



Data Cleansing Strategies on Data Sets Become Data Science

Sardjono^{1*}, R Yadi Rakhman Alamsyah², Marwondo³, Elia Setiana⁴

^{1,2,3,4} *Department of Informatic, Faculty of Technology and Informatic, Universitas Informatika dan Bisnis Indonesia, Bandung, Indonesia.*

** Corresponding author email: sardjono@unibi.ac.id*

Abstract

The digital era very grows up with the increasing using of smartphone and many organization or companies was implemented of a system to support their business. That is who will increase the volume of usage and dissemination of data, neither through open nor closed internet networks. Because there is the need to process large data and how to get it from different store resource, so requirement strategy to process the data according to the rule of good, effective and efficient in activity data cleansing until the data set can be use as mature and very useful information for their business purpose. By using the R languaged who can process large data and has data complexity for the data loaded from different storage resource can be done as well as. To using R languaged maximally, so we have to a basic skill that needed to process the data set which will be used to be data scient for organizations or companies by good data cleansing techniques. In this research on Data Cleansing Strategies on data set owned by organizations, will describe the correct step by step to obtaining data that very useful to be uses as data science for organization so by the data that generated after the data cleansing process is very meaningful and useful for making decisions, other than that this research give basic overview and guide to the beginner all data scientists by doing data cleansing in the way stages and also provides a way to analyze from the result of execution some functions used.

Keywords: Data, Data Scient, Data Cleansing, Data Set, data Profiling, R Languaged, factor, Step-by Step, function, Library, Data Enrichment.

1. Introduction

Today, our world filled by data, that is trigeering by increasingly used smarphone who massive and extended, in addition to utilizing system computerizing an organization or company (Wu et al., 2013; Huerta and Jensen, 2017; Brayne 2017; Kowalczyk and Buxmann, 2014; Sivaparthipan et al., 2020). However the data owned a organization that store in computer not convince sufficient for

making the information that needed and beneficial by organization when the data incomprehensive processable (Dietrich, 2015; Whyte et al., 2016).

To be able processing data according properly, effectively and efficiently, that it requires the ability in terms of a specific programming language that is specifically dedicated to processing data or information from many sources in the organization or the internet to become a useful data scientist for organizations or businesses (Endel and Piringer, 2015; Faisal, 2016;)

The R language appears not only as a programming language in general, that it also has a myriad of specific capabilities for processing very large data. However, with the many advantages of the powerful R language, there are obstacles for beginners who will use the R language to process data sets for data cleansing (Kandel et al., 2011; Mailund, 2017; Patil and Hiremath, 2018).

The purpose of this research about the strategy of data cleansing who will be data science for organizational is the first to provide an overview of the initial stages that must be carried out in the data cleansing process and the second, to provide examples of the application of various basic functions in the R language which is often used to process data sets.

The usefulness of this research is to provide insight to prospective data scientists to pay attention the simple thing, but very important in the data cleansing process as well as provide a step-by-step guide that is composed of the basic stages that can be used as recommendations for someone that want using R Language to processing data sets, also provide examples of how to analyze the results of the implementation of the basic functions on R Language used to process data.

2. Research Method

The method on this reasearch to implementation of data cleansing process is literature study, literature books and translation of the output results from the script program that executed by the R language, the next step is how to translate the output of the functions used in R programming so that the get meaning, purpose, and benefits of a function used on data cleansing in the R language programming can be found. To complete the implementation of the proving program so that the output can be analyzed, it is necessary to support other methods that are used to support the implementation of data cleansing, including:

- a. Data Profiling : The first step in the data cleansing process is doing data profiling. Data profiling activities are activities of scientists to identify the patterns of data that exist in data sets and comparing the data that encountered with common data behaviors. By this profiling data activities, we will trace the entire data set to be cleansed.
- b. Utilizing `str()` function : The **`str()`** function is a function that provides information about structural from dataset that will be cleansed. This function will present the information each column of the dataset by a row format for each dataset. The syntax that used in the R Programming Language is **`str(object)`**.
- c. Utilizing `summary()` function : The **`summary()`** function is a function that will be used to obtain information related to the structure of the dataset by providing a brief summary of the data so that will be used as a reference by a scientists to analyzing the contents of the dataset. Syntax the `summary()` function that used by the R language is **`Summary(object)`**.
- d. Modify Data type : By using the **`str()`** and **`summary()`** functions, turns out that there is still a lot of hidden information, in order to get the comprehensive information from these functions, so it

is necessary to change the data type on the dataset, which is the original character data type, so we must change it into a data type factor. To implement this on the R language provides the function of turning characters into factors in the following way:

```
data.pelanggan$Active<- as.factor(data.pelanggan$Active)
```

- e. Applied bpa library : Activities on Profiling data using the summary() function with a change in data type is indeed quite useful for identifying numeric data type on dataset and obtaining the frequency distribution of factor values, but this function still has shortcomings if we use it to identify the correct of the text data pattern according to common practice, in order to gain information better, R language provide a function that carry on this problem. By Basic_pattern_analysis (bpa)function can useful for analyzing text pattern on dataset according to prevalence.
- f. Utilizing regex and grepl function : The technique data filtering using bpa function that in combination with the symbol double equal operator (==) can only be used to recognize unusual text patterns specifically and exactly match the text that will be search for. However, if we are searching text patterns that have certain characters, for example we search the text patterns that contain characters that are not letters, of course we have to using the bps function combination with the symbol == (equal to double) cannot handle it, for that purpose the R language provides a function that named grepl, the grepl function has regular expression constructs which are useful for filtering data based on various text patterns according to the wishes of a scientists

3. Diagnosing Data and Discussion Implemented Data Claensing

Figure 1. the sample of dataset that will be data cleansing process.

Cust.id	Full Name	Address	Birth date	Active	Postal code	Phone number	Total in year
1	Sri Resti Agung	Aranama Perawat IV, No. 2 - Kota D	1 Desember 1964	0	768201	085736296706007	779900
2	Mbak Dien Sukowati	Aranama Perawat IV, No. 1 - Kota D	25 Juli 1974	FALSE	768201	085796817092325	1527400
3	Irfan Putra Wijaya	Aranama Pelajar No. 22 A - Pondok Birma Sakti	25 November 1962	1	768204	089084358708389	525300
4	Cynthia Agas	Aranama Pelajar No. 12 D - Pondok Birma Sakti	10/05/1988	1	768204	08281254686937962	
5	Ibu Ajuar, Susanto	Apartment Cliffton, Lantai 12 No. 5	02/28/1969	1	868205	0324005784325249	851600
6	Mario Setiawan	Jl. Puri Anteri Raya, No. 88 - Kota T	09 Agustus 1972	1	876511	082089111222230	895600
7	Setiawan Mario	Jl. Puri Anteri Raya, No. 88 - Kota T	19 Maret 1950	1	876511	082089111222230	1412900
8	Puzpita Citra	Perum Birmasakti Raya, Blok A No. 10	19 Maret 1950	1	764490	0828379328882114	950200
9	Chandra Rachmat	Perum Birmasakti Raya, Blok A No. 10	12-01-1968	1	764490	0828933761795300	415100
10	Sriwati Wiyanto	Meta Residences, No. 21C	23 November 1962	TRUE	764490	08572595303369	801700
11	Suharno Jamar	Meta Residences, No. 1A	07/25/1974	1	764490	0328151813054888	237400
12	Tiah Peris	Kota T, Jalan Taman Kencaha No. 11132	8 Maret 1955	1	876612	08367465521990	399900
13	Indan Tri Wahyuni	Kompleks Selatan-Selatan, No. 121	09/05/1990	1	521321	082859454241140	495800
14	Sumartono Salim	Kompleks Selatan-Selatan, No. 111	12-12-1950	1	521321	0828894258062822	
15	Safira Hana Sahmanti	Taman Sungsang Langit, Jl. Ubara No. 5	02/20/1979	1	712984	0828681250283826	725600
16	Sidharta Paul	Taman Sungsang Langit, Jl. Timur No. 1	24 Januari 1952	1	712984	0328725681847845	398200
17	Eli NS Alexander	Taman Sungsang Langit, Jl. Selatan No. 12	22 Februari 2000	0	712984	03281413705348545	311000
18	Bapak Sanjaya Priyantoro	Taman Sungsang Langit, Jl. Barat Laut No. 6	26 Agustus 1983	1	712984	082817960005464	1491900
19	Rahmat Chandra	Rumah Susun Eunos, Lantai 2 No. 2	08/26/1985	1	635421	086210781145764	1034600
20	Agnes Rita	Ruko Azalea, No. 3 RT 001/002	21-05-1980	1	511481	0828908681794088	1128000
21	Andreas Santanto	Ruko Almond Marnis, Blok C7/B	07/17/1987	0	511481	0828706674937758	536600
22	Talia Teguh	Purple Lock, No. 88P, Kota Y	1 Desember 1964	1	511482	081902807450191	1451900
23	Djoko Wardoyo, Drs.	Villa Bukit Sagaritama, Blok A2 No. 1	25-11-1962	0	877321	0828487102958165	536000
24	Khairul Nissa	Taman Vivo Indah, Blok AA No. 7	10/23/91	0	712983	082871322137140	1536200
25	Kaka Ari Lima	Taman Vivo Indah, Blok AA No. 7	02/28/1969	0	712983	08350956735507	1518500
26	Lenny Samrini	Jln. Kanguru No. 92, RT 005 - Kota R	12/01/1964	0	868212	0828194199897108	514700
27	Roger Sritati	Jln. G. Auri Mawan Harum Blok G No. 9	01/31/01	1	868213	08888862170254	1474200
28	Lina Ding	Jl. Wilma Terentang Saja, No. A21	29-03-1969	0	868262	0828551768109921	494900
29	Maria Yuniarti	Jl. Wilma Terentang Saja, No. A22	25-11-1962	TRUE		08925174796282	735500
30	Rachmat Chandra	Rusun Kerinci Indah, Lt. 6 No. 1	24-01-1987	1	635429	0828352522514257	538400
31	Ayu	Rusun Kerinci Indah, Lt. 5 No. 6	01/01/01	1	635429	0328320518708137	598000

Figure 1. sample Dataset

If we look dataset like figure 1 we can analyze and get the following information:

- a. Observe “Cust.Id” column is The field that holds code from customer and has typical a unique column and becomes the primary key of customer dataset;
- b. Observe “Full.Name” column is the field that data store name of customer and this field is not unique;
- c. Observe “Address” column is the field that data store Address of customer and not unique;
- d. Observe “Birth.date” column is the field that data store Birth date of customer and not unique;
- e. Observe “Active” column is field that data store state of active or deactivate a customer;
- f. Observe “Postal.code” is field that data store postal code of home customer;
- g. Observe “Phone.number” column is that data store about information phone number of customer that can calling;

Observe “Total.in.year” column is that data store about information sum of purchase in periode a year.

3.1 Diagnosing and analysis on dataset fields

After we know all fields with basic characteristics on dataset which is data stored, so the next step is we carry out a more in-depth analysis of the fields.

a. Diagnosing and analysis on Cust.Id column

Field Cust.Id is very important field of the dataset that we have because Cust.id is a field as the primary key where this field have a very regular pattern, from the dataset shown in Figure 4.1 we can find regular patterns of this column are as follows :

- a. possessed text prefix that definite valueable “C-“
- b. possessed suffix that with five digit number format.

From the two patterns that we found, it can be concluded that the total length of the column is 8 characters / digits

However, if we look closely at certain lines there are patterns that do not match, where the number of digits behind "C-" there are 2 or 4 and also a prefix that is not "C-" as shown in Figure 2. By finding that case in data, so the contents of the Cust.Id field have different length and pattern inconsistencies.

	Cust.Id
1	Cust.Id
2	K-00095
3	C-00022
4	C-00017
5	C-00037
6	C-00108
7	C-00015
8	C-00083
9	C-0047
10	C-00149
11	C-00003
12	c-43
13	C00135
14	c-00050

Figure 2. Field Cust.Id Analysis

b. Diagnosing and analysis on Full.Name column

Observe to the field Full.Name in Figure 1, where this field is the second field in the customer's dataset with some of the contents displaying as shown in Figure 3.

Full.Name
Edi %\$ Alexander
Bapak Sanjaya Priyantoro
Bpk. Rahmat Chandra
Ibu Agnes Rita
Andreas Sutanto
Taka Teguh
Djoko Wardoyo, Drs.
Khairul Nissa
Kaka Ari Lima
Leny Sarmini
Roger Sirait
Lilis Ong

Figure 3. Analysis Full.Name column

Observe the results of the analysis in the field Full.Name, it can be seen that there is a writing nickname for "Mrs. Agnes Rita", "Mr. Sanjaya Priyantoro", and Bpk. Rahmat Chandra", this case may not be a problem for some organizations, but it will be problematic if there is an organization that requires standardization of names, so the name of this nickname must be removed.

c. Diagnosing and analysis on Birth.date Column

Observe again at Figure 1 with the "Birth.date" field, this field is another important column which is usually paired with a name, for the purpose of identifying individuals. A Part of the display of the "Birth.date" field it is can be seen in Figure 4.

Birth.date
1 Desember 1964
25 Juli 1974
23 November 1962
10/03/1988
2/15/1950
09 Agustus 1972
19 Maret 1950
19 Maret 1950
12-01-1968
23 November 1962
07/25/1974

Figure 4. Analysis “Birth.date” Field

From this data, we can find problems in the “Birth.date” column, the case is there are several different calendar patterns, as seen in Figure 4 there is a separation using a minus sign (-), a slash (/) and writing the month's name which is not a number. Writing like this certainly needs to be standardized and improved so that it can be further processed for analysis.

In addition, it is possible to find the year of birth of a customer whose contents are valid, but from a business point of view, the year of birth of the customer may not be logical, for example, there is data whose birth year was in 1879 and we need to fix it.

d. Diagnosing on duplicate data

In addition to non-standard data contents such as what happened in the field “Cust.id”, “Full.name” and “birth.date”, it is possible to find that this customer dataset also has duplicates for the same customer if found from field “Full.name” and “address”, and “birth. Date” of the two rows of data but that are have content different “Cust.Id”.

This case will have a large consequence or impact on the business. If each customer already has a transaction, the customer codes will all be different. And at the time of data analysis, all the transaction data will be separated by three and the total value is never obtained.

4. Using Analysis Tool In R languaged

The data cleansing steps that have been done, such as in points “a” to point d, are manual data profiling activities because we are looking for data abnormalities in plain view and this will be time consuming and tiring for a data scientist. Imagine if the existing dataset has dozens of field or columns that must be analyzed, therefore to produce maximum, effective and efficient data cleansing, we need to use all the data cleansing tools that the R language has. The basic tools that are often used on R language are:

a. Str() and summary() functions

To read and review customer datasets in figure 1, we can use the “str()” or “summary()” function, these two functions have the same use, namely to view the structure of the file contents or customer datasets, the difference is that the “str()” function will present information each column of

the dataset in the format one line at a time, whereas the “summary()” function will provide a brief summary and analyzing of the data. Figure 5 shows the difference in the execution results of the “str()” and “summary()” functions.

```

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
[Icons] Go to file/function Addins
[Icons] Source on Save Run Source
0 Untitled* pola.data.pelanggan data.pelanggan
1 #load library openxlsx
2 library(openxlsx)
3 #membaca dataset pelanggan
4 data.pelanggan <- read.xlsx("D://penelitian/sardjono/pelanggan.xlsx")
5 #menggunakan function str() dan summary()
6 str(data.pelanggan)
7 summary(data.pelanggan)
8
9

B:1 (Top Level) 5 R Script 5
Console Terminal
> str(data.pelanggan)
'data.frame': 155 obs. of  8 variables:
 $ cust.id      : chr  "K-00095" "C-00023" "C-00017" "C-00037" ...
 $ full.name    : chr  "Sri Resti Agung" "Mbak Dian Sukowati" "Irfan Putra Wijaya" "Cynthia Agus" ...
 $ address      : chr  "Asrama Perawat IV, No. 2 - Kota D" "Asrama Perawat IV, No. 1 - Kota D" "Asrama Pelajar No. 22 A - Pondok Bima Sakti" "Asrama Pelajar No. 11 B - Pondok Bima Sakti" ...
 $ birth.date   : chr  "1 Desember 1964" "25 Juli 1974" "23 November 1962" "10/03/1988" ...
 $ active       : chr  "0" "FALSE" "1" "1" ...
 $ postal.code  : chr  "768031" "768031" "768034" ...
 $ phone.number: chr  "085736296760607" "085796817992325" "089984358708389" "+6283155468652762" ...
 $ total.in.year: num  779900 1527400 525300 NA 851600 ...
> summary(data.pelanggan)
  cust.id      Full.Name      Address      Birth.date
Length:155    Length:155    Length:155    Length:155
Class :character Class :character Class :character Class :character
Mode :character Mode :character Mode :character Mode :character

```

Figure 5. difference result of execution str() dan summary()

The Result of execution from “str()” function provides information that the dataset being read is a data frame form that contains 155 objects (data rows) and has 8 variables (columns).

Whereas for the results of the summary() function, it provides more detailed information from the read data set, namely:

- to columns that containing data by numeric type, the summary will provide the information about minimum, maximum, median, mean, and others
- to a column that containing data by character type, the summary() function will display the data type and length.
- To a column that containing data that factor type, the summary() will give information about factor’s values and count of occurrences of frequency.

b. Change data type on “Active” column by factor data type

In the output results of execution using the “summary()”function, there is not much information about the condition of the real existing data, it appears that the “summary()” function only displays character and length information, whereas the information stored in the "Active" column does not provide more meaningful information than it should be.

To get better information from the "Active" column in the data set, we must change the data type of the column to the "factor" data type. Figure 6, is an implementation of how to change a column to the "factor" type .

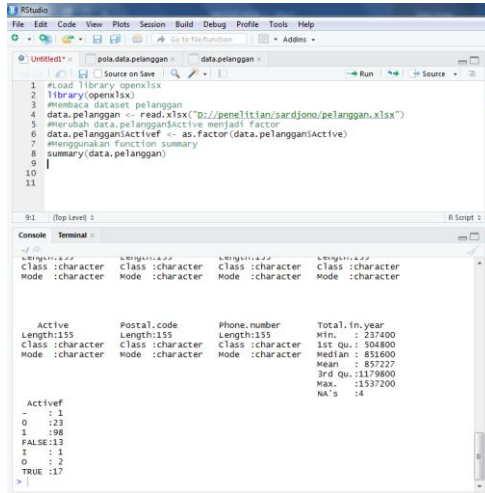


Figure 6. Result of cheng to factor type

There is a difference in the results when the data type "Active" column is changed to "factor", this can be seen from the results of the execution that the active column provides information that there are data with a minus sign (-) that sum of totaling 1, there is 23 row contain 0(zero), there is 98 row data contain 1, containing 13 row data with false, letter I has 1, letter O has 2 and True is 17.

By combining the “summary()” function and changing the data type from the dataset to be "factor", so it will produce more meaningful information for a data scientist. For this purpose, we have to change the other columns to the data type "factor".

Figure 7, is the result of execution for all columns in the dataset (Figure 1) which has been converted into a data type "factor"

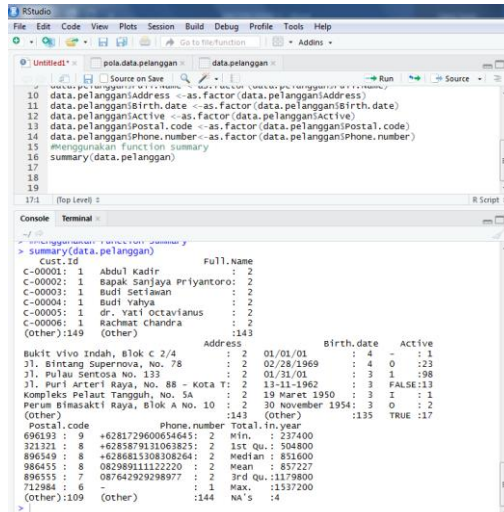
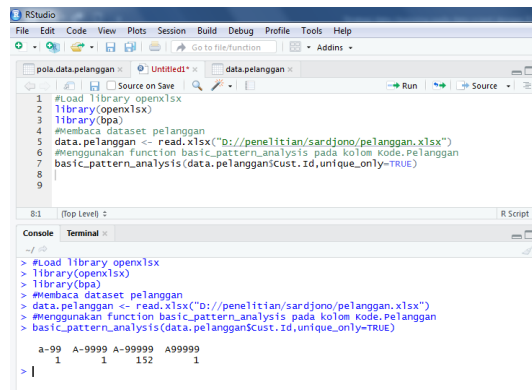


Figure 7. Result of execution changed data type

c. Utilizing library (bpa)

Executing profiling using the “summary()” function by combining changes in data types into "factors" is indeed quite useful for analyzing and identifying data type of factor and numeric in datasets. However, if we are going to identify a data type with a character type or text pattern, then the “summary()” function that combined with the conversion to the factor cannot do it, to overcome this we can use basic_pattern_analysis (bpa) function.

Figure 8 is the result of the implementation of the library (bpa) function which will be used for profiling the "Cust.Id" column.



```

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
[Icons] Source on Save [Icons] Run [Icons] Source
1 #load library openxlsx
2 library(openxlsx)
3 library(bpa)
4 #membaca dataset pelanggan
5 data.pelanggan <- read.xlsx("D://penelitian/sardjono/pelanggan.xlsx")
6 #menggunakan function basic_pattern_analysis pada kolom kode.pelanggan
7 basic_pattern_analysis(data.pelanggan$Cust.Id,unique_only=TRUE)
8
9

8:1 (Top Level) R Script
Console Terminal
> #load library openxlsx
> library(openxlsx)
> library(bpa)
> #membaca dataset pelanggan
> data.pelanggan <- read.xlsx("D://penelitian/sardjono/pelanggan.xlsx")
> #menggunakan function basic_pattern_analysis pada kolom kode.pelanggan
> basic_pattern_analysis(data.pelanggan$Cust.Id,unique_only=TRUE)

a-99 A-9999 A-99999 A99999
1 1 152 1
> |

```

Figure 8. library bpa function impelementing

The results of using the “basic_pattern_analysis()” function which is applied to analyze the Cust.id column provide information that in the column found data of type a-99 are 1, A-9999 is 1, A-99999 is 152 data, and A99999 has 1 data. Thus we can conclude that the Cust.id column has data that does not have the same or consistent pattern, data that is inconsistent. The data found to be anomalous must be given filtering action.

d. Filtering on Data Anomaly pattern

In the profiling process that has been done using basic_pattern_analysys in the Cust.Id column, we find anomaly data, namely "a-99", "A-9999" and "A99999", with this anomaly, the next data cleansing step is how we do filtering. on the anomaly data by taking a data set with an anomaly pattern as data to be corrected.

There are 2 processes that can be carry out for filtering the anomaly data, the first is by using the double equal sign (==). Figure 9 is the result of filtering anomaly data patterns:

```

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Source
Console Terminal
~/
> #menggunakan function basic_pattern_analysis pada kolom kode.pelanggan
> basic_pattern_analysis(data.pelangganCust.Id)=="A-9999"
[1] FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE FALSE
[14] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[27] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[40] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[53] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[66] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[79] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[92] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[105] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[118] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[131] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[144] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
> basic_pattern_analysis(data.pelangganCust.Id)=="a-99"
[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE
[14] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[27] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[40] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[53] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[66] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[79] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[92] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[105] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[118] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[131] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[144] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
> basic_pattern_analysis(data.pelangganCust.Id)=="A99999"
[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE
[14] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[27] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[40] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[53] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[66] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[79] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[92] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[105] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[118] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[131] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[144] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

```

Figure 9 Result of execution filtering function

By executing the command `basic_pattern_analysis (data.pelanggan$Cust.Id) == "A-9999"`, it will produce information on the location of the data that has anomalies with the pattern "A-9999" which is on row 8 (eight), it can be seen from the results with the appearance of "True". Likewise with `basic_pattern_analysis (data.Customer$Cust.Id) == "a-99"`, it will find anomaly data pattern on row 11 (eleven) and finally command `basic_pattern_analysis (data.customer$Cust.Id) == "A99999"` will find the anomaly data pattern at the position of row 12 (twelfth).

e. Utilizing grepl function

Filtering technique using the "==" (equal to double) operator can only be used to search or find data that has a specific text pattern. However, when we are looking for data that has certain characters, for example, we will do profiling on the "Full.name" column and we will look for data with text that contains non-letter characters, if we do this using the operator "==" then there will be a lot of it. patterns that can occur and this of course is not efficient, because the shape of the filter is a long list of text and not necessarily correct. For this purpose, the R language provides a function that can handle filtering based on regular expression (regex) patterns, where this "regex" will be used to detect various text patterns. The "regex" will function when combined with the "grepl" function. Figure 10 is an use of the "regex" function combined with the grepl function to find complex or nonspecific text data patterns.

```

1 #load library openxlsx
2 library(openxlsx)
3 library(bpa)
4 #memuat dataset pelanggan
5 data.pelanggan <- read.xlsx("D://penelitian/sardjono/pelanggan.xlsx")
6
7 #menggunakan fungsi grepl untuk mengambil pola nama tidak lazim
8 data.pelanggan[grepl(pattern="[^Aaw..]",x=basic_pattern_analysts(data.pelanggan$Full.name)),]
9
10 #>
11 #>
12 #>
13 #>
14 #>
15 #>
16 #>
17 #>
18 #>
19 #>
20 #>
21 #>
22 #>
23 #>
24 #>
25 #>
26 #>
27 #>
28 #>
29 #>
30 #>
31 #>
32 #>
33 #>
34 #>
35 #>
36 #>
37 #>
38 #>
39 #>
40 #>
41 #>
42 #>
43 #>
44 #>
45 #>
46 #>
47 #>
48 #>
49 #>
50 #>
51 #>
52 #>
53 #>
54 #>
55 #>
56 #>
57 #>
58 #>
59 #>
60 #>
61 #>
62 #>
63 #>
64 #>
65 #>
66 #>
67 #>
68 #>
69 #>
70 #>
71 #>
72 #>
73 #>
74 #>
75 #>
76 #>
77 #>
78 #>
79 #>
80 #>
81 #>
82 #>
83 #>
84 #>
85 #>
86 #>
87 #>
88 #>
89 #>
90 #>
91 #>
92 #>
93 #>
94 #>
95 #>
96 #>
97 #>
98 #>
99 #>
100 #>
101 #>
102 #>
103 #>
104 #>
105 #>
106 #>
107 #>
108 #>
109 #>
110 #>
111 #>
112 #>
113 #>
114 #>
115 #>
116 #>
117 #>
118 #>
119 #>
120 #>
121 #>
122 #>
123 #>
124 #>
125 #>
126 #>
127 #>
128 #>
129 #>
130 #>
131 #>
132 #>
133 #>
134 #>
135 #>
136 #>
137 #>
138 #>
139 #>
140 #>
141 #>
142 #>
143 #>
144 #>
145 #>
146 #>
147 #>
148 #>
149 #>
150 #>
151 #>
152 #>
153 #>
154 #>
155 #>
156 #>
157 #>
158 #>
159 #>
160 #>
161 #>
162 #>
163 #>
164 #>
165 #>
166 #>
167 #>
168 #>
169 #>
170 #>
171 #>
172 #>
173 #>
174 #>
175 #>
176 #>
177 #>
178 #>
179 #>
180 #>
181 #>
182 #>
183 #>
184 #>
185 #>
186 #>
187 #>
188 #>
189 #>
190 #>
191 #>
192 #>
193 #>
194 #>
195 #>
196 #>
197 #>
198 #>
199 #>
200 #>
201 #>
202 #>
203 #>
204 #>
205 #>
206 #>
207 #>
208 #>
209 #>
210 #>
211 #>
212 #>
213 #>
214 #>
215 #>
216 #>
217 #>
218 #>
219 #>
220 #>
221 #>
222 #>
223 #>
224 #>
225 #>
226 #>
227 #>
228 #>
229 #>
230 #>
231 #>
232 #>
233 #>
234 #>
235 #>
236 #>
237 #>
238 #>
239 #>
240 #>
241 #>
242 #>
243 #>
244 #>
245 #>
246 #>
247 #>
248 #>
249 #>
250 #>
251 #>
252 #>
253 #>
254 #>
255 #>
256 #>
257 #>
258 #>
259 #>
260 #>
261 #>
262 #>
263 #>
264 #>
265 #>
266 #>
267 #>
268 #>
269 #>
270 #>
271 #>
272 #>
273 #>
274 #>
275 #>
276 #>
277 #>
278 #>
279 #>
280 #>
281 #>
282 #>
283 #>
284 #>
285 #>
286 #>
287 #>
288 #>
289 #>
290 #>
291 #>
292 #>
293 #>
294 #>
295 #>
296 #>
297 #>
298 #>
299 #>
300 #>
301 #>
302 #>
303 #>
304 #>
305 #>
306 #>
307 #>
308 #>
309 #>
310 #>
311 #>
312 #>
313 #>
314 #>
315 #>
316 #>
317 #>
318 #>
319 #>
320 #>
321 #>
322 #>
323 #>
324 #>
325 #>
326 #>
327 #>
328 #>
329 #>
330 #>
331 #>
332 #>
333 #>
334 #>
335 #>
336 #>
337 #>
338 #>
339 #>
340 #>
341 #>
342 #>
343 #>
344 #>
345 #>
346 #>
347 #>
348 #>
349 #>
350 #>
351 #>
352 #>
353 #>
354 #>
355 #>
356 #>
357 #>
358 #>
359 #>
360 #>
361 #>
362 #>
363 #>
364 #>
365 #>
366 #>
367 #>
368 #>
369 #>
370 #>
371 #>
372 #>
373 #>
374 #>
375 #>
376 #>
377 #>
378 #>
379 #>
380 #>
381 #>
382 #>
383 #>
384 #>
385 #>
386 #>
387 #>
388 #>
389 #>
390 #>
391 #>
392 #>
393 #>
394 #>
395 #>
396 #>
397 #>
398 #>
399 #>
400 #>
401 #>
402 #>
403 #>
404 #>
405 #>
406 #>
407 #>
408 #>
409 #>
410 #>
411 #>
412 #>
413 #>
414 #>
415 #>
416 #>
417 #>
418 #>
419 #>
420 #>
421 #>
422 #>
423 #>
424 #>
425 #>
426 #>
427 #>
428 #>
429 #>
430 #>
431 #>
432 #>
433 #>
434 #>
435 #>
436 #>
437 #>
438 #>
439 #>
440 #>
441 #>
442 #>
443 #>
444 #>
445 #>
446 #>
447 #>
448 #>
449 #>
450 #>
451 #>
452 #>
453 #>
454 #>
455 #>
456 #>
457 #>
458 #>
459 #>
460 #>
461 #>
462 #>
463 #>
464 #>
465 #>
466 #>
467 #>
468 #>
469 #>
470 #>
471 #>
472 #>
473 #>
474 #>
475 #>
476 #>
477 #>
478 #>
479 #>
480 #>
481 #>
482 #>
483 #>
484 #>
485 #>
486 #>
487 #>
488 #>
489 #>
490 #>
491 #>
492 #>
493 #>
494 #>
495 #>
496 #>
497 #>
498 #>
499 #>
500 #>
501 #>
502 #>
503 #>
504 #>
505 #>
506 #>
507 #>
508 #>
509 #>
510 #>
511 #>
512 #>
513 #>
514 #>
515 #>
516 #>
517 #>
518 #>
519 #>
520 #>
521 #>
522 #>
523 #>
524 #>
525 #>
526 #>
527 #>
528 #>
529 #>
530 #>
531 #>
532 #>
533 #>
534 #>
535 #>
536 #>
537 #>
538 #>
539 #>
540 #>
541 #>
542 #>
543 #>
544 #>
545 #>
546 #>
547 #>
548 #>
549 #>
550 #>
551 #>
552 #>
553 #>
554 #>
555 #>
556 #>
557 #>
558 #>
559 #>
560 #>
561 #>
562 #>
563 #>
564 #>
565 #>
566 #>
567 #>
568 #>
569 #>
570 #>
571 #>
572 #>
573 #>
574 #>
575 #>
576 #>
577 #>
578 #>
579 #>
580 #>
581 #>
582 #>
583 #>
584 #>
585 #>
586 #>
587 #>
588 #>
589 #>
590 #>
591 #>
592 #>
593 #>
594 #>
595 #>
596 #>
597 #>
598 #>
599 #>
600 #>
601 #>
602 #>
603 #>
604 #>
605 #>
606 #>
607 #>
608 #>
609 #>
610 #>
611 #>
612 #>
613 #>
614 #>
615 #>
616 #>
617 #>
618 #>
619 #>
620 #>
621 #>
622 #>
623 #>
624 #>
625 #>
626 #>
627 #>
628 #>
629 #>
630 #>
631 #>
632 #>
633 #>
634 #>
635 #>
636 #>
637 #>
638 #>
639 #>
640 #>
641 #>
642 #>
643 #>
644 #>
645 #>
646 #>
647 #>
648 #>
649 #>
650 #>
651 #>
652 #>
653 #>
654 #>
655 #>
656 #>
657 #>
658 #>
659 #>
660 #>
661 #>
662 #>
663 #>
664 #>
665 #>
666 #>
667 #>
668 #>
669 #>
670 #>
671 #>
672 #>
673 #>
674 #>
675 #>
676 #>
677 #>
678 #>
679 #>
680 #>
681 #>
682 #>
683 #>
684 #>
685 #>
686 #>
687 #>
688 #>
689 #>
690 #>
691 #>
692 #>
693 #>
694 #>
695 #>
696 #>
697 #>
698 #>
699 #>
700 #>
701 #>
702 #>
703 #>
704 #>
705 #>
706 #>
707 #>
708 #>
709 #>
710 #>
711 #>
712 #>
713 #>
714 #>
715 #>
716 #>
717 #>
718 #>
719 #>
720 #>
721 #>
722 #>
723 #>
724 #>
725 #>
726 #>
727 #>
728 #>
729 #>
730 #>
731 #>
732 #>
733 #>
734 #>
735 #>
736 #>
737 #>
738 #>
739 #>
740 #>
741 #>
742 #>
743 #>
744 #>
745 #>
746 #>
747 #>
748 #>
749 #>
750 #>
751 #>
752 #>
753 #>
754 #>
755 #>
756 #>
757 #>
758 #>
759 #>
760 #>
761 #>
762 #>
763 #>
764 #>
765 #>
766 #>
767 #>
768 #>
769 #>
770 #>
771 #>
772 #>
773 #>
774 #>
775 #>
776 #>
777 #>
778 #>
779 #>
780 #>
781 #>
782 #>
783 #>
784 #>
785 #>
786 #>
787 #>
788 #>
789 #>
790 #>
791 #>
792 #>
793 #>
794 #>
795 #>
796 #>
797 #>
798 #>
799 #>
800 #>
801 #>
802 #>
803 #>
804 #>
805 #>
806 #>
807 #>
808 #>
809 #>
810 #>
811 #>
812 #>
813 #>
814 #>
815 #>
816 #>
817 #>
818 #>
819 #>
820 #>
821 #>
822 #>
823 #>
824 #>
825 #>
826 #>
827 #>
828 #>
829 #>
830 #>
831 #>
832 #>
833 #>
834 #>
835 #>
836 #>
837 #>
838 #>
839 #>
840 #>
841 #>
842 #>
843 #>
844 #>
845 #>
846 #>
847 #>
848 #>
849 #>
850 #>
851 #>
852 #>
853 #>
854 #>
855 #>
856 #>
857 #>
858 #>
859 #>
860 #>
861 #>
862 #>
863 #>
864 #>
865 #>
866 #>
867 #>
868 #>
869 #>
870 #>
871 #>
872 #>
873 #>
874 #>
875 #>
876 #>
877 #>
878 #>
879 #>
880 #>
881 #>
882 #>
883 #>
884 #>
885 #>
886 #>
887 #>
888 #>
889 #>
890 #>
891 #>
892 #>
893 #>
894 #>
895 #>
896 #>
897 #>
898 #>
899 #>
900 #>
901 #>
902 #>
903 #>
904 #>
905 #>
906 #>
907 #>
908 #>
909 #>
910 #>
911 #>
912 #>
913 #>
914 #>
915 #>
916 #>
917 #>
918 #>
919 #>
920 #>
921 #>
922 #>
923 #>
924 #>
925 #>
926 #>
927 #>
928 #>
929 #>
930 #>
931 #>
932 #>
933 #>
934 #>
935 #>
936 #>
937 #>
938 #>
939 #>
940 #>
941 #>
942 #>
943 #>
944 #>
945 #>
946 #>
947 #>
948 #>
949 #>
950 #>
951 #>
952 #>
953 #>
954 #>
955 #>
956 #>
957 #>
958 #>
959 #>
960 #>
961 #>
962 #>
963 #>
964 #>
965 #>
966 #>
967 #>
968 #>
969 #>
970 #>
971 #>
972 #>
973 #>
974 #>
975 #>
976 #>
977 #>
978 #>
979 #>
980 #>
981 #>
982 #>
983 #>
984 #>
985 #>
986 #>
987 #>
988 #>
989 #>
990 #>
991 #>
992 #>
993 #>
994 #>
995 #>
996 #>
997 #>
998 #>
999 #>
1000 #>

```

Figure 10. result of using combination function grepl with regex

By giving the command "data.pelanggan[grepl(pattern = "[^ Aaw..]", X = basic_pattern_analysis (data.pelanggan\$Full.Name))]", we will get information that column "Full.name" has data which is unusual. There are 7 unusual data, namely on the rows : 7,46,50,58,71,80 and 103.

For columns in the customer dataset, profiling must be done with the grepl function combined with the regex function in order to perform data cleansing quickly and precisely for further improvement.

5. Conclusion

The results of this research can be used as a reference for those who will career into a data scientist, where the basic skills that must be possessed are started from understanding the basic functions which are very intensively used in doing data cleansing or data profiling processes. The following are guidelines that must be done in carrying out the data profiling process which is the first stage in data cleansing activities. In the data profiling process we carry out very important activities, namely:

- Identifying data patterns contained in dataset columns or data science;
- Analyze the feasibility of data by comparison of expectations or reasonable scientific measures, to find data that needs to be corrected or changed

Both that processes can be done using the functions and operators in R languaged has, namely :

- summary();
- basic_pattern_analysis();
- operator == (double equal) and
- grepl for retrieve data.

By having these profiling skills, we can recognize and see outliers that are easily found and corrected as needed. In other words, without the ability to identify data columns and retrieve this data, of course the data cleansing process or data correction cannot be done.

References

- Brayne, S. (2017). Big data surveillance: The case of policing. *American sociological review*, 82(5), 977-1008.
- Dietrich, D. (2015). *Data science and big data analytics: Discovering, analyzing, visualizing and presenting data*. New York: John Wiley & Sons.
- Endel, F., and Piringer, H. (2015). Data Wrangling: Making data useful again. *IFAC-PapersOnLine*, 48(1), 111-112.
- Faisal, M. R. (2016). *Seri Belajar Pemrograman: Pengenalan Bahasa Pemrograman R*, Jakarta: Indonesia Net Developer Community.
- Huerta, E., and Jensen, S. (2017). An accounting information systems perspective on data analytics and Big Data. *Journal of Information Systems*, 31(3), 101-114.
- Kandel, S., Heer, J., Plaisant, C., Kennedy, J., Van Ham, F., Riche, N. H and Buono, P. (2011). Research directions in data wrangling: Visualizations and transformations for usable and credible data. *Information Visualization*, 10(4), 271-288.
- Kowalczyk, M., and Buxmann, P. (2014). Big data and information processing in organizational decision processes. *Business & Information Systems Engineering*, 6(5), 267-278.
- Mailund, T. (2017). *Beginning Data Science in R: Data Analysis, Visualization, and Modelling for the Data Scientist*. Apress. Denmark: Apress.
- Patil, M. M., and Hiremath, B. N. (2018). A systematic study of data wrangling. *Int. J. Inf. Technol. Comput. Sci.(IJITCS)*, 1, 32-39.
- Sivaparthipan, C. B., Karthikeyan, N., and Karthik, S. (2020). Designing statistical assessment healthcare information system for diabetics analysis using big data. *Multimedia Tools and Applications*, 79(13), 8431-8444.
- Whyte, J., Stasis, A., and Lindkvist, C. (2016). Managing change in the delivery of complex projects: Configuration management, asset information and 'big data'. *International Journal of Project Management*, 34(2), 339-351.
- Wu, X., Zhu, X., Wu, G. Q., & Ding, W. (2013). Data mining with big data. *IEEE transactions on knowledge and data engineering*, 26(1), 97-107.