Low-Mg Carbonate XRD synthetic data analysis, 0-100% range calcite

John M. Jaeger

May 25, 2017

## code description

This code is designed to import and analyze synthetic X-ray diffraction data modeled at the University of Florida using Materials Data Inc. Jade v.9.1.1 software. The data are synthetic mixtures of aragonite (PDF File #00-041-1475) and calcite (PDF File ##00-005-0586). Mass ratios of mineral phases are set in Jade and peak intensities are then calculated for these rations based on the I/Ic RIR values for each mineral phase and peak areas are modeled using a psedo-Voight shape as set in Jade. See Methods and Supplementary File ## for additional details on how synthetic data were generated.

This code analyzes a full range of synthetic calcite/aragonite mixtures.

Code generated by: John M. Jaeger Associate Professor 241 Williamson Hall P.O. Box 112120 Dept. of Geological Sciences University of Florida Gainesville FL 32611-2120, USA (352) 846-1381 ORCID ID# orcid.org/0000-0003-0248-489X <http://people.clas.ufl.edu/jmjaeger/>

### Load packages

# if necessary uncomment and install packages.  
# install.packages("dplyr")  
# install.packages("knitr")  
library(dplyr)  
require(knitr)  
library(ezknitr)  
library(ezknitr)  
library(ggplot2)  
library(splines)  
#library(propagate)

Import the data:

# load the End-member Dataset and give the dataset a shorter name  
  
XRD.data <- read.table("../raw\_data/UF\_LMC\_syndata\_20170808.csv",header=T,sep=",")   
XRDdata <- tbl\_df(XRD.data)

Calculate peak area ratios:

area.ratio <- XRDdata$LMC\_area/(XRDdata$LMC\_area+XRDdata$Aragonite\_area)  
LMC.conc <- XRDdata$LMCconc

Develop polynomial calibration model for low-Mg calcite:

model.fit3<- lm(LMC.conc~ poly(area.ratio,3))  
summary(model.fit3)

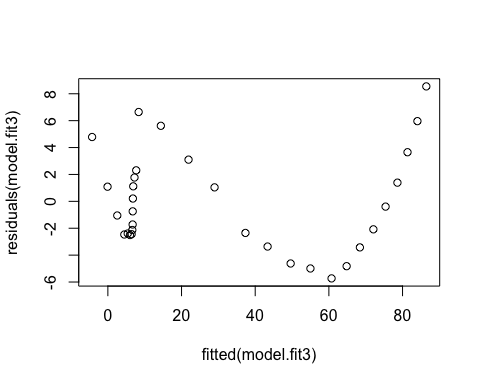
##   
## Call:  
## lm(formula = LMC.conc ~ poly(area.ratio, 3))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -5.7303 -2.4541 -0.7448 2.0393 8.5472   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 32.2581 0.7022 45.940 < 2e-16 \*\*\*  
## poly(area.ratio, 3)1 157.0883 3.9096 40.180 < 2e-16 \*\*\*  
## poly(area.ratio, 3)2 60.8930 3.9096 15.575 5.16e-15 \*\*\*  
## poly(area.ratio, 3)3 31.4570 3.9096 8.046 1.20e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 3.91 on 27 degrees of freedom  
## Multiple R-squared: 0.9861, Adjusted R-squared: 0.9846   
## F-statistic: 640.6 on 3 and 27 DF, p-value: < 2.2e-16

# Confidence intervales for model parameters  
confint(model.fit3, level=0.95)

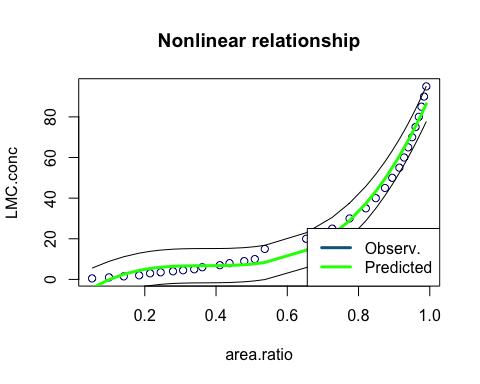
## 2.5 % 97.5 %  
## (Intercept) 30.81731 33.69882  
## poly(area.ratio, 3)1 149.06654 165.11011  
## poly(area.ratio, 3)2 52.87126 68.91483  
## poly(area.ratio, 3)3 23.43518 39.47875

Plot data and model:

# Plot of fitted vs residuals  
# No clear pattern should show in the residual plot if the model is a good fit  
plot(fitted(model.fit3),residuals(model.fit3))



# Predicted values and confidence intervals  
polypredicted.intervals <- predict(model.fit3,data.frame(x=area.ratio),interval='prediction',  
 level=0.95)  
  
# Add lines to the existing plot  
plot(area.ratio,LMC.conc,type='p',col='navy',main='Nonlinear relationship',lwd=1)  
  
lines(area.ratio,polypredicted.intervals[,1],col='green',lwd=3)  
lines(area.ratio,polypredicted.intervals[,2],col='black',lwd=1)  
lines(area.ratio,polypredicted.intervals[,3],col='black',lwd=1)  
  
# Add a legend  
legend("bottomright",c("Observ.","Predicted"),   
 col=c("deepskyblue4","green"), lwd=3)



The next step is to develop uncertainty in % LMC for a given area ratio, third-order polynomial.

# Predicted values and confidence intervals  
testvalue = data.frame(area.ratio=0.1)  
predicted.value <- predict(model.fit3,testvalue,interval='prediction', level=0.95)  
predicted.value

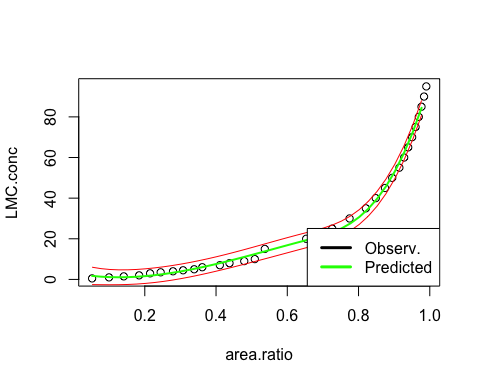
## fit lwr upr  
## 1 -0.0594941 -9.032527 8.913539

Now a spline model: # <https://www.r-bloggers.com/splines-opening-the-black-box/> # <http://stackoverflow.com/questions/15837763/b-spline-confusion> # <https://www.rdocumentation.org/packages/splines2/versions/0.2.4/topics/predict>

library(splines)  
spline1.pred <- lm(LMC.conc ~ bs(area.ratio, df=4))#df=4 means knots at end points and quantiles of X  
summary(spline1.pred)

##   
## Call:  
## lm(formula = LMC.conc ~ bs(area.ratio, df = 4))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.4620 -1.1388 -0.0358 1.0841 3.3875   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.729 1.285 1.346 0.190   
## bs(area.ratio, df = 4)1 -3.679 2.983 -1.233 0.228   
## bs(area.ratio, df = 4)2 15.210 2.537 5.995 2.49e-06 \*\*\*  
## bs(area.ratio, df = 4)3 26.904 2.133 12.614 1.38e-12 \*\*\*  
## bs(area.ratio, df = 4)4 89.883 1.466 61.313 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.638 on 26 degrees of freedom  
## Multiple R-squared: 0.9977, Adjusted R-squared: 0.9973   
## F-statistic: 2769 on 4 and 26 DF, p-value: < 2.2e-16

plot(area.ratio,LMC.conc)  
u=seq(min(area.ratio),max(area.ratio),by=.025)  
B=data.frame(area.ratio=u)  
splinepredicted.interval=predict(spline1.pred,newdata=B,interval ="prediction",level=0.95)  
lines(u,splinepredicted.interval[,1],lwd=2,col="green")  
lines(u,splinepredicted.interval[,2],col='red',lwd=1)  
lines(u,splinepredicted.interval[,3],col='red',lwd=1)  
  
# Add a legend  
legend("bottomright",c("Observ.","Predicted"),   
 col=c("black","green"), lwd=3)

 The last step is to develop uncertainty in % LMC for a given area ratio, spline model.

# Predicted values and confidence intervals  
predicted.splinevalue <- predict(spline1.pred,testvalue,interval='prediction', level=0.95)  
predicted.splinevalue

## fit lwr upr  
## 1 1.140364 -2.631956 4.912683

Export fits and prediction intervals:

polyout <- cbind(area.ratio,polypredicted.intervals)  
splineout <- cbind(u,splinepredicted.interval)  
write.csv(polyout, file="../reports/UF\_syn\_polyfit.csv")  
write.csv(splineout, file="../reports/UF\_syn\_splinefit.csv")