

# G4.P-1

David Emanuel Craciunescu      Laura Pérez Medeiro

October 15, 2019

## Introduction

The main purpose of this introductory laboratorio practice is to get students accustomed to the R development environment and to teach to these the basics of data science with such environment and language.

Work-wise, the practice is divided in the two following parts:

### Development on provided dataset

In this section, the student will have to formalize and develop on a provided exercise and given dataset by the professor. This does not only enable the professor to analyze if such student has completely and correctly understood the contents of the laboratory practice, but it also serves as a starting point for any student unable to face the workload of such project without any kind of previous guidance or help.

This specific practice provides the dataset *satellites.txt*, which contains the **names** and **radii** of some of the most common moons of Uranus. Working with this dataset has been thoroughly explained by the professor in the almost three laboratory classes dedicated to this specific practice, so an explanation or any kind of further analysis into this specific dataset is considered not only redundant, but time-wasting as well.

### Development on obtained dataset

This section of the laboratory practice aims to create a deeper understanding in the nature of dataset retrieval by the students and to enhance their ability in the labors of recognising good quality data sources and valid information retrieval from the various analysis performed on these datasets.

The chosen data source for this exercise was scraped off the webpage *www.a-z-animals.com*, and the data itself is formed by tuples of the names of animals and their respective life expentancies. A Python scraping script has been developed solely with the objective to obtain this data.

Also, as extra work, the authors have completed the analysis of all the mentioned datasets in Python as well.

## Data analysis

### Satelites dataset

This dataset has been provided in the *txt* format. In order to input it into R, the following command must be used.

```
> while (!"satelites.txt" %in% list.files(getwd()))
+ {
+   print("Data file not found. Add \"satelites.txt\" to the current directory.")
+   invisible(readline(prompt="Press [enter] to continue"))
+ }
> satelites <- read.table("satelites.txt")
> satelites
```

	nombre	radio
1	CORDELIA	13
2	OFELIA	16
3	BIANCA	22
4	CRESIDA	33
5	LESDEMONA	39
6	JULIETA	42
7	ROSALINDA	27
8	BELINDA	34
9	LUNA-1986U1020	20
10	CALIBANO	30
11	LUNA-119	20
12	LUNA_119U2	15

```
> radius <- satelites $radio
> radius

[1] 13 16 22 33 39 42 27 34 20 30 20 15

>
```

When loading external datasets into R, it is important to take into account that the working directory must be the same as the file's directory when calling `read.table()`. Otherwise, one the route to where the found can be found must be indicated.

Once the data has been read, the authors will proceed to analyze it in the following way:

### Absolute and relative frequencies

#### Absolute Frequency

```
> absoluteFreq <- function(set) {table(set)}
> absoluteFreq(radius)
```

set
13 15 16 20 22 27 30 33 34 39 42
1 1 1 2 1 1 1 1 1 1 1

### Cumulative Absolute Frequency

```
> cumAbsoluteFreq <- function(set) {cumsum(absoluteFreq(set))}
> cumAbsoluteFreq(radius)

13 15 16 20 22 27 30 33 34 39 42
 1  2  3  5  6  7  8  9 10 11 12
```

### Relative Frequency

```
> relativeFreq <- function(set) {table(set) / length(set)}
> relativeFreq(radius)

set
      13      15      16      20      22      27      30
0.08333333 0.08333333 0.08333333 0.16666667 0.08333333 0.08333333 0.08333333
      33      34      39      42
0.08333333 0.08333333 0.08333333 0.08333333
```

### Cumulative Relative Frequency

```
> cumRelativeFreq <- function(set) {cumsum(relativeFreq(set))}
> cumRelativeFreq(radius)

      13      15      16      20      22      27      30
0.08333333 0.16666667 0.25000000 0.41666667 0.50000000 0.58333333 0.66666667
      33      34      39      42
0.75000000 0.83333333 0.91666667 1.00000000
```

### Arithmetic mean

```
> arithmeticMean <- function(set, usrTrim = 0) (mean(set, trim = usrTrim))
> arithmeticMean(radius)

[1] 25.91667
```

### Measures of dispersion

For this specific section, the following webpage has been used as a <http://iridl.ldeo.columbia.edu/dochelp/Stat/>

#### Range

```
> range <- function(set) {max(set) - min(set)}
> range(radius)

[1] 29
```

#### Standard Deviation

```
> stdDeviation <- function(set)
+ {
+ sd(set) * (sqrt((length(set) - 1) / length(set)))
+ }
> stdDeviation(radius)
```

```
[1] 9.277736
```

#### Variance

```
> variance <- function(set) {var(set) * (length(set) - 1 / length(set))}  
> variance(radius)
```

```
[1] 1118.993
```

#### Root mean square

```
> rootMeanSqr <- function(set) {sqrt(mean(set ^ 2))}  
> rootMeanSqr(radius)
```

```
[1] 27.52726
```

#### Root mean square anomaly

```
> rootMeanSqrAn <- function(set) {sqrt(sum(set - mean(set)) ^ 2) / length(set)}  
> rootMeanSqrAn(radius)
```

```
[1] 1.184238e-15
```

#### Interquartile

```
> interQuartRange <- function(set) {IQR(set)}  
> interQuartRange(radius)
```

```
[1] 14.25
```

#### Median absolute deviation

```
> medAbsDeviation <- function(set) {mad(set)}  
> medAbsDeviation(radius)
```

```
[1] 12.6021
```

#### Measures of order

##### Median

```
> getMedian <- function(set) {median(set)}  
> getMedian(radius)
```

```
[1] 24.5
```

##### Mode

```
> getMode <- function(set)  
+ {  
+   uniqueVal <- unique(set)  
+   uniqueVal[which.max(tabulate(match(set, uniqueVal)))]  
+ }  
> getMode(radius)
```

```
[1] 20
```

## Quartiles

```
> getQuartiles <- function(set) {quantile(set)}  
> getQuartiles(radius)  
  
 0%   25%   50%   75%  100%  
13.00 19.00 24.50 33.25 42.00
```

## 54th quantile

```
> getQuantiles <- function(set, range = 0) {quantile(set, probs = range)}  
> getQuantiles(radius)  
  
0%  
13
```

## Cardata dataset

The same analysis the authors have performed on the previous dataset will be performed on the Cardata dataset. This time, the variable to use will be called *mpg* and the 54th quantile and the frequencies are not needed.

In order to analyze *.sav* format, R needs to import the *foreign* library.

```
> library(foreign)
```

Once the file is read, only the data related to *mpg* is going to matter. Also, there may be empty rows or NAs in these records, one must filter these in order to perform a correct statistical analysis.

```
> dataset = read.spss("cardata.sav", to.data.frame=TRUE)  
> mpg = dataset$mpg  
> mpg = mpg[!is.na(mpg)]
```

Once the data is prepared, the exact same functions as the previous section can be used.

## Arithmetic mean

```
> arithmeticMean(mpg)  
  
[1] 28.79351
```

## Measures of dispersion

### Range

```
> range(mpg)  
  
[1] 31.1
```

### Standard Deviation

```
> stdDeviation(mpg)
```

```
[1] 7.353219
```

#### **Variance**

```
> variance(mpg)
```

```
[1] 8380.823
```

#### **Root mean square**

```
> rootMeanSqr(mpg)
```

```
[1] 29.7176
```

#### **Root mean square anomaly**

```
> rootMeanSqrAn(mpg)
```

```
[1] 1.522592e-15
```

#### **Interquartile range**

```
> interQuartRange(mpg)
```

```
[1] 11.725
```

#### **Median absolute deviation**

```
> medAbsDeviation(mpg)
```

```
[1] 8.37669
```

#### **Measures of order**

##### **Median**

```
> getMedian(mpg)
```

```
[1] 28.9
```

##### **Mode**

```
> getMode(mpg)
```

```
[1] 36
```

#### **Animal dataset**

```
> animals <- read.csv(head= T, sep=",", "animals2.csv" )  
> animals
```

	name	lifespan
1	Aardvark	23.0
2	Abyssinian	15.0
3	Adelie Penguin	10.0
4	Affenpinscher	12.0
5	Afghan Hound	14.0
6	African Bush Elephant	60.0
7	African Civet	15.0
8	African Clawed Frog	8.0
9	African Forest Elephant	60.0
10	African Palm Civet	15.0
11	African Penguin	10.0
12	African Tree Toad	3.0
13	African Wild Dog	10.0
14	Ainu Dog	12.0
15	Airedale Terrier	12.0
16	Akbash	11.0
17	Akita	10.0
18	Alaskan Malamute	14.0
19	Albatross	12.0
20	Aldabra Giant Tortoise	80.0
21	Alligator	30.0
22	Alpine Dachsbracke	12.0
23	American Bulldog	14.0
24	American Cocker Spaniel	15.0
25	American Coonhound	12.0
26	American Eskimo Dog	15.0
27	American Foxhound	11.0
28	American Pit Bull Terrier	12.0
29	American Staffordshire Terrier	12.0
30	American Water Spaniel	12.0
31	Anatolian Shepherd Dog	13.0
32	Angelfish	8.0
33	Ant	3.0
34	Anteater	9.0
35	Antelope	10.0
36	Appenzeller Dog	12.0
37	Arctic Fox	7.0
38	Arctic Hare	3.0
39	Arctic Wolf	7.0
40	Armadillo	4.0
41	Asian Elephant	55.0
42	Asian Giant Hornet	3.0
43	Asian Palm Civet	15.0
44	Australian Cattle Dog	15.0
45	Australian Kelpie Dog	14.0
46	Australian Mist	14.0
47	Australian Shepherd	14.0
48	Australian Terrier	14.0
49	Avocet	10.0

50	Axolotl	10.0
51	Aye Aye	10.0
52	Baboon	15.0
53	Bactrian Camel	35.0
54	Badger	4.0
55	Balinese	15.0
56	Banded Palm Civet	15.0
57	Bandicoot	5.0
58	Barb	3.0
59	Barn Owl	5.0
60	Barnacle	8.0
61	Barracuda	10.0
62	Basenji Dog	12.0
63	Basking Shark	20.0
64	Basset Hound	12.0
65	Bat	10.0
66	Bavarian Mountain Hound	14.0
67	Beagle	15.0
68	Bear	15.0
69	Bearded Collie	14.0
70	Bearded Dragon	6.0
71	Beaver	15.0
72	Bedlington Terrier	16.0
73	Beetle	3.0
74	Bengal Tiger	18.0
75	Bernese Mountain Dog	8.0
76	Bichon Frise	15.0
77	Binturong	10.0
78	Bird	1.0
79	Birds Of Paradise	5.0
80	Birman	20.0
81	Bison	15.0
82	Black Bear	15.0
83	Black Rhinoceros	47.5
84	Black Russian Terrier	11.0
85	Black Widow Spider	3.0
86	Bloodhound	12.0
87	Blue Lacy Dog	16.0
88	Blue Whale	30.0
89	Bluetick Coonhound	12.0
90	Bobcat	12.0
91	Bolognese Dog	14.0
92	Bombay	13.0
93	Bongo	10.0
94	Bonobo	30.0
95	Booby	12.0
96	Border Collie	20.0
97	Border Terrier	15.0
98	Bornean Orang-utan	30.0
99	Borneo Elephant	55.0



100	Boston Terrier	15.0
101	Bottle Nosed Dolphin	20.0
102	Boxer Dog	11.0
103	Boykin Spaniel	16.0
104	Brazilian Terrier	14.0
105	Brown Bear	20.0
106	Budgerigar	3.0
107	Buffalo	15.0
108	Bull Mastiff	10.0
109	Bull Shark	16.0
110	Bull Terrier	15.0
111	Bulldog	14.0
112	Bullfrog	6.0
113	Bumble Bee	1.0
114	Burmese	12.0
115	Burrowing Frog	10.0
116	Butterfly	40.0
117	Butterfly Fish	6.0
118	Caiman	20.0
119	Caiman Lizard	10.0
120	Cairn Terrier	14.0
121	Camel	40.0
122	Canaan Dog	15.0
123	Capybara	8.0
124	Caracal	12.0
125	Carolina Dog	19.0
126	Cassowary	40.0
127	Cat	15.0
128	Caterpillar	3.0
129	Catfish	8.0
130	Cavalier King Charles Spaniel	15.0
131	Centipede	5.0
132	Cesky Fousek	15.0
133	Chameleon	4.0
134	Chamois	18.0
135	Cheetah	10.0
136	Chesapeake Bay Retriever	13.0
137	Chicken	2.0
138	Chihuahua	18.0
139	Chimpanzee	50.0
140	Chinchilla	10.0
141	Chinese Crested Dog	13.0
142	Chinook	14.0
143	Chinstrap Penguin	15.0
144	Chipmunk	4.0
145	Chow Chow	10.0
146	Cichlid	5.0
147	Clouded Leopard	11.0
148	Clown Fish	4.0
149	Clumber Spaniel	15.0

150	Coati	11.5
151	Cockroach	18.0
152	Collared Peccary	10.0
153	Collie	13.0
154	Common Buzzard	12.0
155	Common Frog	5.0
156	Common Loon	15.0
157	Common Toad	2.0
158	Coral	15.0
159	Cottontop Tamarin	8.0
160	Cougar	10.0
161	Cow	12.0
162	Coyote	10.0
163	Crab	1.0
164	Crab-Eating Macaque	15.0
165	Crane	15.0
166	Crested Penguin	15.0
167	Crocodile	20.0
168	Cross River Gorilla	35.0
169	Curly Coated Retriever	14.0
170	Cuscus	8.0
171	Cuttlefish	1.0
172	Dachshund	15.0
173	Darwin's Frog	10.0
174	Deer	10.0
175	Desert Tortoise	25.0
176	Deutsche Bracke	12.0
177	Dhole	10.0
178	Dingo	7.0
179	Discus	8.0
180	Doberman Pinscher	12.0
181	Dodo	10.0
182	Dog	15.0
183	Dogo Argentino	13.0
184	Dogue De Bordeaux	12.0
185	Dolphin	20.0
186	Donkey	15.0
187	Dormouse	2.0
188	Dragonfly	6.0
189	Drever	14.0
190	Duck	4.0
191	Dugong	50.0
192	Dunker	14.0
193	Dusky Dolphin	18.0
194	Dwarf Crocodile	40.0
195	Eagle	15.0
196	Earwig	1.0
197	Eastern Gorilla	35.0
198	Eastern Lowland Gorilla	35.0
199	Echidna	15.0

200	Edible Frog	5.0
201	Egyptian Mau	14.0
202	Electric Eel	15.0
203	Elephant	55.0
204	Elephant Seal	18.0
205	Elephant Shrew	2.0
206	Emperor Penguin	15.0
207	Emperor Tamarin	8.0
208	Emu	12.0
209	English Cocker Spaniel	15.0
210	English Shepherd	16.0
211	English Springer Spaniel	12.0
212	Entlebucher Mountain Dog	12.0
213	Epagneul Pont Audemer	13.0
214	Eskimo Dog	14.0
215	Estrela Mountain Dog	16.0
216	Falcon	12.0
217	Fennec Fox	10.0
218	Ferret	7.0
219	Field Spaniel	14.0
220	Fin Whale	50.0
221	Fire-Bellied Toad	10.0
222	Fish	1.0
223	Fishing Cat	10.0
224	Flamingo	15.0
225	Flat Coat Retriever	12.0
226	Flounder	3.0
227	Fly	25.0
228	Flying Squirrel	5.0
229	Fossa	15.0
230	Fox	3.0
231	Fox Terrier	14.0
232	French Bulldog	12.0
233	Frigatebird	15.0
234	Frilled Lizard	10.0
235	Frog	1.0
236	Fur Seal	12.0
237	Galapagos Penguin	15.0
238	Galapagos Tortoise	125.0
239	Gar	10.0
240	Gecko	5.5
241	Gentoo Penguin	15.0
242	Geoffroys Tamarin	8.0
243	Gerbil	4.0
244	German Pinscher	15.0
245	German Shepherd	12.0
246	Gharial	30.0
247	Giant African Land Snail	6.5
248	Giant Clam	60.0
249	Giant Panda Bear	20.0

250	Gibbon	32.5
251	Gila Monster	25.0
252	Giraffe	20.0
253	Glass Lizard	10.0
254	Glow Worm	5.0
255	Goat	12.5
256	Golden Lion Tamarin	11.5
257	Golden Oriole	8.0
258	Golden Retriever	12.0
259	Goose	12.0
260	Gopher	4.0
261	Gorilla	42.5
262	Grasshopper	1.0
263	Great Dane	7.0
264	Great White Shark	30.0
265	Greater Swiss Mountain Dog	12.0
266	Green Bee-Eater	12.0
267	Greenland Dog	12.0
268	Grey Mouse Lemur	3.0
269	Grey Reef Shark	25.0
270	Grey Seal	21.5
271	Greyhound	11.0
272	Grizzly Bear	15.0
273	Grouse	1.0
274	Guinea Fowl	10.0
275	Guinea Pig	3.0
276	Guppy	2.0
277	Hammerhead Shark	22.5
278	Hamster	2.5
279	Hare	5.0
280	Harrier	16.0
281	Havanese	15.0
282	Hedgehog	4.5
283	Hercules Beetle	3.0
284	Hermit Crab	5.5
285	Heron	15.0
286	Highland Cattle	18.5
287	Himalayan	14.0
288	Hippopotamus	30.0
289	Honey Bee	6.0
290	Horn Shark	12.0
291	Horned Frog	6.5
292	Horse	27.5
293	Horseshoe Crab	20.0
294	Howler Monkey	17.5
295	Human	70.0
296	Humboldt Penguin	15.0
297	Hummingbird	3.0
298	Humpback Whale	55.0
299	Hyena	22.5

300	Ibis	8.0
301	Ibizan Hound	14.0
302	Iguana	17.5
303	Impala	13.5
304	Indian Elephant	55.0
305	Indian Palm Squirrel	3.0
306	Indian Rhinoceros	47.5
307	Indian Star Tortoise	30.0
308	Indochinese Tiger	18.0
309	Indri	15.0
310	Insect	2.0
311	Irish Setter	16.0
312	Irish WolfHound	9.0
313	Jack Russel	16.0
314	Jackal	11.5
315	Jaguar	12.0
316	Japanese Chin	14.0
317	Japanese Macaque	25.0
318	Javan Rhinoceros	37.5
319	Javanese	14.0
320	Jellyfish	3.0
321	Kakapo	57.5
322	Kangaroo	7.0
323	Keel Billed Toucan	15.0
324	Killer Whale	55.0
325	King Crab	22.5
326	King Penguin	15.0
327	Kingfisher	6.0
328	Kiwi	8.0
329	Koala	15.0
330	Komodo Dragon	25.0
331	Kudu	11.0
332	Labradoodle	12.0
333	Labrador Retriever	15.0
334	Ladybird	2.0
335	Leaf-Tailed Gecko	2.0
336	Lemming	2.0
337	Lemur	12.0
338	Leopard	10.0
339	Leopard Cat	10.0
340	Leopard Seal	22.0
341	Leopard Tortoise	50.0
342	Liger	18.0
343	Lion	8.0
344	Lionfish	10.0
345	Little Penguin	10.0
346	Lizard	15.5
347	Llama	17.5
348	Lobster	50.0
349	Long-Eared Owl	40.0

350	Lynx	16.0
351	Macaroni Penguin	17.5
352	Macaw	50.0
353	Magellanic Penguin	15.0
354	Magpie	8.0
355	Maine Coon	12.0
356	Malayan Civet	15.0
357	Malayan Tiger	18.0
358	Maltese	17.0
359	Manatee	60.0
360	Mandrill	24.0
361	Manta Ray	15.0
362	Marine Toad	10.0
363	Markhor	10.0
364	Marsh Frog	5.0
365	Masked Palm Civet	15.0
366	Mastiff	12.0
367	Mayfly	1.5
368	Meerkat	10.0
369	Millipede	7.0
370	Minke Whale	40.0
371	Mole	4.5
372	Molly	3.0
373	Mongoose	12.5
374	Monitor Lizard	19.0
375	Monkey	20.0
376	Monte Iberia Eleuth	1.0
377	Moorhen	1.0
378	Moose	13.0
379	Moray Eel	20.0
380	Moth	40.0
381	Mountain Gorilla	42.5
382	Mountain Lion	15.0
383	Mouse	3.5
384	Mule	17.5
385	Neanderthal	42.5
386	Neapolitan Mastiff	10.0
387	Newfoundland	10.0
388	Newt	8.5
389	Nightingale	1.0
390	Norfolk Terrier	15.0
391	Norwegian Forest	15.0
392	Numbat	6.0
393	Nurse Shark	22.5
394	Ocelot	8.0
395	Octopus	8.5
396	Okapi	20.0
397	Old English Sheepdog	12.0
398	Olm	10.0
399	Opossum	4.5

400	Orang-utan	30.0
401	Ostrich	50.0
402	Otter	20.0
403	Oyster	1.0
404	Pademelon	6.0
405	Panther	12.0
406	Parrot	40.0
407	Patas Monkey	12.0
408	Peacock	12.0
409	Pekingese	13.0
410	Pelican	16.0
411	Penguin	20.0
412	Persian	19.0
413	Pheasant	7.0
414	Pied Tamarin	8.0
415	Pig	11.5
416	Pika	4.5
417	Pike	12.0
418	Pink Fairy Armadillo	5.0
419	Piranha	20.0
420	Platypus	9.0
421	Pointer	13.0
422	Poison Dart Frog	3.0
423	Polar Bear	20.0
424	Pond Skater	1.0
425	Poodle	16.0
426	Pool Frog	1.0
427	Porcupine	10.0
428	Possum	6.5
429	Prawn	2.0
430	Proboscis Monkey	15.0
431	Puffer Fish	4.0
432	Puffin	15.0
433	Pug	14.0
434	Puma	10.0
435	Purple Emperor	1.0
436	Puss Moth	3.0
437	Pygmy Hippopotamus	30.0
438	Pygmy Marmoset	10.0
439	Quail	3.0
440	Quetzal	20.0
441	Quokka	5.0
442	Quoll	4.5
443	Rabbit	6.0
444	Raccoon	14.0
445	Raccoon Dog	3.0
446	Radiated Tortoise	30.0
447	Ragdoll	14.0
448	Rat	3.5
449	Red Knee Tarantula	25.0

450	Red Panda	8.0
451	Red Wolf	10.0
452	Reindeer	13.5
453	Rhinoceros	37.5
454	River Dolphin	15.0
455	River Turtle	20.0
456	Robin	1.0
457	Rock Hyrax	5.0
458	Rockhopper Penguin	15.0
459	Roseate Spoonbill	10.0
460	Rottweiler	9.0
461	Royal Penguin	15.0
462	Russian Blue	14.0
463	Sabre-Toothed Tiger	20.0
464	Saint Bernard	10.0
465	Salamander	12.5
466	Sand Lizard	5.0
467	Saola	8.0
468	Scorpion	14.5
469	Scorpion Fish	10.0
470	Sea Dragon	2.0
471	Sea Lion	18.5
472	Sea Otter	13.5
473	Sea Slug	2.5
474	Sea Squirt	18.5
475	Sea Turtle	70.0
476	Sea Urchin	107.5
477	Seahorse	2.0
478	Seal	20.0
479	Serval	10.0
480	Sheep	7.5
481	Shih Tzu	15.0
482	Shrimp	1.5
483	Siamese	12.0
484	Siamese Fighting Fish	1.0
485	Siberian	15.0
486	Siberian Husky	14.0
487	Siberian Tiger	18.0
488	Silver Dollar	2.0
489	Skunk	6.5
490	Sloth	32.5
491	Slow Worm	10.0
492	Snail	1.0
493	Snake	30.0
494	Snapping Turtle	26.0
495	Snowshoe	14.0
496	Snowy Owl	10.0
497	Somali	19.0
498	South China Tiger	18.0
499	Spadefoot Toad	6.0



500	Sparrow	4.0
501	Spectacled Bear	20.0
502	Sperm Whale	50.0
503	Spider Monkey	21.0
504	Spiny Dogfish	52.5
505	Sponge	22.5
506	Squid	17.5
507	Squirrel	5.0
508	Squirrel Monkey	15.0
509	Sri Lankan Elephant	55.0
510	Staffordshire Bull Terrier	14.0
511	Stag Beetle	3.0
512	Starfish	15.0
513	Stellers Sea Cow	50.0
514	Stick Insect	1.5
515	Stingray	15.0
516	Stoat	5.0
517	Striped Rocket Frog	10.0
518	Sumatran Elephant	55.0
519	Sumatran Orang-utan	30.0
520	Sumatran Rhinoceros	37.5
521	Sumatran Tiger	18.0
522	Sun Bear	15.0
523	Swan	8.0
524	Tang	8.0
525	Tapanuli Orang-utan	30.0
526	Tapir	22.5
527	Tarsier	12.0
528	Tasmanian Devil	6.5
529	Tawny Owl	4.0
530	Termite	2.0
531	Tetra	3.5
532	Thorny Devil	16.0
533	Tibetan Mastiff	15.0
534	Tiffany	15.0
535	Tiger	21.5
536	Tiger Salamander	12.5
537	Tiger Shark	35.0
538	Tortoise	90.0
539	Toucan	12.0
540	Tree Frog	3.0
541	Tropicbird	10.0
542	Tuatara	75.0
543	Turkey	5.5
544	Turkish Angora	14.0
545	Uakari	16.0
546	Uguisu	2.0
547	Umbrellabird	12.0
548	Vampire Bat	8.0
549	Vervet Monkey	12.0

550	Vulture	20.0
551	Wallaby	13.5
552	Walrus	40.0
553	Warthog	15.0
554	Wasp	22.0
555	Water Buffalo	20.0
556	Water Dragon	15.0
557	Water Vole	0.5
558	Weasel	1.0
559	Welsh Corgi	13.0
560	West Highland Terrier	15.0
561	Western Gorilla	35.0
562	Western Lowland Gorilla	35.0
563	Whale Shark	65.0
564	Whippet	15.0
565	White Faced Capuchin	28.0
566	White Tiger	10.0
567	Wild Boar	15.0
568	Wildebeest	17.5
569	Wolf	11.0
570	Wolverine	12.5
571	Wombat	23.0
572	Woodlouse	2.0
573	Woodpecker	8.5
574	Woolly Mammoth	60.0
575	Woolly Monkey	8.5
576	Wrasse	3.0
577	X-Ray Tetra	2.0
578	Yak	17.5
579	Yellow-Eyed Penguin	10.0
580	Yorkshire Terrier	15.0
581	Zebra	20.0
582	Zebra Shark	15.0
583	Zebu	14.0
584	Zonkey	15.0
585	Zorse	15.0

```
> lifespan <- animals $lifespan
```

```
> lifespan
```

[1]	23.0	15.0	10.0	12.0	14.0	60.0	15.0	8.0	60.0	15.0	10.0	3.0
[13]	10.0	12.0	12.0	11.0	10.0	14.0	12.0	80.0	30.0	12.0	14.0	15.0
[25]	12.0	15.0	11.0	12.0	12.0	12.0	13.0	8.0	3.0	9.0	10.0	12.0
[37]	7.0	3.0	7.0	4.0	55.0	3.0	15.0	15.0	14.0	14.0	14.0	14.0
[49]	10.0	10.0	10.0	15.0	35.0	4.0	15.0	15.0	5.0	3.0	5.0	8.0
[61]	10.0	12.0	20.0	12.0	10.0	14.0	15.0	15.0	14.0	6.0	15.0	16.0
[73]	3.0	18.0	8.0	15.0	10.0	1.0	5.0	20.0	15.0	15.0	47.5	11.0
[85]	3.0	12.0	16.0	30.0	12.0	12.0	14.0	13.0	10.0	30.0	12.0	20.0
[97]	15.0	30.0	55.0	15.0	20.0	11.0	16.0	14.0	20.0	3.0	15.0	10.0
[109]	16.0	15.0	14.0	6.0	1.0	12.0	10.0	40.0	6.0	20.0	10.0	14.0

[121]	40.0	15.0	8.0	12.0	19.0	40.0	15.0	3.0	8.0	15.0	5.0	15.0
[133]	4.0	18.0	10.0	13.0	2.0	18.0	50.0	10.0	13.0	14.0	15.0	4.0
[145]	10.0	5.0	11.0	4.0	15.0	11.5	18.0	10.0	13.0	12.0	5.0	15.0
[157]	2.0	15.0	8.0	10.0	12.0	10.0	1.0	15.0	15.0	15.0	20.0	35.0
[169]	14.0	8.0	1.0	15.0	10.0	10.0	25.0	12.0	10.0	7.0	8.0	12.0
[181]	10.0	15.0	13.0	12.0	20.0	15.0	2.0	6.0	14.0	4.0	50.0	14.0
[193]	18.0	40.0	15.0	1.0	35.0	35.0	15.0	5.0	14.0	15.0	55.0	18.0
[205]	2.0	15.0	8.0	12.0	15.0	16.0	12.0	12.0	13.0	14.0	16.0	12.0
[217]	10.0	7.0	14.0	50.0	10.0	1.0	10.0	15.0	12.0	3.0	25.0	5.0
[229]	15.0	3.0	14.0	12.0	15.0	10.0	1.0	12.0	15.0	125.0	10.0	5.5
[241]	15.0	8.0	4.0	15.0	12.0	30.0	6.5	60.0	20.0	32.5	25.0	20.0
[253]	10.0	5.0	12.5	11.5	8.0	12.0	12.0	4.0	42.5	1.0	7.0	30.0
[265]	12.0	12.0	12.0	3.0	25.0	21.5	11.0	15.0	1.0	10.0	3.0	2.0
[277]	22.5	2.5	5.0	16.0	15.0	4.5	3.0	5.5	15.0	18.5	14.0	30.0
[289]	6.0	12.0	6.5	27.5	20.0	17.5	70.0	15.0	3.0	55.0	22.5	8.0
[301]	14.0	17.5	13.5	55.0	3.0	47.5	30.0	18.0	15.0	2.0	16.0	9.0
[313]	16.0	11.5	12.0	14.0	25.0	37.5	14.0	3.0	57.5	7.0	15.0	55.0
[325]	22.5	15.0	6.0	8.0	15.0	25.0	11.0	12.0	15.0	2.0	2.0	2.0
[337]	12.0	10.0	10.0	22.0	50.0	18.0	8.0	10.0	10.0	15.5	17.5	50.0
[349]	40.0	16.0	17.5	50.0	15.0	8.0	12.0	15.0	18.0	17.0	60.0	24.0
[361]	15.0	10.0	10.0	5.0	15.0	12.0	1.5	10.0	7.0	40.0	4.5	3.0
[373]	12.5	19.0	20.0	1.0	1.0	13.0	20.0	40.0	42.5	15.0	3.5	17.5
[385]	42.5	10.0	10.0	8.5	1.0	15.0	15.0	6.0	22.5	8.0	8.5	20.0
[397]	12.0	10.0	4.5	30.0	50.0	20.0	1.0	6.0	12.0	40.0	12.0	12.0
[409]	13.0	16.0	20.0	19.0	7.0	8.0	11.5	4.5	12.0	5.0	20.0	9.0
[421]	13.0	3.0	20.0	1.0	16.0	1.0	10.0	6.5	2.0	15.0	4.0	15.0
[433]	14.0	10.0	1.0	3.0	30.0	10.0	3.0	20.0	5.0	4.5	6.0	14.0
[445]	3.0	30.0	14.0	3.5	25.0	8.0	10.0	13.5	37.5	15.0	20.0	1.0
[457]	5.0	15.0	10.0	9.0	15.0	14.0	20.0	10.0	12.5	5.0	8.0	14.5
[469]	10.0	2.0	18.5	13.5	2.5	18.5	70.0	107.5	2.0	20.0	10.0	7.5
[481]	15.0	1.5	12.0	1.0	15.0	14.0	18.0	2.0	6.5	32.5	10.0	1.0
[493]	30.0	26.0	14.0	10.0	19.0	18.0	6.0	4.0	20.0	50.0	21.0	52.5
[505]	22.5	17.5	5.0	15.0	55.0	14.0	3.0	15.0	50.0	1.5	15.0	5.0
[517]	10.0	55.0	30.0	37.5	18.0	15.0	8.0	8.0	30.0	22.5	12.0	6.5
[529]	4.0	2.0	3.5	16.0	15.0	15.0	21.5	12.5	35.0	90.0	12.0	3.0
[541]	10.0	75.0	5.5	14.0	16.0	2.0	12.0	8.0	12.0	20.0	13.5	40.0
[553]	15.0	22.0	20.0	15.0	0.5	1.0	13.0	15.0	35.0	35.0	65.0	15.0
[565]	28.0	10.0	15.0	17.5	11.0	12.5	23.0	2.0	8.5	60.0	8.5	3.0
[577]	2.0	17.5	10.0	15.0	20.0	15.0	14.0	15.0	15.0			

## Absolute and relative frequencies

### Absolute Frequency

```
> absoluteFreq(lifespan)
```

```
set
```

0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	6.5
1	20	3	17	2	25	3	11	5	17	3	10	5
7	7.5	8	8.5	9	10	11	11.5	12	12.5	13	13.5	14
8	1	23	4	4	56	8	4	52	5	11	4	34

14.5	15	15.5	16	17	17.5	18	18.5	19	20	21	21.5	22
1	83	1	14	1	8	12	3	4	26	1	2	2
22.5	23	24	25	26	27.5	28	30	32.5	35	37.5	40	42.5
6	2	1	7	1	1	1	14	2	7	3	9	3
47.5	50	52.5	55	57.5	60	65	70	75	80	90	107.5	125
2	9	1	8	1	5	1	2	1	1	1	1	1

### Cumulative Absolute Frequency

> cumAbsoluteFreq(lifespan)

0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6	6.5
1	21	24	41	43	68	71	82	87	104	107	117	122
7	7.5	8	8.5	9	10	11	11.5	12	12.5	13	13.5	14
130	131	154	158	162	218	226	230	282	287	298	302	336
14.5	15	15.5	16	17	17.5	18	18.5	19	20	21	21.5	22
337	420	421	435	436	444	456	459	463	489	490	492	494
22.5	23	24	25	26	27.5	28	30	32.5	35	37.5	40	42.5
500	502	503	510	511	512	513	527	529	536	539	548	551
47.5	50	52.5	55	57.5	60	65	70	75	80	90	107.5	125
553	562	563	571	572	577	578	580	581	582	583	584	585

### Relative Frequency

> relativeFreq(lifespan)

set

0.5	1	1.5	2	2.5	3
0.001709402	0.034188034	0.005128205	0.029059829	0.003418803	0.042735043
3.5	4	4.5	5	5.5	6
0.005128205	0.018803419	0.008547009	0.029059829	0.005128205	0.017094017
6.5	7	7.5	8	8.5	9
0.008547009	0.013675214	0.001709402	0.039316239	0.006837607	0.006837607
10	11	11.5	12	12.5	13
0.095726496	0.013675214	0.006837607	0.088888889	0.008547009	0.018803419
13.5	14	14.5	15	15.5	16
0.006837607	0.058119658	0.001709402	0.141880342	0.001709402	0.023931624
17	17.5	18	18.5	19	20
0.001709402	0.013675214	0.020512821	0.005128205	0.006837607	0.044444444
21	21.5	22	22.5	23	24
0.001709402	0.003418803	0.003418803	0.010256410	0.003418803	0.001709402
25	26	27.5	28	30	32.5
0.011965812	0.001709402	0.001709402	0.001709402	0.023931624	0.003418803
35	37.5	40	42.5	47.5	50
0.011965812	0.005128205	0.015384615	0.005128205	0.003418803	0.015384615
52.5	55	57.5	60	65	70
0.001709402	0.013675214	0.001709402	0.008547009	0.001709402	0.003418803
75	80	90	107.5	125	
0.001709402	0.001709402	0.001709402	0.001709402	0.001709402	

### Cumulative Relative Frequency

```
> cumRelativeFreq(lifespan)
```

0.5	1	1.5	2	2.5	3
0.001709402	0.035897436	0.041025641	0.070085470	0.073504274	0.116239316
3.5	4	4.5	5	5.5	6
0.121367521	0.140170940	0.148717949	0.177777778	0.182905983	0.200000000
6.5	7	7.5	8	8.5	9
0.208547009	0.222222222	0.223931624	0.263247863	0.270085470	0.276923077
10	11	11.5	12	12.5	13
0.372649573	0.386324786	0.393162393	0.482051282	0.490598291	0.509401709
13.5	14	14.5	15	15.5	16
0.516239316	0.574358974	0.576068376	0.717948718	0.719658120	0.743589744
17	17.5	18	18.5	19	20
0.745299145	0.758974359	0.779487179	0.784615385	0.791452991	0.835897436
21	21.5	22	22.5	23	24
0.837606838	0.841025641	0.844444444	0.854700855	0.858119658	0.859829060
25	26	27.5	28	30	32.5
0.871794872	0.873504274	0.875213675	0.876923077	0.900854701	0.904273504
35	37.5	40	42.5	47.5	50
0.916239316	0.921367521	0.936752137	0.941880342	0.945299145	0.960683761
52.5	55	57.5	60	65	70
0.962393162	0.976068376	0.977777778	0.986324786	0.988034188	0.991452991
75	80	90	107.5	125	
0.993162393	0.994871795	0.996581197	0.998290598	1.000000000	

### Arithmetic mean

```
> arithmeticMean(lifespan)
```

```
[1] 15.86581
```

### Measures of dispersion

For this specific section, the following webpage has been used as a <http://iridl.ldeo.columbia.edu/dochelp/Stat>

#### Range

```
> range(lifespan)
```

```
[1] 124.5
```

#### Standard Deviation

```
> stdDeviation(lifespan)
```

```
[1] 14.4033
```

#### Variance

```
> variance(lifespan)
```

```
[1] 121568.7
```

#### Root mean square

```
> rootMeanSqr(lifespan)
```

```
[1] 21.42846
```

**Root mean square anomaly**

```
> rootMeanSqrAn(lifespan)
```

```
[1] 3.491984e-16
```

**Interquartile range**

```
> interQuartRange(lifespan)
```

```
[1] 9.5
```

**Median absolute deviation**

```
> medAbsDeviation(lifespan)
```

```
[1] 7.413
```

**Measures of order**

**Median**

```
> getMedian(lifespan)
```

```
[1] 13
```

**Mode**

```
> getMode(lifespan)
```

```
[1] 15
```

**Quartiles**

```
> getQuartiles(lifespan)
```

0%	25%	50%	75%	100%
0.5	8.0	13.0	17.5	125.0

**54th quantile**

```
> getQuantiles(lifespan)
```

0%
0.5