Section Worksheet: Regular Expressions

March 19, 2015

Write down a regular expression to match the following:

- Words with @ symbols in them, e.g., vi@gra
- An IP address (4 sets of 1-3 digits separated by periods, e.g., 124.32.6.240)
- A typical email address that ends with .com, .edu, .net, .org, or .gov

Consider the following character vector,

```
movies
```

```
## [1] "The Shawshank Redemption (1994)"
## [2] "The Godfather (1972)"
## [3] "The Godfather: Part II (1974)"
## [4] "Pulp Fiction (1994)"
## [5] "The Good, the Bad and the Ugly (1966)"
## [6] "12 Angry Men (1957)"
```

What is the return expression from each of the following function calls:

```
grep("I{2,}", movies)
```

```
grep("Go+d", movies)
```

```
gregexpr("\\(.*\\)", movies[1])
```

```
gsub("[0-9]", "", movies[6])
```

```
gsub("[[:blank:]].*$", "", movies[5])
```

```
gsub(" \\(.*$", "", movies[5])
```

Suppose we want to match the word 'cat' or 'at' or 't' but don't want to match 'cat' embedded within another word There can be other words present:

```
cats = c("diplocat", "Hi cat", "mat", "at", "t!", "ct")
```

The < stands for beginning of a word and > stands for the end of a word In R we have to escape the \ with an extra \.

```
grep("\\<(cat|at|t)\\>", cats)
```

```
## [1] 2 4 5
```

```
grep("\\<(ca|a)?t\\>", cats)
```

```
## [1] 2 4 5
```

The following do not work as expected can you figure out why?

```
grep("\\<c?a?t)\\>", cats)
```

```
## integer(0)
```

```
grep("^(cat|at|t)$", cats)
```

```
## [1] 4
```

Find the word cat or caat or caaat, etc.

```
caats = c("cat", "caat.", "caats", "caaaat", "my cat")
grep("\\<ca+t\\>", caats)
```

```
## [1] 1 2 4 5
```

```
# the {1,} is equivalent to +
grep("\\<ca{1,}t\\>", caats)
```

```
## [1] 1 2 4 5
```

Now we want to find dog anywhere in the string We don't care about capitals

```
dogs = c("dogmatic", "TopDog","Doggone it!", "RUN DOG RUN")
# The tolower function is handy here.
grep("dog", tolower(dogs))
```

```
## [1] 1 2 3 4
```

```
grep("[Dd][Oo][Gg]", dogs)
```

```
## [1] 1 2 3 4
```

Finally we are looking at character vectors where each entry must be a number. The number can have an optional sign in front of it The number can have an optional decimal point followed by digits

```
nums = c("1.2", "-3000", "5lo", "hi2", "12.", "+57")
grep("^[-+]?[[:digit:]]+(\\.[[:digit:]]+)?$", nums)
```

```
## [1] 1 2 6
```