Numerical Transformation Introduction

8 min

We've all heard it; a lot of work goes into getting your data just right for the task you have at hand. Perhaps you are trying to find insight into customer behavior or predicting the best time to send a marketing email. Whatever problem you may be trying to solve - you have probably already spent some time wrangling your data and getting it just right, and now you are at the stage where you need to prepare your data for machine learning.

For example, some machine learning models, like logistic regression and neural networks, can only handle numbers. Then random forest and decision tree models can take both numbers and text. We will call these text features categorical data, but more on that later. We also need to understand the spread of our data, as some models have a tough time handling data that has extreme outliers.

This process is called *numerical transformation*, when we take our numerical data and change it into another numerical value. This is meant to change the scale of our values or even adjust the skewness of our data. You may be thinking, "we already have our data in numbers. Why would we want to change those?" Well, first of all, that is a great question. We'll dive deep into the "why we do this" throughout this lesson. To put it simply, we do this to help our model better compare features and, most importantly, improve our model's accuracy and interpretability. That sounds like some good reasons to put the time and effort into numerical transformations if I do say so myself.

We'll focus on the following numerical transformations:

- Centering
- Standard Scaler
- Min and Max Scaler
- Binning
- Log transformations

Let's get to know the data frame we will be using. This dataset has just over 100 responses from customers where they were asked about a recent Starbucks experience. You will soon notice that we have a mix of numerical and categorical data, but we'll focus only on the numerical features for this lesson.

Instructions

1. Checkpoint 1 Passed

1.

We have provided you with a csv file under the name starbucks_customers.csv and have already imported pandas for you. Import the file and set it to a variable called coffee.

Stuck? Get extra guidance

2. Checkpoint 2 Passed

2.

Examine the features in your new data frame by printing the columns.

Stuck? Get extra guidance

3. Checkpoint 3 Passed

3.

Look at each feature within your data frame by printing .info()

script.py

```
import pandas as pd

## add code below
## set up dataframe
coffee = pd.read_csv('starbucks_customers.csv')

## print the column names
print(coffee.columns)

## get information on your data
print(coffee.info())
```

```
Index(['spent', 'nearest_starbucks', 'age',
'rate_quality', 'rate_price',
       'rate_promo', 'ambiance', 'wifi', 'service',
'meetings_hangout'],
      dtype='object')
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 122 entries, 0 to 121
Data columns (total 10 columns):
spent                122 non-null int64
nearest_starbucks    122 non-null int64
age                  122 non-null int64
rate_quality         122 non-null int64
rate_price           122 non-null int64
rate_promo           122 non-null int64
ambiance             122 non-null int64
wifi                 122 non-null int64
service              122 non-null int64
meetings_hangout     122 non-null int64
dtypes: int64(10)
memory usage: 9.7 KB
None
```