Para aumentar o poder de análise de dados, você deve conhecer a distribuição de frequência

Encontre todos os fundamentos da distribuição de frequência em 7 minutos de leitura

Zubair Hossian 8 de agosto · 7 min de leitura ★

Foto de Yoann Boyer no Unsplash

Data plays a key role in every organization because it helps business leaders to make suitable decisions based on facts, statistical numbers, and trends. Due to this growing scope of data, data science came into the picture which is a multidisciplinary field. In data science, data analysis is the most vital part. To understand data clearly we have to know the knowledge of Frequency Distribution of statistics.

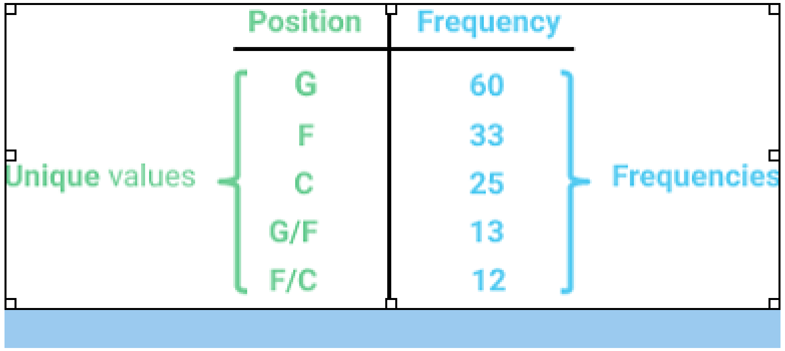
The main purpose of the data analysis is to gain information from data so that we can take better decisions for our system, organization, or any problem. What’s going on in your mind?. We can easily analyze data just by looking in a table format. Yeah! we can when the dataset is small. What if for a large dataset!!! Imagine you have a dataset of 1000 rows and 50 columns. Can you analyze this dataset just by looking? In order to analyze this type of large dataset, at first, we have to simplify it. Frequency distribution is one of the important techniques to analyze the data.

Table Of Contents:

2. Figure Out Frequency Table Using Real World Example.
3. Frequency Table For Ordinal, Interval, or Ratio Scales Variable.
4. What is Relative frequency and percentage frequency.
5. How to Make Grouped Frequency Distribution Table.
6. Frequency Distribution of a Continuous Variable.
7. Final Words.

What is Frequency Distribution ?

A frequency distribution is a representation, either in a graphical or tabular format, that displays the number of observations within a given interval or categories. Sometimes, it is also called the Frequency Distribution table.



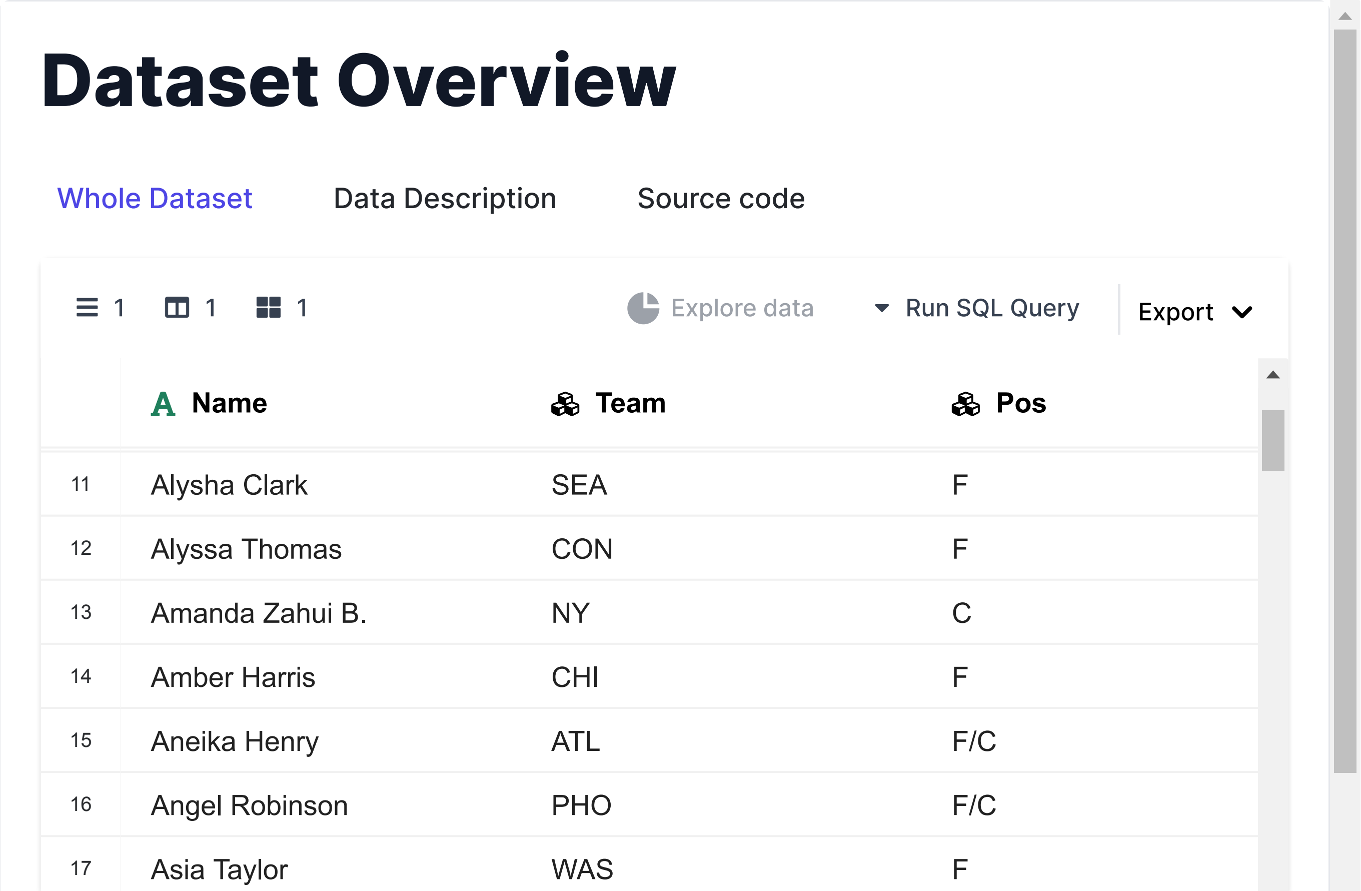
Position	Frequency
G	60
F	33
C	25
G/F	13
F/C	12

Unique values
Frequencies

Photo by author

Let’s have look at the table. It has two columns. One column records the unique variable’s name. Another column records the number of observations or occurrences of each unique value.

Through this article, we are using *wnba.csv* dataset. It has 143 rows and 32 columns. A complete overview of the dataset is given below. Feel free to observe the dataset.



Dataset Overview

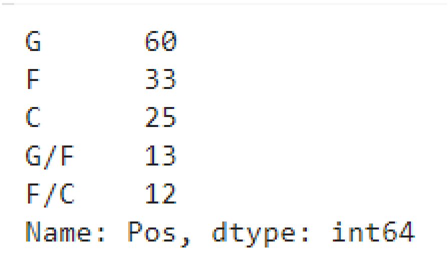
	Name	Team	Pos
11	Alysha Clark	SEA	F
12	Alyssa Thomas	CON	F
13	Amanda Zahui B.	NY	C
14	Amber Harris	CHI	F
15	Aneika Henry	ATL	F/C
16	Angel Robinson	PHO	F/C
17	Asia Taylor	WAS	F

Dataset for Showing Frequency Distribution created with datapane How was this built?

It’s time to try something new. We are going to create a frequency table using python. we can use `Series.value_counts()` method. And we will try to make an frequency table of `pos(position)` column in our dataset.

```
import pandas as pd
wnba = pd.read_csv(‘wnba.csv’)
freq_dis_pos = wnba[‘Pos’].value_counts()
freq_dis_pos
```

Output:



```
G      60
F      33
C      25
G/F    13
F/C    12
Name: Pos, dtype: int64
```

We can also get the frequency table of other columns using the same type of code. But keep in mind that it will only work fine for the categorical variables.

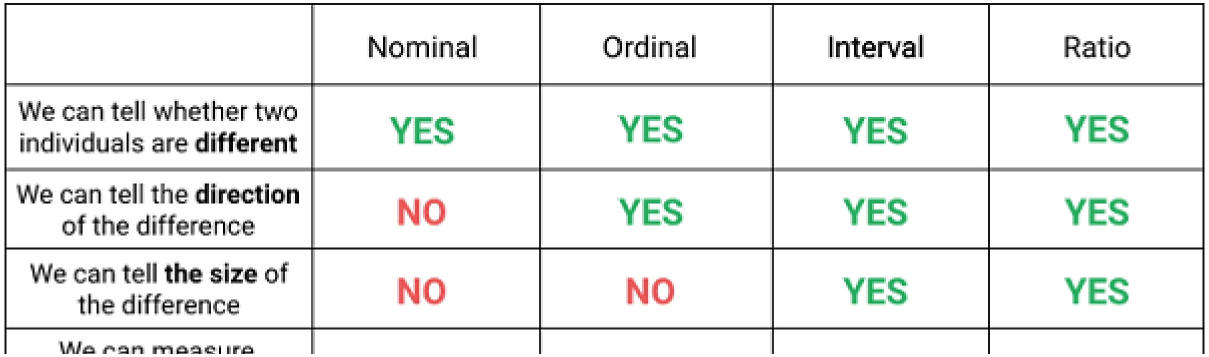
In our output, we see that values are in descending order. This order helps us to know that which has the maximum value of frequency. This order helps us if we have a nominal variable case. If your variable is measured in ordinal, interval, or ratio scales, it becomes more difficult to analyze. In order to understand the variable, you can see our previous post on variable.

Get Familiar with the Most Important Weapon of Data Science ~Variables

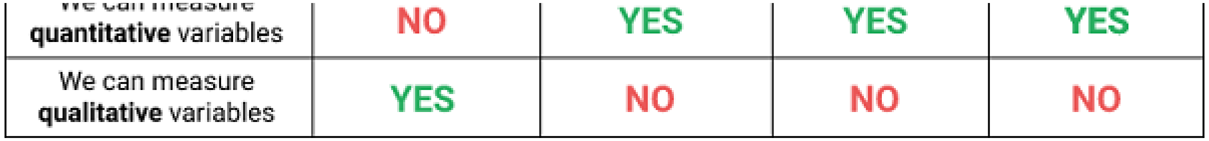
Basic concept of variable types, levels of measurement and different representation techniques with python

towardsdatascience.com

In summary, this table will help you to find the variable type.



	Nominal	Ordinal	Interval	Ratio
We can tell whether two individuals are **different**	YES	YES	YES	YES
We can tell the **direction** of the difference	NO	YES	YES	YES
We can tell **the size** of the difference	NO	NO	YES	YES
We can measure				



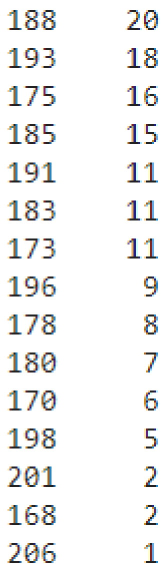
We can measure **quantitative** variables	NO	YES	YES	YES
We can measure **qualitative** variables	YES	NO	NO	NO

Photo by author (Frequency Table For Ordinal, Interval, or Ratio Scales Variable)

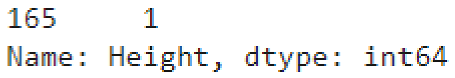
Now we try to find the frequency table of height column.

```
freq_dis_height = wnba['Height'].value_counts()
freq_dis_height
```

Output:



```
188    20
193    18
175    16
185    15
191    11
183    11
173    11
196     9
178     8
180     7
170     6
198     5
201     2
168     2
206     1
```

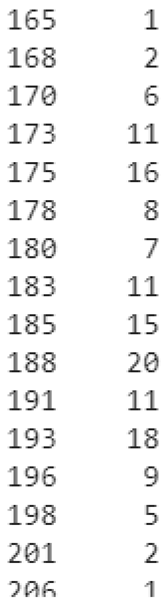


```
165     1
Name: Height, dtype: int64
```

Sometimes, you are in trouble. To solve this problem you might have to sort the table with respect to index value .Then you can use `Series.sort_index()` method.

```
freq_dis_height = wnba[‘Height’].value_counts().sort_index()
freq_dis_height
```

Output:



```
165     1
168     2
170     6
173    11
175    16
178     8
180     7
183    11
185    15
188    20
191    11
193    18
196     9
198     5
201     2
206     1
```

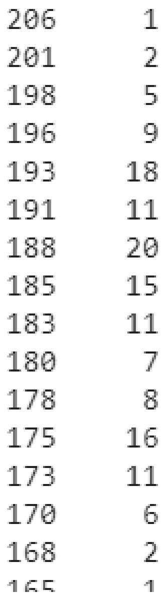
```
Name: Height, dtype: int64
```

Sometimes you need to transform data in ascending order. To represent data in ascending order, we can set the ascending parameter false.

```
freq_dis_height =wnba[‘Height’].value_counts().sort_index(ascending=
False)

freq_dis_height
```

Output:



```
206     1
201     2
198     5
196     9
193    18
191    11
188    20
185    15
183    11
180     7
178     8
175    16
173    11
170     6
168     2
165     1
```

165	1

Can you make the above table on an ordinal scale? Yeah, you can. At first, divide the whole range of height into some intervals. For example, in the given dataset the height ranges from 165 cm to 206 cm. Suppose, total range is 45 cm. Now, divide it into 5 categories each with a 9 cm interval. And give some categorial name to each interval like tall, short or 1st, 2nd, etc. Then calculate the frequency.

What is Relative frequency and percentage frequency?

The above discussed frequency is known as absolute frequencies of a certain variable.

(i)Relative Frequency:

The relative frequency of a particular observation or class interval is found by dividing the frequency(f) by the number of observation(n).

```
Relative frequency = frequency ÷ number of observations
```

(ii)Percentage Frequency:

The percentage frequency is found by multiplying 100 to the relative frequency.

Percentage frequency = relative frequency X 100

In pandas library, we can compute all the proportions at once by dividing each frequency with the total number of instances. An example is shown below with the `Women's National Basketball Association (wnba.csv)` dataset.

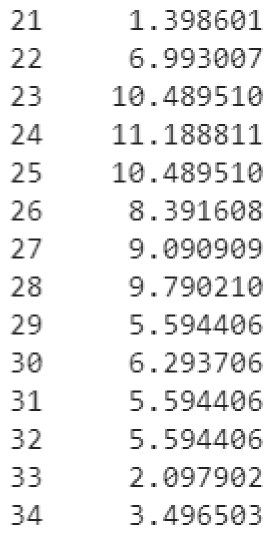
```
wnba['Age'].value_counts() / len(wnba))
```

But it's slightly been faster by setting the `Series.value_counts()` normalize value became True. Then simply the output multiply with 100.

```
percentages_pos =
wnba['Age'].value_counts(normalize=True).sort_index() * 100

percentages_pos
```

Output:



```
21     1.398601
22     6.993007
23    10.489510
24    11.188811
25    10.489510
26     8.391608
27     9.090909
28     9.790210
29     5.594406
30     6.293706
31     5.594406
32     5.594406
33     2.097902
34     3.496503
```

This percentage will help us to find the important information as per our need.

Percentiles rank

The percentile rank of a score is the percentage of score in its distribution and lower than it. To find percentiles rank, We can use a library called `scipy.stats.percentileofscore` in python.

If we want to find the percentiles rank of index 25. We just write the code as below.

```
from scipy.stats import percentileofscore

percentile_of_25 = percentileofscore(wnba['Age'], 25, kind = 'weak')

percentile_of_25
```

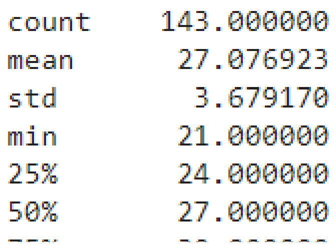
Output:

40.55944055944056

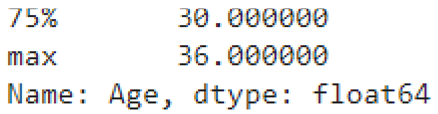
You are very much surprised to know that we can easily find percentiles just write one line code. Pandas `Series.describe()` method help us to find percentiles.

```
persecntiles = wnba['Age'].describe()
```

Output:



```
count    143.000000
mean      27.076923
std        3.679170
min       21.000000
25%       24.000000
50%       27.000000
```



```
75%         30.000000
max         36.000000
Name: Age, dtype: float64
```

We are not interested in the value of the first three rows. The 25th,50th, and 75th are returned by default, the scores have divided the distribution into four equal parts. Also known as quartiles. The first quartile (also called the lower quartile) is 24 (note that 24 is also the 25th percentile). That means 25% of the total data are within 0 to 24 years. The second quartile (also called the middle quartile) is 27 (note that 27 is also the 50th percentile). And the third quartile (also called the upper quartile) is 30 (note that 30 is also the 75th percentile).

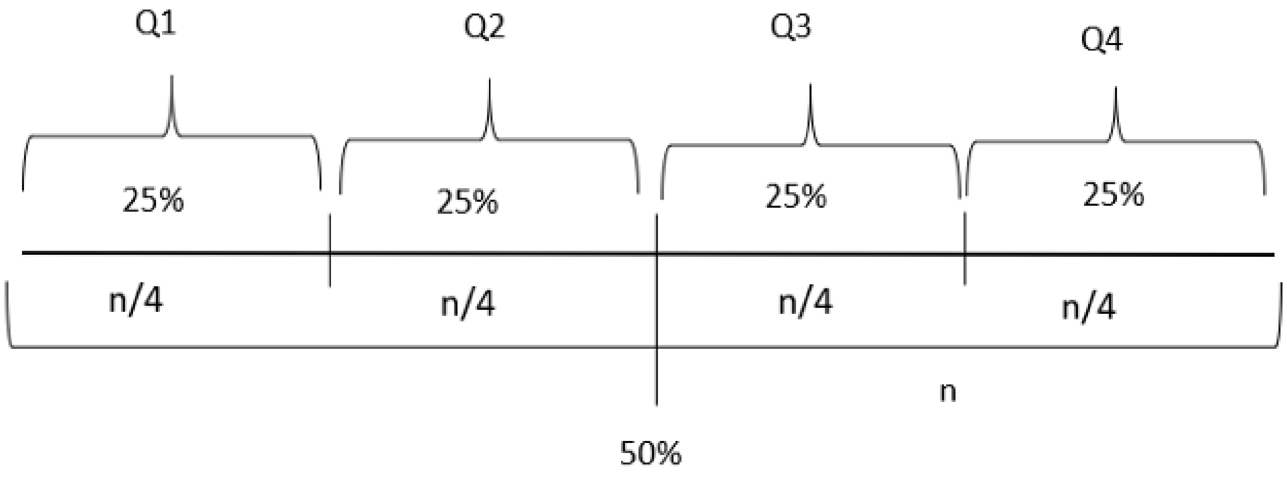


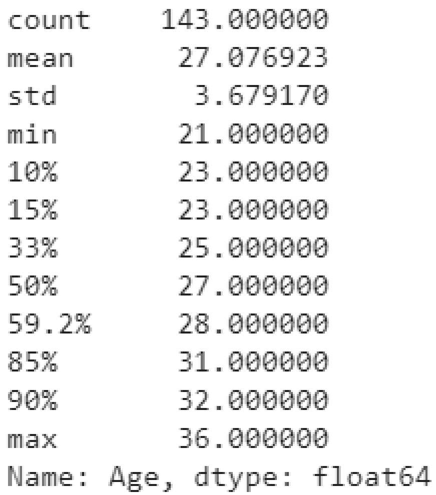
Foto do autor (visualização gráfica do conceito de quartis)

Podemos estar interessados em encontrar os percentis para porcentagens diferentes de 25%, 50% ou 75%. Para isso, podemos usar o parâmetro percentis dos pandas `Series.describe()` . Este método requer que as percentagens sejam passadas como desejamos entre 0 e 1.

```
persecntiles = wnba ['Age']. describe (percentiles = [.1, .15, .33,
.5, .592, .85, .9])

persecntiles
```

Saída:



```
count    143.000000
mean      27.076923
std        3.679170
min       21.000000
10%       23.000000
15%       23.000000
33%       25.000000
50%       27.000000
59.2%     28.000000
85%       31.000000
90%       32.000000
max       36.000000
Name: Age, dtype: float64
```

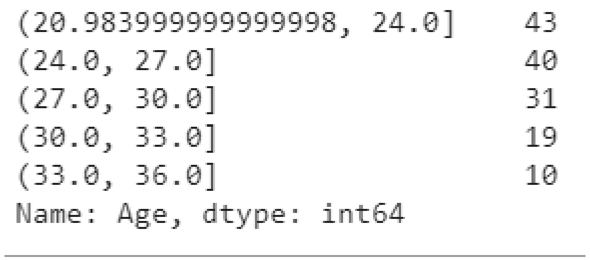
Como fazer uma tabela de distribuição de frequência agrupada?

Às vezes, as tabelas de distribuição de frequência não são bem organizadas. Então, temos que encontrar a tabela de distribuição de frequência agrupada. Definimos o limite do valor agrupado apenas alterando o parâmetro bins no `series.value_counts()` método pandas .

```
grouped_freq = wnba ['Idade']. value_counts (bins = 5) .sort_index ()

grouped_freq
```

Saída:



```
(20.983999999999998, 24.0]    43
(24.0, 27.0]                  40
(27.0, 30.0]                  31
(30.0, 33.0]                  19
(33.0, 36.0]                  10
Name: Age, dtype: int64
```

Às vezes, esse intervalo não dá uma saída melhor. Para obter uma melhor saída, temos que criar um intervalo personalizado. **`Pandas`** dê-nos a oportunidade de criar uma gama personalizada.

```
 ntervals = pd.interval_range (start = 0, end = 600, freq = 100)

 intervalos
```

Saída:

```
 IntervalIndex([(0, 100], (100, 200], (200, 300], (300, 400], (400,
500], (500, 600]], closed='right', dtype='interval[int64]')
```

Aqui, temos que fornecer três parâmetros. O parâmetro inicial fornece o ponto de partida do nosso intervalo. O parâmetro final fornece o ponto final do intervalo personalizado e o valor Freq fornece o número do valor em cada frequência.

Distribuição de frequência de uma variável contínua

Para uma variável contínua, se tomarmos uma classe para cada valor distinto da variável, o número de classes se tornará indevidamente grande, anulando assim o propósito da tabulação.

Quando variáveis contínuas são usadas em tabelas, **seus valores costumam ser agrupados em categorias.** Aqui, podemos usar o conceito de intervalo que aprendemos anteriormente.

Por último,

Ao longo do artigo, temos que aprender como organizar os dados usando a Tabela de distribuição de frequência. Precisamos saber o quão poderosa é a tabela de distribuição de frequência! A tabela de distribuição de frequência nos ajuda a entender os dados profundamente. Porém, é hora de saber como visualizar esses dados organizados. Para saber sobre isso, por favor, fique comigo. Voltarei em breve com as técnicas de visualização necessárias.

Série anterior de artigos sobre os fundamentos da ciência de dados

Menos é mais; a 'Arte' da Amostragem

Aumente seu poder de análise de dados de um vasto conjunto de dados com amostra

paradatascience.com

Familiarize-se com a arma mais importante da ciência de dados ~ Variáveis

Conceito básico de tipos de variáveis, níveis de medição e diferentes técnicas de representação com python

paradatascience.com

Artigo interessante que o ajudará a saber como incorporar o conjunto de dados interativo com artigos

Diga adeus à captura de tela e use o Datapane para relatório de ciência de dados

Um tutorial completo do Datapane em 7 minutos de leitura

paradatascience.com

Zubair Hossain

- *Se você gostou do artigo, siga-me no **meio** para saber mais.*
- *Conecte-me no **LinkedIn** para colaboração.*

Ciência de Dados Aprendizado de Máquina Inteligência artificial Estatisticas Análise de dados