Brian Gould, Jack Dunne, William Shea, Jaewon Yu
ECON 380
Final Project

Social Media vs In Person Interactions: How Different Types of Interactions Affect Election Results and Opinion Distributions

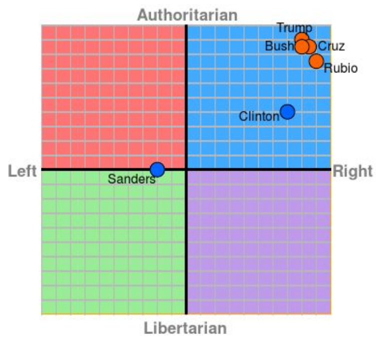
Abstract

Advancements in the facilitation of ideas have shaped human history for millennia. These inventions of immense influence and importance include the likes of language conception, written history, the printing press, and in this modern era, social media. Social media has taken the democratizing effects of the internet to its extremes by allowing freedoms in communication that has never been seen before in the history of mankind. However, in a time in which the world is at it most connected through technology, there seems to be a growing divide between the people of the world. A wave of partisan rhetoric has swept the world, creating an ideological divide that has taken the forefront of everyday life.

Through our study, we examine and model how human interactions can cause such polarization in opinions and how this divide can be remedied. We examine factors that influence human opinion and apply the modern phenomenon of social media to this framework. Our results demonstrated the effects that factors such as influence from media, education, and willingness to listen have on the distribution of opinions. Those with normal distribution of opinions became more polarized after being exposed to polarizing news. In contrast, populations that were already polarized became more centered around the median once they were exposed to centrist views on the news. Furthermore, both of these distributions of opinions shifted to one side of the spectrum after interacting with each other. More educated populations became increasingly centrist and moderate after exposure to low credibility news sources and interactions with one another compared to less educated populations. Populations that had higher willingness to listen became moderate after exposure to moderate news sources and interaction with one another. In contrast, populations with low willingness to listen stayed polarized even after they had received the same treatment as the population with a higher willingness to listen. Our findings show that human opinion and beliefs are subject to change and evolution. As such, it is both susceptible to falsehoods as well as truths. The subject of human interactions is an important topic.

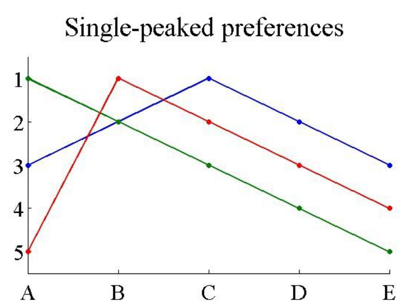
Introduction

In the aftermath of the 2016 election, it would appear that the median voter theorem had failed. A candidate that by most measures, was very extreme managed to win the electoral college against a more median candidate.



However, this may not be an outright failure of the median voter theorem. Instead, we assert that the distribution of voters has been changed by the presence of misinformation throughout social media and traditional media. For context, the median voter theorem states that whichever candidate is closest to the median voter will win the election in a "first past the post" voting system.

The median voter theorem makes three key assumptions that allow it to work. First, that the median voter is only calculated by taking the median of only those who vote. Second, the voters are able to place their options on a one dimensional spectrum that can encapsulate all the possible stances of a candidate. Finally, it assumes that everyone has single peaked preferences, that is to say that they have one preferred outcome and as you get further away from that on the spectrum, the preference of it always goes down.

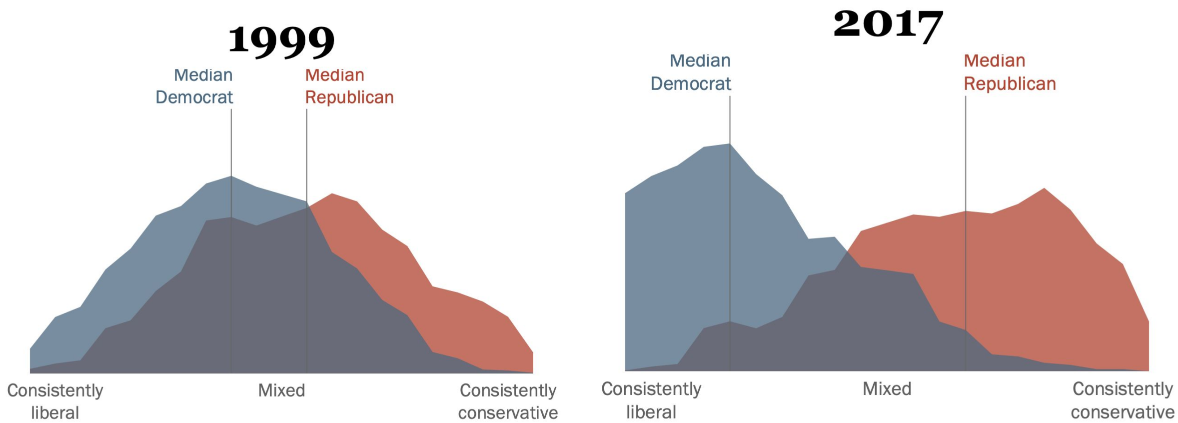


Our experiment will examine how certain factors including education, willingness to listen, sociability and others affect the shift in political opinions. We will use this distribution to model elections both before and after the influence of misinformation and examine the change in outcomes.

An important consideration to make in support of the median voter theorem is the existence of primaries in American elections. This will produce candidates that are closest to the median voter within their respective party, but unrepresentative of the voters as a whole.

Another consideration to make is that "first past the post" creates a two party system where third parties only act to hurt the candidate that they are closest to, known as the spoiler effect. This means that a range of candidates that can represent the entire voting population is not available and voters are constrained to the two choices that the parties present to them. Consider an extreme example where the two candidates selected in the primaries are at complete opposite ends of the spectrum, if the median voter is at the center, the candidate closest to them will still be extreme.

The question emerging from this is why? The 2016 election and the years following have exposed extreme political polarization within the United States according to Pew Research ("The Shift in the American Public's Political Values."). There are less and less people that agree with the other side on any issues which makes compromise and long lasting legislation effectively impossible. If we can understand the origin of this polarization, we can take steps to mitigate it.



Anecdotal evidence such as the murder of Heather Heyer in Charlottseville, Virgina, along with the injuring of 19 others in a car attack at a counterprotest of a far right rally, has served to show the violence that has become part of these extreme political idealogies. We could go so far as to say that this paper is less about the median voter theory and more so an issue of public safety. Social media undoubtedly played a massive role in the 2016 presidential election, we want to see what factors play the biggest role in the subsequent split. Using our model we can determine these factors and their importance. With that information in hand, it is possible to suggest social programs or initiatives for social media platforms that could decrease polarization.
In the following paper, we will discuss the literature we used to determine our parameters, state our thesis, lay out our goals in more detail, detail our methodology, and break down our code and what it means. Finally, we will show the outcomes of the simulations with different parameters and analyze what the results mean.

Literature

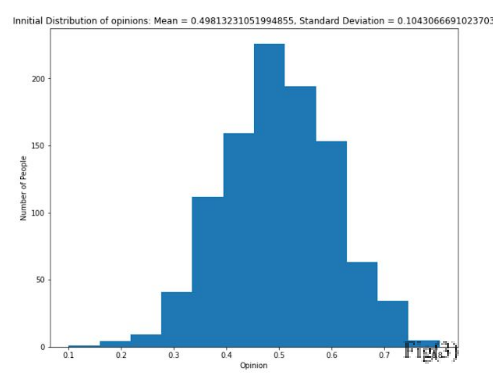
In recent history, America's political landscape has become increasingly polarized. Social media has contributed substantially towards this political divergence (pew research). Presidential debates from the previous two election cycles have included more misinformation than any debate prior. Social media has become an echochamber for many Americans, showing us only what we want to hear and who we want to hear from. Heltzel et al. claims that present American politics is either in a cycle of self-reinforcing partisanship, or that the pendulum of American politics is swinging. Now more than ever, a query into the nature of human politics is necessary. Our goal is to further understand the way social media and political interaction shape the political divide.

Thesis

Social media has become a mainstay in modern society in terms of how people communicate, connect, and spread information. In the time that it took for social media to take root in society, its users have found new and creative ways to employ this powerful tool. However, with greater knowledge came an even greater understanding that social media had become a beast that had taken a life of its own beyond the scope of its original intent. Far from the trivial entertainment tool used to express oneself and connect with others, social media has become a means to spread ideas across millions on the planet. This has been used for good but it has also been used for more sinister purposes. The 2016 U.S. presidential election saw immense efforts by foreign entities to engage in disinformation campaigns on social media in order to influence the results of the election. Social media was used extensively in an effort to influence people. The recent 2020 U.S. presidential election also saw similar efforts. More than ever, we have become aware of the polarizing effects that social media can have on people's opinions. Social media has created a wealth of information but has exposed people to an unprecedented amount of misinformation. This will create a more extreme political climate. We aim to create a theoretical framework through agent based modeling that can model the distribution of opinions in a society due to small scale changes on the way people can interact. We aim to create a preliminary model that can theoretically describe the partisan effects of social media.

Methods

To simulate how a society will change it's opinions, we have chosen to use agent based modeling. This will allow us to observe how small changes in the way people interact individually will change the overall distribution of opinions of the entire society. First, we create a list of voters, each with specific attributes outlined in the documentation in the Voter class. We call this list of voters a "population". Most importantly, each voter has an opinion attribute on a range of 0 to 1 that indicates a voter's opinion on an issue. We will observe how the distribution of opinions of the entire population changes over the course of the simulation. We then create a "news space" which contains various forms of news that people can interact with. In the simulation, we first expose everyone to all the news articles in the news space through the news_interaction function outlined in the documentation. Then, we allow voters to have one-on-one interactions with each other. After a certain number of interactions, the simulation terminates.



Fig(3)

We use histograms to represent how the opinions of a society are distributed. For example, Fig (3) displays an approximately normal distribution of opinions, indicating the society generally agrees on the issue. For a given experiment, 3 things will be returned and observed. The 1st item is the initial population distribution of opinions. This is controlled based on the input given by the user/constructor of the experiment.

Distributions can be normal, n-modal, binary, or uniform. We will also observe the distribution of news sources and their political leanings. This will give us an idea of the landscape that we start out with. For example, a society could start out with a normal distribution of political opinions, and a bi-modal distribution of news sources. We then run the interaction function, resulting in the population being exposed to news sources, which influences their opinions. They then interact with one another, and depending on their attributes, influence one another's opinions, model a real life political discussion/interaction. Finally, we observe the final political distribution of the society after the model has been run. We can directly compare the society before and after the interaction to observe how the particular factors influenced political opinions.

Interaction Method

For agent-based modeling, it is important to understand fundamentally what occurs at the interaction level so that you can explain why small changes in interactions lead to changes in the overall distribution of opinions in the society. Here, we will go into more detail about how the interaction works mathematically.

For each interaction, we specify a voter and an influencer. The influencer changes the opinion of the voter. For a news interaction, the piece of news is the influencer and the voter is the voter. For a one-on-one voter interaction between 2 voters, say voter1 and voter2, 2 interactions actually occur. One interaction specifies voter1 as the influencer and voter2 as the voter. The other interaction switches the roles.

The interaction begins by checking each voter's "willingness to listen" parameter. If the difference between the influence's opinion and the voter's opinion is greater than the "willingness to listen" parameter, the voter will not change their opinion at all[1]. This simulates a phenomenon of confirmation bias where people will only listen to people or sources that support their opinion.

Each interaction begins with a parameter x. Mathematically,

$$x_0 = rn(1-f)(1-max(0,\ e_v - e_i))$$

Where r is a random number, n is a number specifying how influential the influencer is, f indicates how formed the voter's opinion is, e_v is the education of the voter, e_i is the education of the influencer[2]. All variables in this equation lie in the interval [0,1], so x_0 will also be in the interval [0,1]. Values close to 0 indicate that the influencer is not effective at convincing the voter. Values close to 1 indicate that the voter was influenced by the influencer.

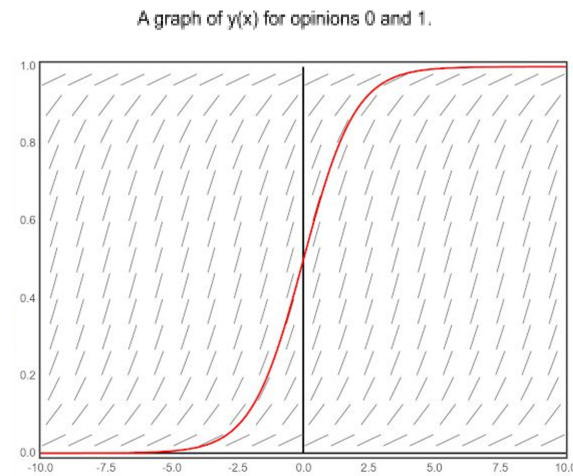
We then apply a linear transformation to x_0 to take it from the interval [0,1] to the interval [-10, 10] by the equation

[1] This feature can be "turned off".

[2] All of these variables can be "turned off" in the model. For instance, if you do not want to consider differences in education, our code has a feature that will set all education levels to 0, effectively removing any effect it has on the model.

$$x = 20x_0 - 10$$

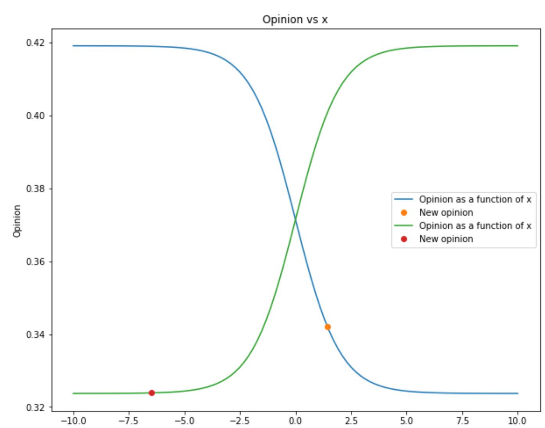
We then plug x into our influence profile curve *y(x)*, which is defined by the differential equation



$$\frac{dy}{dx} = sign\{O_i - O_v\}(y - O_v)(y - O_i);\ y(0) = 0$$

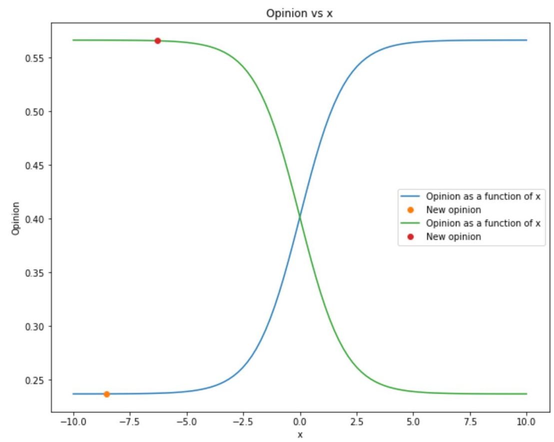
Where y is the new opinion of the voter, O_v is the old opinion of the voter, and O_f is the opinion of the influencer. Notice that $\frac{dy}{dx}$ has 0's at O_v and O_i, enforcing that the voter's new opinion lies between O_v and O_i. Large positive values of x will cause the voter's opinion to change a lot, while large negative values will cause the voter's opinion to change only a little bit. We chose this shape of a curve because studies have shown that people are less likely to have minor shifts in opinion during a discussion; it is most likely that they will retain their own position or switch heavily to the other side[3].

3 examples of voter one-on-one interactions are displayed graphically below:

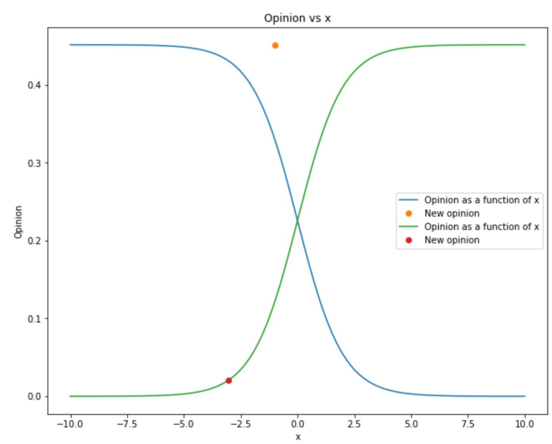


y(x) of each voter is represented by the blue and green lines. The new opinions of the voters are represented by the orange and red dots respectively. In this interaction, the red voter was not influenced by the orange voter, but the orange voter was influenced by the red voter to change his opinion from .42 to .34.

[3] At the moment, this curve is the only y(x) that is offered for modeling interactions. In future updates we hope to add more possible y(x) to model interactions and allow the user to define custom y(x) to use as influence profile curves.



In this figure, neither voter changed opinions.



In this figure, the red voter's opinion was outside the orange voter's "willingness to listen" radius. Even though the orange voter had an x value that could influence their opinion, orange did not change their opinion.

Code Documentation

One of our goals in this project is to create a theoretical framework for modeling how societies interact. We used object oriented programming to create a dynamic model with many parameters that can be adjusted with almost infinite possibilities in order to allow many different types of societies with vastly different social conventions to be modeled.

Although we do not explicitly focus on how parameters in our model can be measured from real data from societies, such work should be done in the future. For this reason, we will include the documentation of all our code here so that future researchers can easily use it to model specific societal behaviors for different types of interactions.

Our code largely depends on 2 main classes, the Voter class and the News class. These classes can “interact” with each other, and the Voter class can change opinions. We then use functions such as society_generator and experiment, as well as the NewsSpace class, to allow easy creation of any society of voters and news with different characteristics. The following documentation is designed to allow others to easily reproduce our models and create their own models using our code.

Documentation:

```
Voter class
Init signature: Voter(opinion, educ, wtl, formed, social, influence, formingRate=0.05)
Docstring:
The voter class represents 1 individual person. This person has attributes and an opinion that can be
changed after interactions with News classes and other Voter classes. The voters and news that the
person interacts with will be referred to as an "influence"
Init docstring:
Parameters:
----------------------------
opinion: float between 0 and 1; represents the voter's opinion on 1 specific topic
educ: float between 0 and 1; education level on the topic, from 0 to 1. (.4 = average highschool grad,
    .6 = average college grad,.8 and above: expert in the field)
wtl: float; represents a person's "willingness to listen radius" if an influence has an opinion that is
    further away from the voters own opinion than the value of wtl
    will not affect the voter's opinion. For example, an influence with an opinion of .9 will not
    affect the opinion of a voter with opinion .3 and wtl = .2 since .9 is not in the range of
    [.3-.2, .3+.2]
formed: float between 0 and 1; represents how formed a voter's views are. The higher the value of
    formed is, the less any influence will affect him. Each time the voter interacts, formed increases.
social: float between 0 and 1; how likely a voter is to interact with another voter in 1 round of a
    simulation. 0 will cause a voter to never interact with other voters. 1 will cause the voter to
    interact with another voter every round.
influence: float between 0 and 1; how likely the voter is to change another voter's opinion. Higher
    values make the voter more likely to influence other voters in any given interaction
formingRate (Optional): float between 0 and 1; represents how quickly the voter will form an opinion.
    Higher values cause self.formed to increase faster in each interaction

    FUNCTIONS:
    Signature: Voter.interaction(self, influencer, plot='false', axGiven=0)
    Docstring:
    Simulates how the voter's opinion will change based on an interaction with another voter based on
    characteristics of the other voter, the voter, and random other factors. This method will change
the
    voter's opinion characteristic.

    Parameters:
    -----------------------------------------
    influencer: Voter object; the news that the voter will interact with
    plot (optional): string; if plot = "true", a plot of the interaction will be created
    axGiven (optional): plot axis; if given and plot = "true", the plot will be placed on the given
        axis

    Returns: nothing

    Signature: Voter.news_interaction(self, influencer, plot='false', axGiven=0)
    Docstring:
    Simulates how the voter's opinion will change based on a piece of news media based on
```

```
    characteristics of the news, the voter, and random other factors.

    Parameters:
    ------------------------------------------
    influencer: News object; the news that the voter will interact with
    plot (optional): string; if plot = "true", a plot of the interaction will be created
    axGiven (optional): plot axis; if given and plot = "true", the plot will be placed on the given
        axis

    Returns: nothing


    Signature: Voter.voterTraits(self)
    Docstring:
    Prints a string with the voter's current characteristics

    Parameters: none

    Returns: string; describes the voter's current characteristics

News class
Init signature: News(opinion=0.5, influence=0.5, credibility=0.5)
Docstring:      Represents a piece of media relevant to the issue in the simulation
Init docstring:
Parameters:
----------------------------------
opinion: float between 0 and 1; represents the political stance the news takes
influence: float between 0 and 1; represents how convincing the news article is. This will affect
    voters of all education levels similarly
credibility: float between 0 and 1; represents how credible the news is. Less credible sources will
    influence voters with higher education less

FUNCTIONS: none

NewsSpace class
Init signature: NewsSpace()
Docstring:      Stores the collection of all news available to voters

FUNCTIONS
    Signature: NewsSpace.add1News(self, news)
    Docstring:
    adds a specific news object to the news space

    Parameters:
    news: News object; the news item to be added to the news space

    Returns: nothing

    Signature:
    NewsSpace.addNews(
        self,
        num_of_news=1,
        opinion_msd=(0.5, 0.2),
        influence_msd=(0.5, 0.2),
        credibility_msd=(0.5, 0.2),
        opinions='spread',
    )
    Docstring:
    Adds multiple news articles to the news space according to the distribution specified in the
```

```
    parameters

    Parameters:
    ------------------------
    num_of_news: int greater than 0; number of news articles to add to the news space
    opinion_msd: array [m, sd]; array should have the form [m, sd]. m and sd represent the mean and
        standard deviation of opinions of the news that will be generated with a normal distribution
    influence_msd: array [m, sd]; m and sd represent the mean and standard deviation of influence of
        the news that will be generated with a normal distribution
    credibility_msd: array [m, sd]; m and sd represent the mean and standard deviation of credibility
        of the news that will be generated with a normal distribution

    Returns: nothing


    Signature: NewsSpace.getNews(self)
    Docstring:
    Parameters: none
    Returns: list; the list of news objects making up the news space


INDEPENDENT FUNCTIONS:

society_generator function
Signature:
society_generator(
    population,
    opinion_msd=(0.5, 0.175),
    opinions='normal',
    opinionPeaks=1,
    educ_scale='off',
    wtl_msd='off',
    formed_scale='off',
    social_msd='off',
    influence_msd='off',
)
Docstring:
society_generator generates a list of Voter objects based on the distribution specified by the input
parameters.

Parameters:
-----------------------------------
population: int; number of voters in the society
opinion_msd (optional): array, either of shape [m,sd] or [[m1,sd1], [m2, sd2],...]; represents the mean
    of the opinion distribution
    if opinions == "spread": array should have the form [m, sd]. m and sd represent the mean and
        standard deviation of opinions of the population that will be generated with a normal
        distribution
    if opinions == "n-modal": array should have the form [[m1,sd1], [m2, sd2],...]. m1, sd1 represent
        the mean and standard deviation of the first normal distribution. m2, sd2 represent the mean
        and standard deviation of the second normal distribution, and so on.
    if opinions == "binary": array should have the form [m, sd]. m and sd represent the mean and
        standard deviation of opinions of the population that will be generated with a characteristic
        equation-like distribution. The distribution will follow the form of the function outlined in
        charEq()
    if opinions == "uniform": opinions_msd does not affect the distribution and can be left blank
opinions (optional): string; describes how the opinions of the society will be distributed.
    if opinions == "normal": Opinions will be normally distributed
    if opinions == "n-modal": Opinions will be a superposition of normal distributions
```

```
    if opinions == "binary": Opinions will be distributed so that it is likely that opinions will be
        either close to 0 or close to 1. The distributions will be defined by the function outlined in
        charEq
    if opinions == "uniform": Opinions will be uniformly distributed
opinionPeaks (optional): int greater than 0; if opinions == "n-modal", this is the number of peaks that
    the distribution will have educ_scale (optional): float; represents how educated the population is.
    Education is represented by a gamma distribution with k = 3, and this is the scale of the
    distribution
    if educ_scale == "off", all voters will have the maximum education and hence the model will not
        take education into account
wtl_msd (optional): array (m, sd); represents the mean and standard deviation that the willingness to
    listen values of the population will have
    if wtl_msd == "off", all voters will have wtl = 1 and the willingness to listen radius will not be
        taken into account in the model
formed_scale (optional): float; represents how formed the population's opinions are. formed is
    determined by a gamma distribution with k = 3, and this is the scale of the distribution
    if formed_scale == "off", all voters will have formed = 0
social_msd (optional): array (m, sd); represents the mean and standard deviation that the social values
    of the population will have
    if social_msd == "off", all voters will have social = 1 and hence all voters will interact every
        round
influence_msd (optional): array (m, sd); represents the mean and standard deviation that the influence
    values of the population will have
    if influence_msd == "off", all voters will have influence = 1 and hence the model will not take
        into account people's influence

Returns: list; returns a list of voter objects with properties specified by the parameters

experiment function
Signature:
experiment(
    society,
    num_periods,
    newsList=0,
    plot_each_round='false',
    plot_each_int='false',
    plot_news='false',
    newsMethod='at start',
    bins=20,
)
Docstring:
Runs a simulation of a society and how interactions change the opinions of the society. First, the
simulation exposes voters to news. Then the simulation allows voters to talk to each other and change
each other's opinions.

experiment makes 3 histograms:
    Histogram 1: the initial distribution of voter opinions
    Histogram 2: the distribution of voter opinions after being exposed to news
    Histogram 3: the distribution of voter opinions after interacting with each other. This displays
        the final state of the society.

Parameters:
-----------------------------------
society: list of Voter objects; list of voters that the experiment will simulate
num_periods: int; number of "rounds", or opportunities to interact, that the voters will have
newsList (optional): list of News objects; list of News objects that voters will be exposed to. If left
    blank, the voters will not be exposed to news.
plot_each_round (optional): string; if plot_each_round = "true", a histogram of opinion distributions
    will be shown after each round of interactions
```

plot_each_int (optional): string; if plot_each_int == "true", a plot of every interaction will be produced. Only turn on for small simulations.
newsMethod: string; unused. If we add different ways for voters to consume news, this will specify the desired method.
bins (optional): int, number of bins in the histogram
Returns: nothing

SUPPLEMENTAL FUNCTIONS:

charEq function
Signature: charEq(lims=[0, 1], switch=0, xrange=[-10, 10], sd=6.283185307179586)
Docstring:
Returns a function y = f(x) that is the solution to the differential equation: y' = C(y-a)*(y-b)
You must provide initial conditions

Parameters:
----
lims: array [a, b]; where a, b are constants in the equation in the description
switch: float; the x value where y = (a+b)/2
xrange: array [xmin, xmax]; the range for which you want the function to provide stable solutions to the differential equation
sd: float; sd = c/2pi

Returns: function; returns a function object y = f(x) that is the solution to the differential equation
y' = C(y-a)*(y-b) that is stable within the xrange given in Parameters

Graph of the solution will look like this:

```
------------------------
a.....
         .
           .
            s
             .
               .
                   ......b
-------------------------
```

At a, y = a
At b, y = b
At s, (x,y) = (switch, (a+b)/2)
The function will flip around s if xrange[0] > xrange[1] (I think, I am not sure. Either way it is better to transform the function after it has been created)

interact function
Signature: interact(p1, p2, plot='false', axGiven=0)
Docstring:
Simulates an interaction between 2 people. Based off of parameters given in the Voter class, each voter's opinion may change

Parameters:
----
p1: Voter; one of the people in the interaction
p2: Voter; the other person in the interaction. Order of p1, p2 does not patter
plot: string; if plot = "true", a plot of the interaction is shown
axGiven: plotting axis; axis on which to plot the interaction; if none is given and plot = "true", a new figure and plot will be created.

Returns: nothing

Sample code:

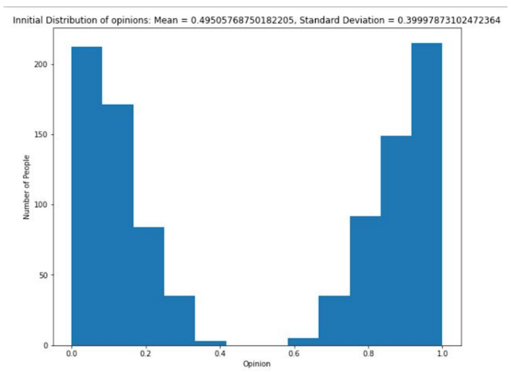
Here is some sample code that shows how to correctly set up and run an experiment. Please note that if you import this as a module, you will have to add a path to each of the functions we define. For instance, say you type

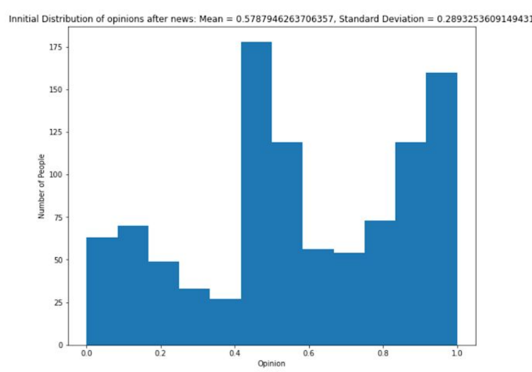
import voterSimulation as vs

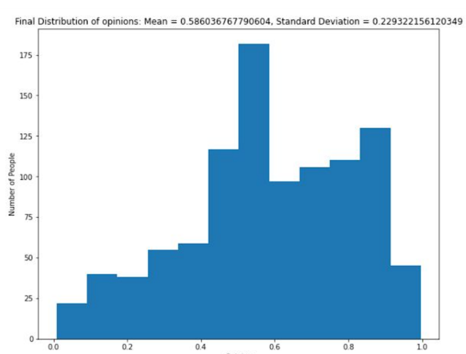
instead of society_generator, you would have to type vs.society_generator.

```
"""Example of how to run an expiriment"""
#creates a society with the specified characheristics
pop = society_generator(1000,
                        opinion_msd = [[.1, .1], [.9,.1]],
                        opinions = "n-modal",
                        opinionPeaks = 2,
                        educ_scale = .15,
                        wtl_msd = (.8, .2),
                        formed_scale = "off",
                        social_msd = (.5, .2),
                        influence_msd = (.8, .2))
pop.append(Voter(opinion = .1, educ = .1, wtl = .05, formed = .8, social = 1, influence = .8)) #adds a voter with these specified traits to the population
newsSpace = NewsSpace() #creates a news space
news = News(opinion = .95, influence = .7, credibility = .75) #creates a news article
newsSpace.add1News(news) #adds that news article to the news space
newsSpace.addNews(num_of_news = 1, opinion_msd = (.5, .05), influence_msd = (.7, .1), credibility_msd = (.75, .1)) #adds a bunch of news articles to the news space
experiment(pop, num_periods = 100, newsList = newsSpace.getNews(), bins = 12, plot_each_round = "false", plot_each_int = "false")#runs the expiriment
```

Output:

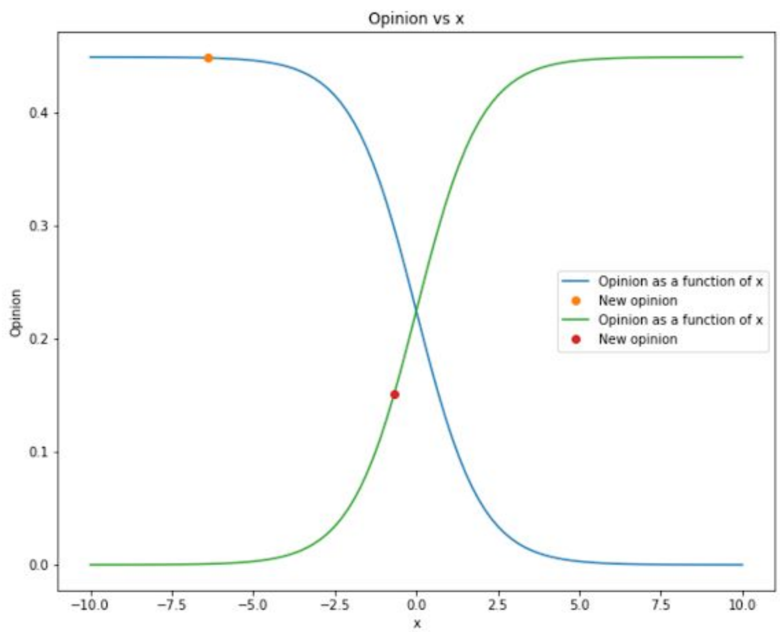






Here is some sample code showing how to get a plot of an interaction between two people:

```
pop = society_generator(1000,
                        opinion_msd = [[.1, .5], [.9,.5]],
                        opinions = "n-modal",
                        opinionPeaks = 2,
                        educ_scale = .15,
                        wtl_msd = (.8, .2),
                        formed_scale = "off",
                        social_msd = (.5, .2),
                        influence_msd = (.8, .2))
interact(pop[0], pop[1], plot = "true")
```

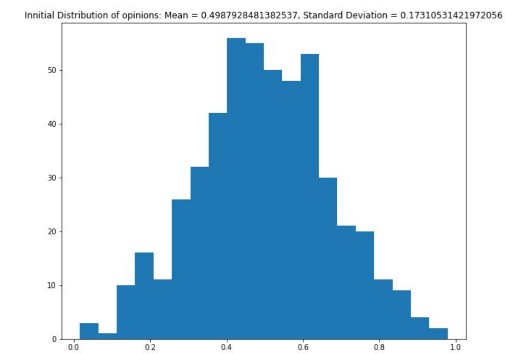


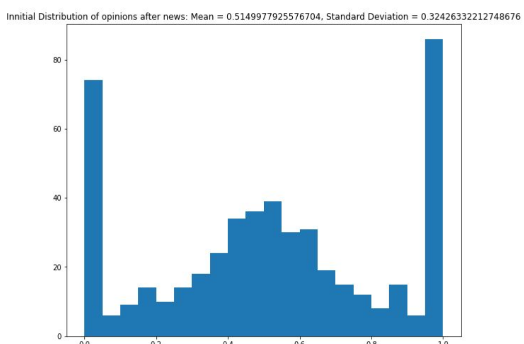
Results

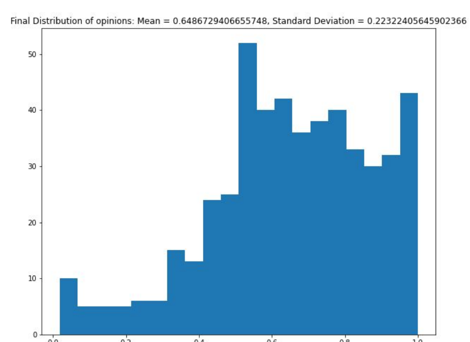
In this section, we will show some interesting behavior that our model exhibited after very limited usage. We have developed a framework with infinite possibilities for models that can be created, each with its own unique behavior. Rather than attempt to show every possible model that is possible, we will outline a few models that demonstrate interesting behavior in order to demonstrate how our software can be used.

Simulation 1 (see appendix for simulation parameters)

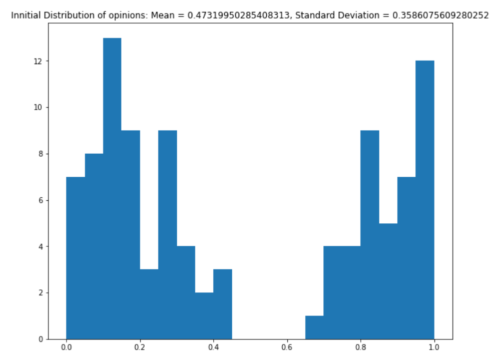
Example 1(top to bottom)

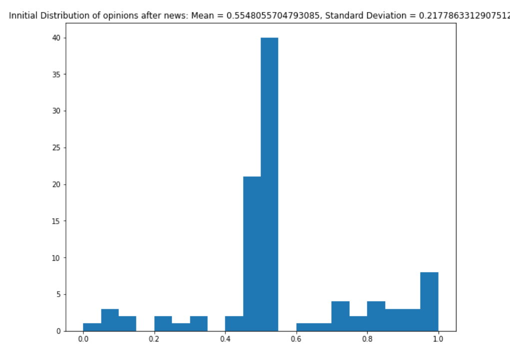


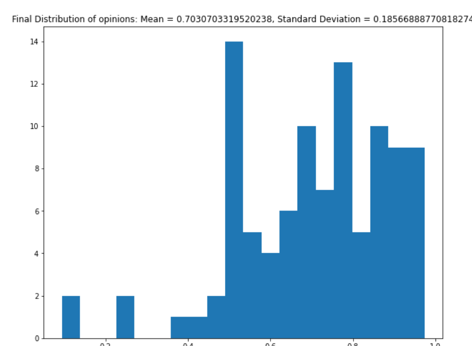




Example 2(top to bottom)



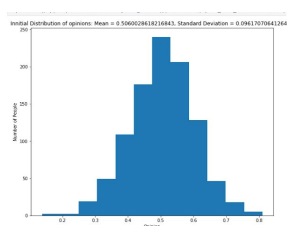
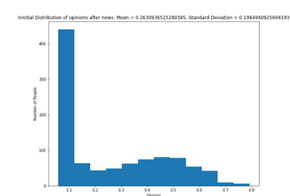
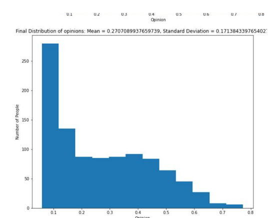
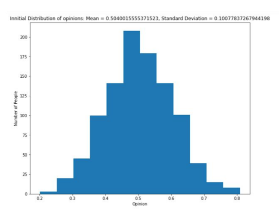
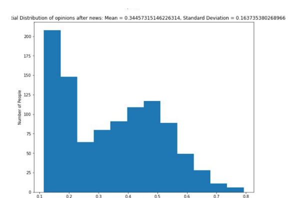
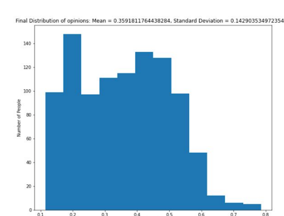
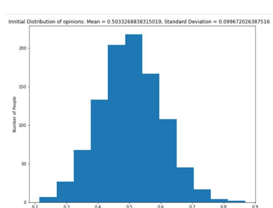
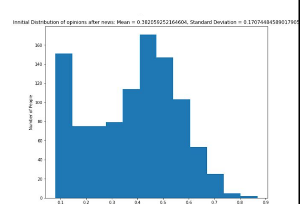
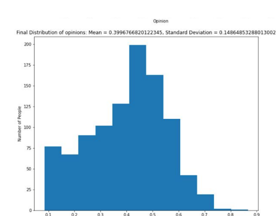




Example 1 shows us a society with an initially normal political distribution. Meaning most voters are centrists or close to being centrists, and there are little to no extremists within the population. The second graph shows the distribution of opinions after being exposed to binarily distributed news sources. One can observe that the news has created a drastic change in political opinions, with many people holding extreme beliefs at both ends of the spectrum. Once they interact with one another, the overall beliefs shift heavily toward one end of the spectrum. We posit that because of the way our model is set up, certain circumstances might cause a difference between the ends of the spectrum that could exponentially change the results due to the nature of interactions, thus why we see a heavily skewed distribution. Regardless, opinions have shifted drastically due to the news sources' influence.

In example 2, we observe a society with a bi-modal distribution. Most people's beliefs reside on or towards either end of the spectrum. We introduce very centrist news sources, and run the interaction function. The resulting opinion distribution has many people holding centrist beliefs, as well as a skew of people toward one end of the spectrum.

How populations with different educations respond to news with low credibility

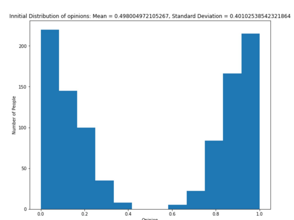
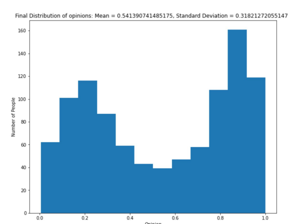
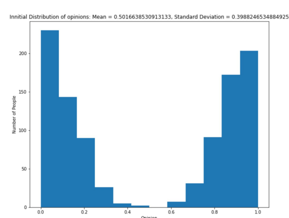
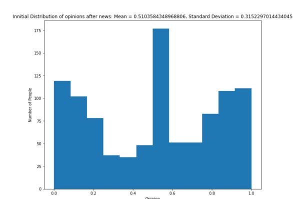
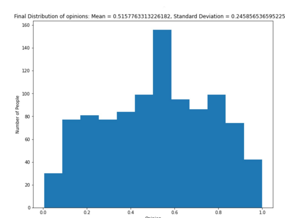
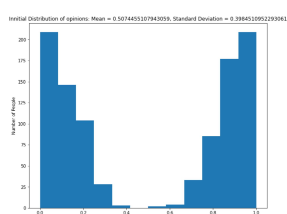
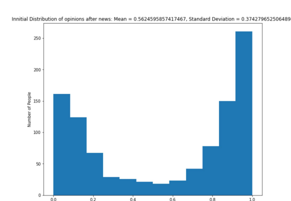
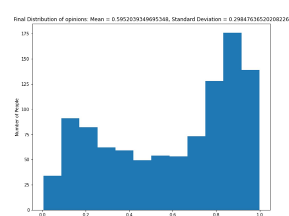
Simulation 2	Original distribution of opinions	Distribution of opinions after news	Distribution of opinions after discussions
Population 1: Low educated population (educ_scale = .05)[4]			
Population 2: Average educated population (educ_scale= .1)			
Population 3: Well educated population (educ_scale = .15)			

[4] See the documentation for society_generator for more information on the specific distribution of education. We used a gamma distribution with k = 3 for the distribution of educations.

In the above simulation, we begin with a society that has normally distributed opinions around .5. Each society is then exposed to news with low credibility and opinion = .1. The tables show how the opinions of a low, average, well educated society respond to the news according to our model. Models ran for 100 rounds of interactions and demonstrated a steady state of opinion distributions in the final rounds.

The low educated society forms an almost unimodal distribution around the opinion of the news. The well educated society has a group that believes the fake news at first, but then are convinced through interacting with other people to return to a normal distribution of opinions. The average educated society has a mix of the effects of both the well and low educated society, eventually forming a bimodal distribution of opinions.

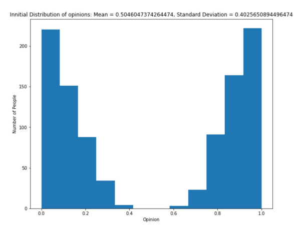
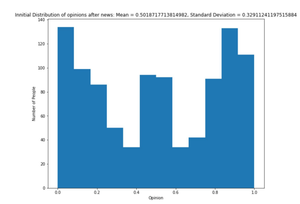
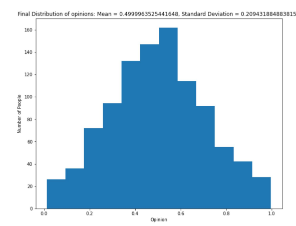
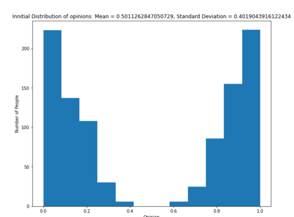
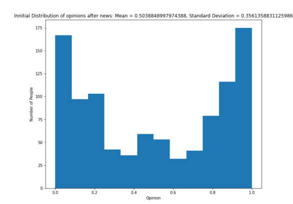
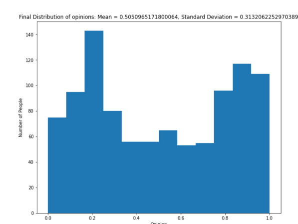
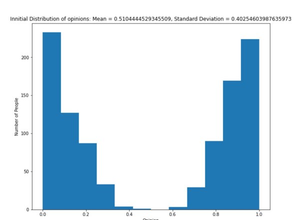
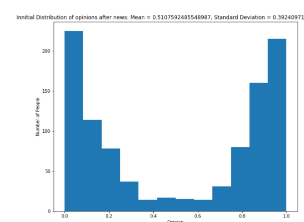
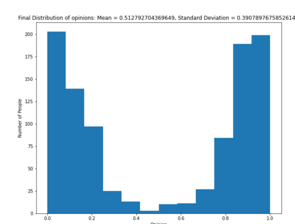
How different news sources affect a partisan distribution of opinions

Simulation 3	Original distribution of opinions	Distribution of opinions after news	Distribution of opinions after discussions
Population 1: No news source		(same as before news)	
Population 2: Credible news source with opinion = .5			
Population 3: Credible news sources, half with opinion = .1, and half with opinion = .9			

In the above model, a society with a heavily partisan distribution of opinions is exposed to various types of news. Models ran for 100 rounds of interactions and demonstrated a steady state of opinion distributions in the final rounds.

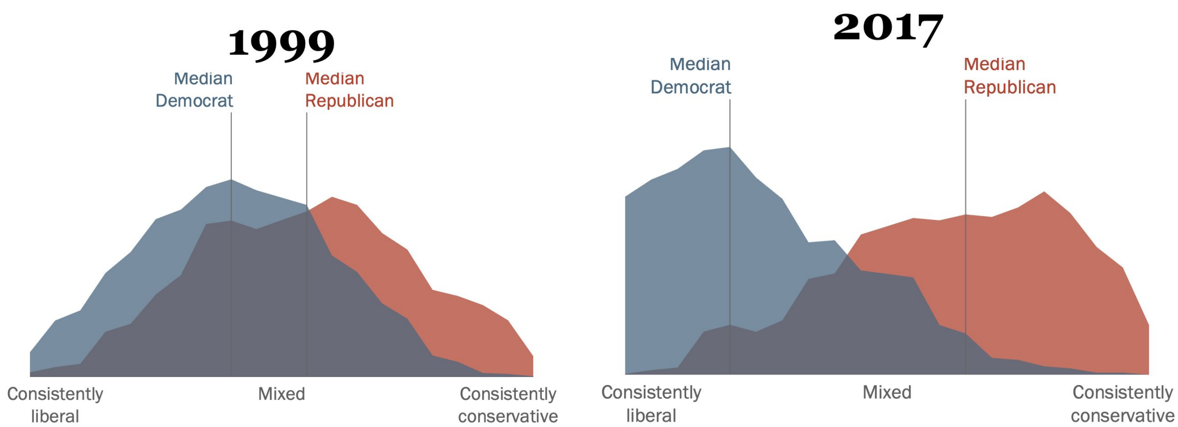
Population 1 relaxed its opinion distribution to a bimodal distribution. Interestingly Population 3 also relaxed into a bimodal distribution with a lower standard deviation of opinions than Population 1, showing that the addition of partisan news to a partisan society still created a slightly less partisan distribution of opinions. This may be because the mean willingness to listen radius was high, so opposition news consistently influenced voters. Population 2 relaxed into a unimodal distribution, showing that exposure to moderate news heavily reduced partisan opinions.

How partisan populations with different willingness to listen radii react to moderate news

Simulation 4	Original distribution of opinions	Distribution of opinions after news	Distribution of opinions after discussions
Population 1 Mean willingness to listen = .8			
Population 2 Mean willingness to listen = .5			
Population 3 Mean willingness to listen = .2			

In the above simulation, societies with various willingness to listen radii were exposed to moderate news. Population 1 formed a unimodal distribution, Population 2 formed a moderate bimodal distribution, and Population 3 formed a severe bimodal distribution. Unsurprisingly, populations with higher willingness to listen radii had demonstrably less partisan opinion distributions.

We include this example because it has direct applications to social media and how it influences how opinions of a population are distributed. Studies have shown that people are less likely to listen to others with differing opinions online than they are in a face-to-face conversation. This quantitatively shows how influential this phenomenon is on the scale of an entire population. Notice how similar these histograms are to the histograms of opinion distribution in the United States outlined in the introduction.



Notice how in 1999, before social media became widely used, the partisan divide was small. In 2017, when social media use was common, a larger partisan divide developed. While we do not claim causality here as we do not have the data to back this claim up, it is possible that the general lowering of the "willingness to listen radius" in the United States due to social media has played some part in widening the partisan divide.

Conclusion

Our beliefs and views are not set in stone. They can change and evolve through our own volition as well as through the influence of others. The adaptive nature of our opinions can be both positive and negative. It demonstrates the faculty for human beings to learn and grow from our misconceptions. In an ever changing world, this ability to reevaluate one's beliefs is invaluable.

However, it can also mean that we are subject to misinformation and influence from false sources. It can also mean that our opinions can change for the worse based on our biases and our capacity or willingness to be a discerning thinker. The results showed the great influence that new sources have on people's opinions and views. For example, those with normal distribution of opinions became more polarized after being exposed to polarizing news. In contrast, populations that were already polarized became more centered around the median once they were exposed to centrist views on the news. Furthermore, both of these distributions of opinions shifted to one side of the spectrum after interacting with each other.

These results show that one's opinions are constantly susceptible to change and influence. As mentioned above, this can be both a positive and negative thing. Different characteristics of the population can have an influence on the distribution of opinions as well. Our results showed that the more educated population became increasingly centrist and moderate after exposure to low credibility news sources and interactions with one another. This demonstrates that more educated populations are less susceptible to influence from misinformation than less educated populations. Another characteristic that was studied was a population's willingness to listen. Populations that had higher willingness to listen became moderate after exposure to moderate news sources and interaction with one another. In contrast, populations with low willingness to listen stayed polarized even after they had received the same treatment as the population with a higher willingness to listen.

This is especially prevalent in today's society since research has shown that people have a lower willingness to listen on social media compared to personal interactions. In a world in which interactions are becoming increasingly centered around virtual mediums such as social media, this inability to change one's mind does not bode well. However, this study has shown that there are multiple factors that influence one's opinions. Human interactions are complex and convoluted and at times it may seem like we are becoming more divided than ever. However, being cognizant of our influences and striving to understand these factors can bridge the ever growing divide among humanity. This divide exists because our beliefs and opinions are not set in stone, they are subject to change and growth. It is important to remember that the forces that divide us can just as easily be used to mend and connect this polarized world.

Future Work

Add a more dynamic news publishing system

One of the major shortcomings of our experiment is the way the voters interact with news. Our method exposes voters to news one time, and runs the interaction where they talk and influence one another. In ideal circumstances, the voters would be continuously exposed to different news sources. This would allow our model to be more in tune with the social media model we were aiming for, where a majority of the political influence is fueled by interactions with social media. For a hypothetical second iteration of our work, we would incorporate continuous exposition to news sources in order to more realistically fit our model to real-world habits.

For a more realistic model, voters would ideally be exposed mostly to news that aligns closely with their current political beliefs, as most people get their news from sources that they trust (confirmation bias). From this some more interesting observations might occur. We might find that the news has more or less impact thanks to the nature of voters being mostly exposed to news sources that reinforce their opinions. We can hypothesize that these circumstances would make events where opinions diverge more divergent, and events where opinions converge more convergent, making overall opinion changes more volatile.

Add the possibilities of sub-groups (like parties)

A potential element that could be incorporated into a second iteration of our project would be the ability to create sub groups, such as party affiliation. This would be a tool that would further allow us to simulate real world experiences, where being subscribed to an individual political party influences how a person votes. Additional ways of grouping individuals might also affect how they vote, such as the church they go to, their school or where they work, etc. By incorporating this ability to group people, we would be able to draw conclusions about group think, and the way people influence each other, particularly in groups where outside opinions are unaccepted. This would be important to understand how easily people are influenced by others, particularly by a group that they belong to which generally thinks the same.

Quantify physical parameters

Additionally, quantifying physical parameters could be an option to draw interesting conclusions about voting habits. For example, we could create a hypothetical experiment where things such as race/ethnicity, gender, sexual orientation, religious beliefs, etc are quantified. We could then observe how these factors contribute to people's political beliefs. Such tools would allow us to understand the influence that these factors have on our politics. For example, if one is a minority ethnicity and part of the lgbtq+ population, they might be more likely to be progressive and support more liberal policies. We could then compare this to an experiment where we do not quantify these factors, and see if/what things change between the two societies. Overall, we are not very limited in terms of future applications and iterations for our project. The voter's influence factors can be changed to simulate a wide array of circumstances, allowing us to understand many different things about the way our political beliefs are formulated and influenced.

Sources

"The Shift in the American Public's Political Values." Pew Research Center - U.S. Politics & Policy. Pew Research Center, July 2, 2020. https://www.pewresearch.org/politics/interactives/political-polarization-1994-2017/

Compass, The Political. "The US Presidential Candidates 2016." The Political Compass. Accessed December 11, 2020. https://politicalcompass.org/uselection2016.

"Single Peaked Preferences." Wikipedia. Wikimedia Foundation, December 11, 2020. https://en.m.wikipedia.org/wiki/Single_peaked_preferences.

Doherty, Carroll. "Americans' Growing Partisan Divide: 8 Key Findings." *Pew Research Center*, Pew Research Center, 28 Aug. 2020, www.pewresearch.org/fact-tank/2017/10/05/takeaways-on-americans-growing-partisan-divide-over-political-values/.

Heltzel, Gordon, and Kristin Laurin. "Polarization in America: Two Possible Futures." *Current Opinion in Behavioral Sciences*, Elsevier Ltd., Aug. 2020, www.ncbi.nlm.nih.gov/pmc/articles/PMC7201237/.

Appendix

Simulation 1 parameters

Example 1

```
society_generator parameters:
      population = 1000
      opinion_msd = [[.1, .1], [.9,.1]]
      opinions = "normal"
      opinionPeaks = 0
      educ_scale = 0.5
      wtl_msd = (.4, .2)
      formed_scale = 0.1
      social_msd = (.5, .2)
      influence_msd = (.5, .2))
News added:
newsSpace = NewsSpace()
news = News(opinion = .1, influence = .8, credibility = .1)
newsSpace.addNews(num_of_news = 40, opinionPeaks = 0,
                  Opinion_msd = (0.5, 0.001), influence_msd = (1, 0.001),
                  Credibility_msd = (0.5, 0.01), opinions="normal")
experiment(population, newsList = newsSpace.getNews())
```

Example 2

```
society_generator parameters:
      population = 1000
      opinion_msd = [[.1, .1], [.9,.1]]
      opinions = "n-modal"
      opinionPeaks = 2
      educ_scale = 0.5
      wtl_msd = (.4, .2)
      formed_scale = 0.1
      social_msd = (.5, .2)
      influence_msd = (.5, .2))
News added:
newsSpace = NewsSpace()
news = News(opinion = .1, influence = .8, credibility = .1)
newsSpace.addNews(num_of_news = 40, opinionPeaks = 0,
                  Opinion_msd = (0.5, 0.001), influence_msd = (1, 0.001),
                  Credibility_msd = (0.5, 0.01), opinions="normal")
experiment(population, newsList = newsSpace.getNews())
```

Simulation 2 parameters

```
society_generator parameters:
      population = 1000
      opinion_msd = [[.1, .1], [.9,.1]]
```

```
      opinions = "n-modal"
      opinionPeaks = 2
      educ_scale =
            Population 1: .05
            Population 2: .10
            Population 3: .15
      wtl_msd = (.8, .2)
      formed_scale = "off"
      social_msd = (.5, .2)
      influence_msd = (.8, .2))
News added:
news = News(opinion = .1, influence = .8, credibility = .1)
The experiment was run for 100 rounds
```

Simulation 3 parameters

```
Society Generator parameters:
      population = 1000,
      opinion_msd = [.5, .1]
      opinions = "normal"
      opinionPeaks = 1
      educ_scale = .15
      wtl_msd = (.8, .2)
      formed_scale = "off"
      social_msd = (.5, .2)
      influence_msd = (.8, .2))
News added:
      Population 1: was not exposed to news
      Population 2:
            news1 = News(opinion = .5, influence = .7, credibility = .75)
            news2 = News(opinion = .5, influence = .7, credibility = .75)
      Population 3:
            news1 = News(opinion = .95, influence = .7, credibility = .75)
            news2 = News(opinion = .05, influence = .7, credibility = .75)
The experiment was run for 100 rounds
```

Simulation 4 parameters

```
Society_generator parameters:
      population = 1000,
      opinion_msd = [[.1, .1], [.9,.1]],
      opinions = "n-modal",
      opinionPeaks = 2,
      educ_scale = .15,
      wtl_msd =
            Population 1: (.8, .2)
            Population 2: (.5, .2)
            Population 3: (.2, .2)
```

```
      formed_scale = "off",
      social_msd = (.5, .2),
      influence_msd = (.8, .2))
News added:
news1 = News(opinion = .5, influence = .7, credibility = .75)
```