WILLIAM & MARY

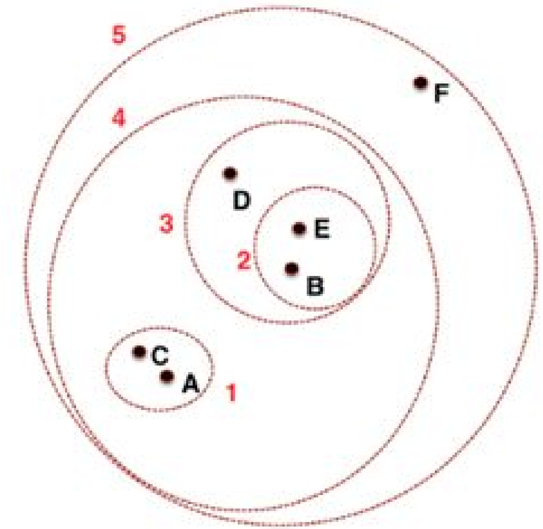
CHARTERED 1693

Hierarchical Clustering

Team 8

What Is Hierarchical Clustering?

Hierarchical clustering is an **unsupervised learning method** that allows us to visualize and analyze a collection of data as **a series of hierarchical groupings.**



Each data point is in a group called a "cluster" and each cluster is contained inside some larger cluster.

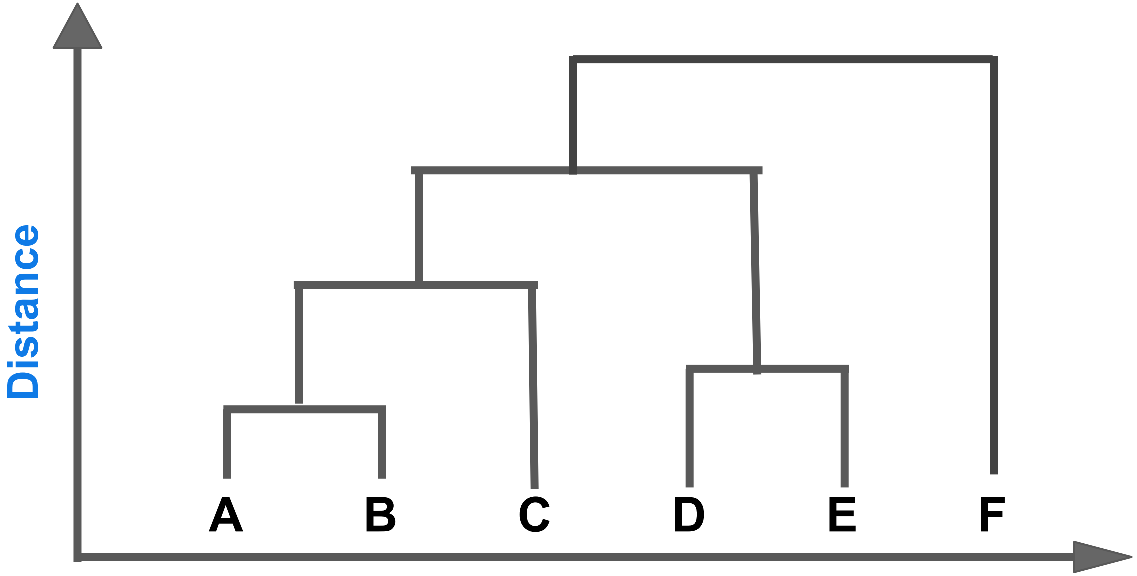
How is it Different than K-means Clustering?

Unlike K-means clustering, Hierarchical Clustering:

- does not require us to choose a K value (number of clusters)
- doesn’t specifically require use of centroids for linkage (more on this later)
- gives us a nice breakdown of the data in a tree based representation known as a dendrogram

What is a Dendrogram?

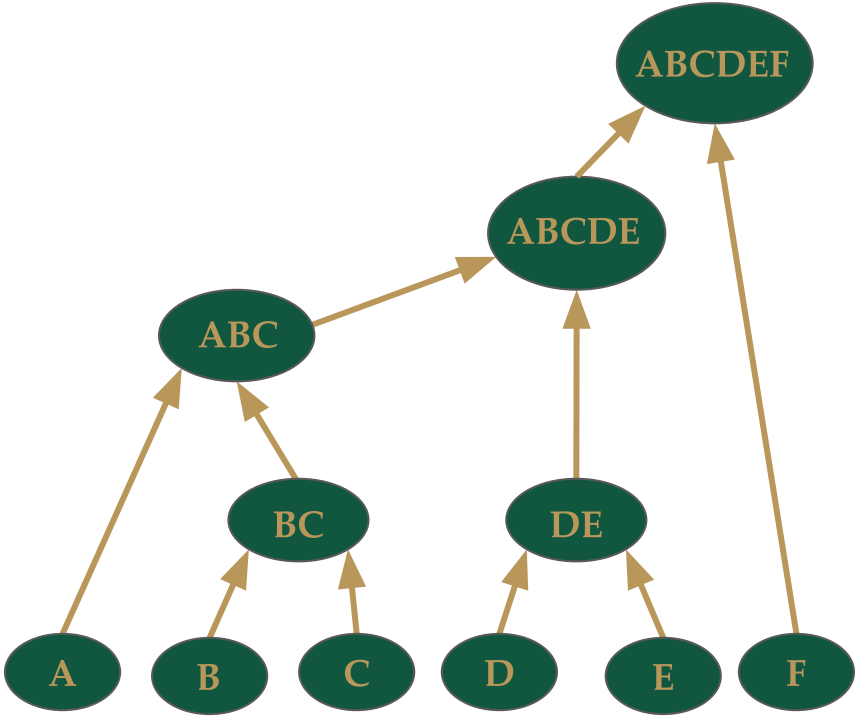
A dendrogram is a tree diagram that provides a visualization of clusters.



Bottom-up vs. Top-down Hierarchical Clustering

Bottom-up (agglomerative) approach:

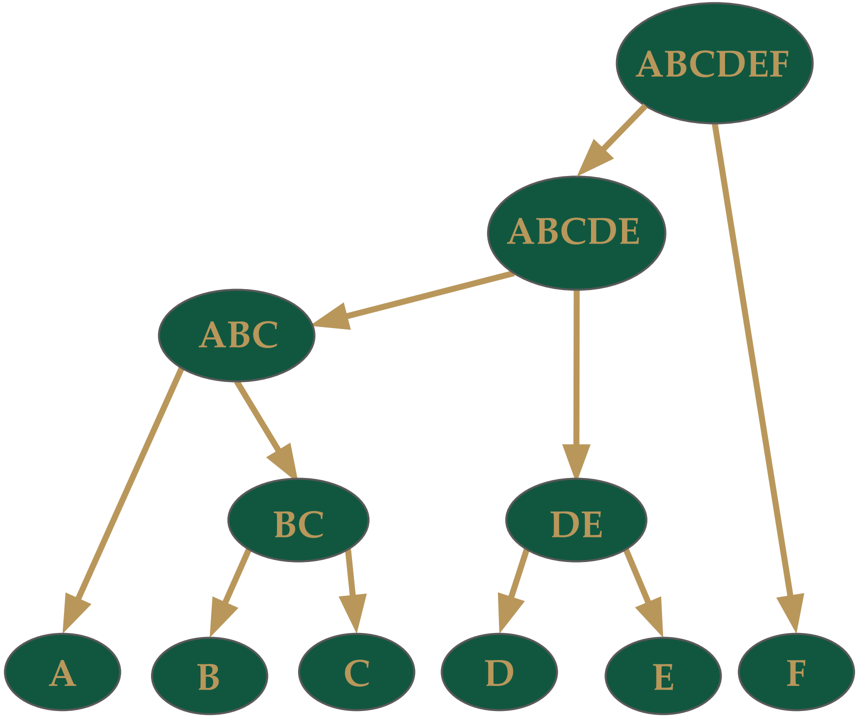
- Most common type
- Start with each node as its own cluster
- Then merge clusters iteratively until one cluster remains
- Use **linkage functions** to find the distance between clusters
- Once two clusters are joined at one level, they remain joined in all higher levels of the hierarchy.



Bottom-up vs. Top-down Hierarchical Clustering

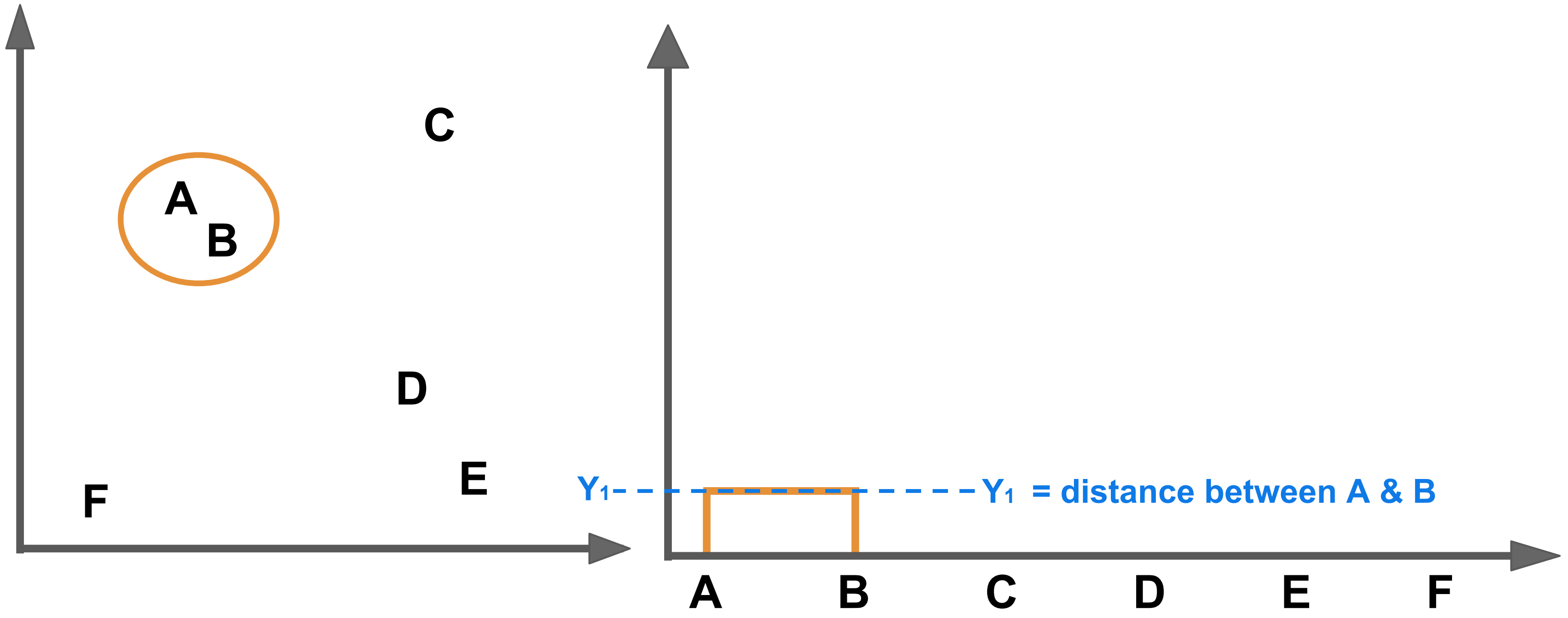
Top-down (divisive) approach:

- Start with one cluster
- Then split the most dissimilar cluster recursively until each cluster is a single node
- Use any of the same linkage functions as bottom up to calculate distance



How is an Agglomerative Dendrogram Created?

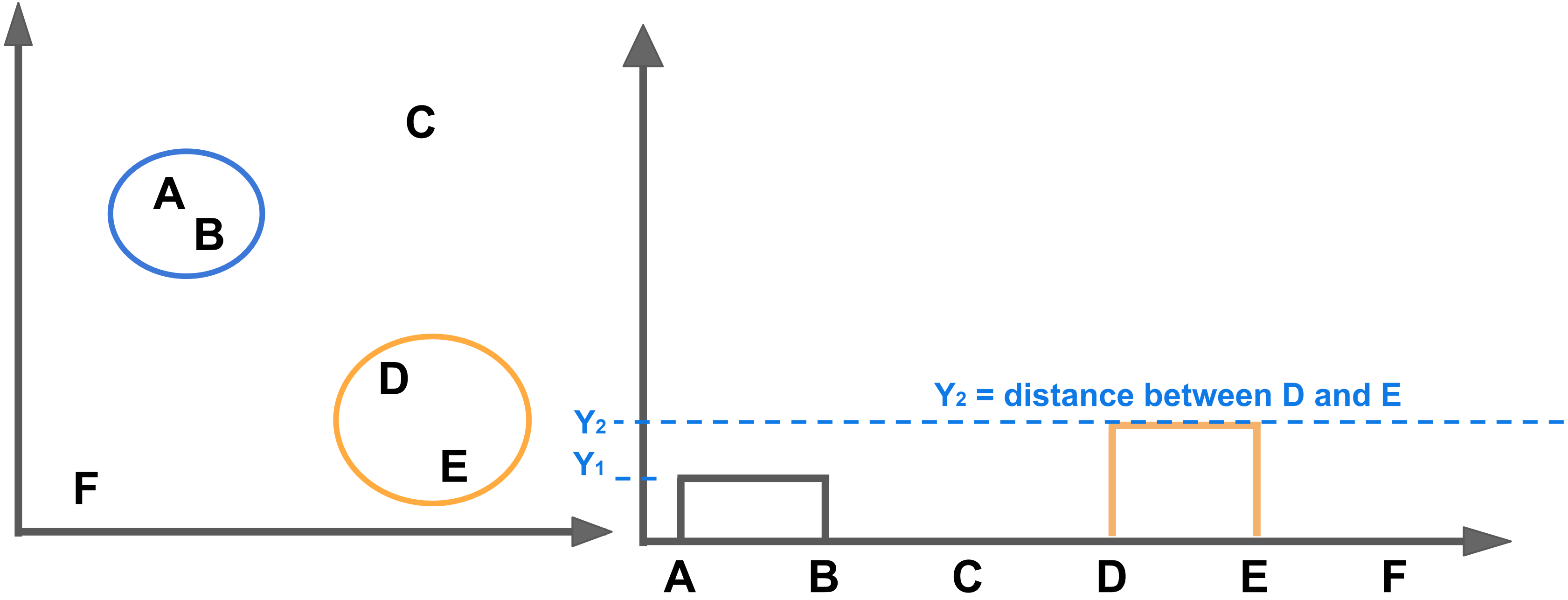
1. Start with each node as a separate cluster.
2. Identify the two most similar clusters and join them.



How is an Agglomerative Dendrogram Created?

3. Identify the next shortest distance between clusters (use linkage function).

4. Join those two clusters.

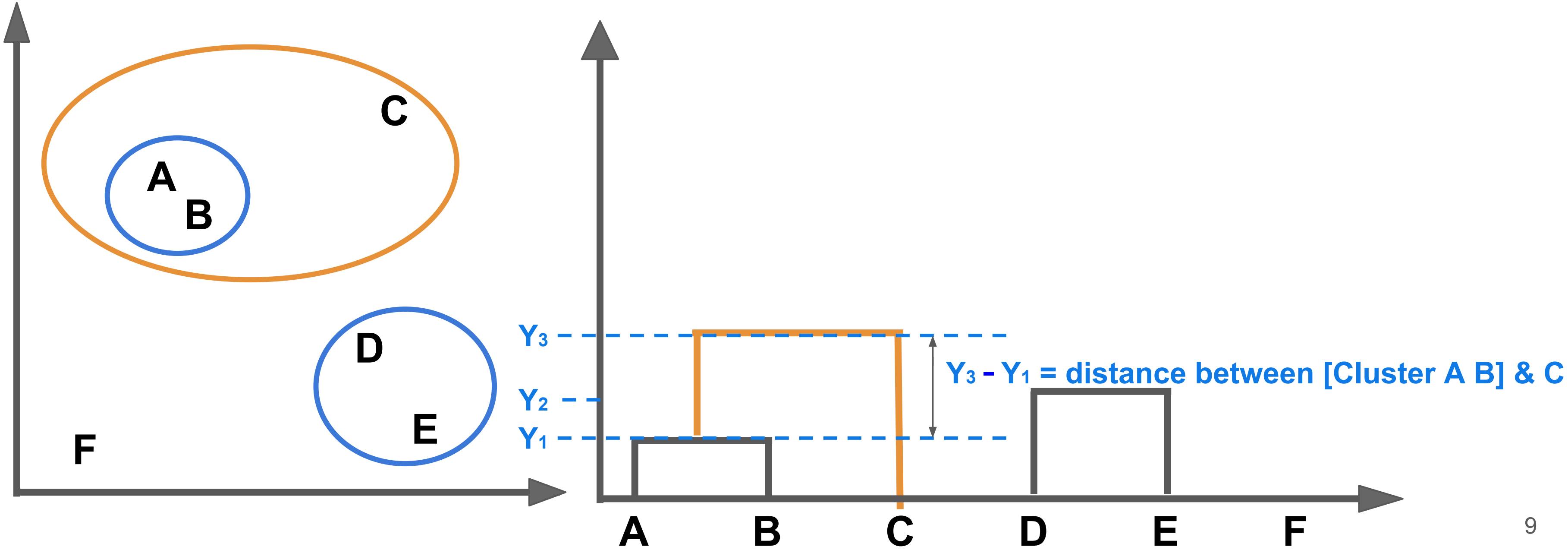


Y_2 = distance between D and E

How is an Agglomerative Dendrogram Created?

5. Again, identify the next shortest distance between clusters.

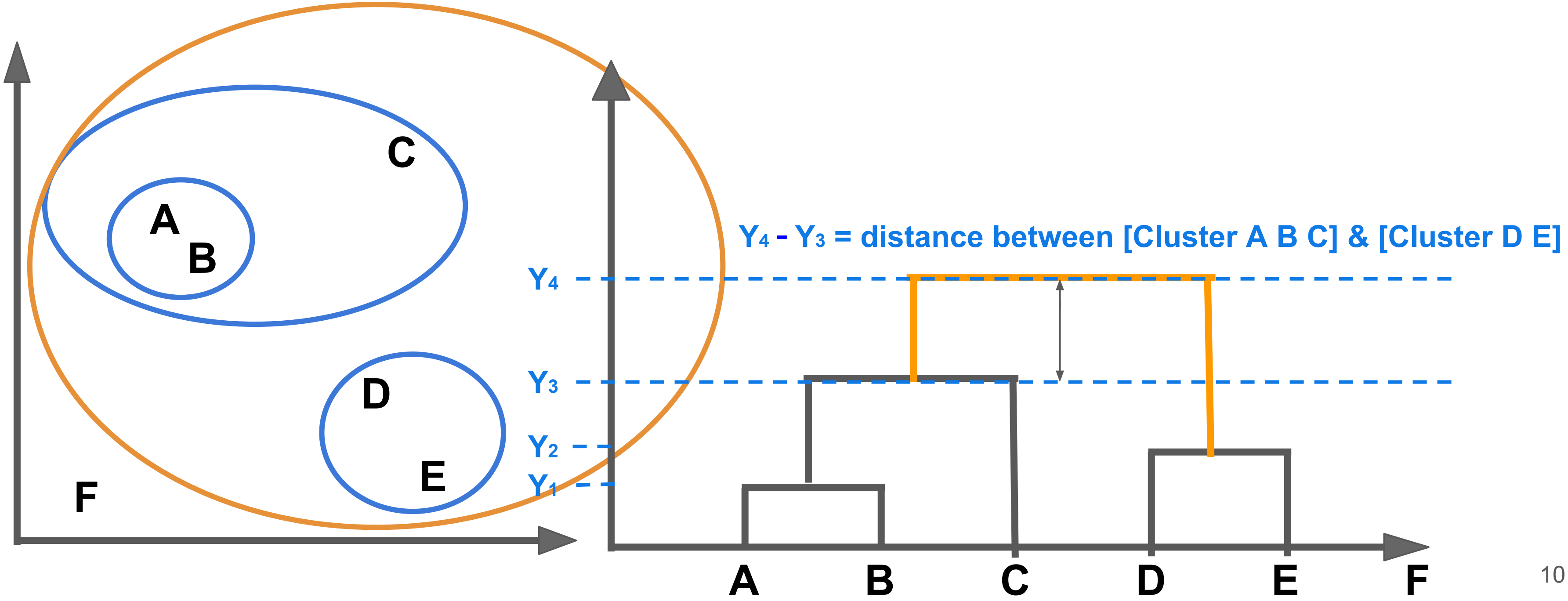
6. Join those two clusters.



Y_3 - Y_1 = distance between [Cluster A B] & C

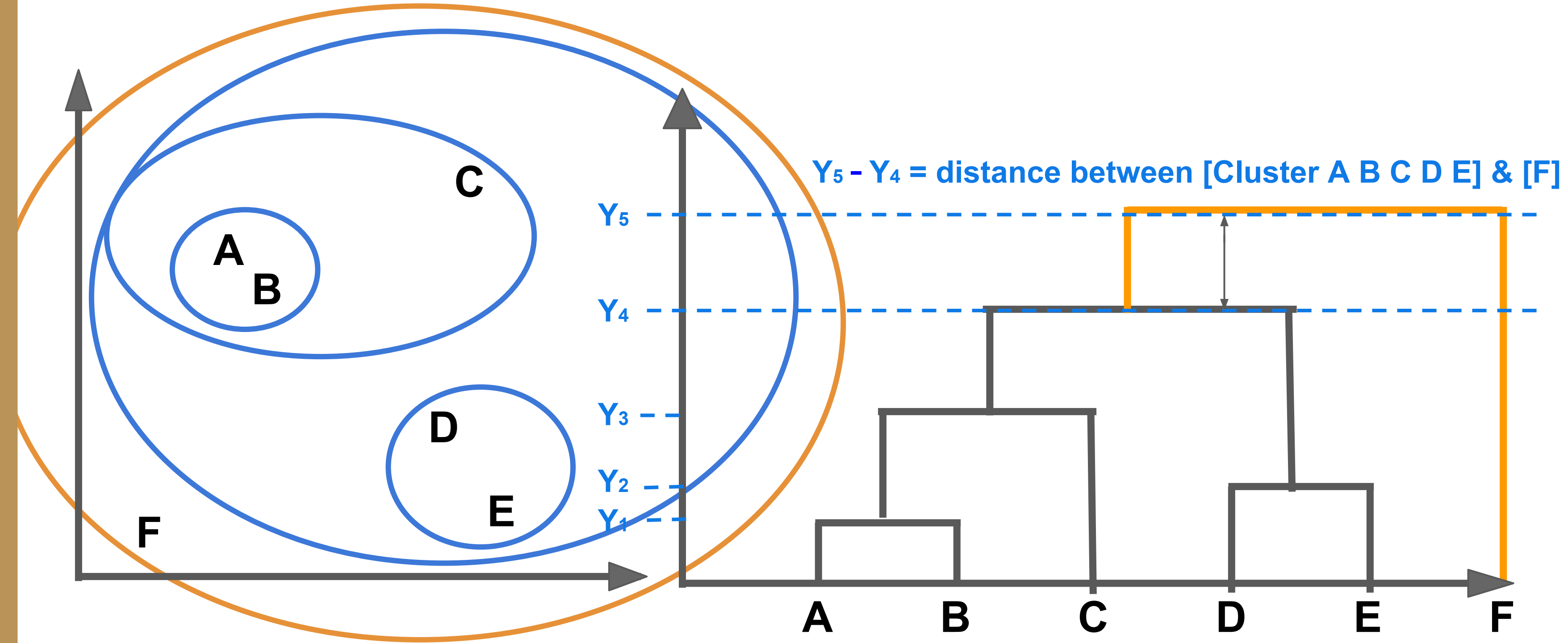
How is an Agglomerative Dendrogram Created?

7. Keep identifying the next shortest distance between clusters, and join them.


C
A
B
D
E
F
Y_4 - Y_3 = distance between [Cluster A B C] & [Cluster D E]
Y_4
Y_3
Y_2
Y_1
A
B
C
D
E
F

How is an Agglomerative Dendrogram Created?

8. Continue until you join the final two clusters.

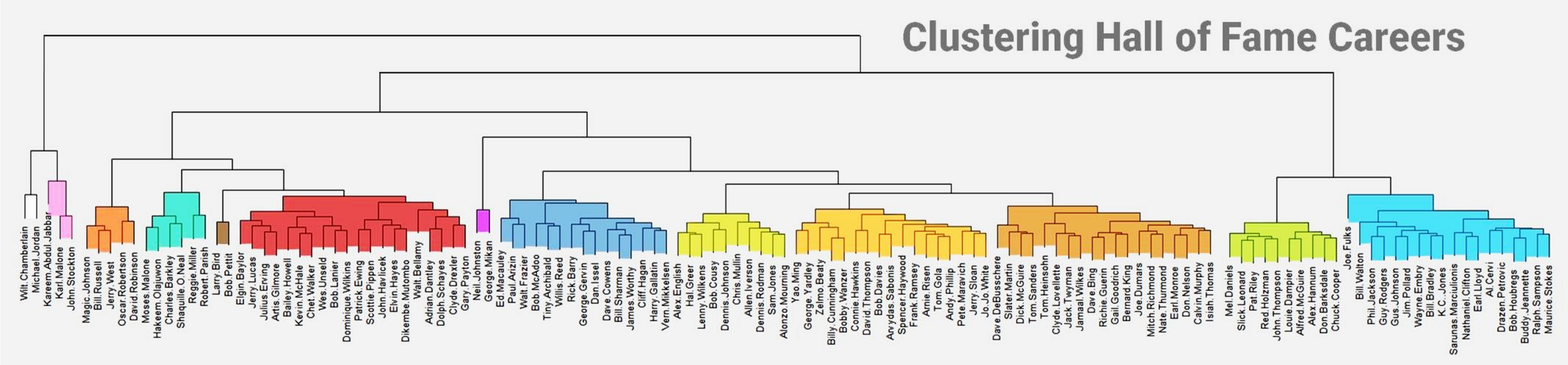


Features of Bottom-up vs. Top-down Clustering

- **Complexity**
  - **Bottom-up/Agglomerative:** O(n^3), too slow for large data sets
  - **Top-down/Divisive:** O(2^n), even worse
- **Global Structure**
  - **Bottom-up/Agglomerative:**
    - Only looks at pairs in its first step
  - **Top-down/Divisive:**
    - Has access to all of the data in its first step
    - Can find the best possible split in two parts, similar to decision trees
    - Therefore has a better global view of the structure

How To Interpret a Dendrogram

- **Y-axis:** represents **distance between clusters**
- **X-axis:** nodes are arranged in **no particular order** (except to avoid line crossing)



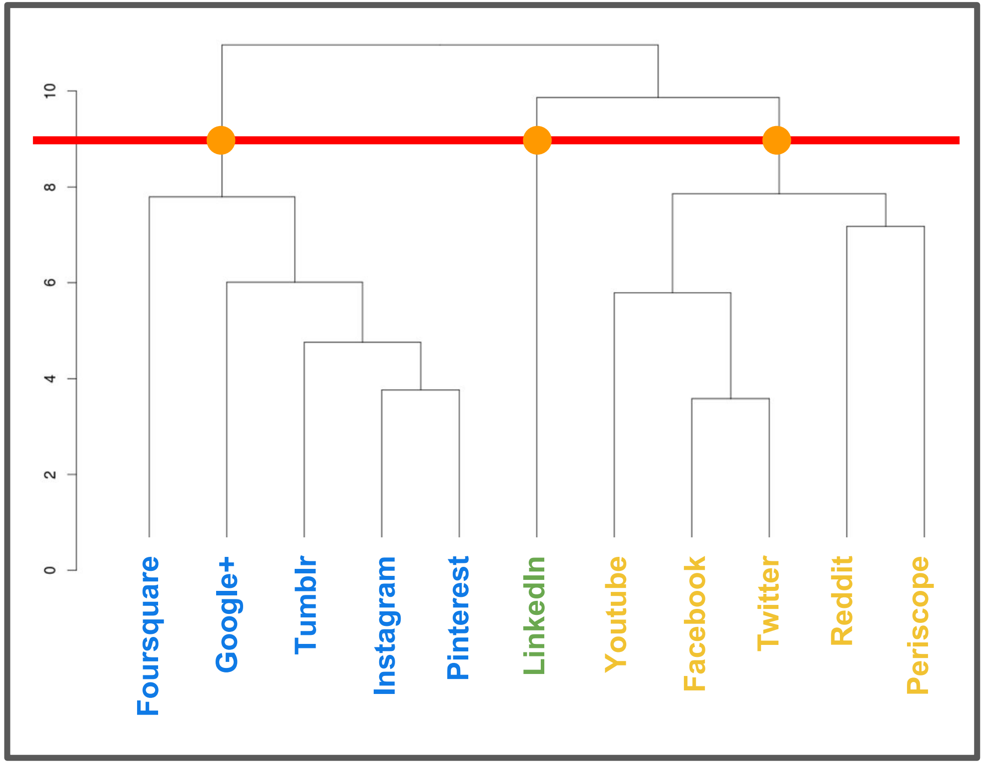
Clustering Hall of Fame Careers

Clipping

Clusters are created by **clipping:**

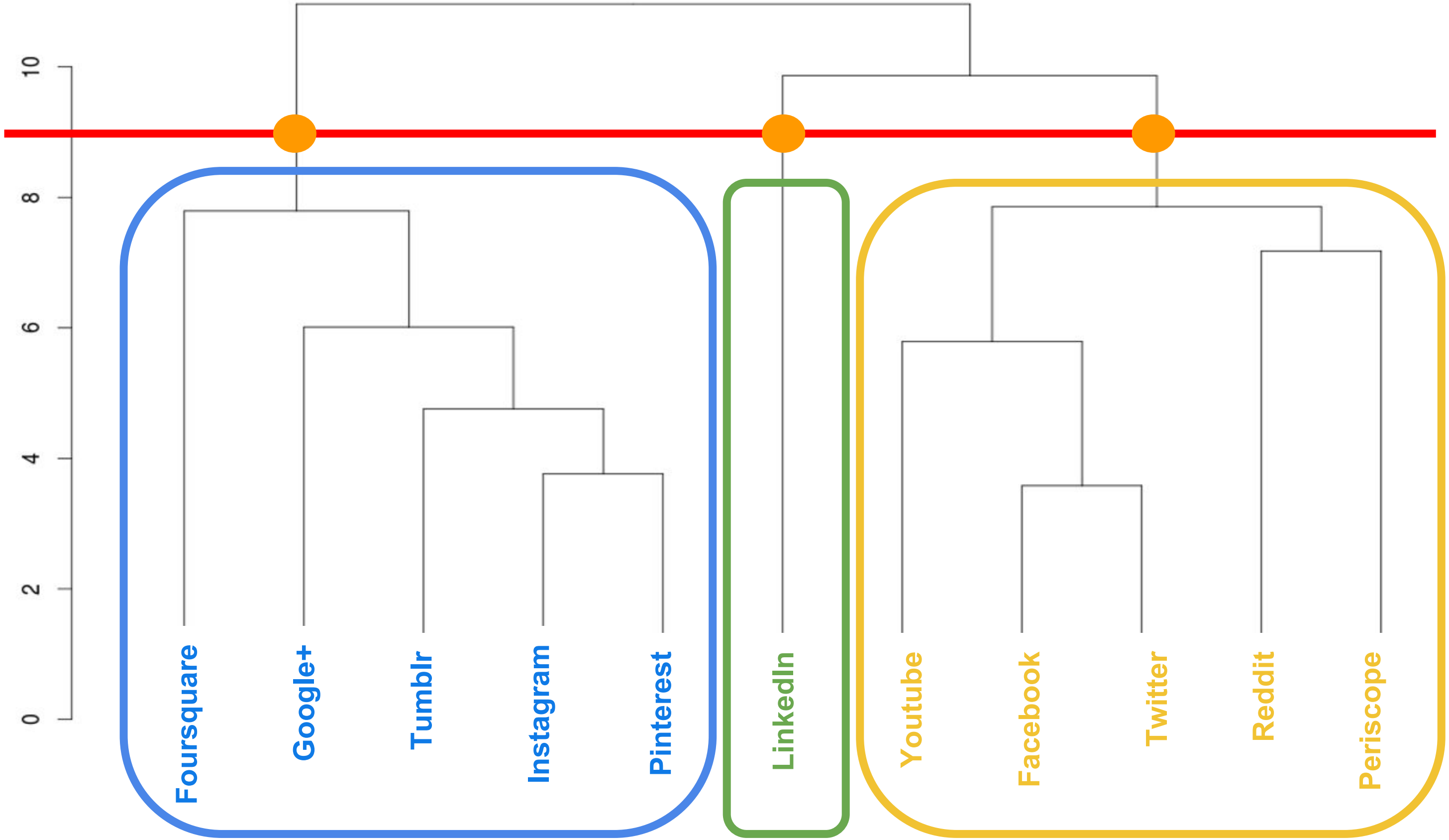
- Pick a height on the y axis and draw a horizontal line there (e.g. y = 9)
- The number of clusters = the number of vertical lines you cross on the dendrogram
- Ignore everything above the line

Social Media Skills



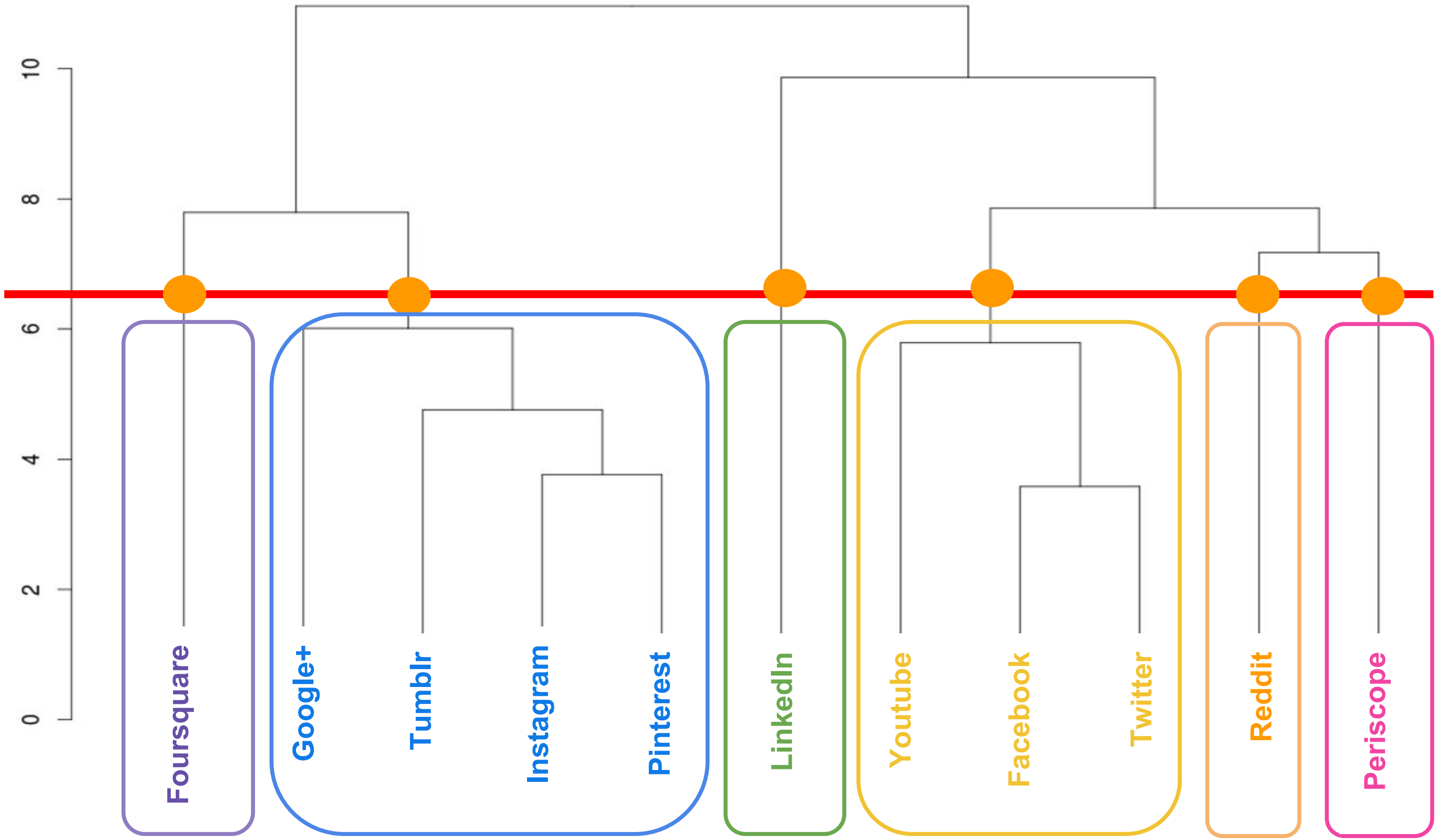
Clipping at y = 9 creates three clusters.

Social Media Skills



Clipping at y = 9 creates three clusters.

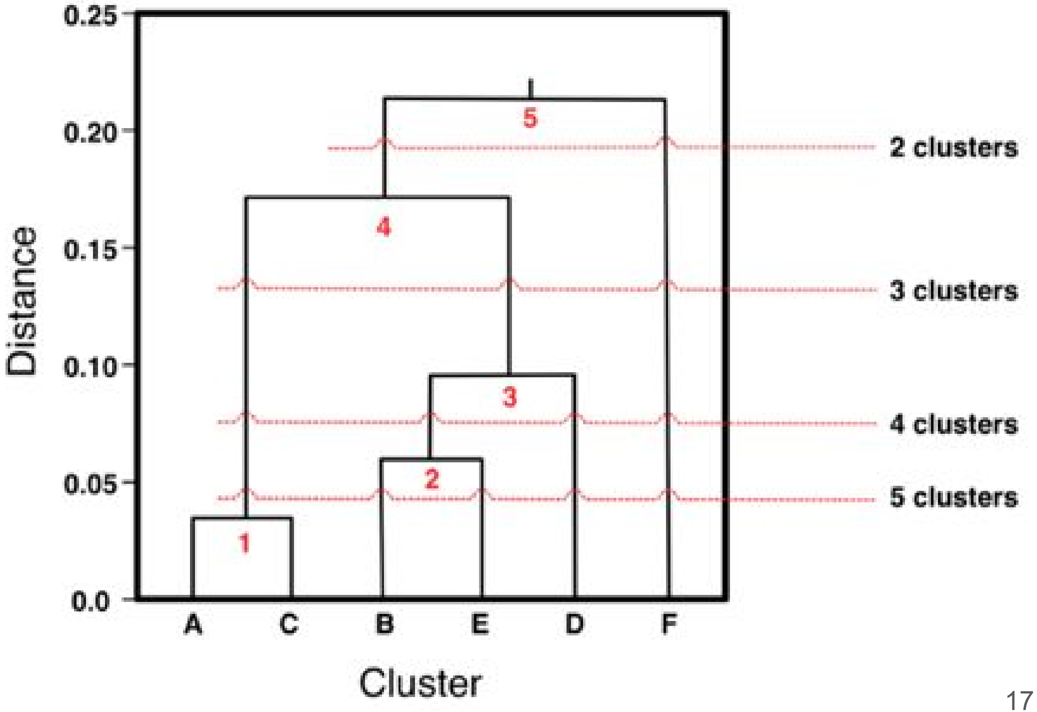
Social Media Skills



Clipping at y = 6.5 creates six clusters.

Choosing Where to Clip:

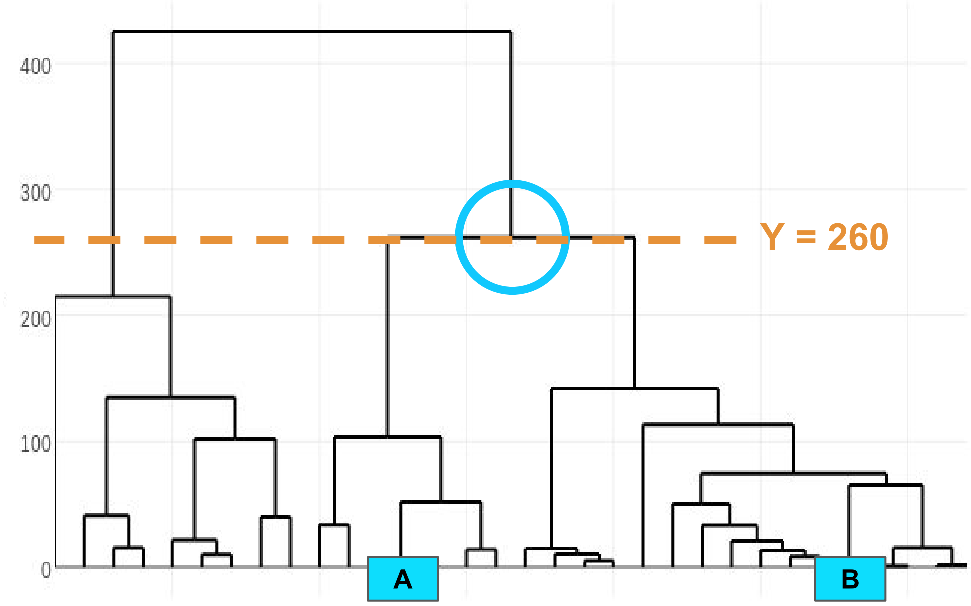
- Preferred number of clusters
- Look for lots of height between junctions
- Business rules



How to Interpret a Dendrogram: Making Comparisons

- Similarity/dissimilarity is indicated by **vertical distance between clusters**

- To find out how different two nodes or clusters are, find where they connect, and look at the y-axis



Distance between A and B is 260

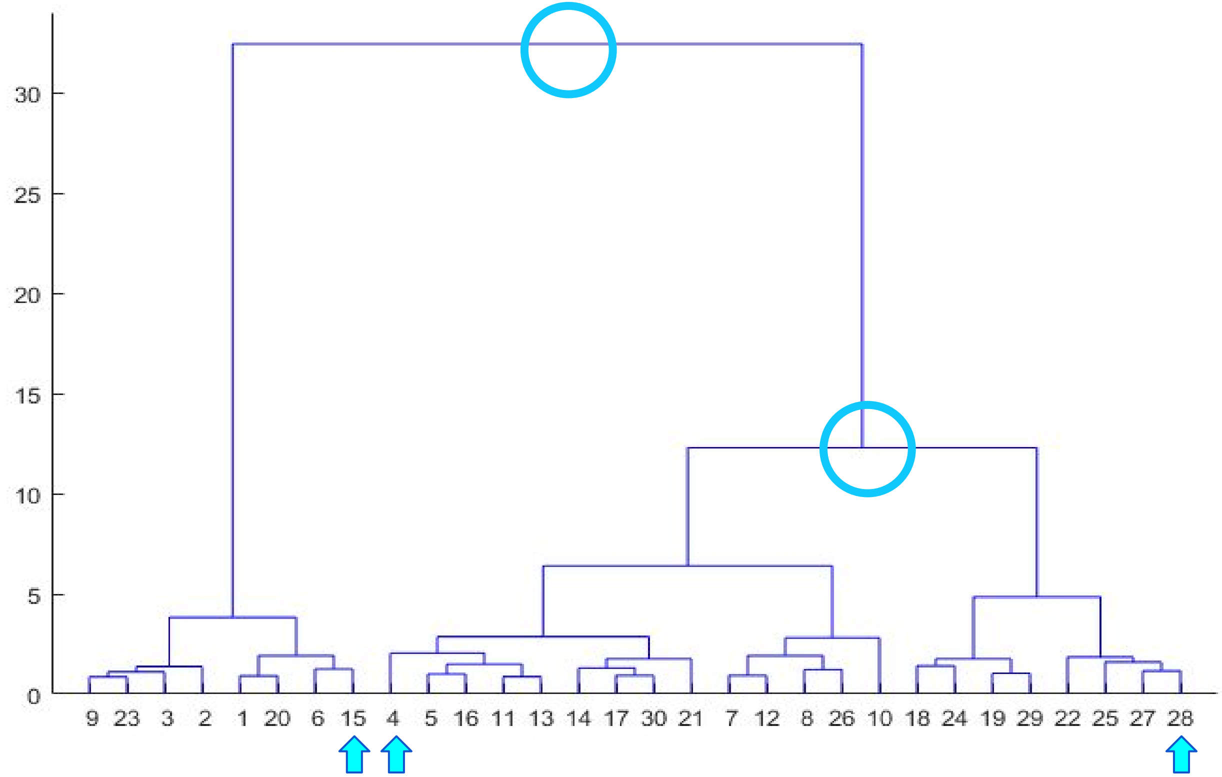
Common Misinterpretations

Remember: distance is only shown by the **y-axis,** not by how far apart nodes are on the x-axis

Q: Which pair is more distant:

15 and 4 or **28 and 4**

A: 15 and 4



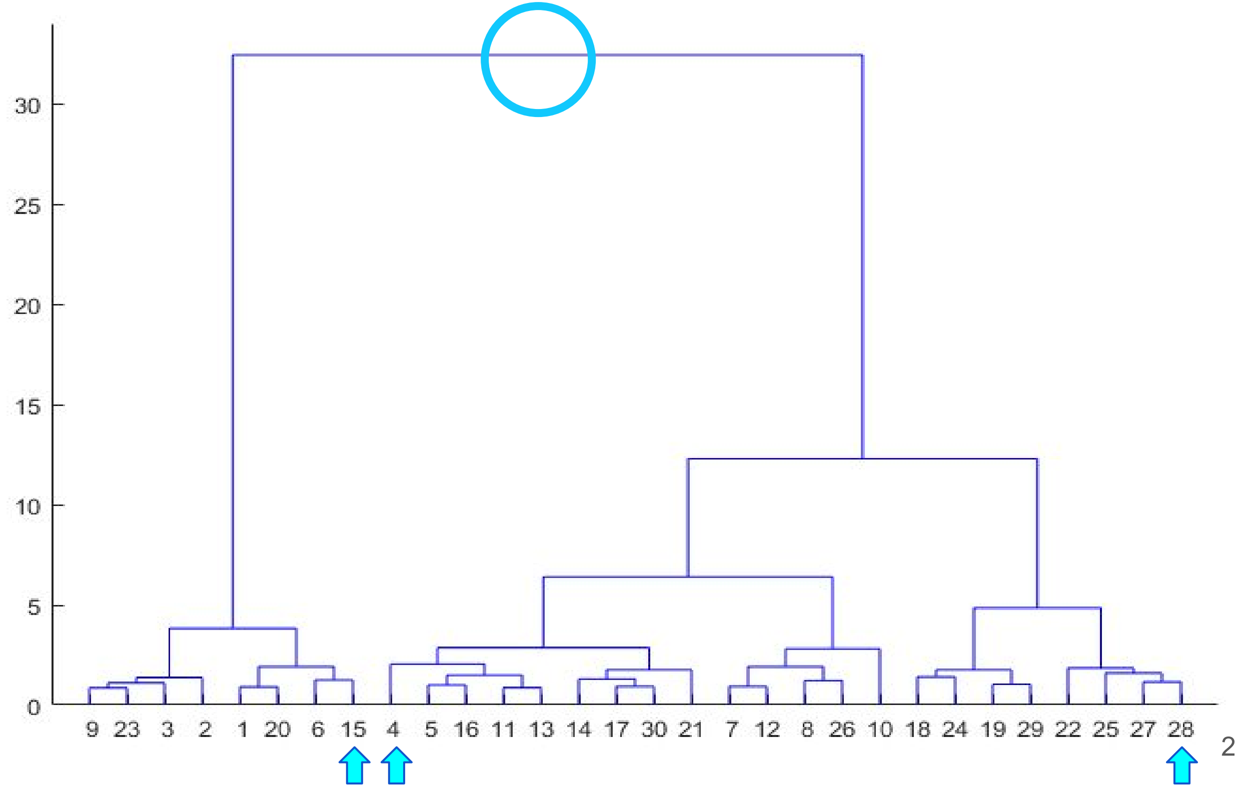
Common Misinterpretations

Remember: distance is only shown by the **y-axis,** not by how far apart nodes are on the x-axis

Q: Which pair is more distant:

15 and 4 or **15 and 28**

A: Neither!



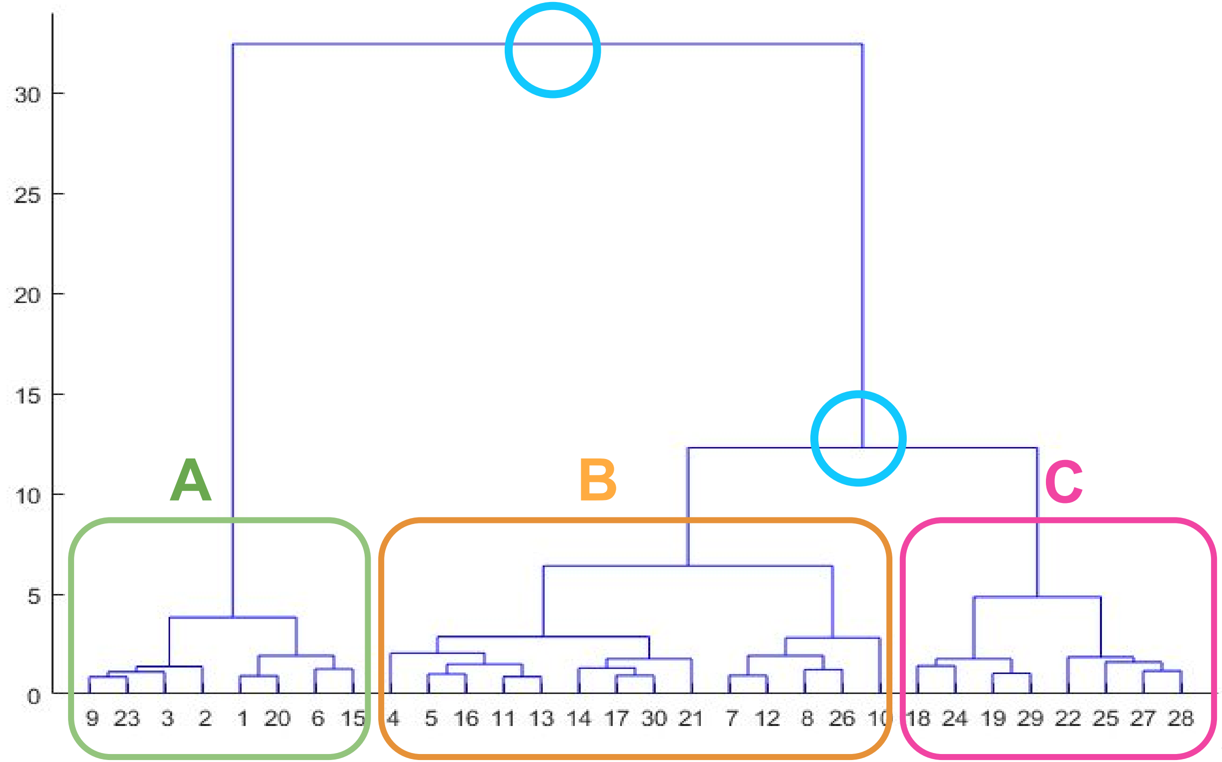
Common Misinterpretations

Remember: distance is only shown by the **y-axis,** not by how far apart nodes are on the x-axis

Q: Which clusters are more distant:

A & B or B & C

A: A & B



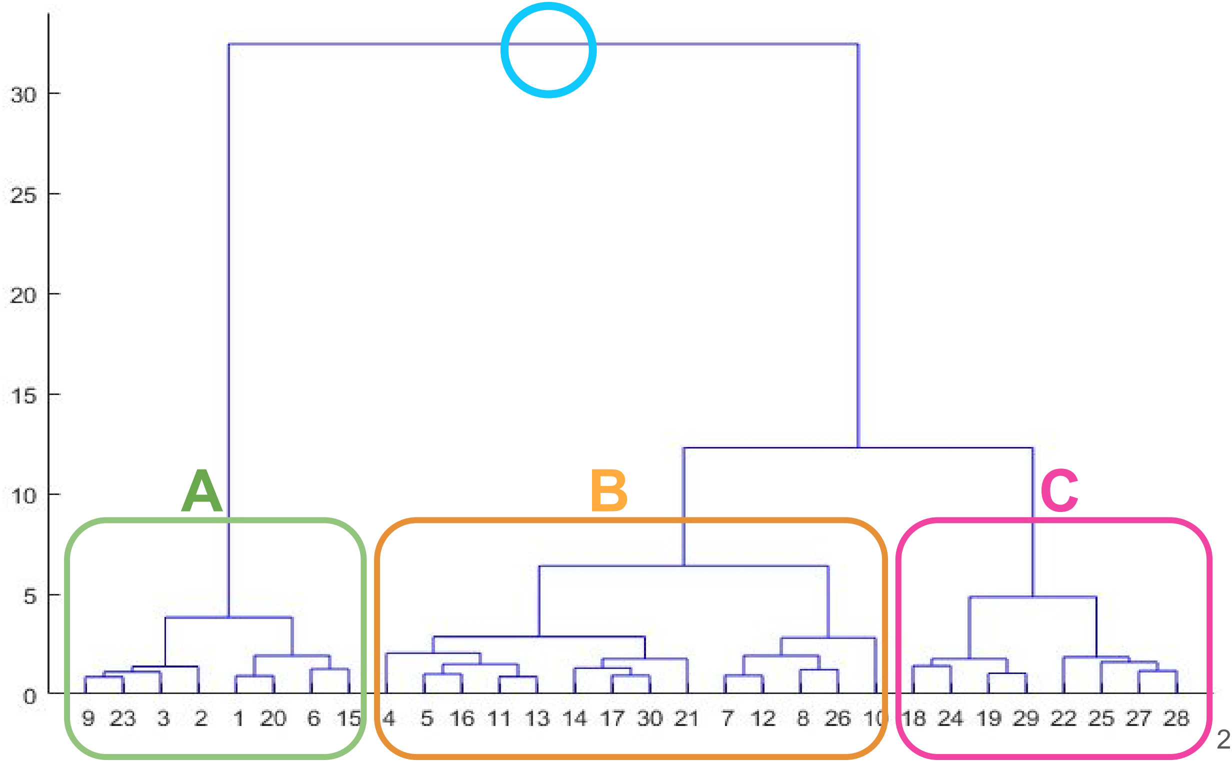
Common Misinterpretations

Remember: distance is only shown by the **y-axis,** not by how far apart nodes are on the x-axis

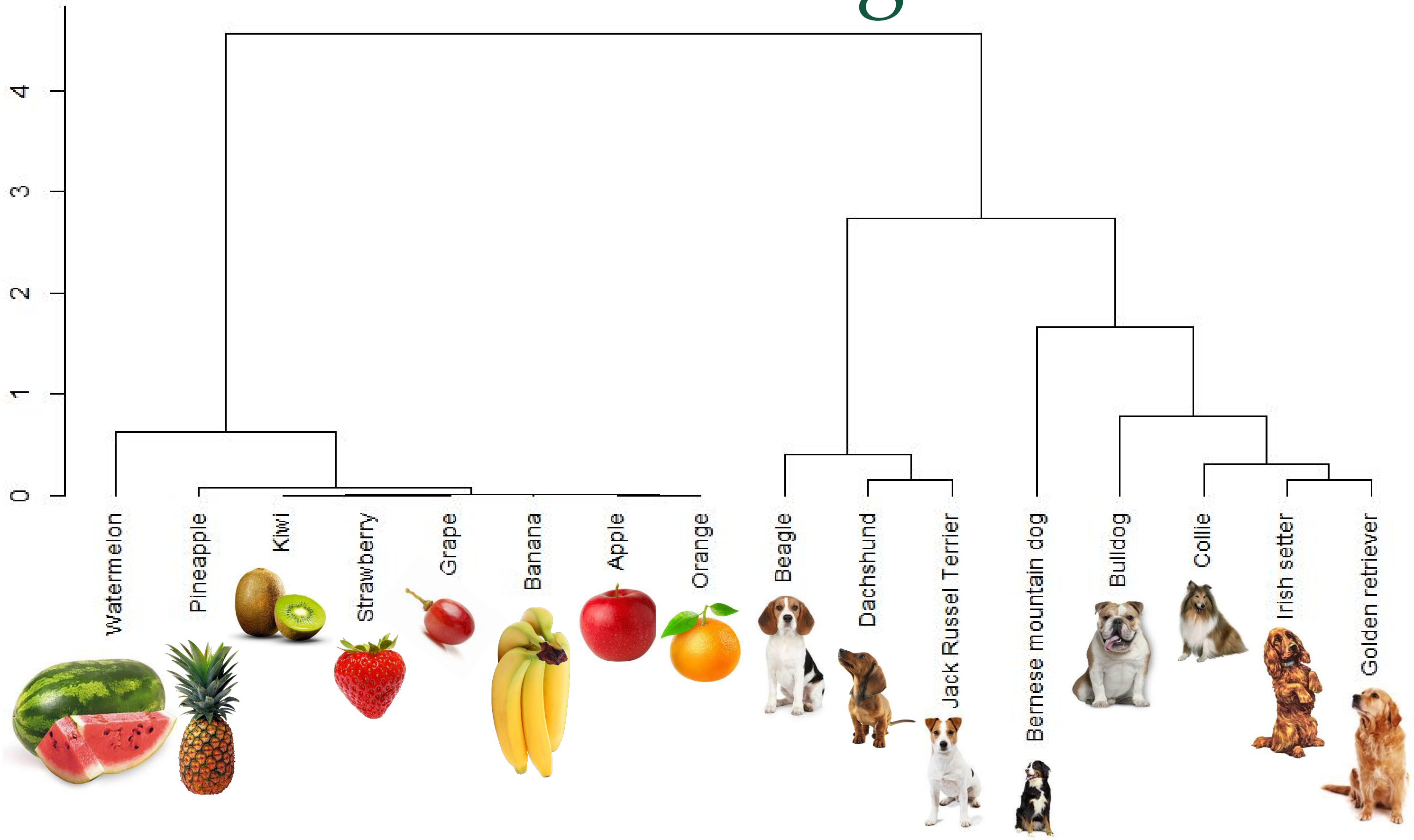
Q: Which clusters are more distant:

A & B or A & C

A: Neither!

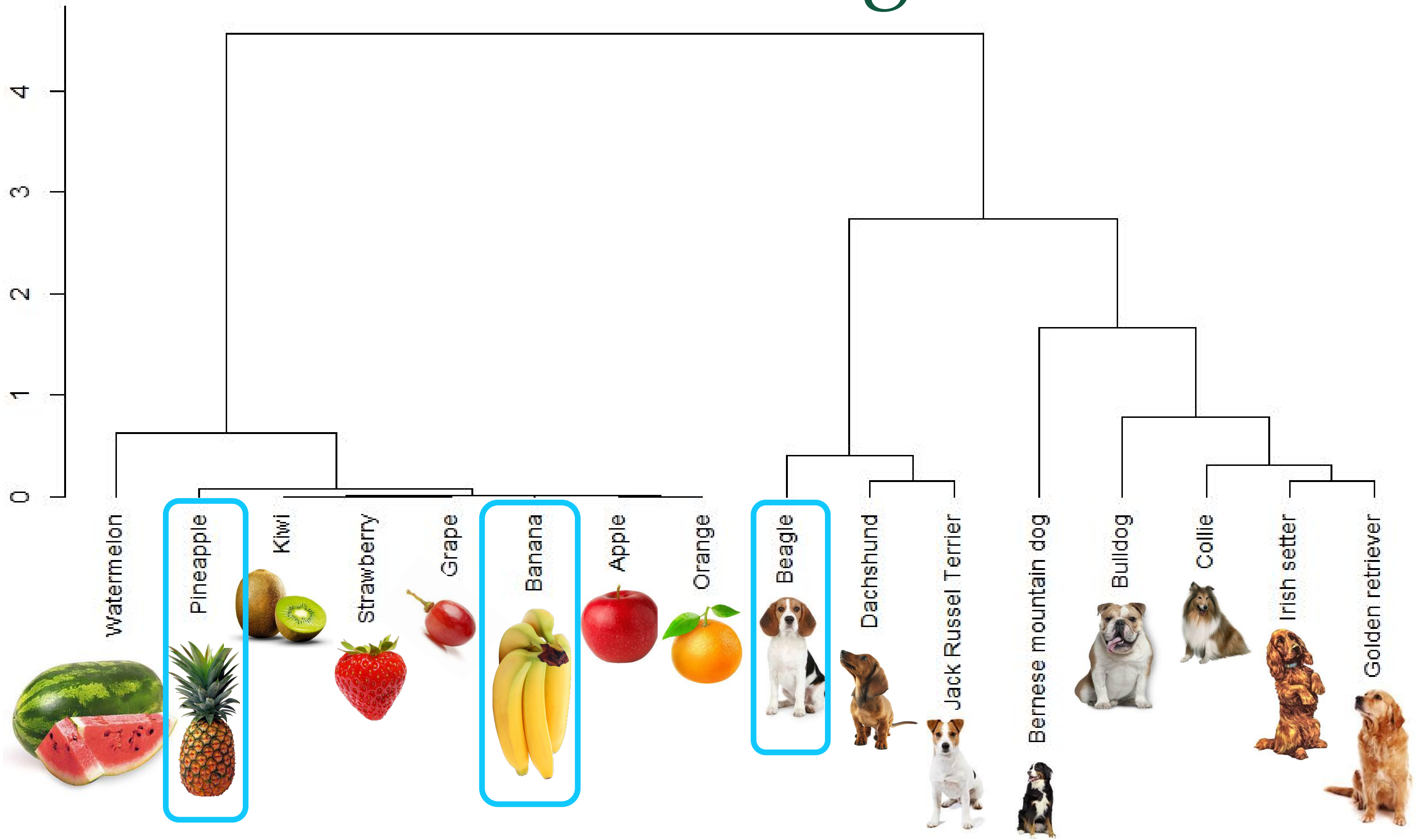


Fruits vs. Dogs



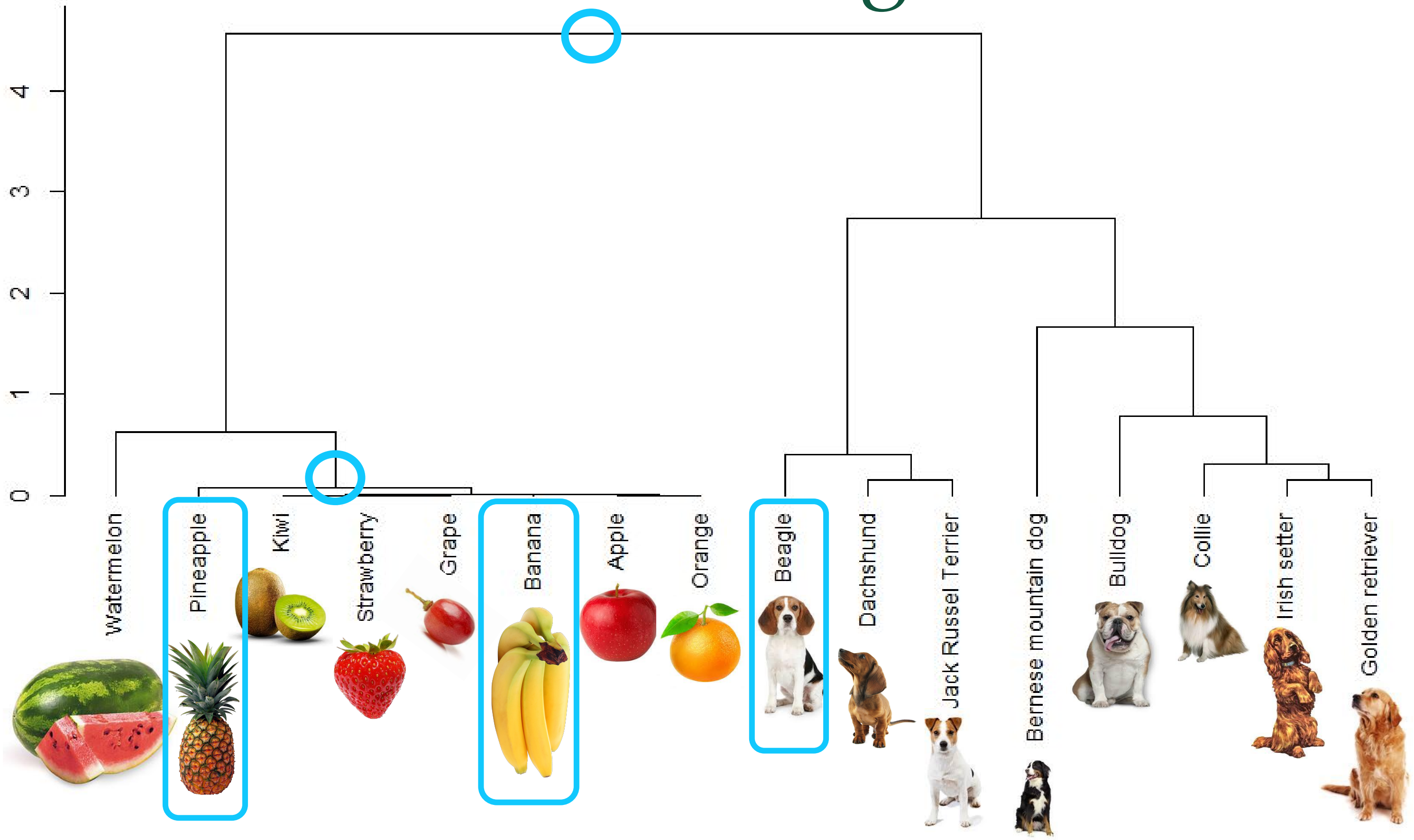
Data: weight, number of legs, number of ears, number of tails

Fruits vs. Dogs



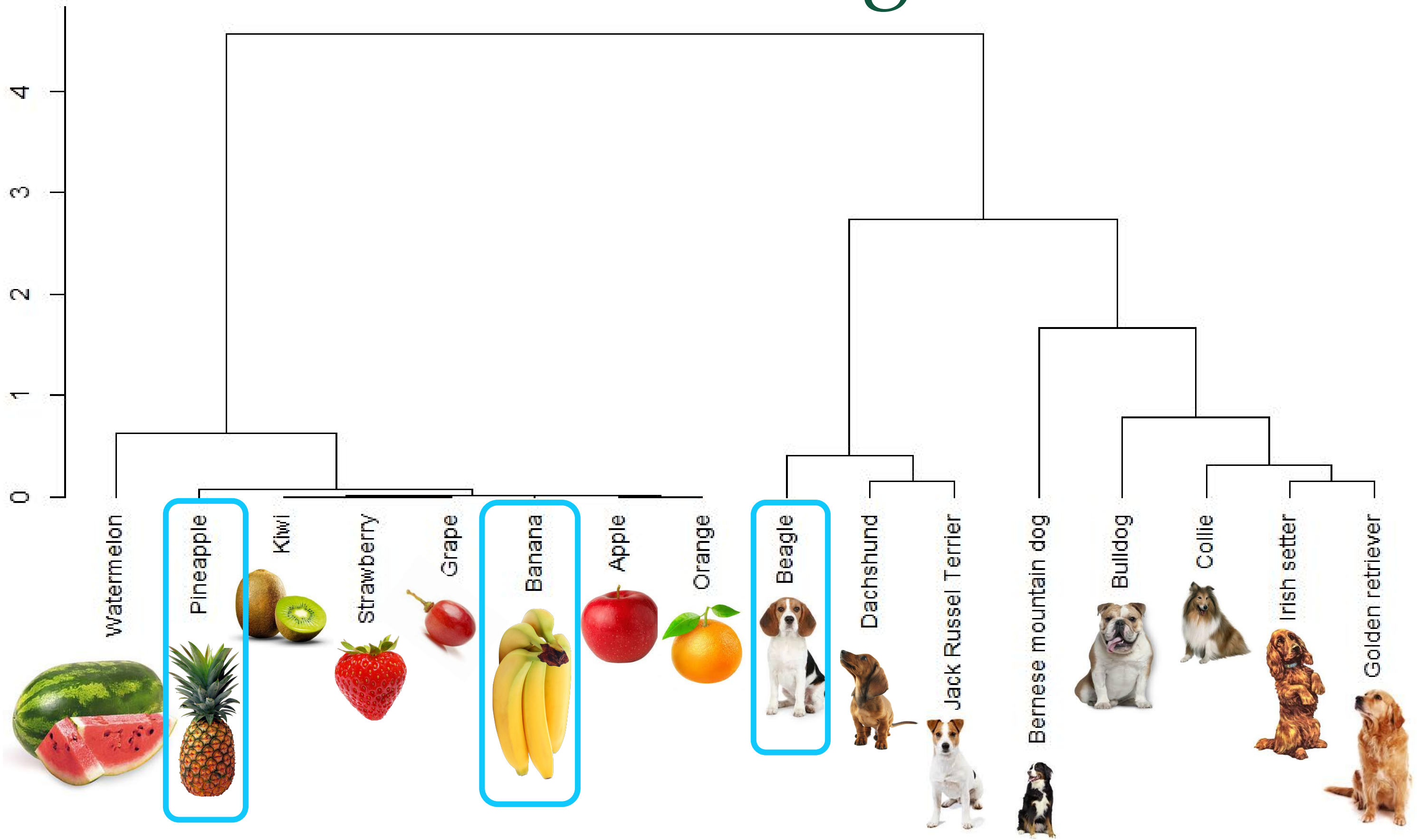
Q: Which pair is more distant: **Banana and Pineapple** or **Banana and Beagle?**

Fruits vs. Dogs



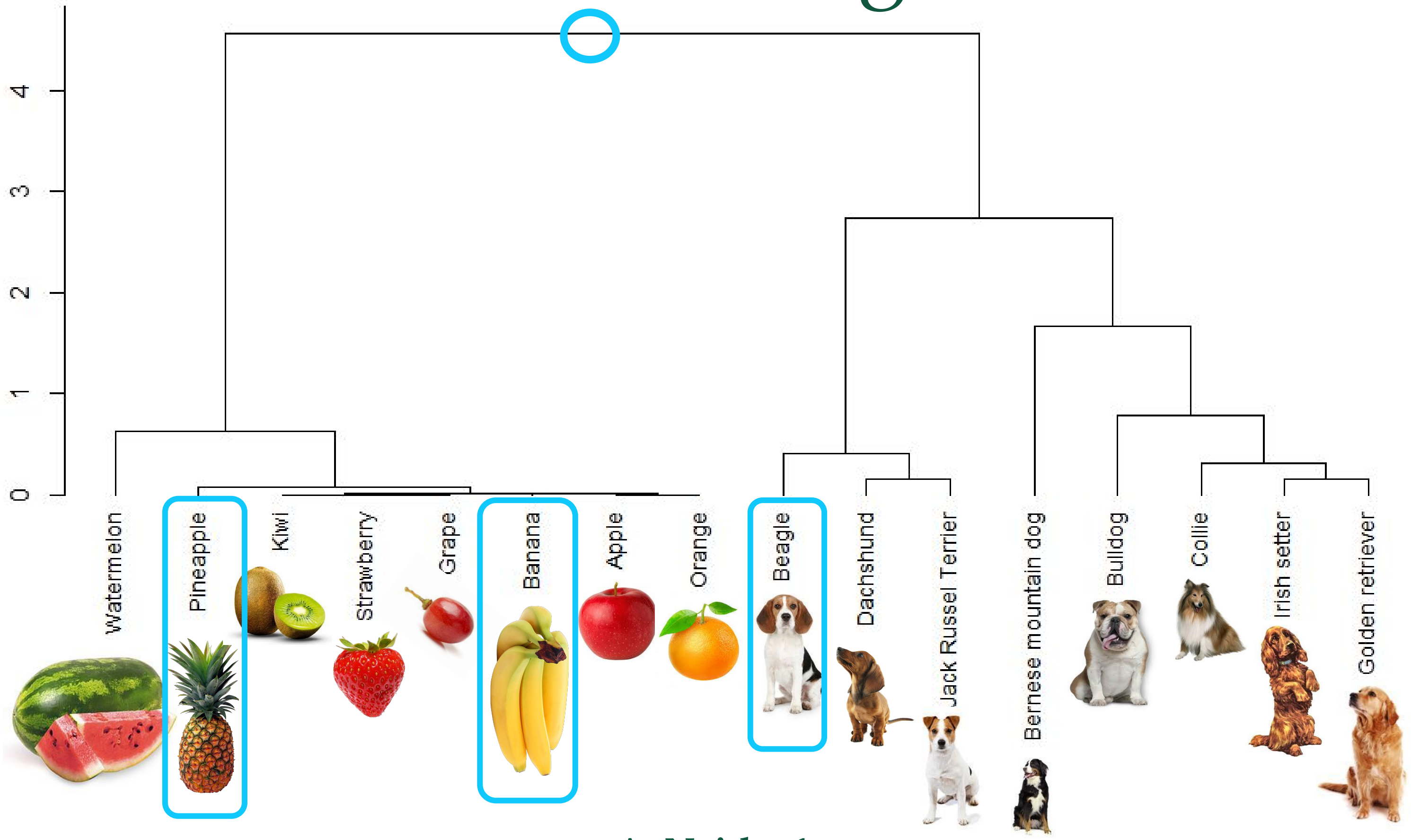
A: Banana and Beagle are more distant

Fruits vs. Dogs



Q: Which pair is more distant: **Banana and Beagle** or **Pineapple and Beagle?**

Fruits vs. Dogs



A: Neither!

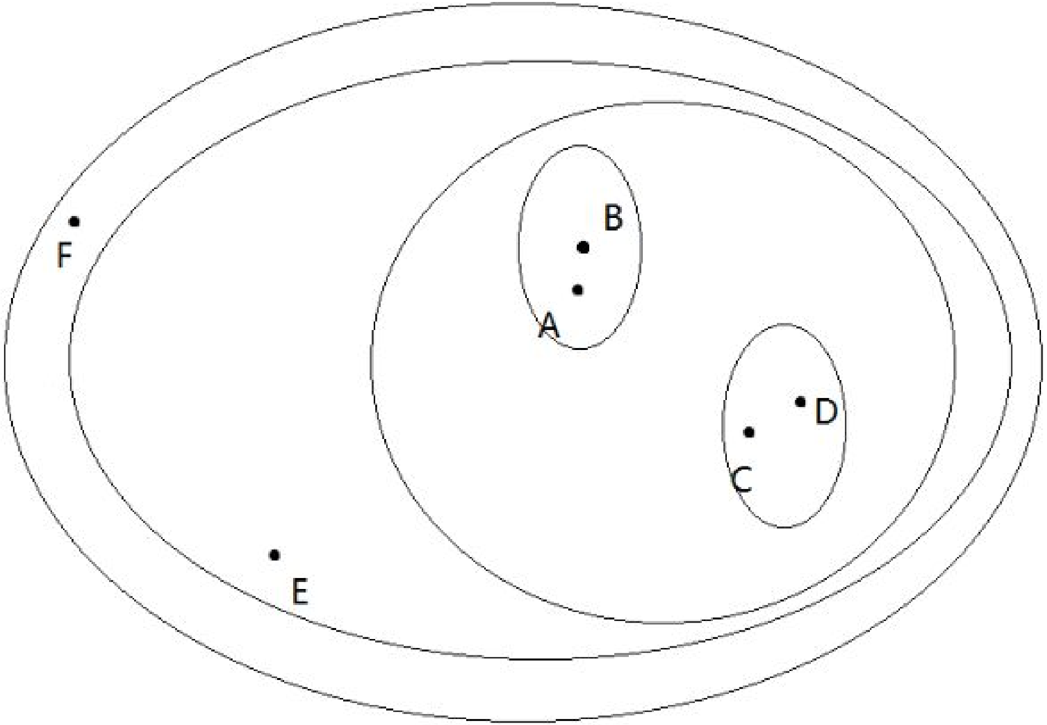
Activity:
Draw-A-Dendrogram Competition

We will use Murray’s Excel flair to randomly select four “volunteers”.

These volunteers will compete to Draw-A-Dendrogram!

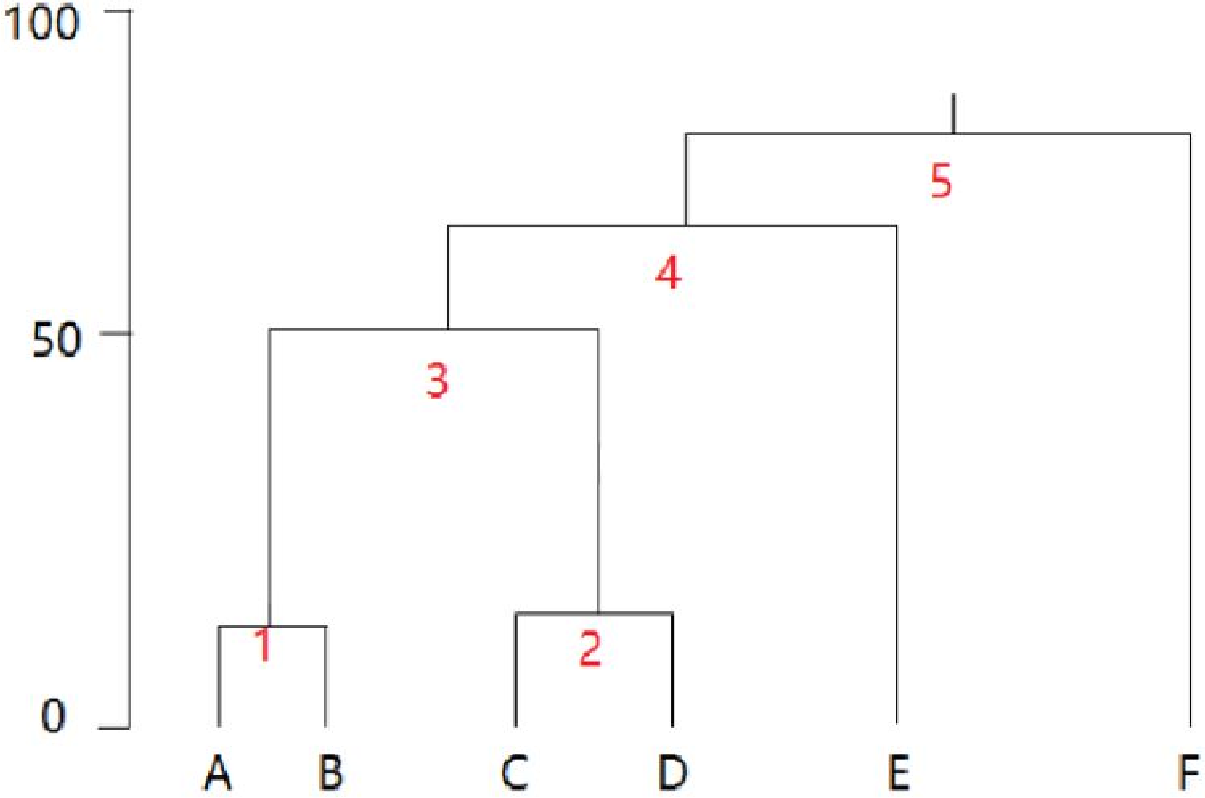
Let’s go!

Draw-A-Dendrogram Competition



Draw-A-Dendrogram Competition

Answer



What are the benefits of a dendrogram?

- Allows the data analyst to see the groupings—the “landscape” of data similarity—before deciding on the number of clusters to extract.

- No need to input k, the number of clusters

- Easy to spot outliers

Drawbacks of a dendrogram

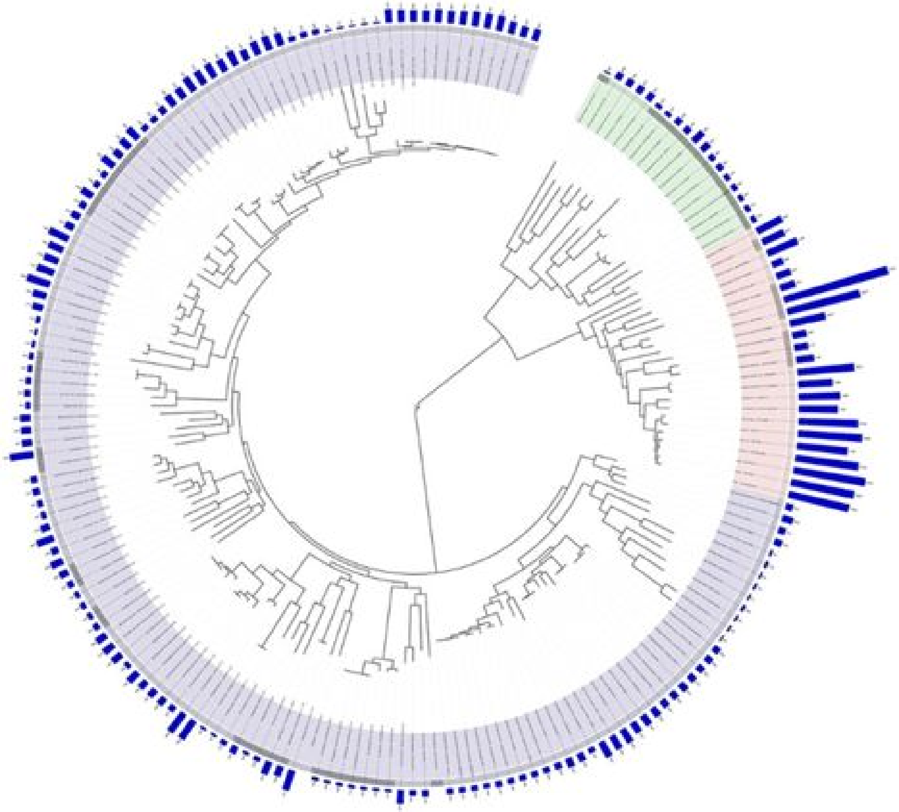
- Structure can vary greatly when using different subsets of data
  - Not always a drawback
  - Run it multiple times with different subsets
  - This can lead to valuable insights

- Computationally complex, so not great for large datasets

Applications: Biology

“Tree of Life”

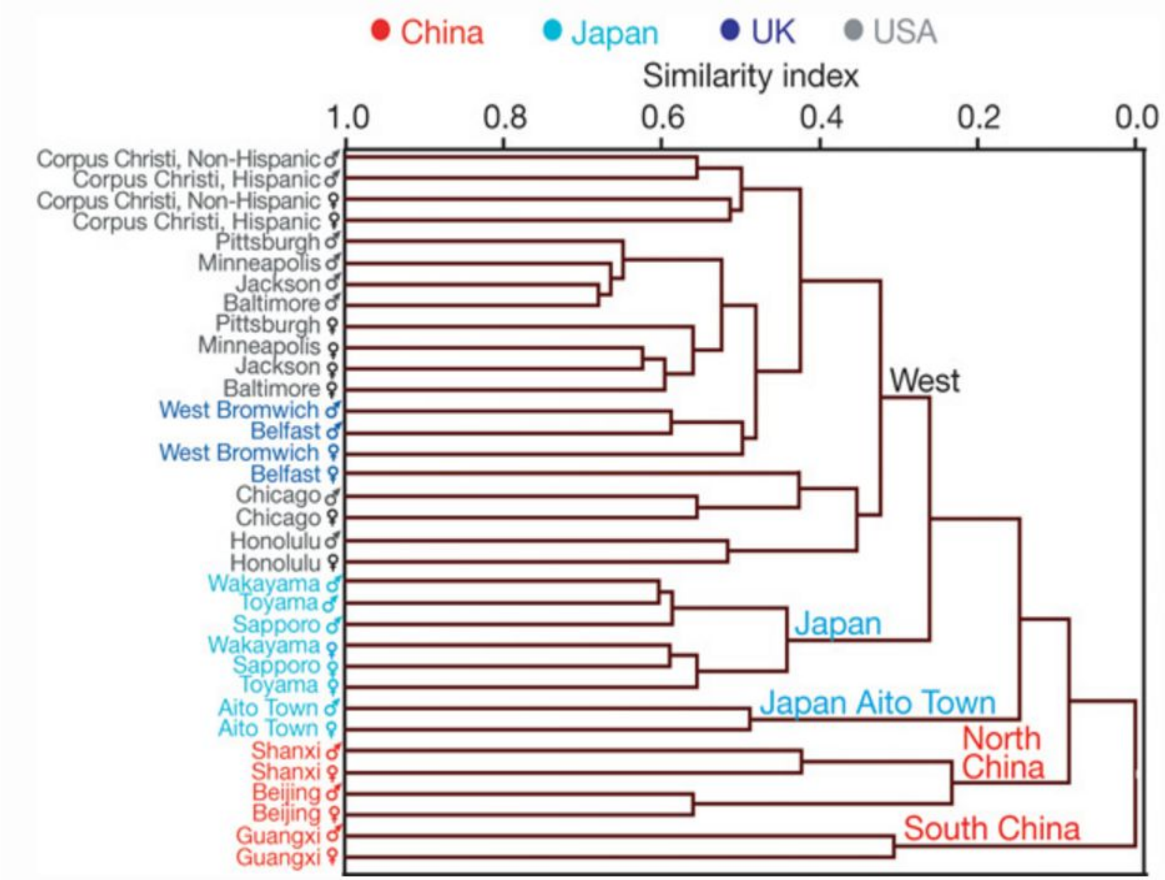
- A hierarchical structure describing the interrelationships of species
- A fundamental concept in systematic biology
- Each node is a different species
- Distance calculates genetic (dis)similarity



Applications: Health

Phenotype Diversity

- Human metabolic phenotype diversity and its association with diet and blood pressure (hierarchical cluster analysis)
- Demonstrates overall similarity/dissimilarity between population samples (urine samples)



Case Study:
Job Categories

Using a dendrogram to make a business decision.

Let’s go!

Case Study: Job Categories

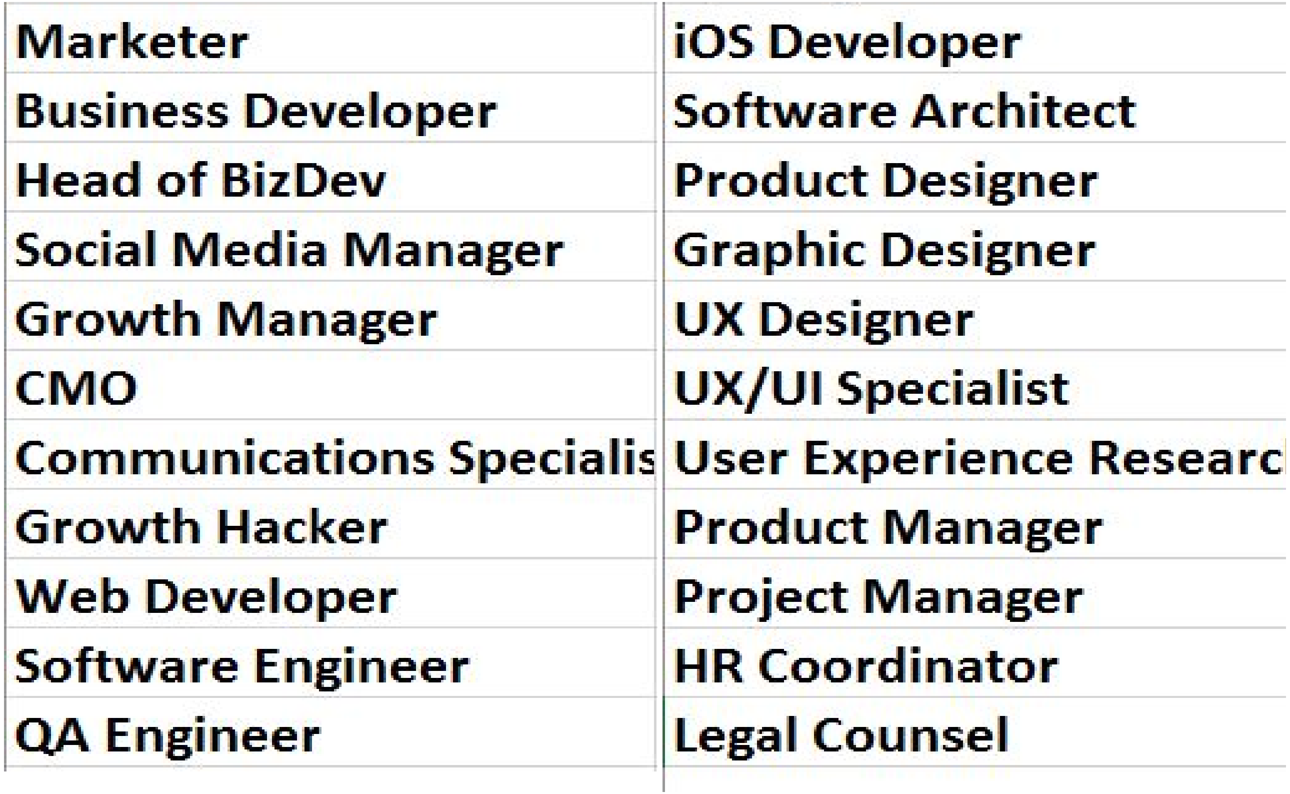
How many different weekly newsletters to send?

- Business & Marketing?
- Software & Engineering?
- Product & Design?
- “Other”?

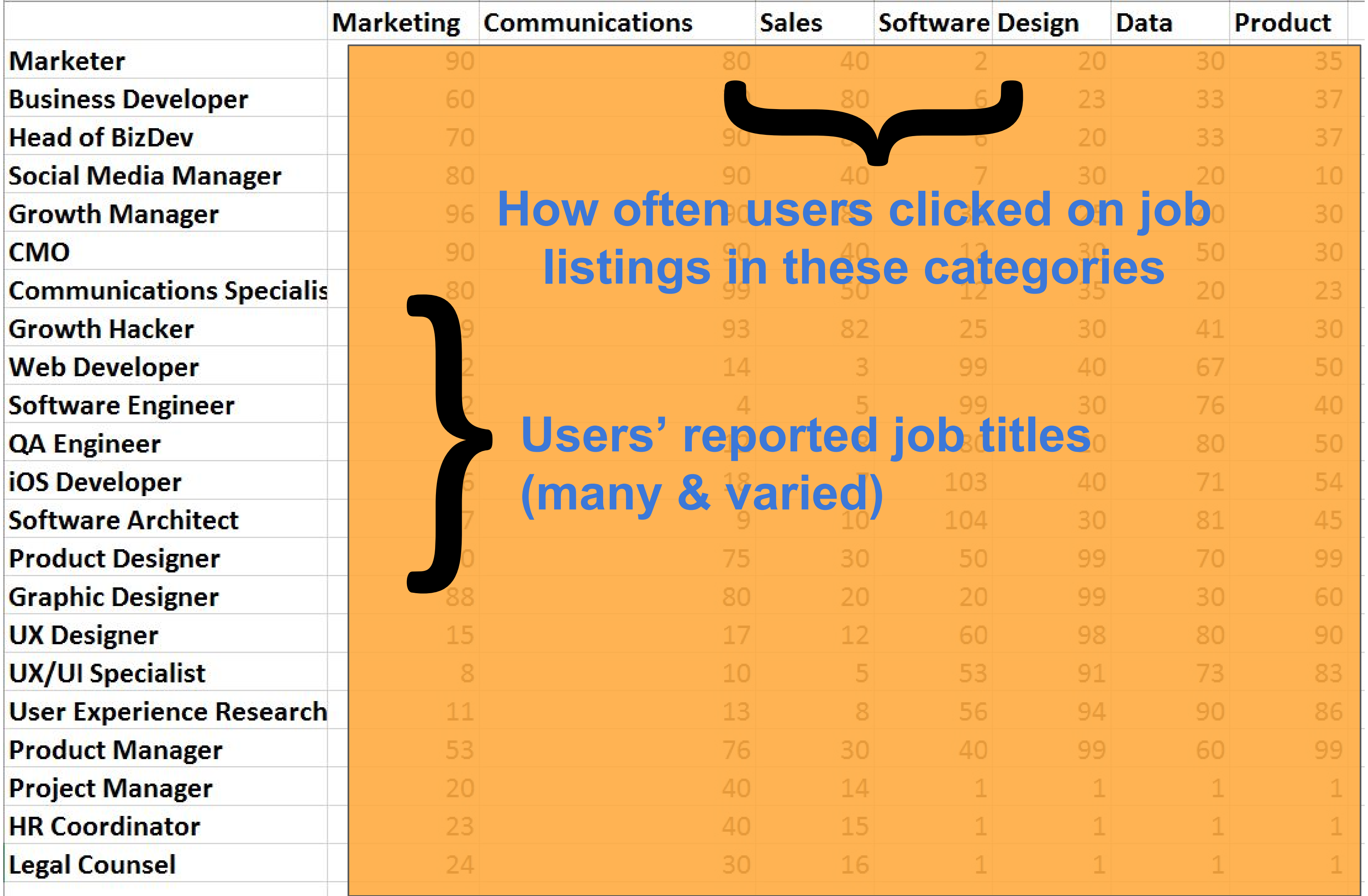
Who to send which newsletter?

Case Study: Job Categories

How many newsletters would you send to these users? Which categories?



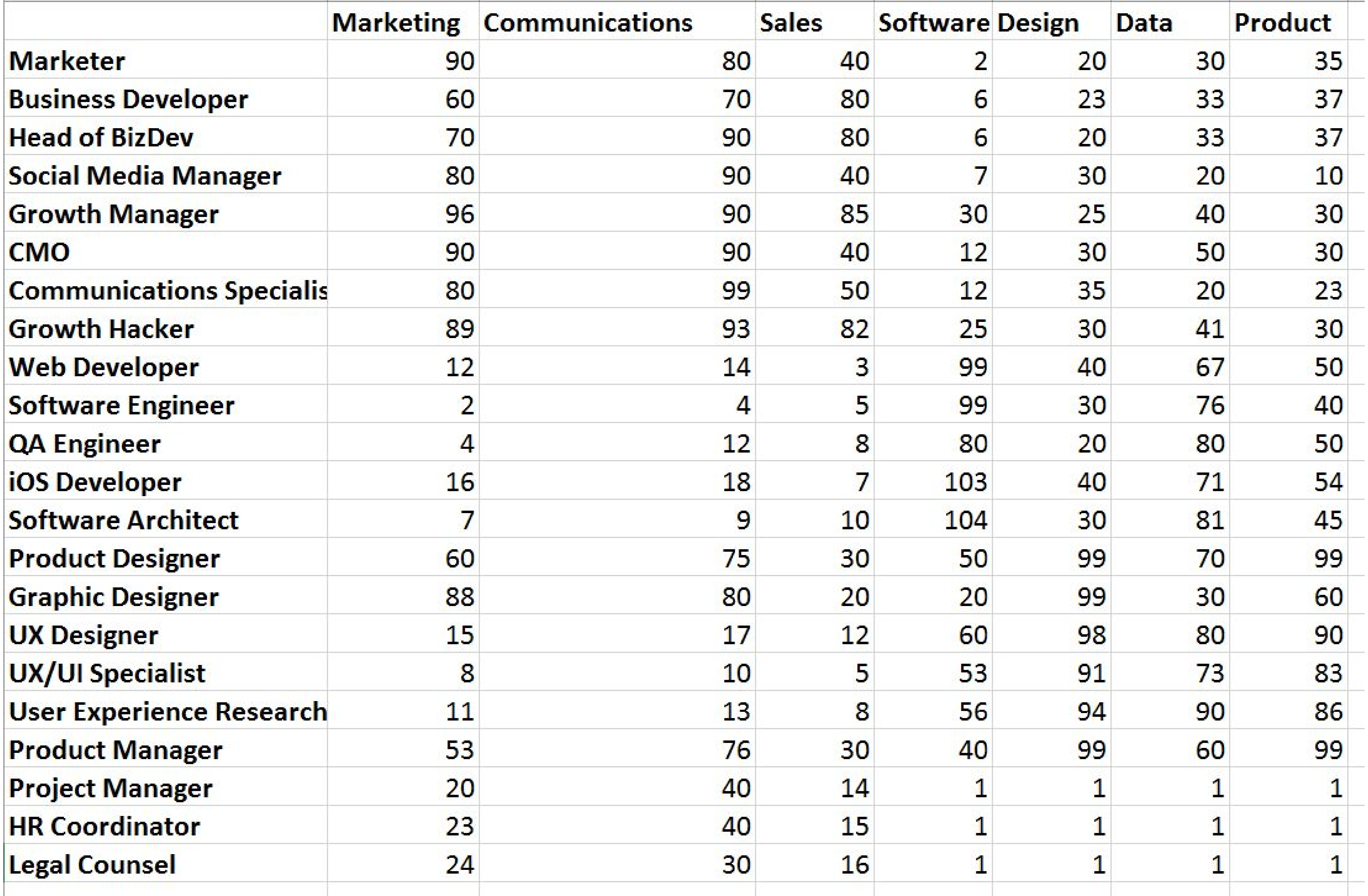
Marketer	iOS Developer
Business Developer	Software Architect
Head of BizDev	Product Designer
Social Media Manager	Graphic Designer
Growth Manager	UX Designer
CMO	UX/UI Specialist
Communications Specialis	User Experience Researc
Growth Hacker	Product Manager
Web Developer	Project Manager
Software Engineer	HR Coordinator
QA Engineer	Legal Counsel



How often users clicked on job listings in these categories

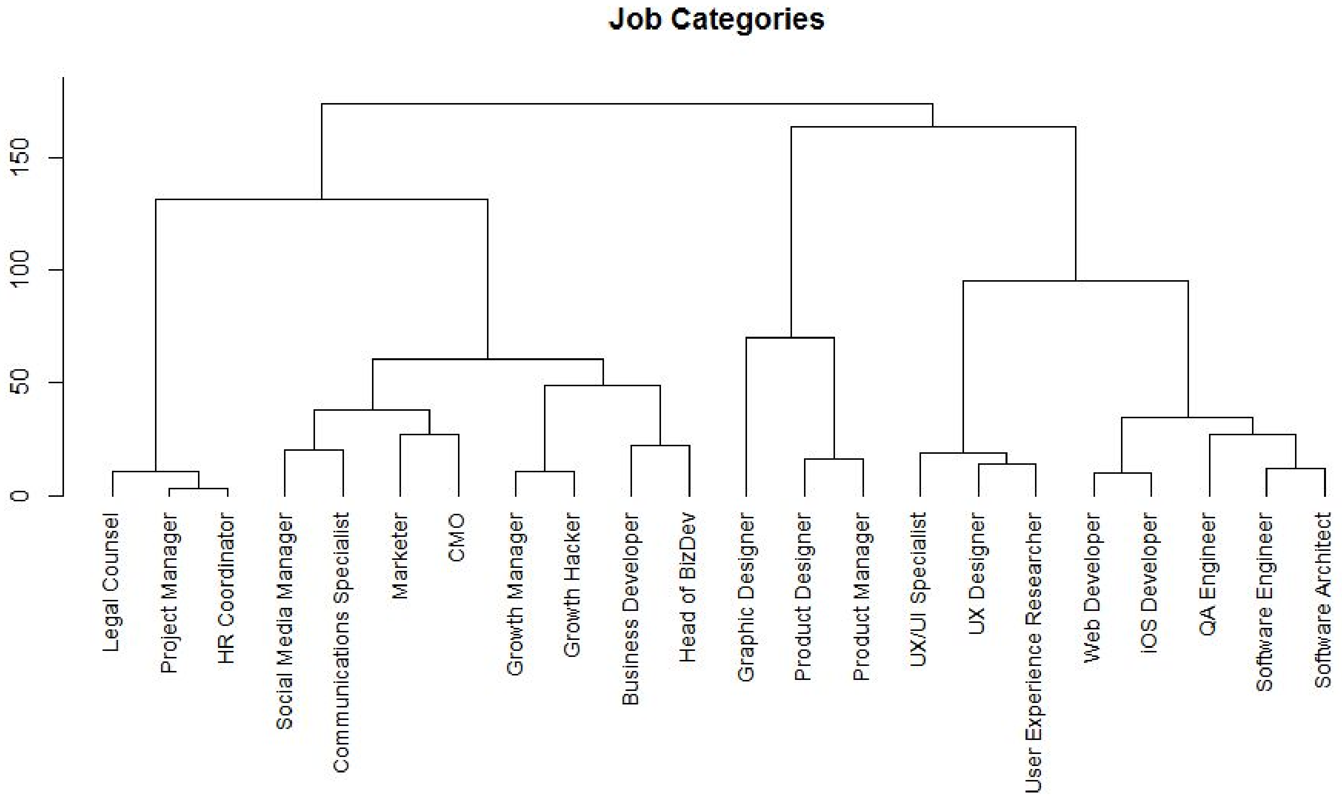
Users’ reported job titles (many & varied)

	Marketing	Communications	Sales	Software	Design	Data	Product
Marketer	90	80	40	2	20	30	35
Business Developer	60	[illegible]	80	6	23	33	37
Head of BizDev	70	90	[illegible]	6	20	33	37
Social Media Manager	80	90	40	7	30	20	10
Growth Manager	96	90	[illegible]	[illegible]	[illegible]	[illegible]	30
CMO	90	90	40	12	30	50	30
Communications Specialis	80	99	50	12	35	20	23
Growth Hacker	[illegible]	93	82	25	30	41	30
Web Developer	[illegible]	14	3	99	40	67	50
Software Engineer	[illegible]	4	5	99	30	76	40
QA Engineer	[illegible]	[illegible]	[illegible]	80	[illegible]	80	50
iOS Developer	[illegible]	18	7	103	40	71	54
Software Architect	[illegible]	9	10	104	30	81	45
Product Designer	[illegible]	75	30	50	99	70	99
Graphic Designer	88	80	20	20	99	30	60
UX Designer	15	17	12	60	98	80	90
UX/UI Specialist	8	10	5	53	91	73	83
User Experience Research	11	13	8	56	94	90	86
Product Manager	53	76	30	40	99	60	99
Project Manager	20	40	14	1	1	1	1
HR Coordinator	23	40	15	1	1	1	1
Legal Counsel	24	30	16	1	1	1	1

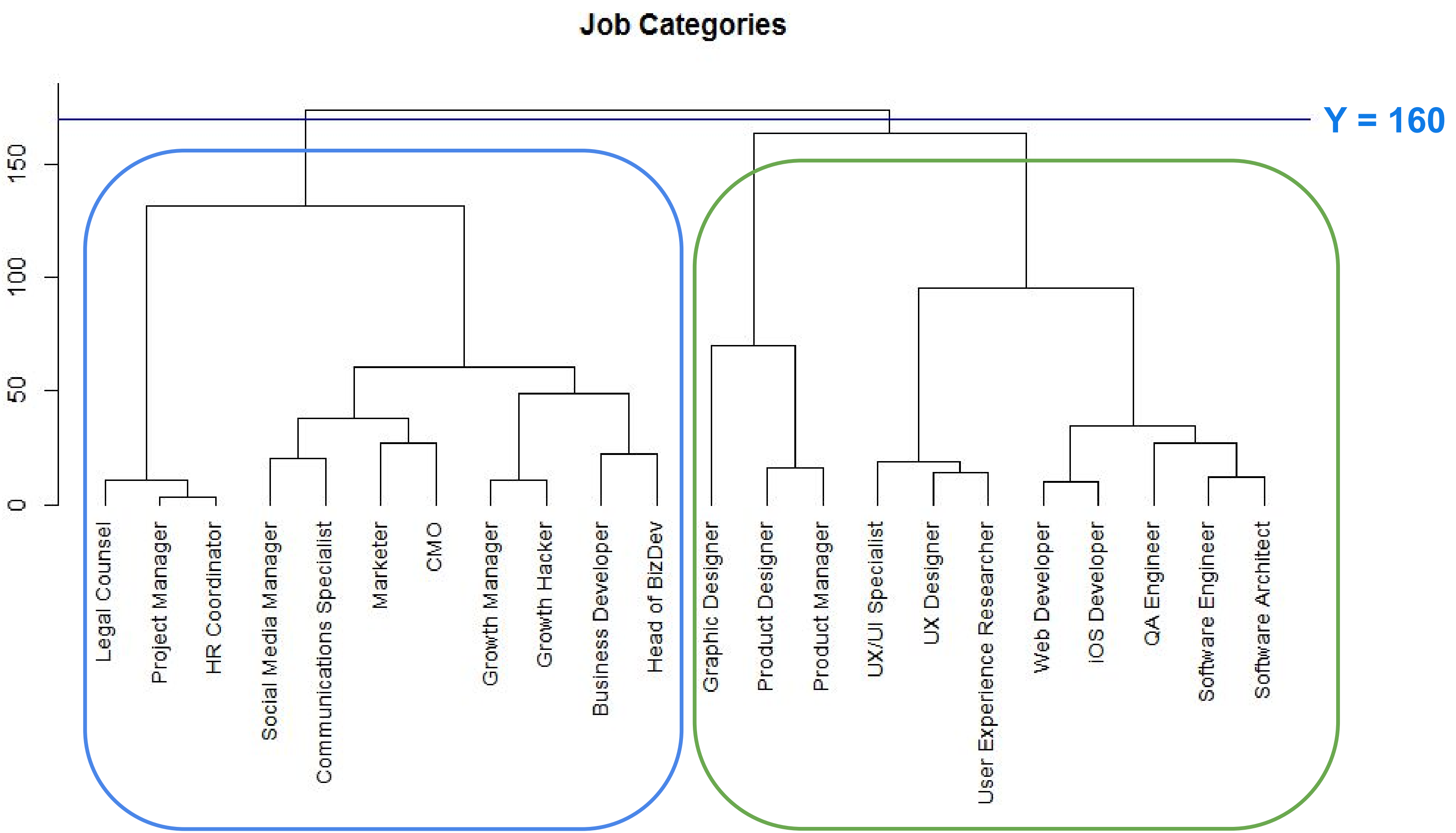


	Marketing	Communications	Sales	Software	Design	Data	Product
Marketer	90	80	40	2	20	30	35
Business Developer	60	70	80	6	23	33	37
Head of BizDev	70	90	80	6	20	33	37
Social Media Manager	80	90	40	7	30	20	10
Growth Manager	96	90	85	30	25	40	30
CMO	90	90	40	12	30	50	30
Communications Specialis	80	99	50	12	35	20	23
Growth Hacker	89	93	82	25	30	41	30
Web Developer	12	14	3	99	40	67	50
Software Engineer	2	4	5	99	30	76	40
QA Engineer	4	12	8	80	20	80	50
iOS Developer	16	18	7	103	40	71	54
Software Architect	7	9	10	104	30	81	45
Product Designer	60	75	30	50	99	70	99
Graphic Designer	88	80	20	20	99	30	60
UX Designer	15	17	12	60	98	80	90
UX/UI Specialist	8	10	5	53	91	73	83
User Experience Research	11	13	8	56	94	90	86
Product Manager	53	76	30	40	99	60	99
Project Manager	20	40	14	1	1	1	1
HR Coordinator	23	40	15	1	1	1	1
Legal Counsel	24	30	16	1	1	1	1

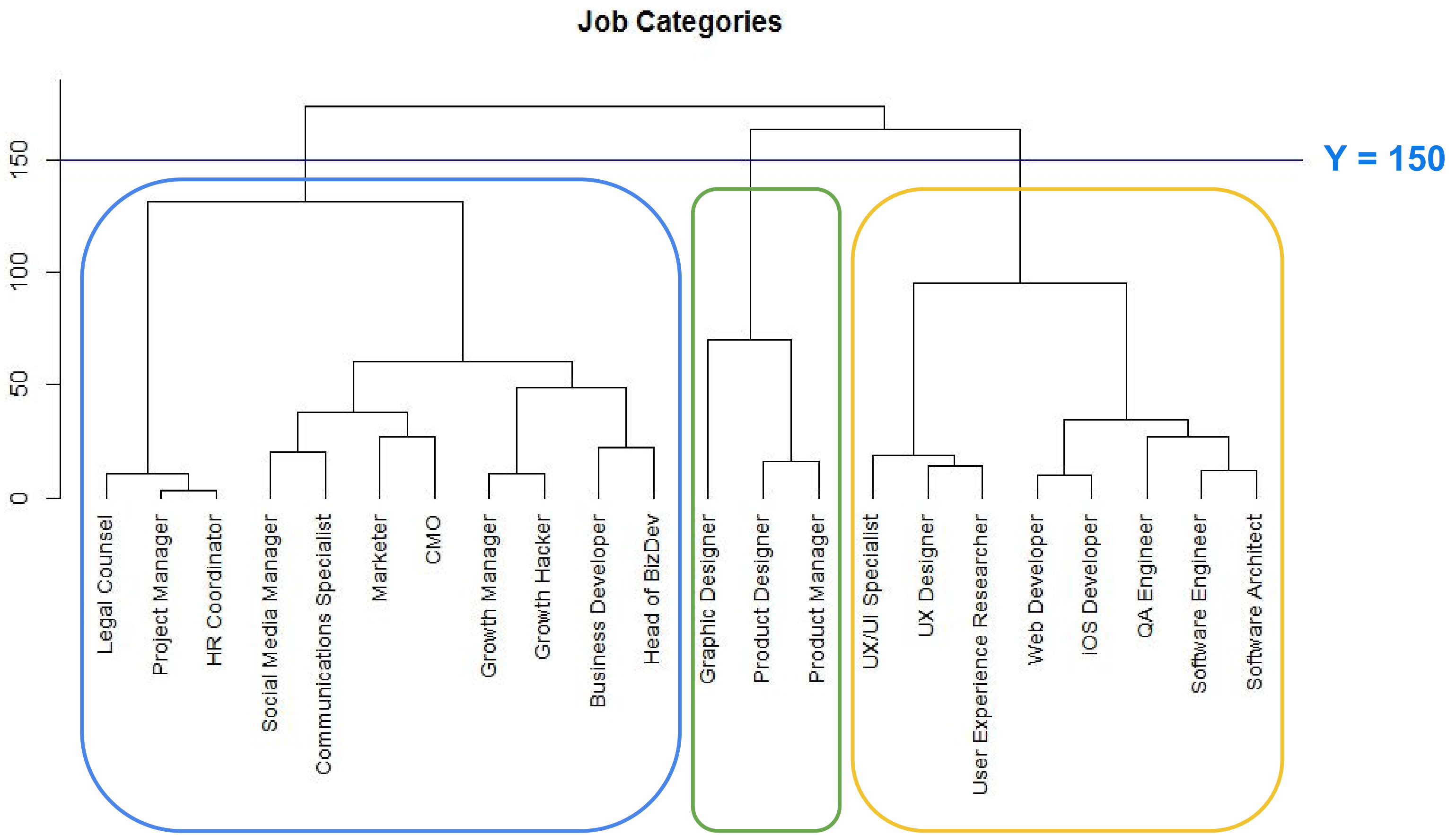
Case Study: Job Categories



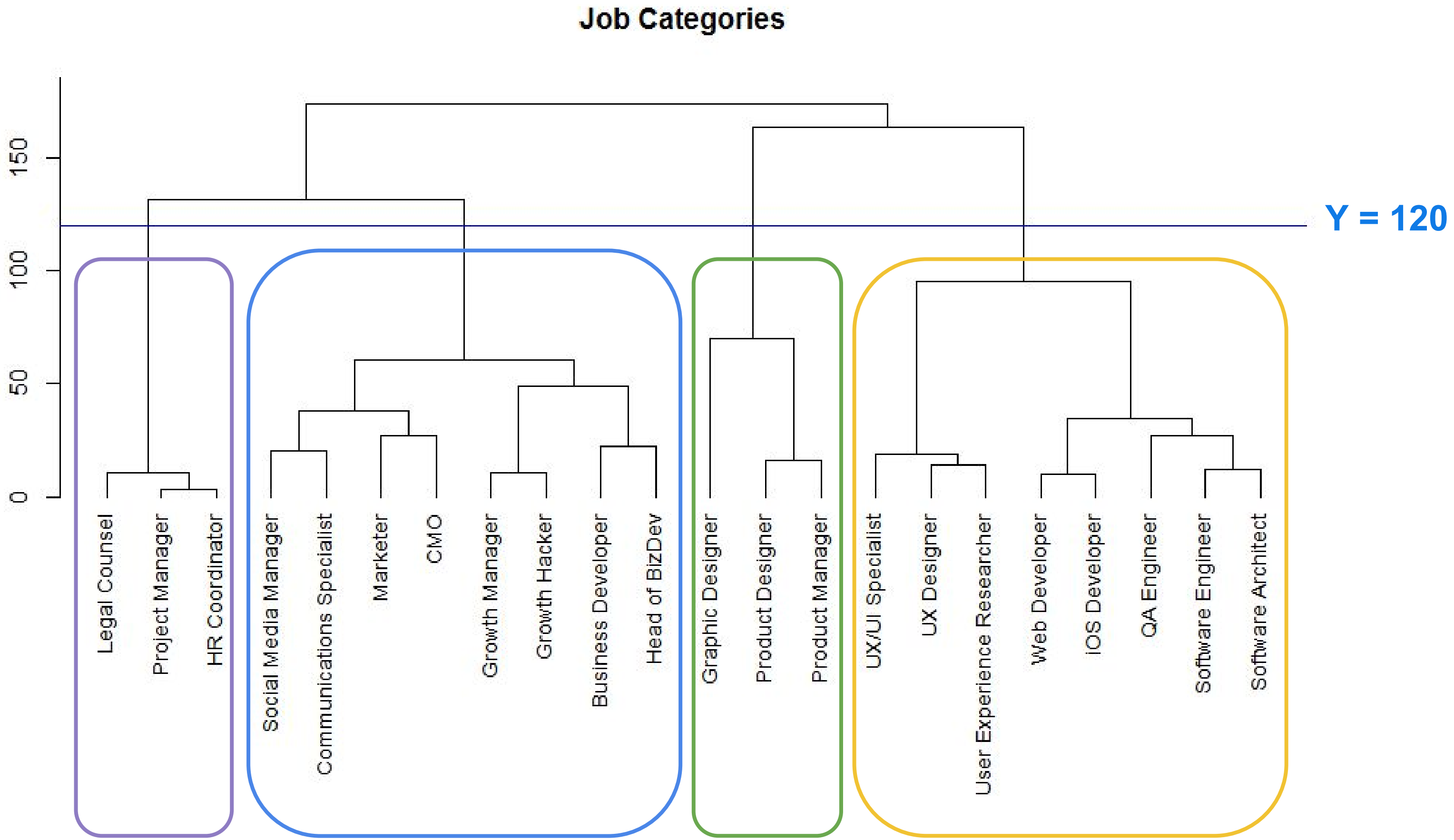
Case Study: Job Categories



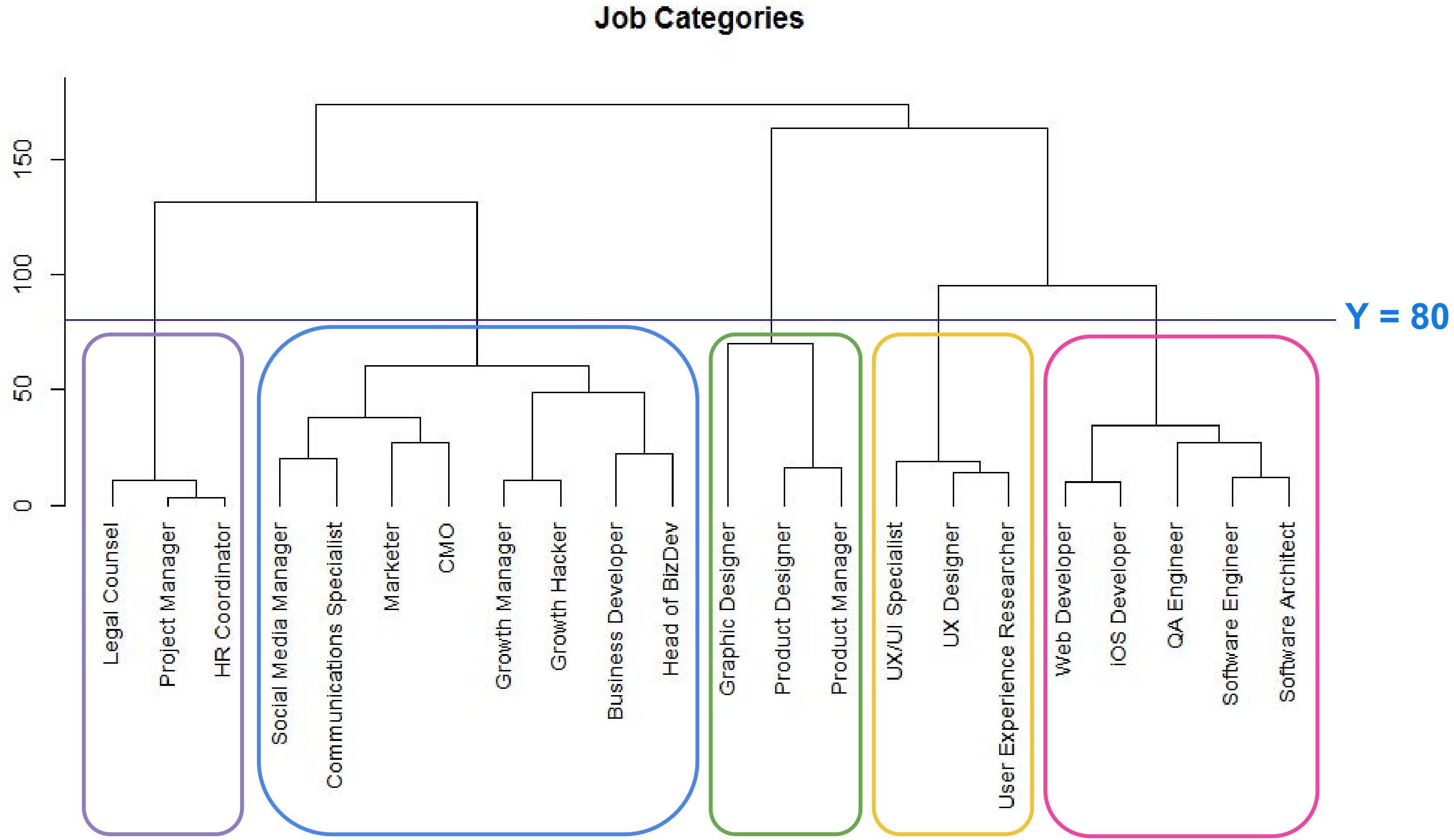
Case Study: Job Categories



Case Study: Job Categories



Case Study: Job Categories



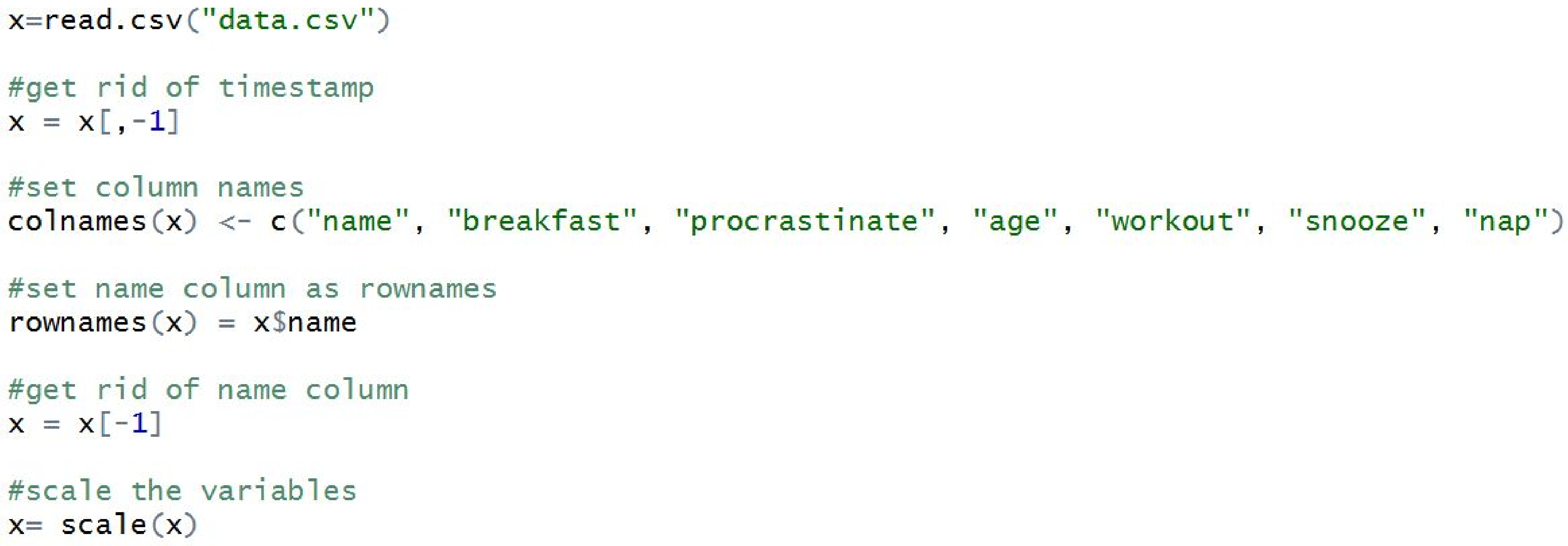
Activity:
Class Generated Dendrogram

You have two minutes to complete a quick survey.

Check Slack or your W&M email for the link.

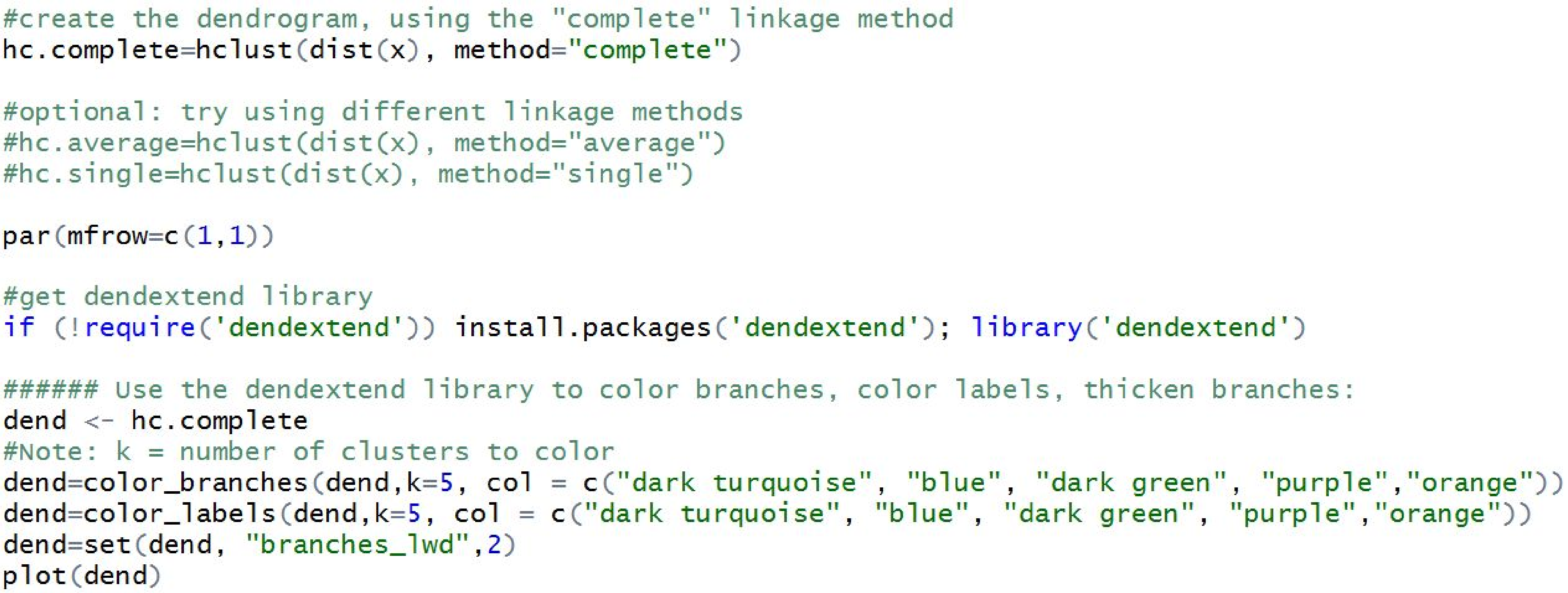
Go!

Code: Class Generated Data Part 1



```
x=read.csv("data.csv")

#get rid of timestamp
x = x[,-1]

#set column names
colnames(x) <- c("name", "breakfast", "procrastinate", "age", "workout", "snooze", "nap")

#set name column as rownames
rownames(x) = x$name

#get rid of name column
x = x[-1]

#scale the variables
x= scale(x)
```

Code: Class Generated Data Part 2



```
#create the dendrogram, using the "complete" linkage method
hc.complete=hclust(dist(x), method="complete")

#optional: try using different linkage methods
#hc.average=hclust(dist(x), method="average")
#hc.single=hclust(dist(x), method="single")

par(mfrow=c(1,1))

#get dendextend library
if (!require('dendextend')) install.packages('dendextend'); library('dendextend')

###### Use the dendextend library to color branches, color labels, thicken branches:
dend <- hc.complete
#Note: k = number of clusters to color
dend=color_branches(dend,k=5, col = c("dark turquoise", "blue", "dark green", "purple","orange"))
dend=color_labels(dend,k=5, col = c("dark turquoise", "blue", "dark green", "purple","orange"))
dend=set(dend, "branches_lwd",2)
plot(dend)
```

Questions?