1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

http://codebase.mql4.com/4332

Regression Channel

Linear Regression Channel consists of two parallel lines, equidistant up and down from the line of linear regression trend. The distance between frame of the channel and regression line equals to the value of maximum close price deviation from the regression line.

Linear Regression Channel

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

Second-power (Parabolic) Polynomial Regression Channel

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

10 comments To post a new comment, please **log in** or **register**

i-Regr can be modified to apply to a Momentum instead of closing prices ? Merci

16.08.2011 14:21 kamel99

Hi, thanks for the EA but I can not run it, install it correctly but no operations. Can you help me? thanks

02.08.2011 07:28 luisg

Great expert, but hard to optimise, has anyone got good optimised inputs ?

30.08.2010 20:25 rjpickup

onewithzachy wrote:

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

I have to admit that it looks good. Even when compare iRegr polynomial degree 1 to "fixed" Linear Regression (with " Y = intercept + Slope * X " formula), the shape of Polynomial iRegr is much better. But there's a problem...

The heavy Gauss-matrix calculations is taking too much memory and processor resources, especially when using it as icustom. I end up having MT4 not responding and twice error not enough memory. I tried to reduce the calculation, by having it to calculate several calculation *only once*, like calculation for sum of square X (which is very easy) or the Gauss-matrix calculation (then we move the matrix result calculation to the second matrix for further calculation, like calculation for polynomial coefficient) and even tried to use different ways for matrix calculation either using Gauss or Gauss-Jordan elimination, or using elementary row calculation to back substitution, but I end up having wrong result value.

My current (and hopefully the only) problem that I can't figure is the Gauss elimination on sum of XY.

I hope its just me the fool one, but is there anyone that can solve this problem, so that the iRegr or another Polynomial Regression doesn't have to calculate on every tick.

The second (little kind a funny) problem is... I rewrite the iRegr so it only draw the first line of the center line and its deviation line only, so I can see the previous lines, but when compare it to LWMA especially on smaller period (<5) the center line looks just almost the same line with LWMA.

Would the following change ensure that the work is done only on the open of each bar?

```
int start()
{
  if (Bars < bars) return;
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
  if (LastSignal == Time[0]) return;
//----
  LastSignal=Time[0];
```

30.05.2010 01:56 engcomp

Could you pls add a condition such that (True / False) which selects if the EA would initiate trade only in the direction of the trean
Hence

if the Channel is sloping up, only Longs would be initiated and when the proce hits lower band
If the Channel is sloping Down, only Shorts would be initiated when the price hits upper band

09.03.2010 18:16 MANUBAJWA3000

I downloaded the EA 2 days ago and installed it (EA) correctly, but it is not trading on GBP/USD-M5. Settings are at default. Any suggestions, please?

Thanking you in advance.

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

27.01.2010 16:33 Hercs78

I have to admit that it looks good. Even when compare iRegr polynomial degree 1 to "fixed" Linear Regression (with " Y = intercept + Slope * X " formula), the shape of Polynomial iRegr is much better. But there's a problem...

The heavy Gauss-matrix calculations is taking too much memory and processor resources, especially when using it as icustom. I end up having MT4 not responding and twice error not enough memory. I tried to reduce the calculation, by having it to calculate several calculation *only once*, like calculation for sum of square X (which is very easy) or the Gauss-matrix calculation (then we move the matrix result calculation to the second matrix for further calculation, like calculation for polynomial coefficient) and even tried to use different ways for matrix calculation either using Gauss or Gauss-Jordan elimination, or using elementary row calculation to back substitution, but I end up having wrong result value.

My current (and hopefully the only) problem that I can't figure is the Gauss elimination on sum of XY.

I hope its just me the fool one, but is there anyone that can solve this problem, so that the iRegr or another Polynomial Regression doesn't have to calculate on every tick.

The second (little kind a funny) problem is... I rewrite the iRegr so it only draw the first line of the center line and its deviation line only, so I can see the previous lines, but when compare it to LWMA especially on smaller period (<5) the center line looks just almost the same line with LWMA.

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

10.01.2010 12:06 onewithzachy

i just wanna know how i can put the RSI, CCI, momentum on this e regr

17.03.2009 22:39 torito

Hi,

what means:

trailingOn

trailingStart

trailingSize

Thanks fot the answer. befor

Meta1

18.09.2008 19:20 Meta1

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

How do Alert BUY or SELL

13.09.2008 01:52 Mehmet

[[[[[[[[[[[[[[[[[[[[[[[[

polynomial regression channel

I was wondering if sc had an indicator called polynomial regression channel which is somewhat different from the linear regression channel you do have. If not would you be willing to add it to your indicator list. More info here:http://codebase.mql4.com/4332. This was taken from ninjatrader (sorry to mention a competitor).:eek:

Re: polynomial regression channel

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

Quote:

> Originally Posted by **rburns**
> *Well then if we make one that works properly look at all the new users Sierra will have.*

here is ninja code written by a russian if you have a programming background you will see that it recalculates, redraws up to 10 bar back.

http://www.sierrachart.com/supportboard/showthread.php?t=27303

http://www.sierrachart.com/supportboard/showthread.php?t=27303

```
#region Using declarations
using System;
using System.ComponentModel;
using System.Diagnostics;
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
using System.Drawing;
using System.Drawing.Drawing2D;
using System.Xml.Serialization;
using NinjaTrader.Cbi;
using NinjaTrader.Data;
using NinjaTrader.Gui.Chart;
#endregion

// This namespace holds all indicators and is required. Do not change it.
namespace NinjaTrader.Indicator
{
    /// <summary>
    /// Polnomial Regression Channel
    /// </summary>
    [Description("Polynomial Regression Channel")]
        //port of indicator from MQL4 to NinjaTrader.  Original author:
fxcanada
        // http://codebase.mql4.com/4332
        [Gui.Design.DisplayName(" PRC")]
    public class PRC : Indicator
    {
        #region Variables
        // Wizard generated variables
            private int degree = 3; // Default setting for Degree
            private int period = 250; // Default setting for Period 250
            private double strdDev = 1.62; // Default setting for StrdDev 2
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
                        private double strdDev2 = 2.000; // Default setting for
StrdDev 2
        // User defined variables (add any user defined variables below)
                        private double[,] ai = new double[10,10];
                        private double[] b = new double[10];
                        private double[] x = new double[10];
                        private double[] sx = new double[10];
                        private double sum;
                        private int ip;
                        private int p;
                        private int n;
                        private int f;
                        private double qq;
                        private double mm;
                        private double tt;
                        private int ii;
                        private int jj;
                        private int kk;
                        private int ll;
                        private int nn;
                        private double sq;
                        private double sq2;
                        private int i0 = 0;
                        private int mi;
                private int sounds = 0;
                private bool Standard = true;
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
                private bool fillZones = false;
                private bool showLine = false;
                private Color fillColor = Color.Yellow;
        #endregion

        /// <summary>
        /// This method is used to configure the indicator and is called once
before any bar data is loaded.
        /// </summary>
        protected override void Initialize()
        {
            Add(new Plot(new Pen(Color.DimGray, 2), PlotStyle.Line,
"Fx"));//Plot0
            Add(new Plot(Color.FromKnownColor(KnownColor.Red),
PlotStyle.Line, "Sqh"));//Plot1
            Add(new Plot(Color.FromKnownColor(KnownColor.Blue),
PlotStyle.Line, "Sql"));//Plot2
                        Add(new Plot(Color.FromKnownColor(KnownColor.Red),
PlotStyle.Line, "Sqh2"));//Plot3
            Add(new Plot(Color.FromKnownColor(KnownColor.Blue),
PlotStyle.Line, "Sql2"));//Plot4
                        Plots[0].Pen.DashStyle = DashStyle.DashDot;

                        Add(new Plot(new Pen(Color.Orange, 2), PlotStyle.Line,
"TEMA"));//Plot5
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
                        Add(new Plot(new Pen(Color.Peru, 2), PlotStyle.Line,
"HATEMA"));//Plot6

            CalculateOnBarClose       = false;
            Overlay                           = true;
            PriceTypeSupported = true;
        }

        /// <summary>
        /// Called on each bar update event (incoming tick)
        /// </summary>
        protected override void OnBarUpdate()
        {
                        if( CurrentBar < period) return;

                        ip = period;
                        p = ip;
                        sx[1] = p + 1;
                        nn = degree + 1;
                        //----------------------sx----------------------------
-------------------------------------
                        for(mi=1;mi<=nn*2-2;mi++) // математическое выражение -
для всех mi от 1 до nn*2-2
                        {
                                sum=0;
                                for(n=i0;n<=i0+p;n++)
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
                                {
                                sum+=Math.Pow(n,mi);
                                }
                                sx[mi+1]=sum;
                        }
                        //----------------------syx-----------
                        for(mi=1;mi<=nn;mi++)
                        {
                                sum=0.00000;
                                for(n=i0;n<=i0+p;n++)
                                {
                                if(mi==1) sum+=Close[n];
                                else sum+=Close[n]*Math.Pow(n,mi-1);
                                }
                                b[mi]=sum;
                        }

        //===============Matrix=================================================
=======================================================
                        for(jj=1;jj<=nn;jj++)
                        {
                                for(ii=1; ii<=nn; ii++)
                                {
                                kk=ii+jj-1;
                                ai[ii,jj]=sx[kk];
                                }
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
                        }

        //===============Gauss=====================================================
==========================================================
                        for(kk=1; kk<=nn-1; kk++)
                        {
                                ll=0;
                                mm=0;
                                for(ii=kk; ii<=nn; ii++)
                                {
                                if(Math.Abs(ai[ii,kk])>mm)
                                {
                                        mm=Math.Abs(ai[ii,kk]);
                                        ll=ii;
                                }
                                }
                                if(ll==0) return;
                                if (ll!=kk)
                                {
                                for(jj=1; jj<=nn; jj++)
                                {
                                        tt=ai[kk,jj];
                                        ai[kk,jj]=ai[ll,jj];
                                        ai[ll,jj]=tt;
                                }
                                tt=b[kk];
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
                        b[kk]=b[ll];
                        b[ll]=tt;
                        }
                        for(ii=kk+1;ii<=nn;ii++)
                        {
                        qq=ai[ii,kk]/ai[kk,kk];
                        for(jj=1;jj<=nn;jj++)
                        {
                                if(jj==kk) ai[ii,jj]=0;
                                else ai[ii,jj]=ai[ii,jj]-qq*ai[kk,jj];
                        }
                        b[ii]=b[ii]-qq*b[kk];
                        }
                }
                x[nn]=b[nn]/ai[nn,nn];
                for(ii=nn-1;ii>=1;ii--)
                {
                        tt=0;
                        for(jj=1;jj<=nn-ii;jj++)
                        {
                        tt=tt+ai[ii,ii+jj]*x[ii+jj];
                        x[ii]=(1/ai[ii,ii])*(b[ii]-tt);
                        }
                }
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
        //=================================================================
==================================================
                        for(n=i0;n<=i0+p;n++)
                        {
                                sum=0;
                                for(kk=1;kk<=degree;kk++)
                                {
                                sum+=x[kk+1]*Math.Pow(n,kk);
                                }
                                Fx.Set(n,x[1]+sum);
                        }
                        //-----------------------------------Std---------------
-------------------------------------------------------------
                        sq=0.0;
                        for(n=i0;n<=i0+p;n++)
                        {
                                sq+=Math.Pow(Close[n]-Fx[n],2);
                        }
                        sq=Math.Sqrt(sq/(p+1))*strdDev;

                        for(n=i0;n<=i0+p;n++)
                        {
                                Sqh.Set(n,Fx[n]+sq);
                                Sql.Set(n,Fx[n]-sq);
                        }
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
                        if(showLine)
                        {
                        sq2=0.0;
                        for(n=i0;n<=i0+p;n++)
                        {
                                sq2+=Math.Pow(Close[n]-Fx[n],2);
                        }
                        sq2=Math.Sqrt(sq2/(p+1))*strdDev2;

                        for(n=i0;n<=i0+p;n++)
                        {
                                Sqh2.Set(n,Fx[n]+sq2);
                                Sql2.Set(n,Fx[n]-sq2);
                        }
                        }

                        //----------Fill Zones
                        if(fillZones && showLine)
                        {
                        DrawRegion("HiZone", CurrentBar, 0, Sqh,  Sqh2,
Color.Empty, fillColor, 2);
                        DrawRegion("LoZone", CurrentBar, 0, Sql,  Sql2,
Color.Empty, fillColor, 2);
                        }
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
                        //----------Alerts
                        if(CrossAbove(Close, Sqh, 1) && sounds > 0 ||
CrossBelow(Close, Sql, 1) && sounds > 0)
                        {if(Standard)
                        {if(sounds == 1)

        {Alert("Ready",NinjaTrader.Cbi.Priority.High,"Ready","c:\\windows\\med
ia\\chimes.wav",4,Color.Black,Color.Yellow);}
                        if(sounds == 2)

        {Alert("Ready",NinjaTrader.Cbi.Priority.High,"Ready","c:\\windows\\med
ia\\ding.wav",4,Color.Black,Color.Yellow);}
                        if(sounds == 3)

        {Alert("Ready",NinjaTrader.Cbi.Priority.High,"Ready","alert4.wav",4,Co
lor.Black,Color.Yellow);}}
                        else
                        {if(sounds == 1)
                                {PlaySound(@"c:\\windows\\media\\chimes.wav");}
                        if(sounds == 2)
                                {PlaySound(@"c:\\windows\\media\\ding.wav");}
                        if(sounds == 3)
                                {PlaySound(@"\\alert4.wav");}}
                        }
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
        }

        #region Properties
        [Browsable(false)]     // this line prevents the data series from
being displayed in the indicator properties dialog, do not remove
        [XmlIgnore()]          // this line ensures that the indicator can be
saved/recovered as part of a chart template, do not remove
        public DataSeries Fx
        {
            get { return Values[0]; }
        }

        [Browsable(false)]     // this line prevents the data series from
being displayed in the indicator properties dialog, do not remove
        [XmlIgnore()]          // this line ensures that the indicator can be
saved/recovered as part of a chart template, do not remove
        public DataSeries Sqh
        {
            get { return Values[1]; }
        }

        [Browsable(false)]     // this line prevents the data series from
being displayed in the indicator properties dialog, do not remove
        [XmlIgnore()]          // this line ensures that the indicator can be
saved/recovered as part of a chart template, do not remove
        public DataSeries Sql
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
        {
            get { return Values[2]; }
        }

                [Browsable(false)]     // this line prevents the data series
from being displayed in the indicator properties dialog, do not remove
        [XmlIgnore()]          // this line ensures that the indicator can be
saved/recovered as part of a chart template, do not remove
        public DataSeries Sqh2
        {
            get { return Values[3]; }
        }

        [Browsable(false)]     // this line prevents the data series from
being displayed in the indicator properties dialog, do not remove
        [XmlIgnore()]          // this line ensures that the indicator can be
saved/recovered as part of a chart template, do not remove
        public DataSeries Sql2
        {
            get { return Values[4]; }
        }

        [Description("type of polynomial 1, 2, 3 or 4")]
        [Category("Parameters")]
        public int Degree
        {
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
            get { return degree; }
            set { degree = Math.Min(4,Math.Max(value,1)); }
        }

        [Description("nmbr of bars to use in calculation")]
        [Category("Parameters")]
        public int Period
        {
            get { return period; }
            set { period = Math.Max(1, value); }
        }

        [Description("Standard Deviations")]
        [Category("Parameters")]
        public double StrdDev
        {
            get { return strdDev; }
            set { strdDev = Math.Max(0.5, value); }
        }

               [Description("Standard Deviations")]
        [Category("Parameters")]
        public double StrdDev2
        {
            get { return strdDev2; }
            set { strdDev2 = Math.Max(0.5, value); }
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
        }


                [Description("0 = no sound, 1 = Chimes, 2 = Ding, 3 =
Alert4(NT)")]
                [Gui.Design.DisplayName ("\t\tSound Selection")]
        [Category("Options")]
        public int Sounds
        {
            get { return sounds; }
            set { sounds = Math.Min(3,Math.Max(value,0)); }
        }

                [Description("Standard alert (true) or constant sound barrage
(false)")]
                [Gui.Design.DisplayName ("\t\tStandard Alert?")]
        [Category("Options")]
        public bool standard
        {
            get { return Standard; }
            set { Standard = value; }
        }

                [Description("Fill the zones created by the two Sqh and Sql
lines")]
                [Gui.Design.DisplayName ("Fill OB/OS zones?")]
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
        [Category("Options")]
        public bool FillZones
        {
            get { return fillZones; }
            set { fillZones = value; }
        }

                [Description("This will plot the second set of lines which
then gives the option of filling the zone")]
                [Gui.Design.DisplayName ("\tPlot Sqh2 and Sql2 lines?")]
        [Category("Options")]
        public bool ShowLine
        {
            get { return showLine; }
            set { showLine = value; }
        }


                [Browsable(false)]
                public string fillColorSerialize
                {
                        get { return
NinjaTrader.Gui.Design.SerializableColor.ToString(fillColor); }
                        set { fillColor =
NinjaTrader.Gui.Design.SerializableColor.FromString(value); }
                }
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
                [XmlIgnore()]
                [Description("Fill zone color.")]
                [Category("Options")]
                [Gui.Design.DisplayNameAttribute("Fill zone color")]
                public Color FillColor
                {
                        get { return fillColor; }
                        set { fillColor = value; }
                }
        #endregion
    }
}

#region NinjaScript generated code. Neither change nor remove.
// This namespace holds all indicators and is required. Do not change it.
namespace NinjaTrader.Indicator
{
    public partial class Indicator : IndicatorBase
    {
        private PRC[] cachePRC = null;

        private static PRC checkPRC = new PRC();

        /// <summary>
        /// Polynomial Regression Channel
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
        /// </summary>
        /// <returns></returns>
        public PRC PRC(int degree, int period, double strdDev, double
strdDev2)
        {
            return PRC(Input, degree, period, strdDev, strdDev2);
        }

        /// <summary>
        /// Polynomial Regression Channel
        /// </summary>
        /// <returns></returns>
        public PRC PRC(Data.IDataSeries input, int degree, int period, double
strdDev, double strdDev2)
        {
            checkPRC.Degree = degree;
            degree = checkPRC.Degree;
            checkPRC.Period = period;
            period = checkPRC.Period;
            checkPRC.StrdDev = strdDev;
            strdDev = checkPRC.StrdDev;
            checkPRC.StrdDev2 = strdDev2;
            strdDev2 = checkPRC.StrdDev2;

            if (cachePRC != null)
                for (int idx = 0; idx < cachePRC.Length; idx++)
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
                    if (cachePRC[idx].Degree == degree &&
cachePRC[idx].Period == period && Math.Abs(cachePRC[idx].StrdDev - strdDev)
<= double.Epsilon && Math.Abs(cachePRC[idx].StrdDev2 - strdDev2) <=
double.Epsilon && cachePRC[idx].EqualsInput(input))
                        return cachePRC[idx];

            PRC indicator = new PRC();
            indicator.BarsRequired = BarsRequired;
            indicator.CalculateOnBarClose = CalculateOnBarClose;
            indicator.Input = input;
            indicator.Degree = degree;
            indicator.Period = period;
            indicator.StrdDev = strdDev;
            indicator.StrdDev2 = strdDev2;
            indicator.SetUp();

            PRC[] tmp = new PRC[cachePRC == null ? 1 : cachePRC.Length + 1];
            if (cachePRC != null)
                cachePRC.CopyTo(tmp, 0);
            tmp[tmp.Length - 1] = indicator;
            cachePRC = tmp;
            Indicators.Add(indicator);

            return indicator;
        }
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
    }
}

// This namespace holds all market analyzer column definitions and is
required. Do not change it.
namespace NinjaTrader.MarketAnalyzer
{
    public partial class Column : ColumnBase
    {
        /// <summary>
        /// Polynomial Regression Channel
        /// </summary>
        /// <returns></returns>
        [Gui.Design.WizardCondition("Indicator")]
        public Indicator.PRC PRC(int degree, int period, double strdDev,
double strdDev2)
        {
            return _indicator.PRC(Input, degree, period, strdDev, strdDev2);
        }

        /// <summary>
        /// Polynomial Regression Channel
        /// </summary>
        /// <returns></returns>
        public Indicator.PRC PRC(Data.IDataSeries input, int degree, int
period, double strdDev, double strdDev2)
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
        {
            return _indicator.PRC(input, degree, period, strdDev, strdDev2);
        }

    }
}

// This namespace holds all strategies and is required. Do not change it.
namespace NinjaTrader.Strategy
{
    public partial class Strategy : StrategyBase
    {
        /// <summary>
        /// Polynomial Regression Channel
        /// </summary>
        /// <returns></returns>
        [Gui.Design.WizardCondition("Indicator")]
        public Indicator.PRC PRC(int degree, int period, double strdDev,
double strdDev2)
        {
            return  indicator.PRC(Input, degree, period, strdDev, strdDev2);
        }

        /// <summary>
        /// Polynomial Regression Channel
        /// </summary>
```

1-23-2012_Mon_Second-power (Parabolic) Polynomial Regression Channel
C:\Users\a\Documents_Polynomial Regression Channel
(305)OnTrial666-7890
((((((((((((((((((((((((((
#property copyright "© 2008 BJF Trading Group" #p
http://codebase.mql4.com/source/10881
http://codebase.mql4.com/source/10881

```
#property copyright "© 2008 BJF Trading Group"
#property link      "www.iticsoftware.com"

#define major   1
#define minor   1

extern string _tmp1_ = " --- Trade params ---";
extern string TradeTime = "3:00-21:20";
extern double Lots = 0.1;
extern int StopLoss = 0;
extern int TakeProfit = 0;
extern int Slippage = 3;
extern int Magic = 20080829;

extern string _tmp2_ = " --- i-Regr ---";
extern int Regr.degree = 3;
extern double Regr.kstd = 1.0;
```

```
        /// <returns></returns>
        public Indicator.PRC PRC(Data.IDataSeries input, int degree, int
period, double strdDev, double strdDev2)
        {
            if (InInitialize && input == null)
                throw new ArgumentException("You only can access an indicator
with the default input/bar series from within the 'Initialize()' method");

            return _indicator.PRC(input, degree, period, strdDev, strdDev2);
        }

    }
}
```

#endregion

]]]]]]]]]]]]]]]]]]]]]]]]]]]