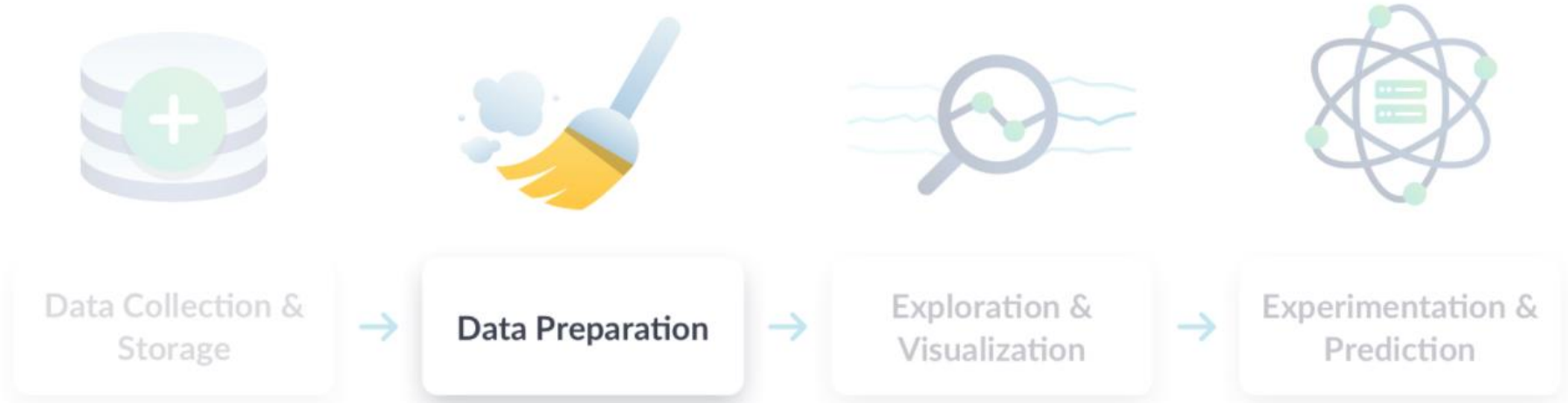
Data Wrangling/Preprocessing

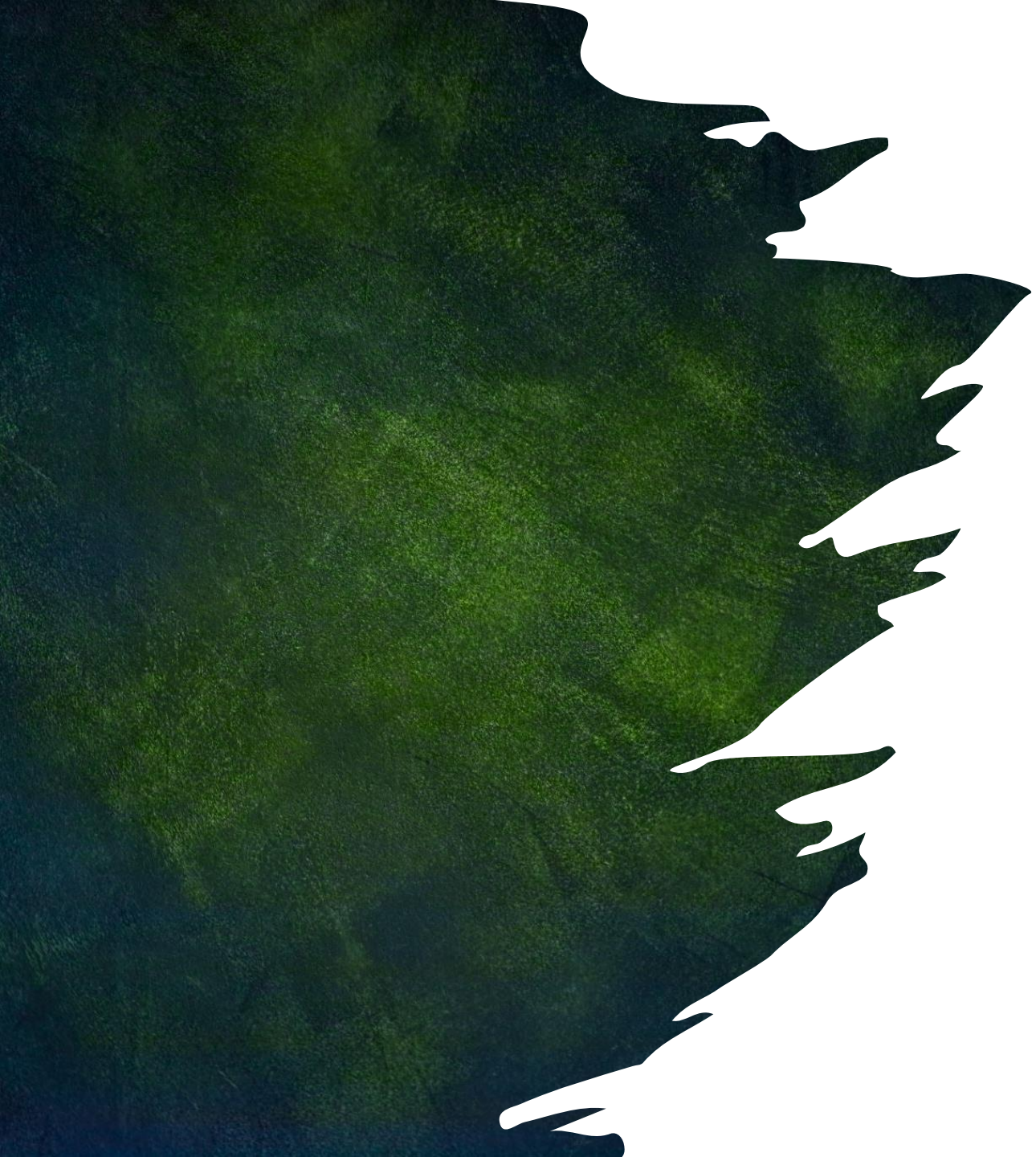
GGE 6505/GGE5405 Introduction to Big Data & Data Science

Data Science Workflow



Outline

- Does the data already exist that can solve my problem?
- The data you have: is it enough?
- Be able to identify the potential challenges that you can face when wrangling big data.
- Apply different techniques to make data ready

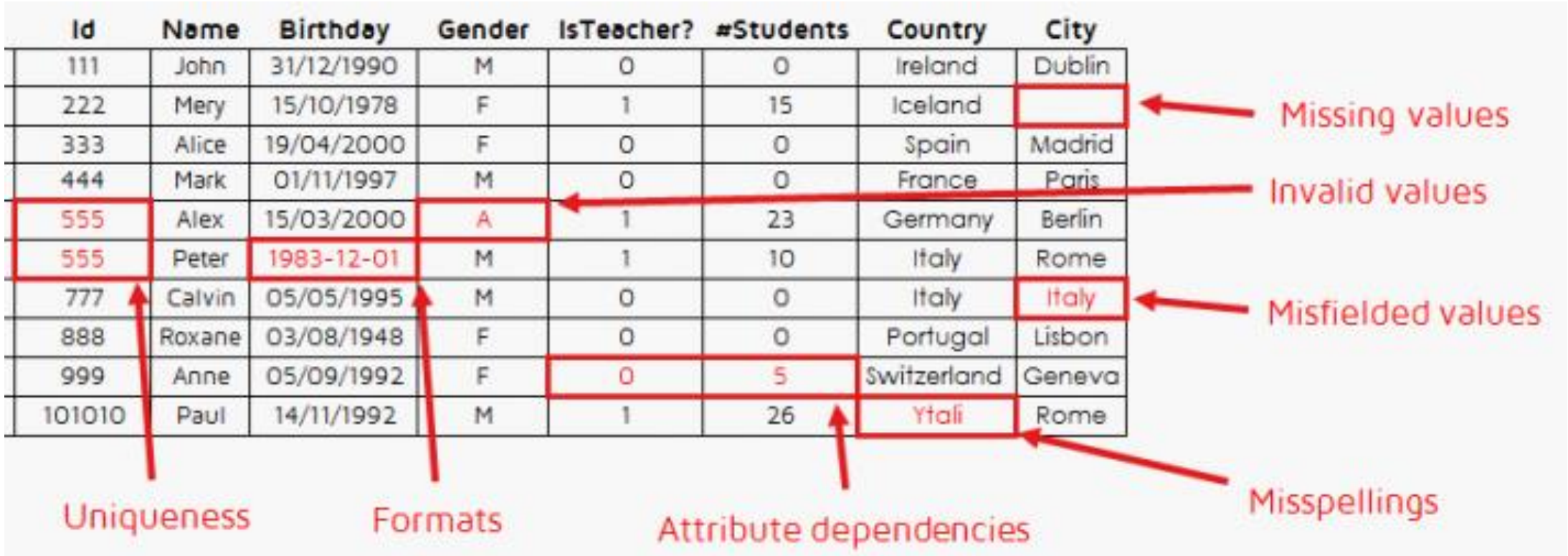


It is known as

- Data Wangling
- Data Preprocessing
- Data Preparation
- Data Cleansing
- Data Scrubbing
- Data Munging
- Data Transformation
- Data Fold, Spindle, Mutilate…

What is data wrangling?

- It is a process of transforming “raw” data into actionable data that can be analyzed to generate valid meaningful insights.
- Wrangling isn't a task with steps that can be prescribed exactly before hand. Every case is different and takes CREATIVITY to get good results.
- Good wrangling comes down to solid planning before wrangling and then some guessing and checking to see that works.
- Spending extra time on data wrangling can save you a lot of pain later.

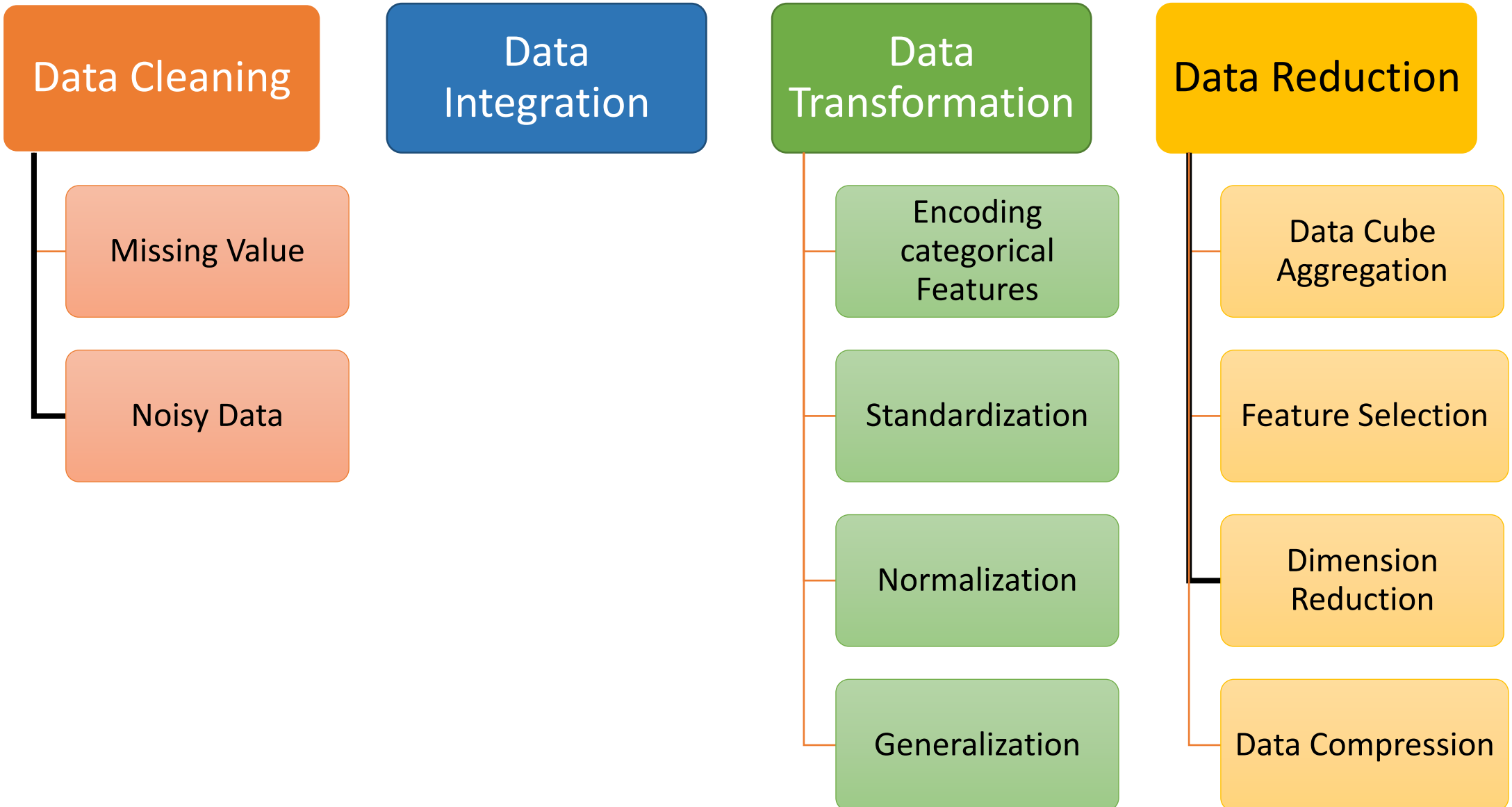


Data cleansing & data transformation | Quantdare

color	director_name	duration	gross	movie_title	anguag	country	budget	title_year	imdb_score
Color	Martin Scorsese	240	116866727	The Wolf of Wall StreetÂ	English	USA	100000000	2013	8.2
Color	Shane Black	195	408992272	Iron Man 3Â	English	USA	200000000	2013	7.2
color	Quentin Tarantino	187	54116191	The Hateful EightÂ	English	USA	44000000	2015	7.9
Color	Kenneth Lonergan	186	46495	MargaretÂ	English	usa	14000000	2011	6.5
Color	Peter Jackson	186	258355354	The Hobbit: The Desolation of SmaugÂ	English	USA	225000000	2013	7.9
	N/A	183	330249062	Batman v Superman: Dawn of JusticeÂ	English	USA	250000000	202	6.9
Color	Peter Jackson	-50	303001229	The Hobbit: An Unexpected JourneyÂ	English	USA	180000000	2012	7.9
Color	Edward Hall	180		RestlessÂ	English	UK		2012	7.2
Color	Joss Whedon	173	623279547	The AvengersÂ	English	USA	220000000	2012	8.1
Color	Joss Whedon	173	623279547	The AvengersÂ	English	USA	220000000	2012	8.1
	Tom Tykwer	172	27098580	Cloud AtlasA	English	Germany	102000000	2012	-7.5
Color	Null	158	102515793	The Girl with the Dragon TattooÂ	English	USA	90000000	2011	7.8
Color	Christopher Spencer	170	59696176	Son of GodÂ	English	USA	22000000	2014	5.6
Color	Peter Jackson	164	255108370	The Hobbit: The Battle of the Five ArmiesÂ	English	New Zealand	250000000	2014	7.5
Color	Tom Hooper	158	148775460	Les MisÃ©rablesÂ	English	USA	61000000	2012	7.6
Color	Tom Hooper	158	148775460	Les MisÃ©rablesÂ	English	USA	61000000	2012	7.6

Cleaning a messy dataset using Python | by Reza Rajabi | Well Red | Medium

Data preprocessing Techniques



Data Cleaning

- Data cleaning is the process of fixing or removing incorrect, corrupted, incorrectly formatted, duplicate, or incomplete data within a dataset

- **Missing Values**
  - Ignore Tuple
  - Fill the Gap

- **Noisy Data**
  - Binning
  - Clustering
  - ML Algorithms

1- Missing Value

- Ignore the data: usually done when class label is missing (assuming the tasks in classification—not effective when the percentage of missing values per attribute varies considerably).
- Fill in the missing value manually: tedious + infeasible?
- Fill in it automatically with
  - Categorical variable: a global constant : e.g., “NaN”, a new class?!
  - Numerical variable: the mean, nearest neighbor, interpolation

	summary	city	state	date_time	shape	duration	stats	report_link	text	posted	city_latitude
0	My wife was driving southeast on a fairly popu...	Chester	VA	2019-12-12T18:43:00	light	5 seconds	Occurred : 12/12/2019 18:43 (Entered as : 12/...	http://www.nuforc.org/webreports/151/S151739.html	My wife was driving southeast on a fairly popu...	2019-12-22T00:00:00	37.343152
1	I think that I may caught a UFO on the NBC Nig...	Rocky Hill	CT	2019-03-22T18:30:00	circle	3-5 seconds	Occurred : 3/22/2019 18:30 (Entered as : 03/2...	http://www.nuforc.org/webreports/145/S145297.html	I think that I may caught a UFO on the NBC Nig...	2019-03-29T00:00:00	41.664800
2	I woke up late in the afternoon 3:30-4pm. I we...	NaN	NaN	NaN	NaN	NaN	Occurred : 4/1/2019 15:45 (Entered as : April...	http://www.nuforc.org/webreports/145/S145556.html	I woke up late in the afternoon 3:30-4pm. I w...	NaN	NaN
3	I was driving towards the intersection of fall...	Ottawa	ON	2019-04-17T02:00:00	teardrop	10 seconds	Occurred : 4/17/2019 02:00 (Entered as : 04-1...	http://www.nuforc.org/webreports/145/S145697.html	I was driving towards the intersection of fall...	2019-04-18T00:00:00	45.381383
4	In Peoria Arizona, I saw a cigar shaped craft ...	Peoria	NY	2009-03-15T18:00:00	cigar	2 minutes	Occurred : 3/15/2009 18:00 (Entered as : 03/1...	http://www.nuforc.org/webreports/145/S145723.html	In Peoria, Arizona, I saw a cigar shaped craft...	2019-04-18T00:00:00	NaN

Handling Missing Value

mean/median

Pros:

- Easy and fast.
- Works well with small numerical datasets.

Cons:

- Doesn't factor the correlations between features. It only works on the column level.
- Will give poor results on encoded categorical features (do NOT use it on categorical features).
- Not very accurate.
- Doesn't account for the uncertainty in the imputations.

ID	Name	Age	Size	Country
0	Sara	27	1.77	"BE"
1	Lis	30	1.70	"USA"
2	Hadrien		1.80	"FR"

Mean()

ID	Name	Age	Size	Country
0	Sara	27	1.77	"BE"
1	Lis	30	1.70	"US"
2	Hadrien	28	1.80	"FR"

Handling Missing Value

(Zero/Constant) Value

Pros:

- Works well with categorical features

Cons:

- It also doesn't factor the correlations between features.
- It can introduce bias in the data.

ID	Name	Age	Size	Country
0	Sara	27	1.77	"BE"
1	Lis	30	1.70	"USA"
2	Hadrien		1.80	"FR"

fillna(0)

ID	Name	Age	Size	Country
0	Sara	27	1.77	"BE"
1	Lis	30	1.70	"US"
2	Hadrien	0	1.80	"FR"

Handling Missing Value

k-NN

Pros:

- Can be much more accurate than the mean, median or most frequent imputation methods (It depends on the dataset).

Cons:

- Computationally expensive. KNN works by storing the whole training dataset in memory.
- K-NN is quite sensitive to outliers in the data (unlike SVM)

ID	Name	Age	Size	Country
0	Sara	27	1.77	"BE"
1	Lis	30	1.70	"USA"
2	Hadrien		1.80	"FR"

ID	Name	Age	Size	Country
0	Sara	27	1.77	"BE"
1	Lis	30	1.70	"US"
2	Hadrien	0	1.80	"FR"

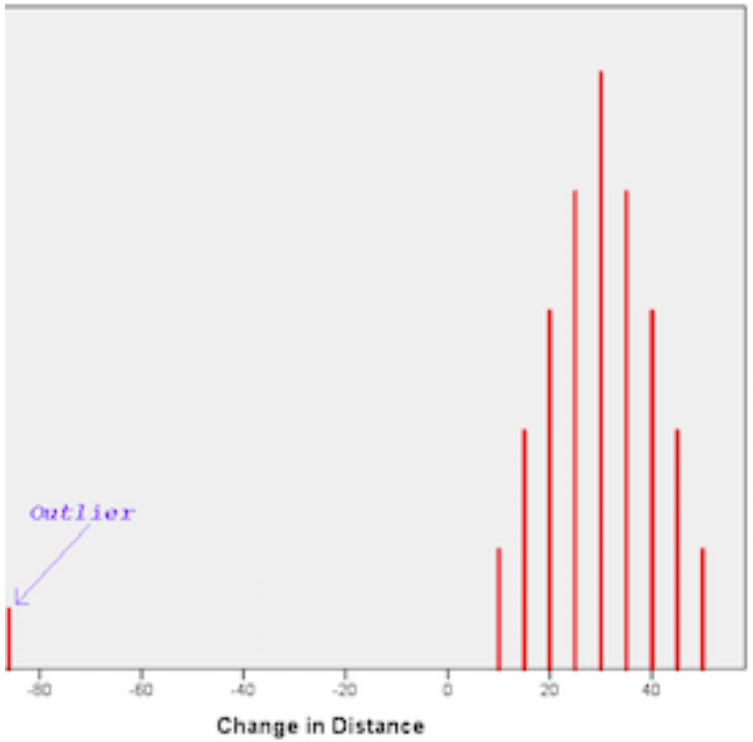
Imputation of missing values with Python

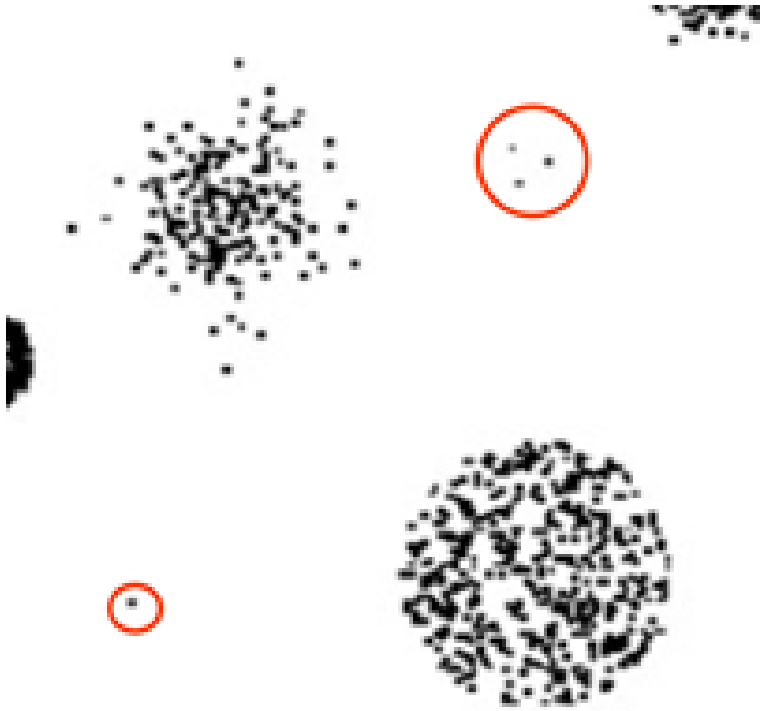
- 6.4. Imputation of missing values — scikit-learn 1.0.2 documentation

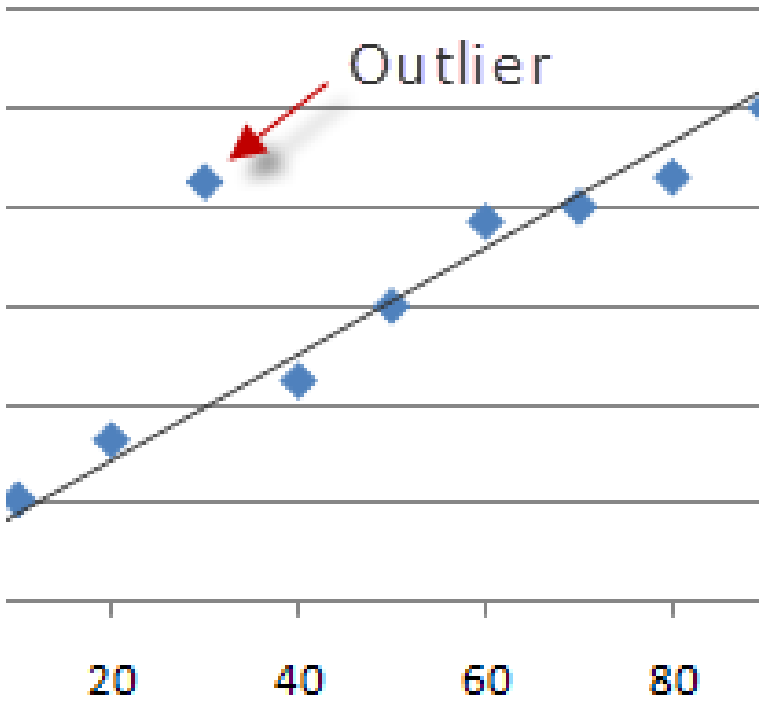
2- Noisy Data

- Noisy data is meaningless data. The term has often been used as a synonym for corrupt data.
- Any data that has been received, stored, or changed in such a manner that it cannot be read or used by the program that originally created it can be described as noisy.
- Noisy attribute values may due to
    - faulty data collection instruments
    - data entry problems
    - data transmission problems
    - technology limitation
    - inconsistency in naming convention

Anomaly or Outlier







2- Noisy Data

- Unnecessarily increases the amount of storage space required
- Adversely affect the results of any data mining and machine learning

Student ID	Student Name	Age	GPA
100122014	Joseph	21	3.5
100232015	Patrick	200	3.2
100122012	Seller	24	3.0
100342013	Roger	23	234
100942012	Davis	2.8	3.7
	Travis	23	3.4
100982015	Alex	27	
100982013	Trevor	-22	4.0

Handling Noisy Data

Binning

The sorted data values are put into the number of buckets and considering the neighboring values in each bin, the local smoothing is performed.

- first sort data and partition into (equal-frequency) bins
- then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.

Binning Method Example

1. Sort raw data from lowest price to highest price of a product

   4,8,9,15,21,21,24,25,26,28,29,34

2. Partition into bins

   Bin 1: 4,8,9,15

   Bin 2: 21,21,24,25

   Bin 3 : 26,28,29,34

3. Smoothing by bin means

   Bin 1: 9,9,9,9

   Bin 2: 23,23,23,23

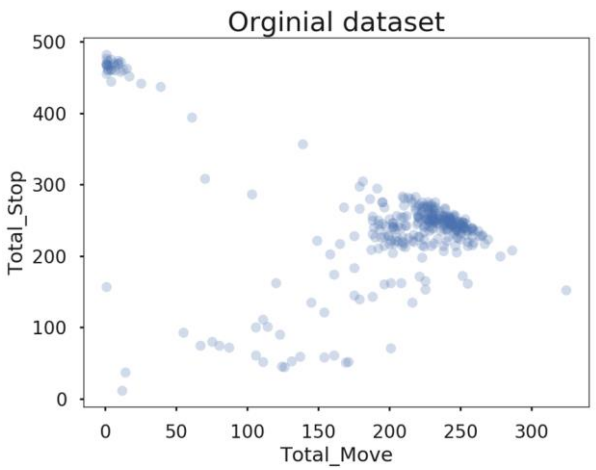
   Bin 3 : 29,29,29,29

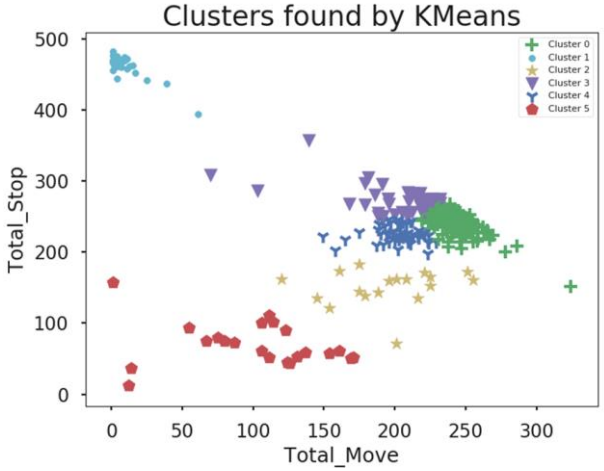
Handling Noisy Data

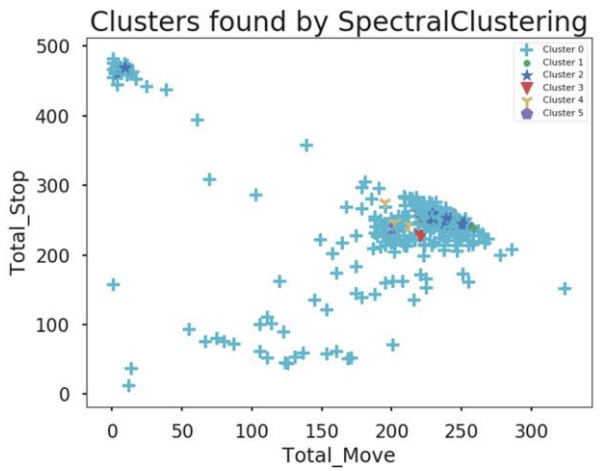
Clustering

A data point is an outlier if it does not belong to a strong cluster

- Computation of clusters using:







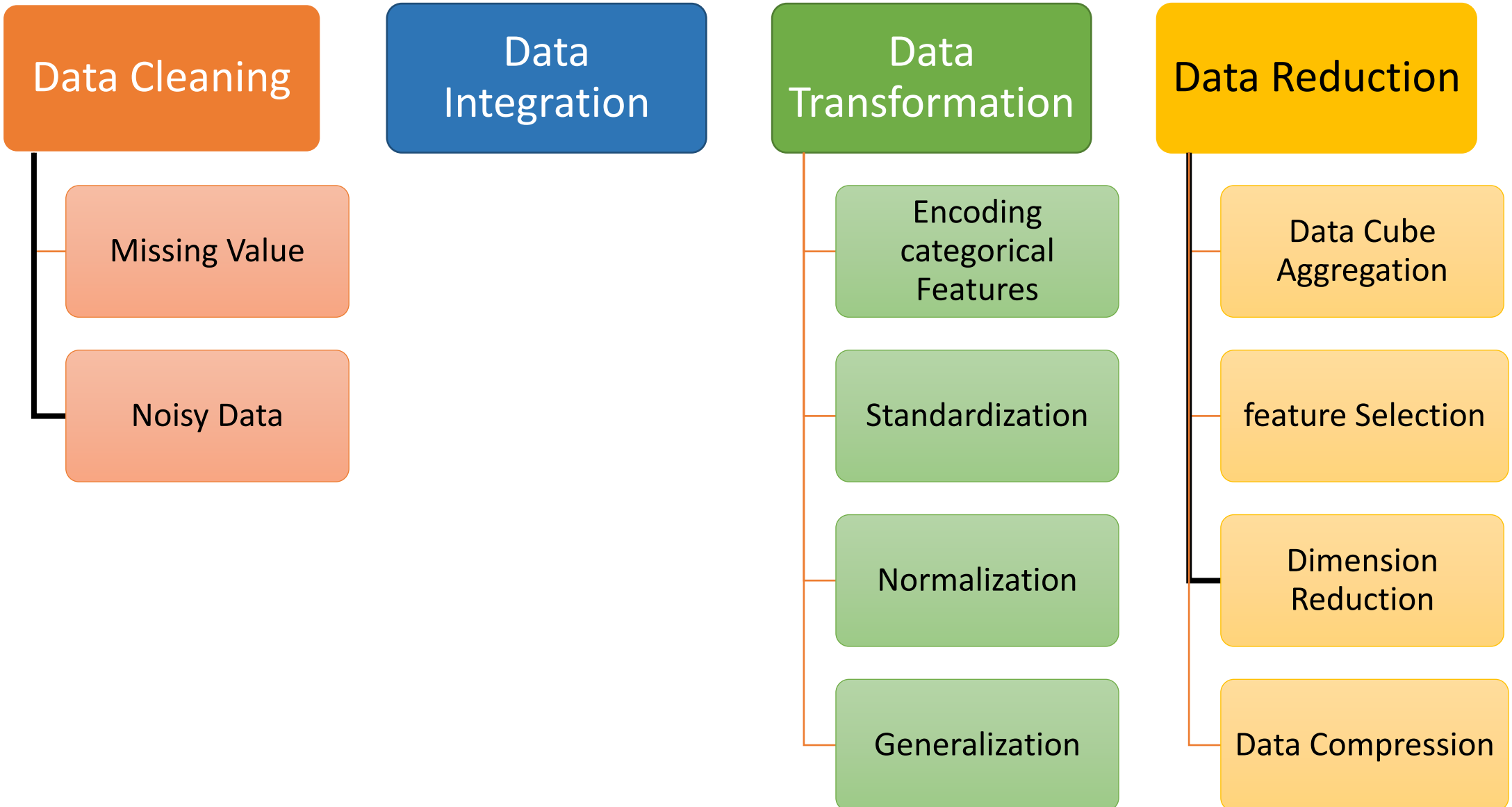
- Anomaly Functions:
  - f (x) = relative distance (median distance to all data points to the cluster centroid)

Handling Noisy Data

Regression

- Regression is used to smooth noisy data. Regression matches the data values to a function such as linear regression which identifies the relationship between two variables so that, one attribute helps in identifying the value of another attribute.
- Replace observed values by predicted values
- Look at: 10 Assumptions of Linear Regression - Full List with Examples and Code (r-statistics.co)

Data preprocessing Techniques



Data Integration

- Combine data from multiple sources into a coherent database
- Attribute identification problem: “same” attributes from multiple data sources may have different names Instance integration
- Detect duplicate records from different sources involves approximate matching of attribute values e.g. 3.14283 ≡ 3.1, Schwartz ≡ Schwarz

Treatment

- Merge corresponding tables
- Use attribute values as synonyms
- Remove duplicate records data warehouses are already integrated

- Merge in Python
- df = pd.merge(df1, df2, on = 'Student_id')

	Student_id	Mark	City
0	1	95	Chennai
1	2	70	Delhi
2	3	98	Mumbai
3	4	75	Pune
4	5	89	Kochi

	Student_id	Age	Gender	Grade	Employed
0	1	19	Male	1st Class	yes
1	2	20	Female	2nd Class	no
2	3	18	Male	1st Class	no
3	4	21	Female	2nd Class	no
4	5	19	Male	1st Class	no

	Student_id	Mark	City	Age	Gender	Grade	Employed
0	1	95	Chennai	19	Male	1st Class	yes
1	2	70	Delhi	20	Female	2nd Class	no
2	3	98	Mumbai	18	Male	1st Class	no
3	4	75	Pune	21	Female	2nd Class	no
4	5	89	Kochi	19	Male	1st Class	no

Redundant in Data Integration

What is Data Redundancy ?

During data integration in data mining, various data stores are used. This can lead to the problem of redundancy in data. An attribute (column or feature of data set) is called redundant if it can be derived from any other attribute or set of attributes. Inconsistencies in attribute or dimension naming can also lead to the redundancies in data set.

Example –

We have a data set having three attributes:

person_name, is_male, is_female.

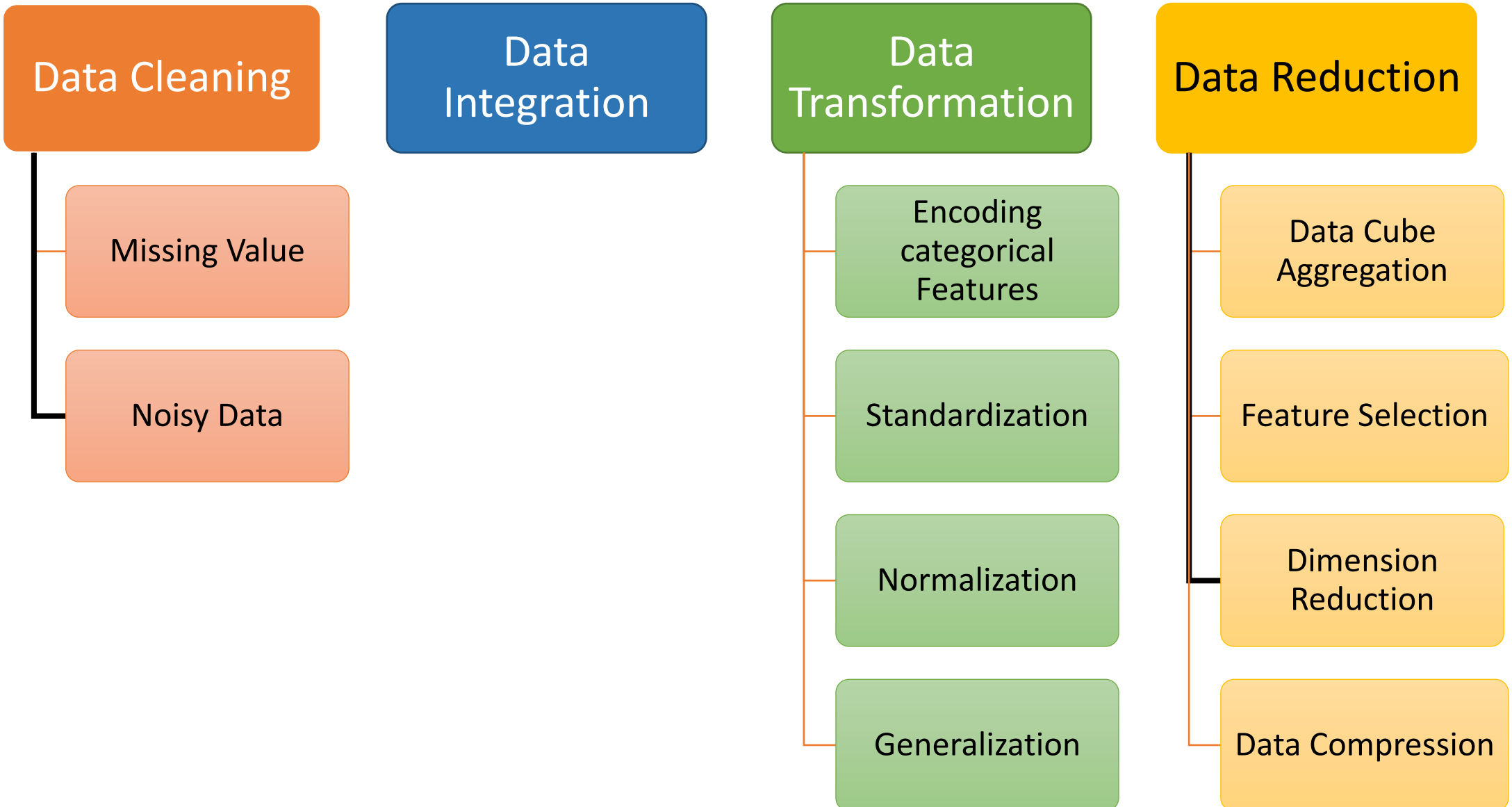
- is_male is 1 if the corresponding person is a male else it is 0 .
- is_female is 1 if the corresponding person is a female else it is 0

person_name	is_male	is_female
Aman	1	0
Abhinav	1	0
Ashutosh	1	0
Dishi	0	1
Abhishek	1	0
Avantika	1	0
Ayushi	0	1

HIGHLY CORRELATED ATTRIBUTES

One attribute can be removed without any information loss. As one attribue can easily determine the other.

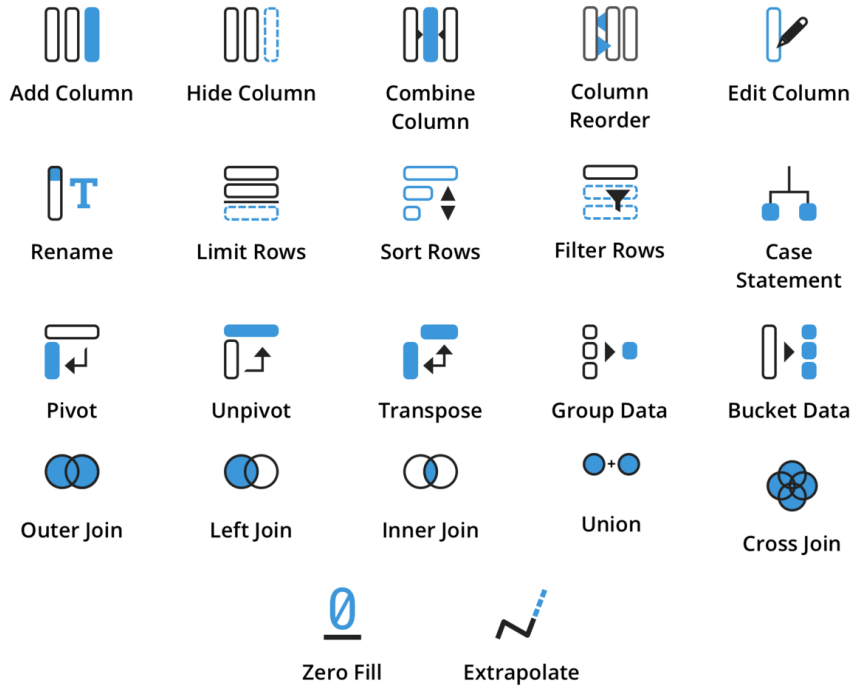
Data preprocessing Techniques



Data Transformation

Data transformation is a technique of conversion as well as mapping of data from one format to another. The tools and techniques used for data transformation depend on the format, complexity, structure and volume of the data.

- ❖Encoding Categorical Features
- ❖Normalization
- ❖Standardization
- ❖Generalization



Data Transformation Icons by Steven Lewis on Dribbble

1- Encoding categorical Features

To address the problems associated with categorical data, we can use encoding. This is the process by which we convert a categorical variable into a numerical form. Here, we will look at three simple methods of encoding categorical data.

- **Replacing**
- **Label Encoding**
- **One-Hot Encoding**

Replacing

- We encode categorical data numerically because math is generally done using numbers. Just like that, our algorithms cannot run and process data if that data is not numerical. Therefore, data scientists need to have tools at their disposal to transform colors like red, yellow, and blue into numbers like 1, 2, and 3 for all the backend math to take place.

	Gender	Grade	Employed
0	Male	1st Class	yes
1	Female	2nd Class	no
2	Male	1st Class	no
3	Male	2nd Class	no
4	Female	3rd Class	no
5	Female	2nd Class	yes

	Gender	Grade	Employed
0	0	1	1
1	1	2	0
2	0	1	0
3	0	2	0
4	1	3	0
5	1	2	1

Label Encoding

- This is a technique in which we replace each value in a categorical column with numbers from 0 to N-1

```
>>> from sklearn.preprocessing import LabelEncoder
>>> label_encoder=LabelEncoder()
>>> input_classes=['Havells','Philips','Syska','Eveready','Lloyd']
>>> label_encoder.fit(input_classes)
>>> for i,item in enumerate(label_encoder.classes_):
        print(item,'-->',i)
```

SAFETY-LEVEL (TEXT)	SAFETY-LEVEL (NUMERICAL)
None	0
Low	1
Medium	2
High	3
Very-High	4

One-hot encoding

id	color
1	red
2	blue
3	green
4	blue

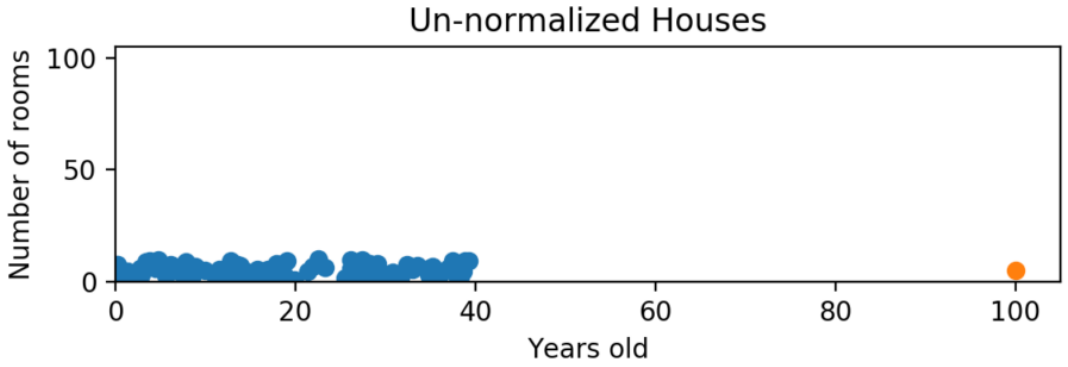
id	color_red	color_blue	color_green
1	1	0	0
2	0	1	0
3	0	0	1
4	0	1	0

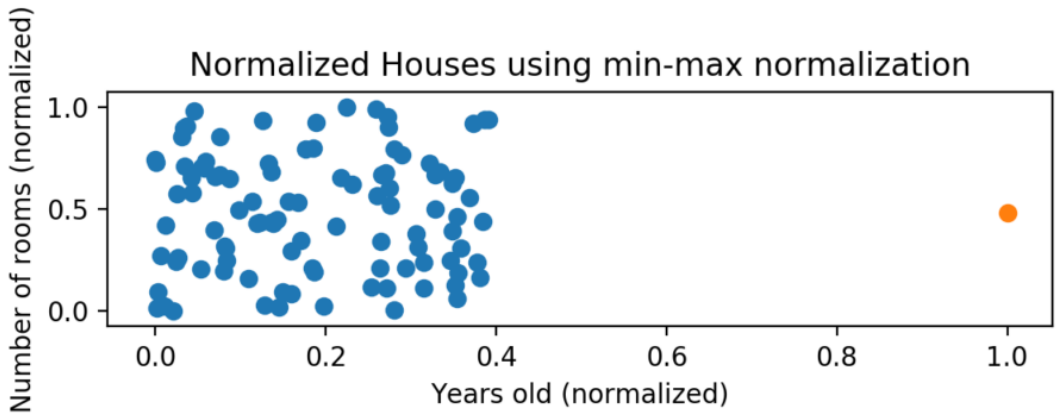
- Label Encoder vs. One Hot Encoder in Machine Learning | by Sunny Srinidhi | Medium

2- Normalization

The numerical attributes are scaled up or down to fit within a specified range. In this approach, we are constraining our data attribute to a particular container to develop a correlation among different data points. Normalization can be done in multiple ways, which are highlighted here:

- Min-max normalization
- Z-Score normalization
- Decimal scaling normalization





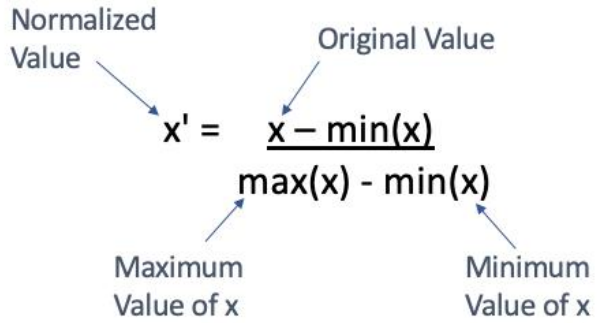
person_name	Salary	Year_of_ experience	Expected Position Level
Aman	100000	10	2
Abhinav	78000	7	4
Ashutosh	32000	5	8
Dishi	55000	6	7
Abhishek	92000	8	3
Avantika	120000	15	1
Ayushi	65750	7	5

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.

Normalization Formulas

- Min-max normalization

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$



- z-score normalization

$$v' = \frac{v - \mu_A}{\sigma_A}$$

μ = Mean

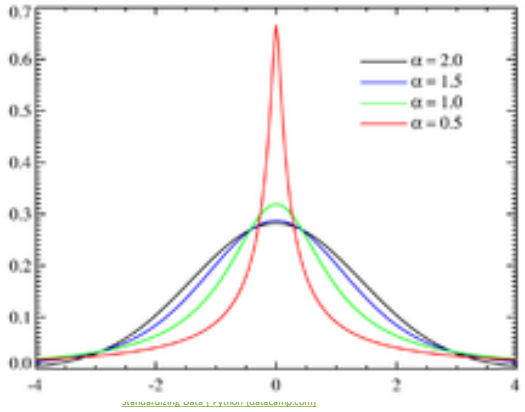
σ = Standard Deviation

- Normalization by decimal scaling

$$v' = \frac{v}{10^j}$$

where j is the smallest integer such that $Max(|v'|) < 1$

3- Standardization



- Data standardization helps improve the quality of your data by transforming and standardizing it.
- Standardization is used on the data values that are normally distributed. Further, by applying standardization, we tend to make the mean of the dataset as 0 and the standard deviation equivalent to 1.
- In practice we often ignore the shape of the distribution and just transform the data to center it by removing the mean value of each feature, then scale it by dividing non-constant features by their standard deviation.
- Also, Standardization is the process of bringing data into a uniform format

Standardization techniques

- **0-1 scaling**: each variable in the data set is recalculated as (*V* - min *V*)/(max V - min *V*), where V represents the value of the variable in the original data set. This method allows variables to have differing means and standard deviations but equal ranges. In this case, there is at least one observed value at the 0 and 1 endpoints.

- Dividing each value by the **range**: recalculates each variable as *V* /(max V - min *V*). In this case, the means, variances, and ranges of the variables are still different, but at least the ranges are likely to be more similar.

- Dividing each value by the **standard deviation**. This method produces a set of transformed variables with variances of 1, but different means and ranges.

- Zscore

4- Generalization

- The low-level or granular data that we have converted to high-level information by using concept hierarchies. We can transform the primitive data in the address like the city to higher-level information like the country.
- For example, house addresses can be generalized to higher-level definitions, such as town or country.

Original Data:

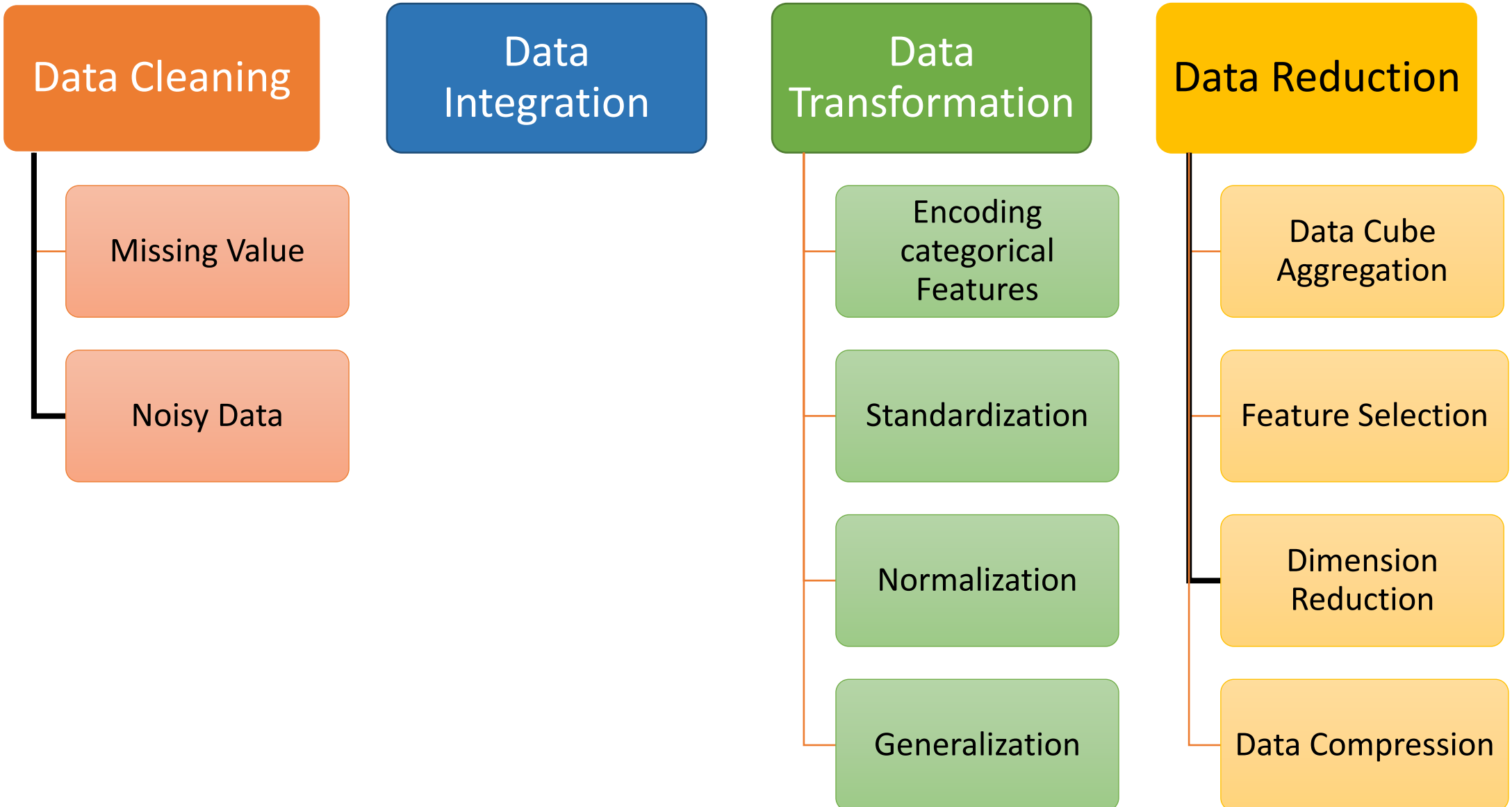
Ages: 26, 28, 31, 33, 37, 42, 42, 46, 48, 49, 54, 57, 57, 58, 59

Generalized Data:

Ages:

- 20-29 (2)
- 30-39 (3)
- 40-49 (5)
- 50-59 (5)

Data preprocessing Techniques



Name	Type	HP	Attack	Defense	Speed	Generation
Bulbasaur	Grass	45	49	49	45	1
Ivysaur	Grass	60	62	63	60	1
Venusaur	Grass	80	82	83	80	1
Charmander	Fire	39	52	43	65	1
Charmeleon	Fire	58	64	58	80	1

	HP	Attack	Defense	Speed	Generation
count	5.0	5.0	5.0	5.0	5.0
mean	56.4	61.8	59.2	66.0	1.0
std	15.9	13.0	15.4	14.7	0.0
min	39.0	49.0	43.0	45.0	1.0
25%	45.0	52.0	49.0	60.0	1.0
50%	58.0	62.0	58.0	65.0	1.0
75%	60.0	64.0	63.0	80.0	1.0
max	80.0	82.0	83.0	80.0	1.0

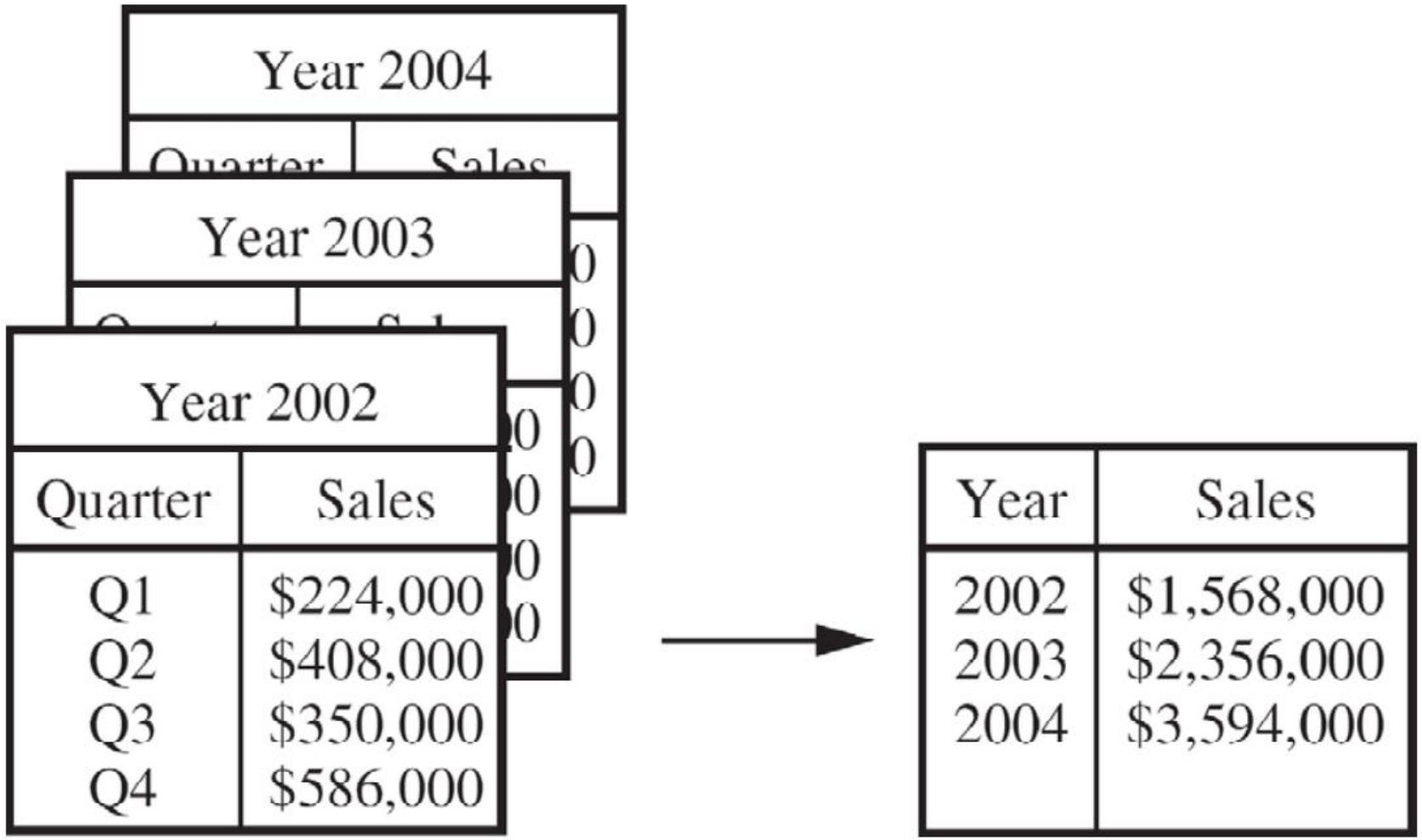
Data Reduction

- Data reduction strategies applied on huge data set. Complex data and mining on huge amounts of data can take a long time, making such analysis impractical or infeasible.
- Data reduction techniques can be applied to obtain a reduces data should be more efficient yet produce the same analytical results. Strategies for data reduction include the following

  - **Attribute subset selections**
  - **Data Cube Aggregation**
  - **Dimensionality reduction**
  - **Data Compression**

1- Data Cube Aggregation

- In the case of data aggregation, the data is pooled together and presented in a unified format for data analysis.
- Working with a large amount of high-quality data allows for getting more reliable results from the ML model.
- If we want to build a neural network algorithm that simulates the style of Vincent Van Gogh, we need to provide as many paintings by this famous artist as we can to provide enough material for training. The images need to have the same digital format, and we will use data transformation techniques to achieve that


Year 2004
Year 2003
Year 2002
Quarter
Sales
Q1
$224,000
Q2
$408,000
Q3
$350,000
Q4
$586,000
Year
Sales
2002
$1,568,000
2003
$2,356,000
2004
$3,594,000

France	TRUE	couplerio-sandb	2915	29	Hand LLC deal	22	USD	8/31/2019	10/18/2019
France	TRUE	couplerio-sandb	2947	30	Ortiz, Farrell anc	669	USD	8/31/2019	10/18/2019
France	TRUE	couplerio-sandb	1604	28	Roob-Nader dea	622	USD	7/31/2018	10/18/2019
France	TRUE	couplerio-sandb	2995	30	Medhurst-Padbe	830	USD	9/30/2019	10/18/2019
France	TRUE	couplerio-sandb	3093	29	Hintz, Doyle and	497	USD	10/31/2019	10/18/2019
France	TRUE	couplerio-sandb	3112	29	Parker-Schmele	75	USD	10/31/2019	10/18/2019
France	TRUE	couplerio-sandb	3123	29	Kreiger-Sauer de	142	USD	10/31/2019	10/18/2019
France	TRUE	couplerio-sandb	3142	29	Keeling, Crooks	51	USD	10/31/2019	10/18/2019
France	TRUE	couplerio-sandb	3170	29	Wisozk-Ullrich d	228	USD	10/31/2019	10/18/2019
France	TRUE	couplerio-sandb	1628	28	Adams, Kling an	628	USD	7/31/2018	10/18/2019
France	TRUE	couplerio-sandb	3245	29	Harvey, Wolf anc	647	USD	11/30/2019	10/18/2019
France	TRUE	couplerio-sandb	3283	30	Schinner, Glover	945	USD	11/30/2019	10/18/2019
France	TRUE	couplerio-sandb	3294	29	Mitchell-Purdy d	867	USD	11/30/2019	10/18/2019
France	TRUE	couplerio-sandb	2915	30	Hand LLC deal	39	USD	11/30/2019	10/18/2019
France	TRUE	couplerio-sandb	1644	28	Barto				
France	TRUE	couplerio-sandb	1647	28	Bauc				
France	TRUE	couplerio-sandb	1679	28	Stron				
France	TRUE	couplerio-sandb	1698	28	Braku				
France	TRUE	couplerio-sandb	1704	30	Mant				
France	TRUE	couplerio-sandb	1708	28	Hartn				
France	TRUE	couplerio-sandb	1714	30	Adan				
France	TRUE	couplerio-sandb	1726	28	Hilll L				
France	TRUE	couplerio-sandb	1733	28	Walsl				

Country	Conversion rate	Total revenue
Australia	32.01%	$42,851.00
Canada	28.57%	$38,630.00
Denmark	28.47%	$34,078.00
France	27.89%	$39,561.00
Germany	30.10%	$43,460.00
Netherlands	28.09%	$45,102.00
Ukraine	28.04%	$31,025.00
United Kingdom	33.44%	$52,149.00
United States	29.50%	$40,088.00
United Arab Emirates	27.84%	$38,934.00

Total deals	Total revenue	Average deal life time (days)
299	$45,102.00	50

2- Attribute/Feature Selection

- Techniques for data transformation can also be used for data reduction. If you construct a new feature combining the given features in order to make the data mining process more efficient, it is called an attribute selection.
- For example, the features ***male/female*** and ***student*** can be constructed into *male student/female student*. This can be useful if we conduct research about how many men and/or women are students but their study field doesn't interest us.

- **Redundant attributes**
  - duplicate much or all of the information contained in one or more other attributes
  - Example: purchase price of a product and the amount of sales tax paid
- **Irrelevant attributes**
  - contain no information that is useful for the data mining task at hand
  - Example: students' ID is often irrelevant to the task of predicting students' GPA

- different types of general feature selection methods
  - Filter methods
  - Wrapper methods
  - Embedded methods

- The following image best describes filter-based feature selection methods:

Feature Selection in Python Sklearn - DataCamp

Filter Method

- Filter methods measure the relevance of features by their correlation with dependent variable
- Filter method uses the exact assessment criterion which includes distance, information, dependency, and consistency. The filter method uses the principal criteria of ranking technique and uses the rank ordering method for variable selection

- Some examples of some filter methods include the Chi-squared test, information gain, and correlation coefficient scores

Feature\Response	Continuous	Categorical
Continuous	Pearson's Correlation	LDA
Categorical	Anova	Chi-Square

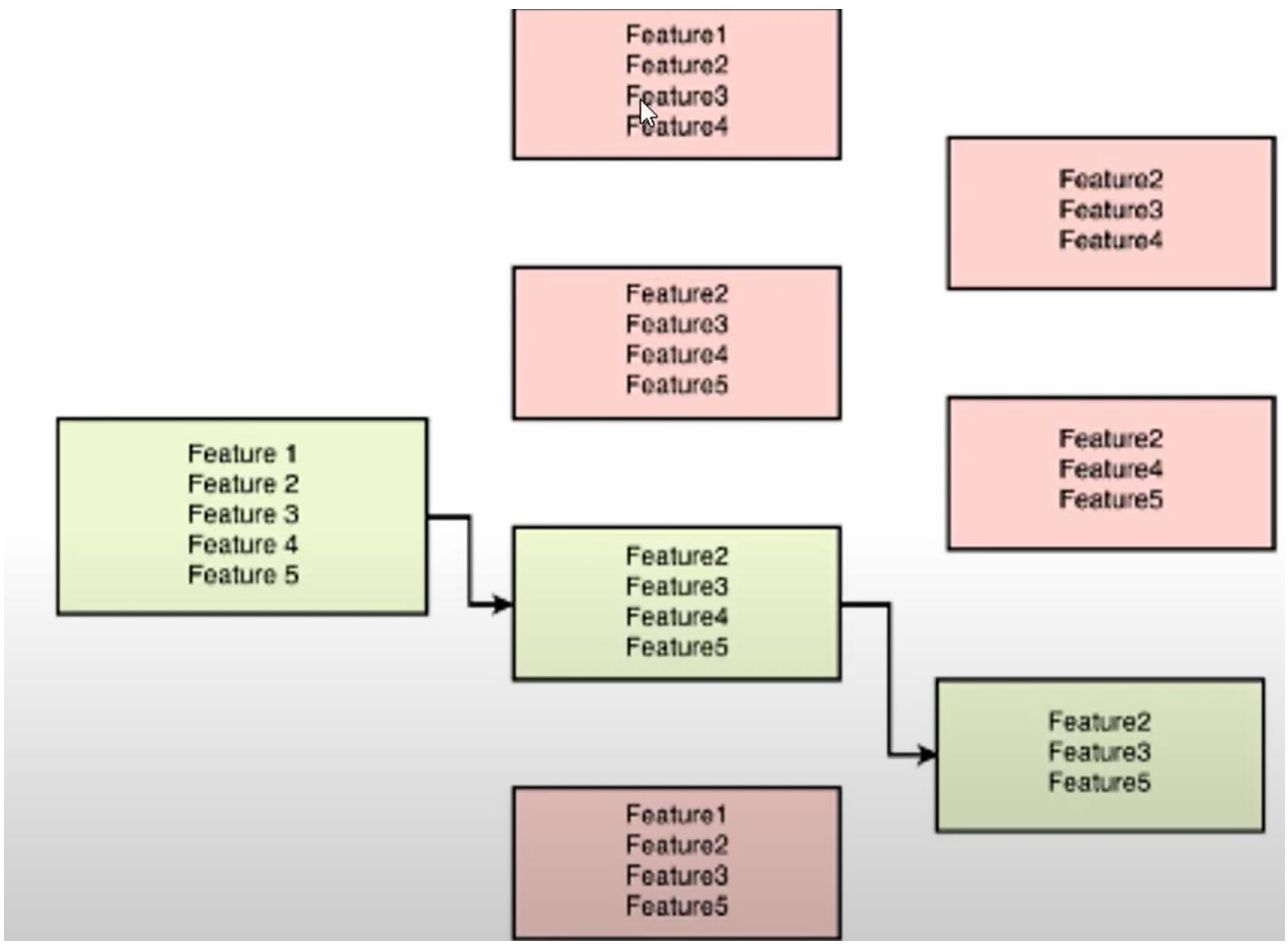
Wrapper Method

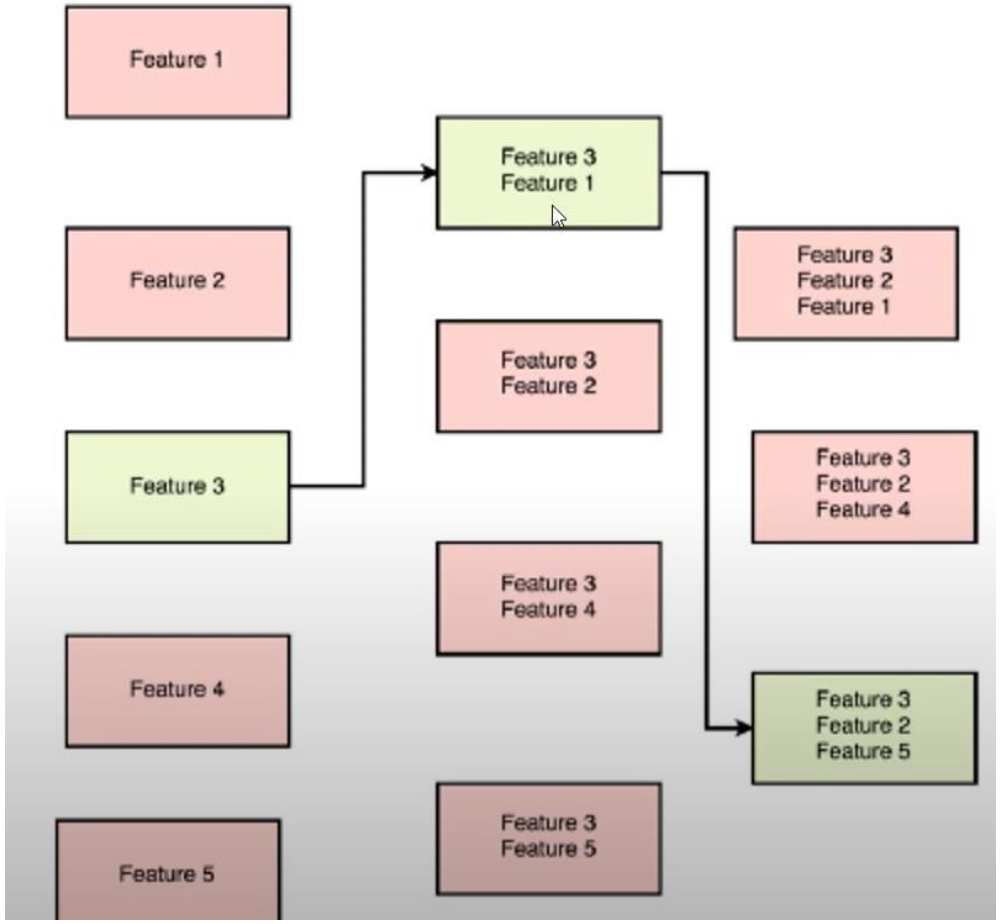
- This method searches for a feature which is best-suited for the machine learning algorithm and aims to improve the mining performance. Some typical examples of wrapper methods are:

  - Forward Selection: The procedure starts with an empty set of features. The best of the original features is determined and added to the reduced set. At each subsequent iteration, the best of the remaining original attributes is added to the set.

  - Backward Elimination: The procedure starts with the full set of attributes. At each step, it removes the worst attribute remaining in the set.

  - Combination of forward selection and backward elimination: The stepwise forward selection and backward elimination methods can be combined so that, at each step, the procedure selects the best attribute and removes the worst from among the remaining attributes.


Feature1
Feature2
Feature3
Feature4
Feature2
Feature3
Feature4
Feature2
Feature3
Feature4
Feature5
Feature2
Feature4
Feature5
Feature 1
Feature 2
Feature 3
Feature 4
Feature 5
Feature2
Feature3
Feature4
Feature5
Feature2
Feature3
Feature5
Feature1
Feature2
Feature3
Feature5


Feature 1
Feature 3
Feature 1
Feature 2
Feature 3
Feature 2
Feature 1
Feature 3
Feature 2
Feature 3
Feature 4
Feature 3
Feature 4
Feature 4
Feature 3
Feature 2
Feature 5
Feature 5
Feature 3
Feature 5

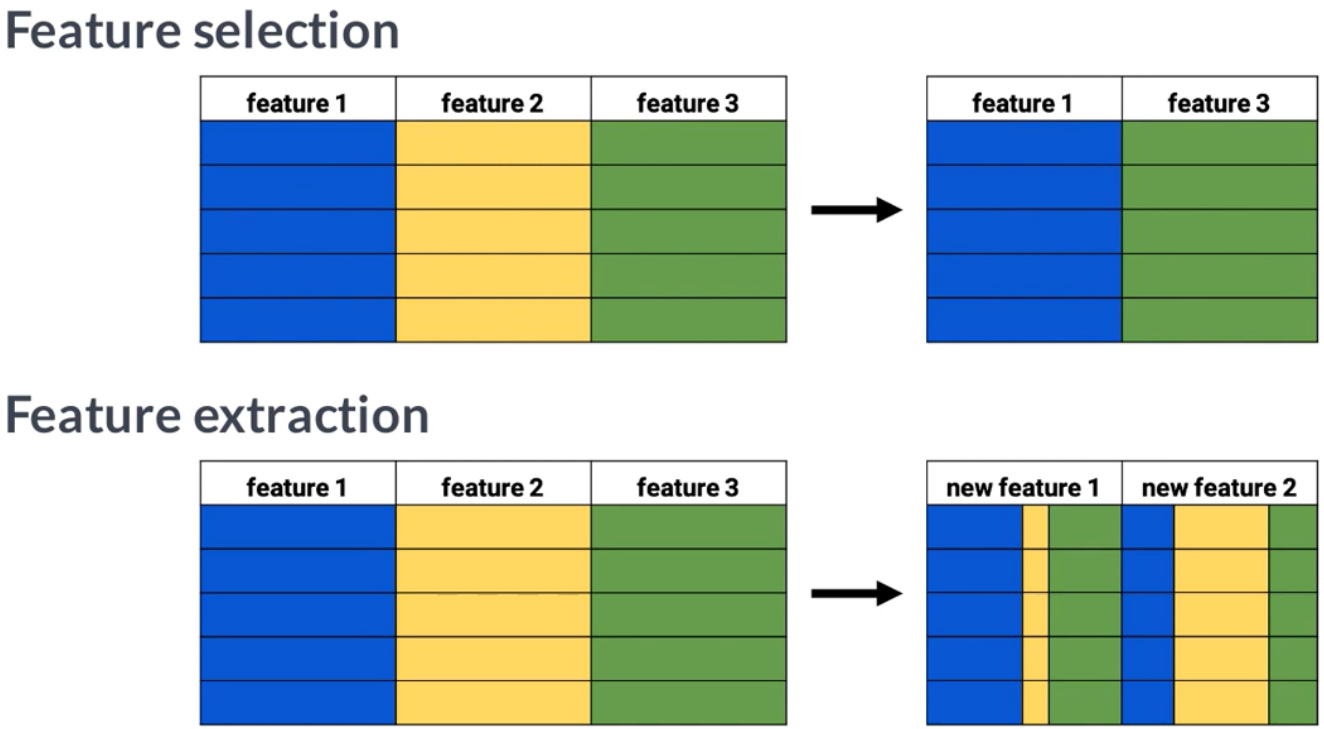
Embedded Method

- Embedded methods are iterative in a sense that takes care of each iteration of the model training process and carefully extract those features which contribute the most to the training for a particular iteration.

    - Lasso Regression
    - Ridge Regression
    - Tree Regression

Feature selection vs extraction:

- **Feature selection**, also known as variable selection, is the process of selecting a subset of relevant features (variables, predictors) for use in model construction.
- **Feature extraction** is for creating a new, smaller set of features that still captures most of the useful information.
- Feature selection keeps a subset of the original features while feature extraction creates new ones.

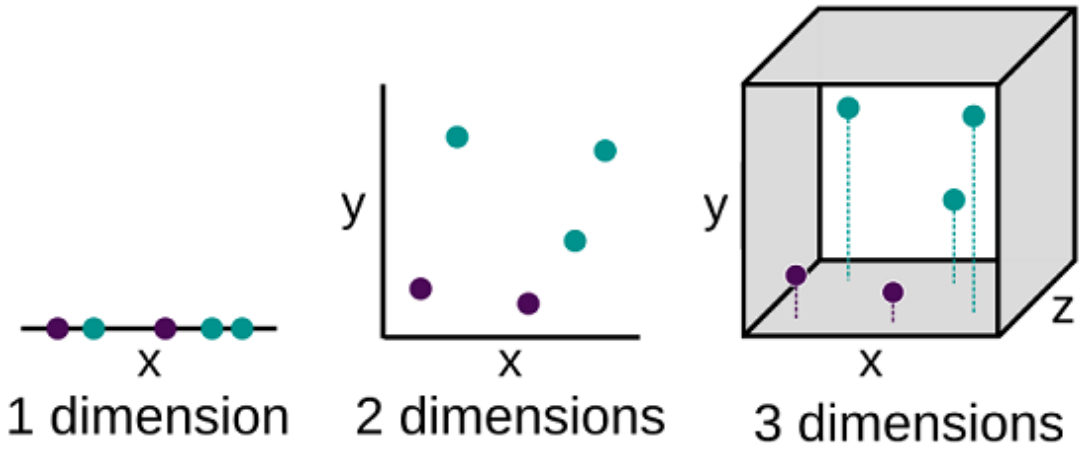


Python Tutorial: Feature selection vs feature extraction - YouTube

3- Dimension Reduction/Feature Extraction

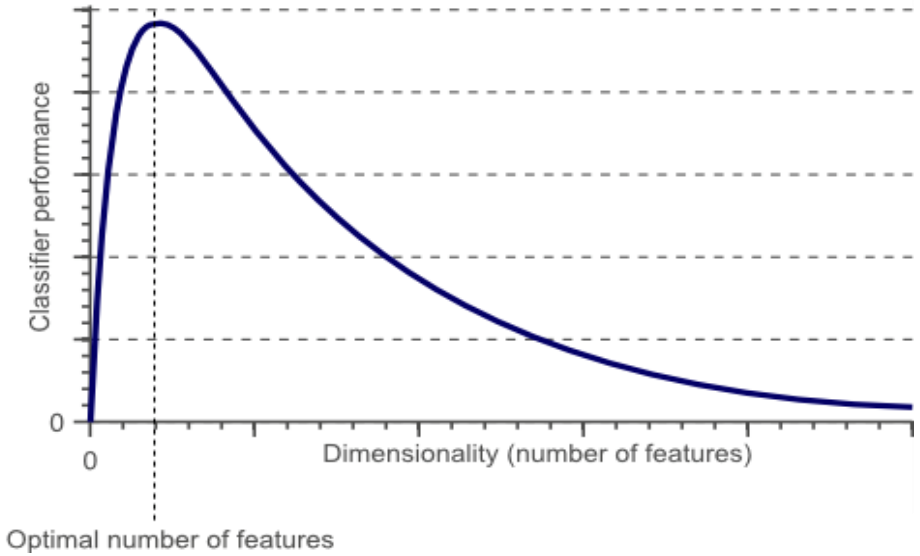
- Datasets that are used to solve real-life tasks have a huge number of features. Computer vision, speech generation, translation, and many other tasks cannot sacrifice the speed of operation for the sake of quality. It’s possible to use dimensionality reduction to cut the number of features used.

Dimensionality reduction is used to solve the curse of dimensionality. Higher dimensions causes data sparsity or require huge datasets for true representations.



Data become more sparse as the number of dimensions increases. Two classes are shown in one, two and three-dimensional feature space. The dotted lines in the three-dimensional representation are to clarify the position of the points along the z axis. Note the increasing empty space with increased dimensions.

- Any machine learning model that will be used to analyze the data will be affected by the number of variables/features.
- As the number of features increases, the model becomes more complex.
- A machine learning model that is trained on a large number of features, gets increasingly dependent on the data it was trained on and in turn overfitted, resulting in poor performance on real data, beating the purpose.



Advantages of using Dimensionality reduction:

- Less misleading data means model accuracy improves, avoids overfitting.
- Less dimensions mean less computing. Less data means that algorithms train faster.
- Less data means less storage space required.
- Extremely useful for data visualization
- Less dimensions allow usage of algorithms unfit for a large number of dimensions
- Removes redundant features and noise.

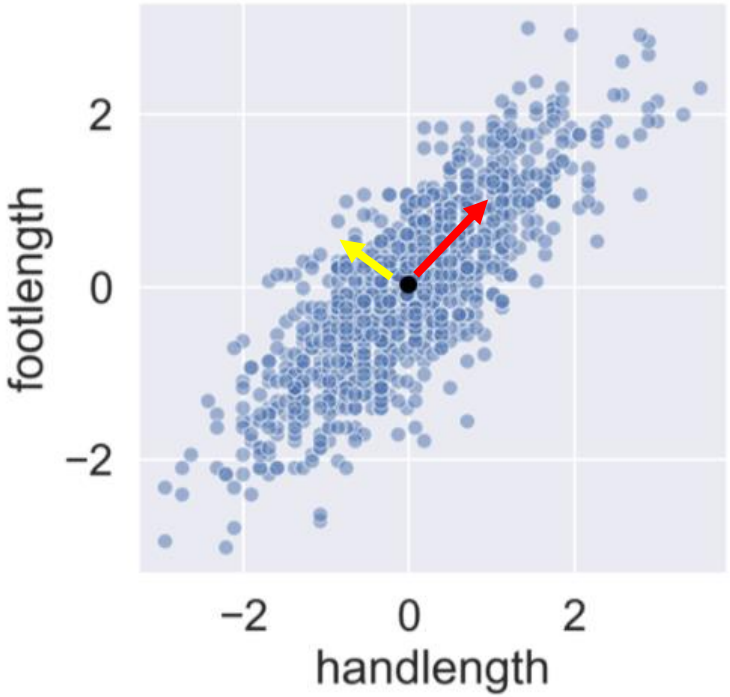
Feature extraction

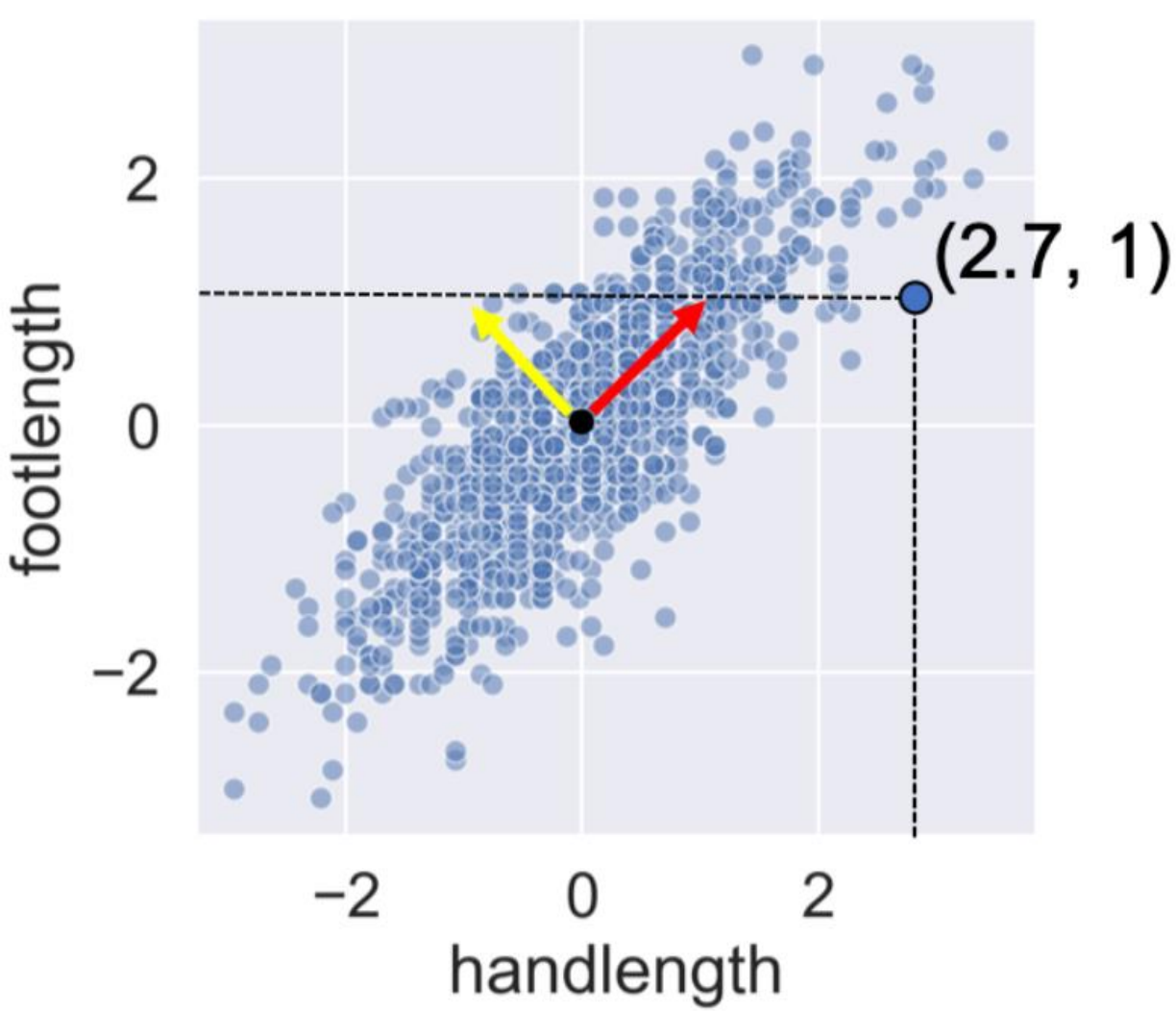
1) **Linear Dimensionality Reduction Methods**
   a) Principal Component Analysis (PCA). Singular value decomposition (SVD)
   b) Factor Analysis (FA)
   c) Linear discriminant Analysis (LDA)
2) **Non-Linear Dimensionality Reduction Methods**
   a) Multi-dimensional scaling (MDS)
   b) Isometric Feature Mapping (Isomap)
   c) Locally Linear Embedding (LLE)
   d) Kernel PCA,
   e) Hessian Eigenmapping (HLLE)
   f) Spectral Embedding (Laplacian Eigenmaps)
   g) t-distributed Stochastic Neighbor Embedding (t-SNE)

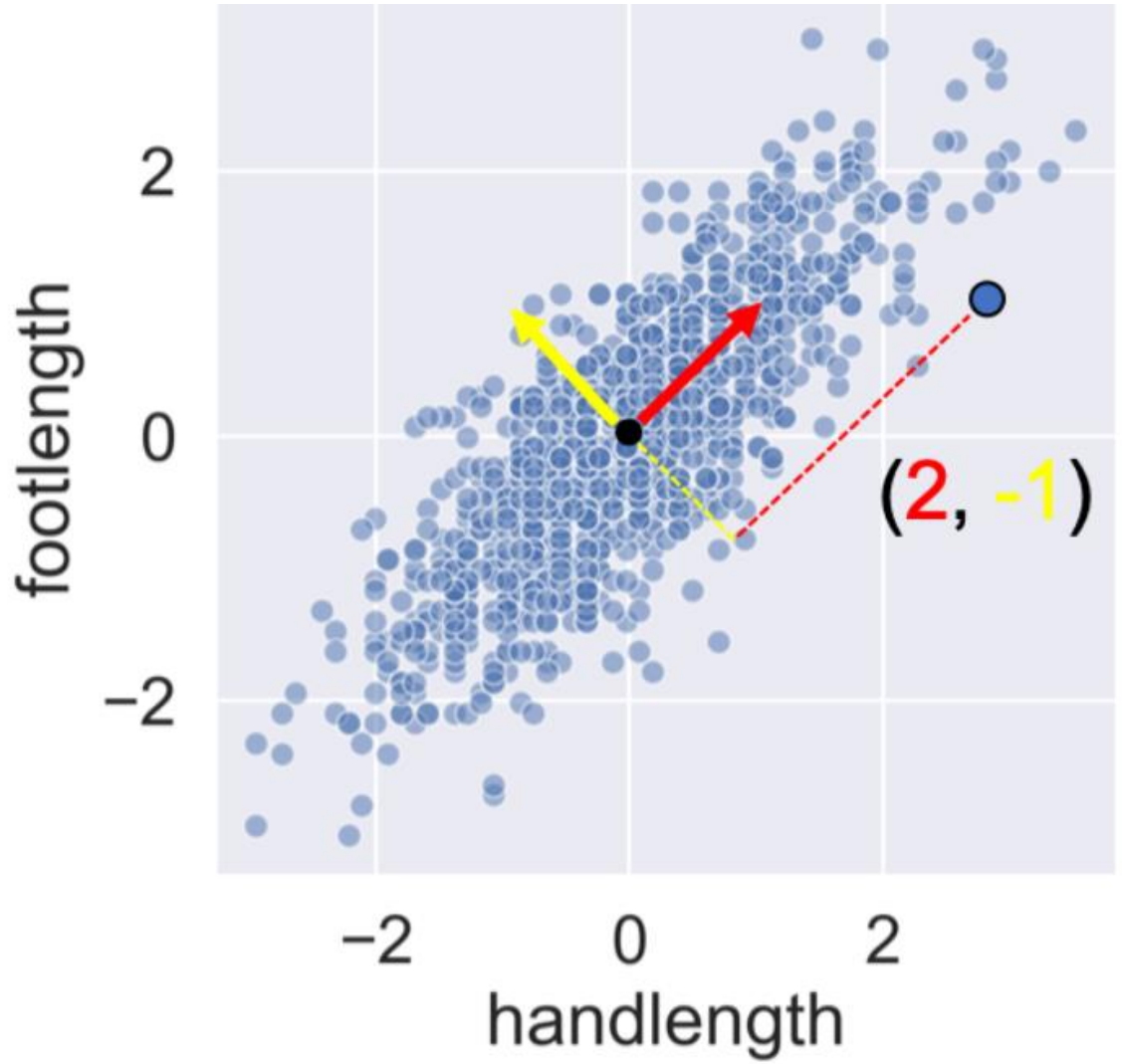
PCA: Principal Component Analysis

- For this technique it's important to scale the features first, so that their values are easier to compare.

```
scaler = StandardScaler()
df_std = pd.DataFrame(scaler.fit_transform(df), columns = df.columns)
```




2
0
−2
footlength
(2.7, 1)
−2
0
2
handlength

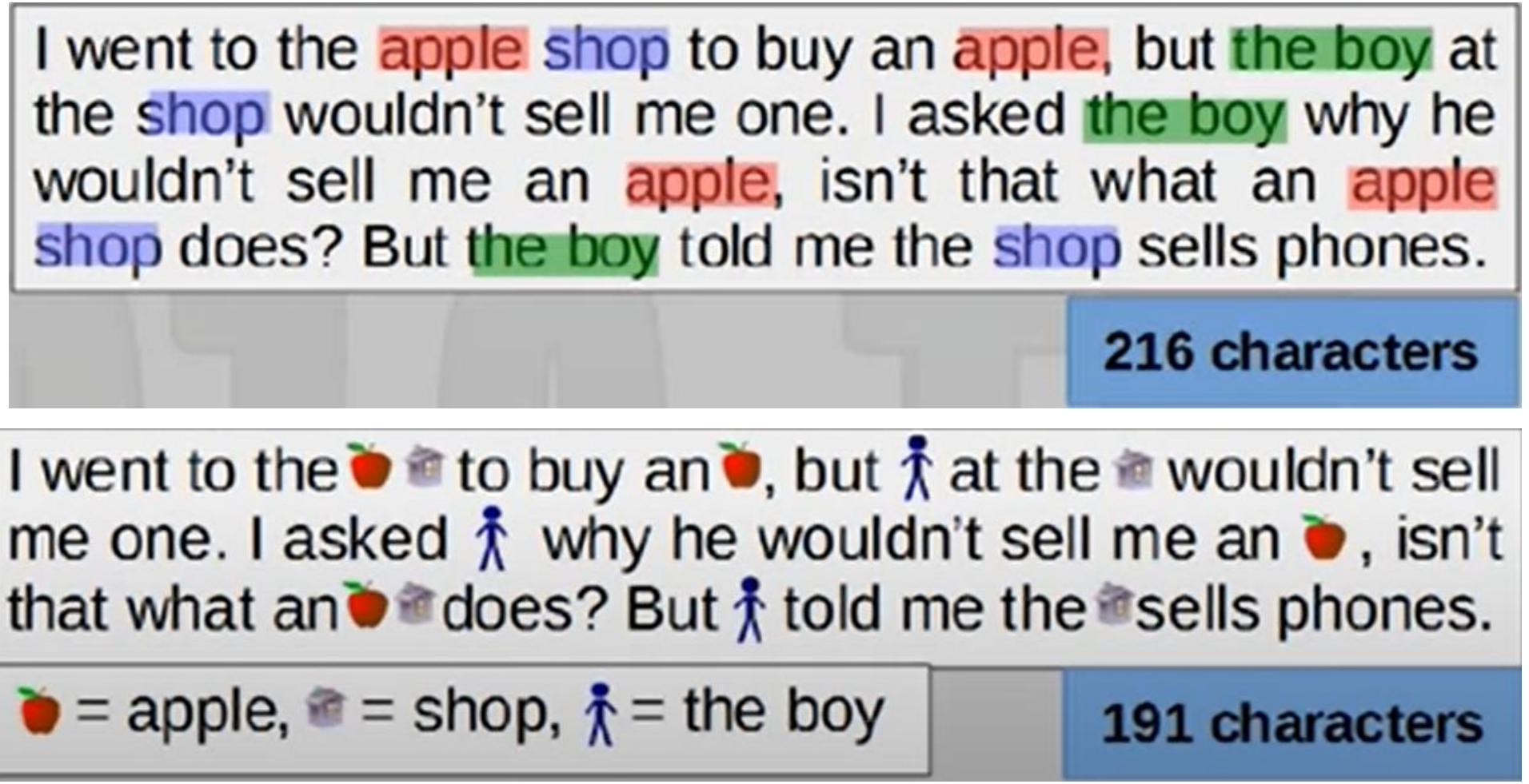

2
0
−2
footlength
(2, -1)
−2
0
2
handlength

4- Data Compression

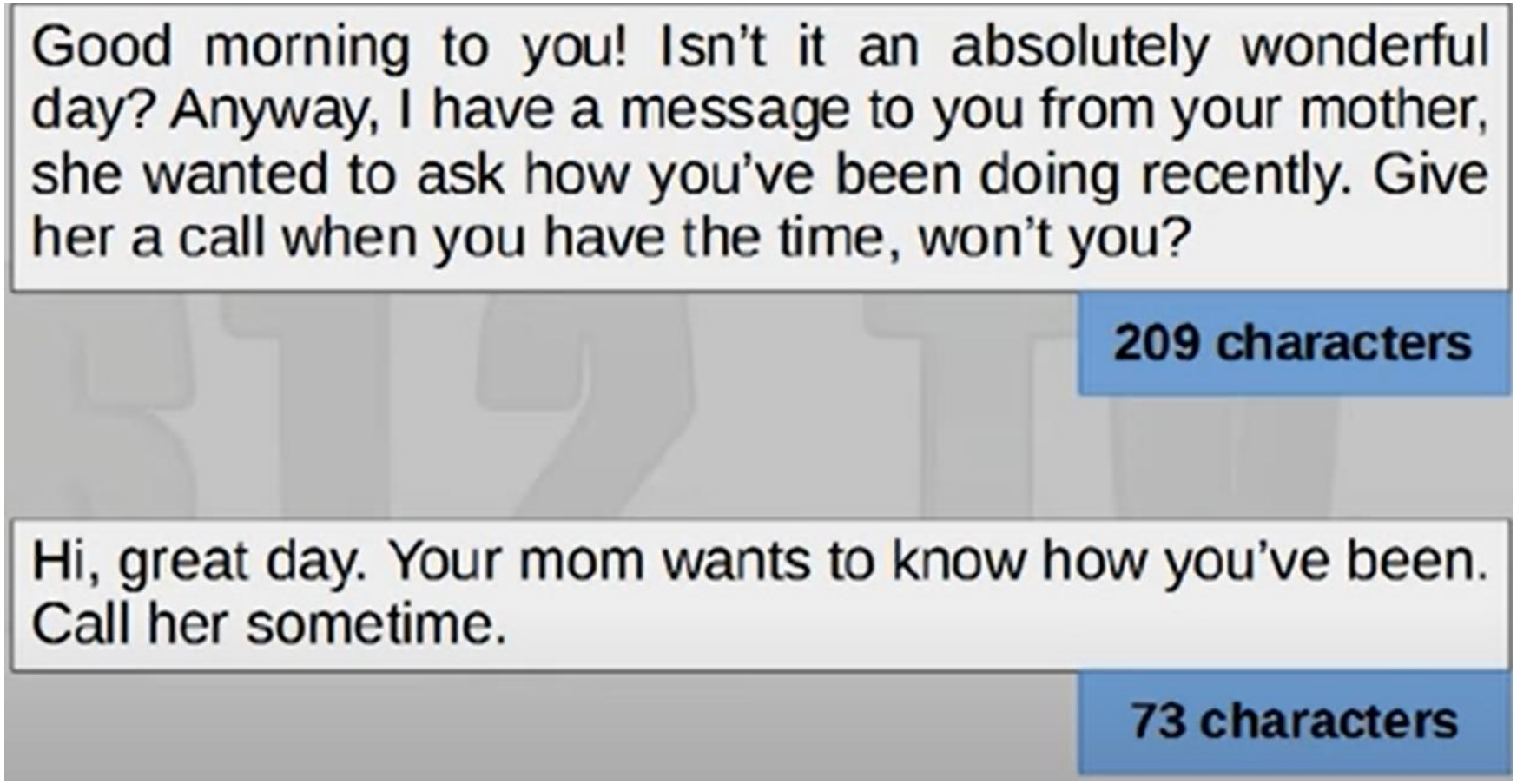
- Data compression is the process of encoding, restructuring or otherwise modifying data in order to reduce its size
- Data Compression is also referred to as bit-rate reduction or source coding. This technique is used to reduce the size of large files.
- The advantage of data compression is that it helps us save our disk space and time in the data transmission.
- There are mainly two types of data compression techniques
  1. **Lossless Data Compression**
  2. **Lossy Data Compression**

- **Lossless Compression –**
  Encoding techniques (Run Length Encoding) allows a simple and minimal data size reduction. Lossless data compression uses algorithms to restore the precise original data from the compressed data.
- **Lossy Compression –**
  Methods such as Discrete Wavelet transform technique, PCA (principal component analysis) are examples of this compression. For e.g., JPEG image format is a lossy compression, but we can find the meaning equivalent to the original the image. In lossy-data compression, the decompressed data may differ to the original data but are useful enough to retrieve information from them.

Lossless Data Compression



Lossy Data Compression



Data Preprocessing Tools

- Talend data preparation
- Trifacta Wranger
- OpenRefine
- Google DataPrep
- Tabula
- Python
  - Pandas
- …

Data Preprocessing with Pandas DataFrames in Python

Pandas

- Numpy, Scipy, Cython and Panda are the tools available in python which can be used fast processing of the data.

- Working with Panda is fast, easy and more expressive than other tools.

- Pandas integrates well with matplotlib library, which makes it very handy tool for analyzing the data.

Pandas Data Structures

- Pandas provides two very useful data structures to process the data i.e. Series and DataFrame

10	23	56	17	52	61	73	90	26	72

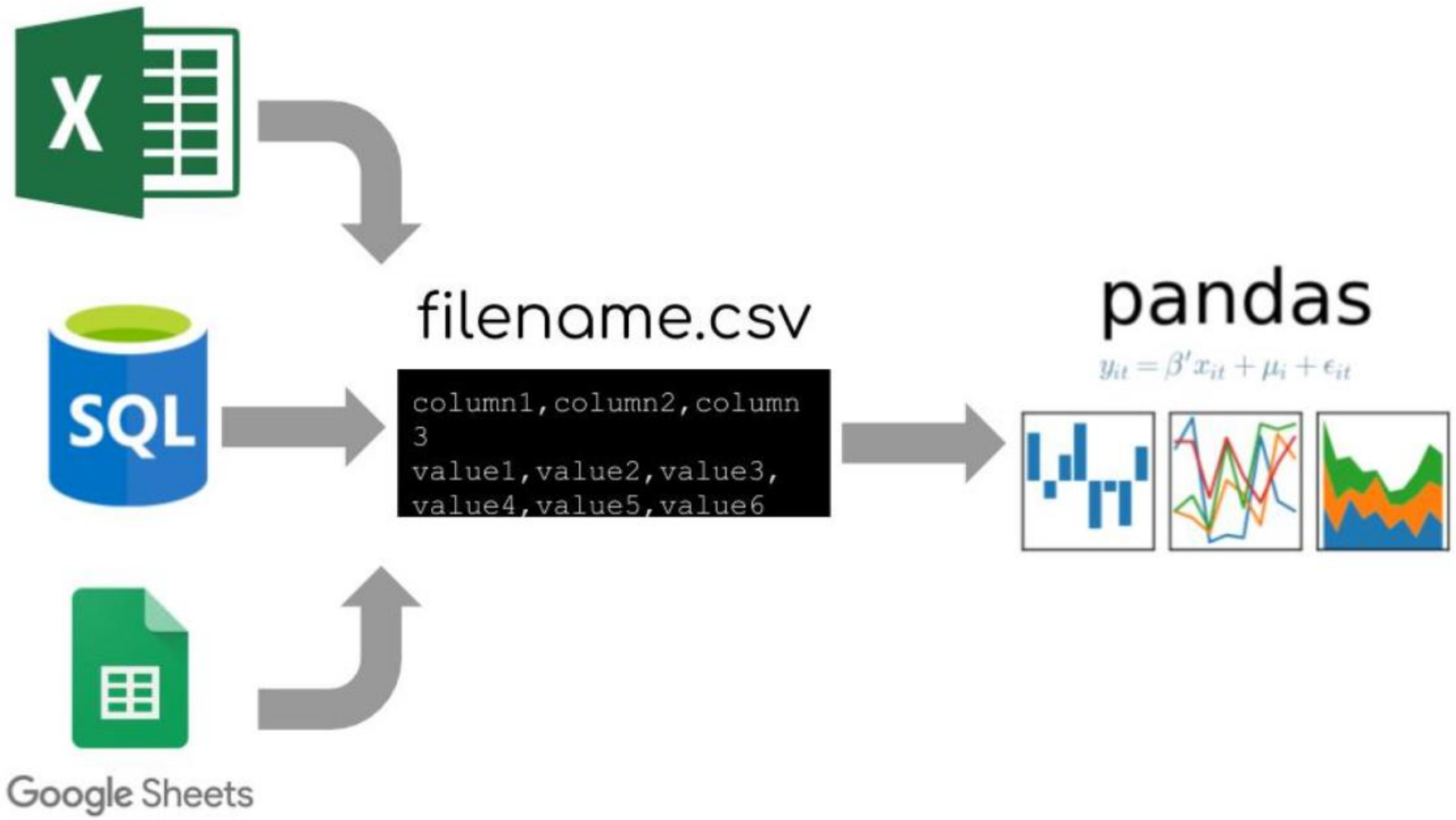
Name	Age	Gender	Rating
Steve	32	Male	3.45
Lia	28	Female	4.6
Vin	45	Male	3.9
Katie	38	Female	2.78

Data Structure	Dimensions	Description
Series	1	1D labeled homogeneous array, size-immutable.
Data Frames	2	General 2D labeled, size-mutable tabular structure with potentially heterogeneously-typed columns.

Pandas DataFrame

- DataFrame is the widely used data structure of pandas
- DataFrame can be used with two dimensional arrays. DataFrame has two different index i.e. column-index and row-index.

- The most common way to create a DataFrame is by using the dictionary of equal-length. Further, all the spreadsheets and text files are read as DataFrame, therefore it is very important data structure of pandas(such as csv, txt, and xlsx).


filename.csv
column1,column2,column
3
value1,value2,value3,
value4,value5,value6
pandas
$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$
SQL
Google Sheets

What can you do with a Pandas DataFrame?

- Filter
  - Select rows/columns
- Sort
- Numerical or Mathematical operations (e.g. mean)
- Group by column(s)
- Many others!

  https://pandas.pydata.org/pandas-docs/stable/

DataFrame Basic Functionality

1	**T** Transposes rows and columns.
2	**axes** Returns a list with the row axis labels and column axis labels as the only members.
3	**dtypes** Returns the dtypes in this object.
4	**empty** True if NDFrame is entirely empty [no items]; if any of the axes are of length 0.
5	**ndim** Number of axes / array dimensions.
6	**shape** Returns a tuple representing the dimensionality of the DataFrame.
7	**size** Number of elements in the NDFrame.
8	**values** Numpy representation of NDFrame.
9	**head()** Returns the first n rows.
10	**tail()**