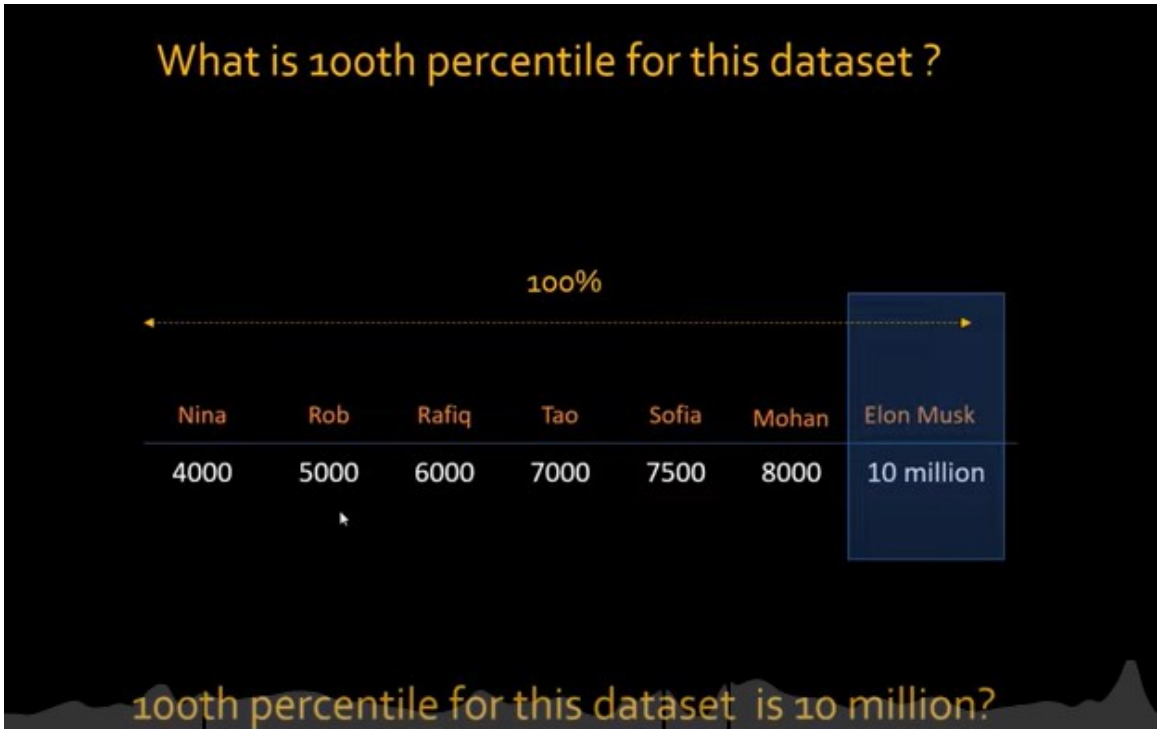
Median, Mean, Mode, Percentile

Name	Monthly Income ($)
Rob	5000
Rafiq	6000
Nina	4000
Sofia	7500
Mohan	8000
Tao	7000
Average	6250

Name	Monthly Income ($)
Rob	5000
Rafiq	6000
Nina	4000
Sofia	7500
Mohan	8000
Tao	7000
Elon Musk	10 million
Average	1.43 million


50%
50%
Nina
Rob
Rafiq
Tao
Sofia
Mohan
Elon Musk
4000
5000
6000
7000
7500
8000
10 million
median
50th percentile for this dataset is 7000


What is 100th percentile for this dataset ?
100%
Nina
Rob
Rafiq
Tao
Sofia
Mohan
Elon Musk
4000
5000
6000
7000
7500
8000
10 million
100th percentile for this dataset is 10 million?

What is 25th percentile for this dataset ?

Total values =7

25% of 7 is 1.75 which is approximately 2 data points.

Nina	Rob	Rafiq	Tao	Sofia	Mohan	Elon Musk
4000	5000	6000	7000	7500	8000	10 million

25^{th} percentile is 5500

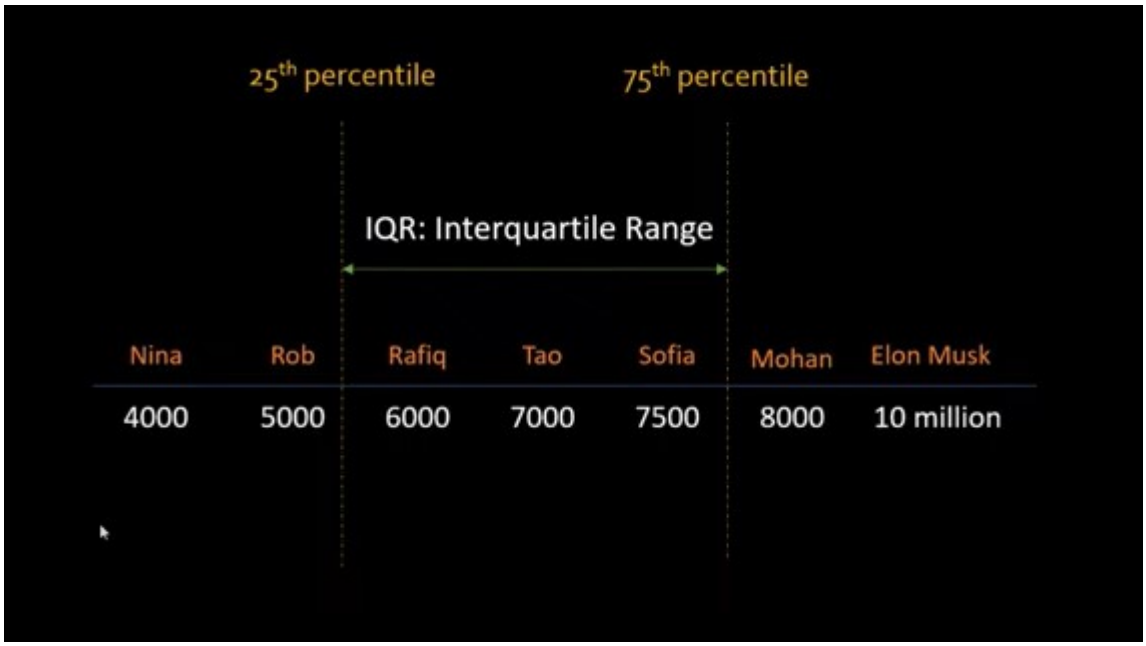
What is 75th percentile for this dataset ?

Total values =7

75% of 7 is 5.25 ~ 5 data points.

Nina	Rob	Rafiq	Tao	Sofia	Mohan	Elon Musk
4000	5000	6000	7000	7500	8000	10 million

75^{th} percentile is 7750



3. How to Find a Percentile

Need help? Check out our tutoring page!

Example question: Find out where the 25th percentile is in the above list.

Step 1: Calculate what rank is at the 25th percentile. Use the following formula:
Rank = Percentile / 100 * (number of items + 1)
Rank = 25 / 100 * (8 + 1) = 0.25 * 9 = 2.25.
A rank of 2.25 is at the 25th percentile. However, there isn't a rank of 2.25 (ever heard of a high school rank of 2.25? I haven't!), so you must either round up, or round down. As 2.25 is closer to 2 than 3, I'm going to round down to a rank of 2.

Step 2: Choose either definition 1 or 2:
Definition 1: The lowest score that is **greater** than 25% of the scores. That equals a score of 43 on this list (a rank of 3).
Definition 2: The smallest score that is greater than **or equal to** 25% of the scores. That equals a score of 33 on this list (a rank of 2).

Depending on which definition you use, the 25th percentile could be reported at 33 or 43! A third definition attempts to correct this possible misinterpretation:

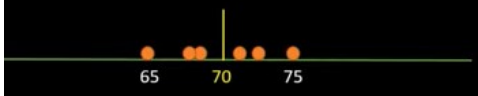
```
df.income.quantile(0.25, interpolation="higher")
```

What is Standard Deviation and Mean Absolute Deviation

History Test

Name	Score
Mohan	75
Andrea	72
Sofia	68
Joe	65
Virat	67
Abdul	73

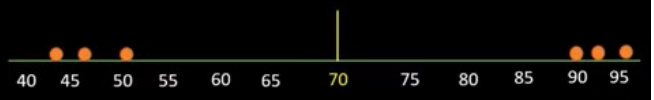
Average = 70



Math Test

Name	Score
Mohan	93
Andrea	96
Sofia	43
Joe	47
Virat	51
Abdul	90

Average = 70



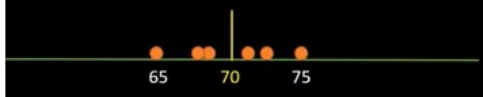
History Test

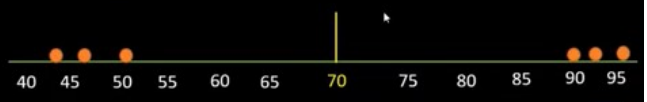
Name	Score	Abs (Score – Avg)
Mohan	75	5
Andrea	72	2
Sofia	68	2
Joe	65	5
Virat	67	3
Abdul	73	2
	Mean	3.16

Math Test

Name	Score	Abs (Score – Avg)
Mohan	93	23
Andrea	96	26
Sofia	43	27
Joe	47	23
Virat	51	19
Abdul	90	20
	Mean	23

Mean Absolute Deviation

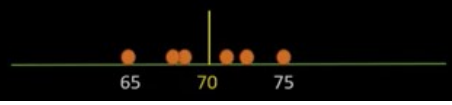




Name	Score	Abs (Score – Avg)
Mohan	75	5
Andrea	72	2
Sofia	68	2
Joe	65	5
Virat	67	3
Abdul	73	3
	MAD	3.33

History Test

Name	Score	Abs (Score – Avg)
Mohan	83	13
Andrea	70	0
Sofia	70	0
Joe	63	7
Virat	70	0
Abdul	70	0
	MAD	3.33



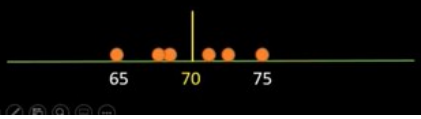
Average = 70

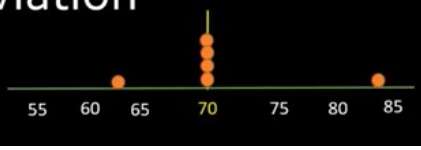
Name	Score	Abs (Score – Avg)	(Score – Avg)^2
Mohan	75	5	25
Andrea	72	2	4
Sofia	68	2	4
Joe	65	5	25
Virat	67	3	9
Abdul	73	3	9
	Avg	12.66	
	$\sqrt{Avg}$	3.55	

Average = 70

Name	Score	Abs (Score – Avg)	(Score – Avg)^2
Mohan	83	13	169
Andrea	70	0	0
Sofia	70	0	0
Joe	63	7	49
Virat	70	0	0
Abdul	70	0	0
	Avg	36.33	
	$\sqrt{Avg}$	6.02	

Standard Deviation





Formula

$$\sigma = \sqrt{\frac{\sum(x_i - \mu)^2}{N}}$$

σ = population standard deviation

N = the size of the population

x_i = each value from the population

μ = the population mean

From the web

To find the **standard deviation**, we take the square root of the variance. From learning that **SD** = 13.31, we can say that each score deviates from the mean by 13.31 points on average. Sep 17, 2020

https://www.scribbr.com › statistics › standard-deviation

Standard Deviation | A Step by Step Guide with Formulas

Both measure the dispersion of your data by computing the distance of the data to its mean.

1. the **mean absolute deviation** is using norm L1 (it is also called Manhattan distance or rectilinear distance)
2. the **standard deviation** is using norm L2 (also called Euclidean distance)

The difference between the two norms is that the **standard deviation** is calculating the square of the difference whereas the **mean absolute deviation** is only looking at the absolute difference. Hence large outliers will create a higher dispersion when using the standard deviation instead of the other method. The Euclidean distance is indeed also more often used. The main reason is that the **standard deviation** have nice properties when the data is normally distributed. So under this assumption, it is recommended to use it. However people often do this assumption for data which is actually not normally distributed which creates issues. If your data is not normally distributed, you can still use the standard deviation, but you should be careful with the interpretation of the results.

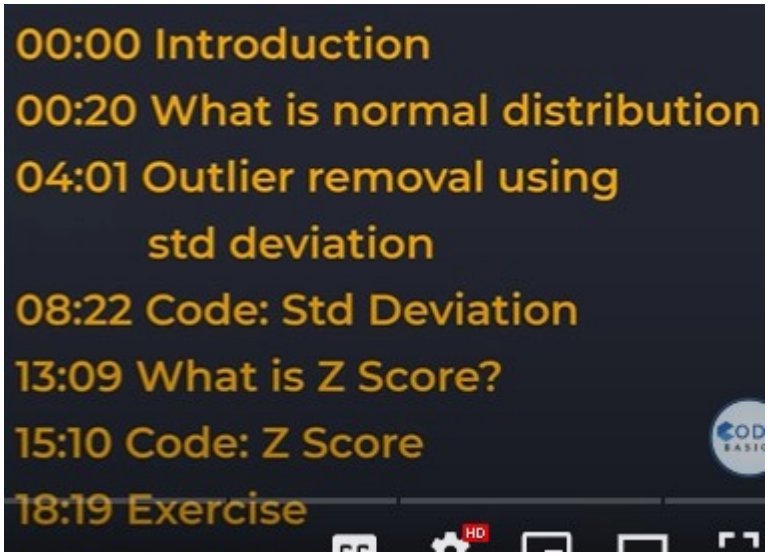
Finally you should know that both measures of dispersion are particular cases of the Minkowski distance, for p=1 and p=2. You can increase p to get other measures of the dispersion of your data.

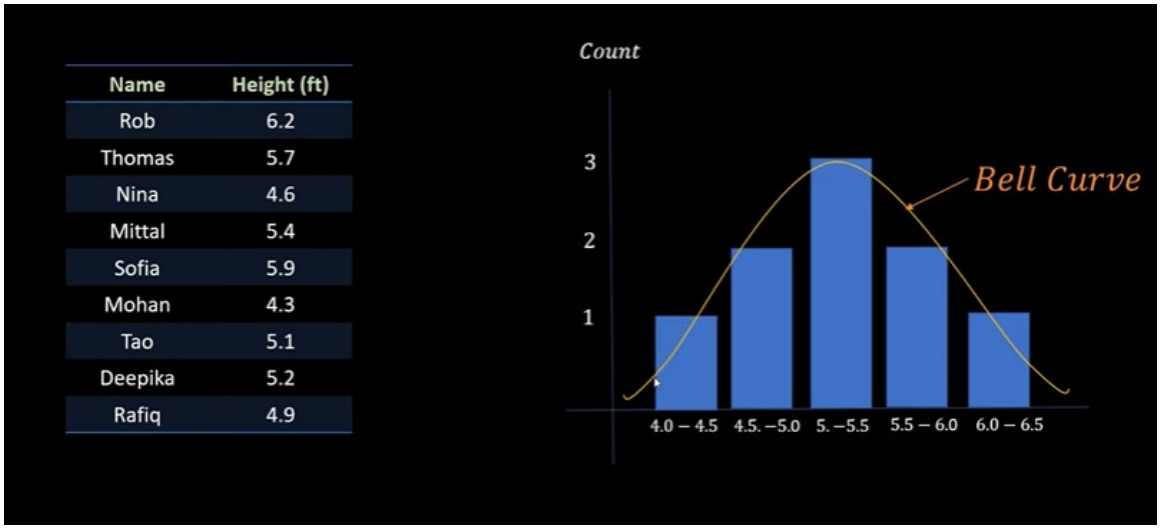
Share Cite Improve this answer Follow edited Mar 5 '14 at 3:04 answered Mar 5 '14 at 2:51

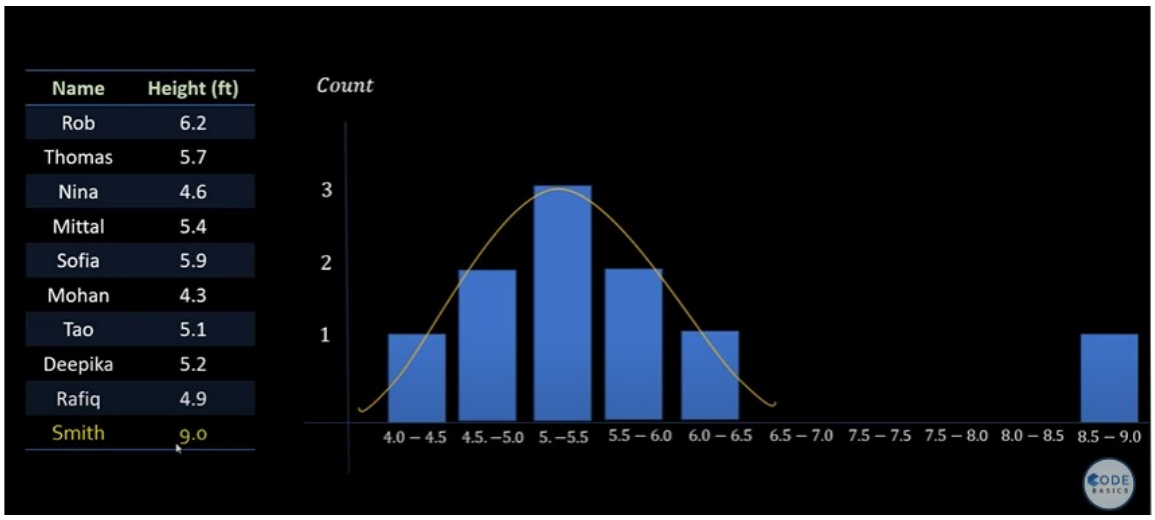
RockScience

2,613 4 26 44

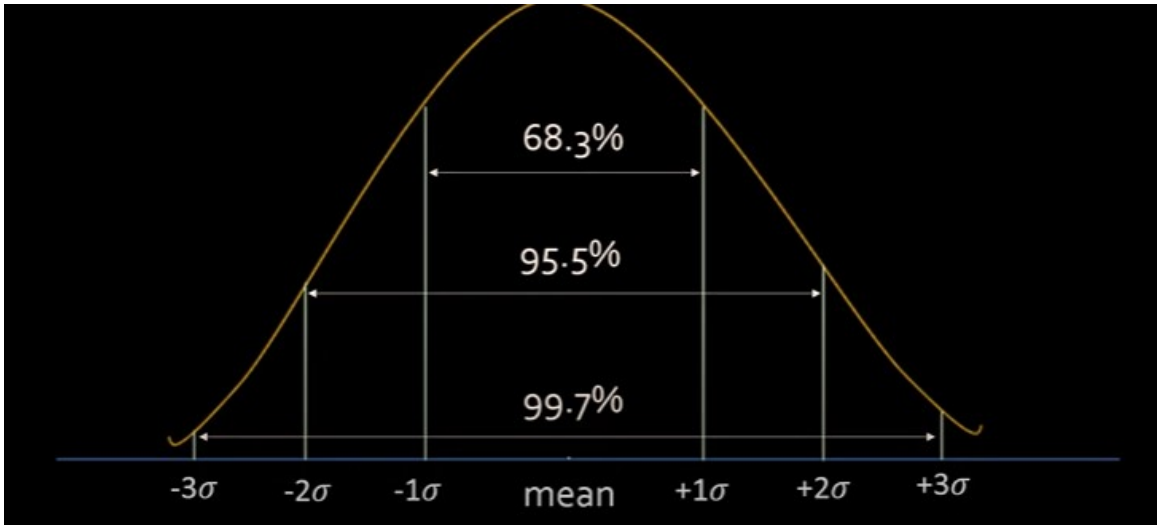
Normal Distribution and Z Score


00:00 Introduction
00:20 What is normal distribution
04:01 Outlier removal using
std deviation
08:22 Code: Std Deviation
13:09 What is Z Score?
15:10 Code: Z Score
18:19 Exercise


Name
Height (ft)
Rob
6.2
Thomas
5.7
Nina
4.6
Mittal
5.4
Sofia
5.9
Mohan
4.3
Tao
5.1
Deepika
5.2
Rafiq
4.9
Count
3
2
1
Bell Curve
4.0 – 4.5
4.5. –5.0
5. –5.5
5.5 – 6.0
6.0 – 6.5


Name
Height (ft)
Rob
6.2
Thomas
5.7
Nina
4.6
Mittal
5.4
Sofia
5.9
Mohan
4.3
Tao
5.1
Deepika
5.2
Rafiq
4.9
Smith
9.0
Count
3
2
1
4.0 – 4.5
4.5. –5.0
5. –5.5
5.5 – 6.0
6.0 – 6.5
6.5 – 7.0
7.5 – 7.5
7.5 – 8.0
8.0 – 8.5
8.5 – 9.0

What formula do we use to remove outliers?




```
sn.histplot(df.height, kde=True)
```

```
In [8]: std_deviation = df.height.std()
        std_deviation
Out[8]: 3.847528120795573

In [9]: mean - 3*std_deviation
Out[9]: 54.824975392479274

In [10]: mean + 3*std_deviation
Out[10]: 77.91014411725271
```

Z Score

Z Score: How many standard deviation away a datapoint is from mean

Name	Height (ft)	Z Score
Rob	6.2	(6.2-5.25) / 0.61 = 1.53
Thomas	5.7	(5.7-5.25) / 0.61 = 0.72
Nina	4.6	(4.6 – 5.25) / 0.61 = -1.06
Mittal	5.4	(5.4 – 5.25) / 0.61 = 0.23
Sofia	5.9	(5.9 – 5.25) / 0.61 = 1.04
Mohan	4.3	(4.3 – 5.25) / 0.61 = -1.55
Tao	5.1	(5.1 – 5.25) / 0.61 = -0.25
Deepika	5.2	(5.2 – 5.25) / 0.61 = -0.09
Rafiq	4.9	(4.9 – 5.25) / 0.61 = -0.58

$$z = \frac{x - u}{\sigma}$$

$u = mean$

$\sigma = std\ dev$

$Average = 5.25$

$Standard\ Deviation = 0.61$

What is logarithm?

125 $

With an initial or base investment of 5$ and 5x return, how many years will it take for my money to become 125$?

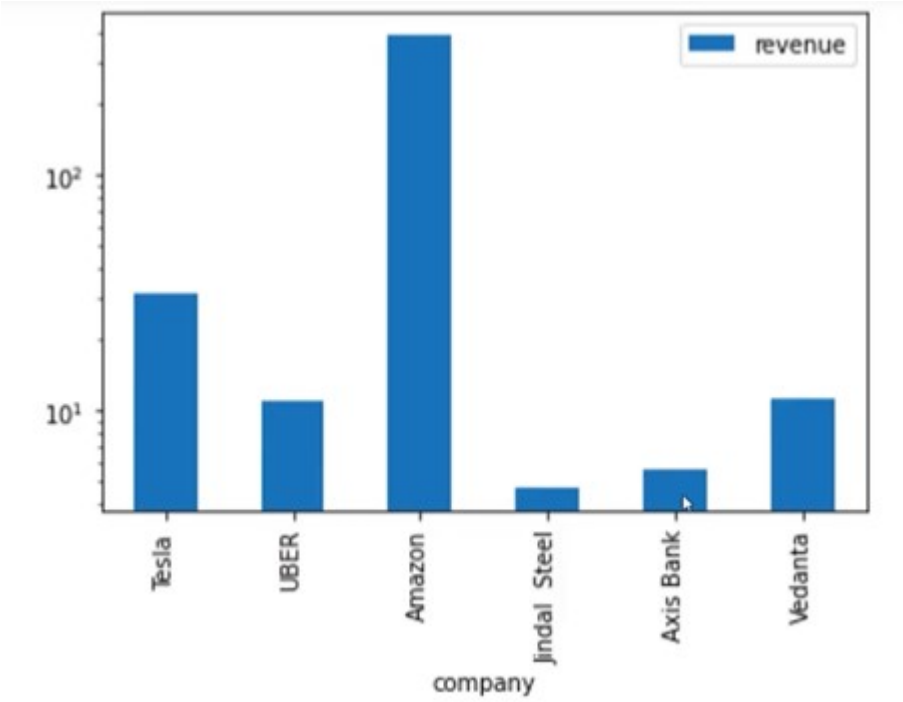
$\log_5 125 \rightarrow 3$

$\log_{10} 10 \rightarrow 1$

$\log_{10} 100 \rightarrow \log_{10} 10^2 \rightarrow 2 \log_{10} 10 \rightarrow 2$

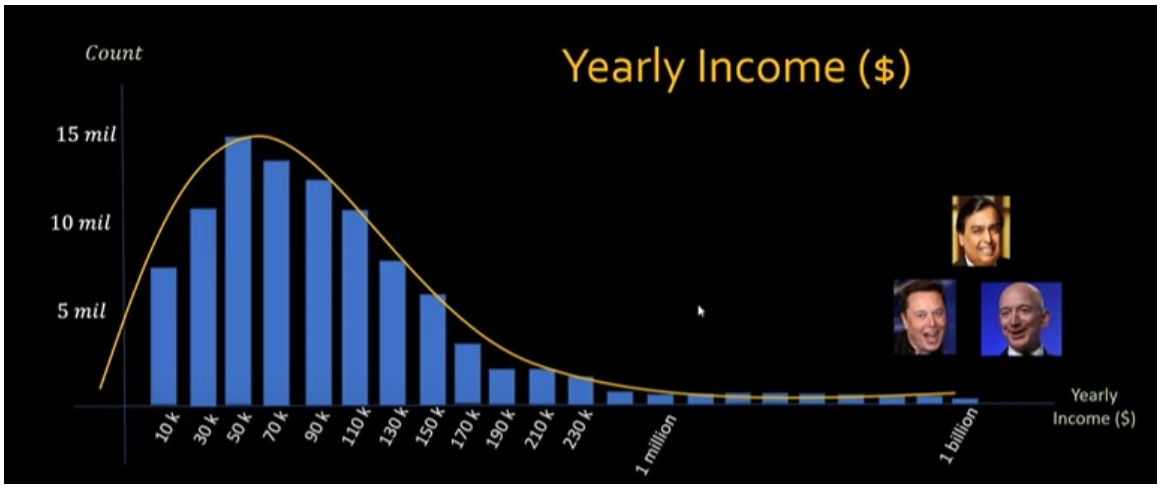
$\log_{10} 1000 \rightarrow \log_{10} 10^3 \rightarrow 3 \log_{10} 10 \rightarrow 3$

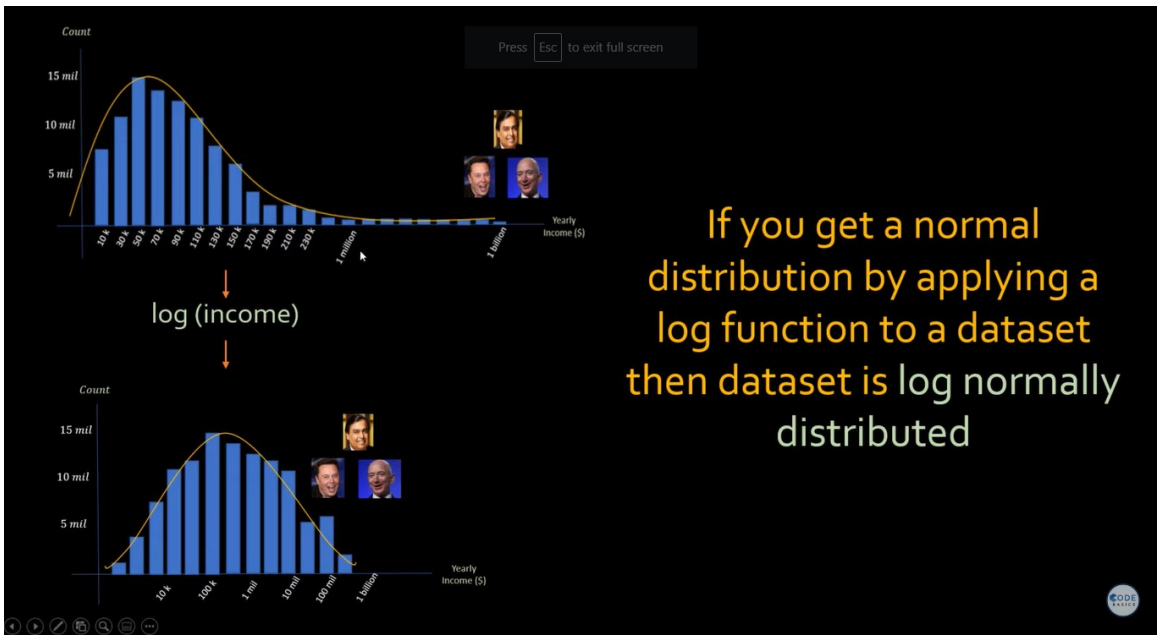
```
df.plot(x='company', y='revenue',kind='bar', logy=True)
```



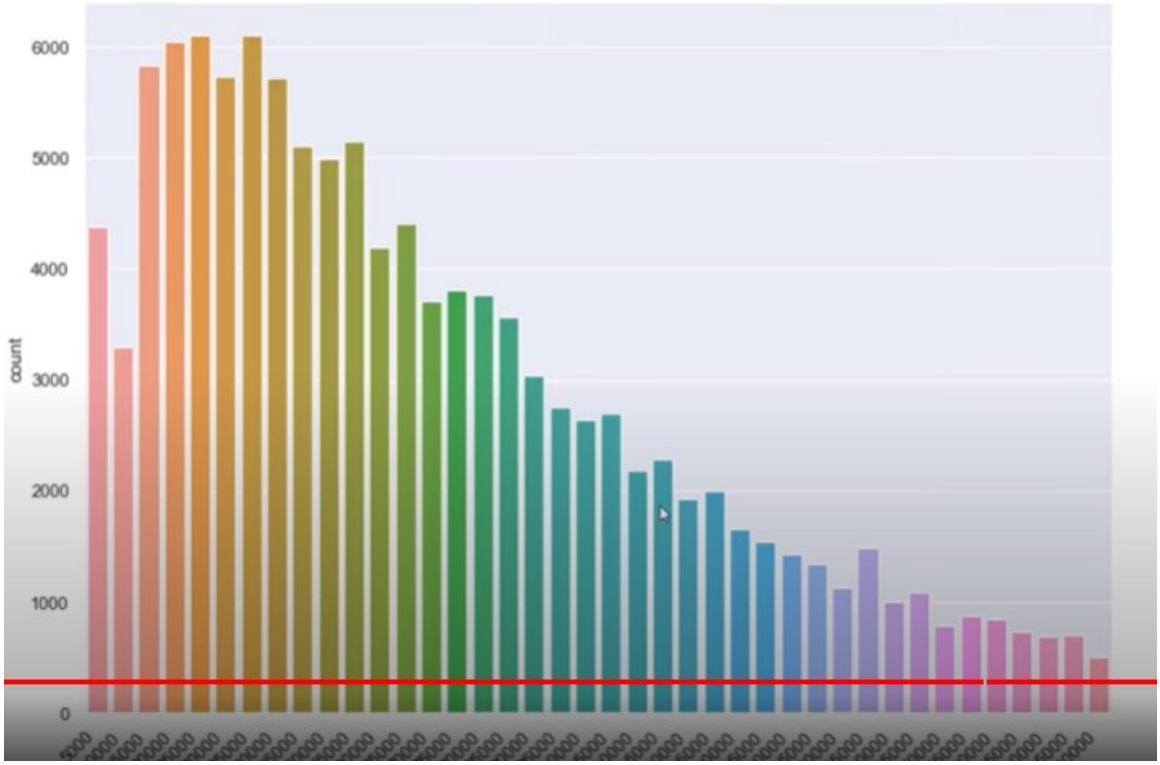
person name	credit score	income	age	loan approved?	log income
Rob	750	80000	32	Y	4.903089987
Tom	310	32000	45	N	4.505149978
Xi	475	77000	33	Y	4.886490725
Mohan	600	65000	51	N	4.812913357
Pooja	820	**550000**	35	Y	5.740362689
Sofiya	780	75000	31	Y	4.875061263

Log normal distribution

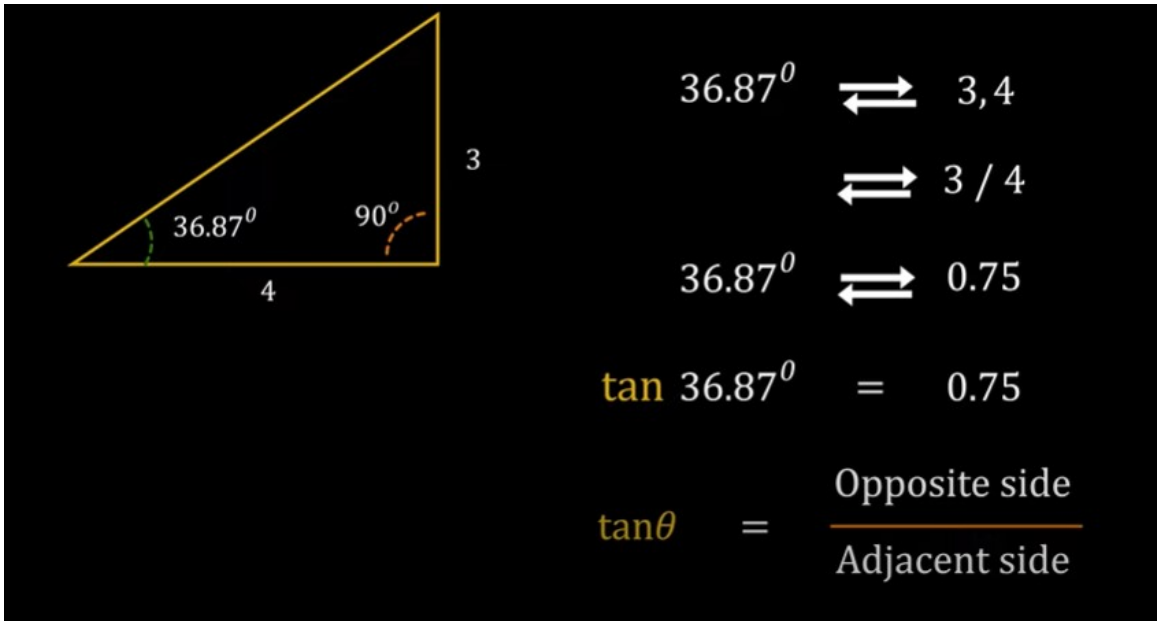


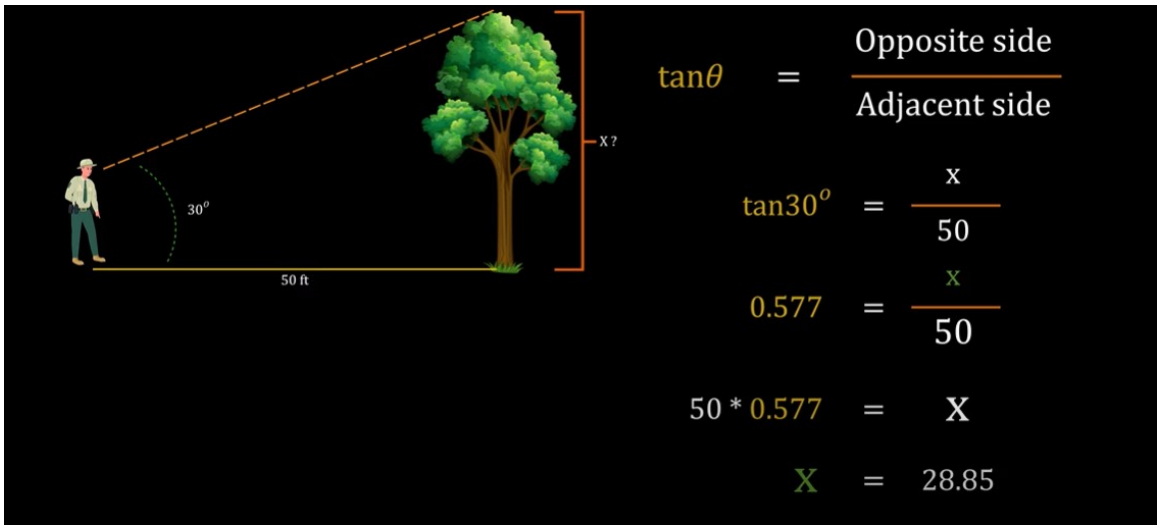


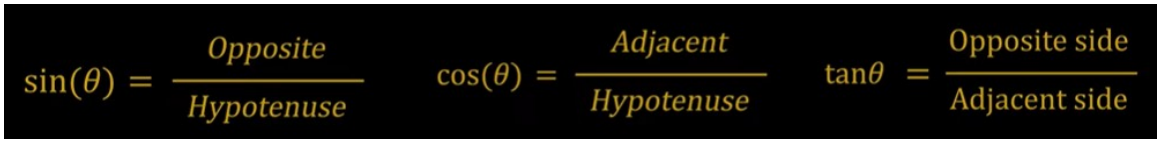

```
sns.set(rc={'figure.figsize':(11.7,8.27)})
g = sns.barplot(x='income',y='count',data=df)
g.set_xticklabels(g.get_xticklabels(),
                  rotation=45,
                  horizontalalignment='right');
```



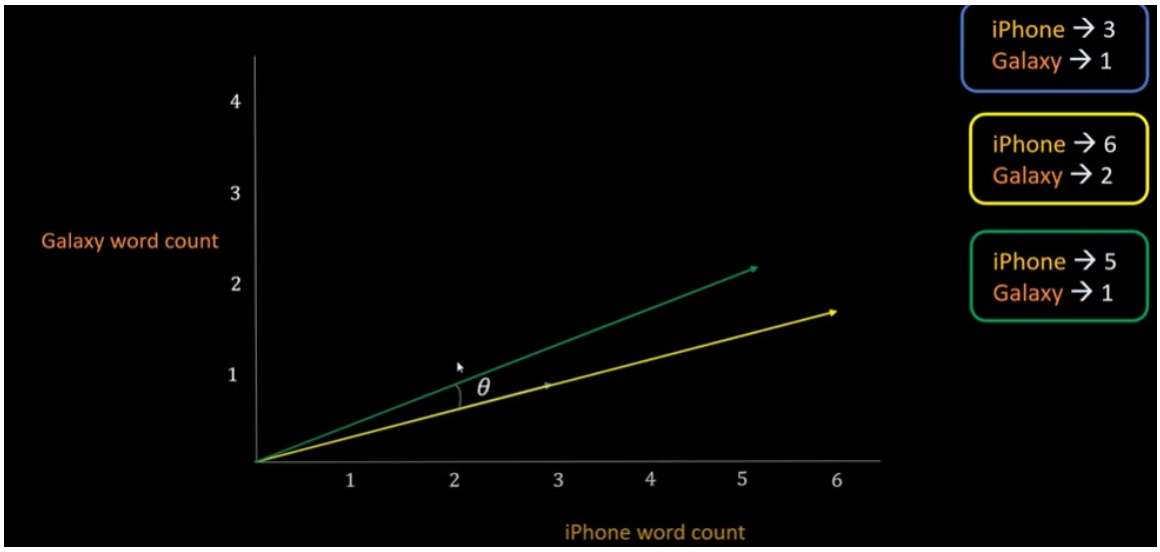
sin cos tan explained. Explanation using real life example

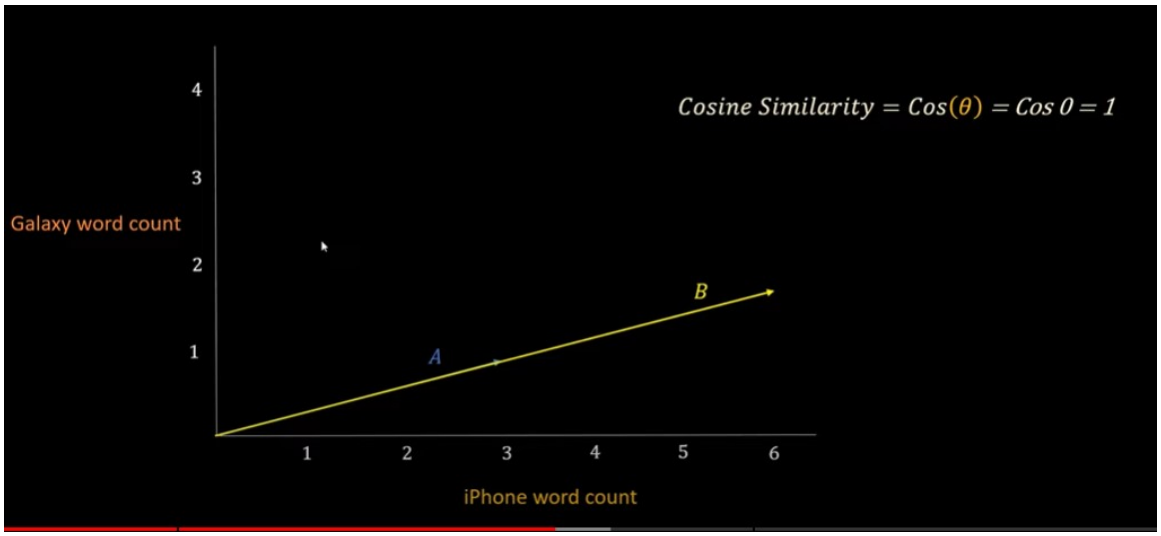






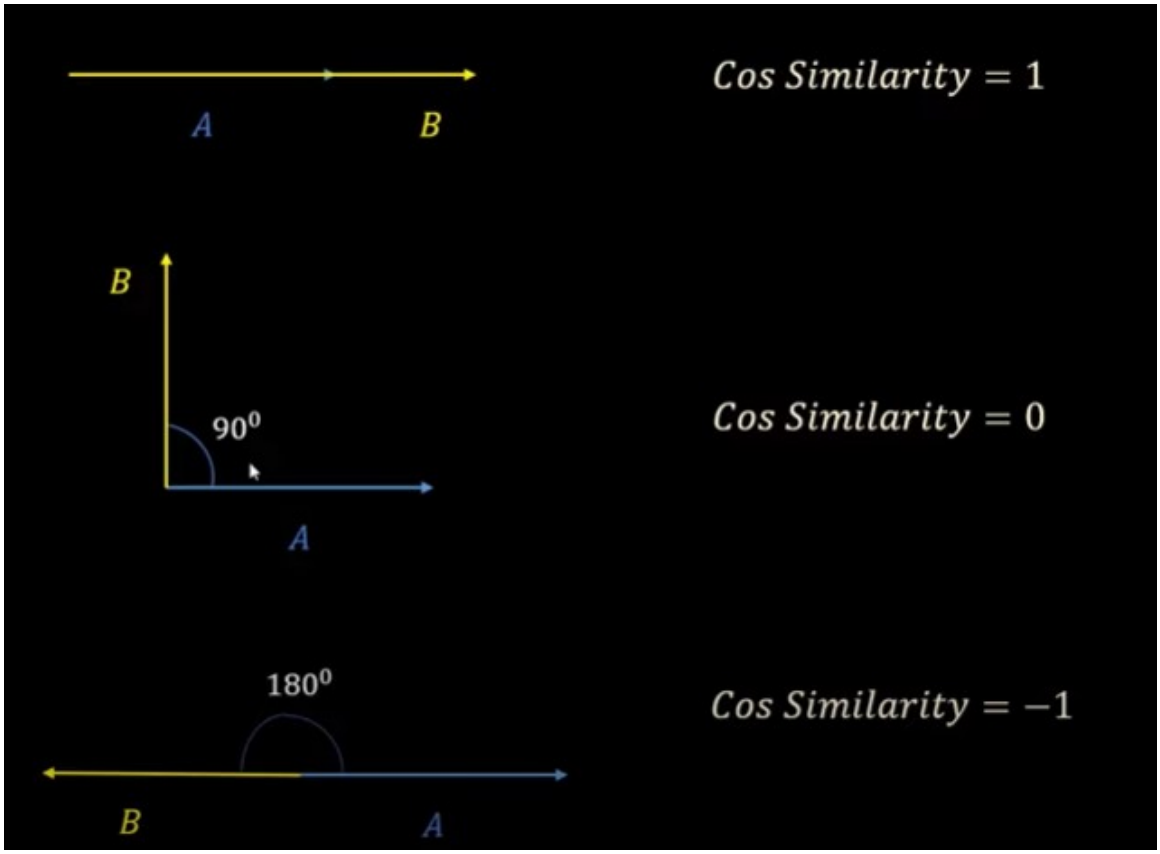
Cosine similarity, cosine distance explained



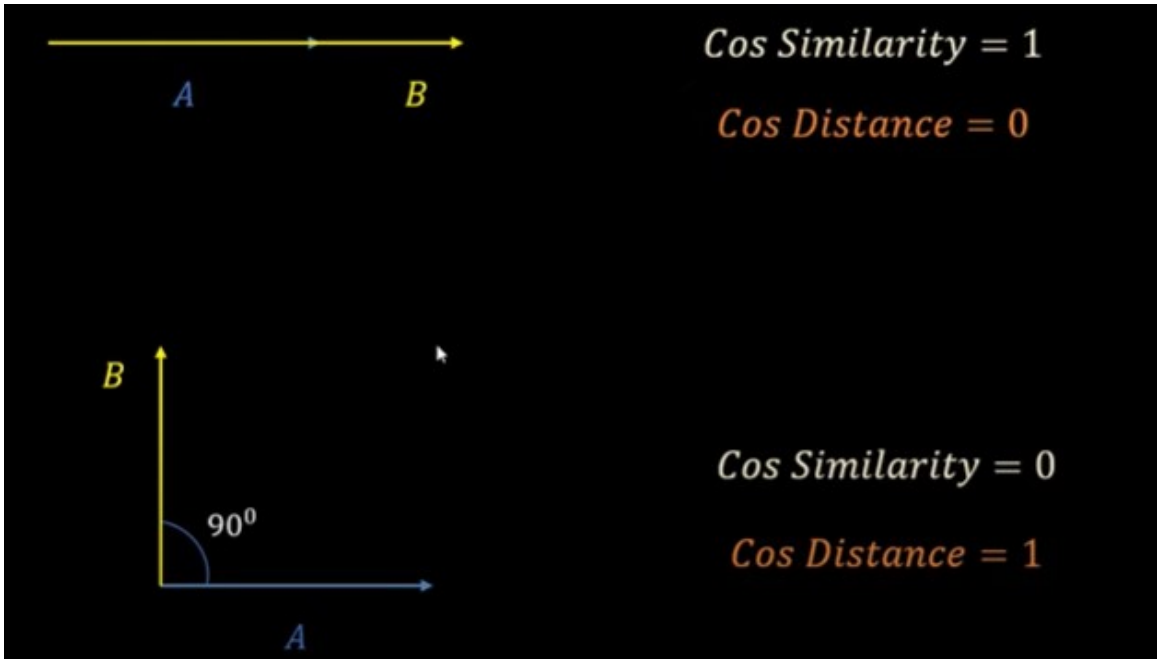


$$Cosine\ Similarity = \frac{A \cdot B}{\|A\|\|B\|}$$

$$Cosine\ Similarity = \frac{\|A\|\|B\| * Cos(\theta)}{\|A\|\|B\|}$$

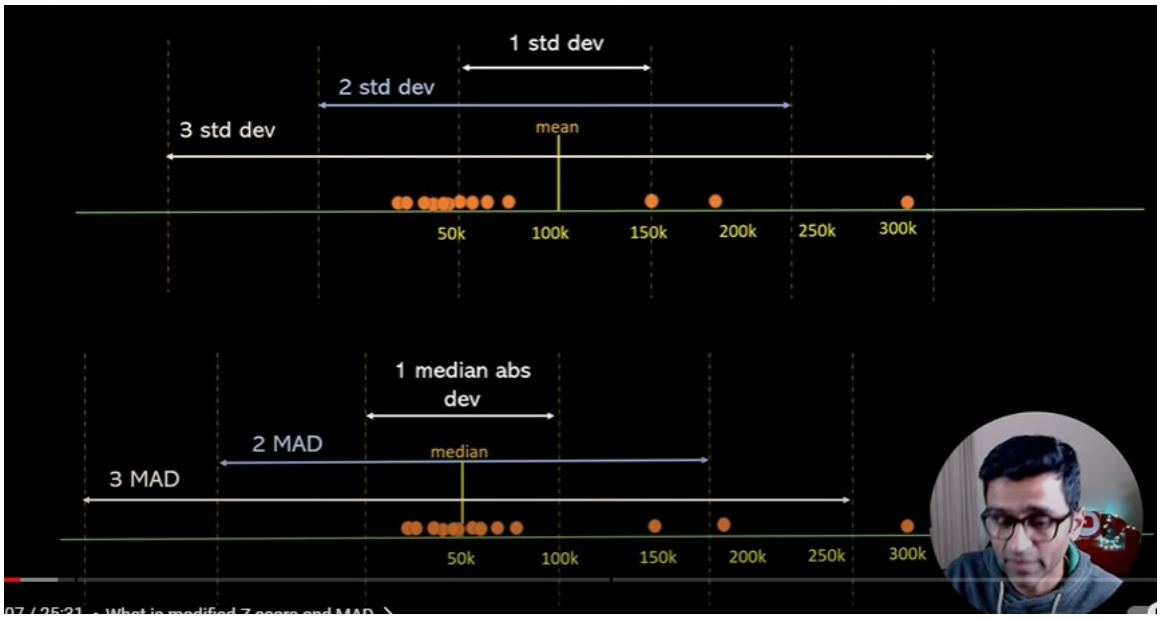


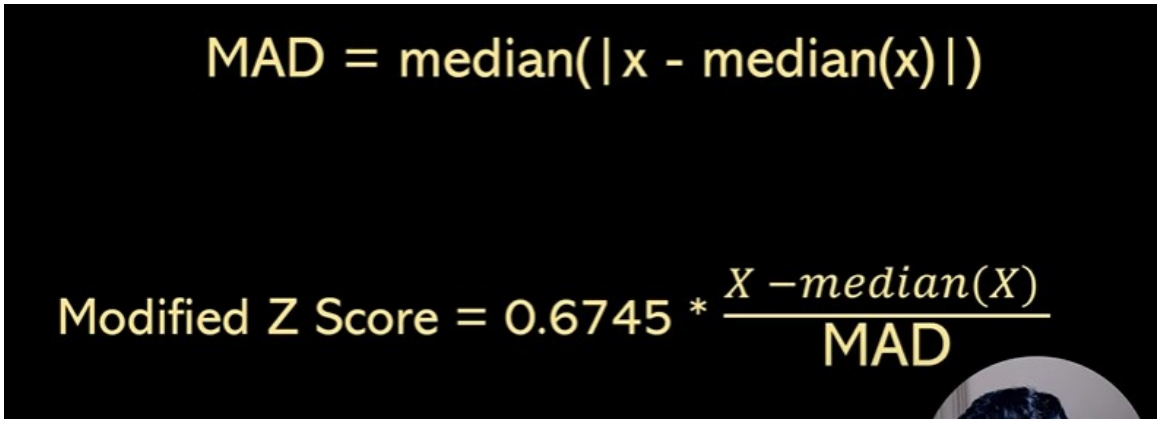
$$Cosine\ Distance = 1 - Cosine\ Similarity$$



```
from sklearn.metrics.pairwise import cosine_similarity, cosine_distances
```

Simple explanation of Modified Z Score | Modified Z Score to detect outliers





Height	Z Score	Z score > 3		Height	Height - median he	Mod Z Score	mod Z score > 3.5 ?
5.2	-0.5671	FALSE		5.2	0.3	-0.3	FALSE
4.9	-0.7397	FALSE		4.9	0.6	-0.6	FALSE
4.5	-0.96982	FALSE		4.5	1.0	-1.0	FALSE
5.5	-0.3945	FALSE		5.5	0.0	0.0	FALSE
7.0	0.468474	FALSE		7.0	1.5	1.4	FALSE
10.0	2.19443	FALSE		10.0	4.5	4.3	TRUE
6.2	0.008219	FALSE		6.2	0.7	0.7	FALSE
6.2	Average			5.5	Median Height		
1.738167	Std dev			0.7	MAD		