# Welcome to Data Science Online Bootcamp

## Week#2 - Day 1

dφ

Democratizing Data Science Learning

# Learning Objectives

Measures of Dispersion

Transformations

EDA with Seaborn

Additional Python Topics

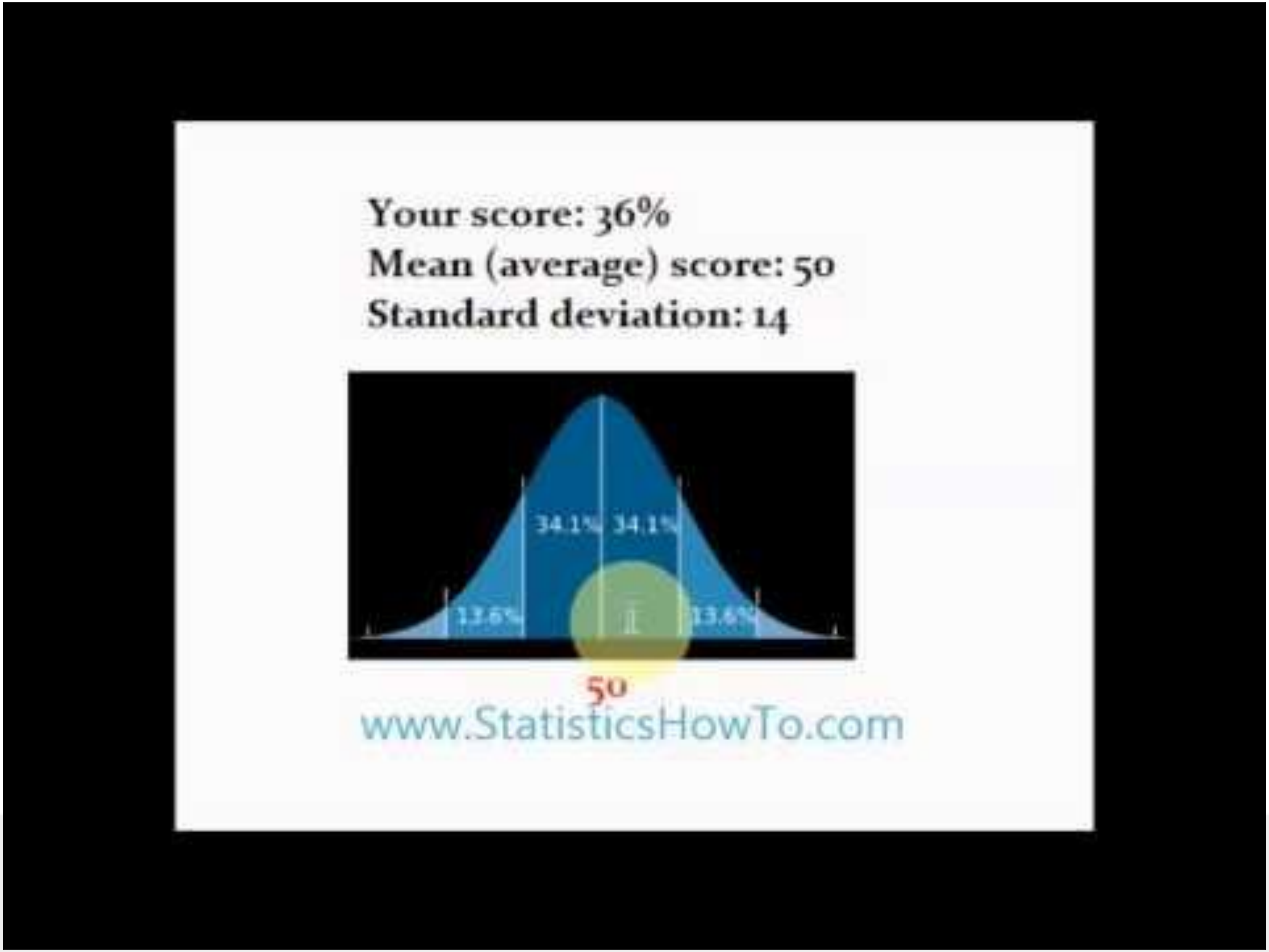
# Measures of Dispersion

- **Range:** It is the difference between highest value and the lowest value in the data set.
  For a given list of numbers: 10, 20, 40, 10, 70 the range is 70 - 10 = 60.

- **Variance:** The average of the squared differences from the mean.
  Steps to calculate variance:
  - Calculate mean (mean is nothing but average)
  - Find difference of each data from mean
  - Square all the differences
  - Take the average of the squares.

- **Standard Deviation:** It shows you how much your data is spread out around the mean. Its symbol is **σ** (the greek letter sigma). It is the square root of the **variance.**

# Standard Deviation



**Your score: 36%**
**Mean (average) score: 50**
**Standard deviation: 14**

www.StatisticsHowTo.com

# Calculating Variance

**Steps to calculate variance:**

- Calculate mean
- Find difference of each data from mean
- Square all the differences
- Take the average of the squares.

Consider the list of numbers: 10, 20, 40, 10, 70.

- Mean of the number is 30.
- Difference of each data from the mean: -20, -10, 10, -20, 40.
- Square of all the differences: 400, 100, 100, 400, 1600
- Take the average of the squares:
  (400 + 100 + 100 + 400 + 1600) / 5 = 2600 / 5 = 520

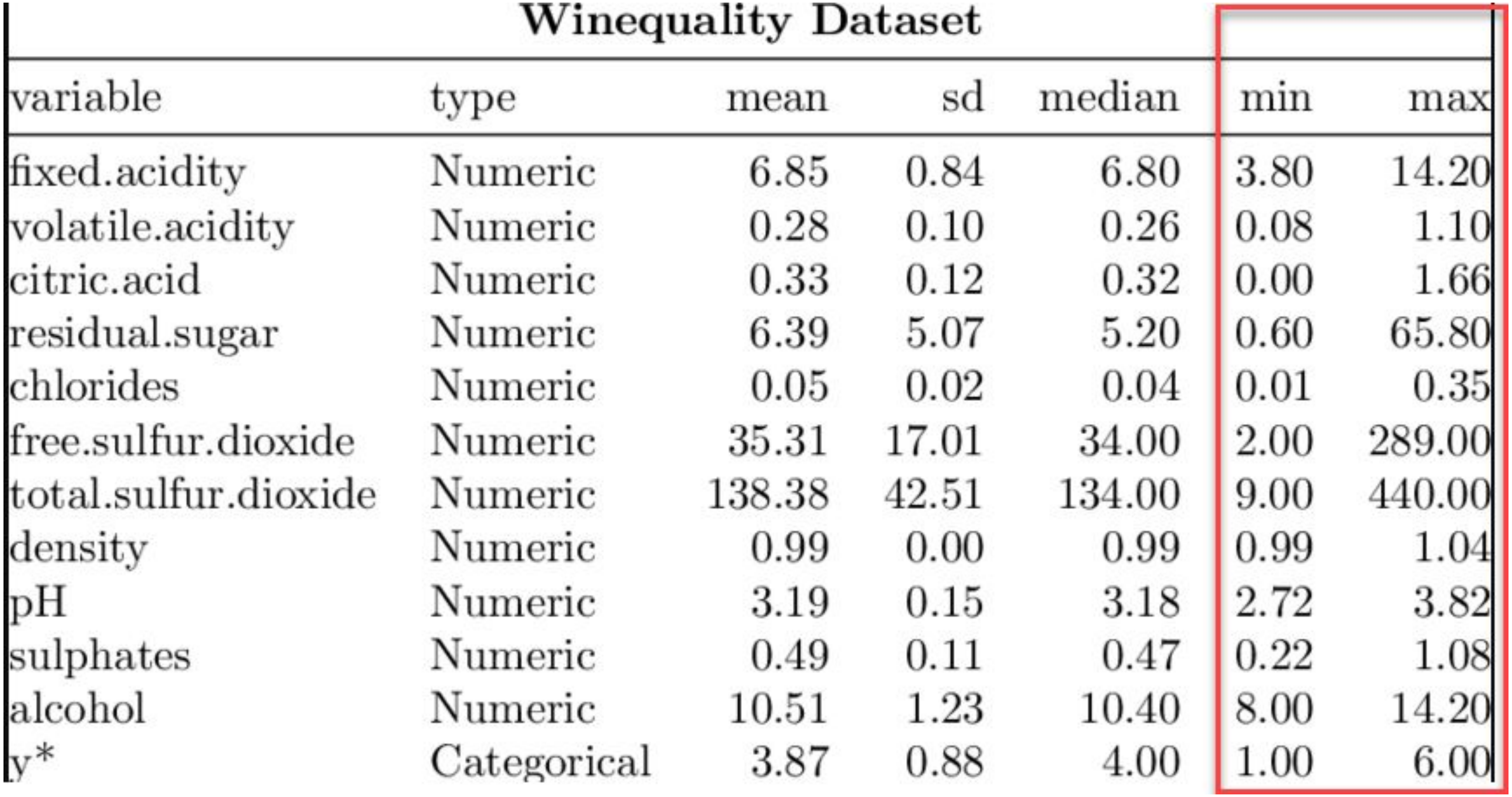
# Standardization/normalization:

Often variables in a real dataset come with a wide range of data values.

**For example** let’s look at the wine dataset given in next slide, the **fixed.acidity variable has values ranging from 3.8 to 14.2**. Similarly, if we look at **volatile.acidity, it has values ranging from 0.08 to 1.10**. Basically, they are not on a common scale.

**Now how does standardization/normalization help?**
Performing standardization/normalization would bring all the variables in a dataset to a common scale so that it could further help in implementing various machine learning models (where standardization/normalization is a pre-requisite to apply such models). Again, don’t take this for granted, there are some smart algorithms which doesn’t need this and we will explore them soon!

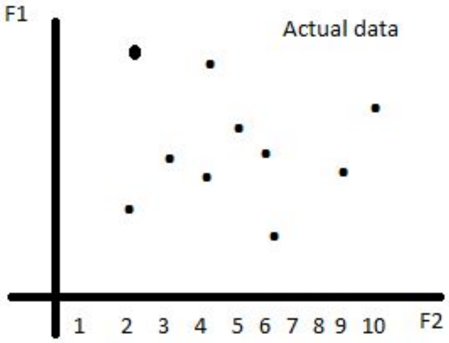
# Standardization/normalization:

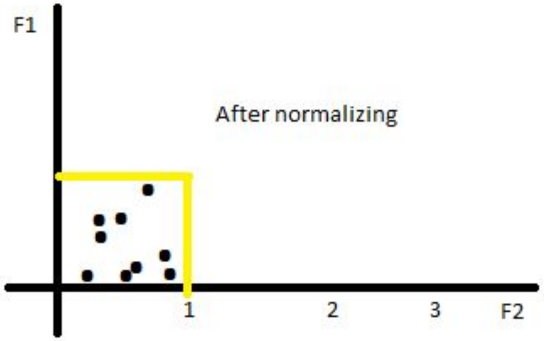


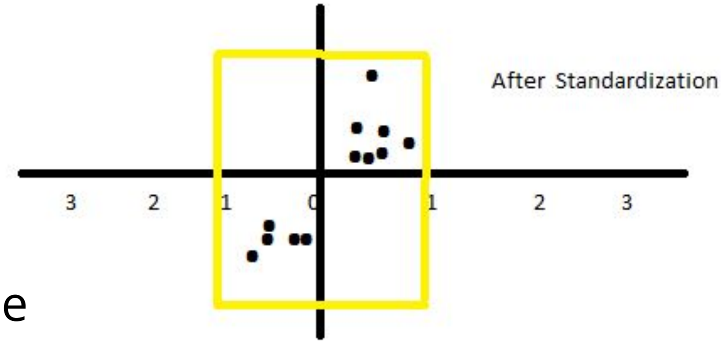
**Winequality Dataset**

| variable | type | mean | sd | median | min | max |
|---|---|---|---|---|---|---|
| fixed.acidity | Numeric | 6.85 | 0.84 | 6.80 | 3.80 | 14.20 |
| volatile.acidity | Numeric | 0.28 | 0.10 | 0.26 | 0.08 | 1.10 |
| citric.acid | Numeric | 0.33 | 0.12 | 0.32 | 0.00 | 1.66 |
| residual.sugar | Numeric | 6.39 | 5.07 | 5.20 | 0.60 | 65.80 |
| chlorides | Numeric | 0.05 | 0.02 | 0.04 | 0.01 | 0.35 |
| free.sulfur.dioxide | Numeric | 35.31 | 17.01 | 34.00 | 2.00 | 289.00 |
| total.sulfur.dioxide | Numeric | 138.38 | 42.51 | 134.00 | 9.00 | 440.00 |
| density | Numeric | 0.99 | 0.00 | 0.99 | 0.99 | 1.04 |
| pH | Numeric | 3.19 | 0.15 | 3.18 | 2.72 | 3.82 |
| sulphates | Numeric | 0.49 | 0.11 | 0.47 | 0.22 | 1.08 |
| alcohol | Numeric | 10.51 | 1.23 | 10.40 | 8.00 | 14.20 |
| y* | Categorical | 3.87 | 0.88 | 4.00 | 1.00 | 6.00 |

# Standardization vs Normalization







Look at the above graphs.
The actual data is scattered across a wide range.

Normalisation helps in bringing the whole data within a particular range.

Standardisation distributes the data in a manner such that it now has a mean of 0 and standard deviation of 1.

# Transformation

Some machine learning algorithms are quite sensitive to the scale of the numeric values provided.

Consequently, in order for the algorithm to converge faster or to provide a more exact solution, rescaling the distribution is necessary. Rescaling mutates the range of the values of the features and can affect variance, too. You can perform features rescaling in two ways:

1. Using statistical **standardization** (z-score normalization)
   Standardization typically means rescaling data to have a mean of 0 and a standard deviation of 1 (unit variance).

2. Using the min-max transformation (or **normalization**)
   Normalization typically means rescaling the values into a range of [0,1].

# EDA with Seaborn

In today’s session, we’ll be performing EDA by visualising data with Seaborn (specifically with scatterplot, countplot, distplot, boxplot and heatmap).

Please go through the following material to understand these different plots:

https://towardsdatascience.com/how-to-perform-exploratory-data-analysis-with-seaborn-97e3413e841d

https://towardsdatascience.com/analyze-the-data-through-data-visualization-using-seaborn-255e1cd3948e

# Additional Python Topics

**(This is for self-learning for today’s session and may not be required in near time  - you consider this as optional)**

# Python Lambda Function

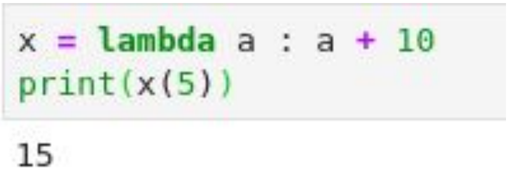
- A lambda function is a small anonymous function.

- A lambda function can take any number of arguments, but can only have one expression.

- Syntax:

  lambda arguments : expression

- Example:
  A lambda function that adds 10 to the number passed in as an argument, and prints the result:



```
x = lambda a : a + 10
print(x(5))
```

```
15
```

Here’s a short article about lambda function and its applications:
https://www.programiz.com/python-programming/anonymous-function

# Python List Comprehension

- One of the Python’s most distinctive features is the list comprehension, which you can use to create powerful functionality within a single line of code.

- List comprehension is generally more compact and faster than normal functions and loops for creating list.

- MUST READ: Learn about list comprehension from here: https://www.programiz.com/python-programming/list-comprehension

# Python Regular Expression

- A regular expression is a special sequence of characters that helps you match or find other strings or sets of strings, using a specialized syntax held in a pattern.

- Must Read: Go through the following link to get an overview of how regular expression are used in Python: https://www.tutorialspoint.com/python/python_reg_expressions.htm

- It is usually a confusing topic for most programmers. So don’t worry if you’re not able to fully understand it.

# Session Details

- **Tutor:** Joinal Ahmed
- **Topic:** Introduction to descriptive statistics and exploratory data analysis
- **Notebooks link:** https://github.com/dphi-official/Exploratory-Data-Analysis

- **Date & time:** 3rd June, at 8:30 pm IST (please locate your time in your timezone here).

- **Youtube live link:** https://youtu.be/5CoETeAdi9A

# That’s it for the day. Thank you!

Feel free to post any queries in the #help channel on Slack