Bike Rental Report

Louise Braithwaite

31/07/2020

1. Business Overview

Over a two year period a bike rental company has collected data on the number of daily rentals, daily weather conditions and day type. The company has commisioned the NICD to analyse and deliver business insights from this data set. The company would like help to understand how the change in weather effects the number of bikes the company rent out each day.

1.1 Data mining goals and success criteria

This project will focus on data description and summarisation of the data set to help develop insights into the impact weather has on bike rentals.

The goal of this project is to:

Identify how the change in weather effect the number of bikes rented out each day.

1.2 Requirements, assumptions, and constraints

Requirements:

- A report with a maximum of 500 words and 2 figures, detailing how the data has been interpreted and what insights have been gained.
- A Git repository or the work completed
- A ProjectTemplate folder containing reports and source code

Assumptions and constraints:

- We can only work with the data set provided
- No meta date has been provided. All data assumptions should be logged so they can be verified later

2. Data Overview and Preparation

One data set was provided by the bike rental company as a csv file. The data set is complete with no missing values.

The next step was to investigate what information is contained within the data set and evaluate which variables relate to the project goal. The section below contain:

- a snapshot of the first 5 vectors of the data file. This provides the column headers and vector class
- a table providing details on each variable
- a summary explaining which elements are most helpful at addressing the project goal
- a summary of data assumptions
- data preparation considerations: any initial thoughts that should be considered before data preparation or analysis

2.1 Data Overview

Review the variables in the original bike.rental.data csv file. Please note that the original data has been saved as a data frame in R named rental.data.

```
head(rental.data, 5)
```

```
## # A tibble: 5 x 12
##   season    yr mnth  holiday weekday workingday weathersit  temp   hum windspeed
##   <chr>  <int> <chr> <chr>   <chr>   <chr>      <chr>      <dbl> <dbl>     <dbl>
## 1 SPRING  2011 JAN   NO HOL~ SAT     NO WORKIN~ MISTY       8.18  80.6      10.7
## 2 SPRING  2011 JAN   NO HOL~ SUN     NO WORKIN~ MISTY       9.08  69.6      16.7
## 3 SPRING  2011 JAN   NO HOL~ MON     WORKING D~ GOOD        1.23  43.7      16.6
## 4 SPRING  2011 JAN   NO HOL~ TUE     WORKING D~ GOOD        1.4   59.0      10.7
## 5 SPRING  2011 JAN   NO HOL~ WED     WORKING D~ GOOD        2.67  43.7      12.5
## # ... with 2 more variables: cnt <int>, days_since_2011 <int>
```

.

.

Table 1: Summary of the variables from bike_rental_data csv file

Column Header	Class	Example	Description
season	character	"SPRING"	The season, spring, summer, fall or winter
yr	integer	2011	The year, either 2011 or 2012
mnth	character	"JAN"	The month, all in shortened character form (e.g. "JAN", "FEB", "MAR")
holiday	character	"NO HOLIDAY"	Whether the day is a holiday "HOLIDAY" or not "NO HOLIDAY"
weekday	character	"SAT"	The day of the week, all in shortened character form (e.g. "SAT", "SUN", "MON")
weathersit	character	"MISTY"	The overall weather category ("GOOD", "MISTY" or "RAIN/SNOW/STORM")
temp	numeric	8.18	Temperature (degrees celsius, °C)
hum	numeric	80.6	Relative humidity (%)
wind speed	numeric	10.7	Wind speed (mph)
cnt	integer	985	The number of bikes rented that day (rental count)
days_since_2011	integer	0,1,2	A sequence of numbers starting a 0 on 1 January 2011 and increasing by 1 each day

2.1.1 Data Summary An initial review of the data shows that the fields of most relevant to the project goal are:

- cnt
- season
- weathersit
- temp
- hum
- windspeed

2.1.1 Data Assumptions The following data assumptions were made. They should be verified with the data owner at the next opportunity.

- Temperature is measured in degrees celsius (°C)
- Humidity measure is relative humidity (%)
- Wind speed is measured in miles per hour (mph)
- Temperature, humidity and wind speed can be rounded to integers, as this level of detail is most appropriate and will make analysis and understanding easier.

2.2 Data Preparation Notes

1. Update the date variables (yr, month, weekday and days_since_2011)

i) The days_since_2011 variable is helpful as it indicates the date for each observation. It would be easier to navigate the data if proper date fields were created.
ii) Create date column, use the days_since_2011 variable to quality check the new date variable. Eg. observation 31 of days_since_2011 is 30 but this equates to 31 January 2011.
iii) Introduce the lubridate package to create new date variables, just.day, just.dayofweek, just.dayofweek2 (character labels) , just.month, just.month2 (character labels) and year
iv) Remove the original date columns, which are no longer needed (yr, mnth, weekday)

2. Convert categorical character variables to factors (season, holiday, workingday and weathersit)
3. Rename variables for clarity and to create naming convention ('cnt' to 'rental.count', 'weathersit' to 'weather.category', 'temp' to 'temperature, 'hum' to 'humidity', 'windspeed' to 'wind.speed', 'workingday' to 'working.day' and 'days_since_2011' to 'days.since.2011')
4. Reorder variables so the response variable is listed first
5. Round the temperature, wind speed and humidity columns so they become integers
6. Create a new data frame for regression

i) Substitute numbers for the categorical, factor values (season, weather.category, holiday and working.day)

2.2.1 Final Data Frames After preparing the data we are left with three data frames

Data Frame	Description	Dimensions
rental.data	The original data	731 obs. and 12 variables
bike.rental.data	Data preparation steps 1-5 applied	731 obs. and 16 variables
bike.rental.data.reg	Data preparation steps 1-6 applied	731 obs. and 16 variables

2.3 Initial EDA

The table function is used create table summaries of what information is included in each of of the columns of categorical data.

```
table(bike.rental.data$season)
```

```
##
##   FALL SPRING SUMMER WINTER
##    188    181    184    178
```

```
table(bike.rental.data$weather.category)
```

```
##
##            GOOD           MISTY RAIN/SNOW/STORM
##             463             247              21
```

```
table(bike.rental.data$holiday)
```

```
##
##    HOLIDAY NO HOLIDAY
##         21        710
```

```
table(bike.rental.data$working.day)
```

```
##
## NO WORKING DAY    WORKING DAY
##            231            500
```

```
table(bike.rental.data$year)
```

```
##
## 2011 2012
##  365  366
```

```
table(bike.rental.data$just.month2)
```

```
##
## Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
##  62  57  62  60  62  60  62  62  60  62  60  62
```

```
table(bike.rental.data$just.dayofweek2)
```

```
##
## Sun Mon Tue Wed Thu Fri Sat
## 105 105 104 104 104 104 105
```

It would be more interesting to see how these factors affect the daily rental count

```
bike.rental.data %>%
  group_by(season) %>%
  summarise(total.rentals = sum(rental.count),
            median.rentals = median(rental.count),
            mean.rentals = mean(rental.count),
            sd.rentals = sd(rental.count),
            percentage = round((total.rentals/3292679)*100)
)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 4 x 6
##   season total.rentals median.rentals mean.rentals sd.rentals percentage
##   <fct>          <int>          <dbl>        <dbl>      <dbl>      <dbl>
## 1 FALL         1061129          5354.        5644.      1460.         32
## 2 SPRING        471348          2209         2604.      1400.         14
```

```
## 3 SUMMER        918589          4942.        4992.      1696.         28
## 4 WINTER        841613          4634.        4728.      1700.         26
```

```
bike.rental.data %>%
  group_by(weather.category) %>%
  summarise(total.rentals = sum(rental.count),
            median.rentals = median(rental.count),
            mean.rentals = mean(rental.count),
            sd.rentals = sd(rental.count),
            percentage = round((total.rentals/3292679)*100)
  )
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 3 x 6
##   weather.category total.rentals median.rentals mean.rentals sd.rentals
##   <fct>                    <int>          <int>        <dbl>      <dbl>
## 1 GOOD                   2257952           4844        4877.      1879.
## 2 MISTY                   996858           4040        4036.      1809.
## 3 RAIN/SNOW/STORM          37869           1817        1803.      1240.
## # ... with 1 more variable: percentage <dbl>
```

```
bike.rental.data %>%
  group_by(holiday) %>%
  summarise(total.rentals = sum(rental.count),
            median.rentals = median(rental.count),
            mean.rentals = mean(rental.count),
            sd.rentals = sd(rental.count),
            percentage = round((total.rentals/3292679)*100)
  )
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 6
##   holiday    total.rentals median.rentals mean.rentals sd.rentals percentage
##   <fct>              <int>          <dbl>        <dbl>      <dbl>      <dbl>
## 1 HOLIDAY            78435           3351        3735       2103.          2
## 2 NO HOLIDAY       3214244           4558        4527.      1929.         98
```

```
bike.rental.data %>%
  group_by(working.day) %>%
  summarise(total.rentals = sum(rental.count),
            median.rentals = median(rental.count),
            mean.rentals = mean(rental.count),
            sd.rentals = sd(rental.count),
            percentage = round((total.rentals/3292679)*100)
  )
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 6
##   working.day    total.rentals median.rentals mean.rentals sd.rentals percentage
##   <fct>                  <int>          <dbl>        <dbl>      <dbl>      <dbl>
## 1 NO WORKING DAY       1000269           4459        4330.      2052.         30
## 2 WORKING DAY          2292410           4582        4585.      1878.         70
```

```
bike.rental.data %>%
  group_by(year) %>%
```

```
  summarise(total.rentals = sum(rental.count),
            median.rentals = median(rental.count),
            mean.rentals = mean(rental.count),
            sd.rentals = sd(rental.count),
            percentage = round((total.rentals/3292679)*100)
  )
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 6
##    year total.rentals median.rentals mean.rentals sd.rentals percentage
##   <dbl>         <int>          <dbl>        <dbl>      <dbl>      <dbl>
## 1  2011       1243103           3740        3406.      1379.         38
## 2  2012       2049576           5927        5600.      1789.         62
```

```
bike.rental.data %>%
  group_by(just.month2) %>%
  summarise(total.rentals = sum(rental.count),
            median.rentals = median(rental.count),
            mean.rentals = mean(rental.count),
            sd.rentals = sd(rental.count),
            percentage = round((total.rentals/3292679)*100)
  )
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 12 x 6
##    just.month2 total.rentals median.rentals mean.rentals sd.rentals percentage
##    <ord>               <int>          <dbl>        <dbl>      <dbl>      <dbl>
##  1 Jan                134933           1939         2176.      1162.          4
##  2 Feb                151352           2402         2655.      1145.          5
##  3 Mar                228920           3216.        3692.      1899.          7
##  4 Apr                269094           4294.        4485.      1776.          8
##  5 May                331686           4890.        5350.      1299.         10
##  6 Jun                346342           5308.        5772.      1240.         11
##  7 Jul                344948           5446.        5564.      1274.         10
##  8 Aug                351194           5230.        5664.      1495.         11
##  9 Sep                345991           5384         5767.      1810.         11
## 10 Oct                322352           5013         5199.      1988.         10
## 11 Nov                254831           4081         4247.      1286.          8
## 12 Dec                211036           3444         3404.      1550.          6
```

```
bike.rental.data %>%
  group_by(just.dayofweek2) %>%
  summarise(total.rentals = sum(rental.count),
            median.rentals = median(rental.count),
            mean.rentals = mean(rental.count),
            sd.rentals = sd(rental.count),
            percentage = round((total.rentals/3292679)*100)
  )
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 7 x 6
##   just.dayofweek2 total.rentals median.rentals mean.rentals sd.rentals
##   <ord>                   <int>          <dbl>        <dbl>      <dbl>
## 1 Sun                    444027           4334         4229.      1872.
```

```
## 2 Mon                    455503          4359         4338.      1793.
## 3 Tue                    469109          4576.        4511.      1827.
## 4 Wed                    473048          4642.        4549.      2038.
## 5 Thu                    485395          4721         4667.      1939.
## 6 Fri                    487790          4602.        4690.      1875.
## 7 Sat                    477807          4521         4551.      2197.
## # ... with 1 more variable: percentage <dbl>
```