1 Overview

In modern scientific research, collaboration has become increasingly important, especially with scientific consortia. For example, the number of published papers indexed by PubMed that are related to scientific consortia is increasing much faster than the number of other papers. Additionally, the level of participation in scientific consortia seems to be on the rise because the number of authors on consortia-related papers seems to be higher than the author count on other papers.

This work focuses on the temporal dynamics of collaboration in these scientific consortia, examining the relationship between consortium researchers and outside scientists who may use the data or other information provided by the consortium. We viewed these collaborations in terms of co-authorship networks, seeing the patterns of how a consortium grows based on the published papers each year.

We have focused on two consortia as case studies: ENCODE [1], which aims to create a comprehensive list of functional elements in the human genome; and modENCODE [2], which seeks to identify the functional elements in the Caenorhabditis elegans (worm) and Drosophila melanogaster (fruit fly) genomes.
2 Data Collection

2.1 Obtaining Publication Information

To obtain preliminary data, I wrote a Python script (see `getEncode.py`) to scrape the ENCODE webpage listing all of their publications. To accomplish this task, I used a program called Selenium Webdriver [3], which specializes in browser automation. This program obtained a list of PubMed IDs that were used to obtain each publication’s XML data from PubMed.

This XML data could then be easily parsed for the publication date, the author list, as well as a number of other attributes of the paper. I wrote an R program (see `processEncodeXML.R`) to extract and aggregate the relevant information from the XML files downloaded by the previous script into a single table.

Obtaining the data for modENCODE followed a very similar procedure, scraping the list of publications, obtaining a set of PubMed IDs, downloading the corresponding XML files, and transforming it into a table.

Later, after contacting the ENCODE and modENCODE consortia, we were able to obtain EndNote [4] libraries detailing the publications related to each one. This was better because it contained more up-to-date information than their online webpages, which are not updated too frequently. Using another script (see `downloadEndnote.py`) I was able to parse an export of these libraries for sets of PubMed IDs, which were again used to download the XML information that was used to build the data tables.

2.2 Determining Consortium Members

We had hoped to obtain a membership list directly from the consortia themselves, but unfortunately this was not possible, so we were needed to classify the members ourselves.
Initially, we had separated the publications into “internal” papers directly funded by the consortium and “external” papers using data from the consortium. We had classified the consortium members as those who have co-authored at least one of those internal papers. However, after discussing with other researchers, this was deemed a poor idea because it appeared to include too many people who were not actually members of the consortia.

Both ENCODE and modENCODE were founded based on several key papers containing hundreds of authors, so we used these authors as the basis for membership. Additionally, we also classified as members the researchers who co-authored a paper on which one of those original members was the corresponding author (see correspondingAuthors.R). Thus, the non-members would be those who have co-authored a paper related to the consortium, but do not appear on the previously defined set of members. Containing similar code to the scripts mentioned earlier, there is a script (see downloadPubmed.py) for downloading specific papers’ XML data from PubMed.

3 Results

All of subsections in this section refer to the code in encodeAuthorNetwork.R, which contains the majority of the network analysis for this project. For usage examples, please see encodeDriver.R or modDriver.R.

3.1 Fixing Author Names

There was one small issue with the author lists from PubMed XML data. On some papers, the authors would be listed with just their first and last names, but in other cases, they would also include their middle initial or their full middle names. In order to avoid having the same person have multiple nodes in the graph simply due to inconsistencies in how their names were represented on the paper, we needed to make their names more consistent.
We considered two people the same if their first and last names were the same, and there were no conflicts in their middle names. Two middle names were considered to match if either one of them is blank, if one is an initial matching the first letter of the other, or if they are the same. In the rare case where there are three authors with the same first and last name, with two having a middle name and the last without a middle name, it becomes unclear which author the last one is referring to. Because this was such a rare case, occurring only twice in the list of 10,000 authors, we simply assigned the author without a middle name to the author whose middle name came first alphabetically. If it was necessary to disambiguate this case more accurately, we could have looked at the XML data for these papers and looked at other fields, such as the affiliation of the paper, to make a better guess as to whom the author without a middle name is referring to. I created script (see verifyNames.R) to generate a mapping of original author names to “normalized” names, which consist of only a first and last name, unless a middle name is necessary.

3.2 Building the Temporal Co-authorship Networks

To generate the co-authorship networks for a given year, we select all papers published during or prior to that given year. Every author on each of these papers is a node, and the weight of the edge \((u, v)\) between these nodes is the number of papers that authors \(u\) and \(v\) have co-authored. To accomplish this, we can simply initialize an empty adjacency matrix \(M\), and for every author pair \((u, v)\) on each paper, we increment \(M[u, v]\) by one. This method works well for our datasets because ENCODE, the larger dataset we have, contains less than 10,000 authors, and the sum of edge weights is less than 10,000,000.

3.3 Identifying Brokers

There are several individuals of special interest, who have strong connectivity with both members and non-members. We identified these people as “brokers” who serve to join the consortium with
outside researchers. A person is defined as a broker if they are connected to at least one member and at least some number of non-members determined by the network size.

3.4 Network Visualization

In R, there is a package called “igraph” [5] that was specifically written for working on graphs. One of the many features of the “igraph” package is the ability to draw these graphs. Conveniently, one of the ways for igraph to generate a graph is from an adjacency matrix. In order to visualize the graph in a meaningful way, we obtained an initial layout using the Fruchterman-Reingold [5][6] algorithm, a force-directed graph drawing algorithm that seeks to draw a graph such that edges are of approximately equal length with as few crossing edges as possible.

Then, we separated the authors into four groups: member authors, member brokers, non-member brokers, and non-member authors. We modified the layout such that those four aforementioned groups would be drawn generally from the center outward. This means that the first group, the member authors, would be closer to the center, and the last group, the non-member authors, would be closer to the outside ring.

Finally, we color-coded each node based on those four groups to make it clear which group each node belonged to: member authors (yellow), member brokers (green), non-member brokers (dark-red), non-member authors (light-red). The temporal graphs for both ENCODE (Fig. 1) and modENCODE (Fig. 2) can be found in the appendix, and they show how the networks grow from year to year.

3.5 Modularity and Cluster Count

In addition to generating visualizations of the co-authorship network, we also computed the modularity score and cluster count for the member and non-member subgraphs. The modularity score measures how divided a network is into modules or communities, which are a set of nodes
that are densely connected amongst themselves, but have few connections to outside nodes. A high modularity score indicates presence of many individual communities, and a low modularity score indicates that the network is generally well-connected. The modularity was of interest to us because it estimates how connected each graph, such that a high modularity score indicates collaboration amongst small groups of scientists, perhaps individual laboratories, which a low modularity score indicates collaboration on a larger scale.

These were computed using the Walktrap \cite{5}\cite{7} algorithm for finding communities in networks. This algorithm tries to find densely connected subgraphs in a graph via random walks based on the idea that short random walks tend to stay in the same community.

The plots for ENCODE modularity (Fig. 3) and cluster count (Fig. 4) and modENCODE modularity (Fig. 5) and cluster count (Fig. 6) can be found in the appendix.

3.6 Author Connectivity Distribution

In order to see the distribution of the authors based on their connectivity to members and non-members, we generated a plot (see neighborGraph.R) based on the number of edges to member and non-member nodes for each author node for both ENCODE (Fig. 7) and modENCODE (Fig. 8) as shown in the appendix. The color-coding scheme is the same as mentioned in section 3.4. Notice that while most member nodes have strong connectivity to members and non-members have strong connectivity to non-members, the member broker nodes tend to have strong connectivity to both members and non-members, which the non-member broker nodes tend to have some connectivity to members and strong connectivity to non-members.
4 Discussion

These scientific consortia tend to start out by building a core of members, with few non-members in the early years. Initially, the members at the core are not particularly well-connected, but as the consortium members publish more papers together, the core becomes very tightly-connected. This can be seen in the plot of modularity scores for both ENCODE (Fig. 3) and modENCODE (Fig. 5). For ENCODE, in 2007, there is a sharp decrease in the network modularity score for members, suggesting that this group coalesced into a well-connected component. A similar trend can be observed in 2011 for modENCODE, when the modularity score for the member subgraph drops sharply. It is interesting to note that many of the modENCODE members are also ENCODE members.

For the non-members, the modularity score tends to remain high, largely due to many laboratories using the consortium data to author publications without collaborating much outside of their own laboratory. These can be seen in Fig. 1 and Fig. 2 as the small red clusters on the outermost ring. Both ENCODE and modENCODE non-member subgraphs experience a small drop in the modularity score in the middle of the timeline, which may result from some of the initial external laboratories finally collaborating. Since more individual laboratories publish their own papers, the modularity score for the non-members of the consortia tend to be on the rise, especially during the early years. For a large consortium like ENCODE, as the network grows from year to year, the modularity score for non-members actually begins to decrease in the later years, suggesting that the non-members are becoming more and more connected, perhaps through the brokers. For modENCODE, the modularity score for the non-members only started decreasing again in 2014, but as this network continues to grow, perhaps it will follow a similar trend as
ENCODE as the non-member groups begin collaborating more with the consortium members and amongst themselves and with.

Because it seems that the brokers tend to play a major role in connecting the consortia, to outside researchers, to foster greater collaboration amongst the scientific community, it would be beneficial for consortia members to reach out to external laboratories; likewise, it would be beneficial for outside laboratories to not only use consortium data, but also to connect with the members.

5 Future Work

In the future, we may want to observe other consortia to see if they follow similar trends. Additionally, it may be useful to compare the results of these consortia with a random networks of scientists to show that these trends do not appear as a result of randomness. Finally, it may be useful to redraw the layout for the large graphs in Fig. 1 and Fig. 2 so that it becomes easier to see the connectivity. Currently, there are many edges in the interior, making it difficult to see details, especially for the connectivity of the inner core of member author (yellow) nodes, and the member broker (green) nodes to a lesser extent. Daifeng and I are currently working on a manuscript based on this work, and we hope that it can get published sometime in the future.

6 Acknowledgements

I would like to thank Daifeng Wang and the Gerstein Lab for mentoring and supporting me on this project. I would also like to thank the ENCODE and modENCODE consortia for making their publication information available for me to use in this project. Finally, I would like to thank Professor Andrew Sherman for being my advisor on this project.
7 References


Figure 1: Temporal co-authorship networks for the ENCODE consortium. The nodes are put into four groups: member authors (yellow), member brokers (green), non-member brokers (dark-red), and non-member authors (light-red). A higher resolution version of the original figure is available in Fig_1.pdf.
Figure 2: Temporal co-authorship networks for the modENCODE consortium. The nodes are put into four groups: member authors (yellow), member brokers (green), non-member brokers (dark-red), and non-member authors (light-red). A higher resolution version of the original figure is available in Fig_2.pdf.
Figure 3: The modularity score of the ENCODE member and non-member subgraphs from the years 2004 to 2014.

Figure 4: The cluster count for the ENCODE members and non-member subgraphs from the years 2004 to 2014.
Figure 5: The modularity score of the modENCODE member and non-member subgraphs from the years 2007 to 2014.

Figure 6: The cluster count for the modENCODE members and non-member subgraphs from the years 2007 to 2014.
Figure 7: Visualization of the connectivity of each ENCODE author node based on their edges to members and non-members. The color coding scheme is the same as before: member authors (yellow), member brokers (green), non-member brokers (dark-red), and non-member authors (light-red).
Figure 8: Visualization of the connectivity of each ENCODE author node based on their edges to members and non-members. The color coding scheme is the same as before: member authors (yellow), member brokers (green), non-member brokers (dark-red), and non-member authors (light-red).