STATISTICAL RETHINKING WINTER 2019
HOMEWORK, WEEK 4 SOLUTIONS

1. To compute the entropies, we just need a function to compute the entropy. Information entropy, as defined in lecture and the book, is simply:

$$H(p) = -\sum_i p_i \log(p_i)$$

where p is a vector of probabilities summing to 1. In R code this would look like:

```
H <- function(p) -sum(p*log(p))
```

I'll make a list of the birb distributions and then push each through the function above.

```
IB <- list()
IB[[1]] <- c( 0.2 , 0.2 , 0.2 , 0.2 , 0.2 )
IB[[2]] <- c( 0.8 , 0.1 , 0.05 , 0.025 , 0.025 )
IB[[3]] <- c( 0.05 , 0.15 , 0.7 , 0.05 , 0.05 )
sapply( IB , H )
```

```
[1] 1.6094379 0.7430039 0.9836003
```

The first island has the largest entropy, followed by the third, and then the second in last place. Why is this? Entropy is a measure of the evenness of a distribution. The first islands has the most even distribution of birbs. This means you wouldn't be very surprised by any particular birb. The second island, in contrast, has a very uneven distribution of birbs. If you saw any birb other than the first species, it would be surprising.

Now we need K-L distance, so let's write a function for it:

```
DKL <- function(p,q) sum( p*(log(p)-log(q)) )
```

This is the distance from q to p, regarding p as true and q as the model. Now to use each island as a model of the others, we need to consider the different ordered pairings. I'll just make a matrix and loop over rows and columns:

```
Dm <- matrix( NA , nrow=3 , ncol=3 )
for ( i in 1:3 ) for ( j in 1:3 ) Dm[i,j] <- DKL( IB[[j]] , IB[[i]] )
round( Dm , 2 )
```

```
     [,1] [,2] [,3]
[1,] 0.00 0.87 0.63
```

```
[2,] 0.97 0.00 1.84
[3,] 0.64 2.01 0.00
```

The way to read this is each row as a model and each column as a true distribution. So the first island, the first row, has the smaller distances to the other islands. This makes sense, since it has the highest entropy. Why does that give it a shorter distance to the other islands? Because it is less surprised by the other islands, due to its high entropy.

2. I won't repeat the models here. They are in the text. Model `m6.9` contains both marriage status and age. Model `m6.10` contains only age. Model `m6.9` produces a confounded inference about the relationship between age and happiness, due to opening a collider path. To compare these models using WAIC:

```
compare( m6.9 , m6.10 )
```

```
        WAIC pWAIC dWAIC weight    SE  dSE
m6.9  2714.0   3.7   0.0      1 37.54   NA
m6.10 3101.9   2.3 387.9      0 27.74 35.4
```

The model that produces the invalid inference, `m6.9`, is expected to predict much better. And it would. This is because the collider path does convey actual association. We simply end up mistaken about the causal inference. We should not use WAIC (or LOO) to choose among models, unless we have some clear sense of the causal model. These criteria will happily favor confounded models.

3. These are the models:

```
library(rethinking)
data(foxes)
d <- foxes
d$W <- standardize(d$weight)
d$A <- standardize(d$area)
d$F <- standardize(d$avgfood)
d$G <- standardize(d$groupsize)

m1 <- quap(
    alist(
        W ~ dnorm( mu , sigma ),
        mu <- a + bF*F + bG*G + bA*A,
        a ~ dnorm(0,0.2),
        c(bF,bG,bA) ~ dnorm(0,0.5),
        sigma ~ dexp(1)
    ), data=d )
m2 <- quap(
    alist(
        W ~ dnorm( mu , sigma ),
```

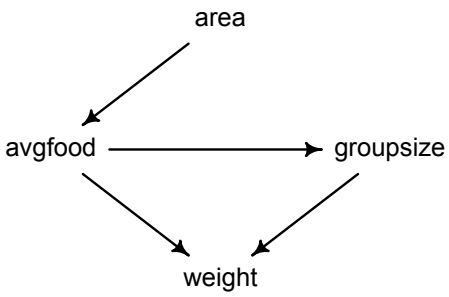
```
        mu <- a + bF*F + bG*G,
        a ~ dnorm(0,0.2),
        c(bF,bG) ~ dnorm(0,0.5),
        sigma ~ dexp(1)
    ), data=d )
m3 <- quap(
    alist(
        W ~ dnorm( mu , sigma ),
        mu <- a + bG*G + bA*A,
        a ~ dnorm(0,0.2),
        c(bG,bA) ~ dnorm(0,0.5),
        sigma ~ dexp(1)
    ), data=d )
m4 <- quap(
    alist(
        W ~ dnorm( mu , sigma ),
        mu <- a + bF*F,
        a ~ dnorm(0,0.2),
        bF ~ dnorm(0,0.5),
        sigma ~ dexp(1)
    ), data=d )
m5 <- quap(
    alist(
        W ~ dnorm( mu , sigma ),
        mu <- a + bA*A,
        a ~ dnorm(0,0.2),
        bA ~ dnorm(0,0.5),
        sigma ~ dexp(1)
    ), data=d )
```

Comparing with WAIC:

```
compare( m1 , m2 , m3 , m4 , m5 )
```

```
    WAIC pWAIC dWAIC weight    SE  dSE
m1 322.9   4.7   0.0   0.47 16.28   NA
m3 323.9   3.7   1.0   0.28 15.68 2.90
m2 324.1   3.9   1.2   0.25 16.14 3.60
m4 333.4   2.4  10.6   0.00 13.79 7.19
m5 333.7   2.7  10.8   0.00 13.79 7.24
```

To remind you, the DAG from last week is:



Notice that the top three models are `m1`, `m3`, and `m2`. They have very similar WAIC values. The differences are small and smaller in all cases than the standard error of the difference. WAIC sees these models are tied. This makes sense, given the DAG, because as long as a model has `groupsize` in it, we can include either `avgfood` or `area` or both and get the same inferences. Another way to think of this is that the influence of good, adjusting for group size, is (according to the DAG) the same as the influence of area, adjusting for group size, because the influence of area is routed entirely through food and group size. There are no backdoor paths.

What about the other two models, `m4` and `m5`? These models are tied with one another, and both omit group size. Again, the influence of area passes entirely through food. So including only food or only area should produce the same inference—the total causal influence of area (or food) is just about zero. That's indeed what the posterior distributions suggest:

```
coeftab(m4,m5)
```

```
      m4      m5
a          0       0
bF     -0.02      NA
sigma   0.99    0.99
bA        NA    0.02
nobs     116     116
```