Preliminary

A = 0 = (0)*2^1 + (0)*2^0 = 00 in binary
C = 1 = (0)*2^1 + (1)*2^0 = 01 in binary
G = 2 = (1)*2^1 + (0)*2^0 = 10 in binary
T = 3 = (1)*2^1 + (1)*2^0 = 11 in binary

The use of the term datavector is meant to imply the 2-layer list (which is also a vector in mathematica) containing a given list of data.
For example {{0,1,2,3,4}} is the datavector containing digits 0 to 4 in order

At some point, want to define a function that will take a DNA set of some # 2^(n-1) < d < 2^n base pairs and fill it with 0's until d = 2^n for some whole number n.
Likely need to use a While[] fn.
Also SparseArray Fn

Look at what you can do with ArrayFlatten fn

Get less singular values from SVD for systems with less entanglement
Perfectly classical systems have no entanglement, therefore they only have 1 singular value
Take operator $\hat{X}$ in some large dimensional Hilbert space. Then it's classical form could be reduced down to 1 singular value, so $\hat{X} \rightarrow x$ by SVD

```
M = {{1, 2, 3}, {2, 1, 4}, {3, 4, 1}};
```

```
{u, w, v} = SingularValueDecomposition[N[M]]
```

```
{{{-0.505785, 0.255232, 0.824038},
  {-0.584374, 0.601302, -0.544925}, {-0.634577, -0.757161, -0.154979}},
 {{7.07467, 0., 0.}, {0., 3.18788, 0.}, {0., 0., 0.886791}},
 {{-0.505785, -0.255232, -0.824038},
  {-0.584374, -0.601302, 0.544925}, {-0.634577, 0.757161, 0.154979}}}
```

Need to operate on N[M] rather than just M, otherwise slots appear in the expression, which increases the computation time too much.

```
u.w.Transpose[v]
```

```
{{1., 2., 3.}, {2., 1., 4.}, {3., 4., 1.}}
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

```
{u1, w1, v1} = SingularValueDecomposition[{{1, 2}, {1, 2}}]
```

$$\left\{\left\{\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right\}, \left\{\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right\}\right\}, \left\{\left\{\sqrt{10}, 0\right\}, \{0, 0\}\right\}, \left\{\left\{\frac{1}{\sqrt{5}}, -\frac{2}{\sqrt{5}}\right\}, \left\{\frac{2}{\sqrt{5}}, \frac{1}{\sqrt{5}}\right\}\right\}$$

```
u1.w1.Transpose[v1]
```

```
{{1, 2}, {1, 2}}
```

Can use FortranForm function to convert mathematica to Fortran, which has a similar syntax to python. Helpful when unsure of how to structure a code.
Can import Fortran codes to mathematica in order to deal with operations which are much difficult to do in code. For instance, can import a code containing integrals,
use splice to do the integrals then have it output back to the fortran file.

```
FortranForm[SingularValueDecomposition[N[A]]]
```

```
SingularValueDecomposition(A)
```

Use SVD code from https://jeremykun.com/2016/05/16/singular-value-decomposition-part-2-theorem-proof-algorithm/
Just need to replace curly brackets here with square brackets for python and the matrix below with the appropriate genetic info vector

```
from numpy.linalg import svd

movieRatings = [
  [2, 5, 3],
  [1, 2, 1],
  [4, 1, 1],
  [3, 5, 2],
  [5, 3, 1],
  [4, 5, 5],
  [2, 4, 2],
  [2, 2, 5],
]

U, singularValues, V = svd(movieRatings)
```

```
{{2, 5, 3},
 {1, 2, 1},
 {4, 1, 1},
 {3, 5, 2},
 {5, 3, 1},
 {4, 5, 5},
 {2, 4, 2},
 {2, 2, 5}}
```

This is tha mathematica form so you can compare SVD in python to the SVD done in mathematica to make sure everything checks out, done explicitly in the next cell

```
Do[Print[SingularValueDecomposition[N[{{2, 5, 3},
  {1, 2, 1},
  {4, 1, 1},
  {3, 5, 2},
  {5, 3, 1},
  {4, 5, 5},
  {2, 4, 2},
  {2, 2, 5}}]][[i]] // MatrixForm], {i, 1, 3}]
(* Gives the components of the SVD of the python example *)
```

$$\begin{pmatrix} -0.394585 & -0.239236 & -0.354459 & -0.380622 & -0.298368 & -0.494648 & -0.307032 & -0.297633 \\ -0.158302 & -0.0305491 & -0.152998 & -0.453348 & 0.311229 & 0.23892 & -0.373133 & 0.672235 \\ -0.221552 & 0.520861 & 0.393349 & -0.149748 & -0.65964 & 0.00488292 & -0.00783684 & 0.259346 \\ -0.396926 & 0.0864901 & -0.410529 & 0.743874 & -0.106295 & 0.0137257 & -0.179593 & 0.263335 \\ -0.346303 & 0.641288 & 0.0738286 & -0.0449415 & 0.580007 & -0.258062 & 0.00211823 & -0.241547 \\ -0.533474 & -0.191689 & 0.199493 & -0.039426 & 0.00424495 & 0.687157 & -0.0695756 & -0.40033 \\ -0.316605 & -0.0610983 & -0.305995 & -0.196118 & -0.0133427 & 0.0144698 & 0.851859 & 0.194635 \\ -0.328402 & -0.459704 & 0.623548 & 0.178304 & 0.176312 & -0.398795 & 0.060659 & 0.257716 \end{pmatrix}$$

$$\begin{pmatrix} 15.0963 & 0. & 0. \\ 0. & 4.30057 & 0. \\ 0. & 0. & 3.40702 \\ 0. & 0. & 0. \\ 0. & 0. & 0. \\ 0. & 0. & 0. \\ 0. & 0. & 0. \\ 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} -0.541848 & 0.751523 & 0.376316 \\ -0.67071 & -0.116809 & -0.732464 \\ -0.506506 & -0.649283 & 0.567347 \end{pmatrix}$$

Note that the results are the same as those obtained in python except the rows and columns are switched in the python one (which has the same bracket setup, may be able to fix this by some arrangement of brackets)

Repeat ~n/2 times Might be best to do some orderings in mathematica.

Need to learn how to access DNA data from an online database using python or Fortran and how to incorporate that into code

```
FortranForm[Table[Flatten[H1Avec][[(71 * (k1 - 1)) + k2]], {k1, 1, 11}, {k2, 1, 71}]]
(*gives matrix repesentation of H1A in ForTran*)
```

```
      List(List(3,0,2,2,1,3,2,1,2,3,3,2,2,2,2,1,1,3,3,3,3,3,3,3,1,2,1,0,3,1,1,3,2,1,3,3,1
    - List(3,3,0,2,3,1,0,1,1,0,3,2,3,1,3,2,0,0,0,1,0,2,3,2,1,1,3,1,1,1,2,1,1,1,1,1,2,1,1,
    - List(1,0,0,2,0,0,2,2,1,0,0,0,2,0,0,0,1,1,3,2,1,3,0,0,2,2,1,3,2,1,0,2,1,0,2,1,1,3,1,
    - List(0,2,1,3,2,0,3,1,2,3,2,1,0,2,2,1,3,2,1,3,3,1,1,3,1,1,3,1,3,0,0,2,2,0,2,1,2,3,2,
    - List(1,3,2,2,1,2,2,1,1,2,1,0,2,2,1,3,0,1,2,0,1,2,3,2,2,0,2,0,0,2,0,0,1,0,0,1,0,2,1,
    - List(1,0,0,2,2,2,0,0,1,2,3,3,2,2,3,2,1,0,2,0,1,0,0,0,2,2,2,3,0,1,1,2,2,0,2,1,1,3,1,
    - List(1,1,2,3,2,2,0,0,0,1,1,0,0,2,1,1,1,2,2,1,2,1,1,3,1,0,0,0,2,2,3,2,2,1,3,0,1,0,0,
    - List(0,0,0,0,0,2,2,1,1,0,1,2,2,2,2,2,1,3,0,2,1,0,0,0,0,0,2,0,2,1,2,3,1,0,0,2,0,1,3,
    - List(2,0,0,0,3,1,1,3,1,1,0,0,2,0,0,3,1,1,0,0,0,0,0,0,0,1,1,1,0,0,0,0,1,3,2,3,0,0,0,
    - List(1,3,0,0,2,2,1,3,2,3,0,0,0,0,1,1,1,0,0,2,2,1,2,2,1,1,0,0,2,2,1,3,0,2,2,2,3,2,0,
    - List(2,1,0,1,1,1,0,0,2,0,0,0,0,0,2,3,0,0,0,3,3,1,0,2,3,3,0,2,0,0,2,3,3,3,1,3,3,1,3,
```

```
FortranForm[Nest[SingularValueDecomposition, N[M], 3]]
```

```
SingularValueDecomposition(SingularValueDecomposition(SingularValueDecomposition(M)))
```

Histone H1A

H1A gene data vector representation

Define a 1x781 vector H1Avec which contains all the base pair info of the H1A gene

```
H1Avec =
  {{3, 0, 2, 2, 1, 3, 2, 1, 2, 3, 3, 2, 2, 2, 2, 1, 1, 3, 3, 3, 3, 3, 3, 3, 1, 2, 1, 0, 3, 1, 1, 3, 2,
    1, 3, 3, 1, 2, 3, 1, 0, 2, 2, 3, 3, 3, 0, 3, 0, 1, 1, 0, 1, 3, 3, 3, 0, 3, 3, 3, 2, 2, 3, 2, 3,
    2, 1, 3, 2, 3, 2, 3, 3, 0, 2, 3, 1, 0, 1, 1, 0, 3, 2, 3, 1, 3, 2, 0, 0, 0, 1, 0, 2, 3, 2, 1, 1,
    3, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 1, 1, 2, 1, 3, 3, 1, 3, 2, 1, 3, 2, 1, 3, 1, 1, 3, 2, 0,
    2, 0, 0, 0, 1, 1, 3, 3, 3, 0, 2, 1, 3, 2, 2, 1, 0, 0, 2, 0, 0, 2, 2, 1, 0, 0, 0, 2, 0, 0,
    0, 1, 1, 3, 2, 1, 3, 0, 0, 2, 2, 1, 3, 2, 1, 0, 2, 1, 0, 2, 1, 1, 3, 1, 1, 0, 0, 2, 0, 0,
    0, 0, 0, 0, 1, 1, 1, 2, 1, 3, 2, 2, 1, 1, 1, 3, 3, 1, 1, 2, 3, 2, 3, 1, 0, 2, 0, 2, 1, 3,
    2, 0, 3, 1, 2, 3, 2, 1, 0, 2, 2, 1, 3, 2, 1, 3, 3, 1, 1, 3, 1, 1, 3, 1, 3, 0, 0, 2, 2, 0,
    2, 1, 2, 3, 2, 2, 3, 2, 2, 3, 2, 3, 2, 3, 1, 2, 3, 3, 2, 2, 1, 0, 2, 1, 3, 1, 3, 3, 0, 0,
    0, 0, 0, 2, 2, 1, 2, 1, 3, 2, 2, 1, 2, 2, 1, 1, 2, 1, 0, 2, 2, 1, 3, 0, 1, 2, 0, 1, 2, 3,
    2, 2, 0, 2, 0, 0, 2, 0, 0, 1, 0, 0, 1, 0, 2, 1, 1, 2, 1, 0, 3, 3, 0, 0, 2, 1, 3, 2, 2, 2,
    1, 0, 3, 3, 0, 0, 2, 0, 2, 1, 1, 3, 2, 2, 3, 0, 0, 2, 1, 0, 0, 2, 2, 2, 0, 0, 1, 2, 3, 3,
    2, 2, 3, 2, 1, 0, 2, 0, 1, 0, 0, 0, 2, 2, 2, 3, 0, 1, 1, 2, 2, 0, 2, 1, 1, 3, 1, 2, 2, 2,
    3, 3, 1, 1, 3, 3, 1, 0, 0, 2, 1, 3, 1, 0, 0, 1, 0, 0, 2, 0, 0, 2, 2, 1, 2, 3, 1, 1, 3, 1,
    1, 2, 3, 2, 2, 0, 0, 0, 1, 1, 0, 0, 2, 1, 1, 1, 2, 2, 1, 2, 1, 1, 3, 1, 0, 0, 0, 2, 2, 3,
    2, 2, 1, 3, 0, 1, 0, 0, 0, 0, 0, 1, 3, 0, 0, 2, 2, 1, 0, 0, 1, 2, 2, 2, 3, 2, 1, 0, 3, 1,
    3, 0, 0, 0, 0, 0, 2, 1, 3, 1, 0, 0, 0, 0, 0, 2, 2, 1, 1, 0, 1, 2, 2, 2, 2, 2, 1, 3, 0, 2,
    1, 0, 0, 0, 0, 0, 2, 0, 2, 1, 2, 3, 1, 0, 0, 2, 0, 1, 3, 1, 1, 2, 0, 0, 0, 0, 0, 2, 2, 1,
    3, 0, 0, 0, 0, 0, 2, 1, 1, 3, 2, 1, 2, 2, 1, 0, 0, 1, 0, 0, 2, 2, 0, 0, 0, 3, 1, 1, 3, 1,
    1, 0, 0, 2, 0, 0, 3, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 1, 3, 2, 3, 0, 0, 0,
    2, 1, 1, 1, 0, 0, 2, 0, 0, 0, 2, 3, 0, 2, 1, 3, 0, 0, 0, 0, 2, 1, 1, 1, 3, 2, 1, 3, 0,
    0, 0, 2, 1, 3, 0, 0, 2, 2, 1, 3, 2, 3, 0, 0, 0, 0, 1, 1, 1, 0, 0, 2, 2, 1, 2, 2, 1, 1,
    0, 0, 2, 2, 1, 3, 0, 2, 2, 2, 3, 2, 0, 1, 2, 0, 0, 2, 1, 1, 0, 0, 0, 2, 0, 1, 3, 2, 1,
    1, 0, 0, 0, 1, 1, 1, 0, 0, 2, 0, 0, 0, 2, 1, 2, 2, 1, 0, 1, 1, 1, 0, 0, 2, 0, 0, 0, 0,
    0, 2, 3, 0, 0, 0, 3, 3, 1, 0, 2, 3, 3, 0, 2, 0, 0, 2, 3, 3, 3, 1, 3, 3, 1, 3, 0, 2, 3,
    0, 0, 1, 1, 1, 0, 0, 1, 2, 2, 1, 3, 1, 3, 3, 3, 3, 0, 0, 2, 0, 2, 1, 1, 0, 1, 1, 3, 0}};
```

```
H1Avec // MatrixForm
```

```
⟨3  0  2  2  1  3  2  1  2  3  3  2  2  2  2  1  1  3  3  3  3  3  3  3  1  2  1  0  3  1  1  3  2  1  3  3  1
```

```
(71 * 10) + 2
```

712

```
H1Avec[[1, (70 * 11) + 2]]
```

2

```
Flatten[H1Avec][[(70 * 11) + 2]]
```

2

So Flatten[H1Avec][[k]] = H1Avec[[1, k]]

SVD of H1Avec (Not important except for examples of methods used)

Note: H1Avec vector has 781 digits, didn't put in fillers to get to 2^10 =1024 yet, but seem to be getting good results.

```
PrimeQ[781]
```

```
False
```

```
(*SingularValueDecomposition[N[H1Avec]]*)
```

Very large but note alot of repeating numbers. Some are
0.
-0.0624323
-0.0416215
-0.0208108
-0.001222910
-0.00244583
-0.00366874
-0.000815276

This function only breaks H1Avec down to 3 matrices (can check this fact more quickly using ctrl F on double brackets {{) , not decomposing every number.

So the high degree of repetition here likely implies a high degree of correlation. (Check with Dr. Muccciolo)

```
(*SingularValueDecomposition[N[H1Avec]][[1]]*)
```

```
(*SingularValueDecomposition[N[H1Avec]][[2]] *)
(* double bracket [[]] calls the matrix*)
```

So the first matrix of the SVD only has one term, the second matrix only has 1 non-zero term. Is the 2nd matrix empty just because of the nature of this SVD or because there's a high degree of correlation
So almost all of the relevant info is contained in the the 3rd matrix, so this is the only one that needs to be decomposed further.

Also: want to see how many unique numbers there are in this decomposition - use Gather function on 3rd matrix for this purpose

```
(*SingularValueDecomposition[N[H1Avec]][[1]][[1]]*)
```

Use SingularValueDecomposition[N[H1Avec]][[3]][[k]] to call the 3rd matrix as a list

```
(*Length[SingularValueDecomposition[N[H1Avec]][[3]]]*)
```

Can check that
SingularValueDecomposition[N[H1Avec]][[3]][[781]] exists while

SingularValueDecomposition[N[H1Avec]][[3]][[782]] doesn't

So want to join SingularValueDecomposition[N[H1Avec]][[3]][[k]], k∈{1, 781} into one list and then use gather function

```
(* Gather[SingularValueDecomposition[N[H1Avec]][[3]][[782]]] *)
(*SingularValueDecomposition[N[H1Avec]][[2]]*)
(*SingularValueDecomposition[N[H1Avec]][[3]][[1]]*)
(*SVD3List=Table[Union[SingularValueDecomposition[N[H1Avec]][[3]][[k]]],{k,1,781}];*)
```

Union removes repeated elements from each SingularValueDecomposition[N[H1Avec]][[3]][[k]]

Use Flatten fn to remove extra list brackets

```
(*Union[Flatten[SVD3List]]*)
(*Length[SVD3List]*)
(*H1AfirstSVDuniqueterms=Union[Flatten[SingularValueDecomposition[N[H1Avec]]]]*)
(*Length[H1AfirstSVDuniqueterms]*)
```

Only 19 unique numbers in the SVD of 781 base pair long Histone H1Avec gene!

And this is only the first SVD, need to / can do up to 5, since 1024 = 2^10 = 4^5 > 781 > 2^9 = 4^(4.5) = 512, (4-bit)

```
(*Length[Union[Flatten[SingularValueDecomposition[N[H1Avec]]]]]*)
(* Part we need for 1st SVD of genes below *)
```

Further want to do rest of SVDs

Also change ordering of the base pairs and see what that does to the SVD and the number of unique terms and the values of the unique terms

Matrix Representation

```
H1Avec =
  {{3, 0, 2, 2, 1, 3, 2, 1, 2, 3, 3, 2, 2, 2, 2, 1, 1, 3, 3, 3, 3, 3, 3, 3, 1, 2, 1, 0, 3, 1, 1, 3, 2,
    1, 3, 3, 1, 2, 3, 1, 0, 2, 2, 3, 3, 3, 0, 3, 0, 1, 1, 0, 1, 3, 3, 3, 0, 3, 3, 3, 2, 2, 3, 2, 3,
    2, 1, 3, 2, 3, 2, 3, 3, 0, 2, 3, 1, 0, 1, 1, 0, 3, 2, 3, 1, 3, 2, 0, 0, 0, 1, 0, 2, 3, 2, 1, 1,
    3, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 1, 1, 2, 1, 3, 3, 1, 3, 2, 1, 3, 2, 1, 3, 1, 1, 3, 2, 0,
    2, 0, 0, 0, 1, 1, 3, 3, 3, 0, 2, 1, 3, 2, 2, 1, 0, 0, 2, 0, 0, 2, 2, 1, 0, 0, 0, 2, 0, 0,
    0, 1, 1, 3, 2, 1, 3, 0, 0, 2, 2, 1, 3, 2, 1, 0, 2, 1, 0, 2, 1, 1, 3, 1, 1, 0, 0, 2, 0, 0,
    0, 0, 0, 0, 1, 1, 1, 2, 1, 3, 2, 2, 1, 1, 1, 3, 3, 1, 1, 2, 3, 2, 3, 1, 0, 2, 0, 2, 1, 3,
    2, 0, 3, 1, 2, 3, 2, 1, 0, 2, 2, 1, 3, 2, 1, 3, 3, 1, 1, 3, 1, 1, 3, 1, 3, 0, 0, 2, 2, 0,
    2, 1, 2, 3, 2, 2, 3, 2, 2, 3, 2, 3, 2, 3, 1, 2, 3, 3, 2, 2, 1, 0, 2, 1, 3, 1, 3, 3, 0, 0,
    0, 0, 0, 2, 2, 1, 2, 1, 3, 2, 2, 1, 2, 2, 1, 1, 2, 1, 0, 2, 2, 1, 3, 0, 1, 2, 0, 1, 2, 3,
    2, 2, 0, 2, 0, 0, 2, 0, 0, 1, 0, 0, 1, 0, 2, 1, 1, 2, 1, 0, 3, 3, 0, 0, 2, 1, 3, 2, 2, 2,
    1, 0, 3, 3, 0, 0, 2, 0, 2, 1, 1, 3, 2, 2, 3, 0, 0, 2, 1, 0, 0, 2, 2, 2, 0, 0, 1, 2, 3, 3,
    2, 2, 3, 2, 1, 0, 2, 0, 1, 0, 0, 0, 2, 2, 2, 3, 0, 1, 1, 2, 2, 0, 2, 1, 1, 3, 1, 2, 2, 2,
    3, 3, 1, 1, 3, 3, 1, 0, 0, 2, 1, 3, 1, 0, 0, 1, 0, 0, 2, 0, 0, 2, 2, 1, 2, 3, 1, 1, 3, 1,
    1, 2, 3, 2, 2, 0, 0, 0, 1, 1, 0, 0, 2, 1, 1, 1, 2, 2, 1, 2, 1, 1, 3, 1, 0, 0, 0, 2, 2, 3,
    2, 2, 1, 3, 0, 1, 0, 0, 0, 0, 0, 1, 3, 0, 0, 2, 2, 1, 0, 0, 1, 2, 2, 2, 3, 2, 1, 0, 3, 1,
    3, 0, 0, 0, 0, 0, 2, 1, 3, 1, 0, 0, 0, 0, 0, 2, 2, 1, 1, 0, 1, 2, 2, 2, 2, 2, 1, 3, 0, 2,
    1, 0, 0, 0, 0, 0, 2, 0, 2, 1, 2, 3, 1, 0, 0, 2, 0, 1, 3, 1, 1, 2, 0, 0, 0, 0, 0, 2, 2, 1,
    3, 0, 0, 0, 0, 0, 2, 1, 1, 3, 2, 1, 2, 2, 1, 0, 0, 1, 0, 0, 2, 2, 0, 0, 0, 3, 1, 1, 3, 1,
    1, 0, 0, 2, 0, 0, 3, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 1, 3, 2, 3, 0, 0, 0,
    2, 1, 1, 1, 0, 0, 2, 0, 0, 0, 2, 3, 0, 2, 1, 3, 0, 0, 0, 0, 2, 1, 1, 1, 3, 2, 1, 3, 0,
    0, 0, 2, 1, 3, 0, 0, 2, 2, 1, 3, 2, 3, 0, 0, 0, 0, 1, 1, 1, 0, 0, 2, 2, 1, 2, 2, 1, 1,
    0, 0, 2, 2, 1, 3, 0, 2, 2, 2, 3, 2, 0, 1, 2, 0, 0, 2, 1, 1, 0, 0, 0, 2, 0, 1, 3, 2, 1,
    1, 0, 0, 0, 1, 1, 1, 0, 0, 2, 0, 0, 0, 2, 1, 2, 2, 1, 0, 1, 1, 1, 0, 0, 2, 0, 0, 0, 0,
    0, 2, 3, 0, 0, 0, 3, 3, 1, 0, 2, 3, 3, 0, 2, 0, 0, 2, 3, 3, 3, 1, 3, 3, 1, 3, 0, 2, 3,
    0, 0, 1, 1, 1, 0, 0, 1, 2, 2, 1, 3, 1, 3, 3, 3, 3, 0, 0, 2, 0, 2, 1, 1, 0, 1, 1, 3, 0}};
```

Now we construct a matrix from this vector data, since we can perform a meaningful SVD on this matrix, but an SVD of a vector doesn't tell us much

781 = 11 * 71 when prime decomposed, so we make a 11x71 matrix

May also use a filler to get H1Avec to be a power of 2 so that after each SVD it breaks down to a lesser power of 2 until we get to 1x1 matrices

Note that Flatten[H1Avec][[k]] = H1Avec[[1, k]] is just the k-th term of the vector, do this to remove the lists

```
Table[Flatten[H1Avec][[(71 * (k1 - 1)) + k2]], {k1, 1, 11}, {k2, 1, 71}] // MatrixForm
```

$$\begin{pmatrix}
3 & 0 & 2 & 2 & 1 & 3 & 2 & 1 & 2 & 3 & 3 & 2 & 2 & 2 & 2 & 1 & 1 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 1 & 2 & 1 & 0 & 3 & 1 & 1 & 3 & 2 & 1 & 3 & 3 & 1 \\
3 & 3 & 0 & 2 & 3 & 1 & 0 & 1 & 1 & 0 & 3 & 2 & 3 & 1 & 3 & 2 & 0 & 0 & 0 & 1 & 0 & 2 & 3 & 2 & 1 & 1 & 3 & 1 & 1 & 1 & 2 & 1 & 1 & 1 & 1 & 1 & 2 \\
1 & 0 & 0 & 2 & 0 & 0 & 2 & 2 & 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 1 & 1 & 3 & 2 & 1 & 3 & 0 & 0 & 2 & 2 & 1 & 3 & 2 & 1 & 0 & 2 & 1 & 0 & 2 & 1 & 1 \\
0 & 2 & 1 & 3 & 2 & 0 & 3 & 1 & 2 & 3 & 2 & 1 & 0 & 2 & 2 & 1 & 3 & 2 & 1 & 3 & 3 & 1 & 1 & 3 & 1 & 1 & 3 & 1 & 3 & 0 & 0 & 2 & 2 & 0 & 2 & 1 & 2 \\
1 & 3 & 2 & 2 & 1 & 2 & 2 & 1 & 1 & 2 & 1 & 0 & 2 & 2 & 1 & 3 & 0 & 1 & 2 & 0 & 1 & 2 & 3 & 2 & 2 & 0 & 2 & 0 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 2 & 2 & 2 & 0 & 0 & 1 & 2 & 3 & 3 & 2 & 2 & 3 & 2 & 1 & 0 & 2 & 0 & 1 & 0 & 0 & 0 & 2 & 2 & 2 & 3 & 0 & 1 & 1 & 2 & 2 & 0 & 2 & 1 & 1 \\
1 & 1 & 2 & 3 & 2 & 2 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 2 & 1 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 3 & 1 & 0 & 0 & 0 & 2 & 2 & 3 & 2 & 2 & 1 & 3 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 2 & 2 & 1 & 1 & 0 & 1 & 2 & 2 & 2 & 2 & 2 & 1 & 3 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 2 & 1 & 2 & 3 & 1 & 0 & 0 & 2 & 0 \\
2 & 0 & 0 & 0 & 3 & 1 & 1 & 3 & 1 & 1 & 0 & 0 & 2 & 0 & 0 & 3 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 3 & 2 & 3 & 0 \\
1 & 3 & 0 & 0 & 2 & 2 & 1 & 3 & 2 & 3 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 2 & 2 & 1 & 2 & 2 & 1 & 1 & 0 & 0 & 2 & 2 & 1 & 3 & 0 & 2 & 2 & 2 & 3 \\
2 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 2 & 3 & 0 & 0 & 0 & 3 & 3 & 1 & 0 & 2 & 3 & 3 & 0 & 2 & 0 & 0 & 2 & 3 & 3 & 3 & 1 & 3 & 3
\end{pmatrix}$$

```
H1A = Table[Flatten[H1Avec][[(71 * (k1 - 1)) + k2]], {k1, 1, 11}, {k2, 1, 71}];
(* Dont include //Matrixform in definition, messes with the following stuff *)
```

```
H1A // MatrixForm
```

$$\begin{pmatrix}
3 & 0 & 2 & 2 & 1 & 3 & 2 & 1 & 2 & 3 & 3 & 2 & 2 & 2 & 2 & 1 & 1 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 1 & 2 & 1 & 0 & 3 & 1 & 1 & 3 & 2 & 1 & 3 & 3 & 1 \\
3 & 3 & 0 & 2 & 3 & 1 & 0 & 1 & 1 & 0 & 3 & 2 & 3 & 1 & 3 & 2 & 0 & 0 & 0 & 1 & 0 & 2 & 3 & 2 & 1 & 1 & 3 & 1 & 1 & 1 & 2 & 1 & 1 & 1 & 1 & 1 & 2 \\
1 & 0 & 0 & 2 & 0 & 0 & 2 & 2 & 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 1 & 1 & 3 & 2 & 1 & 3 & 0 & 0 & 2 & 2 & 1 & 3 & 2 & 1 & 0 & 2 & 1 & 0 & 2 & 1 & 1 \\
0 & 2 & 1 & 3 & 2 & 0 & 3 & 1 & 2 & 3 & 2 & 1 & 0 & 2 & 2 & 1 & 3 & 2 & 1 & 3 & 3 & 1 & 1 & 3 & 1 & 1 & 3 & 1 & 3 & 0 & 0 & 2 & 2 & 0 & 2 & 1 & 2 \\
1 & 3 & 2 & 2 & 1 & 2 & 2 & 1 & 1 & 2 & 1 & 0 & 2 & 2 & 1 & 3 & 0 & 1 & 2 & 0 & 1 & 2 & 3 & 2 & 2 & 0 & 2 & 0 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 2 & 2 & 2 & 0 & 0 & 1 & 2 & 3 & 3 & 2 & 2 & 3 & 2 & 1 & 0 & 2 & 0 & 1 & 0 & 0 & 0 & 2 & 2 & 2 & 3 & 0 & 1 & 1 & 2 & 2 & 0 & 2 & 1 & 1 \\
1 & 1 & 2 & 3 & 2 & 2 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 2 & 1 & 1 & 1 & 2 & 2 & 1 & 2 & 1 & 1 & 3 & 1 & 0 & 0 & 0 & 2 & 2 & 3 & 2 & 2 & 1 & 3 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 2 & 2 & 1 & 1 & 0 & 1 & 2 & 2 & 2 & 2 & 2 & 1 & 3 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 2 & 1 & 2 & 3 & 1 & 0 & 0 & 2 & 0 \\
2 & 0 & 0 & 0 & 3 & 1 & 1 & 3 & 1 & 1 & 0 & 0 & 2 & 0 & 0 & 3 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 3 & 2 & 3 & 0 \\
1 & 3 & 0 & 0 & 2 & 2 & 1 & 3 & 2 & 3 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 2 & 2 & 1 & 2 & 2 & 1 & 1 & 0 & 0 & 2 & 2 & 1 & 3 & 0 & 2 & 2 & 2 & 3 \\
2 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 2 & 3 & 0 & 0 & 0 & 3 & 3 & 1 & 0 & 2 & 3 & 3 & 0 & 2 & 0 & 0 & 2 & 3 & 3 & 3 & 1 & 3 & 3
\end{pmatrix}$$

```
H1A[[1]] (*Gives the first row of the matrix*)
Length[H1A[[1]]]
```

```
{3, 0, 2, 2, 1, 3, 2, 1, 2, 3, 3, 2, 2, 2, 2, 1, 1, 3, 3, 3, 3,
 3, 3, 3, 1, 2, 1, 0, 3, 1, 1, 3, 2, 1, 3, 3, 1, 2, 3, 1, 0, 2, 2, 3, 3, 3,
 0, 3, 0, 1, 1, 0, 1, 3, 3, 3, 0, 3, 3, 3, 2, 2, 3, 2, 3, 2, 1, 3, 2, 3, 2}
```

```
71
```

So H1A is 11x71, as desired

SVD of H1A

```
SingularValueDecomposition[N[H1A]] (* 1st SVD, need more,
for a filled 2^n data st, need about n/2 or (n-1)/2 SVDs, CHECK *)
```

```
{ ··· 1 ··· }
large output | show less | show more | show all | set size limit...
```

For a filled 2^n data st, need about n/2 or (n-1)/2 SVDs, CHECK
Sould be ~ n/2 since 2^n = 4^(n/2) and we have 4 bases (00,01,10,11)
Need to SVD further, but can't just do $H1ASVD_2$ = SingularValueDecomposition[$H1ASVD_1$] for a non 2^n data set, I don't think, need to round it out first

For some 2^n dataset could define a function
SVD[i_,M_]:=Nest[SingularValueDecomposition,N[M], i]
Which gives the i-th SVD of a matrix M

Then to get a list of SVDs from the 1st to the $\frac{(n-1)}{2}$ SVD, do Table[SVD[i_,M_],{i,1,Ceiling[$\frac{(n-1)}{2}$]}]
Note: Either $\frac{(n-1)}{2}$ or $\frac{n}{2}$ is a whole number, depends on n. For 2^n = 1024 = 4^(n/2), $\frac{n}{2}$ = 5, do SVD 5 times, Use Ceiling[$\frac{(n-1)}{2}$], which will use whichever of these is the whole number

Need SVD[i_,M_] to perform i-th SVD on middle matrix of the (i-1)-th SVD, since SVD technically gives 3 matrices . Use Part to specify this condition within the nest

```
(* A={{1,2,3},{4,5,6},{7,8,9}}
  SingularValueDecomposition[N[A]]
  SingularValueDecomposition[N[A]][[2]]
  SingularValueDecomposition[SingularValueDecomposition[N[A]][[2]]]*)

SingularValueDecomposition[
  SingularValueDecomposition[SingularValueDecomposition[N[H1A]][[2]]][[2]]];
```

```
SVD[i_, M_] := Nest[SingularValueDecomposition[Part[#, 2]] &,
   SingularValueDecomposition[N[M]], i - 1];
 (* Defines i-th SVD of matrix M for i>0 *)
```

This does The SVD directly on matrix M iff i = 1, otherwise, for i > 1, it does the SVD on the middle matrix of the previous SVD.
If we want to do the SVD on another part of the matrix, modify the part function

```
(* Λ[i_,M_]:=SVD[i,M][[2]]; *)
(* Defines the middle matrix of the i-th SVD of matrix M *)

SVD[2, H1A];(* Output should show that it's defined as desired *)

(* n =10 here if we were to fill H1Avec up to 1024 = 2^10 places *)

(* Table[SVD[i,M_],{i,1,Ceiling[(n-1)/2]}] *)

SVD[1, H1A]; (*NOTE: Middle matrix mostly 0,
as desired! But others don't have many 0's or repeating numbers*)

Flatten[SVD[1, H1A]]
Union[Flatten[SVD[1, H1A]]];
Length[Union[Flatten[SVD[1, H1A]]]]
Length[Flatten[SVD[1, H1A]]]
Count[Flatten[SVD[1, H1A]], 0.] (* Important to make it find 0., not 0,
as we add periods to every term in defining SVD[i_,M_] in terms of N[M] *)
```

```
{-0.448207, 0.197445, 0.0845012, -0.159649, 0.285422, 0.11021, 0.436264,
 -0.393195, -0.417234, ··· 5925 ···, -0.00575726, -0.0177282, -0.0564673,
 -0.0269022, -0.0536008, -0.0443342, -0.00100048, 0.040746, 0.914226}
```

large output | show less | show more | show all | set size limit...

5174

5943

770

```
5174/5943 // N
```

```
0.870604
```

So 87% of terms of entire SVD are unique - although we mainly care about middle matrix
Now look at this middle matrix SVD[1,H1A][[2]] :

```
SVD[1, H1A][[2]] // MatrixForm;
SVD[2, H1A][[2]] // MatrixForm;
SVD[3, H1A][[2]] // MatrixForm;
SVD[52, H1A][[2]] // MatrixForm; (* All equal *)
```

So can't make this any simler, want to SVD the other matrices SVD[1,H1A][[1]] and SVD[1,H1A][[3]]
Want program to take the first SVD, so M -> U.Λ.V , Λ = eigenvalue matrix
then decompose each of those, Λ -> $\mathcal{I}$.Λ.$\mathcal{I}$ ($\mathcal{I}$= identity), U -> U1.Λu1.U2, V -> V1.Λv1.V2,
U1.Λu1.U2 U1.Λu1.U2 U1.Λu1.U2 U1.Λu1.U2
U1.Λu1.U2 U1.Λu1.U2 U1.Λu1.U2 U1.Λu1.U2 U1.Λu1.U2 U1.Λu1.U2
U1.Λu1.U2 U1.Λu1.U2
Do that Ceiling$\left[\frac{(n-1)}{2}\right]$ times (5 max in this case, since n=10 with filler)
Then will have set of 3^5 such matrices
Exclude the identities then use Union fn on this set to get

Also need to factor U,V so that they're some prime x some prime in size

```
SVD[1, H1A][[1]] // MatrixForm
SVD[1, H1A][[1]][[All, 1]](*gives 1st column*)
Length[SVD[1, H1A][[1]][[All, 1]]]
```

$$\begin{pmatrix}
-0.448207 & 0.197445 & 0.0845012 & -0.159649 & 0.285422 & 0.11021 & 0.436264 & -0.393195 \\
-0.33998 & -0.523011 & -0.0188313 & 0.295415 & -0.243319 & 0.0549875 & 0.259149 & -0.0476981 \\
-0.25508 & 0.0623049 & -0.0802013 & -0.466887 & 0.424995 & -0.468745 & -0.193443 & 0.23209 \\
-0.378105 & 0.199961 & 0.267623 & 0.251939 & -0.0072451 & 0.146136 & -0.682976 & -0.0581778 \\
-0.293373 & -0.133443 & 0.271613 & -0.373473 & -0.369092 & 0.180147 & 0.0802573 & 0.650827 \\
-0.305633 & -0.300142 & 0.249406 & 0.397794 & 0.146324 & -0.528082 & -0.023304 & 0.0148782 \\
-0.260005 & 0.432093 & 0.25059 & -0.0387371 & -0.172062 & -0.0811676 & 0.350381 & -0.0785266 \\
-0.21833 & -0.138567 & -0.0115148 & 0.0201607 & 0.451582 & 0.651295 & -0.09364 & 0.0569358 \\
-0.196224 & -0.44409 & -0.420045 & -0.371338 & -0.0360293 & -0.0192227 & -0.11789 & -0.270203 \\
-0.241565 & 0.163672 & -0.172242 & -0.184666 & -0.534693 & -0.0316021 & -0.278214 & -0.368632 \\
-0.289631 & 0.325279 & -0.713751 & 0.366283 & -0.017045 & -0.0169776 & 0.0939055 & 0.379167
\end{pmatrix}$$

```
{-0.448207, -0.33998, -0.25508, -0.378105, -0.293373,
 -0.305633, -0.260005, -0.21833, -0.196224, -0.241565, -0.289631}
```

```
11
```

So U is Sqaure!

```
SVD[1, H1A][[3]] // MatrixForm;
SVD[1, H1A][[3]][[All, 3]](*gives 1st column*)
Length[SVD[1, H1A][[3]][[All, 3]]]
```

```
{-0.146246, 0.0295717, 0.139611, 0.174757, -0.0455785, 0.0260724, 0.0696985,
 -0.126682, -0.101554, 0.133126, 0.1877, 0.10597, 0.0123513, 0.207914, 0.0278466,
 -0.161945, 0.0663036, 0.0967882, 0.171917, -0.130185, -0.0470275, 0.0142044,
 0.11176, 0.0510661, -0.0799354, -0.186396, 0.127226, -0.102858, 0.0952849,
 0.0870343, -0.0541802, -0.102092, -0.0656082, -0.323296, -0.00360326, -0.266376,
 -0.156114, 0.0880571, -0.0909725, 0.0234217, -0.0355825, -0.109928, 0.100499,
 0.283542, 0.0858302, -0.0431706, 0.0991741, 0.261068, 0.0922434, -0.0541357,
 -0.134638, 0.0373976, -0.0316961, -0.0690701, -0.108659, -0.0616297, -0.0481776,
 -0.114236, -0.110913, 0.0649539, 0.0797986, -0.00225, -0.0051519, -0.179854,
 -0.0527495, -0.0389097, -0.0312642, 0.176117, 0.000878084, -0.0781328, 0.0792417}
```

```
71
```

So V is square too! Lucky in this case, but for a data set for which the # of base pairs can be factored into many more primes would have to further factor U,V

```
SingularValueDecomposition[SVD[1, H1A][[1]]];
```

```
SVD[i_, M_] := Nest[SingularValueDecomposition[Part[#, 2]] &,
   SingularValueDecomposition[N[M]], i - 1];
 (* Defines i-th SVD of matrix M for i>0 *)
```

```
Sum[2^(k), {k, 0, N}]
```

$$-1 + 2^{1+N}$$

```
NumDistinctMatrices[i_] := (-1 + 2^(i-1)) + 3 * (2^(i-1));
(* Number of distinct matrices after i-
 th SVD on all components MINIMUM before factorization of U, V*)
```

```
NumDistinctMatrices[5] (* So will have 63 non identity matrices after i SVDs MINIMUM *)
```

```
63
```

```
SVD1[i_, M_, a_] := Nest[SingularValueDecomposition[Part[#, a]] &,
   SingularValueDecomposition[N[M]], i - 1];
 (* Defines i-th SVD of matrix M for i>0 *)
```

```
SingularValueDecomposition[N[H1A]][[2]] // MatrixForm
```

$$\begin{pmatrix}
38.9672 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\
0. & 11.4721 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\
0. & 0. & 10.6083 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 9.94483 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 9.51711 & 0. & 0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & 9.01469 & 0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & 8.91901 & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 8.27073 & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 7.16585 & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 6.45364 & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 5.892
\end{pmatrix}$$

```
SVD1[1, H1A, 2] // MatrixForm
```

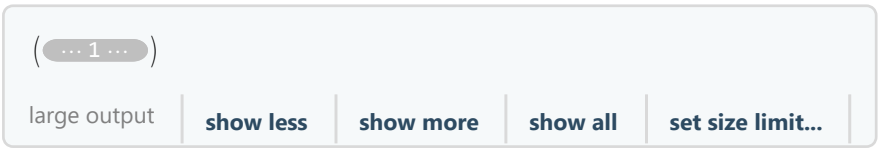
$$\begin{pmatrix}
\{\{-0.448207, 0.197445, 0.0845012, -0.159649, 0.285422, 0.11021, 0.436264, -0.393195, -0.41
\end{pmatrix}$$

large output | show less | show more | show all | set size limit...

```
SVD1[3, H1A, 1] // MatrixForm
```

$$\begin{pmatrix}
\begin{pmatrix} 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ -1. \\ 0. \\ 0. \\ 0. \end{pmatrix} &
\begin{pmatrix} 0. \\ 0. \\ 0. \\ 1. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \end{pmatrix} &
\begin{pmatrix} -0.582634 \\ 0.217355 \\ 0.127352 \\ 0. \\ -0.206777 \\ -0.402206 \\ 0.171949 \\ 0. \\ 0.353908 \\ 0.287098 \\ -0.394091 \end{pmatrix} &
\begin{pmatrix} 0.317829 \\ 0.118163 \\ -0.582274 \\ 0. \\ -0.154698 \\ -0.474662 \\ 0.36524 \\ 0. \\ -0.321734 \\ 0.243725 \\ 0.0207129 \end{pmatrix} &
\begin{pmatrix} 0.218414 \\ 0.41741 \\ 0.409648 \\ 0. \\ -0.593832 \\ -0.129908 \\ 0.0534872 \\ 0. \\ 0.0723902 \\ -0.161652 \\ 0.454431 \end{pmatrix} &
\begin{pmatrix} -0.159134 \\ 0.146474 \\ -0.573028 \\ 0. \\ 0.0272768 \\ -0.0674044 \\ -0.0890719 \\ 0. \\ 0.514775 \\ -0.551052 \\ 0.207334 \end{pmatrix} &
\begin{pmatrix} 0 \\ 0 \\ 0 \\ \\ 0 \\ -(\\ -(\\ \\ -(\\ -(\\ - \end{pmatrix} \\
\begin{pmatrix} 1. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \end{pmatrix} &
\begin{pmatrix} 0. \\ 1. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \end{pmatrix} &
\begin{pmatrix} 0. \\ 0. \\ 1. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \end{pmatrix} &
\begin{pmatrix} 0. \\ 0. \\ 0. \\ 1. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \end{pmatrix} &
\begin{pmatrix} 0. \\ 0. \\ 0. \\ 0. \\ 1. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \end{pmatrix} &
\begin{pmatrix} 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 1. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \end{pmatrix} & \\
\begin{pmatrix} -0.203525 \\ -0.143591 \\ 0.27915 \\ -0.234441 \\ 0.552078 \\ -0.0377495 \\ 0.245082 \\ 0. \\ 0.54059 \\ 0.381413 \\ 0.0328402 \end{pmatrix} &
\begin{pmatrix} -0.0940434 \\ 0.167737 \\ 0.142206 \\ -0.226493 \\ -0.174382 \\ 0.708945 \\ -0.120078 \\ 0. \\ -0.248898 \\ 0.527516 \\ -0.0620083 \end{pmatrix} &
\begin{pmatrix} 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ 0. \\ -1. \\ 0. \\ 0. \\ 0. \end{pmatrix} &
\begin{pmatrix} -0.324094 \\ 0.590837 \\ -0.0109934 \\ -0.177835 \\ 0.297195 \\ 0.0385864 \\ 0.446018 \\ 0. \\ -0.296032 \\ -0.330626 \\ -0.168625 \end{pmatrix} &
\begin{pmatrix} -0.150415 \\ 0.217666 \\ 0.0745488 \\ 0.586318 \\ -0.280075 \\ 0.0965192 \\ 0.43458 \\ 0. \\ 0.174206 \\ 0.153234 \\ 0.500225 \end{pmatrix} &
\begin{pmatrix} -0.0622807 \\ 0.332582 \\ -0.287872 \\ -0.0194922 \\ 0.0585911 \\ -0.577921 \\ -0.17109 \\ 0. \\ -0.249141 \\ 0.603361 \\ 0.0971933 \end{pmatrix} &
\begin{pmatrix} -(\\ -(\\ 0 \\ 0 \\ 0 \\ 0. \\ -(\\ \\ -(\\ 0. \\ -(\end{pmatrix}
\end{pmatrix}$$

```
SVD1[5, H1A, 3] // MatrixForm
```



Full SVD

Pre

```
lengthvec[M_] := Length[M[[1, All]]]
Filler[M_] := Table[4, {i, 1, FilledSize - lengthvec[M]}]
FilledVec[M_] := Join[Flatten[M], Filler[M]]

(*
For[npow=1,npow<1000,npow++,If[lengthvec[M]<(2^npow),Break[]]]
  npower[M_]:=npow *)

For[npow = 1, npow < 1000, npow++, If[lengthvec[H1Avec] < (2^npow), Break[]]];
(* gives npow such that  2^npow > lengthvec[M] > 2^(npow -1) *)
FilledSize = 2^npow;

npow
Length[Filler[H1Avec]]
```

```
10
```

```
243
```

```
Length[Join[Flatten[H1Avec], Filler[H1Avec]]]
```

```
1024
```

So now FilledVec[M_] will give us the vector filled up the the nearest power of 2, with the filler (list of 4's) at the end

```
Table[Flatten[H1Avec][[(71 * (k1 - 1)) + k2]], {k1, 1, 11}, {k2, 1, 71}] // MatrixForm
```

3	0	2	2	1	3	2	1	2	3	3	2	2	2	2	1	1	3	3	3	3	3	3	3	1	2	1	0	3	1	1	3	2	1	3	3	1
3	3	0	2	3	1	0	1	1	0	3	2	3	1	3	2	0	0	0	1	0	2	3	2	1	1	3	1	1	1	2	1	1	1	1	1	2
1	0	0	2	0	0	2	2	1	0	0	0	2	0	0	0	1	1	3	2	1	3	0	0	2	2	1	3	2	1	0	2	1	0	2	1	1
0	2	1	3	2	0	3	1	2	3	2	1	0	2	2	1	3	2	1	3	3	1	1	3	1	1	3	1	3	0	0	2	2	0	2	1	2
1	3	2	2	1	2	2	1	1	2	1	0	2	2	1	3	0	1	2	0	1	2	3	2	2	0	2	0	0	2	0	0	1	0	0	1	0
1	0	0	2	2	2	0	0	1	2	3	3	2	2	3	2	1	0	2	0	1	0	0	0	2	2	2	3	0	1	1	2	2	0	2	1	1
1	1	2	3	2	2	0	0	0	1	1	0	0	2	1	1	1	2	2	1	2	1	1	3	1	0	0	0	2	2	3	2	2	1	3	0	1
0	0	0	0	0	2	2	1	1	0	1	2	2	2	2	2	1	3	0	2	1	0	0	0	0	0	2	0	2	1	2	3	1	0	0	2	0
2	0	0	0	3	1	1	3	1	1	0	0	2	0	0	3	1	1	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	3	2	3	0
1	3	0	0	2	2	1	3	2	3	0	0	0	0	1	1	1	0	0	2	2	1	2	2	1	1	0	0	2	2	1	3	0	2	2	2	3
2	1	0	1	1	1	0	0	2	0	0	0	0	0	2	3	0	0	0	3	3	1	0	2	3	3	0	2	0	0	2	3	3	3	1	3	3

```
w0 = Table[Flatten[FilledVec[H1Avec]][[((FilledSize/2) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}];
w0 = w0;
W0 = w0;

w0 // MatrixForm;
Length[w0[[1, All]]]
Length[w0[[All, 1]]]
```

512

2

```
SingularValueDecomposition[N[w0]];

M[i_] := SingularValueDecomposition[N[w0]][[i]] // MatrixForm

Ma[i_] := SingularValueDecomposition[N[wa]][[i]] // MatrixForm

√SingularValueDecomposition[N[w0]][[2]] ;

w1 = √SingularValueDecomposition[N[w0]][[2]] .SingularValueDecomposition[N[w0]][[3]];
w2 = √SingularValueDecomposition[N[w1]][[2]] .SingularValueDecomposition[N[w1]][[3]];
w3 = √SingularValueDecomposition[N[w2]][[2]] .SingularValueDecomposition[N[w2]][[3]];
w4 = √SingularValueDecomposition[N[w3]][[2]] .SingularValueDecomposition[N[w3]][[3]];

For[a = 0, a < npow, a++,
 wa+1 = √SingularValueDecomposition[N[wa]][[2]] .SingularValueDecomposition[N[wa]][[3]]]
```

This For function defines the w matrices iteratively. Can check that this works by comparing these iterative fns to the w1, w2, w3, w4, which are defined explicitly above. Defines it up to a < npow = 10 here, so entering w_a will ouput w_{10}

```
w4  // MatrixForm;
(*defined explicitly above *)
w4  // MatrixForm;
(*defined in the iterartion above,
these two should be the same if iteration working properly *)

w9 // MatrixForm
```

$$\begin{pmatrix} -0.0426717 & 0.0637524 & -0.0488166 & -0.049993 & -0.0242122 & -0.0745974 & -0.049993 & -0.0249965 \\ -0.0124598 & -0.0146498 & -0.0190823 & 0.0386133 & -0.0191571 & 0.0386881 & 0.0386133 & 0.0193067 \end{pmatrix}$$

```
ub_ := SingularValueDecomposition[N[wb]][[1]]
λb_ := SingularValueDecomposition[N[wb]][[2]]

mb_ := ub.√λb
```

So that w_b =

$U_b \Lambda_b V_b = U_b \sqrt{\Lambda_b}\, w_{b+1} = M_b\,(w_{b+1}) = M_b M_{b+1}\,(w_{b+2}) = M_b M_{b+1} M_{b+2}\,(w_{b+3}) = \ldots$

$w_0 = M_0\,(w_1) = M_0 M_1\,(w_2) =$

$M_0\ M_1\ M_2\ (\ w_3\ )\ =\ \ldots\ =\ (\ M_0\ M_1\ M_2\ \ldots\ M_8\ )\ (\ w_9\ )\ =\ \text{Product}[M_i,\ \{i,\ 0,\ \text{npow}-2\}]\ w_{\text{npow}-1}$

```
Join[Table[Mi, {i, 0, npow - 1}], Table[wi, {i, 0, npow - 1}]];

w0[[2, 512]]
```

4

```
MatrixRank[SingularValueDecomposition[N[w0]][[2]]]
MatrixRank[
 √SingularValueDecomposition[N[w0]][[2]] .SingularValueDecomposition[N[w0]][[3]]]
```

2

2

MatrixRank[SingularValueDecomposition[w_a][[2]]] = d_{a+1}

Or should it be

MatrixRank[$\sqrt{\text{SingularValueDecomposition}[N[w_a]][[2]]}$.
SingularValueDecomposition[N[w_a]][[3]]] ?

Or are they always the same, as in the example above this

No minus 1 term since here we start from i = 1, not i = −1, as in the notes

W_{a+1} =

Table[W_a[[k1, k2]], {k1, 1, (MatrixRank[$\sqrt{\text{SingularValueDecomposition}[N[W_a]][[2]]}$.
SingularValueDecomposition[N[W_a]][[3]])}, {k2, 1, $\frac{\text{FilledSize}}{(2\text{^}(a+1))}$}]

```
Flatten[FilledVec[H1Avec]];

W0 = Table[Flatten[FilledVec[H1Avec]][[((FilledSize/2) * (k1 - 1)) + k2]],
   {k1, 1, 2}, {k2, 1, FilledSize/2}];

For[a = 0, a < npow, a++,
 wa+1 = √SingularValueDecomposition[N[wa]][[2]] .SingularValueDecomposition[N[wa]][[3]]]

w1[[1, 14]]
(√SingularValueDecomposition[N[w0]][[2]] .SingularValueDecomposition[N[w0]][[3]])[[
 1, 14]]
```

−0.417031

−0.417031

```
W0 // MatrixForm
```

3	0	2	2	1	3	2	1	2	3	3	2	2	2	2	1	1	3	3	3	3	3	3	3	1	2	1	0	3	1	1	3	2	1	3	3	1
2	1	3	0	2	1	0	0	0	0	0	2	0	2	1	2	3	1	0	0	2	0	1	3	1	1	2	0	0	0	0	0	2	2	1	3	0

```
SingularValueDecomposition[N[W0]][[3]]
```

```
{{-0.0423155, 0.0632203, -0.0484092, -0.0495758, -0.0240101, -0.0739748, -0.0495758,
  -0.0247879, ··· 496 ···, -0.0232324, -0.0232324, 0.00155548, -0.0232324,
  -0.0480203, -0.0480203, -0.0480203, -0.0480203}, ··· 510 ···, {··· 1 ···}}
```

large output | show less | show more | show all | set size limit...

Main Iterative Function

First Try

```
For[b = 0, b < npow, b++, Wb+1 = Table[
   (√SingularValueDecomposition[N[Wb]][[2]].SingularValueDecomposition[N[Wb]][[3]])[[
    k1, k2]], {k1, 1, (MatrixRank[√SingularValueDecomposition[N[Wb]][[2]].
       SingularValueDecomposition[N[Wb]][[3]]])}, {k2, 1, FilledSize/(2^(b + 2))}]]
```

```
W4;
```

Seems to work well, but should check it by doing a few explicitly

Also need to check that same thing done in step 4 of the notes is done here

```
Do[Print[Wi // MatrixForm], {i, 0, 9}]
```

$$\begin{pmatrix} 3 & 0 & 2 & 2 & 1 & 3 & 2 & 1 & 2 & 3 & 3 & 2 & 2 & 2 & 2 & 1 & 1 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 1 & 2 & 1 & 0 & 3 & 1 & 1 & 3 & 2 & 1 & 3 & 3 & 1 & 2 & 3 & 1 & 0 & 2 \\ 2 & 1 & 3 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 2 & 1 & 2 & 3 & 1 & 0 & 0 & 2 & 0 & 1 & 3 & 1 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 1 & 3 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix}$$

$$\begin{pmatrix} -0.361631 & 0.540285 & -0.413708 & -0.423678 & -0.205192 & -0.632193 & -0.423678 & -0.211839 & -0.423678 \\ -0.0670854 & -0.0788769 & -0.102742 & 0.2079 & -0.103144 & 0.208302 & 0.2079 & 0.10395 & 0.2079 \end{pmatrix}$$

$$\begin{pmatrix} -0.127116 & 0.219195 & -0.164633 & -0.16981 & -0.0814536 & -0.25299 & -0.16981 & -0.0849052 & -0.169 \\ 0.112034 & -0.0517999 & 0.145977 & -0.0677682 & 0.108613 & -0.0304037 & -0.0677682 & -0.0338841 & -0.0677 \end{pmatrix}$$

$$\begin{pmatrix} -0.081631 & 0.188062 & -0.121193 & -0.118757 & -0.0610025 & -0.178947 & -0.118757 & -0.0593784 & -0.1187 \\ 0.102943 & -0.0758696 & 0.162571 & -0.0579745 & 0.118043 & -0.0134464 & -0.0579745 & -0.0289872 & -0.0579 \end{pmatrix}$$

$$\begin{pmatrix} -0.0838193 & 0.150195 & -0.111196 & -0.117428 & -0.0545593 & -0.174065 & -0.117428 & -0.058714 & -0.11742 \\ 0.147101 & -0.079913 & 0.193978 & -0.0720602 & 0.141329 & -0.019411 & -0.0720602 & -0.0360301 & -0.07206 \end{pmatrix}$$

$$\begin{pmatrix} -0.0690133 & 0.235209 & -0.143719 & -0.130629 & -0.0740409 & -0.200307 & -0.130629 & -0.0653144 & -0.1306 \\ 0.139613 & -0.135027 & 0.258751 & -0.0921161 & 0.187854 & -0.0212184 & -0.0921161 & -0.0460581 & -0.0921 \end{pmatrix}$$

$$\begin{pmatrix} -0.166383 & 0.130688 & -0.147544 & -0.20175 & -0.0647378 & -0.284557 & -0.20175 & -0.100875 \\ 0.32495 & -0.00178618 & 0.335858 & -0.112375 & 0.242635 & -0.019152 & -0.112375 & -0.0561877 \end{pmatrix}$$

$$\begin{pmatrix} -0.47472 & -0.0503484 & -0.455681 & -0.0713477 \\ 0.0804863 & -0.163872 & -0.21377 & 0.325753 \end{pmatrix}$$

$$\begin{pmatrix} -0.555886 & 0.306827 \\ -0.0745188 & -0.236039 \end{pmatrix}$$

$$\begin{pmatrix} -0.685256 \\ 0.252184 \end{pmatrix}$$

These are the W from W_0 up to $W_{npow-1} = W_9$ per step 9 in the notes.

W_9 is a vector so we recover a scalar S later, as we want

```
Length[W0[[1, All]]]
Length[W1[[1, All]]] (*Want to make sure W1 has half the # of colums as W0*)
```

512

256

Second Try

```
For[b = 0, b < npow, b++, Wb+1 = Table[
    (√SingularValueDecomposition[N[Wb]][[2]].SingularValueDecomposition[N[Wb]][[3]])[[
     k1, k2]], {k1, 1, (MatrixRank[√SingularValueDecomposition[N[Wb]][[2]].
         SingularValueDecomposition[N[Wb]][[3]]])}, {k2, 1, FilledSize/(2^(b + 2))}]]
```

```
{{1, 2}, {3, 4}} // MatrixForm
Diagonal[((1 2) (3 4))]
Diagonal[((1 2) (3 4))][[2]]
```

```
√SingularValueDecomposition[N[W3]][[2]] // MatrixForm
```

$$\begin{pmatrix} 1.03693 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0.824735 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

```
Length[Diagonal[√SingularValueDecomposition[N[W3]][[2]]]]
(* (= d3 = its rank) iff the Λ's have no non zero diagonal *)
MatrixRank[√SingularValueDecomposition[N[W3]][[2]]]
Diagonal[√SingularValueDecomposition[N[W3]][[2]]][[1]]
√Diagonal[SingularValueDecomposition[N[W3]][[2]]][[1]]
```

2

2

1.03693

1.03693

```
√SingularValueDecomposition[N[W3]][[2]] // MatrixForm;
SingularValueDecomposition[N[W3]][[3]] // MatrixForm;

Length[SingularValueDecomposition[N[W3]][[3]][[1, All]]]
Length[SingularValueDecomposition[N[W3]][[3]][[All, 1]]]
```

64

64

```
1024 / (2^(2 + 2))
```

64

```
If[EvenQ[2], 1, 0]
```

```
1
```

```
Diagonal[√SingularValueDecomposition[N[W3]][[2]]][[i]]
```

```
WW4[[1, 3]] = √SingularValueDecomposition[N[W3]][[2]]
```

```
WW4[[2 (i - 1) + j0, (j - j0)/2]] =
 Diagonal[√SingularValueDecomposition[N[W3]][[2]]][[i]] .
  (SingularValueDecomposition[N[W3]][[3]][[i, j]])
```

```
j0 → If[EvenQ[j], 1, 0]
```

```
(* WW4[[ 2(1-1)+ 1, (7-1)/2 ]]= WW4[[ 0 + 1, 3 ]]= WW4[[ 1,3 ]]= *)
(Diagonal[√SingularValueDecomposition[N[W3]][[2]]][[1]])
 (SingularValueDecomposition[N[W3]][[3]][[1, 7]])
```

```
-0.117428
```

```
W0 = Table[Flatten[FilledVec[H1Avec]][[((FilledSize/2) * (k1 - 1)) + k2]],
   {k1, 1, 2}, {k2, 1, FilledSize/2}];
```

```
Table[ (Diagonal[√SingularValueDecomposition[N[W0]][[2]]][[i]])
  ((SingularValueDecomposition[N[W0]][[3]][[i, j]])),
 {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}, {j, 1, FilledSize/(2^(0 + 2))}]
```

```
W11[[2 (i - 1) + If[EvenQ[j], 1, 0], If[EvenQ[j], j/2, (j - 1)/2]]]
```

```
Table[WW11[[2 (i - 1) + 1 + If[EvenQ[j], 1, 0], ((j - 1) - If[EvenQ[j], 1, 0])/2 + 1]],
 {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}, {j, 1, FilledSize/(2^(0 + 2))}]
```

Note that we can replace `J = j' = (j - j0)/2 → If[EvenQ[j], j/2, (j - 1)/2]`,
since `j0 = If[EvenQ[j], 1, 0]`

Here we also let i and j from the notes go to `i → (i - 1)` and `j → (j - 1)` on the left hand sides,
since we start at (1, 1) place, not (0, 0) place of matrices
Also add `+ 1` to both defining sides of W1

```
Table[ (Diagonal[√SingularValueDecomposition[N[W0]][[2]]][[i]])
    ((SingularValueDecomposition[N[W0]][[3]][[i, j]])),
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]},
  {j, 1, FilledSize/(2^(0 + 2))}] // MatrixForm
(* Need to rearrange this to get W1 as desired in step 4 of the notes *)
```

$$\begin{pmatrix} -0.361631 & 0.540285 & -0.413708 & -0.423678 & -0.205192 & -0.632193 & -0.423678 & -0.211839 & -0 \\ -0.0670854 & -0.0788769 & -0.102742 & 0.2079 & -0.103144 & 0.208302 & 0.2079 & 0.10395 & 0 \end{pmatrix}$$

```
(* Length[Flatten[Table[WW11[[2(i-1)+1+ If[EvenQ[j],1,0], ((j-1)-If[EvenQ[j],1,0])/2 +1]],
    {i,1,MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]},{j,1,FilledSize/(2^(0+2))}]]]
 Length[Union[Flatten[Table[WW11[[2(i-1)+1+ If[EvenQ[j],1,0], ((j-1)-If[EvenQ[j],1,0])/2 +1]],
    {i,1,MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]},{j,1,FilledSize/(2^(0+2))}]]]] *)
(*These being equal tells us that this table fn only contains each WW11[i,j]]
 once for a given i and j *)
(* Table[WW11[[2(i-1)+1+ If[EvenQ[j],1,0], ((j-1)-If[EvenQ[j],1,0])/2 +1]],
 {i,1,MatrixRank[√SingularValueDecomposition[N[W0]][[2]] ]},{j,1,FilledSize/(2^(0+2))}] *)
Length[Table[WW11[[2 (i - 1) + 1 + If[EvenQ[j], 1, 0], ((j - 1) - If[EvenQ[j], 1, 0])/2 + 1]],
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]},
  {j, 1, FilledSize/(2^(0 + 2))}][[1, All]]]
```

Part::partd : Part specification WW11[[1, 1]] is longer than depth of object. ≫

Part::partd : Part specification WW11[[2, 1]] is longer than depth of object. ≫

Part::partd : Part specification WW11[[1, 2]] is longer than depth of object. ≫

General::stop : Further output of Part::partd will be suppressed during this calculation. ≫

256

Now in order to define W1 as in the notes, let

$2\,(i-1)+1+j0 = x$

so that $i = \frac{(x-j0-1)}{2}+1$

$\frac{((j-1)-j0)}{2}+1 = y$

so that $j = 2\,(y-1)+j0+1$

W1[[x, y]] =

```
Table[ (Diagonal[√SingularValueDecomposition[N[W0]][[2]]][[ (x - j0 - 1)/2 + 1]])
```

```
    (((SingularValueDecomposition[N[W₀]][[3]][[
        (x - j0 - 1)/2 + 1, 2 (y - 1) + j0 + 1]]))),
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W₀]][[2]]]},
  {j, 1, FilledSize/(2^(0 + 2))}] // MatrixForm
```

```
Table[2 (i - 1) + 1, {i, 1, MatrixRank[√SingularValueDecomposition[N[W₀]][[2]]]}]
Table[2 (i - 1) + 2, {i, 1, MatrixRank[√SingularValueDecomposition[N[W₀]][[2]]]}]
Table[((j - 1) - If[EvenQ[j], 1, 0])/2 + 1, {j, 1, FilledSize/(2^(0 + 2))}]
Length[Table[((j - 1) - If[EvenQ[j], 1, 0])/2 + 1, {j, 1, FilledSize/(2^(0 + 2))}]]
```

```
{1, 3}

{2, 4}

{1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10, 11, 11, 12, 12, 13, 13, 14, 14, 15,
 15, 16, 16, 17, 17, 18, 18, 19, 19, 20, 20, 21, 21, 22, 22, 23, 23, 24, 24, 25, 25, 26,
 26, 27, 27, 28, 28, 29, 29, 30, 30, 31, 31, 32, 32, 33, 33, 34, 34, 35, 35, 36, 36, 37,
 37, 38, 38, 39, 39, 40, 40, 41, 41, 42, 42, 43, 43, 44, 44, 45, 45, 46, 46, 47, 47, 48,
 48, 49, 49, 50, 50, 51, 51, 52, 52, 53, 53, 54, 54, 55, 55, 56, 56, 57, 57, 58, 58, 59,
 59, 60, 60, 61, 61, 62, 62, 63, 63, 64, 64, 65, 65, 66, 66, 67, 67, 68, 68, 69, 69, 70,
 70, 71, 71, 72, 72, 73, 73, 74, 74, 75, 75, 76, 76, 77, 77, 78, 78, 79, 79, 80, 80, 81,
 81, 82, 82, 83, 83, 84, 84, 85, 85, 86, 86, 87, 87, 88, 88, 89, 89, 90, 90, 91, 91, 92,
 92, 93, 93, 94, 94, 95, 95, 96, 96, 97, 97, 98, 98, 99, 99, 100, 100, 101, 101, 102, 102,
 103, 103, 104, 104, 105, 105, 106, 106, 107, 107, 108, 108, 109, 109, 110, 110, 111, 111,
 112, 112, 113, 113, 114, 114, 115, 115, 116, 116, 117, 117, 118, 118, 119, 119, 120,
 120, 121, 121, 122, 122, 123, 123, 124, 124, 125, 125, 126, 126, 127, 127, 128, 128}

256
```

```
(* For[i=1,i<MatrixRank[√SingularValueDecomposition[N[W₀]][[2]]], i++,
 WW11[[2(i-1)+1+ If[EvenQ[j],1,0], ((j-1)-If[EvenQ[j],1,0])/2+1]] =
   (Diagonal[√SingularValueDecomposition[N[W₀]][[2]]][[i]])
   ((SingularValueDecomposition[N[W₀]][[3]][[i,j]]))

   {i,1,MatrixRank[√SingularValueDecomposition[N[W₀]][[2]]]},{j,1,FilledSize/(2^(0+2))}] *)
```

```
WD = Table[1,
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}, {j, 1, FilledSize/(2^(0 + 2))}]
```

```
{{1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1},
 {1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1}}
```

```
Length[WD[[1, All]]]
```

```
256
```

```
WD[[3, 4]] =  (Diagonal[√SingularValueDecomposition[N[W0]][[2]]][[(x - j0 - 1)/2 + 1]])
  ((SingularValueDecomposition[N[W0]][[3]][[(x - j0 - 1)/2 + 1, 2 (y - 1) + j0 + 1]]))
```

WD

```
{{-0.361631, -0.413708, -0.205192, -0.423678, -0.423678, -0.635517, -0.423678, -0.420355,
  -0.201869, -0.635517, -0.62887, -0.632193, -0.208516, -0.205192, -0.635517, -0.211839,
  -0.417031, -0.632193, -0.211839, -0.635517, 0., -0.420355, -0.625547, 0.00332331,
  0.00664662, -0.211839, -0.208516, -0.635517, 0.00664662, -0.635517, -0.413708,
  -0.632193, -0.632193, -0.211839, -0.417031, -0.423678, -0.632193, -0.423678, -0.211839,
  -0.211839, 0., -0.420355, -0.211839, -0.423678, 0.00332331, -0.205192, -0.423678,
  -0.423678, -0.208516, -0.208516, -0.211839, -0.211839, -0.211839, -0.201869, -0.205192,
  -0.413708, -0.635517, -0.211839, -0.420355, -0.632193, -0.205192, -0.201869,
  -0.635517, 0.00664662, 0.00996994, 0., -0.205192, -0.625547, 0.00996994, -0.211839,
  -0.423678, -0.208516, 0., 0.00664662, -0.420355, -0.205192, 0.00332331, -0.423678,
  0.00664662, -0.201869, -0.62887, -0.205192, 0.00664662, -0.420355, -0.211839,
  -0.417031, 0.00332331, -0.211839, -0.417031, -0.208516, -0.205192, 0.00332331,
  -0.423678, 0.00332331, 0.00332331, 0., -0.211839, -0.423678, -0.632193, -0.417031,
  -0.211839, -0.632193, -0.211839, -0.417031, -0.423678, -0.211839, -0.417031,
  -0.423678, -0.635517, 0.00996994, -0.211839, -0.625547, -0.211839, -0.423678,
  -0.205192, -0.413708, -0.632193, -0.201869, -0.625547, -0.205192, -0.211839,
  0.00332331, -0.420355, 0., -0.205192, -0.632193, -0.420355, -0.413708, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1},
 {0.540285, -0.423678, -0.632193, -0.211839, -0.635517, -0.417031, -0.417031,
  -0.205192, -0.632193, -0.635517, -0.635517, -0.625547, -0.420355, 0., -0.211839,
  -0.635517, -0.205192, -0.625547, -0.423678, -0.211839, -0.417031, -0.632193,
  -0.62887, -0.62887, -0.208516, 0., -0.635517, -0.62887, -0.635517, -0.635517,
  -0.420355, -0.413708, -0.420355, -0.635517, -0.635517, -0.625547, 0.00332331,
  -0.635517, 0., -0.211839, -0.632193, -0.632193, -0.635517, 0., 0.00996994, 0.00996994,
  -0.635517, -0.205192, -0.632193, -0.211839, -0.417031, -0.211839, -0.205192,
  -0.423678, -0.208516, -0.211839, -0.635517, -0.62887, -0.208516, -0.413708,
  -0.632193, -0.211839, -0.423678, -0.420355, 0., -0.205192, -0.632193, -0.62887,
  -0.423678, -0.635517, -0.420355, 0.00332331, -0.423678, 0.00664662, -0.417031,
  0.00332331, 0., 0.00664662, 0.00332331, -0.211839, -0.417031, -0.625547, 0.,
  -0.417031, -0.635517, -0.208516, -0.423678, 0., -0.211839, -0.625547, -0.208516,
  0., 0., 0.00332331, 0., -0.205192, -0.211839, -0.205192, -0.417031, -0.208516,
  -0.208516, -0.632193, -0.211839, -0.635517, -0.635517, 0., 0.00996994, -0.211839,
  -0.413708, -0.632193, -0.417031, -0.413708, 0.00664662, -0.423678, -0.625547,
  -0.201869, -0.625547, -0.208516, -0.211839, -0.625547, -0.635517, 0.00332331,
  -0.423678, -0.420355, -0.417031, -0.413708, -0.625547, -0.413708, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1}}
```

```
Table[ (Diagonal[√SingularValueDecomposition[N[W₀]][[2]]][[i]])
    ((SingularValueDecomposition[N[W₀]][[3]][[i, j]])),
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W₀]][[2]]]}, {j, 1, FilledSize/(2^(0 + 2))}];
Length[Table[ (Diagonal[√SingularValueDecomposition[N[W₀]][[2]]][[i]])
     ((SingularValueDecomposition[N[W₀]][[3]][[i, j]])),
   {i, 1, MatrixRank[√SingularValueDecomposition[N[W₀]][[2]]]},
   {j, 1, FilledSize/(2^(0 + 2))}][[1, All]]]
```

256

```
Do[Print[WD[[2 (i - 1) + 1 + If[EvenQ[j], 1, 0], ((j - 1) - If[EvenQ[j], 1, 0])/2 + 1]] =
    (Diagonal[√SingularValueDecomposition[N[W₀]][[2]]][[i]])
     ((SingularValueDecomposition[N[W₀]][[3]][[i, j]]))],
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W₀]][[2]]]}, {j, 1, FilledSize/(2^(0 + 2))}]
```

−0.361631

0.540285

−0.413708

−0.423678

−0.205192

−0.632193

−0.423678

−0.211839

−0.423678

−0.635517

−0.635517

−0.417031

−0.423678

−0.417031

−0.420355

−0.205192

−0.201869

−0.632193

−0.635517

−0.635517

−0.62887

−0.635517

```
-0.632193
-0.625547
-0.208516
-0.420355
-0.205192
0.
-0.635517
-0.211839
-0.211839
-0.635517
-0.417031
-0.205192
-0.632193
-0.625547
-0.211839
-0.423678
-0.635517
-0.211839
0.
-0.417031
-0.420355
-0.632193
-0.625547
-0.62887
0.00332331
-0.62887
0.00664662
-0.208516
-0.211839
0.
-0.208516
-0.635517
-0.635517
-0.62887
0.00664662
-0.635517
```

−0.635517

−0.635517

−0.413708

−0.420355

−0.632193

−0.413708

−0.632193

−0.420355

−0.211839

−0.635517

−0.417031

−0.635517

−0.423678

−0.625547

−0.632193

0.00332331

−0.423678

−0.635517

−0.211839

0.

−0.211839

−0.211839

0.

−0.632193

−0.420355

−0.632193

−0.211839

−0.635517

−0.423678

0.

0.00332331

0.00996994

−0.205192

0.00996994

−0.423678

−0.635517

```
-0.423678
-0.205192
-0.208516
-0.632193
-0.208516
-0.211839
-0.211839
-0.417031
-0.211839
-0.211839
-0.211839
-0.205192
-0.201869
-0.423678
-0.205192
-0.208516
-0.413708
-0.211839
-0.635517
-0.635517
-0.211839
-0.62887
-0.420355
-0.208516
-0.632193
-0.413708
-0.205192
-0.632193
-0.201869
-0.211839
-0.635517
-0.423678
0.00664662
-0.420355
0.00996994
0.
```

```
0.
-0.205192
-0.205192
-0.632193
-0.625547
-0.62887
0.00996994
-0.423678
-0.211839
-0.635517
-0.423678
-0.420355
-0.208516
0.00332331
0.
-0.423678
0.00664662
0.00664662
-0.420355
-0.417031
-0.205192
0.00332331
0.00332331
0.
-0.423678
0.00664662
0.00664662
0.00332331
-0.201869
-0.211839
-0.62887
-0.417031
-0.205192
-0.625547
0.00664662
0.
```

−0.420355

−0.417031

−0.211839

−0.635517

−0.417031

−0.208516

0.00332331

−0.423678

−0.211839

0.

−0.417031

−0.211839

−0.208516

−0.625547

−0.205192

−0.208516

0.00332331

0.

−0.423678

0.

0.00332331

0.00332331

0.00332331

0.

0.

−0.205192

−0.211839

−0.211839

−0.423678

−0.205192

−0.632193

−0.417031

−0.417031

−0.208516

−0.211839

−0.208516

−0.632193

−0.632193

−0.211839

−0.211839

−0.417031

−0.635517

−0.423678

−0.635517

−0.211839

0.

−0.417031

0.00996994

−0.423678

−0.211839

−0.635517

−0.413708

0.00996994

−0.632193

−0.211839

−0.417031

−0.625547

−0.413708

−0.211839

0.00664662

−0.423678

−0.423678

−0.205192

−0.625547

−0.413708

−0.201869

−0.632193

−0.625547

−0.201869

−0.208516

−0.625547

−0.211839

−0.205192

−0.625547

−0.211839

−0.635517

0.00332331

0.00332331

−0.420355

−0.423678

0.

−0.420355

−0.205192

−0.417031

−0.632193

−0.413708

−0.420355

−0.625547

−0.413708

−0.413708

Set::partw : Part 3 of
{{−0.361631, −0.413708, −0.205192, −0.423678, −0.423678, −0.635517, −0.423678, −0.420355, −0.201869, −0.635517, ≪31≫, −0.420355, −0.211839, −0.423678, 0.00332331, −0.205192, −0.423678, −0.423678, −0.208516, −0.208516, ≪206≫}, {≪19≫, ≪49≫, ≪206≫}} does not exist. ≫

−0.0670854

Set::partw : Part 4 of
{{−0.361631, −0.413708, −0.205192, −0.423678, −0.423678, −0.635517, −0.423678, −0.420355, −0.201869, −0.635517, ≪31≫, −0.420355, −0.211839, −0.423678, 0.00332331, −0.205192, −0.423678, −0.423678, −0.208516, −0.208516, ≪206≫}, {≪19≫, ≪49≫, ≪206≫}} does not exist. ≫

−0.0788769

Set::partw : Part 3 of
{{−0.361631, −0.413708, −0.205192, −0.423678, −0.423678, −0.635517, −0.423678, −0.420355, −0.201869, −0.635517, ≪31≫, −0.420355, −0.211839, −0.423678, 0.00332331, −0.205192, −0.423678, −0.423678, −0.208516, −0.208516, ≪206≫}, {≪19≫, ≪49≫, ≪206≫}} does not exist. ≫

General::stop : Further output of Set::partw will be suppressed during this calculation. ≫

−0.102742

0.2079

−0.103144

0.208302

0.2079

0.10395

0.2079

0.31185

0.31185

0.000805444

0.2079

0.000805444

0.104353

−0.103144

−0.206692

0.208302

0.31185

0.31185

0.104755

0.31185

0.208302

0.00120817

0.000402722

0.104353

−0.103144

0.

0.31185

0.10395

0.10395

0.31185

0.000805444

−0.103144

0.208302

0.00120817

0.10395

0.2079

0.31185

0.10395

0.

0.000805444

0.104353

```
0.208302
0.00120817
0.104755
-0.103547
0.104755
-0.207094
0.000402722
0.10395
0.
0.000402722
0.31185
0.31185
0.104755
-0.207094
0.31185
0.31185
0.31185
-0.102742
0.104353
0.208302
-0.102742
0.208302
0.104353
0.10395
0.31185
0.000805444
0.31185
0.2079
0.00120817
0.208302
-0.103547
0.2079
0.31185
0.10395
0.
0.10395
```

0.10395

0.

0.208302

0.104353

0.208302

0.10395

0.31185

0.2079

0.

−0.103547

−0.310641

−0.103144

−0.310641

0.2079

0.31185

0.2079

−0.103144

0.000402722

0.208302

0.000402722

0.10395

0.10395

0.000805444

0.10395

0.10395

0.10395

−0.103144

−0.206692

0.2079

−0.103144

0.000402722

−0.102742

0.10395

0.31185

0.31185

0.10395

0.104755

0.104353

0.000402722

0.208302

−0.102742

−0.103144

0.208302

−0.206692

0.10395

0.31185

0.2079

−0.207094

0.104353

−0.310641

0.

0.

−0.103144

−0.103144

0.208302

0.00120817

0.104755

−0.310641

0.2079

0.10395

0.31185

0.2079

0.104353

0.000402722

−0.103547

0.

0.2079

−0.207094

−0.207094

0.104353

0.000805444

−0.103144

−0.103547

−0.103547

0.

0.2079

−0.207094

−0.207094

−0.103547

−0.206692

0.10395

0.104755

0.000805444

−0.103144

0.00120817

−0.207094

0.

0.104353

0.000805444

0.10395

0.31185

0.000805444

0.000402722

−0.103547

0.2079

0.10395

0.

0.000805444

0.10395

0.000402722

0.00120817

−0.103144

0.000402722

−0.103547

0.

0.2079

0.

−0.103547

```
-0.103547
-0.103547
0.
0.
-0.103144
0.10395
0.10395
0.2079
-0.103144
0.208302
0.000805444
0.000805444
0.000402722
0.10395
0.000402722
0.208302
0.208302
0.10395
0.10395
0.000805444
0.31185
0.2079
0.31185
0.10395
0.
0.000805444
-0.310641
0.2079
0.10395
0.31185
-0.102742
-0.310641
0.208302
0.10395
0.000805444
0.00120817
```

−0.102742

0.10395

−0.207094

0.2079

0.2079

−0.103144

0.00120817

−0.102742

−0.206692

0.208302

0.00120817

−0.206692

0.000402722

0.00120817

0.10395

−0.103144

0.00120817

0.10395

0.31185

−0.103547

−0.103547

0.104353

0.2079

0.

0.104353

−0.103144

0.000805444

0.208302

−0.102742

0.104353

0.00120817

−0.102742

−0.102742

```
((SingularValueDecomposition[N[W0]][[3]]))
```

```
{{-0.0423155, 0.0632203, -0.0484092, -0.0495758, -0.0240101, -0.0739748, -0.0495758,
  -0.0247879, -0.0495758, ··· 494 ···, -0.0480203, -0.0232324, -0.0232324, 0.00155548,
  -0.0232324, -0.0480203, -0.0480203, -0.0480203, -0.0480203}, ··· 510 ···, {··· 1 ···}}
```

large output | show less | show more | show all | set size limit...

```
Do[WB[[2 (i - 1) + 1 + If[EvenQ[j], 1, 0], ((j - 1) - If[EvenQ[j], 1, 0])/2 + 1]] ==
   (Diagonal[√SingularValueDecomposition[N[W0]][[2]]][[i]])
    ((SingularValueDecomposition[N[W0]][[3]][[i, j]])),
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}, {j, 1, FilledSize/(2^(0 + 2))}]
```

Part::partd : Part specification WB[[1, 1]] is longer than depth of object. ≫

Part::partd : Part specification WB[[2, 1]] is longer than depth of object. ≫

Part::partd : Part specification WB[[1, 2]] is longer than depth of object. ≫

General::stop : Further output of Part::partd will be suppressed during this calculation. ≫

```
WW11[[2 (i - 1) + 1 + If[EvenQ[j], 1, 0], ((j - 1) - If[EvenQ[j], 1, 0])/2 + 1]]
 (Diagonal[√SingularValueDecomposition[N[W0]][[2]]][[i]])
  ((SingularValueDecomposition[N[W0]][[3]][[i, j]]))
```

Second Function

```
m // MatrixForm
```

$$\begin{pmatrix} -0.361631 & 0.540285 & -0.413708 & -0.423678 & -0.205192 & -0.632193 & -0.423678 & -0.211839 & -0 \\ -0.0670854 & -0.0788769 & -0.102742 & 0.2079 & -0.103144 & 0.208302 & 0.2079 & 0.10395 & 0 \end{pmatrix}$$

```
(* m[[1,2]]= *)
```

```
0.540285
```

```
(* For[b=0,b<npow,b++,Wb+1=Table[

   (√SingularValueDecomposition[N[Wb]][[2]].
      SingularValueDecomposition[N[Wb]][[3]])[[k1,k2]],


   {k1,1,(MatrixRank[√SingularValueDecomposition[N[Wb]][[2]].
       SingularValueDecomposition[N[Wb]][[3]]])},{k2,1,FilledSize/(2^(b+2))}]] *)
```

```
m = Table[ (Diagonal[√SingularValueDecomposition[N[W0]][[2]]][[i]])
     ((SingularValueDecomposition[N[W0]][[3]][[i, j]])),
   {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}, {j, 1, FilledSize/(2^(0 + 2))}];
```

The above is a rearranged form of the W1 in the notes??? but contains all the terms, so we'll work with this for now (very difficult to rearrange in that way)
W1 should be 4x256 here

```
SingularValueDecomposition[N[W0]][[3]][[1, 512]]
SingularValueDecomposition[N[W0]][[3]][[1, 513]];
```

```
-0.0480203
```

```
Part::partw : Part 513 of
    {{-0.0423155, 0.0632203, -0.0484092, -0.0495758, -0.0240101, -0.0739748, -0.0495758, -0.0247879, -0.0495758, -
        0.0743637, <<31>>, -0.048798, -0.0491869, -0.0739748, -0.073197, -0.0735859, 0.00038887, -0.0735859, 0.000777741,
      -0.024399, <<462>>}, <<49>>, <<462>>} does not exist. >>
```

```
Diagonal[√SingularValueDecomposition[N[W0]][[2]]][[1]]
  (SingularValueDecomposition[N[W0]][[3]][[1, 3]])
```

```
-0.413708
```

```
WP[i_, j_] := Diagonal[√SingularValueDecomposition[N[W0]][[2]]][[i]]
   (SingularValueDecomposition[N[W0]][[3]][[i, j]])

G[j_] := WP[1, j]
H[i_] := WP[i, j]

Table[G[j], {j, 1, FilledSize/(2^(0 + 2))}];

Table[WP[i, j], {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]},
   {j, 1, FilledSize/(2^(0 + 2))}][[2, 5]]

WP[
 2,
 5]
```

```
-0.103144
```

```
-0.103144
```

Now in order to define W1 as in the notes, let

$2(i-1)+1+j0 = x$

so that $i = \frac{(x - j0 - 1)}{2} + 1$

$\frac{((j-1)-j0)}{2} + 1 = \frac{y}{2}$

```
(* Defined like this so that j is integer for integer y *)
```

$$\text{so that } j = 2\left(\frac{y}{2} - 1\right) + j0 + 1 = y + j0 - 1$$

$$(j+1) - j0 = y$$

So $j = 1 \Rightarrow y = (1+1) - 1 = 1$

So $j = 256 \Rightarrow y = (256+1) - 0 = 257$

$j0 = 0$ if j even, $= 1$ if j odd

So y is odd for all values of j

x is even iff j is odd (so $j0 = 1$ for even x)

x is odd iff j is even (so $j0 = 0$ for odd x)

So use the function

```
Ω[x_, y_] := If[OddQ[x], WP[(x + 1)/2, y - 1], WP[(x)/2, y]]
```

$$i \in \{1, 2\} \Rightarrow x \in \{1, 2, 3, 4\}$$

$$j \in \left\{1, \ldots, \frac{\text{FilledSize}}{(2\text{^}(0+2))}\right\} = \{1, \ldots, 256\} \Rightarrow y \in \{1, \ldots, \sim 256\}$$

```
Ω[x_, y_] := If[OddQ[x], WP[(x + 1)/2, y - 1], WP[(x)/2, y]]

Table[Ω[x, y], {x, 1, 4}, {y, 1, FilledSize/(2^(0 + 2))}] // MatrixForm
```

$$\begin{pmatrix}
8.54607\,\text{List} & -0.361631 & 0.540285 & -0.413708 & -0.423678 & -0.205192 & -0.632193 & -0.423678 \\
-0.361631 & 0.540285 & -0.413708 & -0.423678 & -0.205192 & -0.632193 & -0.423678 & -0.211839 \\
5.41981\,\text{List} & -0.0670854 & -0.0788769 & -0.102742 & 0.2079 & -0.103144 & 0.208302 & 0.2079 \\
-0.0670854 & -0.0788769 & -0.102742 & 0.2079 & -0.103144 & 0.208302 & 0.2079 & 0.10395
\end{pmatrix}$$

```
Length[Table[Ω[x, y], {x, 1, 4}, {y, 1, FilledSize/(2^(0 + 2))}][[1, All]]]
```

256

```
WP[i_, j_] := Diagonal[√SingularValueDecomposition[N[W0]][[2]]][[i]]
  (SingularValueDecomposition[N[W0]][[3]][[i, j]])

WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];

xeven[i_] := 2 (i - 1) + 2

yeven[j_] := ((j - 1))/2 + 1/2

xodd[i_] := 2 (i - 1) + 1

yodd[j_] := ((j - 1))/2 + 1
```

```
Do[WK[[xodd[i], yodd[j]]] = WP[i, j], {j, 1, FilledSize/(2^(0 + 2)), 2},
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}]
Do[WK[[xeven[i], yeven[j]]] = WP[i, j], {j, 2, FilledSize/(2^(0 + 2)), 2},
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}]
```

```
WWW = Table[WK[[i, j]],
    {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}, {j, 2, FilledSize/(2^(0 + 2))}];
```

```
SingularValueDecomposition[N[WWW]][[2]] // MatrixForm
```

$$\begin{pmatrix} 6.24656 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 2.51121 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 1.46362 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0.979701 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

```
W0 = Table[Flatten[FilledVec[H1Avec]][[((FilledSize/2) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}];

For[b = 0, b < npow, b++;
 Do[WK[[xodd[i], yodd[j]]] = WP[i, j], {j, 1, FilledSize/(2^(0 + 2)), 2},
  {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}],
 Do[WK[[xeven[i], yeven[j]]] = WP[i, j], {j, 2, FilledSize/(2^(0 + 2)), 2},
  {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}];
 Wb+1 =
  Table[
    WK[[i, j]]
    , {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]},
    {j, 2, FilledSize/(2^(b + 2))}] // MatrixForm]
```

Part::partw : Part 2 of SingularValueDecomposition[(≪1≫)] does not exist. ≫

Do::iterb : Iterator {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]} does not have appropriate bounds. ≫

Part::partw : Part 2 of SingularValueDecomposition[(≪1≫)] does not exist. ≫

Do::iterb : Iterator {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]} does not have appropriate bounds. ≫

Part::partw : Part 2 of SingularValueDecomposition[(≪1≫)] does not exist. ≫

General::stop : Further output of Part::partw will be suppressed during this calculation. ≫

Do::iterb : Iterator {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]} does not have appropriate bounds. ≫

General::stop : Further output of Do::iterb will be suppressed during this calculation. ≫

Table::iterb : Iterator {i, 1, 2 MatrixRank[√SingularValueDecomposition[(≪4≫)][[2]]]} does not have appropriate bounds. ≫

Table::iterb : Iterator {i, 1, 2 MatrixRank[√SingularValueDecomposition[(≪4≫)][[2]]]} does not have appropriate bounds. ≫

Table::iterb : Iterator {i, 1, 2 MatrixRank[√SingularValueDecomposition[(≪4≫)][[2]]]} does not have appropriate bounds. ≫

General::stop : Further output of Table::iterb will be suppressed during this calculation. ≫

```
Length[W1[[1, All]]]
Length[W1[[All, 1]]]
```

256

2

```
Do[Print[Wi // MatrixForm], {i, 0, 9}]
```

```
{{3, 0, 2, 2, 1, 3, 2, 1, 2, 3, 3, 2, 2, 2, 2, 1, 1, 3, 3, 3, 3, 3, 3, 3, 1, 2, 1, 0, 3, 1, 1, 3, 2, 1, 3, 3, 1, 2, 3, 1, 0, 2},
 {2, 1, 3, 0, 2, 1, 0, 0, 0, 0, 0, 2, 0, 2, 1, 2, 3, 1, 0, 0, 2, 0, 1, 3, 1, 1, 2, 0, 0, 0, 0, 0, 2, 2, 1, 3, 0, 0, 0, 0, 0, 2}}

{{-0.361631, 0.540285, -0.413708, -0.423678, -0.205192, -0.632193, -0.423678, -0.211839, -0.423678},
 {-0.0670854, -0.0788769, -0.102742, 0.2079, -0.103144, 0.208302, 0.2079, 0.10395, 0.2079}}

{{-0.127116, 0.219195, -0.164633, -0.16981, -0.0814536, -0.25299, -0.16981, -0.0849052, -0.169
 0.112034, -0.0517999, 0.145977, -0.0677682, 0.108613, -0.0304037, -0.0677682, -0.0338841, -0.0677}

{{-0.081631, 0.188062, -0.121193, -0.118757, -0.0610025, -0.178947, -0.118757, -0.0593784, -0.1187
 0.102943, -0.0758696, 0.162571, -0.0579745, 0.118043, -0.0134464, -0.0579745, -0.0289872, -0.0579}

{{-0.0838193, 0.150195, -0.111196, -0.117428, -0.0545593, -0.174065, -0.117428, -0.058714, -0.11742
 0.147101, -0.079913, 0.193978, -0.0720602, 0.141329, -0.019411, -0.0720602, -0.0360301, -0.07206}

{{-0.0690133, 0.235209, -0.143719, -0.130629, -0.0740409, -0.200307, -0.130629, -0.0653144, -0.1306
 0.139613, -0.135027, 0.258751, -0.0921161, 0.187854, -0.0212184, -0.0921161, -0.0460581, -0.0921}

{{-0.166383, 0.130688, -0.147544, -0.20175, -0.0647378, -0.284557, -0.20175, -0.100875},
 {0.32495, -0.00178618, 0.335858, -0.112375, 0.242635, -0.019152, -0.112375, -0.0561877}}

{{-0.47472, -0.0503484, -0.455681, -0.0713477},
 {0.0804863, -0.163872, -0.21377, 0.325753}}

{{-0.555886, 0.306827},
 {-0.0745188, -0.236039}}

{{-0.685256},
 {0.252184}}
```

```
SingularValueDecomposition[N[W8]][[3]] // MatrixForm
```

$$\begin{pmatrix} -0.858494 & 0.512823 \\ 0.512823 & 0.858494 \end{pmatrix}$$

U Λ and M

```
Ub_ := SingularValueDecomposition[N[Wb]][[1]]
Λb_ := SingularValueDecomposition[N[Wb]][[2]]
(* Mb_:=Ub.√Λb *)

db_ := If[b ≥ 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]], 1]

d0
d2
```

```
1
```

```
2
```

```
Do[Print[Ui // MatrixForm], {i, 0, 9}]
(*Gives U's *)
```

$$\begin{pmatrix} -0.427498 & 0.904016 \\ -0.904016 & -0.427498 \end{pmatrix}$$

$$\begin{pmatrix} 0.961808 & -0.273726 \\ -0.273726 & -0.961808 \end{pmatrix}$$

$$\begin{pmatrix} 0.995077 & -0.0991079 \\ 0.0991079 & 0.995077 \end{pmatrix}$$

$$\begin{pmatrix} 0.998625 & 0.0524203 \\ -0.0524203 & 0.998625 \end{pmatrix}$$

$$\begin{pmatrix} 0.987251 & -0.159168 \\ 0.159168 & 0.987251 \end{pmatrix}$$

$$\begin{pmatrix} 0.869604 & 0.493749 \\ -0.493749 & 0.869604 \end{pmatrix}$$

$$\begin{pmatrix} 0.52662 & -0.850101 \\ -0.850101 & -0.52662 \end{pmatrix}$$

$$\begin{pmatrix} 0.986155 & -0.165828 \\ 0.165828 & 0.986155 \end{pmatrix}$$

$$\begin{pmatrix} 0.99598 & -0.0895765 \\ -0.0895765 & -0.99598 \end{pmatrix}$$

$$\begin{pmatrix} -0.938467 & 0.34537 \\ 0.34537 & 0.938467 \end{pmatrix}$$

```
Do[Print[Λ_i // MatrixForm], {i, 0, 9}]
(*Gives Λ's *)
```

$$\begin{pmatrix} 73.0353 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 29.3743 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 6.71737 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 1.93338 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 1.99801 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \\ 0. & 0.932615 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \end{pmatrix}$$

$$\begin{pmatrix} 1.07523 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \\ 0. & 0.680188 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0.739238 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \\ 0. & 0.579629 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0.600633 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0.5179 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 0.587484 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0.451872 & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 0.669376 & 0. & 0. & 0. \\ 0. & 0.421557 & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 0.637135 & 0. \\ 0. & 0.241825 \end{pmatrix}$$

$$\begin{pmatrix} 0.730187 \\ 0. \end{pmatrix}$$

```
(* Do[Print[M_i//MatrixForm],{i,0,9}] *)
(*Gives M's *)
```

$$\begin{pmatrix} -3.65342 & 4.8996 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ -7.72578 & -2.31696 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 2.4928 & -0.380605 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ -0.70944 & -1.33736 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 1.40655 & -0.0957105 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0.14009 & 0.960965 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 1.03551 & 0.0432329 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \\ -0.0543564 & 0.823601 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0.848828 & -0.12118 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0.136851 & 0.751628 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 0.673948 & 0.355328 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ -0.382658 & 0.625813 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 0.403641 & -0.57145 & 0. & 0. & 0. & 0. & 0. \\ -0.651581 & -0.354001 & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 0.806827 & -0.107668 & 0. & 0. \\ 0.135673 & 0.640285 & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 0.794998 & -0.0440499 \\ -0.0715006 & -0.48978 \end{pmatrix}$$

$$\begin{pmatrix} -0.801929 \\ 0.295122 \end{pmatrix}$$

Think defn of M's are off - M's in notes contract with each other

```
Length[Λ1[[1, All]]]
```

256

Recombining terms

$W_0 = U_0 . \sqrt{\Lambda_1} . \left(\sqrt{\Lambda_1} . V_1 \right) = M_0 . W_1 =$

It's fine that we altered the # of columns of the W's in the main iteration function, now need to find the M's. M's are what we're really recombining.

Test

```
B = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};
B // MatrixForm
```

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$

```
A = {{1, 2, 3, 4}, {5, 6, 7, 8}, {9, 10, 11, 12}, {13, 14, 15, 16}};
A // MatrixForm
```

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{pmatrix}$$

```
SingularValueDecomposition[N[A]][[1]] // MatrixForm
SingularValueDecomposition[N[A]][[2]] // MatrixForm
SingularValueDecomposition[N[A]][[3]] // MatrixForm
```

$$\begin{pmatrix} -0.134722 & 0.825742 & -0.546879 & 0.0303851 \\ -0.340758 & 0.428817 & 0.75182 & 0.367106 \\ -0.546793 & 0.0318923 & 0.136998 & -0.825368 \\ -0.752829 & -0.365033 & -0.341938 & 0.427876 \end{pmatrix}$$

$$\begin{pmatrix} 38.6227 & 0. & 0. & 0. \\ 0. & 2.07132 & 0. & 0. \\ 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} -0.428412 & -0.718653 & -0.163388 & -0.522785 \\ -0.474373 & -0.273808 & 0.607512 & 0.575265 \\ -0.520333 & 0.171038 & -0.724859 & 0.417826 \\ -0.566293 & 0.615884 & 0.280736 & -0.470306 \end{pmatrix}$$

```
AP[i_, j_] := Diagonal[Sqrt[SingularValueDecomposition[N[A]][[2]]]][[i]]
  (SingularValueDecomposition[N[A]][[3]][[i, j]])

Table[C11[[2 (i - 1) + 1 + If[EvenQ[j], 1, 0], ((j - 1) - If[EvenQ[j], 1, 0])/2 + 1]],
  {i, 1, MatrixRank[Sqrt[SingularValueDecomposition[N[A]][[2]]]]}, {j, 1, 4}] // MatrixForm
```

Part::partd : Part specification C11⟦1, 1⟧ is longer than depth of object. ≫

Part::partd : Part specification C11⟦2, 1⟧ is longer than depth of object. ≫

Part::partd : Part specification C11⟦1, 2⟧ is longer than depth of object. ≫

General::stop : Further output of Part::partd will be suppressed during this calculation. ≫

$$\begin{pmatrix} C11⟦1, 1⟧ & C11⟦2, 1⟧ & C11⟦1, 2⟧ & C11⟦2, 2⟧ \\ C11⟦3, 1⟧ & C11⟦4, 1⟧ & C11⟦3, 2⟧ & C11⟦4, 2⟧ \end{pmatrix}$$

```
x[i_, j_] := 2 (i - 1) + 1 + If[EvenQ[j], 1, 0]
y[i_, j_] := ((j - 1) - If[EvenQ[j], 1, 0])/2 + 1

Table[x[i, j], {i, 1, MatrixRank[Sqrt[SingularValueDecomposition[N[A]][[2]]]]}, {j, 1, 4}] //
 MatrixForm
Table[y[i, j], {i, 1, MatrixRank[Sqrt[SingularValueDecomposition[N[A]][[2]]]]}, {j, 1, 4}] //
 MatrixForm
```

$$\begin{pmatrix} 1 & 2 & 1 & 2 \\ 3 & 4 & 3 & 4 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 1 & 2 & 2 \\ 1 & 1 & 2 & 2 \end{pmatrix}$$

```
A1 = (AP[1, 1] | AP[1, 3]
      AP[1, 2] | AP[1, 4]
      AP[2, 1] | AP[2, 3]
      AP[2, 2] | AP[2, 4])
```

{{-2.66246, -1.01541}, {-4.46623, -3.24896}, {-0.682721, 0.874337}, {-0.394067, 0.827926}}

```
AAA = Table[0, {i, 1, MatrixRank[√SingularValueDecomposition[N[A]][[2]]]}, {j, 1, 4}]
```

```
{{0, 0, 0, 0}, {0, 0, 0, 0}}
```

```
AAA[[x[i, j], y[i, j]]] = AP[i, j]
```

```
Part::pkspec1 : The expression i cannot be used as a part specification. >>

Part::pkspec1 : The expression i cannot be used as a part specification. >>

Part::pkspec1 : The expression i cannot be used as a part specification. >>

General::stop : Further output of Part::pkspec1 will be suppressed during this calculation. >>

Set::pkspec1 : The expression 1 + 2 (–1 + i) cannot be used as a part specification. >>
```

```
{6.21471, 1.43921, 0., 0.}[[i]] {{-0.428412, -0.718653, -0.163388, -0.522785},
    {-0.474373, -0.273808, 0.607512, 0.575265}, {-0.520333, 0.171038, -0.724859, 0.417826},
    {-0.566293, 0.615884, 0.280736, -0.470306}}[[i, j]]
```

```
Table[AP[i, j], {i, 1, 2}, {j, 1, 4}] // MatrixForm
```

$$\begin{pmatrix} -2.66246 & -4.46623 & -1.01541 & -3.24896 \\ -0.682721 & -0.394067 & 0.874337 & 0.827926 \end{pmatrix}$$

```
A1 // MatrixForm
```

$$\begin{pmatrix} -2.66246 & -1.01541 \\ -4.46623 & -3.24896 \\ -0.682721 & 0.874337 \\ -0.394067 & 0.827926 \end{pmatrix}$$

```
y[1, 3]
```

```
2
```

```
AK = Table[0, {i, 1, MatrixRank[√SingularValueDecomposition[N[A]][[2]]]}, {j, 1, 4}]
```

```
{{0, 0, 0, 0}, {0, 0, 0, 0}}
```

```
Table[j, {j, 2, 4, 2}]
Table[j, {j, 1, 4, 2}]
```

```
{2, 4}
```

```
{1, 3}
```

```
x[i_, j_] := 2 (i - 1) + 1 + If[EvenQ[j], 1, 0]
y[i_, j_] := ((j - 1) - If[EvenQ[j], 1, 0]) / 2 + 1

xeven[i_] := 2 (i - 1) + 2
yeven[j_] := ((j - 1)) / 2 + 1/2
xodd[i_] := 2 (i - 1) + 1
yodd[j_] := ((j - 1)) / 2 + 1
```

```
Do[AK[[xodd[i], yodd[j]]] = AP[i, j], {i, 1, 2}, {j, 1, 4, 2}]
Do[AK[[xeven[i], yeven[j]]] = AP[i, j], {i, 1, 2}, {j, 2, 4, 2}]
```

Set::partw : Part 3 of {{–2.66246, –1.01541, 0, 0}, {–4.46623, –3.24896, 0, 0}} does not exist. >>

Set::partw : Part 3 of {{–2.66246, –1.01541, 0, 0}, {–4.46623, –3.24896, 0, 0}} does not exist. >>

Set::partw : Part 4 of {{–2.66246, –1.01541, 0, 0}, {–4.46623, –3.24896, 0, 0}} does not exist. >>

Set::partw : Part 4 of {{–2.66246, –1.01541, 0, 0}, {–4.46623, –3.24896, 0, 0}} does not exist. >>

```
Do[AK[[xodd[i], yodd[j]]] = AP[i, j], {j, 1, 4, 2}, {i, 1, 2}]
Do[AK[[xeven[i], yeven[j]]] = AP[i, j], {j, 2, 4, 2}, {i, 1, 2}]
```

```
Flat
```

```
A1 // MatrixForm
AK // MatrixForm
```

$$\begin{pmatrix} -2.66246 & -1.01541 \\ -4.46623 & -3.24896 \\ -0.682721 & 0.874337 \\ -0.394067 & 0.827926 \end{pmatrix}$$

$$\begin{pmatrix} -2.66246 & -1.01541 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -4.46623 & -3.24896 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.682721 & 0.874337 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.394067 & 0.827926 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\text{A1} = \begin{pmatrix} \text{AK[[1, 1]]} & \text{AK[[1, 2]]} \\ \text{AK[[2, 1]]} & \text{AK[[2, 2]]} \\ \text{AK[[3, 1]]} & \text{AK[[3, 2]]} \\ \text{AK[[4, 1]]} & \text{AK[[4, 2]]} \end{pmatrix}$$

```
{{-2.66246, -1.01541, 0, 0, 0, 0, 0, 0, 0, 0}, {-4.46623, -3.24896, 0, 0, 0, 0, 0, 0, 0, 0},
  {-0.682721, 0.874337, 0, 0, 0, 0, 0, 0, 0, 0}, {-0.394067, 0.827926, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0}}[4]
```

```
MatrixRank[AK]
```

2

```
Table[Table[AP[i, j], {i, 1, 2}], {j, 1, 4}] // MatrixForm
```

$$\begin{pmatrix} -2.66246 & -0.682721 \\ -4.46623 & -0.394067 \\ -1.01541 & 0.874337 \\ -3.24896 & 0.827926 \end{pmatrix}$$

```
AK = Table[0, {i, 1, 10}, {j, 1, 10}];
```

```
Do[ReplacePart[AK[[x[i, j], y[i, j]]] → AP[i, j]], {i, 1, 2}, {j, 1, 4}]
```

```
ReplacePart[AK[[xeven[i], yeven[j]]] = AP[i, j], {i, 1, 2}, {j, 2, 4, 2}]
```

```
W0 = Table[Flatten[FilledVec[H1Avec]][[((FilledSize/2) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}];

A1 = (AP[1, 1] | AP[1, 3]
      AP[1, 2] | AP[1, 4]
      AP[2, 1] | AP[2, 3]
      AP[2, 2] | AP[2, 4])

B1 = (BP[1, 1] | BP[1, 3]
      BP[1, 2] | BP[1, 4]
      BP[2, 1] | BP[2, 3]
      BP[2, 2] | BP[2, 4]);

B1[[1]]
```

```
{BP[1, 1], BP[1, 3]}
```

```
A1 // MatrixForm
```

```
 -2.66246  -1.01541
 -4.46623  -3.24896
-0.682721  0.874337
-0.394067  0.827926
```

```
ΩAA[x_, y_] := If[OddQ[x], AP[(x + 1)/2, y - 1], AP[(x)/2, y]]

Table[ΩAA[x, y], {x, 1, 4}, {y, 1, 2}] // MatrixForm
```

```
6.21471 List  -2.66246
  -2.66246    -4.46623
1.43921 List  -0.682721
 -0.682721    -0.394067
```

```
Table[ΩAAB[x, y], {x, 1, 2, 1/3}, {y, 1, 4, 2}] // MatrixForm
```

```
ΩAAB[1, 1]    ΩAAB[1, 3]
ΩAAB[4/3, 1]  ΩAAB[4/3, 3]
ΩAAB[5/3, 1]  ΩAAB[5/3, 3]
ΩAAB[2, 1]    ΩAAB[2, 3]
```

```
WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];

Do[WK[[xodd[i], yodd[j]]] = AP[i, j], {j, 1, 4, 2}, {i, 1, 2}]
Do[WK[[xeven[i], yeven[j]]] = AP[i, j], {j, 2, 4, 2}, {i, 1, 2}]
```

Working Fn

```
WP[i_, j_, b_] := Diagonal[√SingularValueDecomposition[N[Wb]][[2]]][[i]]
   (SingularValueDecomposition[N[Wb]][[3]][[i, j]])

WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];
```

Table::iterb : Iterator {i, 1, FilledSize} does not have appropriate bounds. ≫

```
xeven[i_] := 2 (i - 1) + 2
yeven[j_] := ((j - 1))/2 + 1/2
xodd[i_] := 2 (i - 1) + 1
yodd[j_] := ((j - 1))/2 + 1
```

```
Do[WK[[xodd[i], yodd[j]]] = WP[i, j, 0], {j, 1, FilledSize/(2^(0 + 2)), 2},
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}]
Do[WK[[xeven[i], yeven[j]]] = WP[i, j, 0], {j, 2, FilledSize/(2^(0 + 2)), 2},
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}]
```

```
WWW = Table[WK[[i, j]],
   {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}, {j, 2, FilledSize/(2^(0 + 2)), 2}];
```

```
SingularValueDecomposition[N[WWW]][[2]] // MatrixForm
```

$$\begin{pmatrix} 6.24656 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 2.51121 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 1.46362 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0.979701 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

```
Do[WK[[xodd[i], yodd[j]]] = WP[i, j, 0], {j, 1, FilledSize/(2^(0 + 2)), 2},
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}]
Do[WK[[xeven[i], yeven[j]]] = WP[i, j, 0], {j, 2, FilledSize/(2^(0 + 2)), 2},
  {i, 1, MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]}]
Table[WK[[i, j]], {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[W0]][[2]]]},
   {j, 2, FilledSize/(2^(0 + 2)), 2}];

WP[i_, j_, b_] := Diagonal[√SingularValueDecomposition[N[Wb]][[2]]][[i]]
   (SingularValueDecomposition[N[Wb]][[3]][[i, j]])
WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];
xeven[i_] := 2 (i - 1) + 2
yeven[j_] := ((j - 1))/2 + 1/2
xodd[i_] := 2 (i - 1) + 1
yodd[j_] := ((j - 1))/2 + 1
```

```
For[b = -1, b < npow, b++; Print[b]]
```

```
0
1
2
3
4
5
6
7
8
9
10
```

```
W0 = Table[Flatten[FilledVec[H1Avec]][[((FilledSize/2) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}];
For[b = 0, b < npow, b++, Do[WK[[xodd[i], yodd[j]]] = WP[i, j, b], {j, 1, FilledSize/(2^(b + 2)), 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}] &&
  Do[WK[[xeven[i], yeven[j]]] = WP[i, j, b], {j, 2, FilledSize/(2^(b + 2)), 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}];
 Wb+1 =
  Table[
   WK[[i, j]]
   , {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]},
   {j, 2, FilledSize/(2^(b + 2))}];
 WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}]
]
```

SingularValueDecomposition::matrix : Argument {} at position 1 is not a non-empty rectangular matrix. ≫

Part::partw : Part 2 of SingularValueDecomposition[{}] does not exist. ≫

Table::iterb : Iterator {i, 1, 2 MatrixRank[√SingularValueDecomposition[{}][[2]]]} does not have appropriate bounds. ≫

Part::partw : Part 2 of SingularValueDecomposition[{}] does not exist. ≫

Table::iterb : Iterator {i, 1, 2 MatrixRank[√SingularValueDecomposition[{}][[2]]]} does not have appropriate bounds. ≫

```
SingularValueDecomposition[N[W1]][[2]] // MatrixForm
(* If this matches the WWW found in the subsubsection
 "Second Function" then the iterative function above works *)
```

```
( 6.24656    0.       0.       0.      0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
    0.    2.51121    0.       0.      0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
    0.       0.    1.46362    0.      0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
    0.       0.       0.   0.979701 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. )
```

```
W1 // MatrixForm
```

```
Do[Print[SingularValueDecomposition[N[Wi]][[2]] // MatrixForm], {i, 0, 9}]
```

```
( 73.0353    0.    0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
    0.    29.3743 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. )

( 6.24656    0.       0.       0.      0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0
    0.    2.51121    0.       0.      0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0
    0.       0.    1.46362    0.      0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0
    0.       0.       0.   0.979701 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0

( 2.31149    0.       0.       0.       0.       0.       0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
    0.    1.32508    0.       0.       0.       0.       0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
    0.       0.    1.14276    0.       0.       0.       0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
    0.       0.       0.   0.877645    0.       0.       0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
    0.       0.       0.       0.    0.78702    0.       0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
    0.       0.       0.       0.       0.   0.707891    0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
    0.       0.       0.       0.       0.       0.   0.657807    0.     0. 0. 0. 0. 0. 0. 0. 0.
    0.       0.       0.       0.       0.       0.       0.   0.553629 0. 0. 0. 0. 0. 0. 0. 0. )

( 1.46803    0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
    0.    0.933632     0.        0.        0.        0.        0.        0.        0.        0.        0.
    0.       0.     0.898608     0.        0.        0.        0.        0.        0.        0.        0.
    0.       0.        0.     0.822333     0.        0.        0.        0.        0.        0.        0.
    0.       0.        0.        0.     0.752986     0.        0.        0.        0.        0.        0.
    0.       0.        0.        0.        0.     0.721083     0.        0.        0.        0.        0.
    0.       0.        0.        0.        0.        0.     0.708602     0.        0.        0.        0.
    0.       0.        0.        0.        0.        0.        0.     0.634192     0.        0.        0.
    0.       0.        0.        0.        0.        0.        0.        0.     0.580338     0.        0.
    0.       0.        0.        0.        0.        0.        0.        0.        0.     0.553222     0.
    0.       0.        0.        0.        0.        0.        0.        0.        0.        0.     0.492
    0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
    0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
    0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
    0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
    0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
```

```
1.17767   0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      1.05845   0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.      1.05224   0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.      1.02482   0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.      0.920966  0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.      0.870059  0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.      0.849043  0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.      0.730698  0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.      0.704301  0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.      0.660177  0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.      0.57343
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
```

```
1.36974   0.        0.        0.        0.        0.       0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      1.34529   0.        0.        0.        0.       0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.      1.2919    0.        0.        0.       0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.      1.24902   0.        0.       0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.      1.16052   0.       0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.      1.103    0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.     0.621426 0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
  0.        0.        0.        0.        0.        0.       0.     0. 0. 0. 0. 0. 0. 0. 0.
```

```
1.59886    0.         0.       0.  0.  0.  0.
  0.     1.53847      0.       0.  0.  0.  0.
  0.       0.       0.793555   0.  0.  0.  0.
  0.       0.         0.       0.  0.  0.  0.
  0.       0.         0.       0.  0.  0.  0.
  0.       0.         0.       0.  0.  0.  0.
  0.       0.         0.       0.  0.  0.  0.
  0.       0.         0.       0.  0.  0.  0.
  0.       0.         0.       0.  0.  0.  0.
  0.       0.         0.       0.  0.  0.  0.
  0.       0.         0.       0.  0.  0.  0.
  0.       0.         0.       0.  0.  0.  0.
  0.       0.         0.       0.  0.  0.  0.
  0.       0.         0.       0.  0.  0.  0.

0.891024  0.  0.
   0.     0.  0.
   0.     0.  0.
   0.     0.  0.
   0.     0.  0.
   0.     0.  0.

0.
0.
```

```
SingularValueDecomposition::matrix : Argument {} at position 1 is not a non-empty rectangular matrix. >>

Part::partw : Part 2 of SingularValueDecomposition[{}] does not exist. >>

SingularValueDecomposition[{}][[2]]
```

```
SingularValueDecomposition[N[W8]][[3]] // MatrixForm;
```

Results of H1A SVD

Run this

This contains the gene vector as well as the
 definitions of some terms used in the SVD iteration functions
First define some M = DataVector by putting quaternary rep of base pairs in vector form

```
DataVector;
```

```
M = DataVector;
```

General Data Vector Representation

```
MPorcupine1 = {{0, 0, 0, 3, 3, 3, 1, 3, 2, 0, 1, 0, 0, 0, 1, 2, 3, 3, 0, 1, 0, 2, 2, 2, 3, 2, 1, 3, 2,
     1, 3, 1, 3, 2, 1, 0, 0, 1, 2, 2, 3, 1, 0, 1, 1, 0, 2, 0, 1, 3, 1, 1, 1, 2, 1, 3, 1, 3, 1, 1,
0, 0, 1, 0, 0, 2, 2, 3, 0, 1, 3, 1, 0, 1, 0, 2, 1, 0, 2, 3, 0, 2, 0, 1, 0, 2, 2, 3, 1, 0, 1,
     3, 2, 1, 2, 3, 3, 2, 3, 1, 1, 3, 3, 2, 0, 2, 0, 3, 1, 3, 0, 2, 2, 0, 2, 1, 3, 1, 1, 0,
1, 0, 1, 3, 1, 2, 0, 3, 0, 0, 2, 3, 0, 0, 2, 3, 3, 2, 1, 1, 3, 3, 1, 3, 3, 3, 0, 1, 3, 2, 1,
     0, 2, 3, 0, 3, 3, 1, 3, 3, 3, 0, 3, 3, 1, 3, 2, 1, 3, 2, 2, 3, 1, 3, 2, 3, 3, 1, 1, 3,
3, 3, 1, 2, 1, 3, 3, 3, 1, 3, 1, 2, 0, 3, 2, 3, 2, 2, 1, 0, 2, 1, 2, 2, 2, 1, 0, 1, 1, 0, 0,
     0, 0, 3, 0, 1, 1, 0, 1, 3, 3, 1, 0, 1, 3, 3, 3, 0, 3, 3, 0, 0, 0, 0, 2, 3, 3, 3, 2, 1,
3, 3, 1, 3, 3, 1, 0, 1, 0, 0, 0, 0, 3, 3, 0, 2, 1, 2, 0, 0, 1, 1, 1, 1, 3, 2, 3, 0, 2, 2, 3,
     2, 2, 2, 2, 3, 2, 3, 3, 1, 2, 2, 1, 1, 3, 3, 1, 1, 3, 1, 0, 3, 3, 0, 1, 1, 1, 3, 1, 1,
```

```
3, 1, 2, 1, 1, 0, 0, 1, 0, 0, 3, 0, 0, 0, 0, 3, 0, 0, 3, 1, 0, 0, 0, 3, 0, 2, 2, 2, 0, 2, 0,
     3, 3, 2, 2, 2, 0, 2, 1, 3, 1, 1, 1, 2, 3, 0, 3, 3, 3, 3, 1, 3, 3, 2, 1, 2, 1, 3, 1, 2,
3, 1, 3, 3, 1, 2, 2, 0, 0, 2, 2, 0, 3, 3, 0, 3, 3, 2, 0, 2, 0, 2, 3, 2, 0, 0, 1, 0, 1, 1, 1,
     0, 1, 1, 3, 3, 3, 3, 0, 3, 2, 3, 2, 2, 3, 3, 2, 2, 2, 2, 3, 1, 1, 2, 1, 3, 3, 1, 3, 3,
1, 1, 0, 3, 3, 1, 3, 3, 1, 3, 3, 0, 1, 3, 2, 2, 2, 1, 0, 3, 2, 3, 3, 2, 1, 3, 2, 1, 3, 2, 0,
     2, 2, 3, 2, 1, 3, 2, 1, 1, 2, 0, 2, 2, 3, 2, 1, 3, 2, 1, 1, 2, 1, 3, 2, 1, 1, 2, 0, 0,
2, 3, 2, 1, 2, 1, 3, 2, 2, 3, 0, 0, 3, 0, 1, 3, 3, 0, 1, 0, 2, 1, 2, 1, 0, 1, 3, 3, 1, 3, 3,
     3, 1, 2, 3, 3, 3, 3, 1, 0, 2, 1, 3, 0, 3, 2, 0, 1, 2, 3, 0, 3, 1, 1, 0, 0, 2, 2, 0, 2,
2, 1, 2, 3, 3, 3, 1, 1, 2, 1, 0, 2, 0, 1, 2, 0, 0, 2, 0, 1, 0, 1, 1, 2, 1, 1, 1, 1, 1, 2, 1,
     0, 2, 1, 1, 0, 3, 1, 3, 3, 2, 2, 1, 1, 0, 2, 0, 3, 1, 1, 3, 1, 1, 2, 1, 1, 2, 1, 1, 2,
1, 1, 1, 1, 3, 2, 2, 1, 3, 1, 2, 3, 1, 1, 0, 1, 1, 1, 1, 1, 2, 1, 1, 0, 1, 1, 2, 3, 3, 0, 1,
     1, 2, 1, 3, 2, 2, 0, 2, 0, 0, 2, 2, 0, 0, 0, 0, 3, 2, 2, 1, 0, 3, 1, 3, 3, 1, 0, 0,
1, 0, 1, 1, 1, 2, 1, 1, 3, 1, 3, 1, 1, 1, 2, 1, 0, 1, 1, 0, 3, 1, 2, 2, 3, 3, 0, 3, 0, 1, 3,
     2, 3, 1, 0, 0, 2, 0, 0, 0, 0, 1, 1, 0, 1, 0, 2, 3, 1, 0, 2, 0, 0, 1, 2, 1, 1, 1, 3, 1,
1, 3, 2, 2, 0, 0, 3, 2, 3, 2, 2, 0, 1, 0, 3, 2, 0, 3, 2, 0, 2, 0, 3, 3, 3, 0, 0, 3, 0, 3, 3,
     0, 0, 3, 2, 0, 3, 3, 3, 3, 1, 3, 3, 1, 1, 1, 1, 1, 0, 2, 2, 0, 2, 2, 2, 2, 2, 1, 3, 1,
0, 0, 0, 1, 1, 1, 1, 1, 3, 1, 0, 1, 3, 2, 3, 2, 1, 1, 1, 3, 3, 3, 2, 0, 0, 3, 0, 1, 3, 0, 1,
     0, 2, 0, 0, 3, 0, 0, 2, 2, 0, 0, 2, 2, 3, 3, 0, 0, 2, 2, 3, 3, 2, 0, 0, 3, 3, 1, 3, 2,
2, 1, 1, 1, 3, 2, 1, 3, 1, 1, 1, 1, 0, 0, 3, 1, 0, 1, 1, 1, 0, 2, 2, 2, 3, 2, 0, 1, 0, 2, 2,
     2, 2, 0, 2, 3, 2, 2, 2, 1, 3, 1, 1, 0, 1, 3, 2, 1, 3, 2, 3, 3, 0, 3, 3, 1, 3, 0, 2, 0,
3, 2, 0, 3, 0, 0, 1, 3, 3, 3, 2, 3, 0, 0, 1, 0, 0, 0, 2, 2, 1, 1, 0, 0, 3, 2, 1, 1, 1, 3, 0,
     0, 1, 1, 3, 0, 3, 2, 0, 1, 1, 1, 1, 3, 0, 3, 2, 3, 0, 0, 0, 1, 3, 0, 1, 3, 1, 1, 3, 1,
1, 1, 2, 1, 1, 0, 3, 0, 1, 1, 0, 3, 0, 0, 1, 1, 1, 0, 2, 1, 1, 1, 3, 3, 1, 3, 1, 1, 3, 0, 1,
     1, 0, 1, 3, 1, 1, 1, 2, 2, 3, 0, 1, 3, 3, 3, 0, 1, 1, 1, 1, 2, 0, 0, 0, 1, 1, 3, 2, 3,
1, 1, 3, 3, 2, 0, 3, 2, 2, 2, 0, 1, 0, 0, 3, 1, 2, 0, 3, 3, 0, 1, 3, 3, 1, 1, 0, 0, 1, 1, 1,
     0, 0, 3, 0, 0, 1, 0, 0, 0, 0, 2, 0, 0, 0, 3, 1, 0, 0, 1, 3, 1, 3, 2, 2, 1, 3, 2, 0, 2,
0, 1, 3, 0, 1, 0, 0, 0, 1, 3, 0, 1, 3, 2, 2, 0, 0, 0, 3, 2, 3, 0, 2, 0, 1, 1, 0, 3, 2, 3, 0,
     2, 2, 1, 1, 3, 1, 2, 2, 1, 0, 1, 3, 2, 1, 2, 3, 3, 1, 2, 0, 0, 0, 0, 1, 0, 2, 3, 0, 3,
0, 3, 0, 1, 2, 0, 1, 1, 0, 2, 2, 0, 1, 3, 0, 1, 0, 0, 3, 0, 3, 1, 1, 2, 3, 0, 3, 0, 0, 1, 1,
     0, 3, 2, 3, 0, 3, 2, 3, 0, 1, 0, 0, 3, 3, 1, 0, 2, 0, 2, 0, 0, 3, 3, 3, 0, 0, 3, 1, 3,
3, 0, 0, 0, 2, 0, 1, 1, 1, 1, 1, 1, 0, 1, 3, 3, 0, 0, 1, 1, 1, 3, 0, 0, 2, 3, 2, 0, 0, 3, 0,
     0, 3, 0, 0, 0, 0, 0, 1, 1, 0, 3, 3, 0, 1, 2, 0, 0, 2, 3, 2, 0, 3, 0, 0, 0, 0, 0, 0, 2,
0, 1, 3, 1, 0, 2, 3, 0, 0, 3, 3, 3, 0, 3, 3, 3, 1, 0, 3, 0, 3, 2, 2, 0, 0, 0, 3, 3, 1, 0, 2,
     2, 2, 1, 0, 3, 2, 2, 2, 2, 2, 2, 2, 0, 0, 0, 2, 2, 2, 3, 2, 0, 1, 2, 0, 0, 1, 3, 2, 2,
1, 1, 1, 1, 1, 3, 3, 1, 1, 3, 1, 1, 2, 3, 2, 2, 0, 3, 3, 2, 3, 3, 1, 3, 2, 3, 0, 2, 1, 0, 3,
     3, 1, 3, 3, 1, 1, 0, 0, 0, 0, 3, 0, 1, 1, 0, 0, 2, 2, 0, 0, 2, 3, 0, 0, 3, 1, 1, 3, 1,
1, 2, 0, 3, 0, 2, 0, 2, 0, 2, 1, 3, 3, 1, 3, 0, 1, 0, 2, 1, 3, 0, 2, 2, 0, 1, 0, 2, 1, 0, 2,
     3, 3, 2, 0, 2, 2, 0, 2, 3, 0, 1, 1, 0, 3, 3, 1, 1, 0, 0, 1, 2, 2, 2, 2, 3, 1, 3, 2, 0,
3, 3, 2, 1, 3, 2, 2, 3, 0, 0, 3, 1, 0, 2, 0, 0, 3, 0, 1, 3, 2, 1, 2, 2, 2, 1, 1, 0, 0, 0, 0,
     0, 0, 2, 2, 3, 0, 1, 0, 2, 3, 3, 1, 1, 0, 1, 1, 3, 3, 3, 0, 2, 3, 1, 3, 1, 3, 0, 1, 0,
2, 3, 1, 0, 0, 3, 2, 2, 0, 3, 0, 3, 1, 2, 0, 3, 1, 0, 1, 0, 1, 0, 2, 3, 1, 3, 1, 0, 2, 3, 0,
     2, 0, 3, 1, 0, 3, 1, 1, 1, 0, 1, 2, 2, 1, 0, 2, 1, 1, 0, 0, 1, 1, 0, 3, 0, 0, 0, 0, 2,
3, 1, 0, 3, 1, 0, 0, 3, 0, 0, 1, 0, 0, 1, 1, 0, 1, 3, 3, 1, 3, 3, 1, 0, 1, 1, 0, 3, 2, 2, 3,
     0, 0, 1, 1, 0, 3, 1, 1, 1, 0, 1, 1, 0, 1, 3, 3, 2, 3, 3, 3, 1, 2, 0, 2, 2, 3, 2, 2, 3,
3, 3, 1, 1, 0, 2, 3, 0, 3, 2, 3, 2, 2, 3, 3, 3, 1, 1, 2, 2, 2, 3, 1, 3, 2, 1, 0, 0, 0, 0, 3,
     3, 0, 2, 1, 0, 2, 1, 1, 1, 0, 3, 3, 3, 2, 1, 3, 3, 3, 3, 0, 1, 1, 0, 1, 0, 1, 1, 1, 0,
2, 2, 3, 2, 2, 1, 1, 1, 1, 0, 1, 0, 0, 3, 2, 0, 1, 2, 3, 2, 3, 0, 1, 0, 3, 3, 2, 2, 3, 1, 3,
     3, 1, 1, 0, 0, 3, 1, 0, 1, 2, 1, 3, 3, 1, 3, 2, 1, 0, 3, 3, 3, 3, 1, 1, 1, 2, 1, 3, 1,
0, 1, 3, 3, 3, 1, 0, 0, 0, 0, 2, 3, 3, 1, 0, 2, 1, 1, 0, 2, 1, 1, 1, 2, 1, 2, 2}};

MPorcupine2 =
  {{2, 3, 1, 3, 3, 3, 3, 3, 3, 0, 3, 1, 0, 1, 3, 3, 1, 2, 3, 0, 0, 3, 2, 2, 3, 3, 3, 3, 3, 0, 3, 3,
```

```
0, 3, 3, 1, 0, 1, 3, 3, 0, 2, 2, 2, 3, 3, 0, 0, 2, 3, 2, 2, 2, 2, 2, 2, 3, 1, 3, 3, 3, 0, 0,
2, 0, 3, 3, 0, 0, 0, 3, 3, 1, 3, 1, 3, 2, 0, 0, 3, 3, 2, 3, 0, 1, 0, 3, 0, 1, 0, 3, 2, 2, 3,
3, 0, 3, 0, 1, 2, 2, 0, 3, 0, 3, 3, 2, 3, 0, 2, 3, 1, 1, 3, 2, 2, 3, 1, 2, 3, 0, 3, 0, 3, 0,
1, 3, 2, 3, 3, 3, 3, 1, 2, 0, 0, 1, 2, 1, 0, 2, 3, 2, 1, 1, 2, 0, 2, 2, 1, 1, 3, 0, 1, 0, 3,
2, 2, 3, 1, 3, 0, 1, 0, 3, 3, 3, 1, 1, 0, 2, 3, 0, 2, 3, 3, 3, 2, 3, 0, 2, 3, 1, 3, 1, 0, 2,
1, 1, 0, 2, 0, 2, 3, 3, 2, 0, 3, 3, 3, 1, 3, 3, 3, 3, 2, 3, 3, 0, 3, 3, 2, 2, 2, 3, 3, 2, 2,
0, 0, 2, 3, 0, 0, 3, 1, 2, 0, 3, 3, 2, 3, 1, 1, 3, 0, 3, 1, 0, 0, 2, 2, 0, 1, 0, 2, 2, 3, 3,
3, 1, 2, 2, 2, 2, 3, 0, 0, 0, 2, 3, 0, 1, 1, 2, 2, 2, 0, 2, 3, 2, 2, 3, 0, 2, 2, 0, 2, 0, 0,
2, 2, 2, 1, 3, 2, 2, 2, 3, 3, 0, 3, 2, 2, 3, 0, 3, 2, 2, 1, 2, 2, 2, 0, 2, 2, 0, 2, 3, 0, 2,
3, 3, 3, 0, 1, 0, 3, 0, 2, 2, 2, 2, 3, 1, 0, 3, 0, 2, 2, 3, 3, 0, 2, 2, 2, 1, 0, 3, 3, 2, 2,
1, 1, 3, 3, 3, 2, 3, 3, 0, 1, 0, 0, 0, 2, 3, 3, 0, 3, 1, 0, 3, 1, 3, 0, 2, 0, 0, 3, 0, 0, 1,
0, 2, 1, 0, 2, 3, 2, 2, 0, 2, 1, 1, 1, 0, 1, 3, 1, 1, 1, 1, 3, 2, 3, 1, 0, 1, 1, 1, 3, 2, 2,
2, 3, 2, 0, 3, 3, 2, 2, 2, 2, 0, 2, 1, 0, 2, 2, 2, 1, 1, 0, 2, 0, 0, 3, 3, 1, 0, 0, 1, 1, 3,
3, 0, 0, 1, 1, 3, 3, 1, 1, 3, 3, 0, 3, 3, 1, 3, 2, 3, 0, 2, 3, 0, 3, 3, 1, 0, 0, 0, 2, 2, 2,
1, 0, 1, 0, 2, 3, 2, 0, 2, 2, 2, 2, 2, 3, 3, 3, 2, 0, 2, 1, 1, 1, 1, 1, 3, 1, 1, 3, 2, 2, 2,
2, 2, 0, 0, 2, 0, 0, 0, 0, 3, 1, 0, 3, 3, 0, 0, 3, 0, 3, 3, 0, 0, 0, 3, 1, 3, 1, 0, 3, 1, 0,
3, 2, 3, 1, 1, 0, 1, 0, 3, 3, 1, 1, 0, 2, 2, 0, 2, 2, 2, 1, 2, 3, 3, 1, 3, 2, 0, 1, 3, 2, 3,
2, 2, 3, 3, 3, 3, 1, 3, 3, 2, 0, 1, 0, 2, 3, 0, 3, 0, 0, 1, 1, 2, 0, 3, 2, 2, 3, 2, 1, 2, 2,
2, 0, 2, 0, 2, 2, 1, 2, 2, 2, 3, 2, 3, 3, 2, 0, 0, 2, 0, 3, 2, 1, 1, 0, 3, 3, 3, 3, 3, 1, 1,
3, 3, 1, 3, 1, 1, 0, 2, 1, 2, 2, 3, 0, 0, 1, 2, 2, 3, 2, 2, 1, 2, 2, 2, 2, 2, 3, 2, 2, 0, 1,
2, 0, 2, 1, 1, 0, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 1, 2, 2, 0, 2, 2, 0, 3, 1, 3, 2, 2, 1, 1, 0,
0, 2, 0, 3, 2, 2, 1, 3, 2, 1, 2, 2, 2, 2, 2, 1, 2, 2, 3, 2, 3, 1, 3, 3, 1, 2, 3, 1, 3, 2, 1,
2, 2, 0, 0, 0, 1, 2, 1, 1, 3, 1, 1, 3, 3, 2, 2, 0, 3, 0, 1, 2, 3, 1, 0, 3, 1, 2, 1, 3, 2, 0,
0, 0, 0, 1, 2, 0, 0, 0, 2, 0, 0, 2, 3, 2, 1, 2, 1, 3, 2, 3, 0, 0, 2, 3, 0, 3, 3, 0, 1, 1,
0, 2, 1, 2, 1, 0, 1, 3, 3, 1, 2, 2, 1, 0, 2, 1, 2, 2, 1, 0, 2, 1, 0, 1, 1, 3, 1, 2, 2, 1, 0,
2, 1, 0, 1, 1, 3, 1, 0, 2, 1, 0, 2, 1, 0, 0, 1, 0, 3, 2, 1, 1, 1, 0, 2, 1, 0, 0, 2, 0, 0,
2, 0, 2, 3, 2, 2, 0, 0, 2, 0, 0, 2, 1, 2, 2, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 3, 0, 0, 0,
0, 2, 2, 3, 2, 2, 2, 3, 2, 3, 3, 1, 0, 1, 2, 1, 3, 2, 0, 0, 3, 0, 0, 3, 1, 1, 3, 3, 1, 1,
2, 0, 0, 2, 0, 1, 2, 0, 2, 1, 2, 1, 0, 0, 2, 0, 0, 0, 0, 3, 0, 1, 2, 2, 2, 0, 2, 1, 3, 1,
1, 1, 0, 0, 3, 1, 3, 1, 1, 1, 3, 0, 3, 3, 3, 2, 0, 3, 3, 0, 3, 3, 3, 3, 0, 3, 3, 2, 3, 3,
2, 2, 1, 2, 0, 2, 2, 0, 0, 2, 2, 3, 0, 0, 3, 2, 0, 2, 2, 0, 2, 2, 2, 1, 1, 2, 0, 0, 1, 0,
1, 1, 2, 1, 0, 1, 1, 3, 0, 1, 0, 2, 2, 2, 2, 3, 3, 1, 2, 1, 3, 0, 0, 3, 3, 3, 3, 2, 3, 2,
0, 0, 2, 0, 0, 2, 1, 0, 0, 0, 1, 3, 3, 3, 3, 0, 0, 3, 0, 0, 0, 2, 3, 2, 0, 0, 2, 3, 2, 2,
3, 0, 3, 3, 3, 3, 2, 2, 3, 2, 1, 1, 1, 2, 1, 3, 2, 1, 1, 0, 1, 0, 3, 1, 2, 0, 2, 0, 0, 0,
2, 1, 2, 0, 0, 0, 2, 2, 0, 0, 1, 0, 2, 0, 3, 1, 0, 2, 1, 0, 2, 0, 0, 3, 0, 0, 0, 2, 0, 0,
3, 0, 3, 3, 2, 1, 0, 2, 3, 0, 0, 0, 2, 0, 0, 2, 2, 1, 0, 0, 1, 3, 3, 0, 1, 3, 2, 0, 3, 0,
2, 0, 0, 3, 2, 3, 2, 2, 0, 2, 1, 3, 1, 1, 3, 0, 2, 0, 3, 1, 3, 1, 0, 0, 2, 2, 0, 1, 0, 0,
1, 2, 2, 0, 2, 3, 2, 0, 1, 1, 3, 1, 3, 1, 3, 0, 1, 3, 2, 1, 3, 2, 3, 2, 0, 2, 3, 0, 1, 1,
3, 3, 2, 3, 3, 2, 2, 0, 2, 0, 2, 1, 2, 2, 2, 0, 2, 3, 1, 3, 2, 2, 3, 2, 0, 1, 1, 2, 3, 3,
2, 1, 0, 2, 0, 2, 1, 0, 2, 1, 0, 1, 1, 1, 3, 2, 3, 0, 0, 1, 2, 3, 3, 3, 2, 3, 1, 0, 2, 0,
0, 0, 3, 3, 3, 1, 1, 2, 1, 2, 2, 2, 1, 3, 2, 2, 1, 3, 2, 0, 0, 1, 3, 3, 3, 3, 2, 0, 0, 0,
2, 3, 2, 0, 2, 1, 2, 2, 2, 0, 0, 0, 0, 3, 2, 1, 0, 2, 0, 0, 2, 1, 2, 3, 2, 0, 3, 3, 2, 2,
0, 0, 2, 0, 1, 2, 0, 0, 3, 2, 3, 0, 1, 0, 1, 2, 3, 1, 0, 3, 3, 2, 3, 2, 2, 2, 2, 1, 1, 0,
1, 1, 3, 2, 2, 2, 3, 2, 3, 2, 2, 1, 0, 0, 0, 0, 2, 1, 0, 0, 0, 3, 2, 2, 2, 1, 3, 2, 1, 3,
0, 0, 3, 3, 3, 3, 2, 1, 0, 2, 0, 1, 1, 1, 2, 2, 0, 0, 0, 1, 1, 0, 1, 0, 3, 0, 1, 3, 2, 2,
0, 0, 0, 1, 1, 0, 1, 1, 3, 0, 2, 0, 0, 0, 1, 0, 0, 2, 3, 2, 2, 3, 2, 2, 2, 0, 3, 2, 2, 3,
3, 0, 1, 1, 0, 3, 2, 2, 3, 2, 0, 0, 2, 0, 0, 2, 3, 2, 2, 3, 3, 2, 3, 3, 0, 3, 3, 2, 0, 3,
2, 0, 1, 3, 3, 3, 3, 0, 3, 2, 2, 1, 3, 2, 2, 1, 3, 2, 1, 1, 2, 3, 2, 2, 2, 0, 3, 2, 0, 3,
1, 3, 0, 1, 3, 2, 0, 2, 0, 1, 3, 1, 3, 2, 3, 2, 0, 3, 1, 2, 0, 3, 0, 3, 1, 1, 3, 3, 3, 2,
0, 1, 3, 2, 3, 3, 2, 0, 2, 0, 1, 3, 0, 0, 0, 2, 2, 3, 2, 2, 0, 0, 1, 3, 2, 3, 0, 1, 1, 3,
3, 3, 3, 3, 3, 2, 2, 1, 1, 1, 2, 1, 0, 2, 3, 0, 3, 3, 1, 3, 2, 0, 3, 3, 0, 1, 1, 0, 2, 1,
```

```
    0, 0, 3, 1, 0, 2, 0, 1, 1, 1, 1, 2, 3, 3, 2, 2, 0, 0, 3, 2, 2, 3, 0, 1, 3, 1, 1, 3, 1, 0,
    0, 1, 3, 2, 1, 3, 2, 3, 1, 1, 1, 0, 2, 1, 3, 2, 3, 0, 2, 0, 0, 2, 1, 3, 1, 3, 1, 3, 0, 3,
    1, 2, 2, 0, 2, 2, 0, 3, 3, 0, 1, 3, 3, 1, 1, 3, 3, 2, 2, 3, 0, 3, 3, 3, 3, 2, 2, 0, 0, 2,
    0, 0, 3, 2, 1, 3, 0, 1, 0, 2, 0, 0, 1, 0, 0, 3, 1, 1, 0, 1, 2, 2, 0, 2, 2, 0, 0, 2, 2, 2,
    2, 2, 1, 1, 0, 2, 3, 3, 1, 2, 3, 1, 0, 1, 1, 1, 3, 3, 3, 1, 1, 1, 1, 1, 1, 0, 3, 2, 1,
    1, 1, 3, 2, 0, 0, 3, 3, 3, 1, 1, 0, 3, 0, 3, 2, 0, 0, 0, 3, 0, 0, 0, 3, 3, 0, 1, 3, 2, 0}};
```

Define a 1x781 vector H1Avec which contains all the base pair info of the H1A gene

```
H1Avec =
  {{3, 0, 2, 2, 1, 3, 2, 1, 2, 3, 3, 2, 2, 2, 2, 1, 1, 3, 3, 3, 3, 3, 3, 3, 1, 2, 1, 0, 3, 1, 1, 3, 2,
    1, 3, 3, 1, 2, 3, 1, 0, 2, 2, 3, 3, 3, 0, 3, 0, 1, 1, 0, 1, 3, 3, 3, 0, 3, 3, 3, 2, 2, 3, 2, 3,
    2, 1, 3, 2, 3, 2, 3, 3, 0, 2, 3, 1, 0, 1, 1, 0, 3, 2, 3, 1, 3, 2, 0, 0, 0, 1, 0, 2, 3, 2, 1, 1,
    3, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 1, 1, 2, 1, 3, 3, 1, 3, 2, 1, 3, 2, 1, 3, 1, 1, 3, 2, 0,
    2, 0, 0, 0, 1, 1, 3, 3, 3, 0, 2, 1, 3, 2, 2, 1, 0, 0, 2, 0, 0, 2, 2, 1, 0, 0, 0, 2, 0, 0,
    0, 1, 1, 3, 2, 1, 3, 0, 0, 2, 2, 1, 3, 2, 1, 0, 2, 1, 0, 2, 1, 1, 3, 1, 1, 0, 0, 2, 0, 0,
    0, 0, 0, 0, 1, 1, 1, 2, 1, 3, 2, 2, 1, 1, 1, 3, 3, 1, 1, 2, 3, 2, 3, 1, 0, 2, 0, 2, 1, 3,
    2, 0, 3, 1, 2, 3, 2, 1, 0, 2, 2, 1, 3, 2, 1, 3, 3, 1, 1, 3, 1, 1, 3, 1, 3, 0, 0, 2, 2, 0,
    2, 1, 2, 3, 2, 2, 3, 2, 2, 3, 2, 3, 2, 3, 1, 2, 3, 3, 2, 2, 1, 0, 2, 1, 3, 1, 3, 3, 0, 0,
    0, 0, 0, 2, 2, 1, 2, 1, 3, 2, 2, 1, 2, 2, 1, 1, 2, 1, 0, 2, 2, 1, 3, 0, 1, 2, 0, 1, 2, 3,
    2, 2, 0, 2, 0, 0, 2, 0, 0, 1, 0, 0, 1, 0, 2, 1, 1, 2, 1, 0, 3, 3, 0, 0, 2, 1, 3, 2, 2, 2,
    1, 0, 3, 3, 0, 0, 2, 0, 2, 1, 1, 3, 2, 2, 3, 0, 0, 2, 1, 0, 0, 2, 2, 2, 0, 0, 1, 2, 3, 3,
    2, 2, 3, 2, 1, 0, 2, 0, 1, 0, 0, 0, 2, 2, 2, 3, 0, 1, 1, 2, 2, 0, 2, 1, 1, 3, 1, 2, 2, 2,
    3, 3, 1, 1, 3, 3, 1, 0, 0, 2, 1, 3, 1, 0, 0, 1, 0, 0, 2, 0, 0, 2, 2, 1, 2, 3, 1, 1, 3, 1,
    1, 2, 3, 2, 2, 0, 0, 0, 1, 1, 0, 0, 2, 1, 1, 1, 2, 2, 1, 2, 1, 1, 3, 1, 0, 0, 0, 2, 2, 3,
    2, 2, 1, 3, 0, 1, 0, 0, 0, 0, 0, 1, 3, 0, 0, 2, 2, 1, 0, 0, 1, 2, 2, 2, 3, 2, 1, 0, 3, 1,
    3, 0, 0, 0, 0, 0, 2, 1, 3, 1, 0, 0, 0, 0, 0, 2, 2, 1, 1, 0, 1, 2, 2, 2, 2, 2, 1, 3, 0, 2,
    1, 0, 0, 0, 0, 0, 2, 0, 2, 1, 2, 3, 1, 0, 0, 2, 0, 1, 3, 1, 1, 2, 0, 0, 0, 0, 0, 2, 2, 1,
    3, 0, 0, 0, 0, 0, 2, 1, 1, 3, 2, 1, 2, 2, 1, 0, 0, 1, 0, 0, 2, 2, 0, 0, 0, 3, 1, 1, 3, 1,
    1, 0, 0, 2, 0, 0, 3, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 1, 3, 2, 3, 0, 0, 0,
    2, 1, 1, 1, 0, 0, 2, 0, 0, 0, 2, 3, 0, 2, 1, 3, 0, 0, 0, 0, 2, 1, 1, 1, 3, 2, 1, 3, 0,
    0, 0, 2, 1, 3, 0, 0, 2, 2, 1, 3, 2, 3, 0, 0, 0, 0, 1, 1, 1, 0, 0, 2, 2, 1, 2, 2, 1, 1,
    0, 0, 2, 2, 1, 3, 0, 2, 2, 2, 3, 2, 0, 1, 2, 0, 0, 2, 1, 1, 0, 0, 0, 2, 0, 1, 3, 2, 1,
    1, 0, 0, 0, 1, 1, 1, 0, 0, 2, 0, 0, 0, 2, 1, 2, 2, 1, 0, 1, 1, 1, 0, 0, 2, 0, 0, 0, 0,
    0, 2, 3, 0, 0, 0, 3, 3, 1, 0, 2, 3, 3, 0, 2, 0, 0, 2, 3, 3, 3, 1, 3, 3, 1, 3, 0, 2, 3,
    0, 0, 1, 1, 1, 0, 0, 1, 2, 2, 1, 3, 1, 3, 3, 3, 3, 0, 0, 2, 0, 2, 1, 1, 0, 1, 1, 3, 0}};
```

SVD of H1Avec (Not important except for examples of methods used)

Note: H1Avec vector has 781 digits, didn't put in fillers to get to 2^10 =1024 yet, but seem to be getting good results.

```
PrimeQ[781]
```

```
False
```

```
(*SingularValueDecomposition[N[H1Avec]]*)
```

Very large but note alot of repeating numbers. Some are

0.

-0.0624323

-0.0416215

-0.0208108
-0.001222910
-0.00244583
-0.00366874
-0.000815276

This function only breaks H1Avec down to 3 matrices (can check this fact more quickly using ctrl F on double brackets {{) , not decomposing every number.

So the high degree of repetition here likely implies a high degree of correlation. (Check with Dr. Muccciolo)

```
(*SingularValueDecomposition[N[H1Avec]][[1]]*)
```

```
(*SingularValueDecomposition[N[H1Avec]][[2]] *)
(* double bracket [[]] calls the matrix*)
```

So the first matrix of the SVD only has one term, the second matrix only has 1 non-zero term. Is the 2nd matrix empty just because of the nature of this SVD or because there's a high degree of correlation
So almost all of the relevant info is contained in the the 3rd matrix, so this is the only one that needs to be decomposed further.

Also: want to see how many unique numbers there are in this decomposition - use Gather function on 3rd matrix for this purpose

```
(*SingularValueDecomposition[N[H1Avec]][[1]][[1]]*)
```

Use SingularValueDecomposition[N[H1Avec]][[3]][[k]] to call the 3rd matrix as a list

```
(*Length[SingularValueDecomposition[N[H1Avec]][[3]]]*)
```

Can check that
SingularValueDecomposition[N[H1Avec]][[3]][[781]] exists while
SingularValueDecomposition[N[H1Avec]][[3]][[782]] doesn't

So want to join SingularValueDecomposition[N[H1Avec]][[3]][[k]], k∈{1, 781} into one list and then use gather function

```
(* Gather[SingularValueDecomposition[N[H1Avec]][[3]][[782]]] *)
```

```
(*SingularValueDecomposition[N[H1Avec]][[2]]*)
```

```
(*SingularValueDecomposition[N[H1Avec]][[3]][[1]]*)
```

```
(*SVD3List=Table[Union[SingularValueDecomposition[N[H1Avec]][[3]][[k]]],{k,1,781}];*)
```

Union removes repeated elements from each SingularValueDecomposition[N[H1Avec]][[3]][[k]]
Use Flatten fn to remove extra list brackets

```
(*Union[Flatten[SVD3List]]*)

(*Length[SVD3List]*)

(*H1AfirstSVDuniqueterms=Union[Flatten[SingularValueDecomposition[N[H1Avec]]]]*)

(*Length[H1AfirstSVDuniqueterms]*)
```

Only 19 unique numbers in the SVD of 781 base pair long Histone H1Avec gene!

And this is only the first SVD, need to / can do up to 5, since 1024 = 2^10 = 4^5 > 781 > 2^9 = 4^(4.5) = 512, (4-bit)

```
(*Length[Union[Flatten[SingularValueDecomposition[N[H1Avec]]]]]*)
(* Part we need for 1st SVD of genes below *)
```

Further want to do rest of SVDs
Also change ordering of the base pairs and see what that does to the SVD and the number of unique terms and the values of the unique terms

General Definitional Terms

```
M = H1Avec;

lengthvec[M_] := Length[M[[1, All]]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^(npow)), Break[]];
  FilledSize = 2^(npow + 1)];
Filler[M_] := Table[4, {i, 1, FilledSize - lengthvec[M]}]
FilledVec[M_] := Join[Flatten[M], Filler[M]]

lengthvec[M_] := Length[M[[1, All]]]
Filler[M_] := Table[4, {i, 1, FilledSize - lengthvec[M]}]
FilledVec[M_] := Join[Flatten[M], Filler[M]]

For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^npow), Break[]]];
(* gives npow such that  2^npow > lengthvec[M] > 2^(npow -1) *)
FilledSize = 2^npow;
```

Function

```
FilledSize (*Run this, should be 1024 for M=H1Avec *)
```

1024

```
M // MatrixForm
W0 // MatrixForm
```

(3 0 2 2 1 3 2 1 2 3 3 2 2 2 2 1 1 3 3 3 3 3 3 3 1 2 1 0 3 1 1 3 2 1 3 3 1

$$\begin{pmatrix} 3 & 0 & 2 & 2 & 1 & 3 & 2 & 1 & 2 & 3 & 3 & 2 & 2 & 2 & 2 & 1 & 1 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 1 & 2 & 1 & 0 & 3 & 1 & 1 & 3 & 2 & 1 & 3 & 3 & 1 \\ 2 & 1 & 3 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 2 & 1 & 2 & 3 & 1 & 0 & 0 & 2 & 0 & 1 & 3 & 1 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 1 & 3 & 0 \end{pmatrix}$$

```
W0 = Table[Flatten[FilledVec[M]][[((FilledSize/2) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}];
```

```
W0 = Table[Flatten[FilledVec[M]][[((FilledSize/2) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}];
WP[i_, j_, b_] := Diagonal[√SingularValueDecomposition[N[Wb]][[2]]][[i]]
   (SingularValueDecomposition[N[Wb]][[3]][[i, j]])
WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];
xeven[i_] := 2 (i - 1) + 2
yeven[j_] := ((j - 1))/2 + 1/2
xodd[i_] := 2 (i - 1) + 1
yodd[j_] := ((j - 1))/2 + 1
For[b = 0, b < npow, b++, Do[WK[[xodd[i], yodd[j]]] = WP[i, j, b], {j, 1, FilledSize/(2^(b + 2))}, 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}] &&
  Do[WK[[xeven[i], yeven[j]]] = WP[i, j, b], {j, 2, FilledSize/(2^(b + 2))}, 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}];
 Wb+1 =
  Table[
   WK[[i, j]]
   , {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]},
    {j, 2, FilledSize/(2^(b + 2))}];
 WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];
 If[b > (npow - 3), Break[]]
]
```

;

If[b > (npow - 3), Break[]] used at the end of the For function to prevent the algorithm from going past where needed, which would give an error message

Put in terms of npow so that we don ' t have to change it if we change the datavector M, as npow already includes that change in size

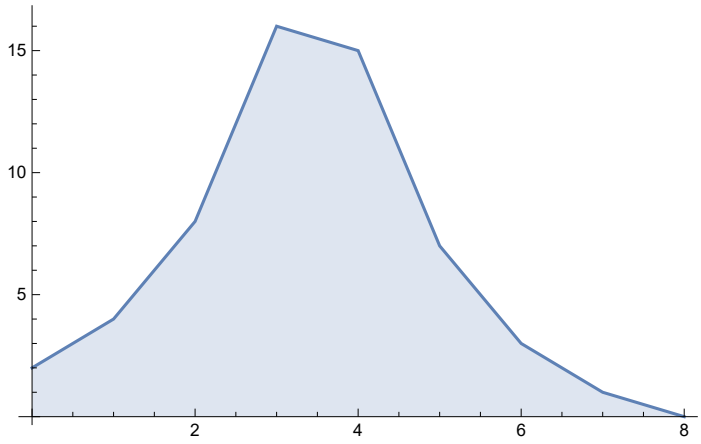
It ' s npow -3 since the algorithm is structured

to go up to $\frac{2\text{^}k}{(2\text{^}(b+2))}$ for a 2^k length filled datavector

```
Do[Print[{i, MatrixRank[SingularValueDecomposition[N[Wi]][[2]]]}], {i, 0, 8}]
Sum[MatrixRank[SingularValueDecomposition[N[Wi]][[2]]], {i, 0, 8}]
```

```
{0, 2}
{1, 4}
{2, 8}
{3, 16}
{4, 15}
{5, 7}
{6, 3}
{7, 1}
{8, 0}
56
```

```
ListLinePlot[
 {{0, 2}, {1, 4}, {2, 8}, {3, 16}, {4, 15}, {5, 7}, {6, 3}, {7, 1}, {8, 0}}, Filling → Axis]
```



This ^ tells us the number of Singular values
 contained in the SVD of W_i given as {i, ⌗ of singular values}
e.g. The SVD of W_4 has 15 singular vaulues
So in total we have 56 singular values for H1A !
And 56 < 2^8 = 256, so decomposes nicely
Increases geometrically for a bit but turns around at the SVD of W_4

Note : the W_i are defined as in step 4 of the notes,
this is why the SVD of W_9 can ' t be defined
The Λ parts of the SVDs are shown below

```
Do[Print[SingularValueDecomposition[N[Wi]][[2]] // MatrixForm], {i, 0, 8}]
```

$$\begin{pmatrix} 73.0353 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 29.3743 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 6.24656 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \\ 0. & 2.51121 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \\ 0. & 0. & 1.46362 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \\ 0. & 0. & 0. & 0.979701 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0 \end{pmatrix}$$

```
2.31149    0.        0.        0.        0.        0.        0.        0.       0.  0.  0.  0.  0.  0.  0.  0.
  0.     1.32508     0.        0.        0.        0.        0.        0.       0.  0.  0.  0.  0.  0.  0.  0.
  0.       0.     1.14276      0.        0.        0.        0.        0.       0.  0.  0.  0.  0.  0.  0.  0.
  0.       0.        0.     0.877645     0.        0.        0.        0.       0.  0.  0.  0.  0.  0.  0.  0.
  0.       0.        0.        0.     0.78702      0.        0.        0.       0.  0.  0.  0.  0.  0.  0.  0.
  0.       0.        0.        0.        0.     0.707891     0.        0.       0.  0.  0.  0.  0.  0.  0.  0.
  0.       0.        0.        0.        0.        0.     0.657807     0.       0.  0.  0.  0.  0.  0.  0.  0.
  0.       0.        0.        0.        0.        0.        0.     0.553629  0.  0.  0.  0.  0.  0.  0.  0.
```

```
1.46803    0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.    0.933632     0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.     0.898608     0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.     0.822333     0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.     0.752986     0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.     0.721083     0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.     0.708602     0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.     0.634192     0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.     0.580338     0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.     0.553222     0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.     0.492
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
```

```
1.17767    0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.    1.05845      0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.     1.05224      0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.     1.02482      0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.     0.920966     0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.     0.870059     0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.     0.849043     0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.     0.730698     0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.     0.704301     0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.     0.660177     0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.     0.57343
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.       0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
```

```
1.36974   0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      1.34529  0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       1.2919   0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       1.24902  0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       1.16052  0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       1.103  0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.621426 0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.     0.       0. 0. 0. 0. 0. 0. 0. 0.
```

```
1.59886   0.        0.       0. 0. 0. 0.
  0.      1.53847   0.       0. 0. 0. 0.
  0.      0.        0.793555 0. 0. 0. 0.
  0.      0.        0.       0. 0. 0. 0.
  0.      0.        0.       0. 0. 0. 0.
  0.      0.        0.       0. 0. 0. 0.
  0.      0.        0.       0. 0. 0. 0.
  0.      0.        0.       0. 0. 0. 0.
  0.      0.        0.       0. 0. 0. 0.
  0.      0.        0.       0. 0. 0. 0.
  0.      0.        0.       0. 0. 0. 0.
  0.      0.        0.       0. 0. 0. 0.
  0.      0.        0.       0. 0. 0. 0.
  0.      0.        0.       0. 0. 0. 0.
```

```
0.891024 0. 0.
   0.    0. 0.
   0.    0. 0.
   0.    0. 0.
   0.    0. 0.
   0.    0. 0.
```

```
0.
0.
```

```
W8
```

```
{{0}, {0}}
```

```
Do[Print[Diagonal[SingularValueDecomposition[N[Wi]][[2]]]], {i, 0, 8}]
(*Gives list of singular values *)
(* Do[Print[Union[Diagonal[SingularValueDecomposition[N[Wi]][[2]]]]],{i,0,8}]
(*Gives list of unique singular values *) *)
```

{94.0849, 40.7435}

{6.72528, 3.22118, 2.34235, 1.55655}

{2.42534, 1.4693, 1.22453, 1.10226, 1.01453, 0.939, 0.811617, 0.573627}

{1.42863, 1.11626, 0.941489, 0.872288, 0.833359, 0.81256, 0.781588, 0.725087,
 0.642506, 0.593684, 0.568707, 0.542864, 0.483272, 0.439268, 0.412184, 0.235364}

{1.15682, 1.0737, 0.990454, 0.898195, 0.873224, 0.810665, 0.774148, 0.737206,
 0.719013, 0.709979, 0.67557, 0.649116, 0.625524, 0.591819, 0.539611, 0.502497,
 0.466543, 0.434379, 0.355667, 0.346345, 0.328543, 0.319671, 0.287973, 0.264292,
 0.188941, 0.164932, 0.129272, 0.0958419, 0.0519779, 0.0217474, 0.0133621, 0.}

{1.26693, 1.19432, 1.14071, 1.11983, 1.07928, 1.04137, 1.03742,
 1.00228, 0.94935, 0.922319, 0.868611, 0.815161, 0.77647, 0.732418,
 0.686985, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.}

{1.48783, 1.46419, 1.42013, 1.39864, 1.32671, 1.29688, 0.884118, 0., 0., 0., 0., 0., 0., 0., 0.}

{1.70407, 1.64312, 0.942201, 0., 0., 0., 0.}

{0.970804, 0., 0.}

Can SVDs of DNA be used to study mutations ?

Some SVD algorithms designed to predict wether patterns and disease outbreaks. Might they be similaly used to study mutability of a given gene ?

Do[Print[SingularValueDecomposition[N[W_i]][[3]] // MatrixForm], {i, 0, 8}]

−0.0423155	0.0632203	−0.0484092	−0.0495758	−0.0240101	−0.0739748	−0.0495758	−0.0
−0.0123778	−0.0145534	−0.0189567	0.0383592	−0.019031	0.0384335	0.0383592	0.01
−0.04884	0.0178911	0.997446	−0.00154054	−0.00144567	−0.00264851	−0.00154054	−0.00
−0.0117066	0.0615515	−0.00148749	0.996302	−0.000375344	−0.00481007	−0.00369791	−0.00
−0.0306089	0.00166884	−0.00145452	−0.000410705	0.999099	−0.000963994	−0.000410705	−0.00
−0.0299377	0.0777738	−0.00258663	−0.00482775	−0.000919792	0.993505	−0.00482775	−0.00
−0.0117066	0.0615515	−0.00148749	−0.00369791	−0.000375344	−0.00481007	0.996302	−0.00
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	0.9
−0.0117066	0.0615515	−0.00148749	−0.00369791	−0.000375344	−0.00481007	−0.00369791	−0.00
−0.0175599	0.0923272	−0.00223124	−0.00554687	−0.000563015	−0.0072151	−0.00554687	−0.00
−0.0175599	0.0923272	−0.00223124	−0.00554687	−0.000563015	−0.0072151	−0.00554687	−0.00
−0.0364622	0.0324446	−0.00219826	−0.00225966	−0.0010889	−0.00336903	−0.00225966	−0.00
−0.0117066	0.0615515	−0.00148749	−0.00369791	−0.000375344	−0.00481007	−0.00369791	−0.00
−0.0364622	0.0324446	−0.00219826	−0.00225966	−0.0010889	−0.00336903	−0.00225966	−0.00
−0.0240844	0.046998	−0.00184288	−0.00297879	−0.000732121	−0.00408955	−0.00297879	−0.00
−0.0306089	0.00166884	−0.00145452	−0.000410705	−0.000901226	−0.000963994	−0.000410705	−0.00
−0.0429867	−0.0128846	−0.0018099	0.000308421	−0.001258	−0.000243475	0.000308421	0.00
−0.0299377	0.0777738	−0.00258663	−0.00482775	−0.000919792	−0.00649458	−0.00482775	−0.00
−0.0175599	0.0923272	−0.00223124	−0.00554687	−0.000563015	−0.0072151	−0.00554687	−0.00
−0.0175599	0.0923272	−0.00223124	−0.00554687	−0.000563015	−0.0072151	−0.00554687	−0.00
−0.0423155	0.0632203	−0.00294201	−0.00410862	−0.00127657	−0.00577406	−0.00410862	−0.00
−0.0175599	0.0923272	−0.00223124	−0.00554687	−0.000563015	−0.0072151	−0.00554687	−0.00
−0.0299377	0.0777738	−0.00258663	−0.00482775	−0.000919792	−0.00649458	−0.00482775	−0.00
−0.0546933	0.0486669	−0.00329739	−0.00338949	−0.00163335	−0.00505354	−0.00338949	−0.00
−0.0182311	0.0162223	−0.00109913	−0.00112983	−0.000544449	−0.00168451	−0.00112983	−0.00
−0.0240844	0.046998	−0.00184288	−0.00297879	−0.000732121	−0.00408955	−0.00297879	−0.00
−0.0306089	0.00166884	−0.00145452	−0.000410705	−0.000901226	−0.000963994	−0.000410705	−0.00
0.	0.	0.	0.	0.	0.	0.	0
−0.0175599	0.0923272	−0.00223124	−0.00554687	−0.000563015	−0.0072151	−0.00554687	−0.00
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.00
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.00
−0.0175599	0.0923272	−0.00223124	−0.00554687	−0.000563015	−0.0072151	−0.00554687	−0.00
−0.0364622	0.0324446	−0.00219826	−0.00225966	−0.0010889	−0.00336903	−0.00225966	−0.00
−0.0306089	0.00166884	−0.00145452	−0.000410705	−0.000901226	−0.000963994	−0.000410705	−0.00
−0.0299377	0.0777738	−0.00258663	−0.00482775	−0.000919792	−0.00649458	−0.00482775	−0.00
−0.0546933	0.0486669	−0.00329739	−0.00338949	−0.00163335	−0.00505354	−0.00338949	−0.00
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.00

-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
0.	0.	0.	0.	0.	0.	0.	0
-0.0364622	0.0324446	-0.00219826	-0.00225966	-0.0010889	-0.00336903	-0.00225966	-0.00
-0.0240844	0.046998	-0.00184288	-0.00297879	-0.000732121	-0.00408955	-0.00297879	-0.00
-0.0299377	0.0777738	-0.00258663	-0.00482775	-0.000919792	-0.00649458	-0.00482775	-0.00
-0.0546933	0.0486669	-0.00329739	-0.00338949	-0.00163335	-0.00505354	-0.00338949	-0.00
-0.0423155	0.0632203	-0.00294201	-0.00410862	-0.00127657	-0.00577406	-0.00410862	-0.00
-0.0123778	-0.0145534	-0.000355384	0.000719126	-0.000356777	0.000720519	0.000719126	0.000
-0.0423155	0.0632203	-0.00294201	-0.00410862	-0.00127657	-0.00577406	-0.00410862	-0.00
-0.0247556	-0.0291069	-0.000710768	0.00143825	-0.000713554	0.00144104	0.00143825	0.000
-0.0182311	0.0162223	-0.00109913	-0.00112983	-0.000544449	-0.00168451	-0.00112983	-0.000
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
0.	0.	0.	0.	0.	0.	0.	0
-0.0182311	0.0162223	-0.00109913	-0.00112983	-0.000544449	-0.00168451	-0.00112983	-0.000
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.0423155	0.0632203	-0.00294201	-0.00410862	-0.00127657	-0.00577406	-0.00410862	-0.00
-0.0247556	-0.0291069	-0.000710768	0.00143825	-0.000713554	0.00144104	0.00143825	0.000
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.04884	0.0178911	-0.00255365	-0.00154054	-0.00144567	-0.00264851	-0.00154054	-0.000
-0.0240844	0.046998	-0.00184288	-0.00297879	-0.000732121	-0.00408955	-0.00297879	-0.00
-0.0299377	0.0777738	-0.00258663	-0.00482775	-0.000919792	-0.00649458	-0.00482775	-0.00
-0.04884	0.0178911	-0.00255365	-0.00154054	-0.00144567	-0.00264851	-0.00154054	-0.000
-0.0299377	0.0777738	-0.00258663	-0.00482775	-0.000919792	-0.00649458	-0.00482775	-0.00
-0.0240844	0.046998	-0.00184288	-0.00297879	-0.000732121	-0.00408955	-0.00297879	-0.00
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.0364622	0.0324446	-0.00219826	-0.00225966	-0.0010889	-0.00336903	-0.00225966	-0.00
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
-0.0546933	0.0486669	-0.00329739	-0.00338949	-0.00163335	-0.00505354	-0.00338949	-0.00
-0.0299377	0.0777738	-0.00258663	-0.00482775	-0.000919792	-0.00649458	-0.00482775	-0.00
-0.0123778	-0.0145534	-0.000355384	0.000719126	-0.000356777	0.000720519	0.000719126	0.000
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
0.	0.	0.	0.	0.	0.	0.	0
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
0.	0.	0.	0.	0.	0.	0.	0
-0.0299377	0.0777738	-0.00258663	-0.00482775	-0.000919792	-0.00649458	-0.00482775	-0.00
-0.0240844	0.046998	-0.00184288	-0.00297879	-0.000732121	-0.00408955	-0.00297879	-0.00
-0.0299377	0.0777738	-0.00258663	-0.00482775	-0.000919792	-0.00649458	-0.00482775	-0.00
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
0.	0.	0.	0.	0.	0.	0.	0
-0.0123778	-0.0145534	-0.000355384	0.000719126	-0.000356777	0.000720519	0.000719126	0.000
-0.0371334	-0.0436603	-0.00106615	0.00215738	-0.00107033	0.00216156	0.00215738	0.001
-0.0306089	0.00166884	-0.00145452	-0.000410705	-0.000901226	-0.000963994	-0.000410705	-0.000
-0.0371334	-0.0436603	-0.00106615	0.00215738	-0.00107033	0.00216156	0.00215738	0.001
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
-0.0306089	0.00166884	-0.00145452	-0.000410705	-0.000901226	-0.000963994	-0.000410705	-0.000
-0.0182311	0.0162223	-0.00109913	-0.00112983	-0.000544449	-0.00168451	-0.00112983	-0.000
-0.0299377	0.0777738	-0.00258663	-0.00482775	-0.000919792	-0.00649458	-0.00482775	-0.00
-0.0182311	0.0162223	-0.00109913	-0.00112983	-0.000544449	-0.00168451	-0.00112983	-0.000
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.0364622	0.0324446	-0.00219826	-0.00225966	-0.0010889	-0.00336903	-0.00225966	-0.00
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000

-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.0306089	0.00166884	-0.00145452	-0.000410705	-0.000901226	-0.000963994	-0.000410705	-0.000
-0.0429867	-0.0128846	-0.0018099	0.000308421	-0.001258	-0.000243475	0.000308421	0.000
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
-0.0306089	0.00166884	-0.00145452	-0.000410705	-0.000901226	-0.000963994	-0.000410705	-0.000
-0.0182311	0.0162223	-0.00109913	-0.00112983	-0.000544449	-0.00168451	-0.00112983	-0.000
-0.04884	0.0178911	-0.00255365	-0.00154054	-0.00144567	-0.00264851	-0.00154054	-0.000
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.0423155	0.0632203	-0.00294201	-0.00410862	-0.00127657	-0.00577406	-0.00410862	-0.00
-0.0240844	0.046998	-0.00184288	-0.00297879	-0.000732121	-0.00408955	-0.00297879	-0.00
-0.0182311	0.0162223	-0.00109913	-0.00112983	-0.000544449	-0.00168451	-0.00112983	-0.000
-0.0299377	0.0777738	-0.00258663	-0.00482775	-0.000919792	-0.00649458	-0.00482775	-0.00
-0.04884	0.0178911	-0.00255365	-0.00154054	-0.00144567	-0.00264851	-0.00154054	-0.000
-0.0306089	0.00166884	-0.00145452	-0.000410705	-0.000901226	-0.000963994	-0.000410705	-0.000
-0.0299377	0.0777738	-0.00258663	-0.00482775	-0.000919792	-0.00649458	-0.00482775	-0.00
-0.0429867	-0.0128846	-0.0018099	0.000308421	-0.001258	-0.000243475	0.000308421	0.000
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
-0.0247556	-0.0291069	-0.000710768	0.00143825	-0.000713554	0.00144104	0.00143825	0.000
-0.0240844	0.046998	-0.00184288	-0.00297879	-0.000732121	-0.00408955	-0.00297879	-0.00
-0.0371334	-0.0436603	-0.00106615	0.00215738	-0.00107033	0.00216156	0.00215738	0.001
0.	0.	0.	0.	0.	0.	0.	0
0.	0.	0.	0.	0.	0.	0.	0
-0.0306089	0.00166884	-0.00145452	-0.000410705	-0.000901226	-0.000963994	-0.000410705	-0.000
-0.0306089	0.00166884	-0.00145452	-0.000410705	-0.000901226	-0.000963994	-0.000410705	-0.000
-0.0299377	0.0777738	-0.00258663	-0.00482775	-0.000919792	-0.00649458	-0.00482775	-0.00
-0.0546933	0.0486669	-0.00329739	-0.00338949	-0.00163335	-0.00505354	-0.00338949	-0.00
-0.0423155	0.0632203	-0.00294201	-0.00410862	-0.00127657	-0.00577406	-0.00410862	-0.00
-0.0371334	-0.0436603	-0.00106615	0.00215738	-0.00107033	0.00216156	0.00215738	0.001
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
-0.0240844	0.046998	-0.00184288	-0.00297879	-0.000732121	-0.00408955	-0.00297879	-0.00
-0.0182311	0.0162223	-0.00109913	-0.00112983	-0.000544449	-0.00168451	-0.00112983	-0.000
-0.0123778	-0.0145534	-0.000355384	0.000719126	-0.000356777	0.000720519	0.000719126	0.000
0.	0.	0.	0.	0.	0.	0.	0
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
-0.0247556	-0.0291069	-0.000710768	0.00143825	-0.000713554	0.00144104	0.00143825	0.000
-0.0247556	-0.0291069	-0.000710768	0.00143825	-0.000713554	0.00144104	0.00143825	0.000
-0.0240844	0.046998	-0.00184288	-0.00297879	-0.000732121	-0.00408955	-0.00297879	-0.00
-0.0364622	0.0324446	-0.00219826	-0.00225966	-0.0010889	-0.00336903	-0.00225966	-0.00
-0.0306089	0.00166884	-0.00145452	-0.000410705	-0.000901226	-0.000963994	-0.000410705	-0.000
-0.0123778	-0.0145534	-0.000355384	0.000719126	-0.000356777	0.000720519	0.000719126	0.000
-0.0123778	-0.0145534	-0.000355384	0.000719126	-0.000356777	0.000720519	0.000719126	0.000
0.	0.	0.	0.	0.	0.	0.	0
-0.0117066	0.0615515	-0.00148749	-0.00369791	-0.000375344	-0.00481007	-0.00369791	-0.00
-0.0247556	-0.0291069	-0.000710768	0.00143825	-0.000713554	0.00144104	0.00143825	0.000
-0.0247556	-0.0291069	-0.000710768	0.00143825	-0.000713554	0.00144104	0.00143825	0.000
-0.0123778	-0.0145534	-0.000355384	0.000719126	-0.000356777	0.000720519	0.000719126	0.000
-0.0429867	-0.0128846	-0.0018099	0.000308421	-0.001258	-0.000243475	0.000308421	0.000
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.0423155	0.0632203	-0.00294201	-0.00410862	-0.00127657	-0.00577406	-0.00410862	-0.00
-0.0364622	0.0324446	-0.00219826	-0.00225966	-0.0010889	-0.00336903	-0.00225966	-0.00
-0.0306089	0.00166884	-0.00145452	-0.000410705	-0.000901226	-0.000963994	-0.000410705	-0.000
-0.0546933	0.0486669	-0.00329739	-0.00338949	-0.00163335	-0.00505354	-0.00338949	-0.00
-0.0247556	-0.0291069	-0.000710768	0.00143825	-0.000713554	0.00144104	0.00143825	0.000
0.	0.	0.	0.	0.	0.	0.	0
-0.0240844	0.046998	-0.00184288	-0.00297879	-0.000732121	-0.00408955	-0.00297879	-0.00
-0.0364622	0.0324446	-0.00219826	-0.00225966	-0.0010889	-0.00336903	-0.00225966	-0.00
-0.00585331	0.0307757	-0.000743747	-0.00184896	-0.000187672	-0.00240503	-0.00184896	-0.000
-0.0175599	0.0923272	-0.00223124	-0.00554687	-0.000563015	-0.0072151	-0.00554687	-0.00

−0.0364622	0.0324446	−0.00219826	−0.00225966	−0.0010889	−0.00336903	−0.00225966	−0.00
−0.0182311	0.0162223	−0.00109913	−0.00112983	−0.000544449	−0.00168451	−0.00112983	−0.000
−0.0123778	−0.0145534	−0.000355384	0.000719126	−0.000356777	0.000720519	0.000719126	0.000
−0.0117066	0.0615515	−0.00148749	−0.00369791	−0.000375344	−0.00481007	−0.00369791	−0.00
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.000
0.	0.	0.	0.	0.	0.	0.	0
−0.0364622	0.0324446	−0.00219826	−0.00225966	−0.0010889	−0.00336903	−0.00225966	−0.00
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.000
−0.0182311	0.0162223	−0.00109913	−0.00112983	−0.000544449	−0.00168451	−0.00112983	−0.000
−0.0546933	0.0486669	−0.00329739	−0.00338949	−0.00163335	−0.00505354	−0.00338949	−0.00
−0.0306089	0.00166884	−0.00145452	−0.000410705	−0.000901226	−0.000963994	−0.000410705	−0.000
−0.0182311	0.0162223	−0.00109913	−0.00112983	−0.000544449	−0.00168451	−0.00112983	−0.000
−0.0123778	−0.0145534	−0.000355384	0.000719126	−0.000356777	0.000720519	0.000719126	0.000
0.	0.	0.	0.	0.	0.	0.	0
−0.0117066	0.0615515	−0.00148749	−0.00369791	−0.000375344	−0.00481007	−0.00369791	−0.00
0.	0.	0.	0.	0.	0.	0.	0
−0.0123778	−0.0145534	−0.000355384	0.000719126	−0.000356777	0.000720519	0.000719126	0.000
−0.0123778	−0.0145534	−0.000355384	0.000719126	−0.000356777	0.000720519	0.000719126	0.000
−0.0123778	−0.0145534	−0.000355384	0.000719126	−0.000356777	0.000720519	0.000719126	0.000
0.	0.	0.	0.	0.	0.	0.	0
0.	0.	0.	0.	0.	0.	0.	0
−0.0306089	0.00166884	−0.00145452	−0.000410705	−0.000901226	−0.000963994	−0.000410705	−0.000
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.000
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.000
−0.0117066	0.0615515	−0.00148749	−0.00369791	−0.000375344	−0.00481007	−0.00369791	−0.00
−0.0306089	0.00166884	−0.00145452	−0.000410705	−0.000901226	−0.000963994	−0.000410705	−0.000
−0.0299377	0.0777738	−0.00258663	−0.00482775	−0.000919792	−0.00649458	−0.00482775	−0.00
−0.0364622	0.0324446	−0.00219826	−0.00225966	−0.0010889	−0.00336903	−0.00225966	−0.00
−0.0364622	0.0324446	−0.00219826	−0.00225966	−0.0010889	−0.00336903	−0.00225966	−0.00
−0.0182311	0.0162223	−0.00109913	−0.00112983	−0.000544449	−0.00168451	−0.00112983	−0.000
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.000
−0.0182311	0.0162223	−0.00109913	−0.00112983	−0.000544449	−0.00168451	−0.00112983	−0.000
−0.0299377	0.0777738	−0.00258663	−0.00482775	−0.000919792	−0.00649458	−0.00482775	−0.00
−0.0299377	0.0777738	−0.00258663	−0.00482775	−0.000919792	−0.00649458	−0.00482775	−0.00
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.000
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.000
−0.0364622	0.0324446	−0.00219826	−0.00225966	−0.0010889	−0.00336903	−0.00225966	−0.00
−0.0175599	0.0923272	−0.00223124	−0.00554687	−0.000563015	−0.0072151	−0.00554687	−0.00
−0.0117066	0.0615515	−0.00148749	−0.00369791	−0.000375344	−0.00481007	−0.00369791	−0.00
−0.0175599	0.0923272	−0.00223124	−0.00554687	−0.000563015	−0.0072151	−0.00554687	−0.00
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.000
0.	0.	0.	0.	0.	0.	0.	0
−0.0364622	0.0324446	−0.00219826	−0.00225966	−0.0010889	−0.00336903	−0.00225966	−0.00
−0.0371334	−0.0436603	−0.00106615	0.00215738	−0.00107033	0.00216156	0.00215738	0.001
−0.0117066	0.0615515	−0.00148749	−0.00369791	−0.000375344	−0.00481007	−0.00369791	−0.00
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.000
−0.0175599	0.0923272	−0.00223124	−0.00554687	−0.000563015	−0.0072151	−0.00554687	−0.00
−0.04884	0.0178911	−0.00255365	−0.00154054	−0.00144567	−0.00264851	−0.00154054	−0.000
−0.0371334	−0.0436603	−0.00106615	0.00215738	−0.00107033	0.00216156	0.00215738	0.001
−0.0299377	0.0777738	−0.00258663	−0.00482775	−0.000919792	−0.00649458	−0.00482775	−0.00
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.000
−0.0364622	0.0324446	−0.00219826	−0.00225966	−0.0010889	−0.00336903	−0.00225966	−0.00
−0.0546933	0.0486669	−0.00329739	−0.00338949	−0.00163335	−0.00505354	−0.00338949	−0.00
−0.04884	0.0178911	−0.00255365	−0.00154054	−0.00144567	−0.00264851	−0.00154054	−0.000
−0.00585331	0.0307757	−0.000743747	−0.00184896	−0.000187672	−0.00240503	−0.00184896	−0.000
−0.0247556	−0.0291069	−0.000710768	0.00143825	−0.000713554	0.00144104	0.00143825	0.000
−0.0117066	0.0615515	−0.00148749	−0.00369791	−0.000375344	−0.00481007	−0.00369791	−0.00
−0.0117066	0.0615515	−0.00148749	−0.00369791	−0.000375344	−0.00481007	−0.00369791	−0.00
−0.0306089	0.00166884	−0.00145452	−0.000410705	−0.000901226	−0.000963994	−0.000410705	−0.000
−0.0546933	0.0486669	−0.00329739	−0.00338949	−0.00163335	−0.00505354	−0.00338949	−0.00
−0.04884	0.0178911	−0.00255365	−0.00154054	−0.00144567	−0.00264851	−0.00154054	−0.000
−0.0429867	−0.0128846	−0.0018099	0.000308421	−0.001258	−0.000243475	0.000308421	0.000
−0.0299377	0.0777738	−0.00258663	−0.00482775	−0.000919792	−0.00649458	−0.00482775	−0.00
−0.0546933	0.0486669	−0.00329739	−0.00338949	−0.00163335	−0.00505354	−0.00338949	−0.00
−0.0429867	−0.0128846	−0.0018099	0.000308421	−0.001258	−0.000243475	0.000308421	0.000
−0.0182311	0.0162223	−0.00109913	−0.00112983	−0.000544449	−0.00168451	−0.00112983	−0.000
−0.0546933	0.0486669	−0.00329739	−0.00338949	−0.00163335	−0.00505354	−0.00338949	−0.00

```
 -0.0546933   0.0486669   -0.00329739  -0.00338949  -0.00163335  -0.00505354  -0.00338949  -0.00
 -0.00585331  0.0307757   -0.000743747 -0.00184896  -0.000187672 -0.00240503  -0.00184896  -0.00
 -0.0306089   0.00166884  -0.00145452  -0.000410705 -0.000901226 -0.000963994 -0.000410705 -0.00
 -0.0546933   0.0486669   -0.00329739  -0.00338949  -0.00163335  -0.00505354  -0.00338949  -0.00
 -0.00585331  0.0307757   -0.000743747 -0.00184896  -0.000187672 -0.00240503  -0.00184896  -0.00
 -0.0175599   0.0923272   -0.00223124  -0.00554687  -0.000563015 -0.0072151   -0.00554687  -0.00
 -0.0123778  -0.0145534   -0.000355384  0.000719126 -0.000356777  0.000720519  0.000719126  0.000
 -0.0123778  -0.0145534   -0.000355384  0.000719126 -0.000356777  0.000720519  0.000719126  0.000
 -0.0240844   0.046998    -0.00184288  -0.00297879  -0.000732121 -0.00408955  -0.00297879  -0.00
 -0.0117066   0.0615515   -0.00148749  -0.00369791  -0.000375344 -0.00481007  -0.00369791  -0.00
  0.          0.           0.           0.           0.           0.           0.
 -0.0240844   0.046998    -0.00184288  -0.00297879  -0.000732121 -0.00408955  -0.00297879  -0.00
 -0.0306089   0.00166884  -0.00145452  -0.000410705 -0.000901226 -0.000963994 -0.000410705 -0.00
 -0.0364622   0.0324446   -0.00219826  -0.00225966  -0.0010889   -0.00336903  -0.00225966  -0.00
 -0.0299377   0.0777738   -0.00258663  -0.00482775  -0.000919792 -0.00649458  -0.00482775  -0.00
 -0.04884     0.0178911   -0.00255365  -0.00154054  -0.00144567  -0.00264851  -0.00154054  -0.00
 -0.0240844   0.046998    -0.00184288  -0.00297879  -0.000732121 -0.00408955  -0.00297879  -0.00
 -0.0546933   0.0486669   -0.00329739  -0.00338949  -0.00163335  -0.00505354  -0.00338949  -0.00
 -0.04884     0.0178911   -0.00255365  -0.00154054  -0.00144567  -0.00264851  -0.00154054  -0.00
 -0.04884     0.0178911   -0.00255365  -0.00154054  -0.00144567  -0.00264851  -0.00154054  -0.00
 -0.0546933   0.0486669   -0.00329739  -0.00338949  -0.00163335  -0.00505354  -0.00338949  -0.00
 -0.0117066   0.0615515   -0.00148749  -0.00369791  -0.000375344 -0.00481007  -0.00369791  -0.00
 -0.0175599   0.0923272   -0.00223124  -0.00554687  -0.000563015 -0.0072151   -0.00554687  -0.00
 -0.0364622   0.0324446   -0.00219826  -0.00225966  -0.0010889   -0.00336903  -0.00225966  -0.00
 -0.0175599   0.0923272   -0.00223124  -0.00554687  -0.000563015 -0.0072151   -0.00554687  -0.00
 -0.0306089   0.00166884  -0.00145452  -0.000410705 -0.000901226 -0.000963994 -0.000410705 -0.00
 -0.0240844   0.046998    -0.00184288  -0.00297879  -0.000732121 -0.00408955  -0.00297879  -0.00
 -0.0299377   0.0777738   -0.00258663  -0.00482775  -0.000919792 -0.00649458  -0.00482775  -0.00
 -0.0175599   0.0923272   -0.00223124  -0.00554687  -0.000563015 -0.0072151   -0.00554687  -0.00
 -0.0240844   0.046998    -0.00184288  -0.00297879  -0.000732121 -0.00408955  -0.00297879  -0.00
 -0.0240844   0.046998    -0.00184288  -0.00297879  -0.000732121 -0.00408955  -0.00297879  -0.00
 -0.0429867  -0.0128846   -0.0018099    0.000308421 -0.001258    -0.000243475  0.000308421  0.00
  0.          0.           0.           0.           0.           0.           0.
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
 -0.0553645  -0.0274381   -0.00216528   0.00102755  -0.00161478   0.000477044  0.00102755   0.000
 -0.0670712   0.0341134   -0.00365278  -0.00267037  -0.00199012  -0.00433302  -0.00267037  -0.00
 -0.0553645  -0.0274381   -0.00216528   0.00102755  -0.00161478   0.000477044  0.00102755   0.000
 -0.0670712   0.0341134   -0.00365278  -0.00267037  -0.00199012  -0.00433302  -0.00267037  -0.00
 -0.0670712   0.0341134   -0.00365278  -0.00267037  -0.00199012  -0.00433302  -0.00267037  -0.00
 -0.0495112  -0.0582138   -0.00142154   0.0028765   -0.00142711   0.00288208   0.0028765    0.00
 -0.0495112  -0.0582138   -0.00142154   0.0028765   -0.00142711   0.00288208   0.0028765    0.00
 -0.0495112  -0.0582138   -0.00142154   0.0028765   -0.00142711   0.00288208   0.0028765    0.00
 -0.0495112  -0.0582138   -0.00142154   0.0028765   -0.00142711   0.00288208   0.0028765    0.00
 -0.0495112  -0.0582138   -0.00142154   0.0028765   -0.00142711   0.00288208   0.0028765    0.00
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
 -0.0553645  -0.0274381   -0.00216528   0.00102755  -0.00161478   0.000477044  0.00102755   0.000
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
 -0.0553645  -0.0274381   -0.00216528   0.00102755  -0.00161478   0.000477044  0.00102755   0.000
 -0.0670712   0.0341134   -0.00365278  -0.00267037  -0.00199012  -0.00433302  -0.00267037  -0.00
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
 -0.0553645  -0.0274381   -0.00216528   0.00102755  -0.00161478   0.000477044  0.00102755   0.000
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
 -0.0553645  -0.0274381   -0.00216528   0.00102755  -0.00161478   0.000477044  0.00102755   0.000
 -0.0553645  -0.0274381   -0.00216528   0.00102755  -0.00161478   0.000477044  0.00102755   0.000
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
 -0.0553645  -0.0274381   -0.00216528   0.00102755  -0.00161478   0.000477044  0.00102755   0.000
 -0.0495112  -0.0582138   -0.00142154   0.0028765   -0.00142711   0.00288208   0.0028765    0.00
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
 -0.0553645  -0.0274381   -0.00216528   0.00102755  -0.00161478   0.000477044  0.00102755   0.000
 -0.0670712   0.0341134   -0.00365278  -0.00267037  -0.00199012  -0.00433302  -0.00267037  -0.00
 -0.0495112  -0.0582138   -0.00142154   0.0028765   -0.00142711   0.00288208   0.0028765    0.00
 -0.0553645  -0.0274381   -0.00216528   0.00102755  -0.00161478   0.000477044  0.00102755   0.000
 -0.0612178   0.00333767  -0.00290903  -0.00082141  -0.00180245  -0.00192799  -0.00082141  -0.00
```

−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00

−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.00:
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0670712	0.0341134	−0.00365278	−0.00267037	−0.00199012	−0.00433302	−0.00267037	−0.00
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.00(

```
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0670712   0.0341134  -0.00365278  -0.00267037  -0.00199012  -0.00433302   -0.00267037  -0.00
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0670712   0.0341134  -0.00365278  -0.00267037  -0.00199012  -0.00433302   -0.00267037  -0.00
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0670712   0.0341134  -0.00365278  -0.00267037  -0.00199012  -0.00433302   -0.00267037  -0.00
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0670712   0.0341134  -0.00365278  -0.00267037  -0.00199012  -0.00433302   -0.00267037  -0.00
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0670712   0.0341134  -0.00365278  -0.00267037  -0.00199012  -0.00433302   -0.00267037  -0.00
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0670712   0.0341134  -0.00365278  -0.00267037  -0.00199012  -0.00433302   -0.00267037  -0.00
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0670712   0.0341134  -0.00365278  -0.00267037  -0.00199012  -0.00433302   -0.00267037  -0.00
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0495112  -0.0582138  -0.00142154   0.0028765   -0.00142711   0.00288208    0.0028765    0.001
-0.0612178   0.00333767 -0.00290903  -0.00082141  -0.00180245  -0.00192799   -0.00082141  -0.000
-0.0553645  -0.0274381  -0.00216528   0.00102755  -0.00161478   0.000477044   0.00102755   0.000
-0.0670712   0.0341134  -0.00365278  -0.00267037  -0.00199012  -0.00433302   -0.00267037  -0.00
```

−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.001
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.001
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.001
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.001
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.000
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.000
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0495112	−0.0582138	−0.00142154	0.0028765	−0.00142711	0.00288208	0.0028765	0.001
−0.0553645	−0.0274381	−0.00216528	0.00102755	−0.00161478	0.000477044	0.00102755	0.000
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.000
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.000
−0.0612178	0.00333767	−0.00290903	−0.00082141	−0.00180245	−0.00192799	−0.00082141	−0.000

−0.0955816	0.031852	0.0563488	0.190303	−0.12046	−0.072546	−0.0971755	−(
−0.0995442	0.1284	0.0470801	0.0214115	0.0397151	−0.0213801	0.0456517	−(
−0.075582	−0.0816845	−0.0726628	0.046941	−0.126447	−0.100162	−0.00632614	0
−0.132142	0.0385374	−0.124094	0.0132355	0.0640171	0.0854778	0.0863746	−0
−0.12066	−0.125776	−0.032754	−0.0714892	0.963524	−0.0268277	−0.0220218	0
−0.0970094	−0.0548783	−0.00928037	−0.103386	−0.0279863	0.975361	−0.0166827	0.
−0.0659695	−0.0971808	0.0680109	−0.0331283	−0.0190987	−0.0135574	0.98103	0.
−0.0967841	0.146208	0.0986558	0.0748165	0.0172376	0.00962334	0.00487366	0
−0.155793	−0.0323604	−0.147568	0.0451324	−0.0227288	−0.0133467	−0.00397063	0.(
−0.150273	0.00325668	−0.0444161	0.151942	−0.00312651	0.00429624	−0.000514742	−0
−0.142754	−0.064509	0.0376552	−0.101674	−0.0321783	−0.0281378	−0.023296	−0.
−0.0740245	0.0504803	0.0212199	−0.0151408	−0.00327836	−0.00575892	−0.00259781	−(
−0.0181309	−0.0352807	0.0796779	0.138707	0.0111124	0.0154547	−0.00188323	−0
−0.099233	−0.152582	−0.0961364	0.0788379	−0.0245561	−0.00882538	−0.0120018	0
−0.108491	0.109435	−0.10062	−0.0186614	−0.00574905	−0.00897018	0.00670761	−(
−0.0632094	−0.0793722	0.119587	0.0202767	−0.00929757	−0.00473591	−0.0172423	−0
−0.142754	−0.064509	0.0376552	−0.101674	−0.0321783	−0.0281378	−0.023296	−0.
−0.0802107	0.0493242	−0.0749048	−0.00180863	−0.00666267	−0.00670953	0.00269203	−0
−0.099233	−0.152582	−0.0961364	0.0788379	−0.0245561	−0.00882538	−0.0120018	0
−0.0497073	0.0869171	0.0376668	−0.16718	−0.0110065	−0.020263	−0.00600442	−0
−0.125955	0.0396935	−0.0279692	−0.0000966526	−0.0108546	−0.0102078	−0.00392135	−0
−0.14066	−0.0122396	0.0962576	0.0718731	−0.00615907	−0.00262399	−0.0128224	−0
−0.075227	0.164837	0.0635269	−0.130628	−0.000291713	−0.0137022	−0.000260899	−0
−0.0193335	0.0790756	0.121985	0.0232201	0.0140991	0.00751142	0.00045368	−0
−0.0236511	−0.0708977	−0.0234736	0.0318969	−0.0084899	−0.00218824	−0.00533912	0.
−0.10573	0.127244	−0.0490446	0.0347436	0.00405212	−0.000148722	0.00843555	−0
−0.14894	−0.0656651	−0.0584696	−0.0883419	−0.0355626	−0.0290884	−0.0180062	0.
−0.0793194	0.21595	0.0260047	0.0562517	0.0223432	0.010861	0.0155026	−0
−0.155793	−0.0323604	−0.147568	0.0451324	−0.0227288	−0.0133467	−0.00397063	0.(
−0.0921554	0.0151996	0.100898	0.123566	0.00783408	0.00969574	−0.00448104	−0
−0.114474	−0.12462	0.0633708	−0.0848213	−0.0330919	−0.0258771	−0.0273116	0.
−0.121327	−0.0913152	−0.0257272	0.048653	−0.0202582	−0.0101354	−0.013276	−0.
−0.108491	0.109435	−0.10062	−0.0186614	−0.00574905	−0.00897018	0.00670761	−(
−0.126621	0.0741544	−0.0209425	0.120046	0.00536339	0.00648449	0.00482438	−0
−0.121863	−0.0114198	0.00955303	−0.186976	−0.0334895	−0.034771	−0.0196849	0.
−0.0647669	−0.211537	0.0257039	0.0823585	−0.0220854	−0.00561412	−0.0213072	0
−0.132142	0.0385374	−0.124094	0.0132355	−0.014239	−0.0111584	0.00136849	−0
−0.0236511	−0.0708977	−0.0234736	0.0318969	−0.0084899	−0.00218824	−0.00533912	0.
−0.0519309	−0.0107868	−0.0491892	0.0150441	−0.00757628	−0.00444889	−0.00132354	0.0
−0.0814133	0.16368	−0.0325978	−0.117295	−0.00367602	−0.0146528	0.00502894	−0
−0.125955	0.0396935	−0.0279692	−0.0000966526	−0.0108546	−0.0102078	−0.00392135	−0
−0.108491	0.109435	−0.10062	−0.0186614	−0.00574905	−0.00897018	0.00670761	−(
−0.0473021	−0.141795	−0.0469472	0.0637938	−0.0169798	−0.00437649	−0.0106782	0
0.0130387	−0.0321486	0.185223	−0.146807	−0.00944942	−0.0147911	−0.0193254	−0.
−0.00785232	−0.0852378	0.213325	−0.0615046	−0.00813815	−0.00815787	−0.0229366	−0
−0.132142	0.0385374	−0.124094	0.0132355	−0.014239	−0.0111584	0.00136849	−0
−0.0687296	−0.114989	0.0164352	−0.0865333	−0.0288999	−0.0223788	−0.0206982	0
−0.102304	0.110591	−0.00449564	−0.0319936	−0.00236475	−0.00801957	0.00141777	−0
−0.0491708	0.00702174	0.00238654	0.0684492	0.00222489	0.00437257	0.000404401	−0
−0.0733583	0.0160194	0.0141932	−0.135283	−0.0194964	−0.0224512	−0.0113435	−0.(

−0.0519309	−0.0107868	−0.0491892	0.0150441	−0.00757628	−0.00444889	−0.00132354	0.0
−0.0450785	−0.0440916	0.0399088	−0.11843	−0.02041	−0.0201906	−0.0153591	0.
−0.0719305	0.10275	0.0798224	0.158406	0.0227408	0.0197548	0.00787587	−0
−0.0429846	0.00817788	0.0985113	0.055117	0.00560919	0.00532318	−0.00488544	−0
−0.0673017	−0.028259	0.0820644	0.207156	0.0133373	0.0198272	−0.00147883	−0
−0.155793	−0.0323604	−0.147568	0.0451324	−0.0227288	−0.0133467	−0.00397063	0.0
−0.101638	0.0761303	−0.0115224	−0.152136	−0.0185828	−0.0247119	−0.00732796	−0
−0.0693957	−0.0805284	0.0234619	0.0336089	−0.0126819	−0.00568652	−0.0119525	−0.
−0.114474	−0.12462	0.0633708	−0.0848213	−0.0330919	−0.0258771	−0.0273116	0.
−0.0995442	0.1284	0.0470801	0.0214115	0.00743642	0.000801888	0.00314571	−0
−0.0436507	0.0426388	0.105538	0.175259	0.0218272	0.0220155	0.00386029	−0
−0.127513	−0.0924713	−0.121852	0.0619852	−0.0236425	−0.011086	−0.00798621	0.
−0.0476133	0.139187	0.0962693	0.00636732	0.0150127	0.00525078	0.00446926	−0
0.00828023	0.0534256	0.154727	0.160215	0.0294035	0.0264644	0.00518383	−0
−0.0214274	0.0268062	0.0633824	−0.150327	−0.0119201	−0.0180023	−0.01002	−0.0
−0.0995442	0.1284	0.0470801	0.0214115	0.00743642	0.000801888	0.00314571	−0
−0.14066	−0.0122396	0.0962576	0.0718731	−0.00615907	−0.00262399	−0.0128224	−0
−0.0482794	0.173647	0.103296	0.12651	0.0312307	0.0219431	0.013215	−0
−0.108491	0.109435	−0.10062	−0.0186614	−0.00574905	−0.00897018	0.00670761	−0
−0.100436	−0.038226	−0.0538294	−0.0366489	−0.0215694	−0.0167686	−0.00966487	0.
−0.0174648	−0.0697416	0.0726511	0.0185647	−0.00510559	−0.00123763	−0.010629	−0.
−0.0565597	0.120222	−0.0514312	−0.0337055	0.00182723	−0.00452129	0.00803115	−0
0.0123725	0.00231228	0.192249	−0.0266643	0.00676861	0.00190122	−0.0105797	−0
−0.0942493	−0.0370698	0.0422953	−0.0499811	−0.0181851	−0.015818	−0.0149547	−0
−0.0147047	−0.0519331	0.124227	0.0719698	0.00469558	0.00758382	−0.00890101	−0
0.00276008	0.0178085	0.0515757	0.053405	0.00980117	0.00882146	0.00172794	−0
−0.0404497	−0.1751	0.0421508	−0.0696805	−0.0298135	−0.0201182	−0.0247138	0
0.00894635	0.0189647	0.1477	0.0400729	0.0131855	0.00977207	−0.0035619	−0
−0.0436507	0.0426388	0.105538	0.175259	0.0218272	0.0220155	0.00386029	−0
−0.11514	−0.090159	0.0703975	0.0353209	−0.0168739	−0.0091848	−0.0185659	−0
−0.0926918	0.095095	0.136178	−0.112063	−0.0053973	−0.0149398	−0.0108899	−0
0.00552015	0.0356171	0.103151	0.10681	0.0196023	0.0176429	0.00345589	−0
−0.0942493	−0.0370698	0.0422953	−0.0499811	−0.0181851	−0.015818	−0.0149547	−0
−0.108491	0.109435	−0.10062	−0.0186614	−0.00574905	−0.00897018	0.00670761	−0
−0.0666356	−0.0627199	0.0750377	0.0870139	−0.00288071	0.00313493	−0.0102246	−0
−0.0537996	0.13803	0.000144543	0.0196995	0.0116284	0.00430017	0.0097591	−0
−0.0236511	−0.0708977	−0.0234736	0.0318969	−0.0084899	−0.00218824	−0.00533912	0.
−0.0700618	−0.0460675	0.0304887	0.153751	0.00353616	0.0110058	−0.00320677	−0
−0.0954519	0.0772865	0.0846023	−0.165468	−0.0151985	−0.0237613	−0.0126178	−0
−0.0429846	0.00817788	0.0985113	0.055117	0.00560919	0.00532318	−0.00488544	−0
0.00276008	0.0178085	0.0515757	0.053405	0.00980117	0.00882146	0.00172794	−0
−0.0473021	−0.141795	−0.0469472	0.0637938	−0.0169798	−0.00437649	−0.0106782	0
0.00618627	0.00115614	0.0961247	−0.0133322	0.00338431	0.00095061	−0.00528984	−0
0.00276008	0.0178085	0.0515757	0.053405	0.00980117	0.00882146	0.00172794	−0
−0.0214274	0.0268062	0.0633824	−0.150327	−0.0119201	−0.0180023	−0.01002	−0.0
−0.0519309	−0.0107868	−0.0491892	0.0150441	−0.00757628	−0.00444889	−0.00132354	0.0
−0.0687296	−0.114989	0.0164352	−0.0865333	−0.0288999	−0.0223788	−0.0206982	0
−0.1179	−0.107968	0.0188218	−0.0180842	−0.026675	−0.0180063	−0.0202938	0.
−0.0666356	−0.0627199	0.0750377	0.0870139	−0.00288071	0.00313493	−0.0102246	−0
−0.0485047	−0.0274392	−0.00464019	−0.051693	−0.0139931	−0.0123197	−0.00834133	0.
−0.149606	−0.0312042	−0.0514428	0.0318002	−0.0193445	−0.0123961	−0.00926047	−0
−0.0519309	−0.0107868	−0.0491892	0.0150441	−0.00757628	−0.00444889	−0.00132354	0.0
−0.126621	0.0741544	−0.0209425	0.120046	0.00536339	0.00648449	0.00482438	−0
−0.132142	0.0385374	−0.124094	0.0132355	−0.014239	−0.0111584	0.00136849	−0
−0.0236511	−0.0708977	−0.0234736	0.0318969	−0.0084899	−0.00218824	−0.00533912	0.
−0.0315034	−0.156136	0.189851	−0.0296077	−0.016628	−0.0103461	−0.0282757	0
−0.075582	−0.0816845	−0.0726628	0.046941	−0.0160662	−0.00663713	−0.00666266	0.
−0.117234	−0.142428	0.011795	−0.138226	−0.0428931	−0.0346986	−0.0290396	0
−0.0731331	0.217106	0.122129	0.0429196	0.0257275	0.0118116	0.0102128	−0
−0.0733583	0.0160194	0.0141932	−0.135283	−0.0194964	−0.0224512	−0.0113435	−0.0
−0.108954	−0.0890029	0.166522	0.0219887	−0.0134895	−0.00823419	−0.0238557	−0
−0.0167987	−0.104203	0.0656244	−0.101577	−0.0213236	−0.0179299	−0.0193747	0
−0.103862	−0.0215736	−0.0983784	0.0300883	−0.0151526	−0.00889778	−0.00264709	0.0
−0.0926918	0.095095	0.136178	−0.112063	−0.0053973	−0.0149398	−0.0108899	−0
−0.0570231	−0.0782161	0.215711	0.00694456	−0.00591326	−0.0037853	−0.0225322	−0
−0.142754	−0.064509	0.0376552	−0.101674	−0.0321783	−0.0281378	−0.023296	−0.
−0.0402245	0.0259864	0.150087	0.108522	0.0154104	0.0141446	−0.00315749	−0
−0.0909528	−0.0991567	0.0585908	0.239053	0.00484743	0.017639	−0.00681795	−0

−0.0926918	0.095095	0.136178	−0.112063	−0.0053973	−0.0149398	−0.0108899	−0
−0.108491	0.109435	−0.10062	−0.0186614	−0.00574905	−0.00897018	0.00670761	−(
0.00618627	0.00115614	0.0961247	−0.0133322	0.00338431	0.00095061	−0.00528984	−0
−0.101102	−0.00376505	−0.0468026	0.0834933	−0.0053514	−0.0000763221	−0.000919143	−0
−0.0531335	0.10357	−0.00688218	−0.100443	−0.00458963	−0.0123921	0.00101337	−0
−0.0678382	0.0516364	0.117345	−0.0284729	0.000105944	−0.00480831	−0.00788765	−0
−0.114474	−0.12462	0.0633708	−0.0848213	−0.0330919	−0.0258771	−0.0273116	0.
−0.119103	0.00638874	0.0611288	−0.133571	−0.0236884	−0.0259495	−0.0179569	−0
−0.085303	−0.0181052	0.189996	−0.00990821	−0.00499965	−0.00604595	−0.0185166	−0
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	

0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.

$$\begin{pmatrix}
-0.192422 & 0.143718 & -0.0558238 & -0.0718138 & -0.0752506 & -0.178894 & -0.134185 & -0.176142 \\
0.146466 & 0.0728795 & -0.0941748 & -0.179021 & 0.106488 & 0.125248 & -0.0724953 & -0.0408429 \\
0.144864 & 0.131688 & 0.0445061 & 0.0124362 & -0.206657 & 0.0260496 & 0.18558 & 0.0980023 \\
0.212505 & -0.0428469 & 0.0209189 & -0.185178 & 0.0611127 & -0.235157 & 0.181272 & -0.0903725 \\
0.143532 & 0.118234 & 0.0478126 & -0.0674535 & -0.0227978 & 0.0950665 & -0.0625228 & -0.137447 \\
0.151966 & -0.175734 & -0.196956 & 0.108819 & 0.0970504 & -0.0625406 & 0.0735335 & -0.0456253 \\
0.104701 & -0.157844 & 0.150565 & -0.170296 & -0.00582031 & -0.0979118 & -0.143707 & 0.0987016 \\
0.134625 & 0.12686 & 0.032311 & -0.0393387 & 0.0798708 & 0.0900659 & -0.0206045 & -0.246766 \\
0.0920199 & 0.292559 & -0.144142 & -0.119505 & 0.0242659 & -0.0821799 & -0.116938 & 0.0938687 \\
0.196702 & -0.0938125 & 0.115135 & -0.090885 & -0.0428966 & -0.0823823 & 0.0239853 & 0.124515 \\
0.0271067 & -0.0276845 & 0.328718 & 0.036985 & -0.0943151 & 0.0626534 & -0.135209 & 0.038625 \\
0.0814579 & 0.12941 & 0.00304205 & 0.00783901 & 0.0367578 & -0.064543 & 0.248049 & 0.10313 \\
0.120223 & 0.250173 & 0.151896 & -0.1087 & -0.0848982 & -0.0607872 & 0.115435 & -0.101318 \\
0.189692 & -0.0179426 & 0.0344782 & -0.217281 & -0.123212 & -0.239278 & 0.049716 & -0.010729 \\
0.205821 & -0.0578691 & -0.0974118 & -0.0967481 & 0.0166102 & 0.151436 & 0.0689167 & -0.149004 \\
0.116607 & -0.0513178 & 0.236878 & -0.0802675 & 0.085991 & -0.173227 & 0.140556 & 0.0268774 \\
0.152759 & -0.262322 & 0.0276915 & -0.225584 & 0.013144 & 0.195209 & -0.0626359 & -0.231327 \\
0.0881876 & -0.0536234 & 0.025799 & -0.158704 & 0.0863047 & -0.154387 & -0.146415 & -0.19857 \\
0.0319352 & 0.238951 & 0.0826549 & 0.00680794 & 0.133284 & -0.0451316 & 0.0190092 & 0.155875 \\
0.106169 & 0.140475 & 0.124373 & 0.0232709 & 0.186156 & -0.0889318 & -0.00634731 & -0.0798585 \\
0.0667578 & 0.121456 & -0.185929 & -0.144241 & -0.17075 & -0.0016989 & -0.209784 & 0.181239 \\
0.121771 & 0.0491334 & -0.0450848 & 0.196093 & 0.0986975 & 0.23642 & 0.20585 & -0.0316911 \\
0.10511 & 0.171572 & 0.0169321 & -0.0625148 & 0.00847071 & 0.113985 & 0.139496 & 0.0430421 \\
0.0695957 & 0.0878687 & -0.014041 & 0.0345336 & 0.110854 & -0.0246286 & -0.13272 & 0.178616 \\
0.053982 & 0.0224185 & -0.0270824 & -0.0894198 & 0.0821918 & 0.0089556 & -0.143697 & 0.123448 \\
0.122766 & -0.0486081 & 0.0475331 & 0.17772 & -0.122909 & -0.203661 & -0.161224 & 0.0787042 \\
0.199888 & -0.0126771 & -0.160887 & 0.201793 & 0.174639 & -0.167951 & 0.0803321 & -0.0175159 \\
0.0930916 & -0.151398 & 0.169082 & -0.136541 & 0.0655409 & 0.109784 & -0.0640315 & -0.0795583 \\
0.160961 & 0.183568 & 0.0258849 & -0.0326228 & -0.116622 & 0.0932285 & 0.100067 & -0.0369337 \\
0.165009 & -0.130982 & -0.103328 & 0.214841 & 0.162774 & -0.126944 & 0.0221334 & -0.0175639 \\
0.0600153 & -0.102023 & 0.208933 & 0.127561 & -0.271603 & -0.102157 & -0.0588142 & -0.00316333 \\
0.0520723 & 0.0444684 & 0.224969 & 0.00808252 & -0.0286008 & 0.159688 & 0.178419 & 0.16536 \\
0.172624 & 0.166815 & 0.0783714 & 0.142057 & -0.272781 & -0.175925 & -0.0269527 & -0.107443 \\
0.0846549 & -0.064115 & 0.144927 & -0.10388 & 0.240454 & -0.0775642 & -0.0917888 & 0.0139904 \\
0.0491408 & 0.182665 & -0.0275074 & 0.113725 & 0.0248735 & 0.0317567 & 0.088054 & 0.0207456 \\
0.0930569 & -0.0651017 & 0.00207031 & 0.160793 & -0.0026667 & 0.210082 & -0.000170058 & 0.196969 \\
0.0721748 & 0.0332745 & -0.0928533 & 0.144176 & -0.176912 & -0.019867 & -0.0211813 & -0.0467079 \\
0.0852408 & 0.0758306 & -0.109722 & -0.114474 & -0.297273 & 0.113989 & 0.0471299 & -0.033141 \\
0.182822 & -0.148236 & -0.0723252 & 0.158612 & -0.0425636 & -0.116943 & -0.0617032 & 0.201074 \\
0.0747613 & -0.0512334 & 0.207321 & 0.085322 & -0.197221 & 0.07137 & -0.125449 & -0.17257 \\
0.107388 & 0.160082 & 0.0607967 & 0.0440219 & 0.133441 & 0.0482353 & 0.0749082 & -0.0872223 \\
0.106343 & 0.19575 & -0.0355644 & 0.140797 & -0.0845652 & -0.0953475 & 0.0297466 & -0.0247596 \\
0.120598 & -0.118962 & -0.073403 & -0.0816759 & -0.112035 & -0.0734614 & 0.217475 & 0.0101835 \\
0.0886338 & -0.0737397 & 0.221588 & -0.0244568 & -0.103291 & 0.144424 & -0.078707 & -0.0641365 \\
0.0734734 & -0.19622 & -0.0858582 & 0.0491273 & 0.06171 & -0.004379 & 0.0249331 & -0.0325026 \\
0.025402 & -0.0207422 & 0.0201867 & 0.0884719 & -0.130597 & 0.0474157 & 0.0198558 & -0.00390998 \\
0.0264464 & -0.0564101 & 0.116548 & -0.00830302 & 0.0874094 & 0.190998 & 0.0650174 & -0.0663727 \\
0.158877 & -0.0784912 & -0.090863 & -0.154755 & 0.0390341 & 0.177193 & 0.0664274 & 0.0402906 \\
0.11725 & 0.0643198 & -0.135437 & 0.0828706 & 0.00794796 & -0.015912 & -0.12296 & -0.0325788 \\
0.135469 & 0.064172 & -0.0359699 & -0.0725794 & 0.0677569 & -0.118976 & -0.113468 & -0.226832 \\
0.151135 & 0.0437459 & 0.0538243 & 0.0931909 & 0.225285 & -0.212434 & -0.0348239 & -0.0585586 \\
0.106458 & -0.149013 & -0.134351 & -0.161518 & -0.162909 & 0.0322023 & -0.0987069 & 0.215934 \\
0.144487 & -0.0444714 & -0.158175 & -0.167108 & 0.168041 & 0.0275024 & -0.103461 & 0.169751 \\
0.0575929 & -0.00689113 & 0.213521 & 0.234846 & 0.171361 & -0.00242512 & -0.329947 & 0.151213 \\
0.152797 & 0.0421946 & -0.177029 & 0.081269 & -0.0785195 & 0.13634 & -0.330749 & -0.135513 \\
0.110057 & 0.156034 & 0.070105 & -0.131614 & -0.0283138 & -0.0905268 & -0.0895809 & 0.279115 \\
0.196553 & -0.00136302 & -0.112147 & 0.147311 & 0.0200403 & 0.2241 & -0.079001 & 0.15823 \\
0.200797 & -0.25764 & -0.0473892 & 0.157958 & -0.196854 & -0.103949 & 0.121226 & -0.0278429 \\
0.0842011 & 0.0854395 & 0.228993 & 0.165913 & 0.171646 & 0.131187 & -0.0158121 & -0.0936796 \\
0.0888412 & -0.0878046 & 0.0240713 & 0.0964476 & 0.0327147 & 0.0656579 & 0.199131 & 0.0554405 \\
0.0449008 & -0.0284023 & 0.245389 & -0.0576829 & -0.0217918 & 0.0601783 & -0.0449684 & 0.208636 \\
0.169635 & 0.128613 & -0.048622 & -0.0803623 & 0.0168285 & 0.203013 & -0.0845582 & 0.0891734 \\
0.0981046 & 0.0639472 & -0.0350987 & 0.180359 & -0.080229 & 0.0929112 & -0.194322 & -0.23706 \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 0.
\end{pmatrix}$$

0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.

-0.0942285	-0.253201	-0.198299	0.182288	0.0546024	-0.132922	-0.127568	0.189254	0.
-0.181559	-0.0881362	0.184628	-0.0377494	0.27459	0.331528	-0.114458	-0.0861457	0.
-0.224156	-0.0244752	-0.178424	0.0594265	-0.1135	0.188861	0.208189	0.19949	-0
0.249211	-0.135689	-0.204183	-0.17008	0.210322	0.168998	0.115071	0.04819	0.
0.121761	0.387765	-0.120189	0.165364	-0.216607	0.0851947	-0.0410128	-0.0927412	-0.
0.15327	-0.00758593	-0.137593	0.189386	0.120942	0.384341	0.0772959	0.178598	0.
0.21655	-0.0890426	0.11547	0.231987	-0.00865568	0.24832	-0.0831539	-0.248302	0.
0.0224538	-0.251718	0.121112	0.491726	-0.310821	0.0787612	-0.186276	-0.176883	0.
0.106024	-0.11909	-0.165893	-0.163532	-0.214963	0.192337	0.136002	0.221231	-0
0.201022	0.0840713	-0.14255	-0.374804	0.0567887	-0.0677363	0.0459708	-0.282794	0.
0.149973	-0.0818353	-0.081727	-0.143846	-0.187554	0.126473	-0.0470619	0.0598681	0.
0.146266	-0.162907	-0.0490315	-0.111885	-0.0547198	-0.0451446	-0.240667	0.0192904	-0.
0.191369	-0.206692	0.236174	0.0934549	0.132132	-0.0236452	0.265518	0.259628	-0
0.230394	-0.0408461	-0.0482782	-0.10268	0.0404343	0.343745	-0.174331	-0.121786	-0.
0.188352	0.37258	0.19103	0.0707437	-0.0726144	-0.0480952	0.31517	-0.205089	-0.
0.18601	-0.121538	-0.0306008	0.121336	0.0220568	0.232023	0.498425	-0.158472	-0
0.223953	0.113939	-0.00195895	-0.283358	0.0738602	-0.0263913	0.0136313	0.104912	-0.
0.105674	0.106422	-0.0641034	0.0982932	0.298196	-0.26259	0.086628	0.0958043	0.
0.203215	-0.139752	0.345457	0.0221619	0.0445583	-0.232963	0.109268	0.118564	-0
0.138237	0.00630763	-0.201801	-0.0235642	-0.0859432	-0.0315694	-0.169995	0.197377	-0.
0.134947	-0.0823414	0.44282	-0.0237277	-0.00709001	0.0375073	0.149126	0.0308504	0.
0.0620695	0.0291015	0.20193	0.0554735	0.03979	-0.195637	-0.14747	0.00338301	-0.
0.131044	0.18388	0.0659001	0.170258	-0.14216	0.0564872	-0.11468	0.180328	0.
0.0947439	-0.293193	-0.174139	0.128988	0.0182998	-0.082028	0.0876477	0.0670289	-0.
0.15418	0.0343961	-0.0301258	0.145013	0.134542	-0.102711	0.00653773	0.4209	0.
0.196158	-0.377901	0.121012	-0.282129	-0.383723	-0.107681	-0.0808201	-0.102095	-0.
0.247796	-0.109881	-0.3555	0.182254	-0.00458231	-0.328424	0.0845222	-0.280883	0.
0.352458	0.0418622	0.0954629	0.0459976	0.327422	0.0395877	-0.42377	0.0050027	-
0.21225	0.271565	-0.051684	0.114618	-0.0582307	0.113728	-0.145885	0.128261	-0
0.172181	0.177602	0.0361105	-0.0177244	-0.424186	-0.0830444	0.0410252	0.252433	0.
0.110719	-0.0234338	-0.220213	0.212019	0.109512	-0.157936	0.075915	-0.202656	-0
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	

```
{{ 0.330949   0.406066   -0.153491   0.201317   -0.151475   0.0623878   -0.0219113   0.617174   -0.
 -0.249023   0.0540639  -0.440871   0.265437   -0.368975   0.181067    0.284528   -0.00164968  -0.
  0.214886  -0.193086   -0.0618505  -0.223125   -0.40388   -0.132836   -0.190463   -0.291028   -0.
 -0.0532902 -0.0348993  0.0869216   0.166302   -0.177737   0.152483    0.588279    0.0663474    0.1
  0.0721687   0.193849   0.439783   0.543556    0.0910337  0.353841   -0.398227   -0.115392   -0.
 -0.0164588  0.398989   -0.134671  -0.305263   -0.478292   0.317106   -0.307866   -0.109587    0.3
 -0.26045   -0.377749   -0.38098    0.342886    0.029163   0.292603   -0.23865    -0.0360166   0.1
 -0.21821    0.414877   -0.0670567 -0.261815    0.236804  -0.0128936   0.0458107   0.100729    -0.
 -0.368828   0.148909    0.411842  -0.0198161  -0.198946  -0.139457    0.0897508   0.0790389    0.3
 -0.405973  -0.0252115   0.0793375 -0.202797   -0.15572    0.116428   -0.196216    0.00686948  -0.
 -0.102588   0.168266   -0.425803  -0.115289    0.473777  -0.00194195  -0.130232    0.0171293    0.1
  0.139084   0.0733614  -0.172068   0.321619   -0.224676  -0.666316   -0.188152   -0.0250559    0.3
  0.500518   0.150779   -0.0991168 -0.0522065  0.0887356   0.267471    0.200958   -0.433062    0.2
 -0.277807   0.42786    -0.0926921  0.271361    0.074654  -0.248064    0.0342041  -0.474119    -0.
 -0.0101608  0.140703    0.0653056  0.0469601   0.00262049  0.0286689   0.288661   -0.27144    -0.0
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.
     0.          0.          0.          0.          0.          0.          0.          0.}}
```

```
{{ 0.656287    -0.36331    0.0346682   -0.422643   -0.225506   0.453354   -0.032903   0. 0. 0. 0. 0.
  0.64247     -0.0275304  -0.0810565   0.161138    0.119873  -0.733997    0.0285854  0. 0. 0. 0. 0.
 -0.136014    -0.790558    0.0249693   0.239437    0.535826   0.0441169  -0.0975108  0. 0. 0. 0. 0.
  0.125549    -0.126382    0.0341542   0.8038     -0.529368   0.201424    0.0142081  0. 0. 0. 0. 0.
 -0.228934    -0.312068    0.696287   -0.220536   -0.412999  -0.379689    0.0449292  0. 0. 0. 0. 0.
 -0.264178    -0.349804   -0.70982    -0.208194   -0.439895  -0.258153   -0.022526   0. 0. 0. 0. 0.
-0.00753662  -0.0808919  -0.0421649  -0.00137619  0.0579816  0.0489293   0.992907   0. 0. 0. 0. 0.
     0.          0.          0.          0.          0.          0.          0.     0. 0. 0. 0. 0.
     0.          0.          0.          0.          0.          0.          0.     0. 0. 0. 0. 0.
     0.          0.          0.          0.          0.          0.          0.     0. 0. 0. 0. 0.
     0.          0.          0.          0.          0.          0.          0.     0. 0. 0. 0. 1.
     0.          0.          0.          0.          0.          0.          0.     0. 0. 0. 1. 0.
     0.          0.          0.          0.          0.          0.          0.     0. 0. 1. 0. 0.
     0.          0.          0.          0.          0.          0.          0.     0. 1. 0. 0. 0.
     0.          0.          0.          0.          0.          0.          0.     1. 0. 0. 0. 0.}}
```

```
{{ 0.400889   0.916061   -0.0109654  0. 0. 0. 0.
 -0.915865   0.401032    0.0190892  0. 0. 0. 0.
 0.0218844  0.00239014   0.999758   0. 0. 0. 0.
    0.          0.          0.      0. 0. 0. 1.
    0.          0.          0.      0. 0. 1. 0.
    0.          0.          0.      0. 1. 0. 0.
    0.          0.          0.      1. 0. 0. 0.}}
```

```
{{ 1. 0. 0.
 0. 0. 1.
 0. 1. 0.}}
```

```
{ 1. }
```

Porcine Circovirus SVD

Comparing DNA Seqs & combined SVDs

PCV Samples - Run this

PCV Type 1 samples

$PCV1_1$ =

```
{{0, 1, 1, 0, 2, 1, 2, 1, 0, 1, 3, 3, 1, 2, 2, 1, 0, 2, 1, 2, 2, 1, 0, 2, 1, 0, 1, 1, 3, 1, 2, 2, 1,
  0, 2, 1, 2, 3, 1, 0, 2, 3, 2, 0, 0, 0, 0, 3, 2, 1, 1, 0, 0, 2, 1, 0, 0, 2, 0, 0, 0, 0, 2, 1, 2, 2,
  1, 1, 1, 2, 1, 0, 0, 1, 1, 1, 1, 0, 3, 0, 0, 2, 0, 2, 2, 3, 2, 2, 2, 3, 2, 3, 3, 1, 0, 1, 1, 1,
  3, 3, 0, 0, 3, 0, 0, 3, 1, 1, 3, 3, 1, 1, 2, 0, 2, 2, 0, 2, 2, 0, 2, 0, 0, 0, 0, 0, 1, 0, 0, 0,
  0, 3, 0, 1, 2, 2, 2, 0, 2, 1, 3, 3, 1, 1, 0, 0, 3, 1, 3, 1, 1, 1, 3, 3, 3, 3, 3, 2, 0, 3, 3,
  0, 3, 3, 3, 3, 2, 3, 3, 3, 2, 1, 2, 2, 0, 2, 0, 2, 2, 0, 0, 2, 2, 3, 3, 3, 2, 2, 0, 0, 2, 0,
  2, 2, 2, 3, 0, 2, 0, 0, 1, 3, 1, 1, 3, 1, 0, 1, 1, 3, 1, 1, 0, 2, 2, 2, 2, 3, 3, 3, 2, 1, 2,
  0, 0, 3, 3, 3, 3, 2, 1, 3, 0, 0, 2, 0, 0, 2, 1, 0, 2, 0, 1, 3, 3, 3, 3, 0, 0, 1, 0, 0, 2, 2,
  3, 2, 0, 0, 2, 3, 2, 2, 3, 0, 3, 3, 3, 3, 2, 2, 3, 2, 1, 1, 1, 2, 1, 3, 2, 1, 1, 0, 1, 0, 3,
  1, 2, 0, 2, 0, 0, 0, 2, 1, 2, 0, 0, 0, 2, 2, 0, 0, 1, 1, 2, 0, 1, 1, 0, 2, 1, 0, 2, 0, 0, 3,
  0, 0, 0, 2, 0, 0, 3, 0, 1, 3, 2, 1, 0, 2, 3, 0, 0, 0, 2, 0, 0, 2, 2, 1, 1, 0, 1, 0, 3, 0, 1,
  3, 3, 0, 3, 1, 2, 0, 2, 3, 2, 3, 2, 2, 0, 2, 1, 3, 1, 1, 2, 1, 2, 2, 0, 0, 1, 1, 0, 2, 2, 2,
  2, 0, 0, 2, 1, 2, 1, 0, 2, 1, 2, 0, 1, 1, 3, 2, 3, 1, 3, 0, 1, 3, 2, 1, 3, 2, 3, 2, 0, 2, 3,
  0, 1, 1, 1, 3, 3, 3, 3, 2, 2, 0, 2, 0, 1, 2, 2, 2, 2, 3, 1, 3, 3, 3, 2, 2, 3, 2, 0, 1, 3, 2,
  3, 0, 2, 1, 1, 2, 0, 2, 1, 0, 2, 3, 3, 1, 1, 1, 3, 2, 3, 0, 0, 1, 2, 3, 0, 3, 2, 3, 2, 0,
  2, 0, 0, 0, 3, 3, 3, 1, 1, 2, 1, 2, 2, 2, 1, 3, 2, 2, 1, 3, 2, 0, 0, 1, 3, 3, 3, 3, 2, 0,
  0, 0, 2, 3, 2, 0, 2, 1, 2, 2, 2, 0, 0, 2, 0, 3, 2, 1, 0, 2, 1, 0, 2, 1, 2, 3, 2, 0, 3, 3,
  2, 2, 0, 0, 2, 0, 1, 0, 2, 1, 3, 2, 3, 0, 1, 0, 1, 2, 3, 1, 0, 3, 0, 2, 3, 2, 2, 2, 1, 1,
  1, 2, 1, 1, 1, 2, 2, 3, 3, 2, 3, 2, 2, 2, 0, 0, 2, 0, 2, 1, 1, 0, 2, 3, 2, 2, 2, 1, 1, 1,
  2, 3, 0, 0, 3, 3, 3, 3, 2, 1, 3, 2, 0, 2, 1, 1, 3, 0, 2, 1, 2, 0, 1, 0, 1, 1, 3, 0, 1, 3,
  2, 2, 0, 0, 2, 1, 1, 3, 0, 2, 3, 0, 2, 0, 0, 0, 3, 0, 0, 2, 3, 2, 2, 3, 2, 2, 2, 0, 3, 2,
  2, 0, 3, 0, 3, 1, 0, 3, 2, 2, 0, 2, 0, 0, 2, 0, 0, 2, 3, 3, 2, 3, 3, 2, 3, 3, 3, 3, 2, 2,
  0, 3, 2, 0, 3, 3, 3, 3, 3, 0, 3, 2, 2, 1, 3, 2, 2, 3, 3, 0, 1, 1, 3, 3, 2, 2, 2, 0, 3, 2,
  0, 3, 1, 3, 0, 1, 3, 2, 0, 2, 0, 1, 3, 2, 3, 2, 3, 2, 0, 1, 1, 2, 2, 3, 0, 3, 1, 1, 0, 3,
  3, 2, 0, 1, 3, 2, 3, 0, 2, 0, 2, 0, 1, 3, 0, 0, 0, 2, 2, 2, 2, 2, 3, 0, 1, 3, 2, 3, 3, 1,
  1, 3, 3, 3, 3, 3, 3, 2, 2, 1, 1, 1, 2, 1, 0, 2, 3, 0, 3, 3, 3, 3, 2, 0, 3, 3, 0, 1, 1, 0,
  2, 1, 0, 0, 3, 1, 0, 2, 2, 1, 1, 1, 1, 1, 1, 0, 2, 2, 0, 0, 3, 2, 2, 3, 0, 1, 3, 1, 1, 3,
  1, 0, 0, 1, 3, 2, 1, 3, 2, 3, 1, 1, 1, 0, 2, 1, 3, 2, 3, 0, 2, 0, 0, 2, 1, 3, 1, 3, 1, 3,
  0, 3, 1, 2, 2, 0, 2, 2, 0, 3, 3, 0, 1, 3, 0, 1, 3, 3, 3, 2, 1, 0, 0, 3, 3, 3, 3, 2, 2, 0,
  0, 2, 0, 1, 3, 2, 1, 3, 2, 2, 0, 2, 0, 0, 1, 0, 0, 3, 1, 1, 0, 1, 2, 2, 0, 2, 2, 3, 0, 1,
  1, 1, 2, 0, 0, 2, 2, 1, 1, 2, 0, 3, 3, 3, 2, 0, 0, 2, 1, 0, 2, 3, 2, 2, 0, 1, 1, 1, 0, 1,
  1, 1, 3, 2, 3, 2, 1, 1, 1, 3, 3, 3, 3, 1, 1, 1, 0, 3, 0, 3, 0, 0, 0, 0, 3, 0, 0, 0, 3, 3,
  0, 1, 3, 2, 0, 2, 3, 1, 3, 3, 3, 3, 3, 3, 2, 3, 3, 0, 3, 1, 0, 1, 0, 3, 1, 2, 3, 0, 0, 3,
  2, 2, 3, 3, 3, 3, 3, 0, 3, 3, 3, 3, 3, 0, 3, 3, 3, 0, 3, 3, 3, 0, 2, 0, 2, 2, 2, 1, 1, 3,
  3, 3, 3, 0, 2, 2, 0, 3, 0, 0, 0, 3, 3, 1, 3, 1, 3, 2, 0, 0, 3, 3, 2, 3, 0, 1, 0, 3, 0, 0,
  0, 3, 0, 2, 3, 1, 0, 2, 1, 1, 3, 3, 0, 1, 1, 0, 1, 0, 3, 0, 0, 3, 3, 3, 3, 2, 2, 2, 1, 3,
  2, 3, 2, 2, 1, 3, 2, 1, 0, 3, 3, 3, 3, 2, 2, 0, 2, 1, 2, 1, 0, 3, 0, 2, 1, 1, 2, 0, 2, 2,
  1, 1, 3, 2, 3, 2, 3, 2, 1, 3, 1, 2, 0, 1, 0, 3, 3, 2, 2, 3, 2, 3, 2, 2, 2, 3, 0, 3, 3, 3,
  0, 0, 0, 3, 2, 2, 0, 2, 1, 1, 0, 1, 0, 2, 1, 3, 2, 2, 3, 3, 3, 1, 3, 3, 3, 3, 0, 3, 3, 0,
  3, 3, 3, 2, 2, 2, 3, 2, 2, 0, 0, 1, 1, 0, 0, 3, 1, 0, 0, 3, 3, 2, 3, 3, 3, 2, 2, 3, 1, 1,
  0, 2, 1, 3, 1, 0, 2, 2, 3, 3, 3, 2, 2, 2, 2, 2, 3, 2, 0, 0, 2, 3, 0, 1, 1, 3, 2, 2, 0, 2,
```

```
3, 2, 2, 3, 0, 2, 2, 3, 0, 0, 0, 2, 2, 2, 1, 3, 2, 1, 1, 3, 3, 0, 3, 2, 2, 3, 2, 3, 2, 2,
1, 2, 2, 2, 0, 2, 2, 0, 2, 3, 0, 2, 3, 3, 0, 0, 3, 0, 3, 0, 2, 2, 2, 2, 3, 1, 0, 3, 0, 2,
2, 1, 1, 0, 0, 2, 3, 3, 2, 2, 3, 2, 2, 0, 2, 2, 2, 2, 2, 3, 3, 0, 1, 0, 0, 0, 2, 3, 3, 2,
2, 1, 0, 3, 1, 1, 0, 0, 2, 0, 3, 0, 0, 1, 0, 0, 1, 0, 2, 3, 2, 2, 0, 1, 1, 1, 0, 0, 1, 0,
1, 1, 3, 1, 3, 3, 3, 2, 0, 3, 3, 0, 2, 0, 2, 2, 3, 2, 0, 3, 2, 2, 2, 2, 3, 1, 3, 1, 3, 2,
2, 2, 2, 3, 0, 0, 0, 0, 3, 3, 1, 0, 3, 0, 3, 3, 3, 0, 2, 1, 1, 3, 3, 3, 1, 3, 0, 0, 3, 0,
1, 2, 2, 3, 0, 2, 3, 0, 3, 3, 2, 2, 0, 0, 0, 2, 2, 3, 0, 2, 2, 2, 2, 3, 0, 2, 2, 2, 2, 2,
3, 3, 2, 2, 3, 2, 1, 1, 2, 1, 1, 3, 2, 0, 2, 2, 2, 2, 2, 2, 2, 0, 2, 2, 0, 0, 1, 3, 2, 2,
1, 1, 2, 0, 3, 2, 3, 3, 2, 0, 0, 3, 3, 3, 2, 0, 2, 2, 3, 0, 2, 3, 3, 0, 0, 1, 0, 3, 3, 1,
1, 0, 0, 2, 0, 3, 2, 2, 1, 3, 2, 1, 2, 3, 2, 3, 0, 3, 1, 1, 3, 1, 1, 3, 3, 3, 3, 0, 3, 2,
2, 3, 2, 0, 2, 3, 0, 1, 0, 0, 0, 3, 3, 1, 3, 2, 3, 0, 2, 0, 0, 0, 2, 2, 1, 2, 2, 2, 0, 0,
3, 3, 2, 0, 0, 2, 0, 3, 0, 1, 1, 1, 2, 3, 1, 3, 3, 3, 1, 2, 2, 1, 2, 1, 1, 0, 3, 1, 3, 2,
3, 0, 0, 1, 2, 2, 3, 3, 3, 1, 3, 2, 0, 0, 2, 2, 1, 2, 2, 2, 2, 3, 2, 3, 2, 1, 1, 0, 0, 0,
3, 0, 3, 2, 2, 3, 1, 3, 3, 1, 3, 1, 1, 2, 2, 0, 2, 2, 0, 3, 2, 3, 3, 3, 1, 1, 0, 0, 2, 2,
3, 2, 2, 1, 3, 2, 1, 2, 2, 2, 2, 1, 1, 2, 2, 2, 3, 1, 1, 3, 3, 1, 3, 3, 1, 3, 2, 1, 2, 2,
3, 0, 0, 1, 2, 1, 1, 3, 1, 1, 3, 3, 2, 2, 1, 1, 0, 1, 2, 3, 1, 0, 3, 1, 1, 3, 0, 3, 0, 0,
0, 0, 2, 3, 2, 0, 0, 0, 2, 0, 0, 2, 3, 2, 1, 2, 1, 3, 2, 1, 3, 2, 3, 0, 2, 3, 0, 3, 3}};
```

PCV1$_2$ =

```
{{0, 1, 1, 0, 2, 1, 2, 1, 0, 1, 3, 3, 1, 2, 2, 1, 0, 2, 1, 2, 2, 1, 0, 2, 1, 0, 1, 1, 3, 1, 2, 2, 1,
  0, 2, 1, 2, 3, 1, 0, 2, 3, 2, 0, 0, 0, 0, 3, 2, 1, 1, 0, 0, 2, 1, 0, 0, 2, 0, 0, 0, 0, 2, 1, 2, 2,
  1, 1, 1, 2, 1, 0, 0, 1, 1, 1, 1, 0, 3, 0, 0, 2, 0, 2, 2, 3, 2, 2, 2, 3, 2, 3, 3, 1, 0, 1, 1, 1,
  3, 3, 0, 0, 3, 0, 0, 3, 1, 1, 3, 3, 1, 1, 2, 0, 2, 2, 0, 2, 2, 0, 2, 0, 0, 0, 0, 0, 1, 0, 0, 0,
  0, 3, 0, 1, 2, 2, 2, 0, 2, 1, 3, 3, 1, 1, 0, 0, 3, 1, 3, 1, 1, 1, 3, 3, 3, 3, 3, 2, 0, 3, 3,
  0, 3, 3, 3, 3, 2, 3, 3, 3, 2, 1, 2, 2, 0, 2, 0, 2, 2, 0, 0, 2, 2, 3, 3, 3, 2, 2, 0, 0, 2, 0,
  2, 2, 2, 3, 0, 2, 0, 0, 1, 3, 1, 1, 3, 1, 0, 1, 1, 3, 1, 1, 0, 2, 2, 2, 2, 3, 3, 3, 2, 1, 2,
  0, 0, 3, 3, 3, 3, 2, 1, 3, 0, 0, 2, 0, 0, 2, 1, 0, 2, 0, 1, 3, 3, 3, 3, 0, 0, 1, 0, 0, 2, 2,
  3, 2, 0, 0, 2, 3, 2, 2, 3, 0, 3, 3, 3, 3, 2, 2, 3, 2, 1, 1, 1, 2, 1, 3, 2, 1, 1, 0, 1, 0, 3,
  1, 2, 0, 2, 0, 0, 0, 2, 1, 2, 0, 0, 0, 2, 2, 0, 0, 1, 1, 2, 0, 1, 1, 0, 2, 1, 0, 2, 0, 0, 3,
  0, 0, 0, 2, 0, 0, 3, 0, 1, 3, 2, 1, 0, 2, 3, 0, 0, 0, 2, 0, 0, 2, 2, 1, 1, 0, 1, 0, 3, 0, 1,
  3, 3, 0, 3, 1, 2, 0, 2, 3, 2, 3, 2, 2, 0, 2, 1, 3, 1, 1, 2, 1, 2, 2, 0, 0, 1, 1, 0, 2, 2, 2,
  2, 0, 0, 2, 1, 2, 1, 0, 2, 1, 2, 0, 1, 1, 3, 2, 3, 1, 3, 0, 1, 3, 2, 1, 3, 2, 3, 2, 0, 2, 3,
  0, 1, 1, 1, 3, 3, 3, 3, 2, 2, 0, 2, 0, 1, 2, 2, 2, 2, 3, 1, 3, 3, 3, 2, 2, 3, 2, 0, 1, 3, 2,
  3, 0, 2, 1, 1, 2, 0, 2, 1, 0, 2, 3, 3, 1, 1, 1, 3, 2, 3, 0, 0, 1, 2, 3, 0, 3, 2, 3, 2, 0,
  2, 0, 0, 0, 3, 3, 3, 1, 1, 2, 1, 2, 2, 2, 1, 3, 2, 2, 1, 3, 2, 0, 0, 1, 3, 3, 3, 3, 2, 0,
  0, 0, 2, 3, 2, 0, 2, 1, 2, 2, 2, 0, 0, 2, 0, 3, 2, 1, 0, 2, 1, 0, 2, 1, 2, 3, 2, 0, 3, 3,
  2, 2, 0, 0, 2, 0, 1, 0, 2, 1, 3, 2, 3, 0, 1, 0, 1, 2, 3, 1, 0, 3, 0, 2, 3, 2, 2, 2, 1, 1,
  1, 2, 1, 1, 1, 2, 2, 3, 3, 2, 3, 2, 2, 2, 0, 0, 2, 0, 2, 1, 1, 0, 2, 3, 2, 2, 2, 1, 1, 1,
  2, 3, 0, 0, 3, 3, 3, 3, 2, 1, 3, 2, 0, 2, 1, 1, 3, 0, 2, 1, 2, 0, 1, 0, 1, 1, 3, 0, 1, 3,
  2, 2, 0, 0, 2, 1, 1, 3, 0, 2, 3, 0, 2, 0, 0, 0, 3, 0, 0, 2, 3, 2, 2, 3, 2, 2, 2, 0, 3, 2,
  2, 0, 3, 0, 3, 1, 0, 3, 2, 2, 0, 2, 0, 0, 2, 0, 0, 2, 3, 3, 2, 3, 3, 2, 3, 3, 3, 3, 2, 2,
  0, 3, 2, 0, 3, 3, 3, 3, 3, 0, 3, 2, 2, 1, 3, 2, 2, 3, 3, 0, 1, 1, 3, 3, 2, 2, 2, 0, 3, 2,
  0, 3, 1, 3, 0, 1, 3, 2, 0, 2, 0, 1, 3, 2, 3, 2, 3, 2, 0, 1, 1, 2, 2, 3, 0, 3, 1, 1, 0, 3,
  3, 2, 0, 1, 3, 2, 3, 0, 2, 0, 2, 0, 1, 3, 0, 0, 0, 2, 2, 1, 2, 2, 3, 0, 1, 3, 2, 3, 3, 1,
  1, 3, 3, 3, 3, 3, 3, 2, 2, 1, 1, 1, 2, 1, 0, 2, 3, 0, 3, 3, 3, 3, 2, 0, 3, 3, 0, 1, 1, 0,
  2, 1, 0, 0, 3, 1, 0, 2, 2, 1, 1, 1, 1, 1, 1, 0, 2, 2, 0, 0, 3, 2, 2, 3, 0, 1, 3, 1, 1, 3,
  1, 0, 0, 1, 3, 2, 1, 3, 2, 3, 1, 1, 1, 0, 2, 1, 3, 2, 3, 0, 2, 0, 0, 2, 1, 3, 1, 3, 1, 3,
  0, 3, 1, 2, 2, 0, 2, 2, 0, 3, 3, 0, 1, 3, 0, 1, 3, 3, 3, 2, 1, 0, 0, 3, 3, 3, 3, 2, 2, 0,
  0, 2, 0, 1, 3, 2, 1, 3, 2, 2, 0, 2, 0, 0, 1, 0, 0, 3, 1, 1, 0, 1, 2, 2, 0, 2, 2, 3, 0, 1,
  1, 1, 2, 0, 0, 2, 2, 1, 1, 2, 0, 3, 3, 3, 2, 0, 0, 2, 1, 0, 2, 3, 2, 2, 0, 1, 1, 1, 0, 1,
  1, 1, 3, 2, 3, 2, 1, 1, 1, 3, 3, 3, 3, 1, 1, 1, 0, 3, 0, 3, 0, 0, 0, 0, 3, 0, 0, 0, 3, 3,
  0, 1, 3, 2, 0, 2, 3, 1, 3, 3, 3, 3, 3, 3, 2, 3, 3, 0, 3, 1, 0, 1, 0, 3, 1, 2, 3, 0, 0, 3,
```

```
    2, 2, 3, 3, 3, 3, 3, 0, 3, 3, 3, 3, 3, 0, 3, 3, 3, 0, 3, 3, 3, 0, 2, 0, 2, 2, 2, 3, 1, 3,
    3, 3, 3, 0, 2, 2, 0, 3, 0, 0, 0, 3, 3, 1, 3, 1, 3, 2, 0, 0, 3, 3, 2, 3, 0, 1, 0, 3, 0, 0,
    0, 3, 0, 2, 3, 1, 0, 2, 1, 1, 3, 3, 0, 1, 1, 0, 1, 0, 3, 0, 0, 3, 3, 3, 3, 2, 2, 2, 1, 3,
    2, 3, 2, 2, 1, 3, 2, 1, 0, 3, 3, 3, 3, 2, 2, 0, 2, 1, 2, 1, 0, 3, 0, 2, 1, 1, 2, 0, 2, 2,
    1, 1, 3, 2, 3, 2, 3, 2, 1, 3, 1, 2, 0, 1, 0, 3, 3, 2, 2, 3, 2, 3, 2, 2, 2, 3, 0, 3, 3, 3,
    0, 0, 0, 3, 2, 2, 0, 2, 1, 1, 0, 1, 0, 2, 1, 3, 2, 2, 3, 3, 3, 1, 3, 3, 3, 3, 0, 3, 3, 0,
    3, 3, 3, 2, 2, 2, 3, 2, 2, 0, 0, 1, 1, 0, 0, 3, 1, 0, 0, 3, 3, 2, 3, 3, 3, 2, 2, 3, 1, 1,
    0, 2, 1, 3, 1, 0, 2, 2, 3, 3, 3, 2, 2, 2, 2, 2, 3, 2, 0, 0, 2, 3, 0, 1, 1, 3, 2, 2, 0, 2,
    3, 2, 2, 3, 0, 2, 2, 3, 0, 0, 0, 2, 2, 2, 1, 3, 2, 1, 1, 3, 3, 0, 3, 2, 2, 3, 2, 3, 2, 2,
    1, 2, 2, 2, 0, 2, 2, 0, 2, 3, 0, 2, 3, 3, 0, 0, 3, 0, 3, 0, 2, 2, 2, 2, 3, 1, 0, 3, 0, 2,
    2, 1, 1, 0, 0, 2, 3, 3, 2, 2, 3, 2, 2, 0, 2, 2, 2, 2, 2, 3, 3, 0, 1, 0, 0, 0, 2, 3, 3, 2,
    2, 1, 0, 3, 1, 1, 0, 0, 2, 0, 3, 0, 0, 1, 0, 0, 1, 0, 2, 3, 2, 2, 0, 1, 1, 1, 0, 0, 1, 0,
    1, 1, 3, 1, 3, 3, 3, 1, 0, 3, 3, 0, 2, 0, 2, 2, 3, 2, 0, 3, 2, 2, 2, 2, 3, 1, 3, 1, 3, 2,
    2, 2, 2, 3, 0, 0, 0, 0, 3, 3, 1, 0, 3, 0, 3, 3, 3, 0, 2, 1, 1, 3, 3, 3, 1, 3, 0, 0, 3, 0,
    1, 2, 2, 3, 0, 2, 3, 0, 3, 3, 2, 2, 0, 0, 0, 2, 2, 3, 0, 2, 2, 2, 2, 3, 0, 2, 2, 2, 2, 2,
    3, 3, 2, 2, 3, 2, 1, 1, 2, 1, 1, 3, 2, 0, 2, 2, 2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 0, 0, 1, 3, 2, 2,
    1, 1, 2, 0, 3, 2, 3, 3, 2, 0, 0, 3, 2, 3, 2, 0, 2, 2, 3, 2, 2, 3, 3, 0, 0, 1, 0, 3, 2, 1,
    1, 0, 0, 2, 0, 3, 2, 2, 1, 3, 2, 1, 2, 0, 2, 3, 0, 3, 1, 1, 3, 1, 1, 3, 3, 3, 3, 0, 3, 2,
    2, 3, 2, 0, 2, 3, 0, 1, 0, 0, 0, 3, 3, 1, 3, 1, 3, 0, 2, 0, 0, 0, 2, 2, 1, 2, 2, 2, 0, 0,
    3, 3, 2, 0, 0, 2, 0, 3, 0, 1, 1, 1, 2, 3, 1, 3, 3, 3, 1, 2, 2, 1, 2, 1, 1, 0, 3, 1, 3, 2,
    3, 0, 0, 1, 2, 2, 3, 3, 3, 1, 3, 2, 0, 0, 2, 2, 1, 2, 2, 2, 2, 3, 2, 3, 2, 1, 1, 0, 0, 0,
    3, 0, 3, 2, 2, 3, 1, 3, 3, 1, 3, 1, 1, 2, 2, 0, 2, 2, 0, 3, 2, 3, 3, 3, 1, 1, 0, 0, 2, 0,
    3, 2, 2, 1, 3, 2, 1, 2, 2, 2, 2, 2, 1, 2, 2, 2, 3, 1, 1, 3, 3, 1, 3, 3, 1, 3, 2, 1, 2, 2,
    3, 0, 0, 1, 2, 1, 1, 3, 1, 1, 3, 3, 2, 2, 1, 1, 0, 1, 2, 3, 1, 0, 3, 1, 1, 3, 0, 3, 0, 0,
    0, 0, 2, 3, 2, 0, 0, 0, 2, 0, 0, 2, 3, 2, 1, 2, 1, 3, 2, 1, 3, 2, 3, 0, 2, 3, 0, 3, 3}};
```

PCV1$_3$ =

```
  {{0, 1, 1, 0, 2, 1, 2, 1, 0, 1, 3, 3, 1, 2, 2, 1, 0, 2, 1, 2, 2, 1, 0, 2, 1, 0, 1, 1, 3, 1, 2, 2, 1,
    0, 2, 1, 2, 3, 1, 0, 2, 3, 2, 0, 0, 0, 0, 3, 2, 1, 1, 0, 0, 2, 1, 0, 0, 2, 0, 0, 0, 0, 2, 1, 2, 2,
    1, 1, 1, 2, 1, 0, 0, 1, 1, 1, 1, 0, 3, 0, 0, 2, 0, 2, 2, 3, 2, 2, 2, 3, 2, 3, 3, 1, 0, 1, 1, 1,
    3, 3, 0, 0, 3, 0, 0, 3, 1, 1, 3, 3, 1, 1, 2, 0, 2, 2, 0, 2, 2, 0, 2, 0, 0, 0, 0, 0, 1, 0, 0, 0,
    0, 3, 0, 1, 2, 2, 2, 0, 2, 1, 3, 3, 1, 1, 0, 0, 3, 1, 3, 1, 1, 1, 3, 3, 3, 3, 3, 2, 0, 3, 3,
    0, 3, 3, 3, 3, 2, 3, 3, 3, 2, 1, 2, 2, 0, 2, 0, 2, 2, 0, 0, 2, 2, 3, 3, 3, 2, 2, 0, 0, 2, 0,
    2, 2, 2, 3, 0, 2, 0, 0, 1, 3, 1, 1, 3, 1, 0, 1, 1, 3, 1, 1, 0, 2, 2, 2, 2, 3, 3, 3, 2, 1, 2,
    0, 0, 3, 3, 3, 3, 2, 1, 3, 0, 0, 2, 0, 0, 2, 1, 0, 2, 0, 1, 3, 3, 3, 3, 0, 0, 1, 0, 0, 2, 2,
    3, 2, 0, 0, 2, 3, 2, 2, 3, 0, 3, 3, 3, 3, 2, 2, 3, 2, 1, 1, 1, 2, 1, 3, 2, 1, 1, 0, 1, 0, 3,
    1, 2, 0, 2, 0, 0, 0, 2, 1, 2, 0, 0, 0, 2, 2, 0, 0, 1, 1, 2, 0, 1, 1, 0, 2, 1, 0, 2, 0, 0, 3,
    0, 0, 0, 2, 0, 0, 3, 0, 1, 3, 2, 1, 0, 2, 3, 0, 0, 0, 2, 0, 0, 2, 2, 1, 1, 0, 1, 0, 3, 0, 1,
    3, 3, 0, 3, 1, 2, 0, 2, 3, 2, 3, 2, 2, 0, 2, 1, 3, 1, 1, 2, 1, 2, 2, 0, 0, 1, 1, 0, 2, 2, 2,
    2, 0, 0, 2, 1, 2, 1, 0, 2, 1, 2, 0, 1, 1, 3, 2, 3, 1, 3, 0, 1, 3, 2, 1, 3, 2, 3, 2, 0, 2, 3,
    0, 1, 1, 1, 3, 3, 3, 3, 2, 2, 0, 2, 0, 1, 2, 2, 2, 2, 3, 1, 3, 3, 3, 2, 2, 3, 2, 0, 1, 3, 2,
    3, 0, 2, 1, 1, 2, 0, 2, 1, 0, 2, 3, 3, 1, 1, 1, 3, 2, 3, 0, 0, 1, 2, 3, 0, 3, 2, 3, 2, 0,
    2, 0, 0, 0, 3, 3, 3, 1, 1, 2, 1, 2, 2, 2, 1, 3, 2, 2, 1, 3, 2, 0, 0, 1, 3, 3, 3, 3, 2, 0,
    0, 0, 2, 3, 2, 0, 2, 1, 2, 2, 2, 0, 0, 2, 0, 3, 2, 1, 0, 2, 1, 0, 2, 1, 2, 3, 2, 0, 3, 3,
    2, 2, 0, 0, 2, 0, 1, 0, 2, 1, 3, 2, 3, 0, 1, 0, 1, 2, 3, 1, 0, 3, 0, 2, 3, 2, 2, 2, 1, 1,
    1, 2, 1, 1, 1, 2, 2, 3, 3, 2, 3, 2, 2, 2, 0, 0, 2, 0, 2, 1, 1, 0, 2, 3, 2, 2, 2, 1, 1, 1,
    2, 3, 0, 0, 3, 3, 3, 3, 2, 1, 3, 2, 0, 2, 1, 1, 3, 0, 2, 1, 2, 0, 1, 0, 1, 1, 3, 0, 1, 3,
    2, 2, 0, 0, 2, 1, 1, 3, 0, 2, 3, 0, 2, 0, 0, 0, 3, 0, 0, 2, 3, 2, 2, 3, 2, 2, 2, 0, 3, 2,
    2, 0, 3, 0, 3, 1, 0, 3, 2, 2, 0, 2, 0, 0, 2, 0, 0, 2, 3, 3, 2, 3, 3, 2, 3, 3, 3, 3, 2, 2,
    0, 3, 2, 0, 3, 3, 3, 3, 3, 0, 3, 2, 2, 1, 3, 2, 2, 3, 3, 0, 1, 1, 3, 3, 2, 2, 2, 0, 3, 2,
    0, 3, 1, 3, 0, 1, 3, 2, 0, 2, 0, 1, 3, 2, 3, 2, 3, 2, 0, 1, 1, 2, 2, 3, 0, 3, 1, 1, 0, 3,
    3, 2, 0, 1, 3, 2, 3, 0, 2, 0, 2, 0, 1, 3, 0, 0, 0, 2, 2, 1, 2, 2, 3, 0, 1, 3, 2, 3, 3, 1,
```

```
1, 3, 3, 3, 3, 3, 3, 2, 2, 1, 1, 1, 2, 1, 0, 2, 3, 0, 3, 3, 3, 3, 2, 0, 3, 3, 0, 1, 1, 0,
2, 1, 0, 0, 3, 1, 0, 2, 2, 1, 1, 1, 1, 1, 1, 0, 2, 2, 0, 0, 3, 2, 2, 3, 0, 1, 3, 1, 1, 3,
1, 0, 0, 1, 3, 2, 1, 3, 2, 3, 1, 1, 1, 0, 2, 1, 3, 2, 3, 0, 2, 0, 0, 2, 1, 3, 1, 3, 1, 3,
0, 3, 1, 2, 2, 0, 2, 2, 0, 3, 3, 0, 1, 3, 0, 1, 3, 3, 3, 2, 1, 0, 0, 3, 3, 3, 3, 2, 2, 0,
0, 2, 0, 1, 3, 2, 1, 3, 2, 2, 0, 2, 0, 0, 1, 0, 0, 3, 1, 1, 0, 1, 2, 2, 0, 2, 2, 3, 0, 1,
1, 1, 2, 0, 0, 2, 2, 1, 1, 2, 0, 3, 3, 3, 2, 0, 0, 2, 1, 0, 2, 3, 2, 2, 0, 1, 1, 1, 0, 1,
1, 1, 3, 2, 3, 2, 1, 1, 1, 3, 3, 3, 3, 1, 1, 1, 0, 3, 0, 3, 0, 0, 0, 0, 3, 0, 0, 0, 3, 3,
0, 1, 3, 2, 0, 2, 3, 1, 3, 3, 3, 3, 3, 3, 2, 3, 3, 0, 3, 1, 0, 1, 0, 3, 1, 2, 3, 0, 0, 3,
2, 2, 3, 3, 3, 3, 3, 0, 3, 3, 3, 3, 3, 0, 3, 3, 3, 0, 3, 3, 3, 0, 2, 0, 2, 2, 2, 3, 1, 3,
3, 3, 3, 0, 2, 2, 0, 3, 0, 0, 0, 3, 3, 1, 3, 1, 3, 2, 0, 0, 3, 3, 2, 3, 0, 1, 0, 3, 0, 0,
0, 3, 0, 2, 3, 1, 0, 2, 1, 1, 3, 3, 0, 1, 1, 0, 1, 0, 3, 0, 0, 3, 3, 3, 3, 2, 2, 2, 1, 3,
2, 3, 2, 2, 1, 3, 2, 1, 0, 3, 3, 3, 3, 2, 2, 0, 2, 1, 2, 1, 0, 3, 0, 2, 1, 1, 2, 0, 2, 2,
1, 1, 3, 2, 3, 2, 3, 2, 1, 3, 1, 2, 0, 1, 0, 3, 3, 2, 2, 3, 2, 3, 2, 2, 2, 3, 0, 3, 3, 3,
0, 0, 0, 3, 2, 2, 0, 2, 1, 1, 0, 1, 0, 2, 1, 3, 2, 2, 3, 3, 3, 1, 3, 3, 3, 3, 0, 3, 3, 0,
3, 3, 3, 2, 2, 3, 3, 2, 2, 0, 0, 1, 1, 0, 0, 3, 1, 0, 0, 3, 3, 2, 3, 3, 3, 2, 2, 3, 1, 1,
0, 2, 1, 3, 1, 0, 2, 2, 3, 3, 3, 2, 2, 2, 2, 2, 3, 2, 0, 0, 2, 3, 0, 1, 1, 3, 2, 2, 0, 2,
3, 2, 2, 3, 0, 2, 2, 3, 0, 0, 0, 2, 2, 2, 1, 3, 2, 1, 1, 3, 3, 0, 3, 2, 2, 3, 2, 3, 2, 2,
1, 2, 2, 2, 0, 2, 2, 0, 2, 3, 0, 2, 3, 3, 0, 0, 3, 0, 3, 0, 2, 2, 2, 2, 3, 1, 0, 3, 0, 2,
2, 1, 1, 0, 0, 2, 3, 3, 2, 2, 3, 2, 2, 0, 2, 2, 2, 2, 2, 3, 3, 0, 1, 0, 0, 0, 2, 3, 3, 2,
2, 1, 0, 3, 1, 1, 0, 0, 2, 0, 3, 0, 0, 1, 0, 0, 1, 0, 2, 3, 2, 2, 0, 1, 1, 1, 0, 0, 1, 0,
1, 1, 3, 1, 3, 3, 3, 1, 0, 3, 3, 0, 2, 0, 2, 2, 3, 2, 0, 3, 2, 2, 2, 2, 3, 1, 3, 1, 3, 2,
2, 2, 2, 3, 0, 0, 0, 0, 3, 3, 1, 0, 3, 0, 3, 3, 3, 0, 2, 1, 1, 3, 3, 3, 1, 3, 0, 0, 3, 0,
1, 2, 2, 3, 0, 2, 3, 0, 3, 3, 2, 2, 0, 0, 0, 2, 2, 3, 0, 2, 2, 2, 2, 3, 0, 2, 2, 2, 2, 2,
3, 3, 2, 2, 3, 2, 1, 1, 2, 1, 1, 3, 2, 0, 2, 2, 2, 2, 2, 2, 2, 0, 2, 2, 0, 0, 1, 3, 2, 2,
1, 1, 2, 0, 3, 2, 3, 3, 2, 0, 0, 3, 3, 3, 2, 0, 2, 2, 3, 2, 2, 3, 3, 0, 0, 1, 0, 3, 2, 1,
1, 0, 0, 2, 0, 3, 2, 2, 1, 3, 2, 1, 2, 0, 2, 3, 0, 3, 1, 1, 3, 1, 1, 3, 3, 3, 3, 0, 3, 2,
2, 3, 2, 0, 2, 3, 0, 1, 0, 0, 0, 3, 3, 1, 3, 2, 3, 0, 2, 0, 0, 0, 2, 2, 1, 2, 2, 2, 0, 0,
3, 3, 2, 0, 0, 2, 0, 3, 0, 1, 1, 1, 2, 3, 1, 3, 3, 3, 1, 2, 2, 1, 2, 1, 1, 0, 3, 1, 3, 2,
3, 0, 0, 1, 2, 2, 3, 3, 3, 1, 3, 2, 0, 0, 2, 2, 1, 2, 2, 2, 2, 3, 2, 3, 2, 1, 1, 0, 0, 0,
3, 0, 3, 2, 2, 3, 1, 3, 3, 1, 3, 1, 1, 2, 2, 0, 2, 2, 0, 3, 2, 3, 3, 3, 1, 1, 0, 0, 2, 0,
3, 2, 2, 1, 3, 2, 1, 2, 2, 2, 2, 2, 1, 2, 2, 2, 3, 1, 1, 3, 3, 1, 2, 3, 1, 3, 2, 1, 2, 2,
3, 0, 0, 1, 2, 1, 1, 3, 1, 1, 3, 3, 2, 2, 1, 1, 0, 1, 2, 3, 1, 0, 3, 1, 1, 3, 0, 3, 0, 0,
0, 0, 2, 3, 2, 0, 0, 0, 2, 0, 0, 2, 3, 2, 1, 2, 1, 3, 2, 1, 3, 2, 3, 0, 2, 3, 0, 3, 3}};
```

$PCV1_4$ =

```
{{0, 1, 1, 0, 2, 1, 2, 1, 0, 1, 3, 3, 1, 2, 2, 1, 0, 2, 1, 2, 2, 1, 0, 2, 1, 0, 1, 1, 3, 1, 2, 2, 1,
  0, 2, 1, 2, 3, 1, 0, 2, 3, 2, 0, 0, 0, 0, 3, 2, 1, 1, 0, 0, 2, 1, 0, 0, 2, 0, 0, 0, 0, 2, 1, 2, 2,
  1, 1, 1, 2, 1, 0, 0, 1, 1, 1, 1, 0, 3, 0, 0, 2, 0, 2, 2, 3, 2, 2, 2, 3, 2, 3, 3, 1, 0, 1, 1, 1,
  3, 3, 0, 0, 3, 0, 0, 3, 1, 1, 3, 3, 1, 1, 2, 0, 2, 2, 0, 2, 2, 0, 2, 0, 0, 0, 0, 0, 1, 0, 0, 0,
  0, 3, 0, 1, 2, 2, 2, 0, 2, 1, 3, 3, 1, 1, 0, 0, 3, 1, 3, 1, 1, 1, 3, 3, 3, 3, 3, 2, 0, 3, 3,
  0, 3, 3, 3, 3, 2, 3, 3, 3, 2, 1, 2, 2, 0, 2, 0, 2, 2, 0, 0, 2, 2, 3, 3, 3, 2, 2, 0, 0, 2, 0,
  2, 2, 2, 3, 0, 2, 0, 0, 1, 3, 1, 1, 3, 1, 0, 1, 1, 3, 1, 1, 0, 2, 2, 2, 2, 3, 3, 3, 2, 1, 2,
  0, 0, 3, 3, 3, 3, 2, 1, 3, 0, 0, 2, 0, 0, 2, 1, 0, 2, 0, 1, 3, 3, 3, 3, 0, 0, 1, 0, 0, 2, 2,
  3, 2, 0, 0, 2, 3, 2, 2, 3, 0, 3, 3, 3, 3, 2, 2, 3, 2, 1, 1, 1, 2, 1, 3, 2, 1, 1, 0, 1, 0, 3,
  1, 2, 0, 2, 0, 0, 0, 2, 1, 2, 0, 0, 0, 2, 2, 0, 0, 1, 1, 2, 0, 1, 1, 0, 2, 1, 0, 2, 0, 0, 3,
  0, 0, 0, 2, 0, 0, 3, 0, 1, 3, 2, 1, 0, 2, 3, 0, 0, 0, 2, 0, 0, 2, 2, 1, 1, 0, 1, 0, 3, 0, 1,
  3, 3, 0, 3, 1, 2, 0, 2, 3, 2, 3, 2, 2, 0, 2, 1, 3, 1, 1, 2, 1, 2, 2, 0, 0, 1, 1, 0, 2, 2, 2,
  2, 0, 0, 2, 1, 2, 1, 0, 2, 1, 2, 0, 1, 1, 3, 2, 3, 1, 3, 0, 1, 3, 2, 1, 3, 2, 3, 2, 0, 2, 3,
  0, 1, 1, 1, 3, 3, 3, 3, 2, 2, 0, 2, 0, 1, 2, 2, 2, 2, 3, 1, 3, 3, 3, 2, 2, 3, 2, 0, 1, 3, 2,
  3, 0, 2, 1, 1, 2, 0, 2, 1, 0, 2, 3, 3, 1, 1, 1, 3, 2, 3, 0, 0, 1, 2, 3, 0, 3, 2, 3, 2, 0,
  2, 0, 0, 0, 3, 3, 3, 1, 1, 2, 1, 2, 2, 2, 1, 3, 2, 2, 1, 3, 2, 0, 0, 1, 3, 3, 3, 3, 2, 0,
  0, 0, 2, 3, 2, 0, 2, 1, 2, 2, 2, 0, 0, 2, 0, 3, 2, 1, 0, 2, 1, 0, 2, 1, 2, 3, 2, 0, 3, 3,
```

```
2, 2, 0, 0, 2, 0, 1, 0, 2, 1, 3, 2, 3, 0, 1, 0, 1, 2, 3, 1, 0, 3, 0, 2, 3, 2, 2, 2, 1, 1,
1, 2, 1, 1, 1, 2, 2, 3, 3, 2, 3, 2, 2, 2, 0, 0, 2, 0, 2, 1, 1, 0, 2, 3, 2, 2, 2, 1, 1, 1,
2, 3, 0, 0, 3, 3, 3, 3, 2, 1, 3, 2, 0, 2, 1, 1, 3, 0, 2, 1, 2, 0, 1, 0, 1, 1, 3, 0, 1, 3,
2, 2, 0, 0, 2, 1, 1, 3, 0, 2, 3, 0, 2, 0, 0, 0, 3, 0, 0, 2, 3, 2, 2, 3, 2, 2, 2, 0, 3, 2,
2, 0, 3, 0, 3, 1, 0, 3, 2, 2, 0, 2, 0, 0, 2, 0, 0, 2, 3, 3, 2, 3, 3, 2, 3, 3, 3, 3, 2, 2,
0, 3, 2, 0, 3, 3, 3, 3, 3, 0, 3, 2, 2, 1, 3, 2, 2, 3, 3, 0, 1, 1, 3, 3, 2, 2, 2, 0, 3, 2,
0, 3, 1, 3, 0, 1, 3, 2, 0, 2, 0, 1, 3, 2, 3, 2, 3, 2, 0, 1, 1, 2, 2, 3, 0, 3, 1, 1, 0, 3,
3, 2, 0, 1, 3, 2, 3, 0, 2, 0, 2, 0, 1, 3, 0, 0, 0, 2, 2, 2, 2, 2, 3, 0, 1, 3, 2, 3, 3, 1,
1, 3, 3, 3, 3, 3, 3, 2, 2, 1, 1, 1, 2, 1, 0, 2, 3, 0, 3, 3, 3, 3, 2, 0, 3, 3, 0, 1, 1, 0,
2, 1, 0, 0, 3, 1, 0, 2, 2, 1, 1, 1, 1, 1, 1, 0, 2, 2, 0, 0, 3, 2, 2, 3, 0, 1, 3, 1, 1, 3,
1, 0, 0, 1, 3, 2, 1, 3, 2, 3, 1, 1, 1, 0, 2, 1, 3, 2, 3, 0, 2, 0, 0, 2, 1, 3, 1, 3, 1, 3,
0, 3, 1, 2, 2, 0, 2, 2, 0, 3, 3, 0, 1, 3, 0, 1, 3, 3, 3, 2, 1, 0, 0, 3, 3, 3, 3, 2, 2, 0,
0, 2, 0, 1, 3, 2, 1, 3, 2, 2, 0, 2, 0, 0, 1, 0, 0, 3, 1, 1, 0, 1, 2, 2, 0, 2, 2, 3, 0, 1,
1, 1, 2, 0, 0, 2, 2, 1, 1, 2, 0, 3, 3, 3, 2, 0, 0, 2, 1, 0, 2, 3, 2, 2, 0, 1, 1, 1, 0, 1,
1, 1, 3, 2, 3, 2, 1, 1, 1, 3, 3, 3, 3, 1, 1, 1, 0, 3, 0, 3, 0, 0, 0, 0, 3, 0, 0, 0, 3, 3,
0, 1, 3, 2, 0, 2, 3, 1, 3, 3, 3, 3, 3, 3, 2, 3, 3, 0, 3, 1, 0, 1, 0, 3, 1, 2, 3, 0, 0, 3,
2, 2, 3, 3, 3, 3, 3, 0, 3, 3, 3, 3, 3, 0, 3, 3, 3, 0, 3, 3, 3, 0, 2, 0, 2, 2, 2, 3, 1, 3,
3, 3, 3, 0, 2, 2, 0, 3, 0, 0, 0, 3, 3, 1, 3, 1, 3, 2, 0, 0, 3, 3, 2, 3, 0, 1, 0, 3, 0, 0,
0, 3, 0, 2, 3, 1, 0, 2, 1, 1, 3, 3, 0, 1, 1, 0, 1, 0, 3, 0, 0, 3, 3, 3, 3, 2, 2, 2, 1, 3,
2, 3, 2, 2, 1, 3, 2, 1, 0, 3, 3, 3, 3, 2, 2, 0, 2, 1, 2, 1, 0, 3, 0, 2, 1, 1, 2, 0, 2, 2,
1, 1, 3, 2, 3, 2, 3, 2, 1, 3, 1, 2, 0, 1, 0, 3, 3, 2, 2, 3, 2, 3, 2, 2, 2, 3, 0, 3, 3, 3,
0, 0, 0, 3, 2, 2, 0, 2, 1, 1, 0, 1, 0, 2, 1, 3, 2, 2, 3, 3, 3, 1, 3, 3, 3, 3, 0, 3, 3, 0,
3, 3, 3, 2, 2, 2, 3, 2, 2, 0, 0, 1, 1, 0, 0, 3, 1, 0, 0, 3, 3, 2, 3, 3, 3, 2, 2, 3, 1, 1,
0, 2, 1, 3, 1, 0, 2, 2, 3, 3, 3, 2, 2, 2, 2, 2, 3, 2, 0, 0, 2, 3, 0, 1, 1, 3, 2, 2, 0, 2,
3, 2, 2, 3, 0, 2, 2, 3, 0, 0, 0, 2, 2, 2, 1, 3, 2, 1, 1, 3, 3, 0, 3, 2, 2, 3, 2, 3, 2, 2,
1, 2, 2, 2, 0, 2, 2, 0, 2, 3, 0, 2, 3, 3, 0, 0, 3, 0, 3, 0, 2, 2, 2, 2, 3, 1, 0, 3, 0, 2,
2, 1, 1, 0, 0, 2, 3, 3, 2, 2, 3, 2, 2, 0, 2, 2, 2, 2, 2, 3, 3, 0, 1, 0, 0, 0, 2, 3, 3, 2,
2, 1, 0, 3, 1, 1, 0, 0, 2, 0, 3, 0, 0, 1, 0, 0, 1, 0, 2, 3, 2, 2, 0, 1, 1, 1, 0, 0, 1, 0,
1, 1, 3, 1, 3, 3, 3, 1, 0, 3, 3, 0, 2, 0, 2, 2, 3, 2, 0, 3, 2, 2, 2, 2, 3, 1, 3, 1, 3, 2,
2, 2, 2, 3, 0, 0, 0, 0, 3, 3, 1, 0, 3, 0, 3, 3, 3, 0, 2, 1, 1, 3, 3, 3, 1, 3, 0, 0, 3, 0,
1, 2, 2, 3, 0, 2, 3, 0, 3, 3, 2, 2, 0, 0, 0, 2, 2, 3, 0, 2, 2, 2, 2, 3, 0, 2, 2, 2, 2, 2,
3, 3, 2, 2, 3, 2, 1, 1, 2, 1, 1, 3, 2, 0, 2, 2, 2, 2, 2, 2, 2, 0, 2, 2, 0, 0, 1, 3, 2, 2,
1, 1, 2, 0, 3, 2, 3, 3, 2, 0, 0, 3, 1, 3, 2, 0, 2, 2, 3, 2, 2, 3, 3, 0, 0, 1, 0, 3, 2, 1,
1, 0, 0, 2, 0, 3, 2, 2, 1, 3, 2, 1, 2, 0, 2, 3, 0, 3, 1, 1, 3, 1, 1, 3, 3, 3, 3, 0, 3, 2,
2, 3, 2, 0, 2, 3, 0, 1, 0, 0, 0, 3, 3, 1, 3, 2, 3, 0, 2, 0, 0, 0, 2, 2, 1, 2, 2, 2, 0, 0,
3, 3, 2, 0, 0, 2, 0, 3, 0, 1, 1, 1, 2, 3, 1, 3, 3, 3, 1, 2, 2, 1, 2, 1, 1, 0, 3, 1, 3, 2,
3, 0, 0, 1, 2, 2, 3, 3, 3, 1, 3, 2, 0, 0, 2, 2, 1, 2, 2, 2, 2, 3, 2, 3, 2, 1, 1, 0, 0, 0,
3, 0, 3, 2, 2, 3, 1, 3, 3, 1, 3, 1, 1, 2, 2, 0, 2, 2, 0, 3, 2, 3, 3, 3, 1, 1, 0, 0, 2, 0,
3, 2, 2, 1, 3, 2, 1, 2, 2, 2, 2, 2, 1, 2, 2, 2, 3, 1, 1, 3, 3, 1, 3, 3, 1, 3, 2, 1, 2, 2,
3, 0, 0, 1, 2, 1, 1, 3, 1, 1, 3, 3, 2, 2, 1, 1, 0, 1, 2, 3, 1, 0, 3, 1, 1, 3, 0, 3, 0, 0,
0, 0, 2, 3, 2, 0, 0, 0, 2, 0, 0, 2, 3, 2, 1, 2, 1, 3, 2, 1, 3, 2, 3, 0, 2, 3, 0, 3, 3}};
```

Porcine Circovirus Type 2 samples

PCV2$_1$ is from GenBank: LF939182.1

>>> Note that Length of PCV2 samples is 1767, whereas PCV1 samples are 1759, so we would need to fill the PCV1 samples a bit if we wish to compare them

PCV2$_1$ =

```
{{2, 3, 1, 3, 3, 3, 3, 3, 3, 0, 3, 1, 0, 1, 3, 3, 1, 2, 3, 0, 0, 3, 2, 2, 3, 3, 3, 3, 3, 0, 3, 3, 0,
  3, 3, 1, 0, 1, 3, 3, 0, 2, 2, 2, 3, 3, 0, 0, 2, 3, 2, 2, 2, 2, 2, 2, 3, 1, 3, 3, 3, 0, 0, 2, 0, 3,
```

3, 0, 0, 0, 3, 3, 1, 3, 1, 3, 2, 0, 0, 3, 3, 2, 3, 0, 1, 0, 3, 0, 1, 0, 3, 2, 2, 3, 3, 0, 3, 0,
1, 2, 2, 0, 3, 0, 3, 3, 2, 3, 0, 2, 3, 1, 1, 3, 2, 2, 3, 1, 2, 3, 0, 3, 0, 3, 0, 1, 3, 2, 3, 3,
3, 3, 1, 2, 0, 0, 1, 2, 1, 0, 2, 3, 2, 1, 1, 2, 0, 2, 2, 1, 1, 3, 0, 1, 0, 3, 2, 2, 3, 1, 3,
0, 1, 0, 3, 3, 3, 1, 1, 0, 2, 3, 0, 2, 3, 3, 3, 2, 3, 0, 2, 3, 1, 3, 1, 0, 2, 1, 1, 0, 2, 0,
2, 3, 3, 2, 0, 3, 3, 3, 1, 3, 3, 3, 3, 2, 3, 3, 0, 3, 3, 2, 2, 2, 3, 3, 2, 2, 0, 0, 2, 3, 0,
0, 3, 1, 2, 0, 3, 3, 2, 3, 1, 1, 3, 0, 3, 1, 0, 0, 2, 2, 0, 1, 0, 2, 2, 3, 3, 3, 1, 2, 2, 2,
2, 3, 0, 0, 0, 2, 3, 0, 1, 1, 2, 2, 2, 0, 2, 3, 2, 2, 3, 0, 2, 2, 0, 2, 0, 0, 2, 2, 2, 1, 3,
2, 2, 2, 3, 3, 0, 3, 2, 2, 3, 0, 3, 2, 2, 1, 2, 2, 2, 0, 2, 2, 0, 2, 3, 0, 2, 3, 3, 3, 0, 1,
0, 3, 0, 2, 2, 2, 2, 3, 1, 0, 3, 0, 2, 2, 3, 3, 0, 2, 2, 2, 1, 0, 3, 3, 2, 2, 1, 1, 3, 3, 3,
2, 3, 3, 0, 1, 0, 0, 0, 2, 3, 3, 0, 3, 1, 0, 3, 1, 3, 0, 2, 0, 0, 3, 0, 0, 1, 0, 2, 1, 0, 2,
3, 2, 2, 0, 2, 1, 1, 1, 0, 1, 3, 1, 1, 1, 1, 3, 2, 3, 1, 0, 1, 1, 1, 3, 2, 2, 2, 3, 2, 0, 3,
3, 2, 2, 2, 2, 0, 2, 1, 0, 2, 2, 2, 1, 1, 0, 2, 0, 0, 3, 3, 1, 0, 0, 1, 1, 3, 3, 0, 0, 1, 1,
3, 3, 1, 1, 3, 3, 0, 3, 3, 1, 3, 2, 3, 0, 2, 3, 0, 3, 3, 1, 0, 0, 0, 2, 2, 2, 1, 0, 1, 0, 2,
3, 2, 0, 2, 2, 2, 2, 2, 3, 3, 3, 2, 0, 2, 1, 1, 1, 1, 1, 3, 1, 1, 3, 2, 2, 2, 2, 2, 0, 0, 2,
0, 0, 0, 0, 3, 1, 0, 3, 3, 0, 0, 3, 0, 3, 3, 0, 0, 0, 3, 1, 3, 1, 0, 3, 1, 0, 3, 2, 3, 1, 1,
0, 1, 0, 3, 3, 1, 1, 0, 2, 2, 0, 2, 2, 2, 1, 2, 3, 3, 1, 3, 2, 0, 1, 3, 2, 3, 2, 2, 3, 3, 3,
3, 1, 3, 3, 2, 0, 1, 0, 2, 3, 0, 3, 0, 0, 1, 1, 2, 0, 3, 2, 2, 3, 2, 1, 2, 2, 2, 0, 2, 0, 2,
2, 1, 2, 2, 2, 3, 2, 3, 3, 2, 0, 0, 2, 0, 3, 2, 1, 1, 0, 3, 3, 3, 3, 3, 1, 1, 3, 3, 1, 3, 1,
1, 0, 2, 1, 2, 2, 3, 0, 0, 1, 2, 2, 3, 2, 2, 1, 2, 2, 2, 2, 2, 3, 2, 2, 0, 1, 2, 0, 2, 1, 1,
0, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 1, 2, 2, 0, 2, 2, 0, 3, 1, 3, 2, 2, 1, 1, 0, 0, 2, 0, 3, 2,
2, 1, 3, 2, 1, 2, 2, 2, 2, 2, 1, 2, 2, 3, 2, 3, 1, 3, 3, 1, 2, 3, 1, 3, 2, 1, 2, 2, 0, 0,
0, 1, 2, 1, 1, 3, 1, 1, 3, 3, 2, 2, 0, 3, 0, 1, 2, 3, 1, 0, 3, 1, 2, 1, 3, 2, 0, 0, 0, 0,
1, 2, 0, 0, 0, 2, 0, 0, 2, 3, 2, 1, 2, 1, 3, 2, 3, 0, 0, 2, 3, 0, 3, 3, 0, 1, 1, 0, 2, 1,
2, 1, 0, 1, 3, 3, 1, 2, 2, 1, 0, 2, 1, 2, 2, 1, 0, 2, 1, 0, 1, 1, 3, 1, 2, 2, 1, 0, 2, 1,
0, 1, 1, 3, 1, 0, 2, 1, 0, 2, 1, 0, 0, 1, 0, 3, 2, 1, 1, 1, 0, 2, 1, 0, 0, 2, 0, 0, 2, 0,
2, 3, 2, 2, 0, 0, 2, 0, 0, 2, 1, 2, 2, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 3, 0, 0, 0, 0,
2, 2, 3, 2, 2, 2, 3, 2, 3, 3, 1, 0, 1, 2, 1, 3, 2, 0, 0, 3, 0, 0, 3, 1, 1, 3, 3, 1, 1, 2,
0, 0, 2, 0, 1, 2, 0, 2, 1, 2, 1, 0, 0, 2, 0, 0, 0, 0, 3, 0, 1, 2, 2, 2, 0, 2, 1, 3, 1, 1,
1, 0, 0, 3, 1, 3, 1, 1, 1, 3, 0, 3, 3, 3, 2, 0, 3, 3, 0, 3, 3, 3, 3, 0, 3, 3, 2, 3, 3, 2,
2, 1, 2, 0, 2, 2, 0, 0, 2, 2, 3, 0, 0, 3, 2, 0, 2, 2, 0, 2, 2, 2, 1, 1, 2, 0, 0, 1, 0, 1,
1, 2, 1, 0, 1, 1, 3, 0, 1, 0, 2, 2, 2, 2, 3, 3, 1, 2, 1, 3, 0, 0, 3, 3, 3, 3, 2, 3, 2, 0,
0, 2, 0, 0, 2, 1, 0, 0, 0, 1, 3, 3, 3, 3, 0, 0, 3, 0, 0, 0, 2, 3, 2, 0, 0, 2, 3, 2, 2, 3,
0, 3, 3, 3, 3, 2, 2, 3, 2, 1, 1, 1, 2, 1, 3, 2, 1, 1, 0, 1, 0, 3, 1, 2, 0, 2, 0, 0, 0, 2,
1, 2, 0, 0, 0, 2, 2, 0, 0, 1, 0, 2, 0, 3, 1, 0, 2, 1, 0, 2, 0, 0, 3, 0, 0, 0, 2, 0, 0, 3,
0, 3, 3, 2, 1, 0, 2, 3, 0, 0, 0, 2, 0, 0, 2, 2, 1, 0, 0, 1, 3, 3, 0, 1, 3, 2, 0, 3, 0, 2,
0, 0, 3, 2, 3, 2, 2, 0, 2, 1, 3, 1, 1, 3, 0, 2, 0, 3, 1, 3, 1, 0, 0, 2, 2, 0, 1, 0, 0, 1,
2, 2, 0, 2, 3, 2, 0, 1, 1, 3, 1, 3, 1, 3, 0, 1, 3, 2, 1, 3, 2, 3, 2, 0, 2, 3, 0, 1, 1, 3,
3, 2, 3, 3, 2, 2, 0, 2, 0, 2, 1, 2, 2, 2, 0, 2, 3, 1, 3, 2, 2, 3, 2, 0, 1, 1, 2, 3, 3, 2,
1, 0, 2, 0, 2, 1, 0, 2, 1, 0, 1, 1, 1, 3, 2, 3, 0, 0, 1, 2, 3, 3, 3, 2, 3, 1, 0, 2, 0, 0,
0, 3, 3, 3, 1, 1, 2, 1, 2, 2, 2, 1, 3, 2, 2, 1, 3, 2, 0, 0, 1, 3, 3, 3, 3, 2, 0, 0, 0, 2,
3, 2, 0, 2, 1, 2, 2, 2, 0, 0, 0, 0, 3, 2, 1, 0, 2, 0, 0, 2, 1, 2, 3, 2, 0, 3, 3, 2, 2, 0,
0, 2, 0, 1, 2, 0, 0, 3, 2, 3, 0, 1, 0, 1, 2, 3, 1, 0, 3, 3, 2, 3, 2, 2, 2, 2, 1, 1, 0, 1,
1, 3, 2, 2, 2, 3, 2, 3, 2, 2, 1, 0, 0, 0, 0, 2, 1, 0, 0, 0, 3, 2, 2, 2, 1, 3, 2, 1, 3, 0,
0, 3, 3, 3, 3, 2, 1, 0, 2, 0, 1, 1, 1, 2, 2, 0, 0, 0, 1, 1, 0, 1, 0, 3, 0, 1, 3, 2, 2, 0,
0, 0, 1, 1, 0, 1, 1, 3, 0, 2, 0, 0, 0, 1, 0, 0, 2, 3, 2, 2, 3, 2, 2, 2, 0, 3, 2, 2, 3, 3,
0, 1, 1, 0, 3, 2, 2, 3, 2, 0, 0, 2, 0, 0, 2, 3, 2, 2, 3, 3, 2, 3, 3, 0, 3, 3, 2, 0, 3, 2,
0, 1, 3, 3, 3, 3, 0, 3, 2, 2, 1, 3, 2, 2, 1, 3, 2, 1, 1, 2, 3, 2, 2, 2, 0, 3, 2, 0, 3, 1,
3, 0, 1, 3, 2, 0, 2, 0, 1, 3, 1, 3, 2, 3, 2, 0, 3, 1, 2, 0, 3, 0, 3, 1, 1, 3, 3, 3, 2, 0,
1, 3, 2, 3, 3, 2, 0, 2, 0, 1, 3, 0, 0, 0, 2, 2, 3, 2, 2, 0, 0, 1, 3, 2, 3, 0, 1, 1, 3, 3,
3, 3, 3, 3, 2, 2, 1, 1, 1, 2, 1, 0, 2, 3, 0, 3, 3, 1, 3, 2, 0, 3, 3, 0, 1, 1, 0, 2, 1, 0,
0, 3, 1, 0, 2, 0, 1, 1, 1, 1, 2, 3, 3, 2, 2, 0, 0, 3, 2, 2, 3, 0, 1, 3, 1, 1, 3, 1, 0, 0,

```
1, 3, 2, 1, 3, 2, 3, 1, 1, 1, 0, 2, 1, 3, 2, 3, 0, 2, 0, 0, 2, 1, 3, 1, 3, 1, 3, 0, 3, 1,
2, 2, 0, 2, 2, 0, 3, 3, 0, 1, 3, 3, 1, 1, 3, 3, 2, 2, 3, 0, 3, 3, 3, 3, 2, 2, 0, 0, 2, 0,
0, 3, 2, 1, 3, 0, 1, 0, 2, 0, 0, 1, 0, 0, 3, 1, 1, 0, 1, 2, 2, 0, 2, 2, 0, 0, 2, 2, 2, 2,
2, 1, 1, 0, 2, 3, 3, 1, 2, 3, 1, 0, 1, 1, 1, 3, 3, 3, 1, 1, 1, 1, 1, 1, 1, 0, 3, 2, 1, 1,
1, 3, 2, 0, 0, 3, 3, 3, 1, 1, 0, 3, 0, 3, 2, 0, 0, 0, 3, 0, 0, 0, 3, 3, 0, 1, 3, 2, 0}};
```

Explanation of samples

Note that in these samples there is no particular ordering
Also, if samples have roughly same number of opposite base pairs but significant differences between the number of each base pair, i.e. one sample has roughly 1400 A (0), 500 T (3), whereas the other has 500 A (0), 1400 T (3), we can reasonably conclude that the respective authors/sources read simply switched the base pairs and make the necessary adjustment 0<->3 and/or 1<->2

Porcine circovirus type 1 used (There are different types of viruses under this same species)
Only samples with 1759 base pairs used here

According to tile of research cited in (https://www.ncbi.nlm.nih.gov/nuccore/KJ408798.1) there is a "Lack of genetic diversity in newly sequenced porcine circovirus type 1 strains isolated 20 years apart" based on similarities between samples $PCV1_2$ and $PCV1_3$, which were both examined in that paper.

$PCV1_1$ is from GenBank: JX566507.1
$PCV1_2$ is from GenBank: KJ408798.1
$PCV1_3$ is from GenBank: KJ408799.1 (different samples found 20 yeas apart, just from the same authors, so filed at same time)
$PCV1_4$ is from GenBank: AY754015.1 (Australian strain, so *might* differ a bit more) (***Note:*** we modified this one from it's GenBank source by moving a 46 letter gene which was at the end to the front, as explained below)

```
(PCV1₁ - PCV1₂)[[1, All]]; (* Gives list of difference beween each base pair*)
```

```
Count[(PCV1₁ - PCV1₂)[[1, All]], Except[0]]
(* Number of differences between PCV1₁ and PCV1₂ *)
10
```

```
NumDifferencesPCV1[i_, j_] := Count[(PCV1ᵢ - PCV1ⱼ)[[1, All]], Except[0]]
 (* Number of differences between PCV1ᵢ and PCV1ⱼ *)
Table[NumDifferencesPCV1[i, j], {i, 1, 4}, {j, 1, 4}] // MatrixForm
```

$$\begin{pmatrix} 0 & 10 & 10 & 1324 \\ 10 & 0 & 4 & 1325 \\ 10 & 4 & 0 & 1322 \\ 1324 & 1325 & 1322 & 0 \end{pmatrix}$$

This was the number of differences before we modified $PCV1_4$ as explained below

The above table tells us that 1,2, and 3 are nearly the same whereas 4 is drastically different. Although this difference
Use Reverse function to flip $PCV1_4$ to make sure it's not just tanscribed in reverse

The matrix will have zeroes only on the diagnols if damples are only the same as themselves

Can see that ctrl+F on the sequence 1, 3, 0, 0, 3, 0, 1, 2, 2, 3, 0, 2, 3, 0, 3, 3, 2, 2, 0, 0, 0, 2 shows it can be found in both $PCV1_3$ and $PCV1_4$, so it being reversed is not likely the issue

Searching for 1,1,3,3,1,1 shows this sequence in the same location in the first 3 PCV1 samples but 46 places earlier in $PCV1_4$, and further inspection shows that the first 3 samples start off with a 46 letter apparent gene {0,1,1,0,2,1,2,1,0,1,3,3,1,2,2,1,0,2,1,2,2,1,0,2,1,0,1,1,3,1,2,2,1,0,2,1,2,3,1,0,2,3,2,0,0,0} which appears at the very end of $PCV1_4$
Therefore we modify $PCV1_4$ from its GenBank source, placing this 46 letter gene in front to match the other samples

```
Length[{0, 1, 1, 0, 2, 1, 2, 1, 0, 1, 3, 3, 1, 2, 2, 1, 0, 2, 1, 2,
  2, 1, 0, 2, 1, 0, 1, 1, 3, 1, 2, 2, 1, 0, 2, 1, 2, 3, 1, 0, 2, 3, 2, 0, 0, 0}]
```

46

```
Table[NumDifferencesPCV1[i, j], {i, 1, 4}, {j, 1, 4}] // MatrixForm
```

$$\begin{pmatrix} 0 & 10 & 10 & 8 \\ 10 & 0 & 4 & 3 \\ 10 & 4 & 0 & 4 \\ 8 & 3 & 4 & 0 \end{pmatrix}$$

From this we can see that this modification of $PCV1_4$ is has an ordering of genes in alignment with the rest of the samples

Numerically Sorted Samples

One advantage of numerically sorting the samples is that we can get a sense of the difference between two samples without having to worry about what order the authors of the sample source trancribed the genetic code in - sorting the genes becomes impractical for larger genomes
Another is that large differences can tell us

```
Do[NumPCV1ᵢ = Sort[PCV1ᵢ[[1, All]]], {i, 1, 4}]
```

```
Table[NumPCV1ᵢ, {i, 1, 4}];
```

```
NumPCV1₁ – NumPCV1₂;
```

```
Table[Count[(NumPCV1ᵢ – NumPCV1ⱼ), Except[0]], {i, 1, 4}, {j, 1, 4}] // MatrixForm
```

$$\begin{pmatrix} 0 & 5 & 3 & 4 \\ 5 & 0 & 2 & 1 \\ 3 & 2 & 0 & 1 \\ 4 & 1 & 1 & 0 \end{pmatrix}$$

We can see from this that the numerically sorted samples differ roughly as much (in fact in all these cases less) as the non-sorted samples. So if needed the numerical sortings should give a good estimate of the differences between samples.

Random data compression

For a given k, the following green cell will create a vector of length 2^k composed of random digits from 0 to 4 and
will find the W_i and output a graph showing the number of singular values of the i-th SVD
Will show the total number of singular values after the graph

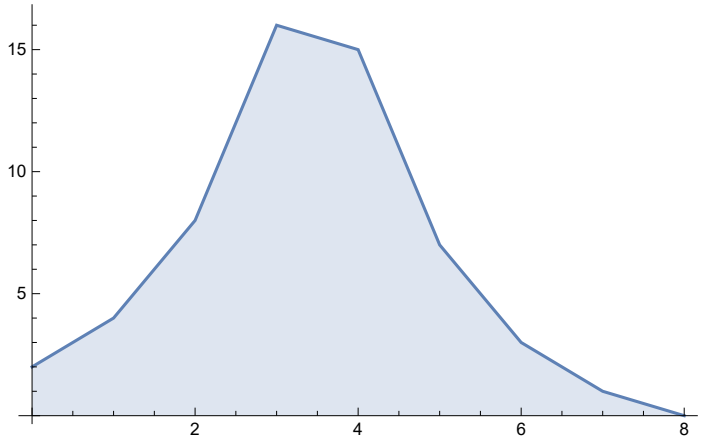
```
k = 10;

RandomDataVector[p_] := {RandomInteger[{0, 4}, 2^p]}
(* RandomDataVector[k] gives a random vector list of length 2^k, Specify k above *)

W0 = Table[
    Flatten[RandomDataVector[k]][[((2^k/2) * (k1 - 1)) + k2]], {k1, 1, 2}, {k2, 1, 2^k/2}];
WP[i_, j_, b_] := Diagonal[√SingularValueDecomposition[N[Wb]][[2]]][[i]]
    (SingularValueDecomposition[N[Wb]][[3]][[i, j]]);
WK = Table[0, {i, 1, 2^k}, {j, 1, 2^k}];
xeven[i_] := 2 (i - 1) + 2;
yeven[j_] := ((j - 1))/2 + 1/2;
xodd[i_] := 2 (i - 1) + 1;
yodd[j_] := ((j - 1))/2 + 1;

For[b = 0, b < k, b++, Do[WK[[xodd[i], yodd[j]]] = WP[i, j, b], {j, 1, 2^k/(2^(b + 2))}, 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}] &&
  Do[WK[[xeven[i], yeven[j]]] = WP[i, j, b], {j, 2, 2^k/(2^(b + 2))}, 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}];
 Wb+1 =
  Table[
   WK[[i, j]]
   , {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]},
    {j, 2, 2^k/(2^(b + 2))}];
 WK = Table[0, {i, 1, 2^k}, {j, 1, 2^k}]; If[b > (k - 3), Break[]]
]

ListLinePlot[
 Table[{i, MatrixRank[SingularValueDecomposition[N[Wi]][[2]]]}, {i, 0, k - 2}],
 Filling → Axis]
Total[Table[MatrixRank[SingularValueDecomposition[N[Wi]][[2]]], {i, 0, k - 2}]] 1
```



56

```
Do[Print[SingularValueDecomposition[N[W_i]][[2]] // MatrixForm], {i, 0, 8}]
```

$$\begin{pmatrix} 71.2499 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 32.2251 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 5.50506 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 3.05262 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 2.85076 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 1.60344 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 2.22469 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 1.50854 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 1.30992 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 1.08659 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 1.00666 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0.916234 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0.749385 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0.5613 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

$$\begin{pmatrix} 1.45457 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 1.09793 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 1.01858 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0.898454 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0.823178 & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0.79035 & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0.726843 & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0.688203 & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0.614348 & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0.585168 & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0.49051 \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{pmatrix}$$

1.24747	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	1.1549	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	1.11993	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	1.04567	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.953775	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.91826	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.850258	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.743561	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.679367	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.641871	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.608605
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

1.38794	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	1.3232	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	1.28659	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	1.25492	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	1.09687	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	1.06587	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.829666	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

```
⎛1.58829   0.        0.      0.  0.  0.  0. ⎞
   0.    1.53092     0.      0.  0.  0.  0.
   0.      0.     0.922305  0.  0.  0.  0.
   0.      0.        0.      0.  0.  0.  0.
   0.      0.        0.      0.  0.  0.  0.
   0.      0.        0.      0.  0.  0.  0.
   0.      0.        0.      0.  0.  0.  0.
   0.      0.        0.      0.  0.  0.  0.
   0.      0.        0.      0.  0.  0.  0.
   0.      0.        0.      0.  0.  0.  0.
   0.      0.        0.      0.  0.  0.  0.
   0.      0.        0.      0.  0.  0.  0.
   0.      0.        0.      0.  0.  0.  0.
⎝  0.      0.        0.      0.  0.  0.  0. ⎠
⎛0.960751  0.  0. ⎞
    0.     0.  0.
    0.     0.  0.
    0.     0.  0.
    0.     0.  0.
⎝   0.     0.  0. ⎠
⎛0.⎞
⎝0.⎠
```

Importantly this shows the general pattern of compressibility for random data and look like much of what we've gotten for when we used DNA samples.

Also show the peaks are generally wider for even k

Compare

Misc.

```
imax = 2;
Clear[M] (*Clears M since we defined M = H1Avec earlier *)
M = {Flatten[Table[PCV1i, {i, 1, imax}]]}; (*Gives the PCV1i as a
 single vector M of length 1759*4 (or however many i) and in order of i,
so the first 1759 digits are PCV11, the second 1759 digits PCV12, and so on so forth *)
(*outer {} brackets used since the function to get the W uses double layer M *)
lengthvec[M_] := Length[M[[1, All]]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^(npow)), Break[]];
  FilledSize = 2^(npow + 1)];
Filler[M_] := Table[4, {i, 1, FilledSize - lengthvec[M]}]
FilledVec[M_] := Join[Flatten[M], Filler[M]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^npow), Break[]]];
(* gives npow such that  2^npow > lengthvec[M] > 2^(npow -1) *)
FilledSize = 2^npow;
FilledSize
npow
```

```
2048
```

```
11
```

```
imax = 4;
lengthvec[M_] := Length[M[[1, All]]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^(npow)), Break[]];
  FilledSize = 2^(npow + 1)];
Filler[M_] := Table[4, {i, 1, FilledSize - lengthvec[M]}]
FilledVec[M_] := Join[Flatten[M], Filler[M]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[PCV1_1] < (2^npow), Break[]]];
(* gives npow such that  2^npow > lengthvec[M] > 2^(npow -1) *)
FilledSize = 2^npow;
Do[W0_i = Table[Flatten[FilledVec[PCV1_i]][[((FilledSize/2) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}], {i, 1, 4}]
(2^12) - (1759 * 2)
(2^13) - (1759 * 3)
(2^13) - (1759 * 4)
```

578

2915

1156

FilledSize should be the power of 2 immediatedly above 1759 * imax ,
which in this case (imax = 3 or 4) is 2^13 = 8192
so it has 8192 - 7036 = 1156 filler terms for imax = 4,
2915 filler terms for imax = 3, and 578 filler terms for imax = 2

```
W0 = Table[Flatten[FilledVec[M]][[((FilledSize/2) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}];
WP[i_, j_, b_] := Diagonal[√SingularValueDecomposition[N[Wb]][[2]]][[i]]
   (SingularValueDecomposition[N[Wb]][[3]][[i, j]])
WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];
xeven[i_] := 2 (i - 1) + 2
yeven[j_] := ((j - 1))/2 + 1/2
xodd[i_] := 2 (i - 1) + 1
yodd[j_] := ((j - 1))/2 + 1
For[b = 0, b < npow, b++, Do[WK[[xodd[i], yodd[j]]] = WP[i, j, b], {j, 1, FilledSize/(2^(b + 2))}, 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}] &&
   Do[WK[[xeven[i], yeven[j]]] = WP[i, j, b], {j, 2, FilledSize/(2^(b + 2))}, 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}];
 Wb+1 =
  Table[
   WK[[i, j]]
   , {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]},
    {j, 2, FilledSize/(2^(b + 2))}];
 WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];
 If[b > npow - 3, Break[]]
]
(* ;
If[b>npow-3,Break[]] part stops the For function at b=npow,
and prevent the error message and saves some comp power *)
```

```
Do[Print[{i, MatrixRank[SingularValueDecomposition[N[Wi]][[2]]]}], {i, 0, 8}]
Sum[MatrixRank[SingularValueDecomposition[N[Wi]][[2]]], {i, 0, 8}]
```

```
{0, 2}

{1, 4}

{2, 8}

{3, 16}

{4, 31}

{5, 15}

{6, 7}

{7, 3}

{8, 1}
87

If[Mod[y, 3] = 0,

Mod[y, 3]

Piecewise[
 {{W0₁[[1, y]], Mod[y, 3] = 0}, {W0₂[[1, y]], Mod[y, 3] = 1}, {W0₃[[1, y]], Mod[y, 3] = 2}}]
(* Piecewise[Table[{W0ᵢ[[1,y]],Mod[y,imax]=(i-1)},{i,1,imax}]] *)

W0₁;

imax = 4;
lengthvec[M_] := Length[M[[1, All]]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^(npow)), Break[]];
  FilledSize = 2^(npow + 1)];
Filler[M_] := Table[4, {i, 1, FilledSize - lengthvec[M]}]
FilledVec[M_] := Join[Flatten[M], Filler[M]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[PCV1₁] < (2^npow), Break[]]];
(* gives npow such that  2^npow > lengthvec[M] > 2^(npow -1) *)
FilledSize = 2^npow;
Do[W0ᵢ = Table[Flatten[FilledVec[PCV1ᵢ]][[((FilledSize/2) * (k1 - 1)) + k2]],
   {k1, 1, 2}, {k2, 1, FilledSize/2}], {i, 1, imax}]
(* each W0 is a 2x1024 matrix *)

(*Wtot[y_]:=Piecewise[Table[{W0ᵢ[[1,y]],Mod[y,imax]==(i-1)},{i,1,imax}]]*)

(* Table[Wtot[y],{y,1,FilledSize*imax}]*)

imax = 4;

(* imax=4;
Table[PCV1(Mod[y,imax]+1)[[1,y]],{y,1,1759*imax}]*)
```

```
PCV1(Mod[3,imax]+1)[[1, 3]]
(Mod[4, imax] + 1)
```

1

1

Working Stuff (Which is run to get the results in the following sections)

```
(* Table[PCV(Mod[y,imax]+1)[[1,Floor[y+(imax-1)/imax]]],{y,1,1759*imax}]*)
```

```
Table[PCV(Mod[y,imax]+1)[[1, Floor[(y + (imax - 1))/imax]]], {y, 1, 1759 * imax}]
```

{PCV$_2$[[1, 1]], PCV$_3$[[1, 1]], PCV$_4$[[1, 1]], PCV$_1$[[1, 1]], PCV$_2$[[1, 2]], PCV$_3$[[1, 2]], PCV$_4$[[1, 2]], PCV$_1$[[1, 2]], PCV$_2$[[1, 3]], PCV$_3$[[1, 3]], PCV$_4$[[1, 3]], PCV$_1$[[1, 3]], PCV$_2$[[1, 4]], PCV$_3$[[1, 4]], PCV$_4$[[1, 4]], PCV$_1$[[1, 4]], PCV$_2$[[1, 5]], ··· 7003 ···, PCV$_2$[[1, 1756]], PCV$_3$[[1, 1756]], PCV$_4$[[1, 1756]], PCV$_1$[[1, 1756]], PCV$_2$[[1, 1757]], PCV$_3$[[1, 1757]], PCV$_4$[[1, 1757]], PCV$_1$[[1, 1757]], PCV$_2$[[1, 1758]], PCV$_3$[[1, 1758]], PCV$_4$[[1, 1758]], PCV$_1$[[1, 1758]], PCV$_2$[[1, 1759]], PCV$_3$[[1, 1759]], PCV$_4$[[1, 1759]], PCV$_1$[[1, 1759]]}

large output | show less | show more | show all | set size limit...

^This lets you viualize what the catenation is doing and verify that it's catenated as desired (PCV not a defined item)

Order of catenation of PCV1$_i$'s doesn't really matter

```
imax = 4;
M = {Table[PCV1(Mod[y,imax]+1)[[1, Floor[(y + (imax - 1))/imax]]], {y, 1, 1759 * imax}]}
 (* = Wtot *);
lengthvec[M_] := Length[M[[1, All]]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^(npow)), Break[]];
  FilledSize = 2^(npow + 1)];
lengthvec[M_] := Length[M[[1, All]]]
Filler[M_] := Table[4, {i, 1, FilledSize - lengthvec[M]}]
FilledVec[M_] := Join[Flatten[M], Filler[M]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^npow), Break[]]];
(* gives npow such that  2^npow > lengthvec[M] > 2^(npow -1) *)
FilledSize = 2^npow;
FilledSize
npow
```

8192

13

```
lengthvec[M]
```

7036

```
W0 = Table[Flatten[FilledVec[M]][[(((FilledSize/2)) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}];
WP[i_, j_, b_] := Diagonal[√SingularValueDecomposition[N[Wb]][[2]]][[i]]
   (SingularValueDecomposition[N[Wb]][[3]][[i, j]])
WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];
xeven[i_] := 2 (i - 1) + 2
yeven[j_] := ((j - 1))/2 + 1/2
xodd[i_] := 2 (i - 1) + 1
yodd[j_] := ((j - 1))/2 + 1
For[b = 0, b < npow, b++, Do[WK[[xodd[i], yodd[j]]] = WP[i, j, b], {j, 1, FilledSize/(2^(b + 2))}, 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}] &&
  Do[WK[[xeven[i], yeven[j]]] = WP[i, j, b], {j, 2, FilledSize/(2^(b + 2))}, 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}];
 Wb+1 =
  Table[
   WK[[i, j]]
   , {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]},
    {j, 2, FilledSize/(2^(b + 2))}];
 WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];
 If[b > (npow - 3), Break[]]
]
```

```
$Aborted
```

```
Do[Print[{i, MatrixRank[SingularValueDecomposition[N[Wi]][[2]]]}], {i, 0, 8}]
Sum[MatrixRank[SingularValueDecomposition[N[Wi]][[2]]], {i, 0, 8}]
```

```
Do[Print[SingularValueDecomposition[N[Wi]][[2]] // MatrixForm], {i, 0, 9}]
```

```
Length[W0]
```

```
2
```

```
W0 = Table[Flatten[FilledVec[M]][[(((FilledSize/2)) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}];
```

Compression of $PCV1_1$ alone

```
imax = 1;
M = PCV1₁ (* = Wtot *);

lengthvec[M_] := Length[M[[1, All]]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^(npow)), Break[]];
  FilledSize = 2^(npow + 1)];
lengthvec[M_] := Length[M[[1, All]]]
Filler[M_] := Table[4, {i, 1, FilledSize - lengthvec[M]}]
FilledVec[M_] := Join[Flatten[M], Filler[M]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^npow), Break[]]];
(* gives npow such that  2^npow > lengthvec[M] > 2^(npow -1) *)
FilledSize = 2^npow;
FilledSize
npow
```

```
2048
```

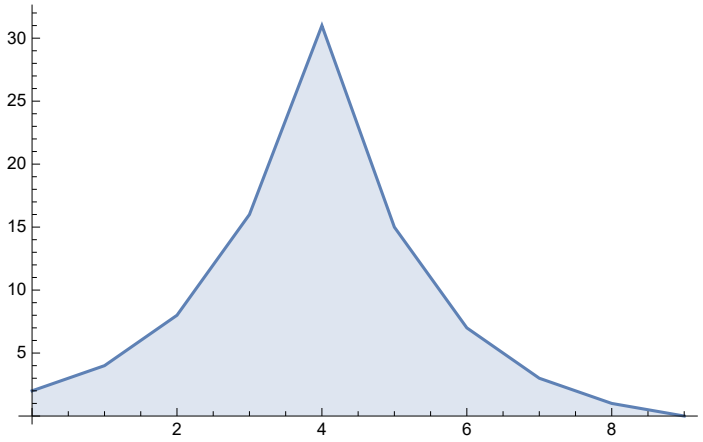
```
11
```

```
W0 = Table[Flatten[FilledVec[M]][[((FilledSize/2) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}];
WP[i_, j_, b_] := Diagonal[√SingularValueDecomposition[N[Wb]][[2]]][[i]]
   (SingularValueDecomposition[N[Wb]][[3]][[i, j]])
WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];
xeven[i_] := 2 (i - 1) + 2
yeven[j_] := ((j - 1))/2 + 1/2
xodd[i_] := 2 (i - 1) + 1
yodd[j_] := ((j - 1))/2 + 1
For[b = 0, b < npow, b++, Do[WK[[xodd[i], yodd[j]]] = WP[i, j, b], {j, 1, FilledSize/(2^(b + 2))}, 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}] &&
  Do[WK[[xeven[i], yeven[j]]] = WP[i, j, b], {j, 2, FilledSize/(2^(b + 2))}, 2},
    {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}];
 Wb+1 =
  Table[
   WK[[i, j]]
   , {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]},
    {j, 2, FilledSize/(2^(b + 2))}];
 WK = Table[0, {i, 1, FilledSize}, {j, 1, FilledSize}];
 If[b > (npow - 3), Break[]]
]
```

```
Do[Print[{i, MatrixRank[SingularValueDecomposition[N[Wi]][[2]]]}], {i, 0, 9}]
Sum[MatrixRank[SingularValueDecomposition[N[Wi]][[2]]], {i, 0, 9}]
```

```
{0, 2}
{1, 4}
{2, 8}
{3, 16}
{4, 31}
{5, 15}
{6, 7}
{7, 3}
{8, 1}
{9, 0}
87
```

```
ListLinePlot[{{0, 2}, {1, 4}, {2, 8}, {3, 16}, {4, 31},
  {5, 15}, {6, 7}, {7, 3}, {8, 1}, {9, 0}}, Filling → Axis] (*For PCV1₁ *)
```




```
Do[Print[SingularValueDecomposition[N[Wᵢ]][[2]] // MatrixForm], {i, 0, 9}]
```

```
(99.2494    0.    0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
   0.    37.8095 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.)

(6.1971    0.       0.       0.    0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
   0.    3.24933    0.       0.    0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
   0.       0.    2.54888    0.    0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
   0.       0.       0.    1.46952 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.)

(2.33258    0.       0.       0.       0.       0.       0.       0.    0. 0. 0. 0. 0. 0. 0. 0.
   0.    1.46076    0.       0.       0.       0.       0.       0.    0. 0. 0. 0. 0. 0. 0. 0.
   0.       0.    1.22052    0.       0.       0.       0.       0.    0. 0. 0. 0. 0. 0. 0. 0.
   0.       0.       0.    1.18142    0.       0.       0.       0.    0. 0. 0. 0. 0. 0. 0. 0.
   0.       0.       0.       0.    1.05893    0.       0.       0.    0. 0. 0. 0. 0. 0. 0. 0.
   0.       0.       0.       0.       0.    0.961439    0.       0.    0. 0. 0. 0. 0. 0. 0. 0.
   0.       0.       0.       0.       0.       0.    0.827335    0.    0. 0. 0. 0. 0. 0. 0. 0.
   0.       0.       0.       0.       0.       0.       0.    0.491553 0. 0. 0. 0. 0. 0. 0. 0.)
```

```
1.39185   0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.    1.04372     0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.     0.953086     0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.        0.     0.888239     0.        0.        0.        0.        0.        0.        0.
  0.      0.        0.        0.     0.860395     0.        0.        0.        0.        0.        0.
  0.      0.        0.        0.        0.     0.814587     0.        0.        0.        0.        0.
  0.      0.        0.        0.        0.        0.     0.756434     0.        0.        0.        0.
  0.      0.        0.        0.        0.        0.        0.     0.696454     0.        0.        0.
  0.      0.        0.        0.        0.        0.        0.        0.     0.684212     0.        0.
  0.      0.        0.        0.        0.        0.        0.        0.        0.     0.610928     0.
  0.      0.        0.        0.        0.        0.        0.        0.        0.        0.     0.5989
  0.      0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
```

```
1.09042   0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.    1.02746    0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.     0.9764      0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.     0.876828     0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.     0.850346     0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.     0.798076     0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.     0.784222     0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.     0.751198     0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.     0.734433     0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.     0.700406     0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.     0.677336
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.        0.        0.
```

1.22086	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	1.19514	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	1.17651	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	1.12434	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	1.10896	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	1.09777	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	1.01791	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.982002	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.955217	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.923904	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.905792	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	

```
1.48335  0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       1.45927  0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       1.41193  0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       1.39794  0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       1.38784  0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       1.33588  0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.804299  0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
0.       0.       0.       0.       0.       0.       0.        0. 0. 0. 0. 0. 0. 0. 0.
```

```
1.68224  0.       0.        0. 0. 0. 0.
0.       1.6674   0.        0. 0. 0. 0.
0.       0.       0.897547  0. 0. 0. 0.
0.       0.       0.        0. 0. 0. 0.
0.       0.       0.        0. 0. 0. 0.
0.       0.       0.        0. 0. 0. 0.
0.       0.       0.        0. 0. 0. 0.
0.       0.       0.        0. 0. 0. 0.
0.       0.       0.        0. 0. 0. 0.
0.       0.       0.        0. 0. 0. 0.
0.       0.       0.        0. 0. 0. 0.
0.       0.       0.        0. 0. 0. 0.
0.       0.       0.        0. 0. 0. 0.
0.       0.       0.        0. 0. 0. 0.
```

```
0.947401 0. 0.
0.       0. 0.
0.       0. 0.
0.       0. 0.
0.       0. 0.
0.       0. 0.
```

```
0.
0.
```

Computation of 4-sample concatenated PCV1 data set

Results of this computation given here, don't want to re-run the algorithm each time as it takes too long, but used the bluebox algorithm to get each SVD

```
Do[Print[{i, MatrixRank[SingularValueDecomposition[N[W_i]][[2]]]}], {i, 0, 11}]
Sum[MatrixRank[SingularValueDecomposition[N[W_i]][[2]]], {i, 0, 11}]
```

```
{0, 2}

{1, 3}
```

```
{2, 6}
{3, 12}
{4, 24}
{5, 48}
{6, 31}
{7, 15}
{8, 7}
{9, 3}
{10, 1}
{11, 0}
152
Do[Print[SingularValueDecomposition[N[Wᵢ]][[2]] // MatrixForm], {i, 0, 11}]
```

```
198.399    0.    0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0
  0.    75.6176 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0
```

```
9.68151    0.       0.     0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
  0.    4.67464    0.     0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.    0.248665 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
```

```
3.02234    0.       0.        0.       0.       0.     0. 0. 0. 0. 0. 0. 0. 0. 0.
  0.    2.07149    0.        0.       0.       0.     0. 0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.    0.714158     0.       0.       0.     0. 0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.     0.609858    0.       0.     0. 0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.        0.     0.37813    0.     0. 0. 0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.        0.       0.    0.321271 0. 0. 0. 0. 0. 0. 0. 0. 0.
```

```
1.22866    0.       0.        0.        0.        0.        0.        0.        0.
  0.    1.08213    0.        0.        0.        0.        0.        0.        0.
  0.      0.    0.600588     0.        0.        0.        0.        0.        0.
  0.      0.       0.     0.556009     0.        0.        0.        0.        0.
  0.      0.       0.        0.     0.538503     0.        0.        0.        0.
  0.      0.       0.        0.        0.     0.451034     0.        0.        0.
  0.      0.       0.        0.        0.        0.     0.412115     0.        0.
  0.      0.       0.        0.        0.        0.        0.     0.283823     0.
  0.      0.       0.        0.        0.        0.        0.        0.     0.121389
  0.      0.       0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.
  0.      0.       0.        0.        0.        0.        0.        0.        0.
```

```
1.00794    0.         0.         0.         0.         0.         0.         0.         0.
  0.     0.913204     0.         0.         0.         0.         0.         0.         0.
  0.       0.     0.778966       0.         0.         0.         0.         0.         0.
  0.       0.         0.     0.759869       0.         0.         0.         0.         0.
  0.       0.         0.         0.     0.706675       0.         0.         0.         0.
  0.       0.         0.         0.         0.     0.64022        0.         0.         0.
  0.       0.         0.         0.         0.         0.     0.556982       0.         0.
  0.       0.         0.         0.         0.         0.         0.     0.518527       0.
  0.       0.         0.         0.         0.         0.         0.         0.     0.456209
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
  0.       0.         0.         0.         0.         0.         0.         0.         0.
```

1.15447	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.939229	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.839008	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.735192	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.657246	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.625161	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.586325	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.5617	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.526427	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	

1.10569	0.	0.	0.	0.	0.	0.	0.	0.
0.	1.02673	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.94646	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.907342	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.89577	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.799538	0.	0.	0.

0.	0.	0.	0.	0.	0.799538	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.798324	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.787103	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.778232	
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.7
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	
0.	0.	0.	0.	0.	0.	0.	0.	0.	

0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.

1.33435	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	1.23314	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	1.22141	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	1.21164	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	1.19268	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	1.14406	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	1.13676	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	1.12352	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	1.09302	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	1.04779
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

```
1.55125   0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.    1.5237     0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.    1.49366     0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.    1.47494     0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.    1.43577     0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.    1.41926     0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.    0.690518    0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
  0.      0.       0.       0.       0.       0.       0.       0. 0. 0. 0. 0. 0. 0.
```

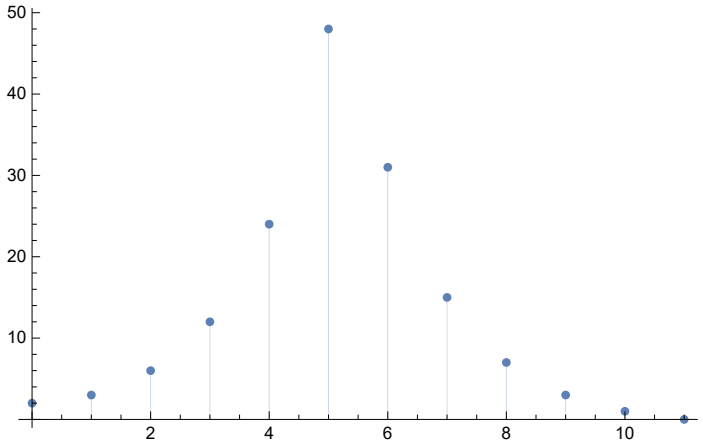
```
1.71416   0.       0.     0. 0. 0. 0.
  0.    1.70349    0.     0. 0. 0. 0.
  0.      0.    0.831061  0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0.
  0.      0.       0.     0. 0. 0. 0.
```

```
0.911629 0. 0.
   0.    0. 0.
   0.    0. 0.
   0.    0. 0.
   0.    0. 0.
   0.    0. 0.
```

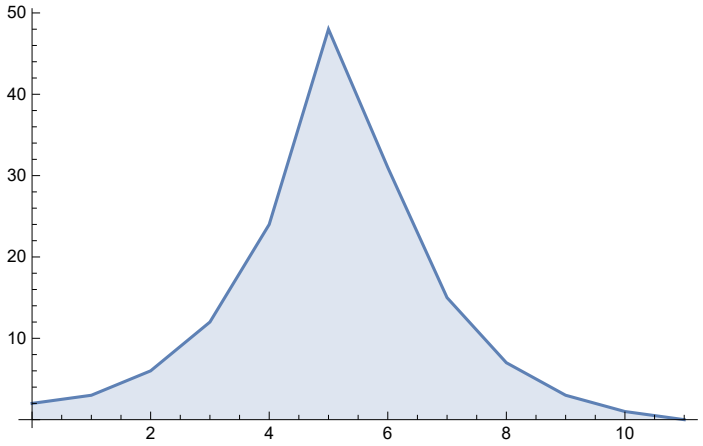
```
0.
0.
```

Results of 4-sample PCV1 SVD

```
ListPlot[{{0, 2}, {1, 3}, {2, 6}, {3, 12}, {4, 24}, {5, 48},
  {6, 31}, {7, 15}, {8, 7}, {9, 3}, {10, 1}, {11, 0}}, Filling → Axis]
```




```
ListLinePlot[{{0, 2}, {1, 3}, {2, 6}, {3, 12}, {4, 24}, {5, 48},
  {6, 31}, {7, 15}, {8, 7}, {9, 3}, {10, 1}, {11, 0}}, Filling → Axis]
```



These plots show the number of Singular values vs. the number of SVD's performed

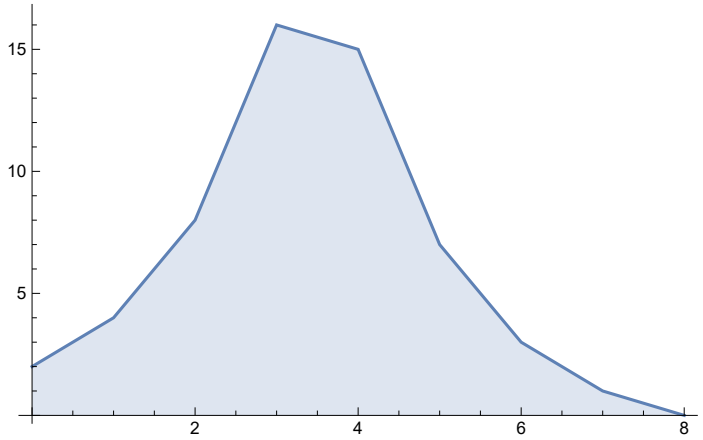
Peak number of Singular values occurs at 5th SVD

2+3+6+12+24+48 = 95 SV's occur for i = 0 to 5, before/up to the peak,

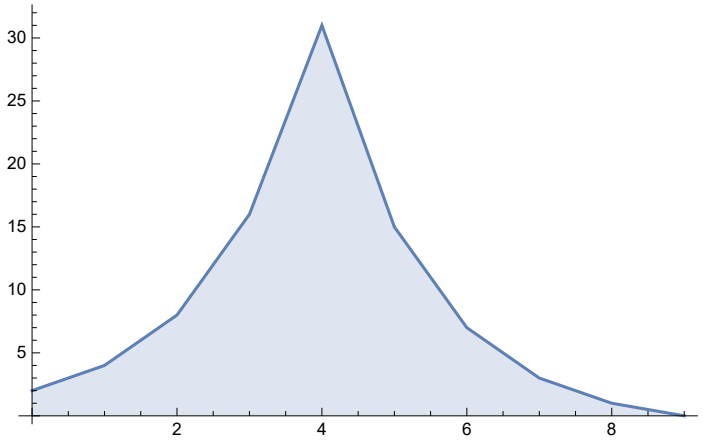
31+15+7+3+1 = 57 SV's occur for i = 6 to 10, after the peak

Compare this plot to that for H1A and $PCV1_1$ single-sample compression:

```
ListLinePlot[{{0, 2}, {1, 4}, {2, 8}, {3, 16}, {4, 15}, {5, 7}, {6, 3}, {7, 1}, {8, 0}},
 Filling → Axis] (*For H1A *)
```




```
ListLinePlot[{{0, 2}, {1, 4}, {2, 8}, {3, 16}, {4, 31},
  {5, 15}, {6, 7}, {7, 3}, {8, 1}, {9, 0}}, Filling → Axis] (*For PCV1₁ *)
```



So this method seems to impove compressibility a bit compared to the compression of H1A, but not as much for the compression of $PCV1_1$ alone

Also note that PCV1 virus lacks genetic diversity, so samples are very similar. This might be why it doesn't compress much better than $PCV1_1$ alone

Timing

Timing function gives CPU time used
AbsoluteTime gives real time used (i.e. CPU time + time it takes to display + other things))
RandomInteger[{0,4}] gives a random integer from 0 to 4 (base pairs + filler number 4)

```
AbsoluteTiming[Total[Range[123456]]]
Total[Range[123456]] // AbsoluteTiming
Total[Range[123456]] // Timing
```

```
{0.00131905, 7620753696}
```

```
{0.000312359, 7620753696}
```

```
{0., 7620753696}
```

```
RandomDataVector[k_] := {RandomInteger[{0, 4}, 2^k]}
(* RandomDataVector[k] gives a random vector list of length 2^k *)
```

```
RandomDataVector[2]
```

```
{{2, 3, 4, 3}}
```

```
RandomDataVector[k_] := {RandomInteger[{0, 4}, 2^k]}

CPUTimeItTakes[k_] := Timing[W0 = Table[
    Flatten[RandomDataVector[k]][[((2^k/2) * (k1 - 1)) + k2]], {k1, 1, 2}, {k2, 1, 2^k/2}];
  WP[i_, j_, b_] := Diagonal[√SingularValueDecomposition[N[Wb]][[2]]][[i]]
    (SingularValueDecomposition[N[Wb]][[3]][[i, j]]);
  WK = Table[0, {i, 1, 2^k}, {j, 1, 2^k}];
  xeven[i_] := 2 (i - 1) + 2;
  yeven[j_] := ((j - 1))/2 + 1/2;
  xodd[i_] := 2 (i - 1) + 1;
  yodd[j_] := ((j - 1))/2 + 1;
  For[b = 0, b < k, b++, Do[WK[[xodd[i], yodd[j]]] = WP[i, j, b], {j, 1, 2^k/(2^(b + 2))}, 2},
     {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}] &&
    Do[WK[[xeven[i], yeven[j]]] = WP[i, j, b], {j, 2, 2^k/(2^(b + 2))}, 2},
     {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}];
   Wb+1 =
    Table[
     WK[[i, j]]
     , {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]},
     {j, 2, 2^k/(2^(b + 2))}];
   WK = Table[0, {i, 1, 2^k}, {j, 1, 2^k}]; If[b > (k - 3), Break[]]
  ]]
AbsTimeItTakes[k_] := AbsoluteTiming[W0 = Table[
    Flatten[RandomDataVector[k]][[((2^k/2) * (k1 - 1)) + k2]], {k1, 1, 2}, {k2, 1, 2^k/2}];
  WP[i_, j_, b_] := Diagonal[√SingularValueDecomposition[N[Wb]][[2]]][[i]]
```

```
    (SingularValueDecomposition[N[Wb]][[3]][[i, j]]);
  WK = Table[0, {i, 1, 2^k}, {j, 1, 2^k}];
  xeven[i_] := 2 (i - 1) + 2;
  yeven[j_] := ((j - 1))/2 + 1/2;
  xodd[i_] := 2 (i - 1) + 1;
  yodd[j_] := ((j - 1))/2 + 1;

  For[b = 0, b < k, b++, Do[WK[[xodd[i], yodd[j]]] = WP[i, j, b], {j, 1, 2^k/(2^(b + 2))}, 2},
     {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}] &&
    Do[WK[[xeven[i], yeven[j]]] = WP[i, j, b], {j, 2, 2^k/(2^(b + 2))}, 2},
     {i, 1, MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]}];
   Wb+1 =
    Table[
     WK[[i, j]]
     , {i, 1, 2 MatrixRank[√SingularValueDecomposition[N[Wb]][[2]]]},
     {j, 2, 2^k/(2^(b + 2))}];
   WK = Table[0, {i, 1, 2^k}, {j, 1, 2^k}]; If[b > (k - 3), Break[]]
  ]]
```

The above blue cell defines the following functions
CPUTimeItTakes[k] calculates how much CPU time is needed to perfrom the SVD algorithm
AbsTimeItTakes[k] calculates the absolute time it takes, CPU time + other
Computer must devote CPU time to other tasks, the time of these tasks as well as the CPU time of the calculation itself are included in Absolute time
So CPU time is a more accurate measure of efficiency

ClearSystemCache[] clears anydata stored. For instance, in the cell below the first function will give around a third of a second of cpu time, whereas the second same function will give 0 since its already been done and stored in the cache. The third fn time is after a clearing of the cache so takes around the same as the first rather than 0
Note that clearing the kernel may not clear the cache, as this does.

```
Timing[N[Pi, 10^6];]
Timing[N[Pi, 10^6];]
ClearSystemCache[]
Timing[N[Pi, 10^6];]
```

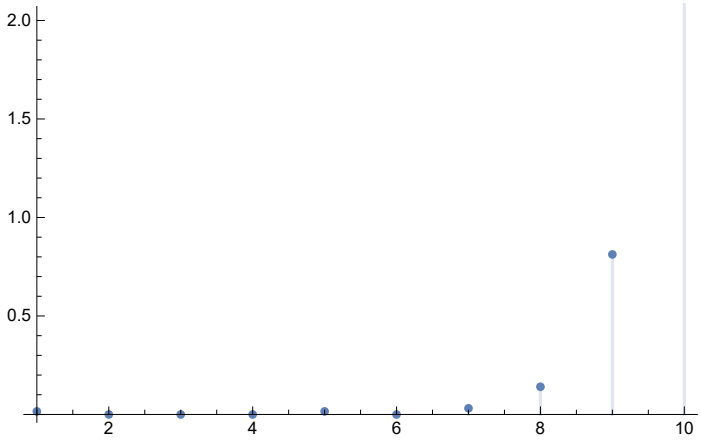
```
{0.359375, Null}
```

```
{0., Null}
```

```
{0.34375, Null}
```

```
ClearSystemCache[]
```

```
ClearSystemCache[]
DiscretePlot[CPUTimeItTakes[k][[1]], {k, 1, 10}]
```




```
2^11
```

2048

```
ClearSystemCache[]
CPUTimeItTakes[10](*Since the 10th pt goes off the graph a bit *)
CPUTimeItTakes[11]
```

$Aborted

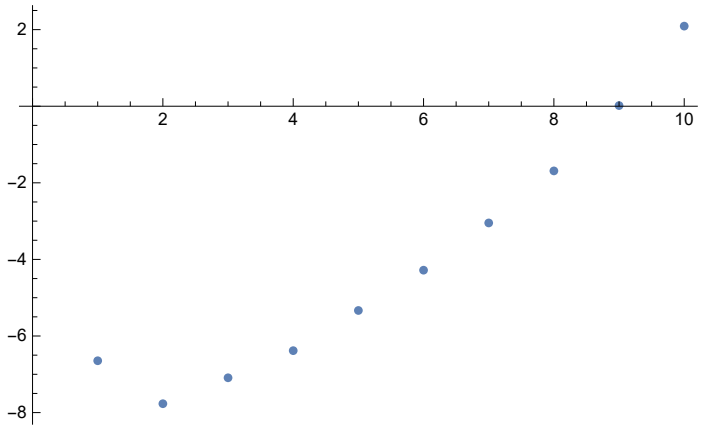
{62.5313, Null}

```
Log[AbsTimeItTakes[2][[1]]]
```

−5.2985

{0.000445335, Null}

```
ListPlot[Table[{k, Log[AbsTimeItTakes[k][[1]]]}, {k, 1, 10}]]
```



```
ClearSystemCache[]
ListLogPlot[{k, AbsTimeItTakes[k][[1]]}, {k, 1, 10}]
```

```
ListLogPlot[{10, 7.99507}, {10, 1, 10}]
```

```
ClearSystemCache[]
AbsTimeItTakes[10] (*Since the 10th pt goes off the graph a bit *)
AbsTimeItTakes[11]
```

```
{6.95672, Null}
```

```
$Aborted
```

```
ClearSystemCache[]
Do[Print[{k, CPUTimeItTakes[k][[1]]}], {k, 1, 10}]
```

```
{1, 0.}
```

```
{2, 0.}
```

```
{3, 0.015625}
```

```
{4, 0.}
```

```
{5, 0.}
```

```
{6, 0.015625}
```

```
{7, 0.03125}
```

```
{8, 0.140625}
```

```
{9, 0.765625}
```

```
{10, 5.14063}
```

```
ClearSystemCache[]
Do[Print[{k, AbsTimeItTakes[k]}], {k, 1, 10}]
```

```
{1, {0.00462604, Null}}
```

```
{2, {0.000418561, Null}}
```

```
{3, {0.000774651, Null}}
```

```
{4, {0.00166978, Null}}
```

```
{5, {0.00473314, Null}}
```

```
{6, {0.0136983, Null}}
```

```
{7, {0.0348071, Null}}
```

```
{8, {0.129583, Null}}
```

```
{9, {0.767484, Null}}
```

```
{10, {6.04451, Null}}
```

```
ClearSystemCache[]
```

```
ClearSystemCache[]
CPUTimeData = Table[{k, CPUTimeItTakes[k][[1]]}, {k, 1, 11}];
AbsTimeData = Table[{k, AbsTimeItTakes[k][[1]]}, {k, 1, 11}];
```

```
CPUTimeData =
```

```
{{1, 0.}, {2, 0.}, {3, 0.}, {4, 0.}, {5, 0.}, {6, 0.03125},
 {7, 0.03125}, {8, 0.140625}, {9, 0.78125}, {10, 5.09375}, {11, 61.4063}}
```

```
AbsTimeData =
```

```
{{1, 0.003697}, {2, 0.000312359}, {3, 0.000648815},
 {4, 0.00130923}, {5, 0.00397812}, {6, 0.010307}, {7, 0.0347317},
 {8, 0.119589}, {9, 0.864417}, {10, 6.68476}, {11, 74.3236}}
```

```
AbsFit1 = Fit[AbsTimeData, {1, x, x^2}, x]
AbsFit2 = Fit[AbsTimeData, {1, 2^x}, x]
AbsFit3 = Fit[AbsTimeData, {2^x}, x]
```

$20.5142 - 12.4783\,x + 1.3438\,x^2$

$-4.34154 + 0.031706 \times 2^x$

0.0285277×2^x

```
CPUFit1 = Fit[CPUTimeData, {1, x, x^2}, x]
CPUFit2 = Fit[CPUTimeData, {1, 2^x}, x]
CPUFit3 = Fit[CPUTimeData, {2^x}, x]
```

$16.9024 - 10.2783\,x + 1.10657\,x^2$

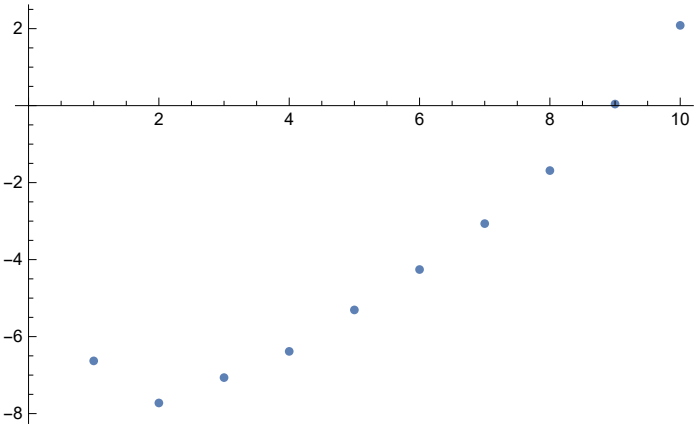
$-3.58894 + 0.0261267 \times 2^x$

0.0234994×2^x

```
Fit[Table[{k, Log[AbsTimeItTakes[k][[1]]]}, {k, 1, 10}], {1, x}, x]
```

$-9.30911 + 0.980193\,x$

```
Sow[ListPlot[Table[{k, Log[AbsTimeItTakes[k][[1]]]}, {k, 1, 10}]],
 Plot[-9.309113819687155` + 0.9801932758223911` x, {x, 0, 12}]]
```

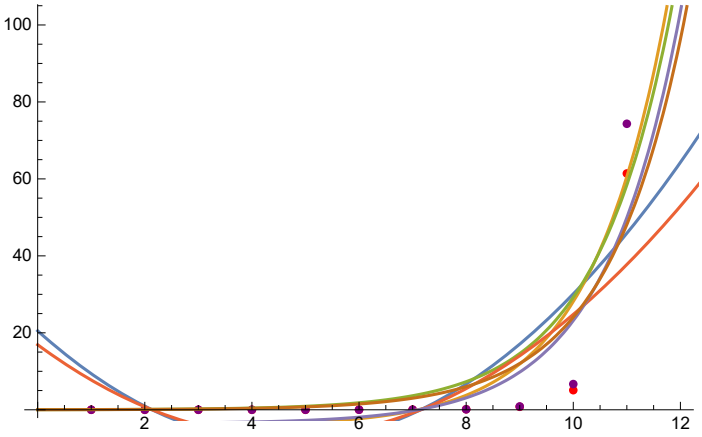


$$\frac{\text{Exp[10]}}{60} \text{ // N}$$

367.108

22

```
Show[ListPlot[CPUTimeData, PlotStyle → Red], ListPlot[AbsTimeData, PlotStyle → Purple],
 Plot[{AbsFit1, AbsFit2, AbsFit3, CPUFit1, CPUFit2, CPUFit3}, {x, 0, 13}],
 PlotRange → {{0, 12}, {0, 100}}]
```




```
CPUTimeData[[All, 2]]
```

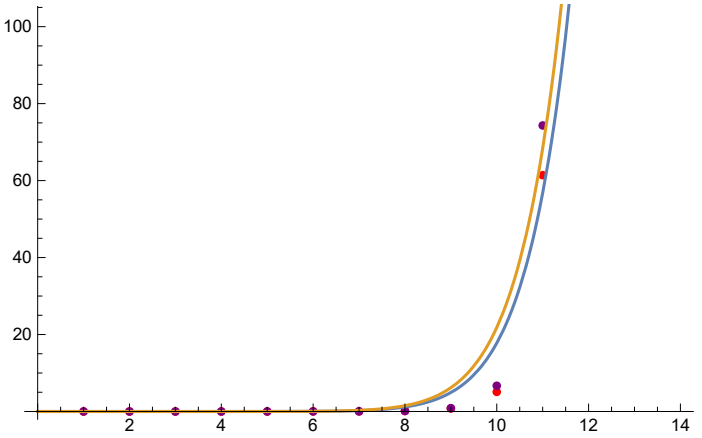
```
{0., 0., 0., 0., 0., 0.03125, 0.03125, 0.140625, 0.78125, 5.09375, 61.4063}
```

```
FindFit[CPUTimeData[[All, 2]], a (x^p), {a, p}, x]
FindFit[AbsTimeData[[All, 2]], (a) (x^p), {a, p}, x]
```

$\{a \to 1.16588 \times 10^{-11}, p \to 12.1833\}$

$\{a \to 2.30266 \times 10^{-11}, p \to 11.9769\}$

```
Show[ListPlot[CPUTimeData, PlotStyle → Red], ListPlot[AbsTimeData, PlotStyle → Purple],
 Plot[{(1.1658817354065915`*^-11) (x^12.18327618356838`),
   (2.3026592684158607`*^-11) (x^11.976886898047853`)},
  {x, 0, 100}], PlotRange → {{0, 14}, {0, 100}}]
```




```
(2.3026592684158607`*^-11) (x^11.976886898047853`) /. x → 13
(*An estimate for the clock time it'd take
 to do the SVD (with this algorithm) for 2^13 digits *)
```

```
505.596
```

```
(* AbsTimeItTakes[13] *)
(*Actual clock time to do the SVD (with this algorithm) for 2^13 digits *)
```

```
$Aborted
```

PCV1 gene data in Python/Fortran form

lists generally the same in python and fortran, so they're given by
`FortranForm[Flatten[PCV1ᵢ]] // InputForm`

```
FortranForm[Flatten[PCV1₁]] // InputForm
```

```
List(0,1,1,0,2,1,2,1,0,1,3,3,1,2,2,1,0,2,1,2,2,1,0,2,1,0,1,1,3,1,2,2,1,0,2,1,2,3,1,0,2,3,2,
  0,2,1,0,0,2,0,0,0,0,2,1,2,2,1,1,1,2,1,0,0,1,1,1,1,0,3,0,0,2,0,2,2,3,2,2,2,3,2,3,3,1,0,1,1
  3,1,1,3,3,1,1,2,0,2,2,0,2,2,0,2,0,0,0,0,0,1,0,0,0,0,3,0,1,2,2,2,0,2,1,3,3,1,1,0,0,3,1,3,1
  0,3,3,0,3,3,3,3,2,3,3,3,2,1,2,2,0,2,0,2,2,0,0,2,2,3,3,3,2,2,0,0,2,0,2,2,2,3,0,2,0,0,1,3,1
  1,0,2,2,2,2,3,3,3,2,1,2,0,0,3,3,3,3,2,1,3,0,0,2,0,0,2,1,0,2,0,1,3,3,3,3,0,0,1,0,0,2,2,3,2
  3,3,3,3,2,2,3,2,1,1,1,2,1,3,2,1,1,0,1,0,3,1,2,0,2,0,0,0,2,1,2,0,0,0,2,2,0,0,1,1,2,0,1,1,0
  0,0,2,0,0,3,0,1,3,2,1,0,2,3,0,0,0,2,0,0,2,2,1,1,0,1,0,3,0,1,3,3,0,3,1,2,0,2,3,2,3,2,2,0,2
  0,0,1,1,0,2,2,2,2,0,0,2,1,2,1,0,2,1,2,0,1,1,3,2,3,1,3,0,1,3,2,1,3,2,3,2,0,2,3,0,1,1,1,3,3
  2,2,2,2,3,1,3,3,3,2,2,3,2,0,1,3,2,3,0,2,1,1,2,0,2,1,0,2,3,3,1,1,1,3,2,3,0,0,1,2,3,0,3,2,3
  3,1,1,2,1,2,2,2,1,3,2,2,1,3,2,0,0,1,3,3,3,3,2,0,0,0,2,3,2,0,2,1,2,2,2,0,0,2,0,3,2,1,0,2,1
  3,2,2,0,0,2,0,1,0,2,1,3,2,3,0,1,0,1,2,3,1,0,3,0,2,3,2,2,2,1,1,1,2,1,1,1,2,2,3,3,2,3,2,2,2
  2,3,2,2,2,1,1,1,2,3,0,0,3,3,3,3,2,1,3,2,0,2,1,1,3,0,2,1,2,0,1,0,1,1,3,0,1,3,2,2,0,0,2,1,1
  0,3,0,0,2,3,2,2,3,2,2,2,0,3,2,2,0,3,0,3,1,0,3,2,2,0,2,0,0,2,0,0,2,3,3,2,3,3,2,3,3,3,3,2,2
  3,0,3,2,2,1,3,2,2,3,3,0,1,1,3,3,2,2,2,0,3,2,0,3,1,3,0,1,3,2,0,2,0,1,3,2,3,2,3,2,0,1,1,2,2
  2,0,1,3,2,3,0,2,0,2,0,1,3,0,0,0,2,2,2,2,2,3,0,1,3,2,3,3,1,1,3,3,3,3,3,3,2,2,1,1,1,2,1,0,2
  3,3,0,1,1,0,2,1,0,0,3,1,0,2,2,1,1,1,1,1,1,0,2,2,0,0,3,2,2,3,0,1,3,1,1,3,1,0,0,1,3,2,1,3,2
  2,3,0,2,0,0,2,1,3,1,3,1,3,0,3,1,2,2,0,2,2,0,3,3,0,1,3,0,1,3,3,3,2,1,0,0,3,3,3,3,2,2,0,0,2
  0,2,0,0,1,0,0,3,1,1,0,1,2,2,0,2,2,3,0,1,1,1,2,0,0,2,2,1,1,2,0,3,3,3,2,0,0,2,1,0,2,3,2,2,0
  2,3,2,1,1,1,3,3,3,3,1,1,1,0,3,0,3,0,0,0,0,3,0,0,0,3,3,0,1,3,2,0,2,3,1,3,3,3,3,3,3,2,3,3,0
  3,0,0,3,2,2,3,3,3,3,3,3,0,3,3,3,3,3,0,3,3,3,0,3,3,3,0,2,0,2,2,2,1,1,3,3,3,3,0,2,2,0,3,0,0,0
  0,3,3,2,3,0,1,0,3,0,0,0,3,0,2,3,1,0,2,1,1,3,3,0,1,1,0,1,0,3,0,0,3,3,3,3,2,2,2,1,3,2,3,2,2
  3,2,2,0,2,1,2,1,0,3,0,2,1,1,2,0,2,2,1,1,3,2,3,2,3,2,1,3,1,2,0,1,0,3,3,2,2,3,2,3,2,2,2,3,0
  2,0,2,1,1,0,1,0,2,1,3,2,2,3,3,3,1,3,3,3,3,3,0,3,3,0,3,3,3,2,2,2,3,2,2,0,0,1,1,0,0,3,1,0,0,3
  1,1,0,2,1,3,1,0,2,2,3,3,3,2,2,2,2,2,3,2,0,0,2,3,0,1,1,3,2,2,0,2,3,2,2,3,0,2,2,3,0,0,0,2,2
  0,3,2,2,3,2,3,2,2,1,2,2,2,0,2,2,0,2,3,0,2,3,3,0,0,3,0,3,0,2,2,2,2,3,1,0,3,0,2,2,1,1,0,0,2
  2,2,2,2,2,3,3,0,1,0,0,0,2,3,3,2,2,1,0,3,1,1,0,0,2,0,3,0,0,1,0,0,1,0,2,3,2,2,0,1,1,1,0,0,1
  2,0,3,3,0,2,0,2,2,3,2,0,3,2,2,2,2,3,1,3,1,3,2,2,2,2,3,0,0,0,0,3,3,1,0,3,0,3,3,3,0,2,1,1,3
  1,2,2,3,0,2,3,0,3,3,2,2,0,0,0,2,2,3,0,2,2,2,2,3,0,2,2,2,2,2,3,3,2,2,3,2,1,1,2,1,1,3,2,0,2
  2,0,0,1,3,2,2,1,1,2,0,3,2,3,3,3,2,0,0,3,3,3,3,2,0,2,2,3,0,2,3,3,0,0,1,0,3,3,1,1,0,0,2,0,3,2,2
  0,3,1,1,3,1,1,3,3,3,3,0,3,2,2,3,2,0,2,3,0,1,0,0,0,3,3,1,3,2,3,0,2,0,0,0,2,2,1,2,2,2,0,0,3
  1,1,1,2,3,1,3,3,3,1,2,2,1,2,1,1,0,3,1,3,2,3,0,0,1,2,2,3,3,3,1,3,2,0,0,2,2,1,2,2,2,2,3,2,3
  3,2,2,3,1,3,3,1,3,1,1,2,2,0,2,2,0,3,2,3,3,3,1,1,0,0,2,2,3,2,2,1,3,2,1,2,2,2,2,1,1,2,2,2,3
  3,2,1,2,2,3,0,0,1,2,1,1,3,1,1,3,3,2,2,1,1,0,1,2,3,1,0,3,1,1,3,0,3,0,0,0,0,2,3,2,0,0,0,2,0
  2,1,3,2,3,0,2,3,0,3,3)
```

```
Export["PCV1_1genesample.txt", Flatten[PCV1₁]]
Export["PCV1_2genesample.txt", Flatten[PCV1₂]]
Export["PCV1_3genesample.txt", Flatten[PCV1₃]]
Export["PCV1_4genesample.txt", Flatten[PCV1₄]]
```

PCV1_1genesample.txt

PCV1_2genesample.txt

PCV1_3genesample.txt

PCV1_4genesample.txt

```
M = {Join[Flatten[PCV1₁], Flatten[PCV1₂], Flatten[PCV1₃], Flatten[PCV1₄]]};
lengthvec[M_] := Length[M[[1, All]]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^(npow)), Break[]];
  FilledSize = 2^(npow + 1)];
lengthvec[M_] := Length[M[[1, All]]]
Filler[M_] := Table[4, {i, 1, FilledSize - lengthvec[M]}]
FilledVec[M_] := Join[Flatten[M], Filler[M]]
For[npow = 1, npow < 1000, npow++, If[lengthvec[M] < (2^npow), Break[]]];
(* gives npow such that  2^npow > lengthvec[M] > 2^(npow -1) *)
FilledSize = 2^npow;
FilledSize
npow
W₀ = Table[Flatten[FilledVec[M]][[((FilledSize/2) * (k1 - 1)) + k2]],
    {k1, 1, 2}, {k2, 1, FilledSize/2}];
W₀ // MatrixForm
```

The purple cell will output a first-concatenation W0, which by comparing to the python code lets us see that the python code is working (get the same W)