Data Processing: Ballot Processing

Group 42: Elise Penn, Manish Vuyyuru, Victor Sheng, Yajaira Gonzalez

Processes general congressional ballot data for input into our model. Needs access to the files available on the team GitHub folder, which are too large and numerous to upload here.

In [ ]:
```
import pandas as pd
import datetime
import numpy as np
import pickle
```

In [ ]:
```
def drop_rows(df, column, value):
    ''' Drop a row where 'value' is in 'column'. Only grabs first item.
    arguments:
        df -- dataframe to modify
        column --  the column you want to search
        value -- if you find this value in the column, drop the row
    returns:
        dataframe without specified rows
    '''
    try:
        new_df = df.drop(df[column][df[column]==value].index[0])
    except IndexError:
        new_df = df
    return new_df
```

In [ ]:
```
def get_election_day(year):
    ''' Get the date of election day in a given year
    arguments:
        year -- year as int
    retuns:
        datetime object of election day. '''
    if year%2 == 1:
        raise ValueError('No election in even years.')
    # possible days = Nov. 2 - Nov. 8
    possible_days = [datetime.datetime(year, 11, d) for d in range(2,9)]
    for day in possible_days:
        if day.weekday()==1: # return if it's a Tuesday
            return day
```

In [ ]:

```
def get_mean_spread(year,n_days_before_election):
    ''' Cleans data and converts each file into mean spread of all the polls
    N days before the election
    arguments:
        year -- (int) the year you want to pull poll data from
                there must be a file with the name
                'Datasets/YYYY_generic_congressional_vote.csv'
        n_days_before_election --  (int) maximum number of days before an
                                    election a poll should end to be included
                                    in your estimate
    returns:
        mean of the spread N days before the election in that year (float)
        '''
    # read in data file
    ballot_df = pd.read_csv('Datasets/'+str(year)+'_generic_congressional_vote.cs

    # Data Cleaning
    ballot_df = drop_rows(ballot_df, 'Poll', 'Final Results')
    ballot_df = drop_rows(ballot_df, 'Poll', 'RCP Average')

    election_day = get_election_day(year)

    # make spread standardized around 0
    if year >= 2014: # they changed their column names after 2014
        ballot_df['Spread'] = ballot_df['Democrats (D)'] - ballot_df['Republicans
    else:
        ballot_df['Spread'] = ballot_df['Democrats'] - ballot_df['Republicans']

    spread = []
    for index, row in ballot_df.iterrows():
        # clean up the date format
        dates = row['Date'].split('-')
        start = datetime.datetime.strptime(str(year)+'/'+dates[0].strip(),'%Y/%m/
        #end = datetime.datetime.strptime(str(year)+'/'+dates[1].strip(),'%Y/%m/%

        # take all the polls which started less than 4 weeks ago
        if (start - election_day).days <= n_days_before_election:
            spread.append(row['Spread'])

    # find the mean of the spread over the last 4 weeks
    return np.mean(spread)
```

In [ ]:
```
def format_national_polls(years, n_days_before_election=28):
    ''' Format generic congressional vote into a dataframe with indices
        of AA_00_0000 (state abbr., district, year). For example, WI_04_2016.
        There must be a file named 'Datasets/YYYY_generic_congressional_vote.csv'
        for each year you want to process data.
    inputs:
         years -- (list) list of years you want to put in the dataframe
         n_days_before_election -- (int) maximum number of days before an
                                   election a poll should end to be included
                                   in your estimate
    returns:
        None.
        For each year, the mean of the spread N days before the election in that
        calculated. Then we throw it into all the districts for that year.
        Dumps a dataframe with the proper indexing into 'Datasets/national_poll.p

    formatted_poll_df = pickle.load(open('Datasets/master_index.p','rb'))
    formatted_poll_df['national_poll'] = np.nan # add new column to the df
    for year in years:
        spread = get_mean_spread(year,n_days_before_election)
        formatted_poll_df.loc[formatted_poll_df['year']==year, 'national_poll'] =
    pickle.dump(formatted_poll_df, open('Datasets/national_poll.p','wb'))
```

In [ ]:
```
# make the clean data file
years = [2002,2004,2006,2008,2010,2012,2014,2016,2018]
format_national_polls(years, n_days_before_election=28)
```

In [ ]:
```
# test the clean data file
pickle.load(open('Datasets/national_poll.p','rb'))
```

In [ ]:
```

```