HW17

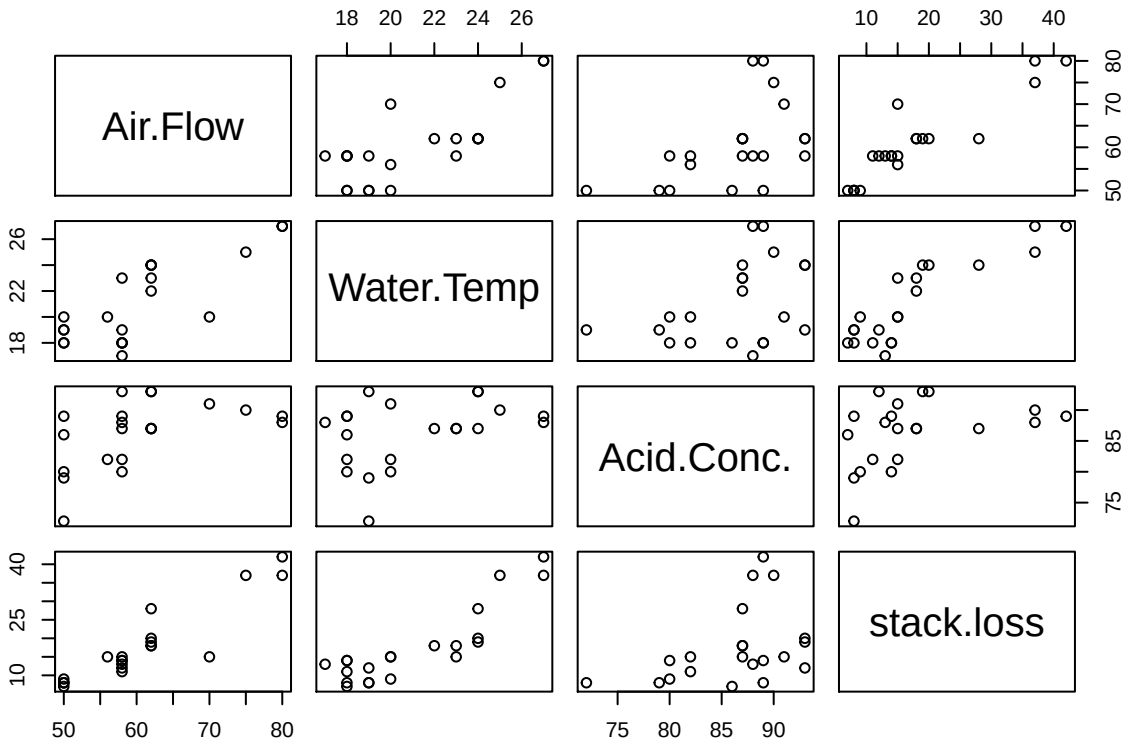
Zach White

November 18, 2016

Exercise 13.8

EDA/Pairs plot

```
pairs(stackloss)
```



Looking at these plots, it seems clear that the explantory variables: air flow, water temperature, and acid concentration are all linear related to stack loss. It seems like the relationship might not be exactly linear with stack loss. Also there is clearly some multicollinearity between the covariates, but none of them are extremely related, which is important. The following shows the correlation coefficients between the different variables.

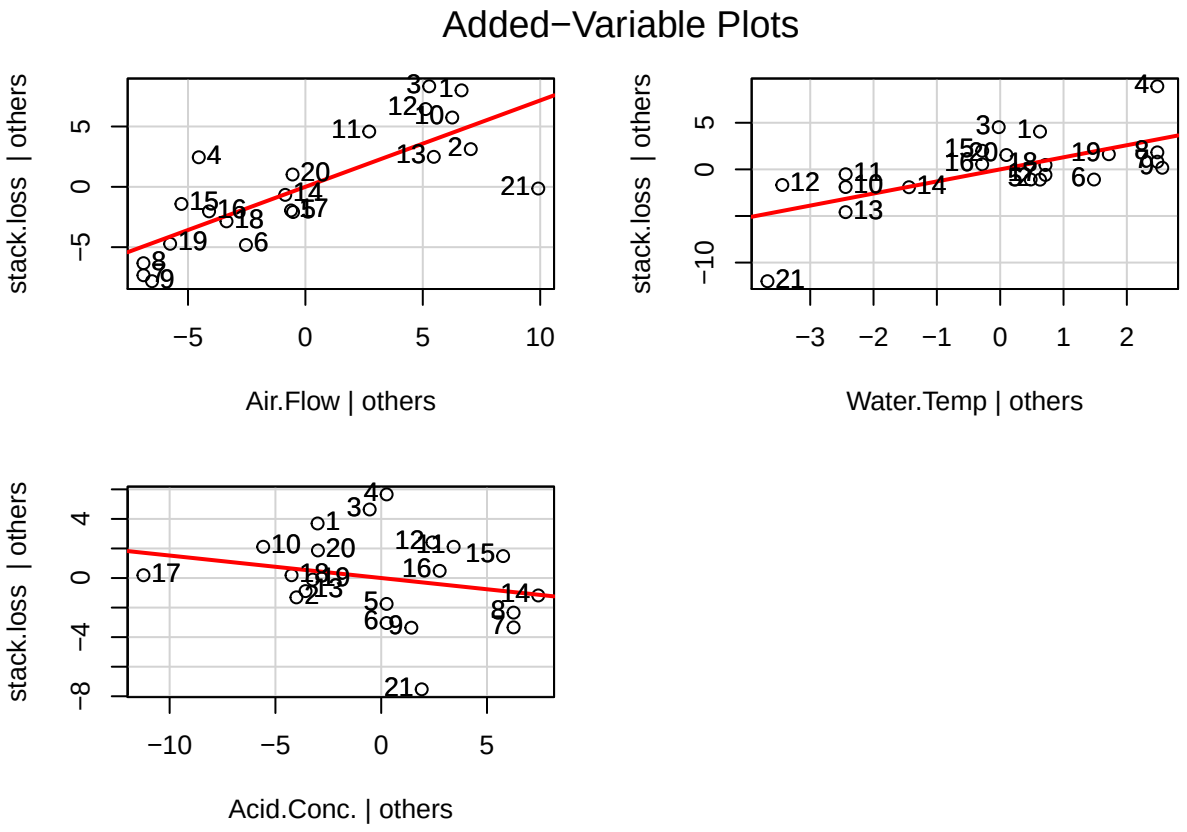
```
cor(stackloss)
```

```
##             Air.Flow Water.Temp Acid.Conc. stack.loss
## Air.Flow   1.0000000  0.7818523  0.5001429  0.9196635
## Water.Temp 0.7818523  1.0000000  0.3909395  0.8755044
## Acid.Conc. 0.5001429  0.3909395  1.0000000  0.3998296
## stack.loss 0.9196635  0.8755044  0.3998296  1.0000000
```

This confirms my initial thoughts. Water temperature seem to have a clear linear relationship with stack loss, while acid concentration doesn't seem as linearly related. Also, I note that water temperature and air flow have a correlation coefficient of .782, which is pretty high and suggests multicollinearity.

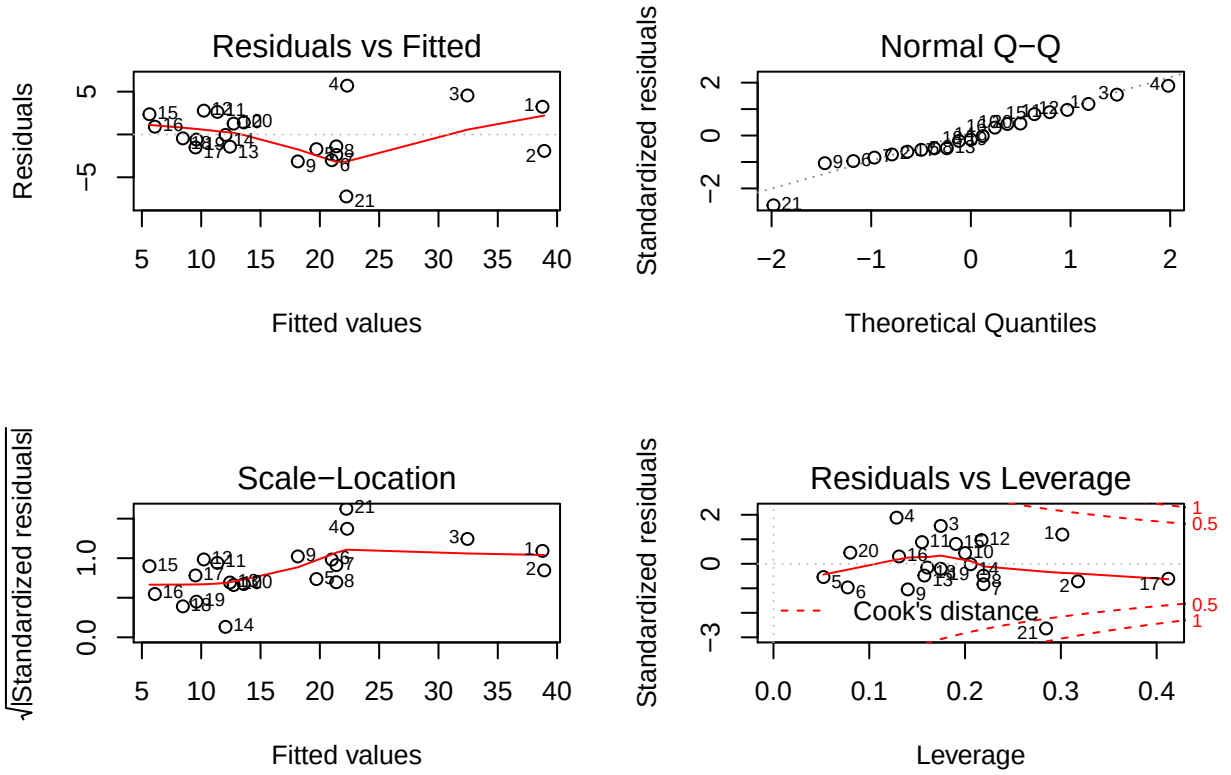
We will now fit a linear model and asses the fit.

```
stack.lm <- lm(stack.loss ~ ., data=stackloss)
avPlots(stack.lm, id.n = 21)
```



Looking at these plots, I can see a few potential outliers, specifically, it seems like there is one consistent outlier, observation 21. There are a few other potential outliers, observations 1, 3, and 4. In a lot of the plots, these observations are a little bit away from the fitted regression line. However, this is a pretty rudimentary analysis, and we will now explore this is a more refined way by calculating Cook's distance for this model.

```
par(mfrow = c(2,2))
plot(stack.lm,id.n = 21)
```

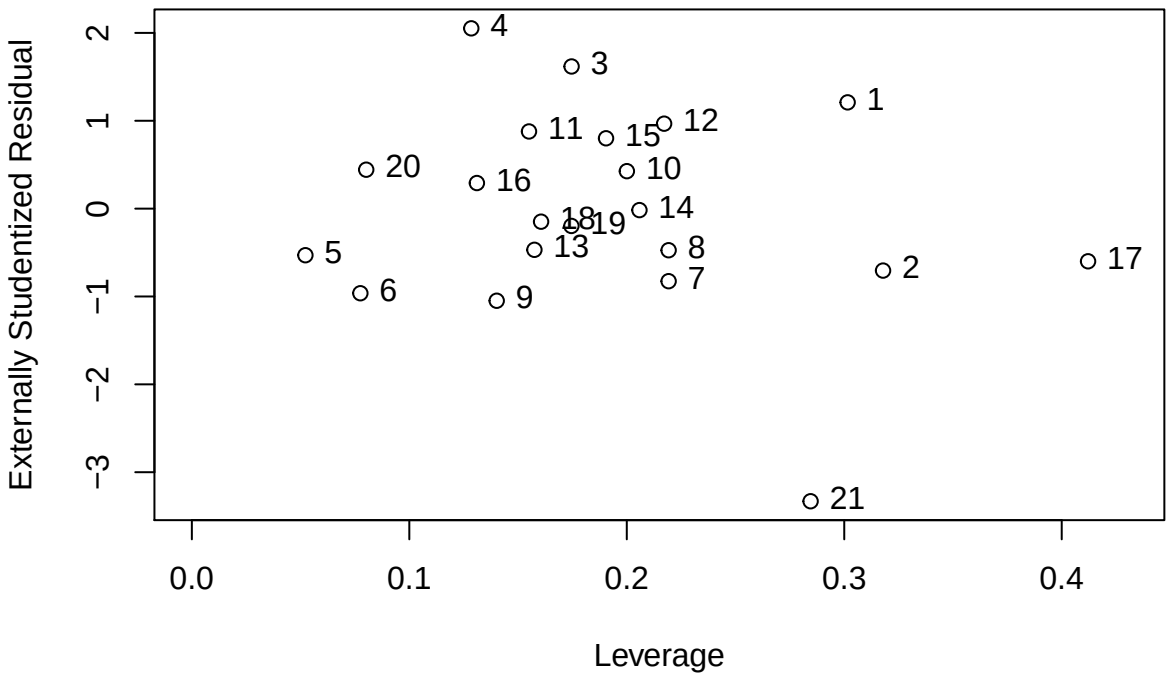


When we look at the lower right plot, observation 21 has a Cook's distance greater than .5, which doesn't guarantee it as an outlier, but it certainly seems like it is, considering the other plots, which shows that this observation could very well be an outlier. However, we don't get as clear of a story on the other potential outliers (1,3, and 4). Observation number 4, though, does seem like it could be an anomaly with residuls and fitted values, but also, the line seems like it could be pulled down due observation 21.

There are a few other techniques to detect influence and whether an observation is an oulier or not. Specifically, we will now analyze studentized residuals, and a Bayesian methodology where we will perform both variable selection and outlier detection through MCMC.

```
im = influence.measures(stack.lm)

plot(rstudent(stack.lm) ~ hatvalues(stack.lm), ylab="Externally Studentized Residual", xlab="Leverage",
with(stack.lm, text(rstudent(stack.lm)~hatvalues(stack.lm), labels = names(rstudent(stack.lm)), pos = 4
```




```
p.val = 2*(1- pt(max(abs(rstudent(stack.lm))), stack.lm$df - 1))
bon = .05/21
p.val < bon
```

```
## [1] FALSE
```

Here we use a Bonferroni correction because we will be making multiple comparisons. So we don't want to run the risk of diong multiple tests. So we will decrease the significance threshold. We can calcuate a p-value, and as we do so, we can see that this isn't siginificant. So based on this along, observation 21 isn't an oulier, which is odd because the other tests seem to indicate that it was or at least it could be.

We will now use a Bayesian model Averaging approach to test for outliers.

```
attach(stackloss)
```

```
## The following object is masked _by_ .GlobalEnv:
##
##     stack.loss
```

```
## The following object is masked from package:datasets:
##
##     stack.loss
```

```
stack.MC3= MC3.REG(stack.loss, stackloss[,-4]
  ,num.its=10000, outliers=TRUE, M0.out=rep(FALSE, 21), outs.list=1:21, M0.var=rep(TRUE, 3))

summary(stack.MC3)
```

```
##
```

```
## Call:
## MC3.REG(all.y = stack.loss, all.x = stackloss[, -4], num.its = 10000,     M0.var = rep(TRUE, 3), M0.
##
## Model parameters: PI = 0.1 K = 7 nu = 0.2 lambda = 0.1684 phi = 9.2
##
##   2034  models were selected
##  Best  5  models (cumulative posterior probability =  0.4478 ):
##
##               prob     model 1  model 2  model 3  model 4  model 5
## variables
##   Air.Flow    0.99999   x        x        x        x        x
##   Water.Temp  0.61153   x        .        x        x        x
##   Acid.Conc.  0.04981   .        .        .        .        .
## outliers
##   1           0.49298   x        .        .        x        .
##   2           0.06260   .        .        .        .        .
##   3           0.51466   x        .        .        x        .
##   4           0.90888   x        x        x        x        .
##   5           0.01725   .        .        .        .        .
##   6           0.02353   .        .        .        .        .
##   7           0.01833   .        .        .        .        .
##   8           0.01519   .        .        .        .        .
##   9           0.02138   .        .        .        .        .
##   10          0.01618   .        .        .        .        .
##   11          0.01681   .        .        .        .        .
##   12          0.01969   .        .        .        .        .
##   13          0.14333   .        .        .        x        .
##   14          0.05927   .        .        .        .        .
##   15          0.01981   .        .        .        .        .
##   16          0.01431   .        .        .        .        .
##   17          0.01687   .        .        .        .        .
##   18          0.01691   .        .        .        .        .
##   19          0.02402   .        .        .        .        .
##   20          0.04729   .        .        .        .        .
##   21          0.98493   x        x        x        x        x
##
## post prob              0.18504  0.13656  0.06933  0.03097  0.02594
```

```
detach(stackloss)
```

Under this approach, we find that the most likely model has a probability of .1857. This model contains two different variables air flow and water temperature. This model also classifies 4 observations as outliers: 1, 3, 4, and 21. The next most likely model has a probability of .137, and it only includes the variable air flow, and it has 2 different observations: 4 and 21. This two observations were the most likely outliers, with probabilities of .9847 and .9082, respectively. These observations seem likely to be outliers. The next two most likely observations to be outliers are 1 and 3, with probabilities of .493 and .51615, respectively.

As far as the variable selection is concerned, the most probability variables are air flow and water temperature. The probability that air flow is a part of the model is 1, which means it is included in the model. The probability water temperature is included is .612, which is also high. The acid concentration has a probability of .05 of being included in the model.

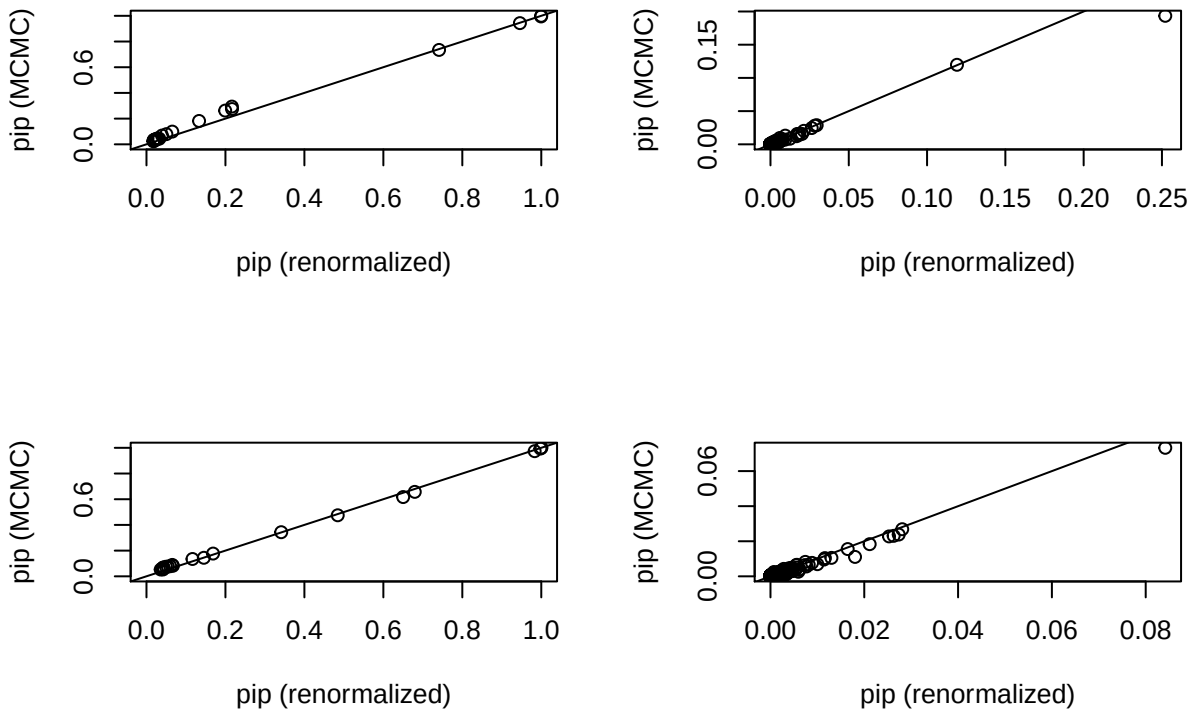
Although BAS isn't designed to detect outliers, we can change it so it can. We will propose it under two different prior distributions: truncated beta-binomial and truncated poisson.

```
n = nrow(stackloss)
stack.out = cbind(stackloss, diag(n))  #add indicators

BAS.stack.beta = bas.lm(stack.loss ~ .,
                    data=stack.out,
                    prior="hyper-g-n", a=3,
                    modelprior=tr.beta.binomial(1, 1, 15),
                    method="MCMC",
                    MCMC.iterations =50000)
BAS.stack.pois = bas.lm(stack.loss ~ .,
                    data=stack.out,
                    prior="hyper-g-n", a=3,
                    modelprior=tr.poisson(4, 15),
                    method="MCMC",
                    MCMC.iterations =50000)

par(mfrow = c(2,2))
diagnostics(BAS.stack.beta, type="pip")
diagnostics(BAS.stack.beta, type="model")

diagnostics(BAS.stack.pois,type = "pip")
diagnostics(BAS.stack.pois, type = "model")
```



I initially ran this for 10000, but the convergence wasn't ideal. I ran it for 50000, and the convergence seems better in this case.

```
BAS.stack.beta
```

```
##
## Call:
## bas.lm(formula = stack.loss ~ ., data = stack.out, prior = "hyper-g-n",     alpha = 3, modelprior = 
##
##
##  Marginal Posterior Inclusion Probabilities:
##  Intercept    Air.Flow  Water.Temp  Acid.Conc.          `1`          `2`
##    1.00000     0.99947     0.21672     0.02816     0.19937     0.05043
##        `3`          `4`         `5`         `6`         `7`         `8`
##    0.21637     0.74110     0.02165     0.02359     0.01944     0.02064
##        `9`         `10`        `11`        `12`        `13`        `14`
##    0.01821     0.01941     0.01924     0.02834     0.13323     0.06536
##       `15`         `16`        `17`        `18`        `19`        `20`
##    0.02241     0.01746     0.02058     0.02008     0.03236     0.03933
##       `21`
##    0.94589
```

```
t(summary(BAS.stack.beta))   # top 5 models
```

```
##                  [,1]         [,2]        [,3]     [,4]        [,5]
## Intercept   1.0000000  1.000000000  1.00000000  1.00000  1.00000000
## Air.Flow    1.0000000  1.000000000  1.00000000  1.00000  1.00000000
## Water.Temp  0.0000000  0.000000000  0.00000000  1.00000  1.00000000
## Acid.Conc.  0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `1`         0.0000000  0.000000000  0.00000000  1.00000  0.00000000
## `2`         0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `3`         0.0000000  0.000000000  0.00000000  1.00000  0.00000000
## `4`         1.0000000  0.000000000  1.00000000  1.00000  1.00000000
## `5`         0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `6`         0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `7`         0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `8`         0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `9`         0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `10`        0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `11`        0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `12`        0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `13`        0.0000000  0.000000000  1.00000000  0.00000  0.00000000
## `14`        0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `15`        0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `16`        0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `17`        0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `18`        0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `19`        0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `20`        0.0000000  0.000000000  0.00000000  0.00000  0.00000000
## `21`        1.0000000  1.000000000  1.00000000  1.00000  1.00000000
## BF          0.1340078  0.008633099  0.08237087  1.00000  0.07353115
## PostProbs   0.2521000  0.119100000  0.02950000  0.02830  0.02640000
## R2          0.9605000  0.927100000  0.96910000  0.98920  0.96870000
## dim         4.0000000  3.000000000  5.00000000  7.00000  5.00000000
## logmarg    22.1689342 19.426639588 21.68226786 24.17879 21.56874516
```

```
BAS.stack.pois
```

```
##
## Call:
## bas.lm(formula = stack.loss ~ ., data = stack.out, prior = "hyper-g-n",     alpha = 3, modelprior = 
##
##
##  Marginal Posterior Inclusion Probabilities:
##  Intercept    Air.Flow  Water.Temp  Acid.Conc.          `1`         `2`
##    1.00000     1.00000     0.48445     0.06696     0.65016     0.14533
##        `3`         `4`         `5`         `6`         `7`         `8`
##    0.67956     0.98325     0.04218     0.05424     0.04047     0.04192
##        `9`        `10`        `11`        `12`        `13`        `14`
##    0.04359     0.04266     0.04269     0.06006     0.34132     0.16867
##       `15`        `16`        `17`        `18`        `19`        `20`
##    0.04931     0.03602     0.04244     0.04159     0.06532     0.11660
##       `21`
##    0.99786
```

```
t(summary(BAS.stack.pois))
```

```
##                [,1]       [,2]       [,3]       [,4]       [,5]
## Intercept   1.00000  1.0000000  1.0000000  1.0000000  1.0000000
## Air.Flow    1.00000  1.0000000  1.0000000  1.0000000  1.0000000
## Water.Temp  1.00000  0.0000000  1.0000000  1.0000000  0.0000000
## Acid.Conc.  0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `1`         1.00000  1.0000000  1.0000000  1.0000000  1.0000000
## `2`         0.00000  0.0000000  0.0000000  1.0000000  0.0000000
## `3`         1.00000  1.0000000  1.0000000  1.0000000  1.0000000
## `4`         1.00000  1.0000000  1.0000000  1.0000000  1.0000000
## `5`         0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `6`         0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `7`         0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `8`         0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `9`         0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `10`        0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `11`        0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `12`        0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `13`        0.00000  1.0000000  1.0000000  0.0000000  0.0000000
## `14`        0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `15`        0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `16`        0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `17`        0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `18`        0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `19`        0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `20`        0.00000  0.0000000  0.0000000  0.0000000  0.0000000
## `21`        1.00000  1.0000000  1.0000000  1.0000000  1.0000000
## BF          1.00000  0.3337882  0.5684913  0.5460933  0.2002082
## PostProbs   0.08420  0.0281000  0.0273000  0.0263000  0.0253000
## R2          0.98920  0.9873000  0.9923000  0.9922000  0.9803000
## dim         7.00000  7.0000000  8.0000000  8.0000000  6.0000000
## logmarg    24.17879 23.0815428 23.6140221 23.5738258 22.5703939
```

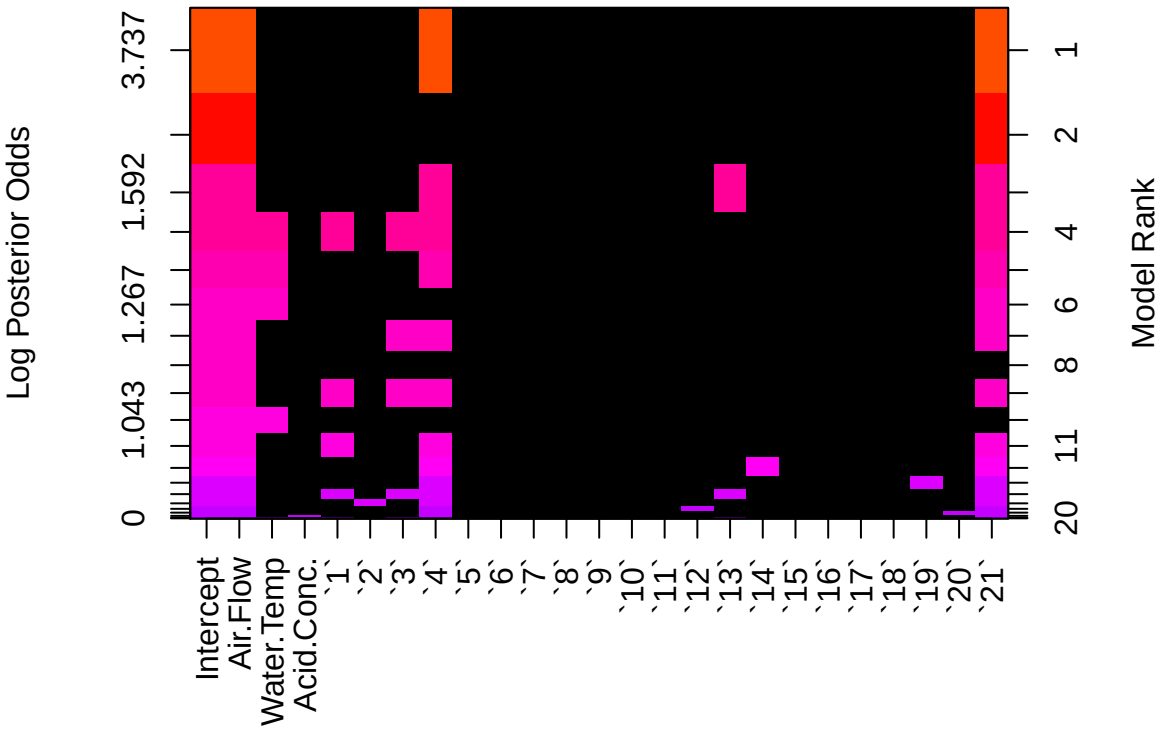
Differences in the most probable models based on the prior. With a beta-binomial prior, we have air flow as the only variable. This model also only has two outliers: 4 and 21. With the truncated poisson prior,

the most likely model has two variables air flow and water temperature with four variables: 1, 3, 4, and 21. The truncated poisson corresponds to the highest probability model from Bayesian model Averaging approach. However, the fourth mots likely model under the truncated beta-binomial is the same as the highest probability model under the truncated prior and Bayesian Model Averaging.

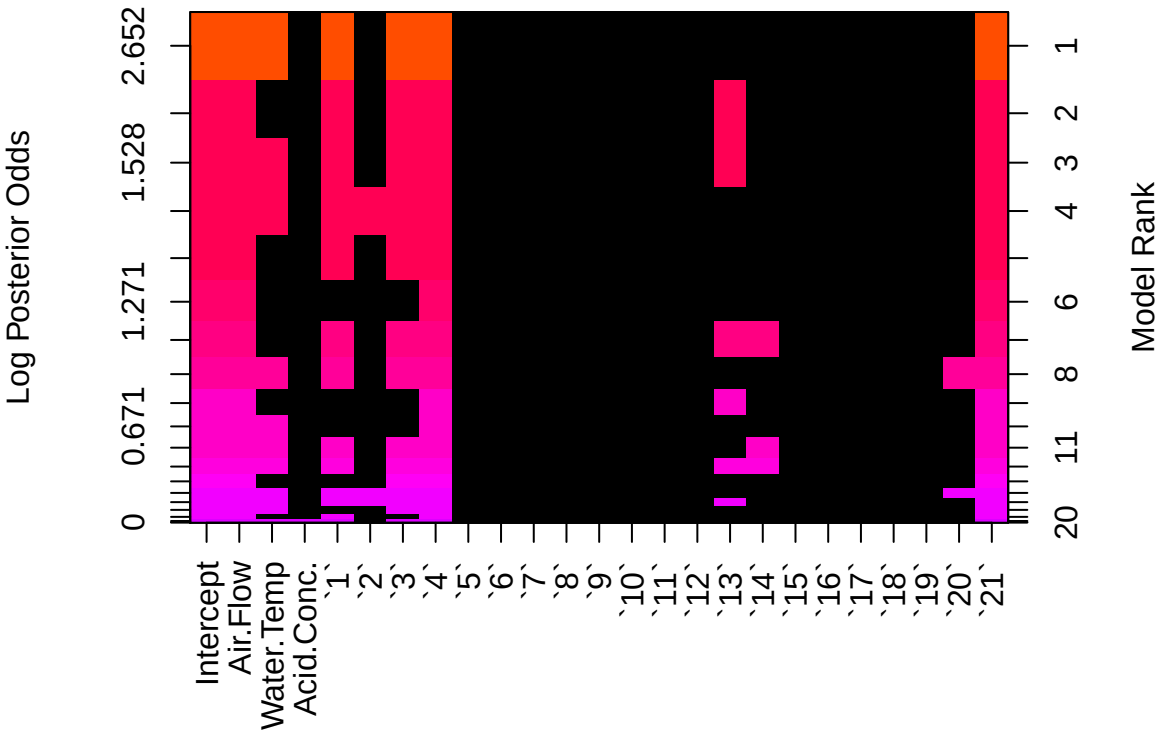
As far as the marginal inclusion probabilities are concerned, the two priors are similar. The marginal inclusion probability for air flow are both basically 1 (under the truncated poisson, it actually is 2). And for water temperature, the truncated poisson gives a much higher probability. Also with the outlier detection, it seems like the truncated poisson gives higher probability almost uniformly. Under the trucated poisson, the four most likely outliers are 1, 3, 4, and 21. Under the truncated beta-binomial, the most probable are 21, 4, 13, and 2. And so under these two different priors the results are actually different.

The following plots show the most probable models under the different prior distributions.

```
par(mfrow = c(1,1))
image(BAS.stack.beta)
```




```
image(BAS.stack.pois)
```



It important to notice that there is more color on the second plot, which means that under the truncated poisson, we are included more variables and detecting more outliers.