HW6: TEAM 3

Zhenyu Tian, Jae Hyun Lee, Presnie Lu, Daniel Deng

Part I

a. Center the 2 predictors, `age` and `lpsa` and using the centered predictors, fit the four possible models with response `lacvol`. Make a table with the R2, the number of predictors p (this does not include the intercept), values of the MLE's under each model, from the OLS fits. Verify that the intercept and its standard error is the same in all models.

```
Prostate_center <- Prostate %>% mutate(age = age-mean(age),lpsa = lpsa-mean(lpsa))
lm1 <- lm(lcavol ~ 1,data = Prostate_center)
lm2 <- lm(lcavol ~ age, data = Prostate_center)
lm3 <- lm(lcavol ~ lpsa, data = Prostate_center)
lm4 <- lm(lcavol ~ age + lpsa, data = Prostate_center)

extract <- function(lm){
  model.summary <- summary(lm)
  result <- list(p = model.summary$df[1] - 1,
         R2 = model.summary$r.squared,
         mle = lm$coefficients,
         intecept.se = model.summary$coefficients[1,2])
  unlist(result) %>% t() %>%  as.data.frame()
}

table1 <- sapply(list(lm1=lm1,lm2=lm2,lm3=lm3,lm4=lm4),extract) %>% reduce(full_join)
kable(table1, caption = "Linear Model Results",digits=3)
```

Table 1: Linear Model Results

p	R2	mle.(Intercept)	intecept.se	mle.age	mle.lpsa
0	0.000	1.35	0.120	NA	NA
1	0.051	1.35	0.117	0.036	NA
1	0.539	1.35	0.082	NA	0.750
2	0.550	1.35	0.081	0.016	0.732

The estimated intercepts for four model are same. But for their standard error is not same. It decreases from null model which has only intercept to full model that has 2 predictor variables.

b. For the four models, compute the log Bayes Factor to compare each model to the null model under the g-prior with $g = n$ where

$$\log BF[M_j : M_0] = \frac{(n - p_j - 1)}{2} \log(1 + g) - \frac{n - 1}{2} \log(1 + g(1 - R_j^2))$$

for the 4 models (j = 0, 1, 2, 3). Exponentiate to obtain the 4 Bayes Factors and complete the table below

```
n = dim(Prostate_center)[1]
p = table1$p
```

```r
R2 = table1$R2
g = n
BF = exp((n-p-1)/2*log(1+g)-(n-1)/2*log(1+g*(1-R2)))
signif(BF,digits = 4)
```

j	p_j	γ_{1j}	γ_{2j}	$BF[M_j : M_0]$
0	0	0	0	1
1	1	1	0	1.191
2	1	0	1	8.291e+14
3	2	1	1	2.443e+14

c. Calculate the posterior probabilities of the four models under the uniform prior distribution,

$$P(M_j \mid Y) = \frac{BF[M_j : M_0]}{\sum_{k=0}^{3} BF[M_k : M_0]}$$

```r
P = BF/sum(BF)
kable(t(P), col.names = c("lm1","lm2","lm3","lm4"),
      caption = "Posterior Probabilities of Each Model")
```

Table 3: Posterior Probabilities of Each Model

lm1	lm2	lm3	lm4
0	0	0.77238	0.22762

d. Calculate the probability that the coefficient for `lpsa` and `age` are not zero,

$$\sum_j \gamma_{1j} P(M_j \mid Y)$$

```r
gamma1 <- c(0,1,0,1)
gamma2 <- c(0,0,1,1)
p.age <- sum(gamma1*P)
p.lpsa <- sum(gamma2*P)
kable(data.frame(age = p.age, lpsa = p.lpsa),
      caption = "Probability to Include the Predictors",
      digits = 3)
```

Table 4: Probability to Include the Predictors

age	lpsa
0.228	1

e. Calculate the posterior mean for β_{lpsa} under the g-prior and model averaging

$$E[\beta_{lpsa} \mid Y] = \frac{g}{1+g} \sum_j \hat{\beta}_{lpsa,M_j} P(M_j \mid Y)$$

where $\hat{\beta}_{lpsa,M_j}$ is the OLS/MLE estimate from your table above. (Repeat for `age`).

```
beta.lpsa <- table1$mle.lpsa
beta.age <- table1$mle.age
postmean.lpsa <- g/(1+g)*sum(beta.lpsa*P,na.rm=T)
postmean.age <- g/(1+g)*sum(beta.age*P,na.rm=T)
kable(cbind(age=postmean.age,lpsa=postmean.lpsa),digits =3,
      caption = "Posterior Mean for Beta_lpsa and Beta_age")
```

Table 5: Posterior Mean for Beta_lpsa and Beta_age

age	lpsa
0.004	0.738

f. Confirm your answers using `bas.lm`

```
baslm <- bas.lm(lcavol ~ age+lpsa,
       data = Prostate_center,
       prior = "g-prior",
       alpha = g,
       modelprior = uniform())
kable(summary(baslm),digits = 3, caption = "BAS Summary")
```

Table 6: BAS Summary

	P(B != 0 | Y)	model 1	model 2	model 3	model 4
Intercept	1.000	1.000	1.000	1.000	1
age	0.228	0.000	1.000	1.000	0
lpsa	1.000	1.000	1.000	0.000	0
BF	NA	1.000	0.295	0.000	0
PostProbs	NA	0.772	0.228	0.000	0
R2	NA	0.539	0.550	0.051	0
dim	NA	2.000	3.000	2.000	1
logmarg	NA	34.351	33.130	0.175	0

```
kable(data.frame(t(BF[c(3,4,2,1)]/BF[3])),
      col.names =  c("Modle 3","Modle 4","Modle 2","Modle 1"),
      digits = 3,
      caption = "Bayes Factors with Model 3 as Baseline")
```

Table 7: Bayes Factors with Model 3 as Baseline

Modle 3	Modle 4	Modle 2	Modle 1
1	0.295	0	0

When we check first column that represent the probability of each variable included in model is same with answer with Q1-d. Moreover, marginal posterior distribution for models are corresponding to answer of Q1-

c. We also can find that R-squared for each models are same with answer of Q1-a. Lastly, at first glance, Bayesian Factor seems to be different from above answers. However, if we consider the largest BF as 1 and divide other BF by the largest BF, we can find that they are exactly same.

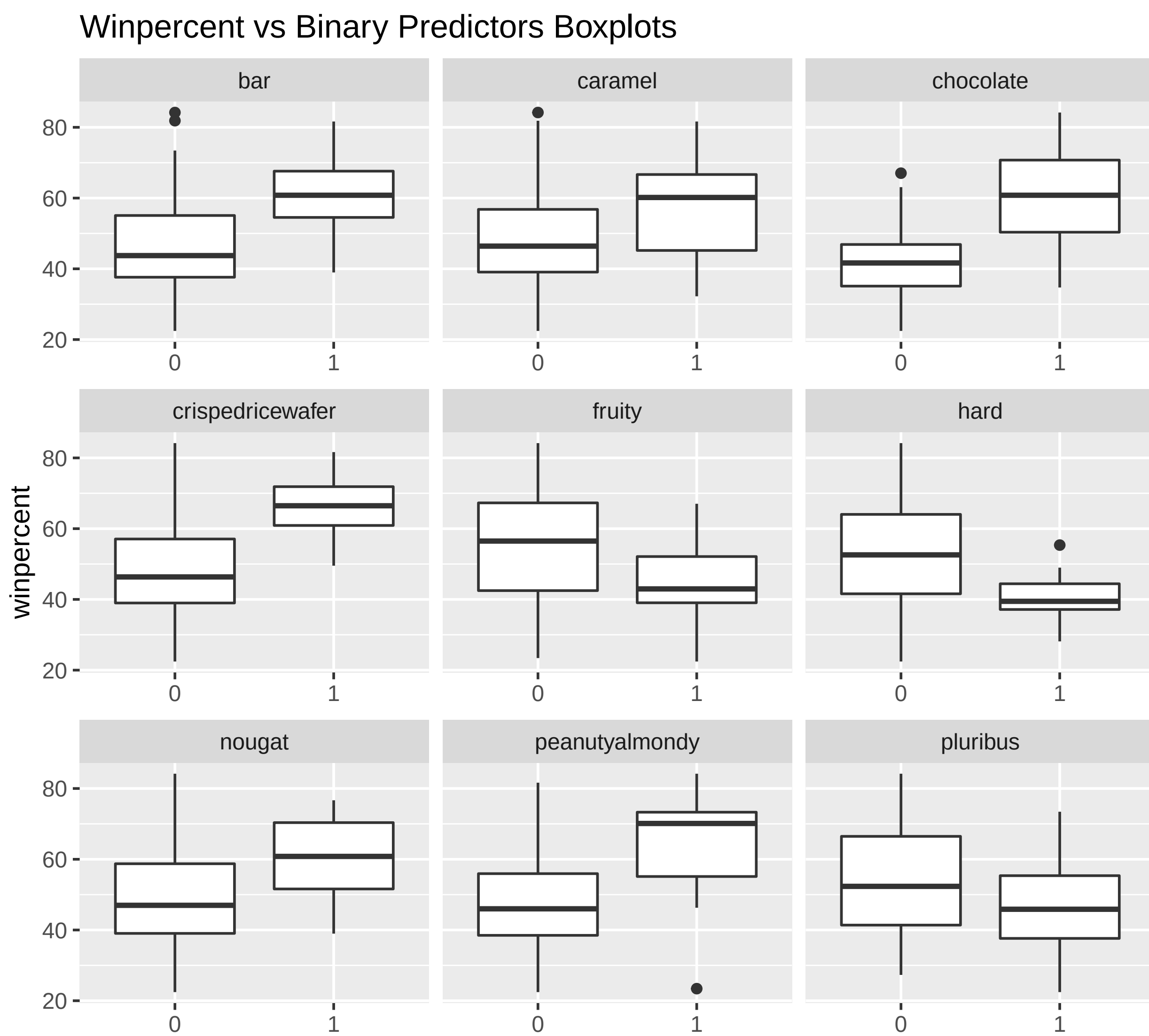
Part 2

Data Description:

Header	Description
chocolate	Does it contain chocolate?
fruity	Is it fruit flavored?
caramel	Is there caramel in the candy?
peanutalmondy	Does it contain peanuts, peanut butter or almonds?
nougat	Does it contain nougat?
crispedricewafer	Does it contain crisped rice, wafers, or a cookie component?
hard	Is it a hard candy?
bar	Is it a candy bar?
pluribus	Is it one of many candies in a bag or box?
sugarpercent	The percentile of sugar it falls under within the data set.
pricepercent	The unit price percentile compared to the rest of the set.
winpercent	The overall win percentage according to 269,000 matchups.

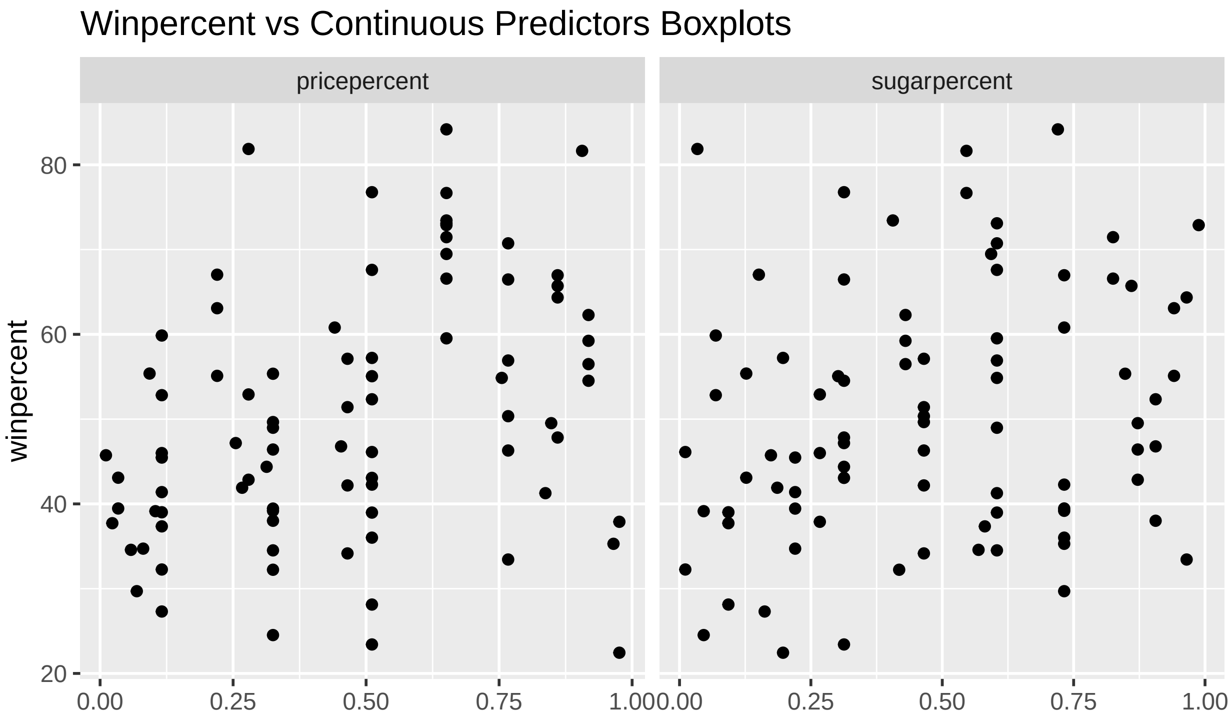
a. Explore the association between `winpercent` and the other other variables graphically and comment.

```
candy <- read.csv("candy-data.csv",header = TRUE)

candy.bin <- candy %>% dplyr::select(chocolate:pluribus,winpercent) %>%
  gather(key = "candy_type", value = value,-winpercent)

ggplot(candy.bin,aes(x = as.factor(value), y = winpercent))+
  geom_boxplot()+
  facet_wrap(~candy_type,scales = "free_x")+
  labs(x = "", title = "Winpercent vs Binary Predictors Boxplots")
```



When we examine boxplots of predictor variables versus winpercent, most of predictors seems to have linear association with response variable, `winpercent`, because the distribution of winpercent significantly varies according to level of predictors.

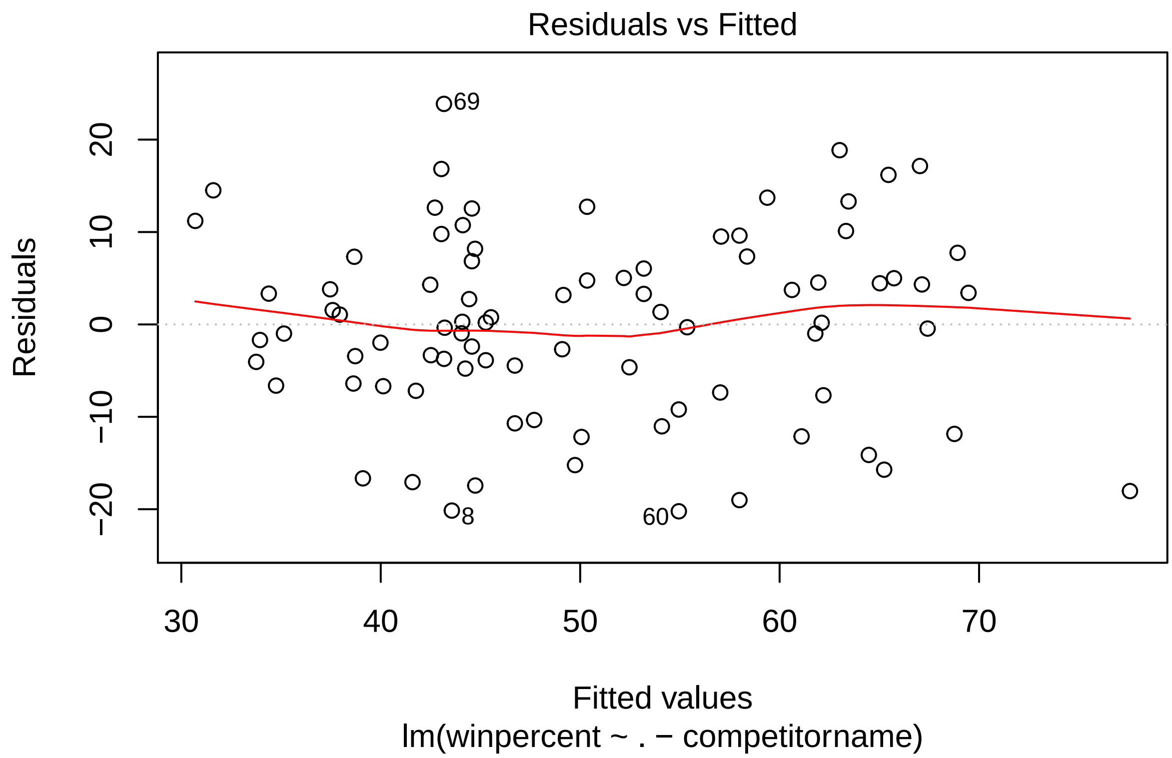
```
candy.cont <- candy %>% dplyr::select(sugarpercent:winpercent) %>%
  gather(key = "type", value = value,-winpercent)

ggplot(candy.cont,aes(x = value, y = winpercent))+
  geom_point()+
  facet_wrap(~type,scales = "free_x")+
  labs(x = "", title = "Winpercent vs Continuous Predictors Boxplots")
```



For continous predictors, it is hard to find any linear association with `winpercent`.

b. Fit the full model with all predictors (except `competitorname`) and plot residuals versus fitted values. Comment on whether the model seems appropriate or you need to transform. Create confidence intervals for all of the coefficients and present in a table sorted by the estimates from high to low. (present as a nicely formated table).

```
lm_full <- lm(winpercent~.-competitorname ,data = candy)

plot(lm_full,which =1)
```




```r
table2 <- cbind.data.frame(names(lm_full$coefficients),lm_full$coefficients,confint(lm_full)) %>%
  `colnames<-` (c("Variable","Beta","Lower_Bound","Upper_Bound")) %>%
  arrange(desc(Beta))
kable(table2, digits = 3, caption = "Confidence intervals for coefficients")
```

Table 9: Confidence intervals for coefficients

Variable	Beta	Lower_Bound	Upper_Bound
(Intercept)	34.534	25.924	43.144
chocolate	19.748	11.978	27.518
peanutyalmondy	10.071	2.864	17.277
fruity	9.422	1.923	16.922
sugarpercent	9.087	-0.200	18.373
crispedricewafer	8.919	-1.580	19.418
caramel	2.224	-5.065	9.514
nougat	0.804	-10.588	12.197
bar	0.442	-9.645	10.528
pluribus	-0.854	-6.913	5.204
pricepercent	-5.928	-16.916	5.060
hard	-6.165	-13.051	0.721

Model seems to be appropriate because we cannot find any signs of violation such as heterogeneity of variance or nonlinearity.

c. Are there any interactions between features that you think might be relevant? Are there any interactions that you think are not really feasible, hard and nougat? Fit the model with all possible interactions and comment on the summary.

```
lm_int <- lm(winpercent~(.-competitorname)^2 ,data = candy)
kable(summary(lm_int)$coefficients,
      caption="Summary of the Linear Model with All Two-way Interactions")
```

Table 10: Summary of the Linear Model with All Two-way Interactions

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	35.6054236	11.581566	3.0743186	0.0040742
chocolate	4.6923434	18.263364	0.2569266	0.7987399
fruity	10.4975762	12.875115	0.8153384	0.4203926
caramel	-1.8083562	26.636659	-0.0678898	0.9462598
peanutyalmondy	-5.2497914	29.263328	-0.1793983	0.8586595
nougat	302.4051494	206.277053	1.4660145	0.1515717
crispedricewafer	122.1005823	133.497916	0.9146254	0.3666468
hard	4.5490050	20.110595	0.2261994	0.8223621
bar	-35.4080094	40.017048	-0.8848231	0.3822905
pluribus	0.4324092	14.394148	0.0300406	0.9762053
sugarpercent	-20.5220971	25.866688	-0.7933794	0.4329027
pricepercent	11.6038881	29.311542	0.3958812	0.6945957
chocolate:fruity	3.2707140	22.727900	0.1439074	0.8863990
chocolate:caramel	3.5348110	43.225754	0.0817756	0.9352910
chocolate:peanutyalmondy	39.6143374	17.747399	2.2321207	0.0321058
chocolate:nougat	-67.8308779	126.890972	-0.5345603	0.5963337
chocolate:pluribus	7.6445209	18.554095	0.4120126	0.6828428
chocolate:sugarpercent	17.0204802	29.698228	0.5731143	0.5702314
chocolate:pricepercent	-19.5836466	32.583670	-0.6010264	0.5516953
fruity:caramel	10.6409663	32.550819	0.3269032	0.7456882
fruity:hard	-9.2685852	18.207447	-0.5090546	0.6139084
fruity:pluribus	5.9955026	15.890853	0.3772927	0.7082343
fruity:sugarpercent	15.3921686	21.921059	0.7021636	0.4872208
fruity:pricepercent	-39.8662516	32.792814	-1.2157008	0.2322354
caramel:peanutyalmondy	12.2537969	49.704817	0.2465314	0.8067112
caramel:nougat	93.0462774	53.106185	1.7520799	0.0885185
caramel:crispedricewafer	40.0453241	100.300066	0.3992552	0.6921310
caramel:hard	-6.1400346	33.092978	-0.1855389	0.8538775
caramel:bar	-62.7206495	54.624359	-1.1482176	0.2586662
caramel:pluribus	-16.2098114	38.070584	-0.4257831	0.6728729
caramel:sugarpercent	-77.2999971	142.120160	-0.5439059	0.5899545
caramel:pricepercent	88.6285275	230.147525	0.3850944	0.7024979
peanutyalmondy:nougat	-8.8992664	49.129546	-0.1811388	0.8573035
peanutyalmondy:bar	-45.2029201	27.375419	-1.6512230	0.1076390
peanutyalmondy:pluribus	-15.8000783	16.616433	-0.9508707	0.3481905
peanutyalmondy:sugarpercent	-45.5144011	29.728289	-1.5310131	0.1347556
peanutyalmondy:pricepercent	36.2092304	61.363236	0.5900802	0.5589276
nougat:sugarpercent	-247.1209393	215.463675	-1.1469262	0.2591924
nougat:pricepercent	-197.2385816	245.043022	-0.8049141	0.4263035
crispedricewafer:bar	-20.4441903	90.535598	-0.2258138	0.8226597
crispedricewafer:sugarpercent	-108.8448767	183.119159	-0.5943937	0.5560719

	Estimate	Std. Error	t value	Pr(>|t|)
crispedricewafer:pricepercent	-56.9423423	60.492374	-0.9413144	0.3529958
hard:pluribus	0.3138291	8.356697	0.0375542	0.9702565
hard:sugarpercent	-0.3194782	12.993988	-0.0245866	0.9805243
hard:pricepercent	-4.7801298	18.142936	-0.2634706	0.7937329
bar:sugarpercent	82.8834258	66.141769	1.2531178	0.2184701
bar:pricepercent	9.1483174	32.354402	0.2827534	0.7790314
pluribus:sugarpercent	2.2066117	15.445891	0.1428608	0.8872194
pluribus:pricepercent	-7.7664982	20.056428	-0.3872324	0.7009289
sugarpercent:pricepercent	49.6726525	34.626231	1.4345383	0.1602942

```r
kable(summary(lm_int)$r.squared,
      caption="R2 of the Linear Model with All Two-way Interactions",
      col.names = "R2")
```

Table 11: R2 of the Linear Model with All Two-way Interactions

R2
0.8009477

Might be relevant: `chocolate` and `peanutyalmondy`, `chocolate` and `caramel`, `fruity` and `hard`, `caramel` and `nougat`, etc.
Not feasible: `nougat` and `hard`, `nougat` and `fruity`, `peanutyalmondy` and `fruity`, maybe `caramel` and `fruity`, `crispdricewafer` and `fruity`.
Summary of model indicates that the model is overfitted because some coefficients are showing considerably large coefficient and standard error which might lead to complete separation. The large number of predictors of model might cause this problem.

d. Using the `step` function with `AIC` which variables and interactions (you do not need to start with all interactions) are in the best AIC model? Provide a summary of the final model.

```r
index <- is.na(lm_int$coefficients)
call <- paste("winpercent~ (.-competitorname)^2",
              paste(lm_int$coefficients[index] %>% names(),
                    collapse = "-"),
              sep = "-")
lm_int2 <- lm(call,data=candy)
lm_AIC <- step(lm_int2, k=2, trace = F)
## or replace above using
# lm_AIC <- step(lm_int,k=e, trace = F)
kable(summary(lm_AIC)$coefficients,
      caption="Summary of the Best AIC Model with All Two-way Interactions")
```

Table 12: Summary of the Best AIC Model with All Two-way Interactions

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	39.567010	5.434413	7.2808250	0.0000000
chocolate	0.735213	6.917932	0.1062764	0.9157569

	Estimate	Std. Error	t value	Pr(>|t|)
fruity	7.739789	6.097232	1.2693938	0.2097444
caramel	10.625174	10.830948	0.9810013	0.3309670
peanutyalmondy	4.833880	12.615763	0.3831619	0.7031040
nougat	224.228937	91.492162	2.4507994	0.0175230
crispedricewafer	70.329183	23.698115	2.9677121	0.0044646
hard	-5.866404	2.882267	-2.0353438	0.0467364
bar	-12.465667	16.221353	-0.7684727	0.4455543
pluribus	4.399382	2.977697	1.4774443	0.1453655
sugarpercent	-19.730597	11.638584	-1.6952748	0.0957829
pricepercent	-7.958604	9.849616	-0.8080117	0.4226292
chocolate:peanutyalmondy	41.386644	11.190864	3.6982527	0.0005094
chocolate:nougat	-56.176552	19.815760	-2.8349430	0.0064360
chocolate:sugarpercent	20.068897	14.102619	1.4230617	0.1604702
fruity:sugarpercent	17.909850	10.665316	1.6792611	0.0988791
fruity:pricepercent	-26.349559	10.640200	-2.4764158	0.0164357
caramel:nougat	57.164333	23.892597	2.3925542	0.0202402
caramel:crispedricewafer	23.417824	18.455871	1.2688550	0.2099351
caramel:bar	-23.760668	20.875508	-1.1382079	0.2600584
caramel:sugarpercent	-58.158696	21.734090	-2.6759205	0.0098453
caramel:pricepercent	36.266724	28.623655	1.2670193	0.2105857
peanutyalmondy:bar	-32.107833	8.391708	-3.8261379	0.0003399
peanutyalmondy:pluribus	-11.673119	7.660406	-1.5238252	0.1333879
peanutyalmondy:sugarpercent	-34.620184	16.235807	-2.1323353	0.0375437
nougat:sugarpercent	-195.628535	110.623592	-1.7684160	0.0826398
nougat:pricepercent	-120.604682	67.450519	-1.7880468	0.0793797
crispedricewafer:sugarpercent	-74.624867	28.016431	-2.6636108	0.0101686
crispedricewafer:pricepercent	-34.371478	29.689625	-1.1576932	0.2520843
bar:sugarpercent	44.421400	33.701793	1.3180723	0.1930435
sugarpercent:pricepercent	46.074621	19.753603	2.3324666	0.0234335

```
kable(summary(lm_AIC)$r.squared,
      caption="R2 of the Best AIC Model with All Two-way Interactions",
      col.names = "R2")
```

Table 13: R2 of the Best AIC Model with All Two-way Interactions

R2
0.7826043

Even after variable selection, some coefficients of predictors that seems to be unstable are remaining. We can also find that Adjustd R-squared of final model is improved compared to previous model.

e. Fit the model selected using `AIC` and create confidence intervals for each of the coefficients formated as above. Do any of the intervals contain zero? Do any intervals seem poorly estimated based on modeling winpercent that is between 0 and 100?

```
CI_AIC <- confint(lm_AIC)

contain0 <- function(interval){
```

```r
  (interval[,1]<=0) & (interval[,2]>= 0)
}

table.CI <- data.frame(names(lm_AIC$coefficients),lm_AIC$coefficients,CI_AIC, contain0(CI_AIC)) %>%
  `colnames<-`(c("Variable","Beta","LB","UB","Contain_0")) %>%
  arrange(desc(Beta))

table.CI$Variable <- str_remove_all(table.CI$Variable,"\\(")
table.CI$Variable <- str_remove_all(table.CI$Variable,"\\)")

kable(table.CI, digits = 3,caption = "95% CI for the coefficients of best AIC model")
```

Table 14: 95% CI for the coefficients of best AIC model

Variable	Beta	LB	UB	Contain_0
nougat	224.229	40.798	407.660	FALSE
crispedricewafer	70.329	22.817	117.841	FALSE
caramel:nougat	57.164	9.263	105.066	FALSE
sugarpercent:pricepercent	46.075	6.471	85.678	FALSE
bar:sugarpercent	44.421	-23.147	111.989	TRUE
chocolate:peanutyalmondy	41.387	18.950	63.823	FALSE
Intercept	39.567	28.672	50.462	FALSE
caramel:pricepercent	36.267	-21.120	93.654	TRUE
caramel:crispedricewafer	23.418	-13.584	60.420	TRUE
chocolate:sugarpercent	20.069	-8.205	48.343	TRUE
fruity:sugarpercent	17.910	-3.473	39.293	TRUE
caramel	10.625	-11.090	32.340	TRUE
fruity	7.740	-4.484	19.964	TRUE
peanutyalmondy	4.834	-20.459	30.127	TRUE
pluribus	4.399	-1.571	10.369	TRUE
chocolate	0.735	-13.134	14.605	TRUE
hard	-5.866	-11.645	-0.088	FALSE
pricepercent	-7.959	-27.706	11.789	TRUE
peanutyalmondy:pluribus	-11.673	-27.031	3.685	TRUE
bar	-12.466	-44.988	20.056	TRUE
sugarpercent	-19.731	-43.065	3.603	TRUE
caramel:bar	-23.761	-65.614	18.092	TRUE
fruity:pricepercent	-26.350	-47.682	-5.017	FALSE
peanutyalmondy:bar	-32.108	-48.932	-15.283	FALSE
crispedricewafer:pricepercent	-34.371	-93.896	25.153	TRUE
peanutyalmondy:sugarpercent	-34.620	-67.171	-2.069	FALSE
chocolate:nougat	-56.177	-95.905	-16.448	FALSE
caramel:sugarpercent	-58.159	-101.733	-14.584	FALSE
crispedricewafer:sugarpercent	-74.625	-130.794	-18.455	FALSE
nougat:pricepercent	-120.605	-255.835	14.625	TRUE
nougat:sugarpercent	-195.629	-417.415	26.158	TRUE

The intervals of `nougat`, `nougat:pricepercent`, `nougat:sugarpercent` seems to be estimated poorly considering that winpercent has values between 0 and 100, because estimated intervals for these variables are so large that they can cause predicted values to have values that are not included between 0 and 100. Also, 18 of the confidence intervals contain 0, which indicates that the corresponding main effects and interactions may not be significant.

f. Use BMA to fit a model to explore which features predict `winpercent` (If your team number is less than or equal to 5 use the g-prior with `a=n`. If your team number is greater than 5 use the `prior='JZS'`. Use `method="MCMC"` and check the diagnostic plots for convergence, rerunning longer if it looks like it has not converged. Provide a summary of the output. *Handling models that are not full rank (as the full model with all 2 way interactions is experimental in BAS; I suggest starting with the AIC model `formula(candy.AIC)` (see the file candy-EDA.Rmd) or be judicious in terms of choosing interactions to go in based on your subjective information on Halloween candy so that run times are not tooooo long.*

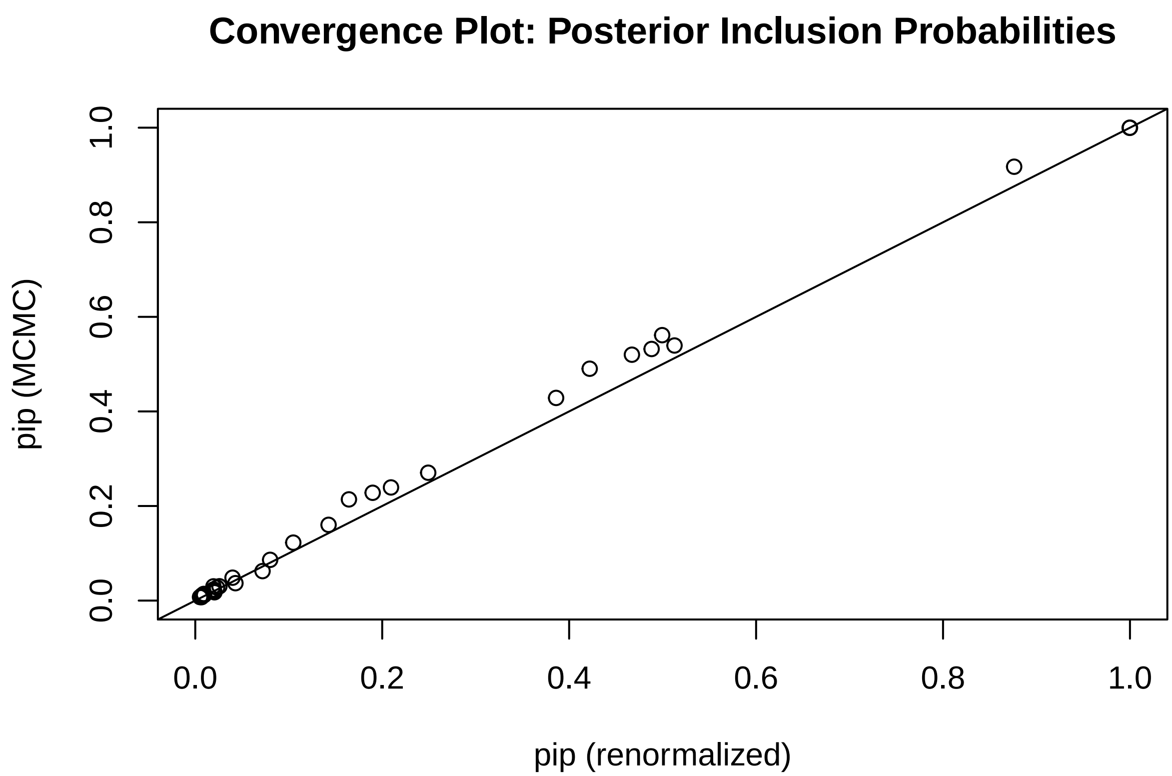
```
blm.candy <- bas.lm(formula(lm_AIC),
       prior="g-prior",
       modelprior=uniform(),
       method="MCMC",
       data = candy,
       alpha = dim(candy)[1])

beta.blm <- coef(blm.candy)
kable(cbind(beta.blm$namesx,
            beta.blm$postmean %>% round(3),
            beta.blm$postsd %>% round(digits=3),
            beta.blm$probne0 %>% signif(digits = 4)),
      col.names = c("variables","post mean","post SD", "post p(B !=0)"),
      caption = "BMA Model Summary")
```

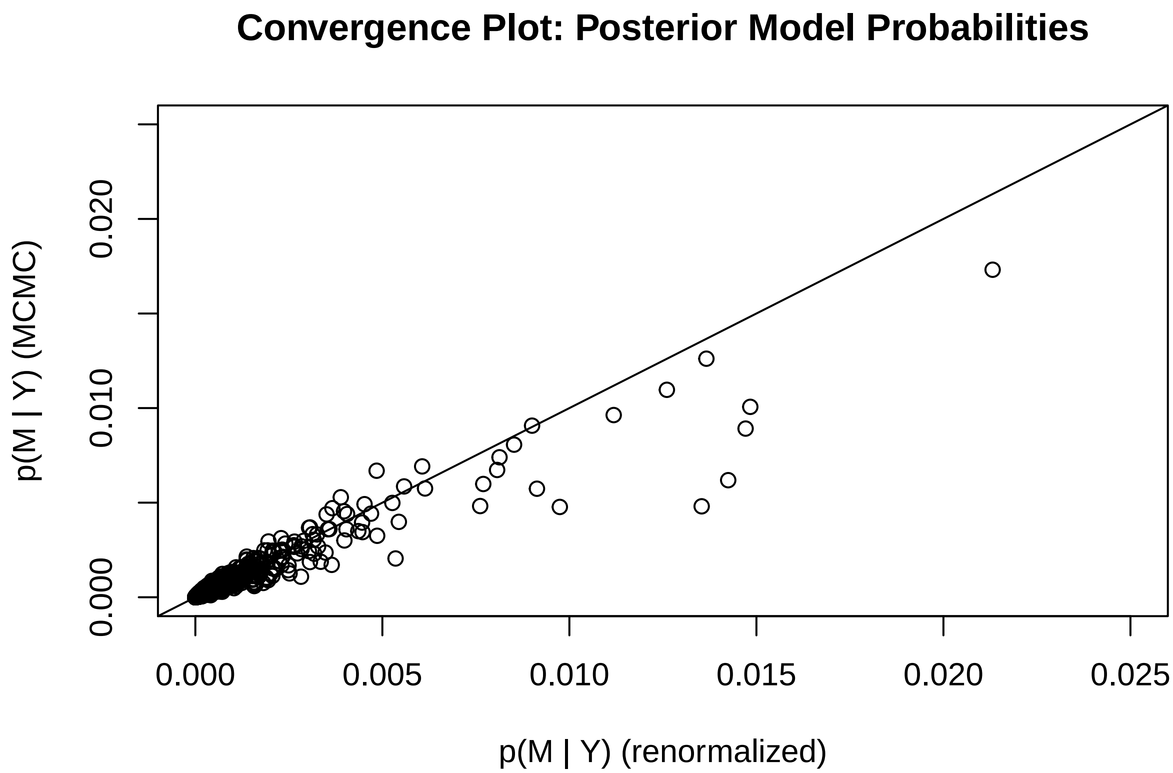
Table 15: BMA Model Summary

variables	post mean	post SD	post p(B !=0)
Intercept	50.317	1.148	1
chocolate	14.569	5.585	0.9997
fruity	4.424	4.988	0.5613
caramel	0.532	2.959	0.228
peanutyalmondy	2.24	12.134	0.9175
nougat	4.428	15.97	0.2392
crispedricewafer	6.862	11.371	0.4903
hard	-1.332	2.799	0.2704
bar	4.337	6.047	0.52
pluribus	0.189	1.409	0.1602
sugarpercent	3.564	5.186	0.532
pricepercent	-0.723	3.891	0.214
chocolate:peanutyalmondy	12.387	14.468	0.5394
chocolate:nougat	-2.734	10.584	0.08616
chocolate:sugarpercent	0.54	3.338	0.06233
fruity:sugarpercent	0.083	1.743	0.03029
fruity:pricepercent	-0.391	2.825	0.03005
caramel:nougat	0.059	1.142	0.007281
caramel:crispedricewafer	0.001	1.05	0.01115
caramel:bar	0.171	1.597	0.02035
caramel:sugarpercent	-0.12	2.149	0.01161
caramel:pricepercent	0.314	3.25	0.01403
peanutyalmondy:bar	-9.127	11.855	0.4285
peanutyalmondy:pluribus	-0.724	3.93	0.04841
peanutyalmondy:sugarpercent	-0.072	2.856	0.0368
nougat:sugarpercent	-2.166	18.447	0.0235

variables	post mean	post SD	post p(B !=0)
nougat:pricepercent	0.289	5.332	0.007408
crispedricewafer:sugarpercent	-5.09	15.667	0.1226
crispedricewafer:pricepercent	0.012	3.417	0.01082
bar:sugarpercent	0.045	2.53	0.0176
sugarpercent:pricepercent	0.657	4.871	0.02741

```r
diagnostics(blm.candy)
```

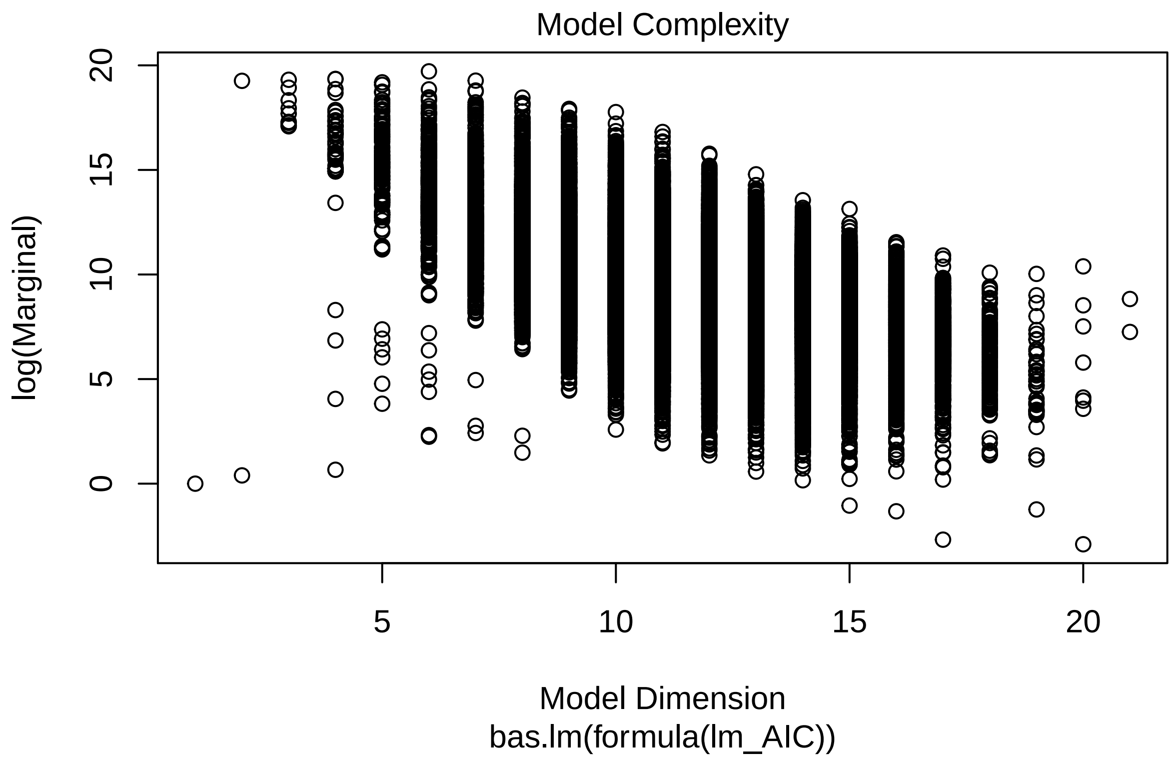




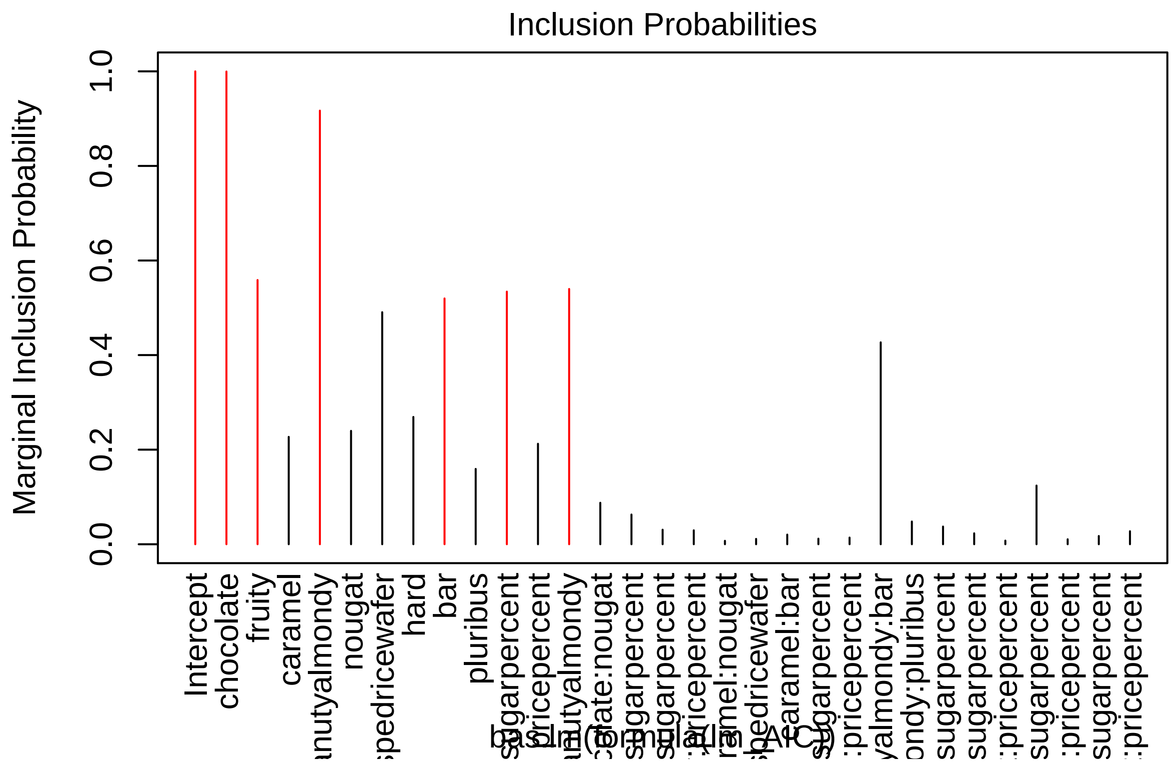
We can confirm that the model converges because our estimates for posterior inclusion probability and posterior model probability are very simliar with theoretical value.

g. Create a plot of the model space and the marginal inclusion probabilities and comment. How do these results compare to AIC?

```
plot(blm.candy, which = 3)
```




```
plot(blm.candy, which = 4)
```



The plot of model compexity shows the distribution of candidate models. We can find that the model which has six variables has the largest marginal posterior probability. Generally, the marginal posterior probabilities of models having less variables have larger value than models having more variables. As an evidence, we can find negative linear association between Margina posterior distribtuin and model dimension.

The plot of posterior inclusion probability shows that predictor variables' average probabilities of being included in models. `Intercept`, `chocolate`, `peanutyalmondy` has nearly 1 probabilities of being included in models and `fruity`, `bar`, `sugarpercent`, `chocolate:peanutyalmondy` has above 0.5 probabilities of being included in models. We also find out that the predictors with high inclusion probabilities are very different from the significant predictors from the AIC model.

h. Provide a table of estimates of the coefficients and credible intervals (sorted as above) and comment on how they compare to the estimates under the best AIC model. According to your model which features are associated with high overall win percentage? What features are associated with low overall win percentage? Which features do not seem to be important? (Be carful with interactions here!) *(optional: create plots of the posterior densities of some key variables - are there any bi-modal distributions, if so comment)*

```r
beta.blm <- coef(blm.candy)
table.beta <- confint(beta.blm)
table.beta <- cbind.data.frame(Variable = rownames(table.beta),
                               table.beta[,1,drop=F],
                               table.beta[,2,drop=F],
                               table.beta[,3,drop=F]) %>%
  `colnames<-`(c("Variable","LB","UB","Beta")) %>%
  arrange(desc(Beta))
kable(table.beta, digits = 3, caption = "Coefficients and Credible intervals of BMA")
```

Table 16: Coefficients and Credible intervals of BMA

Variable	LB	UB	Beta
Intercept	48.003	52.594	50.317
chocolate	3.528	25.655	14.569
chocolate:peanutyalmondy	0.000	41.126	12.387
crispedricewafer	-2.818	36.251	6.862
nougat	-6.915	43.983	4.428
fruity	0.000	14.611	4.424
bar	-1.533	18.456	4.337
sugarpercent	-1.902	15.732	3.564
peanutyalmondy	-23.997	24.787	2.240
sugarpercent:pricepercent	0.000	0.000	0.657
chocolate:sugarpercent	0.000	5.385	0.540
caramel	-3.025	8.917	0.532
caramel:pricepercent	0.000	0.000	0.314
nougat:pricepercent	0.000	0.000	0.289
pluribus	-1.843	4.948	0.189
caramel:bar	0.000	0.000	0.171
fruity:sugarpercent	0.000	0.000	0.083
caramel:nougat	0.000	0.000	0.059
bar:sugarpercent	0.000	0.000	0.045
crispedricewafer:pricepercent	0.000	0.000	0.012
caramel:crispedricewafer	0.000	0.000	0.001
peanutyalmondy:sugarpercent	0.000	0.000	-0.072
caramel:sugarpercent	0.000	0.000	-0.120
fruity:pricepercent	0.000	0.000	-0.391
pricepercent	-12.229	4.380	-0.723
peanutyalmondy:pluribus	-0.096	0.206	-0.724
hard	-8.934	0.183	-1.332
nougat:sugarpercent	0.000	0.000	-2.166
chocolate:nougat	-33.500	0.000	-2.734
crispedricewafer:sugarpercent	-47.271	0.000	-5.090
peanutyalmondy:bar	-31.793	0.000	-9.127

```
comparison <- full_join(table.beta, table.CI, by = "Variable", suffix = c(".bma",".aic"))
kable(comparison[,-8], digits = 3,
      caption = "Coefficients and credible intervals of BMA versus AIC")
```

Table 17: Coefficients and credible intervals of BMA versus AIC

Variable	LB.bma	UB.bma	Beta.bma	Beta.aic	LB.aic	UB.aic
Intercept	48.003	52.594	50.317	39.567	28.672	50.462
chocolate	3.528	25.655	14.569	0.735	-13.134	14.605
chocolate:peanutyalmondy	0.000	41.126	12.387	41.387	18.950	63.823
crispedricewafer	-2.818	36.251	6.862	70.329	22.817	117.841
nougat	-6.915	43.983	4.428	224.229	40.798	407.660
fruity	0.000	14.611	4.424	7.740	-4.484	19.964
bar	-1.533	18.456	4.337	-12.466	-44.988	20.056
sugarpercent	-1.902	15.732	3.564	-19.731	-43.065	3.603
peanutyalmondy	-23.997	24.787	2.240	4.834	-20.459	30.127
sugarpercent:pricepercent	0.000	0.000	0.657	46.075	6.471	85.678

Variable	LB.bma	UB.bma	Beta.bma	Beta.aic	LB.aic	UB.aic
chocolate:sugarpercent	0.000	5.385	0.540	20.069	-8.205	48.343
caramel	-3.025	8.917	0.532	10.625	-11.090	32.340
caramel:pricepercent	0.000	0.000	0.314	36.267	-21.120	93.654
nougat:pricepercent	0.000	0.000	0.289	-120.605	-255.835	14.625
pluribus	-1.843	4.948	0.189	4.399	-1.571	10.369
caramel:bar	0.000	0.000	0.171	-23.761	-65.614	18.092
fruity:sugarpercent	0.000	0.000	0.083	17.910	-3.473	39.293
caramel:nougat	0.000	0.000	0.059	57.164	9.263	105.066
bar:sugarpercent	0.000	0.000	0.045	44.421	-23.147	111.989
crispedricewafer:pricepercent	0.000	0.000	0.012	-34.371	-93.896	25.153
caramel:crispedricewafer	0.000	0.000	0.001	23.418	-13.584	60.420
peanutyalmondy:sugarpercent	0.000	0.000	-0.072	-34.620	-67.171	-2.069
caramel:sugarpercent	0.000	0.000	-0.120	-58.159	-101.733	-14.584
fruity:pricepercent	0.000	0.000	-0.391	-26.350	-47.682	-5.017
pricepercent	-12.229	4.380	-0.723	-7.959	-27.706	11.789
peanutyalmondy:pluribus	-0.096	0.206	-0.724	-11.673	-27.031	3.685
hard	-8.934	0.183	-1.332	-5.866	-11.645	-0.088
nougat:sugarpercent	0.000	0.000	-2.166	-195.629	-417.415	26.158
chocolate:nougat	-33.500	0.000	-2.734	-56.177	-95.905	-16.448
crispedricewafer:sugarpercent	-47.271	0.000	-5.090	-74.625	-130.794	-18.455
peanutyalmondy:bar	-31.793	0.000	-9.127	-32.108	-48.932	-15.283

Comparing to best AIC model, BMA model has smaller intervals for overall variables. Moreover, coefficients for predictors are also more stable than best AIC model because we cannot find predictors which might cause problem that predicts `winpercent` not included in 0 and 100.

Features that associated with overall high `winpercent` are `chocolate`,`chocolate:peanutyalmondy`, `crispedricewafer`, `fruity`, `nougat`, `bar`, `sugarpercent`, `peanutyalmondy`. On the contrary, features that associated with overall low `winpercent` are `nougat:sugarpercent`, `chocolate:nougat`, `nougat:sugarpercent`, `crispedricewafer:sugarpercent`, `peanutyalmondy:bar`.

Features seems to be important are `chocolate` and `peanutyalmondy` because they have positive effect on `winpercent` even does their interaction term.

i. Which variables are included in the Highest Probability Model, the Median Probability Model and the "Best Probability Model" How do these campare to the best AIC model?

```
#HPM
HPM = predict(blm.candy, estimator = "HPM")$best.vars
#MPM

MPM = predict(blm.candy, estimator = "MPM")$best.vars

#BPM
BPM = predict(blm.candy, estimator = "BPM")$best.vars

max.len = max(length(HPM), length(MPM),length(BPM),
              length(names(lm_AIC$coefficients)))
HPM2 = c(HPM, rep(NA, max.len - length(HPM)))
MPM2 = c(MPM, rep(NA, max.len - length(MPM)))
BPM2 = c(BPM, rep(NA, max.len - length(BPM)))
```

```r
kable(data.frame(HPM = HPM2, MPM = MPM2, BPM = BPM2,
           AIC = names(lm_AIC$coefficients)),
      caption = "Variables of Different Models")
```

Table 18: Variables of Different Models

HPM	MPM	BPM	AIC
Intercept	Intercept	Intercept	(Intercept)
chocolate	chocolate	chocolate	chocolate
peanutyalmondy	fruity	fruity	fruity
bar	peanutyalmondy	peanutyalmondy	caramel
chocolate:peanutyalmondy	bar	crispedricewafer	peanutyalmondy
peanutyalmondy:bar	sugarpercent	bar	nougat
NA	chocolate:peanutyalmondy	pluribus	crispedricewafer
NA	NA	sugarpercent	hard
NA	NA	chocolate:peanutyalmondy	bar
NA	NA	peanutyalmondy:pluribus	pluribus
NA	NA	NA	sugarpercent
NA	NA	NA	pricepercent
NA	NA	NA	chocolate:peanutyalmondy
NA	NA	NA	chocolate:nougat
NA	NA	NA	chocolate:sugarpercent
NA	NA	NA	fruity:sugarpercent
NA	NA	NA	fruity:pricepercent
NA	NA	NA	caramel:nougat
NA	NA	NA	caramel:crispedricewafer
NA	NA	NA	caramel:bar
NA	NA	NA	caramel:sugarpercent
NA	NA	NA	caramel:pricepercent
NA	NA	NA	peanutyalmondy:bar
NA	NA	NA	peanutyalmondy:pluribus
NA	NA	NA	peanutyalmondy:sugarpercent
NA	NA	NA	nougat:sugarpercent
NA	NA	NA	nougat:pricepercent
NA	NA	NA	crispedricewafer:sugarpercent
NA	NA	NA	crispedricewafer:pricepercent
NA	NA	NA	bar:sugarpercent
NA	NA	NA	sugarpercent:pricepercent

From the table above, we can see that HPM included 6 variables, MPM included 7 vairiables, and BPM included 10 variables. All of them have significantly less variables than the AIC model.

j. If you were to design a new candy to optimize the winning percent, what features would it have? Create a prediction interval under BMA (not selection) for your designer candy and interpret.

```r
index.bpm <- predict(blm.candy, estimator = "BPM")$best
kable(data.frame(blm.candy$mle[[index.bpm]]) %>%
  `rownames<-`(BPM) %>%
  `colnames<-`("coefficients"),
  digits =3, caption = "Coefficients of BPM"
  )
```

Table 19: Coefficients of BPM

	coefficients
Intercept	50.317
chocolate	15.722
fruity	5.957
peanutyalmondy	-4.394
crispedricewafer	8.260
bar	2.721
pluribus	0.231
sugarpercent	6.500
chocolate:peanutyalmondy	16.541
peanutyalmondy:pluribus	1.394

```
newcandy = data.frame(t(rep(0,11))) %>%
  `colnames<-`(names(candy[-c(1,13)]))
newcandy[c("chocolate","fruity","peanutyalmondy",
           "crispedricewafer","bar","sugarpercent", "pluribus")] <- 1
BMA = predict(blm.candy, newcandy, estimator = "BMA", se.fit = TRUE)
BMA.conf.pred = confint(BMA, parm = "pred")
BMA.conf.pred
```

```
##          2.5%    97.5%     pred
## [1,] 42.81911 101.6774 71.80257
## attr(,"Probability")
## [1] 0.95
## attr(,"class")
## [1] "confint.bas"
```

```
BMA2 = predict(blm.candy, candy, estimator = "BMA", se.fit = TRUE)
BMA.conf.pred2 = confint(BMA2, parm = "pred")
kable(c(max(BMA.conf.pred2[,3]),candy$winpercent[which.max(BMA.conf.pred2[,3])]) %>% t(),
      col.names = c("Predicted Winpercent","Actual Winpercent"),
      caption = "Highest Predicted Winpercent of the Original Dataset")
```

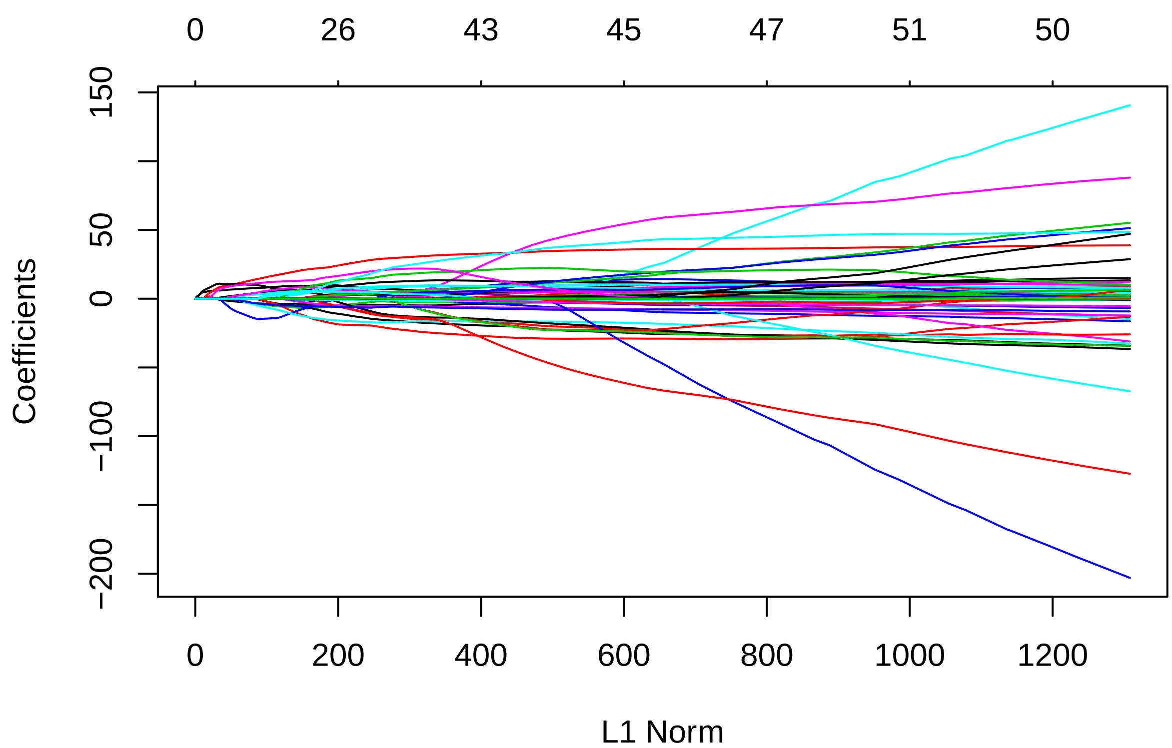
Table 20: Highest Predicted Winpercent of the Original Dataset

Predicted Winpercent	Actual Winpercent
70.7064	72.8879

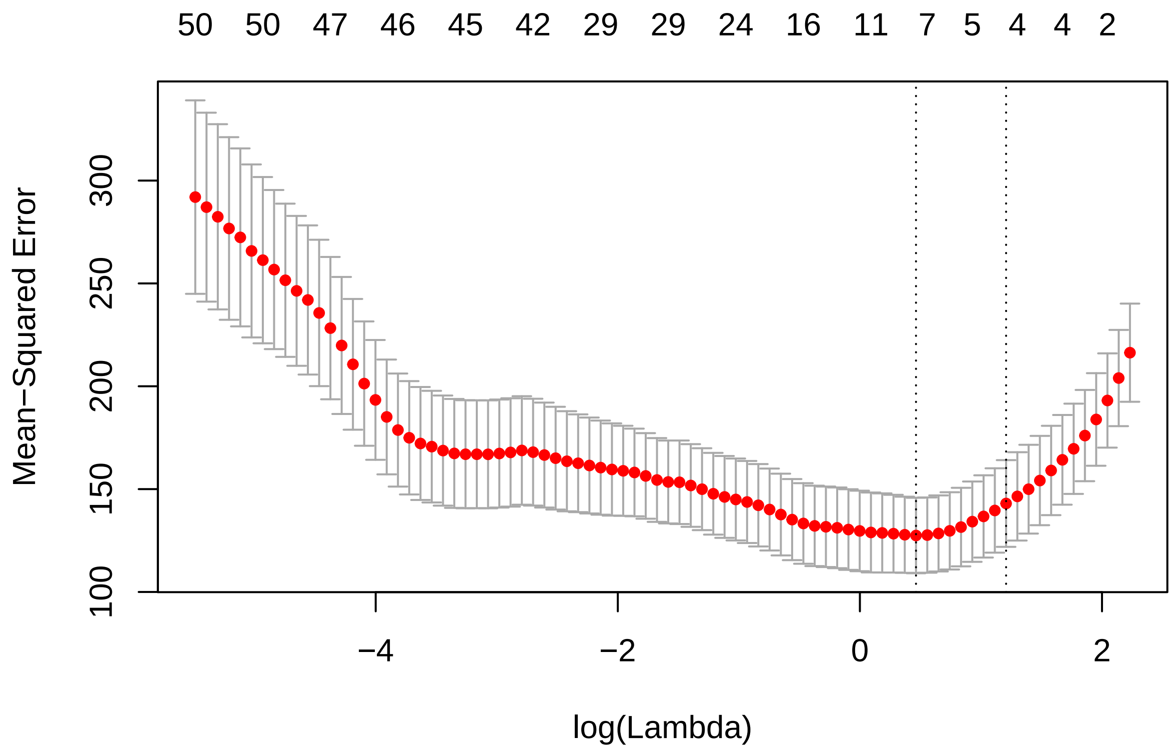
We designed our new candy based on the BPM. From the result of the BPM, we noticed that the only negative coefficient is related to peanutyalmondy, however, the interaction between chocolate and peanutyalmondy has a large positive coefficients. Therefore, we still decide to include it. The new candy included chocolate, fruity, peanutyalmondy, crispedricewafer, bar, pluribus , sugarpercent = 1. We then created the prediction interval of our new candy under BMA and compared the predicted winpercent of the new candy to the original dataset and confirmed that our new candy has higher predicted winpercent than all candies in the dataset.

k. Use the lasso to fit a model to the `winpercent`. Comment on which variables it identifies. How does this compare to the other results? (what is the optimal combination with the lasso?) Can you construct a prediction interval for the `winpercent` for this candy?

```
set.seed(3)
x = model.matrix(winpercent ~ (. -competitorname)^2, data = candy)
y = candy$winpercent
candy.lasso <- glmnet(x,y,
                      standardize=TRUE,
                      alpha = 1)
plot(candy.lasso)
```




```
cv.out=cv.glmnet(x,y,
                 alpha=1)
plot(cv.out)
```




```
bestlam=cv.out$lambda.min

lasso.coef = predict(candy.lasso,type="coefficients",s=bestlam)[1:67,]
lasso.coef[lasso.coef!=0] %>%
  kable(caption = "Coefficients of the Lasso Model",
        col.names = "coefficients",
        digits = 3)
```

Table 21: Coefficients of the Lasso Model

	coefficients
(Intercept)	43.719
chocolate	6.004
hard	-0.613
chocolate:caramel	0.547
chocolate:peanutyalmondy	8.101
chocolate:sugarpercent	10.884
crispedricewafer:bar	6.500

From the plot and table, we can see that the best model selected by lasso includes 6 variables: chocolate, hard, chocolate:caramel,chocolate:peanutyalmondy, chocolate:sugarpercent, crispedricewafer:bar. The only negative coefficient is related to hard. Therefore, the optimal combination with lasso should be chocolate + crispedricewafer + peanutyalmondy + caramel + bar. Sugarpercent should be high, and we set it to be 1.

```r
#Define rmse
rmse = function(ypred, ytest) {
  sqrt(mean((ypred-ytest)^2))
}

newcandy2 = data.frame(t(rep(0,11))) %>%
  `colnames<-`(names(candy[-c(1,13)]))
newcandy2[c("chocolate","peanutyalmondy",
           "crispedricewafer","caramel","bar","sugarpercent")] <- 1

# assign a number to winpercent so that model.matrix can run
# the value itself will not be used
newcandy2["winpercent"] <- 0.5
newcandyx <- model.matrix(winpercent ~ (.)^2-1, data = newcandy2)

prediction2 <- function(){
  sample.index <- sample(1:nrow(candy),size = nrow(candy), replace = TRUE)
  samp <- candy[sample.index,]
  model <- lm(winpercent ~ chocolate + hard + chocolate:caramel
              + chocolate:peanutyalmondy+ chocolate:sugarpercent
              + crispedricewafer:bar, data = samp)
  predict(model, newdata = newcandy2)
}


pred = rep(0,1000)
for (i in 1:1000){
  pred[i] <- prediction2()
}


lasso.pred=predict(candy.lasso,
                   newx= x,
                   s=bestlam)  # s = lambda
rmse.train = rmse(lasso.pred, y)

confint <- quantile(pred,probs = c(0.025,0.975))
predint <- c(confint[1]-1.96*rmse.train,confint[2]+1.96*rmse.train)
kable(predint %>% t(),digits = 3,
      caption = "Prediction Interval of the New Candy")
```

Table 22: Prediction Interval of the New Candy

2.5%	97.5%
52.174	125.746

The prediction interval for the `winpercent` of our new candy is (52.174,125.746). The interval is relatively wide and exceeds the highest possible value 100. However, this interval also indicates that our new candy has a high predicted winpercent.

l. Summarize your modeling efforts in a couple of paragraphs suitable for readers of 538, providing interpretation of coefficients and the interactions on how the they impact the winning percent and details about your optimal candy. (see the 538 blog linked above for inspiration!)

In this project, we used several different modeling methods to evaluate what combination of features of a candy will be able to make it more desirable. We first tried the simple linear regression model with interactions that are selected with AIC criterion. For the final model selected from AIC, we found that `nougat`, `crispedricewafer`, `caramel:nougat`, `sugarpercent:pricepercent`, `chocolate:peanutyalmondy` have coefficient intervals that do not include 0. This means we are 95% confident that these features have positive effect on win percent. Considering the fact even for some interaction term like `sugarpercent:pricepercent`, coefficient of `pricepercent` is negative, the absolute value is smaller than the interaction term, which can still be consider a desirable feature. However, since some of the coefficients are too big such that for adding this feature to the candy, we will have a huge increase of winpercent that exceeds the range of (0,100). Thus, this indicates that the model provides a relatively poor estimate for certain feature.

Then, we moved to the Bayesian models, which have generally smaller intervals for variables. Using Best Probability Model, we eventually decided on what combination of features will be desirable. We observed only one variable with negative coefficient in this model. However, with the combination of other features, we can see that adding this feature to the candy will have an increase of $-4.394 + 16.541 + 1.394 = 13.541$ on `winpercent`. All other variables have positive coefficient, which means adding these features can lead to `higher winpercent`. To be more specific, we can see from the summary table that adding fruity flavor with all other variables fixed will increase the winpercent by 5.957 percentage. Similarly for other features in this model. So, we eventually decided that candies that contains chocolate, peanuts, peanut butter or almonds, crisped rice, wafers, or a cookie component , higher sugar percentile and they are fruity favored, form as a candy bar and present as one of many candies in bag or box are generally more desirable.

Lastly, we selected our model using lasso method. The lasso model suggests that the overall win percentage is significantly influenced by `chocolate`, `hard`, and interaction terms `chocolate:caramel`, `chocolate:peanutyalmondy`, `chocolate:sugarpercent`, `crispedricewafer:bar`. A more detailed explanation corresponds to the table of the lasso model could be: Holding all other variables constant,
If a candy contain chocolate, the overall win percentage of the candy will increase by 6.004%.
If a candy is hard, the overall win percentage of the candy will decrease by 0.613%.
If a candy contains both chocolate and caramel, the wining percentage will increase by 0.547%.
If a candy contains both chocolate and peanuts/peanut butter/almonds, the wining percentage will increase by 8.101%.
If the sugarpercent increase by 10%, the wining percentage of a candy with chocolate will be 1.0884% higher than that of a candy without chocolate.
If a candy contains chrisped rice/wafers/cookie component and it is a candy bar, the wining percentage will increase by 6.5%.
Based on this information, we designed another new candy, which contains chocolate, crisped rice/wafers/cookie component, peanuts/peanut butter/almonds, caramel and formed it as a candy bar. Our predicted winning percentage of this candy will between 52.174 and 125.746. Since a winning percentage of 125.746 is not possible, we can treat it as (52.134,100). In general, we are 95% confident that this candy will have an overall win percentage above 52.134%.