Лабораторна робота №2. Маніпулювання даними

[Іващенко А.В.]

2024-11-06

Table of Contents

## Шаг 1: Создание и подключение к базе данных SQLite

Сначала создадим базу данных SQLite, затем загрузим в неё данные о матчах из пакета engsoccerdata.

library(DBI)  
library(dplyr)

##   
## Присоединяю пакет: 'dplyr'

## Следующие объекты скрыты от 'package:stats':  
##   
## filter, lag

## Следующие объекты скрыты от 'package:base':  
##   
## intersect, setdiff, setequal, union

library(RSQLite)  
library(engsoccerdata)  
  
con <- DBI::dbConnect(RSQLite::SQLite(), path = "data/my\_db.sqlite3")  
  
data("england", package = "engsoccerdata")

## Шаг 2: Экспорт данных в БД и создание индексов

Теперь мы экспортируем данные из engsoccerdata в таблицу в базе данных SQLite и создаем индексы для ускорения работы с запросами.

matches\_sqlite <- copy\_to(con, england, "matches", temporary = FALSE,  
 indexes = list(  
 c("Season", "Date"),  
 "home",  
 "visitor"  
 )  
)

## Шаг 3: Анализ данных с использованием dplyr

Теперь, когда данные в базе, мы можем использовать dplyr для выполнения различных запросов, аналогичных тем, что были указаны в вашем примере.

### 1. Выбор полей для анализа

matches\_sqlite %>%   
 select(Season, Date, home, visitor, hgoal, vgoal)

## # Source: SQL [?? x 6]  
## # Database: sqlite 3.46.0 []  
## Season Date home visitor hgoal vgoal  
## <dbl> <chr> <chr> <chr> <int> <int>  
## 1 1888 1888-09-08 Bolton Wanderers Derby County 3 6  
## 2 1888 1888-09-08 Everton Accrington F.C. 2 1  
## 3 1888 1888-09-08 Preston North End Burnley 5 2  
## 4 1888 1888-09-08 Stoke City West Bromwich Albion 0 2  
## 5 1888 1888-09-08 Wolverhampton Wanderers Aston Villa 1 1  
## 6 1888 1888-09-15 Aston Villa Stoke City 5 1  
## 7 1888 1888-09-15 Blackburn Rovers Accrington F.C. 5 5  
## 8 1888 1888-09-15 Bolton Wanderers Burnley 3 4  
## 9 1888 1888-09-15 Derby County West Bromwich Albion 1 2  
## 10 1888 1888-09-15 Everton Notts County 2 1  
## # ℹ more rows

### 2. Фильтрация матчей с разницей в счете больше 3

matches\_sqlite %>%   
 filter(abs(hgoal - vgoal) > 3)

## # Source: SQL [?? x 12]  
## # Database: sqlite 3.46.0 []  
## Date Season home visitor FT hgoal vgoal division tier totgoal goaldif  
## <chr> <dbl> <chr> <chr> <chr> <int> <int> <chr> <dbl> <int> <int>  
## 1 1888-0… 1888 Asto… Stoke … 5-1 5 1 1 1 6 4  
## 2 1888-0… 1888 Wolv… Presto… 0-4 0 4 1 1 4 -4  
## 3 1888-0… 1888 Blac… West B… 6-2 6 2 1 1 8 4  
## 4 1888-0… 1888 Asto… Notts … 9-1 9 1 1 1 10 8  
## 5 1888-0… 1888 Bolt… Everton 6-2 6 2 1 1 8 4  
## 6 1888-1… 1888 Pres… Stoke … 7-0 7 0 1 1 7 7  
## 7 1888-1… 1888 West… Derby … 5-0 5 0 1 1 5 5  
## 8 1888-1… 1888 Accr… Derby … 6-2 6 2 1 1 8 4  
## 9 1888-1… 1888 Asto… Blackb… 6-1 6 1 1 1 7 5  
## 10 1888-1… 1888 Burn… Wolver… 0-4 0 4 1 1 4 -4  
## # ℹ more rows  
## # ℹ 1 more variable: result <chr>

### 3. Среднее количество голов для каждой команды

matches\_sqlite %>%   
 group\_by(home) %>%   
 summarise(avg\_home\_goals = mean(hgoal, na.rm = TRUE))

## # Source: SQL [?? x 2]  
## # Database: sqlite 3.46.0 []  
## home avg\_home\_goals  
## <chr> <dbl>  
## 1 AFC Bournemouth 1.66  
## 2 AFC Wimbledon 1.28  
## 3 Aberdare Athletic 1.79  
## 4 Accrington 1.54  
## 5 Accrington F.C. 2.15  
## 6 Accrington Stanley 1.99  
## 7 Aldershot 1.64  
## 8 Arsenal 2.02  
## 9 Ashington 2.01  
## 10 Aston Villa 1.94  
## # ℹ more rows

### 4. Среднее количество голов для команд с более чем 100 матчами

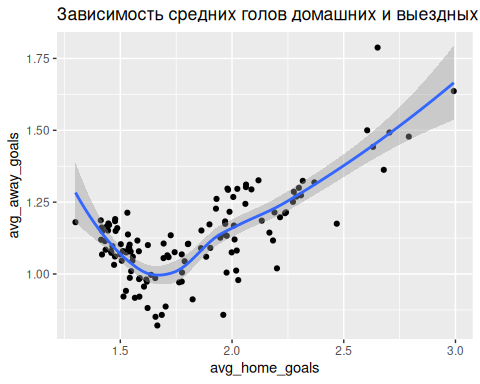
matches\_sqlite %>%   
 group\_by(home) %>%   
 summarise(  
 matches = n(),  
 avg\_home\_goals = mean(hgoal, na.rm = TRUE)  
 ) %>%   
 filter(matches > 100) %>%  
 arrange(desc(avg\_home\_goals))

## # Source: SQL [?? x 3]  
## # Database: sqlite 3.46.0 []  
## # Ordered by: desc(avg\_home\_goals)  
## home matches avg\_home\_goals  
## <chr> <int> <dbl>  
## 1 Darwen 116 2.57  
## 2 Nelson 206 2.15  
## 3 Liverpool 2338 2.11  
## 4 Bradford Park Avenue 1095 2.07  
## 5 Wigan Borough 212 2.07  
## 6 Gateshead 502 2.06  
## 7 Burton Swifts 136 2.06  
## 8 Manchester City 2375 2.04  
## 9 Manchester United 2354 2.02  
## 10 Arsenal 2338 2.02  
## # ℹ more rows

### 5. Разделение данных по сезонам и построение графика

library(ggplot2)  
  
season\_stats <- matches\_sqlite %>%  
 group\_by(Season) %>%  
 summarise(  
 total\_matches = n(),  
 avg\_home\_goals = mean(hgoal, na.rm = TRUE),  
 avg\_away\_goals = mean(vgoal, na.rm = TRUE)  
 )  
  
ggplot(season\_stats, aes(x = avg\_home\_goals, y = avg\_away\_goals)) +  
 geom\_point() +  
 geom\_smooth() +  
 labs(title = "Зависимость средних голов домашних и выездных команд по сезонам")

## `geom\_smooth()` using method = 'loess' and formula = 'y ~ x'



### 6. Количество команд и матчей по сезонам

matches\_sqlite %>%   
 group\_by(Season) %>%   
 summarise(  
 teams = n\_distinct(home),  
 matches = n()  
 )

## # Source: SQL [?? x 3]  
## # Database: sqlite 3.46.0 []  
## Season teams matches  
## <dbl> <int> <int>  
## 1 1888 12 132  
## 2 1889 12 132  
## 3 1890 12 132  
## 4 1891 14 182  
## 5 1892 28 372  
## 6 1893 31 450  
## 7 1894 32 480  
## 8 1895 32 480  
## 9 1896 32 480  
## 10 1897 32 480  
## # ℹ more rows

## Шаг 4: Закрытие соединения

После завершения работы с базой данных, не забудьте закрыть соединение:

dbDisconnect(con)