Introduction to Data Science Portfolio

Week 3 – Intro to Summary Statistics

TianQiu

This week, we'll summarise the dataset and some of the core concepts it represents.

First, we're going to check out a dataset of 2019-2022 hot jitterbug music dataset. (FROM: https://www.kaggle.com/sveta151/datasets,Author: Sveta151) I'm going to preprocess the dataset after and integrate it into a DF.。

* Tag songs from different years in the dataset with their own year of popularity.
* Replace track\_name with a non-repeating year\_number.
* Remove irrelevant columns.
* Round all the values to nearest integer

I found that the same track\_name could be on the hit list in different years, so there were a lot of problems indexing the data behind it, to the point where the zscore calculation wouldn't bring the data into the df, due to the fact that the same song creates different zscores in different years, so I replaced the trackname with a pattern of non-repeating year\_serial numbers to avoid this problem.

After briefly processing the data, we can get a histogram of the distribution of tempo for all songs.

A graph of a graph

Description automatically generated with medium confidence

## Central Tendency

The MEAN, MEDIAN and MODE (most frequent values) of the data were calculated.

They were 122.3, 121.4, 120.0.

## Individual Genres

We can split up these tempo stats into different year and see how they vary between them.

1. We compare the difference in tempo between years by creating new arrays, giving them different colors and calculating the average tempo between years in the same bar chart.

A graph showing different colored rectangles

Description automatically generated

2.Draw the distribution of tempo for different years. It is possible to compare the distribution of tempo in different years.

A group of dots with numbers

Description automatically generated

3. Comparing the tempo distribution of hot music from different years.

A graph of different colored shapes

Description automatically generated with medium confidence

4. The median is a more accurate reflection of the dataset than the mean when there are more extremes, so we then attempted to calculate the median for the 2019 song data.

A screen shot of a computer

Description automatically generated

5. Music mode vs. mean and median for 2020.

A computer screen with text and numbers

Description automatically generated

## Measures of Variation

We noticed some differences in the distribution of these values across the set. We attribute this to how they vary across the sample. By getting some statistics to describe this variation.

1. Get range from grouped items.

A graph showing the number of years

Description automatically generated with medium confidence

1. Get standard deviation from grouped items.

A graph of a number of bars

Description automatically generated with medium confidence

1. Z Scores

For normally distributed data, 95% of the items should be within 2 standard deviations of the mean. This method of calculating the standard deviation of a measurement from the mean is called a zscore. this can be used in a fairly general way to identify outliers.

A graph of blue x marks

Description automatically generated with medium confidence

## Sample Error

There will always be an error between the statistics we report (e.g. the mean) and the true value. We can estimate how confident we can be that the sample is representative of the whole population by using the standard error (std / sqrt (sample size)).

1. tempo standard error of mean

A graph of a bar chart

Description automatically generated with medium confidence

1. Confidence intervals

By calculating confidence intervals, we can see how reliable our estimates of the mean musical tempo are for each year.

A black background with white numbers and symbols

Description automatically generatedA black background with white numbers

Description automatically generated

Num\_sample:100 Num\_sample:200

Adding num\_samples reduces the effect of outliers on the confidence interval and improves the accuracy of the estimate.

1. Making a Plot with confidence intervals on the mean tempo

A graph showing different colored bars

Description automatically generated with medium confidence

## Task

1. Define a function to calculate absolute difference

Absolute\_difference\_tempo=mean\_tempo-median\_tempo

A graph of a number of blue bars

Description automatically generated

The absolute difference between the mean and median rhythms is larger in 2019 and 2021, while the absolute difference is smaller in 2020 and 2022

1. Print the filenames of tracks with a z value greater than 3

A screenshot of a computer program

Description automatically generated

botton The Z score of this song is greater than 3 representing a larger difference in tempo compared to the entire dataset and may have a particular musical drum beat and rhythm.

A graph with blue and red dots

Description automatically generated

We can also find in the graph.

1. Task 3

#3.1

num\_samplesodd\_numbers = [i for i in np.arange(0,100) if i % 2 == 1 ]

num\_samplesodd\_numbers

#3.2

a = [i \* 2 + 1 for i in np.arange(0,50)]

print(a)

#3.3 Sum the list of numbers from 1-i each time

a = [np.sum(np.arange(i)) for i in np.arange(2,12)]

print(a)

#3.4

a = np.random.random(100)

#Use in place conditional

a = [1 if i > 0.5 else 0 for i in a]

print(a)

1. Task 4

Change number of samples in Confidence intervals

### Confidence intervals

confidence\_intervals = []

# num\_samples used to be 100

# out put

# 2019 122.2 +- 5.331

# 2020 119.8 +- 5.019

# 2021 125.5 +- 5.259

# 2022 122.9 +- 4.782

num\_samples = 200

z = 1.96

for label in Year:

#Get mean for genre

mean = mean\_tempos.loc[label]

#Get standard deviation for genre

std = std\_tempos.loc[label]["tempo"]

#Get confidence range

dist = z \* (std / np.sqrt(num\_samples))

print(label, np.round(mean, 1), "+-", np.round(dist,3))

confidence\_intervals.append([dist,dist])

output:

A black background with white numbers and symbols

Description automatically generated The width of the confidence interval decreases