

Projekt med databehandling, dataopresning med Python-Panda-dataframe

Det har repeteret og udvidet min viden om de stærke, hurtige beregningsmetoder med panda dataframe, med god sparring med Gemini AI (stærk til levering af effektive, hurtige kode-stykker, som jeg ihærdigt prøver at forstå -som regel får jeg en god forståelse ved at afprøve, tweeke kodelinjerne.

Jeg er endt ud med et program https://bjarnephys.com/Python/data_cleaning.py [du kan download programmet og afvikle hvis du har et Python-miljø]

C:\Users\bh\AppData\Local\anaconda3\envs\openai2\python.exe data_cleaning.py dirty_data.csv sep=; min Python interpreter , dirty_data.csv er datafilen, her vælges ; som value separator]
der kan indlæse cvs-datafiler, rette mange fejl, opliste problemrækker, der ikke umiddelbart kan afhjælpes, til senere manual inspektion i Excel, jeg får lavet statistik over fejl-forekomster i de forskellige datakolonner. Brugeren kan angive kolonner hvor Null/tomme felter er helt acceptabelt, og derfor ikke annoteres. Brugeren kan vælge få fjernet rækker med fejl, især med tomme felter, men kan også beholde disse rækker, hvor der ud fra hver så oprettes 'Check manually' note, som kan ses i notes-kolonne, når filen åbnes i Excel

Har lavet lille datafilen, dirty_data.csv, med mange typiske fejl:

```
1 Text_column;price;CustomerID;datetime
2 ;0;;2022-01-01
3 'leading; space';;1;
4 trailing space ;2.0;2;22-01-02
5 spaces ;3,0;3;2022-01-03
6 spaces ;3,0;3;2022-01-03
7 many spaces ;5.5;5;2022-01-04
8 many spaces;6.5;6;2022-01-05
9
```

får automatisk udskrevet injer der giver problemer under indlæsningen:

c:\users\bh\documents\programmering\python\data_cleaning.py:492(dette pgm): ParserWarning: Skipping line 5: Expected 1 fields in line 5, saw 2, **retter tallet 3,0 til 3.0**

c:\users\bh\documents\programmering\python\data_cleaning.py:492: ParserWarning: Skipping line 6: Expected 1 fields in line 6, saw 2, **retter 3,0 til 3.0**

c:\users\bh\documents\programmering\python\data_cleaning.py:492: ParserWarning: Skipping line 3: Expected 4 fields in line 3, saw 5,

pgm bør fortolke 'leading; space' som ét felt med quotechar="'", men det sker ikke, pgm ser 5 felter
retter **leading; space;;1;** [produkt-deskription med ; tegn] til **leading space;;1;**

Info about input dataset

No of rows/datalines: 7, nu er alle indlæst

Alle rækker

	Text_column	price	CustomerID	datetime
0	NaN	0.0	NaN	2022-01-01
1	leading space	NaN	1.0	NaN
2	trailing space	2.0	2.0	22-01-02
3	\t\tspaces	3.0	3.0	2022-01-03
4	spaces	3.0	3.0	2022-01
5	many spaces	5.5	5.0	2022-01-04
6	many spaces	6.5	6.0	2022-01-05

Currently using value-separator: ','

NaN =no available value

Output-filen kan åbnes uden problemer i Excel:

	A	B	C	D	E
1	text_column	price	customer_id	datetime	notes
2		0		01-01-2022	Check manually
3	leading_space		1		Check manually
4	trailing space	2	2	02-01-2022	
5	spaces	3	3	03-01-2022	
6	many spaces	5.5	5	04-01-2022	
7	many spaces	6.5	6	05-01-2022	
8					

Ny analyse med stor datafil med 121000 linjer, tidstiming er udført til sub-processer

The calculations takes less than 2s

2 first lines in datafile sales.csv :

SalesOrderLineKey,ResellerKey,CustomerKey,ProductKey,OrderDateKey,DueDateKey,ShipDateKey,SalesTerritoryKey,Order Quantity,Unit Price,Extended Amount,Product Standard Cost,Total Product Cost,Sales Amount,Unit Price Discount Pct,Profit Amount

,203,-1,333,20180718,20180728,,4,2,"469,794","939,588","486,7066","973,4132","939,588",0,"-33,8252"

You can see first value, SalesOrderLineKey is blank(I have deleted this to test)

All lines can be read(of course with some problems, which are now treated:

Currently using value-separator: , ->

Discovered columns: Index(['SalesOrderLineKey', 'ResellerKey', 'CustomerKey', 'ProductKey', 'OrderDateKey', 'DueDateKey', 'ShipDateKey', 'SalesTerritoryKey', 'Order Quantity', 'Unit Price', 'Extended Amount', 'Product Standard Cost', 'Total Product Cost', 'Sales Amount', 'Unit Price Discount Pct', 'Profit Amount'], dtype='object')

Number of columns: 16 -ok, all columns recognized, we are using the correct separator!

-user can try ie ‘;’ but ‘,’ works fine!

Info about input dataset

No of rows/data-lines: 121253

First 5 rows:

	SalesOrderLineKey	ResellerKey	CustomerKey	ProductKey	OrderDateKey	DueDateKey	ShipDateKey
\							
0	NaN	203.0	-1	333	20180718	20180728	NaN
1	46638002.0	NaN	-1	325	20180718	20180728	20180725.0
2	46642010.0	4.0	-1	321	20180720	20180730	NaN
3	46642011.0	4.0	-1	323	20180720	20180730	20180727.0
4	46642012.0	4.0	-1	325	20180720	20180730	20180727.0

	SalesTerritoryKey	Order Quantity	Unit Price	Extended Amount	Product Standard Cost	\
0	4	2	469,794	939,588	486,7066	
1	4	2	469,794	939,588	486,7066	
2	4	2	469,794	939,588	486,7066	
3	4	2	469,794	939,588	486,7066	
4	4	2	469,794	939,588	486,7066	

	Total Product Cost	Sales Amount	Unit Price Discount Pct	Profit Amount
0	973,4132	939,588	0	-33,8252
1	973,4132	939,588	0	-33,8252
2	973,4132	939,588	0	-33,8252
3	973,4132	939,588	0	-33,8252
4	973,4132	939,588	0	-33,8252

NaN means no value, empty field

Choose another separator (e.g., ';' or ',') - press <Enter> to accept current data read: <Enter> is pressed -> pgm continues, I use ,

quotechar: "

Choose another quotechar (e.g., " or ') - press <Enter> to accept current data read: <Enter> -> pgm continues

" is my choice, as some numbers are written "469,794" -> all numbers seems to be recognized ok, in the first 5 rows !

strange numbers, I guess "469,794" is 469794 ?!

Please write the names of columns to be discarded in data input, comma separated, <Enter> for no column to discard: <Enter> -> pgm continues

(i use all columns) ->

Warning: The following columns were not found in the DataFrame and could not be dropped: []

No valid columns were specified for dropping.

Please write the datetime format. Examples:

- For dates like 25.12.2023: %d.%m.%Y

- For dates like 2023-12-25: %Y-%m-%d -the default choice, press <Enter> for this

- For timestamps like 25/Dec/2023 14:30:00: %d/%b/%Y %H:%M:%S

- For dates like 20180725: %Y%m%d

Enter format: %Y%m%d

->

Datetimeformat to be used: ['%Y%m%d', '%y%m%d']

(%Y stands for year with 4 ciphers, %y for year with 2)

Attempting to guess column types based on the first 10 rows:

Column 'SalesOrderLineKey': Guessed as integer

Column 'ResellerKey': Guessed as integer

Column 'CustomerKey': Guessed as integer

Column 'ProductKey': Guessed as integer

Column 'SalesTerritoryKey': Guessed as integer

Column 'Order Quantity': Guessed as integer

Column 'Unit Price': Guessed as numeric (float/mixed numeric)

Column 'Extended Amount': Guessed as numeric (float/mixed numeric)

Column 'Product Standard Cost': Guessed as numeric (float/mixed numeric)

Column 'Total Product Cost': Guessed as numeric (float/mixed numeric)

Column 'Sales Amount': Guessed as numeric (float/mixed numeric)

Column 'Unit Price Discount Pct': Guessed as integer

Column 'Profit Amount': Guessed as numeric (float/mixed numeric)

String-columns: [] -> no text-fields, I do not get a test of the trimming process!

Numerical-columns: ['SalesOrderLineKey', 'ResellerKey', 'CustomerKey', 'ProductKey', 'SalesTerritoryKey', 'Order Quantity', 'Unit Price', 'Extended Amount', 'Product Standard Cost', 'Total Product Cost', 'Sales Amount', 'Unit Price Discount Pct', 'Profit Amount']

Datetime-columns: ['OrderDateKey', 'DueDateKey', 'ShipDateKey']

Please write a column name, which can be used as primary key, with unique values; press <Enter> for use a automatic generated column, ID: <Enter> is pressed -> ID-column is generated

>-Column-name for primary key values, with unique values: ID

read data duration: 24229 ms=24.3s -long time because human interaction!

Number of problems in each column, out of 121253 values(rows)

	Column Name	Missing Count	%rows with Error
0	SalesOrderLineKey	1	0.0
1	ResellerKey	1	0.0
2	CustomerKey	0	0.0
3	ProductKey	0	0.0
4	OrderDateKey	0	0.0
5	DueDateKey	0	0.0
6	ShipDateKey	2115	1.7 -many are unfilled,I choose to accept null for this field
7	SalesTerritoryKey	0	0.0
8	Order Quantity	0	0.0
9	Unit Price	0	0.0
10	Extended Amount	0	0.0
11	Product Standard Cost	0	0.0
12	Total Product Cost	0	0.0
13	Sales Amount	0	0.0
14	Unit Price Discount Pct	0	0.0
15	Profit Amount	0	0.0
16	ID	0	0.0

fejlanalyse() duration: 118 ms=0.118s

Please write list of columns where row value can be empty/none, nullable, comma separated, like col1,col2 -press <Enter> for no such columns: **ShipDateKey**
->nullable columns: ['ShipDateKey']

Correcting problems with redundant spaces in text coloumns(trimming), convert string-numbers to numbers -duration: 665 ms

Converting column types to specified types:
Converted column 'SalesOrderLineKey' to int64.
Converted column 'ResellerKey' to int64.
Converted column 'CustomerKey' to int64.
Converted column 'ProductKey' to int64.
Converted column 'OrderDateKey' to datetime
Converted column 'DueDateKey' to datetime
Converted column 'ShipDateKey' to datetime
Converted column 'SalesTerritoryKey' to int64.
Converted column 'Order Quantity' to int64.
Converted column 'Unit Price' to float64.
Converted column 'Extended Amount' to float64.
Converted column 'Product Standard Cost' to float64.
Converted column 'Total Product Cost' to float64.
Converted column 'Sales Amount' to float64.
Converted column 'Unit Price Discount Pct' to int64.

Converted column 'Profit Amount' to float64.

Converting types -duration: 796 ms

Number of duplicate rows found (excluding the first occurrence): 0

(Only columns in subset ['SalesOrderLineKey', 'ResellerKey', 'CustomerKey', 'ProductKey', 'OrderDateKey', 'DueDateKey', 'ShipDateKey', 'SalesTerritoryKey', 'Order Quantity', 'Unit Price', 'Extended Amount', 'Product Standard Cost', 'Total Product Cost', 'Sales Amount', 'Unit Price Discount Pct', 'Profit Amount'] are considered)

Number of rows before removing duplicates: 121253

Number of rows after removing duplicates: 121253

Search for and removing duplicated rows -duration: 222 ms

Problem analysis after correcting problems with redundant spaces in strings, with number-decimals, duplicated rows:

Number of problems in each column, out of 121253 values(rows)

(Nullable columns are not included, considered)

	Column Name	Missing Count	%rows with Error
0	SalesOrderLineKey	1	0.0
1	ResellerKey	1	0.0
2	CustomerKey	0	0.0
3	ProductKey	0	0.0
4	OrderDateKey	0	0.0
5	DueDateKey	0	0.0
6	SalesTerritoryKey	0	0.0
7	Order Quantity	0	0.0
8	Unit Price	0	0.0
9	Extended Amount	0	0.0
10	Product Standard Cost	0	0.0
11	Total Product Cost	0	0.0
12	Sales Amount	0	0.0
13	Unit Price Discount Pct	0	0.0
14	Profit Amount	0	0.0
15	ID	0	0.0

autogenerated unique values -the column with

Number of rows in the dataset: 121253

Number of problem-rows: 2

List of problem-rows, max 20 are shown:

SalesOrderLineKey ResellerKey CustomerKey ProductKey OrderDateKey DueDateKey
ShipDateKey \

0	<NA>	203	-1	333	2018-07-18	2018-07-28	NaT
1	46638002	<NA>	-1	325	2018-07-18	2018-07-28	2018-07-25

SalesTerritoryKey Order Quantity Unit Price Extended Amount Product Standard Cost \

0	4	2	469.794	939.588	486.7066
1	4	2	469.794	939.588	486.7066

	Total Product Cost	Sales Amount	Unit Price	Discount Pct	Profit Amount	ID
0	973.4132	939.588	0	-33.8252	0	
1	973.4132	939.588	0	-33.8252	1	

<NA>, NaN, NaT means no value, empty field

it seems ok, I have removed these 3 values in the first to rows in the datafile to test!

Dropping problem rows. You can choose to keep these for later manual inspection in Excel,
(these rows will get a 'Check manually' note in a notes-column, when opened in Excel)

-Press <Enter> to keep all problem rows

-Write none to delete all these rows: <Enter> is pressed -> pgm continues, keep all problem rows!

First 5 rows after column corrections, dropping problem rows, dropping row duplets:

	SalesOrderLineKey	ResellerKey	CustomerKey	ProductKey	OrderDateKey	DueDateKey	ShipDateKey \
0	<NA>	203	-1	333	2018-07-18	2018-07-28	NaT
1	46638002	<NA>	-1	325	2018-07-18	2018-07-28	2018-07-25
2	46642010	4	-1	321	2018-07-20	2018-07-30	NaT
3	46642011	4	-1	323	2018-07-20	2018-07-30	2018-07-27
4	46642012	4	-1	325	2018-07-20	2018-07-30	2018-07-27

	SalesTerritoryKey	Order	Quantity	Unit Price	Extended Amount	Product Standard Cost \
0	4	2	469.794	939.588	486.7066	
1	4	2	469.794	939.588	486.7066	
2	4	2	469.794	939.588	486.7066	
3	4	2	469.794	939.588	486.7066	
4	4	2	469.794	939.588	486.7066	

	Total Product Cost	Sales Amount	Unit Price	Discount Pct	Profit Amount	ID	notes
0	973.4132	939.588	0	-33.8252	0	0	Check manually
1	973.4132	939.588	0	-33.8252	1	1	Check manually
2	973.4132	939.588	0	-33.8252	2	2	
3	973.4132	939.588	0	-33.8252	3	3	
4	973.4132	939.588	0	-33.8252	4	4	

Notice, a notes-column has been added automaticly, usefull in a manual data-inspection in Excel

-----Saving file: -----

Status of problem(s) in the dataset, to be saved:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 121253 entries, 0 to 121252
```

Data columns (total 18 columns):

#	Column	Non-Null Count		Dtype
---	-----	-----	-----	-----
0	SalesOrderLineKey	121252	non-null	Int64
1	ResellerKey	121252	non-null	Int64
2	CustomerKey	121253	non-null	Int64
3	ProductKey	121253	non-null	Int64
4	OrderDateKey	121253	non-null	datetime64[ns]
5	DueDateKey	121253	non-null	datetime64[ns]
6	ShipDateKey	119138	non-null	datetime64[ns]
7	SalesTerritoryKey	121253	non-null	Int64
8	Order Quantity	121253	non-null	Int64
9	Unit Price	121253	non-null	float64
10	Extended Amount	121253	non-null	float64
11	Product Standard Cost	121253	non-null	float64
12	Total Product Cost	121253	non-null	float64
13	Sales Amount	121253	non-null	float64
14	Unit Price Discount Pct	121253	non-null	Int64
15	Profit Amount	121253	non-null	float64
16	ID	121253	non-null	int64
17	notes	121253	non-null	object

dtypes: Int64(7), datetime64[ns](3), float64(6), int64(1), object(1)

memory usage: 17.5+ MB

None

Number of problems in each column, out of 121253 values(rows)
(Nullable columns are not included, considered)

	Column Name	Missing Count	%rows with Error
0	SalesOrderLineKey	1	0.0
1	ResellerKey	1	0.0
2	CustomerKey	0	0.0
3	ProductKey	0	0.0
4	OrderDateKey	0	0.0
5	DueDateKey	0	0.0
6	SalesTerritoryKey	0	0.0
7	Order Quantity	0	0.0
8	Unit Price	0	0.0
9	Extended Amount	0	0.0
10	Product Standard Cost	0	0.0
11	Total Product Cost	0	0.0
12	Sales Amount	0	0.0
13	Unit Price Discount Pct	0	0.0
14	Profit Amount	0	0.0
15	ID	0	0.0

Number of rows in the dataset: 121253

Number of problem-rows: 2

List of problem-rows, max 20 are shown:

	SalesOrderLineKey	ResellerKey	CustomerKey	ProductKey	OrderDateKey	DueDateKey	ShipDateKey	\
0	<NA>	203	-1	333	2018-07-18	2018-07-28	NaT	
1	46638002	<NA>	-1	325	2018-07-18	2018-07-28	2018-07-25	

	SalesTerritoryKey	Order Quantity	Unit Price	Extended Amount	Product Standard Cost	\
0	4	2	469.794	939.588	486.7066	
1	4	2	469.794	939.588	486.7066	

	Total Product Cost	Sales Amount	Unit Price	Discount Pct	Profit Amount	ID	notes
0	973.4132	939.588		0	-33.8252	0	Check manually
1	973.4132	939.588		0	-33.8252	1	Check manually

First 5 rows:

	SalesOrderLineKey	ResellerKey	CustomerKey	ProductKey	OrderDateKey	DueDateKey	ShipDateKey	\
0	<NA>	203	-1	333	2018-07-18	2018-07-28	NaT	
1	46638002	<NA>	-1	325	2018-07-18	2018-07-28	2018-07-25	
2	46642010	4	-1	321	2018-07-20	2018-07-30	NaT	
3	46642011	4	-1	323	2018-07-20	2018-07-30	2018-07-27	
4	46642012	4	-1	325	2018-07-20	2018-07-30	2018-07-27	

	SalesTerritoryKey	Order Quantity	Unit Price	Extended Amount	Product Standard Cost	\
0	4	2	469.794	939.588	486.7066	
1	4	2	469.794	939.588	486.7066	
2	4	2	469.794	939.588	486.7066	
3	4	2	469.794	939.588	486.7066	
4	4	2	469.794	939.588	486.7066	

	Total Product Cost	Sales Amount	Unit Price	Discount Pct	Profit Amount	ID	notes
0	973.4132	939.588		0	-33.8252	0	Check manually
1	973.4132	939.588		0	-33.8252	1	Check manually
2	973.4132	939.588		0	-33.8252	2	
3	973.4132	939.588		0	-33.8252	3	
4	973.4132	939.588		0	-33.8252	4	

Strange floating numbers, in the file the first unit price is "469,794" ?!

Notice, a notes-column has been added, to be used in a manual data-inspection in Excel

column with primary field 'ID' has been removed

Column names are corrected to only small letters, - replaced by _

Column Name Mapping (Old -> New):

Column	Old Column Name	New Column Name
0	SalesOrderLineKey	salesorderlinekey
1	ResellerKey	resellerkey
2	CustomerKey	customerkey
3	ProductKey	productkey
4	OrderDateKey	orderdatekey
5	DueDateKey	duedatekey
6	ShipDateKey	shipdatekey
7	SalesTerritoryKey	salesterritorykey
8	Order Quantity	order_quantity
9	Unit Price	unit_price
10	Extended Amount	extended_amount
11	Product Standard Cost	product_standard_cost
12	Total Product Cost	total_product_cost
13	Sales Amount	sales_amount
14	Unit Price Discount Pct	unit_price_discount_pct

```

15      | Profit Amount          | profit_amount
16      | notes                    | notes

```

DataFrame successfully saved to: sales_cleaned.csv

The file open without any problems in Excel, a small part of the 121000 lines:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	salesord	res	cl	pr	orderdate	duedatekey	shipdatekey	sa	ord	unit_p	extended	product_st	total_prod	sales_an	unit	profit_a	notes
2		203	-1	333	18-07-2018	28-07-2018		4	2	469794	939588	4867066	9734132	939588	0	-338252	Check manually
3	46638002		-1	325	18-07-2018	28-07-2018		4	2	469794	939588	4867066	9734132	939588	0	-338252	Check manually
4	46642010	4	-1	321	20-07-2018	30-07-2018		4	2	469794	939588	4867066	9734132	939588	0	-338252	
5	46642011	4	-1	323	20-07-2018	30-07-2018	27-07-2018	4	2	469794	939588	4867066	9734132	939588	0	-338252	
6	46642012	4	-1	325	20-07-2018	30-07-2018	27-07-2018	4	2	469794	939588	4867066	9734132	939588	0	-338252	
7	46660022	166	-1	321	26-07-2018	05-08-2018	02-08-2018	4	2	469794	939588	4867066	9734132	939588	0	-338252	
8	46660029	166	-1	331	26-07-2018	05-08-2018	02-08-2018	4	2	469794	939588	4867066	9734132	939588	0	-338252	
9	46663009	292	-1	323	27-07-2018	06-08-2018	03-08-2018	4	2	469794	939588	4867066	9734132	939588	0	-338252	
10	46668005	231	-1	325	28-07-2018	08-08-2018	05-08-2018	4	2	469794	939588	4867066	9734132	939588	0	-338252	