Problem Set 3.3

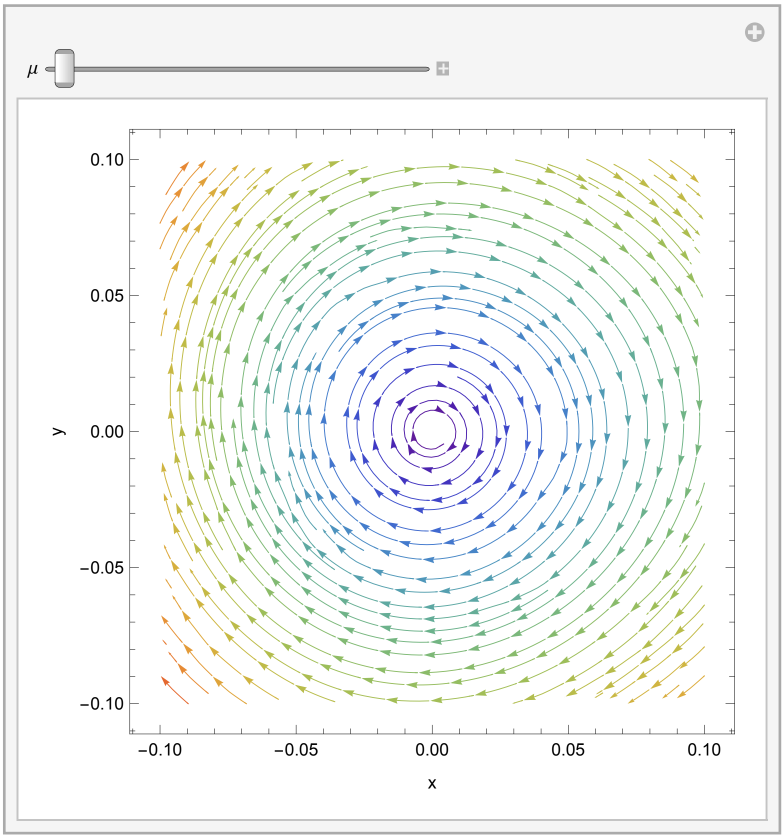
In[1]:=

```
f[x_, y_, μ_] := μ*x + y - x^2
g[x_, y_, μ_] := -x + μ*y + 2*x^2
```

In[3]:=

```
xrange = 0.1;
yrange = 0.1;
μrangemin = 0.05;
μrangemax = 0.08;

Manipulate[
    Show[
        StreamPlot[{f[x,y,μ], g[x,y,μ]}, {x, -xrange, xrange}, {y, -yrange, yrange},
        StreamStyle → Automatic,
        StreamColorFunction → "Rainbow",
        FrameLabel → {"x", "y"},
        StreamPoints → Fine,
        AspectRatio → 1]
    ],
{μ,μrangemin,μrangemax}]
```

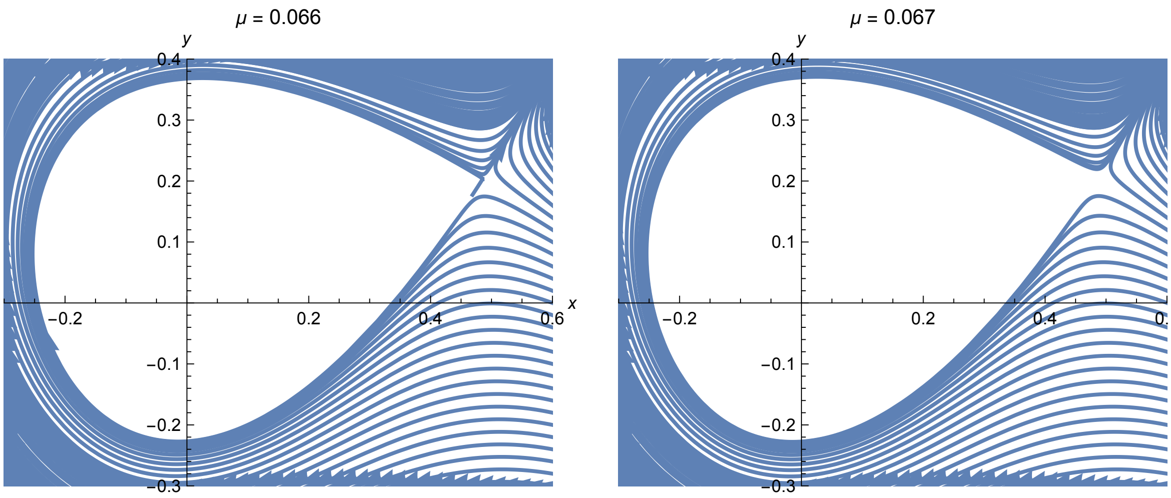
Out[7]=



In[8]:=
```
maxt = 20;
sol[x0_, y0_, μ_] := NDSolve[{x'[t] == μ*x[t] + y[t] - x[t]^2, y'[t] == -x[t] + μ*y[t] +
                                {x,y},
                                {t,0,maxt}]
minx=-0.3;
miny=-0.3;
maxx=0.6;
maxy=0.4;
step = 0.02;
initialC=Join[
            Table[{minx,y},{y,miny,maxy,step}],
            Table[{maxx,y},{y,miny,maxy,step}],
            Table[{x,miny},{x,minx,maxx,step}],
            Table[{x,maxy},{x,minx,maxx,step}]];



μ = 0.066;
p1=Show[
    Table[
        ParametricPlot[
            Evaluate[{x[t],y[t]}/. sol[initialC〚i,1〛, initialC〚i,2〛, μ]],
            {t,0,maxt},
            PlotRange→{{minx,maxx},{miny,maxy}},
            AxesLabel→{x,y}
        ]
        /. Line[x_]:→{Arrowheads[{{0.05, 0.5}, {0.05, 0.0}}],Arrow[x]},{i,1,Length[init
    ],
    PlotLabel → "μ = " <> ToString[μ]
];

μ = 0.067;
p2=Show[
    Table[
        ParametricPlot[
            Evaluate[{x[t],y[t]}/. sol[initialC〚i,1〛, initialC〚i,2〛, μ]],
            {t,0,maxt},
            PlotRange→{{minx,maxx},{miny,maxy}},
            AxesLabel→{x,y}
        ]
        /. Line[x_]:→{Arrowheads[{{0.05, 0.5}, {0.05, 0.0}}],Arrow[x]},{i,1,Length[init
    ],
    PlotLabel → "μ = " <> ToString[μ]
];
```

In[20]:=
```
GraphicsRow[{p1,p2}]
```

Out[20]=



In[21]:=
```
fixedPoints[μ_] := NSolve[{f[x, y, μ] == 0, g[x, y, μ] == 0}, {x, y}]
sol[x0_, y0_, μ_] := NDSolve[{x'[t] == μ*x[t] + y[t] - x[t]^2, y'[t] == -x[t] + μ*y[t] +
                                {x,y},
                                {t,0,maxt}]

fp = fixedPoints[0.067]
```

Out[23]=

{{x → 0.485965, y → 0.203602}, {x → 0., y → 0.}}

In[24]:=
```
maxt = 20;
minx=-1;
miny=-1;
maxx=1;
maxy=1;
step = 0.2;
initialC=Join[
            Table[{minx,y},{y,miny,maxy,step}],
            Table[{maxx,y},{y,miny,maxy,step}],
            Table[{x,miny},{x,minx,maxx,step}],
            Table[{x,maxy},{x,minx,maxx,step}]];
fixedPoints[μ_] := NSolve[{f[x, y, μ] == 0, g[x, y, μ] == 0}, {x, y}]


μ = -0.1;
fp = fixedPoints[μ]
longTime = 100;
startTime = 80;
ic = {x[0] == -0.1, y[0] == 0.0};
solLC = NDSolve[
   {x'[t] == f[x[t], y[t], μ], y'[t] == g[x[t], y[t], μ], ic〚1〛, ic〚2〛},
   {x, y}, {t, 0, longTime}];
```

```
p1=Show[
    Table[
        ParametricPlot[
            Evaluate[{x[t],y[t]}/. sol[initialC[[i,1]], initialC[[i,2]], μ]],
            {t,0,maxt},
            PlotRange→{{minx,maxx},{miny,maxy}},
            AxesLabel→{x,y}
        ]
        /. Line[x_]:→{Arrowheads[{{0.05, 0.5}, {0.05, 0.0}}],Arrow[x]},{i,1,Length[init
    ],
    PlotLabel → "μ = " <> ToString[μ],
Epilog → {
   {Red, PointSize[0.02], Point[{x, y} /. fp[[1]]]},
   {Black, Text[Style["Saddle-Node", 12, Bold], {x, y} /. fp[[1]], {-1, 1}]},
   {Red, PointSize[0.02], Point[{x, y} /. fp[[2]]]},
   {Black, Text[Style["Stable-node", 12, Bold], {x, y} /. fp[[2]], {-1, 1}]}
}
];


μ = 0.035;
fp = fixedPoints[μ]
solLC = NDSolve[
   {x'[t] == f[x[t], y[t], μ], y'[t] == g[x[t], y[t], μ], ic[[1]], ic[[2]]},
   {x, y}, {t, 0, longTime}];
p2=Show[
    Table[
        ParametricPlot[
            Evaluate[{x[t],y[t]}/. sol[initialC[[i,1]], initialC[[i,2]], μ]],
            {t,0,maxt},
            PlotRange→{{minx,maxx},{miny,maxy}},
            AxesLabel→{x,y}
        ]
        /. Line[x_]:→{Arrowheads[{{0.05, 0.5}, {0.05, 0.0}}],Arrow[x]},{i,1,Length[init
    ],
    ParametricPlot[
        Evaluate[{x[t], y[t]} /. solLC],
        {t, startTime, longTime},
        PlotStyle → {Red, Thick}
      ],
    PlotLabel → "μ = " <> ToString[μ],
Epilog → {
   {Red, PointSize[0.02], Point[{x, y} /. fp[[1]]]},
   {Black, Text[Style["Saddle-Node", 12, Bold], {x, y} /. fp[[1]], {-1, 1}]},
   {Red, PointSize[0.02], Point[{x, y} /. fp[[2]]]},
   {Black, Text[Style["Unstable-node", 12, Bold], {x, y} /. fp[[2]], {-1, 1}]},
   {Black, Text[Style["Limit Cycle", 12, Bold], {-0.3, 0.4}]}
}
];
```

```
μ = 0.066;
fp = fixedPoints[μ]
solLC = NDSolve[
   {x'[t] == f[x[t], y[t], μ], y'[t] == g[x[t], y[t], μ], ic⟦1⟧, ic⟦2⟧},
   {x, y}, {t, 0, longTime}];
p3=Show[
    Table[
        ParametricPlot[
            Evaluate[{x[t],y[t]}/. sol[initialC⟦i,1⟧, initialC⟦i,2⟧, μ]],
            {t,0,maxt},
            PlotRange→{{minx,maxx},{miny,maxy}},
            AxesLabel→{x,y}
        ]
        /. Line[x_]:→{Arrowheads[{{0.05, 0.5}, {0.05, 0.0}}],Arrow[x]},{i,1,Length[init
    ],
    ParametricPlot[
        Evaluate[{x[t], y[t]} /. solLC],
        {t, startTime, longTime},
        PlotStyle → {Red, Thick}
      ],
    PlotLabel → "μ = " <> ToString[μ],
Epilog → {
   {Red, PointSize[0.02], Point[{x, y} /. fp⟦1⟧]},
   {Black, Text[Style["Saddle-Node", 12, Bold], {x, y} /. fp⟦1⟧, {-1, 1}]},
   {Red, PointSize[0.02], Point[{x, y} /. fp⟦2⟧]},
   {Black, Text[Style["Unstable-node", 12, Bold], {x, y} /. fp⟦2⟧, {-1, 1}]},
   {Black, Text[Style["Homoclinic Orbit", 12, Bold], {-0.3, 0.4}]}
   }
];


μ = 0.1;
fp = fixedPoints[μ]
p4=Show[
    Table[
        ParametricPlot[
            Evaluate[{x[t],y[t]}/. sol[initialC⟦i,1⟧, initialC⟦i,2⟧, μ]],
            {t,0,maxt},
            PlotRange→{{minx,maxx},{miny,maxy}},
            AxesLabel→{x,y}
        ]
        /. Line[x_]:→{Arrowheads[{{0.05, 0.5}, {0.05, 0.0}}],Arrow[x]},{i,1,Length[init
    ],
    PlotLabel → "μ = " <> ToString[μ],
Epilog → {
   {Red, PointSize[0.02], Point[{x, y} /. fp⟦1⟧]},
   {Black, Text[Style["Saddle-Node", 12, Bold], {x, y} /. fp⟦1⟧, {-1, 1}]},
   {Red, PointSize[0.02], Point[{x, y} /. fp⟦2⟧]},
   {Black, Text[Style["Unstable-node", 12, Bold], {x, y} /. fp⟦2⟧, {-1, 1}]}
}
```

```
];

GraphicsGrid[{{p1,p2},{p3,p4}}]
```

Out[33]=

{{x → 0.531579, y → 0.335734}, {x → 0., y → 0.}}

Out[40]=

{{x → 0.492002, y → 0.224846}, {x → 0., y → 0.}}

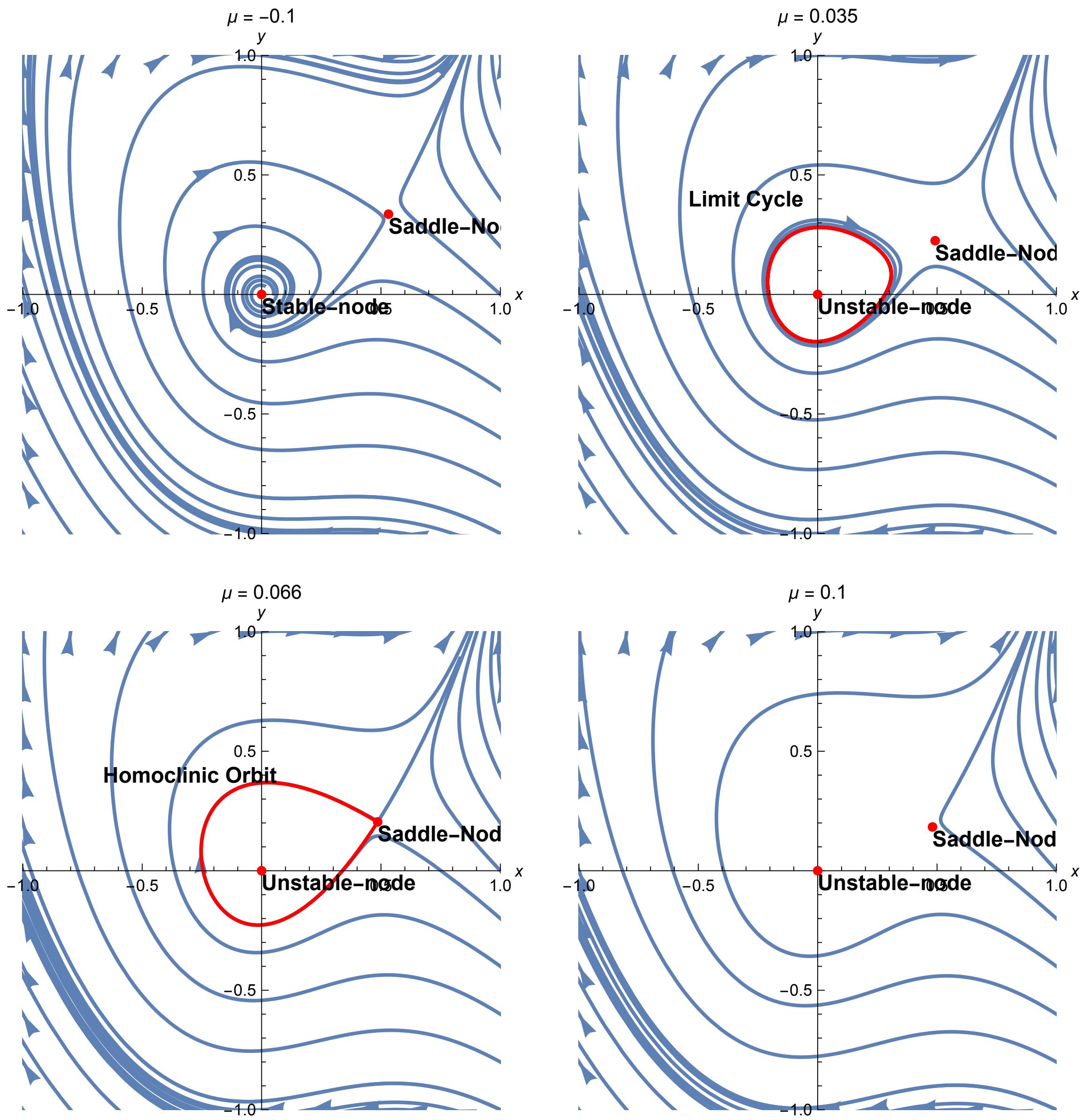
Out[44]=

{{x → 0.486136, y → 0.204243}, {x → 0., y → 0.}}

Out[48]=

{{x → 0.480952, y → 0.18322}, {x → 0., y → 0.}}

Out[50]=



In[51]:=
```
μ =.
jacobian[x_, y_, μ_] := ({
 {μ-2x, 1},
 {-1+2x, μ}
})
yFP = ((μ^2+1)/(μ+2))^2 - μ*(μ^2+1)/(μ+2)
xFP = (μ^2+1)/(μ+2)
eig = Eigenvalues[jacobian[xFP[μ], yFP[μ], μ]]
```

Out[53]=

$$-\frac{\mu\left(1+\mu^2\right)}{2+\mu}+\frac{\left(1+\mu^2\right)^2}{(2+\mu)^2}$$

Out[54]=

$$\frac{1+\mu^2}{2+\mu}$$

Out[55]=

$$\left\{\mu-\frac{1+\mu^2}{2+\mu}[\mu]-\sqrt{-1+2\,\frac{1+\mu^2}{2+\mu}[\mu]+\frac{1+\mu^2}{2+\mu}[\mu]^2},\right.$$
$$\left.\mu-\frac{1+\mu^2}{2+\mu}[\mu]+\sqrt{-1+2\,\frac{1+\mu^2}{2+\mu}[\mu]+\frac{1+\mu^2}{2+\mu}[\mu]^2}\right\}$$

In[70]:=

```
μC = 0.066;
μValues = {0.06595, 0.06585, 0.06575, 0.0656 ,0.0655, 0.065, 0.064, 0.063, 0.062, 0.06
yFP = ((μC^2+1)/(μC+2))^2 - μC*(μC^2+1)/(μC+2);
xFP = (μC^2+1)/(μC+2);
x0 = xFP - 0.1;
y0 = yFP;

longTime = 100;
startTime = 5;
ic = {x[0] == x0, y[0] == y0};




trajectoryFunc[μ_, ic_, longTime_] := NDSolve[
   {
       x'[t] == f[x[t], y[t], μ],
       y'[t] == g[x[t], y[t], μ],
       ic⟦1⟧, ic⟦2⟧
   },
   {x, y}, {t, 0, longTime}
  ];
distance[x_, y_, xFP_, yFP_] := Sqrt[(xFP - x)^2+(yFP - y)^2];

getMinimalDistance[μ_, x0_, y0_] := Module[{sol, distFunc, dmin},
  sol = trajectoryFunc[μ, {x[0] == x0, y[0] == y0}, longTime];
  distFunc[t_] := distance[x[t] /. sol, y[t] /. sol, xFP, yFP];
  dmin = NMinimize[{distFunc[t]⟦1⟧, startTime ≤ t ≤ longTime}, t];
  Return[dmin⟦1⟧]
]

data = Table[
  Module[{dmin},
    dmin = getMinimalDistance[μ, x0, y0];
    If[dmin > 0, {Log[Abs[μ - μC]], Log[dmin]}, {Log[Abs[μ - μC]], -Infinity}]
  ],
  {μ, μValues}
];

lm = LinearModelFit[data, x, x]
```

Out[83]=

```
FittedModel[ 0.119 + 0.6 x ]
```