HW#13 최적화 실습

```
In[6]:= f[x_, y_] = Sin[x + y - 1] + (x - y - 1)^2 - 1.5 * x + 2.5 * y + 1
        f = Function[{x, y}, Sin[x + y - 1] + (x - y - 1)^2 - 1.5 * x + 2.5 * y + 1][x, y]
```

Out[6]= $1 - 1.5\,x + (-1 + x - y)^2 + 2.5\,y - \mathrm{Sin}[1 - x - y]$

Out[7]= $1 - 1.5\,x + (-1 + x - y)^2 + 2.5\,y - \mathrm{Sin}[1 - x - y]$

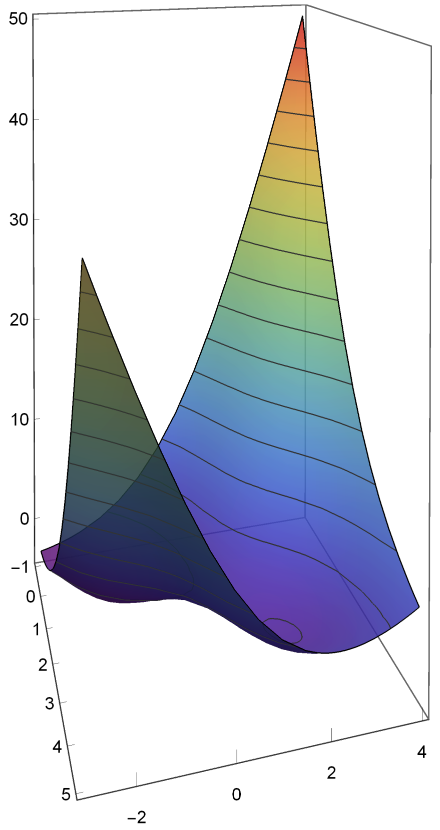
```
In[8]:= grad = Grad[f, {x, y}]
        df[x_, y_] = grad
```

Out[8]= {-1.5 + 2 (-1 + x - y) + Cos[1 - x - y], 2.5 - 2 (-1 + x - y) + Cos[1 - x - y]}

Out[9]= {-1.5 + 2 (-1 + x - y) + Cos[1 - x - y], 2.5 - 2 (-1 + x - y) + Cos[1 - x - y]}

```
In[10]:= Plot3D[f, {x, -1, 5}, {y, -3, 4}, PlotRange → All, ClippingStyle → None, AspectRatio → 2,
          PlotTheme → "Web", PlotStyle → Opacity[.85], ColorFunction → "Rainbow"]
```

Out[10]=




```
In[454]:= p = {RandomReal[{-1, 5}], RandomReal[{-3, 4}]}
```

Out[454]= {-0.788806, 1.12967}

Gradient Descent

In[455]:=
```
lamda = 0.1
x = p[[1]]
y = p[[2]]
pts = {};
For[i = 0, i < 20, i++, {x, y} = {x, y} - lamda * df[x, y]; pts = Append[pts, {x, y}];]
pts = Join[{{p[[1]], p[[2]]}}, pts];
pts // MatrixForm
ContourPlot[Sin[x + y - 1] + (x - y - 1)^2 - 1.5 * x + 2.5 * y + 1, {x, -3, 5}, {y, -3, 4},
 Contours → 10, ContourLabels → True, Epilog → {Red, Line[pts], Point[pts]}]
```

Out[455]= 0.1

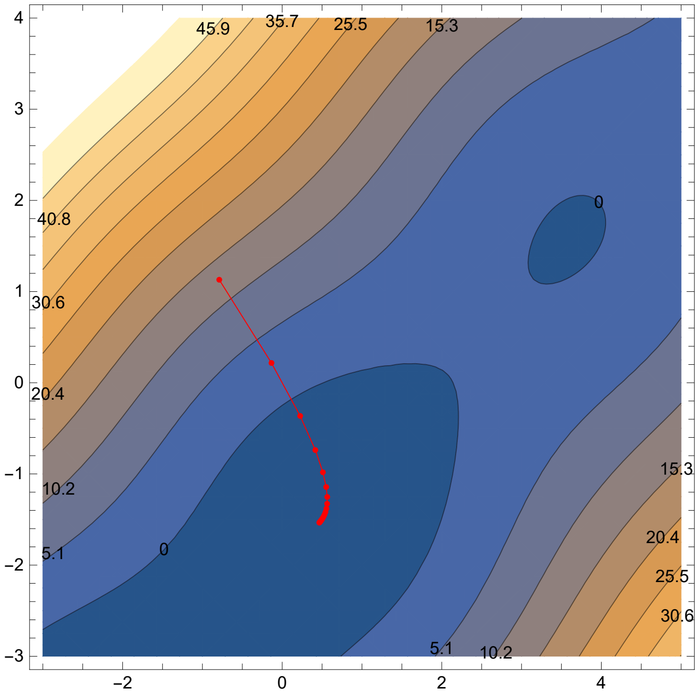
Out[456]= -0.788806

Out[457]= 1.12967

Out[461]//MatrixForm=

-0.788806	1.12967
-0.134163	0.21692
0.225252	-0.364098
0.415518	-0.738092
0.510228	-0.981938
0.551902	-1.1434
0.564912	-1.25227
0.563106	-1.3272
0.554254	-1.37993
0.54263	-1.41788
0.530505	-1.4458
0.519016	-1.46677
0.508669	-1.4828
0.499624	-1.49526
0.491865	-1.50506
0.48529	-1.51287
0.479765	-1.51913
0.475147	-1.52419
0.471302	-1.5283
0.468108	-1.53165
0.465461	-1.5344

Out[462]=




```
In[365]:= lamda = 0.01
x = p[[1]];
y = p[[2]];
pts = {};
For[i = 0, i < 20, i++, {x, y} = {x, y} - lamda * df[x, y]; pts = Append[pts, {x, y}];]
pts = Join[{{p[[1]], p[[2]]}}, pts];
pts // MatrixForm;
ContourPlot[Sin[x + y - 1] + (x - y - 1)^2 - 1.5 * x + 2.5 * y + 1, {x, -3, 5}, {y, -3, 4},
  Contours → 10, ContourLabels → True, Epilog → {Red, Line[pts], Point[pts]}];

Out[365]= 0.01
```

Newton's 방법

```
In[242]:= Clear[x]
Clear[y]
d2f[x_, y_] = Grad[df[x, y], {x, y}]

Out[244]= {{2 + Sin[1 - x - y], -2 + Sin[1 - x - y]}, {-2 + Sin[1 - x - y], 2 + Sin[1 - x - y]}}
```

In[463]:=
```
x = p[[1]]
y = p[[2]]
pts = {};
For[i = 0, i < 5, i++,
 {x, y} = {x, y} - Dot[df[x, y], Inverse[d2f[x, y]]]; pts = Append[pts, {x, y}];]
pts = Join[{{p[[1]], p[[2]]}}, pts];
pts // MatrixForm
ContourPlot[Sin[x + y - 1] + (x - y - 1)^2 - 1.5 * x + 2.5 * y + 1, {x, -3, 5}, {y, -3, 4},
 Contours → 10, ContourLabels → True, Epilog → {Red, Line[pts], Point[pts]}]
```

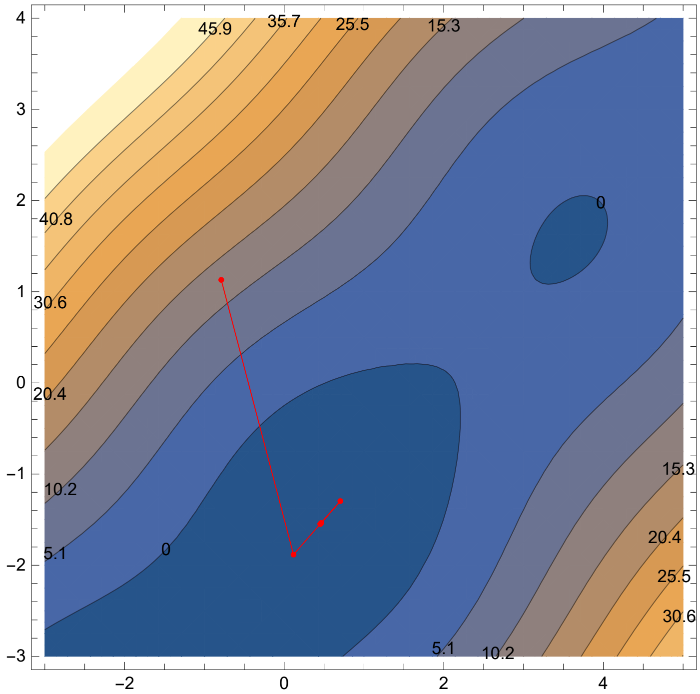
Out[463]= -0.788806

Out[464]= 1.12967

Out[468]//MatrixForm=

$$\begin{pmatrix} -0.788806 & 1.12967 \\ 0.116837 & -1.88316 \\ 0.703994 & -1.29601 \\ 0.464547 & -1.53545 \\ 0.452879 & -1.54712 \\ 0.452802 & -1.5472 \end{pmatrix}$$

Out[469]=



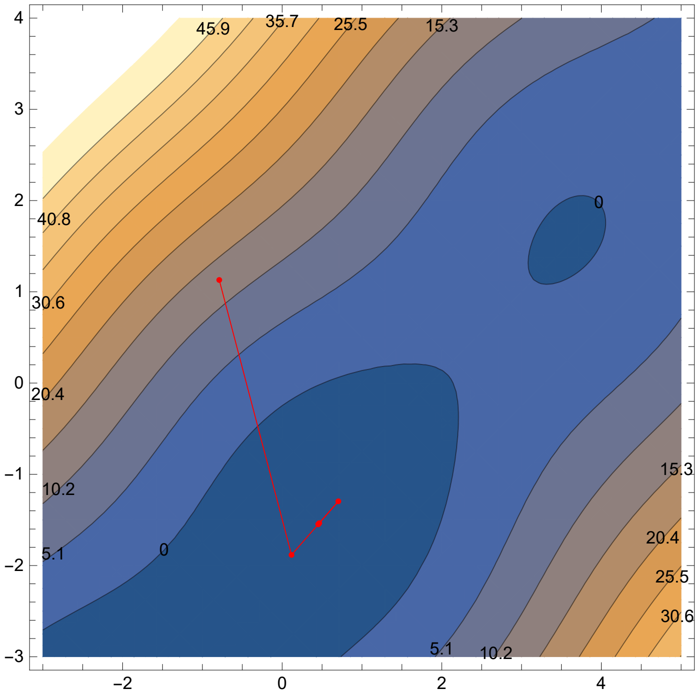
Newton + TrustRegion

In[486]:=
```
j = 1;
pts = Reap[FindMinimum[Sin[x + y - 1] + (x - y - 1)^2 - 1.5 * x + 2.5 * y + 1,
    {{x, p[[1]]}, {y, p[[2]]}}, Method → {"Newton", "StepControl" → "TrustRegion"},
    StepMonitor :→ Print["Step:", j++, " x, y =", Sow[{x, y}]]]][[2, 1]]
pts = Join[{{p[[1]], p[[2]]}}, pts];
ContourPlot[Sin[x + y - 1] + (x - y - 1)^2 - 1.5 * x + 2.5 * y + 1, {x, -3, 5}, {y, -3, 4},
 Contours → 10, ContourLabels → True, Epilog → {Red, Line[pts], Point[pts]}]
```

```
Step:1 x, y ={0.116837, -1.88316}

Step:2 x, y ={0.703994, -1.29601}

Step:3 x, y ={0.464547, -1.53545}

Step:4 x, y ={0.452879, -1.54712}

Step:5 x, y ={0.452802, -1.5472}
```

Out[487]=
```
{{0.116837, -1.88316}, {0.703994, -1.29601},
 {0.464547, -1.53545}, {0.452879, -1.54712}, {0.452802, -1.5472}}
```

Out[489]=



Newton + LineSearch

In[498]:=
```
j = 1;
pts = Reap[FindMinimum[Sin[x + y - 1] + (x - y - 1)^2 - 1.5 * x + 2.5 * y + 1,
    {{x, p[[1]]}, {y, p[[2]]}}, Method → {"Newton", "StepControl" → "LineSearch"},
    StepMonitor :→ Print["Step:", j++, " x, y =", Sow[{x, y}]]]][[2, 1]]
pts = Join[{{p[[1]], p[[2]]}}, pts];
ContourPlot[Sin[x + y - 1] + (x - y - 1)^2 - 1.5 * x + 2.5 * y + 1, {x, -3, 5}, {y, -3, 4},
 Contours → 10, ContourLabels → True, Epilog → {Red, Line[pts], Point[pts]}]
```

```
Step:1 x, y ={0.116837, -1.88316}

Step:2 x, y ={0.394795, -1.60521}

Step:3 x, y ={0.455173, -1.54483}

Step:4 x, y ={0.452806, -1.54719}

Step:5 x, y ={0.452802, -1.5472}
```

Out[499]=
```
{{0.116837, -1.88316}, {0.394795, -1.60521},
 {0.455173, -1.54483}, {0.452806, -1.54719}, {0.452802, -1.5472}}
```

Out[501]=