AMRC: Exercise 8 - 11912007

Analysis

The analysis explores the relationship between temperature and ozone concentration using spline regression techniques. The initial approach uses a training-test split of 2/3 to 1/3, with spline basis functions constructed using 2 knots and order M=4.

The visualization of basis functions reveals smooth polynomial components that capture the underlying relationship between temperature and ozone. When fitting the model to the training data, the predictions follow a reasonable smooth curve after properly sorting the temperature values in ascending order.

The comparison between training and test predictions demonstrates good consistency, indicating that the model generalizes well to unseen data. However, when attempting to extend predictions beyond the original temperature range (0-120°C), the initial model produces unrealistic results due to the dynamic placement of knots based on quantiles of the input data.

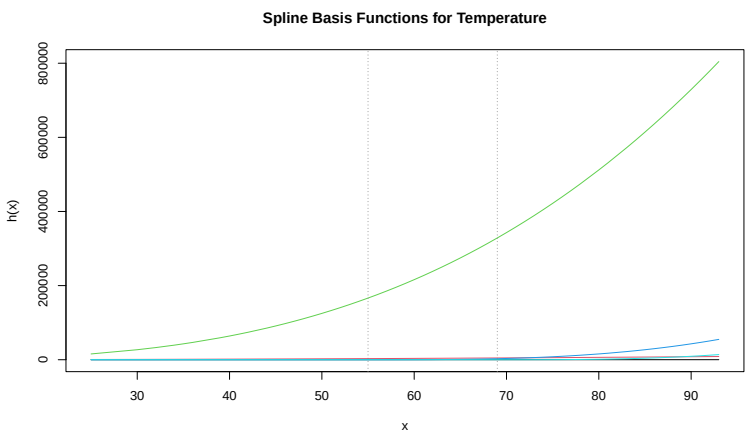
This issue was addressed by modifying the spline function to use fixed knots from the training data, resulting in more stable and sensible predictions for extended temperature ranges. The fixed-knot approach maintains the model's predictive performance while ensuring consistent behavior across different temperature ranges.

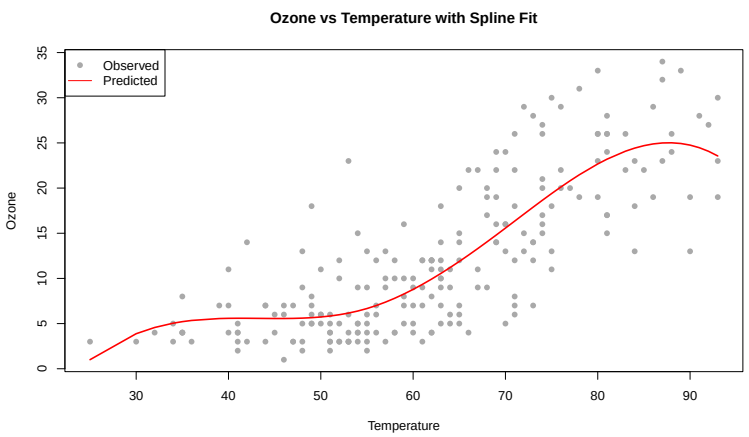
A significant improvement was achieved by addressing the physically impossible negative ozone predictions at low temperatures. The log-transformation of the response variable, followed by exponential transformation of the predictions, successfully constrains the predictions to positive values while maintaining the smooth nature of the spline fit.

The final model, incorporating both fixed knots and log-transformation, provides a more realistic representation of the temperature-ozone relationship, particularly at the extremes of the temperature range. The visualizations clearly show the improvement in prediction quality, with the model maintaining physical constraints while capturing the underlying nonlinear relationship.

Code

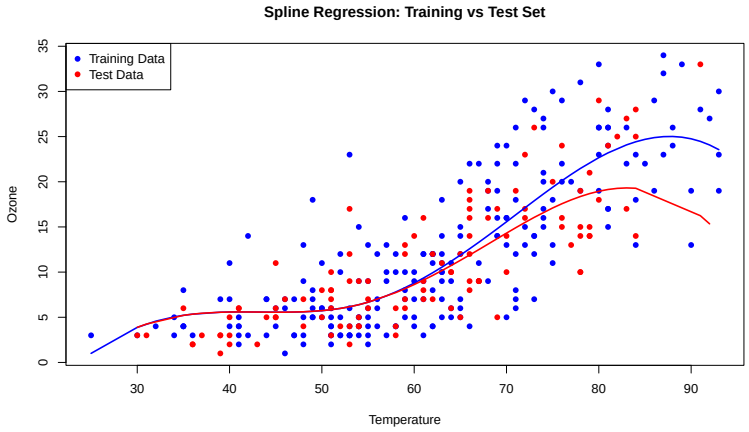
```r
set.seed(42)

data(ozone) # from gclus package

lecturespl <- function(x, nknots=2, M=4) { # from course notes
    n <- length(x)
    # X will not get an intercept column
    X <- matrix(NA, nrow=n, ncol=(M-1)+nknots)
    for (i in 1:(M-1)) {
        X[,i] <- x^i
    }
    # basis functions for the constraints
    quant <- seq(0,1,1/(nknots+1))[c(2:(nknots+1))]
    qu <- quantile(x, quant)
    for (i in M:(M+nknots-1)) {
        X[,i] <- ifelse(x-qu[i-M+1]<0, 0, (x-qu[i-M+1])^(M-1))
    }
    list(X=X, quantiles=quant, xquantiles=qu, x=x)
}

plotspl <- function(splobj,...){ # from assignment
    matplot(splobj$x, splobj$X, type="l", lty=1, xlab="x", ylab="h(x)", ...)
    abline(v=splobj$xquantiles, lty=3, col=gray(0.5))
}

# 2/3 holdout split
sample_idx <- sample(c(TRUE, FALSE), nrow(ozone), replace=TRUE, prob=c(2/3, 1/3))
train_data <- ozone[sample_idx, ]
test_data <- ozone[!sample_idx, ]

# sort data by temperature
train_data <- train_data[order(train_data$Temp), ]
test_data <- test_data[order(test_data$Temp), ]

# generate spline basis functions
spl_train <- lecturespl(train_data$Temp, nknots=2, M=4)
spl_test <- lecturespl(test_data$Temp, nknots=2, M=4)

# fit linear model using spline basis
X <- spl_train$X
y <- train_data$Ozone

train_df <- data.frame(X) # make sure column names match
colnames(train_df) <- paste0("X", 1:ncol(X))
fit <- lm(y ~ ., data=train_df)
train_pred <- predict(fit)

test_df <- data.frame(spl_test$X) # make sure column names match
colnames(test_df) <- paste0("X", 1:ncol(spl_test$X))
test_pred <- predict(fit, newdata=test_df)

# plot basis functions, data, train preds
par(mfrow=c(1,2))
plotspl(spl_train, main="Spline Basis Functions for Temperature")
plot(train_data$Temp, train_data$Ozone, main="Ozone vs Temperature with Spline Fit", xlab="Temperature", ylab="Ozone", pch=16, col="darkgray")
lines(train_data$Temp, train_pred, col="red", lwd=2)
legend("topleft", legend=c("Observed", "Predicted"), col=c("darkgray", "red"),  pch=c(16, NA), lty=c(NA, 1))
```

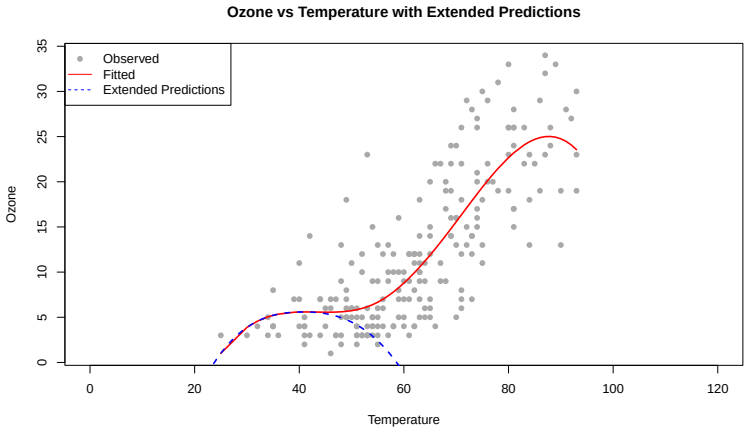




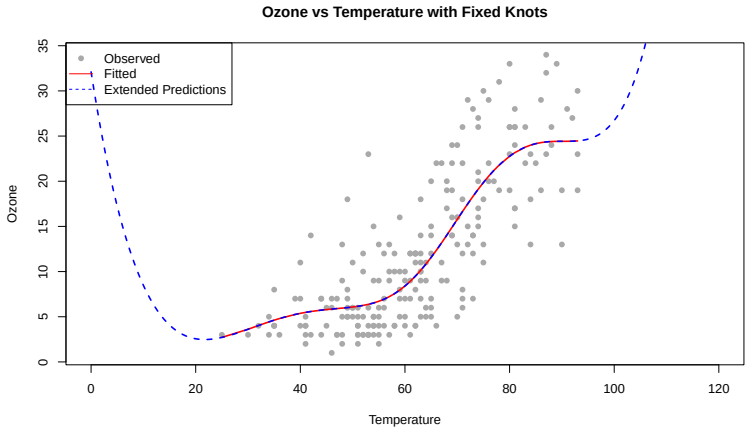

```r
# plot train vs. test preds
plot(train_data$Temp, train_data$Ozone, col = "blue", pch = 16, xlab = "Temperature", ylab = "Ozone", main = "Spline Regression: Training vs Test Set")
points(test_data$Temp, test_data$Ozone, col = "red", pch = 16)
lines(train_data$Temp, train_pred, col = "blue", lwd = 2)
lines(test_data$Temp, test_pred, col = "red", lwd = 2)
legend("topleft", legend = c("Training Data", "Test Data"), col = c("blue", "red"), pch = 16)
```

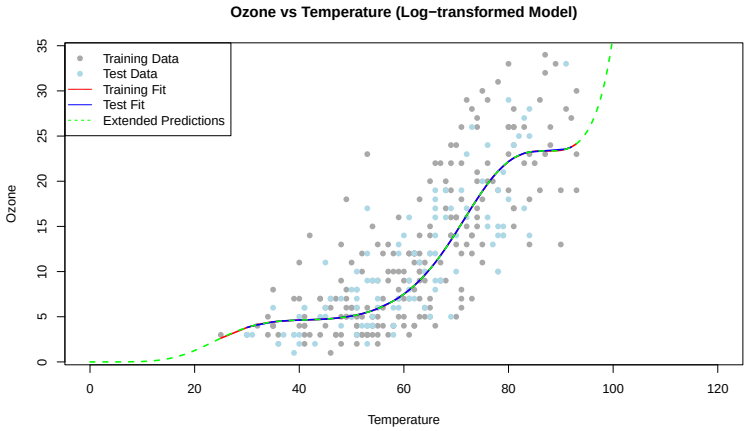
```r
#
# create new data
#

new_temp <- seq(0, 120, length.out = 200) # extended predictions
spl_new <- lecturespl(new_temp, nknots=2, M=4) # new splines

# the columns should be named X1, X2, X3, etc. to match the original model
newdata <- as.data.frame(spl_new$X)
names(newdata) <- paste0("X", 1:ncol(newdata))

new_pred <- predict(fit, newdata=newdata)

# set extended x-axis limits
plot(train_data$Temp, train_data$Ozone, main="Ozone vs Temperature with Extended Predictions", xlab="Temperature", ylab="Ozone", pch=16, col="darkgray", xlim=c(0, 120))
lines(train_data$Temp, train_pred, col="red", lwd=2) # training data predictions
lines(new_temp, new_pred, col="blue", lwd=2, lty=2) # extended range predictions
legend("topleft", legend=c("Observed", "Fitted", "Extended Predictions"), col=c("darkgray", "red", "blue"), pch=c(16, NA, NA), lty=c(NA, 1, 2))
```






```r
#
# fix lecturesppl
#

lecturespl_fixed <- function(x, nknots, M, knots = NULL) { # fix: accept pre-computed knots
    if (is.null(knots)) {
        # If no knots provided, compute them from data
        probs <- seq(0, 1, length = nknots + 2)[2:(nknots + 1)]
        knots <- quantile(x, probs)
    }

    # create basis matrix
    X <- matrix(1, length(x), 1)
    for (j in 1:M) {
        X <- cbind(X, x^j)
    }

    # add truncated power basis terms
    for (k in 1:length(knots)) {
        X <- cbind(X, pmax(0, (x - knots[k]))^M)
    }

    return(list(X = X, knots = knots))
}

# generate spline basis for training data and store knots
spl_train <- lecturespl_fixed(train_data$Temp, nknots=2, M=4)
train_knots <- spl_train$knots

# generate spline basis for test data using training knots
spl_test <- lecturespl_fixed(test_data$Temp, nknots=2, M=4, knots=train_knots)

X <- spl_train$X
y <- train_data$Ozone

train_df <- data.frame(X)
colnames(train_df) <- paste0("X", 1:ncol(X))
fit <- lm(y ~ ., data=train_df)
train_pred <- predict(fit)

test_df <- data.frame(spl_test$X)
colnames(test_df) <- paste0("X", 1:ncol(spl_test$X))
test_pred <- predict(fit, newdata=test_df)

# create extended predictions using training knots
new_temp <- seq(0, 120, length.out = 200)
spl_new <- lecturespl_fixed(new_temp, nknots=2, M=4, knots=train_knots)

newdata <- as.data.frame(spl_new$X)
names(newdata) <- paste0("X", 1:ncol(newdata))
new_pred <- predict(fit, newdata=newdata)

plot(train_data$Temp, train_data$Ozone, main="Ozone vs Temperature with Fixed Knots", xlab="Temperature", ylab="Ozone", pch=16, col="darkgray", xlim=c(0, 120))
lines(train_data$Temp, train_pred, col="red", lwd=2)
lines(new_temp, new_pred, col="blue", lwd=2, lty=2)
legend("topleft", legend=c("Observed", "Fitted", "Extended Predictions"), col=c("darkgray", "red", "blue"), pch=c(16, NA, NA), lty=c(NA, 1, 2))

#
# fix low tempreatures with log-transformed response
#

X <- spl_train$X
y <- log(train_data$Ozone)  # log transform

train_df <- data.frame(X)
colnames(train_df) <- paste0("X", 1:ncol(X))
fit <- lm(y ~ ., data=train_df)

train_pred <- exp(predict(fit))  # transform back

spl_test <- lecturespl_fixed(test_data$Temp, nknots=2, M=4, knots=train_knots)
test_df <- data.frame(spl_test$X)
colnames(test_df) <- paste0("X", 1:ncol(spl_test$X))
test_pred <- exp(predict(fit, newdata=test_df))  # transform back

new_temp <- seq(0, 120, length.out = 200)
spl_new <- lecturespl_fixed(new_temp, nknots=2, M=4, knots=train_knots)
newdata <- as.data.frame(spl_new$X)
names(newdata) <- paste0("X", 1:ncol(newdata))
new_pred <- exp(predict(fit, newdata=newdata))  # transform back

plot(train_data$Temp, train_data$Ozone, main="Ozone vs Temperature (Log-transformed Model)",xlab="Temperature", ylab="Ozone", pch=16, col="darkgray", xlim=c(0, 120))
points(test_data$Temp, test_data$Ozone, pch=16, col="lightblue")
lines(train_data$Temp, train_pred, col="red", lwd=2)
lines(test_data$Temp, test_pred, col="blue", lwd=2)
lines(new_temp, new_pred, col="green", lwd=2, lty=2)
legend("topleft", legend=c("Training Data", "Test Data", "Training Fit", "Test Fit", "Extended Predictions"), col=c("darkgray", "lightblue", "red", "blue", "green"), pch=c(16, 16, NA,
```


Ozone vs Temperature with Fixed Knots
Observed
Fitted
Extended Predictions
Ozone
Temperature
0
5
10
15
20
25
30
35
0
20
40
60
80
100
120


Ozone vs Temperature (Log−transformed Model)
Training Data
Test Data
Training Fit
Test Fit
Extended Predictions
Ozone
Temperature
0
5
10
15
20
25
30
35
0
20
40
60
80
100
120