Data Preprocessing
Data Transformation

Data Transformation

- The data are transformed or consolidated into the forms appropriate of data modelling
- Data Transformation involve
  - Smoothing:
    - Used for removing noise or reducing the effect of noice
    - Techniques: Binning, Regression, Clustering
  - Aggregation:
    - Summery or aggregation operation are applied to the data
    - Analysis of data at multiple granularity
    - Example: Daily sales data, Monthly sales data (aggregated on daily data)
  - Attribute construction (feature construction):
    - New attributes are constructed from the raw-data to help mining process
  - Normalization and standardization

Attribute Normalization

- In the context of machine learning, it is termed as feature normalization
- An attribute is normalised by scaling its value so that they fall within a small specified range (for example 0.0 to 1.0)
- Normalization is particularly useful for classification algorithms involving distance measurements and clustering
- For distance based approaches, normalization helps prevent attributes with large ranges from overweighting attributes with smaller ranges

3

Illustration

Price (x_1)	Score for Sale (x_2)
23500.00	8
23500.00	6
22879.00	2
2300.00	4
34678.00	5
15687.00	8
18945.00	6
8750.00	2
37489.00	4
73567.00	2
52789.00	4
2900.00	3
6570.00	3
21000.00	2

min: 2300.00 2

max: 73567.00 8

y_1	y_2
23000.00	6.5

$$\text{Eucledin Distance (ED)} = \sum_{i=1}^{d} (x_i - y_i)^2$$

ED1 = $(23500.00 - 23000.00)^2 + (8 - 6.5)^2$

ED1 = **250002.25**

Illustration

x_1 Price	x_2 Score for Sale
23500.00	8
23500.00	6
22879.00	2
2300.00	4
34678.00	5
15687.00	8
18945.00	6
8750.00	2
37489.00	4
73567.00	2
52789.00	4
2900.00	3
6570.00	3
21000.00	2

min: 2300.00 2

max: 73567.00 8

y_1	y_2
23000.00	6.5

$$\text{Eucledin Distance (ED)} = \sum_{i=1}^{d} (x_i - y_i)^2$$

ED1 = $(23500.00 - 23000.00)^2 + (8 - 6.5)^2$
ED1 = **250002.25**

ED1 = $(23500.00 - 23000.00)^2 + (6 - 6.5)^2$
ED1 = **250000.25**

Illustration

x_1 Price	x_2 Score for Sale
23500.00	8
23500.00	6
22879.00	2
2300.00	4
34678.00	5
15687.00	8
18945.00	6
8750.00	2
37489.00	4
73567.00	2
52789.00	4
2900.00	3
6570.00	3
21000.00	2

min: 2300.00 2

max: 73567.00 8

y_1	y_2
23000.00	6.5

$$\text{Eucledin Distance (ED)} = \sum_{i=1}^{d} (x_i - y_i)^2$$

ED1 = $(23500.00 - 23000.00)^2 + (8 - 6.5)^2$
ED1 = **250002.25**

ED1 = $(23500.00 - 23000.00)^2 + (6 - 6.5)^2$
ED1 = **250000.25**

ED3 = $(22879.00 - 23000.00)^2 + (2 - 6.5)^2$
ED3 = **14661.25**

Attribute Normalization: Min-Max Normalization

- It performs a linear transformation on the original data
- The transformed data is the scaled version of the original data so that they fall within a small specified range
- Each numeric attributes in a data are normalised separately
- Steps:
  - Compute minimum (mn_A) and maximum (mx_A) values of an attribute A
  - Specify the new minimum (new_mn_A) and new maximum range (new_mx_A)
  - Min-Max normalization maps a value, x of attribute A to $\hat{x}$ in the specified range by computing

$$\hat{x} = \frac{x - mn_A}{mx_A - mn_A}(new\_mx_A - new\_mn_A) + new\_mn_A$$

7

Attribute Normalization: Min-Max Normalization

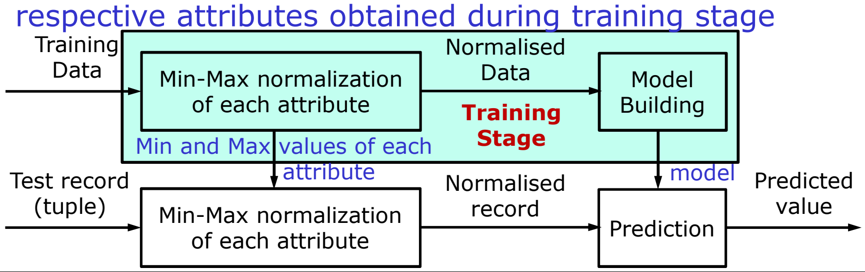
- When new minimum (new_mn_A) and new maximum range (new_mx_A) is 0 and 1 respectively, then the data is scaled to 0.0 to 1.0 range
  - Min-Max normalization maps a value, x of attribute A to $\hat{x}$ in the 0.0 to 1.0 range by computing

$$\hat{x} = \frac{x - mn_A}{mx_A - mn_A}$$

8

Min-Max Normalization during Model Building

- Model building and prediction using machine learning involve two stages:
  - Training stage: Model building
  - Test stage: Prediction using the built model
- Training stage: Normalise each attribute using Min-Max normalization by using the minimum and maximum values from respective attributes
- Test stage: Normalise each test records (samples) using the minimum and maximum values from respective attributes obtained during training stage



Attribute Normalization: Min-Max Normalization

- Min-Max normalization preserves the relationship among the original data values
- It is useful when data has varying ranges among attributes
- It is useful when machine learning (ML) algorithms we are using does not make any assumption about distribution of data
- It is useful when the actual minimum and maximum values for the attribute is known
- Disadvantage: “out-of-bound” error if a future input case for normalization falls outside the original range of attribute `A`
  - This situation arises when the actual minimum and maximum of attribute `A` is unknown

10

Illustration of Min-Max Normalization

1	Temperature	Humidity	Rain
2	25.46875	82.1875	6.75
3	26.19298	83.14912	1762
4	25.17021	85.34043	653
5	24.29851	87.68657	963
6	24.06923	87.64615	254
7	21.20779	95.94805	340
8	23.48571	96.17143	38.3
9	21.79487	98.58974	29.3
10	25.09346	88.3271	4.5
11	25.39423	90.43269	113
12	23.89076	94.53782	736
13	22.5098	99	608
14	22.904	98	718
15	21.72464	99	513

min: 21.20779 82.187 4.5

max: 26.19298 99 1762

Temperature	Humidity	Rain
0.85472	0.00000	0.00128
1.00000	0.05720	1.00000
0.79484	0.18753	0.36876
0.61998	0.32708	0.54545
0.57399	0.32468	0.14213
0.00000	0.81847	0.19078
0.45694	0.83176	0.01921
0.11776	0.97560	0.01408
0.77944	0.36518	0.00000
0.83978	0.49042	0.06146
0.53819	0.73459	0.41613
0.26118	1.00000	0.34315
0.34025	0.94052	0.40589
0.10368	1.00000	0.28937

0.000 0.000 0.000

1.000 1.000 1.000

Illustration of Min-Max Normalization

Price	Score for Sale
23500.00	8
23500.00	6
22879.00	2
2300.00	4
34678.00	5
15687.00	8
18945.00	6
8750.00	2
37489.00	4
73567.00	2
52789.00	4
2900.00	3
6570.00	3
21000.00	2

min: 2300.00 2

max: 73567.00 8

Price	Credit for Sale
0.2975	1.0000
0.2975	0.6667
0.2888	0.0000
0.0000	0.3333
0.4543	0.5000
0.1878	1.0000
0.2336	0.6667
0.0905	0.0000
0.4938	0.3333
1.0000	0.0000
0.7084	0.3333
0.0084	0.1667
0.0599	0.1667
0.2624	0.0000

0.000 0.000

1.000 1.000

Illustration of Min-Max Normalization

Price	Score for Sale
23500.00	8
23500.00	6
22879.00	2
2300.00	4
34678.00	5
15687.00	8
18945.00	6
8750.00	2
37489.00	4
73567.00	2
52789.00	4
2900.00	3
6570.00	3
21000.00	2

min: 2300.00 2

max: 73567.00 8

23000.00	6.5

↓

0.2905	0.75

Illustration

x_1 Price	x_2 Credit for Sale
0.2975	1.0000
0.2975	0.6667
0.2888	0.0000
0.0000	0.3333
0.4543	0.5000
0.1878	1.0000
0.2336	0.6667
0.0905	0.0000
0.4938	0.3333
1.0000	0.0000
0.7084	0.3333
0.0084	0.1667
0.0599	0.1667
0.2624	0.0000

min: 0.00 0.00

max: 1.00 1.00

y_1	y_2
0.2905	0.75

$$\text{Eucledin Distance (ED)} = \sum_{i=1}^{d} (x_i - y_i)^2$$

ED1 = $(0.2975 - 0.2905)^2 + (1 - 0.75)^2$
ED1 = **0.06255**

Illustration

x_1	x_2
Price	**Credit for Sale**
0.2975	1.0000
0.2975	0.6667
0.2888	0.0000
0.0000	0.3333
0.4543	0.5000
0.1878	1.0000
0.2336	0.6667
0.0905	0.0000
0.4938	0.3333
1.0000	0.0000
0.7084	0.3333
0.0084	0.1667
0.0599	0.1667
0.2624	0.0000

min: 0.00 0.00

max: 1.00 1.00

y_1	y_2
0.2905	0.75

$$\text{Eucledin Distance (ED)} = \sum_{i=1}^{d}(x_i - y_i)^2$$

ED1 = $(0.2975 - 0.2905)^2 + (1 - 0.75)^2$
ED1 = **0.06255**

ED2 = $(0.2975 - 0.2905)^2 + (0.6667 - 0.75)^2$
ED2 = **0.00699**

Illustration

x_1	x_2
Price	**Credit for Sale**
0.2975	1.0000
0.2975	0.6667
0.2888	0.0000
0.0000	0.3333
0.4543	0.5000
0.1878	1.0000
0.2336	0.6667
0.0905	0.0000
0.4938	0.3333
1.0000	0.0000
0.7084	0.3333
0.0084	0.1667
0.0599	0.1667
0.2624	0.0000

min: 0.00 0.00

max: 1.00 1.00

y_1	y_2
0.2905	0.75

$$\text{Eucledin Distance (ED)} = \sum_{i=1}^{d}(x_i - y_i)^2$$

ED1 = $(0.2975 - 0.2905)^2 + (1.0 - 0.75)^2$
ED1 = **0.06255**

ED2 = $(0.2975 - 0.2905)^2 + (0.6667 - 0.75)^2$
ED2 = **0.00699**

ED3 = $(0.2888 - 0.2905)^2 + (0.0 - 0.75)^2$
ED2 = **0.56250**

Data Standardization (z-score Normalization)

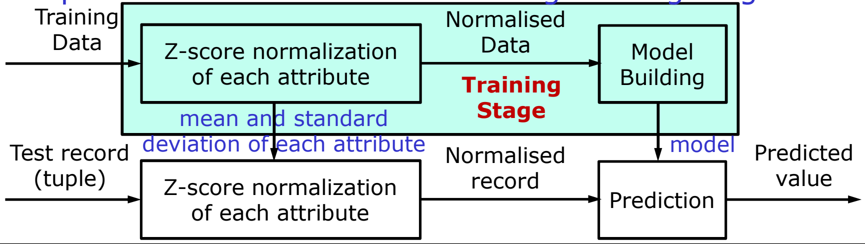
- The process of rescaling one or more attributes so that the transformed data have 0 mean and unit variance i.e. standard deviation of 1
- Standardization assumes that data has a Guassian distribution
  – This assumption does not strictly have to be true, but this technique is more effective if your attribute distribution is Gaussian
- In this process, values of an attribute, A, are normalised based on the mean and standard deviation of A
- A value, x, of attribute A is normalised to $\hat{x}$ by computing

$$\hat{x} = \frac{x - \mu_A}{\sigma_A}$$

- μ_A: mean of attribute A
- σ_A: standard deviation of attribute A

z-score Normalization during Model Building

- Model building and prediction using machine learning involve two stages:
  – Training stage: Model building
  – Test stage: Prediction using the built model
- Training stage: Normalise each attribute using z-score normalization by using the mean and standard deviation from respective attributes
- Test stage: Normalise each test records (samples) using the mean and standard deviation from respective attributes obtained during training stage



Data Standardization (z-score Normalization)

- This method of normalization is useful
  - when the actual minimum and maximum of attribute are unknown
  - when there are outliers that dominates the Min-Max normalization
  - when data has Gaussian distribution (symmetric distribution)
- This method of normalization is useful when the ML algorithms make any assumptions of Gaussian distribution

Illustration of Data Standardization (z-score Normalization)

1	Temperature	Humidity	Rain
2	25.46875	82.1875	6.75
3	26.19298	83.14912	1762
4	25.17021	85.34043	653
5	24.29851	87.68657	963
6	24.06923	87.64615	254
7	21.20779	95.94805	340
8	23.48571	96.17143	38.3
9	21.79487	98.58974	29.3
10	25.09346	88.3271	4.5
11	25.39423	90.43269	113
12	23.89076	94.53782	736
13	22.5098	99	608
14	22.904	98	718
15	21.72464	99	513
μ:	23.80035	91.86	481
σ:	1.58225	6.13	488

⇒

Temperature	Humidity	Rain
1.05444	-1.57673	-0.97166
1.51216	-1.41995	2.62269
0.86576	-1.06268	0.35088
0.31484	-0.68016	0.98680
0.16993	-0.68675	-0.46476
-1.63853	0.66679	-0.28965
-0.19886	0.70321	-0.90714
-1.26749	1.09749	-0.92558
0.81726	-0.57573	-0.97627
1.00735	-0.23244	-0.75508
0.05714	0.43686	0.52138
-0.81564	1.16438	0.25871
-0.56650	1.00134	0.48451
-1.31187	1.16438	0.06517
0.000	0.000	0.000
1	1	1

Summery on Data Transformation

- Data transformation is useful of data modelling
- Normalization:
  - Each attribute is normalised by scaling its value so that they fall within a small specified range (for example 0.0 to 1.0)
  - Min-Max normalization
    - It is useful when data has varying ranges among attributes
- Standarization (z-score normalization):
  - The process of rescaling one or more attributes so that the transformed data have 0 mean and unit variance i.e. standard deviation of 1
  - Standardization assumes that data has a Gaussian distribution
  - It is useful when the actual minimum and maximum of attribute are unknown

21