University of Lancashire

Data Analysis using Statistics

Karen O'Shea

Where opportunity creates success

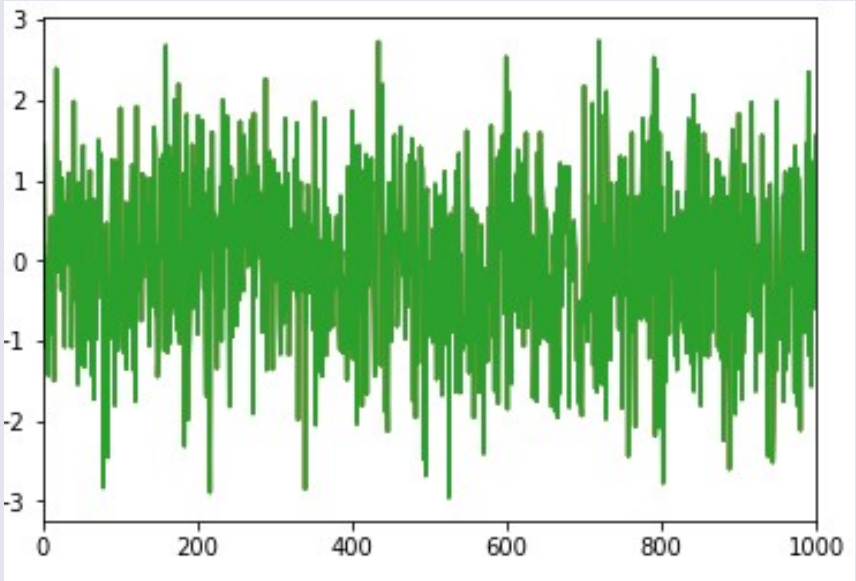
Learning Objectives

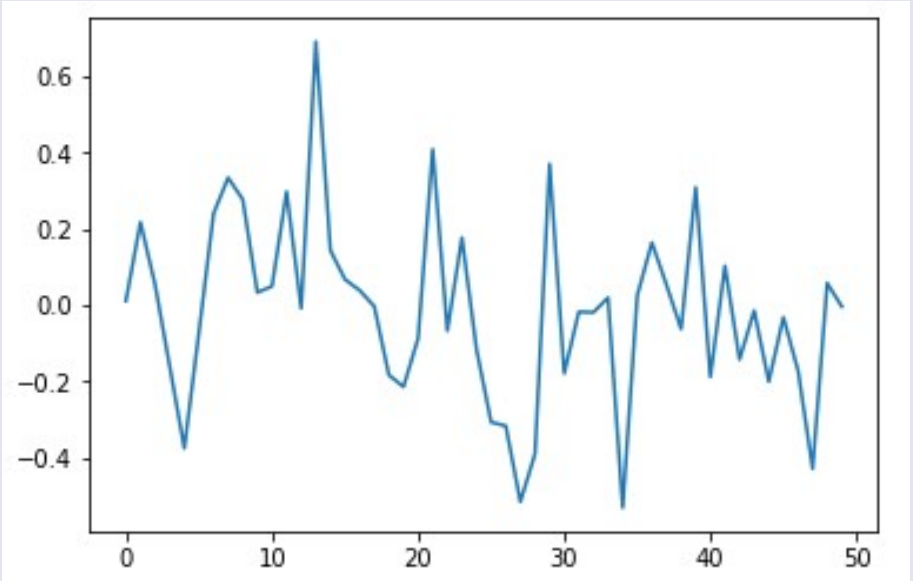
University of Lancashire

Evaluate basic statistics for data analysis

Recap: Why choose appropriate visuals?

University of Lancashire



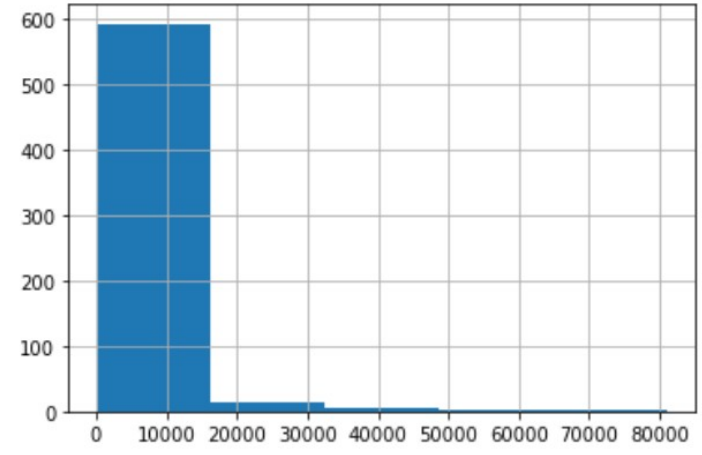


Recap: Comparison – Histogram ‘bins’ (intervals)
(Loan Prediction Dataset)

University of Lancashire

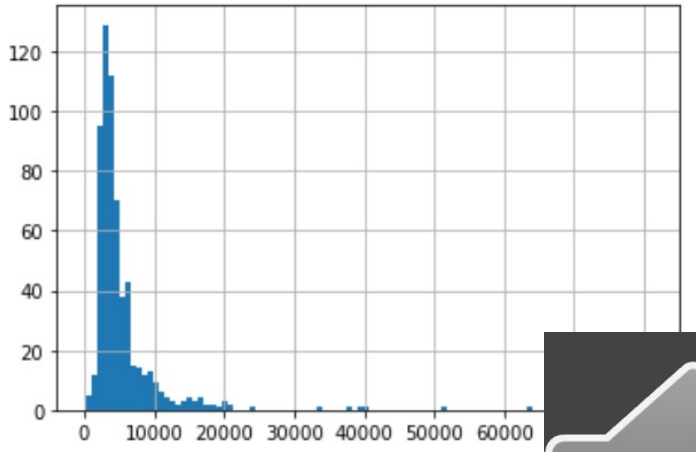
```
df['ApplicantIncome'].hist(bins=5)
```

```
<AxesSubplot:>
```




```
df['ApplicantIncome'].hist(bins=100)
```

```
<AxesSubplot:>
```

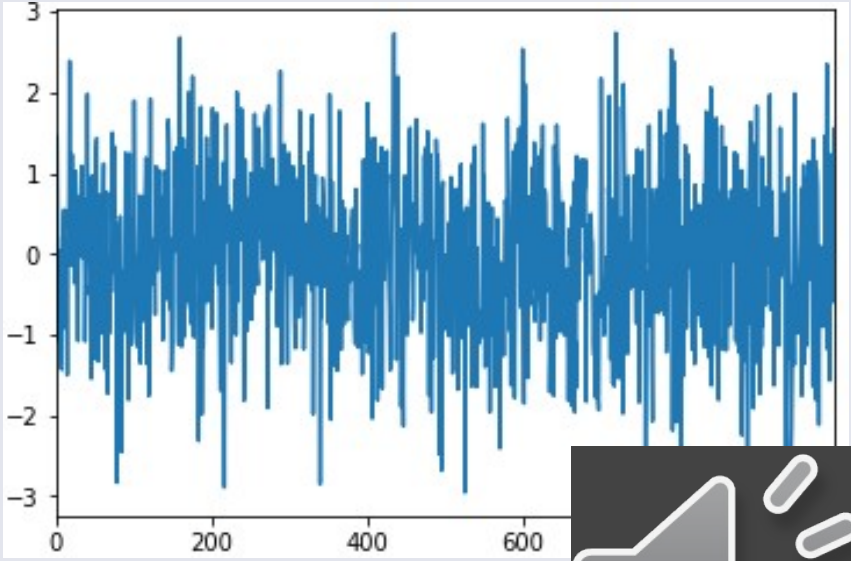


Recap: Hidden Trends – Smooth Data

Smoothing by bin – each value in a bin is replaced by the mean value of the bin.

Smoothing by bin median – each bin value is replaced by its bin median value.

Smoothing by bin boundary – the minimum and maximum values in a given bin are identified as the bin boundaries.



Some Statistics to assist with Data Analysis

- **Mean**
  - The average total of numbers. The sum of the numbers is divided by the number of observations
- **Median**
  - The central number among the total observations arranged from the least to the highest in ascending order….4, 7, 11, the median would be 7.
- **Percentiles**
  - Percentage of observations lying beneath it. Example: 23, 38, 49, 50. Number of values below x divided by the total number of observations. So 49 has a percentile of 50.
- **Standard Deviation**

Task: Calculating Standard Deviation

Calculate:

- Variable (x): 6, 2, 3, 1
- Find the mean
- Square (x-mean)
- Sum results
- Divide by number of data points
- Square root to find SD

$$\mathrm{SD} = \sqrt{\frac{\sum |x - \mu|^2}{N}}$$

Calculating Standard Deviation

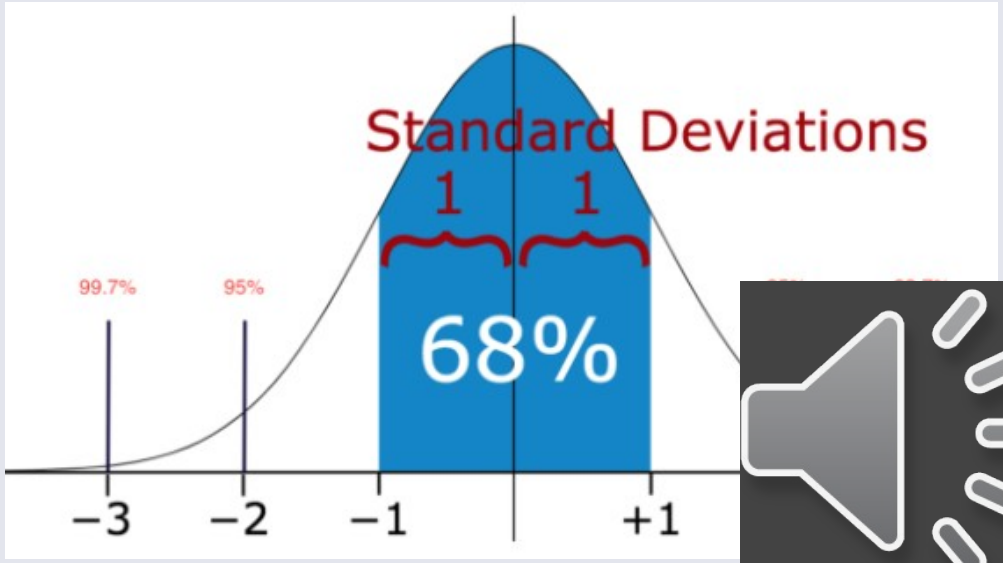
Let's add one more variable. Recalculate:

- Variable (x): 6, 2, 3, 1, 30
- Find the mean
- Square (x-mean)
- Sum results
- Divide by number of data points
- Square root to find SD

$$\mathrm{SD} = \sqrt{\frac{\sum |x - \mu|^2}{N}}$$

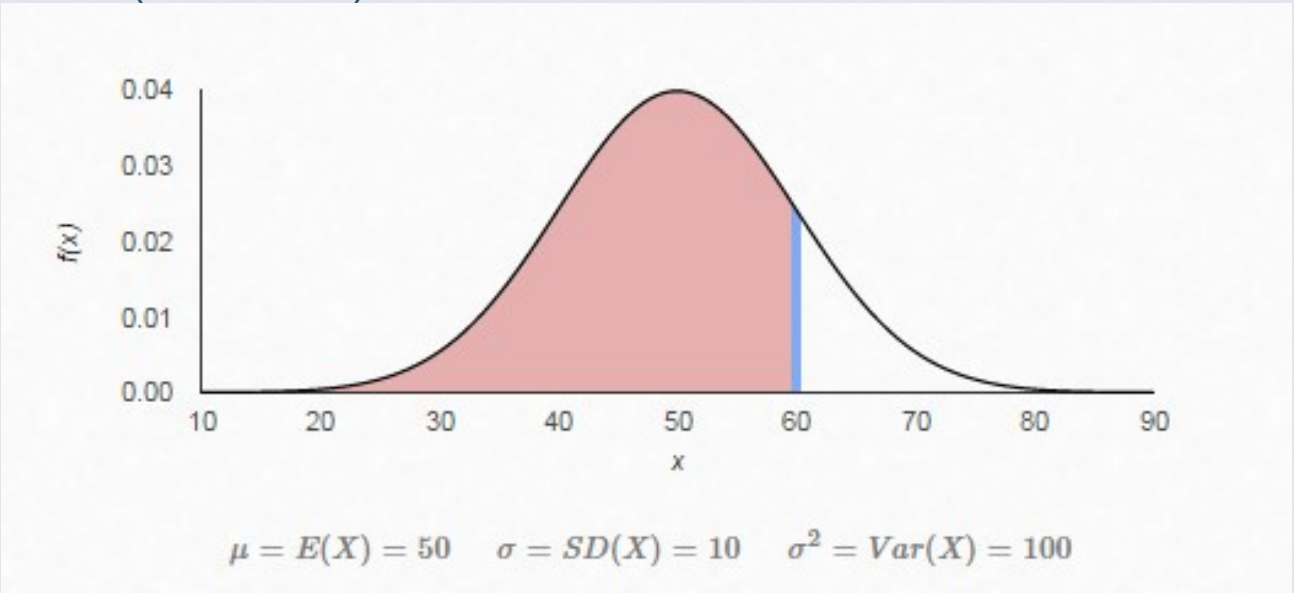
Data Standardisation: Normal Distribution

If a data distribution is approximately normal then about 68% of the data values lie within one standard deviation of the mean and about 95% are within two standard deviations, and about 99.7% lie within three standard deviations.



Standard Deviation: Z-Score (Variance)

- By definition, z-score simply means how many standard deviation a given value is away from the distribution mean. A z-score can be positive or negative.
- We can calculate the z-score as (Score-Mean)/Standard Deviation



Data Normalisation

- Normalisation refers to rescaling real-valued numeric attributes into a 0 to 1 range.

- Data normalisation is used in machine learning/data analytics to make model training less sensitive to the scale of features. This allows our model to converge to better weights and, in turn, leads to a more accurate model.

- Normalisation is generally required when we are dealing with attributes on a different scale, otherwise, it may lead to a dilution in effectiveness of an important equally important attribute (on a lower

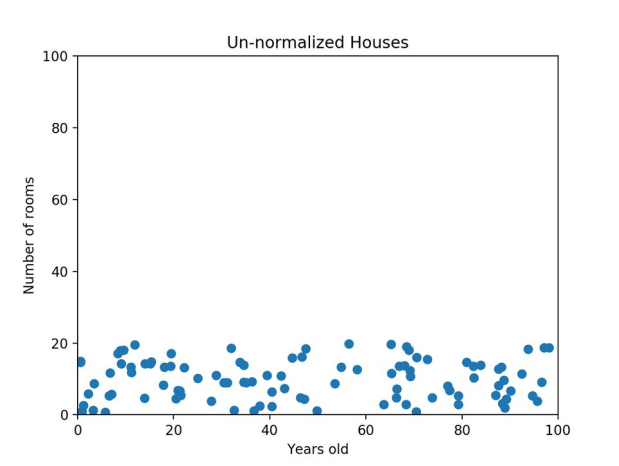
Salary	Year_of_ experience	Expected Position Level
100000	10	2
78000	7	4
32000	5	8
55000	6	7
92000	8	3
120000	15	1
65750	7	5

The attributes salary and year_of_experience are on different scale and hence attribute salary can take high priority over attribute year_of_experience in the model.

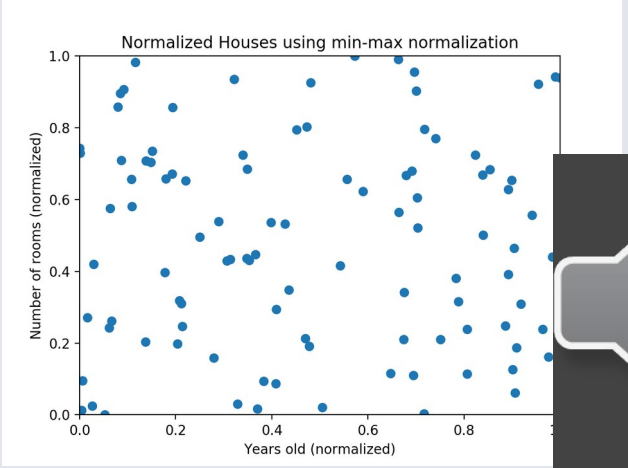
Visualised Example: Normalisation

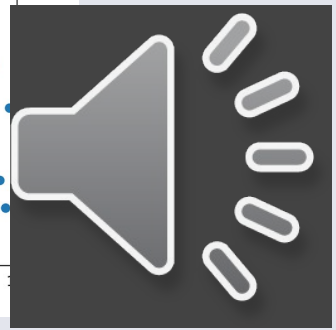
Consider a dataset of houses: Two potential features might be the number of rooms in the house, and the total age of the house in years.

A machine learning algorithm could try to predict which house would be best for you. However, when the algorithm compares data points, the feature with the larger scale will completely dominate the other. Take a look at the image below.



The goal of normalization is to make every datapoint have the same scale so each feature is equally important. The image below shows the same house data normalized using min-max normalisation.





Normalisation Vs Standardisation

University of Lancashire

- Normalisation is good to use when you know that the distribution of your data does not follow a Normal distribution. This can be useful in algorithms that do not assume any distribution of the data.

- Standardisation, on the other hand, can be helpful in cases where the data follows a Normal distribution. Also, unlike normalisation, standardisation does not have a bounding range. So, even if you have outliers in your data, they will not be affected by standardisation.

The choice of using normalisation or standardisation will depend on your problem and the machine learning algorithm you are using. There is no hard and fast rule to tell you when to normalise or standardise your data. You can always start by fitting your model to raw, normalised and standardised data and compare the performance for best results.

Summarising, Aggregating and Grouping Data

University of Lancashire

- Pandas makes the calculation of different statistics very simple, including mean, max, min and standard deviation.

index	date	duration	item	month	network	network_type
0	15/10/14 06:58	34.429	data	2014-11	data	data
1	15/10/14 06:58	13.000	call	2014-11	Vodafone	mobile
2	15/10/14 14:46	23.000	call	2014-11	Meteor	mobile
3	15/10/14 14:48	4.000	call	2014-11	Tesco	mobile
4	15/10/14 17:27	4.000	call	2014-11	Tesco	mobile
5	15/10/14 18:55	4.000	call	2014-11	Tesco	mobile
6	16/10/14 06:58	34.429	data	2014-11	data	data
7	16/10/14 15:01	602.000	call	2014-11	Three	mobile
8	16/10/14 15:12	1050.000	call	2014-11	Three	mobile
9	16/10/14 15:30	19.000	call	2014-11	voicemail	voicemail
10	16/10/14 16:21	1183.000	call	2014-11	Three	mobile
11	16/10/14 22:18	1.000	sms	2014-11	Meteor	mobile

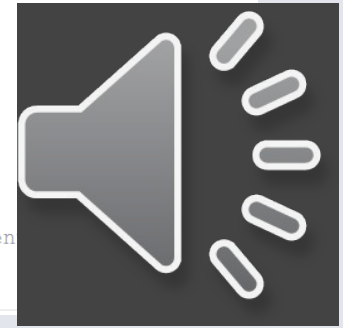
```
# How many rows the dataset
data['item'].count()
Out[38]: 830

# What was the longest phone call / data entry?
data['duration'].max()
Out[39]: 10528.0

# How many seconds of phone calls are recorded in total?
data['duration'][data['item'] == 'call'].sum()
Out[40]: 92321.0

# How many entries are there for each month?
data['month'].value_counts()
Out[41]:
2014-11    230
2015-01    205
2014-12    157
2015-02    137
2015-03    101
dtype: int64

# Number of non-null unique network en
data['network'].nunique()
Out[42]: 9
```



Summarisation: groupby()

- In many cases, apply query-based summarisation:
- What's the total call duration for a particular month?
- This is where groupby() in pandas comes into play.
- groupby() essentially splits the data into different groups depending on a variable of your choice. For example, the expression data.groupby('month') will split the current DataFrame by month.
- The groupby() function returns a groupBy object, but essentially describes how the rows of the original data set has been split.

Note: Functions like max(), min(), mean(), first(), last() can be quickly applied to the GroupBy object to obtain summary statistics for each group

```
# Get the sum of the durations per month
data.groupby('month')['duration'].sum()
Out[70]:
month
2014-11    26639.441
2014-12    14641.870
2015-01    18223.299
2015-02    15522.299
2015-03    22750.441
Name: duration, dtype: float64
```

Further Summarisation and Organisation

University of Lancashire

- You can also group by more than one variable, allowing more complex queries.

```
# How many calls, sms, and data entries are in each month?
data.groupby(['month', 'item'])['date'].count()
Out[76]:
month    item
2014-11  call    107
         data     29
         sms      94
2014-12  call     79
         data     30
         sms      48
2015-01  call     88
         data     31
         sms      86
2015-02  call     67
         data     31
         sms      39
2015-03  call     47
         data     29
         sms      25
Name: date, dtype: int64
```

```
# How many calls, texts, and data are sent per month, split by network_type?
data.groupby(['month', 'network_type'])['date'].count()
Out[82]:
month network_type
2014-11 data 29
 landline 5
 mobile 189
 special 1
 voicemail 6
2014-12 data 30
 landline 7
 mobile 108
 voicemail 8
 world 4
2015-01 data 31
 landline 11
 mobile 160
```

Pandas Statistics

University of Lancashire

- Aggregating statistics: https://pandas.pydata.org/docs/getting_started/intro_tutorials/06_calculate_statistics.html

Learning Objectives

University of Lancashire

Evaluate basic statistics for data analysis