

```
(* © Orly Alter 2005 *)
```

```
(* All Rights Reserved *)
```

```
(* Project Genes x Genes Network Onto Cell Cycle Transcription Factors' DNA-Binding Data *)
```

```
(* Read Cell Cycle Transcription Factors' DNA-Binding Data *)
```

```
stream = "Desktop/Network_Decomposition/Data/Cell_Cycle_Binding.txt";
matrix = Import[stream, "Table"];
{genes, arrays} = Dimensions[matrix] - {1, 7}
Clear[stream];

{2120, 12}

genenames = TakeRows[
  TakeColumns[matrix, {1, 7}],
  {2, genes + 1}];
arraynames = TakeColumns[
  TakeRows[matrix, {1, 1}],
  {8, arrays + 7}];
matrix = TakeColumns[
  TakeRows[matrix, {2, genes + 1}],
  {8, arrays + 7}];
matrix = ToExpression[matrix];

sizes = Flatten[
  Table[
    Dimensions[
      Characters[
        ToString[arraynames[[1, a]]
      ]]],
    {a, 1, arrays}]];
size = Sort[sizes, OrderedQ[{{#2, #1}} &]][[1]];
Do[
  Do[arraynames[[1, a]] = StringJoin[ToString[arraynames[[1, a]]], " ",
    {b, 1, size - sizes[[a]]}],
  {a, 1, arrays}];

(* Convert to Ratios *)

average = Table[1, {a, 1, arrays}];
average = N[average / Sqrt[Dot[average, average]]];
matrix = matrix / N[Outer[Times, Dot[matrix, average], average]];

matrix2 = matrix;
genenames2 = genenames;
arraynames2 = arraynames;
{genes2, arrays2} = Dimensions[matrix2]

{2120, 12}
```

```
(* Calculate SVD *)
```

```
{eigenarrays, eigenexpressions, eigengenex} = SingularValues[matrix];  
eigenarrays = Transpose[eigenarrays];  
fractions = eigenexpressions ^ 2 /  
  Sum[eigenexpressions[[a]] ^ 2, {a, 1, Dimensions[eigenexpressions][[1]]}];  
entropy = -N[Sum[fractions[[a]] * Log[fractions[[a]]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}] /  
  Log[Dimensions[eigenexpressions][[1]]];  
entropy = N[Round[100 * entropy] / 100]
```

```
0.22
```

```
(* Create Fractions Bar Charts Displays *)
```

```
fractions[[2]]
```

```
0.026749
```

```
limit = 0.03;
```

```
alsolimit = fractions[[2]];
```

```
Clear[gridx, framex, framey, sizes];
```

```
gridx = Table[a, {a, 0, limit, N[limit/5]}];
```

```
framex = gridx;
```

```
sizes = Flatten[
```

```
  Table[
```

```
    Dimensions[
```

```
      Characters[
```

```
        ToString[framex[[a]]
```

```
      ]], {a, 1, 6}]]];
```

```
Do[
```

```
  Do[framex[[a]] = StringJoin[ToString[framex[[a]]], " "],  
    {b, 1, 5 - sizes[[a]]},
```

```
    {a, 1, 6}];
```

```
framex = Table[{gridx[[a]], framex[[a]]}, {a, 1, 6}];
```

```
gridx = Table[{gridx[[a]], RGBColor[0, 0, 0]}, {a, 1, 6}];
```

```
framey = Table[{a + 1, arrays - a}, {a, 0, arrays - 2}];
```

```
table = Table[fractions[[arrays - a]], {a, 0, arrays - 2}];
```

```
g = BarChart[table,
```

```
  BarOrientation -> Horizontal,
```

```
  PlotRange -> {{0, alsolimit * 1.0001}, {0.5, 12 - 1 + 0.5}},
```

```
  AspectRatio -> 1,
```

```
  Axes -> False,
```

```
  Frame -> True,
```

```
  FrameTicks -> {None, framey, framex, None},
```

```
  FrameLabel -> {None, None, None, None},
```

```
  GridLines -> {gridx, None},
```

```
  DisplayFunction -> Identity];
```

```
g = FullGraphics[g];
```

```
g[[1, 2]] = g[[1, 2]] /.
```

```
  Text[a_, {b_, c_}, {0., -1.}] ->
```

```
  Text[a, {b, c + 1.75}, {0, 0}, {0, 1}];
```

```
g1 = Show[g,
```

```
  AspectRatio -> 1.25,
```

```
  PlotRange -> All,
```

```
  DisplayFunction -> Identity];
```

```

gridx = Table[a, {a, 0, 1, 0.2}];
framex = gridx;
sizes = Flatten[
  Table[
    Dimensions[
      Characters[
        ToString[framex[[a]]
      ]], {a, 1, 6}]];
Do[
  Do[framex[[a]] = StringJoin[ToString[framex[[a]]], " ",
    {b, 1, size - sizes[[a]]}],
    {a, 1, 6}];
framex = Table[{gridx[[a]], framex[[a]]}, {a, 1, 6}];
gridx = Table[{gridx[[a]], RGBColor[0, 0, 0]}, {a, 1, 6}];
framey = Table[{a + 1, arrays - a}, {a, 0, Dimensions[eigenexpressions][[1]]}];
labelx = ColumnForm[
  {"(b) Eigenbinding Fraction", StringJoin["d2 = ", ToString[entropy]], " "},
  Center];
g = BarChart[
  Table[fractions[[arrays - a]], {a, 0, arrays - 1}],
  BarOrientation -> Horizontal,
  PlotRange -> {{0, 1.0001}, {0.5, arrays + 0.5}},
  AspectRatio -> 1,
  Axes -> False,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, None, labelx, None},
  GridLines -> {gridx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 1.5}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 1.25}, {0, 0}, {0, 1}];
g2 = Show[{g,
  Graphics[{RGBColor[1, 1, 0.8], Rectangle[{0.1, 0.6}, {0.98, 11.3}]},
  Graphics[{Rectangle[{0.1, 0.6}, {0.98, 11.3}, g1]}]},
  AspectRatio -> 1.35,
  PlotRange -> All,
  DisplayFunction -> Identity];

```

```
(* Create Eigenenes 2D Red & Green Raster Display *)
```

```
contrast = 3.5;
displaying = Table[
  If[contrast * eigengenenes[[i, j]] > 0,
    If[contrast * eigengenenes[[i, j]] < 1, {contrast * eigengenenes[[i, j]], 0}, {1, 0}],
    If[contrast * eigengenenes[[i, j]] > -1, {0, -contrast * eigengenenes[[i, j]]}, {0, 1}]],
  {i, 1, Dimensions[eigenexpressions][[1]]}, {j, 1, arrays}];
framex = Table[{a - 0.5, arraynames[[1, a]]}, {a, 1, arrays}];
framey = Table[{a + 1 - 0.5, arrays - a}, {a, 0, Dimensions[eigenexpressions][[1]]}];
labely = "Eigenenes";
labelx = ColumnForm[{"(a) Arrays", " ", " "}, Center];

g = Show[
  Graphics[
    RasterArray[
      Table[
        RGBColor[displaying[[i, j, 1]], displaying[[i, j, 2]], 0],
        {i, Dimensions[eigenexpressions][[1]], 1, -1}, {j, 1, arrays}]]],
  AspectRatio -> 1,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, labely, labelx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 2.2, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 1.5}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 1.25}, {0, 0}, {0, 1}];
g1 = Show[g,
  AspectRatio -> 1.05,
  PlotRange -> All,
  DisplayFunction -> Identity];
```

(* Create Selected Eigengenes Graph Display *)

```

labelx = ColumnForm[{"(c) Arrays"}, Center];
labely = ColumnForm[{" ", "Binding Level"}, Center];
framex = Table[{a - 1, arraynames[[1, a]]}, {a, 1, arrays}];
framey = {-1, -0.5, 0, 0.5, 1};
coordinates = Table[{a - 1, eigengenes[[1, a]]}, {a, 1, arrays}];
points = Table[Point[coordinates[[a]]], {a, 1, arrays}];
line = Line[coordinates];
g = Show[
  {Graphics[{RGBColor[1, 0, 0], PointSize[0.022], points}],
   Graphics[{RGBColor[1, 0, 0], line}]},
  Frame -> True,
  FrameLabel -> {None, labely, labelx, None},
  GridLines -> {None, {{0, RGBColor[0, 0, 0]}}},
  FrameTicks -> {None, framey, framex, None},
  PlotRange -> {-1.05, 1.05},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 3, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 0.5}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 0.2}, {0, 0}, {0, 1}];
g3 = Show[g,
  AspectRatio -> 1.05,
  PlotRange -> All,
  DisplayFunction -> Identity];

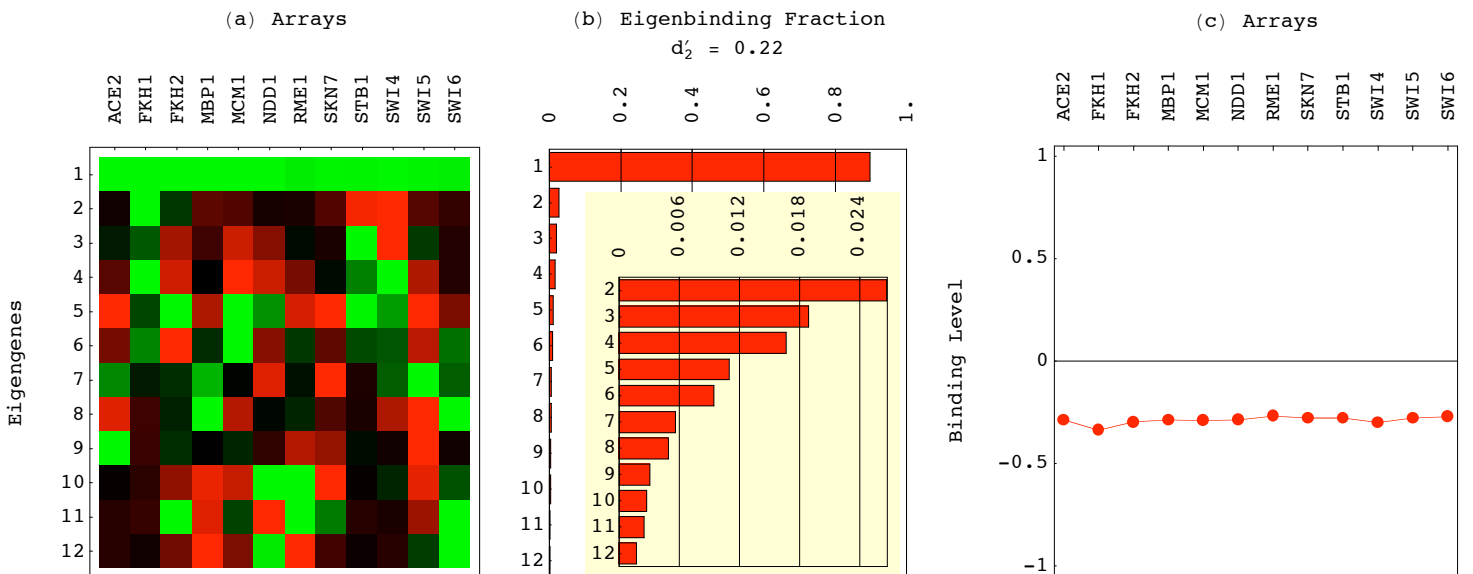
```

(* Display Eigengenes, Fractions and Selected Eigengenes *)

```

Show[GraphicsArray[{g1, g2, g3}],
  GraphicsSpacing -> -0.15];

```



(* Reconstruct Data Without Additive Steady State *)

```
{eigenarrays, eigenexpressions, eigengenes} = SingularValues[matrix];
eigenarrays = Transpose[eigenarrays];
eigenexpressions[[1]] = 0;
matrix = Dot[eigenarrays, Dot[DiagonalMatrix[eigenexpressions], eigengenes]];

matrix2 = matrix;
genenames2 = genenames;
arraynames2 = arraynames;
{genes2, arrays2} = Dimensions[matrix2]

{2120, 12}
```

(* Project mRNA Expression Data Onto Proteins' DNA-Binding Data *)

```
genes = genes1;
genenames = TakeColumns[genenames1, 1];
arrays = arrays1;
arraynames = arraynames1;

externalgenes = genes2;
externalgenenames = TakeColumns[genenames2, 1];
externalarrays = arrays2;
externalarraynames = arraynames2;

list = Flatten[Intersection[genenames, externalgenenames]];
{partialgenes} = Dimensions[list]

{1588}

counter = Table[{Position[genenames, list[[a]]][[1, 1]], list[[a]]}, {a, 1, partialgenes}];
counter = ReplaceAll[counter, {} -> {Null}];
counter = Sort[counter, OrderedQ[{{#1, #2}} &];
list = Flatten[TakeColumns[counter, {2}]];

counter = Table[Flatten[Position[list, genenames[[a, 1]]], {a, 1, genes}];
counter = ReplaceAll[counter, {} -> {Null}];
partialmatrix = AppendRows[counter, genenames1, matrix1];
partialmatrix = Sort[partialmatrix, OrderedQ[{{#1, #2}} &];
partialgenenames = TakeRows[
  TakeColumns[partialmatrix, {2, 8}],
  {1, partialgenes}];
partialmatrix = TakeRows[
  TakeColumns[partialmatrix, {9, arrays + 8}],
  {1, partialgenes}];
originalpartial = partialmatrix;

counter = Table[Flatten[Position[list, externalgenenames[[a, 1]]], {a, 1, externalgenes}];
counter = ReplaceAll[counter, {} -> {Null}];
partialexternalmatrix = AppendRows[counter, externalgenenames, matrix2];
partialexternalmatrix = Sort[partialexternalmatrix, OrderedQ[{{#1, #2}} &];
partialexternalgenenames = TakeRows[
  TakeColumns[partialexternalmatrix, {2, 2}],
  {1, partialgenes}];
partialexternalmatrix = TakeRows[
  TakeColumns[partialexternalmatrix, {3, externalarrays + 2}],
  {1, partialgenes}];

partialmatrix = Dot[partialexternalmatrix, Dot[PseudoInverse[partialexternalmatrix], partialmatrix]];
```

```
(* Examine mRNA Expression Data After Projection *)
```

```
(* Calculate SVD *)
```

```
matrix = partialmatrix;  
genes = partialgenes;  
genenames = TakeColumns[partialgenenames, 1];  
  
{eigenarrays, eigenexpressions, eigengenes} = SingularValues[matrix];  
eigenarrays = Transpose[eigenarrays];  
fractions = eigenexpressions ^ 2 /  
  Sum[eigenexpressions[[a]] ^ 2, {a, 1, Dimensions[eigenexpressions][[1]}];  
entropy = -N[Sum[fractions[[a]] * Log[fractions[[a]]],  
  {a, 1, Dimensions[eigenexpressions][[1]}] /  
  Log[Dimensions[eigenexpressions][[1]]];  
entropy = N[Round[100 * entropy] / 100]
```

```
0.49
```

```
(* Create Fractions Bar Charts Display *)
```

```
fractions[[1]]
```

```
0.537925
```

```
limit = 0.55;
```

```
alsolimit = fractions[[1]];
```

```

gridx = Table[a, {a, 0, limit, N[limit/5]};
framex = gridx;
sizes = Flatten[
  Table[
    Dimensions[
      Characters[
        ToString[framex[[a]]
      ]], {a, 1, 6}];
Do[
  Do[framex[[a]] = StringJoin[ToString[framex[[a]]], " "],
    {b, 1, size - sizes[[a]]},
    {a, 1, 6}];
framex = Table[{gridx[[a]], framex[[a]]}, {a, 1, 6}];
gridx = Table[{gridx[[a]], RGBColor[0, 0, 0]}, {a, 1, 6}];
framey = Table[{a + 1, Dimensions[eigenexpressions][[1]] - a},
  {a, 0, Dimensions[eigenexpressions][[1]] - 1}];
labelx = ColumnForm[
  {"(b) Eigenexpression Fraction", StringJoin["d2 = ", ToString[entropy]], " "},
  Center];
g = BarChart[
  Table[fractions[[Dimensions[eigenexpressions][[1]] - a]],
    {a, 0, Dimensions[eigenexpressions][[1]] - 1}],
  BarOrientation -> Horizontal,
  PlotRange -> {{0, alsolimit*1.0001}, {0.5, Dimensions[eigenexpressions][[1]] + 0.5}},
  AspectRatio -> 1,
  Axes -> False,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, None, labelx, None},
  GridLines -> {gridx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 2.25}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 1.5}, {0, 0}, {0, 1}];
g2 = Show[g,
  AspectRatio -> 1.35 * 14 / 18,
  PlotRange -> All,
  DisplayFunction -> Identity];

```



```
(* Create Eigenenes 2D Red & Green Raster Display *)
```

```
contrast = 3.5;
displaying = Table[
  If[contrast * eigenenes[[i, j]] > 0,
    If[contrast * eigenenes[[i, j]] < 1, {contrast * eigenenes[[i, j]], 0}, {1, 0}],
    If[contrast * eigenenes[[i, j]] > -1, {0, -contrast * eigenenes[[i, j]]}, {0, 1}]],
  {i, 1, Dimensions[eigenexpressions][[1]]}, {j, 1, arrays}];
framex = Table[{a - 0.5, arraynames[[1, a]]}, {a, 1, arrays}];
framey = Table[{a + 1 - 0.5, Dimensions[eigenexpressions][[1]] - a},
  {a, 0, Dimensions[eigenexpressions][[1]] - 1}];
labely = "Eigenenes";
labelx = ColumnForm[{"(a) Arrays", " ", " "}, Center];

g = Show[
  Graphics[
    RasterArray[
      Table[
        RGBColor[displaying[[i, j, 1]], displaying[[i, j, 2]], 0],
        {i, Dimensions[eigenexpressions][[1]], 1, -1}, {j, 1, arrays}]]],
  AspectRatio -> 1,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, labely, labelx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 3, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 2.25}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 2.25}, {0, 0}, {0, 1}];
g1 = Show[g,
  AspectRatio -> 1.05 * 14 / 18,
  PlotRange -> All,
  DisplayFunction -> Identity];
```

(* Create Selected Eigengenes Graph Display *)

```
eigengenes1 = Chop[TrigFit[Drop[eigengenes[[1]], {1}], 2, {x - 1, arrays - 1}], 0.15]
eigengenes2 = Chop[TrigFit[Drop[eigengenes[[2]], {1}], 2, {x - 1, arrays - 1}], 0.15]
```

$$0.29995 \sin\left[\frac{4}{17} \pi (-1 + x)\right]$$

$$0.242079 \cos\left[\frac{4}{17} \pi (-1 + x)\right]$$

```
eigengenes1 = Sqrt[2 / 17.] * Sin[4 * Pi * (x - 1) / 17];
eigengenes2 = Sqrt[2 / 17.] * Cos[4 * Pi * (x - 1) / 17];
```

```
graph = Plot[eigengenes1,
  {x, 1, arrays - 1},
  PlotStyle -> {RGBColor[1, 0, 0], Dashing[{0.03, 0.02}]},
  DisplayFunction -> Identity];
labelx = ColumnForm[{"(c) Arrays"}, Center];
labely = ColumnForm[{" ", "Expression Level"}, Center];
framex = Table[{a - 1, arraynames[[1, a]]}, {a, 1, arrays}];
framey = {-1, -0.5, 0, 0.5, 1};
coordinates = Table[{a - 1, eigengenes[[1, a]]}, {a, 1, arrays}];
points = Table[Point[coordinates[[a]]], {a, 1, arrays}];
line = Line[coordinates];
g = Show[
  {Graphics[{RGBColor[1, 0, 0], PointSize[0.022], points}],
  Graphics[{RGBColor[1, 0, 0], line}],
  graph,
  Graphics[{RGBColor[1, 0, 0], Text[" $\sqrt{\frac{2}{T}} \sin\left(\frac{4 \pi t}{T}\right)$ ", {8.5, 0.7}]}]},
  Frame -> True,
  FrameLabel -> {None, labely, labelx, None},
  GridLines -> {None, {{0, RGBColor[0, 0, 0]}}},
  FrameTicks -> {None, framey, framex, None},
  PlotRange -> {-1.05, 1.05},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 5.4, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 0.625}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 0.25}, {0, 0}, {0, 1}];
p1 = Show[g,
  AspectRatio -> 1.05,
  PlotRange -> All, DisplayFunction -> Identity];
```

```

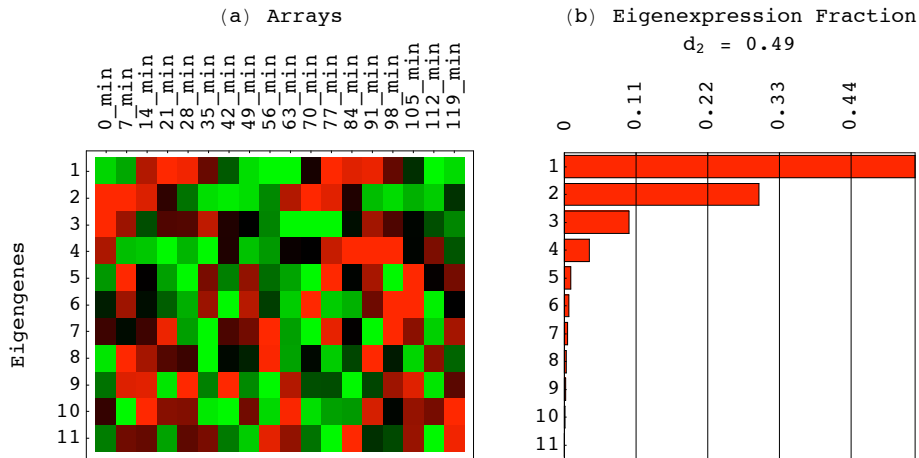
graph = Plot[eigengenes2,
  {x, 1, arrays - 1},
  PlotStyle -> {RGBColor[0, 0, 1], Dashing[{0.03, 0.02}]},
  DisplayFunction -> Identity];
labelx = ColumnForm[{"(d) Arrays"}, Center];
labely = ColumnForm[{" ", " "}, Center];
framex = Table[{a - 1, arraynames[[1, a]]}, {a, 1, arrays}];
framey = {-1, -0.5, 0, 0.5, 1};
coordinates = Table[{a - 1, eigengenes[[2, a]]}, {a, 1, arrays}];
points = Table[Point[coordinates[[a]]], {a, 1, arrays}];
line = Line[coordinates];
g = Show[
  {Graphics[{RGBColor[0, 0, 1], PointSize[0.022], points}],
  Graphics[{RGBColor[0, 0, 1], line}],
  graph,

  Graphics[{RGBColor[0, 0, 1], Text[" $\sqrt{\frac{2}{T}} \cos(\frac{4 \pi t}{T})$ ", {8.5, 0.7}]}]},
  Frame -> True,
  FrameLabel -> {None, labely, labelx, None},
  GridLines -> {None, {{0, RGBColor[0, 0, 0]}}},
  FrameTicks -> {None, framey, framex, None},
  PlotRange -> {-1.05, 1.05},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 5.4, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 0.625}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 0.25}, {0, 0}, {0, 1}];
p2 = Show[g,
  AspectRatio -> 1.05,
  PlotRange -> All,
  DisplayFunction -> Identity];

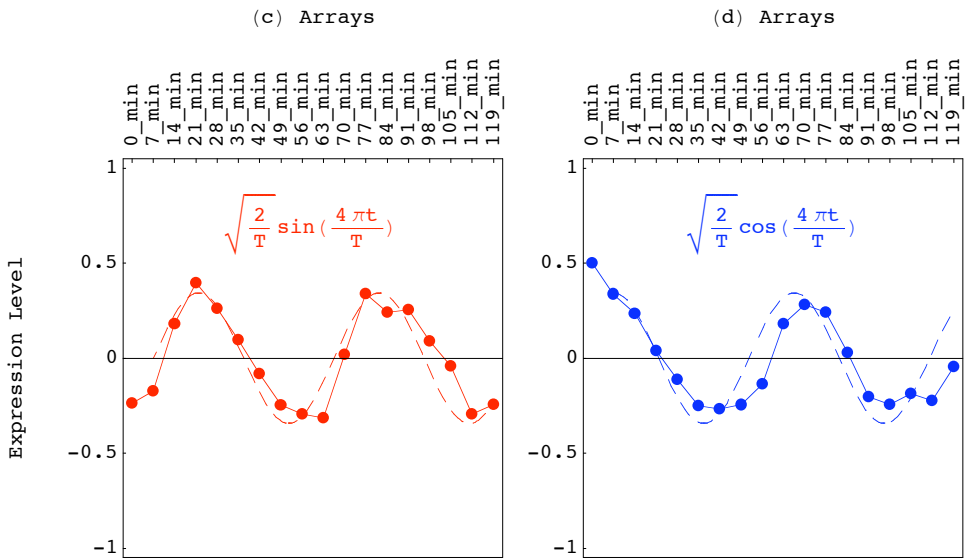
```

(* Display Eigengenes, Fractions and Selected Eigengenes *)

```
Show[GraphicsArray[{g1, g2}],
GraphicsSpacing -> -0.15];
```



```
Show[GraphicsArray[{p1, p2}],
GraphicsSpacing -> -0.15];
```



(* Define Projected Data *)

```
partialmatrix2 = partialmatrix;
partialgenenames2 = partialgenenames;
```

```
(* Display Sorted Significant Eigenarrays *)
```

```
arraypatterns = Transpose[eigenarrays];
```

```
(* Center Eigenarrays *)
```

```
average = Table[1, {a, 1, genes}];
```

```
average = N[average / Sqrt[Dot[average, average]]];
```

```
arraypatterns = arraypatterns - N[Outer[Times, Dot[arraypatterns, average], average]];
```

```
(* Sort Eigenarrays *)
```

```
Do[
```

```
arraypatterns[[a]] = Sort[arraypatterns[[a]], OrderedQ[{{#2}, {#1}}] &],
```

```
{a, 1, Dimensions[eigenexpressions][[1]]}]
```

```
(* Create Sorted Eigenarrays Graph Display *)
```

```
p = Table[0, {a, 1, 2}];
```

```
color = {
```

```
  RGBColor[1, 0.5, 0],
```

```
  RGBColor[1, 0, 0],
```

```
  RGBColor[0, 0, 1],
```

```
  RGBColor[0, 0.5, 0]};
```

```
labelx = "Expression Level";
```

```
labeled = ColumnForm[
```

```
  {"Number of Genes", " ", " ", " ", " ", " ", " ", " ", " ", " ", " "},
```

```
  Center];
```

```
framex = Table[{0.04 * a, a}, {a, 1, 2}];
```

```
framey = {{-150, "150"}, {-750, "750"}, {-genes + 150, "1438"}};
```

```
Do[ {
```

```
  coordinates = Table[
```

```
    If[arraypatterns[[n, a]] + 0.04 * n < -0.08, -0.08,
```

```
    If[arraypatterns[[n, a]] + 0.04 * n > 0.2, 0.2,
```

```
    arraypatterns[[n, a]] + 0.04 * n]],
```

```
    {a, 1, genes}],
```

```
  coordinates = Table[{coordinates[[a]], -a + 1}, {a, 1, genes}],
```

```
  line = Line[coordinates],
```

```
  g = Show[
```

```
    Graphics[{color[[Mod[n, 4] + 1]], line}],
```

```
    Frame -> True,
```

```
    FrameLabel -> {None, labeled, labelx, None},
```

```
    FrameTicks -> {None, framey, framex, None},
```

```
    GridLines -> {{{0.04 * n, RGBColor[0, 0, 0]}}, {{-150, RGBColor[0, 0, 0]},
```

```
    {-750, RGBColor[0, 0, 0]}, {-genes + 150, RGBColor[0, 0, 0]}},
```

```
    PlotRange -> {{-0.08, 0.2}, {57.5, -genes + 1 - 57.5}},
```

```
    DisplayFunction -> Identity],
```

```
  g = FullGraphics[g],
```

```
  g[[1, 2]] = g[[1, 2]] /.
```

```
    Text[labeled, {b_, c_}, {1., 0.}] ->
```

```
    Text[labeled, {b, c}, {0, 0}, {0, 1}],
```

```
  g[[1, 2]] = g[[1, 2]] /.
```

```
    Text[labelx, {b_, c_}, {0., -1.}] ->
```

```
    Text[labelx, {b, c + 152.5}, {0, -1}, {1, 0}],
```

```
  p[[n]] = Show[g,
```

```
    AspectRatio -> 2 / 1.2 / GoldenRatio,
```

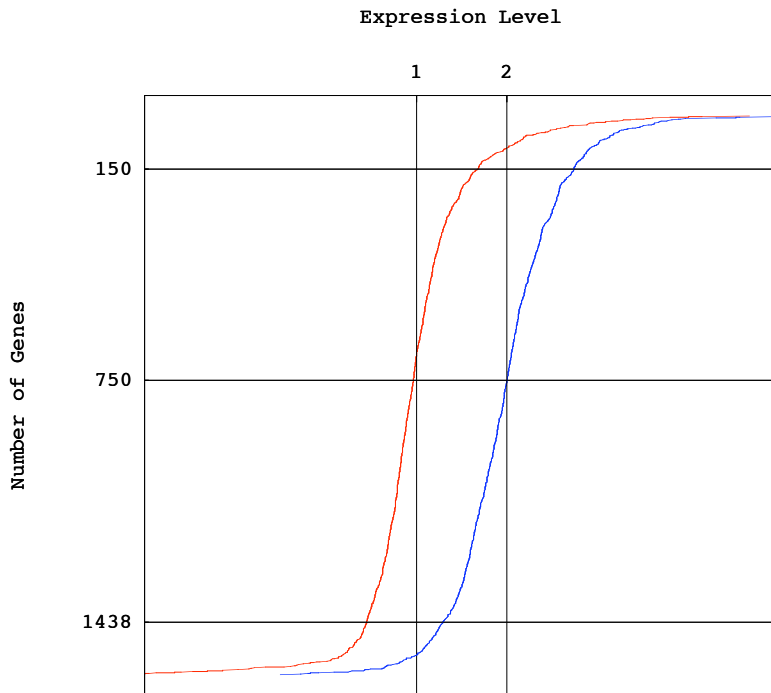
```
    PlotRange -> All,
```

```
    DisplayFunction -> Identity]
```

```
}, {n, 1, 2}];
```

(* Display Sorted Eigenarrays *)

```
Show[Table[p[[a]], {a, 1, 2}],  
      DisplayFunction -> $DisplayFunction];
```



(* Estimate Significance of Association of Eigenarrays with the Cell Cycle *)

(* Use Cell Cycle Traditional or Microarray Classification of Yeast Genes *)

```
most = 150;
annotations = TakeColumns[partialgenenames, {6}];
stages = {"M/G1", "G1", "S", "S/G2", "G2/M", "None"};
numbers = Flatten[Table[{Count[Flatten[annotations], stages[[a]]}], {a, 1, Dimensions[stages][[1]]}]]
counter = Table[{a}, {a, 1, 2}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];

{48, 125, 25, 40, 60, 1290}

Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenarrays, counter[[c]]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  {c, 1, Dimensions[counter][[1]]}]

table1 = AppendRows[
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];
```

(* Estimate Significance of Association of Eigenarrays with the Pheromone Response *)

(* Use Pheromone Microarray Classification of Yeast Genes *)

```
most = 150;
annotations = TakeColumns[partialgenenames, {7}];
stages = {"Up", "Down", "None"};
numbers = Flatten[Table[{Count[Flatten[annotations], stages[[a]]], {a, 1, Dimensions[stages][[1]]}]]
counter = Table[{a}, {a, 1, 2}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];

{53, 87, 1448}

Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenarrays, counter[[c]]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  {c, 1, Dimensions[counter][[1]]}]

table2 = AppendRows[
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];
```


(* Display Significance of Association of Eigenarrays with the Cellular Programs *)

```

headerx = {{
  ColumnForm[{" ", " ", " "}, Left],
  ColumnForm[{" ", " ", "Classification"}, Left],
  ColumnForm[{" ", " ", "Eigenarray"}, Left],
  ColumnForm[{"Most Likely", "Parallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Parallel", "Association"}, Left],
  ColumnForm[{"Most Likely", "Antiparallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Antiparallel", "Association"}, Left]},
{" ", " ", " ", " ", " ", " ", " ", " ", " ", " "}};
spacerx = {" ", " ", " ", " ", " "};
headery = Table[" ", {a, 1, 2 * Dimensions[counter][[1]] + 1}, {b, 1, 2}];
headery[[1]] = {"(a)", "Cell Cycle"};
headery[[Dimensions[counter][[1]] + 2]] = {"(b)", "Pheromone Response"};
association =
  AppendColumns[headerx,
  AppendRows[headery,
  AppendColumns[table1, spacerx, table2]]];
TableForm[association, TableSpacing -> {1, 1}]

```

	Classification	Eigenarray	Most Likely Parallel Association	P-Value of Parallel Association	Most Likely Antiparallel Association	P-Value of Antiparallel Association
(a)	Cell Cycle	1	G1	9.4×10^{-19}	G2/M	1.4×10^{-15}
		2	M/G1	2.2×10^{-12}	G2/M	1.5×10^{-7}
(b)	Pheromone Response	1	Down	$1. \times 10^{-14}$	Up	2.3×10^{-1}
		2	Up	2.3×10^{-4}	Down	1.7×10^{-17}

```
(* Examine the Cell Cycle-Projected Alpha Factor Genes x Genes Correlations Eigenmatrices *)
```

```
(* Read List of Cell Cycle-Regulated Genes *)
```

```
stream = "Desktop/Network_Decomposition/Data/Cycle_Genes.txt";  
genelist = Import[stream, "Table"];  
genelist = Drop[genelist, 1];  
partialgenes = Dimensions[Intersection[genelist, partialgenenames2]][[1]]  
Clear[stream];
```

48

```
list = Flatten[TakeColumns[genelist, {1}]];  
counter = Table[Flatten[Position[list, partialgenenames2[[a, 1]]]],  
  {a, 1, Dimensions[partialgenenames2][[1]]}];  
counter = ReplaceAll[counter, {} -> {Null}];  
partial = AppendRows[counter, partialgenenames2, eigenarrays];  
partial = Sort[partial, OrderedQ[{{#1, #2}} &];  
partialgenenames = TakeRows[  
  TakeColumns[partial, {2, 8}],  
  {1, partialgenes}];  
partialeigenarrays = TakeRows[  
  TakeColumns[partial, {9, Dimensions[eigenexpressions][[1]] + 8}],  
  {1, partialgenes}];
```

```
(* Calculate Eigenmatrices *)
```

```
partialeigenarrays = Transpose[partialeigenarrays];  
eigenmatrices = Table[0, {a, 1, Dimensions[eigenexpressions][[1]]}];  
Do[  
  eigenmatrices[[a]] = Outer[Times, partialeigenarrays[[a]], partialeigenarrays[[a]],  
    {a, 1, Dimensions[eigenexpressions][[1]]}];  
partialeigenarrays = Transpose[partialeigenarrays];
```

```
(* Keep Only Correlations Above Diagonals in Eigenmatrices *)
```

```
Do[  
  Do[  
    Do[  
      eigenmatrices[[a, b, c]] = 0,  
      {c, 1, b}],  
    {b, 1, Dimensions[eigenmatrices[[1]][[1]]}],  
  {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
(* Flatten Eigenmatrices Into Eigenvectors *)
```

```
eigenvectors = Table[0, {partialgenes * (partialgenes - 1) / 2, {Dimensions[eigenexpressions][[1]]}];  
Do[{  
  n = 0,  
  square = eigenmatrices[[a]],  
  Do[{  
    line = square[[b]],  
    Do[{  
      n = n + 1,  
      eigenvectors[[n, a]] = line[[c]]  
    }, {c, b + 1, partialgenes}],  
  }, {b, 1, partialgenes}],  
  {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
(* Create Genes × Genes Annotation Matrices for Selected Genes *)
```

```
partialgenenames = Transpose[partialgenenames];  
annotatematrices = Table[  
  Table[0, {partialgenes}, {partialgenes}],  
  {a, 1, 7}];  
Do[  
  annotatematrices[[a]] = Outer[Times, partialgenenames[[a]], partialgenenames[[a]],  
  {a, 1, 7}]
```

```
(* Keep Only Correlations Above Diagonals in Annotation Matrices *)
```

```
Do[  
  Do[  
    Do[  
      annotatematrices[[a, b, c]] = 0,  
      {c, 1, b}],  
    {b, 1, partialgenes}],  
  {a, 1, 7}];
```

```
(* Flatten Annotation Matrices Into Annotation Vectors *)
```

```
annotatevectors = Table[0, {partialgenes * (partialgenes - 1) / 2}, {7}];  
Do[{  
  n = 0,  
  square = annotatematrices[[a]],  
  Do[{  
    line = square[[b]],  
    Do[{  
      n = n + 1,  
      annotatevectors[[n, a]] = line[[c]]  
    }, {c, b + 1, partialgenes}],  
  }, {b, 1, partialgenes}]  
}, {a, 1, 7}]  
partialgenenames = Transpose[partialgenenames];
```

```
(* Display Sorted Significant Eigenmatrices *)
```

```
arraypatterns = Transpose[eigenvectors];  
genes = Dimensions[arraypatterns][[2]]
```

```
1128
```

```
(* Center Eigenmatrices *)
```

```
average = Table[1, {a, 1, genes}];  
average = N[average / Sqrt[Dot[average, average]]];  
arraypatterns = arraypatterns - N[Outer[Times, Dot[arraypatterns, average], average]];
```

```
(* Sort Eigenmatrices *)
```

```
Do[  
  arraypatterns[[a]] = Sort[arraypatterns[[a]], OrderedQ[{{#2}, {#1}}] &],  
  {a, 1, Dimensions[eigenexpressions][[1]]}]
```

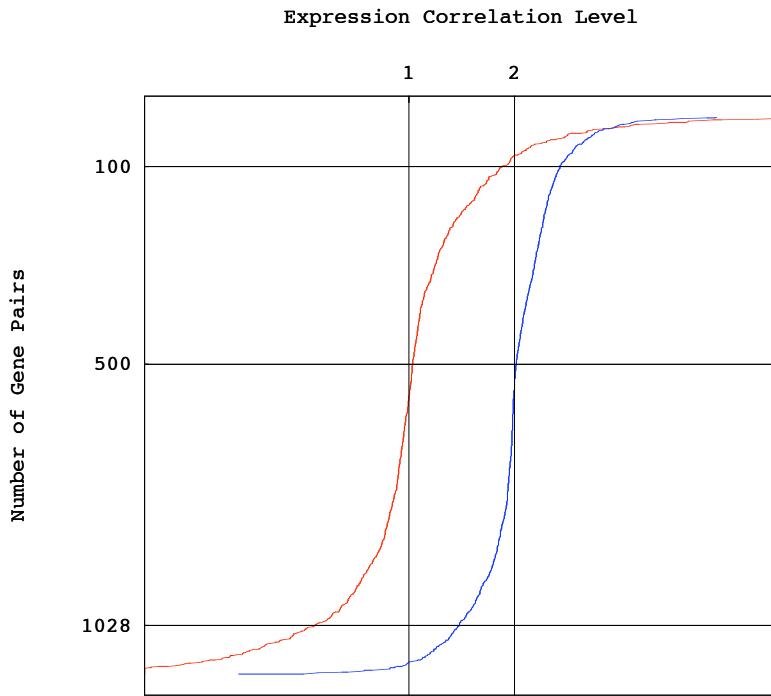
```
(* Create Sorted Eigenmatrices Graph Display *)
```

```
p = Table[0, {a, 1, 2}];  
color = {  
  RGBColor[1, 0.5, 0],  
  RGBColor[1, 0, 0],  
  RGBColor[0, 0, 1],  
  RGBColor[0, 0.5, 0]};  
labelx = "Expression Correlation Level";  
labely = ColumnForm[  
  {"Number of Gene Pairs", " ", " ", " ", " ", " ", " ", " ", " ", " ", " "},  
  Center];  
framex = Table[{0.004 * a, a}, {a, 1, 2}];  
framey = {{-100, "100"}, {-500, "500"}, {-genes + 100, "1028"}};
```

```
Do[{  
  coordinates = Table[  
    If[arraypatterns[[n, a]] + 0.004 * n < -0.006, -0.006,  
      If[arraypatterns[[n, a]] + 0.004 * n > 0.018, 0.018,  
        arraypatterns[[n, a]] + 0.004 * n]],  
    {a, 1, genes}],  
  coordinates = Table[{coordinates[[a]], -a + 1}, {a, 1, genes}],  
  line = Line[coordinates],  
  g = Show[  
    Graphics[{color[[Mod[n, 4] + 1]], line}],  
    Frame -> True,  
    FrameLabel -> {None, labely, labelx, None},  
    FrameTicks -> {None, framey, framex, None},  
    GridLines -> {{{0.004 * n, RGBColor[0, 0, 0]}}, {{-100, RGBColor[0, 0, 0]},  
      {-500, RGBColor[0, 0, 0]}, {-genes + 100, RGBColor[0, 0, 0]}}},  
    PlotRange -> {{-0.006, 0.018}, {42.5, -genes + 1 - 42.5}},  
    DisplayFunction -> Identity],  
  g = FullGraphics[g],  
  g[[1, 2]] = g[[1, 2]] /.  
    Text[labely, {b_, c_}, {1., 0.}] ->  
    Text[labely, {b, c}, {0, 0}, {0, 1}],  
  g[[1, 2]] = g[[1, 2]] /.  
    Text[labelx, {b_, c_}, {0., -1.}] ->  
    Text[labelx, {b, c + 110}, {0, -1}, {1, 0}],  
  p[[n]] = Show[g,  
    AspectRatio -> 2 / 1.2 / GoldenRatio,  
    PlotRange -> All,  
    DisplayFunction -> Identity]  
}, {n, 1, 2}];
```

(* Display Sorted Eigenmatrices *)

```
Show[Table[p[[a]], {a, 1, 2}],  
  DisplayFunction -> $DisplayFunction];
```



(* Estimate Significance of the Association of Eigenmatrices with the Cell Cycle *)

(* Use Cell Cycle Traditional or Microarray Classification of Yeast Genes *)

```
stages = {"M/G1", "G1", "S", "S/G2", "G2/M"};
stagematrix = Outer[Times, stages, stages];
stagevector =
  Table[0, {Dimensions[stages][[1]] * (Dimensions[stages][[1]] + 1) / 2}];
n = 0;
Do[
  Do[{
    n = n + 1,
    stagevector[[n]] = stagematrix[[a, b]]
  }, {b, 1, a}],
  {a, 1, Dimensions[stages][[1]]}];

most = 100;
annotations = TakeColumns[annotatevectors, {6}];
numbers = Flatten[Table[
  {Count[Flatten[annotations], stagevector[[a]]],
  {a, 1, Dimensions[stagevector][[1]]}]]];
counter = Table[{a}, {a, 1, 2}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];

{36, 198, 231, 36, 88, 6, 45, 110, 20, 10, 72, 176, 32, 40, 28}

Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenvectors, counter[[c]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
    Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}]], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
    Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  {c, 1, Dimensions[counter][[1]]}];
```

```

table1 = AppendRows [
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];

table1 = ReplaceAll[table1, "M/G1"2 -> "M/G1 M/G1"];
table1 = ReplaceAll[table1, "G1" "M/G1" -> "G1 M/G1"];
table1 = ReplaceAll[table1, "G1"2 -> "G1 G1"];
table1 = ReplaceAll[table1, "M/G1" "S" -> "M/G1 S"];
table1 = ReplaceAll[table1, "G1" "S" -> "G1 S"];
table1 = ReplaceAll[table1, "S"2 -> "S S"];
table1 = ReplaceAll[table1, "M/G1" "S/G2" -> "M/G1 S/G2"];
table1 = ReplaceAll[table1, "G1" "S/G2" -> "G1 S/G2"];
table1 = ReplaceAll[table1, "S" "S/G2" -> "S S/G2"];
table1 = ReplaceAll[table1, "S/G2"2 -> "S/G2 S/G2"];
table1 = ReplaceAll[table1, "G2/M" "M/G1" -> "G2/M M/G1"];
table1 = ReplaceAll[table1, "G1" "G2/M" -> "G1 G2/M"];
table1 = ReplaceAll[table1, "G2/M" "S" -> "G2/M S"];
table1 = ReplaceAll[table1, "G2/M" "S/G2" -> "G2/M S/G2"];
table1 = ReplaceAll[table1, "G2/M"2 -> "G2/M G2/M"];

```

(* Estimate Significance of the Association of Eigenmatrices with the Pheromone Response *)

(* Use Pheromone Microarray Classification of Yeast Genes *)

```
stages = {"Up", "Down", "None"};
stagematrix = Outer[Times, stages, stages];
stagevector =
  Table[0, {Dimensions[stages][[1]] * (Dimensions[stages][[1]] + 1) / 2}];
n = 0;
Do[
  Do[{
    n = n + 1,
    stagevector[[n]] = stagematrix[[a, b]]
  }, {b, 1, a}],
  {a, 1, Dimensions[stages][[1]]}];

most = 100;
annotations = TakeColumns[annotatevectors, {7}];
numbers = Flatten[Table[
  {Count[Flatten[annotations], stagevector[[a]]],
  {a, 1, Dimensions[stagevector][[1]]}]]];
counter = Table[{a}, {a, 1, 2}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];

{36, 180, 190, 171, 380, 171}

Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenvectors, counter[[c]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
    Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}]], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
    Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  {c, 1, Dimensions[counter][[1]]}];
```



```
table2 = AppendRows [
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];

table2 = ReplaceAll[table2, "Up"2 -> "Up  Up"];
table2 = ReplaceAll[table2, "Down" "Up" -> "Down Up"];
table2 = ReplaceAll[table2, "Down"2 -> "Down Down"];
table2 = ReplaceAll[table2, "None" "Up" -> "None Up"];
table2 = ReplaceAll[table2, "Down" "None" -> "Down None"];
table2 = ReplaceAll[table2, "None"2 -> "None None"];
```

(* Display Significance of Association of Eigenmatrices with the Cellular Programs *)

```

headerx = {{
  ColumnForm[{" ", " ", " "}, Left],
  ColumnForm[{" ", " ", "Classification"}, Left],
  ColumnForm[{" ", " ", "Subnetwork"}, Left],
  ColumnForm[{"Most Likely", "Parallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Parallel", "Association"}, Left],
  ColumnForm[{"Most Likely", "Antiparallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Antiparallel", "Association"}, Left]},
{" ", " ", " ", " ", " ", " ", " ", " ", " ", " "}};
spacerx = {" ", " ", " ", " ", " "};
headery = Table[" ", {a, 1, 2 * Dimensions[counter][[1]] + 1}, {b, 1, 2}];
headery[[1]] = {"(a)", "Cell Cycle"};
headery[[Dimensions[counter][[1]] + 2]] = {"(b)", "Pheromone Response"};
association =
  AppendColumns[headerx,
  AppendRows[headery,
  AppendColumns[table1, spacerx, table2]]];
TableForm[association, TableSpacing -> {1, 1}]

```

	Classification	Subnetwork	Most Likely Parallel Association	P-Value of Parallel Association	Most Likely Antiparallel Association	P-Value of Antiparallel Association
(a)	Cell Cycle	1	G1 G1	1.3×10^{-9}	G1 G2/M	3.7×10^{-23}
		2	M/G1 M/G1	4.8×10^{-12}	G2/M M/G1	3.3×10^{-14}
(b)	Pheromone Response	1	Down Down	6.8×10^{-5}	Down None	4.3×10^{-2}
		2	Up Up	2.5×10^{-5}	Down Up	1.6×10^{-15}

```
(* Display the Cell Cycle-Projected Alpha Factor Genes x Genes Correlations Eigenmatrices *)
```

```
(* Read List of Cell Cycle-Regulated Genes *)
```

```
stream = "Desktop/Network_Decomposition/Data/Cycle_Genes.txt";  
genelist = Import[stream, "Table"];  
genelist = Drop[genelist, 1];  
partialgenes = Dimensions[Intersection[genelist, partialgenenames2]][[1]]  
Clear[stream];
```

48

```
list = Flatten[TakeColumns[genelist, {1}]];  
counter = Table[Flatten[Position[list, partialgenenames2[[a, 1]]]],  
  {a, 1, Dimensions[partialgenenames2][[1]]}];  
counter = ReplaceAll[counter, {} -> {Null}];  
partial = AppendRows[counter, partialgenenames2, eigenarrays, matrix,  
  partialexternalmatrix, originalpartial];  
partial = Sort[partial, OrderedQ[{{#1, #2}} &];  
partialgenenames = TakeRows[  
  TakeColumns[partial, {2, 8}],  
  {1, partialgenes}];  
partialeigenarrays = TakeRows[  
  TakeColumns[partial, {9, Dimensions[eigenexpressions][[1]] + 8}],  
  {1, partialgenes}];  
partialmatrix = TakeRows[  
  TakeColumns[partial,  
    {Dimensions[eigenexpressions][[1]] + 9,  
    Dimensions[eigenexpressions][[1]] + arrays + 8}],  
  {1, partialgenes}];  
partialexternalmatrix = TakeRows[  
  TakeColumns[partial,  
    {Dimensions[eigenexpressions][[1]] + arrays + 9,  
    Dimensions[eigenexpressions][[1]] + arrays + arrays2 + 8}],  
  {1, partialgenes}];  
originalpartial = TakeRows[  
  TakeColumns[partial,  
    {Dimensions[eigenexpressions][[1]] + arrays + arrays2 + 9,  
    Dimensions[eigenexpressions][[1]] + 2 * arrays + arrays2 + 8}],  
  {1, partialgenes}];
```

```
(* Calculate Eigenmatrices *)
```

```
partialeigenarrays = Transpose[partialeigenarrays];  
eigenmatrices = Table[0, {a, 1, Dimensions[eigenexpressions][[1]]}];  
Do[  
  eigenmatrices[[a]] = Outer[Times, partialeigenarrays[[a]], partialeigenarrays[[a]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}];  
partialeigenarrays = Transpose[partialeigenarrays];
```

```
(* Keep Only Correlations Above Diagonals in Eigenmatrices *)
```

```
Do[  
  Do[  
    Do[  
      eigenmatrices[[a, b, c]] = 0,  
      {c, 1, b}],  
    {b, 1, Dimensions[eigenmatrices][[1]][[1]]},  
  {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
(* Set Correlations Cutoffs *)
```

```
cutoffs = Table[Sort[Flatten[Abs[eigenmatrices[[a]]], OrderedQ[#{#2, #1}] &][[200]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
Do[  
  eigenmatrices[[a]] = Chop[eigenmatrices[[a]], cutoffs[[a]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}]
```

```
(* Select Cell Cycle-Regulated Genes With Correlations Above Cutoffs *)
```

```
int = Intersection[  
  Position[Sign[eigenmatrices[[1]]], Table[0, {a, 1, partialgenes}]],  
  Position[Sign[eigenmatrices[[2]]], Table[0, {a, 1, partialgenes}]]];
```

```
Do[{  
  partialmatrix = Drop[partialmatrix, {Flatten[int][[a]]}],  
  partialexternalmatrix = Drop[partialexternalmatrix, {Flatten[int][[a]]}],  
  originalpartial = Drop[originalpartial, {Flatten[int][[a]]}],  
  partialeigenarrays = Drop[partialeigenarrays, {Flatten[int][[a]]}],  
  partialgenenames = Drop[partialgenenames, {Flatten[int][[a]]}],  
  {a, Dimensions[int][[1]], 1, -1}]  
partialgenes = Dimensions[partialgenenames][[1]]
```

39

```
(* Calculate Eigenmatrices For Selected Genes *)
```

```
partialeigenarrays = Transpose[partialeigenarrays];  
eigenmatrices = Table[0, {a, 1, Dimensions[eigenexpressions][[1]]}];  
Do[  
  eigenmatrices[[a]] = Outer[Times, partialeigenarrays[[a]], partialeigenarrays[[a]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}]  
partialeigenarrays = Transpose[partialeigenarrays];
```

```
(* Create Network Decomposition 2D Red & Green Raster Display *)
```

```
(* Create Genes x Genes Correlations 2D Red & Green Raster Display *)
```

```
framex = Table[{a - 0.5, partialgenenames[[a, 2]]}, {a, 1, partialgenes}];  
framey = Table[{a + 1 - 0.5, partialgenes - a}, {a, 0, partialgenes - 1}];  
labelx = "Genes";  
labely = ColumnForm[{StyleForm["", FontSize -> 200], "Genes"}, Center];  
labelz = "Genes";  
p = Table[0, {a, 1, 4}];  
  
contrast = 10;  
labelz = ColumnForm[{StyleForm[" ", FontSize -> 40, FontWeight -> Bold]}, Left];  
Do[{  
  correlation = Dot[partialexternalmatrix, PseudoInverse[partialexternalmatrix]],  
  displaying = Table[  
    If[contrast * correlation[[i, j]] > 0,  
      If[contrast * correlation[[i, j]] < 1, {contrast * correlation[[i, j]], 0}, {1, 0}],  
      If[contrast * correlation[[i, j]] > -1, {0, -contrast * correlation[[i, j]]}, {0, 1}]],  
    {i, 1, partialgenes}, {j, 1, partialgenes}],  
  g = Show[  
    Graphics[  
      RasterArray[  
        Table[  
          RGBColor[displaying[[i, j, 1]], displaying[[i, j, 2]], 0],  
          {i, partialgenes, 1, -1}, {j, 1, partialgenes}]]],  
      AspectRatio -> 1,  
      Frame -> True,  
      FrameTicks -> {None, None, None, None},  
      FrameLabel -> {None, labely, labelx, labelz},  
      DisplayFunction -> Identity],  
    g = FullGraphics[g],  
    g[[1, 2]] = g[[1, 2]] /.  
      Text[labely, {b_, c_}, {1., 0.}] ->  
        Text[labely, {b - 4, c}, {0, 0}, {0, 1}],  
    g[[1, 2]] = g[[1, 2]] /.  
      Text[labelx, {b_, c_}, {0., -1.}] ->  
        Text[labelx, {b, c}, {0, -1}, {1, 0}],  
    p[[a]] = Show[g,  
      AspectRatio -> 0.95,  
      PlotRange -> All,  
      DisplayFunction -> Identity}],  
  {a, 1, 1}]
```

```

contrast = 1;
labelz = ColumnForm[{StyleForm[" ", FontSize -> 40, FontWeight -> Bold]}, Left];
Do[{
  correlation = Dot[originalpartial, Transpose[originalpartial]],
  displaying = Table[
    If[contrast * correlation[[i, j]] > 0,
      If[contrast * correlation[[i, j]] < 1, {contrast * correlation[[i, j]], 0}, {1, 0}],
      If[contrast * correlation[[i, j]] > -1, {0, -contrast * correlation[[i, j]]}, {0, 1}]],
    {i, 1, partialgenes}, {j, 1, partialgenes}],
  g = Show[
    Graphics[
      RasterArray[
        Table[
          RGBColor[displaying[[i, j, 1]], displaying[[i, j, 2]], 0],
          {i, partialgenes, 1, -1}, {j, 1, partialgenes}]]],
      AspectRatio -> 1,
      Frame -> True,
      FrameTicks -> {None, None, None, None},
      FrameLabel -> {None, labely, labelx, labelz},
      DisplayFunction -> Identity],
  g = FullGraphics[g],
  g[[1, 2]] = g[[1, 2]] /.
    Text[labely, {b_, c_}, {1., 0.}] ->
      Text[labely, {b - 4, c}, {0, 0}, {0, 1}],
  g[[1, 2]] = g[[1, 2]] /.
    Text[labelx, {b_, c_}, {0., -1.}] ->
      Text[labelx, {b, c}, {0, -1}, {1, 0}],
  p[[a]] = Show[g,
    AspectRatio -> 0.95,
    PlotRange -> All,
    DisplayFunction -> Identity]],
  {a, 2, 2}]

```

(* Create Genes × Genes Eigencorrelations 2 D Red & Green Raster Displays *)

```

contrast = 1000;
labelz = ColumnForm[
  {" ", " ", " ", " ", " ", " ", " ", " ", StyleForm["", FontSize → 40, FontWeight → Bold]},
  Left];
Do[{
  correlation = eigenmatrices[[a - 2]],
  displaying = Table[
    If[contrast * correlation[[i, j]] > 0,
      If[contrast * correlation[[i, j]] < 1, {contrast * correlation[[i, j]], 0}, {1, 0}],
      If[contrast * correlation[[i, j]] > -1, {0, -contrast * correlation[[i, j]]}, {0, 1}]],
    {i, 1, partialgenes}, {j, 1, partialgenes}],
  g = Show[
    Graphics[
      RasterArray[
        Table[
          RGBColor[displaying[[i, j, 1]], displaying[[i, j, 2]], 0],
          {i, partialgenes, 1, -1}, {j, 1, partialgenes}]]],
    AspectRatio → 1,
    Frame → True,
    FrameTicks → {None, None, None, None},
    FrameLabel → {None, labely, labelx, labelz},
    DisplayFunction → Identity],
  g = FullGraphics[g],
  g[[1, 2]] = g[[1, 2]] /.
    Text[labely, {b_, c_}, {1., 0.}] →
    Text[labely, {b - 4, c}, {0, 0}, {0, 1}],
  g[[1, 2]] = g[[1, 2]] /.
    Text[labelx, {b_, c_}, {0., -1.}] →
    Text[labelx, {b, c}, {0, -1}, {1, 0}],
  p[[a]] = Show[g,
    AspectRatio → 0.95,
    PlotRange → All,
    DisplayFunction → Identity]],
  {a, 3, 3}]

```

```

contrast = 1000;
labelz = ColumnForm[
  {" ", " ", " ", " ", " ", " ", " ", " ", " ", StyleForm[" ", FontSize -> 40, FontWeight -> Bold]},
  Left];
Do[{
  correlation = eigenmatrices[[a - 2]],
  displaying = Table[
    If[contrast * correlation[[i, j]] > 0,
      If[contrast * correlation[[i, j]] < 1, {contrast * correlation[[i, j]], 0}, {1, 0}],
      If[contrast * correlation[[i, j]] > -1, {0, -contrast * correlation[[i, j]]}, {0, 1}]],
    {i, 1, partialgenes}, {j, 1, partialgenes}],
  g = Show[
    Graphics[
      RasterArray[
        Table[
          RGBColor[displaying[[i, j, 1]], displaying[[i, j, 2]], 0],
          {i, partialgenes, 1, -1}, {j, 1, partialgenes}]]],
    AspectRatio -> 1,
    Frame -> True,
    FrameTicks -> {None, None, None, None},
    FrameLabel -> {None, labely, labelx, labelz},
    DisplayFunction -> Identity],
  g = FullGraphics[g],
  g[[1, 2]] = g[[1, 2]] /.
    Text[labely, {b_, c_}, {1., 0.}] ->
    Text[labely, {b - 4, c}, {0, 0}, {0, 1}],
  g[[1, 2]] = g[[1, 2]] /.
    Text[labelx, {b_, c_}, {0., -1.}] ->
    Text[labelx, {b, c}, {0, -1}, {1, 0}],
  p[[a]] = Show[g,
    AspectRatio -> 0.95,
    PlotRange -> All,
    DisplayFunction -> Identity]],
  {a, 4, 4}]

```


(* Create Coefficients of Eigencorrelations 2D Red & Green Raster Displays *)

```

q = Table[0, {a, 1, 3}];
contrast = 2;
Do[{
  correlation = {{fractions[[a]]}},
  displaying = Table[
    If[contrast * correlation[[i, j]] > 0,
      If[contrast * correlation[[i, j]] < 1, {contrast * correlation[[i, j]], 0}, {1, 0}],
      If[contrast * correlation[[i, j]] > -1, {0, -contrast * correlation[[i, j]]}, {0, 1}]],
    {i, 1, 1}, {j, 1, 1}],
  g = Show[
    Graphics[
      RasterArray[
        Table[
          RGBColor[displaying[[i, j, 1]], displaying[[i, j, 2]], 0],
          {i, 1, 1, -1}, {j, 1, 1}]],
      AspectRatio -> 1,
      Frame -> True,
      FrameTicks -> {None, None, None, None},
      FrameLabel -> {None, None, None, None},
      DisplayFunction -> Identity],
  g = FullGraphics[g],
  g[[1, 2]] = g[[1, 2]] /.
    Text[labely, {b_, c_}, {1., 0.}] ->
    Text[labely, {b - 8, c}, {0, 0}, {0, 1}],
  g[[1, 2]] = g[[1, 2]] /.
    Text[labelx, {b_, c_}, {0., -1.}] ->
    Text[labelx, {b, c}, {0, -1}, {1, 0}],
  q[[a]] = Show[g,
    AspectRatio -> 1.05,
    PlotRange -> All,
    DisplayFunction -> Identity]],
  {a, 1, 2}]
q[[3]] = Show[GraphicsArray[{{q[[1]]}, {q[[2]]}},
  GraphicsSpacing -> 0,
  DisplayFunction -> Identity];

```

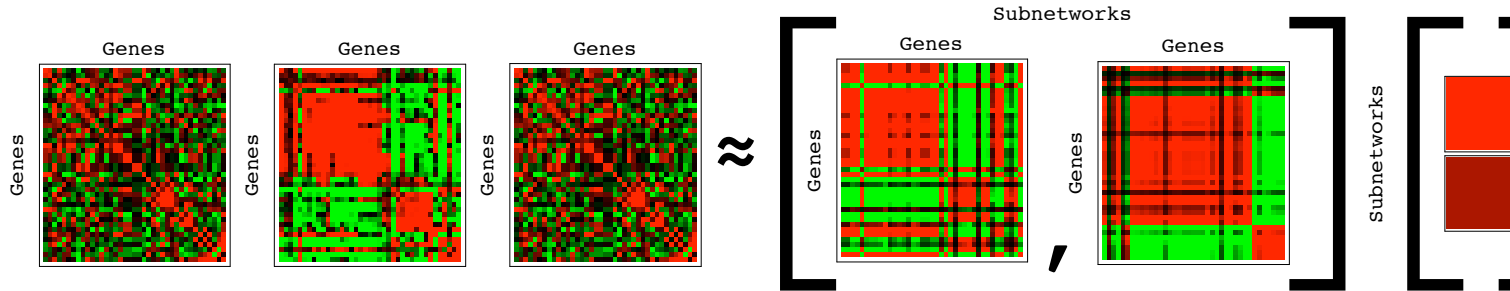
(* Display Network Decomposition *)

```

approx = Show[Graphics[
  Text[StyleForm["≈", FontSize -> 40, FontWeight -> Bold], {0, 0}]
], DisplayFunction -> Identity];
right = Show[Graphics[
  Text[StyleForm[""], FontSize -> 160], {0, 0}]
], DisplayFunction -> Identity];
left = Show[Graphics[
  Text[StyleForm[""], FontSize -> 160], {0, 0}]
], DisplayFunction -> Identity];
labelx = Show[Graphics[
  Text[StyleForm["Subnetworks"], {0, 0}]
], DisplayFunction -> Identity];
labely = Show[Graphics[
  Text[StyleForm["Subnetworks"], {0, 0}, {0, 0}, {0, 1}]
], DisplayFunction -> Identity];

```

```
Show[{
  Graphics[{Rectangle[{0.1, 0}, {1.2, 1}, p[[1]]}],
  Graphics[{Rectangle[{1.0, 0}, {2.1, 1}, p[[2]]}],
  Graphics[{Rectangle[{1.9, 0}, {3.0, 1}, p[[1]]}],
  Graphics[{Rectangle[{2.5, 0}, {3.3, 1}, approx]}],
  Graphics[{Rectangle[{2.1, 0}, {4.1, 1}, left}],
  Graphics[{Rectangle[{3.15, 0}, {4.25, 1}, p[[3]]}],
  Graphics[{Rectangle[{4.15, 0}, {5.25, 1}, p[[4]]}],
  Graphics[{Rectangle[{4.2, 0}, {6.2, 1}, right}],
  Graphics[{Rectangle[{2.1, 0.5}, {6.2, 0.75}, labelx}],
  Graphics[{Rectangle[{4.7, 0}, {6, 1}, labely}],
  Graphics[{Rectangle[{4.5, 0}, {6.5, 1}, left}],
  Graphics[{Rectangle[{5, 0.425}, {6.5, 0.575}, q[[3]]}],
  Graphics[{Rectangle[{5, 0}, {7, 1}, right}]}],
  PlotRange -> All];
```



```
(* Create Genes x Genes Eigencorrelations Network Displays *)
```

```
(* Keep Only Correlations Above Diagonals in Eigenmatrices *)
```

```
Do[  
  Do[  
    Do[  
      eigenmatrices[[a, b, c]] = 0,  
      {c, 1, b}},  
    {b, 1, Dimensions[eigenmatrices[[1]]][[1]]},  
    {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
(* Cutoff Correlations in Eigenmatrices *)
```

```
Do[  
  eigenmatrices[[a]] = Chop[eigenmatrices[[a]], cutoffs[[a]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}]
```

```
sizes = Flatten[  
  Table[  
    Dimensions[  
      Characters[  
        ToString[partialgenenames[[a, 2]]  
        ]]],  
    {a, 1, partialgenes}]];  
size = Sort[sizes, OrderedQ[{{#2, #1}} &]][[1]];
```

```
Do[  
  Do[partialgenenames[[a, 2]] = StringJoin[ToString[partialgenenames[[a, 2]]], " "  
  {b, 1, size - sizes[[a]]}],  
  {a, 1, partialgenes}];
```

```
(* Define Eigencorrelations Networks *)
```

```
networkgenes2 = partialgenenames;  
networks21 = eigenmatrices[[1]];  
networks22 = eigenmatrices[[2]];
```

(* Intersect Alpha Factor and Cell Cycle-Projected Alpha Factor Networks *)

```
genes = Dimensions[networkgenes1][[1]];
genenames = TakeColumns[networkgenes1, 1];

externalgenes = Dimensions[networkgenes2][[1]];
externalgenenames = TakeColumns[networkgenes2, 1];

list = Flatten[Intersection[genenames, externalgenenames]];
{partialgenes} = Dimensions[list]

{35}

counter = Table[{Position[genenames, list[[a]]][[1, 1]], list[[a]]}, {a, 1, partialgenes}];
counter = ReplaceAll[counter, {} -> {Null}];
counter = Sort[counter, OrderedQ[{{#1, #2}} &];
list = Flatten[TakeColumns[counter, {2}]];

counter = Table[Flatten[Position[list, genenames[[a, 1]]]], {a, 1, genes}];
counter = ReplaceAll[counter, {} -> {Null}];
partialmatrix = AppendRows[counter, networkgenes1, networks11, networks12, networks13, networks14];
partialmatrix = Sort[partialmatrix, OrderedQ[{{#1, #2}} &];
partialgenenames = TakeRows[
  TakeColumns[partialmatrix, {2, 8}],
  {1, partialgenes}];
partialmatrix11 = TakeRows[
  TakeColumns[partialmatrix, {9, genes + 8}],
  {1, partialgenes}];
partialmatrix12 = TakeRows[
  TakeColumns[partialmatrix, {genes + 9, 2 * genes + 8}],
  {1, partialgenes}];
partialmatrix13 = TakeRows[
  TakeColumns[partialmatrix, {2 * genes + 9, 3 * genes + 8}],
  {1, partialgenes}];
partialmatrix14 = TakeRows[
  TakeColumns[partialmatrix, {3 * genes + 9, 4 * genes + 8}],
  {1, partialgenes}];

partialmatrix11 = Transpose[partialmatrix11];
partialmatrix12 = Transpose[partialmatrix12];
partialmatrix13 = Transpose[partialmatrix13];
partialmatrix14 = Transpose[partialmatrix14];

partialmatrix = AppendRows[counter, partialmatrix11, partialmatrix12, partialmatrix13, partialmatrix14];
partialmatrix = Sort[partialmatrix, OrderedQ[{{#1, #2}} &];
partialmatrix11 = TakeRows[
  TakeColumns[partialmatrix, {2, partialgenes + 1}],
  {1, partialgenes}];
partialmatrix12 = TakeRows[
  TakeColumns[partialmatrix, {partialgenes + 2, 2 * partialgenes + 1}],
  {1, partialgenes}];
partialmatrix13 = TakeRows[
  TakeColumns[partialmatrix, {2 * partialgenes + 2, 3 * partialgenes + 1}],
  {1, partialgenes}];
partialmatrix14 = TakeRows[
  TakeColumns[partialmatrix, {3 * partialgenes + 2, 4 * partialgenes + 1}],
  {1, partialgenes}];

partialmatrix11 = Transpose[partialmatrix11];
partialmatrix12 = Transpose[partialmatrix12];
partialmatrix13 = Transpose[partialmatrix13];
partialmatrix14 = Transpose[partialmatrix14];
```

```

counter = Table[Flatten[Position[list, externalgenenames[[a, 1]]], {a, 1, externalgenes}];
counter = ReplaceAll[counter, {} -> {Null}];
partialexternalmatrix = AppendRows[counter, networks21, networks22];
partialexternalmatrix = Sort[partialexternalmatrix, OrderedQ[{{#1, #2}} &];
partialexternalmatrix21 = TakeRows[
  TakeColumns[partialexternalmatrix, {2, externalgenes + 1}],
  {1, partialgenes}];
partialexternalmatrix22 = TakeRows[
  TakeColumns[partialexternalmatrix, {externalgenes + 2, 2 * externalgenes + 1}],
  {1, partialgenes}];

partialexternalmatrix21 = Transpose[partialexternalmatrix21];
partialexternalmatrix22 = Transpose[partialexternalmatrix22];

partialexternalmatrix = AppendRows[counter, partialexternalmatrix21, partialexternalmatrix22];
partialexternalmatrix = Sort[partialexternalmatrix, OrderedQ[{{#1, #2}} &];
partialexternalmatrix21 = TakeRows[
  TakeColumns[partialexternalmatrix, {2, partialgenes + 1}],
  {1, partialgenes}];
partialexternalmatrix22 = TakeRows[
  TakeColumns[partialexternalmatrix, {partialgenes + 2, 2 * partialgenes + 1}],
  {1, partialgenes}];

partialexternalmatrix21 = Transpose[partialexternalmatrix21];
partialexternalmatrix22 = Transpose[partialexternalmatrix22];

```

```
(* Define Display Parameters *)
```

```
Clear[x, y, z];
R[theta_] = {{Cos[theta], Sin[theta]}, {-Sin[theta], Cos[theta]}};
polypoints[theta_, x_] = Flatten[Transpose[Dot[R[theta], Transpose[{{0, x}}]]]];
redline[y_, z_] = Graphics[{RGBColor[1, 0, 0],
  Line[{polypoints[2 * y * Pi / partialgenes, 1], polypoints[2 * z * Pi / partialgenes, 1]}]};
greenline[y_, z_] = Graphics[{RGBColor[0, 0.5, 0],
  Line[{polypoints[2 * y * Pi / partialgenes, 1], polypoints[2 * z * Pi / partialgenes, 1]}]};

color1[stage_] =
  If[stage == "Up", RGBColor[0, 0, 0],
    If[stage == "Down", RGBColor[0.5, 0.5, 0.5], RGBColor[1, 1, 1]]
  ];
color2[stage_] =
  If[stage == "M/G1", RGBColor[1, 1, 0],
    If[stage == "G1", RGBColor[0, 0.5, 0],
      If[stage == "S", RGBColor[0, 0, 1],
        If[stage == "S/G2", RGBColor[1, 0, 0],
          If[stage == "G2/M", RGBColor[1, 0.5, 0], RGBColor[1, 1, 1]]
        ]]]];

circle1 = Table[
  Graphics[{
    color1[partialgenenames[[a, 7]]],
    Disk[{0, 0}, 1.35,
      {-2 * (a + 0.5) * Pi / partialgenes + Pi / 2., -2 * (a - 0.5) * Pi / partialgenes + Pi / 2.}],
    {a, partialgenes, 1, -1}];
circle2 = Table[
  Graphics[{
    color2[partialgenenames[[a, 6]]],
    Disk[{0, 0}, 1.3,
      {-2 * (a + 0.5) * Pi / partialgenes + Pi / 2., -2 * (a - 0.5) * Pi / partialgenes + Pi / 2.}],
    {a, partialgenes, 1, -1}];
circle3 = Graphics[{RGBColor[1, 1, 1], Disk[{0, 0}, 1.25]}];
circle4 = Table[
  Graphics[{
    RGBColor[0, 0, 0],
    Disk[polypoints[2 * a * Pi / partialgenes, 1], 0.02]}, {a, 1, partialgenes}];
circle5 = Table[
  Graphics[{
    RGBColor[0, 0, 0],
    Text[partialgenenames[[a, 2]], polypoints[2 * a * Pi / partialgenes, 1.15],
      {0, 0}, polypoints[2 * a * Pi / partialgenes, 1.15]}],
  {a, 1, partialgenes}];
```

(* Display Intersection of Cycle-Projected First AND Alpha Factor Fourth Networks *)

```

int1 = Intersection[
  Position[Sign[partialmatrix14], 1],
  Position[Sign[partialexternalmatrix21], 1]];
int2 = Intersection[
  Position[Sign[partialmatrix14], -1],
  Position[Sign[partialexternalmatrix21], -1]];

redtable = Table[redline[int1[[a, 1]], int1[[a, 2]]], {a, 1, Dimensions[int1][[1]]};
greentable = Table[greenline[int2[[a, 1]], int2[[a, 2]]], {a, 1, Dimensions[int2][[1]]};
p = Show[
  {circle1, circle2, circle3, circle4, circle5, redtable, greentable,
   Graphics[{RGBColor[0, 0, 0], Text[StyleForm["(a)", FontSize -> 24], {-1.1, 1.1}]}]},
  AspectRatio -> 1,
  PlotRange -> {{-1.25, 1.25}, {-1.25, 1.25}},
  Frame -> False,
  FrameTicks -> False,
  FrameLabel -> {None, None, None, None},
  GridLines -> {None, None},
  DisplayFunction -> Identity];
p = FullGraphics[p];
p[[1, 2]] = p[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {-1.18, 0}, {0, 0}, {0, 1}];
p1 = Show[p,
  AspectRatio -> 1.,
  PlotRange -> All,
  DisplayFunction -> Identity];

```

(* Display Intersection of Cycle-Projected Second AND Alpha Factor Third Networks *)

```

int1 = Intersection[
  Position[Sign[partialmatrix13], 1],
  Position[Sign[partialexternalmatrix22], 1]];
int2 = Intersection[
  Position[Sign[partialmatrix13], -1],
  Position[Sign[partialexternalmatrix22], -1]];

redtable = Table[redline[int1[[a, 1]], int1[[a, 2]]], {a, 1, Dimensions[int1][[1]]};
greentable = Table[greenline[int2[[a, 1]], int2[[a, 2]]], {a, 1, Dimensions[int2][[1]]};
p = Show[
  {circle1, circle2, circle3, circle4, circle5, redtable, greentable,
   Graphics[{RGBColor[0, 0, 0], Text[StyleForm["(b)", FontSize -> 24], {-1.1, 1.1}]}]},
  AspectRatio -> 1,
  PlotRange -> {{-1.25, 1.25}, {-1.25, 1.25}},
  Frame -> False,
  FrameTicks -> False,
  FrameLabel -> {None, None, None, None},
  GridLines -> {None, None},
  DisplayFunction -> Identity];
p = FullGraphics[p];
p[[1, 2]] = p[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {-1.18, 0}, {0, 0}, {0, 1}];
p2 = Show[p,
  AspectRatio -> 1.,
  PlotRange -> All,
  DisplayFunction -> Identity];

```

```
(* Project Genes x Genes Network Onto Development Transcription Factors' DNA-Binding Data *)
```

```
(* Read Development Transcription Factors' DNA-Binding Data *)
```

```
stream = "Desktop/Network_Decomposition/Data/Develop_Binding.txt";  
matrix = Import[stream, "Table"];  
{genes, arrays} = Dimensions[matrix] - {1, 7}  
Clear[stream];  
  
{2476, 12}
```

```
genenames = TakeRows[  
  TakeColumns[matrix, {1, 7}],  
  {2, genes + 1}];  
arraynames = TakeColumns[  
  TakeRows[matrix, {1, 1}],  
  {8, arrays + 7}];  
matrix = TakeColumns[  
  TakeRows[matrix, {2, genes + 1}],  
  {8, arrays + 7}];  
matrix = ToExpression[matrix];
```

```
sizes = Flatten[  
  Table[  
    Dimensions[  
      Characters[  
        ToString[arraynames[[1, a]]  
      ]],  
    {a, 1, arrays}];  
size = Sort[sizes, OrderedQ[{{#2, #1}} &]][[1]];  
Do[  
  Do[arraynames[[1, a]] = StringJoin[ToString[arraynames[[1, a]]], " "  
    {b, 1, size - sizes[[a]]},  
    {a, 1, arrays}];
```

```
(* Convert to Ratios *)
```

```
average = Table[1, {a, 1, arrays}];  
average = N[average / Sqrt[Dot[average, average]]];  
matrix = matrix / N[Outer[Times, Dot[matrix, average], average]];
```

```
matrix3 = matrix;  
genenames3 = genenames;  
arraynames3 = arraynames;  
{genes3, arrays3} = Dimensions[matrix3]
```

```
{2476, 12}
```

```
(* Calculate SVD *)
```

```
{eigenarrays, eigenexpressions, eigengenest} = SingularValues[matrix];  
eigenarrays = Transpose[eigenarrays];  
fractions = eigenexpressions^2 /  
  Sum[eigenexpressions[[a]]^2, {a, 1, Dimensions[eigenexpressions][[1]]}];  
entropy = -N[Sum[fractions[[a]] * Log[fractions[[a]]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}] /  
  Log[Dimensions[eigenexpressions][[1]]];  
entropy = N[Round[100 * entropy] / 100]
```

```
0.2
```


(* Create Fractions Bar Charts Displays *)

```
fractions[[2]]
0.0240689

limit = 0.03;
alsolimit = fractions[[2]];

Clear[gridx, framex, framey, sizes];
gridx = Table[a, {a, 0, limit, N[limit/5]}];
framex = gridx;
sizes = Flatten[
  Table[
    Dimensions[
      Characters[
        ToString[framex[[a]]
          ]]], {a, 1, 6}]];
Do[
  Do[framex[[a]] = StringJoin[ToString[framex[[a]]], " "],
    {b, 1, 5 - sizes[[a]]},
  {a, 1, 6}];
framex = Table[{gridx[[a]], framex[[a]]}, {a, 1, 6}];
gridx = Table[{gridx[[a]], RGBColor[0, 0, 0]}, {a, 1, 6}];
framey = Table[{a + 1, arrays - a}, {a, 0, arrays - 2}];
table = Table[fractions[[arrays - a]], {a, 0, arrays - 2}];
g = BarChart[table,
  BarOrientation -> Horizontal,
  PlotRange -> {{0, alsolimit * 1.0001}, {0.5, 12 - 1 + 0.5}},
  AspectRatio -> 1,
  Axes -> False,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, None, None, None},
  GridLines -> {gridx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 1.75}, {0, 0}, {0, 1}];
g1 = Show[g,
  AspectRatio -> 1.25,
  PlotRange -> All,
  DisplayFunction -> Identity];
```

```

gridx = Table[a, {a, 0, 1, 0.2}];
framex = gridx;
sizes = Flatten[
  Table[
    Dimensions[
      Characters[
        ToString[framex[[a]]
          ]]], {a, 1, 6}]];
Do[
  Do[framex[[a]] = StringJoin[ToString[framex[[a]]], " ",
    {b, 1, size - sizes[[a]]}],
    {a, 1, 6}];
framex = Table[{gridx[[a]], framex[[a]]}, {a, 1, 6}];
gridx = Table[{gridx[[a]], RGBColor[0, 0, 0]}, {a, 1, 6}];
framey = Table[{a + 1, arrays - a}, {a, 0, Dimensions[eigenexpressions][[1]]}];
labelx = ColumnForm[
  {"(b) Eigenbinding Fraction", StringJoin["d3 = ", ToString[entropy]], " "},
  Center];
g = BarChart[
  Table[fractions[[arrays - a]], {a, 0, arrays - 1}],
  BarOrientation -> Horizontal,
  PlotRange -> {{0, 1.0001}, {0.5, arrays + 0.5}},
  AspectRatio -> 1,
  Axes -> False,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, None, labelx, None},
  GridLines -> {gridx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 1.5}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 1.25}, {0, 0}, {0, 1}];
g2 = Show[{g,
  Graphics[{RGBColor[1, 1, 0.8], Rectangle[{0.1, 0.6}, {0.98, 11.3}]},
  Graphics[{Rectangle[{0.1, 0.6}, {0.98, 11.3}, g1]}]},
  AspectRatio -> 1.35,
  PlotRange -> All,
  DisplayFunction -> Identity];

```

```
(* Create Eigenenes 2D Red & Green Raster Display *)
```

```
contrast = 3.5;
displaying = Table[
  If[contrast * eigengenenes[[i, j]] > 0,
    If[contrast * eigengenenes[[i, j]] < 1, {contrast * eigengenenes[[i, j]], 0}, {1, 0}],
    If[contrast * eigengenenes[[i, j]] > -1, {0, -contrast * eigengenenes[[i, j]]}, {0, 1}]],
  {i, 1, Dimensions[eigenexpressions][[1]]}, {j, 1, arrays}];
framex = Table[{a - 0.5, arraynames[[1, a]]}, {a, 1, arrays}];
framey = Table[{a + 1 - 0.5, arrays - a}, {a, 0, Dimensions[eigenexpressions][[1]]}];
labely = "Eigenenes";
labelx = ColumnForm[{"(a) Arrays", " ", " "}, Center];

g = Show[
  Graphics[
    RasterArray[
      Table[
        RGBColor[displaying[[i, j, 1]], displaying[[i, j, 2]], 0],
        {i, Dimensions[eigenexpressions][[1]], 1, -1}, {j, 1, arrays}]]],
  AspectRatio -> 1,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, labely, labelx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 2.2, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 1.5}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 1.25}, {0, 0}, {0, 1}];
g1 = Show[g,
  AspectRatio -> 1.05,
  PlotRange -> All,
  DisplayFunction -> Identity];
```

(* Create Selected Eigengenes Graph Display *)

```

labelx = ColumnForm[{"(c) Arrays"}, Center];
labely = ColumnForm[{" ", "Binding Level"}, Center];
framex = Table[{a - 1, arraynames[[1, a]]}, {a, 1, arrays}];
framey = {-1, -0.5, 0, 0.5, 1};
coordinates = Table[{a - 1, eigengenes[[1, a]]}, {a, 1, arrays}];
points = Table[Point[coordinates[[a]]], {a, 1, arrays}];
line = Line[coordinates];
g = Show[
  {Graphics[{RGBColor[1, 0, 0], PointSize[0.022], points}],
  Graphics[{RGBColor[1, 0, 0], line}]},
  Frame -> True,
  FrameLabel -> {None, labely, labelx, None},
  GridLines -> {None, {{0, RGBColor[0, 0, 0]}}},
  FrameTicks -> {None, framey, framex, None},
  PlotRange -> {-1.05, 1.05},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 3, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 0.5}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 0.2}, {0, 0}, {0, 1}];
g3 = Show[g,
  AspectRatio -> 1.05,
  PlotRange -> All,
  DisplayFunction -> Identity];

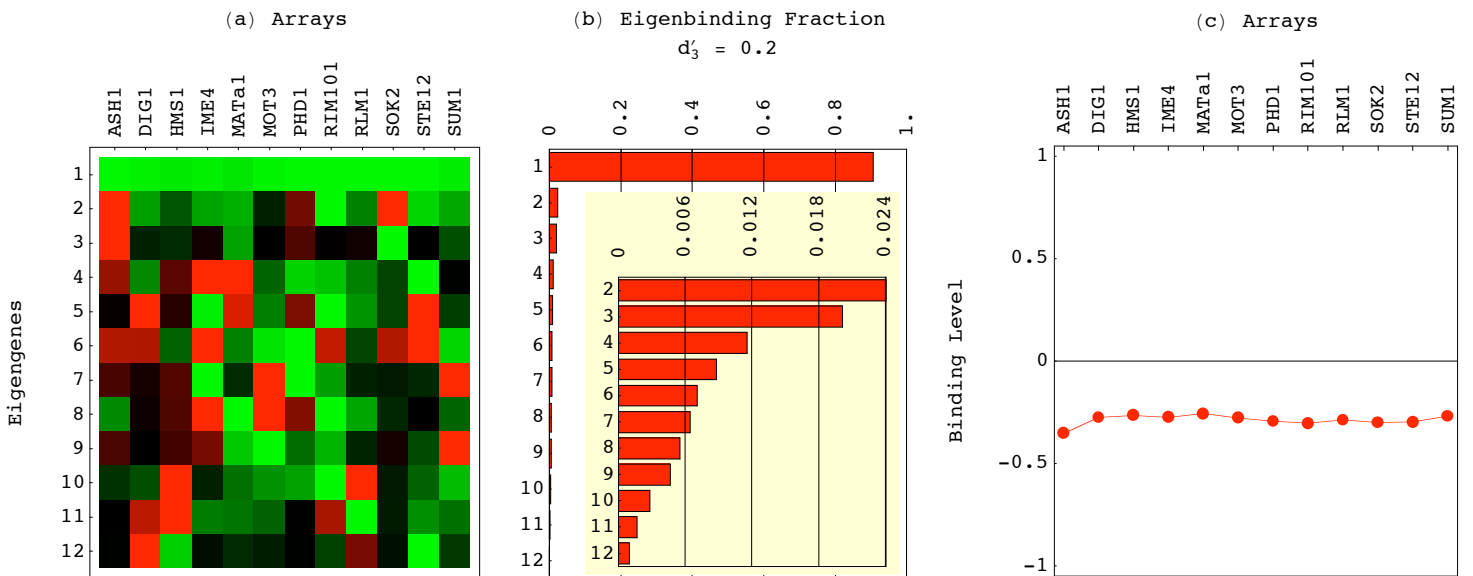
```

(* Display Eigengenes, Fractions and Selected Eigengenes *)

```

Show[GraphicsArray[{g1, g2, g3}],
  GraphicsSpacing -> -0.15];

```



```
(* Reconstruct Data Without Additive Steady State *)
```

```
{eigenarrays, eigenexpressions, eigengenes} = SingularValues[matrix];  
eigenarrays = Transpose[eigenarrays];  
eigenexpressions[[1]] = 0;  
matrix = Dot[eigenarrays, Dot[DiagonalMatrix[eigenexpressions], eigengenes]];  
  
matrix3 = matrix;  
genenames3 = genenames;  
arraynames3 = arraynames;  
{genes3, arrays3} = Dimensions[matrix3]  
  
{2476, 12}
```

```
(* Project mRNA Expression Data Onto Proteins' DNA-Binding Data *)
```

```
genes = genes1;  
genenames = TakeColumns[genenames1, 1];  
arrays = arrays1;  
arraynames = arraynames1;  
  
externalgenes = genes3;  
externalgenenames = TakeColumns[genenames3, 1];  
externalarrays = arrays3;  
externalarraynames = arraynames3;  
  
list = Flatten[Intersection[genenames, externalgenenames]];  
{partialgenes} = Dimensions[list]  
  
{1827}  
  
counter = Table[{Position[genenames, list[[a]]][[1, 1]], list[[a]]}, {a, 1, partialgenes}];  
counter = ReplaceAll[counter, {} -> {Null}];  
counter = Sort[counter, OrderedQ[{{#1, #2}} &];  
list = Flatten[TakeColumns[counter, {2}]];  
  
counter = Table[Flatten[Position[list, genenames[[a, 1]]], {a, 1, genes}];  
counter = ReplaceAll[counter, {} -> {Null}];  
partialmatrix = AppendRows[counter, genenames1, matrix1];  
partialmatrix = Sort[partialmatrix, OrderedQ[{{#1, #2}} &];  
partialgenenames = TakeRows[  
  TakeColumns[partialmatrix, {2, 8}],  
  {1, partialgenes}];  
partialmatrix = TakeRows[  
  TakeColumns[partialmatrix, {9, arrays + 8}],  
  {1, partialgenes}];  
  
counter = Table[Flatten[Position[list, externalgenenames[[a, 1]]], {a, 1, externalgenes}];  
counter = ReplaceAll[counter, {} -> {Null}];  
partialexternalmatrix = AppendRows[counter, externalgenenames, matrix3];  
partialexternalmatrix = Sort[partialexternalmatrix, OrderedQ[{{#1, #2}} &];  
partialexternalgenenames = TakeRows[  
  TakeColumns[partialexternalmatrix, {2, 2}],  
  {1, partialgenes}];  
partialexternalmatrix = TakeRows[  
  TakeColumns[partialexternalmatrix, {3, externalarrays + 2}],  
  {1, partialgenes}];  
  
partialmatrix = Dot[partialexternalmatrix, Dot[PseudoInverse[partialexternalmatrix], partialmatrix]];
```

```
(* Examine mRNA Expression Data After Projection *)
```

```
(* Calculate SVD *)
```

```
matrix = partialmatrix;  
genes = partialgenes;  
genenames = TakeColumns[partialgenenames, 1];  
  
{eigenarrays, eigenexpressions, eigengenes} = SingularValues[matrix];  
eigengenes[[1]] = -eigengenes[[1]];  
eigenarrays[[1]] = -eigenarrays[[1]];  
eigenarrays = Transpose[eigenarrays];  
fractions = eigenexpressions^2 /  
  Sum[eigenexpressions[[a]]^2, {a, 1, Dimensions[eigenexpressions][[1]]}];  
entropy = -N[Sum[fractions[[a]] * Log[fractions[[a]]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}] /  
  Log[Dimensions[eigenexpressions][[1]]];  
entropy = N[Round[100 * entropy] / 100]
```

```
0.17
```

```
(* Create Fractions Bar Charts Display *)
```

```
fractions[[1]]
```

```
0.917079
```

```
limit = 0.95;
```

```
alsolimit = fractions[[1]];
```

```

gridx = Table[a, {a, 0, limit, N[limit/5]};
framex = gridx;
sizes = Flatten[
  Table[
    Dimensions[
      Characters[
        ToString[framex[[a]]
      ]], {a, 1, 6}];
Do[
  Do[framex[[a]] = StringJoin[ToString[framex[[a]]], " "],
    {b, 1, size - sizes[[a]]},
    {a, 1, 6}];
framex = Table[{gridx[[a]], framex[[a]]}, {a, 1, 6}];
gridx = Table[{gridx[[a]], RGBColor[0, 0, 0]}, {a, 1, 6}];
framey = Table[{a + 1, Dimensions[eigenexpressions] [[1]] - a},
  {a, 0, Dimensions[eigenexpressions] [[1]] - 1}];
labelx = ColumnForm[
  {"(b) Eigenexpression Fraction", StringJoin["d3 = ", ToString[entropy]], " "},
  Center];
g = BarChart[
  Table[fractions[[Dimensions[eigenexpressions] [[1]] - a]],
    {a, 0, Dimensions[eigenexpressions] [[1]] - 1}],
  BarOrientation -> Horizontal,
  PlotRange -> {{0, alsolimit*1.0001}, {0.5, Dimensions[eigenexpressions] [[1]] + 0.5}},
  AspectRatio -> 1,
  Axes -> False,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, None, labelx, None},
  GridLines -> {gridx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 2.25}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 1.5}, {0, 0}, {0, 1}];
g2 = Show[g,
  AspectRatio -> 1.35 * 14 / 18,
  PlotRange -> All,
  DisplayFunction -> Identity];

```

(* Create Eigengenes 2D Red & Green Raster Display *)

```

contrast = 3.5;
displaying = Table[
  If[contrast * eigengenes[[i, j]] > 0,
    If[contrast * eigengenes[[i, j]] < 1, {contrast * eigengenes[[i, j]], 0}, {1, 0}],
    If[contrast * eigengenes[[i, j]] > -1, {0, -contrast * eigengenes[[i, j]]}, {0, 1}]],
  {i, 1, Dimensions[eigenexpressions][[1]]}, {j, 1, arrays}];
framex = Table[{a - 0.5, arraynames[[1, a]]}, {a, 1, arrays}];
framey = Table[{a + 1 - 0.5, Dimensions[eigenexpressions][[1]] - a},
  {a, 0, Dimensions[eigenexpressions][[1]] - 1}];
labely = "Eigengenes";
labelx = ColumnForm[{"(a) Arrays", " ", " "}, Center];

g = Show[
  Graphics[
    RasterArray[
      Table[
        RGBColor[displaying[[i, j, 1]], displaying[[i, j, 2]], 0],
        {i, Dimensions[eigenexpressions][[1]], 1, -1}, {j, 1, arrays}]]],
  AspectRatio -> 1,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, labely, labelx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 3, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 2.25}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 2}, {0, 0}, {0, 1}];
g1 = Show[g,
  AspectRatio -> 1.05 * 14 / 18,
  PlotRange -> All,
  DisplayFunction -> Identity];

```



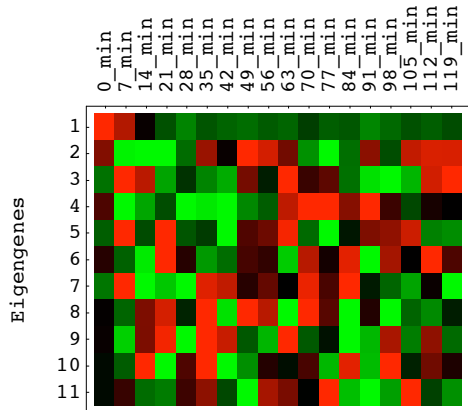
```
(* Create Selected Eigengenes Graph Display *)
```

```
labelx = ColumnForm[{"(c) Arrays", Center];  
labely = ColumnForm[{" ", "Expression Level"}, Center];  
framex = Table[{a - 1, arraynames[[1, a]]}, {a, 1, arrays}];  
framey = {-1, -0.5, 0, 0.5, 1};  
coordinates = Table[{a - 1, eigengenes[[1, a]]}, {a, 1, arrays}];  
points = Table[Point[coordinates[[a]]], {a, 1, arrays}];  
line = Line[coordinates];  
g = Show[  
  {Graphics[{RGBColor[1, 0, 0], PointSize[0.022], points}],  
  Graphics[{RGBColor[1, 0, 0], line}]},  
  Frame -> True,  
  FrameLabel -> {None, labely, labelx, None},  
  GridLines -> {None, {{0, RGBColor[0, 0, 0]}}},  
  FrameTicks -> {None, framey, framex, None},  
  PlotRange -> {-1.05, 1.05},  
  DisplayFunction -> Identity];  
g = FullGraphics[g];  
g[[1, 2]] = g[[1, 2]] /.  
  Text[labely, {b_, c_}, {1., 0.}] ->  
  Text[labely, {b - 5.4, c}, {0, 0}, {0, 1}];  
g[[1, 2]] = g[[1, 2]] /.  
  Text[labelx, {b_, c_}, {0., -1.}] ->  
  Text[labelx, {b, c + 0.625}, {0, -1}, {1, 0}];  
g[[1, 2]] = g[[1, 2]] /.  
  Text[a_, {b_, c_}, {0., -1.}] ->  
  Text[a, {b, c + 0.25}, {0, 0}, {0, 1}];  
p = Show[g,  
  AspectRatio -> 1.05,  
  PlotRange -> All, DisplayFunction -> Identity];
```

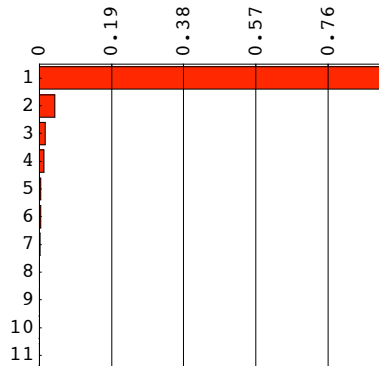
(* Display Eigengenes, Fractions and Selected Eigengenes *)

```
Show[GraphicsArray[{g1, g2}],
GraphicsSpacing -> -0.15];
```

(a) Arrays

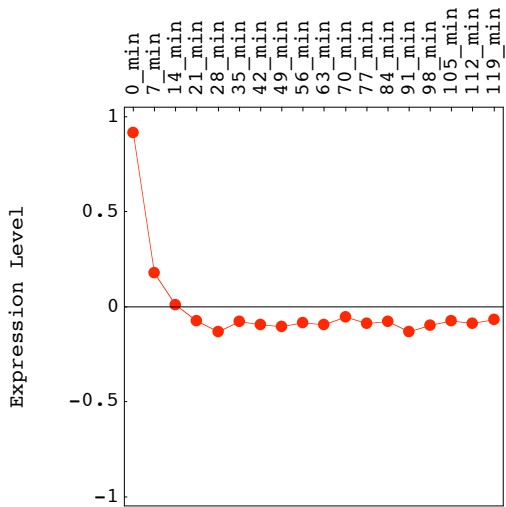


(b) Eigenexpression Fraction
 $d_3 = 0.17$



```
Show[GraphicsArray[{p}],
GraphicsSpacing -> -0.15];
```

(c) Arrays



(* Define Projected Data *)

```
partialmatrix3 = partialmatrix;
partialgenenames3 = partialgenenames;
```

```
(* Display Sorted Significant Eigenarrays *)
```

```
arraypatterns = Transpose[eigenarrays];
```

```
(* Center Eigenarrays *)
```

```
average = Table[1, {a, 1, genes}];
```

```
average = N[average / Sqrt[Dot[average, average]]];
```

```
arraypatterns = arraypatterns - N[Outer[Times, Dot[arraypatterns, average], average]];
```

```
(* Sort Eigenarrays *)
```

```
Do[
```

```
arraypatterns[[a]] = Sort[arraypatterns[[a]], OrderedQ[{{#2}, {#1}}] &],  
{a, 1, Dimensions[eigenexpressions][[1]]}]
```

```
(* Create Sorted Eigenarrays Graph Display *)
```

```
p = Table[0, {a, 1, 1}];
```

```
color = {
```

```
RGBColor[1, 0.5, 0],
```

```
RGBColor[1, 0, 0],
```

```
RGBColor[0, 0, 1],
```

```
RGBColor[0, 0.5, 0]};
```

```
labelx = "Expression Level";
```

```
labeled = ColumnForm[
```

```
{ "Number of Genes", " ", " ", " ", " ", " ", " ", " ", " ", " ", " ", " ",  
Center];
```

```
framex = Table[{0.02 * a, a}, {a, 1, 1}];
```

```
framey = {{-150, "150"}, {-750, "750"}, {-genes + 150, "1558"}};
```

```
Do[ {
```

```
coordinates = Table[
```

```
If[arraypatterns[[n, a]] + 0.02 * n < -0.04, -0.04,
```

```
If[arraypatterns[[n, a]] + 0.02 * n > 0.08, 0.08,
```

```
arraypatterns[[n, a]] + 0.02 * n]],
```

```
{a, 1, genes}],
```

```
coordinates = Table[{coordinates[[a]], -a + 1}, {a, 1, genes}],
```

```
line = Line[coordinates],
```

```
g = Show[
```

```
Graphics[{color[[Mod[n, 4] + 1]], line}],
```

```
Frame -> True,
```

```
FrameLabel -> {None, labeled, labelx, None},
```

```
FrameTicks -> {None, framey, framex, None},
```

```
GridLines -> {{{0.02 * n, RGBColor[0, 0, 0]}}, {{-150, RGBColor[0, 0, 0]},
```

```
{-750, RGBColor[0, 0, 0]}, {-genes + 150, RGBColor[0, 0, 0]}},
```

```
PlotRange -> {{-0.04, 0.08}, {57.5, -genes + 1 - 57.5}},
```

```
DisplayFunction -> Identity],
```

```
g = FullGraphics[g],
```

```
g[[1, 2]] = g[[1, 2]] /.
```

```
Text[labeled, {b_, c_}, {1., 0.}] ->
```

```
Text[labeled, {b, c}, {0, 0}, {0, 1}],
```

```
g[[1, 2]] = g[[1, 2]] /.
```

```
Text[labelx, {b_, c_}, {0., -1.}] ->
```

```
Text[labelx, {b, c + 164.5}, {0, -1}, {1, 0}],
```

```
p[[n]] = Show[g,
```

```
AspectRatio -> 2 / 1.2 / GoldenRatio,
```

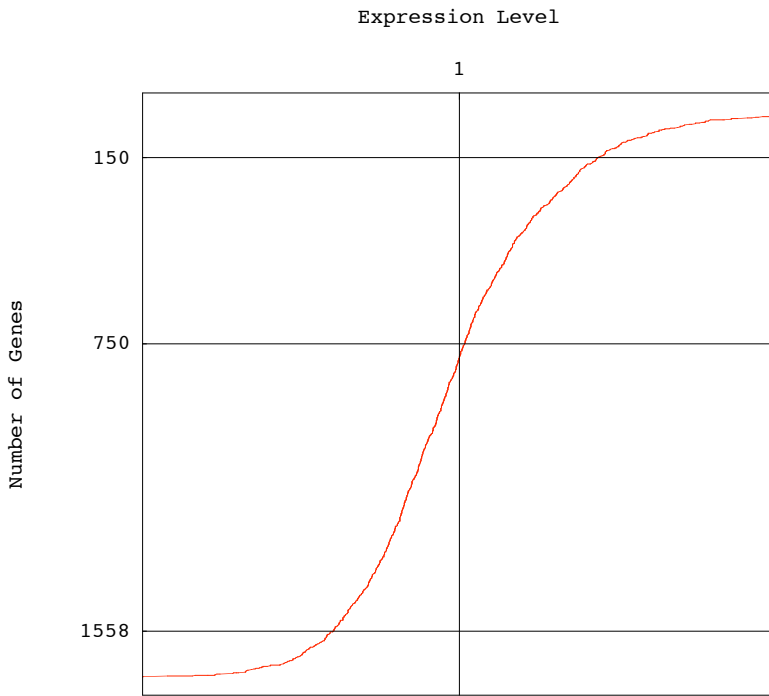
```
PlotRange -> All,
```

```
DisplayFunction -> Identity]
```

```
}, {n, 1, 1}];
```

(* Display Sorted Eigenarrays *)

```
Show[Table[p[[a]], {a, 1, 1}],  
      DisplayFunction -> $DisplayFunction];
```



(* Estimate Significance of Association of Eigenarrays with the Cell Cycle *)

(* Use Cell Cycle Traditional or Microarray Classification of Yeast Genes *)

```
most = 150;
annotations = TakeColumns[partialgenenames, {6}];
stages = {"M/G1", "G1", "S", "S/G2", "G2/M", "None"};
numbers = Flatten[Table[{Count[Flatten[annotations], stages[[a]]], {a, 1, Dimensions[stages][[1]]}]];
counter = Table[{a}, {a, 1, 1}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
```

numbers

```
{51, 129, 23, 41, 75, 1508}
```

```
Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenarrays, counter[[c]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
    Binomial[genes, most]], {b, table[[a, 3]], most}],
    stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1]], 2]},
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
    Binomial[genes, most]], {b, table[[a, 3]], most}],
    stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1]], 2]},
  {c, 1, Dimensions[counter][[1]]}]
```

```
table1 = AppendRows[
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];
```

(* Estimate Significance of Association of Eigenarrays with the Pheromone Response *)

(* Use Pheromone Microarray Classification of Yeast Genes *)

```
most = 150;
annotations = TakeColumns[partialgenenames, {7}];
stages = {"Up", "Down", "None"};
numbers = Flatten[Table[{Count[Flatten[annotations], stages[[a]]], {a, 1, Dimensions[stages][[1]]}]];
counter = Table[{a}, {a, 1, 1}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
```

numbers

```
{68, 85, 1674}
```

```
Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenarrays, counter[[c]]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stages[[a]],
    {a, 1, Dimensions[stages][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}},
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stages[[a]],
    {a, 1, Dimensions[stages][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}},
  {c, 1, Dimensions[counter][[1]]}]
```

```
table2 = AppendRows[
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];
```

(* Display Significance of Association of Eigenarrays with the Cellular Programs *)

```

headerx = {{
  ColumnForm[{" ", " ", " "}, Left],
  ColumnForm[{" ", " ", "Classification"}, Left],
  ColumnForm[{" ", " ", "Eigenarray"}, Left],
  ColumnForm[{"Most Likely", "Parallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Parallel", "Association"}, Left],
  ColumnForm[{"Most Likely", "Antiparallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Antiparallel", "Association"}, Left]},
{" ", " ", " ", " ", " ", " ", " ", " ", " ", " "}};
spacerx = {" ", " ", " ", " ", " "};
headery = Table[" ", {a, 1, 2*Dimensions[counter][[1]] + 1}, {b, 1, 2}];
headery[[1]] = {"(a)", "Cell Cycle"};
headery[[Dimensions[counter][[1]] + 2]] = {"(b)", "Pheromone Response"};
association =
  AppendColumns[headerx,
  AppendRows[headery,
  AppendColumns[table1, spacerx, table2]]];
TableForm[association, TableSpacing -> {1, 1}]

```

	Classification	Eigenarray	Most Likely Parallel Association	P-Value of Parallel Association	Most Likely Antiparallel Association	P-Value of Antiparallel Association
(a)	Cell Cycle	1	M/G1	6.9×10^{-3}	None	9.8×10^{-2}
(b)	Pheromone Response	1	Up	$1. \times 10^{-10}$	None	2.9×10^{-3}

```
(* Examine the Development-Projected Alpha Factor Genes x Genes Correlations Eigenmatrices *)
```

```
(* Read List of Cell Cycle-Regulated Genes *)
```

```
stream = "Desktop/Network_Decomposition/Data/Cycle_Genes.txt";  
genelist = Import[stream, "Table"];  
genelist = Drop[genelist, 1];  
partialgenes = Dimensions[Intersection[genelist, partialgenenames3]][[1]]  
Clear[stream];
```

44

```
list = Flatten[TakeColumns[genelist, {1}]];  
counter = Table[Flatten[Position[list, partialgenenames3[[a, 1]]]],  
  {a, 1, Dimensions[partialgenenames3][[1]]}];  
counter = ReplaceAll[counter, {} -> {Null}];  
partial = AppendRows[counter, partialgenenames3, eigenarrays];  
partial = Sort[partial, OrderedQ[{{#1, #2}} &];  
partialgenenames = TakeRows[  
  TakeColumns[partial, {2, 8}],  
  {1, partialgenes}];  
partialeigenarrays = TakeRows[  
  TakeColumns[partial, {9, Dimensions[eigenexpressions][[1]] + 8}],  
  {1, partialgenes}];
```

```
(* Calculate Eigenmatrices *)
```

```
partialeigenarrays = Transpose[partialeigenarrays];  
eigenmatrices = Table[0, {a, 1, Dimensions[eigenexpressions][[1]]}];  
Do[  
  eigenmatrices[[a]] = Outer[Times, partialeigenarrays[[a]], partialeigenarrays[[a]],  
    {a, 1, Dimensions[eigenexpressions][[1]]}  
  partialeigenarrays = Transpose[partialeigenarrays];
```

```
(* Keep Only Correlations Above Diagonals in Eigenmatrices *)
```

```
Do[  
  Do[  
    Do[  
      eigenmatrices[[a, b, c]] = 0,  
      {c, 1, b}],  
    {b, 1, Dimensions[eigenmatrices[[1]][[1]]}],  
  {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
(* Flatten Eigenmatrices Into Eigenvectors *)
```

```
eigenvectors = Table[0, {partialgenes * (partialgenes - 1) / 2, {Dimensions[eigenexpressions][[1]]}];  
Do[{  
  n = 0,  
  square = eigenmatrices[[a]],  
  Do[{  
    line = square[[b]],  
    Do[{  
      n = n + 1,  
      eigenvectors[[n, a]] = line[[c]]  
    }, {c, b + 1, partialgenes}],  
  }, {b, 1, partialgenes}],  
  {a, 1, Dimensions[eigenexpressions][[1]]}];
```



```
(* Create Genes x Genes Annotation Matrices for Selected Genes *)
```

```
partialgenenames = Transpose[partialgenenames];  
annotatematrices = Table[  
  Table[0, {partialgenes}, {partialgenes}],  
  {a, 1, 7}];  
Do[  
  annotatematrices[[a]] = Outer[Times, partialgenenames[[a]], partialgenenames[[a]],  
  {a, 1, 7}]
```

```
(* Keep Only Correlations Above Diagonals in Annotation Matrices *)
```

```
Do[  
  Do[  
    Do[  
      annotatematrices[[a, b, c]] = 0,  
      {c, 1, b}],  
    {b, 1, partialgenes}],  
  {a, 1, 7}];
```

```
(* Flatten Annotation Matrices Into Annotation Vectors *)
```

```
annotatevectors = Table[0, {partialgenes * (partialgenes - 1) / 2}, {7}];  
Do[{  
  n = 0,  
  square = annotatematrices[[a]],  
  Do[{  
    line = square[[b]],  
    Do[{  
      n = n + 1,  
      annotatevectors[[n, a]] = line[[c]]  
    }, {c, b + 1, partialgenes}],  
  }, {b, 1, partialgenes}]  
}, {a, 1, 7}]  
partialgenenames = Transpose[partialgenenames];
```

```
(* Display Sorted Significant Eigenmatrices *)
```

```
arraypatterns = Transpose[eigenvectors];  
genes = Dimensions[arraypatterns][[2]]
```

```
946
```

```
(* Center Eigenmatrices *)
```

```
average = Table[1, {a, 1, genes}];  
average = N[average / Sqrt[Dot[average, average]]];  
arraypatterns = arraypatterns - N[Outer[Times, Dot[arraypatterns, average], average]];
```

```
(* Sort Eigenmatrices *)
```

```
Do[  
  arraypatterns[[a]] = Sort[arraypatterns[[a]], OrderedQ[{{#2}, {#1}}] &],  
  {a, 1, Dimensions[eigenexpressions][[1]]}]
```

```
(* Create Sorted Eigenmatrices Graph Display *)
```

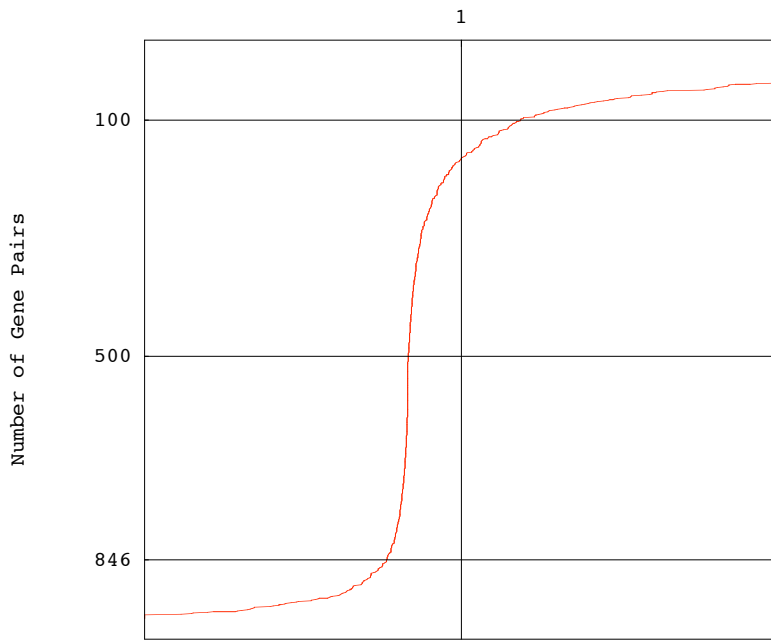
```
p = Table[0, {a, 1, 1}];  
color = {  
  RGBColor[1, 0.5, 0],  
  RGBColor[1, 0, 0],  
  RGBColor[0, 0, 1],  
  RGBColor[0, 0.5, 0]};  
labelx = "Expression Correlation Level";  
labely = ColumnForm[  
  {"Number of Gene Pairs", " ", " ", " ", " ", " ", " ", " ", " ", " ", " "},  
  Center];  
framex = Table[{0.002 * a, a}, {a, 1, 1}];  
framey = {{-100, "100"}, {-500, "500"}, {-genes + 100, "846"}};
```

```
Do[{  
  coordinates = Table[  
    If[arraypatterns[[n, a]] + 0.002 * n < -0.002, -0.002,  
      If[arraypatterns[[n, a]] + 0.002 * n > 0.006, 0.006,  
        arraypatterns[[n, a]] + 0.002 * n]],  
    {a, 1, genes}],  
  coordinates = Table[{coordinates[[a]], -a + 1}, {a, 1, genes}],  
  line = Line[coordinates],  
  g = Show[  
    Graphics[{color[[Mod[n, 4] + 1]], line}],  
    Frame -> True,  
    FrameLabel -> {None, labely, labelx, None},  
    FrameTicks -> {None, framey, framex, None},  
    GridLines -> {{{0.002 * n, RGBColor[0, 0, 0]}, {-100, RGBColor[0, 0, 0]},  
      {-500, RGBColor[0, 0, 0]}, {-genes + 100, RGBColor[0, 0, 0]}}},  
    PlotRange -> {{-0.002, 0.006}, {35, -genes + 1 - 35}},  
    DisplayFunction -> Identity],  
  g = FullGraphics[g],  
  g[[1, 2]] = g[[1, 2]] /.  
    Text[labely, {b_, c_}, {1., 0.}] ->  
    Text[labely, {b, c}, {0, 0}, {0, 1}],  
  g[[1, 2]] = g[[1, 2]] /.  
    Text[labelx, {b_, c_}, {0., -1.}] ->  
    Text[labelx, {b, c + 92.5}, {0, -1}, {1, 0}],  
  p[[n]] = Show[g,  
    AspectRatio -> 2 / 1.2 / GoldenRatio,  
    PlotRange -> All,  
    DisplayFunction -> Identity]  
}, {n, 1, 1}];
```

(* Display Sorted Eigenmatrices *)

```
Show[Table[p[[a]], {a, 1, 1}],  
      DisplayFunction -> $DisplayFunction];
```

Expression Correlation Level



(* Estimate Significance of the Association of Eigenmatrices with the Cell Cycle *)

(* Use Cell Cycle Traditional or Microarray Classification of Yeast Genes *)

```
stages = {"M/G1", "G1", "S", "S/G2", "G2/M"};
stagematrix = Outer[Times, stages, stages];
stagevector =
  Table[0, {Dimensions[stages][[1]] * (Dimensions[stages][[1]] + 1) / 2}];
n = 0;
Do[
  Do[{
    n = n + 1,
    stagevector[[n]] = stagematrix[[a, b]]
  }, {b, 1, a}],
  {a, 1, Dimensions[stages][[1]]}];

most = 100;
annotations = TakeColumns[annotatevectors, {6}];
numbers =
  Flatten[Table[{Count[Flatten[annotations], stagevector[[a]]], {a, 1, Dimensions[stagevector][[1]]}]];
counter = Table[{a}, {a, 1, 1}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];

numbers
{45, 190, 171, 20, 38, 1, 50, 95, 10, 10, 80, 152, 16, 40, 28}

Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenvectors, counter[[c]]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1]], 2]},
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}]], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1]], 2]},
  {c, 1, Dimensions[counter][[1]]}]
```

```

table1 = AppendRows [
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];

table1 = ReplaceAll[table1, "M/G1"2 -> "M/G1 M/G1"];
table1 = ReplaceAll[table1, "G1" "M/G1" -> "G1 M/G1"];
table1 = ReplaceAll[table1, "G1"2 -> "G1 G1"];
table1 = ReplaceAll[table1, "M/G1" "S" -> "M/G1 S"];
table1 = ReplaceAll[table1, "G1" "S" -> "G1 S"];
table1 = ReplaceAll[table1, "S"2 -> "S S"];
table1 = ReplaceAll[table1, "M/G1" "S/G2" -> "M/G1 S/G2"];
table1 = ReplaceAll[table1, "G1" "S/G2" -> "G1 S/G2"];
table1 = ReplaceAll[table1, "S" "S/G2" -> "S S/G2"];
table1 = ReplaceAll[table1, "S/G2"2 -> "S/G2 S/G2"];
table1 = ReplaceAll[table1, "G2/M" "M/G1" -> "G2/M M/G1"];
table1 = ReplaceAll[table1, "G1" "G2/M" -> "G1 G2/M"];
table1 = ReplaceAll[table1, "G2/M" "S" -> "G2/M S"];
table1 = ReplaceAll[table1, "G2/M" "S/G2" -> "G2/M S/G2"];
table1 = ReplaceAll[table1, "G2/M"2 -> "G2/M G2/M"];

```

(* Estimate Significance of the Association of Eigenmatrices with the Pheromone Response *)

(* Use Pheromone Microarray Classification of Yeast Genes *)

```
stages = {"Up", "Down", "None"};
stagematrix = Outer[Times, stages, stages];
stagevector =
  Table[0, {Dimensions[stages][[1]] * (Dimensions[stages][[1]] + 1) / 2}];
n = 0;
Do[
  Do[{
    n = n + 1,
    stagevector[[n]] = stagematrix[[a, b]]
  }, {b, 1, a}],
  {a, 1, Dimensions[stages][[1]]}];

most = 100;
annotations = TakeColumns[annotatevectors, {7}];
numbers =
  Flatten[Table[{Count[Flatten[annotations], stagevector[[a]]], {a, 1, Dimensions[stagevector][[1]]}}];
counter = Table[{a}, {a, 1, 1}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];

numbers
{55, 165, 105, 198, 270, 153}

Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenvectors, counter[[c]]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}]], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  {c, 1, Dimensions[counter][[1]]}]
```

```

table2 = AppendRows [
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];

table2 = ReplaceAll[table2, "Up"2 -> "Up Up"];
table2 = ReplaceAll[table2, "Down" "Up" -> "Down Up"];
table2 = ReplaceAll[table2, "Down"2 -> "Down Down"];
table2 = ReplaceAll[table2, "None" "Up" -> "None Up"];
table2 = ReplaceAll[table2, "Down" "None" -> "Down None"];
table2 = ReplaceAll[table2, "None"2 -> "None None"];

```

(* Display Significance of Association of Eigenmatrices with the Cellular Programs *)

```

headerx = {{
  ColumnForm[{" ", " ", " "}, Left],
  ColumnForm[{" ", " ", "Classification"}, Left],
  ColumnForm[{" ", " ", "Subnetwork"}, Left],
  ColumnForm[{"Most Likely", "Parallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Parallel", "Association"}, Left],
  ColumnForm[{"Most Likely", "Antiparallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Antiparallel", "Association"}, Left]},
{" ", " ", " ", " ", " ", " ", " ", " ", " ", " "}};
spacerx = {" ", " ", " ", " ", " "};
headery = Table[" ", {a, 1, 2 * Dimensions[counter][[1]] + 1}, {b, 1, 2}];
headery[[1]] = {"(a)", "Cell Cycle"};
headery[[Dimensions[counter][[1]] + 2]] = {"(b)", "Pheromone Response"};
association =
  AppendColumns[headerx,
  AppendRows[headery,
  AppendColumns[table1, spacerx, table2]]];
TableForm[association, TableSpacing -> {1, 1}]

```

	Classification	Subnetwork	Most Likely Parallel Association	P-Value of Parallel Association	Most Likely Antiparallel Association	P-Value of Antiparallel Association
(a)	Cell Cycle	1	M/G1 M/G1	1.8×10^{-9}	G1 M/G1	2.8×10^{-7}
(b)	Pheromone Response	1	Up Up	1.8×10^{-23}	Down Up	2.5×10^{-17}

```
(* Display the Cell Cycle-Projected Alpha Factor Genes x Genes Correlations Eigenmatrices *)
```

```
(* Read List of Cell Cycle-Regulated Genes *)
```

```
stream = "Desktop/Network_Decomposition/Data/Cycle_Genes.txt";  
genelist = Import[stream, "Table"];  
genelist = Drop[genelist, 1];  
partialgenes = Dimensions[Intersection[genelist, partialgenenames3]][[1]]  
Clear[stream];
```

```
44
```

```
list = Flatten[TakeColumns[genelist, {1}]];  
counter = Table[Flatten[Position[list, partialgenenames3[[a, 1]]]],  
  {a, 1, Dimensions[partialgenenames3][[1]]}];  
counter = ReplaceAll[counter, {} -> {Null}];  
partial = AppendRows[counter, partialgenenames3, eigenarrays];  
partial = Sort[partial, OrderedQ[{{#1, #2}} &];  
partialgenenames = TakeRows[  
  TakeColumns[partial, {2, 8}],  
  {1, partialgenes}];  
partialeigenarrays = TakeRows[  
  TakeColumns[partial, {9, Dimensions[eigenexpressions][[1]] + 8}],  
  {1, partialgenes}];
```

```
(* Calculate Eigenmatrices *)
```

```
partialeigenarrays = Transpose[partialeigenarrays];  
eigenmatrices = Table[0, {a, 1, Dimensions[eigenexpressions][[1]]}];  
Do[  
  eigenmatrices[[a]] = Outer[Times, partialeigenarrays[[a]], partialeigenarrays[[a]],  
    {a, 1, Dimensions[eigenexpressions][[1]]}  
  partialeigenarrays = Transpose[partialeigenarrays];
```

```
(* Keep Only Correlations Above Diagonals in Eigenmatrices *)
```

```
Do[  
  Do[  
    Do[  
      eigenmatrices[[a, b, c]] = 0,  
      {c, 1, b},  
      {b, 1, Dimensions[eigenmatrices[[1]][[1]]}],  
      {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
(* Set Correlations Cutoffs *)
```

```
cutoffs = Table[Sort[Flatten[Abs[eigenmatrices[[a]]], OrderedQ[{{#2, #1}} &][[200]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
Do[  
  eigenmatrices[[a]] = Chop[eigenmatrices[[a]], cutoffs[[a]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}
```



```
(* Select Cell Cycle-Regulated Genes With Correlations Above Cutoffs *)
```

```
int = Position[Sign[eigenmatrices[[1]]], Table[0, {a, 1, partialgenes}]]];
```

```
Do[{  
  partialeigenarrays = Drop[partialeigenarrays, {Flatten[int][[a]]}],  
  partialgenenames = Drop[partialgenenames, {Flatten[int][[a]]}],  
  {a, Dimensions[int][[1]], 1, -1}]  
partialgenes = Dimensions[partialgenenames][[1]]
```

```
35
```

```
(* Calculate Eigenmatrices For Selected Genes *)
```

```
partialeigenarrays = Transpose[partialeigenarrays];  
eigenmatrices = Table[0, {a, 1, Dimensions[eigenexpressions][[1]]}];  
Do[  
  eigenmatrices[[a]] = Outer[Times, partialeigenarrays[[a]], partialeigenarrays[[a]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}]  
partialeigenarrays = Transpose[partialeigenarrays];
```

```
(* Create Genes  $\times$  Genes Eigencorrelations Network Displays *)
```

```
(* Keep Only Correlations Above Diagonals in Eigenmatrices *)
```

```
Do[  
  Do[  
    Do[  
      eigenmatrices[[a, b, c]] = 0,  
      {c, 1, b},  
      {b, 1, Dimensions[eigenmatrices[[1]]][[1]]}],  
    {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
(* Cutoff Correlations in Eigenmatrices *)
```

```
Do[  
  eigenmatrices[[a]] = Chop[eigenmatrices[[a]], cutoffs[[a]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}]
```

```
sizes = Flatten[  
  Table[  
    Dimensions[  
      Characters[  
        ToString[partialgenenames[[a, 2]]  
        ]],  
    {a, 1, partialgenes}]];  
size = Sort[sizes, OrderedQ[{{#2, #1}} &]][[1]];  
Do[  
  Do[partialgenenames[[a, 2]] = StringJoin[ToString[partialgenenames[[a, 2]]], " "],  
  {b, 1, size - sizes[[a]]}],  
  {a, 1, partialgenes}];
```

```
(* Define Eigencorrelations Networks *)
```

```
networkgenes3 = partialgenenames;  
networks31 = eigenmatrices[[1]];
```

(* Intersect Alpha Factor and Development-Projected Alpha Factor Networks *)

```

genes = Dimensions[networkgenes1][[1]];
genenames = TakeColumns[networkgenes1, 1];

externalgenes = Dimensions[networkgenes3][[1]];
externalgenenames = TakeColumns[networkgenes3, 1];

list = Flatten[Intersection[genenames, externalgenenames]];
{partialgenes} = Dimensions[list]

{32}

counter = Table[{Position[genenames, list[[a]]][[1, 1]], list[[a]]}, {a, 1, partialgenes}];
counter = ReplaceAll[counter, {} -> {Null}];
counter = Sort[counter, OrderedQ[{{#1, #2}} &];
list = Flatten[TakeColumns[counter, {2}]];

counter = Table[Flatten[Position[list, genenames[[a, 1]]]], {a, 1, genes}];
counter = ReplaceAll[counter, {} -> {Null}];
partialmatrix = AppendRows[counter, networkgenes1, networks11, networks12, networks13, networks14];
partialmatrix = Sort[partialmatrix, OrderedQ[{{#1, #2}} &];
partialgenenames = TakeRows[
  TakeColumns[partialmatrix, {2, 8}],
  {1, partialgenes}];
partialmatrix11 = TakeRows[
  TakeColumns[partialmatrix, {9, genes + 8}],
  {1, partialgenes}];
partialmatrix12 = TakeRows[
  TakeColumns[partialmatrix, {genes + 9, 2 * genes + 8}],
  {1, partialgenes}];
partialmatrix13 = TakeRows[
  TakeColumns[partialmatrix, {2 * genes + 9, 3 * genes + 8}],
  {1, partialgenes}];
partialmatrix14 = TakeRows[
  TakeColumns[partialmatrix, {3 * genes + 9, 4 * genes + 8}],
  {1, partialgenes}];

partialmatrix11 = Transpose[partialmatrix11];
partialmatrix12 = Transpose[partialmatrix12];
partialmatrix13 = Transpose[partialmatrix13];
partialmatrix14 = Transpose[partialmatrix14];

partialmatrix = AppendRows[counter, partialmatrix11, partialmatrix12, partialmatrix13, partialmatrix14];
partialmatrix = Sort[partialmatrix, OrderedQ[{{#1, #2}} &];
partialmatrix11 = TakeRows[
  TakeColumns[partialmatrix, {2, partialgenes + 1}],
  {1, partialgenes}];
partialmatrix12 = TakeRows[
  TakeColumns[partialmatrix, {partialgenes + 2, 2 * partialgenes + 1}],
  {1, partialgenes}];
partialmatrix13 = TakeRows[
  TakeColumns[partialmatrix, {2 * partialgenes + 2, 3 * partialgenes + 1}],
  {1, partialgenes}];
partialmatrix14 = TakeRows[
  TakeColumns[partialmatrix, {3 * partialgenes + 2, 4 * partialgenes + 1}],
  {1, partialgenes}];

partialmatrix11 = Transpose[partialmatrix11];
partialmatrix12 = Transpose[partialmatrix12];
partialmatrix13 = Transpose[partialmatrix13];
partialmatrix14 = Transpose[partialmatrix14];

```

```

counter = Table[Flatten[Position[list, externalgenenames[[a, 1]]], {a, 1, externalgenes}];
counter = ReplaceAll[counter, {} -> {Null}];
partialexternalmatrix = AppendRows[counter, networks31];
partialexternalmatrix = Sort[partialexternalmatrix, OrderedQ[{{#1, #2}} &];
partialexternalmatrix31 = TakeRows[
  TakeColumns[partialexternalmatrix, {2, externalgenes + 1}],
  {1, partialgenes}];

partialexternalmatrix31 = Transpose[partialexternalmatrix31];

partialexternalmatrix = AppendRows[counter, partialexternalmatrix31];
partialexternalmatrix = Sort[partialexternalmatrix, OrderedQ[{{#1, #2}} &];
partialexternalmatrix31 = TakeRows[
  TakeColumns[partialexternalmatrix, {2, partialgenes + 1}],
  {1, partialgenes}];

partialexternalmatrix31 = Transpose[partialexternalmatrix31];

```

```
(* Define Display Parameters *)
```

```
Clear[x, y, z];
R[theta_] = {{Cos[theta], Sin[theta]}, {-Sin[theta], Cos[theta]}};
polypoints[theta_, x_] = Flatten[Transpose[Dot[R[theta], Transpose[{{0, x}}]]]];
redline[y_, z_] = Graphics[{RGBColor[1, 0, 0],
  Line[{polypoints[2 * y * Pi / partialgenes, 1], polypoints[2 * z * Pi / partialgenes, 1]}]};
greenline[y_, z_] = Graphics[{RGBColor[0, 0.5, 0],
  Line[{polypoints[2 * y * Pi / partialgenes, 1], polypoints[2 * z * Pi / partialgenes, 1]}]};

color1[stage_] =
  If[stage == "Up", RGBColor[0, 0, 0],
    If[stage == "Down", RGBColor[0.5, 0.5, 0.5], RGBColor[1, 1, 1]]
  ];
color2[stage_] =
  If[stage == "M/G1", RGBColor[1, 1, 0],
    If[stage == "G1", RGBColor[0, 0.5, 0],
      If[stage == "S", RGBColor[0, 0, 1],
        If[stage == "S/G2", RGBColor[1, 0, 0],
          If[stage == "G2/M", RGBColor[1, 0.5, 0], RGBColor[1, 1, 1]]
        ]]]];

circle1 = Table[
  Graphics[{
    color1[partialgenenames[[a, 7]]],
    Disk[{0, 0}, 1.35,
      {-2 * (a + 0.5) * Pi / partialgenes + Pi / 2., -2 * (a - 0.5) * Pi / partialgenes + Pi / 2.}],
    {a, partialgenes, 1, -1}];
circle2 = Table[
  Graphics[{
    color2[partialgenenames[[a, 6]]],
    Disk[{0, 0}, 1.3,
      {-2 * (a + 0.5) * Pi / partialgenes + Pi / 2., -2 * (a - 0.5) * Pi / partialgenes + Pi / 2.}],
    {a, partialgenes, 1, -1}];
circle3 = Graphics[{RGBColor[1, 1, 1], Disk[{0, 0}, 1.25]}];
circle4 = Table[
  Graphics[{
    RGBColor[0, 0, 0],
    Disk[polypoints[2 * a * Pi / partialgenes, 1], 0.02]}, {a, 1, partialgenes}];
circle5 = Table[
  Graphics[{
    RGBColor[0, 0, 0],
    Text[partialgenenames[[a, 2]], polypoints[2 * a * Pi / partialgenes, 1.15],
      {0, 0}, polypoints[2 * a * Pi / partialgenes, 1.15]}],
  {a, 1, partialgenes}];
```

(* Display Intersection of Development-Projected First AND Alpha Factor First Networks *)

```

int1 = Intersection[
  Position[Sign[partialmatrix11], 1],
  Position[Sign[partialexternalmatrix31], 1]];
int2 = Intersection[
  Position[Sign[partialmatrix11], -1],
  Position[Sign[partialexternalmatrix31], -1]];

redtable = Table[redline[int1[[a, 1]], int1[[a, 2]]], {a, 1, Dimensions[int1][[1]]};
greentable = Table[greenline[int2[[a, 1]], int2[[a, 2]]], {a, 1, Dimensions[int2][[1]]};
p = Show[
  {circle1, circle2, circle3, circle4, circle5, redtable, greentable,
    Graphics[{{RGBColor[0, 0, 0], Text[StyleForm["(c)", FontSize -> 24], {-1.1, 1.1}]}}],
  AspectRatio -> 1,
  PlotRange -> {{-1.25, 1.25}, {-1.25, 1.25}},
  Frame -> False,
  FrameTicks -> False,
  FrameLabel -> {None, None, None, None},
  GridLines -> {None, None},
  DisplayFunction -> Identity];
p = FullGraphics[p];
p[[1, 2]] = p[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {-1.18, 0}, {0, 0}, {0, 1}];
p3 = Show[p,
  AspectRatio -> 1.,
  PlotRange -> All,
  DisplayFunction -> Identity];

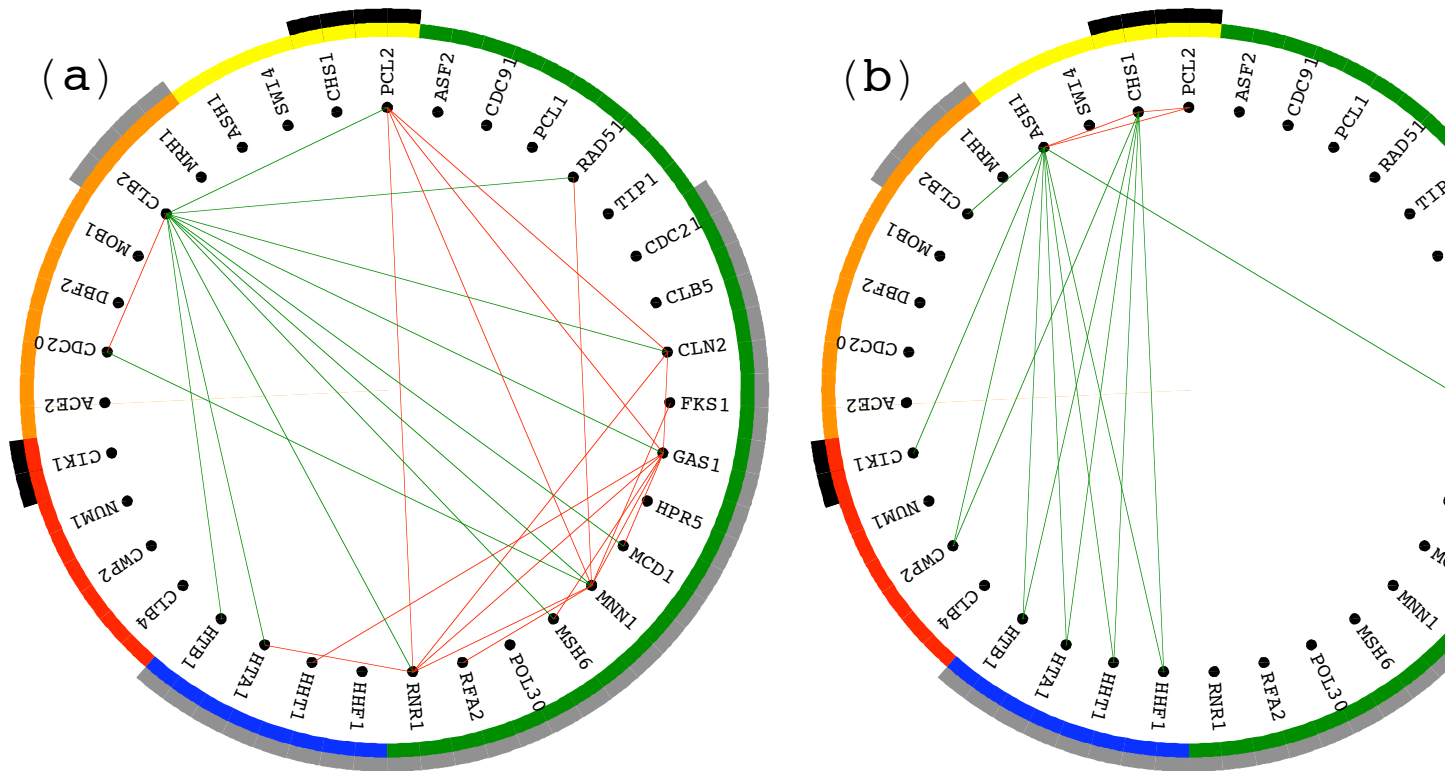
```

(* Display Selected Intersections of Genes x Genes Eigencorrelations Networks *)

```

Show[GraphicsArray[{p1, p2, p3}],
  GraphicsSpacing -> 0];

```



(* Project Genes x Genes Network Onto Biosynthesis Transcription Factors' DNA-Binding Data *)

(* Read Biosynthesis Transcription Factors' DNA-Binding Data *)

```
stream = "Desktop/Network_Decomposition/Data/Biosynthesis_Binding.txt";
matrix = Import[stream, "Table"];
{genes, arrays} = Dimensions[matrix] - {1, 7}
Clear[stream];
```

```
{2943, 8}
```

```
genenames = TakeRows[
  TakeColumns[matrix, {1, 7}],
  {2, genes + 1}];
arraynames = TakeColumns[
  TakeRows[matrix, {1, 1}],
  {8, arrays + 7}];
matrix = TakeColumns[
  TakeRows[matrix, {2, genes + 1}],
  {8, arrays + 7}];
matrix = ToExpression[matrix];
```

```
sizes = Flatten[
  Table[
    Dimensions[
      Characters[
        ToString[arraynames[[1, a]]
        ]]],
    {a, 1, arrays}]];
size = Sort[sizes, OrderedQ[{{#2, #1}} &][[1]]];
Do[
  Do[arraynames[[1, a]] = StringJoin[ToString[arraynames[[1, a]]], " "],
  {b, 1, size - sizes[[a]]},
  {a, 1, arrays}];
```

(* Convert to Ratios *)

```
average = Table[1, {a, 1, arrays}];
average = N[average / Sqrt[Dot[average, average]]];
matrix = matrix / N[Outer[Times, Dot[matrix, average], average]];
```

```
matrix4 = matrix;
genenames4 = genenames;
arraynames4 = arraynames;
{genes4, arrays4} = Dimensions[matrix4]
```

```
{2943, 8}
```

(* Calculate SVD *)

```
{eigenarrays, eigenexpressions, eigengen} = SingularValues[matrix];
eigenarrays = Transpose[eigenarrays];
fractions = eigenexpressions ^ 2 /
  Sum[eigenexpressions[[a]] ^ 2, {a, 1, Dimensions[eigenexpressions][[1]]}];
entropy = -N[Sum[fractions[[a]] * Log[fractions[[a]]],
  {a, 1, Dimensions[eigenexpressions][[1]]}] /
  Log[Dimensions[eigenexpressions][[1]]];
entropy = N[Round[100 * entropy] / 100]
```

```
0.46
```

(* Create Fractions Bar Charts Displays *)

```
fractions[[2]]
0.100777

limit = 0.1;
alsolimit = fractions[[2]];

Clear[gridx, framex, framey, sizes];
gridx = Table[a, {a, 0, limit, N[limit/5]}];
framex = gridx;
sizes = Flatten[
  Table[
    Dimensions[
      Characters[
        ToString[framex[[a]]
          ]], {a, 1, 6}]];
Do[
  Do[framex[[a]] = StringJoin[ToString[framex[[a]]], " "],
    {b, 1, 5 - sizes[[a]]},
  {a, 1, 6}];
framex = Table[{gridx[[a]], framex[[a]]}, {a, 1, 6}];
gridx = Table[{gridx[[a]], RGBColor[0, 0, 0]}, {a, 1, 6}];
framey = Table[{a + 1, arrays - a}, {a, 0, arrays - 2}];
table = Table[fractions[[arrays - a]], {a, 0, arrays - 2}];
g = BarChart[table,
  BarOrientation -> Horizontal,
  PlotRange -> {{0, alsolimit * 1.0001}, {0.5, 8 - 1 + 0.5}},
  AspectRatio -> 1,
  Axes -> False,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, None, None, None},
  GridLines -> {gridx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 1.5}, {0, 0}, {0, 1}];
g1 = Show[g,
  AspectRatio -> 1.25,
  PlotRange -> All,
  DisplayFunction -> Identity];
```

```

gridx = Table[a, {a, 0, 1, 0.2}];
framex = gridx;
sizes = Flatten[
  Table[
    Dimensions[
      Characters[
        ToString[framex[[a]]
          ]]], {a, 1, 6}];
Do[
  Do[framex[[a]] = StringJoin[ToString[framex[[a]]], " ",
    {b, 1, size - sizes[[a]]}],
    {a, 1, 6}];
framex = Table[{gridx[[a]], framex[[a]]}, {a, 1, 6}];
gridx = Table[{gridx[[a]], RGBColor[0, 0, 0]}, {a, 1, 6}];
framey = Table[{a + 1, arrays - a}, {a, 0, Dimensions[eigenexpressions][[1]]}];
labelx = ColumnForm[
  {"(b) Eigenbinding Fraction", StringJoin["di = ", ToString[entropy]], " "},
  Center];
g = BarChart[
  Table[fractions[[arrays - a]], {a, 0, arrays - 1}],
  BarOrientation -> Horizontal,
  PlotRange -> {{0, 1.0001}, {0.5, arrays + 0.5}},
  AspectRatio -> 1,
  Axes -> False,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, None, labelx, None},
  GridLines -> {gridx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 1}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 0.9375}, {0, 0}, {0, 1}];
g2 = Show[{g,
  Graphics[{RGBColor[1, 1, 0.8], Rectangle[{0.15, 0.6}, {0.98, 7.3}]}],
  Graphics[{Rectangle[{0.1, 0.6}, {0.98, 7.3}, g1]}]},
  AspectRatio -> 1.35,
  PlotRange -> All,
  DisplayFunction -> Identity];

```


(* Create Eigenenes 2 D Red & Green Raster Display *)

```

contrast = 3.5;
displaying = Table[
  If[contrast * eigengenenes[[i, j]] > 0,
    If[contrast * eigengenenes[[i, j]] < 1, {contrast * eigengenenes[[i, j]], 0}, {1, 0}],
    If[contrast * eigengenenes[[i, j]] > -1, {0, -contrast * eigengenenes[[i, j]]}, {0, 1}]],
  {i, 1, Dimensions[eigenexpressions][[1]]}, {j, 1, arrays}];
framex = Table[{a - 0.5, arraynames[[1, a]]}, {a, 1, arrays}];
framey = Table[{a + 1 - 0.5, arrays - a}, {a, 0, Dimensions[eigenexpressions][[1]]}];
labely = "Eigenenes";
labelx = ColumnForm[{"(a) Arrays", " ", " "}, Center];

g = Show[
  Graphics[
    RasterArray[
      Table[
        RGBColor[displaying[[i, j, 1]], displaying[[i, j, 2]], 0],
        {i, Dimensions[eigenexpressions][[1]], 1, -1}, {j, 1, arrays}]]],
  AspectRatio -> 1,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, labely, labelx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 2.2, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 1}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 0.9375}, {0, 0}, {0, 1}];
g1 = Show[g,
  AspectRatio -> 1.05,
  PlotRange -> All,
  DisplayFunction -> Identity];

```

(* Create Selected Eigengenes Graph Display *)

```

labelx = ColumnForm[{"(c) Arrays"}, Center];
labely = ColumnForm[{" ", "Binding Level"}, Center];
framex = Table[{a - 1, arraynames[[1, a]]}, {a, 1, arrays}];
framey = {-1, -0.5, 0, 0.5, 1};
coordinates = Table[{a - 1, eigengenes[[1, a]]}, {a, 1, arrays}];
points = Table[Point[coordinates[[a]]], {a, 1, arrays}];
line = Line[coordinates];
g = Show[
  {Graphics[{RGBColor[1, 0, 0], PointSize[0.022], points}],
   Graphics[{RGBColor[1, 0, 0], line}]},
  Frame -> True,
  FrameLabel -> {None, labely, labelx, None},
  GridLines -> {None, {{0, RGBColor[0, 0, 0]}}},
  FrameTicks -> {None, framey, framex, None},
  PlotRange -> {-1.05, 1.05},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 2, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 0.5}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 0.2}, {0, 0}, {0, 1}];
g3 = Show[g,
  AspectRatio -> 1.05,
  PlotRange -> All,
  DisplayFunction -> Identity];

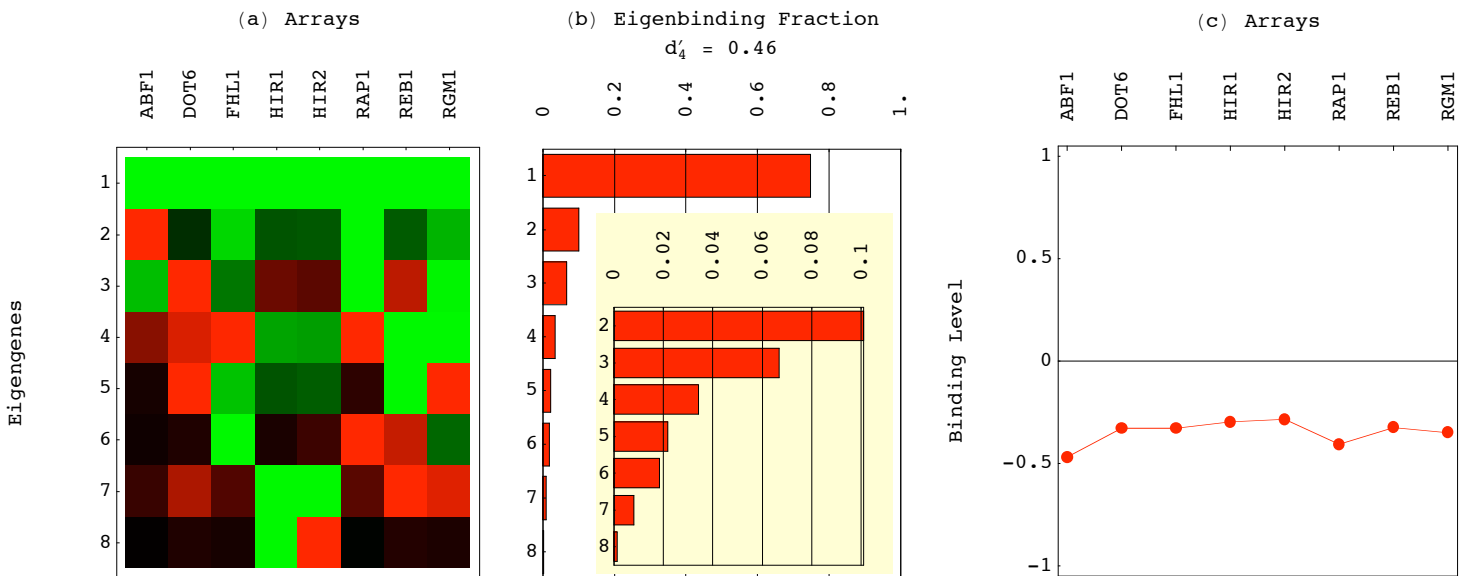
```

(* Display Eigengenes, Fractions and Selected Eigengenes *)

```

Show[GraphicsArray[{g1, g2, g3}],
  GraphicsSpacing -> -0.15];

```



```
(* Reconstruct Data Without Additive Steady State *)
```

```
{eigenarrays, eigenexpressions, eigengenes} = SingularValues[matrix];  
eigenarrays = Transpose[eigenarrays];  
eigenexpressions[[1]] = 0;  
matrix = Dot[eigenarrays, Dot[DiagonalMatrix[eigenexpressions], eigengenes]];  
  
matrix4 = matrix;  
genenames4 = genenames;  
arraynames4 = arraynames;  
{genes4, arrays4} = Dimensions[matrix4]  
  
{2943, 8}
```

```
(* Project mRNA Expression Data Onto Proteins' DNA-Binding Data *)
```

```
genes = genes1;  
genenames = TakeColumns[genenames1, 1];  
arrays = arrays1;  
arraynames = arraynames1;  
  
externalgenes = genes4;  
externalgenenames = TakeColumns[genenames4, 1];  
externalarrays = arrays4;  
externalarraynames = arraynames4;  
  
list = Flatten[Intersection[genenames, externalgenenames]];  
{partialgenes} = Dimensions[list]  
  
{2254}  
  
counter = Table[{Position[genenames, list[[a]]][[1, 1]], list[[a]]}, {a, 1, partialgenes}];  
counter = ReplaceAll[counter, {} -> {Null}];  
counter = Sort[counter, OrderedQ[{{#1, #2}} &];  
list = Flatten[TakeColumns[counter, {2}]];  
  
counter = Table[Flatten[Position[list, genenames[[a, 1]]], {a, 1, genes}];  
counter = ReplaceAll[counter, {} -> {Null}];  
partialmatrix = AppendRows[counter, genenames1, matrix1];  
partialmatrix = Sort[partialmatrix, OrderedQ[{{#1, #2}} &];  
partialgenenames = TakeRows[  
  TakeColumns[partialmatrix, {2, 8}],  
  {1, partialgenes}];  
partialmatrix = TakeRows[  
  TakeColumns[partialmatrix, {9, arrays + 8}],  
  {1, partialgenes}];  
  
counter = Table[Flatten[Position[list, externalgenenames[[a, 1]]], {a, 1, externalgenes}];  
counter = ReplaceAll[counter, {} -> {Null}];  
partialexternalmatrix = AppendRows[counter, externalgenenames, matrix4];  
partialexternalmatrix = Sort[partialexternalmatrix, OrderedQ[{{#1, #2}} &];  
partialexternalgenenames = TakeRows[  
  TakeColumns[partialexternalmatrix, {2, 2}],  
  {1, partialgenes}];  
partialexternalmatrix = TakeRows[  
  TakeColumns[partialexternalmatrix, {3, externalarrays + 2}],  
  {1, partialgenes}];  
  
partialmatrix = Dot[partialexternalmatrix, Dot[PseudoInverse[partialexternalmatrix], partialmatrix]];
```

```
(* Examine mRNA Expression Data After Projection *)
```

```
(* Calculate SVD *)
```

```
matrix = partialmatrix;  
genes = partialgenes;  
genenames = TakeColumns[partialgenenames4, 1];  
  
{eigenarrays, eigenexpressions, eigengenes} = SingularValues[matrix];  
eigengenes[[2]] = -eigengenes[[2]];  
eigenarrays[[2]] = -eigenarrays[[2]];  
eigenarrays = Transpose[eigenarrays];  
fractions = eigenexpressions^2 /  
  Sum[eigenexpressions[[a]]^2, {a, 1, Dimensions[eigenexpressions][[1]]}];  
entropy = -N[Sum[fractions[[a]] * Log[fractions[[a]]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}] /  
  Log[Dimensions[eigenexpressions][[1]]];  
entropy = N[Round[100 * entropy] / 100]
```

```
0.56
```

```
(* Create Fractions Bar Charts Display *)
```

```
fractions[[1]]
```

```
0.673729
```

```
limit = 0.75;
```

```
alsolimit = fractions[[1]];
```

```

gridx = Table[a, {a, 0, limit, N[limit/5]};
framex = gridx;
sizes = Flatten[
  Table[
    Dimensions[
      Characters[
        ToString[framex[[a]]
      ]], {a, 1, 6}];
Do[
  Do[framex[[a]] = StringJoin[ToString[framex[[a]]], " "],
    {b, 1, size - sizes[[a]]},
  {a, 1, 6}];
framex = Table[{gridx[[a]], framex[[a]]}, {a, 1, 6}];
gridx = Table[{gridx[[a]], RGBColor[0, 0, 0]}, {a, 1, 6}];
framey = Table[{a + 1, Dimensions[eigenexpressions][[1]] - a},
  {a, 0, Dimensions[eigenexpressions][[1]] - 1}];
labelx = ColumnForm[
  {"(b) Eigenexpression Fraction", StringJoin["d4 = ", ToString[entropy]], " "},
  Center];
g = BarChart[
  Table[fractions[[Dimensions[eigenexpressions][[1]] - a]],
    {a, 0, Dimensions[eigenexpressions][[1]] - 1}],
  BarOrientation -> Horizontal,
  PlotRange -> {{0, alsolimit*1.0001}, {0.5, Dimensions[eigenexpressions][[1]] + 0.5}},
  AspectRatio -> 1,
  Axes -> False,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, None, labelx, None},
  GridLines -> {gridx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 2.25}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 1.5}, {0, 0}, {0, 1}];
g2 = Show[g,
  AspectRatio -> 1.35 * 14 / 18,
  PlotRange -> All,
  DisplayFunction -> Identity];

```

(* Create Eigengenes 2D Red & Green Raster Display *)

```

contrast = 3.5;
displaying = Table[
  If[contrast * eigengenes[[i, j]] > 0,
    If[contrast * eigengenes[[i, j]] < 1, {contrast * eigengenes[[i, j]], 0}, {1, 0}],
    If[contrast * eigengenes[[i, j]] > -1, {0, -contrast * eigengenes[[i, j]]}, {0, 1}]],
  {i, 1, Dimensions[eigenexpressions][[1]]}, {j, 1, arrays}];
framex = Table[{a - 0.5, arraynames[[1, a]]}, {a, 1, arrays}];
framey = Table[{a + 1 - 0.5, Dimensions[eigenexpressions][[1]] - a},
  {a, 0, Dimensions[eigenexpressions][[1]] - 1}];
labely = "Eigengenes";
labelx = ColumnForm[{"(a) Arrays", " ", " "}, Center];

g = Show[
  Graphics[
    RasterArray[
      Table[
        RGBColor[displaying[[i, j, 1]], displaying[[i, j, 2]], 0],
        {i, Dimensions[eigenexpressions][[1]], 1, -1}, {j, 1, arrays}]]],
  AspectRatio -> 1,
  Frame -> True,
  FrameTicks -> {None, framey, framex, None},
  FrameLabel -> {None, labely, labelx, None},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 3, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 2.25}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 2}, {0, 0}, {0, 1}];
g1 = Show[g,
  AspectRatio -> 1.05 * 14 / 18,
  PlotRange -> All,
  DisplayFunction -> Identity];

```

(* Create Selected Eigengenes Graph Display *)

```
eigengenes1 = Chop[TrigFit[Drop[eigengenes[[1]], {1}], 2, {x - 1, arrays - 1}], 0.175]
eigengenes2 = Chop[TrigFit[Drop[eigengenes[[2]], {1}], 2, {x - 1, arrays - 1}], 0.125]
eigengenes3 = Chop[TrigFit[Drop[eigengenes[[3]], {1}], 2, {x - 1, arrays - 1}], 0.175]
```

```
0.242531 Sin[ $\frac{4}{17} \pi (-1 + x)$ ]
```

```
-0.142393 Cos[ $\frac{4}{17} \pi (-1 + x)$ ] + 0.193335 Sin[ $\frac{2}{17} \pi (-1 + x)$ ]
```

```
-0.193797 Cos[ $\frac{4}{17} \pi (-1 + x)$ ]
```

```
eigengenes1 = Sqrt[2 / 17.] * Sin[4 * Pi * (x - 1) / 17];
eigengenes2 = -Sqrt[2 / 17.] * Cos[4 * Pi * (x - 1) / 17];
eigengenes3 = -Sqrt[2 / 17.] * Cos[4 * Pi * (x - 1) / 17];
```

```
graph = Plot[eigengenes1,
  {x, 1, arrays - 1},
  PlotStyle -> {RGBColor[1, 0, 0], Dashing[{0.03, 0.02}]},
  DisplayFunction -> Identity];
labelx = ColumnForm[{"(c) Arrays"}, Center];
labely = ColumnForm[{" ", "Expression Level"}, Center];
framex = Table[{a - 1, arraynames[[1, a]]}, {a, 1, arrays}];
framey = {-1, -0.5, 0, 0.5, 1};
coordinates = Table[{a - 1, eigengenes[[1, a]]}, {a, 1, arrays}];
points = Table[Point[coordinates[[a]]], {a, 1, arrays}];
line = Line[coordinates];
g = Show[
  {Graphics[{RGBColor[1, 0, 0], PointSize[0.022], points}],
  Graphics[{RGBColor[1, 0, 0], line}],
  graph,
  Graphics[{{RGBColor[1, 0, 0], Text[" $\sqrt{\frac{2}{T}} \sin(\frac{4 \pi t}{T})$ ", {8.5, 0.75}]}]}],
  Frame -> True,
  FrameLabel -> {None, labely, labelx, None},
  GridLines -> {None, {{0, RGBColor[0, 0, 0]}}},
  FrameTicks -> {None, framey, framex, None},
  PlotRange -> {-1.05, 1.05},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 5.4, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 0.625}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 0.25}, {0, 0}, {0, 1}];
p1 = Show[g,
  AspectRatio -> 1.05,
  PlotRange -> All,
  DisplayFunction -> Identity];
```

```

graph = Plot[eigengenes2,
  {x, 1, arrays - 1},
  PlotStyle -> {RGBColor[0, 0, 1], Dashing[{0.03, 0.02}]},
  DisplayFunction -> Identity];
labelx = ColumnForm[{"(d) Arrays"}, Center];
labely = ColumnForm[{" ", " "}, Center];
framex = Table[{a - 1, arraynames[[1, a]]}, {a, 1, arrays}];
framey = {-1, -0.5, 0, 0.5, 1};
coordinates = Table[{a - 1, eigengenes[[2, a]]}, {a, 1, arrays}];
points = Table[Point[coordinates[[a]]], {a, 1, arrays}];
line = Line[coordinates];
g = Show[
  {Graphics[{RGBColor[0, 0, 1], PointSize[0.022], points}],
  Graphics[{RGBColor[0, 0, 1], line}],
  graph,
  Graphics[{RGBColor[0, 0, 1], Text["- $\sqrt{\frac{2}{T}} \cos(\frac{4 \pi t}{T})$ ", {8.5, 0.75}]}]},
  Frame -> True,
  FrameLabel -> {None, labely, labelx, None},
  GridLines -> {None, {{0, RGBColor[0, 0, 0]}}},
  FrameTicks -> {None, framey, framex, None},
  PlotRange -> {-1.05, 1.05},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 5.4, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 0.625}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 0.25}, {0, 0}, {0, 1}];
p2 = Show[g,
  AspectRatio -> 1.05,
  PlotRange -> All,
  DisplayFunction -> Identity];

```



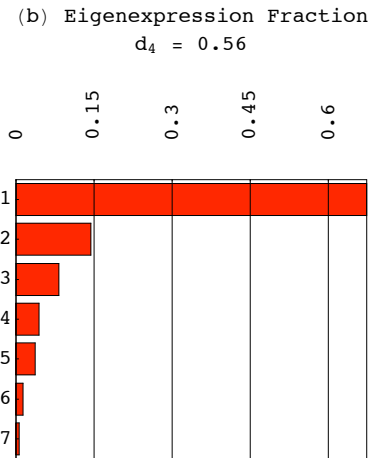
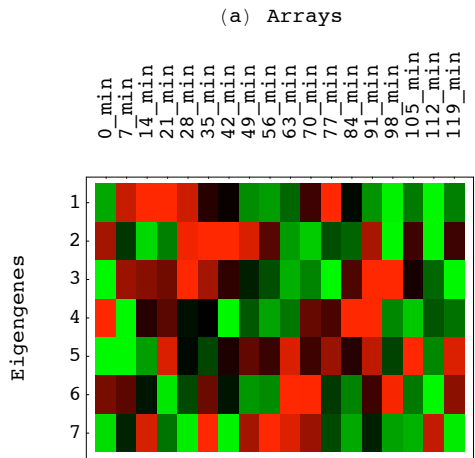
```

graph = Plot[eigengenes3,
  {x, 1, arrays - 1},
  PlotStyle -> {RGBColor[0, 0.5, 0], Dashing[{0.03, 0.02}]},
  DisplayFunction -> Identity];
labelx = ColumnForm[{"(e) Arrays"}, Center];
labely = ColumnForm[{" ", " "}, Center];
framex = Table[{a - 1, arraynames[[1, a]]}, {a, 1, arrays}];
framey = {-1, -0.5, 0, 0.5, 1};
coordinates = Table[{a - 1, eigengenes[[3, a]]}, {a, 1, arrays}];
points = Table[Point[coordinates[[a]]], {a, 1, arrays}];
line = Line[coordinates];
g = Show[
  {Graphics[{RGBColor[0, 0.5, 0], PointSize[0.022], points}],
  Graphics[{RGBColor[0, 0.5, 0], line}],
  graph,
  Graphics[{{RGBColor[0, 0.5, 0], Text[" $-\sqrt{\frac{2}{T}} \cos(\frac{4\pi t}{T})$ ", {8.5, 0.75}]}]}],
  Frame -> True,
  FrameLabel -> {None, labely, labelx, None},
  GridLines -> {None, {{0, RGBColor[0, 0, 0]}}},
  FrameTicks -> {None, framey, framex, None},
  PlotRange -> {-1.05, 1.05},
  DisplayFunction -> Identity];
g = FullGraphics[g];
g[[1, 2]] = g[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {b - 5.4, c}, {0, 0}, {0, 1}];
g[[1, 2]] = g[[1, 2]] /.
  Text[labelx, {b_, c_}, {0., -1.}] ->
  Text[labelx, {b, c + 0.625}, {0, -1}, {1, 0}];
g[[1, 2]] = g[[1, 2]] /.
  Text[a_, {b_, c_}, {0., -1.}] ->
  Text[a, {b, c + 0.25}, {0, 0}, {0, 1}];
p3 = Show[g,
  AspectRatio -> 1.05,
  PlotRange -> All,
  DisplayFunction -> Identity];

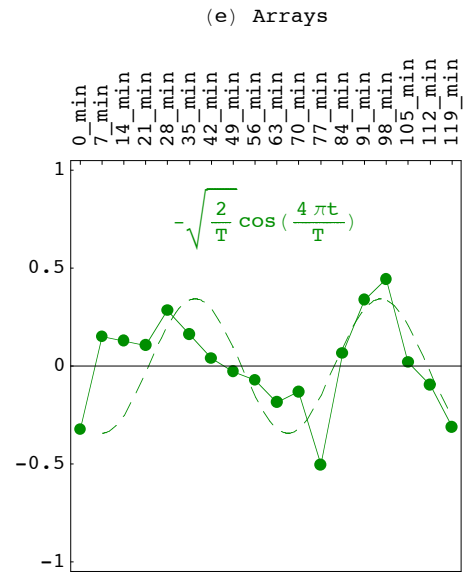
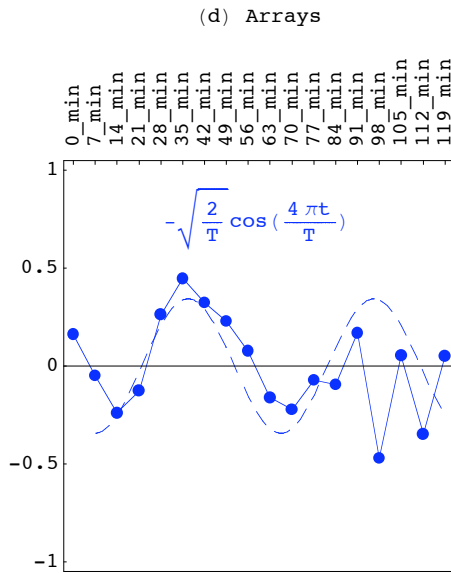
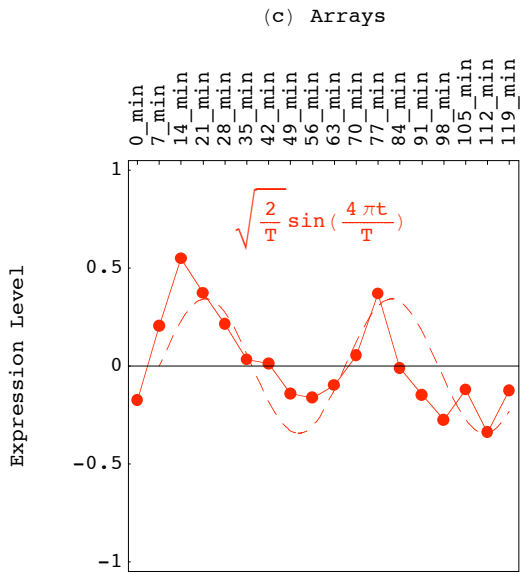
```

(* Display Eigengenes, Fractions and Selected Eigengenes *)

```
Show[GraphicsArray[{g1, g2}],
GraphicsSpacing -> -0.15];
```



```
Show[GraphicsArray[{p1, p2, p3}],
GraphicsSpacing -> -0.15];
```



(* Define Projected Data *)

```
partialmatrix4 = partialmatrix;
partialgenenames4 = partialgenenames;
```

```
(* Display Sorted Significant Eigenarrays *)
```

```
arraypatterns = Transpose[eigenarrays];
```

```
(* Center Eigenarrays *)
```

```
average = Table[1, {a, 1, genes}];
```

```
average = N[average / Sqrt[Dot[average, average]]];
```

```
arraypatterns = arraypatterns - N[Outer[Times, Dot[arraypatterns, average], average]];
```

```
(* Sort Eigenarrays *)
```

```
Do[
```

```
arraypatterns[[a]] = Sort[arraypatterns[[a]], OrderedQ[{{#2}, {#1}}] &],  
{a, 1, Dimensions[eigenexpressions][[1]]}]
```

```
(* Create Sorted Eigenarrays Graph Display *)
```

```
p = Table[0, {a, 1, 3}];
```

```
color = {
```

```
RGBColor[1, 0.5, 0],
```

```
RGBColor[1, 0, 0],
```

```
RGBColor[0, 0, 1],
```

```
RGBColor[0, 0.5, 0]};
```

```
labelx = "Expression Level";
```

```
labeled = ColumnForm[
```

```
{ "Number of Genes", " ", " ", " ", " ", " ", " ", " ", " ", " ", " ", " ",  
Center];
```

```
framex = Table[{0.02 * a, a}, {a, 1, 3}];
```

```
framey = {{-150, "150"}, {-1000, "1000"}, {-genes + 150, "2104"}};
```

```
Do[ {
```

```
coordinates = Table[
```

```
If[arraypatterns[[n, a]] + 0.02 * n < -0.04, -0.04,
```

```
If[arraypatterns[[n, a]] + 0.02 * n > 0.12, 0.12,
```

```
arraypatterns[[n, a]] + 0.02 * n]],
```

```
{a, 1, genes}],
```

```
coordinates = Table[{coordinates[[a]], -a + 1}, {a, 1, genes}],
```

```
line = Line[coordinates],
```

```
g = Show[
```

```
Graphics[{color[[Mod[n, 4] + 1]], line}],
```

```
Frame -> True,
```

```
FrameLabel -> {None, labeled, labelx, None},
```

```
FrameTicks -> {None, framey, framex, None},
```

```
GridLines -> {{{0.02 * n, RGBColor[0, 0, 0]}}, {{-150, RGBColor[0, 0, 0]},
```

```
{-1000, RGBColor[0, 0, 0]}, {-genes + 150, RGBColor[0, 0, 0]}},
```

```
PlotRange -> {{-0.04, 0.12}, {81.5, -genes + 1 - 81.5}},
```

```
DisplayFunction -> Identity],
```

```
g = FullGraphics[g],
```

```
g[[1, 2]] = g[[1, 2]] /.
```

```
Text[labeled, {b_, c_}, {1., 0.}] ->
```

```
Text[labeled, {b, c}, {0, 0}, {0, 1}],
```

```
g[[1, 2]] = g[[1, 2]] /.
```

```
Text[labelx, {b_, c_}, {0., -1.}] ->
```

```
Text[labelx, {b, c + 217.5}, {0, -1}, {1, 0}],
```

```
p[[n]] = Show[g,
```

```
AspectRatio -> 2 / 1.2 / GoldenRatio,
```

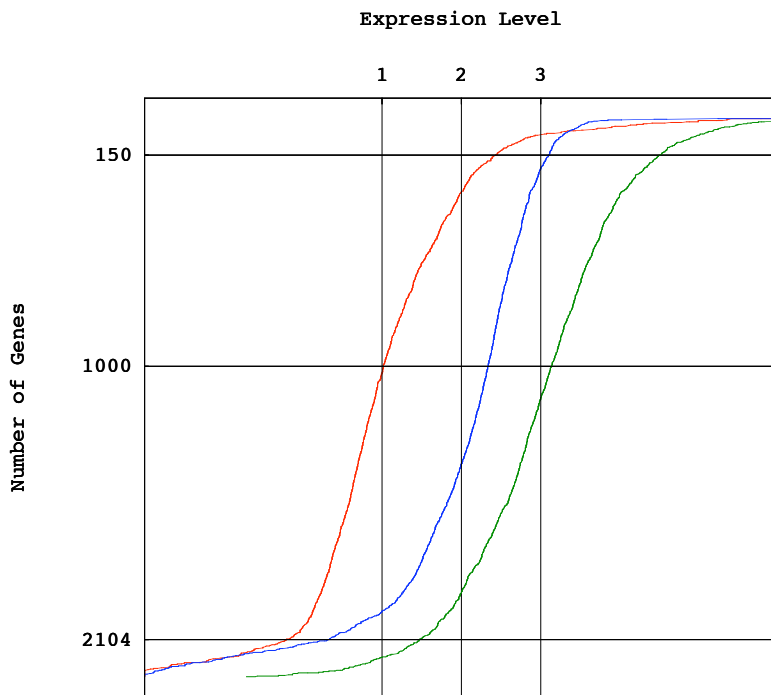
```
PlotRange -> All,
```

```
DisplayFunction -> Identity]
```

```
}, {n, 1, 3}];
```

(* Display Sorted Eigenarrays *)

```
Show[Table[p[[a]], {a, 1, 3}],  
      DisplayFunction -> $DisplayFunction];
```



(* Estimate Significance of Association of Eigenarrays with the Cell Cycle *)

(* Use Cell Cycle Traditional or Microarray Classification of Yeast Genes *)

```
most = 150;
annotations = TakeColumns[partialgenenames, {6}];
stages = {"M/G1", "G1", "S", "S/G2", "G2/M", "None"};
numbers = Flatten[Table[{Count[Flatten[annotations], stages[[a]]}], {a, 1, Dimensions[stages][[1]]}]]
counter = Table[{a}, {a, 1, 3}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];

{42, 111, 30, 48, 75, 1948}

Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenarrays, counter[[c]]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3]], most}],
    stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  {c, 1, Dimensions[counter][[1]]}]

table1 = AppendRows[
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];
```

(* Estimate Significance of Association of Eigenarrays with the Pheromone Response *)

(* Use Pheromone Microarray Classification of Yeast Genes *)

```
most = 150;
annotations = TakeColumns[partialgenenames, {7}];
stages = {"Up", "Down", "None"};
numbers = Flatten[Table[{Count[Flatten[annotations], stages[[a]]]}, {a, 1, Dimensions[stages][[1]]}]]
counter = Table[{a}, {a, 1, 3}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];

{58, 91, 2105}

Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenarrays, counter[[c]]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3], most]},
    stages[[a]],
    {a, 1, Dimensions[stages][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  table = Table[{
    stages[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}]], stages[[a]]],
    {a, 1, Dimensions[stages][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
      Binomial[genes, most]], {b, table[[a, 3], most]},
    stages[[a]],
    {a, 1, Dimensions[stages][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  {c, 1, Dimensions[counter][[1]]}]

table2 = AppendRows [
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];
```

(* Display Significance of Association of Eigenarrays with the Cellular Programs *)

```

headerx = {{
  ColumnForm[{" ", " ", " "}, Left],
  ColumnForm[{" ", " ", "Classification"}, Left],
  ColumnForm[{" ", " ", "Eigenarray"}, Left],
  ColumnForm[{"Most Likely", "Parallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Parallel", "Association"}, Left],
  ColumnForm[{"Most Likely", "Antiparallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Antiparallel", "Association"}, Left]},
{" ", " ", " ", " ", " ", " ", " ", " ", " ", " "}};
spacerx = {" ", " ", " ", " ", " "};
headery = Table[" ", {a, 1, 2 * Dimensions[counter][[1]] + 1}, {b, 1, 2}];
headery[[1]] = {"(a)", "Cell Cycle"};
headery[[Dimensions[counter][[1]] + 2]] = {"(b)", "Pheromone Response"};
association =
  AppendColumns[headerx,
  AppendRows[headery,
  AppendColumns[table1, spacerx, table2]]];
TableForm[association, TableSpacing -> {1, 1}]

```

	Classification	Eigenarray	Most Likely Parallel Association	P-Value of Parallel Association	Most Likely Antiparallel Association	P-Value of Antiparallel Association
(a)	Cell Cycle	1	G1	$9. \times 10^{-8}$	None	1.9×10^{-3}
		2	S	5.3×10^{-4}	None	2.1×10^{-2}
		3	G2/M	1.9×10^{-5}	G1	$3. \times 10^{-2}$
(b)	Pheromone Response	1	Down	2.6×10^{-1}	None	1.2×10^{-1}
		2	Down	7.6×10^{-2}	None	7.6×10^{-3}
		3	Down	6.3×10^{-5}	Down	7.6×10^{-2}

```
(* Examine the Biosynthesis-Projected Alpha Factor Genes x Genes Correlations Eigenmatrices *)
```

```
stream = "Desktop/Network_Decomposition/Data/Cycle_Genes.txt";  
genelist = Import[stream, "Table"];  
genelist = Drop[genelist, 1];  
partialgenes = Dimensions[Intersection[genelist, partialgenenames4]][[1]]  
Clear[stream];
```

39

```
list = Flatten[TakeColumns[genelist, {1}]];  
counter = Table[Flatten[Position[list, partialgenenames4[[a, 1]]],  
  {a, 1, Dimensions[partialgenenames4][[1]]}];  
counter = ReplaceAll[counter, {} -> {Null}];  
partial = AppendRows[counter, partialgenenames4, eigenarrays];  
partial = Sort[partial, OrderedQ[{{#1, #2}} &];  
partialgenenames = TakeRows[  
  TakeColumns[partial, {2, 8}],  
  {1, partialgenes}];  
partialeigenarrays = TakeRows[  
  TakeColumns[partial, {9, Dimensions[eigenexpressions][[1]] + 8}],  
  {1, partialgenes}];
```

```
(* Calculate Eigenmatrices *)
```

```
partialeigenarrays = Transpose[partialeigenarrays];  
eigenmatrices = Table[0, {a, 1, Dimensions[eigenexpressions][[1]]}];  
Do[  
  eigenmatrices[[a]] = Outer[Times, partialeigenarrays[[a]], partialeigenarrays[[a]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}];  
partialeigenarrays = Transpose[partialeigenarrays];
```

```
(* Keep Only Correlations Above Diagonals in Eigenmatrices *)
```

```
Do[  
  Do[  
    Do[  
      eigenmatrices[[a, b, c]] = 0,  
      {c, 1, b}],  
    {b, 1, Dimensions[eigenmatrices][[1]][[1]]},  
    {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
(* Flatten Eigenmatrices Into Eigenvectors *)
```

```
eigenvectors = Table[0, {partialgenes * (partialgenes - 1) / 2, {Dimensions[eigenexpressions][[1]]}];  
Do[{  
  n = 0,  
  square = eigenmatrices[[a]],  
  Do[{  
    line = square[[b]],  
    Do[{  
      n = n + 1,  
      eigenvectors[[n, a]] = line[[c]]  
    }, {c, b + 1, partialgenes}],  
  }, {b, 1, partialgenes}],  
{a, 1, Dimensions[eigenexpressions][[1]]}];
```



```
(* Create Genes × Genes Annotation Matrices for Selected Genes *)
```

```
partialgenenames = Transpose[partialgenenames];  
annotatematrices = Table[  
  Table[0, {partialgenes}, {partialgenes}],  
  {a, 1, 7}];  
Do[  
  annotatematrices[[a]] = Outer[Times, partialgenenames[[a]], partialgenenames[[a]],  
  {a, 1, 7}]
```

```
(* Keep Only Correlations Above Diagonals in Annotation Matrices *)
```

```
Do[  
  Do[  
    Do[  
      annotatematrices[[a, b, c]] = 0,  
      {c, 1, b}],  
    {b, 1, partialgenes}],  
  {a, 1, 7}];
```

```
(* Flatten Annotation Matrices Into Annotation Vectors *)
```

```
annotatevectors = Table[0, {partialgenes * (partialgenes - 1) / 2}, {7}];  
Do[{  
  n = 0,  
  square = annotatematrices[[a]],  
  Do[{  
    line = square[[b]],  
    Do[{  
      n = n + 1,  
      annotatevectors[[n, a]] = line[[c]]  
    }, {c, b + 1, partialgenes}],  
  }, {b, 1, partialgenes}]  
}, {a, 1, 7}]  
partialgenenames = Transpose[partialgenenames];
```

```
(* Display Sorted Significant Eigenmatrices *)
```

```
arraypatterns = Transpose[eigenvectors];  
genes = Dimensions[arraypatterns][[2]]
```

```
741
```

```
(* Center Eigenmatrices *)
```

```
average = Table[1, {a, 1, genes}];  
average = N[average / Sqrt[Dot[average, average]]];  
arraypatterns = arraypatterns - N[Outer[Times, Dot[arraypatterns, average], average]];
```

```
(* Sort Eigenmatrices *)
```

```
Do[  
  arraypatterns[[a]] = Sort[arraypatterns[[a]], OrderedQ[{{#2}, {#1}}] &],  
  {a, 1, Dimensions[eigenexpressions][[1]]}]
```

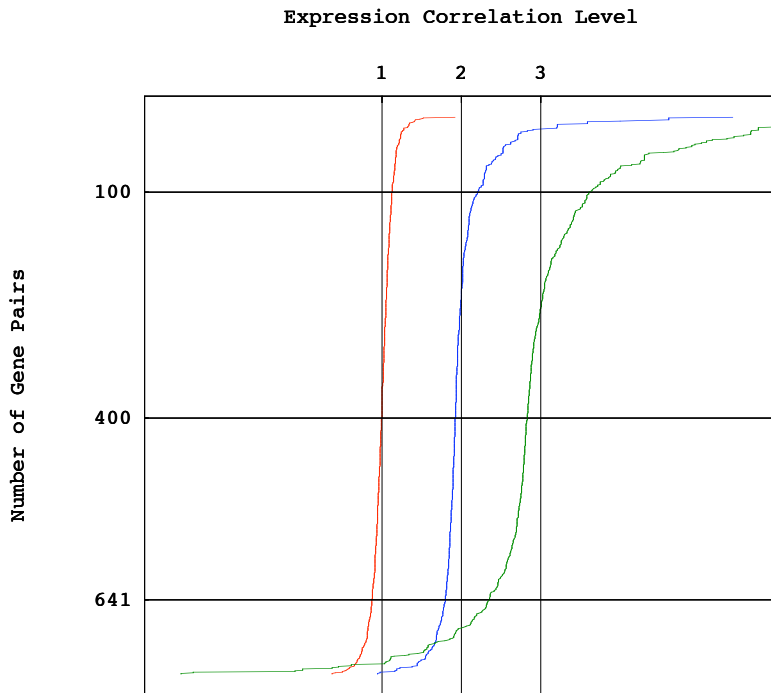
```
(* Create Sorted Eigenmatrices Graph Display *)
```

```
p = Table[0, {a, 1, 3}];  
color = {  
  RGBColor[1, 0.5, 0],  
  RGBColor[1, 0, 0],  
  RGBColor[0, 0, 1],  
  RGBColor[0, 0.5, 0]};  
labelx = "Expression Correlation Level";  
labely = ColumnForm[  
  {"Number of Gene Pairs", " ", " ", " ", " ", " ", " ", " ", " ", " ", " "},  
  Center];  
framex = Table[{0.002 * a, a}, {a, 1, 3}];  
framey = {{-100, "100"}, {-400, "400"}, {-genes + 100, "641"}};
```

```
Do[{  
  coordinates = Table[  
    If[arraypatterns[[n, a]] + 0.002 * n < -0.004, -0.004,  
      If[arraypatterns[[n, a]] + 0.002 * n > 0.012, 0.012,  
        arraypatterns[[n, a]] + 0.002 * n]],  
    {a, 1, genes}],  
  coordinates = Table[{coordinates[[a]], -a + 1}, {a, 1, genes}],  
  line = Line[coordinates],  
  g = Show[  
    Graphics[{color[[Mod[n, 4] + 1]], line}],  
    Frame -> True,  
    FrameLabel -> {None, labely, labelx, None},  
    FrameTicks -> {None, framey, framex, None},  
    GridLines -> {{{0.002 * n, RGBColor[0, 0, 0]}}, {{-100, RGBColor[0, 0, 0]},  
      {-400, RGBColor[0, 0, 0]}, {-genes + 100, RGBColor[0, 0, 0]}}},  
    PlotRange -> {{-0.004, 0.012}, {27.5, -genes + 1 - 27.5}},  
    DisplayFunction -> Identity],  
  g = FullGraphics[g],  
  g[[1, 2]] = g[[1, 2]] /.  
    Text[labely, {b_, c_}, {1., 0.}] ->  
    Text[labely, {b, c}, {0, 0}, {0, 1}],  
  g[[1, 2]] = g[[1, 2]] /.  
    Text[labelx, {b_, c_}, {0., -1.}] ->  
    Text[labelx, {b, c + 72.5}, {0, -1}, {1, 0}],  
  p[[n]] = Show[g,  
    AspectRatio -> 2 / 1.2 / GoldenRatio,  
    PlotRange -> All,  
    DisplayFunction -> Identity]  
}, {n, 1, 3}];
```

(* Display Sorted Eigenmatrices *)

```
Show[Table[p[[a]], {a, 1, 3}],  
      DisplayFunction -> $DisplayFunction];
```



(* Estimate Significance of the Association of Eigenmatrices with the Cell Cycle *)

(* Use Cell Cycle Traditional or Microarray Classification of Yeast Genes *)

```
stages = {"M/G1", "G1", "S", "S/G2", "G2/M"};
stagematrix = Outer[Times, stages, stages];
stagevector =
  Table[0, {Dimensions[stages][[1]] * (Dimensions[stages][[1]] + 1) / 2}];
n = 0;
Do[
  Do[{
    n = n + 1,
    stagevector[[n]] = stagematrix[[a, b]]
  }, {b, 1, a}],
  {a, 1, Dimensions[stages][[1]]}];

most = 100;
annotations = TakeColumns[annotatevectors, {6}];
numbers = Flatten[Table[
  {Count[Flatten[annotations], stagevector[[a]]],
  {a, 1, Dimensions[stagevector][[1]]}]]];
counter = Table[{a}, {a, 1, 3}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];

{21, 77, 55, 56, 88, 28, 28, 44, 32, 6, 63, 99, 72, 36, 36}

Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenvectors, counter[[c]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
    Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}]], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
    Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  {c, 1, Dimensions[counter][[1]]}];
```

```

table1 = AppendRows [
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];

table1 = ReplaceAll[table1, "M/G1"2 -> "M/G1 M/G1"];
table1 = ReplaceAll[table1, "G1" "M/G1" -> "G1 M/G1"];
table1 = ReplaceAll[table1, "G1"2 -> "G1 G1"];
table1 = ReplaceAll[table1, "M/G1" "S" -> "M/G1 S"];
table1 = ReplaceAll[table1, "G1" "S" -> "G1 S"];
table1 = ReplaceAll[table1, "S"2 -> "S S"];
table1 = ReplaceAll[table1, "M/G1" "S/G2" -> "M/G1 S/G2"];
table1 = ReplaceAll[table1, "G1" "S/G2" -> "G1 S/G2"];
table1 = ReplaceAll[table1, "S" "S/G2" -> "S S/G2"];
table1 = ReplaceAll[table1, "S/G2"2 -> "S/G2 S/G2"];
table1 = ReplaceAll[table1, "G2/M" "M/G1" -> "G2/M M/G1"];
table1 = ReplaceAll[table1, "G1" "G2/M" -> "G1 G2/M"];
table1 = ReplaceAll[table1, "G2/M" "S" -> "G2/M S"];
table1 = ReplaceAll[table1, "G2/M" "S/G2" -> "G2/M S/G2"];
table1 = ReplaceAll[table1, "G2/M"2 -> "G2/M G2/M"];

```

(* Estimate Significance of the Association of Eigenmatrices with the Pheromone Response *)

(* Use Pheromone Microarray Classification of Yeast Genes *)

```
stages = {"Up", "Down", "None"};
stagematrix = Outer[Times, stages, stages];
stagevector =
  Table[0, {Dimensions[stages][[1]] * (Dimensions[stages][[1]] + 1) / 2}];
n = 0;
Do[
  Do[{
    n = n + 1,
    stagevector[[n]] = stagematrix[[a, b]]
  }, {b, 1, a}],
  {a, 1, Dimensions[stages][[1]]}];

most = 100;
annotations = TakeColumns[annotatevectors, {7}];
numbers = Flatten[Table[
  {Count[Flatten[annotations], stagevector[[a]]],
  {a, 1, Dimensions[stagevector][[1]]}]]];
counter = Table[{a}, {a, 1, 3}];
probability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
parallelprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiannotation = Table[{0}, {a, 1, Dimensions[counter][[1]]}];
antiprobability = Table[{0}, {a, 1, Dimensions[counter][[1]]}];

{10, 100, 190, 70, 280, 91}

Do[{
  pattern = TakeColumns[Sort[
    AppendRows[TakeColumns[eigenvectors, counter[[c]], annotations],
    OrderedQ[{{#2}, {#1}}] &], {2}],
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {1, most}]], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
    Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  parallelannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  parallelprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  table = Table[{
    stagevector[[a]],
    numbers[[a]],
    Count[Flatten[TakeRows[pattern, {genes - most + 1, genes}], stagevector[[a]]],
    {a, 1, Dimensions[stagevector][[1]]}],
  probability = Table[{
    Sum[N[Binomial[table[[a, 2]], b] * Binomial[genes - table[[a, 2]], most - b] /
    Binomial[genes, most]], {b, table[[a, 3]], most}],
    stagevector[[a]],
    {a, 1, Dimensions[stagevector][[1]]}],
  antiannotation[[c]] = {Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 2]]},
  antiprobability[[c]] = {ScientificForm[Sort[probability, OrderedQ[{{#1}, {#2}}] &][[1, 1], 2]}],
  {c, 1, Dimensions[counter][[1]]}];
```

```

table2 = AppendRows [
  counter,
  parallelannotation,
  parallelprobability,
  antiannotation,
  antiprobability];

table2 = ReplaceAll[table2, "Up"2 -> "Up  Up"];
table2 = ReplaceAll[table2, "Down" "Up" -> "Down Up"];
table2 = ReplaceAll[table2, "Down"2 -> "Down Down"];
table2 = ReplaceAll[table2, "None" "Up" -> "None Up"];
table2 = ReplaceAll[table2, "Down" "None" -> "Down None"];
table2 = ReplaceAll[table2, "None"2 -> "None None"];

(* Display Significance of Association of Eigenmatrices with the Cellular Programs *)

```

```

headerx = {{
  ColumnForm[{" ", " ", " "}, Left],
  ColumnForm[{" ", " ", "Classification"}, Left],
  ColumnForm[{" ", " ", "Subnetwork"}, Left],
  ColumnForm[{"Most Likely", "Parallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Parallel", "Association"}, Left],
  ColumnForm[{"Most Likely", "Antiparallel", "Association"}, Left],
  ColumnForm[{"P-Value of", "Antiparallel", "Association"}, Left]},
{" ", " ", " ", " ", " ", " ", " ", " ", " ", " "}};
spacerx = {" ", " ", " ", " ", " "};
headery = Table[" ", {a, 1, 2 * Dimensions[counter][[1]] + 1}, {b, 1, 2}];
headery[[1]] = {"(a)", "Cell Cycle"};
headery[[Dimensions[counter][[1]] + 2]] = {"(b)", "Pheromone Response"};
association =
  AppendColumns[headerx,
  AppendRows[headery,
  AppendColumns[table1, spacerx, table2]]];
TableForm[association, TableSpacing -> {1, 1}]

```

	Classification	Subnetwork	Most Likely Parallel Association	P-Value of Parallel Association	Most Likely Antiparallel Association	P-Value of Antiparallel Association
(a)	Cell Cycle	1	G1 S/G2	2.6×10^{-4}	S S/G2	$4. \times 10^{-11}$
		2	S S	2.8×10^{-24}	M/G1 S	$4. \times 10^{-7}$
		3	S S	1.1×10^{-26}	G1 S	1.6×10^{-17}
(b)	Pheromone Response	1	Down Down	4.6×10^{-3}	Down Down	7.6×10^{-2}
		2	Down Down	5.1×10^{-11}	Down Up	5.9×10^{-7}
		3	Down Down	$4. \times 10^{-23}$	Down Down	$9. \times 10^{-10}$

```
(* Display the Cell Cycle-Projected Alpha Factor Genes x Genes Correlations Eigenmatrices *)
```

```
(* Read List of Cell Cycle-Regulated Genes *)
```

```
stream = "Desktop/Network_Decomposition/Data/Cycle_Genes.txt";  
genelist = Import[stream, "Table"];  
genelist = Drop[genelist, 1];  
partialgenes = Dimensions[Intersection[genelist, partialgenenames4]][[1]]  
Clear[stream];
```

39

```
list = Flatten[TakeColumns[genelist, {1}]];  
counter = Table[Flatten[Position[list, partialgenenames4[[a, 1]]]],  
  {a, 1, Dimensions[partialgenenames4][[1]]}];  
counter = ReplaceAll[counter, {} -> {Null}];  
partial = AppendRows[counter, partialgenenames4, eigenarrays];  
partial = Sort[partial, OrderedQ[{{#1, #2}} &];  
partialgenenames = TakeRows[  
  TakeColumns[partial, {2, 8}],  
  {1, partialgenes}];  
partialeigenarrays = TakeRows[  
  TakeColumns[partial, {9, Dimensions[eigenexpressions][[1]] + 8}],  
  {1, partialgenes}];
```

```
(* Calculate Eigenmatrices *)
```

```
partialeigenarrays = Transpose[partialeigenarrays];  
eigenmatrices = Table[0, {a, 1, Dimensions[eigenexpressions][[1]]}];  
Do[  
  eigenmatrices[[a]] = Outer[Times, partialeigenarrays[[a]], partialeigenarrays[[a]],  
    {a, 1, Dimensions[eigenexpressions][[1]]}  
  partialeigenarrays = Transpose[partialeigenarrays];
```

```
(* Keep Only Correlations Above Diagonals in Eigenmatrices *)
```

```
Do[  
  Do[  
    Do[  
      eigenmatrices[[a, b, c]] = 0,  
        {c, 1, b}],  
    {b, 1, Dimensions[eigenmatrices[[1]][[1]]}],  
    {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
(* Set Correlations Cutoffs *)
```

```
cutoffs = Table[Sort[Flatten[Abs[eigenmatrices[[a]]], OrderedQ[{{#2, #1}} &][[200]],  
  {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
Do[  
  eigenmatrices[[a]] = Chop[eigenmatrices[[a]], cutoffs[[a]],  
    {a, 1, Dimensions[eigenexpressions][[1]]}
```



```
(* Select Cell Cycle-Regulated Genes With Correlations Above Cutoffs *)
```

```
int = Intersection[
  Position[Sign[eigenmatrices[[1]]], Table[0, {a, 1, partialgenes}]],
  Position[Sign[eigenmatrices[[2]]], Table[0, {a, 1, partialgenes}]],
  Position[Sign[eigenmatrices[[3]]], Table[0, {a, 1, partialgenes}]]];
```

```
Do[{
  partialeigenarrays = Drop[partialeigenarrays, {Flatten[int][[a]]}],
  partialgenenames = Drop[partialgenenames, {Flatten[int][[a]]}],
  {a, Dimensions[int][[1]], 1, -1}]
partialgenes = Dimensions[partialgenenames][[1]]
```

32

```
(* Calculate Eigenmatrices For Selected Genes *)
```

```
partialeigenarrays = Transpose[partialeigenarrays];
eigenmatrices = Table[0, {a, 1, Dimensions[eigenexpressions][[1]]}];
Do[
  eigenmatrices[[a]] = Outer[Times, partialeigenarrays[[a]], partialeigenarrays[[a]],
  {a, 1, Dimensions[eigenexpressions][[1]]}];
partialeigenarrays = Transpose[partialeigenarrays];
```

```
(* Create Genes x Genes Eigencorrelations Network Displays *)
```

```
(* Keep Only Correlations Above Diagonals in Eigenmatrices *)
```

```
Do[
  Do[
    Do[
      eigenmatrices[[a, b, c]] = 0,
      {c, 1, b}],
    {b, 1, Dimensions[eigenmatrices[[1]][[1]]}],
  {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
(* Cutoff Correlations in Eigenmatrices *)
```

```
Do[
  eigenmatrices[[a]] = Chop[eigenmatrices[[a]], cutoffs[[a]],
  {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
Do[
  Do[
    Do[
      eigenmatrices[[a, b, c]] = 0,
      {c, 1, b}],
    {b, 1, Dimensions[eigenmatrices[[1]][[1]]}],
  {a, 1, Dimensions[eigenexpressions][[1]]}];
```

```
sizes = Flatten[
  Table[
    Dimensions[
      Characters[
        ToString[partialgenenames[[a, 2]]
        ]],
    {a, 1, partialgenes}]];
size = Sort[sizes, OrderedQ[{{#2, #1}} &]][[1]];
```

```
Do[
  Do[partialgenenames[[a, 2]] = StringJoin[ToString[partialgenenames[[a, 2]]], " "],
  {b, 1, size - sizes[[a]]}],
  {a, 1, partialgenes}];
```

```
(* Define Display Parameters *)
```

```
Clear[x, y, z];
R[theta_] = {{Cos[theta], Sin[theta]}, {-Sin[theta], Cos[theta]}};
polypoints[theta_, x_] = Flatten[Transpose[Dot[R[theta], Transpose[{{0, x}}]]]];
redline[y_, z_] = Graphics[{RGBColor[1, 0, 0],
  Line[{polypoints[2 * y * Pi / partialgenes, 1], polypoints[2 * z * Pi / partialgenes, 1]}]};
greenline[y_, z_] = Graphics[{RGBColor[0, 0.5, 0],
  Line[{polypoints[2 * y * Pi / partialgenes, 1], polypoints[2 * z * Pi / partialgenes, 1]}]};

color1[stage_] =
  If[stage == "Up", RGBColor[0, 0, 0],
    If[stage == "Down", RGBColor[0.5, 0.5, 0.5], RGBColor[1, 1, 1]]
  ];
color2[stage_] =
  If[stage == "M/G1", RGBColor[1, 1, 0],
    If[stage == "G1", RGBColor[0, 0.5, 0],
      If[stage == "S", RGBColor[0, 0, 1],
        If[stage == "S/G2", RGBColor[1, 0, 0],
          If[stage == "G2/M", RGBColor[1, 0.5, 0], RGBColor[1, 1, 1]]
        ]]]];

circle1 = Table[
  Graphics[{
    color1[partialgenenames[[a, 7]]],
    Disk[{0, 0}, 1.35,
      {-2 * (a + 0.5) * Pi / partialgenes + Pi / 2., -2 * (a - 0.5) * Pi / partialgenes + Pi / 2.}],
    {a, partialgenes, 1, -1}];
circle2 = Table[
  Graphics[{
    color2[partialgenenames[[a, 6]]],
    Disk[{0, 0}, 1.3,
      {-2 * (a + 0.5) * Pi / partialgenes + Pi / 2., -2 * (a - 0.5) * Pi / partialgenes + Pi / 2.}],
    {a, partialgenes, 1, -1}];
circle3 = Graphics[{RGBColor[1, 1, 1], Disk[{0, 0}, 1.25]}];
circle4 = Table[
  Graphics[{
    RGBColor[0, 0, 0],
    Disk[polypoints[2 * a * Pi / partialgenes, 1], 0.02]}, {a, 1, partialgenes}];
circle5 = Table[
  Graphics[{
    RGBColor[0, 0, 0],
    Text[partialgenenames[[a, 2]], polypoints[2 * a * Pi / partialgenes, 1.15],
      {0, 0}, polypoints[2 * a * Pi / partialgenes, 1.15]}],
  {a, 1, partialgenes}];
```

```
(* Compare Genes × Genes Eigencorrelations Networks Using Logic *)
```

```
(* Display Intersection of First AND Second AND Third Networks *)
```

```
int1 = Intersection[
  Position[Sign[eigenmatrices[[1]]], 1],
  Position[Sign[eigenmatrices[[2]]], 1],
  Position[Sign[eigenmatrices[[3]]], 1]];
int2 = Intersection[
  Position[Sign[eigenmatrices[[1]]], -1],
  Position[Sign[eigenmatrices[[2]]], -1],
  Position[Sign[eigenmatrices[[3]]], -1]];

redtable = Table[redline[int1[[a, 1]], int1[[a, 2]]], {a, 1, Dimensions[int1][[1]]};
greentable = Table[greenline[int2[[a, 1]], int2[[a, 2]]], {a, 1, Dimensions[int2][[1]]};
p = Show[
  {circle1, circle2, circle3, circle4, circle5, redtable, greentable,
  Graphics[{RGBColor[0, 0, 0], Text[StyleForm["(a)", FontSize → 24], {-1.1, 1.1}]}],
  \AspectRatio -> 1,
  PlotRange -> {{-1.25, 1.25}, {-1.25, 1.25}},
  Frame -> False,
  FrameTicks -> False,
  FrameLabel -> {None, None, None, None},
  GridLines -> {None, None},
  DisplayFunction -> Identity];
p = FullGraphics[p];
p[[1, 2]] = p[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {-1.18, 0}, {0, 0}, {0, 1}];
p1 = Show[p,
  AspectRatio -> 1.,
  PlotRange -> All,
  DisplayFunction -> Identity];
```

```
(* Define Eigencorrelations Networks *)
```

```
networkgenes4 = partialgenenames;
networks41 = eigenmatrices[[1]];
networks42 = eigenmatrices[[2]];
networks43 = eigenmatrices[[3]];
```

(* Intersect Alpha Factor and Biosynthesis-Projected Alpha Factor Networks *)

```
genes = Dimensions[networkgenes1][[1]];
genenames = TakeColumns[networkgenes1, 1];

externalgenes = Dimensions[networkgenes4][[1]];
externalgenenames = TakeColumns[networkgenes4, 1];

list = Flatten[Intersection[genenames, externalgenenames]];
{partialgenes} = Dimensions[list]

{30}

counter = Table[{Position[genenames, list[[a]]][[1, 1]], list[[a]]}, {a, 1, partialgenes}];
counter = ReplaceAll[counter, {} -> {Null}];
counter = Sort[counter, OrderedQ[{{#1, #2}} &];
list = Flatten[TakeColumns[counter, {2}]];

counter = Table[Flatten[Position[list, genenames[[a, 1]]]], {a, 1, genes}];
counter = ReplaceAll[counter, {} -> {Null}];
partialmatrix = AppendRows[counter, networkgenes1, networks11, networks12, networks13, networks14];
partialmatrix = Sort[partialmatrix, OrderedQ[{{#1, #2}} &];
partialgenenames = TakeRows[
  TakeColumns[partialmatrix, {2, 8}],
  {1, partialgenes}];
partialmatrix11 = TakeRows[
  TakeColumns[partialmatrix, {9, genes + 8}],
  {1, partialgenes}];
partialmatrix12 = TakeRows[
  TakeColumns[partialmatrix, {genes + 9, 2 * genes + 8}],
  {1, partialgenes}];
partialmatrix13 = TakeRows[
  TakeColumns[partialmatrix, {2 * genes + 9, 3 * genes + 8}],
  {1, partialgenes}];
partialmatrix14 = TakeRows[
  TakeColumns[partialmatrix, {3 * genes + 9, 4 * genes + 8}],
  {1, partialgenes}];

partialmatrix11 = Transpose[partialmatrix11];
partialmatrix12 = Transpose[partialmatrix12];
partialmatrix13 = Transpose[partialmatrix13];
partialmatrix14 = Transpose[partialmatrix14];

partialmatrix = AppendRows[counter, partialmatrix11, partialmatrix12, partialmatrix13, partialmatrix14];
partialmatrix = Sort[partialmatrix, OrderedQ[{{#1, #2}} &];
partialmatrix11 = TakeRows[
  TakeColumns[partialmatrix, {2, partialgenes + 1}],
  {1, partialgenes}];
partialmatrix12 = TakeRows[
  TakeColumns[partialmatrix, {partialgenes + 2, 2 * partialgenes + 1}],
  {1, partialgenes}];
partialmatrix13 = TakeRows[
  TakeColumns[partialmatrix, {2 * partialgenes + 2, 3 * partialgenes + 1}],
  {1, partialgenes}];
partialmatrix14 = TakeRows[
  TakeColumns[partialmatrix, {3 * partialgenes + 2, 4 * partialgenes + 1}],
  {1, partialgenes}];

partialmatrix11 = Transpose[partialmatrix11];
partialmatrix12 = Transpose[partialmatrix12];
partialmatrix13 = Transpose[partialmatrix13];
partialmatrix14 = Transpose[partialmatrix14];
```

```

counter = Table[Flatten[Position[list, externalgenenames[[a, 1]]], {a, 1, externalgenes}];
counter = ReplaceAll[counter, {} -> {Null}];
partialexternalmatrix = AppendRows[counter, networks41, networks42, networks43];
partialexternalmatrix = Sort[partialexternalmatrix, OrderedQ[{{#1, #2}} &];
partialexternalmatrix41 = TakeRows[
  TakeColumns[partialexternalmatrix, {2, externalgenes + 1}],
  {1, partialgenes}];
partialexternalmatrix42 = TakeRows[
  TakeColumns[partialexternalmatrix, {externalgenes + 2, 2 * externalgenes + 1}],
  {1, partialgenes}];
partialexternalmatrix43 = TakeRows[
  TakeColumns[partialexternalmatrix, {2 * externalgenes + 2, 3 * externalgenes + 1}],
  {1, partialgenes}];

partialexternalmatrix41 = Transpose[partialexternalmatrix41];
partialexternalmatrix42 = Transpose[partialexternalmatrix42];
partialexternalmatrix43 = Transpose[partialexternalmatrix43];

partialexternalmatrix =
  AppendRows[counter, partialexternalmatrix41, partialexternalmatrix42, partialexternalmatrix43];
partialexternalmatrix = Sort[partialexternalmatrix, OrderedQ[{{#1, #2}} &];
partialexternalmatrix41 = TakeRows[
  TakeColumns[partialexternalmatrix, {2, partialgenes + 1}],
  {1, partialgenes}];
partialexternalmatrix42 = TakeRows[
  TakeColumns[partialexternalmatrix, {partialgenes + 2, 2 * partialgenes + 1}],
  {1, partialgenes}];
partialexternalmatrix43 = TakeRows[
  TakeColumns[partialexternalmatrix, {2 * partialgenes + 2, 3 * partialgenes + 1}],
  {1, partialgenes}];

partialexternalmatrix41 = Transpose[partialexternalmatrix41];
partialexternalmatrix42 = Transpose[partialexternalmatrix42];
partialexternalmatrix43 = Transpose[partialexternalmatrix43];

```

```
(* Define Display Parameters *)
```

```
Clear[x, y, z];
R[theta_] = {{Cos[theta], Sin[theta]}, {-Sin[theta], Cos[theta]}};
polypoints[theta_, x_] = Flatten[Transpose[Dot[R[theta], Transpose[{{0, x}}]]]];
redline[y_, z_] = Graphics[{RGBColor[1, 0, 0],
  Line[{polypoints[2 * y * Pi / partialgenes, 1], polypoints[2 * z * Pi / partialgenes, 1]}]};
greenline[y_, z_] = Graphics[{RGBColor[0, 0.5, 0],
  Line[{polypoints[2 * y * Pi / partialgenes, 1], polypoints[2 * z * Pi / partialgenes, 1]}]};

color1[stage_] =
  If[stage == "Up", RGBColor[0, 0, 0],
    If[stage == "Down", RGBColor[0.5, 0.5, 0.5], RGBColor[1, 1, 1]]
  ];
color2[stage_] =
  If[stage == "M/G1", RGBColor[1, 1, 0],
    If[stage == "G1", RGBColor[0, 0.5, 0],
      If[stage == "S", RGBColor[0, 0, 1],
        If[stage == "S/G2", RGBColor[1, 0, 0],
          If[stage == "G2/M", RGBColor[1, 0.5, 0], RGBColor[1, 1, 1]]
        ]]]];

circle1 = Table[
  Graphics[{
    color1[partialgenenames[[a, 7]]],
    Disk[{0, 0}, 1.35,
      {-2 * (a + 0.5) * Pi / partialgenes + Pi / 2., -2 * (a - 0.5) * Pi / partialgenes + Pi / 2.}],
    {a, partialgenes, 1, -1}];
circle2 = Table[
  Graphics[{
    color2[partialgenenames[[a, 6]]],
    Disk[{0, 0}, 1.3,
      {-2 * (a + 0.5) * Pi / partialgenes + Pi / 2., -2 * (a - 0.5) * Pi / partialgenes + Pi / 2.}],
    {a, partialgenes, 1, -1}];
circle3 = Graphics[{RGBColor[1, 1, 1], Disk[{0, 0}, 1.25]}];
circle4 = Table[
  Graphics[{
    RGBColor[0, 0, 0],
    Disk[polypoints[2 * a * Pi / partialgenes, 1], 0.02]}, {a, 1, partialgenes}];
circle5 = Table[
  Graphics[{
    RGBColor[0, 0, 0],
    Text[partialgenenames[[a, 2]], polypoints[2 * a * Pi / partialgenes, 1.15],
      {0, 0}, polypoints[2 * a * Pi / partialgenes, 1.15]}],
  {a, 1, partialgenes}];
```

(* Display Intersection of Biosynthesis-Projected First AND Second AND Third AND Alpha Factor Third *)

```
int1 = Intersection[
  Position[Sign[partialmatrix13], 1],
  Position[Sign[partialexternalmatrix41], 1],
  Position[Sign[partialexternalmatrix42], 1],
  Position[Sign[partialexternalmatrix43], 1]];
int2 = Intersection[
  Position[Sign[partialmatrix13], -1],
  Position[Sign[partialexternalmatrix41], -1],
  Position[Sign[partialexternalmatrix42], -1],
  Position[Sign[partialexternalmatrix43], -1]];

redtable = Table[redline[int1[[a, 1]], int1[[a, 2]]], {a, 1, Dimensions[int1][[1]]};
greentable = Table[greenline[int2[[a, 1]], int2[[a, 2]]], {a, 1, Dimensions[int2][[1]]};
p = Show[
  {circle1, circle2, circle3, circle4, circle5, redtable, greentable,
  Graphics[{RGBColor[0, 0, 0], Text[StyleForm["(b)", FontSize -> 24], {-1.1, 1.1}]}]},
  AspectRatio -> 1,
  PlotRange -> {{-1.25, 1.25}, {-1.25, 1.25}},
  Frame -> False,
  FrameTicks -> False,
  FrameLabel -> {None, None, None, None},
  GridLines -> {None, None},
  DisplayFunction -> Identity];
p = FullGraphics[p];
p[[1, 2]] = p[[1, 2]] /.
  Text[labely, {b_, c_}, {1., 0.}] ->
  Text[labely, {-1.18, 0}, {0, 0}, {0, 1}];
p2 = Show[p,
  AspectRatio -> 1.,
  PlotRange -> All,
  DisplayFunction -> Identity];
```

(* Display Selected Genes x Genes Eigencorrelations Networks and Their Intersections *)

```
Show[GraphicsArray[{p1, p2}],  
GraphicsSpacing -> 0];
```

