

Homework 5

Nathaniel Haulk

10/28/2021

```
library("devtools")
```

```
## Loading required package: usethis
```

```
devtools::install_github("daijiang/neonDivData")
```

```
## WARNING: Rtools is required to build R packages, but is not currently installed.
```

```
##
```

```
## Please download and install Rtools 4.0 from https://cran.r-project.org/bin/windows/Rtools/.
```

```
## Skipping install of 'neonDivData' from a github remote, the SHA1 (04548331) has not changed since last
```

```
## Use 'force = TRUE' to force installation
```

```
library("neonDivData")
```

```
library("dplyr")
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
## filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
## intersect, setdiff, setequal, union
```

Question 1:

```
#sort(unique(data_plant$taxon_name))
```

```
## Pulls the genus for each plant
```

```
data_plant$genus = sub(".*", "", data_plant$taxon_name)
```

```
## Selects a random number 100 times between 1 and the length of data_plant
```

```
i = sample(1:nrow(data_plant), 100, replace=FALSE)
```

```
## Records the genus at the random
genus_samp = data_plant$genus[i]
```

```
## Prints out 100 genus
print(genus_samp)
```

```
## [1] "Symphyotrichum" "Dioscorea"      "Maianthemum"    "Trillium"
## [5] "Trientalis"     "Amelanchier"    "Mitchella"      "Fraxinus"
## [9] "Carex"          "Quercus"        "Justicia"       "Bernardia"
## [13] "Salix"          "Amelanchier"    "Amaranthus"     "Liatris"
## [17] "Calliandra"     "Echinocereus"   "Prosartes"      "Ambrosia"
## [21] "Hesperostipa"   "Evolvulus"     "Salvia"         "Gelsemium"
## [25] "Fraseria"       "Betula"         "Geranium"       "Erigeron"
## [29] "Xyris"          "Tilia"         "Frangula"       "Paspalum"
## [33] "Celtis"         "Bromus"         "Boerhavia"      "Dalea"
## [37] "Vaccinium"      "Smilax"         "Rhynchosia"     "Dichanthelium"
## [41] "Oxalis"         "Triticum"       "Tetraclea"      "Symphyotrichum"
## [45] "Ilex"          "Aristida"       "Juncus"         "Chamaecrista"
## [49] "Hedeoma"       "Ipomoea"        "Aira"           "Galium"
## [53] "Linum"         "Stylosanthes"   "Bouteloua"      "Polygonum"
## [57] "Mimosa"        "Bromus"         "Callisia"       "Sporobolus"
## [61] "Dryopteris"    "Artemisia"      "Bromus"         "Salix"
## [65] "Sphaeralcea"   "Populus"        "Pterocaulon"    "Sporobolus"
## [69] "Pyrola"        "Oxalis"         "Betula"         "Gaultheria"
## [73] "Ledum"         "Mammillaria"    "Ilex"           "Plantago"
## [77] "Pleopeltis"    "Galium"         "Picea"          "Mirabilis"
## [81] "Carex"         "Carex"         "Triticum"       "Petasites"
## [85] "Rubus"         "Xyris"         "Sporobolus"     "Quercus"
## [89] "Acer"          "Dichanthelium" "Muhlenbergia"   "Euphorbia"
## [93] "Carex"         "Leucanthemum"  "Acer"           "Galium"
## [97] "Solidago"      "Gentiana"       "Acer"           "Verbena"
```

Question 2:

```
data_plant$taxon_name2 = sub("(\\w\\s\\w+).*", "\\1", data_plant$taxon_name)
```

```
## Selects a random number 100 times between 1 and the length of data_plant
i = sample(1:nrow(data_plant), 100, replace=FALSE)
```

```
## Records the genus and species name at the random
taxon2_samp = data_plant$taxon_name2[i]
```

```
print(taxon2_samp)
```

```
## [1] "Fraxinus sp"           "Celtis occidentalis"
## [3] "Chapmannia sp"        "Vaccinium vitis"
## [5] "Parthenocissus quinquefolia" "Geocaulon lividum"
## [7] "Barbarea vulgaris"    "Sphaeralcea coccinea"
```

## [9]	"Vernonia baldwinii"	"Atriplex canescens"
## [11]	"Vaccinium stamineum"	"Commelina communis"
## [13]	"Scleria ciliata"	"Carex aquatilis"
## [15]	"Rubus hispidus"	"Boehmeria cylindrica"
## [17]	"Dalea purpurea"	"Engelmannia peristenia"
## [19]	"Bassia scoparia"	"Viburnum dentatum"
## [21]	"Uvularia sessilifolia"	"Toxicodendron radicans"
## [23]	"Eriophorum vaginatum"	"Agrostis perennans"
## [25]	"Pinus taeda"	"Eriophorum russeolum"
## [27]	"Andropogon ternarius"	"Agrostis scabra"
## [29]	"Quercus geminata"	"Erioneuron pilosum"
## [31]	"Melothria pendula"	"Achnatherum occidentale"
## [33]	"Tephrosia virginiana"	"Asimina pygmaea"
## [35]	"Sorghastrum secundum"	"Cornus drummondii"
## [37]	"Scleria sp"	"Cirsium arvense"
## [39]	"Symphyotrichum lateriflorum"	"Distictis lactiflora"
## [41]	"Krameria lanceolata"	"Smilax herbacea"
## [43]	"Equisetum arvense"	"Microstegium vimineum"
## [45]	"Schaefferia frutescens"	"Lycopodium annotinum"
## [47]	"Ledum palustre"	"Bouteloua gracilis"
## [49]	"Picea engelmannii"	"Sphaeralcea coccinea"
## [51]	"Amphiachyris dracunculoides"	"Polygonum bistorta"
## [53]	"Quercus sp"	"Psoralidium tenuiflorum"
## [55]	"Triticum aestivum"	"Ilex decidua"
## [57]	"Artemisia frigida"	"Cephalanthus occidentalis"
## [59]	"Sedum sp"	"Verbesina occidentalis"
## [61]	"Vaccinium oxycoccos"	"Gratiola pilosa"
## [63]	"Dryopteris carthusiana"	"Vaccinium sp"
## [65]	"Fouquieria sp"	"Bromus diandrus"
## [67]	"Sericocarpus tortifolius"	"Heterotheca canescens"
## [69]	"Liriodendron tulipifera"	"Schizachyrium scoparium"
## [71]	"Toxicodendron radicans"	"Ipomoea pandurata"
## [73]	"Rosa arkansana"	"Botrychium virginianum"
## [75]	"Salvia reflexa"	"Oreoxis alpina"
## [77]	"Hamamelis virginiana"	"Allionia incarnata"
## [79]	"Potentilla diversifolia"	"Gaultheria shallon"
## [81]	"Evax prolifera"	"Betula lenta"
## [83]	"Chamaesyce florida"	"Physalis sp"
## [85]	"Stenaria nigricans"	"Carex spp"
## [87]	"Chimaphila sp"	"Carex aquatilis"
## [89]	"Zizia aptera"	"Pilea pumila"
## [91]	"Scoparia dulcis"	"Robinia pseudoacacia"
## [93]	"Thelesperma filifolium"	"Veronica officinalis"
## [95]	"Liriodendron tulipifera"	"Maianthemum racemosum"
## [97]	"Quercus velutina"	"Pityopsis graminifolia"
## [99]	"Dichanthelium acuminatum"	"Lupinus concinnus"

Question 3

```
data_plant$sample_area_m2 == "1"
```

```
## [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```
## [99807] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99821] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99835] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99849] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99863] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99877] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99891] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99905] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99919] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99933] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99947] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99961] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99975] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [99989] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [ reached getOption("max.print") -- omitted 815540 entries ]
```