Unsupervised Learning (clustering)

DATA 202 21FA

Logistics

- Modeling quizzes
- Modeling homework
- Final Projects

Statistical Inference vs Predictive Modeling

> Do we need to test for normality etc. for linear regression?

If you want to make **inferences**.

- **Questions about underlying relationships**
  - Is height related to weight?
  - How must faster can you run with caffeine?
  - Does our new website lead to higher sales?
- **Questions about prediction**
  - Guess weight, given height and other details
  - Predict running speed.
  - Predict today's sales.

More Q&A

> When do we use each model?

`parsnip` makes it easy to try many models! But:

- Does it need to be **accurate**? Try many models, probably RF.
- Does it need to be **understandable**? Decision Tree, or sometimes linear regression
  - Very simple shallow trees can be great for decision-makers.
- Will it need to **extrapolate**, e.g., over time? Linear regression.
- etc.

> Don't home prices change year-to-year?

Yeah, we're oversimplifying the Ames examples. Final project?

More Q&A

> Why would MAPE be different than MAE?

- Predicting $5k for a $1k sale: $4k absolute error, 400% percent error.
- Predicting $105k for a $101k sale: $4k absolute error, 5% percent error.

> Can we change final projects after proposal?

Yes.

> Can we use multiple variables in linreg?

Yes! But beware: interpreting coefficients can get tricky.

Unsupervised Learning

- So far we have been doing *supervised* learning, where have a *target* we're trying to predict.
  - "How much will these homes sell for?"
  - "How long will this person spend watching this video?"
- **Unsupervised** learning works when we don't have an exact target to predict, or we want to explore relationships in the data.
  - "What general types of homes are on the market right now?"
  - "What are some different segments of our customer base?"
  - "Are there distinct types of Covid-19 symptoms?"
- **Clustering** is one very common type of unsupervised learning.

Clustering

Goal: put observations into groups

- Those in the *same* group should be *similar to each other*
- Those in *different* groups should be *different*.

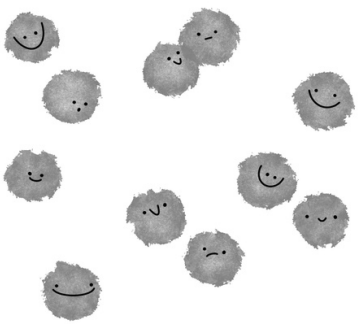
Crucial questions:

- How many groups?
- How do we define "similar" / "different"?

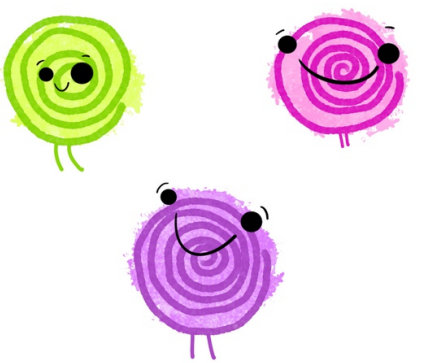
k-means clustering

: assign each observation to one of k clusters based on the nearest cluster centroid.

OBSERVATIONS



cluster CENTROIDS

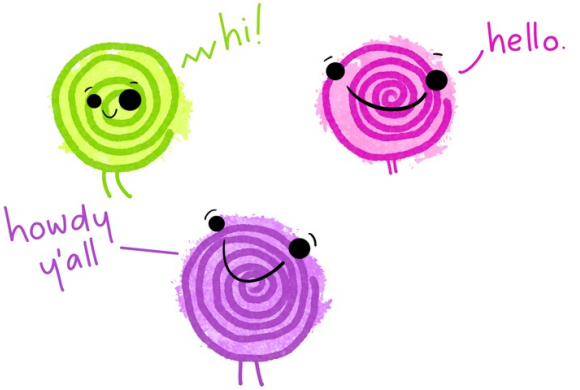


@allison_horst

Artwork by @allison_horst

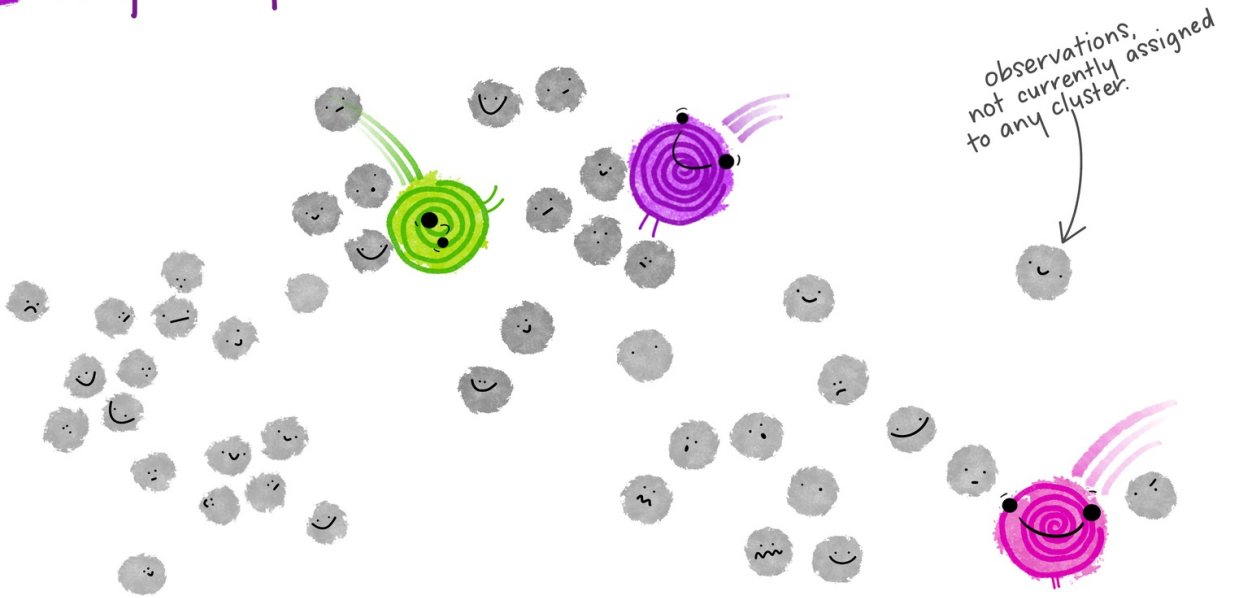
① Specify the number of clusters (in this example, k = 3).

Then imagine k cluster **centroids** are created.



@allison_horst

2 Those k centroids get randomly placed in your space.

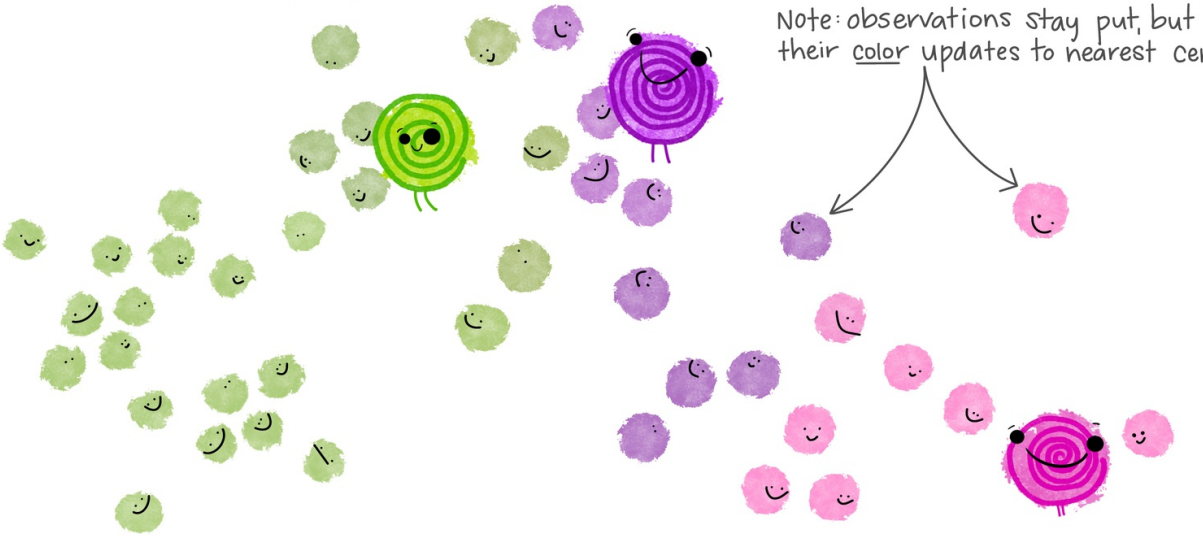


@allison_horst

3

Each observation gets temporarily "assigned" to its closest centroid.

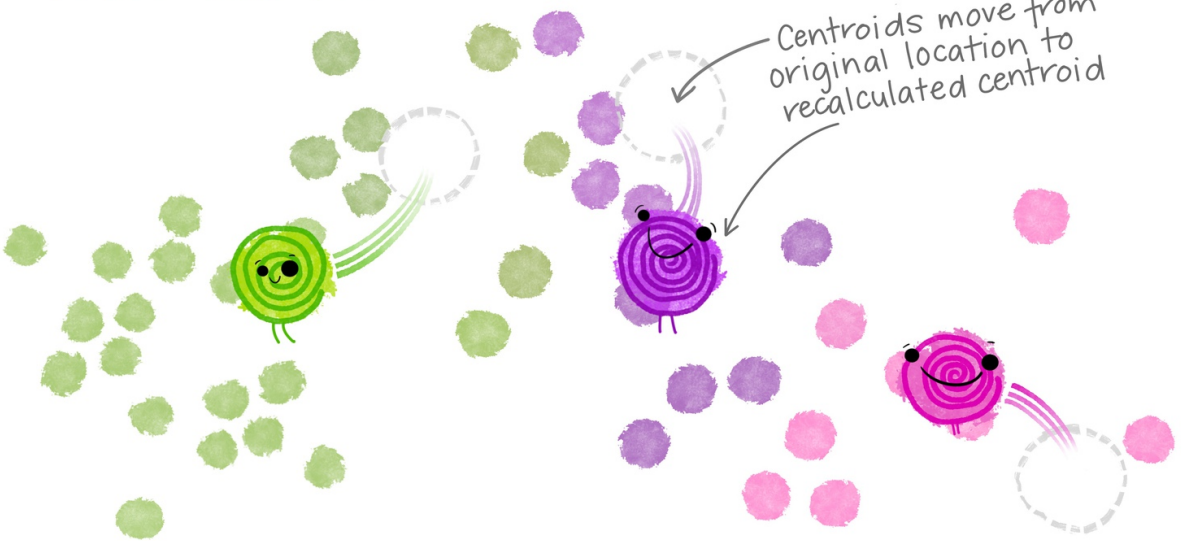
(e.g. by Euclidean distance)



@allison_horst

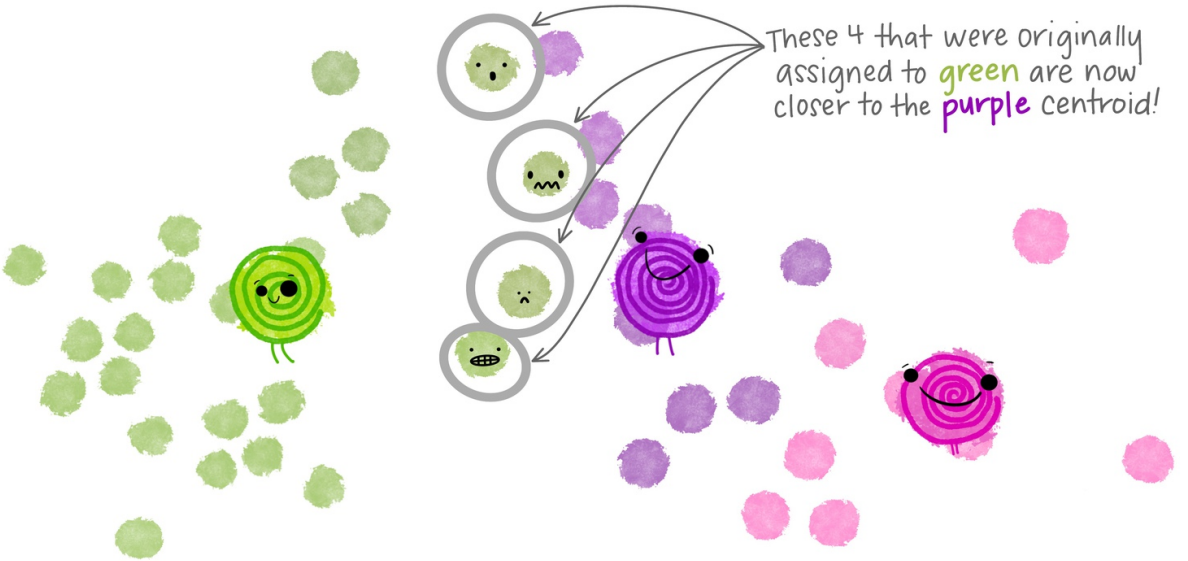
4

Then the centroid of each cluster is calculated based on all observations assigned to that cluster...



@allison_horst

UH OH. Now that the cluster centroids have moved, some of the observations are now closer to a different centroid!

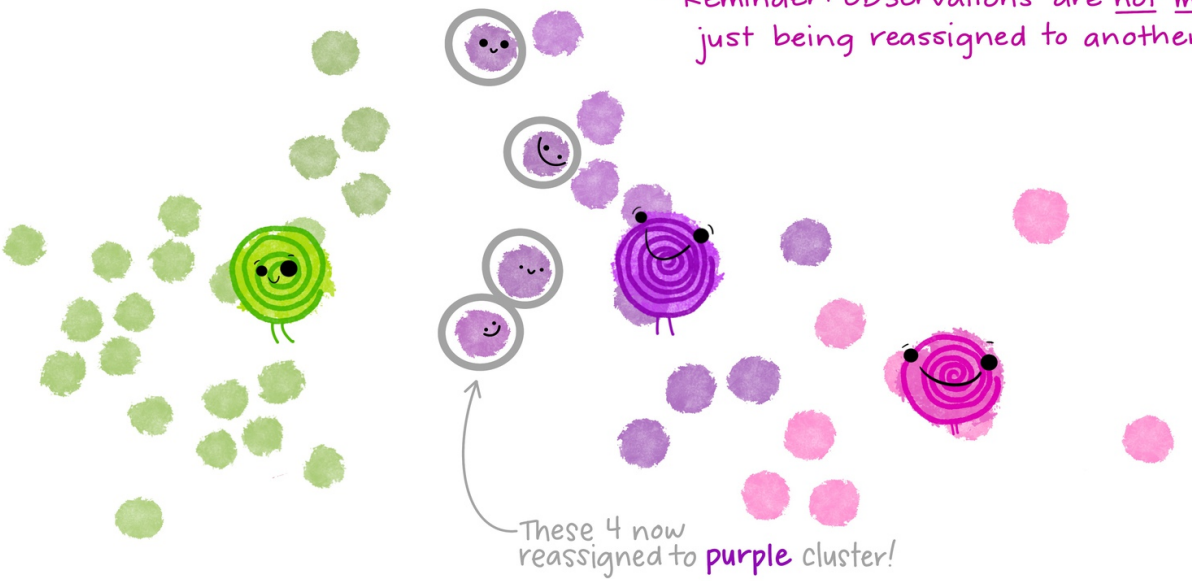


@allison_horst

5

NO PROBLEM!
Observations get reassigned* to a different cluster based on the recalculated centroid.

*Reminder: observations are not moving, just being reassigned to another cluster.



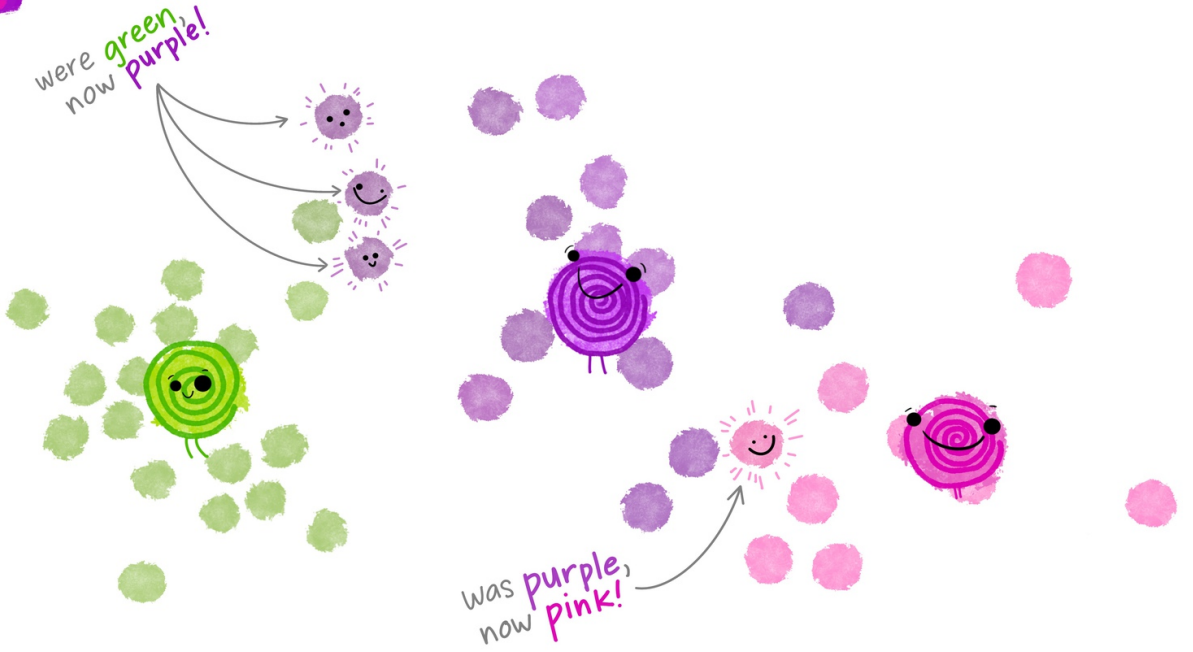
@allison_horst

6

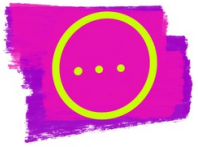
But now that observations have been reassigned, the centroids need to move again [recalculate centroids from updated clusters]

@allison_horst

Again, now observations are reassigned as needed to the closest centroid.



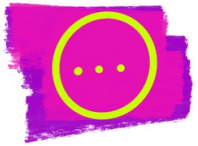
@allison_horst



Then the centroid for each cluster is recalculated...

...which means observations will be reassigned...

@allison_horst



That iterative process of

Recalculate cluster centroids
↳ Reassign observations to nearest centroid
↳ Recalculate cluster centroids
↳ Reassign observations to nearest centroid
↳ Recalculate cluster centroids
↳ Reassign observations to nearest centroid

↓

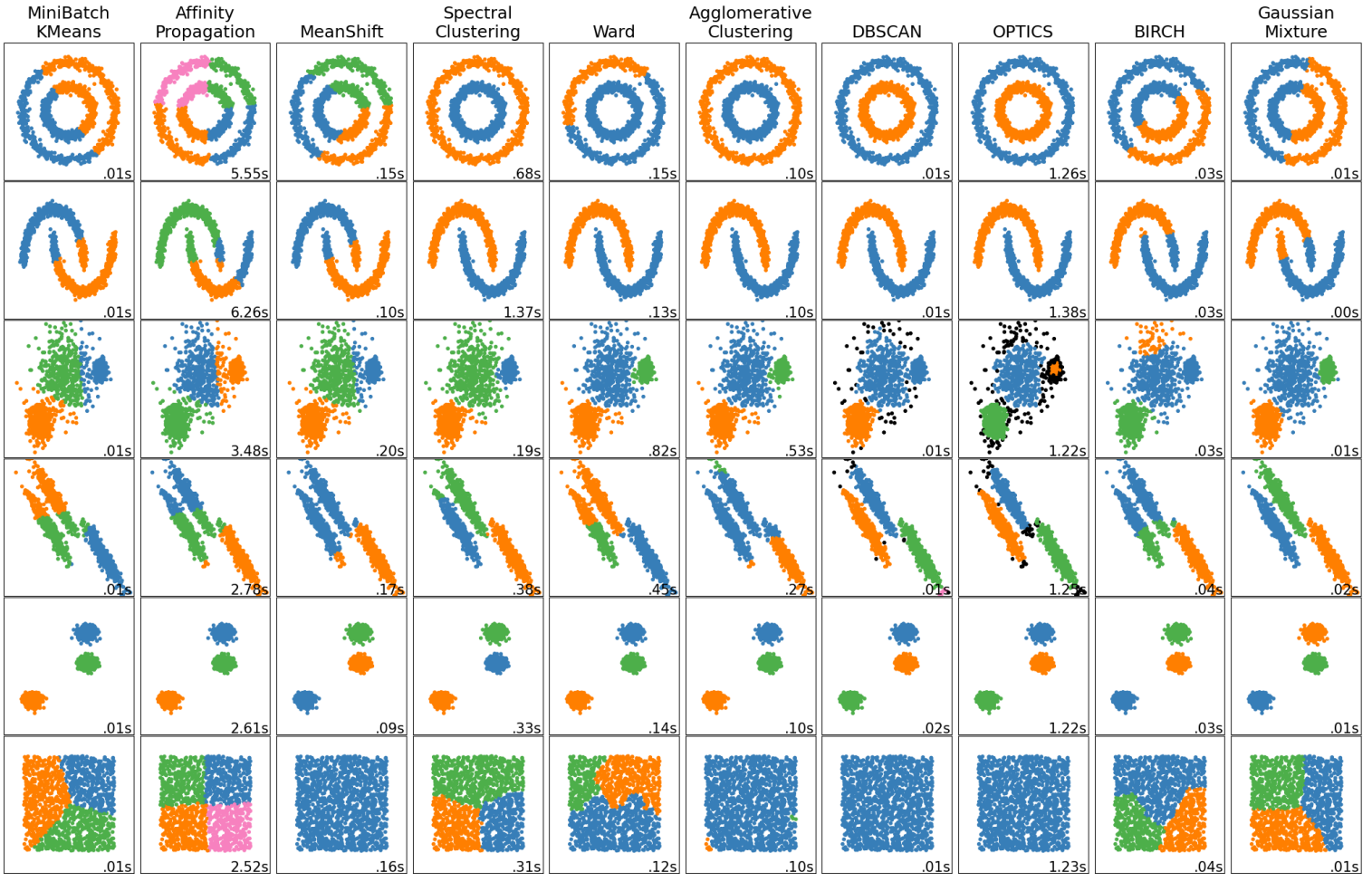
Continues until nothing is moving or being reassigned anymore!

@allison_horst

Which means the iteration is done and each observation is assigned to its final cluster.

@allison_horst

Many types of clustering algorithms



Source: sklearn documentation

```r
set.seed(20201120)
data_for_clustering <- ames_all %>%
  select(Latitude, Longitude) %>%
  #select(Year_Built, Gr_Liv_Area) %>%
  mutate(
    #Gr_Liv_Area = rescale(Gr_Liv_Area, to = c(0, 1)),
    #Year_Built = rescale(Year_Built, to = c(0, 1))
  )
clustering_results <- kmeans(
  data_for_clustering, nstart = 4, centers = 3
)


ames_with_clusters <- ames_all %>%
  mutate(cluster = as.factor(clustering_results$cluster))
```

```r
glance(clustering_results)
```

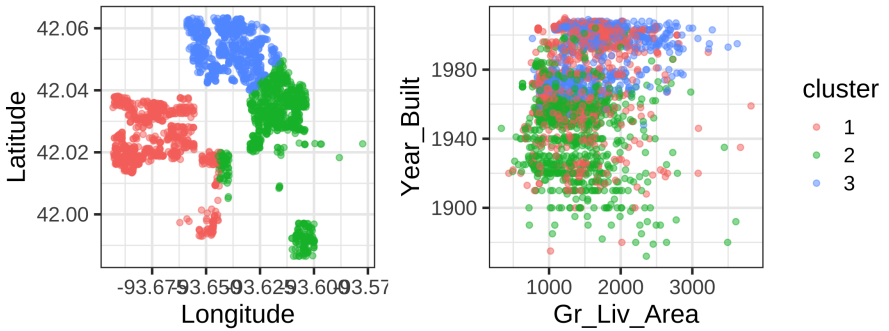
```
# A tibble: 1 × 4
  totss tot.withinss betweenss  iter
  <dbl>        <dbl>     <dbl> <int>
1  2.43        0.687      1.75     3
```

```r
tidy(clustering_results)
```

```
latlong_plot <-
  ggplot(ames_with_clusters, aes(y = Latitude, x = Longitude, color = cluster)) +
    geom_point(alpha = .5)

year_area_plot <-
  ggplot(ames_with_clusters, aes(x = Gr_Liv_Area, y = Year_Built, color = cluster)) +
  #coord_equal() +
    geom_point(alpha = .5)

library(patchwork)
latlong_plot + year_area_plot + plot_layout(guides='collect')
```

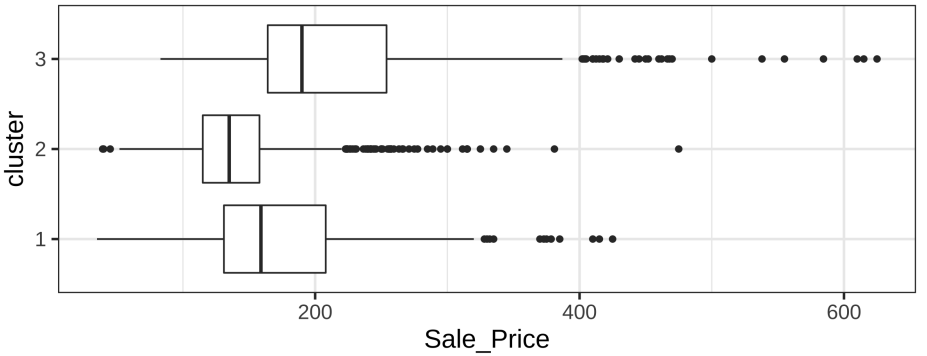


Activities

1. What differences do you notice between the plot on the left and the plot on the right?
2. Try increasing the number of `centers`. What changes about both plots?
3. Use only `Year_Built` for clustering (removing latitude and longitude). What can you say about the age of homes in different parts of town?
4. Try clustering using `select(Latitude, Longitude,` changes about both plots? Why are they different?
5. Try scaling `Gr_Liv_Area` to have a maximum of 1 using `Gr_Liv_Area = rescale(Gr_Liv_Area, to = c(0, 1))` etc. What changes about both plots? Why?
6. Try adding scaling for `Latitude` (but not `Longitude`). What changes and why?
7. Now add scaling for for `Longitude`. What changes and why?
8. Try changing the maximum

Do the patterns captured by these clusters also happen to relate to sale price?

```
ames_with_clusters %>%
  ggplot(aes(x = Sale_Price, y = cluster)) + geom_boxplot()
```



Appendix

```
include_graphics("img/sphx_glr_plot_cluster_comparison_001.png")
```

```
#data(ames, package = "modeldata")
ames <- AmesHousing::make_ames()
ames_all <- ames %>%
  filter(Gr_Liv_Area < 4000, Sale_Condition == "Normal") %>%
  mutate(across(where(is.integer), as.double)) %>%
  mutate(Sale_Price = Sale_Price / 1000)
rm(ames)
```