Splines vs. polynomes for fitting non-linear relationships

Brice Ozenne

March 8, 2021

Note: this is document is inspired from http://stackoverflow.com/questions/15837763/b-spline-confusion

1 Simulate data

```
library(splines)
library(data.table)
library(ggplot2)
library(mgcv)

set.seed(1)
n <- 400

x <- 0:(n-1)/(n-1)
dt <- data.table(X = x,
   Ytrue =  0.2*x^11*(10*(1-x))^6+10*(10*x)^3*(1-x)^10)
dt[,Y := Ytrue + rnorm(n, 0, sd = 0.5)]
```

2 Apply spline transformation to X

```
dt[,c(X2,X3,X4,X5,X6) := .(X^2, X^3,X^4,X^5,X^6)]

SplineTempo <- bs(dt$X, knots = c(0.2, 0.5, 0.7))
dt <- cbind(dt,
     setNames(as.data.frame(SplineTempo),
       paste0("S",1:ncol(SplineTempo))))
```

3 Fit models

```
lmPoly <- lm(Y ~ X + X2 + X3 + X4 + X5 + X6, data = dt)

lmSpline <- lm(Y ~ bs(x, knots = c(0.2, 0.5, 0.7)), data = dt)

lmSplineI <- lm(Y ~ S1 + S2 + S3 + S4 + S5 + S6, data = dt)

autoSpline <- gam(Y ~ s(X), data = dt)
```

Note that

```
range(coef(lmSpline)-coef(lmSplineI)) # same as lmSpline
```

```
[1] 0 0
```

Residual degree of freedom:

```
c(df.residual(lmPoly),df.residual(lmSpline), df.residual(autoSpline))
```

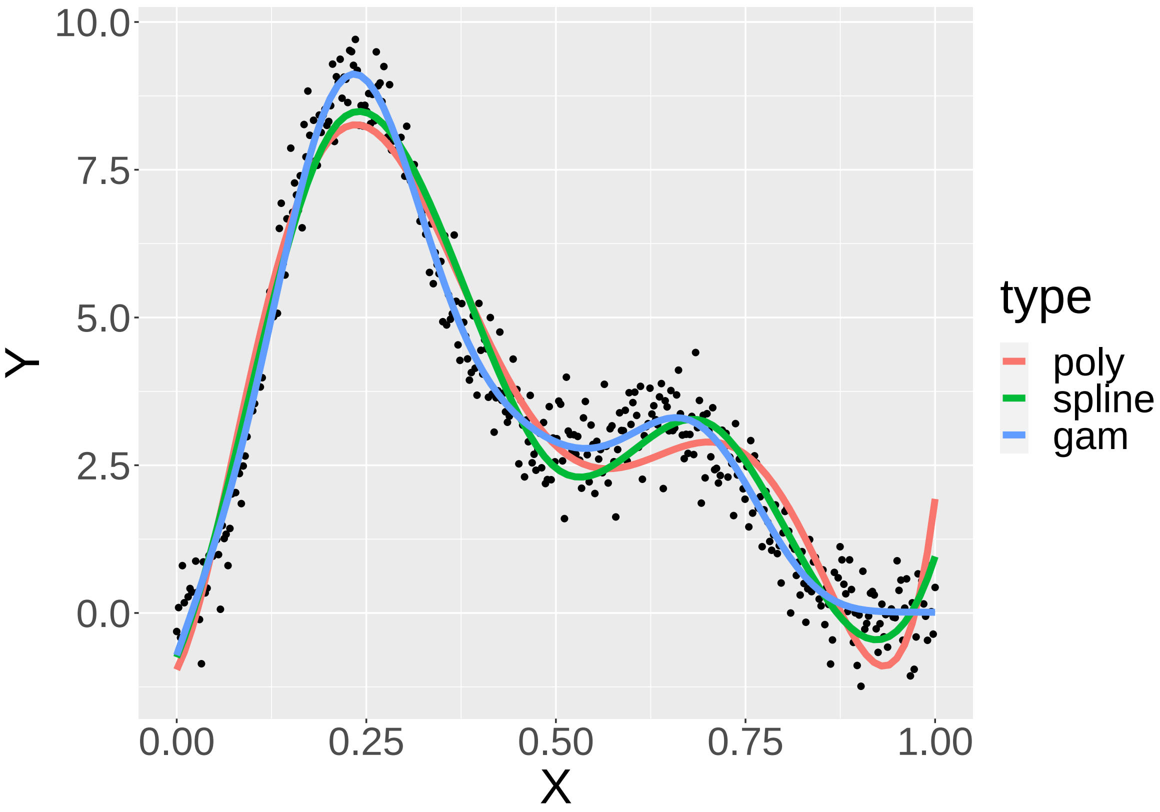
```
[1] 393.0000 393.0000 390.0559
```

4 Extract the fitted values

```
seqX <- seq(min(dt$X), max(dt$X), length = 100)

dt2 <- data.table(Y = dt$Y, X = dt$X, type = "observed")

predPoly <- predict(lmPoly, newdata = data.frame(X = seqX, X2 = seqX^2, X3 =
    seqX^3, X4 = seqX^4, X5 = seqX^5, X6 = seqX^6))
dt2 <- rbind(dt2, data.frame(Y = predPoly, X = seqX, type = "poly"))

predSpline <- predict(lmSpline, newdata = data.frame(x = seqX))
dt2 <- rbind(dt2, data.frame(Y = predSpline, X = seqX, type = "spline"))

predGam <- predict(autoSpline, newdata = data.frame(X = seqX))
dt2 <- rbind(dt2, data.frame(Y = predGam, X = seqX, type = "gam"))
```

5 Display fit

```
ggbase <- ggplot(dt2[dt2$type == "observed",], aes(x = X, y = Y)) + geom_
   point()
ggbase <- ggbase + geom_line(data = dt2[dt2$type != "observed",],
        aes(x = X, y = Y, group = type, color = type),
        size = 2)
ggbase <- ggbase + theme(text = element_text(size=30))
```



Splines give a better fit compared to a 3rd order polynomial when the knots are correctly placed