Bayesian Chap 7 Assignment

Meng Yuan

Question 1

Information entropy is defined as $H(p) = -\sum_{i=1}^{n} p_i \log(p_i)$, where p_i is the probability of the ith event, the probabilities of n events sum to 1. Here's the function to compute H(p) in R:

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

Here are the information entropies of three islands:

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

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

Island 1's birb distribution has the largest entropy, island 2 has the smallest. Information entropy measures the uncertainty of a distribution, the more even the distribution is, the more uncertainty it has, the larger entropy it has. Island 1 has the most even distribution, each probability of the birb is not surprising, hence it has the largest uncertainty. In contrast, island 2 has the most uneven distribution, it has the least uncertainty and smallest entropy.

Next, we can calculate the K-L divergence of each island from others, to see which island's birb distribution best predicts the other two. K-L divergence is the amount of additional uncertainty added by using probabilities from one distribution to describe another distribution.

Here's the function to compute K-L divergence in R:

```
DKL <- function(p,q) sum( p*(log(p)-log(q)) ) # distance from pq to p
```

Here are K-L distances in different ordered pairings:

```
D <- matrix( NA , nrow=3 , ncol=3 )
for ( i in 1:3 ) for ( j in 1:3 ) D[i,j] <- DKL( island[[j]] , island[[i]])
D
```

```
##           [,1]     [,2]      [,3]
## [1,] 0.0000000 0.866434 0.6258376
## [2,] 0.9704061 0.000000 1.8388452
## [3,] 0.6387604 2.010914 0.0000000
```

Each row is a model, and each column is a true distribution. Each value represents the K-L distance from the model to the true distribution. The K-L distance between a model and itself is zero. Island 1 has the smallest distances to other islands, as the first row shows.

Island 1 best predicts the other two islands, because it has the largest entropy, and it's less surprised by the probabilities of other islands.

Question 2

Here are model m6.9 and m6.10:

```
# m6.9
d2$mid <- d2$married + 1
m6.9 <- quap(
    alist(
        happiness ~ dnorm( mu , sigma ),
        mu <- a[mid] + bA*A,
        a[mid] ~ dnorm( 0 , 1 ),
        bA ~ dnorm( 0 , 2 ),
        sigma ~ dexp(1)
    ) , data=d2 )
precis(m6.9,depth=2)
```

```
##             mean         sd       5.5%      94.5%
## a[1]  -0.2350877 0.06348986 -0.3365568 -0.1336186
## a[2]   1.2585517 0.08495989  1.1227694  1.3943340
## bA    -0.7490274 0.11320112 -0.9299447 -0.5681102
## sigma  0.9897080 0.02255800  0.9536559  1.0257600
```

```
# m6.10
m6.10 <- quap(
    alist(
        happiness ~ dnorm( mu , sigma ),
        mu <- a + bA*A,
        a ~ dnorm( 0 , 1 ),
        bA ~ dnorm( 0 , 2 ),
        sigma ~ dexp(1)
    ) , data=d2 )
precis(m6.10)
```

```
##                mean         sd       5.5%     94.5%
## a      1.649248e-07 0.07675015 -0.1226614 0.1226617
## bA    -2.728620e-07 0.13225976 -0.2113769 0.2113764
## sigma  1.213188e+00 0.02766080  1.1689803 1.2573949
```

Model m6.9 includes age and an index variable marriage status, while model m6.10 only includes age as predictor. Model m6.9 suggests that age is negatively associated with happiness, while model m6.10 suggests no association between age and happiness.

In model m6.9, marriage status is a collider that was conditioned on, it leads to spurious

statistical association among the causes (age and marriage status) as well as erroneous causal inference.

Next, we can look at the predictive accuracies of both models through WAIC:

```
compare( m6.9 , m6.10 , func=WAIC )
```

```
##           WAIC       SE    dWAIC      dSE    pWAIC       weight
## m6.9  2713.971 37.54465   0.0000       NA 3.738532 1.000000e+00
## m6.10 3101.906 27.74379 387.9347 35.40032 2.340445 5.768312e-85
```

WAIC is an information criterion which approximates the out-of-sample K-L Divergence. It can be used to compare model predictive accuracies. Model 6.9 has better predictive accuracy (100% of weight) as the collider shows actual association. The model provides erroneous causal inference but has better prediction. This suggests causal inference and predictive accuracy should be considered separately, WAIC (or LOO) should not be used for causal inference.

Question 3

I plan to use three sets of priors for the intercept α and slope β:

1. Regularized priors: $\alpha \sim Normal(0, 0.2), \beta \sim Normal(0, 0.3)$
2. Weak priors: $\alpha \sim Normal(0, 1), \beta \sim Normal(0, 1)$
3. Strong priors: $\alpha \sim Normal(0, 0.1), \beta \sim Normal(0, 0.1)$

Here are the codes for models using `quap` with regularized priors for the slopes and intercepts, for example:

```
# F, G, A -> W
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 )
# F, G -> W
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 )
# G, A -> W
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 )
# F -> W
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 )
# A -> W
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 )
```

Model comparison

Here's model comparison using WAIC:

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

```
##        WAIC       SE     dWAIC      dSE    pWAIC      weight
## m1 322.8847 16.27783  0.000000       NA 4.656959 0.465363694
## m3 323.8985 15.68240  1.013749 2.899417 3.718565 0.280323674
## m2 324.1284 16.13964  1.243666 3.598475 3.859897 0.249881396
## m4 333.4444 13.78855 10.559695 7.193396 2.426279 0.002370193
## m5 333.7239 13.79447 10.839215 7.242069 2.650636 0.002061043
```

Model m1, m2 and m3 are the top three models, with smaller WAIC scores than the rest, and they share nearly all of the weight. But the differences in WAIC are small for all the models,

compared to the standard error. With regard to WAIC, the top three models are similar, they all have groupsize as a predictor. This could mean that when groupsize is included as a predictor, the inclusion of avgfood or area as predictors doesn't affect the inference. According to the DAG, the influence of area on weight in through avgfood and groupsize, this also explains why the first three models are tied together.

When groupsize is not included as a predictor, model m4 and m5 are tied. area and avgfood have no back door and they both have no influence on weight, according to the posterior predictions.

```
precis(m4)
```

```
##                 mean         sd       5.5%     94.5%
## a     -3.349017e-06 0.08360119 -0.1336142 0.1336075
## bF    -2.421183e-02 0.09088632 -0.1694657 0.1210421
## sigma  9.911586e-01 0.06466096  0.8878179 1.0944993
```

```
precis(m5)
```

```
##                 mean         sd       5.5%     94.5%
## a     -9.364016e-08 0.08360866 -0.1336229 0.1336227
## bA     1.883354e-02 0.09089581 -0.1264355 0.1641026
## sigma  9.912660e-01 0.06466647  0.8879165 1.0946155
```

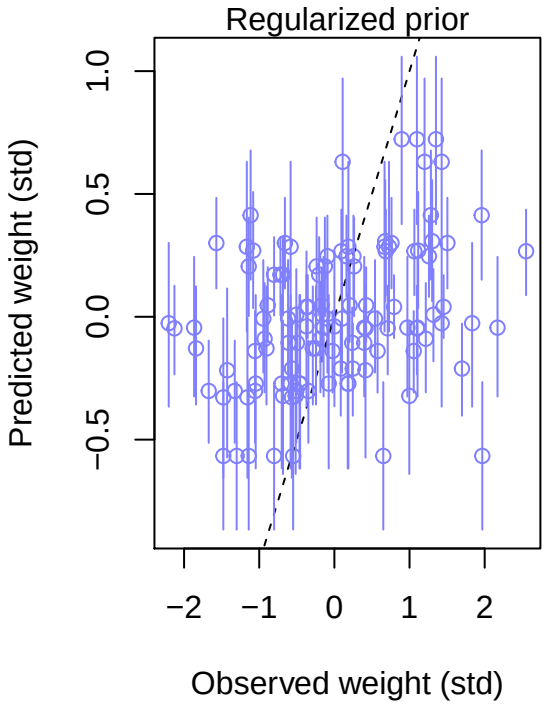
Prior choice

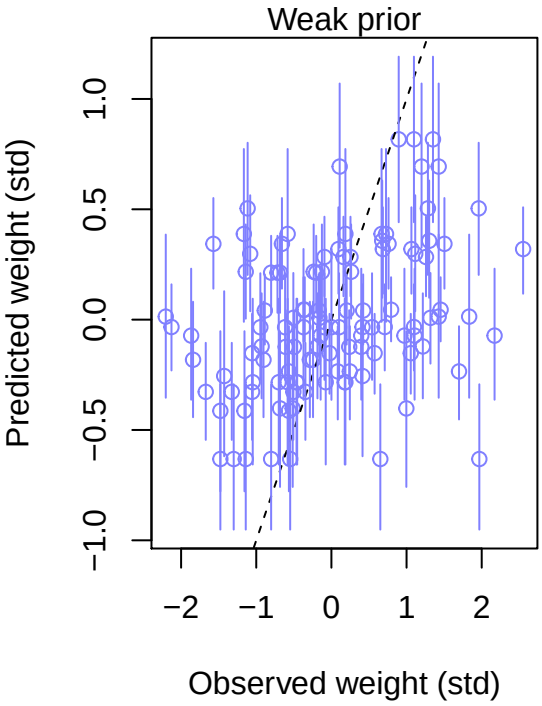
Using model m1 as an example, here's the comparison between different priors.

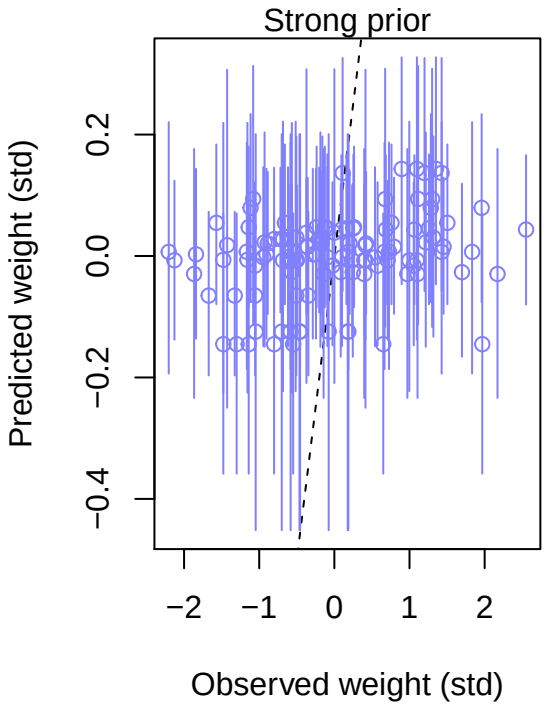
```
# weak priors
m1.1 <- quap(
    alist(
        W ~ dnorm( mu , sigma ),
        mu <- a + bF*F + bG*G + bA*A,
        a ~ dnorm(0,1),
        c(bF,bG,bA) ~ dnorm(0,1),
        sigma ~ dexp(1)
    ), data=d )

# strong priors
m1.2 <- quap(
    alist(
        W ~ dnorm( mu , sigma ),
        mu <- a + bF*F + bG*G + bA*A,
        a ~ dnorm(0,0.1),
        c(bF,bG,bA) ~ dnorm(0,0.1),
        sigma ~ dexp(1)
```

```
    ), data=d )
```

Based on the posterior predictive checks, regularized and weak priors provide better prediction than the strong prior. Though three models have poor predictions on some extreme values.







WAIC supports that models m1 (regularized prior) and m1.1 (weak prior) are better than model m1.2 (strong prior). The model using regularized prior has the best prediction.

```
compare(m1 , m1.1 , m1.2)
```

```
##          WAIC       SE     dWAIC      dSE    pWAIC    weight
## m1   323.3045 16.29568 0.0000000       NA 4.834002 0.5950673
## m1.1 324.2509 16.83877 0.9464168 1.179126 5.609542 0.3707271
## m1.2 329.0171 14.19136 5.7125704 5.823590 2.385523 0.0342056
```

Model m1 and m1.1 also have close posterior means, with contrast to model m1.2.

```
coeftab(m1 , m1.1 , m1.2)
```

```
##       m1      m1.1    m1.2
## a          0       0       0
## bF      0.30    0.38    0.02
## bG     -0.64   -0.75   -0.12
## bA      0.28    0.30    0.06
## sigma   0.93    0.93    0.97
## nobs     116     116     116
```

These results show that regularized priors can let the model learn from the data while reducing the risk of overfitting, and provide better prediction. In this case, model m1 uses regularized priors and has the best prediction. Model m1.1 using vague priors does not work a lot worse, meaning there're no issues with overfitting. Model m1.2 using strong priors suffers from underfitting and provides the least accurate prediction.