CSE 416 Section 6!

Zoom University – Global Pandemic Summer

Note: Your have used up 57.6% of your year 2020.

Crossed off ~~anything~~ from your 2020 must-do list yet?

JULY 30, 2020

HONGJUN JACK WU 😆

This material is made with color blind folks in mind.

If there is anything that is not clear or you cannot distinguish **PLEASE** let us know so we can fix it ASAP.

Goal for today!

MAIN GOAL:

NUMPY + VARIABLE ENCODING + CLUSTERING

Materials of the Day

There are three notebooks about:

NumPy: In section demo that I write today.

Variable Encoding: Implementation of VE slides.

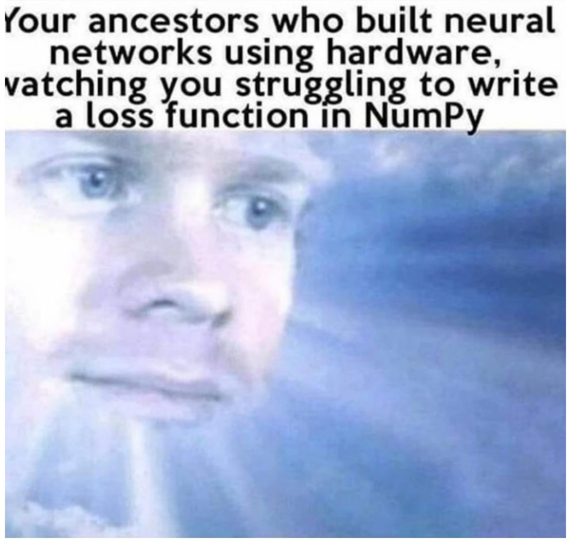
Method Review: **VERY HELPFUL REVIEW**.

We will post them on the course website after we are done with all sections, as always.

Very helpful if you want to dig deep into the implementation of algorithms (and how to actually use them in Python) and might benefit you from doing your next assignment!

Index

I have a joke about Early Stopping but


Your ancestors who built neural
networks using hardware,
vatching you struggling to write
a loss function in NumPy

NumPy

PART I

NUMPY

NumPy:

A library that supports large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

Import Convention: `import numpy as np`

Joke I tell every quarter if I teach NumPy:

NumPy: Here you go, this is the array you asked for.

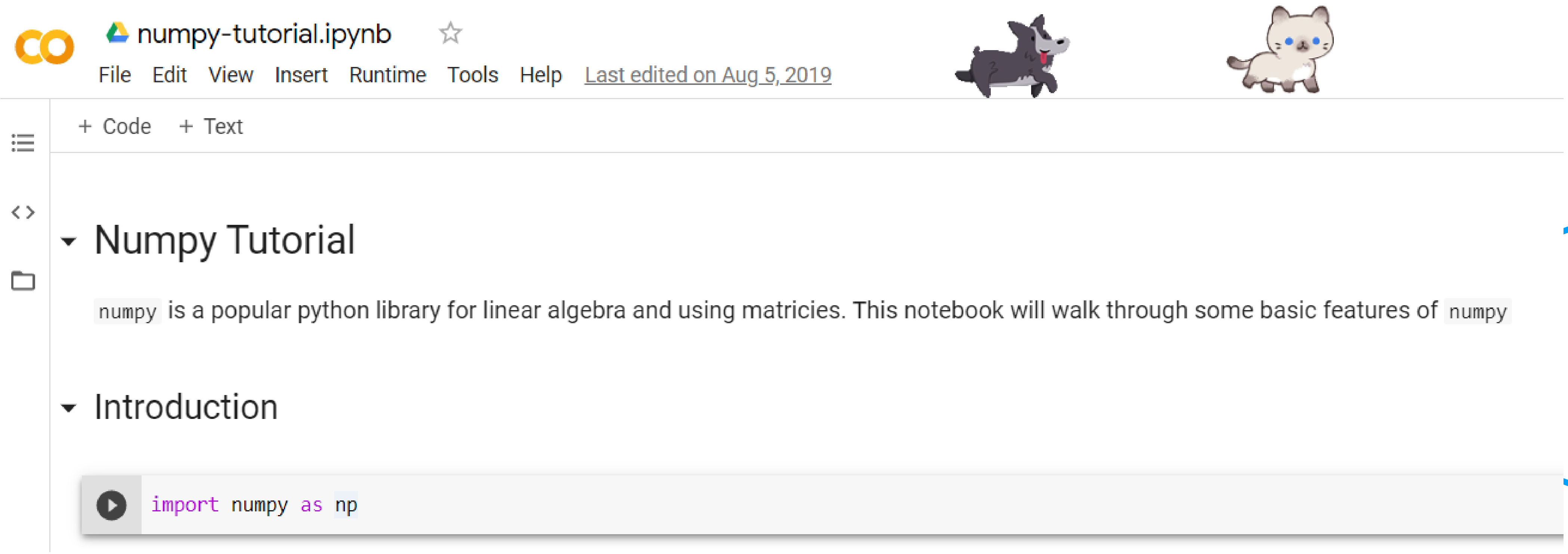
Me: Thank you NumPy!

NumPy: np!

when someone asks you why do you call numpy as np

NUMPY TUTORIAL

We have a notebook called "`numpy-tutorial.ipynb`" for you.
It will be posted no later than the end of today.



Numpy Tutorial

`numpy` is a popular python library for linear algebra and using matricies. This notebook will walk through some basic features of `numpy`

Introduction

```
import numpy as np
```

Variable Encoding

A required pre-processing step when working with categorical data for machine learning algorithms.

PART II

TYPES OF VARIABLES

Numerical data: Features that are only composed of **numbers**, such as integers or floating-point values.

Categorical data: Variables that contain **label values** rather than numeric values. Categorical variables are often called nominal.

Nominal Variable (*Categorical*): Variable comprises a **finite set of discrete values with no relationship** between values.

Ordinal Variable: Variable comprises **a finite set of discrete values with a ranked ordering** between values.

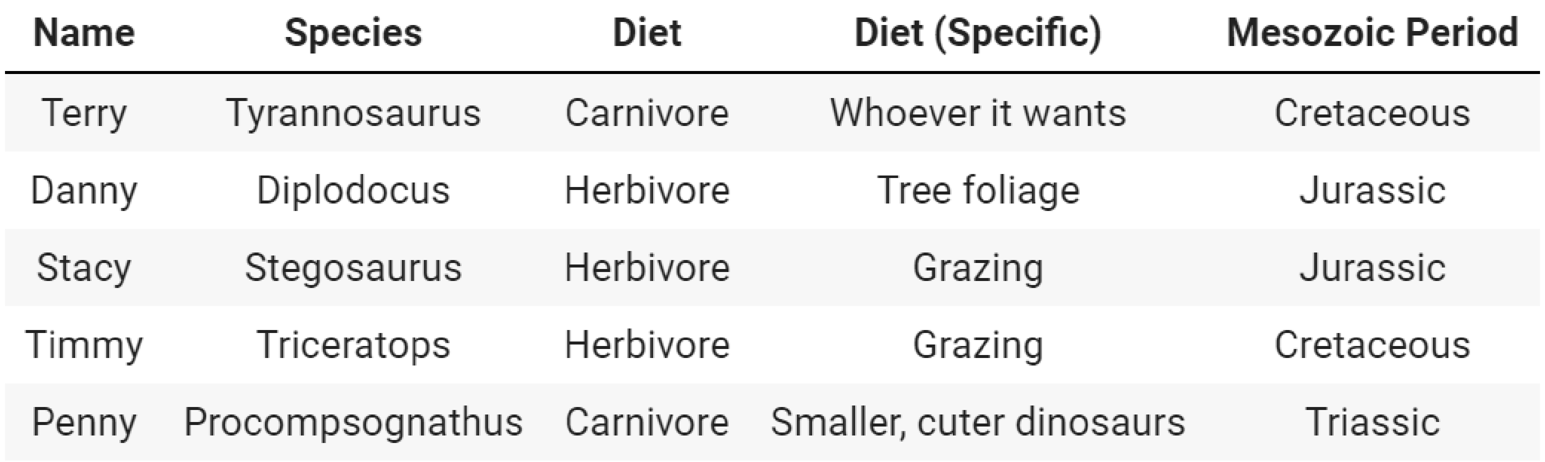
Discretization: A numerical variable can be converted to an ordinal variable by **dividing the range of the numerical variable into bins and assigning values to each bin**. (For example, a numerical variable between 1 and 10 can be divided into an ordinal variable with 5 labels with an ordinal relationship: 1-2, 3-4, 5-6, 7-8, 9-10).

Credit: [1]

VARIABLE ENCODING WITH PYTHON

Many of the previous methods are usable with both categorical and continuous data, but when using **any model with categorical data** there are some extra data processing steps we should take into consideration.

Consider this data set about five dinosaurs.



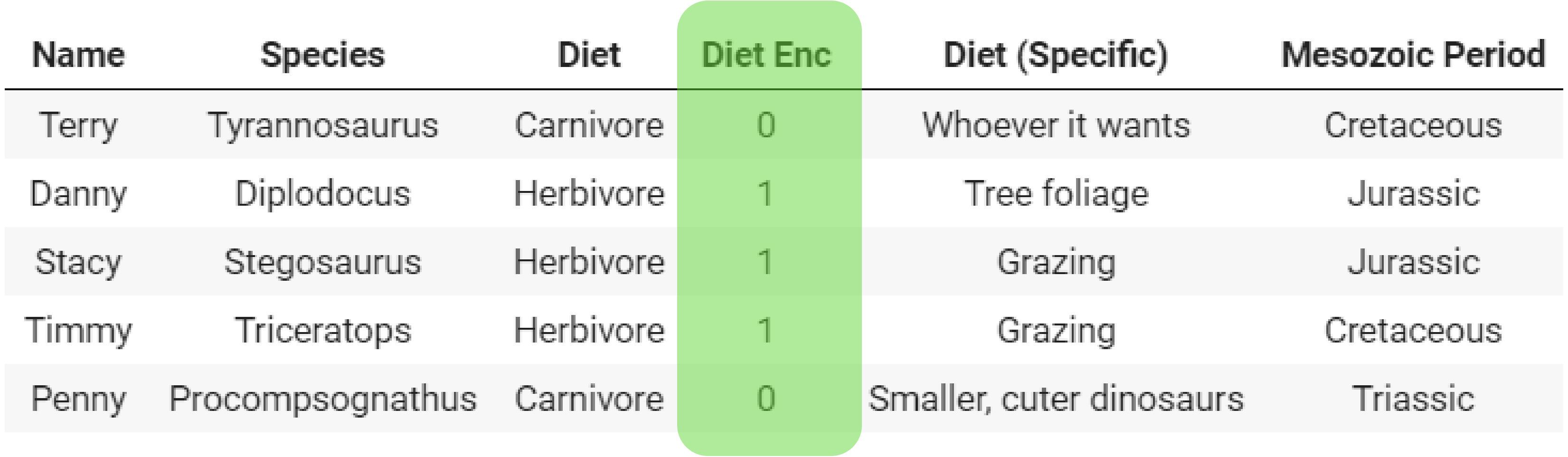
Name	Species	Diet	Diet (Specific)	Mesozoic Period
Terry	Tyrannosaurus	Carnivore	Whoever it wants	Cretaceous
Danny	Diplodocus	Herbivore	Tree foliage	Jurassic
Stacy	Stegosaurus	Herbivore	Grazing	Jurassic
Timmy	Triceratops	Herbivore	Grazing	Cretaceous
Penny	Procompsognathus	Carnivore	Smaller, cuter dinosaurs	Triassic

We have three categories, Diet, Diet (Specific), and Mesozoic Period that have respectively 2, 4, and 3 categories each.

How we treat these depends on how we want to use these variables, as well as what type of model we intend to fit.

BINARY ENCODING

The simplest encoding, when we **only have two categories**, is to assign one category a value of 0 , and the other a category of 1.



Name	Species	Diet	Diet Enc	Diet (Specific)	Mesozoic Period
Terry	Tyrannosaurus	Carnivore	0	Whoever it wants	Cretaceous
Danny	Diplodocus	Herbivore	1	Tree foliage	Jurassic
Stacy	Stegosaurus	Herbivore	1	Grazing	Jurassic
Timmy	Triceratops	Herbivore	1	Grazing	Cretaceous
Penny	Procompsognathus	Carnivore	0	Smaller, cuter dinosaurs	Triassic

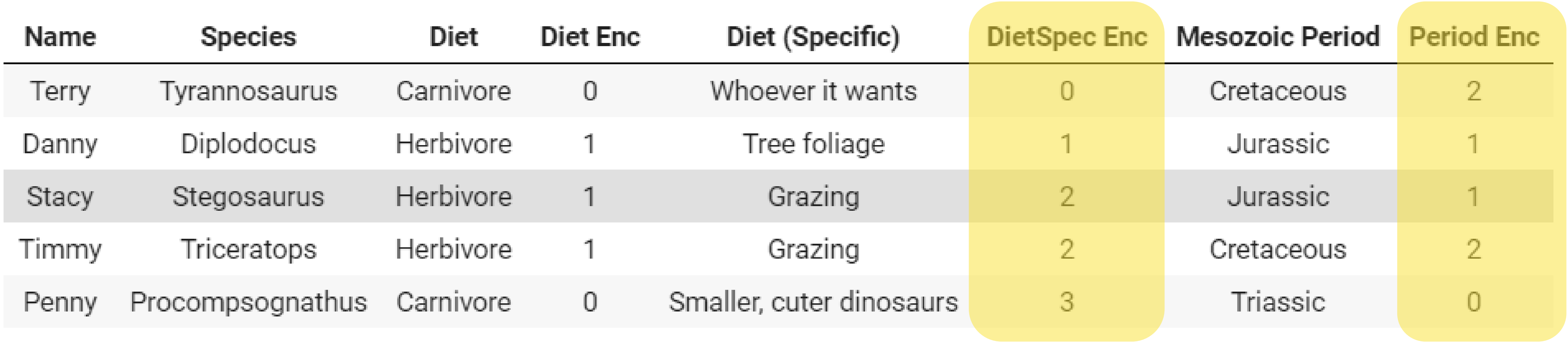
A binary classification allows us to use the Diet variable in a model such as **Linear Regression**, as a modification to the intercept or as an interaction with other variables by creating interaction terms.

A **Decision Tree** can split on the variable based on the value being <.5 or >.5, or the diet can influence the score of a Logistic Regression while treating it the same way we would any other numeric valued feature.

N-CLASS ENCODING

When a feature has multiple classes, such as the *Period* or *Diet (Specific)* features, a simple binary encoding won't suffice.

We could consider **assigning each value a distinct integer to tell them apart**, such as this table.



Name	Species	Diet	Diet Enc	Diet (Specific)	DietSpec Enc	Mesozoic Period	Period Enc
Terry	Tyrannosaurus	Carnivore	0	Whoever it wants	0	Cretaceous	2
Danny	Diplodocus	Herbivore	1	Tree foliage	1	Jurassic	1
Stacy	Stegosaurus	Herbivore	1	Grazing	2	Jurassic	1
Timmy	Triceratops	Herbivore	1	Grazing	2	Cretaceous	2
Penny	Procompsognathus	Carnivore	0	Smaller, cuter dinosaurs	3	Triassic	0

This potentially works for one variable, but causes many problems for the other.

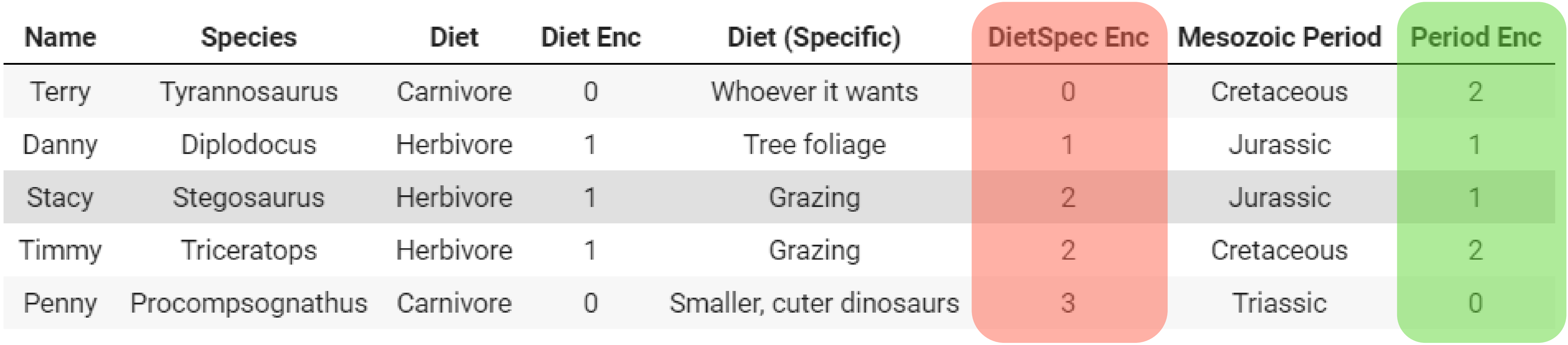
Which is okay, and why? Which has problems?

K OR K-1 BINARY ENCODING

It makes no sense to treat the Diet (Specific) variable as **being in some way ordered.**

If we were to use it in the current state in a decision tree, it might try to identify herbivores by splitting on DietSpec Enc>0.5 , classifying the Tyrannosaurus correctly and the Procompsognathus incorrectly.

The ordering of these categories is meaningless; while the split might improve prediction accuracy, it is similarly meaningless.

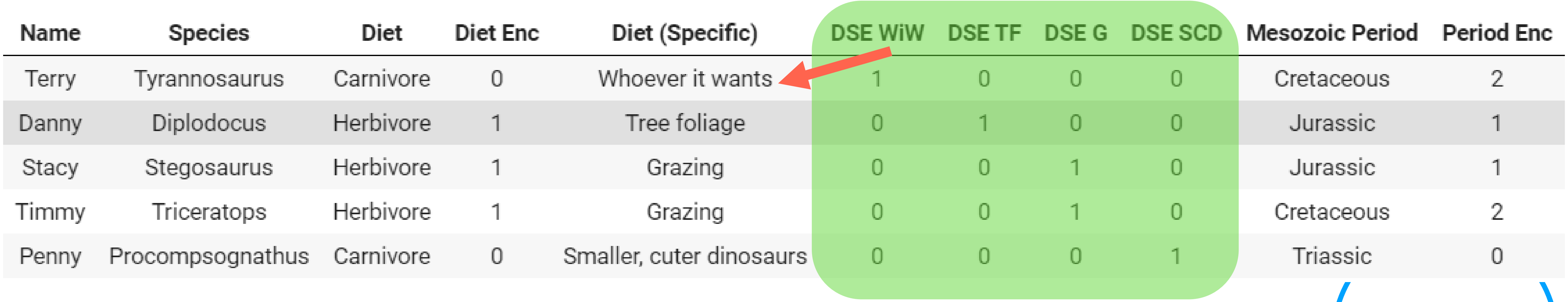


Name	Species	Diet	Diet Enc	Diet (Specific)	DietSpec Enc	Mesozoic Period	Period Enc
Terry	Tyrannosaurus	Carnivore	0	Whoever it wants	0	Cretaceous	2
Danny	Diplodocus	Herbivore	1	Tree foliage	1	Jurassic	1
Stacy	Stegosaurus	Herbivore	1	Grazing	2	Jurassic	1
Timmy	Triceratops	Herbivore	1	Grazing	2	Cretaceous	2
Penny	Procompsognathus	Carnivore	0	Smaller, cuter dinosaurs	3	Triassic	0

K OR K-1 BINARY ENCODING CONT.

We instead choose to encode the variable as either K or K−1 binary encodings, where K is the number of observed classes.

The data set would then look like so.



Name	Species	Diet	Diet Enc	Diet (Specific)	DSE WiW	DSE TF	DSE G	DSE SCD	Mesozoic Period	Period Enc
Terry	Tyrannosaurus	Carnivore	0	Whoever it wants	1	0	0	0	Cretaceous	2
Danny	Diplodocus	Herbivore	1	Tree foliage	0	1	0	0	Jurassic	1
Stacy	Stegosaurus	Herbivore	1	Grazing	0	0	1	0	Jurassic	1
Timmy	Triceratops	Herbivore	1	Grazing	0	0	1	0	Cretaceous	2
Penny	Procompsognathus	Carnivore	0	Smaller, cuter dinosaurs	0	0	0	1	Triassic	0

K OR K-1 BINARY ENCODING CONT.

We can either choose to leave one of these categories off, or include all K categories.

If we include all K categories:

1. Unobserved classes can be handled, and will use no baseline with each encoding equal to 0.
2. Some methods may have co-linearity issues that cause convergence problems (Linear regression).

If we drop one class down to $K-1$:

1. Unobserved classes will be treated as an instance of the dropped class, which is rolled into model 'intercept' terms.
2. Avoids co-linearity problems when not using regularization.

Our Diet Enc variable is a $K-1$ encoding that could not handle an 'Omnivore' in the dataset.

Similar to previous binary encoding, this lets us treat the variables just as we would any other continuous, numerical values.

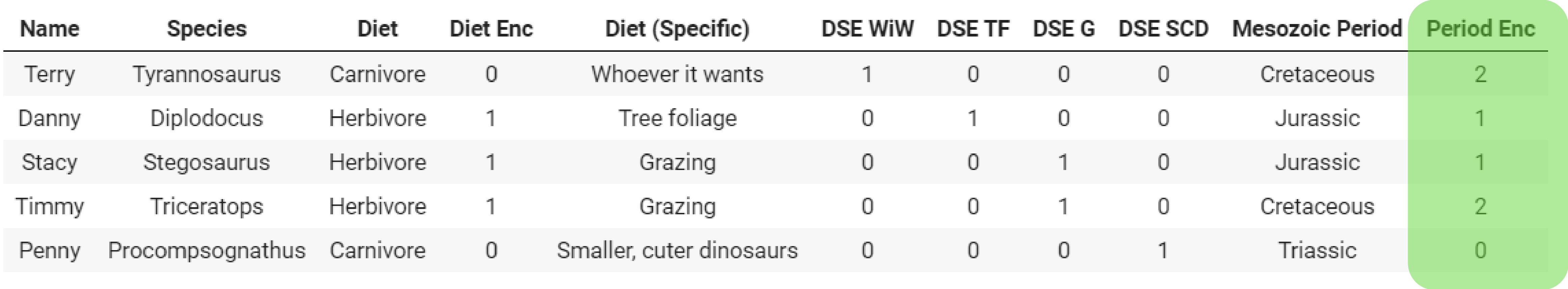
ORDINAL ENCODING

Unlike the encoding of the specific diets, the order of the Period Enc does have meaning.

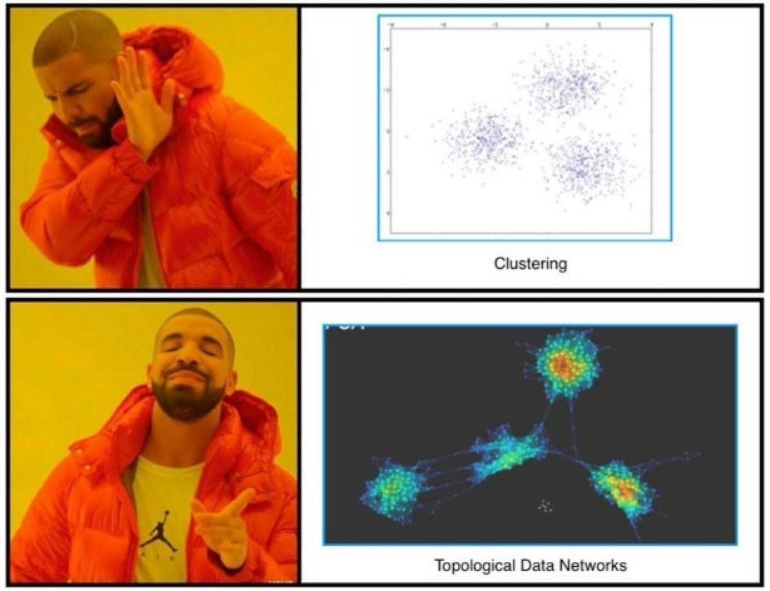
The numeric value of each encoding corresponds chronologically to the order in which each period took place.

If a decision tree were to split on **PeriodEnc>0.5** it would specifically be splitting on dinosaur species that lived after the Triassic period.

Due to the ordered nature of the categories, we can take advantage of a similarly ordered encoding.



Name	Species	Diet	Diet Enc	Diet (Specific)	DSE WiW	DSE TF	DSE G	DSE SCD	Mesozoic Period	Period Enc
Terry	Tyrannosaurus	Carnivore	0	Whoever it wants	1	0	0	0	Cretaceous	2
Danny	Diplodocus	Herbivore	1	Tree foliage	0	1	0	0	Jurassic	1
Stacy	Stegosaurus	Herbivore	1	Grazing	0	0	1	0	Jurassic	1
Timmy	Triceratops	Herbivore	1	Grazing	0	0	1	0	Cretaceous	2
Penny	Procompsognathus	Carnivore	0	Smaller, cuter dinosaurs	0	0	0	1	Triassic	0

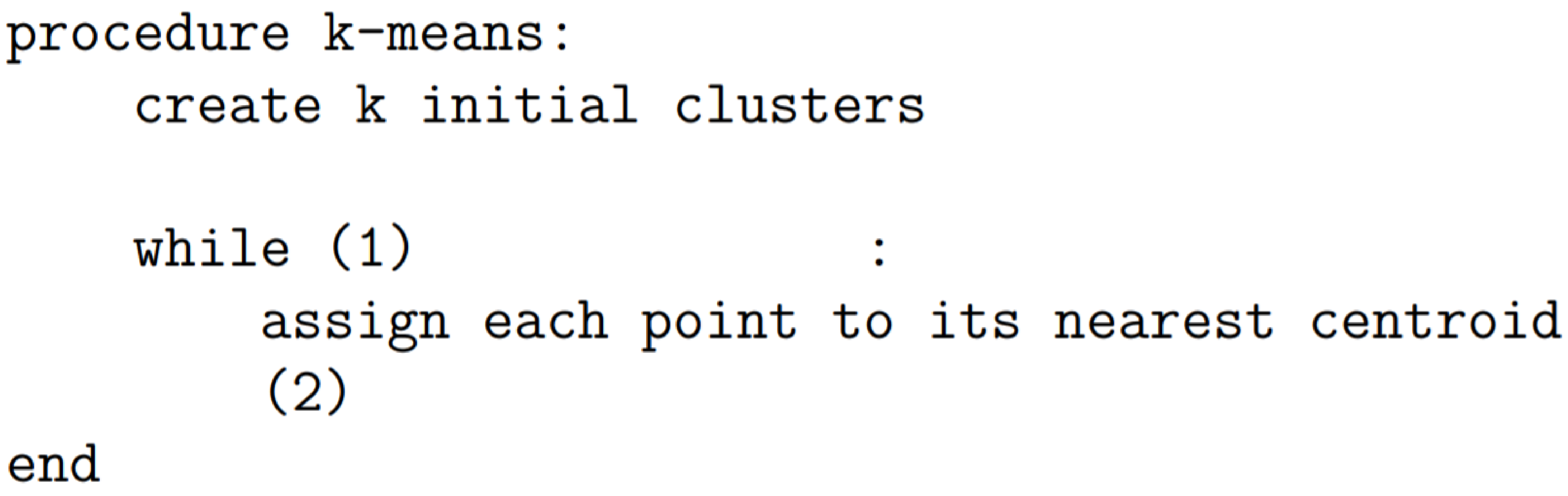


Clustering Review

PART III

K-MEANS CLUSTERING #1 QUESTION

Below we have provided partial pseudo-code for the k-means algorithm. Fill in the missing parts of the algorithm at locations marked (1) and (2).

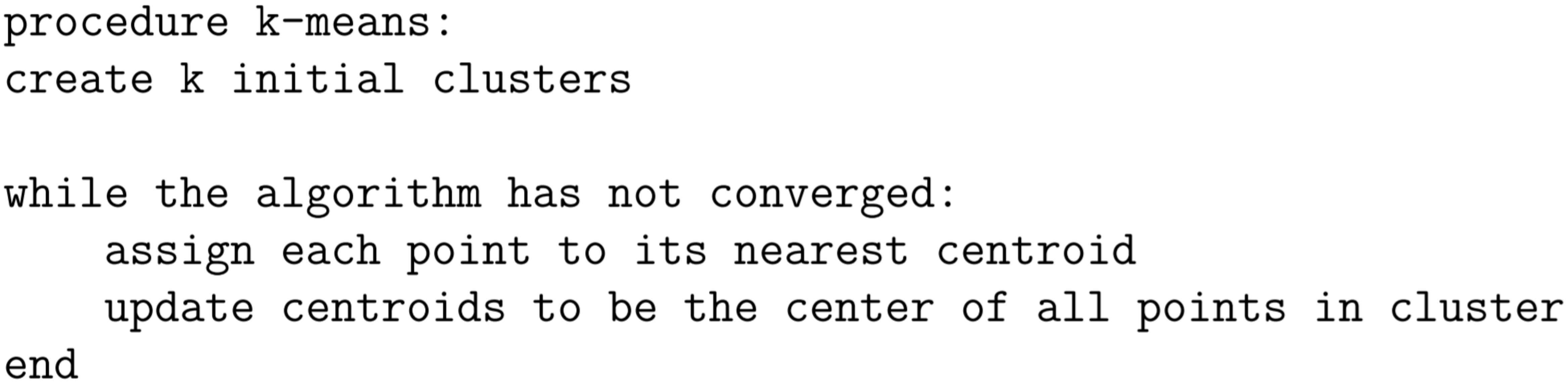


```
procedure k-means:
    create k initial clusters

    while (1)                  :
        assign each point to its nearest centroid
        (2)
end
```

K-MEANS CLUSTERING #1 ANSWER

Below we have provided partial pseudo-code for the k-means algorithm. Fill in the missing parts of the algorithm at locations marked (1) and (2).



```
procedure k-means:
create k initial clusters

while the algorithm has not converged:
    assign each point to its nearest centroid
    update centroids to be the center of all points in cluster
end
```

K-MEANS CLUSTERING #2 QUESTION

Compare the merits and drawbacks of **k-means** to **hierarchical clustering** with regards to the following:

(a) Efficiency?

(b)Hyper-parameters?

K-MEANS CLUSTERING #2 ANSWER

Compare the merits and drawbacks of **k-means** to **hierarchical clustering** with regards to the following:

(a) Efficiency

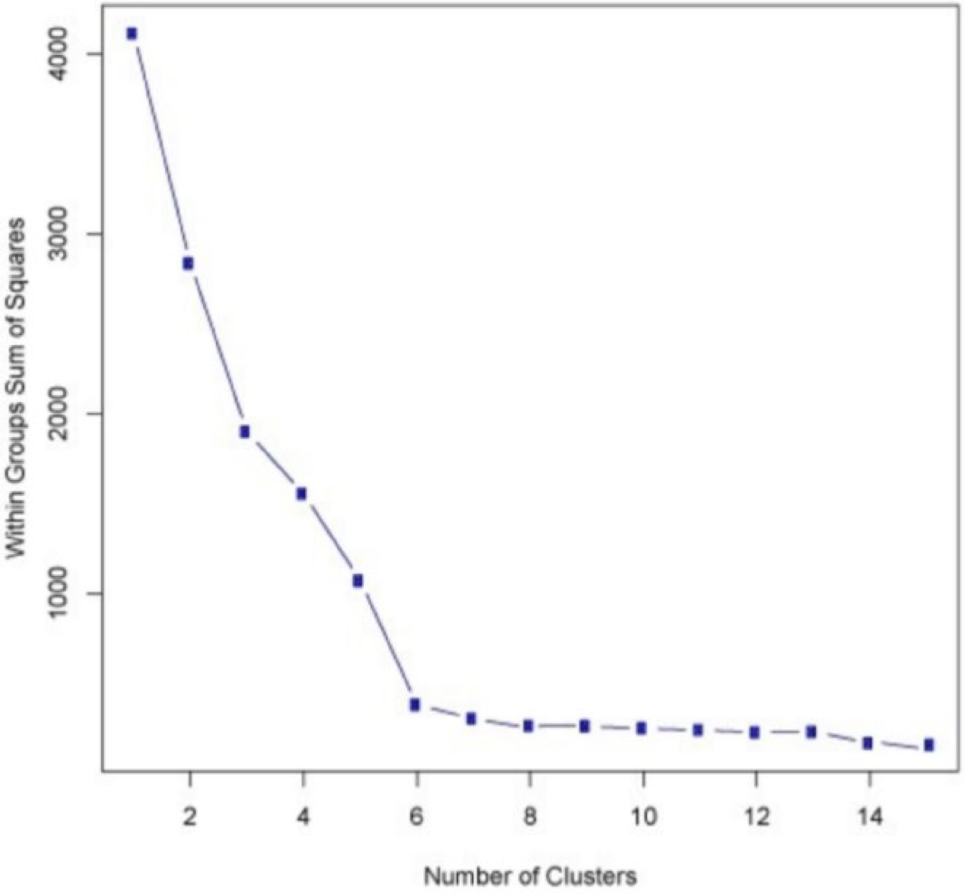
Solution: **k-means is more efficient in general than hierarchical clustering.**

(b)Hyper-parameters

Solution: **k-means requires picking k before the algorithm begins, whereas you can pick clusters for hierarchical after the algorithm has run.** However, you must still pick a distance metric for hierarchical before starting

K-MEANS CLUSTERING #3 QUESTION

Given the following graph, what is a common default for the **number of clusters** for our k-means algorithm?



K-MEANS CLUSTERING #3 ANSWER

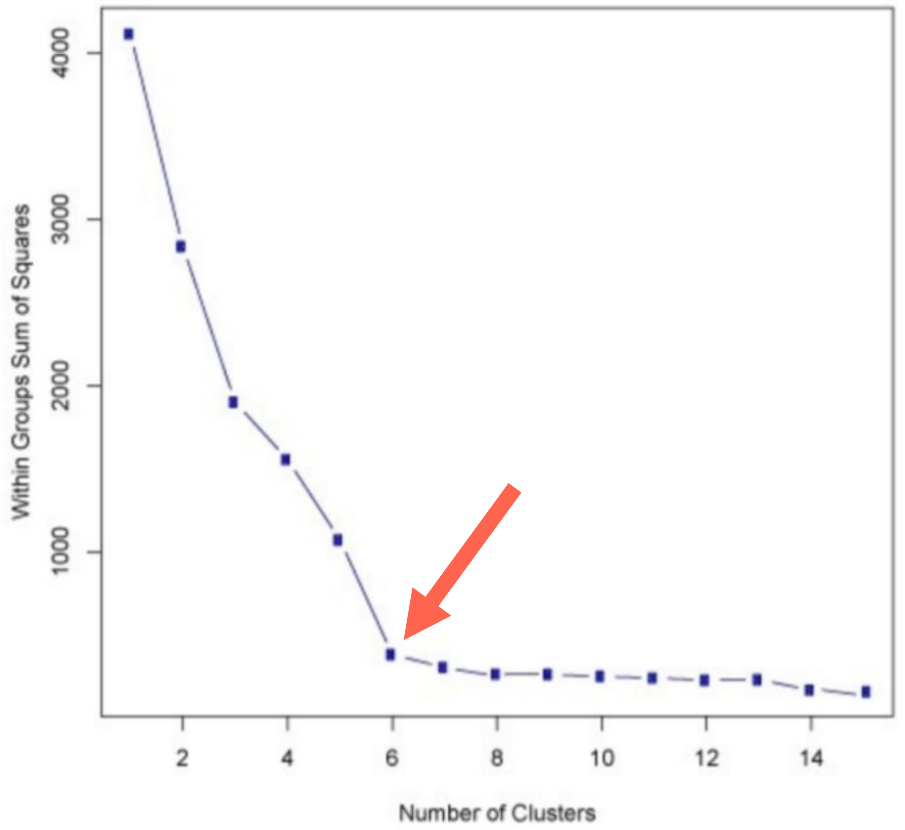
Given the following graph, what is a common default for the **number of clusters** for our k-means algorithm?

Solution:

6 is the optimal number of clusters.

Recall from lecture that **cluster heterogeneity decreases monotonically as k approaches n**.

Therefore we want to pick a value of k such that heterogeneity is low but does not decrease by a trivial amount with more clusters.



K-MEANS CLUSTERING #4 QUESTION

True or False:

Between two iterations of the k-means algorithm it is possible that no points are assigned to different clusters.

Justify your answer.

K-MEANS CLUSTERING #4 ANSWER

True or False:

Between two iterations of the k-means algorithm it is possible that no points are assigned to different clusters.

Justify your answer.

Solution:

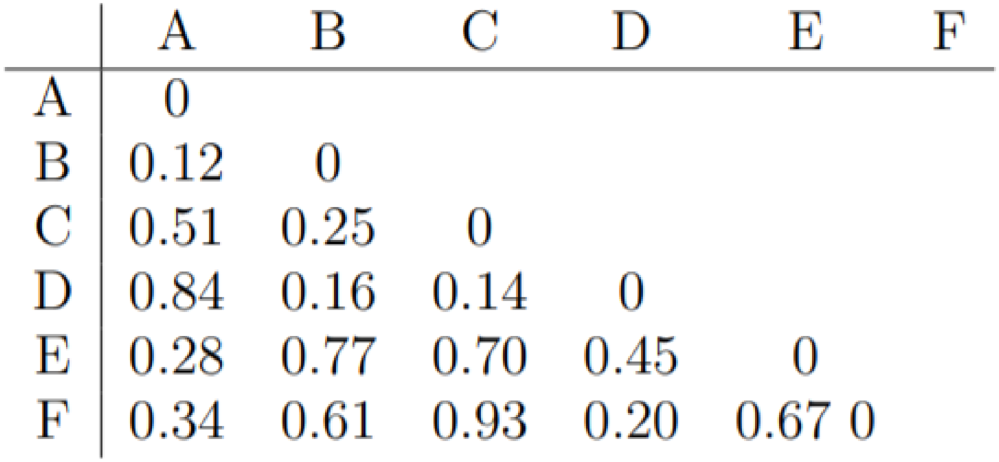
True. Consider the state of cluster assignments once the algorithm has reached a local minima. The centroid will not move and all points will be classified the same between iterations.

HIERARCHICAL CLUSTERING QUESTION

Suppose that the following distance matrix is given for 6 objects:

(a)Show the final result of hierarchical clustering with single linkage by drawing a dendrogram.

(b) Show the final result of hierarchical clustering with complete linkage by drawing a dendrogram.



	A	B	C	D	E	F
A	0					
B	0.12	0				
C	0.51	0.25	0			
D	0.84	0.16	0.14	0		
E	0.28	0.77	0.70	0.45	0	
F	0.34	0.61	0.93	0.20	0.67	0

Single Linkage:

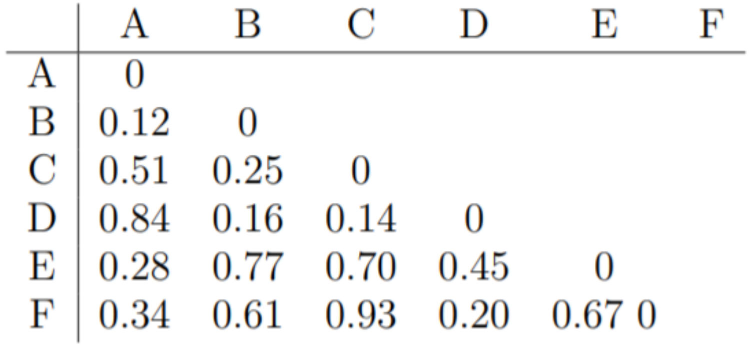
$$\min_{x_i \in C_1, x_j \in C_2} d(x_i, x_j)$$

Complete Linkage:

$$\max_{x_i \in C_1, x_j \in C_2} d(x_i, x_j)$$

HIERARCHICAL CLUSTERING A ANSWER

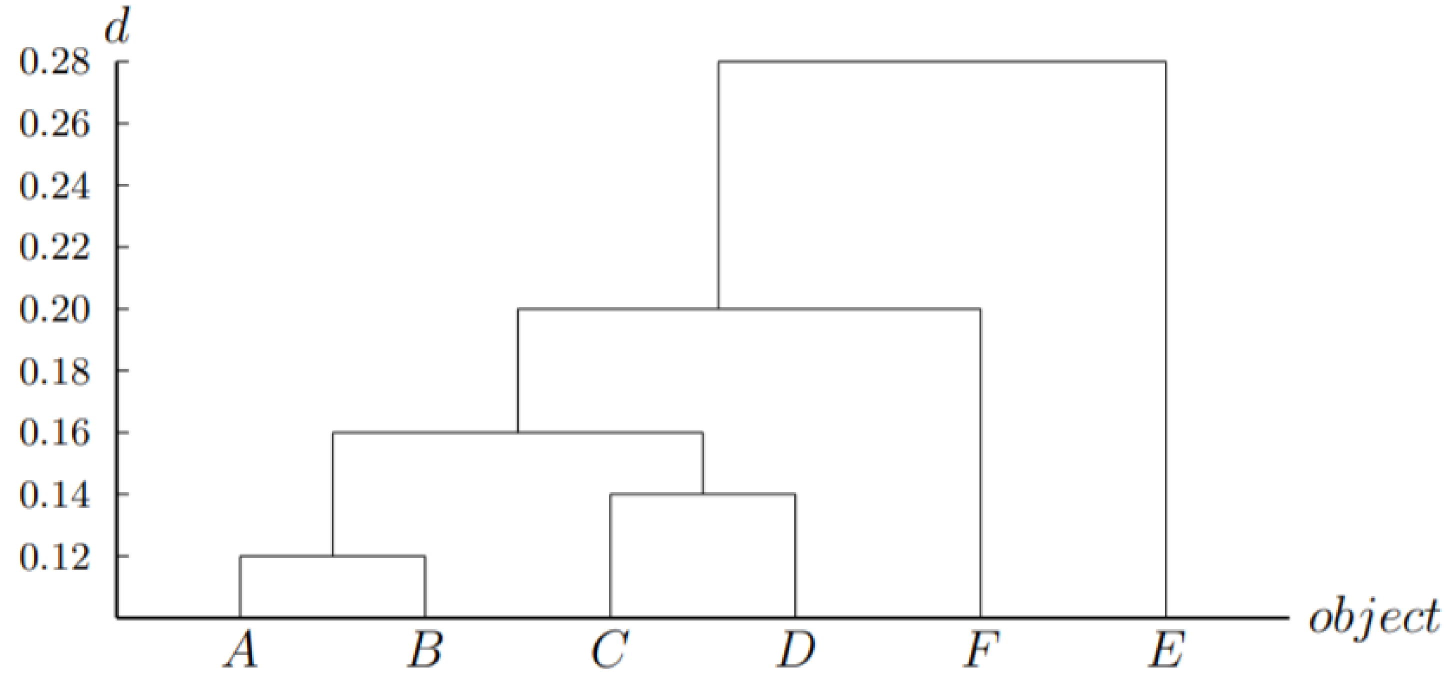
(a) Show the final result of hierarchical clustering with single linkage by drawing a dendrogram.



	A	B	C	D	E	F
A	0					
B	0.12	0				
C	0.51	0.25	0			
D	0.84	0.16	0.14	0		
E	0.28	0.77	0.70	0.45	0	
F	0.34	0.61	0.93	0.20	0.67	0

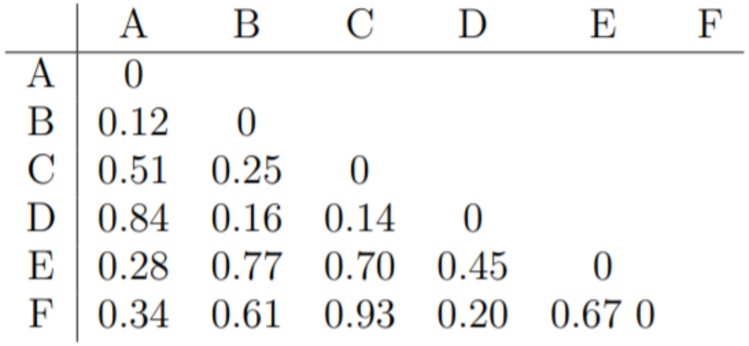
Single Linkage:

$$\min_{x_i \in C_1, x_j \in C_2} d(x_i, x_j)$$



HIERARCHICAL CLUSTERING B ANSWER

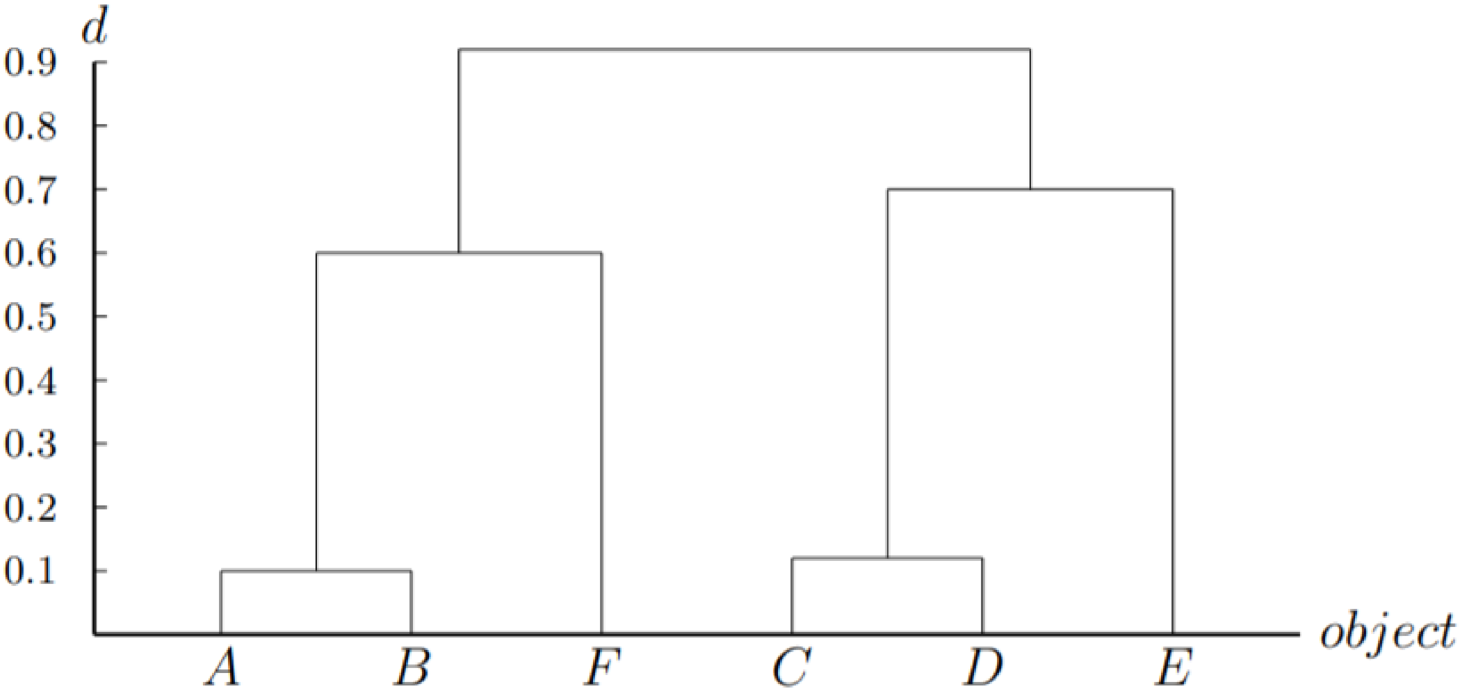
(b) Show the final result of hierarchical clustering with complete linkage by drawing a dendrogram.

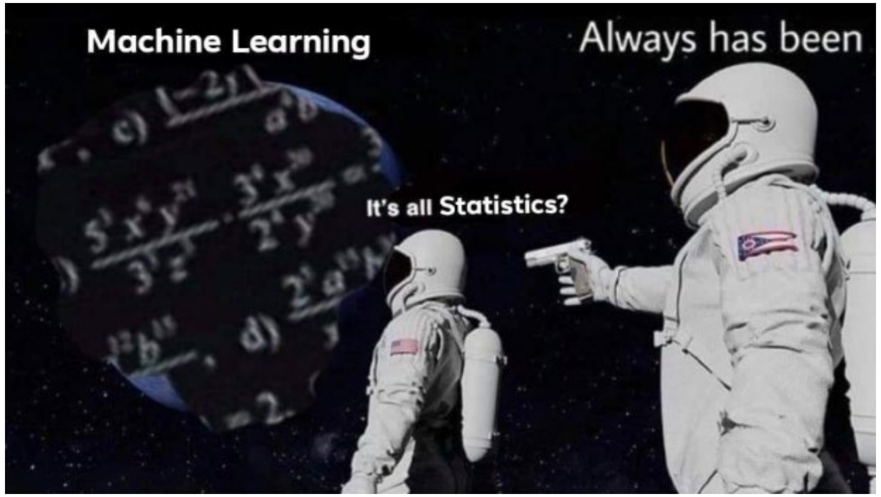


	A	B	C	D	E	F
A	0					
B	0.12	0				
C	0.51	0.25	0			
D	0.84	0.16	0.14	0		
E	0.28	0.77	0.70	0.45	0	
F	0.34	0.61	0.93	0.20	0.67	0

Complete Linkage:

$$\max_{x_i \in C_1, x_j \in C_2} d(x_i, x_j)$$





Other Stuff

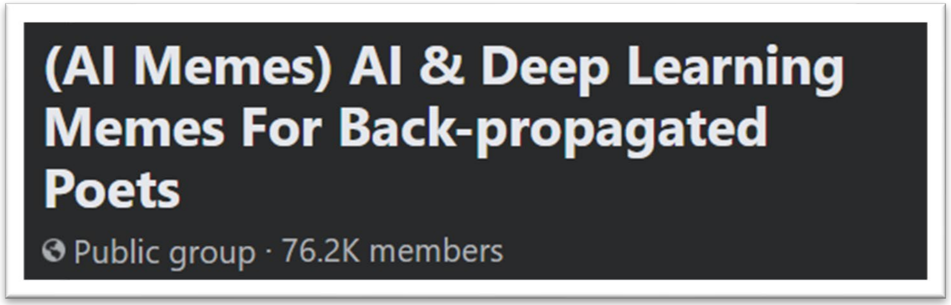
PART IDK

MEMES

To be honest the most fun thing (at least for me) after taking 416 is you start to understand memes about machine learning…

Here’s my source of memes lol as the quarter goes you’ll understand these memes more and more!

https://www.facebook.com/groups/1638417209555402

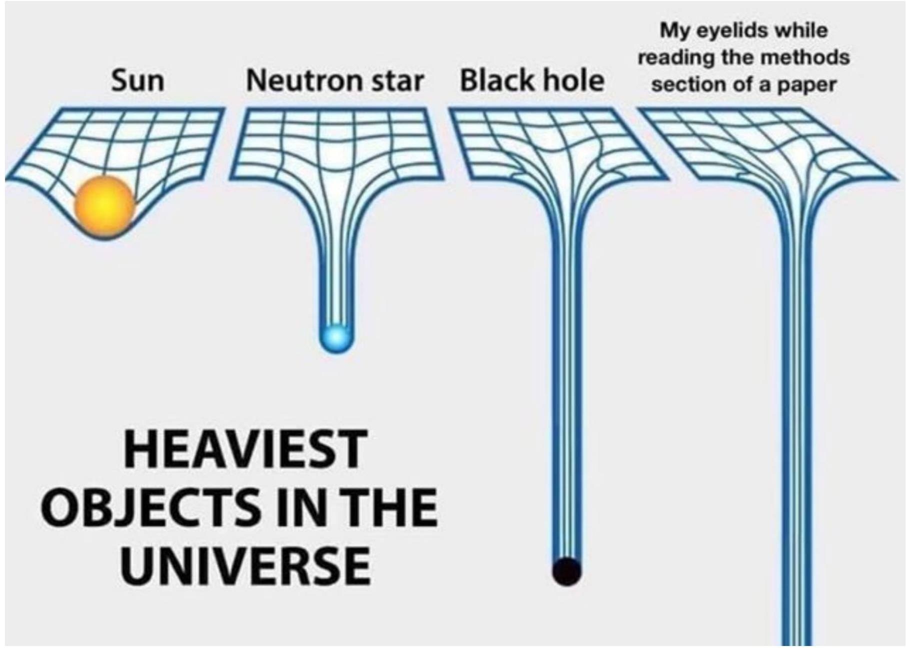


(AI Memes) AI & Deep Learning Memes For Back-propagated Poets

Public group · 76.2K members

CREDITS

1. Ordinal and One-Hot Encodings for Categorical Data.
2. Clustering handout by Hunter Schafer, Solutions.
3. NumPy Tutorial by Hunter Schafer (Posted on Website).
4. Variable Encoding by Anne Wagner (Posted on Website).
5. Methods Review by Anne Wagner (Posted on Website).


Sun
Neutron star
Black hole
My eyelids while reading the methods section of a paper
HEAVIEST OBJECTS IN THE UNIVERSE

LICENSE

This material is originally made by Hongjun Wu for the course CSE416: Introduction to Machine Learning in the Summer 2020 quarter taught by Vinitra Swamy, at University of Washington Paul G. Allen School of Computer Science and Engineering.

It was originally made for educational purpose, in a section taught by teaching assistants to help students explore material in more depth.

Any other materials used are cited in the Credits section.

Anyone, especially other educators and students, are welcomed and strongly encouraged to study and use this material.