COMPUTER SCIENCE DEPARTMENT
FACULTY OF COMPUTERS AND INFORMATION
CAIRO UNIVERSITY

PhenoGene

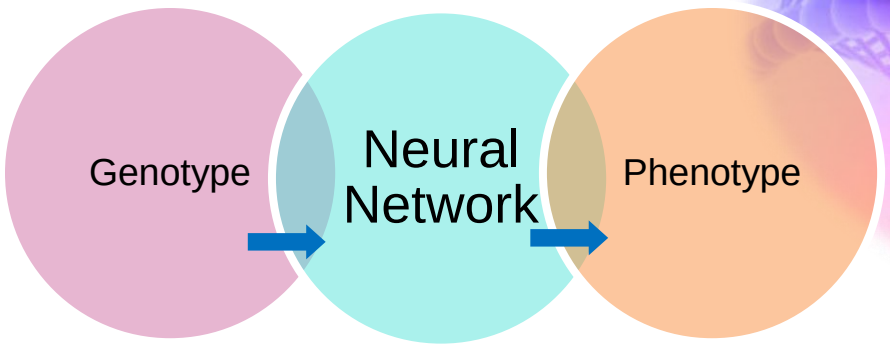
[Graduation Project]

By

Rehab Sabry Ahmed 20080403
Salma Shehab Raouf 20090152

Prof. Hesham Hassan

TABLE OF CONTENTS



CHAPTER 1|INTRODUCTION

1.1 ABSTRACT

MOTIVATION

Prediction of complex phenotypes using genotypes is the central concept and recent interest in the advocated fields of research.

One of the implications was the public wonder if it's actually possible to predict a human phenotypic trait given only his genome DNA sequence/ genotype.

APPROACH

We're to build software that makes use of the Neural Networks capabilities and potentials.

The network is trained on a given database of DNA SNPs variants of a set of individuals having a specific trait described. Then the trained network is used to predict the individuals' trait.

We've specifically tested the eye color trait, analyzing a database grading a number of SNPs data markers associated to eye color variation.

PRODUCT

A computer-software that will be able to:

- Predict a human's phenotypic trait having been provided by a database of SNPs variations associated specifically to that trait.
- Study the effect of each gene variation related to that trait, giving a representation of its contribution or role in the appearance of such trait, and its effect on validation of such prediction.

The application is extendible for various human phenotypes.

IMPLICATIONS

The project should affirm the research findings, as well as applying the results in different fields, such as: Forensic Science and different scientific research areas. Our application targets genetic researchers for genetic determinism purposes and further verification to their studies.

1.2 INTRODUCTION

Computer Science's power doesn't end at the automation and invention of electronic machines, or simulating the conditions of a real problem to obtain its answer; but rather extends its abilities to the Biology field, where computers are actually used to enhance our understanding of the biological nature of several organisms including us, humans.

Bioinformatics, a field that's grown widely over the past few decades due to the vast need of computer storage and data manipulation techniques aiding significant biological content, is where computers are being used to serve biologists understand the functions of different organisms as well as intervening with these functions.

New discoveries and foundations are made every day; researches are being able to constantly come up with new biological information complementing the necessity of Computer Science and Bioinformatics fields to bond.
Methods for manipulating and processing data, storing and applying algorithms in general, are being facilitated using computer machines.

In our case, for example, in Genetics field, Bioinformatics has been of great importance for organizing genomic data, sequencing, understanding and performing analysis outputting genuine results and new insights.

Understanding the human's genome and DNA sequencing and protein structure for example is a complex process in Biology, requiring the use of more complex machinery.

That is Bioinformatics' main concern. Its goal is to provide software programs developed to help with management of such complex scientific data. It has used many fields to perform so, mathematics, engineering and computational approaches mainly.

Analysis and interpretation of this data has been researchers' main concern, here comes the part of Computational Biology within bioinformatics where new software and tools are to be developed not only for the storage and querying data, but also for managing and applying algorithms and techniques for finding out new relations and further knowledge about it.

Such studies performed by researchers – geneticists – in genetics' field, one of which is a genome-wide-association study (GWAS) where researchers study the effect of gene variations defining or affecting human traits, where these examinations require the field to merge with computer.

GWA studies will be our project's main concern, the project's willing to cover a critical issue, and we are to test several genetic variations (SNPs) for significance in predicting the human's eye color specifically for further aiding and conformation to them.
Genetic determination purposes and further verification can be of great help to the application's users.

In Previous Genome Wide Association Studies, researchers have examined different genetic components of individuals and have been able to connect between one's genotype and phenotype, stating that the human eye color morphology has a strong genetic component.
This fact is of great importance, it has been their key to find out the relationship between genome encoded in DNA and hereditary information based on it.

Basic knowledge of biological concepts and terminology are necessary to grasp the idea, this will be covered in the upcoming few chapters.

Such genuine studies after have been proven correct can contribute to and assess many recognizable areas. One of the most important fields is Forensic Science, or known as Forensics.
This field, also Forensic Criminalistics, is the use of science and new technologies together for identifying or finding evidence for crimes.
The use of DNA analysis has been very critical in Forensics, it's very smart for criminals not to leave evidence but for DNA found in objects.
It has been of a great motivation indeed to be of such help to this field. This will be further more discussed in the coming chapters.

Chapter 1, as you can see, is to give you an introduction about our project's idea and a decent summing up to the most interesting facts and scientific evidences.

Chapter 2 covers the background, motivation, usages of the idea and its contribution to DNA Forensics and research.

The backgrounds will cover the fields used in the project, the use of Genetics, Biology and these sciences together with Computer Science, and finally Bio-Informatics.

As for Chapter 3, we have included several surveys of similar software and applications previously developed and their uses.

Chapter 4 describes the requirement analysis of the project, whereas Chapter 5 and 6 digs deeper in explaining the design and implementation phases of the project.

Chapter 7 states the conclusions, and Chapter 8 the future work to be made upon this project.

Chapter 9 speaks about the user manual and a programmer guide on how to extend this application.

At the end of this document, appendix B is provided for covering the most used and common technical terminology needed to understand about these different fields and their connections and usage as a unit, And for further explanation to them.

Also, a reference to the major studies will be provided. Besides, references to other supporting papers and researches.

CHAPTER 2|MOTIVATION & BACKGROUND

2.1 MOTIVATION

Having previously demonstrated a brief to our project's idea, here more illustration to our objective and motivation can be presented.

FORENSIC PURPOSES

DNA analysis is starting to be interesting, as well as important, for different and critical fields. One of which is forensic science, as mentioned before. Forensic science is to aid in criminalistics using trace evidence analysis and DNA forensic profiling.

For instance, genetic material of suspects left at a crime scene can answer important questions giving some details about them, taking into consideration the suspects' unique DNA, as fingerprint analysis for example.

Analyzing these genetic materials can't be performed by humans alone. Machines are needed to assess us getting more accurate results in a reasonable amount of time.

GWA studies are of great contribution to forensics. To explain, if we are able to study human's genes and genes' variants to tell about their outer structure and traits, and further predict them, placing a suspect of a crime could be a lot more maintainable.

That way, forensic sciences can strengthen their ground in their research and become more reliable and accurate using DNA-based identification and analysis, enhancing its power.

DNA markers related to eye color have been identified, making it feasible for predicting human's phenotype using genome variations, which are valuable to forensic studies.

RESEARCH-WISE

Furthermore, another one of the main motives is supporting researchers, mainly genetic researchers. This reason possesses huge interest, as the need for geneticists increase for more progress to identify new genome sequences, to study the effect of genes alleles that can lead to human morphology structure, moreover, medically, to lead to new discoveries for diagnoses and treatment, possibly earlier, of diseases.

Researchers are always working on making new impacts on studies, research and results that lead to constant advances widening the spectrum of biological and medical findings, with the help of bioinformatics.

If researchers are able to test genetic mutations for the appearance of certain traits or diseases, and get confirmed results, this is to be a great aid for their search, as well as extending their help to other profitable fields.

Our application provides potential help in this area, it allows researchers to test certain variants and study its effect on phenotypes, being provided by their database of SNPs.

Additional information about the application benefits to researchers is to be discussed profoundly in details in the following chapter.

2.2 BACKGROUND

BIOLOGICAL BACKGROUND

DNA

The human body consists of trillions of cells. Most of them contain the genetic material known as the DNA.
DNA carries genetic information of living beings, resulting in the heredity substance inherited. While a gene is the molecular unit having the information needed for organisms to function, DNA is what encodes these information. So a gene is encoded in a sequence of nucleic acids (mostly DNA).

Adding more, DNA is recorded using a sequence of nucleotides which are: guanine, adenine, thymine, and cytosine; and mostly referred using their 4 initial letters: G, A, T, and C.
So DNA, just like a binary compiled file that consists of 0's and 1's, is a compiled file consisting of these four letters.

Mainly, humans' DNA contains 3.2 billion letters.

And just like an executable binary file, the DNA is an executable biological file. The instructions encoded in the DNA strands get translated by cellular organelles called the ribosomes.

No two organisms can have the same DNA sequence, making it a genetic fingerprint-like identifier.

GENES

The DNA is being cut into smaller segments called "genes". Each gene is responsible for being translated into a specific protein. 30,000 genes have been identified so far.

A gene is the basic instruction; on the other hand, an "Allele" is one variant of that gene.

Gene variants at certain loci are responsible for traits in human being such as eye or hair color.

Genetic Researchers are in the quest of identifying more variants.

SNP

Single-nucleotide Polymorphism is a DNA variation sequence occurring within a locus in a DNA fragment and differs from another one.

Almost all common SNPs have only two alleles, as SNP is a single base-pair where a mutation at a specific locus occurs.
SNPs are used in GWA studies; with the assistance of bioinformatics, researchers have been trying to include more variations in a database of SNPs found in certain genes, which can be useful in genetic mapping, its influence on human traits and diseases.

SNP is an important concept in our application. An example of a SNP is “rs1800407” In gene OCA2 which is mainly responsible for the human eye color trait.

EYE COLOR GENETICS

Scientists have discovered that genes are responsible for eye color variations. Attempting to find all gene SNPs that influence this trait, up till now around 27 genes were found associated with that trait.

Several recent studies have been trying to address the genetics of eye color’s issue, trying to analyze several SNP data markers for determination of the trait.

Genetics of eye color is a very complex process; as GWA studies have demonstrated. Multiple genes could be involved; however, it’s been found out that the genes OCA2 and HERC2, existing at chromosome 15, are the two main genes that explain the variation in eye color.

Different SNPs within these two genes can be responsible for different eye colors. For example, gene OCA2 is said to control the blue/brown eye color spectrum.

Small changes – SNPs – that happen to genes are the main contribution to this phenotypic trait. Specific mutations in genes may be responsible for specific eye color. It takes a genuine amount of time and study to determine which.

So genes and their multiple variants interact both to determine the phenotypic trait, what we're trying to do is validate a DNA-based phenotypic prediction using these variants that are provided, in attempt to find linkage between a specific variant using genome sequence and the chosen phenotype; we've tested mainly on eye-color trait.

We've used nine different SNP variations associated to eye color for prediction. Details and conclusions are to be provided later.

BIOINFORMATICS

The science of recruiting computers to serve biologists accomplish their tasks accurately in a feasible amount of time, managing biological information in an easy and user friendly manner.

Since computers have been engaged with biology, the research productivity has escalated enormously.

Moreover, in the light of our project, predicting complex human phenotypes from genotypes has gained great interest lately given the amount of benefits to various other fields.

HGP

The first project that should be highlighted in this section is the HGP (Human Genome Project); a 13-year project that is coordinated by the U.S. Department of Energy and the National Institutes of Health. Several countries have contributed to this project. It was completed at the year 2003.

The HGP's objective was to fully sequence the humans' genome to identify its functions. Readers should be aware that humans share 99% of their DNA, so sequencing one's DNA is a huge milestone.

Identifying the different gene's functionalities is another great ongoing-achievement to be fully accomplished by the continuous research being done.

CHAPTER 3|SURVEY

Several studies were conducted and have already set the theoretical basis for predicting hair and eye color, demonstrating that highly accurate predictions are indeed feasible using only DNA samples.

The following are some projects that are built for same/similar purpose as ours.

STRANGER VISIONS

As quoted from their website:

> *"In Stranger Visions artist "Heather Dewey-Hagborg" creates portrait sculptures from analyses of genetic material collected in public places. Working with the traces strangers unwittingly leave behind, Dewey-Hagborg calls attention to the impulse toward genetic determinism and the potential for a culture of genetic surveillance."*

Heather was able to extract basic heredity traits from the genetic material, including hair and eye color, ethnicity and gender.

While a lot of guesswork had to be done to get the 3D portraits.

ANCESTARY-DNA

Speculate ones family trees' ethnicity/race using their DNA structure.

DIGITAL PHYSIOGNOMY

Quoted by the editor:

> *"Looking for a program both entertaining and enlightening? Digital Physiognomy reads a person's face and tells you about the secret character traits of that parson with remarkable accuracy. People's facial features - eyes, noses, ear lobes, chins, eye brows, cheekbones, and foreheads all carry valuable information easily revealed with this software"*

PROPHECY MASTER

It's a computer-software that employs latest technologies in facial recognition, face transformation, and mathematical transformations to age the face several years ahead.

> *"The product detects a face on any picture, analyzes its facial features, and applies sophisticated mathematical transformations to age the face 20 years."*

CHAPTER 4|REQUIREMENTS ANALYSIS

4.1 INTRODUCTION

PURPOSE

- To build software that associates an organism's genotype with its phenotype.
- To contribute an open-source software to the field of forensic sciences.
- To assist the genetic researchers, as well as interested individuals.

ISSUE STATEMENT

The DNA contains enormous information that we haven't fully understood yet, for instance, the relation between the genotype and phenotype of an individual. How, given the gene sequence of an individual, could we determine some phenotype trait of his?

OBJECTIVE

- The software should predict a desired physical trait given some chosen DNA SNP variations.
- The software should confirm a gene SNP is affecting a specific phenotype, given enough data.

4.2 PROPOSED SOLUTION

PROBLEM SCOPE

- This software is phenotype independent, i.e. the user chooses what physical features to predict.
- This software is animal-type independent, i.e. it can be used on animals other than Homo sapiens.
- This software provides further analysis to the results, like error measurements and PDF reports.

USERS OF THE SYSTEM

- The program is intended for genetics researchers.
- Interested individuals with little or no knowledge of neural networks.

4.3 FUNCTIONAL REQUIREMENTS

1- TRAIN

Given multiple user's gene data and corresponding physical feature, train the neural network to learn and produce appropriate weights.

This is necessary for first-time use of the system, in which the user doesn't have a weights file to predict the physical feature of an unknown.

Output: produce weights file that represents the relation between trained input/output data.

2- TEST

Given one or more individual's gene data, and weights file, the system will attempt to predict the physical feature to each individual.

Output: predicts with an accuracy that corresponds to that specific at the training stage.

3- ANALYSIS

After the training stage the system will provide the user with error measurements and further analysis that should help the researcher in the decision make process.

Output: PDF report of the analysis.

4.4 NON-FUNCTIONAL REQUIREMENTS

EXECUTION QUALITIES

- Security: The system is open-source, hence back-door free.
- Usability: The system is user friendly, with an advanced mode GUI for experts, hence highly usable.
- Efficiency: The system should output the result in a reasonable amount of time.

EVOLUTION QUALITIES

The output should be measured by weather the expected output is met by the actual output.

EXTENSIBILITY

- An HTML website is built to provide technical documentation to guide any interested programmer in extending the software.
- The software is open source under the GPL license; hence any one can contribute to it.

ASSUMPTIONS

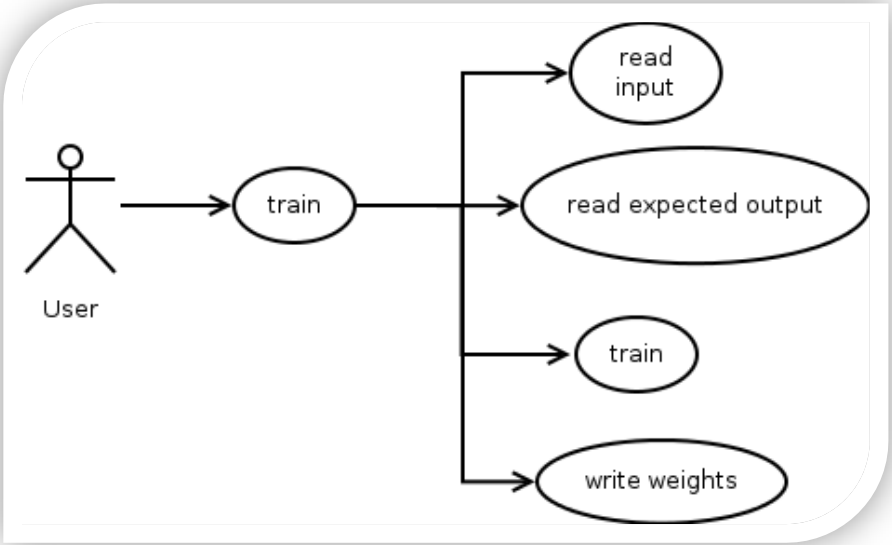
- ✓ The user has read the user manual.
- ✓ The user has fair background about genotypes and phenotypes.
- ✓ The user knows the number of input genes to train/test.
- ✓ The user knows the output keywords to be learnt.
- ✓ There's enough data to train the neural network.
- ✓ The input data follows the format specified in the user manual.
- ✓ The expected output data follows the format specified in the user manual.

MEASUREMENTS

- ✓ Cross entropy error measurement is to be used.
- ✓ Percentage of mismatch between expected output and actual output.

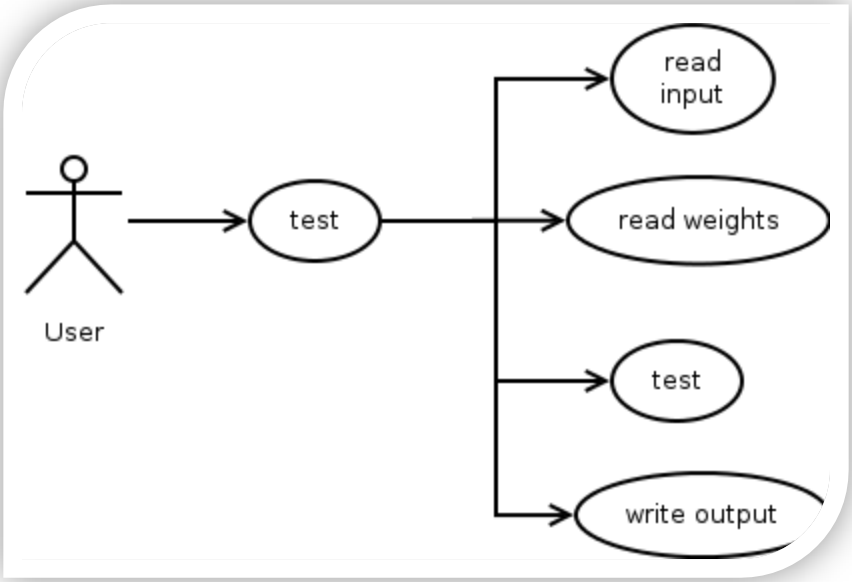
4.5 SYSTEM MODEL

USE CASE MODEL|1



Use Case ID	1
Use Case Name	Train
Actor	User
Preconditions	The user is at the “train” tab. Input file in the correct format. Expected output file in the correct format. All network's parameters are filled with valid values.
Post-conditions	Weights file produced
Flow of events	1- The user selects the input file 2. The user selects the expected output file 3. The user adjusts the network parameters to his needs 4. The user clicks the start button
Exceptions	1- Input file isn't selected 2- Expected output file isn't selected 3- Network parameters aren't complete
Includes	Read input. Read expected output. Writes weights.

USE CASE MODEL|2



Use Case ID	2
Use Case Name	Test
Actor	User
Preconditions	The user is at the “test” tab. Input file in the correct format. Weights file in the correct format. All network's parameters are filled with valid values.
Post-condition	Output file produced
Flow of events	1- The user selects the input file 2. The user selects the weights file 3. The user adjusts the network parameters to his needs 4. The user clicks the start button
Exceptions	1- Input file isn't selected 2- Weights isn't selected 3- Network parameters aren't complete
Includes	Read input. Read weights. Writes output.

SCENARIO|1

- Name: Training Scenario.
- Participating actors: User.
- Flow of events:

  - The user chooses the train tab, browses both the input and expected output text files, then makes sure all the parameters are filled. Then clicks start.
  - The progress bar increases as the program train the network, the system reports success and weights file to be found in the input's directory.
  - The analysis tab contains meaningful information.

SCENARIO|2

- Name: Missing parameters>
- Participating actors: User.
- Flow of events:

  - The user chooses the train tab, browses both the input and expected output text files. Then clicks start.
  - An error message appears stating that some neural network parameter is not filled and that it should be filled.

CHAPTER 5|DESIGN PHASE

5.1 DEVELOPMENT ENVIRONMENT

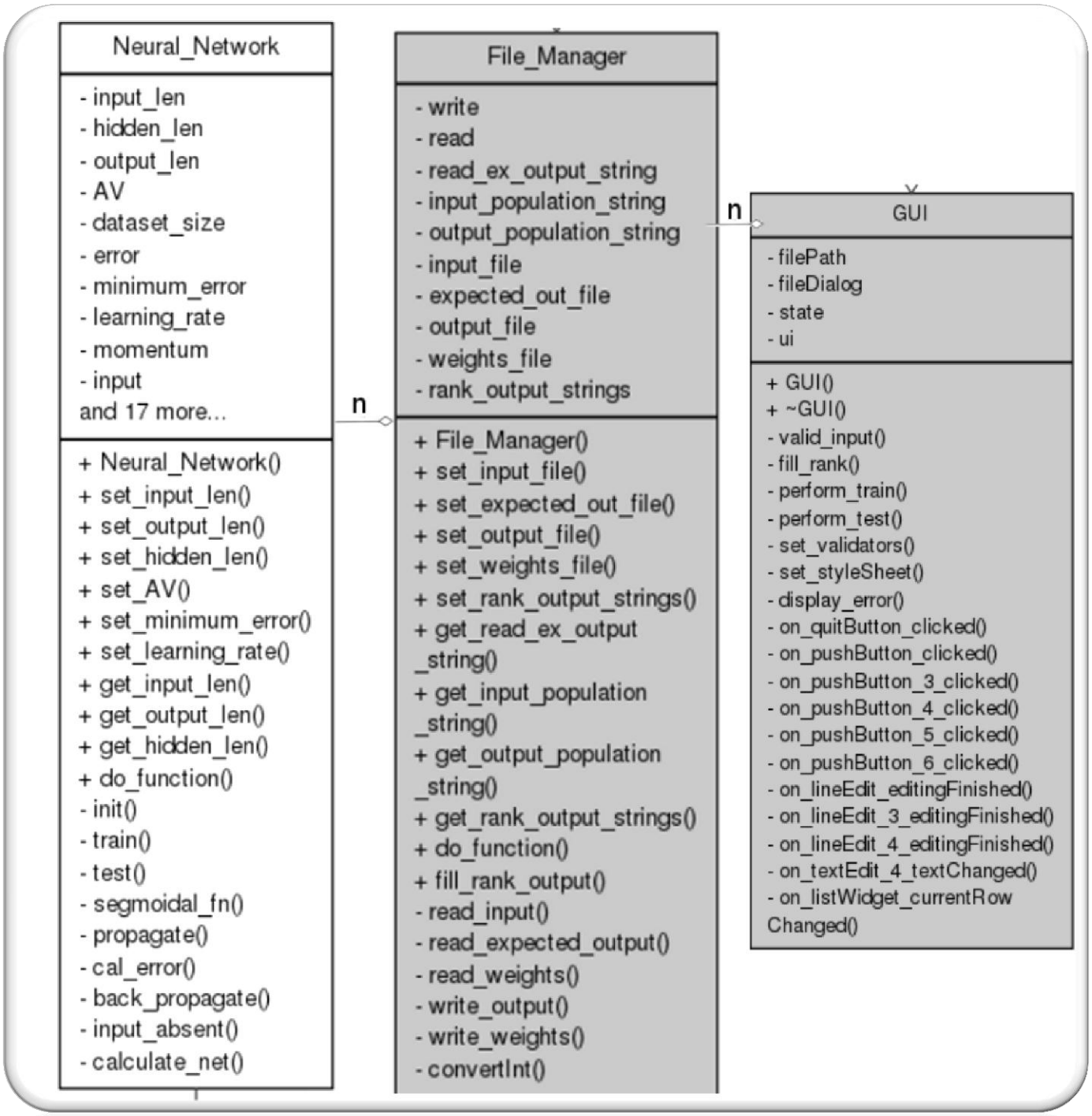
LANGUAGE AND TOOLS

- ✓ Git: used for source code management.
- ✓ Qt framework: used for development.
- ✓ Qt Creator: used for GUI development.
- ✓ C++: The main code.
- ✓ R: “ropensnp” to obtain the data from gene’s database.
- ✓ DoxyGen: used to create a website which serves as a programmer guide.

ALGORITHMS AND DATA STRUCTURES

- ✓ Feed forward Neural Network.
- ✓ Back propagation is used.
- ✓ Multilayer, input, hidden and output.
- ✓ Array, Vectors and Maps data structures are used.
- ✓ Softmax activation function is used.
- ✓ Cross entropy error function is used.

5.2 CLASS DIAGRAM



5.3 DATABASE

✓ Data is stored in plain text files.

Input File Example:

```
1
AA
CG
AT
5
TA
CC
GC
```

- Where 1 and 5 are the individual's ID.
- Each individual have 3 SNPs.
- The SNPs base letters are A,C,G and T.

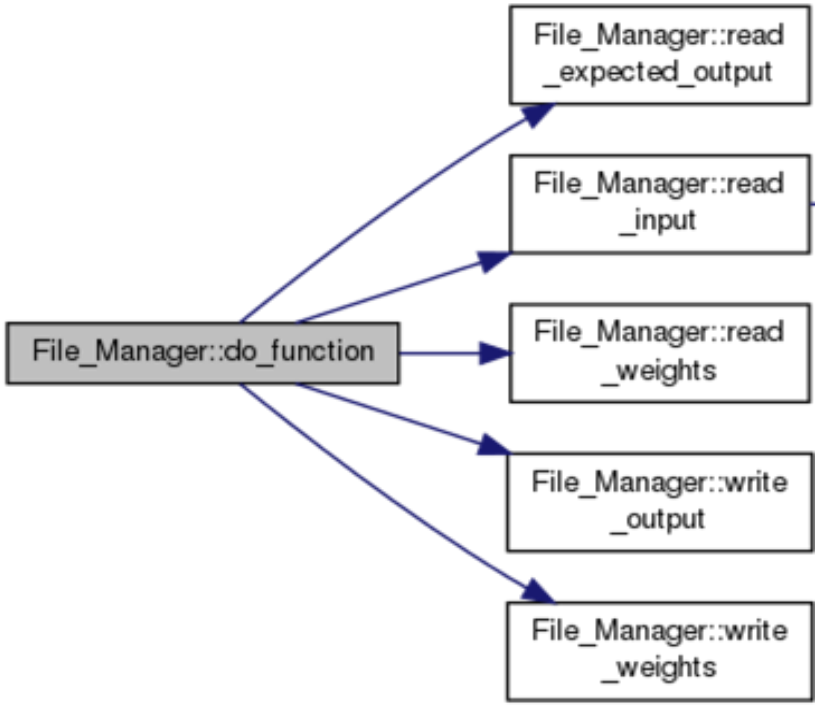
A corresponding expected output file Example:

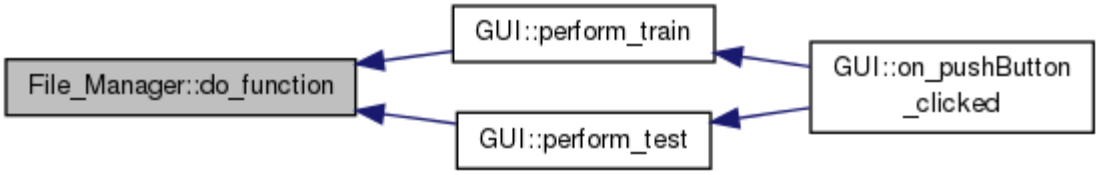
```
brown
blue
```

- Where brown is first individual's expected output.
- And blue is the second individual's expected output.

5.4 SYSTEM STRUCTURE

MAIN FUNCTIONS

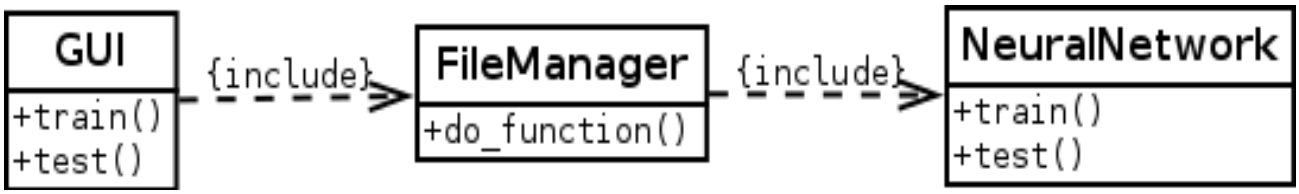




5.5 DESIGN PATTERN

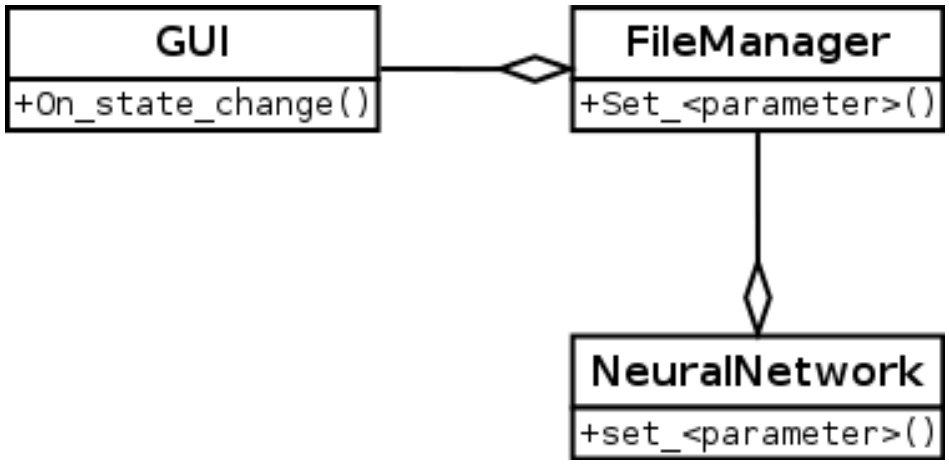
SINGELTON

Singleton is used implicitly in the implementation:



OBSERVER

Observer is also used implicitly, where any change in the GUI parameters results in internal changes in both the FileManager and the NeuralNetowk parameters.



5.6 GUI DESIGN

- ✓ The application comes with two GUI modes:
    - Basic: for users with zero or little knowledge of neural networks.
    - Expert: for advanced users to fully customize the network to their needs.

CHAPTER 6|IMPLEMENTATION

6.1 DATA SOURCE

Ropensnp

Is a library that is developed under the R-project. It was used to obtain the genotypes and phenotypes of several individuals, in our case, around 300.

R script

```
library('ropensnp')
sink("output", append=TRUE, split=FALSE)
pheno <- phenotypes_byid(phenotypeid=1, return_ = 'users')
for (i in 1:400)
{
user <- pheno[i, 1]
us <- paste0("'0,",user,"'")
gen1<-gen2<-gen3<-gen4<-gen5<-gen6<-gen7<-gen8<-gen9<--1
try(gen1 <- genotypes(snp='rs12913832', userid=us ,df=TRUE),silent=TRUE)
try(gen2 <- genotypes(snp='rs1800407', userid=us ,df=TRUE),silent=TRUE)
try(gen3 <- genotypes(snp='rs1393350', userid=us ,df=TRUE),silent=TRUE)
try(gen4 <- genotypes(snp='rs12896399', userid=us ,df=TRUE),silent=TRUE)
try(gen5 <- genotypes(snp='rs16891982', userid=us ,df=TRUE),silent=TRUE)
try(gen6 <- genotypes(snp='rs12203592', userid=us ,df=TRUE),silent=TRUE)
try(gen7 <- genotypes(snp='rs7495174', userid=us ,df=TRUE),silent=TRUE)
try(gen8 <- genotypes(snp='rs6497268', userid=us ,df=TRUE),silent=TRUE)
try(gen9 <- genotypes(snp='rs11855019', userid=us ,df=TRUE),silent=TRUE)
cat(user,"\n")
result <- 1
result <- try(cat(gen1[[7]],"\n"),silent=TRUE)
if (!is.null(result))cat(gen1,"\n")
result <- 2
result <- try(cat(gen2[[7]],"\n"),silent=TRUE)
if (!is.null(result))cat(gen2,"\n")
result <- 3
result <- try(cat(gen3[[7]],"\n"),silent=TRUE)
if (!is.null(result))cat(gen3,"\n")
result <- 4
result <- try(cat(gen4[[7]],"\n"),silent=TRUE)
if (!is.null(result))cat(gen4,"\n")
result <- 5
result <- try(cat(gen5[[7]],"\n"),silent=TRUE)
if (!is.null(result))cat(gen5,"\n")
result <- 6
result <- try(cat(gen6[[7]],"\n"),silent=TRUE)
if (!is.null(result))cat(gen6,"\n")
result <- 7
result <- try(cat(gen7[[7]],"\n"),silent=TRUE)
if (!is.null(result))cat(gen7,"\n")
result <- 8
result <- try(cat(gen8[[7]],"\n"),silent=TRUE)
if (!is.null(result))cat(gen8,"\n")
result <- 9
result <- try(cat(gen9[[7]],"\n"),silent=TRUE)
if (!is.null(result))cat(gen9,"\n")
}
pheno[[2]]
sink()
```

6.2 SIGNIFICANT CODE FRAGMENTS

TRAIN

```
88  void Neural_Network::train()
89  {
90      init(); //initialize weights,bias
91      double temp = -10000.00;
92      int rank=-1;
93      iterations = 0;
94      least_error=10000;
95      error = 10000.0;
96      while (error > minimum_error && iterations < max_iterations)//
97      {
98          mismatch=0;
99          iterations ++ ;
.00         error = 0.0;
.01         fork(0, dataset_size)
.02         {
.03             // Get current dataset of input and expected output
.04             fori(0,input_len) input[i]=input_dataset[k][i];
.05             fori(0,output_len) expected_o[i]=output_dataset[k][i];
.06
.07             // Propagate input
.08             propagate();
.09
.10             // Calculate Error
.11             error+=cal_error();
.12             temp = -100000;
.13             // Check if mismatch
.14             forj(1,output_len+1)
.15                     if (output[j-1]>temp)
.16                     {
.17                         temp = output[j-1];
.18                         rank = j;
.19                     }
.20             if(expected_o[rank]!=1)mismatch++;
.21
.22             // Back propagagte the error
.23             back_propagate();
.24         }
.25         //error/=output_len;
.26         if (error < least_error) least_error=error;
.27         //cout << "min_err" << error << "   ";
.28     }
.29 }
.30
```

TEST

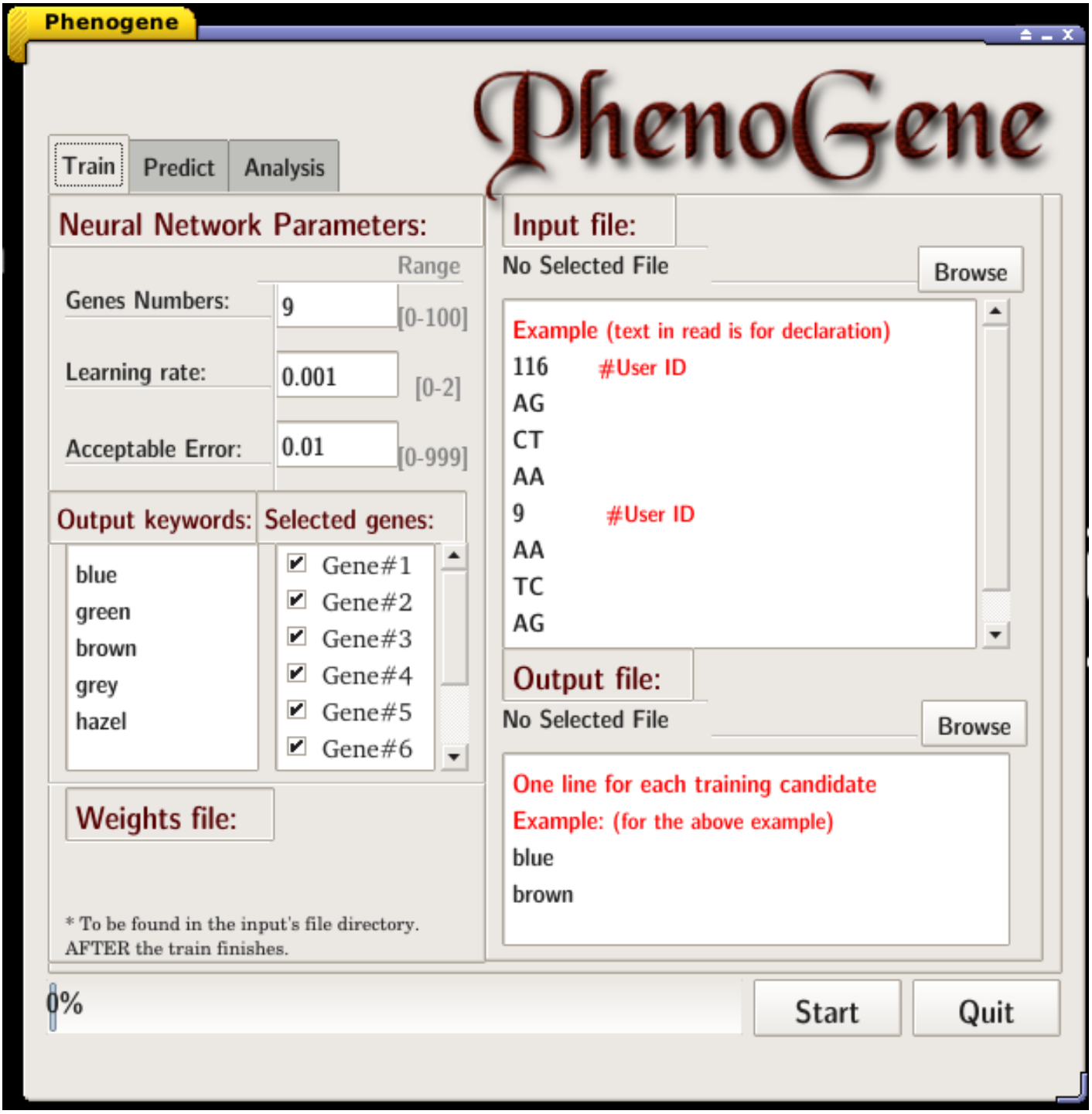
```
141 void Neural_Network::predict()
142 {
143     error = 0;
144     output_dataset.clear();
145     output_dataset.resize(dataset_size);
146     fork(0,dataset_size)
147     {
148         // Get current dataset of input
149         fori(0,input_len)
150                 input[i]=input_dataset[k][i];
151         // Calculate Output
152         propagate();
153         output_dataset[k].resize(output_len);
154         forj(0,output_len)
155                 output_dataset[k][j]=output[j];
156     }
157     return;
158 }
```

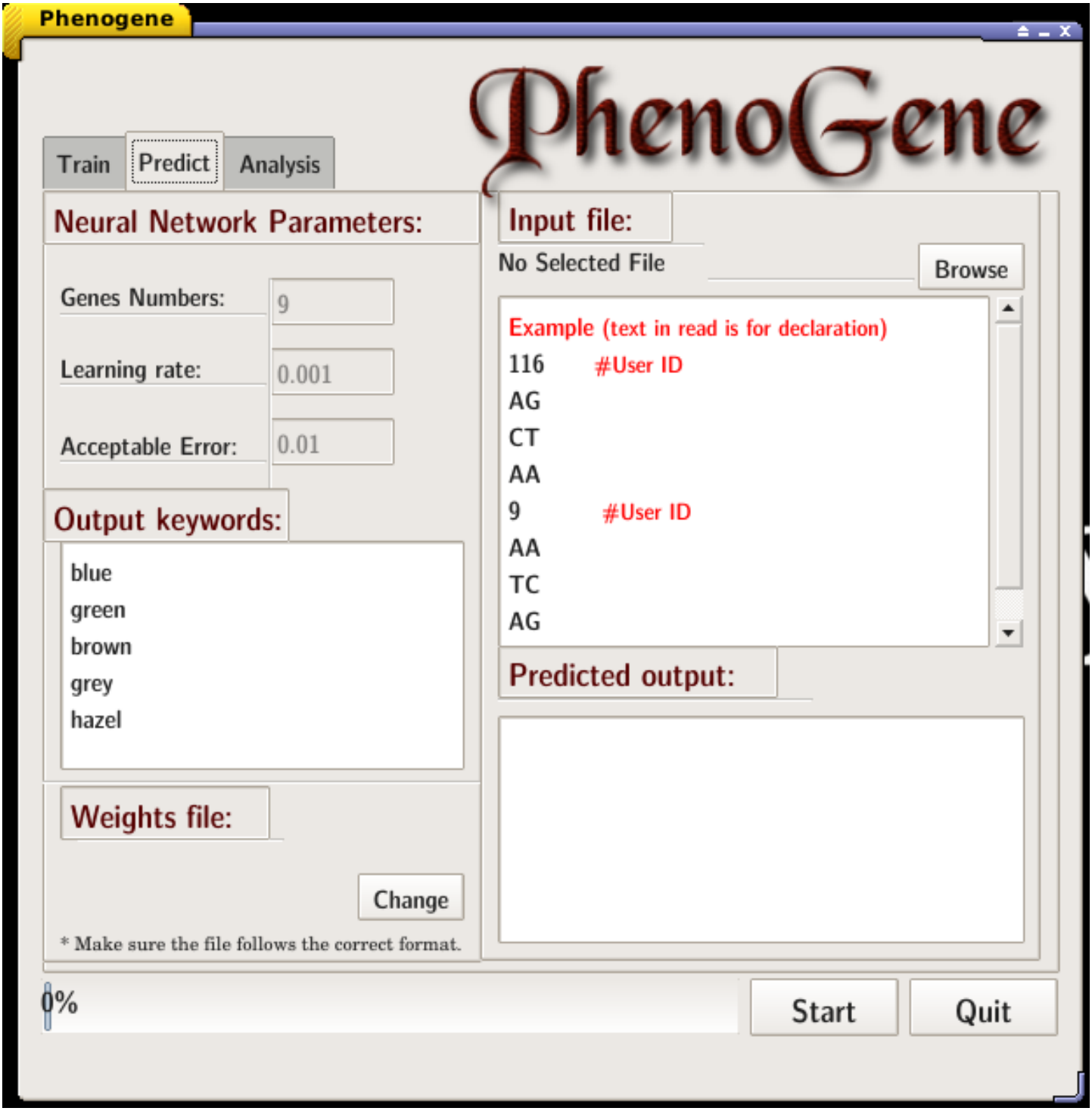
BACK PROPAGATION

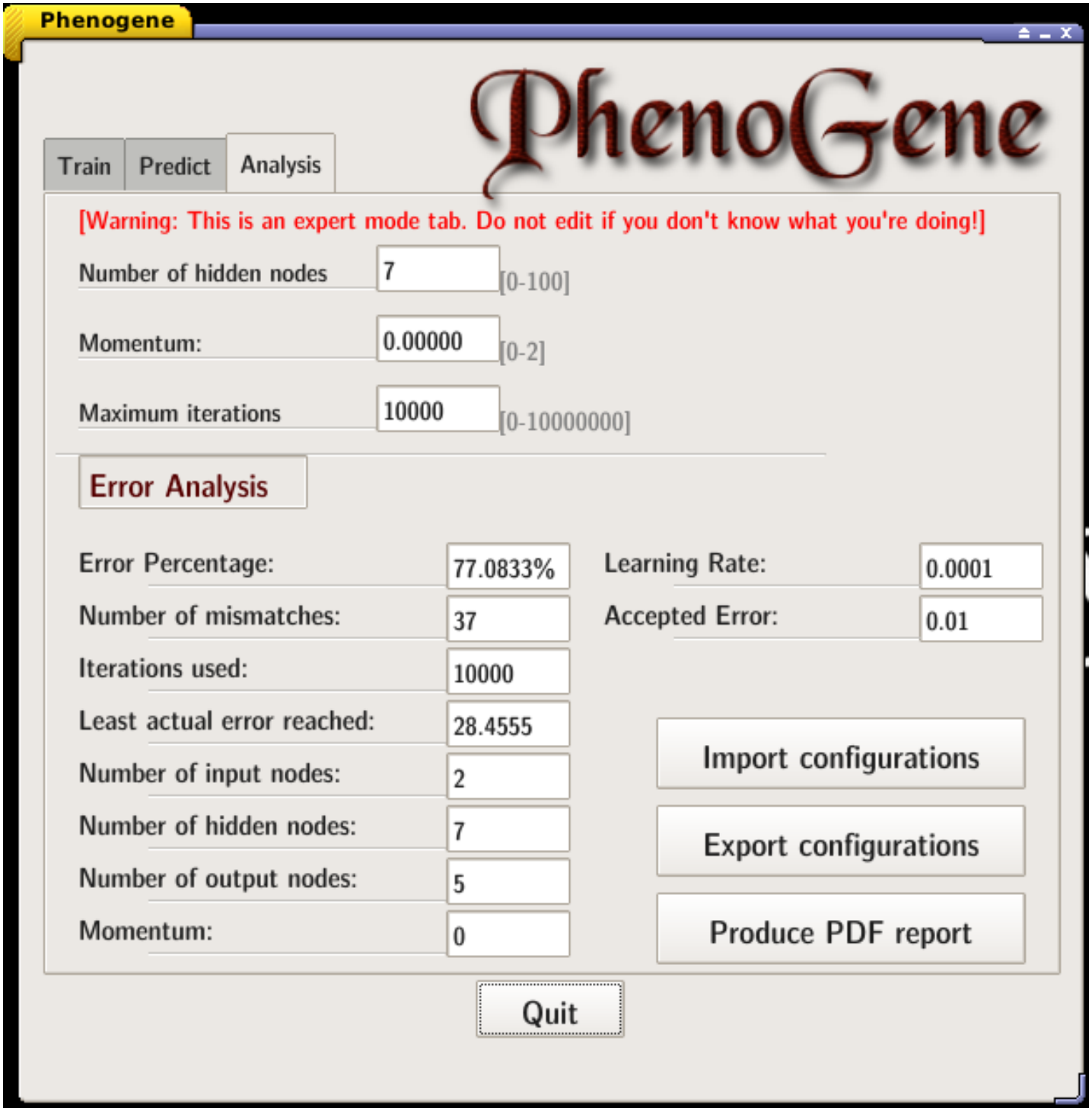
```
168 void Neural_Network::propagate()
169 {
170     fill_n(hidden, hidden_len, 0);
171     fill_n(output, output_len, 0);
172
173     double max = -1000.0;
174     double maxO = -1000.0;
175
176     /*Calculate the net (weights x input)*/
177     calculate_net(max,maxO);
178     double exp_HT=0.0;
179     double exp_OT=0.0;
180
181     /*Softmax activation function*/
182     fori(0,hidden_len) exp_HT+=hidden[i]=exp(netH[i]);
183     fori(0,output_len) exp_OT+=output[i]=exp(netO[i]);
184
185     fori(0,hidden_len) hidden[i]=hidden[i]/exp_HT;
186     fori(0,output_len) output[i]=output[i]/exp_OT;
187     return;
188 }
```

```
198 void Neural_Network::back_propagate()
199 {
200     // bias update
201     fori(0,output_len)
202             bias_O[i]-=(learning_rate*delta_O[i])+(momentum*bias_O[i]);
203     fori(0,hidden_len)
204             bias_H[i]-=(learning_rate*delta_H[i])+(momentum*bias_H[i]);
205
206     // output weights update
207     fori(0,output_len)
208             forj(0,hidden_len)
209             Wo[i][j]-=(learning_rate*delta_O[i]*hidden[j]) + (momentum*Wo[i][j]);
210
211     // hidden weights update
212     fori(0,hidden_len)
213             forj(0,input_len)
214             Wh[i][j]-=(learning_rate*delta_H[i]*input[j]) + (momentum*Wh[i][j]);
215     return;
216 }
```

6.3 GUI SCREEN SHOTS




Phenogene
PhenoGene
Train
Predict
Analysis
Neural Network Parameters:
Genes Numbers:
9
Learning rate:
0.001
Acceptable Error:
0.01
Output keywords:
blue
green
brown
grey
hazel
Weights file:
Change
* Make sure the file follows the correct format.
Input file:
No Selected File
Browse
Example (text in read is for declaration)
116 #User ID
AG
CT
AA
9 #User ID
AA
TC
AG
Predicted output:
0%
Start
Quit


Phenogene
PhenoGene
Train
Predict
Analysis
[Warning: This is an expert mode tab. Do not edit if you don't know what you're doing!]
Number of hidden nodes
7
[0-100]
Momentum:
0.00000
[0-2]
Maximum iterations
10000
[0-10000000]
Error Analysis
Error Percentage:
77.0833%
Number of mismatches:
37
Iterations used:
10000
Least actual error reached:
28.4555
Number of input nodes:
2
Number of hidden nodes:
7
Number of output nodes:
5
Momentum:
0
Learning Rate:
0.0001
Accepted Error:
0.01
Import configurations
Export configurations
Produce PDF report
Quit

CHAPTER 7|CONCLUSION

7.1 ERROR MEASURMENTS

ERROR TOLERANCE

As the bioinformatics field is fairly new, a 20-30% of correct results is accepted, that leaves a space of 70-80% of error tolerance.

The application reported 78% false predictions. That may be due to the small size of available dataset that is associated with phenotypic traits.

That, together with the probability that the SNPs tested weren't actually as significant as the research studies stated, resulted in the neural network not being able to learn and classify the input.

7.2 PARAMETERS DETERMINATION

LEARNING RATE

The learning rate is directly proportional to the dataset size.

We've noticed that as the dataset size decreases, a high learning rate would result in a higher error percentage in the training process. While a smaller learning rate would provide better results.

MOMENTUM

The momentum is only useful when the training is slow and the learnt information is too high for the neural to classify fast.

We've noticed that setting the momentum to any value higher than zero when training small datasets would result in a failure to learn.

CHAPTER 8|FUTURE WORK

PHENOTYPES

The software might be tested in other phenotypic traits, even in other organisms, to affirm or refute the results of research findings. As well as for prediction of unknown phenotypic traits upon a successful train.

NEURAL NETWORK DESIGN

Further improvements to the design of the neural network may be conducted. Using of different error function, activation function or a mix of different activation functions on the different layers.

BETTER DATA

The software may be tested as more phenotype/genotype data becomes available.

OPEN SOURCE

The software will be released as an open-source software in which other interested programmers may contribute to it.

CHAPTER 9|MANUALS

9.1 PROGRAMMER MANUAL

- ✓ The manual was generated using “doxygen”, which produced an HTML website to help the programmer navigate the code easily.
- ✓ The code commenting format follows one understood by doxygen. A programmer might view these comments in the website as well.
- ✓ The website is attached with the provided CD’s.

9.2 USER MANUAL

A user manual is attached to appendix A. The manual should guide the user through the steps of preparing their data, using the application, and make use of the analysis tools the application provides.

CHAPTER10|REFERENCES

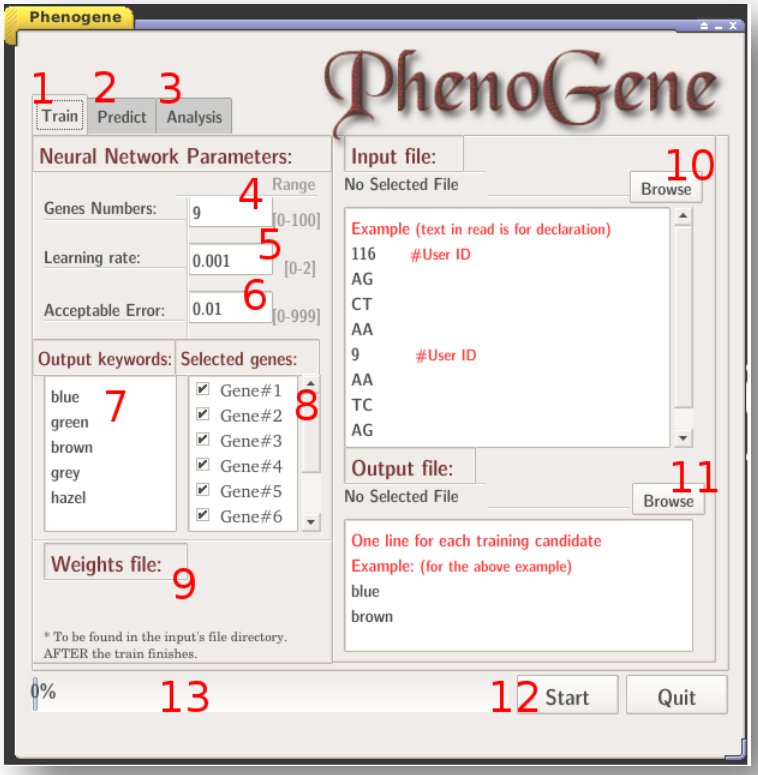
1. **Cross Entropy Error Function in Neural Networks: Forecasting Gasoline Demand:**
   G.E. Nasr, E.A. Badr and C. Joun
   School of Engineering and Architecture
   Lebanese American University, Byblos, Lebanon
   e-mail: genasr@lau.edu.lb
2. **The Error Entropy Minimization Algorithm for Neural Network Classification:**
   Jorge M. Santos, Luís A. Alexandre and Joaquim Marques de Sá INEB - Instituto de Engenharia Biomédica
   R. Roberto Frias, Porto, Portugal
   e-mail: {jms@isep.ipp.pt, lfbaa@di.ubi.pt, jmsa@fe.up.pt}
3. **Backpropagation Learning:**
   15-486/782: Artificial Neural Networks
   David S. Touretzky
4. **A Three–Single-Nucleotide Polymorphism Haplotype in Intron 1 of OCA2 Explains Most Human Eye-Color Variation:**
   David L. Duffy,* Grant W. Montgomery,* Wei Chen, Zhen Zhen Zhao, Lien Le, Michael R. James, Nicholas K. Hayward, Nicholas G. Martin, and Richard A. Sturm
5. **Eye color and the prediction of complex phenotypes from genotypes:**
   Current Biology Vol 19 No 5 R192
   Fan Liu1, Kate van Duijn1, Johannes R. Vingerling2,3, Albert Hofman2, André G. Uitterlinden2,4, A. Cecile J.W. Janssens2 and Manfred Kayser1,*
6. **Free Executive Summary - The Evaluation of Forensic DNA Evidence:**
   ISBN: 978-0-309-05395-2, 272 pages, 6 x 9, hardback (1996)

Additional references.

7. **Softmax activation function:**
   http://cseweb.ucsd.edu/~elkan/254/apr3.html

APPENDIX A | USER MANUAL

TRAIN TAB



1. Training tab
2. Predicting tab
3. Analysis tab
4. Number of input genes, i.e. how many gene SNPs do you have for each user?
5. Learning rate: This is usually a number between 0.3-0.7, if the dataset is too small choose even a smaller number. This number is directly proportional to your dataset size.
6. The least error you'd be tolerant about. Usually if this is the first time you're training the network you won't know this value, choose a small number or even zero then.
   The network will then continue to train until maximum iterations number is reached.
7. Phenotypic traits you're predicting. These keywords should match with your expected_output_file keywords.
8. Select which genes to train. This is mainly used for studying the effect of excluding a gene or more on the prediction and error rates.
9. The weights file will be produced upon successful training. You can find it in the same directory as the input and expected files you browsed earlier.

Mainly, the weights file describes the relation between the input and the output, and it will be used later on for actual prediction on new data.

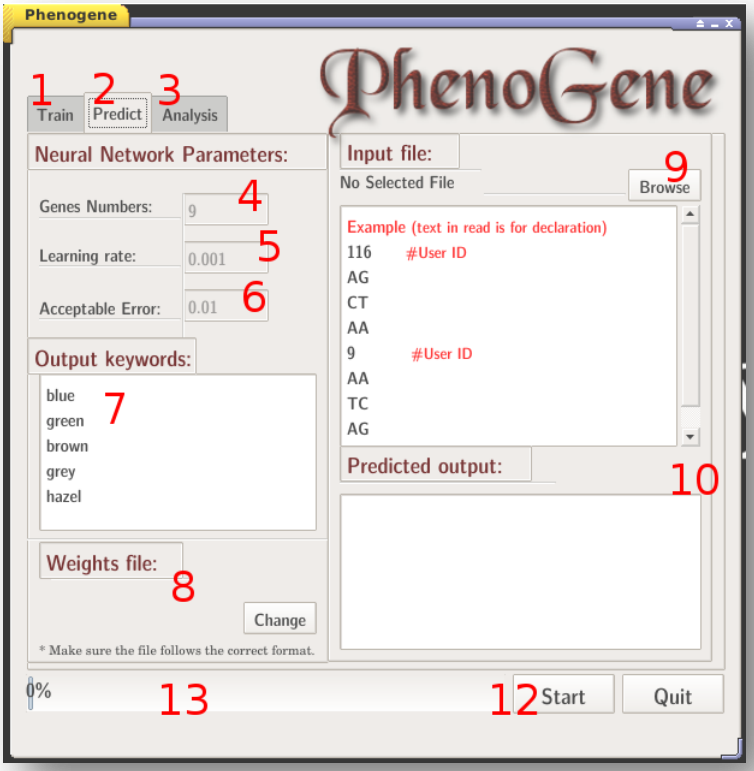
10. Browse your input file. Choose one that follows the correct format specified below:

    Your data file is a text file that has many individual records. For each individual the record starts with an integer number, his ID, follows the 2-letters gene SNPs of that individual each in a line. Then the second ID for the second individual follows his gene SNPs and so on so forth.
    Example:
    1
    AA
    AT
    2
    CC
    CG
11. Browse your expected file. Choose one that follow the correct format specified below:

    Your data file is a text file that has many individual records. For each individual there's exactly one line. The line may have one keyword or more than one keyword separated by a comma.
    Example:
    blue
    green

12. Press this button when you have filled all the data.
13. Progress bar, this is updated each time the network makes a progress.

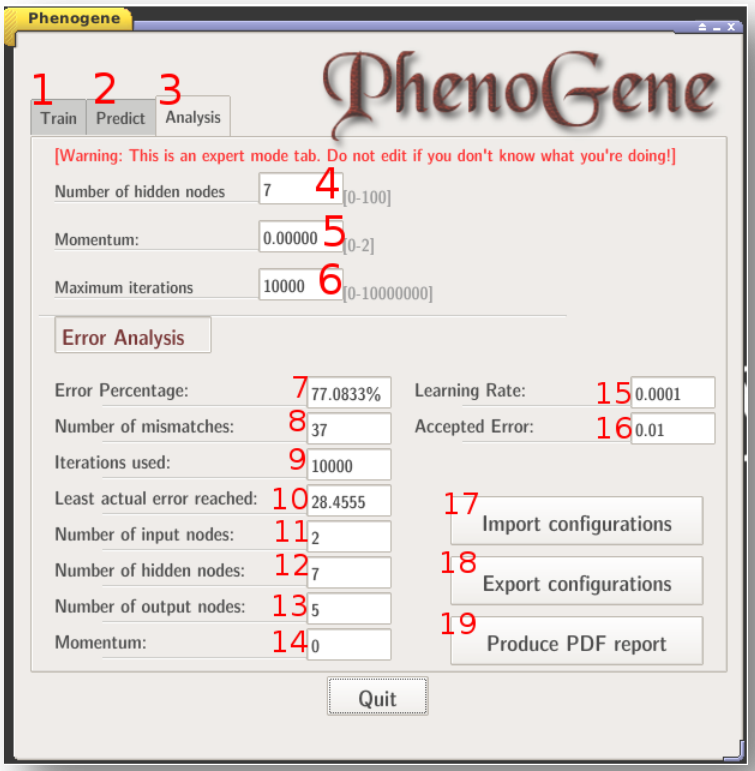
PREDICT TAB



1. training tab
2. Predicting tab
3. Analysis tab

   4, 5, 6 and 7 are not editable in the predicting tab. If you really want to edit these parameters then go to the training tab and edit, the fields will be automatically synchronized then.
8. Choose the weights file produced earlier in the training phase.
9. Browse input file, it has the same format as the training tab specified earlier.
10. This is the predicted output. Another output file with the same data is produced in the same directory as the input file.

ANALYSIS TAB



This is an expert mode tab

4. Number of hidden nodes is usually a number between input and output nodes number.
5. Momentum is used to accelerate the learning when the network is too slow, or the dataset is too big.
6. Maximum iterations number is reached when the acceptable error isn't reached during the training phase.

   7-16 error measurements to be produced after the training has finished.

17. Import all the configurations saved earlier.
18. Save all the current configurations to a file so you can restore them later.
19. Produce a PDF report of the current results. The .pdf is to be found on the input's directory. Make sure no previous PDF's is written in the same directory as this will overwrite any previous reports.

APPENDIX B | TERMINOLOGY

This appendix is presented for reference to the most used genetic terminology that might need to be shed more light on.

- **Allele**: A variant of the DNA sequence at a given locus; each unique form of a single gene.
- **DNA**: a molecule encoding the genetic instructions used in the development and functioning of all known living organisms.
- **Forensic Science:** is the application of a broad spectrum of sciences and technologies to investigate and establish facts of interest in relation to criminal or civil law.
- **Gene:** is a molecular unit of heredity of a living organism.
- **Gene Sequence:** is the process of determining the precise order of nucleotides within a DNA molecule, used to determine the order of the four bases; G, A, T, and C.
- **Genome:** is the entirety of an organism's hereditary information; encoded in DNA.
- **Genotype:** is the genetic makeup of an organism; full hereditary information.
- **GWAS:** a genome-wide association study, is an examination of many common genetic variants in different individuals to see if any variant/SNPs is associated with a trait.
- **Locus:** (plural **loci)** is the specific location of a gene or DNA sequence on a chromosome.
- **Morphology**: is a branch of bioscience dealing with the study of the form and structure of organisms and their specific structural features

- **Phenotype:** is an organism's an actual observed properties or traits.
- **SNP**: single-nucleotide polymorphism; is a DNA sequence variation occurring when a single nucleotide — A, T, C or G — in the genome differs between members of a biological species or paired chromosomes in a human

APPENDIX C | PROGRAMMER GUIDE

This website is to be found on the provided CD's as well.