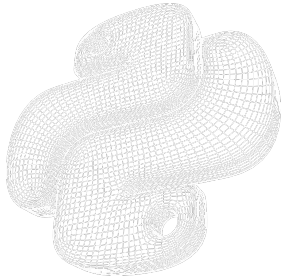
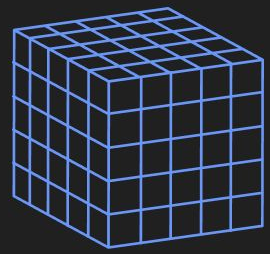
Python для Data Science



Заняття 6.
Пошук і видалення пропущених значень

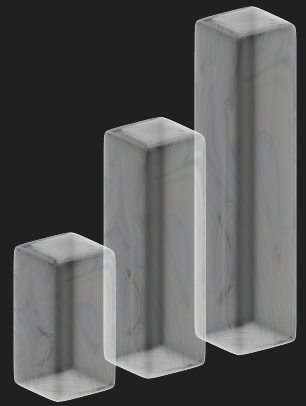
План заняття

- Типи пропущених значень: випадково / невипадково / внаслідок помилки
- Способи обробки пропущених значень
  - видалення
  - заповнення
- Обробка пропущених даних у часових рядах



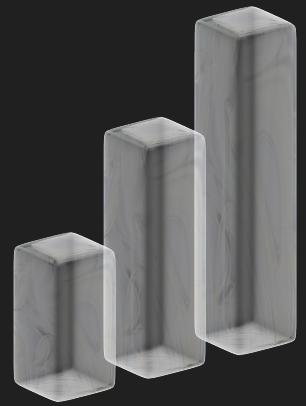
Чому ми аналізуємо пропущені значення?

- Більшість реальних датасетів містять пропущені значення
- Пропущені дані можуть спотворити результати моделей машинного навчання або знизити точність моделі
- Деякі моделі машинного навчання взагалі не працюють із пропущеними даними



Типи пропущених значень

- Missing completely at random, MCAR (пропущені повністю випадковим чином)
- Missing at random, MAR (пропущені випадково)
- Missing not at random, MNAR (пропущені невипадково)



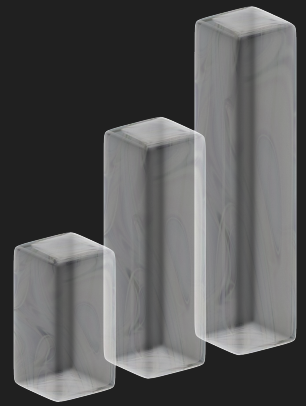
Missing completely at random, MCAR

Імовірність відсутності не залежить від даних. Тобто причина відсутності даних не має нічого спільного ні зі значеннями даних спостереження, ні зі значеннями пропущених даних.

P(Missing | Observed_values, Missing_values) = const

Приклади:

- ➔ дані анкети не записалися в базу даних
- ➔ зразок крові було пошкоджено в лабораторії



Missing at random, MAR

Імовірність відсутності не залежить від значень пропущених даних, але залежить від даних спостереження. Іншими словами, ми можемо передбачити, наскільки ймовірно значення буде відсутнє на основі даних, які не пропущені.

P(Missing | Observed_values, Missing_values) = f(Observed_values)

Приклади:

- ➔ вагу вимірюють лише у тих пацієнтів, які мають високий тиск
- ➔ чоловіки рідше заповнюють анкети про їхній рівень стресу, але це ніяк не залежить від їхнього рівня стресу як такого

MAR є ширшим і більш реалістичним, ніж MCAR.

Missing not at random, MNAR

Причина відсутності даних пов’язана з неспостережуваними даними, тобто даними, яких у нас немає, відсутність пов’язана з факторами, які ми не врахували.

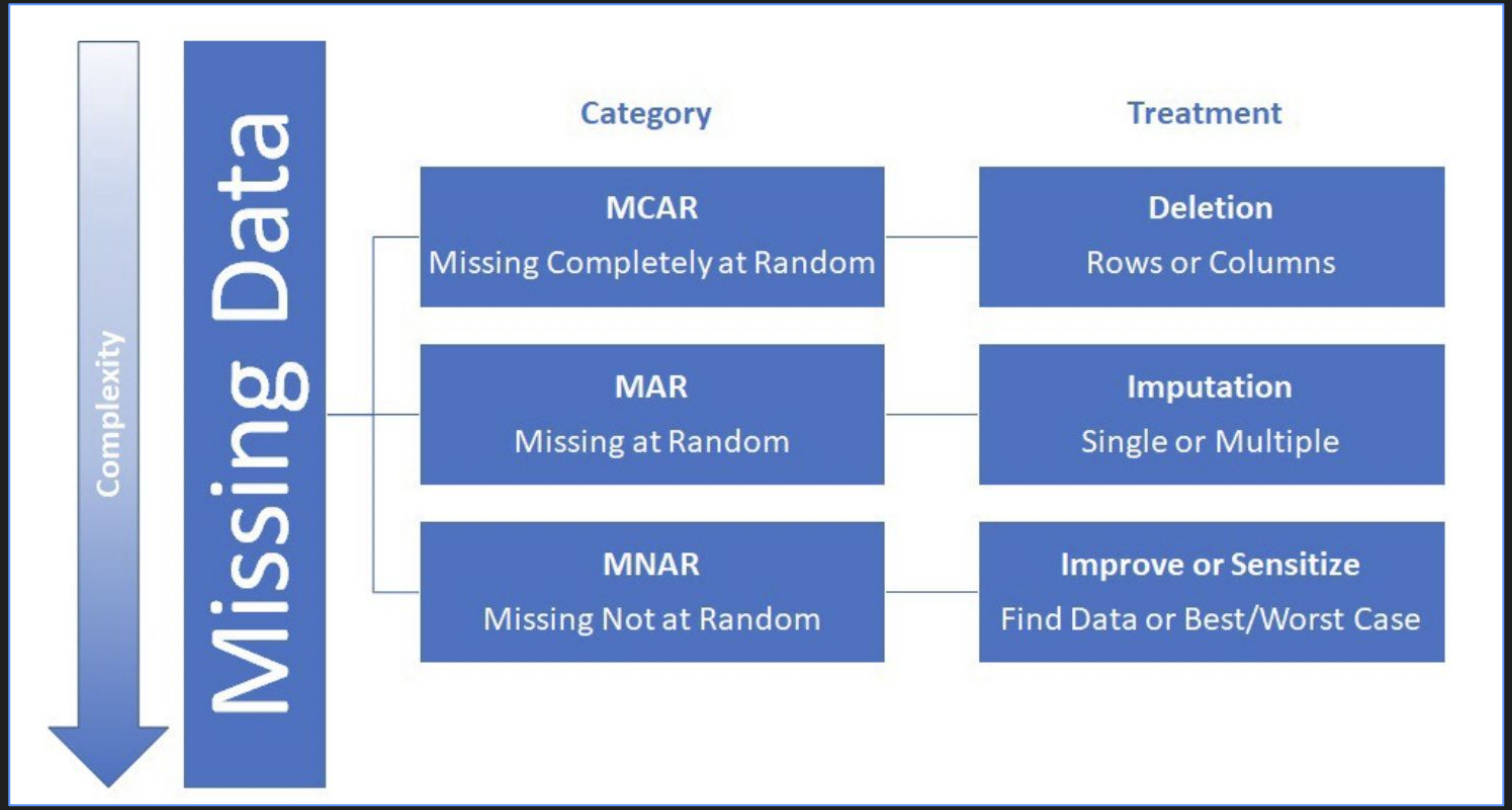
P(Missing | Observed_values, Missing_values) = f(Observed_values, Missing_values)

Приклади:

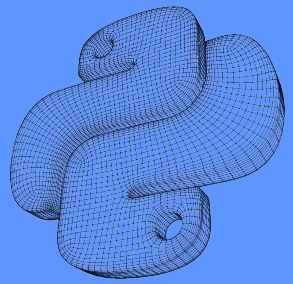
- ➔ кількість опадів не можна виміряти через надто сильний дощ
- ➔ пацієнт пропустив подачу аналізів через погане самопочуття від препаратів, що він приймає (а ми намагаємося побудувати модель оцінки впливу препарату на одужання від певної хвороби)

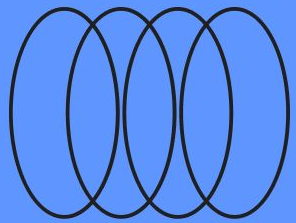
Складність обробки MNAR полягає в тому, що причини пропуску залежать від невідомої нам інформації. Стратегії обробки MNAR полягають у пошуку додаткових даних про причини відсутності або виконанні аналізу what-if, щоб побачити, наскільки чутливими є результати за різних сценаріїв.

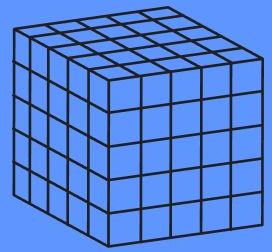
Handling missing data depends on type



Запитання

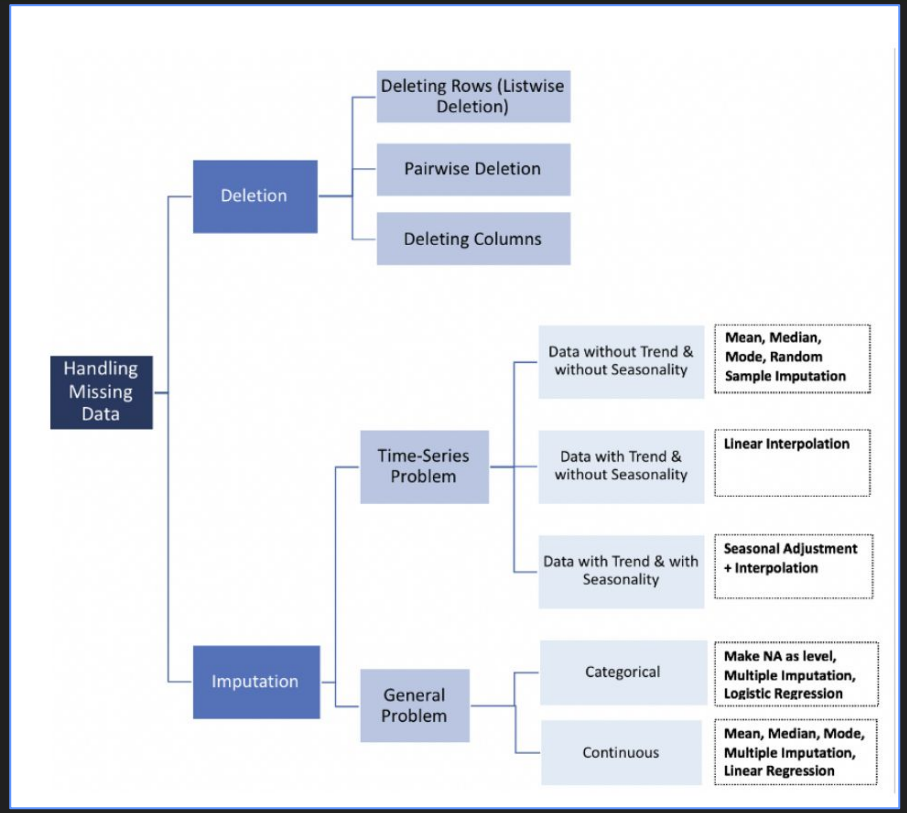






Handling missing data

- нічого не робити
- видалення
- заповнення



Do nothing

В такому разі ви дозволяєте алгоритму самому опрацьовувати пропущені значення.

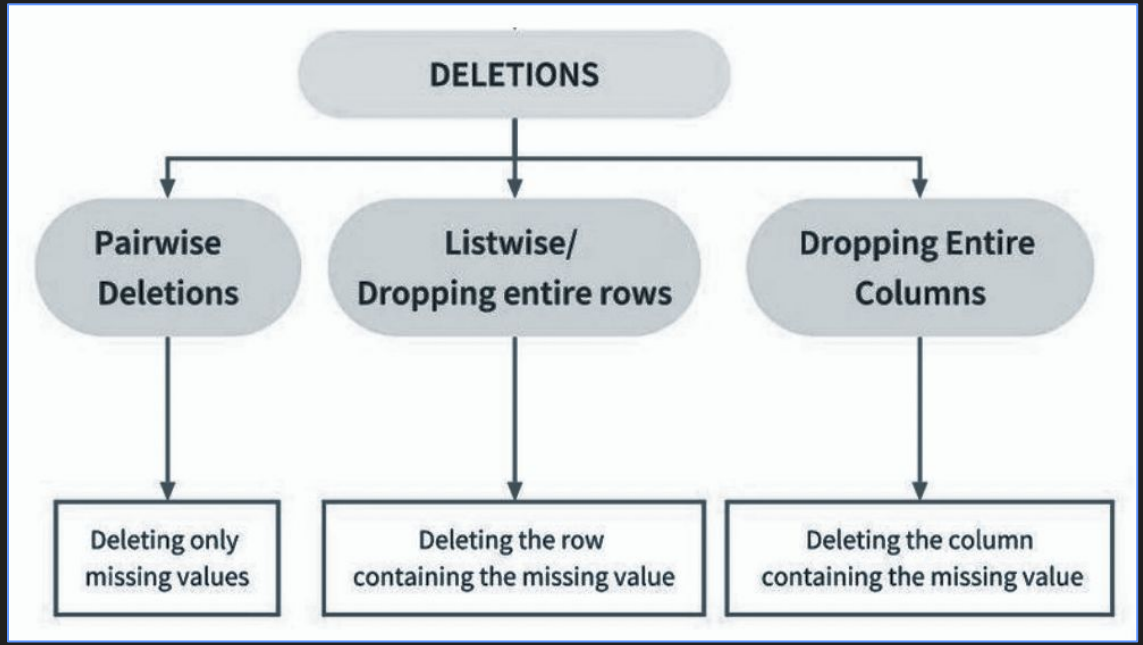
Деякі алгоритми можуть враховувати пропущені значення і визначати значення для їх заповнення на основі зменшення функції втрат при навчанні (XGBoost).

Деякі алгоритми можуть просто ігнорувати пропущені значення (LightGBM — use_missing = false).

Але багато алгоритмів видаватимуть помилку під час отримання даних з пропущеними значеннями (Scikit learn — LinearRegression). В такому разі потрібно опрацьовувати й очищати пропущені дані перед тим, як тренувати алгоритм.

Deletion

! Видалення — це втрата даних. Тому цей підхід можна використовувати, якщо обсяг даних дозволяє і частка пропущених значень невелика (2 %).



Deletion: pairwise (available case analysis — ACA)

Досліджуючи залежності між колонками, ми видаляємо лиш ті точки даних (рядки), де є пропущені значення в досліджуваних стовпчиках. Тобто, якщо ми дивимося на кореляцію стовпчиків A і B, то нас не цікавлять пропущені значення в C.

+ для побудови моделі використовують усю доступну інформацію

- якщо видалити попарно, ви отримаєте різну кількість спостережень, які впливатимуть на різні частини вашої моделі, що може ускладнити інтерпретацію

Mobile ID	Mobile Package	Download Speed	Data Limit Usage	
1	Fast+	157	80%	
2	Lite	99	70%	
3	Fast+	167	10%	
4	Fast+	N/A	80%	Delete
5	Lite	76	70%	
6	Fast+	155	10%	
7	N/A	N/A	95%	Delete
8	Lite	76	77%	
9	Fast+	180	N/A	Delete

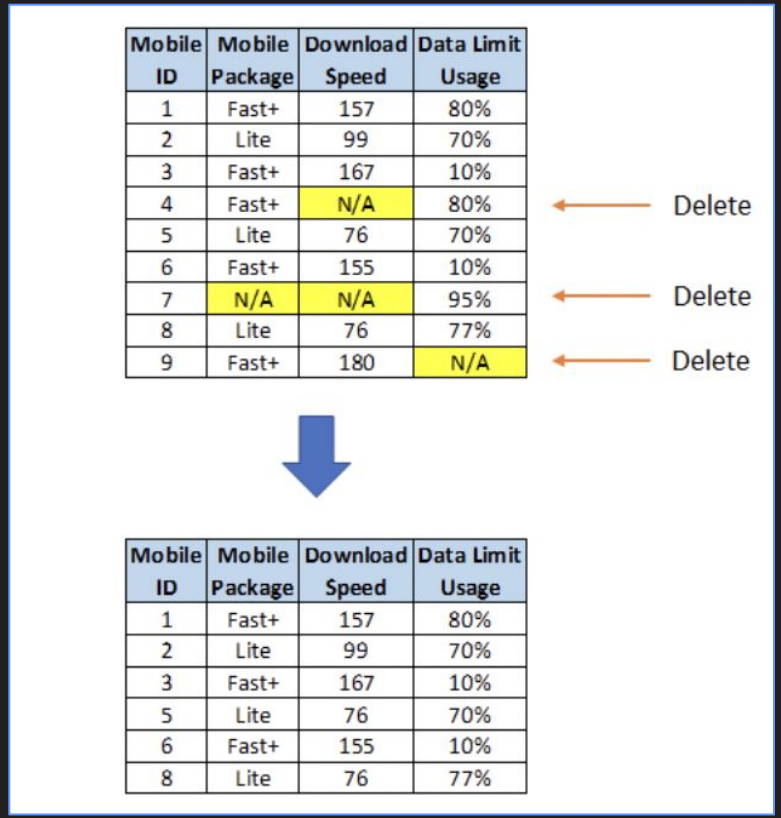
Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+		80%
5	Lite	76	70%
6	Fast+	155	10%
7			95%
8	Lite	76	77%
9	Fast+	180	

Deletion: listwise (Complete-case analysis — CCA)

Метод обробки пропущених значень, за якого видаляють усі спостереження (рядки) з пропущеними ознаками.

+ найпростіший спосіб обробки пропущених значень

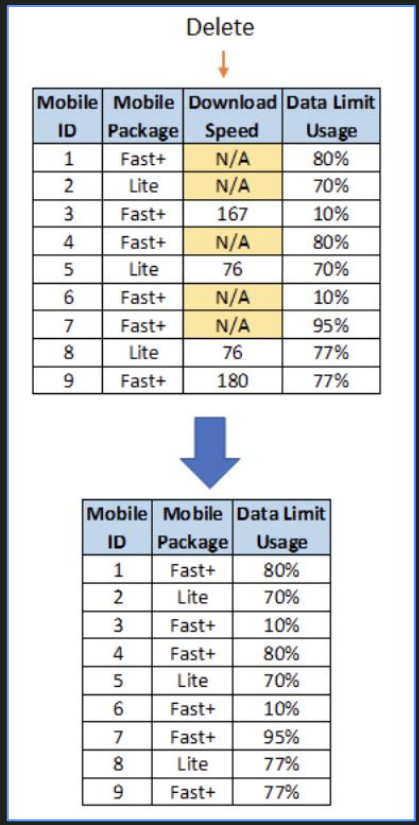
- втрачається багато інформації для навчання моделі
- у випадку MAR, MNAR такий спосіб призведе до внесення упередженості (bias) в модель. Наприклад, деякі респонденти відмовляються відповідати на якісь особисті питання. І у разі видалення таких відповідей ми матимемо дані лише про людей готових відповідати на подібні запитання



Deletion: dropping variable

Якщо для змінної (одного стовпчика) бракує забагато даних (25–50 %), можна видалити змінну або стовпчик із набору даних. Немає емпіричного правила, в яких саме випадках варто це робити, це залежить від ситуації, і належний аналіз даних потрібен перед тим, як змінну буде повністю відкинуто.

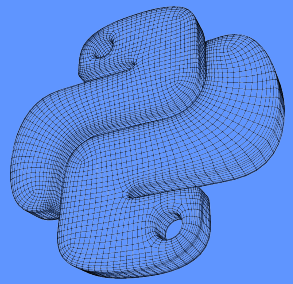
Це досить радикальний варіант і потрібно перевірити, чи покращується якість моделі після видалення змінної.

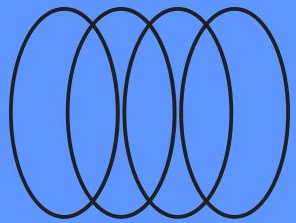


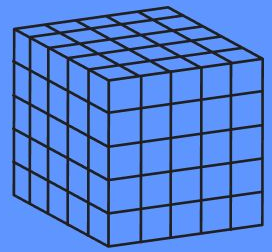
Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	N/A	80%
2	Lite	N/A	70%
3	Fast+	167	10%
4	Fast+	N/A	80%
5	Lite	76	70%
6	Fast+	N/A	10%
7	Fast+	N/A	95%
8	Lite	76	77%
9	Fast+	180	77%

Mobile ID	Mobile Package	Data Limit Usage
1	Fast+	80%
2	Lite	70%
3	Fast+	10%
4	Fast+	80%
5	Lite	70%
6	Fast+	10%
7	Fast+	95%
8	Lite	77%
9	Fast+	77%

Запитання







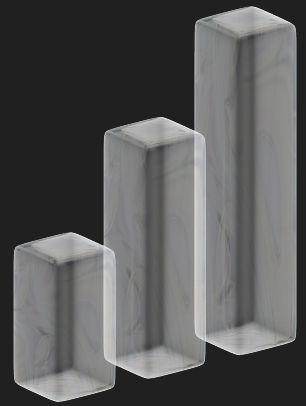
Data imputation

Одномірні методи — для заповнення пропущених значень змінної (стовпчика) використовують лише значення тієї самої змінної (стовпчика).

Мультиваріативні методи дають змогу визначити пропущені значення змінної, переглядаючи дані з інших стовпчиків і оцінюючи кращий прогноз для кожного пропущеного значення на їх основі.

Також розрізняють методи для:

- ➔ неперервних змінних (continuous variables)
- ➔ категоріальних змінних (categorical variables)



Data imputation. Continuous variables

➔ Mean, Median and Mode

Викривлення початкової дисперсії та коваріації з іншими змінними в наборі даних є двома основними недоліками цього методу.

Mean (Download Speed) = 130

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	N/A	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	N/A	95%
8	Lite	76	77%
9	Fast+	180	95%

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	130	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	130	95%
8	Lite	76	77%
9	Fast+	180	95%

Median (Download Speed) = 155

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	N/A	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	N/A	95%
8	Lite	76	77%
9	Fast+	180	95%

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	155	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	155	95%
8	Lite	76	77%
9	Fast+	180	95%

Mode (Download Speed) = 200

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	200	80%
2	Lite	100	70%
3	Fast+	200	10%
4	Fast+	N/A	80%
5	Lite	50	70%
6	Fast+	200	10%
7	Fast+	N/A	95%
8	Lite	200	77%
9	Fast+	180	95%

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	200	80%
2	Lite	100	70%
3	Fast+	200	10%
4	Fast+	200	80%
5	Lite	50	70%
6	Fast+	200	10%
7	Fast+	200	95%
8	Lite	200	77%
9	Fast+	180	95%

Data imputation. Continuous variables

➔ Random Sampling Imputation

За принципом цей метод подібний до заповнення середнім/медіаною, оскільки спрямований на збереження статистичних параметрів вихідної змінної, для якої відсутні дані. Випадкова вибірка складається із взяття випадкового спостереження з пулу доступних спостережень і використання цього випадково вилученого значення для заповнення NA. У випадковій вибірці беруть стільки випадкових спостережень, скільки пропущених значень є у змінній. Врахування випадкової вибірки припускає, що дані відсутні повністю випадковим чином (MCAR). Якщо це так, має сенс замінити відсутні значення значеннями, отриманими з початкового розподілу змінних.

Data imputation. Continuous variables

➔ Arbitrary Value Imputation

В ідеалі довільне значення має відрізнятися від медіани / середнього значення / моди та має бути не в межах нормальних значень змінної. Зазвичай використовують довільні значення 0, 999, -999 (або інші комбінації 9) або -1 (якщо розподіл позитивний). Такий підхід досить добре працює для числових характеристик переважно позитивного значення та для моделей на основі дерев (MAR).

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	N/A	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	N/A	95%
8	Lite	76	77%
9	Fast+	180	95%

Arbitrary value 999

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	999	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	999	95%
8	Lite	76	77%
9	Fast+	180	95%

Data imputation. Continuous variables

➔ Adding a variable to capture NA

Не застосовують у випадку MCAR. В інших випадках ми можемо відтворити важливість відсутності, створивши додаткову змінну, яка вказує, чи були дані відсутні для цього спостереження (1) чи ні (0). Додаткова змінна є бінарною: вона приймає лише значення 0 і 1, причому 0 вказує на наявність значення для цього спостереження, а 1 вказує на те, що значення відсутнє. А самі пропущені значення зазвичай заповнюються середнім/медіаною.

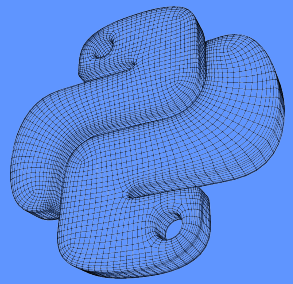
Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	200	80%
2	Lite	100	70%
3	Fast+	200	10%
4	Fast+	N/A	80%
5	Lite	50	70%
6	Fast+	200	10%
7	Fast+	N/A	95%
8	Lite	200	77%
9	Fast+	180	95%

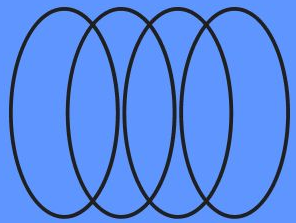
Median

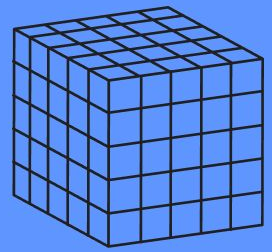
New Feature

Mobile ID	Mobile Package	Download Speed	DL Speed Missing	Data Limit Usage
1	Fast+	200	0	80%
2	Lite	100	0	70%
3	Fast+	200	0	10%
4	Fast+	200	1	80%
5	Lite	50	0	70%
6	Fast+	200	0	10%
7	Fast+	200	1	95%
8	Lite	200	0	77%
9	Fast+	180	0	95%

Запитання







Data imputation. Categorical variables

➔ Adding a category to capture NA

Цей метод полягає в заповненні пропущених значень додатковою міткою або категорією змінної. Усі відсутні спостереження згруповано в новоствореному ярлику Missing. Ця категорія не робить ніяких припущень про реальне значення змінної. Але цей метод добре працює, коли кількість відсутніх даних велика.

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	N/A	99	70%
3	Fast+	167	10%
4	Fast+	90	80%
5	Lite	76	70%
6	N/A	155	10%
7	Fast+	200	95%
8	Lite	76	77%
9	N/A	180	95%

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Missing	99	70%
3	Fast+	167	10%
4	Fast+	90	80%
5	Lite	76	70%
6	Missing	155	10%
7	Fast+	200	95%
8	Lite	76	77%
9	Missing	180	95%

Data imputation. Categorical variables

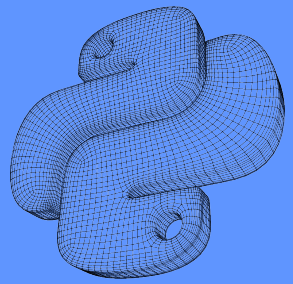
➔ Frequent category imputation

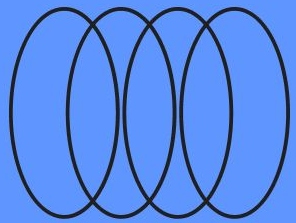
Цей спосіб — аналог заповнення модою і полягає в тому, що пропущені значення заповнюють значеннями, які найчастіше трапляються в цьому стовпчику.

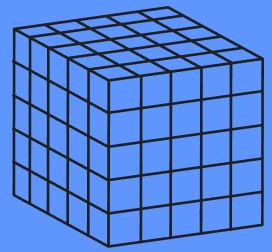
Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	N/A	99	70%
3	Fast+	167	10%
4	Fast+	90	80%
5	Lite	76	70%
6	N/A	155	10%
7	Fast+	200	95%
8	Lite	76	77%
9	N/A	180	95%

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Fast+	99	70%
3	Fast+	167	10%
4	Fast+	90	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	200	95%
8	Lite	76	77%
9	Fast+	180	95%

Запитання



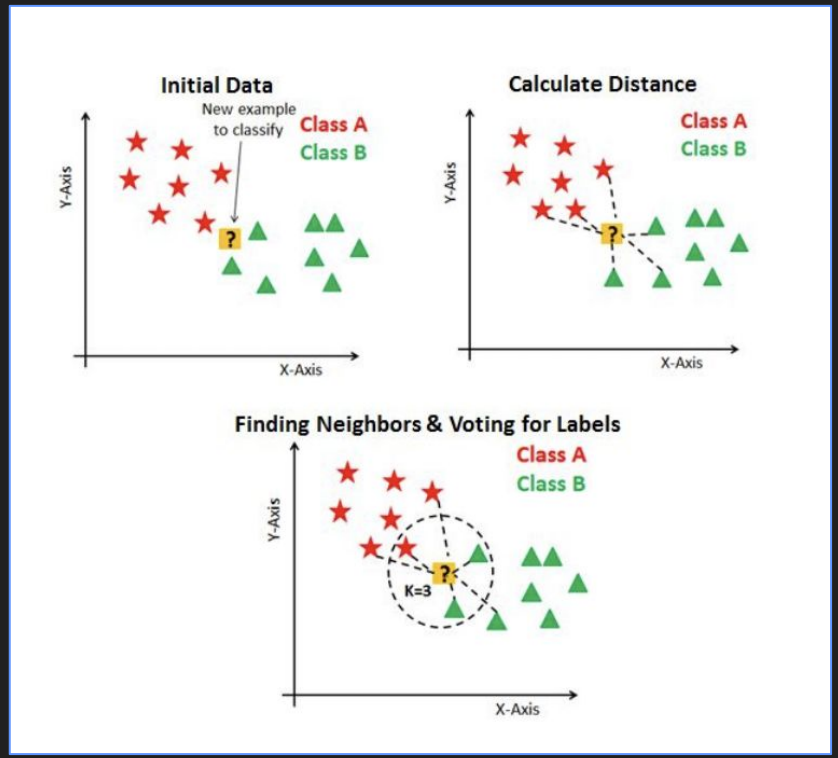




Data imputation. kNN (k Nearest Neighbour)

Модель kNN — простий алгоритм класифікації з урахуванням схожості даних. Під час прогнозування відсутніх значень це може бути корисним для знаходження найближчих k сусідів до спостереження з відсутніми даними та наступного заповнення їх на основі непропущених значень на околі.

*Зручно виконувати заповнення методом kNN, використовуючи клас бібліотеки *sklearn impute.KNNImputer()*.

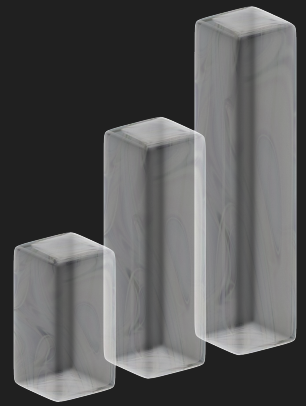


Data imputation. kNN (k Nearest Neighbour)

+ часто цей метод може давати більш точні результати порівняно із заповненням константою, але все залежить від даних.

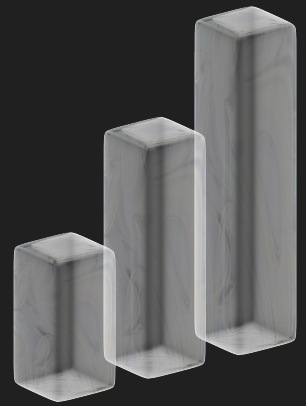
- потребує багато обчислювальних ресурсів; KNN працює, зберігаючи у пам'яті весь навчальний набір даних

- kNN досить чутливий до викидів у даних



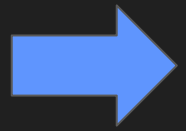
Multivariate Imputation by Chained Equation (MICE)

MICE спочатку обчислює середнє значення кожного стовпчика, де є пропущене значення, і використовує середнє значення як заповнювач. Потім він запускає серію регресійних моделей (ланцюжкових рівнянь), щоб послідовно врахувати кожне відсутнє значення. Як і в будь-якій моделі регресії, MICE використовує матрицю ознак і цільову змінну для навчання, і в цьому випадку цільовою змінною є стовпчик з відсутніми значеннями. MICE прогнозує та оновлює відсутні значення в цільових стовпчиках. Ітеративно MICE повторює цей процес кілька разів, постійно змінюючи заповнювачі змінних прогнозами з попередньої ітерації. Зрештою він досягає надійної оцінки.



MICE. Example

AGE	EXPERIENCE	SALARY (K)	PERSONAL LOAN
25	1	50	1
27	3	70	1
29	5	80	0
31	7	90	0
33	9	100	1
35	11	130	0



AGE	EXPERIENCE	SALARY (K)
25		50
27	3	
29	5	80
31	7	90
33	9	100
	11	130

MICE. Example

1. Заповнимо пропущені значення середнім

AGE	EXPERIENCE	SALARY (K)
25	7	50
27	3	90
29	5	80
31	7	90
33	9	100
29	11	130

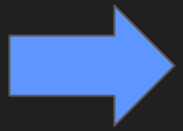
MICE. Example

2. Використовуючи дві колонки передбачимо значення третьої за допомогою, до прикладу, лінійної регресії. Потім виконаємо ті самі дії з другою і третьою колонками

Train set

AGE	EXPERIENCE	SALARY (K)
25	7	50
27	3	90
29	5	80
31	7	90
33	9	100
	11	130

Хочемо передбачити це значення



А тепер хочемо передбачити це значення

AGE	EXPERIENCE	SALARY (K)
25		50
27	3	90
29	5	80
31	7	90
33	9	100
34.99	11	130

MICE. Example

Таким чином заповнили значення в усіх трьох колонках

AGE	EXPERIENCE	SALARY (K)
25	0.98	50
27	3	70
29	5	80
31	7	90
33	9	100
34.99	11	130

MICE. Example

3. Пулінг. Знаходимо різницю між нашим початковим заповненням і тим, яке отримали після застосування регресій. Наша ціль — зменшити цю різницю до нуля. Для цього потрібно зробити багато ітерацій кроків 2–3

AGE	EXPERIENCE	SALARY (K)
25	7	50
27	3	90
29	5	80
31	7	90
33	9	100
29	11	130

minus

AGE	EXPERIENCE	SALARY (K)
25	0.98	50
27	3	70
29	5	80
31	7	90
33	9	100
34.99	11	130

AGE	EXPERIENCE	SALARY (K)
25	6.02	50
27	3	20
29	5	80
31	7	90
33	9	100
-5.99	11	130

MICE. Example

Технічно алгоритм завершує роботу, коли різниця між двома наборами даних буде близька до нуля (дуже маленькі значення різниці). Також можна обмежувати максимальну кількість ітерацій

Iteration 2

AGE	EXPERIENCE	SALARY (K)
25	0.98	50
27	3	70
29	5	80
31	7	90
33	9	100
34.99	11	130

First Dataset

After all imputations

AGE	EXPERIENCE	SALARY (K)
25	0.975	50
27	3	70
29	5	80
31	7	90
33	9	100
34.95	11	130

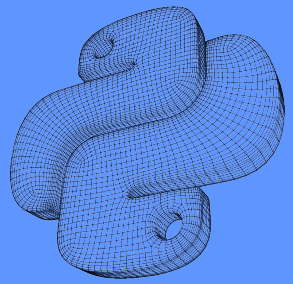
Second Dataset

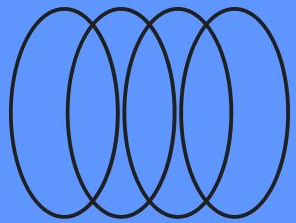
After Second - First

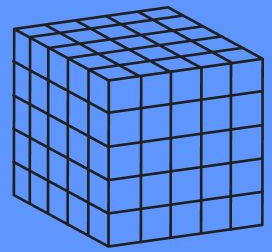
AGE	EXPERIENCE	SALARY (K)
25	0.005	50
27	3	0
29	5	80
31	7	90
33	9	100
0.004	11	130

Difference Matrix

Запитання







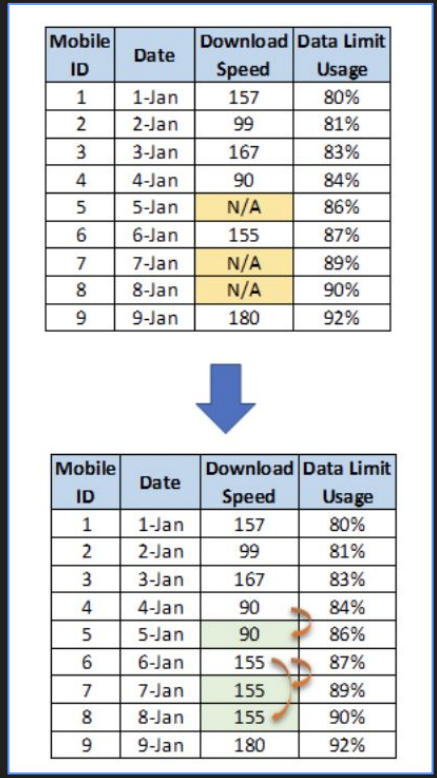
Missing values in time series

Часовий ряд (time series) — це ряд точок даних, проіндексованих (або перелічених, або відкладених на графіку) в хронологічному порядку. Найчастіше часовий ряд є послідовністю, взятою на рівновіддалених точках у часі, які йдуть одна за одною.

Missing values in time series

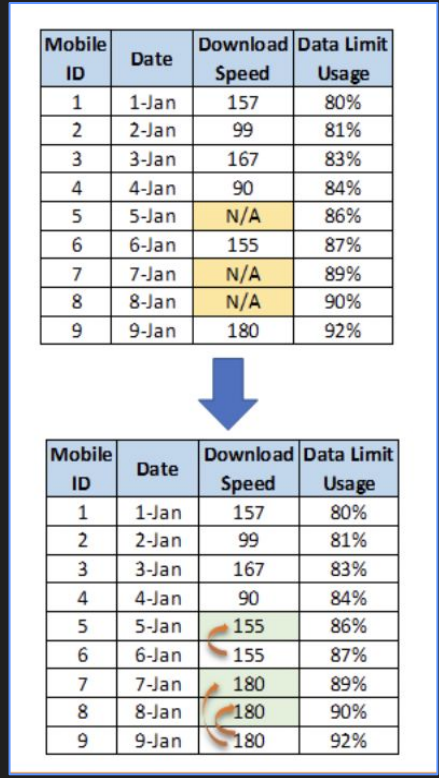
➔ Last Observation Carried Forward (LOCF) & Next Observation Carried Backward (NOCB):

щоразу, коли значення відсутнє, воно замінюється останнім спостережуваним значенням (LOCF), NOCB працює навпаки, беручи перше спостереження після відсутнього значення та переносячи його назад



Mobile ID	Date	Download Speed	Data Limit Usage
1	1-Jan	157	80%
2	2-Jan	99	81%
3	3-Jan	167	83%
4	4-Jan	90	84%
5	5-Jan	N/A	86%
6	6-Jan	155	87%
7	7-Jan	N/A	89%
8	8-Jan	N/A	90%
9	9-Jan	180	92%

Mobile ID	Date	Download Speed	Data Limit Usage
1	1-Jan	157	80%
2	2-Jan	99	81%
3	3-Jan	167	83%
4	4-Jan	90	84%
5	5-Jan	90	86%
6	6-Jan	155	87%
7	7-Jan	155	89%
8	8-Jan	155	90%
9	9-Jan	180	92%



Mobile ID	Date	Download Speed	Data Limit Usage
1	1-Jan	157	80%
2	2-Jan	99	81%
3	3-Jan	167	83%
4	4-Jan	90	84%
5	5-Jan	N/A	86%
6	6-Jan	155	87%
7	7-Jan	N/A	89%
8	8-Jan	N/A	90%
9	9-Jan	180	92%

Mobile ID	Date	Download Speed	Data Limit Usage
1	1-Jan	157	80%
2	2-Jan	99	81%
3	3-Jan	167	83%
4	4-Jan	90	84%
5	5-Jan	155	86%
6	6-Jan	155	87%
7	7-Jan	180	89%
8	8-Jan	180	90%
9	9-Jan	180	92%

Missing values in time series

- **Rolling Statistical**
  Статистичні методи можна використати для визначення відсутніх значень шляхом агрегування попередніх непропущених значень.

- **Moving Average:**
  $P_t = (P_{t-1} + P_{t-2} + P_{t-3} + \dots P_{t-n}) / n$

- **Weighted Moving Average:**
  $P_t = (N*P_{t-1} + (N-1)*P_{t-2} + (n-2)*P_{t-3} \dots 1*P_{t-n}) / (N*(N+1)/2)$

Missing values in time serie

Interpolation

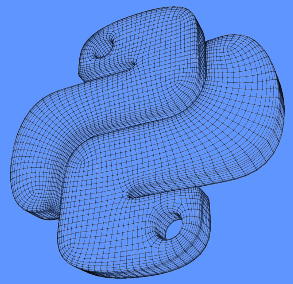
Методи інтерполяції оцінюють відсутні значення, припускаючи зв’язок у діапазоні точок даних. У ковзних статистичних методах, де враховувалися лише попередні значення для імпутації відсутніх значень, метод інтерполяції оцінює з використанням минулих і майбутніх відомих точок даних.

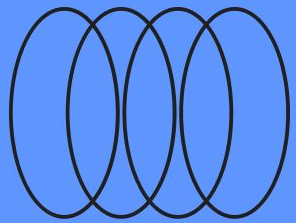
- ➔ linear: припускає лінійне співвідношення ч/б діапазону точок даних
- ➔ spline: оцінює значення, які мінімізують загальну кривизну, таким чином отримуючи гладку поверхню, що проходить через точки введення
- ➔ time: оцінює відсутні значення, зосереджуючись більше на найближчих точках, ніж на віддалених.

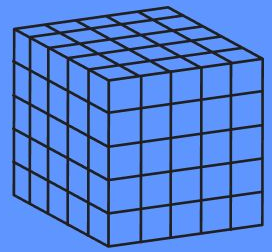
*Pandas Interpolate

The Combination of Seasonal Adjustment and other methods

Запитання







Live coding / Практика

Q&A