Election Fraud?

AUTHOR
Pari Pandya

Load Libraries

```
library(tidyverse)
```

```
── Attaching core tidyverse packages ────────────────────────
tidyverse 2.0.0 ──
✔ dplyr     1.1.3     ✔ readr     2.1.4
✔ forcats   1.0.0     ✔ stringr   1.5.0
✔ ggplot2   3.4.4     ✔ tibble    3.2.1
✔ lubridate 1.9.3     ✔ tidyr     1.3.0
✔ purrr     1.0.2
── Conflicts ──────────────────────────────────────────
tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>)
to force all conflicts to become errors
```

```
library(stat20data)
library(patchwork)
data(iran)
New_York_Data_set <- read_csv("https://raw.githubusercontent.com
```

```
Rows: 806 Columns: 6
── Column specification
────────────────────────────────────────────────────────
Delimiter: ","
chr (4): county, office, party, candidate
dbl (1): votes
lgl (1): district

ℹ Use `spec()` to retrieve the full column specification for
this data.
ℹ Specify the column types or set `show_col_types = FALSE` to
quiet this message.
```

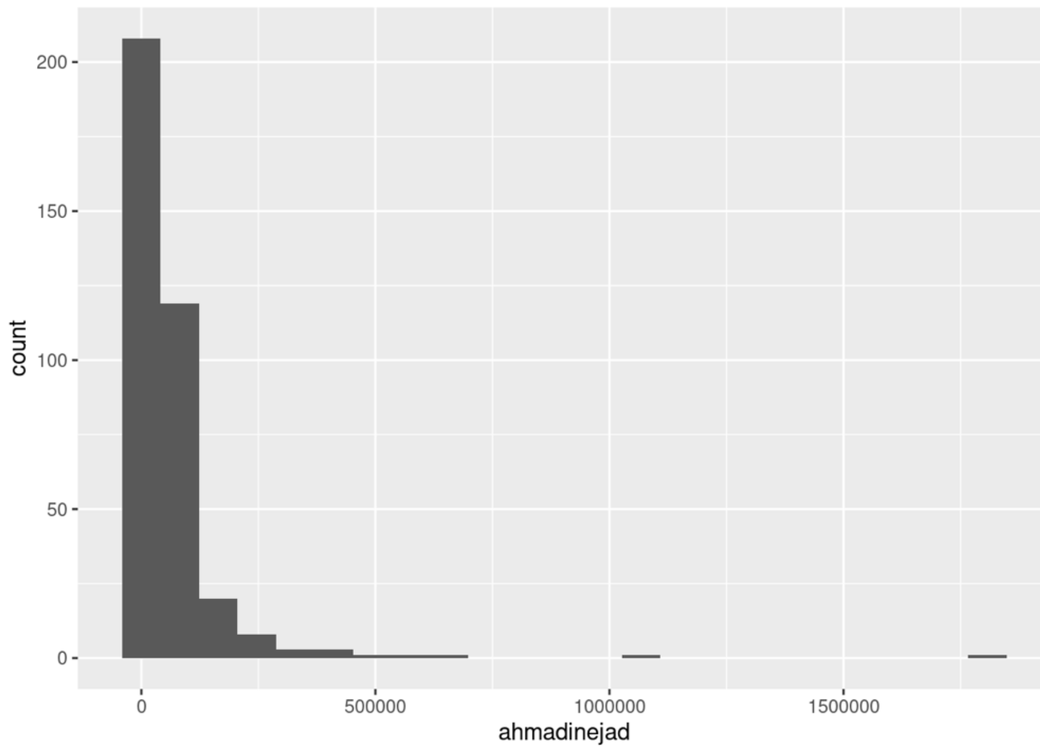
Question 1:

The unit of observation in the Iran data frame is cities.

Question 2:

Plot:

```
ggplot(data = iran, mapping = aes(x = ahmadinejad)) +
  geom_histogram(bins = 23)
```



Numerical Summaries:

```
iran %>%
  summarise("Mean" = mean(ahmadinejad),
            "Center" = sd(ahmadinejad))
```

```
# A tibble: 1 × 2
    Mean  Center
   <dbl>   <dbl>
```

```
1 66981. 130010.
```

Interpretation:

This plot clearly shows a rightward skew. The lack of a normal distribution may likely be an indicator of biased or tampered data.

Question 3:

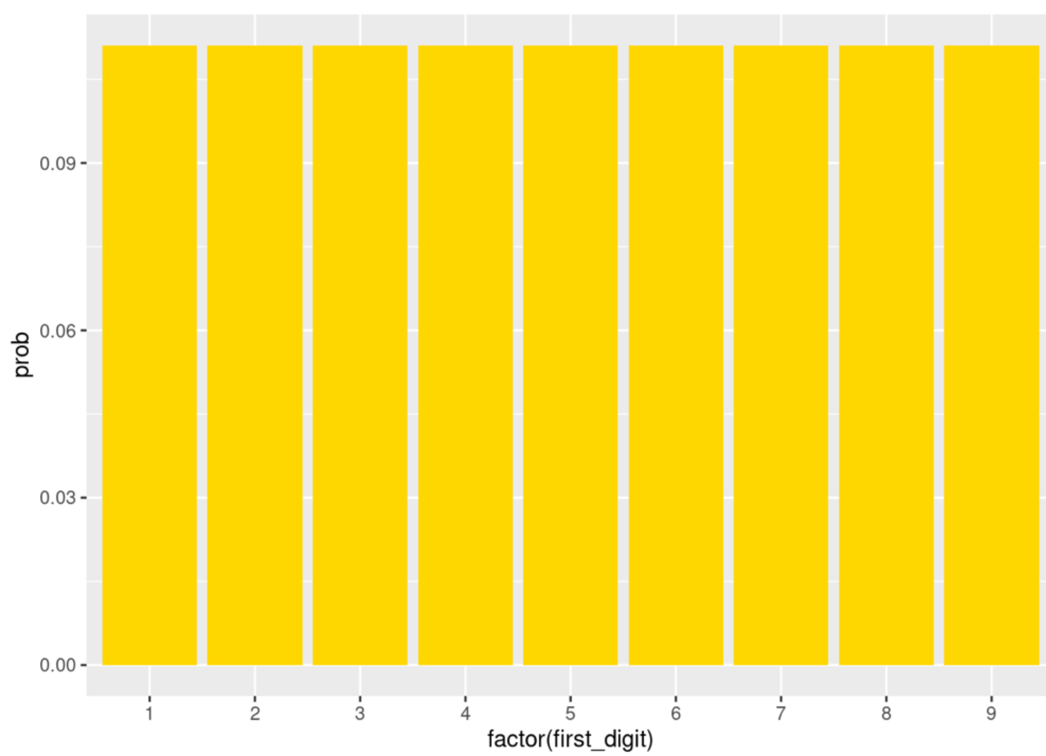
Mutating and Saving the Prob Column:

```
fd_unif <- data.frame(first_digit = seq(1, 9))

fd_unif <- mutate(fd_unif, prob = 1/9)
```

Plot:

```
fd_unif %>%
  ggplot(aes(x = factor(first_digit),
             y = prob)) +
  geom_col(fill = "gold")
```



Question 4:

```
fd_unif <- mutate(fd_unif, "expected_val" = (prob) * (first_dig

fd_unif %>%
  summarize(expected = sum(expected_val))
```

```
  expected
1        5
```

Question 5:

```
fd_unif <- mutate(fd_unif, "x_squ" = (first_digit)*(first_digit

fd_unif %>%
  summarize(variance = sum((x_squ)*(prob)) - 25)
```

```
  variance
1 6.666667
```

Question 6:

```
fd_benford <- data.frame(first_digit = seq(1, 9))

fd_benford <- mutate(fd_benford, prob = log10(1 + (1/first_digi

fd_benford %>%
  summarize(sum_prob = sum(prob) == 1)
```

```
  sum_prob
1     TRUE
```

Question 7:

Expected Value (Benford)

```
fd_benford <- mutate(fd_benford, "expected_val" = (prob) * (fir

expected <- summarize(fd_benford, expected = sum(expected_val))

print(expected)
```

```
  expected
1 3.440237
```

Variance (Benford)

```
fd_benford <- mutate(fd_benford, "x_squ" = (first_digit)*(first_

fd_benford %>%
  summarize(variance = sum((x_squ)*(prob)) - (3.440237)^2)
```

```
  variance
1 6.056512
```

Question 8:

```
fd_benford %>%
  slice_sample(n = 366,
               replace = TRUE,
               weight_by = prob)
```

```
   first_digit       prob expected_val x_squ
1            3 0.12493874    0.3748162     9
2            1 0.30103000    0.3010300     1
3            5 0.07918125    0.3959062    25
4            2 0.17609126    0.3521825     4
5            3 0.12493874    0.3748162     9
6            5 0.07918125    0.3959062    25
7            6 0.06694679    0.4016807    36
8            1 0.30103000    0.3010300     1
9            8 0.05115252    0.4092202    64
10           8 0.05115252    0.4092202    64
11           2 0.17609126    0.3521825     4
12           1 0.30103000    0.3010300     1
13           9 0.04575749    0.4118174    81
14           1 0.30103000    0.3010300     1
```

15	1	0.30103000	0.3010300	1
16	5	0.07918125	0.3959062	25
17	4	0.09691001	0.3876401	16
18	3	0.12493874	0.3748162	9
19	1	0.30103000	0.3010300	1
20	5	0.07918125	0.3959062	25
21	4	0.09691001	0.3876401	16
22	5	0.07918125	0.3959062	25
23	3	0.12493874	0.3748162	9
24	3	0.12493874	0.3748162	9
25	9	0.04575749	0.4118174	81
26	1	0.30103000	0.3010300	1
27	4	0.09691001	0.3876401	16
28	6	0.06694679	0.4016807	36
29	1	0.30103000	0.3010300	1
30	1	0.30103000	0.3010300	1
31	9	0.04575749	0.4118174	81
32	2	0.17609126	0.3521825	4
33	2	0.17609126	0.3521825	4
34	8	0.05115252	0.4092202	64
35	4	0.09691001	0.3876401	16
36	3	0.12493874	0.3748162	9
37	4	0.09691001	0.3876401	16
38	5	0.07918125	0.3959062	25
39	2	0.17609126	0.3521825	4
40	5	0.07918125	0.3959062	25
41	5	0.07918125	0.3959062	25
42	1	0.30103000	0.3010300	1
43	7	0.05799195	0.4059436	49
44	4	0.09691001	0.3876401	16
45	4	0.09691001	0.3876401	16
46	1	0.30103000	0.3010300	1
47	7	0.05799195	0.4059436	49
48	3	0.12493874	0.3748162	9
49	1	0.30103000	0.3010300	1
50	6	0.06694679	0.4016807	36
51	1	0.30103000	0.3010300	1
52	1	0.30103000	0.3010300	1
53	5	0.07918125	0.3959062	25
54	1	0.30103000	0.3010300	1
55	2	0.17609126	0.3521825	4
56	6	0.06694679	0.4016807	36
57	9	0.04575749	0.4118174	81
58	3	0.12493874	0.3748162	9
59	6	0.06694679	0.4016807	36

60	4	0.09691001	0.3876401	16
61	1	0.30103000	0.3010300	1
62	6	0.06694679	0.4016807	36
63	3	0.12493874	0.3748162	9
64	7	0.05799195	0.4059436	49
65	3	0.12493874	0.3748162	9
66	2	0.17609126	0.3521825	4
67	1	0.30103000	0.3010300	1
68	8	0.05115252	0.4092202	64
69	9	0.04575749	0.4118174	81
70	1	0.30103000	0.3010300	1
71	2	0.17609126	0.3521825	4
72	4	0.09691001	0.3876401	16
73	5	0.07918125	0.3959062	25
74	6	0.06694679	0.4016807	36
75	2	0.17609126	0.3521825	4
76	4	0.09691001	0.3876401	16
77	5	0.07918125	0.3959062	25
78	3	0.12493874	0.3748162	9
79	2	0.17609126	0.3521825	4
80	2	0.17609126	0.3521825	4
81	1	0.30103000	0.3010300	1
82	1	0.30103000	0.3010300	1
83	5	0.07918125	0.3959062	25
84	1	0.30103000	0.3010300	1
85	5	0.07918125	0.3959062	25
86	9	0.04575749	0.4118174	81
87	9	0.04575749	0.4118174	81
88	3	0.12493874	0.3748162	9
89	8	0.05115252	0.4092202	64
90	4	0.09691001	0.3876401	16
91	1	0.30103000	0.3010300	1
92	1	0.30103000	0.3010300	1
93	6	0.06694679	0.4016807	36
94	5	0.07918125	0.3959062	25
95	2	0.17609126	0.3521825	4
96	6	0.06694679	0.4016807	36
97	2	0.17609126	0.3521825	4
98	1	0.30103000	0.3010300	1
99	5	0.07918125	0.3959062	25
100	3	0.12493874	0.3748162	9
101	3	0.12493874	0.3748162	9
102	2	0.17609126	0.3521825	4
103	5	0.07918125	0.3959062	25
104	6	0.06694679	0.4016807	36

105	1	0.30103000	0.3010300	1
106	6	0.06694679	0.4016807	36
107	9	0.04575749	0.4118174	81
108	3	0.12493874	0.3748162	9
109	3	0.12493874	0.3748162	9
110	3	0.12493874	0.3748162	9
111	2	0.17609126	0.3521825	4
112	8	0.05115252	0.4092202	64
113	6	0.06694679	0.4016807	36
114	1	0.30103000	0.3010300	1
115	1	0.30103000	0.3010300	1
116	7	0.05799195	0.4059436	49
117	3	0.12493874	0.3748162	9
118	6	0.06694679	0.4016807	36
119	5	0.07918125	0.3959062	25
120	3	0.12493874	0.3748162	9
121	6	0.06694679	0.4016807	36
122	4	0.09691001	0.3876401	16
123	8	0.05115252	0.4092202	64
124	5	0.07918125	0.3959062	25
125	2	0.17609126	0.3521825	4
126	7	0.05799195	0.4059436	49
127	1	0.30103000	0.3010300	1
128	1	0.30103000	0.3010300	1
129	4	0.09691001	0.3876401	16
130	6	0.06694679	0.4016807	36
131	1	0.30103000	0.3010300	1
132	6	0.06694679	0.4016807	36
133	1	0.30103000	0.3010300	1
134	9	0.04575749	0.4118174	81
135	3	0.12493874	0.3748162	9
136	2	0.17609126	0.3521825	4
137	4	0.09691001	0.3876401	16
138	3	0.12493874	0.3748162	9
139	4	0.09691001	0.3876401	16
140	2	0.17609126	0.3521825	4
141	4	0.09691001	0.3876401	16
142	3	0.12493874	0.3748162	9
143	2	0.17609126	0.3521825	4
144	4	0.09691001	0.3876401	16
145	3	0.12493874	0.3748162	9
146	3	0.12493874	0.3748162	9
147	1	0.30103000	0.3010300	1
148	2	0.17609126	0.3521825	4
149	6	0.06694679	0.4016807	36

150	8	0.05115252	0.4092202	64
151	8	0.05115252	0.4092202	64
152	1	0.30103000	0.3010300	1
153	9	0.04575749	0.4118174	81
154	5	0.07918125	0.3959062	25
155	1	0.30103000	0.3010300	1
156	1	0.30103000	0.3010300	1
157	1	0.30103000	0.3010300	1
158	5	0.07918125	0.3959062	25
159	5	0.07918125	0.3959062	25
160	1	0.30103000	0.3010300	1
161	3	0.12493874	0.3748162	9
162	1	0.30103000	0.3010300	1
163	9	0.04575749	0.4118174	81
164	8	0.05115252	0.4092202	64
165	6	0.06694679	0.4016807	36
166	5	0.07918125	0.3959062	25
167	2	0.17609126	0.3521825	4
168	4	0.09691001	0.3876401	16
169	2	0.17609126	0.3521825	4
170	8	0.05115252	0.4092202	64
171	2	0.17609126	0.3521825	4
172	1	0.30103000	0.3010300	1
173	2	0.17609126	0.3521825	4
174	6	0.06694679	0.4016807	36
175	6	0.06694679	0.4016807	36
176	4	0.09691001	0.3876401	16
177	1	0.30103000	0.3010300	1
178	1	0.30103000	0.3010300	1
179	4	0.09691001	0.3876401	16
180	3	0.12493874	0.3748162	9
181	2	0.17609126	0.3521825	4
182	1	0.30103000	0.3010300	1
183	5	0.07918125	0.3959062	25
184	1	0.30103000	0.3010300	1
185	1	0.30103000	0.3010300	1
186	1	0.30103000	0.3010300	1
187	6	0.06694679	0.4016807	36
188	1	0.30103000	0.3010300	1
189	6	0.06694679	0.4016807	36
190	5	0.07918125	0.3959062	25
191	2	0.17609126	0.3521825	4
192	4	0.09691001	0.3876401	16
193	2	0.17609126	0.3521825	4
194	1	0.30103000	0.3010300	1

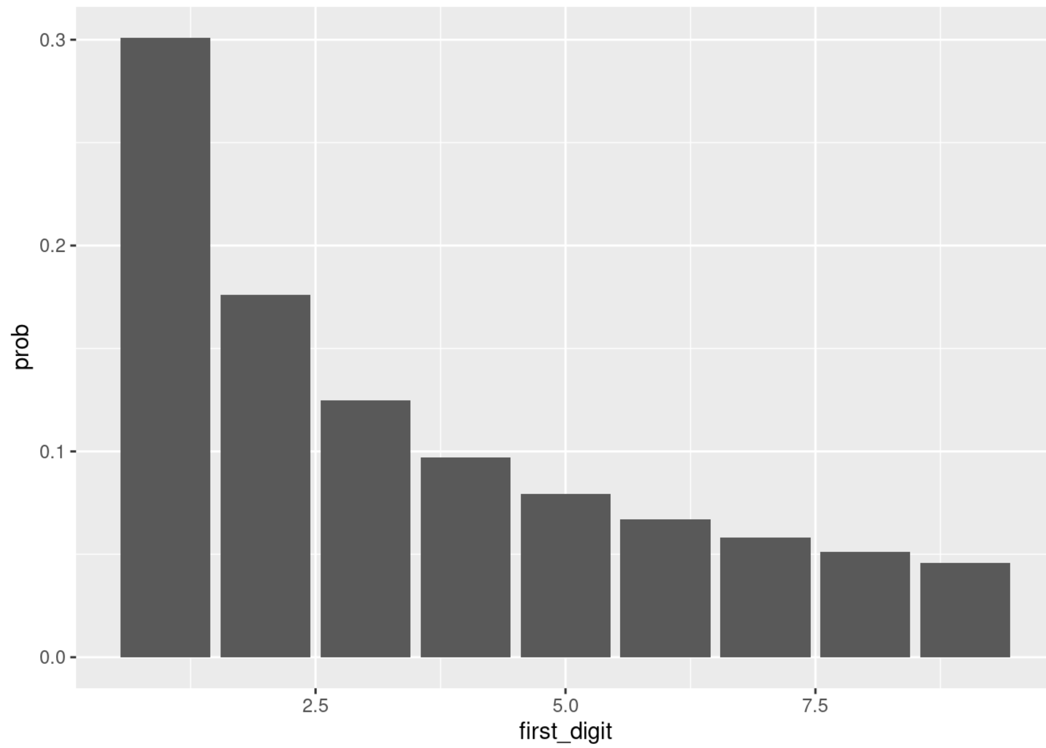
195	2	0.17609126	0.3521825	4
196	6	0.06694679	0.4016807	36
197	1	0.30103000	0.3010300	1
198	5	0.07918125	0.3959062	25
199	3	0.12493874	0.3748162	9
200	3	0.12493874	0.3748162	9
201	9	0.04575749	0.4118174	81
202	2	0.17609126	0.3521825	4
203	6	0.06694679	0.4016807	36
204	9	0.04575749	0.4118174	81
205	5	0.07918125	0.3959062	25
206	3	0.12493874	0.3748162	9
207	3	0.12493874	0.3748162	9
208	1	0.30103000	0.3010300	1
209	1	0.30103000	0.3010300	1
210	2	0.17609126	0.3521825	4
211	8	0.05115252	0.4092202	64
212	3	0.12493874	0.3748162	9
213	1	0.30103000	0.3010300	1
214	2	0.17609126	0.3521825	4
215	2	0.17609126	0.3521825	4
216	9	0.04575749	0.4118174	81
217	4	0.09691001	0.3876401	16
218	9	0.04575749	0.4118174	81
219	2	0.17609126	0.3521825	4
220	6	0.06694679	0.4016807	36
221	5	0.07918125	0.3959062	25
222	5	0.07918125	0.3959062	25
223	3	0.12493874	0.3748162	9
224	1	0.30103000	0.3010300	1
225	1	0.30103000	0.3010300	1
226	1	0.30103000	0.3010300	1
227	3	0.12493874	0.3748162	9
228	3	0.12493874	0.3748162	9
229	1	0.30103000	0.3010300	1
230	1	0.30103000	0.3010300	1
231	6	0.06694679	0.4016807	36
232	8	0.05115252	0.4092202	64
233	1	0.30103000	0.3010300	1
234	9	0.04575749	0.4118174	81
235	8	0.05115252	0.4092202	64
236	4	0.09691001	0.3876401	16
237	3	0.12493874	0.3748162	9
238	1	0.30103000	0.3010300	1
239	1	0.30103000	0.3010300	1

240	2	0.17609126	0.3521825	4
241	8	0.05115252	0.4092202	64
242	5	0.07918125	0.3959062	25
243	1	0.30103000	0.3010300	1
244	1	0.30103000	0.3010300	1
245	9	0.04575749	0.4118174	81
246	1	0.30103000	0.3010300	1
247	7	0.05799195	0.4059436	49
248	1	0.30103000	0.3010300	1
249	2	0.17609126	0.3521825	4
250	2	0.17609126	0.3521825	4
251	3	0.12493874	0.3748162	9
252	2	0.17609126	0.3521825	4
253	1	0.30103000	0.3010300	1
254	1	0.30103000	0.3010300	1
255	3	0.12493874	0.3748162	9
256	4	0.09691001	0.3876401	16
257	1	0.30103000	0.3010300	1
258	1	0.30103000	0.3010300	1
259	6	0.06694679	0.4016807	36
260	2	0.17609126	0.3521825	4
261	1	0.30103000	0.3010300	1
262	4	0.09691001	0.3876401	16
263	5	0.07918125	0.3959062	25
264	3	0.12493874	0.3748162	9
265	2	0.17609126	0.3521825	4
266	7	0.05799195	0.4059436	49
267	2	0.17609126	0.3521825	4
268	2	0.17609126	0.3521825	4
269	4	0.09691001	0.3876401	16
270	1	0.30103000	0.3010300	1
271	5	0.07918125	0.3959062	25
272	5	0.07918125	0.3959062	25
273	1	0.30103000	0.3010300	1
274	3	0.12493874	0.3748162	9
275	4	0.09691001	0.3876401	16
276	2	0.17609126	0.3521825	4
277	5	0.07918125	0.3959062	25
278	7	0.05799195	0.4059436	49
279	7	0.05799195	0.4059436	49
280	2	0.17609126	0.3521825	4
281	1	0.30103000	0.3010300	1
282	2	0.17609126	0.3521825	4
283	1	0.30103000	0.3010300	1
284	3	0.12493874	0.3748162	9

285	1	0.30103000	0.3010300	1
286	4	0.09691001	0.3876401	16
287	1	0.30103000	0.3010300	1
288	1	0.30103000	0.3010300	1
289	1	0.30103000	0.3010300	1
290	7	0.05799195	0.4059436	49
291	8	0.05115252	0.4092202	64
292	7	0.05799195	0.4059436	49
293	1	0.30103000	0.3010300	1
294	1	0.30103000	0.3010300	1
295	3	0.12493874	0.3748162	9
296	4	0.09691001	0.3876401	16
297	1	0.30103000	0.3010300	1
298	3	0.12493874	0.3748162	9
299	1	0.30103000	0.3010300	1
300	6	0.06694679	0.4016807	36
301	3	0.12493874	0.3748162	9
302	1	0.30103000	0.3010300	1
303	3	0.12493874	0.3748162	9
304	1	0.30103000	0.3010300	1
305	1	0.30103000	0.3010300	1
306	1	0.30103000	0.3010300	1
307	5	0.07918125	0.3959062	25
308	1	0.30103000	0.3010300	1
309	1	0.30103000	0.3010300	1
310	7	0.05799195	0.4059436	49
311	2	0.17609126	0.3521825	4
312	3	0.12493874	0.3748162	9
313	4	0.09691001	0.3876401	16
314	3	0.12493874	0.3748162	9
315	4	0.09691001	0.3876401	16
316	1	0.30103000	0.3010300	1
317	2	0.17609126	0.3521825	4
318	1	0.30103000	0.3010300	1
319	2	0.17609126	0.3521825	4
320	1	0.30103000	0.3010300	1
321	1	0.30103000	0.3010300	1
322	1	0.30103000	0.3010300	1
323	4	0.09691001	0.3876401	16
324	3	0.12493874	0.3748162	9
325	9	0.04575749	0.4118174	81
326	2	0.17609126	0.3521825	4
327	1	0.30103000	0.3010300	1
328	3	0.12493874	0.3748162	9
329	6	0.06694679	0.4016807	36

```
330            1 0.30103000    0.3010300    1
331            8 0.05115252    0.4092202   64
332            5 0.07918125    0.3959062   25
333            3 0.12493874    0.3748162    9
334            5 0.07918125    0.3959062   25
335            4 0.09691001    0.3876401   16
336            4 0.09691001    0.3876401   16
337            3 0.12493874    0.3748162    9
338            2 0.17609126    0.3521825    4
339            8 0.05115252    0.4092202   64
340            2 0.17609126    0.3521825    4
341            8 0.05115252    0.4092202   64
342            6 0.06694679    0.4016807   36
343            1 0.30103000    0.3010300    1
344            2 0.17609126    0.3521825    4
345            8 0.05115252    0.4092202   64
346            3 0.12493874    0.3748162    9
347            3 0.12493874    0.3748162    9
348            6 0.06694679    0.4016807   36
349            1 0.30103000    0.3010300    1
350            3 0.12493874    0.3748162    9
351            1 0.30103000    0.3010300    1
352            4 0.09691001    0.3876401   16
353            1 0.30103000    0.3010300    1
354            1 0.30103000    0.3010300    1
355            4 0.09691001    0.3876401   16
356            2 0.17609126    0.3521825    4
357            7 0.05799195    0.4059436   49
358            3 0.12493874    0.3748162    9
359            7 0.05799195    0.4059436   49
360            3 0.12493874    0.3748162    9
361            1 0.30103000    0.3010300    1
362            2 0.17609126    0.3521825    4
363            2 0.17609126    0.3521825    4
364            6 0.06694679    0.4016807   36
365            8 0.05115252    0.4092202   64
366            5 0.07918125    0.3959062   25
```

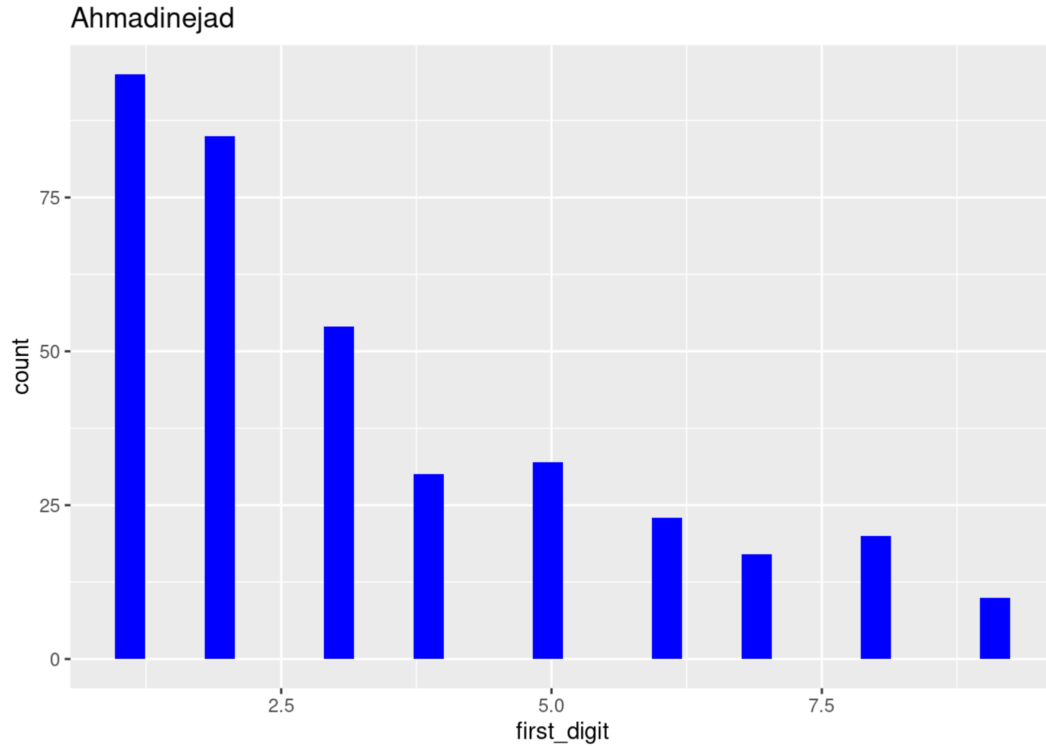
```
fd_benford %>%
  ggplot(aes (x = first_digit, y = prob)) +
  geom_col()
```



Question 9:

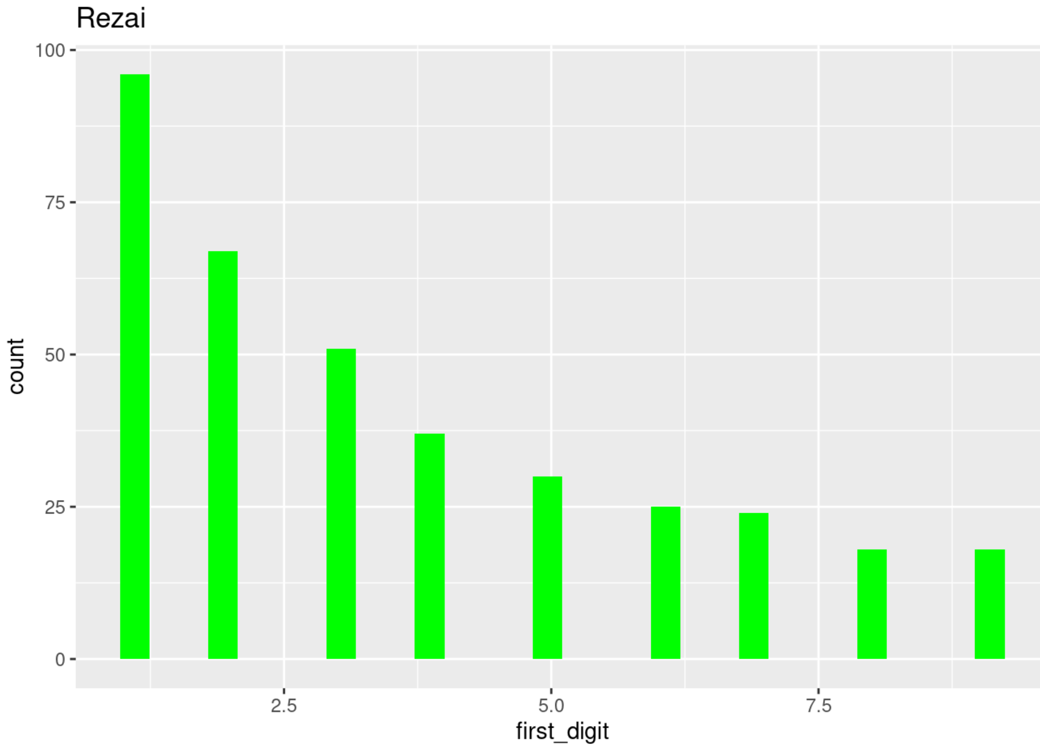
```
iran %>%
  mutate(first_digit = get_first(ahmadinejad)) %>%
  select(ahmadinejad, first_digit) %>%
  ggplot(aes(x = first_digit)) +
  geom_histogram(fill = "blue") +
  ggtitle('Ahmadinejad')
```

```
`stat_bin()` using `bins = 30`. Pick better value with
`binwidth`.
```



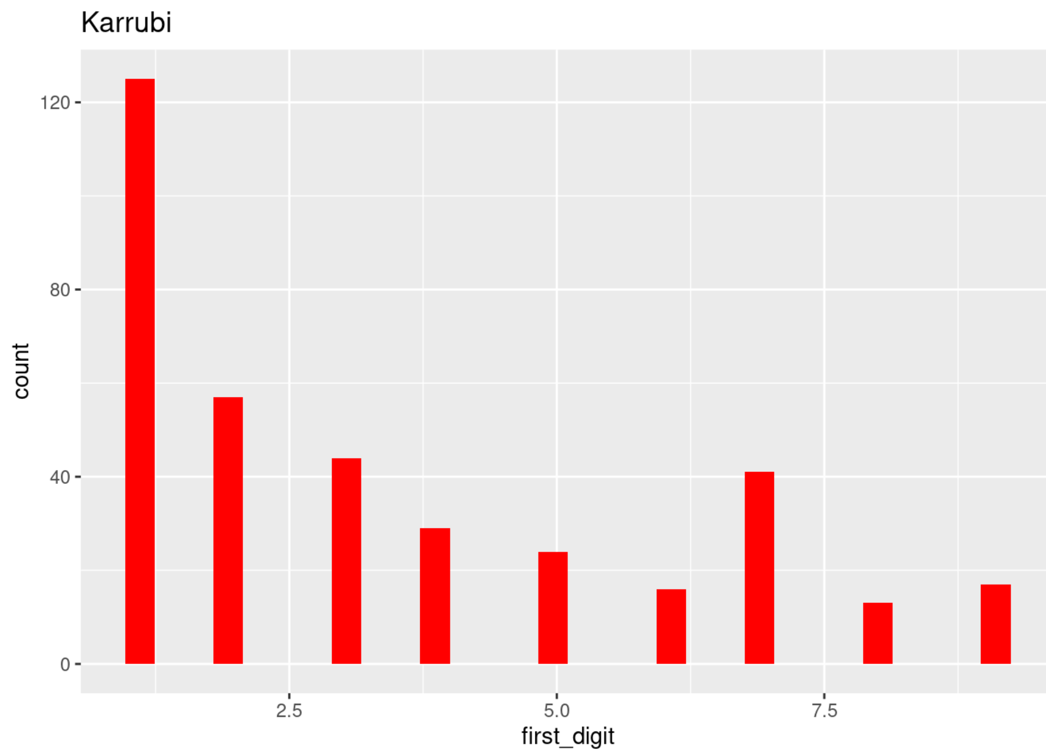

```
iran %>%
  mutate(first_digit = get_first(rezai)) %>%
  select(rezai, first_digit) %>%
  ggplot(aes(x = first_digit)) +
  geom_histogram(fill = "green") +
  ggtitle('Rezai')
```

```
`stat_bin()` using `bins = 30`. Pick better value with
`binwidth`.
```



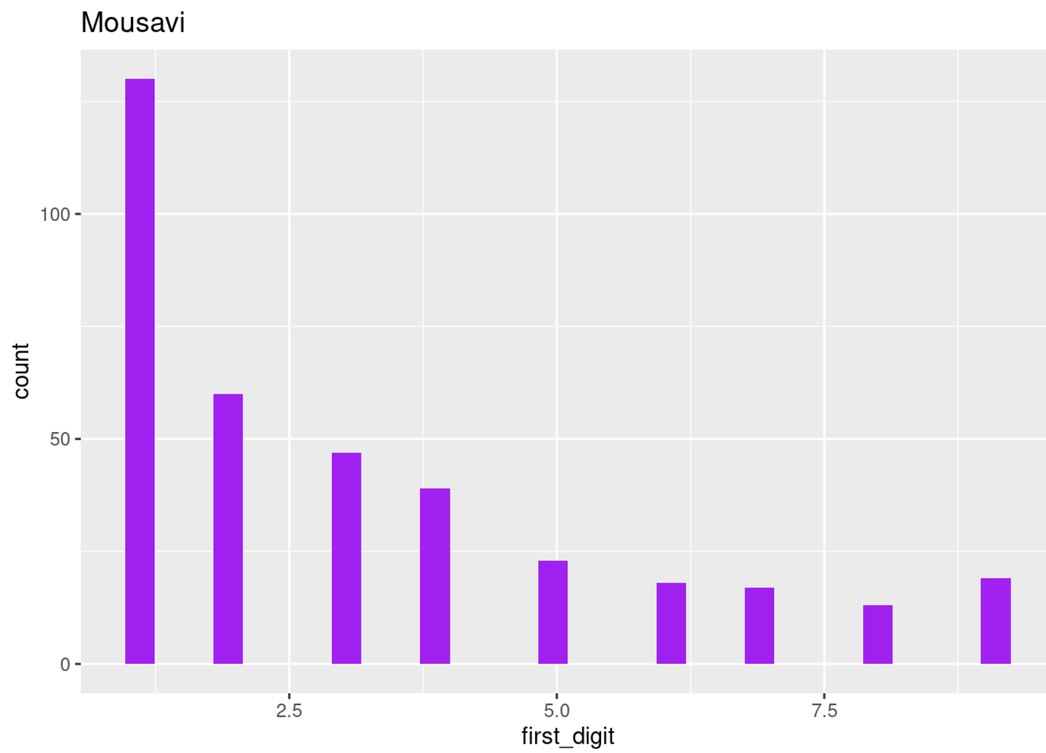

```
iran %>%
  mutate(first_digit = get_first(karrubi)) %>%
  select(karrubi, first_digit) %>%
  ggplot(aes(x = first_digit)) +
  geom_histogram(fill = "red") +
  ggtitle('Karrubi')
```

```
`stat_bin()` using `bins = 30`. Pick better value with
`binwidth`.
```




```
iran %>%
  mutate(first_digit = get_first(mousavi)) %>%
  select(mousavi, first_digit) %>%
  ggplot(aes(x = first_digit)) +
  geom_histogram(fill = "purple") +
  ggtitle('Mousavi')
```

```
`stat_bin()` using `bins = 30`. Pick better value with
`binwidth`.
```



plot1 + plot2 + plot3 + plot4

Question 10:

While it initially seems like Ahmadinejad has the most differences from the Benford's law, it can be seen that the largest difference actually occurs with the Karrubi plot because of the steep drop between the first two bars.

U.S. Elections

Question 11:

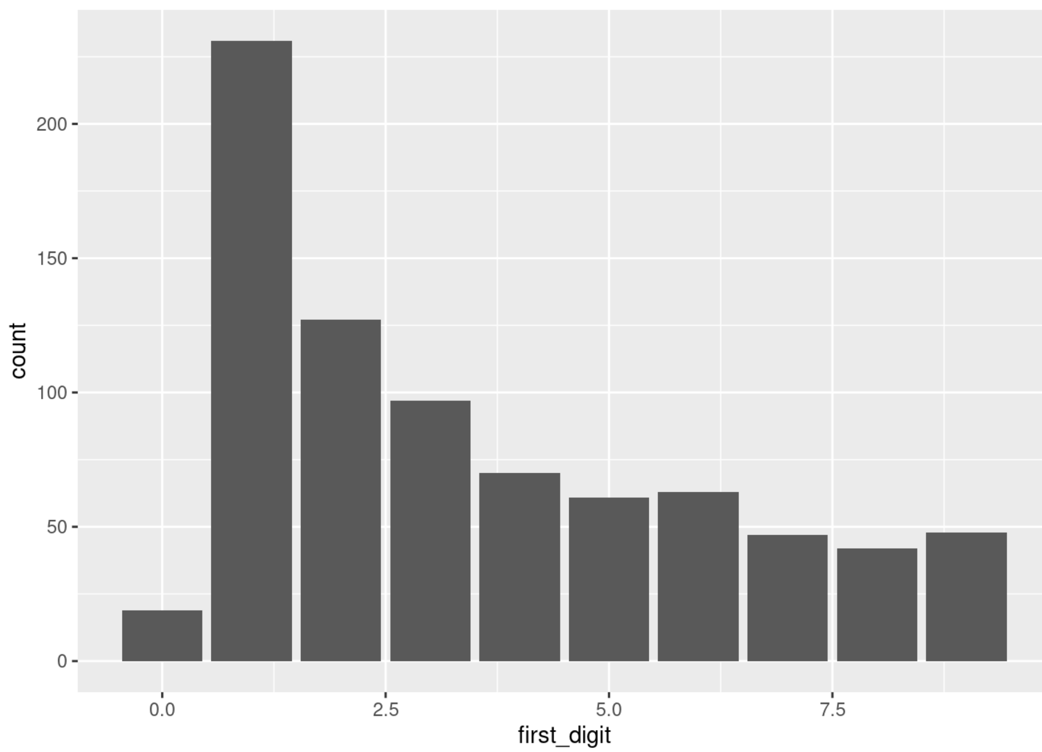
The state I chose to study was New York. The unit of observation in New York's data frame appears to be counties as that is what each row entry is differentiated by. The dimensions of this data frame are 807 x 6.

Question 12:

```
New_York_Data_set %>%
  mutate(first_digit = get_first(votes)) %>%
  select(votes, first_digit) %>%
  ggplot(aes(x = first_digit, fill = votes)) +
    geom_bar()
```

```
Warning: Removed 1 rows containing non-finite values
(`stat_count()`).

Warning: The following aesthetics were dropped during
statistical transformation: fill
ℹ This can happen when ggplot fails to infer the correct
grouping structure in
  the data.
ℹ Did you forget to specify a `group` aesthetic or to convert a
numerical
  variable into a factor?
```



Question 13:

This data seems to fit the Benford model better than the Iran data set. This is extremely important because had there been significant variability, then there the U.S. elections could have been tampered with.