PROYECTO R 2019

CAMILLA MOSSEDDU

16/11/2019

Contents

Sección 1 Expresiones regulares

Las expresiones regulares son una secuencia de caracteres que conforma un patrón de búsqueda. Permiten filtrar textos y encontrar coincidencias. Además se utilizan para validar datos, como por ejemplo fechas, documentos de identidad o contraseñas. Se utilizan también para reemplazar un texto con unas determinadas características concretas por otro.

Actualmente muchos lenguajes de programación cuentan con el soporte necesario para el uso de estas expresiones regulares.

Estos characters pueden ser:

Sección 2 Markdown

Markdown es un lenguaje de marcado que facilita la aplicación de formato a u texto, empleando unos caracteres (como marcas) de una forma especial. Entre dichos caracteres existen cabeceras, listas, links, imágenes, tablas, videos, etc. La intención de Markdown es facilitar el uso de un simple texto plano que pueda ser convertido en un (X)HTML valido. Es decir, que el usuario escriba mas rápido y obtenga un resultado final mejor.

Algunos de los comandos que se utilizan son los siguientes:

Existen algunos editores de Markdown que se recomienda usar como: Typora, iA Writer, Visual Studio Code, etc.

Anchors

^	Start of string, or start of line in multi-line pattern
\A	Start of string
$	End of string, or end of line in multi-line pattern
\Z	End of string
\b	Word boundary
\B	Not word boundary
\<	Start of word
\>	End of word

Character Classes

\c	Control character
\s	White space
\S	Not white space
\d	Digit
\D	Not digit
\w	Word
\W	Not word
\x	Hexadecimal digit
\O	Octal digit

POSIX

[:upper:]	Upper case letters
[:lower:]	Lower case letters
[:alpha:]	All letters
[:alnum:]	Digits and letters
[:digit:]	Digits
[:xdigit:]	Hexadecimal digits
[:punct:]	Punctuation
[:blank:]	Space and tab
[:space:]	Blank characters
[:cntrl:]	Control characters
[:graph:]	Printed characters
[:print:]	Printed characters and spaces
[:word:]	Digits, letters and underscore

Assertions

?=	Lookahead assertion
?!	Negative lookahead
?<=	Lookbehind assertion
?!= or ?<!	Negative lookbehind
?>	Once-only Subexpression
?()	Condition [if then]
?()|	Condition [if then else]
?#	Comment

Quantifiers

*	0 or more	{3}	Exactly 3
+	1 or more	{3,}	3 or more
?	0 or 1	{3,5}	3, 4 or 5

Add a ? to a quantifier to make it ungreedy.

Escape Sequences

\	Escape following character
\Q	Begin literal sequence
\E	End literal sequence

"Escaping" is a way of treating characters which have a special meaning in regular expressions literally, rather than as special characters.

Common Metacharacters

^	[	.	$
{	*	(	\
+	)	|	?
<	>		

The escape character is usually \

Special Characters

\n	New line
\r	Carriage return
\t	Tab
\v	Vertical tab
\f	Form feed
\xxx	Octal character xxx
\xhh	Hex character hh

Groups and Ranges

.	Any character except new line (\n)
(a|b)	a or b
(...)	Group
(?:...)	Passive (non-capturing) group
[abc]	Range (a or b or c)
[^abc]	Not (a or b or c)
[a-q]	Lower case letter from a to q
[A-Q]	Upper case letter from A to Q
[0-7]	Digit from 0 to 7
\x	Group/subpattern number "x"

Ranges are inclusive.

Pattern Modifiers

g	Global match
i *	Case-insensitive
m *	Multiple lines
s *	Treat string as single line
x *	Allow comments and whitespace in pattern
e *	Evaluate replacement
U *	Ungreedy pattern

* PCRE modifier

String Replacement

$n	nth non-passive group
$2	"xyz" in /^(abc(xyz))$/
$1	"xyz" in /^(?:abc)(xyz)$/
$`	Before matched string
$'	After matched string
$+	Last matched string
$&	Entire matched string

Some regex implementations use \ instead of $.

Figure 1: Regex Cheatsheet

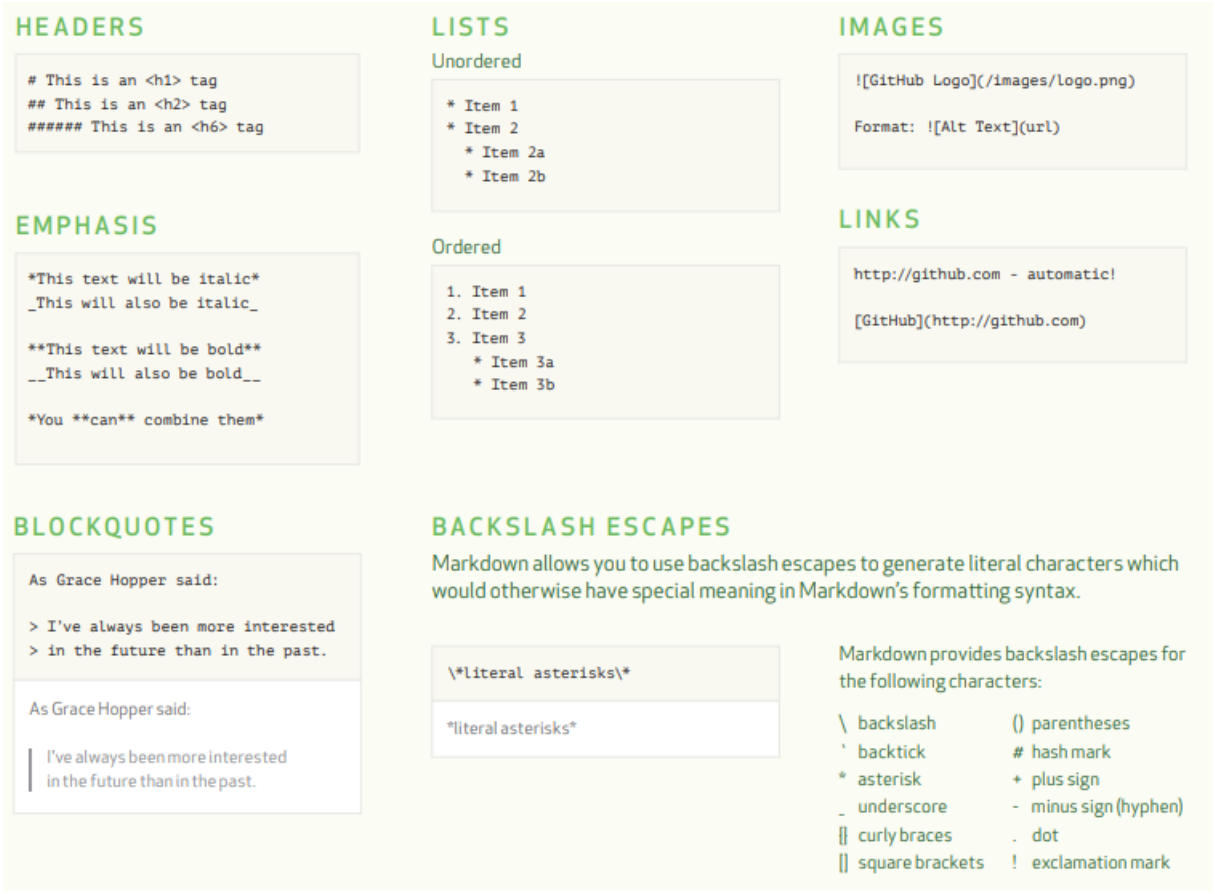


Figure 2: Markdown Cheatsheet

Sección 3

Una sección en la que se carguen, estudien, representen gráficamente unos datos de tu elección. No se trata de hacer un análisis estadístico (está fuera de alcance) pero sí un estudio descriptivo mínimo de los aspectos más relevantes de los datos. Este estudio tiene que contener, al menos: * Un gráfico generado con `ggplot2`. * Una tabla.

En primer lugar, nosotros hemos utilizado la técnica de Web Scraping para obtener los datos de milanuncios, concretamente los datos de coches.

Entre los datos concretos de coches, nosotros hemos seleccionado el precio, los kilómetros de uso que tienen, el año del modelo, el tipo de combustible que utilizan, la cilindrada y el número de puertas.

En este caso, ha sido necesario realizar un tratamiento a los datos. En primer lugar, hemos tenido que eliminar las unidades (año, kms, puertas,etc) de las columnas, puesto que no se ha podido modificar mediante la selección de nodos de Web Scrapping.

En Milanuncios, se dibujaba el tipo de combustible en dos elementos distintos (gasolina y diésel), por tanto, a la hora de obtener dicha información, se creaban dos columnas. Por tanto, hemos tenido que combinar dicha información en una única columna (llamada combustible).

Una vez eliminados las unidades y habiendo combinado las columnas, hemos exportado los datos en un fichero llamado Coches.csv.

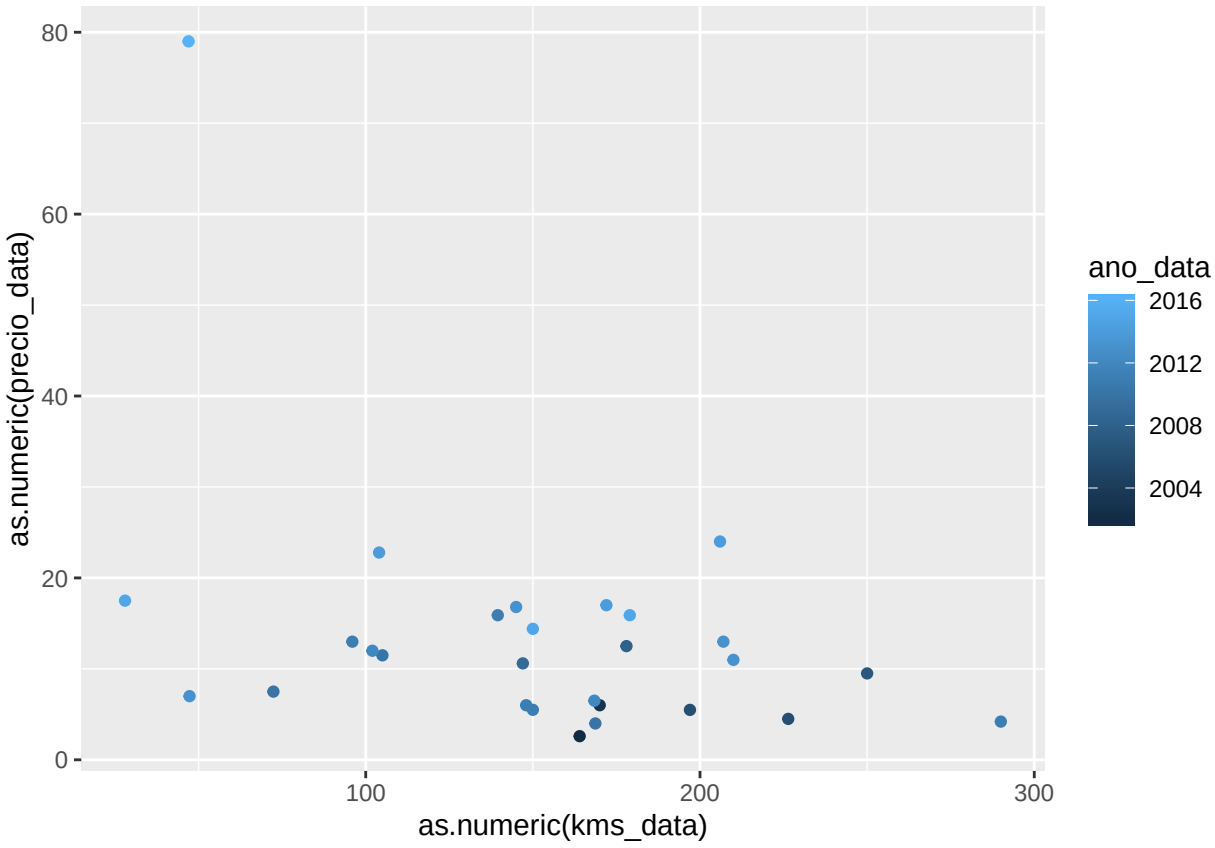
A continuación se muestran algunos de los datos ya limpios.

```
## # A tibble: 6 x 6
##   precio_data kms_data ano_data cc_data puertas_data combustible
##         <dbl>    <dbl>    <dbl>   <dbl>        <dbl> <chr>
## 1        11.0     210      2013     116            5 diesel
## 2        15.9     140.     2011     250            3 gasolina
## 3         2.6     164      2002     100            3 gasolina
## 4        16.8     145      2013     184            5 diesel
## 5        17       172      2014     136            5 diesel
## 6        12.5     178      2008     204            2 diesel
```

Una vez que se tienen los datos limpios (habiendo eliminado aquellos datos que contenían nulos) se puede comenzar a estudiar y analizar los datos mediante gráficos.

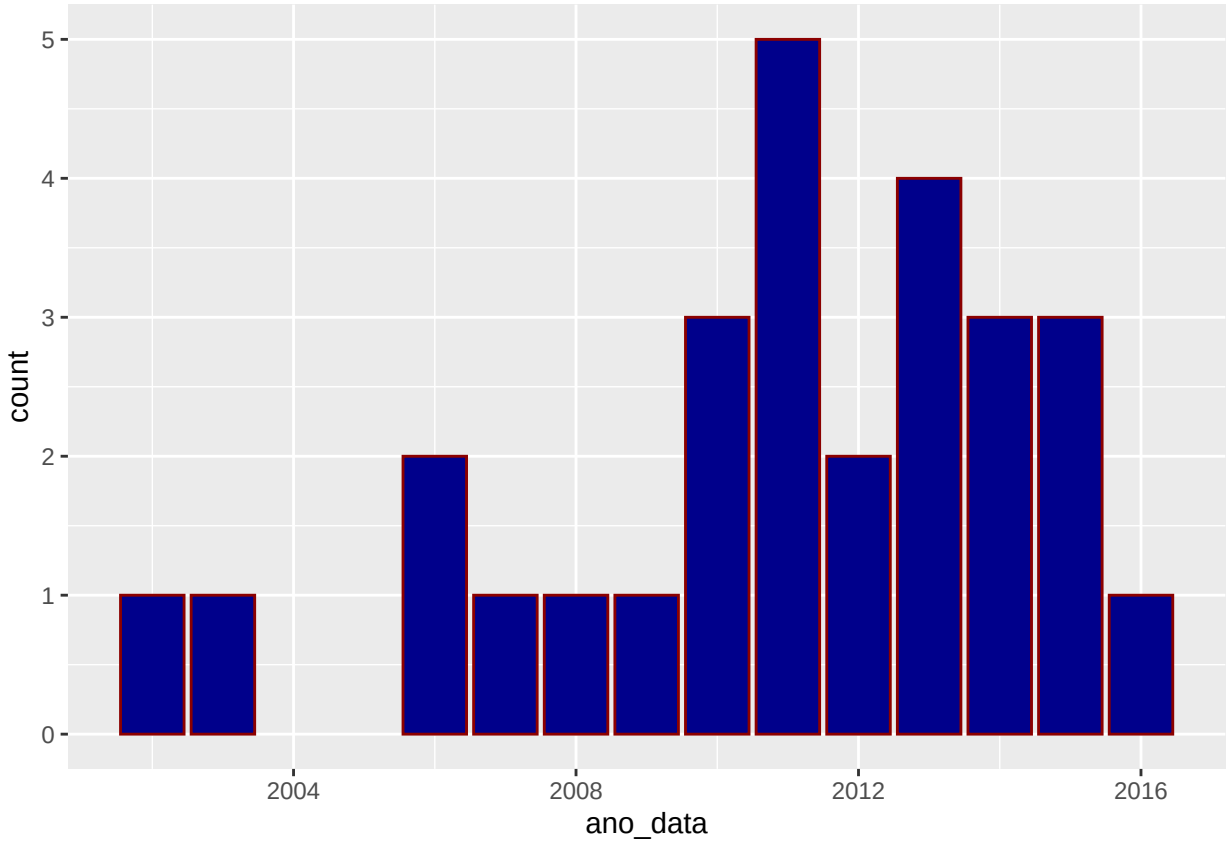
Hacemos un plot entre el precio y el combustible (coloreandolo por los años)

Vemos que no existe una relación detectable a simple vista entre el precio y el número de kilómetros.



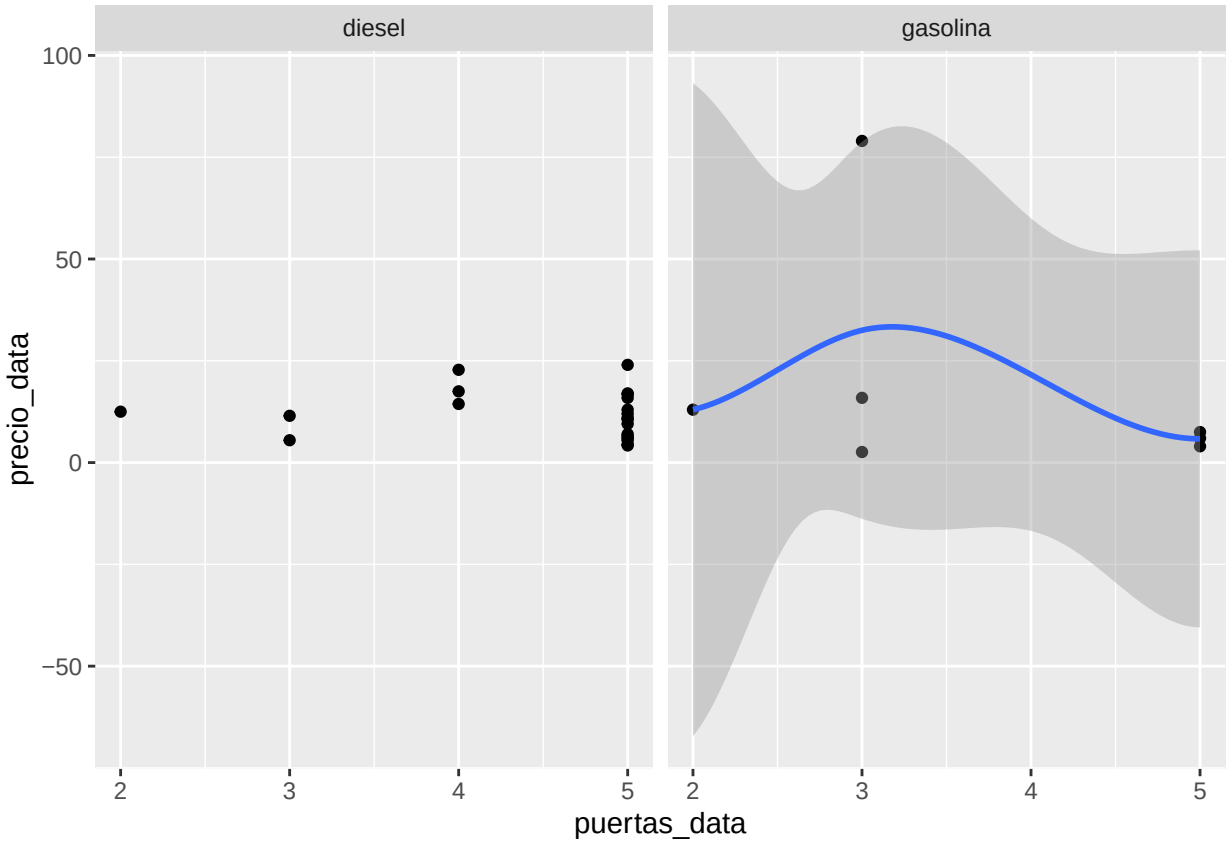
Hacemos un plot del número de coches por año de version.

Vemos que los coches la gran mayoría de coches que se están vendiendo son del 2011.



Hacemos un plot de facetas

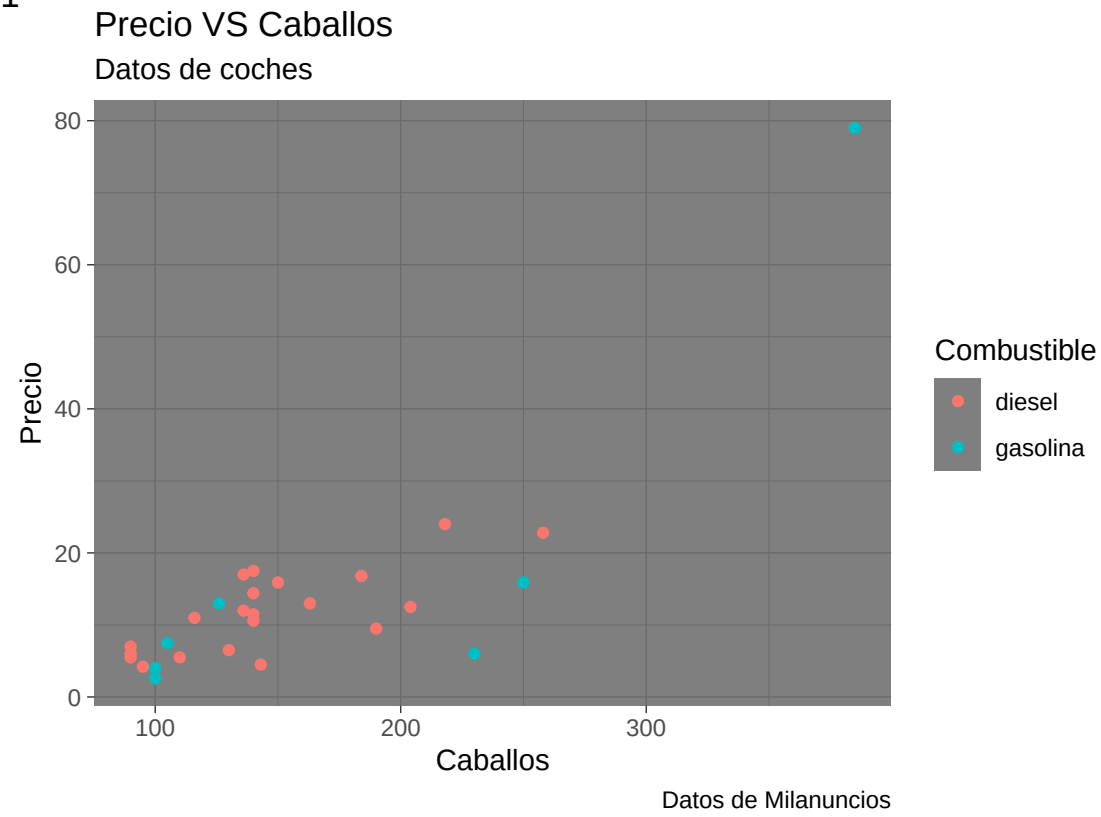
Se ha hecho un plot por facetas, utilizando el precio, el combustible y el número de puertas de cada vehículo. Vemos que la mayoría de los vehículos son de tipo diésel y tienen 5 puertas.



Hacemos un plot con un tema distinto

Como se puede observar, vemos que existe una relación lineal entre el precio y los caballos. Como cabe esperar, cuanto mayor sea la potencia de un coche, mayor será su precio.

Figure 1



Usamos el paquete EXPSS para crear tablas

Tabla sobre los precios y las puertas de los coches

coches_df$puertas_data	
2	
3	
4	
5	
coches_df$precio_data	
2.6	
1	
3.999	
1	
4.2	
1	
4.499	
1	

5.49
1
5.5
1
5.999
1
6
1
6.499
1
6.999
1
7.499
1
9.5
1
10.6
1
10.99
1
11.49
1
11.99
1
12.5
1
12.99
1
1
14.4
1
15.9
1
1
16.8
1

17
1
17.5
1
22.79
1
24
1
79
1
#Total cases
2
5
3
18

Tabla sobre los vehiculos en distintos años, su combustible y el numero de puertas

#Total
diesel
gasolina
2
3
4
5
coches_df$ano_data
2002
3.6
14.3
20
2003
3.6
14.3
5.6
2006
7.1
9.5

20

5.6

2007

3.6

4.8

5.6

2008

3.6

4.8

50

2009

3.6

4.8

5.6

2010

10.7

4.8

28.6

20

11.1

2011

17.9

14.3

28.6

50

20

16.7

2012

7.1

9.5

11.1

2013

14.3

19

22.2

2014

10.7

14.3

33.3

11.1

2015

10.7

14.3

66.7

5.6

2016

3.6

14.3

20

#Total cases

28

21

7

2

5

3

18