Identification and analysis of candidate genes and co-expression networks involved with glucosinolates in *Brassica napus*

Y390668

March 29, 2023

Abstract

Brassica napus, commonly known as oilseed rape, is plant species of the family Brassicaceae. It is cultivated globally due to its oil-rich seed and has a wide range of uses such as in livestock feed, biofuels and drugs. Additionally, it is a model organism in many genetic studies [1]. Glucosinolates are produced as a secondary metabolite in oilseed rape and different concentrations have shown to have both beneficial and negative effects on animals. Thus, exploring the underlying genetics behind glucosinolate content is useful for future control of glucosinolates and even the potential engineering of varieties with desired levels. In this report, data on gene expression and glucosinolate content were tested for significant correlations and yielded 21 associated candidate genes. Weighted gene co-expression network analysis found 4 gene clusters, 2 independently expressed genes, as well as a highly co-expressed gene of interest. Further sequence analysis matched this gene to a resembling gene in *Arabidopsis thaliana* that encoded a MYB transcription factor.

1 Introduction

Glucosinolates (GLS) have a dose-dependent effect on the consumer. Some of its hydrolysis products are toxic whilst others have protective effects in animals [2]. GLS content shows a diverse range in different plant lines of *B. napus*. There are several genetic components within biological pathways that are essential to study. These are the genes directly involved in biosynthesis and other genes that positively or negatively regulate these synthesis genes. In the upcoming sections, this study will aim to find these key components and understand the genetics involved in the biological pathway for GLS.

2 Results and discussion

2.1 Candidate genes

The first step with the given data was to determine the pool of candidate genes which were involved with the plant line's glucosinolate content. To do this, RStudio was used to calculate linear models of the relationship between the gene expression and trait levels (Figure 1). Performing this for every gene, each linear model was then statistically tested using ANOVA and using a p-value of 0.0001, and 21 significantly associated candidate genes were selected under this condition. This specific p-value was chosen such that less than a single occurrence of candidate gene due to a type I error was to be expected. Following this observation, the next

step was to find any negatively associated genes and gene clusters that would add complexity to the system.

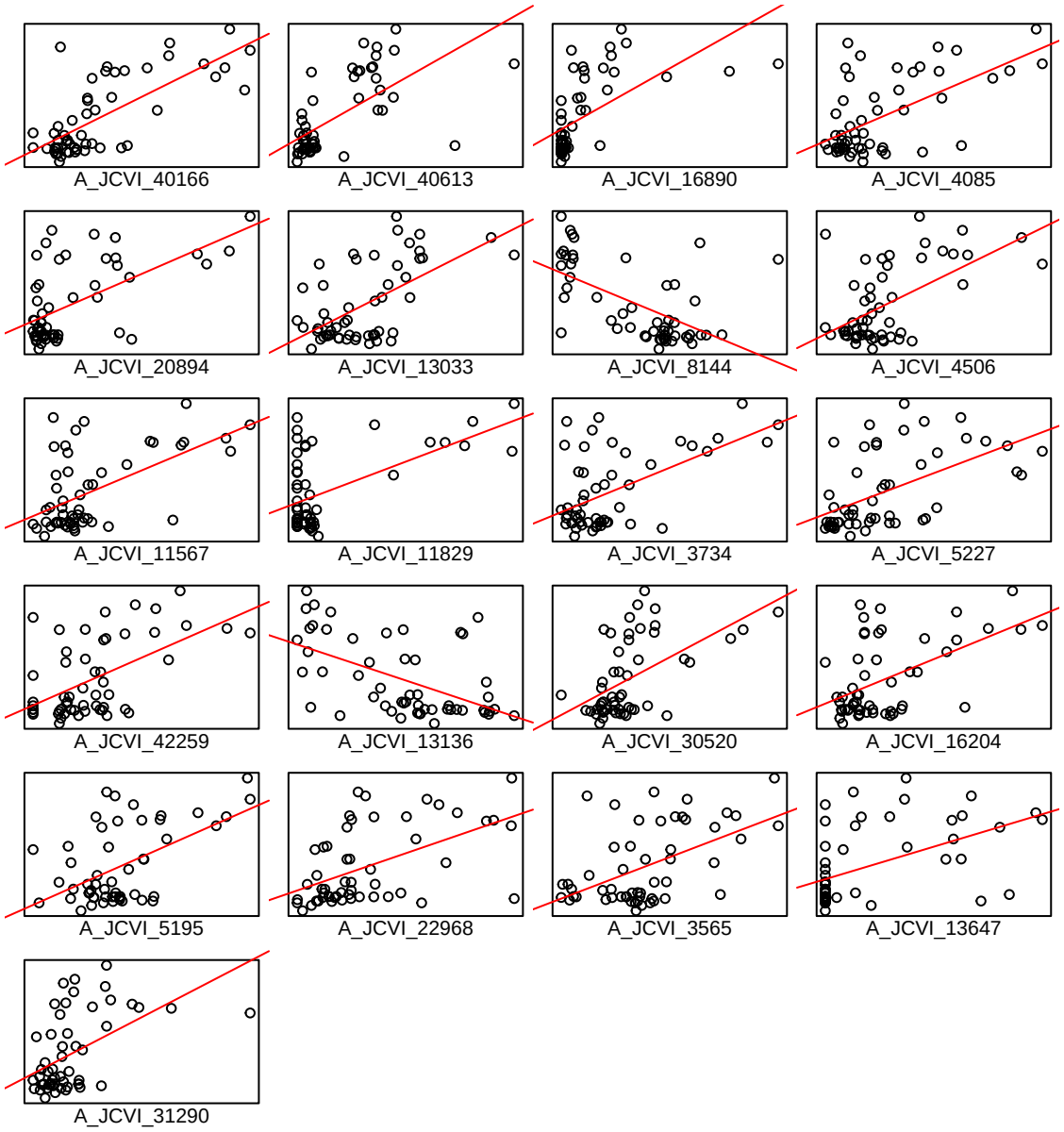

Figure 1: Scatter plots and fitted linear models of candidate gene expression in RPKM (x-axis) and glucosinolate content in % (y-axis). Two genes, A_JCVI_13136 A_JCVI_8144, can be seen with negative correlation to the trait.

2.1.1 Negatively-associated genes

Because of the continuous phenotype of complex traits, there are certain genes that will reduce the phenotype. For simplicity these genes will be referred to as “suppressors” in this report. Out of the pool of 21 candidates, 2 suppressors, showed a negative correlation with the trait, detailing that expression of these suppressors in certain plant lines was linked to lower glucosinolate content. The mechanism behind lowering glucosinolate content could be by transcription regulation and preventing expression of candidates; or another mechanism interfering

post-transcription. The type of suppressor will be analysed in the upcoming sections.

2.2 Co-expression Networks and Clusters

Gene clusters are sets of genes that are co-expressed to serve a particular function. By performing weighted gene co-expression network analysis (WGCNA) [3] the gene clusters in this system can be identified. This involves first calculating the co-expression correlation matrix which then is conditioned into an adjacency matrix under a certain threshold, 0.5 as the soft threshold in this case, and the adjacency matrix can be analysed as a network. With the igraph package in RStudio, functions to obtain and analyse the correlation matrix, adjacency matrix were used along with visualisation. From analysis, 4 gene clusters as well as 2 independently expressed genes were discovered within the set of candidates. Looking at Figure 2, it is observed that gene 2, A_JCVI_40613, has the most connections as it is the largest node. Then because it is co-expressed with many other candidates, this naturally led to the hypothesis of the gene's involvement in regulation. I.e. codes for a transcription factor that stimulates expression of other candidates. The data also shows that the suppressors are connected to multiple candidates which could also imply possible involvement in transcription regulation. Furthermore, the suppressors are highly correlated with one another which could suggest that they are paralogous genes arising from a gene duplication. In fact, it is useful to know for the rest of the candidates whether co-expressed gene clusters had formed due to either paralogous genes or functionally related non-paralogous genes. However, it was confusing trying to differentiate the two types with the given data, and sequence analysis in the next section also provided inconclusive.

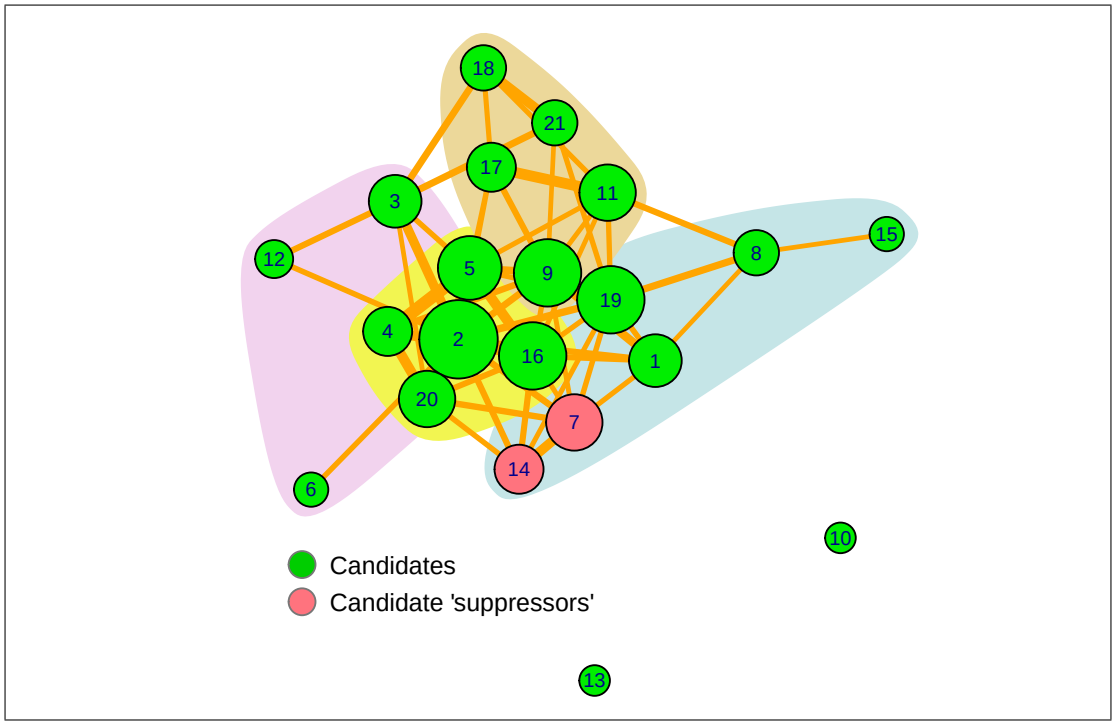

Figure 2: Co-expression network of candidate genes using a soft threshold of 0.5. Node sizes are a function of node connections and edge widths a function of correlation weight. Plotted using igraph package on RStudio[4][5][6]

2.3 Comparative genomics

So far, we have determined the genes and clusters are involved but there is still little knowledge of their function within the biological pathway for controlling GLS content. In order to better understand their functions, comparison of sequences of candidates with another closely related species was performed. *Arabidopsis thaliana*, of Brassicaceae, is a model plant used in many genetic studies and thus the functions of many genes has been already learned. By matching candidate genes to similar genes in *A. thaliana* their involvement in the GLS pathway can be determined.

Candidate queries	*A. thaliana* match	Description	E.value
A_JCVI_40613	MYB28	transcription factor	2e-102
A_JCVI_13136	PEX14	involved in peroxisome assembly	1.99e-68
A_JCVI_5195	CPK12	calmodulin-like domain calcium dependend protein kinase	2e-165
A_JCVI_31290	UGT74B1	Encodes a UDP-glucosyltransferase involved in glucosinolate biosynthesis	2e-157
A_JCVI_11567	PDR2	A member of the eukaryotic type V subfamily (P5) of P-type ATPase cation pumps	0
A_JCVI_16890	BAT5	Required for the biosynthesis of methionine-derived glucosinolates. Involved in the transport of 2-keto acids between chloroplasts and the cytosol.	6e-126
A_JCVI_5227	FMO GS-OX1	belongs to the flavin-monooxygenase (FMO) family, encodes a glucosinolate S-oxygenase	3e-133
A_JCVI_13033	UMAMIT41	nodulin MtN21-like transporter family protein	9e-151
A_JCVI_4506	AT5G25770	alpha/beta-Hydrolases superfamily protein	0

Figure 3: Table of BLAST results. Descriptions are based on summaries fromthe RefSeq database (NCBI)[7]

The given sequence data used included the sequences of 16 put of 21 candidates which included genes of interest: A_JCVI_40613 and suppressor A_JCVI_13136. These sequences were formatted in a .fasta file and using blastx, BLAST [8], the nucleotide sequences were translated into protein sequences and queried for similar proteins in *A. thaliana*. A_JCVI_13136 was mapped to the PEX14 gene on chromosome 5. In *A. thaliana*, PEX14 encodes for peroxin proteins that assemble peroxisomes. It is reported that peroxisomes are involved with the PEN2 glucosinolate metabolism pathway which produces glycides that aid in the plants defences [9] though no literature was found directly linking PEX14 directly to glucosinolate content. From this finding it is implied that the suppressor reduces GLS after transcription in the biodegradation pathway rather than negative regulation. Could this mean that the blue cluster from Figure 2 is involved in GLS pathway for plant defense? For the analysis of GOI A_JCVI_40613, the translated nucleotide sequence showed a significantly close alignment to MYB28, a transcription factor involved with the positive regulation of aliphatic GLS[10]. This supported the hypothesis that A_JCVI_40613 is involved in transcription regulation and explains why the gene was

so highly co-expressed with the other candidates. It was difficult trying to determine whether genes were paralogs. Results from the nucleotide BLAST search mapped every candidate to a different loci in *A. thaliana* meaning any gene duplication events that produced the candidate genes would have occurred before the divergence between *A. thaliana* and *B. napus*.

3 Conclusions

To conclude this report, candidate gene, co-expression network and sequence analysis when used together can be a powerful tool in understanding the pathway behind a complex trait. In this study, just from trait data and gene expression data from different plant lines, along with gene sequence data, a lot was uncovered about the underlying GLS pathway in *B. napus*. This included: candidate genes and negatively associated suppressors; the co-expression clusters and independent genes; and also the possible functions of these genes in the pathway. Though more analysis could use locus data of the genes to determine any linkage, this study alone could provide useful insight as to modify GLS content in *B. napus* variants, and techniques used can be and are also applied in analysis of other traits for other organisms.

References

[1] Kun Lu, Lijuan Wei, Xiaolong Li, Yuntong Wang, Jian Wu, Miao Liu, Chao Zhang, Zhiyou Chen, Zhongchun Xiao, Hongju Jian, Feng Cheng, Kai Zhang, Hai Du, Xinchao Cheng, Cunming Qu, Wei Qian, Liezhao Liu, Rui Wang, Qingyuan Zou, Jiamin Ying, Xingfu Xu, Jiaqing Mei, Ying Liang, You-Rong Chai, Zhanglin Tang, Huafang Wan, Yu Ni, Yajun He, Na Lin, Yonghai Fan, Wei Sun, Nan-Nan Li, Gang Zhou, Hongkun Zheng, Xiaowu Wang, Andrew H. Paterson, and Jiana Li. Whole-genome resequencing reveals Brassica napus origin and genetic loci involved in its improvement. *Nature Communications*, 10(1):1154, March 2019.

[2] Nieves Baenas, M. Elena Cartea, Diego A. Moreno, María Tortosa, and Marta Francisco. Chapter 6 - processing and cooking effects on glucosinolates and their derivatives. In Charis M. Galanakis, editor, *Glucosinolates: Properties, Recovery, and Applications*, pages 181–212. Academic Press, 2020.

[3] Leah I. Elizondo, Paymaan Jafar-Nejad, J. Marietta Clewing, and Cornelius F. Boerkoel. Gene clusters, molecular evolution and disease: a speculation. *Current Genomics*, 10(1):64–75, March 2009.

[4] RStudio Team. *RStudio: Integrated Development Environment for R*. RStudio, PBC., Boston, MA, 2020.

[5] Gabor Csardi and Tamas Nepusz. The igraph software package for complex network research. *InterJournal*, Complex Systems:1695, 2006.

[6] Katherine Ognyanova. Network visualization with R. *Kateto*, June 2021.

[7] Nuala A. O'Leary, Mathew W. Wright, J. Rodney Brister, Stacy Ciufo, Diana Haddad, Rich McVeigh, Bhanu Rajput, Barbara Robbertse, Brian Smith-White, Danso Ako-Adjei, Alexander Astashyn, Azat Badretdin, Yiming Bao, Olga Blinkova, Vyacheslav Brover, Vyacheslav Chetvernin, Jinna Choi, Eric Cox, Olga Ermolaeva, Catherine M. Farrell, Tamara Goldfarb, Tripti Gupta, Daniel Haft, Eneida Hatcher, Wratko Hlavina, Vinita S.

Joardar, Vamsi K. Kodali, Wenjun Li, Donna Maglott, Patrick Masterson, Kelly M. McGarvey, Michael R. Murphy, Kathleen O'Neill, Shashikant Pujar, Sanjida H. Rangwala, Daniel Rausch, Lillian D. Riddick, Conrad Schoch, Andrei Shkeda, Susan S. Storz, Hanzhen Sun, Francoise Thibaud-Nissen, Igor Tolstoy, Raymond E. Tully, Anjana R. Vatsan, Craig Wallin, David Webb, Wendy Wu, Melissa J. Landrum, Avi Kimchi, Tatiana Tatusova, Michael DiCuccio, Paul Kitts, Terence D. Murphy, and Kim D. Pruitt. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Research*, 44(D1):D733–745, January 2016.

[8] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, October 1990.

[9] Tong Su, Wenjing Li, Pingping Wang, and Changle Ma. Dynamics of peroxisome homeostasis and its role in stress response and signaling in plants. 10:705, February 2019.

[10] Christian Dubos, Ralf Stracke, Erich Grotewold, Bernd Weisshaar, Cathie Martin, and Loïc Lepiniec. MYB transcription factors in Arabidopsis. *Trends in Plant Science*, 15(10):573–581, 2010.

Supplementary methods

BrassicaData.R

```
#### SECTION 1 #### Setup and importing

## 1.1 Initial Setup
rm(list=ls()) # Clear image environment
setwd("M:/w2k/BigDataBio/Brassica Report")  # set working directory

## 1.2 Importing all the datasets
# Full Brassica napus RPKM #
raw_rpkm <- read.table(url("https://learn-eu-cen...)) # from URL on VLE
write.table(raw_RPKM, "raw_rpkm.txt") # make local copy .txt file
raw_rpkm <- read.table("raw_rpkm.txt")
# Brassica napus trait data #
raw_trait <- read.table("Glucosinolates.txt") # text was copy pasted from URL
 ↪  onto .txt

#### SECTION 2 #### Tidying datasets

## 2.1 Cleaning RPKM data ##
# Manipulation #
rpkm <- raw_rpkm[-1,-1] # remove row and column names
rpkm <- as.data.frame(lapply(rpkm, as.numeric)) # converts all elements to
 ↪  numeric
# Setting row and column names #
geneNames <- raw_rpkm[-1,1] # char vector of gene names
plantLines <- raw_rpkm[1,-1] g# char vector of plant lines
rownames(rpkm) <- geneNames
colnames(rpkm) <- plantLines

## 2.2 Cleaning Trait data ##
# Manipulation #
trait <- as.data.frame(as.numeric(raw_trait[-1,-1]))
# Setting names #
rownames(trait) <- raw_trait[-1,-1]
colnames(trait <- raw_trait[1,-1]

## 2.3 Merging RPKM and Trait data ##
merge.trait <- merge(t(RPKM), trait, by="row.names") # merging data
rownames(merge.trait) <- merge.trait[,1]
merge.trait <- merge.trait[,-1] # remove first col

#### SECTION 3 #### Saving variables

save(rpkm, merge.trait, geneNames, plantLines, file="brassicaData.rda")
```

CandidatesAnalysis.R

```
#### SECTION 1 #### Setup and importing

rm(list=ls())
```

```r
setwd("M:/w2k/BigDataBio/Brassica Report")
load("brassicaData.rda") # loading tidied datasets

#### SECTION 2#### Finding Candidate Associated Genes

## 2.1 List of linear models of correlation between genes and trait ##
lmlist <- lapply(geneNames, function(gene)
      lm(merge.trait$Trait~merge.trait[,gene])
    )
names(lmlist) <- geneNames

## 2.2 List of ANOVA results of linear models ##
anovalist <- lapply(lmlist, function(gene)
  anova(gene)
)
names(anovalist) <- geneNames

## 2.3 Creating a results df ##
results.trait <- as.data.frame(matrix(nrow = 0, ncol = 8))
colnames(results.trait) <- c("Df", "Sum.Sq", "Mean.Sq", "F.value",
↪  "P.value", "R2", "Intercept", "Gradient")
for (gene in 1:length(geneNames)){
  anova <- as.data.frame((anovalist[[gene]])[1,])
  intercept <- as.data.frame(coefficients(lmlist[[gene]]))[1,1]
  gradient <- as.data.frame(coefficients(lmlist[[gene]]))[2,1]
  R2 <- as.data.frame(summary(lmlist[[gene]])$r.squared)
  nextrow <- as.data.frame(c(anova, R2, intercept, gradient))
  colnames(nextrow) <- colnames(results.trait)
  results.trait <- rbind(results.trait, nextrow)
} # for loop construction of results df
rownames(results.trait) <- geneNames
results.trait <- results.trait[-which(is.na(results.trait
↪  [,"Gradient"])==T),] # removing unexpressed genes
results.trait <- results.trait[,-1] # all results are 1 dof

## 2.4 Determining significantly associated genes ##
results.trait <- results.trait[order(results.trait$P.value, decreasing =
↪  F),] # order in ascending p value
pval <- 0.0001
results.trait <- subset(results.trait, P.value<=pval)
candidates <- rownames(results.trait) # 21 candidate genes

## 2.5 Finding suppressors ##
suppressors <- rownames(subset(results.trait, Gradient<0)) # 2 genes
suppressor1 <- suppressors[1]
suppressor2 <- suppressors[2]

#### SECTION 3 #### Plots

```

```r
## 3.1 Candidates and trait multiplot ##
dev.new(width=5.83,height=6.2,unit="in") # scale
par(mfrow=c(6,4)) # 6x4 plot matrix
for (i in 1:21){
  plot(merge.trait[,geneNames.candidates[i]], merge.trait$Trait,
       xlab = geneNames.candidates[i], ylab = NA,
       xaxt = "n", yaxt = "n",
       mgp = c(0, 1, 0))
  abline(lm(merge.trait$Trait~merge.trait[,geneNames.candidates[i]]), col
  ↪  ="red")
} loop plots every candidate gene expressions to trait
multiplot.candidate <- recordPlot()
pdf(file="candidates_trait_multiplot.pdf") # write plot to pdf

#### SECTION 4 #### Saving

save(candidates, suppressor1, suppressor2, results.trait,
↪  file="candidateAnalysis.rda")
```

WGCNA.R

```r
#### SECTION 1 #### Setup and importing

rm(list=ls())
setwd("M:/w2k/BigDataBio/Brassica Report")
load("brassicaData.rda")
load("candidateAnalysis.rda")
library(igraph) # for constructing and plotting networks

### SECTION 2 ### Weighted Gene Co-expression Network Analysis

## 2.1 Calculating correlation matrix on co-expression of candidates ##
covariance <- cor(t(rpkm[candidates,]), method="pearson")
correlation <- cov2cor(covariance) # scales to Pearson's correlation

## 2.2 Constructing adjacency matrix ##
threshold <- 0.5 # soft threshold value.
adjacency <- covariance
adjacency[adjacency>-threshold & adjacency<threshold] <- 0 # sets correlations
↪  smaller than threshold to 0 (nodes unconnected)

## 2.3 Constructing co-expression network ##
network <- graph_from_adjacency_matrix(adjacency,
    diag = F, # sets diag to 0, removing self-loops
    weighted = T, # includes correlation as weight
    "undirected" # removes edge directions
)

## 2.4 Optimal co-expression clusters within candidates ##
clusters <- (cluster_optimal(network))$membership
clusters_list <- list()
```

```
for(i in 1:max(clusters))
        clusters_list <- list((which(clusters==i))) # creates a list of
    clusters (node vectors)
     # list of 6 but 2 solo genes so 4 clusters

#### SECTION 3 #### Plots

## 3.1 Setting node and edge sizes ##
V(network)$size <- 5*sqrt(degree(network)) # node size ~ number of connections
E(network)$width$ <- 3*abs(E(network)$weight$) #edge width ~ weight

## 3.2 Colors ##
V(network)$color <- rep("green2", vcount(network)) # node color
V(network)[c(suppressor1, suppressor2)]$color <- "#ff737e" # red for suppr.
E(network)$color <- rep("orange", ecount(network)) # edge color
col.clusters <- c("#c5e5e7", "#f2d3ee", "#f2f551", "#ecd89a") # cluster color

## 3.4 Plotting network graph ##
l <- layout_with_fr(network, niter=500) # Fruchterman-Reingold
plot.igraph(network, layout = b,
    vertex.label.family="Helvetica",vertex.label.cex=0.7,vertex.label=c(1:21),
    mark.group=clusters_list[1:4],mark.col=col.clusters,mark.border=NA)

#### SECTION 4 ####

save(adjacency, network, clusters, list_clusters, threshold, file="wgcna.rda")
```

SequenceAnalysis.R

```
#### SECTION 1 ####

rm(list=ls())
setwd("M:/w2k/BigDataBio/Brassica Report")
load("brassicaData.rda")
load("candidateAnalysis.rda")
library(seqinr) # for dealing with sequence data

### SECTION 2 ### Importing and tidying

## 2.1 Importing and Manipulating sequence data ##
raw_seqs <-
↪  read.delim("http://www-users.york.ac.uk/~ah1309/BigData/data/genes.txt")
seqs <- raw_seqs[,-1]
names(seqs) <- raw_seqs[,1]

## 2.2 Subset to only include candidates ##
seqs <- seqs[which(names(seqs)%in%candidates)]

## 2.3 Write to .txt and .fasta ##
write.table(seqs, "seqs.txt")
write.fasta(sapply(seqs, s2c), names=names(seqs), file.out="seqs.fasta")
```