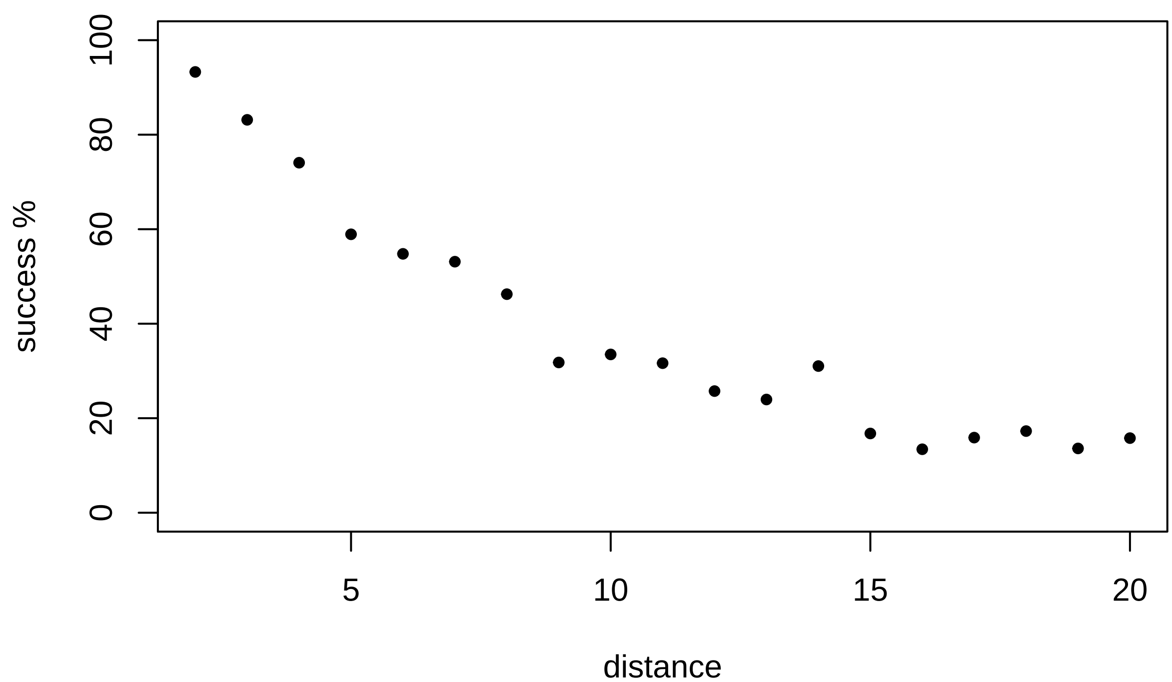
Assign4

Daniel Smith

4/14/2021

```
library(rjags)
putts=read.table('putting.dat',header=T)
plot(putts$dist,100*putts$Nsucc/putts$Ntrys,xlab='distance',ylab='success %', ylim=c(0,100), pch=20)
```




```
#Q1.
# a
# see model in Appendix 1
putting.model=jags.model("putt.model", putts)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
```

```
##    Observed stochastic nodes: 19
##    Unobserved stochastic nodes: 2
##    Total graph size: 118
##
## Initializing model
```

```r
putting.samps=coda.samples(putting.model,
variable.names = c("alpha", "beta"), n.iter=1e4)

HPDinterval(putting.samps)
```

```
## [[1]]
##            lower      upper
## alpha  2.1212002  2.3487315
## beta  -0.2699043 -0.2438064
## attr(,"Probability")
## [1] 0.95
```

```r
# b

# see model in Appendix 2
putts1=read.table("putting1.dat", header=TRUE)
putting1.model=jags.model("putt1.model", putts1)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 19
##    Unobserved stochastic nodes: 23
##    Total graph size: 145
##
## Initializing model
```

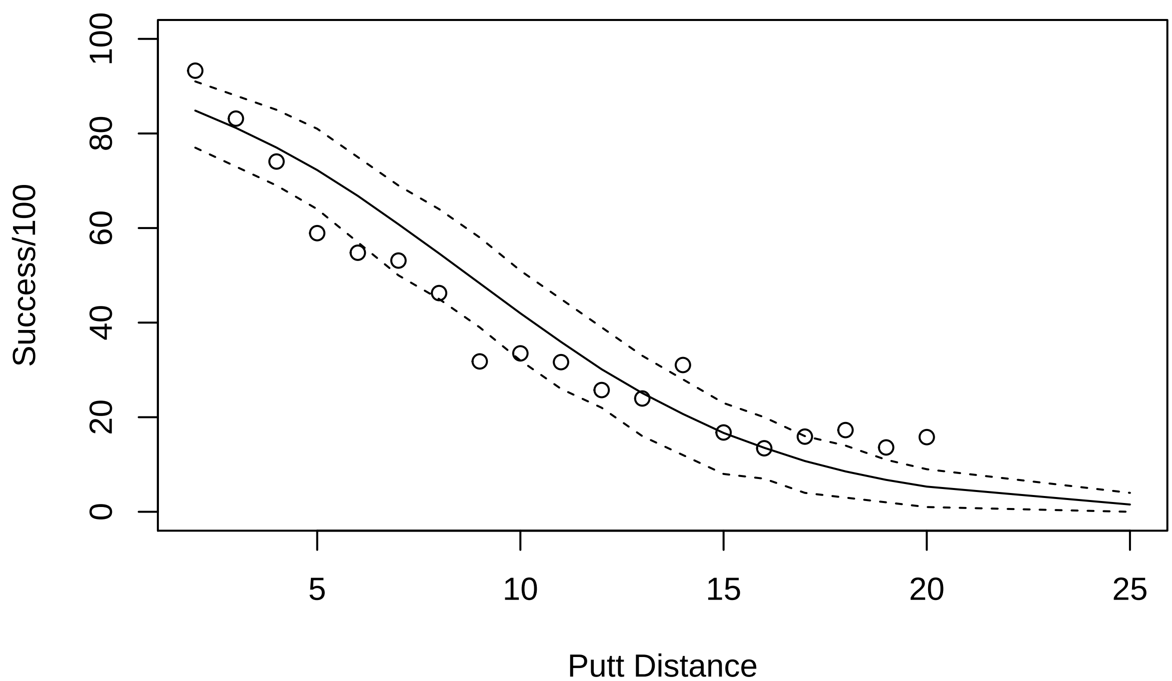
```r
putting1.samps=coda.samples(putting1.model,
                            variable.names = c("Nsuccstar"), n.iter=1e4)
HDR=HPDinterval(putting1.samps[[1]])
HDR
```

```
##               lower upper
## Nsuccstar[1]     77    91
## Nsuccstar[2]     73    88
## Nsuccstar[3]     69    85
## Nsuccstar[4]     64    81
## Nsuccstar[5]     57    75
## Nsuccstar[6]     50    69
## Nsuccstar[7]     45    64
## Nsuccstar[8]     39    58
## Nsuccstar[9]     32    51
## Nsuccstar[10]    26    45
## Nsuccstar[11]    22    39
## Nsuccstar[12]    16    33
```

```
## Nsuccstar[13]    12    28
## Nsuccstar[14]     8    23
## Nsuccstar[15]     7    20
## Nsuccstar[16]     4    16
## Nsuccstar[17]     3    14
## Nsuccstar[18]     2    11
## Nsuccstar[19]     1     9
## Nsuccstar[20]     0     4
## attr(,"Probability")
## [1] 0.95
```

```r
plot(putts1$dist, 100*putts1$Nsucc/putts1$Ntrys, ylim=c(0,100),
     ylab="Success/100", xlab="Putt Distance")
lines(putts1$dist, colMeans(putting1.samps[[1]]))

lines(putts1$dist, HDR[,1], lty=2)
lines(putts1$dist, HDR[,2], lty=2)
```



c

The 95% HDR interval for successes at 25ft is between 0 and 4. Based on the plot, the model may be predicting too low of a success rate at 25ft. From 15ft-20ft the decline in success rate is slower than the decline from 0-10ft. From watching and playing golf I think the difference in success rate between 20 and

25 feet is not that big. I would not be 95% confident that a pro golfers success rate out of 100 from 25ft is below 4.

```r
#Q2.
# a
# see model in Appendix 3
putting2.model=jags.model("putt2.model", putts1)
```

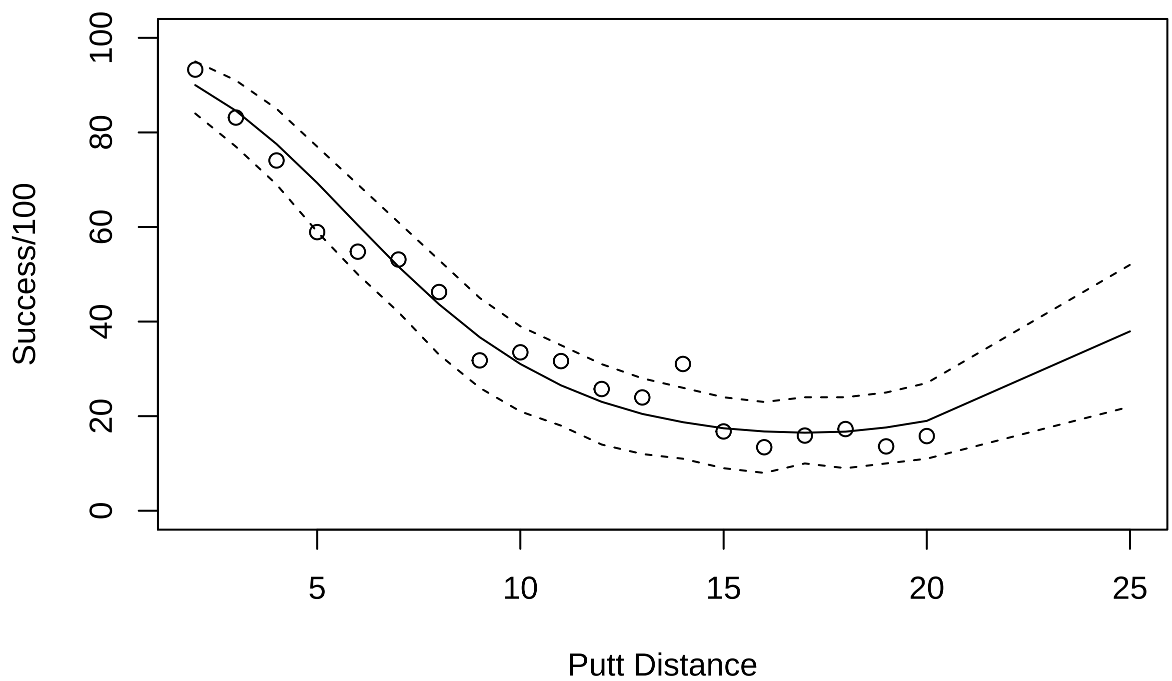
```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 19
##    Unobserved stochastic nodes: 24
##    Total graph size: 166
##
## Initializing model
```

```r
putting2.samps=coda.samples(putting2.model,
                            variable.names = c("Nsuccstar"), n.iter=1e4)
HDR2=HPDinterval(putting2.samps[[1]])
HDR2
```

```
##               lower upper
## Nsuccstar[1]     84    95
## Nsuccstar[2]     77    91
## Nsuccstar[3]     69    85
## Nsuccstar[4]     59    77
## Nsuccstar[5]     50    69
## Nsuccstar[6]     42    61
## Nsuccstar[7]     33    53
## Nsuccstar[8]     26    45
## Nsuccstar[9]     21    39
## Nsuccstar[10]    18    35
## Nsuccstar[11]    14    31
## Nsuccstar[12]    12    28
## Nsuccstar[13]    11    26
## Nsuccstar[14]     9    24
## Nsuccstar[15]     8    23
## Nsuccstar[16]    10    24
## Nsuccstar[17]     9    24
## Nsuccstar[18]    10    25
## Nsuccstar[19]    11    27
## Nsuccstar[20]    22    52
## attr(,"Probability")
## [1] 0.95
```

```r
# b
plot(putts1$dist, 100*putts1$Nsucc/putts1$Ntrys, ylim=c(0,100),
     ylab="Success/100", xlab="Putt Distance")
lines(putts1$dist, colMeans(putting2.samps[[1]]))

lines(putts1$dist, HDR2[,1], lty=2)
lines(putts1$dist, HDR2[,2], lty=2)
```



This model does not lead to believable predictions at 25ft. The HDR at 25ft is between 23 and 55 out of 100 putts. The plot does not make sense as there is a greater success rate for 25ft putts than predicted at distances between 15 and 20ft. This should not be the case as putts are more difficult as the length increases. The quadratic component of the probability function is not accurate.

```
#Q3.
# a
# see model in Appendix 4
putting3.model=jags.model("putt3.model", putts1)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 19
##    Unobserved stochastic nodes: 23
##    Total graph size: 165
##
## Initializing model
```

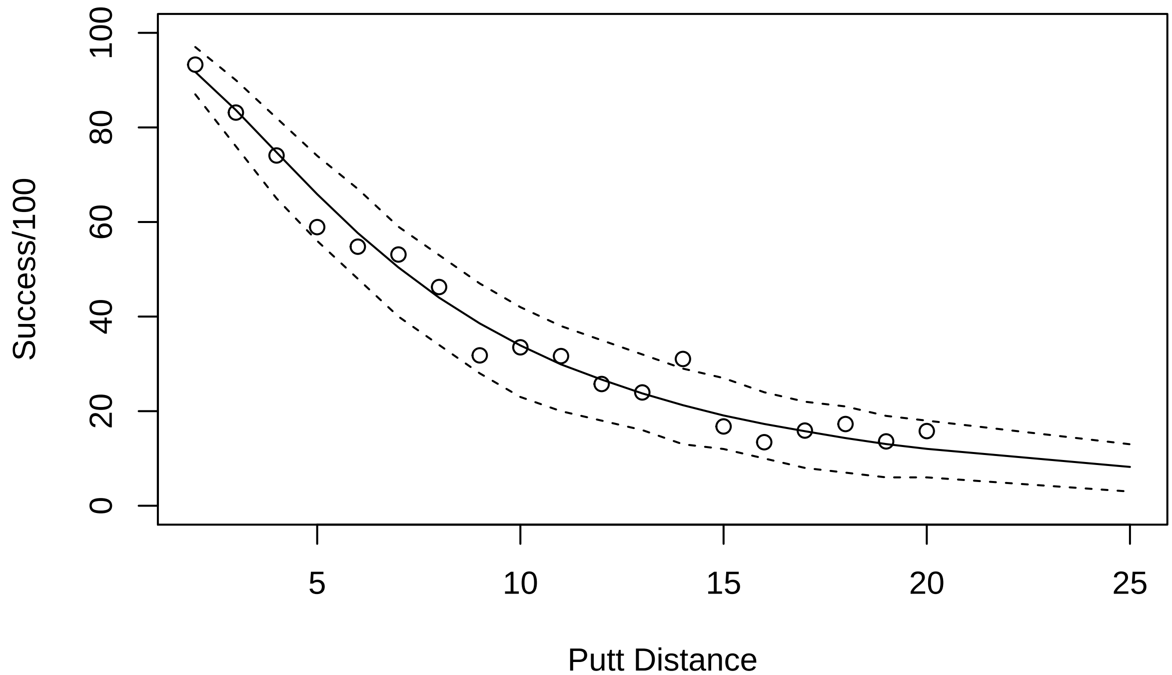
```
putting3.samps=coda.samples(putting3.model,
                            variable.names = c("Nsuccstar"), n.iter=1e4)
HDR3=HPDinterval(putting3.samps[[1]])
HDR3
```

```
##                 lower upper
```

```
## Nsuccstar[1]     87    97
## Nsuccstar[2]     76    90
## Nsuccstar[3]     65    82
## Nsuccstar[4]     56    74
## Nsuccstar[5]     48    67
## Nsuccstar[6]     40    59
## Nsuccstar[7]     34    53
## Nsuccstar[8]     28    47
## Nsuccstar[9]     23    42
## Nsuccstar[10]    20    38
## Nsuccstar[11]    18    35
## Nsuccstar[12]    16    32
## Nsuccstar[13]    13    29
## Nsuccstar[14]    12    27
## Nsuccstar[15]    10    24
## Nsuccstar[16]     8    22
## Nsuccstar[17]     7    21
## Nsuccstar[18]     6    19
## Nsuccstar[19]     6    18
## Nsuccstar[20]     3    13
## attr(,"Probability")
## [1] 0.95
```

```
#b
plot(putts1$dist, 100*putts1$Nsucc/putts1$Ntrys, ylim=c(0,100),
     ylab="Success/100", xlab="Putt Distance")
lines(putts1$dist, colMeans(putting3.samps[[1]]))

lines(putts1$dist, HDR3[,1], lty=2)
lines(putts1$dist, HDR3[,2], lty=2)
```



The HDR intervals at shorter putts are more realistic. In the 2 previous model the HDR intervals were too low for short putts. The HDR at 25ft is slightly lower than the success rate at 20ft. I believe this is an accurate representation of pro golfers putting success rate at that distance. The quadratic model was wrong as it showed an increase in success rate after a certain distance which is not accurate. The linear model decreased too fast as the putt distance went beyond 20ft. When we use the log of distance the model follows the same pattern as the plotted data. Based on the slope of the plotted data-points on 15-20ft putts and my own expectation of putting success rates at 25ft I believe the log model is the best predictor at that distance.

APPENDIX

1

```
model {
 # likelihood
 for (i in 1:length(Nsucc)) {
   Nsucc[i] ~ dbin(p[i], Ntrys[i])
   logit(p[i]) = alpha + beta*dist[i]
 }
 # prior
 alpha ~ dunif(-10,10)
 beta ~ dunif(-10,10)
}
```

2

```
model {
 # likelihood
 for (i in 1:length(Nsucc)) {
   Nsucc[i] ~ dbin(p[i], Ntrys[i])
   logit(p[i]) = alpha + beta*dist[i]
   Nsuccstar[i] ~ dbin(p[i], 100)
 }
 # prior
 alpha ~ dunif(-10,10)
 beta ~ dunif(-10,10)
}
```

3

```
model {
 # likelihood
 for (i in 1:length(Nsucc)) {
   Nsucc[i] ~ dbin(p[i], Ntrys[i])
   logit(p[i]) = alpha + beta*dist[i] + lambda*dist[i]*dist[i]
   Nsuccstar[i] ~ dbin(p[i], 100)
 }
 # prior
 alpha ~ dunif(-10,10)
 beta ~ dunif(-10,10)
 lambda ~ dunif(-10,10)
}
```

4

```
model {
 # likelihood
 for (i in 1:length(Nsucc)) {
   Nsucc[i] ~ dbin(p[i], Ntrys[i])
   logit(p[i]) = alpha + beta*log(dist[i])
   Nsuccstar[i] ~ dbin(p[i], 100)
 }
 # prior
 alpha ~ dunif(-10,10)
 beta ~ dunif(-10,10)
}
```