Chapter 7 Solutions

Easy.

1. State the three motivating criteria that define information entropy.

a) continuous – a small change in probabilities should not precipitate a massive (step-wise) change in uncertainty

b) “increasing” – when more events are possible there is inherently more uncertainty in the system

c) additive – different ways of combining events in a system should all add up to the same thing

2. Suppose a coin is weighted such that, when it is tossed and lands on a table, it comes up heads 70% of the time. What is the entropy

```r
p <- c(.7, .3)
-sum(p*log(p))
```

```
## [1] 0.6108643
```

3. Suppose a four-sided die is loaded such that, when tossed onto a table, it shows ‘1’ 20%, ‘2’ 25%, ‘3’ 25%, and ‘4’ 30% of the time. What is the entropy of this die?

```r
p <- c(.2, .25, .25, .3)
-sum(p*log(p))
```

```
## [1] 1.376227
```

4. Suppose another four-sided die is loaded such that it never shows ‘4’. The other three sides show equally often. What is the entropy of this die?

```r
p <- c(.33, .33, .33)
-sum(p*log(p))
```

```
## [1] 1.097576
```

Medium

1. Write down and compare the definitions of AIC and WAIC. Which of these criteria is most general? Which assumptions are required to transform the more general criterion into a less general one?

AIC = D_train + 2p = -2lppd + 2p WAIC(y,o) = -2(lppd - sum(var(o)* log(p(y|o))))

WAIC is most general because it makes no assumption about the shape of the posterior. To transform WAIC into AIC, you would need the following two assumptions: * Priors are flat or overwhelmed by the likelihood * The posterior distribution is approximately multivariate Gaussian

2. Model selection vs model comparison.

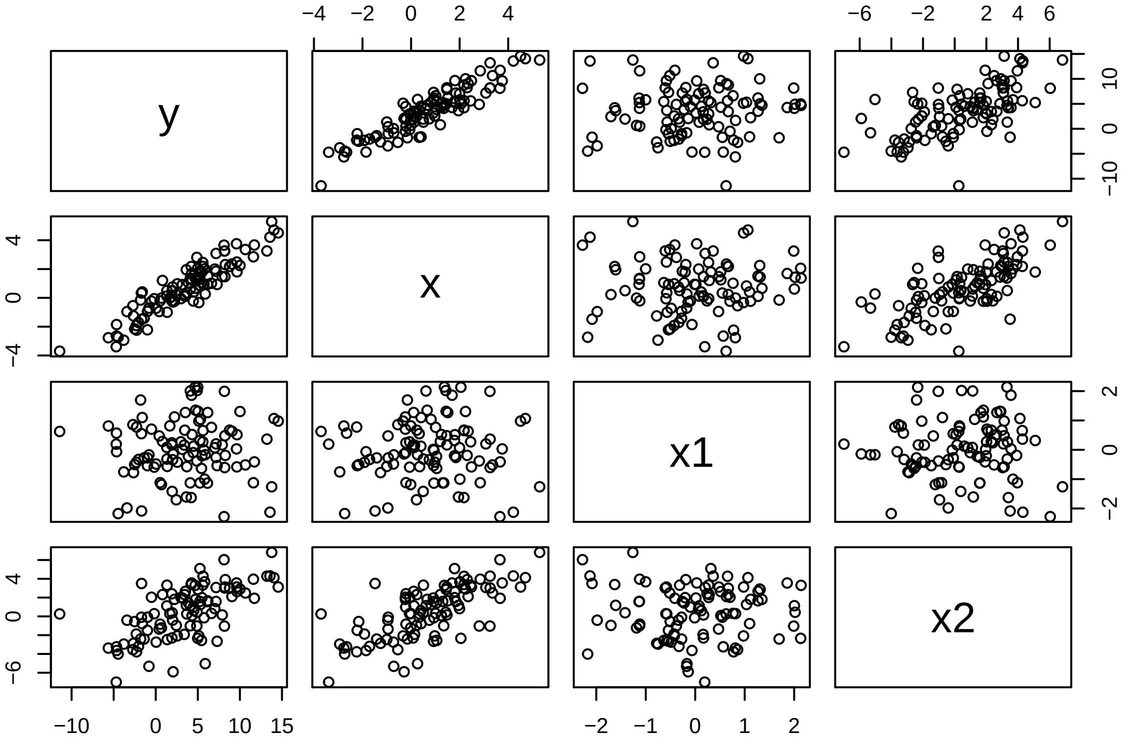
Model comparison evaluates different models across certain criteria. Model selection chooses the best performing model and discards the rest. Under model selection, you lose the information from all of the other models.

3. When comparing models with an information criterion, why must all models be fit to exactly the same observations? What would happen to the information criterion values, if the models were fit to different numbers of observations? Perform some experiments, if you are not sure.

Since information criteria is increasing, then one model could be fit to less data (and therefore less uncertainty), or one model could "get lucky" and be fit to "easier data" which would make it appear to be the best performing.

4. What happens to the effective number of parameters, as measured by PSIS or WAIC, as a prior becomes more concentrated? Why?

```r
x <-  rnorm(100, mean=1, sd=2)
x1 <- rnorm(100, mean=0, sd=1)              # not associated with outcome
x2 <- rnorm(100, mean=x, sd=2)              # spurious assocation
y <- rnorm(100, mean = 2 + 2.4*x, sd=2)
d <- data.frame( y=y, x=x, x1=x1, x2=x2)
pairs(d)
```



```r
library(rethinking)
```

```
## Loading required package: rstan

## Loading required package: StanHeaders

## Loading required package: ggplot2
## rstan (Version 2.21.2, GitRev: 2e1f913d3ca3)

## For execution on a local, multicore CPU with excess RAM we recommend calling
## options(mc.cores = parallel::detectCores()).
## To avoid recompilation of unchanged Stan programs, we recommend calling
## rstan_options(auto_write = TRUE)

## Loading required package: parallel

## rethinking (Version 2.13)

##
## Attaching package: 'rethinking'

## The following object is masked from 'package:stats':
##
##     rstudent
```

```
wide_prior <- alist(y ~ dnorm(mu, sigma),
                    mu ~ a + b*x + c*x + e*x,
                    c(b,c,e) ~ dnorm(0,10),
                    a ~ dnorm(0,5),
                    sigma ~ dunif(0,20))

tight_prior <- alist(y ~ dnorm(mu, sigma),
                    mu ~ a + b*x + c*x + e*x,
                    c(b,c,e) ~ dnorm(0,4),
                    a ~ dnorm(0,2),
                    sigma ~ dunif(0,2))

wide_mod <- quap(wide_prior, d)
tight_mod <- quap(tight_prior, d)
```

```
library(rethinking)
WAIC(wide_mod)
```

```
##       WAIC      lppd  penalty std_err
## 1 419.6017 -206.8776 2.923299 13.2106
```

```
WAIC(tight_mod)
```

```
##       WAIC      lppd  penalty  std_err
## 1 420.0351 -206.8966 3.120913 13.35139
```

As priors become more concentrated…? i think WAIC should decrease but I should come back to this one. Having a hard time showing it

5. Provide an informal explanation of why informative priors reduce overfitting.

Informative priors are a kind of regularization that prevents the model from "going to far." They can only reach within the bounds of the informative priors so they won't be able to fit the data as closely as a prior that allows the model a "longer leash"

6. Provide an informal explanation of why overly informative priors result in underfitting.

If the leash is too short though, then the opposite problem can occur in which case your model won't be able to discover anything about the parameters.

Hard

1. I want you to actually fit a curve to these data, found in data(Laffer). Consider models that use tax rate to predict tax revenue. Compare, using WAIC or PSIS, a straight-line model to any curved models you like. What do you conclude about the relationship between tax rate and tax revenue?

```
data(Laffer)
d <- Laffer
head(d)
```

```
##   tax_rate tax_revenue
## 1     0.07       -0.06
## 2     8.81        2.45
## 3    12.84        3.58
## 4    16.24        2.19
## 5    19.18        2.46
## 6    19.29        1.95
```

Fit a straight line model first.

```
flist <- alist(tax_revenue ~ dnorm(mu, sigma),
               mu <- a + b*tax_rate,
               a ~ dnorm(-5, 5),
               b ~ dnorm(1, .3),
               sigma ~ dunif(0,5))

straight_line_model <- quap(flist, d)
precis(straight_line_model)
```

```
##             mean         sd       5.5%     94.5%
## a     0.94758963 1.00439795 -0.6576323 2.5528115
## b     0.08850145 0.03625178  0.0305641 0.1464388
## sigma 1.70206101 0.22623705  1.3404905 2.0636315
```

```
WAIC(straight_line_model)
```

```
##       WAIC      lppd  penalty  std_err
## 1 126.5264 -55.97502 7.288174 23.75249
```

Fit curved model.

```
flist <- alist(tax_revenue ~ dnorm(mu, sigma),
               mu <- a + b*tax_rate + b2* tax_rate^2,
               a ~ dnorm(-5, 5),
               b ~ dnorm(1, .3),
               b2 ~ dnorm(0,2),
               sigma ~ dunif(0,5))

curved_model <- quap(flist, d)
precis(curved_model)
```

```
##               mean          sd       5.5%       94.5%
## a     -1.705242234 1.342804487 -3.8513032  0.440818686
## b      0.445746011 0.127772398  0.2415410  0.649950982
## b2    -0.008779857 0.002992279 -0.0135621 -0.003997618
## sigma  1.639121743 0.221725723  1.2847612  1.993482272
```

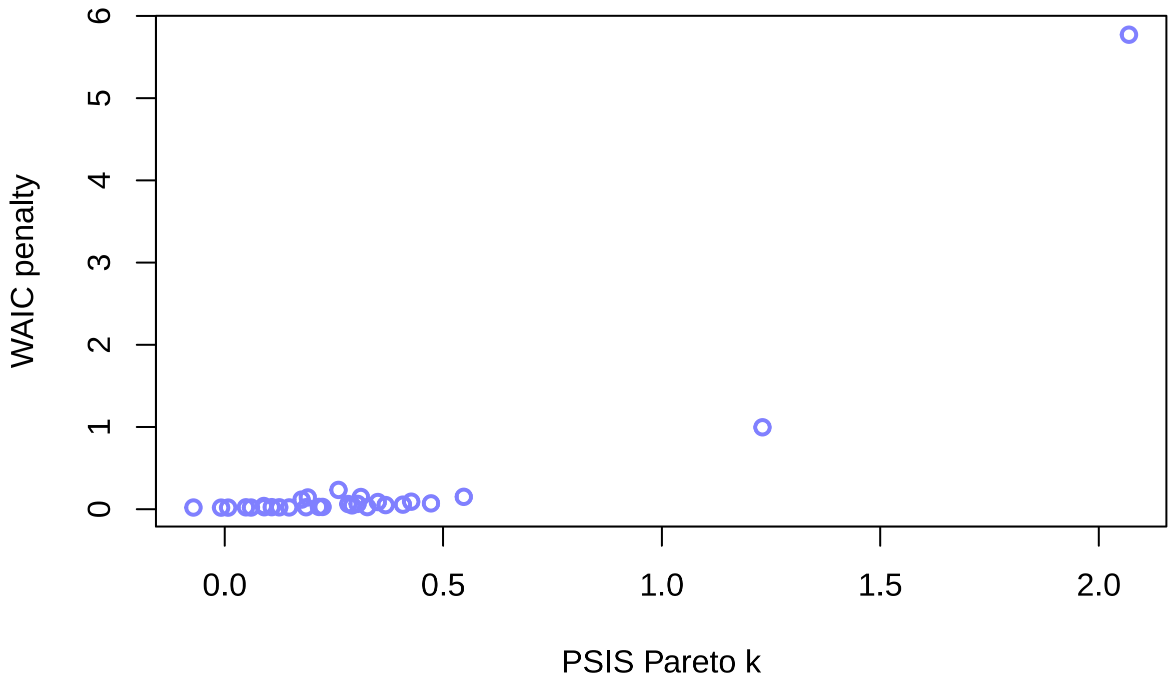
```
WAIC(curved_model)
```

```
##       WAIC      lppd  penalty std_err
## 1 125.8586 -54.95509 7.974209 22.3503
```

```
set.seed(24071847)
PSIS_cm <- PSIS(curved_model,pointwise=TRUE)
```

```
## Some Pareto k values are very high (>1). Set pointwise=TRUE to inspect individual points.
```

```
set.seed(24071847)
WAIC_cm <- WAIC(curved_model,pointwise=TRUE)
plot( PSIS_cm$k , WAIC_cm$penalty , xlab="PSIS Pareto k" ,
ylab="WAIC penalty" , col=rangi2 , lwd=2 )
```



2. In the Laffer data, there is one country with a high tax revenue that is an outlier. Use PSIS and WAIC to measure the importance of this outlier in the models you fit in the previous problem. Then use robust regression with a Student's t distribution to revisit the curve fitting problem. How much does a curved relationship depend upon the outlier point?

```
flist <- alist(tax_revenue ~ dstudent(2, mu, sigma),
               mu <- a + b*tax_rate + b2* tax_rate^2,
               a ~ dnorm(-5, 5),
               b ~ dnorm(1, .3),
               b2 ~ dnorm(0,2),
               sigma ~ dunif(0,5))

curved_model2 <- quap(flist, d)
precis(curved_model2)
```

```
##                mean          sd         5.5%        94.5%
## a     -1.804316052 1.355691693 -3.970973216  0.362341111
## b      0.397230356 0.058665070  0.303472243  0.490988469
## b2    -0.007516542 0.001070057 -0.009226701 -0.005806384
## sigma  1.002606997 0.250131403  0.602848705  1.402365288
```

```
PSIS(curved_model2)
```

```
## Some Pareto k values are high (>0.5). Set pointwise=TRUE to inspect individual points.
```

```
##      PSIS     lppd  penalty  std_err
## 1 114.775 -57.3875 6.100749 11.41713
```

7H3. First, compute the entropy of each island's bird distribution. Interpret these entropy values.

```
isl1 <- c(.2, .2, .2, .2, .2)
isl2 <- c(.8, .1, .05, .025, .025)
isl3 <- c(.05, .15, .7, .05, .05)

e1 <- -sum(isl1*log(isl1))
e2 <- -sum(isl2*log(isl2))
e3 <- -sum(isl3*log(isl3))
```

```
e1
```

```
## [1] 1.609438
```

```
e2
```

```
## [1] 0.7430039
```

```
e3
```

```
## [1] 0.9836003
```

Island 1 has the most entropy because you're the most uncertain about what birds you will find there. Island 2 has the least entropy because your least uncertain about what birds you'll find there (mostly Species A). Island 3 is in the middle.

Use each island's bird distribution to predict the other two. This means compute the K-L Divergence of each island from the others, treating each island as if it were a statistical model of the other islands. You should end up with 6 different K-L Divergence values. Which island predicts the others best. Why?

Recall that divergence is the additional uncertainty induced by using probabilities from one distribution to describe another distribution.

```
kl_1_from2 <- sum(isl1*(log(isl1/isl2)))
kl_1_from3 <- sum(isl1*(log(isl1/isl3)))

kl_2from1 <- sum(isl2*(log(isl2/isl1)))
kl_2from3 <- sum(isl2*(log(isl2/isl3)))

kl_3from1 <- sum(isl3*(log(isl3/isl1)))
kl_3from2 <- sum(isl3*(log(isl3/isl2)))
```

```
island1 = kl_2from1 + kl_3from1
island2 = kl_1_from2 + kl_3from2
```

```
island3 = kl_1_from3 + kl_2from3

island1
```

```
## [1] 1.492272
```

```
island2
```

```
## [1] 2.809251
```

```
island3
```

```
## [1] 2.649675
```

Island 1 predicts the others best because the sum of KL divergence using 1 to predict is smallest.

4. Recall the marriage, age, and happiness collider bias example from Chapter 6. Run models m6.9 and m6.10 again. Compare these two models using WAIC (or LOO, they will produce identical results). Which model is expected to make better predictions? Which model provides the correct causal inference about the influence of age on happiness? Can you explain why the answers to these two questions disagree?

```
d <- sim_happiness( seed=1977 , N_years=1000 )
d2 <- d[ d$age>17 , ] # only adults
d2$A <- ( d2$age - 18 ) / ( 65 - 18 )
d2$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 <- 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
```

```r
WAIC(m6.9)
```

```
##       WAIC      lppd  penalty  std_err
## 1 2713.971 -1353.247 3.738532 37.54465
```

```r
WAIC(m6.10)
```

```
##       WAIC      lppd  penalty  std_err
## 1 3101.906 -1548.612 2.340445 27.74379
```

According to these results, m6.9 is expected to make better predictions. However, m6.10 provides the correct causal inference about the influence of age on happiness (ie, age does not influence happiness). The reason they disagree is because sometimes the best causal model doesn't produce the most accurate forecasts.

5. Revisit the urban fox data, data(foxes), from the previous chapter's practice problems. Use WAIC or PSIS based model comparison on five different models, each using weight as the outcome,and containing these sets of predictor variables:

(1) avgfood + groupsize + area
(2) avgfood + groupsize
(3) groupsize + area
(4) avgfood
(5) area Can you explain the relative differences in WAIC scores, using the fox DAG from last week's homework? Be sure to pay attention to the standard error of the score differences (dSE).

```r
data(foxes)
d <- foxes
```

```r
flist1 <- alist(weight ~ dnorm(mu, sigma),
           mu <- a + b*avgfood + c* groupsize + e*area,
           a ~ dnorm(0,2),
           b ~ dnorm(0,2),
           c ~ dnorm(0,2),
           e ~ dnorm(0,2),
           sigma ~ dunif(0,10))

m1 <- quap(flist1, d)


flist2 <- alist(weight ~ dnorm(mu, sigma),
           mu <- a + b*avgfood + c* groupsize,
           a ~ dnorm(0,2),
           b ~ dnorm(0,2),
           c ~ dnorm(0,2),
           sigma ~ dunif(0,10))

m2 <- quap(flist1, d)


flist3 <- alist(weight ~ dnorm(mu, sigma),
           mu <- a + c* groupsize + e*area,
           a ~ dnorm(0,2),
           c ~ dnorm(0,2),
           e ~ dnorm(0,2),
           sigma ~ dunif(0,10))

m3 <- quap(flist3, d)

flist4 <- alist(weight ~ dnorm(mu, sigma),
           mu <- a + b* avgfood,
           a ~ dnorm(0,2),
           b ~ dnorm(0,2),
           sigma ~ dunif(0,10))

m4 <- quap(flist4, d)

flist5 <- alist(weight ~ dnorm(mu, sigma),
           mu <- a + b* area,
           a ~ dnorm(0,2),
           b ~ dnorm(0,2),
           sigma ~ dunif(0,10))

m5 <- quap(flist5, d)
```

```
compare(m1, m2, m3, m4, m5, func = WAIC)
```

```
##        WAIC       SE      dWAIC       dSE    pWAIC      weight
## m1 362.1627 16.42412  0.0000000        NA 4.766350 0.455431017
## m2 362.7421 16.58007  0.5794105 0.2223571 5.114686 0.340882902
## m3 363.8090 16.09146  1.6463879 3.0363656 4.135692 0.199946593
## m4 372.9722 13.67702 10.8095230 8.1775247 2.466186 0.002047220
## m5 373.3530 13.64328 11.1903482 8.0913475 2.795676 0.001692269
```

M1 fits the data best because it makes use of all the data. M2 is close to M1 because after you know groupsize and avgfood, you don’t gain much from also knowing area. M3 lacks avg food M5 doesn’t have groupsize M4 doesn’t have area so it does worst