

Zeru-Zhou-project09

November 2, 2021

1 Project 9 – Zeru Zhou

TA Help: NA

Collaboration: NA

- Get help from Dr. Ward's video

1.1 Question 1

```
[5]: benfords_law <- function(digit) {log((digit+1)/digit)/log(10)}
```

```
[6]: sapply(1:9, benfords_law)
```

```
1. 0.301029995663981 2. 0.176091259055681 3. 0.1249387366083 4. 0.0969100130080564
5. 0.0791812460476248 6. 0.0669467896306132 7. 0.0579919469776867 8. 0.0511525224473813
9. 0.0457574905606751
```

```
[7]: benfords_law(7)
```

```
0.0579919469776867
```

Benfords_law is created as a function. When digit is 7, the value returned is 0.05799.

1.2 Question 2

```
[8]: benfords_law <- function(digit) {
      if ((digit < 1) | (digit > 9)) {stop("digit is out of range")}
      log((digit+1)/digit)/log(10)
    }
```

```
[10]: benfords_law(0)
```

```
Error in benfords_law(0): digit is out of range
```

```
Traceback:
```

```
1. benfords_law(0)
2. stop("digit is out of range")    # at line 2 of file <text>
```

Error catching statement is added. When running `benfords_law(0)`, the error message shows up.

1.3 Question 3

```
[11]: benfords_law_old <- function(digit) {  
      if ((digit < 1) | (digit > 9)) {stop("digit is out of range")}  
      log((digit+1)/digit)/log(10)  
    }
```

```
[12]: benfords_law <- function(v) {  
      sapply(v, benfords_law_old)  
    }
```

```
[13]: benfords_law(0:5)
```

```
Error in FUN(X[[i]], ...): digit is out of range  
Traceback:  
  
1. benfords_law(0:5)  
2. sapply(v, benfords_law_old)    # at line 2 of file <text>  
3. lapply(X = X, FUN = FUN, ...)  
4. FUN(X[[i]], ...)  
5. stop("digit is out of range")  # at line 2 of file <text>
```

```
[14]: benfords_law(1:6)
```

```
1. 0.301029995663981 2. 0.176091259055681 3. 0.1249387366083 4. 0.0969100130080564  
5. 0.0791812460476248 6. 0.0669467896306132
```

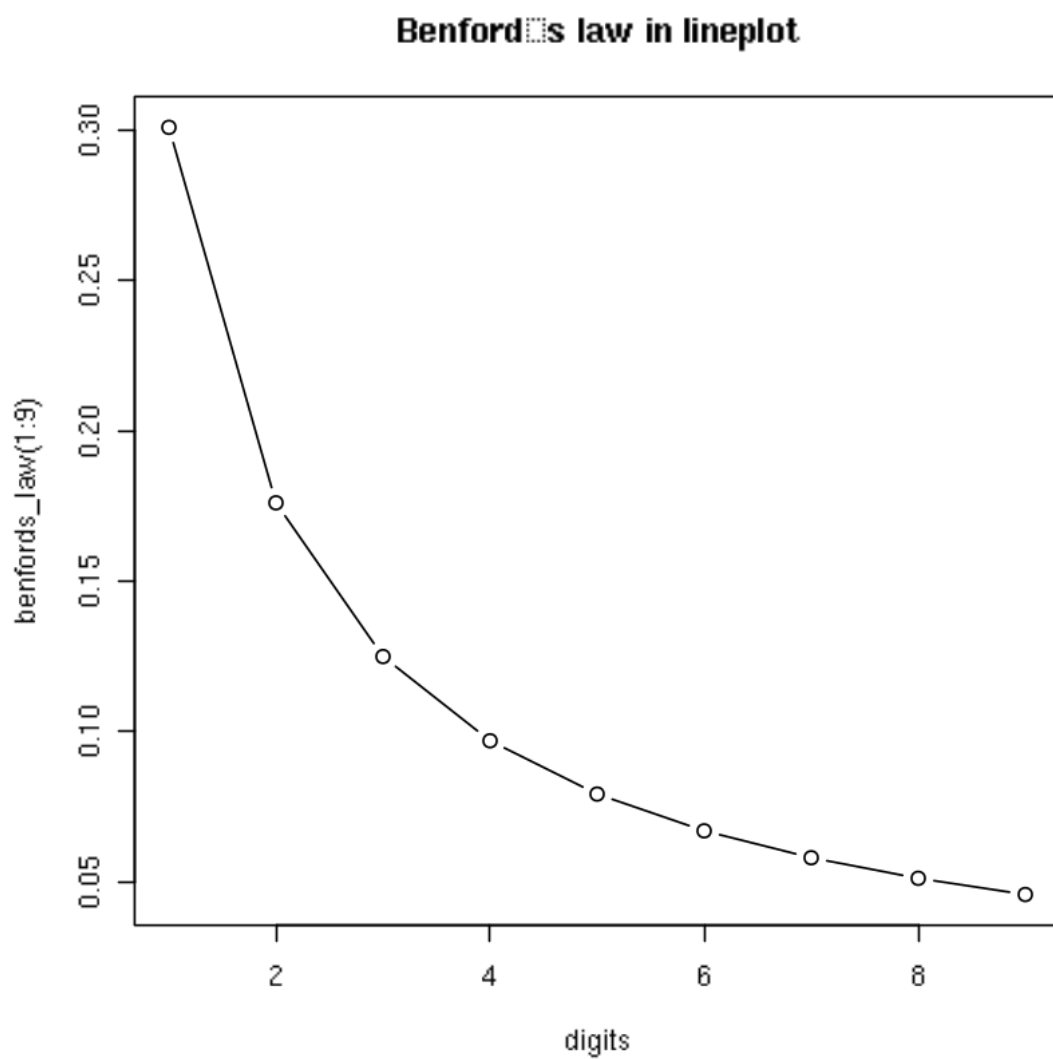
Here we see we vectorized the function by `sapply` function. Results are listed above.

1.4 Question 4

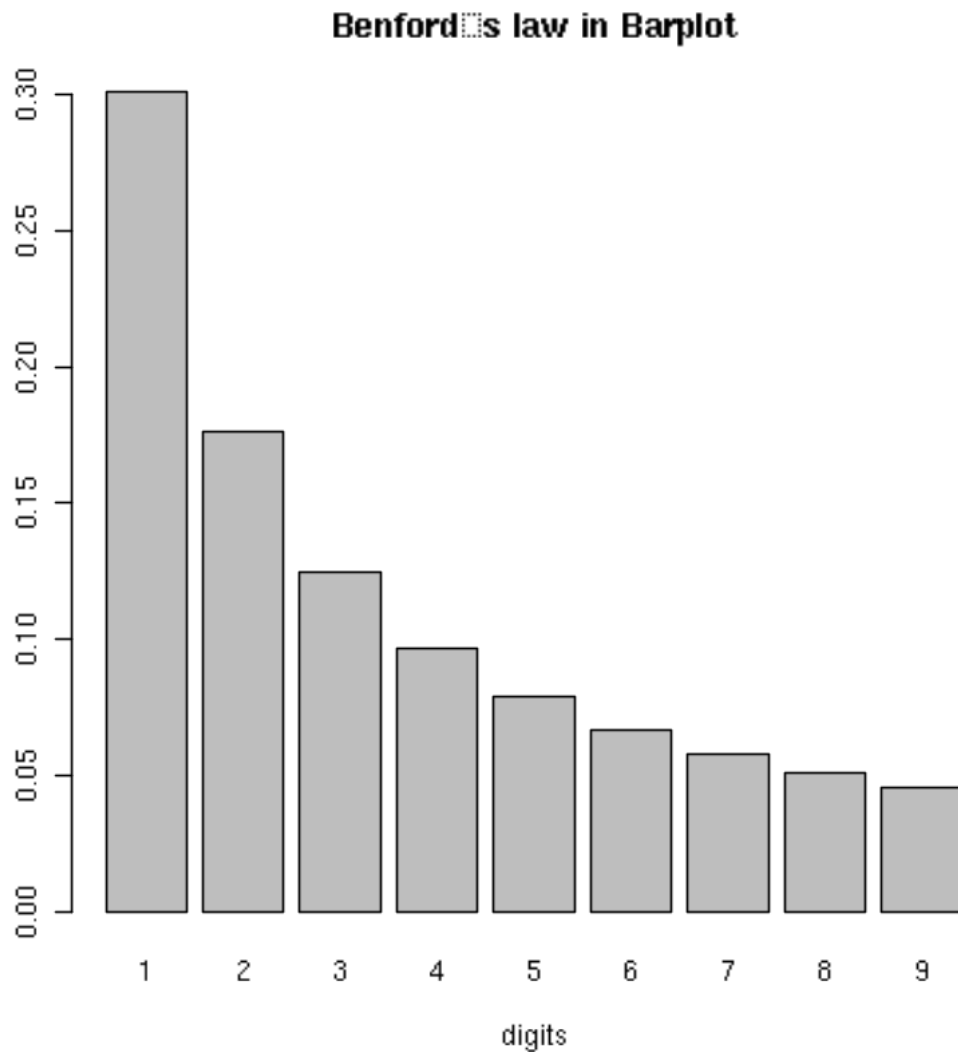
```
[19]: benfords_law(1:9)
```

```
1. 0.301029995663981 2. 0.176091259055681 3. 0.1249387366083 4. 0.0969100130080564  
5. 0.0791812460476248 6. 0.0669467896306132 7. 0.0579919469776867 8. 0.0511525224473813  
9. 0.0457574905606751
```

```
[31]: plot(benfords_law(1:9), xlab="digits", main="Benford's law in lineplot",  
      ↪type="b")
```



```
[30]: barplot(benfords_law(1:9), main="Benford's law in Barplot", xlab="digits",  
names.arg=c("1","2","3","4","5","6","7","8","9"))
```



Here are the lineplot and barplot of the benford's law of all digits.

1.5 Question 5

```
[1]: get_starting_digit <- function(transaction_vector) {  
      as.numeric(substr(transaction_vector,1,1))  
    }
```

```
[2]: get_starting_digit(c(10,400,535))
```

1. 1 2. 4 3. 5

```
[3]: str(get_starting_digit(c(100,2,50,689,1)))
```

```
num [1:5] 1 2 5 6 1
```

Here I use substr function to get the first digit and as.numeric function to convert it as numeric value.

1.6 Pledge

By submitting this work I hereby pledge that this is my own, personal work. I've acknowledged in the designated place at the top of this file all sources that I used to complete said work, including but not limited to: online resources, books, and electronic communications. I've noted all collaboration with fellow students and/or TA's. I did not copy or plagiarize another's work.

As a Boilermaker pursuing academic excellence, I pledge to be honest and true in all that I do. Accountable together – We are Purdue.