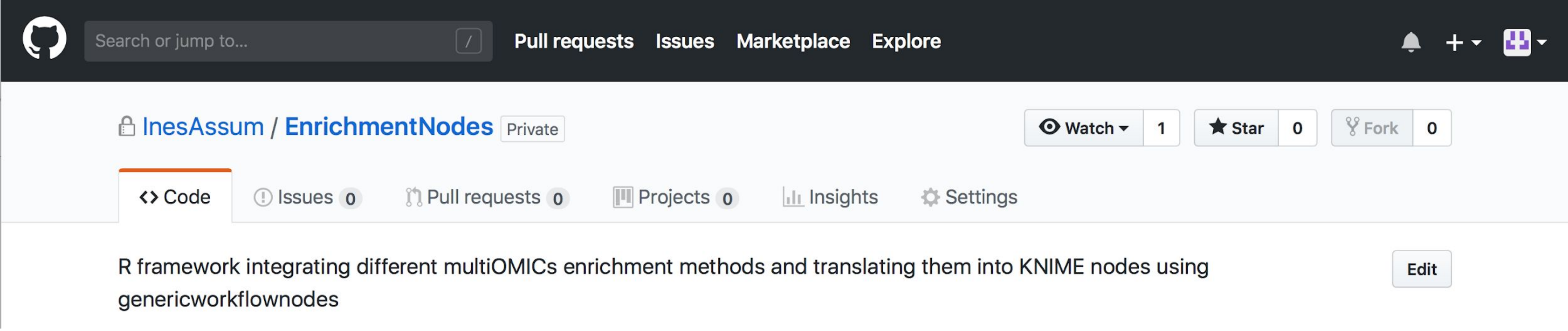
Pathway Enrichment with KNIME

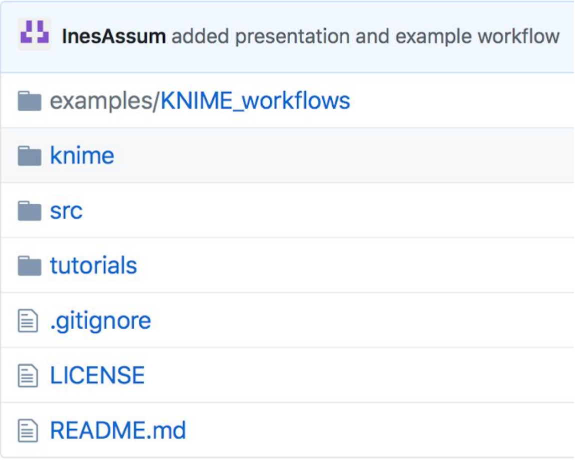


R framework integrating different multiOMICs enrichment methods and translating them into KNIME nodes using genericworkflownodes

1. **Pathway Enrichment using a modular R library**

2. **Pathway Enrichment with KNIME for Users**

3. **Pathway Enrichment with KNIME for Developers**

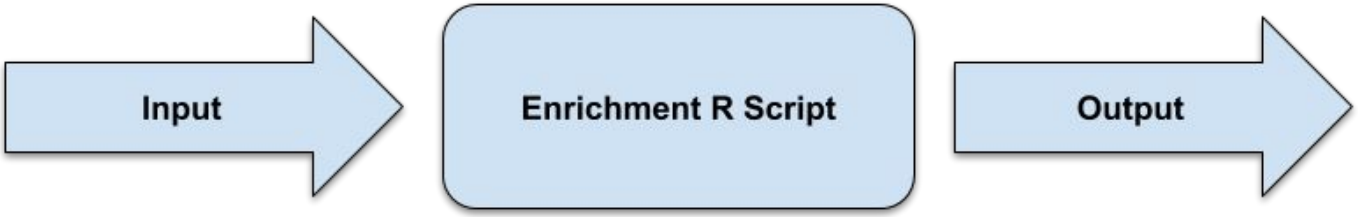


InesAssum added presentation and example workflow

- examples/KNIME_workflows
- knime
- src
- tutorials
- .gitignore
- LICENSE
- README.md

Modular R library integrating multiple Pathway Enrichment methods

Modular R Framework



Expression data file (RDS, RData, txt, csv)

Optional: GMT-file (KEGG, GO etc.)

Parameters:

- **Method:** R function
- **General**: e.g. gene set size
- **method specific**, e.g. cutoff, number of permutations

RDS file
list(

- **Table:** pathways, score, pvals, probability
- **Log file:** method, parameters
- **Method specific output object:** (e.g. incl. leadingEdge in GSEA)

)

Current methods: MONA, GSEA

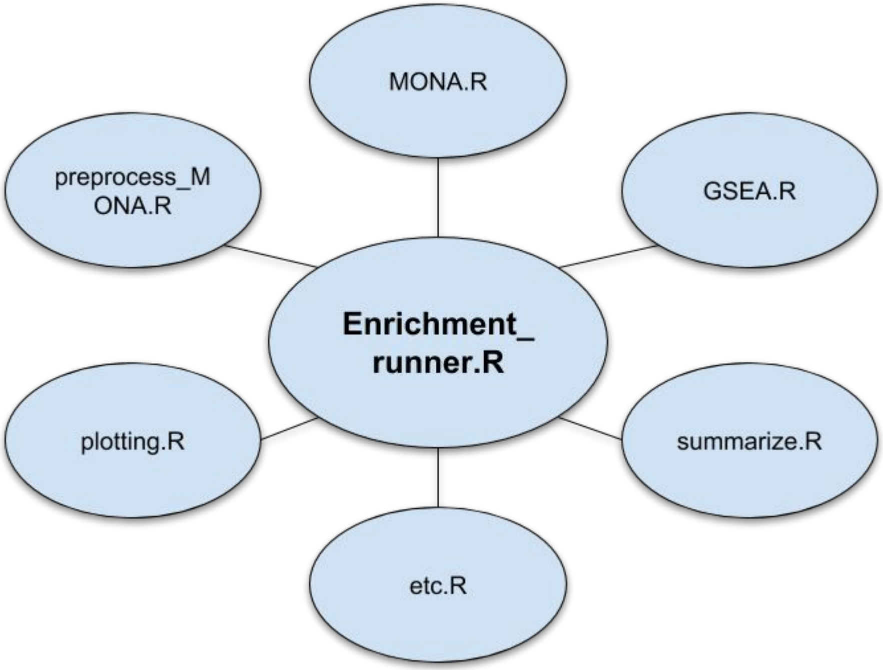
How to deal with different input data requirements,

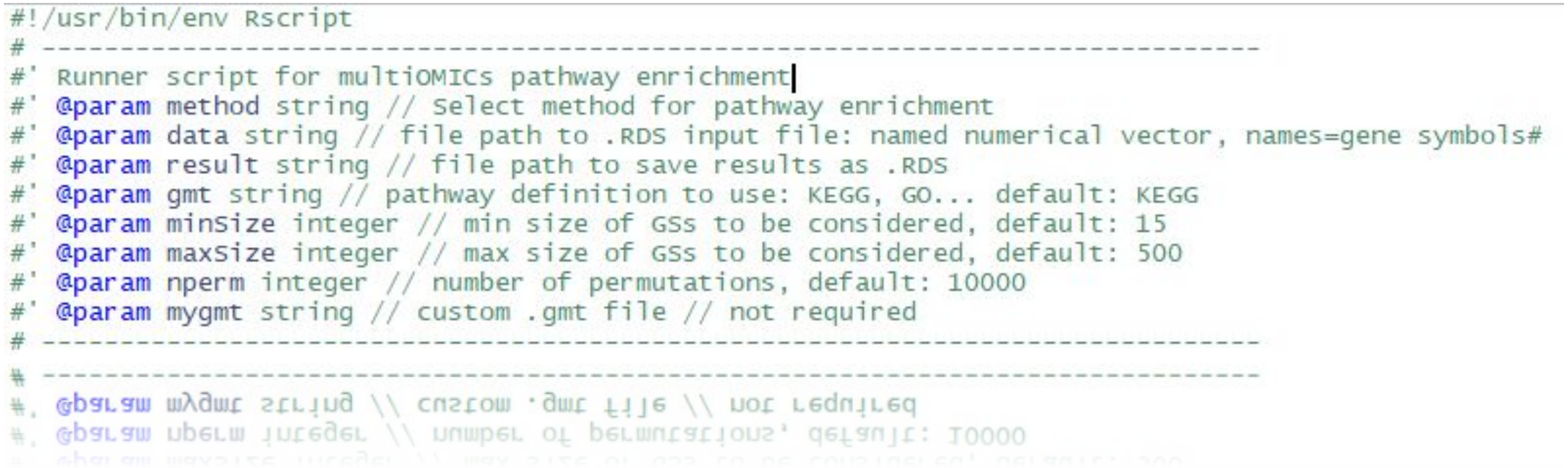
i.e. raw data vs. summary statistics?

GSEA: Subramanian et al., PNAS, 2005, MONA: Sass et al., NAR, 2013

Modular R Framework

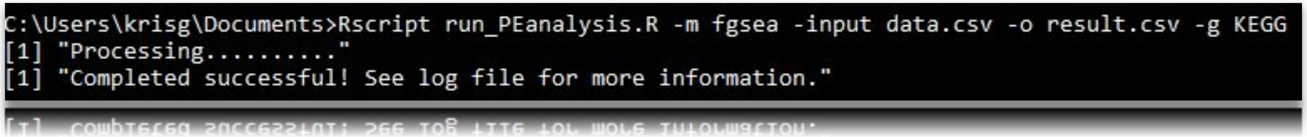
- Central runner script
- calls methods as R functions
- Idea: methods share common starting point (raw data)
- necessary preprocessing handled internally





```
#!/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 vector, names=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
# ------------------------------------------------------------------------------
```

Modular R Framework



```
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."
```

- Run from command line
- run locally in R or
- use Docker support
- Full code and documentation available on GitHub

Suited for: - big/advanced projects (simulations, benchmarking)

- anyone, who is using R anyway

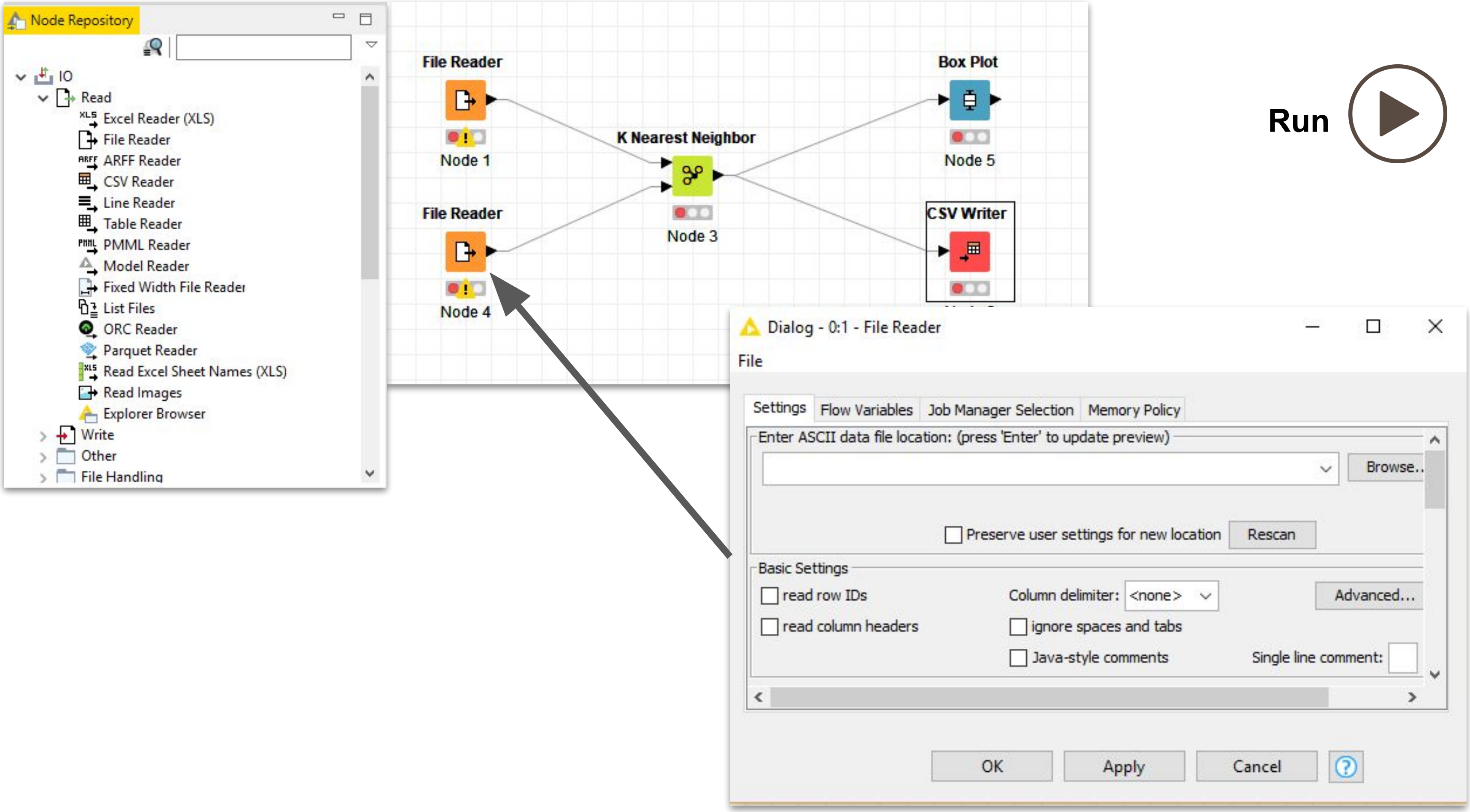
...but what if you don’t have a programming background ?

KNIME Enrichment for Users

KNIME - The Konstanz Information Miner, M. R. Berthold, B. Wiswedel, et al.

KNIME Analytics Platform

KNIME is a free software for interactive data analysis



Dialog - 0:1 - File Reader

File

Settings | Flow Variables | Job Manager Selection | Memory Policy

Enter ASCII data file location: (press 'Enter' to update preview)

Browse...

□ Preserve user settings for new location   Rescan

Basic Settings

□ read row IDs

□ read column headers

Column delimiter: <none>

□ ignore spaces and tabs

□ Java-style comments

Advanced...

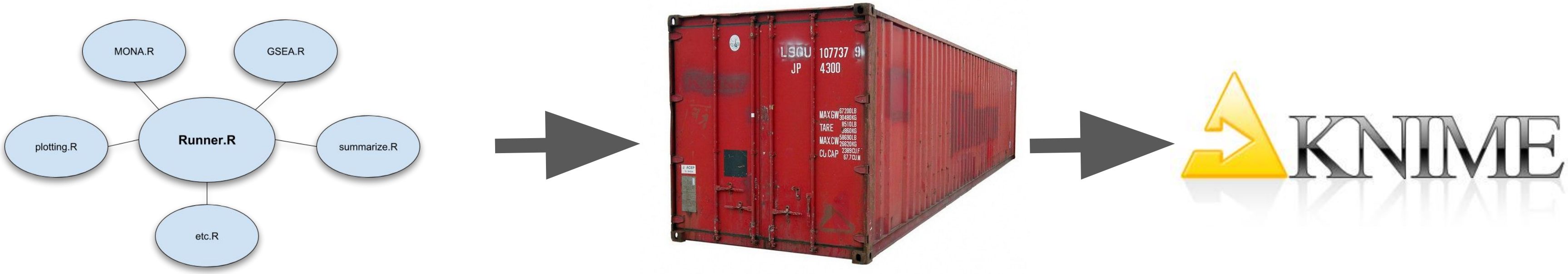
Single line comment:

OK   Apply   Cancel   ?

Run

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 and gets downloaded automatically



KNIME Pathway Enrichment with EnrichmentNodes

Requirements:

Platform independent using Docker Images

CPU must support VT-x or AMD-v

- KNIME >= 3.1: http://www.knime.org
- Docker >= 1.9: https://www.docker.com/
- Generic KNIME Node (with Docker support): (https://github.com/genericworkflownodes/GenericKnimeNodes)

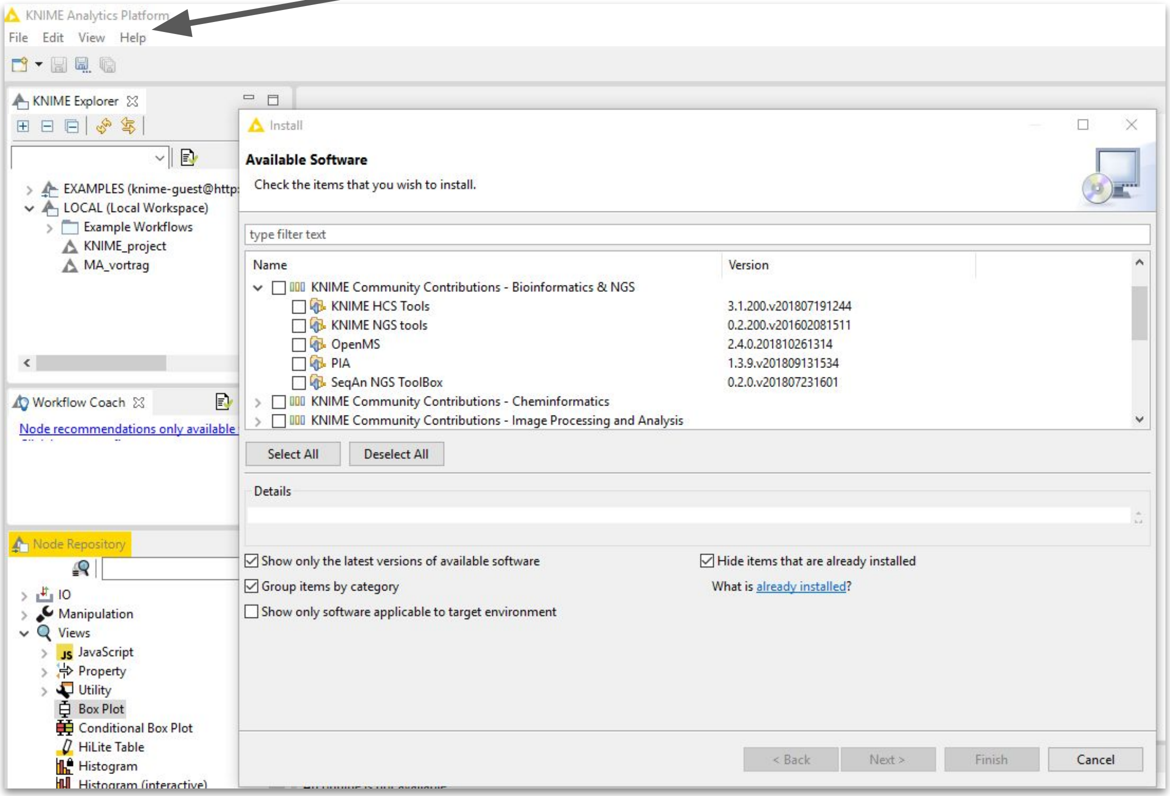
KNIME Pathway Enrichment with “EnrichmentNodes”

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)



Available Software

Check the items that you wish to install.

type filter text

Name	Version
□ KNIME Community Contributions - Bioinformatics & NGS	
□ KNIME HCS Tools	3.1.200.v201807191244
□ KNIME NGS tools	0.2.200.v201602081511
□ OpenMS	2.4.0.201810261314
□ PIA	1.3.9.v201809131534
□ SeqAn NGS ToolBox	0.2.0.v201807231601
□ KNIME Community Contributions - Cheminformatics	
□ KNIME Community Contributions - Image Processing and Analysis	

Details

☑ Show only the latest versions of available software

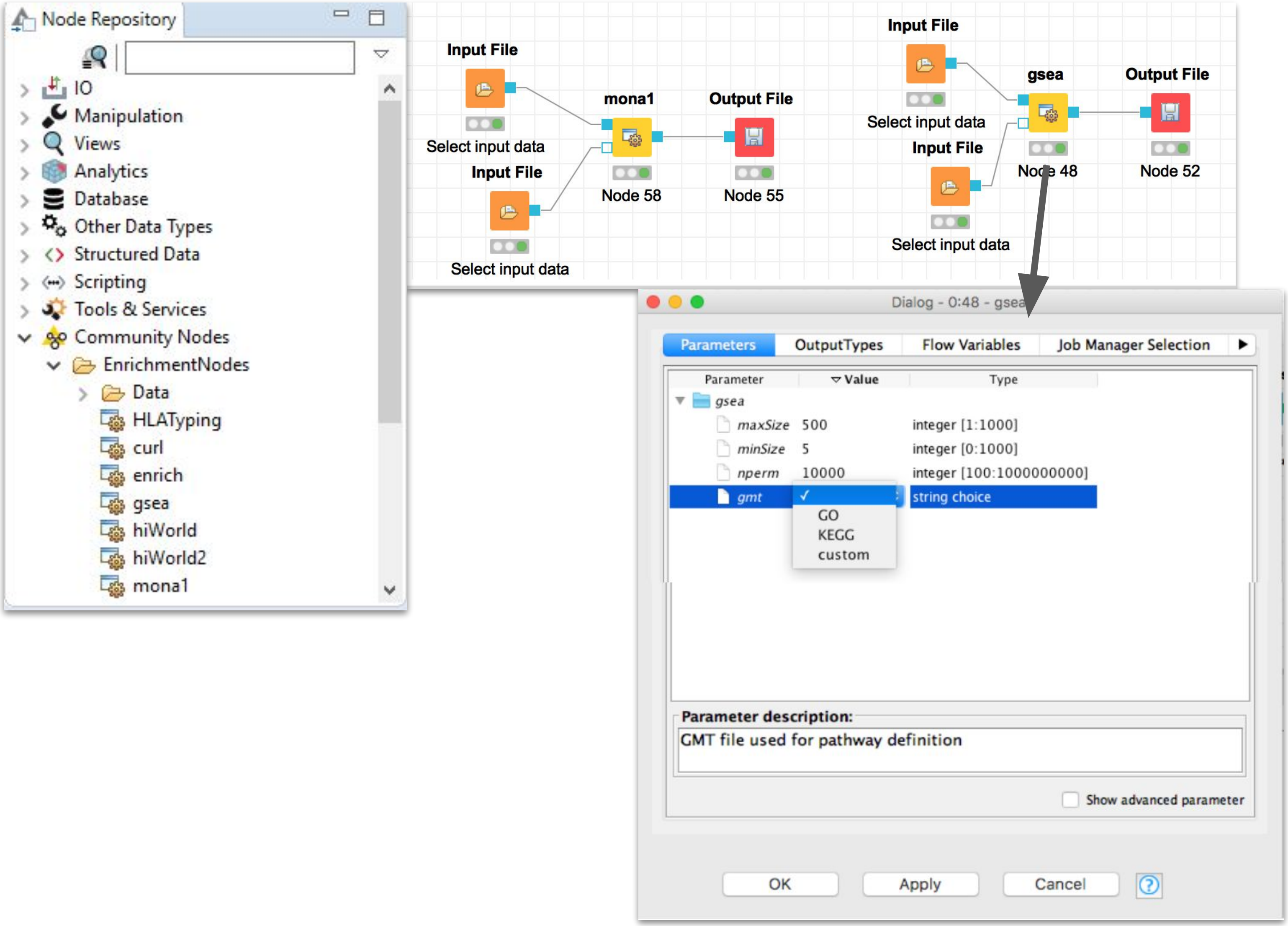
☑ Group items by category

□ Show only software applicable to target environment

☑ Hide items that are already installed

What is already installed?

Using KNIME



Parameter	Value	Type
gsea		
maxSize	500	integer [1:1000]
minSize	5	integer [0:1000]
nperm	10000	integer [100:1000000000]
gmt	✓ GO KEGG custom	string choice

Parameter description:

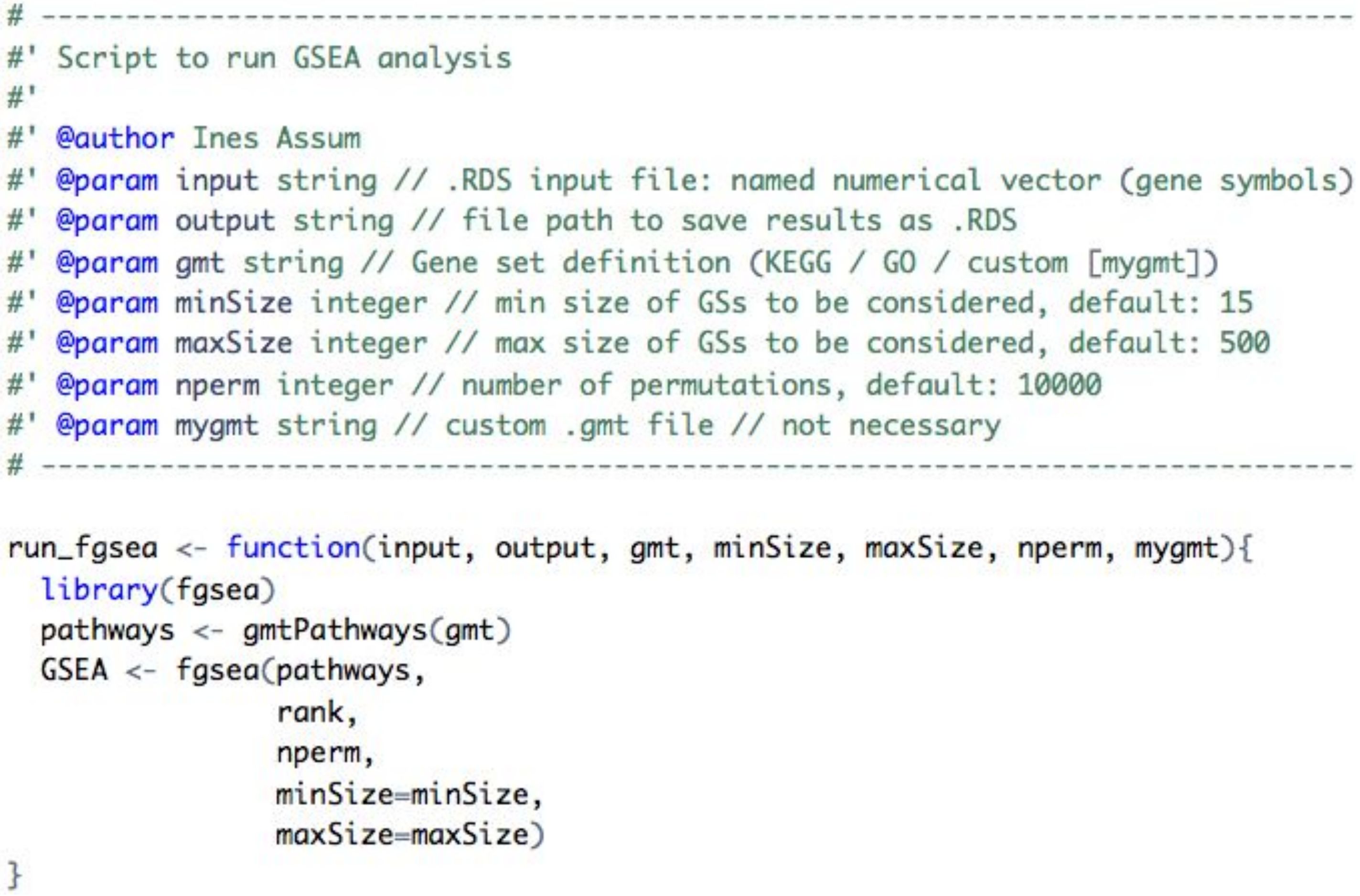
GMT file used for pathway definition

□ Show advanced parameter

KNIME Enrichment for Developers

Integrating own Methods into KNIME Nodes

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

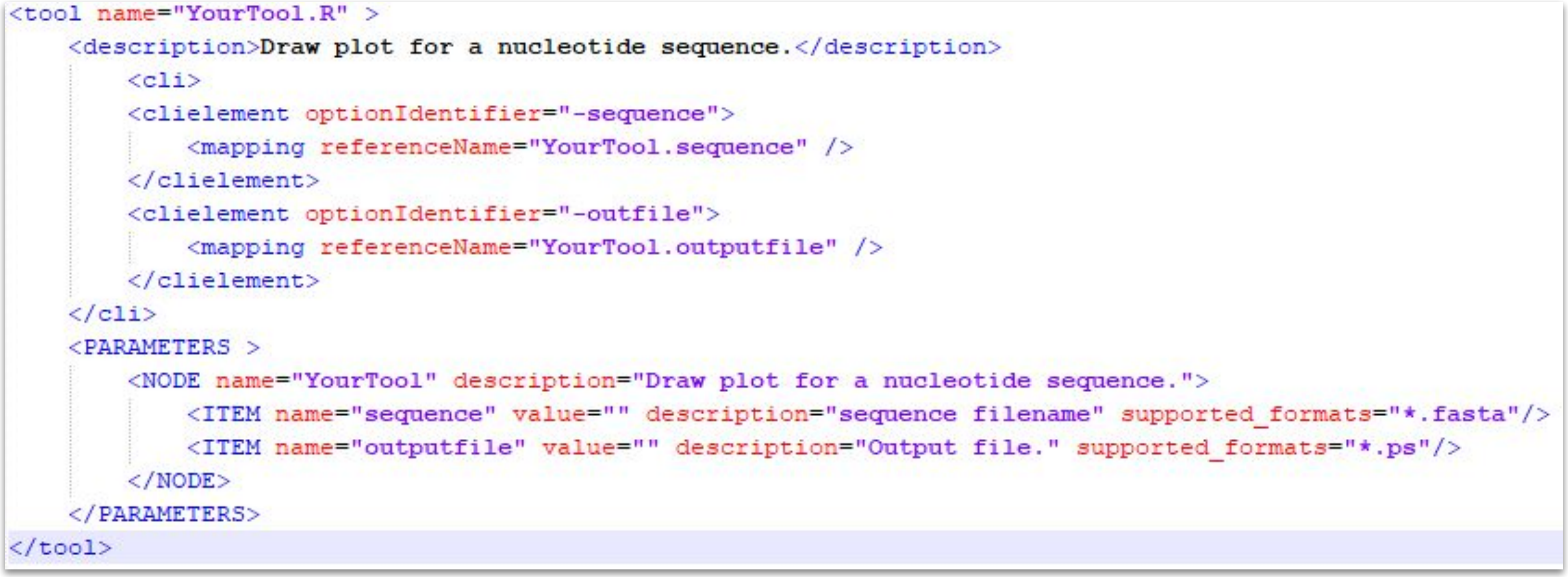


```
# ------------------------------------------------------------------------------
#' Script to run GSEA analysis
#'
#' @author Ines Assum
#' @param input string // .RDS input file: named numerical vector (gene symbols)
#' @param output string // file path to save results as .RDS
#' @param gmt string // 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)
}
```

fgsea: Sergushichev, bioRxiv, 2016

Integrating own Methods into KNIME

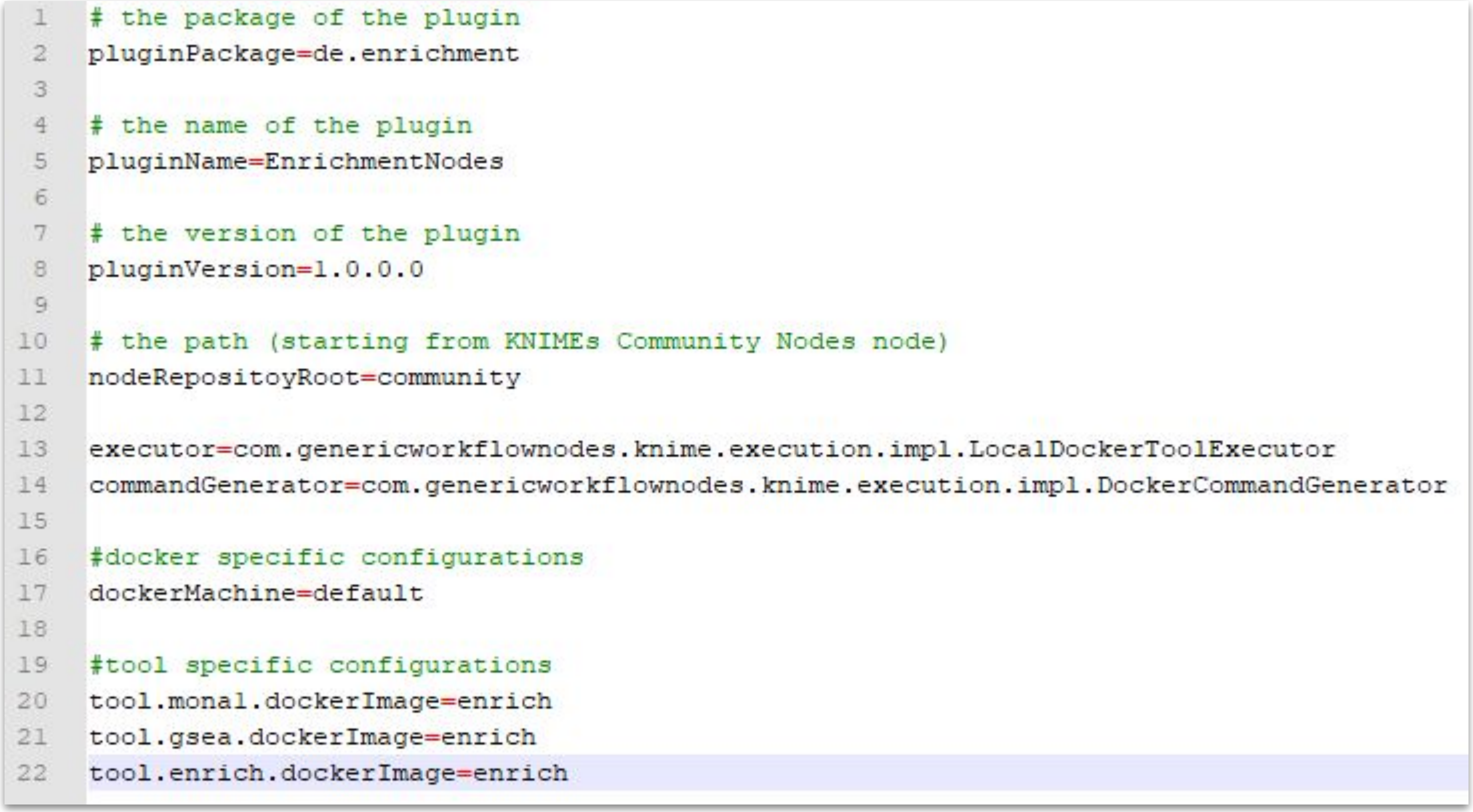
Step 2: Provide a CTD-file



```
<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

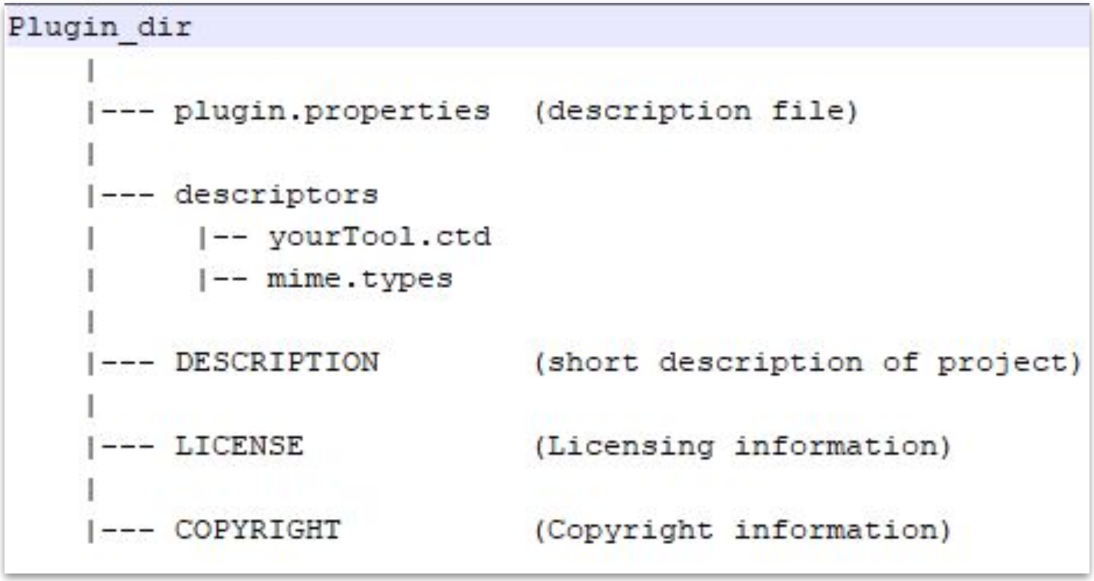
Step 3: Specify plugin.properties file



```
# the package of the plugin
pluginPackage=de.enrichment

# the name of the plugin
pluginName=EnrichmentNodes

# the version of the plugin
pluginVersion=1.0.0.0

# the path (starting from KNIMEs Community Nodes node)
nodeRepositoyRoot=community

executor=com.genericworkflownodes.knime.execution.impl.LocalDockerToolExecutor
commandGenerator=com.genericworkflownodes.knime.execution.impl.DockerCommandGenerator

#docker specific configurations
dockerMachine=default

#tool specific configurations
tool.monal.dockerImage=enrich
tool.gsea.dockerImage=enrich
tool.enrich.dockerImage=enrich
```

Integrating own Methods into KNIME Nodes

4. 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)
```

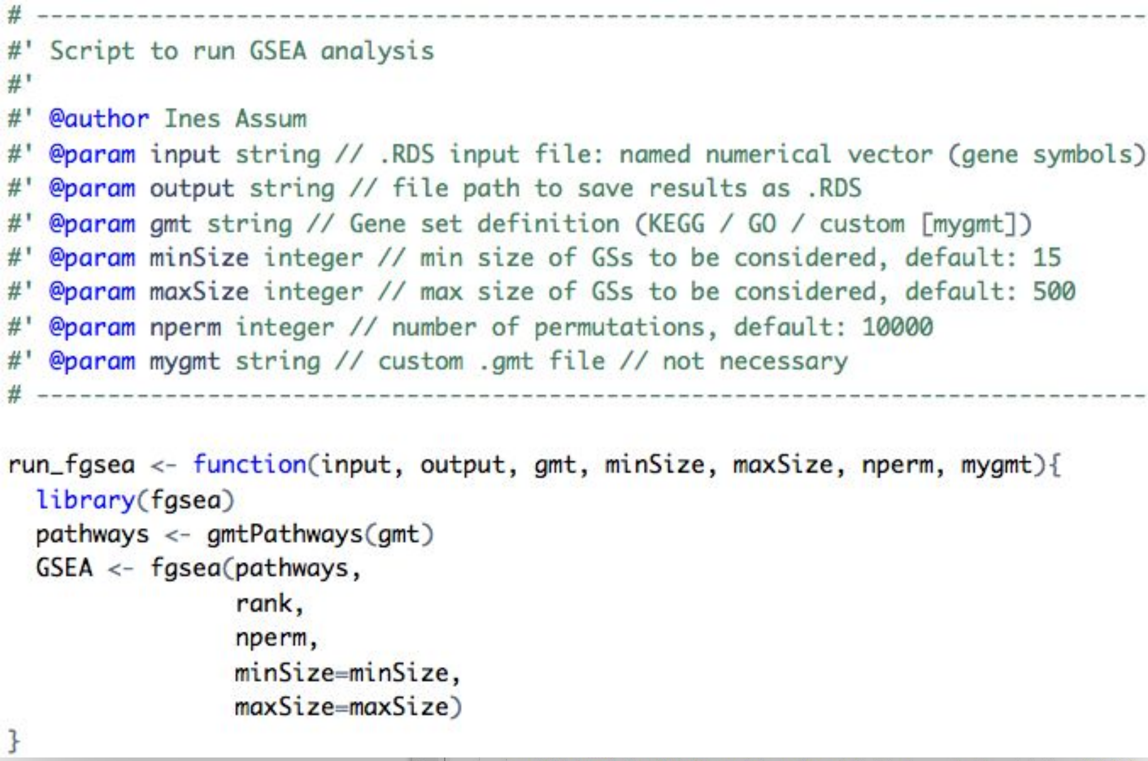
5. Get GenericKnimeNodes[2] and run ant

6. Import and run on KNIME SDK

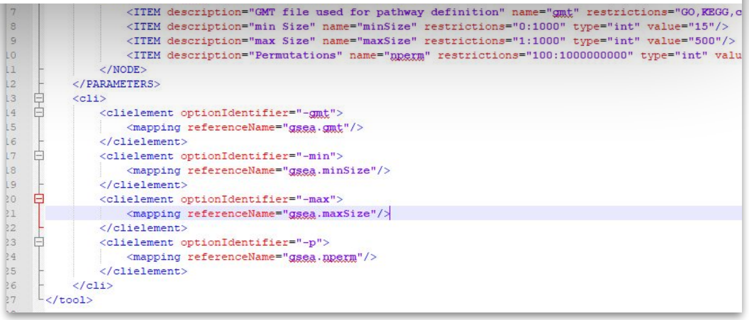
Voila!

[2]: GenericKNIMENodes (GKN)

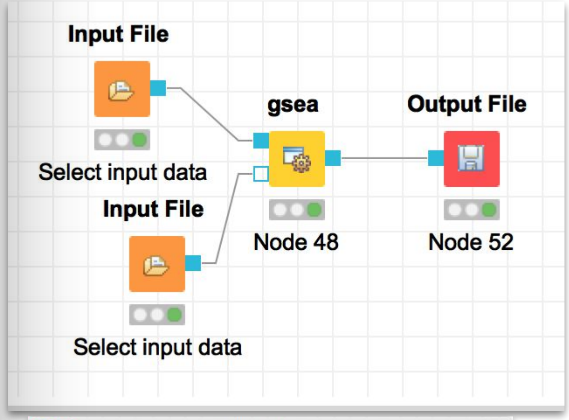
Example

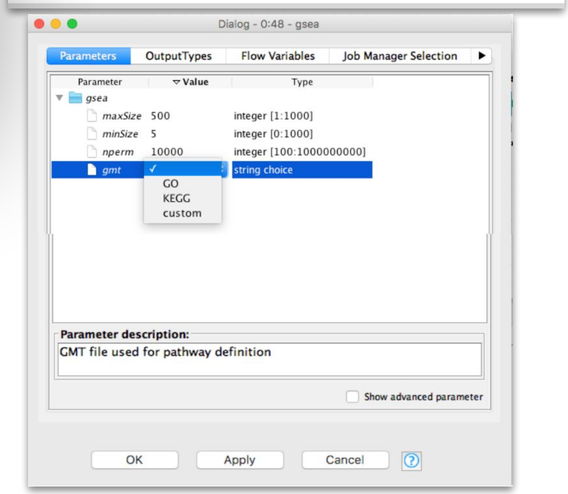


```
# ---------------------------------------------------------------------------
#' Script to run GSEA analysis
#'
#' @author Ines Assum
#' @param input string // .RDS input file: named numerical vector (gene symbols)
#' @param output string // file path to save results as .RDS
#' @param gmt string // 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)
}
```



```
            <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>
```





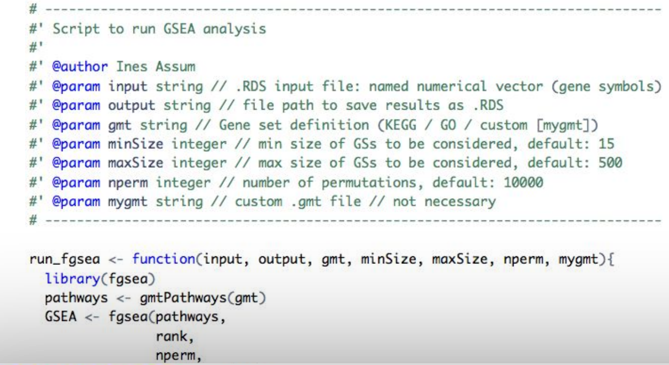
Parameter	Value	Type
gsea		
maxSize	500	integer [1:1000]
minSize	5	integer [0:1000]
nperm	10000	integer [100:1000000000]
gmt	✓	string choice

Parameter description:

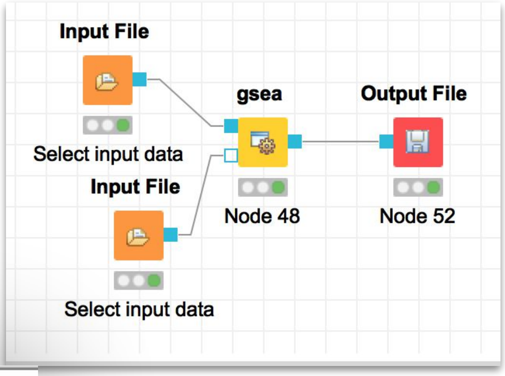
GMT file used for pathway definition

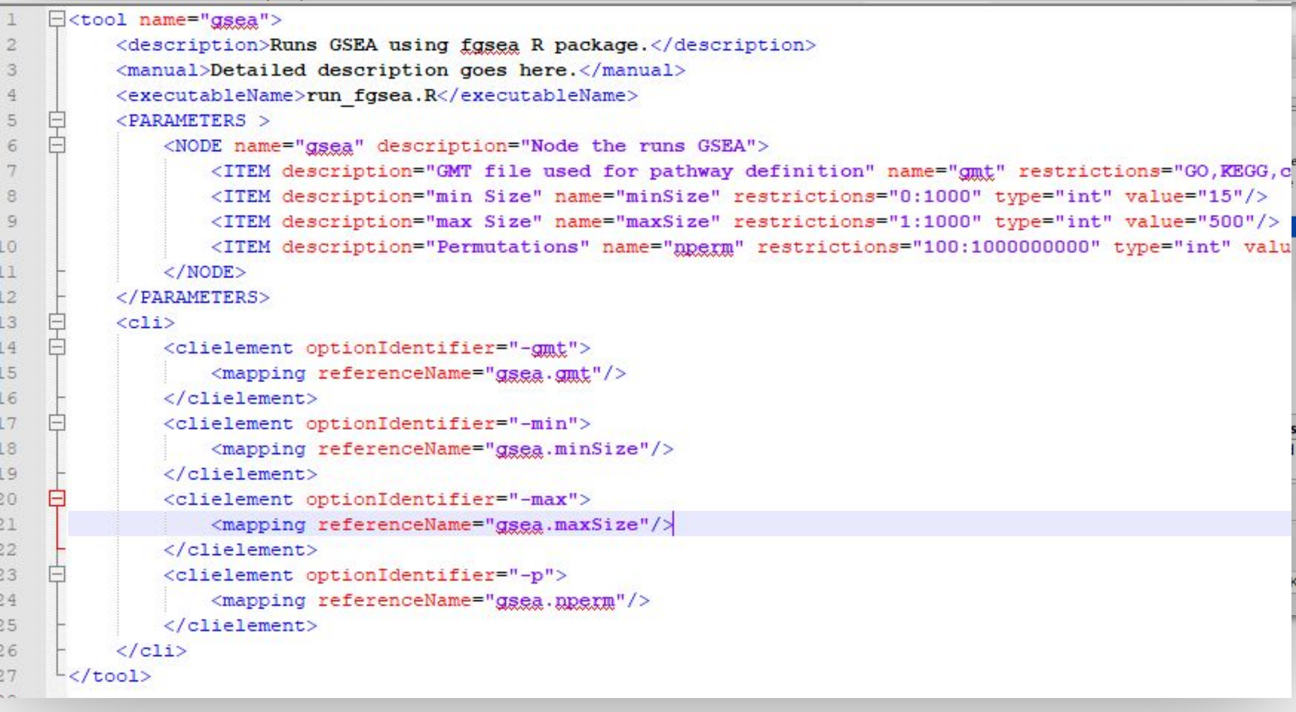
□ Show advanced parameter

Example

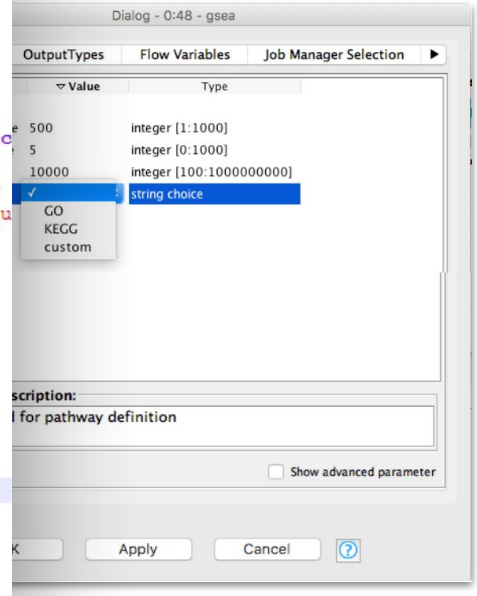


```r
# --------------------------------------------------------------------------
#' Script to run GSEA analysis
#'
#' @author Ines Assum
#' @param input string // .RDS input file: named numerical vector (gene symbols)
#' @param output string // file path to save results as .RDS
#' @param gmt string // 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,
```





```xml
<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>
```



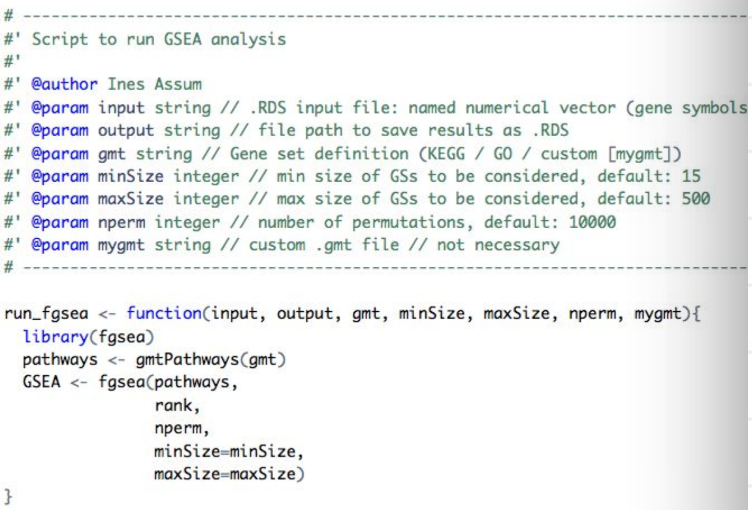
Value	Type
500	integer [1:1000]
5	integer [0:1000]
10000	integer [100:1000000000]
✓	string choice

GO
KEGG
custom

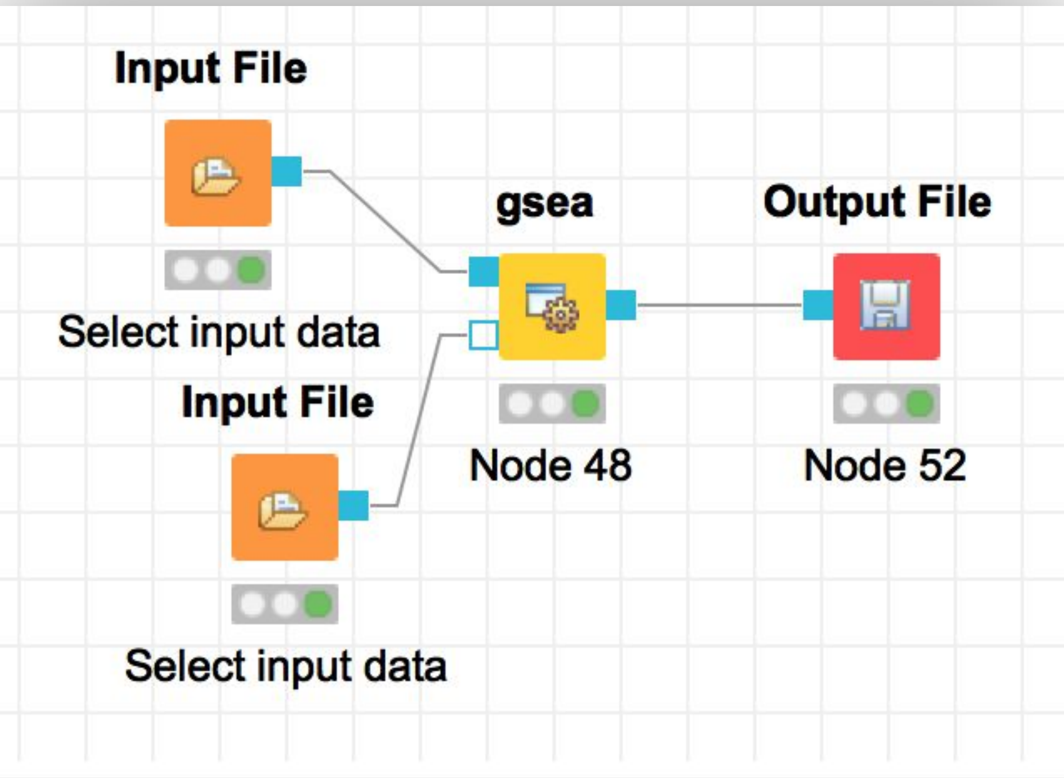
scription:
for pathway definition

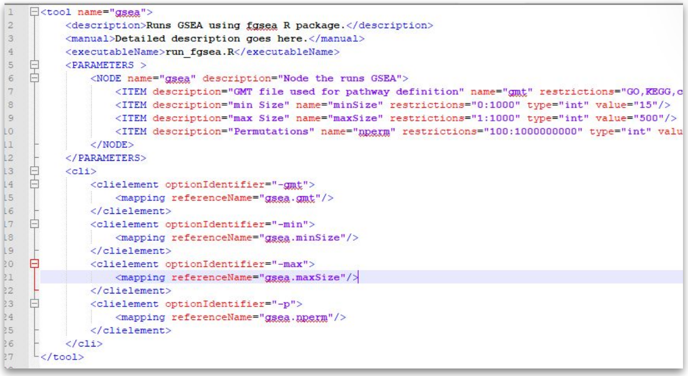
□ Show advanced parameter

Example

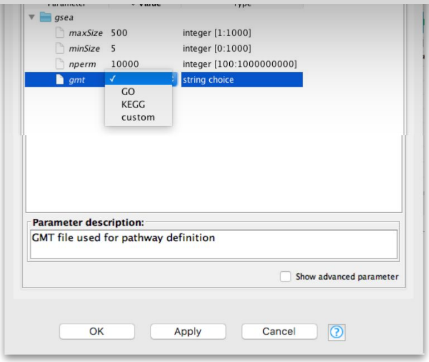


```
# ----------------------------------------------------------------------------
#' Script to run GSEA analysis
#'
#' @author Ines Assum
#' @param input string // .RDS input file: named numerical vector (gene symbols
#' @param output string // file path to save results as .RDS
#' @param gmt string // 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)
}
```





```
<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>
```

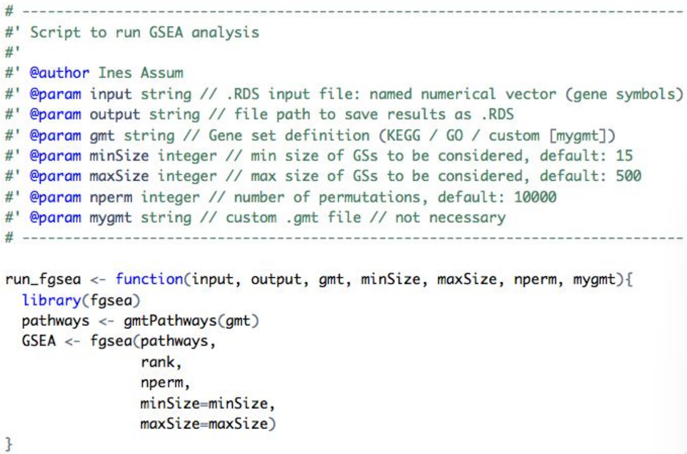


gsea		
maxSize	500	integer [1:1000]
minSize	5	integer [0:1000]
nperm	10000	integer [100:1000000000]
gmt	✓	string choice

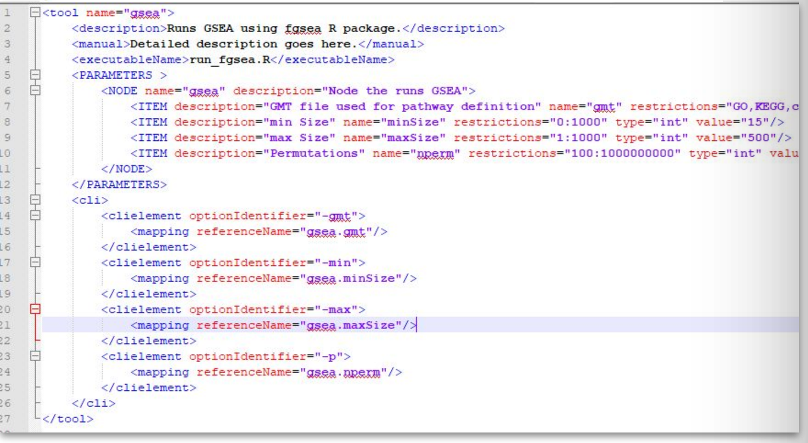
Parameter description:
GMT file used for pathway definition

☐ Show advanced parameter

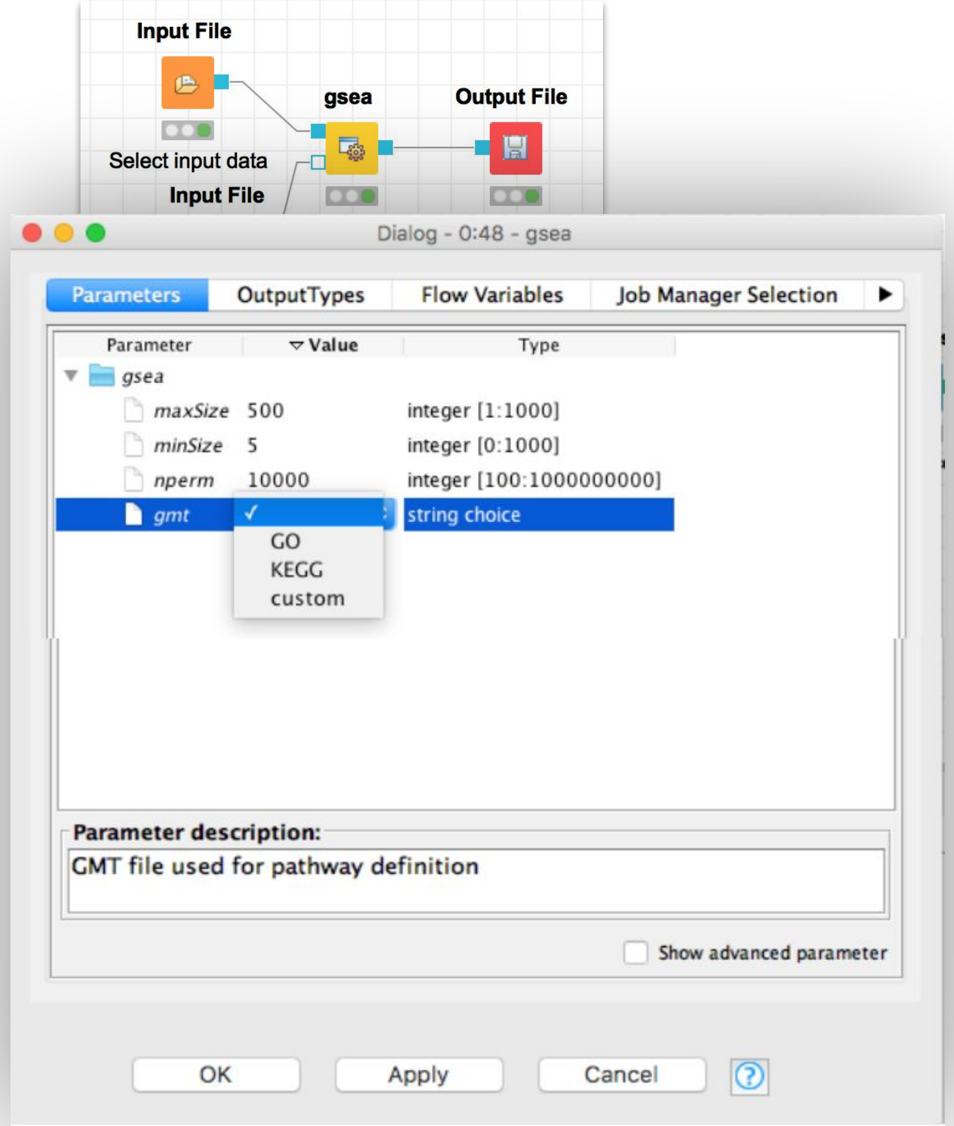
Example



```r
# ------------------------------------------------------------------------------
#' 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)
}
```



```xml
<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>
```

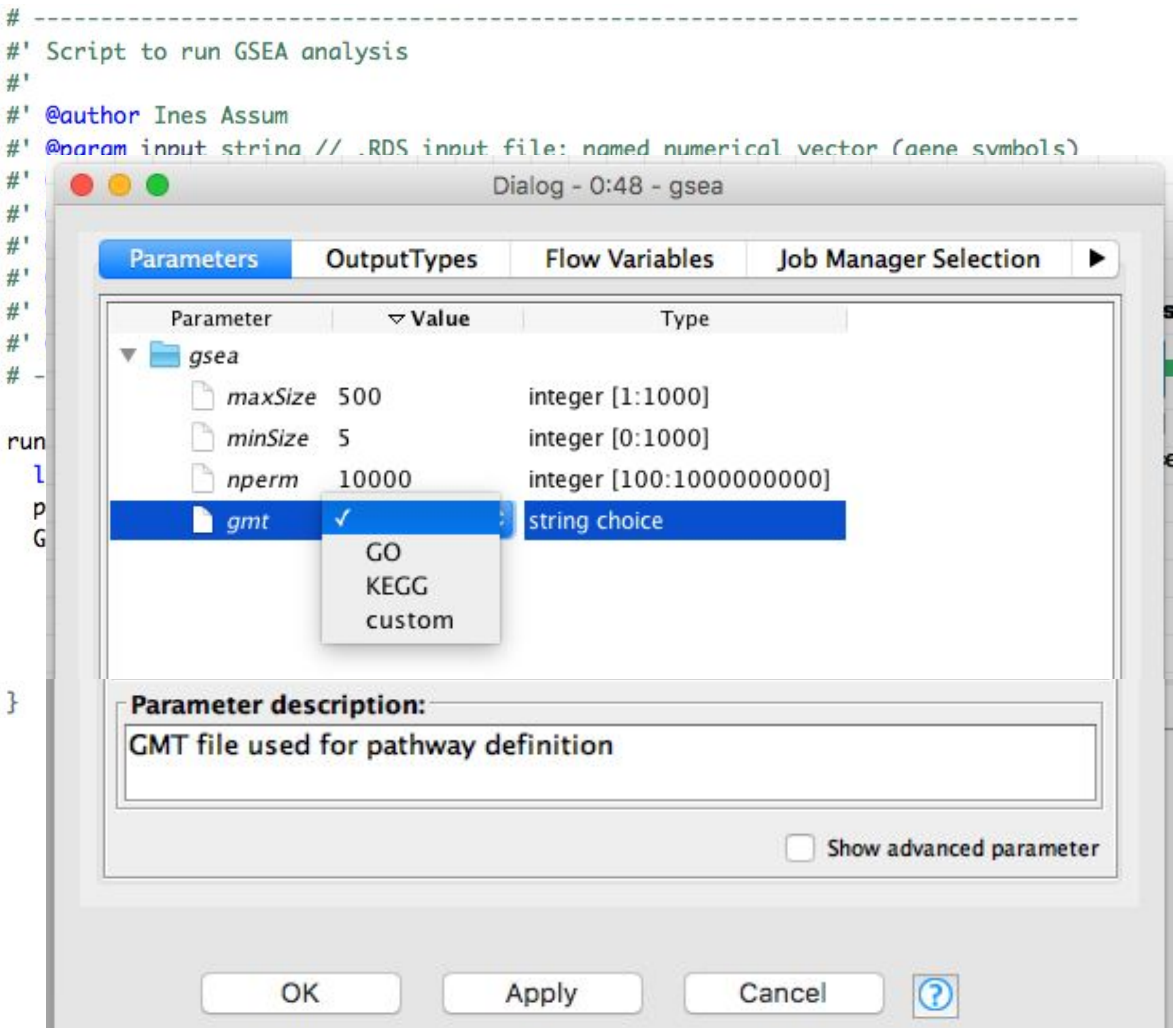


Parameter	Value	Type
gsea		
maxSize	500	integer [1:1000]
minSize	5	integer [0:1000]
nperm	10000	integer [100:1000000000]
gmt	✓	string choice

Parameter description:

GMT file used for pathway definition

☐ Show advanced parameter



```
# ------------------------------------------------------------------------
#' Script to run GSEA analysis
#'
#' @author Ines Assum
#' @param input string // .RDS input file: named numerical vector (gene symbols)
#'
#'
#'
#'
#'
#'
# -

run
  l
  p
  G

}
```

Dialog - 0:48 - gsea

Parameter	Value	Type
gsea		
maxSize	500	integer [1:1000]
minSize	5	integer [0:1000]
nperm	10000	integer [100:1000000000]
gmt	✓	string choice

Parameter description:

GMT file used for pathway definition

□ Show advanced parameter

Summary

- Modular R library integrated into easy-to-use KNIME nodes
- 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

- Check out Benni’s ImmunoNodes: github.com/FRED-2/ImmunoNodes/

Acknowledgements

Epigenereg group

Matthias Heinig

Ines Assum

Thomas Walzthöni

ICB (MONA)

Nikola Müller

Florian Büttner

Andreas Kopf

GenericKnimeNodes **&** **ICB**

Julianus Pfeuffer

Benjamin Schubert

-> ImmunoNodes