Data deidentification and modification

ETC5512 Assignment 3, Master of Business Analytics

Prepared by Prachi Jaiswal, 32192673, pjai0005@student.monash.edu (mailto:pjai0005@student.monash.edu)

2021-06-04

Open data are the structured data that are purposely made available to the people and organizations to use and republish. These data are mainly used for analysis that can help to improve services. Data can be freely used, reused, and redistributed to anyone. However, the data that is made available can contain some sensitive information, which is why it is necessary to maintain the integrity of data ethics by protecting the privacy of people.

Analysis

```
survey_data<- readRDS(here::here("raw_data/survey_data.rds"))
#survey_data<- read_rds(here::here("raw_data/survey_data.rds"))
```

The open data can encounter breaches if the user knows personal details via **Direct identifiers**. These identifiers are the types of information that directly links the variable to subjects, people or institutions. ("Australian Privacy Principles", 2021) Common examples are an individual's name, signature, address, telephone number, date of birth, medical records, bank account details, employment details and commentary or opinion about a person.

Therefore we need to remove direct identifiers in order to obey the principles of data ethics to support the ethical use of data digitally. According to (Kennedy, 2021), there are eight techniques of Data Modification for de-identifying the direct identifiers:

1. Sampling
2. Choice of variables
3. Aggregation
4. Perturbation
5. Swapping
6. Manufacturing synthetic data
7. Encryption of identifiers
8. Top and bottom coding

The report consists of some of the given techniques to remove direct identifiers.

Identify and remove the direct identifiers from the data.

```
data <- survey_data %>%
  mutate(StartDate = as.Date(StartDate),
         EndDate = as.Date(EndDate),
         RecordedDate = as.Date(RecordedDate),
         QID6= as.integer(QID6),
         QID19= as.integer(QID19),
         QID21= as.integer(QID21),
         QID22= as.integer(QID22))%>%

  select(-c(QID28, IPAddress, ResponseID,
            ResponseLastName, ResponseFirstName,
            LocationLatitude, LocationLongitude))
```

The above code removes the following variables with the help of the **Choice of variables** Technique:

- QID28: This variable consists of individual's email addresses which are considered as a unique identity because every person has a unique email id.
- IPAddress: An IP address is considered to be unique because it can recognize the device used for the survey. It can also track down the location of the respondent.
- ResponseLastName: Respondent's last name is a direct identifier here because we have a total of 1000 respondents and it makes it easier to discover the respondent.
- ResponseFirstName: Respondent's first name is a direct identifier here because we have a total of 1000 respondents and it makes it easier to discover the respondent.
- ResponseID: This is Respondent's Id which is recorded while taking the survey. Every respondent is assigned a unique response number.
- LocationLatitude: Latitude is a geographic measurement of the Earth that has the ability to specify the current location of the respondent.

- LocationLongitude: Longitude is a geographic measurement of the Earth that has the ability to specify the current location of the respondent.

Removal of direct identifiers is one of an integral part of De-identification, however, it doesn’t solely remove the vulnerability of identification. So the next section focuses on other approaches to Data Modification.

De-identification strategy

De-identification is a process that helps to prevent someone’s personal information from being revealed by removing or masking personally identifiable variables.

```
de_idendified <- data %>%

  mutate(Age_group = as.character(cut(QID6, breaks = c(20,40,60,90))),     #Used Aggregation for Age
         Children = as.integer(sample(QID8, 1000, replace = FALSE)),          #Used Swapping for Children
         Postcode = as.integer(sample(QID15, 1000, replace = FALSE))) %>%      #Used Swapping for Postcode

  mutate(Recorded_Year = year(RecordedDate),
         Recorded_Month = month(RecordedDate),
         Recorded_Day = day(RecordedDate)) %>%

  group_by(StartDate) %>%
  mutate(Recorded_Day = sample(Recorded_Day,n(),replace = FALSE)) %>%    #Used Swapping for day

  mutate(Age_group = ifelse(Progress < 10, median(Age_group, na.rm = TRUE), Age_group),
         Postcode = ifelse(Progress < 20, median(Postcode, na.rm = TRUE), Postcode),
         Children = ifelse(Progress < 10, median(Children, na.rm = TRUE), Children),
         Adults = ifelse(Progress < 20, median(QID7, na.rm = TRUE), QID7)) %>%

  select(-c(QID6, QID8, QID15, RecordedDate, QID7))
```

- For the respondent’s age, the **Aggregation** method is used. This process combines the information that might be individually identifiable into groups. Here the age is grouped into 3 ranges *([20 to 40], [40 to 60], [60 to 90])*, rather than displayed as respondent’s actual age. The data consist of two people who age more than 80, hence the last age range is taken till 90yrs instead of 80yrs.
- **Swapping** method is used for the respondent’s Postcode and the number of Children (>18yrs) living in their houses. This strategy helps to observe individual’s data without having to make distributed assumptions. It is very useful between groups of individuals in particular geographic areas, and this report talks about the COVID survey conducted in Melbourne which comes of as the only city of interest.
- There were 21 respondents who hadn’t finish their survey. And the possibility of them getting identified was high because there were 16 distinct dates where these 21 respondents started their survey. This could perhaps have increased the risk of recognition. To mitigate this, Recorded day undergoes **Swapping.**
- For income of years, 2019, 2020 and 2021, two data modification techniques were used.

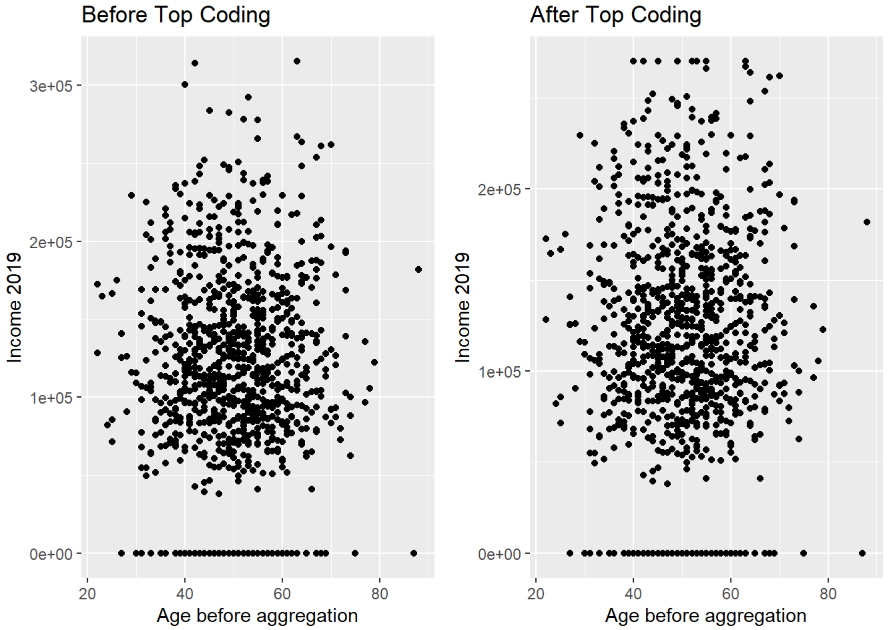
```
de_identity <- de_idendified %>%
  mutate(QID19 = ifelse(QID19 > 250000, 250000, QID19),
         QID21 = ifelse(QID21 > 250000, 250000, QID21),
         QID22 = ifelse(QID22 > 250000, 250000, QID22))%>%     #Used Top Down to remove outliers

  mutate(Income21 = ifelse(QID19 != 0, QID19 + rnorm(n(), 0, 5000), QID19),
         Income20 = ifelse(QID21 != 0, QID21 + rnorm(n(), 0, 5000), QID21),
         Income19 = (QID22 + rnorm(n(), 0, 5000))) %>%      #Used Perturbation for Incomes

  select(-c (QID19, QID21, QID22, ))
```

One of the two strategies used is **Perturbation** which helps to add a small amount of noise to the data in order to preserve the privacy of an individual. This is used to modify the values that are likely to identify a person. Using this method the incomes get censored, consequently reducing the identification risk.

However, this doesn’t truncate the concept of de-identification because it consists of outliers as shown in the graph below. Hence, along with perturbation, **Top and Bottom Coding** is used to lessen the fragility of being identified.



With Top and Bottom Coding, rare individuals whose incomes are exceptionally high or low are masked. This method uses truncation of outliers resulting in reduced uncertainty. This allows dispersion of low and high-risk data to be appropriately masked.

Check strategy

The strategy which helps to describes the testing approach of any strategy used for the de-identification process is called check strategy. An example from each strategy used for data modification is shown below:

```
#Aggregation
age_check <-  de_identity %>%
  group_by(Age_group)%>%
  mutate(Freq = n()) %>%
  ungroup() %>%
  filter(Freq == 1)

#Swapping
children_check <-  de_identity %>%
  group_by(Children)%>%
  mutate(Freq = n()) %>%
  ungroup() %>%
  filter(Freq == 1)

#Perturbation and Top-Bottom
income_check <-  de_identity %>%
  group_by(Income21)%>%
  mutate(Freq = n()) %>%
  ungroup() %>%
  filter(Freq == 1)
```

- Aggregation: No individual can be identified as there are zero unique observations because the age has been transformed into 3 range group.
- Swapping: No individual can be identified as there are zero unique observations for the number of children who live in the respondent’s house.
- Perturbation and Top-Bottom: No individual can be identified even though income has 893 observations because the Perturbation method was used to add noise making it difficult for anyone to discover a particular person in the survey. Moreover, the use of the Top-Bottom approach helped to reduce the potential possibility of discovering subjects with exceptional incomes.

Computer-readable structure

The data that can be processed by a computer and is structured properly is computer-readable data eg. CSV, RDF, XML, JSON files (“Machine-readable data - Wikipedia”, 2021). These formats are only machine-readable if the data contained within them is formally structured.

```
wfh <- function(x){
  x = str_replace_all(x, c("0" = "Didn’t answer",
                           "1" = "Didn’t work",
                           "2" = "Never",
                           "3" = "Sometimes",
                           "4" = "About half the time",
                           "5" = "Most of the time",
                           "6" = "Always"))
}

whrs <- function(x){
  x = str_replace_all(x, c("0" = "Didn't answer",
                           "1" = "Didn't work",
                           "2" = "Less than 10hrs",
                           "3" = "10-20 hrs",
                           "4" = "20-30 hrs",
                           "5" = "30-40 hrs",
                           "6" = "40+ hrs"))
}

sch <- function(x){
 x = str_replace_all( x, c("1" = "No changes",
                           "2" = "Some small changes",
                           "3" = "Varying",
                           "4" = "Unpredictable"))
}

men_hel <- function(x){
  x = str_replace_all(x,c("1" = "Good",
                          "2" = "Some Challenges",
                          "3" = "Significant Challenges"))
}
```

```
final <- de_identity %>%
  rename(Agreement = QID29,
         WFH_20 = QID10,
         WFH_19 = QID12,
         WFH_hrs_19 = QID14,
         WFH_hrs_20 = QID16,
         WW19_1 = QID17_1,
         WW19_2 = QID17_2,
         WW19_3 = QID17_3,
         WW19_4 = QID17_4,
         WW19_5 = QID17_5,
         WW20_1 = QID18_1,
         WW20_2 = QID18_2,
         WW20_3 = QID18_3,
         WW20_4 = QID18_4,
         WW20_5 = QID18_5,
         HL19_1 = QID24_1,
         HL19_2 = QID24_2,
         HL19_3 = QID24_3,
         HL19_4 = QID24_4,
         HL19_5 = QID24_5,
         HL19_6 = QID24_6,
         HL20_1 = QID25_1,
         HL20_2 = QID25_2,
         HL20_3 = QID25_3,
         HL20_4 = QID25_4,
         HL20_5 = QID25_5,
         HL20_6 = QID25_6,
         Work_Schedule_19 = QID20,
         Work_Schedule_20 = QID23,
         Mental_Health_19 = QID26,
         Mental_Health_20 = QID27,) %>%

  mutate(Finished = str_replace_all(Finished, c("1"= "Yes", "0" ="No")),
         WFH_19 = wfh (WFH_19),
         WFH_20 = wfh (WFH_20),
         WFH_hrs_19 = whrs(WFH_hrs_19),
         WFH_hrs_20 = whrs(WFH_hrs_20),
         Work_Schedule_19 = sch(Work_Schedule_19),
         Work_Schedule_20 = sch(Work_Schedule_20),
         Mental_Health_19 = men_hel(Mental_Health_19),
         Mental_Health_20 = men_hel(Mental_Health_20))   #used functions
```

- Above steps are carried out to make the data more understandable and structured.

- Previously, the variable names of the data were difficult to comprehend, therefore, functions are created to replace numbers with strings.
- This data is not in its cleanest format (pivoted long). This is because the data is supposed to be deployed as open data, and usually open data are in wider formats.

Save data in a csv form in the data folder

```
#write.csv(final, here::here("data/release-data-Prachi-Jaiswal.csv"))
write.csv(final, "../data/release-data-Jaiswal-Prachi.csv")
```

Resources

Software used:

- R Core Team (2020). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/ (https://www.R-project.org/).
- Excel, M. (2021). Microsoft Excel Spreadsheet Software | Microsoft 365. Retrieved 4 June 2021, from https://www.microsoft.com/en-ww/microsoft-365/excel (https://www.microsoft.com/en-ww/microsoft-365/excel)

R libraries used:

- Wickham et al., (2019). Welcome to the tidyverse. Journal of Open Source Software, 4(43), 1686, https://doi.org/10.21105/joss.01686 (https://doi.org/10.21105/joss.01686)
- H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016.
- Garrett Grolemund, Hadley Wickham (2011). Dates and Times Made Easy with lubridate. Journal of Statistical Software, 40(3), 1-25. URL https://www.jstatsoft.org/v40/i03/ (https://www.jstatsoft.org/v40/i03/).
- Nicholas Tierney, Di Cook, Miles McBain and Colin Fay (2021). naniar: Data Structures, Summaries, and Visualisations for Missing Data. R package version 0.6.1. https://CRAN.R-project.org/package=naniar (https://CRAN.R-project.org/package=naniar)
- Alboukadel Kassambara (2020). ggpubr: ‘ggplot2’ Based Publication Ready Plots. R package version 0.4.0. https://CRAN.R-project.org/package=ggpubr (https://CRAN.R-project.org/package=ggpubr)

Sources used:

- Kennedy, L. (2021). Lecture 11: Data Privacy [Ebook]. Monash University, Department of Econometrics and Business Statistics. Retrieved from https://wcd.numbat.space/lectures/lecture-11.pdf (https://wcd.numbat.space/lectures/lecture-11.pdf)
- What is a Codebook?. (2021). Retrieved 1 June 2021, from https://www.icpsr.umich.edu/icpsrweb/content/shared/ICPSR/faqs/what-is-a-codebook.html (https://www.icpsr.umich.edu/icpsrweb/content/shared/ICPSR/faqs/what-is-a-codebook.html)
- Oecd.org. 2021. 2018 Database - PISA. https://www.oecd.org/pisa/data/2018database/ (https://www.oecd.org/pisa/data/2018database/)
- Australian Privacy Principles. (2021). Retrieved 3 June 2021, from https://www.oaic.gov.au/privacy/australian-privacy-principles (https://www.oaic.gov.au/privacy/australian-privacy-principles)
- Machine-readable data - Wikipedia. (2021). Retrieved 4 June 2021, from https://en.wikipedia.org/wiki/Machine-readable_data# (https://en.wikipedia.org/wiki/Machine-readable_data#):~:text=Machine%2Dreadable%20data%2C%20or%20computer,be%20processed%20by%20a%20computer.&text=These%20formats