A Software Developed to Identify Gene Pairs with Violated Correlation in Disease Phenotype

BY

CHAI HAOQIANG (U086101J)

November, 2010
ZB3288
Supervisor: Prof Choi Kwok Pui
Co-Supervisor: Prof Limsoon Wong

Acknowledgement

I wish to express my gratitude to my supervisors, Prof Choi Kwok Pui and Prof Limsoon Wong, for their invaluable guidance and patience throughout the course of this project. I deeply appreciate the time they sacrificed for me.

I also thank my dear friends Cao Fan and Pang Long who shared their ideas on this project with me.

Abstract

The method to identify the factors causing a disease phenotype remains an unsolved problem in many fields. We hypothesized that the differential gene expression observed in normal and disease phenotypes maybe caused by disrupted regulation of a gene. In this project, a software was developed to identify disrupted correlation between two genes, and target gene pairs were successfully identified from the type II diabetes samples.

Table of Contents

1. Introduction

Although many research has been done, the way to identify the factors causing a disease phenotype remains an unsolved problem(Burchell B, & Hume R, 1999; Gasche, Alizadeh, & Peña, 2003; Ahmad T et al., 2002). In this project, we introduced a method to identify such factors based on analysis of gene expression profiles. We hypothesized that disrupted regulation of a gene was likely to be the factor causing the differential gene expression observed in normal and disease phenotypes.

The objective of this project is to develop a software which identifies gene pairs that are highly correlated in normal samples, but are less or not correlated in disease samples. For convenience, such gene pairs are referred to as target pairs in the following parts of this report.

2. Methods and Materials

The approach was to choose pathways that are possibly related to a specific disease. All the genes in the same pathway were paired up. For each pair, if a high correlation was shown in the normal samples, but disrupted in disease samples, this gene pair was output.

2.1 Gene Expression File

The gene expression files: normal and disease samples for type II diabetes were retrieved from NCBI Gene Expression Omnibus (GEO) (Wu X, Wang J, Cui X, & Maianu L et al.,). These samples were used to demonstrate and test the software. It contained 40 normal samples and 30 disease samples.

2.2 Pathway API

Pathway API is an aggregated database combining and unifying databases from Wikipathways, Ingenuity and KEGG (Soh, Dong, Guo, & Wong, 2010). This API was installed into mySQL database.

2.3 Pre-Modification of Gene Expression File

In this section, the gene expression file are modified by marking each gene in each sample as up-regulated (noted as 1), down-regulated (noted as -1), or not regulated (noted as 0). This marking process id done is by comparing the expression value of each gene with the mean of expression value for all genes of one sample. This method for checking up or down regulation is commonly used in analysis of gene profiling data when different sampling points are not available (Yeoh et al., 2002).

The detailed procedures are listed below.

1/ Mean and standard deviation (STD) of all genes were calculated for each sample. A parameter k = {0, 0.5, 1} was chosen.This parameter will later determine how many standard deviation will be added or subtracted from mean.

2/ The expression value G for each gene was modified according to following formula:

$$G = \begin{cases} 1, & G > MEAN + k * \text{STD} \\ 0, & MEAN - \text{k} * \text{STD} < G < MEAN + k * \text{STD} \\ -1, & G < MEAN - \text{k} * \text{STD} \end{cases}$$

2.4 Correlation Calculation

Correlation was defined as the degree of relatedness between two genes. The range of correlation is [-1, 1]. A correlation of 1 implied a perfect positive correlation. A correlation of -1 implied a perfect negative correlation. Both were interesting. The following procedures were carried out for both normal and disease samples. A pathway

was chosen to study. With the help of Pathway API, genes from this pathway were selected. All the genes were paired up. For each pair Gene(i) and Gene(j), their correlation (i, j) was calculated as follows

1/ initial correlation(i, j) = 0

2/ for (int k=0; k<sampleNumber; k++)

$$\text{correlation}(i,j) = \begin{cases} \text{correlation}(i,j) + 1, & G(i,k) = G(j,k)! = 0 \\ \text{correlation}(i,j) - 1, & G(i,k)! = G(j,k) \text{ AND } G(i,k)! = 0 \text{ AND } G(j,k)! = 0 \end{cases}$$

3/ $\text{correlation}(i,j) = \text{correlation}(i,j)/\text{sampleNumber}$

Where k is a counter starts at 0 and ends when all samples are processed, sampleNumber is the number of samples in this dataset, and G(x,k) is the expression value of Gene(i) in the sample k.

Thereby the correlation(i,j) would be ranging from -1 to 1, and the closer to 1 or -1, the stronger correlation was.

2.5 Comparison Between Normal and Disease Samples

Applying step in 2.4, the correlation value for each gene pair Gene(i) and Gene (j) in the normal sample is calculated as $\text{correlation}_{\text{normal}}(i,j)$, and their correlation value in the disease sample is calculated as $\text{correlation}_{\text{disease}}(i,j)$. A threshold = {0.85, 0.9, 0.95} was chosen. For each gene pair (i,j), if

$\text{correlation}_{\text{noraml}}(i,j) > t\_res\_old \; AND \; \text{corelation}_{\text{disease}}(i,j) < t\_res\_old$

OR

$\text{correlation}_{\text{noraml}}(i,j) < -\text{threshold AND corelation}_{\text{disease}}(i,j) > -\text{threshold}$

this gene pair was output, together with $\text{correlation}_{\text{normal}}(i,j)$ and $\text{correlation}_{\text{disease}}(i,j)$.

An example was presented in the Appendix II.

3 Results

3.1 Software Developed

A software was developed to carry out the methods described in previous section. The software interface was shown in Figure 1.

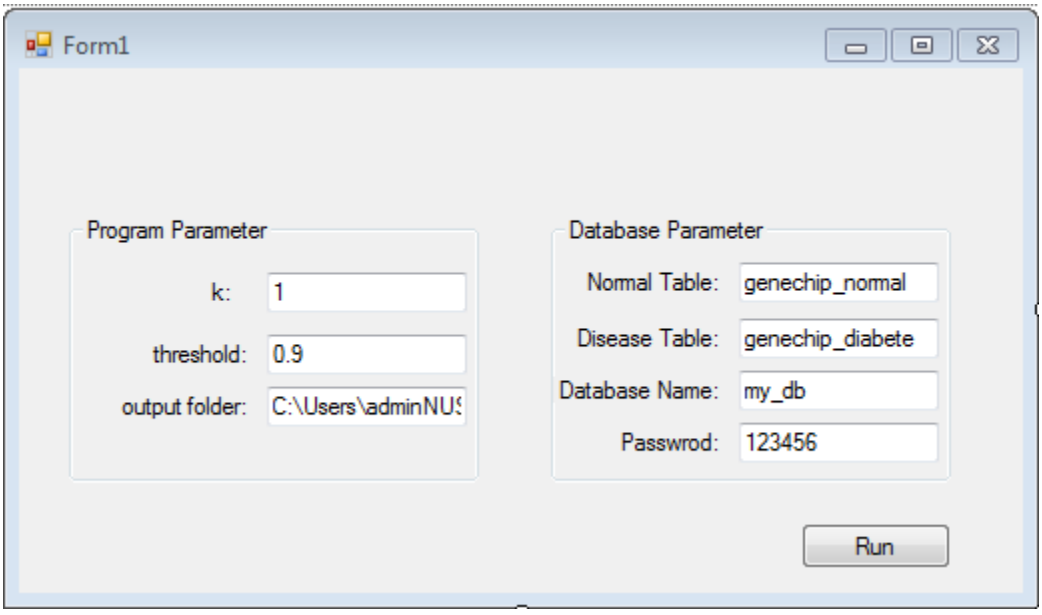

Figure 1: Software interface. The left side of interface showed software parameters and the right side of interface showed database parameters.

Execution of Software

System Requirement: Windows XP or higher, with MySQL database installed, and also should have installed Pathway API into database.

Pathway API could be downloaded from: http://www.Pathway API.com

Installation guide of the Pathway API could be found on:

http://archives.postgresql.org/pgsql-general/2001-03/msg01485.php

After all these requirements were met, the following steps were carried out.

1/ Pre-modification of data. The gene expression files for both normal and disease samples were downloaded from NCBI in .xls format. Each file was opened with Excel. The rows represented different genes, and the columns represented different samples. The first 2 columns were “ID_REF” and “IDENTIFIER” respectively, used for identifying different genes. For each sample, the mean and standard deviation of all the genes were calculated, which could be done using Excel integrated formula. The ID_REF of mean and standard deviation was named as “MEAN” and “STD” respectively, and then the file was saved in .csv format.

2/ Two .csv files were loaded into MySQL database, and the name for normal table and disease table were noted down. The method to load .csv files into MySQL database could be found at: http://blog.tjitjing.com/index.php/2008/02/import-excel-data-into-mysql-in-5-easy.html.

3/ The software was executed

“k” was the number of STD, by default it was 1;

“threshold” is the threshold as mentioned in Methods and Materials, by default it was 0.9;

“Database name” and “Database password” were the name and password of MySQL database respectively.

“Normal Table” and “Disease Table” were the name for normal table and disease table in your database as mentioned in step 1/.

It took roughly 1.5 hours to execute the software on a laptop of CPU 2.53 GHz.

4/ Output Format

For each run, the output was stored in a text file named as following format:

output_k=[kSTD]_threshold=[threshold].txt

where [kSTD] and [threshold] represented the parameter for this run.

The internal file format was as follows:

First line stated k and threshold again.

Following it was the result of each pathway, if target pairs were found in this pathway, the pathway name and all the genes belonging to it would be shown, as well as those pairs that were identified together their correlation value in normal and disease. If no target pair could be found in this pathway, only its name was shown with “Not Found” associated behind.

3.2 Identified Gene Pairs

The software was executed 3 times with different k= {0, 0.5, 0.1} and threshold = 0.9.

For k=0, 271 pathways were found to contain pairs with broken correlation.

For k=0.5, 221 pathways were found to contain pairs with broken correlation.

For k=1, 208 pathways were found to contain pairs with broken correlation.

For detailed output of each execution, refer to the “output” folder attached to this report.

4 Discussion

4.1 Consideration of An Alternative Method

Apart from the method introduced in Methods and Materials, another method "calculating covariance" was considered for this project, but this method is not suitable in this context. This method was outlined as followings. For a gene pair (i, j), their covariance in normal samples and disease samples was calculated. If the covariance in normal samples exceeding a threshold of 0.9 but not exceeding this threshold in disease samples, then this pair was selected as a target pair. However, this method had limitations when normal samples and disease samples were not produced from the same batch of gene expression profiling experiments. In this context, two datasets might have batch effect, which would introduce an inaccuracy for this method.

4.2 Select Genes from Same Pathway Instead of All Genes on Gene Chip

Pairs were obtained from genes belonging to the same pathway rather than all the genes on the gene chip. This was because the approach with all the genes was shown to produce a large number of false positives, which meant that many unrelated genes could be found to have a high correlation in normal samples. A "correlation frequency distribution graph" was plotted to show the inappropriateness of this method.

How is the Figure 2 obtained?

First, for all genes on gene chip, find all possible pairs.

Second, for each pair, calculate it's frequency of positive correlation in all samples (For example, if one pair shows positive correlation in 8 out of 10 samples, it's frequency will be 80%);

Third, count the number of gene pairs belong to each frequency interval (For example, 22% will belong to the interval of 20% - 30%); Please note that, 0 and 100% are particularly separated out from other intervals, because it's found that there're many gene pairs have a frequency of 100% or 0%.

Finally, plot the column graph.

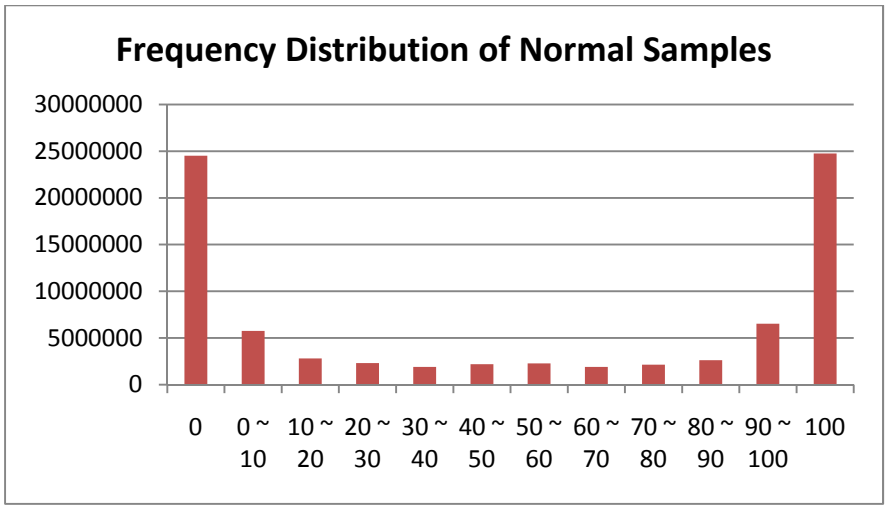

Figure 2: Frequency distribution of normal samples.

As shown in the Figure 2, it was observed that most gene pairs showed a very high positive (100% bar) or negative correlation (0% bar) using the "Marking UP and DOWN based on MEAN and STD" algorithm. While two randomly chosen genes were not supposed to have such a high correlation, there were therefore many false positives in the results. This was the rationale why this particular method was not chosen.

However, this issue could be possibly solved by choosing genes that belonged to a same pathway. Thus the false positive will be reduced vastly, since a pair of genes belonging to a same pathway will very likely to have a correlation with the other. In this project, most[1] of the pathways listed in the Pathway API were selected and genes belonging to them were studied.

[1]"Most" instead of "All" is because among all 397 pathways in Pathway API, there're 34 entries having database errors and cannot be used to select genes. Thereby these pathways are not chosen in this project. For a list of excluded pathways, please see the appendix I.

4.3 Selection of Threshold

Currently there is only one threshold set to determine if a gene pair is highly correlated in normal sample as well as not correlated in disease sample. This threshold was set to be very high, chosen from {0.85, 0.9, 0.95}. The rationale behind this high threshold is that thus only when a gene pair shows a same correlation in almost all the samples that two genes in this pair will be considered as correlated.

However, this method of choosing threshold has two problems.

The first problem is that the deliberately chosen threshold may not be an appropriate cutting point for defining “strong” correlation. A better way would be to form the distribution of correlation(i,j) over all (i,j) and discover from this the threshold.

The second problem is that using the same threshold for separating strong correlation and broken-correlation is not a very accurate way. E.g., if correlation (i,j) in normal is 0.9 and in disease is 0.89, although this is not a significant change, it will still be marked as broken correlating using our method. A suggested improvement is to have two thresholds t1 and t2. In order for correlation(i,j) to be significant, the correlation in normal should be above t1 and the correlation in disease should be below t2. Again, it’s better to determine t1 and t2 by forming the distribution of correlation(i,j) over all (i,j).

Due to time constrains, these two modification won’t be done in this project and will be left for further research.

4.4 Other Problems Remain Unsolved

Due to time constraint, there’re some problems with the software that haven’t been solved yet.

Validation of results requires literature review for the gene pairs identified by the software, this part will be continued as further work. However, by observing the output of "k=0 and threshold=0.9", it was observed that those pathways related with type II diabetes (Glycolysis, Fructose Regulation, Glycolipid Metabolism etc) were shown to contain a lot of target pairs. Hence, to some extent, the result was consistent with biological knowledge.

Many repeated gene pairs were found in the output, such as in the pathway 1 "Glycolysis / Gluconeogenesis - Homo sapiens (human)", the pair "PKM2, PKLR" was identified twice. Such repeated pairs exist very commonly in the output. This was because in the gene chip used, one particular gene could be represented by more than one probe set, and such situation was not handled during data processing. However, because these probe sets represented different parts of the same genes, and they usually have similar expression values. Thereby it did not influence the result too much.

Another problem was that too many gene pairs were output. After executing the software with different parameter sets, it was found that there were a large amount of gene pairs been identified as target pairs. It was very time consuming and inefficient to inspect these pairs individually. Hence, it would be very useful if there was a filtering mechanism in the software, and only the significant pairs would be output. Due to insufficient of time, this modification of software had to be left for further research.

5 Conclusion

Using “Marking UP and DOWN based on Mean and STD” method, the software was developed, and it had been shown to be able to successfully identify target pairs from given gene chip array data set. This validated our hypothesis that the differential gene expression observed in normal and disease phenotypes might be caused by disrupted regulation of a gene. Future work needs to be done on these areas, including validating of result, handling of repeated probe sets, and modification of software by adding filtering mechanism.

6. References

Ahmad T, Armuzzi A, Bunce M, Mulcahy-Hawes K, Marshall SE, & Orchard TR, et al. (2002). The molecular classification of the clinical manifestations of Crohn's disease. Gastroenterology, 122(4), 854-866.

Burchell B, & Hume R (1999). Molecular genetic basis of Gilbert's syndrome. Ital J Gastroenterol Hepatol, 14(10), 960-966.

Gasche, C., Alizadeh, B. Z., & Peña, A. S. (2003). Genotype-phenotype correlations: how many disorders constitute inflammatory bowel disease? Eur J Gastroenterol Hepatol, 15(6), 599-606. doi:10.1097/01.meg.0000059142.68845.e6

Soh, D., Dong, D., Guo, Y., & Wong, L. (2010). Consistency, comprehensiveness, and compatibility of pathway databases. BMC Bioinformatics, 11, 449.

Wu X, Wang J, Cui X, & Maianu L et al. GEO Dataset. Retrieved 2010.7.30, from http://www.ncbi.nlm.nih.gov/sites/GDSbrowser?acc=GDS3715

Yeoh, E. J., Ross, M. E., Shurtleff, S. A., Williams, W. K., Patel, D., & Mahfouz, R., et al. (2002). Classification, subtype discovery, and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling. Cancer Cell, 1(2), 133-143.

Appendix I. A list of pathways that have database connection error

When software wants to select genes from these pathways, it will stop working and show an “out of waiting time” exception.

123, 128, 131, 133, 167, 252, 255, 258, 263, 273, 278, 298, 299, 308, 311, 314, 335, 337, 342, 345, 351, 360, 362, 366, 377, 378, 379, 380, 381, 382, 386, 389, 390, 395, 396

Appendix II. An Example for "Marking UP and DOWN based on MEAN and STD" Method

Here's a work through for readers to have a better understanding of this methods.

First, giving the expression value of G1 and G2 for normal samples and disease samples, as well as the pre-calculated MEAN and STD, use k=1, and threshold=0.5.

Normal Samples:

	Sample 1	Sample 2	Sample 3	Sample 4
G1	9	9	8	10
G2	5	6	6	5
MEAN	7.5	6.5	7	7.5
STD	1	2	0.5	1

Disease Samples:

	Sample 1	Sample 2	Sample 3	Sample 4
G1	8	9	8	10
G2	9	8	9	9
MEAN	7.5	6.5	7	7.5
STD	1	1	0.5	1

Transform the

Normal Samples:

	Sample 1	Sample 2	Sample 3	Sample 4
G1	1	1	1	1
G2	-1	0	-1	-1

$correlation_{noraml}(G1,G2)=((-1)+(-1)+(-1))/4=-0.75$

Disease Samples:

	Sample 1	Sample 2	Sample 3	Sample 4
G1	0	1	1	1
G2	1	1	1	1

$correlation_{disease}(G1,G2)=(1+1+1)/4=0.75$

$coreraltion_{normal}$<-threshold AND $correlation_{disease}$>-threshold, satisfy the criteria, thereby this pair is selected and outputted.

Appendix III. C# Code for Software Developed

Form1.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.IO;

namespace UROPS_BrokenCorrelationAnalysis_V2._0
{
   public partial class Form1 : Form
   {
      double _kSTD;
      double _threshold;
      string _outPutFolder;
      string _normalTableName;
      string _diseaseTableName;
      Logic logic;
      public Form1()
      {
         InitializeComponent();
         logic = new Logic();
         Run();
      }
      private void Run()
      {
         _kSTD = Convert.ToDouble(textBox1.Text);
         _threshold = Convert.ToDouble(textBox2.Text);
         _normalTableName = textBox4.Text;
         _diseaseTableName = textBox5.Text;
         _outPutFolder = textBox3.Text;
         int countFound = 0;
         int[] foundPathwayNumber = new int[397];
         StreamWriter sw = new StreamWriter(_outPutFolder + "/outPut_k=" + textBox1.Text +
"_threshold=" + textBox2.Text + ".txt", true);
         sw.WriteLine("k=" + textBox1.Text + "\t" + "threshold=" + textBox2.Text);
         sw.WriteLine("========================");
         for (int i = 1; i < 397 ; i++)
         {
            int[] badPatyway = { 123, 128, 131, 133, 167, 252, 255, 258, 263, 273, 278, 298, 299, 308,
311, 314, 335, 337, 342, 345, 351, 360, 362, 366, 377, 378, 379, 380, 381, 382, 386, 389, 390, 395,
396 };
            for (int j = 0; j < badPatyway.Count(); j++)
               if (i == badPatyway[j])
                  i++;
            string[] outPut = singlePathwayRun(i);
```

```
            if (outPut[0] == "Found")
            {
               sw.WriteLine(outPut[1]);
               foundPathwayNumber[countFound] = i;
               countFound++;
               StreamWriter sw1 = new StreamWriter(_outPutFolder + "/test/" + i + " Found.txt");
               sw1.Close();
            }
            else
            {
               string pathwayName = logic.getPathwayName(i);
               sw.WriteLine("Pathway " + Convert.ToString(i) + "(Not Found):\n" + pathwayName +
"\n");
               StreamWriter sw1 = new StreamWriter(_outPutFolder + "/test/" + i + " NotFound.txt");
               sw1.Close();
            }
            sw.WriteLine("========================");
         }
         StreamWriter sw2 = new StreamWriter(_outPutFolder + "/" + "Data.txt", true);
         sw2.WriteLine("Number of pathway found:" + Convert.ToString(countFound));
         sw2.WriteLine("Pathway Numbers:" + "\n");
         for (int i = 0; i < countFound; i++)
            sw2.Write(Convert.ToString(foundPathwayNumber[i]) + ",");
         sw2.Close();
         sw.Close();
      }
      private string[] singlePathwayRun(int pathwayNumber)
      {
         string[, ,] geneChipNormal = logic.dataBaseConnection(_normalTableName, pathwayNumber,
_kSTD);
         string[, ,] geneChipDisease = logic.dataBaseConnection(_diseaseTableName,
pathwayNumber, _kSTD);

         double[,] outPut_normal = logic.stackMatrices(geneChipNormal,
Convert.ToInt16(geneChipNormal[0, 0, 0]), Convert.ToInt16(geneChipNormal[0, 0, 1]), _threshold);
         double[,] outPut_diabete = logic.stackMatrices(geneChipDisease,
Convert.ToInt16(geneChipDisease[0, 0, 0]), Convert.ToInt16(geneChipDisease[0, 0, 1]), _threshold);

         string[] compareResult = logic.compare(outPut_normal, outPut_diabete, geneChipNormal,
Convert.ToInt16(geneChipNormal[0, 0, 0]), _threshold);
         compareResult[1] = "Pathway " + Convert.ToString(pathwayNumber) + ": \n" +
logic.getPathwayName(pathwayNumber) + "\n----------\n" + "All genes: \n" + compareResult[1];
         return compareResult;
      }
   }
}
```

Logic.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data;
using MySql.Data.MySqlClient;
using System.IO;

namespace UROPS_BrokenCorrelationAnalysis_V2._0
{
   class Logic
   {
      #region Database Connection and Data Retrieving
      /// <summary>
      /// GenechipArray[0,0,0]=rowNumber, GenechipArray[0,0,1]=columnNumber
      /// </summary>
      /// <param name="tableName"></param>
      /// <param name="pathwayNumber"></param>
      /// <param name="kSTD"></param>
      /// <returns></returns>
      public string[,,] dataBaseConnection(string tableName, int pathwayNumber, double kSTD)
      {
         //StreamWriter sw = new StreamWriter("C:/Users/adminNUS/Desktop/UROP/Diabetes
6/output/test/test1.txt", true);

         MySqlConnection connection = new MySqlConnection();
         MySqlDataAdapter data = new MySqlDataAdapter();
         MySqlDataAdapter data_MEAN_STD = new MySqlDataAdapter();
         connection.ConnectionString =
         "server=localhost;"
         + "database=my_db;"
         + "uid=root;"
         + "password=123456;";
         connection.Open();

         MySqlCommand command_select = connection.CreateCommand();
         command_select.CommandText =
         "select " + tableName + ".* from " + tableName + ", gene_mapping, pathway_genes where "
         + "pathway_genes.gene_id=gene_mapping.gene_id AND "
         + tableName + ".IDENTIFIER=gene_mapping.gene_name AND "
         + "pathway_genes.pathway_id="
         + pathwayNumber;

         MySqlCommand command_select_MEAN_STD = connection.CreateCommand();
         command_select_MEAN_STD.CommandText =
         "select * from "+tableName
         + " where ID_REF='Mean' OR ID_REF='Std'";

         data.SelectCommand = command_select;
         DataSet dataset = new DataSet();
         data.Fill(dataset, "sample_data");
```

```
        data_MEAN_STD.SelectCommand = command_select_MEAN_STD;
        DataSet dataset_MEAN_STD = new DataSet();
        data_MEAN_STD.Fill(dataset_MEAN_STD, "sample_data");

        int M = 2000; // Assume genes for each pathways doesn't exceed 2000

        IDataReader dataReader = dataset.CreateDataReader();
        IDataReader dataReader_MEAN_STD = dataset_MEAN_STD.CreateDataReader();
        string[,,] geneChipArray = new string[2,M, dataReader.FieldCount];
        string[,] meanSTDArray = new string[2, dataReader.FieldCount];

        int rowNumber = 0;
        while (dataReader.Read())
        {
           for (int i = 0; i < dataReader.FieldCount; i++)
           {
              geneChipArray[1, rowNumber, i] = dataReader.GetString(i);
              //sw.Write(geneChipArray[1, rowNumber, i] + ",");
           }
           //sw.WriteLine();
           rowNumber++;
        }

        int rowNumber2 = 0;
        while (dataReader_MEAN_STD.Read())
        {
           for (int i = 0; i < dataReader_MEAN_STD.FieldCount; i++)
           {
              meanSTDArray[rowNumber2, i] = dataReader_MEAN_STD.GetString(i);
           }
           rowNumber2++;
        }

        for (int i = 0; i < rowNumber; i++)
        {
           for (int j = 2; j < dataReader.FieldCount; j++) // First two columns are not useful
           {
              if (Convert.ToDouble(geneChipArray[1, i, j]) > Convert.ToDouble(meanSTDArray[0, j])
+ kSTD * Convert.ToDouble(meanSTDArray[1, j]))
              {
                 geneChipArray[1, i, j] = "1"; // Bigger than MEAN+kSTD (normal)
                 //sw.Write(geneChipArray[1, i, j] + ",");
              }
              else if (Convert.ToDouble(geneChipArray[1, i, j]) <
Convert.ToDouble(meanSTDArray[0, j]) - kSTD * Convert.ToDouble(meanSTDArray[1, j]))
              {
                 geneChipArray[1, i, j] = "-1"; // Smaller than MEAN-kSTD
                 //sw.Write(geneChipArray[1, i, j] + ",");
              }
              else
              {
                 geneChipArray[1, i, j] = "0";
```

```
            //sw.Write(geneChipArray[1, i, j] + ",");
          }
        }
    }
    geneChipArray[0, 0, 0] = Convert.ToString(rowNumber);
    geneChipArray[0, 0, 1] = Convert.ToString(dataReader.FieldCount);


    //sw.WriteLine(geneChipArray[0, 0, 0]);
    //sw.WriteLine(geneChipArray[0, 0, 1]);
    //sw.Close();

    return geneChipArray;
}

public string getPathwayName(int pathwayNumber)
{
    MySqlConnection connection = new MySqlConnection();
    MySqlDataAdapter data = new MySqlDataAdapter();
    MySqlDataAdapter data_MEAN_STD = new MySqlDataAdapter();
    connection.ConnectionString =
    "server=localhost;"
    + "database=my_db;"
    + "uid=root;"
    + "password=123456;";
    connection.Open();

    MySqlCommand command_select = connection.CreateCommand();
    command_select.CommandText =
    "select pathway_name from pathway_names where pathway_id="
    + Convert.ToString(pathwayNumber);

    data.SelectCommand = command_select;
    DataSet dataset = new DataSet();
    data.Fill(dataset, "sample_data");

    int M = 2; // Assume genes for each pathways doesn't exceed 2000

    IDataReader dataReader = dataset.CreateDataReader();
    string[,] pathway = new string[M, dataReader.FieldCount];
    for (int i = 0; i < 2; i++) pathway[i, 0] = "0";

    int rowNumber = 0;
    while (dataReader.Read())
    {
        for (int i = 0; i < dataReader.FieldCount; i++)
        {
            pathway[rowNumber, i] = dataReader.GetString(i);
        }
        rowNumber++;
    }
    return pathway[0, 0];
}
```

```
        #endregion

        #region Stack Matrices
        public double[,] stackMatrices(string[, ,] geneChipArray, int rowNumber, int columnNumber,
double threshold)
        {
            double[,] outPut = new double[rowNumber, rowNumber];
            for (int i = 0; i < rowNumber; i++)
            {
                for (int j = 0; j < rowNumber; j++)
                    outPut[i, j] = 0;
            }

            for (int i = 0; i < rowNumber; i++)
            {
                for (int j = i+1; j < rowNumber; j++)
                {
                    outPut[i, j] = 0;
                    for (int k = 2; k < columnNumber; k++)
                    {
                        if (geneChipArray[1, i, k] == geneChipArray[1, j, k] && geneChipArray[1, i, k] !=
"0")
                            outPut[i, j] = outPut[i, j] + 1;
                        else if (geneChipArray[1, i, k] != geneChipArray[1, j, k] && geneChipArray[1, i, k] !=
"0" && geneChipArray[1, j, k] != "0")
                            outPut[i, j] = outPut[i, j] - 1;
                    }
                    outPut[i, j] = outPut[i,j] / Convert.ToDouble(columnNumber - 2); // The range thereby
will be between -1 to 1, and closer to -1 or 1 the higher correaltion
                }
            }
            return outPut;
        }

        #endregion

        #region Compare Between Normal and Disease
        public string[] compare(double[,] array1, double[,] array2, string[,,] geneChipArray, int
rowNumber, double threshold)
        {
            string[] outPut = new string[2];
            outPut[0] = "NotFound";
            outPut[1] = null;
            for (int i = 0; i < rowNumber; i++)
            {
                for (int j = 0; j < rowNumber; j++)
                {
                    if ((array1[i, j] > threshold && array2[i, j] < threshold) || (array1[i, j] < -threshold &&
array2[i, j] > -threshold))
                    {
                        outPut[0] = "Found";
                        outPut[1] += geneChipArray[1, i, 1] + "," + geneChipArray[1, j, 1] + " (" + array1[i, j]
+ ", " + array2[i, j] + ");" + "\n";
```

```
                }
            }
        }
        outPut[1] = "\n------------\nGene Pairs Found:\n" + outPut[1];
        for (int i = rowNumber-1; i >= 0; i--)
            outPut[1] = geneChipArray[1, i, 1] + "," + outPut[1];
        return outPut;
    }
    #endregion
  }
}
```

Wong Lim Soon

From: Limsoon Wong [wongls@comp.nus.edu.sg]
Sent: Saturday, 13 November, 2010 1:00 PM
To: Chai Haoqiang; Choi Kwok Pui
Cc: wongls@comp.nus.edu.sg
Subject: RE: [UROPS Continued] Method with 2 Thresholds and the Pathway Connection Problem

Dear Haoqiang,

1/ Thanks for following up.

2/ This analysis is much better. As for the two major gene/proteins:

PTPN12 is linked to diabetes II through Graves disease.

ACTL6A is linked to diabetes through effect on hepatic metabolism.

3/ Good luck with all the final exams.

Limsoon

From: Chai Haoqiang [mailto:u0806101@nus.edu.sg]
Sent: Saturday, 13 November, 2010 10:09 AM
To: Choi Kwok Pui; Limsoon Wong
Cc: Chai Haoqiang
Subject: [UROPS Continued] Method with 2 Thresholds and the Pathway Connection Problem

Dear Prof Choi and Prof Wong,

After yesterday's presentation, I went back and did the method with 2 thresholds as suggested by Prof Wong. This method filtered out large amount of noises in the original output. In the condition when k=0.2, threshold1=0.9, threshold2=0.4 (as highlighted below), 17 pathways are identified.

This is the detailed process and result:

Criteira: (correlation_normal > t1 && correlation_disease < t2) OR (correlation_normal < -t1 && correlation_disease>-t2)

t1 is to determine if the correlation in normal is strong enough, t2 is to determine if the correlation in disease is weak enough. Because correlation ranges from [-1, 1], both ±threshold should be considered.

Output:

k=0.2, t1=0.9, t2=0.6: 136 pathways identified with target pairs, still too many noises.

k=0.2, t1=0.9, t2=0.4: 17 pathways identified (this is already good enough for analysis)

k=0.2, t1=0.9, t2=0.1: 1 pathway identified

k=0.2, t1=0.9, t2=0: 0 pathways identified.

Since I don’t have much time at present, I haven’t done the situation for other k and threshods, but I think 17 pathways is a fairly acceptable number for analysis already. I did some literature reviews and came up with a short analysis for the result of “t2=0.4”, which is attached with this email. In this attachment, I also included raw outputs for the 4 sets of parameters.

Also for those pathways that have a database connection problem, I’ve checked them as Prof Wong suggested. I think it should not because those genes are from Ingenuity. Access to Ingenuity is not required for program to run. All the pathway and contained genes are included in the database before program is launched. Also, I tried but can’t find a way to check if a pathway is selected from Ingenuity or not, because a pathway list cannot be found on Ingenuity website.

I’m sorry that I didn’t manage to do this before submission of report and presentation. At that time many things compact together. I have to work very late at nights and there’s hardly time to focus on the program and wait it to run. Although UROPS is closed already, I’ll keep working on this topic, and I’ll be very much appreciated if you can continue to give me guidance and instructions on it.

Best Regards,
Haoqiang

Analysis for the Result of “k=0.2, t1=0.9, t2=0.4”

In total 17 pathways were found to contain target pairs.

Gene pair (JUN, SOCS1) appeared in 4 pathways, and there are study showeing a relation of JUN and SOCS1 with diabetes independently, but no research shown their co-effect on diabetes.

A significant pathway may be "Insulin Signaling", because it is very much related with Diabetes. 2 gene pairs were identified as broken. One is (JUN, SOCS1), the other is (JUN, IRS1).

There're 4 pathways contains a lot of target pairs, and a common genes was found in almost all these pairs. For "TNF-alpha/NF-kB Signaling Pathway" the "common gene" is ACTL6A, for "Circadianfor pathway" the "common gene" is CLOCK, and for "EGFR1 Signaling Pathway" and "T Cell Receptor Signaling Pathway", the "common gene" are the same, which is PTPN12.

However, for all these 3 genes, there is no research showing that any of them has an explicitly relation with diabetes. Also through the description of these three genes, I cannot tell how they may relate with diabetes using my knowledge. A description of each gene is attached after the pathways contain these genes.

Following is a simplified version of output result, includes only identified pathway and the target gene pairs it contains. Short analysis is also given for some pathways.

Pathway 74: Pyruvate metabolism - Homo sapiens (human)

```
ME1,ME2 (0.925, 0.366666666666667);
```

One study showed that there is substantial ME2 activity present in pancreatic islets of human(MacDonald, Longacre, & Kendrick, 2009).

Another study identified ME1 as a key regulator of a Type 2 Diabetes (T2D) in mouse and provided support for association of ME1 to T2D in human(Zhong et al., 2010).

But there’re no study found to show that these two genes’ correlation has an important role in Diabetes.

Pathway 140: Apoptosis - Homo sapiens (human)

```
TNFSF10,Fas (0.925, 0.366666666666667);
CHP2,Fas (-0.925, -0.366666666666667);
```

Pathway 159: Natural killer cell mediated cytotoxicity - Homo sapiens (human)

```
CSF2,TNFSF10 (-0.95, -0.333333333333333);
TNFSF10,Fas (0.925, 0.366666666666667);
CHP2,Fas (-0.925, -0.366666666666667);
```

Pathway 164: Circadian rhythm - Homo sapiens (human)

```
NR1D1,CLOCK (-0.95, -0.333333333333333);
NR1D1,CLOCK (-0.95, -0.333333333333333);
CRY2,CLOCK (-0.95, -0.333333333333333);
CLOCK,PER1 (-0.95, -0.333333333333333);
CLOCK,ARNTL (0.95, 0.333333333333333);
```

```
CLOCK,CSNK1D (-0.95, -0.333333333333333);
CLOCK,CSNK1E (-0.95, -0.333333333333333);
CLOCK,CRY1 (0.95, 0.333333333333333);
CLOCK,NPAS2 (0.95, 0.333333333333333);
CLOCK,NPAS2 (0.95, 0.333333333333333);
CLOCK,CLOCK (0.95, 0.333333333333333);
```

Pathway 176: Maturity onset diabetes of the young - Homo sapiens (human)

```
HNF1A,FOXA2 (0.925, 0.333333333333333);
```

Pathway 208: Acute Phase Response Signaling

```
JUN,SOCS1 (-0.925, -0.366666666666667);
```

One research showed that suppressor of SOCS1 regulates the sensitivity of pancreatic beta cells to tumor necrosis factor (Chong, Thomas, & Kay, 2002)

Pathway 218: Death Receptor Signaling

```
TNFSF10,MAP2K4 (0.95, 0.333333333333333);
```

Pathway 241: pi3k-akt Signaling

```
NOS3,RPS6KB1 (-0.925, -0.366666666666667);
```

Pathway 243: PPAR-alpha/RXR-alpha Signaling

```
JUN,IRS1 (0.95, 0.3);
```

Pathway 260: IL-5 Signaling Pathway

```
JUN,SOCS1 (-0.925, -0.366666666666667);
```

This is the same gene pair as identified from pathway 208.

Pathway 282: TNF-alpha/NF-kB Signaling Pathway

```
RELA,ACTL6A (0.925, 0.1);
HSP90AB1,ACTL6A (-0.925, -0.1);
TRAF6,ACTL6A (0.925, 0.1);
YWHAZ,ACTL6A (-0.925, -0.1);
GAB1,ACTL6A (0.925, 0.1);
GSK3B,ACTL6A (0.925, 0.0666666666666667);
RELA,ACTL6A (-0.925, -0.1);
CDC34,ACTL6A (-0.925, -0.1);
CDC34,ACTL6A (-0.925, -0.1);
MARK2,ACTL6A (0.925, 0.0666666666666667);
TRAF5,ACTL6A (0.925, 0.1);
MAP3K3,ACTL6A (-0.925, -0.1);
NFKB1,ACTL6A (-0.925, -0.1);
YWHAH,ACTL6A (-0.925, -0.1);
NFKBIA,ACTL6A (-0.925, -0.1);
TNFRSF1A,ACTL6A (-0.925, -0.1);
AKT1,ACTL6A (-0.925, -0.1);
USP11,ACTL6A (-0.925, -0.1);
IQGAP2,ACTL6A (0.925, 0.1);
MAP2K5,ACTL6A (0.925, 0.1);
MAP2K5,ACTL6A (0.925, 0.1);
BIRC3,ACTL6A (0.925, 0.1);
```

TRADD,ACTL6A (-0.925, -0.0666666666666667);
TRAF2,ACTL6A (-0.925, -0.1);
PKN1,ACTL6A (-0.925, -0.1);
CASP10,ACTL6A (0.925, 0.1);
BCL3,ACTL6A (-0.925, -0.1);
MCC,ACTL6A (0.925, 0.1);
REL,ACTL6A (0.925, 0.1);
CFLAR,ACTL6A (0.925, 0.1);
PTPN11,ACTL6A (0.925, 0.1);
PTPN11,ACTL6A (0.925, 0.1);
MAP3K8,ACTL6A (0.925, 0.1);
SRC,ACTL6A (-0.925, -0.1);
MAP3K1,ACTL6A (0.925, 0.1);
SKP1,ACTL6A (-0.925, -0.1);
AKT2,ACTL6A (0.925, 0.1);
RPS6KB1,ACTL6A (0.925, 0.1);
TNF,ACTL6A (-0.925, -0.1);
PTK2,ACTL6A (0.925, 0.1);
RPL8,ACTL6A (-0.925, -0.1);
RASAL2,ACTL6A (0.925, 0.1);
RPL30,ACTL6A (-0.925, -0.1);
KCNQ1,ACTL6A (-0.925, -0.1);
PSMD13,ACTL6A (-0.925, -0.1);
YWHAB,ACTL6A (-0.925, -0.1);
RPS11,ACTL6A (-0.925, -0.1);
TNFRSF11A,ACTL6A (0.925, 0.1);
PEBP1,ACTL6A (-0.925, -0.1);
FLNA,ACTL6A (-0.925, -0.1);
FLNA,ACTL6A (-0.925, -0.1);
FBL,ACTL6A (-0.925, -0.1);
STAT1,ACTL6A (0.925, 0.1);
GAB1,ACTL6A (0.925, 0.1);
MAP3K1,ACTL6A (0.925, 0.1);
TRADD,ACTL6A (0.925, 0.0333333333333333);
STAT1,ACTL6A (0.925, 0.0666666666666667);
RPL4,ACTL6A (-0.925, -0.1);
RPS13,ACTL6A (-0.925, -0.1);
CHUK,ACTL6A (-0.925, -0.1);
CASP8,ACTL6A (-0.925, -0.0666666666666667);
CASP8,ACTL6A (0.925, 0.0666666666666667);
TRAF2,ACTL6A (-0.925, -0.1);
TNFRSF1B,ACTL6A (-0.925, -0.1);
SMARCE1,ACTL6A (-0.925, -0.1);
CREBBP,ACTL6A (0.925, 0.1);
HSP90AB1,ACTL6A (-0.925, -0.1);
GAB1,ACTL6A (0.925, 0.1);
IKBKAP,ACTL6A (0.925, 0.1);
HDAC2,ACTL6A (0.925, 0.1);
CASP2,ACTL6A (0.925, 0.1);
GNB2L1,ACTL6A (-0.925, -0.1);
GNB2L1,ACTL6A (-0.925, -0.1);
YWHAZ,ACTL6A (-0.925, -0.1);

PSMD6,ACTL6A (-0.925, -0.1);
SMARCC2,ACTL6A (-0.925, -0.1);
MAP3K8,ACTL6A (0.925, 0.1);
PAPOLA,ACTL6A (0.925, 0.1);
AKAP8,ACTL6A (-0.925, -0.1);
TRAF5,ACTL6A (0.925, 0.1);
COPS3,ACTL6A (-0.925, -0.1);
KPNA6,ACTL6A (-0.925, -0.1);
MCC,ACTL6A (0.925, 0.1);
POLR2H,ACTL6A (-0.925, -0.0666666666666667);
KPNA3,ACTL6A (-0.925, -0.1);
HDAC6,ACTL6A (-0.925, -0.0666666666666667);
POLR2L,ACTL6A (-0.925, -0.1);
IKBKB,ACTL6A (-0.925, -0.0666666666666667);
IKBKG,ACTL6A (-0.925, -0.1);
PTK2,ACTL6A (0.925, 0.1);
CASP3,ACTL6A (0.925, 0.1);
PRKCZ,ACTL6A (0.925, 0.0666666666666667);
POLR1C,ACTL6A (0.925, 0.1);
ALPL,ACTL6A (-0.925, -0.1);
RELA,ACTL6A (-0.925, -0.1);
NR2C2,ACTL6A (0.925, 0.1);
HSPB1,ACTL6A (-0.925, -0.1);
IQGAP2,ACTL6A (0.925, 0.1);
RNF216,ACTL6A (0.925, 0.1);
CSNK2A2,ACTL6A (-0.925, -0.1);
PSMB5,ACTL6A (-0.925, -0.1);
NSMAF,ACTL6A (0.925, 0.1);
GLG1,ACTL6A (-0.925, -0.1);
TRAF4,ACTL6A (-0.925, -0.1);
CASP7,ACTL6A (0.925, 0.1);
NFKB1,ACTL6A (-0.925, -0.1);
UBE2I,ACTL6A (-0.925, -0.1);
FADD,ACTL6A (-0.925, -0.1);
PML,ACTL6A (-0.925, -0.1);
TNIP1,ACTL6A (-0.925, -0.1);
TNIP1,ACTL6A (-0.925, -0.1);
PSMD3,ACTL6A (-0.925, -0.1);
FBL,ACTL6A (-0.925, -0.1);
CAPN3,ACTL6A (-0.925, -0.1);
BTRC,ACTL6A (-0.925, -0.1);
CYLD,ACTL6A (-0.925, -0.1);
DPF2,ACTL6A (-0.925, -0.1);
PEG3,ACTL6A (0.925, 0.1);
CUL1,ACTL6A (-0.925, -0.1);
TANK,ACTL6A (0.925, 0.1);
CASP10,ACTL6A (0.925, 0.1);
KPNA6,ACTL6A (-0.925, -0.1);
PSMD7,ACTL6A (-0.925, -0.1);
NFKB2,ACTL6A (0.925, 0.1);
SMARCB1,ACTL6A (-0.925, -0.1);
TRPC4AP,ACTL6A (-0.925, -0.1);

```
KCNQ1,ACTL6A (-0.925, -0.1);
RIPK1,ACTL6A (0.925, 0.1);
NFKBIB,ACTL6A (-0.925, -0.1);
MAP2K5,ACTL6A (-0.925, -0.1);
ACTL6A,RIPK2 (0.925, 0.1);
ACTL6A,RPS6KA5 (0.925, 0.1);
ACTL6A,ZFAND5 (-0.925, -0.1);
ACTL6A,REL (0.925, 0.1);
ACTL6A,REL (0.925, 0.1);
ACTL6A,SKP1 (-0.925, -0.1);
ACTL6A,PRKACA (-0.925, -0.1);
ACTL6A,SMARCC1 (0.925, 0.1);
ACTL6A,RIPK1 (0.925, 0.1);
ACTL6A,POLR2L (-0.925, -0.1);
ACTL6A,CDC37 (-0.925, -0.1);
ACTL6A,MAP2K5 (-0.925, -0.1);
ACTL6A,NFKB2 (-0.925, -0.1);
ACTL6A,PRKACA (-0.925, -0.1);
ACTL6A,PRKACA (-0.925, -0.1);
ACTL6A,PSMC3 (-0.925, -0.1);
ACTL6A,CSNK2A1 (-0.925, -0.1);
ACTL6A,TNFAIP3 (0.925, 0.1);
ACTL6A,PSMC1 (-0.925, -0.1);
ACTL6A,CRADD (-0.925, -0.1);
ACTL6A,DDX3X (0.925, 0.1);
ACTL6A,UBE2D2 (-0.925, -0.1);
ACTL6A,UBE2I (-0.925, -0.1);
ACTL6A,TRAF1 (-0.925, -0.1);
ACTL6A,PSMD7 (-0.925, -0.1);
ACTL6A,MARK2 (0.925, 0.0666666666666667);
ACTL6A,STAT1 (0.925, 0.1);
ACTL6A,STAT1 (-0.925, -0.1);
```

From this output, it seems that ACTL6A is very important gene, because it shows violated correlation with so many genes, and the degree of violation is very high (0.925, 0.1).

However, after searching in PUBMED, no articles are found that relates this gene with diabetes or insulin. Also, through the description of this gene in NCBI, I cannot tell how it is related with Diabetes.

For your reference, here's the description of this gene. "This gene encodes a family member of actin-related proteins (ARPs), which share significant amino acid sequence identity to conventional actins. Both actins and ARPs have an actin fold, which is an ATP-binding cleft, as a common feature. The ARPs are involved in diverse cellular processes, including vesicular transport, spindle orientation, nuclear migration and chromatin remodeling. This gene encodes a 53 kDa subunit protein of the BAF (BRG1/brm-associated factor) complex in mammals, which is functionally related to SWI/SNF complex in S. cerevisiae and Drosophila; the latter is thought to facilitate transcriptional activation of specific genes by antagonizing chromatin-mediated transcriptional repression. Together with beta-actin, it is required for maximal ATPase activity of BRG1, and for the association of the BAF complex with chromatin/matrix. Three transcript variants that encode two different protein isoforms have been described." (NCBI, 2010)

Pathway 283: Adipogenesis

```
CYP26A1,IRS1 (-0.925, -0.333333333333333);
```

Pathway 290: Apoptosis

```
TNFSF10,JUN (0.925, 0.3);
TNFSF10,MAP2K4 (0.95, 0.333333333333333);
TNFSF10,Fas (0.925, 0.366666666666667);
```

Pathway 333: Circadian

```
HSPA8,CLOCK (-0.95, -0.333333333333333);
HSPA8,CLOCK (-0.95, -0.333333333333333);
RBPMS,CLOCK (-0.95, -0.333333333333333);
ETV6,CLOCK (0.95, 0.3);
AZIN1,CLOCK (0.95, 0.333333333333333);
ZFR,CLOCK (0.95, 0.333333333333333);
ZFR,CLOCK (0.95, 0.333333333333333);
AZIN1,CLOCK (0.95, 0.3);
AZIN1,BTG1 (0.925, 0.2);
GSTP1,CLOCK (-0.95, -0.333333333333333);
HSPA8,CLOCK (-0.95, -0.333333333333333);
RBPMS,CLOCK (0.95, 0.333333333333333);
UCP3,CLOCK (-0.95, -0.333333333333333);
UCP3,CLOCK (-0.95, -0.333333333333333);
UGP2,CLOCK (0.95, 0.333333333333333);
NR1D2,CLOCK (0.95, 0.333333333333333);
CRY2,CLOCK (-0.95, -0.333333333333333);
CLOCK,HIST1H2BN (0.925, 0.266666666666667);
CLOCK,HIST1H2BN (0.95, 0.333333333333333);
CLOCK,DAZAP2 (-0.95, -0.3);
CLOCK,PER1 (-0.95, -0.333333333333333);
CLOCK,ERC2 (0.95, 0.333333333333333);
CLOCK,ARNTL (0.95, 0.333333333333333);
CLOCK,IDI1 (-0.95, -0.333333333333333);
CLOCK,GFRA1 (0.95, 0.333333333333333);
CLOCK,BTG1 (0.95, 0.166666666666667);
CLOCK,UGP2 (-0.925, -0.333333333333333);
CLOCK,CRY1 (0.95, 0.333333333333333);
CLOCK,G0S2 (-0.95, -0.333333333333333);
CLOCK,CEBPB (-0.95, -0.333333333333333);
CLOCK,VAPA (-0.95, -0.333333333333333);
CLOCK,STBD1 (-0.95, -0.3);
CLOCK,CLDN5 (-0.95, -0.333333333333333);
CLOCK,DNAJA1 (0.95, 0.333333333333333);
CLOCK,PPP1R3C (-0.95, -0.333333333333333);
CLOCK,PPP1R3C (-0.95, -0.333333333333333);
CLOCK,HSPA8 (0.925, 0.333333333333333);
CLOCK,MYF6 (-0.95, -0.333333333333333);
CLOCK,QKI (0.95, 0.233333333333333);
CLOCK,KLF9 (-0.95, -0.333333333333333);
CLOCK,ZFR (0.95, 0.333333333333333);
CLOCK,TOB1 (-0.95, -0.3);
```

```
CLOCK,HSPA8 (-0.95, -0.333333333333333);
CLOCK,PURA (-0.95, -0.333333333333333);
CLOCK,EIF4G2 (-0.95, -0.333333333333333);
CLOCK,TUBB3 (-0.95, -0.333333333333333);
CLOCK,CLOCK (0.95, 0.333333333333333);
CLOCK,GSTP1 (-0.95, -0.333333333333333);
CLOCK,PPP2CB (-0.95, -0.333333333333333);
```

The description of CLOCK from NCBI is “This gene encodes a protein that belongs to the basic helix-loop-helix (bHLH) family of transcription factors. Polymorphisms within the encoded protein have been associated with circadian rhythm sleep disorders. A similar protein in mice is a circadian regulator that acts as a transcription factor and forms a heterodimer with aryl hydrocarbon receptor nuclear translocator-like to activate transcription of mouse period 1. [provided by RefSeq]”.

Pathway 341: EGFR1 Signaling Pathway

```
MAPK3,PTPN12 (-0.95, -0.233333333333333);
DUSP1,PTPN12 (-0.95, -0.233333333333333);
PLCG1,PTPN12 (0.95, 0.233333333333333);
PLCG2,PTPN12 (-0.95, -0.233333333333333);
RPS6KA1,PTPN12 (-0.95, -0.233333333333333);
MAP2K1,PTPN12 (-0.95, -0.2);
MAP2K2,PTPN12 (-0.95, -0.233333333333333);
TNK2,PTPN12 (-0.95, -0.233333333333333);
PIK3CB,PTPN12 (0.95, 0.233333333333333);
PTK6,PTPN12 (-0.95, -0.233333333333333);
PRKD1,PTPN12 (0.95, 0.233333333333333);
STAT2,PTPN12 (-0.95, -0.233333333333333);
GAB1,PTPN12 (0.95, 0.233333333333333);
EEF1A1,PTPN12 (-0.95, -0.233333333333333);
MAP3K3,PTPN12 (-0.95, -0.233333333333333);
SMAD3,PTPN12 (-0.925, -0.233333333333333);
PRKCA,PTPN12 (0.95, 0.2);
JAK1,PTPN12 (0.95, 0.233333333333333);
PTPN12,EPS8 (0.95, 0.233333333333333);
PTPN12,NRAS (0.95, 0.233333333333333);
PTPN12,AKT1 (-0.95, -0.233333333333333);
PTPN12,GRB2 (-0.95, -0.233333333333333);
PTPN12,HRAS (-0.95, -0.233333333333333);
PTPN12,PRKCI (0.95, 0.233333333333333);
PTPN12,PRKCI (0.95, 0.233333333333333);
PTPN12,JUND (-0.95, -0.233333333333333);
PTPN12,MAP2K3 (-0.95, -0.233333333333333);
PTPN12,PTPRR (0.95, 0.233333333333333);
PTPN12,PTPRR (0.95, 0.233333333333333);
PTPN12,RASA1 (0.95, 0.2);
PTPN12,GRB7 (-0.95, -0.233333333333333);
PTPN12,MAP2K5 (0.95, 0.233333333333333);
PTPN12,MAP2K5 (0.95, 0.233333333333333);
PTPN12,ARAF (-0.95, -0.233333333333333);
PTPN12,ARAF (-0.95, -0.233333333333333);
```

```
PTPN12,CSK (-0.95, -0.233333333333333);
PTPN12,ELK1 (-0.95, -0.233333333333333);
PTPN12,MCF2 (0.95, 0.233333333333333);
PTPN12,MCF2 (0.95, 0.233333333333333);
PTPN12,MYC (-0.95, -0.233333333333333);
PTPN12,PTPN11 (0.95, 0.233333333333333);
PTPN12,PTPN11 (0.95, 0.233333333333333);
PTPN12,JUN (0.95, 0.1);
PTPN12,RAF1 (-0.925, -0.233333333333333);
PTPN12,VAV1 (0.95, 0.166666666666667);
PTPN12,SMAD2 (0.95, 0.233333333333333);
PTPN12,MYC (-0.95, -0.233333333333333);
PTPN12,SRC (-0.95, -0.233333333333333);
PTPN12,KRAS (0.95, 0.2);
PTPN12,SMAD3 (-0.95, -0.233333333333333);
PTPN12,MAPK14 (0.95, 0.233333333333333);
PTPN12,PKN2 (-0.95, -0.233333333333333);
PTPN12,MAP3K1 (0.95, 0.233333333333333);
PTPN12,RAC1 (-0.95, -0.233333333333333);
PTPN12,MAPK8 (0.95, 0.233333333333333);
PTPN12,MAPK8 (0.95, 0.233333333333333);
PTPN12,MAP2K3 (-0.95, -0.233333333333333);
PTPN12,FOS (-0.95, -0.233333333333333);
PTPN12,RALB (0.95, 0.233333333333333);
PTPN12,SOS2 (0.95, 0.233333333333333);
PTPN12,DOK2 (-0.95, -0.233333333333333);
PTPN12,CAMK2A (-0.95, -0.2);
PTPN12,REPS1 (0.95, 0.233333333333333);
PTPN12,KRAS (0.95, 0.233333333333333);
PTPN12,WNK1 (0.95, 0.233333333333333);
PTPN12,WASL (0.95, 0.233333333333333);
PTPN12,PIK3R3 (0.95, 0.233333333333333);
PTPN12,PRKCA (0.95, 0.233333333333333);
PTPN12,YWHAB (-0.95, -0.233333333333333);
PTPN12,GJA1 (0.95, 0.233333333333333);
PTPN12,CEBPA (-0.95, -0.233333333333333);
PTPN12,PEBP1 (-0.95, -0.233333333333333);
PTPN12,PLSCR1 (0.95, 0.233333333333333);
PTPN12,STAT1 (0.95, 0.233333333333333);
PTPN12,RPS6KA2 (-0.95, -0.233333333333333);
PTPN12,GAB1 (0.95, 0.233333333333333);
PTPN12,NCK2 (-0.95, -0.233333333333333);
PTPN12,NCK2 (-0.95, -0.233333333333333);
PTPN12,MAP3K1 (0.95, 0.233333333333333);
PTPN12,ELK1 (-0.95, -0.233333333333333);
PTPN12,STAT1 (-0.925, -0.233333333333333);
PTPN12,STAT1 (0.95, 0.2);
PTPN12,CRKL (0.95, 0.233333333333333);
PTPN12,PTK6 (0.95, 0.233333333333333);
PTPN12,PIK3CD (-0.95, -0.233333333333333);
PTPN12,PIK3CD (-0.95, -0.233333333333333);
PTPN12,PTK2B (-0.95, -0.233333333333333);
```

PTPN12,KRT8 (-0.95, -0.233333333333333);
PTPN12,STAT5B (-0.95, -0.233333333333333);
PTPN12,ABI1 (0.95, 0.233333333333333);
PTPN12,STXBP1 (0.95, 0.233333333333333);
PTPN12,GAB1 (0.95, 0.233333333333333);
PTPN12,CAV2 (0.95, 0.233333333333333);
PTPN12,MAPK8 (0.95, 0.233333333333333);
PTPN12,PIK3R2 (-0.95, -0.233333333333333);
PTPN12,KRT17 (-0.95, -0.233333333333333);
PTPN12,PLCG1 (-0.95, -0.233333333333333);
PTPN12,CBL (0.95, 0.233333333333333);
PTPN12,PIK3R3 (0.95, 0.233333333333333);
PTPN12,USP6NL (0.95, 0.233333333333333);
PTPN12,TNK2 (-0.95, -0.233333333333333);
PTPN12,SNRPD2 (-0.95, -0.233333333333333);
PTPN12,CRKL (-0.95, -0.233333333333333);
PTPN12,PIK3R1 (0.95, 0.233333333333333);
PTPN12,CBLB (0.95, 0.233333333333333);
PTPN12,MAP3K4 (0.95, 0.233333333333333);
PTPN12,MAP3K4 (0.95, 0.233333333333333);
PTPN12,HRAS (-0.95, -0.233333333333333);
PTPN12,KRT18 (0.95, 0.2);
PTPN12,PIK3CG (0.95, 0.233333333333333);
PTPN12,ATF1 (0.95, 0.233333333333333);
PTPN12,PRKCZ (0.95, 0.2);
PTPN12,PRKCG (-0.95, -0.233333333333333);
PTPN12,INPPL1 (-0.95, -0.233333333333333);
PTPN12,STAT2 (0.95, 0.233333333333333);
PTPN12,PKN2 (0.95, 0.233333333333333);
PTPN12,SNCA (0.95, 0.233333333333333);
PTPN12,EGFR (0.95, 0.233333333333333);
PTPN12,RFXANK (0.95, 0.233333333333333);
PTPN12,JAK2 (0.95, 0.233333333333333);
PTPN12,CREB1 (0.95, 0.233333333333333);
PTPN12,SH3GL3 (0.95, 0.233333333333333);
PTPN12,GRB10 (-0.95, -0.233333333333333);
PTPN12,HTT (-0.95, -0.233333333333333);
PTPN12,PITPNA (-0.925, -0.2);
PTPN12,PRKCA (0.95, 0.233333333333333);
PTPN12,SHC1 (-0.95, -0.233333333333333);
PTPN12,ITCH (0.95, 0.233333333333333);
PTPN12,CEBPB (-0.95, -0.233333333333333);
PTPN12,ELK4 (0.95, 0.233333333333333);
PTPN12,ELK4 (0.95, 0.233333333333333);
PTPN12,PXN (-0.95, -0.233333333333333);
PTPN12,RAF1 (-0.95, -0.2);
PTPN12,RAF1 (-0.95, -0.233333333333333);
PTPN12,PIK3R2 (-0.95, -0.233333333333333);
PTPN12,ABI1 (0.95, 0.1);
PTPN12,TNIP1 (-0.95, -0.233333333333333);
PTPN12,TNIP1 (-0.95, -0.233333333333333);
PTPN12,WNK1 (0.95, 0.233333333333333);

```
PTPN12,STAT3 (-0.925, -0.166666666666667);
PTPN12,SH3BGRL (0.95, 0.266666666666667);
PTPN12,CDC42 (0.95, 0.233333333333333);
PTPN12,WASL (0.95, 0.233333333333333);
PTPN12,TNK2 (-0.95, -0.233333333333333);
PTPN12,SH3GL2 (0.95, 0.233333333333333);
PTPN12,CTNND1 (0.95, 0.233333333333333);
PTPN12,CTNND1 (-0.95, -0.233333333333333);
PTPN12,PRKCA (0.95, 0.233333333333333);
PTPN12,REPS2 (0.95, 0.233333333333333);
PTPN12,GRB14 (0.95, 0.233333333333333);
PTPN12,RIPK1 (0.95, 0.233333333333333);
PTPN12,PIK3CA (0.95, 0.233333333333333);
PTPN12,RAC1 (-0.95, -0.233333333333333);
PTPN12,EEF1A1 (-0.95, -0.233333333333333);
PTPN12,EEF1A1 (-0.95, -0.233333333333333);
PTPN12,KLF11 (0.925, 0.233333333333333);
PTPN12,SOCS3 (-0.95, -0.233333333333333);
PTPN12,HIP1 (0.95, 0.233333333333333);
PTPN12,MAP2K5 (-0.95, -0.233333333333333);
PTPN12,KRT7 (-0.95, -0.233333333333333);
PTPN12,KRT7 (0.95, 0.233333333333333);
PTPN12,PLD2 (-0.95, -0.233333333333333);
PTPN12,RPS6KA5 (0.95, 0.233333333333333);
PTPN12,JUND (-0.95, -0.233333333333333);
PTPN12,JUND (-0.95, -0.233333333333333);
PTPN12,RBBP7 (0.95, 0.233333333333333);
PTPN12,PRKCI (0.925, 0.233333333333333);
PTPN12,JAK1 (-0.95, -0.233333333333333);
PTPN12,CRKL (0.95, 0.233333333333333);
PTPN12,PIK3C2B (-0.95, -0.233333333333333);
PTPN12,PRKAR1A (-0.95, -0.233333333333333);
PTPN12,RGS16 (-0.95, -0.233333333333333);
PTPN12,HAT1 (0.95, 0.233333333333333);
PTPN12,STAT5B (-0.95, -0.233333333333333);
PTPN12,RIPK1 (0.95, 0.266666666666667);
PTPN12,CASP9 (-0.925, -0.233333333333333);
PTPN12,MAP2K5 (-0.95, -0.233333333333333);
PTPN12,CBLB (0.95, 0.233333333333333);
PTPN12,CBLB (0.95, 0.233333333333333);
PTPN12,RAB5A (0.95, 0.233333333333333);
PTPN12,CRK (0.95, 0.233333333333333);
PTPN12,CREB1 (0.95, 0.233333333333333);
PTPN12,CDC42 (0.95, 0.233333333333333);
PTPN12,CDC42 (-0.95, -0.233333333333333);
PTPN12,MAPK1 (0.95, 0.2);
PTPN12,CEACAM1 (0.95, 0.233333333333333);
PTPN12,NCK1 (0.95, 0.233333333333333);
PTPN12,STAT1 (0.95, 0.233333333333333);
PTPN12,STAT1 (-0.95, -0.233333333333333);
JUN,SOCS1 (-0.925, -0.366666666666667);
```

It seems PTPN12 is a very important gene, because it appears in almost all the target pairs, except the pair (JUN,SOCS1), which also appears in other two pathways before.

Just like ACTL6A, there's also no research shown connection between PTPN12 and diabetes. Here's the description of PTPN12 from NCBI. "The protein encoded by this gene is a member of the protein tyrosine phosphatase (PTP) family. PTPs are signaling molecules that regulate a variety of cellular processes including cell growth, differentiation, mitotic cycle, and oncogenic transformation. This PTP contains a C-terminal PEST motif, which serves as a protein-protein interaction domain, and may regulate protein intracellular half-life. This PTP was found to bind and dephosphorylate the product of the oncogene c-ABL and thus may play a role in oncogenesis. This PTP was also shown to interact with, and dephosphorylate, various products related to cytoskeletal structure and cell adhesion, such as p130 (Cas), CAKbeta/PTK2B, PSTPIP1, and paxillin. This suggests it has a regulatory role in controlling cell shape and mobility. Alternative splicing results in multiple transcript variants encoding distinct isoforms."

Pathway 352: Insulin Signaling

JUN,SOCS1 (-0.925, -0.366666666666667);

JUN,IRS1 (0.95, 0.3);

Pathway 391: T Cell Receptor Signaling Pathway

```
MAPK3,PTPN12 (-0.95, -0.233333333333333);
PLCG1,PTPN12 (0.95, 0.233333333333333);
MAP2K1,PTPN12 (-0.95, -0.2);
MAP2K2,PTPN12 (-0.95, -0.233333333333333);
SH3BP2,PTPN12 (0.95, 0.233333333333333);
SLA,PTPN12 (0.95, 0.233333333333333);
PTPN3,PTPN12 (-0.95, -0.233333333333333);
PTPN12,ITK (0.95, 0.233333333333333);
PTPN12,ITK (0.95, 0.233333333333333);
PTPN12,ZAP70 (-0.95, -0.233333333333333);
PTPN12,TUBB (-0.95, -0.233333333333333);
PTPN12,JAK3 (-0.95, -0.233333333333333);
PTPN12,AKT1 (-0.95, -0.233333333333333);
PTPN12,GRB2 (-0.95, -0.233333333333333);
PTPN12,TXK (0.95, 0.233333333333333);
PTPN12,SYK (-0.95, -0.233333333333333);
PTPN12,ABL1 (-0.95, -0.233333333333333);
PTPN12,RASA1 (0.95, 0.2);
PTPN12,PTPRJ (0.95, 0.233333333333333);
PTPN12,PTPN11 (0.95, 0.233333333333333);
PTPN12,PTPN11 (0.95, 0.233333333333333);
PTPN12,JUN (0.95, 0.1);
PTPN12,VAV1 (0.95, 0.166666666666667);
PTPN12,SRC (-0.95, -0.233333333333333);
PTPN12,MAP3K1 (0.95, 0.233333333333333);
PTPN12,JAK3 (-0.95, -0.233333333333333);
PTPN12,LYN (0.95, 0.233333333333333);
PTPN12,ABL1 (-0.95, -0.233333333333333);
PTPN12,FOS (-0.95, -0.233333333333333);
```

PTPN12,DUSP3 (-0.95, -0.233333333333333);
PTPN12,SOS2 (0.95, 0.233333333333333);
PTPN12,PTK2 (0.95, 0.233333333333333);
PTPN12,TRA@ (0.95, 0.233333333333333);
PTPN12,TRA@ (0.95, 0.233333333333333);
PTPN12,PTPN3 (-0.95, -0.233333333333333);
PTPN12,NCL (-0.95, -0.233333333333333);
PTPN12,LYN (0.95, 0.266666666666667);
PTPN12,DNM2 (-0.95, -0.233333333333333);
PTPN12,RAC2 (0.95, 0.166666666666667);
PTPN12,ITPR1 (0.95, 0.233333333333333);
PTPN12,ITPR1 (0.95, 0.233333333333333);
PTPN12,TRB@ (0.95, 0.233333333333333);
PTPN12,STAT1 (0.95, 0.233333333333333);
PTPN12,CD5 (-0.95, -0.233333333333333);
PTPN12,MAP3K1 (0.95, 0.233333333333333);
PTPN12,TRAT1 (0.95, 0.233333333333333);
PTPN12,TUBA4A (-0.95, -0.233333333333333);
PTPN12,STAT1 (-0.925, -0.233333333333333);
PTPN12,STAT1 (0.95, 0.2);
PTPN12,TXK (0.95, 0.233333333333333);
PTPN12,CRKL (0.95, 0.233333333333333);
PTPN12,GRAP (-0.95, -0.233333333333333);
PTPN12,PTK2B (-0.95, -0.233333333333333);
PTPN12,CREBBP (0.95, 0.233333333333333);
PTPN12,STAT5B (-0.95, -0.233333333333333);
PTPN12,ABI1 (0.95, 0.233333333333333);
PTPN12,PIK3R2 (-0.95, -0.233333333333333);
PTPN12,PLCG1 (-0.95, -0.233333333333333);
PTPN12,CBL (0.95, 0.233333333333333);
PTPN12,CD2AP (0.95, 0.233333333333333);
PTPN12,PSTPIP1 (-0.925, -0.133333333333333);
PTPN12,CRKL (-0.95, -0.233333333333333);
PTPN12,PIK3R1 (0.95, 0.233333333333333);
PTPN12,CD4 (0.95, 0.233333333333333);
PTPN12,WIPF1 (0.95, 0.233333333333333);
PTPN12,CBLB (0.95, 0.233333333333333);
PTPN12,SKAP2 (0.95, 0.233333333333333);
PTPN12,PTK2 (0.95, 0.233333333333333);
PTPN12,CD3E (-0.95, -0.233333333333333);
PTPN12,TUBA4A (-0.95, -0.233333333333333);
PTPN12,ACP1 (-0.95, -0.233333333333333);
PTPN12,GIT2 (0.95, 0.233333333333333);
PTPN12,PTPN22 (0.95, 0.233333333333333);
PTPN12,SYK (-0.95, -0.166666666666667);
PTPN12,CD247 (0.925, 0.233333333333333);
PTPN12,CREB1 (0.95, 0.233333333333333);
PTPN12,ARHGEF6 (-0.95, -0.233333333333333);
PTPN12,NEDD9 (0.95, 0.233333333333333);
PTPN12,CABIN1 (-0.95, -0.233333333333333);
PTPN12,UNC119 (-0.95, -0.233333333333333);
PTPN12,SHC1 (-0.95, -0.233333333333333);

PTPN12,PPP3CB (0.95, 0.233333333333333);
PTPN12,CD3D (0.95, 0.2);
PTPN12,CEBPB (-0.95, -0.233333333333333);
PTPN12,RASGRP2 (-0.95, -0.233333333333333);
PTPN12,RASGRP2 (-0.95, -0.233333333333333);
PTPN12,PXN (-0.95, -0.233333333333333);
PTPN12,RAP1A (0.95, 0.233333333333333);
PTPN12,MUC1 (-0.95, -0.233333333333333);
PTPN12,MUC1 (-0.95, -0.233333333333333);
PTPN12,MUC1 (0.95, 0.2);
PTPN12,SKAP1 (0.95, 0.233333333333333);
PTPN12,GRAP2 (-0.95, -0.233333333333333);
PTPN12,GRAP2 (0.95, 0.233333333333333);
PTPN12,GRAP2 (0.95, 0.233333333333333);
PTPN12,BCL10 (0.95, 0.233333333333333);
PTPN12,PIK3R2 (-0.95, -0.233333333333333);
PTPN12,ABI1 (0.95, 0.1);
PTPN12,PRKCQ (-0.95, -0.233333333333333);
PTPN12,WAS (-0.95, -0.233333333333333);
PTPN12,WAS (-0.95, -0.233333333333333);
PTPN12,CD3G (0.95, 0.233333333333333);
PTPN12,LCP2 (0.95, 0.233333333333333);
PTPN12,KHDRBS1 (-0.95, -0.233333333333333);
PTPN12,ABL1 (-0.95, -0.233333333333333);
PTPN12,CDC42 (0.95, 0.233333333333333);
PTPN12,PPP3CB (-0.95, -0.233333333333333);
PTPN12,DLG1 (0.95, 0.233333333333333);
PTPN12,DLG1 (0.95, 0.233333333333333);
PTPN12,FYN (0.95, 0.233333333333333);
PTPN12,FYN (0.95, 0.266666666666667);
PTPN12,PTPRC (0.95, 0.233333333333333);
PTPN12,PTPRC (-0.95, -0.233333333333333);
PTPN12,PTPRJ (0.95, 0.233333333333333);
PTPN12,SHB (-0.925, -0.233333333333333);
PTPN12,CD8A (0.95, 0.233333333333333);
PTPN12,CTNNB1 (-0.95, -0.233333333333333);
PTPN12,STK39 (0.95, 0.233333333333333);
PTPN12,STK39 (0.95, 0.233333333333333);
PTPN12,DUSP3 (-0.95, -0.233333333333333);
PTPN12,DUSP3 (-0.95, -0.233333333333333);
PTPN12,RIPK2 (0.95, 0.233333333333333);
PTPN12,CRKL (0.95, 0.233333333333333);
PTPN12,FYB (0.95, 0.233333333333333);
PTPN12,STAT5B (-0.95, -0.233333333333333);
PTPN12,CBLB (0.95, 0.233333333333333);
PTPN12,CBLB (0.95, 0.233333333333333);
PTPN12,SYK (0.95, 0.233333333333333);
PTPN12,ITPR1 (0.95, 0.233333333333333);
PTPN12,CRK (0.95, 0.233333333333333);
PTPN12,CD4 (0.925, 0.233333333333333);
PTPN12,CREB1 (0.95, 0.233333333333333);
PTPN12,MUC1 (-0.95, -0.233333333333333);

```
PTPN12,CDC42 (0.95, 0.233333333333333);
PTPN12,CDC42 (-0.95, -0.233333333333333);
PTPN12,MAPK1 (0.95, 0.2);
PTPN12,NCK1 (0.95, 0.233333333333333);
PTPN12,STAT1 (0.95, 0.233333333333333);
PTPN12,STAT1 (-0.95, -0.233333333333333);
```

References

Chong, M., Thomas, H., & Kay, T. (2002). Suppressor of cytokine signaling-1 regulates the sensitivity of pancreatic beta cells to tumor necrosis factor. *J Biol Chem., 277*(31), 27945-27952.

MacDonald, M., Longacre, M., & Kendrick, M. (2009). Mitochondrial malic enzyme (ME2) in pancreatic islets of the human, rat and mouse and clonal insulinoma cells. *Arch Biochem Biophys., 488*(2), 100-104.

(8, Nov, 2010). *ACTL6A actin-like 6A [Homo sapiens].* Retrieved 13, Nov, 2010, from http://www.ncbi.nlm.nih.gov/pubmed?Db=gene&Cmd=retrieve&dopt=full_report&list_uids=86

Zhong, H., Beaulaurier, J., Lum, P., Molony, C., Yang, X., & Macneil, D., et al. (2010). Liver and adipose expression associated SNPs are enriched for association to type 2 diabetes. *PLoS Genet.,*