

# Introduction to the pageRank Package

Hongxu Ding

Department of Biomolecular Engineering and Genomics Institute,  
University of California, Santa Cruz, Santa Cruz, CA, USA

April 26, 2022

## Contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction</b>                                 | <b>1</b>  |
| 1.1      | Background . . . . .                                | 1         |
| 1.2      | Installation . . . . .                              | 2         |
| <b>2</b> | <b>PageRank Analysis</b>                            | <b>2</b>  |
| 2.1      | Temporal PageRank . . . . .                         | 2         |
| 2.2      | Multiplex PageRank . . . . .                        | 3         |
| 2.3      | Adjusting PageRank Calculations . . . . .           | 4         |
| <b>3</b> | <b>Prioritizing TFs in GRNs</b>                     | <b>5</b>  |
| 3.1      | Generating GRNs from Multi-Omics Profiles . . . . . | 5         |
| 3.2      | Filter GRNs with Expression Profiles . . . . .      | 7         |
| 3.3      | Session Information . . . . .                       | 27        |
| <b>4</b> | <b>References</b>                                   | <b>30</b> |

## 1 Introduction

### 1.1 Background

The *pageRank* package provides implementations of temporal PageRank as defined by [1], as well as multiplex PageRank as defined by [2]. As the extension of original steady-state PageRank [3,4] in temporal networks, temporal PageRank ranks nodes based on their connections that change over time. Multiplex PageRank, on the other hand, extends PageRank analysis to multiplex networks. In such networks, the same nodes might interact with one another in different layers. Multiplex PageRank is calculated according to the topology of a predefined

base network, with regular PageRank of other supplemental networks as edge weights and personalization vector.

PageRank-related approaches can be applied to prioritize key transcriptional factors (TFs) in gene regulatory networks (GRNs). Specifically, the *pageRank* package provides functions for generating temporal GRNs from corresponding static counterparts. The *pageRank* package also provides functions for converting multi-omics, e.g. gene expression, chromatin accessibility and chromosome conformation profiles to multiplex GRNs. Such temporal and multiplex GRNs can thus be used for temporal and multiplex PageRank-based TF prioritization, respectively.

## 1.2 Installation

*pageRank* requires the R version 4.0 or later, packages *BSgenome.Hsapiens.UCSC.hg19*, *TxDb.Hsapiens.UCSC.Hsapiens.org.Hs.eg.db*, *annotate*, *GenomicFeatures*, *JASPAR2018*, *TFBSTools* and *bcellViper*, to run the examples. After installing R, all required components can be obtained with:

```
if (!requireNamespace("BiocManager", quietly=TRUE)) install.packages("BiocManager")
BiocManager::install("BSgenome.Hsapiens.UCSC.hg19")
BiocManager::install("TxDb.Hsapiens.UCSC.hg19.knownGene")
BiocManager::install("org.Hs.eg.db")
BiocManager::install("annotate")
BiocManager::install("GenomicFeatures")
BiocManager::install("JASPAR2018")
BiocManager::install("TFBSTools")
BiocManager::install("bcellViper")
```

## 2 PageRank Analysis

### 2.1 Temporal PageRank

We applied `diff_graph()` to calculate temporal PageRank. This is a simplified version of temporal PageRank described by [1] by only analyzing temporally adjacent graph pairs.

```
> library(pageRank)
> set.seed(1)
> graph1 <- igraph::erdos.renyi.game(100, 0.01, directed = TRUE)
> igraph::V(graph1)$name <- 1:100
> #the 1st graph with name as vertex attributes
> set.seed(2)
> graph2 <- igraph::erdos.renyi.game(100, 0.01, directed = TRUE)
> igraph::V(graph2)$name <- 1:100
> #the 2nd graph with name as vertex attributes
> diff_graph(graph1, graph2)
```

```

IGRAPH 93b8378 DN-- 98 190 --
+ attr: name (v/c), pagerank (v/n), moi (e/n)
+ edges from 93b8378 (vertex names):
 [1] 1 ->60 2 ->15 2 ->57 3 ->10 3 ->16 3 ->84 4 ->43 5 ->20 5 ->6
[10] 5 ->72 5 ->81 5 ->91 6 ->25 6 ->50 7 ->37 7 ->67 7 ->73 7 ->8
[19] 8 ->80 9 ->90 10->26 11->6 11->100 12->70 12->82 12->92 13->3
[28] 13->48 13->51 13->61 15->74 15->77 16->85 17->31 17->32 17->3
[37] 17->50 17->58 19->17 19->96 20->23 20->79 20->87 21->41 21->4
[46] 22->4 22->41 23->57 24->61 25->66 26->34 26->39 26->72 27->2
[55] 27->43 28->98 29->95 30->84 32->49 33->10 34->16 34->99 35->8
[64] 36->17 36->33 36->45 36->53 36->77 37->33 37->54 38->6 38->1
+ ... omitted several edges

```

Differential graph graph1-graph2 will be outputed. The Differential graph has "moi (mode of interaction, 1 and -1 for interactions gained and losed in graph1, respectively)" as edge attribute. The Differential graph has "pagerank" and "name" as vertex attributes.

## 2.2 Multiplex PageRank

We applied `multiplex_page_rank()` to calculate multiplex PageRank following definition by [2].

```

> set.seed(1)
> graph1 <- igraph::erdos.renyi.game(100, 0.01, directed = TRUE)
> igraph::V(graph1)$name <- 1:100
> igraph::V(graph1)$pagerank <- igraph::page_rank(graph1)$vector
> #the base graph with pagerank and name as vertex attributes.
> set.seed(2)
> graph2 <- igraph::erdos.renyi.game(100, 0.01, directed = TRUE)
> igraph::V(graph2)$name <- 1:100
> igraph::V(graph2)$pagerank <- igraph::page_rank(graph2)$vector
> #the supplemental graph with pagerank and name as vertex attributes.
> multiplex_page_rank(graph1, graph2)

```

|             |             |             |             |             |             |
|-------------|-------------|-------------|-------------|-------------|-------------|
| 1           | 2           | 3           | 4           | 5           | 6           |
| 0.024486930 | 0.003587882 | 0.003269234 | 0.025062625 | 0.002517812 | 0.014031152 |
| 7           | 8           | 9           | 10          | 11          | 12          |
| 0.019560780 | 0.002517812 | 0.010657975 | 0.024750578 | 0.003587882 | 0.002517812 |
| 13          | 14          | 15          | 16          | 17          | 18          |
| 0.002517812 | 0.002517812 | 0.012543315 | 0.011993811 | 0.011752012 | 0.002517812 |
| 19          | 20          | 21          | 22          | 23          | 24          |
| 0.002517812 | 0.005019851 | 0.005073934 | 0.019579420 | 0.010917862 | 0.006654581 |
| 25          | 26          | 27          | 28          | 29          | 30          |

|             |             |             |             |             |             |
|-------------|-------------|-------------|-------------|-------------|-------------|
| 0.008481052 | 0.024875556 | 0.018813575 | 0.012145212 | 0.002517812 | 0.005371332 |
| 31          | 32          | 33          | 34          | 35          | 36          |
| 0.028390794 | 0.003870287 | 0.022958947 | 0.007132217 | 0.026500261 | 0.014220612 |
| 37          | 38          | 39          | 40          | 41          | 42          |
| 0.003894189 | 0.014025489 | 0.007048515 | 0.006489236 | 0.009884435 | 0.011620308 |
| 43          | 44          | 45          | 46          | 47          | 48          |
| 0.021776702 | 0.005804823 | 0.007274354 | 0.005973955 | 0.002517812 | 0.003231192 |
| 49          | 50          | 51          | 52          | 53          | 54          |
| 0.008363678 | 0.018470262 | 0.007252872 | 0.007734145 | 0.007333127 | 0.008132101 |
| 55          | 56          | 57          | 58          | 59          | 60          |
| 0.002517812 | 0.009882306 | 0.012570845 | 0.005099961 | 0.009773330 | 0.005728022 |
| 61          | 62          | 63          | 64          | 65          | 66          |
| 0.008887585 | 0.009392001 | 0.002517812 | 0.012318772 | 0.002517812 | 0.012403356 |
| 67          | 68          | 69          | 70          | 71          | 72          |
| 0.003894189 | 0.008046953 | 0.006637398 | 0.012164635 | 0.004952221 | 0.025846022 |
| 73          | 74          | 75          | 76          | 77          | 78          |
| 0.007717015 | 0.017071807 | 0.004497441 | 0.031878419 | 0.006205317 | 0.006125093 |
| 79          | 80          | 81          | 82          | 83          | 84          |
| 0.007674159 | 0.004657952 | 0.036708345 | 0.004133414 | 0.003587882 | 0.008317756 |
| 85          | 86          | 87          | 88          | 89          | 90          |
| 0.019805589 | 0.003587882 | 0.010071696 | 0.003779210 | 0.002517812 | 0.010708381 |
| 91          | 92          | 93          | 94          | 95          | 96          |
| 0.009826976 | 0.006014406 | 0.020117463 | 0.010635582 | 0.006048082 | 0.004657952 |
| 97          | 98          | 99          | 100         |             |             |
| 0.012988768 | 0.015761377 | 0.004243860 | 0.003249976 |             |             |

Multiplex PageRank values corresponded to nodes in graph1 (base network) will be outputed.

### 2.3 Adjusting PageRank Calculations

The `clean_graph()` can remove nodes by residing subgraph sizes, vertex names and PageRank values. We thus can adjust graphs for PageRank calculation.

```
> set.seed(1)
> graph <- igraph::erdos.renyi.game(100, 0.01, directed = TRUE)
> igraph::V(graph)$name <- 1:100
> igraph::V(graph)$pagerank <- igraph::page_rank(graph)$vector
> #the graph to be cleaned, with pagerank and name as vertex attributes.
> clean_graph(graph, size=5)
```

```
IGRAPH d44f647 DN-- 82 96 -- Erdos-Renyi (gnp) graph
+ attr: name (g/c), type (g/c), loops (g/l), p (g/n), name (v/n),
| pagerank (v/n)
```

```

+ edges from d44f647 (vertex names):
 [1] 72-> 1 88-> 3 22-> 4 11-> 6 65-> 6 87-> 6 60-> 7 85->
 [9] 84-> 9 33-> 10 100-> 10 11->100 2-> 15 40-> 15 3-> 16 34->
[17] 19-> 17 46-> 17 5-> 20 69-> 20 100-> 20 92-> 21 27-> 22 83->
[25] 42-> 24 6-> 25 10-> 26 74-> 27 94-> 27 43-> 31 36-> 33 38->
[33] 59-> 35 90-> 35 60-> 36 70-> 36 53-> 38 26-> 39 46-> 40 88->
[41] 21-> 41 71-> 41 49-> 42 65-> 42 77-> 42 87-> 43 100-> 43 52->
[49] 21-> 45 54-> 46 32-> 49 92-> 49 6-> 50 17-> 50 43-> 52 54->
+ ... omitted several edges

```

Adjusted graph will be outputted, with "pagerank" and "name" as vertex attributes.

The `adjust_graph()` can re-calculate PageRank with updated damping factor, personalized vector and edge weights.

```

> set.seed(1)
> graph <- igraph::erdos.renyi.game(100, 0.01, directed = TRUE)
> igraph::V(graph)$name <- 1:100
> igraph::V(graph)$pagerank <- igraph::page_rank(graph, damping=0.85)$vect
> #the graph to be adjusted, with pagerank and name as vertex attributes.
> adjust_graph(graph, damping=0.1)

```

```

IGRAPH 382adda DN-- 100 98 -- Erdos-Renyi (gnp) graph
+ attr: name (g/c), type (g/c), loops (g/l), p (g/n), name (v/n),
| pagerank (v/n)
+ edges from 382adda (vertex names):
 [1] 72-> 1 88-> 3 22-> 4 11-> 6 65-> 6 87-> 6 60-> 7 85->
 [9] 84-> 9 33-> 10 100-> 10 11->100 2-> 15 40-> 15 3-> 16 34->
[17] 19-> 17 46-> 17 5-> 20 69-> 20 100-> 20 92-> 21 27-> 22 83->
[25] 42-> 24 6-> 25 10-> 26 74-> 27 94-> 27 63-> 30 43-> 31 36->
[33] 38-> 35 59-> 35 90-> 35 60-> 36 70-> 36 53-> 38 26-> 39 46->
[41] 88-> 40 21-> 41 71-> 41 49-> 42 65-> 42 77-> 42 87-> 43 100->
[49] 52-> 44 21-> 45 54-> 46 32-> 49 92-> 49 6-> 50 17-> 50 13->
+ ... omitted several edges

```

Adjusted graph will be outputted, with updated "pagerank" and "name" as vertex attributes.

Please note `diff_graph()`, `multiplex_page_rank()`, `clean_graph()` and `adjust_graph()` can be used in combination for customized PageRank analysis tasks.

## 3 Prioritizing TFs in GRNs

### 3.1 Generating GRNs from Multi-Omics Profiles

The `aracne_network()` can re-format ARACNe network in regulon object for PageRank analysis. It can also handle GRNs reverse engineered using other algorithms, as long as such

GRNs are written in regulon object.

```
> library(bcellViper)
> data(bcellViper)
> head(aracne_network(regulon[1:10]))
```

|   | reg  | target   | direction |
|---|------|----------|-----------|
| 1 | AATF | SAMM50   | 1         |
| 2 | AATF | DRG1     | 1         |
| 3 | AATF | ATIC     | 1         |
| 4 | AATF | SMARCC1  | 1         |
| 5 | AATF | AHCY     | 1         |
| 6 | AATF | HSD17B10 | 1         |

The `accessibility_network()` can build network from accessibility, e.g. ATAC-Seq peaks.

```
> table <- data.frame(Chr=c("chr1", "chr1"), Start=c(713689, 856337), End=
+                      row.names=c("A", "B"), stringsAsFactors=FALSE)
> regulators=c("FOXF2", "MZF1")
> #peaks and regulators to be analyzed
>
> library(GenomicRanges)
> library(GenomicFeatures)
> library(TxDb.Hsapiens.UCSC.hg19.knownGene)
> library(org.Hs.eg.db)
> library(annotate)
> promoter <- promoters(genes(TxDb.Hsapiens.UCSC.hg19.knownGene))
> names(promoter) <- getSYMBOL(names(promoter), data="org.Hs.eg")
> promoter <- promoter[!is.na(names(promoter))]
> #get promoter regions
>
> library(JASPAR2018)
> library(TFBSTools)
> library(motifmatchr)
> pfm <- getMatrixSet(JASPAR2018, list(species="Homo sapiens"))
> pfm <- pfm[unlist(lapply(pfm, function(x) name(x))) %in% regulators]
> #get regulator position frequency matrix (PFM) list
>
> library(BSgenome.Hsapiens.UCSC.hg19)
> accessibility_network(table, promoter, pfm, "BSgenome.Hsapiens.UCSC.hg19")

```

|   | target       | reg   |
|---|--------------|-------|
| 1 | LOC100288069 | FOXF2 |

```

2 LOC100288069 MZF1
3   LINC02593 FOXF2
4     SAMD11 FOXF2
5   LINC02593 MZF1
6     SAMD11 MZF1

```

The `conformation_network()` can build network from conformation, e.g. HiChIP records.

```

> table <- data.frame(Chr1=c("chr1", "chr1"), Position1=c(569265, 713603),
+                    Chr2=c("chr4", "chr1"), Position2=c(206628, 715110),
+                    row.names=c("A", "B"), stringsAsFactors=FALSE)
> regulators=c("FOXF2", "MZF1")
> #peaks and regulators to be analyzed
>
> promoter <- promoters(genes(TxDb.Hsapiens.UCSC.hg19.knownGene))
> names(promoter) <- getSYMBOL(names(promoter), data="org.Hs.eg")
> promoter <- promoter[!is.na(names(promoter))]
> #get promoter regions
>
> pfm <- getMatrixSet(JASPAR2018, list(species="Homo sapiens"))
> pfm <- pfm[unlist(lapply(pfm, function(x) name(x))) %in% regulators]
> #get regulator position frequency matrix (PFM) list
>
> conformation_network(table, promoter, pfm, "BSgenome.Hsapiens.UCSC.hg19")

```

```

      target  reg
1     ZNF876P MZF1
2 LOC100288069 FOXF2
3 LOC100288069 MZF1

```

### 3.2 Filter GRNs with Expression Profiles

The `P_graph()` can filter GRNs by quantifying joint and margin probability distributions of regulator-target pairs. Statistically significant non-random regulator-target pairs will be kept.

```

> dset <- exprs(dset)
> net <- do.call(rbind, lapply(1:10, function(i, regulon){
+   data.frame(reg=rep(names(regulon)[i], 10),
+             target=names(regulon[[i]][[1]])[1:10],
+             stringsAsFactors = FALSE)}, regulon=regulon))
> P_graph(dset, net, method="difference", null=NULL, threshold=0.05)

```

|  
|  
|=  
|=  
|=  
|=|=  
|=|=|  
|=|=|  
|=|=|=|  
|=|=|=|=|  
|=|=|=|=|  
|=|=|=|=|=|  
|=|=|=|=|=|  
|=|=|=|=|=|=|  
|=|=|=|=|=|=|  
|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|=|  
|=|=|=|=|=|=|=|=|





A table with 10 columns and 10 rows. Each cell contains a vertical bar on the left and a horizontal line of equals signs on the right. The equals signs are arranged in a grid pattern, starting from the top-left cell and ending at the bottom-right cell.

A series of 15 horizontal lines, each consisting of a solid top line, a dashed middle line, and a solid bottom line, typical of primary school handwriting practice paper.

A series of 18 horizontal dashed lines, each preceded by a vertical bar on the left side, serving as a template for text entry.

|  
|  
|  
|=  
|  
|=  
|  
|==  
|  
|===  
|  
|====  
|  
|=====  
|  
|=====

|  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|  
 |  
 |-----|  
 |  
 |=====|



Handwriting practice lines consisting of 20 rows. Each row is defined by a vertical line on the left and a vertical line on the right. Between these lines are three horizontal lines: a solid top line, a dashed middle line, and a solid bottom line.



|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=|  
|  
|=|  
|  
|==|  
|  
|===|



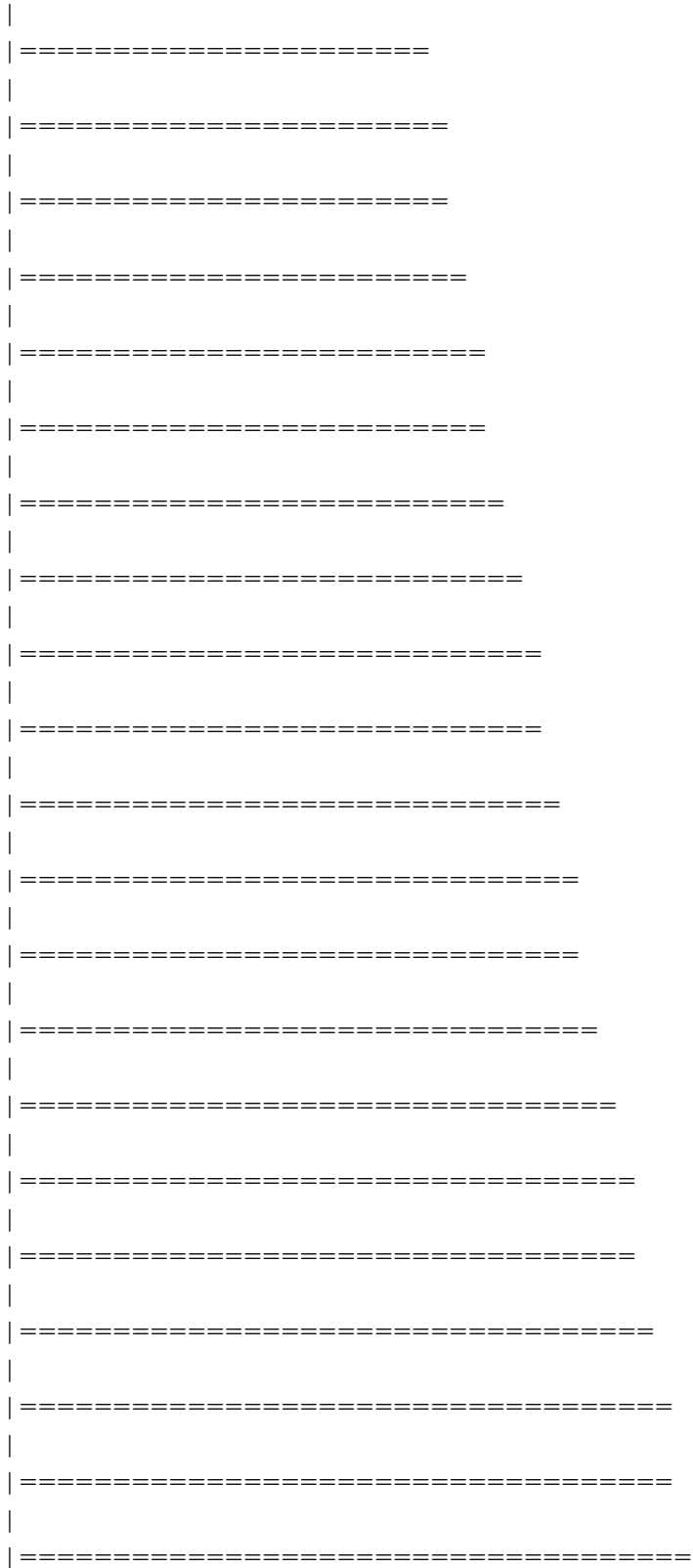
|  |       |
|--|-------|
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |
|  |       |
|  | ===== |



Handwriting practice lines consisting of a vertical line on the left and a dashed horizontal line for tracing. The lines are arranged in a series of 18 rows, each starting with a vertical line and followed by a dashed horizontal line.

|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=====|  
|  
|-----|  
|  
|=|  
|  
|=|  
|  
|==|  
|  
|===|  
|  
|====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|

|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|  
|  
|=====|





Handwriting practice lines consisting of a vertical line on the left, a dashed midline, and a solid baseline. The page contains 18 such sets of lines, providing a guide for letter height and placement.

Handwriting practice lines consisting of 18 sets of three horizontal lines (top, middle, bottom) separated by vertical bars on the left and right sides.

```

|
|=====
|
|=====
|
|=====
|
|=====
|
|=====
|
|=====
|
+ attr: name (v/c), pagerank (v/n), pvalue (e/n)
+ edges from e886629 (vertex names):
[1] PPM1G ->AATF   CTBP2 ->APP   TAGLN ->APP   MTSS1 ->APP   JMJD1C->AR

```

### 3.3 Session Information

```
> sessionInfo()
```

```
R version 4.2.0 RC (2022-04-19 r82224)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 20.04.4 LTS
```

```
Matrix products: default
```

```
BLAS: /home/biocbuild/bbs-3.15-bioc/R/lib/libRblas.so
```

```
LAPACK: /home/biocbuild/bbs-3.15-bioc/R/lib/libRlapack.so
```

```
locale:
```

```
[1] LC_CTYPE=en_US.UTF-8      LC_NUMERIC=C
[3] LC_TIME=en_GB            LC_COLLATE=C
[5] LC_MONETARY=en_US.UTF-8  LC_MESSAGES=en_US.UTF-8
[7] LC_PAPER=en_US.UTF-8     LC_NAME=C
[9] LC_ADDRESS=C             LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
```

```
attached base packages:
```

```
[1] stats4      stats      graphics  grDevices  utils      datasets  methods
[8] base
```

```
other attached packages:
```

```
[1] BSgenome.Hsapiens.UCSC.hg19_1.4.3
[2] BSgenome_1.64.0
```

[3] rtracklayer\_1.56.0  
[4] Biostrings\_2.64.0  
[5] XVector\_0.36.0  
[6] motifmatchr\_1.18.0  
[7] TFBSTools\_1.34.0  
[8] JASPAR2018\_1.1.1  
[9] annotate\_1.74.0  
[10] XML\_3.99-0.9  
[11] org.Hs.eg.db\_3.15.0  
[12] TxDb.Hsapiens.UCSC.hg19.knownGene\_3.2.2  
[13] GenomicFeatures\_1.48.0  
[14] AnnotationDbi\_1.58.0  
[15] GenomicRanges\_1.48.0  
[16] GenomeInfoDb\_1.32.0  
[17] IRanges\_2.30.0  
[18] S4Vectors\_0.34.0  
[19] bcellViper\_1.31.0  
[20] Biobase\_2.56.0  
[21] BiocGenerics\_0.42.0  
[22] pageRank\_1.6.0

loaded via a namespace (and not attached):

[1] bitops\_1.0-7 matrixStats\_0.62.0  
[3] DirichletMultinomial\_1.38.0 bit64\_4.0.5  
[5] filelock\_1.0.2 progress\_1.2.2  
[7] httr\_1.4.2 tools\_4.2.0  
[9] utf8\_1.2.2 R6\_2.5.1  
[11] seqLogo\_1.62.0 DBI\_1.1.2  
[13] colorspace\_2.0-3 prettyunits\_1.1.1  
[15] tidyselect\_1.1.2 curl\_4.3.2  
[17] bit\_4.0.4 compiler\_4.2.0  
[19] cli\_3.3.0 xml2\_1.3.3  
[21] DelayedArray\_0.22.0 caTools\_1.18.2  
[23] scales\_1.2.0 readr\_2.1.2  
[25] rappdirs\_0.3.3 digest\_0.6.29  
[27] stringr\_1.4.0 Rsamtools\_2.12.0  
[29] R.utils\_2.11.0 pkgconfig\_2.0.3  
[31] MatrixGenerics\_1.8.0 dbplyr\_2.1.1  
[33] fastmap\_1.1.0 rlang\_1.0.2  
[35] RSQLite\_2.2.12 BiocIO\_1.6.0  
[37] generics\_0.1.2 BiocParallel\_1.30.0  
[39] gtools\_3.9.2 dplyr\_1.0.8

|      |                        |                             |
|------|------------------------|-----------------------------|
| [41] | R.oo_1.24.0            | RCurl_1.98-1.6              |
| [43] | magrittr_2.0.3         | GO.db_3.15.0                |
| [45] | GenomeInfoDbData_1.2.8 | Matrix_1.4-1                |
| [47] | Rcpp_1.0.8.3           | munsell_0.5.0               |
| [49] | fansi_1.0.3            | lifecycle_1.0.1             |
| [51] | R.methodsS3_1.8.1      | stringi_1.7.6               |
| [53] | yaml_2.3.5             | SummarizedExperiment_1.26.0 |
| [55] | zlibbioc_1.42.0        | BiocFileCache_2.4.0         |
| [57] | plyr_1.8.7             | grid_4.2.0                  |
| [59] | blob_1.2.3             | parallel_4.2.0              |
| [61] | crayon_1.5.1           | CNer_1.32.0                 |
| [63] | lattice_0.20-45        | hms_1.1.1                   |
| [65] | KEGGREST_1.36.0        | pillar_1.7.0                |
| [67] | igraph_1.3.1           | rjson_0.2.21                |
| [69] | biomaRt_2.52.0         | reshape2_1.4.4              |
| [71] | TFMPvalue_0.0.8        | glue_1.6.2                  |
| [73] | vctrs_0.4.1            | png_0.1-7                   |
| [75] | tzdb_0.3.0             | gtable_0.3.0                |
| [77] | powerLaw_0.70.6        | purrr_0.3.4                 |
| [79] | assertthat_0.2.1       | cachem_1.0.6                |
| [81] | ggplot2_3.3.5          | xtable_1.8-4                |
| [83] | restfulr_0.0.13        | pracma_2.3.8                |
| [85] | tibble_3.1.6           | GenomicAlignments_1.32.0    |
| [87] | memoise_2.0.1          | ellipsis_0.3.2              |

## 4 References

1. Rozenshtein, Polina, and Aristides Gionis. "Temporal pagerank." Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Cham, 2016.
2. Halu, Arda, et al. "Multiplex pagerank." PloS one 8.10 (2013).
3. Brin, Sergey, and Lawrence Page. "The anatomy of a large-scale hypertextual web search engine." (1998).
4. Page, Lawrence, et al. The pagerank citation ranking: Bringing order to the web. Stanford InfoLab, 1999.