

# Package ‘CoSIA’

January 9, 2025

**Type** Package

**Title** An Investigation Across Different Species and Tissues

**Version** 1.6.0

**Description** Cross-Species Investigation and Analysis (CoSIA) is a package that provides researchers with an alternative methodology for comparing across species and tissues using normal wild-type RNA-Seq Gene Expression data from Bgee. Using RNA-Seq Gene Expression data, CoSIA provides multiple visualization tools to explore the transcriptome diversity and variation across genes, tissues, and species. CoSIA uses the Coefficient of Variation and Shannon Entropy and Specificity to calculate transcriptome diversity and variation. CoSIA also provides additional conversion tools and utilities to provide a streamlined methodology for cross-species comparison.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Depends** R (>= 4.3.0), methods (>= 4.3.0), ExperimentHub (>= 2.7.0)

**Imports** dplyr (>= 1.0.7), magrittr (>= 2.0.1), RColorBrewer (>= 1.1-2), tidyr (>= 1.2.0), plotly (>= 4.10.0), stringr (>= 1.4.0), ggplot2 (>= 3.3.5), tibble (>= 3.1.7), org.Hs.eg.db (>= 3.12.0), org.Mm.eg.db (>= 3.12.0), org.Dr.eg.db (>= 3.12.0), org.Ce.eg.db (>= 3.12.0), org.Dm.eg.db (>= 3.12.0), org.Rn.eg.db (>= 3.12.0), AnnotationDbi (>= 1.52.0), biomaRt (>= 2.46.3), homologene (>= 1.4.68.19), annotationTools (>= 1.64.0), readr (>= 2.1.1), tidyselect (>= 1.1.2), stats (>= 4.1.2)

**RoxygenNote** 7.2.3

**Suggests** BiocStyle (>= 2.22.0), tidyverse (>= 1.3.1), knitr (>= 1.42), rmarkdown (>= 2.20), testthat (>= 3.1.6), qpdf (>= 1.3.0)

**biocViews** Software, BiologicalQuestion, GeneExpression, MultipleComparison, ThirdPartyClient, DataImport, GUI

**BugReports** <https://github.com/lasseignelab/CoSIA/issues>

**URL** <https://www.lasseigne.org/>

**VignetteBuilder** knitr

**git\_url** <https://git.bioconductor.org/packages/CoSIA>

**git\_branch** RELEASE\_3\_20

**git\_last\_commit** dfc812d

**git\_last\_commit\_date** 2024-10-29

**Repository** Bioconductor 3.20

**Date/Publication** 2025-01-09

**Author** Anisha Haldar [aut] (<<https://orcid.org/0000-0002-1395-9793>>),  
 Vishal H. Oza [aut] (<<https://orcid.org/0000-0001-6990-0267>>),  
 Amanda D. Clark [cre, aut] (<<https://orcid.org/0000-0002-1186-3114>>),  
 Nathaniel S. DeVoss [aut] (<<https://orcid.org/0000-0003-0465-2770>>),  
 Brittany N. Lasseigne [aut] (<<https://orcid.org/0000-0002-1642-8904>>)

**Maintainer** Amanda D. Clark <[amanda@freshairfamily.org](mailto:amanda@freshairfamily.org)>

## Contents

CoSIAn-class . . . . .	2
CoSIAn-Constructor Constructor for the CoSIAn Class . . . . .	3
getConversion . . . . .	4
getConversion,CoSIAn-method . . . . .	5
getGEx . . . . .	6
getGEx,CoSIAn-method . . . . .	6
getGExMetrics . . . . .	7
getGExMetrics,CoSIAn-method . . . . .	8
getTissues . . . . .	9
plotCVGEx . . . . .	9
plotCVGEx,CoSIAn-method . . . . .	10
plotDSGEx . . . . .	11
plotDSGEx,CoSIAn-method . . . . .	11
plotSpeciesGEx . . . . .	12
plotSpeciesGEx,CoSIAn-method . . . . .	13
plotTissueGEx . . . . .	14
plotTissueGEx,CoSIAn-method . . . . .	14
viewCoSIAn . . . . .	15
viewCoSIAn,CoSIAn-method . . . . .	16
<b>Index</b>	<b>17</b>

---

CoSIAn-class

*CoSIAn Class S4 Object*

---

### Description

This Class is used to specify the information needed to conduct analysis using CoSIA's other methods. An object of this class will pass as an argument to other methods in CoSIA

### Value

CoSIAn object

**Slots**

gene\_set character. A singular gene or a list of genes.  
 i\_species character. The species corresponding to gene\_set.  
 input\_id character. The type of id corresponding to gene\_set.  
 o\_species character. The singular or list of species that the gene set is being converted to.  
 output\_ids character. The singular or list of id types that the gene set is being converted to.  
 mapping\_tool character. The mapping tool, BiomaRt or annotationDBI, being used to map the conversion between IDs.  
 ortholog\_database character. The ortholog database, HomoloGene or NCBIOrtho, being used to map the conversion between species.  
 converted\_id data frame. Output of getConversion.  
 map\_tissues character. A list of tissues being investigated.  
 map\_species character. A list of species being investigated.  
 gex data frame. Output of gene expression data.  
 metric\_type character. A list of possible metric the user wants to calculate.  
 metric data frame. Output of coefficient of variation of gene expression data.

---

CoSIAn-Constructor Constructor for the CoSIAn Class  
*Constructor for the CoSIAn Object*

---

**Description**

The CoSIAn constructor creates a CoSIAn object from character vector(s).

**Usage**

```

CoSIAn(
  gene_set,
  i_species,
  input_id,
  o_species,
  output_ids,
  mapping_tool = "annotationDBI",
  ortholog_database = "HomoloGene",
  map_tissues,
  map_species,
  metric_type
)

```

**Arguments**

gene_set	A singular gene or a list of genes.
i_species	The species corresponding to gene_set.
input_id	The type of id corresponding to gene_set.
o_species	The singular or list of species that the gene set is being converted to.

output_ids	The singular or list of id types that the gene set is being converted to.
mapping_tool	The mapping tool, BiomaRt or annotationDBI, being used to map the conversion between IDs.
ortholog_database	The ortholog database, HomoloGene or NCBIOrtho, being used to map the conversion between species.
map_tissues	A list of tissues being investigated
map_species	A list of species being investigated
metric_type	A list of possible metric the user wants to calculate.

**Value**

An S4 CoSIAn object with character vector(s) as slots.

**Examples**

```
Kidney_Genes <- CoSIAn(
  gene_set = c(
    "ENSG00000008710", "ENSG00000118762",
    "ENSG00000152217"
  ), i_species = "h_sapiens", input_id = "Ensembl_id",
  o_species = c(
    "d_melanogaster", "m_musculus", "h_sapiens", "d_rerio",
    "c_elegans", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = "heart", map_species = c("m_musculus"),
  metric_type = "DS_Gene"
)
```

---

getConversion	<i>getConversion Generic</i>
---------------	------------------------------

---

**Description**

getConversion Generic

**Usage**

```
getConversion(object)
```

**Arguments**

object            CoSIAn object with all user accessible slots filled

**Value**

initializes a generic function for getConversion as preparation for defining the getConversion Method

**Examples**

```
Kidney_Genes <- CoSIAn(
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id",
  o_species = c(
    "d_melanogaster", "m_musculus", "h_sapiens", "d_rerio",
    "c_elegans", "r_norvegicus"
  ),
  output_ids = c("Ensembl_id", "Symbol"), mapping_tool = "annotationDBI",
  ortholog_database = "HomoloGene", map_tissues = "heart",
  map_species = c("m_musculus"), metric_type = "DS_Gene"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
```

---

getConversion, CoSIAn-method

*getConversion Method*

---

**Description**

getConversion Method

**Usage**

```
## S4 method for signature 'CoSIAn'
getConversion(object)
```

**Arguments**

object                    CoSIAn object with all user accessible slots filled

**Value**

CoSIAn object with converted\_id slot filled

**Examples**

```
Kidney_Genes <- CoSIAn(
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id",
  o_species = c(
    "d_melanogaster", "m_musculus", "h_sapiens", "d_rerio",
    "c_elegans", "r_norvegicus"
  ),
  output_ids = c("Ensembl_id", "Symbol"), mapping_tool = "annotationDBI",
  ortholog_database = "HomoloGene", map_tissues = "heart",
  map_species = c("m_musculus"), metric_type = "DS_Gene"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
```

---

getGEx                      *getGEx Generic*

---

**Description**

getGEx Generic

**Usage**

```
getGEx(object)
```

**Arguments**

object                      CoSIAn object with all user accessible slots filled with converted\_id slot filled

**Value**

initializes a generic function for getGEx as preparation for defining the getGEx Method

**Examples**

```
Kidney_Genes <- CoSIAn(
  gene_set = c("ENSG0000008710", "ENSG0000118762", "ENSG0000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id",
  o_species = c(
    "h_sapiens", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = "heart", map_species = c("r_norvegicus"),
  metric_type = "DS_Gene"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
Kidney_gene_gex <- getGEx(Kidney_gene_conversion)
```

---

getGEx, CoSIAn-method      *getGEx Method*

---

**Description**

getGEx Method

**Usage**

```
## S4 method for signature 'CoSIAn'
getGEx(object)
```

**Arguments**

object                      CoSIAn object with all user accessible slots filled with converted\_id slot filled

**Value**

CoSIA object with gex slot filled

**Examples**

```
Kidney_Genes <- CoSIA(
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id",
  o_species = c(
    "h_sapiens", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = "heart", map_species = c("r_norvegicus"),
  metric_type = "DS_Gene"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
Kidney_gene_gex <- getGEx(Kidney_gene_conversion)
```

---

getGExMetrics

*getGExMetrics Generic*

---

**Description**

getGExMetrics Generic

**Usage**

```
getGExMetrics(object)
```

**Arguments**

object                    CoSIA object with all user accessible slots filled with converted\_id slot filled

**Value**

initializes a generic function for getGExMetrics as preparation for defining the getGExMetrics Method

**Examples**

```
Kidney_Genes <- CoSIA(
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id", o_species = c(
    "h_sapiens", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = "heart", map_species = c("r_norvegicus"),
  metric_type = "DS_Gene"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
Kidney_gene_metric <- getGExMetrics(Kidney_gene_conversion)
```

---

getGExMetrics, CoSIAn-method  
*getGExMetrics Method*

---

## Description

getGExMetrics Method

## Usage

```
## S4 method for signature 'CoSIAn'  
getGExMetrics(object)
```

## Arguments

object            CoSIAn object with all user accessible slots filled with converted\_id slot filled

## Value

CoSIAn Object with metric slot filled

## References

Kohl M (2022). MKdescr: Descriptive Statistics. R package version 0.8, <https://github.com/stamats/MKdescr>.

Zhang JD, Hatje K, Sturm G, Broger C, Ebeling M, Burtin M, Terzi F, Pomposiello SI, Badi L (2017). "Detect tissue heterogeneity in gene expression data with BioQC." BMC Genomics, 18(1), 277. <http://accio.github.io/BioQC/>.

## Examples

```
Kidney_Genes <- CoSIAn(  
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),  
  i_species = "h_sapiens", input_id = "Ensembl_id", o_species = c(  
    "h_sapiens", "r_norvegicus"  
  ), output_ids = c("Ensembl_id", "Symbol"),  
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",  
  map_tissues = "heart", map_species = c("r_norvegicus"),  
  metric_type = "DS_Gene"  
)  
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)  
Kidney_gene_metric <- getGExMetrics(Kidney_gene_conversion)
```



---

`getTissues`*getTissues*

---

**Description**`getTissues`**Usage**`getTissues(species)`**Arguments**

<code>species</code>	name of a species or multiple species that you want to get available tissue list for
----------------------	--

**Value**

list of tissues that are common/available among the species or multiple species inputted

**Examples**

```
tissue <- getTissues(c("m_musculus"))
```

---

`plotCVGEx`*plotCVGEx Generic*

---

**Description**`plotCVGEx Generic`**Usage**`plotCVGEx(object)`**Arguments**

<code>object</code>	CoSIAn object with all user accessible slots filled in as well as the <code>converted_id</code> and <code>metric</code> slot filled
---------------------	---

**Value**

initializes a generic function for `plotCVGEx` as preparation for defining the `plotCVGEx Method`

**Examples**

```

Kidney_Genes <- CoSIAn(
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id", o_species = c(
    "h_sapiens", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = c("adult mammalian kidney", "heart"),
  map_species = c("h_sapiens", "r_norvegicus"),
  metric_type = "CV_Tissue"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
Kidney_gene_metric <- getGExMetrics(Kidney_gene_conversion)
plot <- plotCVGEx(Kidney_gene_metric)

```

---

plotCVGEx, CoSIAn-method

*plotCVGEx Method*

---

**Description**

plotCVGEx Method

**Usage**

```

## S4 method for signature 'CoSIAn'
plotCVGEx(object)

```

**Arguments**

object            CoSIAn object with all user accessible slots filled in as well as the converted\_id and metric slot filled

**Value**

plot object

**References**

Dot plot in R with the dotchart function [with examples]. R CODER. (2020, November 20). Retrieved from <https://r-coder.com/dot-plot-r/>

**Examples**

```

Kidney_Genes <- CoSIAn(
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id", o_species = c(
    "h_sapiens", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = c("adult mammalian kidney", "heart"),
  map_species = c("h_sapiens", "r_norvegicus"),
  metric_type = "CV_Tissue"
)

```

```

)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
Kidney_gene_metric <- getGExMetrics(Kidney_gene_conversion)
plot <- plotCVGEx(Kidney_gene_metric)

```

---

plotDSGEx

*plotDSGEx Generic*


---

## Description

plotDSGEx Generic

## Usage

```
plotDSGEx(object)
```

## Arguments

object            CoSIA object with all user accessible slots filled in as well as the converted\_id and metric slot filled

## Value

initializes a generic function for plotDSGEx as preparation for defining the plotDSGEx Method

## Examples

```

Kidney_Genes <- CoSIA(
  gene_set = c("ENSG0000008710", "ENSG0000118762", "ENSG0000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id", o_species = c(
    "d_melanogaster", "m_musculus",
    "h_sapiens", "d_rerio", "c_elegans", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = "heart", map_species = c("m_musculus"),
  metric_type = "DS_Gene"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
Kidney_gene_metric <- getGExMetrics(Kidney_gene_conversion)
plotDSGEx(Kidney_gene_metric)

```

---

plotDSGEx, CoSIA-method

*plotDSGEx Method*


---

## Description

plotDSGEx Method

**Usage**

```
## S4 method for signature 'CoSIAn'
plotDSGEx(object)
```

**Arguments**

object            CoSIAn object with all user accessible slots filled in as well as the converted\_id and metric slot filled

**Value**

plot object

**Examples**

```
Kidney_Genes <- CoSIAn(
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id", o_species = c(
    "d_melanogaster", "m_musculus",
    "h_sapiens", "d_rerio", "c_elegans", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = "heart", map_species = c("m_musculus"),
  metric_type = "DS_Gene"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
Kidney_gene_metric <- getGExMetrics(Kidney_gene_conversion)
plotDSGEx(Kidney_gene_metric)
```

---

plotSpeciesGEx

*plotSpeciesGEx Generic*

---

**Description**

plotSpeciesGEx Generic

**Usage**

```
plotSpeciesGEx(object, single_tissue, single_gene)
```

**Arguments**

object            CoSIAn object with all user accessible slots filled in as well as the converted\_id and gex slot filled

single\_tissue    one tissue that the user wants to investigate across the mapped species

single\_gene      one ensembl id that the user wants to investigate across the mapped species

**Value**

initializes a generic function for plotSpeciesGEx as preparation for defining the plotSpeciesGEx Method

**Examples**

```

Kidney_Genes <- CoSIAn(
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id", o_species = c(
    "d_melanogaster", "m_musculus",
    "h_sapiens", "d_rerio", "c_elegans", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = "heart", map_species = c("m_musculus"),
  metric_type = "DS_Gene"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
Kidney_gene_gex <- getGEx(Kidney_gene_conversion)
plotSpeciesGEx(Kidney_gene_gex, "liver", "ENSG00000008710")

```

---

plotSpeciesGEx, CoSIAn-method

*plotSpeciesGEx Method*

---

**Description**

plotSpeciesGEx Method

**Usage**

```

## S4 method for signature 'CoSIAn'
plotSpeciesGEx(object, single_tissue, single_gene)

```

**Arguments**

object	CoSIAn object with all user accessible slots filled in as well as the converted_id and gex slot filled
single_tissue	one tissue that the user wants to investigate across the mapped species
single_gene	one ensembl id that the user wants to investigate across the mapped species

**Value**

plot object

**Examples**

```

Kidney_Genes <- CoSIAn(
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id", o_species = c(
    "d_melanogaster", "m_musculus",
    "h_sapiens", "d_rerio", "c_elegans", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = "heart", map_species = c("m_musculus"),
  metric_type = "DS_Gene"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
Kidney_gene_gex <- getGEx(Kidney_gene_conversion)
plotSpeciesGEx(Kidney_gene_gex, "liver", "ENSG00000008710")

```

---

plotTissueGEx                      *plotTissueGEx Generic*

---

**Description**

plotTissueGEx Generic

**Usage**

```
plotTissueGEx(object, single_species, single_gene)
```

**Arguments**

object	CoSIAn object with all user accessible slots filled in as well as the converted_id and gex slot filled
single_species	one species that the user wants to investigate across the mapped tissues
single_gene	one ensembl id that the user wants to investigate across the mapped tissues

**Value**

initializes a generic function for plotTissueGEx as preparation for defining the plotTissueGEx Method

**Examples**

```
Kidney_Genes <- CoSIAn(
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id", o_species = c(
    "d_melanogaster", "m_musculus",
    "h_sapiens", "d_rerio", "c_elegans", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = "heart",
  map_species = c("m_musculus"), metric_type = "DS_Gene"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
Kidney_gene_gex <- getGEx(Kidney_gene_conversion)
plotTissueGEx(Kidney_gene_gex, "m_musculus", "ENSG00000008710")
```

---

plotTissueGEx, CoSIAn-method  
*plotTissueGEx Method*

---

**Description**

plotTissueGEx Method

**Usage**

```
## S4 method for signature 'CoSIAn'
plotTissueGEx(object, single_species, single_gene)
```

**Arguments**

object	CoSIAn object with all user accessible slots filled in as well as the converted_id and gex slot filled
single_species	one species that the user wants to investigate across the mapped tissues
single_gene	one ensembl id that the user wants to investigate across the mapped tissues

**Value**

plot object

**Examples**

```
Kidney_Genes <- CoSIAn(
  gene_set = c("ENSG00000008710", "ENSG00000118762", "ENSG00000152217"),
  i_species = "h_sapiens", input_id = "Ensembl_id", o_species = c(
    "d_melanogaster", "m_musculus",
    "h_sapiens", "d_rerio", "c_elegans", "r_norvegicus"
  ), output_ids = c("Ensembl_id", "Symbol"),
  mapping_tool = "annotationDBI", ortholog_database = "HomoloGene",
  map_tissues = "heart", map_species = c("m_musculus"),
  metric_type = "DS_Gene"
)
Kidney_gene_conversion <- CoSIA::getConversion(Kidney_Genes)
Kidney_gene_gex <- getGEx(Kidney_gene_conversion)
plotTissueGEx(Kidney_gene_gex, "m_musculus", "ENSG00000008710")
```

---

viewCoSIAn

*viewCoSIAn Generics*

---

**Description**

viewCoSIAn Generics

**Usage**

```
viewCoSIAn(object, slot_name)
```

**Arguments**

object	CoSIAn object with all user accessible slots filled
slot_name	name of output slots

**Value**

initializes a generic function for viewCoSIAn as preparation for defining the viewCoSIAn Method

**Examples**

```
Kidney_Genes <- CoSIAn(  
  gene_set = c("ENSG00000008710", "ENSG00000152217"),  
  i_species = "h_sapiens", input_id = "Ensembl_id",  
  o_species = c(  
    "d_melanogaster"  
  ),  
  output_ids = c("Ensembl_id", "Symbol"), mapping_tool = "annotationDBI",  
  ortholog_database = "HomoloGene", map_tissues = "heart",  
  map_species = c("d_melanogaster"), metric_type = "DS_Gene"  
)  
viewCoSIAn(Kidney_Genes, "converted_id")
```

---

viewCoSIAn, CoSIAn-method

*viewCoSIAn*

---

**Description**

viewCoSIAn

**Usage**

```
## S4 method for signature 'CoSIAn'  
viewCoSIAn(object, slot_name)
```

**Arguments**

object	CoSIAn object with all user accessible slots filled
slot_name	name of output slots

**Value**

slots in CoSIAn object

**Examples**

```
Kidney_Genes <- CoSIAn(  
  gene_set = c("ENSG00000008710", "ENSG00000152217"),  
  i_species = "h_sapiens", input_id = "Ensembl_id",  
  o_species = c(  
    "d_melanogaster"  
  ),  
  output_ids = c("Ensembl_id", "Symbol"), mapping_tool = "annotationDBI",  
  ortholog_database = "HomoloGene", map_tissues = "heart",  
  map_species = c("d_melanogaster"), metric_type = "DS_Gene"  
)  
viewCoSIAn(Kidney_Genes, "converted_id")
```



# Index

CoSIAn (CoSIAn-Constructor Constructor  
for the CoSIAn Class), [3](#)  
CoSIAn-class, [2](#)  
CoSIAn-Constructor Constructor for the  
CoSIAn Class, [3](#)

[getConversion](#), [4](#)  
[getConversion](#), CoSIAn-method, [5](#)  
[getGEx](#), [6](#)  
[getGEx](#), CoSIAn-method, [6](#)  
[getGExMetrics](#), [7](#)  
[getGExMetrics](#), CoSIAn-method, [8](#)  
[getTissues](#), [9](#)

[plotCVGEx](#), [9](#)  
[plotCVGEx](#), CoSIAn-method, [10](#)  
[plotDSGEx](#), [11](#)  
[plotDSGEx](#), CoSIAn-method, [11](#)  
[plotSpeciesGEx](#), [12](#)  
[plotSpeciesGEx](#), CoSIAn-method, [13](#)  
[plotTissueGEx](#), [14](#)  
[plotTissueGEx](#), CoSIAn-method, [14](#)

[viewCoSIAn](#), [15](#)  
[viewCoSIAn](#), CoSIAn-method, [16](#)