Assignment 4

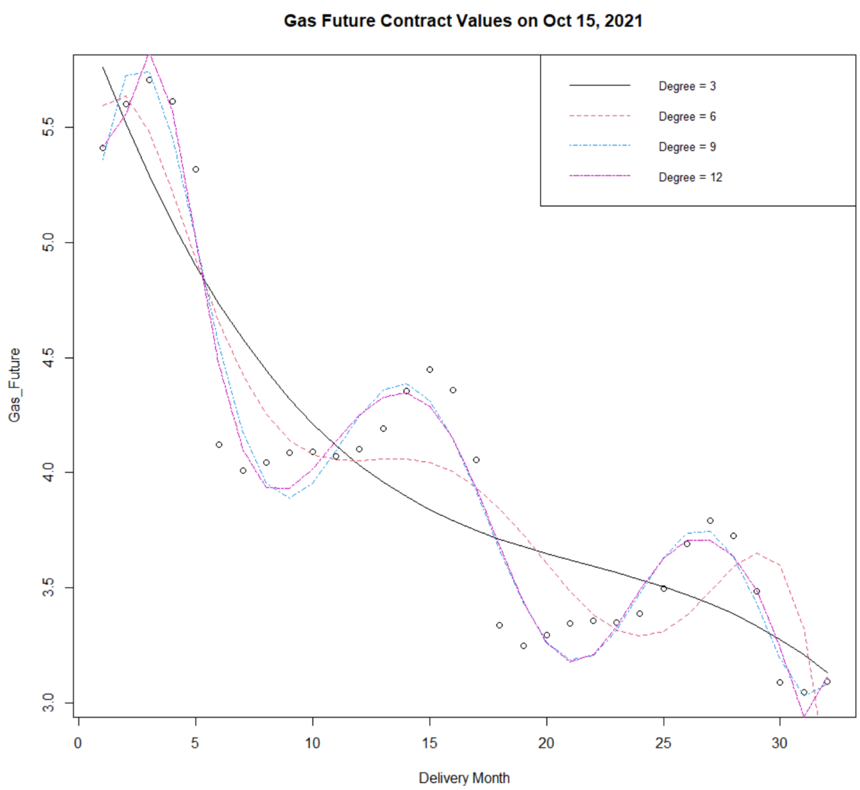
1) Gas Futures Analysis

1.1)

```
Gas_Future <- read.table("GAS_Future_2021_Oct15.csv",header = T, sep=",")
attach(Gas_Future)
head(Gas_Future)
```

```
# 1.1 Polynomial LS Regression

  # empirical curve
Gas_Future <- SETTLE[1:32]
Delivery_month <- seq(1,32)
plot(Delivery_month, Gas_Future, xlab="Delivery Month", main = "Gas Future Contract Values on Oct 15, 2021")

  # polynomial regression

lines(Delivery_month,fitted(lm(Gas_Future ~ poly(Delivery_month, 3))),lty = 1, col = 1,lwd=1.5)
lines(Delivery_month,fitted(lm(Gas_Future ~ poly(Delivery_month, 6))),lty = 2, col = 2,lwd=1.5)
lines(Delivery_month,fitted(lm(Gas_Future ~ poly(Delivery_month, 9))),lty = 4, col = 4,lwd=1.5)
lines(Delivery_month,fitted(lm(Gas_Future ~ poly(Delivery_month, 12))),lty = 6, col = 6,lwd=1.5)

legend("topright", c("Degree = 3", "Degree = 6", "Degree = 9","Degree = 12"), lty = c(1,2,4,6), col = c(1,2,4,6), lwd=c(1.5,1.5,1.5,1.5), cex = 0.75)
```

Polynomial LS Regression (p = 3, 6, 9, 12)

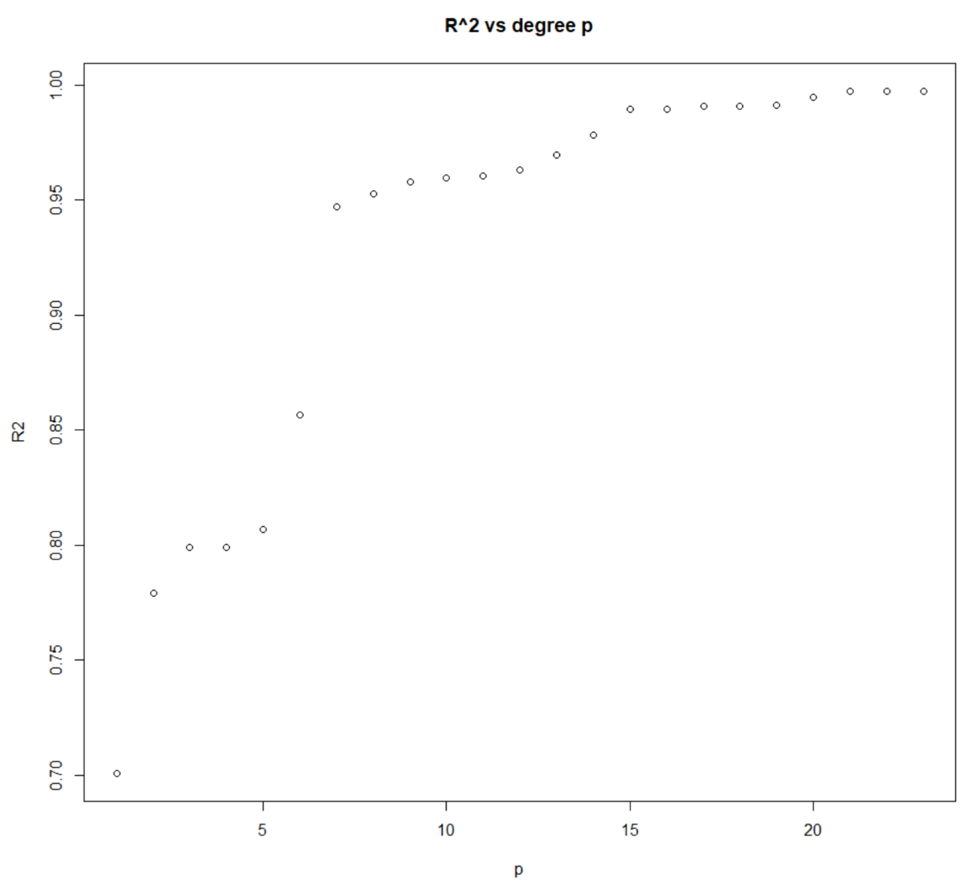


1.2)

Polynomial Degree Justification:

R^2

```
   # R-sqr
R2 <- rep(0,23)
for (i in 1:23)
{
  fit <- lm(Gas_Future ~ poly(Delivery_month,i))
  R2[i] <- summary.lm(fit)$r.squared
}

plot(R2,xlab="p",main="R^2 vs degree p")
```



Based on R^2, the selected degree is 7. Huge jumps in the R^2 would warrant the selection of that degree. From degree 6 to 7, R^2 increased from 0.85 to 0.95. Afterwards, there are no more huge jumps in R^2, which means the degrees after 7 are negligible. We would want to pick a less complex model as well; therefore degree 7 is a good choice.

ANOVA

```
fit_6 <- lm(Gas_Future ~ poly(Delivery_month, 6, raw = TRUE))
fit_7 <- lm(Gas_Future ~ poly(Delivery_month, 7, raw = TRUE))
anova(fit_6, fit_7)

Analysis of Variance Table

Model 1: Gas_Future ~ poly(Delivery_month, 6, raw = TRUE)
Model 2: Gas_Future ~ poly(Delivery_month, 7, raw = TRUE)
  Res.Df     RSS Df Sum of Sq      F    Pr(>F)
1     25 2.69373
2     24 0.99394  1    1.6998 41.044 1.265e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Based on the determined degree based on the analysis of R^2, the ANOVA of degree 6 and 7 will determine whether they are significant from each other. The low p-value (1.265×10^{-6}) of the F statistics indicate that degree 7 and 6 are indeed significant from each other.

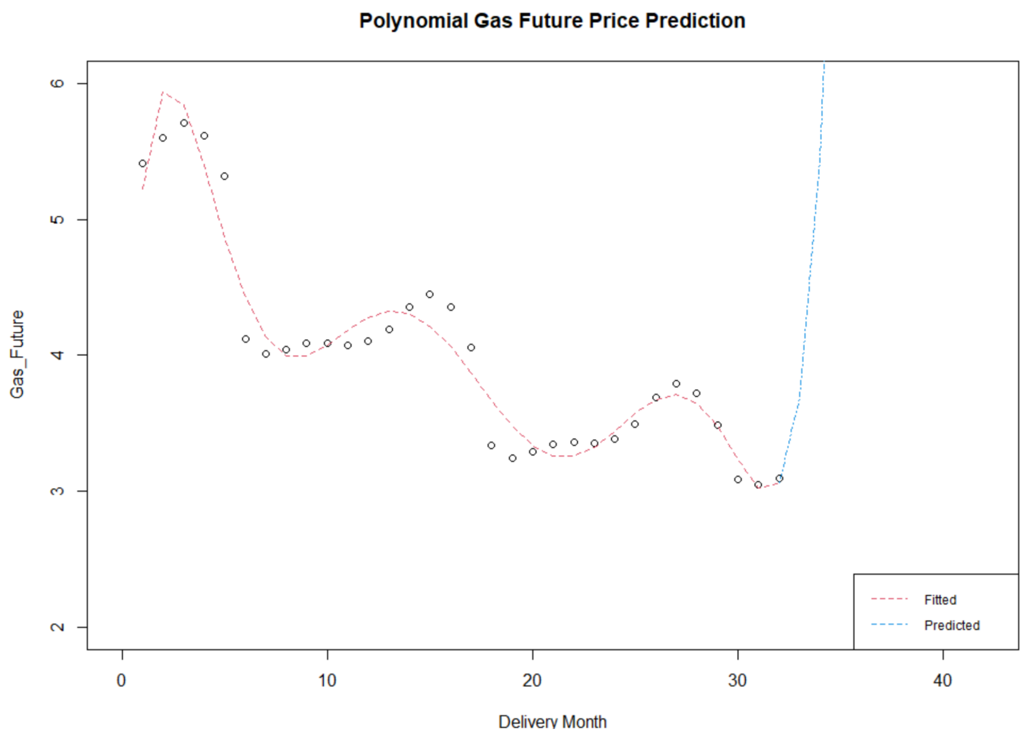
```
fit_8 <- lm(Gas_Future ~ poly(Delivery_month, 8, raw = TRUE))
anova(fit_7, fit_8)

Analysis of Variance Table

Model 1: Gas_Future ~ poly(Delivery_month, 7, raw = TRUE)
Model 2: Gas_Future ~ poly(Delivery_month, 8, raw = TRUE)
  Res.Df     RSS Df Sum of Sq      F Pr(>F)
1     24 0.99394
2     23 0.88328  1   0.11065 2.8813 0.1031
```

Additionally, an ANOVA of degree 7 and 8 shows that they are not significant and are equal to each other. Based on the high p-value (0.1031) of the F statistics, they are not significant.

Therefore, degree 7 is a good choice for the polynomial regression model because it can increase the R^2 significantly and is also significant from the lower order degrees.

1.3)

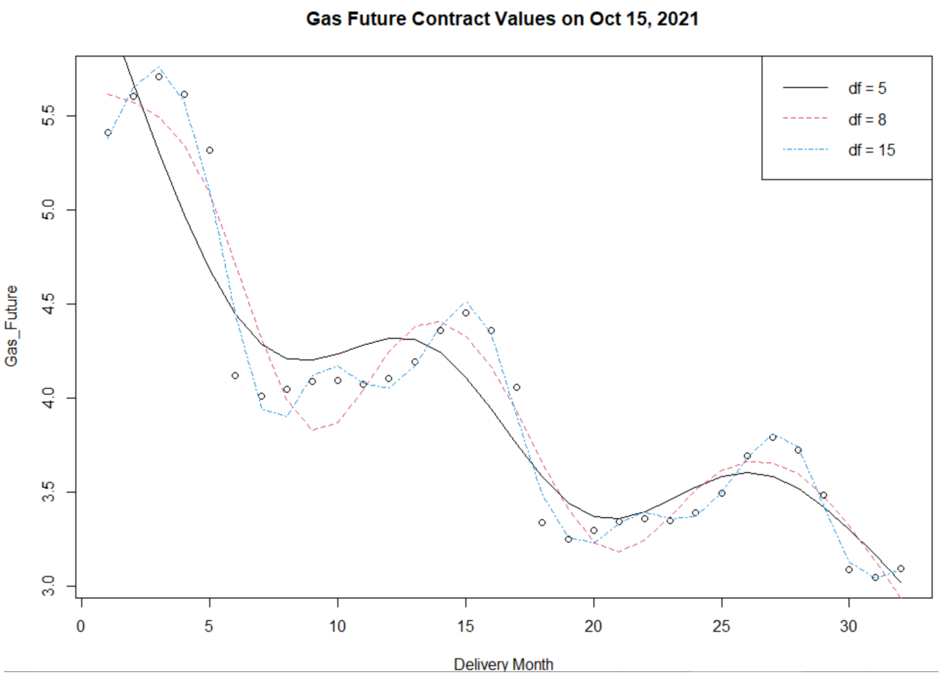
```
# 1.3 Prediction Future Prices (t+6)

p <- 7
I <- seq(32,38)
T<-length(I)
coef <- lm(Gas_Future ~ poly(Delivery_month,  degree = p, raw = TRUE))$coef
pred <- rep(0,T)
for (i in 1:T)
{
  pred[i] <- sum(coef * I[i]^(0:p))
}

plot(Gas_Future,xlim=c(0,42),ylim=c(2,6),xlab="Delivery Month",  main = "Polynomial Gas Future Price Prediction")
lines(Delivery_month,fitted(lm(Gas_Future ~ poly(Delivery_month, p))), lty = 2, col = 2,lwd=1.5)
lines(I,pred, lty = 4, col = 4,lwd=1.5)
legend("bottomleft", c("Fitted", "Predicted"), lty = c(2,2), col = c(2,4), lwd=c(1.5,1.5))
```



The prediction of the future price for t+6 is unreliable. Graphically, the predicted line does not even follow the trend of the data points. The reason is because at degree 7, the prediction at time 38 is 38^7, which is an enormous number. The model cannot be used to predict the future prices.

1.4)

```r
# 1.4 Natural Splines

library(splines)

  # empirical curve
plot(Delivery_month, Gas_Future, xlab="Delivery Month", main = "Gas Future Contract Values on Oct 15, 2021")

  # natural splines
n <- length(Gas_Future)
x <- seq(1,n)
lines(x,fitted(lm(Gas_Future~ns(x,df=5))),col=1,lwd=1.5)
lines(x,fitted(lm(Gas_Future~ns(x,df=8))),lty=2,col=2,lwd=1.5)
lines(x,fitted(lm(Gas_Future~ns(x,df=15))),lty=4,col=4,lwd=1.5)
legend("topright", c("df = 5", "df = 8", "df = 15"), lty = c(1,2,4), col = c(1,2,4), lwd=c(1.5,1.5,1.5))
```

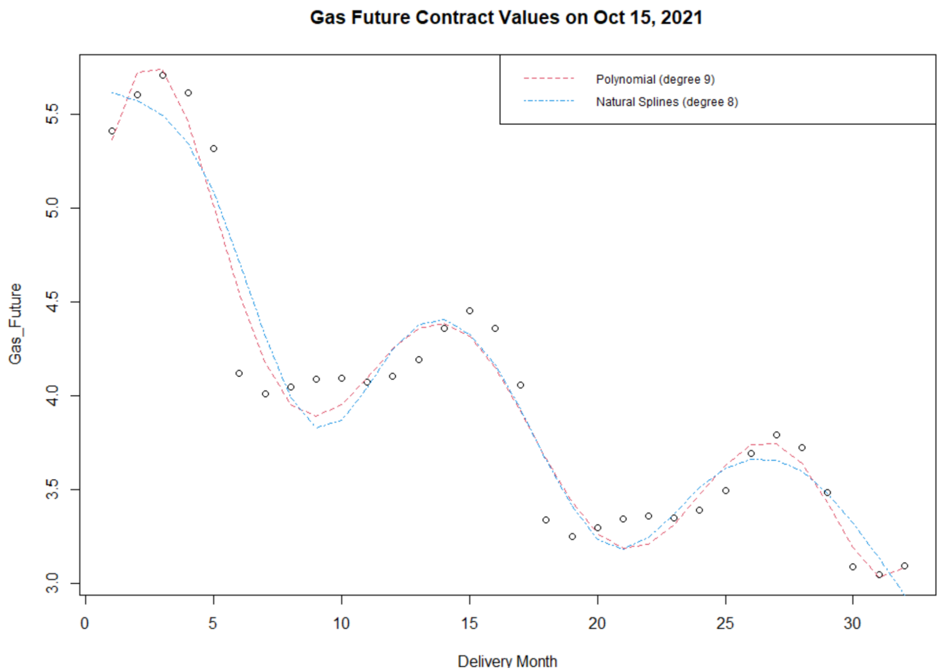


1.5)

Comparison of Polynomial Regression (p = 9) and Natural Splines (df = 8):

Fit

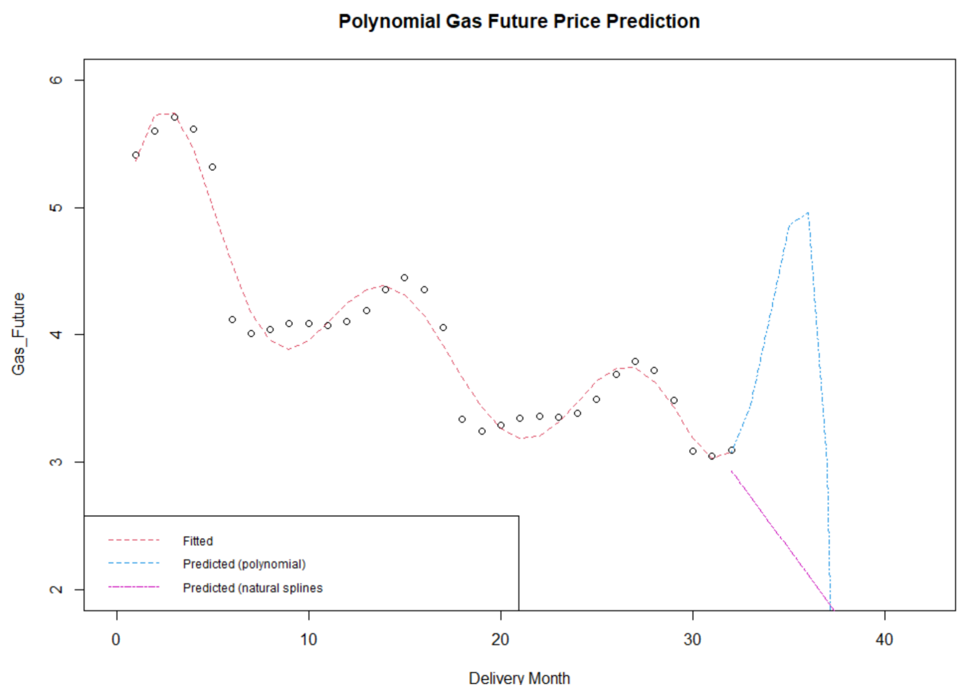
```r
  # comparison - fit
plot(Delivery_month, Gas_Future, xlab="Delivery Month", main = "Gas Future Contract Values on Oct 15, 2021")
lines(Delivery_month,fitted(lm(Gas_Future ~ poly(Delivery_month, 9))),lty = 2, col = 2,lwd=1.5)
lines(x,fitted(lm(Gas_Future~ns(x,df=8))),lty=4,col=4, lwd=1.5)

legend("topright", cex = 0.75, c("Polynomial (degree 9)", "Natural Splines (degree 8)"), lty = c(2,4), col = c(2,4), lwd=c(1.5,1.5))
```



The fit of both models are quite similar and are able to model the data points efficiently.

<u>Forecast</u>

```r
  # comparison - forecast

  # polynomial
p <- 9
I <- seq(32,38)
T<-length(I)
coef <- lm(Gas_Future ~ poly(Delivery_month,  degree = p, raw = TRUE))$coef
pred <- rep(0,T)
for (i in 1:T)
{
  pred[i] <- sum(coef * I[i]^(0:p))
}

plot(Gas_Future,xlim=c(0,42),ylim=c(2,6),xlab="Delivery Month",  main = "Polynomial Gas Future Price Prediction")
lines(Delivery_month,fitted(lm(Gas_Future ~ poly(Delivery_month, p))), lty = 2, col = 2,lwd=1.5)
lines(I,pred, lty = 4, col = 4,lwd=1.5)

  # natural splines
ns_8 <- lm(Gas_Future~ns(x,df=8))
pred_ns <- predict(ns_8, newdata=data.frame(x=I))
lines(I,pred_ns, lty = 6, col = 6,lwd=1.5)

legend("bottomleft", cex = 0.75, c("Fitted", "Predicted (polynomial)", "Predicted (natural splines"), lty = c(2,2, 6), col = c(2,4, 6), lwd=c(1.5,1.5,1.5))
```



However, the forecasts of both the models are problematic. Both the models do not account for trend and seasonality correctly and cannot be used for prediction.