Using *Mathematica* for Scalar 1st Order ODEs (Bifurcations)

Solving Differential Equations with various initial conditions

Let's solve the initial value problem

$$y' = y^2 - \alpha\, y,\;\; y(0) = \beta$$

for various values of the initial condition $y(0) = \beta$ and for various values of the parameter α.

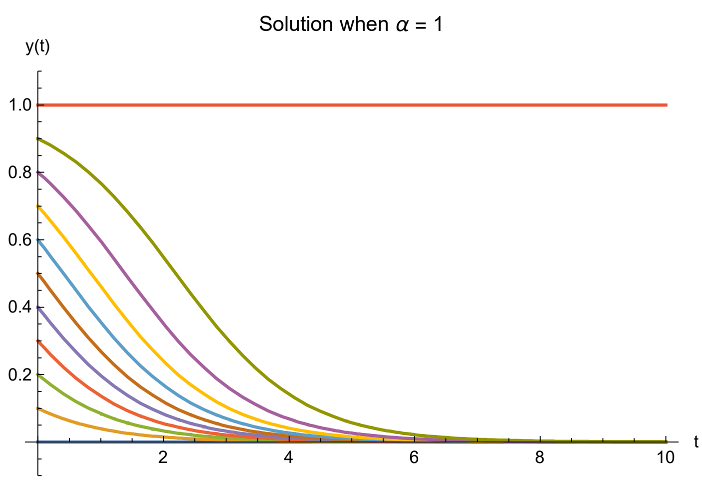
In[ ]:=
```
ode[α_] = D[y[t], t] == y[t]^2 - α y[t];
iniCond[β_] = y[0] == β;
```

In[ ]:=
```
sol[α_, β_] = FullSimplify[Quiet[DSolve[{ode[α], iniCond[β]}, y[t], t]]]
```

Out[ ]= $\left\{\left\{y[t] \to \dfrac{\alpha\,\beta}{e^{t\,\alpha}\,(\alpha-\beta)+\beta}\right\}\right\}$

In[ ]:=
```
Plot[Evaluate[Table[y[t] /. sol[α, β] /. {α → 1}, {β, 0, 1, .1}]], {t, 0, 10},
 PlotRange → {-.1, 1.1}, PlotLabel → "Solution when α = 1", AxesLabel → {"t", "y(t)"}]
```

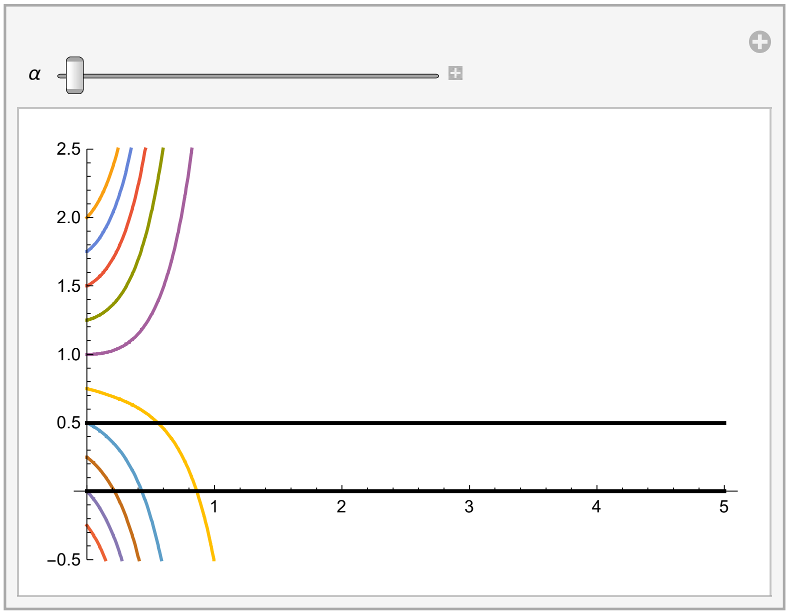
Out[ ]=



Here's we will do the same thing, but we will also show the equilibrium solutions on the plot as well. The variable p1 represents the same plot as before, while p2 represents the equilibrium solutions. The Show command allows us to display both plots on the same graph within the same Manipulate.

In[ ]:=
```
Manipulate[
 p1 = Plot[Evaluate[Table[y[t] /. sol[α, β], {β, -1, 2, .25}]], {t, 0, 5},
   PlotRange → {-.5, 2.5}, ExclusionsStyle → Directive[Black, Dashed]];
 p2 = Plot[{0, α}, {t, 0, 5},
   PlotStyle → {{Black, Thick}, {Black, Thick}},
   PlotRange → {-1, 2.5},
   ExclusionsStyle → {None, None}];
 Show[p1, p2], {α, 1 / 2, 3 / 2}]
```

Out[ ]=

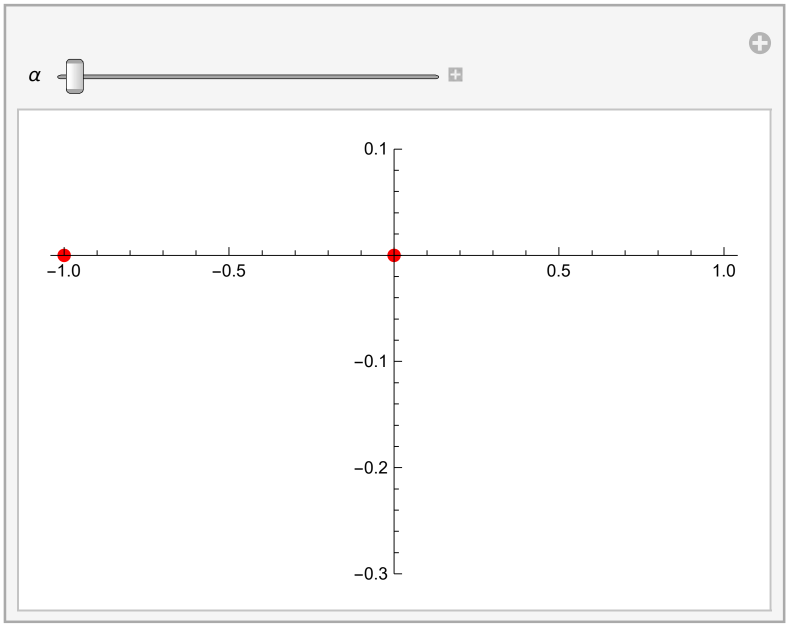


Plotting the Phase Line for various parameter values

In[ ]:=
```
F[y_, α_] = y² - α y;
```

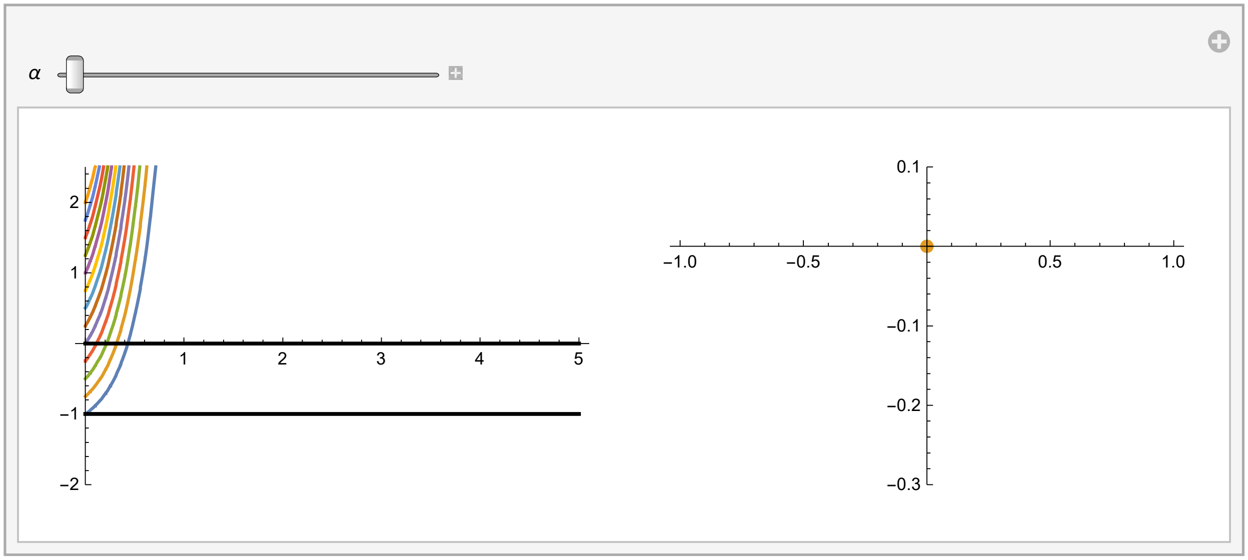
In[ ]:=
```
Manipulate[
  p3 = Plot[F[y, α], {y, -1, 2.5}, PlotRange → {-.3, .1}];
  p4 = ListPlot[{{0, 0}, {α, 0}}, PlotStyle → {Red, PointSize[Large]}];
  Show[p3, p4], {α, -1, 3 / 2}]
```

Out[ ]=



In[ ]:=
```
Manipulate[
 p1 = Plot[Evaluate[Table[y[t] /. sol[α, β], {β, -1, 2, .25}]], {t, 0, 5},
   PlotRange → {-2, 2.5}, ExclusionsStyle → Directive[Black, Dashed]];
 p2 = Plot[{0, α}, {t, 0, 5},
   PlotStyle → {{Black, Thick}, {Black, Thick}},
   PlotRange → {-1, 2.5},
   ExclusionsStyle → {None, None}];
 PA = Show[p1, p2];
 p3 = Plot[F[y, α], {y, -1, 2.5}, PlotRange → {-.3, .1}];
 p4 = ListPlot[{{0, 0}, {A, 0}}, PlotStyle → {Red, PointSize[Large]}];
 PB = Show[p3, p4];
 GraphicsGrid[{{PA, PB}}], {α, -1, 3 / 2}]
```

Out[ ]=



Solving for the Equilibrium Points

Let's define the function F(p,h) as well as the derivative with respect to p. Notice that I take the derivative with respect to a separate variable "P" and then substitute in "p". This is so that the function FPrime returns the value of df/dp evaluated at p.

In[ ]:=
```
F[y_, α_] = y^2 - α y
FPrime[y_, α_] = D[F[Y, α], Y] /. Y → y;
```

Out[ ]= $y^2 - y\,\alpha$

In[ ]:=
```
eqPts[α_]  =  Solve[F[y, α] == 0, y]
```

Out[ ]= {{y → 0}, {y → α}}

We can evaluate f'(p,h) at the equilibrium points using the following:

In[ ]:=
```
Simplify[FPrime[y, α] /. eqPts[α][[1]]]
```
Out[ ]= $-\alpha$

In[ ]:=
```
Simplify[FPrime[y, α] /. eqPts[α][[2]]]
```
Out[ ]= α

Note that eqPts[[1]] returns the first equilibrium point that the Solve command returned. We can even plot both equilibrium points by defining the `BifurcationPlot` command below (thanks to Gavin & Gabriel for the link):

In[ ]:=
```
BifurcationPlot[eqpts_, fPrime_, varrange_, plotlabels_] :=
 Module[{plotStable, plotUnstable, condition},
  For[j = 1, j ≤ Length[eqpts], j++,
   condition = (fPrime /. eqpts[[j]]) > 0;
   plotStable_j =
    Plot[If[condition, y /. eqpts[[j]]], {α, -1, 1}, PlotStyle → {Black, Thick}];
   plotUnstable_j = Plot[If[Not[condition], y /. eqpts[[j]]],
     {α, -1, 1}, PlotStyle → {Red, Thick, Dashed}]];
  Show[Table[{plotStable_j, plotUnstable_j}, {j, 1, Length[eqpts]}],
   PlotRange → All, plotlabels]]
```

Now that we have defined the “BifurcationPlot” command, we can call it as follows:

In[ ]:=
```
BifurcationPlot[eqPts[α], FPrime[y, α], {α, -1, 1},
 {PlotLabel → "Bifurcation Diagram for y' = y^2-α y", AxesLabel → {"α", "y"}}]
```

Out[ ]=