Big Data Analytics
Outliers

Muhammad Affan Alim

Outliers

- An outlier is a data point that is significantly different from the remaining data
- Outliers may also affect the performance of some machine learning models, such as linear regression or AdaBoost

Outliers- cont…

How can we engineer outliers?

- One way to handle outliers is to perform variable discretization

- An alternative way to handle outliers is to assume that the information is missing, treat the outliers together with the remaining missing data, and carry out any of the missing imputation techniques

Outliers –cont…

- We will also discuss how to use the mean and standard deviation for normally distributed variables or the inter quartile range for skewed features or using percentiles, in a process commonly known as **winsorization**

Trimming outliers from the dataset

1. Import the required Python libraries:

```
>> import pandas as pd
>> import numpy as np
>> import matplotlib.pyplot as plt
>> import seaborn as sns
>> from sklearn.datasets import load_boston
```

Trimming outliers from the dataset-cont…

2. Let's load the Boston House Prices dataset from scikit-learn:

```
>> boston_dataset = load_boston()
```

3. Let's capture three of the variables, RM, LSTAT, and CRIM, in a pandas dataframe:

```
>> boston = pd.DataFrame(boston_dataset.data,
columns=boston_dataset.feature_names)[['RM', 'LSTAT',
'CRIM']]
```

Trimming outliers from the dataset-cont…

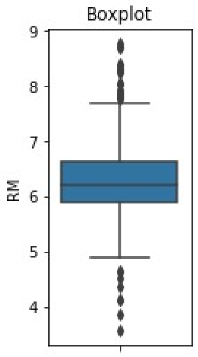
- Let's make a boxplot of the RM variable to visualize outliers:

```
>> sns.distplot(boston['RM'], bins=30)
>> sns.boxplot(boston[‘Rm’])
```

- The outliers are the asterisks sitting outside the whiskers, which delimit the interquartile range proximity rule boundaries:

Trimming outliers from the dataset-cont…

- The outliers are the asterisks sitting outside the whiskers, which delimit the interquartile range proximity rule boundaries:



Trimming outliers from the dataset-cont…

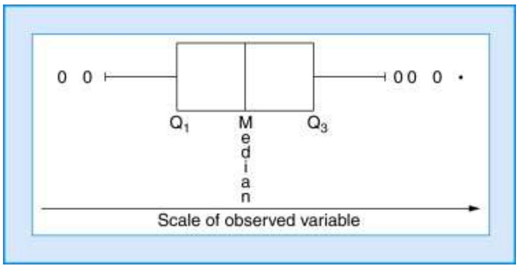
5. Let's create a function to find the boundaries of a variable distribution, using the inter-quartile range proximity rule:

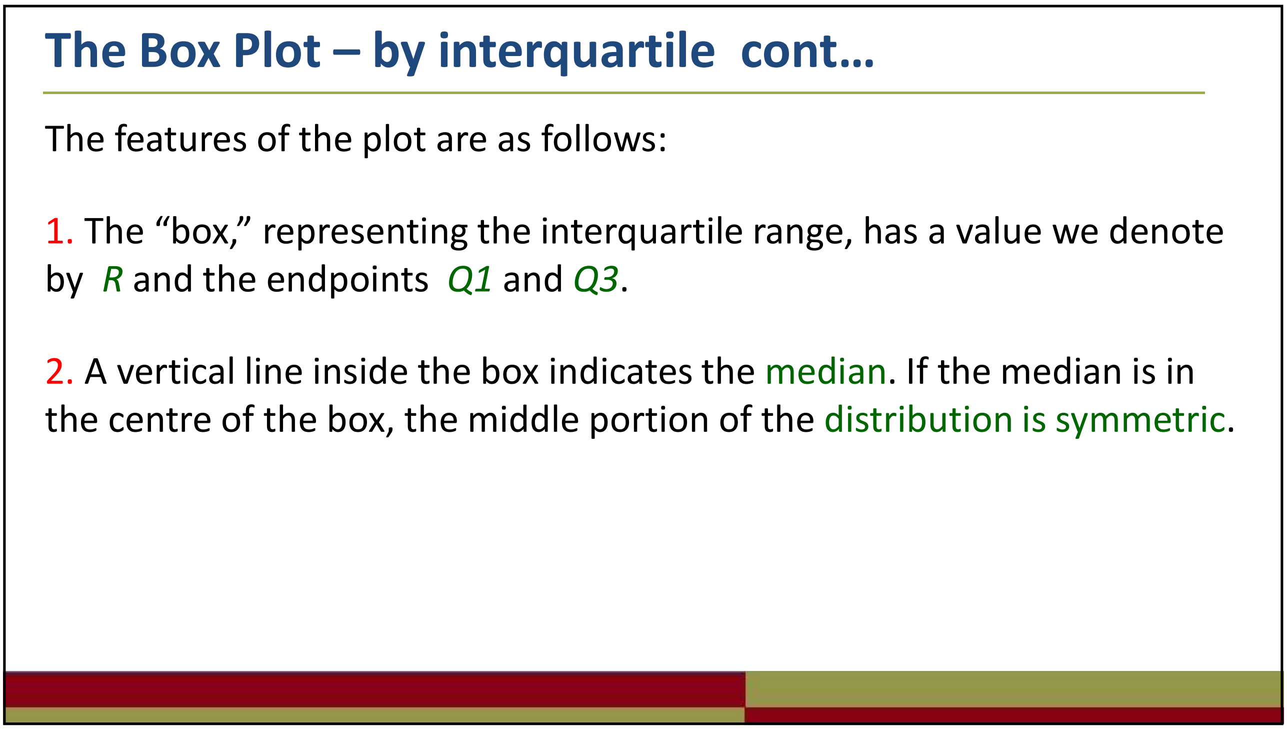
- The IQR describes the middle 50% of values when ordered from lowest to highest. To find the interquartile range (IQR), first find the median (middle value) of the lower and upper half of the data. These values are quartile 1 (Q1) and quartile 3 (Q3). The IQR is the difference between Q3 and Q1.

IQR = Q3 – Q1

The Box Plot – by interquartile

- The box plot is used to show distributional shapes and to detect unusual observations.



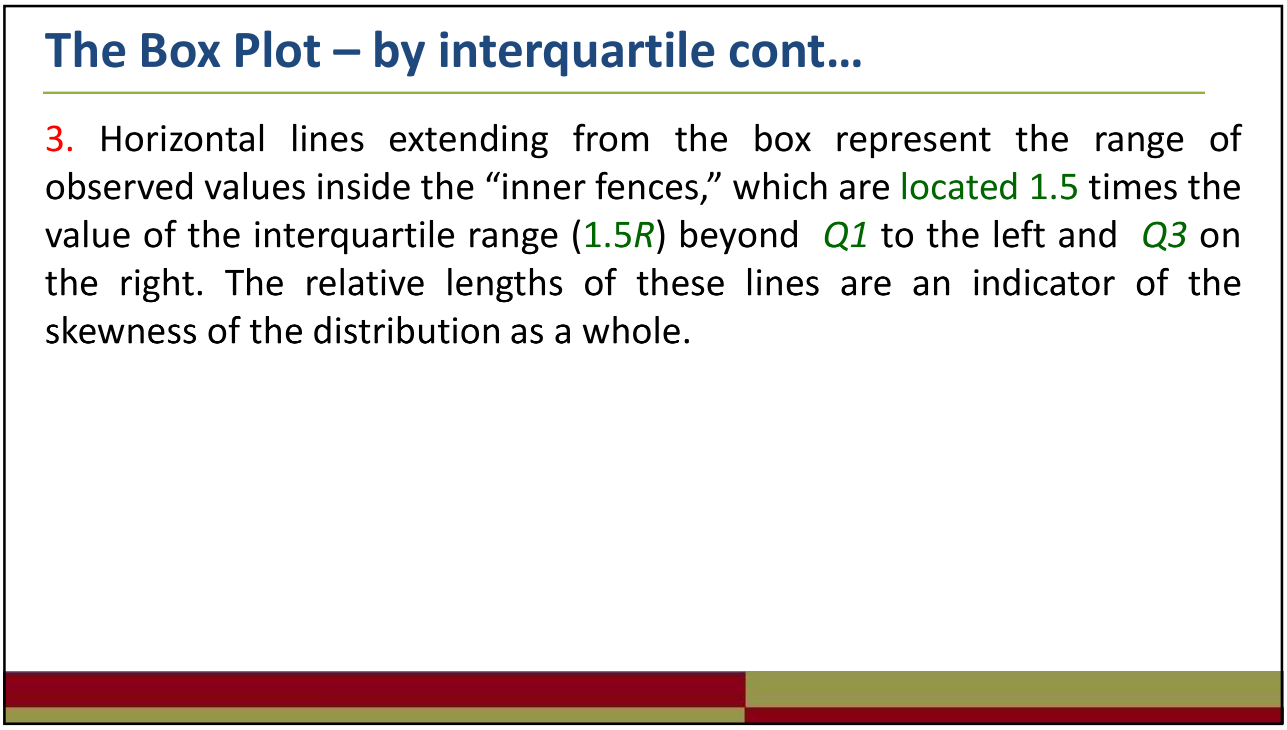


The Box Plot – by interquartile cont…

The features of the plot are as follows:

1. The “box,” representing the interquartile range, has a value we denote by *R* and the endpoints *Q1* and *Q3*.

2. A vertical line inside the box indicates the median. If the median is in the centre of the box, the middle portion of the distribution is symmetric.



The Box Plot – by interquartile cont…

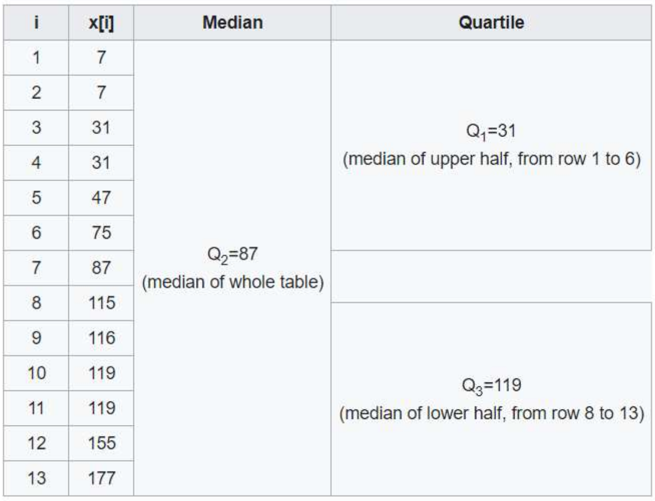
3. Horizontal lines extending from the box represent the range of observed values inside the “inner fences,” which are located 1.5 times the value of the interquartile range (1.5*R*) beyond *Q1* to the left and *Q3* on the right. The relative lengths of these lines are an indicator of the skewness of the distribution as a whole.

The Box Plot – by interquartile cont…

4. Individual symbols O represent “mild” outliers, which are defined as values between the inner and outer fences that are located *3R* units beyond *Q1* and *Q3*.

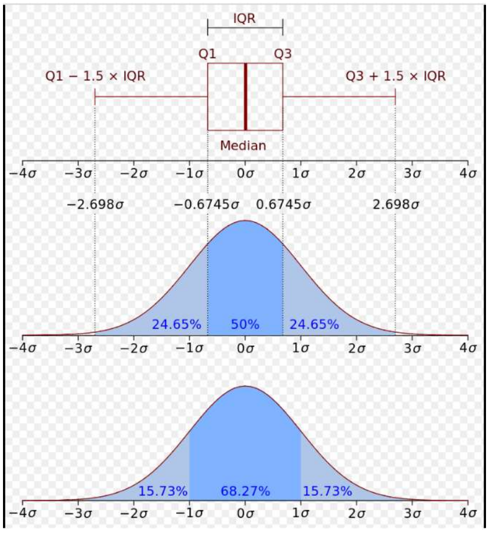
5. Individual symbols represent the location of extreme outliers, which are defined as being beyond the outer fences. Different computer programs may use different symbols for outliers and may provide options for different formats.

The Box Plot – by interquartile cont…

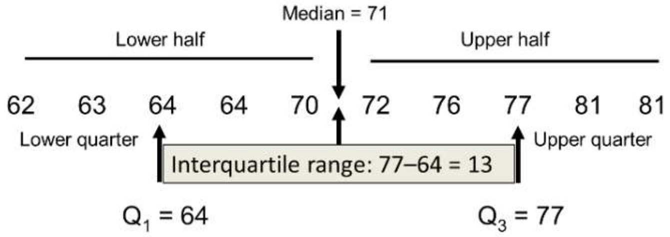


i	x[i]	Median	Quartile
1	7	Q_2=87 (median of whole table)	Q_1=31 (median of upper half, from row 1 to 6)
2	7		
3	31		
4	31		
5	47		
6	75		
7	87		
8	115		Q_3=119 (median of lower half, from row 8 to 13)
9	116		
10	119		
11	119		
12	155		
13	177		

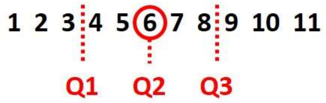
For the data in this table the interquartile range is IQR = $Q_3 - Q_1$ = 119 - 31 = 88.



The Box Plot – by interquartile cont…

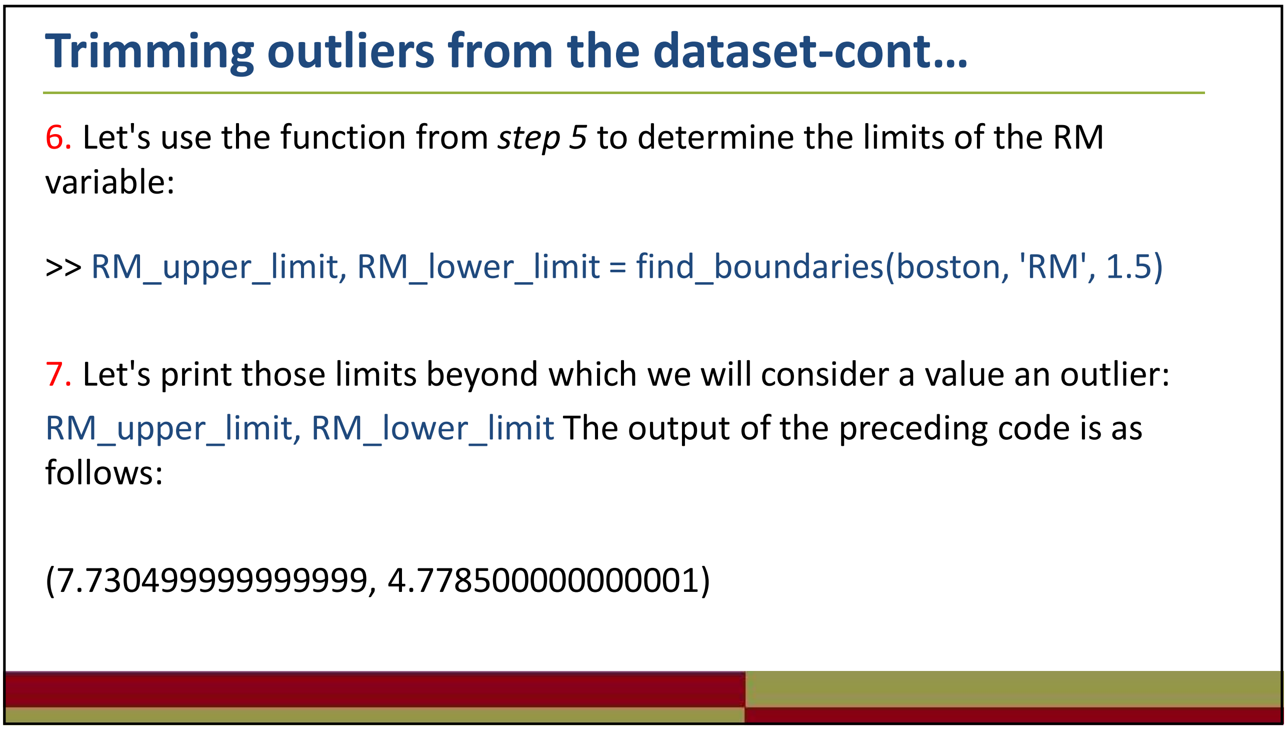


$$
\begin{aligned}
IQR &= Q3 - Q1 \\
&= 8.5 - 3.5 \\
&= 5
\end{aligned}
$$



Trimming outliers from the dataset-cont…

```
>> def find_boundaries(df, variable, distance):
        IQR = df[variable].quantile(0.75) - df[variable].quantile(0.25)
        lower_boundary = df[variable].quantile(0.25) - (IQR * distance)
        upper_boundary = df[variable].quantile(0.75) + (IQR * distance)
        return upper_boundary, lower_boundary
```



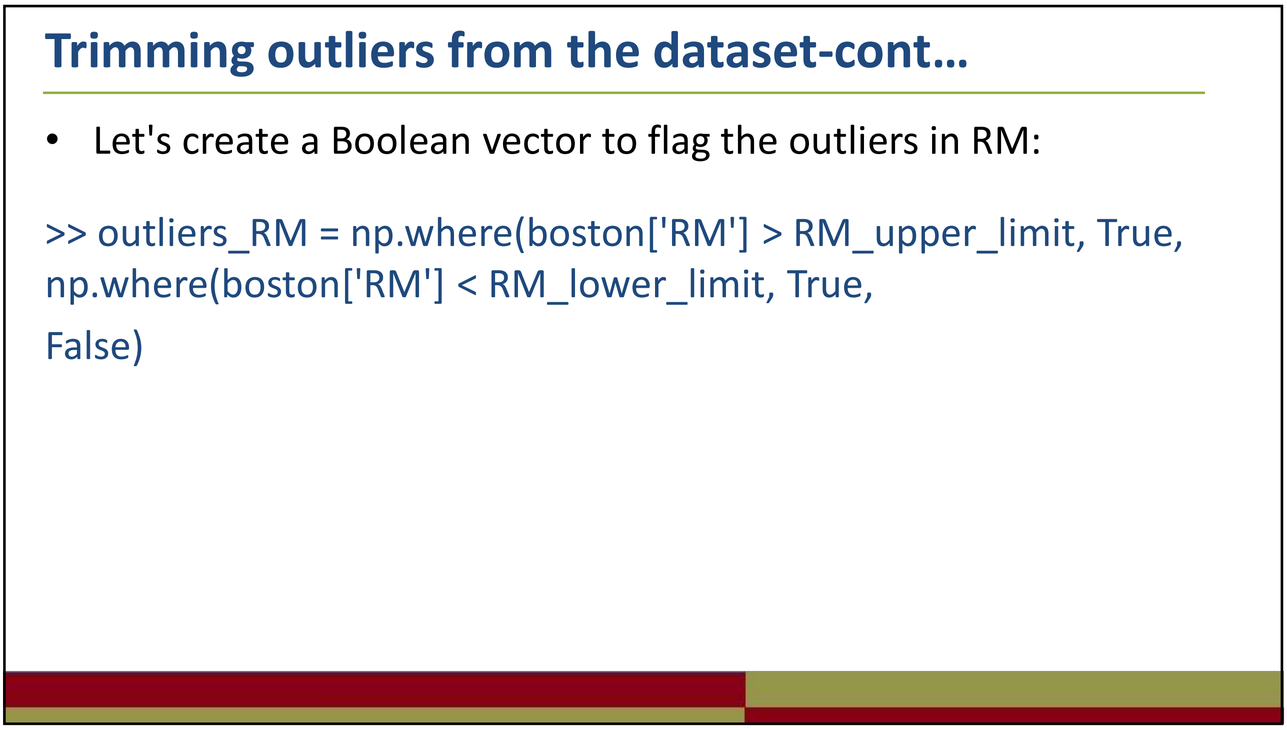
Trimming outliers from the dataset-cont…

6. Let's use the function from *step 5* to determine the limits of the RM variable:

```
>> RM_upper_limit, RM_lower_limit = find_boundaries(boston, 'RM', 1.5)
```

7. Let's print those limits beyond which we will consider a value an outlier: RM_upper_limit, RM_lower_limit The output of the preceding code is as follows:

```
(7.730499999999999, 4.778500000000001)
```



Trimming outliers from the dataset-cont…

- Let's create a Boolean vector to flag the outliers in RM:

```
>> outliers_RM = np.where(boston['RM'] > RM_upper_limit, True,
np.where(boston['RM'] < RM_lower_limit, True,
False)
```

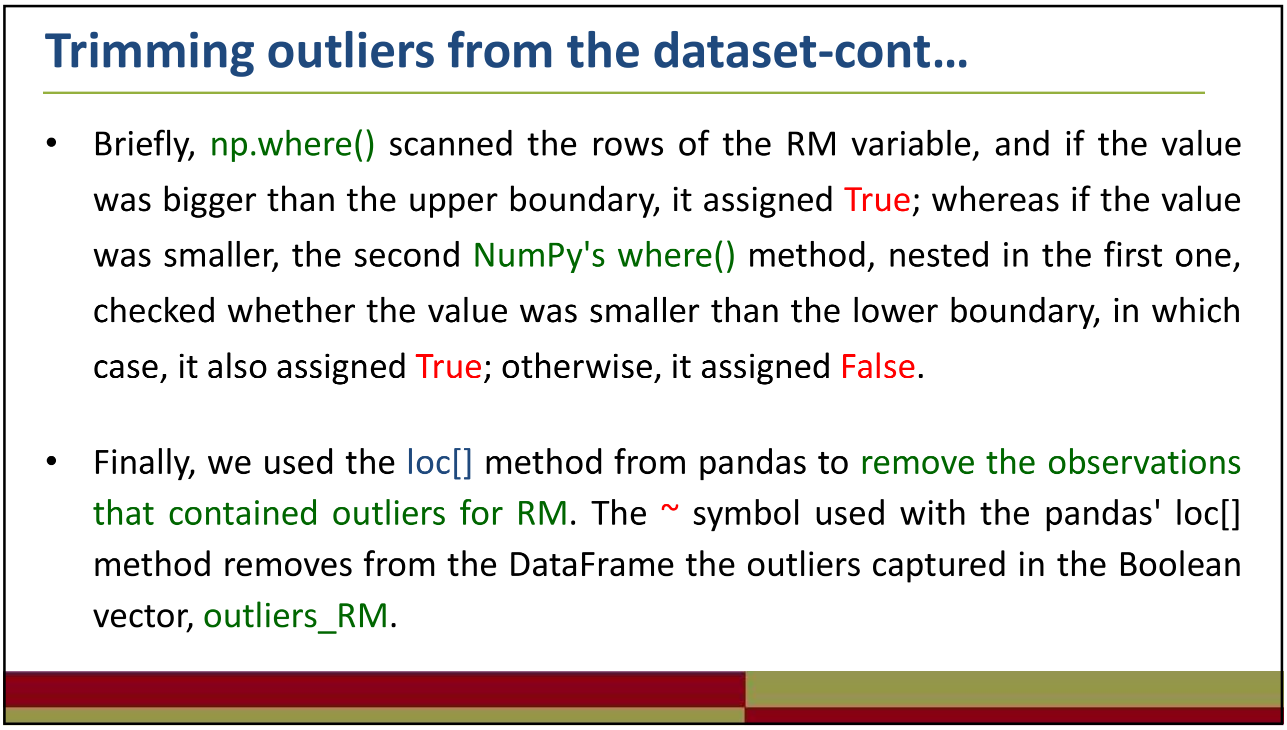
Trimming outliers from the dataset-cont…

- Finally, let's remove the outliers from the dataset:

```
>> boston_trimmed = boston.loc[~(outliers_RM)]
```

Trimming outliers from the dataset-cont…

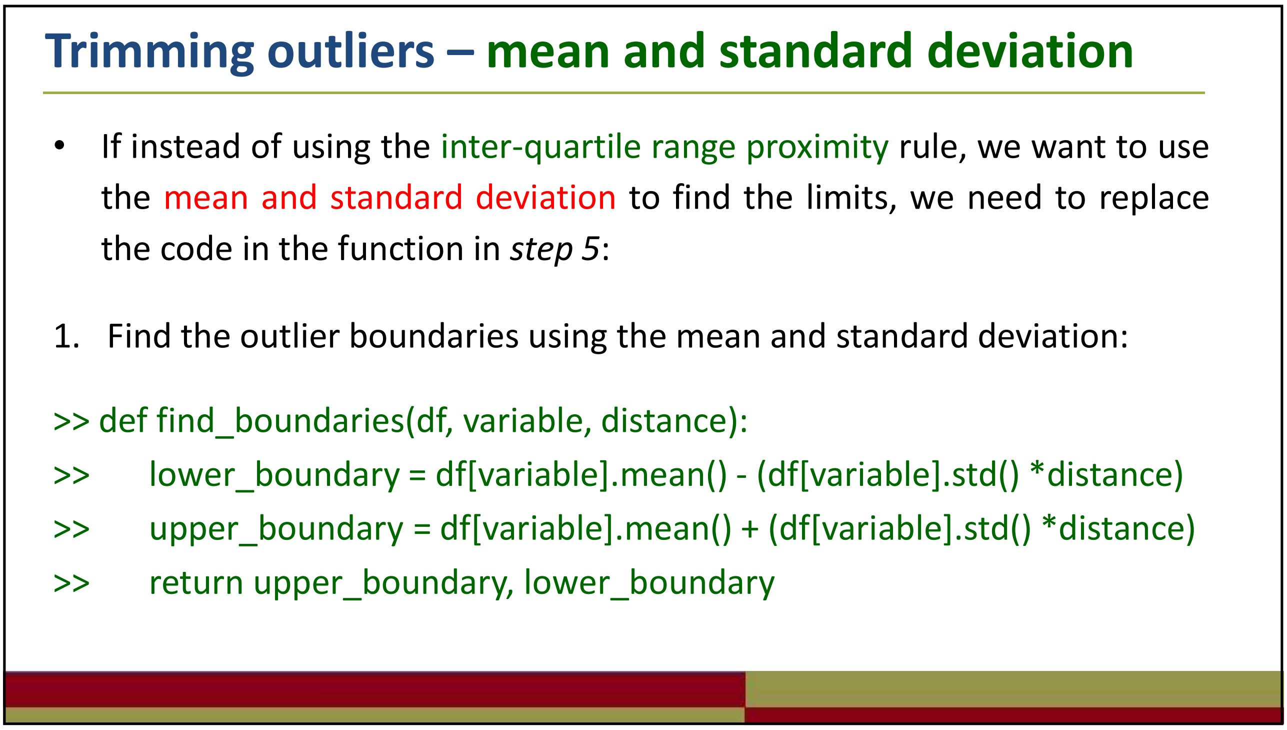
- With the pandas' quantile() method, we can calculate the values for the 25th (0.25) and 75th quantiles (0.75).
- We then used this function to return the upper and lower boundaries for the RM variable.
- To find the outliers of RM, we used *np.where*(), which produced a Boolean vector with True if the value was an outlier, that is, if the value was bigger or smaller than the upper or lower boundaries determined for RM.



Trimming outliers from the dataset-cont…

- Briefly, np.where() scanned the rows of the RM variable, and if the value was bigger than the upper boundary, it assigned True; whereas if the value was smaller, the second NumPy's where() method, nested in the first one, checked whether the value was smaller than the lower boundary, in which case, it also assigned True; otherwise, it assigned False.

- Finally, we used the loc[] method from pandas to remove the observations that contained outliers for RM. The ~ symbol used with the pandas' loc[] method removes from the DataFrame the outliers captured in the Boolean vector, outliers_RM.



Trimming outliers – mean and standard deviation

- If instead of using the inter-quartile range proximity rule, we want to use the mean and standard deviation to find the limits, we need to replace the code in the function in *step 5*:

1. Find the outlier boundaries using the mean and standard deviation:

```
>> def find_boundaries(df, variable, distance):
>>      lower_boundary = df[variable].mean() - (df[variable].std() *distance)
>>      upper_boundary = df[variable].mean() + (df[variable].std() *distance)
>>      return upper_boundary, lower_boundary
```

Trimming outliers – mean and standard deviation

To calculate the boundaries for the RM variable with the preceding function, we run the following code.

2. Calculate the boundaries for RM:

```
>> RM_upper_limit, RM_lower_limit = find_boundaries(boston, 'RM', 3)
```

Trimming outliers – alternate method of Quantile

- Alternatively, if we want to use quantiles to calculate the limits, we should write the function like in the next step.

3. Find the outlier boundaries using quantiles:

```
>> def find_boundaries(df, variable):
>>      lower_boundary = df[variable].quantile(0.05)
>>      upper_boundary = df[variable].quantile(0.95)
>>      return upper_boundary, lower_boundary
```

Trimming outliers – mean and standard deviation

4. Calculate the boundaries for RM:

```
>> RM_upper_limit, RM_lower_limit = find_boundaries(boston, 'RM')
```

- The rest of the procedure is identical to the one described in *step 8* and *step 9*, in the *How to do it...* section of the recipe

Trimming outliers – mean and standard deviation

5. Let's calculate the boundaries for the RM, LSTAT, and CRIM variables:

```
>> RM_upper_limit, RM_lower_limit = find_boundaries(boston, 'RM', 1.5)
>> LSTAT_upper_limit, LSTAT_lower_limit = find_boundaries(boston,'LSTAT', 1.5)
>> CRIM_upper_limit, CRIM_lower_limit = find_boundaries(boston, 'CRIM', 1.5)
```

Trimming outliers – mean and standard deviation

6. Let's create Boolean vectors that flag the outliers for each one of RM, LSTAT, and

CRIM:

```
>> outliers_RM = np.where(boston['RM'] > RM_upper_limit, True,
np.where(boston['RM'] < RM_lower_limit, True, False))
>> outliers_LSTAT = np.where(boston['LSTAT'] > LSTAT_upper_limit, True,
np.where(boston['LSTAT'] < LSTAT_lower_limit, True,False))
>> outliers_CRIM = np.where(boston['CRIM'] > CRIM_upper_limit, True,
np.where(boston['CRIM'] < CRIM_lower_limit, True, False))
```

Trimming outliers – mean and standard deviation

7. Finally, let's remove the observations with outliers in any of the variables:

```
>> boston_trimmed = boston.loc[~(outliers_RM + outliers_LSTAT +
outliers_CRIM)]
```

Standard Deviation

- In statistics, the **standard deviation** is a measure of the amount of variation or dispersion of a set of values
- A low standard deviation indicates that the values tend to be close to the mean (also called the expected value) of the set
- while a high standard deviation indicates that the values are spread out over a wider range

Standard Deviation

- The Formula for Standard Deviation

$$\text{Standard Deviation} = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n-1}}$$

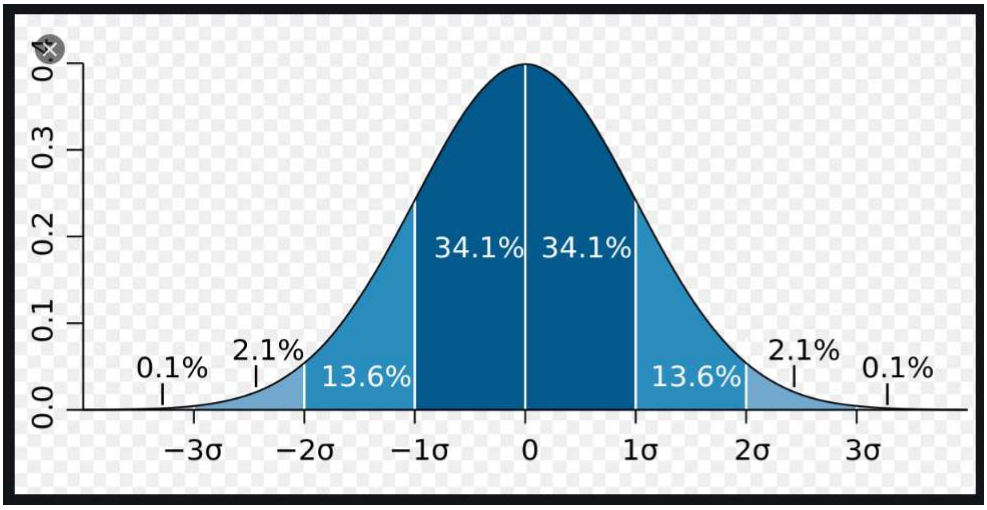
where:

$x_i = $ Value of the i^{th} point in the data set

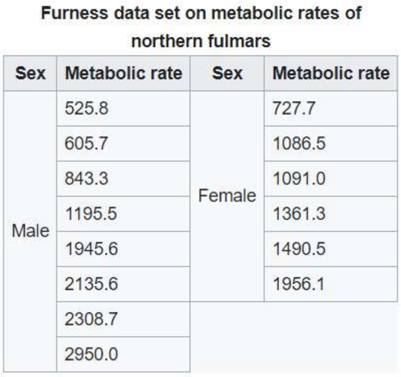
$\overline{x} = $ The mean value of the data set

$n = $ The number of data points in the data set

Standard Deviation

- 

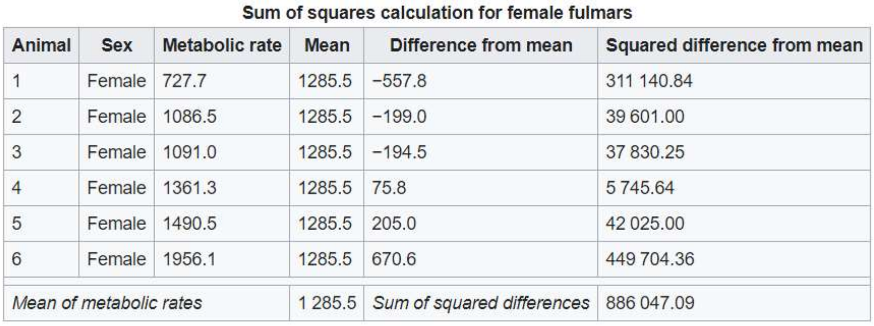
Standard Deviation-Example

- **Sample standard deviation of metabolic rate of northern fulmars**
- Furness and Bryant measured the resting metabolic rate for 8 male and 6 female breeding northern fulmars. The table shows the Furness data set.
- 

Furness data set on metabolic rates of northern fulmars

Sex	Metabolic rate	Sex	Metabolic rate
Male	525.8	Female	727.7
	605.7		1086.5
	843.3		1091.0
	1195.5		1361.3
	1945.6		1490.5
	2135.6		1956.1
	2308.7		
	2950.0		

Standard Deviation-Example

- 

Sum of squares calculation for female fulmars

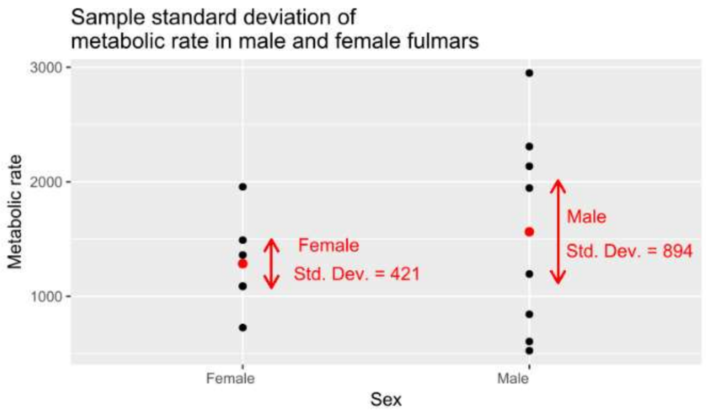
Animal	Sex	Metabolic rate	Mean	Difference from mean	Squared difference from mean
1	Female	727.7	1285.5	−557.8	311 140.84
2	Female	1086.5	1285.5	−199.0	39 601.00
3	Female	1091.0	1285.5	−194.5	37 830.25
4	Female	1361.3	1285.5	75.8	5 745.64
5	Female	1490.5	1285.5	205.0	42 025.00
6	Female	1956.1	1285.5	670.6	449 704.36
Mean of metabolic rates			1 285.5	*Sum of squared differences*	886 047.09

Standard Deviation-Example

$$s = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N-1}} = \sqrt{\frac{886047.09}{5}} = 420.96.$$

- For the male fulmars, a similar calculation gives a sample standard deviation of 894.37, approximately twice as large as the standard deviation for the females.
- The graph shows the metabolic rate data, the means (red dots), and the standard deviations (red lines) for females and males.

Standard Deviation-Example



Performing winsorization

- Winsorization, or winsorizing, is the process of transforming the data by limiting the extreme values, that is, the outliers, to a certain arbitrary value, closer to the mean of the distribution
- Winsorizing is different from trimming because the extreme values are not removed, but are instead replaced by other values. A typical strategy involves setting outliers to a specified percentile.

Performing winsorization

- For example, with 90% winsorization, we set all data below the 5th percentile to the value at the 5th percentile and all data above the 95th percentile to the value at the 95th percentile

Performing winsorization- How to do it...

1. Import the required Python libraries:

```
>> import pandas as pd
>> import numpy as np
>> import matplotlib.pyplot as plt
>> import seaborn as sns
>> from sklearn.datasets import load_boston
```

Performing winsorization- How to do it...

2. Let's load the Boston House Prices dataset from scikit-learn:

```
>> boston_dataset = load_boston()
```

3. Let's capture three of the variables, RM, LSTAT, and CRIM, in a pandas dataframe:

```
>> boston = pd.DataFrame(boston_dataset.data,
    columns=boston_dataset.feature_names)[['RM', 'LSTAT', 'CRIM']]
```

Performing winsorization- How to do it...

4. Let's make a function to winsorize a variable to arbitrary upper and lower limits:

```
>> def winsorize(df, variable, upper_limit, lower_limit):
>>      return np.where(df[variable] > upper_limit, upper_limit,
>>      np.where(df[variable] < lower_limit, lower_limit,df[variable]))
```

5. Let's winsorize the RM variable:

```
>> boston['RM']= winsorize(boston, 'RM', boston['RM'].quantile(0.95),
>> boston['RM'].quantile(0.05))
```

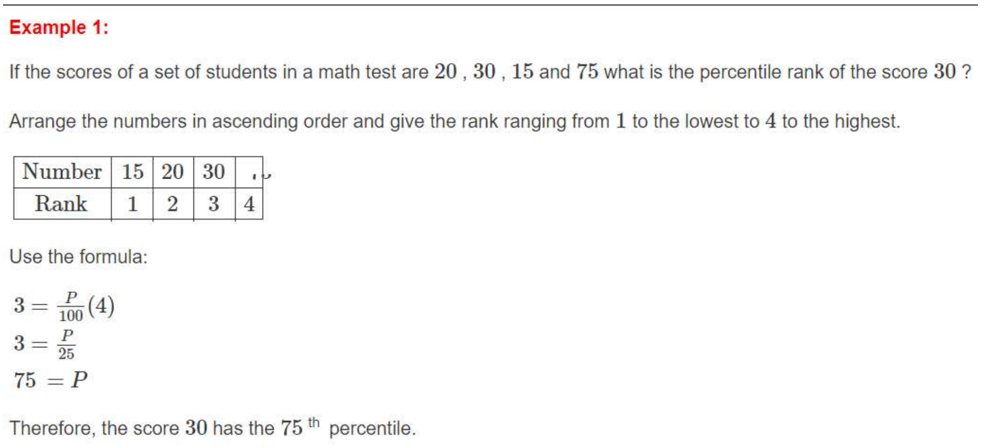
Percentile

- A percentile is a comparison score between a particular score and the scores of the rest of a group.
- It shows the percentage of scores that a particular score surpassed. For example, if you score 75 points on a test, and are ranked in the 85th percentile, it means that the score 75 is higher than 85% of the scores.

The percentile rank is calculated using the formula

Percentile

- The percentile rank is calculated using the formula $R = \frac{P}{100}(N)$
- where P is the desired percentile and N is the number of data points.



Example 1:

If the scores of a set of students in a math test are 20 , 30 , 15 and 75 what is the percentile rank of the score 30 ?

Arrange the numbers in ascending order and give the rank ranging from 1 to the lowest to 4 to the highest.

Number	15	20	30	[illegible]
Rank	1	2	3	4

Use the formula:

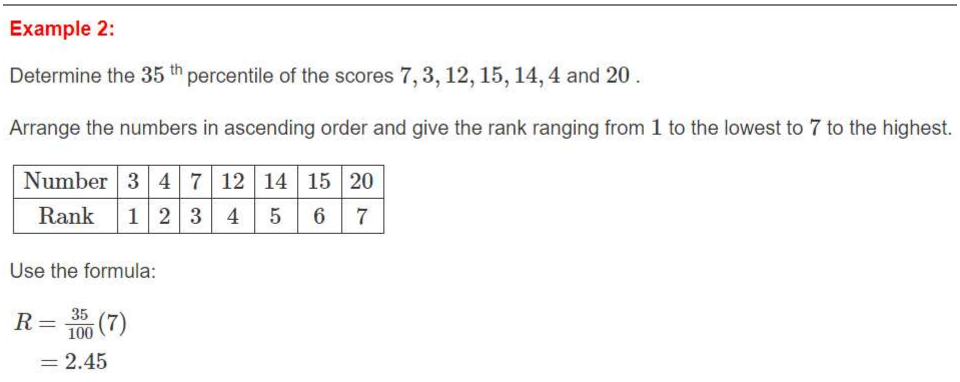
$$3 = \frac{P}{100}(4)$$
$$3 = \frac{P}{25}$$
$$75 \ = P$$

Therefore, the score 30 has the 75 th percentile.

Percentile

- Note that, if the percentile rank R is an integer, the Pth percentile would be the score with rank R when the data points are arranged in ascending order.
- If R is not an integer, then the Pth percentile is calculated as shown.
- Let I be the integer part and be the decimal part of D of R . Calculate the scores with the ranks I and I+1 . Multiply the difference of the scores by the decimal part of R . The Pth percentile is the sum of the product and the score with the rank I .

Percentile



Example 2:

Determine the 35^{th} percentile of the scores $7, 3, 12, 15, 14, 4$ and 20 .

Arrange the numbers in ascending order and give the rank ranging from 1 to the lowest to 7 to the highest.

Number	3	4	7	12	14	15	20
Rank	1	2	3	4	5	6	7

Use the formula:

$$\begin{aligned} R &= \tfrac{35}{100}(7) \\ &= 2.45 \end{aligned}$$

- The integer part of R is 2 , calculate the score corresponding to the ranks 2 and 3 . They are 4 and 7 . The product of the difference and the decimal part is 0.45(7−4) = 1.35 .
- Therefore, the 35th percentile is 2+1.35 = 3.35 .