87%E5%85%A9%E7%A8%AE%E9%99%8D%E7%B6%AD%E6%96%B9%E6%B3%95-pca-%E8%88%87-t-sne-d4254916925b)
- Prediction
    - How to use R prcomp results for prediction? (https://stats.stackexchange.com/questions/72839/how-to-use-r-prcomp-results-for-prediction)
    - Once I have a t-SNE map, how can I embed incoming test points in that map? (https://lvdmaaten.github.io/tsne/)