The Distribution of First Digits

In this lab, you will explore the distribution of first digits in real data. For example, the first digits of the numbers 52, 30.8, and 0.07 are 5, 3, and 7 respectively. In this lab, you will investigate the question: how frequently does each digit 1-9 appear as the first digit of the number?

Question 0

Make a prediction.

1. Approximately what percentage of the values do you think will have a *first* digit of 1? What percentage of the values do you think will have a first digit of 9?
2. Approximately what percentage of the values do you think will have a *last* digit of 1? What percentage of the values do you think will have a last digit of 9?

(Don't worry about being wrong. You will earn full credit for any justified answer.)

Percentage of values having the first digit of 1 is 10%. Percentage of values having the first digit of 9 is 10%. Percentage of values having the last digit of 1 is 10%. Percentage of values having the last digit of 9 is 10%.

I predict this because the odds are the first and/or last digit is about 1 out of 9, which is about 10%.

Question 1

The S&P 500 is a stock index based on the market capitalizations of large companies that are publicly traded on the NYSE or NASDAQ. The CSV file `sp500.csv` contains data from February 1, 2018 about the stocks that comprise the S&P 500. We will investigate the first digit distributions of the variables in this data set.

Read in the S&P 500 data. What is the unit of observation in this data set? Is there a variable that is natural to use as the index? If so, set that variable to be the index. Once you are done, display the `DataFrame`.

In [1]:
```
# ENTER YOUR CODE HERE.
import pandas as pd
df = pd.read_csv("sp500.csv")
df.head(10)
```

Out[1]:

	date	Name	open	close	volume
0	2018-02-01	AAL	$54.00	$53.88	3623078
1	2018-02-01	AAPL	$167.16	$167.78	47230787
2	2018-02-01	AAP	$116.24	$117.29	760629
3	2018-02-01	ABBV	$112.24	$116.34	9943452
4	2018-02-01	ABC	$97.74	$99.29	2786798
5	2018-02-01	ABT	$61.75	$62.18	8101584
6	2018-02-01	ACN	$160.16	$160.46	1692576
7	2018-02-01	ADBE	$199.12	$199.38	2366120
8	2018-02-01	ADI	$91.25	$91.65	2312175
9	2018-02-01	ADM	$42.77	$42.46	2921389

The unit of observation in this data set are date, stock names, stock open and close prices, and number of shares. The natural variable to index is volume.

Question 2

We will start by looking at the `volume` column. This variable tells us how many shares were traded on that date.

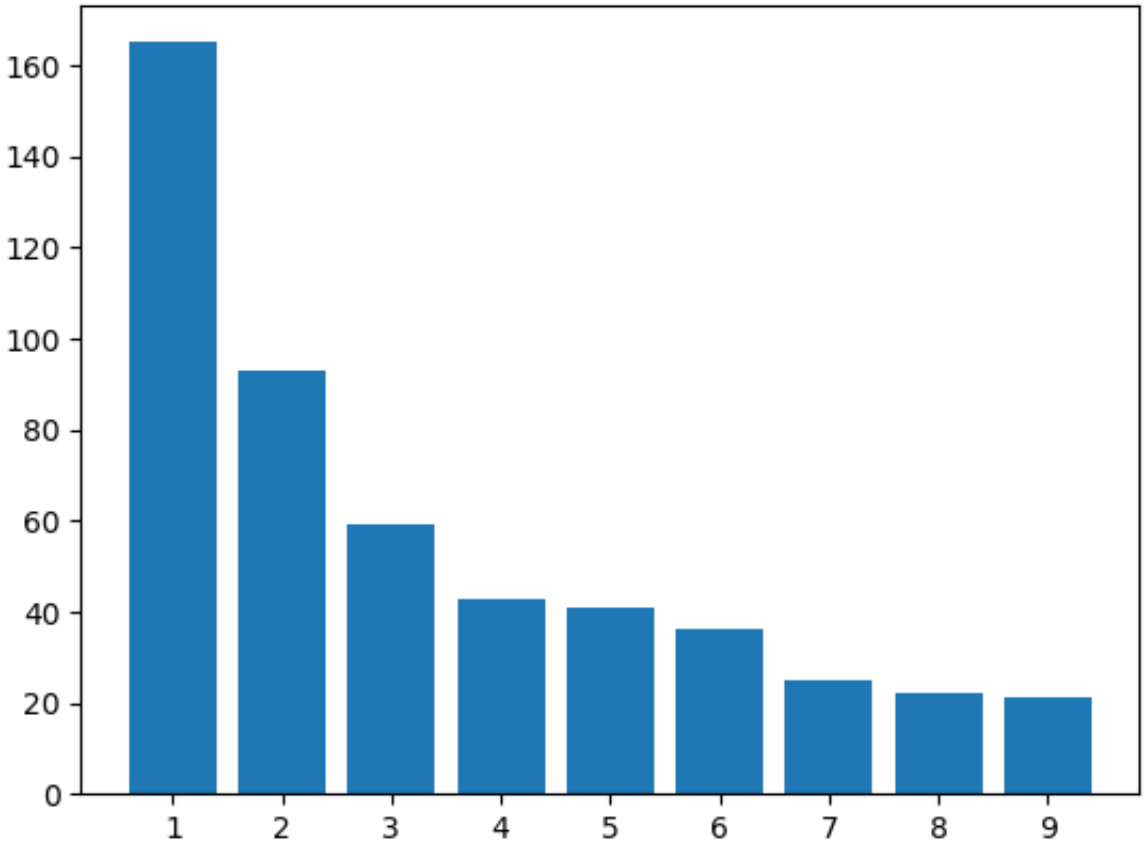
Extract the first digit of every value in this column. (*Hint:* First, turn the numbers into strings. Then, use the text processing functionalities of `pandas` to extract the first character of each string.) Make an appropriate visualization to display the distribution of the first digits. (*Hint:* Think carefully about whether the variable you are plotting is quantitative or categorical.)

How does this compare with what you predicted in Question 0?

My prediction was wrong for both the probability of the first digit being 1 and the first digit being 9. There are significantly more data starting with the digit 1. There significantly less data starting with the digit 9.

In [2]:
```
# ENTER YOUR CODE HERE.

import matplotlib.pyplot as plt
%matplotlib inline

import pandas as pd
df = pd.read_csv("sp500.csv")

# temp = df.volume.astype(str);
# temp.str[0]

df.volume = df.volume.apply(str)
digit_volume = df.volume.str[0].value_counts()
print(digit_volume)

plt.bar(digit_volume.index,digit_volume)
plt.show()
#plt.hist(df.volume.str[0].apply(int))
```

```
1    165
2     93
3     59
4     43
5     41
6     36
7     25
8     22
9     21
Name: volume, dtype: int64
```

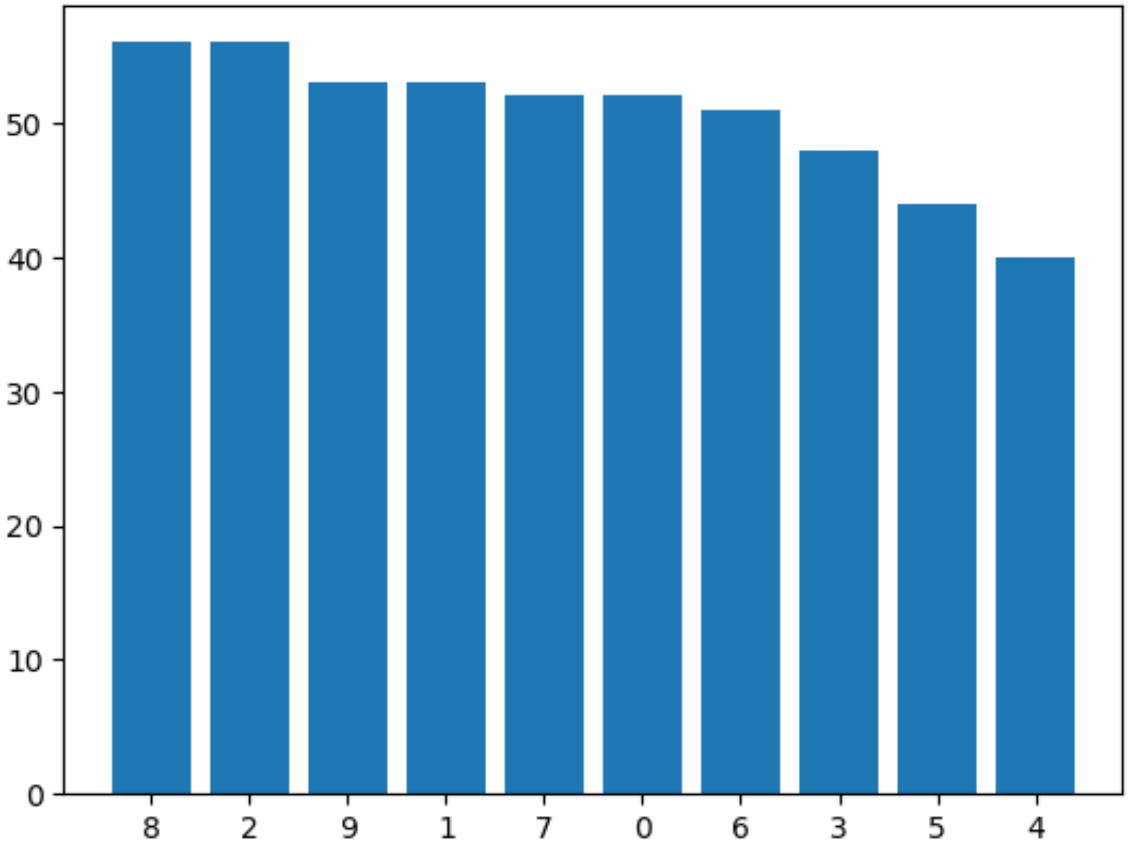


Question 3

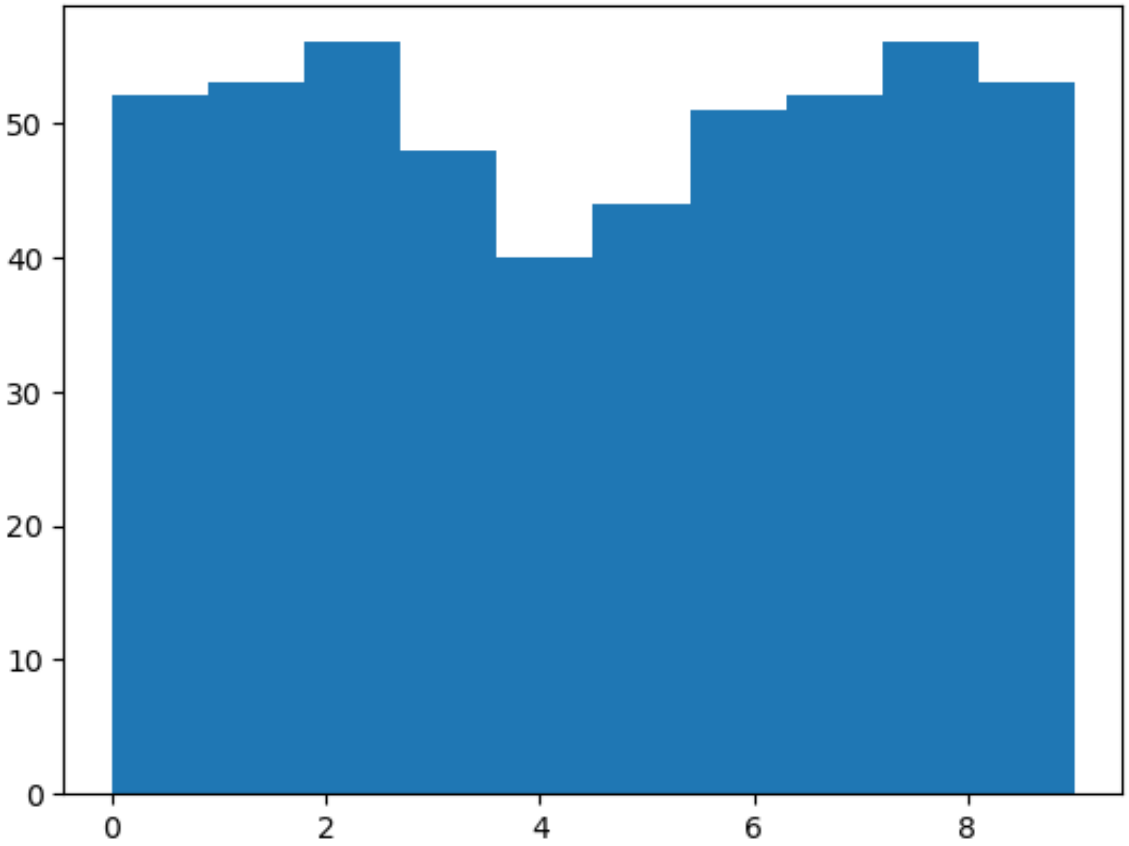
Now, repeat Question 2, but for the distribution of *last* digits. Again, make an appropriate visualization and compare with your prediction in Question 0.

In [3]:
```
# ENTER YOUR CODE HERE.
import matplotlib.pyplot as plt
%matplotlib inline

import pandas as pd
df = pd.read_csv("sp500.csv")

# temp = df.volume.astype(str);
# temp.str[0]

df.volume = df.volume.apply(str)
digit_volume = df.volume.str.strip().str[-1].value_counts()
print(digit_volume)

plt.bar(digit_volume.index,digit_volume)
plt.show()
plt.hist(df.volume.str.strip().str[-1].apply(int))

#df['LastDigit'] = df['UserId'].str.strip().str[-1]
```

```
8    56
2    56
9    53
1    53
7    52
0    52
6    51
3    48
5    44
4    40
Name: volume, dtype: int64
```



Out[3]:
```
(array([52., 53., 56., 48., 40., 44., 51., 52., 56., 53.]),
 array([0. , 0.9, 1.8, 2.7, 3.6, 4.5, 5.4, 6.3, 7.2, 8.1, 9. ]),
 <BarContainer object of 10 artists>)
```



My previous prediction is wrong. The probabilities of the last digit either being 1 or 9 are equivalent.

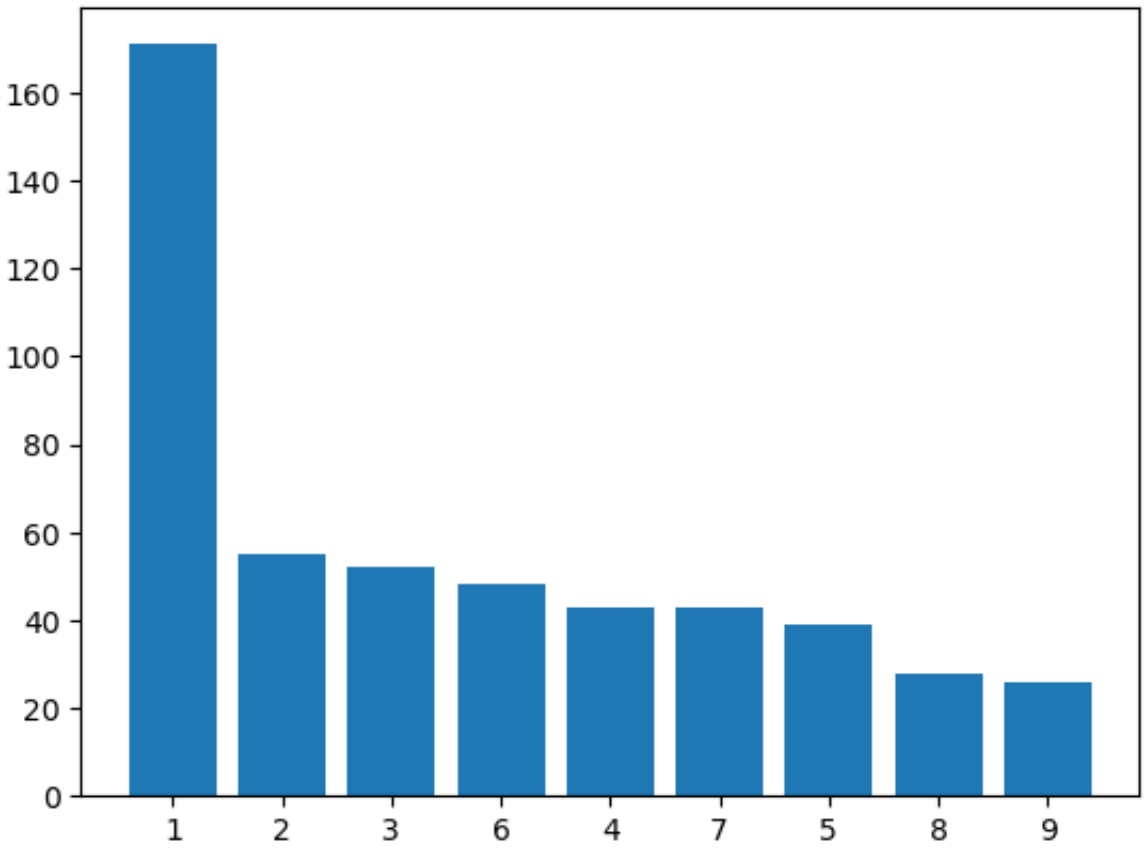
Question 4

Maybe the `volume` column was just a fluke. Let's see if the first digit distribution holds up when we look at a very different variable: the closing price of the stock. Make a visualization of the first digit distribution of the closing price (the `close` column of the `DataFrame`). Comment on what you see.

(*Hint:* What type did `pandas` infer this variable as and why? You will have to first clean the values using the text processing functionalities of `pandas` and then convert this variable to a quantitative variable.)

In [4]:
```
import matplotlib.pyplot as plt
%matplotlib inline

import pandas as pd
df = pd.read_csv("sp500.csv")

#temp = df.close.astype(str);
#temp.str[1]

df.close = df.close.apply(str)
digit_close = df.close.str[1].value_counts()
print(digit_close)

plt.bar(digit_close.index,digit_close)
plt.show()
#plt.hist(df.volume.str[0].apply(int))
```

```
1    171
2     55
3     52
6     48
4     43
7     43
5     39
8     28
9     26
Name: close, dtype: int64
```



ENTER YOUR WRITTEN EXPLANATION HERE.

There was no difference between the probabilities of the first digit either being 1 or 9 between volume and close. Oddly enough the results are almost the same.

Submission Instructions

Once you are finished, follow these steps:

1. Restart the kernel and re-run this notebook from beginning to end by going to `Kernel > Restart Kernel and Run All Cells`.
2. If this process stops halfway through, that means there was an error. Correct the error and repeat Step 1 until the notebook runs from beginning to end.
3. Double check that there is a number next to each code cell and that these numbers are in order.

Then, submit your lab as follows:

1. This quarter, you don't need to demo Lab 1. The first lab to demo will be Lab 2.
2. Upload your .ipyn Notebook to Canvas and pdf to Gradescope.