Subject: Added Eigen
Posted by koldo on Tue, 07 Jun 2011 05:37:25 GMT
View Forum Message <> Reply to Message

Hello all

Eigen library has been included to Bazaar. See here for details.

It includes matrix algebra and math algorithms.

In addition to the Eigen package, there is an Eigen_demo package with many demos from very simple ones to non-linear equations systems solving and optimization.

U++ Bazaar Eigen packages have been cooked by Honza (dolik.rce) and koldo (me).

Subject: Re: Added Eigen
Posted by mdelfede on Thu, 09 Jun 2011 07:47:11 GMT
View Forum Message <> Reply to Message

Thank you

That one could be really useful on next versions of my app

Ciao

Max

Subject: Re: Added Eigen
Posted by koldo on Thu, 09 Jun 2011 10:34:00 GMT
View Forum Message <> Reply to Message

Excellent.

If somebody needs algorithms not included, s/he can ask for them.

I would probably need an ODE solver and a DAE solver.

Subject: Re: Added Eigen
Posted by forlano on Fri, 10 Aug 2012 06:42:56 GMT
View Forum Message <> Reply to Message

koldo wrote on Tue, 07 June 2011 07:37Hello all

Eigen library has been included to Bazaar. See here for details.

It includes matrix algebra and math algorithms.

In addition to the Eigen package, there is an Eigen_demo package with many demos from very simple ones to non-linear equations systems solving and optimization.

U++ Bazaar Eigen packages have been cooked by Honza (dolik.rce) and koldo (me).

Hello,

in the last week I needed a robust non linear fitting algorithm. I looked for it on the net and downloaded several. Unfortunately they do not compile on my windows machine or needed other libraries.
I was giving up when I realized it was already in my computer since one year... and the best one!
Thanks for providing this excellent package.

I have modified it for my need and of course it work as expected. Here my demo for a logistic fit:

```
#include <Core/Core.h>

using namespace Upp;

#include <plugin/Eigen/Eigen.h>
#include <plugin/Eigen/unsupported/Eigen/NonLinearOptimization>

using namespace Eigen;

// Generic functor
template<typename _Scalar, int nx = Dynamic, int ny = Dynamic>
struct Functor {
 typedef _Scalar Scalar;
 enum {
  InputsAtCompileTime = nx,
  ValuesAtCompileTime = ny
 };
 typedef Matrix<Scalar,InputsAtCompileTime,1> InputType;
 typedef Matrix<Scalar,ValuesAtCompileTime,1> ValueType;
 typedef Matrix<Scalar,ValuesAtCompileTime,InputsAtCompileTime> JacobianType;

 const int m_inputs, m_values;

 Functor() : m_inputs(InputsAtCompileTime), m_values(ValuesAtCompileTime) {}
 Functor(int inputs, int values) : m_inputs(inputs), m_values(values) {}

 int inputs() const {return m_inputs;}
 int values() const {return m_values;}
```

```
 // you should define that in the subclass :
 virtual  void operator() (const InputType& x, ValueType* v, JacobianType* _j=0) const {};
};

struct LogisticA_functor : Functor<double> {
 LogisticA_functor() : Functor<double>(3,15) {}
 static const double x[15];
 static const double y[15];
 int operator()(const VectorXd &b, VectorXd &fvec) const {
  ASSERT(b.size()==3);
  ASSERT(fvec.size()==15);
  for(int i=0; i<15; i++)
   fvec[i] = b[0] / (1.0 + b[1]*exp(-1.0 * b[2] * x[i])) - y[i];
  return 0;
 }
};
const double LogisticA_functor::x[15] = {-280.0, -240.0, -200.0, -160.0, -120.0, -80.0, -40.0, 0.0,
40.0, 80.0, 120.0, 160.0, 200.0, 240.0, 280.0};
const double LogisticA_functor::y[15] = {0.061276, 0.071429, 0.091574, 0.112821, 0.132959,
0.131597, 0.167887, 0.198380, 0.221380, 0.292292, 0.351831, 0.445803, 0.497754 ,
0.609337,0.632353};


void NonLinearOptimization() {

  VectorXd x(3);
  x << 5., 5., 0.01;  // Initial values

//first run
  LogisticA_functor functor;
  NumericalDiff<LogisticA_functor> numDiff(functor);
  LevenbergMarquardt<NumericalDiff<LogisticA_functor> > lm(numDiff);
  int ret = lm.minimize(x);
  if (ret == LevenbergMarquardtSpace::ImproperInputParameters ||
   ret == LevenbergMarquardtSpace::TooManyFunctionEvaluation)
   Cout() << "\nNo convergence!: " << ret;
  else {
      for (int i=0; i<3; i++) Cout() << "Parameter: "<< i << " = " << x[i] << "\n";
  }

//second run with new data of different length and same curve to fit
//
//  x[13] = {-280.0, -240.0, -200.0, -160.0, -120.0, -80.0, -40.0, 0.0, 40.0, 80.0, 120.0, 160.0,
200.0};
//  y[13] = {0.061276, 0.071429, 0.091574, 0.112821, 0.132959, 0.131597, 0.167887, 0.198380,
0.221380, 0.292292, 0.351831, 0.445803, 0.497754};
//      ..... ??? .....
```

}

CONSOLE_APP_MAIN
{ NonLinearOptimization();
}

Now, and here comes the problems, I would like to use the same curve, with the same number of parameters, but with a NEW dataset [X,Y] of different size.
I can duplicate the code (new functor and new drive) and it should work. But I need to do it up to 16 different dataset and my way is very, very silly.

It should be a way to modify the template of the functor to permit to feed at request a data set [X,Y] of N values.
Unfortunately the template structure and the operator() scary me and I do not know where to put my hands.

Can I ask a more easy way to drive the same functor with different dataset leaving unchanged the fitting curve?

Thanks a lot for your patience.
Luigi

Subject: Re: Added Eigen
Posted by Sender Ghost on Fri, 10 Aug 2012 10:45:43 GMT
View Forum Message <> Reply to Message

Hello, Luigi.
forlano wrote on Fri, 10 August 2012 08:42Now, and here comes the problems, I would like to use the same curve, with the same number of parameters, but with a NEW dataset [X,Y] of different size.
Following is a possible implementation:
Toggle Spoiler

#include <Core/Core.h>

using namespace Upp;

#include <plugin/Eigen/Eigen.h>
#include <plugin/Eigen/unsupported/Eigen/NonLinearOptimization>

using namespace Eigen;

// Generic functor
template<typename _Scalar, int nx = Dynamic, int ny = Dynamic>

```
struct Functor {
 typedef _Scalar Scalar;
 enum {
  InputsAtCompileTime = nx,
  ValuesAtCompileTime = ny
 };
 typedef Matrix<Scalar,InputsAtCompileTime,1> InputType;
 typedef Matrix<Scalar,ValuesAtCompileTime,1> ValueType;
 typedef Matrix<Scalar,ValuesAtCompileTime,InputsAtCompileTime> JacobianType;

 int m_inputs, m_values;
 void SetInputsCount(int count) { m_inputs = count; }
 void SetValuesCount(int count) { m_values = count; }

 Functor() : m_inputs(InputsAtCompileTime), m_values(ValuesAtCompileTime) {}
 Functor(int inputs, int values) : m_inputs(inputs), m_values(values) {}

 int inputs() const {return m_inputs;}
 int values() const {return m_values;}

 // you should define that in the subclass :
 virtual  void operator() (const InputType& x, ValueType* v, JacobianType* _j=0) const {};
};

class LogisticA_functor : public Functor<double> {
protected:
 double *vx;
 double *vy;
public:
 void Set(double *x, double *y) { vx = x; vy = y; }

 int operator()(const VectorXd &b, VectorXd &fvec) const {
  ASSERT(b.size()==m_inputs);
  ASSERT(fvec.size()==m_values);
  for(int i=0; i<m_values; i++)
   fvec[i] = b[0] / (1.0 + b[1]*exp(-1.0 * b[2] * vx[i])) - vy[i];
  return 0;
 }
};

void DoLevenbergMarquardt(LogisticA_functor& functor, VectorXd& x, double *j, double *k, int
inputs, int values)
{
 functor.Set(j, k);
 functor.SetInputsCount(inputs);
 functor.SetValuesCount(values);
 NumericalDiff<LogisticA_functor> numDiff(functor);
 LevenbergMarquardt<NumericalDiff<LogisticA_functor> > lm(numDiff);
```

```
 int ret = lm.minimize(x);
 if (ret == LevenbergMarquardtSpace::ImproperInputParameters ||
  ret == LevenbergMarquardtSpace::TooManyFunctionEvaluation)
  Cout() << "\nNo convergence!: " << ret << '\n';
 else
  for (int i = 0; i < inputs; ++i)
   Cout() << "Parameter: " << i << " = " << x[i] << '\n';
}

void NonLinearOptimization() {
 const int inputs = 3;
 VectorXd x(inputs);
 x << 5., 5., 0.01; // Initial values

 LogisticA_functor functor;
 Cout() << "First run\n";
 double jx[13] = {-280.0, -240.0, -200.0, -160.0, -120.0, -80.0, -40.0, 0.0, 40.0, 80.0, 120.0, 160.0,
200.0},
  jy[13] = {0.061276, 0.071429, 0.091574, 0.112821, 0.132959, 0.131597, 0.167887, 0.198380,
0.221380, 0.292292, 0.351831, 0.445803, 0.497754};
 VectorXd j = x;
 DoLevenbergMarquardt(functor, j, jx, jy, inputs, 13);

 Cout() << "Second run with new data of different length and same curve to fit\n";
 double kx[15] = {-280.0, -240.0, -200.0, -160.0, -120.0, -80.0, -40.0, 0.0, 40.0, 80.0, 120.0, 160.0,
200.0, 240.0, 280.0},
  ky[15] = {0.061276, 0.071429, 0.091574, 0.112821, 0.132959, 0.131597, 0.167887, 0.198380,
0.221380, 0.292292, 0.351831, 0.445803, 0.497754 ,  0.609337,0.632353};
 VectorXd k = x;
 DoLevenbergMarquardt(functor, k, kx, ky, inputs, 15);
}

CONSOLE_APP_MAIN
{
 NonLinearOptimization();
}
```

Subject: Re: Added Eigen
Posted by forlano on Fri, 10 Aug 2012 11:10:35 GMT
View Forum Message <> Reply to Message

Thank you Sender,

that is simply perfect! Now I can use a loop to fit everything.

One day I should learn how template and operator works...

Luigi

Subject: Re: Added Eigen
Posted by forlano on Fri, 10 Aug 2012 12:48:43 GMT
View Forum Message <> Reply to Message

Just one more little issue.

If I compile eigen_demo within the bazaar folder everything is ok.
But my program is in MyApps. When I try to add the eigen package (it is in bazaar/plugin) Theide can't see it and I'am not able to compile my program. Theide miss all the content of bazaar/plugin. Instead it see the content of /uppsrc/plugin

How to let theide to see c:\upp\bazaar\plugin\eigen package?

thanks,
luigi

Subject: Re: Added Eigen
Posted by mdelfede on Fri, 10 Aug 2012 13:00:44 GMT
View Forum Message <> Reply to Message

Simply add Bazaar to MyApps nest.

Ciao

Max

Subject: Re: Added Eigen
Posted by forlano on Fri, 10 Aug 2012 13:48:16 GMT
View Forum Message <> Reply to Message

mdelfede wrote on Fri, 10 August 2012 15:00Simply add Bazaar to MyApps nest.

Ciao

Max

Ops... it works, thanks

Let me translate for some other unaware as me.
The MyApps.var file in the upp folder must have the bazaar path:

```
UPP = "C:\\MyApps;C:\\upp\\uppsrc;C:\\upp\\bazaar";
COMMON = "C:\\upp\\common";
OUTPUT = "C:\\upp\\out";
```

Luigi

Subject: Re: Added Eigen
Posted by koldo on Sat, 11 Aug 2012 13:47:52 GMT
View Forum Message <> Reply to Message

Hello Luigi

How is that?. You did not have Bazaar included in MyApps

Subject: Re: Added Eigen
Posted by forlano on Sat, 11 Aug 2012 16:11:24 GMT
View Forum Message <> Reply to Message

koldo wrote on Sat, 11 August 2012 15:47Hello Luigi

How is that?. You did not have Bazaar included in MyApps

Hello Koldo,

now I have. Usually I got from bazaar the packages I was interested in and moved in MyApps. Because I use them in production application I was afraid of change that can broke my program.

Regards,
Luigi
