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.)

ENTER YOUR WRITTEN EXPLANATION HERE. 1) Assuming 1-9 has an equal chance to appear in the 1st position, the answer of the two questions is 1/9 because 0 cannot be the first digit. 2) Assuming every digit has an equal chance to appear in the last position, the answer of the two question is 1/10.

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()
df.set_index('Name', inplace=True)
df
```

Out[1]:

Name	date	open	close	volume
AAL	2018-02-01	$54.00	$53.88	3623078
AAPL	2018-02-01	$167.16	$167.78	47230787
AAP	2018-02-01	$116.24	$117.29	760629
ABBV	2018-02-01	$112.24	$116.34	9943452
ABC	2018-02-01	$97.74	$99.29	2786798
...	...	...	...	...
XYL	2018-02-01	$72.50	$74.84	1817612
YUM	2018-02-01	$84.24	$83.98	1685275
ZBH	2018-02-01	$126.35	$128.19	1756300
ZION	2018-02-01	$53.79	$54.98	3542047
ZTS	2018-02-01	$76.84	$77.82	2982259

505 rows × 4 columns

ENTER YOUR WRITTEN EXPLANATION HERE. Dollars are the unit of observation. Name is natural to use as the index.

Question 2

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

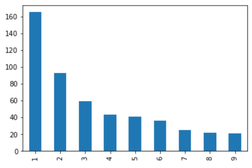
Extract the first digit of every value in this column. (*Hint:* First, turn the numbers into strings. Then, use the text processing functionalities of `pandas` to extract the first character of each string.) Make an appropriate visualization to display the distribution of the first digits. (*Hint:* Think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

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

In [2]:
```
# ENTER YOUR CODE HERE.
import pandas as pd
df = pd.read_csv("sp500.csv")
df.volume = df.volume.apply(str)
first_digits = df.volume.str[0]
unique_nums = first_digits.value_counts()
print(unique_nums)
import matplotlib.pyplot as plt
%matplotlib inline
unique_nums.plot.bar()
```

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

Out[2]:
```
<AxesSubplot:>
```



ENTER YOUR WRITTEN EXPLANATION HERE. Clearly, the data is skewed heavily to the right, meaning smaller numbers appear at a far higher frequency than larger numbers for the 1st digit in the volume column. This result certainly deviates from my prediction in Q0 that all first digits would appear around 1/9 of the time.

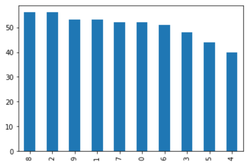
Question 3

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 pandas as pd
df = pd.read_csv("sp500.csv")
df.volume = df.volume.apply(str)
last_digits = df.volume.str[-1]
unique_nums = last_digits.value_counts()
print(unique_nums)
import matplotlib.pyplot as plt
%matplotlib inline
unique_nums.plot.bar()
```

```
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]:
```
<AxesSubplot:>
```



ENTER YOUR WRITTEN EXPLANATION HERE. This result more closely alligns to my prediction in Q0 than the result from Q2. My prediction that each digit would appear ~1/10 of the time is supported by the data.

Question 4

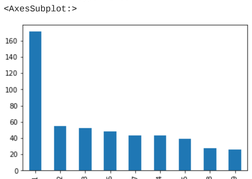
Maybe the `volume` column was just a fluke. Let's see if the first digit distribution holds up when we look at a very different variable: the closing price of the stock. Make a visualization of the first digit distribution of the closing price (the `close` column of the `DataFrame`). Comment on what 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]:
```
# ENTER YOUR CODE HERE.
import pandas as pd
df = pd.read_csv("sp500.csv")
df.close = df.close.apply(str)
first_digits = df.close.str[1]
unique_nums = first_digits.value_counts()
print(unique_nums)
import matplotlib.pyplot as plt
%matplotlib inline
unique_nums.plot.bar()
```

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

Out[4]:
```
<AxesSubplot:>
```



ENTER YOUR WRITTEN EXPLANATION HERE. This result closely lines up with the data distribution found in Q2. This is very interesting as Q2 focused on quanity of stocks being sold and bought on 2/18/18, while Q4 is focused on dollar amounts. There most likely is no correlation between numbers related to stocks and smaller first digit frequency, but this is certainly an interesting coincidence.

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. Demo your lab to obtain credit.
2. Upload your .ipyn Notebook to iLearn and pdf to Gradescope.