STAT 215A Fall 2017 Week 7

Rebecca Barter
10/10/2017

Midterm study notes

See the crude but surprisingly thorough notes posted in the github repo for the lab class “midterm / midterm_study.pdf”.

I apparently wrote these for myself back in 2014 when I was studying for the midterm (I learn by writing things down…)

Note:

- I think our midterm was much later in the semester - some topics will differ haven’t been covered yet.
- I make a lot of little notes to myself. Please ignore them. I’m terribly embarrassed.

Timeline for next few weeks

Week 8: Friday October 13

- Lab 2 peer reviews due
- Lab 3 released (short lab - 1 week)

Week 9: Thursday October 19

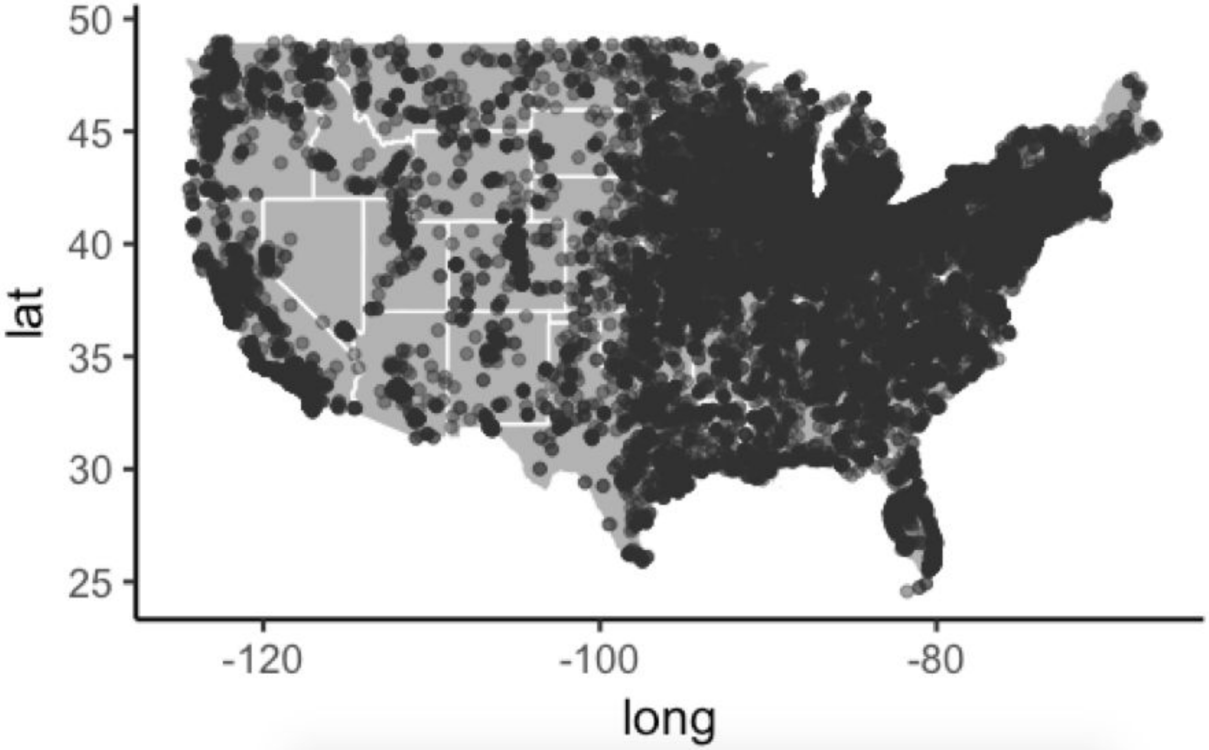
- Lab 3 due

Week 10: Thursday/Friday October 26/27

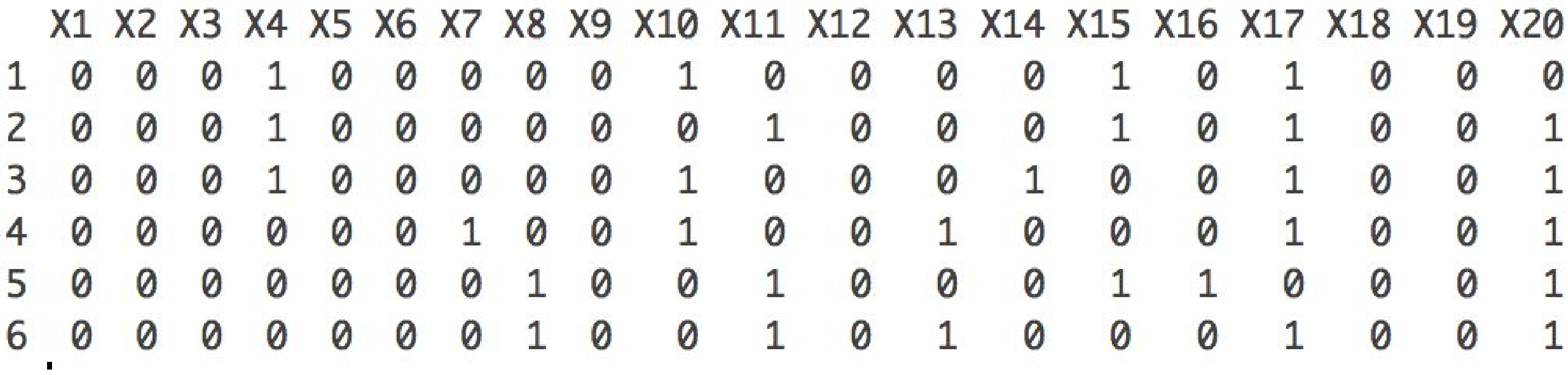
- Midterm (in-class)
- Lab 3 peer reviews due
- Lab 4 released (group project)

Lab 2 finale

Observational unit: individual person



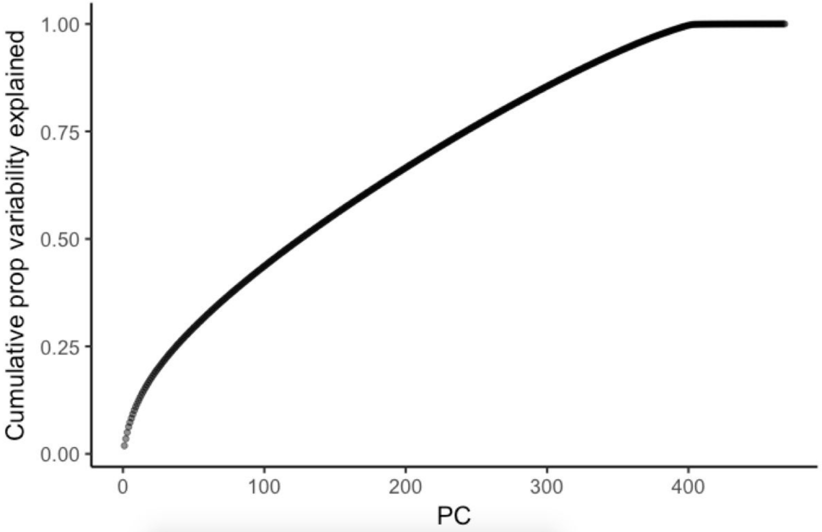
Observational unit: individual person



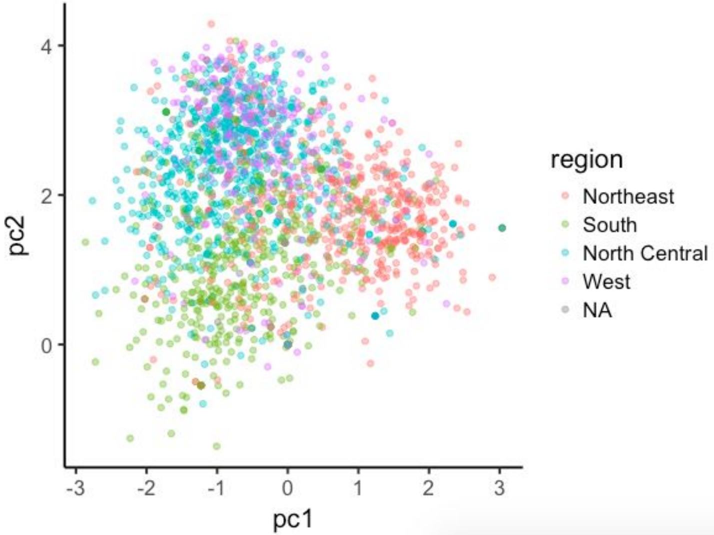
```
  X1 X2 X3 X4 X5 X6 X7 X8 X9 X10 X11 X12 X13 X14 X15 X16 X17 X18 X19 X20
1  0  0  0  1  0  0  0  0  0   1   0   0   0   0   1   0   1   0   0   0
2  0  0  0  1  0  0  0  0  0   0   1   0   0   0   1   0   1   0   0   1
3  0  0  0  1  0  0  0  0  0   1   0   0   0   1   0   0   1   0   0   1
4  0  0  0  0  0  0  1  0  0   1   0   0   1   0   0   0   1   0   0   1
5  0  0  0  0  0  0  0  1  0   0   1   0   0   0   1   1   0   0   0   1
6  0  0  0  0  0  0  0  1  0   0   1   0   1   0   0   0   1   0   0   1
```

Observational unit: individual person

Scree-plot



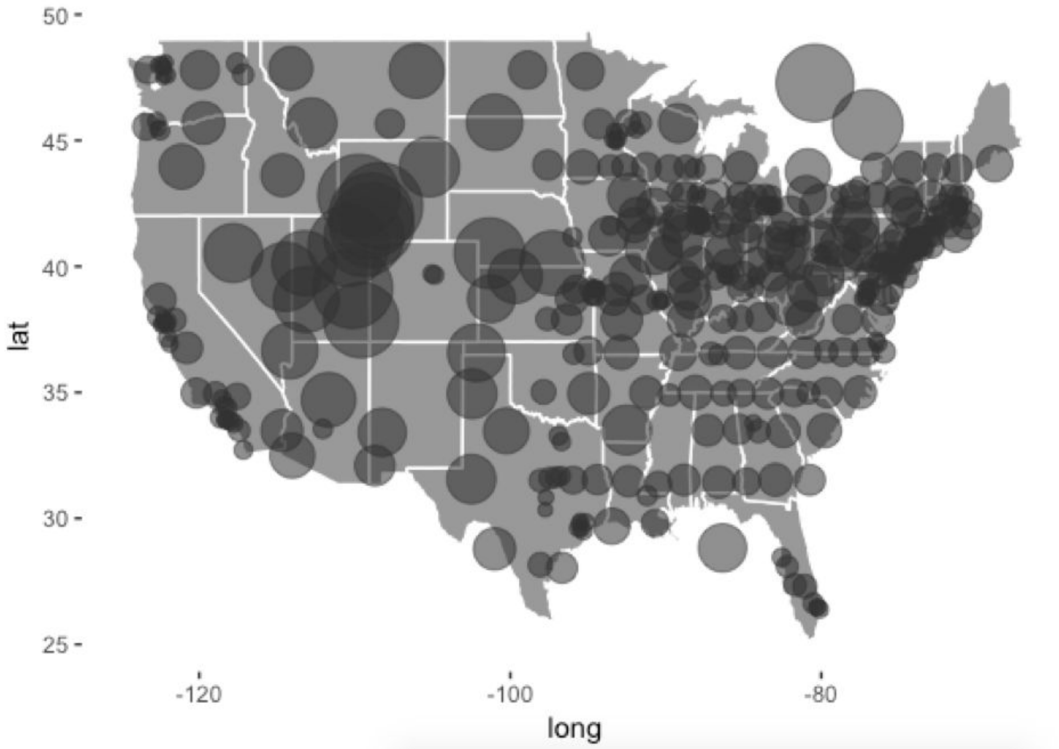
PCA-projection



Observational unit: group by lat/long bin

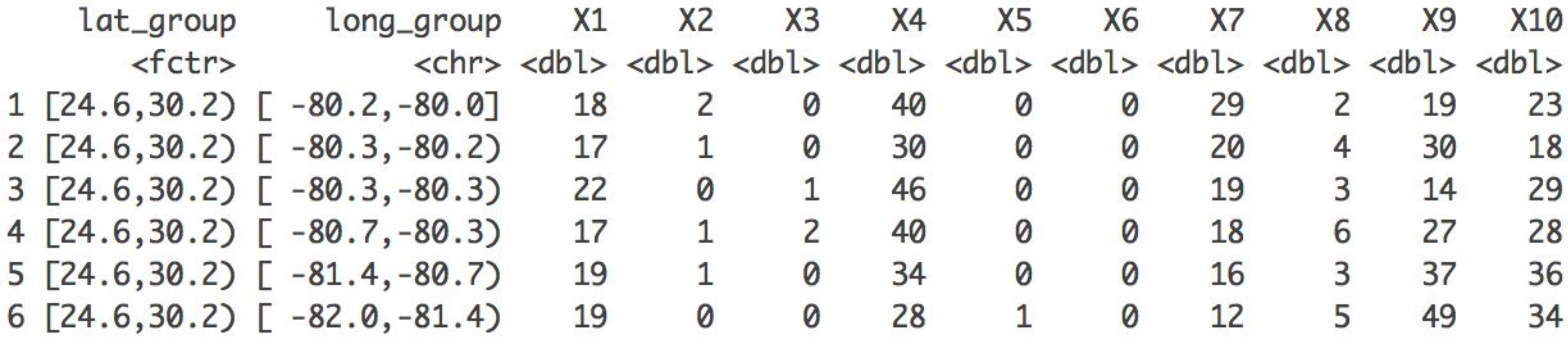
- **Bin by latitude** and within each latitude **bin by longitude**

- Bins are chosen so that each bin has approx the same number of people in it



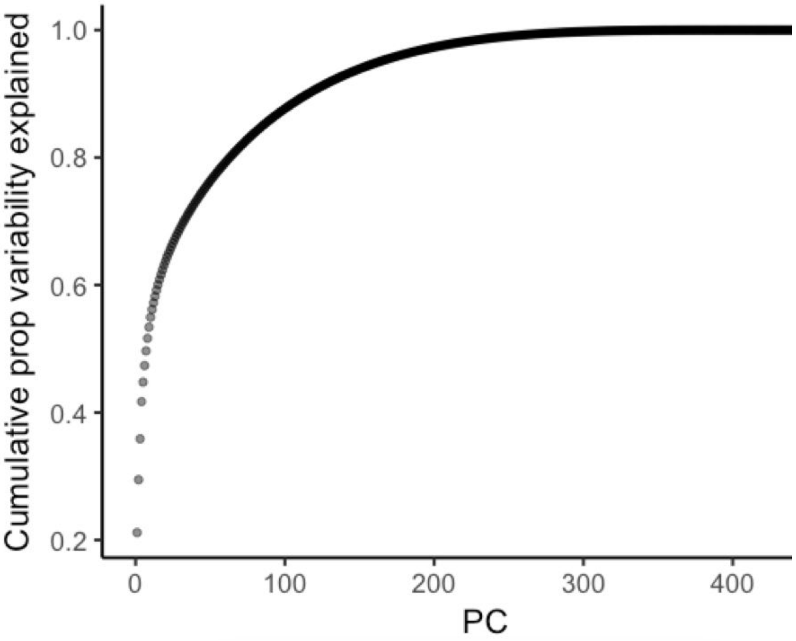
The size of each point is prop. to the geographical size of the corresponding bin

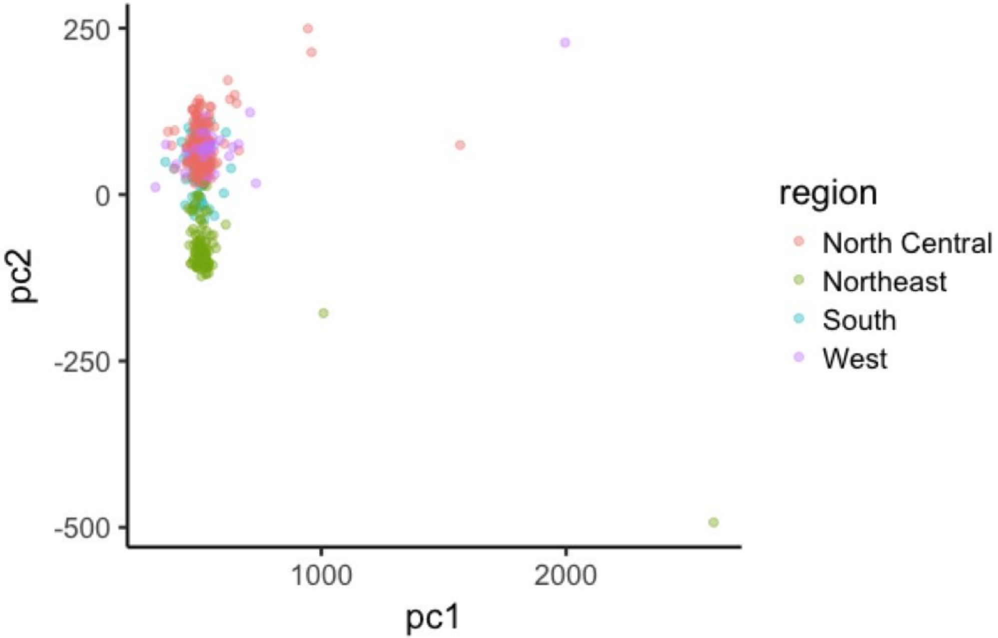
Observational unit: group by lat/long bin



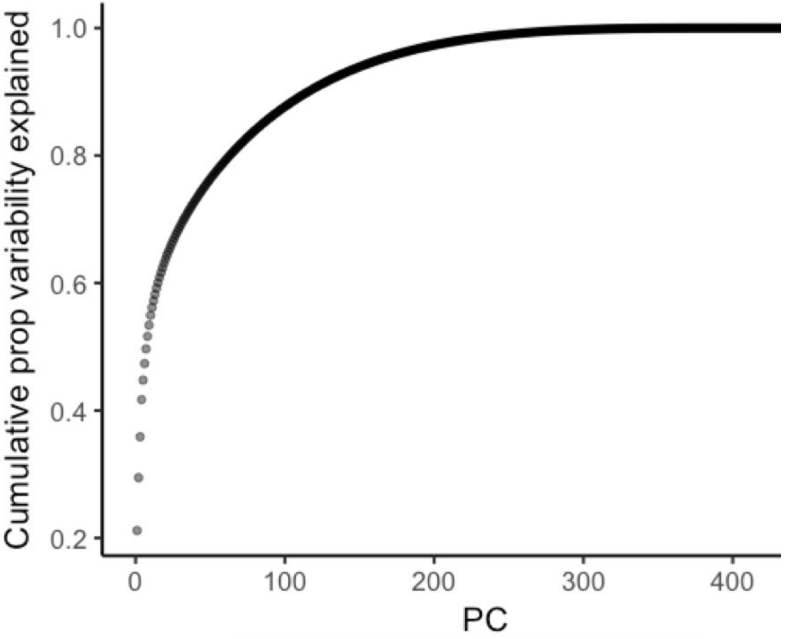
```
    lat_group       long_group    X1    X2    X3    X4    X5    X6    X7    X8    X9   X10
       <fctr>            <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 [24.6,30.2) [ -80.2,-80.0]    18     2     0    40     0     0    29     2    19    23
2 [24.6,30.2) [ -80.3,-80.2)    17     1     0    30     0     0    20     4    30    18
3 [24.6,30.2) [ -80.3,-80.3)    22     0     1    46     0     0    19     3    14    29
4 [24.6,30.2) [ -80.7,-80.3)    17     1     2    40     0     0    18     6    27    28
5 [24.6,30.2) [ -81.4,-80.7)    19     1     0    34     0     0    16     3    37    36
6 [24.6,30.2) [ -82.0,-81.4)    19     0     0    28     1     0    12     5    49    34
```

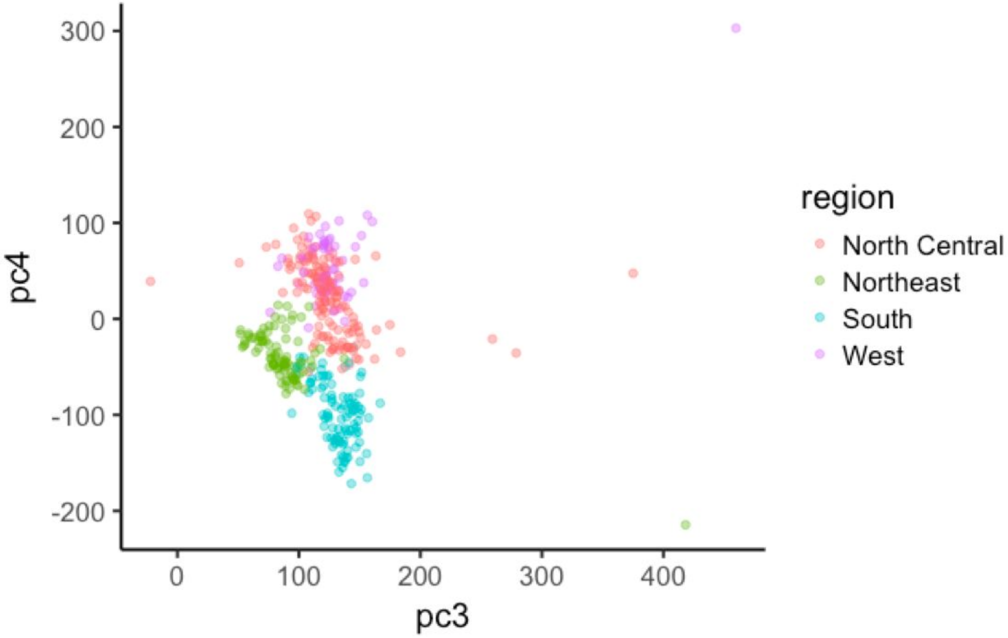
Observational unit: group by lat/long bin





Observational unit: group by lat/long bin





Stability

Stability: two types of questions

Computational stability

Stability: two types of questions

Computational stability

If I re-run the (possibly stochastic) algorithm again (possibly tweaking parameters) on the same data, do I get the same results?

Stability: two types of questions

Computational stability

If I re-run the (possibly stochastic) algorithm again (possibly tweaking parameters) on the same data, do I get the same results?

Generalization stability

Stability: two types of questions

Computational stability

If I re-run the (possibly stochastic) algorithm again (possibly tweaking parameters) on the same data, do I get the same results?

Generalization stability

If I re-run the algorithm again on a **new sample of data points from the same source**, do I get the same results?

Stability: two types of questions

Computational stability

If I re-run the (possibly stochastic) algorithm again (possibly tweaking parameters) on the same data, do I get the same results?

Asking about the randomness in the algorithm...

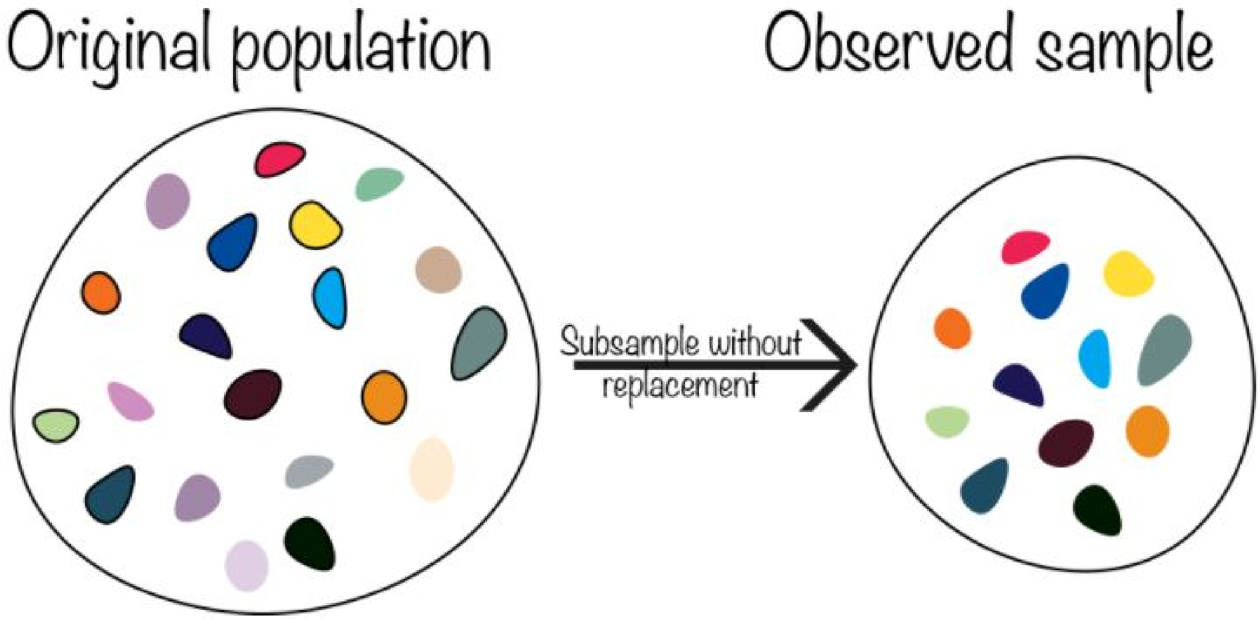
Generalization stability

If I re-run the algorithm again on a **new sample of data points from the same source**, do I get the same results?

Asking about randomness in the data...

Generalization stability sampling methods

The purpose of sampling methods is to simulate sampling procedure from the original population

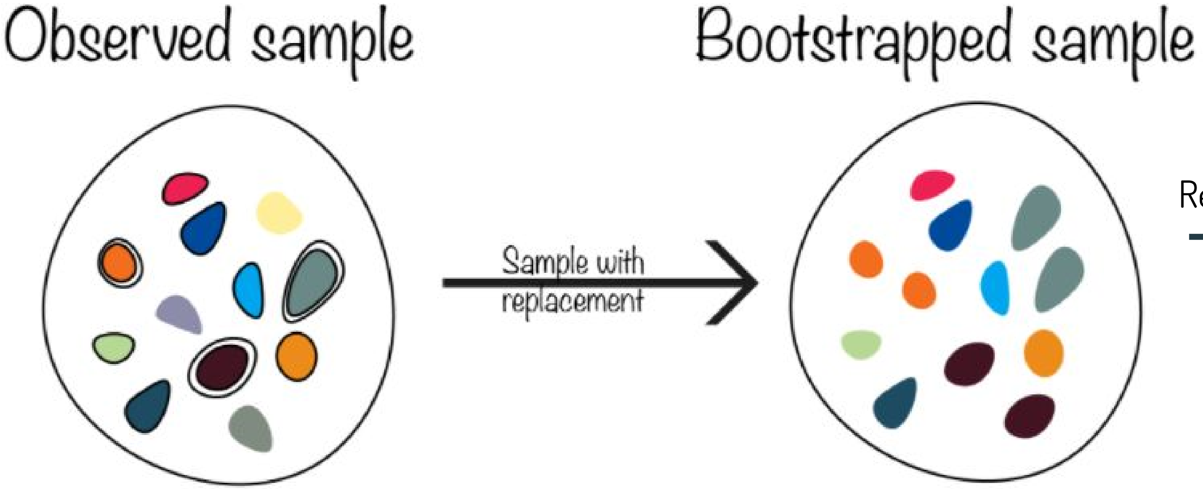


Do some stats →

Draw some conclusions (clustering, predictions, etc)

Bootstrap (non-parametric)

- sample with replacement
    - Repeat a pre-specified number of times (e.g. 1000)
- the bootstrap sample has the same sample size as the observed sample
- random sampling

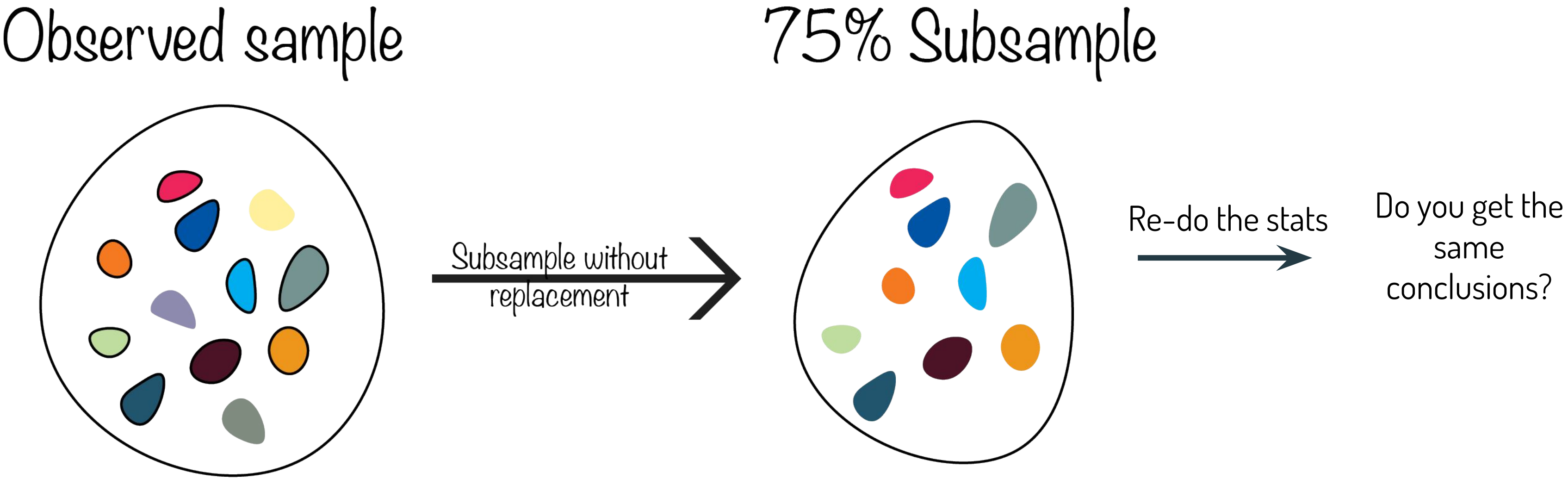


Re-do the stats →

Do you get the same conclusions?

Subsampling

- sample without replacement
  - Repeat a pre-specified number of times (e.g. 1000)
- the subsample has the a smaller sample size than the observed sample
- random sampling



Observed sample

75% Subsample

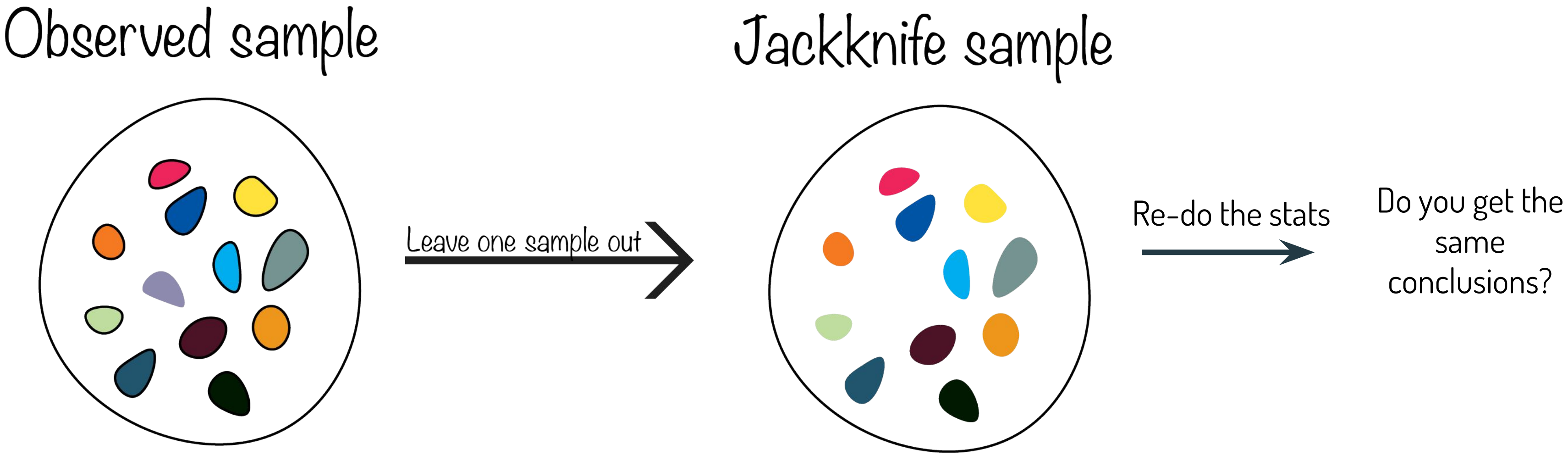
Subsample without replacement

Re-do the stats

Do you get the same conclusions?

Jackknife resampling

- obtain a subsample containing all but one of the data points
    - Repeat for all possible excluded data points
- the subsample has the a smaller sample size than the observed sample
- non-random sampling



Observed sample

Leave one sample out

Jackknife sample

Re-do the stats

Do you get the same conclusions?

Resampling techniques

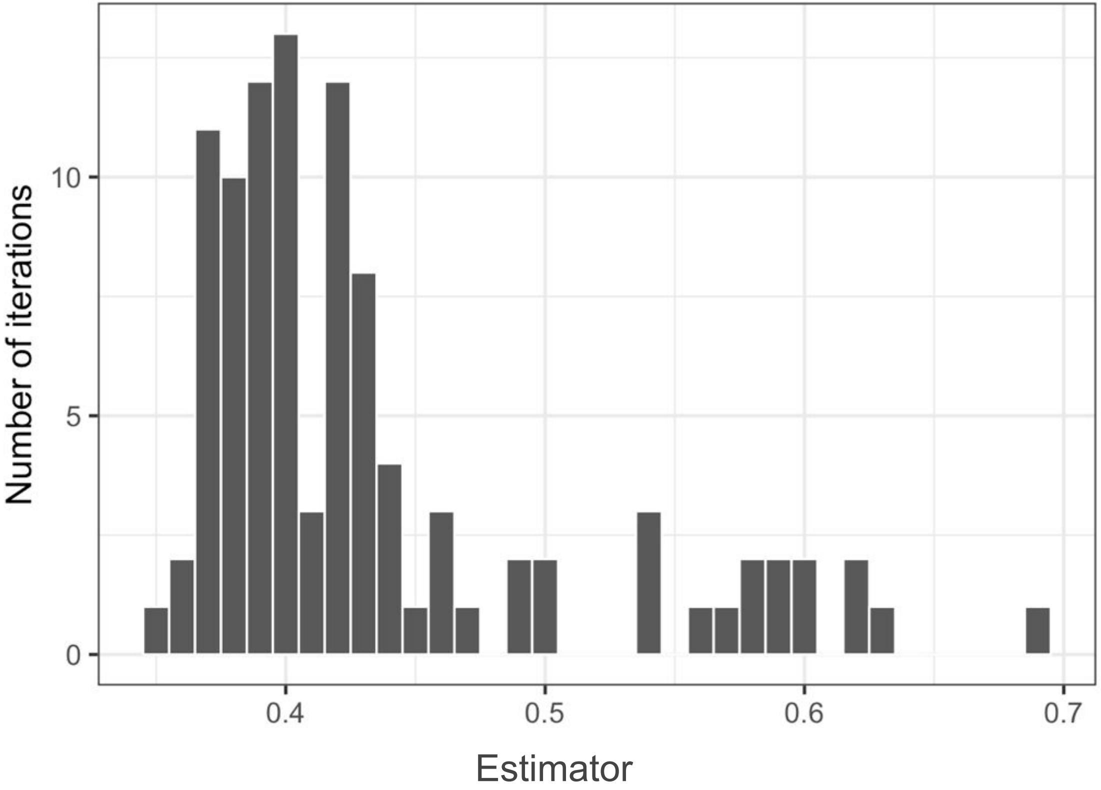
At the end of the day, no matter what resampling approach you use, you will have many versions of a particular estimator.

You can use these different versions of the estimate to approximate its distribution as if you had re-drawn samples from the original population.

Resampling techniques

The estimator is a random variable

- this is an empirical estimate of its distribution drawn from 100 bootstrapped samples



Question:

Which resampling method should you use?

Question:

Which resampling method should you use?

Answer:

???

Question:

How are these methods related to cross-validation?

Question:

How are these methods related to cross-validation?

Answer:

In **CV**, you build a model using the sampled data and evaluate the model using the left-out data.

In **subsampling/bootstrapping/etc**, you re-calculate statistics on the sampled data and ignore the left-out data entirely

Stability for clustering: an example (wines_stability.Rmd)

Remember the wine clustering example from a few weeks ago?

Let’s evaluate the stability of the clusters using these techniques!

1. Test algorithmic stability: re-generate the clusters using the same dataset
   a. Compare the groupings obtained (how?)

2. Test generalization stability: re-generate the clusters using different datasets (bootstrap, subsample, jackknife)
   a. Compare the groupings obtained (how?)