NLCEQ: The basic (and trivial) idea

In[2073]:=

```
x = 0; Remove["Global`*"]; DateList[Date[]] // Most
```

Out[2073]=

```
{2020, 4, 7, 8, 47}
```

Bellman Equation

Examples with closed-form solution

Optimal growth model.
u[c]: utility function
f[k]: gross output function

In[2074]:=
```
Bellman = -Vnow[k] + u[c] + β Vnxt[f[k] - c];
```

u[c] - Vnow[k] + β Vnxt[-c + f[k]]

In[2075]:=
```
foc = D[Bellman, c]
```

Out[2075]= u′[c] - β Vnxt′[-c + f[k]]

In[2076]:=
```
Solve[foc == 0, Vnxt′[-c + f[k]]][[1]]
```

Out[2076]= $\left\{\text{Vnxt}'[-c + f[k]] \to \frac{u'[c]}{\beta}\right\}$

In[2077]:=
```
env = D[Bellman, k]
```

Out[2077]= -Vnow′[k] + β f′[k] Vnxt′[-c + f[k]]

Tastes and Technology

Assume that the production function is Cobb-Douglas (A is chosen so that steady state is k=1)

In[2078]:=
```
Clear[f]; f[k_] = A k^α; A = 1/(α β);
```

$$f[k\_] = A\,k^{\alpha};\quad A = \frac{1}{\alpha\,\beta};$$

and that the utility function is the log function

In[2079]:=
```
u[c_] = Log[c];
```

Closed-form solutions for value function and consumption functions

In[2080]:=
```
Vtrue[k_] = - (α Log[k])/(-1 + α β) - Log[(1-α β)/(α β)]/(-1 + β)
```

Out[2080]=
$$-\frac{\alpha \operatorname{Log}[k]}{-1+\alpha\beta}-\frac{\operatorname{Log}\left[\frac{1-\alpha\beta}{\alpha\beta}\right]}{-1+\beta}$$

In[2081]:=
```
θ = 1 - α β; Ctrue[k_] = θ f[k]
```

Out[2081]=
$$\frac{k^{\alpha}(1-\alpha\beta)}{\alpha\beta}$$

plots

In[2082]:=
```
sav = f[k] - Ctrue[k] - k;
```

In[2083]:=
```
% // Together // Numerator
```

Out[2083]= $-k + k^{\alpha}$

In[2084]:=
```
kss = 1;
```

In[2085]:=
```
α = 1 / 4;
β = 95 / 100;
```

In[2087]:=
```
true = Ctrue[k]
```

Out[2087]= $$\frac{61\,k^{1/4}}{19}$$

In[2088]:=
```
lin = Series[Ctrue[k], {k, kss, 1}] // Normal
```

Out[2088]= $$\frac{61}{19} + \frac{61}{76}(-1 + k)$$

In[2089]:=
```
quad = Series[Ctrue[k], {k, kss, 2}] // Normal
```

Out[2089]= $$\frac{61}{19} + \frac{61}{76}(-1 + k) - \frac{183}{608}(-1 + k)^2$$

In[2090]:=
```
cub = Series[Ctrue[k], {k, kss, 3}] // Normal
```

Out[2090]= $$\frac{61}{19} + \frac{61}{76}(-1 + k) - \frac{183}{608}(-1 + k)^2 + \frac{427\,(-1 + k)^3}{2432}$$

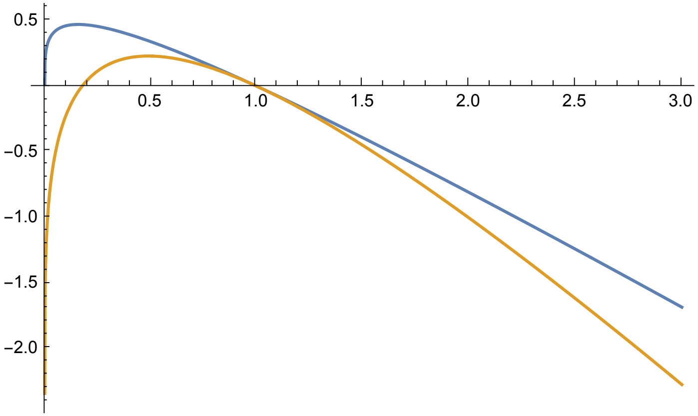
In[2091]:=
```
quar = Series[Ctrue[k], {k, kss, 4}] // Normal
```

Out[2091]= $$\frac{61}{19} + \frac{61}{76}(-1 + k) - \frac{183}{608}(-1 + k)^2 + \frac{427\,(-1 + k)^3}{2432} - \frac{4697\,(-1 + k)^4}{38\,912}$$

savings

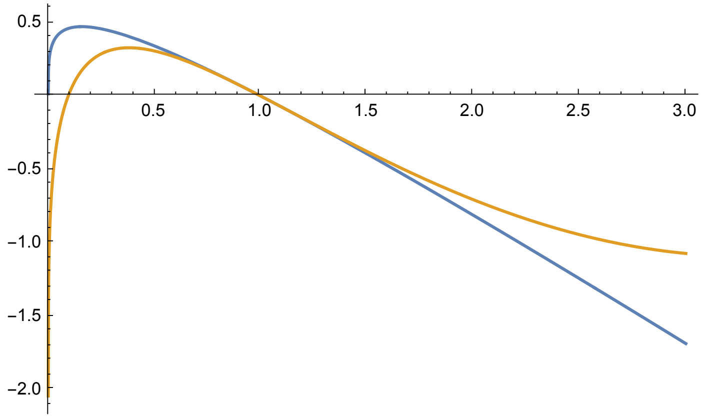
In[2093]:=
```
Plot[{f[k] - true - k, f[k] - lin - k}, {k, 0, 3}]
```

Out[2093]=



In[2094]:=
```
Plot[{f[k] - true - k, f[k] - quad - k}, {k, 0, 3}]
```

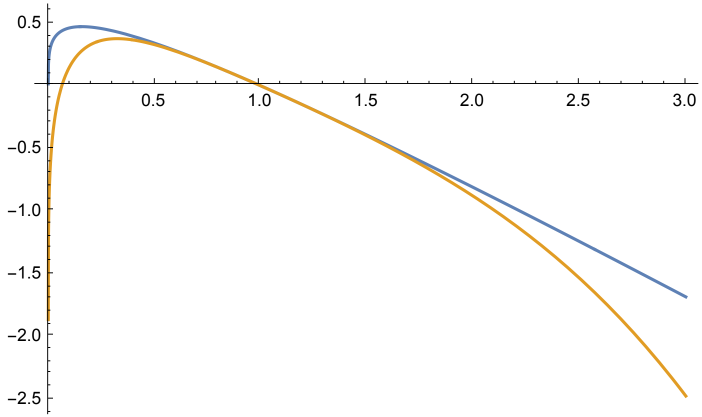
Out[2094]=



In[2095]:=

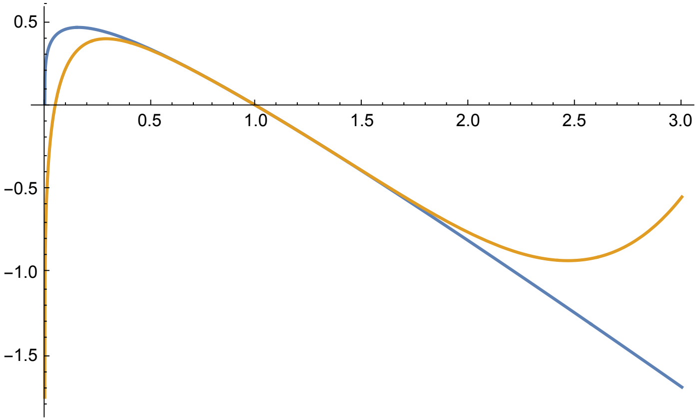
```
Plot[{f[k] - true - k, f[k] - cub - k}, {k, 0, 3}]
```

Out[2095]=



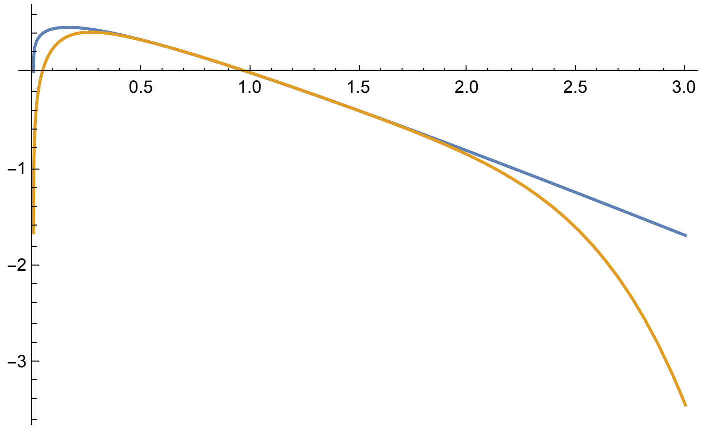
In[2096]:= `Plot[{f[k] - true - k, f[k] - quar - k}, {k, 0, 3}]`

Out[2096]=



In[2097]:=
```
Plot[{f[k] - true - k, f[k] - quin - k}, {k, 0, 3}]
```

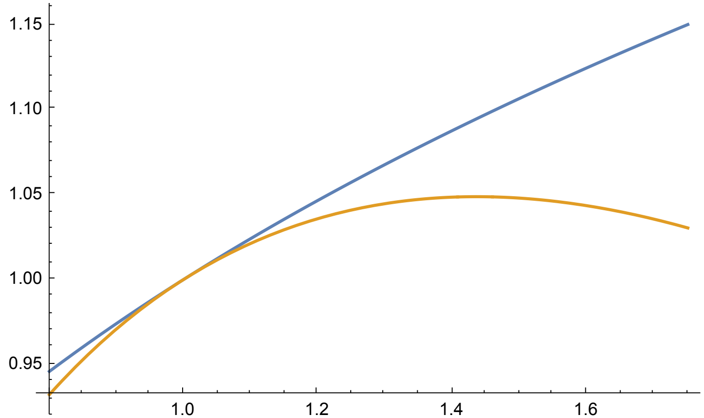
Out[2097]=



next period's k

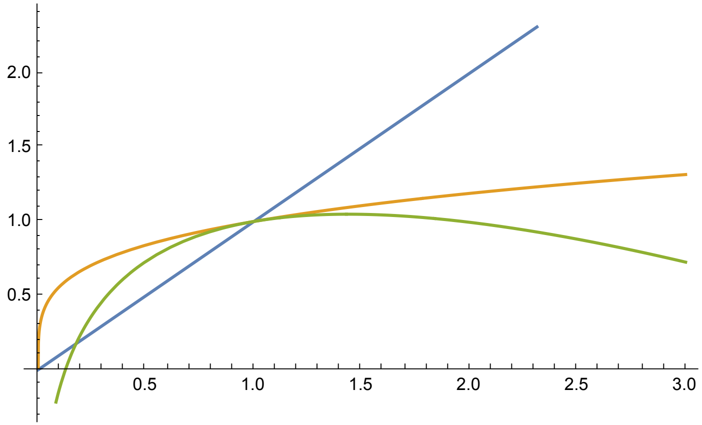
```
In[2098]:= Plot[{f[k] - true, f[k] - lin}, {k, 0.8, 1.75}]
```

Out[2098]=



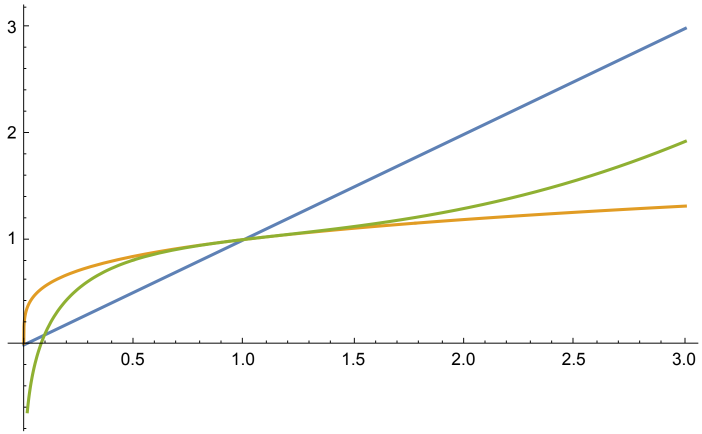
In[2099]:=
```
Plot[{k, f[k] - true, f[k] - lin}, {k, 0, 3}]
```

Out[2099]=



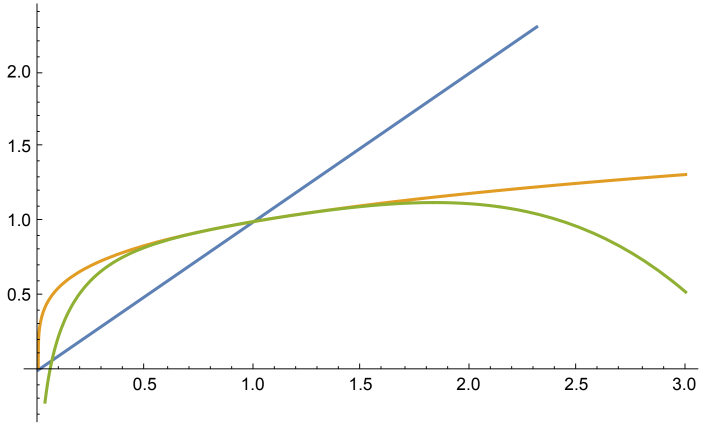
In[2100]:=
```
Plot[{k, f[k] - true, f[k] - quad}, {k, 0, 3}]
```

Out[2100]=



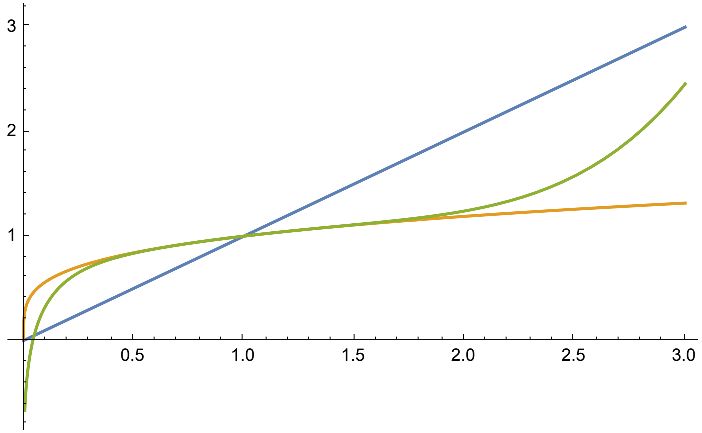
In[2101]:= `Plot[{k, f[k] - true, f[k] - cub}, {k, 0, 3}]`

Out[2101]=



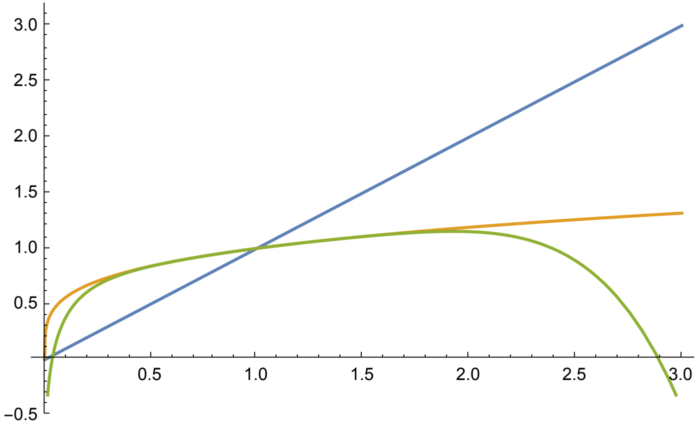
In[2102]:= `Plot[{k, f[k] - true, f[k] - quar}, {k, 0, 3}]`

Out[2102]=



In[2103]:= `Plot[{k, f[k] - true, f[k] - quin}, {k, 0, 3}]`

Out[2103]=



nlceq

List states

```
In[2104]:= states = Range[30] 0.1

Out[2104]= {0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1., 1.1, 1.2, 1.3, 1.4, 1.5,
 1.6, 1.7, 1.8, 1.9, 2., 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.}
```

Compute initial consumption given initial state (normally using numerical method)

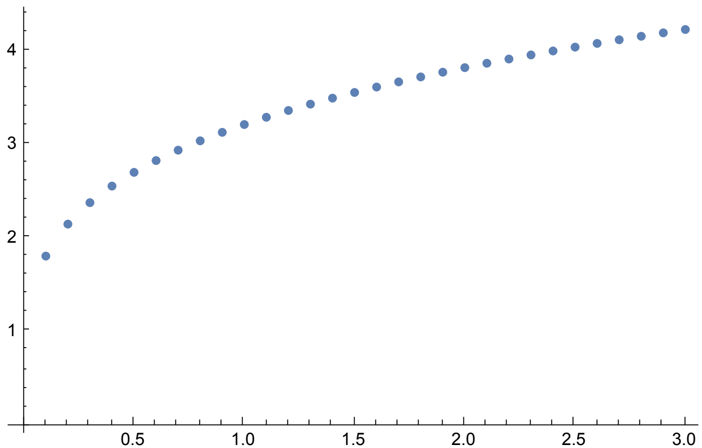
```
In[2105]:= cons = Ctrue /@ states

Out[2105]= {1.80541, 2.14701, 2.37606, 2.55324, 2.69972, 2.82562, 2.93664,
 3.03633, 3.12706, 3.21053, 3.28794, 3.36025, 3.42817, 3.49227,
 3.55303, 3.61082, 3.66597, 3.71873, 3.76933, 3.81798, 3.86484, 3.91005,
 3.95374, 3.99603, 4.03702, 4.0768, 4.11545, 4.15304, 4.18963, 4.22529}

In[2106]:= data = {states, cons} // Transpose;

In[2107]:= ListPlot[data]
```

Out[2107]=



In[2108]:= `ffit[k_] = Fit[data, {1, k, k^2, k^3, k^4, k^5, k^6, k^7}, k]`

Out[2108]= $1.37216 + 5.13137\,k - 8.16326\,k^2 + 8.86605\,k^3 - 5.77319\,k^4 + 2.17343\,k^5 - 0.435358\,k^6 + 0.0358522\,k^7$

In[2109]:= `Plot[ffit[k] - true, {k, 0, 3}]`

Out[2109]=