2. A pig weighing 200 pounds gains 5 pounds per day and costs 45 cents a day to keep. Suppose the price for pigs after t days is

$$p(t) = 0.65e^{-(.01/.65)t} \text{ dollars/pound.}$$

(i) Show the price for pigs is falling by one cent/day at time $t = 0$. What happens as t increases?

Differentiating as

$$p'(t) = 0.65e^{-(.01/.65)t}\left(\frac{.01}{.64}\right) = -.01e^{-(.01/.65)t}$$

and evaluating at $t = 0$ as

$$p'(0) = -.01e^{-(.01/.65)(0)} = -.01e^{0} = -.01$$

shows the price is falling by one cent/day.

As t increases the value of the pig decreases exponentially. This means the amount of decrease is less over time. Note, unlike the linear approximation, the price per pound of the pig never goes negative.

Graphically the price is

In [1]:
```
using Symbolics
D(f,x)=expand_derivatives(Differential(x)(f))
@variables t
```

Out[1]:
```
1-element Vector{Num}:
 t
```

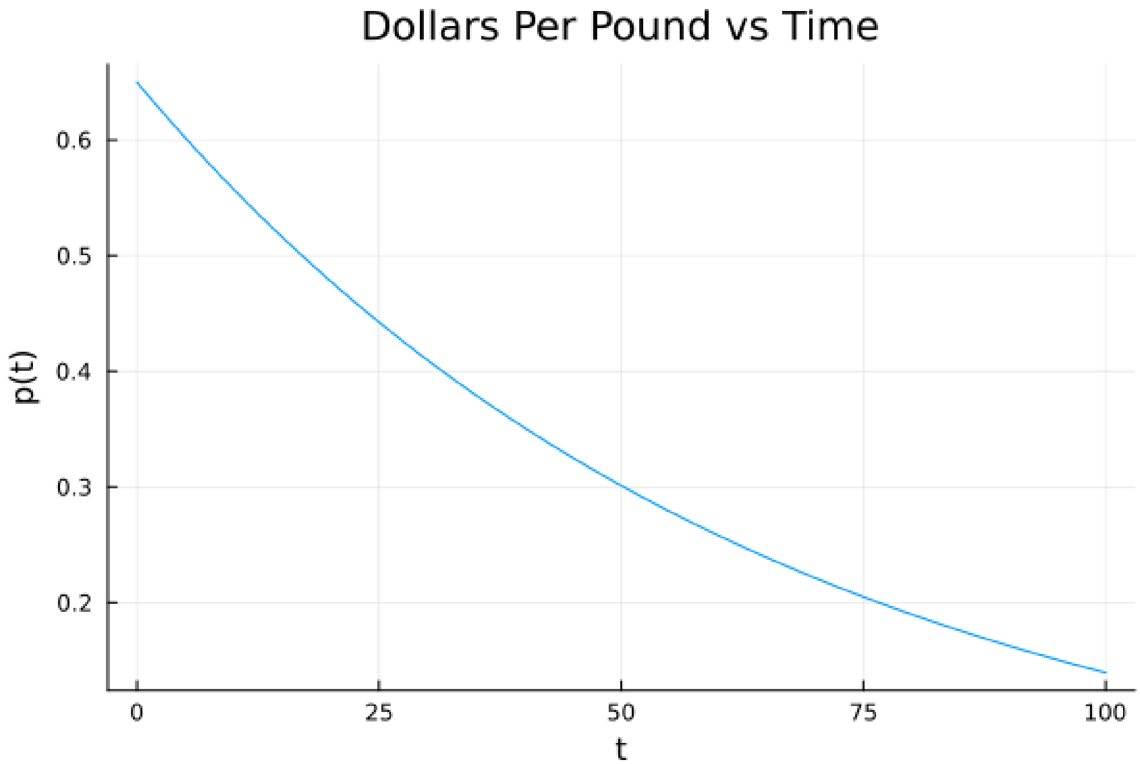
In [2]:
```
p(t)=0.65*exp(-(.01/.65)*t)
p(t)
```

Out[2]:
```
0.65exp(-0.015384615384615384t)
```

In [3]:
```
using Plots
```

In [4]:
```
plot(p,0:0.1:100,
    xlabel="t",ylabel="p(t)",title="Dollars Per Pound vs Time",
    legend=false)
```

Out[4]:



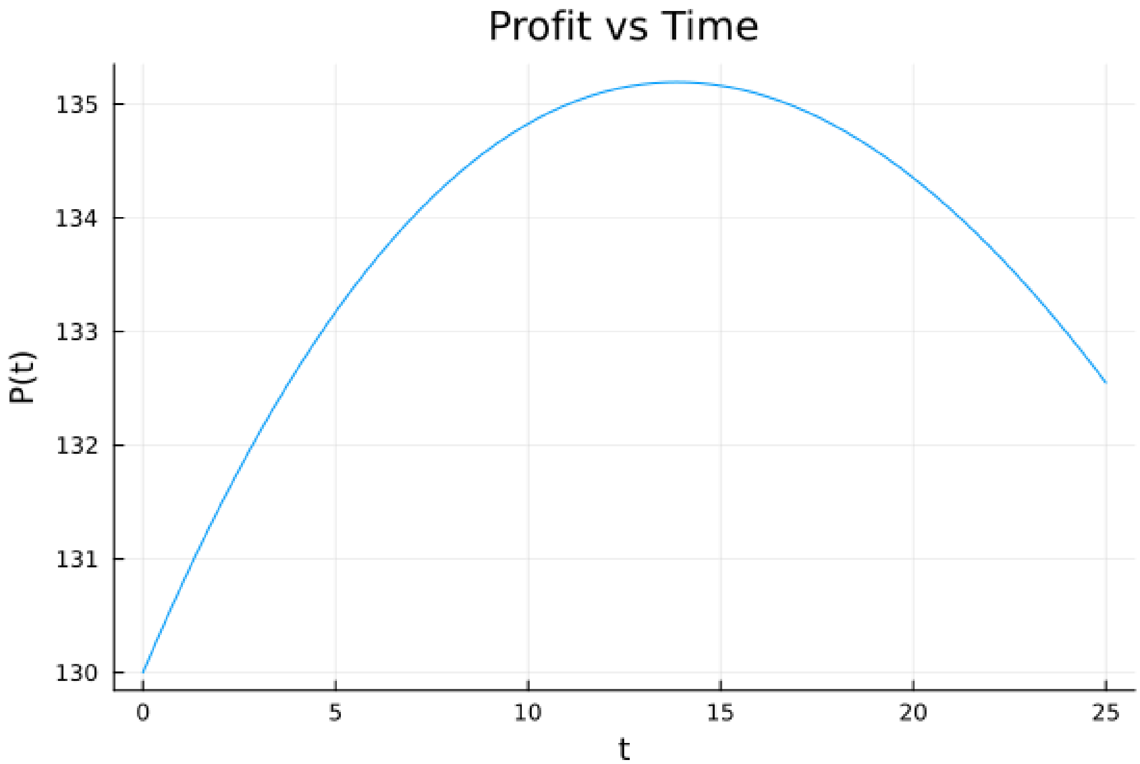
(ii) Plot the profit $P(t)$ versus the time t to sell the pig.

In [5]:
```
w(t)=200+5*t
C(t)=0.45*t
R(t)=w(t)*p(t)
P(t)=R(t)-C(t)
```

Out[5]: P (generic function with 1 method)

In [6]:
```
plot(P,0:0.1:25,
    xlabel="t",ylabel="P(t)",title="Profit vs Time",
    legend=false)
```

Out[6]:



Visual inspection of the plot suggests the optimal time to sell the pig is arount $t \approx 14$.

(iii) Find the optimal time to sell the pig using Newton's method. Model as a one-variable non-linear optimization problem.

In [7]:
```
dpn=expand(D(P(t),t))
dps="dp(t)="*string(Symbolics.toexpr(dpn))
eval(Meta.parse(dps))
dp(t)
```

Out[7]:
```
-0.45 + 1.25exp(-0.015384615384615384t) - 0.05t*exp(-0.015384615384615384t)
```

To find the maximum of of $P(t)$ find t such that $P'(t) = 0$. Do this by writing $g(t) = P'(t)$ and then using Newton's method

$$t_{n+1} = t_n - g(t_n)/g'(t_n) \qquad \text{where} \qquad t_0 = 14.$$

Note the value $t_0 = 14$ was chosen because that is close to the observed maximum in the graph.

In [8]:
```
g(t)=dp(t)
dgn=expand(D(g(t),t))
dgs="dg(t)="*string(Symbolics.toexpr(dgn))
eval(Meta.parse(dgs))
dg(t)
```

Out[8]:
```
-0.06923076923076923exp(-0.015384615384615384t) + 0.0007692307692307692t*ex
p(-0.015384615384615384t)
```

In [9]:
```
tn=14
for n=1:4
    tn=tn-g(tn)/dg(tn)
    println(n," ",tn)
end
topt=tn
Popt=P(topt)
println("\ntopt=",topt)
println("Popt=",Popt)
```

```
1 13.860549574911676
2 13.860826286794973
3 13.860826287886802
4 13.860826287886802

topt=13.860826287886802
Popt=135.19404935454668
```

The optimal time to sell the pig is $t_{\text{opt}} \approx 13.86$ and the optimal profit is $P_{\text{opt}} \approx 135.19$.

(iv) The parameter 0.01 represents the rate α at which price is falling at $t = 0$. Thus,

$$p(t) = 0.65e^{-(\alpha/.65)t} \text{ dollars/pound.}$$

Perform a sensitivity analysis with respect to the parameter α. Compute $S(t, \alpha)$ and $S(P, \alpha)$ evaluated at $\alpha = 0.01$. Here P is the profit obtained at the optimal time t to sell the pig.

Since $g(t_{\text{opt}}) = 0$ use implicit differentiation to compute $dt/d\alpha$ as follows

$$\frac{dg}{d\alpha} = \frac{\partial g}{\partial t}\frac{dt}{d\alpha} + \frac{\partial g}{\partial \alpha} = 0.$$

It follows that

$$\left.\frac{dt}{d\alpha}\right|_{t=t_{\text{opt}}} = -\frac{\partial g}{\partial \alpha} \Big/ \left.\frac{\partial g}{\partial t}\right|_{t=t_{\text{opt}}}.$$

In [10]:
```
@variables alpha
p(t)=0.65*exp(-(alpha/.65)*t)
p(t)
```

Out[10]:
```
0.65exp(-1.5384615384615383alpha*t)
```

In [11]:
```
P(t)
```

Out[11]:
```
-0.45t + 0.65(200 + 5t)*exp(-1.5384615384615383alpha*t)
```

In [12]:
```
dpn=expand(D(P(t),t))
dps="dp(t)="*string(Symbolics.toexpr(dpn))
eval(Meta.parse(dps))
dp(t)
```

Out[12]:
```
-0.45 + 3.25exp(-1.5384615384615383alpha*t) - 200.0alpha*exp(-1.53846153846
15383alpha*t) - 5.0alpha*t*exp(-1.5384615384615383alpha*t)
```

In [13]:
```
g(t)=dp(t)
pgpalpha=D(g(t),alpha)
```

Out[13]:
```
-200.0exp(-1.5384615384615383alpha*t) - 10.0t*exp(-1.5384615384615383alpha*
t) + 307.6923076923077alpha*t*exp(-1.5384615384615383alpha*t) + 7.692307692
307692alpha*(t^2)*exp(-1.5384615384615383alpha*t)
```

In [14]:
```
pgpt=D(g(t),t)
```

Out[14]:
```
-10.0alpha*exp(-1.5384615384615383alpha*t) + 307.6923076923077(alpha^2)*exp
(-1.5384615384615383alpha*t) + 7.692307692307692(alpha^2)*t*exp(-1.53846153
84615383alpha*t)
```

In [15]:
```
vopt=[t=>topt,alpha=>0.01]
dtdalpha=substitute(-pgpalpha/pgpt,vopt)
```

Out[15]:
```
-4800.881966946403
```

In [16]:
```
Stalpha=0.01/topt*dtdalpha
println("S(t,alpha)=",Stalpha)
```

```
S(t,alpha)=-3.4636333124973726
```

To compute $dP/d\alpha$ at $t = t_{\text{opt}}$ note $P'(t_{\text{opt}}) = 0$ implies

$$\frac{dP}{d\alpha} = \frac{\partial P}{\partial t}\frac{d\alpha}{d\alpha} + \frac{\partial P}{\partial \alpha} = \frac{\partial P}{\partial \alpha}\bigg|_{t=t_{\text{opt}}}.$$

In [17]:
```
dPdalphan=substitute(D(P(t),alpha),vopt)
```

Out[17]:
```
-3732.7777844922293exp(-0.21324348135210464)
```

In [18]:
```
# Evaluate the expression
dPdalpha=eval(Symbolics.toexpr(dPdalphan))
```

Out[18]:
```
-3015.9328625872377
```

In [19]:
```
SPalpha=0.01/Popt*dPdalpha
println("S(P,alpha)=",SPalpha)
```

```
S(P,alpha)=-0.22308177593511883
```

In summary, the sensitivities are

$$S(t,\alpha) \approx -3.46 \qquad \text{and} \qquad S(P,\alpha) \approx -0.223.$$

Estimate using the difference quotients

$$\frac{dt}{d\alpha} \approx \frac{\Delta t}{\Delta \alpha} \qquad \text{and} \qquad \frac{dP}{d\alpha} \approx \frac{\Delta P}{\Delta \alpha}$$

to confirm the above sensitivities.

In [20]:
```
# Take the perturbed alpha close to 0.01 for accuracy
alphanew=0.01001
alpha=alphanew
p(t)
```

Out[20]:
```
0.65exp(-0.015399999999999999t)
```

In [21]:
```
dpn=expand(D(P(t),t))
dps="dp(t)="*string(Symbolics.toexpr(dpn))
eval(Meta.parse(dps))
dp(t)
```

Out[21]:
```
-0.45 + 1.2480000000000002exp(-0.015399999999999999t) - 0.05005t*exp(-0.015399999999999999t)
```

In [22]:
```
g(t)=dp(t)
dgn=expand(D(g(t),t))
dgs="dg(t)="*string(Symbolics.toexpr(dgn))
eval(Meta.parse(dgs))
dg(t)
```

Out[22]:
```
-0.0692692exp(-0.015399999999999999t) + 0.0007707699999999999t*exp(-0.015399999999999999t)
```

In [23]:
```
tn=14
for n=1:4
    tn=tn-g(tn)/dg(tn)
    println(n," ",tn)
end
tnew=tn
Pnew=P(tnew)
println("\ntnew=",tnew)
println("Pnew=",Pnew)
```

```
1 13.812362974310679
2 13.812864114635024
3 13.812864118219844
4 13.812864118219846

tnew=13.812864118219846
Pnew=135.1639477232832
```

In [24]:
```
deltat=(tnew-topt)/(alphanew-0.01)
```

Out[24]:
```
-4796.216966695805
```

In [25]:
```
deltaP=(Pnew-Popt)/(alphanew-0.01)
```

Out[25]: -3010.163126347843

The consistency between

$$-4800.881 \approx \frac{dt}{d\alpha} \approx \frac{\Delta t}{\Delta \alpha} \approx -4796.22 \qquad \text{and} \qquad -3015.93 \approx \frac{dP}{d\alpha} \approx \frac{\Delta P}{\Delta \alpha}$$

suggest the earlier computations for $S(t, \alpha)$ and $S(P, \alpha)$ are correct.

In [ ]: