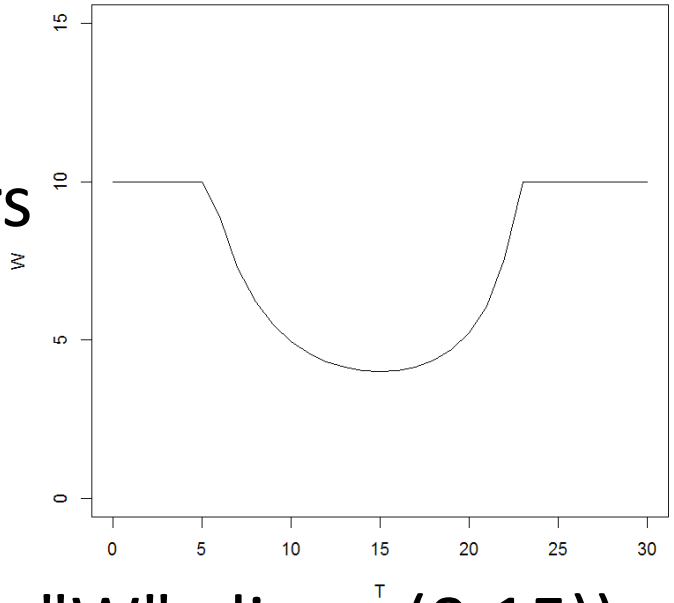
Correction to nls exercise

1. Write a function that calculates W as a function of T given Tmin, Topt, Tmax, Wmin, Wmax

```
magarey<-function (T, Tmin, Topt, Tmax, Wmin, Wmax)
{
   fT <- ((Tmax - T)/(Tmax - Topt)) *
         (((T - Tmin)/(Topt - Tmin))^((Topt - Tmin)/(Tmax - Topt)))

   W <- Wmin/fT
   W[W > Wmax | T < Tmin | T > Tmax] <- Wmax
   return(W)
}
```

2. Use the parameters of Bremia lactucae to calculate W as a function of temperature. Draw the graph.

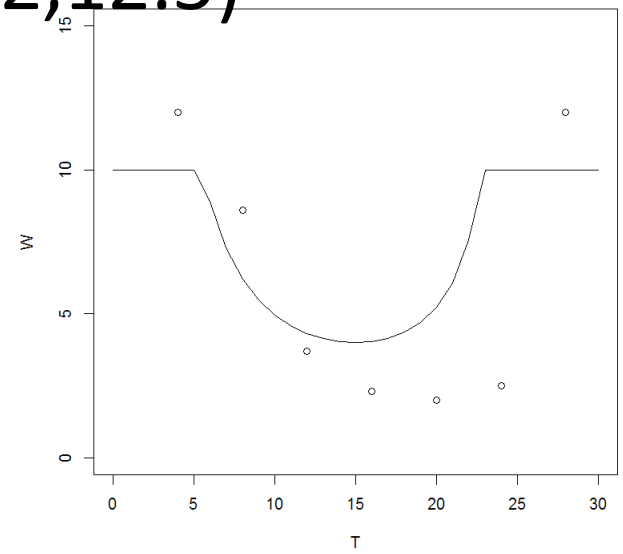



```
# Use Bremia lactucae prameters
T<-0:30
W<-magarey(T,1,15,25,4,10)
cbind(T,W)
plot(T,W,type="l", xlab="T", ylab="W",ylim=c(0,15))
lines(TData,WData,type="p")
```

3. Add the points for Pseudoperonospora cubensis to the graph

```
# data for Pseudoperonospora cubensis
TData<-c(4,8,12,16,20,24,28,32)
WData<-c(12,8.6,3.7,2.3,2,2.5,12,12.5)
cbind(TData,WData)

lines(TData,WData,type="p")
```



4. Use the data for Pseudoperonospora cubensis to calibrate the model for that disease.

```
# nls for Pseudoperonospora cubensis
# start from Bremia lactucae parameters
TminStart<-1
ToptStart<-15
TmaxStart<-25
WminStart<-4
WmaxStart<-10

pseudo<-nls(WData~magarey(T,Tmin,Topt,Tmax,Wmin,Wmax),
            data=list(T=TData),
            start=list(Tmin=TminStart,Topt=ToptStart,
            Tmax=TmaxStart,Wmin=WminStart,Wmax=WmaxStart))
```

5. Examine the output of nls. Are all the parameters well estimated?

summary(pseudo)

> summary(pseudo)

Formula: WData ~ magarey(T, Tmin, Topt, Tmax, Wmin, Wmax)

Parameters:

	Estimate	Std. Error	t value	Pr(>|t|)	
Tmin	0.5237	4.8833	0.107	0.921366	
Topt	20.0918	0.4772	42.106	2.95e-05	***
Tmax	27.9218	1.9403	14.391	0.000727	***
Wmin	1.9762	0.1969	10.034	0.002107	**
Wmax	12.1667	0.1368	88.930	3.13e-06	***

Tmin poorly estimated

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.237 on 3 degrees of freedom

Number of iterations to convergence: 7
Achieved convergence tolerance: 8.871e-06

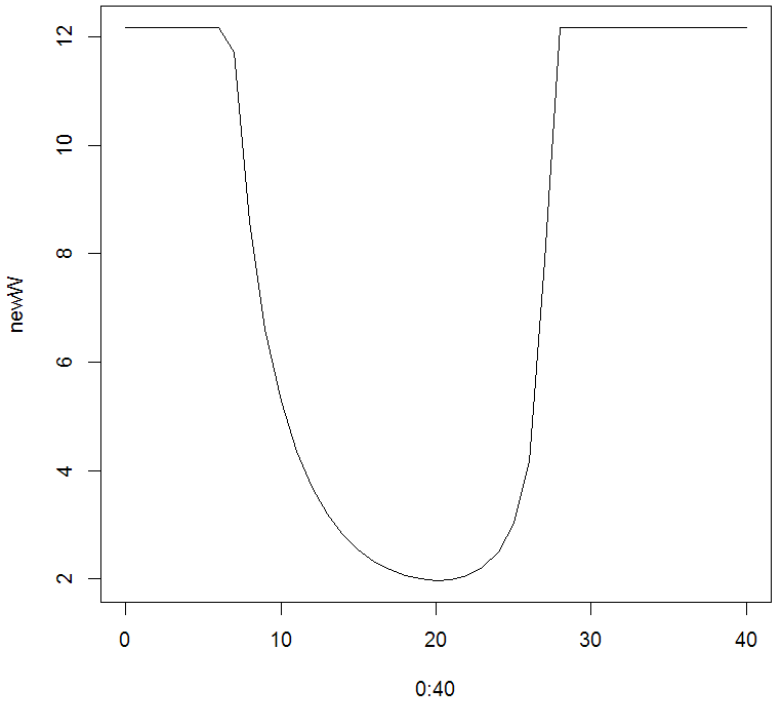
6. Use the calibrated model to predict W for T<-0:40

```
newW<-predict(pseudo,newdata=list(T=0:40))
```

```
> predict(pseudo,newdata=list(T=0:40))
 [1] 12.166667 12.166667 12.166667 12.166667 12.166667 12.166667 12.166667 12.166667 11.725234  8.600955
[10]  6.616999  5.286920  4.357655  3.687709  3.193371  2.822878  2.543163  2.332689  2.177443
[19]  2.068704  2.001906  1.976350  1.995838  2.070805  2.223686  2.503037  3.027119  4.164034
[28]  7.885149 12.166667 12.166667 12.166667 12.166667 12.166667 12.166667 12.166667 12.166667
[37] 12.166667 12.166667 12.166667 12.166667 12.166667
```

7. Graph the model results for T from 0 to 40°C (type=“l”)

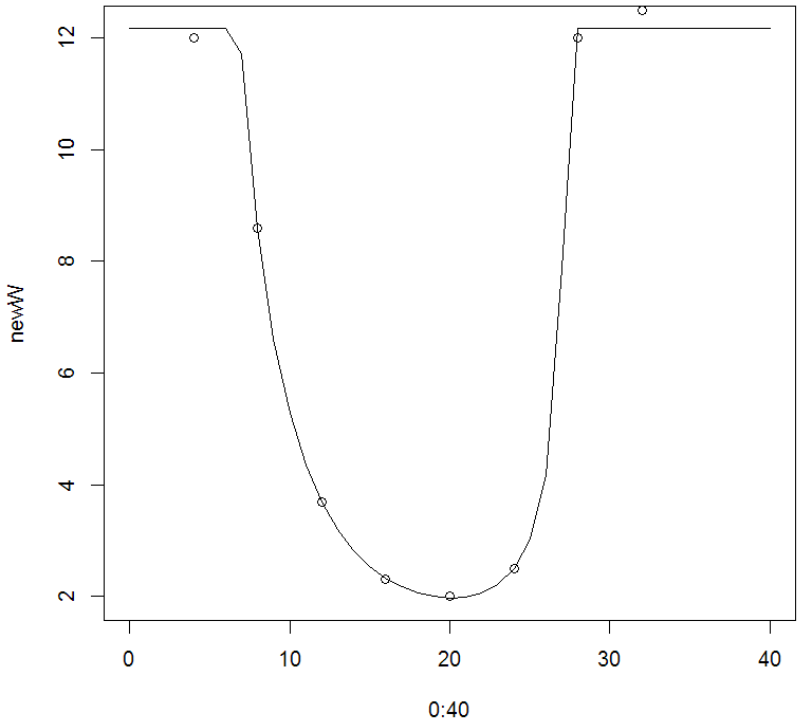
plot(0:40,newW,type="l")



8. Add the data points. How well does the model fit the data?

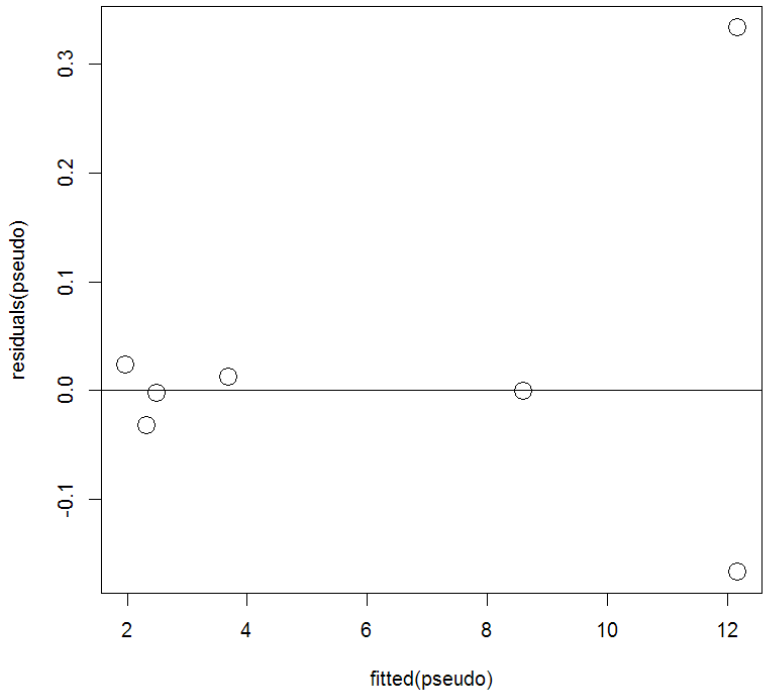
lines(TData,WData,type="p")

Very good fit



9. Plot residuals (observed- simulated) versus simulated values. Add horizontal line at y=0

```
plot(fitted(pseudo),residuals(pseudo),cex=2)
abline(0,0)
```



10. Is this a “correct model”

11. Are the residuals “homoscedastic”?

- Model looks o.k.
  - No obvious structure in residuals
- Is variance constant?
  - Maybe not, hard to tell, not enough data

END