Projekt Zaliczeniowy

Generated by Doxygen 1.9.3

Chapter 1

File Index

1.1 File List

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

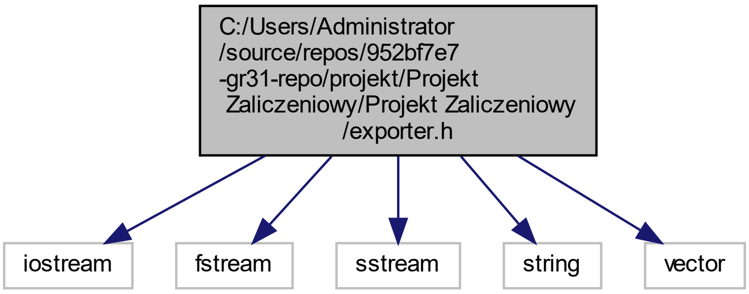
Chapter 2

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:


C:/Users/Administrator/source/repos/952bf7e7-gr31-repo/projekt/Projekt Zaliczeniowy/Projekt Zaliczeniowy/exporter.h
iostream
fstream
sstream
string
vector

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 nagłowkowy...

2.1.2 Function Documentation

2.1.2.1 exportClustered()

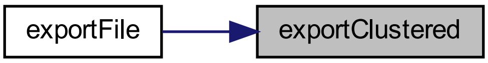
```
void exportClustered (
            std::ofstream & myFile,
            const std::vector< std::vector< double > > & data,
            const std::vector< int > & clustering,
            int numClusters,
            int decimals )
```

Export cluster resolution to output file.

Parameters

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

Here is the caller graph for this function:



2.1.2.2 exportData()

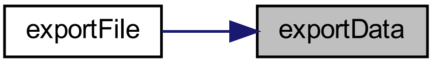
```
void exportData (
            std::ofstream & myFile,
            const std::vector< std::vector< double > > & data,
            int decimals,
            bool indices,
            bool newLine )
```

Exports data to exportFile function.

Parameters

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

Here is the caller graph for this function:



2.1.2.3 exportFile()

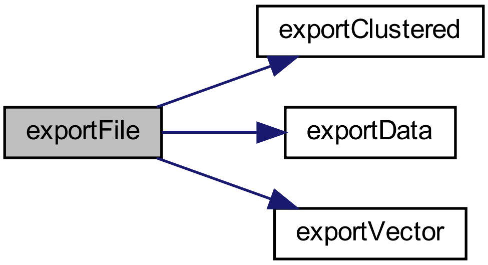
```
void exportFile (
            std::string fileName,
            const std::vector< std::vector< double > > & rawData,
            const std::vector< int > & clustering,
            int numClusters,
            int decimals )
```

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

Parameters

fileName	name of outnput file
rawData	imported data
clustering	final groups
numClusters	number of clusters to split for data

Here is the call graph for this function:



2.1.2.4 exportVector()

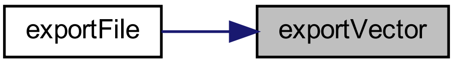
```
void exportVector (
             std::ofstream & myFile,
             const std::vector< int > & vector,
             bool newLine )
```

Exports groups of splitted clusters to ountput file.

Parameters

myFile	opened output file
vector	data represented for line in file
newLine	new line in file.txt

Here is the caller graph for this function:



2.2 exporter.h

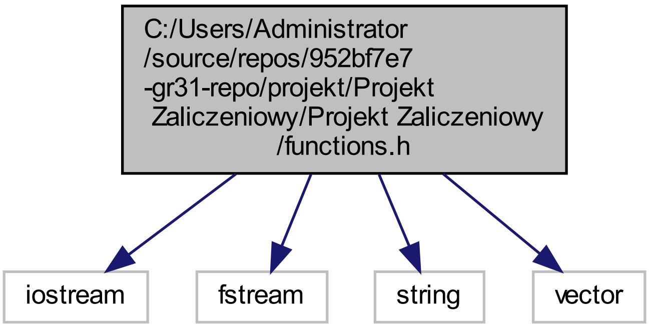
Go to the documentation of this file.

```

#pragma once
#include <iostream>
#include <fstream>
#include <sstream>
#include <string>
#include <vector>


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

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

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

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)

  Normalizating 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 nagłowkowy...

2.3.2 Function Documentation

2.3.2.1 allocate()

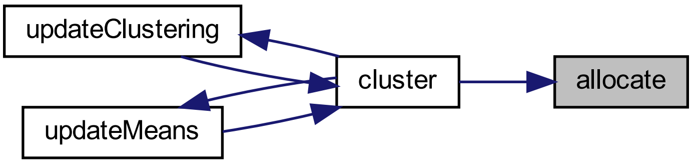
```
std::vector< std::vector< double > > allocate (
            int numClusters,
            int numColumns )
```

Prepare memory for data.

Parameters

numClusters	number of clusters to split for data
numColumns	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.

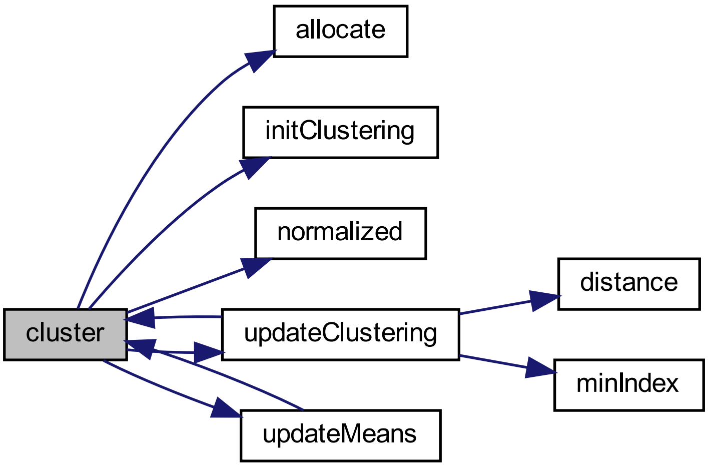
Parameters

rawData	imported unclustered data
numClusters	number of clusters to split for data

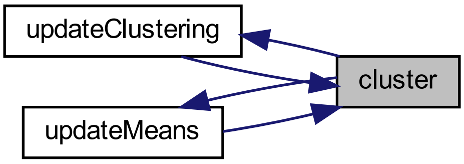
Copyright

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()

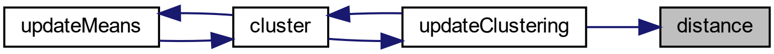
```
double distance (
             const std::vector< double > & tuple,
             const std::vector< double > & mean )
```

Calculate distances between vectors.

Parameters

tuple	one row from file
mean	current mean

Here is the caller graph for this function:



2.3.2.4 initClustering()

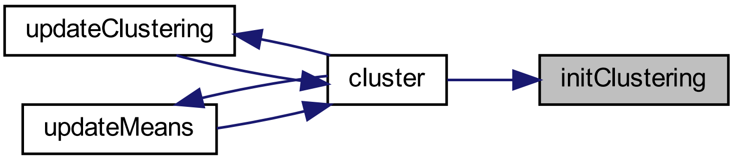
```
std::vector< int > initClustering (
             int numTuples,
             int numClusters,
             int seed )
```

Propose random clustering for data.

Parameters

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

Here is the caller graph for this function:



2.3.2.5 minIndex()

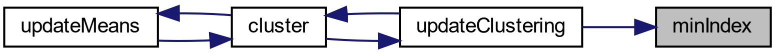
```
int minIndex (
            const std::vector< double > & distances )
```

Get the closest centroid.

Parameters

distances	distances from all centroids

Here is the caller graph for this function:



2.3.2.6 normalized()

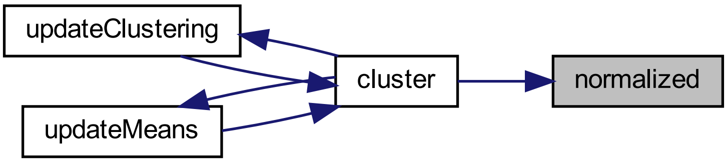
```
std::vector< std::vector< double > > normalized (
            const std::vector< std::vector< double > > & rawData )
```

Normalizating input data using standard deviation.

Parameters

rawData	imported unclustered data

Here is the caller graph for this function:



2.3.2.7 updateClustering()

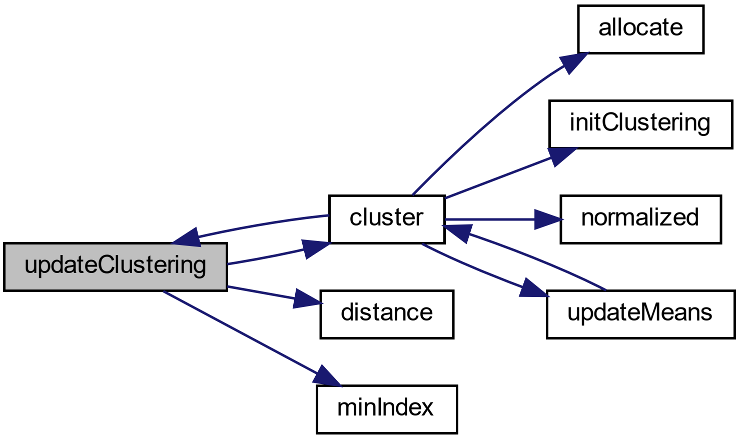
```
bool updateClustering (
            const std::vector< std::vector< double > > & data,
            std::vector< int > & clustering,
            const std::vector< std::vector< double > > & means )
```

Calculating new clusters grouping based on new centroids.

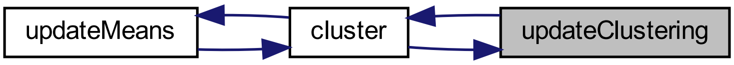
Parameters

data	data from file
clustering	clustering from previous iteration
means	new centroinds calculated

Here is the call graph for this function:



Here is the caller graph for this function:



2.3.2.8 updateMeans()

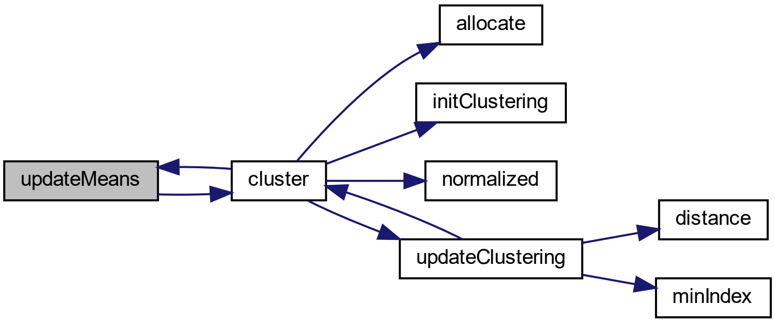
```
bool updateMeans (
            const std::vector< std::vector< double > > & data,
            std::vector< int > & clustering,
            std::vector< std::vector< double > > & means )
```

Calculate centroids for cluster groups.

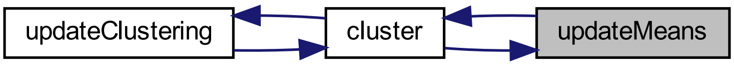
Parameters

data	data from file
clustering	clustering from previous iteration
means	old means - centroids

Here is the call graph for this function:



Here is the caller graph for this function:



2.4 functions.h

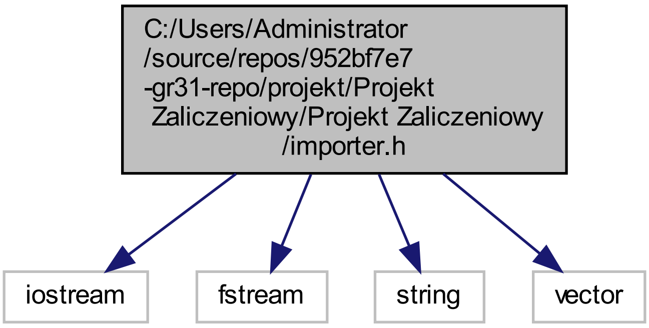
Go to the documentation of this file.

```

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


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

std::vector<std::vector<double» normalized(const std::vector<std::vector<double»& rawData);

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

std::vector<std::vector<double» allocate(int numClusters, int numColumns);

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

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

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

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 nagłowkowy...

2.5.2 Function Documentation

2.5.2.1 readFile()

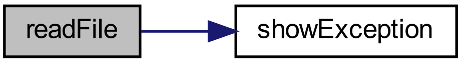
```
std::vector< std::vector< double > > readFile (
            std::string fileName )
```

Imports unclustered data from input file in format of txt.

Parameters

fileName	filepath with data

Here is the call graph for this function:



2.6 importer.h

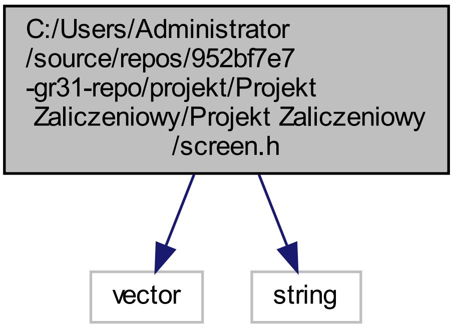
Go to the documentation of this file.

```

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


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 algoritm 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 nagłowkowy...

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

data	input data
clustering	final clusters
numClusters	number of clusters to split data
decimals	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 algoritm with results in console.

Parameters

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

2.7.2.3 showException()

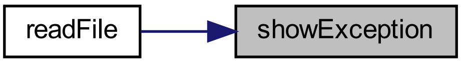
```
void showException (
             std::string message )
```

Showing finded exception in input file

Parameters

message	massage 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

vector	display one vector as a line
newLine	should add new line at the end

2.8 screen.h

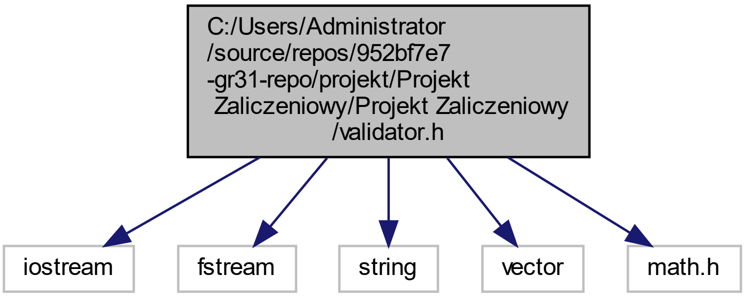
Go to the documentation of this file.

```

#pragma once
#include <vector>
#include <string>


void showData(const std::vector<std::vector<double»& data, int decimals, bool indices, bool newLine);

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

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

void showException(std::string message);

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 arugmetns for they rights forms.
- bool isNumber (const std::string &str)

    Check if string is number.

2.9.1 Detailed Description

Plik nagłowkowy...

2.9.2 Function Documentation

2.9.2.1 isNumber()

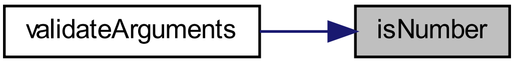
```
bool isNumber (
            const std::string & str )
```

Check if string is number.

Parameters

str	string to beeing checked

Here is the caller graph for this function:



2.9.2.2 validateArguments()

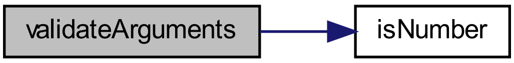
```
bool validateArguments (
            std::string inputFile,
            std::string outputFile,
            std::string clusterNumberString,
            std::string dimensionsNumberString )
```

Validate arugmetns for they rights forms.

Parameters

inputFile	given input file
outputFile	given output file
clusterNumberString	number of clusters to group
dimensionsNumberString	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.

```

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


bool validateArguments(std::string inputFile, std::string outputFile, std::string clusterNumberString,
      std::string dimensionsNumberString);
bool isNumber(const std::string& str);
```

Index