Expedia Training Data-set Analysis

SarahLynne Palomo

26/04/2021

Exploring NULL values

```
# Load Training data-set
tr <- read.csv("training_set_VU_DM.csv")

#which(is.null(tr$srch_id))  # <-- this does not work!
#which(tr$srch_id == "NULL")  # Not a real NULL value; it's only a string
#which(tr$comp3_rate == "NULL") # testing function on a column that contains "NULL"

# Create a new table for analyzing NULL stats
df_null_ratios <- data.frame(matrix(ncol = 3, nrow = 54))
names(df_null_ratios) <- list("Column_Name", "NULL_Count", "NULL_Percentage")

tr_row <- nrow(tr)
tr_col <- ncol(tr)
i <- 0

for (i in 1:tr_col) {
  # Populate column names from the training set into new table
  df_null_ratios[i, 1] <- names(tr)[i]

  # Find the rows in the training set that contain string "NULL"
  null_rows <- which(tr[,i] == "NULL")

  # Populate "NULL" counts into new table
  num_nulls <- nrow(as.matrix(null_rows))
  df_null_ratios[i, 2] <- num_nulls

  # Populate percentage of counts into new table
  null_perc <- (num_nulls / tr_row) * 100
  df_null_ratios[i, 3] <- null_perc
}

df_null_ratios
```

```
##                    Column_Name NULL_Count NULL_Percentage
## 1                       srch_id          0         0.00000
## 2                     date_time          0         0.00000
## 3                       site_id          0         0.00000
## 4  visitor_location_country_id          0         0.00000
## 5      visitor_hist_starrating    4706481        94.92036
## 6         visitor_hist_adr_usd    4705359        94.89774
```

```
## 7              prop_country_id          0         0.00000
## 8                      prop_id          0         0.00000
## 9              prop_starrating          0         0.00000
## 10           prop_review_score          0         0.00000
## 11             prop_brand_bool          0         0.00000
## 12        prop_location_score1          0         0.00000
## 13        prop_location_score2    1090348        21.99015
## 14   prop_log_historical_price          0         0.00000
## 15                    position          0         0.00000
## 16                   price_usd          0         0.00000
## 17              promotion_flag          0         0.00000
## 18         srch_destination_id          0         0.00000
## 19         srch_length_of_stay          0         0.00000
## 20         srch_booking_window          0         0.00000
## 21           srch_adults_count          0         0.00000
## 22         srch_children_count          0         0.00000
## 23             srch_room_count          0         0.00000
## 24    srch_saturday_night_bool          0         0.00000
## 25   srch_query_affinity_score    4640941        93.59855
## 26   orig_destination_distance    1607782        32.42577
## 27                random_bool          0         0.00000
## 28                 comp1_rate    4838417        97.58125
## 29                  comp1_inv    4828788        97.38705
## 30    comp1_rate_percent_diff    4863908        98.09535
## 31                 comp2_rate    2933675        59.16639
## 32                  comp2_inv    2828078        57.03671
## 33    comp2_rate_percent_diff    4402109        88.78179
## 34                 comp3_rate    3424059        69.05646
## 35                  comp3_inv    3307357        66.70281
## 36    comp3_rate_percent_diff    4485550        90.46462
## 37                 comp4_rate    4650969        93.80080
## 38                  comp4_inv    4614684        93.06900
## 39    comp4_rate_percent_diff    4827261        97.35626
## 40                 comp5_rate    2735974        55.17916
## 41                  comp5_inv    2598327        52.40309
## 42    comp5_rate_percent_diff    4117248        83.03671
## 43                 comp6_rate    4718190        95.15651
## 44                  comp6_inv    4697371        94.73663
## 45    comp6_rate_percent_diff    4862173        98.06036
## 46                 comp7_rate    4642999        93.64006
## 47                  comp7_inv    4601925        92.81168
## 48    comp7_rate_percent_diff    4819832        97.20643
## 49                 comp8_rate    3041693        61.34490
## 50                  comp8_inv    2970844        59.91602
## 51    comp8_rate_percent_diff    4343617        87.60212
## 52                click_bool          0         0.00000
## 53         gross_bookings_usd    4819957        97.20895
## 54               booking_bool          0         0.00000
```

Check the fields with more than 90% NULL values against the Assignment description of the data-set for significance! i.e. Not all NULLs can be converted to 0 since this would become a real value.

Exploring srch_id

Find the unique srch_id's and count how many of them are in the submission_sample data-set:

```r
# Find unique search id's
uniq_srch <- unique(tr$srch_id)
num_uniq_srch <- nrow(as.matrix(uniq_srch))
head(uniq_srch, 10)
```

```
##  [1]  1  4  6  8 11 12 17 21 25 28
```

```r
tail(uniq_srch, 10)
```

```
##  [1] 332765 332768 332772 332774 332776 332777 332781 332782 332784 332785
```

```r
last_srch_id <- tail(uniq_srch, n=1)  # Highest srch_id in (incidentally) ascending list
perc_missing_srch <- (1 - (num_uniq_srch / last_srch_id)) * 100
perc_missing_srch
```

```
## [1] 39.96274
```

Here, we can see that a good portion of *srch_id* values are missing from the training data-set sequence (40%)

Exploring date_time

Dates are not in sequential order. It may be necessary to split *date_time* field into Date and Time.

```r
min(tr$date_time)
```

```
## [1] "2012-11-01 00:08:29"
```

```r
max(tr$date_time)
```

```
## [1] "2013-06-30 23:58:24"
```

Exploring site_id

Number of website country locations

```r
uniq_site <- unique(tr$site_id)
sort(uniq_site)
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
## [26] 26 27 28 29 30 31 32 33 34
```

Exploring visitor_location_country_id

```r
uniq_visitor_country <- unique(tr$visitor_location_country_id)
sort(uniq_visitor_country)
```

```
##   [1]   1   2   3   4   5   6   7   9  10  11  12  13  14  15  16  17  18  19
##  [19]  20  21  22  23  25  26  27  28  29  30  31  32  33  34  35  36  37  38
##  [37]  39  40  41  42  44  45  46  47  48  50  51  52  53  54  55  56  57  58
##  [55]  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  76  77
##  [73]  78  79  80  81  82  83  84  85  86  87  88  90  91  92  93  94  95  97
##  [91]  98  99 100 101 102 103 105 106 107 108 109 110 111 112 113 114 115 116
## [109] 117 118 120 121 122 123 125 126 127 128 129 130 131 132 133 134 135 136
## [127] 137 138 139 140 142 145 146 148 149 150 151 152 153 154 155 156 157 158
## [145] 160 161 162 163 164 166 167 168 169 170 172 173 174 176 177 178 179 180
## [163] 181 182 183 184 185 186 187 188 189 190 191 193 194 195 196 198 199 200
## [181] 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218
## [199] 219 220 221 222 223 224 226 227 228 229 230 231
```

```
num_uniq_v_country <- nrow(as.matrix(uniq_visitor_country))
max_uniq_v_country <- max(uniq_visitor_country)
missing_v_countries_perc <- (max_uniq_v_country - num_uniq_v_country) / max_uniq_v_country * 100
```

9% of visitor countries are missing from the data-set