NSDC Spring 2025 Project - Data Cleanup

Lea Choe, Elias Grant, Brandon Liu, Ivan Chuang

4/23/25

The steps below are in accordance to

- YouTube: Explore your data using R programming
- YouTube: Clean your data with R

Chosen Data Set

Financial Transactions Dataset: Analytics

- `cards_data.csv`
  - Credit and debit card details - card limits, types, and activation dates.
- `transactions_data.csv`
  - Detailed transaction records including amounts, timestamps, and merchant details
  - Covers transactions throughout the 2010s
  - Features transaction types, amounts, and merchant information
  - Perfect for analyzing spending patterns and building fraud detection models
- `users_data.csv`
  - Demographic information about customers
  - Account-related details
  - Enables customer segmentation and personalized analysis

Goal: Analyze the three datasets and uncover insight about financial records, fraud detection, customer behavior analysis, or another relevant topic.

DATA SET 1: `cards_data.csv`

```r
# tools
# tinytex::install_tinytex()
library("dplyr")
```

```
## Warning: package 'dplyr' was built under R version 4.4.2
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
# data
cards_data <- read.csv("C://Users//elias//Downloads//NSDC//cards_data.csv", header = TRUE, sep = ",")
```

Exploring Data Set

The basic structure and look of the data set

```r
str(cards_data)  # 6146 obs, 13 variables (6146 rows, 13 cols)
```

```
## 'data.frame':    6146 obs. of  13 variables:
##  $ id                   : int  4524 2731 3701 42 4659 4537 1278 3687 3465 3754 ...
##  $ client_id            : int  825 825 825 825 825 1746 1746 1746 1746 1746 ...
##  $ card_brand           : chr  "Visa" "Visa" "Visa" "Visa" ...
##  $ card_type            : chr  "Debit" "Debit" "Debit" "Credit" ...
##  $ card_number          : num  4.34e+15 4.96e+15 4.58e+15 4.88e+15 5.72e+15 ...
##  $ expires              : chr  "12/2022" "12/2020" "02/2024" "08/2024" ...
##  $ cvv                  : int  623 393 719 693 75 736 972 48 722 908 ...
##  $ has_chip             : chr  "YES" "YES" "YES" "NO" ...
##  $ num_cards_issued     : int  2 2 2 1 1 1 2 2 2 1 ...
##  $ credit_limit         : chr  "$24295" "$21968" "$46414" "$12400" ...
##  $ acct_open_date       : chr  "09/2002" "04/2014" "07/2003" "01/2003" ...
##  $ year_pin_last_changed: int  2008 2014 2004 2012 2009 2012 2011 2015 2015 2012 ...
##  $ card_on_dark_web     : chr  "No" "No" "No" "No" ...
```

```r
head(cards_data) # first few rows
```

```
##     id client_id card_brand       card_type  card_number expires cvv has_chip
## 1 4524       825       Visa           Debit 4.344677e+15 12/2022 623      YES
## 2 2731       825       Visa           Debit 4.956966e+15 12/2020 393      YES
## 3 3701       825       Visa           Debit 4.582313e+15 02/2024 719      YES
## 4   42       825       Visa          Credit 4.879494e+15 08/2024 693       NO
## 5 4659       825 Mastercard Debit (Prepaid) 5.722875e+15 03/2009  75      YES
## 6 4537      1746       Visa          Credit 4.404899e+15 09/2003 736      YES
##   num_cards_issued credit_limit acct_open_date year_pin_last_changed
## 1                2       $24295        09/2002                  2008
## 2                2       $21968        04/2014                  2014
## 3                2       $46414        07/2003                  2004
## 4                1       $12400        01/2003                  2012
## 5                1          $28        09/2008                  2009
## 6                1       $27500        09/2003                  2012
##   card_on_dark_web
## 1               No
## 2               No
## 3               No
## 4               No
## 5               No
## 6               No
```

```
tail(cards_data) # last few rows
```

```
##        id client_id card_brand card_type  card_number expires cvv has_chip
## 6141 4046       185 Mastercard     Debit 5.916545e+15 07/2024 314      YES
## 6142 5361       185       Amex    Credit 3.006098e+14 01/2024 663      YES
## 6143 2711       185       Visa    Credit 4.718517e+15 01/2021 492      YES
## 6144 1305      1007 Mastercard    Credit 5.929512e+15 08/2020 237       NO
## 6145  743      1110 Mastercard     Debit 5.589769e+15 01/2020 630      YES
## 6146 3199      1110       Visa    Credit 4.994011e+15 12/2020 120      YES
##      num_cards_issued credit_limit acct_open_date year_pin_last_changed
## 6141                1       $16415        07/2016                  2016
## 6142                1        $6900        11/2000                  2013
## 6143                2        $5700        04/2012                  2012
## 6144                2        $9200        02/2012                  2012
## 6145                1       $28074        01/2020                  2020
## 6146                1       $14400        05/2017                  2017
##      card_on_dark_web
## 6141               No
## 6142               No
## 6143               No
## 6144               No
## 6145               No
## 6146               No
```

Table cd_var_type - shows each variable and its type

```
cd_var_type <- tibble(
  variable = names(cards_data),                    # col 1
  type = sapply(cards_data, function(x) class(x)) # col 2
)

cd_var_type <- cd_var_type |> arrange(desc(type))
print(cd_var_type)
```

```
## # A tibble: 13 x 2
##    variable              type
##    <chr>                 <chr>
##  1 card_number           numeric
##  2 id                    integer
##  3 client_id             integer
##  4 cvv                   integer
##  5 num_cards_issued      integer
##  6 year_pin_last_changed integer
##  7 card_brand            character
##  8 card_type             character
##  9 expires               character
## 10 has_chip              character
## 11 credit_limit          character
## 12 acct_open_date        character
## 13 card_on_dark_web      character
```

Make it easier to see categories of qualitative variables

```r
# Subset the data to only the selected columns and apply unique()
selected_columns <- cards_data[c("card_brand", "card_type", "has_chip", "card_on_dark_web")]
unique_values <- lapply(selected_columns, unique)

# Print the unique values for the selected columns
print(unique_values)
```

```
## $card_brand
## [1] "Visa"       "Mastercard" "Discover"   "Amex"
##
## $card_type
## [1] "Debit"           "Credit"          "Debit (Prepaid)"
##
## $has_chip
## [1] "YES" "NO"
##
## $card_on_dark_web
## [1] "No"
```

```r
# Show table of select columns
table(cards_data$card_brand)
```

```
##
##       Amex   Discover Mastercard       Visa
##        402        209       3209       2326
```

```r
table(cards_data$card_type)
```

```
##
##          Credit           Debit Debit (Prepaid)
##            2057            3511             578
```

```
table(cards_data$has_chip)
```

```
## 
##   NO  YES 
##  646 5500 
```

```
table(cards_data$card_on_dark_web)
```

```
## 
##   No 
## 6146 
```

```
table(cards_data$num_cards_issued)
```

```
## 
##    1    2    3 
## 3114 2972   60 
```

Quantitative/Numeric variables:

- `id, client_id, card_number, cvv, num_cards_issued`

Qualitative/Categorical variables:

- `card_brand, card_type, expires, has_chip, credit_limit, card_on_dark_web`
- we can make `card_brand, card_type, has_chip, credit_limit,``card_on_dark_web` into numeric
- we can make `acct_open_type` into a numeric BUT not quantitative because format is MM/YYYY (date)

Cleaning Data Set

Find and Deal with Missing Data - none

```
cards_data |>
  filter(!complete.cases(cards_data)) # there are no observations with NA
```

```
##  [1] id                    client_id             card_brand           
##  [4] card_type             card_number           expires              
##  [7] cvv                   has_chip              num_cards_issued     
## [10] credit_limit          acct_open_date        year_pin_last_changed
## [13] card_on_dark_web     
## <0 rows> (or 0-length row.names)
```

Find and Deal with Duplicates - none

```r
cards_data[duplicated(cards_data)] # NO duplicated observations!
```

```
## data frame with 0 columns and 6146 rows
```

Make card_brand, card_type, has_chip, credit_limit, card_on_dark_web into numeric: card_brand --> card_brand_num

```r
# MUTATE - permanently create card_brand_num -- quantitative version of card_brand
# card_brand still exists
cards_data <-
  cards_data |>
  mutate(card_brand_num = recode(card_brand,
                                "Amex" = 1,
                                "Discover" = 2,
                                "Mastercard" = 3,
                                "Visa" = 4))
```

card_type --> card_type_num

```r
# MUTATE - permanently create card_type_num -- quantitative version of card_type
# card_type still exists
cards_data <-
  cards_data |>
  mutate(card_type_num = recode(card_type,
                               "Credit" = 0,
                               "Debit" = 1,
                               "Debit (Prepaid)" = 2))
```

has_chip --> has_chip_num

```r
# MUTATE - permanently create has_chip_num -- quantitative version of has_chip
# has_chip still exists
cards_data <-
  cards_data |>
  mutate(has_chip_num = recode(has_chip,
                              "YES" = 1,
                              "NO" = 0,))
```

credit_limit --> credit_limit_num

```
# MUTATE - permanently create credit_limit_num -- quantitative version of credit_limit
# credit_limit still exists
# Remove dollar sign and convert to numeric
cards_data$credit_limit_num <- as.numeric(gsub("\\$", "", cards_data$credit_limit))
print(cards_data$credit_limit_num[1]) # test that it worked
```

```
## [1] 24295
```

card_on_dark_web --> card_on_dark_web_num

```
# MUTATE - permanently create card_on_dark_web_num -- quantitative version of card_on_dark_web
# card_on_dark_web still exists
cards_data <-
  cards_data |>
  mutate(card_on_dark_web_num = recode(card_on_dark_web,
                                       "Yes" = 1, # nonexistent!
                                       "No" = 0))

# Note: There are no cards on the dark web!
```

Now see the whole thing

```
head(cards_data, 5)
```

```
##     id client_id card_brand       card_type  card_number expires cvv has_chip
## 1 4524       825       Visa           Debit 4.344677e+15 12/2022 623      YES
## 2 2731       825       Visa           Debit 4.956966e+15 12/2020 393      YES
## 3 3701       825       Visa           Debit 4.582313e+15 02/2024 719      YES
## 4   42       825       Visa          Credit 4.879494e+15 08/2024 693       NO
## 5 4659       825 Mastercard Debit (Prepaid) 5.722875e+15 03/2009  75      YES
##   num_cards_issued credit_limit acct_open_date year_pin_last_changed
## 1                2       $24295        09/2002                  2008
## 2                2       $21968        04/2014                  2014
## 3                2       $46414        07/2003                  2004
## 4                1       $12400        01/2003                  2012
## 5                1          $28        09/2008                  2009
##   card_on_dark_web card_brand_num card_type_num has_chip_num credit_limit_num
## 1               No              4             1            1            24295
## 2               No              4             1            1            21968
## 3               No              4             1            1            46414
## 4               No              4             0            0            12400
## 5               No              3             2            1               28
##   card_on_dark_web_num
## 1                    0
## 2                    0
## 3                    0
## 4                    0
## 5                    0
```

“personal” variables (no levels or categories within the variable)

- **id, client_id, card_number, expires, cvv, credit_limit, acct_open_date**

```
# NEW DATAFRANE: nonp_cards_data
# nonp_cards_data = only the "nonpersonal" variables that people may have in common
# arranged by card_brand, then card_type, has_chip, num_cards_issued, card_on_dark_web
nonp_cards_data <- cards_data |>
  select(card_brand, card_type, has_chip, num_cards_issued, card_on_dark_web) |>
  arrange(card_brand, card_type, has_chip, num_cards_issued, card_on_dark_web)

# shows how many people have in common with unique combinations of nonp_cards_data
nonp_cards_data |>
  count(card_brand, card_type, has_chip, num_cards_issued, card_on_dark_web) |>
  arrange(desc(n))
```

```
##    card_brand       card_type has_chip num_cards_issued card_on_dark_web   n
## 1  Mastercard           Debit      YES                1               No 984
## 2  Mastercard           Debit      YES                2               No 960
## 3        Visa           Debit      YES                1               No 612
## 4        Visa           Debit      YES                2               No 564
## 5        Visa          Credit      YES                2               No 364
## 6        Visa          Credit      YES                1               No 350
## 7  Mastercard          Credit      YES                1               No 284
## 8  Mastercard          Credit      YES                2               No 271
## 9        Amex          Credit      YES                2               No 183
## 10       Amex          Credit      YES                1               No 181
## 11 Mastercard Debit (Prepaid)      YES                1               No 178
## 12 Mastercard Debit (Prepaid)      YES                2               No 162
## 13 Mastercard           Debit       NO                1               No 119
## 14 Mastercard           Debit       NO                2               No 108
## 15   Discover          Credit      YES                1               No  99
## 16       Visa Debit (Prepaid)      YES                2               No  88
## 17   Discover          Credit      YES                2               No  87
## 18       Visa Debit (Prepaid)      YES                1               No  81
## 19       Visa           Debit       NO                1               No  76
## 20       Visa           Debit       NO                2               No  63
## 21       Visa          Credit       NO                1               No  44
## 22       Visa          Credit       NO                2               No  44
## 23 Mastercard          Credit       NO                1               No  43
## 24 Mastercard          Credit       NO                2               No  27
## 25 Mastercard Debit (Prepaid)       NO                1               No  24
## 26       Amex          Credit       NO                1               No  18
## 27       Amex          Credit       NO                2               No  17
## 28 Mastercard           Debit      YES                3               No  17
## 29 Mastercard Debit (Prepaid)       NO                2               No  15
## 30   Discover          Credit       NO                1               No  11
## 31       Visa Debit (Prepaid)       NO                1               No  10
## 32       Visa Debit (Prepaid)       NO                2               No  10
## 33   Discover          Credit       NO                2               No   9
## 34 Mastercard          Credit      YES                3               No   9
## 35       Visa          Credit      YES                3               No   8
## 36       Visa Debit (Prepaid)      YES                3               No   5
## 37       Visa           Debit      YES                3               No   4
## 38       Amex          Credit      YES                3               No   3
## 39   Discover          Credit      YES                3               No   3
## 40 Mastercard           Debit       NO                3               No   3
## 41 Mastercard Debit (Prepaid)      YES                3               No   3
## 42 Mastercard          Credit       NO                3               No   1
## 43 Mastercard Debit (Prepaid)       NO                3               No   1
## 44       Visa          Credit       NO                3               No   1
## 45       Visa           Debit       NO                3               No   1
## 46       Visa Debit (Prepaid)       NO                3               No   1
```

Export

```
cards_data <-
  cards_data |>
  arrange(card_brand, card_type, credit_limit)


# Create new file and export
write.csv(cards_data, file = "clean_cards_data.csv")

# Detected any NA values (none)
# Detected any duplicate values (none)
# Rerarranged data by 1) card brand, 2) card type, 3) credit limit
# Note: There are no values for "YES" on card_on_dark_web... data is useless for fraud detection.
```