homework iv

Nirbhay Pherwani, Dhiren Chandnani

2019-10-04

Initialization

Here we load the tidyverse packages and the `data.table` package and load the nyc311 data set. Then we fix the column names of the nyc311 data so that they have no spaces.

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------

## v ggplot2 3.1.0       v purrr   0.3.0
## v tibble  2.0.1       v dplyr   0.8.0.1
## v tidyr   0.8.2       v stringr 1.4.0
## v readr   1.3.1       v forcats 0.4.0

## -- Conflicts ------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(data.table)
```

```
##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last

## The following object is masked from 'package:purrr':
##
##     transpose
```

```
nyc311<-fread("311_Service_Requests_from_2010_to_Present.csv")
names(nyc311)<-names(nyc311) %>%
  stringr::str_replace_all("\\s", ".")
```

Tidying the NYC311 Data

Removing unwanted columns

```
nyc311 <- subset(nyc311, select= -c(Landmark, Park.Borough, School.City, School.State, School.Zip, Taxi
```

Complaints

We observed that there were 243 unique complaint types. A lot of complaints had just 1 or 2 instances. So for analysis only major complaints would be required. Hence, only complaints with counts greater than 60 are selected to maintain relevancy.

```
nyc311<-nyc311 %>%
  group_by(Complaint.Type) %>%
  filter(n() >= 60)
```

Complaint types had unwanted characters and also were not in a formatted manner. So we made it all upper case and replaced those characters.

```
library(dplyr)
nyc311$Complaint.Type <- toupper(nyc311$Complaint.Type)
nyc311$Complaint.Type <- gsub('/', ' ', nyc311$Complaint.Type)
nyc311$Complaint.Type <- gsub('-', ' ', nyc311$Complaint.Type)
```

Location Types

We observed that there were 138 unique Location types. A lot of locations had just 1 or 2 instances. So for analysis only major locations would be required. Hence, only locations with counts greater than 10 are selected to maintain relevancy.

```
nyc311<-nyc311 %>%
  group_by(Location.Type) %>%
  filter(n() >= 10)
```

Agencies

We observed that there were 64 unique Agency types. A lot of Agencies had just 1 or 2 instances. So for analysis only major agencies would be required. Hence, only agencies with counts greater than 10 are selected to maintain relevancy.

```
nyc311<-nyc311 %>%
  group_by(Agency) %>%
  filter(n() >= 10)
```

Community Board

Community Board had a combination of location code and the Borough name. As we already had the Borough column, we just kept the community code using tidyr separate command.

```
nyc311<-nyc311 %>% separate(Community.Board, c("Community.Code"), sep=" ", extra = "drop")
```

Incident Address

Incident Address had a combination of location code and the Street name. As we already had the street name column, we just kept the location code using tidyr separate command.

```
nyc311<-nyc311 %>% separate(Incident.Address, c("Incident.Code"), sep=" ",extra = "drop")
```

Zipcodes

The zipcodes were not in a correct 5-digit format. We used the zipcode package to clean the zip codes.

```
#install.packages("zipcode")
library(zipcode)
nyc311$Incident.Zip<-clean.zipcodes(nyc311$Incident.Zip)
```

City

Although the city columnn seems a little ambiguous, we did not delete it altogether. We just changed the case to Upper for now.

```
library(dplyr)
nyc311$City <- toupper(nyc311$City)
```

Missing Values

We wanted to replace all kinds of missing values to NA, but since the data count is too high, it was turning into a very expensive operation. We will think of faster ways to achieve this. For now we have included the code (Commented) that we tried.

```
#nyc311[nyc311=="" | nyc311=="N/A" | nyc311=="N / A" | nyc311==" " | nyc311=="UNKNOWN"]<-NA
```

Second Dataset

Importing Dataset

We have selected the Housing New York Units by Building data. The data is provided by Department of Housing Preservation and Development (HPD) and is avaliable from the NYC Open Data website (https://data.cityofnewyork.us/Housing-Development/Housing-New-York-Units-by-Building/hg8x-zxpr)

```
nycHousingData <-fread("https://data.cityofnewyork.us/api/views/hg8x-zxpr/rows.csv")
names(nycHousingData)<-names(nycHousingData) %>%
  stringr::str_replace_all("\\s", ".")
nycHousingData[nycHousingData=="" | nycHousingData==" "]<-NA
```

The data is clean. Only the Borough names are not in the same case as the NYC 311 data. So we decided to change the case.

```r
nycHousingData$Borough <- toupper(nycHousingData$Borough)
```