**Chromium Code Repository Walk Through**

The code for this project is in the public github repository: <https://github.com/ewulczyn/CR6>

All the data for the project is in the “fenchrom” file share set up by Phil Farell

Data Acquisition (see CR6/acquisition)

The script CR6/acquisition/download.py scrapes the EDF, GEO XY, GEO Z and GEO WELL files for every county from

<http://geotracker.waterboards.ca.gov/data_download_by_county.asp>.

In addition, it scrapes files for every chemical and for every county from

http://geotracker.waterboards.ca.gov/gama/gamamap/public/?CMD=runreport&myaddress=california

The original files can be found in CR6Data/datadump. The files are organized by county in CR6Data/counties. Here each county has it own folder of chemical files.

Alternatively, CR6Data/chems/raw has one file for each chemical, where data has been aggregated over all counties. For files in CR6Data/chems/reduced some columns have been omitted, to reduce the size of the files. For files in CR6Data/chems/yearly\_site\_averages, each row in the file corresponds to the yearly average for that chemical at that well.

Data Cleaning and Preprocessing Tools

The function getAll() in CR6/cleaning/merge.r, is the main tool for preprocessing and aggregating data for analysis. It operates in the CR6Data/counties directory to merge files into a single csv, with one column per chemical in addition to date, well-name and locations, etc, columns.The function getAll() takes the following parameters:

Base: path to CR6Data/counties

Counties: a list of counties to pull data from

Chems: a list of chemicals you want concentrations for

Basechem: the chemical to base the data aggregation off of. This chemical will not have any NA entries even if complete=FALSE

getDepth: Boolean flag specifying if you want well depth data

complete: Boolean flag specifying if you want to remove rows with missing values

Analysis: Simple pair plots of CR6 and other Chemicals

The script CR6/analysis/CR6vsChem.R generates interpretable pair plots between CR6 and a chemical or depth and saves the figures in CR6Data/clean/pairPlots. (Note, here you will find a simple example of using the getAll() function).

Analysis: Exploring differences in concentrations over space and time

The script CR6/analysis/mapConcentrations.R creates a series of maps showing spatial concentrations of a chemical over time. The script takes as input a list of csvs, each of which has the average concentration of the chemical at a well within a specified time range (say 180 days). The csvs are generated by the scripts /CR6/cleaning/ createTimeSeries.R and scripts /CR6/cleaning/ createTimeSeriesFixed.R. The scripts take as input a chemical name, a date range (i.e. 1989-2010) and an aggregation window (i.e. 180 days). The difference between the two scripts is that createTimeSeriesFixed.R only uses wells that have measurements at every aggregation window within the range. This guarantees that you don’t “see” higher concentrations as time goes on because we have more wells measured. The script also produces a time series plot, showing the mean concentration of the chemical in each time range.

Analysis: Exploring spatial differences in correlations between two chemicals

The script CR6/analysis/mapCorrelations.R creates a map showing the nature of correlation between two chemicals at a well and saves the plot in CR6Data/clean/correlations/. It also gives a histogram of correlations to make it easier to see the distribution of correlations. (Often we will see the distribution looks normal around 0!)

Analysis: Exploring the Role of Nitrogen Reducing Bacteria

The script CR6/analysis/NRB.R tries to find a threshold value of nitrogen such that the distributions of the ratio of CR6 to total chromium in wells above and below the nitrogen thresholds are the most different. It also tests the significance of the result by computing the probability of finding such a difference in distributions, under the null hypothesis that there is no influence of NO3 on CR6. (The results are super significant, basically p values of 0).

Analysis: Building a predictive model of CR6

This is very early stage! I tried using a method that elegantly handles a few missing data points called MART, to try build a model for CR6 based on a few chemicals for which we have the most data (not a good choice, since they may have nothing to do with CR6). The script is called CR6/analysis/mart.R. Do not trust the model, I am only including it for reference if you get a more complete data set and want to reuse the code.

Analysis: Clustering

The script CR6/analysis/clustering.R tries to hierarchically cluster chemicals. We try two different measures of similarity, pearson’s correlation which measure how closely the relationship between two chemicals is linear and spearman’s correlation which measures how closely the relationship between two chemicals is monotonic. We try 5 different transformations of the data before computing similarity:

* Including 0 values
* Excluding 0 values
* Mapping 0 values to min observed value
* Log transform of data where 0’s are excluded
* Log transform of data where 0’s are mapped to min observe value

We try 3 different ways of mapping a similarity based on correlation to a distance measure for clustering:

* Map negative correlations to 0: this gives clusters that are highly mutually positively correlated (most interesting and intuitive)
* Take absolute values of correlations: this gives clusters that are highly mutually positively or negatively correlated
* Map positive correlations to 0: this actually does not make a whole lot of sense, you cannot have a group of chemicals larger than 2 that is mutually highly negatively correlated.

The script produces a dendrogram for each condition and saves the result in /CR6Data/dendrograms