Data Quality

Task: Draft a document to the client identifying the data quality issues and strategies to mitigate these issues. Refer to ‘Data Quality Framework Table’ and resources below for criteria and dimensions which you should consider.

1. Import dataset

Four datasets were included for this analysis.

Transactions<- read\_excel("KPMG\_VI\_New\_raw\_data\_update\_final.xlsx",

sheet = "Transactions", skip = 1)

CustomerList <- read\_excel("KPMG\_VI\_New\_raw\_data\_update\_final.xlsx",

sheet = "NewCustomerList", skip = 1)

## New names:

## \* `` -> ...17

## \* `` -> ...18

## \* `` -> ...19

## \* `` -> ...20

## \* `` -> ...21

CustomerDemographic <- read\_excel("KPMG\_VI\_New\_raw\_data\_update\_final.xlsx",

sheet = "CustomerDemographic", skip = 1)

CustomerAddress <- read\_excel("KPMG\_VI\_New\_raw\_data\_update\_final.xlsx",

sheet = "CustomerAddress", skip = 1)

2. Data Quality Dimensions AS-IS

Data quality meets six dimensions: accuracy, completeness, consistency, timeliness, relevancy, validity, and uniqueness.

2.1 Accuracy

The term “accuracy” refers to the degree to which information accurately reflects an event or object described.

2.1.1 Transactions

summary(Transactions)

## transaction\_id product\_id customer\_id

## Min. : 1 Min. : 0.00 Min. : 1.0

## 1st Qu.: 5001 1st Qu.: 18.00 1st Qu.: 857.8

## Median :10000 Median : 44.00 Median :1736.0

## Mean :10000 Mean : 45.36 Mean :1738.2

## 3rd Qu.:15000 3rd Qu.: 72.00 3rd Qu.:2613.0

## Max. :20000 Max. :100.00 Max. :5034.0

##

## transaction\_date online\_order order\_status

## Min. :2017-01-01 00:00:00 Mode :logical Length:20000

## 1st Qu.:2017-04-01 00:00:00 FALSE:9811 Class :character

## Median :2017-07-03 00:00:00 TRUE :9829 Mode :character

## Mean :2017-07-01 14:08:05 NA's :360

## 3rd Qu.:2017-10-02 00:00:00

## Max. :2017-12-30 00:00:00

##

## brand product\_line product\_class product\_size

## Length:20000 Length:20000 Length:20000 Length:20000

## Class :character Class :character Class :character Class :character

## Mode :character Mode :character Mode :character Mode :character

##

##

##

##

## list\_price standard\_cost product\_first\_sold\_date

## Min. : 12.01 Min. : 7.21 Min. :33259

## 1st Qu.: 575.27 1st Qu.: 215.14 1st Qu.:35667

## Median :1163.89 Median : 507.58 Median :38216

## Mean :1107.83 Mean : 556.05 Mean :38200

## 3rd Qu.:1635.30 3rd Qu.: 795.10 3rd Qu.:40672

## Max. :2091.47 Max. :1759.85 Max. :42710

## NA's :197 NA's :197

Everything seems to be alright with the type of data of the variables, minimum values, maximum values, and format. Therefore, we assume the accuracy of the datasets given is high.

2.1.2 Customer List

summary(CustomerList)

## first\_name last\_name gender

## Length:1000 Length:1000 Length:1000

## Class :character Class :character Class :character

## Mode :character Mode :character Mode :character

##

##

##

## past\_3\_years\_bike\_related\_purchases DOB job\_title

## Length:1000 Length:1000 Length:1000

## Class :character Class :character Class :character

## Mode :character Mode :character Mode :character

##

##

##

## job\_industry\_category wealth\_segment deceased\_indicator owns\_car

## Length:1000 Length:1000 Length:1000 Length:1000

## Class :character Class :character Class :character Class :character

## Mode :character Mode :character Mode :character Mode :character

##

##

##

## tenure address postcode state

## Min. : 0.00 Length:1000 Length:1000 Length:1000

## 1st Qu.: 7.00 Class :character Class :character Class :character

## Median :11.00 Mode :character Mode :character Mode :character

## Mean :11.39

## 3rd Qu.:15.00

## Max. :22.00

## country property\_valuation ...17 ...18

## Length:1000 Length:1000 Min. :0.4000 Min. :0.4000

## Class :character Class :character 1st Qu.:0.5700 1st Qu.:0.6375

## Mode :character Mode :character Median :0.7500 Median :0.8200

## Mean :0.7473 Mean :0.8390

## 3rd Qu.:0.9200 3rd Qu.:1.0319

## Max. :1.1000 Max. :1.3750

## ...19 ...20 ...21 Rank

## Min. :0.4000 Min. :0.3400 Min. : 1.0 Min. : 1.0

## 1st Qu.:0.7125 1st Qu.:0.6587 1st Qu.: 250.0 1st Qu.: 250.0

## Median :0.9125 Median :0.8426 Median : 500.0 Median : 500.0

## Mean :0.9427 Mean :0.8705 Mean : 498.8 Mean : 498.8

## 3rd Qu.:1.1430 3rd Qu.:1.0625 3rd Qu.: 750.2 3rd Qu.: 750.2

## Max. :1.7188 Max. :1.7188 Max. :1000.0 Max. :1000.0

## Value

## Min. :0.3400

## 1st Qu.:0.6495

## Median :0.8600

## Mean :0.8817

## 3rd Qu.:1.0750

## Max. :1.7188

We see problems with the purchases on bike related purchases, DOB, postcode, and property valuation for having a wrong format. Additionally there is a lack of customer ID, which makes validation of information more difficult between datasets.

2.1.3 Customer Demographics

summary(CustomerDemographic)

## customer\_id first\_name last\_name gender

## Min. : 1 Length:4000 Length:4000 Length:4000

## 1st Qu.:1001 Class :character Class :character Class :character

## Median :2000 Mode :character Mode :character Mode :character

## Mean :2000

## 3rd Qu.:3000

## Max. :4000

##

## past\_3\_years\_bike\_related\_purchases DOB job\_title

## Min. : 0.00 Length:4000 Length:4000

## 1st Qu.:24.00 Class :character Class :character

## Median :48.00 Mode :character Mode :character

## Mean :48.89

## 3rd Qu.:73.00

## Max. :99.00

##

## job\_industry\_category wealth\_segment deceased\_indicator default

## Length:4000 Length:4000 Length:4000 Length:4000

## Class :character Class :character Class :character Class :character

## Mode :character Mode :character Mode :character Mode :character

##

##

##

##

## owns\_car tenure

## Length:4000 Min. : 1.00

## Class :character 1st Qu.: 6.00

## Mode :character Median :11.00

## Mean :10.66

## 3rd Qu.:15.00

## Max. :22.00

## NA's :87

We see problems with, DOB for having a wrong format. However, the most concerning variable is default, since it does not make sense, it is just a group of random characters.

2.1.4 Customer Address

summary(CustomerAddress)

## customer\_id address postcode state

## Min. : 1 Length:3999 Min. :2000 Length:3999

## 1st Qu.:1004 Class :character 1st Qu.:2200 Class :character

## Median :2004 Mode :character Median :2768 Mode :character

## Mean :2004 Mean :2986

## 3rd Qu.:3004 3rd Qu.:3750

## Max. :4003 Max. :4883

## country property\_valuation

## Length:3999 Min. : 1.000

## Class :character 1st Qu.: 6.000

## Mode :character Median : 8.000

## Mean : 7.514

## 3rd Qu.:10.000

## Max. :12.000

Everything seems to be alright with the type of data of the variables, minimum values, maximum values, and format. Therefore, we assume the accuracy of the datasets given is high.

2.2 Completeness

Data is considered “complete” when it fulfills expectations of comprehensiveness. We analyze the missing values in each of the datasets.

2.2.1 Transactions

diagnose(Transactions) %>%

select(-unique\_count, -unique\_rate)%>%

filter(missing\_count != 0)%>%

arrange(desc(missing\_percent))%>%

knitr::kable(align = 'c', format = "markdown")

| **variables** | **types** | **missing\_count** | **missing\_percent** |
| --- | --- | --- | --- |
| online\_order | logical | 360 | 1.800 |
| brand | character | 197 | 0.985 |
| product\_line | character | 197 | 0.985 |
| product\_class | character | 197 | 0.985 |
| product\_size | character | 197 | 0.985 |
| standard\_cost | numeric | 197 | 0.985 |
| product\_first\_sold\_date | numeric | 197 | 0.985 |

The 7 variables displayed above show missing values; however, these values are not significative since the maximum percentage of missing values reach 1.8% at most.

2.2.2 Custumer List

diagnose(CustomerList) %>%

select(-unique\_count, -unique\_rate)%>%

filter(missing\_count != 0)%>%

arrange(desc(missing\_percent))%>%

knitr::kable(align = 'c', format = "markdown")

| **variables** | **types** | **missing\_count** | **missing\_percent** |
| --- | --- | --- | --- |
| job\_title | character | 106 | 10.6 |
| last\_name | character | 29 | 2.9 |
| DOB | character | 17 | 1.7 |

The 3 variables displayed above show missing values, the job\_title variable shows a significant missing percentage with 10.6%; however, this variable might not be critical to the analysis. On the other hand, the other two variables show signifficant completiness since the maximum percentage of missing values reach 2.9% at most.

2.2.3 Custumer Demographics

diagnose(CustomerDemographic) %>%

select(-unique\_count, -unique\_rate)%>%

filter(missing\_count != 0)%>%

arrange(desc(missing\_percent))%>%

knitr::kable(align = 'c', format = "markdown")

| **variables** | **types** | **missing\_count** | **missing\_percent** |
| --- | --- | --- | --- |
| job\_title | character | 506 | 12.650 |
| default | character | 240 | 6.000 |
| last\_name | character | 125 | 3.125 |
| DOB | character | 87 | 2.175 |
| tenure | numeric | 87 | 2.175 |

The 5 variables displayed above show missing values, the job\_title variable shows a significant missing percentage with 12.65%; however, this variable might not be critical to the analysis. On the other hand, the other two variables show signifficant completiness since the maximum percentage of missing values reach 6% at most.

2.2.4 Custumer Address

diagnose(CustomerAddress) %>%

select(-unique\_count, -unique\_rate)%>%

filter(missing\_count != 0)%>%

arrange(desc(missing\_percent))%>%

knitr::kable(align = 'c', format = "markdown")

| **variables** | **types** | **missing\_count** | **missing\_percent** |
| --- | --- | --- | --- |

This means that there are not missing values, therefore the whole dataset is complete.

2.3 Consistency

Consistency refers to having the same data across different datasets.

First, we identify the name of the variables that are repeated accross datasets.

colnames(Transactions)

## [1] "transaction\_id" "product\_id"

## [3] "customer\_id" "transaction\_date"

## [5] "online\_order" "order\_status"

## [7] "brand" "product\_line"

## [9] "product\_class" "product\_size"

## [11] "list\_price" "standard\_cost"

## [13] "product\_first\_sold\_date"

colnames(CustomerList)

## [1] "first\_name" "last\_name"

## [3] "gender" "past\_3\_years\_bike\_related\_purchases"

## [5] "DOB" "job\_title"

## [7] "job\_industry\_category" "wealth\_segment"

## [9] "deceased\_indicator" "owns\_car"

## [11] "tenure" "address"

## [13] "postcode" "state"

## [15] "country" "property\_valuation"

## [17] "...17" "...18"

## [19] "...19" "...20"

## [21] "...21" "Rank"

## [23] "Value"

colnames(CustomerDemographic)

## [1] "customer\_id" "first\_name"

## [3] "last\_name" "gender"

## [5] "past\_3\_years\_bike\_related\_purchases" "DOB"

## [7] "job\_title" "job\_industry\_category"

## [9] "wealth\_segment" "deceased\_indicator"

## [11] "default" "owns\_car"

## [13] "tenure"

colnames(CustomerAddress)

## [1] "customer\_id" "address" "postcode"

## [4] "state" "country" "property\_valuation"

At a first glance it is possible to determine the shared variables between datasets. In particular, Customer List, seems to include most of the variables in Customer Demographics and Customer Address.

2.3.1 Customer List vs Customer Demographic

comparedf(CustomerList, CustomerDemographic)

## Compare Object

##

## Function Call:

## comparedf(x = CustomerList, y = CustomerDemographic)

##

## Shared: 11 non-by variables and 1000 observations.

## Not shared: 14 variables and 3000 observations.

##

## Differences found in 10/10 variables compared.

## 0 variables compared have non-identical attributes.

There are 11 variables that are shared. However, since there is not a user ID, the validation of the consistency will be done through the last name of the user.

summary(comparedf(CustomerList, CustomerDemographic, by="last\_name",

control=comparedf.control(tol.vars = "case")))$diffs.byvar.table %>%

knitr::kable(align = 'c', format = "markdown")

| **var.x** | **var.y** | **n** | **NAs** |
| --- | --- | --- | --- |
| first\_name | first\_name | 3696 | 0 |
| gender | gender | 1906 | 0 |
| DOB | DOB | 3697 | 119 |
| job\_title | job\_title | 3624 | 756 |
| job\_industry\_category | job\_industry\_category | 3142 | 0 |
| wealth\_segment | wealth\_segment | 2439 | 0 |
| deceased\_indicator | deceased\_indicator | 0 | 0 |
| owns\_car | owns\_car | 1845 | 0 |
| tenure | tenure | 3538 | 118 |

The table above shows the number of data that do not match between datasets.

summary(comparedf(CustomerList, CustomerDemographic,

by="last\_name",

tol.char = "case" *#ignores case in character vectors*

))$comparison.summary.table %>%

knitr::kable(align = 'c', format = "markdown")

| **statistic** | **value** |
| --- | --- |
| Number of by-variables | 1 |
| Number of non-by variables in common | 10 |
| Number of variables compared | 9 |
| Number of variables in x but not y | 12 |
| Number of variables in y but not x | 2 |
| Number of variables compared with some values unequal | 8 |
| Number of variables compared with all values equal | 1 |
| Number of observations in common | 3697 |
| Number of observations in x but not y | 899 |
| Number of observations in y but not x | 3803 |
| Number of observations with some compared variables unequal | 101 |
| Number of observations with all compared variables equal | 3596 |
| Number of values unequal | 23887 |

Additionally, there are 3803 observations that are registered in the demographics, but they are not part of the Customer List

2.4 Timeliness

There is no time stamp that shows how the data is managed and the availability of it. Therefore a timeliness parameter cannot be analyzed.

2.5 Relevancy

Some of the variables are not that relevant to some of the datasets. For example the variable default has no relevance in the CustomerDemographic dataset. In the same way, the duplication of data in the CustomerList makes the demographics included there not relevant. Additionally, a CustomerID is missing from this dataset.

2.6 Validity

As seen in the accuracy section some variables do not follow an expected format that they are supposed to, therefore the data is not validated and standardized for these variables.

2.7 Uniqueness

The following sections show the uniqueness value of each variable per dataset.

2.2.1 Transactions

diagnose(Transactions) %>%

select(-missing\_count,-missing\_percent)%>%

arrange(desc(unique\_count))

## # A tibble: 13 x 4

## variables types unique\_count unique\_rate

## <chr> <chr> <int> <dbl>

## 1 transaction\_id numeric 20000 1

## 2 customer\_id numeric 3494 0.175

## 3 transaction\_date POSIXct 364 0.0182

## 4 list\_price numeric 296 0.0148

## 5 standard\_cost numeric 104 0.0052

## 6 product\_id numeric 101 0.00505

## 7 product\_first\_sold\_date numeric 101 0.00505

## 8 brand character 7 0.00035

## 9 product\_line character 5 0.00025

## 10 product\_class character 4 0.0002

## 11 product\_size character 4 0.0002

## 12 online\_order logical 3 0.000150

## 13 order\_status character 2 0.0001

2.2.2 Customer List

diagnose(CustomerList) %>%

select(-missing\_count,-missing\_percent)%>%

arrange(desc(unique\_count))

## # A tibble: 23 x 4

## variables types unique\_count unique\_rate

## <chr> <chr> <int> <dbl>

## 1 address character 1000 1

## 2 last\_name character 962 0.962

## 3 DOB character 962 0.962

## 4 first\_name character 940 0.94

## 5 postcode character 522 0.522

## 6 ...21 numeric 324 0.324

## 7 Rank numeric 324 0.324

## 8 Value numeric 324 0.324

## 9 ...20 numeric 321 0.321

## 10 job\_title character 185 0.185

## # … with 13 more rows

2.2.3 Customer Demographic

diagnose(CustomerDemographic) %>%

select(-missing\_count,-missing\_percent)%>%

arrange(desc(unique\_count))

## # A tibble: 13 x 4

## variables types unique\_count unique\_rate

## <chr> <chr> <int> <dbl>

## 1 customer\_id numeric 4000 1

## 2 last\_name character 3726 0.932

## 3 DOB character 3449 0.862

## 4 first\_name character 3139 0.785

## 5 job\_title character 196 0.049

## 6 past\_3\_years\_bike\_related\_purchases numeric 100 0.025

## 7 default character 93 0.0232

## 8 tenure numeric 23 0.00575

## 9 job\_industry\_category character 10 0.0025

## 10 gender character 6 0.0015

## 11 wealth\_segment character 3 0.00075

## 12 deceased\_indicator character 2 0.0005

## 13 owns\_car character 2 0.0005

2.2.4 Customer List

diagnose(CustomerAddress) %>%

select(-missing\_count,-missing\_percent)%>%

arrange(desc(unique\_count))

## # A tibble: 6 x 4

## variables types unique\_count unique\_rate

## <chr> <chr> <int> <dbl>

## 1 customer\_id numeric 3999 1

## 2 address character 3996 0.999

## 3 postcode numeric 873 0.218

## 4 property\_valuation numeric 12 0.00300

## 5 state character 5 0.00125

## 6 country character 1 0.000250

3. Overall Levels

After analyzing each of the different categories we can determine the level of the data dimension into three cathegories. Red, will mean that there is a lot to improve, and it is a priority to solve before moving into the next stage. Yellow, meaning that it is fairly good and minor changes should be made; and green, meaning that no issues could be found and its ready for the next stage.

Accuracy: Yellow

Completeness: Yellow

Consistency: Red

Timeliness: Red

Relevancy: Yellow

Validity: Yellow

Uniqueness: Green

4. Mitigation measures

Accuracy: As of right now the accuracy of the data is relatively high. Naturally some cleanliness in the data and standardization will increase this accuracy.

Completeness: Most of the data is complete, the factor of missing values is usually non significants. Some of the variables that are not complete, are duplicated with other dataset, so after validating the entrees, the completeness level will increase even more. Additionally, it is necessary to the source of the missing data, if they ar missing at random, they can be ignored for future analysis. If not, they can be predicted with the mean or meadia of the categories missing.

Consistency: This is the main problem of the data. A lot of data is not consistent across datasets. It is recommended to add a Customer ID for this matter in the customer list. There is a significant number of entries that differ across datasets. Mayor cleaning must be performed.

Timeliness: Adding a time stamp would create the timeliness parameter that will allow the company to know when the data is available.

Relevancy: Besides the variable default in the CustomerDemographics dataset, all the variables seem to be relevant.

Validity: Some minor adjustments must be done. Specially on the variables regarding dates. A standardized format must be implemented for future datasets.

Uniqueness: The data seems to have unique entries, however, a better control can be added once all the datasets are dependent on the CustomerID

### Resources to help you with the task

**Data Quality Framework Table**Below is a list of the Data Quality dimensions our team may use to evaluate a dataset. Some of these terms are common to the whole industry, so you may find more information and clarity on these terms by searching online.

