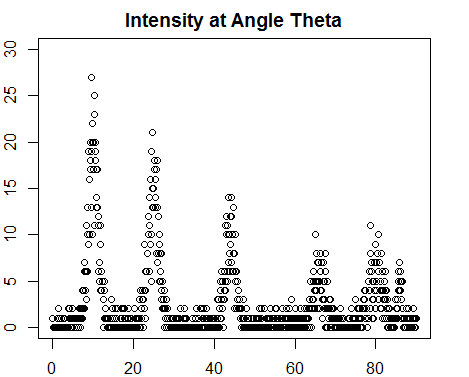
**Introduction**

X-ray diffraction is a technique for characterizing the atomic structure of materials. An X-ray beam is directed through a material, and the X-rays are diffracted at various angles which can be interpreted to measure the composition and properties of the substance. Diffraction occurs when crystalline atoms cause of beam of incident X-rays to diffract at angle θ into many directions resulting in different measures of intensity as in Figure 1. The locations of the peaks reveal the composition of the material and is of particular interest in this study.

*Figure 1. Example peaks observed in X-ray diffraction pattern. Constructive interference results in observed peaks at scatter angle θ.*

**Data and Model Considerations**

The data was collected from a standard X-ray crystallography experiment. The data from the study was loaded into R and contains paired observations of diffraction angle θ and observed intensity Y. The data is sorted by θ in ascending order. In total, 1000 observations of θ and the intensity were measured resulting in a 1000 X 2 matrix.

A Poisson model was used to fit the data with intensity as the response. The rate of occurrence was modeled by the function λ(θ) where λ(θ) was a linear combination of the background intensity, Gaussian peak functions, and peak intensities. Equations below show the modelled relationship. A Bayesian approach was used to model the data.

Y ∼ Poisson[λ(θ)] *Equation 1*

λ(θ) = α0 + ∑φ(θ; γj , τj )βj *Equation 2*

φ(θ; γ, τ) = exp[− (θ−γ)2/2τ2] *Equation 3*

Where,

Y = Intensity of the peaks

θ = Diffraction angle

α0 = Background intensity

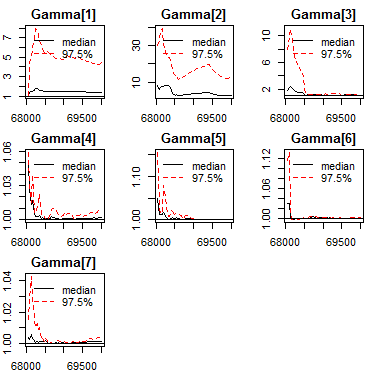
γj = Peak location

τj = Peak width

βj = Peak intensity

Since, predetermined priors could not be assessed for the parameters in the model, three different models were fit to the data with assumed weak priors βj, α0, τj ~ Gamma(1,1) and γj ~ Uniform(0,90). The three models differed in the number of proposed peaks. The data revealed a minimum of five and no more than seven peaks. J was specified as {5,6,7} accordingly. Model 1 specified 5 peaks, Model 2 specified 6 peaks, and Model 3 specified 7 peaks.

**Convergence and Model Selection**

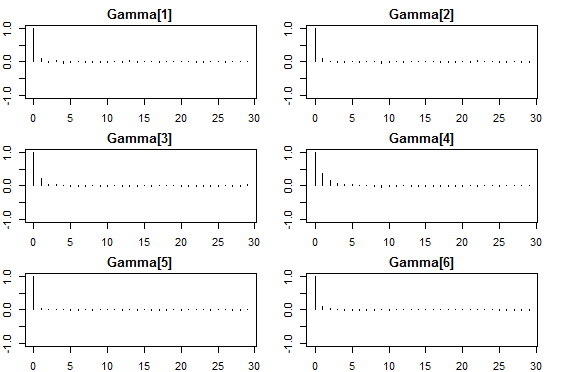
All models were fit in R using JAGS and MCMC sampling. The first 20,000 iterations were discarded as burn in samples and the next 5,000 samples were generated for estimating the posterior of the model parameters. For Model 3, more iterations were burned into attempt to induce convergence. For selecting one of the aforementioned models, convergence criteria was used first to eliminate one of the models. Examining selected Gelman plots for Model 3, we note the first three

*Figure 2. Shrink Factor Plots for Model 3 γj Coefficients*

estimates for γj do not approach one. We also see that some trace plots showing the generation of the posterior distribution did not resemble the white noise process which confirms the lack of convergence in Model 3. Therefore, Model 3 was eliminated from consideration. Table 1 shows the DIC values for Model 1 and Model 2. The DIC values varied little for the two models, but Model 2 was selected to be the final model as the penalized deviance was less than the penalized deviance for Model 1.

|  |  |  |
| --- | --- | --- |
|  | Model 1 (J=5) | Model 2 (J=6) |
| Mean Deviance | 2887 | 2828 |
| Penalty | 15.61 | 71.49 |
| Penalized Deviance | 2903 | 2900 |

*Table 1: DIC for the Two Models*

Diagnostics were performed in order to check if the MCMC algorithm was producing a reliable output for the final model. The Gelman and Rubins shrink factor plots tend to a value of 1 as the number of iterations reached 30,000 which is evidence for convergence of the model. The trace plots for all the parameters resembled white noise which confirmed that conclusion. The effective sample sizes for each parameter were no less than 5825.146 which suggests that the samples were not highly correlated. The Autocorrelation plots were generated to confirm this. Figure 3 shows that the ACF decays after a lag of 5 and is insignificant further supporting the assumption of independence. Thus it can be assumed that the MCMC algorithm produced reliable output for the model parameters.

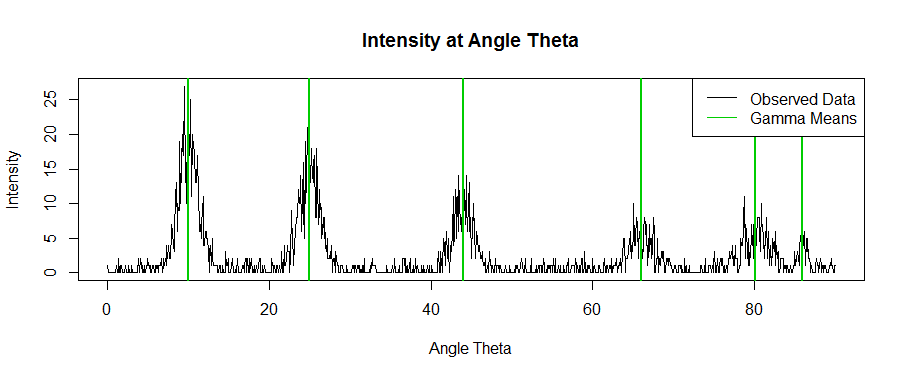
*Figure 3. Autocorrelation plots for* γj.

**Results and Predictions**

The output from the MCMC sampling algorithm was analyzed for determining the locations of the peaks (γj). The 2.5% and 97.5% quantile values for the posterior distribution of all the parameters were examined to determine 95% credible sets. The posterior mean was evaluated to determine a point estimate of γj. None of the credible sets overlap further supporting the proposed model with 6 peaks.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Mean | 2.5% Quant. | 97.5% Qaunt. | 95% Credible Set |
| γ1 | 10.00 | 9.892 | 10.11 | (9.892,10.11) |
| γ2 | 24.92 | 24.796 | 25.05 | (24.796,25.05) |
| γ3 | 43.96 | 43.809 | 44.12 | (43.809,44.12) |
| γ4 | 66.01 | 65.751 | 66.26 | (66.751,66.26) |
| γ5 | 80.13 | 79.876 | 80.39 | (79.876,80.39) |
| γ6 | 85.97 | 85.813 | 86.12 | (85.813,86.12) |

*Table 2. Posterior Distribution Quantiles and Credible Sets*

To further illustrate the validity of the model, the means for each γj were plotted over the original data. Each E(γj) occurs approximately at the center for each peak noted in the observed data. This demonstrates that the model estimates fit the data well. Based on this, we have evidence that the appropriate number of peaks is six.

*Figure 4. Gamma means overlaid on the original data.*

**Summary of Results**

The intensity of constructive interference was modeled using a Bayesian approach and an MCMC algorithm. Intensity was treated as the response with a function of angle θ as the predictor. The model used Gamma and Uniform priors and MCMC sampling was employed to generate the posterior distribution for nineteen parameters. The parameter estimates for γj were returned and examined. Given that the model converged to six different values and none of the prediction intervals overlapped. In other words, there are six different peaks in the data that can be modeled accurately.

**Appendix:**

**setwd("C:/Users/debri/Downloads")**

**data = read.csv("exam2\_data.csv")**

**library(rjags)**

**Y = data$Y**

**theta = data$theta**

**n = length(Y)**

**model = "model{**

**# Likelihood**

**for(i in 1:n){**

**Y[i] ~ dpois(lambda[i])**

**}**

**for(i in 1:n){**

**lambda[i] = alpha + inprod(phi[i,],beta[])**

**for(j in 1:p){**

**phi[i,j] = exp(-0.5\*(theta[i]-gamma[j])^2/(tau[j]^2))**

**}**

**}**

**#Prior for alpha**

**alpha ~ dgamma(1, 1)**

**#Prior for Betas**

**for(j in 1:p){**

**beta[j] ~ dgamma(1, 1)**

**}**

**#Prior for Gamma and Tau**

**for(j in 1:p){**

**gamma[j] ~ dunif(0,90)**

**tau[j] ~ dgamma(1,1)**

**}**

**Gamma[1:p] = sort(gamma)**

**}"**

**model0 <- jags.model(textConnection(model), n.chains=2,**

**data = list(Y=Y, theta=theta, n=n, p = 5))**

**update(model0, 1000)**

**dic1 <- dic.samples(model0,**

**variable.names=c("Gamma"),**

**n.iter=1000, progress.bar="text")**

**samps1 <- coda.samples(model0,**

**variable.names=c("Gamma"),**

**n.iter=1000, progress.bar="text")**

**par(mar = rep(2, 4))**

**autocorr.plot(samps1)**

**effectiveSize(samps1)**

**gelman.plot(samps1)**

**plot(samps1)**

**summary(samps1)**

**dic1**

**model = "model{**

**# Likelihood**

**for(i in 1:n){**

**Y[i] ~ dpois(lambda[i])**

**}**

**for(i in 1:n){**

**lambda[i] = alpha + inprod(phi[i,],beta[])**

**for(j in 1:p){**

**phi[i,j] = exp(-0.5\*(theta[i]-gamma[j])^2/(tau[j]^2))**

**}**

**}**

**#Prior for alpha**

**alpha ~ dgamma(1, 1)**

**#Prior for Betas**

**for(j in 1:p){**

**beta[j] ~ dgamma(1, 1)**

**}**

**#Prior for Gamma and Tau**

**for(j in 1:p){**

**gamma[j] ~ dunif(0,90)**

**tau[j] ~ dgamma(1,1)**

**}**

**Gamma[1:p] = sort(gamma)**

**}"**

**model1 <- jags.model(textConnection(model), n.chains=2,**

**data = list(Y=Y, theta=theta, n=n, p = 6))**

**update(model1, 5000)**

**dic2 <- dic.samples(model1,**

**variable.names=c("Gamma"),**

**n.iter=2000, progress.bar="text")**

**samps2 <- coda.samples(model1,**

**variable.names=c("Gamma"),**

**n.iter=5000, progress.bar="text")**

**par(mar = rep(2, 4))**

**autocorr.plot(samps2)**

**effectiveSize(samps2)**

**gelman.plot(samps2)**

**plot(samps2)**

**summary(samps2)**

**dic2**

**model = "model{**

**# Likelihood**

**for(i in 1:n){**

**Y[i] ~ dpois(lambda[i])**

**}**

**for(i in 1:n){**

**lambda[i] = alpha + inprod(phi[i,],beta[])**

**for(j in 1:p){**

**phi[i,j] = exp(-0.5\*(theta[i]-gamma[j])^2/(tau[j]^2))**

**}**

**}**

**#Prior for alpha**

**alpha ~ dgamma(1, 1)**

**#Prior for Betas**

**for(j in 1:p){**

**beta[j] ~ dgamma(1, 1)**

**}**

**#Prior for Gamma and Tau**

**for(j in 1:p){**

**gamma[j] ~ dunif(0,90)**

**tau[j] ~ dgamma(1,1)**

**}**

**Gamma[1:p] = sort(gamma)**

**}"**

**model2 <- jags.model(textConnection(model), n.chains=2,**

**data = list(Y=Y, theta=theta, n=n, p = 7))**

**update(model2, 20000)**

**dic3 <- dic.samples(model2,**

**variable.names=c("Gamma"),**

**n.iter=2000, progress.bar="text")**

**samps3 <- coda.samples(model2,**

**variable.names=c("Gamma"),**

**n.iter=2000, progress.bar="text")**

**par(mar = rep(2, 4))**

**autocorr.plot(samps3)**

**effectiveSize(samps3)**

**gelman.plot(samps3)**

**plot(samps3)**

**summary(samps3)**

**dic3**

**plot(x = NULL, y = NULL, xlab="Angle Theta", ylab="Intensity", main="Intensity at Angle Theta", xlim=c(0,90),ylim=c(0,30))**

**points(x=theta, y=Y)**

**samps = samps2[[1]]**

**gamma.mn = colMeans(samps)**

**plot(x = theta, y = Y, xlab="Angle Theta", ylab= "Intensity", main="Intensity at Angle Theta",type="l")**

**for(i in 1:6){**

**abline(v=gamma.mn[i],col=3,lwd=2)**

**}**

**legend("topright", c("Observed Data", "Gamma Mea**