DOS Problem Set 3

You should complete the “Sampling in Python DataCamp” and watch the class recording from 1/22 before attempting these problems. Note that the slide deck is also available in BB. In addition, all problem sets from that course can be found here.

> These datasets are in ‘feather’ format so you’ll need to import pyarrow in order to work with them. You can use *read_feather* to load the datasets into a dataframe.

```
df_spot = pd.read_feather('https://raw.githubusercontent.com/ncrowder/datacamp/main/sampling_in_python/spotify_2000_2020.feather')
```

1. Generate a samp20 dataframe from a random sample of 20 rows.
2. Generate a samp2per dataframe from a random sample of 2% of the rows.
3. Generate a random sample of 1% of the ‘duration_ms’ column and compute the mean of the sample and the mean of the full dataset. What is the difference in milliseconds and as a percentage error?
4. Sort the dataframe by ‘duration_ms’ using the *sort_values* method and then find the mean ‘energy’ of the top 50 rows. Compare this with the mean ‘energy’ of the full dataset. What is the difference?
5. Generate a histogram of the ‘energy’ from the top 50 rows after sorting as well as histogram of the full dataset’s ‘energy’ level. Comment on any perceived differences.
6. Using the original dataset, find the standard deviation of the ‘tempo’ of every 4th row.
7. What proportion of the dataset do the top 5 artists in the dataset represent? Use the *value_counts* method with the normalize argument. Then find the proportion of each of the top 5 artists if only looking at the rows representing one of these artists. **Hint: The proportions should add to 1 in the latter case, but not the former case.**
8. Generate a random sample of 5% of the artists from the dataset 2 ways: one that allows for repeated artists to show up and a second way where it is only possible that an artist shows up once in the sample. For the first way you can simply use the sample method off of the dataframe. For the second way you can use use the sample method off of the random package and pass a list of unique artists.
9. Using a dataset that only includes artists that show up at least 10 times, apply the *groupby* method on the ‘artists’ to take a sample of 10% of the rows from each artist. Use a random_state of 2026.
   Helper code to start (make sure you understand it):

```
counts = df_spot.value_counts('artists')

top_artists = counts[counts>=10].index
```

10. Using the result from #9, find the mean, min and max of the ‘loudness’ both for the total dataset, and then grouped by artist.
11. Generate a plot of sample_size vs. relative_error in the mean of ‘duration_ms’ for sample sizes ranging from 10 to 500. The relative_error is computed against the the true mean from the entire population.
12. Plot a sampling distribution of the mean of ‘duration_ms’ using 5000 samples of size 25 and of size 100.
13. Find the margin of error corresponding to a 95% CI for both sampling distributions in #12 (n=25 and n=100) using np.quantile.
14. Generate a histogram of the ‘loudness’ for the entire dataset. Does it appear normally distributed? Explain the shape in terms of symmetry and skewness.
15. Select a single sample of size 100 and estimate the standard error of the mean ‘loudness’ using the formula:

$$SE \approx sn \text{SE} \approx \frac{s}{\sqrt{n}} SE \approx \frac{s}{\sqrt{n}}$$

16. Generate 5000 bootstrap samples using your sample from #14 and compute the mean for each one. The bootstrap standard error is the sample standard deviation of the 5000 bootstrap statistics. Since the mean of this distribution is estimated from the data, use **ddof = 1**.
17. Calculate the difference between the theoretical SE from #14 (a result of the CLT) and the bootstrap SE from #15. **Hint: They should be very close.**
18. Instead of using the mean, let’s use the median or maximum (your choice) instead. Reproduce parts 14-16, bearing in mind that the difference should be much more dramatic, at least in terms of percent difference.
19. The reason the results are different is because the CLT formula for estimating SE can only be applied to means. Generate a histogram of the bootstrap from the median/max statistic (whichever you chose in #17). Does the result appear normal? Explain how this factors in to the difference observed when comparing the means SE vs the median/max SE.
20. Suppose we roll an 8-sided die that has the following sides = [1,2,2,3,3,4,7,9] five times and take the average of the rolls. Simulate doing this 10000 times and plot a sampling distribution that corresponds to this result.
21. Suppose now we roll the above 8-sided die along with a normal 6 sided fair die and take the average. Simulate doing this 10000 times and plot a sampling distribution that corresponds to this result.
22. Generate one single sample from #21 and find the margin of error (MOE) corresponding to a 95% confidence interval using the ppf function (thus using the fact that the sampling distribution will be normally distributed under the CLT) with loc equal to the mean of your single sample and standard error using the above formula for SE. A similar approach, but one using the bootstrap is shown on one of the final slides in the slide deck if you get stuck.

> Written with StackEdit.