Alerts Analysis

Ricardo Maceiras

Introduction

This dataset shows a simulated sample of alerts generated by specific scenarios will be analyzed, these scenarios were simulated for two types of customers, Private Banking and LC&FI.

The topics which were analyzed:

- Alert Volume and Types
- Alert Resolution Times
- Customer Risk Categories
- PEP Status
- Industry alerts
- Additional Analysis

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(readxl)

Alerts_Dataset <- read_excel("alerts.xlsx")
```

```
## New names:
## * `` -> `...1`
```

```r
Alerts_Dataset
```

```
## # A tibble: 10,177 x 14
##    ...1  intID AlertType   AlertState DateCreated DateClosed CaseOpen CaseClosed
##    <chr> <dbl> <chr>       <chr>      <chr>       <chr>      <chr>    <chr>
##  1 1489   1489 New Destin~ Data Crea~ 2014-04-25~ NULL       2014-05~ 2014-05-1~
##  2 1490   1490 New Destin~ Data Crea~ 2014-04-25~ NULL       2014-05~ 2014-05-1~
##  3 1491   1491 New Destin~ Data Crea~ 2014-04-25~ NULL       2014-05~ 2014-05-1~
```

```
##   4 1492   1492 New Destin~ Data Crea~ 2014-04-29~ NULL       2014-05~ 2014-05-1~
##   5 1493   1493 New Destin~ Data Crea~ 2014-04-29~ NULL       2014-05~ 2014-05-1~
##   6 1494   1494 New Destin~ Data Crea~ 2014-04-29~ NULL       2014-05~ 2014-05-1~
##   7 1495   1495 New Destin~ Data Crea~ 2014-04-29~ NULL       2014-05~ 2014-05-1~
##   8 1496   1496 New Destin~ Data Crea~ 2014-04-29~ NULL       2014-05~ 2014-05-1~
##   9 1497   1497 New Destin~ Data Crea~ 2014-04-29~ NULL       2014-05~ 2014-05-1~
##  10 1498   1498 New Destin~ Data Crea~ 2014-04-29~ NULL       2014-05~ 2014-05-1~
## # i 10,167 more rows
## # i 6 more variables: CaseReported <chr>, CaseState <chr>, PEP <chr>,
## #   CusRiskCategory <chr>, Type <chr>, IndustryCode <chr>
```

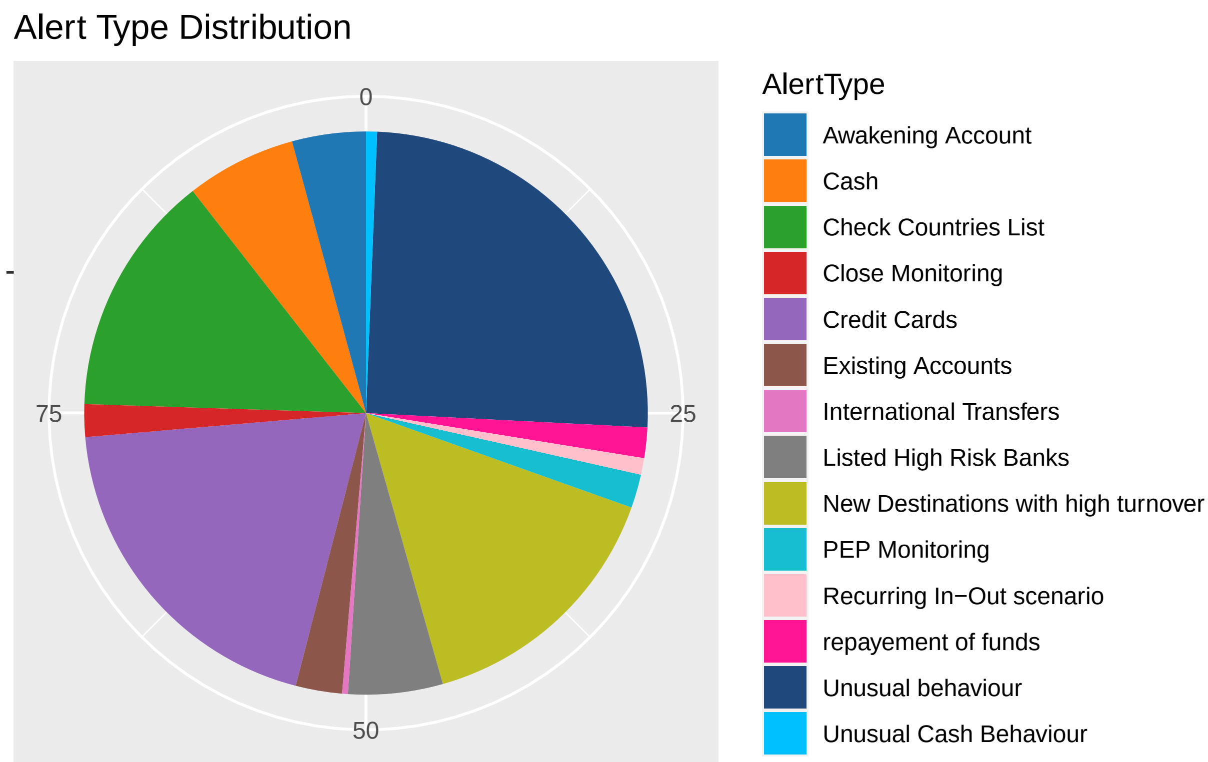
1) Alert Volume and Types

1.a Alert volume

```
alert_typesVOL <- Alerts_Dataset %>%
  select(AlertType) %>%
  group_by(AlertType) %>%
  summarize(Total_values = n()) %>%
  mutate(Percentage = round(Total_values / sum(Total_values) * 100,2)) %>%
  arrange(-Total_values)

alert_typesVOL
```

```
## # A tibble: 14 x 3
##    AlertType                            Total_values Percentage
##    <chr>                                       <int>      <dbl>
##  1 Unusual behaviour                            2563      25.2
##  2 Credit Cards                                 1997      19.6
##  3 New Destinations with high turnover          1543      15.2
##  4 Check Countries List                         1420      14.0
##  5 Cash                                          642       6.31
##  6 Listed High Risk Banks                        552       5.42
##  7 Awakening Account                             430       4.23
##  8 Existing Accounts                             271       2.66
##  9 PEP Monitoring                                196       1.93
## 10 Close Monitoring                              190       1.87
## 11 repayement of funds                           178       1.75
## 12 Recurring In-Out scenario                      97       0.95
## 13 Unusual Cash Behaviour                         64       0.63
## 14 International Transfers                        34       0.33
```

```
colors <- c("#1F77B4", "#FF7F0E", "#2CA02C", "#D62728", "#9467BD", "#8C564B", "#E377C2", "#7F7F7F", "#B

alert_typesVOL_chart <- ggplot(alert_typesVOL, aes(x = "", y = Percentage, fill = AlertType)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar(theta = "y") +
  labs(x = NULL, y = NULL, title = "Alert Type Distribution") +
  scale_fill_manual(values = colors)

alert_typesVOL_chart
```



According to the previous table and chart:

- The alert type "Unusual behavior" has the highest volume of alerts in the sample, approximately 25.2% of the total with 2563 alerts.
- The second and third highest type of alerts are "Credit Cards" and "New Destinations with high turnover" with 1997and 1543 alerts respectively, accounting 19,6% and 15.2% of the total.
- Apparently, the alert types with the lowest numbers of alerts are "Unusual Cash Behaviour" and "International Transfers" with 64 and 34 alerts respectively, accounting 0.63% and 0.33% of the total.

1.b Alert volume: Private Banking vs. LC&FI Analysis

```
Type_status <- Alerts_Dataset %>%
  select(Type, AlertType) %>%
  group_by(Type, AlertType) %>%
  summarize(Total_values = n()) %>%
  mutate(Percentage = round(Total_values / sum(Total_values) * 100,2))
```

```
## `summarise()` has grouped output by 'Type'. You can override using the
## `.groups` argument.
```

```
Type_status
```

```
## # A tibble: 20 x 4
## # Groups:   Type [2]
##    Type  AlertType                          Total_values Percentage
##    <chr> <chr>                                     <int>      <dbl>
##  1 lcfi  Awakening Account                           271       4.28
##  2 lcfi  Cash                                        642      10.1
##  3 lcfi  Check Countries List                       1382      21.8
##  4 lcfi  Close Monitoring                            184       2.9
##  5 lcfi  Credit Cards                               1997      31.5
##  6 lcfi  Existing Accounts                            51       0.8
##  7 lcfi  Listed High Risk Banks                      552       8.71
##  8 lcfi  PEP Monitoring                              168       2.65
##  9 lcfi  Unusual Cash Behaviour                       64       1.01
## 10 lcfi  Unusual behaviour                           848      13.4
## 11 lcfi  repayement of funds                         178       2.81
## 12 pb    Awakening Account                           159       4.14
## 13 pb    Check Countries List                         38       0.99
## 14 pb    Close Monitoring                              6       0.16
## 15 pb    Existing Accounts                           220       5.73
## 16 pb    International Transfers                      34       0.89
## 17 pb    New Destinations with high turnover        1543      40.2
## 18 pb    PEP Monitoring                               28       0.73
## 19 pb    Recurring In-Out scenario                    97       2.53
## 20 pb    Unusual behaviour                          1715      44.7
```

```
Type_status_arrange <- Alerts_Dataset %>%
  select(Type, AlertType) %>%
  group_by(Type, AlertType) %>%
  summarize(Total_values = n()) %>%
  mutate(Percentage = round(Total_values / sum(Total_values) * 100,2)) %>%
  arrange(-Total_values)
```

```
## `summarise()` has grouped output by 'Type'. You can override using the
## `.groups` argument.
```

```
Type_status_arrange
```

```
## # A tibble: 20 x 4
## # Groups:   Type [2]
##    Type  AlertType                          Total_values Percentage
##    <chr> <chr>                                     <int>      <dbl>
##  1 lcfi  Credit Cards                               1997      31.5
##  2 pb    Unusual behaviour                          1715      44.7
##  3 pb    New Destinations with high turnover        1543      40.2
##  4 lcfi  Check Countries List                       1382      21.8
##  5 lcfi  Unusual behaviour                           848      13.4
##  6 lcfi  Cash                                        642      10.1
##  7 lcfi  Listed High Risk Banks                      552       8.71
##  8 lcfi  Awakening Account                           271       4.28
##  9 pb    Existing Accounts                           220       5.73
## 10 lcfi  Close Monitoring                            184       2.9
## 11 lcfi  repayement of funds                         178       2.81
## 12 lcfi  PEP Monitoring                              168       2.65
```

```
## 13 pb    Awakening Account                                  159        4.14
## 14 pb    Recurring In-Out scenario                           97        2.53
## 15 lcfi  Unusual Cash Behaviour                              64        1.01
## 16 lcfi  Existing Accounts                                   51        0.8
## 17 pb    Check Countries List                                38        0.99
## 18 pb    International Transfers                             34        0.89
## 19 pb    PEP Monitoring                                      28        0.73
## 20 pb    Close Monitoring                                     6        0.16
```

According to the previous records:

- The highest types of alerts when the customer is Private Banking are "Unusual behaviour", "New Destinations with high turnover", " Existing Accounts" with 1715, 1543 and 220 alerts respectively, accounting 44.7%, 40.2% abd 5.73% of the total within the group.
- The highest types of alerts when the customer is LC&FI are "Credit Cards", "Check Countries List", "Unusual behaviour" with 1997, 1382, 848 alerts, accounting 31.5%, 21.8%, 13.4% of the total within the group.
- "Credit Cards" alerts occur when the customer is LC&FI.
- "Unusual behavior" alerts are the type with most appearances in both Private Banking and LC&FI.

2) Alert Resolution Times

The elapsed time from the date when 2nd line started the investigation to the Date when 2d line closed the investigation will be analyzed

```
Alerts_Dataset$CaseOpen <- as.POSIXct(Alerts_Dataset$CaseOpen, format="%Y-%m-%d %H:%M:%S")
Alerts_Dataset$CaseClosed <- as.POSIXct(Alerts_Dataset$CaseClosed, format="%Y-%m-%d %H:%M:%S")
Alerts_Dataset$CaseReported <- as.POSIXct(Alerts_Dataset$CaseReported, format="%Y-%m-%d %H:%M:%S")
```

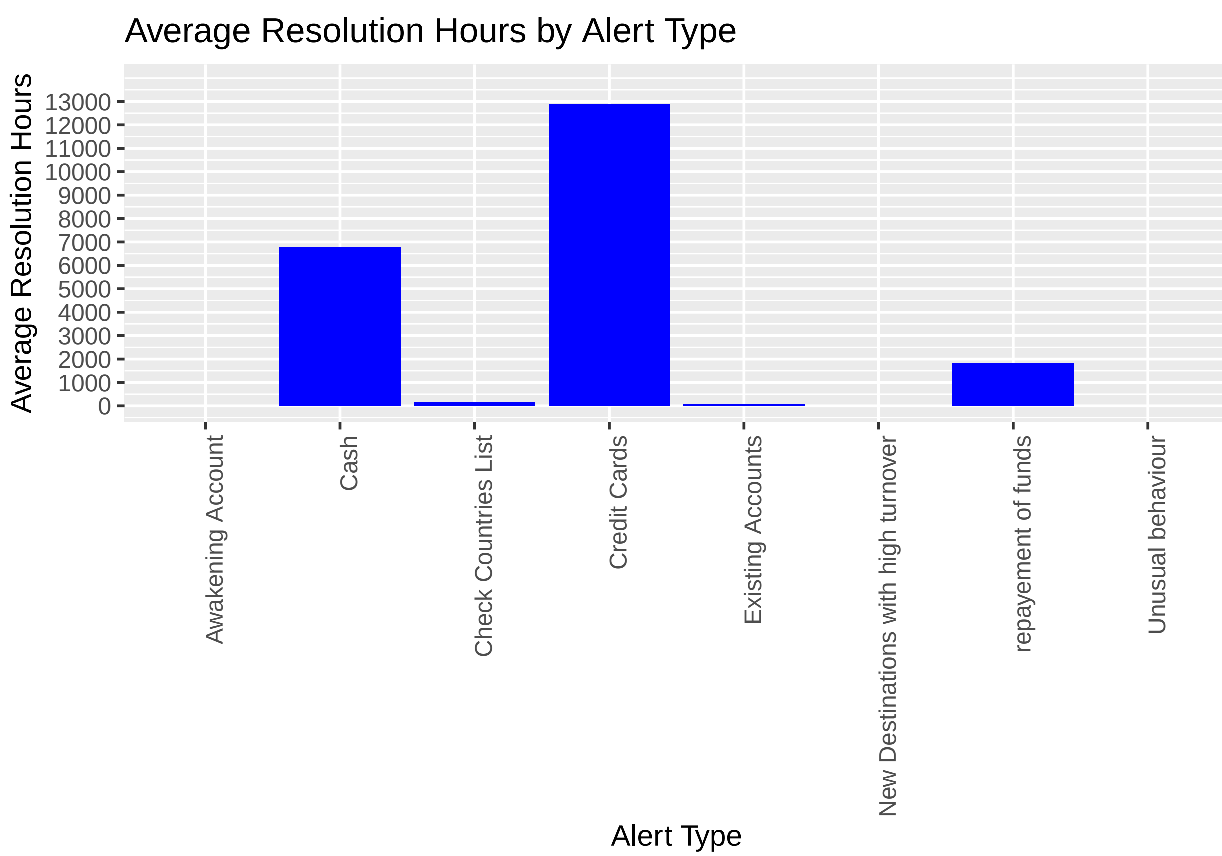
2.a Alert Resolution Times: Case open - Case closed

```
alert_times_CASOP_CASCL <- Alerts_Dataset %>%
  select(AlertType, CaseOpen, CaseClosed, Type) %>%
  mutate(Resolution_hours = round(difftime(CaseClosed, CaseOpen, units = "hours"),2)) %>%
  drop_na()

alert_times_CASOP_CASCL_Summary <- alert_times_CASOP_CASCL %>%
  select(AlertType,Resolution_hours) %>%
  group_by(AlertType) %>%
  summarize(Average_resolution_hours= round(mean(Resolution_hours),2)) %>%
  arrange(-Average_resolution_hours)

alert_times_CASOP_CASCL_Summary
```

```
## # A tibble: 8 x 2
##   AlertType                           Average_resolution_hours
##   <chr>                               <drtn>
## 1 Credit Cards                        12887.43 hours
```

```
## 2 Cash                                  6799.32 hours
## 3 repayement of funds                   1827.00 hours
## 4 Check Countries List                   139.51 hours
## 5 Existing Accounts                       62.26 hours
## 6 New Destinations with high turnover      0.86 hours
## 7 Unusual behaviour                        0.17 hours
## 8 Awakening Account                        0.03 hours
```

```
max_value <- as.numeric(max(alert_times_CASOP_CASCL_Summary$Average_resolution_hours)) + 1000

alert_times_CASOP_CASCL_Summary_chart <- ggplot(alert_times_CASOP_CASCL_Summary, aes(x = AlertType, y =
  geom_bar(stat = "identity", fill = "blue") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  labs(x = "Alert Type", y = "Average Resolution Hours", title = "Average Resolution Hours by Alert Typ
  scale_y_continuous(limits = c(0, max_value), breaks = seq(0, max_value, by = 1000))

alert_times_CASOP_CASCL_Summary_chart
```



According to the previous table and chart:

- Apparently, the alert types which investigation takes the most time to be closed when 2nd line start it are "Credit Cards" alerts which might take 12887.43 hour on average and "Cash" alerts with 6799.32 hours on average.
- Otherwise, the alert types which investigation takes the lowest period of time to be closed by 2nd libe are "Unusual behaviour" which might take 0.17 hours on average and Awakening Account with 0.03 hours on average.

- "Unusual behaviour" and "Credit Cards" are the type with the largest number of alerts, being "Unusual behaviour" cases resolved quickly and efficiently and "Credit Cards" cases not as efficient as others.

2.a.a Alert Resolution Times: Private Banking vs. LC&FI Analysis

```
Types_CASOP_CASCL_Summary <- alert_times_CASOP_CASCL %>%
  select(Type, AlertType,Resolution_hours) %>%
  group_by(Type, AlertType) %>%
  summarize(Average_resolution_hours= round(mean(Resolution_hours),2))
```

```
## `summarise()` has grouped output by 'Type'. You can override using the
## `.groups` argument.
```

```
Types_CASOP_CASCL_Summary
```

```
## # A tibble: 11 x 3
## # Groups:   Type [2]
##    Type  AlertType                        Average_resolution_hours
##    <chr> <chr>                            <drtn>
##  1 lcfi  Awakening Account                    0.03 hours
##  2 lcfi  Cash                              6799.32 hours
##  3 lcfi  Check Countries List               145.14 hours
##  4 lcfi  Credit Cards                     12887.43 hours
##  5 lcfi  Existing Accounts                    0.16 hours
##  6 lcfi  Unusual behaviour                    0.05 hours
##  7 lcfi  repayement of funds               1827.00 hours
##  8 pb    Check Countries List                77.55 hours
##  9 pb    Existing Accounts                  155.42 hours
## 10 pb    New Destinations with high turnover    0.86 hours
## 11 pb    Unusual behaviour                    0.30 hours
```

According to the table:

- Usually the cases of Private Banking are resolved faster and more efficiently than cases of LC&FI by the second line once opened the investigation.

2.b Alert Resolution Times: Case open - Case Reported

```
alert_times_CASOP_CASRE <- Alerts_Dataset %>%
  select(AlertType, CaseOpen, CaseReported,Type) %>%
  mutate(Resolution_hours = round(difftime(CaseReported, CaseOpen, units = "hours"),2)) %>%
  drop_na()

alert_times_CASOP_CASRE_Summary <- alert_times_CASOP_CASRE %>%
  select(AlertType,Resolution_hours) %>%
  group_by(AlertType) %>%
  summarize(Average_resolution_hours= round(mean(Resolution_hours),2)) %>%
  arrange(-Average_resolution_hours)

alert_times_CASOP_CASRE_Summary
```

```
## # A tibble: 5 x 2
##   AlertType               Average_resolution_hours
##   <chr>                   <drtn>
## 1 Listed High Risk Banks 172.42 hours
## 2 Cash                   125.81 hours
## 3 Credit Cards            59.98 hours
## 4 Check Countries List     8.25 hours
## 5 Unusual behaviour        0.05 hours
```

```
max_value2 <- as.numeric(max(alert_times_CASOP_CASRE_Summary$Average_resolution_hours)) + 10

alert_times_CASOP_CASRE_Summary_chart <- ggplot(alert_times_CASOP_CASRE_Summary, aes(x = AlertType, y =
  geom_bar(stat = "identity", fill = "blue") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  labs(x = "Alert Type", y = "Average Resolution Hours", title = "Average Resolution Hours by Alert Typ
  scale_y_continuous(limits = c(0, max_value2), breaks = seq(0, max_value2, by = 10))

alert_times_CASOP_CASRE_Summary_chart
```


```
Types_CASOP_CASRE_Summary <- alert_times_CASOP_CASRE %>%
  select(Type, AlertType,Resolution_hours) %>%
  group_by(Type, AlertType) %>%
  summarize(Average_resolution_hours= round(mean(Resolution_hours),2))
```

```
## `summarise()` has grouped output by 'Type'. You can override using the
## `.groups` argument.
```

```r
Types_CASOP_CASRE_Summary
```

```
## # A tibble: 6 x 3
## # Groups:   Type [2]
##   Type  AlertType              Average_resolution_hours
##   <chr> <chr>                  <drtn>
## 1 lcfi  Cash                   125.81 hours
## 2 lcfi  Check Countries List     9.62 hours
## 3 lcfi  Credit Cards            59.98 hours
## 4 lcfi  Listed High Risk Banks 172.42 hours
## 5 pb    Check Countries List     0.05 hours
## 6 pb    Unusual behaviour        0.05 hours
```

According to the previous table and chart:

- "Listed High Risk Banks" and "Cash" are the alert types that take the most time to be sent to FIU with 172.42 and 125.81 hours on average respectively. Nonetheless, "Unusual behaviour" alerts are sent in 0.05 hours on average.
- Most of the alerts reported to FIU occurred when the customer is LC&FI. Moreover, LC&FI alerts take much longer hours than Private Banking alerts.

2.c Alert Resolution Times: First line to Second line

```r
sum_na= sum(is.na(Alerts_Dataset$CaseOpen))

alert_times_FIRST_SECOND <- Alerts_Dataset %>%
  group_by(AlertType) %>%
  summarize(Percentage_alerts = round(mean(is.na(CaseOpen)) * 100,2)) %>%
  arrange(desc(Percentage_alerts))

alert_times_FIRST_SECOND
```

```
## # A tibble: 14 x 2
##    AlertType                             Percentage_alerts
##    <chr>                                             <dbl>
##  1 Close Monitoring                                  100
##  2 International Transfers                           100
##  3 PEP Monitoring                                    100
##  4 Recurring In-Out scenario                         100
##  5 Unusual Cash Behaviour                            100
##  6 Awakening Account                                  99.5
##  7 Unusual behaviour                                  99.0
##  8 Listed High Risk Banks                             98.9
##  9 New Destinations with high turnover                98.6
## 10 Existing Accounts                                  98.2
## 11 repayement of funds                                97.8
## 12 Cash                                               97.0
## 13 Check Countries List                               96.1
## 14 Credit Cards                                       94.2
```

- The vast majority of the alerts pass from first line to be investigated by second line.
- 94.2 % of "Credit Cards" alerts pass to second line, being the lowest ratio among alert types.

3) Customer Risk Categories

```
Risk_categories <- Alerts_Dataset %>%
  select(CusRiskCategory) %>%
  group_by(CusRiskCategory) %>%
  summarize(Total_values = n()) %>%
  mutate(Percentage = round(Total_values / sum(Total_values) * 100,2)) %>%
  arrange(-Total_values)

Risk_categories
```

```
## # A tibble: 5 x 3
##   CusRiskCategory Total_values Percentage
##   <chr>                  <int>      <dbl>
## 1 Medium Risk             6986      68.6
## 2 Lower Risk              1326      13.0
## 3 Higher Risk             1183      11.6
## 4 Not Specified            628       6.17
## 5 NULL                      54       0.53
```

- 0,53% of the total are NULL values, therefore, can be omitted

3.a Customer Risk Categories: Summary of risk categories and alert type

```
Risk_categories_AlertyType <- Alerts_Dataset %>%
  select(AlertType, CusRiskCategory) %>%
  filter(CusRiskCategory != "NULL") %>%
  group_by(AlertType, CusRiskCategory) %>%
  summarize(Total_values = n()) %>%
  mutate(Percentage = round(Total_values / sum(Total_values) * 100, 2))
```

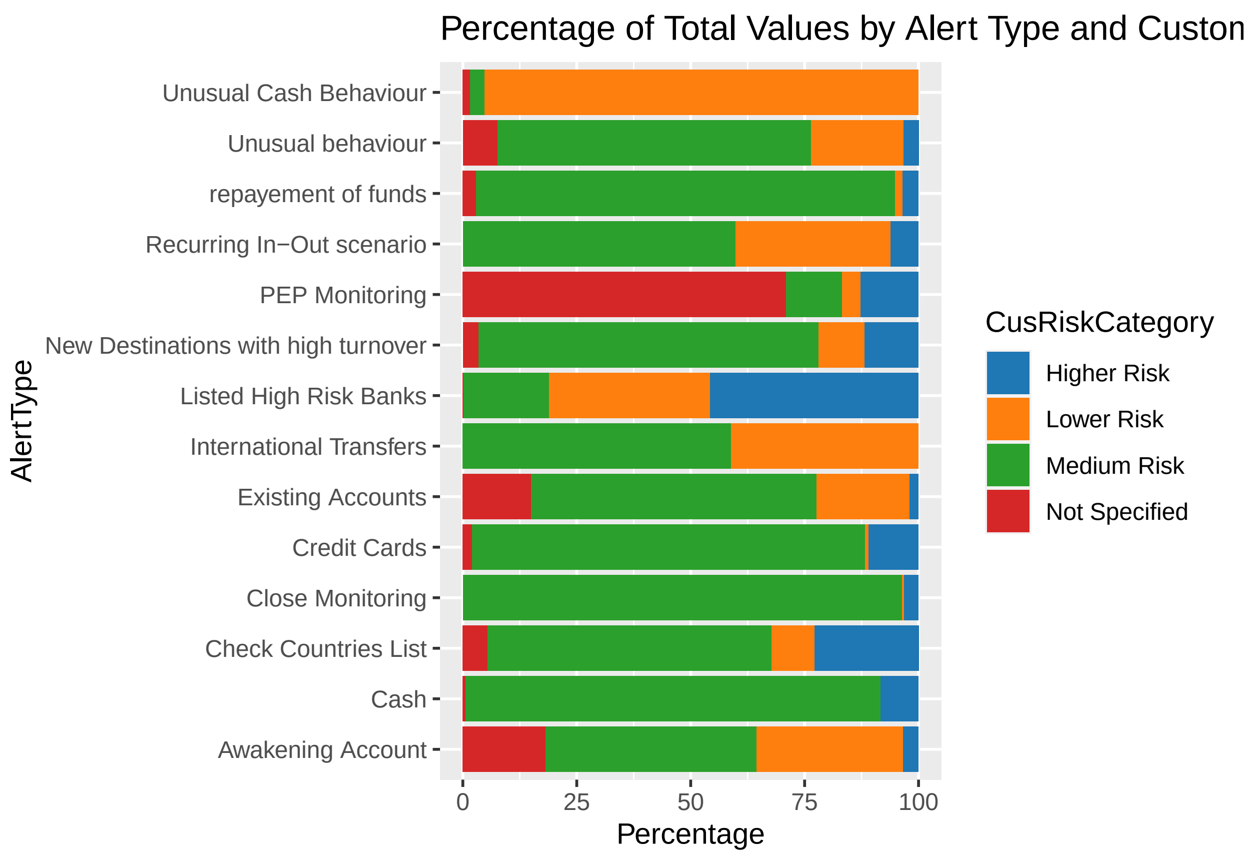
```
## `summarise()` has grouped output by 'AlertType'. You can override using the
## `.groups` argument.
```

```
Risk_categories_AlertyType
```

```
## # A tibble: 50 x 4
## # Groups:   AlertType [14]
##    AlertType             CusRiskCategory Total_values Percentage
##    <chr>                 <chr>                  <int>      <dbl>
##  1 Awakening Account     Higher Risk               15       3.49
##  2 Awakening Account     Lower Risk               138      32.1
##  3 Awakening Account     Medium Risk              199      46.3
##  4 Awakening Account     Not Specified             78      18.1
##  5 Cash                  Higher Risk               54       8.41
```

```
##   6 Cash                    Medium Risk          584       91.0
##   7 Cash                    Not Specified          4        0.62
##   8 Check Countries List Higher Risk          324       22.8
##   9 Check Countries List Lower Risk           134        9.44
## 10 Check Countries List Medium Risk          884       62.3
## # i 40 more rows
```

```
colors <- c("#1F77B4", "#FF7F0E", "#2CA02C", "#D62728", "#9467BD", "#8C564B", "#E377C2", "#7F7F7F", "#B

Risk_categories_AlertyType_chart <- ggplot(Risk_categories_AlertyType, aes(x = Percentage, y = AlertTyp
  geom_bar(stat = "identity", position = "stack") +
  labs(x = "Percentage", y = "AlertType", title = "Percentage of Total Values by Alert Type and Custome
  scale_fill_manual(values = colors)

Risk_categories_AlertyType_chart
```



According to the previous chart and table:

- Most of the alert types are categorized as "Medium Risk" with the exception of "Unusual cash behaviour" which most of their alerts are "Lower Risk" and "Pep monitoring" where most of their alerts are not specified, therefore, hindering the sample.

3.a Customer Risk Categories: Summary of risk categories and type company

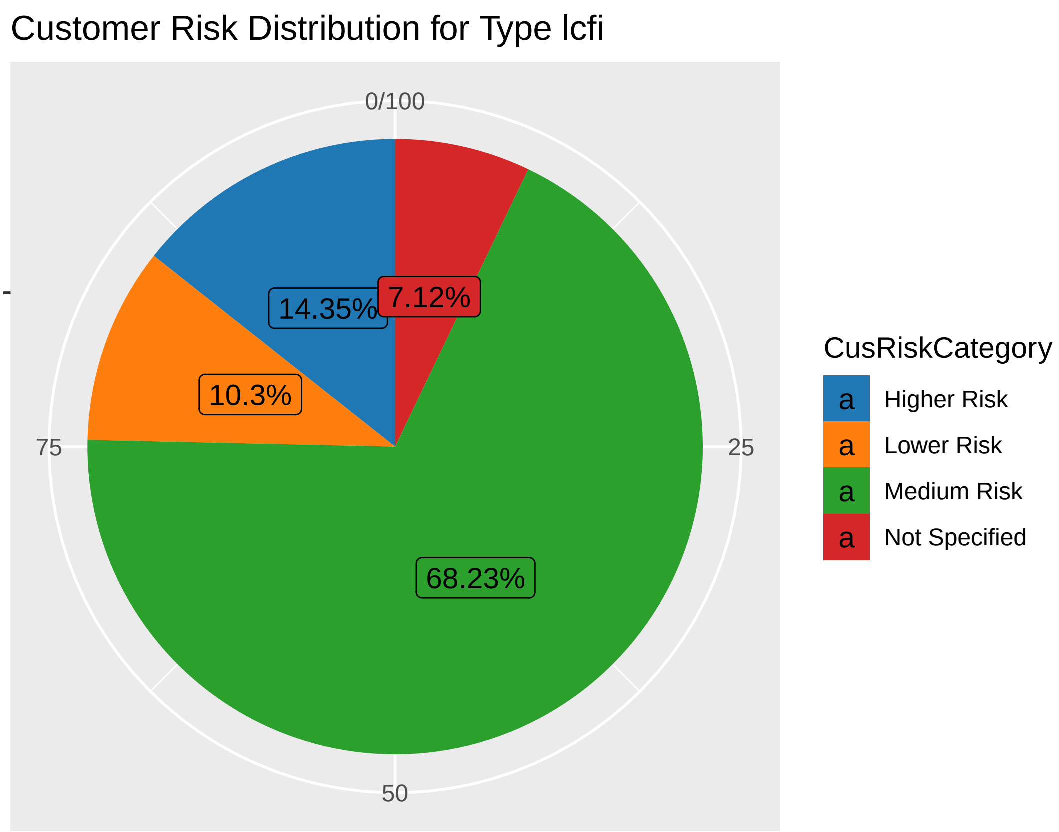
```
Risk_categories_Type <- Alerts_Dataset %>%
  select(Type, CusRiskCategory) %>%
  filter(CusRiskCategory != "NULL") %>%
  group_by(Type, CusRiskCategory) %>%
  summarize(Total_values = n()) %>%
  mutate(Percentage = round(Total_values / sum(Total_values) * 100, 2))
```

```
## `summarise()` has grouped output by 'Type'. You can override using the
## `.groups` argument.
```

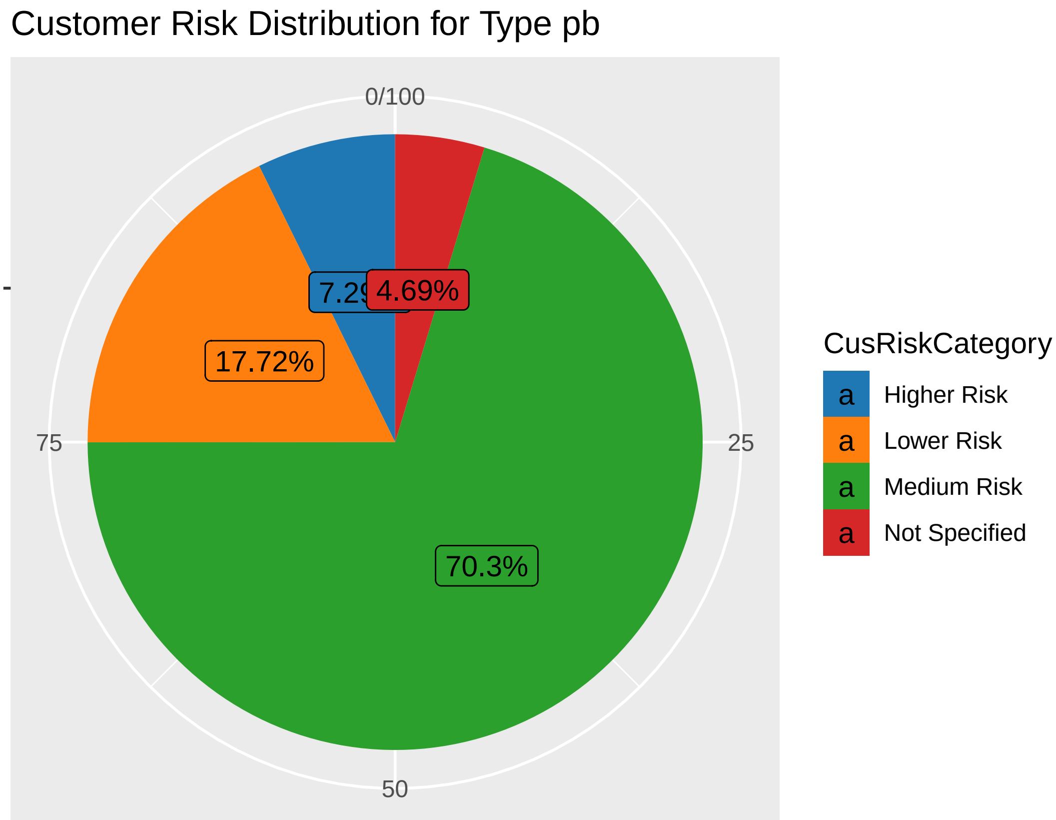
```
Risk_categories_Type
```

```
## # A tibble: 8 x 4
## # Groups:   Type [2]
##   Type  CusRiskCategory Total_values Percentage
##   <chr> <chr>                  <int>      <dbl>
## 1 lcfi  Higher Risk              905      14.4
## 2 lcfi  Lower Risk               650      10.3
## 3 lcfi  Medium Risk             4304      68.2
## 4 lcfi  Not Specified            449       7.12
## 5 pb    Higher Risk              278       7.29
## 6 pb    Lower Risk               676      17.7
## 7 pb    Medium Risk             2682      70.3
## 8 pb    Not Specified            179       4.69
```

```
Risk_Categories_lcfi_chart <- ggplot(Risk_categories_Type%>% filter(Type == "lcfi"), aes(x = "", y = Pe
  geom_bar(stat = "identity", width = 1) +
  coord_polar(theta = "y") +
  labs(x = NULL, y = NULL, title = "Customer Risk Distribution for Type lcfi") +
  scale_fill_manual(values = colors)+
  geom_label(aes(label = paste0(Percentage, "%")), position = position_stack(vjust = 0.5))

Risk_Categories_lcfi_chart
```




```
Risk_Categories_pb_chart <- ggplot(Risk_categories_Type%>% filter(Type == "pb"), aes(x = "", y = Percen
  geom_bar(stat = "identity", width = 1) +
  coord_polar(theta = "y") +
  labs(x = NULL, y = NULL, title = "Customer Risk Distribution for Type pb") +
  scale_fill_manual(values = colors)+
  geom_label(aes(label = paste0(Percentage, "%")), position = position_stack(vjust = 0.5))

Risk_Categories_pb_chart
```



- Most of the alerts of Private Banking and LC&FI are categorized as "Medium Risk".
- Private Banking has more "Lower Risk" alerts and LC&FI has more "Higher Risk" alerts.

4) PEP Status

```
Pep_status <- Alerts_Dataset %>%
  select(PEP) %>%
  group_by(PEP) %>%
  summarize(Total_values = n()) %>%
  mutate(Percentage = round(Total_values / sum(Total_values) * 100,2)) %>%
  arrange(-Total_values)

Pep_status
```

```
## # A tibble: 4 x 3
##   PEP   Total_values Percentage
##   <chr>        <int>      <dbl>
## 1 <NA>          6337      62.3
## 2 N             2784      27.4
## 3 NULL           938       9.22
## 4 Y              118       1.16
```

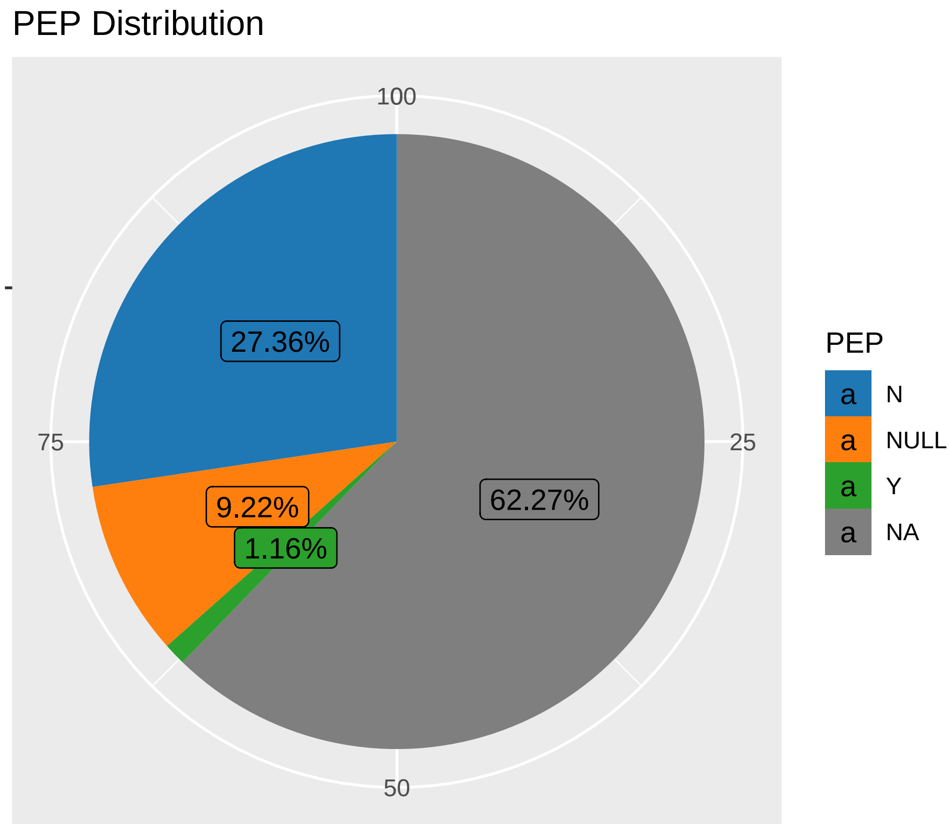
```r
Pep_status_AlertyType <- Alerts_Dataset %>%
  select(AlertType, PEP) %>%
  mutate(PEP = ifelse(is.na(PEP), "Missing Value", PEP)) %>%
  group_by(AlertType, PEP) %>%
  summarize(Total_values = n()) %>%
  mutate(Percentage = round(Total_values / sum(Total_values) * 100, 2))
```

```
## `summarise()` has grouped output by 'AlertType'. You can override using the
## `.groups` argument.
```

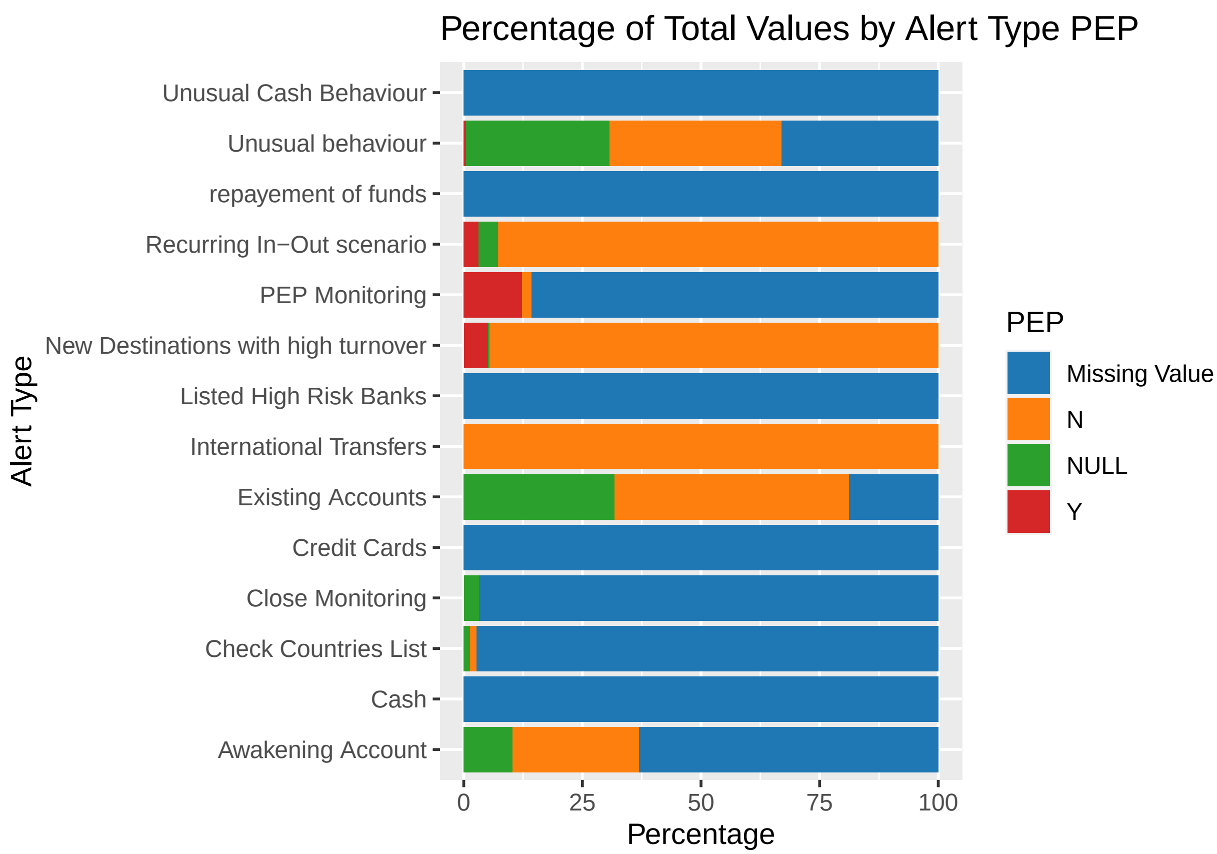
```r
Pep_status_AlertyType
```

```
## # A tibble: 30 x 4
## # Groups:   AlertType [14]
##    AlertType            PEP           Total_values Percentage
##    <chr>                <chr>                <int>      <dbl>
##  1 Awakening Account    Missing Value          271      63.0
##  2 Awakening Account    N                      115      26.7
##  3 Awakening Account    NULL                    44      10.2
##  4 Cash                 Missing Value          642     100
##  5 Check Countries List Missing Value         1382      97.3
##  6 Check Countries List N                       20       1.41
##  7 Check Countries List NULL                    18       1.27
##  8 Close Monitoring     Missing Value          184      96.8
##  9 Close Monitoring     NULL                     6       3.16
## 10 Credit Cards         Missing Value         1997     100
## # i 20 more rows
```

```r
Pep_status_chart <- ggplot(Pep_status, aes(x = "", y = Percentage, fill = PEP)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar(theta = "y") +
  labs(x = NULL, y = NULL, title = "PEP Distribution") +
  scale_fill_manual(values = colors)+
  geom_label(aes(label = paste0(Percentage, "%")), position = position_stack(vjust = 0.5))

Pep_status_chart
```




```
Pep_status_AlertyType_chart <- ggplot(Pep_status_AlertyType, aes(x = Percentage, y = AlertType, fill =
  geom_bar(stat = "identity", position = "stack") +
  labs(x = "Percentage", y = "Alert Type", title = "Percentage of Total Values by Alert Type PEP") +
  scale_fill_manual(values = colors)

Pep_status_AlertyType_chart
```



According to the charts and tables:

- There is a problem of biased data, 62.27 of the values are NA and 9.22% are NULL, which difficult the analyzis, NA values cannot be removed, otherwisde the results would not be representative within the sample.
- However, "New Destinations with high turnover", "Recurring In-Out scenario","International Transfers" alert types can be analyzed, being the majority of their alerts, negative PEP.

5) Industry alerts

```
Industry_code <-Alerts_Dataset %>%
  select(IndustryCode) %>%
  filter(IndustryCode != "NULL" & IndustryCode != 0) %>%
  group_by(IndustryCode) %>%
  summarize(Total_values = n()) %>%
  mutate(Percentage = round(Total_values / sum(Total_values) * 100, 2)) %>%
  arrange(-Total_values)

## adding other dataset
additional_info <- read_excel("additional_info.xlsx", sheet = 2)

Additional_info_edit <- additional_info %>%
  rename(IndustryCode = "Industry Code",RiskScore= "Risk Score")
```

```r
Industry_risk_leftjoin <- left_join(Industry_code, Additional_info_edit, by = "IndustryCode")
Industry_risk_leftjoin
```

```
## # A tibble: 206 x 5
##    IndustryCode Total_values Percentage RiskScore Segment
##    <chr>               <int>      <dbl>     <dbl> <chr>
##  1 64190                 376      17.1         NA <NA>
##  2 55101                 317      14.4         NA <NA>
##  3 43999                  71       3.24        NA <NA>
##  4 20590                  58       2.64        NA <NA>
##  5 96090                  58       2.64        NA <NA>
##  6 52290                  53       2.42        NA <NA>
##  7 28990                  48       2.19        NA <NA>
##  8 28290                  47       2.14        NA <NA>
##  9 69103                  47       2.14        NA <NA>
## 10 66190                  43       1.96       100 high risk
## # i 196 more rows
```

```r
Industry_risk <- inner_join(Industry_code, Additional_info_edit, by = "IndustryCode")

Industry_risk_General <- inner_join(Alerts_Dataset, Additional_info_edit, by = "IndustryCode")

Industry_risk_General
```

```
## # A tibble: 80 x 16
##    ...1   intID AlertType  AlertState DateCreated DateClosed CaseOpen
##    <chr>  <dbl> <chr>      <chr>      <chr>       <chr>      <dttm>
##  1 20045 172010 Check Cou~ Data Crea~ 2021-02-25~ NULL       2021-04-22 12:14:58
##  2 20337 172302 Check Cou~ Data Crea~ 2021-03-10~ NULL       2021-03-24 12:24:14
##  3 20412 172377 Unusual b~ Data Crea~ 2021-03-17~ NULL       2021-04-15 09:25:05
##  4 20415 172380 Check Cou~ Data Crea~ 2021-03-17~ NULL       2021-05-10 10:44:01
##  5 20044 172009 Check Cou~ Closed - ~ 2021-02-25~ 2021-03-0~ NA
##  6 20378 172343 Check Cou~ Closed - ~ 2021-03-13~ 2021-03-1~ NA
##  7 20396 172361 Check Cou~ Closed - ~ 2021-03-14~ 2021-03-1~ NA
##  8 20028 171993 Unusual b~ Closed - ~ 2021-02-25~ 2021-03-0~ NA
##  9 17599  19422 Listed Hi~ Closed - ~ 2018-03-15~ 2018-03-1~ NA
## 10 11258   3068 Awakening~ Closed - ~ 2014-03-06~ 2014-04-0~ NA
## # i 70 more rows
## # i 9 more variables: CaseClosed <dttm>, CaseReported <dttm>, CaseState <chr>,
## #   PEP <chr>, CusRiskCategory <chr>, Type <chr>, IndustryCode <chr>,
## #   RiskScore <dbl>, Segment <chr>
```

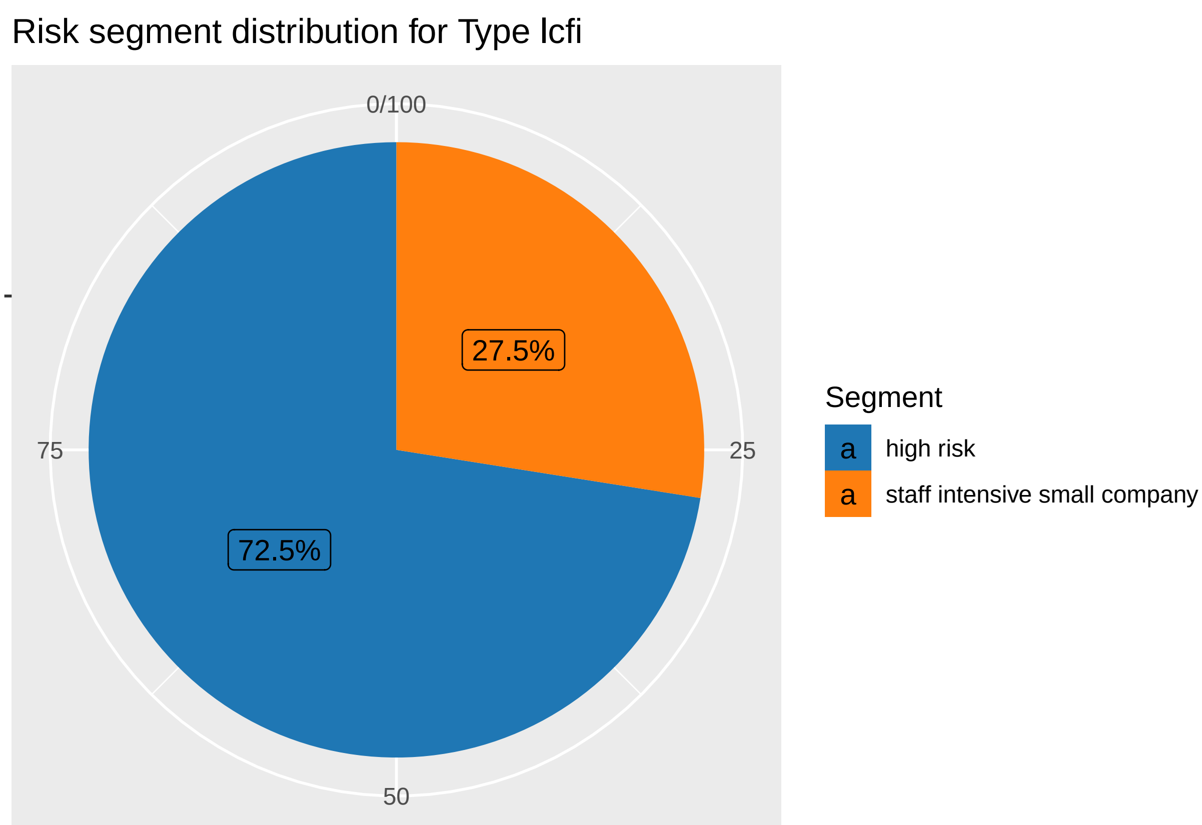
```r
Risk_segment <- Industry_risk_General %>%
  select(Type, Segment) %>%
  group_by(Type, Segment) %>%
  summarize(Total_values = n()) %>%
  mutate(Percentage = round(Total_values / sum(Total_values) * 100, 2))
```

```
## `summarise()` has grouped output by 'Type'. You can override using the
## `.groups` argument.
```

```
Risk_segment
```

```
## # A tibble: 2 x 4
## # Groups:   Type [1]
##   Type  Segment                           Total_values Percentage
##   <chr> <chr>                                    <int>      <dbl>
## 1 lcfi  high risk                                   58       72.5
## 2 lcfi  staff intensive small company               22       27.5
```

```
Risk_segment_chart <- ggplot(Risk_segment%>% filter(Type == "lcfi"), aes(x = "", y = Percentage, fill =
  geom_bar(stat = "identity", width = 1) +
  coord_polar(theta = "y") +
  labs(x = NULL, y = NULL, title = "Risk segment distribution for Type lcfi") +
  scale_fill_manual(values = colors)+
  geom_label(aes(label = paste0(Percentage, "%")), position = position_stack(vjust = 0.5))

Risk_segment_chart
```



Observations:

- The majority of the alerts are not categorized by industry code and segment. Moreover, he metadata does not contain all industry codes of the data set.
- There were only 80 values that matched with the Industry code, being all of them LC&FI with "High Risk"

6) Additional Analysis

What analysis you think should be included in the qualitative validation part of Transaction Monitoring model? What areas of TM models are essential to analyse?.

Analysis to be included:

- PEP details: add information on the role or position held by a PEP so that we can understand the risk associated to this person in concern.
- PEP exposure in High risk countries: Determine whether the PEP has connections or businesses in regions associated to illicit activities
- Regional Risk Assessment: Analyze each industry and where their operations are held in as well as the exposure to high-risk jurisdictions.

Essential areas:

I consider important the analysis of PEP as well as Industry risk. Nonetheless, during the analysis i found these observations:

- Most of the values in this PEP model are missing values or NULL, instead of Non PEP/PEP. There should be a better mapping.
- Most of the industry codes are missing values and the meta data (additional information) does not map all the codes, therefore, it biases the sample.