Exploring the Genetic Origins of Macular Degeneration:

Math 295 - Final Project

Nick Strayer

Introduction:

As many as 11 million people in the United States suffer from Age-related Macular Degeneration (or AMD)[1]. In people 60 years and older it is the leading cause of blindness and its prevalence is growing[1]. AMD manifests itself with extracellular growths in the eye that press the retina from behind causing a slow fading of vision[2]. A 2005 study done out of Rockefeller University in New York City sequenced the genomes of 96 sufferers of AMD and 50 controls at 116,204 sites throughout their DNA [3]. The resultant genome wide association study (GWAS) analysis yielded two single nucleotide polymorphisms (SNPs) of interest: *rs380390* and *rs1329428*. The paper went on to be published in Science and has been hailed as a landmark effort in AMD research.

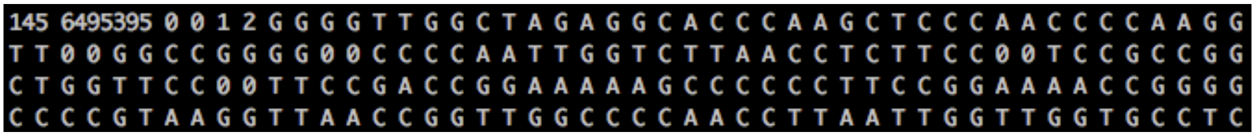
I received the raw data from that publication and went to see if there were any more insights hidden within it. This is an exploratory analysis of that dataset.

Procedure/ methods:

I received the data in the form of a .map and .ped file. I got the inspiration to do this project from taking the class Statistical Genetics with Professor Richard Single this semester so I was familiar with the wrangling of genetic data, but had never actually had to extract it from the raw output files from a sequencer. As statistical genetics is a rather small field and almost all of the sequencing technologies spit out proprietary file formats the process of simply getting the data loaded and cleaned was an adventure.

Opening the files:

The files were too big to open in a standard text editor so I went to the reliable terminal and cat-ed the .ped file. Immediately I realized my mistake, it just kept on spooling text. After exiting the command I was left with output like this…



```
145 6495395 0 0 1 2 G G G G T T G G C T A G A G G C A C C C A A G C T C C C A A C C C C A A G G
T T 0 0 G G C C G G G G 0 0 C C C C A A T T G G T C T T A A C C T C T T C C 0 0 T C C G C C G G
C T G G T T C C 0 0 T T C C G A C C G G A A A A A G C C C C C C T T C C G G A A A A C C G G G G
C C C C G T A A G G T T A A C C G G T T G G C C C C A A C C T T A A T T G G T T G G T G C C T C
```

There was no end in sight to the line so it became clear that of the format must be 146 (individuals) rows by 232,408 (alleles (each letter of a two letter genotype)) columns:

Person	**Allele 1**	**Allele 2**	…
ID#1	A	C	…
ID#2	T	G	…
…			

Inspection of the .map file showed that it had a row for every SNP with columns containing the name, location on the genome and chromosome.

Using this knowledge I attempted to load it into R and was again immediately aware of my mistake. Simply loading the first file took well over two minutes. Consulting the paper I found that due to similar problems they primarily analyzed the first chromosome (or region of the DNA strand). This cuts the number of SNPs down to ~nine thousand. (They found SNPs on chromosome one much more significant.)

A search for how to chop up the data lead me to Plink[4], a project run out of Harvard for dealing with raw genetics data. After downloading the binary and putting it into my directory a quick command of

```
plink —file amd --recode --out amd_chr1.out --noweb --chr 1
```

left me with a 5.4 mb text file rather than a 67.4 mb file. Progress. Now loading it in and cleaning could commence.

Some optimization notes:

In dealing with large amounts of data I ended up having to do a lot of optimization. For example, finding and replacing every element in a data frame, in this case replacing TRUEs with “T”s using the standard call took more than nine minutes on my laptop. Switching to doing it by converting to a matrix and back took 45 seconds.(See cleaning.r lines 18-43 and)

A problem that seems to plague lot’s of genetics data is the order of genotypes. For example GC is the same thing as CG but the computer know that. According to Professor Single this is something that has to be dealt with quite often and will frequently make it all the way to review before it is caught. To deal with this I wrote a function that sorts all of the genotype strings in alphabetical order, thus eliminating this problem (cleaning.r lines 68-72). Doing this sorting in the initial cleaning saves huge amounts of runtime over doing it in later analysis.

Over the course of re-running the scripts just a few times these optimizations became well worth it.

Cleaning conclusion:

The end of the cleaning script outputs a cleaned csv file with the columns:

ID	Sex	Phenotype (case or cont.)	SNP1	SNP2	…

This form was assumed for the further analysis scripts.

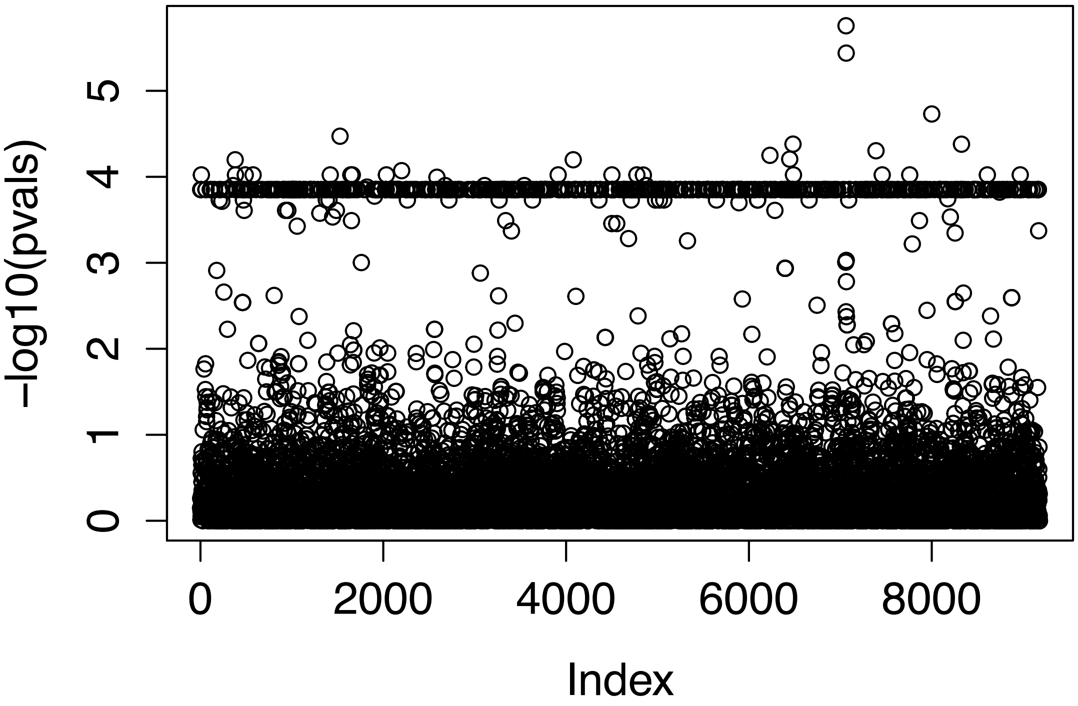
I would estimate that the cleaning took around half of the time of the whole project. I find this experience very valuable and to be expected in my first time reading genetic data.

Analysis:

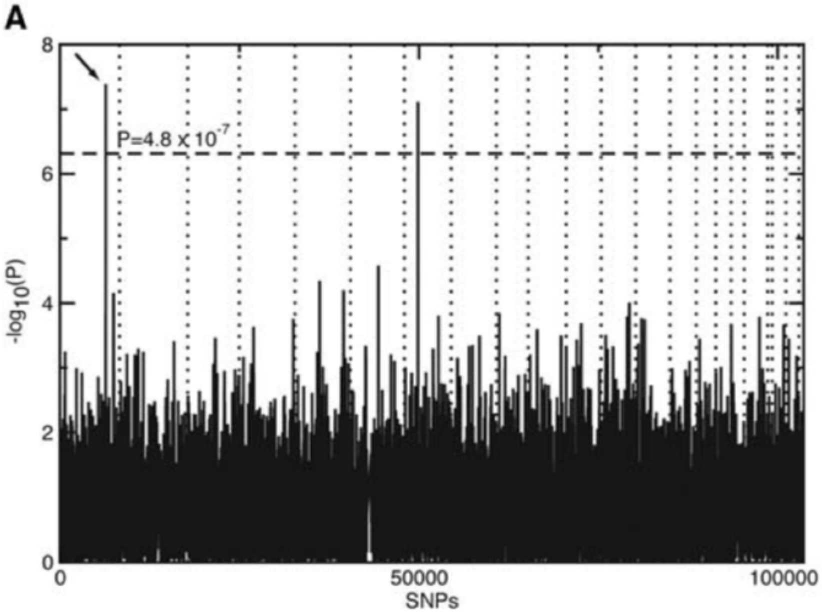
The big tool for GWASs is the manhattan plot. The manhattan plot is a visualization of the association with the trait being investigated by SNP. The name comes from it's resemblance to a city skyline. The actual metric for association varies but in the case of my analysis is a chi-squared test on a table of the form:

Genotype	Case	Control
Genotype 1	a	b
Genotype 2	c	d
Genotype 3	e	f

Once P-values have been computed for all SNPs the negative log of the P-values are plotted in a simple scatterplot. Running the data through the basic plot function in R returns this (what is that line?):



This is what the papers plot looks like (albeit with slightly different p-values as they used a different test for association[1]:



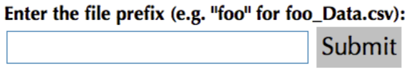
These two plots look rather different to me. The SNPs that they found most significant were rs380390 and rs1329428, there is no real easy way to figure out what my two most significant SNPs are or where significance even lies on the axis (I am using a conservative bonferroni correction). It would be entirely possible to write a few loops to pull out this information, but given the size of this data, every time running a loop is very costly, it would be nice to be able to investigate what is going on inside of the visualization.

Turns out R is not so great for this so I decided to make my own tool.

D3.js interactive manhattan plot visualizer:

Interactive and animated visualizations made in the javascript library D3 have primarily been used as explanatory visualizations. This is because the medium of the web lends itself well to getting messages across to a wide audience by hosting the visualization somewhere and allowing people to explore the carefully manicured data you are displaying. I believe, however, that there are some very promising applications for exploratory visualization using interactivity and animations.

In the case of this data I constructed a web application that sits on a locally hosted server. My R script spits out csv files that contains the results of running the association chi-squared test over the SNPs. The user then types in the file name into a selector box at the top of the visualization:



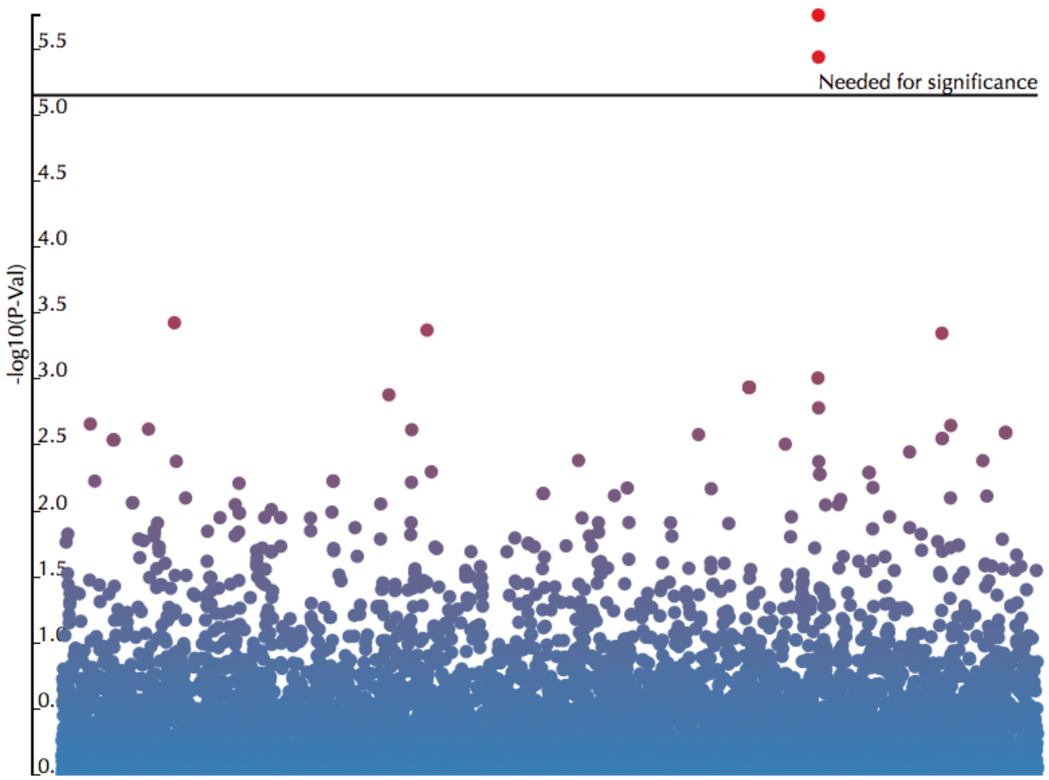
Enter the file prefix (e.g. "foo" for foo_Data.csv):

[1] It is interesting to note that they used a bar chart as opposed to a scatterplot. This does help further the manhattan analogy but representing a value such as a p-value, let alone an inverse, using a bar is a questionable visualization practice as bars are supposed to represent collections or amounts.

Upon pressing enter the manhattan plot is generated onto an SVG canvas. The user can mouse over the individual SNPs to find their name and p-value instantly, without having to write loops. In addition, an automatically generated and scaling line to show the threshold for significance using conservative bonferroni correction is drawn. Having done GWAS analysis before this massively speeds up the process of familiarizing oneself with the data and figuring out problems. (All further plots are generated using this visualizer.)

Identifying the artifact:

What is that line in our manhattan plot? Putting the data into the d3 visualizer draws the same plot. Then by mousing over some SNPs in the line and looking them up (analysis.r 33-34) we can quickly see that the trend causing that line is mono-genotypic SNPs, that is, SNPs with only one genotype. Writing a loop that removes all of these SNPs fixes the line:

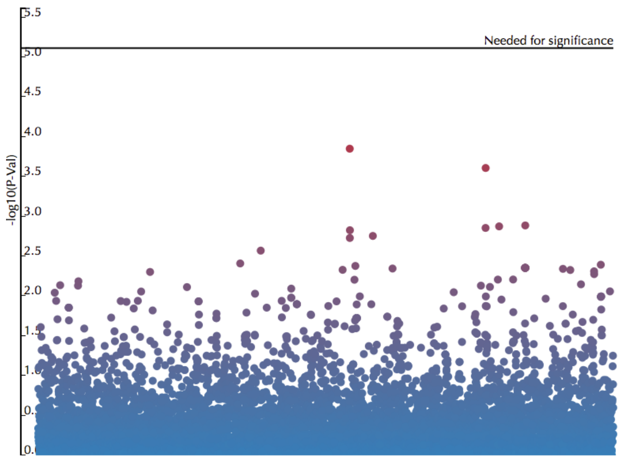


Now that we have the artifact taken care of we can focus on the two lonely SNPs at the top. We find that they are rs380390 and rs1329428, the ones that the paper found most significant. In addition their p-values of 1.7517e-6 and 3.6434e-6 respectively also match those found by the paper in their chi-squared test.

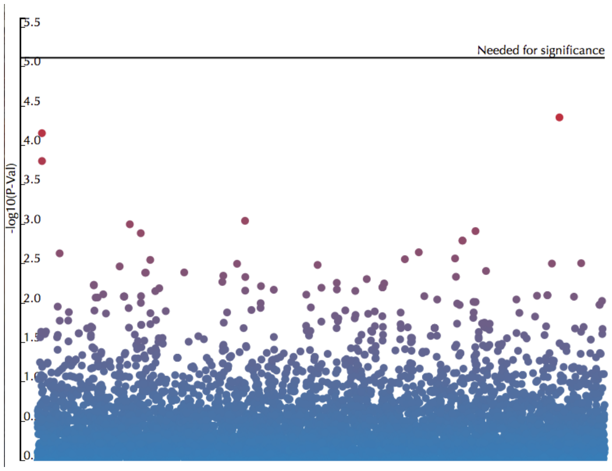
Going further:

Genes have been known to, even on non-sex chromosomes, manifest differently in different genders (e.g. [5]). Due to this, I decided to see if what would happen if I subsetted by gender:

Males



Females



Cursory interaction reveals that there is a good amount of difference. Nothing reaches the bonferroni level of significance, but it is noteworthy that we are using a conservative correction.

In order to better get a grasp on the actual dynamics of the male and female differences it would be nice to see a transition between these two graphs to see how many of the SNPs are highly associated between both of them. The problem is rendering nine thousand plus circles in a browser is rather tough, but animating them is even harder. Because of this I wrote a small section of R code to cut down the output files for the male and female subsets to just the 100 most significant SNPs (analysis.r 100-116). In addition I added a condition to the visualization web app that makes it show animated transitions between smaller files (d3Viz/script.js 73-76).

After outputting the files and loading them into the visualization it is clear to see that just a small handful of SNPs are highly correlated in *both* male and female SNPs. This large difference indicates that the matter of genetic roots may very well be deeper than the initial all-sexes results would imply.

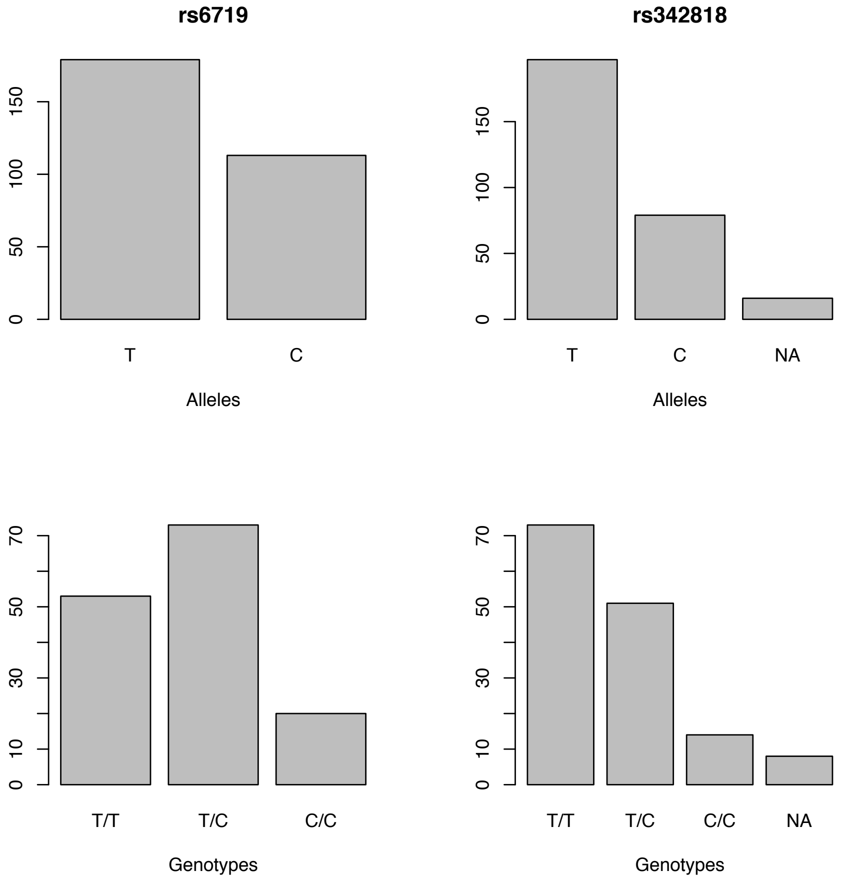
Investigating interesting SNPs:

Choosing highest significance SNPs from each of the three conditions (all, males, females) we proceed with these seven SNPs:

Label	SNP	From condition
SNP1	rs380390	All
SNP2	rs1329428	All
SNP3	rs6719	Males
SNP4	rs1329428	Males
SNP5	rs342818	Females
SNP6	rs1418632	Females
SNP7	rs4845835	Females

Distributions:

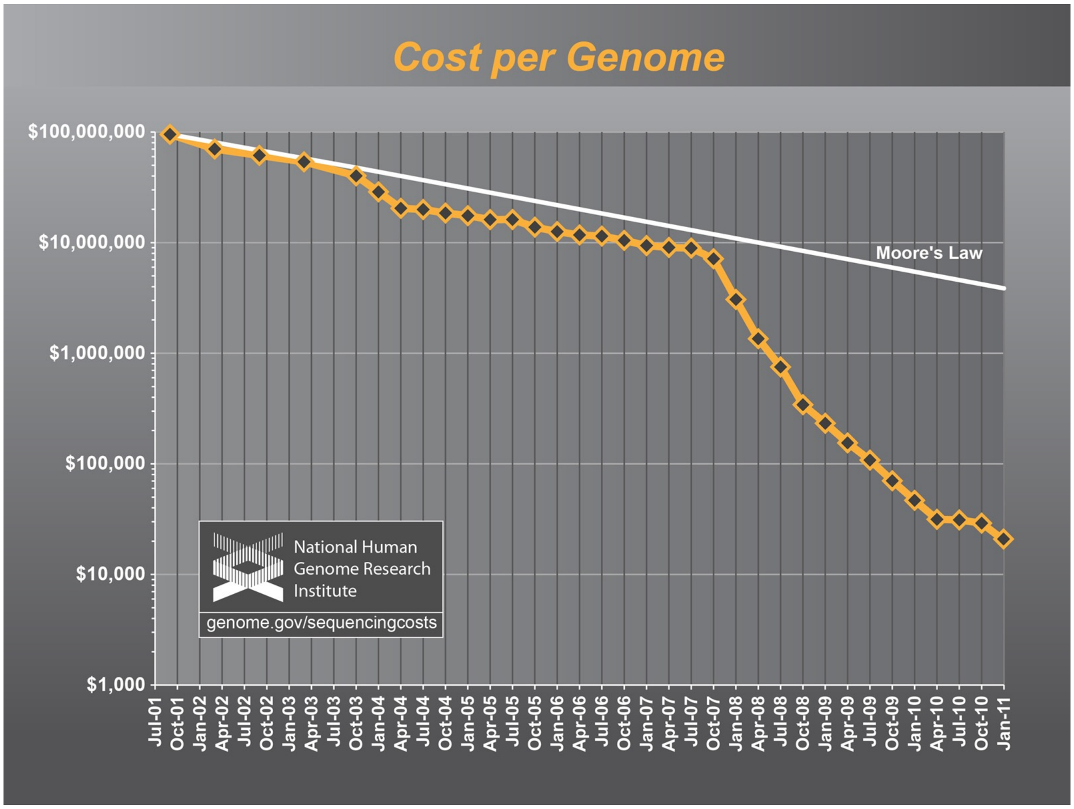
Quickly we will take a look at the distributions for a couple of the SNPs to get a sense if anything is off with them that we didn't pickup in our cleaning:



Further plots of the other SNPs reveal similar trends. We are dealing with normal SNPs here.

A note about partial sequencing:

While this study did look at a huge number of SNPs it is still just a tiny portion of the actual number present in a human genome. The reason for this is full genome sequencing is really expensive.



As we can see from this graphic courtesy of Forbes, at the time of this study (2004) full genome sequencing cost on the order of $10,000,000 per genome. Pretty hard to get a grant at that cost for an *n* of 146. In addition, super computers are still today struggling to keep up with the influx of data. We can see from this graph that cost is falling faster than Moore's law for sequencing, and thus the influx of data from sequencing is now exceeding that of Moore's law. Computers are having a hard time keeping up. All of this adds up to settling for partial sequencing.

Linkage Disequilibrium:

A very convenient feature about our genomes that makes partial sequencing a little more acceptable is the fact that genes close to each other are more likely to be transmitted with each other, i.e. a mutation occurs in a SNP an equivalent mutation occurs in nearby SNPs. The phenomena of these linked SNPs is called Linkage Disequilibrium (or LD). The math behind LD is somewhat involved and involves the construction of a table of the different combinations and their frequencies (see [6] for details). Luckily for me there is the R library Genetics for just this purpose. Multiple statistics can be used to report LD but the one what we will use (and most common) is r^2, which is simply the chi-squared statistic of the table of combinations divided by the number of samples:

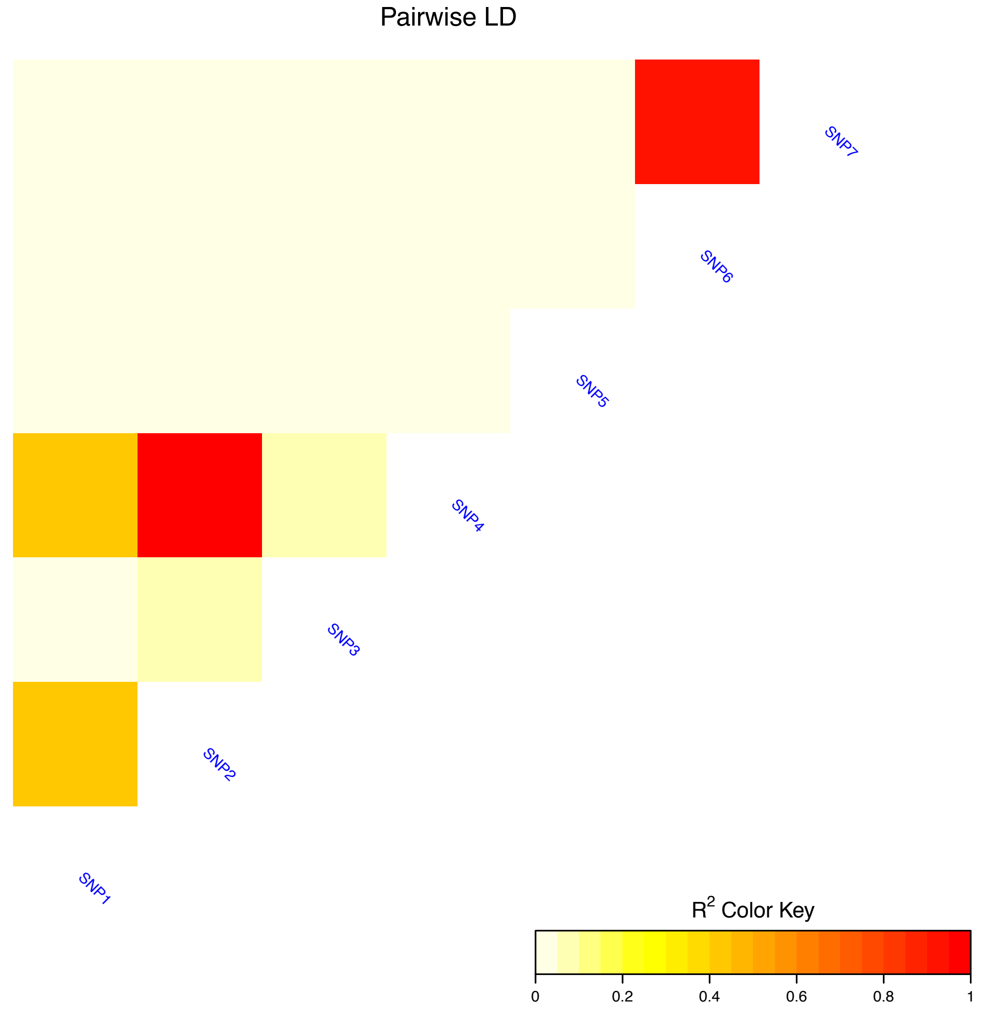
$$r^2 = D^2 / p_A p_B p_a p_b \;=\; \chi_1{}^2 / N \quad ,$$

$$D = p_{AB} \;-\; p_A p_B$$

LD for our SNPs:

We note that in our results that a lot of times the more significant SNPs are clustered near each other. This may indicate that they are in high LD and thus perhaps are both actually responding to the real causal SNP in their neighborhood that wasn't typed for.

Taking our SNPs of interest and plotting their LDs when calculated against each other in the form of a heat map we get: (The different SNP's are labeled as noted in the prior table.) (See LD_analysis.r.)



From this plot it is clear that the two female SNPs are highly linked. In addition, 2 and 4 are as well. This is interesting as they came from different groups (all and male respectively), this could imply a more complex region of association based upon gender or other un-reported stratifications. The other SNPs all seem rather innocuous in their linkage. This means that they are either a SNP of importance or are linked to one of importance.

Conclusion:

The Science paper from which these data are from is very well respected. The authors are very thorough with their analysis. My investigation of the data took me through the tedious, but important steps of cleaning raw data and replicating the results of others. The main results of this effort match those of the paper. I believe though that the paper did not fully explore the data. Simply adding stratification by gender yielded very interesting results. While the resultant tests did not meet very conservative bonferroni corrections for significance they are very close. In addition the fact that the most significant SNPs were almost always exclusive to that gender absolutely calls for more investigation. Looking at linkage disequilibrium helped shed light onto the fact that there almost certainly is a more complex relationship between the disease and genetics than the two SNPs identified by the original paper.

A follow up study, this time hypothesis driven, to test some of the more significant SNPs would be valuable (a la [7]). In addition, the creation and testing of algorithms to attempt to tease out this complex nature of association would be a massive advancement to statistical genetics and biomedical sciences.

Citations:

1. AMD facts: http://www.brightfocus.org/macular/about/understanding/facts.html
2. Mayo Clinic AMD cause: http://www.mayoclinic.org/diseases-conditions/macular-degeneration/basics/definition/con-20075882
3. Complement Factor H Polymorphism in Age-Related Macular Degeneration, R. Klein, C. Zeiss, E. Chew, J. Tsai, R. Sackler1, C. Haynes, A. Henning, J. SanGiovanni, S. Mane, S. Mayne, M. Bracken, F. Ferris, J. Ott, C. Barnstable, J. Hoh, Science 15 April 2005: Vol. 308 no. 5720 pp. 385-389, DOI: 10.1126/science.1109557
4. Plink website: http://pngu.mgh.harvard.edu/~purcell/plink/
5. Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls, *Nature* **447**, 661-678 (7 June 2007) | doi:10.1038/nature05911; Received 26 March 2007; Accepted 11 May 2007
6. Linkage disequilibrium — understanding the evolutionary past and mapping the medical future, *Nature Reviews Genetics* **9**, 477-485 (June 2008) | doi:10.1038/nrg2361
7. Common predisposition alleles for moderately common cancers: bladder cancer, A. E. Kiltie, Current Opinion in Genetics & Development, Volume 20, Issue 3, June 2010, Pages 218–224, DOI 10.1016/j.gde.2010.01.002

Sequencing cost plot: http://blogs-images.forbes.com/matthewherper/files/2011/05/cost_per_genome.jpg