

# Pathway Enrichment with KNIME

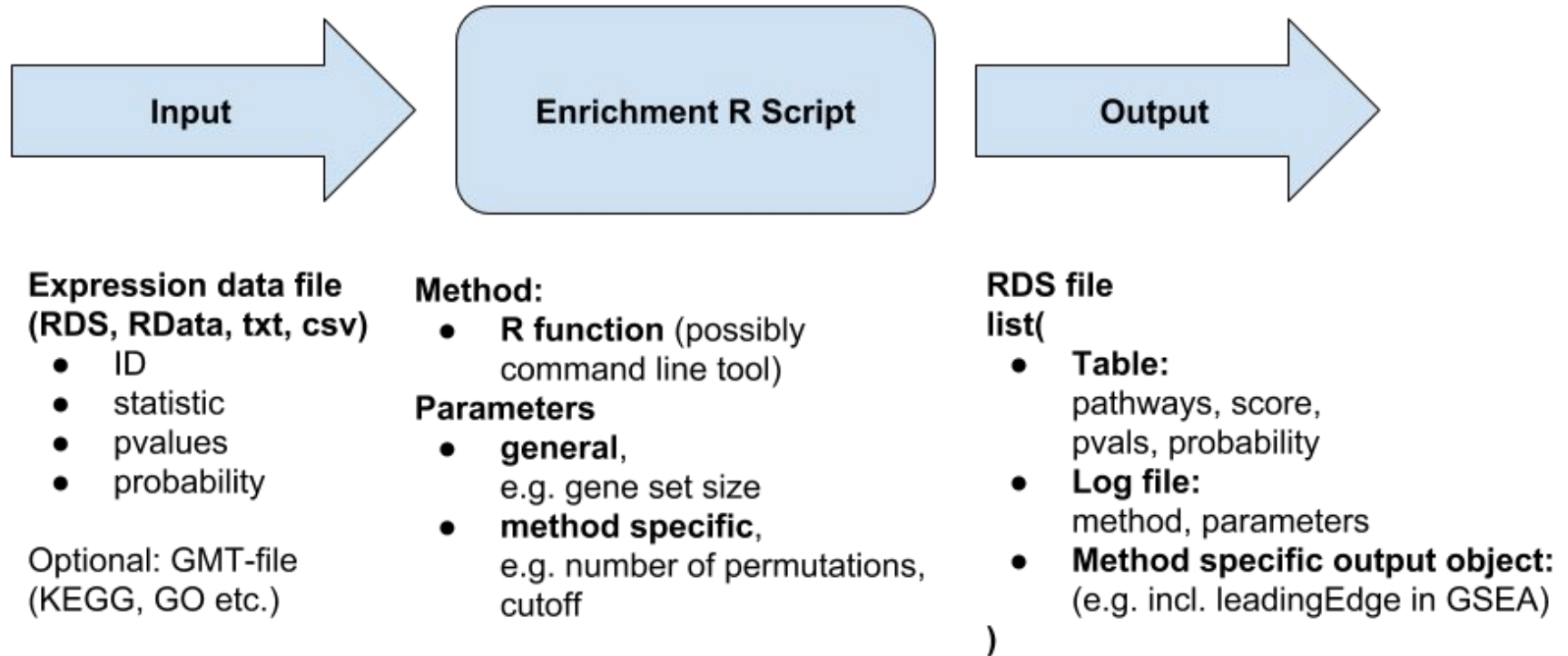
Outline: <https://github.com/InesAssum/EnrichmentNodes>

1. Pathway Enrichment using modular R library
2. Pathway Enrichment with KNIME for Users
3. Pathway Enrichment with KNIME for Developers



# Modular R library integrating multiple Pathway Enrichment methods

# Outline: modular framework



**Current methods include:** MONA, GSEA (...more coming soon!)

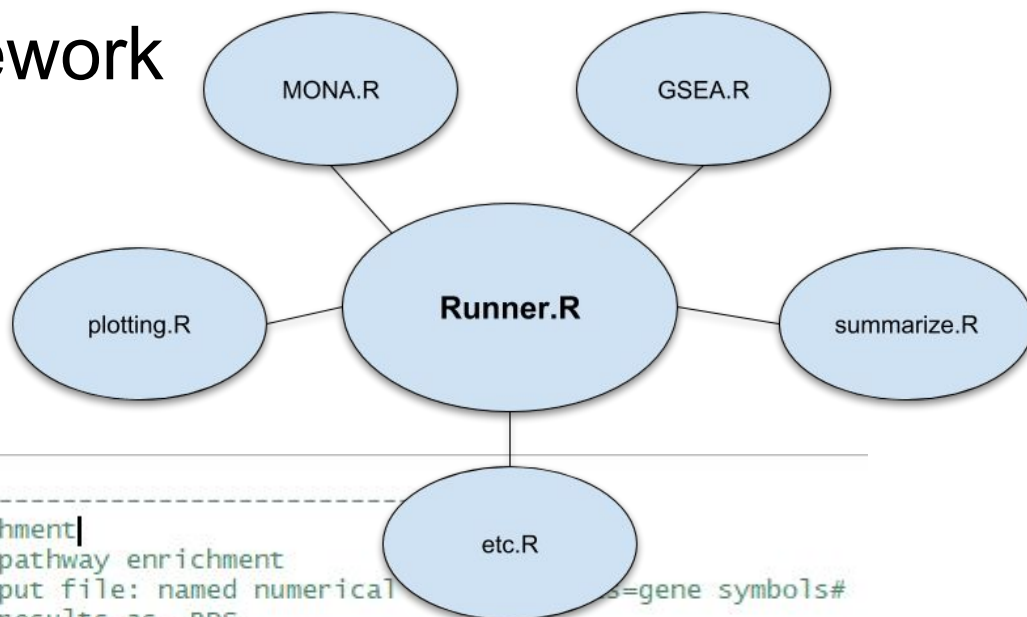
**Suited for:**

- big/advanced projects (simulations, benchmarking)
- anyone, who is using R anyway

How to deal with differences in methods?

# Outline: modular framework

- full code available on GitHub
- run locally in R or
- use docker support



```
#!/usr/bin/env Rscript
#
# Runner script for multiOMICS pathway enrichment
# @param method string // select method for pathway enrichment
# @param data string // file path to .RDS input file: named numerical matrix, gene symbols#
# @param result string // file path to save results as .RDS
# @param gmt string // pathway definition to use: KEGG, GO... default: KEGG
# @param minSize integer // min size of GSS to be considered, default: 15
# @param maxSize integer // max size of GSS to be considered, default: 500
# @param nperm integer // number of permutations, default: 10000
# @param mygmt string // custom .gmt file // not required
#
# -----
# C:\Users\krisg\Documents>Rscript run_Peanalysis.R -m fgsea -input data.csv -o result.csv -g KEGG
# [1] "Processing....."
# [1] "Completed successful! See log file for more information."
# [1] "combinatorial enrichment: 266 top hits for 106 pathways"
```

...but what if you don't have a programming background ?

# KNIME for Users

# KNIME Analytics Platform



KNIME is a free software for interactive data analysis

The screenshot displays the KNIME Analytics Platform interface. On the left, the **Node Repository** window is open, showing a tree view of nodes under the **IO** category. The **Read** sub-category is expanded, listing various file readers. An arrow points from the **File Reader** node in this list to **Node 4** in the main workflow canvas.

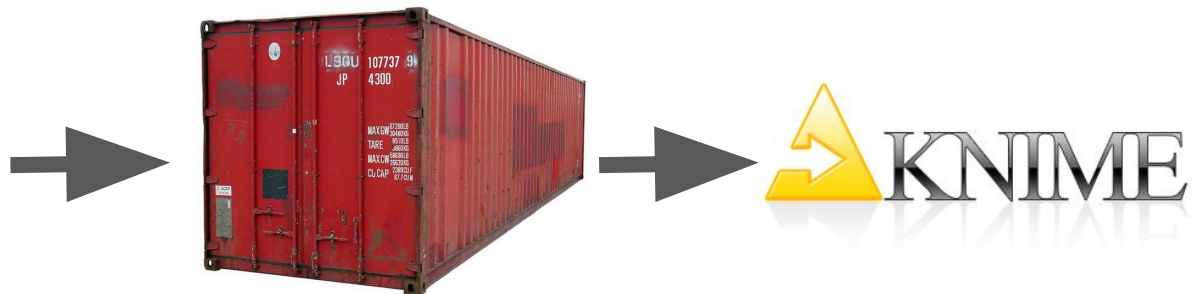
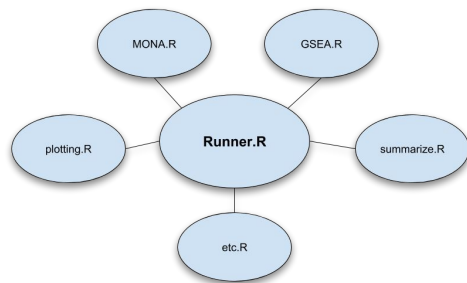
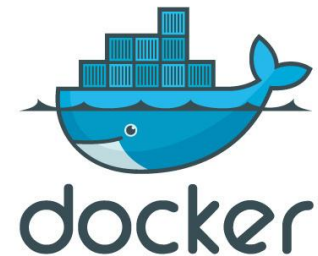
The main workflow canvas shows a process flow: two **File Reader** nodes (**Node 1** and **Node 4**) feed into a **K Nearest Neighbor** node (**Node 3**). The output of **Node 3** is split into two paths: one leading to a **Box Plot** node (**Node 5**) and another leading to a **CSV Writer** node.

In the foreground, the **Dialog - 0:1 - File Reader** window is open, showing the configuration for **Node 4**. The **File** tab is active, displaying the **Enter ASCII data file location** field with a **Browse...** button. Below this, there is a checkbox for **Preserve user settings for new location** and a **Rescan** button. The **Basic Settings** section includes checkboxes for **read row IDs** and **read column headers**, a **Column delimiter** dropdown menu (currently set to **<none>**), checkboxes for **ignore spaces and tabs** and **Java-style comments**, and a **Single line comment** text field. At the bottom of the dialog are **OK**, **Apply**, **Cancel**, and a help icon buttons.

To the right of the workflow canvas, there is a **Run** button and a play icon button.

# Introducing KNIME EnrichmentNodes

- KNIME Nodes are built on top of modular enrichment R scripts
- EnrichmentNodes use Docker containers that include
  - all software
  - all libraries and other dependencies
  - independent of local OS
- Docker image stored online at [dockerhub.io](https://dockerhub.io) and gets downloaded automatically

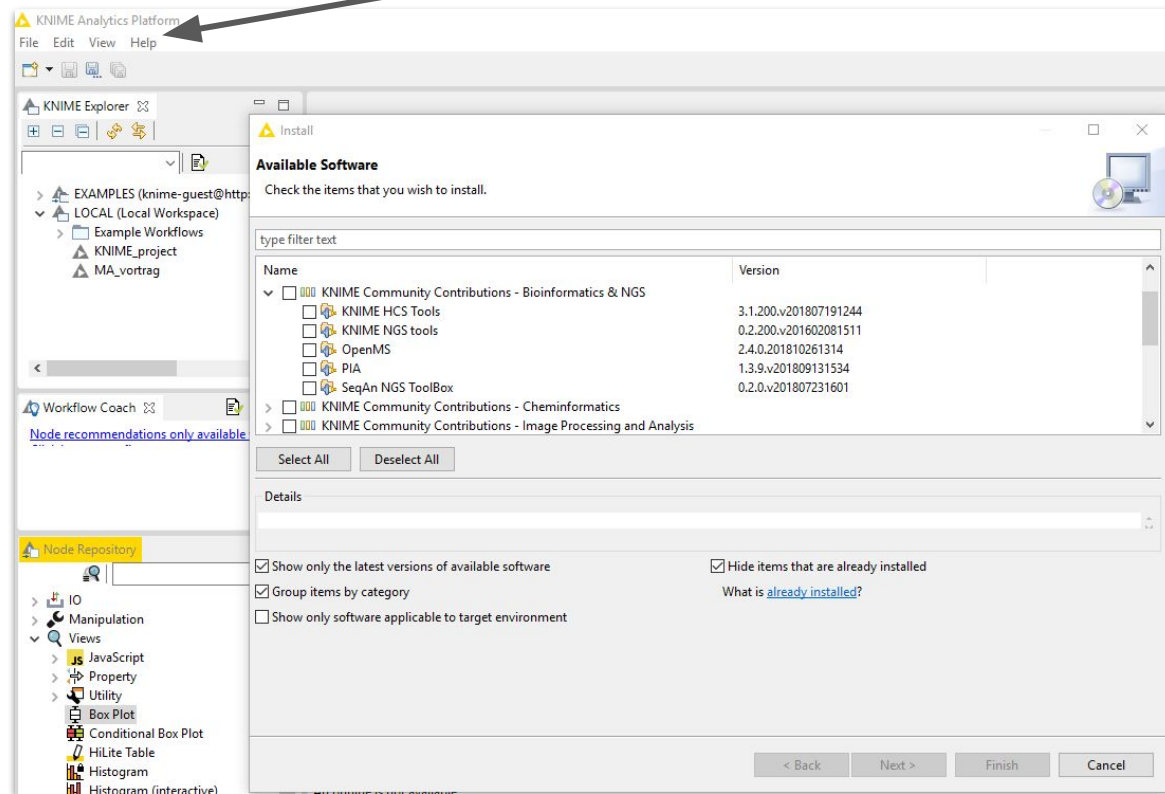




# KNIME Pathway Enrichment with “EnrichmentNodes”

Requirements: KNIME, Docker, CPU supporting VT-x or AMD-v

- Install extensions in KNIME: GKN, EnrichmentNodes (Help->Install software...)
- Docker loads automatically
- alternatively build your own version locally (temporary)



# Using KNIME

The screenshot displays the KNIME software interface. On the left is the **Node Repository** pane, which lists various node categories: IO, Manipulation, Views, Analytics, Database, Other Data Types, Structured Data, Scripting, Tools & Services, and Community Nodes. Under **Community Nodes**, the **EnrichmentNodes** folder is expanded, showing sub-nodes like Data, HLA Typing, curl, enrich, gsea, hiWorld, hiWorld2, and mona1.

The main workspace shows a workflow diagram with two paths. The left path starts with an **Input File** node, followed by a **Select input data** node, then the **mona1** node (Node 58), and finally an **Output File** node (Node 55). The right path starts with an **Input File** node, followed by a **Select input data** node, then the **gsea** node (Node 48), and finally an **Output File** node (Node 52). A large black arrow points from the **gsea** node in the workflow to the configuration dialog box in the foreground.

The foreground dialog box is titled **Dialog - 0:1 - gsea**. It has tabs for **File**, **Flow Variables**, **Job Manager Selection**, and **Memory**. The **Parameters** tab is active, showing a table of configuration parameters for the **gsea** node.

Parameter	Value	Type
<b>gsea</b>		
gmt	KEGG	string choice
<b>minSize</b>	15	integer [0:1000]
maxSize	500	integer [1:1000]
nperm	10000	integer [100:1000000000]

At the bottom of the dialog are buttons for **OK**, **Apply**, **Cancel**, and a help icon.

# KNIME for Developers

# Integrating own Methods into KNIME Nodes

**First Step:** integrate your R script/tool into our modular R framework

```
# -----  
# ' Script to run GSEA analysis  
# '  
# ' @author Ines Assum  
# ' @param input string // .RDS input file: named numerical vector (gene symbols)  
# ' @param output string // file path to save results as .RDS  
# ' @param gmt string // Gene set definition (KEGG / GO / custom [mygmt])  
# ' @param minSize integer // min size of GSs to be considered, default: 15  
# ' @param maxSize integer // max size of GSs to be considered, default: 500  
# ' @param nperm integer // number of permutations, default: 10000  
# ' @param mygmt string // custom .gmt file // not necessary  
# -----  
  
run_fgsea <- function(input, output, gmt, minSize, maxSize, nperm, mygmt){  
  library(fgsea)  
  pathways <- gmtPathways(gmt)  
  GSEA <- fgsea(pathways,  
                rank,  
                nperm,  
                minSize=minSize,  
                maxSize=maxSize)  
}
```

# Integrating own Methods into KNIME

**Requirements:** Your Tool/R-Script, [KNIME SDK](#), [Ant](#) and GenericKnimeNodes ([GKN](#))

## 1. CTD-file

```
<tool name="YourTool.R" >
  <description>Draw plot for a nucleotide sequence.</description>
  <cli>
    <clielement optionIdentifier="-sequence">
      <mapping referenceName="YourTool.sequence" />
    </clielement>
    <clielement optionIdentifier="-outfile">
      <mapping referenceName="YourTool.outputfile" />
    </clielement>
  </cli>
  <PARAMETERS >
    <NODE name="YourTool" description="Draw plot for a nucleotide sequence.">
      <ITEM name="sequence" value="" description="sequence filename" supported_formats="*.fasta"/>
      <ITEM name="outputfile" value="" description="Output file." supported_formats="*.ps"/>
    </NODE>
  </PARAMETERS>
</tool>
```

# Integrating own Methods into KNIME

## 2. Specifying plugin.properties file

```
1  # the package of the plugin
2  pluginPackage=de.enrichment
3
4  # the name of the plugin
5  pluginName=EnrichmentNodes
6
7  # the version of the plugin
8  pluginVersion=1.0.0.0
9
10 # the path (starting from KNIMEs Community Nodes node)
11 nodeRepositoryRoot=community
12
13 executor=com.genericworkflownodes.knime.execution.impl.LocalDockerToolExecutor
14 commandGenerator=com.genericworkflownodes.knime.execution.impl.DockerCommandGenerator
15
16 #docker specific configurations
17 dockerMachine=default
18
19 #tool specific configurations
20 tool.monai.dockerImage=enrich
21 tool.gsea.dockerImage=enrich
22 tool.enrich.dockerImage=enrich
```

# Integrating own Methods into KNIME Nodes

## 3. Provide predefined directory

```
Plugin_dir
|
|--- plugin.properties  (description file)
|
|--- descriptors
|    |-- yourTool.ctd
|    |-- mime.types
|
|--- DESCRIPTION        (short description of project)
|
|--- LICENSE            (Licensing information)
|
|--- COPYRIGHT          (Copyright information)
```

## 4. Run ant

## 5. Import and run on KNIME SDK

**Voila!**

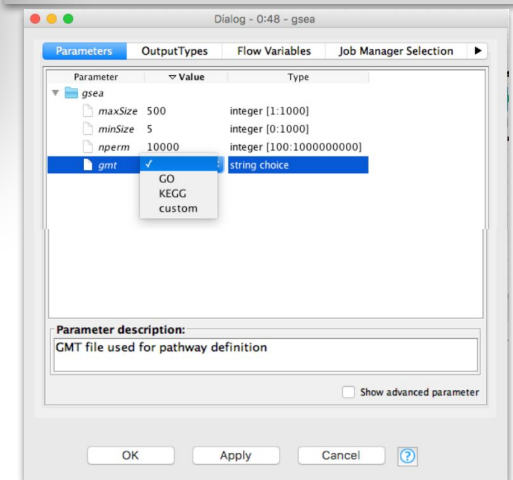
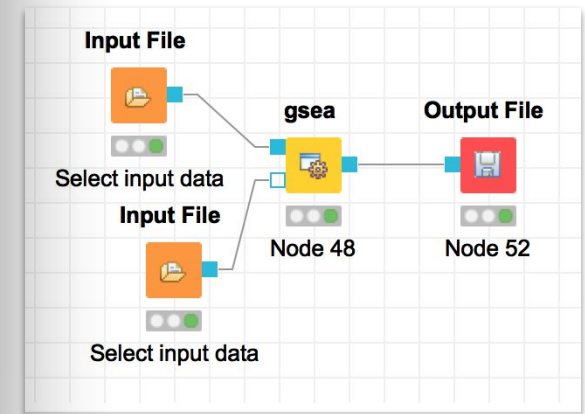


# Example

```
# -----
#' Script to run GSEA analysis
#'
#' @author Ines Assum
#' @param input string // .RDS input file: named numerical vector (gene symbols)
#' @param output string // file path to save results as .RDS
#' @param gmt string // Gene set definition (KEGG / GO / custom [mygmt])
#' @param minSize integer // min size of GSs to be considered, default: 15
#' @param maxSize integer // max size of GSs to be considered, default: 500
#' @param nperm integer // number of permutations, default: 10000
#' @param mygmt string // custom .gmt file // not necessary
# -----

run_fgsea <- function(input, output, gmt, minSize, maxSize, nperm, mygmt){
  library(fgsea)
  pathways <- gmtPathways(gmt)
  GSEA <- fgsea(pathways,
    rank,
    nperm,
    minSize=minSize,
    maxSize=maxSize)
}
```

```
7 <ITEM description="GMT file used for pathway definition" name="gmt" restrictions="GO,KEGG,c
8 <ITEM description="min Size" name="minSize" restrictions="0:1000" type="int" value="15"/>
9 <ITEM description="max Size" name="maxSize" restrictions="1:1000" type="int" value="500"/>
10 <ITEM description="Permutations" name="nperm" restrictions="100:1000000000" type="int" valu
11 </NODE>
12 </PARAMETERS>
13 <cli>
14 <cliElement optionIdentifier="-gmt">
15 <mapping referenceName="gsea.gmt"/>
16 </cliElement>
17 <cliElement optionIdentifier="-min">
18 <mapping referenceName="gsea.minSize"/>
19 </cliElement>
20 <cliElement optionIdentifier="-max">
21 <mapping referenceName="gsea.maxSize"/>
22 </cliElement>
23 <cliElement optionIdentifier="-p">
24 <mapping referenceName="gsea.nperm"/>
25 </cliElement>
26 </cli>
27 </tool>
```

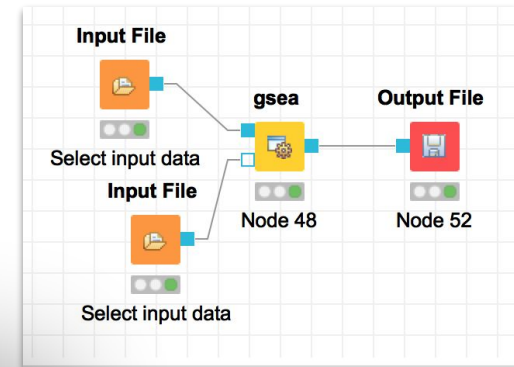




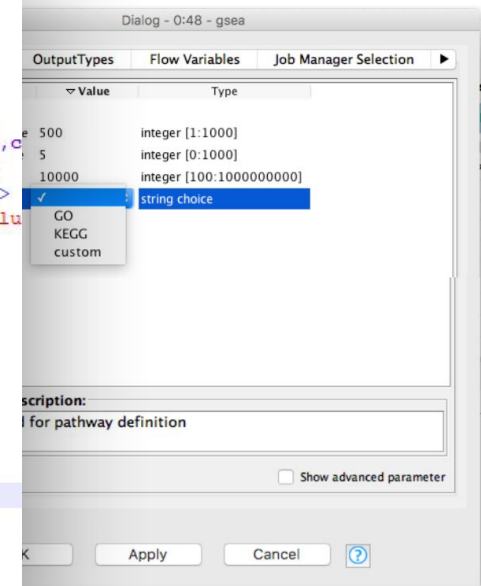
# Example

```
# -----
# Script to run GSEA analysis
#
# @author Ines Assum
# @param input string // .RDS input file: named numerical vector (gene symbols)
# @param output string // file path to save results as .RDS
# @param gmt string // Gene set definition (KEGG / GO / custom [mygmt])
# @param minSize integer // min size of GSs to be considered, default: 15
# @param maxSize integer // max size of GSs to be considered, default: 500
# @param nperm integer // number of permutations, default: 10000
# @param mygmt string // custom .gmt file // not necessary
# -----
```

```
run_fgsea <- function(input, output, gmt, minSize, maxSize, nperm, mygmt){
  library(fgsea)
  pathways <- gmtPathways(gmt)
  GSEA <- fgsea(pathways,
    rank,
    nperm,
    mygmt)
```



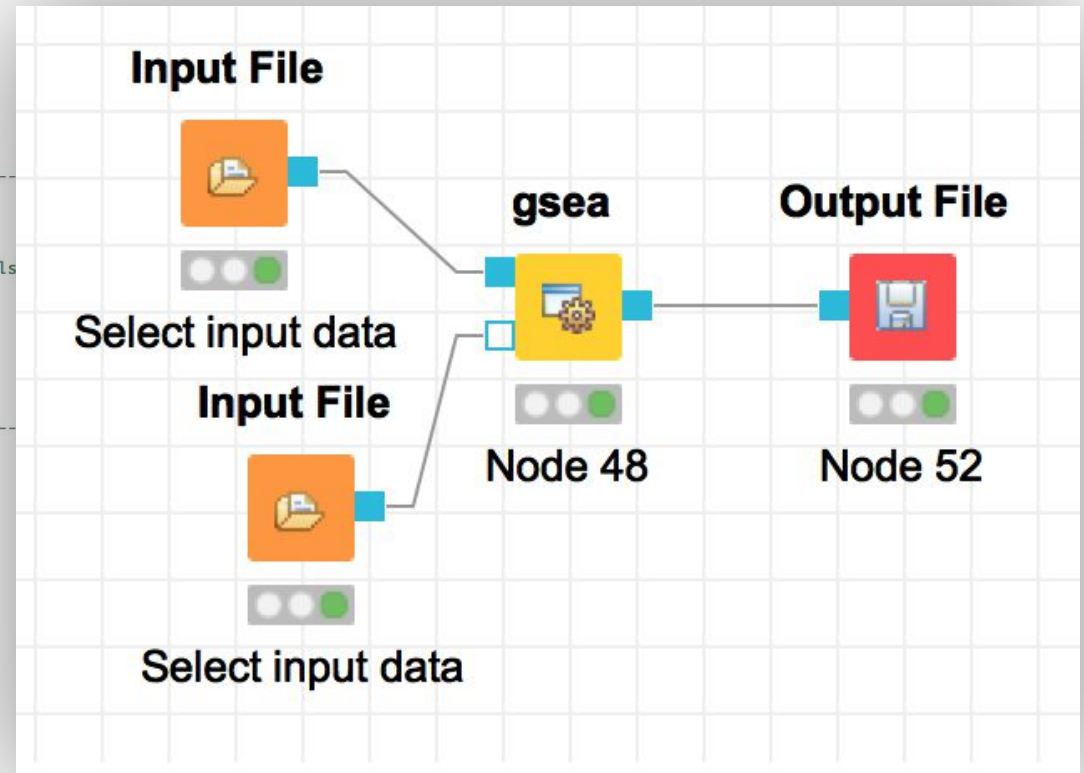
```
<tool name="gsea">
  <description>Runs GSEA using fgsea R package.</description>
  <manual>Detailed description goes here.</manual>
  <executableName>run_fgsea.R</executableName>
  <PARAMETERS >
    <NODE name="gsea" description="Node the runs GSEA">
      <ITEM description="GMT file used for pathway definition" name="gmt" restrictions="GO,KEGG,c
      <ITEM description="min Size" name="minSize" restrictions="0:1000" type="int" value="15"/>
      <ITEM description="max Size" name="maxSize" restrictions="1:1000" type="int" value="500"/>
      <ITEM description="Permutations" name="nperm" restrictions="100:1000000000" type="int" valu
    </NODE>
  </PARAMETERS>
  <cli>
    <clielement optionIdentifier="-gmt">
      <mapping referenceName="gsea.gmt"/>
    </clielement>
    <clielement optionIdentifier="-min">
      <mapping referenceName="gsea.minSize"/>
    </clielement>
    <clielement optionIdentifier="-max">
      <mapping referenceName="gsea.maxSize"/>
    </clielement>
    <clielement optionIdentifier="-p">
      <mapping referenceName="gsea.nperm"/>
    </clielement>
  </cli>
</tool>
```



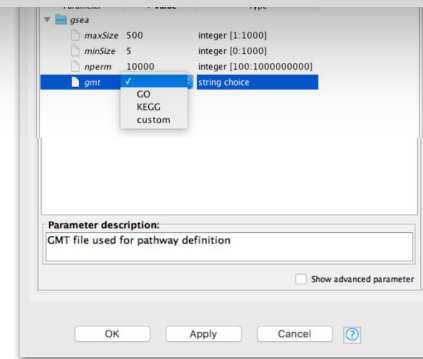
# Example

```
# -----
# Script to run GSEA analysis
#
# @author Ines Assum
# @param input string // .RDS input file: named numerical vector (gene symbols)
# @param output string // file path to save results as .RDS
# @param gmt string // Gene set definition (KEGG / GO / custom [mygmt])
# @param minSize integer // min size of GSs to be considered, default: 15
# @param maxSize integer // max size of GSs to be considered, default: 500
# @param nperm integer // number of permutations, default: 10000
# @param mygmt string // custom .gmt file // not necessary
# -----

run_fgsea <- function(input, output, gmt, minSize, maxSize, nperm, mygmt){
  library(fgsea)
  pathways <- gmtPathways(gmt)
  GSEA <- fgsea(pathways,
    rank,
    nperm,
    minSize=minSize,
    maxSize=maxSize)
}
```



```
1 <tool name="gsea">
2   <description>Runs GSEA using fgsea R package.</description>
3   <manual>Detailed description goes here.</manual>
4   <executableName>run_fgsea.R</executableName>
5   <PARAMETERS>
6     <NODE name="gsea" description="Node the runs GSEA">
7       <ITEM description="GMT file used for pathway definition" name="gmt" restrictions="GO,KEGG,custom" type="string choice">
8         <ITEM description="min Size" name="minSize" restrictions="0:1000" type="int" value="15"/>
9         <ITEM description="max Size" name="maxSize" restrictions="1:1000" type="int" value="500"/>
10        <ITEM description="Permutations" name="nperm" restrictions="100:1000000000" type="int" value="10000"/>
11      </NODE>
12    </PARAMETERS>
13    <cli>
14      <cliElement optionIdentifier="-gmt">
15        <mapping referenceName="gsea.gmt"/>
16      </cliElement>
17      <cliElement optionIdentifier="-min">
18        <mapping referenceName="gsea.minSize"/>
19      </cliElement>
20      <cliElement optionIdentifier="-max">
21        <mapping referenceName="gsea.maxSize"/>
22      </cliElement>
23      <cliElement optionIdentifier="-p">
24        <mapping referenceName="gsea.nperm"/>
25      </cliElement>
26    </cli>
27  </tool>
```

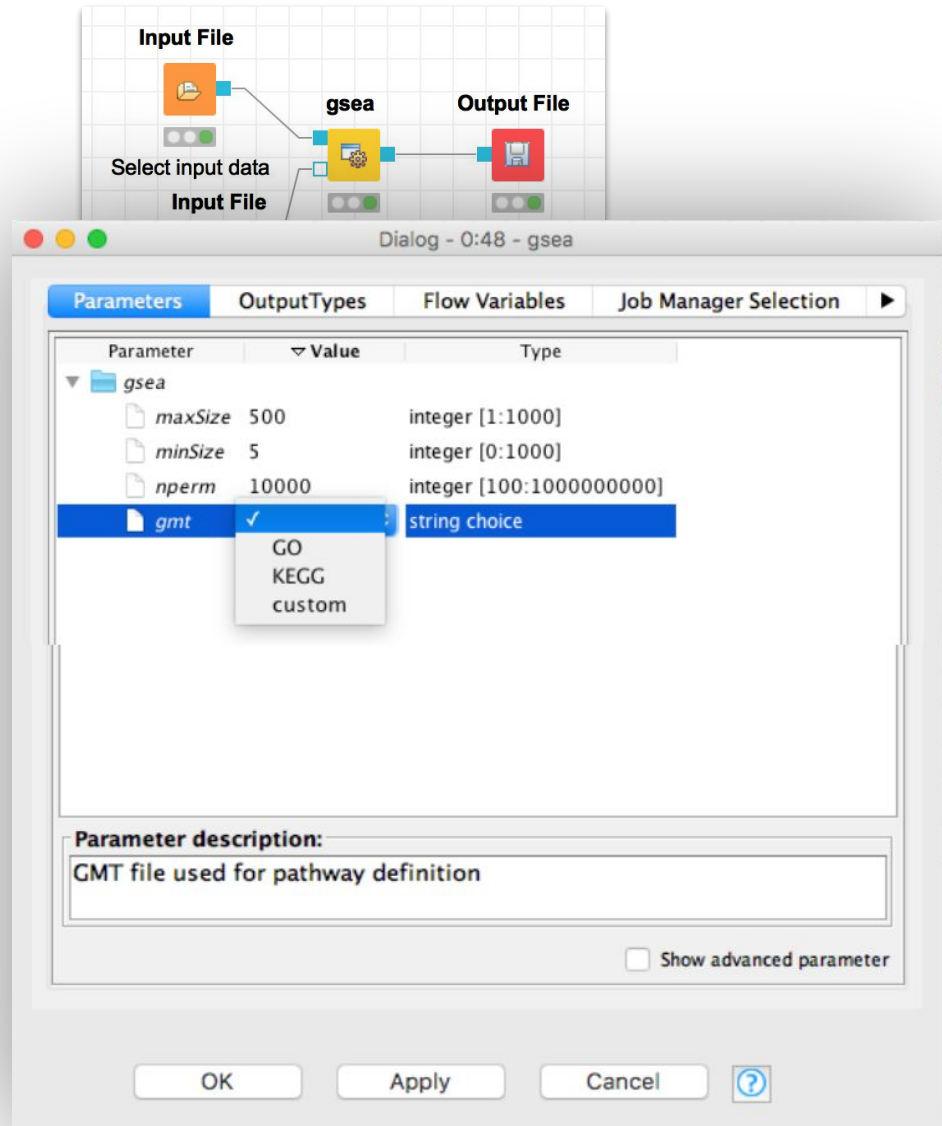


# Example

```
# -----
# ' Script to run GSEA analysis
# '
# ' @author Ines Assum
# ' @param input string // .RDS input file: named numerical vector (gene symbols)
# ' @param output string // file path to save results as .RDS
# ' @param gmt string // Gene set definition (KEGG / GO / custom [mygmt])
# ' @param minSize integer // min size of GSs to be considered, default: 15
# ' @param maxSize integer // max size of GSs to be considered, default: 500
# ' @param nperm integer // number of permutations, default: 10000
# ' @param mygmt string // custom .gmt file // not necessary
# -----
```

```
run_fgsea <- function(input, output, gmt, minSize, maxSize, nperm, mygmt){
  library(fgsea)
  pathways <- gmtPathways(gmt)
  GSEA <- fgsea(pathways,
    rank,
    nperm,
    minSize=minSize,
    maxSize=maxSize)
}
```

```
1 <tool name="fgsea">
2   <description>Runs GSEA using fgsea R package.</description>
3   <manual>Detailed description goes here.</manual>
4   <executableName>run_fgsea.R</executableName>
5   <PARAMETERS >
6     <NODE name="fgsea" description="Node the runs GSEA">
7       <ITEM description="GMT file used for pathway definition" name="gmt" restrictions="GO,KEGG,c
8       <ITEM description="min Size" name="minSize" restrictions="0:1000" type="int" value="15"/>
9       <ITEM description="max Size" name="maxSize" restrictions="1:1000" type="int" value="500"/>
10      <ITEM description="Permutations" name="nperm" restrictions="100:1000000000" type="int" valu
11    </NODE>
12  </PARAMETERS>
13  <cli>
14    <cliElement optionIdentifier="-gmt">
15      <mapping referenceName="fgsea.gmt"/>
16    </cliElement>
17    <cliElement optionIdentifier="-min">
18      <mapping referenceName="fgsea.minSize"/>
19    </cliElement>
20    <cliElement optionIdentifier="-max">
21      <mapping referenceName="fgsea.maxSize"/>
22    </cliElement>
23    <cliElement optionIdentifier="-p">
24      <mapping referenceName="fgsea.nperm"/>
25    </cliElement>
26  </cli>
27 </tool>
```



# Summary

- Modular R library integrated into easy-to-use KNIME workflow
- Expand functionality with minimal effort
- We will soon make the “EnrichmentNodes” project on GitHub public:  
-> <https://github.com/InesAssum/EnrichmentNodes>
- Template for exemplary node provided
- Collect and share methods
- Join our slack workspace for discussion and support  
-> [Slack.com/KNIME-setup](https://slack.com/KNIME-setup)
- Check out Benni’s ImmunoNodes: [github.com/FRED-2/ImmunoNodes/](https://github.com/FRED-2/ImmunoNodes/)

# Acknowledgements

## Epigenereg group

Matthias Heinig

Ines Assum

Thomas Walzthöni

## ICB (MONA)

Nikola Müller

Florian Büttner

Andreas Kopf

## GenericKnimeNodes

Julianus Pfeuffer

## & ICB

Benjamin Schubert

-> [ImmunoNodes](#)

