

Projekt Zaliczeniowy

Generated by Doxygen 1.9.3

1 File Index	1
1.1 File List	1
2 File Documentation	3
2.1 C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt_Zaliczeniowy/Projekt_Zaliczeniowy/exporter.h File Reference	3
2.1.1 Detailed Description	4
2.1.2 Function Documentation	4
2.1.2.1 exportClustered()	4
2.1.2.2 exportData()	4
2.1.2.3 exportFile()	5
2.1.2.4 exportVector()	6
2.2 exporter.h	6
2.3 C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt_Zaliczeniowy/Projekt_Zaliczeniowy/functions.h File Reference	7
2.3.1 Detailed Description	8
2.3.2 Function Documentation	8
2.3.2.1 allocate()	8
2.3.2.2 cluster()	9
2.3.2.3 distance()	10
2.3.2.4 initClustering()	10
2.3.2.5 minIndex()	11
2.3.2.6 normalized()	11
2.3.2.7 updateClustering()	12
2.3.2.8 updateMeans()	13
2.4 functions.h	14
2.5 C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt_Zaliczeniowy/Projekt_Zaliczeniowy/importer.h File Reference	15
2.5.1 Detailed Description	15
2.5.2 Function Documentation	15
2.5.2.1 readFile()	15
2.6 importer.h	16
2.7 C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt_Zaliczeniowy/Projekt_Zaliczeniowy/screen.h File Reference	16
2.7.1 Detailed Description	17
2.7.2 Function Documentation	17
2.7.2.1 showClustered()	17
2.7.2.2 showData()	17
2.7.2.3 showException()	18
2.7.2.4 showVector()	18
2.8 screen.h	19
2.9 C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt_Zaliczeniowy/Projekt_Zaliczeniowy/validator.h File Reference	19
2.9.1 Detailed Description	20

2.9.2 Function Documentation	20
2.9.2.1 isNumber()	20
2.9.2.2 validateArguments()	20
2.10 validator.h	21
Index	23

Chapter 1

File Index

1.1 File List

Here is a list of all documented files with brief descriptions:

C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/[exporter.h](#)
3

C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/[functions.h](#)
7

C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/[importer.h](#)
15

C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/[screen.h](#)
16

C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/[validator.h](#)
19

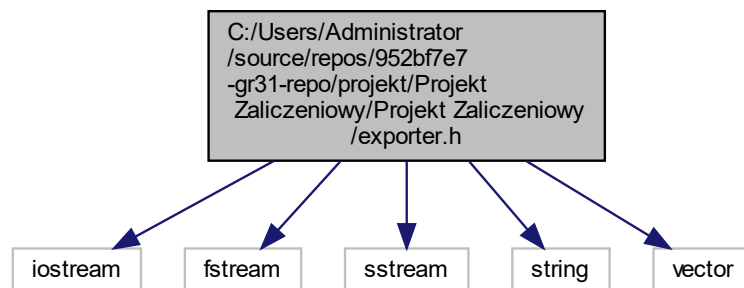
Chapter 2

File Documentation

2.1 C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/exporter.h File Reference

```
#include <iostream>
#include <fstream>
#include <sstream>
#include <string>
#include <vector>
```

Include dependency graph for exporter.h:



Functions

- void `exportFile` (std::string fileName, const std::vector< std::vector< double > > &rawData, const std::vector< int > &clustering, int numClusters, int decimals)
Exports data to output file in format of txt with final form of clusters grouped.
- void `exportData` (std::ofstream &myFile, const std::vector< std::vector< double > > &data, int decimals, bool indices, bool newLine)
Exports data to exportFile function.
- void `exportVector` (std::ofstream &myFile, const std::vector< int > &vector, bool newLine)
Exports groups of splitted clusters to ountput file.
- void `exportClustered` (std::ofstream &myFile, const std::vector< std::vector< double > > &data, const std::vector< int > &clustering, int numClusters, int decimals)
Export cluster resolution to output file.

2.1.1 Detailed Description

Plik nagłówekowy...

2.1.2 Function Documentation

2.1.2.1 exportClustered()

```
void exportClustered (
    std::ofstream & myFile,
    const std::vector< std::vector< double > > & data,
    const std::vector< int > & clustering,
    int numClusters,
    int decimals )
```

Export cluster resolution to output file.

Parameters

<i>myFile</i>	opened output file
<i>data</i>	capacitor for data
<i>clustering</i>	final groups
<i>numClusters</i>	number of clusters to split for data
<i>decimals</i>	decimal digits after point

Here is the caller graph for this function:



2.1.2.2 exportData()

```
void exportData (
    std::ofstream & myFile,
    const std::vector< std::vector< double > > & data,
    int decimals,
    bool indices,
    bool newLine )
```

Exports data to exportFile function.

Parameters

<i>myFile</i>	opened output file
<i>data</i>	capacitor for data
<i>decimals</i>	decimal digits after point
<i>indices</i>	indexes of data line in file
<i>newLine</i>	new line in file.txt

Here is the caller graph for this function:



2.1.2.3 exportFile()

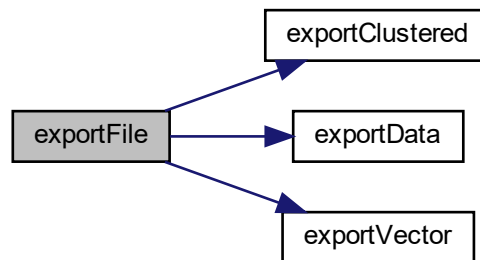
```
void exportFile (
    std::string fileName,
    const std::vector< std::vector< double > > & rawData,
    const std::vector< int > & clustering,
    int numClusters,
    int decimals )
```

Exports data to output file in format of txt with final form of clusters grouped.

Parameters

<i>fileName</i>	name of output file
<i>rawData</i>	imported data
<i>clustering</i>	final groups
<i>numClusters</i>	number of clusters to split for data

Here is the call graph for this function:



2.1.2.4 exportVector()

```

void exportVector (
    std::ofstream & myFile,
    const std::vector< int > & vector,
    bool newLine )
  
```

Exports groups of splitted clusters to ountput file.

Parameters

<i>myFile</i>	opened output file
<i>vector</i>	data represented for line in file
<i>newLine</i>	new line in file.txt

Here is the caller graph for this function:



2.2 exporter.h

[Go to the documentation of this file.](#)

```

1
2 #pragma once
3 #include <iostream>
4 #include <fstream>
5 #include <sstream>
6 #include <string>
7 #include <vector>
8
9
17 void exportFile(std::string fileName, const std::vector<std::vector<double>>& rawData, const
    std::vector<int>& clustering, int numClusters, int decimals);
18
27 void exportData(std::ofstream& myFile, const std::vector<std::vector<double>>& data, int decimals, bool
    indices, bool newLine);
28
35 void exportVector(std::ofstream& myFile, const std::vector<int>& vector, bool newLine);
36
45 void exportClustered(std::ofstream& myFile, const std::vector<std::vector<double>>& data, const
    std::vector<int>& clustering, int numClusters, int decimals);

```

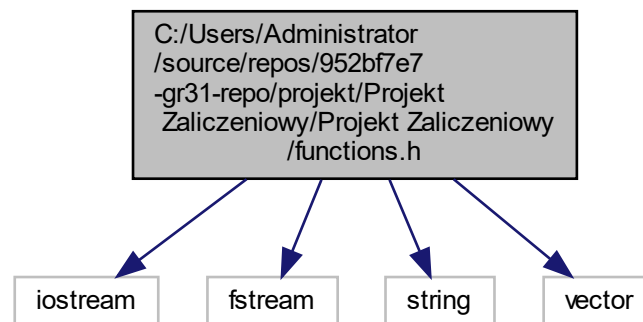
2.3 C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/functions.h File Reference

```

#include <iostream>
#include <fstream>
#include <string>
#include <vector>

```

Include dependency graph for functions.h:



Functions

- `std::vector< int > cluster` (`const std::vector< std::vector< double > > &rawData`, `int numClusters`)
Main clustering function.
- `std::vector< std::vector< double > > normalized` (`const std::vector< std::vector< double > > &rawData`)
Normalizing input data using standard deviation.
- `std::vector< int > initClustering` (`int numTuples`, `int numClusters`, `int seed`)
Propose random clustering for data.
- `std::vector< std::vector< double > > allocate` (`int numClusters`, `int numColumns`)
Prepare memory for data.

- bool [updateMeans](#) (const std::vector< std::vector< double > > &data, std::vector< int > &clustering, std::vector< std::vector< double > > &means)
Calculate centroids for cluster groups.
- bool [updateClustering](#) (const std::vector< std::vector< double > > &data, std::vector< int > &clustering, const std::vector< std::vector< double > > &means)
Calculating new clusters grouping based on new centroids.
- double [distance](#) (const std::vector< double > &tuple, const std::vector< double > &mean)
Calculate distances between vectors.
- int [minIndex](#) (const std::vector< double > &distances)
Get the closest centroid.

2.3.1 Detailed Description

Plik nagłowy...

2.3.2 Function Documentation

2.3.2.1 `allocate()`

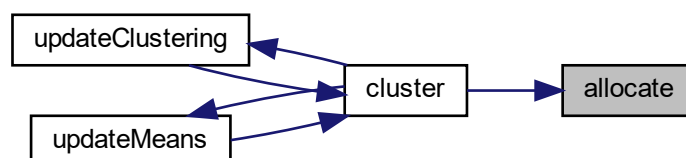
```
std::vector< std::vector< double > > allocate (
    int numClusters,
    int numColumns )
```

Prepare memory for data.

Parameters

<i>numClusters</i>	number of clusters to split for data
<i>numColumns</i>	number of columns in file

Here is the caller graph for this function:



2.3.2.2 cluster()

```
std::vector< int > cluster (
    const std::vector< std::vector< double > > & rawData,
    int numClusters )
```

Main clustering function.

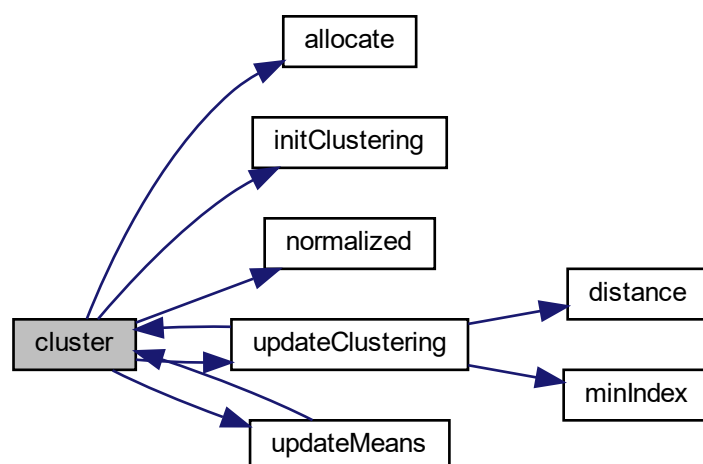
Parameters

<i>rawData</i>	imported unclustered data
<i>numClusters</i>	number of clusters to split for data

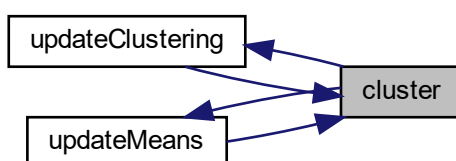
Copyright

Copyright 2022 Jakub Dudek.

Here is the call graph for this function:



Here is the caller graph for this function:



2.3.2.3 distance()

```
double distance (
    const std::vector< double > & tuple,
    const std::vector< double > & mean )
```

Calculate distances between vectors.

Parameters

<i>tuple</i>	one row from file
<i>mean</i>	current mean

Here is the caller graph for this function:



2.3.2.4 initClustering()

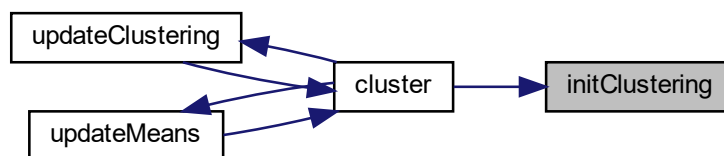
```
std::vector< int > initClustering (
    int numTuples,
    int numClusters,
    int seed )
```

Propose random clustering for data.

Parameters

<i>numTuples</i>	number of rows in file
<i>numClusters</i>	number of clusters to split for data
<i>seed</i>	pseudo random number of splitting configuration

Here is the caller graph for this function:



2.3.2.5 minIndex()

```
int minIndex (
    const std::vector< double > & distances )
```

Get the closest centroid.

Parameters

<i>distances</i>	distances from all centroids
------------------	------------------------------

Here is the caller graph for this function:



2.3.2.6 normalized()

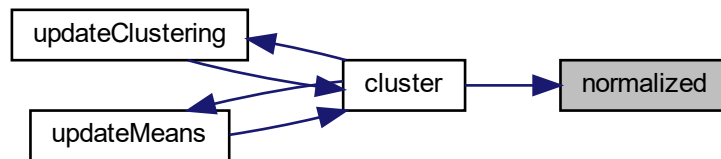
```
std::vector< std::vector< double > > normalized (
    const std::vector< std::vector< double > > & rawData )
```

Normalizing input data using standard deviation.

Parameters

<i>rawData</i>	imported unclustered data
----------------	---------------------------

Here is the caller graph for this function:



2.3.2.7 updateClustering()

```

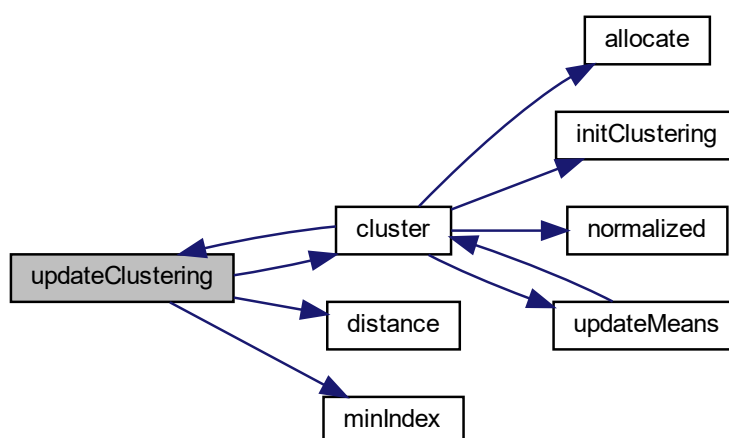
bool updateClustering (
    const std::vector< std::vector< double > > & data,
    std::vector< int > & clustering,
    const std::vector< std::vector< double > > & means )
  
```

Calculating new clusters grouping based on new centroids.

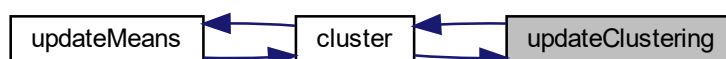
Parameters

<i>data</i>	data from file
<i>clustering</i>	clustering from previous iteration
<i>means</i>	new centroids calculated

Here is the call graph for this function:



Here is the caller graph for this function:



2.3.2.8 updateMeans()

```

bool updateMeans (
    const std::vector< std::vector< double > > & data,
    std::vector< int > & clustering,
    std::vector< std::vector< double > > & means )

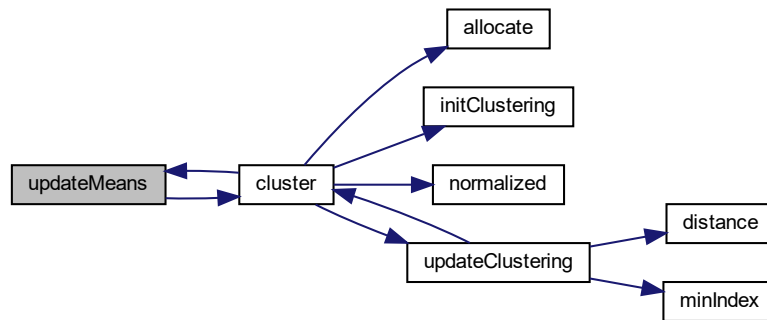
```

Calculate centroids for cluster groups.

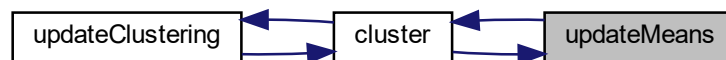
Parameters

<i>data</i>	data from file
<i>clustering</i>	clustering from previous iteration
<i>means</i>	old means - centroids

Here is the call graph for this function:



Here is the caller graph for this function:



2.4 functions.h

[Go to the documentation of this file.](#)

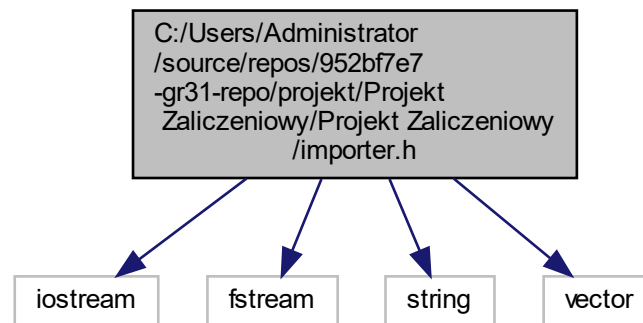
```

1
2 #pragma once
3 #include <iostream>
4 #include <fstream>
5 #include <string>
6 #include <vector>
7
8
14 std::vector<int> cluster(const std::vector<std::vector<double>>& rawData, int numClusters);
15
20 std::vector<std::vector<double>> normalized(const std::vector<std::vector<double>>& rawData);
21
28 std::vector<int> initClustering(int numTuples, int numClusters, int seed);
29
35 std::vector<std::vector<double>> allocate(int numClusters, int numColumns);
36
43 bool updateMeans(const std::vector<std::vector<double>>& data, std::vector<int>& clustering,
    std::vector<std::vector<double>>& means);
44
51 bool updateClustering(const std::vector<std::vector<double>>& data, std::vector<int>& clustering, const
    std::vector<std::vector<double>>& means);
52
58 double distance(const std::vector<double>& tuple, const std::vector<double>& mean);
59
64 int minIndex(const std::vector<double>& distances);

```

2.5 C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/importer.h File Reference

```
#include <iostream>
#include <fstream>
#include <string>
#include <vector>
Include dependency graph for importer.h:
```



Functions

- `std::vector< std::vector< double > > readFile (std::string fileName)`
Imports unclustered data from input file in format of txt.

2.5.1 Detailed Description

Plik nagłówekowy...

2.5.2 Function Documentation

2.5.2.1 readFile()

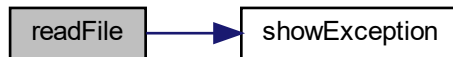
```
std::vector< std::vector< double > > readFile (
    std::string fileName )
```

Imports unclustered data from input file in format of txt.

Parameters

<i>fileName</i>	filepath with data
-----------------	--------------------

Here is the call graph for this function:



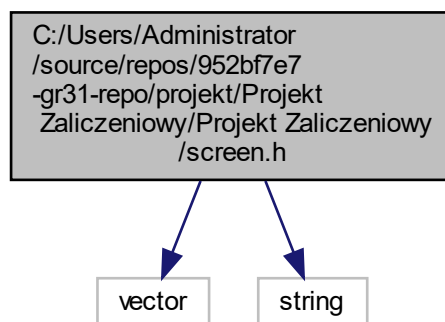
2.6 importer.h

[Go to the documentation of this file.](#)

```
1
2 #pragma once
3 #include <iostream>
4 #include <fstream>
5 #include <string>
6 #include <vector>
7
8
13 std::vector<std::vector<double>> readFile(std::string fileName);
```

2.7 C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/↵ Projekt Zaliczeniowy/Projekt Zaliczeniowy/screen.h File Reference

```
#include <vector>
#include <string>
Include dependency graph for screen.h:
```



Functions

- void `showData` (const std::vector< std::vector< double > > &data, int decimals, bool indices, bool newLine)
Displays whole algorithm with results in console.
- void `showVector` (const std::vector< int > &vector, bool newLine)
Displays vector as a line in console.
- void `showClustered` (const std::vector< std::vector< double > > &data, const std::vector< int > &clustering, int numClusters, int decimals)
Diplays clustered resolution.
- void `showException` (std::string message)
Showing finded exception in input file
- void `showUsage` ()
Instruction of program usage.

2.7.1 Detailed Description

Plik nagłówkowy...

2.7.2 Function Documentation

2.7.2.1 showClustered()

```
void showClustered (  
    const std::vector< std::vector< double > > & data,  
    const std::vector< int > & clustering,  
    int numClusters,  
    int decimals )
```

Diplays clustered resolution.

Parameters

<i>data</i>	input data
<i>clustering</i>	final clusters
<i>numClusters</i>	number of clusters to split data
<i>decimals</i>	decimal digits after point

2.7.2.2 showData()

```
void showData (  
    const std::vector< std::vector< double > > & data,  
    int decimals,
```

```
bool indices,  
bool newLine )
```

Displays whole algorithm with results in console.

Parameters

<i>data</i>	input data from file
<i>decimals</i>	decimals points in double
<i>indices</i>	indexes of rows of data in file
<i>newLine</i>	should add new line at the end

2.7.2.3 showException()

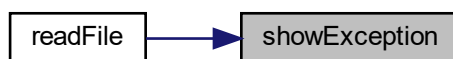
```
void showException (  
    std::string message )
```

Showing finded exception in input file

Parameters

<i>message</i>	message shown in console
----------------	--------------------------

Here is the caller graph for this function:



2.7.2.4 showVector()

```
void showVector (  
    const std::vector< int > & vector,  
    bool newLine )
```

Displays vector as a line in console.

Parameters

<i>vector</i>	display one vector as a line
<i>newLine</i>	should add new line at the end

2.8 screen.h

[Go to the documentation of this file.](#)

```

1
2 #pragma once
3 #include <vector>
4 #include <string>
5
6
14 void showData(const std::vector<std::vector<double>>& data, int decimals, bool indices, bool newLine);
15
21 void showVector(const std::vector<int>& vector, bool newLine);
22
30 void showClustered(const std::vector<std::vector<double>>& data, const std::vector<int>& clustering, int
    numClusters, int decimals);
31
36 void showException(std::string message);
37
41 void showUsage();

```

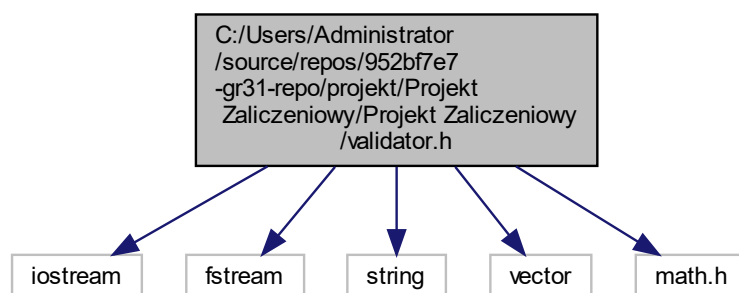
2.9 C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/↵ Projekt Zaliczeniowy/Projekt Zaliczeniowy/validator.h File Reference

```

#include <iostream>
#include <fstream>
#include <string>
#include <vector>
#include <math.h>

```

Include dependency graph for validator.h:



Functions

- bool `validateArguments` (std::string inputFile, std::string outputFile, std::string clusterNumberString, std::string dimensionsNumberString)
Validate arguments for their right forms.
- bool `isNumber` (const std::string &str)
Check if string is number.

2.9.1 Detailed Description

Plik nagłówekowy...

2.9.2 Function Documentation

2.9.2.1 isNumber()

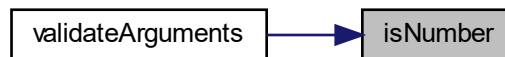
```
bool isNumber (
    const std::string & str )
```

Check if string is number.

Parameters

<i>str</i>	string to beeing checked
------------	--------------------------

Here is the caller graph for this function:



2.9.2.2 validateArguments()

```
bool validateArguments (
    std::string inputFile,
    std::string outputFile,
    std::string clusterNumberString,
    std::string dimensionsNumberString )
```

Validate arugmetns for they rights forms.

Parameters

<i>inputFile</i>	given input file
<i>outputFile</i>	given output file
<i>clusterNumberString</i>	number of clusters to group
<i>dimensionsNumberString</i>	number of dimensions of one row in file

Here is the call graph for this function:



2.10 validator.h

[Go to the documentation of this file.](#)

```
1
2 #pragma once
3 #include <iostream>
4 #include <fstream>
5 #include <string>
6 #include <vector>
7 #include <math.h>
8
9
17 bool validateArguments(std::string inputFile, std::string outputFile, std::string clusterNumberString,
    std::string dimensionsNumberString);
22 bool isNumber(const std::string& str);
```


Index

allocate
 functions.h, 8

C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/exporter.h, 3, 6

C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/functions.h, 7, 14

C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/importer.h, 15, 16

C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/screen.h, 16, 19

C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/validator.h, 19, 21

cluster
 functions.h, 8

distance
 functions.h, 10

exportClustered
 exporter.h, 4

exportData
 exporter.h, 4

exporter.h
 exportClustered, 4
 exportData, 4
 exportFile, 5
 exportVector, 6

exportFile
 exporter.h, 5

exportVector
 exporter.h, 6

functions.h
 allocate, 8
 cluster, 8
 distance, 10
 initClustering, 10
 minIndex, 11
 normalized, 11
 updateClustering, 12
 updateMeans, 13

importer.h
 readFile, 15

initClustering
 functions.h, 10

isNumber
 validator.h, 20

minIndex
 functions.h, 11

normalized
 functions.h, 11

readFile
 importer.h, 15

screen.h
 showClustered, 17
 showData, 17
 showException, 18
 showVector, 18

showClustered
 screen.h, 17

showData
 screen.h, 17

showException
 screen.h, 18

showVector
 screen.h, 18

updateClustering
 functions.h, 12

updateMeans
 functions.h, 13

validateArguments
 validator.h, 20

validator.h
 isNumber, 20
 validateArguments, 20