PLSC 504

Make-Up Class: Cluster Analysis and Item Response Theory

December 13, 2022

Cluster Analysis

Cluster Analysis

"...a statistical operation of grouping objects. The resulting groups are clusters. Clusters have the following properties:

- We find them during the operation and their number is also not always fixed in advance.
- They are the combination of objects having similar characteristics."

"...groups objects (observations, events) based on the information found in the data describing the objects or their relationships. The goal is that the objects in a group will be similar (or related) to one other and different from (or unrelated to) the objects in other groups. The greater the similarity (or homogeneity) within a group, and the greater the difference between groups, the 'better' or more distinct the clustering."

- Classification / Taxonomy (*description*)
- Data Reduction (*measurement*)
- Identify Relationships (*inductive inference*)
- Prediction (typically out-of-sample)

Clustering: Intuition

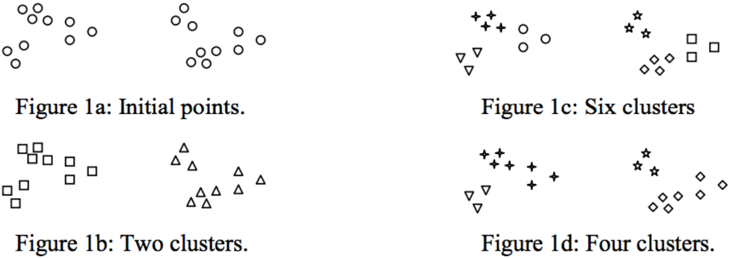


Figure 1a: Initial points.

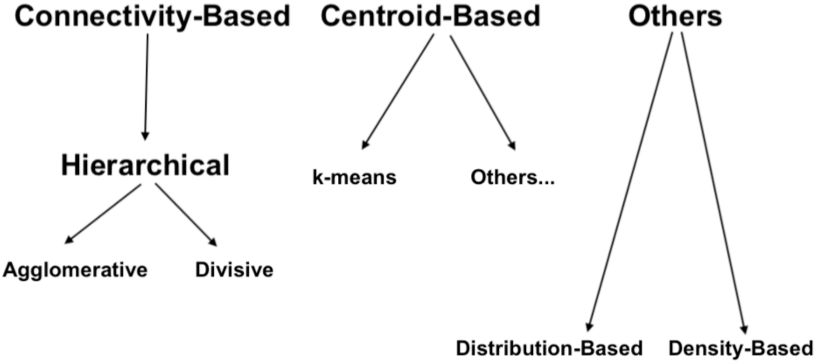
Figure 1b: Two clusters.

Figure 1c: Six clusters

Figure 1d: Four clusters.

(source)

Cluster Analysis: Typology



Euclidean ("L2") Distance:

$$d_{L2}(\mathbf{X},\mathbf{Y}) = \sqrt{\sum_{k=1}^{K}(X_i - Y_i)^2}.$$

"City-Block" / Manhattan ("L1") Distance:

$$d_{L1}(\mathbf{X},\mathbf{Y}) \equiv \|\mathbf{X}-\mathbf{Y}\|_1 = \sum_{k=1}^{K}|X_i - Y_i|.$$

Mahalanobis Distance:

$$d_M(\mathbf{X},\mathbf{Y}) = \sqrt{(\mathbf{X}-\mathbf{Y})'\mathbf{S}^{-1}(\mathbf{X}-\mathbf{Y})}.$$

Distance Example

Data ($N = 2$):

	X	Y	Z
Tick	1	711	0.08
Arthur	0	588	0.27
Tick - Arthur	1	123	-0.19

Euclidean:

$$
\begin{aligned}
D_{L2} &= \sqrt{(1-0)^2 + (711-588)^2 + (0.08-0.27)^2} \\
&= \sqrt{1 + 15129 + 0.0361} \\
&= 123.004
\end{aligned}
$$

Manhattan:

$$
\begin{aligned}
D_{L1} &= |1-0| + |711-588| + |0.08-0.27| \\
&= 1 + 123 + 0.19 \\
&= 124.19
\end{aligned}
$$

Mahalanobis:

$$
\begin{aligned}
D_M &= \sqrt{(\text{Tick} - \text{Arthur})' \hat{\mathbf{S}}^{-1} (\text{Tick} - \text{Arthur})} \\
&= 1.386
\end{aligned}
$$

Lesson: Standardize variables!

For two clusters C_A and C_B, the distance between can be defined in terms of:

- Single-linkage

$$d_{AB} = \min(d_{a,b})$$

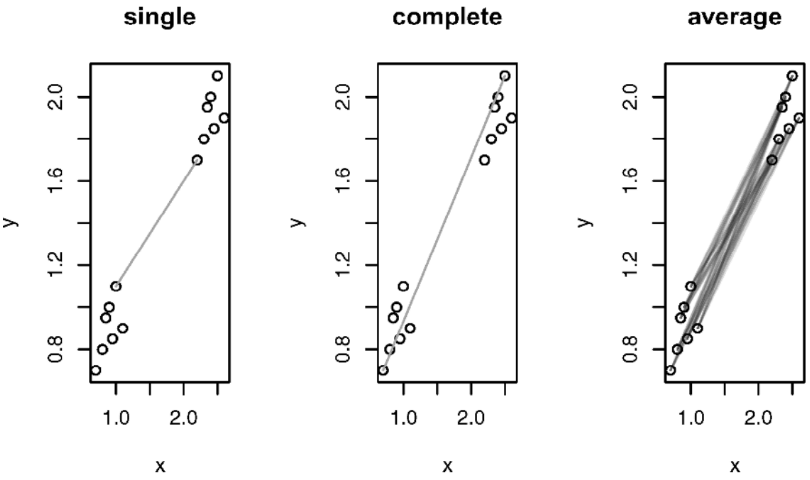
- Complete linkage

$$d_{AB} = \max(d_{a,b})$$

- Group average

$$d_{AB} = \frac{1}{N_A N_B} \sum_{a=1}^{N_A} \sum_{b=1}^{N_b} (d_{a,b})$$

Cluster Linkages

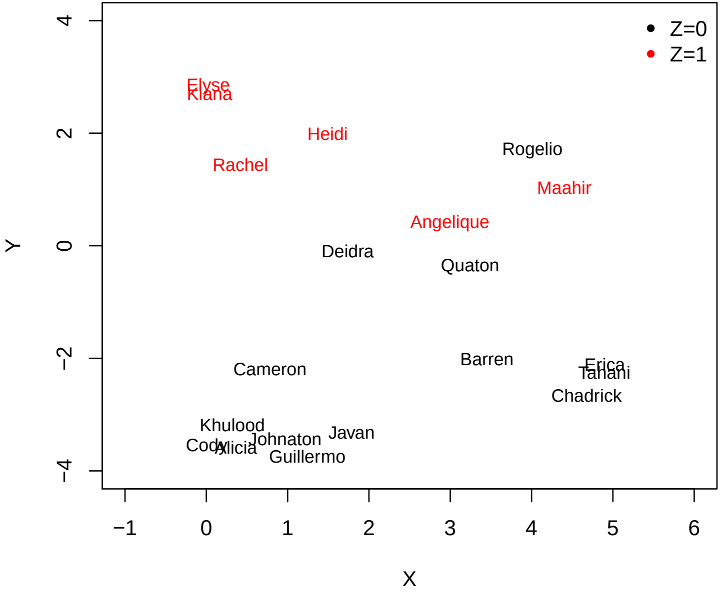


Basic steps:

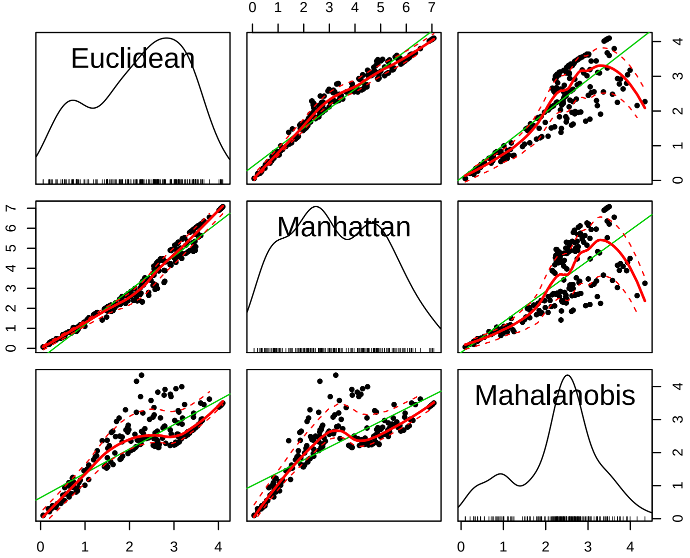
1. Begin with N observations on K variables in $\mathbf{X}$
2. Define each observation as its own "cluster" C_i
3. Find the two clusters C_ℓ and C_m that are "closest" to each other
4. Merge them into a single cluster, and delete the two component clusters
5. Recalculate the distances between all remaining clusters
6. Repeat steps 3-5 until only one cluster remains

```
> N <- 20
> set.seed(7222009)
> Name <- randomNames(N, which.names="first")
> X <- 5*rbeta(N,0.5,0.5)
> Y <- runif(N,-4,4)
> Z <- rbinom(N,1,pnorm(Y/2))

> df <- data.frame(Name=Name,X=X,Y=Y,Z=Z)
> rownames(df)<-df$Name
>
> # Distances:
> #
> # CENTER AND RESCALE / STANDARDIZE THE DATA:
>
> ds <- scale(df[,2:4])
>
> DL2 <- dist(ds) # L2 / Euclidean distance
> DL1 <- dist(ds,method="manhattan") # L1 / Manhattan distance
> DM <- sqrt(D2.dist(ds,cov(ds))) # Mahalanobis distances
```

Simulated Data, Plotted



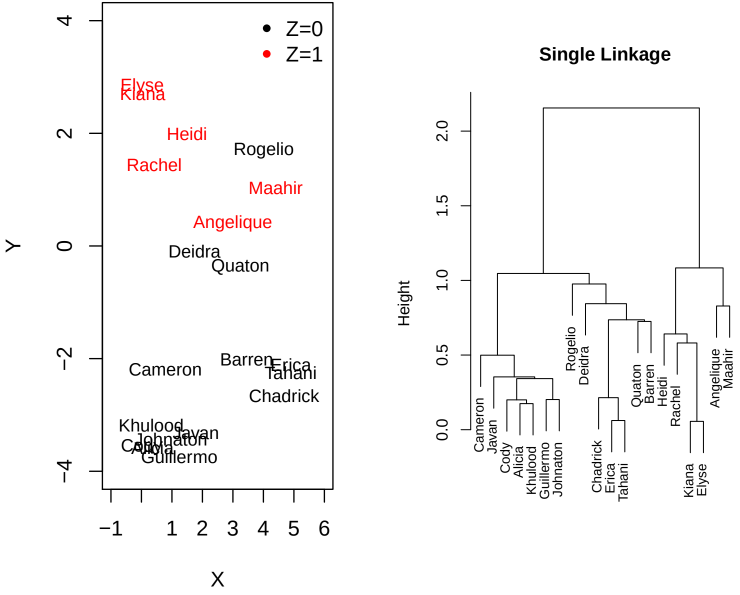
Distance Comparisons



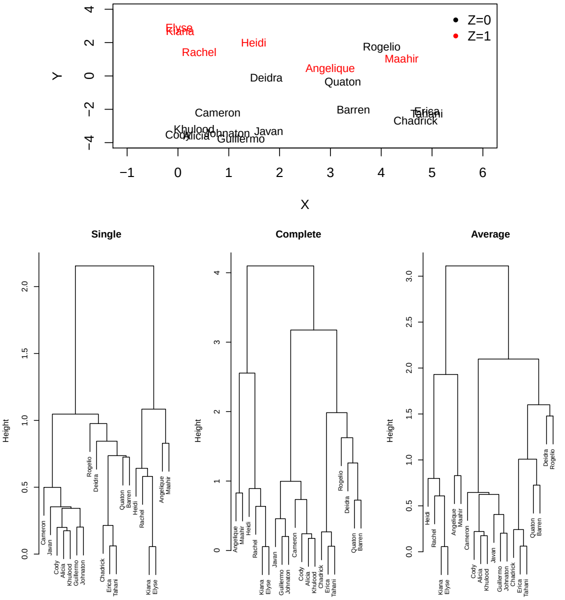
Using hclust (in cluster)

```
> ADL2.s <- hclust(DL2,method="single")
> ADL2.c <- hclust(DL2,method="complete")
> ADL2.a <- hclust(DL2,method="average")

> str(ADL2.s)
List of 7
 $ merge      : int [1:19, 1:2] -17 -15 -5 -9 -1 -19 4 -8 -11 -2 ...
 $ height     : num [1:19] 0.129 0.143 0.36 0.405 0.413 ...
 $ order      : int [1:20] 17 18 2 12 11 8 9 5 10 1 ...
 $ labels     : chr [1:20] "Guillermo" "Rachel" "Deidra" "Quaton" ...
 $ method     : chr "single"
 $ call       : language hclust(d = DL2, method = "single")
 $ dist.method: chr "euclidean"
 - attr(*, "class")= chr "hclust"
```

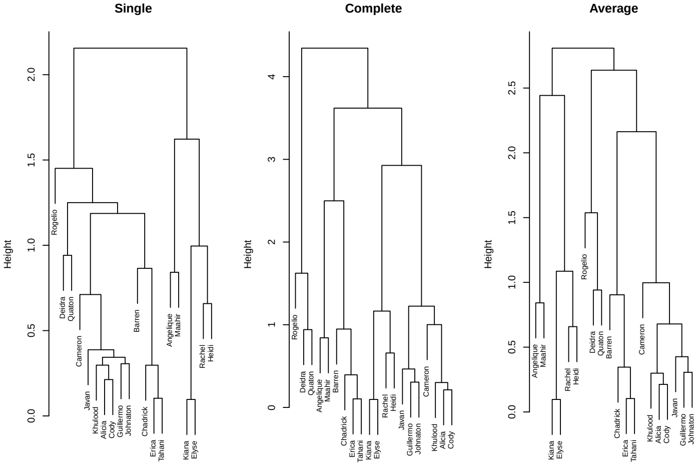
The Dendrogram



Comparing Linkages



Using Mahalanobis Distance



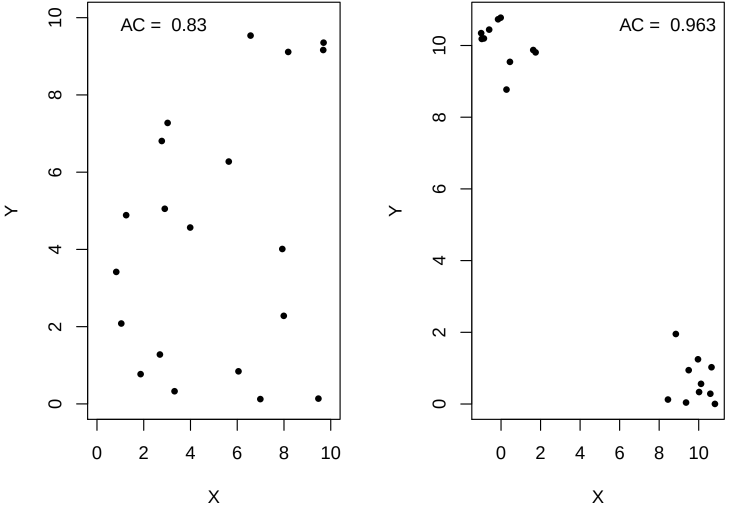
The *agglomeration coefficient* AC measures the clustering structure of the data. For each observation i, define m_i as the dissimilarity of observation i with the first cluster with which it is merged, divided by the dissimilarity in the final iteration (i.e., the greatest dissimilarity). The coefficient is then:

$$AC = \frac{1}{N-1} \sum_{i=1}^{N-1} 1 - m_i$$

Notes:

- Higher values correspond to greater clustering in the data.
- AC increases with N so should not be used to compare datasets of very different sizes

Example AC Values



Example *AC*s: Simulated Data

```
> Agnes.s <- agnes(ds, metric="euclidean",method="single")
> Agnes.s$ac
[1] 0.805

> Agnes.c <- agnes(ds, metric="euclidean",method="complete")
> Agnes.c$ac
[1] 0.8754

> Agnes.a <- agnes(ds, metric="euclidean",method="average")
> Agnes.a$ac
[1] 0.8398

> # Using Mahalanobis distance:
> Agnes.M <- agnes(DM, diss=TRUE, method="average")
> Agnes.M$ac
[1] 0.8071
```

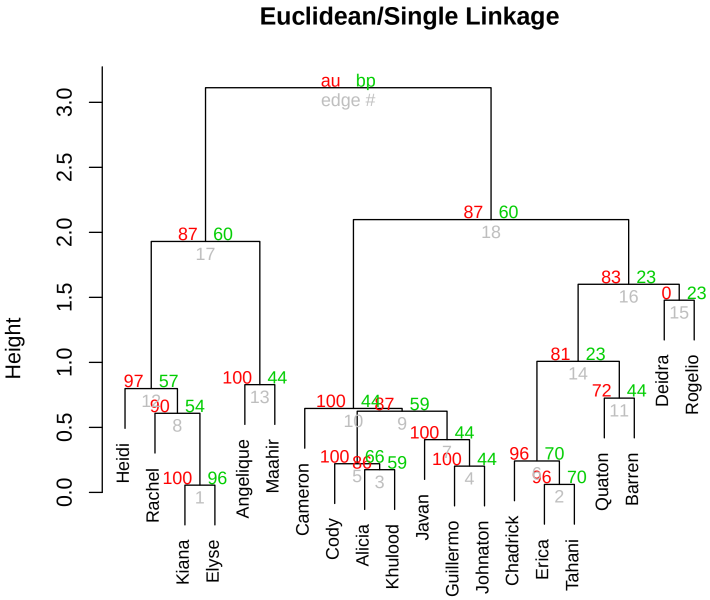
P-Values via Bootstrap

- Can calculate P-values for each cluster (at each agglomeration stage) via multiscale bootstrap resampling
- Reference: Suzuki, R., and H. Shimodaira. 2006. "pvclust: An R package for assessing the uncertainty in hierarchical clustering." *Bioinformatics* 22:1540-1542.
- The R package is `pvclust`
- Reports "approximately unbiased" and "bootstrap probability" P-values (use the former)
- "Clusters with high values... are strongly supported by the data."

P-Values...

```
dst<-data.frame(t(ds))
PVDL2.s <- pvclust(dst,method.hclust="single",
                   method.dist="euclidean",nboot=1001)
                   > PVDL2.s

Cluster method: average
Distance      : euclidean

Estimates on edges:

      au    bp se.au se.bp      v      c  pchi
1  0.997 0.957 0.001 0.003 -2.222  0.501 0.607
2  0.963 0.695 0.005 0.006 -1.147  0.636 0.022
3  0.856 0.593 0.013 0.006 -0.648  0.413 0.000
4  0.999 0.445 0.000 0.006 -1.482  1.621 0.105
5  0.997 0.656 0.001 0.006 -1.599  1.198 0.002
6  0.963 0.695 0.005 0.006 -1.147  0.636 0.022
7  0.999 0.445 0.000 0.006 -1.482  1.621 0.105
8  0.902 0.543 0.020 0.006 -0.701  0.592 0.000
9  0.869 0.594 0.012 0.006 -0.681  0.442 0.000
10 0.999 0.445 0.000 0.006 -1.482  1.621 0.105
11 0.721 0.445 0.019 0.006 -0.223  0.362 0.000
12 0.970 0.569 0.008 0.006 -1.028  0.853 0.065
13 0.999 0.439 0.001 0.006 -1.434  1.589 0.095
14 0.807 0.233 0.091 0.007 -0.069  0.797 0.607
15 0.002 0.233 0.002 0.007  1.837 -1.109 0.607
16 0.834 0.233 0.082 0.007 -0.121  0.849 0.607
17 0.866 0.601 0.012 0.006 -0.682  0.427 0.000
18 0.866 0.601 0.012 0.006 -0.682  0.427 0.000
19 1.000 1.000 0.000 0.000  0.000  0.000 0.000
```

Dendrogram with P-Values...



Practical Agglomerative Clustering: Linkages

"The performances of traditional hierarchical clustering methods have been evaluated for a variety of simulated situations. **Single linkage clustering is simple to understand and compute, but has the tendency to build unphysical elongated chains of clusters joined by a single point**, *especially when unclustered noise is present. Figure 12.4 of Izenman (2008) illustrates how a single linkage dendrogram can differ considerably from the average linkage, complete linkage and divisive dendrograms, which can be quite similar to each other. Kaufman and Rosseeuw (1990, Section 5.2) report that "Virtually all authors agreed that single linkage was least successful in their [simulation] studies." Everitt et al. (2001, Section 4.2) report that "Single linkage, which has satisfactory mathematical properties and is also easy to program and apply to large data sets, tends to be less satisfactory than other methods because of 'chaining'." Ward's method is successful with clusters of similar populations, but tends to misclassify objects when the clusters are elongated or have very different diameters.* **Average linkage is generally found to be an effective technique in simulations, although its results depend on the cluster size.** *Average linkage also has better consistency properties than single or complete linkage as the sample size increases towards infinity (Hastie et al. 2009, Section 14.3)."*

– Eric D. Feigelson and G. Jogesh Babu. 2012. *Modern Statistical Methods for Astronomy: With R Applications*. New York: Cambridge University Press, p. 228.

Divisive Clustering (`diana`)

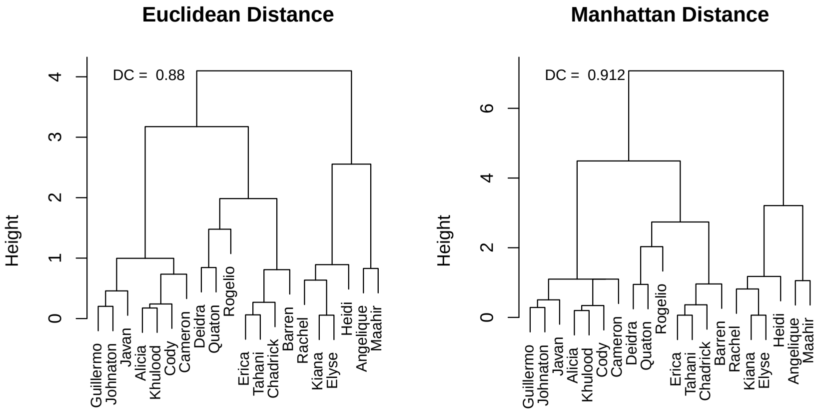
Basic steps:

1. Begin with N observations on K variables in $\mathbf{X}$
2. Select the cluster C_{maxD} with the largest *diameter* (defined as the cluster with the largest dissimilarity between any two of its observations)
3. Select the observation j in C_{maxD} that has the highest average dissimilarity to the other observations in the cluster); this is the "seed" of the "splinter group" $C_{splinter}$
4. Iteratively assign observations to either the splinter group $C_{splinter}$ or the parent cluster C_{parent}, based on their dissimilarity to each.
5. Repeat step 4 until each observation in C_{maxD} is reassigned to either C_{parent} or $C_{splinter}$
6. Iterate steps 2-5 until each observation is its own cluster

Divisive Clustering Example

```
> Diana.L2 <- diana(ds,metric="euclidean")
> Diana.L2
Merge:
      [,1] [,2]
 [1,]  -17  -18
 [2,]  -15  -20
 [3,]   -5   -9
 [4,]   -1   -7
 [5,]    3  -10
 [6,]    2  -19
 [7,]    4   -8
 [8,]   -2    1
 [9,]    5  -11
[10,]    6  -16
[11,]   -6  -13
[12,]   -3   -4
[13,]    8  -12
[14,]    7    9
[15,]   12  -14
[16,]   15   10
[17,]   13   11
[18,]   14   16
[19,]   18   17
Order of objects:
 [1] Guillermo Johnaton  Javan     Alicia    Khulood   Cody
 [7] Cameron   Deidra    Quaton    Rogelio   Erica     Tahani
[13] Chadrick  Barren    Rachel    Kiana     Elyse     Heidi
[19] Angelique Maahir
Height:
 [1] 0.20204 0.45777 0.99653 0.17474 0.24121 0.73438 3.17509 0.84410
 [9] 1.47820 1.98490 0.06146 0.26884 0.80881 4.09856 0.63594 0.05589
[17] 0.89190 2.55486 0.82867
Divisive coefficient:
[1] 0.8798
Available components:
[1] "order"     "height"    "dc"        "merge"     "diss"
[6] "call"      "order.lab" "data"
```

Divisive Clustering: Dendrograms



Non-Hierarchical Clustering: k-Means

k-means clustering "aims to partition the points into k groups such that the sum of squares from points to the assigned cluster centers is minimized."

- Formally, find:

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2 = \arg\min_{\mathbf{S}} \sum_{i=1}^{k} |S_i| \operatorname{Var} S_i$$

  for the set of k clusters $S_1 \ldots S_k$ in $\mathbf{S}$.

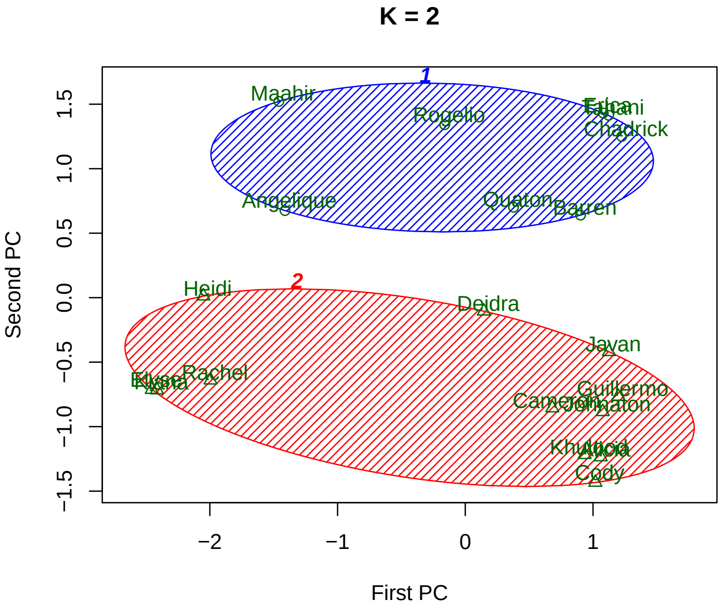
- Requires the analyst to designate the number of clusters desired k *a priori*.

- Standard algorithm:
  0. Initialize a set of k clusters.
  1. Assign each observation to the cluster whose mean is the least "distant" from it
  2. Calculate the new means as the centroids of the resulting clusters
  3. Repeat steps 1-2 until convergence.

k-means Clustering: Example ($k = 2$)

```
> KM2 <- kmeans(ds,2)
> KM2
K-means clustering with 2 clusters of sizes 7, 13

Cluster means:
        X       Y       Z
1 -0.7265 -0.9753 -0.6381
2  0.3912  0.5252  0.3436

Clustering vector:
Guillermo    Rachel    Deidra    Quaton    Alicia Angelique  Johnaton
        1         2         2         2         1         2         1
    Javan   Khulood      Cody   Cameron     Heidi    Maahir   Rogelio
        1         1         1         1         2         2         2
    Erica    Barren     Kiana     Elyse  Chadrick    Tahani
        2         2         2         2         2         2

Within cluster sum of squares by cluster:
[1]  0.9928 35.6954
 (between_SS / total_SS =  35.6 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"
[5] "tot.withinss" "betweenss"    "size"         "iter"
[9] "ifault"
```

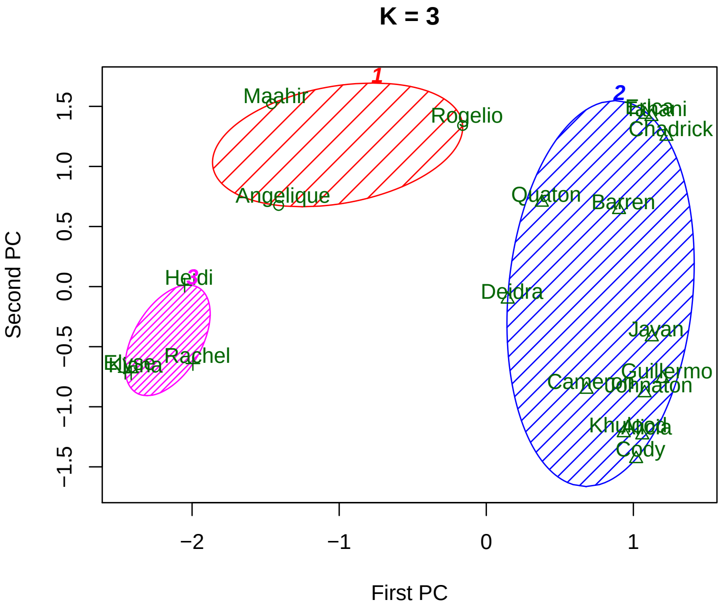
K-Means Clusters vs. Principal Components ($k = 2$)



k-means Clustering: Example ($k = 3$)

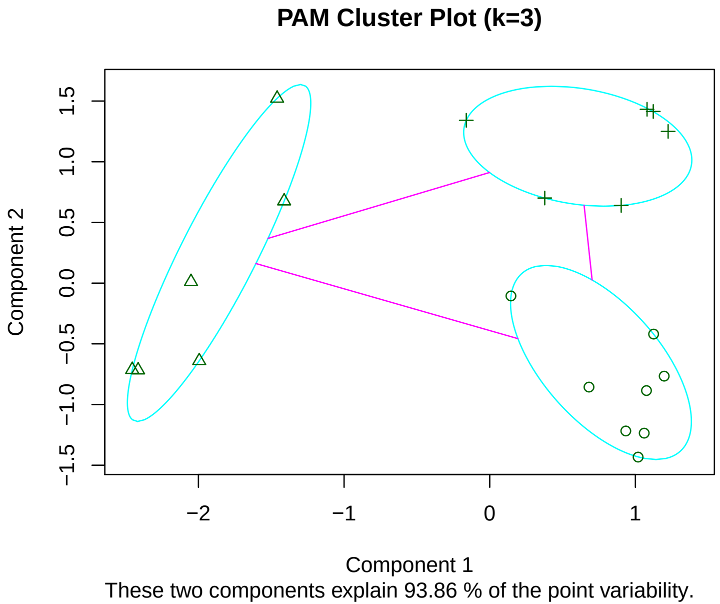
```
> KM3 <- kmeans(ds,3)
> KM3
K-means clustering with 3 clusters of sizes 7, 7, 6

Cluster means:
        X        Y       Z
1 -0.7265 -0.97528 -0.6381
2  0.9769 -0.03947 -0.6381
3 -0.2921  1.18387  1.4888

Clustering vector:
Guillermo    Rachel    Deidra    Quaton    Alicia Angelique  Johnaton
        1         3         2         2         1         3         1
    Javan   Khulood      Cody   Cameron     Heidi    Maahir   Rogelio
        1         1         1         1         3         3         2
    Erica    Barren     Kiana     Elyse  Chadrick    Tahani
        2         2         3         3         2         2

Within cluster sum of squares by cluster:
[1] 0.9928 5.2115 5.8304
 (between_SS / total_SS =  78.9 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"
[5] "tot.withinss" "betweenss"    "size"         "iter"
[9] "ifault"
```

K-Means Clusters vs. Principal Components ($k = 3$)



Alternative: “Partitioning Around Medoids” ($k = 3$)

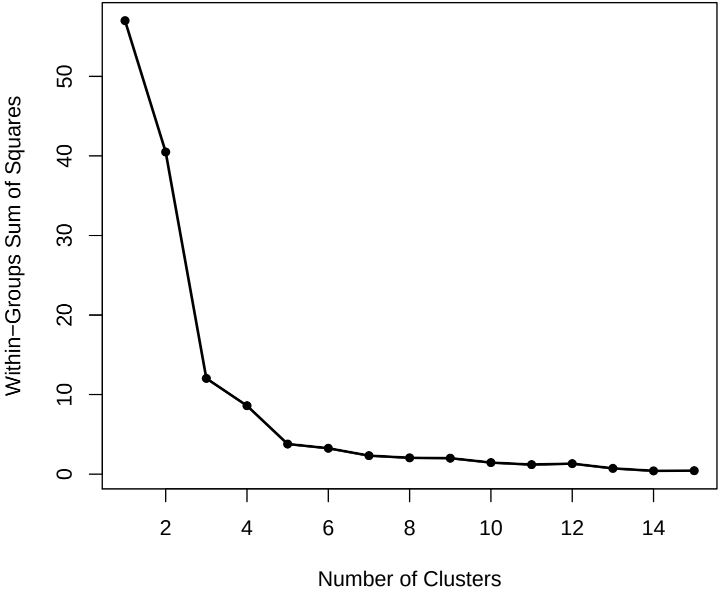
```
> PAM3 <- pam(ds,3)
> PAM3
Medoids:
         ID       X      Y       Z
Johnaton  7 -0.6226 -1.037 -0.6381
Heidi    12 -0.3315  1.297  1.4888
Erica    15  1.5634 -0.468 -0.6381
Clustering vector:
Guillermo    Rachel    Deidra    Quaton    Alicia Angelique  Johnaton
        1         2         1         3         1         2         1
    Javan   Khulood      Cody   Cameron     Heidi    Maahir  Rogelio
        1         1         1         1         2         2         3
    Erica    Barren     Kiana     Elyse  Chadrick    Tahani
        3         3         2         2         3         3
Objective function:
 build   swap
0.7054 0.6573

Available components:
 [1] "medoids"    "id.med"     "clustering" "objective"  "isolation"
 [6] "clusinfo"   "silinfo"    "diss"       "call"       "data"
```


PAM Cluster Plot (k=3)
Component 2
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2
−1
0
1
Component 1
These two components explain 93.86 % of the point variability.

Practical k-Means: Choosing k

- Theory
- Scree plot of WCSS
- “Model-based” approaches

Choosing k: Scree Plot



Other Non-Hierarchical Methods

DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

- *Density-based* method...
- Does not require prespecification of k
- Also does not (necessarily) assign "outlying" observations to clusters
- R packages: `dbscan`, others

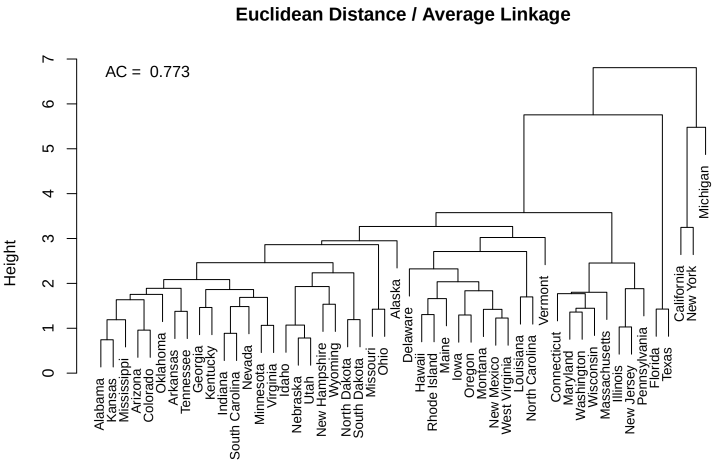
Mean-Shift Clustering

- Operationally similar to DBSCAN
- IME works well with "non-spherical" cluster shapes
- R packages: `meanShiftR`, `LPCM`, etc.

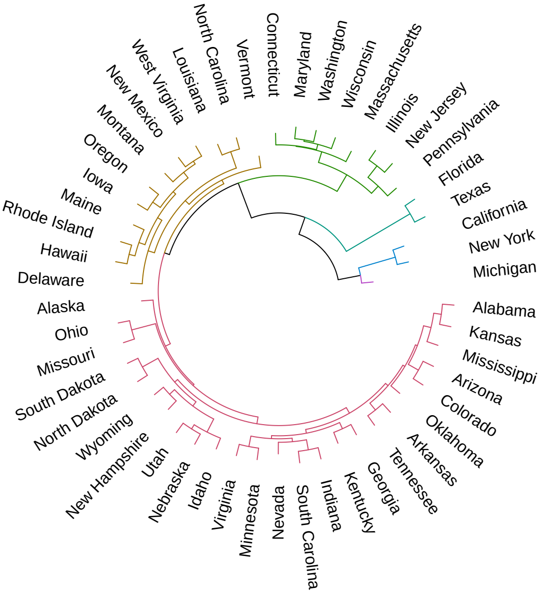
Real-Data Example: U.S. States

```
> url <- getURL("https://raw.githubusercontent.com/PrisonRodeo/
                PLSC504-2020-git/master/Data/States2005.csv")
> States <- read.csv(text = url)
>
> summary(States)
     statename        Year      CitizenIdeology GovernmentIdeology    govstaff
 Alabama   : 1   Min.   :2005   Min.   :28.2    Min.   :10.1       Min.   :  8.0
 Alaska    : 1   1st Qu.:2005   1st Qu.:43.5    1st Qu.:21.9       1st Qu.: 24.0
 Arizona   : 1   Median :2005   Median :53.1    Median :47.9       Median : 39.0
 Arkansas  : 1   Mean   :2005   Mean   :53.2    Mean   :49.9       Mean   : 59.1
 California: 1   3rd Qu.:2005   3rd Qu.:61.3    3rd Qu.:71.8       3rd Qu.: 69.5
 Colorado  : 1   Max.   :2005   Max.   :91.2    Max.   :92.0       Max.   :310.0
 (Other)   :44
   govsalary          legcomp          legsession          pop            lnGDP
 Min.   : 70000   Min.   :   200   Min.   : 25.0   Min.   :  501   Min.   :10.0
 1st Qu.: 95000   1st Qu.: 15876   1st Qu.: 45.0   1st Qu.: 1772   1st Qu.:11.0
 Median :112822   Median : 23696   Median : 67.5   Median : 4210   Median :11.9
 Mean   :115778   Mean   : 31932   Mean   : 79.0   Mean   : 5918   Mean   :11.9
 3rd Qu.:131326   3rd Qu.: 41709   3rd Qu.: 99.2   3rd Qu.: 6398   3rd Qu.:12.6
 Max.   :179000   Max.   :118600   Max.   :352.0   Max.   :36154   Max.   :14.3

> StS <- data.frame(scale(States[,3:10]))
> rownames(StS)<-States$statename
```

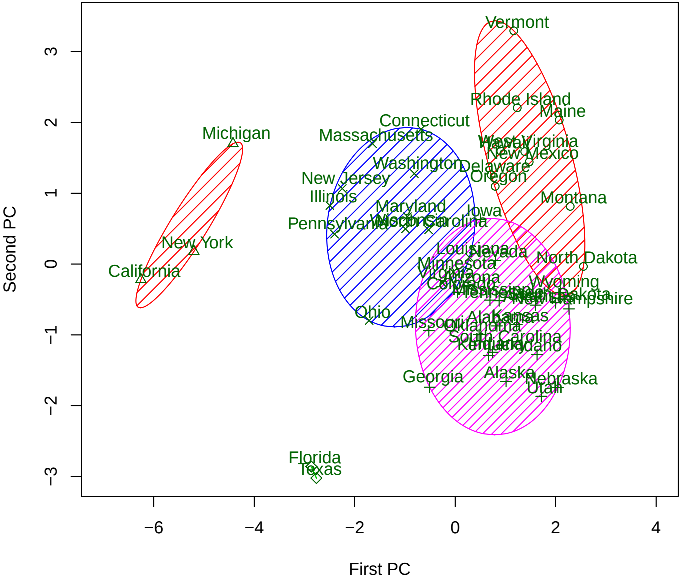
State Data: Agglomerative Dendrogram



State Data: Cooler Agglomerative Dendrogram



State Data: K-Means Results



Useful References

- Johnson, S.C. 1967. "Hierarchical Clustering Schemes." *Psychometrika* 32:241-254.

- Reynolds, A., Richards, G., de la Iglesia, B. and Rayward-Smith, V. 1992. "Clustering Rules: A Comparison of Partitioning and Hierarchical Clustering Algorithms." *Journal of Mathematical Modelling and Algorithms* 5:475-504.

- Kaufman, Leonard, and Peter J. Rousseeuw. 2005. *Finding Groups in Data: An Introduction to Cluster Analysis.* New York: Wiley.

- Hennig, Christian, Marina Meila, Fionn Murtagh, and Roberto Rocci, eds. 2015. *Handbook of Cluster Analysis*. New York: Chapman & Hall.

- Everitt, Brian S., Sabine Landau, Morven Leese, and Daniel Stahl. 2011. *Cluster Analysis*, 5th Ed. New York: Wiley.

- Kassambara, Alboukadel. 2017. *Practical Guide to Cluster Analysis in R.* Createspace.

Useful R Packages and Routines

- `hclust` and `kmeans` (in `stats`)
- `agnes` and `diana` and `pam` (in `cluster`)
- `amap` (alternative agglomerative and *k*-means clustering)
- `dendextend` (additional functionality for dendograms; e.g., comparisons)
- `mclust` (model-based clustering via MLE)
- `FactoClass` (combinations of factorial and clustering methods)

... and many more.

Useful Links

- The Cluster Analysis R Task View: http://cran.cnr.berkeley.edu/web/views/Cluster.html
- The Data Flair R Clustering tutorial: https://data-flair.training/blogs/r-clustering-tutorial/
- The `dendextend` vignette: https://cran.r-project.org/web/packages/dendextend/vignettes/Cluster_Analysis.html

Item Response Theory (IRT)

Item Response Theory ("IRT")

- Origins in psychometrics / testing
- *Measurement* model – (typically) *no* **X**
- *Unidimensional*
- *Discrete* responses **Y**
- Equally descriptive and inferential

We have:

$$Y^* = \text{ latent trait (“ability”)}$$

and:

$$Y = \text{observed measures}$$

- $i \in \{1, 2 \dots N\}$ indexes *subjects* / *units*, and
- $j \in \{1, 2, \dots J\}$ indexes *items* / *measures*.

$$Y_{ij} = \begin{cases} 0 & \text{if subject } i \text{ gets item } j \text{ “incorrect,”} \\ 1 & \text{if subject } i \text{ gets item } j \text{ “correct.”} \end{cases}$$

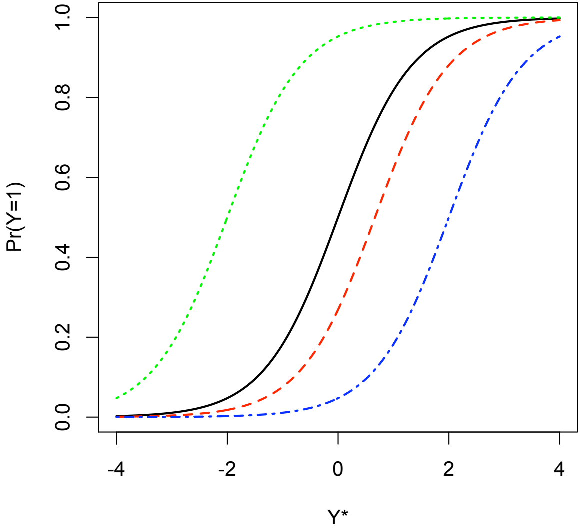
One-Parameter Logistic Model ("1PLM")

Formally:

$$\Pr(Y_{ij} = 1) = \frac{\exp(\theta_i - \beta_j)}{1 + \exp(\theta_i - \beta_j)}$$

Here,

- θ_i = respondent i's *ability*,
- β_j = item j's *difficulty*.
- $\beta_j \equiv$ value of Y^* where $\Pr(Y_{ij} = 1) = 0.50$



Pr(Y=1)
0.0
0.2
0.4
0.6
0.8
1.0
-4
-2
0
2
4
Y*

a.k.a. the “Rasch” model (Rasch 1960):

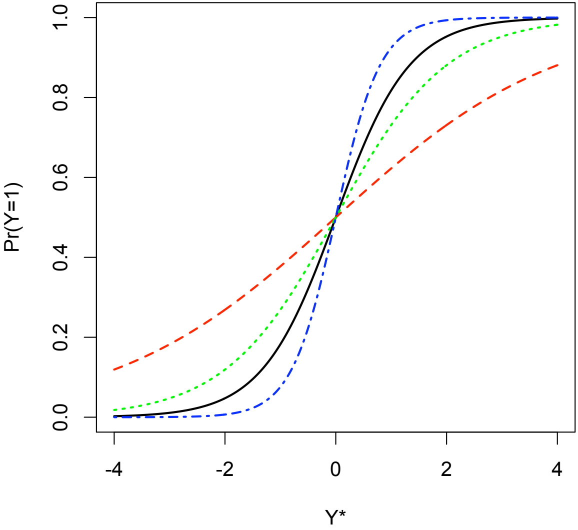
- Implicit “slope” $= 1.0$
- Implies items are equally “discriminating”
- If not...

Two-Parameter Logistic Model ("2PLM")

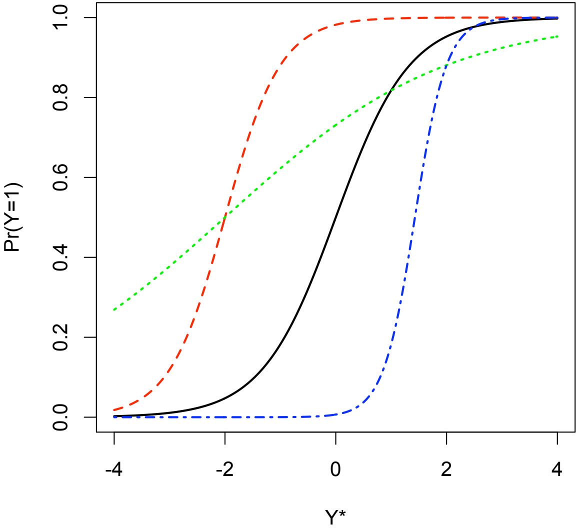
$$\Pr(Y_{ij} = 1) = \frac{\exp[\alpha_j(\theta_i - \beta_j)]}{1 + \exp[\alpha_j(\theta_i - \beta_j)]}$$

- θ_i = respondent *i*'s *ability*,
- β_j = item *j*'s *difficulty*,
- α_j = item *j*'s *discrimination*.

Identical Difficulty, Different Discrimination



Different Difficulty & Discrimination



The 2PLM...

- ... is due to Birnbaum (1968)
- ...is similar to a "typical" logit...
- ...nests the 1PLM as a special case (when $\alpha_j = 1 \forall j$)

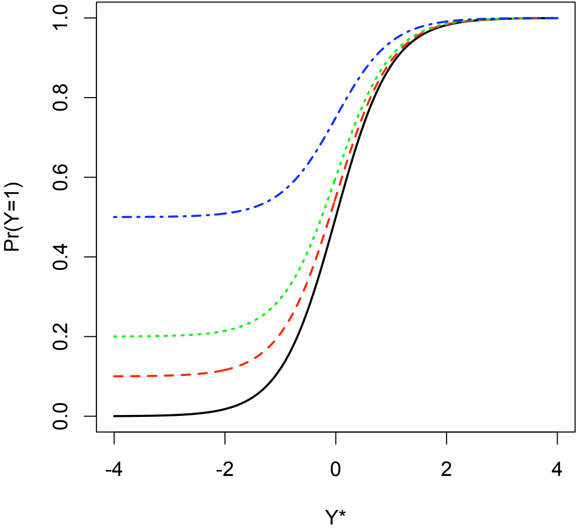
Three-Parameter Logistic Model ("3PLM")

Then there's this:

$$\Pr(Y_{ij} = 1) = \delta_j + (1 - \delta_j) \left\{ \frac{\exp[\alpha_j(\theta_i - \beta_j)]}{1 + \exp[\alpha_j(\theta_i - \beta_j)]} \right\}$$

- θ_i = respondent i's *ability*,
- β_j = item j's *difficulty*,
- α_j = item j's *discrimination*.
- δ_j = *lower asymptote* of $\Pr(Y_{ij} = 1)$ (incorrectly: "guessing" parameter).

3PLM, Constant α & β, Varying δ



- *Unidimensionality*
- *Local Item Independence* ("No LID"):

$$\text{Cov}(Y_{ij}, Y_{ik}|\theta_i) = 0 \; \forall \; j \neq k$$

$$P_{ij} = \Pr(Y_{ij} = 1),$$

$$\begin{aligned} Q_{ij} &= \Pr(Y_{ij} = 0) \\ &= 1 - \Pr(Y_{ij} = 1), \end{aligned}$$

$$\boldsymbol{\Psi} = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_J \\ \alpha_1 \\ \vdots \\ \alpha_J \\ \delta_1 \\ \vdots \\ \delta_J \end{pmatrix}.$$

Estimation: Likelihoods

Known $\Psi = \alpha$, β, δ:

$$L(\mathbf{Y}|\Psi) = \prod_{j=1}^{J} P_{ij}^{Y_{ij}} Q_{ij}^{1-Y_{ij}}.$$

Known θ:

$$L(\mathbf{Y}|\theta) = \prod_{i=1}^{N} P_{ij}^{Y_{ij}} Q_{ij}^{1-Y_{ij}}.$$

Estimation: Likelihoods

$$L(\mathbf{Y}|\Psi, \theta) = \prod_{i=1}^{N} \prod_{j=1}^{J} P_{ij}^{Y_{ij}} Q_{ij}^{1-Y_{ij}}$$

$$\ln L(\mathbf{Y}|\Psi, \theta) = \sum_{i=1}^{N} \sum_{j=1}^{J} Y_{ij} \ln P_{ij} + (1 - Y_{ij}) Q_{ij}.$$

- $N + J$ parameters in the 1PLM,
- $N + 2J$ parameters in the 2PLM,
- $N + 3J$ parameters in the 3PLM.

But...

- NJ observations,
- Asymptotics as $N \to \infty$, $J \to \infty$...

Estimation: Conditional Likelihood

Total score is:

$$T_i = \sum_{j=1}^{J} Y_{ij} \ \in \ \{0, 1, ...J\}$$

$$L = \prod_{i=1}^{N} \frac{\exp[\alpha_j(\theta_t - \beta_j)]}{1 + \exp[\alpha_j(\theta_t - \beta_j)]}$$

θ_t are "score-group" parameters corresponding to the $J + 1$ possible values of T.

- Equivalent to fitting a conditional logit model:

$$\Pr(Y_{ij} = 1) = \frac{\exp(\mathbf{Z}_{ij}\gamma)}{\sum_{j=1}^{J} \exp(\mathbf{Z}_{ij}\gamma)}$$

  with $\mathbf{Z}_{ij} =$ "item dummies."

- Useful only for 1PLM (since T_i is a sufficient statistic for θ_i).

Estimation: Marginal Likelihood

$$L(\mathbf{Y}|\Psi,\theta) = \prod_{i=1}^{N}\left[\int_{-\infty}^{\infty}\prod_{j=1}^{J}P_{ij}^{Y_{ij}}Q_{ij}^{1-Y_{ij}}d\theta\right]$$

- Analogous to "random effects"...
- Eliminates inconsistency as $N \rightarrow \infty$, *but*
- Requires *strong* exogeneity of θ and Ψ.

Estimation: Bayesian Approaches

- Place priors on θ, Ψ;
- Estimate via sampling from posteriors, via MCMC.
- Eliminates problems with $\hat{\alpha}$, $\hat{\beta}$, $\hat{\theta} = \infty$ (see below).
- Easily extensible to other circumstances (hierarchical/multilevel, etc.)

Two Issues:

- *Scale* invariance: $L(\hat{\Psi}) = L(\hat{\Psi} + c)$
- *Rotational* invariance: $L(\hat{\Psi}) = L(-\hat{\Psi})$

Fixes:

- Set one (arbitrary) $\beta_j = 0$, and another (arbitrary) $\beta_k > 0$, or
- Fix two θ_is at specific values.

Further (Potential) Concerns

- $Y_{ij} = 0/1 \; \forall \; i \; \rightarrow \; \beta_j = \pm\infty$.
- $Y_{ij} = 0/1 \; \forall \; j \; \rightarrow \; \theta_i = \pm\infty$.
- Separation / "empty cells" $\rightarrow \; \alpha_j = \pm\infty$.
- Problematic for joint and conditional approaches; more easily dealt with in the Bayesian framework.

- Estimates of $\hat{\alpha}$s, $\hat{\beta}$s, and/or $\hat{\delta}$s, plus $\hat{\theta}$s
- Associated s.e.s / c.i.s
- "Scale-free" quantities of interest...

IRT Models in R

- Library `ltm` (marginal estimation)
  - `rasch` (1PLM)
  - `ltm` (2PLM)
  - `tpm` (3PLM)
- Library `MCMCpack` (Bayesian estimation)
  - 1 and 2PLM
  - Standard, hierarchical, dyamic, multidimensional
- `ideal` (in library `pscl`) (Bayesian estimation)
  - 1 and 2PLM
  - k-dimensional
  - takes a `rollcall` object
- Other packages: `eRm`, `irtoys`, `irtProb`, `MiscPsycho`, etc.

Example: SCOTUS Voting, 1994-2005

```
> summary(SCOTUS)

       id            Rehnquist         Stevens          OConnor          Scalia
 Min.   :   1    Min.   :0.00    Min.   :0.00    Min.   :0.0    Min.   :0.00
 1st Qu.: 377    1st Qu.:0.00    1st Qu.:0.00    1st Qu.:0.0    1st Qu.:0.00
 Median : 753    Median :0.00    Median :1.00    Median :0.0    Median :0.00
 Mean   : 753    Mean   :0.28    Mean   :0.69    Mean   :0.4    Mean   :0.27
 3rd Qu.:1129    3rd Qu.:1.00    3rd Qu.:1.00    3rd Qu.:1.0    3rd Qu.:1.00
 Max.   :1505    Max.   :1.00    Max.   :1.00    Max.   :1.0    Max.   :1.00
                 NA's   :49      NA's   :51      NA's   :55     NA's   :41
    Kennedy           Souter          Thomas          Ginsburg          Breyer
 Min.   :0.00    Min.   :0.0    Min.   :0.00    Min.   :0.00    Min.   :0.00
 1st Qu.:0.00    1st Qu.:0.0    1st Qu.:0.00    1st Qu.:0.00    1st Qu.:0.00
 Median :0.00    Median :1.0    Median :0.00    Median :1.00    Median :1.00
 Mean   :0.37    Mean   :0.6    Mean   :0.25    Mean   :0.61    Mean   :0.57
 3rd Qu.:1.00    3rd Qu.:1.0    3rd Qu.:0.00    3rd Qu.:1.00    3rd Qu.:1.00
 Max.   :1.00    Max.   :1.0    Max.   :1.00    Max.   :1.00    Max.   :1.00
 NA's   :32      NA's   :37     NA's   :44      NA's   :39      NA's   :61
```

1PLM Using rasch

```
> # 1PLM / Rasch Model:
> require(ltm)
> OnePLM<-rasch(SCOTUS[c(2:10)])
> summary(OnePLM)

Model Summary:
 log.Lik   AIC   BIC
   -5529 11079 11132

Coefficients:
                 value std.err z.vals
Dffclt.Rehnquist  0.46   0.040   11.5
Dffclt.Stevens   -0.59   0.030  -19.8
Dffclt.OConnor    0.14   0.030    4.6
Dffclt.Scalia     0.52   0.041   12.5
Dffclt.Kennedy    0.21   0.032    6.5
Dffclt.Souter    -0.36   0.027  -13.1
Dffclt.Thomas     0.60   0.043   13.8
Dffclt.Ginsburg  -0.37   0.027  -13.4
Dffclt.Breyer    -0.26   0.027   -9.9
Dscrmn            3.74   0.130   28.9

Integration:
method: Gauss-Hermite
quadrature points: 21

Optimization:
Convergence: 0
max(|grad|): 0.0027
quasi-Newton: BFGS
```

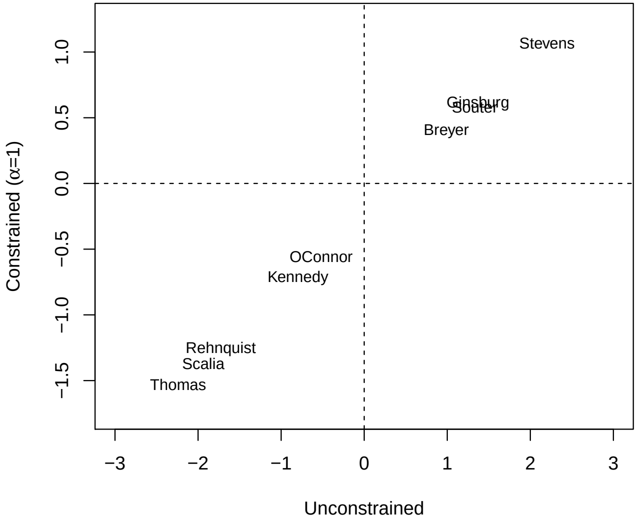
Converted to $\widehat{\Pr(Y_i = 1|\theta_i} = 0)$

```
> # Convert to probabilities given theta=0
>
> coef(OnePLM, prob=TRUE, order=TRUE)
          Dffclt Dscrmn P(x=1|z=0)
Stevens    -0.59    3.7      0.900
Ginsburg   -0.37    3.7      0.797
Souter     -0.36    3.7      0.791
Breyer     -0.26    3.7      0.729
OConnor     0.14    3.7      0.373
Kennedy     0.21    3.7      0.311
Rehnquist   0.46    3.7      0.151
Scalia      0.52    3.7      0.126
Thomas      0.60    3.7      0.096
```

Alternative Model Constraining $\alpha = 1.0$

```
> AltOnePLM<-rasch(IRTData, constraint=cbind(length(IRTData)+1,1))
> summary(AltOnePLM)

Model Summary:
 log.Lik   AIC   BIC
   -6452 12923 12971

Coefficients:
                 value std.err z.vals
Dffclt.Rehnquist  1.26   0.073   17.3
Dffclt.Stevens   -1.07   0.071  -15.1
Dffclt.OConnor    0.56   0.069    8.1
Dffclt.Scalia     1.37   0.074   18.6
Dffclt.Kennedy    0.72   0.069   10.4
Dffclt.Souter    -0.58   0.068   -8.6
Dffclt.Thomas     1.53   0.075   20.3
Dffclt.Ginsburg  -0.61   0.068   -8.9
Dffclt.Breyer    -0.40   0.068   -5.9
Dscrmn            1.00      NA     NA
```

Constrained and Unconstrained 1PLM $\hat{\beta}$s



2PLM

```
> TwoPLM<-ltm(IRTData ~ z1)
> summary(TwoPLM)

Coefficients:
                  value std.err z.vals
Dffclt.Rehnquist  0.44   0.035   12.3
Dffclt.Stevens   -0.63   0.038  -16.7
Dffclt.OConnor    0.14   0.026    5.6
Dffclt.Scalia     0.59   0.042   14.1
Dffclt.Kennedy    0.20   0.028    7.2
Dffclt.Souter    -0.27   0.025  -10.7
Dffclt.Thomas     0.68   0.044   15.2
Dffclt.Ginsburg  -0.29   0.025  -11.8
Dffclt.Breyer    -0.24   0.025   -9.6
Dscrmn.Rehnquist  4.77   0.377   12.7
Dscrmn.Stevens    2.46   0.165   14.9
Dscrmn.OConnor    4.14   0.341   12.1
Dscrmn.Scalia     2.82   0.188   15.0
Dscrmn.Kennedy    4.74   0.448   10.6
Dscrmn.Souter     6.69   0.535   12.5
Dscrmn.Thomas     2.84   0.190   14.9
Dscrmn.Ginsburg   5.83   0.439   13.3
Dscrmn.Breyer     3.76   0.253   14.9
```

2PLM: Probabilities and Testing

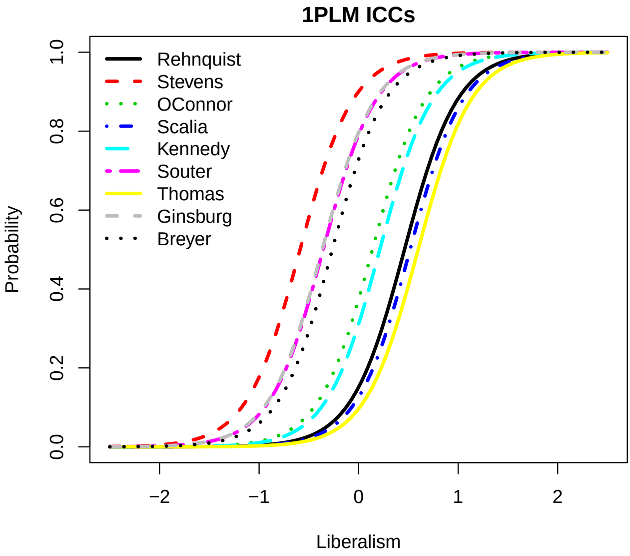
```
> coef(TwoPLM, prob=TRUE, order=TRUE)

          Dffclt Dscrmn P(x=1|z=0)
Stevens    -0.63    2.5       0.82
Ginsburg   -0.29    5.8       0.85
Souter     -0.27    6.7       0.86
Breyer     -0.24    3.8       0.71
OConnor     0.14    4.1       0.35
Kennedy     0.20    4.7       0.28
Rehnquist   0.44    4.8       0.11
Scalia      0.59    2.8       0.16
Thomas      0.68    2.8       0.13
```

```
> anova(OnePLM, TwoPLM)

 Likelihood Ratio Table
         AIC   BIC log.Lik   LRT df p.value
OnePLM 11079 11132   -5529
TwoPLM 10882 10978   -5423 212.7  8  <0.001
```

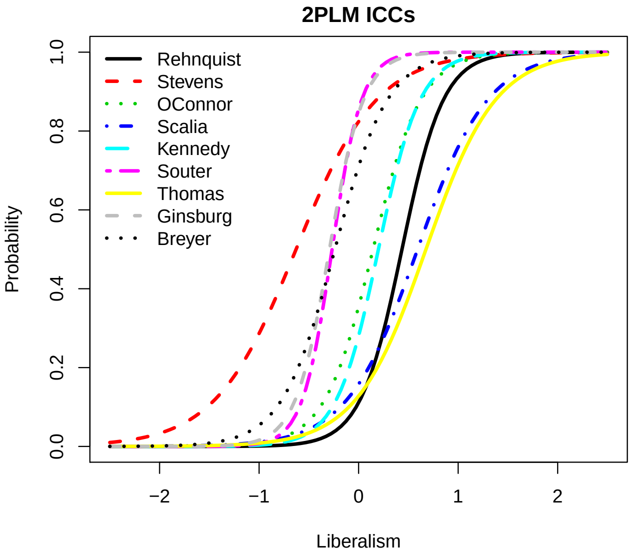
```
> ThreePLM<-tpm(IRTData)
> summary(ThreePLM)

Coefficients:
                   value  std.err  z.vals
Gussng.Rehnquist  0.049    0.008   6.260
Gussng.Stevens    0.000    0.001   0.018
Gussng.OConnor    0.043    0.013   3.415
Gussng.Scalia     0.097    0.011   9.119
Gussng.Kennedy    0.071    0.014   5.162
Gussng.Souter     0.011    0.029   0.386
Gussng.Thomas     0.087    0.010   8.900
Gussng.Ginsburg   0.000    0.000   0.009
Gussng.Breyer     0.000    0.000   0.004
Dffclt.Rehnquist  0.716    0.030  23.511
Dffclt.Stevens   -0.630    0.038 -16.434
Dffclt.OConnor    0.340    0.040   8.537
Dffclt.Scalia     0.759    1.766   0.430
Dffclt.Kennedy    0.500    0.041  12.170
Dffclt.Souter    -0.294    0.063  -4.642
Dffclt.Thomas     0.808   10.610   0.076
Dffclt.Ginsburg  -0.329    0.030 -10.970
Dffclt.Breyer    -0.232    0.031  -7.439
Dscrmn.Rehnquist  8.735    4.259   2.051
Dscrmn.Stevens    2.577    0.181  14.214
Dscrmn.OConnor    3.979    0.439   9.068
Dscrmn.Scalia    26.537  578.889   0.046
Dscrmn.Kennedy    4.408    0.588   7.498
Dscrmn.Souter     6.698    1.416   4.731
Dscrmn.Thomas    34.074 2779.161   0.012
Dscrmn.Ginsburg   5.800    0.509  11.394
Dscrmn.Breyer     3.538    0.231  15.335
```

```
> anova(TwoPLM, ThreePLM)

 Likelihood Ratio Table
           AIC   BIC log.Lik    LRT df p.value
TwoPLM   10882 10978   -5423
ThreePLM 10737 10881   -5342 162.94  9  <0.001
```

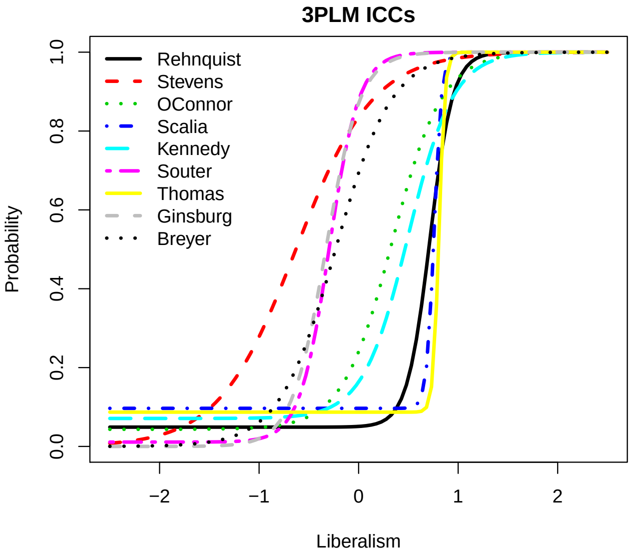
Cool Plots, I



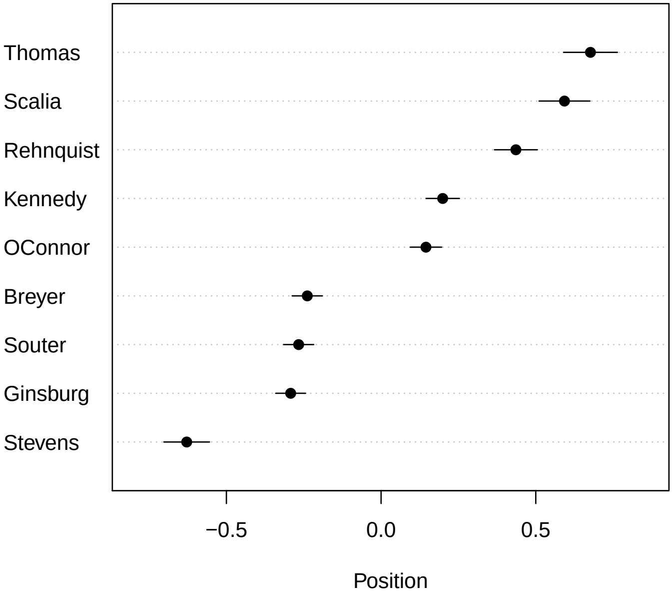
1PLM ICCs
Rehnquist
Stevens
OConnor
Scalia
Kennedy
Souter
Thomas
Ginsburg
Breyer
Probability
0.0
0.2
0.4
0.6
0.8
1.0
−2
−1
0
1
2
Liberalism


2PLM ICCs
Rehnquist
Stevens
OConnor
Scalia
Kennedy
Souter
Thomas
Ginsburg
Breyer
Probability
0.0
0.2
0.4
0.6
0.8
1.0
−2
−1
0
1
2
Liberalism

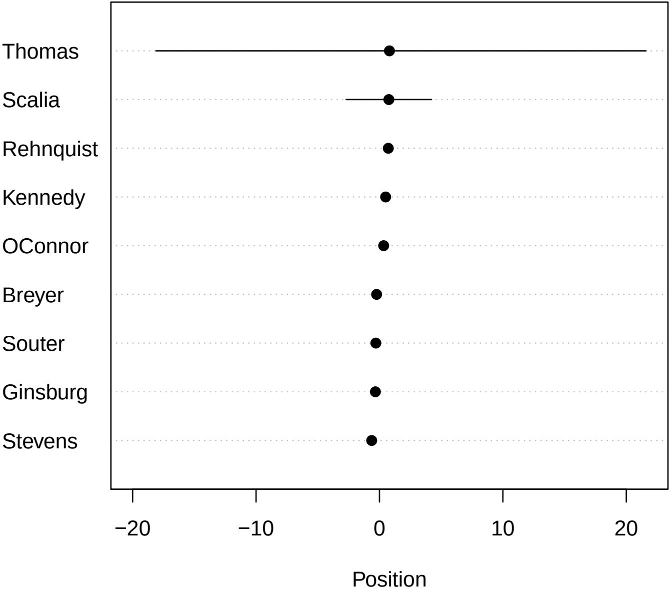
Cool Plots, III



Presenting Measures: Ladderplots (2PLM)



3PLM Ladderplot (#wtf)



Miscellaneous Things, I: Dimensionality

- Usually, *unidimensional*

- Sometimes, *two-dimensional*

- Tests:
  - Tetrachoric correlations among items
  - DIMTEST (Stout & Zhang, etc.)
  - Yen's Q_3
  - 1-D vs. 2-D comparisons (LR tests, etc.)

- *Differential item functioning*
- Formally,

$$\Pr(Y_{ij} = 1) = \Lambda[\alpha_j(\theta_i - \mathbf{X}_i\beta_j)].$$

- $\rightarrow$ violates *local item independence*

Inter alia:

- Nominal/Multinomial Y
- Ordinal Y:
    - *Graded response model* ("GRM") (Samejima 1969)
    - *Partial credit model* (Masters 1982)
    - *Generalized partial credit model* (Muraki 1992)
- Models for mixed response types (Thissen and Wainer 2001, 2003)
- Hierarchical IRT models (e.g. Bolt and Kim 2005)
- Models with covariates (e.g., DeBoeck and Wilson 2004)

Further Reading / Useful References

Hambleton, Ronald K., H. Swaminathan, and H. Jane Rogers. 1991. *Fundamentals of Item Response Theory*. Newbury Park CA: Sage Publications.

Fahrmeier, L., and G. Tutz. 2000. *Multivariate Statistical Modelling Based on Generalized Linear Models*. Berlin: Springer-Verlag.

De Boeck, Paul, and Mark Wilson, Eds. 2004. *Explanatory Item Response Models: A Generalized Linear and Nonlinear Approach.* New York: Springer.

Baker, Frank B., and Seock-Ho Kim. 2017. *The Basics of Item Response Theory Using R*. New York: Springer.

van der Linden, Wim J., Ed. 2018. *Handbook of Item Response Theory* (3 vol.). New York: Chapman & Hall CRC.

Paek, Insu, and Ki Cole. 2019. *Using R for Item Response Theory Model Applications*. New York: Routledge.

de Ayala, R. J. 2022. *The Theory and Practice of Item Response Theory*, 2nd Ed. New York: The Guilford Press.