require(rjags)

require(mgcv)

load.module("glm")

require(scatterplot3d)

require(loo)

library(rstanarm)

setwd("C:/R files BHMRA")

D <- read.table("DS\_12\_2.txt",header=T)

**# model 1, single rank 20 thin plate regression spline in gdp**

jd1 <- jagam(TFR ~ s(gdp, k=20), data =D,family = gaussian, file = "TFR.jags")

jm1 <- jags.model("TFR.jags", data = jd1$jags.data, inits = jd1$jags.ini, n.chains = 2)

sam1 <- jags.samples(jm1, c("b", "rho", "mu"), n.iter = 2500)

jam1 <- sim2jam(sam1, jd1$pregam)

plot(jam1, shade = T, ylab=expression(s(gdp)),xlab="GDP Per Capita")

coda.sam1 <- coda.samples(jm1,c("b","rho","mu"),n.iter=2500)

gelman.diag(coda.sam1,multivariate=F)

summary(coda.sam1)

dic.samples(jm1, n.iter=1000,type="pD")

**# model 2, two separate smooths in gdp and fschool**

jd2 <- jagam(TFR ~ s(gdp, k=20)+s(fschool,k=20), data =D,family = gaussian, file = "TFR.jags")

jm2 <- jags.model("TFR.jags", data = jd2$jags.data, inits = jd2$jags.ini, n.chains = 2)

sam2 <- jags.samples(jm2, c("b", "rho", "mu"), n.iter = 2500)

jam2 <- sim2jam(sam2, jd2$pregam)

plot(jam2, shade = T)

coda.sam2 <- coda.samples(jm2,c("b","rho","mu"),n.iter=2500)

summary(coda.sam2)

gelman.diag(coda.sam2,multivariate=F)

dic.samples(jm2, n.iter=1000,type="pD")

**# model 2, stan\_gamm4**

S <- stan\_gamm4(TFR ~ s(gdp,k=20) + s(fschool,k=20), data = D,

family = gaussian, prior = student\_t(df = 10),

prior\_intercept = student\_t(df = 10), chains = 2, cores = 1, seed = 1234)

summary(S)

plot\_nonlinear(S)

**# model 3, tensor product only**

jd3 <- jagam(TFR ~ te(gdp, fschool), data =D,family = gaussian, file = "TFR.jags")

jm3 <- jags.model("TFR.jags", data = jd3$jags.data, inits = jd3$jags.ini, n.chains = 2)

sam3 <- jags.samples(jm3, c("b", "rho", "mu"), n.iter = 5000)

jam3 <- sim2jam(sam3, jd3$pregam)

plot(jam3, shade = T)

coda.sam3 <- coda.samples(jm3,c("mu"),n.iter=2500)

summary(coda.sam3)

gelman.diag(coda.sam2,multivariate=F)

array <- as.array(coda.sam3)

mu <- numeric(167)

for (i in 1:167) {mu[i] <- mean(array[,i,])}

**# 3-D scatterplot**

scatterplot3d(D$gdp,D$fschool,mu, highlight.3d=T, zlab=expression(s(gdp,fschool)),ylab="Female Education (Years)",xlab="GDP Per Capita")

dic.samples(jm3, n.iter=1000,type="pD")

**# model 4, separate smooths plus tensor product**

jd4 = jagam(TFR ~ s(gdp, k=20)+s(fschool,k=20)+te(gdp, fschool), data =D,family = gaussian,

file = "TFR.jags")

jm4 <- jags.model("TFR.jags", data = jd4$jags.data, inits = jd4$jags.ini, n.chains = 2)

sam4 <- jags.samples(jm4, c("b", "rho", "mu"), n.iter = 2500)

jam4 <- sim2jam(sam4, jd4$pregam)

plot(jam4, shade = T)

coda.sam4 <- coda.samples(jm4,c("b","mu","lambda","tau"),n.iter=2500)

summary(coda.sam4)

gelman.diag(coda.sam4,multivariate=F)

array <- as.array(coda.sam4)

mu <- numeric(167)

for (i in 1:167) {mu[i] <- mean(array[,i,])}

scatterplot3d(D$gdp,D$fschool,mu, highlight.3d=T, zlab=expression(s(gdp,fschool)),ylab="Female Education (Years)",xlab="GDP Per Capita",main="3D Scatterplot")

dic.samples(jm4, n.iter=1000,type="pD")

**# model 5, model 4 with selection of smooth components**

cat(" model { for (i in 1:n) { y[i] ~ dnorm(mu[i],tau)

mu[i] <- b[1]+sum(beta.terms[i,2:61])

for (j in 2:20) {beta.terms[i,j] <- X[i,j]\*b[j]\*J[1]}

for (j in 21:39) {beta.terms[i,j] <- X[i,j]\*b[j]\*J[2]}

for (j in 40:61) {beta.terms[i,j] <- X[i,j]\*b[j]\*J[3]}

LL[i] <- log(L[i])

log(L[i]) <- -0.5\*log(6.283\*s2)-0.5\*tau\*pow(y[i]-mu[i],2)}

# variance and precision (PC prior on s)

s2 <- s\*s

tau <- 1/s2

s ~ dexp(2.3)

b[1] ~ dnorm(0,0.0001)

# prior for s(gdp)

K1 <- S1[1:19,1:19] \* lambda[1] + S1[1:19,20:38] \* lambda[2]

b[2:20] ~ dmnorm(zero[2:20],K1)

# prior for s(fschool)

K2 <- S2[1:19,1:19] \* lambda[3] + S2[1:19,20:38] \* lambda[4]

b[21:39] ~ dmnorm(zero[21:39],K2)

# prior for te(gdp,fschool)

K3 <- S3[1:22,1:22] \* lambda[5] + S3[1:22,23:44] \* lambda[6] + S3[1:22,45:66] \* lambda[7]

b[40:61] ~ dmnorm(zero[40:61],K3)

# selection indicators

for (k in 1:3) {J[k] ~ dbern(0.5)}

# smoothing parameters

for (i in 1:7) { lambda[i] ~ dexp(1)

rho[i] <- log(lambda[i]) }}

", file="TFRcode.jags")

# estimation

jm5 <- jags.model("TFRcode.jags", data = jd4$jags.data, inits = jd4$jags.ini, n.chains = 2)

coda.sam5 <- coda.samples(jm5, c("J","L","LL"),n.iter=2500)

summary(coda.sam5)

array5 <- as.array(coda.sam5)

n <- jd4$jags.data$n

**# OBTAIN pDIC, WAIC and LOO-IC**

dic.samples(jm5, n.iter=1000,type="pD")

c1 <- c2 <- numeric(n)

for (i in 1:n) {c1[i] <- log(mean(array5[,i,])); c2[i] <- sd(array5[,i+n,])^2}

WAIC <- -2\*(sum(c1)-sum(c2))

WAIC

T <- 2500

LL <- matrix(NA,2\*T,n)

for (i in 1:T){for (j in 1:n) {LL[i,j] <- array5[j,j+n,1]; LL[i+T,j] <- array5[i, j+n,2]}}

loo(LL)