

Flood:
An Open Source Neural Networks C++ Library.
www.cimne.com/flood
Installation Guide
Version 2

Roberto Lopez
International Center for Numerical Methods in Engineering (CIMNE)
Barcelona, Spain
E-mail: rlopez@cimne.upc.edu

July 1, 2008

Contents

1	Installing Flood on Linux	2
2	Installing Flood on Windows	3
2.1	Microsoft Visual C++	3
2.2	Bloodshed Dev-C++	4
2.3	Other Compilers	5

1 Installing Flood on Linux

Compilation of Flood in Linux is straight-forward, since none external package is used and simple makefiles are here provided. In order to do that, the following steps must be performed:

1. Extract the "Flood2.zip" file to the installation folder.

To install Flood from the download location `$DOWNLOAD_DIRECTORY` into the installation location `$INSTALLATION_DIRECTORY` use the following commands:

```
>cd $DOWNLOAD_DIRECTORY
>unzip Flood2.zip $INSTALLATION_DIRECTORY
```

You can specify any name for the installation folder, but the name `Flood2` is recommended. Subsequent instructions in this guide refer to this directory as `$FLOOD2`.

2. Run the installation Makefile.

The folder `\$FLOOD2\Makefiles` contains a set of makefiles for the different applications (main functions) included with Flood.

To run the installation makefile type the following commands on the terminal:

```
>cd \$FLOOD2\Makefiles
>make
```

This compiles all the classes included in Flood and builds the `TemplateApplication.cpp` contained in the `Applications` folder.

In order to build more particular applications, for instance the `MultilayerPerceptronApplication.cpp`, type the following:

```
>make -f MultilayerPerceptronMakefile
```

Doing this, only the classes which are needed for that application will be compiled and the `MultilayerPerceptronApplication.cpp` contained in the `Applications` folder will be built.

3. Verify the installation.

To verify the installation, run the Flood executable:

```
$/Flood
```

If nothing has been wrong, the following message should appear on the terminal:

```
Flood Neural Network. Template Application.
```

To create your own application just use the `TemplateApplication.cpp` file in the Applications folder and the Makefile file in the Makefiles folder as templates.

4. Removing a Flood Installation

To remove a Flood installation, enter the following command on the terminal:

```
>rm -rf \${FLOOD2}
```

This will delete the whole Flood folder.

2 Installing Flood on Windows

Compiling Flood on Windows is also easy. No external packages are to be installed and the library comes with project files for Microsoft Visual C++ and Bloodshed Dev-C++, for the latest versions of that compilers at the Flood release date. When working with another compiler is needed, a project file for it must be created.

2.1 Microsoft Visual C++

Microsoft Visual C++ 2008 Express Edition is a free, lightweight, easy-to-use, and easy-to-learn tools for the hobbyist, novice, and student developer. It can be downloaded at

<http://msdn.microsoft.com/visualc>

Flood includes the `Flood.vcproj` project file for that compiler in the `VisualCppProject` folder.

To open the Flood project just double click on that file. A similar window than that depicted in Figure 1 should come up.

Pressing `Ctrl+F5` will compile, build and run the `TemplateApplication.cpp` contained in the Applications folder.

A MS-DOS console should appear with the following message:

```
Flood Neural Network. Template Application.
```

A complete set of application examples (main functions) can be found in the Applications folder. In order to run one of them just replace the `TemplateApplication.cpp` virtual file in the tree view of the Visual C++ project window.

Note that project files of other versions than Visual C++ 2008 Express Edition are not guaranteed to be opened. In that case a new project should be created. Also, it has been found that very old versions of this compiler are not ANSI C++ compliant, and upgrading to a new one becomes necessary.

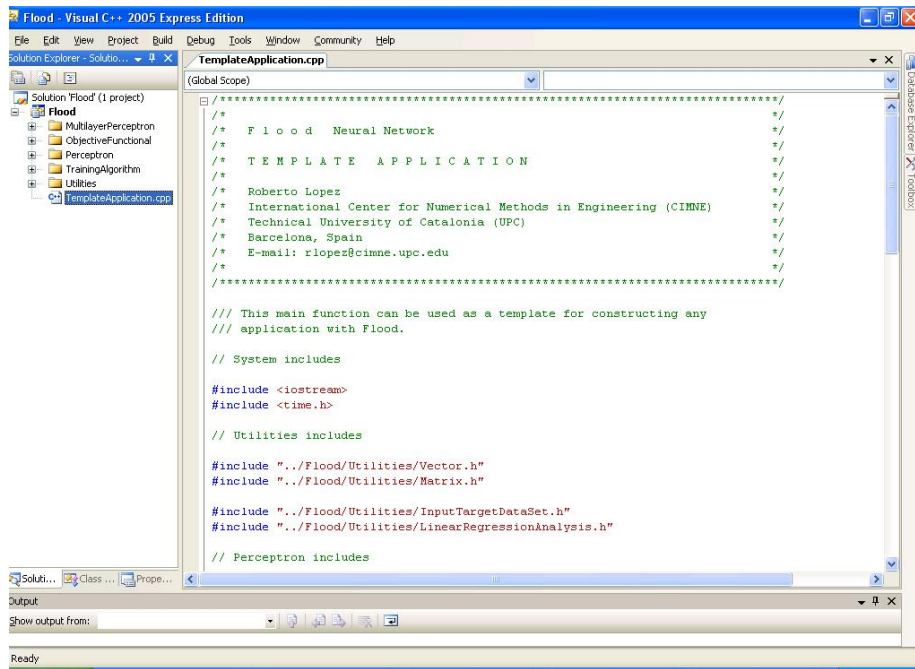


Figure 1: Microsoft Visual C++ 2008 project view.

2.2 Bloodshed Dev-C++

Bloodshed Dev-C++ is an Integrated Development Environment (IDE) for the C/C++ programming language. It uses Mingw port of GCC (GNU Compiler Collection) as its compiler, but it can also be used in combination with Cygwin or any other GCC based compiler. Bloodshed Dev-C++ is free software (under the GNU General Public License) and it can be downloaded at

<http://www.bloodshed.net/devcpp.html>.

To open the Flood project with Dev-C++ double click on the Flood.dev file in the DevCppProject folder. The window shown in Figure 2 should appear.

Use the Execute menu to compile, build and run the TemplateApplication.cpp, which is contained in the Applications folder.

Note that when executing an application with this compiler an MS-DOS console window will appear and then disappear. In order to avoid that you can write:

```
system(" pause ");
```

at any point in the main function where the program is desired to be paused.

A smarter way of executing a program built by Dev-C++ is to open a MS-DOS console, move to the DevCppProject and type:

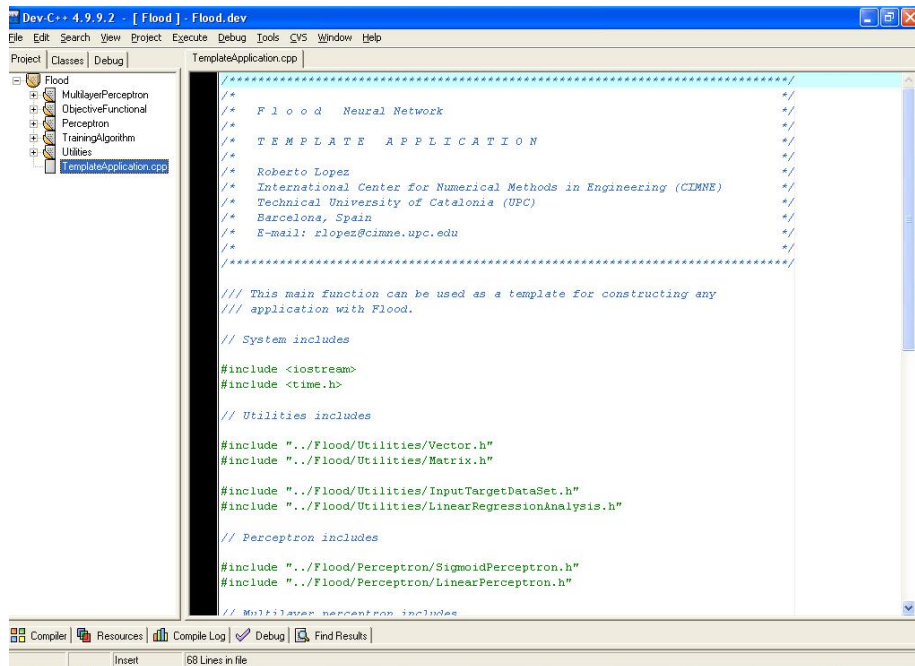


Figure 2: Bloodshed Dev-C++ 4.9 project view.

>Flood.exe

Then, the following message should appear:

Flood Neural Network. Template Application.

A complete set of application examples (main functions) can be found in the Applications folder. In order to run one of them just replace the TemplateApplication.cpp virtual file in the tree view of the Dev-C++ project window.

2.3 Other Compilers

If neither Dev-C++ nor Visual C++ is to be used, a new Project must be created for the new compiler in order to use Flood.

In general, the following steps shall be followed:

1. Select New Project from the File menu.
2. Create a C++ Empty Project and name it Flood2.
3. Save the project file into a folder.
4. Create a Tree View as that depicted in Figure 3.
5. Add the respective files to each folder from the Flood superfolder.

6. Compile, build and run the application.

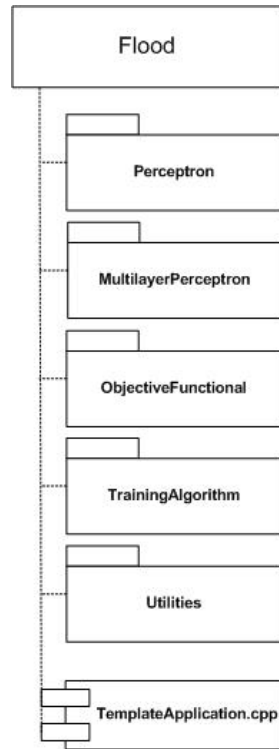


Figure 3: General project tree view.