Problem Set 1

Data Analysis and Visualization using R

RStudy2020

Due by 2020년 10월 11일 일요일 5시 (한국시간)

Information & Instructions

출제자: 박상훈
데이터 파일
- R의 `nycflights13` 패키지가 제공하는 데이터를 사용.

제출
- 정해진 시간까지 정해진 문제에 대한 코드를 작성한 R 스크립트를 `sp23@email.sc.edu` 혹은 Dropbox의 `4_QnA or Discussion` 폴더에 업로드.
- 모든 코드는 다른 작업환경에서 열더라도 재생산가능하게 디렉토리 설정 등을 모두 고려한 결과물이어야 함.

Here starts the actual test

Part 1: Data loading and cleaning

문제 1

패키지의 설치와 라이브러리 이용 `nycflights13` 패키지를 설치하고, 해당 라이브러리에서 `flights` 데이터를 `df` 라는 이름으로 저장하라. 이후의 모든 문제에 대해서 결과를 별도의 데이터에 따로 저장하지 말고 그대로 작업하라. 즉, `new_df <- df %>% drop_na()`가 아니라 `df %>% drop_na()` 로만 작업하여 콘솔창에 그 결과를 바로 확인할 수 있도록 하라.

만약 부득이하게 별도의 데이터를 저장해야 할 경우, 해당 데이터는 `temp`로 명명하라.

문제 2

`dplyr`의 `filter()`

데이터에서 3월에 해당하는 경우만을 선택하라. `month` 변수는 숫자형으로 1은 1월, 2는 2월을 나타내는 형식을 가지고 있다. 결과를 별도의 객체로 저장하지말고 실행만 하는 코드를 제시하라.

기항지(origin airport)가 `JFK` 또는 `LGA`이며 동시에 겨울에 비행한 경우를 선택하라. 이때, 겨울은 11월부터 2월을 의미한다.

문제 3

`dplyr`의 `select()`

변수의 제거 어차피 모든 연도는 2013년이므로 불필요한 연도 변수를 데이터에서 제거하라. 제거한 결과를 다시 `df`에 저장하지 말고, 실행하는 코드만을 제시하라.

변수의 순서 변경 변수 명에 언더스코어(_를 포함한 경우를 제일 앞에 오도록 데이터의 변수 순서를 변경하라.

변수의 부분 선택 `dep_delay`부터 `tailnum` 까지의 변수를 한 번에 선택하라.

변수의 유형별 선택 정수형(integer)인 변수들만 선택하라.

변수의 이름을 이용한 선택 변수명이 `arr` 또는 `dep`로 시작하는 경우만을 선택하라.

문제 4

`dplyr`의 `mutate()`

변수 생성 `total_delay`라는 변수를 만들어라. 이 변수는 도착 시간과 출발 지연 시간을 더한 결과를 보여주는 변수여야 한다. 그리고 `arr_time`, `dep_delay` 보다는 앞에, 다른 변수들보다는 뒤에 위치하도록 순서를 변경하라 (`dplyr::select` 이용).

변수의 조건 변경 모든 문자형 변수들을 요인형(factor)로 변경하고 바뀐 결과를 데이터의 구조를 보여주는 함수로 제시하라.

변수의 조건 생성 `distnace_bins`라는 변수를 만들어라. 이 변수는 거리(`distance`)가 1000보다 짧으면 "short", 1000과 2000 사이면 "medium", 그리고 2000보다 크면 "long"이라는 값을 가져야 한다. `if_else()` 또는 `case_when()`을 이용해 조건을 부여할 수 있다.

보너스 문제

아래의 힌트를 참고하여 `distance_bins`를 순위를 가진 요인형 변수로 만들어라.

```
## case_when()으로 만든 df의 x 함수가 문자형일 때 아래와 같이 요인형으로 만들
## 수 있다.
## 방법 1
df %>% mutate(
  x = case_when(
    condition1 ~ "result1",
    condition2 ~ "result2",
    condition3 ~ "result3",
    TRUE ~ NA_character_
  ) %>% parse_factor(., levels = c("results1", "results2", "results3"),
                     ordered = T, include_na = F)
)
```

```r
## 방법 2
df$x <- factor(c("results1", "results2", "results3"))
as.integer(df$x) # 이 경우 알파벳 순서대로 순위가 매겨진다.

## 방법 3
df$x <- factor(c("results1", "results2", "results3"),
               levels = c("results1", "results2", "results3"))
as.integer(df$x)
## 이 경우 results1 < results2 < results3 로 순위가 매겨진다.
```

만들어진 요인형 변수의 순위를 확인하라. `levels()` 함수를 이용하라.

문제 5

dplyr의 `count()`

빈도 계산 데이터셋에서 매일 이륙한 비행기의 수를 계산하라.

요약통계치 계산 매 달 각 기항지의 평균 출발지연 시간을 계산하라.

dplyr의 `mutate()`와 `group_by()`, `summarise()` 함수 겨울(12월-2월), 봄(3월-5월), 여름(6월-8월), 가을(9월-11월)에 해당하는 계절 변수를 만들어라. 그리고 각 계절별 기항지의 평균 출발지연 시간과 도착지연 시간을 계산하라. 이때, 만들어지는 결과를 `summary`라는 이름으로 저장하고, 티블 혹은 데이타프레임의 형식을 갖도록 저장하라.

Part 2: Data Visualization

문제 6: `ggplot2()`

아래의 요구에 따라 플롯을 작성하라. 단, 모든 플롯은 적절한 축제목과 표제목을 갖추어야 한다.

`dep_delay`와 `arr_delay`의 분포를 가장 잘 보여줄 수 있는 플롯을 제시하라.

앞서 만든 `distnace_bins` 변수를 가장 잘 보여줄 수 있는 방식으로 플롯을 작성하라.

각 기항지의 계절별 평균 출발 지연 시간과 도착 지연 시간을 가장 잘 보여줄 수 있는 방식으로 플롯을 작성하라.