Polynomial Regression for ARC Adjustment

Abdullah Azzam

2025-07-01

Table of contents

1 Relationship between ARC II and ARCs I and III

1.1 Data

Table 10-1 as dataframe

```r
# ARC II values
arc_ii <- c(100:61, 60:30, 25, 20, 15, 10, 5, 0)

# Corresponding ARC I values
arc_i <- c(
  100, 97, 94, 91, 89, 87, 85, 83, 81, 80,
   78, 76, 75, 73, 72, 70, 68, 67, 66, 64,
   63, 62, 60, 59, 58, 57, 55, 54, 53, 52,
   51, 50, 48, 47, 46, 45, 44, 43, 42, 41,
   40, 39, 38, 37, 36, 35, 34, 33, 32, 31,
   31, 30, 29, 28, 27, 26, 25, 25, 24, 23,
   22, 21, 21, 20, 19, 18, 18, 17, 16, 16,
   15, 12,  9,  6,  4,  2,  0
)

# Corresponding ARC III values
arc_iii <- c(
  100,100, 99, 99, 99, 98, 98, 98, 97, 97,
```

```
   96, 96, 95, 95, 94, 94, 93, 93, 92, 92,
   91, 91, 90, 89, 89, 88, 88, 87, 86, 86,
   85, 84, 84, 83, 82, 82, 81, 80, 79, 78,
   78, 77, 76, 75, 75, 74, 73, 72, 71, 70,
   70, 69, 68, 67, 66, 65, 64, 63, 62, 61,
   60, 59, 58, 57, 56, 55, 54, 53, 52, 51,
   50, 43, 37, 30, 22, 13,  0
)

df_cn <- data.frame(
  arc_ii = arc_ii,
  arc_i = arc_i,
  arc_iii = arc_iii
)

# quick check
head(df_cn, 10)
```

```
   arc_ii arc_i arc_iii
1     100   100     100
2      99    97     100
3      98    94      99
4      97    91      99
5      96    89      99
6      95    87      98
7      94    85      98
8      93    83      98
9      92    81      97
10     91    80      97
```

1.2 linear models: arc ii vs i, and ii vs iii

Let's try the simplest model (linear):

$$\text{ARC I} = \alpha_1 + \beta_1\,(\text{ARC II}),$$
$$\text{ARC III} = \alpha_3 + \beta_3\,(\text{ARC II}).$$

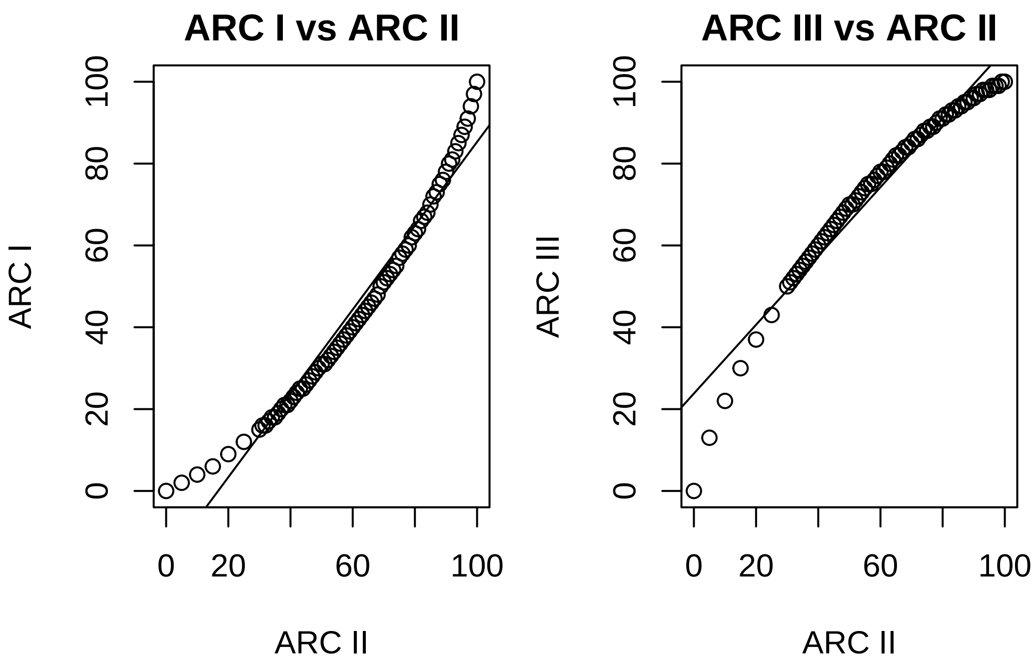
```
mod_i   <- lm(arc_i   ~ arc_ii,  data = df_cn)
mod_iii <- lm(arc_iii ~ arc_ii,  data = df_cn)

ci <- coef(mod_i)
c3 <- coef(mod_iii)

cat("ARC I =", round(ci[2],4), "* ARC II", if(ci[1]>=0) "+" else "-", abs(round(ci[1],4)), "\n"
```

```
ARC I = 1.0243 * ARC II - 17.1276
```

```r
cat("ARC III =", round(c3[2],4), "* ARC II", if(c3[1]>=0) "+" else "-", abs(round(c3[1],4)), "
```

```
ARC III = 0.8403 * ARC II + 23.8025
```

```r
par(mfrow=c(1,2), mar=c(4,4,2,1))
plot(df_cn$arc_ii, df_cn$arc_i,   main="ARC I vs ARC II",
     xlab="ARC II", ylab="ARC I");   abline(mod_i)
plot(df_cn$arc_ii, df_cn$arc_iii, main="ARC III vs ARC II",
     xlab="ARC II", ylab="ARC III"); abline(mod_iii)
```



1.3 Polynomial Regression

Fit raw-power polynomials of degree (d = 1 …4):

$$\text{ARC I}(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4,$$

$$\text{ARC III}(x) = \gamma_0 + \gamma_1 x + \gamma_2 x^2 + \gamma_3 x^3 + \gamma_4 x^4,$$

where (x = ARC II). Select the best (d) by minimizing AIC.

```r
fit_and_evaluate <- function(y, x, max_deg=4, data){
  fits <- lapply(1:max_deg, function(d){
    lm(as.formula(sprintf("%s ~ poly(%s,%d,raw=TRUE)", y, x, d)), data=data)
  })
  metrics <- data.frame(
    degree = 1:4,
    AIC    = sapply(fits, AIC),
    adj_R2 = sapply(fits, function(m) summary(m)$adj.r.squared)
  )
  best <- metrics$degree[which.min(metrics$AIC)]
  list(metrics=metrics, best=best, fit=fits[[best]])
}

res_i   <- fit_and_evaluate("arc_i",   "arc_ii", data=df_cn)
res_iii <- fit_and_evaluate("arc_iii", "arc_ii", data=df_cn)

res_i$metrics
```

```
  degree      AIC    adj_R2
1      1 476.8652 0.9584608
2      2 252.4888 0.9977742
3      3 174.7730 0.9991987
4      4 109.2585 0.9996620
```

```r
res_iii$metrics
```

```
  degree       AIC    adj_R2
1      1 463.94746 0.9483609
2      2 217.60273 0.9979198
3      3 143.34330 0.9992168
4      4  75.66948 0.9996787
```

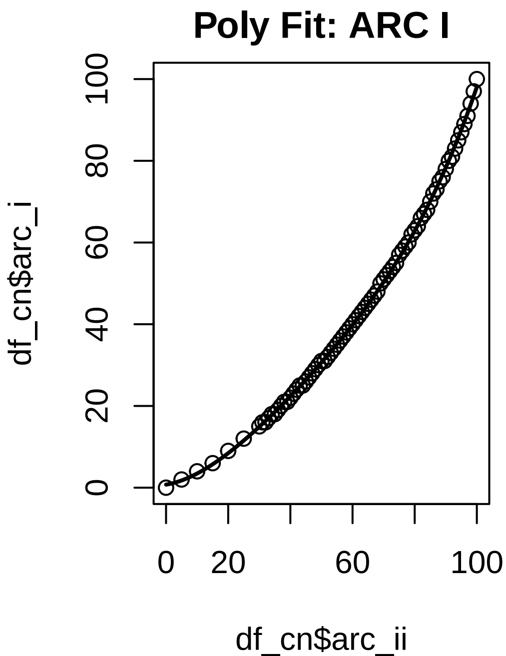
```r
cat("Best degree for ARC I   =", res_i$best,   "\n")
```

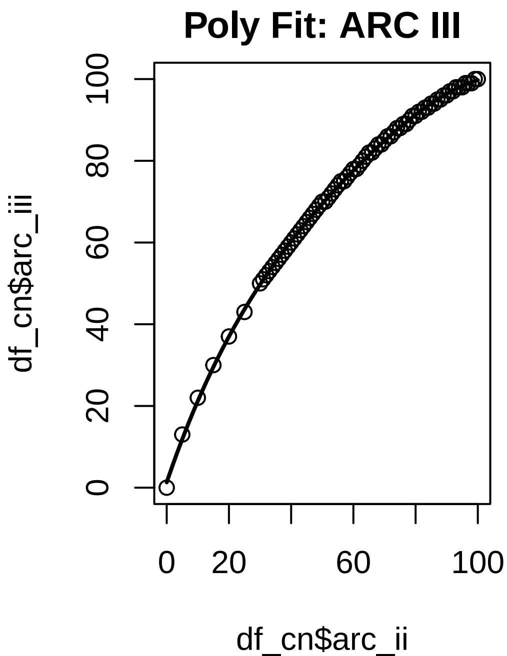
```
Best degree for ARC I   = 4
```

```r
cat("Best degree for ARC III =", res_iii$best, "\n")
```

```
Best degree for ARC III = 4
```

```r
xx <- seq(0,100,length=200)
par(mfrow=c(1,2), mar=c(4,4,2,1))
plot(df_cn$arc_ii, df_cn$arc_i, main="Poly Fit: ARC I")
lines(xx, predict(res_i$fit,   newdata=data.frame(arc_ii=xx)), lwd=2)
plot(df_cn$arc_ii, df_cn$arc_iii, main="Poly Fit: ARC III")
lines(xx, predict(res_iii$fit, newdata=data.frame(arc_ii=xx)), lwd=2)
```





1.4 Final Degree-4 Equations

A 4th-degree polynomial minimizes AIC in both cases. Extracting and ordering the coefficients gives:

$$\text{ARC I}(x) = 1.2211 \times 10^{-6}x^4 - 2.0260 \times 10^{-4}x^3 + 1.6526 \times 10^{-2}x^2 + 0.12577x + 0.74712,$$

$$\text{ARC III}(x) = -1.0054 \times 10^{-6}x^4 + 2.5082 \times 10^{-4}x^3 - 2.7615 \times 10^{-2}x^2 + 2.24312x + 1.30209.$$

```
mod_i4   <- lm(arc_i   ~ arc_ii + I(arc_ii^2) + I(arc_ii^3) + I(arc_ii^4), df_cn)
mod_iii4 <- lm(arc_iii ~ arc_ii + I(arc_ii^2) + I(arc_ii^3) + I(arc_ii^4), df_cn)

poly_i   <- coef(mod_i4)[c("I(arc_ii^4)","I(arc_ii^3)","I(arc_ii^2)","arc_ii","(Intercept)")]
poly_iii <- coef(mod_iii4)[c("I(arc_ii^4)","I(arc_ii^3)","I(arc_ii^2)","arc_ii","(Intercept)")]
poly_i
```

```
  I(arc_ii^4)   I(arc_ii^3)   I(arc_ii^2)        arc_ii   (Intercept)
 1.221107e-06 -2.025995e-04  1.652565e-02  1.257686e-01  7.471240e-01
```

```
poly_iii
```

```
  I(arc_ii^4)   I(arc_ii^3)   I(arc_ii^2)        arc_ii   (Intercept)
-1.005440e-06  2.508244e-04 -2.761460e-02  2.243122e+00  1.302090e+00
```

1.5 Using the conversion function

```r
predict_arc_i <- function(x) {
  round((sapply(x, function(xi) sum(poly_i * xi^(4:0)))),0)
#  ceiling((sapply(x, function(xi) sum(poly_i * xi^(4:0)))))
}
predict_arc_iii <- function(x) {
  round((sapply(x, function(xi) sum(poly_iii* xi^(4:0)))),0)
}
#predict_arc_i(c(30, 50, 75))
#predict_arc_iii(c(30, 50, 75))
#predict_arc_iii(c(72, 57, 88, 91, 45, 34, 66))
predict_arc_i(c(100, 97, 91, 88, 84, 79, 72, 68, 63, 55, 50, 44, 32))
```

```
 [1] 98 92 80 75 69 61 53 48 43 35 31 26 16
```

```r
predict_arc_iii(c(97, 91, 88, 84, 79, 72, 68, 63, 55, 50, 44, 32))
```

```
 [1] 99 97 95 93 91 86 84 80 74 69 64 52
```