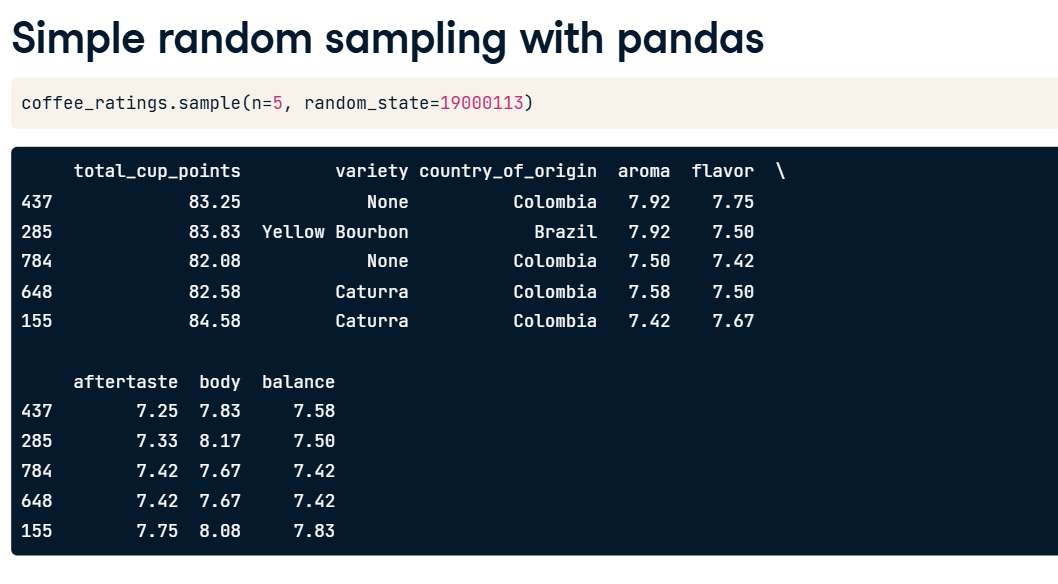
**Simple random and systematic sampling**

There are several methods of sampling from a population. In this video, we'll look at simple random sampling and systematic random sampling.

**2. Simple random sampling**

Simple random sampling works like a raffle or lottery. We start with our population of raffle tickets or lottery balls and randomly pick them out one at a time.



**3. Simple random sampling of coffees**

00:21 - 00:53

In our coffee ratings dataset, instead of raffle tickets or lottery balls, the population consists of coffee varieties. To perform simple random sampling, we take some at random, one at a time. Each coffee has the same chance as any other of being picked. When using this technique, sometimes we might end up with two coffees that were next to each other in the dataset, and sometimes we might end up with large areas of the dataset that were not selected from at all.

**4. Simple random sampling with pandas**

00:53 - 01:22

We've already seen how to do simple random sampling with pandas. We call dot-sample and set n to the size of the sample. We can also set the seed using the random\_state argument to generate reproducible results, just like we did pseudo-random number generation. Previously, by not setting random\_state when sampling, our code would generate a different random sample each time it was run.

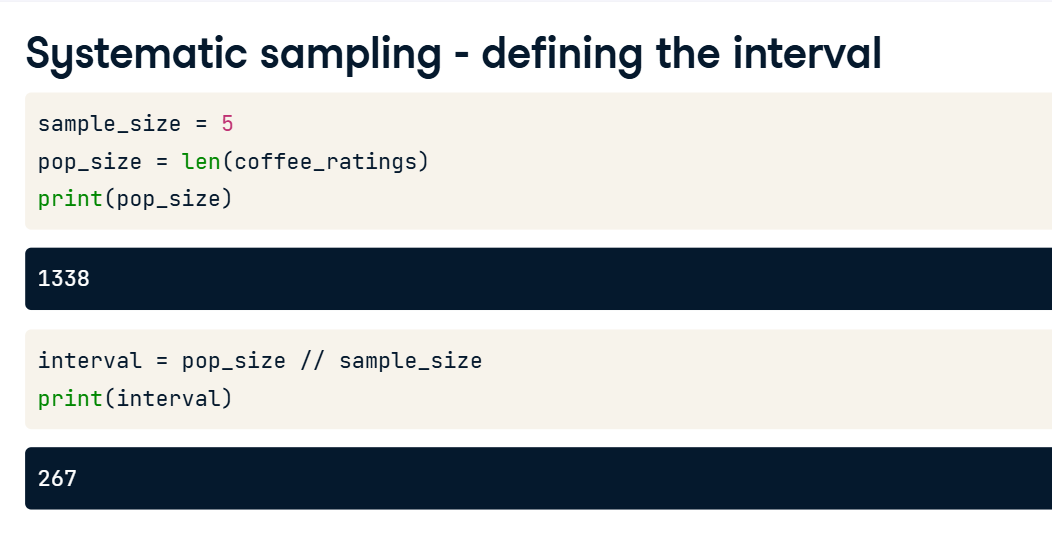
**5. Systematic sampling**

01:22 - 01:37

Another sampling method is known as systematic sampling. This samples the population at regular intervals. Here, looking from top to bottom and left to right within each row, every fifth coffee is sampled.

**6. Systematic sampling - defining the interval**

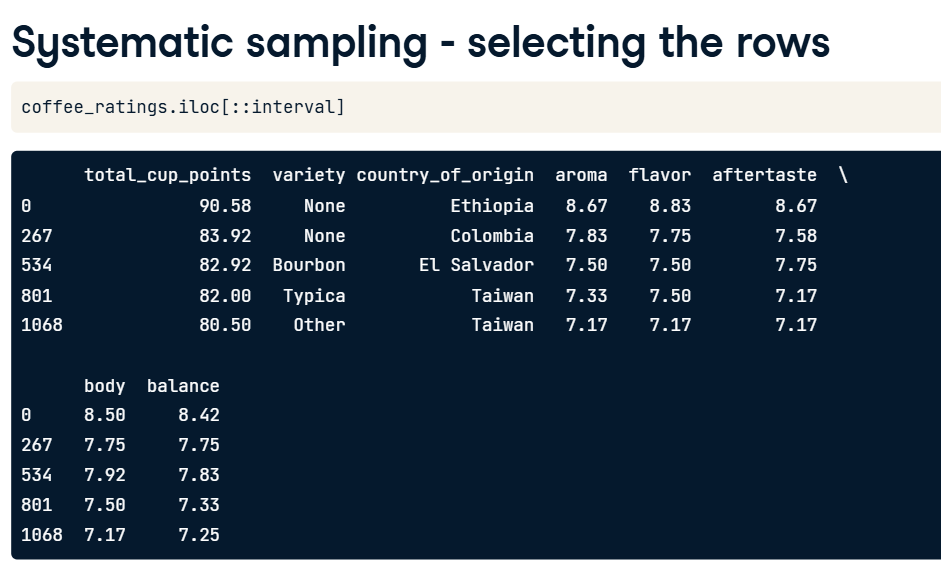
Systematic sampling with pandas is slightly trickier than simple random sampling. The tricky part is determining how big the interval between each row should be for a given sample size. Suppose we want a sample size of five coffees. The population size is the number of rows in the whole dataset, and in this case, it's one thousand three hundred and thirty-eight. The interval is the population size divided by the sample size, but because we want the answer to be an integer, we perform integer division with two forward slashes. This is like standard division but discards any fractional part. One-three-three-eight divided by five is actually two hundred and sixty-seven-point-six, and discarding the fractional part leaves two hundred and sixty-seven. Thus, to get a systematic sample of five coffees, we will select every two hundred sixty-seventh coffee in the dataset.



**7. Systematic sampling - selecting the rows**

02:39 - 03:02

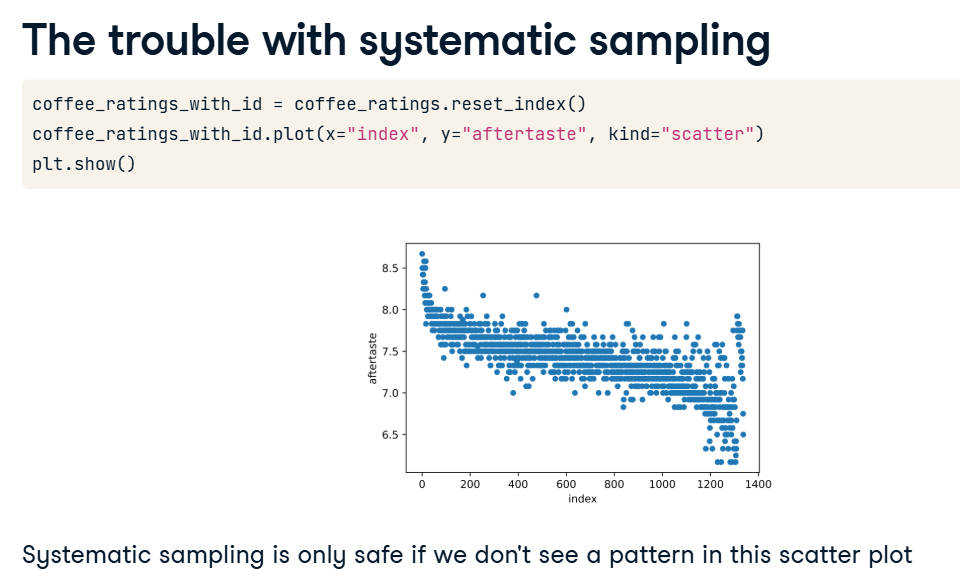
To select every two hundred and sixty-seventh row, we call dot-iloc on coffee\_ratings and pass double-colons and the interval, which is 267 in this case. Double-colon interval tells pandas to select every two hundred and sixty-seventh row from zero to the end of the DataFrame.



**8. The trouble with systematic sampling**

03:02 - 03:43

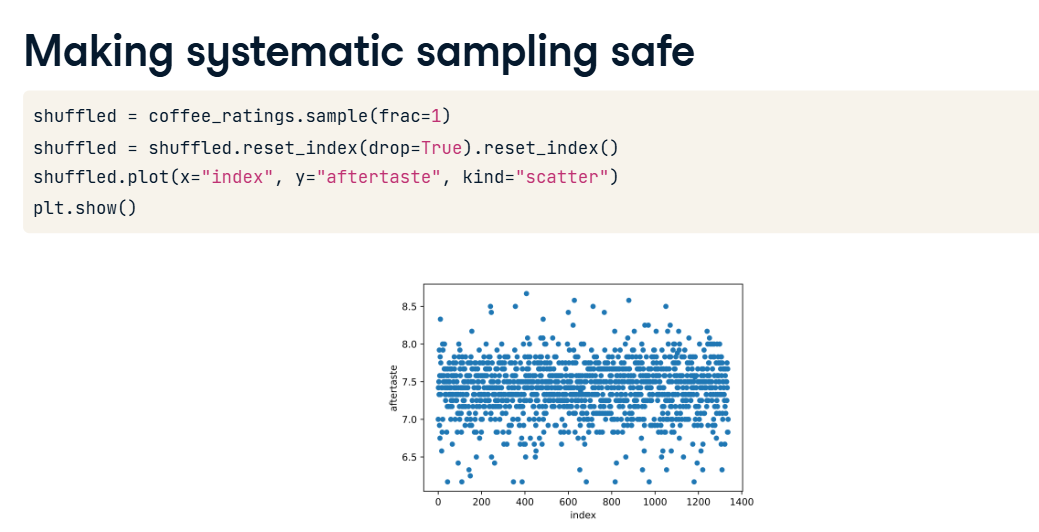
There is a problem with systematic sampling, though. Suppose we are interested in statistics about the aftertaste attribute of the coffees. To examine this, first, we use reset\_index to create a column of index values in our DataFrame that we can plot. Plotting aftertaste against index shows a pattern. Earlier rows generally have higher aftertaste scores than later rows. This introduces bias into the statistics that we calculate. In general, it is only safe to use systematic sampling if a plot like this has no pattern; that is, it just looks like noise.



**9. Making systematic sampling safe**

03:43 - 04:45

To ensure that systematic sampling is safe, we can randomize the row order before sampling. dot-sample has an argument named frac that lets us specify the proportion of the dataset to return in the sample, rather than the absolute number of rows that n specifies. Setting frac to one randomly samples the whole dataset. In effect, this randomly shuffles the rows. Next, the indices need to be reset so that they go in order from zero again. Specifying drop equals True clears the previous row indexes, and chaining to another reset\_index call creates a column containing these new indexes. Redrawing the plot with the shuffled dataset shows no pattern between aftertaste and index. This is great, but note that once we've shuffled the rows, systematic sampling is essentially the same as simple random sampling.



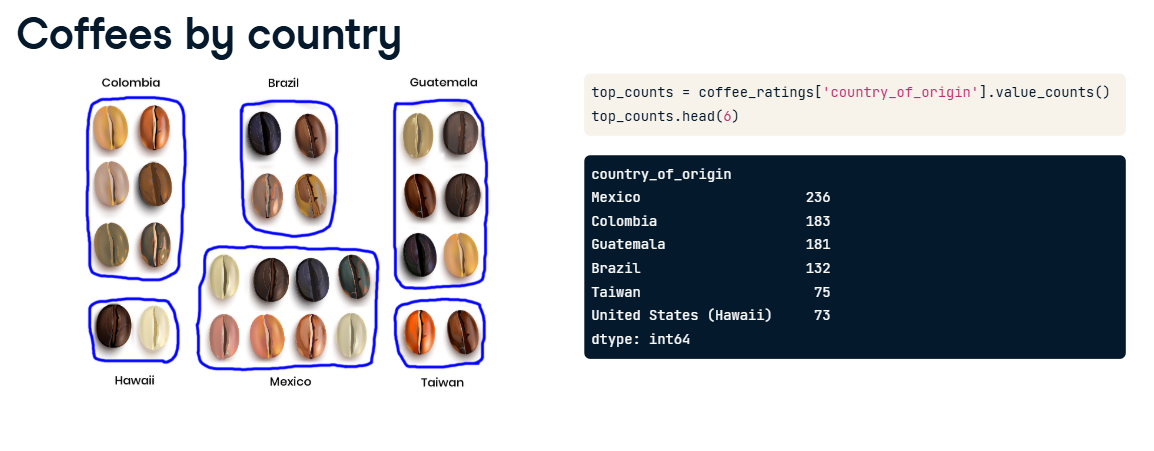
**Stratified and weighted random sampling**

Stratified sampling is a technique that allows us to sample a population that contains subgroups.

**2. Coffees by country**

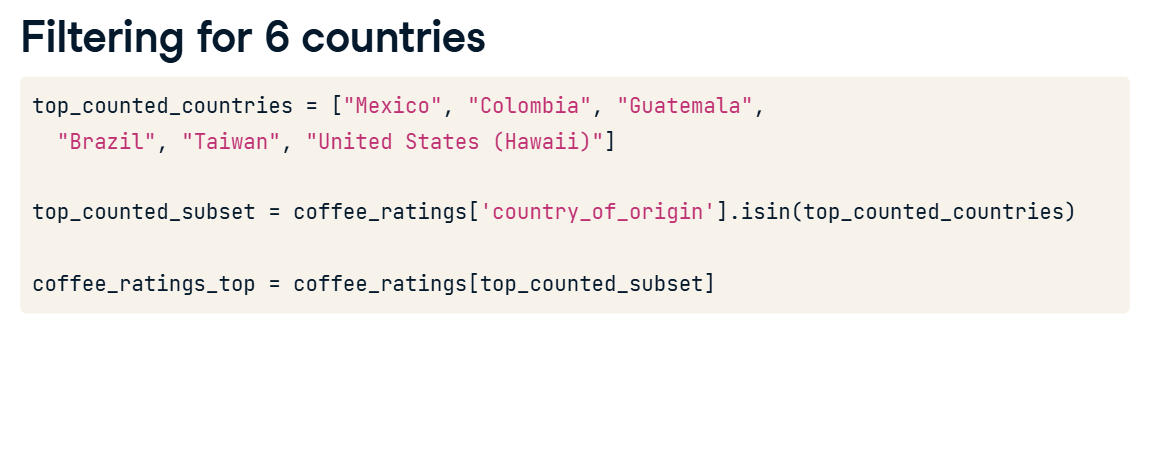
For example, we could group the coffee ratings by country. If we count the number of coffees by country using the value\_counts method, we can see that these six countries have the most data.

1. 1 The dataset lists Hawaii and Taiwan as countries for convenience, as they are notable coffee-growing regions.



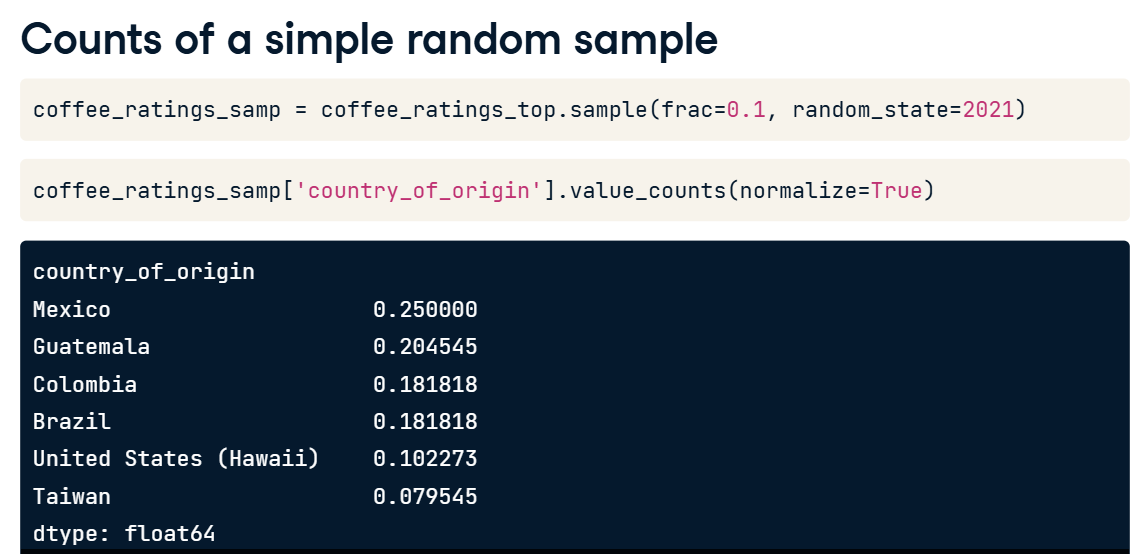
**3. Filtering for 6 countries**

To make it easier to think about sampling subgroups, let's limit our analysis to these six countries. We can use the dot-isin method to filter the population and only return the rows corresponding to these six countries. This filtered dataset is stored as coffee\_ratings\_top.



**4. Counts of a simple random sample**

Let's take a ten percent simple random sample of the dataset using dot-sample with frac set to zero-point-one. We also set the random\_state argument to ensure reproducibility. As with the whole dataset, we can look at the counts for each country. To make comparisons easier, we set normalize to True to convert the counts into a proportion, which shows what proportion of coffees in the sample came from each country.



**5. Comparing proportions**

Here are the proportions for the population and the ten percent sample side by side. Just by chance, in this sample, Taiwanese coffees form a disproportionately low percentage. The different makeup of the sample compared to the population could be a problem if we want to analyze the country of origin, for example.

**6. Proportional stratified sampling**

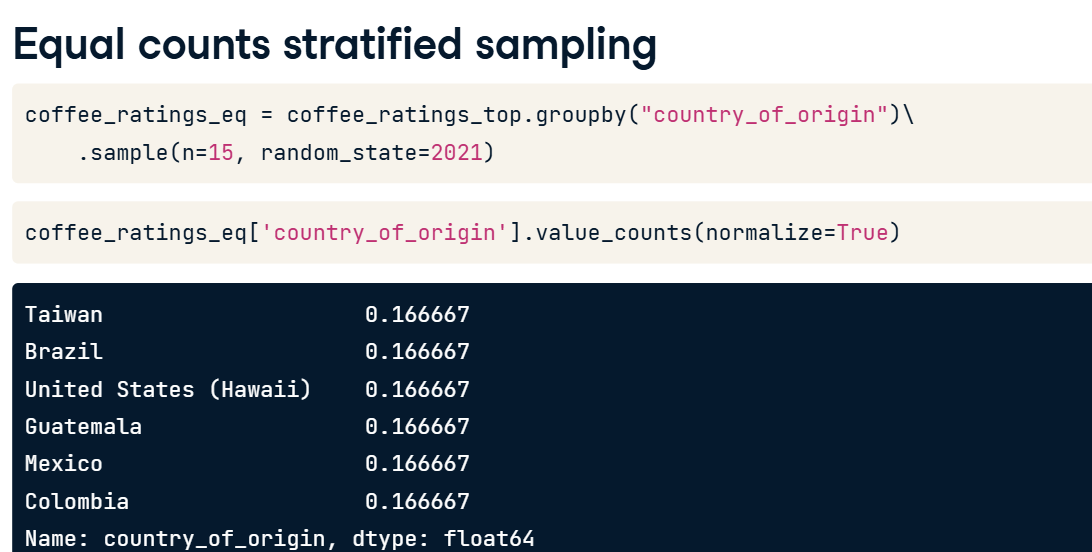
If we care about the proportions of each country in the sample closely matching those in the population, then we can group the data by country before taking the simple random sample. Note that we used the Python line continuation backslash here, which can be useful for breaking up longer chains of pandas code like this. Calling the dot-sample method after grouping takes a simple random sample within each country. Now the proportions of each country in the stratified sample are much closer to those in the population.



**7. Equal counts stratified sampling**

02:05 - 02:32

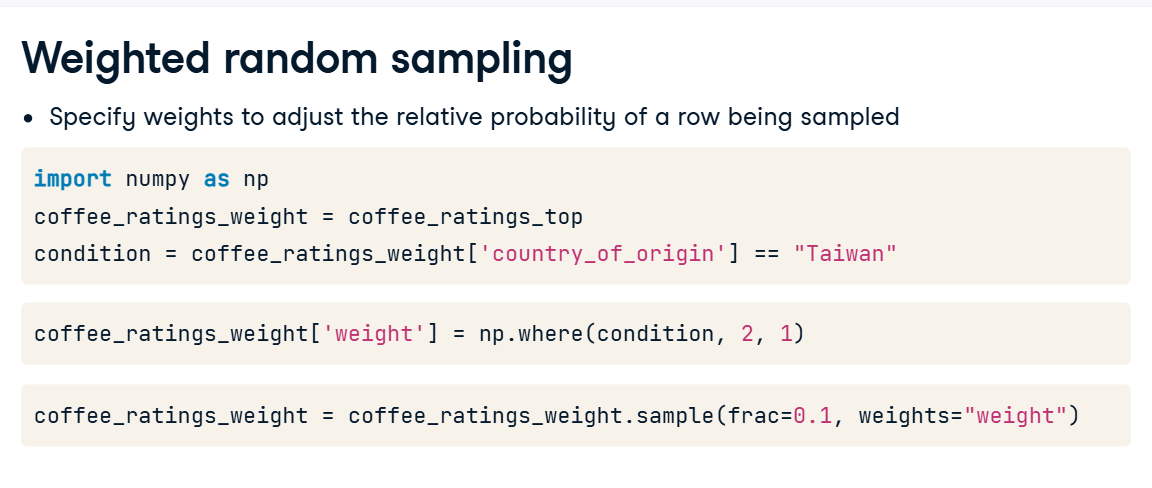
One variation of stratified sampling is to sample equal counts from each group, rather than an equal proportion. The code only has one change from before. This time, we use the n argument in dot-sample instead of frac to extract fifteen randomly-selected rows from each country. Here, the resulting sample has equal proportions of one-sixth from each country.



**8. Weighted random sampling**

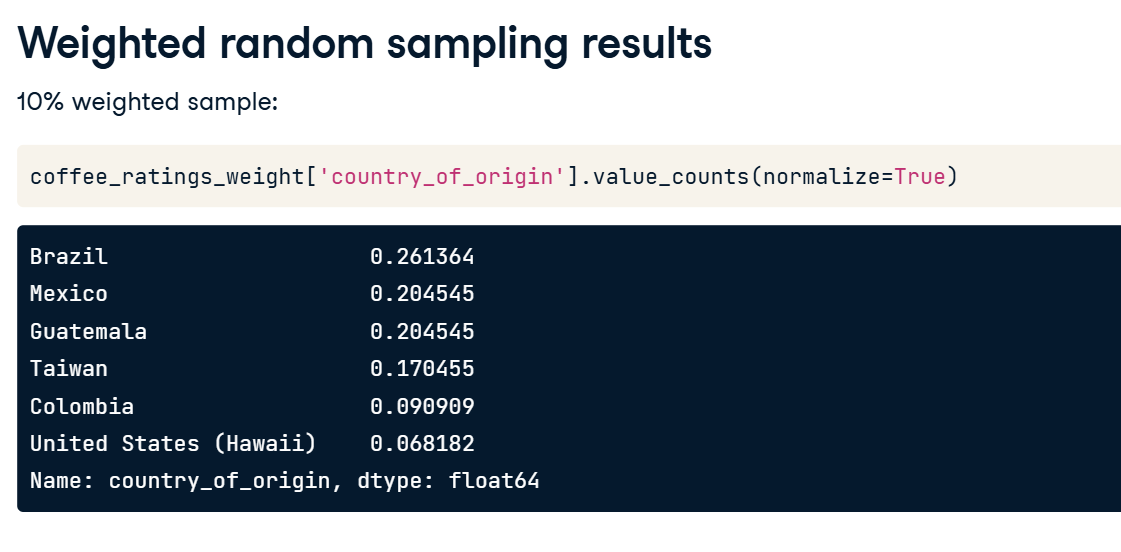
02:32 - 03:26

A close relative of stratified sampling that provides even more flexibility is weighted random sampling. In this variant, we create a column of weights that adjust the relative probability of sampling each row. For example, suppose we thought that it was important to have a higher proportion of Taiwanese coffees in the sample than in the population. We create a condition, in this case, rows where the country of origin is Taiwan. Using the where function from NumPy, we can set a weight of two for rows that match the condition and a weight of one for rows that don't match the condition. This means when each row is randomly sampled, Taiwanese coffees have two times the chance of being picked compared to other coffees. When we call dot-sample, we pass the column of weights to the weights argument.



**9. Weighted random sampling results**

Here, we can see that Taiwan now contains seventeen percent of the sampled dataset, compared to eight-point-five percent in the population. This sort of weighted sampling is common in political polling, where we need to correct for under- or over-representation of demographic groups



**1. Cluster sampling**

One problem with stratified sampling is that we need to collect data from every subgroup. In cases where collecting data is expensive, for example, when we have to physically travel to a location to collect it, it can make our analysis prohibitively expensive. There's a cheaper alternative called cluster sampling.

**2. Stratified sampling vs. cluster sampling**

The stratified sampling approach was to split the population into subgroups, then use simple random sampling on each of them. Cluster sampling means that we limit the number of subgroups in the analysis by picking a few of them with simple random sampling. We then perform simple random sampling on each subgroup as before.

**3. Varieties of coffee**

Let's return to the coffee dataset and look at the varieties of coffee. In this image, each bean represents the whole subgroup rather than an individual coffee, and there are twenty-eight of them. To extract unique varieties, we use the dot-unique method. This returns an array, so wrapping it in the list function creates a list of unique varieties. Let's suppose that it's expensive to work with all of the different varieties. Enter cluster sampling.

\

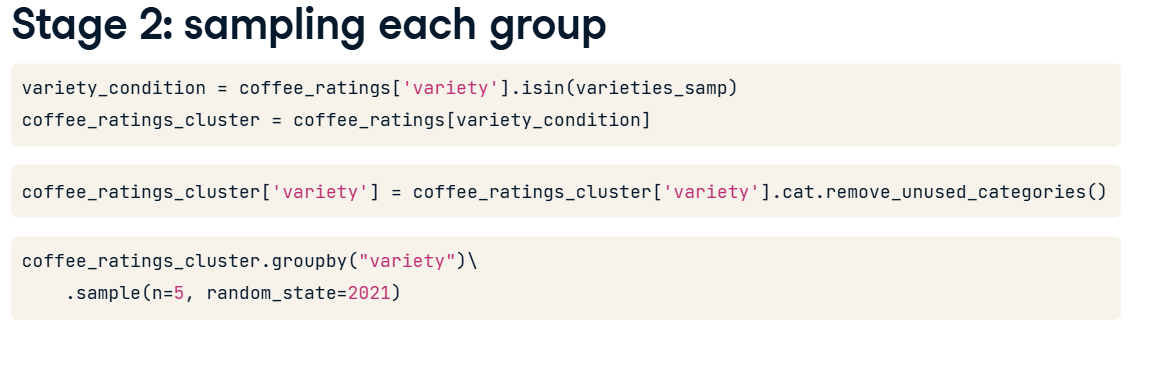
**4. Stage 1: sampling for subgroups**

The first stage of cluster sampling is to randomly cut down the number of varieties, and we do this by randomly selecting them. Here, we've used the random-dot-sample function from the random package to get three varieties, specified using the argument k.



**5. Stage 2: sampling each group**

The second stage of cluster sampling is to perform simple random sampling on each of the three varieties we randomly selected. We first filter the dataset for rows where the variety is one of the three selected, using the dot-isin method. To ensure that the isin filtering removes levels with zero rows, we apply the cat-dot-remove\_unused\_categories method on the Series of focus, which is variety here. If we exclude this method, we might receive an error when sampling by variety level. The pandas code is the same as for stratified sampling. Here, we've opted for equal counts sampling, with five rows from each remaining variety.



**6. Stage 2 output**

Here's the first few columns of the result. Notice that there are the fifteen rows, which we'd expect from sampling five rows from three varieties.



**7. Multistage sampling**

Note that we had two stages in the cluster sampling. We randomly sampled the subgroups to include, then we randomly sampled rows from those subgroups. Cluster sampling is a special case of multistage sampling. It's possible to use more than two stages. A common example is national surveys, which can include several levels of administrative regions, like states, counties, cities, and neighborhoods.

**Comparing sampling methods**

Let's review the various sampling techniques we learned about.

**2. Review of sampling techniques - setup**

00:04 - 00:18

For convenience, we'll stick to the six countries with the most coffee varieties that we used before. This corresponds to eight hundred and eighty rows and eight columns, retrieved using the dot-shape attribute.

**3. Review of simple random sampling**

00:18 - 00:42

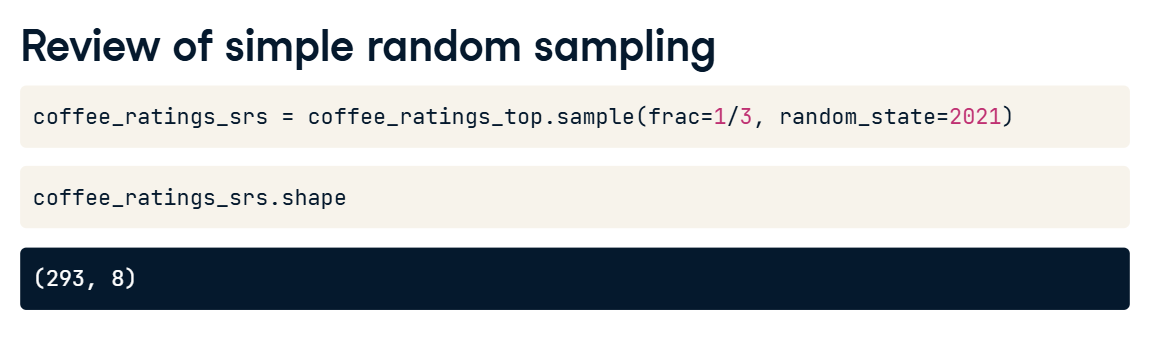
Simple random sampling uses dot-sample with either n or frac set to determine how many rows to pseudo-randomly choose, given a seed value set with random\_state. The simple random sample returns two hundred and ninety-three rows because we specified frac as one-third, and one-third of eight hundred and eighty is just over two hundred and ninety-three.



**4. Review of stratified sampling**

00:42 - 00:59

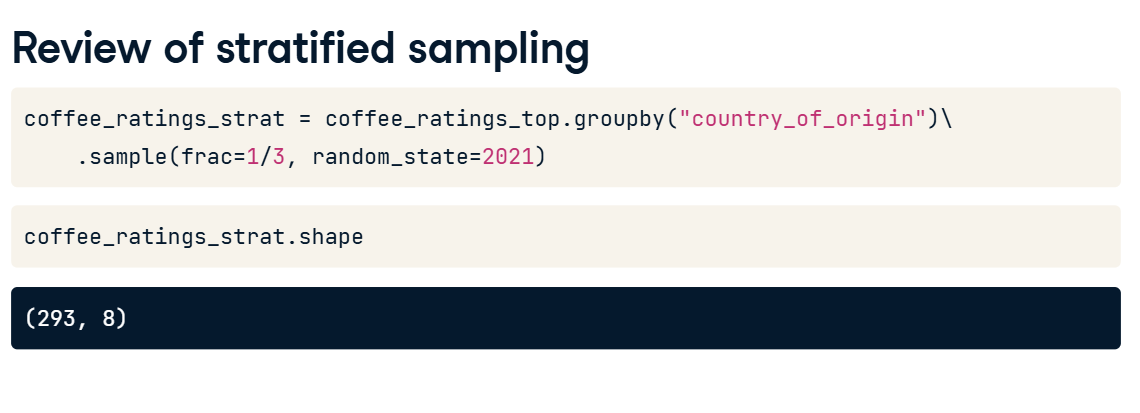
Stratified sampling groups by the country subgroup before performing simple random sampling on each subgroup. Given each of these top countries have quite a few rows, stratifying produces the same number of rows as the simple random sample.



**5. Review of cluster sampling**

00:59 - 01:29

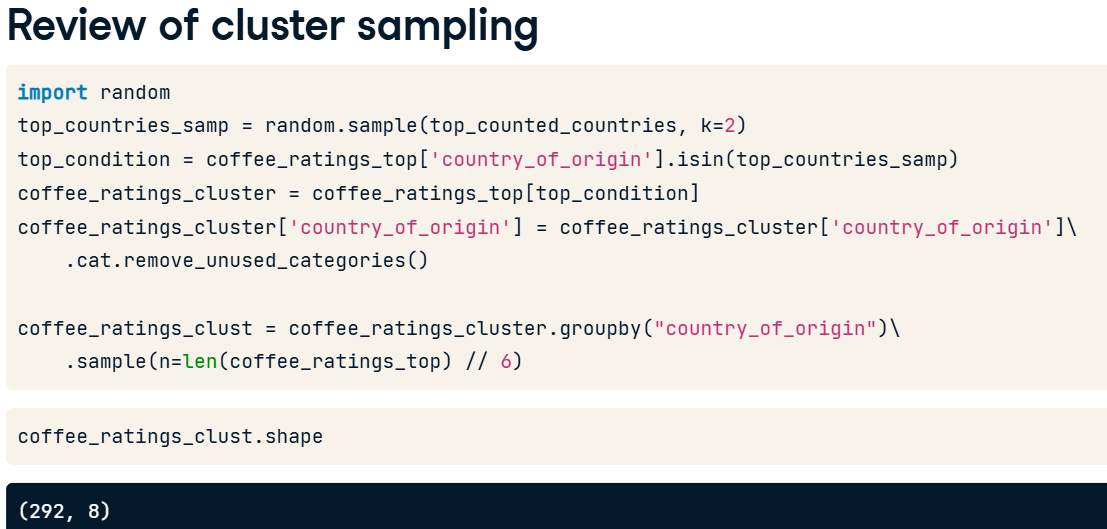
In the cluster sample, we've used two out of six countries to roughly mimic frac equals one-third from the other sample types. Setting n equal to one-sixth of the total number of rows gives roughly equal sample sizes in each of the two subgroups. Using dot-shape again, we see that this cluster sample has close to the same number of rows: two-hundred-ninety-two compared to two-hundred-ninety-three for the other sample types.



**6. Calculating mean cup points**

01:29 - 02:23

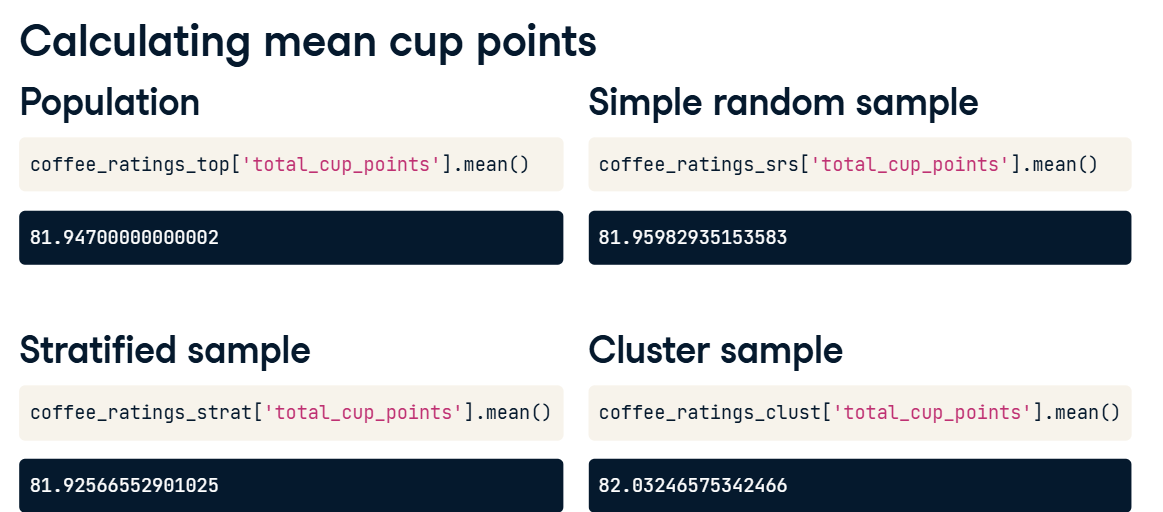
Let's calculate a population parameter, the mean of the total cup points. When the population parameter is the mean of a field, it's often called the population mean. Remember that in real-life scenarios, we typically wouldn't know what the population mean is. Since we have it here, though, we can use this value of eighty-one-point-nine as a gold standard to measure against. Now we'll calculate the same value using each of the sampling techniques we've discussed. These are point estimates of the mean, often called sample means. The simple and stratified sample means are really close to the population mean. Cluster sampling isn't quite as close, but that's typical. Cluster sampling is designed to give us an answer that's almost as good while using less data.’



**7. Mean cup points by country: simple random**

02:23 - 02:44

Here's a slightly more complicated calculation of the mean total cup points for each country. We group by country before calculating the mean to return six numbers. So how do the numbers from the simple random sample compare? The sample means are pretty close to the population means.



**8. Mean cup points by country: stratified**

02:44 - 02:53

The same is true of the sample means from the stratified technique. Each sample mean is relatively close to the population mean.

**9. Mean cup points by country: cluster**

02:53 - 03:13

With cluster sampling, while the sample means are pretty close to the population means, the obvious limitation is that we only get values for the two countries that were included in the sample. If the mean cup points for each country is an important metric in our analysis, cluster sampling would be a bad idea.





