dataframe for analysis

Irina Klein

10/31/2021

Data uploading and cleaning

```
setwd("~/CSP571ProjectGroup")
incident_event_log <- read_csv("incident_event_log_difftime.csv", na = c("?", "NA"))
```

```
## New names:
## * `` -> ...1
## * ...1 -> ...2

## Rows: 138565 Columns: 40

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (23): number, incident_state, caller_id, opened_by, sys_created_by, sys...
## dbl   (7): ...1, ...2, reassignment_count, reopen_count, sys_mod_count, time...
## lgl   (5): active, made_sla, knowledge, u_priority_confirmation, caused_by
## dttm  (5): opened_at, sys_created_at, sys_updated_at, resolved_at, closed_at

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
print(incident_event_log,width = 1000,n = 5)
```

```
## # A tibble: 138,565 x 40
##    ...1  ...2 number     incident_state active reassignment_count reopen_count
##   <dbl> <dbl> <chr>      <chr>          <lgl>               <dbl>        <dbl>
## 1     1     1 INC0000045 New            TRUE                    0            0
## 2     2     2 INC0000045 Resolved       TRUE                    0            0
## 3     3     3 INC0000045 Resolved       TRUE                    0            0
## 4     4     4 INC0000045 Closed         FALSE                   0            0
## 5     5     5 INC0000047 New            TRUE                    0            0
##   sys_mod_count made_sla caller_id   opened_by      opened_at
##           <dbl> <lgl>    <chr>       <chr>          <dttm>
## 1             0 TRUE     Caller 2403 Opened by  8   2016-02-29 01:16:00
## 2             2 TRUE     Caller 2403 Opened by  8   2016-02-29 01:16:00
## 3             3 TRUE     Caller 2403 Opened by  8   2016-02-29 01:16:00
## 4             4 TRUE     Caller 2403 Opened by  8   2016-02-29 01:16:00
## 5             0 TRUE     Caller 2403 Opened by  397 2016-02-29 04:40:00
##   sys_created_by sys_created_at      sys_updated_by sys_updated_at
##   <chr>          <dttm>              <chr>          <dttm>
## 1 Created by 6   2016-02-29 01:23:00 Updated by 21  2016-02-29 01:23:00
## 2 Created by 6   2016-02-29 01:23:00 Updated by 642 2016-02-29 08:53:00
## 3 Created by 6   2016-02-29 01:23:00 Updated by 804 2016-02-29 11:29:00
## 4 Created by 6   2016-02-29 01:23:00 Updated by 908 2016-03-05 12:00:00
## 5 Created by 171 2016-02-29 04:57:00 Updated by 746 2016-02-29 04:57:00
##   contact_type location     category    subcategory     u_symptom   cmdb_ci
##   <chr>        <chr>        <chr>       <chr>           <chr>       <chr>
## 1 Phone        Location 143 Category 55 Subcategory 170 Symptom 72  <NA>
## 2 Phone        Location 143 Category 55 Subcategory 170 Symptom 72  <NA>
## 3 Phone        Location 143 Category 55 Subcategory 170 Symptom 72  <NA>
## 4 Phone        Location 143 Category 55 Subcategory 170 Symptom 72  <NA>
## 5 Phone        Location 165 Category 40 Subcategory 215 Symptom 471 <NA>
##   impact     urgency    priority     assignment_group assigned_to knowledge
##   <chr>      <chr>      <chr>        <chr>            <chr>       <lgl>
## 1 2 - Medium 2 - Medium 3 - Moderate Group 56         <NA>        TRUE
## 2 2 - Medium 2 - Medium 3 - Moderate Group 56         <NA>        TRUE
## 3 2 - Medium 2 - Medium 3 - Moderate Group 56         <NA>        TRUE
## 4 2 - Medium 2 - Medium 3 - Moderate Group 56         <NA>        TRUE
## 5 2 - Medium 2 - Medium 3 - Moderate Group 70         Resolver 89 TRUE
##   u_priority_confirmation notify        problem_id rfc   vendor caused_by
##   <lgl>                   <chr>         <chr>      <chr> <chr>  <lgl>
## 1 FALSE                   Do Not Notify <NA>       <NA>  <NA>   NA
## 2 FALSE                   Do Not Notify <NA>       <NA>  <NA>   NA
## 3 FALSE                   Do Not Notify <NA>       <NA>  <NA>   NA
## 4 FALSE                   Do Not Notify <NA>       <NA>  <NA>   NA
## 5 FALSE                   Do Not Notify <NA>       <NA>  <NA>   NA
##   closed_code resolved_by     resolved_at         closed_at           time_open
##   <chr>       <chr>           <dttm>              <dttm>                  <dbl>
## 1 code 5      Resolved by 149 2016-02-29 11:29:00 2016-03-05 12:00:00       613
## 2 code 5      Resolved by 149 2016-02-29 11:29:00 2016-03-05 12:00:00       613
## 3 code 5      Resolved by 149 2016-02-29 11:29:00 2016-03-05 12:00:00       613
## 4 code 5      Resolved by 149 2016-02-29 11:29:00 2016-03-05 12:00:00       613
## 5 code 5      Resolved by 81  2016-03-01 09:52:00 2016-03-06 10:00:00      1752
##   resolved_updated_diff
##                   <dbl>
## 1                   606
## 2                   156
## 3                     0
## 4                 -7231
## 5                  1735
## # ... with 138,560 more rows
```

We will only consider observations that have incident_state != 'Closed', since we want to predict the resolution time while incident is still not closed. To check whether all the incidents in the dataset are closed:

```
#group rows by status
incident_event_log %>% group_by(incident_state) %>% summarise(n = n())
```

```
## # A tibble: 9 x 2
##   incident_state         n
##   <chr>              <int>
## 1 -100                   5
## 2 Active             38710
## 3 Awaiting Evidence     38
## 4 Awaiting Problem     461
## 5 Awaiting User Info 14641
## 6 Awaiting Vendor      707
## 7 Closed             23426
## 8 New                  36388
## 9 Resolved             24189
```

```
# number of unique incs where state != 'closed'
incident_event_log %>%  filter(incident_state != 'Closed') %>% group_by(number) %>% summarise(n = n())
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1 23361
```

```
# number of unique incs where state == 'closed'
incident_event_log %>%  filter(incident_state == 'Closed') %>% group_by(number) %>% summarise(n = n())
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1 23361
```

```
#note that incident can be closed more than once
incident_event_log %>%  filter(incident_state == 'Closed')  %>% summarise(n = n())
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1 23426
```

```
closed_more_than_once <- incident_event_log %>%  filter(incident_state == 'Closed') %>% group_by(number
closed_more_than_once_v <- as.vector(t(closed_more_than_once))
```

According to ITIL Incident Management standards, incidents should not be closed more than once. We will consider the cases when incident is in status ‘closed’ more than once as exceptions and will exclude them from the analysis.

Incidents with status logs = ‘-100’ we will replace with ‘Active’ status, since they are not closed and -100 does not seem to have a special meaning.

```
# number of unique incs where state == '-100'
incident_event_log %>%  filter(incident_state == '-100') %>% group_by(number) %>% summarise(n = n())
```

```
## # A tibble: 2 x 2
##   number         n
##   <chr>      <int>
## 1 INC0028089     2
## 2 INC0030413     3
```

```
incident_event_log %>% filter(number == 'INC0028089')
```

```
## # A tibble: 6 x 40
##     ...1   ...2 number     incident_state active reassignment_count reopen_count
##    <dbl>  <dbl> <chr>      <chr>          <lgl>               <dbl>        <dbl>
## 1 112837 112837 INC0028089 New            TRUE                    0            0
## 2 112838 112838 INC0028089 -100           TRUE                    0            0
## 3 112839 112839 INC0028089 -100           TRUE                    0            0
## 4 112840 112840 INC0028089 Resolved       TRUE                    0            0
## 5 112841 112841 INC0028089 Resolved       TRUE                    0            0
## 6 112842 112842 INC0028089 Closed         FALSE                   0            0
## # ... with 33 more variables: sys_mod_count <dbl>, made_sla <lgl>,
## #   caller_id <chr>, opened_by <chr>, opened_at <dttm>, sys_created_by <chr>,
## #   sys_created_at <dttm>, sys_updated_by <chr>, sys_updated_at <dttm>,
## #   contact_type <chr>, location <chr>, category <chr>, subcategory <chr>,
## #   u_symptom <chr>, cmdb_ci <chr>, impact <chr>, urgency <chr>,
## #   priority <chr>, assignment_group <chr>, assigned_to <chr>, knowledge <lgl>,
## #   u_priority_confirmation <lgl>, notify <chr>, problem_id <chr>, ...
```

```
incident_event_log %>% filter(number == 'INC0030413')
```

```
## # A tibble: 9 x 40
##     ...1   ...2 number     incident_state active reassignment_count reopen_count
##    <dbl>  <dbl> <chr>      <chr>          <lgl>               <dbl>        <dbl>
## 1 121577 121577 INC0030413 New            TRUE                    0            0
## 2 121578 121578 INC0030413 New            TRUE                    1            0
## 3 121579 121579 INC0030413 Active         TRUE                    1            0
## 4 121580 121580 INC0030413 Active         TRUE                    1            0
## 5 121581 121581 INC0030413 -100           TRUE                    1            0
## 6 121582 121582 INC0030413 -100           TRUE                    1            0
## 7 121583 121583 INC0030413 -100           TRUE                    1            0
## 8 121584 121584 INC0030413 Resolved       TRUE                    1            0
## 9 121585 121585 INC0030413 Closed         FALSE                   1            0
## # ... with 33 more variables: sys_mod_count <dbl>, made_sla <lgl>,
## #   caller_id <chr>, opened_by <chr>, opened_at <dttm>, sys_created_by <chr>,
## #   sys_created_at <dttm>, sys_updated_by <chr>, sys_updated_at <dttm>,
## #   contact_type <chr>, location <chr>, category <chr>, subcategory <chr>,
## #   u_symptom <chr>, cmdb_ci <chr>, impact <chr>, urgency <chr>,
## #   priority <chr>, assignment_group <chr>, assigned_to <chr>, knowledge <lgl>,
## #   u_priority_confirmation <lgl>, notify <chr>, problem_id <chr>, ...
```

The final data set in terms of the observations:

```
incident_event_log <- incident_event_log %>%
     mutate(incident_state = replace(incident_state, incident_state == '-100', 'Active'))  %>%
     filter(!number %in% closed_more_than_once_v) %>%
    filter(!incident_state == 'Closed')

#group rows by status
incident_event_log %>% group_by(incident_state) %>% summarise(n = n())
```

```
## # A tibble: 7 x 2
##   incident_state         n
##   <chr>              <int>
## 1 Active             38540
## 2 Awaiting Evidence     37
## 3 Awaiting Problem     461
## 4 Awaiting User Info 14548
## 5 Awaiting Vendor      701
## 6 New                36264
## 7 Resolved           24110
```

We will remove the columns: active,caller_id, opened_by, opened_at, sys_created_at, sys_created_by, sys_updated_at, cmdb_ci, notify, u_priority_confirmation, rfc, caused_by, vendor, resolved_by, resolved_at, closed_at

```
## active columns
incident_event_log %>% group_by(active) %>% summarise(n = n())
```

```
## # A tibble: 2 x 2
##   active      n
##   <lgl>   <int>
## 1 FALSE       1
## 2 TRUE   114660
```

```
incident_event_log %>% filter(active == FALSE) # error
```

```
## # A tibble: 1 x 40
##    ...1  ...2 number     incident_state active reassignment_count reopen_count
##   <dbl> <dbl> <chr>      <chr>          <lgl>               <dbl>        <dbl>
## 1 76349 76349 INC0018594 Resolved       FALSE                   1            0
## # ... with 33 more variables: sys_mod_count <dbl>, made_sla <lgl>,
## #   caller_id <chr>, opened_by <chr>, opened_at <dttm>, sys_created_by <chr>,
## #   sys_created_at <dttm>, sys_updated_by <chr>, sys_updated_at <dttm>,
## #   contact_type <chr>, location <chr>, category <chr>, subcategory <chr>,
## #   u_symptom <chr>, cmdb_ci <chr>, impact <chr>, urgency <chr>,
## #   priority <chr>, assignment_group <chr>, assigned_to <chr>, knowledge <lgl>,
## #   u_priority_confirmation <lgl>, notify <chr>, problem_id <chr>, ...
```

```
incident_event_log %>% filter(incident_state == 'Resolved')
```

```
## # A tibble: 24,110 x 40
##     ...1  ...2 number     incident_state active reassignment_count reopen_count
##    <dbl> <dbl> <chr>      <chr>          <lgl>               <dbl>        <dbl>
##  1     2     2 INC0000045 Resolved       TRUE                    0            0
##  2     3     3 INC0000045 Resolved       TRUE                    0            0
##  3    12    12 INC0000047 Resolved       TRUE                    1            0
##  4    19    19 INC0000057 Resolved       TRUE                    0            0
##  5    23    23 INC0000060 Resolved       TRUE                    0            0
##  6    31    31 INC0000062 Resolved       TRUE                    1            0
##  7    39    39 INC0000063 Resolved       TRUE                    1            0
##  8    48    48 INC0000064 Resolved       TRUE                    1            0
##  9    61    61 INC0000065 Resolved       TRUE                    6            0
## 10    65    65 INC0000066 Resolved       TRUE                    1            0
## # ... with 24,100 more rows, and 33 more variables: sys_mod_count <dbl>,
## #   made_sla <lgl>, caller_id <chr>, opened_by <chr>, opened_at <dttm>,
## #   sys_created_by <chr>, sys_created_at <dttm>, sys_updated_by <chr>,
## #   sys_updated_at <dttm>, contact_type <chr>, location <chr>, category <chr>,
## #   subcategory <chr>, u_symptom <chr>, cmdb_ci <chr>, impact <chr>,
## #   urgency <chr>, priority <chr>, assignment_group <chr>, assigned_to <chr>,
## #   knowledge <lgl>, u_priority_confirmation <lgl>, notify <chr>, ...
```

```
## caller_id - can differ
incident_event_log %>% group_by(caller_id) %>% summarise(n = n())
```

```
## # A tibble: 5,089 x 2
##    caller_id       n
##    <chr>       <int>
##  1 Caller 10      12
##  2 Caller 1000     2
##  3 Caller 1001    59
##  4 Caller 1002    34
##  5 Caller 1004     2
##  6 Caller 1005     3
##  7 Caller 1006    36
##  8 Caller 1007    51
##  9 Caller 1008     6
## 10 Caller 1009    67
## # ... with 5,079 more rows
```

```
## opened_by - does not matter in our analysis
incident_event_log %>% group_by(opened_by) %>% summarise(n = n())
```

```
## # A tibble: 207 x 2
##    opened_by          n
##    <chr>          <int>
##  1 Opened by  10     59
##  2 Opened by  101    12
##  3 Opened by  104    12
##  4 Opened by  106    10
##  5 Opened by  108  4891
##  6 Opened by  109   116
##  7 Opened by  111     3
##  8 Opened by  118    36
##  9 Opened by  119    13
## 10 Opened by  12    144
## # ... with 197 more rows
```

```
## sys_created_by - as factor. Support engineer who submitted the **first** log.
incident_event_log %>% group_by(sys_created_by) %>% summarise(n = n())
```

```
## # A tibble: 186 x 2
##    sys_created_by     n
##    <chr>          <int>
##  1 Created by 1      11
##  2 Created by 10  20682
##  3 Created by 100    63
##  4 Created by 101    82
##  5 Created by 102     4
##  6 Created by 103    15
##  7 Created by 107   915
##  8 Created by 108   367
##  9 Created by 109   103
## 10 Created by 110     3
## # ... with 176 more rows
```

```
## sys_created_by - as factor. Support engineer who submitted the **first** log.
incident_event_log %>% group_by(sys_created_by) %>% summarise(n = n())
```

```
## # A tibble: 186 x 2
##    sys_created_by     n
##    <chr>          <int>
##  1 Created by 1      11
##  2 Created by 10  20682
##  3 Created by 100    63
##  4 Created by 101    82
##  5 Created by 102     4
##  6 Created by 103    15
##  7 Created by 107   915
##  8 Created by 108   367
##  9 Created by 109   103
## 10 Created by 110     3
## # ... with 176 more rows
```

```
## cmdb_ci - unknown attribute
incident_event_log %>% group_by(cmdb_ci) %>% summarise(n = n())
```

```
## # A tibble: 50 x 2
##    cmdb_ci        n
##    <chr>      <int>
##  1 cmdb_ci 10     7
##  2 cmdb_ci 11    19
##  3 cmdb_ci 12    10
##  4 cmdb_ci 13     5
##  5 cmdb_ci 14    15
##  6 cmdb_ci 15     4
##  7 cmdb_ci 16     3
##  8 cmdb_ci 17    12
##  9 cmdb_ci 18     5
## 10 cmdb_ci 19     6
## # ... with 40 more rows
```

```
## u_priority_confirmation - unknown attribute
incident_event_log %>% group_by(u_priority_confirmation) %>% summarise(n = n())
```

```
## # A tibble: 2 x 2
##   u_priority_confirmation     n
##   <lgl>                   <int>
## 1 FALSE                   93053
## 2 TRUE                    21608
```

```
## notify
incident_event_log %>% group_by(notify) %>% summarise(n = n())
```

```
## # A tibble: 2 x 2
##   notify              n
##   <chr>           <int>
## 1 Do Not Notify 114578
## 2 Send Email        83
```

```
## rfc - unknown attribute
incident_event_log %>% group_by(rfc) %>% summarise(n = n())
```

```
## # A tibble: 182 x 2
##    rfc            n
##    <chr>      <int>
##  1 CHG0000047    17
##  2 CHG0000084     4
##  3 CHG0000089     8
##  4 CHG0000097    10
##  5 CHG0000127     7
##  6 CHG0000132    18
##  7 CHG0000149     3
##  8 CHG0000171     1
##  9 CHG0000177     5
## 10 CHG0000179     4
## # ... with 172 more rows
```

```
## vendor - does not seem relevant in the analysis.
incident_event_log %>% group_by(vendor) %>% summarise(n = n())
```

```
## # A tibble: 5 x 2
##   vendor         n
##   <chr>      <int>
## 1 code 8s      161
## 2 Vendor 1      60
## 3 Vendor 2       2
## 4 Vendor 3       6
## 5 <NA>      114432
```

```
## caused_by - all NAs
incident_event_log %>% group_by(caused_by) %>% summarise(n = n())
```

```
## # A tibble: 1 x 2
##   caused_by      n
##   <lgl>      <int>
## 1 NA        114661
```

We will exclude for now (might consider later): location, u_symptom, knowledge

```
## location - does not seem relevant in this analysis. There are NAs.
incident_event_log %>% group_by(location) %>% summarise(n = n())
```

```
## # A tibble: 221 x 2
##    location         n
##    <chr>        <int>
##  1 Location 10      8
##  2 Location 100     4
##  3 Location 101     3
##  4 Location 102     4
##  5 Location 105     8
##  6 Location 106     2
##  7 Location 107    88
##  8 Location 108 10279
##  9 Location 109    35
## 10 Location 11      5
## # ... with 211 more rows
```

```
## u_symptom - seems to an additional note that may or may not be included in an incident. There are 26
incident_event_log %>% group_by(u_symptom) %>% summarise(n = n())
```

```
## # A tibble: 525 x 2
##    u_symptom       n
##    <chr>       <int>
##  1 Symptom 10   1013
##  2 Symptom 101    51
##  3 Symptom 102   739
##  4 Symptom 103     1
##  5 Symptom 104     2
##  6 Symptom 105   488
##  7 Symptom 106    54
##  8 Symptom 107    15
##  9 Symptom 109     7
## 10 Symptom 11     19
## # ... with 515 more rows
```

```
incident_event_log %>% group_by(category, subcategory,u_symptom ) %>% summarise(n = n())
```

```
## `summarise()` has grouped output by 'category', 'subcategory'. You can override using the `.groups` a
```

```
## # A tibble: 2,083 x 4
## # Groups:   category, subcategory [360]
##    category    subcategory     u_symptom       n
##    <chr>       <chr>           <chr>       <int>
##  1 Category 10 Subcategory 158 Symptom 494     7
##  2 Category 10 Subcategory 158 Symptom 565    10
##  3 Category 10 Subcategory 177 Symptom 494     4
##  4 Category 12 Subcategory 165 Symptom 562     2
##  5 Category 13 Subcategory 174 Symptom 491    77
##  6 Category 13 Subcategory 174 <NA>           25
##  7 Category 13 Subcategory 209 Symptom 379    18
##  8 Category 13 Subcategory 209 Symptom 491     2
##  9 Category 13 Subcategory 209 <NA>            2
## 10 Category 13 Subcategory 302 Symptom 208    16
## # ... with 2,073 more rows
```

```
## knowledge - unknown attribute
incident_event_log %>% group_by(knowledge) %>% summarise(n = n())
```

```
## # A tibble: 2 x 2
##   knowledge     n
##   <lgl>     <int>
## 1 FALSE     93104
## 2 TRUE      21557
```

Keep for analysis: number, incident_state, reassignment_count, reopen_count, sys_mod_count, made_sla, sys_updated_by, contact_type, category, subcategory.

```
## reassignment_count - change of group
incident_event_log %>% filter(number == 'INC0000065') %>% select(number, incident_state, reassignment_c
```

```
## # A tibble: 12 x 4
##    number     incident_state     reassignment_count assignment_group
##    <chr>      <chr>                           <dbl> <chr>
##  1 INC0000065 New                                 0 Group 5
##  2 INC0000065 New                                 0 Group 5
##  3 INC0000065 New                                 0 Group 5
##  4 INC0000065 New                                 1 Group 70
##  5 INC0000065 New                                 2 Group 15
##  6 INC0000065 New                                 2 Group 15
##  7 INC0000065 New                                 3 Group 70
##  8 INC0000065 New                                 4 Group 12
##  9 INC0000065 New                                 5 Group 15
## 10 INC0000065 New                                 6 Group 33
## 11 INC0000065 Awaiting User Info                  6 Group 33
## 12 INC0000065 Resolved                            6 Group 33
```

```
## reopen_count - after status 'resolved'
incident_event_log %>% filter(reopen_count != 0)
```

```
## # A tibble: 1,909 x 40
##     ...1  ...2 number     incident_state     active reassignment_co~ reopen_count
##    <dbl> <dbl> <chr>      <chr>              <lgl>             <dbl>        <dbl>
##  1   216   216 INC0000102 Active             TRUE                  4            1
##  2   217   217 INC0000102 Resolved           TRUE                  4            1
##  3   825   825 INC0000294 Active             TRUE                  1            1
##  4   826   826 INC0000294 Active             TRUE                  1            1
##  5   827   827 INC0000294 Active             TRUE                  1            1
##  6   828   828 INC0000294 Awaiting User Info TRUE                  1            1
##  7   829   829 INC0000294 Awaiting User Info TRUE                  1            1
##  8   830   830 INC0000294 Awaiting User Info TRUE                  1            1
##  9   831   831 INC0000294 Awaiting User Info TRUE                  1            1
## 10   832   832 INC0000294 Awaiting User Info TRUE                  1            1
## # ... with 1,899 more rows, and 33 more variables: sys_mod_count <dbl>,
## #   made_sla <lgl>, caller_id <chr>, opened_by <chr>, opened_at <dttm>,
## #   sys_created_by <chr>, sys_created_at <dttm>, sys_updated_by <chr>,
## #   sys_updated_at <dttm>, contact_type <chr>, location <chr>, category <chr>,
## #   subcategory <chr>, u_symptom <chr>, cmdb_ci <chr>, impact <chr>,
## #   urgency <chr>, priority <chr>, assignment_group <chr>, assigned_to <chr>,
## #   knowledge <lgl>, u_priority_confirmation <lgl>, notify <chr>, ...
```

```
incident_event_log %>% filter(number == 'INC0000294') %>% select(number, incident_state, reopen_count)
```

```
## # A tibble: 20 x 3
##    number     incident_state     reopen_count
##    <chr>      <chr>                     <dbl>
##  1 INC0000294 New                           0
##  2 INC0000294 New                           0
##  3 INC0000294 New                           0
##  4 INC0000294 Awaiting Problem              0
##  5 INC0000294 Awaiting Problem              0
##  6 INC0000294 Awaiting Problem              0
##  7 INC0000294 Awaiting Problem              0
##  8 INC0000294 Resolved                      0
##  9 INC0000294 Active                        1
## 10 INC0000294 Active                        1
## 11 INC0000294 Active                        1
## 12 INC0000294 Awaiting User Info            1
## 13 INC0000294 Awaiting User Info            1
## 14 INC0000294 Awaiting User Info            1
## 15 INC0000294 Awaiting User Info            1
## 16 INC0000294 Awaiting User Info            1
## 17 INC0000294 Active                        1
## 18 INC0000294 Active                        1
## 19 INC0000294 Resolved                      1
## 20 INC0000294 Resolved                      1
```

```
## sys_mod_count - each new log
incident_event_log %>% filter(sys_mod_count != 0)
```

```
## # A tibble: 91,360 x 40
##     ...1  ...2 number     incident_state     active reassignment_co~ reopen_count
##    <dbl> <dbl> <chr>      <chr>              <lgl>             <dbl>        <dbl>
##  1     2     2 INC0000045 Resolved           TRUE                  0            0
##  2     3     3 INC0000045 Resolved           TRUE                  0            0
##  3     6     6 INC0000047 Active             TRUE                  1            0
##  4     7     7 INC0000047 Active             TRUE                  1            0
##  5     8     8 INC0000047 Active             TRUE                  1            0
##  6     9     9 INC0000047 Active             TRUE                  1            0
##  7    10    10 INC0000047 Active             TRUE                  1            0
##  8    11    11 INC0000047 Awaiting User Info TRUE                  1            0
##  9    12    12 INC0000047 Resolved           TRUE                  1            0
## 10    15    15 INC0000057 New                TRUE                  0            0
## # ... with 91,350 more rows, and 33 more variables: sys_mod_count <dbl>,
## #   made_sla <lgl>, caller_id <chr>, opened_by <chr>, opened_at <dttm>,
## #   sys_created_by <chr>, sys_created_at <dttm>, sys_updated_by <chr>,
## #   sys_updated_at <dttm>, contact_type <chr>, location <chr>, category <chr>,
## #   subcategory <chr>, u_symptom <chr>, cmdb_ci <chr>, impact <chr>,
## #   urgency <chr>, priority <chr>, assignment_group <chr>, assigned_to <chr>,
## #   knowledge <lgl>, u_priority_confirmation <lgl>, notify <chr>, ...
```

```
incident_event_log %>% filter(number == 'INC0000047') %>% select(number, incident_state, sys_mod_count)
```

```
## # A tibble: 8 x 3
##   number     incident_state     sys_mod_count
##   <chr>      <chr>                      <dbl>
## 1 INC0000047 New                            0
## 2 INC0000047 Active                         1
## 3 INC0000047 Active                         2
## 4 INC0000047 Active                         3
## 5 INC0000047 Active                         4
## 6 INC0000047 Active                         5
## 7 INC0000047 Awaiting User Info             6
## 8 INC0000047 Resolved                       7
```

```
    # note: sys_mod_count is not always +1. Will leave as is.
    incident_event_log %>% filter(number == 'INC0000045') %>% select(number, incident_state, sys_mod_co
```

```
## # A tibble: 3 x 3
##   number     incident_state sys_mod_count
##   <chr>      <chr>                  <dbl>
## 1 INC0000045 New                        0
## 2 INC0000045 Resolved                   2
## 3 INC0000045 Resolved                   3
```

```
## made_sla - not that many observations
incident_event_log %>% group_by(made_sla) %>% summarise(n = n())
```

```
## # A tibble: 2 x 2
##   made_sla      n
##   <lgl>     <int>
## 1 FALSE         4
## 2 TRUE     114657
```

```
incident_event_log %>% filter(made_sla == FALSE)
```

```
## # A tibble: 4 x 40
##    ...1  ...2 number     incident_state   active reassignment_count reopen_count
##   <dbl> <dbl> <chr>      <chr>            <lgl>               <dbl>        <dbl>
## 1 11257 11257 INC0002588 Active           TRUE                    0            0
## 2 11258 11258 INC0002588 Awaiting Problem TRUE                    0            0
## 3 11259 11259 INC0002588 Resolved         TRUE                    0            0
## 4 76349 76349 INC0018594 Resolved         FALSE                   1            0
## # ... with 33 more variables: sys_mod_count <dbl>, made_sla <lgl>,
## #   caller_id <chr>, opened_by <chr>, opened_at <dttm>, sys_created_by <chr>,
## #   sys_created_at <dttm>, sys_updated_by <chr>, sys_updated_at <dttm>,
## #   contact_type <chr>, location <chr>, category <chr>, subcategory <chr>,
## #   u_symptom <chr>, cmdb_ci <chr>, impact <chr>, urgency <chr>,
## #   priority <chr>, assignment_group <chr>, assigned_to <chr>, knowledge <lgl>,
## #   u_priority_confirmation <lgl>, notify <chr>, problem_id <chr>, ...
```

```
## sys_updated_by - as factor. Support engineer who submitted the log. Will assume that the engineers w
incident_event_log %>% group_by(sys_updated_by) %>% summarise(n = n())
```

```
## # A tibble: 809 x 2
##    sys_updated_by     n
##    <chr>          <int>
##  1 Updated by 1       1
##  2 Updated by 10      2
##  3 Updated by 100     4
##  4 Updated by 101     3
##  5 Updated by 102     3
##  6 Updated by 103     3
##  7 Updated by 105     1
##  8 Updated by 107     1
##  9 Updated by 108     3
## 10 Updated by 109  1032
## # ... with 799 more rows
```

```
## contact_type - factor
incident_event_log %>% group_by(contact_type) %>% summarise(n = n())
```

```
## # A tibble: 4 x 2
##   contact_type        n
##   <chr>           <int>
## 1 Direct opening     13
## 2 Email             161
## 3 Phone          113661
## 4 Self service      826
```

```
## category - factor. There are NAs.
incident_event_log %>% group_by(category) %>% summarise(n = n())
```

```
## # A tibble: 58 x 2
##    category        n
##    <chr>       <int>
##  1 Category 10    21
##  2 Category 12     2
##  3 Category 13   858
##  4 Category 14     4
##  5 Category 15     3
##  6 Category 16     5
##  7 Category 17   433
##  8 Category 19  1343
##  9 Category 2     17
## 10 Category 20  4334
## # ... with 48 more rows
```

```
## subcategory - factor. One subcategory can appear on more than one category. Should be analysed separ
#There are NAs.
incident_event_log %>% group_by(category, subcategory) %>% summarise(n = n()) %>% group_by(subcategory)
```

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

```
## # A tibble: 49 x 2
##    subcategory         n
##    <chr>           <int>
##  1 Subcategory 10      2
##  2 Subcategory 101     2
##  3 Subcategory 102     2
##  4 Subcategory 107     2
##  5 Subcategory 11      2
##  6 Subcategory 115     2
##  7 Subcategory 118     2
##  8 Subcategory 120     2
##  9 Subcategory 135     2
## 10 Subcategory 146     2
## # ... with 39 more rows
```

```
incident_event_log %>% group_by(category, subcategory) %>% summarise(n = n()) %>% filter(subcategory ==
```

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

```
## # A tibble: 2 x 3
## # Groups:   category [2]
##   category    subcategory         n
##   <chr>       <chr>           <int>
## 1 Category 55 Subcategory 102     3
## 2 Category 57 Subcategory 102    19
```

```
incident_event_log %>% group_by(category, subcategory) %>% summarise(n = n()) %>% filter(subcategory ==
```

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

```
## # A tibble: 19 x 3
## # Groups:   category [19]
##    category    subcategory         n
##    <chr>       <chr>           <int>
##  1 Category 10 Subcategory 177     4
##  2 Category 21 Subcategory 177    30
##  3 Category 25 Subcategory 177     5
##  4 Category 29 Subcategory 177    29
##  5 Category 3  Subcategory 177     1
##  6 Category 31 Subcategory 177     6
##  7 Category 33 Subcategory 177    57
##  8 Category 4  Subcategory 177     4
##  9 Category 41 Subcategory 177     4
## 10 Category 42 Subcategory 177     1
## 11 Category 44 Subcategory 177    11
## 12 Category 45 Subcategory 177    16
## 13 Category 5  Subcategory 177    21
## 14 Category 50 Subcategory 177    11
## 15 Category 52 Subcategory 177    11
## 16 Category 54 Subcategory 177    62
## 17 Category 56 Subcategory 177    20
## 18 Category 59 Subcategory 177     5
## 19 Category 6  Subcategory 177     2
## urgency
incident_event_log %>% group_by(urgency) %>% summarise(n = n())
```

```
## # A tibble: 3 x 2
##   urgency          n
##   <chr>        <int>
## 1 1 - High      3486
## 2 2 - Medium 108382
## 3 3 - Low       2793
```

```
## priority
incident_event_log %>% group_by(priority) %>% summarise(n = n())
```

```
## # A tibble: 4 x 2
##   priority          n
##   <chr>         <int>
## 1 1 - Critical   1985
## 2 2 - High       2564
## 3 3 - Moderate 106976
## 4 4 - Low        3136
```

```
## impact
incident_event_log %>% group_by(impact) %>% summarise(n = n())
```

```
## # A tibble: 3 x 2
##   impact           n
##   <chr>        <int>
## 1 1 - High      3067
## 2 2 - Medium 108579
## 3 3 - Low       3015
```

Transform for analysis: problem_id,

```
## problem_id - turn to Boolean: problems exists or not
incident_event_log %>% group_by(problem_id) %>% summarise(n = n())
```

```
## # A tibble: 252 x 2
##    problem_id           n
##    <chr>            <int>
##  1 Problem ID  10      15
##  2 Problem ID  100      8
##  3 Problem ID  101      1
##  4 Problem ID  102      6
##  5 Problem ID  103      8
##  6 Problem ID  104      7
##  7 Problem ID  105      1
##  8 Problem ID  106      1
##  9 Problem ID  107      1
## 10 Problem ID  108      1
## # ... with 242 more rows
```

Final data set in terms of the predictor variables:

```
incident_event_log <- incident_event_log %>%
    select(number, incident_state,reassignment_count, reopen_count, sys_mod_count, made_sla, sys_updated
     mutate(problem_id = if_else(is.na(problem_id),0,1))
head(incident_event_log)
```

```
## # A tibble: 6 x 12
##   number     incident_state reassignment_co~ reopen_count sys_mod_count made_sla
##   <chr>      <chr>                     <dbl>        <dbl>         <dbl> <lgl>
## 1 INC0000045 New                           0            0             0 TRUE
## 2 INC0000045 Resolved                      0            0             2 TRUE
## 3 INC0000045 Resolved                      0            0             3 TRUE
## 4 INC0000047 New                           0            0             0 TRUE
## 5 INC0000047 Active                        1            0             1 TRUE
## 6 INC0000047 Active                        1            0             2 TRUE
## # ... with 6 more variables: sys_updated_by <chr>, contact_type <chr>,
## #   category <chr>, subcategory <chr>, problem_id <dbl>,
## #   resolved_updated_diff <dbl>
```

NAs

In the resulting dataframe the only variables with missing values are the category and subcategory columns. Even though for some of the observations we could assume the category is the same as in later observations for the same incident, we will not do that. Since it is possible for an incident to be created without a category/subcategory we will treat NA as a separate factor (base factor).

```
colSums(is.na(incident_event_log))
```

```
##                number        incident_state    reassignment_count
##                     0                     0                     0
##          reopen_count         sys_mod_count              made_sla
##                     0                     0                     0
##        sys_updated_by          contact_type              category
##                     0                     0                    67
##           subcategory            problem_id resolved_updated_diff
##                    97                     0                     0
```

```
incident_event_log %>% filter(is.na(category))
```

```
## # A tibble: 67 x 12
##    number     incident_state reassignment_co~ reopen_count sys_mod_count made_sla
##    <chr>      <chr>                     <dbl>        <dbl>         <dbl> <lgl>
##  1 INC0000359 Active                        0            0             0 TRUE
##  2 INC0000359 Awaiting User~                0            0             1 TRUE
##  3 INC0000359 Awaiting User~                0            0            12 TRUE
##  4 INC0000359 Awaiting User~                0            0            24 TRUE
##  5 INC0000359 Awaiting User~                0            0            43 TRUE
##  6 INC0000359 Awaiting User~                0            0            44 TRUE
##  7 INC0000359 Awaiting User~                0            0            49 TRUE
##  8 INC0000359 Resolved                      0            0            50 TRUE
##  9 INC0000361 New                           0            0             0 TRUE
## 10 INC0000361 Active                        0            0             1 TRUE
## # ... with 57 more rows, and 6 more variables: sys_updated_by <chr>,
## #   contact_type <chr>, category <chr>, subcategory <chr>, problem_id <dbl>,
## #   resolved_updated_diff <dbl>
```

```
incident_event_log %>% filter(number == 'INC0108121')
```

```
## # A tibble: 8 x 12
##   number     incident_state reassignment_co~ reopen_count sys_mod_count made_sla
##   <chr>      <chr>                     <dbl>        <dbl>         <dbl> <lgl>
## 1 INC0108121 New                           0            0             0 TRUE
## 2 INC0108121 New                           1            0             1 TRUE
## 3 INC0108121 New                           2            0             2 TRUE
## 4 INC0108121 New                           2            0             3 TRUE
## 5 INC0108121 New                           3            0             4 TRUE
## 6 INC0108121 New                           4            0             5 TRUE
## 7 INC0108121 Active                        4            0             6 TRUE
## 8 INC0108121 Resolved                      4            0             7 TRUE
## # ... with 6 more variables: sys_updated_by <chr>, contact_type <chr>,
## #   category <chr>, subcategory <chr>, problem_id <dbl>,
## #   resolved_updated_diff <dbl>
```

```
incident_event_log  <- incident_event_log %>%
    mutate(category = replace(category, is.na(category), "None"), subcategory = replace(subcategory, is
```

Save dataframe

```
#setwd("~/CSP571ProjectGroup")
#write_csv(incident_event_log, "df.csv")
```