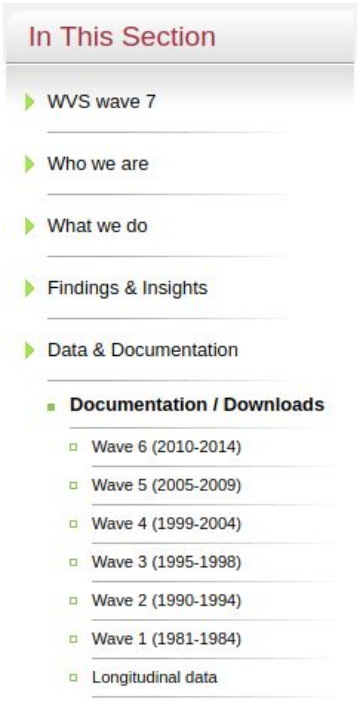
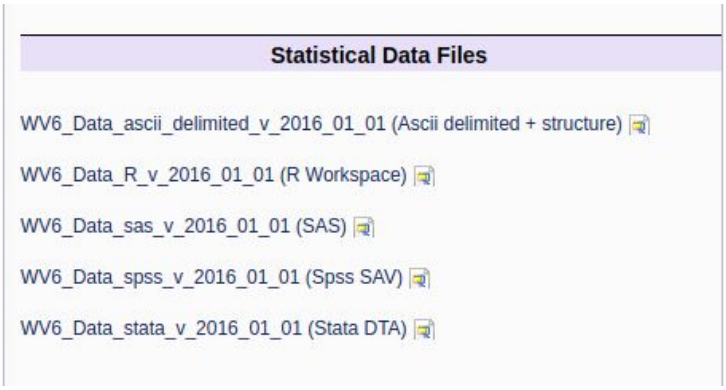
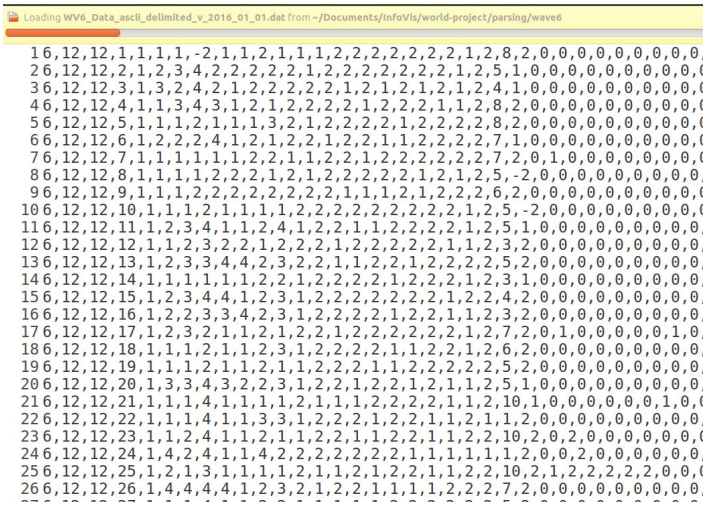
# A small subset of my learning :




Getting data for WVS




---




Handle the format

Find the columns we're looking for

```
VARIABLES
   V1                1   (F1) {Wave}  \V1
   V2                2   (F3) {Country Code}  \V2
   V2A               3   (F3) {Country/regions [with split ups]}  \V2A
   V3                4   (F7) {Interview number}  \V3
   V4                5   (F2) [<= -1]   {Important in life: Family}   \V4
   V5                6   (F2) [<= -1]   {Important in life: Friends}  \V5
   V6                7   (F2) [<= -1]   {Important in life: Leisure time}   \V6
   V7                8   (F2) [<= -1]   {Important in life: Politics}   \V7
   V8                9   (F2) [<= -1]   {Important in life: Work}   \V8
   V9                10  (F2) [<= -1]   {Important in life: Religion}   \V9
```

Find the countries

```
\V2
   -5 'Missing; Unknown'
   -4 'Not asked in survey'
   -3 'Not applicable'
   -2 'No answer'
   -1 'Don't know'
   4 'Afghanistan'
   8 'Albania'
   12 'Algeria'
   16 'American Samoa'
   20 'Andorra'
   24 'Angola'
   28 'Antigua'
   31 'Azerbaijan'
   32 'Argentina'
   36 'Australia'
   40 'Austria'
   48 'Bahrain'
   50 'Bangladesh'
   51 'Armenia'
   52 'Barbados'
   56 'Belgium'
   60 'Bermuda'
   64 'Bhutan'
   68 'Bolivia'
   70 'Bosnia Herzegovina'
```

```
country_to_code = {
    "Afghanistan": 4,
    "Algeria" : 12,
    "Australia" : 36,
    "Bahrain" : 48,
    "Belgium" : 56,
    "Burundi" : 108,
    "Cameroon" : 120,
    "Canada" : 124,
    "Chad" : 148,
    "Czech Republic" : 203,
    "Denmark" : 208,
    "Egypt" : 818,
    "France" : 250,
    "Georgia" : 268,
    "Germany" : 276,
    "India" : 356,
    "Iran" : 364,
    "Iraq" : 368,
```

Find the right column for each value for each wave

```
wave_question_index = {
    3: {
        "politics" : 8,
        "religion" : 10,
        "happiness" : 11,
        "health" : 12,
        "c. independence" : 16,
        "family" : 5,
        "work" : 9,
        "leasure time" : 7
    },
    4: {
        "politics" : 7,
        "religion" : 9,
        "happiness" : 11,
        "health" : 12,
        "c. independence" : 15,
        "family" : 4,
        "work" : 8,
        "leasure time" : 6
    },
```

```
180         if old_country in code_to_country.keys():
181             # last country was important, save the data
182             waves[wave][code_to_country[old_country]] = country_data
```

250 python lines later

```
{'countries': [{'count': 1127,
                'name': 'Pakistan',
                'values': {'c. independence': {'1': 998, '2': 129},
                           'family': {'1': 989, '2': 119, '3': 18, '4': 1},
                           'happiness': {'-1': 2,
                                         '1': 337,
                                         '2': 726,
                                         '3': 54,
                                         '4': 8},
```

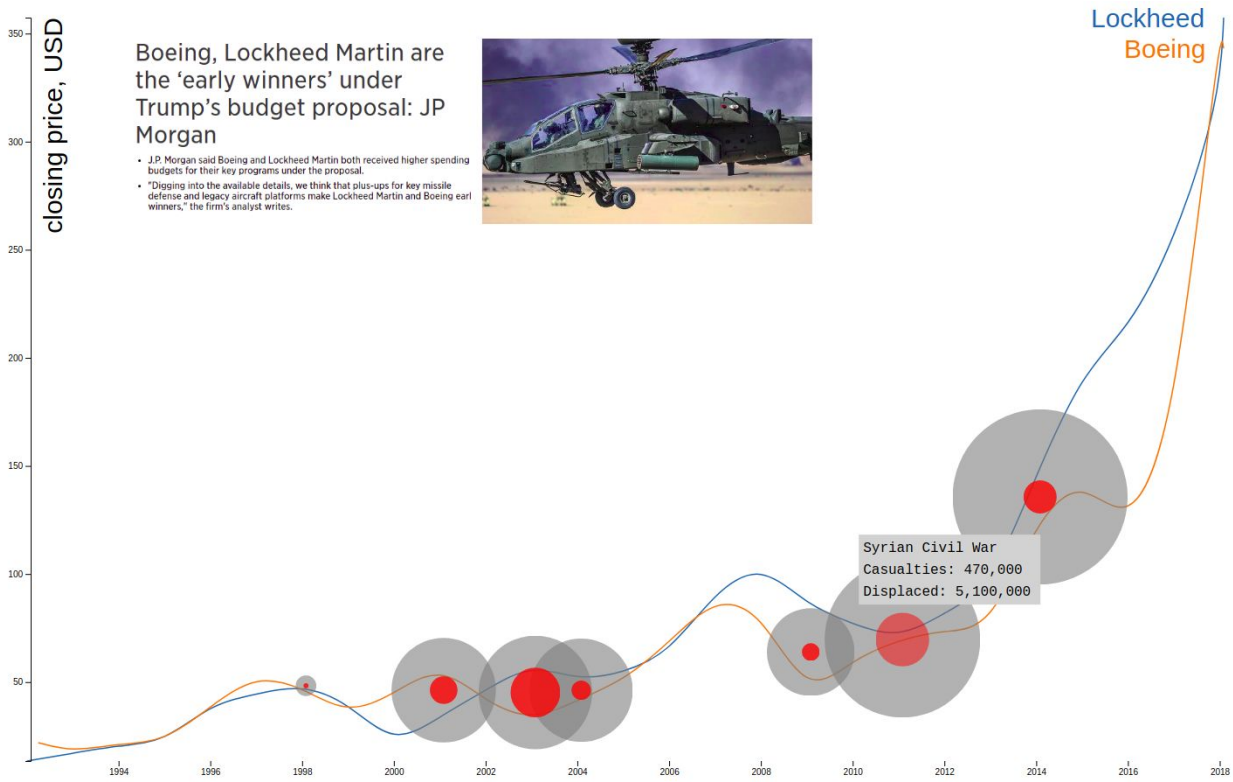
Handle bloated data from gapminder excel-sheets: more python

```
for country in list(data):
    if not (country in countries):
        del data[country]
```

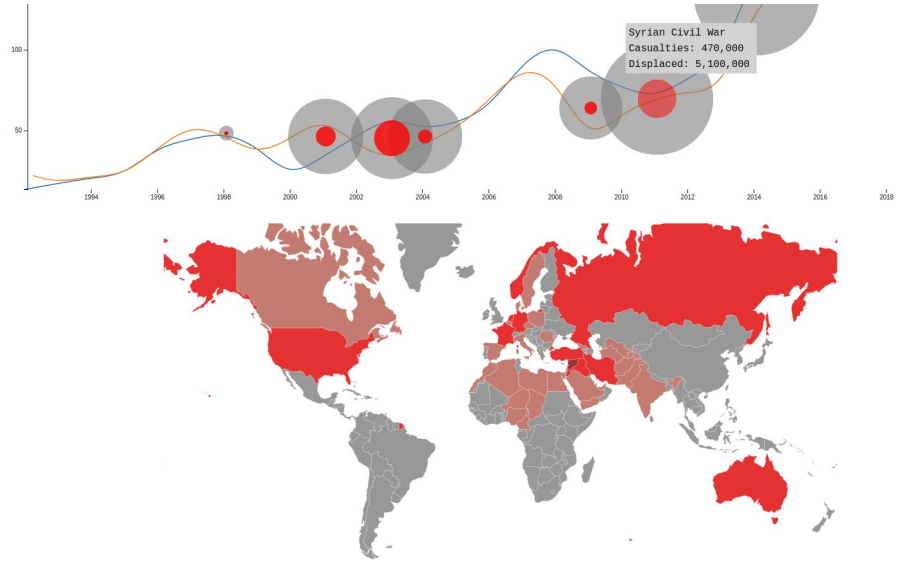
4 MB to 0.7 MB, woo!

Gather context-data,
wikipedia and Yahoo Finance

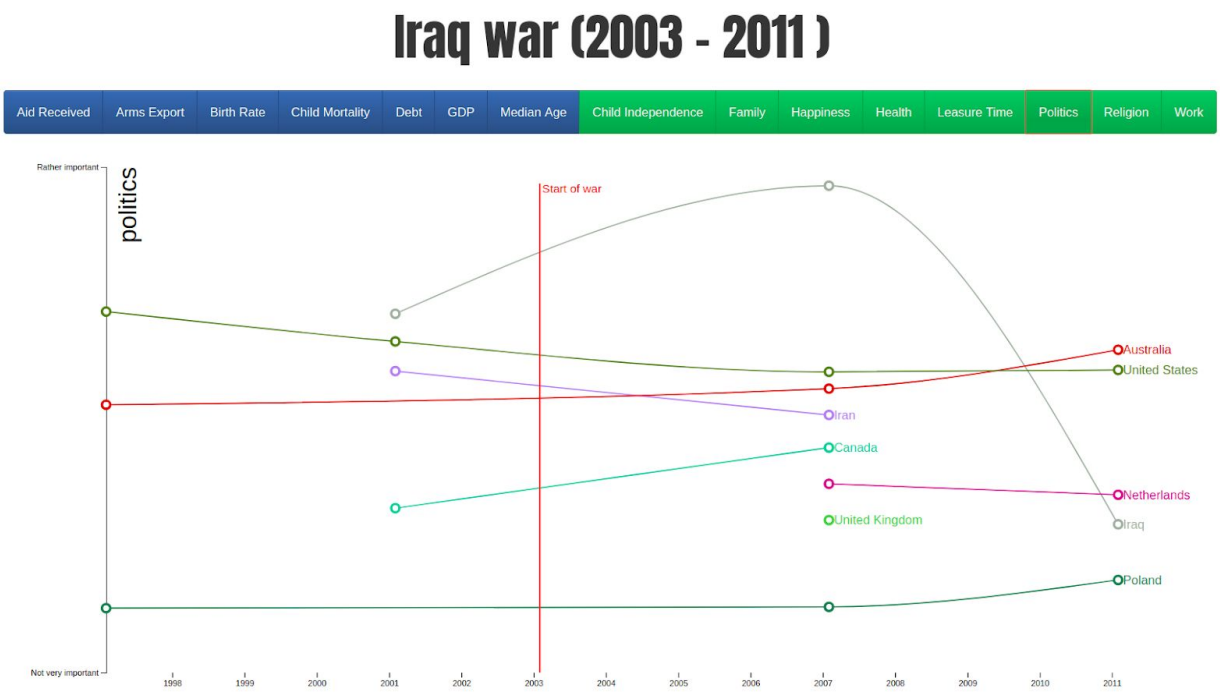

Make combined line chart and scatter plot for context data (learned: mixing plot-types)




Visualize filtered data by linking to world map (learned: maps from vector-data)



Now present gapminder and WVS data as time series data (learned: ordinal y-scale)




Finally details on demand on WVS data (learned: horizontal dotted lines)